Work Integrated Learning Programmes

M.Tech Software Engineering

Open Source Software Engineering – SEZG587

Assignment 1

Report On Linux

Submitted By,

Pavithra.S

2022MT93172

2022mt93172@wilp.bits-pilani.ac.in

INDEX

1. Abstract 4
2. Introduction 5
3. History of Movement 6
3.1 Origins and Genesis 6
3.2 The Birth of the Kernel 6
3.3 Explosive Growth and the GNU Connection 6
3.4 The Rise of Distributions 7
3.5 Corporate Involvement and Mainstream Adoption 7
3.6 Global Impact and Ongoing Development 7
4. Linux Shell and File Structures 8
4.1 The Linux Shell 8
Command Line Interface (CLI) 8
Shell Types 8
Commands and Syntax 8
File System Navigation 9
Process Management 9
Redirection and Pipelines 9
Scripting 9
Customization 10
Remote Access 10
Security 10
4.2 Linux File Structure and Directory Hierarchy 10
key directories in the Linux file structure 10
5. Security: Encryption, Integrity checks and signatures 12
5.1 Encryption 12

Data Encryption 12
Communication Encryption 12
GPG and PGP 13
5.2 Integrity Checks 13
5.2.1 File Integrity Monitoring (FIM) 13
5.2.2 Package Verification 14
5.3 Digital Signatures 15
Package and Software Signatures 15
Secure Email 15
Software Code Signing 15
5.4 Other Security Methodologies 16
Secure Boot 16
Access Controls 16
Firewalls and Network Security 16
SELinux and AppArmor 16
6. Linux Desktop: GNOME, KDE 17
6.1 Common Linux desktop environments 17
6.2 GNOME 17
Key Aspects of GNOME 18
6.3 KDE 20
Key aspects of KDE 20
7. Conclusion 22

1. Abstract

Linux is a very good example of the power of open-source software engineering. This report explores the interesting world of Linux, offering details on its history, structure, security features, and desktop environments. Beginning with an exploration of the origins of the Linux movement, this report identifies significant turning points and figures that have influenced its evolution. It then delves into the fundamental components of Linux, focusing on its shell and file structures, which form the foundation of this adaptable operating system.

An essential component of Linux is security, and this report covers the encryption, integrity checks, and digital signatures that protect Linux systems and information. Additionally, it explores the user interface with a comparison of popular desktop environments, GNOME and KDE, shedding light on their features and usability.

In closing, the report highlights the importance of Linux as a front-runner in the open-source operating system space, impacting the open-source software engineering community and acting as an example of the collaborative strength of the open-source community. Through this comprehensive overview, readers will gain a deeper understanding of the evolution, structure, and security measures within Linux, as well as the user experience offered by its desktop environments. Linux is a prime example of open-source success that has completely changed the field of software engineering.

2. Introduction

In the world of operating systems, Linux—commonly referred to as the "open-source revolution"—has become a potent and a significant platform that completely changed how we use computers and technology. Linux is more than simply an operating system; it's a community-driven software development methodology and a philosophy.

At the core, Linux is an operating system like Unix. It was created by Linus Torvalds in the year 1991. Unlike other proprietary operating systems such as the Windows or macOS, it is open-source. By open-source, it means that anybody can use, edit, and distribute it without restriction. Because Linux is open-source, a variety of multiple versions have emerged, each catering to unique requirements and tastes from different people. This versatility makes Linux a highly adaptable and suitable option for a wide range of computing jobs, ranging from desktop computing to server management.

This report provides an outline and overview of Linux's history, the structure of its shell and file system, its security features, and a comparison of its popular desktop environments, GNOME and KDE. The development of Linux from a small open-source project to a major force in the world of computing is characterized by cooperative innovation. The report highlights Linux's secure architecture and the program's essential parts. Additionally, it displays the variety of desktop environments that are easy to use. This makes Linux a powerful and adaptable open-source operating system.

3. History of Movement

The history of Linux is an intriguing narrative. The power of open-source collaboration is clearly showcased in the case of Linux. It also shows the impact of a single person, Linus Torvalds on the world of computing.

This section covers the overview of the key milestones and events that shaped the Linux movement and its development.

3.1 Origins and Genesis

In the year 1991, Linus Torvalds, who is a computer science student at the University of Helsinki wanted to create operating system like Unix, but open-source. That was the starting point of the Linux story. He wanted the operating system to be available for free for anyone to use. So, Torvalds first developed an initial version of the Linux kernel. This was released as an open-source project. This was done to encourage software developers around the world to contribute and form a global Linux community.

3.2 The Birth of the Kernel

The Linux movement was born with the release of the Linux Kernel. The traditional software's at that time were mostly closed-source. But Linux was developed as an open-source project and allowed anyone to view, edit and contribute to the source code. This attracted large number of enthusiasts who believed in the power of collaborative software development and wanted to contribute for its success.

3.3 Explosive Growth and the GNU Connection

The Linux kernel was compatible with the GNU Project's userland utilities. This led to the formation of a collaboration which is commonly known as GNU/Linux. The combination of the Linux kernel and GNU utilities created a complete and free Unix-like operating system. This partnership had a significantly high contribution to Linux's rapid growth and acceptance within the open-source community.

3.4 The Rise of Distributions

Linux's journey was further accelerated by the emergence of Linux distributions. The various distributions such as Debian, Red Hat, and Slackware packaged the Linux kernel along with a selection of software and tools. This made the deployment of the Linux systems easier for the users on their systems. These distributions were responsible for covering various use cases, ranging from server environments to desktop computing.

3.5 Corporate Involvement and Mainstream Adoption

The late 1990s and early 2000s there was increased corporate involvement within the Linux development. Companies such as IBM, Red Hat, and SUSE understood the potential of Linux for enterprise solutions and started to invest in its development. This support pushed Linux into mainstream adoption, making it a viable and potential alternative to proprietary operating systems.

3.6 Global Impact and Ongoing Development

Now, Linux has a wide and diverse user base. From individual hobbyists to major corporations and government entities, there were a variety of users for Linux. It powers helps to server, mobile devices, embedded systems, and much more. The open-source model of Linux also influenced innovation across various other fields. Cloud computing, cybersecurity, and the Internet of Things (IoT) are few among them.

The Linux movement is a proof to the power of open-source collaboration. The individuals and organizations from around the world are working to contribute and help in the development of a free and open operating system that has revolutionized the world of computing.

4. Linux Shell and File Structures

The Linux operating system includes major fundamental components such as the Linux shell and file structures. This section provides the information about the Linux shell, the importance of it, and the typical file structure and directory hierarchy in Linux.

4.1 The Linux Shell

The Linux shell is a command-line interface (CLI) that allows users to interact with the operating system through commands. It serves as a bridge or interface between the user and the Linux kernel, thereby enabling the users to issue instructions, manage files and directories, and perform system operations.

Here are some key points to understand the Linux shell:

Command Line Interface (CLI)

The Linux shell includes a text-based command-line interface. In this interface, the users can enter commands through the keyboard. It is well known for its speed and flexibility. It is highly efficient for system administrators who mainly work with the commands rather than a UI Interface.

Shell Types

There are multiple types of shell options available for the users to choose from such as,

- Bash (Bourne-Again Shell),
- Zsh,
- Fish,
- Etc

The user can choose the shell that suits their needs and preferences.

The most commonly used shell is Bash.

Commands and Syntax

The users can interact with the shell by entering commands and arguments.

The commands will be followed by options and arguments that will specify what the command should do. The syntax for each command can be different and will each have a different purpose. The users often refer to the manual pages (using the man command) to learn more about specific commands and their options. With this info, they will be able to execute the commands which they wish to execute.

File System Navigation

The users navigate the Linux file system through the shell.

Common commands for file system navigation includes,

- cd to change directories,
- ls to list files,
- pwd to display the current working directory,
- mkdir to create directories,
- rmdir to remove directories,
- etc,

Process Management

The various processes running on the system can be managed by the users easily in Linux. Commands such ps (process status) and kill allows the users to view information about running processes and terminate them, if necessary.

Redirection and Pipelines

The redirection of input and output streams is possible in Linux. It enables the users to redirect the output of one command as the input to another. The pipe symbol (|) is often used to create pipelines, which are sequences of commands that are connected and need to run one after the another.

Scripting

The Linux shells are very powerful scripting environments. It allows the users to execute sequences of commands that are saved in a file. Shell scripts are nothing but sequences of commands that are saved in a single file to execute one after the other. This is useful to automate tasks, perform system maintenance, or create custom applications. Some of the common scripting languages for Linux are

- Bash,
- Perl, and
- Python.

Customization

The shell environment can be customized as per the needs of the user by

- Defining aliases,
- Setting environment variables, and
- Configuring prompt appearances to suit their preferences.

Configuration files like ~/.bashrc and ~/.bash_profile (for Bash) are often used for this purpose.

Remote Access

The secure shell (SSH) is used for accessing remote access to the Linux servers. It allows the user to login into a remote system's shell. After login, they can execute the commands in the remote system's shell as if they were physically present near the system.

Security

In order to maintain proper system security, the user permissions can be configured properly and appropriately for the shell environment. The user management and access control play a very important role in the system security of Linux. So, it's important to set up proper permissions and manage the shell environment.

4.2 Linux File Structure and Directory Hierarchy

In Linux, a hierarchical file structure is followed. The root directory is at the top in the path "/". The structure of the file system is organized like a tree format. This makes it easier to navigate and manage the directories and the files within it.

key directories in the Linux file structure

/ (Root Directory): The top-level directory from which the entire file system hierarchy begins.

/bin: Contains essential binary files and system command binaries required for system recovery.

/etc: Configuration files and system-wide configuration scripts are stored here.

/home: Home directories for individual users are located in this directory. Each user has a subdirectory within /home.

/var: Variable data files, such as logs, spool files, and temporary data, are stored here.

/usr: Contains user data, system libraries, and additional software packages.

/lib: Library files used by system utilities and applications.

/dev: Special device files representing hardware devices are stored in this directory.

/tmp: A directory for temporary files that are typically deleted on system reboot.

/opt: Optional software packages and applications are often installed in this directory.

/proc: A virtual file system that provides information about running processes and system hardware.

/mnt and /media: These directories are used for mounting external storage devices such as USB drives and network shares.

5. Security: Encryption, Integrity checks and signatures

Linux considers security as a very important part and has various features that ensures the security of the data such as,

- Confidentiality.
- Integrity.
- Authenticity of data.

Here are the key features related to the security in Linux:

5.1 Encryption

Encryption is the process of transforming data into unreadable format so that it can be protected from unauthorized access and accidental leak of data.

The primary purposes of encryption are as follows:

- Confidentiality
- Data Security
- Data Integrity
- Authentication

In Linux, different kinds of encryption techniques are used such as,

Data Encryption

In order to protect the data at rest and in transit, Linuz uses a variety of encryption mechanisms. At the file system-level, encryption methods such as eCryptfs or dm-crypt is used which enables users to encrypt their data on the disk. So, even if an unauthorized user gains access to the disk, they won't be able to read the data without the decryption key.

Communication Encryption

Linux uses secure communication protocols such as SSL/TLS (Secure Sockets Layer/Transport Layer Security) to encrypt the data that is transferred over the network. This is very important for securing the traffic from the web, email, and other networked applications.

GPG and PGP

Tools such as GnuPG (GPG is also used in Linux for strong encryption and digital signatures. These tools are commonly used for,

- Encrypting emails,
- Verifying software integrity, and
- Protecting sensitive files.

5.2 Integrity Checks

In Linux, integrity checks are the mechanisms or processes that are used for verifying the integrity of,

- System files,
- Configuration files and
- Other critical components of the OS.

The aim of performing these checks is to detect any unauthorized modification, corruption or tampering of these files. If the files are compromised, it would indicate a security breach or system instability.

The common ways of performing integrity checks in Linux are,

5.2.1 File Integrity Monitoring (FIM)

In Linux, to monitor and detect the changes to the files, the File Integrity Monitoring (FIM) is used. The process involves comparing the current existing state of the files with a known baseline or reference set of hashes or checksums. If the checksum of the file doesn’t match the value that is expected, it indicates an issue in the file integrity.

Some of the common tools that are used for FIM in Linux are,

1. **Tripwire**

   Tripwire is a famous FIM tool that maintains a database of file attributes and cryptographic hashes. On a regular basis, the file system is checked against this database and reports are generated when any changes are detected.

2. **AIDE (Advanced Intrusion Detection Environment)**

   AIDE is another FIM tool that is similar to that of Tripwire that allows us to create a database of file attributes and then verify files against this database to identify any modifications or discrepancies.

3. **Osquery**

   Osquery is an open-source, cross-platform tool. It enables us to query system information, including file integrity data. It is highly extensible and can be used for monitoring and detecting file changes.

5.2.2 Package Verification

Package verification is a way to ensure and confirm the integrity of any installed software.

In Linux distributions that use package managers such as Debian's APT, Red Hat's Yum or DNF, or others, package verification is a good way to ensure integrity. This method involves checking package files and their metadata to ensure they have not been altered.

Common package verification commands are,

1. **Debian/Ubuntu (APT)**

   To verify the integrity of the installed packages, the dpkg command can be used with the –verify option. Also, the apt package manager can check the integrity of package downloads before they are installed in the system.

2. **Red Hat/CentOS (Yum/DNF)**

   The integrity of the package can be verified by using the rpm package manager in combination with the rpm -V command. Yum and DNF also include options for checking package integrity.

To maintain system security and stability, integrity tests are crucial and important. They will help to find out any unauthorized changes which can be caused due to security breaches, errors in the software or any system

misconfigurations. By performing the integrity tests on a regular and periodic basis, the system admins can detect any issues at an early stage and address them promptly. This will ensure reliability and security of the Linux systems.

5.3 Digital Signatures

Digital signatures are the cryptographic techniques that are used for verifying the integrity and authenticity of the data. The data can be documents, files or any messages. They serve as a means to ensure that the digital content in hand is not tampered and belongs to the sender who claims it is theirs.

In Linux, digital signatures are commonly used for,

- Package verification,
- Secure email communication,
- Code signing, and more.

Package and Software Signatures

The identify and prevent the installation of tampered or any kind of malicious software, Linux expects the software packages to be signed digitally by the maintainers of the package. This digital signature is to ensure that the software package has not been tampered or altered since it was signed This provides a level of trust for the package.

Secure Email

Secure email protocols such as S/MIME and PGP/GPG allows the users to digitally sign and encrypt their email messages. Here, the digital signatures in the email are used to verify the sender's identity and the integrity of the message content.

Software Code Signing

In order to prove that the software has not been altered or changed, the software developers sign their code with the help of digital signatures. This will prove that it was not changed after signing and prove it’s authenticity. This plays an important role while ensuring the trustworthiness of software packages.

By using digital signatures, security and trustworthiness of digital communication and transactions in Linux systems is enhanced. The users and the system administrators can rely on digital signatures to verify the authenticity and integrity of software, documents, and communication, helping to prevent tampering, impersonation, and other security risks.

5.4 Other Security Methodologies

Secure Boot

Many Linux distributions and hardware platforms support Secure Boot which requires digitally signed bootloaders and kernel images. This feature is very useful for preventing the execution of unauthorized or malicious boot components. This thereby enhances the security of the system from the very start.

Access Controls

The use of proper access controls will ensure that only the users with the required permission can access sensitive data and system resources. Linux implements strong access controls, such as file permissions, user and group management, and other capabilities.

Firewalls and Network Security

Linux offers powerful firewall solutions like iptables and nftables for controlling network traffic. With the help of these tools, the administrators can define rules to permit or deny specific network connections based on source, destination, and other parameters.

SELinux and AppArmor

Linux distributions often include mandatory access control systems like SELinux and AppArmor. These systems confine the actions of processes and enforce security policies to reduce the attack surface.

6. Linux Desktop: GNOME, KDE

Linux desktop environments are user interfaces (GUI's) that gives the users an easy and user-friendly way to interact and use the Linux OS. It will be useful for both beginners and experienced users by improving the accessibility.

These environments provide a wide range of features and tools. It allows the users to customize the look and feel of their desktop, manage files, launch applications, and perform various tasks at ease. Desktop environments play a crucial role in making Linux more accessible and user-friendly.

6.1 Common Linux desktop environments

Some of the common Linux desktop environment includes,

- **GNOME:** It is known for its user-friendliness and clean design. GNOME is the default desktop environment for many Linux distributions, such as Ubuntu.
- **KDE Plasma:** KDE Plasma offers a feature-rich and highly customizable environment. It is known for its aesthetics and advanced configuration options.
- **Xfce:** Xfce is a lightweight and resource-efficient desktop environment. It provides a balance between functionality and system performance.
- **Cinnamon:** Cinnamon is a modern and straightforward environment. It is often associated with Linux Mint.
- **LXQt:** LXQt is another lightweight desktop environment. It is designed to be fast and resource-efficient.
- **MATE:** MATE is a continuation of the GNOME 2 desktop environment. It provides a traditional and familiar interface.

Two of the most popular and feature-rich desktop environments for Linux are GNOME and KDE. Each of these environments offers a distinct user experience, design philosophy, and set of applications.

6.2 GNOME

GNOME stands for the GNU Network Object Model Environment. It is one of the most popular desktop environments for Linux. It is known for its,

- User-friendly design,
- Focus on simplicity, and
- Modern aesthetics.

GNOME is the default desktop environment in several Linux distributions, such as Ubuntu.

Key Aspects of GNOME

User Interface (UI)

GNOME has a very clean and visually appealing user interface. It has top bar that shows the system indicators and a dash (application launcher) located on the left side. It has a Super key which is often the Windows key to launch the Activities overview. This allows the users to access applications, search for files, and view virtual desktops.

Application Menu

There is a straightforward application menu, which is accessible from the Activities overview. This will allow the users to easily find and launch applications. The menu also supports a quick application search.

Window Management

Window management features such as window tiling and virtual workspaces are also available in GNOME. The users can organize and switch between multiple open windows and applications with ease.

Extensions

GNOME is highly customizable through the use of extensions. These extensions can be used to add extra features and functionalities to the desktop. Users can install extensions from the GNOME Extensions website, making it easy for them to update the environment to their needs.

GNOME Software

GNOME Software is a user-friendly software center for installing and managing applications. It includes user reviews, software updates, and an easy way to discover and install new applications.

Files (Nautilus)

GNOME includes a file manager called Nautilus. It is mainly is designed for ease of use. It supports features like file previews, tabs, and integrated cloud storage services.

System Settings

GNOME's system settings are a central location for configuring various settings and aspects of the desktop and overall system. It covers,

- Network settings,
- Display options,
- Privacy settings, and more.

Accessibility

GNOME has a strong emphasis on accessibility. It provides features like screen readers and magnifiers to ensure that users with disabilities can effectively and easily interact with the desktop.

Integrated Online Accounts

GNOME allows the users to integrate their online accounts such as Google, Microsoft, and social media accounts. This integration enables and provides the user access to online services and synchronization of data like contacts and calendars.

GNOME Shell

The GNOME Shell is the core component of the desktop environment. It is responsible for managing the interface, taskbars, and user interactions. It can be extended through themes and extensions.

Application Ecosystem

GNOME has a suite of native applications such as,

- A web browser (GNOME Web, also known as Epiphany),
- Email client (Evolution), and
- Text editor (GNOME Text Editor).

However, users can install and use other applications of their choice

6.3 KDE

KDE stands for the "K Desktop Environment". It is another popular and highly customizable desktop environment for Linux and other Unix-like operating systems. KDE is mainly known for its rich feature set, flexibility, and aesthetic appeal

Key aspects of KDE

Plasma Desktop

KDE's desktop environment is called "Plasma." It features a modern and visually appealing interface. It has a panel at the bottom of the screen (by default) that contains an application menu, system tray, and taskbar. The desktop can be customized further to suit users' preferences.

KDE Applications

KDE offers a diverse collection of applications seamlessly integrated into its desktop environment. These applications cater to various needs and include:

1. **Dolphin:** This versatile file manager simplifies file organization and navigation.
2. **Konsole:** It's a user-friendly terminal emulator that allows text-based interactions.
3. **Okular:** A document viewer that supports a wide range of file formats.
4. **Kate:** A text editor known for its simplicity and powerful features.

KDE applications are not only functional but also known for their consistent and attractive design.

KDE Plasma Widgets

A standout feature of KDE is the inclusion of desktop widgets, affectionately called "Plasmoids." These widgets can be placed on the desktop or panel, providing quick access to valuable information and tools. Examples include widgets for clocks, weather updates, and system performance monitoring.

Window Management

KDE excels in offering advanced window management capabilities. Users have the flexibility to organize windows through features like tiling, which maximizes screen space utilization. Virtual desktops, referred to as "activities," allow for efficient workspace organization. With KDE, the user had full control over how their windows behave and appear.

KDE System Settings

System configuration in KDE is made easy through the "System Settings" control panel. This interface enables the user to fine-tune various aspects of the desktop environment, including display settings, input devices, and power management.

Discover Software Center

KDE boasts "Discover," a user-friendly software center designed to simplify the installation, management, and updates of applications. Discover provides access to a diverse array of software and packages from repositories, making software management straightforward.

Customization

KDE is known for its high level of customization. Users can easily change the desktop theme, icons, widgets, and other visual elements. Additionally, Plasma desktop layouts can be adjusted to provide traditional, modern, or innovative desktop experiences.

KDE Connect

KDE Connect is a feature that allows seamless integration between a Linux desktop running KDE and an Android smartphone. It enables functions like file sharing, notification synchronization, and remote control of media playback.

Community and Ecosystem

The KDE project has a dedicated and active community that continually develops and improves the desktop environment. It also includes various applications and tools beyond the desktop environment itself.

7. Conclusion

Linux, the heart of open-source ingenuity, has taken us on an extraordinary journey, exploring its rich history, core structure, security fortifications, and diverse desktop landscapes. It's not just an operating system; it's a testament to what can happen when people come together to create something extraordinary.

Think of Linux as the cornerstone of your digital life, quietly and efficiently managing everything behind the scenes. Its core components, the shell, and the organized file structure, make it a dependable workhorse. Whether you're a tech enthusiast or a newcomer, Linux offers you the tools you need to explore, create, and customize your digital world.

When it comes to security, Linux doesn't compromise. It's like a vigilant guardian, encrypting your sensitive data, ensuring the integrity of your files, and confirming the authenticity of software you rely on. Trust is at the core of the Linux experience.

And let's not forget about the choices Linux gives you. GNOME and KDE, like two well-designed rooms in the same house, offer distinct styles and customizable options. It's like having the freedom to arrange your digital space just the way you like it.