ISSN (Online) 3134-884X

Advanced Research Journal of Computer Science

Received: January 24, 2025 | Accepted: May 26, 2025 | Published: June 30, 2025
Volume 01, Issue 01, Pages 14-22

DOI https://doi.org/10.66590/arjcs.2025.01.01.03

OPEN ACCESS

Check for updates

Enhancing Network Security Using NIPS and Honeypot with Telegram-Based Alert System: A Case Study of Diskominfo Serang City

Yudiansywh Fauzi[1*], Fauzi Rustam[2] and D.M. Sueko[3]

[1]Cybersecurity Engineering Study Program, Politeknik Piksi Input Serang, Serang, Jln. Raya Cilegon No.km.8, Kota Serang, 42161, Indonesia
[2]Information Systems Study Program, STMIK Dian Cipta Cendekia, Lampung, Jln. Cut Nyak Dien No.65 Durian Payung Palapa, Bandar Lampung, 52142, Indonesia
[3]Information Systems Study Program, Politeknik Piksi Input Serang, Serang, Jln. Raya Cilegon No.km.8, Kota Serang, 42161, Indonesia

Author Designation: [1,2]Researcher, [2]Lecturer

*Corresponding author: Yudiansywh Fauzi (e-mail: yudiansyahfassuzi@mail.com).

How to Cite the Article:
Fauzi, Yudiansywh, *et al.* " Enhancing Network Security Using NIPS and Honeypot with Telegram-Based Alert System: A Case Study of Diskominfo Serang City." Advanced Research Journal of Computer Science, vol. 01, no. 01, 2025, pp. 14-22. https://doi.org/10.66590/arjcs.2025.01.01.03

Abstract | The increasing number of external cyber threats faced by the Serang City Communication and Information Technology Office is not proportional to the availability of network security infrastructure. Since its establishment in 2017, the office has relied on only one network security device, FortiGate, to protect its computer network. To address this limitation, this study proposes the implementation of a Network Intrusion Prevention System (NIPS) capable of detecting and blocking external attacks in real time. In addition, a high-interaction honeypot model was selected to observe attacker movement and behavior, particularly under high external traffic conditions. To ensure timely delivery of security updates and system development information to users, this research integrates Telegram Bot API, enabling automated notifications and monitoring through the Telegram application. The questionnaire-based beta testing results showed positive user responses, indicating the practicality and usefulness of the proposed system. Furthermore, alpha testing and implementation reports demonstrated that the NIPS and honeypot systems successfully operated across multiple test scenarios. Overall, the implementation of a mobile monitoring system combined with NIPS and honeypot deployment in the Serang City Diskominfo network can serve as an effective alternative network security solution, while also providing valuable information regarding attackers' behavior and intrusion patterns.

Key Words Network Intrusion Prevention System (NIPS), Honeypot, Telegram

INTRODUCTION

Cybersecurity company Kaspersky noted that Indonesia faced more than 11 million cyber-attacks in the first quarter of 2022. From January to March 2022, their products detected and blocked 11,802,558 cyber threats. This number is an increase of 22 per cent compared to the same period last year, with 9,639,740 cyber-attacks in 2021. However, the number of cyber-attacks in the first quarter of 2022 decreased by 2 per cent compared to the fourth quarter (October-December) of 2021. The results of the Kaspersky Lab report for the last quarter of 2018 in Indonesia, 28 percent of computer users were exposed to web-based attacks and more than half or 53.7 percent, were targeted by local threats such as infected Universal Serial Bus (USB) devices [1]. Based on these statistics, Kaspersky noted that Indonesia is at the top of the list in the Southeast Asia region and 60th in the world regarding the dangers posed by surfing the internet [2].

The Department of Communication and Information (DISKOMINFO) Serang City is an agency that has the main task of carrying out government affairs in the fields of communication and informatics, coding and statistics based on the principle of autonomy and assistance tasks based on the vision, mission and program of the Mayor as described in the Regional Medium-Term Development Plan in particular. In the city of attack [3]. Currently, the Diskominfo of Serang City is also not immune from the threat of cyber-attacks by various countries. This is based

on the FortiGate Firewall Log belonging to the Diskominfo of Serang City, recorded from 07 February 2020 to 13 February 2020, detected as many as 2,710 attacks consisting of each level including (severity = low; 1968, severity = moderate; 203, severity = high; 169, severity = critical; 367), where the most attacks came from the United States (United States) and the United Kingdom (United Kingdom), followed by China and Russia [4].

Based on this, the researcher is interested in conducting research titled "Deterrent to Hacker Attacks on Computer Network Infrastructure Using Network Intrusion Prevention System (NIPS) and Integrated Honeypot Notifications through Telegram Applications" [5]. This research aims to add security to the computer network at Diskominfo Serang City so that it is not easily exposed to cyber-attacks from anywhere, either outside or local. That is by using the Network Intrusion Prevention System (NIPS) and Honeypot as a fake server, as well as Telegram, to receive information or cyber-attack notifications in real-time [6].

Literature Review

Proposed Network Topology

The Diskominfo of Serang City has entrusted its computer network security issues to the FortiGate and Pi-Hole Raspberry firewall devices as web-filtering (Figure 1).

There are allegations of security problems, namely the lack of security features because the license for the Diskominfo firewall device in Serang City has not been extended until now. In terms of network security, the researcher proposes additional devices in the form of 2 (two) server units as extra security for external attacks, namely [7-12]:

- Server Network Intrusion Prevention System (NIPS) and Honeypot: Servers designed to detect and stop attacks or intrusions from outside based on established and agreed rules can find out what behaviour the attacker is doing when logging into the server [11]
- Metasploit Servers: This server has many loopholes, usually called Metasploit, that are designed to be the target of attacks from outside. It contains a web server that can be accessed from outside with the name *'Penelitian.serangkota.go.id'* with IP Address 103.102.250.246 (Figure 2) [13-15]

System Workflow

The system workflow in this study is to have a two-stage network security process to secure the server, the first using the Network Intrusion Prevention System (NIPS) security system, namely Snort and the second using the Fake Server, namely Honeypot. In the first stage, if an attacker tries to attack the server, the Network Intrusion Prevention System (NIPS) system in this case, will automatically record data about the attack into the MySQL database and create a log on the system. And will immediately disconnect (drop) the connection from the attacker. The system will warn by sending a message to the network admin via notification via the telegram application using the python programming language [16].

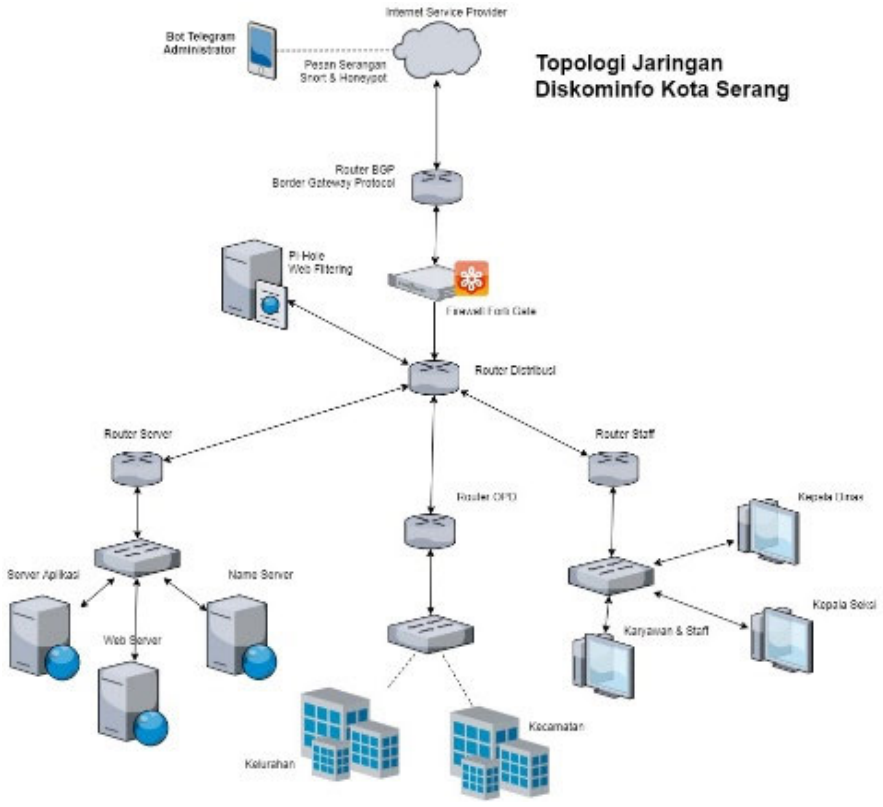

Figure 1: Diskominfo Network Topology Serang City

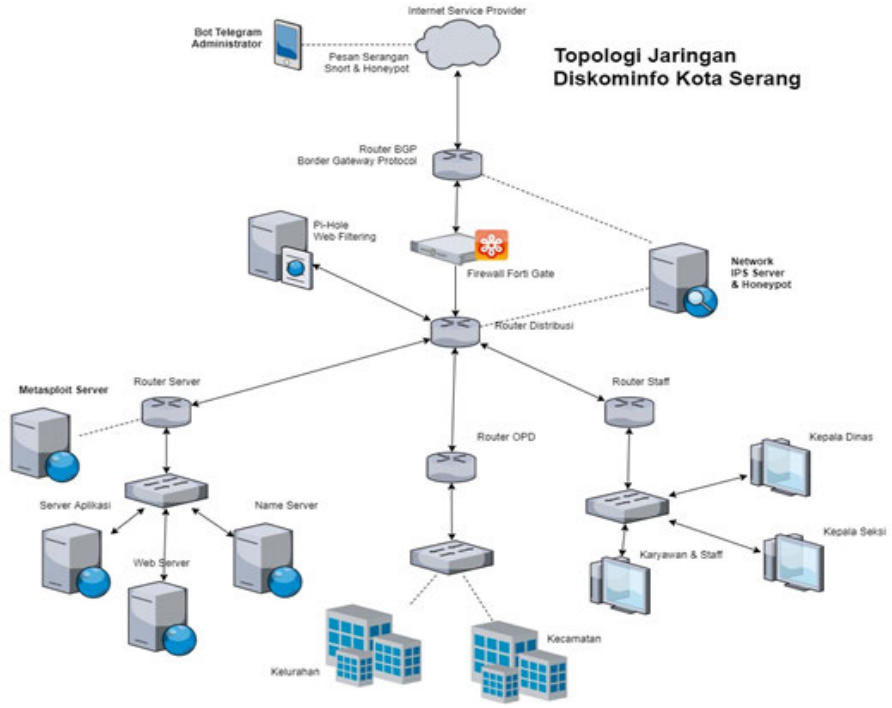

Figure 2: Proposed Network Topology for Serang City Communication and Informatics

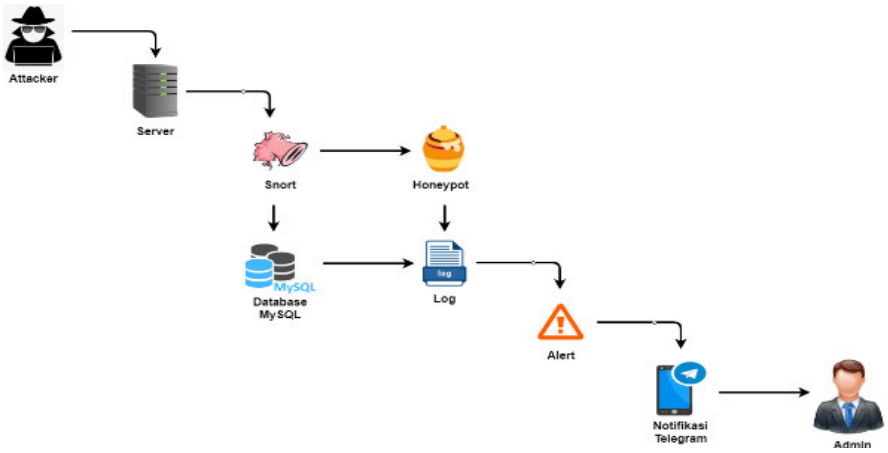

Figure 3: System Workflow

Then the second stage is port 22 or commonly known as Secure Shell (SSH), intentionally opened but will be directed to port 2222 to the SSH Honeypot port; it aims to get a track record of intruders or attackers, all of which will enter the database and will be sent via the telegram application using the python programming language (Figure 3).

Prior Research

The following are some studies related to research conducted by several researchers. Research by Pradipta and Asmunin [1] explained that the network security system is fundamental in maintaining a network. Attacks that can disrupt and even damage the connection system between connected devices will be very detrimental. This is often a consideration in implementing a network security system. Network Intrusion Prevention System (NIPS) can detect attacks and drop attacks. Implementing the Linux operating system using Snort in inline mode can prevent attacks that can threaten.

Agustino *et al*. [2] researched to build Honeypot systems on Cloud Computing services, protect Cloud Computing services from brute force and malware attacks, build Honeypot systems on IaaS-based Cloud Computing, detect brute force attacks with Kippo and malware attacks with Dionaea. This study focuses on two types of attacks: Brute force attacks and malware [17].

Khadijah [3] explained that security threats could be burglary, worms or malware attacks and various actions that threaten system security. One of the security

measures that can be taken is to use a honeypot. The honeypot will collect information about the attacker and then present it into a system log and analysis tool to monitor and analyse it. The honeypot implementation in this study will identify Brute force attacks and pretend to be the original host by providing a fake system. In addition, it will mimic a web server to identify SQL Injection and Cross Site Scripting attacks [18]. From the research done by several previous researchers, none of them has discussed NIPS, Honeypot and Telegram Bots in one study. Most of them are discussed separately between Network Intrusion Prevention System (NIPS) and Honeypot. Therefore, I did the research to provide the latest literature on NIPS, Honeypot and Telegram Bots.

MARTIALS AND METHODS

The research method used in this study is to use the Network Intrusion Prevention System (NIPS) form to detect and prevent attacks on computer networks both from outside and from within and the addition of Honeypot as a method to analyze intruders or attackers when they have successfully entered into the network. A server, what commands are commonly used and how are they logged in. There is no need to worry about our central server because the first one to be attacked is a dummy server. Intruders or attackers will think they have entered the central server even though they are only trapped in a honeypot device [19-22]:

- The data analysis technique aims to describe and solve problems based on the data obtained. The analysis model used in this study is a qualitative descriptive analysis method, namely the data obtained from research that has been conducted at the Serang City Communication and Information Office, while the steps taken in analyzing qualitative descriptive data are as follows
- Collecting the required data and information about the network topology description in the Serang City Communication and Information Technology through interviews and direct observations in the field
- Identify existing problems and analyze in depth the system requirements by studying the components related to the system to be designed
- Make a system design by considering the system requirements following the needs and conditions in the field, such as functional and non-functional requirements
- Provide recommendations on the implementation of the system design that has been made that is suitable to be applied to the Serang City Communication and Information Office, especially the network security system

RESULT AND DISCUSSION

System Implementation

There are three stages in implementing the system in this research, first (1) Installation and Configuration, (2) System Testing and (3) System Evaluation. The installation and configuration process were the first after the topology design and the determination of the IP address had been agreed upon by the researcher and the Serang City Communication and Information Office. Based on the needs of the system installation and configuration process, there are three (3) namely: (Network Intrusion Prevention System, Honeypot and Telegram Bot), all of which will be described in full as follows [23-28]:

Installing the Network Intrusion Prevention System (Snort)

```
wget https://www.snort.org/downloads/snort/daq-2.0.7.tar.gz
tar -zxvf daq-2.0.7.tar.gz
cd daq-2.0.7
./configure
make
autoreconf -f -i
sudo make install
```

After DAQ and Snort have been installed, the next step is to create a Snort directory and this aims to make the configuration easier to organize, the following commands:

```
sudo mkdir /etc/snort
sudo mkdir /etc/snort/rules
sudo mkdir /etc/snort/rules/iplists
sudo mkdir /etc/snort/preproc_rules
sudo mkdir /usr/local/lib/snort_dynamicrules
sudo mkdir /etc/snort/so_rules
```

Then after the Snort directory has been created, the next step is to create some files that contain the rules and IP lists of the Snort directory, following the command:

```
sudo touch /etc/snort/rules/iplists/black_lists.rules
sudo touch /etc/snort/rules/iplists/white_lists.rules
sudo touch /etc/snort/rules/local.rules
sudo touch /etc/snort/s.id-msg.map
```

The next step is to create a simple rule to provide an alert on the server. Type the following command to disable other unused rule commands for anyone who pings using the ICMP protocol:

```
gedit /etc/snort/rules/local.rules
```

Type the following command to check the snort IPS configuration before running:

```
snort -T-c /etc/snort/snort.conf -I ens128
```

Make sure there are no errors and a message appears with the following terms "*Snort successfully validated the configuration*!" (Figure 4):

Honeypot Installation (Cowrie)

Cowrie is a new offshoot of Kippo Honeypot, with feature updates and providing emulation that records attack sessions. By registering for this session, we better understand the TTP attack's tools, tactics and procedures. TTP is becoming an increasingly used term in Cyber defence and Incident Response. In short, we will create IP tables to direct anyone who accesses via port 22 for SSH to port cowrie 2222 and port 23 for Telnet to port cowrie 2223. The following scheme will be used (Figure 5).

Lumina Quest Publishing

```
root@diskominfo: /home/diskominfo
          Using libpcap version 1.7.4
          Using PCRE version: 8.38 2015-11-23
          Using ZLIB version: 1.2.8

          Rules Engine: SF_SNORT_DETECTION_ENGINE  Version 3.1  <Bui
          Preprocessor Object: SF_POP  Version 1.0  <Build 1>
          Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
          Preprocessor Object: SF_DCERPC2  Version 1.0  <Build 3>
          Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
          Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
          Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
          Preprocessor Object: appid  Version 1.1  <Build 5>
          Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
          Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
          Preprocessor Object: SF_SSLPP  Version 1.1  <Build 4>
          Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
          Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
          Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
          Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
          Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>

Snort successfully validated the configuration!
Snort exiting
root@diskominfo:/home/diskominfo#
```

Figure 4: Snort Configuration Validation

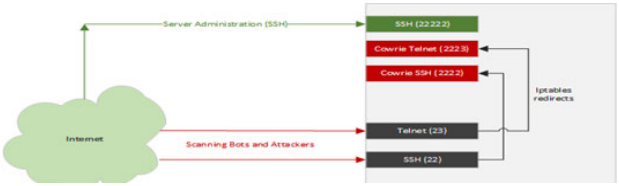

Figure 5: How the Cowrie Honeypot Works

Make sure we have installed the SSH Server, then change the default port to 55992 and check the ssh status to see if the SSH port changes after changing.

```
root@diskominfo : ~# nano /etc/ssh/sshd_config
# What ports, Ips and Protocols we listen for
Port 55992
root@diskominfo : ~# systemctl restart ssh
root@diskominfo : ~# systemctl status ssh
```

Make sure the Listening Port has changed to 55992 as follows (Figure 6):

Do an update, then install the following packages:

```
apt-get update
apt-get install git python-virtualenv libssl-dev libffi-dev build-essential
libpython-dev python2.7-minimal authbind
```

Add a new user with the name 'cowrie' by deactivating the password, then login as 'cowrie:

```
root@cowrie:~# adduser --disabled-password cowrie
Adding user `cowrie' ...
Adding new group `cowrie' (1000) ...
Adding new user `cowrie' (1000) with group `cowrie' ...
Creating home directory `/home/cowrie' ...
Copying files from `/etc/skel' ...
Changing the user information for cowrie
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
root@cowrie1:~# su - cowrie
cowrie@cowrie1:~$
```

To get Honeypot Cowrie, we can use git clone to duplicate or download files sourced from GitHub:

```
Agu 18 10:24:04 diskominfo-desktop systemd[1]: Reloading OpenBSD Secure Shell server.
Agu 18 10:24:04 diskominfo-desktop sshd[1102]: Received SIGHUP; restarting.
Agu 18 10:24:04 diskominfo-desktop systemd[1]: Reloaded OpenBSD Secure Shell server.
Agu 18 10:24:04 diskominfo-desktop sshd[1102]: Server listening on 0.0.0.0 port 55992.
Agu 18 10:24:04 diskominfo-desktop sshd[1102]: Server listening on :: port 55992.
Agu 18 10:24:04 diskominfo-desktop systemd[1]: Reloading OpenBSD Secure Shell server.
Agu 18 10:24:04 diskominfo-desktop sshd[1102]: Received SIGHUP; restarting.
Agu 18 10:24:04 diskominfo-desktop systemd[1]: Reloaded OpenBSD Secure Shell server.
Agu 18 10:24:04 diskominfo-desktop sshd[1102]: Server listening on 0.0.0.0 port 55992.
Agu 18 10:24:04 diskominfo-desktop sshd[1102]: Server listening on :: port 55992.
root@diskominfo-desktop:/home/diskominfo#
```

Figure 6: Change of Server Listening Port to 55992

```
cowrie@cowrie1:~$ git clone http://github.com/michel00sterhof/cowrie
Cloning into 'cowrie'...
remote: Counting objects: 9340, done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 9340 (delta 3), reused 2 (delta 0), pack-reused 9330
Receiving objects: 100% (9340/9340), 7.43 MiB | 2.32 MiB/s, done.
Resolving deltas: 100% (6415/6415), done.
Checking connectivity... done.
cowrie@cowrie1:~$
```

Now we will create a virtual environment to run honeypot cowrie and python script:

```
cowrie@cowrie1:~$ cd cowrie
cowrie@cowrie:~/cowrie$ virtualenv cowrie-env
Running virtualenv with interpreter /usr/bin/python2
New python executable in /home/cowrie/cowrie/cowrie-env/bin/python2
Also creating executable in /home/cowrie/cowrie/cowrie-env/bin/python
Installing setuptools, pkg_resources, pip, wheel...done.
cowrie@cowrie1:~$
```

The next step is to activate the python virtual environment and install the python cowrie package to run:

```
cowrie@cowrie1:~/cowrie$ source cowrie-env/bin/activate
(cowrie-env)  cowrie@cowrie1:~/cowrie$ pip install --upgrade pip
```

To create a Daemon for Honeypot Cowrie configuration, make sure we have entered the cowrie/etc/ folder, then copy the following file:

```
cp cowrie.cfg.dist cowrie.cfg
```

Please change the default hostname to make sure the attacker thinks he's logged into the correct server:

```
hostname = Server Diskominfo
```

Enable Telnet usage for Honeypot Cowrie and SSH is automatically enabled:

```
# Enable Telnet support, disabled by default
enabled = true
```

Configure IP tables to direct traffic 22 and 23 to ports 2222 and 2223 as follows:

```
root@cowrie:~# iptables -t nat -A PREROUTING -p tcp --dport 22 -j
REDIRECT --to-port 2222
root@cowrie:~# iptables -t nat -A PREROUTING -p tcp --dport 23 -j
REDIRECT --to-port 2223
```

To run the Honeypot service, we must log in using the user 'Cowrie' that was created at the beginning, then activate the Honeypot Cowrie service by typing the following command:

```
cowrie@cowrie:~/cowrie$ bin/cowrie start
Using default Python virtual environment "/home/cowrie/cowrie/cowrie-
env"
Starting cowrie: [twistd   --umask 0022 --pidfile var/run/cowrie.pid --logger
cowrie.python.logfile.logger cowrie ]...
```

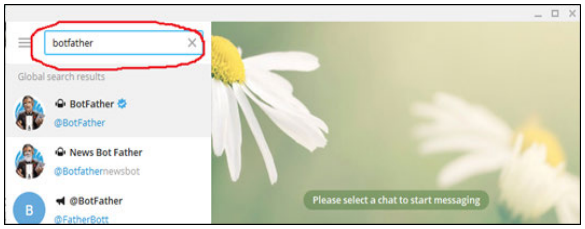

Figure 7: First Steps to Create a Telegram Bot

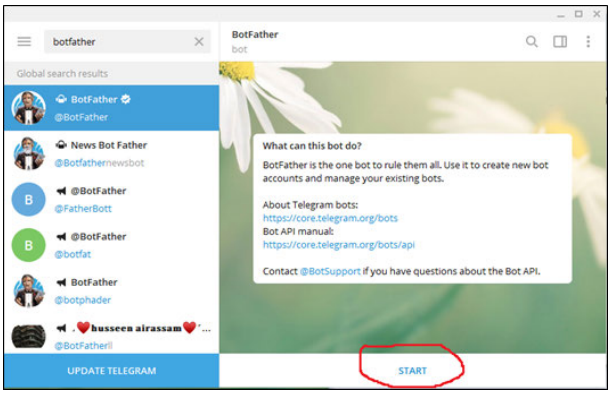

Figure 8: Second Step to Create a Telegram Bot

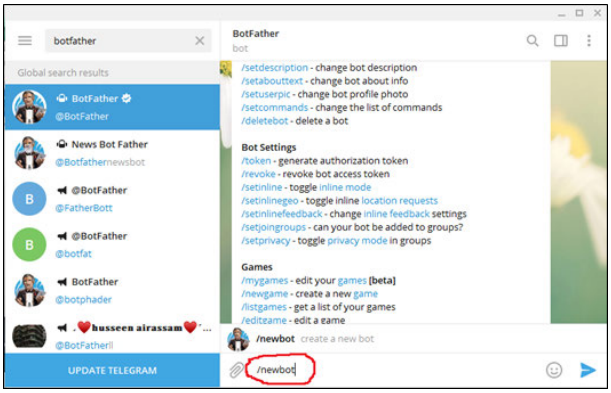

Figure 9: Creating a New Telegram Bot

Monitor activities caught by Honeypot Cowrie by typing the following command:

```
cowrie@cowrie:~/cowrie$ tail -f log/cowrie.log
```

Telegram Bot Installation

After the installation of the Network Intrusion Prevention System (NIPS) and Honeypot is complete, the final step in the installation and configuration process is the creation of the Telegram API for the creation of Telegram services so that all information about NIPS and Honeypot is installed. We sent it directly over the internet to the Telegram bot. Telegram is a provider of making Bots for free and we will get an API Token which will later be used to integrate our NIPS Server and Honeypot with Telegram. The first step that must be done is to search for @BotFather in the Telegram search field (Figure 7).

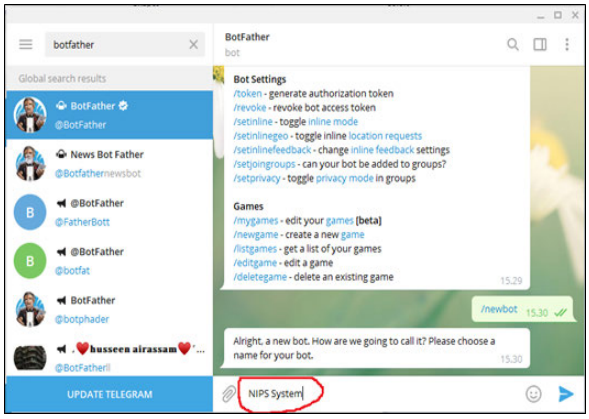

Figure 10: Making Telegram Bot Names

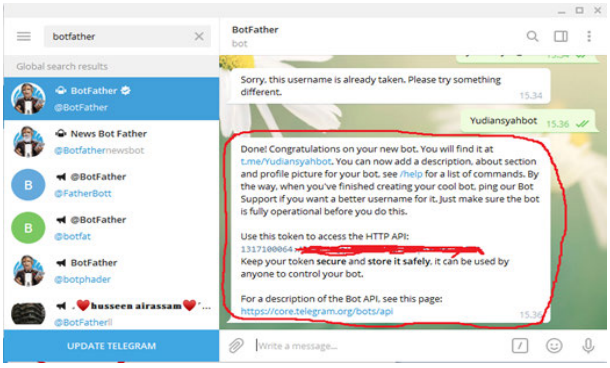

Figure 12: Getting the HTTP API Token

Select @BotFather then press START to start creating a Telegram Bot (Figure 8).

Type/newbot to create a new telegram bot (Figure 9).

Determine the name of the bot to be used. The bot's name cannot be the same as the names of the previous bots, so use a unique name (Figure 10).

After creating a telegram bot name, continue by creating a telegram bot username until @BotFather sends an HTTP API Token (Figure 11).

We need files with a programming language to connect Telegram Bot with NIPS Server and Honeypot. Researchers use the python programming language to integrate Telegram Bot with our Server. The code example is as follows (Figure 12).

Testing

Alpha Test: Alpha testing is done using Black Box Testing with five types of attack attempts, namely: (Ddos Attack, Bruteforce Telnet, NMAP Ping Sweep Scan, NMAP TCP Scan and SSH Login). The purpose of testing this system is to measure the level of curation of attack detection, the speed of information delivery and the running of the system (Figure 13).

Network Intrusion detection and prevention results Prevention System (NIPS). The IPS server successfully detected the instruction via the BASE web application on August 18, 2022, at 01:49:01 with SID '50000001' originating from IP Address 112,215,151.79 with a target of 103,102,250,246 which has port 80.

Figure 12: Coding of Connecting Telegram Bot with Server

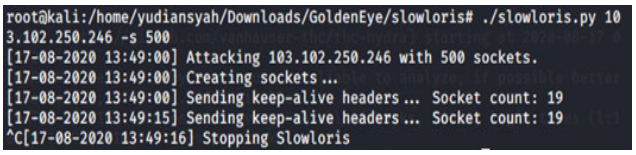

Figure 13: Ddos Attack Test

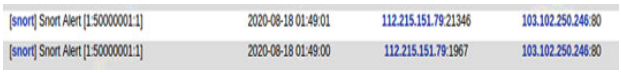

Figure 14: Ddos Attack Data

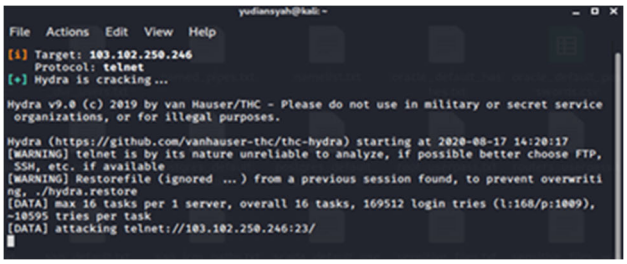

Figure 15: Telnet Brute Force Testing

Figure 16: Telnet Bruteforce Attack Data

Figure 17: Testing NMAP Ping Sweep Scan

NIPS dropped the connection to the IP Address 112.215.151.79 and entered the Ddos Attack trial category. After this Ddos Attack experiment, only the duplicate two packets were entered (Figure 14 and 15). Network Intrusion detection and prevention results Prevention System (NIPS). The IPS server successfully detected the instruction via the BASE web application on August 18, 2022, at 02:20:17 with SID '50000002' originating from IP Address 112,215,151.79 with a target of 103,102,250,246 which has port 23.

Figure 18: Ping Sweep Scan NMAP Attack Data

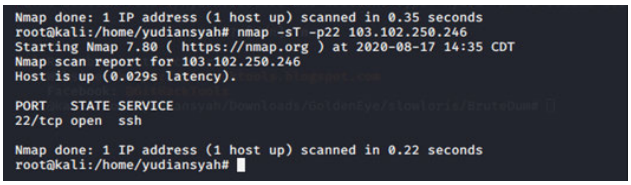

Figure 19: Testing NMAP TCP Scan

Figure 20: NMAP TCP Scan Detection and Prevention Test

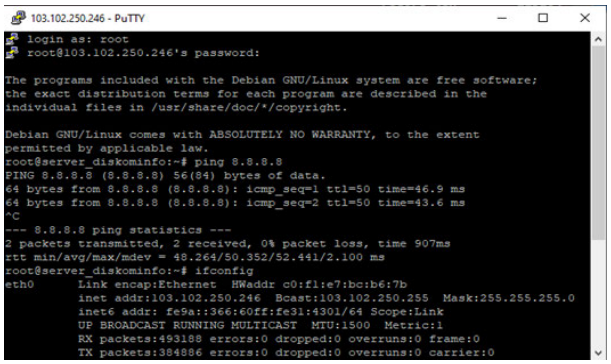

Figure 21: Attempt to Login User Root Password Administrator Successful

Figure 22: SSH Honeypot Login Attack Data

In the conclusion of this Bruteforce Telnet experiment, only a few seconds later, the login attempt using the Bruteforce technique, the connection was successfully dropped by NIPS to the IP Address 112.215.151.79 and entered the Bruteforce Telnet experiment category (Figure 16 and 17).

Detection results of Network Intrusion Prevention System (NIPS). The IPS server successfully detected the instruction via the BASE web application on August 18, 2022, at 02:28:27 with SID '10000004' originating from IP Address 140.213.7.42 with a target of 103.102.250.246. The conclusion of the NMAP Ping Sweep Scan experiment was detected by NIPS to IP Address 140.213.7.42 and was included in the NMAP Ping Sweep Scan experiment category (Figure 18 and 19).

The IPS server successfully detected the instruction via the BASE web application on August 18, 2022, at 02:35:07 with SID '10000005' originating from IP Address 112,215,151.97 with a target of 103,102,250,246.

Table I: Beta Test Conclusion

No	Questionnaire Questions	Answer Criteria Score					Percentage [Y*100]
		SK [N:1]	K [N:2]	B [N:3]	BS [N:4]	R = 20,Y [Total N:R]	
1	Question Number 1	0	0	2	3	0.90	90
2	Question Number 2	0	0	3	2	0.85	85
3	Question Number 3	0	0	3	2	0.85	85
4	Question Number 4	0	0	3	2	0.85	85
5	Question Number 5	0	0	3	2	0.85	85
6	Question Number 6	0	0	3	2	0.85	85
7	Question Number 7	1	1	2	1	0.65	65
8	Question Number 8	0	0	3	2	0.85	85
Mean						0.83	83

Network Intrusion Prevention System (NIPS) detection and prevention results. The conclusion of this NMAP TCP Scan experiment, detected by NIPS to IP Address 112.215.151.97 and entered into the category of NMAP Ping Sweep Scan experiment (Figure 20 and 21).

Attempts to log in with the root username and administrator password are recorded in the Honeypot Cowrie database. After this SSH Honeypot Login experiment, attackers will think that they have successfully entered the destination server 103.102.250.246 or research.serangkota.go.id via port 22. Even though they have entered the Honeypot Cowrie trap, all commands or commands will be recorded so that they can be analyzed further (Figure 22).

Beta Testing

Based on the results of the questionnaire testing, it produced an average value of 83% of the 8 (eight) questions of the questionnaire given and the data can be presented as follows (Table 1):

CONCLUSION

Based on the results of alpha testing, the research that has been done, namely the Network Intrusion Prevention System and the Honeypot system, obtained information about the attacker's behaviour so far. Then the beta test of the questionnaire results showed positive results with a percentage of 83%, that the test went well. This research produces 1 Server Network Intrusion Prevention System (NIPS), Honeypot and 1 Unit Metasploit Server. The results of this study are expected to provide the desired results. In the future, it can be applied to the Serang City Communication and Information Office as additional security on the computer network at the Serang City Communication and Information Office.

REFERENCES

[1] Pradipta, Y.W. and Asmunin. "Implementasi Intrusion Prevention System (IPS) Menggunakan Snort dan IP Tables Berbasis Linux." *Jurnal Manajemen Informatika*, vol. 7, no. 1, 2017

[2] Agustino, D.P. *et al*. "Implementasi honeypot sebagai pendeteksi serangan dan melindungi layanan cloud computing." *Konferensi Nasional Sistem dan Informatika - STMIK STIKOM Bali*, 10 Aug. 2017.

[3] Khadijah, S. "Implementasi Honeypot Pada Infrastruktur Cloud Computing." Politeknik Telkom Bandung, 2019.

[4] Alder, R. *Snort 2.1 Intrusion Detection.* 2nd Edn., Syngress Publishing, 2004.

[5] Ariyus, D. *Intrusion Detection System.* Andi, 2007.

[6] Arkaan, N. and D.V. Sakti. "Implementasi Low Interaction Honeypot Untuk Peningkat Keamanan Server dan Analisa Serangan Pada Protokol SSH." *Jurnal Nasional Teknologi dan Sistem Informasi*, 2019, p. 113.

[7] Arta, Y. *et al*. "Simulasi implementasi Intrusion Prevention System (IPS) Pada Router Mikrotik." *IT Journal Research and Development*, vol. 3, no. 1, Aug. 2018, e-ISSN: 2528-4053.

[8] At Taufiq, M.H. and Hidayati, A. "Rancang Bangun Aplikasi Biro Travel dengan SMS Gateway dan Google Maps API." *Multinetics*, vol. 2, no. 1, 2016, pp. 43–48.

[9] Atmaja, D.T. *et al*. "Notifikasi Adanya Serangan Pada Jaringan Komputer Dengan Mengirim Pesan Melalui Aplikasi Telegram dan Kontrol Server." *Seminar Nasional Sains dan Teknologi Universitas Muhammadiyah Jakarta*, e-ISSN: 2460-8416, 2018.

[10] Binanto, I. *Membangun Jaringan Komputer Praktis Sehari-hari.* Graha Ilmu, 2007.

[11] Cahyani, N.I. *et al*. "Uji Validitas dan Reabilitas Terhadap Implementasi Aplikasi Penjualan dan Pembelian." *Information System for Educators and Professionals*, 2016, pp. 21–34.

[12] Gondohanindijo, J. "Intrusion Prevention System (IPS) Untuk Mencegah Tindak Penyusupan/Intrusi." *Majalah Ilmiah INFORMATIKA*, vol. 3, no. 3, Sept. 2012.

[13] Haryanto, A.T. "Ini Bukti Indonesia Rentan Jadi Sasaran Serangan Siber." *Detik*, 7 Feb. 2019, https://inet.detik.com/security/d-4418609/ini-bukti-indonesia-rentan-jadi-sasaran-serangan-siber. Accessed 20 May 2020.

[14] Hidayat, W. "Pengguna Internet Indonesia Nomor Enam Dunia." *Kementerian Komunikasi dan Informatika*, 24 Nov. 2014, https://kominfo.go.id/content/detail/4286/pengguna-internet-indonesia-nomor-enam-dunia/0/sorotan_media. Accessed 20 May 2020.

[15] Ikhwan, S. and Elfitri, I. "Analisa Delay yang Terjadi pada Penerapan Demilitarized Zone (DMZ) terhadap Server Universitas Andalas." *Jurnal Nasional Teknik Elektro*, 2014, p. 118.

[16] Nawrocki, M.W. *et al*. "A Survey on Honeypot Software and Data Analysis." *arXiv [cs.CR]*, 2016.

[17] Mitchell, A. *An Intelligent Honeypot.* Cork Institute of Technology, 2018.

[18] Monoarfa, M.N. *et al*. "Analisa dan Implementasi Network Intrusion Prevention System di Jaringan Universitas Sam Ratulangi." *E-Journal Teknik Elektro dan Komputer*, vol. 5, no. 4, 2016.

[19] Mustofa, M.M. and Ariwibowo, E. "Penerapan Sistem Keamanan Honeypot dan IDS Pada Jaringan Nirkabel (Hotspot)." *Jurnal Sarjana Teknik Informatika*, vol. 1, no. 1, 2013.

[20] Pinkard, B. and Orebaugh, A. *Nmap in the Enterprise: Your Guide to Network Scanning.* Syngress Publishing, 2008.

[21] Pratomo, Y. "APJII: Jumlah Pengguna Internet di Indonesia Tembus 171 Juta Jiwa." *Kompas*, 16 May 2019, https://tekno.kompas.com/read/2019/05/16/03260037/apjii-jumlah-pengguna-internet-di-indonesia-tembus-171-juta-jiwa. Accessed 20 May 2020.

[22] Purbo, O.W. "Snort IPS." *OnnoWiki*, 25 July 2020, http://onnocenter.or.id/wiki/index.php/Snort_IPS. Accessed 1 June 2020.

[23] Satriawan, E. *et al*. "Implementasi IPS Berbasis Portsentry dan Vulnerability Assessment Berbasis Openvas Untuk Pengamanan Web Server." *Jurnal BITE*, vol. 1, no. 1, June 2019, e-ISSN: 2685-4066.

[24] Suandi, A. *et al*. "Pengujian Sistem Informasi E-Commerce Usaha Gudang Cokelat Menggunakan Uji Alpha dan Beta." *Jurnal INFORM*, vol. 2, no. 21, 2017, pp. 61–70.

[25] Triasanti, D. *Konsep Dasar Python.* Jakarta, 2001.

[26] Utomo, D. *et al*. "Membangun Sistem Mobile Monitoring Keamanan Web Aplikasi Menggunakan Suricata dan Bot Telegram Channel." *Seminar Nasional Teknoka Ke-2*, vol. 2, 2017, ISSN 2502-8782.

[27] Wibowo, R.A. "Analisis dan Implementasi IDS Menggunakan Snort pada Cloud Server di Jogja Digital Valley." AMIKOM Yogyakarta, 2014.

[28] Wijaya, B. *et al*. "Analisis dan Perancangan Keamanan Jaringan Menggunakan Teknik Demilitarized Zone (DMZ)." *Seminar Nasional Teknologi Informasi, Komunikasi dan Manajemen*, Universitas Bina Darma, Palembang, 2014, p. 398.