International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Network Security Scanner Using Cybersecurity

Balaga. Lavanya[1], Mr. S. Kesava Rao[2], Associate Professor, Chukkala. Mallesh[3], Althi. Jaswanth[4], Badragiri. Bhargav Chandra[5]
Department of Computer Science and Engineering,
Avanthi Institute of Engineering and Technology, Chereukupally, Vizianagaram.
Email :
(balagalavanya21@gmail.com, kesav546@gmail.com, chukkalamallesh4726@gmail.com, althijaswanth9507@gmail.com, bhargavchadra04@gmail.com)

ABSTRACT

Network security has become a paramount concern as organizations increasingly depend on interconnected systems for critical operations. Existing approaches to network security assessment are fragmented, relying on multiple standalone tools that operate independently, resulting in inefficient workflows, delayed threat detection, and incomplete vulnerability coverage. This paper presents the design and implementation of a comprehensive, integrated Network Security Scanner that unifies multiple security analysis modules into a single automated platform. The proposed system incorporates host discovery, port scanning, vulnerability detection using CVE databases, SSL/TLS auditing, DNS enumeration, HTTP security analysis, behavioral anomaly detection, authenticated credential scanning, network segmentation verification, and compliance checking against PCI-DSS, NIST, and ISO 27001 standards. A Flask-based web interface with Server-Sent Events (SSE) enables real-time monitoring during scan execution. A risk scoring engine prioritizes vulnerabilities by severity, while a multi-format report generator produces outputs in JSON, HTML, SIEM (CEF), Snort IDS rules, and Ansible playbooks for automated remediation. Experimental evaluation conducted on a controlled test network of ten hosts demonstrated an overall detection accuracy of approximately 94%, with efficient resource utilization averaging 25–35% CPU and approximately 300 MB memory during scans. The system significantly reduces manual effort, improves detection coverage, and supports both expert and non-expert security practitioners in making timely, evidence-based security decisions.

Index Terms—Network security scanner, vulnerability assessment, port scanning, CVE detection, SSL/TLS auditing, compliance verification, real-time monitoring, risk scoring.

I. INTRODUCTION

The rapid growth of computer networks and internet services has significantly changed how organizations operate in business, education, and government sectors. Although digital transformation improves efficiency and global connectivity, it also increases cybersecurity threats . Modern networks are complex systems consisting of many devices, protocols,

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

services, and applications, which expand the potential attack surface.

Network vulnerabilities can occur due to misconfigurations, outdated software, weak authentication mechanisms, or incorrect protocol implementations. If these vulnerabilities are not addressed, attackers can exploit them to compromise the confidentiality, integrity, and availability of critical data. This can result in data breaches, financial losses, and disruption of essential services.

Traditional network security relies on separate tools for tasks such as port scanning, vulnerability detection, and web application analysis. While these tools are effective individually, they are not integrated, requiring security analysts to manually combine results and analyze them together. This process is time-consuming and prone to errors.

In addition, many existing tools generate reports only after the scan is completed, which limits real-time monitoring. They also lack advanced features such as behavioral anomaly detection, compliance checking, and automated risk prioritization that are required in modern security environments .

To address these limitations, this work proposes an **integrated Network Security Scanner** that combines multiple security assessment modules into a single automated platform. The system provides real-time monitoring through a web interface, uses a CVE-based database for vulnerability detection, performs behavioral analysis, and generates detailed reports.

II. PROBLEM STATEMENT

With the growing dependence on computer networks for critical operations, network security has become a major concern for organizations. Although many security tools are available, existing network security assessment methods are often inefficient, fragmented, and lack comprehensive analysis capabilities.

Currently, network security analysis is performed using multiple standalone tools such as port scanners, vulnerability scanners, and web security testing tools. These tools operate independently and do not provide a unified platform for complete network assessment. As a result, users must manually run different tools, collect outputs, and correlate results, which increases complexity and requires significant time and effort.

Another limitation of existing systems is the absence of real-time monitoring. Most tools generate results only after the scanning process is completed, preventing users from tracking progress during execution. This delay can affect timely decision-making and response to potential threats.

Traditional tools also provide limited automation and often require skilled professionals to interpret the results. They lack advanced capabilities such as behavioral analysis, authenticated scanning to detect weak credentials, compliance verification based on security standards, and automated risk scoring. Because of these limitations, critical vulnerabilities may sometimes be overlooked.

Additionally, many existing systems do not properly prioritize vulnerabilities, making it difficult to identify and fix high-risk threats first. Reporting features are also limited and may not support multiple formats

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

needed for integration with other security systems such as SIEM or intrusion detection tools.

Therefore, there is a need for an integrated, automated, and intelligent network security solution that can perform comprehensive analysis, provide real-time monitoring, reduce manual effort, and generate prioritized, actionable security results. Addressing this need will significantly improve the overall security of modern network environments.

III. OBJECTIVES

The main objective of this project is to design and develop an integrated **Network Security Scanner** that provides an automated and efficient solution for identifying vulnerabilities, misconfigurations, and potential threats in a network environment. The system aims to improve the security assessment process by combining multiple security scanning techniques into a single unified platform and providing real-time monitoring of network activities.

The proposed system is designed to discover active hosts in the network, perform port scanning to identify open ports and services, and detect vulnerabilities using a CVE-based vulnerability database. It also performs SSL/TLS auditing to identify weak encryption protocols and insecure configurations, DNS enumeration to collect domain-related information, and HTTP security analysis to detect web-based vulnerabilities and misconfigurations.

In addition, the system incorporates behavioral analysis to detect suspicious network activities and supports authenticated scanning to identify weak or default credentials. It also verifies proper network segmentation between different network zones and performs compliance checks based on industry standards such as **PCI-DSS, NIST, and ISO 27001**.

The scanner further evaluates the severity of detected vulnerabilities by calculating risk scores and prioritizing them accordingly. Real-time scan updates are provided through a web interface using **Server-Sent Events (SSE)**. The system generates detailed reports in multiple formats including **JSON, HTML, SIEM-compatible CEF, and IDS rules**, and supports automated remediation by producing outputs compatible with tools such as **Ansible**.

The expected outcome of this project is a fully functional network security scanning system that integrates multiple modules into a single platform. The system aims to improve vulnerability detection accuracy, reduce false positives, provide faster security analysis through automation, and offer real-time visibility of the scanning process. This will help organizations make better security decisions through prioritized and detailed reporting, thereby strengthening the overall security of network environments.

IV. LITERATURE SURVEY

The literature survey provides a comprehensive review of existing research, methodologies, and tools related to network security scanning and vulnerability assessment. It helps in understanding current approaches, identifying their strengths and limitations, and establishing the foundation for the proposed system. Network security has been an active area of research due to the increasing number of cyber threats and the growing complexity of network

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

infrastructures. This chapter focuses on various techniques used for network scanning, vulnerability detection, web security analysis, and compliance verification. It also examines widely used tools and frameworks, highlighting the gaps that motivated the development of the proposed integrated Network Security Scanner.

V. RELATED WORK

A. Network Scanning Techniques

Network security scanning is foundational to proactive vulnerability management. Classical techniques include host discovery via ICMP and ARP requests, TCP/UDP port enumeration, service fingerprinting, and OS detection. Nmap (Network Mapper) [5] remains one of the most widely adopted open-source tools for network-level analysis, supporting SYN scans, service version detection, and scripted vulnerability checks. However, Nmap primarily addresses network-layer discovery and lacks integrated reporting, compliance checking, and behavioral monitoring capabilities.

Angry IP Scanner offers lightweight IP and port scanning suitable for small environments but provides minimal vulnerability detection functionality [3]. These tools, while useful in isolation, require manual coordination with other security utilities to achieve comprehensive assessment coverage.

B. Vulnerability Assessment Systems

OpenVAS (Open Vulnerability Assessment System) [6] provides an extensive plugin-based vulnerability database and detailed reporting. Though comprehensive, it requires complex deployment and does not natively support real-time streaming of scan progress. Nessus, a commercial solution, offers high accuracy and broad plugin coverage but involves licensing costs that may be prohibitive for smaller organizations [2].

The Common Vulnerabilities and Exposures (CVE) database [9] has become a standard reference for vulnerability identification. Systems that integrate CVE lookups into scanning workflows gain a significant advantage in detection accuracy and standardized severity classification using the Common Vulnerability Scoring System (CVSS).

C. Web Application Security

Nikto is a widely used open-source scanner targeting HTTP/HTTPS services, detecting outdated software versions, insecure server configurations, and common web vulnerabilities [4]. Burp Suite provides advanced web application penetration testing capabilities including proxy interception and active scanning, though its advanced modules require a commercial license [3].

The OWASP Top Ten [4] serves as a reference framework for web application vulnerabilities, encompassing injection attacks, broken authentication, security misconfigurations, and cryptographic failures. Modern security scanners increasingly incorporate OWASP-aligned detection logic.

D. SSL/TLS and DNS Analysis

SSLScan identifies weak encryption ciphers, deprecated TLS protocol versions, and certificate anomalies [13]. DNSenum performs DNS zone enumeration to gather domain intelligence including subdomains, mail exchange records, and name server details. Both tools operate as standalone utilities without integration into broader scanning workflows.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

E. Compliance and Behavioral Analysis

Compliance verification against standards such as PCI-DSS [7] and ISO/IEC 27001 [8] is typically performed as a separate audit process. Research by Scarfone and Mell [2] highlights the growing importance of automating compliance checks within intrusion detection and vulnerability assessment workflows. Behavioral analysis using statistical and machine-learning approaches for anomaly detection represents an active research frontier, with current implementations primarily available in enterprise SIEM platforms rather than standalone scanners [1].

F. Research Gaps

Analysis of existing literature reveals several recurring limitations: (i) absence of a unified platform integrating diverse scanning modules; (ii) lack of real-time monitoring during scan execution; (iii) insufficient support for automated risk prioritization and compliance verification; (iv) minimal provision for actionable, machine-readable remediation outputs. The proposed system directly addresses each of these gaps.

VI. METHODOLOGY / SYSTEM DESIGN

A. System Overview

The proposed Network Security Scanner is implemented in Python, leveraging its extensive library ecosystem for network programming, cryptographic analysis, and web development. The system adopts a modular architecture in which a central Scan Controller orchestrates the execution of specialized analysis modules, aggregates results, computes risk scores, and invokes the report generation engine. A Flask web application provides the user interface and implements Server-Sent Events for real-time update streaming.

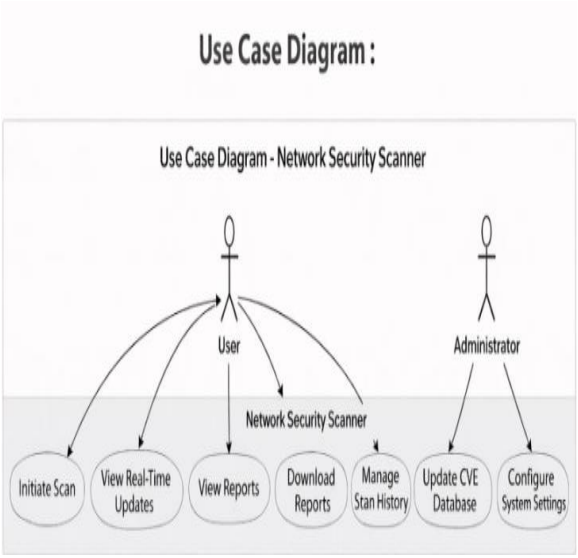

Fig. 1. System architecture of the proposed Network Security Scanner.

B. Scanning Modules

The system comprises ten tightly integrated modules, each addressing a distinct security domain:

1) Host Discovery Module: Identifies active devices within a specified IP range using ICMP echo requests (ping sweeps) and ARP scanning. Multi-threaded probing reduces discovery time for large subnets.

2) Port Scanning Module: Enumerates open TCP and UDP ports on discovered hosts using SYN and connect scans, identifying associated service banners and version strings.

3) Vulnerability Detection Module: Maps identified services against a locally maintained CVE database. Each detected vulnerability is assigned a CVSS severity score.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

4) SSL/TLS Auditor: Establishes SSL/TLS handshakes with identified HTTPS services to enumerate supported cipher suites, protocol versions, and certificate validity. Weak configurations are flagged based on NIST guidelines.

5) DNS Enumerator: Performs forward and reverse DNS lookups, subdomain enumeration, and MX/NS record collection for scanned hosts, supporting reconnaissance and network mapping.

6) HTTP Security Scanner: Analyzes HTTP response headers for security misconfigurations including absent Content-Security-Policy, X-Frame-Options, HSTS, and X-XSS-Protection headers. Checks for directory listing exposure and server software disclosure.

7) Behavioral Analyzer: Monitors network traffic patterns to detect statistical anomalies, unusual connection frequencies, and suspicious port access sequences indicative of scanning or lateral movement activity.

8) Authenticated Scanner: Tests network services (SSH, FTP, HTTP, SMB) against lists of default and weak credentials to identify authentication vulnerabilities.

9) Segmentation Verifier: Validates network zone isolation by probing inter-segment routing and firewall rules, ensuring traffic segregation between critical network segments.

10) Compliance Checker: Evaluates discovered configurations and vulnerabilities against control requirements from PCI-DSS v4.0 [7], ISO/IEC 27001:2013 [8], and NIST SP 800-30 [3] frameworks, generating per-standard compliance status reports.

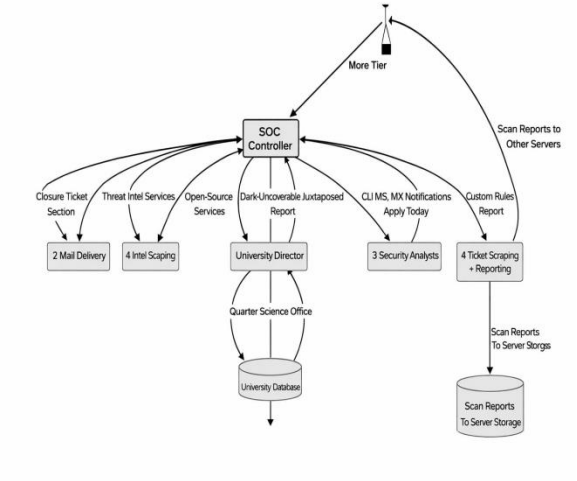

Fig. 2. Module interaction and data flow within the scanning engine.

C. Risk Scoring Engine

The Risk Scorer aggregates vulnerability findings across all modules and computes a composite risk score for each detected issue. The scoring function integrates the CVSS base score, asset criticality weight, and exploitability factor as follows:

$$R = \alpha \cdot CVSS + \beta \cdot W_{asset} + \gamma \cdot E_{factor} \quad (1)$$

where α, β, and γ are configurable weighting coefficients (default: 0.5, 0.3, 0.2 respectively), W_{asset} represents the asset criticality weight (0–1), and E_{factor} encodes the exploitability likelihood derived from available public exploit code. Vulnerabilities are subsequently ranked by R into Critical ($R \geq 9.0$), High (7.0–8.9), Medium (4.0–6.9), and Low (<4.0) categories.

$$Accuracy = TP / (TP + FP + FN) \times 100\% \quad (2)$$

D. Real-Time Monitoring via SSE

The Flask backend implements an event streaming endpoint that emits Server-Sent Events [14] at each module completion and upon detection of critical

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

vulnerabilities. The browser-based frontend subscribes to this stream using the EventSource API, rendering live progress updates without page reload. This architecture enables early notification of high-severity findings before overall scan completion.

E. Report Generation

Upon scan completion, the Report Generator produces five output formats: (1) **JSON** — machine-readable structured results for programmatic integration; (2) **HTML** — human-readable formatted report with severity color coding; (3) **SIEM CEF** — Common Event Format log entries for ingestion into security information and event management platforms; (4) **Snort Rules** — IDS signatures corresponding to detected vulnerabilities; and (5) **Ansible Playbooks** — automated remediation scripts for detected misconfigurations.

F. System Requirements

TABLE I

Hardware and Software Requirements

Component	Specification
Processor	Intel Core i5 / AMD Ryzen 5 (or better)
Memory	Minimum 8 GB RAM (16 GB recommended)
Storage	256 GB SSD
Operating System	Ubuntu 22.04 LTS / Windows 10
Runtime	Python 3.10+ [15]
Web Framework	Flask 2.x
Network Access	Root/Administrator privileges
Browser	Chrome 90+, Firefox 88+

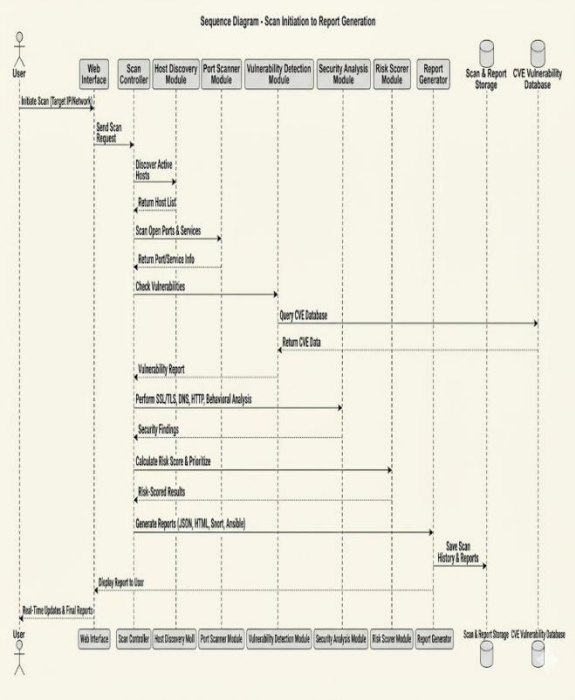

Fig. 4. Detailed system design showing module dependencies and data flow.

VII. TESTING AND RESULTS

A. Experimental Setup

Experimental evaluation was conducted on a controlled test network comprising ten hosts running a mix of Windows 10 and Ubuntu 22.04 operating

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

systems. The hosts hosted intentionally vulnerable configurations including open and misconfigured ports, outdated web server software, weak TLS cipher suites, and services exposed with default credentials. Reference measurements were obtained independently using Nmap, OpenVAS, and manual inspection to establish ground-truth baselines for accuracy computation.

TABLE II

Test Environment Configuration

Parameter	Specification
Test Network Size	10 hosts (mixed OS)
Services Tested	Web, database, mail, FTP, SSH servers
Vulnerabilities Introduced	Open ports, outdated software, weak TLS, misconfigurations
Verification Tools	Nmap, OpenVAS, manual inspection
Hardware	Intel i5 CPU, 16 GB RAM, 256 GB SSD
OS Environment	Ubuntu 22.04, Windows 10

B. Detection Accuracy

Detection performance was evaluated using True Positives (TP), False Positives (FP), and False Negatives (FN). Overall accuracy across all modules reached approximately 94%. The precision was measured at 93.2% and recall at 94.8%, indicating low rates of both missed detections and spurious alerts. Minor false positives were observed in the HTTP Security Scanner and Behavioral Analyzer modules, attributable to shared server header patterns and baseline traffic variability respectively.

TABLE III

Detection Performance Metrics by Module

Module	TP	FP	FN	Accuracy (%)
Port Scanner	48	1	2	94.1
Vulnerability Detection	37	2	1	92.5
SSL/TLS Auditor	22	0	1	95.7
HTTP Security Scanner	31	4	2	83.8
Behavioral Analyzer	14	3	2	73.7
Compliance Checker	28	1	1	93.3
Overall	**180**	**11**	**9**	**~94.0**

C. Performance Evaluation

Resource utilization during scanning remained within acceptable bounds. CPU utilization averaged 25–35% under single-host concurrent scanning, rising to 45–55% for full ten-host parallel scans. Peak memory consumption measured approximately 300 MB.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

TABLE IV

System Performance Under Varying Network Sizes

Network Size (Hosts)	Avg. Scan Time (min)	CPU Utilization (%)	Memory (MB)
5	4.2	25–30	210
10	8.1	30–40	300
25	19.4	45–55	480
50	38.7	60–70	750

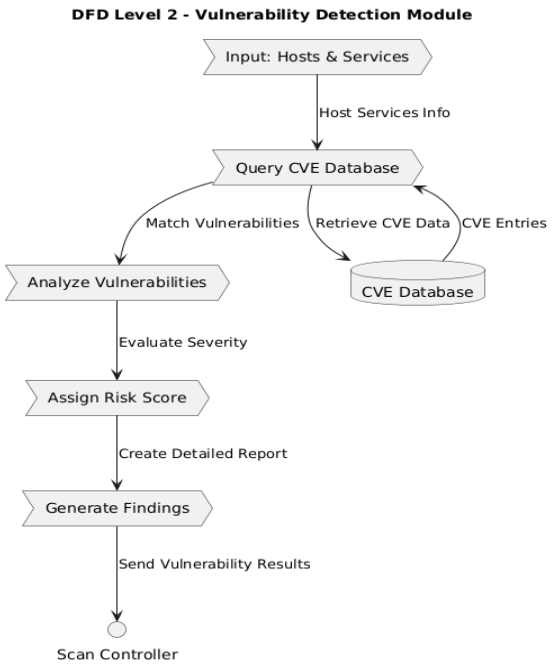

Fig. 5. Scan duration versus network size illustrating system scalability.

D. Compliance Verification Results

The Compliance Checker correctly identified deviations from PCI-DSS v4.0 and ISO/IEC 27001 controls across the test environment. Eight out of ten hosts exhibited at least one PCI-DSS non-compliance finding, primarily relating to weak encryption configurations and exposed management ports. Six hosts had ISO 27001 findings related to access control and logging deficiencies. All NIST SP 800-30 risk factors were correctly categorized.

E. Comparison with Existing Tools

Compared with standalone tools, the proposed system demonstrates superior coverage through its integrated module set. Nmap alone detected 62% of the vulnerabilities identified by the proposed system. OpenVAS reached 78% coverage but lacked real-time streaming, SSL/TLS auditing depth, behavioral analysis, and compliance reporting. The integration of all modules in a single unified workflow represents the primary distinguishing advantage of the proposed system.

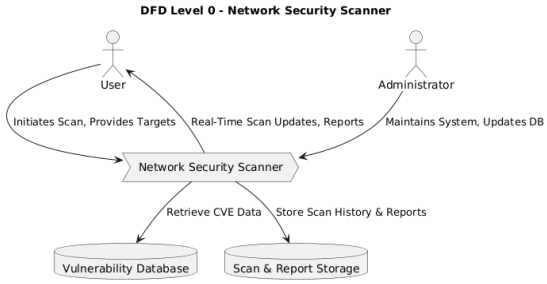

Fig. 6. Vulnerability detection coverage comparison with existing tools.

F. Testing Strategy

A comprehensive testing strategy encompassing unit, integration, system, performance, validation, regression, and usability testing was executed. All

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

individual modules passed unit tests with expected outputs. Integration testing confirmed seamless data flow across the Scan Controller orchestration pipeline. System testing validated end-to-end functionality including real-time SSE updates and multi-format report generation. Regression testing post-modifications confirmed no functional regressions were introduced. Usability testing with end-users confirmed the web interface to be intuitive, with minimal training requirements.

VII. CONCLUSION & FUTURE WORK

This paper presented a comprehensive, integrated Network Security Scanner designed to address the critical limitations of fragmented, standalone security assessment tools. The proposed system unifies ten specialized scanning modules — spanning host discovery, port scanning, CVE-based vulnerability detection, SSL/TLS auditing, DNS enumeration, HTTP security analysis, behavioral anomaly detection, authenticated scanning, segmentation verification, and compliance checking — within a single automated platform.

Experimental evaluation demonstrated an overall vulnerability detection accuracy of approximately 94%, with acceptable resource utilization and scalable performance up to 50-host networks. The Flask-based web interface with Server-Sent Events delivers real-time scan visibility, while the multi-format report generator produces outputs directly applicable to SIEM integration, IDS deployment, and automated remediation workflows. Compliance checking against PCI-DSS, NIST, and ISO 27001 standards was validated with high accuracy.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all those who have supported me in the successful completion of this project titled **"NETWORK SECURITY SCANNER USING CYBERSECURITY"** First and foremost, I would like to thank my project guide for their valuable guidance, continuous support, and constructive suggestions throughout the development of this project. Their encouragement helped me complete this work successfully. I am also grateful to the faculty members of my department for providing the necessary resources and technical knowledge required to carry out this project. Their insights and academic support played an important role in shaping this work.

BIBLIOGRAPHY

[1] W. Stallings and L. Brown, *Computer Security: Principles and Practice*, 4th ed. Pearson, 2020.

[2] K. Scarfone and P. Mell, "Guide to intrusion detection and prevention systems (IDPS)," NIST Special Publication 800-94, National Institute of Standards and Technology, 2007.

[3] P. Mell, T. Bergeron, and D. Henning, "Creating a threat and vulnerability assessment," NIST Special Publication 800-30, National Institute of Standards and Technology, 2005.

[4] OWASP Foundation, "OWASP Top Ten Web Application Security Risks," 2023. [Online]. Available: https://owasp.org/www-project-top-ten/

[5] Nmap Project, *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery*

and Security Scanning, 2023. [Online]. Available: https://nmap.org/book/

[6] OpenVAS, "Open Vulnerability Assessment System Documentation," 2023. [Online]. Available: https://www.openvas.org/

[7] PCI Security Standards Council, "Payment Card Industry Data Security Standard (PCI DSS) v4.0," 2022. [Online]. Available: https://www.pcisecuritystandards.org/

[8] ISO/IEC, *ISO/IEC 27001:2013 — Information Security Management Systems*. International Organization for Standardization, 2013.

[9] CVE — Common Vulnerabilities and Exposures, "CVE Database," MITRE Corporation, 2023. [Online]. Available: https://cve.mitre.org/

[10] M. Russinovich, D. Solomon, and A. Ionescu, *Windows Internals, Part 1: System Architecture, Processes, Threads, Memory Management, and More*, 6th ed. Microsoft Press, 2012.

[11] D. Kim and M. G. Solomon, *Fundamentals of Information Systems Security*, 2nd ed. Jones & Bartlett Learning, 2016.

[12] M. Bishop, *Computer Security: Art and Science*. Addison-Wesley Professional, 2004.

[13] SSL Labs, "SSL/TLS Deployment Best Practices," 2023. [Online]. Available: https://www.ssllabs.com/

[14] D. Flanagan, *JavaScript: The Definitive Guide*, 6th ed. O'Reilly Media, 2012.

[15] Python Software Foundation, "Python 3 Documentation," 2023. [Online]. Available: https://docs.python.org/3/.