International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Special Issue ICCIS-2025, pp 75-79, May-2025
https://doi.org/10.47001/IRJIET/2025.ICCIS-202511
Second International Conference on Computing and Intelligent Systems (ICCIS-2025)

Penetration Testing and Simulation for Vulnerability Assessment in Application Security

[1]T. Gnana Sudha, [2]N. Snehalatha, [3]A. Komala

[1,2,3]Department of CSE-Cybersecurity, Madanapalle Institute of Technology and Science, Madanapalle, AP, India
E-mails: [1]gnanasudha99@gmail.com, [2]snehalatha24102003@gmail.com, [3]komalaa@gmail.com

***Abstract* - This project is a security framework that includes a backend server, a web application firewall (WAF), and a penetration testing tool to protect web applications and identify vulnerabilities. The backend server handles data requests and responses securely using Flask. The WAF acts as a protective layer, checking requests for threats like XSS and SQL Injection, blocking malicious traffic, and forwarding safe requests to the backend. The penetration testing tool scans ports, detects services, grabs banners, and checks for known vulnerabilities like BlueKeep and SMBv3 RCE. It also generates reports with security findings and recommendations. Together, these components secure the application by blocking threats, identifying risks, and providing actionable insights for improvement.**

Keywords: Penetration Testing, Vulnerability Assessment, Application Security, SQL Injection, Cross-Site Scripting, Network Reconnaissance, Python, Cybersecurity, IP and Port Scanning, Security Testing, Automated Penetration Testing.

I. INTRODUCTION

This paper emphasizes the increasing prevalence of fake profiles on social media platforms, highlighting their potential to spread misinformation and disrupt online communities. The authors propose the use of machine learning algorithms for effective detection and mitigation of such fake profiles. By leveraging supervised learning and classification techniques, the research showcases the ability to identify fraudulent activities while maintaining user data integrity. The study stresses the importance of robust datasets and improved algorithmic designs for more accurate detection systems.

This paper addresses challenges and security weaknesses in web applications as the number of internet users grows. The authors discuss the need for rigorous vulnerability assessments and penetration testing to protect organizational data. It highlights the role of security analysts in safeguarding against cyber-attacks that could undermine trust and cause substantial reputational damage. The study emphasizes the importance of continuous security monitoring and the adaptation of security measures to mitigate emerging threats effectively.

This study focuses on the significance of vulnerability scanning tools for enhancing network security. The authors provide a comparative analysis of various tools and discuss their effectiveness in identifying and mitigating security threats. The paper highlights the need for organizations to proactively address security vulnerabilities to protect sensitive data and maintain operational integrity. It underscores the importance of regular network audits and the strategic use of vulnerability assessment tools to prevent potential cyber-attacks.

The paper discusses web application security and the critical role of penetration testing in identifying vulnerabilities. It examines both manual and automated testing methodologies to safeguard web applications from malicious attacks. The study emphasizes the importance of understanding and addressing common security vulnerabilities like SQL injection and cross-site scripting to protect sensitive information and ensure data integrity.

This study investigates the application of dynamic analysis and conducting penetration testing to uncover security vulnerabilities in web applications. It introduces the Tainted Mode Model (TMM), which enhances the identification of inter-module vulnerabilities and integrates penetration testing with dynamic analysis for more effective vulnerability management. The study highlights the importance of these techniques in developing secure web applications and protecting user data from potential security threats.

In this paper, the authors discuss the development of a vulnerability scanning approach that combines SQL Injection and Cross-Site Scripting detection with crawling technique. It emphasizes the need for automated tools to enhance the identification effectiveness and reliability web application protection flaws. The study highlights the use of these tools in identifying weaknesses that could be exploited by attackers and stresses the importance of timely detection and remediation to maintain security.

This paper presents a responsible methodology for penetration testing utilizing the Metasploit framework. It outlines a comprehensive process for testing system and network vulnerabilities, from information gathering to exploitation and post-exploitation analysis. The study emphasizes the significance of ethical hacking as a strategy to enhance protection measures and prevent unauthorized access to systems.

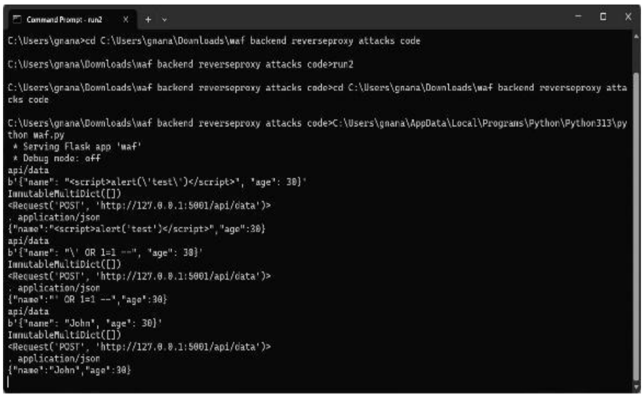
II. RELATED WORK

Korlam et al. utilized various tools like Acentuix and Burp Suite for detect common security weaknesses in web applications. Their study emphasized role of a blend of automated and manual penetration testing in improving protection posture of web applications. The research showcased how different tools are leveraged to detect a broad spectrum of security threats, including CSRF, XSS, and SQL injection, thereby highlighting the robustness of their approach in a real-world setting. Khushboo and Sharma et al. explored Red Team assessments, focusing on their effectiveness in identifying vulnerabilities within information systems. Their methodology included a combination of VAPT, emphasizing use of Red, Blue, and Purple Teams. The study demonstrated how multi-layered testing enhances the security and resilience of systems against various cyber threats, underscoring the comprehensive nature of Red Team strategies. George et al. examined the design and deployment of a web application security scanner aimed at preventing vulnerabilities. This research highlighted the effectiveness of the scanner in detecting and mitigating risks associated with SQL injection and cross-site scripting(CSS). The paper emphasized the importance of accessible security tools for individuals and small businesses, showcasing the scanner's role in enhancing cybersecurity awareness and protection. Bhoure et al. utilized the NMAP tool within their Vulnerability Management and Analysis Framework to identify security weaknesses in network configurations. Their study highlighted the integration of CVE and CVSS databases to assess and prioritize threats, illustrating how structured vulnerability management can significantly reduce potential breaches in network security. Kharat and Chawan et al. presented a comprehensive vulnerability management system that leverages automated tools to detect and manage software vulnerabilities. The study included an in-depth analysis of common cyber threats such as phishing and DDoS attacks, and how a structured vulnerability management approach can mitigate these risks. The research underscored the significance of ongoing vulnerability assessment along with the deployment of robust security policies. Somani and Kulkarni et al. emphasized the development of a Vulnerabilities Management System (VMS) to address software vulnerabilities in firmware and other electronic systems. Their research focused on the integration of advanced detection mechanisms to improve the security measures of electronic devices, illustrating the critical role of vulnerability management in protecting against unauthorized access and cyberattacks. Nair and Ansari et al. explored the consequences of incorporating AI into cybersecurity tools and systems. Their study highlighted how AI enhances identifying and responding to threats while also presenting new vulnerabilities like data poisoning and adversarial attacks. The research advocated for the development of AI-resilient security measures to protect against sophisticated cyber threats. Yaqoob et al. carried out research on penetration testing and security risk evaluation methodologies. Their research provided a detailed overview of common network threats and the effectiveness of various penetration evaluation approaches, like black, white, and gray boxes testing. The paper emphasized the critical role of vulnerability assessments in identifying and mitigating security risks in network environments.

III. METHODOLOGY

3.1 Data Loading and Initial Exploration

Network Configuration and Target Specification: Target IP addresses and port ranges are defined for scanning. Initial configurations for the WAF and backend server are set up, including specifying URLs and ports. Initial Network Scanning: Using the Nmap tool integrated into the system, initial scans are performed to detect open ports across specified IP ranges. This step is crucial for identifying potential entry points for security assessments.

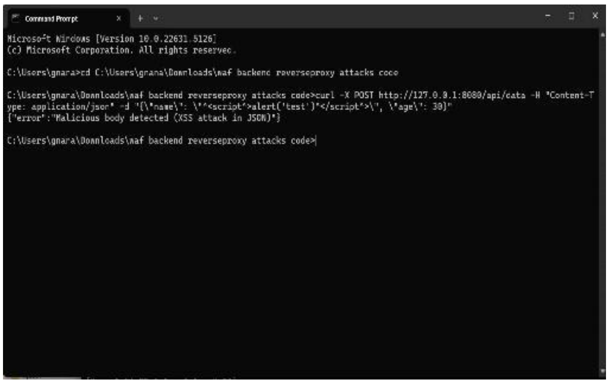


```
C:\Users\gnana>cd C:\Users\gnana\Downloads\waf backend reverseproxy attacks code

C:\Users\gnana\Downloads\waf backend reverseproxy attacks code>run2

C:\Users\gnana\Downloads\waf backend reverseproxy attacks code>cd C:\Users\gnana\Downloads\waf backend reverseproxy atta
cks code

C:\Users\gnana\Downloads\waf backend reverseproxy attacks code>C:\Users\gnana\AppData\Local\Programs\Python\Python313\py
thon waf.py
 * Serving Flask app 'waf'
 * Debug mode: off
api/data
b'{"name": "<script>alert(\'test\')</script>", "age": 30}'
ImmutableMultiDict([])
<Request('POST', 'http://127.0.0.1:5001/api/data')>
. application/json
{"name":"<script>alert('test')</script>","age":30}
api/data
b'{"name": "\' OR 1=1 --", "age": 30}'
ImmutableMultiDict([])
<Request('POST', 'http://127.0.0.1:5001/api/data')>
. application/json
{"name":"' OR 1=1 --","age":30}
api/data
b'{"name": "John", "age": 30}'
ImmutableMultiDict([])
<Request('POST', 'http://127.0.0.1:5001/api/data')>
. application/json
{"name":"John","age":30}
```

3.2 Statistical Analysis and Visualization

Service Identification: For each open port, banner grabbing is conducted to identify the service running on the port. This helps in gathering detailed data about each service, which is essential for subsequent vulnerability assessment. Vulnerability Assessment: Based on the service information, known vulnerabilities associated with these services are identified possibly from existing databases or predefined lists. The assessment includes checking against known vulnerabilities like CVEs (Common Vulnerabilities and

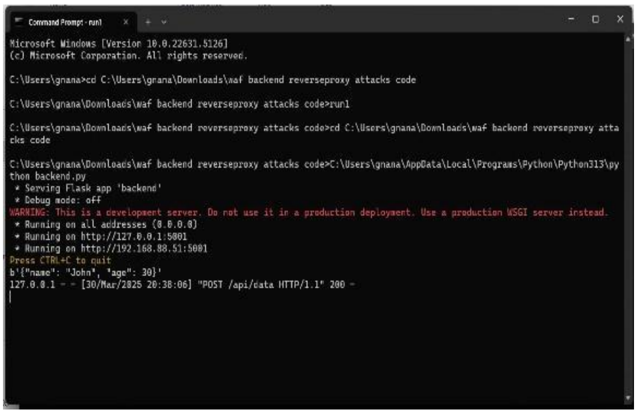
Exposures).Data Visualization: Visualize the distribution of open ports and services using graphical representations to quickly identify highly exposed services or unusual activity. Vulnerabilities and their severity can also be visualized to prioritize response strategies.



```
Microsoft Windows [Version 10.0.22631.5126]
(c) Microsoft Corporation. All rights reserved.

C:\Users\gnara>cd C:\Users\gnana\Downloads\waf backend reverseproxy attacks code

C:\Users\gnara\Downloads\waf backend reverseproxy attacks code>curl -X POST http://127.0.0.1:8080/api/data -H "Content-T
ype: application/json" -d "{\"name\": \"^<script^>alert('test')^</script^>\", \"age\": 30}"
{"error":"Malicious body detected (XSS attack in JSON)"}

C:\Users\gnara\Downloads\waf backend reverseproxy attacks code>
```

3.3 Data Preprocessing

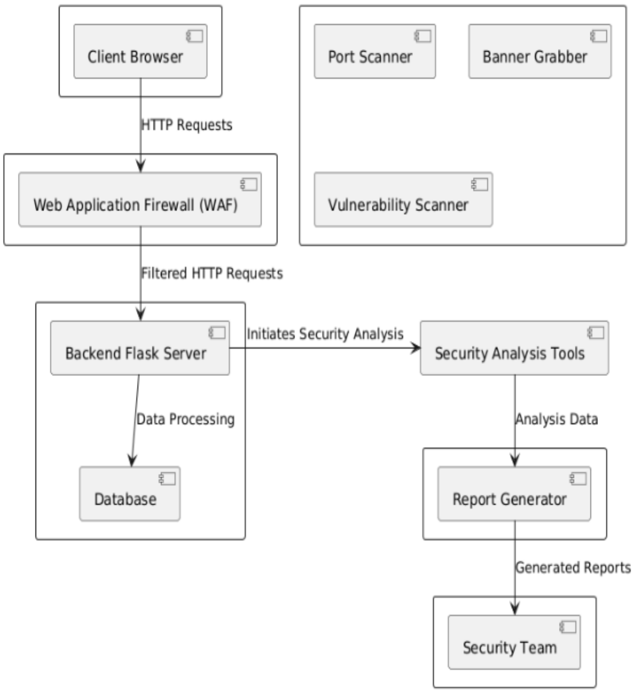
Data Structuring for Analysis: Data collected from scans and assessments are structured into a format suitable for analysis and reporting. This may include converting raw output from tools like Nmap into a more structured form such as JSON or a structured log. Security Analysis: Analyze the structured data to identify patterns or correlations between different types of services and vulnerabilities. This step may involve more sophisticated data processing techniques if the system integrates machine learning or statistical analysis tools. Response and Mitigation Strategy Formulation: Based on the analysis, formulate response strategies for mitigated identified vulnerabilities. Recommendations are generated for each identified issue, tailored to the specific configuration and vulnerability of the service.



```
Microsoft Windows [Version 10.0.22631.5126]
(c) Microsoft Corporation. All rights reserved.

C:\Users\gnana>cd C:\Users\gnana\Downloads\waf backend reverseproxy attacks code

C:\Users\gnana\Downloads\waf backend reverseproxy attacks code>run1

C:\Users\gnana\Downloads\waf backend reverseproxy attacks code>cd C:\Users\gnana\Downloads\waf backend reverseproxy atta
cks code

C:\Users\gnana\Downloads\waf backend reverseproxy attacks code>C:\Users\gnana\AppData\Local\Programs\Python\Python313\py
thon backend.py
 * Serving Flask app 'backend'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5001
 * Running on http://192.168.88.51:5001
Press CTRL+C to quit
b'{"name": "John", "age": 30}'
127.0.0.1 - - [30/Mar/2025 20:38:06] "POST /api/data HTTP/1.1" 200 -
```

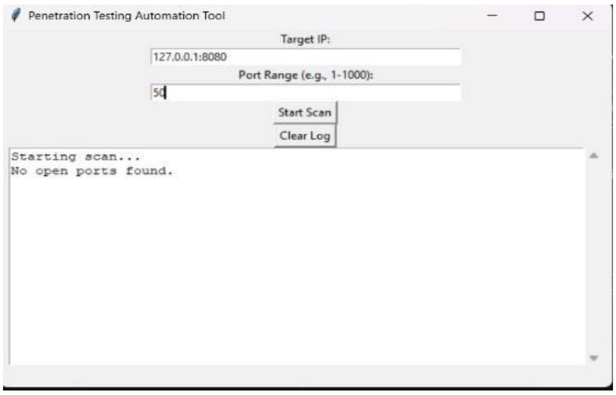
3.4 Reporting and Feedback

Report Generation: Generate detailed reports summarizing the findings from the scans, including identified services, detected vulnerabilities, and recommended mitigation strategies. Reports are formatted in a user-friendly manner, potentially including HTML or PDF formats, to aid in dissemination and review. Feedback Mechanism: Implement a feedback mechanism to adjust scanning parameters based on previous findings or emerging threat landscapes. This could involve adjusting the depth of scans, the range of ports scanned, or the specifics of vulnerability assessments.



3.5 Continuous Monitoring and Update

WAF Integration and Continuous Monitoring: The WAF continuously monitors incoming traffic to the backend server, inspecting requests for potential threats using patterns for XSS and SQL Injection. Detected threats are logged and can trigger alerts or automated responses based on their severity. System Updates and Patch Management: Regular updates to the vulnerability database and scanning tools to include the latest threat information and security patches. Guarantee that the system's security protocols adapt to emerging vulnerabilities and attack vectors.



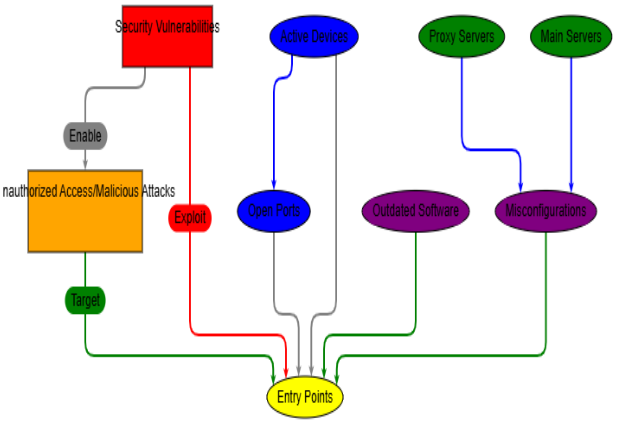
Penetration Testing Automation Tool

Target IP: 127.0.0.1:8080

Port Range (e.g., 1-1000): 50

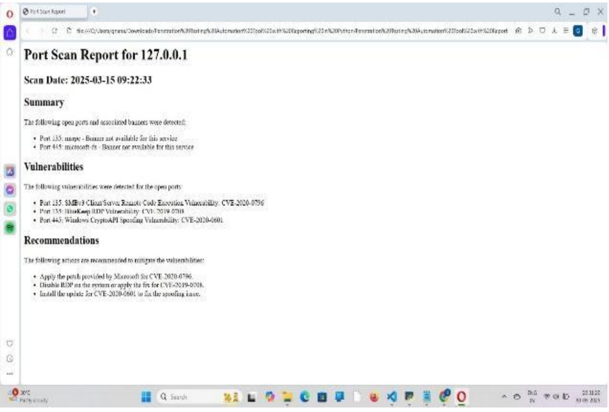
```
Starting scan...
No open ports found.
```

IV. ARCHITECTURE



V. RESULT FINDINGS AND DISCUSSION

The "Results Findings and Discussion" segment is essential in interpreting outcomes of the research and connecting them with the study's objectives. In the context of this paper, the results demonstrate the efficiency of the integrated security framework, which includes Web Application Firewall, a backend server, a penetration testing tool. The penetration testing process successfully identified protection weaknesses while the WAF effectively filtered malicious HTTP requests. The findings highlight that automated vulnerability scanning, coupled with real-time traffic monitoring, significantly enhances application security by identifying high-risk exposure points and preventing unauthorized access. The banner grabbing process offered in-depth insights into services operating on open ports, aiding in map potential attack vectors accurately. The vulnerability assessment showed common weaknesses related to known exploits like BlueKeep and SMBv3 RCE, emphasizing the need for regular patching and updates.



Port Scan Report for 127.0.0.1

Scan Date: 2025-03-15 09:22:33

Summary

The following open ports and associated banners were detected:

- Port 135: msrpc - Banner not available for this service
- Port 445: microsoft-ds - Banner not available for this service

Vulnerabilities

The following vulnerabilities were detected for the open ports:

- Port 135: SMBv3 Client/Server Remote Code Execution Vulnerability: CVE-2020-0796
- Port 135: BlueKeep RDP Vulnerability: CVE-2019-0708
- Port 445: Windows CryptoAPI Spoofing Vulnerability: CVE-2020-0601

Recommendations

The following actions are recommended to mitigate the vulnerabilities:

- Apply the patch provided by Microsoft for CVE-2020-0796.
- Disable RDP on the system or apply the fix for CVE-2019-0708.
- Install the update for CVE-2020-0601 to fix the spoofing issue.

In the discussion, it is evident that the integrated approach of penetration testing and WAF implementation ensures proactive threat mitigation. The results align with previous research, confirming that combining dynamic analysis with penetration testing enhances vulnerability detection. Furthermore, the continuous monitoring provided by the WAF ensures adaptive security against evolving threats. The generated reports offered actionable recommendations, such as implementing secure coding practices and configuring network parameters to reduce attack surfaces. One limitation noted was that the automated scanning might not detect highly sophisticated or zero-day vulnerabilities, suggesting the need for complementary manual testing and advanced machine learning techniques for deeper analysis. Overall, the study demonstrates that a multi-layered security framework not only identifies and mitigates existing vulnerabilities but also establishes a robust defense mechanism capable of adapting to new threats, thereby significantly improving application security and resilience.

VI. FUTURE WORK

The "Future Work" section outlines potential directions for expanding and improving the current research. In the context of this research, future directions could focus on enhancing the penetration testing framework by integrating advanced machine learning models for predictive vulnerability analysis. Machine learning models can help detect zero-day weaknesses and sophisticated attack patterns by analysing past data and evolving risk environments. Additionally, incorporating advanced neural network methods could further enhance the precision of vulnerability identification, especially in identifying complex inter-module dependencies that traditional tools might miss.

Another promising direction is the development of adaptive Web Application Firewalls (WAFs) that leverage artificial intelligence to dynamically update threat signatures and blocking rules in real-time. Such adaptive systems would not only respond to known threats but also predict and mitigate emerging vulnerabilities without manual intervention. Moreover, expanding the penetration testing capabilities to include cloud-based applications and Internet of Things (IoT) devices would make the framework more comprehensive, addressing the growing demand for security in distributed and heterogeneous environments.

Future work could also involve automating the feedback loop between the penetration testing tool and the WAF, enabling real-time adjustments based on detected threats. This would create a self-healing security system that continuously learns and adapts to new attack vectors. Additionally, incorporating threat intelligence feeds can deliver real-time updates on global cybersecurity. trends, enhancing the system's ability to pre-emptively address potential risks. Finally, conducting large-scale case studies across various industries would validate the framework's scalability and effectiveness in real-world scenarios, ensuring that it remains robust, flexible, and capable of addressing the evolving challenges in application security.

REFERENCES

[1] Urshila Ravindran, Raghu Vamsi Potukuchi. "A Review on Web Application Vulnerability Assessment and Penetration Testing." *Review of Computer Engineering Studies*, Vol. 9, No. 1, March 2022, pp. 1-22. DOI: 10.18280/rces.090101.

[2] Dipali N Railkar, Prof. Dr. Shubhalaxmi Joshi. "A Study on Vulnerability Scanning Tools for Network Security." *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, Vol. 8, Issue 6, January-February 2022, pp. 340-350.

[3] Dr. T. Pandikumar, Tseday Eshetu. "Detecting Web Application Vulnerability using Dynamic Analysis with Penetration Testing." *International Research Journal of Engineering and Technology (IRJET)*, Vol. 3, Issue 10, October 2016, pp. 430.

[4] Trupti Bhosale, Shraddha More, Prof. S.N. Mhatre. "Testing Web Application using Vulnerability Scan." *International Research Journal of Engineering and Technology (IRJET)*, Vol. 6, Issue 5, May 2019, pp. 265.

[5] Seema Rani, Ritu Nagpal. "PENETRATION TESTING USING METASPLOIT FRAMEWORK: AN ETHICAL APPROACH." *International Research Journal of Engineering and Technology (IRJET)*, Vol. 6, Issue 8, August 2019, pp. 538.

[6] Korlam Sai Rajesh, Dr. M. Seshashayee. "Bug Hunting using Web Application Penetration Testing techniques." *International Research Journal of Engineering and Technology (IRJET)*, Vol. 6, Issue 3, March 2019, pp. 5412.

[7] Khushboo Amin, Dr. Priyanka Sharma. "Red Team Analysis of Information Security Measures and Response." *International Research Journal of Engineering and Technology (IRJET)*, Vol. 7, Issue 4, April 2020, pp. 4279.

[8] Binny George, Jenu Maria Scaria, Jobin B, Praseetha VM. "Web Application Security Scanner for Prevention and Protection against Vulnerabilities." *International Research Journal of Engineering and Technology (IRJET)*, Vol. 7, Issue 5, May 2020, pp. 6267.

[9] Pravin Kharat, Prof. Pramila M. Chawan. "Vulnerability Management System." *International Research Journal of Engineering and Technology (IRJET)*, Volume 8, Issue 11, November 2021, pp. 25-26.

[10] Pranav Nair, Meraj Farheen Ansari. "Vulnerabilities in AI Systems: The Integration of AI into Cybersecurity Tools and Systems." *International Research Journal of Engineering and Technology (IRJET)*, Volume 11, Issue 7, July 2024, pp. 1159-1160.

Citation of this Article:

T. Gnana Sudha, N. Snehalatha, & A. Komala. (2025). Penetration Testing and Simulation for Vulnerability Assessment in Application Security. In proceeding of Second International Conference on Computing and Intelligent Systems (ICCIS-2025), published in *IRJIET*, Volume 9, Special Issue ICCIS-2025, pp 75-79. Article DOI https://doi.org/10.47001/IRJIET/2025.ICCIS-202511

****