🛡️ Network Vulnerability Scanner A GUI-based Python tool for scanning a local network (subnet) to discover live hosts, detect open ports, and flag known vulnerabilities. Built with Tkinter for a user-friendly interface.

📌 Features 🔍 Subnet scanning (e.g., 192.168.1.0/24)

🌐 Live host detection using ping

🧠 Port scanning on common ports (21, 22, 23, 80, 443)

⚠️ Vulnerability detection for insecure services

📤 Export results to CSV file

🖥️ Tkinter GUI with real-time logs

🛑 Stop scan functionality

🧵 Threaded scanning to keep GUI responsive

🧰 Technologies Used Python 3

Tkinter (built-in GUI library)

socket, os, csv, threading

🚀 How to Run Prerequisites: Python installed (version 3.x)

Steps: Save the script as scanner.py

Open terminal/cmd in the script's directory

Run the script:

python scanner.py GUI will open:

Enter a subnet (e.g., 192.168.1.)

Click Start Scan

View output logs live

Click Stop Scan if needed

Save CSV report after scan completes

📂 Output Scanning output is shown in a scrollable window.

Vulnerabilities flagged in-line.

Final report can be exported as .csv file (IP, Open Ports, Vulnerabilities).

📋 Example Vulnerabilities Flagged Port Service Vulnerability 21 FTP May allow anonymous login 23 Telnet Insecure protocol - data sent in plaintext 80 HTTP No encryption - vulnerable to sniffing 💬 Project Motivation Created to demonstrate basic network scanning and vulnerability detection in a local environment using Python. It simulates real-world pentesting tools in a lightweight, educational format.

📌 Disclaimer This tool is intended for educational and ethical testing only.

Do not scan networks without proper authorization.

👨‍💻 Author Created by Pratham Kaushik with guidance from ChatGPT.

Need help modifying or upgrading it? Just ask!