NIST SP 800-115 Framework Implementation using Black Box Method on Security Gaps Testing on JTD Polinema's Official Website

Dina Nurika Fitriana[1], Putri Elfa Mas'udia[2], Mila Kusumawardani[3]

[1,2,3] Digital Telecommunication Network Study Program,
Electrical Engineering, State Polytechnic of Malang, Indonesia

[1]fdinanurika@gmail.com, [2]putri.elfa@polinema.ac.id, [3]mila.kusumawardani@polinema.ac.id

***Abstract*—The internet is one example of a computer network that can make it easier to obtain information. According to BSSN's December 2021 report, there were 3,483,706 web application attacks. According to the BSSN monthly report, there were 3,483,706 web application attacks at the end of December 2021. The JTD Study Program's official website (psjtd.polinema.ac.id) faced recurrent hacking incidents, exposing it to DDOS assaults and defacing. As a result, security testing must be carried out in accordance with particular standards, such as the National Institute of Standards and Technology (NIST) SP 800-115 framework. Penetration testing was performed in this investigation using the Black Box testing method approach and hardening. The results of testing and analyzing security gaps on the website reveal 10 open ports and 11 various types of security holes with varying levels of vulnerability categorized as 1 high, 3 medium, 5 low, and 2 informational. During penetration testing, one ping packet was sent that could not cause any problems, and then one of the Syn Flooding attacks was carried out, which resulted in the number of shipments reaching 10,000 packets per second.**

Keywords— Black Box Method, Hardening, NIST, Security Gaps, Website.

I. INTRODUCTION

The existence of a computer network can make it easier to access information, one example of a computer network is the internet. This network connects millions of computers spread across the globe. The growth of the internet is one of the largest network implementations in the world, allowing all information in the world to be easily obtained. With the many benefits provided by computer networks, the need for network utilization in conducting data communication is growing rapidly from time to time, coupled with an increasingly cheaper internet connection. Currently, there is a large amount of analog and digital data transferred between business networks around the world in the form of data transmission, therefore it will be very easy for the information to be obtained anytime and anywhere can still access the data contained on the server when we need it. The features provided by the internet network are also so many varieties, one of which is a web server where which is software that provides services in the form of data for clients where the request is given in the form of a website page. One of the inseparable developments of the internet is the website [1]. A website is a collection of pages of a domain containing various information so that it can be accessed easily anytime, and anywhere by anyone via the internet. According to Kominfo, internet users in Indonesia increased by 11% from the previous year, which was 175.4 million to 202.6 million internet users. However, in recent months there have been several cases on the website. According to the monthly report of the State Cyber and Password Agency (BSSN) since the end of 2019 when the outbreak of the Covid-19 pandemic began to increase the occurrence of cyberattacks, to be precise, from January 1 to April 12, 2020, there have been 88,414,296 cyberattacks occurred. At the beginning of 2022, Indonesia occupies the third most position in the world affected by cyberattacks after the positions of Bangladesh and the United States, judging from Honeypot BSSN data on January 1, 2022 to February 2, 2022, there have been 1,038,354 cyberattacks that have occurred. Website applications are very vulnerable to external attacks. At the end of December 2021, based on the results of the BSSN monthly report, there were 3,483,706 web application attacks [2]. Digital Telecommunication Network Study Program (PS-JTD) of Polinema has an official website that contains various information, besides that the PS-JTD website is also used as a means to introduce PS-JTD profiles and also as university branding. Based on the results of observations by interviewing Mr. Usman, who used to be the admin of PS-JTD, it is known that the PS-JTD website has experienced several hacks. One of the impacts is the exposure of a DDOS attack, this is an attack on a computer network server that results in the server going down and a de face attack that can result in a change in the appearance of the website.

This can also happen again on the PS-JTD website. Usually, this happens because of errors when using web creation program code, this is very often used to hack the web, attacks that are often used by attackers are SQL Injection, XSS and Authentication as well as ransomware incidents and other attacks. This is very disturbing because if there are security loopholes/vulnerabilities that can cause losses, so find out what are the security loopholes on the website, you can test the website of the JTD study program by doing it legally resembling a hacker. Therefore, security testing in the form of

penetration testing is needed using the National Institute of Standards and Technology (NIST) SP 800-115 framework to find out security loopholes in the potential to harm the website.

In response to the identified challenges, a study is proposed titled 'Implementation of the NIST SP 800-115 Framework Using the Black Box Method for Evaluating Website Security Gaps in the JTD Polinema Study Program.' This research involves the comprehensive analysis and testing of security vulnerabilities, leveraging the standardized framework outlined by NIST SP 800-115. The NIST framework of this nature offers a systematic approach to conduct penetration testing, incorporating stages that yield insightful recommendations and proposals. The chosen approach for penetration testing involves the Black Box method, wherein external testers assess the system without prior knowledge of any information related to the website.

NIST (National Institute of Standards and Technology) is an information security framework developed by the United States government organization, aiming to enhance an organization's capabilities in preventing, detecting, and responding to cyber attacks [3]. One of the advantages of the NIST framework is its structured and planned format, making it convenient for organizations to implement at the enterprise level [4]. Essentially, the NIST framework is organized into five core functions collectively known as the core, serving as a high-level overview of an organization's cybersecurity posture, including Identification, Protection, Detection, Response, and Recovery [3]. The NIST CSF consists of 5 core functions, 23 categories, and 108 subcategories necessary for implementation in information system audits [5].

NIST provides tools, techniques, and methods for assessing and planning risk-based information security. NIST standards serve as a reference for conducting information security risk management, aiming to anticipate risks to prevent losses for the organization. Thus, risks can be identified, assessed, and mitigated to levels acceptable to the organization. Through risk management, it is expected to reduce the impact of incidents on the information system and technology in higher education institutions, protect critical organizational business processes from security threats, minimize the risk of losses, and avoid serious failures in the organization's information.[6]

In this study, NIST SP 800-115 is employed. NIST SP 800-115 produces key technical assessment techniques, namely target identification, technical analysis, and validation of vulnerability targets [7]. NIST addresses four main fields, including biotechnology, nanotechnology, information technology, and modern manufacturing [8]. NIST SP 800-115 guidelines encompass several stages such as Planning, Discovery, Attack, and Report, which serve to facilitate the website testing process [9].

Testing a website is crucial for maintaining its quality and integrity [10]. It instills confidence in users that the application's functionality is reliable, encouraging them to use it without hesitation [11]. Websites are susceptible to various vulnerabilities, with common occurrences like input validation issues or vulnerabilities related to input forms [12].

The testing methodology employed here is black box testing, which focuses on inputs and outputs without delving into the application's internal processes [13]. Black box testing, particularly with fuzzing techniques, serves as an effective alternative for testing. This approach tests based on inputs and outputs without considering program details, making it accessible for testers without programming knowledge [14]. Black box testing is typically performed after a website's publication [15]. Although the website already has the capability to handle error situations, manage various abnormal data, and scripts [16].

II. METHOD

A. Research Design

Stages of the research to be carried out are stated by the flow chart shown in Fig. 1.

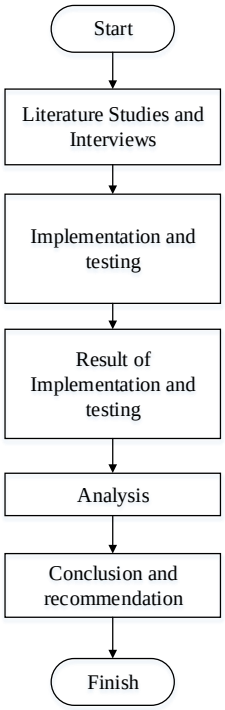

Figure 1. Flowchart of research stages

The explanation of the research design flow chart is as follows:

1. Literature Study & Observation
   Observing problems in the environment, this literature study aims to be able to set research goals appropriately. Literature studies are then carried out by searching for references such as journals, studying supporting books with topics that are in accordance with the research made. Interviews were conducted to find out the real incident of the JTD study program website being hit by a hacking attempt.
2. Testing and Implementation
   This test is an implementation stage on the JTD Polinema Study Program website. This test performs penetration testing based on the NIST SP 800-115 framework and applies hardening.

3. Test and Implementation Results
   At this stage, the types of security gaps/vulnerabilities on the JTD study program website are obtained with different levels of vulnerabilities, namely high, medium, and low.
4. Analysis
   The stage of deciphering each test result that has been carried out according to the plan. The results of the analysis can be used as a basis for creating a report.
5. Conclusions and suggestions
   Make conclusions from the results of the stages that have been carried out based on testing and provide suggestions based on the results of the test.

This testing flowchart explains implementation flow in data retrieval quality of service protocol BATMAN for salinity data transmission that shown in the Fig. 2.

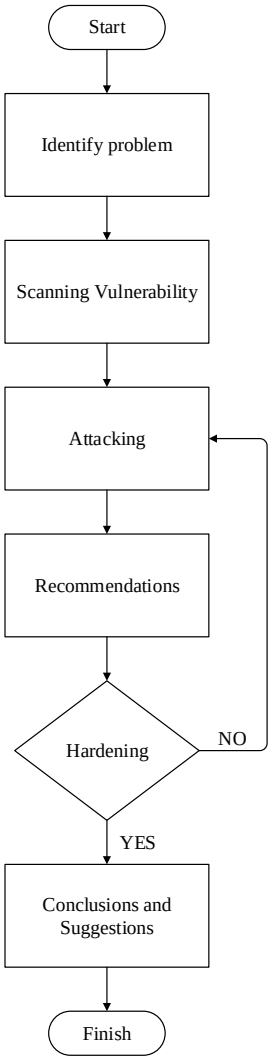

Figure 2. Flowchart of testing stages

The explanation of testing stages flowchart is in the first stage, namely the identification of problems by looking at existing phenomena, looking for information using literature studies. In this study, the NIST SP 800-115 framework applied the Black Box method and used several tools. In scanning vulnerabilities, it aims to get the types of security loopholes on the JTD Polinema study program website. For attacking here, it is used as an attack test carried out on security gaps obtained from the results of scanning ports and scanning websites. Furthermore, hardening, this hardening aims to strengthen or secure a website from attackers. Finally, make conclusions and suggestions obtained from the results of tests and analyzes that have been carried out.

B. System Design

How testing works using the NIST SP 800-115 framework in testing website security gaps and hardening can be seen based on Fig. 3.

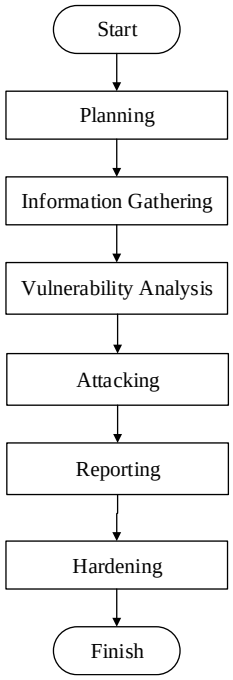

Figure 3. Flowchart of testing

To conduct penetration testing in this study, the NIST framework is used, which has 4 stages, namely planning, discovery, attacking, and reporting. For hardening implementations, this is not included in the NIST SP 800-115 framework. This hardening is only done in security gaps or vulnerabilities that have a high risk.

1) Planning: In this planning stage, the examiner conducts planning to determine the targets, methods, goals and tools that will be used to conduct the test.

2) Discovery: In this discovery stage, it has 2 important steps, namely information gathering and vulnerability analysis. Information gathering is used to collect data or information about websites such as IPs, open ports and hosts, this test uses commands in KaliLinux in the form of ping, whois, and host. The following flow chart from the Information gathering can be seen in Fig. 4:

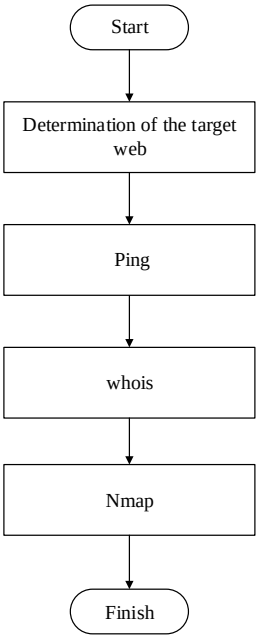

Figure 4. Flowchart of information gathering

In this vulnerability analysis step, it is used to scan using tools, one of which can use the OWASP ZAP tool and perform analysis to detect security gaps in the target web that will be utilized by attackers. The results of the existing security gaps will be divided into 3 categories, namely Hight Priority Alerts, Medium Priority Alerts, Low Priority Alerts and Information Priority Alerts. The following vulnerability analysis step can be seen in the Fig. 5.

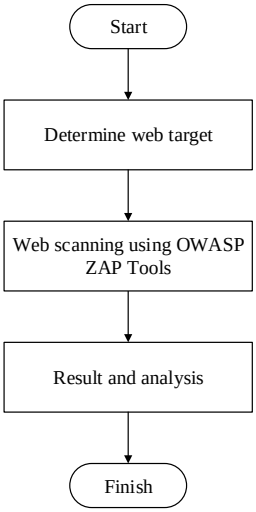

Figure 5. Flowchart of vulnerability analysis

3) Attacking: At this stage of attacking, attack testing is carried out on websites that have the information have been collected. The purpose of attacking is to conduct security testing on the website based on the results of the analysis of security gaps/vulnerabilities in the results of port scanning and discovery stages. The test scenario at the attack stage can be seen in Fig. 6.

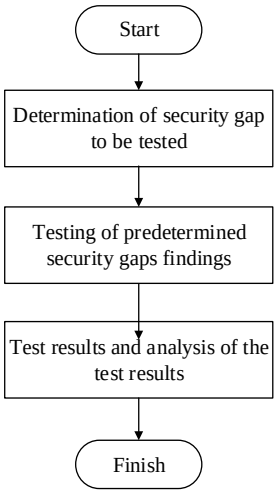

Figure 6. Flowchart attacking

4) Reporting: At this reporting stage, analyze the results of the website attack that has been carried out. Reporting contains a report on the success or failure of an attack carried out on the website of the JTD Polinema study program.

5) Hardening: At this stage, hardening or securing the website based on the results of security loopholes/vulnerabilities obtained with high risk or the results of attacking or providing suggestions or recommendations so that the website becomes more secure (secure). The test scenario can be seen in Fig. 7.

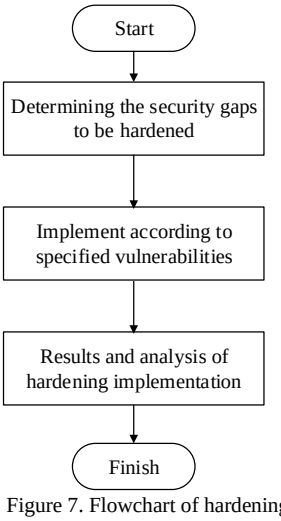

Figure 7. Flowchart of hardening

C. *Test Modeling*

The scenarios used in testing website security gaps are as follows:

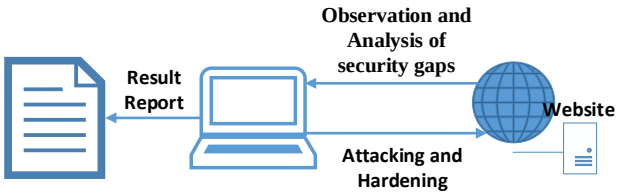

Figure 8. The test modeling

The test scenario for security gaps carried out is to observe or collect data about the website. This data collection is necessary because testing uses the black box method, where the black box method itself is a method that is carried out by a hacker who is not equipped/knows any information about the web to be tested, for the results of this data collection can be seen in table 1 about website information.

TABLE I
WEBSITE INFORMATION

Builtwith.com	Wappalyzer.com
- Web server : Nginx 1.16 - Oprating sistem and servers : Ubuntu - Documen standart : Java Script - Framework : Organization Schema - Content Management System : LearnPress - SSL Certificates : LetsEncrypt dan HSTS	- Programing languges : PHP - Databases : MySQL - Reverse proxies : Nginx 1.16.1 - Caching : WP Rocket - Oprating sistem : Ubuntu - Javascript framework: GSAP dan Backbone.js 1.4.0 - UI Framework : Bootstrap 3.2.0 - Web server : Nginx 1.16 - Widges : OWL Carousel - SSL / TSL sertificate authories : Sectigo - Java script libraries : jQuery UI 1.12.1, jQuery 3.5.1, Underscore.js 1.8.3, jQuery Migrate 3.3.2, core.js 2.6.11 dan Modernizr 2.8.2 - Font scripts : Google Font API dan Ionicons

Furthermore, an analysis of security gaps on the website will be carried out, this is done by scanning vulnerabilities to find out the type and level of security gaps. Furthermore, attacks are carried out in the form of testing attacks on testing the findings of website and port security loopholes and implementing them based on the security loopholes found. From the results of testing security gaps, test results can be made that security gaps have been carried out.

D. Block Diagram

The following block diagram used in testing website security gaps is shown in Fig. 9.

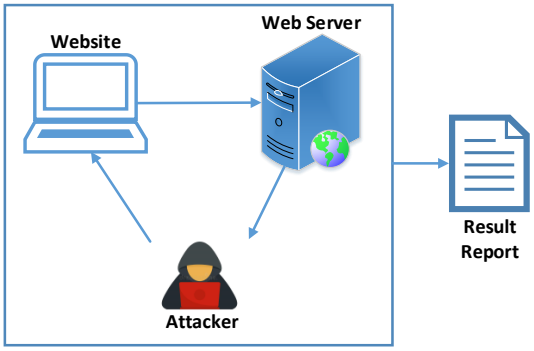

Figure 9. Block diagram

Attackers access the website to be tested by taking the url on the website for scanning to find out security loopholes on the website. Then an attack is carried out on the web server, where web server has an important role in controlling a website. After that, the results of testing that security loopholes have been carried out regarding the success or failure of these types of security loopholes can be attacked. Furthermore, hardening the website that has been attacked by the attacker.

III. RESULTS AND DISCUSSION

A. Results of Port Scanning

The results of port scanning of the JTD Polinema (psjtd.polinema.ac.id) official website using the Nmap tool with IP Address 103.154.145.5 show the ports, states, and services that have been scanned. There are several ports open on the psjtd.polinema.ac.id website including port 21, port 80, port 110, port 143, port 443, port 465, port 587, port 993, port 995, and port 2030.

```
PORT     STATE SERVICE
21/tcp   open  ftp
80/tcp   open  http
110/tcp  open  pop3
143/tcp  open  imap
443/tcp  open  https
465/tcp  open  smtps
587/tcp  open  submission
993/tcp  open  imaps
995/tcp  open  pop3s
2030/tcp open  device2
```

Figure 10. Results of port scanning

B. Vulnerability Analysis

In this Vulnerability Analysis step, scanning is carried out on the website to find out what security loopholes or vulnerabilities are found on the website psjtd.polinema.ac.id and the level of vulnerabilities on the website.

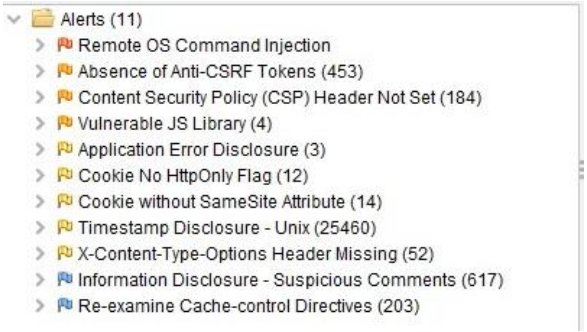

Figure 11. Results of Vulnerability Analysis

Based on the results of scanning vulnerabilities with the OWASP ZAP tool that has been carried out, the results of different types and numbers of security gaps along with the risk of vulnerabilities are obtained. The results are as follows:

TABLE II
DISCOVERY GAP DISCOVERED

Level	Total
High	1
Medium	3
Low	5
Informational	2

In the picture above, there is an alerts box with 11 different types of security loopholes and vulnerabilities, which means that websites psJTD.polinema.ac.id are quite vulnerable to external attacks. Each colour on the alerts has a different meaning, for red means, it has a very fatal or high level of vulnerability to attacks from outside, for orange means, it has a medium vulnerability level that is not too risky but also not too safe from attacks, then for yellow has a low vulnerability level or is difficult to do attack, and finally for blue which is to display information about website security that has been scanned. The following table of vulnerability scanning results:

TABLE III
VULNERABILITY SCANNING RESULT

Security Gaps	Vulnerability levels	Total
Remote OS Command Injection	High Priority Alerts	1
Absence of Anti-CSRF Tokens	Medium Priority Alerts	453
Content Security Policy (CSP) Header Not Set	Medium Priority Alerts	184
Vulnerable JS Library	Medium Priority Alerts	4
Application Error Disclosure	Low Priority Alerts	3
Cookie No HTTPOnly Flag	Low Priority Alerts	12
Coockie without Samesite Attribute	Low Priority Alerts	14
Timestamp Disclosure-Unix	Low Priority Alerts	25.460
X-Content-Type-Options Header Missing	Low Priority Alerts	52
Information Disclosure-Suspicious Comments	Information Priority Alerts	617
Re-examine Cache-control Directives	Information Priority Alerts	203

From the scanning results, the types of security gaps with different numbers of security gaps were obtained and 1 type of security gaps / vulnerabilities with a high level of vulnerabilities (High Priority Alerts) were obtained. Where this security gap with a high level of vulnerability has a large or fatal risk of being exposed to attacks from outside.

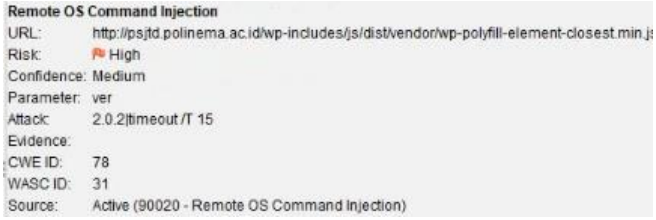

Figure 12. Security gaps with the highest level of vulnerability

From the picture, it can be seen that the JTD Polinema study program website has a security gap with a high level of vulnerability, namely Remote OS Command Injection. Remote OS Command Injection is also known as shell injection where a web security vulnerability allows an attacker or attacker to run a form of attack where the attacker's goal is to execute arbitrary commands on a web server through this vulnerable website.

C. Attacking using Syn Floods

At this stage of attacking, DDoS attacks are carried out on the TCP (Transmission Control Protocol) protocol, this TCP protocol is very widely used by internet users because this protocol easily manages communication between one user and another user to exchange data with each other. In this test, the network topology used in the study used the client server network type by utilizing the internet network publicly in flooding the target website with demand packages that exceeded the target website's hardware services. By flooding the TCP protocol so that normal data becomes data flooding on a website.

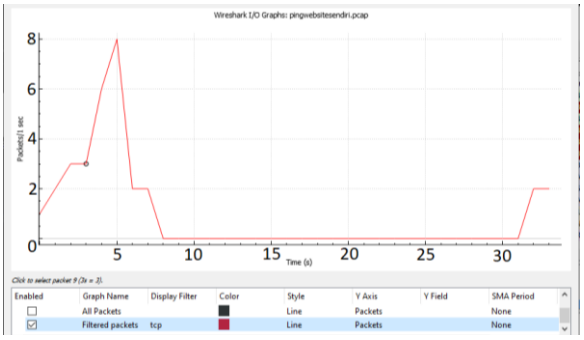

Figure 13. Before syn flood

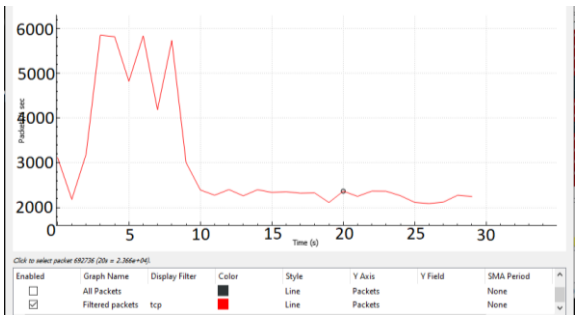

Figure 14. After syn flood

In this test, sending as many requests as possible to the JTD study program website server resulted in the number of packets being sent per second reaching 10,000 packets. After the syn flood attack is carried out, it can be seen that the listing of

packets captured on wireshark is dominated by red packets which indicate a packet attack that flooded the server. These results can also be seen that there are many requests (RST) and ACKs that occur, this is very different from the first test which only sent one ping packet. This syn flood attack exploits the weakness of the TCP connection, the attacker sending random TCP SYN packets to the destination host will send back SYN ACK packets. This type of attack is quite difficult to detect the user's address because the IP address of the sender has been disguised by selecting packets connected to the internet network, with the sender's address that has been disguised, when the SYN packet arrives at the server, then the server will allocate the necessary memory buffer. Then if the memory allocation has been given to the attacker's host, the attacker's host will continue to send SYN packets that have been manipulated by the attacker and the IP address that has been disguised. The attacker host will force the server to accumulate half open connections so that at its peak position the server is unable to accumulate half-open connections so that the server's resources are totally paralyzed.

D. Attacking on security gaps Remote OS Command Injection

The JTD study program website has one security gap/vulnerability with a high category, namely Remote OS Command Injection. Attacking the security loophole is said to be unsuccessful because when attacking the security loophole, there is an obstacle, namely the security loophole cannot be attacked. Therefore, a scan of the website was carried out again using the OWASP ZAP tool.

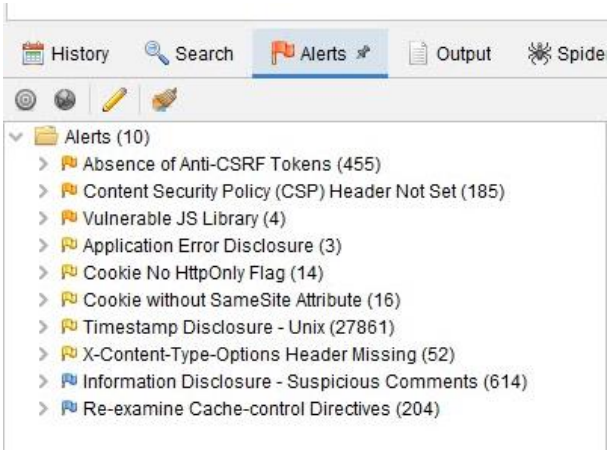

Figure 15. Scanning results when attack running

E. Reporting

From the previous stages, there are security loopholes that cause losses or risks to the target web. The following is a report of the results of attacking using the NIST SP 800-115 framework:

TABLE IV
REPORTING TABLE

Security Gaps	Details	Attacking Results
Port 80	Syn Flood attack is carried out where this attack will flood the server with false requests in a barrage, exploiting and consuming network resources.	Successful
Remote OS Command Injection	This attack is carried out to execute arbitrary commands on the web server through a vulnerable web application. This type of attack can directly impact the server, such as shutting down the server, viewing the contents of the web directory or deleting the server's web directory.	Unsuccessful

F. Hardening

On the JTD Polinema study program website there are ten open ports, where the results of these ports can be seen from the results of port scanning using the nmap tool. Ports that are open but not used on website services can be used as a way for attackers to carry out attacks, therefore it is better to close ports that are not used. Closing open ports on websites is one way of hardening or handling security holes that take advantage of open ports. Closing unused ports can be done by executing the command "sudo ufw deny (target port/port to be closed)".

The Syn Flood attack can be handled in various ways, one of which is by using iptables. Iptables is a firewall tool, with a function to secure the network by filtering traffic on servers. To use iptables this can be done by typing the command "iptables -A INPUT -s (target IP to be blocked) -j DROP".

Software update on the server is done to update the software on the web server including the library used. These updates are not only to protect users from security risks, but also to fix bugs and loopholes. So that attackers who take advantage of this gap can easily be resolved.

IV. CONCLUSION

Based on the planning and testing of the results of the research "Implementation of the NIST SP 800-115 Framework with the Black Box Method on Testing Website Security Gaps of the JTD Polinema Study Program" the following conclusions can be drawn: Security gaps on the website can be found by conducting pentration testing which is run on the KaliLinux operating system in accordance with the NIST SP 800-115 standard and can be done using the black box testing method and with the help of several tools such as Nmap, OWASP ZAP and hping3. From the Nmap tool, the results of several open ports are obtained, namely port 21, port 80, port 110, port 143, port 443, port 465, port 587, port 993, port 995, and port 2030. And from the OWASP ZAP tool, the test results of 1 high-risk level, 3 medium-risk levels, 5 low-risk levels, and 2 informational risk levels with the highest level of security gaps were Remote OS Command Injection which opened a gap to carry out attacks to execute arbitrary commands on the web server. The recommendations given for the JTD Polinema study program website are to close ports that are not used by the website, handle iptables to block syn flood attacks that utilize open ports, and make updates to the software. The

recommendations given for the JTD Polinema study program website are to close ports that are not used by the website, handle iptables to block syn flood attacks that utilize open ports, and make updates to the software. For suggestion the testing for security vulnerabilities can be done with different tools to get different security vulnerabilities.

REFERENCES

[1] P. Agustini, "Warganet Meningkat, Indonesia Perlu Tingkatkan Nilai," aptika.kominfo.go.id, 2021.

[2] B. S. D. S. N. (BSSN), "Laporan Bulanan Hasil Monitoring Keamanan Siber Nasional," Id-SIRTII/CC, 2021.

[3] D. Sulistyowati, F. Handayani, and Y. Suryanto, "Comparative analysis and design of cybersecurity maturity assessment methodology using nist csf, cobit, iso/iec 27002 and pci dss," *Int. J. Informatics Vis.*, vol. 4, no. 4, pp. 225–230, 2020.

[4] P. P. Roy, "A High-Level Comparison between the NIST Cyber Security Framework and the ISO 27001 Information Security Standard," *2020 Natl. Conf. Emerg. Trends Sustain. Technol. Eng. Appl. NCETSTEA 2020*, vol. 53, pp. 27001–27003, 2020.

[5] H. Sama, Licen, J. S. D. Saragi, M. Erline, Kelvin, Y. Hartanto, J. Winata, and M. Devalia, "Studi Komparasi Framework NIST dan ISO 27001 Sebagai Standar Audit Dengan Metode Deskriptif Studi Pustaka," *Rabit: Jurnal Teknologi dan Sistem Informasi*, vol. 6, no. 2, pp. 116-121, 2021.

[6] F. Mahardika, "Manajemen Risiko Keamanan Informasi Menggunakan Framework NIST SP 800-30 Revisi 1 (Studi Kasus: STMIK Sumedang)," *Jurnal Informatika:Jurnal Pengembangan IT (JPIT)*, vol. 2, no. 2, pp. 1-8, 2017.

[7] R. A. Wibowo and S. Widyarto, "Kajian Pustaka: Penetration Testing dengan NIST SP 800-115 dan OSSTMM", Proceedings of the Informatics Conference, vol. 6, no. 10, 2020.

[8] E. Z. Darojata, E. Sediyonob, I. Sembiring, "Vulnerability Assessment Website E-Government dengan NIST SP 800-115 dan OWASP Menggunakan Web Vulnerability Scanner," *Jurnal Sistem Informasi Bisnis*, vol. 01, 2022.

[9] F. Hanifah, A. Budiyono, and A. Widjajarto, "Analisa kerentanan pada Vulnerable Docker menggunakan alienvault dan docker bench for security dengan acuan framework Cis Control," in *e-Proceeding of Engineering*, vol. 8, no. 5. 2021.

[10] B. Wicaksono, Y. R. Kusumaningsih, and Iswahyudi, c.. Pengujian Celah Keamanan Aplikasi Berbasis Web Menggunakan Teknik Penetration Testing dan Dast (Dynamic Application Security Testing)," *jurnal jarkom*, vol. 8, pp. 1-9, 2020.

[11] J, Susanto, Biqirrosyad, M. M. Junaidi, Y. Sudrajat, Y., and T. Desyani, "Pengujian Black Box pada Aplikasi Desktop Penjualan Elektronik Menggunakan Metode Equivalence Partitioning," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 4, no. 1, pp. 38-45, 2021.

[12] I. G. A. S. Sanjaya, G. M. A. Sasmita, and D. M. S. Arsa, "Evaluasi Keamanan Website Lembaga X Melalui Penetration Testing Menggunakan Framework ISSAF," *Jurnal Ilmiah Merpati*, vol. 8, pp. 113-124, 2020.

[13] M. Nurudin, W. Jayanti, R.D. Saputro, M. P. Saputra, and Yulianti, "Pengujian Black Box pada Aplikasi Penjualan Berbasis Web menggunakan Teknik Boundary Value Analysis," *Jurnal Informatika Universitas Pamulang*, vol. 4, no. 4, pp. 143-148, 2019.

[14] I. A. Shaleh, J. Prayogi, Pirdaus, R. Syawal, and A. Saifudin, "Pengujian Black Box pada Sistem Informasi Penjualan Buku Berbasis Web dengan Teknik Equivalent Partitions," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 4, no. 1, pp. 38-45, 2021.

[15] T. Wahyuningrum and D. D. Januarita, "Implementasi dan Pengujian Web Ecommerce untuk Produk Unggulan Desa," *Jurnal Komputer Terapan*, vol. 1, pp. 57-66, 2015.

[16] F. S. Kristara, G. Kanuraga, R. Rohmat, D. Yansah, A. Saifudin, and Yulianti, "Pengujian Kualitas Aplikasi Web E-Learning Universitas Pamulang Menggunakan Metode Black Box," *Jurnal Informatika Universitas Pamulang*, vol. 6, no. 2, pp. 225-231, 2021.