Python Port Scanner

Schifîrneț Petre-Iustin

Petroleum-Gas University of Ploiești
Faculty of Mechanical and Electrical Engineering – Applied Informatics and Automation

Introduction

The Python Port Scanner is a command-line utility developed to identify open TCP ports on a specified target system (local or remote). This tool provides an educational and practical demonstration of how port scanning works using low-level socket connections in Python. It is useful for basic network diagnostics, cybersecurity training, and understanding how services are exposed to a network.

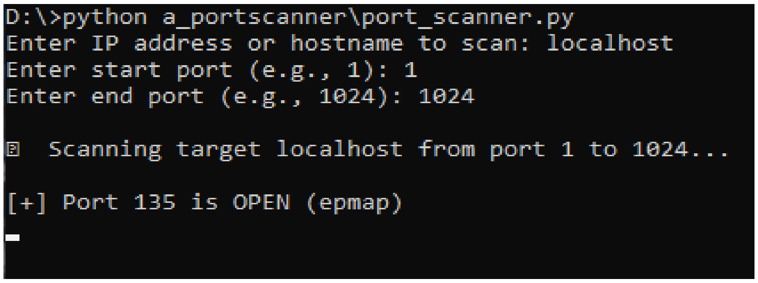
This scanner was developed using only Python’s built-in libraries, which ensures compatibility across operating systems and makes it suitable for lightweight deployments or instructional purposes.

Objective

The main goal of this project is to create a simple yet functional tool that helps users:

- Learn how network ports operate
- Understand how services are bound to specific ports
- Test local or remote systems for open ports

The tool is not intended for unauthorized or malicious use but rather for students, developers, and cybersecurity enthusiasts who want to better understand how devices communicate over TCP/IP networks.



```
D:\>python a_portscanner\port_scanner.py
Enter IP address or hostname to scan: localhost
Enter start port (e.g., 1): 1
Enter end port (e.g., 1024): 1024

⍰  Scanning target localhost from port 1 to 1024...

[+] Port 135 is OPEN (epmap)
```

Fig.1, Running the port scanner on a local target

How It Works

The script prompts the user to enter:

1. A target address (hostname or IP address)
2. A port range (e.g., from 1 to 1024)

For each port in the given range, the tool:

- Attempts to create a TCP connection using Python’s socket library
- If the connection is successful (port is open), it reports the port and its associated service (when known)
- If no connection is possible (port is closed or filtered), it skips silently

The program also measures and displays the total time taken for the scan.

Limitations & Ethical Use

This tool performs a basic TCP connect scan and does not include:

- Advanced techniques like SYN scans, UDP scans, or OS fingerprinting
- Detection of firewalls or intrusion prevention systems
- Parallel scanning (multithreading) for speed

Important: Port scanning should only be performed on systems you own or have explicit permission to test. Unauthorized scanning can be interpreted as malicious activity and is often against network policies or laws.

Conclusion

The Python Port Scanner is a compact and effective script for exploring basic network behavior and improving cybersecurity awareness. It provides hands-on experience with low-level networking, enhances understanding of service discovery, and can be extended into more advanced tools as the user progresses in their technical education.

Its simplicity and clarity make it an ideal starting point for anyone interested in cybersecurity, ethical hacking, or network engineering.