Task 2 – Basic Firewall Configuration Using UFW

1. Introduction

Firewalls play a critical role in protecting systems from unauthorized access and network-based attacks. On Linux systems, UFW (Uncomplicated Firewall) provides a simple and effective interface for managing firewall rules without requiring deep knowledge of underlying iptables configurations.

This report documents Task 2 – Basic Firewall Configuration Using UFW, where a basic firewall policy was implemented to allow essential services while blocking unnecessary ones in a controlled virtual lab environment.

---

2. Objective

The main objectives of this task are to:

- Configure a basic firewall on a Linux system using UFW
- Allow secure remote access through SSH
- Block unnecessary services such as HTTP
- Enforce firewall rules by enabling UFW
- Analyze the security significance of the applied rules

---

3. Environment

The firewall configuration was performed in a virtualized lab environment to ensure safe testing.

- Operating System: Kali Linux
- Platform: Oracle VirtualBox
- Firewall Tool: UFW (Uncomplicated Firewall)

This environment ensures isolation from production or external networks.

---

4. Tool Used

UFW (Uncomplicated Firewall)

UFW is a front-end for managing firewall rules on Linux systems. It simplifies rule creation by using easy-to-understand commands while still providing robust protection.

Key features of UFW include:

- Simple rule syntax
- Support for IPv4 and IPv6
- Default deny policies
- Integration with system startup

---

5. Firewall Configuration Performed

The following firewall rules were configured during this task:

- Allowed SSH (Port 22): To enable secure remote administrative access
- Denied HTTP (Port 80): To block unnecessary web-based services
- Enabled Firewall: To enforce all configured rules

These rules establish a minimal yet secure firewall policy.

---

6. Commands Executed and Explanation

6.1 Checking Firewall Status

The sudo ufw status command was used to check whether the firewall was active and to view existing rules.

6.2 Allowing SSH Traffic

The command sudo ufw allow ssh allows incoming connections on port 22, ensuring secure remote access using the SSH protocol.

6.3 Denying HTTP Traffic

The command sudo ufw deny http blocks incoming traffic on port 80, preventing access to web services that are not required.

6.4 Enabling the Firewall

The sudo ufw enable command activates the firewall and enforces all configured rules.

6.5 Verifying Firewall Rules

The command sudo ufw status verbose displays detailed firewall status, default policies, and applied rules for both IPv4 and IPv6.

---

7. Firewall Status and Results

After applying the rules, the firewall status showed:

- Firewall Status: Active

- Default Policy: Deny incoming, Allow outgoing
- Rules Applied: IPv4 and IPv6
- Allowed Service: SSH (Port 22)
- Blocked Service: HTTP (Port 80)

This confirms that the firewall rules were successfully enforced.

---

8. Security Significance

The configured firewall rules improve system security in the following ways:

- Secure Access: Allowing SSH ensures secure and encrypted administrative access
- Reduced Attack Surface: Blocking HTTP prevents exposure of unnecessary web services
- Best Practices: Default deny policy for incoming traffic aligns with security best practices
- Threat Mitigation: Helps protect against unauthorized access, scanning, and exploitation attempts

---

9. Conclusion

This task successfully demonstrated how to configure a basic firewall on a Linux system using UFW. By allowing essential services and blocking unnecessary ones, the system's exposure to network-based threats was significantly reduced.

The exercise highlights the importance of firewall management as a fundamental component of Linux system security and provides a strong foundation for implementing more advanced firewall policies in real-world environments.