Personal Firewall with Python and GUI

Abstract

This project implements a lightweight personal firewall in Python with a graphical user interface. It provides packet sniffing, live monitoring, and interactive rule management. The firewall goes beyond basic packet filtering by supporting dynamic rule updates, profile-based configurations, and GUI-based controls, enabling both security and usability.

Introduction

Firewalls are an essential security layer that monitor and control network traffic. Traditional firewalls operate at the system or hardware level, but this project demonstrates how a personal firewall can be built entirely in Python using libraries like Scapy for packet sniffing and Tkinter for the GUI. The aim is to provide both educational value and a usable desktop firewall tool.

Tools Used

- Python 3.13: Programming language
- Scapy: Packet sniffing and dissection
- Tkinter: GUI framework for Python
- ReportLab: PDF generation for report
- Linux (Ubuntu VM): Test environment
- Virtualenv: Isolated Python environment

Steps

1. Environment Setup: Created a virtual environment and installed Scapy, Tkinter, and dependencies.
2. Packet Sniffer: Implemented with Scapy to capture live packets, extracting protocol, source, destination, and length.
3. Rules Engine: Designed rule functions (add_rule, reset_rules, check_packet) to allow or block packets dynamically.
4. GUI Monitor: Built with Tkinter's ScrolledText to display live logs from the sniffer.
5. Interactive Controls:
   - Block All ICMP button to block ping/ICMP packets.
   - Reset Rules button to clear rules.
   - Profiles dropdown (Home, Public, Office) with pre-defined rule sets.
6. Integration: Linked GUI to sniffer logs, rules engine, and tested with traffic (ping, curl).
7. Testing: Verified ICMP blocking, rule reset, and profile switching in live environments.

Conclusion

This project successfully demonstrates how to design and implement a personal firewall in Python. The GUI provides ease of use, while Scapy ensures accurate packet sniffing. Unique features such as profile-based rules and GUI-driven interaction make this project more than a simple firewall—it is an educational yet practical network security application. Future work can include per-process blocking, quarantine mode, and anomaly detection to further enhance its capabilities.