My Personal Firewall

For Linux

@SRCybersecurity

Introduction

My Personal Firewall is a moderately advanced project that involves packet sniffing, add rule by Blocking/ Unblocking IP, port, protocol, remove rules, Shows sniffed packets logging and firewall decisions.

Features:

- Add Rules – Allow/block IP, port, protocol.
- View Rules – Display currently added iptables rules.
- Remove Rules – Select and remove rules.
- Packet Logging – Show sniffed packets + firewall decisions.

Abstract

My Personal Firewall is a full GUI firewall application built using Python. By integrating Tkinter for the GUI, Scapy for real-time packet sniffing, and iptables for managing firewall rules on Linux systems, the application provides users with an accessible and interactive way to monitor and control network traffic.

Key features include the ability to add custom rules to allow or block specific IP addresses, ports, and protocols; view currently active iptables rules; and remove selected rules by line number. Additionally, the system performs live packet logging, displaying source and destination IP addresses along with firewall decisions (e.g., blocked or allowed) in real time.

The project emphasizes ease of use, offering a user-friendly yet powerful tool for learning, testing, or managing simple firewall configurations. It is designed for educational and experimental use on Linux systems with root access, making it a valuable project for those exploring system security, network monitoring, or Python-based system applications.

Tools Used:

- GUI Development – Using Tkinter.
- Packet Sniffing – Using Scapy.
- Firewall Rules Management – Interfacing with iptables.
- Logging – Capturing packet info and firewall decisions.

Steps Involved in Building the Project:

Dissected into five phases:

1. **Helper functions** are used to interact with the system's firewall using the Linux iptables command.
2. **Logging functions** are used to display firewall activity and rules in the GUI, and to log rule actions to a file**.**
3. **Sniffing function**s implement real-time network packet sniffing using Scapy, and display basic information about the captured IP packets in the GUI log.
4. **GUI functions** are part of the GUI logic, they handle user interactions for managing firewall rules using iptables.
5. **Main GUI Setup** section builds the graphical interface using Tkinter and initializes the firewall's runtime behaviors.

Conclusion

This project demonstrates how various technologies can be integrated to:

- Monitor network traffic in real-time using Scapy's packet sniffing capabilities.
- Dynamically manage firewall rules through iptables via a Python interface.
- Provide a graphical interface for users to easily add, view, and remove rules using Tkinter.
- Log all actions and packet events to both the GUI and a persistent log file for transparency and debugging.

This project is a great example of how Python can be used for practical cybersecurity applications, offering both technical depth and usability. It serves as a strong foundation for more advanced features like protocol-specific filters, rule persistence, and automatic threat detection in future iterations.

THANK YOU

END