Cyber Security Internship Report
Task-1

Abhijit Sudheer

August 12, 2025

1 Introduction

This report outlines the completion of Task 1: Introduction to Network Security Basics as part of my cybersecurity internship. The task involved researching fundamental network security concepts, implementing basic security measures on my local network using NixOS, and analyzing network traffic with Wireshark to observe these concepts in practice.

2 Network Security Concepts

This section provides a summary of the foundational network threats and security measures as researched for this task.

2.1 Network Threats

- **Viruses:** Malicious code that attaches itself to a legitimate program or file and requires a user to execute it to spread.
- **Worms:** Self-replicating malware that spreads autonomously across a network, exploiting vulnerabilities without requiring user interaction.
- **Trojans:** Malware disguised as legitimate software. Trojans trick users into installing them, after which they can perform a variety of malicious actions.
- **Phishing:** A social engineering attack where malicious actors impersonate a trusted entity to trick users into revealing sensitive information, such as passwords or credit card numbers.

2.2 Security Measures

- **Firewalls:** A security system that monitors and controls incoming and outgoing network traffic based on predefined security rules. It acts as a

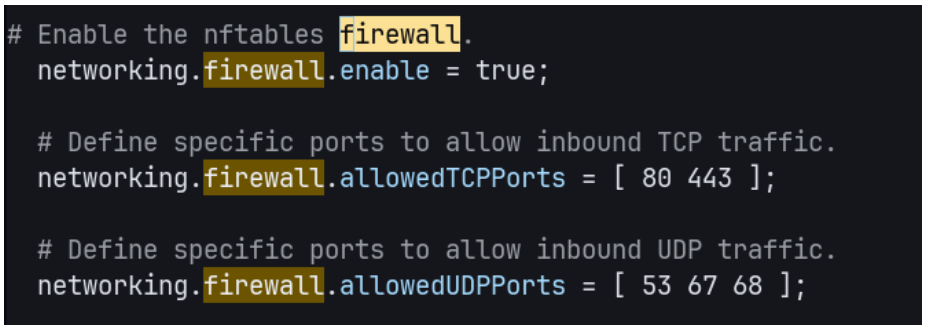

Figure 1: The configuration.nix file showing the added firewall rules. These lines define the specific ports that are allowed through the firewall.

barrier between a trusted internal network and untrusted external networks.

- **Encryption:** The process of encoding information so that only authorized parties can access it. For Wi-Fi networks, protocols like **WPA2 and WPA3** use encryption to protect data transmitted wirelessly.
- **Secure Network Configurations:** Best practices for securing a network, including changing default passwords on routers and other devices, disabling unnecessary services, and using strong authentication methods.

3 Implementation and Analysis

This section details the practical security measures implemented and the subsequent analysis of network traffic to verify their effectiveness.

3.1 Firewall Configuration on NixOS

I configured a firewall on my NixOS system using `nftables`, which is the default firewall framework. The configuration was done through the declarative configuration.nix file. The primary goal was to enforce a "deny-by-default" security posture, allowing only essential network services to communicate with the system from the outside.

The following lines were added to my `/etc/nixos/configuration.nix` file to enable the firewall and specify the allowed ports:

```
networking.firewall.enable = true;
networking.firewall.allowedTCPPorts = [ 80 443 ];
networking.firewall.allowedUDPPorts = [ 53 67 68 ];
```

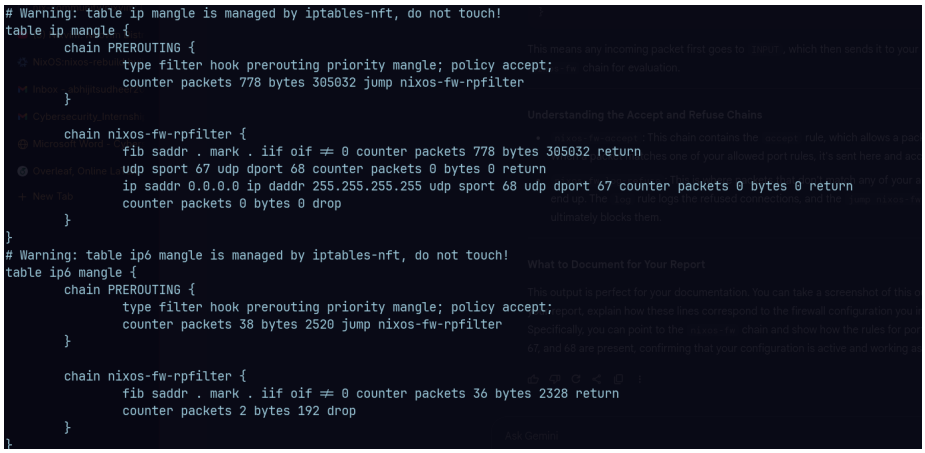

Figure 2: Image 2: The terminal output from sudo nft list ruleset, showing the active firewall rules for IPv4 and IPv6. This confirms that the rules from the configuration file were successfully applied and are actively filtering traffic.

After running `sudo nixos-rebuild switch`, I verified that the rules were active using the nft command. The terminal output confirmed that the specified ports were open for incoming traffic, while all other connections were logged and dropped. This provides evidence that the implemented security measures are functioning correctly.

3.1.1 The Importance of Port Filtering

The specific ports filtered in the configuration play a critical role in network security and functionality.

- **TCP Port 80 (HTTP) 443 (HTTPS):** These ports are essential for web browsing and web services. HTTP traffic (port 80) is unencrypted and vulnerable to snooping, which is why most modern websites use HTTPS (port 443), where data is encrypted, providing confidentiality and integrity.
- **UDP Port 53 (DNS):** This port is used for the Domain Name System, which translates human-readable domain names (e.g., google.com) into machine-readable IP addresses. Filtering this port would prevent a system from resolving domain names and accessing the internet.
- **UDP Ports 67 68 (DHCP):** These ports are used by the Dynamic Host Configuration Protocol, which automatically assigns IP addresses to devices on a network. They are crucial for a device to join and communicate on a local network.

Leaving unnecessary ports open can pose a significant security risk, as each open port is a potential entry point for attackers. By using a firewall to close unused

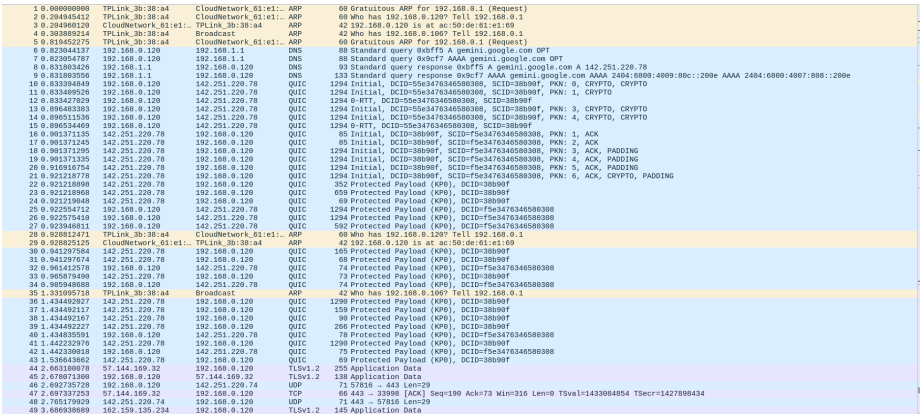

Figure 3: Image 3: A screenshot of the Wireshark capture window, with traffic filtered to highlight different protocols.

ports, the attack surface of the system is drastically reduced, mitigating risks such as unauthorized access, port scanning, and denial-of-service attacks.

3.2 Network Traffic Analysis with Wireshark

To understand how these security concepts apply in a real-world scenario, I used Wireshark to capture network traffic on my `wlan0` interface. I used filters to isolate and analyze specific types of traffic, which provided a clear view of the various protocols and data packets flowing through my network.

Key observations from the analysis included:

- **DNS Traffic (UDP/53):** By using the filter `dns`, I was able to see my computer sending requests to a DNS server to resolve domain names to IP addresses, which is a fundamental part of web browsing.
- **HTTP Traffic (TCP/80):** I used the filter `http` while visiting an unencrypted website. This allowed me to capture and inspect packets containing the plain-text HTTP protocol, highlighting the security risk of unencrypted communication.
- **HTTPS Traffic (TCP/443):** By contrast, using the filter `tls` on traffic to a secure website showed encrypted data, demonstrating how HTTPS protects the integrity and confidentiality of communication.

4 Conclusion

This task provided a comprehensive understanding of network ports, their functions, and the role of a firewall in network security. I learned that a firewall

serves as a critical barrier for a system, and that port filtering is a fundamental technique for reducing a system's attack surface by selectively allowing or denying network traffic. The Wireshark analysis was instrumental in visually demonstrating how network protocols operate and confirming that the implemented firewall rules were actively managing network traffic. This project also highlighted the value of a layered security approach, incorporating additional practices such as regular software updates and multi-factor authentication, to ensure a system remains robustly protected and safe from potential threats.