Development of a Notification-Based Network Security Monitoring System Using Network Development Life Cycle (NDLC)

Ryan Timothy Benget Daulat Butarbutar[a1], **Gusti Made Arya Sasmita**[a2], **I Putu Agus Eka Pratama**[a3]
[a]Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Udayana, Bali, Indonesia
e-mail: [1]ryantimothy12@gmail.com, [2]aryasasmita@unud.ac.id, [3]eka.pratama@unud.ac.id

Abstrak

Perkembangan pada bidang teknologi informasi dan komunikasi yang semakin cepat, dituntut untuk juga meningkatkan kualitas keamanannya. Pemakaian internet yang meningkat, menyebabkan tingginya gangguan atau serangan yang mengeksploitasi kelemahan pada protokol internet, sistem operasi dan software aplikasi. Dikarenakan tingginya serangan, keamanan teknologi informasi merupakan hal yang penting untuk diperhatikan dalam sebuah organisasi maupun individu. Salah satu solusi yang dapat diterapkan untuk mengatasi permasalahan tersebut adalah dengan melakukan monitoring keamanan jaringan. Salah satu penerapan monitoring keamanan jaringan adalah dengan penerapan Intrusion Detection System (IDS). Adapun tujuan dari penelitian ini yaitu untuk merancang sebuah sistem monitoring keamanan jaringan berbasis Snort sebagai alat penerapan IDS untuk mendeteksi serangan dengan mengirimkan notifikasi melalui Bot Telegram yang terintegrasi dengan server yang dimiliki oleh Unit Sumber Daya Informasi Universitas Udayana dengan menggunakan metode dari Network Development Life Cycle (NDLC). Berdasarkan hasil konfigurasi pada penelitian yang dilakukan pada server Unit Sumber Daya Informasi Universitas Udayana yaitu terdapat data terendah pada tanggal 16 Mei 2023 sebanyak 64.418 data dan data tertinggi pada tanggal 04 April 2023 sebanyak 261.672 data serta 2 data serangan Nmap Scan sebagai jenis serangan terendah dan 13.703 data serangan DDoS Attack sebagai jenis serangan tertinggi.

Kata kunci: *Monitoring Keamanan Jaringan, Intrusion Detection System, Snort, Telegram, Network Development Life Cycle*

Abstract

Developments in information and communication technology which are increasing rapidly, it is also required to improve the quality of security. The increase of internet usage has led to high number of intrusions or attacks that exploit weaknesses in internet protocols, operating systems and application software. Due to the high number of attacks, information technology security is an important thing to consider in an organization or individual. One solution that can be applied to overcome this problem is to monitor network security. One of the implementations of network security monitoring is the implementation of an Intrusion Detection System (IDS). The purpose of this research was to design a Snort-based network security monitoring system as a tool for implementing IDS to detect attacks by sending notifications via Telegram Bot which is integrated with the server owned by Unit Sumber Daya Informasi Universitas Udayana using the Network Development Life Cycle (NDLC) method. Based on the configuration results of research conducted on the Unit Sumber Daya Informasi Universitas Udayana server, there was the lowest data on May 16, 2023 with 64,418 data and the highest data on April 04, 2023 with 261,672 data and 2 Nmap Scan attack data as the lowest type of attack and 13,703 DDoS attack data as the highest type of attack.

Keywords : *Network Security Monitoring, Intrusion Detection System, Snort, Telegram, Network Development Life Cycle*

1. Introduction

Developments in the field of information and communication technology are increasing rapidly with various functions that suit needs, indirectly also required to improve the quality of security in every development of its functions. Apart from the development of technology and information. Internet usage is increasing exponentially, which can increase interference or attacks carried out by crackers who exploit weaknesses. When interference can cause a network to experience problems and administrators cannot recover quickly, it will affect the existing network system [20]. Therefore, information technology security is a basic thing that is important to pay attention to in organizational and individual circles [21]. One practical solution that can be applied to overcome this problem is to monitor network security. Monitoring network security is something that is very important and necessary so that the security of the data held can be well maintained. One application of network security monitoring is the implementation of an Intrusion Detection System. The IDS application used in this research is Snort. Snort is an open-source software for implementing the IDS concept which functions to carry out the process of monitoring data traffic by utilizing rules in its configuration. When suspicious access occurs to the system, Snort with the help of its rules will be able to detect and send the information to the network administrator. The Telegram application is used as a medium for sending notifications to the network administrator's smartphone, so that the process of monitoring and controlling the network can be carried out in real-time.

2. Research Method

Development of a Notification-Based Network Security Monitoring System Using Network Development Life Cycle (NDLC) was created using the Network Development Life Cycle method approach. Network Development File Cycle or NDLC is a structured analysis technique used to plan and manage the system development process. The use of the NDLC method can also help in the process of optimizing systems used in agencies [19]. The stages of the NDLC method are explained in detail in the research flow which can be seen in Figure 1.

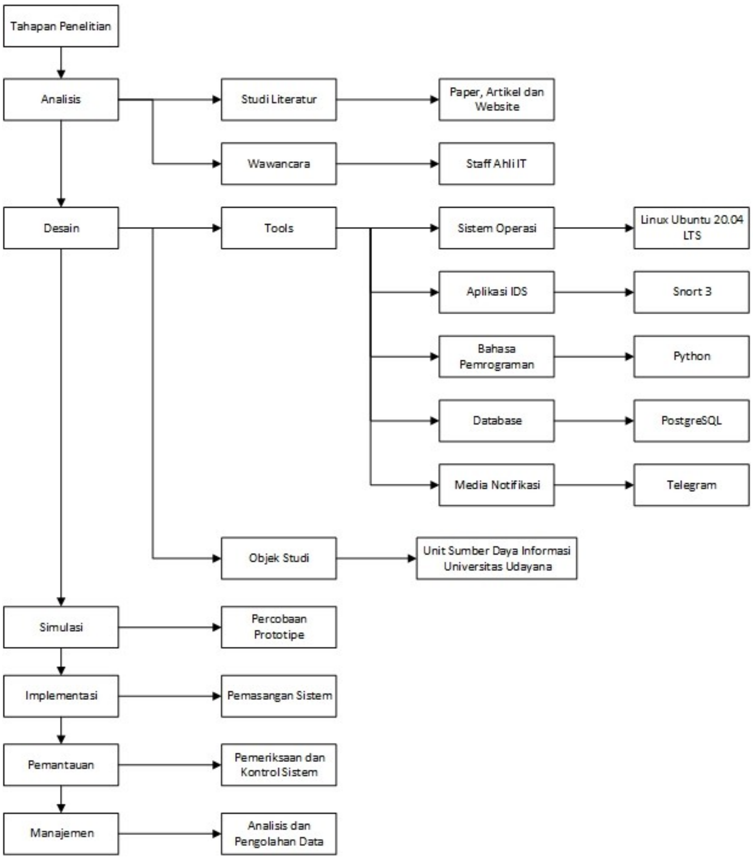


Figure 1. Research Flow

Figure 1 showed the research flow of the development of a notification-based network security monitoring system using the Network Development Life Cycle (NDLC). The NDLC method used in this research has 6 stages, namely Analysis, Design, Simulation, Implementation, Monitoring, and Management. The research began by carrying out an analysis through interviews and literature studies, then carrying out a design related to the network security monitoring system that will be used, then carrying out a system simulation to determine the readiness of the system and implementing or implementing the system directly and then monitoring the data obtained. obtained and then enter the final stage, namely management, where at this stage the data will be processed and then displayed as information. A general overview of the system created can be seen in Figure 2.

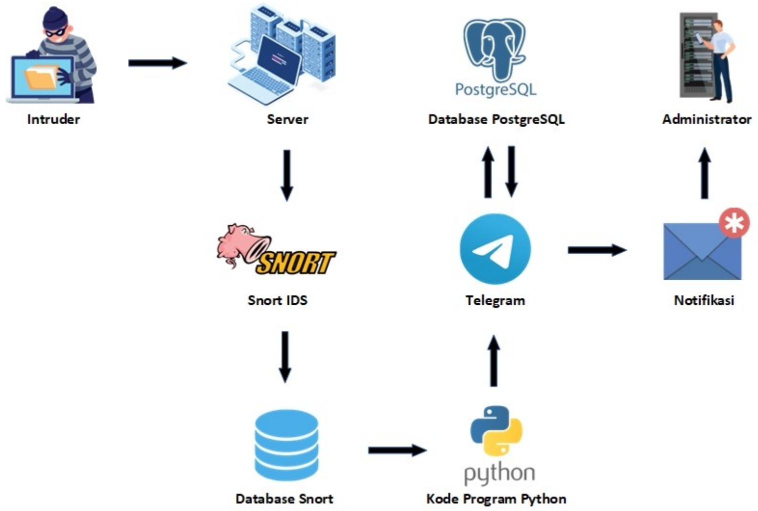


Figure 2. System Overview

Figure 2 showed a general overview of the overall system that was created, namely the system will read all incoming data packets and when there is suspicious activity that enters the network security system, Snort directly as an application for implementing the Intrusion Detection System (IDS) will log the data to the database for later The data in the database is processed and processed into a warning notification, then the warning notification is sent to the network administrator. The flowchart of the system created can be seen in Figure 3.

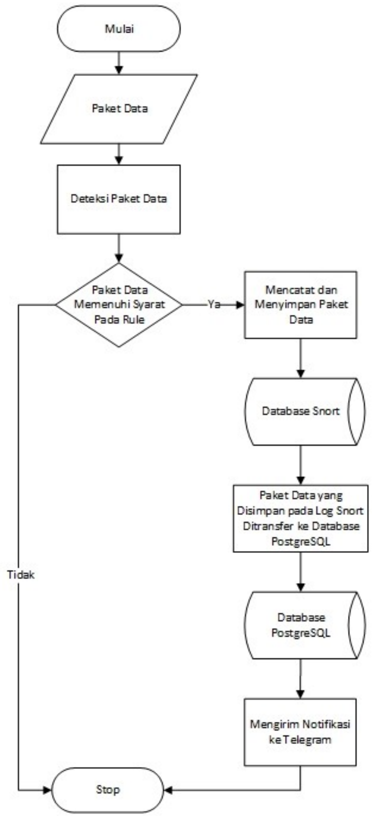

Mulai
Paket Data
Deteksi Paket Data
Paket Data Memenuhi Syarat Pada Rule
Ya
Tidak
Mencatat dan Menyimpan Paket Data
Database Snort
Paket Data yang Disimpan pada Log Snort Ditransfer ke Database PostgreSQL
Database PostgreSQL
Mengirim Notifikasi ke Telegram
Stop

Figure 3. System Flowchart

Figure 3 showed a flowchart image of the system implemented in research on the Development of a Notification-Based Network Security Monitoring System Using the Network Development Life Cycle (NDLC). The system will first capture incoming data packets and then the data packets are detected using the help of Snort. Data packets that are detected as packets that have the characteristics of intrusion packets that comply with the rules set by Snort will be recorded and then saved as log files in Snort storage. Data packets that do not have the characteristics of an intrusion packet will go through the analysis process, so they will not be recorded and stored in the Snort log file. The log files that have been recorded and stored in the Snort database will then be sent to the PostgreSQL database for data processing. The data will be processed and then become information for the administrator through notifications sent using Telegram.

3. Literature Study

The internet network is a computer system that has the ability to exchange data packets which functions to serve several users using the global system of Transmission Control Protocol or TCP/IP as the main standard for data exchange [16]. In the process of data communication or data transfer between users and two different devices in a network consisting of many other devices, a specific destination address is needed so that the internet uses protocols to guarantee that the data exchange process reaches the destination address safely.

Network Development Life Cycle (NDLC) comes from Systems Development Life Cycle (SDLC) which is a structured analysis technique used to plan and manage the system development process. The use of the NDLC method can also help in optimizing computer network systems used in agencies [19].

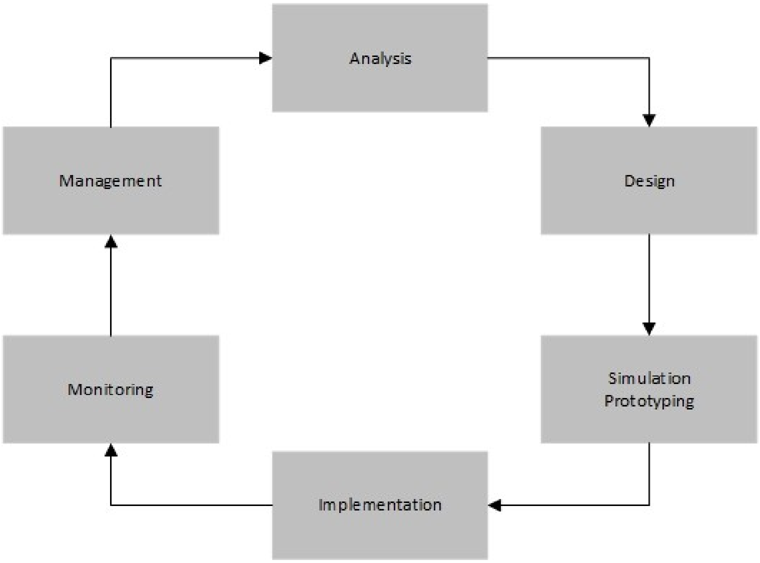


Figure 4. Network Development Life Cycle

Figure 4 showed the stages of the Network Development Life Cycle (NDLC) method. The application of the NDLC method is divided into 6 stages, namely the analysis stage, design stage, simulation stage, implementation stage, monitoring stage, and management stage. The analysis stage is an initial stage carried out to determine user needs, analysis of existing problems, and analysis related to the use of network topology. The design stage is carried out to provide an overview of the system design that will be developed based on the previous system or topology, which already includes the desired needs. The simulation stage is carried out to show the initial performance of the network system to be built. The implementation phase is carried out to implement the network system that has been previously planned and designed. At this stage, changes will be seen from the previous system to a new system that has been developed. The monitoring stage is carried out to monitor the state of the network system that has been previously built. The management stage is carried out to pay attention to details related to the policies that will be created to regulate the system that has been built so that it can run well and have good maintenance.

In a computer network infrastructure, network security is a very important pillar [2]. The purpose of network security is to monitor network access and prevent misuse of network infrastructure. In a network infrastructure, secure data is very important. For good network security, there are three requirements for network infrastructure to be declared safe and secure. There are prevention, observation, and response. These three conditions are very necessary in a network so that the network remains safe.

Network monitoring is the process of collecting and analyzing the amount of data that passes through a network so that the state of connectivity between devices on a network can be known with the aim of maximizing all the resources on a computer network [13]. Network monitoring is an activity carried out to manage a network system in a certain location with a certain network topology. The network monitoring system is used to make it easier for the technical team to carry out routine maintenance and monitoring of network conditions in the field [11].

Port Mirroring or Switched Port Analyzer (SPAN) is a method of monitoring network traffic. The working system of Port Mirroring is by copying all traffic data that passes through a certain port that has been previously configured for copying. Every time the Switch processes a packet, the port makes a copy and sends a copy of the packet. The use of Port Mirroring is a special system prepared to monitor traffic on a port. At Cisco, the Switched Port Analyzer (SPAN) is a special method that can be used to perform port mirroring [10].

Intrusion Detection System or IDS is a system that can detect suspicious activity on a system or network. If suspicious activity is found in network traffic, the IDS will provide a warning to the system or network administrator carry out analysis and look for evidence of attempted intrusion. The following are several types of IDS [1]. First, the Network Intrusion

Detection System or NIDS. Second, Host Intrusion Detection System or HIDS. Third, anomaly-based IDS

Snort is an open-source network Intrusion Detection System that is capable of carrying out real-time analysis and packet logging on IP networks. Snort can perform protocol analysis, content searching or matching, and can be used to detect various attacks and intrusions. Snort is software for detecting intruders and analyzing packets that traverse computer networks in real-time traffic and logging into a database and can detect various attacks originating from outside the network [24].

Python is an open-source programming language maintained by the Python Software Foundation, a non-profit company that holds the intellectual property rights to Python. The Python programming language can be used to create applications, computer commands, and perform data analysis. As a general-purpose language, Python can be used to create various types of programs and solve various problems because Python can be downloaded for free, and is very well integrated with all types of operating systems so that it can increase the speed of system development well [9].

Sending messages using the Telegram messenger application is a real-time communication method. In the Telegram messenger application, there is a special feature provided, namely a chatbot. Chatbot is a computer program designed to simulate a conversation or interactive communication with users either through text, voice or visuals. Chatbots act as conversational agents that can help or replace the role of consultants. Chatbots have a knowledge base that can be used to have conversations with users [8].

PostgreSQL is a powerful open-source object-relational database system that uses and extends the SQL language combined with many features that securely store and scale the most complex data workloads with the help of pgAdmin as a tool that can be used as a database view [25]. The origins of PostgreSQL began in 1986 as part of the POSTGRES project at the University of California at Berkeley and has more than 30 years of active development on the core platform. PostgreSQL also has the ability to support standard SQL commands so that users or developers can use them according to their needs, both individually and in groups.

4. Result and Discussion

4.1. System Planning

The system design stage is the process where the system structure design is carried out. The system was created with the help of several tools that were installed and then configured for each tool so that it could be connected. The process of implementing the design in research was carried out in several stages consisting of Snort prerequisites installation, Snort installation, Snort NIDS configuration, Snort rules configuration, data log configuration, Python prerequisites installation, Python installation, PostgreSQL installation, PostgreSQL configuration, and Telegram Bot configuration.

The Snort prerequisites installation stage is a necessary stage to support the installation process of Snort so that it can be run on the Linux Ubuntu 20.04 LTS operating system which is used as the place where Snort runs. The Ubuntu Linux operating system must be supported by the latest repositories to be able to install the prerequisites of Snort.

The Snort installation stage is the stage in the Snort installation process. Snort installation is done by downloading the DAQ or Data Acquisition file with the help of git for the file download process and then the downloaded file will be installed. After the Snort installation process is complete, Snort can be seen to be running well by using the snort-v command on the Ubuntu Linux Command Line that is being used.

The Snort NIDS configuration stage is the stage of the configuration process for the Snort file that was previously installed. The important configuration that needs to be done is changing several components in the configuration file called snort.lua so that the traffic data capture can be in accordance with the design.

The Snort rules configuration stage is the stage of the process of determining several rules in Snort. Snort NIDS will run according to the rules that have been created. Based on existing rules, Snort will capture traffic data that is similar to the rule provisions that have been created and stored in the Snort file.

The data log configuration stage is the next stage of the process which is carried out after the process of determining several rules to be used. This process will create new data output that can be adjusted according to your wishes. In this research, the output used is the alert_json output.

The Python prerequisites installation stage is a necessary stage to support the Python installation process so that it can be run on the Ubuntu 20.04 LTS Linux operating system which is used as the place where the Python system runs. The Ubuntu Linux operating system must be supported by the latest repositories to be able to perform the prerequisite installation of Python.

The Python installation stage is the stage in the Python installation process. Python installation is done by downloading the Python version 3.8 file and also installing several other libraries such as PIP, pyTelegramBotAPI, and Psycopg2 for use in the monitoring system.

The PostgreSQL installation stage is the stage required to store data in the database. After the installation process is complete, you can execute the psql –version command to validate that the PostgreSQL database has been installed with the appropriate version.

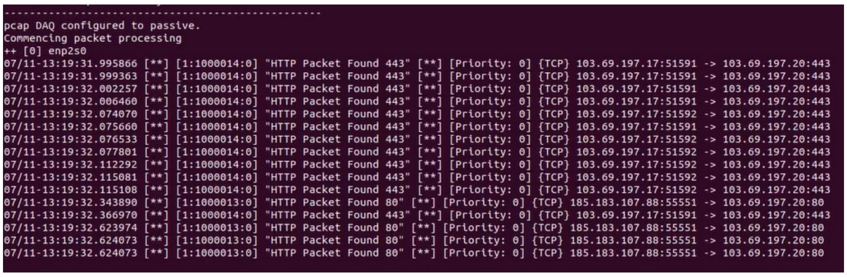
The PostgreSQL configuration stage is the stage for preparing the database to be used with the data structure taken from the Snort log. Data that has been successfully saved in the Snort database log will then be processed again and then stored in the PostgreSQL database for the process of sending information in the form of notifications to Telegram.

The Telegram Bot configuration stage is the stage for displaying data as information through Telegram notifications sent via the Telegram Bot to the administrator. Telegram Bots can be used after registering a Bot account and then getting a username and token which will function as access to the Telegram Bot which is used using Python program code.

4.2. System Testing

The system testing stage is a process where the results of the system structure design are tested and then implemented directly. The system was created with the help of several tools that were installed and then configured for each tool so that it could be connected. The process of implementing the design in research was carried out in several stages consisting of Snort testing, Telegram bot notification testing and analysis of attack log data.

The Snort testing stage is a testing process to find out that the Snort tool used as a monitoring system can run well for the data capture process. The following is a display of the Snort tool when carrying out the monitoring process on the network interface which can be seen in Figure 5.

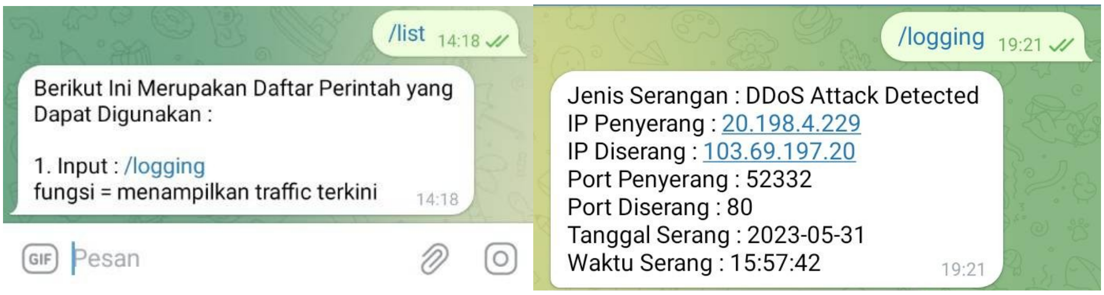


```
--------------------------------------------------
pcap DAQ configured to passive.
Commencing packet processing
++ [0] enp2s0
07/11-13:19:31.995866 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51591 -> 103.69.197.20:443
07/11-13:19:31.999363 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51591 -> 103.69.197.20:443
07/11-13:19:32.002257 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51591 -> 103.69.197.20:443
07/11-13:19:32.006460 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51591 -> 103.69.197.20:443
07/11-13:19:32.074070 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51592 -> 103.69.197.20:443
07/11-13:19:32.075660 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51591 -> 103.69.197.20:443
07/11-13:19:32.076533 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51592 -> 103.69.197.20:443
07/11-13:19:32.077801 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51592 -> 103.69.197.20:443
07/11-13:19:32.112292 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51592 -> 103.69.197.20:443
07/11-13:19:32.115081 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51592 -> 103.69.197.20:443
07/11-13:19:32.115108 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51592 -> 103.69.197.20:443
07/11-13:19:32.343890 [**] [1:1000013:0] "HTTP Packet Found 80" [**] [Priority: 0] {TCP} 185.183.107.88:55551 -> 103.69.197.20:80
07/11-13:19:32.366970 [**] [1:1000014:0] "HTTP Packet Found 443" [**] [Priority: 0] {TCP} 103.69.197.17:51591 -> 103.69.197.20:443
07/11-13:19:32.623974 [**] [1:1000013:0] "HTTP Packet Found 80" [**] [Priority: 0] {TCP} 185.183.107.88:55551 -> 103.69.197.20:80
07/11-13:19:32.624073 [**] [1:1000013:0] "HTTP Packet Found 80" [**] [Priority: 0] {TCP} 185.183.107.88:55551 -> 103.69.197.20:80
07/11-13:19:32.624073 [**] [1:1000013:0] "HTTP Packet Found 80" [**] [Priority: 0] {TCP} 185.183.107.88:55551 -> 103.69.197.20:80
```

Figure 5. Snort Testing

Figure 5 showed a display of the Snort tool when monitoring traffic data on the network interface used. The monitoring process is carried out by entering or executing commands on the Command Line of Linux Ubuntu as the operating system used.

The Telegram Bot notification testing stage is a testing process to find out that the monitoring system that has been created is capable of sending notifications and can run well for the data capture process. The following is a display of several commands that the Telegram Bot has when carrying out the monitoring process on the network interface which can be seen in Figure 6.



/list 14:18

Berikut Ini Merupakan Daftar Perintah yang Dapat Digunakan :

1. Input : /logging
fungsi = menampilkan traffic terkini 14:18

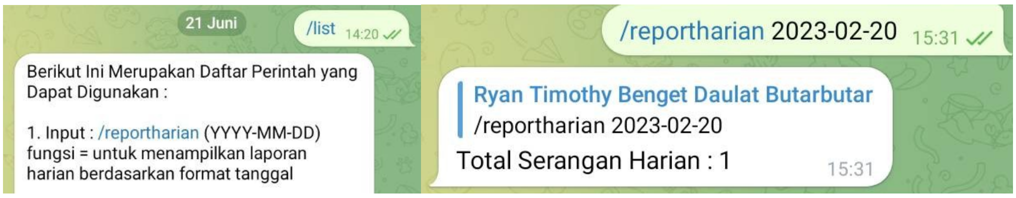
/logging 19:21

Jenis Serangan : DDoS Attack Detected
IP Penyerang : 20.198.4.229
IP Diserang : 103.69.197.20
Port Penyerang : 52332
Port Diserang : 80
Tanggal Serang : 2023-05-31
Waktu Serang : 15:57:42 19:21

(a) (b)

Figure 6. (a) List Bot Monitoring (b) Logging

Figure 6 showed a display of several commands that the Telegram Bot has when carrying out the monitoring process on the network interface. The monitoring process can be carried out by entering a command. To find out the list of commands that can be used, the administrator can enter the initial command, namely /list, which functions to display a list of commands on the Telegram bot being used. The monitoring process can be carried out by entering the command /logging which will then be followed by entering a notification of the monitoring results and simultaneously logging into the database. The following is the appearance of the Telegram Bot when carrying out the reporting process on the network interface which can be seen in Figure 7.



/list 14:20

Berikut Ini Merupakan Daftar Perintah yang Dapat Digunakan :

1. Input : /reportharian (YYYY-MM-DD)
fungsi = untuk menampilkan laporan harian berdasarkan format tanggal

/reportharian 2023-02-20 15:31

Ryan Timothy Benget Daulat Butarbutar
/reportharian 2023-02-20
Total Serangan Harian : 1 15:31

(a) (b)

Figure 7. (a) List Bot Reporting (b) Daily Report

Figure 7 showed a display of several commands that the Telegram Bot has when carrying out the process of reporting traffic data on the network interface. The reporting process can be carried out by entering several commands. To find out the list of commands that can be used, the administrator can enter the initial command, namely /list, which functions to display a list of commands on the Telegram bot being used. The daily reporting process can be done by entering the command /reportharian followed by entering the desired date and then this will be followed by entering a notification of the reporting results whose data is taken from the database.

The attack log data analysis stage is a data analysis process using data that has been obtained to find out the attack log data in more detail using graphical assistance to display the results of data analysis. The following is a graphic display of the process of capturing attack data which is divided by date which can be seen in Figure 8.

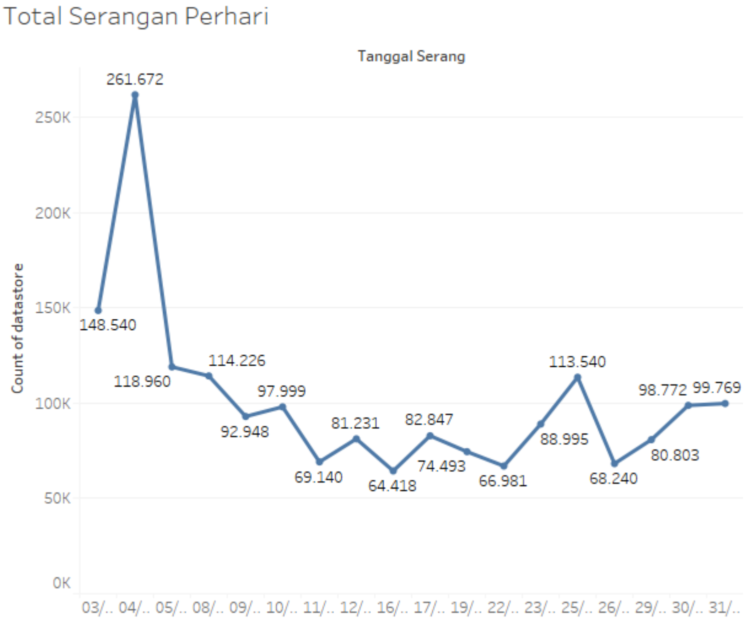


Figure 8. Graphic Display of Total Attacks Per Day

Figure 8 showed a graphic display of the process of capturing attack data which is divided by date. Based on the graphic display, it can be concluded that the lowest data captured by the system occurred on the 16th with a total of 64,418 attack data and the highest data captured by the system occurred on the 4th with a total of 261,672 attack data. The following is a graphic display of the process of capturing attack data which is divided based on the type of attack which can be seen in Figure 9.

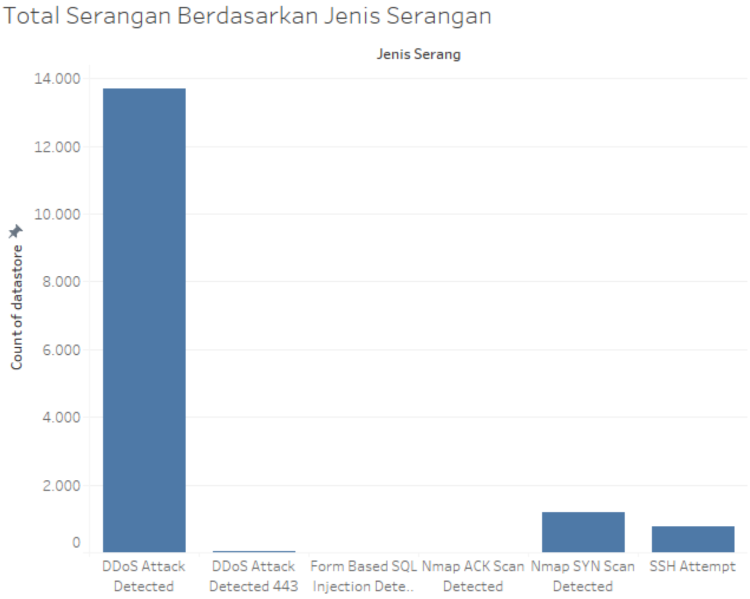


Figure 9. Graphic Display of Total Attacks Based on Attack Type

Figure 9 showed a graphic display of the process of capturing attack data which is divided based on the type of attack. Based on the graphic display, it can be concluded that the lowest data captured by the system is Nmap ACK Scan Detected with a total of 2 attack data and the highest data captured by the system is DDoS Attack Detected with a total of 13,703 attack data. The following is a graphic display of the attack data capture process which has been divided based on the attacker's IP which can be seen in Figure 10.

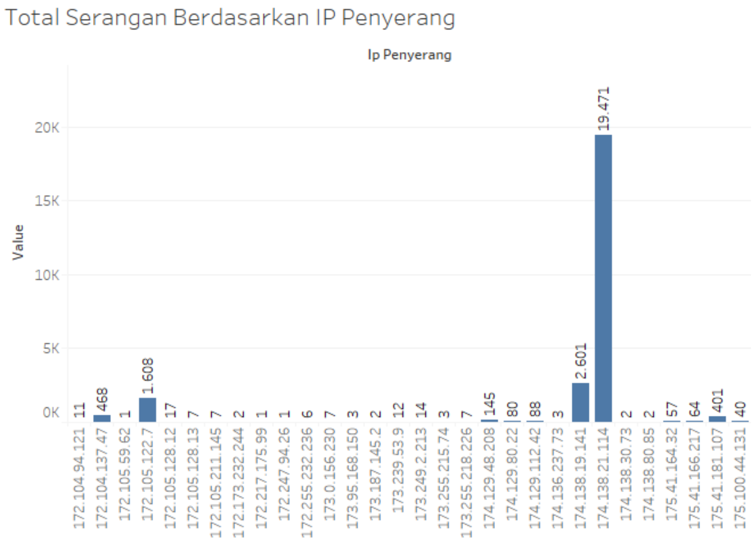


Figure 10. Graphic Display of Total Attacks Based on Attacker IP

Figure 10 showed a graphic display of the process of capturing attack data which is divided based on the attacker's IP. Based on the graphic display, it can be concluded that the lowest data that was successfully captured by the system was ip 172.105.59.62, 172.217.175.99, and 172.247.94.26 with a total of 1 attack data each and the highest data that was successfully captured by the system was ip 174.138.21.114 with a total of as many as 19,471 attack data. The following is a graphic display of the process of capturing attack data which has been divided based on the IP attacked which can be seen in Figure 11.

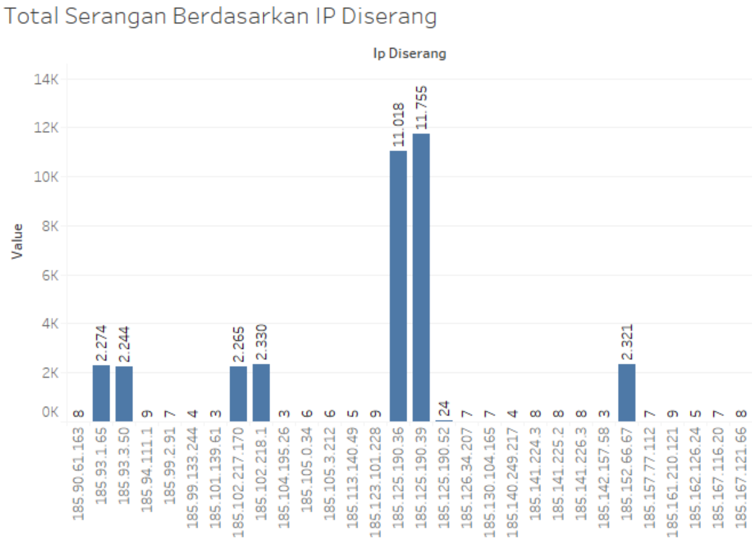


Figure 11. Graphic Display of Total Attacks Based on IP Attacked

Figure 11 showed a graphic display of the process of capturing attack data which is divided based on the IP being attacked. Based on the graphic display, it can be concluded that the lowest data that was successfully captured by the system were IP 185.101.139.61, 185.104.195.26, and 185.142.157.58 with a total of 3 attack data each and the highest data that was successfully captured by the system was IP 185.125.190.39 with a total of 11,755 attack data.

5. Conclusion

Based on the research that has been carried out, there are conclusions that can be drawn from research on the Development of a Notification-Based Network Security Monitoring System Using the Network Development Life Cycle (NDLC).

1. The Snort-based network security monitoring system using the Network Development Life Cycle with the application of the Intrusion Detection System concept is capable of being integrated with the Udayana University Information Resources Unit server.
2. The network security monitoring system is capable of detecting attacks based on the results of the configuration that has been carried out on the Udayana University Information Resources Unit server with a total of 1,848,423 data captured with details, namely the lowest total data occurred on May 16 2023 with the total was 64,418 data and the highest total data occurred on April 4 2023 with 261,672 data and 2 Nmap Scan attack data as the lowest type of attack and 13,703 DDoS Attack data as the highest type of attack.
3. Telegram Bot as a messenger application can create attack warning notifications based on the results of the configuration that has been carried out.

References

[1] Alamsyah, H., -, R., & Al Akbar, A. (2020). Analisa Keamanan Jaringan Menggunakan Network Intrusion Detection and Prevention System. JOINTECS (Journal of Information Technology and Computer Science), 5(1), 17. https://doi.org/10.31328/jointecs.v5i1.1240

[2] Amarudin, A. (2018). Desain Keamanan Jaringan Pada Mikrotik Router OS Menggunakan Metode Port Knocking. Jurnal Teknoinfo, 12(2), 72. https://doi.org/10.33365/jti.v12i2.121

[3] Dar, M. H., & Harahap, S. Z. (2018). Implementasi Snort Intrusion Detection System (Ids) Pada Sistem Jaringan Komputer. Jurnal Informatika, 6(3), 14–23. https://doi.org/10.36987/informatika.v6i3.1619

[4] Dasmen, R. N., Ariyanto, C., Surya, M. H., & Ramadhan, H. (2022). Penerapan Snort Sebagai Sistem Pendeteksi Serangan Keamanan Jaringan. Jurasik (Jurnal Riset Sistem Informasi Dan Teknik Informatika), 7(1), 8. https://doi.org/10.30645/jurasik.v7i1.409

[5] Dewi Paramitha, I. A. S., Sasmita, G. M. A., & Raharja, I. M. S. (2020). Analisis Data Log IDS Snort dengan Algoritma Clustering Fuzzy C-Means. Majalah Ilmiah Teknologi Elektro, 19(1), 95. https://doi.org/10.24843/mite.2020.v19i01.p14

[6] Fernando, N., Humaira, & Asri, E. (2020). Monitoring Jaringan dan Notifikasi dengan Telegram pada Dinas Komunikasi dan Informatika Kota Padang. JITSI : Jurnal Ilmiah Teknologi Sistem Informasi, 1(4), 121–126. https://doi.org/10.30630/jitsi.1.4.17

[7] Gunawan, A. R., Sastra, N. P., & Wiharta, D. M. (2021). Penerapan Keamanan Jaringan Menggunakan Sistem Snort dan Honeypot Sebagai Pendeteksi dan Pencegah Malware. Majalah Ilmiah Teknologi Elektro, 20(1), 81. https://doi.org/10.24843/mite.2021.v20i01.p09

[8] Harahap, D. W., & Fitria, L. (2020). Aplikasi Chatbot Berbasis Web Menggunakan Metode Dialogflow. 01(01), 1–7.

[9] Hardjianto, M. (2022). Sistem Monitoring Serangan Ssh Dengan Metode Intrusion Prevention System (IPS) Fail2ban Menggunakan Python Pada Sistem Operasi Linux. Jurnal TICOM: Technology of Information and Communication, 11(1), 33–38.

[10] Kassim, M., Mahmud, A. R., Amirullah Ramli, M., & Rahman, R. A. (2022). Network Analysis of Students' Online Activities via Port mirroring Switch Port Analyzer. 2022 12th IEEE Symposium on Computer Applications and Industrial Electronics, ISCAIE 2022, 49–54. https://doi.org/10.1109/ISCAIE54458.2022.9794504

[11] Kukuh Prayogi, P., Orisa, M., & Ariwibisono, F. (2020). Rancang Bangun Sistem Monitoring Jaringan Access Point Menggunakan Simple Network Management Protocol (Snmp) Berbasis Web. JATI (Jurnal Mahasiswa Teknik Informatika), 4(1), 192–197. https://doi.org/10.36040/jati.v4i1.2327

[12] Kusuma, D., Darussalam, U., & Hidayatullah, D. (2020). Implementasi Monitoring Jaringan Melalui Aplikasi Sosial Media Telegram Dengan Snort. J I M P - Jurnal Informatika Merdeka Pasuruan, 5(1), 6–9. https://doi.org/10.37438/jimp.v5i1.242

[13] Miftah, Z. (2019). Penerapan Sistem Monitoring Jaringan Dengan Protokol SNMP Pada Router Mikrotik dan Aplikasi Dude Studi Kasus Stikom CKI. Faktor Exacta, 12(1), 58–66. https://doi.org/10.30998/faktorexacta.v12i1.3481

[14] Naim, F., Saedudin, R. R., & Hediyanto, U. Y. K. S. (2022). Analysis of Wireless and Cable Network Quality-of-Service Performance At Telkom University Landmark Tower Using Network Development Life Cycle (Ndlc) Method. JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika), 7(4), 1033–1044. https://doi.org/10.29100/jipi.v7i4.3192

[15] Nuryadi, N., & Nainggolan, E. C. (2021). Implementasi Intrusion Detection System Pada Local Area Network (Studi Kasus : Yayasan Pendidikan Tanah Tingal Tangerang). Jurnal Sains, Teknologi Dan Industri, 19(1), 1–8.

[16] Pratama, I. P. A. E., & Dharmesta, P. A. (2019). Implementasi Wireshark Dalam Melakukan Pemantauan Protocol Jaringan (Studi Kasus : Intranet Jurusan Teknologi Informasi Universitas Udayana). Mantik Penusa, 3(1), 94–99.

[17] Pratama, I. P. A. E., & Handayani, N. K. M. (2019). Implementasi Ids Menggunakan Snort Pada Sistem Operasi Ubuntu. Jurnal Mantik Penusa, 3(1), 176–181.

[18] Sagita, R., Sagita, R., Puspasari, R., Teknik, J., Universitas, I., Utama, P., Jurusan, D., Informatika, T., Potensi, U., Direct, W., & Android, S. (2020). Perancangan Aplikasi Messenger Peer to Peer Berbasis Wifi Direct Pada Smartphone Android. Jurnal FTIK, 307–314.

[19] Sanjaya, T., & Setiyadi, D. (2019). Network Development Life Cycle (NDLC) Dalam Perancangan Jaringan Komputer Pada Rumah Shalom Mahanaim. Mahasiswa Bina Insani, 4(1), 1–10. http://ejournal-binainsani.ac.id/

[20] Sudradjat, B. (2017). Sistem Pendeteksian dan Pencegahan Penyusup Pada Jaringan Komputer Dengan Menggunakan Snort dan Firewall. JISAMAR (Journal of Information System, Applied, Management, Accounting and Research), 1(1), 10–24.

[21] Suhartono. (2017). Sistem Pengamanan Jaringan Admin Server Dengan Metode Intrusion Detection System (Ids) Snort. Jurnal Scientific Pinisi, 3(April), 60–64.

[22] Sulistya, I. M. A., & Sasmita, G. M. A. (2020). Network Security Monitoring System on Snort with Bot Telegram as a Notification. International Journal of Computer Applications Technology and Research, 9(2), 059–064. https://doi.org/10.7753/ijcatr0902.1004

[23] Susanto, B. M., & Guritno, A. T. (2017). Implementasi Snort Ids Menggunakan Android Sebagai Media Notifikasi. Seminar Nasional Teknologi Informasi Dan Komunikasi 2017, 2017(Sentika), 203–212.

[24] Sutarti, Pancaro, A. P., & Saputra, F. I. (2018). Implementasi IDS (Intrusion Detection System) Pada Sistem Keamanan Jaringan SMAN 1 Cikeusal. Jurnal PROSISKO, 5(1), 1–8.

[25] Wibowo Putri, A. A., & Susetyo, Y. A. (2022). Implementation of Flask for Stock checking in Distribution Center & Store on Monitoring Stock Application in Pt. Xyz. Jurnal Teknik Informatika (Jutif), 3(5), 1265–1274. https://doi.org/10.20884/1.jutif.2022.3.5.334

[26] Widiyanto, W. W. (2022). SIMRS Network Security Simulation Using Snort IDS and IPS Methods. Indonesian of Health Information Management Journal (INOHIM), 10(1), 10–17. https://doi.org/10.47007/inohim.v10i1.396

[27] Yanto, H. (2020). Intruder Detection Monitoring System in Computer Networks Using Snort Based Sms Alert (Sistem Monitoring Deteksi Penyusup Dalam Jaringan Komputer Menggunakan Snort Berbasis Sms Alert). 7(2), 159–170.

[28] Zaen, M. T. A., Tantoni, A., & Ashari, M. (2021). DDoS ATTACK MITIGATION WITH INTRUSION DETECTION SYSTEM (IDS) USING TELEGRAM BOTS. JISA(Jurnal Informatika Dan Sains), 4(2), 149–154. https://doi.org/10.31326/jisa.v4i2.1043

[29] Pennulis1 A, Penulis2 B, Judul tulisan,*nama jurnal, tahun, Volume, halaman yang diacu*.

[30] Sudoku D, Jawas G, Tambau K, Implementation of a Direct Access Files,*IEEE Transactions on Information*. 2008; 10(4): 70-77. (*untuk kasus Vol.10, Issues 4, and page 70-77*)