# Router Forensic Analysis against Distributed Denial of Service (DDoS) Attacks

Oldy Ray Prayogo
Department of Information System
Universitas Ahmad Dahlan
Yogyakarta of Indonesia

Imam Riadi
Department of Information System
Universitas Ahmad Dahlan
Yogyakarta of Indonesia

## ABSTRACT

A Distributed Denial of Service (DDoS) attack is a multi-computer attack targeting a single device to increase the amount of network traffic and paralyze the target. The number of DDoS attacks continues to increase and has a more sophisticated variety of attacks so that an effective technique is needed to find out information related to these attacks. This research uses the Network Forensic Generic Process Model which has 8 stages, namely preparation, detection, collection, preservation, examination, analysis, investigation, presentation, and using the live forensic method in the data acquisition process. This research uses the help of tools including Snort, Wireshark, Elasticsearch, Kibana, and Logstash. This research succeeded in obtaining digital evidence containing information related to the attack, namely, there were 5 IP addresses for the attacker, attacks that occurred on port 80 TCP with one target IP address, attacker ID, the total number of attacks totaling 126,286 attack packets and the time of the attack. This research succeeded in obtaining data and information derived from the evidence obtained, from these results it can make it easier to strengthen security at existing points of vulnerability, or as digital evidence in court.

## Keywords

DDoS Attacks, Network Forensic Generic Process Model, Live Forensic.

## 1. INTRODUCTION

Computer technology has been utilized in the lot fields and makes it easy for its users to carry out activities online. However, the convenience and sophistication of this technology does not always have a positive impact but also has a negative impact, one of which is the security problem against Distributed Denial of Service Attack (DDoS), namely an attack technique by flooding the system with a large number of service requests until the system is unable to respond. [1]. Data from the Imperva research lab revealed that DDOS attacks that occurred in the world increased to 680 Gbps in 2018. [2]. Network forensics is a part of digital forensics that deals with investigation events and activities related to digital networks, which include monitoring and capturing network traffic and related data from devices connected to the network to gather evidence in a manner acceptable to a court of law [3]. Live forensics is the activity of collecting evidence from the system running in real-time on memory or other storage media so that it is not lost or changed when the system is shut down. [4]. Wireshark is a tool that is used to capture and analyze packets on the network, but it is also used to analyze the protocol used [5]. Intrusion Detection System (IDS) works by monitoring packets in the network and making comparisons with the existing signature database in the IDS system or the attributes of the attack never known. [6]. The investigation process is carried out to obtain data that can later be used as evidence so that the process is successful, a stage in the data collection process is needed, in this research, the stages of the investigation process are carried out using the Network Forensic Generic Process Model, the process has several stages of the investigation, namely preparation, detection. , incident response, collection, preservation, examination, analysis, investigation, and presentation. [7].

## 1.1 Literature Review

### *1.1.1* Previous Studies

Ram Charan Baishya, Nazrul Hoque, & Dhruba Kumar Bhattacharyya (2017) This study has shown that in this case, the source with the attack packet rate that exceeds the average value is significant and thus can detect the change. [8]

Abhlasha B. Pawade, Prof. Santosh T. Waghmode (2017) has researched to detect DDoS attacks by conducting Multivariate Correlation Analysis (MCA) and ANN analysis on the KDD Cup 99 dataset. This analysis shows good performance and resilience of MCA and shows that the system very well outperforms the other two previously developed approaches in terms of accuracy detection attack. [9].

Rajput, Reena & Agrawal (2017) in their research entitled “Denial of Service Attack Detection Using Random Forest Classifier with Information Gain” found that a randomized approach also had a major influence on the accuracy of the overall analysis [10].

Mazdadi, Muhammad Itqon & Imam Riadi (2017) in their research entitled "Forensic Analysis on RouterOS Using the Live Forensics Method" show that this study found information related to attack information that can be used to perform analysis to disclose attack activities that occur on the Router. [11].

Ali et al., (2016) in their research entitled "A Neural Network Model for Detecting DDoS Attacks Using Darknet Traffic Features" shows that the technique can detect attacks very well compared to the detection system it uses classifier attack online conventional. [12].

### *1.1.2 Network Forensics*

Network Forensics is an activity to capture, record, and analyze events that occur on the network to find the source of security attacks or other incidents of problems [13].

### *1.1.3 Live* Forensics

Live Forensics is a method used to acquire data when the system is running. [14]

### *1.1.4 Network Attack Classification*

Network Attack Classification is a process for grouping network attacks into certain subgroups to determine the same type of attack in the future. The purpose of this grouping is to help us find out more details about the characteristics of the type of attack. [15]

### *1.1.5 Distributed Denial of Service (DDoS)*

DDoS attacks occur because the number of packets sent to the target server is very large, exceeding the capabilities of the server which makes the system slow or even crash. [16].

### *1.1.6 Router*

The router is a hardware used in a computer network and has a function to connect and aim to forward data packets between two or more different networks so that they can communicate with each other. [17].

### *1.1.7 Intrusion Detection System (IDS)*

Intrusion Detection System (IDS) is a method used to detect suspicious activity on a system or network. [18]

### *1.1.8 Snort*

Snort is a network security system that has a function to detect intrusions on the network in the form of attacks, intrusions, reconnaissance, and other threats, besides that it can be used for preventive measures. [19]. Snort also has components that work together in detection, the components in snort are shown in Figure 1.

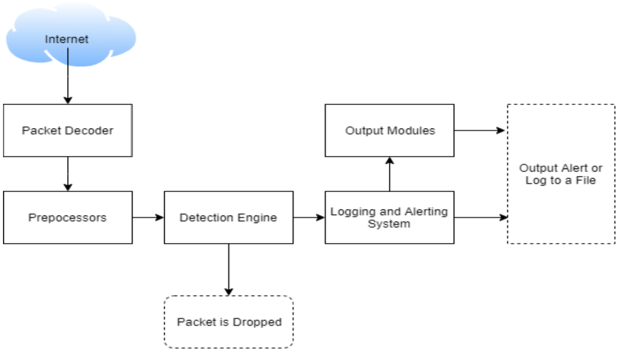


**Figure 1. The Snort component**

The components in the snort system are related to each other so that the system can function properly against suspicious network traffic, and then create outputs that can be analyzed later. [20]

### *1.1.9 Elasticsearch*

Elasticsearch is a machine that is used as a tool to search and analyze all types of data. The process that occurs is the processing of raw data is carried out in several stages before it ends at the Indexing stage [21].

### *1.1.10 Kibana*

Kibana is an additional system in the KIbana engine that is used for data visualization and management of indexed content in the Elasticsearch engine. Kibana has a dashboard feature that functions as a data collector that has been visualized on the dashboard page. [22]

### *1.1.11 Logstash*

Logstash is a system that functions to collect and process data before it is sent to the Elasticsearch engine. All data that has been collected will then be sent to Elasticsearch for the indexing stage. [23]

# 2. METHODOLOGY

## 2.1 Research Scenario

The forensic investigator analyzes the attack loThe case study used in this research is an attack that occurs on a PC router, where the PC router is connected to a local network that contains several clients and servers, the attack is carried out so that the client cannot access the resource because it has been fulfilled by the attack traffic. The research scenario is used to show the steps involved in carrying out the live forensic method in DDoS attacks. The design of DDOS attack simulations on Router devices uses the Pentmenu, to analyze attacks that occur on the Router using the Wireshark, Kibana, Logstash, Elasticsearch, and Snort. DDoS attack scenarios that occur on Router devices are as shown in Figure 2.

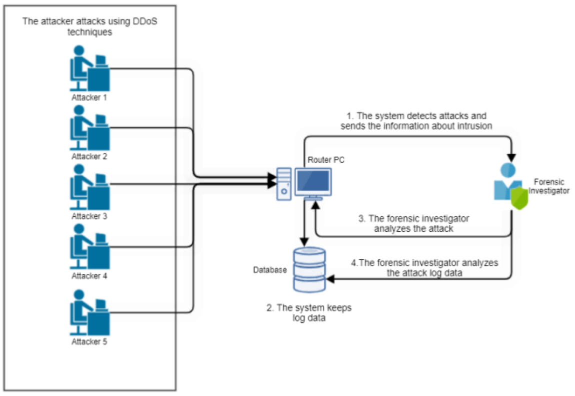

The attacker attacks using DDoS techniques
Attacker 1
Attacker 2
Attacker 3
Attacker 4
Attacker 5
1. The system detects attacks and sends the information about intrusion
Router PC
Forensic Investigator
3. The forensic investigator analyzes the attack
Database
4.The forensic investigator analyzes the attack log data
2. The system keeps log data

**Figure 2. Simulation of a Distributed Denial of Service (DDoS) attack**

In Figure 2 above, it can be seen that DDoS attacks are carried out by utilizing multiple attacks through several devices. Because a network engineer detects an anomaly on a request that occurs on the server then performs a live forensic stage to find out the source of the attack and the type of attack that occurred, which is then carried out in the analysis stage of the attack that occurred.

## 2.2 Research Stages

The process for conducting forensic analysis in this study is based on Network Forensic Generic Process Model which is an analysis model used to perform the detection, data acquisition, and analysis stages that occur in computer networks. The research stages are as shown in Figure 3.

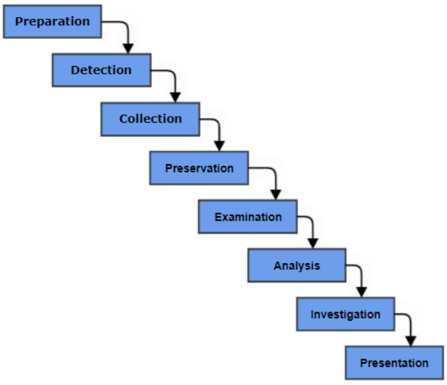


**Figure 3. Network Forensic Generic Process Model**

Figure 3 shows that to research a DDoS attack, the first stage carried out is to make preparations by configuring the devices and systems used, then testing the system that has been created to detect attacks, then the attack data is collected and carried out in stages. analysis of the attack log which is then made into reports.

### *2.2.1 Preparation*

The first step in this research is to design a network monitoring system and attack detection.

### *2.2.2 Detection*

This stage is the stage where the system is carried out in the testing phase with Distributed attacks Denial of Service (DDoS) performed by multiple devices.

### *2.2.3 Collection*

The Collection stage is the stage where the detection system installed on the device detects any unusual activity occurring on the network. The flow of the log data collection process is shown in Figure 4.

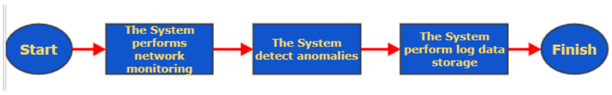

Start
The System performs network monitoring
The System detect anomalies
The System perform log data storage
Finish

**Figure 4. Log Data Collection Process Flow**

Figure 4 shows that the system starts collecting data automatically and stores it for the next forensic stage.

### *2.2.4 Preservation*

After the log data has been collected by the monitoring system, the next stage is the preservation stage, where the authenticity of the data that has been obtained must be maintained so that there are no changes in data that result in inconsistencies in data validity. The data maintenance stage is as shown in Figure 5.

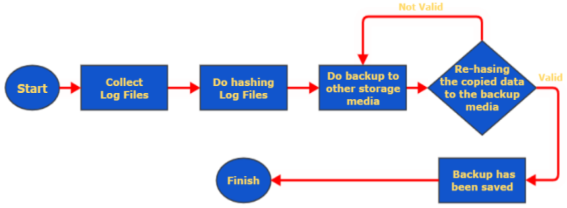


**Figure 5. Data Maintenance Stage**

Figure 5 shows how the flow for maintaining log files by backing up to other storage media, in this flow the hashing stage is carried out 2 times which functions to validate data if there is a data change then the hashing results also provide a different value.

### *2.2.5 Examination*

At this stage, the data that has been collected and obtained is carried out at the stage of checking whether the data is related to events that have occurred and have data that can be used in the analysis stage.

### *2.2.6 Analysis*

The analysis stage is the stage where the data has been obtained processed to be easy information understood so that it can be useful for making decisions.

### *2.2.7 Investigation*

The investigation stage is the stage for compiling data and identifying data findings, such as which IP carried out the attack, where the attack occurred, when the attack occurred, how it happened, and why the attack occurred.

### *2.2.8 Presentation*

The Presentation stage is carried out to show the results of the investigation in a language that is easy to understand so that it is easier to capture information.

## 3. RESULT AND DISCUSSION

The implementation of this research study requires software and hardware as research material for the smooth running of the research process. The tools and materials needed in this study are as shown in table 1.

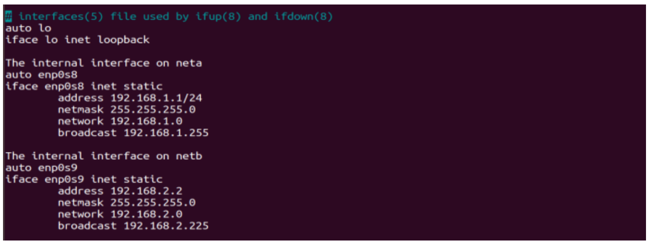
**Table 1. Tools and Materials**

| No. | Device Name | Information |
|---|---|---|
| 1 | Laptop 1 | Core i7, 16Gb RAM, 1Tb HDD |
| 2 | PC router | Core i3, HDD 500GB, 8GB RAM |
| 3 | PC Attacker 1 | Core i3, 8Gb RAM, 500Gb |
| 3 | PC Attacker 2 | Core i3, 8Gb RAM, 500Gb |
| 4 | PC Attacker 3 | Core i3, 8Gb RAM, 500Gb |
| 5 | PC Attacker 4 | Core i3, 8Gb RAM, 500Gb |
| 6 | Windows 10 | The operating system used to conduct research |
| 7 | Kali Linux | The operating system used for the attack process |
| 8 | Ubuntu 16.04 | The operating system used for the router PC |
| 9 | Snort | The system used for network anomaly detection and forensic stage |
| 10 | Elasticsearch | The system used for the log data indexing stage |
| 11 | Logstash | The system is used for network monitoring and anomaly detection |
| 12 | Kibana | The system used for log data visualization |
| 13 | Pentmenu | Tools used as a tool to attack |
| 14 | Wireshark | The system used for network monitoring and packet analysis in the network at the forensic stage |

## 3.1 Preparation

### *3.1.1 Router Configuration*

Router configuration is done to connect between different networks. The configuration is done by giving an IP address to each different interface and later it is used to connect the connection. The router configuration display is as shown in Figure 6.



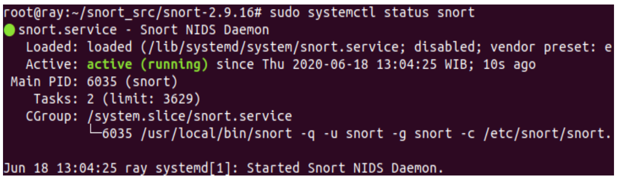
```
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

The internal interface on neta
auto enp0s8
iface enp0s8 inet static
        address 192.168.1.1/24
        netmask 255.255.255.0
        network 192.168.1.0
        broadcast 192.168.1.255

The internal interface on netb
auto enp0s9
iface enp0s9 inet static
        address 192.168.2.2
        netmask 255.255.255.0
        network 192.168.2.0
        broadcast 192.168.2.225
```

**Figure 6.Configuration of the Network Interface Card (NIC)**

Figure 6 shows how to configure the Network Interface Card (NIC). The addition of IP addresses is done on the enp0s8 interface which is connected to the local network and enp0s9 which is connected to the server.

### *3.1.1.1 Snort System Implementation*

The Snort system is used to detect anomalies in the network. The status that the snort system has been running is as shown in Figure 7.



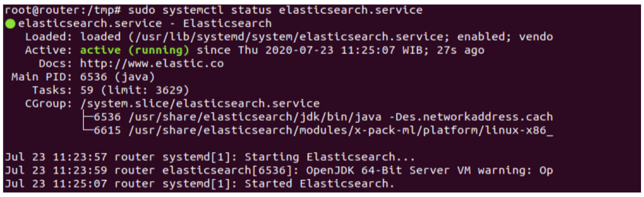
```
root@ray:~/snort_src/snort-2.9.16# sudo systemctl status snort
● snort.service - Snort NIDS Daemon
   Loaded: loaded (/lib/systemd/system/snort.service; disabled; vendor preset: e
   Active: active (running) since Thu 2020-06-18 13:04:25 WIB; 10s ago
 Main PID: 6035 (snort)
    Tasks: 2 (limit: 3629)
   CGroup: /system.slice/snort.service
           └─6035 /usr/local/bin/snort -q -u snort -g snort -c /etc/snort/snort.

Jun 18 13:04:25 ray systemd[1]: Started Snort NIDS Daemon.
```

**Figure 7: Status of Active Snort System**

Figure 7 shows that after configuration and settings, the system has been successfully executed to monitor and detect attacks that occur on the network.

### *3.1.1.2 Elasticsearch System Implementation*

Elasticsearch is a system used to do the Indexing stage. The status that the system elasticsearch has been running as shown in Figure 8.



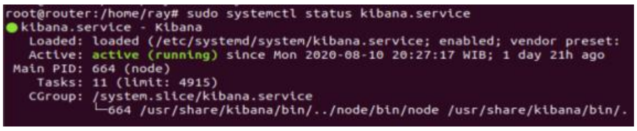
```
root@router:/tmp# sudo systemctl status elasticsearch.service
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vendo
   Active: active (running) since Thu 2020-07-23 11:25:07 WIB; 27s ago
     Docs: http://www.elastic.co
 Main PID: 6536 (java)
    Tasks: 59 (limit: 3629)
   CGroup: /system.slice/elasticsearch.service
           ├─6536 /usr/share/elasticsearch/jdk/bin/java -Des.networkaddress.cach
           └─6615 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_

Jul 23 11:23:57 router systemd[1]: Starting Elasticsearch...
Jul 23 11:23:59 router elasticsearch[6536]: OpenJDK 64-Bit Server VM warning: Op
Jul 23 11:25:07 router systemd[1]: Started Elasticsearch.
```

**Figure 8. Active Elasticsearch System Status**

Figure 8 shows that the Elasticsearch system has been successfully running in the system after the installation and configuration stages were previously carried out which can later be used to do the indexing stage of log data originating from Logstash.

### *3.1.1.3 Implementation of the Kibana System*

Kibana is a system used to visualize data and management of content. The status that the Kibana system has been running is as shown in Figure 9.



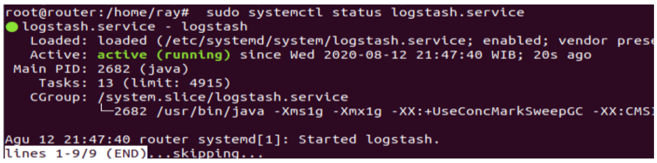
```
root@router:/home/ray# sudo systemctl status kibana.service
● kibana.service - Kibana
   Loaded: loaded (/etc/systemd/system/kibana.service; enabled; vendor preset:
   Active: active (running) since Mon 2020-08-10 20:27:17 WIB; 1 day 21h ago
 Main PID: 664 (node)
    Tasks: 11 (limit: 4915)
   CGroup: /system.slice/kibana.service
           └─664 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bin/.
```

**Figure 9. Status of the Active Kibana System**

Figure 9 shows that the system has been successfully running on the device and then to access the system by opening a web browser on the device, by accessing the previously configured localhost.

### *3.1.1.4 Logstash System Implementation*

Logstash serves as a tool for collecting and processing data before it is sent to the Elasticsearch engine. The status that the Logstash system is running is as shown in Figure 10.



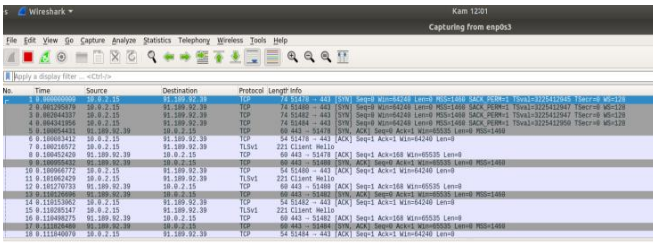
```
root@router:/home/ray#  sudo systemctl status logstash.service
● logstash.service - logstash
   Loaded: loaded (/etc/systemd/system/logstash.service; enabled; vendor prese
   Active: active (running) since Wed 2020-08-12 21:47:40 WIB; 20s ago
 Main PID: 2682 (java)
    Tasks: 13 (limit: 4915)
   CGroup: /system.slice/logstash.service
           └─2682 /usr/bin/java -Xms1g -Xmx1g -XX:+UseConcMarkSweepGC -XX:CMSI

Agu 12 21:47:40 router systemd[1]: Started logstash.
lines 1-9/9 (END)...skipping...
```

**Figure 10. Active Logstash System Status**

Figure 10 shows that the Logstash system has been successfully run on the device and then collaborates with Elasticsearch and Kibana to be able to display network traffic data.

### *3.1.1.5 Wireshark System Implementation*

Wireshark is used as a tool for analyzing network traffic, and it can check any packets that enter the system and can also find out where the packet came from and its destination with an easy-to-understand display so that it can quickly take action against intrusion attempts. Because of its important function, the system is used as a tool for conducting Live Forensics actions. The status that the Logstash system has been running is as shown in     Figure 11.



Wireshark
Kam 12:01
Capturing from enp0s3
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time         Source        Destination   Protocol Length Info
   1 0.000000000  10.0.2.15     91.189.92.39  TCP      74 51478 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3225412945 TSecr=0 WS=128
   2 0.001295879  10.0.2.15     91.189.92.39  TCP      74 51480 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3225412947 TSecr=0 WS=128
   3 0.002044337  10.0.2.15     91.189.92.39  TCP      74 51482 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3225412947 TSecr=0 WS=128
   4 0.004341956  10.0.2.15     91.189.92.39  TCP      74 51484 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3225412950 TSecr=0 WS=128
   5 0.100054431  91.189.92.39  10.0.2.15     TCP      60 443 → 51478 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
   6 0.100083412  10.0.2.15     91.189.92.39  TCP      54 51478 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
   7 0.100216572  10.0.2.15     91.189.92.39  TLSv1    221 Client Hello
   8 0.100452420  91.189.92.39  10.0.2.15     TCP      60 443 → 51478 [ACK] Seq=1 Ack=168 Win=65535 Len=0
   9 0.100955432  91.189.92.39  10.0.2.15     TCP      60 443 → 51480 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
  10 0.100966772  10.0.2.15     91.189.92.39  TCP      54 51480 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
  11 0.101062429  10.0.2.15     91.189.92.39  TLSv1    221 Client Hello
  12 0.101270733  91.189.92.39  10.0.2.15     TCP      60 443 → 51480 [ACK] Seq=1 Ack=168 Win=65535 Len=0
  13 0.110126696  91.189.92.39  10.0.2.15     TCP      60 443 → 51482 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
  14 0.110153062  10.0.2.15     91.189.92.39  TCP      54 51482 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
  15 0.110285147  10.0.2.15     91.189.92.39  TLSv1    221 Client Hello
  16 0.110498275  91.189.92.39  10.0.2.15     TCP      60 443 → 51482 [ACK] Seq=1 Ack=168 Win=65535 Len=0
  17 0.111826480  91.189.92.39  10.0.2.15     TCP      60 443 → 51484 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
  18 0.111840070  10.0.2.15     91.189.92.39  TCP      54 51484 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
```

**Figure 11. Status of Active Wireshark System**

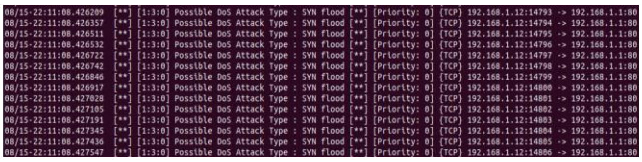
Figure 11 shows that the Wireshark system has been successfully running and is ready to monitor traffic on the network.

## 3.1.2 Detection

This stage is the stage where the system is carried out in the testing phase with Distributed attacks Denial of Service (DDoS) performed by multiple devices

### *3.1.2.1 Detection* With Snort

The first test is carried out to test the attack detection system that is on Snort, this is done to find out whether the configuration carried out can detect anomalies that occur in the network. Display attack detection with a snort as shown in Figure 12.



```
08/15-22:11:08.426209  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14793 -> 192.168.1.1:80
08/15-22:11:08.426357  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14794 -> 192.168.1.1:80
08/15-22:11:08.426511  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14795 -> 192.168.1.1:80
08/15-22:11:08.426532  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14796 -> 192.168.1.1:80
08/15-22:11:08.426722  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14797 -> 192.168.1.1:80
08/15-22:11:08.426742  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14798 -> 192.168.1.1:80
08/15-22:11:08.426846  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14799 -> 192.168.1.1:80
08/15-22:11:08.426917  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14800 -> 192.168.1.1:80
08/15-22:11:08.427028  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14801 -> 192.168.1.1:80
08/15-22:11:08.427105  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14802 -> 192.168.1.1:80
08/15-22:11:08.427191  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14803 -> 192.168.1.1:80
08/15-22:11:08.427345  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14804 -> 192.168.1.1:80
08/15-22:11:08.427436  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14805 -> 192.168.1.1:80
08/15-22:11:08.427547  [**] [1:3:0] Possible DoS Attack Type : SYN flood [**] [Priority: 0] {TCP} 192.168.1.12:14806 -> 192.168.1.1:80
```

**Figure 12. Display of snort detection attack**

Figure 12 shows that the snort system can carry out the monitoring stage and perform the detection stage of anomalies that occur in network traffic to detect an attack.

### *3.1.2.2 Detection With Kibana*

Kibana is a system used to visualize network traffic logs in collaboration with Logstash and Elasticsearch. Figure 13 is a display that the Kibana system can detect that there is an increase in traffic levels in the network.

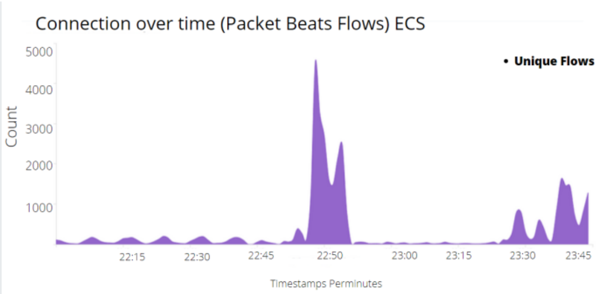
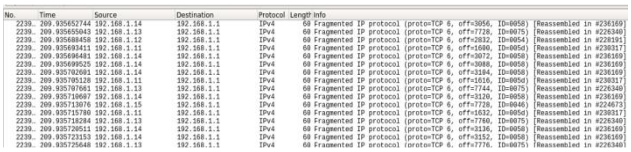


**Figure 13. Display of Attack Detection with Kibana**

Figure 13 shows that there is an unusual increase in traffic on the network and indicates that an attack is taking place.

### *3.1.2.3 Detection With Wireshark*

Wireshark is an application that is used to analyze network traffic in real-time. Display of detection of traffic that occurs in the network as Figure 14.



```
No.     Time          Source        Destination   Protocol Lengtł Info
  2239… 209.935652744 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3056, ID=0058) [Reassembled in #236169]
  2239… 209.935655043 192.168.1.13  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7728, ID=0075) [Reassembled in #226340]
  2239… 209.935688458 192.168.1.12  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=2832, ID=0054) [Reassembled in #228191]
  2239… 209.935693411 192.168.1.11  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=1600, ID=005d) [Reassembled in #230317]
  2239… 209.935696481 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3072, ID=0058) [Reassembled in #236169]
  2239… 209.935699525 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3088, ID=0058) [Reassembled in #236169]
  2239… 209.935702601 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3104, ID=0058) [Reassembled in #236169]
  2239… 209.935705128 192.168.1.11  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=1616, ID=005d) [Reassembled in #230317]
  2239… 209.935707661 192.168.1.13  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7744, ID=0075) [Reassembled in #226340]
  2239… 209.935710607 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3120, ID=0058) [Reassembled in #236169]
  2239… 209.935713076 192.168.1.15  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7728, ID=0046) [Reassembled in #224673]
  2239… 209.935715780 192.168.1.11  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=1632, ID=005d) [Reassembled in #230317]
  2239… 209.935718284 192.168.1.13  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7760, ID=0075) [Reassembled in #226340]
  2239… 209.935720511 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3136, ID=0058) [Reassembled in #236169]
  2239… 209.935723153 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3152, ID=0058) [Reassembled in #236169]
  2239… 209.935725648 192.168.1.13  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7776, ID=0075) [Reassembled in #226340]
```

**Figure 14. Display Detection with Wireshark**

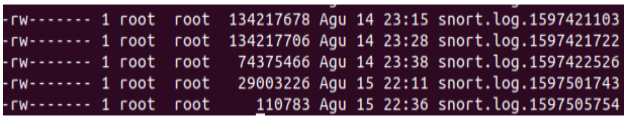
Figure 14 shows that the Wireshark system is showing existing network traffic, to be able to find out information related to attacks.

### 3.1.3 *Collection*

The Collection stage is the stage where data that has been received by various systems and detection sensors are collected for later analysis of the anomalous activity that occurs. The system automatically collects data and saves it into a file.

#### 3.1.3.1 *Collection With Snort*

After the activity is detected by the system, the Snort system will store the activities that occur by storing them in a log. The snort log display is as shown in Figure 15.



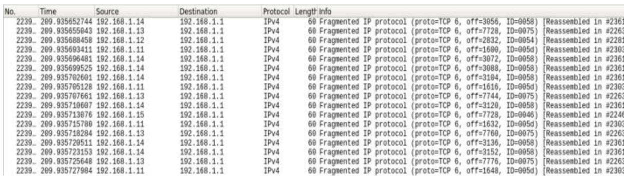
```
-rw------- 1 root  root  134217678 Agu 14 23:15 snort.log.1597421103
-rw------- 1 root  root  134217706 Agu 14 23:28 snort.log.1597421722
-rw------- 1 root  root   74375466 Agu 14 23:38 snort.log.1597422526
-rw------- 1 root  root   29003226 Agu 15 22:11 snort.log.1597501743
-rw------- 1 root  root     110783 Agu 15 22:36 snort.log.1597505754
```

**Figure 15. Display of the Snort Log File**

Figure 15 shows that the Snort system has successfully saved log data into several files that will be used to perform analysis.

#### 3.1.3.2 *Collection With Wireshark*

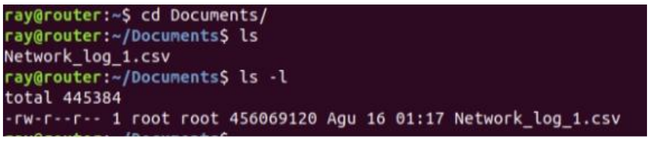
After the activity has been detected, the Wireshark system will immediately display information on every activity that occurs in the network. The display of network traffic data on the Wireshark is as shown in Figure 16.



```
No.     Time          Source        Destination   Protocol Lengtł Info
  2239… 209.935652744 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3056, ID=0058) [Reassembled in #2361
  2239… 209.935655043 192.168.1.13  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7728, ID=0075) [Reassembled in #2263
  2239… 209.935688458 192.168.1.12  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=2832, ID=0054) [Reassembled in #2281
  2239… 209.935693411 192.168.1.11  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=1600, ID=005d) [Reassembled in #2303
  2239… 209.935696481 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3072, ID=0058) [Reassembled in #2361
  2239… 209.935699525 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3088, ID=0058) [Reassembled in #2361
  2239… 209.935702601 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3104, ID=0058) [Reassembled in #2361
  2239… 209.935705128 192.168.1.11  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=1616, ID=005d) [Reassembled in #2303
  2239… 209.935707661 192.168.1.13  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7744, ID=0075) [Reassembled in #2263
  2239… 209.935710607 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3120, ID=0058) [Reassembled in #2361
  2239… 209.935713076 192.168.1.15  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7728, ID=0046) [Reassembled in #2246
  2239… 209.935715780 192.168.1.11  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=1632, ID=005d) [Reassembled in #2303
  2239… 209.935718284 192.168.1.13  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7760, ID=0075) [Reassembled in #2263
  2239… 209.935720511 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3136, ID=0058) [Reassembled in #2361
  2239… 209.935723153 192.168.1.14  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=3152, ID=0058) [Reassembled in #2361
  2239… 209.935725648 192.168.1.13  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=7776, ID=0075) [Reassembled in #2263
  2239… 209.935727984 192.168.1.11  192.168.1.1   IPv4      60 Fragmented IP protocol (proto=TCP 6, off=1648, ID=005d) [Reassembled in #2303
```

**Figure 16. Display Log Files From Wireshark**

Figure 16 shows that the Wireshark system has succeeded in storing log data which will be used for analysis. An example of a file reading the network traffic on the device is shown in Figure 17.



```
ray@router:~$ cd Documents/
ray@router:~/Documents$ ls
Network_log_1.csv
ray@router:~/Documents$ ls -l
total 445384
-rw-r--r-- 1 root root 456069120 Agu 16 01:17 Network_log_1.csv
```

**Figure 17. Display Log Files From Wireshark**

Figure 17 shows that the results of the log storage from Wireshark have been successfully saved, and are stored in the form of a CSV file. Henceforth, it is used for back-up media in case of errors in the analysis of other files.

#### 3.1.3.3 *Collection With Kibana*

Every activity that occurs in network traffic can be read directly by the Logstash system and then the indexing stage is carried out and then sent to Kibana to then visualize the activities that occur in the network. The display of how the Kibana system is as shown in Figure 18.

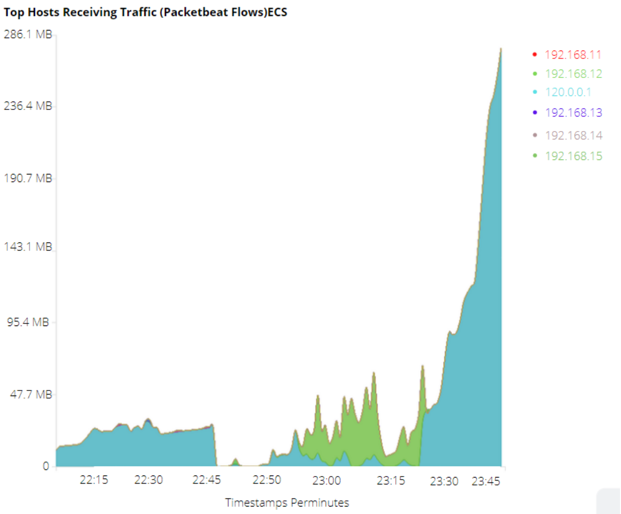


**Figure 18. Display Log Files From Kibana**

Figure 18 shows that Kibana has successfully displayed unusual network traffic data that comes from data obtained by logstash and elasticsearch.

### 3.1.4 *Preservation*

At this stage the data that has been collected and stored must be carried out in the hashing stage or giving identity to the data, then the back-up stage is carried out to the storage media so that the data remains safe in case of errors in data analysis. Stages this is done on any log data stored from Wireshark, snort, and logstash.

### 3.1.5 *Examination*

At this stage, the data that has been collected and obtained is carried out at the stage of checking whether the data is related to the incident that occurred. At this stage, check every log data that has been stored by each detection system.

### 3.1.6 *Analysis*

#### 3.1.6.1 *Analysis with Kibana*

The Kibana system is a system that functions to visualize incoming logs to get analyzed. To know the graph of the increase in the Kibana system as shown in Figure 19.

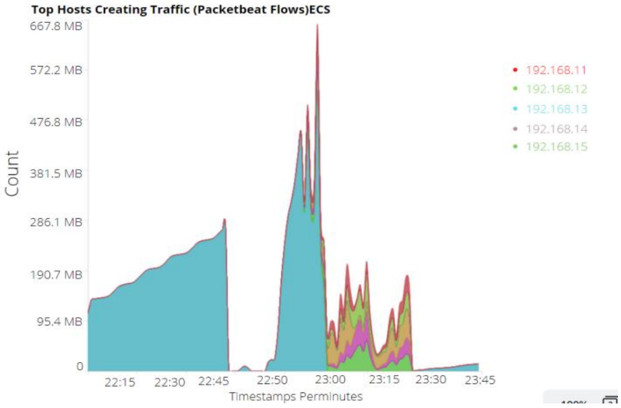


**Figure 19. Display of Network Traffic**

Figure 19 shows that there are IP addresses from outside which increase network traffic significantly. The IP address list is as in table 2.

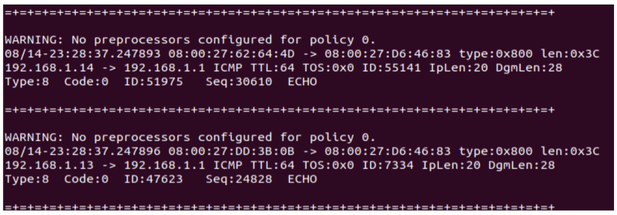
**Table 2. List Attacker of IP Address**

| No. | IP Address |
|---|---|
| 1 | 192.168.1.11 |
| 2 | 192.168.1.12 |
| 3 | 192.168.1.13 |
| 4 | 192.168.1.14 |
| 5 | 192.168.1.15 |

Table 2 shows that the system has successfully stored information related to the IP address used by the attacker in carrying out the attack.

### *3.1.6.2 Analysis with Log Snort*

The snort log analysis stage is carried out by opening the snort log file that has been previously saved. The results of the Snort log are as shown in Figure 20.



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

WARNING: No preprocessors configured for policy 0.
08/14-23:28:37.247893 08:00:27:62:64:4D -> 08:00:27:D6:46:83 type:0x800 len:0x3C
192.168.1.14 -> 192.168.1.1 ICMP TTL:64 TOS:0x0 ID:55141 IpLen:20 DgmLen:28
Type:8  Code:0  ID:51975   Seq:30610  ECHO

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
08/14-23:28:37.247896 08:00:27:DD:3B:0B -> 08:00:27:D6:46:83 type:0x800 len:0x3C
192.168.1.13 -> 192.168.1.1 ICMP TTL:64 TOS:0x0 ID:7334 IpLen:20 DgmLen:28
Type:8  Code:0  ID:47623   Seq:24828  ECHO

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

**Figure 20. Display of Log Snort Analysis**

Figure 20 shows that after analysis several IP addresses had high traffic levels. The list of IP addresses is shown in table 3.

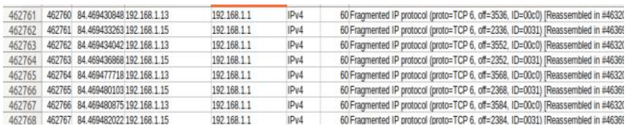
**Table 3. List Attacker of IP Address**

| No. | IP Address | ID | Port | Seq |
|---|---|---|---|---|
| **1** | 192.168.1.11 | 29534 | 80 | 31670 |
| **2** | 192.168.1.12 | 27945 | 80 | 22850 |
| **3** | 192.168.1.13 | 12357 | 80 | 19279 |
| **4** | 192.168.1.14 | 12785 | 80 | 997 |
| **5** | 192.168.1.15 | 28173 | 80 | 54987 |

Table 3 shows that the system has succeeded in storing information related to the attack that occurred and can be used for the analysis phase.

### *3.1.6.3 Analysis with Log Wireshark*

The analysis phase of the Wireshark log is carried out to find information related to the attack. The analysis phase of the Wireshark log is as shown in Figure 21.



```
462761  462760  84.469430848 192.168.1.13  192.168.1.1  IPv4  60 Fragmented IP protocol (proto=TCP 6, off=3536, ID=00c0) [Reassembled in #4632
462762  462761  84.469433263 192.168.1.15  192.168.1.1  IPv4  60 Fragmented IP protocol (proto=TCP 6, off=2336, ID=0031) [Reassembled in #4636
462763  462762  84.469434042 192.168.1.13  192.168.1.1  IPv4  60 Fragmented IP protocol (proto=TCP 6, off=3552, ID=00c0) [Reassembled in #4632
462764  462763  84.469436868 192.168.1.15  192.168.1.1  IPv4  60 Fragmented IP protocol (proto=TCP 6, off=2352, ID=0031) [Reassembled in #4636
462765  462764  84.469477718 192.168.1.13  192.168.1.1  IPv4  60 Fragmented IP protocol (proto=TCP 6, off=3568, ID=00c0) [Reassembled in #4632
462766  462765  84.469480103 192.168.1.15  192.168.1.1  IPv4  60 Fragmented IP protocol (proto=TCP 6, off=2368, ID=0031) [Reassembled in #4636
462767  462766  84.469480875 192.168.1.13  192.168.1.1  IPv4  60 Fragmented IP protocol (proto=TCP 6, off=3584, ID=00c0) [Reassembled in #4632
462768  462767  84.469482022 192.168.1.15  192.168.1.1  IPv4  60 Fragmented IP protocol (proto=TCP 6, off=2384, ID=0031) [Reassembled in #4636
```

**Figure 21. Wireshark Log Analysis**

Figure 21 shows that there is high traffic on devices carried by several IP addresses. The results found in the Wireshark log analysis are that the attack occurred on IPv4 with the TCP protocol with a length of 60 bytes.

### *3.1.7 Investigation*

The investigation stage is the stage for compiling data and identifying data findings. ResultInvestigation is as in table 4.

**Table 4. IP Address Attacker List**

| No. | Timestamp | IP Address | ID | Port | Seq |
|---|---|---|---|---|---|
| **1** | 08 / 14-23: 28: 37 | 192.168.1.11 | 29534 | 80 | 31670 |
| **2** | 08 / 14-23: 28: 37 | 192.168.1.12 | 27945 | 80 | 22850 |
| **3** | 08 / 14-23: 28: 37 | 192.168.1.13 | 12357 | 80 | 19279 |
| **4** | 08 / 14-23: 28: 37 | 192.168.1.14 | 12785 | 80 | 997 |
| **5** | 08 / 14-23: 28: 37 | 192.168.1.15 | 28173 | 80 | 54987 |

Table 4 shows that the system has succeeded in storing information related to attacks that occurred and can be used for increased security or as data in court.

### *3.1.8 Presentation*

The Presentation stage is carried out to show the results of the investigation in a language that is easy to understand. The level of attacks carried out is different in the number of each IP address as shown in Figure 22.

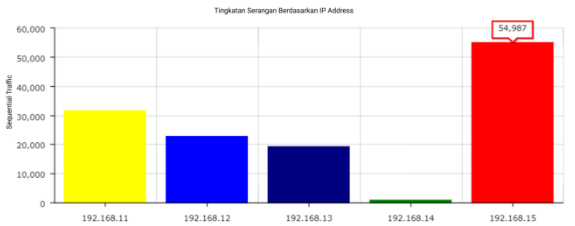


**Figure 22. Graph of Attack Levels Based on IP Address**

Figure 22 shows a graph showing that each IP Address has intensity different attacks and make service disrupted.

### *3.1.9 Result*

The result of live foreansics on the Router can seen in table 5.

| Information | Timestamp | IP Address | ID | Port | Seq |
|---|---|---|---|---|---|
| **Kibana** | ✓ | ✓ | | | |
| **Snort** | ✓ | ✓ | ✓ | ✓ | ✓ |
| **Wireshark** | ✓ | ✓ | | ✓ | |

Based on tabel 5, the results of the study almost get all the information about DDOS attacks.

## 4. CONCLUSION

Based on the results of the research carried out, it can be concluded that the steps taken to analyze Log data carried out using the Wireshark, Snort, Kibana, Elasticsearch, and Logstash tools can effectively detect and store Log data, then the Live Forensic process by utilizing the Network Forensic Generic Process Model. which is done to retrieve data can be effective and safe in carrying out the forensic process by producing some data and graphics, and the results of the analysis in this study produce information, including who is the attacker's IP address, where is the destination port of the attack, when is the time the occurrence attacks, where the destination IP Address is carried out by the attack and how large the sequence of attacks is received.

## 5. REFERENCES

[1] Praseed, A. and Thilagam, PS (2018) 'DDoS Attacks at the Application Layer: Challenges and Research Perspectives for Safeguarding Web Applications', (c), pp. 1–26. doi: 10.1109 / COMST.2018.2870658 ..

[2] Imperva (2016) 'Global DDoS Threat Landscape', Global DDoS Threat Landscape Q1 2016. Available at: https://www.incapsula.com/ddos-report/ddos-report-q1-2016.html.

[3] santhi, BVP, Kanakam, P. and Hussain, SM (2017) 'Cyber Forensic Science to Diagnose Digital Crimes- A study', International Journal of Computer Trends and Technology, 50 (2), pp. 107–113. doi: 10.14445 / 22312803 / ijctt-v50p119.

[4] Kolhe, M. and Ahirao, P. (2017) 'Live Vs Dead Computer Forensic Image Acquisition', International Journal of Computer Science and Information Technologies, 8 (3), pp. 455–457.

[5] Francisco, ARL (2016) Live Vs Dead Computer Forensic Image Acquisition, Journal of Chemical Information and Modeling. doi: 10.1017 / CBO9781107415324.004.

[6] Hambali, A. and Nurmiati, S. (2018) 'Implementation of Intrusion Detection System (IDS) on PC Server Security Against Flooding Data Attacks', 28 (1), pp. 35–43.

[7] Ahmed, AA (2017) 'Investigation approach for network attack intention recognition', International Journal of Digital Crime and Forensics, 9 (1), pp. 17–38. doi: 10.4018 / IJDCF.2017010102.

[8] Baishya, RC, Hoque, N. and Bhattacharyya, DK (2017) 'DDoS attack detection using unique source IP deviation', International Journal of Network Security, 19 (6), pp. 929–939. doi: 10.6633 / IJNS.201711.19 (6) .09.

[9] Pawade, AB and Waghmode, PST (2017) 'Denial-Of-Service Attack Detection Using Artificial Neural Network Based On Genetic Algorithm and Multivariate CorrelationAnalysis', International Journal of Innovative Research in Science, Engineering and Technology, pp. 13055–13062. doi: 10.15680 / IJIRSET.2017.0607097.

[10] Agrawal, S. and Singh Rajput, R. (2017) 'Denial of Services Attack Detection using Random Forest Classifier with Information Gain', International Journal of Engineering Development and Research, 5 (3), pp. 929–938. Available at: www.ijedr.org.

[11] Mazdadi, MI, Riadi, I. and Luthfi, A. (2017) 'Live Forensics on RouterOS using API Services to Investigate Network Attacks', International Journal of Computer Science and Information Security (IJCSIS), 15 (2), pp. 406–410.

[12] Ali, SHA et al. (2016) 'A neural network model for detecting DDoS attacks using darknet traffic features', Proceedings of the International Joint Conference on Neural Networks, 2016-Octob (November 2014), pp. 2979–2985. doi: 10.1109 / IJCNN.2016.7727577.

[13] Mualfah, D. and Riadi, I. (2017) 'Network Forensics For Detecting Flooding Attack On Web Server', IJCSIS) International Journal of Computer Science and Information Security, 15 (2), pp. 326–331. doi: 10.1016 / j.ecss.2004.08.013.

[14] Mazdadi, MI, Riadi, I. and Luthfi, A. (2017) 'Live Forensics on RouterOS using API Services to Investigate Network Attacks', International Journal of Computer Science and Information Security (IJCSIS), 15 (2), pp. 406–410.

[15] Onik, MMH et al. (2018) 'A Novel Approach for Network Attack Classification Based on Sequential Questions', Annals of Emerging Technologies in Computing, 2 (2), pp. 1–14. doi: 10.33166 / aetic.2018.02.001.

[16] Muhammad, AW, Riadi, I. and Sunardi, S. (2017) 'DDoS Attack Detection Using Neural Networks with Fixed Moving Average Window Function', JISKA (Sunan Kalijaga Informatics Journal), 1 (3), p. 115.doi: 10.14421 / jiska.2017.13-03.

[17] Mazdadi, MI, Riadi, I. and Luthfi, A. (2017) 'Live Forensics on RouterOS using API Services to Investigate Network Attacks', International Journal of Computer Science and Information Security (IJCSIS), 15 (2), pp. 406–410.

[18] Hambali, A. and Nurmiati, S. (2018) 'Implementation of Intrusion Detection System (IDS) on PC Server Security Against Flooding Data Attacks', 28 (1), pp. 35–43.

[19] Mualfah, D. and Riadi, I. (2017) 'Network Forensics For Detecting Flooding Attack On Web Server', IJCSIS) International Journal of Computer Science and Information Security, 15 (2), pp. 326–331. doi: 10.1016 / j.ecss.2004.08.013.

[20] Brian Sak, JRR (2016) Mastering Kali Linux Wireless Pentesting, Mastering Kali Linux Wireless Pentesting. Available at: http://apprize.info/linux/kali/9.html.

[21] Hamilton, J. et al. (2018) 'SCADA Statistics monitoring using the elastic stack (Elasticsearch, Logstash, Kibana)', 16th Int. Conf. on Accelerator and Large Experimental Control Systems, pp. 451–455. doi: 10.18429 / JACoW-ICALEPCS2017-TUPHA034.

[22] Hariharan, A., Gupta, A., & Pal, T. (2020, March). CAMLPAD: Cybersecurity Autonomous Machine Learning Platform for Anomaly Detection. In *Future of Information and Communication Conference* (pp. 705-720). Springer, Cham.

[23] Kulkarni, J., Joshi, S., Bapat, S., & Jambhali, K. (2020). Analysis of System Logs for Pattern Detection and Anomaly Prediction. In *Proceeding of International Conference on Computational Science and Applications* (pp. 427-436). Springer, Singapore.