Project Report: Personal Firewall using Python

Intern Name: [KUNAL]

Project Title: Personal Firewall using Python

Tools Used: Python, Scapy, iptables (Linux), Tkinter (Optional GUI)--

1. Introduction

The growing dependency on internet-connected systems has led to an increased need for effective network security. Firewalls play a crucial role in

protecting systems from unauthorized access and malicious traffic. As part of my internship at **Elevate Labs**, I was tasked with developing a

Personal Firewall in Python to monitor, filter, and control network traffic based on custom rules defined by the user. The objective was to build a

lightweight and customizable firewall tool suitable for personal or small-scale use.--

2. Objective

The main goal of the project was to create a personal firewall that:

* **Monitors network traffic** in real-time.

* **Blocks or allows traffic** based on user-defined rules (IP, port, protocol).

* **Logs suspicious or blocked packets** for later review.

* Optionally **uses system-level enforcement** via iptables (Linux).

* Provides a **simple GUI (using Tkinter)** for live monitoring (optional feature).--

3. Tools and Technologies

* **Python:** Core programming language used for development.

* **Scapy:** A powerful Python-based packet manipulation tool used for sniffing and analyzing traffic.

* **iptables:** Linux-based firewall utility used optionally for enforcing filtering at the kernel level.

* **Tkinter:** Built-in Python GUI library used to provide a basic user interface for monitoring and control.--

4. Methodology

a. **Packet Sniffing with Scapy**

b. **Rule Definition and Filtering**

c. **Packet Logging**

d. **iptables Integration (Linux Only)**

e. **Graphical User Interface (Optional)**--

5. Key Features

* Real-time packet capture and analysis.

* Customizable filtering rules.

1

* Packet logging for audit and review.

* CLI and optional GUI support.

* Basic integration with Linux iptables.--

6. Results and Outcomes

* Successfully built a working **CLI-based firewall** that can filter and log network traffic.

* Implemented a **basic rule engine** for traffic control.

* Created **log files** that record all blocked packets with timestamps.

* (Optional) Added GUI using Tkinter for better usability.

* (Optional) Integrated iptables on Linux for stronger system-level control.--

7. Challenges Faced

* Handling high-volume traffic without lag.

* Parsing different protocol types (ICMP, UDP, TCP) dynamically.

* Managing GUI updates in real-time without freezing the application.

* Ensuring compatibility across different Linux distributions for iptables integration.--

8. Conclusion

This project provided hands-on experience in **network traffic analysis**, **packet-level programming**, and **security policy enforcement** using

Python. The use of Scapy allowed deep packet inspection, while iptables added robustness for Linux users. Through this project, I gained valuable

insights into the core principles of firewalls, network filtering, and system-level security mechanisms.--

9. Future Improvements

* Add support for exporting and importing rule sets.

* Add email/SMS alerts for suspicious traffic detection.

* Implement automatic threat intelligence lookup for known malicious IPs.

* Optimize performance for continuous background monitoring.

* Expand GUI to include rule editing and visualization.--

Submitted by:

[KUNAL]

Elevate Labs Cybersecurity Internship

[23., June 2025]