Project Report

Project Title: Personal Firewall using Python
Author: Akash H
Repository: https://github.com/Akash-H-119/Personal-Firewall-Python

1. Project Overview

The Personal Firewall using Python project is a lightweight network security tool designed to monitor, analyze, and control incoming and outgoing packets on a system. It enables users to define custom firewall rules to block or allow specific network traffic based on parameters such as IP addresses, ports, or protocols.

The project leverages Python's networking libraries to provide a simple yet effective firewall mechanism, capable of real-time packet sniffing, logging, and interactive control through both a command-line and optional graphical interface.

Objectives:
- Develop a personal firewall for monitoring network traffic.
- Allow users to create and manage custom rules for packet filtering.
- Log suspicious activities for further analysis.
- Provide an optional GUI interface for real-time control.

2. Technologies and Tools Used

Programming Language: Python 3

Libraries and Tools:
- Scapy: For packet sniffing and traffic manipulation.
- Tkinter: For graphical user interface (GUI).
- Logging: To maintain logs of network events.
- JSON: For storing and managing rules.
- Npcap (Windows) / iptables (Linux): For low-level packet handling.Supported Platforms: Windows and Linux.

3. Key Features

1. Real-time packet sniffing.
2. Custom rule-based filtering.
3. Suspicious packet logging.
4. GUI interface using Tkinter.
5. Cross-platform compatibility.
6. JSON-based configuration for rules.

4. Repository Structure

- firewall.py: Core packet capture and filtering logic.
- gui.py: GUI using Tkinter.
- iptables_helper.py: Interface to system firewall tools.
- logger.py: Logging configuration and event recording.
- rules.json: User-defined rules.

- requirements.txt: Dependencies list.
- Execution_Steps.txt: Setup and run instructions.
- README.md: Documentation and overview.

5. Working Principle

1. Packet Capture – Network packets are captured in real time.
2. Rule Evaluation – Captured packets are compared to user-defined rules.
3. Action Decision – Packets are either blocked or allowed based on the rule match.
4. Logging – Suspicious or blocked packets are recorded in a log file.
5. User Interface – Users can control and view traffic through CLI or GUI modes.

6. Installation and Setup

1. Clone the repository: git clone https://github.com/Akash-H-119/Personal-Firewall-Python.git cd Personal-Firewall-Python

2. Install dependencies: pip install -r

requirements.txt 3. Configure rules in rules.json.

4. Run:
python firewall.py (CLI mode)
python gui.py (GUI mode)

7. Output and Logs

Logs:
- Blocked and suspicious packets are recorded with timestamp, IP, and protocol details.

GUI Display:
- Shows live traffic stats, blocked/allowed packets, and real-time alerts.

8. Applications

- Personal system security.
- Educational use in cybersecurity learning.
- Research on packet filtering.
- Development of advanced security tools.

9. Conclusion

The Personal Firewall using Python project offers a customizable and effective approach to monitoring and controlling network traffic. It achieves real-time packet analysis, rule-based filtering, and logging, making it suitable for both security enthusiasts and learners in the networking field.