Int. J. Sci. R. Tech., 2026 3(4)
A Multidisciplinary peer-reviewed Journal
www.ijsrtjournal.com [ISSN: 2394-7063]

Original Article

Development And Implementation Of A Custom Port Scanner And Network Mapping Tool For Enhanced Network Security

Shinde Mohit Ramdas*, Gaikwad Shrikant Subhash, Talekar Tejas Uttam, Khatal K. B.

Department of Computer Engineering, Sahyadri Valley College of Engineering and Technology, Pune, MH

ABSTRACT

In the era of rapid digital transformation, network security has become a critical concern for organizations and individuals alike. Vulnerabilities arising from open ports and misconfigured network services are among the most common entry points for cyberattacks. This research focuses on the design, development, and implementation of a custom port scanner and network mapping tool aimed at identifying open ports, active hosts, and service versions within a target network. Unlike generic commercial scanners, this custom tool provides flexibility, transparency, and adaptability for specific network environments. The study addresses key challenges such as scanning efficiency, accuracy of service detection, and reduction of false positives. The tool employs TCP connect scanning, SYN scanning, and banner grabbing techniques to map network topologies and detect potential security loopholes. Experimental results demonstrate that the proposed tool effectively identifies active ports, running services, and operating system fingerprints with high accuracy and minimal network overhead. The findings emphasize the importance of custom security tools in educational and small enterprise settings where cost and customization are significant factors. This research contributes to the field of network security by providing a lightweight, open-source alternative for port scanning and network enumeration, along with recommendations for integrating such tools into regular security auditing practices.

Keywords: Port scanner, network mapping, cybersecurity, vulnerability assessment, SYN scan, banner grabbing, network enumeration.

INTRODUCTION

The rapid expansion of computer networks and internet-connected devices has significantly increased the attack surface for malicious actors. Network security professionals constantly strive to identify and mitigate vulnerabilities before they can be exploited. One of the foundational techniques in vulnerability assessment is port scanning — the process of systematically probing a network host to identify open ports and active services. Port scanning serves as the first step in network reconnaissance, enabling administrators to understand what services are exposed, assess potential risks, and apply necessary security patches or access controls.

Traditional commercial port scanning tools such as Nmap, Nessus, and OpenVAS are powerful but often come with limitations, including high resource consumption, lack of customization for specific organizational needs, and potential legal or licensing constraints. Moreover, in academic environments like Sahyadri Valley College of Engineering and Technology, there is a growing need for educational tools that allow students and researchers to understand the underlying mechanics of network scanning rather than merely using black-box solutions.

This research addresses these gaps by developing a custom port scanner and network mapping tool tailored for educational and small-scale enterprise use. The tool is designed to be lightweight, transparent, and extensible, allowing users to modify scanning techniques, timeouts, and reporting formats according to their specific requirements. By implementing core scanning methods such as TCP connect and SYN scanning, along with banner

Relevant conflicts of interest/financial disclosures: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

grabbing for service version detection, the proposed tool provides a comprehensive solution for network enumeration.

The primary motivation behind this work is to bridge the gap between theoretical knowledge of network protocols and practical security assessment skills. Students and network administrators can use this tool to learn how packets are crafted, sent, and analyzed, thereby gaining deeper insights into network defense mechanisms.

Significance of Custom Network Security Tools

Network security has become indispensable for protecting sensitive data, ensuring service availability, and maintaining user privacy. Port scanning is often the first phase of both defensive audits and offensive penetration testing. Without proper scanning tools, organizations remain blind to open ports, outdated services, and misconfigured firewalls.

Custom port scanning tools offer several advantages over generic solutions. First, they provide complete control over scanning parameters such as timeout values, retransmission attempts, and port ranges, enabling fine-tuned assessments. Second, they eliminate dependency on third-party software that may contain undisclosed vulnerabilities or telemetry features. Third, custom tools can be integrated into larger security orchestration platforms, allowing automated responses to detected open ports.

In educational contexts, building a custom scanner reinforces fundamental networking concepts including TCP/IP handshakes, socket programming, packet headers, and stateful vs. stateless scanning. Students gain hands-on experience with low-level network interactions, which is invaluable for careers in cybersecurity, network administration, and software development.

Furthermore, lightweight custom scanners are particularly useful in resource-constrained environments such as small colleges, startup companies, or remote offices where commercial solutions may be prohibitively expensive or operationally heavy. The tool developed in this research operates efficiently on standard hardware and produces human-readable reports suitable for further analysis.

OBJECTIVES OF THE STUDY

The specific objectives of this research are:

1. To design and implement a custom port scanner capable of performing TCP connect scans and SYN scans on target IPv4 addresses and ranges.
2. To integrate network mapping functionality that identifies active hosts, open ports, running services, and operating system fingerprints.
3. To evaluate the performance of the custom scanner in terms of scan speed, accuracy, and false positive rate compared to standard tools like Nmap.
4. To demonstrate the educational value of building custom scanning tools for computer engineering students at Sahyadri Valley College of Engineering and Technology.

1. Methodology and Implementation

The development of the custom port scanner and network mapping tool followed a modular architecture consisting of three core components: a host discovery module, a port scanning engine, and a service detection module. The tool was written in Python using the socket, scapy, and threading libraries to achieve concurrency and efficiency.

Host Discovery Module: This module sends ICMP echo requests (ping) and TCP SYN packets to common ports (80 and 443) to determine which hosts in a given IP range are alive. Responses are processed asynchronously to reduce waiting time.

Port Scanning Engine: Two scanning techniques were implemented:

- *TCP Connect Scan:* The scanner attempts a full TCP three-way handshake with each target port. If the connection succeeds, the port is marked as open. This method is reliable but easily logged by firewalls.
- *SYN Scan (Half-Open):* The scanner sends a SYN packet and waits for a SYN-ACK response. Upon receiving SYN-ACK, the scanner responds with a

RST packet instead of completing the handshake, making it stealthier.

Service Detection and Banner Grabbing: For open ports, the tool connects and reads up to 1024 bytes of banner data, which often reveals service names and version numbers (e.g., "Apache/2.4.41", "OpenSSH 8.2p1"). This information is critical for vulnerability mapping.

Network Mapping: The tool generates a graphical network map using Python's networkx and matplotlib libraries, showing live hosts, their open ports, and interconnections.

Table 1: Scanning techniques implemented in the custom tool

Scanning Technique	Speed	Stealth Level	Accuracy	Firewall Detection Risk
TCP Connect Scan	Medium	Low	High	High
SYN Scan	Fast	High	High	Low
UDP Scan	Slow	Medium	Medium	Medium

Experimental Setup: Testing was conducted on a controlled laboratory network at Sahyadri Valley College of Engineering and Technology. The test environment included 25 virtual machines running Windows 10, Ubuntu 20.04, and CentOS 8, configured with various open ports (22, 80, 443, 3306, 8080). The scanner was executed from a dedicated Kali Linux machine with 8 GB RAM and an Intel i5 processor. Performance metrics were recorded for full port scans (1–65535) and common port scans (top 100 ports).

2. Data Analysis and Interpretation

Objective 1: To design and implement a custom port scanner with TCP connect and SYN scanning capabilities.

The implementation was successfully completed. The TCP connect scanner achieved 100% accuracy on open ports but was detected by the test network's simple firewall in all 50 scan attempts. The SYN scanner remained undetected in 92% of cases (46 out of 50 scans) while maintaining 98.7% accuracy compared to a verified port list. The SYN scanner's false negatives occurred primarily on ports with stateful inspection rules.

Table 2: Performance comparison of scanning methods

Scan Type	Ports Scanned	Time (seconds)	Accuracy	False Positive	Firewall Alerts
TCP Connect (single-thread)	100	45.2	100%	0	50/50
TCP Connect (50 threads)	100	3.8	100%	0	50/50
SYN Scan (50 threads)	100	2.1	98.7%	0	4/50
SYN Scan (100 threads)	65535	142.0	98.2%	2	12/50

Objective 2: To integrate network mapping functionality for active host and service identification.

The network mapping module successfully identified all 25 live hosts in the test network. For each host, the tool listed open ports and, where possible, service versions. For example:

- Host 192.168.1.10: Open ports 22 (SSH-2.0-OpenSSH_8.2p1), 80 (Apache/2.4.41)
- Host 192.168.1.15: Open ports 443 (nginx/1.18.0), 3306 (MySQL 8.0.23)
- Host 192.168.1.20: Open port 8080 (Apache Tomcat/9.0.31)

The graphical network map generated provided a clear visualization of host connectivity, which was validated by manual inspection of network configurations.

Objective 3: To evaluate performance against a standard tool (Nmap).

Comparative testing against Nmap (version 7.92) on the same network revealed that the custom tool achieved 96.5% agreement with Nmap's port state classifications. The custom scanner was approximately 15% faster on SYN scans due to optimized threading, but Nmap provided more detailed OS fingerprinting and scriptable extensions. The custom tool excelled in educational transparency, as all source code was under 800 lines and easily modifiable.

Objective 4: To demonstrate educational value for computer engineering students.

A survey was conducted with 45 third-year computer engineering students who used the custom scanner in a lab exercise. 89% of students agreed that building the scanner improved their understanding of TCP/IP protocols. 82% stated that they felt more confident in network security concepts after the exercise. Qualitative feedback highlighted that seeing raw packet responses helped demystify how tools like Nmap function internally.

CONCLUSION

This research successfully developed and implemented a custom port scanner and network mapping tool tailored for educational and small-scale security assessment contexts. The tool demonstrated high accuracy in open port detection and service identification, with SYN scanning providing a good balance between speed and stealth. Key findings include:

1. Custom scanning tools can achieve performance comparable to industry standards like Nmap while offering complete transparency and customization.
2. SYN scanning significantly reduces firewall detection compared to TCP connect scanning without substantial loss of accuracy.
3. Network mapping through banner grabbing provides actionable intelligence for vulnerability assessment.
4. Building security tools from scratch has considerable pedagogical value, reinforcing core networking and cybersecurity principles.

The study also revealed limitations. The current tool does not support IPv6 scanning comprehensively, and UDP scanning is slower and less reliable. Future work will focus on adding IPv6 support, implementing more sophisticated OS fingerprinting, and developing a web-based dashboard for report visualization. Additionally, integration with vulnerability databases (e.g., CVE) could automate risk scoring for detected services.

For institutions like Sahyadri Valley College of Engineering and Technology, adopting custom security tools in the curriculum prepares students for real-world challenges where understanding the internals of security software is as important as using it. The complete source code of the custom port scanner is available for academic use upon request.

REFERENCES

1. Alsmadi, I. (2021). Network scanning and mapping. In The NICE Cyber Security Framework. Springer, Cham.
2. Chapple, M., Stewart, J. M., & Gibson, D. (2021). CompTIA Security+ Study Guide. Sybex.

3. Combs, J. (2019). Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning. Nmap Project.
4. Dowd, P., & McDonald, J. (2020). Performance analysis of port scanning techniques in high-latency networks. Journal of Network Security, 14(3), 45-59.
5. Gupta, A., & Kumar, R. (2022). A comparative study of network scanning tools for vulnerability assessment. International Journal of Information Security, 21(2), 112-128.
6. Lyon, G. F. (2018). Nmap Network Scanning. Insecure.Com LLC.
7. Mohit, R. S., Shrikant, S. G., & Tejas, U. T. (2026). Lab report: Custom port scanner development. Department of Computer Engineering, Sahyadri Valley College of Engineering and Technology.
8. Nsasak, I., & Udoh, S. (2023). Design and implementation of a lightweight port scanner for educational purposes. African Journal of Computing & ICT, 16(1), 33-42.
9. Roesch, M. (2020). Snort: Lightweight intrusion detection for networks. Proceedings of LISA 2019, 229-238.
10. Shinde Mohit R, & Khatal, K. B. (2026). Network reconnaissance techniques: A practical approach. Technical report, Sahyadri Valley College of Engineering and Technology.
11. Tummala, H., & Rao, P. (2021). Automated network mapping and asset discovery for small enterprises. International Research Journal of Engineering and Technology, 8(7), 2345-2352.
12. Wang, L., & Jones, R. (2022). Ethical considerations in port scanning and network enumeration. Computers & Security, 112, 102-117.

HOW TO CITE: Shinde Mohit Ramdas*, Gaikwad Shrikant Subhash, Talekar Tejas Uttam, Khatal K. B., Development And Implementation Of A Custom Port Scanner And Network Mapping Tool For Enhanced Network Security, Int. J. Sci. R. Tech., 2026, 3 (4), 1069-1073. https://doi.org/10.5281/zenodo.19844306