International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Penetration Testing: An Overview of its Tools and Processes

Jeffin Varghese John[1], Dr. Kala K U[2]

Department of Computer Application (BCA Cybersecurity), Jain-Deemed-To- Be-University, Bangalore[1,2]
Assistant Professor, Department of Computer Application (BCA Cybersecurity), Jain-Deemed- To- Be-University, Bangalore[1,2]
Email: jeffvjohn300803@gmail.com, kala.ku@jainuniversity.ac.in

ABSTRACT –

In recent years, numerous inquiries have explored security testing within corporate environments, networks, and systems. Understanding the progression of methodologies and tools in security testing has become crucial as a result. One significant driver of this evolution is the penetration test, often abbreviated as Pentest. This research aims to provide a comprehensive overview of Pentest, clarifying its application scenarios, models, methodologies, and tools as detailed in published literature. The goal is to equip researchers and security practitioners with insights into Pentest-related aspects and existing solutions.

Through a systematic mapping study, we initially collected 1145 papers, eventually scrutinizing 1090 unique ones. Subsequently, we meticulously selected 54 primary studies for quantitative and qualitative analysis. Our findings include a categorization of the tools and models employed in Pentest, as well as an outline of the main scenarios where these methodologies and tools are applied. Additionally, we pinpoint prevalent unresolved issues and suggest potential avenues for further research in the Pentest domain.Top of Form

Keywords – networks, tools, research, security, analysis.

I. INTRODUCTION

The vulnerabilities confronting companies, organizations, and entities managing sensitive data are conspicuously evident. Frequently, these entities struggle to grasp the intricacies of their communication structures, resulting in limited control. Moreover, the risks heighten when considering the applications operating on their computing infrastructures, potentially leading to significant financial repercussions.

Security measures typically encompass preventive, detective, and responsive actions. Prevention aims to thwart unauthorized access to system resources, detection intervenes when breaches occur, and response endeavors to minimize further damage or unauthorized entry.

Nonetheless, ongoing security assessment is imperative for comprehending existing risks. This is commonly achieved through security testing, underscoring the importance of employing suitable techniques for risk mitigation within any organization.

One effective method for security assessment is penetration testing (Pentest). Pentest entails deliberate attempts to breach a system or network to identify vulnerabilities, employing techniques akin to those used by hackers. This proactive approach facilitates the addressing of vulnerabilities before they can be exploited by unauthorized individuals.

Various common attacks include denial of service (DoS), remote-to-user (R2L), user-to-root (U2R), and probing, each with specific objectives and methodologies. Pentest activities typically encompass data gathering on the target system, scanning for available services/protocols, identifying running systems and applications, and exploiting known vulnerabilities.

Pentest not only serves to evaluate system security but also assesses the effectiveness of security teams and overall security processes within a company. Factors such as pre-existing information levels, test depth, scope, and employed techniques must be taken into account when conducting Pentest.

The objective of this paper is to raise awareness and enhance the practice of network penetration testing. Furthermore, it aims to increase awareness among organizations that have been or may become victims of cybercrime due to their employees' use of technology.

1.1. Types of Penetration Tests

Numerous authors have delineated three approaches to penetration testing [4]. These commonly acknowledged approaches encompass black box, white box, and gray box testing.

1.1.1. Black Box

In black box testing, testers simulate attacks without prior knowledge of the infrastructure. Through their own methods and tools, testers aim to uncover all vulnerabilities. This entails employing various genuine attack techniques such as social engineering and remote access. For instance, testers might obtain only the network's IP address without additional details, and proceed to simulate various attack methods to identify both known and unknown vulnerabilities within the network.

1.1.2. White Box

In white box testing, testers conduct simulations with full access to information about the infrastructure, including operating system specifics, IP addresses, and certain passwords. This approach enables testers to execute attacks leveraging comprehensive knowledge about the target system, such as internal employee personal details. By doing so, the integrity of the organization's network infrastructure is safeguarded, minimizing the potential risk posed by internal threats, such as disgruntled employees.

1.1.3. Gray Box

The gray box approach combines elements of both white and black box testing, integrating internal and external security information. Testers possess partial knowledge about the network infrastructure in this method. Gray box testing aims to address and mitigate both internal and external security vulnerabilities that could be exploited by attackers [5].

1.2. Impact of Hacking on Organizations and Governments

Given the pervasive role of technology in both business and government, safeguarding against attacks has become paramount to protect customers' personal and financial data, particularly as internal threats, such as disgruntled employees, are prevalent. The repercussions of electronic attacks are staggering, resulting in significant financial losses, tarnished reputations, and legal liabilities for affected organizations. Researchers [6] highlight the substantial financial losses reported by hacked companies, citing the 2011 breach of Sony's PlayStation system, which incurred approximately USD 170 million in losses, with recovery proving to be a formidable challenge. Additionally, piracy poses risks of information loss through deletion or alteration of critical files, evidenced by attacks on servers at institutions like the FBI, Interpol, and NASA over the past decade. The fallout from such breaches includes severe damage to reputation, deterring customers from engaging with hacked companies due to concerns about data security, leading to long-term loss of business. Consequently, there has been a surge in demand for IT security services, with penetration testing emerging as a crucial preventive measure in cybersecurity, according to the findings of [6]. The impact of hacking on the financial health and reputation of organizations is exemplified by T-Mobile's experiences. Notably, a data breach in May 2023 affected approximately 800 customers, exacerbating the damage to the company's reputation following earlier breaches earlier that year and in 2022, which collectively cost the company USD 350 million. Therefore, it is imperative for T-Mobile and similar entities to fortify their networks and prioritize employee awareness [7].

1.3. Standards of the Penetration Test

Cyber attackers employ various attack vectors against their targets, taking advantage of ineffective policies and standards, thereby breaching systems and pilfering valuable information. In response, penetration testers utilize certain standards as preventive measures to deter such attacks. The common standards are [**8**]:

1.3.1. Information Systems Security Assessment Framework (ISAAF)

The goal of this standard is to evaluate the application, system, and network controls. There are three phases [**8**]:

- Planning and preparation;
- Assessment; and
- Reporting.

1.3.2. National Institute of Standards and Technology Special Publication 800-115 (NIST SP 800-115)

The NIST standard (SP800-115) offers guidelines for structuring and executing information security testing and assessments. Furthermore, it emphasizes the importance of evaluating results and devising mitigation strategies. While not exhaustive, the standard aims to offer a comprehensive overview of the primary elements of security testing and assessments, with a focus on specific methodologies and the identification of their strengths and weaknesses. It also provides reports and recommendations for implementation. According to the NIST standard (SP800-115), the penetration testing process consists of four key stages: planning, detection, attack, and reporting [8].

1.3.3. Open-Source Security Testing Methodology Manual (OSSTMM)

This manual offers best practices to ensure network security, providing an overview of cybersecurity measures and optimal solutions tailored to the technological environment, aiding in informed decision-making for network protection. This edition was released in 2010 [8].

1.3.4. Penetration Testing Execution Standard (PTES)

Interactions before engagement: the standard ensures that users are prepared for the pentest. Everything revolves around the release of documents and test-related equipment:

- Gathering information;
- Threat modeling;
- Vulnerability analysis;
- Exploitation; and
- Reporting.

1.4. Penetration Testing Tools

Penetration testing entails simulating diverse attack scenarios to pinpoint existing vulnerabilities within a system, employing an array of indispensable tools. Researchers have extensively examined various tools, including:

• Aircrack-ng: A comprehensive suite designed for assessing WiFi network security, encompassing areas such as detection, packet sniffing, and analysis, with a focus on cracking WEP and WPA/WPA2-PSK in 802.11 wireless LANs [6].

• Nmap: A network mapper serving as a penetration-testing tool, utilized for scanning networks to identify ports, hosts, operating systems, and services, facilitating the discovery of vulnerabilities. It is often deployed in the initial phase of penetration testing and is adept at scanning both large and small networks, supporting numerous protocols and systems [9].

• Metasploit: An open-source penetration tool enabling the testing of vulnerabilities in operating systems and applications by executing a predefined set of codes on the target. It provides a framework for penetration testing and is compatible with Linux, Apple Mac OS X, and Microsoft Windows platforms [10].

• BeEF (Browser Exploitation Framework): Tailored for web browsers, BeEF assesses exploitability within the context of web browsers across various operating systems, including Linux, Apple Mac OS X, and Microsoft Windows [10].

• Shadow: A search engine designed to identify specific devices and their types by scanning the entire Internet and analyzing the banners returned by scanned devices. It reveals details such as versions of web servers and anonymous FTP servers at specific locations, along with device model information [1].

• Nessus: A remote advanced scan tool commonly employed in penetration testing, operating from one machine to scan services offered by a remote counterpart. It boasts usage in over 75,000 organizations worldwide [10].

• Wireshark: An open-source program compatible with UNIX, Windows, and several other operating systems, utilizing a graphical user interface and serving as a network sniffer. Wireshark functions as a passive troubleshooting tool for network issues, capturing and analyzing packet traffic discreetly [1].

• Zed Attack Proxy (ZAP): A straightforward and free security solution integrating penetration testing to detect vulnerabilities in web applications. ZAP is particularly well-suited for developers and functional testers new to penetration testing, accommodating individuals across a broad spectrum of security expertise.

• Netcat: A command-line tool leveraging TCP or UDP protocols for reading and writing data over network connections. Regarded as a potent asset in the arsenal of network and system administrators [11].

1.5. Importance of Manual Penetration Testing versus Automated Penetration Testing

Computer systems lack the inherent intelligence to dictate developer behavior; they merely execute commands as programmed. When developers make logical errors in their programs, it can lead to business logic vulnerabilities. Consequently, manual penetration testing, reliant on human insight, remains essential as it can uncover vulnerabilities overlooked by automated scanners. Additionally, manual testing is adept at adapting to changing requirements, unlike automated tests which may falter when old implementations become obsolete. Moreover, manual testing is indispensable for detecting rare vulnerabilities and mitigating the high probability of false positives and false negatives inherent in automated testing. By delving into the intricacies of developers' security techniques, manual testing can effectively reduce the occurrence of false positives in vulnerability detection [12].

Therefore, the objectives of this study include:

- Evaluating the tools utilized in network penetration testing;

- Assessing network penetration testing methodologies;

- Identifying potential attacks on all open ports; and

- Reviewing mitigation techniques employed to safeguard open ports against threats.

2. WLAN AND PENETRATION TESTING

2.1. Overview

We are currently experiencing a digital transformation era heavily reliant on both wired and wireless networks for communication and information sharing across devices. These network-based technologies have seamlessly integrated into the operations of government and private organizations across various sectors, including education, healthcare, commerce, manufacturing, and more. Moreover, individuals also heavily rely on these technologies in their daily lives, evident in activities such as social media usage, further enhancing interactivity and efficiency.

However, with this increased reliance on networks comes heightened vulnerability to security threats, as attackers target these networks with malicious intent. Such security breaches can lead to significant damage, potentially causing complete or partial network infrastructure destruction, halting organizational operations, and resulting in substantial financial losses, even leading to bankruptcy in severe cases.

Wireless networks, particularly WLANs, have gained immense popularity due to their convenience compared to traditional wired technologies. However, this convenience also makes them prime targets for attackers, making security concerns paramount in wireless network environments. To address these concerns, authentication protocols have been developed to prevent unauthorized access to wireless networks, with wired equivalent privacy (WEP) and Wi-Fi protected access (WPA) being the most common encryption technologies employed.

Despite efforts to secure wireless networks, vulnerabilities persist, particularly in the widely used WPA2 protocol. These vulnerabilities, exploited through key recovery attacks (KRACKs), pose serious threats by allowing attackers to intercept supposedly secure encrypted information. This compromised data may include sensitive information such as credit card numbers, passwords, chat messages, emails, and photos. To counteract this threat, robust security measures, including prevention, detection, and response plans, must be implemented.

One effective strategy to mitigate security risks in network infrastructure is through network penetration testing. This essential process mimics the tactics of attackers, identifying and closing vulnerabilities to strengthen network defenses. Studies such as those by Jain, S. et al. [14], Agrawal, A. et al. [15], and Hoque, N. et al. [16] have delved into various aspects of wireless network security and penetration testing methodologies. These studies have utilized a range of tools and technologies to perform attacks and detect anomalies, contributing to the ongoing efforts to enhance network security and mitigate threats such as KRACK attacks.

2.2. Architecture

To initiate WLAN penetration testing, it's essential to establish the testing environment, comprising both hardware and software components [17]. Hardware elements encompass routers, attacker devices like laptops equipped with WLAN cards, and authorized clients such as mobile devices. Software tools such as "airodump-ng" and "aircrack-ng" are utilized for eavesdropping, sniffing, and capturing WLAN traffic. Furthermore, utilities like "mac-changer" and "aireplay-ng" are employed to manipulate the AP and allowed clients. The attacker's laptop, running Kali Linux as the penetration testing OS, is equipped with all necessary software components. The client device, typically a mobile device, is configured with specific operating system settings and connects to the internet through an AP [17].

2.3. Penetration Testing Methodology in WLAN

Various standard methodologies exist for conducting different types of network penetration tests. In the case of WLAN penetration testing, researchers outline the approach to conducting penetration tests within a WLAN environment.

2.3.1. Reconnaissance/Gathering Information

In this phase, testers gather network data and connections, seek information about the attack target, or discreetly establish a presence in a specified area. Additionally, they evaluate the existing security measures in the target system [18]. Moreover, obtaining details about DHCP, DNS, and subnet IP addresses is essential.

2.3.2. Network Scanning

During this phase, security vulnerabilities within remote target networks or local hosts are pinpointed. To achieve this, IP address data is gathered from active hosts and Layer 2 devices. Subsequently, target hosts undergo port scanning using tools like Nmap and Nessus. This process results in the creation of tables containing hosts' IP addresses, their corresponding MAC addresses, and the identification of open ports.

2.3.3. Exploitation

This method is employed to inject diverse forms of attacks into the network, including techniques to breach WLANs. Tests are conducted utilizing tools such as cracking attack tests, DoS attacks, and router password attacks.

2.3.4. Post Exploitation

During this phase, consultations are conducted to offer guidance on defending the target network. The methods outlined in this step aim to assist testers in identifying and documenting sensitive information, configuration settings, communication channels, and relationships with other network devices that could potentially be exploited to gain further access to the network.

2.4. Network Penetration Testing Methodologies

The study by Astrida, M. et al. [11] aimed to assess network vulnerabilities in the wireless local area network (WLAN) at SMP XYZ. The authors employed the penetration testing execution standard (PTES) method to analyze attacks on the XYZ SMP network. Four types of tests were conducted. In the WPA2 cracking test, it was discovered that the WPA2 key could be cracked. Additionally, the DoS test revealed that breaking the client connection to the access point was relatively simple, requiring only the MAC address and SSID of the access point. The password router wireless cracking test indicated a high level of vulnerability due to the use of default passwords by the access point. Finally, an isolation test for the access point demonstrated that clients could be attacked. To address these gaps, the authors proposed solutions including the use of unique and robust WPA2 keys with a minimum of 15 characters, sector antennas for wireless network antennas, strong passwords of at least 15 characters, and configuring AP isolation at the access point.

Alsahlany, A. et al. [17] conducted WLAN penetration tests to evaluate the security of hidden SSIDs, MAC filtering, and WPA2. They found that hidden SSIDs could be easily discovered, MAC filtering was not a significant obstacle for attackers, and WPA2 was vulnerable to brute force and human social factor attacks. Recommendations included disabling the WPS protocol to prevent exploitation of vulnerabilities and using more complex WPA2 passphrases.

Fikriyadi et al. [27] conducted WLAN penetration tests to assess WLAN security using a phased approach including planning, detection, attack, and reporting. The tests revealed vulnerabilities in network resources that attackers could exploit. Kali Linux with the Wireshark application was used for encryption cracking, MAC address bypass, infrastructure attacks, and MITM attacks. The results indicated vulnerabilities in WLAN connections, authentication failures through captive portals, and successful MAC address bypass attacks.

Wahyudi, E. et al. [28] compared two RADIUS server security systems with a captive portal using OpenWRT to provide a secure alternative to high-performance WLANs and WPA2-PSK. The captive portal system was found to be 80 percent more secure than WPA2-PSK.

Syed, S. et al. [18] aimed to assess the security level of Mehran University of Engineering and Technology’s (MUET) campus area network, IP cameras, biometric systems, and switches. A live network penetration test was conducted, with proposed solutions including changing default credentials and preventing unauthorized remote access.

Kumar, R. et al. [29] performed penetration testing in a network lab, demonstrating attacks and penetration of network infrastructure using Kali Linux. The methodology included information gathering, vulnerability analysis, exploitation, and reporting phases, utilizing tools such as Dmitry, Nmap, Nexpose Community, Nessus, Armitage, and Metasploit framework.

3. RESULTS, DISCUSSIONS AND CONCLUSIONS

In this section, the results of the analysis of the previous studies are presented.

3.1 Wireless Local Area Network Penetration Testing

Based on the studies reviewed, vulnerabilities within WLANs were primarily associated with flaws in the IEEE 802.11 encryption protocols, specifically WEP2 and WEP3, making them susceptible to attacks such as KRACK and Downgrade attacks.

3.2 Tools for Detecting Open Ports

The studies analyzed identified several common scanning tools utilized for detecting vulnerable ports, including Nmap, Metasploit, Wireshark, Shadow, Nessus, AirCrack, Burp Suite, BeEF, and SPARTA. Among these, Nmap emerged as the most frequently recommended tool.

3.3 Open Ports

Numerous ports with known vulnerabilities were highlighted in previous studies, posing exploitable risks during the scanning phase of penetration testing. Commonly mentioned vulnerable ports include Transmission Control Protocol (TCP) and File Transfer Protocol (FTP).

3.4 Network Penetration Testing Methodologies

Our findings indicate the existence of various methodologies for network penetration testing, all revolving around a core process: gathering information about the target network, scanning for vulnerabilities, executing attacks, and providing remediation recommendations in detailed reports.

3.5 Types of Attacks Exploiting Open Ports

Previous studies underscored the multitude of attack types targeting vulnerable ports, posing significant security risks. These include Denial of Service (DoS) attacks, brute force attacks, Man-in-the-Middle (MITM) attacks, exploitation of Open SSL library random number generation, packet sniffing, viruses, worms, Trojans, among others. The weak network topology exacerbates vulnerability to a wide array of attack vectors.

3.6 Mitigation Techniques for Protecting Open Ports against Vulnerabilities

The studies identified several common mitigation techniques for detecting and safeguarding open ports, including machine learning algorithms, Vulnerability Assessment and Penetration Testing (VAPT), Dynamic Open Port Analysis (DOPA), Nmap detection rules, Fixing Network Vulnerability Tool (FNSV), and development of penetration testing software equipped with comprehensive toolboxes**.**

IV. SUGGESTED SOLUTIONS

We reside in an era characterized by the rapid advancement of technologies reliant on information systems for critical operations, management, and information sharing. With the looming risk of cyber threats, researchers are increasingly prioritizing the security of these technologies through penetration testing. Our paper offers an overview of network penetration testing techniques and identifies the following future directions:

Firstly, we advocate for further exploration into leveraging machine learning, particularly deep reinforcement learning, to enhance network penetration testing tailored to specific network topologies, such as WLAN networks.

Secondly, despite extensive research on network penetration testing, certain types of attacks, particularly real-world threats like KRACK attacks, remain insufficiently addressed and simulated in testing scenarios.

Thirdly, the imperative in network penetration testing lies in detecting vulnerabilities before exploitation, with minimal false positives. Striking this balance is crucial for effective security measures.

Researchers [6] have highlighted the complexities of manual network penetration testing, citing challenges such as the scarcity of proficient penetration testers, time-intensive processes, and high costs. To address these limitations, there is a pressing need to develop automated tools that amalgamate the expertise of seasoned penetration testers. Such tools, based on machine learning, could offer non-experts comprehensive insights into the security posture of their systems. Therefore, further research is warranted to explore the deployment of automated penetration testing, particularly leveraging deep reinforcement learning, to overcome these challenges effectively.

REFERENCES

1. Adamovic, S. Penetration testing and vulnerability assessment: Introduction, phases, tools and methods. In *Sinteza 2019-International Scientific Conference on Information Technology and Data Related Research*; Singidunum University: Belgrade, Serbia, 2019; pp. 229–234. [**Google Scholar**]

2. Tidy, J. Swedish Coop Supermarkets Shut Due to Us Ransomware Cyber-Attack. *BBC News*. 3 July 2021. Available online: **https://www.bbc.com/news/technology-57707530** (accessed on 17 May 2023).

3. Shah, M.; Ahmed, S.; Saeed, K.; Junaid, M.; Khan, H. Penetration testing active reconnaissance phase–optimized port scanning with nmap tool. In Proceedings of the IEEE 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, Pakistan, 30–31 January 2019; pp. 1–6. [**Google Scholar**]

4. Jayasuryapal, G.; Meher Pranay, P.; Kaur, H. A Survey on Network Penetration Testing. In Proceedings of the IEEE 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM), London, UK, 28–30 April 2021. [**Google Scholar**]

5. Packetlabs. Black-Box vs. Grey-Box vs. White-Box Penetration Testing. 19 April 2022. Available online: **https://www.packetlabs.net/posts/types-of-penetration-testing/** (accessed on 6 May 2023).

6. Khera, Y.; Kumar, D.; Garg, N. Analysis and Impact of Vulnerability Assessment and Penetration Testing. In Proceedings of the IEEE 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 14–16 February 2019. [**Google Scholar**]

7. Press, T.A. T-Mobile Says Breach Exposed Personal Data of 37 Million Customers. *NPR*. 20 January 2023. Available online: **https://www.npr.org/2023/01/20/1150215382/t-mobile-data-37-million-customers-stolen** (accessed on 12 May 2023).

8. Farah, A.-D.; Alshammari, E. Automated penetration testing: An overview. In Proceedings of the 4th International Conference on Natural Language Computing, Copenhagen, Denmark, 31 October–4 November 2018. [**Google Scholar**]

9. Kumar, B.K.; Raj, N.; Dhivvya, J.P.; Muralidharan, D. Fixing Network Security Vulnerabilities in Local Area Network. In Proceedings of the 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 23–25 April 2019; pp. 1349–1354. [**Google Scholar**]

10. Shebli, A.; Mohammed Zaher, H.; Beheshti, B.D. A study on penetration testing process and tools. In Proceedings of the 2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT), Farmingdale, NY, USA, 4–8 May 2018. [**Google Scholar**]

11. Astrida, D.N.; Saputra, A.R.; Assaufi, A.I. Analysis and Evaluation of Wireless Network Security with the Penetration Testing Execution Standard (PTES). *Sink. J. Dan Penelit. Tek. Inform.* **2022**, *7*, 147–154. [**Google Scholar**] [**CrossRef**]

12. Singh, N.; Meherhomji, V.; Chandavarkar, B.R. Automated versus manual approach of web application penetration testing’. In Proceedings of the IEEE 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 1–3 July 2020; pp. 1–6. [**Google Scholar**]

13. Singh; Rajawat, G.; Sharma, J. Wireless Cyberspace. *J. Anal. Comput. (JAC).* **2022**, *16*, 1–4. [**Google Scholar**]

14. Jain, S.; Pruthi, S.; Yadav, V. Ethical Hacking of IEEE 802.11 Encryption Protocols. *J. Xi’an Shiyou Univ. Nat. Sci. Ed.* **2009**, *18*, 108–112. [**Google Scholar**]

15. Agrawal, A.; Chatterjee, U.; Maiti, R.R. CheckShake: Passively detecting anomaly in Wi-Fi security handshake using gradient boosting based ensemble learning. *IEEE Trans. Dependable Secur. Comput.* **2023**, 1–13. [**Google Scholar**] [**CrossRef**]

16. Hoque, N.; Rahbari, H.; Rezendes, C. Systematically Analyzing Vulnerabilities in the Connection Establishment Phase of Wi-Fi Systems. In Proceedings of the 2022 IEEE Conference on Communications and Network Security (CNS), Austin, TX, USA, 3–5 October 2022; pp. 64–72. [**Google Scholar**]

17. Alsahlany, A.M.; Alfatlawy, Z.H.; Almusawy, A.R. Experimental Evaluation of Different Penetration Security Levels in Wireless Local Area Network. *J. Commun.* **2018**, *13*, 723–729. [**Google Scholar**] [**CrossRef**]

18. Syed, S.; Khuhawar, F.; Arain, K.; Kaimkhani, T.; Syed, Z.; Sheikh, H.; Khan, S. *Case Study: Intranet Penetration Testing of MUET*; Mehran University of Engineering and Technology: Jamshoro, Pakistan, 2020; pp. 17–19. [**Google Scholar**]

19. Cadiente, K.A.; Castro, R.A.; Gica, E.V.; Mora, K.M.; Ternio, J.V. Applying vulnerability assessment and penetration testing (vapt) and network enhancement on the network. *Infrastruct. Journey Tech Inc. Innov.* **2020**, *3*, 1. [**Google Scholar**]

20. Shi, P.; Qin, F.; Cheng, R.; Zhu, K. The penetration testing framework for large-scale network based on network fingerprint. In Proceedings of the IEEE 2019 International Conference on Communications, Information System and Computer Engineering (CISCE), Haikou, China, 5–7 July 2019. [**Google Scholar**]

21. Patel, A.M.; Patel, H.R. Analytical study of penetration testing for wireless infrastructure security. In Proceedings of the IEEE 2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET), Chennai, India, 21–23 March 2019. [**Google Scholar**]

22. Iyamuremye, B.; Hisato, S. Network security testing tools for SMEs (small and medium enterprises). In Proceedings of the IEEE 2018 International Conference on Applied System Invention (ICASI), Tokyo, Japan, 13–17 April 2018. [**Google Scholar**]

23. Overstreet, D.; Wimmer, H.; Haddad, R.J. Penetration Testing of the Amazon Echo Digital Voice Assistant Using a Denial-ofService Attack. In Proceedings of the IEEE 2019 SoutheastCon, Huntsville, AL, USA, 11–14 April 2019. [**Google Scholar**]

24. U Nisa, M.; Kashif, K. Detection of slow port scanning attacks. In Proceedings of the IEEE 2020 International Conference on Cyber Warfare and Security (ICCWS), Norfolk, VA, USA, 12–13 March 2020. [**Google Scholar**]

25. Bagyalakshmi, G.; Rajkumar, G.; Arunkumar, N.; Easwaran, M.; Narasimhan, K.; Elamaran, V.; Solarte, M.; Hernández, I.; Ramirez-Gonzalez, G. Network vulnerability analysis on brain signal/image databases using Nmap and Wireshark tools. *IEEE Access* **2018**, *6*, 57144–57151. [**Google Scholar**] [**CrossRef**]

26. Muin, Y. MikroTik Router Vulnerability Testing for Network Vulnerability Evaluation using Penetration Testing Method. *Int. J. Comput. Appl.* **2022**, *975*, 8887. [**Google Scholar**]

27. Fikriyadi, F.; Ritzkal, R.; Prakosa, B.A. Security Analysis of Wireless Local Area Network (WLAN) Network with the Penetration Testing Method. *J. Mantik* **2020**, *4*, 1658–1662. [**Google Scholar**]

28. Wahyudi, E.; Luthfi, E.T.; Efendi, M.M.; Mataram, S.T. Wireless penetration testing method to analyze WPA2-PSK system security and captive portal. *J. Explor. Stmik Mataram* **2019**, *9*, 1. [**Google Scholar**] [**CrossRef**]

29. Kumar, R.; Katlego, T. Internal network penetration testing using free/open source tools: Network and system administration approach. In Proceedings of the International Conference on Advanced Informatics for Computing Research, Shimla, India, 14–15 July 2018; Springer: Singapore, 2018. [**Google Scholar**]

30. Pandey, R.; Vutukuru, J.; Chopra, U.K. Vulnerability assessment and penetration testing: A portable solution Implementation. In Proceedings of the IEEE 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), Bhimtal, India, 25–26 September 2020. [**Google Scholar**]

31. Liao, S.; Zhou, C.; Zhao, Y.; Zhang, Z.; Zhang, C.; Gao, Y.; Zhong, G. A Comprehensive detection approach of Nmap: Principles, rules and experiments. In Proceedings of the IEEE 2020 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Chongqing, China, 29–30 October 2020. [**Google Scholar**]

32. Ernawati, T.; Fachrozi, M.F.; Syaputri, D.D. Analysis of Intrusion Detection System Performance for the Port Scan Attack Detector, Portsentry, and Suricata. In *IOP Conference Series: Materials Science and Engineering*; IOP Publishing: Bristol, UK, 2019; Volume 662. [**Google Scholar**]

33. Hartpence, B.; Kwasinski, A. Combating TCP port scan attacks using sequential neural networks. In Proceedings of the IEEE 2020 International Conference on Computing, Networking and Communications (ICNC), Big Island, HI, USA, 17–20 February 2020. [**Google Scholar**]

34. Gupta, A.; Sharma, L.S. Mitigation of dos and port scan attacks using snort. *Int. J. Comput. Sci. Eng.* **2019**, *7*, 248–258. [**Google Scholar**] [**CrossRef**]

35. Neu, C.V.; Tatsch, C.G.; Lunardi, R.C.; Michelin, R.A.; Orozco, A.M.; Zorzo, A.F. Lightweight IPS for port scan in OpenFlow SDN networks. In Proceedings of the IEEE NOMS 2018—2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 23–27 April 2018. [**Google Scholar**]

36. Wu, D.; Gao, D.; Chang, R.K.; He, E.; Cheng, E.K.; Deng, R.H. Understanding open ports in Android applications: Discovery, diagnosis, and security assessment. In Proceedings of the Network and Distributed System Security Symposium 26th NDSS 2019, San Diego, CA, USA, 24–27 February 2019; p. 1. [**Google Scholar**]

37. Luswata, J.; Zavarsky, P.; Swar, B.; Zvabva, D. Analysis of scada security using penetration testing: A case study on modbus tcp protocol. In Proceedings of the IEEE 2018 29th Biennial Symposium on Communications (BSC), Toronto, ON, Canada, 6–7 June 2018. [**Google Scholar**]

38. Shah, N.; Shravan, S. Server Stress Test Using DDoS Attack. *Int. J. Res. Eng. Sci.* **2021**, *9*, 53–58. [**Google Scholar**]

39. Chaudhary, S.; O'Brien, A.; Xu, S. Automated post-breach penetration testing through reinforcement learning. In Proceedings of the 2020 IEEE Conference on Communications and Network Security (CNS), Avignon, France, 29 June–1 July 2020. [**Google Scholar**]

40. Hu, Z.; Beuran, R.; Tan, Y. Automated penetration testing using deep reinforcement learning. In Proceedings of the 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Genoa, Italy, 7–11 September 2020. [**Google Scholar**]

41. Niculae, S.; Dichiu, D.; Yang, K.; Bäck, T. *Automating Penetration Testing Using Reinforcement Learning*; Experimental Research Unit Bitdefender: Bucharest, Romania, 2020. [**Google Scholar**]

42. Ghanem, M.C.; Chen, T.M.; Nepomuceno, E.G. Hierarchical reinforcement learning for efficient and effective automated penetration testing of large networks. *J. Intell. Inf. Syst.* **2022**, *60*, 281–303. [**Google Scholar**] [**CrossRef**]

43. Erdődi, L.; Sommervoll, Å.Å; Zennaro, F.M. Simulating SQL injection vulnerability exploitation using Q-learning reinforcement learning agents. *J. Inf. Secur. Appl.* **2021**, *61*, 102903. [**Google Scholar**] [**CrossRef**]

44. Motghare, V.; Kasturi, A.; Kokare, A.; Sankhe, A. Securezy—A Penetration Testing Toolbox. *Int. Res. J. Eng. Technol.* **2022**, *9*, 2375–2378. [**Google Scholar**]