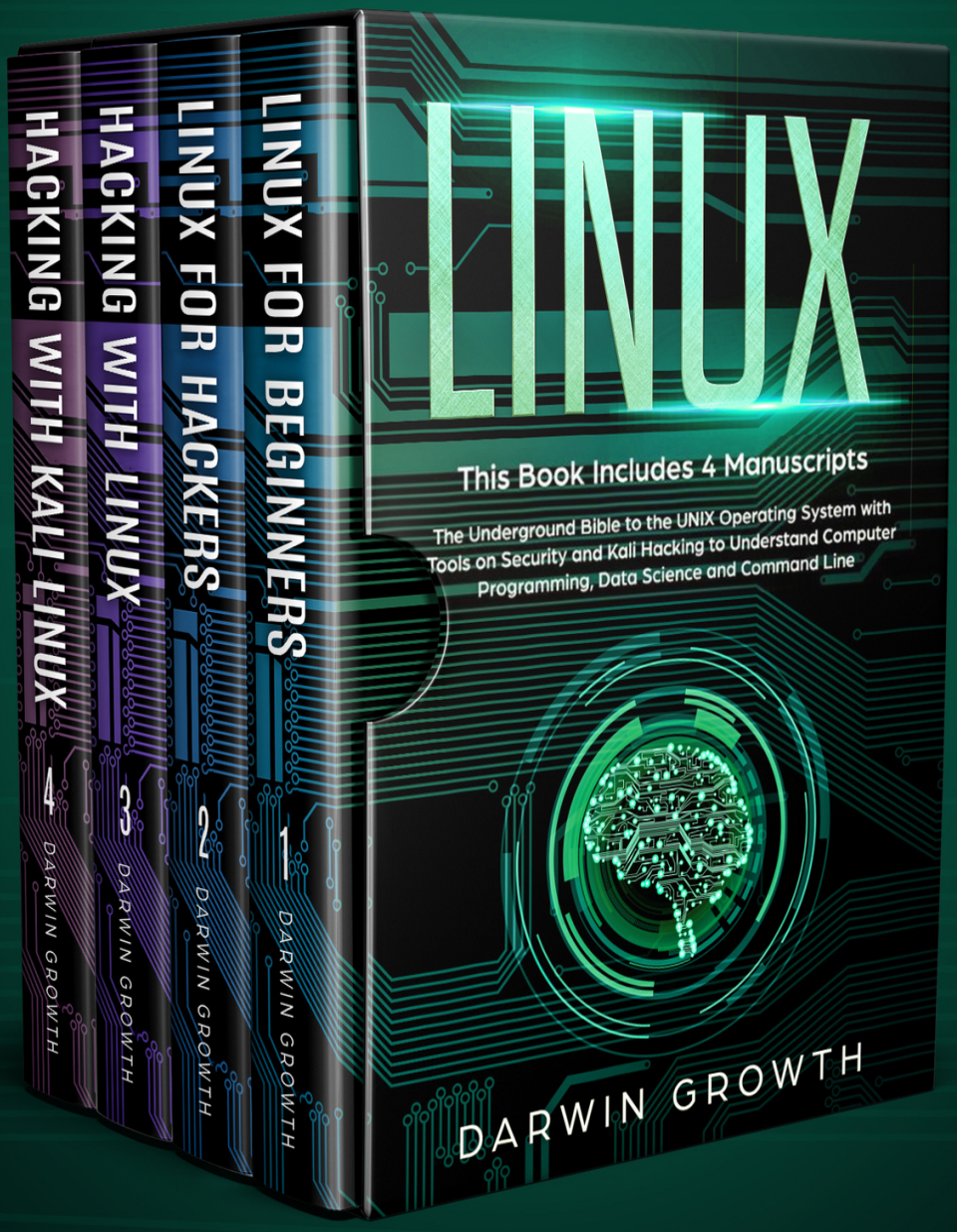
LINUX

This Book Includes 4 Manuscripts

The Underground Bible to the UNIX Operating System with Tools On Security and Kali Hacking to Understand Computer Programming, Data Science and Command Line

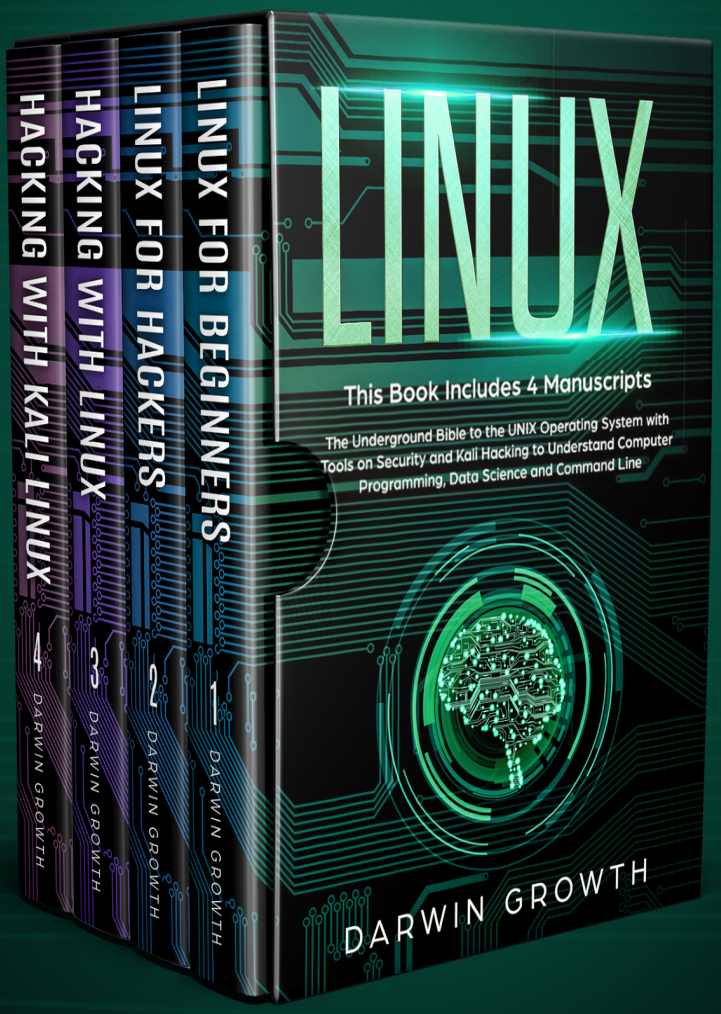


DARWIN GROWTH

LINUX

This Book Includes 4 Manuscripts

The Underground Bible to the UNIX Operating System with Tools On Security and Kali Hacking to Understand Computer Programming, Data Science and Command Line



DARWIN GROWTH

Linu x

This Book Includes 4 Manuscripts. The Underground Bible to the UNIX Operating System with Tools on Security and Kali Hacking to Understand Computer Programming, Data Science and Command Line

Darwin Growth

© Copyright 2019 Darwin Growth - All rights reserved .

The content contained within this book may not be reproduced, duplicated or transmitted without direct written permission from the author or the publisher.

Under no circumstances will any blame or legal responsibility be held against the publisher, or author, for any damages, reparation, or monetary loss due to the information contained within this book. Either directly or indirectly.

Legal Notice:

This book is copyright protected. This book is only for personal use. You cannot amend, distribute, sell, use, quote or paraphrase any part, or the content within this book, without the consent of the author or publisher.

Disclaimer Notice:

Please note the information contained within this document is for educational and entertainment purposes only. All effort has been executed to present accurate, up to date, and reliable, complete information. No warranties of any kind are declared or implied. Readers acknowledge that the author is not engaging in the rendering of legal, financial, medical or professional advice. The content within this book has been derived from various sources. Please consult a licensed professional before attempting any techniques outlined in this book.

By reading this document, the reader agrees that under no circumstances is the author responsible for any losses, direct or indirect, which are incurred as a result of the use of information contained within this document, including, but not limited to, — errors, omissions, or inaccuracies.

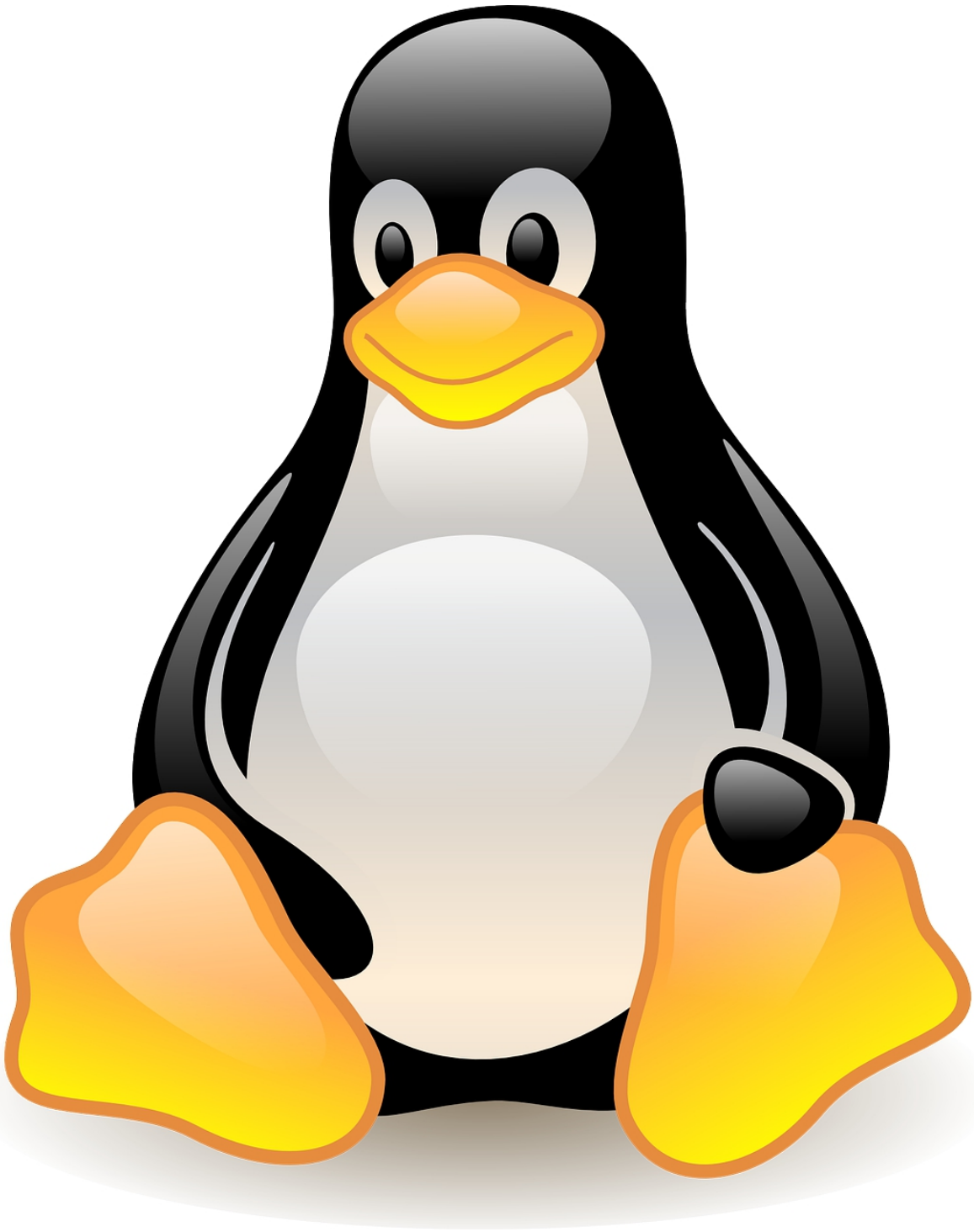


Table of Contets

Linux for Beginners

Introduction

Linux File System Structure
Links
The Path

Chapter 1: Basic Operating System Concepts

The Concept Of The Operating System, Its Purpose, And Function
Classification Of Modern Operating Systems
Functional Components Of Operating Systems
I / O Management
File Management And File Systems
Network Support
Data security
User Interface
Conclusions

Chapter 2: Operating System Architecture

Basic Concepts Of Operating System Architecture
System Kernel - Privileged Mode And User Mode
Implementation Of Operating System Architecture
Monolithic Systems
Multilevel Systems
Micronucleus Systems
The Concept Of Virtual Machines
Architecture Features: UNIX and Linux
The Purpose Of The Linux Kernel And Its Features
Kernel Modules
Features Of System Libraries
User Application
Installation
System Configuration
Conclusions

Chapter 3: Process and Flow Management

Basic concepts of processes and flows

Process and Flow Models
Process and Flow Components
Process And Flow States
Description Of Processes And Flows
Managing Process And Flow Blocks
Process And Flow Images
Creating and Completing Processes and Threads
Managing Address Space When Creating Processes
Run The Application With One System Call
Features Of Process Completion
Synchronous And Asynchronous Execution Of Processes
Creating and Completing Streams
Features Completion Streams
Managing Processes In UNIX and Linux
Identification Information And Process Security Attributes
Process Control Unit
Creation Process
Completing The Process
Waiting For Process To Complete
5/3 7 Alerts
Types Of Signals
Layout Of Signals
Signal Blocking
Setting The Signal Layout
Signal Generation
Organization Of Asynchronous Execution Of Processes
Managing Threads in Linux
The Disadvantages Of Traditional Multitasking Support For Linux
Linux Kernel Threads
The Flow Management Software Interface
Completion of POSIX Streams
Waiting For Completion Of Threads
Conclusions

Chapter 4: Process and Flow Planning

General Principles Of Planning
Planning Mechanisms and Policy
Planning Principles

Types Of Planning
Long-Term Planning
Medium-Term Planning
Short-Term Planning
Planning Strategies - Displacement And Non-Displacement Multitasking
Planning Algorithms
FIFO Planning
Circular Planning
Priority Planning
Planning Based On Performance Characteristics
Multilevel Feedback Queues
Lottery Planning
Implementation Of Planning In Linux
Real-Time Scheduling Of Kernel Processes
Traditional Scheduling Algorithm
Planning Procedure Call Conditions
Planning Procedure
The Beginning Of A New Era
Dynamic Priority Calculation
Recounting A Quantum When Creating A New Process
Modern Approaches To Planning Implementation
The Programming Interface Of Planning
Conclusions

Chapter 5: Flow Interaction

Basic Principles Of Flow Interaction
The Main Problems Of Flow Interaction
Critical Sections And Locks
Locking
Problems With Implementation Of Blocking
Basic Mechanisms Of Flow Synchronization
Semaphores
Features Of Use Of Semaphores
Realization Of The Problem Of Producers-Consumers With The Help Of Semaphores
Mutexes
Interprocess
Types Of Interprocess Interaction

Conclusions
Chapter 6: Managing Memory
Fundamentals of Virtual Memory Technology
The Concept Of Virtual Memory
Problems of virtual memory implementation: Memory fragmentation
Logical And Physical Memory Addressing
Approach Of Base And Boundary Registers
Memory Segmentation
Segmentation Implementation In IA-32 Architecture
Page Memory Organization
Basic Principles Of Page Organization Of Memory
Comparative Analysis Of Page Organization Of Memory And Segmentation
Multi-Level Page Tables
Implementation Of Page Tables In IA-32 Architecture
Associative Memory
Conclusion
Linux for Hackers
Introduction
Chapter 1: Preview of the Linux Operating SystemPreview of How Linux Works
Relationship Between the Applications, Hardware, and Kernels
Benefits of Using Linux
Chapter 2: Hacking and the Skills Hackers Need
What is Hacking?
Types of Hacking Techniques and Precautions to Take
Hacking Skills with the Linux System
How to Protect Yourself When Hacking
Chapter 3: Linux Distributions and Backup File System
How to Use the Linux Operating System
What is Linux Distribution?
Types of Linux Distributions
Directories in Linux
Backup Linux File Systems
Chapter 4: Text Manipulation

How Linux is Useful in Text Manipulation
Why Use Text Manipulation
Top Seven Linux command Line tools for text manipulation
Managing networks
Dynamic host configuration protocol (DHCP)
Process of hacking with kali Linux
Installation of Kali Linux
Storage management
Block storage
Chapter 5: Detailed Overview of Linux and How to Hack with Linux
How does Linux work?
How to hack and what you need
Python scripting for the hacker
Why is Kali Linux the best for starters
Bash scripting
Automatic tasks
Chapter 6:
Process of Web Hacking
Web hacking
Penetration testing
Conclusion

Hacking with Linux

Introduction
Chapter 1: Basic and Advanced Linux Concepts
Advanced Linux Concepts
Chapter 2: Linux Installation
Kali Linux Installation Requirements
The Installation Process
Chapter 3: Bash and Python Scripting
Different Types of Hacking
How to be Secure and Anonymous while Hacking
Chapter 4: Ethical and Unethical Hacking
Hackers Hierarchy (Using and Abusing Services)
Chapter 5: Servers and Networks

Chapter 6: Cyber-attacks and Malware
Chapter 7: Cryptography
Advantages of Cryptography
Types of Cryptography
Chapter 8: Wireless and Network Exploitation
The Dark Web Explained
Accessing the Dark Web
Conclusion
Hacking with Kali Linux
Chapter 1: Introduction to Linux and Hacking
What is Linux?
Why is Linux the Best Operating System?
What is Hacking?
Who are Hackers?
Installation of Linux Distros
Chapter 2: Basic Linux Commands
How to Find Help while Using Linux?
User Management in Linux
File Management in Linux
Process Management System
Chapter 3: Basic Shell Programming
What is a Shell?
Built-in Shell Commands
Fundamentals of Shell Programming
Chapter 4: Hacking Procedure
Chapter 5: Web Hacking Tools
Scanning of Webservers
Hacking a WordPress Website
Chapter 6: Network Hacking Tools
What is a Network?
What is Routing?
Chapter 7: Web Hierarchies and Cybersecurity Ethics
Why Do Hackers Fit Into Hierarchies?

Cybersecurity Ethics
What is Penetration Testing?
Chapter 8: TOR & VPN in Linux
How to Use the TOR Network in Kali Linux?
What is TOR?
Chapter 9: Advanced Kali Linux Hacking Tools
Burp Suite
Metasploit
Conclusion

Linux for Beginners

The Science of Linux Operating System and Programming Tools for Installation, Configuration and Command-Line with a Basic Guide on Networking, Cybersecurity, and Ethical Hacking

Darwin Growth

Introduction

Congratulations on choosing *Linux For Beginners* and thank you for doing so.

There are plenty of books on this subject on the market, thanks again for choosing this one! Every effort was made to ensure it is full of as much useful information as possible, please enjoy it!

There are three ways to access a Linux system:

- Through a text console. In this method the user connects directly to the computer that has Linux installed and then they can access it through a non-graphic system
- From a graphical session manager (X Window). Here the user connects directly to the computer that has Linux installed and accesses the system through a graphical program
- From a remote computer using telnet or secure shell.

In any of the previous situations, the following will appear (more or less):

- Login: (Username is typed)
- Password: (The password is typed, which is not visible on the screen)
    - For security reasons the password must meet certain conditions such as:
        - Contain at least six characters
        - Contain at least one numerical or special character and two alphabetic characters
        - Be different from the login name

The first time the system is accessed, the password used will be the one provided by the administrator of the system. There are several ways to end the work session in Linux, depending on whether we are in graphic mode or text.

In-text mode:

1. Press the <ctrl> d keys
2. Enter the exit command.

In graphic mode:

The output of X Window depends on the window manager that is running and will be explained later.

Linux File System Structure

Files: Types
The basis of the Linux filing system will include the file, which is nothing other than the structure used by the operating system to store information on a physical device such as a hard drive, a floppy disk, a CD-ROM or a DVD. Naturally, a file can contain any type of information that you would ilke, from an image in PNG or JPEG format to a text or a web page in HTML format. The file system is the structure that allows Linux to handle the files it contains .

All Linux files have a name, which must comply with certain rules:

- A file name can be between 1 and 255 characters
- You can use any character except the slash / and it is not recommended to use the characters with special meaning in Linux, which are the following: = \ ^ ~ "'` *; -? [] ()! & ~ <>
    - To use files with these characters or spaces you must enter the name of the file between quotation marks
- Numbers can also be used if desired.
- Uppercase and lowercase letters are considered different, and therefore letter.txt is not the same as Letter.txt or letter.Txt

As in Windows, a certain "type" criterion can be used to mark the different kinds of files using a series of characters at the end of the name that indicates the type of file in question. So, the text files, HTML, PNG or JPEG images have extensions .txt, .htm (or .html), .png and .jpg (or .jpeg) respectively.

Despite this, Linux only distinguishes three types of files:

- Files or ordinary files are those mentioned above
- Directories (or folders), is a special file that groups other files in a structured way

- Special files are the basis on which Linux sits since they represent the devices connected to a computer, such as a printer. In this way, enter information in that file is equivalent to sending information to the printer. For the user, these devices have the same appearance and use than ordinary files

Links

Links are an ordinary file type that have the goal to create a new name for a determined file. Once the symbolic link is created, it allows access to the file that links in the same way as if we had copied the contents of it to another file, with the advantage that it has not really been copied. The symbolic links are especially useful when you want a group of people to work on the same file since they allow sharing the file but centralizing the modifications.

As an example, you can assume the existence of a file called balance.1999.txt, to which a link is created symbolic balance.txt. Any access to balance.txt is translated by the system so that you access the balance content.1999.txt.

The Path

In any modern operating system, the file structure is hierarchical and depends on the directories. In general, the file system structure resembles a tree structure, each node being composed of a directory or folder, which contains other directories or files. In Windows, each disk drive is identified as a basic folder that serves as root to others. On Linux, there is a single root called / from which all files and directories hang, and that is independent of which devices are connected to the computer.

The path or path of a file or directory is the sequence of directories that must be traversed to access a certain file separated by /.

There are two forms of the path:

- The absolute path that shows the entire path to a file, /home/luis/Carta.txt

- The path relative to a certain directory, for example, if we cannot find the / home directory, the path relative to the file Letter.txt is Luis / Letter.txt

    - To complicate matters further, all directories contain two special directories:

- The current directory, represented by the point “."
- The parent directory represented by a colon ".."

Being in the directory/home you can access Carta.txt with /home/luis/Carta.txt (absolute path) or /luis/Carta.txt (relative path). In Luis as ./Carta.txt or simply Letter.txt.

The Linux file system follows all UNIX conventions, which means it has a structure determined, compatible and homogeneous with the rest of UNIX systems. Unlike in Windows or MS-DOS the file system on any Unix system is not directly linked to the hardware structure, that is, it does not depend on whether a given computer has 1, 2 or 7 hard drives to create the c: \, d: \ or m: \ drives.

Every UNIX file system has a unique root or root origin represented by /. Under this directory you will find all the files that the operating system can access. These files are organized in different directories whose mission and name are standard for all Unix systems.

- /: Root of the file system
- / dev: Contains system files representing the devices that are physically installed in the computer
- / etc: This directory is reserved for system configuration files

In this directory, no binary files (programs) should appear. Under this, two other subdirectories should appear:

- / etc / X11: X Window configuration files
- / etc / skel: Basic configuration files that are copied to the user's directory when one is created new
- / lib: Contains the libraries needed to run programs that reside in / bin (not libraries of user programs)

- / proc: Contains special files that either receive or send information to the system kernel (It recommends not modifying the contents of this directory and its files)
- / sbin: Contains programs that are only accessible to the superuser or root
- / usr: This is one of the most important directories of the system since it contains the usage programs common to all users. Its structure is usually similar to the following:
- / usr / X11R6: Contains the programs to run X Window
- / usr / bin: General purpose programs, which includes the C / C ++ compiler
- / usr / doc: General system documentation
- / usr / etc: General configuration files
- / usr / include: C / C ++ header files (.h)
- / usr / info: GNU information files
- / usr / lib: General libraries of the programs
- / usr / man: Manuals accessible with the man command
- / usr / sbin: System administration programs
- / usr / src: Source code of programs

In addition to the previous ones, there are other directories that are usually located in the / usr directory, such as the folders of the programs that are installed in the system.

- / var: This directory contains temporary information about the programs (which does not imply that you can delete its content, in fact, it should not be done)

Chapter 1:
Basic Operating System Concepts

The Concept Of The Operating System, Its Purpose, And Function

The Concept of An Operating System

The main idea and start of our modern day operating systems was the need to create ISG including them in the use of computer systems (By *computer cybernetic system,* we mean a set of hardware and software). The computer cybernetic system initially developed for liability solutions and practical problems of users. Because it was difficult to do this with hardware alone, applications were created. These programs required general operations of hardware management, distribution of hardware from resources, and the like. These operations are grouped under a separate layer of software, which is known as the operating system.

Further, the capabilities of operating systems went far beyond the basic set of operations required by applications, but the intermediate position of such systems between applications and hardware remained unchanged .

Purpose Of The Operating System

Operating systems provide us with a score of cybernetic system, and secondly, efficiency and reliability of its work. The first function is characteristic of the OS as an extended machine, the second - the OS as a distributor of hardware resources.

Operating System As An Extended Machine

Using the operating system, the application programmer (and through his programs and the user) should have the impression that they are working with an advanced machine. The hardware is not well adapted for direct use in applications. For example, if you consider working with I / O devices at the command level of the respective controllers, you can see that the set of such commands is limited, and for many devices - primitive. The operating system hides such *a hardware interface* but instead offers the programmer *an application programming interface* that uses higher-level concepts (called abstractions).

For example, when working with a disk, a typical abstraction is a file. it is easier to work with files than directly with a disk controller (no need to consider moving the drive heads, starting and stopping the motor, etc.), as a result, the programmer can focus on the essence of his application. The operating system is responsible for interacting with the disk controller .

Abstraction highlighting makes it easy for OS and application code to change when migrating to new hardware. For example, if you install a new type of disk device on your computer (provided that it is supported by the OS), all its features will be taken into account at the OS level, and applications will continue to use the files as before. This characteristic of the system is called *hardware independence.* OS can be said to provide a hardware-independent environment for executing applications.

Operating System As A Resource Allocator

The operating system must allocate resources efficiently. It acts as the manager of these resources and provides them to applications on demand. There are two main types of resource allocation. In the case of *the spatial distribution of* resource access will be for multiple customers simultaneously, and each one of them can use the resources (the shared memory). In the case of *temporal distribution, the* system queues and, according to it, allows them to use the entire resource for a limited time (so the processor is distributed in single-processor systems).

When allocating resources, the OS resolves possible conflicts, prevents unauthorized access of programs to those resources, on which they have no rights, ensures the efficient operation of the computer system .

Classification Of Modern Operating Systems

Consider the classification of modern operating systems, depending on their scope. First of all, note the OS of *large* computers (mainframes). The main characteristic of the hardware for which they are designed is the performance of I / O: large computers provide a large number of peripherals (disks, printers, terminals, etc.). Such a computer cybernetic system is used

for the reliable processing of large amounts of data. This OS should effectively support this process (in batch mode or time allocation). An example of an OS of this class would be IBM's OS /390.

The following category includes *server* operating systems. The main feature of such operating systems is the ability to serve a large number of user requests for shared resources. Network support plays an important role for them. There are specialized server OSes that exclude elements that are not related to the performance of their basic functions (for example, support for user applications). Universal servers (UNIX or Windows XP systems) are now more commonly used to implement servers.

The most massive category is personal OS. Some operating systems in this category, developed with the expectation of the user (Windows 95/98 / Me) by Microsoft are simplified versions of the universal OS. Particular attention in the personal OS is given to the support of the graphical user interface and multimedia technologies .

There is also a *real-time* OS. In such a system, each operation must be guaranteed to be performed within a specified time range. Real-time OS can control the flight of a spaceship, process or video demonstration. There are specialized real-time OSes such as QNX and VxWorks.

Another category is *embedded* OS. These include managing applications for various microprocessor systems used in military technology, consumer electronics systems, smart cards, and other devices. Such systems pose special requirements: placing a small amount of memory and support for specialized OS devices. Often, built-in OS is developed for a specific device; universal systems include embedded Linux and Windows CE.

Functional Components Of Operating Systems

An operating system can be considered as a set of components, each of which is responsible for the implementation of a specific function of the system. Consider the most important features of the modern OS and the components that implement them.

The way the system is built from components and their relationship is determined by the architecture of the operating system. Each operating system is going to be a bit different in the kind of work that it can handle, and its organizational structure, so learning this and how to put it all together can be important.

Process And Flow Management

As we mentioned, one of the most important functions of OS is to execute applications. Code and application data is stored in the computer cybernetic system on disk in a special executable manner. After the user decides to run either OS to perform a file system creates the basic unit of a computer, called a *process* . You can specify the following: a process is a program that executes it.

The operating system allocates resources between processes. These resources include CPU time, memory, devices, disk space as files. For the allocation of memory of each process, undertake its *address space* - set address memory, which allows you access. The process space is stored in the address space. The allocation of disk space for each process formed a list of open files similarly.
The processes protect the resources they possess. For example, the process address space cannot be accessed directly from other processes (it is secure), and when working with files, a mode can be specified that denies access to the file to all processes except the current one .

The allocation of processor time between processes is necessary because the processor executes instructions one by one (ie, at a particular time, only one process can physically execute on it), and for the user, the processes should appear as sequences of instructions executed in parallel. To achieve this effect, the OS provides the processor with each process for a short time, after which it switches the processor to another process; in this case, the execution of the processes resume from the place where they were interrupted. In a *multiprocessor system*, processes can run in parallel on different processors.

Modern operating systems in addition to processes can support multitasking, which provides in the process, the presence of several sequences of instructions (*threads),* which run in parallel to the user, like most processes in the OS. Unlike processes, threads do not provide resource protection (for example, they share the address space of their process).

Memory Management

While executing the code, the processor takes instructions and data from the computer's (main) memory. This memory is displayed as an array of bytes, each of which has an address.

As mentioned, the main memory is a type of resource between processes. OS is responsible for the allocation of memory. The address space is protected during the process and released only after the execution process is completed. The amount of memory available to the process can vary in the course of the distribution of memory.

OS must be capable of programs, individually or in the aggregate amount available for the main memory. To this end, virtual memory technology must be realized. This technology allows placing in the main memory only those instructions and processes that are needed at the current time, while the contents of the rest of the address space are stored on disk.

I / O Management

The operating system is responsible for managing I / O devices connected to the computer's memory. Support for such devices in the OS is usually performed at two levels. The first lower level includes *device drivers* - software modules that control devices of a particular type, taking into account all their features. The second level includes a *versatile I / O interface* convenient for use in applications.

The OS should implement a common I / O driver interface through which they interact with other system components. This interface makes it easy to add drivers for new devices. Modern OSes provide a large selection of

ready-made drivers for specific peripherals. The more devices the OS supports, the more chance it has of practical use .

File Management And File Systems

For OS users and programmers, disk space is provided as a set of *files* organized into a *file system.* A file is a set of files on a file system that can be accessed by name. The term "file system" can be used for two concepts: the principle of organizing data in the form of files and a specific set of data (usually the corresponding part of the disk) organized in accordance with this principle. As part of the OS, it can be implemented simultaneously supported and ICA several file systems.

File systems are considered at the logical and physical levels. The logical level defines the external representation of the system as a collection of files (usually located in directories), as well as performing operations on files and directories (creation, deletion, etc.). The physical layer defines the principles of allocation of data structures of the file system on the drive.

Network Support

Network systems

Modern operating systems are adapted to work on the network, they are called *network operating systems* . Networking support enables the OS to:

- To make local resources (disk space, printers, etc.) publicly available over the network, ie to function as a server
- Refer to other computer resources through a network that is functioning as a client

Implementing the functionality of server and client based on *vehicles* responsible for the transmission of data between computers according to the rules specified network protocols.

Distributed systems

Network OSes do not hide the presence of a network from the user. The network support in them does not determine the structure of the system and enriches it with additional capabilities. There are also *distributed* OSs that allow pooling the resources of several computers in a *distributed system.* It appears to the user as one computer with multiple processors working in parallel. Distributed and multiprocessor systems are two major categories of OS that use multiple processors.

Data security

Data security in the OS means ensuring the reliability of the system (data protection against loss in case of failure) and protection of data against unauthorized access (accidental or intentional). To protect against unwarranted access, the OS should ensure the availability of *authentication* of users (such means allow to determine whether the users are actually who they say they are. This is usually used for system passwords) and their *authorization* (to verify user rights which have been authenticated to perform a specific operation).

User Interface

There are two types of user interaction means running: *shell* (shell) and *a graphical user interface* (GUI). The command interpreter enables users to interact with the OS using a special command language (online or through startup) to execute batch files. Commands of this language force the OS to perform certain actions (for example, run applications, work with files).

The graphical user interface allows it to interact with the OS by opening windows and executing commands with menus or buttons. There are many approaches to implementing a GUI: for example, in Windows systems, its support systems are built into the system, and in UNIX, they are external to the system and rely on standard I / O controls.

Conclusions

- An operating system is a level of software that lies between the levels of applications and computer hardware. Its main purpose - to make use of computer systems easier and improve efficiency

- The main functional components of the OS include process management, memory management, I / O management, file management and file system support, network support, data protection, user interface implementation

Chapter 2:
Operating System Architecture

An operating system can be considered as a set of components, each of which is responsible for certain functions. The set of such components and the order of their interaction with each other and with the external environment is determined by the *architecture of the operating system.*

Consider the basic concepts of operating system architecture, approaches to them, features of OS interaction with the external environment. Let's consider the implementation of the architecture on the example of Linux.

Basic Concepts Of Operating System Architecture

System Kernel - Privileged Mode And User Mode

The basic components of the OS, which are responsible for its most important functions, are usually in the memory constantly and are executed in a privileged mode, called the *kernel of the operating system* .

Current approaches to the design of OS architecture differently define the functionality of the kernel. The most important functions of the OS, the teaching of which, of course, lay in the core include interrupt processing, memory control, input-output control. The kernel's reliability and performance make it more demanding.

The core feature of the kernel is that it runs in privileged mode. Consider the features of this model. To ensure efficient resource management computer’s memory operating system should have certain privileges regarding applications. It is necessary that applications do not operate the OS, and simultaneous OS should be able to intervene in a program, such as CPU or switching solution liabilities conflict in the struggle for resources.

Realizing these benefits requires hardware support for processors. They must support at least two modes - *privileged* (protected mode kernel mode, kernel mode) and *user mode* (user mode) in the invalid user commands that are critical to system operation (switching tasks to appeal to memory, given the limits access I / O devices, etc.).

Consider how different CPU modes are used when interacting with the kernel and applications. After loading the kernel switches, the processor in privileged mode and gets complete control of a computer. Each application is started and executed in user mode, where it does not have access to kernel resources and other applications. When you need to perform an action implemented in the kernel, the application makes *a system call* . The kernel intercepts it, switches the processor to privileged mode, performs an action, switches the processor back to user mode and returns the result of the application.

A system call is slower than a call to a user-implemented function because the processor switches between modes twice. For some OSes, some of the functionality is implemented in user mode, so you don't need to use system calls to access it.

Implementation Of Operating System Architecture

Consider several approaches to implementing an operating system architecture. In real OS, they usually use some combination of these approaches.

Monolithic Systems

The OSes, in which all the basic functions are concentrated in the kernel, are called *monolithic systems* . In the implementation of a monolithic kernel, the OS becomes more productive (the processor does not switch between modes during the interaction between its components), but less reliable (all its code is executed in privileged mode, and the error in each of the components is critical) .

A monolithic kernel does not mean that all of its components must reside in the memory. Modern operating systems allow placing code fragments *(kernel modules)* dynamically in the kernel address space. The implementation of kernel modules also makes it possible to achieve its

extensibility (to add new functionality, it is enough to develop and load the appropriate module in memory).

Multilevel Systems

Components of *multilevel* OS form a hierarchy of levels (layers), each of which relies on functions of the previous level. The lowest level interacts directly with the hardware, the highest level implements system calls.

In traditional multilevel operating systems, transferring control from the top to the bottom is implemented as a system call. The top-level must have the right to execute this call and these rights are verified with the support of the hardware. An example of such a system is the Multics OS, developed in the '60s. The practical application of this approach is limited today because of poor performance.

Levels can also be isolated in a monolithic kernel; in this case, they are programmatically supported and cause system implementation to be disrupted. In the monolithic system, a kernel determines the levels:

- *Abstraction means from equipment* that one interacts with the hardware directly, releasing other components of the system from such interaction
- *Core kernels* that are responsible for the most basic, simplest kernel actions, such as writing a block of data to disk. By these means, upper-level instructions are executed with the management of resources
- *Resource management tools* (or resource managers) that implement basic OS functions (process management, memory, input-output, etc.). At this level, the most important resource management decisions are made that are made using the core kernel
- *A system call interface* that is used to communicate with system and application software

Micronucleus Systems

One of the trends of modern operating systems is that in privileged mode, a small proportion of kernel functions, which are a *microkernel*, *are* implemented. Other OS functions are performed by user-mode processes (server processes). Servers are responsible for supporting the file system for dealing with process memory.

The micronucleus communicates between system components and performs a basic resource allocation. To execute a system call, the process (client process) accesses the microkernel. The microkernel sends a request to the server, the server performs the work and sends the response back. And the micronucleus forwards it to the client. Customers can not only be single users and other operating units.

The advantages of a micro-core approach are:

- The small size of microkernel which simplifies its development and debugging
- High system reliability due to the fact that the servers run in user mode and do not have direct access to the hardware
- Greater flexibility and expandability of the system (unnecessary components do not take up space in memory ' memory, expanding the functionality of the system is reduced to adding to it new server)
- The possibility of adapting to network conditions (way to exchange data between the server and the related network or they are on the same computer)

The main disadvantage of the micro-core approach is the decrease in performance. Instead of the two-time switchings of the processor mode in case of a system call, four occur (two - during the exchange between the client and the micronucleus, two – during the exchange between the server and the micronucleus).

This disadvantage is rather theoretical, in practice, the performance and reliability of the micronucleus depend primarily on the quality of its implementation. Thus, in QNX OS, the micro-kernel takes up several kilobytes of memory and provides a minimal set of functions, with the system corresponding to real-time OS performance.

The Concept Of Virtual Machines

In *virtual machine systems* programmatically create copies of the hardware (emulation occurs). These copies (*virtual machines*) run in parallel, each running software that interacts with applications and users.

Architecture Features: UNIX and Linux

UNIX Basic Architecture

UNIX is an example of fairly simple OS architecture. Much of the functionality of this system is contained in the kernel, the kernel communicates with applications through system calls.

The system has three main components: the process management subsystem, the file system, and the I / O subsystem. *The process management subsystem* controls the creation and deletion of processes, the allocation of system resources between them, inter-process interaction, memory management .

The file subsystem provides a single interface for accessing data stored on disk drives and peripherals. Such an interface is one of the most important features of UNIX. They use the same system calls to either communicate with the disk or output to a terminal or printer (the program works with the printer just like a file). In this case, the file system forwards requests to the relevant modules of the I / O subsystem, and those - directly to the peripheral devices. In addition, the file subsystem controls file access rights, which largely determine the privileges of the user on the system.

The I / O subsystem performs file system requests by interacting with device drivers. In UNIX, two types of devices: character (eg, printer) and block (eg, hard disk). The main difference between them is that the block device allows direct access. To increase the productivity of block devices using the buffer cache - plot memory which stores data last scanned from the disc. Subsequent accesses to these data can be retrieved from the cache.

Linux Architecture

There are three main parts to Linux :

- *Core* , which implements the basic functions of the OS (process management, input-output, memory, etc.)
- *System libraries* that define a standard set of functions for use in applications (performing such functions does not require a transition to privileged mode)
- *System utilities* (applications that perform specialized tasks)

The Purpose Of The Linux Kernel And Its Features

Linux implements a monolithic kernel technology. All kernel data and data structures are in the same address space. There are several functional components in the kernel.

- *Process Scheduler* - Responsible for the implementation of multitasking in the system (interrupt processing, timer operation, creation and completion of processes, context switching)
- *Memory Manager* - allocates a separate address space for each process and implements support for virtual memory
- *Virtual File System* - Provides a universal interface for interacting with various file systems and me/ O devices
- *Device Drivers* - Provide direct access to peripherals. They are accessed through the virtual file system interface
- *Network Interface* - Provides access to the implementation of network protocols and network device drivers
- *Interoperability Subsystem* - Offers mechanisms that allow different processes in the system to communicate with each other

Some of these subsystems are logical components of the system, they are loaded into memory with the core and remain there

permanently. Components of other subsystems (such as device drivers) to implement profitable so that their code could upload them on demand. To accomplish this, Linux supports the concept of kernel modules.

Kernel Modules

The Linux kernel makes it possible to load and unload individual sections of code on demand. These sections are called *kernel modules* and run in privileged mode. Kernel modules offer a number of advantages.

- Code modules can be loaded in memory ' Five in the process of the system, which simplifies debugging kernel components, especially drivers
- They have the ability to change a set of core components at runtime: the ones that are currently not in use, cannot be loaded into memory
- Modules are an exception to the rule that code that extends kernel functionality under a Linux license must be open. This allows hardware manufacturers to develop drivers for Linux, even if they do not have access to their source code

Linux module support has three components.

- Module management tools make it possible to load modules into memory and communicate between modules and the rest of the kernel
- Driver logging tools allow modules to report to another part of the kernel that a new driver has come to fruition
- Conflict resolution tools allow device drivers to reserve hardware resources and protect them from being accidentally used by other drivers

A single module can register multiple drivers if required (for example, for two different access mechanisms). The modules can be downloaded in advance - at system start-up (boot modules) or in the run of a program that triggers their functions. After loading, the module code is in the same

address space as the other kernel code. A module error is critical to the system .

Features Of System Libraries

Linux system *libraries* are *dynamic libraries* that are only loaded into memory when needed. They perform a number of functions:

- Implementation of system call packers
- Extension of system call functionality (such libraries include the C language I / O library, which implements functions such as printf () based on system calls
- implementation of user-mode service functions (sorting, row processing, etc.)

User Application

Linux user applications use functions from system libraries and interact with the kernel through system calls.

Installation

Unlike what happens with Microsoft Windows, installing Linux is not a simple process since Linux allows the control and customization of a greater number of parameters and options. In spite of everything, they are making great progress looking for the installation of Linux to be the least traumatic process possible, depending on the simplicity of it of the distribution used .

Despite everything, before proceeding to install Linux it is necessary to take into account a number of fundamental aspects. The first one is to read the information contained in the installation CD, this information can appear in two different ways, the regular HOW TO or in the form of manuals developed for distribution.

The general steps to follow for the installation of this operating system, in most of its distributions, are summarized in the following steps:

Installation process:

- Language Choice: The installation program gives us the possibility to configure our system in a large number of languages that exist
- Acceptance of the license agreement with the company that owns the distribution we own
- Configuration of the type and model of our keyboard and mouse
- Choice of security level: In which there are four default security levels to choose from, depending on the use of the computer on which we are performing this installation, being able to choose levels from an internet client to a server with a large number of connections to support
- Partitioning: One of the main concepts to consider before installation is partitio n

Each operating system organizes the information of the files it contains differently, using your own file system. For reference, the file system name of different operating systems:

This generally prevents multiple mixed operating systems from being installed on the same hard drive. To solve this problem there are the so-called partitions with which a certain hard disk is divided so that it can contain both file systems. For all purposes, making a partition is equivalent to the hard drive is divided into two, although of course it is not divided into a physical but logical way. The fundamental problems when installing Linux come from the fact that in most cases the user wants to keep Windows and all programs for this system. There are currently several distributions that allow the installation of Linux on a Windows file system, either in what is called an image disk (A very large file), of the Corel Linux distributions, or directly in the Windows file system (WinLinux 2000). However to obtain a good performance it is preferable to install Linux on a partition different from that of Windows using Linux's own file system, so it is usually necessary to perform a partition of the hard disk, since Windows when installed, usually appropriates the entire size of the hard drive. Until

recent times this division meant the irremediable and inevitable loss of all the information that was contained on the hard disk.

At the moment in the installation of the distributions of Linux, it is allowed to divide the hard disk without losing information, in order to reduce the size of the Windows partition and create a new one during the installation of Linux. In the case of RedHat or Mandrake, there is a possibility of automatic assignment that will be useful in the case of or knowing how to manipulate the partitions. For Linux, at least one root native partition must be created. It should be the desired size and a swapping partition that must be at least twice the size of the RAM of your equipment, up to a maximum of 512 MB.

During this process, care must be taken not to erase the partition where Windows resides, since in that case ALL the information will be lost permanently.

- System installation: Selection of the various packages to be included with the Linux operating system and its subsequent installation.

Some of these packages are:

- Workstation
- Server
- Graphic environment
- Office work station.
- Webserver / FTP
- KDE station
- Play Station
- Mail.
- Gnome station
- Multimedia station
- Database server
- Internet station
- Firewall Server / Router
- DNS / NIS
- Scientific work station
- Console tools
- Network server

System Configuration

- Root password: The installation program asks you to enter a root or administrator password with which you will be able to have access to all the configuration areas of the system, which as a normal user is not possible
- Creation of a user: The installation program asks for a login and password, for the creation of the first user, which does not have the necessary administrator permissions for security reasons
- Boot loader: Another point to consider is how you want to boot Linux. If Windows exists, the easiest way is to install LILO or GRUB, programs that are responsible for booting both operational systems according to what the user indicates when starting the PC. It must be selected when setting the option of saving the magazine in the first sector of the disk. Problems will arise to boot Linux is if there is an option to reinstall Windows again, since this operating system assumes control of the PC and boot system, removing LILO (and preventing Linux boot). Another very simple way would be to use the boot floppy disk that is created during the installation of Linux
- Installation exit: The user is informed of the installed services and the installation program is finished

Once Linux is first started, the initial wizard will start and allow you to configure basic system items, such as the type of desktop to use; KDE, Gnome, WindowMaker, etc. and some services like mail.

Conclusions

- The OS architecture defines a set of components, as well as how they interact with each other and with the environment
- The most important for studying the architecture of the OS is the concept of the core of the system. The core feature of the kernel is that it runs in privileged mode
- The main types of OS architecture are monolithic and micro-kernel based architecture. Monolithic architecture requires that the core

functions of the system be concentrated in the kernel, its most important advantage being performance. In microkernel systems in privileged mode, only the basic functions are performed, the main advantages of such systems are reliability and flexibility

- The operating system directly interacts with the computer hardware. Modern computer architectures offer many operating system support tools
- The operating system interacts with applications. It provides a set of system calls to access functions implemented in the kernel. For applications, system calls, along with system library tools, are accessible through the application programming interface

Chapter 3: Process and Flow Management

Consider two major abstractions of the operating system that describe the execution of the program code - processes and flows - features of their implementation in the operating system, the states in which they may reside, basic mechanisms for working with them (means of creation, completion, and termination, etc.).

Basic concepts of processes and flows

Processes and flows in modern OS

In today's operating system, the kernel code (which belongs to its various subsystems) and the code of the user programs are executed at the same time. There are different actions: some programs and subsystems follow the instructions of the processor. Others are busy with I / O, some still wait for requests from the user or other applications. To simplify the management of these actions in the system, it is advisable to allocate a set of elementary active elements and to define the interface of OS interaction with these elements.

The *process* means the abstraction of the OS, which combines everything necessary to execute one program at a certain point in time. The program is some sequence of machine commands stored on disk, and if necessary, is loaded into memory and executed. It can be said that at run time, the program represents a process.

A one-to-one correspondence between a program and a process is established only at a specific point in time: one process at a time can execute the code of several programs, the code of one program can execute several processes at the same time.

Successful programs require some resources. They include:

- Resources required for sequential execution of program code (primarily CPU time)

- Resources that allow storing information that ensures the execution of the program code (CPU registers, RAM, etc.); these resource groups identify two components of the process
- The sequence of executable commands of the processor
- Set addresses memory ' memory (*address space*), which are these commands and data for them

The selection of these parts is also justified by the fact that within the same address space may be several sequentially executed sequences of commands that share the same data. The need to distinguish between the sequence of commands and the address space leads to the concept of flow .

A thread (control thread) is called a set of sequentially executed processor commands that use the shared process space. Since there can be many threads at a time, the OS's task is to organize the switching of the processor between them and to plan their execution. In multiprocessor systems, the code for individual threads can be executed on separate processors.

Thus, the *process is* called the set of one or more threads and the protected address space in which these threads are executed. The security of the process address space is its most important characteristic. The process code and data cannot be directly read or overwritten by another process; thus protecting themselves from the rich bugs and unauthorized access attempts. Naturally, only *direct* access is inadmissible (for example, writing to memory using a simple data transfer instruction); the exchange of data between processes is, in principle, possible, but for this purpose special means, which are called means of *interprocess interaction,* should be used. Such tools are more difficult than direct access and run slower, but provide protection against accidental errors when accessing data.

Unlike the process *flows managing total memory, stream* data is not secured against access to other streams provided that they are all executed in the address space of the same process. This provides additional application development capabilities but complicates programming .

The protected alert process space sets the abstraction of code execution on a single machine, and the thread provides an abstraction of sequential command execution on a single dedicated processor. The address space of the process does not always correspond to RAM addresses. For example, it may display files or registers of the I / O controllers, so writing to a specific address in this space will either write to a file or perform an I / O operation. This technology is called the *reflection in memory* (memory mapping).

Process and Flow Models

The maximum number of processes (secure address spaces) and the threads that they execute may vary across systems.

- In single-tasking systems, there is only one address space in which one stream can be executed at any one time
- Some embedded systems also have one address space (one process), but it allows many threads to execute. In this case, parallel computations can be organized, but data protection is not implemented
- In systems similar to traditional UNIX versions, many processes are allowed, but only one thread is executed within the process address space. This is a traditional single-threaded *process model.* The concepts of flow in this model do not apply but use the term "switching between processes", "process planning", " sequence of process commands" and the like (here by the term process, we mean a single flow)
- Most modern operating systems (such as the Windows XP line, modern versions of UNIX) can have many processes in the address space of each process. These systems support multithreading or implement *a flow model.* A process in such a system is called a *multithreaded process*

In the future, the term “flow” is used to refer to the sequence of executing commands, except in situations where the implementation of the process model will be discussed.

Process and Flow Components

The elements of the process include:

- Protected address space
- Data common to the whole process (these data can be shared usage of all its streams)
- Resource usage information (open files, network connections)
- Process flow information

The stream contains the following elements:

- The state of the processor (a set of current data from its registers), including the counter of the current instruction of the processor
- Flow stack (the memory area where the local flow variables and function return addresses are called in its code)

Process And Flow States

The following states are allowed for the stream:

- *Creation* (new) - the thread is in the process of creation
- Execution (running) - instruction stream processor performs (at a particular time on a single processor, only one thread can be in this state)
- *Waiting* (waiting) - awaiting some event flow (such as completion operations); this state is also called with a blocked and terminated flow
- *Readiness* (ready) - Stream expects scheduler switches CPU at him when it has all the necessary resources, in addition to CPU time
- *Terminated (* terminated) - the thread has completed execution (unless its resources have been removed from the system, it goes into an additional state - *a zombie state*)

The transition of flows between standby and readiness states is based on *task planning* or *flow planning.* When scheduling threads, determine which threads need to be restored after completing the I / O operation, how to organize the system expectations.

The transition between standby and execution states requires a *processor time schedule* . Based on the algorithms of such planning, determine which of the finished threads need to be executed at a specific time, when do you need to interrupt the execution of the thread to switch to another ready thread and so on.

With regard to systems that implement the process model, it is customary to talk about the state of processes, not flows, and the planning of processes; in fact, the states of the process in this case clearly correspond to the states of its single flow.

Description Of Processes And Flows

As we already know, one of the main tasks of the operating system is the allocation of resources between processes and flows. These resources are primarily CPU time (distributed between streams during scheduling), I / O, and memory (shared between processes) .

To manage the allocation of resources, the OS must support data structures that contain information describing processes, flows, and resources. The following data structures include:

- Resource allocation tables: memory tables, I / O tables, file tables, etc.
- Process tables and flow tables that provide information about the processes and flows present in systems at a specific point in time

Managing Process And Flow Blocks

Information about the processes and flows in the system is stored in special data structures, which are called process control blocks and flow control blocks. These structures are very important for the OS because on the basis of their information system manages processes.

Managing power flow (Thread Control Block or TCB) to an active flow, ie that which is in a state of readiness, expectation or execution. This block may contain the following information:

- Flow identity (usually its unique identifier)
- Flow processor status: custom processor registers, instruction counter, the stack pointer
- Flow planning informatio n

Table streams - is a list or array of flow control units. It is located in a protected area of the OS memory. *The managing unit process* (Process Control Block or PCB) corresponds to the process that is present in the system. Such block may contain:

- Process identity (unique identifier information about other processes related to adan)
- Information about the threads that are running in the address space of the process (for example, pointers to their control blocks)
- Information that allows you to determine the rights of a process to use different resources (for example, the user ID that created the process)
- Process address allocation information
- Information about I / O resources and files used by the process

Process And Flow Images

The totality of the information displayed in the memory process called the *image process* , and all information flow (thread image). The image process includes:

- Process control unit

- User code
- User data (global program data common to all threads)
- Process flow pattern information

The user program code, user data, and stream information are loaded into the process address space. The image process is generally not a continuous section of memory, its parts are unloaded to disk. The streaming image includes:

- Flow control unit
- Kernel stack (the flow stack used when executing kernel flow code)
- User stack (stream stack available in user mode)

Creating and Completing Processes and Threads

The process creation and termination tools allow you to dynamically change the set of applications that are running on the operating system. Thread creation and completion tools are the basis for creating multithreaded applications.

Creating Processes

Basic principles of process creation

Processes can be created by the kernel of the system during its initialization. For example, in UNIX- compatible systems, such a process can be the process of initialization of the in-it system, in Windows XP - the processes of subsystems (Win 32 or POSIX). This creation process, however, is the exception, not the PR and the fork.

Most often, processes are created when performing other processes. In this case, a process that creates another process is called *an ancestor* and the process created by it is called a *descendant.* New processes can be created while the application is running according to its logic (the compiler can create processes for each compilation step, the web server can handle the

incoming requests), or directly at the request of the user (for example, from a command interpreter, graphical shell, or file manager).

Interactive And Background Processes

There are two types of background processes in terms of their interaction with the user.

- *Interactive processes* interact with users directly, receiving input from the keyboard, mouse, and more. An example of an interactive process is the process of a text editor or an integrated development environment
- *Background processes* do not interact with the user directly. They are usually started at system startup and are waiting for requests from other applications. Some of them (system processes) support the functioning of the system (implement background printing, networking tools, etc.), others exclude specialized tasks (implement web servers, database servers, etc.). Background processes are also called services (services, in Windows XP systems) or demo about us (daemons, on UNIX)

Managing Address Space When Creating Processes

Because the core element of the process is secure address space, it is important to solve the problem of allocating it when creating a new process. Consider two different approaches.

System Calls Fork () Exec ()

In the first approach, the descendant's address space is created as an exact copy of the ancestor's address space. This operation is implemented by a system call, which is POSIX systems, which is called fork (). In this case, not only the address space is copied, but also the counter of the mainstream process commands, so after calling fork () the ancestors and descendants

will follow the same instruction. The developer must determine which of the two processes should be managed. This can be done based on the differences between the fork () return codes for the ancestor and the descendant .

When the creation of a new process occurs by duplicating the ancestral address space, there is a need for special means of loading the code into the address space of the process. Such tools are implemented by a system call, which is POSIX systems, which is called exec (). As an option to call exec() must specify the full path to the executable program to be loaded into memory. In fork () -based systems, in order to run the program, you must immediately call exec () after calling fork () (this is called fork + exec technology).

Run The Application With One System Call

The second approach does not separate address space duplication and code loading - these steps are combined here into one. In this case, the system call triggers the specified application to execute (usually it needs to specify the entire path to the executable file of this application). It is to be divided into two stages of implementation of the system call:

- Allocation of memory ' memory address space under the new process (with no information from the address space ancestor is not copied)
- Download executable code from the specified file into the highlighted address spac e

The approach using fork() and exec() is more flexible, as it allows you to limit yourself to one single step of starting an application if necessary. Modern operating systems mainly implement a combination of first and second approaches.

Features Of Process Completion

Consider three options for completing processes. The process *correctly completes* on its own after completing its task (for interactive processes, this is often done at the initiative of a user who, for example, used the appropriate menu item). To do this, the process code must perform a system call to complete the process. This call on POSIX systems is called _ exit (). It can turn the process that caused it.

The process *crashes* due to an error. This output can be provided by the programmer (error when processing a decision that the program cannot continue) and may be due to the generation of interruption (division by zero, access to a secure area memory ' memory, etc.).

The process is *completed by another process* or system kernel. For example, before the OS shutdown, the kernel stops performing all processes. The process can terminate a process through systems is the many calls which in POSIX -a system called kill () .

Once the process completes its work memory reserved for its address space is freed and can be used for other purposes. All flows of this process also cease to exist. If there are descendants in this process, their work is preferably not terminated after the work of the ancestor. The online processes are usually terminated when the user logs off.

Synchronous And Asynchronous Execution Of Processes

When the system is a new process for the old process, there are two basic options:

- Continue execution in parallel with the new process - this mode of operation is called *asynchronous execution*
- Suspend execution until the new process is complete - this is called *synchronous execution.* (In this case, they use a special system call, which is POSIX systems, is called wait ()

The choice of a particular model depends on the specific task. For example, a web server can create descendant processes to handle requests (if the descendant set is not enough to handle that request). In this case, the processing must be asynchronous, as soon as the descendant is created and the ancestor must be ready to receive the next request. Compiler C, on the other hand, must wait for the completion of each step before proceeding to the next stage, in which case they use synchronous processing.

Creating and Completing Streams

Features Of Creating Streams

Setting up streams is different, related primarily to the fact that the currents are created within an existing address space (or a particular kernel process). There are several situations that can be a new stream.

If a process is created by a fork () system call, flow is automatically created after the distribution of the address space within that process (often the main application flow). You can create streams from the user code with an appropriate system call.

Many operating systems have special threads that create the kernel of the system (the kernel code can also be executed in threads). When creating a stream, the system must follow these steps.

1. Create data structures that reflect the flow in the OS
2. Select city at the stack of flow
3. Set the processor command counter to the beginning of the code to be executed in the thread;

this code is called *a stream procedure* or *function, and the* ng pointer is referred to as a stream call or system call .

Note that when creating flows, as opposed to the creation process, there is no need to allocate memory for the new address space, so usually flows are

created faster and with lesser resources.

Features Completion Streams

At the end of the stream, its resources are released (first of all, the folder); this operation is usually faster than completing the process. The flow may be completed when the control reaches the end of the flow procedure; there are also special system calls intended for early termination of flows.

Like processes, threads can be executed synchronously and asynchronously. A thread that created another thread may pause its execution until it is completed. This expectation is called *joining streams (* thread joining). Upon completion of the attached thread, the thread that was waiting to complete it may receive execution status. When creating a stream or joining it, if the stream cannot connect, it is called *disconnected or* detached. If the stream is not disconnected, it is called *Pluggable* , after which it is necessary to join them to stop memory leakage .

Managing Processes In UNIX and Linux

Image Of The Process

In UNIX systems, the process image contains the following components:

- Process control unit
- The code of the program executed by the process
- The process stack where temporary data is stored (procedure parameters, return values, local variables, etc.)
- Global data common to the whole process

Identification Information And Process Security Attributes

The process identifier (PID) is unique throughout the system and is used to access this process. The init process identifier is always one. The process ID of the ancestor (ppid) is specified when it is created. If the ancestor of process P completes execution, the ancestor of that process automatically becomes init, so ppid for P will be unity.

With the process as fl ' related set of security attributes.

- The real user and process group identifiers (uid , gid) correspond to the user who started the program, resulting in the corresponding process appearing on the system
- Effective user and process group identifiers (euid , egid) are used in a special process execution mode - are owned by the owner

Process Control Unit

Consider the structure of the process control unit in Linux.

The process control block on Linux is displayed by the task _ struct data structure. The most important fields in this structure are fields that contain the following information:

- Identification data (including pid - process identifier)
- Process status (execution, expectations, etc.)
- Pointers to ancestral and descendant structures
- Process creation time and total execution time (so-called process timers)
- Processor state (contents of registers and counter of instructions)
- Process security attributes (id , gid, euid , egid)

Note that the Linux kernel does not have a separate data structure for the stream, so the processor status information is contained in the control block of the process. In addition to the above, task _ struct has several custom fields required for different Linux subsystems :

- Information for signal processing
- Process planning information
- About files and directories floor ' related to the process
- Data structure for managing memory subsystems

The task _ struct field data can be shared by several special purpose processes, in which case these processes are actually threads. The control units of the process are stored in the kernel in a special data structure. Prior to the release of kernel 2.4, this structure was called a system process table; it was an array whose maximum length could not exceed 4K. In the core processes, it has been replaced by two dynamic data structures that do not have such a restriction:

- A hash table (where the pid of the process acts as a hash), this structure allows you to quickly find the process by its ID
- Ring list, this structure ensures the implementation of actions in the loop for all system processe s

Now the limit on the maximum number of processes is checked only within the implementation of the fork () function and depends on the amount of available memory (for example, there is information that on a system with 512 MB of memory it is possible to create about 32000 processes).

The implementation of a control unit in Linux is different from its traditional implementation in UNIX systems. In most versions of UNIX (System V, BSD), the process control block consists of two data layers - a process structure (proc) and a user structure (u).

Creation Process

Implementation Of The Fork () in Linux

In UNIX- compatible systems, processes create an already known system call fork (). Let's consider its implementation in Linux.

1. Allocate memory for the new process control block (task _ struct). If memory is not enough, it returns an error
2. All values from the ancestral data structure are copied to the descendant data structure. After that, the fields whose values should be different from the original ones will be changed. If the user exceeds the specified number of processes for him or if the number of processes in the system exceeds the maximum possible value (which depends on the amount of available memory), the process creation stops and an error returns
3. An identifier (pid) is generated for the process using a special algorithm to guarantee uniqueness
4. For the descendant, the necessary additional ancestral data structures (table of file descriptors, information about the current directory, table of signal handlers, etc.) are copied
5. Form the address space of the process
6. The process data structure is placed in the list and hash table of system processes
7. The process is put into readiness for execution

Using Fork () In Applications

Consider an example of using a fork () system call to create a process. The description of fork () according to POS IX looks like this:

Include < unistd . h >

pid _ t fork (); // type pid _ t is an integer

Since executing fork () causes the control of both the ancestor and the descendant to switch to an operator, which after calling fork () (both begin

to execute one instruction), practically the only difference between the ancestor and the descendant in the programmer's view is in the code return fork () .

For descendant fork () returns zero and for ancestor the id (pid) of the created descendant process. When for some reason the descendant was not created, fork () will return –1. Therefore, the normal code for and fork () looks like this:

```
pid _ t pid ;
if (( pid = fork ()) == - 1)
     { / * error, crash * /) }
    if (pid == 0)
     {
       // is a descendant
      }

else
    {
      // this is the ancestor
     printf (" Descendant started with code % d \ n ", pid );
     }
```

Once the process is created, it can access the identifying information by a system call to getpid () which returns the current process ID, and getppid (), which returns the process ID of the ancestor .

```
#include <unistd. h>
pid_t mypid, parent_pid;
mypid = getpid ();
parent_pid = getppid ();
```

Completing The Process

To end the process, UNIX systems use the _ exit () system call. Consider implementing this system call in Linux. The following actions occur during its execution.

1. The status of the process shines through TASK _ ZOMBIE
2. The process informs the ancestors and descendants that it has ended (with the help of special signals)
3. The resources allocated during the fork () call are released
4. The scheduler is informed that the context can be switched

You can use either the _ exit () system call or its exit () library function to complete the process in applications. This function closes all process threads, correctly releases all resources, and calls _ exit ( ) to actually complete it:

```
#include <unistd.h>
void _exit (int status); // status specifies the return cod e
#include <stdlib.h>
void exit (int status); // status specifies the return code
exit (128);
```

Note that it is better not to call exit () when a process can use resources in conjunction with other processes (for example, it is a descendant process with an ancestor, and a descendant inherited resource descriptors from the ancestor). The reason is that in this case, trying to release the resources in the offspring will result in them being released from the ancestors as well. You must use _ exit () to complete these processes.

Waiting For Process To Complete

When the process is completed, its control block is not immediately removed from the process list and hash and remains there until another process (ancestor) removes it from there. If the process is actually missing in the system (it is finished), but only its control unit, then the process is called a *zombie process* .

Waitpid () system call

You can use the wait () system call to remove process information from Linux, but more often use the more versatile version of it - waitpid (). This call checks if the control unit of the corresponding process is in the system. If it is, and the prz is not in the zombie state (that is, it is still running), then the process in the case of waitpid () will go to standby. When the descendant process is completed, the ancestor exits standby and removes the descendant control unit. If the ancestor does not call waitpid () for the descendant, it may remain in the zombie state for a long time.

Synchronous Implementation Of Processes In Applications

Let's consider the synchronous execution of processes based on waitpid (). According to POSIX, this system call is defined as:

```
# Include < sys /wait. h >
pid _ t waitpid ( pid _ t pid , // pid of the expected process
        int * status,// descendant completion status information
        int options ); // suppress as 0
```

The pid parameter can be set to 0, which means that the process is waiting for the same group to which the ancestor belongs, or as –1, which means the expectation of any descendant. Here is an example of the implementation of synchronous execution with the expectation:

```
pid _ t pid ;
if ((pid = fork ()) == - 1 )
   exit (-1);
```

```
if (pid == 0)
{
// descendant - call exec ()
}
else
{
// ancestor - wait for posterity
int status;
waitpid (pid, & status, 0);
// continue execution
}
```

The status value gives you more information about the completed descendant process. There are a number of macros with < sys/wait for .h>:

- WIFEXITED (status) - a non-zero value when the descendant of a step has completed normally;
- WEXITSTATUS (status) - descendant return code (only when WIFEXITED ()! = 0).

The descendant return code is obtained as follows :

```
waitpid (pid, & status, 0);
if (WIFEXITED (status))
        printf (" Descendant ended with code % d \ n",
    WEXITST A TUS (status));
```

5/3 7 Alerts

In the case of multitasking, there is a need to report processes about events occurring in the system or other processes. The simplest mechanism for this alert, defined by POSIX , is the *alerts* . The process after receiving the signal immediately responds to his call special functions - a *handler* of the signal (signal handler), or action by default for this signal with each signal

related to its number, which is unique in the system. No other information along with the signal can be transmitted.

Alarms are the simplest mechanism for interprocess on UNIX systems but since they allow the transmission of limited data, they are mainly used for event reporting.

Types Of Signals

Depending on the circumstance, the signals are divided into synchronous and asynchronous. *Synchronous signals* occur during the execution of an active process stream (usually through an error - access to the wrong memory area, incorrect floating-point operation, execution of incorrect and n processor design). These signals are generated by the OS kernel immediately
sends them to the process whose thread was causing the error.

Asynchronous signals can be received by the process at any time of execution:

- A programmer can send an asynchronous signal to another process using a system call, which is POSIX systems, is called kill (), the parameters of that call are the signal number and process identifier
- The signal may also be caused by some external event (keystroke, termination, etc.)

During the processing of such signals, the system must interrupt the execution of the current code, execute the processor code and return to the same instruction that was executed at the time of receiving the signal.

Layout Of Signals

On receipt of the signal process can respond in one of the three ways (method response process called signal alert disposition):

- Call the signal handler
- Ignore a signal that in this case "disappears" and does not take any action
- Use the default allocation (this is the default setting for each signal, often ending the process)

The process can set individual exposure for each signal.

Signal Blocking

The process can not only set the position of signals but also determine its willingness to receive signals of a certain type at that moment. If the process is not ready, it can block these signals. If a locked signal is to be delivered to a process, the system queues it where it will stay and remains until the process unlocks it. The process locks and unlocks signals by changing the *process signal mask* (a special data structure that stores information about what signals can be delivered to the process immediately, usually stored in its control block. The descendant processes inherit the ancestral signal mask.

Examples Of Signals

Consider the signals by POSIX and supported in Linux (in parentheses next to the name ' pit signal is given his room). By synchronous signals include, for example, the signal SIGSEGV (11) that generates the system while recording a protected area of memory .

Asynchronous signals include:

- SIGHUP (1) - a gap called ' communication (eg, user out of the system)
- SIGINT and SIGQUIT (2,3) - program interrupt signals from the keyboard (generated by a user pressing Ctrl + C and Ctrl + \ respectively)

- SIGKILL (9) - immediate termination of the program (this signal cannot be changed position)
- SIGUSR 1 SIGUSR 2 (10,12) - user signals that can use applications
- SIGTERM (15) - Offer the program to shut down (this signal, unlike SIGKILL, may be ignored)

The default action for all of these signals, except for SIGSEGV, is to terminate the program (an additional *memory dump* (core dump) is generated for SIGSEGV - a file that stores the image of the process address space for later analysis).

Setting The Signal Layout

A sigaction () system call is used to set the signal disposition.

```
#include <signal.h>
int sigaction (int signum, // signal numbe r
        struct sigaction * action, // new layout
        struct sigaction * old_action); // return the previous disposition
```

The exposition is described using the sigaction structure with the following fields:

- sa _ handler - pointer to the signal processor function
- sa _ mask - the signal mask that specifies which signals will be blocked inside the processor
- sa_flag - additional checkboxes

Limit to zeroing sa _ mask and sa _ flag (without blocking any signal):

```
struct sigaction action = {0};
```

The sa _ handler field should be set as a pointer to a previously declared function that looks like this:

```
void user_handler (int signum)
{
  // signal processing
}
```

This feature becomes a signal processor. The signum parameter determines which signal is sent to the handler (the same handler can be logged for multiple signals by several sigaction () calls). After logging in, the handler will always be called when it receives the appropriate signal:

```
#include <signal.h>
void sigint_handler (int signum)
{
  // SIGINT processing
}
// …… ..
action.sa_handler = sigint_handler;
sigaction (SIGINT, & action, 0);
```

If we need to organize the wait for the signal, the simplest way is to open a system call pause (). In this case, the process enters standby mode, from which it will output any signal:

```
// ask the handlers for help with sigaction ()
pause (); // wait for a signa l
```

Signal Generation

Let's consider how to send a process signal. This uses the kill () system call.

```
#include (signal.h>
```

```
int kill (pid_t pid, // process ID
      int signum ); // signal number
```

For example, sending a SIGHUP signal to a process pid that is specified at the command line:

```
kill ( atoi ( argv [1], SIGHUP );
```

Organization Of Asynchronous Execution Of Processes

Consider how you can use signal processing to organize asynchronous process execution. As you know, calling waitpid () causes an offspring to execute synchronously. When to start the descendant process is asynchronous. It seems natural not to call waitpid () in the ancestor process for that descendant. But upon completion of the descendant process, it will turn into a zombie process.

To avoid this, you must take advantage of the fact that the ancestor process receives a special SIGCHLD signal at the end of the descendant process. Calling waitpid () in this signal handler will remove the descendant process information from the process table, leaving zombies in the system.

```
void clear_zombie (int signum)
{
   waitpid (-1, NULL, 0);
}
struct sigaction action = {0};
action.sa_handler = clear_zombie;
sigaction (SIGCHLD, & action, 0);
if ((pid = fork ()) == - 1)
   _ exit ();
if ( pid == 0)
{
   // descendant running asynchronously
```

```
    exit ();
}
    else
{
    // ancestor does not have a waitpid () call
}
```

Managing Threads in Linux

Basic support for multithreading

Until recently, the only multithreading support for Linux was the clone () system call, which made it possible to create a new process based on an existing one. This call is similar to the fork () call but has several other features.

- For the new process, you need to specify a special set of checkboxes that determine how resources will be distributed between the ancestor and the descendant. The resources that can be shared include address space, open file information, signal handlers. Note that in the traditional clone () implementation, there is no special use of process identifier (pid), ancestor ID, and some other important attributes
- The new process poribno set a new stack (as a consequence call clone () two processes can share a common memory diamond, are unacceptable for general use stack)

Support clone () means implementing multithreading scheme 1: 1. The primary ones in the system are processes, not threads (in Linux, the instructions given by the kernel on which the kernel runs are called processes, not kernel threads). The traditional multithreading implementation on Linux determines that user threads are mapped to processes in the kernel. Therefore, it is impractical to look at the flow data structures in Linux separately - it is displayed with the same task _ struct structure as the process.

The Disadvantages Of Traditional Multitasking Support For Linux

Multithreading basics in the Linux kernel have not undergone the principle of changes since the clone () system call. In the meantime, Linux has evolved from an experimental system to an industry-wide system where enterprise-class heavy-duty applications are implemented.

This use of the system has led to the implementation of multi-threaded high reliability and scalability. It became apparent that the clone () system call implementation did not meet these requirements. We had the full implementation of multithreaded processing facilities at STI in the nucleus.

Flow control is part of the standard POSIX 1996 corresponding API was named *streams POSIX*.

Linux Kernel Threads

In addition to user processes and threads, Linux also supports a special kind of planned objects that are already known as kernel threads. Such flows are scheduled as normal processes and flows, each with its own id (pid). Differences of kernel threads from processes and user threads are that:

- The flow functions for them are defined in the kernel code
- They run only in kernel mode
- Them inaccessible areas of memory ' memory allocated in user mode

The Flow Management Software Interface

Creating Threads

To add a new thread to the current process, POSIX uses the pthread _ create () function with the following syntax:

```
#include <pthread.h>
int pthread_create (pthread_t * th, pthread_attr * attr,
void * (* thread_fun) (void *). void * arg);
```

Consider the parameters of this function:

- th is a pointer to a predefined structure of type pthread _ t which will then be passed to other functions of work with flows; then call the AI to *handle the flow* (thread handle)
- Attr is a pointer to a structure with flow attributes (you need to pass a null pointer to use the default attributes; some attributes will be reviewed later)
- thread _ fun is a pointer to a stream function that should be described as

  ```
  void * mythread_fun (void * value)
  {
  // executing the stream code
  }
  ```
- arg - the data passed to the stream function (there they will get a parameter value)

An example of creating a POSIX stream:

```
# include < pthread . h >
// flow function
void * thread_fun (void * num)
{
   printf (" stream number % d \ n", (int) num);
}
// ……
pthread_ t th;
// create the second thread
```

```
pthread_create (& th, NULL, thread_fun, (void *) ++ thread_num);
// there are two threads running in parallel
```

The new thread begins to run in parallel with the thread that created it. For example, if a thread was created inside the main () function, then two threads would continue to execute: an original program executing the main () code and a new one.

Completion of POSIX Streams

To end a thread, just go to the end of its thread function thread thread _ fun () or call pthread_exit () from it:

```
#include <pthread.h>
void pthread_exit (void * retval);
```

The retval parameter specifies the return code. Here is an example of ending a stream:

```
void * turead _ fun ( void * num )
{
printf (" stream number % d \ n", (int) num);
pthread _ exit (0);
}
```

There are two different situations for the application startup code:

- Executing the return statement or executing all statements by the end of main () *completes the whole process* with all threads also completing their execution
- Executing the pthread _ exit () function inside the main () function completes only the initial thread, this action does not affect other threads in the program - they will continue executing until they are completed

Calling the pthread _ exit () function is used only to join threads because it does not release the flow resources - this is the responsibility of the threaded joiner.

Waiting For Completion Of Threads

To await completion of threads, use the pthread _ join () function. It has the following syntax:

int pthread_join (pthread_t the ,, void ** status);

This function blocks the thread where it was called until the thread specified by the handle deserves its execution. If the status is not a null pointer, it will point to the data returned by the stream after the call (argument pthread _ exit ()). If the thread is already complete by the time the pthread _ join () function is called, its information should be read immediately.

Conclusions

- Processes and threads are the active resources of computer systems that implement code execution. The thread is called a set of sequentially executed processor commands. A process is a collection of one or more threads and the protected address space in which they are executed. Streams in the same address space can share data together, and for streams of different processes without special means, this is not possible. In traditional systems, each process can execute only one thread code execution of processes. Modern operating systems support the concept of multithreading.
- Using threads in an application means paralleling it - the ability to perform actions simultaneously with different snippets of code. Program parallelism reflects the asynchronous nature of the outside world, its sources being code execution on multiple processors, I / O operations, user interaction, and client application

requests. Bahatopotokovist allows applications to implement this course and achieve parallelism and efficiency.

- There are user threads running in the slugger mode in the process address space, and the kernel threads with which the OS kernel works. The relationship between them determines the scheme of implementation of the flow model. In practice, they often use the 1: 1 scheme, when each thread of the user corresponds to one thread of the kernel, and the kernel itself is responsible for managing the threads of the user.
- The flow can be in different states (execution, expectations, readiness, etc.). The principles of moving from one state to another depending on the principles of flow planning and processor time planning. Moving the process from execution state to any other state is about switching the context - transferring control from one thread to another while preserving the state of the processor.
- Each process and flow in the system corresponds to its control unit - a data structure containing all the necessary information. During the creation process (usually by the system call fork ()) create one control unit, allocate memory and launch the mainstream. Creating flow is faster and easier because they do not need to allocate memory to the new address.

Chapter 4:
Process and Flow Planning

The ability to run streams in parallel depends on the number of processors available. If the processor is single, parallel execution is impossible in principle (only one thread can be executed at any one time). If the number of processors is N > 1, parallel execution can only be implemented for N threads (one thread per processor).

The main purpose of scheduling for a single-processor system is to arrange multiple threads on a single processor that would give the system user the impression that they are running at the same time. This definition can be extended to multiprocessor systems if a scheduling task occurs when the number of threads exceeds the number of available processors.
Consider the basic types of planning, their principles, and algorithms.

General Principles Of Planning

Consider the basic principles underlying planning.

Features Of Thread Execution

In terms of scheduling, flow can be depicted as a cycle of alternating computation (CPU usage) and I / O periods. The time interval during which the flow executes only the instructions of the processor is called the CPU usage interval (CPU burst), the time interval when the flow is waiting for I / O, the I / O burst interval. Most of these intervals are from 2 to 8 ms.

Streams that spend more time computing and less - on the IO referred *disabilities processor (CPU bound)* . They actively use the processor. Their main characteristic is the time spent on computing, the intervals of CPU usage for them are longer. Flows that take most of the time are pending IO referred *disabled input-output* (I / O bound). Such threads load the processor much less, and the average length of the processor usage interval for them is small. The higher the clock speed of the processor, the more flows can be attributed to the second category.

Processor-limited flow (matrix multiplication)
I / O restricted stream (text editor)

Interval input-output (I / O bound)
and the interval CPU usage (CPU bound)

Planning Mechanisms and Policy

The planning *mechanism* and *policy* should be differentiated. Scheduling mechanisms include context switching tools, flow synchronization tools, etc., and scheduling policies are tools for determining when to switch contexts. The part of the system that is responsible for policy planning called *scheduler* (scheduler) and the algorithm used to this - *scheduling algorithm.*

There are various evaluation criteria for planning policies, some of which apply to all systems, others to batch systems or interactive only. Today, three criteria are most commonly used to evaluate goal achievement.

- *Minimum response time* . This is the most important criterion for interactive systems. Response time refers to the time between running a stream (or typing an interactive command) and getting the first response. For modern systems, we have taken the time of 50-150 ms
- *Maximum bandwidth* . This is the number of tasks that the system can perform per unit of time (for example, per second). Such a criterion is appropriate to apply in batch systems; in interactive systems, it can be used for background tasks. To increase bandwidth, you must:
- Reduce wasted load time (for example, the time it takes to switch context)
- Use resources more efficiently (to ensure that neither the processor nor the I / O devices are idle)
- The third criterion is *equity* , which means that the CPU time is allocated according to their importance. Fairness provides such a distribution of processor time that all threads advance in their execution, and none proto. Note that implementing a fair planning policy does not always reduce the average response time. Sometimes this requires making the system less fair

Planning Principles

The principles of flow planning are applicable first of all to multithreaded systems with 1: 1 scheme implementation (only kernel flows are planned here), as well as to systems with the implementation of process modules. In the latter case, the term “process” can be used instead of the term “flow” and the information necessary for planning can be stored in the structures of these processes. More complex scheduling principles are used in multi-threaded systems for which the number of user threads does not match the number of kernel threads (Schemes 1: M and M: N). They require two schedulers: one for kernel-level operation, the other for user mode.

Types Of Planning

Distinguish between long - term scheduling, medium-term scheduling, and short-term scheduling.

Long-Term Planning

Long-term planning means determining which of the programs should be in the vicinity to perform the download. This scheduling is also called static because it does not depend on the current state of the system. It played an important role in batch systems when it was known in advance what processes needed to be completed and the tasks could be scheduled. In interactive systems (eg systems with time-sharing) loading processes in principally engaged users, and it is not subject to planning; so they usually use a simplified long-term planning strategy. The system allows you to create processes and threads to reach some maximum limit, and then further attempts to create a new process or flow will cause an error. This strategy is also based on the psychology of users, who, feeling uncomfortable in an overloaded system, can interrupt work with it, which leads to reduced load.

Medium-Term Planning

Medium-term scheduling controls the transition of paused flows to standby and back. In the control unit which is ready for execution, threads are immediately organized in a structure called queue of ready threads (or ready queue).

The transition to a suspended state can be caused by the following factors:

- Waiting for an I / O operation
- Waiting for the completion of another thread (joining)
- Blocking a stream because it needs to be synchronized with other streams

Usually, for the correct organization of this expectation, in addition to the queue of ready threads, they implement an additional set of queues. Each such queue is associated with a resource that may cause flow expectations (eg, an I / O device); These queues are called scheduling queues or *waiting for queues* (wait for queues). The mid-term scheduler manages all these queues by moving threads between them and the queue of ready threads.

Short-Term Planning

Short-term planning or *scheduling processor (* CPU scheduling) is the most important type of planning. It answers two basic questions:

- When to interrupt a stream?
- Which of the ready-to-execute threads should the processor transfer at this point?

A short-term scheduler is an OS subsystem that interrupts the active flow in the event of a need and chooses the one that should be executed from the queue. Its performance is top-notch because it receives control very often. There is also a controller (dispatcher), which directly controls pre-selected streams (context switches) .

The format of the queue of ready flows depends on the implementation of short-term planning. This stage can be organized on the principle of FIFO,

set the priorities, wood or disordered, which is called the related list.

All of the planning strategies and algorithms that we will look at below are short-term planning.

Planning Strategies - Displacement And Non-Displacement Multitasking

Before considering basic planning strategies, let's list the options for transferring controls from one thread to another:

- After the flow has entered a standby state (for example, during input-output or connection)
- After completion of the flow
- Explicit (the thread itself gives the processor other threads for a while until it is busy)
- Interrupts (for example, a timer interrupt can interrupt a stream that is running longer than allowed)

The latter option differs from the others in that the flow cannot control when the control transfer time has come, which is the responsibility of the operating system scheduler. Depending on this variant of control transmission, there are two main strategies for flow planning - displacement and non-displacement multitasking.

In pre-emptive multitasking, logical flows may be temporarily interrupted without the need for them to transfer control to other flows. Interrupting the flow and transfer control to another stream often done in the interrupt handler of the system (and also the mayor). This strategy is implemented in all modern operating systems.

In multitasking, flows may be performed during an unlimited time and cannot be used in interrupted OSes. For non-redundant multitasking, the latter control transmission is not implemented, and the streams themselves must give control of the OS for transmission to other threads or at least go

to standby. If a thread forgets or fails to do so, for example, it will take the processor an endless loop, other threads will not be able to continue their work. This strategy was implemented in Nowell Net Ware.

Naturally, the implementation of non-redundant multitasking in the general case makes the system quite unstable (any incorrectly written application of the user can cause a "hang" of the whole system). The practice has shown that non-redundant multitasking in systems with user applications cannot be implemented. Such a strategy, however, can be used in systems where all applications are executed in kernel mode and are actually system drivers. The development of such applications requires high skills of programmers, requirements for reliability of applications can be compared with the requirements for the OS itself. With the simplicity of implementation and absence of external interruptions of flows from the OS, the scheduler can increase system productivity for the limited range of tasks (for example, in the case of Net Ware OS it was to use the system as a file server).

Planning Algorithms

The scheduling algorithm allows a short-term scheduler to choose from the ready-to-run threads the one to be executed next. It can be said that planning algorithms implement planning policies.

Depending on the planning strategy implemented by the algorithms, they are divided into displacement and non-displacement. Displacement algorithms interrupt flows during their execution, non-displacement algorithms do not interrupt. Some algorithms fit only one of these strategies, others may have both displacement and non-displacement implementations.

FIFO Planning

Consider the simplest ("naive") indelible algorithm in which flows are executed in order of their appearance on the system and executed before

going to standby explicit transmission or completion. The queues of finished streams are organized according to the FIFO principle, so the algorithm is called the FIFO algorithm.

As soon as a new stream is created in the system, its control unit is added to the tail of the queue. When the processor is released, it is provided to the stream from the queue head.

This algorithm has many disadvantages:

- It is by definition irreversible
- The average response time for it may be quite significant (for example, if the former receives a stream with a long CPU usage interval, other streams will wait even if they themselves use only short intervals)
- It is subject to the convoy effect (convoy effect)

The effect of the convoy can be explained by this situation. Suppose there is a single thread (let's call it T_{CPU}) in the system, limited by the processor capabilities, and many T_{io} threads limited by the I / O capabilities. Sooner or later, the T_{CPU} thread will be available to the processor and will execute instructions with a long interval of use of the CPU. During this time, the other T_{io} streams will complete the I / O, queue the finished threads and wait there, with the idle I / O devices. When T_{CPU} finally block and will transfer control of all flows T_{io} quickly follow the instructions of their ranges CPU and then move to the IO. After that, T_{CPU} will again capture the processor for a long time, etc.

Circular Planning

The easiest to understand and the fairest algorithm is a *circular plan* algorithm (round-robin scheduling). In the Middle Ages, the term “round-robin ” was coined in petitions where signatures were done in a circle so

that one could not know who signed up first (this name indicates that for such an algorithm all flows are equal).

Each stream allocated time period in which this stream is allowed to run is called a *time slice* (or time quantum). When the thread is still running after the time quantum is exhausted, it is interrupted and the processor switches to perform instructions from another thread. When it blocks or completes its execution until the time quantum runs out, the processor is also passed to another thread. The length of time quantum for the whole system is the same.

Such an algorithm is quite easy to implement. For this, the turn of the ready threads should be a cyclic list. When a stream has exhausted its quantum of time, it is moved to the end of the list, where new flows are provided. Checking the exhaustion of the quantum of time is performed in the interrupt handler from the system timer .

The only characteristic that affects the operation of the algorithm is the length of the quantum of time. The balance between time spent switching contexts and having to respond to many simultaneous interactive queries should be kept in mind here.

Tasking too short a quantum of time leads to a lot of context switches, and a significant percentage of CPU time is spent not on useful work, but on those switches. On the other hand, the task of a too-long quantum, while saving CPU time, also reduces the response time for interactive queries because if ten users press a key at the same time, ten threads will be in the ready list, resulting in the last one expecting ten long quanta time. In the case of a quantum of infinite length, circular scheduling is reduced to the FIFO algorithm (all threads have time to block or end before the quantum is exhausted). In practice, it is recommended to set the quantum length to 10 - 100 ms.

Note that traditional circular scheduling can "skew" toward streams that are constrained by processor capabilities. Such threads preferably utilize their quantum completely, whereas threads limited by I / O often transmit control until the quantum runs out, and as a result, they have less processing

time. To solve this problem, you can increase the length of the quantum (given the problems described earlier) or enter an additional queue of completed I / O streams, which takes precedence over completed queues .

Priority Planning

Circular duty scheduling assumes that all flows are equally important. Otherwise, *prioritized planning* should be used. The basic idea is simple: each thread is given priority, while the highest priority stream from the queue of ready threads will be executed. Priorities can be given statically or dynamically.
One of the approaches to the implementation of planning priorities algorithm is *multi-queue* (multilevel queues). In this case, several bursts organize groups with different priorities streams (streams each group usually have different purposes).

The decision to choose the thread to execute is made as follows:

- If the highest priority streams are streams, they should use a simpler scheduling algorithm (such as circular scheduling), without regard for streams in other queues
- If there are no threads in the queue, the threads with the lower it priority move to the queue

Different scheduling algorithms can be used for different queues, and each queue can be allocated a certain amount of CPU time. Sharing priorities is a difficult task, and failing to resolve them can cause streams of low priority processes to wait a long time. For example, in 1973, a machine was stopped at the Massachusetts Institute of Technology, which found a low-priority process - it was queued for execution in 1967 and has not been able to run since. This situation is called *starvation.*

There are different ways to solve the problem of starvation. For example, a scheduler can gradually reduce the priority of the stream they perform (this process is called aging), and when it becomes lower than the next priority

stream, switch the context to that stream. You can, on the contrary, gradually increase the priority flows waiting in UT.

Planning Based On Performance Characteristics

An important class of prioritized scheduling algorithms is algorithms in which decisions about the choice of flow to execute are made on the basis of knowledge or evaluation of the characteristics of its further execution.

First of all, it should be noted that in the " *first - with the shortest time of execution* " algorithm (Shortest Time to Completion First, STCF), when associated with each flow duration interval following the use of the CPU, each is selected to perform at the stream in which this short interval. As a result, streams that capture the CPU for a shorter time get an advantage during scheduling and leave the system faster .

The STCF algorithm is theoretically optimal by the criterion of the average response time, that is, it can be proved that for the selected group of flows, the average response time in the application of this algorithm will be minimal compared to any other algorithm. Unfortunately, for short-term planning, it is impossible to implement it because this implementation requires anticipation of the expected characteristics. For long-term planning, it is used quite often (in this case, when setting the task for the operator, the operator must indicate the expected time of completion, which the system will take into account when choosing). Note also that the optimality of such an algorithm is inseparable from its "unfairness" to flows with longer CPU usage intervals.

For short-term scheduling, an approximation to this algorithm may be implemented based on an estimate of the length of the next processor usage interval, taking into account previous intervals of the same flow. You can use recursion in the formula to calculate this estimate

$t_{n+1} = aT_n + (1- a)\, t_n , 0 \leq a \leq 1, \quad t_0 = T_{0,}$

where t_{n+1} - estimate the length of the interval;

t_n is an estimate of the length of the previous interval;

T_n is the true length of the previous interval .

The most commonly used values are a = 0.5, in this case, it is sufficient to calculate the average between the previous estimate and the actual interval value.

Shortest Remaining Time to Completion First, SRTCF.

Its difference from STCF is that when a new thread is queued for ready threads, the next CPU usage time is shorter than the time remaining before the current thread completes, the current thread is interrupted and a new thread becomes in its place.

Multilevel Feedback Queues

Algorithms *of multi bursts with feedback* (multilevel feedback queues) are scheduling algorithms (using settings it can be reduced to any other algorithm), but one of them is used in the implementation. In terms of organizing data structures, this algorithm is similar to the conventional multilevel queue algorithm: there are several queues of ready threads with different priorities, with lower priority queue threads only running when all upper-level queues are empty.

The differences between the two algorithms are:

- Flows are allowed to cross from level to level
- Flows in a line are united not by priority and the length of the use interval flows with shorter intervals are in line with higher priority

Within all queues, except the lowest, use circular scheduling (the lowest FIFO algorithm works). The different queues correspond to different lengths of time quanta - the higher the priority, the shorter the quantum (usually the length of the quantum for adjacent queues is halved). If the flow has exhausted its time quantum, it moves to the tail of the queue with a

lower priority (and with a longer quantum). As a result, flows with shorter intervals (such as limited I / O) remain with high priority, and flows with longer intervals extend their time. You can also automatically move non-managed streams from the lower tier to the tier in and out.

Lottery Planning

Planning lottery (lottery scheduling) - algorithm, easy to understand and easy to implement and it has great potential. The idea of lottery planning is that:

- The stream receives an insane amount of lottery tickets, each of which entitles the processor to use T for a time
- The scheduler selects one lottery ticket at random over a period of time T
- The winnable stream gets control to the next dra w

Lottery planning allows you to:

- Emulate circular scheduling, giving each stream the same number of tickets
- Emulate priority scheduling, distributing tickets according to stream priorities
- Emulate SRTCF - give short streams more tickets than long ones (if the stream received at least one ticket, it would not starve)
- Ensure that the processing time is split between threads - give each thread a number of tickets proportional to the fraction of processor time it needs to allocate (for example, if there are three threads, and it is necessary for thread A to occupy 50% of the processor and threads B and C, 25 each %, it is possible to give stream A two tickets and flows B and C one each)
- To change priorities, selecting and adding tickets on the go

Although most of these tasks can be solved by lottery planning, only approximately, with some likelihood, in practice, they are quite

satisfactory. The longer the system works, the closer the results will be to the theoretical values (by the law of large numbers). In fact, lottery planning takes advantage of the fact that all planning ideology is largely heuristic since one cannot accurately predict the nature of flow behavior in the system .

Implementation Of Planning In Linux

Consider two options for implementing Linux scheduling - the traditional one (which includes kernels up to 2.4 inclusive) and the new one included in the 2.6 kernels. The Linux kernel does not distinguish between processes and flows when planning, so for certainty, we will continue to discuss process planning. All processes in the system can be divided into three groups: real-time with FIFO scheduling, real-time with circular scheduling, normal.

Real-Time Scheduling Of Kernel Processes

Regarding real-time processes, it is enough to say that:

- They will always have priority over normal processes when planning
- The FIFO scheduling process is performed until it itself gives in to the processor (for example, as a result of suspension or termination) or until it is superseded by a higher-priority real-time process
- The same applies to the circular scheduling process, except that it will be further displaced after the quantum of time is exhauste d

Traditional Scheduling Algorithm

Consider an algorithm for scheduling ordinary processes. At the core of the algorithm is the allocation of CPU time to *the epoch* (epochs). During the epoch, each process has a quantum of time, the length of which is calculated at the time of the epoch. For the most part, different processes

have quanta of different lengths. When the process has exhausted its quantum, it is displaced and during the current era, it will no longer be performed. Management is given to another process. If, however, the process is paused for I / O or due to synchronization, its quantum is not considered exhausted and it may be exhausted by the scheduler during the current era. The era ends when all ready-to-complete processes have exhausted their quanta. In this case, the scheduling algorithm recalculates the quanta for all processes and begins a new era.

The quantum, which is set at the beginning of the era, is called the *basic quantum of the processing time.* Its values can be dynamically modified by the nice () and setpriority () system calls. The descendant process always inherits the basic quantum of its ancestor.

There are two types of process priority: fixed, for real-time processes that are set only during the process creation, and dynamic, for normal processes, which depend on the basic priority and the time remaining until the quantum is exhausted. Dynamic priority to any conventional process is always lower than the priority of any real process for the first time.

We describe the most important fields of the process data structure for planning:

- Policy - determines to which group the process belongs (normal, real-time FIFO algorithm, etc.)
- Nice - specifies the value on which the base quantum of the processing time is based (in the future, for simplicity, we will consider nice to be equal to the base quantum; in fact, this is not quite so)
- Counter - Contains the number of timer interrupts left before the process quantum is exhausted. At the beginning of the era of a counter, the values of the base quantum are reduced and reduced by one in the timer interrupt handler

Planning Procedure Call Conditions

Consider a situation where a scheduling call (called a schedule ()) occurs.

- When the process needs to be blocked because the resource it needs is unavailable at this time. In this case, its control unit is first added to the corresponding waiting queue, and then the redevelopment takes place
- With the *delayed start* (lazy invocation). The delayed startup is that at a certain point in time, special field needs _ reached process structures are assigned a value of 1. This is echoed in the following cases: when the current process has exhausted its quantum; when a process whose priority is higher than the current one goes into standby; when the process is clearly inferior to its right to execute through an appropriate system call. This redevelopment happens immediately, but later, when the process must regain control after the interrupt, it checks whether the field is needed _ reached unit. If equal, the scheduling process is started

Planning Procedure

This procedure first checks that the current process is not in the standby state, and if so, removes it from the process queue. Then a process is selected for execution. To do this, view the queue of finished processes, for each process evaluate the dynamic priority and select the process and with its maximum value. For a process that has run out of its quantum of time, it will be zero. If no process has been selected, the current process continues. When the choice is made, the context is switched to a new process .

The Beginning Of A New Era

A special situation arises when for all processes in the queue of ready processes the value of dynamic priority is zero, that is, they have all exhausted their quantum, and it is time to start a new era. However, this does not mean that the system does not have processes for which the

quantum is not exhausted - they can be queued (most often, these are processes that are limited by I / O.

When a new era opens, the quantum is recalculated for all system processes (not just standby processes). In this case, the length of the quantum for each process is set equal to the sum of its base priority and half of the remaining quantum in it:

```
for each_task (p)
   p.counter = (p.counter / 2) + p.nice;
```

Since the beginning of a new era of quantum remains nonzero only in processes that are not in a state of readiness, this algorithm provides an advantage to processes disabilities output. In this case, the quantum value for the process can never be greater than twice its base priority value .

Dynamic Priority Calculation

Let us return to the calculation of dynamic priority. To do this, use the goodness () function. Consider the possible values that it can return.

- 0 - when the process has run out of time. This process will not be selected for execution. Except when it is in the queue of finished processes first, and all other processes in the queue have also exhausted their quantum
- 0 to 1000 - if the process has not exhausted its quantum of time. This value is calculated based on the value of the base quantum of the process and the portion of the current quantum remaining in it. This can be summarized as follows:

c = p.counter + p.nice; where p is a pointer to the control block of the process.

This implies that the longer the process takes to complete and the longer it's base quantum, the higher its priority. In addition, further increasing the

value per unit for processes that use the same memory as ancestors (for example, if the process shows the flow created by using the clone ()) .

Recounting A Quantum When Creating A New Process

Consider what happens when you create a new process. The simplest solution (to copy the counter value to the descendant data structure) can cause processes to artificially extend their quant by creating new descendants that execute the same code. In order to prevent this, after the fork () function, the values of the counter are split in half: one half goes to descendants, the other remains ancestral.

List the disadvantages of the algorithm.

- The choice of process to execute is due to the calculation of the dynamic priority for all processes in the queue of finished processes. With the increasing number of ready processes in the system, it is unprofitable to view the queue from start to finish during each call of the scheduling procedure
- If the number of processes is very large, it will take a long time to list all the dynamic priorities at the beginning of a new era. On the other hand, epochs change less often with more processes in the system
- The algorithm is designed to reduce response time for processes with limited I / O capabilities, even if they are not interactive (eg, background search engine indexing) and do not require a short response time
- With the increase in the number of processors, maintaining common queues that do not take into account the presence of different processors becomes unprofitable

Modern Approaches To Planning Implementation

These shortcomings began to significantly affect the operation of the system when it was operating under load conditions. Under normal circumstances, traditional Linux scheduling worked quite effectively. Work on fixing the shortcomings has continued. As a result, a new implementation of the scheduling algorithm has been integrated into the kernel of version 2.6. Consider briefly how it helps the solution ' bind previously mentioned problems.

First of all, this algorithm supports separate queues of finishing processes for each processor, ensuring efficient operation under conditions of multiprocessing. Another problem that has solutions connected with the need to rely on the old algorithm for the dynamic priority of all ready processes during each procedure call planning. The decisions are as follows: each queue of the digestion processes is an array of ready-made process queues, where the elements are sorted by dynamic priority. As a result, when choosing a process to execute, it is enough to look at the highest priority queue for the first process that can be started. This procedure does not depend on the number of ready processes with a STEM.

There are two arrays of queues of ready processes - an array of queues of active processes and an array of queues of processes with an exhausted quantum. After the process has exhausted its quantum, it is transferred from the first array to a long one. When there is no process in the active queue, both arrays are replaced by missions, and the sequence of steps is repeated from the beginning. As a result, the depletion of quanta processes increases the likelihood of running those processes that have not yet been managed.

The Programming Interface Of Planning

Let's take a look at the Linux system calls that can handle the underlying process priority (the value nice) and thus influence their scheduling.
To change the base priority of the process, use the call set priority ():

#include <sys / resource.h>
int setpriority (int which, int who, int priority);

In particular, the parameter which can be set to PRIO_PROCESS or PRIO_USER, respectively, indicating that the parameter will be interpreted as a process identifier or user ID. In the first case, they set a priority for a specific process (or for the current process, if who equals zero), in the second - for all processes of that user.

The priority parameter sets a new priority. Priority may vary from –20 to 20, smaller values indicate higher priority. The default value is 0. Negative priority values can only be set by users with admin privileges.

Use the getpriority () call to get information about the current base priority :

Int getpriority (int which, int who);
This call returns the priority value, the which and who parameters for it have the same meaning as for the setpriority () function.
Here's an example of how to use these calls:
// set a priority for the current process
setpriority (PRIO _ PROCESS , 0.10);
// find out about the current priority value
printf (" current priority: % d \ n ", getpriority (PRIO _ PROCESS , 0));

You can also use the nice () system call to change the underlying priority of the current process relatively :
# Include < unistd . h >
int nice (int inc); // changes the priorities of the current process to inc .

Conclusions

- The task of scheduling a thread is to organize the execution of multiple threads on a single processor so that users have the impression that they are running at the same time. Planning goals are: minimizing response time, maximizing throughput and fairness. Major planning strategies include displacement and non-displacement multitasking. In modern operating systems, multitasking

is used when decisions about switching the flow context are made in the system kernel code and not in the flow code.

- Distinguish between long-term, middle-term and short-term planning. The most important here is a short-term scheduler, which is used to decide which flow to run at a given time. Basic short-term scheduling algorithms include circular and priority planning.

Chapter 5:
Flow Interaction

Let's consider the basic principles of interaction of flows of one process. The focus will be on synchronizing access to the shared data of such streams.

Basic Principles Of Flow Interaction

Threads that are executed in a parallel process can be independent or interact with each other. A stream is independent if it does not affect the execution of other process threads, does not detect the impact on their part, and has no data in common with them. Its implementation is clearly dependent on the input data and is called deterministic.

All other threads are interacting. These threads have data shared with other threads (they are in the address space of their process). Their implementation depends not only on the input data but also on the execution of other threads, ie they *are non-deterministic* (let us consider further examples of such non-deterministic).

Independent thread execution results can always be repeated, which is not the case with interacting threads. Data Worms are common to several streams, called shared data. This is the most important concept of multi-threaded programming. Any stream can change this data at any time. Mechanisms for ensuring correct access to shared data are called flow synchronization mechanisms.

Working with independent streams is easier than interacting with threads. A programmer may disregard the fact that others are running concurrently with such threads, and ignore the state of the shared data that the thread is working with.

However, there are several reasons why you cannot implement thread interaction.

- It is necessary to organize the sharing of information when working with streams. For example, database or web server users may want to

request the same information at one time, and the system must ensure that it is received concurrently by the threads that serve those users

- ❖ Proper implementation of this interaction and the use of appropriate algorithms can significantly speed up the computing process on multiprocessor systems. This task is divided into subtasks that perform simultaneously on different processors, and the results are collected together for the final solution liabilities. This technology is called *parallel computing technology*
- ❖ In tasks requiring concurrent execution of I / O calculations, I / O threads should be able to signal to other threads with the completion of their operations
- ❖ Such an organization makes it possible to divide tasks into separate executable modules, designed as separate threads, which, at the same time, the output of one module can be an input for another, and also increases the flexibility of the system, since individual modules can be changed without tapping others

The need for concurrent execution of interacting streams requires mechanisms for exchanging data between them and ensuring that they are synchronized.

The Main Problems Of Flow Interaction

The Problem Of Competition

Let's look at a simpler example of what can happen when interacting streams share common data without additional synchronization steps. Suppose that in banking organization systems for the service of each user allocate a separate flow (then trying to improve the system performance in the case of a large number of simultaneous requests). Suppose that putting data on a user's contribution is reduced by increasing the global variable total _ amount. At this time, each sweat and k when changing the deposit is executed by the following simple operator:

```
total_amount = total_amount + new_amount;
```

The question is: is it possible to guarantee that, due to the contribution, the flow, corresponding to each user, will be able to increase the value of total_amount by the required value?
In fact, this at first glance is the simplest operator boils down to a sequence of actions:

- Get the current value of total _ amount of global memory ' memory
- To increase its new _ amount and store the result in a global memory ' memory

Consider the case where two users A and B share the same account. There are 100 monetary units in the account. User A is going to deposit 1000 User at the same time - 100 Flow T_A corresponding to the user A, T flow $_{in}$ - the user B.

Consider the following sequence of events (option 1).

1. Stream T_A reader mentioned total _ amount, equal to 100
2. The stream T_B reads the value total _ amount, also equal to 100
3. Stream T_A increases the read values in step 1 total _ amount in 1000, received 1100 and stores it in memory
4. The T_B stream increases the value of total _ amount in step 2 by 100, gets 200, and also stores it in global memory, overwriting what the stream $T_{A\ has}$ saved

User A's contribution is completely lost.

Now consider another sequence of events (Option 2).

1. Stream T_A reads total _ amount, increase it to 1000 and records the value 1100 in the global memory
2. Stream T_{in} reading total _ amount, equal to 1100, it increased to 100 in 1200 and writes the value into a global city

As a result, both contributions were successfully registered.

As you can see, the result of executing the simplest code snippet above depends on the sequence of execution of threads in the system. This leads to the following: in one situation the code may work, in the other - no, and in general it is impossible to predict the occurrence of an error. This situation is referred to *as race* (or race condition), which is one of the most difficult bugs captured faced by programmers. It is practically non-negotiable (since it is unrealistic to override all possible combinations of flow execution sequences, especially if there are many).

Attempts to bind these necessitated problems *into synchronization streams.* Immediately note that the problems of synchronization and organization of parallel computing are some of the most difficult in practical programming. Therefore, the development and debugging of multithreaded programs in particular often seen as a kind of "art" that is not known to all programmers.

In fact, such development and debugging is not an art, but a strict discipline, subject to one basic principle: because for multithreaded programs, traditional debugging is not suitable, the programmer must write the code so that at the stage of development, there is room for errors. Let us familiarize ourselves with the rules that must be followed in order for the generated code to comply with this principle. Consider the main approaches to competition problems.

Sometimes (but rarely enough) you can just ignore these mistakes. This may make sense when we are not interested in the accurate recording of particular data, but collect statistics about them, so individual errors will not

affect the overall result. For example, a global counter is a value on which to calculate the average number of requests to the system per day, and you can ignore the error of registering such requests that occur every few hours. Unfortunately, in most cases, this approach is not accepted.

Sometimes, the use of global data is not dictated by the specifics of the task. In this case, the straightforward solution is to create local copies of data for each stream and operate them only. In practice, this works very well and should be used wherever possible. For example, if the specifics of the problem allow the creation of a separate counter for each thread (or global array of counters, where each element is changed only by a certain flow), the implementation of such data structures solves the problem. Again, such a solution cannot be applied in all cases (for example, in the situation with bank accounts, different flows, after all, must somehow modify the total for all accounts).

In all other cases, changes must be protected against the effects of other flows. This is the main task of synchronization. Consider different approaches to solving it.

Critical Sections And Locks

The Concept Of Critical Section

Consider using a simpler idea to solve a competition problem. It is easy to see how the source of our mistake is that, from the outside, the simplest money-transferring operation actually breaks down into several operations, with the chance of any other flow interfering between them. In this case, it is said that the operation is not an atom but a molecule.

It follows that the solution to the problem of competition is to transform the code snippet that causes the problem into an atomic operation, that is, one that is guaranteed to be performed completely without interference from other threads. This piece of code called a *critical section* :

// start critical section

total_amount = total_amount + new_amount;
// end of critical section

Now, when two threads are going to execute the code of the critical section at the same time, the one that started the first one will execute all its code completely before the second starts its execution (the second thread will wait until the first one completes the code of the critical section). As a result, we are guaranteed to have a sequence of events in Option 2 in our program, and the competition will never take place.

Consider the properties that a critical section should have.

- ❖ Mutual exclusion: In a particular time code, a critical section can execute only one thread
- ❖ Progesterone if multiple threads are present at the entrance to the critical section, one of them must be necessarily required to enter into it (they cannot block each other completely)
- ❖ Limited mode, a process that tries to enter a critical section, sooner or later is necessarily required to enter into it

The simple question remains: "How can we make the system perceive multiple operations as one atomic operation?". The simplest solution to such a task would be to prohibit interruptions for the duration of the critical section. This approach, although solves the problem in principle, in practice, cannot be applied as a result because the cycling program in the critical section of the whole system can remain as locked interrupts, and therefore, present an unfavorable condition.

Locking

The tidier solution is to use *locks.* A lock is a mechanism that prevents more than one thread from executing the code of a critical section. Using a lock comes in two steps: commit (lock, acquire _ lock ()) and unlock (lock, release _ lock ()). In the case of a lock, it is checked that it has not already been made by another thread, and if so, this thread enters the

standby state, otherwise, it enforces the lock and enters the critical section. After leaving the critical section, the flow removes the blockage .

```
acquire _ lock ( lock );
// critical section
release_lock (lock);
```

So realize mutual exclusion property, hence another name for blocking is mutex, short for mutual exclusion.

Problems With Implementation Of Blocking

Consider a naive implementation of a critical section using *block variables.* With each critical section associated integer variable, which assumes the value as a unit during freezing and zero after forging. Here's what the code for this implementation looks like:

```
int lock = 0;
void aquire_lock (int lock)
{
// if lock does not have ( lock == 0), roll it (set lock = 1)
// and exit - we are in the critical section
// otherwise wait until the lock is removed
while (lock! = 0); // (1)
lock = 1; // (2)
}
void release_lock (int lock)
{
// remove the lock
lock = 0;
}
```

The main problem with this implementation is that checking the value of the locking variable and changing it is not part of a single atomic operation.

Basic Mechanisms Of Flow Synchronization

Modern OSes provide a wide range of ready-made synchronization mechanisms. *Synchronization mechanisms* are operating system tools that help to solve the main task of synchronization - to ensure the coordination of flows that work with shared data. If such tools are the minimum blocks for building multithreaded programs, they are called *synchronization primitives.*

Synchronization primitives are divided into the following main categories:

- Versatile, low-level, which can be used in different ways (*semaphores*)
- Simple, low-level, each of which is adapted to the solution
- High-level universal expressed through simple; this group includes the concept of a *monitor* which can be expressed through mutexes and conditional variables
- High level, adapted to the solution of a specific synchronization problem (blocking read-write and barriers)

Let us look at the different mechanisms and evaluate the advantages that each of them has, as well as their possible disadvantages. To further illustrate the material, let's take a classic example that demonstrates the need for synchronization - the problem of producer-consumers or the problem of limited buffer (bounded buffer).

The formulation of the problem is simple. Suppose we have production threads and consumer threads. The manufacturer creates objects, the consumer receives them. The task is to synchronize their work so that the consumer cannot attempt to retrieve objects that have not yet been created, and the manufacturer cannot create more objects than the consumer can.

To synchronize flows, place a buffer of fixed length *n* between the manufacturer and the consumer. The manufacturer can place objects on the buffer, the consumer - take them from there. If the consumer takes the

object, it is removed from the buffer. There are several requirements to be met:

1. When the manufacturer or consumer is working with the buffer, the rest of the threads should wait for it to finish
2. When the manufacturer is trying to put an about object in the buffer and the buffer is full, it must wait until it appears in place
3. When a consumer tries to pick an about object of the buffer and the buffer is empty, it must wait until it is on ' appears about ' object

Semaphores

Concept semaphore proposed in 1965 by E. Dijkstra - known Dutch specialist in Computer Books Science. Semaphores are the oldest synchronization primitives of those used in practice. Semaphore - a shared net ample integer counter asked for which initial value and determined following atomic operations:

- *Down Semaphore (down)* : If the value of the semaphore is greater than zero, it is reduced by one, if the value is zero, this stream goes to standby until it is greater than zero (everyone who is "waiting at the semaphore" or "Blocked at the semaphore"). This operation is also called *expectations* - wait. Here is her pseudocode:

```
void down (semaphore_t sem)
{
   if (sem> 0)
       sem--;
   else sleep ();
}
```

- *Increase semaphore* (up): increase semaphore values by one; when there are streams waiting at the semaphore, one of them

goes out of the waiting and performs its own operation. If more than one thread is expected on a semaphore, its up value remains null due to the up operation but one of the threads continues to execute (in most implementations, the choice of this thread will be random). This operation is also called post- *signaling* . Here is her dog in the doc:

```
void up (semaphore_t sem)
{
   sem ++;
   if (waiting_threads ())
      wakeup (some_thread);
}
```

In fact, a semaphore value determines the number of threads that can pass through that semaphore without blocking it. When a semaphore is set to zero initial value, it will block all threads until some thread "opens" it by performing the up operation. Operations up and down can be performed by any flows that have access to the semaphore.

Features Of Use Of Semaphores

Semaphores can be used to solution ' Liabilities synchronize two different tasks.

1. With their help, it is possible to organize mutual exclusion (to protect the code of critical sections from execution by more than one thread). It is most convenient to use a *binary semaphore* , which can take two values: 0 and 1. Here is an example of implementing a critical section using a binary semaphore:

```
semaphore _ t sem = 1; // at the beginning the semaphore is open
down (sem);
// critical section
up (here);
```

2.

With their help, you can organize the expectation of the fulfillment of some condition. Suppose that one needs to organize the wait for one thread to complete another (analog of the join operation). In this case, a semaphore with an initial value of 0 (closed) can be used. The stream that is waiting must perform a down operation for this semaphore and in order to signal the completion of the stream, its completion function requires up to perform the same semaphore. Here is the pseudocode (this_thread stands for current thread):

```
Void thread_init ()
{
   this _ thread. sem = 0; // At the beginning of the stream execution the semaphore is exposed
}
void thread _ exit ()
{
   up ( this _ thread . sem ); // wake up the pending stream, if any
}
void thread_join (thread_t thread)
{
down (thread.sem); // Waiting for thread to finish
}
```

Realization Of The Problem Of Producers-Consumers With The Help Of Semaphores

First, let's call the synchronization actions required to solve this problem.

1.

Put the operations of direct buffer change (for the manufacturer - placing the object in the buffer, for the consumer - removing the object from the buffer) in the critical sections

2. Arrange expectations according to requirement 2 (manufacturer's expectations in case of full buffer). In doing so, the consumer must inform the expectant manufacturers that he has taken the object from the buffer (ie the buffer has become incomplete, if complete)
3. Arrange expectations according to requirement 3 (consumer expectations in case of full buffer). In doing so, the manufacturer must inform consumers who expected that the matched buffer is non-empty and if it was empty.

Now let's look at the synchronization primitives we need. Each synchronization operation requires a separate semaphore:

1. To use the critical section, we use the binary semaphore, as we did before. Let's call it to lock. It will be used by both the manufacturer and the consumer, protecting access to the buffer from other threads (again, both manufacturers and consumers)
2. In order to organize the manufacturer's expectations in the case of a full buffer, we will need a semaphore whose current value is equal to the number of free spaces in the buffer. Let's call it empty _ items. The manufacturer, before attempting to add a new object to the buffer, reduces this semaphore to the standby state if it is equal to 0. The consumer, after removing the object from the buffer, will increase the semaphore by informing the manufacturers (and waking up one of them)
3. In order to organize consumer expectations in the case of an empty buffer, a semaphore whose current value is equal to the number of seats in the buffer will be required. Let's call it full _ items. The consumer before attempting to pick up an object from the buffer reduces this semaphore, going into a state of waiting

Here is the pseudocode solution to this problem:

```
semaphore _ t lock = 1; // for the critical section
    semaphore _ t empty _ items = n;// 0 buffer full, empty from the beginning
    semaphore_t full_items = 0; // if 0 - the buffer is empty

    // manufacture r
    void producer ()
    {
       item_t item = produce (); // create about ' object
       down (empty_otems); // is there a place for information ' facility?
       down ( lock ); // enter the critical section
       append_to_buffer (item); // add about ' object item to the clipboard
       up (lock); // exit critical section
       up ( full _ items ); // inform consumers that there is a new object
    }

    // consumer
    void consumer ()
    {
       item_t item;
    down ( full _ items ); // not empty buffer?
    down ( lock ); // enter the critical section
    item = receive_from_buffer (); // pick about ' Object item from the clipboard
    up ( lock ); // exit critical section
    up ( empty _ items ); // notify manufacturers of the location
    consume ( item ); // consume the object
    }
```

Mutexes

The concept of mutex largely coincides with the notion of blocking defined previously (Section 5.2). Synchronization primitives called Mutexes are the ones that prevent the execution of a piece of code in more than one

stream. In fact, the mutex is an implementation of OS-level blocking. Mutex, as its name implies, eliminates mutual exclusion. Its main task is to block all threads trying to access the code when that code is already executing a thread. A mutex can be in two states: the free and the busy. The initial state is "free" and has two possible atomic operations.

- *Take mutex* (mutex _ lock): if mutex was free, he is busy, and the flow continues its execution (entering the critical section); if the mutex was busy, the stream enters a standby state (said to be a "wait on a mutex" or "locked on a mutex"), the execution continues another stream. Stream, which took mutex called *the owner of the mutex:* mutex _ lock (mutex _ t mutex)

```
{
   if (mutex.state ==== free)
   {
      mutex.state = locked;
      mutex.owner = this_thread;
   }
else
   sleep ();
}
```

- *To release the mutex* (mutex _ unlock): mutex becomes free; if several streams are expected on it, one of them is selected, it starts executing, occupies the mutex and enters the critical section. In most implementations, the flow selection will be random. Only the owner can release the mutex. Here is the pseudocode of this operation:

```
mutex_unlock (mutex_t mutex)
{
   if (mutex.owner! = this_thread)
      return error;
   mutex.state = free;
   if (waiting_threads ())
```

```
        wakeup (some_thread);
    }
```

Some implementations provide a third operation try to take the mutex (mutex_trylock): if the mutex is free, to act similarly to mutex_lock, if busy - to immediately return the error and continue execution.

Here is the simplest implementation of the critical section using a mutex.

```
mutex_t mutex;
mutex_lock (mutex);
// critical section
mutex_unlock (mutex);
```

The main difference in the mutex of binary semaphores is that the release of mutex can only be done by its owner, while the semaphore can change the value of any stream, which has access to it.

Interprocess

So far, we have considered the interaction of flows of one process. The main feature of this interaction is the ease of technical implementation of data exchange between them - one process flow using one address space and therefore, can freely access the co-usage of variant data as if they were their own. Since there are no technical difficulties with the implementation of data exchange, the main problem that needs to be solved in this case is the synchronization of flows.

On the other hand, each thread is executed within the address space of a process, so the task of organizing interaction between the flows of different processes is often a problem. The question itself about interprocess interaction (interprocess communication or IPC) .

For streams of different processes, the issues of synchronization are also relevant, but in most cases, they are not based on the concept of shared data (such default data for processes are missing). In addition, the challenge of securing exchange between protected address spaces is added. Approaches to its solution define different types of interprocess interaction.

Types Of Interprocess Interaction

Implementation of inter processing is accomplished by three basic methods: *message passing, memory allocation,* and *displayed memory.*
Another method of IPC is to consider the signal technology discussed earlier.

Technology Expanded Memory (Mapped Memory)

In a number of operating systems, the displayed memory is the basic system mechanism on which other types of interprocess interaction and system decisions are based. Of course, the display memory used in conjunction with the file system interface, in this case, talking about *the files displayed in the memory* (memory-mapped files).

This technology is to ensure that a special system call (usually mmap ()) a part of the process address space uniquely floor containing the file. After that, an operation to write to such memory causes the contents of the displayed file to change, which immediately becomes available to all applications that have access to that file. Other applications can also display the same file in their address space and communicate with one another through it.

Conclusions

- Streams of one interacting process have access to shared data hosted in the address space of that process. Any stream is able to change this data at any time and cause a state of competition when the outcome depends on the flow sequence.
- Flow synchronization mechanisms are used to ensure correct access to shared data; the main one is to ensure mutual exclusion when only one stream can access such data at a particular time. Blocking is used to organize mutual exclusion.
- Various synchronization primitives can be used to solve synchronization problems. The simplest primitives include

semaphores, mutexes, conditional variables, read-write locks, and barriers.

- A higher-level mechanism is a monitor concept that combines mutexes and conditional variables and sets some rules for their interaction to protect shared data. Using monitors is the most consistent approach to stream synchronization.

Chapter 6:
Managing Memory

Different types of memory are organized into a hierarchy. At the lower levels of such a hierarchy, there is a cheaper and slower memory of a larger volume, and as the hierarchy moves upwards the memory becomes more expensive and faster (and its volume becomes smaller). The cheapest and slowest storage device is the hard drive. It is also called secondary storage. Faster and more expensive memory is stored in the memory chips installed on the computer - a memory called the main memory. Faster storage facilities have different cache processors, and at least, these caches are more limited.

Managing memory is a rather difficult task. The required memory characteristics are often not enough identified, and in order not to interfere with the work of the user, it is necessary to implement means of coordination of different types of memory. Thus, the current application cannot fit entirely in the main memory, while unused code can use tymch and COBOL stored on the hard drive. In this section, we look at technologies that use primary memory.

Fundamentals of Virtual Memory Technology

We first consider the prerequisites for introducing the concept of virtual memory. In this situation, each process is loaded into its own continuous portion of physical memory, and the next process begins immediately after the previous one. If you analyze the memory allocation features of this approach, the following questions may arise.

- How do I perform processes that require more physical memory than installed on my computer?
- What happens when the process performs a write operation at the wrong address (eg P2- at 0x7500)?
- What should I do if a process (such as P1) needs more memory to execute?
- When will the process receive information about the specific physical memory address, which memory will start executing it, and how should the memory addresses used in its code be converted?

- What should I do if the process doesn't need all the memory allocated to it?

Direct loading of processes into physical memory does not provide an answer to these questions. Obviously, some memory translation tools are needed to allow processes to use address sets that are different from physical memory addresses. Before we understand the peculiarities of these addresses, let us briefly dwell on the peculiarities of composing and downloading applications .

The program usually resides on disk as a binary executable file obtained after compilation and layout. It must be loaded into memory (process address space) for its execution. Modern architectures allow processes to be located anywhere in physical memory, and the same program can accommodate different processes loaded into different memory locations. It is unknown in advance which memory area the program will be loaded into.

During execution, the process accesses different addresses, in particular, when a function is called, it uses its address (this is the code address), and the global variable accesses the memory address to store the value of that variable (this is the data address).

The programmer does not normally use memory addresses directly in his program, instead, they use symbolic names (functions, global variables, etc.). Due to compilation and layout, these names are linked to moving addresses (such addresses are specified in relative units, such as "100 bytes from the beginning of the module"). When running the program, the relocated addresses, in turn, are bound to absolute addresses in memory. In fact, each bind is a mapping of one set of addresses to another.

- *Memory protection* . Addressing errors that occur in a process code should only affect the execution of that process. When the P2 process does a write operation at 0x7500, it must be interrupted by an error. The memory security strategy is that a valid address range is stored for each process, and each memory access operation is checked for an address belonging to that range

- *Lack of binding to physical memory addresses* . The process can be performed regardless of its memory location and physical memory size. The process address space is allocated as a large static address set, with each address of the set being moved. Processor and hardware should be able to convert these addresses to physical addresses of main memory (the same address placeable at different times or for different processes may correspond to different places including his address)

The Concept Of Virtual Memory

Virtual memory - a technology that introduces I level additional conversions between memory addresses used by the process and physical memory addresses used by the desktop app. Such transformations should ensure that the memory is protected and that the process is not bound to physical memory addresses.

With virtual memory, the physical memory of the process address space can be fragmented, since the bulk of the memory occupied by the process remains free most of the time. There is a so-called ninety to ten rule or a localization rule that states that 90% of memory accesses in the process account for 10% of its address space. You can move the addresses so that only the sections of the process address space that are actually in use at that time correspond to the main memory.

In this case, unused partitions of the address space can be mapped to slower memory, such as hard disk space, and at this time other processes can use the main memory, which previously displayed the addresses of these partitions. When a partition is needed, its data is loaded from disk to main memory, perhaps instead of partitions that became unnecessary at a particular moment (and which, in turn, will now be stored on disk). Data can be read from disk at a village thus a key memory LDPE time addressing them.

In this way, you can significantly increase the size of the process address space and ensure that processes larger than the main memory are executed.

Problems of virtual memory implementation: Memory fragmentation

The main problem that occurs when using virtual memory is the efficiency of its implementation. Because address conversions need to be done every time memory is accessed, careless implementation of these conversions can have the worst effect on the performance of the entire system. If most memory accesses make the system actually have to access the disk (which is tens of thousands of times slower than the main memory), it will be virtually impossible to operate such a system. The issue of improving the performance of virtual memory implementation will be discussed.

Another problem is the memory fragmentation that occurs when free memory cannot be used. There are *internal* and *external memory fragmentations.* The external is reduced to the fact that due to the allocation and subsequent freeing of memory, it creates free blocks of small size - *holes* . Because of this situation may arise in which it is impossible to provide a continuous memory block size N, because there is no contiguous free block of size $S \geq N$, although in general, the amount of free memory space more than N. So in the above case, there is not enough space to execute the P5 process due to external fragmentation.

Internal fragmentation is the result that, on request, memory blocks larger than actually used are allocated, resulting in unused areas inside the dedicated blocks that can no longer be allocated to something else.

Logical And Physical Memory Addressing

The most important concept of the concept of virtual memory is addressing the logical and physical memory .

Logical or Virtual Address - An address generated by a program running on some processor. Addresses that use the instructions of a particular processor are logical addresses. A set of logical addresses is called a logical address space.

Physical Address - The address that the memory chip operates on. An application in modern computers never deals with physical addresses. A special MMU (memory management unit) is responsible for converting logical addresses into physical ones. The totality of all available physical addresses is called a physical address space. So, if your computer has a 128MB chipset, then that amount of memory is physically addressed. Usually, much more memory is logically addressed.

The specifics of converting logical addresses into physical ones are determined by different approaches to managing memory, the study of which will be the topic of this section.

Approach Of Base And Boundary Registers

When implementing virtual memory, it is necessary to protect memory, move processes in memory, and share memory across multiple processes. One of the easiest ways to meet these requirements is to approach the *base and boundary registers* . For each process in two CPU registers store two values - *the base address* (base) and *limits* (bounds). Each access to a logical address is converted to a physical address by adding a logical address to the base address. If the received physical address does not fall within the range (base, base + bounds), consider that the address is incorrect and generate an error.

This approach is the simplest example of a dynamic movement of processes in memory. All the other approaches that will be discussed in this section are different options for developing this basic scheme. For example, the fact that each process has its own values of base and boundary registers when using this approach is the simplest implementation of the process address space concept, which is based on the fact that each process has its own memory mapping.

To organize memory protection in this situation, it is necessary that the use of the user cannot change the value of the base and border registers. It is enough to make such changes available only in the preferred mode of the processor.

The advantages of this approach include simplicity, modest hardware requirements (only two registers are required), and high efficiency. However, today it is practically not used because of a number of disadvantages, due primarily to the fact that the address space of the process is still mapped to one continuous block of physical memory: it is unclear how to dynamically expand the address space of the process; different processes cannot share memory; no code and data distribution.

In this approach, only one pair of base address-limit values is allocated to the process. The natural evolution of this idea was the mapping of the process address space through several physical memory ranges, each of which sets its own pair of the base address and boundary values. That's how the concept of memory segmentation came about.

Memory Segmentation

Features Of Memory Segmentation

Memory segmentation allows you to depict a logical address space as a set of independent variable-length blocks called segments. Each segment usually contains data for one purpose, for example, one may have a stack, the other, a program code, etc.

Each segment has a name and length (for ease of implementation, use names alongside names). The logical address consists of a segment number and an offset within the segment; the application program works with such segments. Compilers often create separate segments for different application data (code segment, the data segment, stack segment). When a program is loaded into memory, it creates a table of process segment descriptors, each element of which corresponds to one segment and consists of a base address, boundary value, and access rights.

When forming an address, its segment portion points to the corresponding element of the process segment descriptor table. If the offset is greater than the limit setpoint (or if the process access rights do not match the segment set rights), then the hardware generates an error. When all is well, the sum

of the base and offset in the case of pure segmentation will result in a physical address in the main memory. If a segment is uploaded to a disk, trying to access it causes it to be loaded from the disk into the main memory. As a result, each segment corresponds to a contiguous block of memory of the same length, located in an arbitrary location of physical memory or on a disk. Here are the benefits of memory segmentation.

- There is an opportunity to organize several independent segments of memory for the process and use them to store data of different nature. However, the access rights to each such segment can be set differently
- Individual segments can be shared by different processes, and their segment descriptor tables must contain the same elements that describe that segment
- Physical memory corresponding to the process address space should not necessarily be continuous now. In fact, segmentation allows individual portions of the address space to be mapped, not to the main memory, but to the disk and reloaded as needed, ensuring that processes of any size are executed

This approach is not without its disadvantages.

- The need to introduce an additional layer of memory conversion causes a decrease in performance (this drawback is inherent in any full implementation of virtual memory). Appropriate hardware support is required for effective segmentation implementation
- Managing variable length memory blocks with the need to store them on a disk can be quite complicated
- The requirement that each segment corresponds to a contiguous block of the physical memory of the appropriate size causes external memory fragmentation. Internal fragmentation does not occur in this case because the segments have a variable length and you can always select a segment of the length required to execute the program

Today, segmentation is used rather limited, primarily because of the fragmentation and complexity of implementing efficient memory and disk-

sharing. Wider use of memory allocation got into blocks of fixed length - page memory organization that will be discussed .

Segmentation Implementation In IA-32 Architecture

In the IA-32 architecture, logical addresses in the program are formed using segmentation and look like a "shift selector". The values of the selector are loaded into a special register of the processor (segment register) and used as an index in the table of descriptors of the segment, which is in memory and is an analog of the table of segments, described earlier. Six segment registers are supported in IA-32 architecture. This means that the executable code at the same time can address these six independent d cops.

The selector contains a descriptor index in the table, bits of the local or global table indicator, and the required privilege level.
For system set common *global descriptor table* (Global D e scriptor Table, GDT), and for each task - *local descriptor table* (L o cal Descriptor Table, LDT). The descriptors in IA-32 are 64 bits long. They define the properties of software objects (for example, memory segments or descriptor tables). Handle contains the value base (base), which corresponds to the address of the object (for example, the beginning segment); limit value (limit); object type (segment, descriptor table, etc.); protection characteristics.

Access to tables of descriptors is supported by hardware. If the security characteristics specified in the handle do not correspond to the privilege level specified by the selector, it will not be possible to access the memory using it. They provide memory protection. However, it has never been mentioned that a physical address is stored in the handle. The fact is that for the IA-32 architecture, due to the conversion of the logical address, they receive not another physical address, but another type of address, which is called a linear address.

Page Memory Organization

The main technologies of virtual memory implementation other than segmentation owned -*page memory organization* (paging). Its main idea is the allocation of memory blocks of fixed length. These blocks are called *pages and so are we* . This technology is the most common approach to virtual memory implementation in modern operating systems.

Basic Principles Of Page Organization Of Memory

In the case of page memory organization, a logical address is also called a linear or virtual address. Such addresses belong to one set (for example, a 32 bits non-negative number can be a linear address). Physical memory is divided into blocks of fixed length - *frames* , or page blocks (frames). Logical memory, in turn, divided into blocks of the same length - *page* . When the process starts, its pages are loaded into available physical memory frames from disk or other media. The paging organization must have hardware support. Each address generated by the processor is divided into two parts: *the page number* and the *page offset* . The page number is used as an index in the page table.

A page table is a data structure containing page - table entries (PTE), each of which contains information about the page number, the number of the corresponding physical memory frame (or directly it's the base address) and access rights. The page number is used to find the item in the table. After finding it, the page frame is added to the base address of the corresponding frame, which determines the physical address.

Page size is a power of 2, modern OS uses pages ranging from 2 to 8 KB. In custom addressing modes, you can work with larger pages. For each process, they create their own page table. When the process begins to run, the OS calculates its page size and the number of frames in physical memory. Each page is uploaded to a corresponding frame, after which its number is written to the process page table.

The process of memory mapping is different from the actual physical memory state. At the logical level, all memory is represented by a

continuous block and belongs only to this process, and physically it is spread over the address space of the memory chip, alternating with the memory of other processes. The process cannot access memory whose address is not specified in its page table (so memory protection is implemented) .

The operating system must have information about the current state of physical memory (frame occupancy, frame number, etc.). This information is usually stored in a *frame table* . Each element corresponds to the frame and contains all the information about it.

Comparative Analysis Of Page Organization Of Memory And Segmentation

Page memory organization and segmentation have more in common than differences. The main difference between the two approaches is that all pages have a fixed length and segments have a variable length. Other basics (no requirement for continuity of physical memory, ability to unload memory blocks to disk, need to support conversion tables, etc.) are fundamentally different. Let's take a look at the main benefits of page memory organization over page memory segmentation. They are determined primarily by the fact that all pages have the same length.

- The implementation of memory allocation and memory is simplified. All pages are equal in terms of process, so you can maintain a list of free pages and, if necessary, select the first page from that list and, upon release, return the page to the list. The segments do so not because each segment can only be used for its purpose (try to use a segment for other purposes likely lead to what will be needed in the segment to some other thing)
- Disk communication with the disk is also simplified. To arrange such an exchange, the portion of the disk used to store page information discharged from memory (*support* space or backing store) can also be broken down into blocks of fixed size equal to the size of the frame

The page organization of memory is not without its drawbacks.

- First of all, this approach causes internal fragmentation due to the fact that the page size is always fixed, and if necessary to allocate a memory block of a specific length, its size will be multiple of the page size. On average, the amount of unused memory is approximately half a page for each dedicated memory block (similar to a segment). This fragmentation can be reduced by reducing the number and size of the blocks that appear and fuse
- Page tables should be larger than segment tables. For example, to allocate a contiguous memory range of 100 KB you will need one element of the segment table that describes the segment allocated for that range. On the other hand, if we use 4KB pages to describe this range, we will need 25-page table elements, one for each pag e

Multi-Level Page Tables

To address the logical address space of a large volume with a single page table, it has to be very large. For example, in an IA-32 architecture with a standard page size of 4 KB (addressing inside such a page requires 12 bits) the index in the table remains 20 bits, which corresponds to a page table of 1 million elements.

To avoid such large spreadsheets, the technology of *tiered page* tables is proposed. Page tables themselves are broken down into pages whose information is stored in top-level page tables. The number of levels rarely exceeds 2, but can reach 4.

When there are two levels of tables, the logical address is broken down into an index in the top-level table, an index in the lower-level table, and offset. This technology has two main advantages. First, page tables are smaller in size, so they can be searched faster. Second, not all page tables should be in memory at a specific point in time. For example, if the process does not use some memory block, then the contents of all the lower-level pages of an unused block may be temporarily stored on the disk .

Implementation Of Page Tables In IA-32 Architecture

The IA-32 architecture utilizes a two-tier page organization, starting with the Intel 80386. A table-top level called *directory pages* for each task should be given a separate directory page, a physical address that is stored in a special control register and where it automatically loads the hardware when switching context. Table bottom level is called a *page table* .

The linear address is divided into three fields:

- *Directory* (Directory) - defines the directory entry pages, indicating the desired page table
- *Table* (Table) -vyznachaye page table element that indicates the desired frame memory
- *Offset* (Offset) - determines the offset within the frame, combined with the address of the frame forms a physical address

The size of the directory and table fields is 10 bits, giving page tables containing 1024 elements, the size of the offset field is 12 bits, giving pages and frames 4 KB in size. One lower-level page table addresses 4 MB of memory (1 MB of frames) and the entire directory of pages has 4 GB .

The page table elements at all levels have the same structure. Here are the following element fields:

- *Flag presence* (Present), equal to one of the page is in physical memory (it corresponds to the frame); equality of this flag means that the page is not in physical memory, and the operating system may use other element fields for its purposes
- *The 20 most significant bits* that set the frame start address to be multiple of 4 KB (1 MB of different start addresses can be specified)
- *Flag Access* (Accessed), which is equal to one pin each time the reference device support paged to the corresponding frame

- *Flag Changes* (Dirty), which confer equal to one while recording each transaction in the corresponding frame
- *Flag read-write* (Read / Write), which specifies the rights to access the page or page table (for reading and write or read-only)
- *The privilege* (User / Supervisor) *check* box that specifies the processor mode required to access the page. If this checkbox is zero, the page can only be addressed from privileged mode, if units are also available from user mode

Presence, access, and change checkboxes can be used by the OS to organize virtual memory .

Associative Memory

When implementing page tables, you have to access memory several times to access the physical memory byte. If two-tier pages are used, three access operations are required: the page directory, the page table, and directly to the address of this byte, and the three-level tables require four operations. This slows down memory access and reduces overall system performance.

As already noted, the ninety to ten rule indicates that most memory access to a process belongs to a small subset of its pages, and the composition of this subset changes rather slowly. A way to improve the performance of a page-based memory organization is to cache the addresses of the memory frames corresponding to this subset of pages.

To solve this problem was proposed technology *associative memory* or *cache translation* (translation look-aside buffers or TLB). In high-speed memory (faster than main memory), they create a set of several elements (different architectures assign associative memory from 8 to 2048 elements, in the architecture of IA-32 such elements to Pentium 4 were 32, starting with Pentium 4 - 128). Each element of the translation cache corresponds to one element of the page table.

Now when generating a physical address, the corresponding element of the table is first searched for in the cache (in IA-32 - by the directory field, the table field and the offset), and if it is found, the address of the corresponding frame becomes available, which can be immediately used to access the memory. If there is no corresponding element in the cache, then memory is accessed through the page table, and then the element of the page table is stored in the cache instead of the oldest element.

Unfortunately, when switching contexts in the IA-32 architecture, the entire cache needs to be cleared, since each process has its own page table, and the same page numbers for different processes may correspond to different frames in physical memory. Clearing the translation cache is a very slow operation that must be avoided in every possible way.

An important feature of the translation cache is the percentage of hits, that is, the percentage of times that the required elements of the page table is in the cache and does not require memory access. It is known that at 32 elements 98% of hits are provided. Note also that this percentage decrease in performance hits when using a two-level page table compared to the single-level is 28%, but the benefits derived when allocating memory would be making a decline.

Conclusion

Thank you for making it through to the end of *Linux For Beginners* , let's hope it was informative and able to provide you with all of the tools you need to achieve your goals whatever they may be.

In Linux, unlike other operating systems, by default, the work with Linux is not done in a graphical way, but by entering commands manually. Linux has several programs that are responsible for interpreting the commands entered by the user and perform the appropriate actions in response. These programs called shell are the typical mode of communication on all UNIX systems including Linux. For many people having to enter the commands manually may seem intimidating and difficult, although as you must have seen in the book Linux commands are relatively simple and very powerful.

Linux is also a multitasking and multi-user operating system and it is capable of running several programs (or tasks) simultaneously and host several users simultaneously. Therefore, all users of Linux must have a user account on the system that establishes its privileges. Linux organizes users into groups so that privileges can be established for a particular workgroup, for access to certain files or system services.

Finally, there is another fundamental concept when installing and using Linux, which is a Super User or User root. This user is the system administrator and is created during installation. As an administrator, you can access and modify (as well as destroy) all system information, so try to avoid working as a root user as much as possible. We hope you learned all of this and a lot more is seen from this book.

Finally, if you found this book useful in any way, a review on Amazon is always appreciated!

Linux for Hackers

The Advanced Guide on Kali Linux Operating System to Change Your Computer into an Underground Hacking Machine and Master the Science of CyberSecurity, Networking and Scripting Tools

Darwin Growth

Introduction

In this book, we shall learn how an operating system called Linux works and how hackers can use this tool to help them hack. Linux is an operating system (OS) created by Linus Torvalds early in the 1990s and is a popular choice to work with today. The main part behind this to ensure that it works is the Linux Kernel which is available for free. Unlike Microsoft Windows which is easier to use, Linux may seem a bit more difficult to work with. But this is mostly because we have to actually do some coding and use the command line, rather than the graphical interfaces, to make things work and to bring up the programs we want.

An operating system helps you go about tasks on a computer like writing, browsing the internet or doing many other activities. If the installation of the operating system is not on your desktop, then you will be unable to use your computer. Even for you to be able to read this content you need an operating system. The kernel is the heartbeat of the Linux system, and they can be of different types such as Ubuntu, Fedora, and Debian.

Here, you will get to understand the different types of Linux distributions, how each of the distribution work and which one can best suit your preferences. The skills needed for you to start the hacking techniques to enable a smooth hacking process. With the detailed review of what you need, then getting started will become easy for you.

Chapter 1:
Preview of the Linux Operating System

Preview of How Linux Works

Linux Kernel distinguishes this OS from other OSs. The Kernel is the coding used to develop Linux, and over the years, many developers keep upgrading the version. That is why for you to hack using Linux, you must understand how Linux Kernel works. The Kernel is in the innermost part of the operating system where it does jobs like:

- managing the hardware.
- Do the essentials services of the OS.
- Allocates resources evenly.

When you want to execute the Kernel, then you need to implement it in the kernel-mode as oppose to the user mode. So, any applications running in the system communicates with Kernel using what we call "*system call*". The system enables the user to have a smooth interaction utilizing an interface to reach the Kernel. Did you know that every function that possibly interacts with the system automatically translates to a system call? Here is an example

```
Void main ( ) {
}
```

Let's say this is one of the most obvious things you will see. Here it gains access via the main entrance and later exists. To get the picture of what we are saying, here is a picture of the levels of kernels.

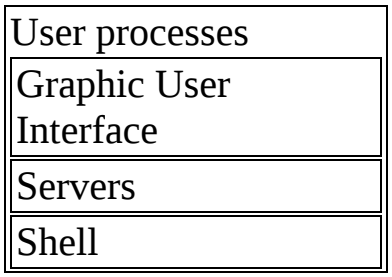
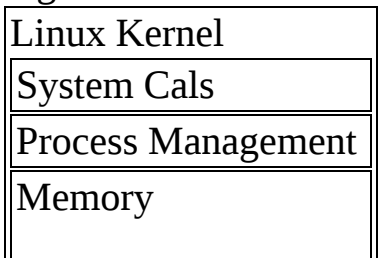


Figure 1.1



Management
Device Drivers

Figure 1.2

Hardware
Processor
Main Memory
Disks
Network parts

We shall explain in detail how the kernel and hacking are related as we go through this guidebook. But before we get there, it is fundamental to understand some basics; this is just an introduction.

Relationship Between the Applications, Hardware, and Kernels

From the diagram, we can see that the user process runs in the user mode whereas, the Kernel runs in the kernel- mode. That is if a code is running in the kernel mode can access processor (CPU) and the main memory without any restrictions. Although this might make you happy, this feature of the kernel can easily crash a whole system down.

The user mode, therefore, restricts access directly to the processor. That is why when any users mistakenly go to the processor and alter the functioning of the CPU; it crashes the whole system. Where cases of a crash are not mild, then the kernel tries to clean up the messes done. Although if you had running programs, then they will not work as usual. When you alter the user-face, it may not cause severe damages compared to the kernel .

Hardware is the physical part of the computer holding the memory of the machine that is the Random-Access Memory. It stores information that has been saved in files and also assisting the central processing unit in loading files faster and efficiently. All inputs and outputs of the computer run

through the main memory. There are some memories found in the internal memory that you need to understand.

Primary Memory – here we have the Random-Access Memory (RAM) and Read-Only Memory (ROM).

1. RAM – Data stored in this form can be easily accessed anytime when a user does a prompt. This type of memory is exceptionally volatile, where you can write or read. Suppose power goes off, then this memory cannot be traced. We have different types of Random-access Memory.

- SRAM – This is static random-access memory. It is one of the most expensive memories you can ever get. There are several versions of this memory such as DDR2, DDR3, and DD4, all giving the user a better performance than DDR. It consists of six capacitors in each cell.
- DRAM – Dynamic random-access memory is the most commonly used type of memory in computers. Compared to SRAM, this kind of memory is slower with a capacitor and transistor in each cell.

1. ROM – As the name suggests this type of data can only be read but unable to write. It is a fast kind of data for the users. Unlike random-access memory, when the power goes off, one can access this data later on. We can say, even when power is off the computer does not need software to help the CPU communicate.

This memory has millions of transistors and capacitors. Now, when these two parts are combined, they form what we call a memory cell denoted by a single bit. A capacitor bucket stores electrons though at times the bucket may leak. The dynamic random-access memory helps recharge the emptied buckets before doing the disbarment. The process is repeated repeatedly in very many cycles to enable the processor to read and write memory back.

This is a key point to understand since, at times during hacking our data can be traced if you do not understand which memory to work with. When you can to hack the memory of a computer, you can do stuff like Vacuums, game exploits, good modes and much more. In any case, if memory manipulation is impossible, then it will be challenging to do the manipulation.

The kernel is the heartbeat of Linux. Figure 1.1 gives an overview of what the Linux kernel does. In the process part of it, the Kernel selects which process is to be allowed first before the other. In the memory, Kernel helps keep a record of memory like the amount of memory to allocate to a particular process or the amount of memory that can be shared. In the

device, drivers act as an interface between the hardware and processes, for processes to have a communication to the kernel they use system calls.

Benefits of Using Linux

Here the benefits of using Linux:

1. Security – Compared to other operating systems, Linux is the most secure OS to use. You will find that other operating systems are very vulnerable when malware attacks it. What makes it more interesting is that before any virus attacks or authorized logins, one has to insert the password.
2. Reliable – It has excellent stability that enables the users to use it efficiently - no need of keeping rebooting your computer because of some errors.
3. Free to use – It is possible to go to your browser and download the operating system without asked to insert the activation key.
4. Multitask – You are able to perform different tasks using the system without a lot of difficulties like we find in the other operating systems. It does not decrease its speed or efficiency even when you need to take on a lot of tasks. It is great since it provides a hacker with the opportunity to monitor different things all at once.
5. Different distributions – With the Linux system it is possible to do many distributions, these distributions are called bistros.
6. Open source - Something good about Linux is that its sources are available for everyone. Depending on the exact thing that you want to do, then you are free to execute, add or modify the codes; which is a rare thing found on the other operating systems.
7. Updates – When using Linux, you will notice so many software updates prompts. The software updates are easy to updates which takes a few seconds for them to updates.

You’ll notice that it has unique advantages when using it as opposed to another operating system. All these features combined give you the need to use this OS’s when doing the hacking. Ubuntu is the straightest forward Linux language to use. This is because you will learn how to use it once you understand how the commands get executed.

Chapter 2:
Hacking and the Skills Hackers Need

What is Hacking?

Before going ahead with some of the techniques that we can use with hacking, first, you need to understand what hacking is in detail. When we talk about hacking we are looking at a process that will actively search for vulnerabilities in the system. Once you find these vulnerabilities, then you use the weakness to help you gain access illegally. This is done by cracking the operating system algorithm to be able to access without a password.
When computers interact with each other externally, they run the risk of getting hacked. So, who is a hacker? This is the person behind the hacking. He tries to access the computer and tries to get the weakness of a computer. This means hackers are very skillful with computers for them to be able to know how to alter with data manipulation – their classification is the following categories.

- Phreaker – Tries to identify a weakness in a telephone.
- Script kiddies – this type of hacker is not skilled. He uses tools online to enable him to do the hacking.
- Hacktivist – This kind, hacks websites and leaves messages on the hacked sites. The message he leaves maybe about political, religious, social, global, government issues related. Most often, they are negative messages.
- Black hat – His main aim is to access a computer to get data mainly for self-gain. He intends to violate the copyrights or steal money.
- Ethical hacker – He is also known as a white hat. He hacks the system of the computer to identify the weakness of a computer and fix it. They can test a system to see if it is vulnerable to hackers.
- Grey hat – This person hacks a computer system without authority. His main aim is to identify the problem and tell the system owner.

Therefore, ethical hacking is the most recommended since you get access to the computer using by legally accessing. As an ethical hacker, make sure you get permission to access the system owner. You have the mandate to secure your clients' data from being hacked by users instead inform the necessary personnel. This person can protect data or if you are a guru, then do the protection.

Types of Hacking Techniques and Precautions to Take

Here are some of the hacking methods that a hacker uses while hacking:

Passive attacks - This type is where the hacker tries to hack the computers by looking at the weakness available. The main mission done by this type of hacking is to get information about a specific company and not necessarily change any information.
The attack can either be active or passive reconnaissance. In the active reconnaissance the attacker may try to access through ports scans while in passive, the attacker monitors the activities done.

Methods of Passive

- Dumpster diving – Here the intruder gets some of the information from dustbins. These are saved passwords in the dustbin. They use the passwords to access the locked information and covert the system entries. This tells you not to save passwords in your computer or forgetting to delete them permanently in case anything happens to your machine, system or network.
- War diving - This method is done by intruders getting vulnerable WIFI networks through the processing of scanning. Mainly, this is done by a portable antenna by a moving vehicle. Many people like accessing free WIFI in public places; this gives the intruders the chance to access those open networks easily. The main aim of this hacking method is to access internet connections.

Waterhole attacks – This name was inspired by these predators who like staying near the watering hole to attack the prey. Now, the intruders stay near the niche of the sites to hunt when the users access the website and try to attack using malware.
They attack the site by sending affected Html codes when the user clicks on the site it redirects on other sites. Making them unable to access the specific information they were looking for. Afterward, the compromised site infects the targeted users once they access the link.

Ways to Defend Against Waterhole Attack

- Two-factor authentication – You will find what they steal your passwords and username. To prevent being attacked use two-factor authentication which, makes it hard for them to guess the codes or any other way used in this way.
- Disable tools vulnerable to watering hole – Some software is easily affected or vulnerable to attacks so if possible, remove them. Some of this software is adobe's flash.
- Security audit – Depending on which country you are, you can decide to get a security audit in one of the companies well known in IT services.
- Update software – Attackers take advantage of hacking a system because of not updating software regularly. Security is key for every business database so keep updating some of the software on your computer or else you become a victim of hacking.
- Monitor network – Have a regular update monitoring, this helps you keep track of those people visiting the site. SSL – traffic checker can help you do that.

Keylogger – It is also called a monitoring software. It is like a CCTV that monitors every activity taking place loggings, downloads, chats, etc. They do that by logging into your keyboard so that you do not suspect that anything is going on. They can access sensitive information such as credit card numbers, passwords and things like account information.
Keylogging can be done in a positive way, like parents monitoring their children from going to dangerous sites that may lead to cyber crimes and frauds. Sometimes it might be difficult to

detect if you have a keylogger. You can note when your keyboard keeps redirecting you to a certain site, slower performance than usual or screen error messages .

How to Protect Against a Keylogger

- Password manager – Most of these password managers use the feature of autofill that helps you remember the password. But did you know this is dangerous? Yes, hackers can save the password. Although some keyloggers save passwords when typed on keyboards, it is good to check whether if your computer/system has this suspicious keylogger. As a company, a keylogger can be used to check unusual activities.
- Be cautions – This happens almost to everyone where you receive links from unknown people. Some of these links are infected and are sent to prompt you to click them. The click of those links is the beginning of the hacking. You'll find once you click on these links your computer starts misbehaving or redirecting to specific sites.
- Keyboard layout – Perhaps, try to change the design of your computer. Keylogger is available in QWERTY layout, try changing it to DVORAK. This is because it becomes hard to convert the captures in that layout.
- Security tools – To put extra security measures in your computer, you need to put additional security tools. Some tool like Ghost press is an anti-keylogger that stops other programs from running behind the system without knowing.
- Change passwords – In case you suspect someone is logging to your mails then try to put the hard password but easy to remember. A hacker may steal your password and not necessarily use it immediately. Keep changing your password just in case in future he tries to hack the system unexpectedly.

Cookie theft – Cookie data is available when in transit. It is always sent in the form of headers in sites this makes them vulnerable to attackers. In public places, this becomes riskier since they access easily.

Cookie Manipulation

- Did you know cookies can be manipulated? Yes, cookies are manipulated. Most users like saving their data using cookies, only save non-sensitive data with cookies. The reason being, an attacker, will definitely use your cookies to get some sensitive information.
- An alternative for this is saving your data in the server-side session. The session is more of a database file that keeps records of your personal information. It helps the user avoid the danger of data being manipulated.
- To help yourself from the manipulation, put cookies expiry dates. So, after a while, the information will be no more, thus lingers it from exploitation. Visiting non-SSL sites puts one in danger of exposing their data to malicious sites. SSL encrypted sites are more saver sites to visit since they have secured cookies that prevent you from damage.
- Another way to save yourself from cookie manipulation is by using suitable algorithms. The algorithms help in both encryption and decryption of values. This is very significant because supposing the cookies are in plain text and fail to operate. Then they are never prompt to attacks because the algorithms in place are hard to crack.
- Additionally, people decide to sign their cookies as a protection defense tool.

Bait and switch – This seems tricky kind of hacking technique used in today's world. What hackers do, is hack a website/a system and start posting articles not related to that site. The

company itself may not beware of what is happening. The attackers then prompt you with the ad's not necessarily related to the site.
They insert links that only Google sees them while the users are unable to see. They benefit from this since some of those links are ads. Accessing your computer becomes easier since when you download the contents, virus attacks your computer. You've probably seen something like, a banking site having bitcoins articles. That should keep you thinking and make you realize that the contents are manipulated.

Avoiding Bait and Switch Tactics

- Free things affect human beings. Almost everybody will click to an ad when they see the word "FREE". Let us look at it from a real-world perspective; here are two companies both offering the same product. Company A tells you I will offer the product with $10 while company B offers you for free. To be real, many people or you yourself will go to company B.
- You have to note that there is nothing like free things in this world. Actually, there's a hidden cost behind free things. There are two possibilities, either there's a deal that will make you spend some cash in the near future or a form of bait to make you likely to get into some extra cash right away. Free things make people make a hasty decision which in the long run, cost you a lot.
- Avoid exposing your personal information. Some adverts want you to sign up with your details like email, phone number and such. Be cautious just because they said you would earn $30 on your first sign up .

Hacking Skills with the Linux System

Suppose you want to be a hacker then know you need to know programming languages. This is a kind of language that will help you program computer programs. They are of different varieties varying from one to another. Should I learn how to program, is it necessary anyway? Yes, it is. Find out why you need to understand the languages:

- Sometimes you may encounter an error while hacking. Understanding some errors that occur is necessary.
- You will be able to solve some problems since you are able to write programs.
- Now they are many open sources you can get online and customize them according to your preferences.

Perhaps, you are asking yourself which programs should you start learning. This might not have a definite answer since this depends on which sector you want to specialize in. Some languages to learn are like:

- HTML – When you learn to use HTML, you will be able to identify a weakness in a code. It helps you get how to maneuver to get data and interpret it.
- C and C++ - This type of language can help you to create your own shellcode, by advancing it to your desired mode.

- PHP –is one of the well-known languages widely used. It is good to understand the language. Since if not written well, sites can become vulnerable to attacks. Once you understand how to do the coding, then it is possible to detect voids and errors within the site.
- JavaScript – Once you understand how to use this language, you will be able to understand how to execute the code on the client browser. Additionally, this language will help you read any saved cookies.
- Python – Helps you understand how to customize tools already available in the market.

It is good to have the basics of some of this simple programming language. With the basics, it is possible to go about with other languages. In some instances, you will realize that some of these languages correlate with one other. So, it becomes easy to know how to go about it. If you are not adjustable to learning concepts, it might be tricky to go about hacking. Since technology is really growing fast and new inventions are being made now and then. It is good to be flexible and learn the new techs released often.

How to Protect Yourself When Hacking

From what we've learned so far, we see there many ways how data can be stolen by hackers. If attackers find loopholes or any weakness, then they get a leeway to steal sensitive information.
Here are different ways to protect yourself from being hacked:

1. Erase data saved data in your cookies. This will prevent you from cookies manipulation because of saved passwords. This applies to your personal information in dustbins or autofill.
2. Avoid public WIFI because they are not secured. Anyone can use internet connections to access data on your site.
3. Have anti-virus too, premium level to help you identify any virus that happens to be on your computer without your knowledge.
4. Keep away from free things. As seen above, free things always have a trap behind them. Nothing is free.
5. Disable third-party permissions. This often forgotten by many people, it may seem something small, but it's a key thing. When some apps are downloaded on our phones, you will find the permission icon that is always on. It may or may not notify the user of what is happening behind but check the apps. Some apps get access to phone contacts, chats and sensitive information upon installation.
6. Go ahead of the hacker by running test of your sites or computer to check any weakness that is susceptible to attack. No website or device is 100% secure; the only way to go is being smart before going to the bait.
7. Avoid going to links from untrusted people or sites. Unsecure sites are always the places where the attacks take place. Mark them as spam once you see a redirection to

certain sites or downloads of apps.

8. While using Linux, make sure you change your hostname since they will definitely know something is going on. This has to be done by using alternative internal network matches.
9. Get the tor browser and proxy. Download these two and install them in your computer and remember to change the proxy settings. The tor internet protects your data by helping you use the internet anonymously. While using this browser, no one knows who you are or what you are up-to on the internet.

What makes it more secure is that it encrypts your data when submitted on the internet in multiple layers looking like onions. Once it sends the layer, they are piled in multiple relays until the last packet leaves the relay to your desired destination. The process is called onion routing, which is one of the best ways to secure to protect yourself.

Chapter 3:
Linux Distributions and Backup File System

How to Use the Linux Operating System

This is an operating system that has been used over the years and upgraded from time to time. The older version was not user-friendly, but as it increases its popularity, the versions are upgraded to user-friendly modes. Some of the famous OS that we know are like windows, mac, and others. Programmers prefer this OS since it is easy to customize it to their desired Operating system making it one of the most downloaded OS.
You may ask yourself, should I get started with Linux? Yes, you should. The reason because it is the mother of many operating users currently used across the broad and you can never run away from it. In fact, learning Linux is one of the simplest things you can ever do as a hacker. There are different desktop environments of the Linux operating system which we shall look afterward.

What is Linux Distribution?

Linux often shortened as “Linux distro” is a type of an open-sourced program usually packed with components such as tools and software. Having understood this, you get to understand there are hundreds of distributions available for free for installation. Each type of distribution is made to run specific purposes; for example, android is one the common distribution is known by many.

Types of Linux Distributions

Ubuntu – It is one of the known Linux distributions with its existence since 2004. It was invented by Mark Shuttle Worth on South African’s Millionaire. It is available in many languages with ease to use. It is always a free operating system; you do not need any cash to have this system. Often it is released in cycles, then shipped after every 6 months. So, you can have regular updates on this system without spending any extra cash on it.
This operating system is an open-sourced program that gives the user the privilege to customize it to their best interests. It is a flexible type of system because you can use it in the desktop version or as a server. Ubuntu can be run in your computer or alternatively using it as side-side OS using it with your existing operating system.

For it runs, then it needs the help of Linux architecture to be able to send communications to the computer's hardware. You will realize it uses a Graphic user interface just like other operating systems like Windows, Mac OS, and Android. You will agree that many people have used this operating system making it easier to use the library files in Ubuntu.

Gentoo Linux –Is a source-based kind of distribution. This means you have to figure the source; therefore, it is a distribution for experienced developers. Due to the portability of this operating system, it can be used in any kind of environment without any difficulties. In regard to this feature, you are able to install it using the common processors PA-RICC, x86 and more.

Using this distribution is not a walk in the park, mastering the Linux terminal will help hackers who wish to explore more on this OS. It is more like starting Linux from scratch, although once you get to understand these incredible distros. You get lots of opportunities to build and personalize the features in a great way.

It has its own port called portage. The portages are fundamental since they help in safe installations, virtual packages, and the do configuration of files for you. Therefore, giving you a great performance score for you.

Let us get in details:

Advantages of Using Gentoo Linux

- Highly secured with Linux.
- It is a dynamic operating system that is easily administered in any type of computer.
- It has a package that is essential, especially to those who are new users.
- It has a diverse number of tools that will help you manage the OS in an easy way.
- They have Gentoo community that are available for any queries suppose you get somewhere.
- It supports different types of architect's processors to choose from .

Common User Problems

- User is unable to know how to use the Gentoo.
- User is unable to install applications.
- User is unable to use portages.
- User is unable to use terminals.
- The user is not reading the manual that is the documents that explain all about Gentoo Linux.

You may become interested in this Gentoo Linux, here is what you need to know.

- Know the terminal – This is obviously the most powerful tool you need to master.
- Know how to build applications – in every operating system you will definitely find
- Applications. Knowing how to install and use them the right way is essential.
- Every distribution comes with a document. Make it a habit of reading this will help you master the basics in that distribution or any other for that matter.
- Keep learning Gentoo skills to improve on what you do not know. This way, it helps you master the basics faster.

Linux mint – it is a widely used operating system with many users using it. It was launched in 2006 and having a computer themed desktop version. This can be a good operating system for beginners since it has good usability, easy for them to begin .
Linux mint has been built among other software layers such as the Linux Kernel, and cinnamon desktop. In relation to this, you may find that it relies on an operating system such as Ubuntu and Debian as its system base. The main goal for Linux mint is to create a usable environment for users while using their desktops to perform tasks.

The goal of the day is to create a simplified user interference that allows companies and business-people to upgrade their skills. Therefore, giving the targeted users the opportunity to expand their knowledge on the advanced technologies instead of simplifying things. It provides them with the freedom to improvise their ideas on the real desktop.

Editors

This one is a release that is done in Linux mint to help modify to suit a particular need. Some of the popular editions that you are used by hackers are:

- VIM – You may find it's getting boring using default "vi" editor as your main editor. Worry not, because VIM gives you the best performance with many options to choose from. The word VIM means "vi improved" just its pronunciation meaning an editor improvised to suit the need of a developer.
- Geany - It is among the top used editors by Linux developers with the integration of the GTK+ toolkit. It is an editor that supports many programming languages.
- Kwrite – It was first developed by KDE and later released in the year 2000. With its main aim to help edit remote files and encode your files.
- Nano – It is a known editor used mostly used with UNIX operating system. It was first released in 2000 with many additional functionalities in it a powerful text editor. Something else is that it can only run using a command-line only.
- GNUS Emacs – It is the oldest text editor in the Linux environment that you can use as your editor. The editor was developed by a man called Richard Stall Man, who is also the project founder of GNU. The good thing with this operating system is that many programmers have used it and have been proven to work well with almost all programming languages.
- Gedit – If you have happened to use GNOME you've probably used this text editor because it comes as a default editor. Its first release was
- Kate – Kubuntu operating system comes with Kate as a default editor. With this editor, you are able to work on many files to do your editing. Something fun about this editor i that it supports many languages as it auto-detects languages.
- Eclipse – This editor is mostly used by design and end developers who are looking forward to coming up with a robust of exciting coding. For experienced hackers with different

knowledge of programming languages, can use this editor to improvise on their editing.

Mandriva – The OS was first introduced as Mandrake Linux with some unfamiliar features. Today it is among the best operating system currently being used. It has user-friendly features like the Mandriva control center for upcoming hackers. This feature makes it easy to configure certain settings.

Some of the features likely to find are:

- Graphical installer – This feature helps any user to complete the operating system on their desktops or laptops without having to the process of the command line. This feature is good, especially for those still learning or just started learning Linux.
- Graphical configuration – This feature has been improvised in a way where, instead of having to go through the process of editing configuration settings. The feature helps you to change the configuration using the graphical tools. To help you understand this, take an example before editors arrived at the public. Initially, they used to edit a computer's IP address to do the configuration. This is now different from the advancement of the technology, editors like VIM will help you do that.

Puppy Linux – It is an old OS that is ideal for old computer versions. It has a variety of featured configurations that users can use to fit their needs. It is good to understand this type of distribution is a combination of different operating systems.

It has four main versions released mainly:

- Lucid puppy – When it was first created it was called Lucid, then later added the word puppy when it was released to the public by a guy called Larry Short. The word puppy reflected its aspect of being good looking and user-friendly.
- Wary puppy – It was a second release of the puppy from a guy called Barry. This kind was mostly developed for old computers with fewer requirements. Therefore, it is not a popular distribution in the technology world. Its main aim was to make the old desktop become functional and useful. It used the old version of the kernel that supports the old hardware, where it can no longer be supported by the new version of the kernel.
- Slacko puppy – This is the third release of puppy Linux that was developed and maintained by a guy called Mick Amadio. It became known due to its functionalities, and the news about Slacko gained popularity in the community with positive feedback. For this reason, it went creating new releases the latest one being Slacko 5.7 that is still used to date.
- Racy puppy - This is the fourth release by Barry Racy Puppy, the same person who created a warry puppy. It resembles the features of warry; the only difference is that this one is targeted for new versions of the desktop. It uses the new version of the kernel, making it more popular than a warry puppy.
- Precise Puppy – This was the fifth release in the list by Barry again. It was built with the compatibilities of Ubuntu with few bells and whistles that required a more advanced technology as opposed to Barry's main aim. Among Barry releases, this one gained more popularity and is still used to date.

Directories in Linux

Whatever you see in Linux has a file system in them. Each file system is composed of different directories mostly known to us as folders (its root is “/” which is the base root). The directories may have some different partitions either on the operating system or a different one but still based on the file system. They follow a certain file system hierarchy, but at times they may deviate from that order. Here is a list of the regularly used directories in Linux.

- [/] – This the primary hierarchy root in the whole file system hierarchy.
- [/bin] – Helps store the user’s commands.
- [/boot/] – Has the system startup files.
- [/dev/] – Has the device files.
- [/etc/] – Here is the location of directories and configurations files.
- [/home/] – Default location where home directories are found.
- [/initrd/] - Assist in the loading of device modules.
- [/lib/ and /usr/lib/] – This directory holds the library files used by programs.
- [/lost + found/] – Holds those files with no names also known as orphaned files.
- [/opt/] – Suppose if you have third party software, this directory helps in the installing and uninstalling of that software.
- [/proc/] – It is a virtual directory that holds system data required by specific programs.
- [/sbin/] – This directory helps in storing user commands.
- [/tmp/] – The directory keeps temporary system directories.
- [/usr/] – Contains a file system related to users.
- [/var/] – Files that keep changing such as printer are contained here.

Backup Linux File Systems

The backup of these file systems is mostly done by backup commands such as tar, dump, and spio. As a hacker, it is good to understand how the backup takes place because many times, you will be needed to do the backup. The commands listed there may not be sufficient to do the whole process of backup but let us look at the different ways you need to do while backing up.

a. Backing using tar command – (Tape achieve) the backup can be done both on a single file backup. With this command, you cannot backup special characters and only works on mounted files.
b. Backing using cpio command – This type can be used to both single and multiple files. Unlike tar, this type of command backs special characters and the block files. Additionally, for you to backup with this command, it requires you to have a list of files.

Backup can be done by a user using the dump utility that helps you do the full backup and restore process. The backup can be done in various ways like a user using the remote system, tape or restoring selected or fill files. It is easy to backup or restores what you are doing using the Linux operating system; you'll need to master some commands or programs that can help you to restore options.

Types of Desktop Environments

GNOME – Stands for GNU Network Object Model Environment. It is a graphical user interface used for desktop applications to enhance the usability of the user. You do need to know how to do programming for you to sue this application. It is a user-friendly application that helps newbies go about the Linux operating system.

The desktop version has widgets libraries that help hackers or programmers develop apps that intergrade with the GNOME environment. It is available in different forms such as Fedora and getting started with it is easy because you can perform a different task at the same time.

KDE – Stands for the K desktop environment. It is a popular Linux desktop environment, but more complex compared to GNOME. This type has made it easy for users since it uses windows. It gives the user the choice of choosing their customized desktop environment since it comes with different distributions features like the Konqueror browser and Koffice software. The browser works like windows, that help you browse the local files or similarly it is a browser itself. If you are looking for a desktop option with many configuration options, then this can best suit your requirements. It has its basis on the QT toolkit, meaning it uses different programs compared to other desktops.

Xfce – This Linux environment has similar functions and features like that of GNOME. This one can be more ideal if you want more of a traditional and lightweight environment. For this one, it uses the GTK toolkit which has installed programs such as text editor, and image viewer.

Cinnamon – The developer who created this, developed it for Linux Mint distributions. It is a modern kind of desktop environment with graphical interferences based on GNOME 3. This type of environment can also be used in Ubuntu as your preferred desktop environment.

MATE – This type is composed of an open-source that is free to use and runs on the Linux operating system. Its name is derived from a South American plant called yerba mate. Mate Advanced Traditional Environment was supported by many people of this community that it advances to become a free desktop environment for Linux. The good thing with the type of desktop environment is compatible with both the old and new versions of desktops.

Unity – It was developed by a company behind Ubuntu distributions. The main aim of this desktop environment was developed to improve the real estate screen to run in netbooks. Its first release was made in the year 2010, with consistent improvements from time to time. It is kind different from other desktop environments because it uses a different user interface. On the left side is the launcher, and the dash icon on top of it.

For a new user, it is difficult to note where one can disable or enable this notification on the configuration settings. The user will have to search on the web browser to see how to go about it. One disadvantage of this desktop environment is that it has very little default tools that can be used customization.

Chapter 4:
Text Manipulation

Text manipulation is the process of modifying the text to suit our needs. There are many formatting features that can be done to a document. They include but not limited to, changing the word of every character of a word to uppercase, maybe there is a misspelled word in a document, and we want to change every instance of the misspelled word in the whole document.
When we manually edit the text in a document, it can be very tiresome; moreover, we cannot eliminate the possibilities of human error. Thus it is very vital to find a way to find a command that can be used to modify words in a text file of any size in a document with ease. The Linux command line comes in hand, as it allows someone to type a single command line and edit the whole document in less than a document.
Each and every command found in the Linux operating system can perform only one job, except that it will only do that one job and no other. The Linux command-line tools can be linked together like bricks the way we want and use the command lines to our advantage.

How Linux is Useful in Text Manipulation

The process of connecting the command lines is known as piping. The symbol for a pipe is I (always remember it is not a lower-case L or any other character, but only I) .
Another feature that Linux offers is the use of the right-angle bracket >. By default, the Linux operating system displays commands on the screen when we input the command in the terminal. However, when we want to create files, we need to redirect the output of the command using the > then followed by the name of the file. Please note if the file is written already existed, it will be overwritten; hence, it becomes important to append the file if you don’t want it to be overwritten. In order to append a file, you need to use the >> to APPEND (added) to the file. In instances when you append a file, the results will be saved at the end of the existing file. Note only existing files can be appended.

Why Use Text Manipulation

Consider an instant where you are trying to access a web page, and you have forgotten your password or your username or even both. You may be required to attempt guessing the password or the username. However, with

passwords, they are usually case sensitive; hence, you may be required to try various versions of the password in order to come up with the exact password. This process might be very tiresome; hence, it becomes important to come up with a wordlist to give us various versions of the password or username.
Linux comes in handy as it will help you to come up with various wordlist which you will use in trying the version of the password that you have forgotten. Please note trying to access accessing people's pages or systems without their consent is unethical and should not be tried as it may land you in the hands of the law .

Creating a Word List

Take an example of the word Unix and Linux.
How would we use text manipulation in this case ?
In this case, each word would be on its own line; using this example, the word Linux would be on line one. When creating a word list using the word Unix, you may want to try passwords such as Unix, UNIX, UNIX, and Unix. If at this instant we want to perform this action every line on our word file and append the result on our original word list, the word list would be four times bigger than our original word list, hence it is crucial to consider our file size while creating files like this since their size would be bigger as one gigabyte or even more.
Explanation
For example, using, the word Unix there are different combinations Unix this is where first char is an upper case

- UNIX all characters are uppercase
- UNIX alternative options for uppercase
- Unix alternative options for lowercase
- Think of how many options there would be for the word UNIX, or even if there would be numeric after the word?

Top Seven Linux command Line tools for text manipulation

- **The vim command** - The command is a text editor that can be used in a command-line mode or as GUI mode. Vim is very strong and can be used in editing any kind of text. One demerit with vim is that it that its learning curve is very steep.

Features of vim

- ❖ Runs on very low memory.
- ❖ It supports very many files and encodings.
- ❖ It is compatible across all Unix systems

- ❖ **Grep command** - The command is one of the frequently used commands due to its high speed in most Unix based systems. With other programs, a grey command can be used with other programs to read inputs in various. In a file or a standard output, grep can be used to find strings or matched regular extensions.

- ❖ **Sort command** - As its name suggests, it used in sorting a file alphanumerically as it will arrange the lines following a specific order. The command arranges files in ASCII by default.

Ways in which sort command sorts lines

1. Sort command arrange lines one at a time
2. The sort command program prints all lines of inputs in an argument ordered list
3. The sort command supports sorting of lines in alphabetically, by the use of reverse order, numerically or even by word of mouth.
4. The command can also ignore case sensitive and return on if the line is sorted or not
5. The entire input key is assumed to be a sort key by default, and the blank space is taken by the field separator

Features of a sort command

- ❖ The lines that are starting with numbers will come before lines that start with alphabets
- ❖ Lines whose first letter appears first on the alphabet list are placed before lines whose first letter comes later in the alphabet list.
- ❖ Lines beginning with a letter that appears earlier in the alphabet will come before lines starting with a letter that appears later on the alphabet
- ❖ Lines that begin with an uppercase letter will be placed before lines starting with a lowercase letter of the same word

Example:

Suppose you are creating a file with the name file.txt
$ cat > file.txt

- Emailing
- Client
- command
- variable

sorting the file| using the sort comman d
syntax:

$ sort filename.txt
Command:
$ sort file.txt
Output

- Client
- Command
- Emailing
- Variable
- **Cut command** - This command is used while removing characters, bytes, and fields from files. By default, the command output sorted output leaving the unchanged original file. Various parameters are considered while determining the files to be cut.
- **Uniq command** - The command is used in removing duplicate text in lines of already sorted lines. Hence the duplicate lines must be on their own lines for them to be identified and sorted. This command is typically used with a unique command.
- **SED command** - The command manipulates text based on special commands that are written using the sed language in Unix systems. The sed language is used in creating sed scripts that use the hashing or file extension. Mostly, sed is used while replacing text.
- **Less command** - This command is used in the buffering the output on the screen.
- **Diff command** - This is a simple to use command. It is used in comparing text sources, then output the differences of the tex t

Managing networks

Network management is the process of using tools to administrate, run and manage a computer network. Network management is mostly about the capability and efficiency of the data transfer channels. However, there is no precise definition of network management, as it is an extensive topic. The main areas in network management are

- **Network administration**

It involves the tracking of many computer resources, such as monitoring the transition lines, hubs, switches, routers, and servers. You can monitor the performance and updating the associated software-mostly network management software, network operating system and software applications that are distributed and used by network users.
Like any other OS, you need to be connected to a network. It can be your internet service provider network, your company's network or even WIFI. You will then be required to learn how to connect Linux OS to a network.

The Linux network interface

There are different versions of Linux, and they name their network differently. Generally, all Linux OS will have the following two

interfaces.

Loopback. The loopback (Lo) interface will always have an IP address of 127.0.0.0 that represents the host. For example, you want to access a web page running on the same server as that you are, and then you will need to type, HTTP:127.0.0.1 on your browser .

Ethernet. The Ethernet 0 (eth0) interface is basically the local network connection. When running Linux on a virtual machine, you will still be required to connect to a physical network interface of the host. Commonly, you will be required to cross-check if the Ethernet has an IP address and the eth0 is in an UP-state.

MAC address

The media access control (MAC) is an identifier assigned to a network at layer 2 of the data link layer. A network interface must have a MAC address which is commonly called the hardware address.MAC addresses are always to any adopter when they are manufactured or if it's a virtual adapter, the address is assigned to the adapter when it is created and always appears in six groups of two hexadecimal digits each.The Ethernet interface explained above the MAC address is known as ether address or the link.

IP addressing

The IP addresses of all devices are unique on a particular network; hence, every device has at least one address. IP addresses normally fall between 1.1.1.1 and 255.255.255.

Example 1

To view the IP address used, you will type the IP address commands which will look like the one below

<LOOPBACK,UP,LOWER_UP> 6536 qdisc noqueue state UKNOWN group default

Lin/loopback 00.00.00.00.00.00 brd 00.00.00.00.00.0 0

inet 127.0.0.1/8 scope host Io

valid_Ift forever preferred_If forever

inte 6::1/128 scope host

valid_Ift forever preffered_Ift forever

There is one Linux networking tool that everyone conversant with Linux has heard of, i.e. ping. It began as an acronym and now has gotten to enjoy its status as a fully flagged word and now it's the most commonly used tool in testing the reachability of a network

How Ping Works

Ping starts by sending an internet control message protocol (ICMP) across the network; then, it will notify when there is a response or not. If the host is able to connect to the network, it will send an ICMP success response. However, if the user is unable to connect to the network, it will send a ping test failed

Another tool used by Linux in the troubleshooting process is the **trace roo** t. Traceroot usually Probes the network between the local system and the destination; hence, it will gather all information; hence, it will gather all information about the IP router in the path. Tracetoot is useful when there is an issue in the path, such as network issues intermediary nodes. showing slow response then you may want to know which node is showing the slow response .

Dynamic host configuration protocol (DHCP)

What if you have tens, thousands or an infinite number of computers within your network? It will be time-consuming to try to assign and track which machines have which IP address. Here is when the power of DHCP comes to play. It will obtain the IP address of a host or a device when it actually connects to the specific network. However, DHCP is commonly used with devices that do not have side effects occurring from a dynamic IP address change in the server system. Administrators can create DHCP reservations or manually configure static IP addresses to ensure every network adapter connected to the network will always get the same IP address when connected to the network.

DNS (Dynamic name system)

Interconnected computers mostly connect to each other using IP addresses. It would be very difficult to keep remembering everything you need to connect to on the internet. imagine you want to connect to google.com even facebook.com and then you are required to remember the IP of google, facebook or any other website every time you want to search them on the web. It would be very hectic to access the internet. DNS comes in handy as it is used to map the web pages to specific IP addresses that we can easily remember .

Network statistics and counters

While undertaking the troubleshooting process, it becomes vital to answer questions. Firstly, is the interface sending the data it is required to transmit if it is not sending then is it sending some errors and lastly what process is sending all that traffic.

Having covered how to view network information, then we need to know how to set up some common network configurations. The process is as shown below.

Process

Start by changing an IP address. Note that when making changes in Linux you can have these two types of changes

- Immediate effective changes that are not persistent. These changes are not effective once you restart off your operating system
- Changes that are effective even after restarting your computer

For immediate effective changes to be made on the IP configuration, then you will use the link, route, and address command options. However, you will be required to have the iproute2 package installed on the Linux OS. By default, the package is usually pre-installed but will highly depend on the version of the OS you are using .

Network interface bonding

You may need more network bandwidth than a single interface can give. Otherwise, you may be having network issues, or maybe you want some form of risk redundancy. The risk redundancy has many names depending on the vendor; such names are EtherChannell, VMware, PortGroups, bonds, and risk aggression groups are just a few. Linux also provides such features and is named bonding. Bonding allows a user to create a single logical network and later link with multiple logical networks, then scale to add more interface hence the provision of load balancing, thus preventing interface failure protection. Using the bonding feature requires one to install the kernel module via the modprobe command since the modrobe command will help in adding additional capabilities in the Linux kernel.

Basic Linux network bonding modules

- **Balancer rr** - It’s the default round-robin bonding that provides fault tolerance and also loads balancing.
- **Active back up** - This module provides fault tolerance where one slave can be active at a time and in an event it fails the other slave takes over.

- **Balance xor** - Based on hash, it offers fault tolerance and load balancing.
- **Broadcast** - Offers fault tolerance through transmitting data on the slave interface.
- **802.3ad** - It creates aggression groups for any link sharing the same velocity as a duplex for it to provide load balancing and fault tolerance. Its standard dynamic link is IEE.802.3ad. 802.ad is the common type of bonding, and it uses LACP for it to communicate with a bond from the other side
- **Balance tbl** - It adaptively transmits load balancing without the help of any special switch support.
- **Balance alb** - It's an adaptive load balancing that requires no special switch support due to its use of ARP.d

Network operation

This includes the smooth network operation according to plan, including the overseeing of activities so as to easily solve any problem even before the user of the system can notice.

Network maintenance

It involves repair on time and upgrading of computer resources and putting in place preventive measures through close monitoring of computer networks by administrators. Simply network maintenance is what is required in order to keep a network running. Examples include upgrading computer resources including the routers, transmission lines, and even the switches .

Network provision

The process involves the configuration of network resources to suit the needs of certain services. These services may include expanding broadband to increase more users into the network and improving voice capabilities.

Process of hacking with kali Linux

Linux operating system comes with over 300 pre-installed tools which you will probably need in your hacking process, while the other Linux tools can be downloaded easily from the internet. With Linux, the hacking process becomes easy when you know how to use the tools and how to manipulate the tools to your specifications.

What you need to know before you start the hacking procedure

Hacking is an art and hence needs years of practice in order to master it in full and also how to handle various attacks on systems. Having no idea about hacking is okay while you start practicing the hacking process. Be warned a newbie with general knowledge about computers and no

knowledge about programming can be very risky. Hence In order to be an expert in hacking, it will require you to make a lot of effort in practice.

What you require in order to start with Linu x

For beginners without any knowledge, you need first to visit the kali Linux official page then read about the basics of the Linux OS. Then visit their download page and download the Kali Linux iOS file. Before beginning, you will need to know how to install the latest Kali Linux OS. The procedure below will show how to install it.

Installation of Kali Linux

There are two ways in which you can install the kali such as, installation on the hard disk using USB, and installation using VMware. Our focus will be On the later:

Installation of Kali Linux in VMware

Step one

First and foremost, you should identify your computer capability, whether it is the 64 bit or the 32 bit. If your computer is the 64-bit, then you will need the 64-bit version. If you are not sure of the version of your computer or it is the 32-bit version, then you will need the 32-bit version

Step 2

You then download Linux via the browsers download manager by clicking the ISO or by clicking the torrent. Torrent is highly recommended as it is faster. When the download is complete, create a virtual machine.

Step 3

Select an installer disk image file (ISO) on the window that appears, then select the Kali Linux ISO file on the browse section. Once you select, then click Next.

Step 4

On the toolbox that appears, select a title for your virtual machine. You can have a name such as Virtual kali. You should also select a location for it. It is recommended you name your folder Virtual machines in my documents then click Next.

Step 5

The next step is to set the maximum size of your kali, which should not be less than 20 GB. However, you are recommended to use 30 GB as kali tends to expand over time. After selecting the size of your kali, on the next option change to store virtual disk as a single file, then click Next.

Step 6

Customize some hardware settings on the window that appears on the Customize Hardware Option Button.

Step 7

Now, a window for hardware settings will be presented for you. Then select Memory in the left window pane, then on the right window and slide the slider to at least 512 MB. Be aware that you should assign a virtual machine a maximum of half the Ram installed. For example, if your computer has a ram of 8 GB, then a maximum of 4 GB is what should be memory given to the slider. In the left plane, highlight processors on the left plane. The option will heavily depend on your computer's processor. If your computer has multiple processors, then you should select multiple processors for efficient performance. On the left, click the New Adaptor, then on the right side move the dot to the bridged (top) option and click configure the Adapters button. A small window will automatically pop up where you will uncheck all the options in the exception of the one next to the network adaptor then click OK. At the bottom of the hardware, close then Click Finish in the wizard.

Step 8

The new virtual machine will have been added to the VM library once you clicked the finish button. Start Linux in order to install it. Highlight, the name of the created virtual machine in the right pane in order to finish the installation.

Step 9

On the boot menu that appears, use the down arrow keys for scrolling down to Graphical install and hit enter.

Step 10

Select your preferred language; by using either the mouse or the arrow keys to scroll then hit Enter.

Step 11

Select your location on the next screen that appears and hit continue.

Step 12

You will be asked your keymap, for example, you can choose the Arabic keyboard. *Note this will depend on the language that you understand.

Step 14

Kali will delete the hardware on your computer, then click continue but do not configure the network yet .

Step 15

You will set the computer hostname which will act as your computer name. Click continue

Step 16

On the next window set your preferred root password on the main account. Then confirm the inputted password and click continue.

Step 17

Linux will now determine the disk partitions then you will be presented with a dialogue box, select the first option and continue and make sure that all files to be on one file and hit continue. Then confirm the desired changes to be made by selecting Finish partitioning and make changes to the disk and confirm the changes. The OS will now install and can take a duration of up to 30 minutes.

Step 18

Then select no when asked about the network mirror

Step 19

The installer will ask whether to install the GRUG and accept it. The installation process is completed by now, and a success notification will be displayed

The OS will restart, and then you can log in using the password you heard the set.

Storage management

Storage management refers to the process of managing computer storage devices. Linux has many tools that can be used to manage storage drives and hardware devices. Linux can perform the following in regards to storage management:

Memory management

The kernel allows processes to access the memory when they need them since it has access to the system memory.

Device management

The kernels allow processes to access peripheral devices through device drivers.

Task management

The kernel allows application execution. It performs context switching between software and hardware

Methods of storage management

- Provision of options to failback of the LUN to the required paths
- Monitoring and notifying if there is any change in the path
- Provision of options to the customization of names of the device-mapper to multipath devices.
- Provision of policy-based paths and grouping for a user to the customization of I/O through defined paths
- Facilitation of balancing of the loads among many paths.

Block storage

Block storage in Linux kernel is simply a synonym for the word block device. A block device is any hardware device that stores data same as flash memory, solid-state drive(SSD) and hard disk drive (HDD). Basically, a block device is commonly known as disk storage in the computer. Once it has been set the block device acts as an addition of the current file system on the device hence, you can read or even write information on the drive.

How Linux manages storage devices

Linux operating system represents almost everything by a file that encompasses the storage devices present on the /dev category. Files in charge of the storage devices always named starting with sd or HD then followed by a letter. For instance, the first drive in a server can be represented by /dev/sda. Similarly, partitioned files within these drives have files within the /dev category. The partitions are usually appended by adding a number after the drive name. For instance, the second partition from the previous example would be /dev/sda2.

Since the /dev/sda* and /dev/hd* device files represent the olden ways of partitioning drives, it then becomes a disadvantage when the values are used in isolation. When the device changes its nodes during the booting process, it poses confusion as to which drive has been assigned a particular

name, since the Linux kernel decides which drive gets a certain name during each and every booting process .
Solving the issue with the /dev/disk, subdirectories correspond with a defined way of identifying the partitions and disks on the system. The subdirectories include symbolic links generated during booting to correct/dev/(sh)da* files. These links are usually named in accordance with the directory identifying the files.

Subdirectories under /dev/disk

- **By label:** File systems must have a method of naming files in order to assign a name to a particular disk or partition.
- **By uuid:** The UUID or the universally unique identifiers are a long string of alphanumerical that can be used as an identification of a storage device. The strings are not readable by people but guaranteed to be specific across systems. It will be much advisable to use uuids to refer to storage that may move across various systems since naming collisions cannot occur.
- **By partlabel and by partuuid** : GPT tables usually have specific labels and even the UUIDs used in identification. This directory function similar to the previous two directories but uses GPT specific identifiers.
- **By id:** They are dynamic hardware generated directories having their serial numbers and the hardware attached to them.
- **By path:** the directory entirely depends on the storage devices that are connected to the system. However, connection to a different port alters the original value of the directory; hence, it faces similar kinds of drawbacks to by ids.

Mounting block devices

Files contained within the /dev are used in communication with the kernel driver for the system. More abstraction is required to make the device as a portion of the remaining space. In Linux and similar Unix OS despite the number of devices involved the system is represented in a single unified file tree. While accessing the driver's contents you will access them from there. There are various mounting options used that alter the mounted devise behavior. To prevent the contents from being altered, it is wise to mount the drive in the read-only mode.
Using the file system hierarchy standard, you can mount file systems under the /mnt or its subdirectory. However, the hierarchy offers no recommendation on where to place permanent storage; hence, you can mount them under the /mnt or /mnt subdirectories.

Making permanent mounts with /etc/fstab

The Linux system checks on the file system table(**/etc/fstab**) to gauge the file systems that it will and those that it will not mount during the boot process. With the exception of file systems having the system. Mont unit files, the other file system having no entry won't be

mounted automatically. This file system is simple as each line must represent a file system that should be mounted.

More complex storage management
Complex management paradigms are used when we want to achieve flexibility, more performance, additional management structures or even redundancy .

What is raid
It means a redundant array of an independent disk. It allows someone to categorize drives and later manage them as a single unit with more capabilities as it uses virtualization technology and also it is storage management. The features of raid arrays always depend on the raid level. The raid level, on the other hand, defines the relationship of the disk in the array. The level will also impact the performance and the redundancy of the set.
Most common levels

- **RAID 0 -** This level indicates the drive splitting phase. By splitting it means that the data is written on the array, then split and later distributed among the various disk in that particular set. It has merit since it can be read or written. However, it faces drawbacks as failure to a single drive can lead to loss of data from the entire array. This is due to the fact that no single disk contains enough information about the content needed to rebuild.
- **RAID 1 -** Can be used in making a copy of some data. For example, you have 2 TB hard drives; in total, you would have 4TB hard drives. In mirroring you will be able to see only the 2 TB or the logical hard drive. However, while saving data it will write on both drives. The minimum number of dives must be more than two since in an event one of the drives fails, you will get the data from the other disk; hence ensuring there is no data loss.

Characteristics of RAID 1

I. It provides fault tolerance capability.
II. Rebuilt is usually fast.
III. Its performance is usually fast.
IV. It can be used in a small scale database and operating system.
V. The reading of the contents is usually good.
VI. The writing performance of the raid is usually slow.
VII. Half of the total space capacity will be lost.

- **RAID 5 -** This level works by partial distribution, where part of the information will be used in rebuilding the data. This ensures our data is protected in the eventuality of a drive failure. The partial distributed information is stored in each of the drives. This RAID can survive from a single drive failure, but if more than one drive fails, it can cause a lot of data loss.

Characteristics of RAID 5

I. It has excellent performance.
II. Reading is usually fast due to the high speed involved.
III. It provides fault tolerance of drives.
IV. One disk must be under parity.

V. It can be used in strong servers such as web browsers, file servers, and important backups.
VI. Can rebuild parity information from all the drives.

❖ **RAID 10 -** Raid 10 has similar characteristics as raid 5 but does not parity distributes system. It needs a minimum of four drives so as to ensure even if two of the drives fail data can be rebuilding while replacing the fault drives .

Characteristics of raid 5

a) The reading performance is good.
b) Poor performance.
c) It provides fault tolerance.
d) Rebuilding is done from two parity drives.
e) It may be used in long arrays.
f) It is used on a large scale, backups and video streaming.

Basic disk knowledge is important in order to know how to the raid setup.

File system

While performing storage management, the file system is very important. Deciding which file system to be suited for the computer relies on the type of computing you want to perform.
The root directory of a file system contains directories and files, At the top of the directory denoted by /.

Types of pathnames

A. Absolute pathnames

Files are always identified by their absolute pathnames, which starts from the root in an inverted manner. The (/) has 2 different meanings in absolute parenthesis.ie.

- Specification of the file directory.
- Abbreviation of the home directory using tilde.
- Abbreviation of the user's name using tilde.

B. Relative path names

Relative parenthesis starts from the current location, unlike the absolute parenthesis.

Listing Files

The Is command can be used in listing the contents of the directory. More so, you can list files in a deferent
directory by typing Is the pathname.

Changing the working directory

To change you need to enter your typing the command cd followed by the directory you need to change. You could also go back to the home page by typing cd followed by the directory when you get lost in a file system.

The root user

Linux system has a special user named root, which is a powerful administrator of the Linux system. The root can access all the files, including the other users. The root can perform all the formatting features to the files. The dream of upcoming hackers is to become the root user of a system.

Chapter 5:
Detailed Overview of Linux and How to Hack with Linux

How does Linux work?

The Linux operating system is flexible compared to the windows operating system. However, Linux can be complicated to use than the Windows operating system. Linux carries multiple advantages when compared to other operating systems. Linux has two well-known features that make it stand out: Linux kernel and server platform. In some cases, the Linux system can be said to have five primary components.

Linux kernel: It is a feature that allows one to modify the operating systems. Linux operating system can be changed or customized to fit the specific needs of the user. For example, most android phones and some of the appliances use Linux operating system. This feature has allowed the developers of the tools to customize the operating system to fit their needs.

Server platform: this means that one can use the operating system as the basis for building another operating system. Therefore, it offers a better option for most developers and manufacturers in device manufacturing.

Shells and GUIs: Through shells and GUIs the users can communicate to the kernel. The user uses the shell to execute the functions they want. Through the shells, the user can input commands into the system, and the kernel will run the commands.

System utilities: The system utilities include any applications and programs that the individual will use to carry out the tasks that they want. Without the system utilities, managing the device will be close to impossible and therefore, they make an essential part of the Linux operating system.

Application programs: These are various applications that can be used for various tasks by the user. These are applications that the user will install or upload into the system that they will use for multiple tasks. For example, a browser, an antivirus program if need be. They are mostly introduced from external sources and are not pre-installed like system applications.

Linux works as a multi-front operating system and can serve different purposes according to the customization. However, just like in other operating systems, the central role is to connect the user to the device. Linux allows the user to access any services that they may need from the device. Linux is not a single program or group of tools. When one downloads the Linux operating system, it comes with other programs called the GNU programs. There are different distributions of Linux that are in use today. However, each is customized to suit a user’s needs. These

distributions remain open source meaning one can modify or customize them if they feel that they are hard to use or operate.

Once one has downloaded the best distribution for themselves, they can then download free applications to use with Linux. Most of the claims are free to download, just like the operating system. Once one has the apps on their device, they can then use them like any other operating system. The applications can be used to browse the internet, play games, office suite, among other activities. In case you do not like something about the operating system, one can customize it so that it is similar to their need. Therefore, in Linux, one will customize the operating system by switching various features. The Windows operating system does not allow for one to change some functions. Consequently, one can only add third-party applications to an already existing operating system. Linux will allow one to use what they wish to and eliminate or replace what they do not need in their operating system. This is making it straightforward to operate and flexible for many organizations that need to make their own custom operating systems.

One can say that the Linux operating system provides the user with only the core components of an operating system. From the core components, the user will then move to make their operating system. This will be done through the user adding or removing some of the elements that they do not like. Through this elimination and replacement method, the user will eventually be able to customize or build an operating system of their liking. This is one of the features that the other operating systems lack. Other operating systems come fully developed, and very little can be changed or customized about them. This can be taken or explained best using a building. The windows operating system usually presents the user with a whole building, and for this reason, very few things can be changed. However, Linux provides the user with the foundation of a building. The user can then build the building they want from that foundation. Therefore, on installing the Linux operating system, the user will add applications that they prefer, customize the desktop to their preferred design.

Shell- one of the most important features of Linux, is the shell. It allows the user to enter commands through the command line into the system. In windows, the shell is the command prompt. However, the command line has a significant advantage over the command prompt. While one can change the functioning of a windows system and some of its features, a user

can use the command line to improve the functioning of Linux. From the command line, one can modify how the system will work. The shell is one of the essential tools for a hacker that allows them to develop scripts and programs that can be used for hacking.

The boot loader is a component of Linux that the user first interacts with. The boot loader is automatically installed when one installs Linux. Most users do not see it, but it is always working. When a system is powered on, the operating system uses the boot loader to open access to the user. In some cases, the users may have multiple operating systems installed in their machines. When the users have various operating systems, the boot loader allows the user to choose which operating system to use. The boot loader will then open the operating system and grant access to the user. The boot loader is automatic and runs from the minute the system is opened to the minute the user can access the desktop and applications. It prepares the system for use.

When one wants to use a Linux system, the first step is to power the system on. Powering the system means that the screen will light up and the boot loader will kick on. the boot loader loads the systems in the device. If there are multiple operating systems, then the user is allowed to select one to use. After selection, the boot loader will allow you to enter a password if any is required. The successful opening of the device and loading of the operating system will give the user access to the desktop of the device. Accessing the desktop may be a simple task, but the desktop of a Linux operating system changes from one device to the other. The Linux operating system has very many desktop environments that change how the desktop looks. This is unlike in windows operating systems where the desktop environment is static, and anyone can use it. The Linux system allows multiple desktop environments, and this means how one can access programs and applications can change.

Linux and windows are very different from each other. Therefore, establishing the differences between the two operating systems will help one understand how Linux works. Windows is very common to most users and can serve as a good point of comparison. Various features on windows are replaced on the Linux system by other elements of Linux variation.

One difference can be seen in that Linux has very many customizable distributions. This is unlike in windows where there are no variants. For example, Linux has options such as kali, ubuntu, that are customized to fit

user needs. Windows only has versions such as windows 10, windows XP among other variants. Another difference is seen in the command line. The Linux distributions all have a command line that allows the user to customize the operating system. However, the command prompt in windows will enable one to carry out limited tasks but never to customize the operating system. While running and installing Linux takes a shorter time, Linux can run very complicated tasks. Windows, on the other hand, will take a prolonged installation period due to the many items that have to be installed at once.

Linux only requires one to install a few items and add the rest as they need them. This makes the installation complicated due to the customization and adding kernels. In Linux, the installation of updates is only when on needs or chooses to. This makes sure that the updates are stable and easy to install as one can update bit by bit. However, windows updates need to run altogether, taking longer. When they fail to install altogether, the operation of the operating system is profoundly affected. Linux operating systems are hard to break through, making it more secure. This is different from windows operating systems where a person has to install an antivirus program to avoid viruses and malware. Malware and viruses in windows can come from very many different places, and therefore one has to be cautious.

Linux for hacking

When one talks about using Linux to hack, they are speaking in a comprehensive manner. This is because there are multiple distributions of Linux with each having numerous hacking tools. This means that there are very many options that one can use to hack using Linux. However, the process of hacking using Linux is almost the same for every distribution and every tool. Therefore one can create an outline of the process of hacking using Linux.

The first step into learning how to hack using Linux is to understand the Linux operating system. In Linux, hackers have a significant advantage that presents itself in the form of the command-line interface. The command-line interface is a similar tool to command prompt in windows that allows the user to write scripts. However, the main advantage that the command-line interface has is that it can be customized to fit the needs of the user. In other words, the user has more control over their operating system in Linux

than in windows. Therefore, being aware of how to use the whole system is a significant step to learning how to use it for hacking.
The next step is that one needs to understand more about networking. If one is going to be breaking the rules, then they need to understand the rules and how they work. Therefore, when one is learning how to hack, they need to know how the systems work. Networking involves understanding how computers exchange data and how they work. Hacking consists of understanding the working of networks and their processes so that one can exploit these processes and gain access to the chain and exchange of information.
Linux is one of the closest friends to a hacker. This is because, through Linux, a hacker can be able to develop a platform through which they can hack into systems and applications. Therefore, to use Linux to hack, one needs to have some necessary skills in using Linux. One needs to develop skills that will help them navigate through the Linux system and build scripts that they can use to hack into systems. Further, Linux has very many tools that are used in hacking. It is essential that one has Linux skills so that they can be able to use these tools appropriately. The hacker must also understand various security concepts that are used in protecting programs. Linux is used to develop most securities for these programs. Therefore, through understanding the Linux system, the hacker will be able to know how they can avoid these security traps used by administrators in protecting programs, applications, and even websites.
Linux is among the best-operating systems that one can use to learn to script. This is because it offers a user very many languages to use to write scripts. The user can choose any of the languages that they will think is simple to understand. Linux allows the user to interchange languages which is not the case for windows. Windows is not very accommodating for different code languages. Therefore, Linux is a better option to learn the programming language with. The user can use Linux distributions to learn scripting skills.

How to hack and what you need

Hacking is the use of computer skills and software to overcome a technical problem. Most people will have a different reaction when the word hacking is mentioned. Some people will feel that hacking is a bad thing while it is

not. There are various types of hacking, and a kind of hacking is ethical hacking. Hacking can be used to resolve any issues that the modern worker. In contrast, it has also been used in very many contrary acts that resulted in hacking developing a negative reputation. It is, however, essential to learning how to hack various systems for the sake of acquiring multiple information. A person needs numerous tools and skills to complete a hack .

Requirements

1. ***Linux operating system*** - this is the platform through which one will be able to run the hack. Linux allows the user to write and implement various programs and therefore makes it easy for one to use during hacking. The operating system can be in multiple distributions such as Kali, Ubuntu, Backtrack, among other distributions.
2. ***HTML writing knowledge*** - HyperText Mark-Up Language is the primary language that the computer uses and can be used to give instructions to the network. Without programming knowledge, it is essential that one has some expertise in HTML. The ability to write HTML will help in writing codes.
3. ***Basic programming knowledge*** - most websites and applications are based on a program. Programming is the knowledge of how these websites and apps are created and how they run. Hacking is breaking the rules to these platforms, hacking into these platforms, then one must know the rules. Programmers use various languages when programming, including Python, Javascript, and Java. Each of the languages used by programmers offers them a set of advantages and disadvantages.

Once one has the requirements, they will have to follow various steps into hacking. When the levels are correctly monitored, then it will be a simple process. The steps can be known as phases, and there are five phases. The stages include reconnaissance, scanning, gaining access, maintaining access, and clearing tracks. Each of these steps is important and plays an essential role in the hacking process.

1) **Reconnaissance** - In this stage, one needs to collect as much information about the platform they are hacking. The information should

be collected in three parts, including network, host, and people involved. The data collection can be done into methods called footprinting. It can be active footprinting which consists of collecting information through interacting with the target. Passive footprinting consists of gathering information on the goal without interacting with the goal.

2) **Scanning** - This involves data collection of relevant information that one can use to gain access. In the scanning process, one will also evaluate which of the weaknesses found will be used to ensure that the hacker has been able to gain access to the system. Scanning should be conducted according to the various types of scans. One should first conduct a scan of the network and ports to understand the program or the system better. Use the ideas gained to know where there can be possible vulnerabilities. It is essential that scanning is conducted systematically that will ensure that the hacker does not forget to scan for anything. In the current phase, there are three types of scanning, including port scanning, vulnerability scanning, and network mapping.

   - ❖ Port scanning- in this scanning, the hacker will look for services that run on the target, open ports, and live systems.
   - ❖ Vulnerability scanning- Look for weaknesses in the target that one can use to gain access to the system. Once the defects have been identified, they can be exploited later.
   - ❖ Network mapping- Involves mapping out and getting an idea of how the target is arranged. For example, what network is being used, routers, and firewalls that protect the system. This information is essential in the hacking process and organizing the hack.

3) **Gaining Access** - the hacker will attempt to break into the system. This can be through the use of various tools. The hacker will also need to increase their privilege so that they can accomplish their task.

4) **Maintaining Access** - once the hacker has gained access, they can be kicked out or locked out by the security system of the target. The hacker has to maintain access through uploading trojan, rootkits, or malicious files. This may reduce the privilege for the administrator or even help the hacker to keep the connection without knowledge of the administrator.

5) **Clearing Tracks -** a hacker can be seen as a thief or an intruder. After accessing the system, the administrator will want to catch the intruder. The hacker, therefore, needs to cover up their tracks to avoid being caught. This step, therefore, will involve the hacker modifying the logs to the system to remove any evidence of them accessing the network. If the hacker had uploaded any files or installed any applications, they need to uninstall them to ensure that the administrator cannot detect their presence.

The above phases are a basic outline of the hacking process. All hacks follow a similar process. Various modifications can be made by the hacker to fit the needs of the hacker. For example, if one wants to prove or show the weakness of a system, they do not have to clear tracks. Failure to cover their tracks will help the administrator identify that there was an intruder in the order. Once the hacker has infiltrated the system, they can do whatever they desire. It is significant to note that in many cases that hackers are seen as a threat to the system despite having good intentions.

Python scripting for the hacker

Introduction to Python

Python is an interpretive language accompanied by elegant syntax, which makes an outstanding choice for scripting and rapid application development in many areas. It is a dynamically typed programming language. Python is also better for data manipulation and repeated tasks. Its ability to manipulate data makes it very good for hacking. It has very easy language and syntax that a hacker can use to make the best scripts. It has a large database and very many libraries that allow the user to have access to much information that they may need to hack any system.

Further, the fact that the language has very little syntax makes it simple to use. Experts also prefer to use because one does not have to struggle with the syntax. This will mean that a beginner will have an easy time creating scripts using python. Its high performance is also another reason as to why it has a very large number of users.

Applications of Python

1) GUI based desktop applications

- ❖ Image processing
- ❖ Games
- ❖ graphic design applications
- ❖ Scientific and computational applications

2) Operating systems
3) Web frameworks and web applications
4) Prototyping
5) Enterprise and business applications
6) Language development

A python script is the text file with statements that are used to develop a python program. Once a person has a python script, they can execute the program without having to rewrite it repeatedly. Therefore, a python script is an essential part of developing applications as it is more effective. Python is preferred to other scripting languages due to its high number of simple libraries. Python can be used both in complex and straightforward scripting can be used to develop a program of almost any use. NASA has been known to use python to write scripts for its application; this shows that python can be used to accomplish various tasks.

There are several reasons why most hackers prefer to use python instead of other languages.

a) Simple language- python is not only easy to learn; it is also incredibly easy to use. This advantage makes it more preferred in other languages when it comes to hacking. Beginners, as well as experts to perform a considerable variant of tasks, can use Python.

b) High performance- python has been proven to work effectively for different purposes. Python has been used and is useful in both programs involving both long and short scripts. These high-performance standards have resulted in it being a preferred language for hackers.

c) Extensive community and support- Python has a large population of users. The users can help in resolving any problems that one may encounter when they are using it. The large community ensures that issues are resolved as fast as possible. For this reason, most hackers will prefer to use python as their programming language.

d) Large market- Python presents the user with a large pool of opportunities. Programs that are run using python are ion create demand. The high demand results in hackers having access to many career opportunities.

e) Presence of Third Party Modules: The Python Package Index (PyPI) has many third-party modules that help Python to interact with most of the other languages and platforms. This means that python can be used on very many platforms, and it also ensures that many users can use its script.

When writing a python script, there are various ways to do it. Python scripts have to be written first than be run so that they can become programs. The first method of writing a python script is through typing the script in a text editor. After writing the script, the script will later be run on Terminal. The second method is through typing the script directly into Python's Interactive Interpreter. Using the interpreter allows the user to type the commands directly into the Terminal. Python is an essential and well-established programming language and loaded with secure usage of the code lines, maintenance is easy to handle, and debugging the system is very easy. It has gained importance across the globe as computer giant Google has made it one of its official programming languages.

Scripting Using Python

Scripting using Python is very easy, given the simple syntaxes that it has. This allows the user to write scripts that they understand and scripts that they can be able to execute. When writing a script using python, one can be able to verify what they are scripting as the language is very simple. It is, however, essential that the user can understand and be focused on avoiding any mistakes that they are likely to make.

There are steps that one needs to follow to write a script using python.

1. Local environment set up- Before one can use python for scripting, then they must have python in their computers. In some operating systems, Python is pre-installed. In some operating systems, one has to acquire python from a different source. For the user to ascertain whether they have python, one needs to open a terminal window and type 'python' to establish whether it is

installed or not. There are various varieties of python that is installed in the various operating system; this step allows one to identify the type installed as well.

2.
Getting python - In case of python is not installed on the device, one will need to download it. The user will need to visit the official python website, where they will get the latest source codes, news, and documentation. The website will not only help the user to get python; the user will get access to various documentation to guide on using python.

3.
Installing python- there are a few steps to follow to install and begin using python. When using a Linux operating system, one will visit the official python website and download python source code in the form of a zip file. Once downloaded, extract the files and edit the setup file in case one needs to make modifications. The next step is running the ./configure script. Click make. Then click make install.

4.
Setting PATH

The path is the directories that the operating system will search for executables. Executables are an essential part of the codes, and therefore the user gets a chance to define how they will be obtained. In every operating system, adding directories is different. In Linux, one has to follow these steps to add python directories to the path.

- **In the csh shell** − type setenv PATH "$PATH:/usr/local/bin/python" and press Enter.
- **In the bash shell (Linux)** − type export PATH="$PATH:/usr/local/bin/python" and press Enter.
- **In the sh or ksh shell**- type PATH="$PATH:/usr/local/bin/python" and press Enter.
- **Note** − /user/local/bin/python is the path of the Python directory

Python scripting using Kali Linux

Once a user has opened their computer and identifies that it runs on Kali Linux you will need to confirm what version of python you are using so as to begin scripting. These are the steps to follow before one can begin scripting using Linux.

1. Verify the python version to achieve this, and the user will need to click on the terminal on the desktop. The next step is to type the word python on the terminal, and the computer will respond with the name of the python language installed. The process will look as shown below.

Username: *-$ python*
<Python version> <date> <time>

2. Proceed to test if python is working properly. It can be through the application of a simple test such as running a script that you already know about python. One can also use arithmetic tests as follows. Simply type your text on the screen

>>> 20+40 then press enter button
60
The answer to the sum will be displayed on the terminal.
One can proceed to perform other tests on python to ascertain its functionality.

3. Close the test so as to begin scripting. Closing the script involves running one command only as follows.

>>> exit( )

4. When one wants to use prebuilt packages, one can visit the python index official website and select what package they will be using in their hack.

5.

Once the installation of the packages is complete, one can then proceed to run and type their scripts. It is essential to understand the syntax that should be used in python scripting. It is, however, quite a simple syntax.

On the terminal typing on python will look as follows.
In [1]: print "call user."
Call use r
The specific words that one uses to help insert command and run them on python are called syntax. For example, the use of the word 'print' to ask the program to display certain words. Therefore it is essential that one when scripting is able to understand the syntax as they are essential to ensure that the script is executable later.
Print () - A function to display the output of a program.
Range () - returns a list of integers, the sequence of which is defined by the arguments passed to it.
Sets - Sets are collections of unique but unordered items.
Str ()- allows the user to represent the content of a variable as a string.

Managing Linux kernel modules

The Linux kernel is Linux operating system`s core part. It is in charge of operating the computer and the hardware. There are two types of kernels which include the microkernel and monolithic kernel. The microkernel is in charge of the functionality of the operating system. The microkernel will manage the memory and CPU time. The monolithic kernel is in charge of the device drivers. When the two types are combined and work together, they help manage the operations of the operating system as well as the hardware. Modules in the Linux operating system are pieces of code that can be loaded or unloaded upon demand by the operating system. The function of these modules is to help extend the functionality of the OS without having to shut down the whole system.
The ability to load and remove modules from Linux is one of the most significant advantages of this operating system. This is because it helps to reduce the size of the operating system. Without the ability to load modules or remove them, then it would mean that the operating system would come with systems to help complete any anticipated functions loaded in the base kernel. Loading the modules will lead to wasting of memory and with users having different needs, then they would be useless to some users. Linux has

three major kernel modules: file system drivers, device drivers, and system calls.
One major reason why people prefer to compile their kernels is that some drivers (and other kernel features) need to be patched. At times some of the kernel features may get broken. When this is the case, patching up may need the user to develop their kernels. Another case where a person may develop a kernel is when they need to update the drivers or needs to use new ones. They may result in developing a new kernel. Compiling one's kernels will mean that the person will remove some kernels and build new ones. The process of removing the older kernels and replacing them with new ones will mean that the system is optimized. The optimization, in many cases, can result in a faster system for the user. However, this is just an advantage and not a reason to build and replace kernels.

Inserting modules
The presence of only three kernels leads to the need to add some more kernels to help enhance functionality. In the normal Linux OS, modules are kept in the */lib/modules//kernel/* directory, and they have a .ko extension. To insert a module into the kernel, one should use the *insmod* command. The user will simply type the syntax *insmod <module_name>.ko*
For example: # insmod /lib/ modules/ 4.4.0-21-generic /kernel/ drivers/cpufreq /SpeedStep- lib.ko

Removing modules
At times, one may decide that some modules are not useful at the time. They will, therefore, need to remove them from the kernel. To remove the modules from the kernel, the user needs to use the *rmmod* command. *rmmod <module_name>.ko* is the syntax for removal of modules. However, it is essential to note that a module in use by a program cannot be removed.
For example: # rmmod /lib/ modules /4.4.0-21-generic /kernel/ drivers/ cpufreq/ speedstep-lib.ko

Creating and loading a kernel module
One can be able to create and add their new kernels to their operating systems. Creating new kernels is a simple step process. Below are the steps to follow in the process of creating a kernel.

1. Add the header lines. The general format for adding the header lines is as follows.

```
#include <Linux/ kernel.h>
#include <Linux/ init.h>
#include <Linux/ module.h>
```

2. The next step is to include a description of the module being created, the details of the author or person creating the kernel. The license is also essential in this section. This step helps to specify the module being created by giving it a name, the author, and license. The general format to follow under this section is as follows.

```
MODULE_DESCRIPTION("Kernel module 2");
MODULE_AUTHOR("Author Name");
MODULE_LICENSE("GPL");
```

3. After the above steps, the user can then compile a kernel module before they can load the module. The compilation of the module is done using two files a Kbuild file and Makefile. When this process is complete, the kernel is ready

4. The final step is inserting the module. The insertion process involves the use of the *insmod* command

Why is Kali Linux the best for starters

Kali Linux is a Debian-based Linux distribution that is mainly used for advanced Penetration Testing and auditing security systems of applications and websites. Kali has a few hundred tools used in performing various information security tasks, such as Forensics, Penetration Testing, and Reverse Engineering. Offensive Security is a leading company in the information security industry and also helps in training on information security. The company developed funds and maintains the Kali Linux distribution which has been essential in the maintenance of programs and website security. On 13th March 2013, Kali Linux was released. Kali Linux is a Linux distribution that is generally a top to the bottom remake of the

BackTrack Linux. For a starter, the Kali Linux is preferred because it offers multiple advantages compared to other distributions.

Advantages of Kali Linux

Penetration testing tools

The Kali Linux is loaded with over 600 penetration testing tools. As a starter, one may not understand which is the best tool for penetration testing. Therefore, with the full range of devices, the hacker can pick one that they feel is easy for them to us.

Further, the extensive collection allows the user to have access to multiple tools to conduct the test. Using various tools allows the hacker to establish as many weaknesses to exploit as possible. For starters, the more the flaws, the better for the hackers as it makes the work easier. Therefore, Kali Linux has a significant advantage in the form of more penetrating tools.

Free to use

The Kali Linux is free to use. The developers of Kali Linux developed it for access by anyone. Charges on various operating systems mean that starters may not be able to afford to use the operating systems. Most beginner hackers may not have high levels of financial layout required to pay for operating systems .

Open source Git tree

The open-source code of the Kali Linux allows one to customize the OS. Customization is essential for beginners because it will allow the beginner to use an operating system that they understand. For a beginner, many things may be confusing; however, if they can customize the operating system, then they can understand the processes easily. The beginner can tweak with the packages so that they can develop one that will be easiest for them to use. With experience, the user can also make changes that will fit their new level of understanding of the operating system.

Secure environment

The Kali Linux OS is developed in a manner that protects the user both during acquiring and use. Starters may not understand how to protect themselves from some threats, and therefore the Kali Linux team can protect them.

Wide-ranging wireless device support

Some of the Linux distributions are quite rigid to the hardware they can be installed into. However, this is not the case for Kali Linux. This OS is developed and customized so that it is fit for use by multiple users on multiple platforms. Beginners may not have sophisticated hardware for use by some distributions, but the Kali Linux OS allows the user to access it through multiple wireless devices .

Multi-language support

One of the barriers for starters in hacking is that they may not find an operating system that supports their language. However, Kali Linux can support more languages. Although some of the operations can only be accomplished in English, it allows the user to use their native language for some operations. Accommodating more languages makes it easier for hackers to learn how to navigate around the system.

Kali Linux is secure

Kali Linux provides a secure environment that has minimal threats. This means that the user will not have to worry about risks such as malware, spyware, worms, and even trojans. Using Kali Linux does not even require on to use an antivirus. Therefore, these operating systems allow a starter to learn to hack without the high risk of being hacked. This is a great advantage as it will protect the user from threats that are possible in a hacker’s life.

Fast and stable

In hacking, the stability of the system is very important. Taking, for instance, that one has been able to hack into another system, and while they still have not completed their goal, their system crashes. A system being unstable or slow is one of the issues that can frustrate an aspiring hacker. However, the Kali Linux operating system ensures that the hacker is protected from the system crashing or failing during hacks. Speed is also essential for a hacker. Hacking is more of a race against time. The longer one takes before completing the hack, the higher the chances and risks of getting caught. A hacker will, therefore, prefer an operating system that is fast to avoid getting caught. Kali Linux provides the hacker with speed and therefore is a friend of the hacker.

These added advantages to the Kali Linux allow it to be easy to use. The added features make it easy for a beginner not only to set their own rules but also assist the beginner in the process of hacking. Therefore, the Kali

Linux operating system is one of the best options for a beginner hacker to use.

Why hackers use Linux

Linux Offers Granular control

Today, most hackers prefer to use Linux rather than Windows for a couple of reasons. For one, Linux allows hackers maximum control over their operations. Linux allows hacker granular control which implies that they can control almost every aspect of the Operating system. Unlike windows that restrict users to only a certain level of control, Linux offers its users total control. Hackers can easily and quickly program the OS using Python and BASH.

Linux is Open Source

Linux is quite different from Windows in almost every aspect. Open-source implies that Linux avails the source code to users, unlike Windows. Hackers are, therefore, able to manipulate the operating system's source code at any time. The fact that users can change the source code is what enables hackers to successful hack into other systems.

Linux is Transparent

Hackers have to understand their OS as well as that of their target. The transparent nature of Linux allows hackers to see just that. Linux allows users to access and manipulate almost every part of the system. Hackers can customize their aspects of the operating system. Linux also presents hackers with a command-line interface that is strongly integrated, stronger than what Windows has to offer. The fact that Linux offers hackers maximum security makes using it even more appealing to hackers.

Linux allows for Customization

Linux allows users to customize their desktop to their preferences. For users who prefer to customize their systems to fit their needs, Linux is more than perfect. Linux provides users with several icon themes in addition to the option of theme installation.

Linux is Reliable

Hackers prefer Linux to Windows due to its extreme reliability. Unlike Windows that requires one to keep re-installing software, Linux has none of that. Linux allows smooth operation and allows the system to run smoothly longer than Windows. Linux does not bother hackers to keep rebooting their

systems for reasons such as software installation. Another fact that makes Linux more reliable is because of the web servers that use it like Facebook .

Linux is Free

Linux is free to all members of the public, making it more suitable for all hackers. Linux does not require any fee from its users. Hackers can freely access Linux distro like Fedora and Ubuntu. Hackers barely spend anything on Linux, as they do not have to purchase any license before using the software. Linux has a general public license that saves hackers a whole lot of money. Most of the hacking tools are also written for Linux users, making their work even more comfortable.

Easy maintenance and Multitasking

Linux is easy to use and maintain. Hackers only need to install the software, and they are good to go. Linux has its software repository that makes using it even more comfortable. Hackers have access to embedded systems as well as high –performance applications when using this server. Linux allows users to perform several tasks all at once. Linux does not present issues such as hanging or freezing due to being 'overworked.' The system will enable users to run as many programs as possible without affecting each other as compared to Windows. Linux is also free from issues such as slow processing that can slow down the activities of hackers.

Linux is network friendly and Easy to Install

Almost all Linux distributions are user-friendly. The Linux distribution allows for the easy installation of the operating system. the distributions also provide hackers with user-friendly software. Linux also has a swift boot system compared to that used by other systems. All hackers need a faster network to complete their activities in the shortest time possible. Linux allows hackers to test their networks using various commands. Linux is not only fast but also reliable enough to allow access to a faster network compared to other systems.

Bash scripting

Bash is drawn from the initials, Bourne Again Shell. Bash is a command language with syntax. Bash uses natural human language to formulate the commands. Bash also has additional features such as iteration, job control, string manipulation, functions, expansion, and more. Each of these Bash features allows for maximum control and maintenance as well as batch

processing. Bash interprets and processes all commands fed into the shell. Bash forms a central part of system administration as well as the development of the system as a whole. A script is basically where a user writes their command. Linux has bash within its system by default, an added advantage to hackers. Shell scripting entails leaving the shell to interpret and execute the command A shell script is a program that shell has been tasked with to complete.

The bash script is the script that a computer follows when operating. The script tells the computer what to do or say after the execution of specific actions. Bash scripting is similar to a play or an act where the actors will follow and say what the script says or asks them to do. Bash scripting generally allows the user to use or input any commands that can be used or executed one by one in a script. Executing commands one by one and daily can be tedious as well as there would be many mistakes. Therefore it is essential to ensure that the task is reduced and that matters are automated. Automation does not only make things more comfortable; it also reduces mistakes that would otherwise prevent programs from running. Therefore, to avoid the process of having to execute commands one by one, a script is used. The script will include all commands necessary for a program to run being run together.

The development of a bash script only requires one to have basic knowledge of using the command line. the process has three steps to follow, as explained below.

Create a bin directory

This is a subdirectory that stores executable programs. This bin acts as a store or home for the bash scripts that are created. It can be done by applying the following commands.

cd ~/

mkdir bin

Export bin directory to the PATH

export PATH=/ home / $USER/bin: $PATH

Adding the bin to the PATH allows the user to be able to find the programs easily. Add your bin directory to your path. Edit your .bashrc file (or equivalent) to add in ~/bin to your path. copy executables into this bin directory or create a symbolic link from within your user bin directory to the executable you want to use .

Creating a bash script

When one is creating a bash script, they need to tell the computer that they are creating a bash script. The way to achieve this is through typing *#!/bin/bash* . This command will inform the computer that a bash script is being made. This will be followed by the commands that the user wants the program to display. There is a syntax that should be used when scripting. For example, when the user wants the computer to display something when the program is run, they will use the syntax *echo* followed by what they want to say in quotes. For example, you want the program to display the word welcome after it is run on the script; one will type the following:

echo "welcome."

When the program is run, it will display the word welcome on the screen. To input time between the response of the program and the next action, one will use the syntax sleep followed by the length of time they want the pause to be. For example

Sleep 5, the command will mean that the program will always pause for five seconds before it can execute the next command. When done one will simply press CTRL + X to exist, and they will be asked to save the file according to the name they gave the program. It is important to note that bash scripts can be very long. The size of the script will rely on the number of simultaneous activities that one wants the program to run automatically without the user having to get involved. For example, when one presses the shut down button on a computer, the computer runs on a bash script that allows it to save all the work and close the running applications automatically before it can power off. All these activities follow a bash script and so do many applications and actions on a computer.

Once a person has saved the bash script, they have to make the script executable. This involves changing the mode of the script to a version that is executable. Changing the mode will involve the user type the following.

Chmod +X <name of the file>

After this step, the user will then type

./<name of the file> then press enter

Once you press enter, the script will run according to how it was input and executed. The bash script will also include more syntax and commands that will help to condense the many commands that are run manually but can be run automatically. This will ensure that the user does not have to do too

much, but rather the program will run automatically. The bash script, therefore, provides the program with a certain level of automation.

Variables

Variables in bash scripting are essential in completing the whole script. The user has to assign variables and meanings to them. For example, one can assign the letter X a variable. On the script, one will type as follows.

#!/bin/bash

X= "1."

The variables mean that the user when they insert a certain variable in the code, suggests the value assigned to it. It helps to shorten the number of words in the script. Retrieving the variables involves typing the variable and syntax echo on the terminal. The process will go as follows.

echo $X

1

Syntax and the signs that are used in the development of the script are essential. Failure to use the right signs and syntax will mean that the results of the program will not be the same as expected. Therefore it is essential to get the syntax and signs right. For example, if one forgets to place the $ sign before the variable they want to retrieve, then the program will give an undesired result. Care should, therefore, be taken when scripting otherwise mistakes lead to programs failing to run. Avoidance of errors in the script involves constant testing of the script in bits completed to ensure that they are working. Proofreading the script is also essential to ensure that one has gotten the script right, syntaxes are correct, and even spelling is correct. Any mistakes in the scripts are reflected in the program. For example,

#!/bin/bash

echo "welcome."

Instead of

#!/bin/bash

echo "welcome."

This mistake will mean that the program will display the wrong information at startup. The program instead of displaying "welcome" will show "welcome." therefore proofreading the script is an essential step after completion of the script and constant running will help to identify such mistakes .

Automatic tasks

Automatic tasks are those tasks that the program will run on their own without having any manual interruption. The user does not necessarily need to input any commands for them to run. For example, when opening a program, the user will have to start it. The few steps and phases that follow are automatic. In most programs, once they are run, they begin by displaying the word ‘welcome.’ The word will then be followed by actions such as gathering information, then asks the user to input their password. All these actions run at a specified interval of time. While all these actions are running from the welcome message to request for a password, the user has not interfered in any way. The user may not have input any commands to the program as well. Therefore, these actions or tasks are running alone. The running of these tasks once the application has been running is what makes them automatic tasks. One action sets in motion another action or many more actions to help reduce the number of activities that the user has to carry out. The automatic tasks are also known as scheduled tasks as they are set to take place right after something else happens. They are, in a way triggered by other tasks.

Scheduling tasks on Linux

Scheduling tasks on Linux is like preparing and setting a sequence for actions or tasks to initiate themselves when specific actions take place. On Linux, one has to use the crown daemon that will help to run the tasks at specific times. When tasks are added to crontab files using the right syntax, then they will run automatically as previously planned. In many cases, scheduling serves in automating backups, maintaining systems, and even running repetitive tasks. At times the tasks can be too many for the user to complete single-handedly, they will, therefore, be automated to ensure that they run on themselves.

Scheduling/ automating tasks

The automation of tasks on Linux follows a few steps to complete. The steps will be executed as follows.

1. Open Crontab

In this step, the user will open a terminal on their computer. When the terminal is open, the user will type *crontab -e* . This command opens the

user account's crontab file. In this file, commands run on permissions from the user. If one wants commands to run on the system's approval, one will use the sudo crontab -e command. This command helps the user to access the root account's crontab file. The system may ask one to select an editor and in most cases, select nano. Nano is selected in many cases as it is the easiest to use. Selection is made by simply typing the number of the editor and pressing enter.

2.
Adding new tasks

After opening the crontab, the user will use the page down tab to move to the bottom of the file. From there, they can schedule their tasks. There is a specific format required to be followed when scheduling tasks or automating them. Setting the time for the tasks to begin, one used the following format *minute (0-59) hour (0-23) day (1-31) month (1-12) weekday (0-6)* command. In this command, it is imperative to note that in minutes and weekdays, the first is 0 this means that the first day of the week will be given zero, thirty minutes will be 29 and so on. The asterisk (*) is used to show repetition and can be used to represent any value. For example, if you want a command to run every day at 12.30 am, the command will be as follows.
*29 0 * * * /usr/bin/ example*
Use values separated by commas to specific multiple times. For instance, a line such as
*0,14,29,44 * * * * /usr/ bin/ example2*
runs */usr/bin/example2* at the 15-minute mark on every hour, every day. Make sure you add each new task on a new line.

3.
Save the file

Once you have scheduled the tasks, the user can then save them as they want. The saving process is achieved through the use of *Ctrl - O* then *enter.* After saving the file, one can exist by merely using the shortcut *Ctrl - X.*

Chapter 6:
Process of Web Hacking

Web hacking

Web hacking is a common practice all around the world now with the main aim being to acquire access to data on the websites. While it is not possible to hack every site on comes across, one can hack some websites. The level of difficulty in hacking a website varies depending on how protected the site is. Therefore, you will find that while some websites are straightforward to hack, some cannot be hacked due to their security levels. The presence of vulnerabilities on the site facilitates web hacking. Many methods can be used to hack a website. Some hackers might use cross-scripting, SQL injection, among other methods of hacking a website. Below are steps that one can follow to hack a website.

1. *Analyze the website for weaknesses or vulnerabilities.* A website can be vulnerable in its infrastructure or even programming language. The hacker should first begin the website hacking by identifying these weaknesses to the system as they will be the way into the websites.

Types of vulnerabilities

There are very many types of vulnerabilities that are mostly present in websites. Every vulnerability can be exploited in its way. The more vulnerabilities in a website, the easier it is to access and gain control of the website. Some of the potential weaknesses of websites are as explained below.

- SQL injections- this can be a string of code that can be used to help the hacker upload, edit modify, and tamper with the information on the website. Once SQL is successfully inside the site, it will not work as it should. This alteration in functionality allows the hacker to access data that they want, alter it, or complete any activity that they need.
- Cross-site scripting- in this type of vulnerability, the attacker will inject malicious code into the website through any weakness noted. The script will start to steal each of the visitors' cookies. For every visit to the site, the script is activated and cookies are stolen. The code then sends the cookies to the attacker. Through the cookies,

the attacker can have access to visitor's data, and they can use it to access the website when they want.

- Broken authentication and session management- here, the attacker will focus on attacking active sessions. Once they have attacked the active session, they will then use the credentials of the user to access the website. Access to the account means that the attacker can collect any data and information that need.

There are other types of vulnerabilities that can be exploited in a website. The more the vulnerabilities, the easier to hack as well as the higher the threat. The threat is not only to the website owner but also to the users of the websites as they can quickly lose data or get attacked as well.

Ways to find vulnerabilities

There are many methods through which a hacker can use to scan a website or program for vulnerabilities. The hacker can try various methods if one does not work. The methods include the use of VEGA, using ZapProxy, the hacker can also use database tools to assess for vulnerabilities. These tools include sqlmap, SQL ninja, among other tools. The hacker can also use CMS scanning tools which include WPScan, Joomscan, and other tools. Another set of tools to use in looking for vulnerabilities in the SSL scanning tools and the W3af. The hacker can use any of these tools according to their level of competence in using each of them. The more scanning tools that a person knows how to use the better as they have many options to help find vulnerabilities. Hence, it is essential that one learns how to use multiple scanning methods as they will help them to scan for problems quickly and identify vulnerabilities even where they are well protected.

2. *Exploit the vulnerability to help you gain access* . In this step, the hacker already found the vulnerability of the website. They need to use it so that they can gain access to the website and the content.

Ways to exploit vulnerabilities

Exploiting vulnerabilities is a process that involves using a tool to help slow or even destroy how the website works. The hacker can use the weakness in the systems in the following ways to help in gaining access. The vulnerabilities can be used to introduce malicious code, data, scripts, among

other material that can affect the website and its functionality. Through the introduction of this material, the website may be slower than usual, deny access to users or even introduce traffic on the website. Through doing this, the hacker will have time to gather the data that they want and cover their tracks as they leave. The hacker will use the time the admin uses to solve the problem to complete their mission.

3.
Maintain access to the website

Maintaining access is a very important stage of a hack. Most of the functions of a hack depend on whether the hacker can maintain access. Being able to maintain access will help the hacker to complete its mission successfully. Hackers have to keep access to the system to allow them sufficient time to complete their activities and leave. Maintaining access can be done in multiple ways. One of the ways is through being able to ensure that the hacker remains undetected for as long as possible. Therefore, the hacker in such cases will ensure that the logs cannot give the administrator information about their presence. Maintaining access can involve the hacker denying the administrator access to the system for as long as they can while they complete their mission. Through executing a Denial of Service (DOS) attack or a denial of service, the hacker can ensure that the users of the system cannot access it. This will ensure that the administrator is also unable to access the system or can only be able to execute a few activities that cannot be able to expel the hacker. However, most hackers prefer to steal usernames and passwords for other users. By keeping these login details, it will help to ensure that the hackers can reaccess the system at a later date. They can be able to access the system as many times as possible without detection as they will look like an average user. Either way, the hacker chooses to use, they should be able to maintain their access to the account or program for as long as they need to ensure that they have been able to complete their mission.

4.
Cover tracks once have done.

Hacking, whether ethical or unethical involves accessing programs or systems without the permission of the user or administrator. This, therefore, means that to some level, a hacker is an intruder. Therefore, after a hack,

mainly if any critical information has been accessed, the owner of the program will be looking for the hacker. In some cases, legal actions can be taken against a hacker. It is, therefore, essential that one does not get caught. During a hack, a hacker leaves tracks and footprints. These footprints can be followed to establish who the hacker is. Therefore, it is essential to cover the tracks of the hack and ensure that one cannot establish who the hacker is. There are multiple ways of being able to cover one's tracks. The ways are explained below and how one can be able to successfully their tracks after a hack .

Disable auditing is one of the ways that one can use to cover their tracks. Once a hacker has gained access to the system and has administrator privileges, they can simply turn off auditing. Audits allow the system to collect information on what is happening in the system. Disabling verification will enable the hacker to go about their business without being recorded. Once they are done, the hacker will enable verification. Enabling verification will only require the hacker to run the ***auditpol.exe*** program. Without audit records, the administrator cannot tell what happened in the system for a specific time.

Clearing logs is another way of covering one's tracks during a hack. As one may already know, a system, program, or applications keeps records of activities. These records of activities can be identified as logs. Therefore, when a hacker enters an order, the records of their activities in the order are recorded. The administrator will be able to tell what happened in the system through reading the logs. After a hacker is done, they can clear the logs so that there is no record of activities during the hack. When the administrator looks at the system, they will not be able to establish what was done efficiently. This will also mean that they cannot be able to establish who hacked their system. Some of the utilities that can be used to clear logs include ***clear logs. exe*** and ***meterpreter shell.***

Modifying logs and registry files is where the hacker will change the logs and records of the system. Once the system has recorded the activities of the hacker, they will simply edit the logs. The hacker will replace the logs with activities that the administrator would expect to be happening. This method is quite safe as it ensures that the administrator will not even suspect that there was a hack into the system. The administrator will simply think that the system has normally been running even when there was a hacker in the system.

Removing all files and folders created is one essential step that needs to take. It is seen as a complementary action even when other measures to cover tracks have been taken. During a hack, a hacker may need to upload files, create folders, and create files that will help to access the system or perform their activities. Most administrators know what is in their systems, and therefore they can be able to identify any foreign files in their system as quickly as possible. For this reason, it is important that one deletes the files that they have uploaded or clears the new folders created. If the foreign material is present, but other methods have been used to cover tracks, the administrator will still manage to tell that there was a hack. It is therefore essential that one clears the materials that they have introduced into the system.

Penetration testing

Hackers are only getting smarter with their hacking. Organizations have to try and outsmart these hackers in an attempt to keep their private information, like trade secrets, private from any unauthorized persons. Advanced penetration testing is a risk management strategy that identifies potential loop-holes in network security and system security. After identification of all potential risks in the system, then prompt action is taken to try and control that risk. Penetration testing can be described as a cybersecurity strategy as it unveils any security weaknesses within the system.

How Advanced Penetration works

IT experts have the duty to ensure organizations' network security and systems are safe from all hackers. Through advanced penetration, they ensure that all hacking attempts are detected and stopped before they even occur. Advanced penetration usually involves experts utilizing the same techniques or methods often used by hackers to get into their systems. Hackers begin by monitoring an organization's technical landscape to try and identify any loopholes they can get through. Once they identify any loop-holes within the system, then they go-ahead to launch their attacks. Similarly, pen testers take their time exploring the network and security system and simulate similar attacks to try and identify those areas which could be potential loop-holes. Pen testers use penetration testing when they

are augmenting a web application firewall (WAF). The main aim of penetration testing is detecting any inputs that could be at risk of code injection attacks.

Penetration testing stages

1) Planning and reconnaissance

The first stage in pen-testing involves outlining a clear plan for the test. The pen tester has to lay out a strategy with goals on how they will carry out the test, the systems to be tested, and the methods that will be used during testing. The pen tester then goes ahead to gather all relevant information concerning the target for the test. For instance, the pen tester needs to seek intelligence on domain and network names before conducting the test.

2) Scanning

The second stage involves trying to predict how the target application will respond upon intrusion. Pen testers have access to two tools that they can use to understand the target application more effectively.

a) Static analysis

This tool allows pen testers to investigate an application to determine how it will react while running. This method involves examining a code without necessarily running any program in the system. Static analysis allows experts to have an in-depth understanding of the code structure. Static analysis allows for identification of any system errors that could potentially make the system vulnerable to cyber attacks

b) Dynamic analysis

After conducting a static analysis, a dynamic analysis should be conducted right after. Dynamic analysis operates by identifying more subtle errors that could not be identified during the static analysis. Dynamic analysis involves investigation of the code while running a program. One advantage of dynamic analysis is that it provides real-time information that allows easy identification of vulnerabilities within the system. Other than being reliable in the identification of errors, the dynamic analysis also allows programmers to eliminate programs that are unnecessary in the system. The dynamic analysis also cross-checks on the compatibility of the program being tested with other programs.

3) Gaining Access

In gaining access, pen testers have to use web application attacks to try and identify vulnerabilities within the target. T launch this attack, programmers

can use any of the web application attacks running from backdoors, cross-site scripting, and SQL-injection. Once an attack is launched, pen testers use different strategies to try and exploit all vulnerable points. Testers can either intercept traffic, steal available data, or escalate some of the privileges in an attempt to see how the system will behave.

4) Maintaining access

The next step involves investigating how long hackers might last in the system after they gain access. Testers have to imitate advanced persistent threats to try to determine how long attackers might last within their systems. Advanced persistent threat attacks are often executed with the intention of spending longer periods within the system with the aim of gaining in-depth access to organizations' systems. Through imitating this type of attack, testers get to see how long a potential threat could last within their systems.

5) Analysis

The final stage involves a thorough analysis of all the activities conducted during penetration testing. Pen testers write a comprehensive report that includes;

- ❖ All sensitive data that was accessed
- ❖ All vulnerabilities that were exploited
- ❖ Amount of time is taken within the target before detectio n

Penetration Testing Methods

1) Internal testing

In this case, a tester simulates attacks by a ‘malicious insider.’ In this case, the tester has access to the application as they can get behind the system's firewall.

2) External testing

Pen testers during an external testing target all company assets that can be accessed on the internet. The goal of external testing is to access either company domain name servers or company email to try and extract data from these targets.

3) Blind testing

In blind-testing, the tester is only presented with a company name. the tester then has to act like a real hacker and try to maneuver their way into the system. Organizations use this method to study real-time hacker activities.

4) Double-blind testing

In this case, both the tester and the security personal goes in blind. The tester only has a name to work with. The security personnel is also not informed prior to the attack. The main aim is to study how fast the security personnel will respond to an attack.

5) Targeting testin g

In this case, the testers work hand in hand with the security personnel. the testers update the security team of any movements they make within the system so that the security tea, can counter with a secure move.

Penetration Testing Methodologies

1. Black-box penetration testing

In black-box testing, the pen tester in given no source code or any vital information on the system. the tester lacks prior knowledge of the system, and the testing relies entirely on their analysis of the systems and any currently running programs. Testers have to utilize all available scanning tools to try and identify any vulnerabilities. Black-box testing has proven to be inefficient as testers ay end up not detecting any vulnerabilities for failing to breach the security system.

2. Gray-box penetration testing

A grey-box tester, on the other hand, has some level of knowledge on the network's internal system. Gray-box testers are well-informed of the systems architecture and documentation. A grey-box tester only has to focus on systems that pose the greatest risk. Pen testers have to prioritize on high-risk systems, instead of wasting time trying to understand the information first.

3. White-box penetration testing

White-box testing is also known as open-box or auxiliary testing. Much like gray-box testing, pen testers are provided with full information on the system and any architectural documents. In this method, testers conduct static code analysis as well as dynamic analysis. White-box penetration allows testers to conduct both internal and external assessment of the system. By combining both static and dynamic analysis, pen testers ensure that they do not miss any vulnerable entry points that hackers can maneuver their way into. white-box penetration is a more integrated target method as

it involves pen testers working together with developers to identify and fix all vulnerabilities.

- ❖ Speed, Efficiency, and Coverage

Black-box testing is the fastest method in penetration testing when compared to gray-box and white-box testing. One shortcoming with black-box testing, however, is on the fact that testers have limited access to any information pertaining to the system. the limited information lowers the efficiency of the methodology as testers may end up missing some of the essential vulnerabilities. Pen testers using black-box testing are bound to miss some vulnerabilities as they have little to no information on high-risk targets within the system.

Gray-box testing, on the other hand, provides wider coverage though slower compared to black-box testing. When using gray-box testing, testers are provided with substantial information on the system and its documentation. With this information, testers can efficiently cover a wide area and effectively analyze all potential vulnerabilities. Testers using gray-box can effectively gain internal access to the network and identify all vulnerable targets.

The most effective methodology is white-box penetration testing. White-box testing is more sophisticated compared to gray-box testing and thus much slower than the other two methodologies. The comprehensive data that testers are presented with take time to analyze and understand, and thus the penetration test itself will take longer with this method

- Engagement and Accuracy

Out of the three methodologies, black-box testing is a more real-life method as it emulates a real-life hacker. Much like hackers, testers using black-box testing have no information on any architectural documentation. Black-box testing is more of a blind testing method as testers have no prior knowledge of the system. white-box and gray-box methodologies were specifically designed to reduce engagement time for testers. Allowing testers access to crucial documentation means they will know the areas to target. With white-box testing, however, testers are still at risk of missing some vulnerabilities as they have access to all the information. Testers may end up overlooking crucial areas and missing vulnerabilities entirely. Gray-box testing is more of a break-even point for the three methodologies. While black-box testing offers no information, white-box testing offers testers all

crucial data. Gray-box testing, however, allows testers only limited access to the system's data, simulating the period a hacker would take studying their systems.

Advantages of Penetration Testing

I. Allows detection of security threats

Penetration testing will enable organizations to anticipate any potential security threats. Pen testers simulate the specific activities of hackers through launching attacks on their own systems to identify all vulnerabilities. The accuracy, speed, and coverage of the detection process rely entirely on the methodology an organization decides to use; such as black-box testing, gray-box testing, or white-box testing. Once testers identify vulnerabilities of the system, security personnel respond by securing the system using new and sophisticated tools that hackers cannot get through. Security experts have to ensure they arrange all the potential threats and prioritize these vulnerabilities. Penetration testing allows security personnel to defend their systems form both external and internal attacks before they even occur.

II. Meet monitoring necessities

Organizations operate under guidelines provided by various legislations like the SARBANES – OXLEY. Organizations have to comply with each of the regulations stated within the legislation to avoid paying any penalties that may arise. IT experts, for instance, are bestowed the duty of overseeing that all other departments comply with each of the regulations. IT departments also have to ensure they meet the testing standards in NIST/FISMA commands by availing a comprehensive report of the penetration test results. The results of these tests act as proof of the steps the IT department is taking to safeguard their systems. The report also provides regulatory bodies with some of the steps the organizations are taking to boost their security system.

III. Safeguard company image

Company image relies entirely on the safety of their systems. Companies that are associated with frequent cyberattacks are bound to have a bad public image. Customers prefer to associate themselves with a company

with a safe security system that will prioritize their privacy. Customers want companies that will take all steps to ensure their data is safe from any malicious persons. Detecting system vulnerabilities protects the company from all potential threats. Having a strong security system ensures that all vulnerabilities are addressed and a safer security system installed. Penetration testing points out the vulnerabilities that are then addressed by IT experts who install a safer system that protects crucial company information.

IV. Prevents operation downtime

Security breaches not only cost organizations financially but also waste time and resources. Companies have to undertake several activities and use multiple resources to their systems back to safety. Security breaches may result in reduce employee performance, legal troubles, and lower revenues for companies. All of these consequences end up putting a complementary great financial strain on the company. Penetration testing operates by detecting security breaches before they even occur, saving the company on losses made during downtime.

Disadvantages of Penetration testing

Penetration testing gives organizations a false sense of security. Once pen testers ascertain that the system is able to withstand all penetration attacks, it may seem as though the system is totally safe from hackers. On the contrary, pen-testing only outlines possible vulnerabilities that may or may not be targeted by hackers. Hackers attack when security personnel is most unaware and counteracting the movements of hackers is usually almost impossible. Penetration testing often occurs when security personnel and well prepared and well aware of their movements which is not the case during a real attack by hackers.

Penetration testing is also labor-intensive and can put a financial strain on a company. Companies have to budget for getting experts who will thoroughly assess their systems and identify all potential vulnerabilities. In other cases, an external expert may be required, which will be even more costly. Methodologies like gray-box and white-box penetration require companies to provide documentation and all relevant data on the system to the pen testers. The labor that goes into the data collection, organization, synthesis, and analysis may be a discouraging factor to some companies.

The engagement period that it takes to successfully go through data and understand it may require more resources than a company is ready and willing to allocate.

Pen testing is not a conclusive security audit. Combing through the system for vulnerabilities does not guarantee that one will find all potential areas of interest to hackers. The methodology used also determines the range of vulnerabilities that the tester can discover and help the company secure. Pen testing can also be a distraction to normal operations as it involves probing all security systems.

Conclusion

The Linux operating system is flexible compared to the windows operating system. The flexibility makes it very good for hackers to use which would make this easier on you. Linux works as a multi-front operating system and can serve different purposes according to the customization. Unlike other operating systems, Linux comes only as a foundation on which one builds their operating system. The OS is booted in order to let the users add what they need as they customize it to fit needs. The first step into learning how to hack using Linux is to understand the Linux operating system which is what this book has helped you with. Once you are able to understand the basics you can move on to the more complicated aspects of this subject such as networking.

After learning how to use the OS then you can move to the other steps that include the following:

- Understand more about networking
- Learn how to obtain the necessary hacking skills in using Linux and learn how to script

A person needs numerous tools, following phases and skills to complete a hack. The steps include the following:

- Reconnaissance
- Scanning
- Gaining access
- Maintaining access
- Clearing tracks

These are extremely important to pulling this off successfully which is why they are important to make sure that you have this down. It is essential that one when scripting can understand the syntax, as they are necessary to ensure that the script is executable later. Combing through the system for vulnerabilities does not guarantee that one will find all potential areas of interest to hackers.

This book has also been able to give you the information on text manipulation and understand why it is important. If you can use this to your benefit, you will be able to perform the tasks that you need to with ease and set the words up the way you need to. This is a useful skill that will come in handy.

Hacking is a very complicated series of processes that take a lot of effort and there are many things that you will need to learn to make sure you are doing correctly instead of incorrectly. Hopefully, this book gave you the most basic information so that you will be able to do this properly. If you are able to follow these tips and use the information that we have given you in this book, you should be able to perform the tasks that you need to with ease and learn how to understand the Linux system without any difficulty.

Finally, if you found this book useful in any way, a review on Amazon is always appreciated!

Hacking with Linux

Underground Beginners Tools to Learn the Basics of CyberSecurity and Become a Hacker by Breaking into Every Operating System with Ethical Linux and Precise Computer Configuration

Darwin Drowth

Introduction

Congratulations on choosing the book *Hacking with Linux: Underground Beginners Tools to Learn the Basics of CyberSecurity and Become a Hacker by Breaking into Every Operating System with Ethical Linux and Precise Computer Configuration.* There are many books on this topic, but you choose this title to help you understand the world of hacking with Linux.

Hacking is a trendy and cool thing right now. More people and businesses have incorporated technology into their lives to make work easy. Hackers have also benefitted from advanced technology because they have tools to further their agendas. There are plenty of tools used to hack into systems without detection. While companies have put in measures to stop hacking activities, some have not been fortunate to stop it entirely.

As you continue to read, you will get educated on the basics of Kali Linus, the concepts and much more that will help you create a complete picture of what you need to know in regards to Linux operations. You will also understand the complexities that surround the use of Linux and why it still comes out as a top choice for many.

As already mentioned, many other authors have penned down books about Linux, and it is indeed a privilege that you chose this book to help you advance in the use of Linux. I say thank you again, and as an assurance, I can certify that this book was prepared with the utmost care to ensure that it will not only be useful to you today but in the future as well.

Chapter 1:
Basic and Advanced Linux Concepts

Linus Torvalds developed Linux, an open-source OS in 1970. Because it is an open-source operating system, it can be modified by many users. It is also possible to create different variations of the code. Linux has different variations commonly referred to as distributions which can be used by several computers. There are primary and advanced Linux concepts that users must be familiar with before starting the installation process. Basic concepts include:

- PWD: Users are supposed to be at certain paths whenever terminals are opened. PWD is used to know the path of the directory. It is effective because it reveals the current path and gets users in the right way. Moreover, it highlights the whole path where users are present. PWD stands for the Present Work Directory. It makes work easy by giving the absolute path which comprises of the root, the basis of the Linux file system.
- CD: This concept is used to move to a certain directory. It is helpful because it gives direction and saves time. For example, if you are in the project folder and want to go to the home folder, the CD comes in handy. All you have to do is to write the cd home name, and you will be taken to that particular folder.
- Is: This concept is an initial for a list. Its work is to list all the directories and files that are in a certain directory. One can use it alongside other combinations and get the results. It helps to get hidden and hard to get files. It also helps to get recent files on a directory.
- Touch- This is a concept that is used to create a file on any path. It has no restrictions, and the file can be anything. One can create an empty, text file or zip file.
- Sudo- This concept is widely used in Linux. Users employ it to perform any task and get administrative privileges. It is preferred by many because it makes work easy and gives them access to services reserved for some people. Sudo stands for SuperUser Do.
- The Terminal- Any time you want access to the cloud server, you must use a terminal shell. It enables users to carry out commands within minutes. Moreover, administrative tasks can be done through the terminal. It facilitates package installation, file manipulation, and user management. Owners are not worried about poor usage because they can monitor and track users. It is also interactive, and users only need to specify commands for it to run. Once the user types and presses enter, it executes the command.
- Navigation. All Linux file systems are categorized in a directory tree. It allows users to create directories inside other directories and gives them easy access. This means that files can be in any directory. PWD helps users to see which directory they are in. For example, PWD can create /home/foo, which means that the current directory is foo. A user can also create a new directory in the current working directory. For instance, to create a directory roof: mkdir roof, then cd or delete roof if it is no longer useful. The user will only delete empty directories.
- Another basic Linux concept is file manipulation- It is impossible to use files with CD (Change Directory). We can only view files. For example, if we have file biz in our current directory, it translates to cat biz. It prints out the information of biz to the terminal. It is impossible to read content in the log files. The output can be paginated into less biz to facilitate the printout of the contents of the biz. It will be done one terminal at a time, from the beginning of the file. A spacebar can be used to advance a page. Alternatively, arrow keys can be employed to move up and down on a line. You can press q to remove less. The user can create a new file called foobiz by using the touch bar.

- CP: This is another concept that plays an integral role in Linux. It is used to copy files from one source directory to another. For it to work, the first source needs to specify the location where the files will be copied. The destination also needs to be on the second location.
- MV: This concept is used to move a file from a specific destination folder. It creates space by deleting the copy from the source path and placing the file in the new destination. Users can also use the Echo when they want to move data. Other basic but important Linux concepts include cat, nano, and tail. The tail is employed to print the last lines of files. It shows the last ten lines on the output. If a user wants to see more lines, he/she needs to specify tail-n. N stands for the number of lines you want to see.

The Kernel- Every computer system has basic programs known as operating systems. The Kernel is the most important program in the basic set. It is put in the RAM when the system boots and has vital steps required for the system to function optimally. Other programs lack vital utilities. The kernel provides interactive experiences for users and users and enables a computer to do all the work it is meant to do. It also gives crucial facilities to everything on the system and gauges various features of higher software. Therefore, the operating system is used in place of the kernel. The operating system has two main functions: To avail of an execution environment to the applications that run on the computer system. The second function is to interact with hardware components and service programmable elements. Some OS enable all user programs to interact with hardware elements.

On the contrary, the Linux OS hides all the details about the physical structure of the computer from the applications used. This is where the kernel comes in handy to evaluate the request and interact with hardware components. Modern OS depends on the accessibility of particular hardware features to instill this mechanism. It relies on characteristics that do not allow user program interactions with low-level hardware parts.

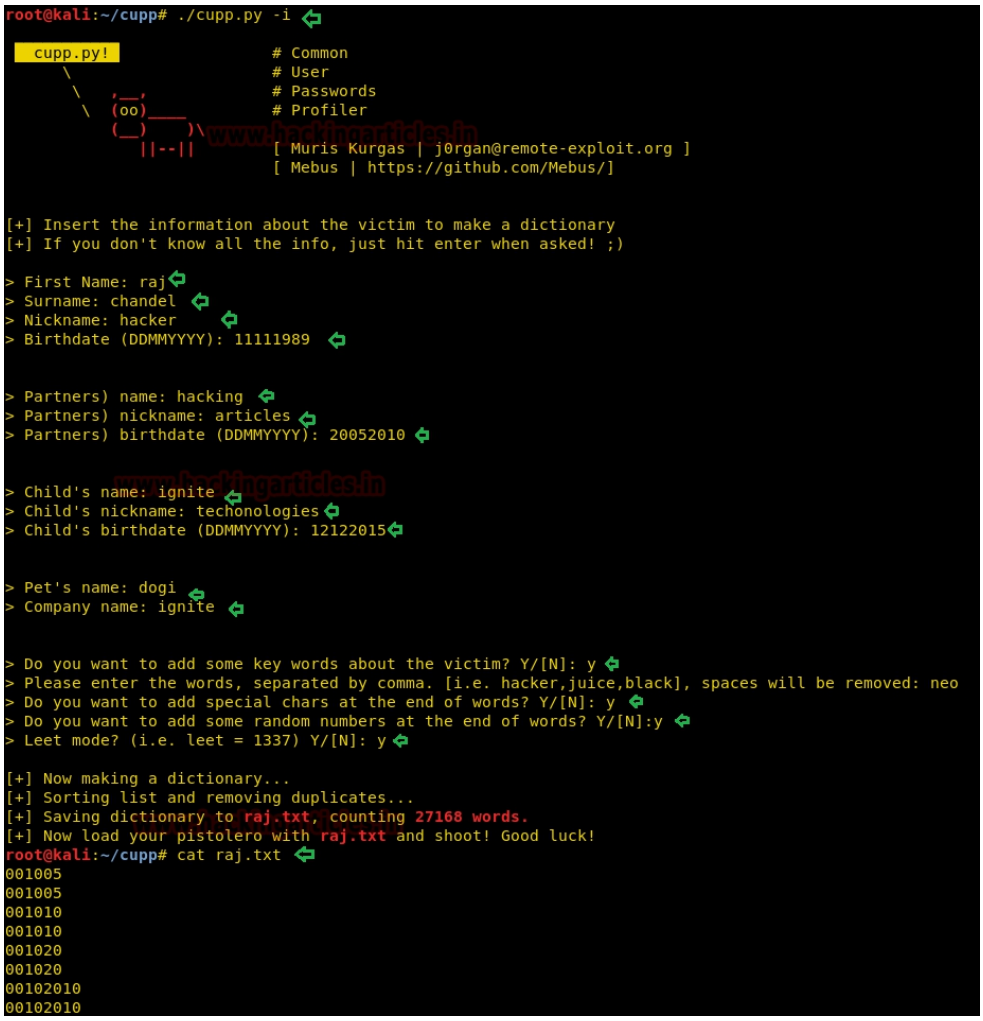
Multiuser systems- This is a computer that can carry out numerous applications from many users. It executes them independently and concurrently. This means that applications are active in the same period and struggle for similar resources, including memory, CPU, and hard disk. A computer executes functions independently when each application does its tasks without worrying about what the other one is doing.

Every application is self-reliant and does not need the other one to function. When a user has to switch from one application to another, it slows them

down and makes it difficult to move to do other things on time. Most of the issues with modern-day OS kernels are there to reduce delays put on every program to give users fast response. Multiuser systems should have several components, including a protection mechanism to fight off buggy users. It should have programs to block external applications from running in the system. It needs an authentication mechanism to verify user identity. OS should use the hardware protection linked to the CPU privilege to guarantee protection mechanisms.

Every user has privacy when using a multiuser system. Users are given about a quota of the disk space to keep sensitive information. The privacy portion must be visible in the operating system. It ought to ensure that no user can intrude into another user's privacy. It ensures the privacy of every user and treats them equally. User ID is used to identify all users and restrict the number of people who have access to a computer system. Whenever a user starts a working session, he/she must log into the system. This requirement is in place to prevent unauthorized access to the system. Users log in with their names and passwords. Because the password is a secret, users are guaranteed privacy. Users are put in groups to enable them to share information with other members. Every group has an ID which they use to interact and share information. For instance, access can be limited in such a way that every user owning the file is required to read and write privileges. Users who do not meet set requirements are denied access to the file.

There is an individual user called root and the system administrator logs in using that name to tackle user accounts. The administrator also maintains the accounts and upgrade programs. Operating systems use a crucial abstraction called the process, defined as the execution context. The role of a process is to execute one sequence of requirements in an address. Multiuser systems are supposed to execute environments where numerous processes are active at the same time and compete for system resources. One should be able to distinguish processes from programs with difficulty. The majority of kernels are monolithic.



Advanced Linux Concepts

There are plenty of advanced Linux concepts everyone should know. The first one is the VI Editor. For instance, you can use GUI tools such as Kate to edit text files easily. They all look like notepads in Windows but are more powerful. Moreover, they have notable features which make them unique, including syntax highlighting and multiple file editing, which makes work easy for you. Whenever you open an HTML folder, it understands and highlights them. You can also use the Vi Editor to simplify work. Many people are confused when they first encounter vi editors, but soon realize that it is relatively easy to use. The first time you open a folder in vi, it can be difficult to type, and the systems beep continuously. This can be frustrating and cause many to give up. There is confusion in the beginning because the vi editor works in two modes. The first one is the

command mode, where commands are given. You can command it to open or replace the file. The second mode is the insert view which allows users to type texts. Users should understand that vi does not have an appealing GUI interface but is still efficient. It edits texts powerfully and facilitates learning. Knowing that many people find vi editors difficult to use the first time they encounter it; you can use a tutorial to understand how it works. It opens v with the tutorial in it.

Another advanced Linux concept is Grep. It is a useful concept that searches any file a user needs. It is commonly used alongside other concepts to search for specific strings. In case a user wants to check out the log file on a web server, the grep does the job correctly. If you want to determine the occurrence of “right now” in the text, all you have to do is type grep “right now” text, and it pops up. You can proceed to check the grep’s location by typing location grep. There are many options to choose from. Another advanced Linux concept is PS. This is a command tool that displays all the tasks being run on a system. It does the same work as Windows Task Manager. There is a GUI version of ps. You can see the process running on a system simply by logging in as root in the Linux system and typing ps-aux. However, users can only see processes run by them for security purposes. In the past, it was discovered that some people mishandled other people’s processes and security measures were introduced.

Kill is another important concept of Linux. It is a command that complements the ‘ps’ command and allows users to end the process that is revealed by previous commands. You can use syntax to kill a process in case it is not responding. You can use kill-0 PID to eliminate the unresponsive process. -HUP can also be used with the ‘kill’ concept. It does not get rid of the process, but pauses and forces it to reload its configuration. The -HUP comes in handy if you have a running service and wants to restart owing to changes made in the configuration file. It is a great alternative to the reload command. It is possible to chain groups of commands in Linux because of the great power of Linux concepts. You can employ tools to perform tasks and pass to another one. For instance, when you run the ps aux command, unreadable outputs may pop up on the screens. The pipe symbol can be used to send the output to ‘grep’ to search

for it. This process is called ‘piping,’ where a pipe is used to connect two things.

Alias is a great concept in Linux to use. It is a neat command which allows users to make shortcut keywords for long commands. For example, if you do not want to type ps aux/less in full, you can create an alias and name it ‘pl.’ so instead of typing it in full, you just time pl, and it runs the ps aux/less. It makes work seamless. Users may want to redirect outputs of commands to text files for more processing. To do this, open a DOS command prompt and use the ‘>’ operator. The good thing is that these functions are supported by Linux. Watch is an excellent command tool to execute programs periodically and place the contents on the screen. You can run several commands if you put the command inside quotes.

Chapter 2:
Linux Installation

Kali Linux Installation Requirements

Before you jump into the bandwagon, you need to understand that Kali Linux is not everyone's cup of tea. Just because it delivered amazing results for someone does not mean it will do the same for you. just because someone else found it hectic or irritating does not mean you will have similar experiences. It boils down to your needs and knowledge. Kali Linux has become popular over the years, and there is a good explanation for it. Nowadays people find hacking cool thanks to TV series and the thrill that comes with it. More people are becoming hacking experts in the comfort of their homes. The strange thing is that people with barely any knowledge in the computer of Linux are using Kali as the main Linux distribution. There are many Kali Linux tools used for hacking, and they are increasing every day. However, Kali was not meant for that purpose. What many people do not realize is that it is not a must for them to use Kali. On the contrary, it is a distribution tool that makes a specific task easier.

Now that you know how amazing Linux is, you must be eager to install it. However, you need to keep in certain things in mind before you rush into installing the software. You must know its hardware requirements and challenges. Everything has benefits and drawbacks. Understand that Linux was developed by users which means that users have access to its hardware. Another challenge is that many firms have decided to keep the hardware interface propriety. This is a challenge to Linux volunteer developers who cannot write drivers for such devices. Firms that choose to keep their proprietary interfaces write their drivers for OS and users know nothing about the interface. Something can be done about the situation in some cases. Hackers have tried to write drivers by examining assumptions about the interface.

Linux consists of particular features reserved for laptops, such as APM and PCMCIA with drivers to support the devices. The majority of hardware support for Linux is still being developed. Therefore, some distributions may or may not existing features. Users should keep an eye on hardware suppliers who change the latest version of a system component without considered what was ordered. If you want to install Linux, there are CPU and motherboard requirements to consider. Currently, Linux supports

systems with intel 80486, Pentium Pro, and Pentium. It has also been ported to non-intel architectures such as Motorola, MIPS, AXP, and SPARC. Some ports are naturally more mature than others. People who have intel 80386 and beyond should use math coprocessors to make work easier. Moreover, the system motherboard should employ EISA, ISA, and PCI bus layout. It is important to pay attention to these terms because they determine how the system interface will look like.

You also need to pay attention to memory requirements before you install Linux. Linux is preferred by many because it needs little memory to operate, compared to other operating systems. This means less hassle to get a big memory for installation. You only need 8MB of RAM, and you will be good to go. To be on the safe side, have at least 16 MB. While a big memory is not needed, having more space will cause the system to run faster. Having more memory equals to having a processor that works fast. If it is for personal use, 16 megabytes is enough. Linux users assign a portion of the hard drive as a space to be used as a virtual RAM. It is proper to use a swap space even if you have sufficient physical space.
Although swipe space cannot replace physical RAM, it enables the system to run big applications by removing inactive parts of the code disk. Several factors influence the amount of swap space you allocate. There are hard drive controller requirements you need to keep in mind. Floppy is more than enough to run Linux. It is not a must that you have a hard drive to run Linux. However, the standard way of doing things is to use a Linux with a hard drive. Linux must support important IDE controllers, and if it is possible to access the drive from Windows, the same should apply to Linux. You need to create space on the hard drive to install Linux. The space required depends on the amount of software you are installing at a time. Your needs also influence the amount of space needed. Moreover, you should create space on the hard drive for virtual RAM.

For those who are pursuing a career in information security, having a security-oriented operating system is crucial. An effective operating system helps in doing tedious duties on time. There are numerous Linux-based operating systems, but Kali Linux is still the best of them all. It is used for ethical hacking. Network security assessment and penetration tests. The best Linux distribution so far is Kali Linux for several reasons. Regarding hacking, penetration testing, and ethical hacking, Kali Linux is the best

option. It is loved by many because it comes pre-packaged with several command hacking tools. It is arguably the best tool for ethical hacking around the world.

The Installation Process

It is relatively easy to install Kali Linux on your computer. You need to have compatible computer hardware to facilitate the process. The hardware requirements are not many, but having good hardware leads to better performance. You will need at least 20GB disk space before you start the installation process. You will also need a RAM for i386 and at least 1GB. You will also need a CD-DVD Drive or a USB boot support. Once you have all the requirements, download Kali Linux, burn it to DVD and set your computer to boot from CD in the BIOS.

There are video and monitor adapter requirements that affect the Linux installation process. Linux is known to support all standard CGA, Hercules, VGA, and EGA monitors for the interface. You will likely use a mouse for the most part, but other Linux applications not linked to the graphics environment also use it. Start by booting the selected installation medium to start your installation. You will be greeted with a Kali Boot screen and give options to choose from. You can choose a text-mode or graphical install. Choose the language that you prefer and your location. You will be asked to configure the keyboard with a suitable keymap. Once you do that, you can indicate your geographic location.

The person who is installing Kali Linux will copy the image to your hard drive and ask for your network interface. Once you give the information, you will be asked to enter the hostname for the system. You can enter ‘kali’ as the hostname. You are allowed to use a domain name for the system to use. Apart from the hostname, you also need to give the name for a non-root use. The system will create a default user ID which can be changed easily.

Once you set the time, the installer will review your disk and give four options to choose from. For more granular configuration choices, experienced users can opt for manual partitioning. You can either choose to keep the files in one partition or separate them. Before the installer makes

final changes, you will have the final chance to analyze your disk configuration. You have some time to make changes before the installer locks you out. Once you click Continue, the installer will work on the disk, and the installation process will almost be over.

The good thing about Linux is that it is an open-source and free OS that can be used by anyone with programming skills. It can be modified and adapted to a new operating system based on a user's requirements. Furthermore, Linux is user-friendly and has several features that make it ideal. There is no need for antivirus with Linux. It is also reliable when used with servers and can run nonstop for years with the boot. It is low-maintenance, and this makes it convenient for all. Linux has several distributions including Redhat, Ubuntu, and Fedora and the installation process is the same for all. You can install Linux using CD-ROM. You can download the ISO files from the Internet and keep it in the CD-ROM.

Alternatively, you can make it bootable and store it in the USB stick. To boot it into the USB stick, you need to follow certain steps. After attaching CR-ROM into the computer, restart it, then press enter when boot starts. Proceed to select the CD-ROM to begin the booting process. You can opt for a manual boot by pressing F12 which will give you different boot options to choose from. You need to select one option from the list given. Keep in mind that you will see a new screen called GNU GRUB when the computer boots.

It is the boot loader responsible for the installations of Linux. It will appear when there is more than one OS. After booting into the USB stick, you need to select the derive. Choose the drive upon which the installation of the operating system will be done. If you want to replace the current OS, select 'erase the disk and install new one' but if you want something different, click the install number. Once you have determined where the installation process will be completed, start the process. You will be asked to confirm the installation by a small panel. If you are satisfied with the information provided, click continue. Choose your location on the map provided and start the installation process. Ensure that you give the login details. When the installation is done, you will be told to restart your computer to complete the installation process. To make operations easy, download drivers that you like in the system settings menu.

Some people prefer to install encrypted Kali Linux. Sometimes we have sensitive data that we do not want others to access. In such a scenario, we go the extra mile to protect the data, and this is where encrypted Kali Linux comes in. Thanks to Kali, you can now install an LVM encrypted install on a USB drive or Hard Disk. If you are not comfortable with one, you can use the other one. The good thing is that the installation process is similar to the regular Kali Linux install.

The only difference comes in picking an encrypted LVM partition when installing it. For an encrypted Kali Linux install, you need compatible computer hardware. The requirements are similar to normal Kali Linux install. Once you are set, download Kali Linux, and burn it in a DVD and set the computer to boot from the CD. The installation process is similar only that now you are dealing with an LVM encrypted Kali Linux install. Before you start, you need an install medium.

It is better to use a USB, and proceed to pick an image from the Kali download page. Ensure that you have an empty drive. Another thing to note is that it might take some time to write, so you need to be patient throughout the process. Once you get the USB, insert it into the computer you intend to install the Kali and boot it. Choose the USB drive as the booting device. Installing Kali Linux is not easy, but if you know Debian, it should not be problematic. After configuring the drives, Kali will start the installation process. Things may seem different if you are a Debian user. The installation process does not take long. Once it is done, you will be alerted on what to do next.

There are several reasons why you should install Kali Linux. Unlike other software, it is free of charge.

It also has more tools to pick fr and users are free to weigh their options. It is also multi-language support, enabling users to operate in their native language. While the while penetration tools are written in English, users have the liberty to use the language they are comfortable with. Moreover, it is easy to locate and use tools at any time. Kali Linux is popular because it is customizable.

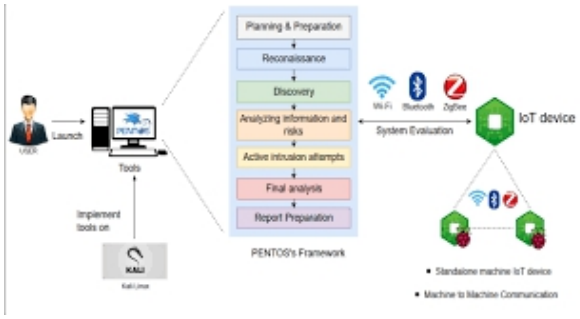
The developers know that not everybody accepts their design model and allows people to customize Kali Linux to their tastes. Kali Linux comes with a set of tools that makes it more appealing. It comes with an aircrack-ng, which is a set of tools used to get Wi-Fi network security. It deals with pertinent areas of Wi-Fi security, such as attacking, monitoring, cracking and testing. Monitoring refers to packet capturing and export of data to text files for extensive analysis. This is done using third-party tools. Testing is the process of checking Wi-Fi card abilities. Attacking is examining fake-access points, de-authentication, and replay attacks.

The tools are made in such a way that they allow heavy scripting. Another tool that comes with Kali Linux and why you should install it is Nmap. This is a network mapper that is an open-source for security auditing and network discovery. It employs raw IP packers in know hosts that are available on a network and types of services they offer. It also considers operating systems run by the available host. Nmap is also used to perform network inventory and monitoring hosts. THC Hydra is another tool that comes with Kali Linux. It is effective when in need of a force to crack an authentication service. It is also used to perform rapid dictionary attacks on telnets, SMB, HTTP, and FTP. It can be employed to crack web scanners or packet crafters. Nessus can also be used in a similar manner. It is a tool that is used to scan for computer vulnerabilities.

You need to do a couple of things after installing Kali Linux. While Kali Linux is efficient, it does not have everything required to run daily penetration testing. You can adopt some tricks and tips to start using Kali Linux with ease. The majority of Linux Distributions are customizable, which means that it is hectic to personalize the penetration testing distribution. However, you can do some things to improve your interaction with the OS .

Immediately the installation process ends, install Git, an open-source software control application. It is used to edit or share code. It is an important tool used for penetration testers who want to expand toolsets to surpass what is currently there in the Kali Repositories. It is also important to configure bash aliases after installing Kali Linux. These are great for creating customized commands. Users can choose to create new low privilege users. Never open or use certain applications as root users because

they can be problematic. Instead, install a terminal multiplexer to open terminal sessions easily.



Chapter 3:
Bash and Python Scripting

Bash and Python scripting are the most popular automation programming languages around the world. They have advantages and disadvantages users should keep in mind. It can be difficult to choose which one to use. To pick the right one, consider the task at hand, the context, and the ease or complexity of the task. Bash is a Linux shell command language and is wonderful for writing shell scripts that use CDL (Command Line Interface) utilities.

Bash is preferred by many because it can use command-line commands the way it is. Furthermore, it has better startup time compared to Python. However, it has poor execution time performance. Bash is not compatible with all shells such as zsh, and csh. It does not have good utilities and debugging tools compared to Python. It also lacks various functions, multi-threading, and data structures. On the other hand, Python performs general functions than Bash. It can be used for any task and has simple syntax. It has improved debugging tools than Bash. This makes it ideal for developing complex software apps.

Many times, people prefer to use graphical based interfaces than those from command lines. However, this is a wrong approach because the graphical user interface tends to be for those who believe that the software does what it is supposed to do, but that is not always the case.

It is far from the truth in the case of shell programming windows where malware and viruses cause big issues. Therefore, those who use the command-line interface think that what they are doing is correct. Other times you are stuck, and the computer hangs because of a simple graphic error. It is better to work on a command-line interface because it is fast, and you know what you are dealing with.

However, not everyone likes writing programs and using shells. Apart from writing programs, using some software would be worse than using a graphic user interface. Shell scripting is elegant to use, and it allows you to transfer the output of files. Shell scripting is fairly simple, but Python is more powerful. Its strength lies in the sed or the Stream Text Editor. You can combine shell and Python scripts to create and execute a chain of commands. Some people argue that Python can replace Bash shell. Both

Bash and Python are for general language purposes, and they have strengths and shortcomings.

Bash is a default terminal in Linux distributions and is faster regarding performance. However, that does not mean that it can replace Python. It gets complicated when dealing with large programs, but Python does not. Moreover, Python can be employed as an object-oriented language, and it is not a must to know the differences between the software. If you are searching for an elegant scripting language, Python is the one. However, this does not mean that Bash is less desirable. Bash is good at piping out commands. Shell scripting is seamless when handling files and is good at copying, storing file inputs, and cloning disks .

No matter how messy or cluttered your file system is, take delight in the fact that your computer comes with an alternate reality found in the command-line interface. It provides a direct way of interacting with your computer and acts as a relief from dragging and clicking small boxes that assume text-based commands. These commands tend to be mouse-agnostic and hard to work with. We use several small utility applications in the development of software. These applications lack graphical frontends and are not important. Instead, command-line interfaces are used to interact with these utilities. You can use utilities offered by the OS or a third party. Y

ou should care about the functionality for a couple of reasons. For starters, it gives insight into how the OS works. Furthermore, some routine tasks can only be done using the command prompt. There are also vital utilities that you can use in your everyday automation tasks. You need to install the Bash shell to facilitate smooth hacking. Bash has gained popularity over the years as a command line that is built into Linux and OSX.

It is currently offered as a built-in feature in Windows 10 versions. It is alluring to think that a command-shell is part of your operating system. Others believe that it is integrated into the computer without realizing that it is just a program. The operation of the shell is made in such a way that users enter commands from a set location and directs the OS on how to do that. If you launch a program by clicking an icon, you will get similar results. You can issue commands individually or through shell scripts.

Shell scripts are files with many commands organized in a manner that they perform complex functions. The command that a shell recognizes makes up a programming language, albeit narrow. Bash comprises of many features that are underused. There are diverse environments and work styles that affect hacking. One of the best things you can do to become a good hacker is to learn how to use Bash history. Learning how to use Bash history effectively is an effective trick that not many know or care about.

To increase chances of success, enable the histappend option to your shell. Enabling this option is easy by running this command: shopt -s histappend. Doing this facilitates several terminal sessions to write to history quickly. You find that this option is not enabled in most environments which leads to loss of histories when more than one Bash session is opened. Alternatively, you can use sudo to repeat the last command. For instance, if you want to create a new directory mkdir/etc/asp/ft.d, this command will fail if you are not the root.

Many users press the up arrow, scroll down and hit the sudo command. However, this is unnecessary work and there is an easy way of doing things. Run the command by typing: Sudo!! and Bash will run it and previous commands. It looks like this when you run the sequence: mkdir: cannot create the directory/ect/asp/ permission denied. It can also look like this: sudo mkdir -etc/asp/ft.d. when you key in !!, the full command is derived from the terminal and informs you what was executed. Another trick is to use the!* shortcut, which informs the Bash that you want to repeat the arguments from the previous command to the current one.

It comes in handy for commands with several arguments you want to repeat. To make work easy, create some files and change the permission on them. It is easy to change the permission on the files and becomes difficult only in some instances. You can search your history by pressing ctrl + r, and it will pop up. The good thing about Bash is that it does a reverse search on your history and when results appear immediately you start typing.

When running a shell, keep in mind that it is running inside a predefined destination in your file system. The system has a directory where the shell starts out. If you get lost, run the ‘pwd’ command to check which folder you are in. We use a ‘cd’ command to change directories, which is followed

by a valid destination directory. To run a program, type the name of the program and press enter. It will open fast and make you think you clicked it on the GUI.

Every hacker should make an effort to learn how to use a programming language to improve their skills. While it is okay to use available resources, you should grow and develop better tools to help you maneuver the web world. Because you are the one who knows what you want, you are in a position to get useful tools to help you reach your target. Technology has changed the world in unimaginable ways. Every day there are new hacks that are developed, and hackers are not slowing down. If you continue using old hacking tools, you will never reach where you want to be. If you launch malware on a machine that has improved its security system will not only stop the attack, but it could be traced back to you.

Another trick for navigating the dark web is to use quality and updated tools. The number of hackers has increased tremendously, and new tools are developed every day. As a hacker, you must know the current tools and how they are used. If you want to stay relevant in the hacking game and make more gains, know tools that people are using to stay ahead. The Internet is your friend, and there is no need to be caught off-guard. Moreover, some people offer free or affordable training on how to use such tools. Hacking is for people who understand the web works and staying informed. Do not make the mistake of thinking that you know everything because that is the beginning of the end. While you cannot trust just anyone, there are few trustworthy people you can work with to improve your hacking game.

Aspiring hackers cannot go far without having basic scripting skills. Lack of such skills will limit you to using tools developed by others, which lowers your chances of success. Moreover, it increases the chances of getting caught out by antivirus software. Companies know that hacking is on the rise, and they have adopted safety measures to protect themselves. There are intrusion detection systems that alert companies in case of intrusion. This is why it is important to learn python scripting skills. Having these skills elevates you to a professional hacker, increase chances of success, and minimize risks.

In fact, it eliminates risks! Python has features that make it important for hacking. It comes with built-in libraries with strong functionality. You should consider adding python modules because they provide a wide range of capabilities such as math modules, exception handling, and file handling. They also offer cryptographic services. Despite the advantages that come with having python, you still need to have third-party modules. There are extensive third-party modules for python, and that is why hackers choose it for scripting. In case you need to install a third-party module, you can download, and uncompress it and then run the python install command. After the download, you uncompress it using tar. For example, kali>tar-xf python-nmap-0.2 tar gz. After this, change the directory to a new one. Kali>cd python-nmap-03. You can install the new module by keying in: kali>python setup.py install. Now that you are familiar with how to install modules in Python, it is time to know basic terms and concepts.

Formatting is the first concept of Python, and it is important, unlike other scripting languages. Formatting is used by a Python interpreter to know how code is grouped. Consistency is more important than details. For example, if you have a group of code to start with double indentation, consistency matters for Python to know that the code belongs in the same group. In other formatting languages, formatting is optional, but here it is a must. Similar to any scripting and programming language, Python can add comments. It is important to add comments to inform the viewer of what you are talking about. Comments are sentences or words that explain the purpose of the code. While they are not necessary, it helps when you come back later and cannot recall what the script was talking about. The interpreter does not see the comments and lines with comments are skipped until it reaches a line of code. Similar to other languages, # is used at the start of a line as a comment. Another important Python concept is modules. Python allows users to group codes in modules. You must import a module in case you want to use it. Modules are the important features that give Python all its powers. They form the basis of the Python scripting language. There are also literal constants which require users to take texts for their literal values.

Ethical hackers have the habit of writing nifty scripts, and recently, many have opted for Python as the language of choice for such tasks. Python is used for ethical hacking for various reasons. Hacking started many years

ago and was the term was coined at the Railway Club of MIT. Hacking has evolved over the years to include sophisticated tools and tricks to maneuver the dark web world. Since many people are aware of data privacy and data protection, it has become a bit tricky to hack into systems .

Only those who are really professional at hacking and have been doing it for some time will be able to sneak onto a network without being noticed. Hacking is an illegal activity, and anyone found engaging in the practice risks a jail term, depending on the harm caused. Companies have embraced several measures to protect themselves from outside intrusion. Ethical hacking is now a norm in several companies. Ethical hackers have the duty of detecting and fixing security issues for companies before outside hackers spot them. Python is used in ethical hacking because it is easy to understand and use. Moreover, it is powerful and assures you of great results. It also has nifty Python libraries including NAPALM and Pulsar, which make the development of network tools easy. Most ethical hackers prefer developing small scripts and Python offers amazing performance. What is amazing about Python is that you get more than you bargained for. Learning Python opens doors for other opportunities you could not have known.

There are tips you can use to become a good hacker using Kali Linux. Hacking is just like any other activity, whether it is ethical and unethical. That said, you need to have certain skills to be an excellent hacker. You must be willing to learn new things continuously. A person who is a know it all or is rigid cannot be a good hacker. You cannot excel with Kali Linux if you are not open to learning different ways of doing things. It is okay to believe in your skills, but that does not mean that you brush off everything you hear .

Apart from having a learning attitude, you should have a good understanding of at least one of the coding languages. It is not a must for you to know every aspect of the coding language, but have a rough idea of what it entails is important. You need to learn how to code, understand the basic concepts of OS, markup many technologies, and fundamentals of security and network. It has been said without a number that you can only succeed using Bash and Python scripting if you understand what they entail. You also need to know the platform you will code in. This is very important because it gives a rough idea of what to expect and how to execute plans.

You should learn Python because it is popular owing to its portability. To become an expert at hacking, understand the operating system level of operations of a particular language you are using.

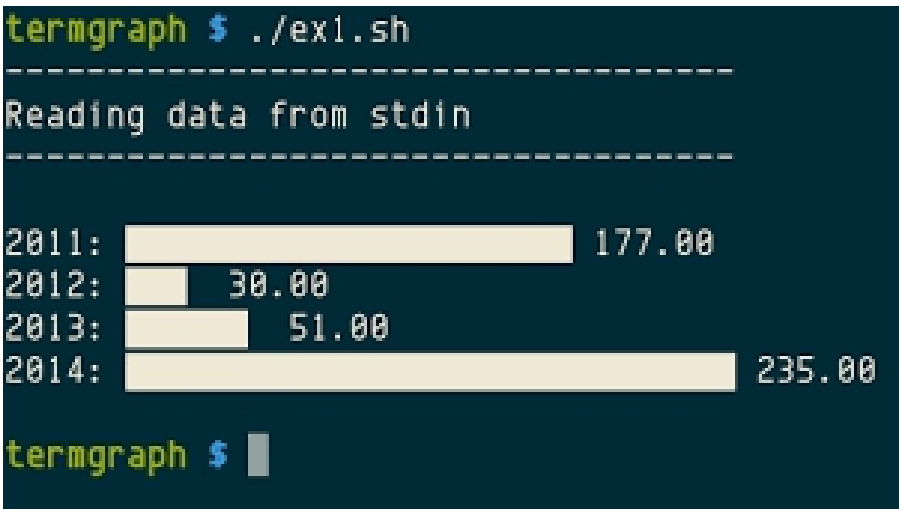
You also need to manage your expectations. There are many people who set high expectations the first day they start hacking and end up disappointed. They put unnecessary pressure on themselves, which hinders them from enjoying the process. Make things easy by managing your expectations and accepting the outcome. You might not manage to hack into a system right away, but appreciating the milestones you have made makes it easy to move on. It also gives you the motivation to try another time. Stay motivated for what will come next instead of stressing over what went wrong. Everyone has heard that they should never underestimate the power of system administrators and networks. They have the ability to make you a slave in the hacking world .

An application that resembles a free hoodie is not appealing. Any hacker wants to use an application that boosts their confidence and gives them the zeal to hack into systems. Let no one cheat you that how an application looks are not important because that is far from the truth. The truth is what you see is what you get. When an application is not pleasant to look at, the chances are that it is not that great to use. Hence, don't let anyone cheat you into using just any application, especially those that you are not sure of. Applications are similar to clothes; how they look determines whether you will buy them or not. There is a lot of different software that offers fancy and glitzy experiences, but all of us know that there is more to it than what meets the eye.

There are layers that users will never see. For example, TechOps must start the server and confirm that it is working properly. You may have a hoodie that is a universal cover and can be worn anytime. Similarly, the Bash shell is what keeps several software tools together. Apart from being universal, it is a recurring TechOps problem. The syntax may be obscure in some cases, and users get used to it. However, this does not make it okay. The aim of bash-bashing is to minimize the use of the shell. Since shell comes is now common, some examples can be given on how to translate shell scripts into Python. The shell can be used for many things such as reading configuration files, performing date arithmetic, managing files, and running

applications. They are also known to manage network resources with ease, hence their popularity. It allows remote access of resources using wget and curl. Managing network resources is easy using these programs.

Moreover, the script has vital steps that should be followed to increase the chances of success. It kills the process by reading the file to get process ID and then kill the commands. It also creates a directory to be used in the next steps. The shell script runs an analytic application and copies the output file in another location. Bash and Python Scripting play a big role in hacking with Kali Linux, and users must understand how they work, how and when to use them.



Different Types of Hacking

Many businesses and individuals use computers to do several things. Laptops and computers are essential for organizations, and this makes them an easy target. Hackers are classified into groups to enable easy identification. There are white hat hackers who engage in ethical hacking and do things by the book. They are allowed by organizations to compromise a system.

However, there are rules that they must follow. Black hat is the second category of hackers. These are the villains who compromise systems for personal gains. The third category is the grey hat hacker who hacks with a

good intention only that they do not ask for permission. People hack for different reasons. Others see it as a way to earn quick money, while others do it just for the sake. Some are lured by hacking series and TV shows. Hacking is considered cool by young people who see it as a way to challenge systems and disorient organizations.

Hacking refers to gaining unauthorized access to computer systems. The number of hackers has mushroomed over the years. They hack into systems by cracking passwords and gain access to systems. The term cracking is used to describe the method of getting the code or password to access a system. A hacker is a person who cracks into a system. It can be one person or a group of people hacking into a single system or a group of systems. A hacker can decide to hack a website, LAN network, or a social media account. To obtain access to a password, hackers crack algorithm programs. Hacking is not a new phenomenon, but the techniques change every day.

In attempts to tame the dark world wide web, governments spend billions, but that does not deter people from cracking into systems. Hackers do not just focus on major brands alone, but also small ones, provided they have vital information. A distributed denial of services (DDoS) attacks is on the rise. While it is not exactly a form of hacking, it is a major concern for online brands. It works by making services unavailable by flooding a website with fake requests or traffic. To pull off DDoS, the hacker uses a big portion of previously hacked computers to carry out the attack. The computer used is known as a botnet, and the hacker asks to access the targeted site many times to overwhelm the server and eventually bring it down.

Another hacking commonly done is injection attacks. This is where a hacker injects code into a site to carry out remote commands to modify the database. This type of attack is common because it can be done from easily accessible input points such as login forms or contact forms where the site permits public user input. These inputs are used to make SQL queries in attempts to interact with the site’s database and crucial access data. It also gives the hacker a chance to modify the database immediately. A hacker can also decide to carry out the Cross-Site Scripting (XSS) attack on a system.

This is a type of code injection, only that the attacker injects malicious code into a website then executes a malicious client-side script whenever the victims visit the website. For you to run a malicious script in the victim's web page, you must find a way to inject code into the said browser. You need to trail the victim and find the website that he/she visits frequently. Knowing where the victim visit regularly gives the hacker access to his/her cookies and enables him to send HTTP requests.

People hack into systems through DNS Spoofing. Using this method, the hacker manages to divert traffic from a server to a malicious one. It leads unsuspecting users to malicious web pages. For a hacker to spoof DNS, he brings in a corrupt domain name system data into the DNS cache. He then determines where the DNS requests go and steal sensitive information. After getting the needed information, the attacker redirects traffic. Hackers love this method because it allows them to divert traffic-legitimate browsers to malicious ones.

To gain unauthorized access into systems, hackers can practice clickjacking. This is where the attacker manipulates a website user's clicks by hiding hyperlinks under clickable content. This tactic allows hackers to trick web page users into clicking links without their knowledge. In this case, the hacker is not primarily concerned with hacking but causing the user to click on a web page. Clickjacking is done by concealing hyperlinks under something that the site user will be lured to click. For example, the attacker can place an attractive ad to motivate the surfer to click on the link.

It can be a social sharing button, and in case it is malicious, the attacker sends the user another website where a different attack is carried out. Google hacking is another way hackers use to gain access to web pages. It is where a hacker looks for sensitive information or victims through search engines such as Google. This is an easy way for them to find easy targets. The Google Hacking Database has a list of questions people search on Google. Attackers use these queries to identify sensitive information about the target or web pages. Google tried and blocked Google hacking queries, but this did not stop hackers from finding tools to crack websites.

Another great way to hack into systems is to use malware to compromise a computer system. To pull off attacks, attackers disguise malicious software

in video or music files to trick the victim into downloading and installing it. Trojan horse virus is an example of malware that is commonly used by hackers. A trojan is introduced with other downloads or emails that users trust. After download, they serve as the backdoor and contacts a remote controller which gains access into the system.

Hackers also use symbolic links to hack into systems easily. This is a method employed to hack Linux servers. It is a shortcut that surfers know and use regularly. Attackers use symbolic links to access servers' root directories. It makes hacking possible even when the user has limited access to the server. This method works by creating a symbolic link from the directory, where there is restricted permission to the directory. For instance, an employee with limited access has an easy time accessing the server using a symbolic link.

Once he has access to the root servers, it becomes easy to change files and to insert malicious code. The attacker can decide to expose the data or conceal it. Arbitrary code execution is another great way to hack into systems. This technique involves executing commands on target computers. Once the victim's computer is separated from the attacker's machine, a remote code executed is deemed to have taken place. It is done by assuming control of a program's instruction pointer, which indicates the next line of code to be processed.

A malware infection can be used to perform arbitrary code execution. Alternatively, an attacker can use fake wireless access points to infiltrate a system. If you have been keen, you have most likely witnessed a large number of open wireless access points. Hackers usually set up fake wireless access points (WAPs) to appeal to free Wi-Fi users. When you are connected to a Wi-Fi that is managed by a hacker, he/she can see everything that you are doing. It is scary how the hacker has access to all your information, immediately you connect to WAPs. It allows the hacker to see when you type the passwords and credit card information .

WAPs are set up in busy neighborhoods to lure many people. If you are not into the waiting game and want to carry out a quick attack on the system, you can use a brute force attack method. If you have tried every technique there is, and nothing seems to be working, you can become a brute force

attacker. In this technique, you try different encryption and passwords until you get the right answer. Seasoned attackers use brute force attack tools until they find what they are looking for. It is a method of trial and error.

The good thing with brute force attack is that many people use the same passwords for several accounts so once you have it, chances are you have access to everything you need. If a hacker manages to crack a Gmail password, chances of cracking the phone are extremely high. You can also use a directory traversal hacks to maneuver the hacking game. This method you access to commands, files, and directories that are outside a website's root directory. It works when the hacker keys in malicious character sequence into the search engine in a manner that the site executes the command.

How to be Secure and Anonymous while Hacking

It is essential for you to stay secure and anonymous while hacking. The truth is that we all hear about hackers from time to time. The sad part is that media outlets portray hackers as mythological creatures and not like humans. One of the most important things to learn before you start hacking is how to stay anonymous. If you do not have time to learn many things, at least make time to know how to stay secure and anonymous while hacking. Failure to secure yourself exposes you to all types of risks.

It is not hacking if anyone knows who and where you are. It is possible to trace everything on the Internet, and most of them are traced. It is safe to assume that someone is watching your activities online because that is the reality. There is no such thing as privacy on the Internet. Apart from the government monitoring online activities, your employer, parents, or guardians are also interested in what you are doing on the Internet. However, no one questions what you do unless you do something illegal or bad. The point is, hackers do illegal things, and that is why they take extra caution to protect their identity. They steal sensitive information, shut down services, and destroy data. You want to do things under the radar in case you are a hacker, and this is the sole reason why hackers remain anonymous.

The first secret of being secure and anonymous while hacking is to choose a good location. The perfect place to hack is a coffee shop. The coffee shop is the ideal place for hacking. A hacker cannot conduct his business in his house, and a place that can be traced back to them. This is disastrous and the same as hacking in the streets. Unless you have a secure location, don't attempt to hack anything. You will put yourself at risk and jeopardize future hacking activities.

Hackers know that everything they do can be traced back to them and that is why they go the extra mile to hide their identities. They pick a coffee shop because many people frequent such places and chances of getting caught and minimal. In case they discover what you are doing, they will go to the coffee shop, not your house. Moreover, it is virtually impossible to monitor people's activities in a coffee shop. However, going to a coffee shop is not enough to be secure and anonymous online. You must choose the coffee shop carefully and ensure that it does not have cameras. Some shops have spy cameras and make it hard to hide your online activities. Coffee shops are not equal, and your security online depends on the one you choose.

Pick a coffee shop that is managed by a few people who are focused on their jobs, not idling to see what you are doing. This creates a safe environment for hacking. Most coffee shops offer free Wi-Fi to customers or have a home Internet connection. You will have no logging while using these Internet services, and this explains why hackers carry out attacks at coffee shops. Avoid coffee shops that require you to register the Wi-Fi using your phone number or email. The secret to staying secure and anonymous online is to use a Wi-Fi that does not require your data. Pick a small and comfortable place. Ensure that you are the only person on a table. Some people are nosy and will distract you from the task at hand. Others cannot mind their business and snoop to see what you are doing. Avoid going to the same coffee shop twice and pick crowded cities. Remember that you are trying to be anonymous, so don't give people the chance to recall your face .

Another tip for staying secure and anonymous while hacking is to never open your email, Google Account, or Facebook. The Wi-Fi might seem secure, but they may have a way of tracking all the users. As a hacker, don't

ever get comfortable or feel you are safe and can do anything anywhere. Whatever you do, do not use Google Chrome because it is linked to your Google account and gives away your data.

Do not log into online services using your account. Ensure that you have disconnected from everything that can be traced back to you. To minimize the chances of getting caught, use Linux. While Windows is a good system, it was not meant for hacking. Linux was also not meant for that purpose, but it is easy to adapt it for hacking. You want complete control when trying to be secure online. You want to know what goes into the Internet and who has access to it. Linux allows you to pick a distribution that caters to your needs and makes it silent. Kali Linux comes in handy when figuring out how to stay anonymous online. It has all the tools that you need and makes hacking easy. A hacker cannot limit himself to using just Linux.

They can use live distribution because it does not require installation. Make sure that you increase your security, even if you delete cookies. Someone can still trace your steps even if you delete cookies after hacking. They may not be able to figure you out but can get a sense of what you were doing, and this is not good. Do not take risks and disable cookies .

You can stay anonymous while hacking by changing your MAC address. This is a unique identifier engraved in your network card. It is found almost in everything, including Wi-Fi. Its role is to let your PC talk over a network. It is not used to identify users, but it can perform that role. As a matter of fact, if someone gains access to your PC, he/she can know your MAC address. It can recall that you are the person who carried out the attack and will be discovered. The solution is to change your MAC address. At this moment, you are fairly secure, and someone has to follow you to a coffee shop or track you while hacking to bust you.

Another trick to stay secure and anonymous while hacking is to never reveal your movements or actions to anyone. Don’t trust even your closest friend because you don’t know if he can turn on you if he gets a good offer. Besides, people disagree, and some tend to overreact if they do not get their way. Someone may blackmail you if they know that you are a hacker. While some may be understanding, others may demand that you teach them how to hack and this will derail your hacking.

A smart hacker operates alone and never trusts anyone. When you start giving up your secrets, that is the moment you start preparing for your downfall. A smart hacker knows the importance of keeping secrets because you never know who you can trust. While it might be tempting to reveal to a close friend what you are doing and feel untouchable, don’t be fooled. The person you share such intimate secrets may cause your downfall. It is easy to stay anonymous online, provided you do your part and know when to act.

Chapter 4:
Ethical and Unethical Hacking

Ethical hackers are individuals who have permission to hack into a system. On the other hand, unethical hackers are people who gain access to unauthorized computer systems. Many people frown at the term “ethical hackers” without bothering to find out what it means. White hat hackers do this type of hacking under the instruction of an organization. White hat hackers are allowed to access a network to identify security risks. The hacker then advises the company in question accordingly. This is done to prevent black hat hackers from gaining access to the system.

Once white hat hackers notify the company of existing vulnerabilities, they can propose solutions or leave that part to the company. Their role is to identify risks. Many companies pay heavy prices to get white hat hackers to help them identify risks. It is a highly sought-after service in all industries. Some firms offer hackings services in the form of penetrating testing. Hacking is considered ethical when it is done to benefit a company or an individual. For example, one could be facing security risks and asks a hacker to look into the system. In such a case, the hacker is deemed to be doing a good job of helping the person identify and eliminate threats. Moreover, hacking is also ethical when it helps a state spot external security threats, which may be a minute of big .

The hacker determines the extent of the threat and advice the state accordingly. Therefore, hacking is considered ethical if it leads to more gain than harm. Companies do not mind spending thousands of dollars to avert security threats. Almost every company nowadays hires a white hacker to check their system. They know that the cost of tackling existing threats is costly and would rather prevent it from happening.

On the contrary, hacking is deemed unethical if it serves the personal gains of a hacker, commonly known as a black hacker. A black hat hacker does not look out for the interests of a company or a person but is only concerned about how he/she can benefit from the process. Such an individual can wreck-havoc in an organization because they are willing to do anything to get their way.

Therefore, hacking becomes unethical when it does not look at the broader picture. Hacking is done for myriad reasons, such as exposing corrupt individuals, bringing down fraudulent firms or shutting down services. Any

hacking that does not help others is unethical and is fought against by relevant authorities. The number of unethical hackers has increased over the years, with more people becoming interested in the dark web.

Hackers Hierarchy (Using and Abusing Services)

There is a hacker's hierarchy that informs the hacking world. The first level is the Script Kiddies, and many hackers fall under this category. A large number of hackers are bored teenagers who are looking for ways to have fun. Those who have access to computers delve into the world of hacking without prior knowledge of what it entails. Others have basic programming skills and feel that they are ready to start hacking. Essentially, script kiddies are amateurs and know nothing about hacking.

They use hacking programs developed by others and do not know how to develop their own. They depend on others and cannot make basic hacking decisions with confidence. We all know that a hacker must be confident to pull off risky parts of this work. Knowledge gives you power and confidence. An amateur lacks basic knowledge and question everything he/she does. Many people start as amateurs and progress to become renowned in the field. The only that is needed is patience to learn and grow. If someone tells you that they knew how to hack without depending on someone they are lying to you. No one was born knowing everything, and this also applies to hack. There is no shame in starting as an amateur.

It only becomes problematic if you do not move past that stage. One of the reasons why script amateurs never grow is because they only hack to show off to their peers. They hack for fun and do not have a plan or end goal. They find thrill in crashing systems, and that is the end of it. This level of hacking is less risky but a nuisance. While they do not pose big threats, other times, they get lucky and hit targets successfully .

The second level of hackers is the Hacking Group. They have more power than script kiddies and can cause great harm to organizations. The common hacking group is LulSec which launched attacks on Sony, and XFactor companies. They pose a big threat to the IT industry, and measures have been implemented to stop their activities. However, nothing much has changed, and hacking groups are still hacking companies.

The third level of hackers is hacktivists. These are people who hack with religious, political, or social goals in mind. Sometimes they hack into systems to post-religious messages or warn their adversaries. For example, a hacktivist hacker can hack a child-porn site to air his/her opposition. Such people are pushed by strong believes and want to change people's mindsets or viewpoints. Other times they hack into systems to spread pollical messages and support their candidates. This group of hackers is harmless because they mainly focus on pertinent issues in the society such as child-pornography, or trafficking. Hacktivism is on the rise in the security industry and government bodies. Police forces are mainly targeted by this group of hackers.

The fourth level of hackers is Black Hat Professionals. No one poses more threat to the IT world that black hat hackers. They have unmatched skills that allow them to get away with any hacking activity they participate in. What sets black hat professionals from other hackers is the fact that they take time to learn to hack. They are not ordinary hackers you meet or hear about on the news. These are seasoned hackers who pass for ordinary citizens. When you meet a black hat professional you would think that they work across the streets.

They have mastered the tricks of hacking, and it is hectic to figure them out. It takes equally good hackers to discover what they are doing and even so, it is still hard to bring them down. A black hat profession knows the loopholes to look out for and strikes at the right time. He does not get involved in activities that might compromise his identity and chooses his companions wisely. One of the things that bring down seasoned hackers is trusting the wrong people. It can be tempting to boast to others about your achievement in the hacking world, but a black hat professional is not that careless. He/she knows the value of silence in the dark web.

They only open up when talking to fellow hackers or someone that feel will never betray them. Even so, they prefer listening over sharing their experiences. They have coding skills and are determined to gain access to unauthorized services. It might be difficult at first, but a black hat profession never loses hope. After all, no one said that it would be easy. Instead of comparing their gains to others, black hat professionals keep their eyes on the prize. They do not hack to destroy information but to look

for new ways of accessing data that restricted. They find pleasure in getting access to impenetrable targets.

The fifth level of hacking is organized, criminal gangs. As the name suggests, organized criminal gangs are led by criminals. They comprise of professional criminals who lead a group of people to hack computer systems. They attract code crafters most of the time. They work under strict rules and members who disregard them face tough penalties. They operate under rules to protect their operations from scrutiny. They can be hired by people who want to hack into systems, or they can organize their operations.

To work in an organized criminal gang, you must meet set expectations and work under supervision. Only those who follow instructions stay in the gang for a long time. It can be risky working in a group in case something goes wrong. For example, if one is caught, he/she may rat out the entire gang and spoil the operation. The good thing with organized criminal gangs is that there are many people to work with or consult if something is unclear.

Moreover, you are likely to get hacking gigs in a gang than on your own. Also, there is comfort in working in a gang, compared to being alone and taking care of all the risks that come with hacking. However, organized criminal gangs face many challenges and are dangerous if you trust the wrong person. Hacking becomes scary ones you add another person to the equation, let alone a group of people. Organized criminal gangs have leaders who create rules, add and remove members. They are also in charge of discipline in the group and making payments.

The sixth level of hacker's hierarchy is nation-states. The nation-state is on top of the hacking ladder. It is a common belief that with state nations, there are trained cybercriminals who work for the government. This means that every state has a group of hackers who hack into systems. While this is common knowledge, no state has ever admitted to it. It is no secret that every state wants to know what the other one is doing for economic, political, or social gains. It is no wonder they go to great lengths to train cybercriminals to get them sensitive information about other states. Nation-states have at their disposal crack squads to find out what the other state is

doing. More nations have realized the need to work with hackers not only to protect themselves but also to know more about other states. It gives them political leverage and informs them where they stand on the economic scale. Hackers have enabled countries to know the economic positions of countries they want to trade without them divulging such information. Nation-state hackers are emerging trends that people are on the lookout for.

Level seven is the use of an automated tool. It is important for a computer user to know the six levels of hackers, but they do not meet the standards because they do not address automated tools that cause great harm to people and businesses. They cause great harm at a little cost. Some technologies and computer applications have gained popularity more than others over the last 20 years. Hence, slight weakness in an application can cause many computers around the world to be vulnerable to exploitation. Automated tools are pieces of software that play the role of a worm virus and affect many computers to have a big framework. Companies have put in measures to protect themselves from risks, but still, face many risks. Others have educated employees and warned them against clicking some sites, but hackers always find a way to lure them into the bait.

Chapter 5:
Servers and Networks

The US introduced the first computer network in 1996. It was a big deal because it was the first of its kind and created new ways of connecting computers. It also lay the foundation for the development of the Internet. Previously, computer networks were limited and performed limited functions. Today they have been developed to include a wide range of devices. The purpose of servers is to ensure that information flows between computer networks.

A server can be a system or computer that gives data to other computers through networks. Other computers that servers share information are called clients. There are different types of servers, including virtual servers, mail servers, and web servers. One system has the capacity to give data and use them across systems in the same period. This implies that a device can act as a client and server concurrently. The first servers were different from what is in the market today. They were minicomputers or mainframe computers and were smaller in size. When technology was advanced, they became larger than desktops.

They were no longer minicomputers but microcomputers. During the initial stages, people connected servers to terminals, types of servers, that did not do a good job. The role of the terminals was to receive input through a keyboard and return the outcome to a printer. Its limited role affected the overall function of devices. Computing was done on servers. Things changed over time, and servers could be connected to single and powerful computers over a network to less-powerful client computers.

The situation where both the server and the client computer have the computing powers is known as the client-server model. However, some tasks are assigned to the server in the client-server model. Previously, the mainframe-terminal model acted as a server although it was not known by that name. The meaning and use of a server have evolved as technology progressed.

It can now perform more tasks than it did in the beginning. It has reached an extent where servers can be software on computer devices. Servers who are installed as software are known as virtual servers. Initially, virtual servers were employed to improve the number of server functions. It has progressed to become a multi-purpose software. Virtual servers can now be

run by a third party on hardware through the Internet. Cloud computing comes in handy when dealing with virtual servers. Servers can be created to perform single or multiple tasks. A mail server is an example of a single task performed by a server. Print and file servers are examples of several tasks performed by servers.

Devices are configured to listen to requests from clients to work as a server. It must also be connected to a network and can be installed as an application. If you use Microsoft's Windows server, you get to listen and respond to the client's request. It is also suitable because it sends requests to clients over the network. It receives and responds to requests with the right information. Some applications are notorious for responding with the wrong information, thereby affecting the entire process. This server is known as the call and response model. Servers perform several tasks such as checking the identity of the person who sent a request and ensuring that the client is allowed to access information.

Different servers perform different functions. Networks have more than one type of server. Types of servers include file servers, print servers, application servers, mail servers, DNS servers, and web servers. A web server is one of the most abundant types of servers. It is a type of server that hosts data requests and programs across the Internet. They respond to client requests from the browser operating on a client computer for websites. Examples of web servers include Microsoft Information Services, Apache Web Servers, and Nginx Servers. Mail servers are used as application servers. They are common and used by many.

The work of a mail server is to receive emails sent to users and keep them until they are requested by the client. Therefore, it is the duty of the mail server to store emails and keep them in good condition until they are demanded by the user. A mail server enables the configuration of a single machine and attaches it to a network at all times. Once the configuration process is done, it is ready to send and receive messages instead of expecting every client machine to have email subsystems running all the time. DNS (Domain Name System) Servers provide name resolutions to client computers by changing names that humans understand easily into machine-readable IP addresses. It comes in handy because it helps when the client wants the address of the system.

Another type of server is proxy servers. It is an intermediary between a server and a client. It is commonly used to isolate the client servers for security reasons. The proxy server receives the feedback from the second serve application and replies to the first client. Servers use client servers to run applications. They are used to run intensive applications shared by many users. It eliminates the need for every client to have enough resources to run applications. It also removes the burden to find and run applications. Users are also free from the burden of installing and maintaining software on many computers. Print servers facilitate the management and distribution of printing functions. Instead of attaching printers in all workstations, one print is enough to do all the work. It has the ability to respond to printing requests from several clients.

Moreover, some high-end printers come with built-in servers that cater for the cost of printing services. It removes the burden of buying computer-based print servers. Another type of server is the file server, which stores and distributes files. Many clients can share files kept on a server. Furthermore, storing files offer backup in case users lose sensitive information. The server hardware can be built in such a way that they maximize read and write speeds to promote efficiency. Virtual servers are not here to play. They are taking over the world, and users have started noticing. Virtual servers are defined in specialized software known as a hypervisor. Hypervisors run on thousands of virtual servers at once. Virtual servers are considered the best servers the world has ever seen. The hypervisor gives virtual hardware to the server in a similar manner to physical hardware. It uses the virtual hardware and passes the computation to the actual hardware underneath that is shared by other virtual servers.

Computer networks are put into three categories based on the size, the structure, and distance. The categories include LAN, WAN, and MAN. Two devices are considered to be in-network if the process in one can share information with another. They are also referred to as mediums of connections between nodes or machines. Networks comprise of computer systems, networking devices, and servers that are linked together to exchange information. Wireless media can be used to connect computer systems. Types of computer networks are discussed below.

- LAN (Local Area Network)

LAN is a privately own computer network that covers a small geographical area of networks such as offices, homes, schools. It is used to connect computing devices to enable the exchange of information. Devices connected in LAN can share a wide range of information and build strong connections. LAN is small in size compared to other types of networks and devices are connected to central devices known as Hub via a cable. Today, wireless technologies can be used to install LANs and uses access points to send and receive data. In some cases, one computer acts as a server and caters to all the computers called Clients.

For instance, libraries can have wired LAN networks for users to connect local networking gadgets. They can connect servers and printers to connect to the Intranet or Internet. There are two types of LANs, namely Ethernet and ARCnet. ARCnet is the simplest, oldest, and most affordable type of LAN. It was commonly available for microcomputers and gained popularity in the 1980s. The good this with ARCnet is that it allows different kinds of transmission media to be part of the same network. Ethernet is the second type of LAN, and it is a family of computers that networking techs for LAN. It was introduced in 1980 and grew to replace wired local network techs. It uses a star or bus topology network. It also supports the transfer of data at the rates of 10 Mbps. It uses the SCMA/CD access technique to tackle demands that come at the same time. It is the most implemented LAN standard in history. 100Base-T is the new version of the Ethernet network.

- MAN (Metropolitan Area Networks)

MAN is bigger than LAN and occupies the area of one city. It rare for them to go beyond 100 KM and consists of a mixture of transmission and hardware media. MAN can an individual network like a cable TV network. It is based on the IEE 802 standard method called DQDB. It employs two unidirectional cables and connects all the machines to them. Standardization and security are the two most important aspects of the Metropolitan Area Network. Security plays a vital role because data is being shared between dissimilar systems. In such cases, it is important to put everything in order and prioritize security. Standardization is crucial because it ensures reliable data communication. Users must ensure that the two aspects are taken care of. MAN interconnects many LAN using via

high capacity links and avail link services to the Internet. Protocols are found at the data link level that is explained by ITU-T, or IEEE.

- WAN (Wide Area Network)

WAN serves as a telecommunication network is a network of networks. It connects LANs that are on opposite sides of structures and across the world. They are known to have the slowest data communication rate. Moreover, they have the largest distance. There are two types of WANs, namely, Global WAN and an enterprise WAN. WANs are typically connected to public networks, satellites or leased lines. The Internet is the largest WAN in existence. Certain parts of the Internet, such as VPN are considered to be WANs. Several WANs have been created including banking networks, corporate networks, and stock brokerage networks. Firms that support WANs through the Internet Protocol are called Network Service Providers and forms the basis of the Internet. WANs use costly networking equipment compared to MAN or LAN. They use Frame Relays, ATM, and SONET. An enterprise WAN is preferred because it connects the whole firm, including LAN sites. The term is used by large organizations, governments or universities.

On the other hand, Global WANs also reach the whole world, but it is not a must for them to connect to LANs in a firm. An example of a Global WAN is the Internet because it connects different areas, institutions, and organizations across the world. Global WANs can either be public or private, where the latter is called Intranet that belongs to a firm. The former is open to everyone, and people have access to resources.

- WLANs- Wireless Local Area Network

WLANs are also known as LAWN, which stands for Local Area Wireless Network. It gives wireless network communication over a short distance. It uses infrared or radio signals to provide wireless networks instead of traditional network cables. WLANs provide a framework for users to connect to local area networks via wireless connections. It extends to a wired local area network and is developed by attaching a device known as the access point to a wired network. Users communicate with the access point through a wireless network adapter that works similar to a traditional

Ethernet adapter. One of the most important issues for WLANs is network security .

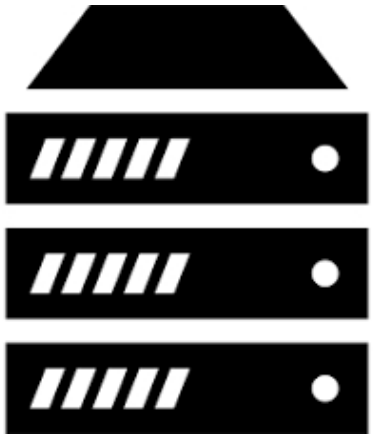
To ensure security, Random wireless clients are denied access to WLAN. Components that have the ability to connect into a wireless medium are referred to as stations. The stations have wireless network interface controllers. There are two types of WLAN, Private home and Enterprise-class WLAN. Enterprise-class WLAN uses several access points to broadcast signals to wide areas. These access points have many features apart from small offices or homes. A private home WLAN employs one or two access points to broadcast a signal. Most retail stores have tools for installing home or small office WLAN. Another type of network is SAN (Storage Area Network). It is a kind of LAN with high speed and special use networks. It facilitates data storage and replication on business networks via servers. SANs technology resembles network-attached storage.



Technology server icon



Virtual private servers



Professional servers

Chapter 6:
Cyber-attacks and Malware

A cyberattack is a concerted effort by an individual or a group of people to breach the information system belonging to another person/people. The attacker is usually after some benefits from infiltrating the victim's network. The rate of cyberattacks has skyrocketed over the years, and officials are worried. Despite measures being implemented to thwart similar attacks, this has not prevented new attackers from joining the dark web. Businesses are affected by cyberattacks every day, and there is no guarantee that they can stop for good. Cybercrime is on the rise because people want to take advantage of vulnerable business systems.

Most of the time, attackers are after ransom money and other times they do it for fun. Others attack systems to retaliate against past treatment. Cyberthreats are also being launched with wrong motives as some attackers want to render systems useless to show that they are hacktivists. A network that is infected by malicious software is known as a botnet. Attackers have the ability to control botnet without its owner knowing what is happening. They do this to increase the severity of their attacks. Most of the time, a botnet is used to overwhelm systems by launching distributed denial of services (DDoS).

There are different types of cyberattacks, and one of them is malware. Malware is used to describe malicious software such as ransomware, spyware, worms, and viruses. Malware takes advantage of a vulnerability to launch an attack. It does this by sending malicious sites to victim's pages, and when they click, they are exposed to attacks. Users can also click on dangerous email attachments which they proceed to install into their systems. Once dangerous software is installed in a system, the attacker gets an opportunity to launch attacks. Malware can do several things once it is inside the computer system. It can block access to crucial components of the network, get information by transferring data from the hard drive. It can also install more deadly software to harm the system further. Malware can interfere with some components and make the system useless. Once the attacker has infiltrated the system, he can do virtually anything he wants.

There are different types of malware, but the most destructive and common one is ransomware. It is meant to freeze files and demand ransom. The number of cyber-attackers has increased with more opting to engage in ransomware because of what they stand to gain. The thing that attackers do

is steal sensitive information and place huge ransom for it. In turn, organizations pay heavily in exchange for the data. Attackers figured that they could make more from stealing information and demanding a ransom. Today, even after organizations pay huge sums of money to recover data, they are not let off the hook. Attackers continue demanding more money, while others do not return data after receiving payment. It becomes a cycle of making a payment without getting the information. Moreover, some attackers copy data in another folder and lie to victims that they have not kept or shared the data with anyone else. They use the information as leverage to demand more payment.

Another type of cyberattack is phishing. Phishing is the act of sending dubious communication that seems to have come from a reputable source. They are usually sent to email to convince the victim that they are valid. Attackers use phishing techniques to steal sensitive information from victims, such as credit card information. They can also do this to install malware on the victim's system. Few people can know when a phishing technique is being used, and the majority fall for the trap. Another common strategy attacker uses to obtain is information is Man in the Middle (MiM) where they position themselves in the middle of a two-party transaction. The attacker waits until he interrupts the traffic to filter and steal sensitive information. There are two entry points for MiM attacks. When malware has breached a system, the attacker can install the software to obtain access to the victim's information. Another way is to use unsecure points in a system.

Another type of cyberattack is injection attacks, where data is injected into a web application cheat the application and steal the right information. For example, attackers can use log injection, SQL Injection, or code injection to manipulate applications. DNS Spoofing is becoming a common method of a cyberattack. It is a kind of computer security hacking where data is put in a DNS revolver's cache, making the name server to return the invalid IP address.

Some attackers use brute force to steal information from clients. This is a trial and error strategy where the attacker is not completely sure of what he is doing but does it anyway. This kind of attack results in a lot of guesses which attackers use to get data such as user password and bank details.

Criminals are the ones who use this method to crack encrypted data. Alternatively, attackers can use session hijacking to steal information. It is an attack on user protected sessions. Attackers steal cookies that store user sessions to acquire data. Some users disable cookies, but the majority do not know its significance. A denial of service attack is where services are made unavailable to users. This is done by causing traffic to the target or triggering a crash. The single system is used to cause a denial of service. Another type of cyberattack is a dictionary attack, where a list of commonly used passwords is checked to get the original password.

There are system-based attacks that expose users to risks. The first one is a virus, a malicious software program sent to computer files without the user's knowledge. It multiplies by inserting its copies in other systems when executed. Worms can also be used to carry out attacks. A worm is a type of malware which replicates itself so that it can spread to other systems. It works in a similar way like a virus and comes from email attachments that fool victims into thinking that they come from trusted sources. Backdoor is a method that uses scrupulous tactics to bypass the verification process. A backdoor can be created to enable access to an application. Attackers also use the Trojan horse tactic to access systems. This is a malicious program that causes unexpected changes to computer settings. It cheats users of its true intention and looks like a normal application, but when it is opened, malicious codes run in the system. Some attackers prefer to use Bots to infiltrate into systems. This is an automated process that interacts with other services, and examples include malicious bots, chatroom bots, and crawlers.

Companies have realized that paying attackers will not make the problem go away. If anything, it makes things worse. Other types of malware include Adware, Rootkits, Bots, Spyware, Viruses, and Remote Access Tool. Adware is a software that not only downloads but also displays unwanted ads that enable attackers to collect data without users' knowledge. Many attackers are using Adware because it allows them to carry out attacks without being detected. It is also used to redirect searches to specific web pages. Organizations that are having low traffic on their pages can hire attackers to redirect users to their websites. This is a popular strategy used to cheat users into visiting unwanted web pages. Some use it to draw customers from visiting rival's websites. Bots is an automated script that takes control of a computer. It enables attackers to launch attacks

on the Internet. Spyware is a software used by attackers to steal information. It works by sending data from the hard drive without the user's knowledge. The number of people who use spyware has increased over time because of its accessibility and ease of use.

Rootkits are software meant to conceal the fact that a system is infiltrated by changing vital executables. They allow malware to stay in the open by copying normal files. It is crucial to understand how malware gets into the system. Cyber-attackers have found clever ways to maneuver security solutions and such as firewalls and antivirus. They know that humans are unpredictable and use weak links against them. One such way malware gets into systems is through phishing, where corrupted links are sent to users. Because users have been warned against clicking suspicious links, attackers have adopted clever ways tactics of getting into systems. They use malicious ads to get victims. Nowadays, attackers buy ads from the Internet and inject malicious codes into them. What makes these ads dangerous is the fact that it is not a must for the victim to click them for the system to be infected with malware. However, this method requires skills, and most attackers lack the patience and willpower to learn. However, some attackers are learning this new technique of infecting systems with malware, and it is proving deadly.

Once malware is in a system, its goal is to communicate to the attacker who sent it. It does not work alone but receives orders from command and control servers. These servers are hidden on the Internet, and users have no way of spotting them. Many attackers use DNS to get their malware map a domain to an IP location which helps to manage victims. It also allows easy access even when the user adds or removes features. The servers are the ones that command malware to steal information, spread risk to other computers, capture keystrokes, and enable the camera. It also gives the command to erase avoid detection. Attackers have mastered the skills, and it has become hectic, if not impossible, to catch them. When an attacker erases the server after attacking a system, it is difficult to know an attack happened.

Cybersecurity refers to the process of protecting networks, programs, and devices from attacks. Cyberattacks are on the rise, and organizations are taking steps to protect themselves. The best defense against cyberattacks is

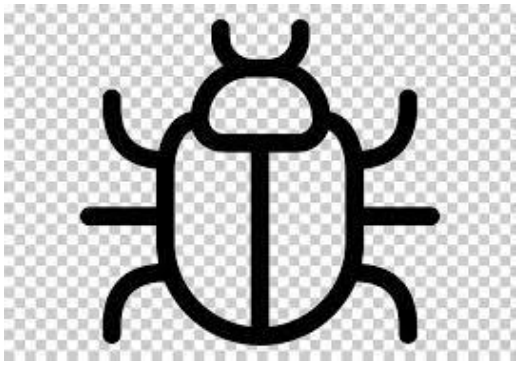
to have strong cybersecurity and layer protections across systems. For a cybersecurity program to work, it needs strong cyber defense decisions and technology. The good news that you do not need to be a security specialist to protect yourself from attacks. There are various types of cyberthreats that threaten devices each day, and they fall into three groups, integrity, confidentiality, and availability. Attacks on confidentiality include things like stealing identity information and bank details. Attackers usually steal confidential information and sell them on the dark web.

They are then used by people who are far away. Attacks on availability occur when attackers block users from accessing their information until they pay a ransom. Attackers infiltrate the victim's network and block him from accessing the site. Organizations sometimes pay the ransom and address the security issue afterward to avoid vulnerability. Attacks on integrity occur when personal information is leaked and influence people to lose trust in you. Cybersecurity has evolved over the years to include sophisticated strategies for ensuring safety. Businesses protect themselves on the Internet by staying informed and be careful about what they share online. There is a reason why they say the Internet is forever. Companies that did not see the need for cybersecurity are now employing security specialists to address their areas of concern and ensure safety. There is no strategy guarantees safety on the Internet, but implementing security measures such as using antivirus and firewall goes a long way in curbing attacks. More and more people are implementing security measures.

VPN stands for Virtual Private Network and is a type of service that protects user privacy while on the Internet. It does this by encrypting online traffic to safeguard sensitive information from attackers when you are on the Internet. Organizations can use VPNs to protect themselves from hackers. It also hides the IP address and protects your identity online. The Internet is a dangerous space, and you must take measures into your own hands to protect yourself. People or companies that use VPN enjoy online anonymity and helps them to bypass limitations set to prevent people from accessing certain web pages. On the other hand, a firewall is a hardware or software that monitors incoming and outgoing network traffic.

It can be employed to safeguard computers by blocking access to unsafe websites. It also works by denying some programs connected to a web

page. The role of a firewall is to ascertain that malicious files and unauthorized users have no access to systems connected to the web. Companies that this step to protect themselves from outside intrusion. Firewalls can also be used to block many websites and online services. Whether a firm chooses to employ a VPN or a virus is entirely up to them. The most important thing is that they are safe from attacks.



Malware icon



No malware icon



Laptops infected with malware



Malware explained



Malware icon

Chapter 7:
Cryptography

Have you ever wondered how passwords are stored securely? Or how credit card information remains private when buying goods online? The answer to those questions is cryptography. The rate of cyberattacks has increased over the years, and online companies are not taking chances. Most Internet sites have incorporated cryptography to protect sensitive information about clients. Cryptography helps them provide security to users.

Certain information can appear less important but can be used by attackers to cause great harm. For example, information about an email account is encrypted to prevent misuse. By now, you are wondering what cryptography is. Well, cryptography is the process of transmitting information safely against party access. Many companies swear by cryptography because it helps them avoid losing information or third-party interference. People jam into conversations for several reasons. Some do it to steal information, destroy other people's reputation, or for financial gain.

Whatever the reason, cryptography help individuals and organizations evade risks and secure data. There are encryption algorithms used in the encryption process. The first one is symmetric key encryption, which uses one common key to lock and unlock the encryption bar. It gives both the sender and receiver a similar key. The good thing with symmetric key algorithms is that they are fast because it is not a must for keys to belong. However, there is a risk when sharing the key because a third-party can intercept it, and this can compromise the entire system. The fear of interception is what causes some to shy away from using the symmetric key encryption.

The other type of encryption algorithm is asymmetric key encryption. In this algorithm, only the receiver has the key and can choose to send a lock known as the public key. Asymmetric key encryption works by the receiver generating two public keys n and e, and one private key d. He then chooses two large prime numbers q & p, such that n=q*p.
Another crucial aspect of encryption is signing messages which enable you to verify senders and avoid sending data to the wrong person. You can sign a message by creating the signature M, such as T= m^d mod n and then send T together with the message. Always remember than d is the private key. Cryptography deals with safeguarding data and communication strategies. It is the concept of preventing information and communication

from unwanted access by embracing codes with the intention of creating a safe platform. It is an application to help organizations and individuals realize secure and safe communication and information processes from third-party access.

Advantages of Cryptography

When encryption services entered the market, they were not capable enough to meet security needs in the market. The rate of intrusion was high, and people did not know how to protect themselves. They implemented tough security measures, but nothing was working. It was until the new and improved encryption services were introduced that people felt save. It not only protected them from third-party interference but gave them confidence when communicating or transacting online.

Previously, people were cautious about what they said online because they did not know who was listening. Some chose not to share financial information or purchase goods online to be saved. The new encryption restored users' faith in online transactions. It meant that they could talk about anything without being intercepted. When companies saw the rise and benefits of encryption, they implemented it.

- Privacy- Encryption guarantees users' privacy online. Data can be accessed by those it was intended and third parties. Protecting sensitive data is the best way of ensuring that even if the data reaches an unintended person, it will still be secure. Encryption enables you to remain anonymous and mitigate opportunities provided by criminals to decode sensitive information. You do not have to worry about hackers with encrypted data.
- Maintains integrity- As a business owner, integrity is an important element to growth. Without notice of suspicious activities, information cannot be changed. When a hacker is able to identify sensitive information, he can hack it and use it to commit fraud. With encryption, you can modify sensitive data which helps to maintain integrity. It also enables a firm to respond swiftly to cyber-crime.
- Protects authentication and data across devices- cryptography enables people to identify the sender and the receiver and the origin or destination of the information. Mobile phones are essential to humans and help them receive and send information. Moreover, mobile phones have storage features that enable them to keep information for long periods. It also allows users to verify the source of data and to detect unauthorized senders.

Types of Cryptography

There are different types of cryptography that users should be informed about. Cryptography has evolved over the years to become polished and effective. It is applied in different industries such as e-commerce, digital currencies, and law enforcement agencies. There are different types of cryptography algorithms analyzed in this section.

- The first one is secret-key cryptography- this type of cryptography is primarily used to maintain security and privacy. It employs one key for encryption and decryption. It is also known as symmetric-key encryption.
- Another type of cryptography is public-key cryptography, also known as asymmetric encryption. It is used for authentication purposes. A hash function is the other type of encryption which is mostly used for message integrity. It utilizes a mathematic algorithm to encrypt messages, and SHA-1 is an example of a hash algorithm. Public key cryptography is said to be the most important development in cryptography. Martin Hellman, Stanford University professor, was the first to describe Public-key cryptography. He stated that it depends on the availability of one key. If you are using multiplication versus factorization, for example, if you have two prime numbers 5 and 7, and want to calculate the product, it should not take long to derive the value, which is 35. Now assume that you have a number that is a product of two primes, 35 and want to know those primes. You will eventually come up with the numbers, but it will take longer than it took to get the value of the two prime numbers. The problem is magnified if we use prime numbers that have 200 digits or so because the end product will have -400. While this example seems trivial, it represents functional pairs used with public-key encryption.

Generic public-key encryption uses two keys that are connected, but knowledge of one key does not enable one to know the other key. One key is employed to encrypt plaintext while the other is used to decrypt it. It does not matter which key is used first, but that both keys are needed for it to work. In public-key encryption, one key is considered the public key, and the owner can advertise it if he wants to. The other key is the private key and remains private. For example, if Winnie wants to send a message to Paul, she will encrypt some information using Paul’s public key. Paul will decipher the message using his private key, and this technique can be used to determine who sent the message. When Paul decrypts the message using Winnie’s public key, he will know that she sent the message and she cannot deny that fact.

Another name for Hash functions is one-way encryption. It uses no key and has a fixed-length hash on the plaintext that makes it difficult to recover the length of the plaintext. Hash algorithms are employed to get the digital fingerprint of files and make sure that it is free from viruses. They are also

used by operating systems to encrypt passwords. Therefore, it provides a strategy to guarantee the safety and integrity of files. Since hash functions are one-way, it is impossible to decrypt files. Some websites claim to decipher such files, but the only thing that they do is to find suitable strings that produce the hash. Hash algorithms commonly used today include Message Digest (MD) algorithms, MD2, and MD4.

Extensions of hash algorithms are applied for various information security, such as has libraries, rolling hashes, and fuzzy hashes. Rolling ashes refer to hash values computed based on a fixed length of the gliding window via the input. It might be computed bytes 1-10 of one file. Hash libraries are sets of hash values that correspond to known files. It has the hash values of files that are part of a certain operating system. For instance, it could be part of a known file set and might be overlooked in an investigation for malware. Fuzzy hashes form part of intense research and stand for hash values that represent two inputs of similar values.

These three encryption techniques are important for a couple of reasons. Every technique is optimized for certain cryptographic applications. For example, hash algorithms are ideal for ensuring data integrity because changes made to the content leads to the receiver calculating another hash value than the one given by the sender. Because it is unlikely that the same message will give two different hash values, hash functions ensure data integrity. Secret key encryption is perfect for encrypting messages, provides confidentiality and privacy. The sender is in a position to generate a session key to encrypt the message. The receiver needs that key to decode the message. Asymmetric algorithms can be employed in non-repudiation and user-authentication. In case the sender can get the session key encrypted with the private key, only he could have sent the message. Public key encryption can also be used to encrypt messages, but it seldom happens because secret key values can be computed up to 100 times faster than cryptographic values. Size does not matter in cryptography. When the key is large, it is hard to crack a block of encrypted data.

This is because large keys give more protection than is noticed. Moreover, computers have made it possible to attack ciphertext through brute force instead of aiming at mathematics. In the past, the brute force could not be used in computers, but things have changed. While cryptography is good,

long keys can interfere with the nature of data files. The US. The government controls tightly the export of crypto products. Safe use of cryptography needs trust, and lack of it can jeopardize the entire process. While hash codes can ensure integrity and secret key cryptography can ensure security and confidentiality, they cannot work without trust .

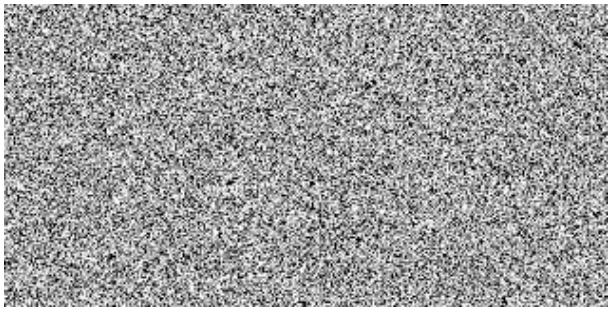
To help you understand the concept of encryption, let us use an example, suppose a person B sends a text to friend C who stays in a different state through a public platform. However, someone named D hacks the security of the communication and redirects the message from B to C. This is where cryptography comes in. B uses a key to encrypt the message and sends, but this time person C cannot decode the message because it is encrypted. Receiver C will be given a decryption key to use once he receives the message. Even if D manages to hack the system midway and changes the message, C will get error information when he attempts to decrypt it. This way, he will know that the message was intercepted and is unreliable. People who do not encrypt messages face the risk of getting the wrong message and never knowing about it.

Cryptography is used to protect the integrity and authenticity of messages. The idea of having electronic keys in a communication network is to make sure that the recipient gets the right message without interference. Through the help of a secure system, the recipient can decode the message using cryptographic keys and algorithms. Anytime communication takes place over an electronic network; there is always a risk of interference. Hence, people take security measures via cryptography to overcome risks. There are important elements that create the platform for cryptography, including authentication, privacy, non-repudiation, and integrity. There are also different cryptography algorithms and keys to suit the needs of users. Cryptography plays a crucial role in securing information and the image of a company .

Cryptography is used by big companies like Uber, GEICO, and eBay. This is because security threats have increased in the last few years and customers feel safe using services that prioritize their safety. When a company goes the extra mile to protect its customers, it boosts customers' confidence in the brand. On the other hand, some firms care less about customer safety and do not want to incur extra expenses. In today's society,

customers are informed and opt for brands that care about their welfare. Firms are hiring security specialists to educate them on cryptography and its application. Cryptography has employment opportunities for professionals. However, it takes time and commitment to learning how to works. Patient and committed individuals have an easy time understanding and encrypting messages.

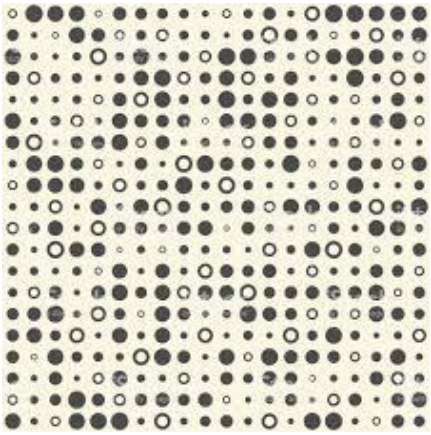
There are different aspects of security and applications that users should know more about. They range from secure commerce and payments to protecting health care information. Cryptography is an essential aspect of securing information. However, it is necessary to note that cryptography alone is not enough to guarantee security. We start with unencrypted data in encryption, commonly known as plaintext. It is encrypted into ciphertext which can be decrypted into plaintext. The kind of cryptography scheme being used determines the encryption and decryption processes. Cryptography is linked to the development of mathematical algorithms employed to encrypt and decrypt messages.



Example of visual cryptography



Cryptography



Seamless cryptography pattern

Pixel of secret Image	Encryption rules		The stacked results	Probability
	Share#1	Share#2		
				P = 0.5
				P = 0.5
				P = 0.5
				P = 0.5

Visual cryptography

Chapter 8:
Wireless and Network Exploitation

Wireless connectivity has gained popularity over the years. It has been around since 1971 when the first demonstration of a wireless packet data network was provided. It was forgotten until the 1980s when ALOHAnet's technique of sending packets of data was enhanced. Wi-Fi technology gained ground in the late 1990s. Companies realized the importance of connection mobility in business expansion.

For example, what if someone wants to present a PowerPoint presentation with no computer? He might have a laptop but does not have the Internet. The PowerPoint presentation needs a stream of the Internet. The use of Wireless Local Area Network (WLAN) in the firm allows him to move around with a laptop and give the presentation. Wireless works in a similar way to wired Ethernet protocol. An access point that supports wireless is used to grab packets coming through the wired connection. It then emits it in a radio frequency which is captured by the receiving node. The fact that Wi-Fi is mobile makes it fascinating, but since data is constantly passed through the air, it presents great security risks.

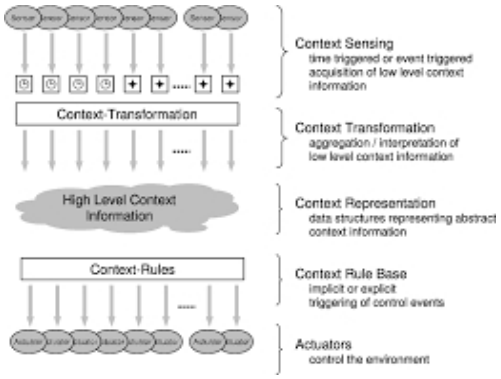
There are various wireless security protocols used to protect air-borne data through encryption. Wired Equivalent Privacy (WEP) is sometimes used. Wireless Access Point (WAP/2) is safer than WEP. However, these protocols can be interfered with, and although one may take some time to crack them, all of them face risks. People should not assume that the internal network is secure simply because it has a password. There are tricks of gaining access to systems without putting the correct password. Errors arise when the frequency is being transmitted regardless of what you want, even when you opt not to show the Service Set Identification (SSID). Anyone can be close to the SSID and grab encrypted packets and analyze patterns to decrypt data. Once the attacker is connected, he can harm because SSID is hard-wired in the entire business. While cracking a wireless network seems easy to those with knowledge on how to go about it, encryption techniques are helping firms are individuals overcome such risks.

The most exploitable Wireless Security Protocol is WEP, followed by 802.1x. 802. 1X is also known as WPA/2 Enterprise. It was created in response to the lack of security WEP offered because it had a problematic stream cipher. It was adopted in the business environment and facilitated

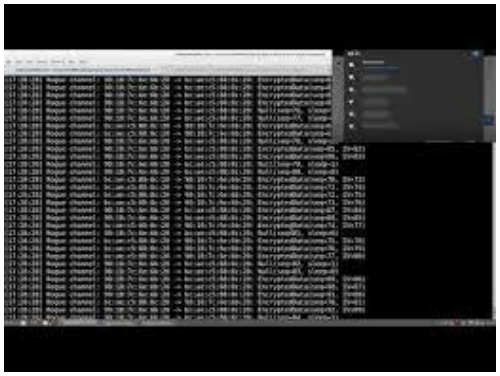
authentication by using the software. The client used software called supplicant.

Various tools can be used to exploit the security protocols, and one of them is the aircrack suite. This tool is everything you require to test wireless networks. It has to crack WEP, and WPA/2 is regarded as one of the best tools for exploiting wireless and networks. It is a gold standard of wireless exploitation and is automated. It enables users to picture what they are cracking, instead of sticking to the command line. It is GUI based and makes cracking WEP easy. It enables users to perfect aircrack features and is primarily for the use of WEP although it can also be used for WAP/2. Minidwep is another fantastic GUI based tool from the aircrack suite. It is an auditing tool used to audit wireless networks.

The good thing with using Minidwep is that it is fully automated and users do not have to fine-tune it. Moreover, it tests every attack against Access Point until there is progress and gives you feedback. It is a great tool to use in wireless exploitation. While it might take some time to break into a wireless network using Minidwep, it is ideal for any situation. Wifite is another awesome tool that automates the aircrack suite. This tool is not GUI based like the rest but is an automated command line. It automates the entire wireless network and is more of an “execute and go’ kind of tool. Although the aircrack suite is used for WEP or WPA/2 attacks, it does not a good job in exploiting WPS wireless security protocol. It is best to use two tools against WPS, wash, and reaver. They can be fine-tuned based on the WPS network you encounter. The role of Wash is to scan the air for wireless any wireless network nearby that enables your WPS. Once you get this information, you put it into reaver which launches the WPS attack. These tools are vital in the exploitation of wireless networks. They are found in penetration testing specific Linux Distribution. Below are examples of wireless exploitation:


Context-Transformation
High Level Context Information
Context-Rules
Context Sensing
time triggered or event triggered acquisition of low level context information
Context Transformation
aggregation / interpretation of low level context information
Context Representation
data structures representing abstract context information
Context Rule Base
implicit or explicit triggering of control events
Actuators
control the environment







Xiaopan is a wonderful operating system used to pentest wireless networks. It is based on a tiny core Linux. It has all the tools mentioned above, which makes it ideal for wireless exploitation. Moreover, it is lightweight and stands at less than a hundred megabytes. Its features make it the perfect tool for testing wireless networks quickly. WifiSlax is another wonderful Linux Distribution that helps with wireless exploitation. It has all the tools and other tools that represent aircrack in different ways. It is also lightweight which makes it better than other operating systems. WEP is the most vulnerable protocol for a reason. It was decommissioned in 2004 due to its lack of security. An RC4 key is used to encrypt WEP. It is done by taking the initialization vector (IV) and concatenates it with the password key that is connected to the AP.

Every number of WEPs represents the bit value of the WEP connection. Regardless of the WEP type, bits are kept for the encryption process. The kinds of attacks on WEP networks are steadily inclined focused on accommodating efficient techniques of acquiring the WEP key. The FMS attack was the first to take place and was named after its makers. PTW Method was the second attack that made it possible to crack WEP passwords. The creators employed statistical information found in the FMS attack method and established a loophole that could be used against ARP requests. Korek attack is another WEP exploitation method used by many. It is an unknown person who discovered exploits in the WEP protocol. The only way to stop attacks against WEP networks is to re-invent the wheel.

Others have implemented restrictive features to enable authorized people into networks. While this method has gone a long way in helping people stopping attacks, it has not mitigated the problem completely. A mitigation

technique that was used to prevent attacks was hiding SSIDs, but it was not successful. Hiding SSID is inconvenient because they were not meant to be hidden in the first place. Others believe that the best method of mitigating attacks on WEP is found on Dynamic WEP. However, it is the 802.1X standard implemented in the form of WEP and changes the password. User-friendly approach to securing WEP and is only considered to be a fix in the corporate world.

Similar to WPA2, it is not a good approach for protecting WEP and is a quick fix in the corporate world. The best way to tackle WEP problems is to use it the way it is. It was decommissioned in 2004 and WPA took over. Although WPA employs the same stream cipher, it secures your system better than WEP. WPA is Wi-Fi Protected Access. It is the solution to WEP. It brought about a secure protocol that did not have numerous exploits like WEP. However, it was offered a temporary solution and used to hold off attacks. WPA differs from WEP in several ways. Instead of employing the RC4 stream cipher for IV with the passphrase technique, they increased it to 24bit of IV to reach 128bit.

When WPS was developed, it seemed as if we went back on wireless security. It provides convenience for device connection to networks by enabling it to click a button and connect to an access point. Doing this enables users to save time in the configuration connection settings to the AP. They simply click and go. The access point allows the client to connect to the access point easily through the pin technique. To open the system, the client must know the secret pin implemented by the access point .

Some WEP vulnerabilities extend to WPA because TKIP is an improved IV that passes through the RC4 stream cipher. One way of compromising networks through attack vectors is to exploit the Temporal Key Integrity Protocol (TKIP). Because it utilizes RC4 stream cipher similar to WEP, you can use a packet and decrypt them. Capturing and deconstructing the 4-way handshake is the most prominent exploitation. It involves the use of brute force where the network is only safe when the password is. Following the same method to crack WPA and WPA2 is ideal because it reveals the password to the network. WPS is the final exploitation against WPA that is very common. It is a simple hack that is carried out on WPA/2 that allows

WPS by default. Sometimes hackers use brute force to gain access. WPA2 is the safest wireless security protocol right now.

Computer network exploitation refers to steal information from unsuspecting victims. The rate of computer exploitation has increased tremendously, and organizations have adopted measures to restore safety. However, nothing seems to be working to get rid of exploitation. If anything, the number of attackers is increasing with governments, organizations, and individuals taking steps to protect themselves. Some of the measures implemented to ensure security include the use of a firewall and encryption of data. Firms have also introduced training programs to educate employees on computer exploitation. All this is done with the intention of creating awareness and safety. Attackers have taken the game a notch higher by using sophisticated and appealing techniques to lure victims .

It is worse if they send links to people who do not know about lurking dangers or live in oblivion. Some people do not care about perils that they can encounter online and live a carefree life, even on the internet. This has proved costly for some people in leadership positions because they shared more than they were supposed to. Others have not installed firewalls and think that cyberattack is a myth propagated by people who are against technology. The truth of the matter is, computer network exploitation is on the rise, and everyone is trying to find a solution. People exploit networks for varied reasons.

Others do it to get back at their previous employer, others do it for fun, while others want financial gain. As discussed in previous chapters, the numbers of attackers who exploit networks and then demand ransom to return sensitive information are on the rise. Hackers are trying new ways of getting the most out of the situation. Because the industry has many hackers, everyone is trying to survive. The person who spots and takes advantage of vulnerable networks is the one that makes it. Some people also exploit networks to test their skills. Some computer students want to test their skills, and the best way for them to do that is to determine whether they can compromise systems. Others do it because of peer pressure and wanting to appear cool in front of their friends. There are many reasons why people exploit networks. The only sure thing is that it is rising rapidly

and organizations need to find a solution quickly before things get out of hand. With TV series and movies showing how cool it is to infiltrate networks, it is no doubt that network exploitation will not end any time soon. Organizations are doing everything to protect themselves from exploitation by boosting their security programs. However, a lot still needs to be done to mitigate against wireless and network exploitation. The first step is educating people about security risks, their role in preventing it and using effective strategies to mitigate risks once and for all.

The Dark Web Explained

There is the web, which basically is a term that denotes the existence of the world wide web. On the other hand, however, exists a dark web. The dark web is what you would consider being a zone beyond the safety of the common respectable Google, Amazon and other websites. It is like a dark room at the corner of a room and what is surprising even more is that there is a bigger picture when it comes to the dark web, and this is known as the deep web. The world of dark websites can be complicated, and this is why we compile this chapter to enlighten you about the dark web. This chapter tells you everything you need to know and helps you decide whether the dark web is worth you take a visit or not.

Unlike the traditional web, the dark web exists on what is known as the dark network and on the overlay network. These networks, just like the normal web, use the internet but require configurations, software and in other cases, some special software to access. This dark web does not end at this as it is part of a larger platform that is known as the deep web .

The deep web is special because it is not indexed as a part of the web search engines that are used in everyday life. Sometimes, the dark we are used interchangeably albeit mistakenly to refers to the dark web. Examples of the darknet include peer-to-peer networks which are small, to bigger, more popular networks such as 12P, Tor and Ruffle. Surprisingly, the bigger ventures are run either individuals or big, public organizations. As a user of the regular network, you may refer to yourself simply as a user of the network and may only designate the name dark web to websites such as those described above. However, the people that use the dark web would

call the regular internet the clearnet. While this term may be unfamiliar, the clearnet is simply the dark web user's way of referring to the unencrypted nature of the regular net. Below, we enter into a deeper discussion of the dark web and what you can understand from it.

To access the dark web, you have to go through platforms such as Tor. You may probably not know this if you are still new to the world of the dark web, then it would be beneficial to know that Tor stands for The Onion Ring. There is also the I2P which stands for the Invisible Internet Project. Tor and other sites that are accessible from the Tor browser are accessible to dark web users and are easily identifiable through the domain "onion." Tor is for anonymous access to the internet while I2P provides a different kind of service, which is hosting sites anonymously. As such the location and identities of darknet users remain unknown and untraceable mainly because the encryption system is layered .

The encryption technology deployed by platforms such as intermediate servers works to ensure anonymity and protection of the human identity. The information transmitted is not decrypted by a node is subsequent which is then leads to the exit node of the scheme. The node path is complicated, and therefore, it is almost impossible for reproduction and decryption of the information layer by layer. The high encryption level of the websites prevents the internet protocol and geographical location, and thus users are unable to get information from or about the host. Users within the network can allow the users to blog, talk and exchange information confidentially and in confidence.

Accessing the Dark Web

The paragraphs above highlight some of the ways through which the dark web can be accessed. However, below, we go into detail. We start with what has already been mentioned in the paragraphs above, and that is the Tor browser.

Tor in accessing the dark web

Interestingly, Tor was originally designed to help the United States safeguard its online intelligence communication. Over time, however, the

role of Tor has evolved, and it can be used as a browser that helps people gain access to parts of what we consider the dark web. One of the ways through which it does this is by allowing access to .onion websites which can only be found on the web. If you are wondering what Tor is still, continue below :

Tor is one of the versions of Firefox which a well-known and widely-used web browser in the world. Tor, however, helps the user to browse the web while staying anonymous. Interestingly, this process of helping protect the user's identity starts from the time when he/she is trying to download the browser. The browser has some systems put in place to help the user by advising them on what they should do to help keep their identities private. They give tips such as asking the user to cover their cameras with dark tape just in case someone is spying on them.

From a more technical perspective, Tor provides an encryption tool through which all sites on the dark web are able to hide their location, identity, and activity. When you use Tor, you can make yourself appear as if you are in a different country to your actual location. As such, Tor functions a lot like a VPN. Running a website through Tor produces the same effect. To be able to visit the dark website through the protection of Tor encryption, the web user needs to use Tor. The end of the internet protocol in such a case will be bounced through the layers of encryption so that the IP addresses do not appear as the actual addresses but as other IP addresses on the Tor network. This is why many individuals are able to visit the dark web, but it can be next to impossible to establish who is really behind the screen compared to when a person uses the traditional search engines .

VPN service

A VPN is a virtual private network. These are servers connect you to the web and can help you mask the place where you are no matter where in the world you currently reside. There are strong names in the market, such as Kaspersky that are lenient enough to provide customers with up to 200MB of data each month while utilizing their free VPN version. There are also other VPN alternatives that are paid including Nord and Express VPN. However, the ones mentioned here are just a drop in the ocean. There are many more VPNs, and they also provide superior service. However, it is

your duty to sift through the multitude and find the right VPN of choice for you as you contemplate whether to access the deep web or not.

DuckDuck Go

Over the last few years, new technologies have been coming up and shaking the status quo as we know. Google was and is still the search engine giant. However, this does not mean that it does not have equals. For example, the DuckDuck Go search engine. This engine may not be your friend on the mainstream platforms, but it is only on DuckDuck go that you will find sites that are not indexed on Google and yet they do the same job. When you try to access the dark web through the DuckDuck Go option, you will realize that it tells you that the search engine does not track you. Your activities are, therefore, anonymous and nothing is likely to come back to you again .

Make use of the dark web search engine

You may have caught the domain name .onion from the discussions in the above paragraphs in reference to the dark web. This domain is the same as a regular domain, but there is a slight difference in that you can only access it with a special browser such as Tor. DuckDuck Go mentioned above is one harmless address that you can easily try out and access. There are many other .onion websites that you can use with little to no effort, and all you have to do is to take some time to find them.

With this information at hand, once again, you are reminded that while in the dark web, you are likely to bump into things you may consider dangerous and even illegal. Take your time to accept the likelihood of such realities before venturing even further into the dark web.

The good side of the dark web

While the dark web is notorious for being a hotbed of criminal activity that is not all there is to it. The beginning of Tor was benevolent. Even today, the browser still serves a valuable purpose when it comes to communication. For example, it can be used to help people to communicate in places where free speech is prohibited. As such, there is a common use of

the internet where internet access is criminalized. Also, it provides a platform for people to learn about cryptocurrency and protection of privacy and much more. There are several private email services, and others are even encrypted. Through the darknet, you can also get instructions about how to operate anonymous operating systems and find tips to help enhance your privacy if you are privacy-conscious.

Also, the dark web could be a good source of information that you would commonly find on the public web. There are platforms on which you can discuss current events anonymously on platforms such as Intel Exchange even on the dark web. There are whistleblower sites on the public web. However, this is not the only platform where you can find such solutions. For instance, there is Wikileak's dark web version. Even BitTorrent and Pirate Bay that are not found on the public web after a commotion with authorities are very much existent on the dark web. There is also a Facebook version on the dark web.
In recent times, even legitimate companies are establishing their presence on the dark web, a move that could signify the beginning of a new era. Also, this shows that companies are increasingly becoming aware of the importance and even the power of the dark web. Even when the dark web is increasingly becoming open to the outside world, it is important that you remain vigilant and always be on the lookout because there are still potential threats that companies face in the presence of the dark web.

They include malicious code. Well, in the public domain, it is possible for an individual to become a victim of phishing. However, there are some steps that can be taken to prevent the eventualities of hacking. With the dark web in place, however, it has become easier for hackers to get access to attack codes which traditionally would have taken them too long to build from scratch. The dark web is a place that provides anybody available with the skills and materials to help them unleash malicious code. As such, there are malware packages on sale on the dark web that could prove detrimental to any organization. Some of these include the Dr0p 1t-Framework, which is a Trojan that is able to download malware then convert it into a word document file. This makes the file appear to be just like any other file until it infects the entire system. The worst part is that these codes come with instructions for those who have the less technical knowledge, which makes it easier to spread the malice.

If you still would like to visit the dark web at this point, then read on below because we outline some of the safe places that you can visit:

- The Hidden Wiki

The Hidden Wiki is a relatively safe place when visiting the dark web for the first time. This site operates like its open web counterpart Wikipedia. The Hidden Wiki contains a lot of links and information that can even give you the insight that will help you navigate the Dark Web. This is one of the major and most used stalwarts in the .onion domains and shall remain so for many years as well. If you are, therefore contemplating visiting the dark web for the first time, then this can be a relatively safe place to start. Here, you will not only get guidance about how to navigate the rest of the dark web, but your activities will be relatively secure.

- Dream Market

You are probably thinking of the darknet because you are in search of a place to buy something which does not necessarily have to be illegal. If this is the case, then you are in luck because Dream Market provides you with a platform that you can use to browse and even buy goods. It is a marketplace just like Amazon and can be trusted because over the years, the FBI has been carrying out operations to wipe out illegal trade, and a platform such as Dream Market has stood the test of time when other major names such as Silk Road have been stamped out. This can, therefore, be the best place to start.

- The Hidden Wallet

There are a lot of things that can be bought over the darknet. Given the way that the dark networks, you will have to find a way through which you can pay for your goods. The Hidden Wallet works just like a digital wallet. The major difference is that this platform promotes the use of bitcoins as currency and that unlike other digital wallets, the Hidden Wallet is anonymous and nobody can trace your transactions in this way. Additionally, due to the nature of these platforms, the site does not comply with the financial regulations of any country. Just like the name suggests, the wallet is indeed hidden from the outside world.

- Private Hosting

Local web hosting companies have become a pain in the neck because most of them are always asking for the user to look into the user agreement. Sometimes the terms of these agreements may raise concern for the user as it may create paranoia about what is being done with the information the sites require. If you are the kind of person that worries about these things, then private hosting has come to your rescue. Private Hosting is a platform that offers you an anonymous hosting on the web that also remains secure. Even Linux PHP gives a good offer for private hosting where an individual gets unlimited bandwidth and 100 MB for a low price of 170 yearly.

- Facebook

The name Facebook and the dark web do not look like they can go together. However, there is a Facebook on the dark web with the .onion address. Well, if you are wondering what this Facebook is all about then, it would be fair to let you in on the fact that this Facebook was developed as an alternative for people who would have wanted their social networks to remain anonymous. The point of social media, however, is to stay social so it is quite perplexing that a site like this would require anonymity. However, the general reason as to why people use it on Facebook is because it does not keep logs of user activity.

- Buying bitcoins

Bitcoins are a superb option of currency for many because it helps you remain anonymous as you transact. This especially is the reason why many people use bitcoin on Tor. The benefits there, however, are more than just the anonymity but also the fact that the .onion address has an HTTPS certificate attached to it, which shows a higher level of security can be expected.

- Rent-a –Hacker

One of the other very popular reasons for someone scouring through the pages of the darknet is that they are trying to hack into systems or even other people’s accounts. While this may be wrong on all counts, hacking

services are also some of the most commonly sought after on the dark web. Rent-a-Hacker is a freelance hacking service that gives people who want to hack a chance to hack at a price. The prices begin at approximately 250 euros.

Disadvantages of using the dark web

While reading above, the dark web may seem like the answer to most of the questions you may still have unanswered. Additionally, it may sound thrilling and even exotic. After all, there is a curiosity in many that occasionally as humans, we may feel we need to feel. But just like the wide, beautiful sea is beautiful, it is also dangerous as many creatures are hidden within. Such is the case of the dark web. You need to understand before venturing out that the dark web is not a place where you go blindly.

You may just be going about your day on the dark web and unluckily stumble into people or even things that could result in very dire consequences for you. The darknet also has a negative side. It can be used for activities that are illegal including illegal forums, trade, and exchange of terrorists and pedophiles on the media. In other words, most of the cybercrime you know of today happens in the dark web. Cybercriminals are no longer just after money but much more. These hardened criminals will take much more than just money, and some of the information anyone can access there includes personal information, credit card information and just about anything imaginable. In the same spirit, these items can be bought and sold as they wish. In the worst-case scenario, you are likely to run into problems with law enforcement agencies in the long-term. We expound on some of the consequences of getting involved with the dark web below.

- ❖ Firearms, cash, and precious metals

Expect to meet the buying and sale of firearms, gold, and even cash on the dark web. It is not unusual to hear on the news that one or more people were nabbed by the authorities for dealing in firearms sales illegally or for stealing cash and most recently, getting involved in illegal bitcoin trading on the internet. Another popular commodity being dealt with on the internet is gold. Even hitmen can be hired from these places.

- ❖ Distribution of drugs

It is also not unusual to find people who sell illicit drugs on the dark web under different trade sites. They sell opioids and many other drugs that are considered illegal, and they use various strategies to avoid getting caught by the authorities. Some of these strategies include the use of cryptocurrency, proxies and even VPNs.

- Kidnap, Pornography and Sex trafficking

Those who deal with pornography, kidnapping, and sex trafficking are also present on the dark web. While this is depraved and not the best of ventures that you would want to undertake on the internet, it is important to understand that this is a part of the dark web too. Not once have people hit the news headlines for kidnap and trying to sell their victims on the dark web to make a living

The dark web is, therefore, a place that can present deep challenges for those who wish to explore it and care should be taken to ensure that you know exactly what you are getting yourself into before you decide to be a part of the dark web.

How to protect yourself on the dark web

A lot has been discussed with reference to the dark web, and so far, we have established that the streets of the dark web may not be the best in terms of safety and there are indeed some terrifying activities that you may find there. As you browse on the dark web, however, there are some steps that you can take to ensure that you stay safe:

- Update your Tor browser

Like every other browser you know by now, the Tor browser is also likely to have its own weaknesses. That is why it is important for you to remain vigilant at all times. Keep your browser up-to-date and always be on the look-out for vulnerability notices as this is the time that any are likely to get caught unprepared and the criminals take advantage.

- Make use of a reputable VPNs

Using VPNs is important for dark web users as it is a platform that helps you keep your personal information, including address and identity safe while also protecting you from data breaches. However, when it comes to the issue of VPNs, everything is not a one size fits all. Your VPN should, therefore, at least, meet minimum criteria. One important factor to consider in this case would, therefore, be that the VPN you chose should be from a country that encourages data retention. When this is the case, there is a higher likelihood that your information will be better protected.

- Avoid Macros

There are apps that run scripts, and there are also Macros which can be risky for you if you want your location to stay private. These may be needed on normal sites such as YouTube. However, when you are on the dark web, and a site begins to ask you to enable scripts, then there is definitely a problem, and you should look out to ensure that you do not get your computer infected by malware and viruses.

- Be careful about your downloads

The dark web is not safe at all, and that is why you have to be careful about what you download on there. Unfortunately, malicious code can be packaged in any type of file, and the worst part is that it can be hard for you to realize that the packaging is that of a virus until it is too late and the malware is eating into your system. Sometimes, it is even better to use a virtual machine so that in case there is malware involved; you can isolate it from the rest of the operating system.

- Adjust your min d

You may be going to the web daily and roam the streets even when you understand there is a significant threat. Until you get attacked by malware on the internet, there is a probability that you will not understand fully the eminent danger that surrounds browsing the internet, especially the dark web. If you think that the darknet is a safe place to surf, the results could be fatal. As such, you should not trust anyone on the darknet and you should always be careful so that you do not unknowingly infect your computer with malware. In short, be very skeptical of people on the dark web as it is

highly likely that they are in search of something, do not fall prey to the antics of these people at any one point. Train your mind to recognize this place as a potential hazard.

Conclusion

Kali Linux has become increasingly important for businesses and individuals as well, and therefore, hacking is a skill that is much needed in the world today. One can only get away with hacking by using advanced tools and have proper skills so if you want to be able to do this, these are skills you need to master. Hackers should use Kali Linux because it gives them all the tools they need. Furthermore, other tools can be downloaded easily to start the hacking process. The user must use Kali Linux because its goal is simple; to include as many security auditing tools and penetration as possible. It also delivers, unlike other hacking tools that people have been using. Moreover, users should use Kali Linux because it is a convenient solution.

It is not a must for them to maintain a Linux install. However, using it as a daily driver OS is something that you want to avoid, and the reason is that it is meant for penetration testing and should not be used for other purposes. It is important to note that Kali is not for everyone. It is not something you can install into your laptop and start running a hacker operating system. There are instances when you can run Kali temporarily from a USB drive. There are other times when you need a Kali to install for long-term testing. You should use Kali Linux because it comes with over 600 penetration tools from the forensics and security fields. It also comes with a custom kernel, and it is developed in a safe environment. Knowing this information is vital when understanding the Linux system and how to use it to your benefit.

The ethical and unethical parts of hacking are a hot topic, and many argue back and forth repeatedly on this subject as many think that hacking is something that no one should be doing and others think that it’s necessary and needed. Because this is such a hot topic there have been literally thousands of debates on this and this is why we have dedicated a portion of this book to bring light to the subject.

We have also brought your attention to exploitation as well as malware and cyber-attacks. Being aware of these issues is important if you are wanting to do this well. Cyber-attacks can come out of nowhere and everyone knows the dangers of malware, but we make you totally aware in case there is any doubt in your mind as to just how dangerous. There are many things that

you need to be aware of when you are trying to understand how to and what you should be doing as well. As such, we have given you solid information on everything to do with the Linux system so that you know how to use it properly and to your advantage.

If you are able to follow these tips and use the information that we have given you in this book, you should be able to perform the tasks that you need to with ease and learn how to understand the Linux system without any difficulty .

Finally, if you found this book useful in any way, a review on Amazon is always appreciated!

Hacking with Kali Linux

The Advanced Guide about CyberSecurity to Learn the Secret Coding Tools that Every Hacker Must Use to Break All Computer Configurations with Networking, Scripting, and Testing

Darwin Growth

Introduction

Congratulations on choosing *Hacking with Linux: Underground Beginners Tools to Learn the Basics of CyberSecurity and Become a Hacker by Breaking into Every Operating System with Ethical Linux and Precise Computer Configuration* and thank you for doing so.

The following chapters will discuss hacking with Linux systems in detail. Hacking is an art of exploitation and can be used in various useful and dangerous purposes. This book helps us to understand hacking concepts in layman terms. Apart from a thorough explanation, we will also get an example that will help us to expand the horizons of the topic.

Hacking is usually complex and may take a lot of time to master. As a matter of fact, according to a recent anonymous study, it usually takes ten years to become a professional hacker. In the computer industries, there is obviously a lot of need for ethical hackers due to various reasons. For these reasons, you need to master hacking using Linux for better career opportunities.

This book delivers a lot of topics in a smooth way. We will first discuss the importance of Linux and hacking in detail and then move forward with a lot of concepts .

Why Linux is used in this book?

You may have your reasons to avoid Linux but as far as hacking is considered Kali Linux is the best bet you can take. Also, Kali Linux is an operating system that is open-source and can work out things in a better way, unlike Windows which always block hacking tools via a firewall.

How to get the most out of this book?

This book at first will give a layman explanation to the topic and in the next step provide commands so that there will be no misunderstandings of the subject. Also, at the end of every chapter, a clear explanation of the things we have learned is described. So, do follow it.

There are plenty of books on this subject on the market, thanks again for choosing this one! Every effort was made to ensure it is full of as much useful information as possible, please enjoy it!

Chapter 1:
Introduction to Linux and Hacking

This introductory chapter in detail will introduce you to the world of Hacking and Linux in detail. Hacking is a process to exploit or take control of a system. Whereas Linux is an operating system that is used for various tasks that a usual everyday operating system like Windows or Mac does. In this chapter, we will give a brief description of hacking and how it works along with a brief step by step theoretical introduction on why Linux is the best operating system for hackers.

Let us start the chapter now with a brief introduction to Linux in the next section. Remember to take notes while trying to learn this stuff as this is a layman's introduction to the topic. Try to explain the concepts you learned in this book to someone who never studied the subject by using your own words to make a good foundation on the subject. Let us start now the exciting journey about Hacking using Linux.

What is Linux?

Linux is an operating system that is so much related to UNIX one of the few pioneers in the early computing industry. One of the most important things for the success of UNIX is its excellent portability. Anyone can obtain the source code and write their performing system in back those days unlike commercial operating systems like Windows and Mac OS.

With the success of UNIX operating systems in the programmer's arena, its creator has started the GNU project to expand the possibilities of the project. A lot of programmers enthusiastically participated in this small world. Out of those programmers, a college-going student Linux Torvalds has started writing his code for the project. And within few months he has released the code which got the immediate attention of the other enthusiasts. Due to its robust and simplistic nature, Linux has become everyone's favorite within a short frame of time.

Its creator has used repositories to let people contribute to the project. Within few years Linux has expanded into a level of the commercial operating system with the help of thousands of programmers voluntarily contributing the project.

From then Linux has expanded into various Distros and different editions developed by different enthusiasts. At present, Linux is one of the most used software by programmers, database experts and most importantly by hackers. In the next section, we will discuss in detail the reasons why Linux has become popular?

Why is Linux the Best Operating System?

One of the most important success notions for Linux is that it has a great adaptability for server-side systems. Database administrators are more comfortable in using command-line interfaces than Graphical user interface systems which have become common. Below we will discuss other reasons in detail.

1) Open-sourced

Linux is one of the few open-source operating systems that are available. It means that anyone can download it for free and can use it to develop custom or third-party operating systems, for example like Red hat Linux.

2) Degree of Modularity

Linux kernel consists of five different parts that can be changed or organized according to their actual needs. That is one can select or filter the functions that are important for them. This made individual users interact with the operating system more comfortable.

3) Hardware support

commercial operating systems need to update their hardware support after a few months. Whereas Linux operating system gives much before hardware linkage due to its large number of contributors that write code independently. Due to this reason, there are a lot of hardware systems that only work with Linux.

4) Security

People have a misconception that open-source systems are prone to vulnerabilities and easy attacks. But this works oppositely because Linux

has more Authorization and administration security features than windows. Users can select their security features in both basic and advanced levels.

5) Multi-user & Multitasking

Linux can be used by different users and on different levels by the options they select. It doesn't decrease the speed or change the way the system works. Apart from multi-user functionality, Linux is also very good at Multitasking. That is performing a lot of tasks in the stack.

6) High level of portability

As we said before Linux systems can be connected with any other systems from high-end servers to low-level Arduinos. This is the reason why Linux is loved by system programmers to robotic developers.

In this section, we have given a detailed explanation about Linux and its characteristics that made it a favorite and famous operating system. In the next section, we will give a brief introduction to hacking and explain in the later section about why Linux is the most preferred system by hackers .

What is Hacking?

Hacking is one of the most controversial and creepy terms that has taken over our technological generation. A lot of years back when there are telephone lines and are used extensively for communication few people tried to exploit these networks of telephone lines with a system called phreaking so that they can make calls for free. This is one of the first methods of system exploiting that is known in a wide range to programmers who are developing software systems.

Hackers by a definition mean that people who are trying to exploit the system either a network or software or hardware by using different techniques either by a program or devices. Hacking has been divided into a lot of hierarchies and types which we will discuss in detail in the next chapters. But for now, assume that hacking means to exploit systems by code.

Who are Hackers?

Hackers are the individuals who try to exploit systems or understand the loopholes in the system to fix them. People always treat hackers as bad guys but try to understand that all the security administrators who try to protect the billion-dollar servers and databases are also technically called Hackers. In the next, we will discuss the most important characteristics of hackers in detail .

Important Hacking Skills to Have

This section will in detail explain the most important areas where a hacker should be perfect. We will organize these characteristics in such a way that it would be easy for the reader to go through this book. Let us start!

1) Fundamentals of the operating system they are using.

A good hacker will always have a sound knowledge about the operating system that he is using. It deals with a lot of technical knowledge like memory and process management along with a lot of commands that will make the work easier. A lot of professional hackers will have a detailed knowledge of the Kernel they are working on. So, try to consume the operating system knowledge as deep as you can.

2) Scripting

A good hacker will always write his programs to automate the tasks that otherwise may take a lot of time to do. Get accustomed to a scripting language like python and start writing your programs. If you can't write your code there is less chance of being called a good hacker. So always try to experiment with things using scripting.

3) Knowledge about the Internet and we b

The most basic task of hackers is to exploit networks and web applications. Thus, a thorough knowledge of these technologies is a must. Always get yourself updated with the latest server and web technologies that are being

developed upon. Learn about port scanning and vulnerability testing in detail for making fast progress as a hacker.

4) Hacking tools

It is practically not possible to write your tool for every task that you need to perform. So, you need to understand the usage of a lot of hacking tools that are available to perform a lot of basic, moderate and advanced tasks. Always try to experiment with new tools and you can have fun exploiting systems.

5) Ethics

A hacker can perform both good and bad actions. However, remember that there is a lot of satisfaction in stopping the bad boys from making money using loopholes. Always understand what you are into before doing a certain attack or exploitation. It's you that should decide to play on which side.

These are the basic characteristics that need to be learned or one should be aware of thoroughly to become a hacker. In the next section, we will in detail explain why Linux is the best operating system for hackers. Let's jump right into it .

Why Hackers Love Linux?

As we said before Linux is the popular operating system among Hackers and system administrators. Many reasons made hackers from generations to choose Linux as their default operating system. In this section, we will discuss this in detail.

1) Control

The normal windows have a graphical user interface that can be easy to work on with most of the users. But apart from easy interface windows and Mac restrict its users to manipulate the system or the organization they have created. Whereas Linux works on the contrary way. The command-line interface which is a basic building block of Linux can help users to

manipulate the system they are working on in any way. This lets the users perform complex tasks and that's what hackers need all the time.

2) Security

For now, Windows has a huge number of applications being developed followed by Mac because there are a lot of novice users using these systems to perform their daily life tasks. This has made them easy prey for crackers and hackers who in a huge number try to find loopholes in the systems. But Linux has more security features among the all other OS and thus is often preferred by hackers who take their anonymity very strongly.

3) Changing the OS features

Linux, when compared to other lets its users change things as they like. People can sort out the important networking tools they use and can customize it on their desktop. A lot of other functionalities like Ram management and Mac address can be easily manipulated using Linux, unlike windows. This customization capability is one of the most important reasons why hackers prefer Linux. There are a lot of customized GUI interfaces like GNOME, KDE, MATE for Linux.

4) Expanding things to fit your preferences

You might have heard about Star Wars already. It is one of the most famous science fiction books that have changed the whole genre. It worked on a concept that one can create their storylines using the star war universe. Linux too works in the same way. By using the kernel source code one can create their Operating system called Distros in Linux world. This has made much easier for things to organize in a better way. And hackers too utilized this functionality to make things better for them. In the next section, we will discuss this in detail.

What is Linux Distro for Hackers?

As said before Linux provides the capability of creating Distros which are predefined systems. Some professional Hacker communities who wanted to give back to the community have created backtrack Distro for hackers. It has been a huge success and has become a pathway for novice hackers.

Backtrack provided a lot of open-source hacking tools pre-loaded in the system along with a lot of editing tools and IDE's. After a few years backtrack has discontinued and evolved in a new form as Kali Linux which is now one of the most used Operating systems by Hackers. It has a lot of hacking tools neatly organized. Apart from Kali Linux, there are also Distros like Parrot Linux that are used by hackers. In this book, we will use Kali Linux as default Distro to explain the concepts for a wide overview. In the next section, we will explain how to install Kali Linux in detail. Follow along to understand things easily. Let's go.

Installation of Linux Distros

In this section, we will discuss how to install any Linux Distro with step by step instructions. We will also give a detailed analysis of topics like partition, boot loader, and others to get a good overview of the Linux essentials. Let us start now!

Essentials for installation

Step 1:

Get the Boot ISO file of the Linux Distro you need to install on your system. You can download it from the websites to load into a bootable CD or USB.

Step 2:

If you are interested to use windows at the same time in your system you can use double boot option to choose the operating system at boot up time. Anyone of them will be selected to boot according to your choice.

Step 3:

You can also use a virtual machine software like VMware in windows to boot into Linux. This is the most preferable way by hackers because it becomes tough to track when you are using a virtual system.

By this explanation, we have understood the different ways we can use Linux. Now in this section, we will look at the step by step installation of

any Linux Distro. We will assume for our convenience as Kali Linux as we are learning about hacking.

The Steps You Need to Install

1) Start the bootable CD or USB or virtual machine to enter into the Graphical user interface of the installation process. There is also an option to install the Kali Linux by using the command-line interface.

2) In the next step, the Boot file automatically detects the required drivers. And if there are any malfunctions drivers the system will not further proceed into the installation. You can google the error if you are stuck with any to get a step by step troubleshooting procedure .

3) In the next step, you can select your preferred language. Most of them would select the option with " English (us) ". Choose whatever language you want the interface to be.

4) In the next step, you can select the time zone and country. Select the options you want to use.

5) In the next step, we need to select the network monitoring settings. Connect to any wireless network if you want to. You can even select the network drivers that you may need to install in this step.

6) In this step, we will go through the most important step in an installation that is partition. The partition will help us to determine the file system we wanted to use. There is an option for ext3 or fat32, choose whatever you like.

And in the next step choose the hard-disk drives that need to be partitioned. There is an option for complete partition which will delete everything and do a fresh install. You can also use custom partition according to your needs.

7) In the next step, we can input a username and password for the operating system. Re-enter the credentials and start the installation process .

8) After some time, the installation procedure finishes, and we will be welcomed with a welcome screen of Kali Linux or any other Linux Distro you are trying to install.

That's it. It's all we need to know to install a Linux Distro on the computer. We are now full of knowledge and practically ready with everything to experiment with Linux and start hacking. Get ready to have fun with Linux basics that we are going to learn in the next chapter. A good overview of Linux structure can help us hack things efficiently. So, let's start to go!

Chapter 2:
Basic Linux Commands

In the previous chapter, we had a brief overview of Linux and hacking along with a step by step installation of Linux. In this chapter, we will talk about a lot of Linux basic concepts like process management and file management with commands in detail. This chapter helps us to use the Linux structural concepts to exploit systems with efficiency. Before starting the concepts, we will learn about the help section in brief.

How to Find Help while Using Linux?

Linux is a pretty open-source operating system with a lot of commands that one can operate with. You can use the below choices to find help if you are stuck with any command or scenario.

1) Use the man page in the Linux system. This is the default help page for any Linux Distro. It approximately holds 2600 commands and its use cases. You can search among them to know about things you want to seek.

2) Use Stack Overflow when you are struck with any error or warning messages.

3) You can search in Linux forums or GitHub issue trackers to find any bugs or issues in the operating system. In this way, you can improve your knowledge exponentially.

In the next section, we will discuss various Linux basic topics in detail. Let's start our journey into the Linux world.

For a better understanding of these important concepts, we have divided the concepts into three main structures. They are described below.

1) User management

2) File management

3) Process management

We will discuss these three Linux building blocks in detail in the next sections starting with user management.

User Management in Linux

Linux offers a lot of options for maintaining its user groups. There can be both individual users with only reading abilities and individual users with writing ability. Groups can also be used for easy organization of workspaces and teams working on different projects on the same server system.

Why user management matters to Hackers?

Before learning in detail about the user management system in Linux it makes sense to know why hackers should know this. We will explain this in a simple scenario.

Assume that an attacker wants to exploit a server system of a big software company. He uses simple port scanning at first (we will talk about this in later chapters) to find open ports. With the help of open ports, he will try to find access to the system. But if he tries to attack an open port, he may get detected by system administrators easily. So, to make things easy he needs to find user groups with reading abilities and get access to the system.

In this similar way, professional hackers use the user management system to exploit the systems that they got access to. For this reason, a clear understanding of the user management system is important for anyone who takes hacking seriously.

User management system in detail

Linux uses a process called usernames to distinguish between different users. People can also use a password to protect their accounts.

There are three types of users in normal in a Linux system which we will describe in detail.

1) Normal users:

Normal users are the ones who can access their directory only. They can't access other directories, if they are permitted to access other directories then they are restricted to write any files. Normal users are given a mandatory UID which will help others to recognize the user.

2) Root users:

Root users are the ones who are the administrators of the system. They are called super users and have an Id which is 0. Root users can write and modify any part of the file. If a hacker can get access to a root user account, then he can completely delete the files that are present. Usually, server administrators are Superusers, so it is very tough for hackers to get administrative privileges of a system.

3) System users

System users are not real users but users that are created by the programs that are run in the system. For example, when the Chrome browser is started certain system processes will start with the name of system Id. It is important to track and sort these system IDs for better usage of the user system that Linux provides.

Below we will give a command that will help us understand the Linux user system:

linuxexample @ host: systemid 2344

Apart from users Linux also offers group systems. Groups are important for the management of a huge number of users working on the same project. It also helps to organize things in a better way. By default, every user in the Linux system belongs to a group.

Every group is represented by GID in Linux. The only Root user can create groups and organize them in a definite way. However, a user can be in one or more groups according to specifications.

Below we give some commands which can help to know your current user ID and group id:

linuxexample @ host : get UID

linuxexample @ host : get GID

As discussed earlier every Linux system account needs a username and password. Passwords, as we all know, are prone to attacks and are the first thing that can be tried to exploit. Passwords can give one-way access to all the sensitive information for hackers.

Linux usually holds all passwords in two files. They are

1) /etc/passwor d

2) /etc/shadow

The first one stores the password of the current user and can be easily read by the user.

Whereas the second one is more sensible and contains passwords of all the users in the system. This can be only read by the root user and will not be visible for normal users.

In the next section, we will describe various commands that can help us to modify user groups. Follow along:

1) New user

In Linux, if you are willing to add a new user to the system you need to enter the following command

linuxexample @ host : new UID 5363

Whenever you tried to create a user using the following command with a name you would be opted to choose a password which is then stored in both /etc/password and /etc/shadow. After the successful addition of a user, a separate UID will be given to that particular account.

Apart from that, a new home directory will be created for the user. It should be remembered that a default user group will be created on the user name. There is also a special directory called /etc/skel where all of the configurational files of the user will be stored.

This is all you have to know about creating a new user in the Linux systems

2) Change password

The password is the single pathway to access all the directories and files in the user system. You can use the following command to give a password to the user.

linuxexample @ host : pw UID 2435 to strange

It should be remembered that ordinary users cannot use their username as the password. However, root users do not apply to this condition.

3) Modify the user

Modifying is always a preferable option for hackers. Hackers usually insert an exploit or trojan in the directories and try to spread them into the whole system. For this reason, hackers should be aware of modifying a system.

user mod command is one of the most important commands that can be used to modify the home directories that are present. You can also use grep command to display the content that is present in the system directory .

- m option can be used to change the default directory of the user so that everything present in this directory has advanced privileges.

4) Delete the user

Sometimes hackers after getting what they want from the system will try to delete the user system to delete any log files that may be used to detect the attacker's identity. This is one of the most important commands to learn if you are looking to attack systems with huge security.

By using the user del command as shown below you can delete the normal user and if it is performed by the root user he can delete any user that is present in the system.

linuxexample @ host : UID del 239844

5) Add group

User Groups are created to make things easy for the administrators to make the mess clear away. In Linux, there is a separate directory called /etc/group that will store all the information related to user groups.

Below we give the command that can be used to add the new group. Follow along:

linuxexample @ host : add GID 378 3

After entering the command, you can cross-check in the /etc/group to confirm whether a group is created or not.

6) Delete a group

Like the user system, it is also productive to delete the user groups if there is no use of it anymore. Hackers too evidently use this command to wipe out a set of user groups that they find easy to be get traced.

Below is the command to delete a user group:

linuxexample @ host : groupdel GID 3453

Remember that every directory and information that is present and related to every user will be wiped off. There are very fewer chances of getting the data back unless one uses professional backup and recovery tools.

7) View the User's

There is a special command in Linux called uses that can be used to check all the users that are present in the system no matter if they are alive or not at the moment.

Here is the command with an output for the user's command

linuxexample @ host : view UI D

Output:

2443 sample1
8942 example

0987 admin

4673 systemuser

8) Finger Command

Finger command is used differently and is used to find the awake user or group systems at the present moment. This can be highly beneficial when performing complex system tasks.

Below is the command for the finger functionality with an output:

linuxexample @ host : finger UID

Output:

2443 sample1
8942 exampl e

9) Cron Command

Linux system runs on services. Services are basic tasks that can be used by operating systems to perform their desired tasks. Cron command is a special set of commands that can be used to automate tasks like starting an antivirus automatically when you boot up the system or switching into a VPN profile when accessing a particular software.

Below we explain with an example about cron command in detail:

service crond start

This starts the cron service and lets us run things.

Service crond status

With this command, you can easily check what is going on.

All of these tasks will be stored in the /etc/crontab directory.

By this, we have explained a detailed explanation about the Linux user management system and in the next section, we will in detail explain about file management system which is one of the most pioneer branches that a hacker should be perfect at. Let us go!

File Management in Linux

Every operating system follows its own algorithms to organize files and called them a file management system. Windows users as a simple file management system that can be easily modified or applied using keyboard shortcuts or graphical user interfaces.

But in Linux, a set of commands should be learned to perform even simple tasks like moving or copying within the directory. This gives a good level of secure functionality to Linux. In this section, we will learn about all this stuff in detail.

Why do hackers need to be aware of file management in Linux?

Linux is an operating system that deals with files and directories in a whole lot different than windows. Hackers need to learn about modifying directories so that they can easily exploit the system when they gain access. We will know about a few of these now.

1) Absolute path

Every directory or file has a distinguished path that can be used to notice or manage things easily. There is also quite a complex successor of it called a relative path about which we will discuss in the next section.

For example, consider this path

/Srujan/downloads/red.mp4

This is called an absolute path because it all over points towards the file.

2) Current directory

When we are working in Linux, we often want to know the current directory so that we can easily modify things. You can use the following PWD command to know the details of the current directory.

vulnerhost @ example : cd /etc/get

3) Relative path

Relative paths are the upper and lower levels of the absolute path we are working with. They are represented by the dots and are called as special directories by the Linux creators. Below is a small example using the command to understand in a better way.

vulnerhost @ example : cd /etc/get
.....
vulnerhost @ example : cd /etc/

4) Touch to Create Command

Linux uses various methods or commands to process files. Here we will explain about creating a new file of any type using touch command.

Linux can create a lot of file types such as .txt or.MP3 by using the touch command as shown below using the command.

First of all, you need to be in the directory you wish to create the file and then enter in the following way:

vulnerhost @ example: touch sample.mp3

After entering the check command, you can recheck using the -l command for viewing the files in a current directory.

5) Delete the file

Deleting a file is a basic task any hacker would need to learn. In windows, you can delete with a click, but Linux offers a special security layer that

will not let you delete any system files such as log files. To remove any files, you need to use the rm command as shown in the following example

vulnerhost @ example: del example.gif

6) Move or rename files

Linux provides a command that can be used for both moving and renaming files. Moving files may become essential to hide forensic investigators about the attack you have done. It may be also a lifesaver to easily get sensitive information.

This command can also be used to rename files. Below is the example that describes the following two use cases:

vulnerhost @ example : mv example.gif

[for moving the file]

vulnerhost @ example : mv example.gf to report.gif

[for renaming the files]

7) View the file

Normally text editors or IDEs can be used to view the file content. Certain file types for suppose like an MP3 can be opened by a music player software. However, Linux provides a command called the cat that will let us read the file in the same way that Linux kernel reads it. Below is the command for the cat command with an example.

vulnerhost @ example: cat songs.tx t

8) Head Command

Normally files consist of a lot of information and can cause crashes will opening using the cat command. To get rid of this disadvantage you can just see the first 10 lines of any file using the head command. This will help for a fast recheck of log files when there is an attack or delete your login information when you are the attacker.

Below is the command to view the file:

vulnerhost @ example: head songs.txt

9) Tail Command

If the head command lets the users see the first 10 lines of the files, the tail lets the users see the bottom 10 lines of the code. This can be used to easily organize the system programs by using their results or to see the final result of the system processes.

Here is an example that describes the following command:

vulnerhost @ example: tail songs.txt

10) Make a directory

We have already described a directory as a file system that hierarchically stores files. This mkdir command can be used to create a new directory on Ur preferred location .

vulnerhost @ example: mkdir songs

11) Delete directories

Normally hackers delete a lot of directories just for fun. Or sometimes they do it to keep their traces off. By using rmdir you can easily delete the directory you prefer.

Below is an example that explains the command in detail:

vulnerhost @ example: rmdir songs

12) Copy Directories

Replication is one of the important concepts that hackers need to learn and should be perfect at. When you attack and get access to the system you will look for sensitive information such as usernames, credit card names that lie in the system. System administrators would get alert if any of the data is

deleted or moved. So obviously a lot of hackers replicate the data into their servers or physical devices using the cp command.

Below we give an example for cp command in detail:

vulnerhost @ example: cp songs to movie s

13) chmod to Change Permissions

Linux uses permissions to make things work in a better way. Usually, only root users can modify any file or directory that is present. Sometimes hackers will get access to the system but will get unsuccessful in performing system tasks due to no valid permissions.

However, Linux provides a command called as chmod that can be used to change permissions or give access to your files to other users. However, remember the fact that the root user can have access to all of your information.

Below we explain with commands about this functionality in detail:

vulnerhost @ example: chmod +x songsdir to movies

14) Find Files

It's usually difficult to find files in Linux than in Windows because it uses a separate file system, unlike windows that sort out files easily. Certain commands can be used to find the file you are looking for easily.

a) General search

This is the normal way to search in databases and searches every directory that is present to get the desired file you want. Below we will give some commands that will explain the general search in detail .

Finding an mp3 file named “beatles.mp3”

We enter the command find with the pathname and file name as shown below:

vulnerhost @ example: find file beatles.mp3

By this, the find command will start searching the required file and will give the results. If nothing is found for the name, then the empty screen will be displayed.

There is also an option with an asterisk that will show every file with the extension. Suppose we only give .mp3 extension as a file name using asterisk then find command will search every mp3 extension that is present in the directory.

Although find is used extensively to search for things in a Linux system it is often slow. The reason for this is that it starts from the root directory to search the file name. For this reason, there is another command called as locate that will reduce the search time. We will learn about locating in the next section.

b) Database lookup

Linux contains system files and databases that exist with them. As explained earlier it is difficult to search every application database due to their length and huge numbers. For this exact reason database lookup is introduced in Linux.

This will search and input the database files that are present according to the search term. Before using the locate command you need to use updated so that everything will be refreshed to display the fresh results.

Below is the command for locating command. You can use the file directory that you prefer or command will search all directories that are present.

vulnerhost @ example: updated

vulnerhost @ example: locate songs.mp3

c) Find the execution file

Execution files are special installation files that are used to make programs run for the first time. Windows use this extensively but Linux can also run

them with certain software. You can also search execution files in Linux with a command called. This will display the path of the system file.

Below is the command example for this:

vulnerhost @ example: exec demon.ex e

In the next section, we will discuss in detail about compression technologies present in Linux. Let us go!

15) File Compression

Users use compression technologies to reduce the file size. Compressed files can also be encrypted easily to store sensitive information and can be opened with only a password. Winrar is the most famous compression utility present in windows. However, Linux is open source and uses other software's which we will explain below in detail.

a) gzip

The usage of gzip is pretty simple and straightforward. All we need to do is enter the following set of commands with the file name we are trying to compress. This utility is mainly used to compress configuration files as it is good at making a set of small files into a compressed package fastly. Below are the commands:

vulnerhost @ example: gzip start beatles.mp3

file compressed successfully

b)tar

Tar is another famous compression utility tool that is very famous in Linux. It also works as gzip utility but has additional integration capabilities that gzip doesn't offer. The best thing about tar is it compressed and integrates at the same time. Thus, for hackers who are trying to exploit and compress a lot of files at the same time, this is the best choice. Below we will give some commands that will help us understand how it functions. Take a look at it:

vulnerhost @ example: tar start beatles.avi

file compressed successfully

By this, we have completed a brief overview of various file management system concepts in Linux. In the next section, we will discuss process management in detail. It is obvious by now that Linux runs by processes. Learning about the process can help us work more efficiently. Let's dive into it in detail.

Process Management System

So before going into further details, we must know what a process is and why it is important in the context of Linux and its system functions.

What is the process?

As deep it goes process is just a term that explains that the operating system is currently filtering things to process the task. Processes are dynamic and nowadays a lot of computer systems are capable of handling multiple processes. Supercomputers are said to process trillions of them.

Linux being a kernel modified system has certain free will restrictions to processes. Usually, processes in Linux consist of three cycles namely starting, running and blocking. We will explain these three states in detail as shown below.

a) Starting state

The start state, in brief, describes that the process is getting ready to be allocated by the CPU and other resources to run.

b) Running state

Running state explains that the process that we are currently dealing with is running with a certain part of resources. There are tools like task manager that can show the process monitoring.

c) Blocking state

The blocking state explains that a particular process cannot be run anymore. They may have been killed on purpose or can be crashed due to errors and warnings. Whatever the reason may be blocked state initiated the killing of the process .

Processes, Procedures, and Programs

We have already discussed processes in detail. Procedures are processes that are aligned systematically and linearly. Whereas programs are a set of procedures that can be combined to create a good system software.

There are some important characteristics of processes that hackers should be aware of. We will explain some of them below:

There are two types of processes. The first of them are mutually exclusive processes and the second one is synchronization processes.

Mutually exclusive processes are little skeptical processes because you cannot run them in the background or correlation with others. For example, like a recovery software or like using a printer to print papers.

Synchronization processes are usual processes that are that can be run in the background. You can consider a media player process for this example.

Commands for process management

Usually, the Linux system runs an abundant number of processes. When you start and boot up into the system tens of programs get initiated. It may be an antivirus or backup system software. To look at all of the processes that are running in the system you may enter ps command with parameters.

vulnerhost @ example: ps PID 34534

Some of the parameters that are present are listing, displaying process id and CPU percentage used. You can even make processes sleep, pause and give priorities. We will discuss some of them in detail here:

a) When we enter ps command we will get total processes that are running. The first line describes the names of the process and the PID that is assigned to them.

What is PID?

Like User I'd (UID) we have discussed before PID is a number that is assigned to a particular process. These are not permanent and will get collapsed when the process ends, unlike permanent UID.

b) The second line describes the initiation time of the process and the percentage of the system power it is consuming. This is important for hackers because whenever a process is taking a lot of system power one can end it easily using the kill command which we will describe in the next section.

c) The third line can be used to declare priorities for the processes. For example, when you are performing exploitation and backup at the same time you can give priority to one of them as you wish to give more system resources. This will help hackers when they are doing injections to a lot of databases or networks.

And at last, we will learn about the termination of processes. A lot of processes can decrease the system efficiency and will make Linux buggy. So, for a certain interval of time, you need to kill some processes by using the kill command. Kill command will stop all the processes at once by using killall or will just end the process you want to.

Below is a command that explains how kill works:

vulnerhost @ example: kill example process

d) You can even restart the killed processes for once using the same command. This just acts like a recycle bin for the processes.

vulnerhost @ example : restart exampleprocess

With this, we have completed our brief explanation about process management and for hackers, this is important and can help them use their

resources efficiently.

In this chapter, we have given a brief introduction to important building structures of Linux that are user management, file management, and process management. In the next chapter, we will learn about scripting and in particular about shell programming in detail. Before trying to enter into the next chapter practice the commands, we have discussed in a Linux device. The practice is the only way hackers can prosper. Let's go and learn shell now!

Chapter 3:
Basic Shell Programming

In the previous chapter, we had learned about Linux essentials for hackers in detail. In this chapter, we will dive into one of the hacker's secrets weapons that are scripting. There is quite a small difference between programmers and hackers. Programmers use scripting to build systems whereas hackers use scripting to exploit systems. Without creating their scripts, a novice hacker can never become a professional Hacker but will remain as a script kiddie who just uses other tools to crack systems. Thus, scripting knowledge is a must in the checklist for anyone trying to master hacking. To help you out this chapter will introduce a lot of bash scripting concepts with real-world coding examples. Let's have fun with some scripting now!

First of all, we need to know about the shell in detail.

What is a Shell?

A shell is that cursor you observe when you first connect to a server using a password or when you make yourself connected to a system using remote desktop tools like SSH. In other words, if you want it to look solely from a programming point of view you would be delighted because it acts just as an interpreter between the user and system just like how an operating system does .

But it just sends the input advises from the user to the Linux kernel and sends the output that is the result back to the system user.

Types of shell

There are types of shells that exist according to the Linux official documents. Out of both the first one stands for a GUI whereas the second one stands for CLI.

There are a lot of shells that have been manufactured with the Distros like Bourne shell, C shell, korne shell, and Bash shell.

Out of all different types of shells that are present bash shell is one of the most famous that ever existed. It is pre-installed in almost all the famous Linux Distros. It also acts as an interceptive language that helps the Linux

kernel understand the instructions we are giving logically. In the next section, we will look at this in detail.

How does the bash shell work?

There are two types of modes that one can work shell with. Out of which one is an interactive mode and the other is a script mode.

a) Interactive mode

In this mode, the Linux user can enter the functions or the bash code one by one and wait until the result is given. If there is an error in the middle the user cannot proceed further. It just works interactively like in an old handheld video game.

b) Script mode

In this mode first of all the bash code will be written in a text file and then will make to run the script file using the command-line interface. While using this mode the user can get all the results that he was looking for all at once. Hackers need to be more perfect in script mode because it will be easy to exploit systems fast using this way. However, programmers that are who create systems will prefer interactive mode more.

Before knowing more about the grammatical structure of the shell it would be better if we have a good overview of the advantages of the shell.

Advantages of the shell

1) It is very easy to learn inline programming languages that differ a lot in both execution and implementation.

2) It has a lot of help documents that will help the hackers to rectify the errors as soon as possible .

3) It has an added advantage because it is an explanatory language. That is, it need not be compiled before running. So, hackers can easily cross-check the code before trying to implement it on the victim's system.

Apart from this shell is also fast and works efficiently. Due to all these reasons, hackers should mandatorily learn about the implementation of the shell.

In the next section, we will give an example shell script that will help us to understand the basic grammatical structure of shell programming. Let's go!

First shell script

Let us create a shell script of the name sample.sh. create a shell script using the following command.

cat sample.sh

Location of the shell program

echo " This is very regressive"

Now we will explain this in detail. Line by line.

a) #! This represents the starting of a shell script that we are trying to write .

b) If it starts with only a hash # then it is called a comment. Comments are annotations that are used to make it easy for reference or for other users that want to look at the code. It may seem unnecessary to write comments for small shell scripts. But it is a good practice to start writing comments.

c) echo is a shell default command that lets the interpreter display the content that is written.

Here to run the shell you need to use the following command:

bash sample.sh

And then the output will appear

For this example, the output is as follows:

This is very regressive

There is also another feature that will let you run the script with additional permissions. As we discussed earlier Linux has a set of permissions and if you don't provide with necessary permissions the script may not run perfectly. So, to provide permission use the following command

In the next section, we will start discussing the debugging of the shell scripts. Debugging is one of the most important programming tasks. Even hackers need to be perfect at this because wrong debugging of code may result in bad adaptation of the task.

Debugging the shell scripts

Normally when you enter a wrong default command in the script such as shown below in the code.

vulnerhost @ example: ech 7898

This code will not run and show an error as shown below

the ech command is not found

This is exactly what debugging is. Debugging features informs the user about the errors in the script.

There are also a lot of shell debugging tools that perform the tasks we do in a command-line interface. Tools like bashdb are famous for this. In the next section, we will learn in detail about a handful number of important bash built-in commands. This section will help hackers understand the basics of shell programming effectively. Let's go !

Built-in Shell Commands

Before entering into a deep discussion about the built-in commands it is important to know that these built-in commands cannot be used as variable names. Variables are an important functionality of shell programming that helps to define things.

Linux provides a command called type that will let you cross-check whether a command is a built-in command or not.

The command works as follows:

type echo

The output is as follows:

echo is a built-in shell

Another example,

type trump

The output is as follows:

Trump is not a built-in shell

Also, remember that two dots that is (••) is used to determine the successful working or execution of the script. Here are the in-built commands we are going to discuss in detail.

1) Alias

Normally Linux commands are a little trickier to type. I suppose you have to type echo every time it may be difficult. For this reason, you can use an alias to give shortcut for a command.

Below we describe the command that needs to be entered to make alias work.

example@ linuxwar : Alias groupecho

However, it should be remembered that aliases work only until the shell environment is open. That is if the shell is exited there is no way to access it again. Also, the alias functions are stored in the bashrc directory of the user environment. Aliases are an easy way to increase productivity. Hackers use a lot of aliases to make deciphering the script a lot trickier.

2) Unalias

As you might have guessed already unalias is used to delete the alias systems that are present. By using this command, you can delete any alias command that you have created before.

Below is the command that will let us understand how it works:

example@ linuxwar : unalias groupecho

You can also use -a to delete all the aliases that are present at once. However, as we said before ending a shell environment will delete all the aliases that are present but using unalias. will help you to delete things while you are still scripting.

3) bg,fg,jobs

A lot of shellcode is done in interactive mode. Sometimes when you are trying to exploit a system you need to perform various tasks at once. These tasks are called jobs in Linux terms. So for everyone's convenience jobs are divided into two types. The first one is a foreground job where we can see the procedure that is going on. The classic example of foreground jobs is the installation of system programs. You can't handle other jobs while doing foreground jobs.

Solely, for this reason, background jobs are developed. Background jobs can help things run in the background. Hackers should be well aware of this because they are ought to work with multiple processes.

Below are the commands for the job functionalities:

example@ linuxwar : bg job 1

example@ linuxwar : fg job2

example@ linuxwar : view jobs

4) cd

Cd is the classic Linux shell command that is famous for its huge usage. When performing tasks users usually are thrown into the root directory by

the shell. Huge usage of root directory can make it scattered and messy. For this reason, Linux users use CD to change their directory and perform actions.

Below is the command for the change directory:

example@ linuxwar : cd /etc/read

5) Declaring variables

Variables are classic programming declarations. They are usually used to declare a position for the data. Variables also have a type declaration known as data types. With this, we can easily assign the type they are ought to use. There are many data types such as int, float, string. We will discuss all of these programming concepts in detail in the next section with examples.

6) Break

Scripting languages usually include Conditionals and loops that are used for repetitive and logical tasks. They can be used for both of them aligned. While doing repetitive tasks it is obvious that there should be some endpoints for better interaction and processing.

For this reason, shell language uses a statement called break that will stop the task of the logic it has provided satisfies. Break statements can also be used to print the statements using the echo command.

Here is an example for break command:

example@ linuxwar : beep.sh

```
for(i=0)
x>1
y>2

if(b>2)
break:
```

7) Continue

In loops where a break can stop the loop at once, there is a statement called continue that can help to switch the available loops. Suppose if there are five loop statements in the shellcode by using continue the loops can be interchanged. This will help to create a detailed exploiting code that can compromise systems and do multiple cross re-checks.

Below is an example of the continue statement:

example@ linuxwar : beep.sh

for(i=0)
x>1
y>2

if(b>2)
continue:

8) Eval

Linux and shellcode usually consist of a lot of arguments that need to be processed. There will be a lot of problems if strings and variables are parsed in the same way. For this reason, a command called eval is introduced in the bash shell.

eval command replaces the arguments that are present with the variables that are pointed out. You can even use eval command to parse strings into commands for execution as shown below.

example@ linuxwar : eval beep.s h

9) Exec

Execution is a process that is said to start the task. Every installation file is an execution format because it starts a new system shell in the background. Hackers should be aware of execution shells because they perform a lot of initiation and analysis tasks.

When the exec command is entered in the Linux command shell the screen or the interface that we are working on refreshes out. Exec can also be used

in script files to start dependency installations as we used in the first chapter to install kali Linux.

This is the command that explains exec command in detail:

example@ linuxwar : exec beep.sh

10) Exit

A shell window is complex and deals with a lot of tasks. However, when you complete the task it is a good practice to exit the shell. If not, the processes would still be running and may result in unnecessary system power consumption. For this, a quick command called exit has been introduced in the shell language for this exact reason.

Exit command also clears all the tasks that are present. So, make sure that everything is fine before making this happen. You can also exit individual processes or programs with a shell script.

Below is a simple command example for exit:

example@ linuxwar : exit shell.sh

11) Export

Usually when the system boots up the first shell command is created in the kernel system. This is called a parent shell. And the next ones that followed are called child shells unless the parent shell is exited or killed.

So, while working with the different numbers of shells we will deal with a lot of variables. And sometimes we may need the same variables that we used in the other shell environments. For this exporting of variables, the shell provides us with an export option.

After using the export command all the child shells can use the parent variables. However, remember that the child shells cannot use their sibling's variables that are the other child shells that are present.

Here is an example that demonstrates this process:

example@ linuxwar : export example.sh to /etc/di r

12) Kill

Killing processes is an important skill to learn for hackers. Processes have three distinct distinctions. One of them is an interaction that is a usual shell interface. The second one is a batch process where everything that needs to be applied is done in a sequential process. And the last one is monitoring the process where everything that is being done is monitored. For example, a task management system.

When you are running a bunch of these processes you may get distracted with signals that they come with. For this reason, killing processes is a good process if you find that they are unnecessary. When you are trying to exploit a system you need to kill the antivirus process that is running in the background. You can even use the killing process to stop the logging files that record your every move.

Below is the simple demonstration of kill command:

example@ linuxwar : kill process

example@ linuxwar : killall

13) Read

Read command can be used to read the bash scripts when you are performing the tasks. Or it can be even used to perform a thorough check of the shellcode that has been written. Read statement is a basic shell command and can help hackers interpret and cross-check things easily.

Here is the command with an example:

example@ linuxwar : read bash.sh

14) ulimit

Priority is one of the most underused functionalities of the shellcode. Priority can be both incremental and decrement also. By using ulimit the

hacker can increase or decrease the priority of the process.

Why prioritizing processes is necessary?

When a user is dealing with a lot of background processes that are constantly functioning this may decrease the processing speed of other processes. For this reason, administrators prioritize anti-virus software's at first level of the priority. With this method, any boot level executions can be eliminated.

However, a lot of system administrators don't take this problem seriously and make hackers exploitation easy. Below is the example command that explains how to prioritize the processes that are present.

example@ linuxwar : ulimit PID 234 5

Always remember that PID is the most necessary thing that needs to be used to prioritize.

15) Test

Shellcode consists of a lot of loop and conditional codes. Before experimenting with these repetitive tasks with a system shell it is a good practice to check them in the old shell. For this purpose, the test command is introduced.

Below is the command that takes care of the testing:

example@ linuxwar : test PID 3634

With this, we have completed an explanation about some of the built-in shell commands that are present in Linux systems. This should have given a good overview of the hacking environment to you.

In the next section, we will discuss some of the program concepts that need to be learned before writing the shell script. But before learning about it let us learn about the installation of a bash shell, the most famous shell environment.

Installation of bash environment

1) you can use wget to get the system files that are present in the server. The below command will download the files from the mirror websites and will help users install them in their system.

wget bash.com/download

2) in the next step you need to mention or input the configuration of the system that you are using. Otherwise, it may not get installed.

3) After installation, you may need to check the settings and input the default directory. Normally it is entered as the root directory. Also, the shell versions can be easily known using the help command.

With this, we have installed the bash environment and good to go to learn about some fundamentals of shell programming. Remember that these topics very much coincides with python scripting which is another good alternative scripting language for hackers.

Fundamentals of Shell Programming

The shell consists of a group of systematic instructions or commands. Let us go and learn about some of the basic components that comprise the shell language .

Variables

Variables are a piece of memory in the computer random access system that is used to store the data. As discussed before variables are the most important components of a programming language and it is often called while writing functions or templates.

Usually, there are two types of variables:

a) Local variables

b) environmental variables

Let us discuss the functionalities of these variables in detail along with few command-line examples.

a) Local variables

Local variables are the one can which can be used in a private or single environment that is these cannot be used in other shell scripts even with a reference. These types of variables are used in short shell scripts.

b) Global variables

Global variables are also known as environmental variables. These variables differ from the first one because these can be used in any shell script with a reference. For using the global variables, you need to export them to the local shell script file.

Variable naming

Variables present in the shell language should follow some varied instructions while naming. Remember that shell in-built commands like exit cannot be used for the names of variables.

Here are the instructions that need to be followed for naming a variable:

a) In shell language, variables differ from the capital and small letters.

b) A variable name should never start with a number or special character. Doing this may give an error saying that the variable name cannot be initialized.

Here are some of the various examples that can be used

love

dude

ra344

And here are some of the variable names that cannot be used

1hjsd
#feg e

Variable assignments

Variable assignments are the assignment values that are used to give a value to the variable.

It works in the following way:

variable name = variable value

You can insert a lot of data types in the variable value. Data types are the ones in which variables are defined. Some data types consist of integers, floating-point numbers, and even strings sometimes.

Here is an example that describes the assignment value:

sample = 22

Special variables

Special variables such as usnet can be used to delete a defined variable from the memory. In this, we can store the random memory management.

There are also special variables that start with a $ parameter. These variables can be used to define additional parameters that the system may require during the process execution or advanced shell analysis.

Here is the example for some of the special variable commands with a $

$dude = ' string'

The biggest advantage of using special variables is that they can be easily filtered and aliased with the help of the character that is present at the beginning of the variable.

Arrays

The array is a famous data structure that is capable of holding multiple items of elements. Programming languages use a lot of arrays because they are easy to implement unlike other data structures like trees or graphs and

also they are fast. Shell also supports arrays to input elements. In this section, we will discuss arrays in detail.

a) definition of array

Normally arrays consist of a subscript that defines the number of the element along with the name of the array.

Here is the command for an array example:

example[]

b) Giving value to an array

All elements that are present in an array can be given a value using the symbol. You can also give the value using the individual array assignment like as shown below

sample = “America”

You can insert any data type in arrays just like variables and the use of arrays can be very essential when you are dealing with loops and conditionals with complex code in it.

You can even connect both arrays to get the desired results. This scenario is shown below for your better understanding:

sample[2] = value

Constants

As we all know already that constant means something that cannot be changed. Constant values exist always and can be used to explain values like pi that have a constant numerical value. Whenever you try to change this constant variable an error or warning will appear in front of the shell that says this cannot be possible.

Here is an example of the constant command:

pi = 3.142 7

Namespaces

We already have discussed variables in detail and namespaces mean that variables that are in a defined scope. These are created for a reason that whenever users try to create variables that are of similar type a conflict always occurs and makes things difficult to organize.

For this exact reason, namespaces are invented and are used to define citations and references which can be used multiple times in a shell interface or a shell script.

Here is an example that explains in detail about this concept:

Operators

Operators are the most important regions of a scripting language. Operators can help to mix or change the variable values. As of the shell, programming goes there are a lot of operators that can be used. We will discuss some of them now in detail:

1) Arithmetic operators

Arithmetic operators deal with mathematical calculations such as addition and subtraction. These are important for programming because they can add up things and multiply variable count easily.

Here is an example command for the arithmetic operators:

```
>>> 2+3

>>> 2-1

>>> 6 * 2

>>> 7/3

>>>  4 % 7
```

2) Relational operators

Relational operators are significant operators and can be used to change things easily. They can be used to compare two things easily. Few of the relational operators are AND, OR and NOT. These relational operators can be easily implemented in any shell language code.

Here is a command-line example that deals with relational operators:

```
>>>> 2 ! = 7

>>>> x === y
```

3) Assignment operator

The assignment operator just gives the value to a variable or loop code. By using this operator one can easily assign things to the element.

Here is the example for the assignment operator:

```
x = 7
```

With this, we have completed a brief exploration of the scripting world. Shell language is a must for any hacker that is serious about his job. You can also implement these concepts with python programming for doing advanced tasks in hacking. In the next chapter, we will start with an exploration of going into a hacker's mind and how they plan things. Let us start!

Chapter 4:
Hacking Procedure

This chapter is a pathway to help you start thinking like a hacker. To stop attacks that would come frequently you should make yourself accustomed to the hacking methodology. In this chapter, we will in detail explain the phases that one needs to perform to call oneself a hacker. Let us start to the exciting world of the hacking process.

a) Foot-printing the system

b) Scanning the targets

c) Getting access to the syste m

d) Using the access for exploitation

e) Continuing the access

f) Creating backdoors in the system

In the next sections, we will in detail explain the six phases in detail. We will give out some example tools which can be used for individual phases.

1) Foot-printing the system

A good hacker would always at first try to know a lot of information about the target he is going to attack. This collection of information about the target is known as reconnaissance. Many hackers use social engineering techniques to get information from the users themselves.

A good hacker has good communication skills that can help him to manipulate things to get information about the target he is trying to attack. To say using an analogy a hacker works like a detective to track the target. He looks at all the publicly available information and will form a roadmap for a better strategy to attack.

As said before hackers manipulate individuals to perform tasks like resetting passwords or sending one-time passwords using social engineering techniques.

A lot of hackers also use Google search in-depth to get as much information about the target. This is one of the most important phases of hacking.

Kali Linux provides software such as nmap and burp to perform reconnaissance.

2) Scanning the targets

This is considered the second phase of the hacking process. In this step, we will try to scan the target and find any ports that are open to getting a successful linkage to attack. We will also use a concept called enumeration in this phase to get a lot of advanced information about the users. All this useful information can be further analyzed by hackers to get varied results.

In this phase, the attackers usually start network scanning using the available network tools like Nmap. These network tools are made available to run on systems so that the available open ports can be detected. Open ports are vulnerable and can help us to create a backdoor to the system.

However, the attackers should keep in mind that fast searching of the systems or sending a lot of packets can give a huge increase in network traffic and can make the system administrators alert. For this reason, experienced hackers extend this phase for at least a week so that they send packets slowly in such a way that the very advanced intrusion detection systems can never detect the attack that is going on.

In this phase, we can even analyze the ports to know about the operating systems and technologies that are being used. A lot of hackers after this stage will search databases like exploited to find the open vulnerabilities for the version of the software. If lucky, you can find a vulnerability that can be further used to attack the system.

Many novice hackers use automatic scanners like burp suite to detect the vulnerabilities that are present. Even though of being advanced scanners they will not accurately detect them always. They can be used for learning the basic implementation of scanning but not as a sole tool that can scan the targets. That is all about this phase and let us move on to the third one that is when we get successful access to the system.

3) Getting access to the system

This is an important step in the hacking process. After having a brief scan and obtaining information about the systems in this step hackers will start attacking the system using various methods. A good hacker always chooses his way of attacking according to the environment that he is attacking on. A novice hacker can read hundreds of books but if he cannot use this information depending on the environment and resources, he has then there is no way that the access will be cracked.

There are infinite ways of getting access to the system. Out of all these, the most classical way is to use social engineering abilities to trick the users that are present in the network area. It may be by sending an attachment to the receptionist or by getting connected to the modem of the LAN network using someone's landline phone. Getting access to the system doesn't result in successful exploiting because of less or void permissions. Some introduction detection systems can detect your access to system providers with a message.

After getting access to the system a hacker will further move to the deeper areas of the network that is to the closer areas of the root directory for full administrative privileges. Follow along with the next section to understand what one can do using the exploitation abilities.

4) Using Access to Further Exploit

After having successful access in the next stage hackers try to stay as much as the time in the system. An attacker usually tries to extend his capabilities or reach in the area and tries to acquire the root privileges which can help him get additional use cases to perform.

The main reason why hackers can get succeeded in this phase is because of bugs and vulnerabilities that are present in the web application systems or the login interfaces that the system users use. Professional hackers use hardware hacking devices like keyloggers to know passwords or secret root directories. In the next section, we will describe the most important phase of hacking in detail .

5) Continuing the Access with the Systems

Hackers are crazy and like to do things that can be repeated. When a hacker compromises a system, he tries to expand the time he spends therewith using tools called rootkits. Rootkits are hackers' tools and will delete everything or footprints that he leaves while hacking effectively. Apart from this hacker also has a fudge to get access as many systems as possible. For this character trait, they usually try to get access to /etc folder and access all the user passwords that are present. Rootkits will help the hacker in extending his connection or relation with the system in a definite way.

If the hacker is gaining money with this method, he may get accustomed to the fact that many are trying the same. So, he will make sure that the vulnerability he has found is not available to anyone. For this reason, he makes shell scripting code that will spoof the other attackers and make them not access the system. Hackers also in this process exploit as much as they can and will back up important files or sensitive information into their directory using network packeting tools and delete those traces forever.

6) Creating Backdoors in the System

After getting access to a system for a long time and understanding every pathway and directory system intelligent hackers create backdoors to continue their exploitation even if the vulnerability is patched. It is often difficult for security administrators to determine a backdoor until it causes system damage because they are often cleverly inserted into the system by hackers. In this section, we will discuss backdoor and how to make one.

How to make a backdoor:

1) First of all, look at the system as you are in-depth and try to change the system code in a way such that you can easily get access to the system for the next time.

2) Backdoor injection tools will have the ability to send the password changing information using its exploitation tools that are present.

3) Backdoors can also be created in a clumsy manner and of a lot of variables with weird names so that the programmers can never detect the original attacker. This anti-spoofing mechanism will lead to a change of the

system code which will be easily obtained by the hacker using the other backdoors he has implemented.

By this, we have given a complete tour of how a hacker's mind works. It may feel overwhelming sometimes, but hackers work it out in a hard way. So, if you want to be a professional hacker you need to create your working process or follow this straightforward methodology that has been said by many famous hackers .

In the next section, we will discuss the ethical hacking toolkit or prerequisites a hacker should be aware of before starting web hacking and network hacking for testing the quality of the systems. Follow along with this checklist and use it whenever you are starting an attack.

a) Get permissions

First of all, if you are attacking a system with a ton of security and intrusion detection systems you need to get valid permission from the system owners. Otherwise, this may land you in trouble even after using a lot of safety tools because forensic investigators of the industries are always working hard to find the traces of the attackers.

b) Don't use a lot of tools

Usually, hackers overwhelm themselves with learning a lot of tools. Tools are just a way to make the process work. You need tools to automate things but not to change your perspective on looking at things. For this exact reason try out as many as tools that are present and select the best tools that are working for you.

c) Analytics

A lot of software's now a day are providing well-reported analytics of the performance of the system for an easy understanding of the situation that is going on. Hackers should be aware of all of the technical terms dealing with analytics for better productivity and understanding of system analytics.

d) Reportin g

Usually when a test is performed penetration testers use manual skills to pitch a report of the attack. It is always best to do a manual report because in no way a machine can think about the effect of this vulnerability to the organization in a humane way. However, it is time-saving to use inbuilt features in the web application interception software to report the pen testing reports.

By this, we have completed a detailed explanation about how hackers work and even given a checklist of things that need to be done by or testers. In the next chapter, we will have a detailed introduction to web hacking and some of its tools. From the next chapter, we will be dealing with practical things so get ready to have fun with hacking. Let's go!

Chapter 5:
Web Hacking Tools

After a brief discussion of the hacking procedure in the previous chapter, we will now go a long discussion about web hacking in this chapter. As we all know in today's world both web and mobile applications are the pioneers of technology. A lot of hackers try to find loopholes and exploit them for their personal use. So, a thorough understanding of the web is necessary for security professionals and wannabe hackers.

For this reason, we will go in a practical approach to web hacking tools. We will discuss web hacking tools like Uniscan in detail. Let us enter into the world of web hacking. First of all, we will give a small introduction to the web and protocols.

What is the web?

The web is an interconnected system of networks that displays both static and dynamic information in the form of web applications nowadays.

What are the protocols?

It is just a way to transmit information between the client and the server .

Http and HTTPS are the famous protocols that are used for web communication. We will look at six tools that do different tasks.

Scanning of Webservers

Web servers are used to store information in particular. They consist of a lot of information both static and user-based information. If a hacker can get access to a web server, he can exploit any information he wants to.

Usually, hackers do a brief fingerprinting test about the webserver before attacking. This is one of the most important hacking processes that need to be done. If webserver has any potential vulnerability it would be easy to crack into it using a payload.

There will be a lot of web vulnerabilities that need to be checked on the target server. It will be time-consuming to check every one of them manually. So we can use a tool like Nikto to automate the work.Nikto is one

of the famous web hacking tools that are pre-bundled with Kali Linux. It scans a webserver using its huge database that consists of potential vulnerabilities of web servers.

Here we will describe some of the excellent features of the Nikto web server scanner.

1) Saving reports XML, HTM L

All the reports that are obtained using the automatic web scanner can be easily converted to XML and HTML formats.

2) Metasploit usage

Metasploit is a console tool that can be used to make exploits. With this tool, you can insert Metasploit exploits.

3) Mutation techniques to fish for content on web servers

There are techniques such as the mutation that can easily sniff or duplicate the content that is present in the web servers. Web servers' fish these things to display good results.

4) Subdomain guessing

This web scanners also use techniques in a way such that the subdomains that are present can be easily found out. These web scanners also sometimes web servers that are not in the scope.

5) Doing a test based on a tuning paramete r

In this tool when we encounter a vulnerability or bug, we usually test it out. The testing of it sometimes does in a varied structure called turning parameters that has a huge ability to concern the things.

Below is a brief process that takes place when an automatic scanner starts.

A) Starting nikto on a webserver

For the starting of the web scanning server, you need to have a host address and hostname along with a tuning mechanism. By using this command, you can easily detect the versions of the webserver or the programming language that has been used

Here is the command for the starting of the Nikto :

example@ linuxwar : start Nikto www.exampleweb.com

B) Running all tasks

Usually, there are a lot of hosts that we can attack. Hackers try to do things at a fast rate by attacking all of the hosts at once. For this reason, Nikto provided a tool that lets you insert the word file so that you can scan all of them at once.

Here is the command that can be used to run all tasks :

example@ linuxwar : run Nikto 193.3234.33.23

C) Running against multiple hosts

Where the prior command attacks on different servers at once with a single address in this process we will use different network addresses while attacking the host interfaces.

Here is the command that explains this process.

example@ linuxwar : run hosts host1 host2 host3

With this, we have given a complete introduction to the manual web scanners and in the next section, we will start learning about Wordpress and its vulnerabilities in detail.

Hacking a WordPress Website

Normally websites are developed from scratch using different web programming languages like PHP and javascript. But normally not every

small business can afford good web programmers to write separate code for them.

So, a lot of internet users rely on content management systems. And out of a lot that is available WordPress is the most famous. It is used in more than 25% of the websites that are present .

It offers good security features along with a lot of themes and plugins that can be used. However, WordPress is not fully safe from a few vulnerabilities. There are more chances of an XSS or CSRF vulnerability to be found. And the worst part of using WordPress is plugins and themes can be used to insert malicious code. A lot of hackers use this strategy to steal information from the WordPress servers.

To get rid of this problem, we can use a tool called WPscan to scan WordPress websites.

a) First of all, before starting the Wordpress scanner test you need to update the system so that there will be no way that any outdated vulnerabilities can be found.

b) After using the update, you can start the real start with the scanner. All you need to do is to enter the Wordpress URL that needs to be scanned.

Here is the command that needs to be used

example@ linuxwar : start wpscan www.exampleweb.com

c) In the next step, we can use the tool to get the list of users who are present is the Wordpress system. Wordpress consists of a directory of systematic users that maintain or a part of that website. For this reason, this scanner should be used as an enumeration tool whenever it is possible.

d) There are also options in the scanner that lets you brute force the system for root privilege or stop the enumeration system that is present on the website.

If you are the owner of a Wordpress website, you can use this tool to check the security of your website and if it doesn't turn out well you need to install

web server security technologies like cloud fare for an additional layer of security mechanisms.

websploit

Webservers in common consist of directories. Directories consist of files. It should be noted that not all directories are visible to everyone. These are called hidden directories and can be only visible for the root user.

Hidden directories consist of sensible configuration files that can compromise the system within a very short time. These hidden directories can also consist of private password details. We use a web split tool for making a cross-examination over the available networks.

This software is made of python and is free to download from the wget package manager system. After installing, start it using the tool name in the Linux shell.

Below is the command using apt shell command that will install the websploit framework:

example@ linuxwar : wget websploit

Web sploit consists of a lot of modules such as wireless and network modules for example. It should be remembered that web sploit works in coincidence with the Metasploit. Metasploit an exploit maker tool and can help users insert exploits in various files. We will learn about these Metasploit functionalities in detail in the last chapter.

Now we will look at the procedure where hidden directories can be discovered.

a) Before starting you can filter out the error options to wield out some of the most famous errors like 404 error not found.

b) It is already installed in Kali Linux and can be found in the module’s directory. The directory scanner module which is one of the most important modules in the websploit tool can be used to find out the directories and its syntax forms .

Below we explain some of the commands that are present in the directory scanner module which can be used to scan hidden directories:

1) Show - This command will display all the web modules that are present. When hackers deal with a lot of exploits they often get confused and mess up things. For this exact reason, web sploit consists of inbuilt modules that are used to find vulnerable directories. And this show command can help us find some of these for us.

example@ linuxwar : show websploit www.exampleweb.com

2) Verbosity

Verbosity is a simple statement that lets us set the number of results that can appear on the Linux shell. When we start searching hidden directories usually a lot are found and can make things confusing. To get away from this problem we can use verbosity command to display the custom directories. There is also an option that will make us look at all the things in a static form.

3) And the next important command is RUN which helps us to run the exploits with the websites we desire. We can also use this network transmitting system.

example@ linuxwar : run websploit www.exampleweb.co m

When you follow these commands with perfection then there are huge chances of websites getting compromised.

Cloud flare web sploit

We even have a second set of webs sploit commands that can be used to resolve cloud technology. Cloud fire is one of the most important security layers for websites and is now maintaining and securing the utmost two million websites from dangerous attacks.

First of all, to use the cloud flare module you need to find for it in the web sploit modules list as shown below:

example@ linuxwar : websploit select cloudfare www.exampleweb.com

Now after getting the interface you can install cloud flare in any of the sample websites to check whether it works or not. The working process of cloud fare deals with changing the original network address of the system to one of its servers. Thus, if there are any injection attacks or brute force attacks it would stop or ban that address at once. Cloud fare acts like an intrusion detection system for the websites at a very low cost.

Why the cloud flare is still easy to bypass ?

As we said before it just spoofs the attacker with an IP address. Many hackers started collecting hundreds of Cloudflare addresses and started to abandon them whenever they attack. Some tricks can be still used like using this cloud flare scanner to find all the IP addresses that the website hosts with.

In the next, we will learn about uniscan one of the most important web fingerprinting tools.

Uniscan

Uniscan is used normally for the remote code execution or remote file insertion of the vulnerability scanners. It also can perform network commands like ping, traceroute, software detection.

Here is the command that searches to determine the operating system using the uniscan

example@ linuxwar : uniscan select domain

Uniscan also provides a tool like NMap open port detection. It specifically checks the os version of the server and scans the service.

Uniscan also provides a way to report the scanning reports using the export options as shown below

example@ linuxwar : uniscan export domain to domai n

With this, we have completed a brief introduction to uniscan and would now leave for the next section which will deal with the listing of subdirectories.

Sublist3r

Websites consist of a lot of subdomains. Usually, domains that are in the scope can be used easily to manipulate using applications like burp suite. For suppose, Gmail has a lot of subdomains and if we can find access to one of these, we can easily manipulate the whole website.

This is the reason why subdomain enumeration is one of the most important concepts hackers should learn. We have a lot of tools that will help us find subdomains. In this section, we will use a sublist3r to do the task.

Sublist3r is not present in the Kali Linux tools list. For this reason, we need to install it from the git repository. Below we explain how to install sublist3r. You can use this method to install any third-party applications that are not available in the Kali Linux repository

Installing sublist3r

1) Select the directory you need the tool should be installed using the cd command.

example@ linuxwar : cd install sublist3 r

2) GitHub is an online repository that makes things easy for programmers when cloning the application. It is different from package managers because one can actively contribute to the application using git console.

Here is the command that can be used to install sublist3r

git clone (URL)

3) Now every git folder has a requirements file that will help the system to install the other tools that are needed to be installed to make the software work. These are called dependencies. Hackers should have a good understanding of this because of many encounter errors.

We will let you understand dependencies with a perfect example. Imagine that you are trying to install a java application like Android studio. If there is no java installed in the system you cannot install and use the Android studio. That is what a dependency stands for.

In the next section, we will explain the working of the sublist3r. Follow along to learn about it in detail.

Starting the sublist3r tool

Run the following command to start the subdomain searching tool .

example@ linuxwar : run sublist3r

We will get a lot of options that can be chosen from this tool. We will even have a help section when entered -h that explains the commands that can be used.

Here are the functionalities that this tool can perform in detail:

a) Domain -d

You can enter this parameter to insert the domain you want to find subdomains for.

b) Brute force

This parameter can be used to start a brute force attack using a lot of domain lists that are entered in a text file. This can be particularly used when you have a lot of domains to test.

c) Ports

Ports are the functionalities that can be used for easy sub domain referencing. This parameter can help the user find vulnerable subdomains using the open ports .

Apart from these basic parameters, you can use output functionality to get the results into a text file from a shell interface.

We will now give some example commands that will let you understand subdomain enumeration in detail:

1) Python sublist3r.py -d amazon.com

This command will start an execution that will display all the subdomains that are present after using the

2) You can use this command to brute force the first 100 domains that are present in the text file.

python sublist3r text.txt

Through these simple techniques, you can find subdomains. After finding the subdomains list you can use techniques like a recon to further dig a lot of information about them.

While trying to hack a web application, hackers should prefer this methodology to easily catch the easy way to get onto the application. That's it we have completed a brief explanation about the available web application tools and in the next section, we will start a detailed explanation about network tools. Let's go!

Chapter 6:
Network Hacking Tools

The most important and complex to handle while hacking is dealing with networks. Network in layman terms is just a system of interconnected networks. The Internet is the biggest network that has changed lives. Companies and a lot of industries run with interconnected networks. Whenever a hacker gets successful in entering a network, he will try to hack the other subnetworks too. We will in this chapter learn about a lot of network concepts and commands that will make hackers crack the networks easily. Let us go!

What is a Network?

A network is a group of systems that are bound to work together or in a total sense used to exchange information from each of them. For the appraisal of this definition, we use a lot of technical devices that are used to exchange information. Some of these devices are routers, modems, antennas, wires and even groups of satellites that continuously track geographic variants for the working of GPS.

Kali Linux consists of a lot of network tools that can be used to connect to an embedded system or for being used as a server maintenance tool. We even have used the network card configuration and wireless integration during the installation of the operating system. In the next section, we will discuss ifconfig one of the most important network tools that can be used to learn about the network details that we are dealing with. All we need to do is to send packet signals and we will learn about in detail.

ifconfig

ifconfig is usually the default command hackers use to know about the network information. It will display a lot of information like Ethernet address, physical MAC address, IP address, and even the network mask.

There is also a section that describes the number of packets that are released and received. Another line describes the collisions of network packets. While some of them may be useless for hackers they can be used to determine the network strength of the system they are trying to exploit.

Below is a command and output that shows about the ifconfig in detail :

ifconfig

Here are few things that ifconfig command can do:

a) ifconfig can be used to specify the IP address with a netmask. This command can be used when performing a wireless packet injection using the network routers.

Here is the command

example@ linuxwar : ifconfig netmask 223.2.1.2

b) ifconfig on a whole can also be used to determine the broadcast address that the system is on. It can be further manipulated with the deviation of the netmask.

Here is the command

example@ linuxwar : ifconfig broadcast 212.11.1.1

c) ifconfig can be used to end the network devices. Just like killing the processes sometimes it becomes necessary for hackers to end the network drivers or devices they are connected to get rid of sniffing or leaving any traces.

Here is the command for switching off the Ethernet driver

example@ linuxwar : ifconfig eth0 212.1.1.1

In this way, we can use ifconfig to switch on or off the network drivers. However, ifconfig often only helps us to understand the information that is already available and will not help us to manipulate any network information. For this, we have to use advanced sniffing or clickjacking software's where we can manipulate the configuration files and resources that are present. In the next section, we will discuss some of these complex network manipulation tasks in detail. Follow along!

Manipulating the network configuration file

Usually, when we run an ifconfig command everything that is acquired or obtained by sending network packets is stored in a configuration file. Configuration files are special types of files they maintain a certain order so that the system can detect it. Not every file need not be a configuration file.

However, when we boot or switch off the system everything that is the input protocol information will be deleted forever. So, instead of rekindling the entire procedure other time it is best to write into a configuration file which can be easily found in the, etc folder.

In the configuration file enter the network address and network mask in detail. It should be important because the manipulated network configuration file can be used to do a fingerprinting about the system.

Routing and gateway settings

A network card consists of a gateway and also consists of a routing protocol that looks at how things are functioned here. So before starting the network hacking details, we will learn what a gateway is.

Gateway is like an entrance to the network system that we are dealing with. We usually have computers and these are called hosts and they have an immediate physical address known as MAC address. Gateway works as a gate or checking point that checks the network packets and sends them to the servers the user is trying to contact. It does the same when a response is received. For this reason, gateways are the most important part of network systems.

Why are gateways important for hackers?

When you are willing to use a VPN or proxy server then you need to enter the gateway. Otherwise, the packets will just pass through the default router gateway.

Here are commands that can be used to change the default gateway:

example@ linuxwar : gateway = 214.133.1.1

In the next section we will discuss routing and why is important for the better performance of the network systems.

What is Routing?

Routing is a network protocol system that sends packets in a definite way so that the time that takes to transfer the packets reduces. When a lot of packets are being sent it usually takes a lot of time and can result in less network bandwidth. For this reason, efficient routing protocols are being developed for better transportation of packets.

For this, we need to create a routing table and should calculate for the flags that may be present in the network system.

In the next section, we will discuss network hosts in detail. Follow along!

1) /etc/hosts

In the early worlds of computing people used to enter the IP address to send or receive any information from the system. But after the successful introduction of the host system, it is no longer used. The host is just a text interpretation of the manual input address.

There is a host file that is present in the etc folder where all the DNS profiles can be added. Nowadays this is usually made by google DNS server which consists of every web address. Also, hosts can be used to know about the number of packets transmitted.

We now discuss here the hosts' directory and give some commands:

2) Search hosts

In this option, we can use a command to search any name server that is present in the DNS profile. When we use this tool, we will get the input address of the website we are searching for. This will help us to find other important fingerprinting technologies.

Below is the command to find the hosts that are present:

example@ linuxwar : search host 23.1.12.1

3) Statistics

Statistics that are present can be used to display the host information. Hosts are very important for a successful connection to the server. You can use various tools that can help us deal with these functionalities.

Apart from using it is a network tool hosts are very favorite of software crackers. Usually, when the software is reverse engineered and cracked by hackers it will have an ability to connect to the internet and redo everything. This is where hackers make use of host profiles. They will add a line of code in the hosts' files to redirect the software to the localhost. With this, if there is any interruption of services that will be regained.

In the next section, we will discuss one of the most important and easily used network tools that are ping.

4) Ping tool

Hackers often when trying to deal with networks after getting access to the system will try to do work on the ping tool. This is because the ping tool sends an SMP request to the tool and will let the packets received. When you enter the ping command in the Linux terminal the tool will start analyzing the packets and will display the response packets that are receiving. For this reason, the ping tool is often considered the initiation of networking tools that need to be mastered.

Here is a command that explains the working of the ping command:

ping www.hackkali.com

a) when we enter this command, the network will start analyzing the packets that are flowing and will give the following output.

128 packets received 192.674.34.2
64 packets received 192.675.34.2
.....
.....

A ping is a command tool that displays the content in the shell so there is no way you can stop the tool unless you close the shell window. You can further check the ping statistics using the following command and can also be exported into a text file.

192.674.34.2 ping statistics

With this command, all of the statistics that deals with the domain will be displayed. In the next section, we will explain about traceroute another important network tool that can trace where the packet has traveled.

5) Traceroute

Networks are used to send information from one system to another that is present in the same network or to other systems that are present elsewhere. However, have you ever wondered how this information will be passed on?

The transportation of the information is done using packets and routers. Packets are basic traffic that exchanges the information. Traceroute in the basic idea will inspect these packets and will trace whatever they are doing. This is an advanced concept that deals with things that networks are ought to do. Check yourself about the packets that are passing through using the traceroute command.

example@ linuxwar : traceroute 192.23.2.1

With this, we have completed a brief explanation of the network hacking tools and in the next chapter, we will have a brief explanation about hacking hierarchies.

Chapter 7:
Web Hierarchies and Cybersecurity Ethics

In today’s world, it has become very difficult for people who are more concerned about their privacy and security. Hackers tend to work in different ways to exploit systems by using known vulnerabilities. A lot of information is available for some of the most famous vulnerabilities like XSS, SQL injections and CSRF vulnerabilities to understand and exploit them using one's code.

Often hackers use automatic vulnerability scanners to find loops in the web application or a network system. However, there are a lot of hierarchies that are divided based on the motives of the attacker and their goals. In this chapter, we will help you understand all the hierarchies that are present. Follow along to know about hacking hierarchies in detail. This is a bit of theoretical subject so try to think about them in your own words for an easy understanding.

Why Do Hackers Fit Into Hierarchies?

Often people and security researchers face a lot of people who are trying to attack corporate and personal systems. Many times, white hat hackers who in common terms try to protect the systems decide the danger of an attack with the attacker steps and logfiles that they left while trying to exploit the system. With the help of this information, they tend to decide the motivation behind the attack. For better forensic analysis security researchers from a long time are using hierarchies to organize the level of attack. Hierarchies are important for a better understanding and analysis of the attack that took place.

There are seven types of hierarchies as explained below. We will describe each of them in detail in the next section. Follow along for a detailed explanation of each of the.

Hierarchy 1: Script kiddies

Script kiddies are usually the high number of users who call themselves hackers but have very little technical and scripting knowledge. This hierarchy of hackers are often not so talented and can never break into a system without a step by step procedure or explanation that is often found

in hacking websites and forums or social networking groups like YouTube and Facebook.

But they should not be taken of less importance because there is a high chance of script kiddies exploiting system if it consists of outdated technology that is viable to a lot of vulnerabilities.

Hierarchy 2: A group of Novice hackers

Novice here synonyms to quite good. A lot of hackers spend their time in hacking or cracking forums trying to exchange their knowledge and exploiting things together. This hierarchy of hackers due to it for fun. They often try to crack websites like Netflix and try to sell it to them for cheap prices. These novice hackers use cracking tools with brute force ability to constantly login with a bunch of usernames and passwords they have collected using a SQL injection vulnerability.

As we talked about the process you might have understood that this group of hackers purely depends on luck. They cannot exploit usernames or passwords that they don't find. For this reason, people should use secure strong passwords that cannot be exploited by this hierarchy of hackers.

Hierarchy 3: Hacktivists

Hacktivists are advanced hackers who use their knowledge and hacking skills to give sensitive or shocking information to the world that is otherwise is not possible to know. Anonymous a famous hacking group falls under this category. They try to give information about the government or shocking emails that will make the population understand the problems the world is facing. Wiki leaks are also one of them and have shaken the world with their leaked emails of various country presidents. However, this hierarchy of hackers are very rare and are always with a motive that cannot be judged due to its sensitivity.

Hierarchy 4: Black hat hackers

Black hat hackers are the evilest group of the hierarchies and use their skills to exploit normal users. They use a lot of social engineering techniques to

lurk the user to give their sensitive information like passwords and credit card numbers to them. They often rely on a lot of malware tools like Trojans, keyloggers to enter into the user system and acquire the information. They use a lot of techniques and strategies to exploit systems.

Black hat hackers are highly professional and advanced hackers who are very difficult to get caught because they use their sock proxies that can go undetected. As far as they make a mistake it is highly impossible to know their identity. And the worst thing is there are more than 5% of hackers who solely sell the stolen credit cards in the dark web. Security analysts and web programmers should be aware of the technologies and tools these hierarchy users use and should develop intrusion detection systems in a way that they will be stopped or even caught.

Hierarchy 5: Criminal gangs

These are a group of black hat hackers who work in a group. This makes things worse than before. Criminal gangs use a lot of hacking resources to originate cybercriminal gangs. They try to smuggle narcotics, guns and other illegal stuff in the dark web. They are very professional hackers who could spoof the packages that are being sent as genuine using their hacking techniques. These criminal gangs are also responsible for huge black markets on the dark web. Criminal gangs work so efficiently that it is even difficult for the FBI to catch their whereabouts.

Hierarchy 6: State-sponsored hackers

Hackers are not ethical people in general but are patriots most of the time. You might have already seen some middle eastern hackers writing their country slogans after hacking celebrity social networking accounts. A lot of countries mainly Russia and China hire a lot of professional hackers to hack into other country databases where legal or sensitive information can be found.

These hackers can get into any traffic signals or webcams and can control them. They are very anonymous people that work for the benefits of the country and even indulge in cyber warfare with criminal gangs on the dark web. Hacking is a strange and gloomy world if you want to look deep into

it. A lot of people fight around the corners of the internet to get access to the farthest point of the internet. This is where everything is present, and these hierarchy hackers work for it.

Hierarchy 7: Automated tools (Bots that spread an exploit)

It is to be noted this hierarchy doesn't consist of humans. A lot of systems cannot be accessed without personal access to the system. For this reason, some highly professional hackers nowadays are creating very small worms (like robots) that can automate or think itself and access the system. These are very dangerous hierarchy tools that are used to attack nuclear power stations and other highly secured places. It is very complex to understand this hierarchy right now because less information is available in the public domain but there are reports that worms with malicious malware are the next huge weapon for hackers.

By this, we have completed a brief and thorough explanation about hacking hierarchies in detail. In the next section, we will discuss in detail about cyber ethics and some of the famous malware attacks in detail.

Cybersecurity Ethics

The cybersecurity field has expanded its horizon by leaps and bounds. As new software systems started to develop there came a lot of importance for the cybersecurity department due to potential vulnerabilities that may make the companies lose money by some anonymous attacks. For this reason, a lot of industries started offering security solutions for example like Cisco security solutions to protect their clients from potentially dangerous attacks.

For this reason, they started to recruit a lot of system engineers who have sound knowledge of database security. Decades later now cybersecurity is one of the pioneering fields in computer technology. There are a lot of software and tools that provide automated solutions. And there is still a lot of necessity of manual labor.

In this chapter, we will discuss the ethics of cybersecurity in detail. First of all, recite a simple fact that everyone who deals with security is called a

hacker. We will now discuss the three most famous types of hackers that are distinguished for a better understanding.

a) White hat hackers

b) Black hat hackers

c) Grey hat hackers

That's it. We will now discuss each of them in detail.

White hat hackers

These are the type of hackers who try to protect databases and servers from being exploited by the bad boys. They usually try to find vulnerabilities in the system and try to fix them as soon as possible. They use different hacking and network tools to track the resources and systems they are dealing with. White hat hackers are a role model and will inspire a lot of novice hackers to walk on the good side.

Black hat hackers

These are the type of hackers who try to exploit systems using different techniques. These guys are usually professional and will follow a lot of precautions to not get caught. They create their defective mechanisms like trojans and worms to exploit systems. Black hat hackers use services like VPN, Tor to hide their identity.

Grey hat hackers

Grey hat hackers are a special category of hackers who are not into the bad game but are willing to help the bad guys to play. These types of hackers usually have fun in detecting vulnerabilities. When they detect a vulnerability, they will sell that information in a dark web which will further be abused by many black hat hackers. However, these people will not exploit the system by their won due to various reasons.

Through this explanation, we have got a deep understanding of the different types of hackers. In the next section, we will have a brief description about penetration testing.

What is Penetration Testing?

Penetration testing is a technical name that lets users find loopholes in the system to fix them. Usually, security researchers perform penetration testing attacks to help the company.

Penetration testing tool kit

Kali Linux provides a lot of tools for penetration testing in its resources. As explained before there are a lot of stages that need to be performed to confirm that the system is completely safe from any attacks. For this purpose, you need to perform all the penetration tasks like scanning, vulnerability testing, mock exploiting. It is better and safe if you can maintain a backup procedure for all the data that is present .

In the next section, we will discuss some of the famous cyber-attacks that had happened. This is theoretical information and can help you understand the scope of hacking and threats hacking may possess to both economic and safety prospects.

What are cyber-attacks?

The Internet is a weird and dangerous place. A lot of people try to find the real place of the internet which is buried deep beneath the dark web. The dark web is a place that cannot be accessed by normal browsers like chrome in any way. To access dark web websites, you need to use the TOR browser, about which we have discussed already.

Dark web consists of a lot of websites that sell stolen information by hackers. This is where hackers who have found a loophole in a famous system will try to take advantage of it by selling it to other hackers. A lot of cyberattacks happen for two reasons. Out of them, one is money and the other is cyberwar between countries.

You might have heard about ransomware virus that had spread into millions of systems a few years back. This is a perfect example of a cyber-attack. Here are the top three cyberattacks that have happened in the last decade or so.

1) Yahoo hacking target

Yahoo is a good pioneer technological company that deals with search engines and services like mail and news. However, whatever the reason maybe it has been a victim to one of the largest cyberattacks that have ever been.

In three years, yahoo user accounts have been compromised for more than three times and millions of accounts were just available in the dark market for sale.

This is the single reason why the company has put on sale and was bought for less than what it deserves. Seems like hackers have taken advantage of a broken hash file structure that is present is the yahoo databases.

2) Equifax cyber attack

Equifax is a credit card issuing company and has suffered a major breach where hundreds of thousands of credit card numbers have been stolen. The hacker has tried to sell these credit cards in dark web markets and the company didn't state unless the vulnerability that is responsible for the breach has been fixed.

3) Ransomware attack

Just a couple of years back ransomware virus has infected a lot of businesses in the European Union. This virus has used an existing vulnerability that has been patched by the windows way before the attack. But for varied reasons, many businesses didn't update their systems and got locked out to the ransomware system virus which threats that the data will be deleted unless money is paid within a set of timeframes. This is the reason why system administrators should keep updated with the system and should get ready to follow journals for pointing out existing vulnerabilities.

Apart from these China, Russia, and America usually has a lot of cyberattacks going on every second for various reasons. Cybersecurity space has become crowded and a lot of individuals, businesses, corporates, and governments are trying to get control over them. It is obvious one should be aware of the misjudging or threats of hacking and should try to learn and follow cybersecurity ethics. That is all about this chapter. In the next chapter, we will discuss hiding the attacker's location and information using TOR and VPN. Let us go!

Chapter 8:
TOR & VPN in Linux

Hacking is a risky task. Novice hackers are always easy to get caught if they don't use varied precautions. Almost every professional hacker hides his original identity with different tools or his written code to get detected. Several tools are included with Kali Linux to help you stay anonymous while attacking systems. However, remember the hard fact that tour government, Isp providers and sometimes even the VPN services that you are using can track you. As you get significant experience in the hacking field you will develop various strategies that can help you spoof your identity with ease and conviction. For now, follow along this chapter to know in detail about TOR, VPN and how to spoof your MAC address in detail. We will give a set of commands and examples in detail for your better understanding of the topic.

How to Use the TOR Network in Kali Linux?

Normally if you want to install Tor in any of the windows or Mac systems you need to install the tor browser bundle that will start an anonymous server and routes all the traffic through it.

But in the Linux terminals, we will get an opportunity to start the tor routing server from the command line itself.

Use the following command to start tor bundle service:

start tor

First of all before connecting to the tor server check your IP address. Because you need to recheck whether the Tor server is working or not look at the IP address.

What is TOR?

TOR is a chain of interconnected computers that are present from the different locations around the world that are used to spoof the location of the system. Few network engineers who have the motivation to make the internet secure and stop getting monitored by government agencies has started this TOR project.

How TOR works?

Whenever a request is sent from the TOR service or ToR browser the request is sent to one of the TOR networks. You might have confused now because this is how proxies work and how it is different from it and how it can be different from it?

This is where TOR achieved something no one ever did. TOR service, unlike proxies, doesn't route through one network but a bunch of interrelated anonymous networks thus making the detection very difficult.

Tor browser bundle

If you are not skeptical about the tor service, you can use the tor browser bundle which sends the request through the service. Every hacker should learn using the TOR browser because it makes detection very very difficult and also users can look at the dark web and its important contribution to the hackers.

What is the dark web?

The dark web is like the down layer of the internet that is often untouched and undiscovered by normal search engines and browsers due to various reasons. TOR websites have a .onion link and are very statically built websites.

The dark web is famous for black hat hackers because everything that goes around there is illegal and theft. We did not advise you to do anything illegal, but it would be a good case study to look at the people who ethical hackers will be fighting on.

Are there any things better than TOR?

As of now, the TOR browser bundle is the best way to hide your identity or spoof things. But virtual private networks (VPN) can also be used to maintain anonymity and from a different wide variety of servers. We will talk about virtual private networks in detail in the next section .

How to use VPN in Linux?

Before discussing the installation process, we will, in brief, discuss the VPN technology.

Virtual private networks are at first used in private industries and government institutions to maintain a group of people even when they are off-site. However, people have found a way to share their servers with other people without transferring any information about the sender and receiver. This made VPN popular and useful for a lot of users and mostly for hackers because of its ability to hide the location of the user.

Why is VPN useful to hackers?

a) Virtual private networks hide the network address of your system and link it with another so that you will have anonymity.

b) Virtual private networks can help you get past through a lot of firewalls and intrusion detection systems that are built-in industries.

c) Virtual private networks not only help you hide your information but also helps to make sure that all of your network information that is being transmitted is encrypted. Not sure how this works? Let us give an example.

Imagine that you are using a public WiFi network where a lot of other users are connected. Not all WiFi dongles have WPA2 advanced security configuration so your network packets that consist of a lot of sensitive information including your passwords, credit cards can be easily compromised by someone who is on the same network and using wireless networks sniffing tools like air crack.

Due to these complications, virtual private networks are recommended to be used when you are not an in-home network. Now in the next section, we will look at how to install any VPN in any Linux systems.

Installation of VPN in Kali Linux

1) First of all, install the tools or certificates that need to be used to install the VPN. Normally a CA certificate should be downloaded and uploaded to

the system. This certificate consists of a hash code that will let us connect to the server with an anonymous linkage.

2) In the next step, we need to enter the network manager to start the VPN connection. When you started a new VPN connection enter the gateway details that are provided by the VPN provider.

3) In the next step, you can give the authentication details that briefly consist of your credentials in the VPN service provider application or website.

With this, we have completed the starting of a virtual private network. There are a lot of good VPN service providers like open VPN, Cisco VPN, hide my ass. Choose the one which gives a huge number of servers and security. VPN is a must of every hacker nowadays due to increased surveillance from governments all around the world. You can also use a VPN for peer to peer network communications such as downloading torrent files and seeding them. That's it in the next section we will learn about changing the physical address of the system that is MAC address.

How to change the MAC address using Linux?

First of all, we need to know about system addresses. Every system that is ever manufactured has a serial number to it called a MAC address. Every system has a unique address by default, unlike the network address that often interchanges between the systems that are connected.

Usually, during network surveillance, IP addresses will help the attacker find the location of the system which may be easily spoofed by Virtual private networks and almost every company uses a VPN for this purpose.

However, a lot of people don't change their system address due to various reasons. First of all, the warranty will become void and any necessary help will be not provided by most of the industry providers. For this reason, a lot of users will not change it.

However, hackers for whom anonymity is their important consideration should be well aware of the procedure that can make spoof a system address. Below we will describe the method in detail.

What is spoofing?

Spoofing is just faking the receiver that the address is different. This on a whole will not change the system specifications but will show some random or manually inputted information to the attacker or forensic specialists.

Spoofing a MAC address

a) First of all, before trying to spoof the address you need to know what your MAC address is. It differs from system to system. You can even find it in the warranty booklet. But as we are geeks, we will use the Linux command shell to find the MAC address of the system we are working on. Follow along!

b) First of all, enter into the Linux command shell and inter the ifconfig tool. After successfully starting the ifconfig module enter the below command

ifconfig mac

This will display an output that shows you the MAC address of your present system. It will also give the manufacturer details along with the network packet structure it uses.

c) Now you need to install a tool called Mac changer that can spoof the system address. Install the Macchanger using the following dependency injection command

wget macchanger

d) After installing the Mac changer, you are good to go to use the commands that will change the system address both randomly and manually. We will discuss both of these functionalities in detail below.

e) Random MAC address just changes your MAC address into a random number that is per listed in the code.

f) Custom MAC address will, however, make you change to ur desired MAC address using the following command.

root @ host : change MAC to as:23:1w:2w:3e

Whenever you use the changed MAC address there is very little chance of being identified or traced. By this, we have explained all the topics that can help you deal with maintaining your anonymity while hacking. Before ending this chapter, we will have a brief explanation about proxies for a better understanding of this topic.

What are proxies?

Proxies just act like a middle man between the client and the server. And moreover, proxies can be used to send data through the modem to an anonymous server. Proxies are available free of cost on the internet, but they are of no use because of their structure and configuration?

What proxies should hackers use?

Hackers should get acquainted with SOCK5 proxies which can be used for a limited period. They are not available for free but can be purchased in proxy markets or you can even make one server. SOCK5 proxies are secure and even have a location configuration such that there will be less blacklisting of the proxies.

There are also proxies called upstream proxies that can be used in brute-forcing web application software like the burp suite.

With this, we have completed a brief introduction to everything that relates to the security of the hacker. In the next chapter, we will discuss some of the famous Linux tools in detail. Let us go!

Chapter 9:
Advanced Kali Linux Hacking Tools

This chapter is a final implementation of all the concepts we have learned in this book. There are a lot of hacking tools available now. Linux Distro Kali Linux provides approximately 350 tools from various categories. We will in this chapter discuss some of the famous hacking tools that are used for web hacking, network hacking, and password cracking. Remember that hacking tools are just an easy interface that will let us complete our work fastly and efficiently. Good hackers don't always rely on tools but create their tools to exploit the system as soon as can. However, hacking tools are better options for security testing and other tasks like fingerprinting or brute-forcing. We will now discuss some of the hacking tools in detail. Let's get started!

Burp Suite

In today's world web applications are the most attacked ones by hackers. They are somewhat easy to break into when compared to network systems and can be easily manipulated to cash out easy money or sensitive information.

Hackers use brute force tools like Hydra that are available in Linux to brute force (that is to send a lot of requests automatically) login pages. Security researchers should test web applications in depth to clear any vulnerabilities that are present .

To make this vulnerability testing process smooth and easy port swigger has created a tool named as burp suite that has been white hat hackers favorite since then. In this section, we will discuss the components of the burp suite in detail. Let us start!

Note: Burp Suite is a tool that can be used by hackers to manipulate pages. So always take the permission of the website owners that you are trying to experiment on. Otherwise, you can use sample test websites that are available either online or as Linux iso.

How does the burp suite work?

Burp suite uses a proxy mechanism to send requests as a middle man unlike acting as a client. This process will help the software to analyze every protocol it uses and every request it sends and the subsequent response it

receives. By using this information burp suite can be used to send repeated or manipulated requests that can help the users to understand the process that is going on.

In this chapter, we will discuss in more of a practical way about the burp suite in detail. We will look at a scenario to understand the tool we are learning.

Practical scenario:

We have a login page for our website www.amazonkali.com that uses https protocol. Our novice hackers' task is to log in to the website using the payloads that are present. Let us do it!

Try to do this assignment by yourself for the first time while using the tool and if you are unable to get the task done follow the below section to understand the process that goes on.

Solution or strategy for the task:

First of all, after the successful installation of the burp suite, you need to enter the CA certificate in the browser options to make burp suite work on https websites. This is necessary to be done because https use an encrypted protocol that can be only read by the proxy interpretation tools when a CA certificate is installed. For the installation of the certificate, you need to start a proxy first.

Follow along with the instructions briefly:

a) Open the burp suite proxy tab and enter the proxy 127.0.0.1 as an interception address. After entering the details, you need a select a browser preferably Mozilla Firefox because chrome spends a lot of computer usage and being a hacker coordination of processes is an important thing for smoother results.

b) In the Mozilla Firefox, proxy settings enter the same interception address and start the proxy server using the intercept on/off button. With this, every request or response will first go through the burp suite proxy server and will get recorded.

With this, we have set up the burp suite with the browser and we are now all set to exploit the www.amazonkali.com website. Follow along for the procedure.

Hacking a login page using burp suite:

1) In the first step start the intercepting proxy and enter the URL address in your browser. When you press the enter button you will see a request pop up in the burp suite proxy tab. You can look at the GET request and understand that the browser is requesting a burp suite to accept this request and send it to the original server. When this is being done our burp, console stores the request and response information.

2) A website consists of a lot of subdomains and this may become a problem when we are trying to intercept using a proxy. A lot of unnecessary requests will be processed and will make the console tab chaotic. For this reason, the burp suite gives a scoping tool that will let us select the main domain for testing purposes. All the out of scope sub-domains will be filtered and will not be sent through the proxy service.

3) Now accept all the requests that have been sent by the browser. Now go and look back at the browser and our homepage will be displayed. This homepage has a login form and we need to brute force this with payloads using brute force hoping for a successful cracking.

4) Look at the requests that have been monitored and you can observe that all are GET requests which are used by the client to let the server know that the system is asking information. We will now use intruder a tool in the burp suite to brute force payloads to the login form.

What is an intruder?

Intruder in common words is an anonymous person that enters the house without any permission or for theft. This is the exact way how an intruder works. When we use this tool burp suite sends requests fastly and anonymously so that the system won't detect it as malicious.

Step by step procedure

1) Now enter some fake data in both username and password fields and send it through the proxy server. With this procedure, all the data will be sent and the request will be of POST category. The post is an HTML request category where the arguments are sent along with the request.

2) Now select all the post requests and send them to the intruder tab. In the intruder tab, you can select the entered username and password arguments using a dollar sign. After the arguments becoming highlighted user can select the payload procedure type. There are a lot of procedures like a single hammer, cluster bomb, and pitcher fork.

We will explain these three terms in detail here:

a) single hammer

Here only one argument is selected and is brute-forced using the payload. By using the single hammer one can easily

b) cluster bomb

Cluster bomb highlights two arguments in a way that the arguments are sent.

c) pitcher fork

Pitcher fork also highlights two arguments but uses the payload in a way that the arguments are given in a co-linear way.

After selecting the type of attack, we can go into the next interface and select the payload that we need to send into the login page. Before knowing about different types of payloads we need to first of all what a payload means.

What is a payload?

The payload is a systematic collection of commands or syntactic statements that can be used to exploit or crash the system while brute-forcing.

Payloads present in the burp suite

a) SQL injection payloads

Burp suite consists of SQL injection payloads that can easily crash an injected database to enter into the system.

b) XSS payloads

XSS is one of the most frequent web application vulnerabilities and can be easily found out using the XSS commands present in the burp suite.

c) custom payloads

With this option, you can create payloads that are random with the alphabets and numbers. With this payload, you can create a lot of complex crypto passwords that can be used to crack advanced systems.

d) Runtime payload

Usually, payloads are generated but if there are a lot of payloads that need to be inserted you can use this payload to insert the list in the runtime memory for faster execution.

e) recursive payloads

Recursion is a system that sends the payloads in a varied significant signature process. Although being insignificant of nature these payloads are used to effect systems like cloud flare to get access.

After entering the required payloads enter into the next section to start the brute-forcing process.

In the next interface select the numbers of threads and the speed on which the payloads should be sent. Also, you can insert upstream proxies that will change in random so that even if the intrusion detection systems block the access you can access using other proxies. You can even use the tor system proxies as upstream proxies for additional security and to avoid detection.

Now click the start button and you will get an interface that will let you analyze the hacking process. If the login was successful you will get 200 success information in the status bar. If the login was not successful you

will get a 404 or 303 error. In this way, you can easily hack a login web page with a burp suite.

In the next section, we will discuss Metasploit one of the most famous exploit binding software that can help us to spoof any system.

Metasploit

Metasploit is one of the most important hacking tools due to its huge number of features that can be used on a target. The best thing about Metasploit is although being a tool that offers a lot of features and it is available as open-source. It matters because the burp suite which we discussed before has a premium license and restricts some of its features like automatic scanning for free users. Just like burp suite Metasploit also offers a professional license .

The major difference between both versions is that in the free version you can only use 32 hosts at a time. This may be quite difficult for hackers trying to exploit large systems at once. So choose the version according to your preferences.

Metasploit is pre-installed in the Kali Linux. If you want to install Metasploit in other Linux versions use the following command

wget Metasploit

Metasploit also gives web interface access which can help us to access all the hosts we are using. Create an account and manage everything about the exploits here.

Practical scenario: Use Metasploit to make an exploit app that consists of malicious code which can take control of an android phone and can read its files, contacts, and messages and send them to a web server of yours.

Don't forget to try it out by yourself before looking at the solution that we describe here.

The process to create an exploitable app

a) For this practical exercise, it's better to use a virtual machine to leave no traces and an android emulator to check the app before sending it into the victim via a mail or by a person.

Why only android app can be made?

Usually, android is an open-source Linux system and is easy to exploit using tools like Metasploit. Other famous mobile operating systems like iOS use package managers with extensions IPA unlike app of android. Although there is a module splitter that can split the IPA files as of now there are no remote execution tools for Apple operating systems. So if you are trying to trick a user with Mac OS you need to find other ways.

However, if your victim is an android user then you can follow the below instructions to get the exploit into his device:

b) First of all, start the Metasploit in the device using the Metasploit command msfconsole

c) when the Linux shell shows the Metasploit interface select the payloads options. As we discussed earlier payloads are already proven bugs or vulnerabilities that can be achieved on a target running on a particular version of the software.

d) Here our target machine is android that is a Linux kernel machine. From the payloads shell search for msfvenom payload using the following command:

root @ hostname : msfvenom payload selec t

e) It gives five arguments that need to be filled out with information. Here we will discuss those five parameters in detail with commands.

a) -p

This needs to be used whenever you are trying to create an exploit using Metasploit. Here our payload is msfvenom

b) LHOST

This is the argument that describes our input network address. We have already learned about finding the IP address of our system using the ifconfig tool. The IP address is essential because we need to make a regular connection between the host and the victim so that the data can be transferred.

c) LPORT

Just like the previous one, this describes the port that we are willing to offer to this Metasploit program so that the victim app can send us data and other sensitive information.

d) R

This is where the apk format should be selected using the options. We are dealing with raw format information so this should be mentioned. If you are dealing with system software's execution files should be selected.

e) Location

This argument helps us to select the apk that we are referring to. You can simply give the location so that the Metasploit can start making it as an exploit. But before this process, we need to make some certificate installations so that everything runs in perfect. We will learn about this process in detail in this section.

Why certificates should be signed for android?

Installing signed applications is a mandatory thing in iOS however android doesn't have that restriction. But from the latest versions of android google made things difficult for hackers.

In the past with the help of remote execution hackers used to install trojans and worms with malicious content with a click. However, nowadays it has become a lot difficult because the user needs to grant the permissions manually. For this reason, you need to even polish your social engineering skills before trying to send the exploit to the victim. Make them believe with your words that this is a necessary application that needs to be

installed. That's what all hackers do to manipulates things for their exploitation.

We have a wide variety of signing tools like jar signer in the Metasploit interface. Select any one of them and use the following command

root @ examplelinux : msfvenom signer -ssh 23.2.2.1

The signer tool has the following attributes which will be explained in the next section in detail:

a) Type of certificate

There are a lot of certificate signs that need to be reviewed. You can use either an SHA way or by RSA way.

b) -verify

This command will make the tool to verify the app with the certification that the user has selected.

Now after verification, you need to use aligning tools that can mix up the exploit that you have developed with the apk tool.

c) align

Aligning is a process of inserting the exploit into the application. There are a lot of exploits that are available in the Metasploit database. Or you can even create an own shell script that can send back information to our server. For the visual demonstration of this topic we will explain about a shell script that performs the following functions:

a) The shell script should collect all the user contacts that are present on the phone.

b) The shell script should identify any new messages that are received and should send them to the Metasploit server.

Few might have got a doubt about how the Metasploit server works. It consists of a URL that is inserted into the exploit and we need to enter the

same URL into the browser in that network. In this way, we can access all the information that is being sent to the server.

d) Packing

After aligning the exploit into an app, you need to repackage it so that the apk looks perfectly normal. There is maybe a small marginal change in the size of the apk. To get undetected by the antiviruses you may use additional security options that can spoof the phone security.

You can use the following command to package the apk and exploit:

root @ example : pack location seems.ap k

After packaging the app, you should find a smart social engineering technique to send it to the victim's system. You can use an email with a rar file to send it to the phone. And when the victim successfully installs the application on his phone the app starts running in the background and will send all of the required data to the Metasploit server.

That's it about the exploit and its implementation using the Metasploit interface console. In the next chapter, we will give a brief introduction about the network sniffing tool known as wire shark and end this chapter.

Wireless Network sniffing tools

In a whole lot of kali Linux tools that are available, this stands among the most popular tools because of the huge expansion of wireless networks in the everyday world. From Bluetooth devices to wireless echo devices there are a lot of wireless devices that can be easily sniffed if they are connected to an unsecured wireless network.

What is sniffing?

Sniffing, in general, is a term that is defined for peeking into other stuff. In technical terms, sniffing means to look at network packets that are coming from other devices using tools like wire shark and air crack-ng. We will discuss both of these two tools in detail in this section

a) Wireshark

Wireshark is a network sniffing tool that tracks every packet that the network is dealing with. After starting the sniffer tool you need to wait at least 24 hours so that the sniffer catches a sufficient number of packets for analysis. During analysis, one can easily find all the sensitive information like passwords and credit card numbers. There is also a chance of using sniffing tools to hack unencrypted files and emails that are being sent. This is the reason why hackers should use encrypted mail services and virtual private networks for better security.

b) aircrack ng

Aircrack ng is another network sniffer tool that is extensively used to crack WiFi passwords. Usually, wpa2 WiFi routers are considered as the most secure network routers, unlike WPS routers which leave a bug to easily hack their passwords. This aircrack ng uses this vulnerability to boot scan the dictionary attacks and connect to the wireless network. Aircrack is one of the most popular wireless network programs that are available in the market right now.

By this, we have completed a detailed tour of some of the most important kali Linux tools with an overview of practical applications that can be done using these tools. We hope that with this chapter you have gained a lot of meaningful information.

This is the end of the book and I end this book with a clear explanation of what a hacker should try to be. A hacker learns skills and applies them with utmost hard work to crack into systems. Cracking doesn't mean to be exploiting and using them for their mischief purposes. Cracking means checking whether the walls are built strong or not. That's it that is what a hacker needs to remember all the time. All the best to our adventures of hacking!

Conclusion

Thank you for making it through to the end of Hacking with Linux, let’s hope it was informative and able to provide you with all of the tools you need to achieve your goals whatever they may be.

The next step is to use these concepts in real-world and exploit environments or protect them with goodwill. All the best!

Finally, if you found this book useful in any way, a review on Amazon is always appreciated!