FORENSIC NETWORK ANALYSIS AND IMPLEMENTATION OF SECURITY ATTACKS ON VIRTUAL PRIVATE SERVERS

Naikson Fandier Saragih[1], Ridho Agus Wery Nanda Panjaitan[2], Mufria Jonatan Purba[3]
[1,2,3]Indonesian Methodist University
Jalan Hang Tuah No. 8 Medan, North Sumatra 20152
E-mail :saragihnaikson@gmail.com

ABSTRACT- PT Kodinglab Integrasi Indonesia's Virtual Private Server (VPS) product requires good quality standards, including security. The challenge that arises is still frequent disruptions to the protection of PT Kodinglab's VPS customers, where it is difficult to identify the source of the attack. Network forensics in the form of dead forensics and live forensics using the NIST method with the stages of collection, examination, Analysis, and reporting are used to find the source of the attack. Data for dead forensics comes from snort tools, and data for live forensics comes from capture Wireshark. The collection stage involves collecting attack data from snort logs and wireshark for life forensics. While the examination dataset stages are further analyzed and mapped. Advanced check on the server via syslog snort.
From the attack testing carried out to obtain information in the form of the attacker's IP address, destination IP address, date of the attack, server time, and type of attack from testing the TCP Flooding and UDP Flooding attacks, all attacks on the customer's VPS can be identified. The information obtained regarding the attacker is in the form of the date and time the attack occurred, the attacker's IP address and the victim's IP address, and the protocol used.

Keywords: Network Forensic, Dead Forensic, Live Forensic, Virtual Private Server, DDos, TCP Flooding, UDP Flooding.

1. INTRODUCTION

PT Kodinglab Integrasi Indonesia, one of the services provided is a Virtual Private Server for customers. Standards are prepared with high quality, especially regarding security in providing information system development and development services, mobile applications, development, e-government, and digital marketing. The Kodinglab server has now accommodated many web applications from government and private agencies using the Docker VPS. However, throughout 2021 there will still be 20 percent security disturbances to VPS customers of PT Kodinglab, such as backdoor and DDoS attacks.

Virtual Private Server (VPS) is one of the virtual machines provided for Internet hosting service providers that have been widely used since 2010. VPS is widely used when website traffic is high and requires more resources when you need Private IP access and create applications based on file sharing, streaming, and MAP Server [1]. Therefore VPS is inseparable from the target of security attacks.

To find the source of the attack, Network Forensic can be used by monitoring and analyzing resource usage, amount of traffic, and user activity, whether the server is on or off. The results of research using Network Forensic, investigators can quickly detect an attack and identify the attacker. In addition, the data needed for the investigation process, among other things, the date of the attack, the time of the attack, the attacker's MAC Address, the attacker's IP Address, the Victim's MAC Address, and the Victim's IP Address, can be presented quickly and precisely so that they can assist investigators in making decisions on attacks that occur [2].

2. RESEARCH CONTENT

2.1 Method

1. Network Forensics

The Network forensic section is a science that focuses on an area of computer networks and devices connected to a network to find attacker information and to find evidence of an attack on a computer network [3]. Network forensic stages can be seen in the image below:

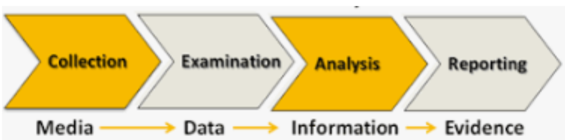


Figure 1 Forensic Process Model

a. collection

At this stage, it is carried out to obtain evidence according to procedures approved by the authorities.

b. examination

At this stage, it checks the activities that occur on the server.

c. Analysis

This stage is to analyze the data that has been collected to obtain estimates, possibilities, and hypotheses from the data that will potentially become evidence.

d. reporting

All of the steps above must be well documented to facilitate making reports easy for others to understand from the investigative activities carried out.

1. Live Forensic and Dead Forensic

Live forensics collects data/evidence of attack when the server is on, while Dead forensics collects data/evidence of attack when the server is down, which is generally stored in the snort log.

2. Research Flow

To carry out the research process to get the desired results, the researcher makes a research flow, as shown in Figure 2

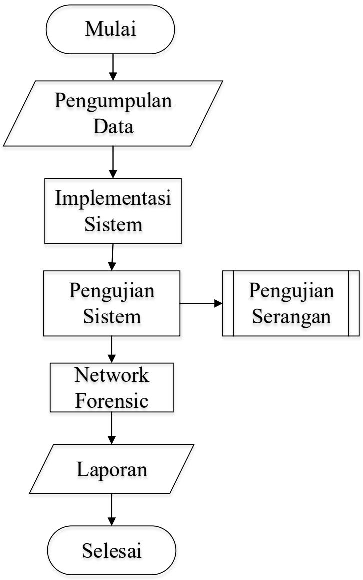


Figure 2 Research Flow

3. System Implementation

This stage is carried out by hiring a VPS service from the herza.id provider. The rented system is a KVM VPS with the Ubuntu 20.04 operating system. The specifications for the VPS that are rented can be seen in Table 1

Table1. KVM VPS Specifications

CPU used	Cores 4
Ram	4GB
Hard drive	80GB
Bandwidth	Unlimited
Operating system	Ubuntu 20.04

Next, install snort on the ubuntu server. Figure 3 shows that snort has been successfully installed.

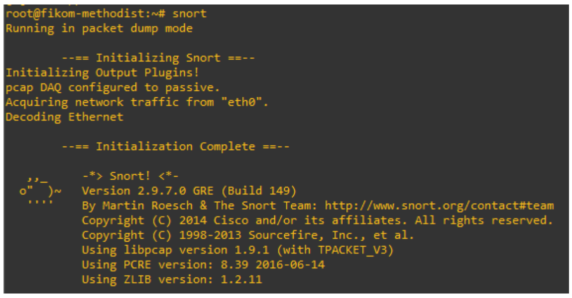


Figure 3 Display snort that has been installed

The image above shows the version of snort that has been successfully installed, namely snort version 2.9.7.0.

In Snort, there are rules to Snort's intelligence. The set rules provide intelligence at the time of protection, so the investigation process depends on rule intelligence. The script below is a configuration for saving rules. When there is an attack on the server, the log of the attack will be stored in the rules.

Next, add the DDoS attack rules with the script below

=> Added snort rules

alert ICMP any any -> $HOME_NET any (msg:"ICMP Flooding"; detection_filter:tracks by_src, count 30, second 60; sid1000006; rev2;)

alert tcp any -> $HOME_NET any (msg: there is a "TCP Flooding" attack; detection_filter: tracks by_src, count 30, second 60; sid1000006; rev2;)

alert udp any any -> $HOME_NET any (msg:"UDP Flooding"; detection_filter:tracks by_src, count 30, second 60; sid1000003; rev1;)

From the script above, the following results from changing the snort rules configuration script on the ubuntu server. Figure 4 is the snort rules configuration.

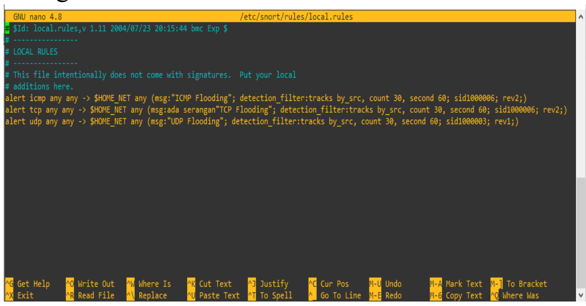


Figure 4 Configuring snort rules.

Next, we will install Wireshark on the client's PC. The following shows the Wireshark that has been successfully installed, shown in Figure 5

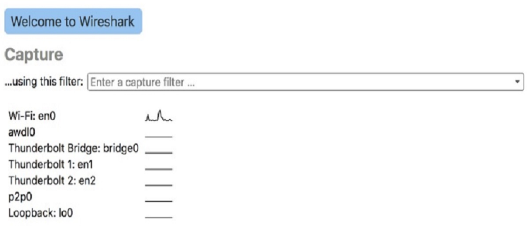


Figure 5 Wireshark View

2.4 Attack Testing

After implementing the system and configuring it, we then carry out attack testing to see the snort's response in monitoring data packets. For testing, only use attacks*Distributed Denial of Service*(DDoS) TCP Flooding and UDP Flooding using the LOIC application. Previously carried out the following tests. The CPU performance conditions in the initial state on the VPS overview tab are shown in Figure 6

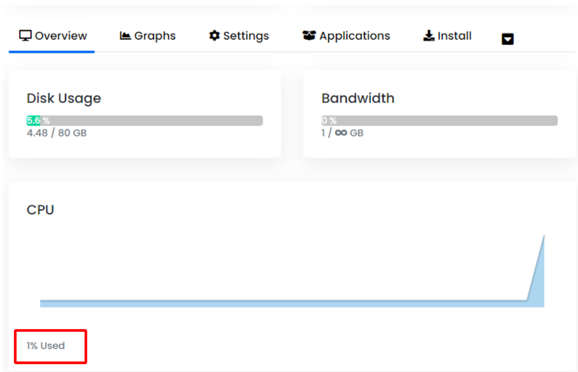


Figure 6 Initial CPU conditions

From the picture above, the results of the initial CPU condition data are obtained, which are shown in Table 2

Table 2Initial CPU state description

No	Tab	Information
1	Disk Usage	5.6%
2	Bandwidth	0%
3	CPUs	1% Used

To get network forensic results in finding evidence of security attacks, researchers conducted DDoS TCP Flooding and UDP Flooding attack scenarios, with information in Table 3

Table 3Attack Testing

No	Attack Type	Number of Tests
1	TCP Flooding	2
2	UDP Flooding	2

1. TCP Flooding

An attack that targets the TCP protocol takes advantage of the protocol connection connected to the server by flooding it*(flooding),* Experiments in making TCP connections that are accommodated by the server in buffers are indeed different. Still, the capacity is no more than a few hundred connections. By sending many packets into TCP, the buffer location will be full and cause the server to not work properly. Figure 7 is a test of the TCP Flooding attack.

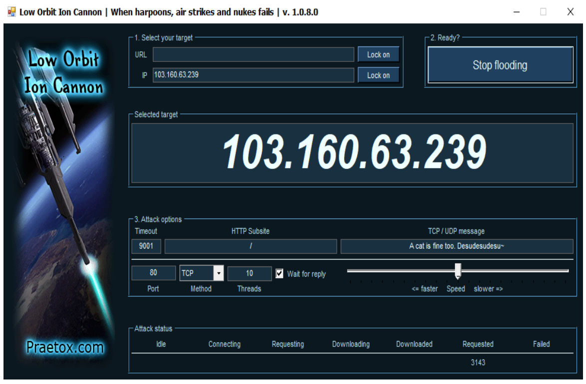


Figure 7 Types of TCP Flooding attacks

In the test above, the LOIC application sends many TCP packets so the server becomes full of these packets. With the "top" tools on Linux Ubuntu, network performance for the CPU and memory running on the server can be seen in Figure 8.

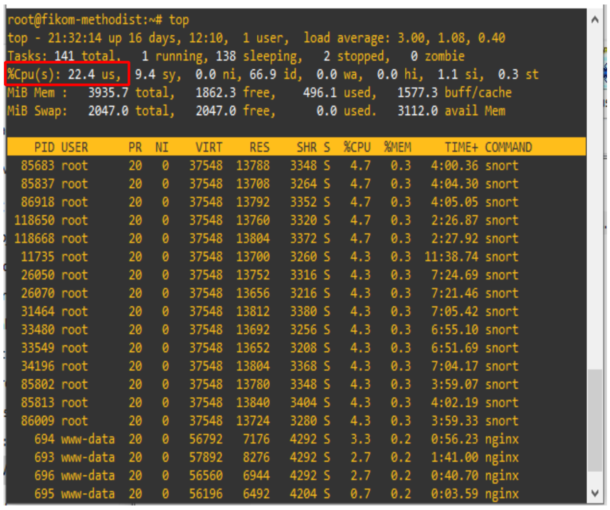


Figure 8 Server performance during a TCP Flooding attack

From the picture above, the CPU performance is at 22.4%

2. UDP Flooding

Testing the second attack is done by testing DDoS attacks with UDP-type *flooding* using the LOIC application. Figure 9 is a test of the UDP Flooding attack.

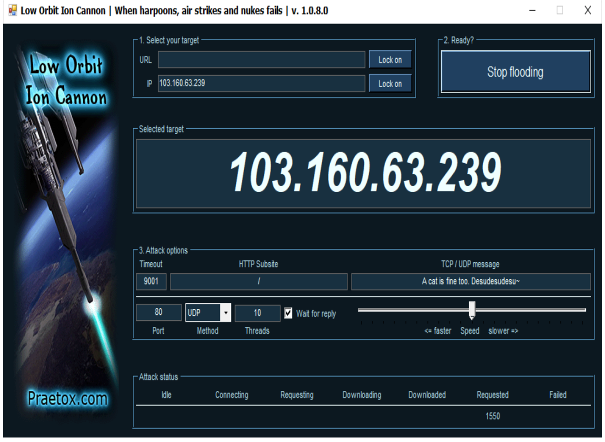
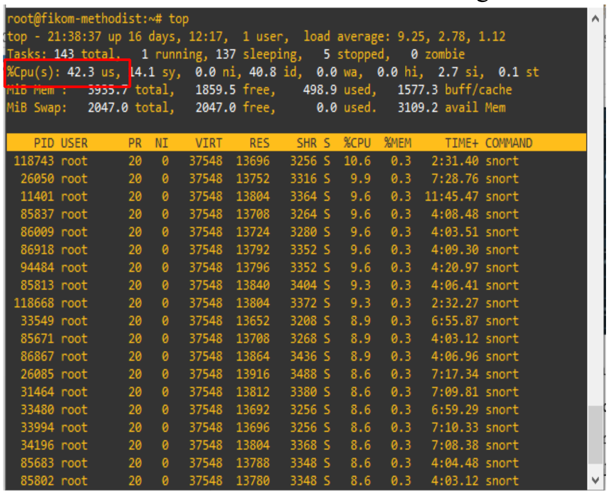


Figure 9 UDP Flooding type attack

The test above shows that LOIC floods User Datagram Protocol requests by sending many UDP packets so that the server becomes full. CPU performance and server memory with the "top" command on Linux ubuntu can be seen in Figure 10



```
root@fikom-methodist:~# top
top - 21:38:37 up 16 days, 12:17,  1 user,  load average: 9.25, 2.78, 1.12
Tasks: 143 total,   1 running, 137 sleeping,   5 stopped,   0 zombie
%Cpu(s): 42.3 us, 14.1 sy,  0.0 ni, 40.8 id,  0.0 wa,  0.0 hi,  2.7 si,  0.1 st
MiB Mem :   3935.7 total,   1859.5 free,    498.9 used,   1577.3 buff/cache
MiB Swap:   2047.0 total,   2047.0 free,      0.0 used.   3109.2 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
 118743 root      20   0   37548  13696   3256 S  10.6   0.3   2:31.40 snort
  26050 root      20   0   37548  13752   3316 S   9.9   0.3   7:28.76 snort
  11401 root      20   0   37548  13804   3364 S   9.6   0.3  11:45.47 snort
  85837 root      20   0   37548  13708   3264 S   9.6   0.3   4:08.48 snort
  86009 root      20   0   37548  13724   3280 S   9.6   0.3   4:03.51 snort
  86918 root      20   0   37548  13792   3352 S   9.6   0.3   4:09.30 snort
  94484 root      20   0   37548  13796   3352 S   9.6   0.3   4:20.97 snort
  85813 root      20   0   37548  13840   3404 S   9.3   0.3   4:06.41 snort
 118668 root      20   0   37548  13804   3372 S   9.3   0.3   2:32.27 snort
  33549 root      20   0   37548  13652   3208 S   8.9   0.3   6:55.87 snort
  85671 root      20   0   37548  13708   3268 S   8.9   0.3   4:03.12 snort
  86867 root      20   0   37548  13864   3436 S   8.9   0.3   4:06.96 snort
  26085 root      20   0   37548  13916   3488 S   8.6   0.3   7:17.34 snort
  31464 root      20   0   37548  13812   3380 S   8.6   0.3   7:09.81 snort
  33480 root      20   0   37548  13692   3256 S   8.6   0.3   6:59.29 snort
  33994 root      20   0   37548  13696   3256 S   8.6   0.3   7:10.33 snort
  34196 root      20   0   37548  13804   3368 S   8.6   0.3   7:08.38 snort
  85683 root      20   0   37548  13788   3348 S   8.6   0.3   4:04.48 snort
  85802 root      20   0   37548  13780   3348 S   8.6   0.3   4:03.12 snort
```

Figure 10 Server performance during a UDP Flooding attack

From the picture above, CPU performance has increased to 42.3%.

2.5 Network Forensics

Network forensics includes a collection stage to deal with previously committed crimes. The focus is to prevent future crimes.

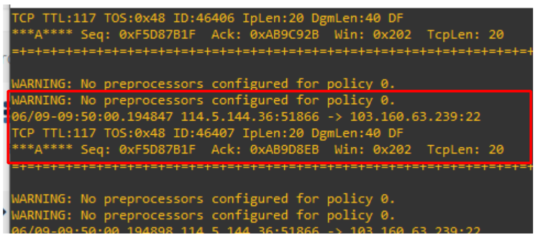
2.3.1 Collections

Retrieval of attack data using snort tools, namely on*logs*snort and also on wireshark for life forensics From the attack, testing is carried out to obtain information in the form of the attacker's IP address, destination IP address, date of the attack, server time, and type of attack.

1. *Log Data on Snort*

a. Log Data for TCP Flooding attacks

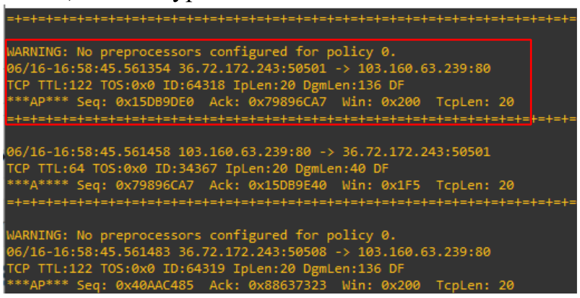
Attack trials were carried out twice, each shown in Figure 11 and Figure 12. The first attack was carried out on June 9, and the second on June 16, 2022



```
TCP TTL:117 TOS:0x48 ID:46406 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xF5D87B1F  Ack: 0xAB9C92B  Win: 0x202  TcpLen: 20
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
WARNING: No preprocessors configured for policy 0.
06/09-09:50:00.194847 114.5.144.36:51866 -> 103.160.63.239:22
TCP TTL:117 TOS:0x48 ID:46407 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xF5D87B1F  Ack: 0xAB9D8EB  Win: 0x202  TcpLen: 20
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
WARNING: No preprocessors configured for policy 0.
```

Figure 11 Snort log testing the first TCP Flooding attack

From Figure 11 above, it can be seen that the attacker's IP address is 114.5.144.36, the attack date is June 6, the server was attacked at 09.50, with id 46407, and the type of attack is a TCP attack.



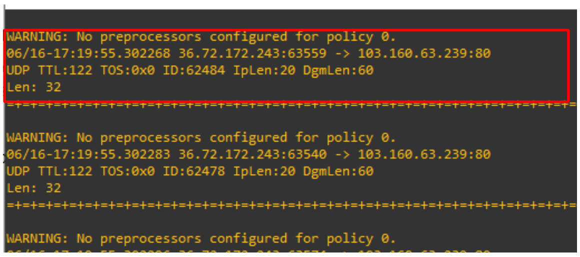
```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
06/16-16:58:45.561354 36.72.172.243:50501 -> 103.160.63.239:80
TCP TTL:122 TOS:0x0 ID:64318 IpLen:20 DgmLen:136 DF
***AP*** Seq: 0x15DB9DE0  Ack: 0x79896CA7  Win: 0x200  TcpLen: 20
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

06/16-16:58:45.561458 103.160.63.239:80 -> 36.72.172.243:50501
TCP TTL:64 TOS:0x0 ID:34367 IpLen:20 DgmLen:40 DF
***A**** Seq: 0x79896CA7  Ack: 0x15DB9E40  Win: 0x1F5  TcpLen: 20
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
06/16-16:58:45.561483 36.72.172.243:50508 -> 103.160.63.239:80
TCP TTL:122 TOS:0x0 ID:64319 IpLen:20 DgmLen:136 DF
***AP*** Seq: 0x40AAC485  Ack: 0x88637323  Win: 0x200  TcpLen: 20
```

Figure 12 Snort log of TCP attack testing Second flooding

From figure 12 above, it can be seen that the attacker's IP address is 36.72.172.243, the attack date is June 16, the server was attacked at 16.58, with id 64318, and the type of attack is a TCP attack.

b. Log Data for UDP Flooding attacks

Attack trials were carried out twice, respectively shown in **figure 13** and figure 14. the first attack was carried out on June 16, and the second on June 18, 2022



```
WARNING: No preprocessors configured for policy 0.
06/16-17:19:55.302268 36.72.172.243:63559 -> 103.160.63.239:80
UDP TTL:122 TOS:0x0 ID:62484 IpLen:20 DgmLen:60
Len: 32
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
06/16-17:19:55.302283 36.72.172.243:63540 -> 103.160.63.239:80
UDP TTL:122 TOS:0x0 ID:62478 IpLen:20 DgmLen:60
Len: 32
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
```

Figure 13 Snort log of the first UDP Flooding attack test

From Figure 13 above, it can be seen that the attacker's IP address is 36.72.172.243, the attack date is June 16, the server was attacked at 17.19, with id 62484, and the type of attack is a UDP attack.

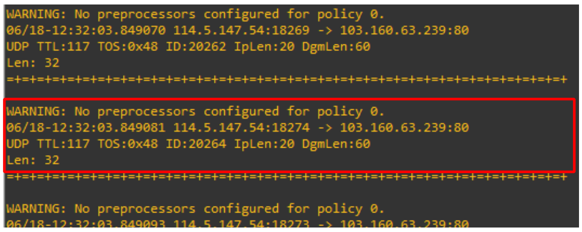


Figure 14 Snort log testing the second UDP Flooding attack

From Figure 14 above, it can be seen that the attacker's IP address is 114.5.147.54, the attack date is June 18, the server was attacked at 12.32, with id 20264, and the type of attack is a UDP attack.

2. ***Log Data on Wireshark***

In this tool, you can see the time, source, destination, and also protocol used.

a. TCP Flooding attack logs

Attack trials were carried out twice, with the results shown in Figure 15 and Figure 16, respectively.

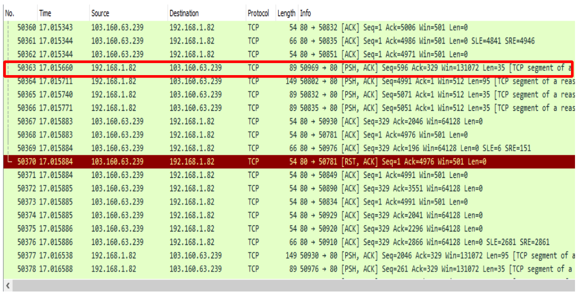


Figure 15 Data packet for testing the TCP Flooding attack first

From Figure 15, it can be seen that the attacker's IP number 50363 is 192.168.1.82, the destination IP is 103.160.63.239, and the protocol is TCP.

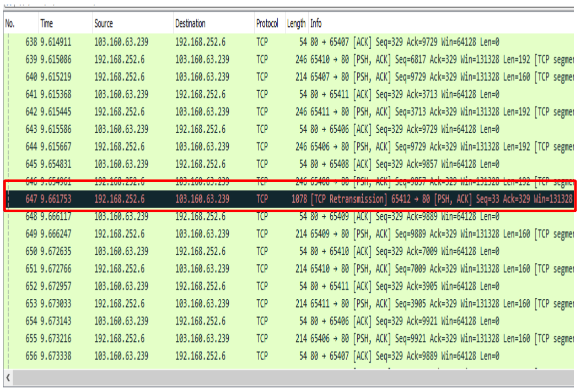


Figure 16 Data packet testing the second TCP Flooding attack

From the picture above, it can be seen that the attacker's IP at number 647 is 192.168.252.6, the destination IP is 103.160.63.239, and the protocol is TCP.

b. UDP Flooding Attack Data Logs

Attack trials were carried out twice, with the results shown in Figure 17 and Figure 18, respectively.

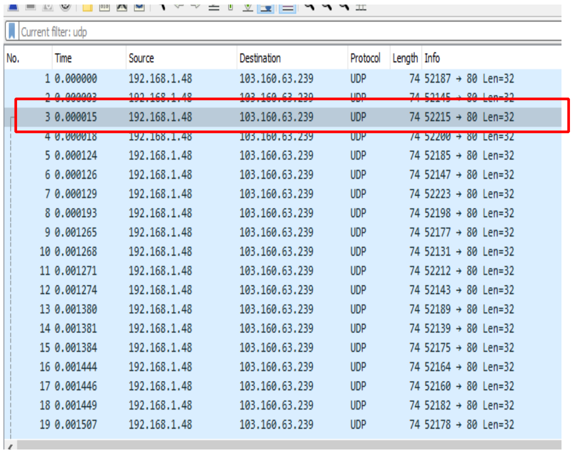


Figure 17 First UDP Flooding attack test data packet

From the picture above, it can be seen that the attacker's IP at number 3 is 192.168.1.48, the destination IP is 103.160.63.239, and the protocol is UDP.

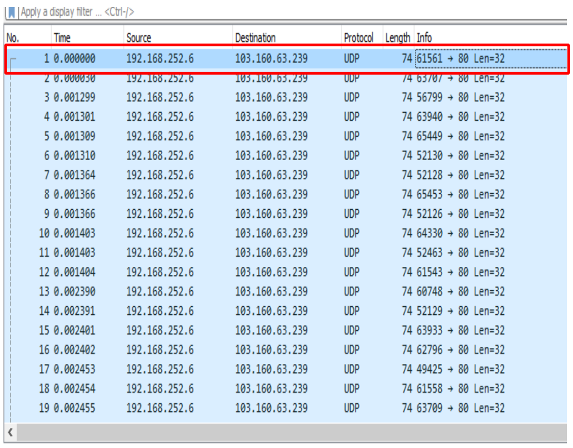


Figure 18 Data packet testing the second UDP Flooding attack

From the picture above, it can be seen that the attacker's IP at number 1 is 192.168.252.6, the destination IP is 103.160.63.239, and the protocol is TCP.

2.3.2 Examination

The data obtained forms a dataset and can be analyzed and mapped. Inspection is not only carried out on attack data obtained previously using tools but also checking activities carried out on the server via Syslog, as shown in Figure 19.

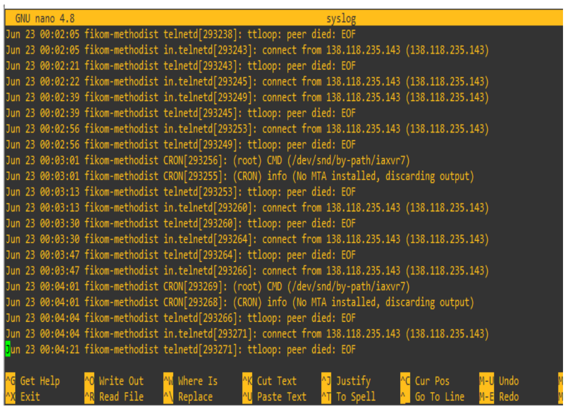


Figure 19 syslog snort

From the picture above, the activities carried out on the server will be recorded by snort, which is stored in the syslog, so from the information obtained, the server user can see irregularities if at any time there is a misuse of the server.

2.3.2 Analysis

From stages *collection*, evidence of an attack has been obtained, and a series of information obtained is evidence that there was an attack on the server, the server's IP address. The Analysis was carried out in the form of a comparison of CPU performance in the initial state before the attack was carried out and when the attack was carried out.

1. Server conditions on ubuntu in an initial state

To see the condition of the CPU on the server, you can do it with the "top" command. After the command is entered, we will see the results, as shown in figure 20 below.

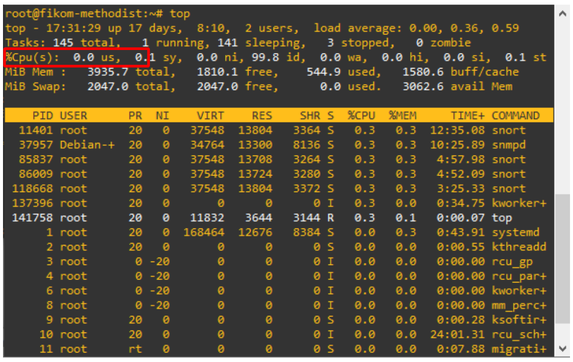


Figure 20 The initial condition of the server on ubuntu

From the picture above, it can be seen that the initial condition of the CPU is at 0.0%, and it can be concluded that the performance of the CPU is in a normal state.

2. Server condition on ubuntu after an attack

After the attack was carried out, the CPU performance condition increased, namely at 43.6%, which can be seen in Figure 21 below.

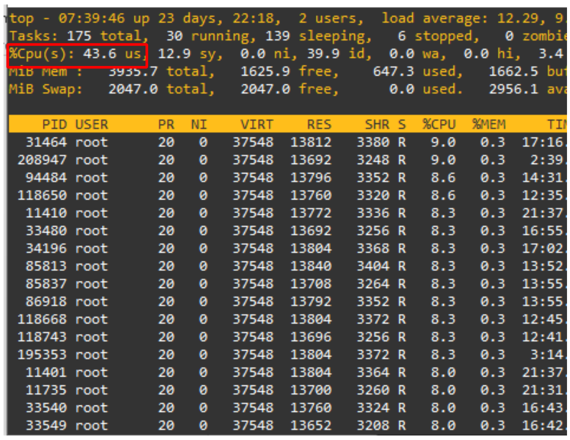


Figure 21 Server condition on ubuntu after testing the attack

Thus it can be concluded that the increase in CPU conditions before and after the attack was carried out was 43.6%.

2.3.2 Reports

1. Snort data source

The results of the investigation with the snort data source have occurred on the PVS server from several computers whose complete information can be seen in Table 4

Table 4 Details of data packets from Snort

N	Attacker Ip	The date of the attack	Destination IP	Server Time	ID	Protocols
1	114.5.144.36	June 9	103.160.63.239	09:50:00	46407	TCP
2	36.72.172.243	June 16	103.160.63.239	16:58:45	64318	TCP
3	36.72.172.243	June 16	103.160.63.239	17:19:55	62484	UDP
4	114.5.147.54	June 18	103.160.63.239	12:32:03	20264	UDP

2. Wireshark data source

The results of the investigation with the Wireshark packet capture data source have been an attack on the PVS server from several computers whose complete information can be seen in Table 5

Table 5 Details of data packets from Wireshark

NO	time	Attacker IP	Destination IP	Protocols
1	17.015884	192.168.1.82	103.160.63.239	TCP

2	9.66.17 53	192.168.25 2.6	103.160.63. 239	TCP
3	0.00001 5	192.168.1. 48	103.160.63. 239	UDP
4	0.00000 0	192.168.25 2.6	103.160.63. 239	UDP

3 CONCLUSION

Conclusion

Some conclusions are as follows:

1. *Network forensics*, in collecting evidence of attacks in the snort log, has been able to identify the attacker in the form of the date of the attack, the time of the attack, the attacker's IP address, the victim's IP address, ID, the protocol can be presented quickly and precisely so that it can assist investigators in making decisions about the attack that occurred on a Virtual Private Server (VPS).
2. Collecting evidence of attacks on wireshark log data in the form of the time of the attack, attacker's IP address, victim's IP address, and protocol.

4 CLOSING

In this study, the authors provide suggestions that are necessary for the next development, which are as follows:

1. The research is also expected to provide recommendations for tools used in detecting and identifying attacks and attackers.
2. Future research also examines how to handle and prevent security attacks on Virtual Private Servers (VPS).

BIBLIOGRAPHY

[1] Dewi, EK (2017). Snort Log Analysis Using Network Forensic. JIPI (Scientific Journal of Informatics Research and Learning), 2(2), 72–79. https://doi.org/10.29100/jipi.v2i2.370

[2] Eka, R., Rachman, A., & Wahyu, T. (2010). Virtual Private Server (VPS) as an Alternative to a Dedicated Server. Seminar on Intelligent Technology and Its Applications, SITIA, 2–7.

[3] Fahana, J., Umar, R., & Ridho, F. (2017). QUERY: Journal of Information Systems Volume: 01, Number: 02, October 2017 ISSN 2579-5341 (online) Utilization of Telegram as an Attack Notification for Network Forensics Purposes QUERY: Journal of Information Systems Volume: 01, Number: 02, October 2017. 5341(October), 6–14.

[4] Hae, Y. (2021). Network Security Analysis on the Web From Sniffing Attacks With Experimental Methods. JATISI (Journal of Informatics Engineering and Information Systems), 8(4), 2095–2105. https://doi.org/10.35957/jatisi.v8i4.1196

[5] Hafizh, MN, Riadi, I., & Fadlil, A. (2020). Network Forensics Against ARP Spoofing Attacks using the Live Forensic Method. Journal of Telecommunications and Computers, 10(2), 111. https://doi.org/10.22441/incomtech.v10i2.8757

[6] Hunt, R., & Zeadally, S. (2012). Network Forensics: An Analysis of. 36–43.
Information, FT (2017). Network Forensic On Cloud-Based Networks Scientific Articles. 672011212.

[7] Kamajaya, GEA, Riadi, I., Prayudi, Y., & Dahlan, UA (2020). STATIC FORENSICS INVESTIGATION ANALYSIS OF MAN ATTACK IN THE ARP POISONING BASED ON MAN IN THE MIDDLE ATTACK IN STATIC. 3(1), 6–12. https://doi.org/10.33387/jiko

[8] Muhyidin, A., & Yogyakarta, UN (2019). Ubuntu Server Fundamentals book (Ubuntu Camp 2016). march.
Nasution, MAH, & Laksono, AT (2020). Investigation of Backdoor Remote Access Trojan (RAT) Attacks Against Smartphones. 7(4), 505–510. https://doi.org/10.30865/jurikom.v7i4.2301

[9] Netsec. (2020). Implementation of Snort as a detection tool. https://netsec.id/snort-nids/%0Ahttps://media.neliti.com/media/publications/142056-ID-implementasi-snort-As-alat-pendetek.pdf%0A

[10] Supervisor, D., Hidayanto, BC, & Information, DS (2018). NETWORK FORENSIC ANALYSIS OF DOS ATTACKS ON UBUNTU SERVER OPERATING SYSTEMS 16. 04 DAN MICROSOFT WINDOWS SERVER 2016 NETWORK FORENSICS ANALYSIS ON DOS ATTACK AT UBUNTU SERVER 16. 04 AT MICROSOFT WINDOWS SERVER.

[11] http://blog.iweb.com/en/2010/06/why-should-i-care-about-sql-server/4772.html, March 17, 2011, 19.30