Journal of Information Systems Engineering and Management
20--, Vol(Iss)
e-ISSN: 2468-4376
https://jisem-journal.com/

Research Article

Strategic Network Fingerprinting: Advanced Network Analysis and Penetration Testing for Large-Scale Networks

Shaik Rasheed Nehal, Mudavath Sindhu
Undergraduate, Department of Computer Science - Cybersecurity, Institute of Aeronautical Engineering, Hyderabad, India
Undergraduate, Department of Computer Science - Cybersecurity, Institute of Aeronautical Engineering, Hyderabad, India

ARTICLE INFO

Received: dd Month 20--
Accepted: dd Month 20--

ABSTRACT

Penetration testing provides proactive security to the enterprise by simulating hacker attacks and detecting system weaknesses legally. Despite their efficiency, manual procedures require much time and resources, especially when dealing with large networks that have a large number of attack surfaces. The traditional approach carries a lot of risk to business continuity because it interferes with network operations. This solution breaks the constraints by including NSE (Nmap Scripting Engine) plug-ins and internet search engines in the network framework. Cyberspace search engines are applied to map the Internet to make it easier to examine infrastructure and devices efficiently. Utilizing NSE plug-ins increases the capabilities of Nmap, reduces scanning times, and increases the efficiency of the framework

Keywords: Penetration Testing, Cyberspace Search Engines, NSE (Nmap Scripting Engines)

INTRODUCTION

The growing dependence of contemporary society on information networks for both personal and professional purposes reveals how crucial it is that network security has to be. It clearly shows that security measures cannot be treated as optional as databases like China National Vulnerability Database (CNVD) have been showing constant growth in flaws reported over time.

Penetration testing is crucial because it simulates the attacks of hackers to determine vulnerabilities in systems before hackers can exploit them. The innovative framework for extensive network penetration testing elevates this procedure by using advanced network fingerprinting technology. It improves security assessments and increases their effectiveness and impact in protecting vital systems through easier target identification and data analysis.

It's revolutionizing things for penetration testers / engineers doing the testing while improving the speed of gathering and analyzing the target collected data; hence, newly developed security vulnerabilities are detected together with solutions. The benefits include being technical, as it gives a reason behind engaging all-rounded data security measures, reminding one why professionals indulged in cybersecurity training still need to acquire new pieces of information and increase one's level of knowledge in this field. This way it empowers practitioners to always lead in changing threats while there is a culture of education, cooperation, and sharing information among cybersecurity experts. Such a comprehensive strategy is designed to ensure that there is an assurance of updated weapons and tactics for securing a network as the system defense gets improved.

This advanced framework is a highly targeted collector framework and mainly emphasizes data processing. The process could vastly improve new security flaws' detection. Furthermore, because of improvement that takes place concerning both aspects of data as well as network, it comes out to be the most valued instrument for conducting penetration testing by any research or engineer. The following is how much importance there is seen in continuous learning and becoming more evolved within professionals regarding cybersecurity. It is constantly changing, so the

need to be abreast of the latest technologies and methods is imperative. It keeps the practitioners updated with the ever- changing landscape by providing an opportunity for skill building and a culture of teamwork.

This proposed penetration testing methodology will remain at par with the potential threats as it automates vulnerability detection and uses machine learning along with advanced analytics for forecasting risk. The system quickly reacts to new threats due to an analysis of the user's behavior and network data. Its innovative procedure will not only enhance an organization's effectiveness in the prevention of future threats but will also set a whole new benchmark for cybersecurity innovations. This gives a practical way of protecting essential information systems from threats and therefore keeps them safe within an ever-changing threat environment.

At this hour, with the complexity of cyberattacks on the rise daily, this framework is integral to the protection of not just the individual businesses but also the gigantic digital infrastructure. Such frameworks need to be in place in order to maintain strong defense and security of international information networks. This framework incorporates advanced techniques, such as network fingerprinting, to create a modern digital environment that will actually promote and enhance the efficacies of security assessments whilst encouraging continued learning and collaboration.

Almost every aspect of our life is affected by technology, especially the internet, which allows us to engage in a wide range of online activities. It exposes us to serious hazards, such as cyberattacks, even though it also presents amazing prospects and rewards. The fierce rivalry between for-profit and nonprofit groups that depend on networks to provide their services is the source of these concerns. Networks rely on open ports to work and the same ports are at the same time a conduit for hostile actors. In this regard, stringent measures of cybersecurity become cardinal in safeguarding our systems that enable our way of modern living and are hence not a choice but mandatory.

This has led to massive disruption both for the individual and business, with disruptions ranging from network failure for personal networks to the total halt of business operations. For instance, in Sweden in 2021, a ransomware attack led to the temporary closure of nearly 500 Coop supermarkets, an example of how such cyber threats affect companies around the world.

Against such threats, companies employ a term known as penetration testing that can be described as ethical hacking or white-hat attacks. Penetration testing is the approach towards proactive identification of computer network, application, or system security weaknesses before bad guys exploit them. The weakness then gets identified and companies can improve with specific steps on their security measures to minimize chances for catastrophic breaches.

Penetration testing is an active approach toward the identification of weaknesses in digital assets by checking the assets from the point of view of an attacker, trying to exploit the weaknesses. As the laws of the land, like GDPR, are becoming important to the people, there has been a need to have cybersecurity goals such as availability, confidentiality, and integrity of data. Tools like Nmap under the set of security tools Kali Linux allow all these organizations to play the role of penetration tester while developing their capability in defense.

Penetration tests are the essentials in discovering system vulnerabilities and in finding out how they will be addressed. In a real attack simulation on a targeted system, testers systematically, and in a safe way, discover weaknesses. Reports summarize the findings, highlighting flaws that need to be repaired and suggesting security patches that should be applied by the management. This is how a risk assessment tool gets to work and proves one's network security. It is a matter of utmost concern for the organizations, though it is expensive and very time-consuming. There, hence is an urgent necessity of specially tailored penetration testing methodologies for the safeguarding systems and devices with resultant safe information and network security.

1.1 Network Penetration Testing

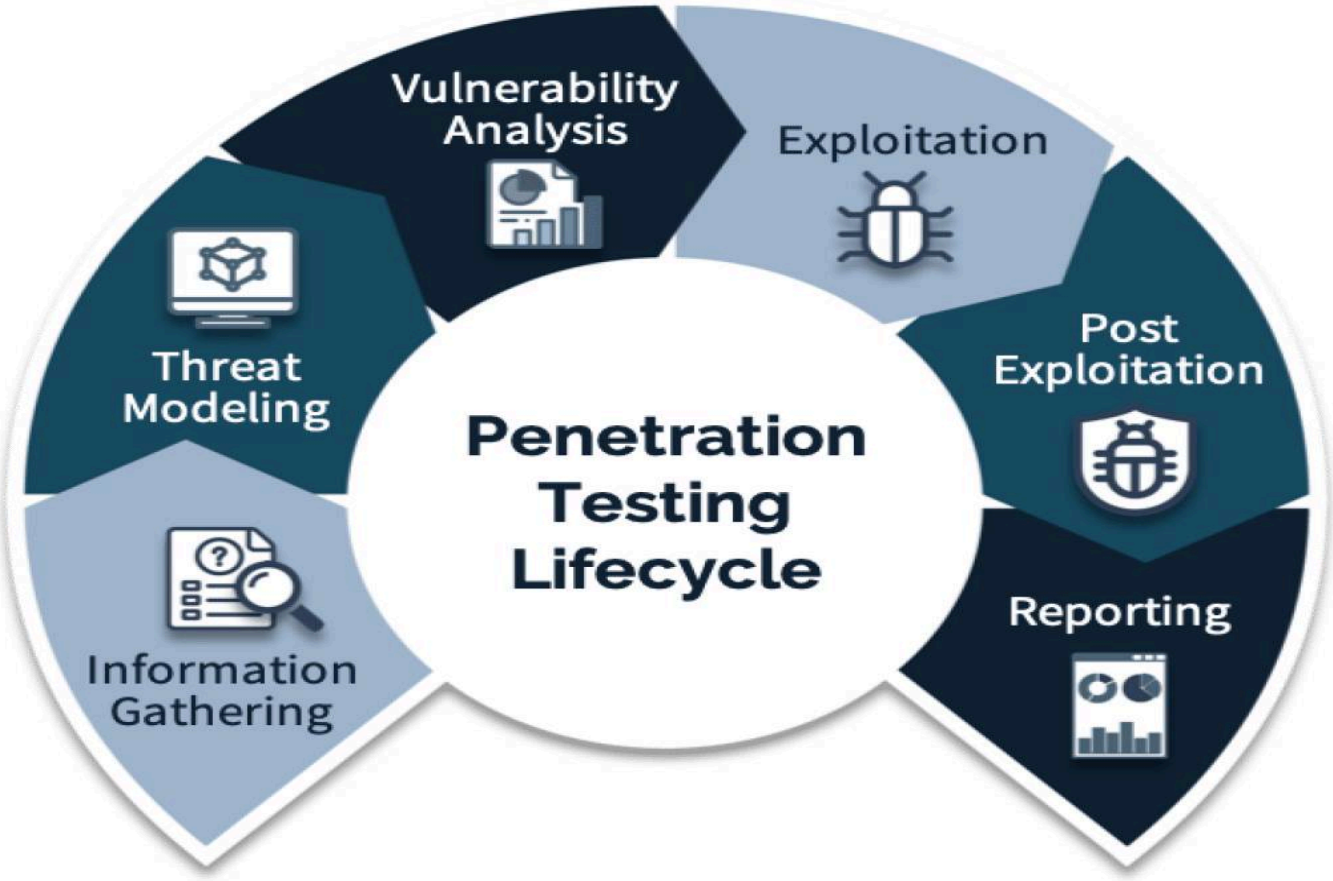
Network penetration testing simulates malicious insider or outsider attacks. In this way, through the proactive approach, it is possible to find out possible flaws in systems, apps, or network infrastructure, so that vulnerability of access by an attacker to reach unauthorized, interfere services, or steal confidential information may be located.

Reconnaissance/Gathering: Information It involves gathering information concerning a target network. That's sometimes referred to as making web searches on publicly available information or using certain kinds of tools that reveal details of devices and services within reach.

Network scanning: ZMap or Masscan is a rapid port scanner used to scan a broad IP range to find possibly active devices together with open ports on the network under attack.

Exploitation: It will try to exploit the previously identified vulnerabilities by the scanning phase. In this attempt, automation scripting languages such as Python or Ruby are used.

Post Exploitation: It is the phase where the exploited vulnerability ensures continued access to the targeted system and moves into the network to get more.



1.2 Advanced Penetration Testing Techniques for Large-Scale Networks

Once the target information is picked up through these advanced search engines, automated testing scripts execute penetration tests on each target successively. These scripts contain the latest security advisories and exploitation databases like CVE and Exploit-DB for detecting potential vulnerabilities.

Shodan and its counterparts provide all information that may be utilized for fine-tuning penetration tests with particular vulnerabilities. For instance, by applying filters provided by the search in Shodan, one can find those devices which utilize old software versions or expose ports and further target them with custom scripts. Those scripts have been optimized to exploit known vulnerabilities that would make the testing process more efficient with a timely finish.

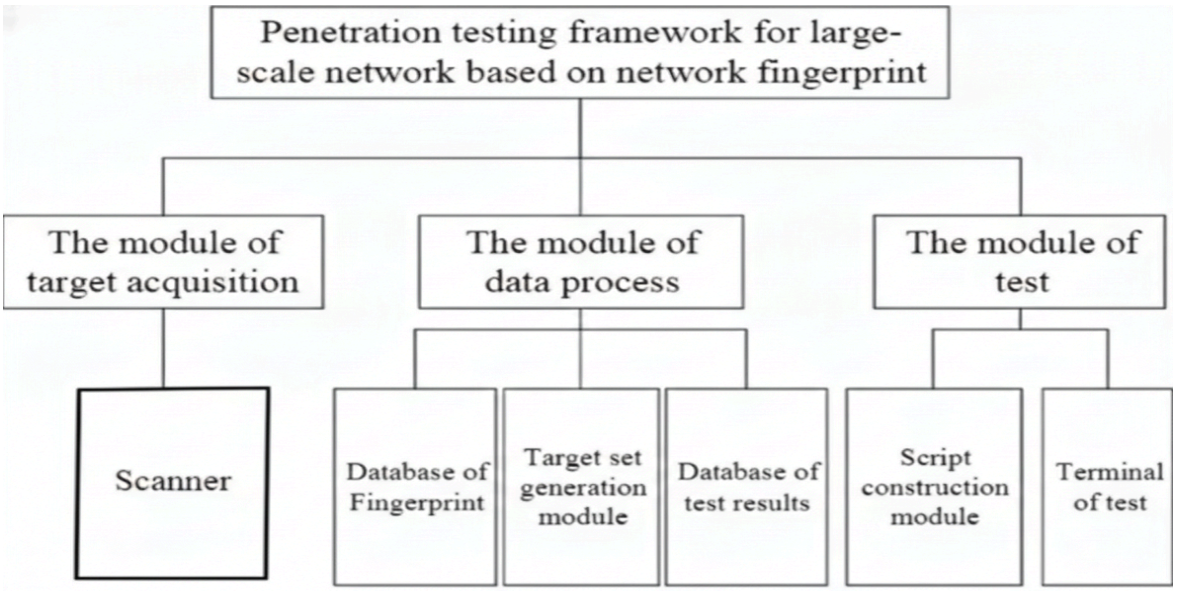
Besides that, testers typically utilize some scripts along with 0- day exploits and newly discovered vulnerabilities. In order to be able to encompass broad coverage on possible security threats while performing penetration testing, it was intended to be used in conjunction with the information coming from a search engine and script engines. These advanced techniques include resource-efficient usage of computing, and thus mass network testing is done in a fraction of the time.

I. EXISTING SYSTEM

The current techniques in large-scale network penetration testing are either more or less based on the traditional approach that includes high-speed port scanning and automated test scripts. Though very popular, they are associated with several limitations.

High-speed port: scanning tools like Z-Map and Masscan do their job pretty well on rapidly discovering open ports on the target hosts. However, in furnishing detailed information on target systems beyond a status on the port's status, these tools are inadequate. This deficiency may bring along some errors, thus providing false positives and inefficiency while picking targets for deeper research and testing.

Automated test scripts: highly used in penetration testing of identified targets. Although automated, they fail to keep up with the evolving threats and sometimes miss complex vulnerabilities that have to be inspected manually. Their focus on certain types of vulnerabilities limits them from giving a comprehensive security assessment of different attack vectors. The rigidity of the automated test scripts limits the flexibility of dynamically adjusting to the evolving cybersecurity.



Penetration Testing Framework

II. PROBLEM STATEMENT

1. Traditional penetration testing is usually accompanied by a myriad of deficiencies, such as false positives and inaccuracies, especially in large networks.

2. Lack of improvement of efficiency through the services of traditional intrusion testing because the architecture network surface is so huge.

3. No accurate identification of threats, more false positives as in traditional pentesting of large networks, as in the old days no single corner of the network was covered because it lacked the use of search engines in cyber-space, which has accuracy to exploit or yet to be exploited.

III. PROPOSED SYSTEM

The proposed system for strategic Networking Fingerprinting: Advanced network analysis for large-scale penetration testing encompasses a whole approach of both technical and organizational measures. Some of the strategies include:

A. Cyberspace search engines: Cyberspace search engines include Zoomeye, Censys, Shodan, among others. As such, they facilitate strong searching and sophisticated filtering; thus, they allow searching a particular kind of Internet-associated devices, services, as well as vulnerabilities.

B. Nmap Scripting Engine (NSE): These NSE scripts written from Nmap network scanner is one of the powerful add-on features that enables end users to automate a variety of tasks associated with network investigations and security auditing. By creating unique behaviors and added abilities for Nmap to further its capability, this collection of Lua programming languages should be used.

OBJECTIVES

Improved Testing Accuracy and Hit Rate and Reduced False Positives Traditional methods might flag irrelevant systems due to open ports. Network fingerprinting helps eliminate such false positives by identifying the specific services running on those ports. Enhanced Target Acquisition and Efficiency and Scalable Testing Capabilities Design the framework to handle large-scale network testing while maintaining efficient script execution and result collection. Integration with Penetration Testing Frameworks Utilize existing frameworks like Metasploit to streamline script development for identified vulnerabilities. This leverages established functionalities for efficient test execution.

LITERATURE REVIEW

[1] Yan Shujun et al. provide one of the most efficient web- fingerprinting identification techniques since in their paper they quote that "An Effective Web Fingerprint Identification Method". Over years of research and real life applications, the authors deliver innovations and tools to be helpful for professionals engaged with a cybersecurity field while providing safety levels for web application and web networks. This book is invaluable for performing professionals and fresh talent looking at deepening web security understanding and making defensive attacks even more formidable against cyber threats.

[2] Huang Z, Xia C, Sun B, et al. Analyzing and Summarizing the Web Server Detection Technology Based on HTTP, an All-inclusive book exploring web server detection techniques based on HTTP. It has value and practical methodology which enhances understanding and competency, so it was an invaluable and indispensable source of information that cybersecurity professionals and others had researched in academic circles.

[3] The book "BlindElephant: Web Application Fingerprinter & Vulnerability Inferencing" had already been published by Thomas P. for the cybersecurity practitioners as well as the professionals. It describes a completely

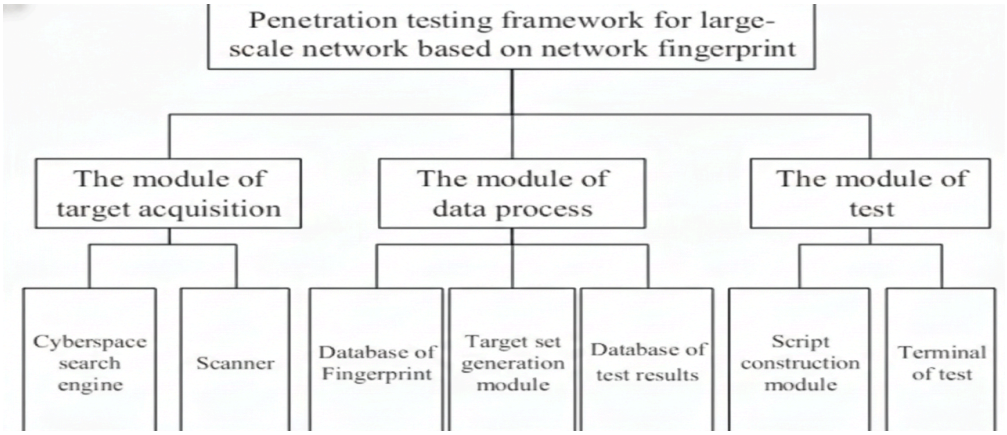
new fingerprinting technique for a web application, inferred vulnerabilities with the help of hands-on experience through the concrete knowledge of discovery & prevention of a web application's vulnerability using BlindElephant.

[4] Lee D and Rowe J have coauthored one of the most seminal works in the field of cyber security: "Detecting and Defending Against Web- server Fingerprinting". With knowledge in both network security and web technologies, they provide comprehensive information on how to detect and defend against web-server fingerprinting attacks. It offers readers a powerful defense along with handy tools to keep web servers safe from continuously evolving cyber threats. A very valuable book for cyber security professionals, researchers, and students.

METHODOLOGY

A. Advanced Network Fingerprinting: Develop advanced techniques of network fingerprinting that could correctly identify and classify services running on open ports across the network by avoiding false positives through deep packet inspection, protocol analysis, and behavior-based identification while ensuring proper target selection, integrating seamlessly into already established penetration testing frameworks, such as Metasploit, in order to smooth out the identification and exploitation processes.

B. Scalable Architectural Design: It designs distribution architecture which supports concurrent testing across nodes so it efficiently manages large networks, dealing with their scalability concerns. Some of the optimal techniques, like load distribution and task allocation strategies, are used for efficient resource usage without compromising testing time. It will ensure that the framework adapts well to a great network environment without sacrificing precision or performance.



Penetration Testing Framework based on network fingerprint

C. Automated Testing with NSE and Existing Frameworks: architecture that supports concurrent testing across nodes so it can efficiently handle large networks and deal with their scalability issues. Optimal techniques like load distribution and task allocation strategies are used for the efficient usage of resources without compromising the testing time. This will ensure that the framework adapts well to a great network environment without losing precision or performance. Metasploit is the best place for exploit attempts development and running. It is a fully integrated set of tools and functionalities for the development and running of exploit attempts. The integration provides efficient testing through automation since it will make exploits based on the outcome of vulnerability scans conducted by NSE plug-ins. Further, vulnerabilities found by NSE scripts can be exploited and verified through Metasploit to have a proper view of the security posture of the network.

D. Cyberspace Search Engine and Network Fingerprinting: Using the cyberspace search engines, like Shodan and Censys, one can find the target devices and potential vulnerabilities in the network. Network fingerprinting techniques analyze the open ports and services running on those ports, eliminating some of the false positives from the traditional scanning methods.

- This particular search engine specializes in the work of indexing and cataloging every device connected to the internet, which provides valuable insights into the world of digital. By interrogating such search platforms, cybersecurity specialists will obtain information about the devices, services, and configurations within an organizational perimeter. It enables proactive reconnaissance that opens up a route for organizations to explore their digital footprint and vulnerability zones.
- Network fingerprinting is the deep packet inspection, protocol analysis, and behavior-based identification meant to build comprehensive profiles of what's inside a network from devices to services. It allows deep packet inspection: the inspection of packet content.

IMPLEMENTATION

A penetration testing framework implemented in large networks is aimed at ensuring the framework implemented is efficient, robust, and able to handle extensive networks. The process can thus be divided into several significant steps, all of which contribute to the overall function and effectiveness of the framework.

1. Import Required Modules: All the modules and libraries needed to build the penetration testing framework will be imported. Flask will be used for the web application and also to render_template and request for rendering an HTML template and handling HTTP requests respectively. The re module is used for regular expressions used in validation and processing the input data. Shodan and requests are fairly primitive while working with third party APIs. This also includes public cyberspace search engines to query information. Base64 module is included in this project for encoding and decoding of data, possibly needed when handling sensitive information securely**.**

2. Create Project Structure: Organizing your project into a clear, well-structured directory structure is fundamental to making it scalable and maintainable. It would have templates, the folder where it would be storing all its HTML files; static includes - CSS, JavaScript, images; and main application folders which were designated for its logic and routes. This way, separation between each aspect of a project is more accessible for further adjustments on the different phases of developing the framework. A more structured methodology supports collaboration as well and easy debugging.

3. Initialize the Flask Application: Once all the preparatory work is done, an instance of the Flask application is initialized. In doing this, it includes the primary framework with initial configurations of the app through routes handling the requests of the user together with imported modules, intercommunication between the web interface and back-end logic. The Flask app is the heart of the framework, through which users will be able to access all its features in a user- friendly manner and enable smooth communication between the application and other external tools like Shodan.

4. Define API Keys: Replace the keys with actual ones from Shodan, ZoomEye, and Hunter. Those keys authenticate the requests that are sent to these services so that the framework is able to query their databases. Shodan and ZoomEye return data pertaining to connected devices, and Hunter fetches data on domains and emails. The right API integration of these keys will be sure enough to enable the fetching of relevant data for penetration testing.

5. Definition of URL and IP validation Pattern: This regex pattern is designed to validate the user input, including the URLs and IP addresses. It allows only the correct formats for processing; this reduces more chances of errors while making a query to an external API. It verifies several formats, such as HTTP, HTTPS, IPv4, and IPv6 address, and the domain name as well. If the earliest stage of input validation is used, it does not process any kind of invalid or malicious data, which reduces more accurate results and boosts the level of security.

6. Apply Base64 URL encoding: Base64 URL encoding is safely applied on the query strings to encode them before their transmission in API requests. Such encoding handles special characters that may disrupt the HTTP over a network, including a space or symbols like + and /. This function ensures that the query strings are

transformed into a safe format to be transmitted over the network without any errors. Base64 Encoding is a must, so every type of user input, whether complex URLs or strings, should be processed perfectly and securely.

7. Query the Hunter API: A function is created that sends queries to the Hunter API, fetches information about domains and email addresses. The function encodes the query string if needed and then processes the response from the API. Hunter is very helpful in trying to gather information, like email patterns that may be related to a domain or even find information concerning the ownership of the domain. With the Hunter API, the framework can enable the penetration tester.

8. Set up index page route: In the Flask application, it defines a route for index page (/) to handle both GET and POST requests. When the query form is submitted, the route checks whether the input is matched with one of URL, domain or IP address by the defined regex depending on which type, then calls functions that will query Shodan, ZoomEye or Hunter APIs respectively. The results are then sent back to the user with detailed network-related information retrieved from the respective services to ensure a smooth user experience.

9. Define the following functions to download data from Shodan and ZoomEye: Implement query functions that interface with the Shodan and ZoomEye APIs for different types of inputs such as an IP address, a domain, or port retrieval of the relevant information. These should be able to send API requests and then process a response along with extracting whatever information may be relevant pertaining to the device, vulnerability, or open ports in question. The functions should be efficient while gathering and formatting data to incorporate into the penetration testing framework.

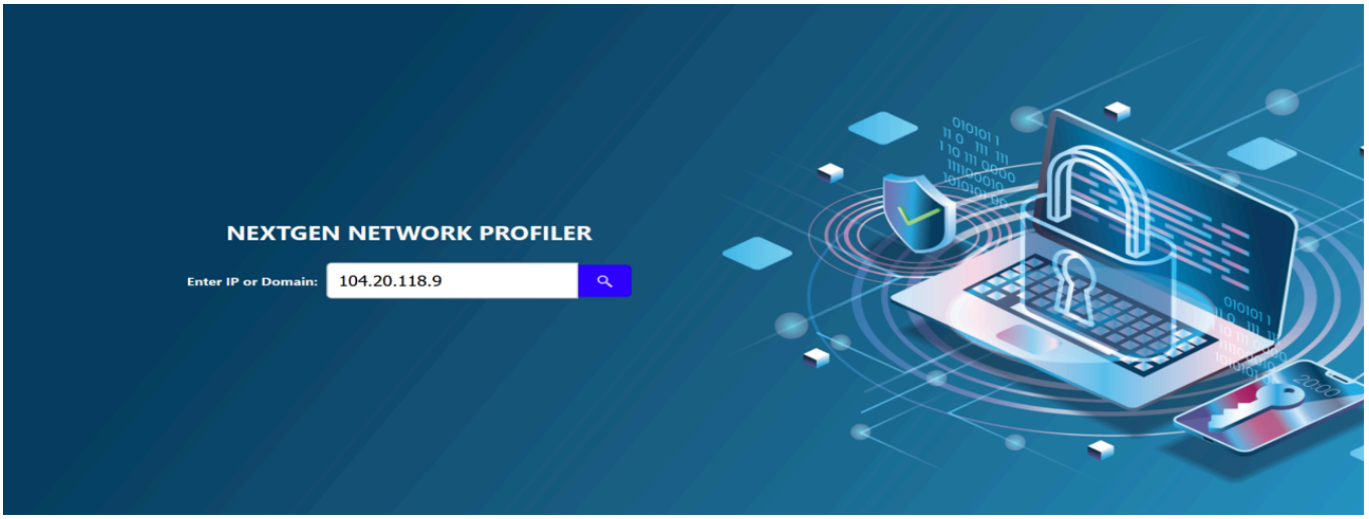
10. Define a function that returns URL information: Create a dummy function. Most probably, it's going to fetch information on a URL. In this instance, it should act as if it's fetching some information on the URL or the SSL certificates or perhaps the response headers or maybe even DNS data. But the function does not commit it to an actual API. It is rather an ante-cursor that makes it possible for the framework to respond to whatever query is built on the URL, a starting point for developing other functionalities.

11. Run the Flask: App Run the Flask application in debug mode, enabling development and testing. The opportunity to report errors immediately while running an app in debug mode will include live reload of code changes, print of detailed stack traces, and fast resolution of bugs while developing the framework. In this regard, the application will run as expected then further allow testing and deployment.

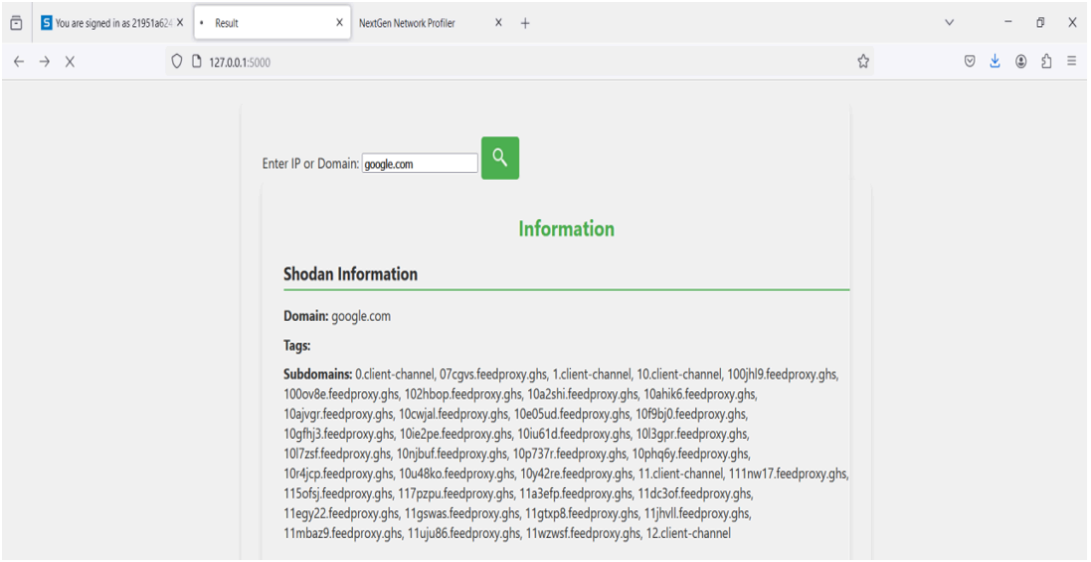
RESULTS

This penetration testing framework on basis of network fingerprint in huge networks takes much more time to complete the entire procedure in relation to the completion by other frameworks, this because of the fact of taking data from different areas along with data processing need, however, this specific large network-based network fingerprint-based penetration test framework possesses a special advantage against hit and its thorough-going testing efficiency is also exceptional. The advantages of this network fingerprint-based penetration testing methodology for large-scale networks are as follows based on the test results:

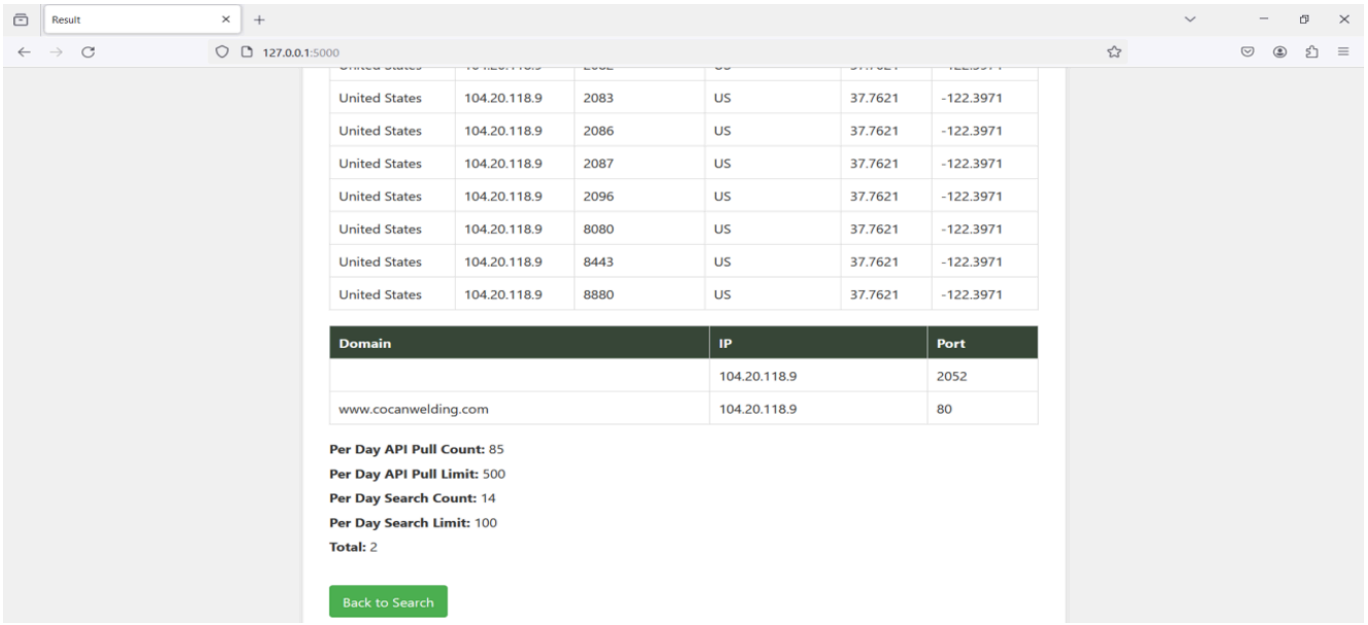
The testing patience: The patience of testing has been made more robust. Target set quality has also improved along with a significant rise in the number of related target identifications. Along with these, there will be more hits in a set time period.



Much testing resource saved: You have saved much duplicated effort by bringing the search engine back into cyberspace to replace some part of the active scanning activity. This effort could make good use of some public data resources on the Internet. The IP addresses which are obtained with help from Shodan come out useful for security investigation and analysis.



They are connected to many network infrastructure elements, including servers, webcams, switches, and routers. With Shodan, users can determine the configurations and vulnerabilities related to these devices and thus contribute to network management and cyber security.



DISCUSSION

Comparison to Traditional Methods:

The traditional penetration testing approach is often time-consuming and resource-intensive, particularly when dealing with large networks that have numerous attack surfaces. These manual procedures can also carry risks to business continuity by interfering with network operations. The rigidity of older automated test scripts limits their flexibility to adjust to the constantly changing cybersecurity landscape.

Traditional methods that rely on high-speed port scanning (like Z-Map and Masscan) are often inadequate for furnishing detailed information beyond port status, which can lead to false positives and inefficient target selection for deeper testing. Similarly, automated test scripts used in existing systems may fail to keep up with evolving threats and can miss complex vulnerabilities.

Advantages of the Proposed Framework:

The proposed framework breaks these constraints by incorporating advanced techniques like **Cyberspace Search Engines** and the **Nmap Scripting Engine (NSE)** plug-ins into the network framework. This novel approach provides several key advantages:

- **Improved Target Quality and Hits:** Leveraging network fingerprint recognition technology and data from cyberspace search engines allows the framework to more accurately identify relevant targets, significantly improving the **quality** of the target set and increasing the number of successful "hits" within a set timeframe.
- **Resource Efficiency:** Integrating cyberspace search engines (like Shodan, ZoomEye, and Censys) replaces a portion of active network scanning activity. This efficient use of publicly available data saves significant testing resources and minimizes unnecessary active scanning efforts.
- **Enhanced Capability:** Utilizing NSE plug-ins increases Nmap's capabilities, reduces scanning times, and overall increases the efficiency of the framework. Cyberspace search engines are applied to map the Internet, which makes it easier to examine infrastructure and devices efficiently.

Framework Structure and Implementation:

The framework is structurally divided into three main modules: **Target Acquisition**, **Data Process**, and **Test**. Key components of the framework's implementation include:

- **Cyberspace Search and Network Fingerprinting:** Tools like **Shodan** and **Censys** are used to find target devices and potential vulnerabilities. Network fingerprinting analyzes open ports and services, which helps eliminate false positives that often occur with traditional scanning methods.
- **Automation:** Automated testing scripts are executed using the information picked up from the advanced search engines. Vulnerabilities found by **NSE scripts** can be exploited and verified using **Metasploit** to gain a proper view of the network's security posture.
- **Implementation Tools:** The framework uses **Flask** for the web application, **Shodan** and **requests** for working with third-party APIs, and the **Base64** module for secure data handling.

Constraints and Future Direction:

Despite its efficiencies, the framework has constraints. The **timeliness of the targeted information** is the most significant issue, as the update frequency of data from internet search engines can vary from days to months, potentially leading to inconsistencies in real-time information. Furthermore, the inherent risk of missing key information exists due to the data collection capacities of the cyberspace search engines.

Therefore, even though the framework is highly efficient, it must be used in conjunction with other methods to ensure a complete, modern penetration testing procedure. Future scope for development includes integrating **AI and machine learning** to enhance target identification and vulnerability evaluation, developing **real-time threat intelligence feeds** to address data timeliness, and **automating remediation and reporting** actions.

CONCLUSION

Network fingerprinting-based penetration testing frameworks for large networks bring notable productivity and resource usage benefits. The framework may more accurately identify relevant targets by leveraging network fingerprint recognition technology, enhancing the quality and relevance of testing efforts. By making efficient use of publicly available data from internet search engines, this approach minimizes unnecessary active scanning and maximizes the number of successful finds within a given timeframe. As a consequence, this technique optimizes the testing assets and increases the overall capability of comprehensive security assessments.

This framework has some constraints. The most significant constraint is to the timeliness of the targeted information as the update frequency of data from the internet search engines may differ from a few days up to several months, leading to potential inconsistencies in the real-time targeted information. Moreover, the inherent risk of missing key information is due to data collection capacities of cyberspace search engines. That is why, even if the framework is very efficient, saves a lot of resources, and is used individually, it is necessary to put it together with other methods to ensure a complete, modern penetration testing procedure.

The network fingerprinting-based penetration testing framework along with cyber space search engines, leaves ample promising avenue for further developments and sophistications:

Advancement of AI and Machine Learning: Integration of latest AI and machine learning will make precise the target's identification, and vulnerability evaluation. Handling of huge volumes of data through Cyberspace search engines may be made better for detecting patterns, anomalies, and emerging threats.

Real-Time Threat Intelligence: Feeds must be developed for real-time threat intelligence that can talk about timeliness to information pertaining to the target by updating and refreshing the same data being fetched from other sources from the cyberspace search engine, hence providing minute insights that are evolving about cybersecurity risk.

Automate Remediation and Reporting: Improved automatic remediation action for detected vulnerabilities will enhance the mitigation. Reporting can also be enhanced with a comprehensive view along with actionable recommendations to aid the scope of decision making for a cyber security team. This framework is optimized for scaling with support to the larger network environment and ease in global operations of cybersecurity. Integration of clouds can also enhance reach, flexibility, and use of resources.

REFERENCES

[1] Yan Shujun, Wang Wenjie, Zhang Yuqing,2019, An effective web fingerprint identification method Journal of the Chinese Academy of Sciences, 33(05): 679- 685.

[2] Dukes L S, Yuan X, Akowuah F.2018 A case study on web application security testing with tools manual testing[C]//Southeast, Proceedings of IEEE. IEEE, 2014:1-6.

[3] Huang Z, Xia C, Sun B, et al. Analyzing and summarizing the webserver detection technology based on HTTP[C]//Software Engineering and Service Science (ICSESS), 2016.

[4] Goethem T V, Chen P, Nikiforakis N, et al. Large-scale security analysis of the web: challenges and findings[J]. Lecture Notes in Computer Science, 2016, 8564:110-126.

[5] Ma Cheng. Principle Research and Security Application of Cyberspace Search Engine [J]. Cyberspace Security, 2016, 7(05): 6-10. Zhao Jianjun. Research on Identification Technology of Network Space Terminal Equipment [D]. Lanzhou Technology University, 2016.

[6] Chen Zhuo. Design and implementation of scanning platform for industrial control systems based on stateless connection [D]. Beijing University of Posts and Telecommunications, 2015.REFERENCES.

[7] Lu Wendi, Li Xiaohui, Wang Han. Research and Implementationof Security Access Method for Internet of Things Based on Device Fingerprint [J]. Electronic Design Engineering, 201, 27(03): 136- 141.

[8] Huang Z, Xia C, Sun B, et al. Analyzing and summarizing the web server detection technology based on HTTP[C]//Software Engineering and Service Science (ICSESS), 2015 6th IEEE International Conference on. IEEE, 2020:1042-1045.

[9] Karthik R, Kamath S. W3-Scrape-A windows-based reconnaissance tool for web application fingerprinting[R]. arXiv:1306.6839.

[10]Kozina M, Golub M, Groš S. A method for identifying Web applications[J]. International Journal of Information Security,2014,8(6):455-467.

[11] Lee D, Rowe J, Ko C, et al. Detecting and defending against Web-server fingerprinting[C]//Proceedings of Computer Security Applications Conference, 2002. 18th Annual. IEEE, 2021:321-330.

[12] Atana worabhan S, Livshits B, Zorn BG. Noxes: A client- side solution for mitigating cross-site scripting attacks. In Proceedings of the Network and Distributed System Security Symposium (NDSS), 2006.

[13] BGP by cautiously adopting routes. In Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, 2003.

[14]Karasaridis A, Rexford J. Pretty Good BGP: Improving BGP by cautiously adopting routes. In Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, 2003.

[15]Chatzimisios P, Boucouvalas AC, Vitsas V. Security issues in mobile ad hoc networks— a survey. International Journal of Communication Systems, 2003

[16][1] Holz T, Gorecki C, Rieck K, Freiling FC. Measuring and detecting fast-flux service networks. In Proceedings of the 1st USENIX Workshop on Large-Scale Exploits and Emergent Threats, 2002.

[17] Maniraj, S. P., Saini, A., Ahmed, S., & Sarkar, S. D. Credit Card Fraud Detection using Machine Learning and Data Science. International Journal of Engineering Research & Technology (IJERT), 8(09).

[18] Dubey, S., Mundhe, K., & Kadam, A. (2001). Credit Card Fraud Detection using Artificial Neural Network and BackPropagation. Proceedings of 4th International Conference on Intelligent Computing and Control Systems, Madurai.

[19] Yu, X., Li, X., Dong, Y., & Zheng, R. (2000). A Deep Neural Network Algorithm for Detecting Credit Card Fraud. International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Fuzhou, China, 181-183.

[20] Jebaseeli, T. Jemima, Venkatesan, R., & Ramalakshmi, K. (2000). Fraud detection for credit card transactions using random forest algorithm. Intelligence in Big Data Technologies—Beyond the Hype, Springer, Singapore, 189- 197.