Network Forensic on Distributed Denial of Service Attacks using National Institute of Standards and Technology Method

Arifaleo Nurdin
Department of Informatics
Universitas Ahmad Dahlan

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

ABSTRACT

Router hardware is a network device that can be used to connect multiple networks, either the same network or different networks. One of the attacks that can be carried out on a router is a Distributed Denial of Service attack, an attack that is carried out by consuming available resources, this can make the function of a computer or server not run properly and can make it difficult for other parties. to gain access to services from the attacked computer. The mechanism that can be used is a network forensic mechanism as evidence that can later ensnare criminals. The object of this research is a router network device. The data collection method used is literature study and simulation of DDoS attacks on routers. The research process was carried out using the National Institute of Standards and Technology method. NIST methods include Collection, Examination, Analysis, and Reporting. The system used is the Intrusion Detection System using Snort as a detection sensor, using the Basic Analysis System Engine application, and analyzing attack log files to obtain digital evidence. The results of the study prove that the detection system built using snort 100% can detect DDoS attacks on routers, based on the analysis process, DDoS attacks can make the CPU Load on the router increase up to 100% and the capacity of Free Memory before the attack of 40.0 MB is reduced to 37, 6 MB in no time. The attack information found in the form of attack time information, the source of the attack, the purpose of the attack, what attack was carried out, and the number of attack packets sent, this information can be used as evidence that there is a DDoS attack on the router network device.

Keywords

DDoS, IDS, Network Forensics, NIST, Snort.

1. INTRODUCTION

One of the most important devices in a wide area network is the router. The main target of intruders by monitoring data traffic to enter the main system is the transfer of information resources between router networks, this can be used to damage, delete, and steal sensitive data stored in the main system. This can be very dangerous. to individuals, companies and institutions[1]. Distributed Denial of Service is an attack crime that has recently occurred frequently. In the first quarter of 2020, the number of DDoS attacks increased by more than 278% compared to the previous quarter in 2019 and 542% higher compared to the fourth quarter of 2019. The COVID-19 pandemic is also believed to be a factor causing this increase, this is due to The COVID-19 pandemic has forced many workers to work from home, thus making cybercriminals attack by flooding ISPs (internet service providers) that provide internet [2].The AWS Shield enterprise cybersecurity system successfully warded off multiple threats in the first quarter (Q1) 2020. A 2.3 terabit per second (Tbps) DDoS attack occurred in February 2020. This attack is 44% larger than any previous attack ever detected by the company. This attack caused an increase in attack threats for 3 days a week in February 2020. After the attack was detected, further attacks but not too large, the data can be seen as shown in Figure 1[3].

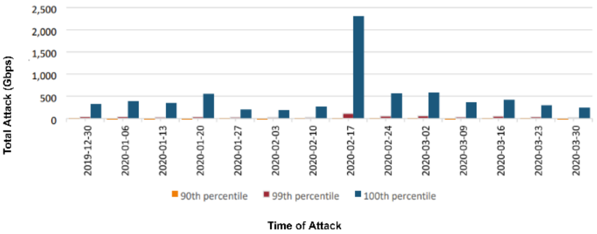


Figure 1. DDoS Attack Data Successfully Countered by AWS Shield Enterprise Cybersecurity System

P 90, P 99, and P 100 volumetric events measured in gigabits per second (Gbps) were observed by AWS in Q1[3]. Detection of suspicious activity in a system or network can be done using an application or software Intrusion Detection System [4]. The steps that must be taken in the search for evidence of the crime are needed, one of which is the National Institute of Standards and Technology method. NIST is a method of analyzing digital evidence, in this method, there are several steps including Collection, Examination, Analysis, and [5].

1.1 Study Literature

1.1.1 Previous Study

This study refers to five previous studies conducted to compare the current study with previous studies. The first research is entitled "Router Forensic Analysis to Detect Distributed Danial of Service (DDoS) Attacks in Real-Time". This research aims to analyze Distributed Danial of Service attacks in real-time by utilizing an attack detection system using Intrusion Detection System on a forensics router using the Ubuntu operating system to be able to find and collect digital evidence. Distributed Danial of Service attacks are carried out by simulating attacks on routers, when an attack is detected it will immediately send a notification to the WhatsApp application, the results of this study expect that the analysis process goes well by utilizing the intrusion detection system and administrators can find digital evidence of a Distributed attack. Denial of Service onrouter [1].

The second research entitled "Implementing Logs in Router Forensics against Distributed Denial of Service (DDoS) Attacks". The research was conducted by designing and building a system that is useful for detecting Distributed Denial of Service attacks by utilizing snort and data traffic data contained in routers stored in logs. The research was conducted using a forensic method which has stages of Collection, Examination, Analysis, and Reporting. The test is carried out by simulating a Distributed Denial of Service attack on the router and then it is produced that by utilizing the Basic Analysis and Security Engine and also using the Wireshark tool to carry out the analysis process and using the rules that exist in the Snort Distributed Denial of Service attack can be detected and this is evidence basis of attack[4].

The third study entitled "Analysis of Evidence of Address Resolution Protocol Spoofing Attacks Using the National Institute of Standard Technology Method". This research aims to analyze and prove the existence of Address Resolution Protocol (ARP) Spoofing attacks using the National Institute of Standards and Technology method which has the stages of Collection, Examination, Analysis, and Reporting. This study carried out 2 attack simulations, the first attack was carried out on laptop devices and the second simulation was carried out on router board devices that were connected to the network. The results of the research simulation obtained 2 attacks with the results of MAC address information from the attacker and victim and the time of the attack. Based on the test results, all Address Resolution Protocol (ARP) Spoofing attacks on the network can be proven with a 100% success rate[5].

The fourth study is entitled "A Comparative Study of Forensic Tools for WhatsApp Analysis using NIST Measurements". This study aims to evaluate the performance of existing forensic tools in handling cases related to android smartphones and WhatsApp using the parameters of the National Institute of Standards and Technology. This study simulates the use of WhatsApp using a smartphone, the simulation carried out is the use of WhatsApp in general such as sending messages, making calls, and receiving pictures. Based on the results of research Belkasoft has the highest index number of 88.23% followed by Oxygen Forensic with an index number of 82.35% and WhatsApp DB / Key Extractor getting an index number of 23.52%, this is because WhatsApp Key / DB Extractor is less met the National Institute of Standards and Technology (NIST) parameter criteria, but WhatsApp Key / DB Extractor managed to retrieve text message artifacts, WhatsApp contact lists, and WhatsApp call logs using logical acquisition and also has a cost advantage because it is an open-source forensic tool[6].

The fifth research entitled "Honeypot Analysis and Implementation in Detecting Distributed Denial Of Service (DDoS) Attacks on Wireless Networks". This study aims to analyze and implement a honeypot to detect Distributed Denial of Service attacks. Implementation is done by combining honeypot with IP tables on the local network. Iptables can be used as a firewall that can deflect Distributed Denial of Service attacks to unused IP addresses. Simulation attacks are carried out with attacks such as host scanning, Denial of Service, and Distributed Denial of Service. The results of the experiments carried out in this study honeypot succeeded in detecting attacks carried out by Netscan android in scanning hosts on the network, then in the Distributed Denial of Service attack experiment using Loic tools, the average value before the CPU load attack was 15 .25% and after the attack, the CPU load was 45.98% and after the deflection of the attack, the CPU load was 30.83% [7].

1.1.2 Network Forensics

Network forensics is a closely related subset of digital forensics, using the monitoring and analysis of computer network traffic to gather information, legal evidence, or intrusion detection[1]. The forensic phase is carried out by recording and analyzing recorded data traffic and other events on the network[8].

1.1.3 Router

A smart router means that it knows where the route information (packet) goes either to another host on the network or to another network[9].For example, connecting two computer networks with different IP classes, so if network A uses IP 192.168.1.2/24 (class C) and network B uses IP 10.127.11.22/16 (class A), with a router that acts like a bridge located in the middle then the two networks will be connected[4].

1.1.4 Distributed Denial of Service(DDoS)

Distributed Denial of Service (DDoS) attacks are attacks that occur on computer networks, always causing computers or servers to run out of resources due to too many requests for packets[10]. DDoS attacks can be divided into 3 main categories, the main categories are as described below[11]:

1. Volume Based Attack
   These attacks include ICMP Flood, UDP Flood, and other fake packet attacks. The main goal of the attacker is to consume the bandwidth of the target site. The magnitude of the attack is measured in bits per second (Bps).
2. Protocol-Based Attacks
   These attacks include SYN Flood, fragmented packet attack, ping of death, smurf attack, and so on. The main goal of the attacker is to consume the actual server resources, such as firewalls. The magnitude of the attack is measured in packets per second.
3. Application-Based Attacks
   These attacks include attacks such as low-and-slow rate attacks, GET/POST floods, attacks targeting Apache, Windows, or open BSD vulnerabilities, and more. Contains seemingly "legitimate" requests, the purpose of this attack is to crash the webserver, and the magnitude of this attack is measured in requests per second.

The types of Distributed Denial of Service attacks include the following:

1. Ping of Death
   Ping of Death is a popular attack. This attack is carried out using the operating system's ping utility. Ping is usually used to determine if a site or website host or IP address exists[10].
2. Syn Flooding
   Syn Flooding is an attack carried out by exploiting protocol weaknesses in the combining process. When two computers decide to start exchanging data, the sending computer (attacker) sends a synchronization packet, and the recipient (target) also responds and then sends the synchronization back to the sender[12].
3. Remote Controled Attack
   A Remote Control Attack is an attack on a target by managing another network. This type of attack usually has a large impact because the server being attacked has a lot of bandwidth. Tools that are commonly used to carry out attacks are master and client or agent types[10]. After remotely accessing the designated master server, the

attacker uses the server to send zombie attack commands (in this case, the client) from the master server. These clients or agents receive commands from the server and then attack according to the attacker's commands[13].

4. User Datagram Protocol (UDP) Flood
   UDP Flood is a type of attack that uses UDP services. In this type of attack, the attacker has a list of broadcast addresses to send fake UDP packets to, and these packets are sent to random ports, unexpectedly retargeting[14]. UDP flood attacks are very simple, many data packets are sent to the victim, and this attack causes the computer to hang because the data packets sent are large[15].UDP Flood has several parameters including source time, destination, protocol destination port, and length in bytes. Parameters are based on the time of delivery, followed by the return address or return address. The packet is then sent to the destination IP address using a specific protocol on the destination port[16].
5. Smurf Attack
   Smurf Attack An attack by utilizing an echo request Internet Protocol Control Massage Protocol (ICMP), often used to broadcast identity data to broadcast addresses on the network. Smurf Attacks are carried out by hackers by spoofing the source IP address. The large number of ICMP requests causes ICMP traffic to crash not only on the intermediary computer network, but also on the victim's network[17].

1.1.5 *Intrusion Detection System*

An intrusion detection system (IDS) is software that can monitor network or system activity and detect malicious activity as it occurs. There are several IDS programs commonly used in networks, including Snort, Suricata, OSSEC, Sagan, Bro, Solar Winds Logs & Event Manager, and Open WIPS[18].IDS can be divided into 2 types, namely:

1. NIDS (Network-based Intrusion Detection System)
   NIDS is a type of IDS that works automatically, capable of monitoring data packets that enter the network system. All data packets running on the network system will be analyzed and then seen whether there is an attack or intrusion into the network system or not[19].
2. HIDS (Host Intrusion Detection System)
   HIDS is a type of IDS that works on individual hosts or certain devices on a computer network system in real-time. HIDS will only monitor data packets when an intrusion occurs[19].

1.1.6 *Snort*

Snort is one of the NIDS programs that can detect intrusion attempts into computer network systems. Snort is open source and licensed under the GNU Universal License, so this software can be used to secure server systems for free (SMB) and can also send alerts from Unix or Linux computers to Microsoft Windows via Server Message Box[20].The mode of snort is as presented in Figure 2[4].

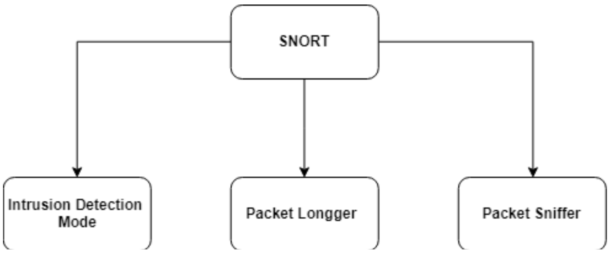


Figure2. Mode Type of Snort

1. PacketSniffer
   This mode is in charge of a monitoring or viewing data traffic on a computer network.
2. PacketLogger
   This mode is in charge of recording all packets that pass on the network and then analyzing them to find evidence in the network forensic process.
3. Intrusion Detection Mode
   In this mode, Snort will detect attacks on the computer network. Using this mode requires the setup of various rules that will distinguish which packets carry an attack and which packets are completely normal.

1.1.7 *Basic Analysis and Security Engine (BASE)*

Basic Analysis and Security Engine (BASE) is the interface used to parse Snort's output. The BASE application is used to intercept attacks on data traffic that is written to log files and stored in the snort database[4].

1.1.8 *Wireshark*

Wireshark is one of the many network analysis tools widely used by network administrators to analyze network and protocol performance. The Wireshark tool interface is preferred by administrators because it uses a graphical user interface (GUI) or graphical display. Wireshark can capture different types of information as it traverses the network, making it easy to analyze packets on the network[10].

1.1.9 *Firewalls*

A firewall is a technology or mechanism implemented to protect hardware, software, or systems. Protection can be implemented by filtering, limiting, or prohibiting one or all relationships between private network segment activity and external networks that are not included in its scope. Segments can be workstations, servers, routers, or LAN networks[21].

1.1.10 *Log*

A log is a file that stores and records computer programming events. Log files can be used as support in the cyber forensics process to obtain digital evidence during the investigative stage. Log cleaning or preprocessing must be done before parsing. It performs pre-processing to remove duplicate data, check for data inconsistencies, and correct data errors such as typos[22].

1.1.11 *National Institute of Standards and Technology (NIST)*

NIST is a frequently used framework because the NIST framework establishes standards, guidelines, and best practices for risk management in all forms of information science and technology[23]. The National Institute of Standards and Technology has several stages in the digital evidence analysis process, namely Collection, Examination, Analysis, and Reporting as presented in Figure 3 [5].

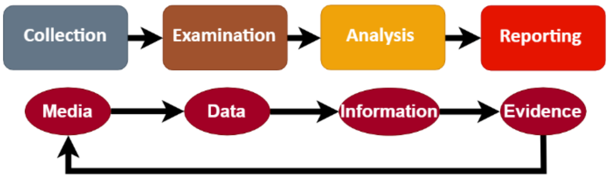


Figure3. National Institute of Standards and Technology proses process flow

The following is an explanation of the stages contained in the National Institute of Standards and Technology (NIST) in the digital evidence analysis process:

1. Collection
   This stage is the collection of data obtained from recording data packets and direct or indirect traffic observations on the network [24].
2. Examination
   The next stage will be the data identification process by conducting tests that can be used as evidence. Once determined, the data will be retrieved and the data retrieval process will be forensically tested [24].
3. Analysis
   The data that has been taken will be analyzed to find things that can be used as evidence, especially computer networks, the thing that will be evident is the Internet Protocol Address[24].
4. Reporting
   The final stage of the NIST method is reporting the results of forensic analysis from beginning to end in the form of a written report so that it can provide recommendations for improvement of policies, guidelines, procedures, tools, and other aspects of the forensic process [24].

2. METHODOLOGY

2.1 Research Scenario

The research scenario begins with building a detection system, namely the Intrusion Detection System using snort on the Linux Ubuntu Bionic operating system which will later detect if there is a DDoS attack on the router. Snort applies the rules that have been created by researchers to detect DDoS attacks. Researchers will act as attackers to test whether the system built can detect DDoS attacks on routers. DDoS attacks are carried out using LOICattack software and also using Command Prompt (CMD)to send attack packets. DDoS attacks are sent in the form of Syn Flooding, UDP Flooding and Smurf Attack.

2.2 Research Stages

2.2.1 Collection

This stage is the stage of collecting data obtained from data packets that have been successfully recorded by the Intrusion Detection System (IDS) through network traffic directly, the data is based on a simulation of a DDoS attack on a router using the LOIC application and also CMD from the Windows 10 operating system. Collection stage carried out is presented in Figure 4.

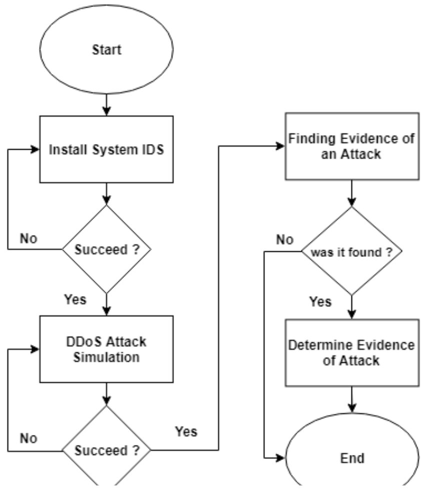


Figure 4.Data collection flowchart

The explanation of Figure 4 is explained as follows:

1. The first process is to install the Intrusion Detection System (IDS) detection system using snort on the ubuntu bionic operating system.
2. The second process is to simulate a DDoS attack on the router using the LOIC and CMD applications from Windows 10.
3. The third process is to find evidence of successful DDoS attack logs stored in the log file in the snort directory.
4. The last process is to determine the evidence of each type of DDoS attack on the router that has been simulated.

The first process is to install the Intrusion Detection System (IDS) detection system using snort on the ubuntu bionic operating system. Then before collecting attack data, the attack is simulated on the router first. Attacks in the form of SYN Flooding attacks on the TCP protocol, Smurf Attacks on the ICMP protocol and UDP Floods on the UDP protocol are sent to the IDS system. The attack simulation scheme is presented in Figure 5.

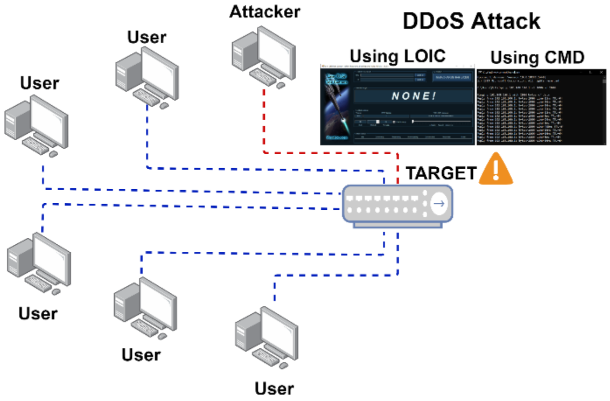
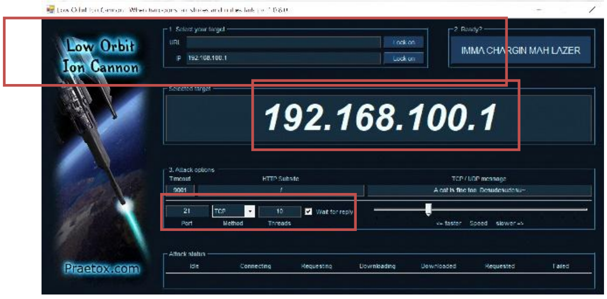


Figure5. DDoS Attack Simulation

Figure 5 shows a DDoS attack on a router carried out using CMD (Command Prompt) and also the Loic application (Low Orbit Ion Cannon) using the Windows 10 operating system. Attack is simulated as follows :

a) Testing Syn Flooding attacks using LOIC software
   Low Orbit Ion Cannon (LOIC) is a network hacker software that is quite easy to use to carry out DDoS attacks. Researchers perform attack simulations by flooding the system with TCP packets. The attack simulation is carried out as shown in Figure 6.



IP 192.168.100.1

Selected target 192.168.100.1

Timeout 9001

HTTP Subsite /

TCP / UDP message A cat is fine too. Desudesudesu~

Port 21

Method TCP

Threads 10

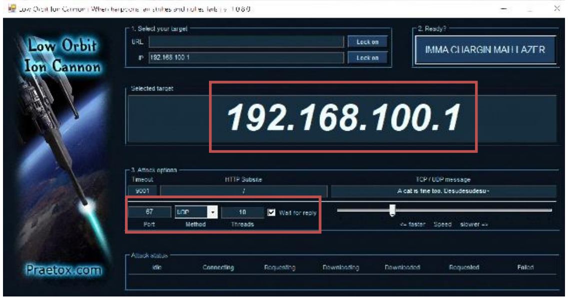
☑ Wait for reply

Figure 6.Sending Syn FloodingPackages using LOIC

Figure 6. shows a simulation of a Syn Flooding DDoS attack using the LOIC application with the IP address 192.168.100.1 which is the router's IP address and uses port 21 and uses the TCP method.

b) Testing UDP Flood attacks with LOIC software

The next DDoS attack simulation is to carry out attacks with the UDP protocol. The simulation is done by flooding the system with UDP packets. The attack simulation was carried out as shown in Figure 7.



IP 192.168.100.1

Selected target 192.168.100.1

Timeout 9001

HTTP Subsite /

TCP / UDP message A cat is fine too. Desudesudesu~

Port 67

Method UDP

Threads 10

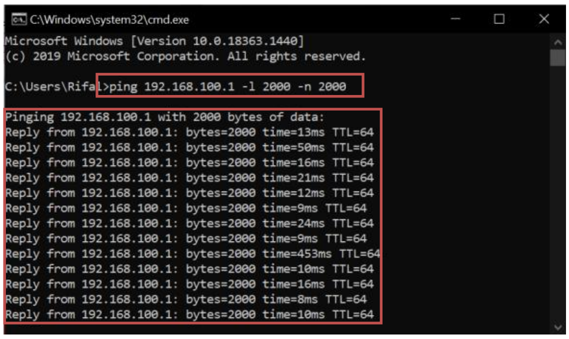
☑ Wait for reply

Figure 7.Sending UDP FloodingPackages using LOIC

Figure 7 shows a simulation of a DDoS UDP Flooding attack aimed at a router device with an IP address of 192.168.100.1 using port 67 and the UDP method.

c) Testing Smurf Attack using CMD

ICMP traffic is usually carried out with the ping command, researchers carry out attacks over a LAN network with the Windows 10 operating system by sending a number of packets as presented in Figure 8.

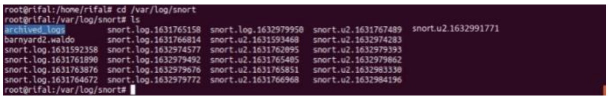


```
Microsoft Windows [Version 10.0.18363.1440]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Rifal>ping 192.168.100.1 -l 2000 -n 2000

Pinging 192.168.100.1 with 2000 bytes of data:
Reply from 192.168.100.1: bytes=2000 time=13ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=50ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=16ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=21ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=12ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=9ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=24ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=9ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=453ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=10ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=16ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=8ms TTL=64
Reply from 192.168.100.1: bytes=2000 time=10ms TTL=64
```

Figure 8.Sending Smurf Attack Packages using CMD

Figure 8 shows a simulation of a DDoS Smurf Attack attack carried out using CMD, the attack is carried out by pinging the IP address of the router device, namely 192.168.100.1 by sending 2000 packets.

After simulating a DDoS attack on a router device, evidence of an attack is generated which is stored in a log file stored in the snort directory. attack log file display as Figure 9.

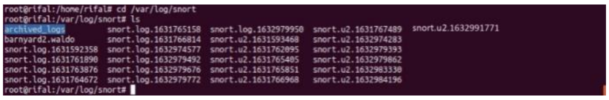


```
root@rifal:/home/rifal# cd /var/log/snort
root@rifal:/var/log/snort# ls
archived_logs        snort.log.1631765158  snort.log.1632979950  snort.u2.1631767489  snort.u2.1632991771
barnyard2.waldo      snort.log.1631766814  snort.u2.1631593468   snort.u2.1632974283
snort.log.1631592358 snort.log.1632974577  snort.u2.1631762095   snort.u2.1632979393
snort.log.1631761890 snort.log.1632979492  snort.u2.1631765405   snort.u2.1632979862
snort.log.1631763876 snort.log.1632979676  snort.u2.1631765851   snort.u2.1632983330
snort.log.1631764672 snort.log.1632979772  snort.u2.1631766968   snort.u2.1632984196
root@rifal:/var/log/snort# 
```

Figure 9. DDoS attack logs contained in Directory Snort

Figure 9 shows the results of the collection of attack logs detected by snort.

Based on the information obtained in the log files collected in the snort directory, the evidence for the 3 attacks that have just been simulated is the last 3 log files. The 3 log files in question are as shown in Figure 10.



```
root@rifal:/home/rifal# cd /var/log/snort
root@rifal:/var/log/snort# ls
archived_logs        snort.log.1631765158  snort.log.1632979950  snort.u2.1631767489  snort.u2.1632991771
barnyard2.waldo      snort.log.1631766814  snort.u2.1631593468   snort.u2.1632974283
snort.log.1631592358 snort.log.1632974577  snort.u2.1631762095   snort.u2.1632979393
snort.log.1631761890 snort.log.1632979492  snort.u2.1631765405   snort.u2.1632979862
snort.log.1631763876 snort.log.1632979676  snort.u2.1631765851   snort.u2.1632983330
snort.log.1631764672 snort.log.1632979772  snort.u2.1631766968   snort.u2.1632984196
root@rifal:/var/log/snort# 
```

Figure 10. Simulated DDoS Attack Proof Log Files

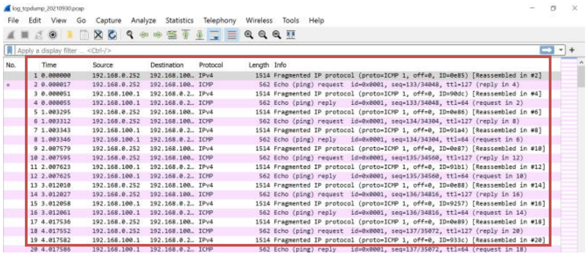
Figure 10 shows 3 DDoS attack log files on a router containing information on simulated attack packets. The log file with file name snort.u2.1632984196 contains information on SYN Flooding attacks, then the log file with file name snort.u2.1632983330 contains information on Smurf Attacks and log files with file name snort.u2.1632991771 contains information on UDP Flooding attacks.

2.2.2 Examination

This stage is done by examining the evidence that has been collected based on the attacks that have been carried out on the router device. An investigation was carried out on the log files that snort managed to save. The first stage is the installation of investigation tools, this is done so that the information that has been collected can be investigated.The next step is to open the attack log file which is done by:

1. Wireshark

The attack log file opened with the wireshark application is a log file generated from the TCPDump application in .pcap format as shown in Figure 11.



```
No.  Time       Source         Destination      Protocol  Length Info
   1 0.000000   192.168.0.252  192.168.100..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=0e85) [Reassembled in #2]
   2 0.000017   192.168.0.252  192.168.100..    ICMP       562 Echo (ping) request  id=0x0001, seq=133/34048, ttl=127 (reply in 4)
   3 0.000051   192.168.100.1  192.168.0.2..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=90dc) [Reassembled in #4]
   4 0.000055   192.168.100.1  192.168.0.2..    ICMP       562 Echo (ping) reply    id=0x0001, seq=133/34048, ttl=64 (request in 2)
   5 1.003295   192.168.0.252  192.168.100..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=0e86) [Reassembled in #6]
   6 1.003312   192.168.0.252  192.168.100..    ICMP       562 Echo (ping) request  id=0x0001, seq=134/34304, ttl=127 (reply in 8)
   7 1.003343   192.168.100.1  192.168.0.2..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=91a4) [Reassembled in #8]
   8 1.003346   192.168.100.1  192.168.0.2..    ICMP       562 Echo (ping) reply    id=0x0001, seq=134/34304, ttl=64 (request in 6)
   9 2.007579   192.168.0.252  192.168.100..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=0e87) [Reassembled in #10]
  10 2.007595   192.168.0.252  192.168.100..    ICMP       562 Echo (ping) request  id=0x0001, seq=135/34560, ttl=127 (reply in 12)
  11 2.007623   192.168.100.1  192.168.0.2..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=91b1) [Reassembled in #12]
  12 2.007625   192.168.100.1  192.168.0.2..    ICMP       562 Echo (ping) reply    id=0x0001, seq=135/34560, ttl=64 (request in 10)
  13 3.012010   192.168.0.252  192.168.100..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=0e88) [Reassembled in #14]
  14 3.012027   192.168.0.252  192.168.100..    ICMP       562 Echo (ping) request  id=0x0001, seq=136/34816, ttl=127 (reply in 16)
  15 3.012058   192.168.100.1  192.168.0.2..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=9257) [Reassembled in #16]
  16 3.012061   192.168.100.1  192.168.0.2..    ICMP       562 Echo (ping) reply    id=0x0001, seq=136/34816, ttl=64 (request in 14)
  17 4.017536   192.168.0.252  192.168.100..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=0e89) [Reassembled in #18]
  18 4.017552   192.168.0.252  192.168.100..    ICMP       562 Echo (ping) request  id=0x0001, seq=137/35072, ttl=127 (reply in 20)
  19 4.017582   192.168.100.1  192.168.0.2..    IPv4      1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=933c) [Reassembled in #20]
  20 4.017586   192.168.100.1  192.168.0.2..    ICMP       562 Echo (ping) reply    id=0x0001, seq=137/35072, ttl=64 (request in 18)
```

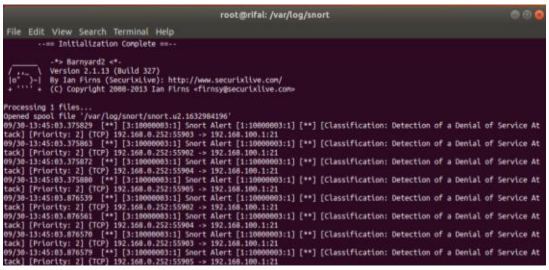
Figure 11. TCPDump Log File Investigation with Wireshark Software

Figure 11 shows the information captured by TCPDump. The information obtained is information based on a simulated DDoS attack.

2. Barnyard2

a) Attack Log File with File Name snort.u2.1632984196.

The opened file is an attack log file stored in the snort directory with the file name snort.u2.1632984196 with the output as Figure 12.

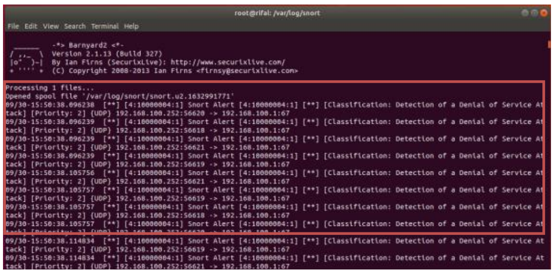


```
root@rifal: /var/log/snort
File  Edit  View  Search  Terminal  Help
      --== Initialization Complete ==--

  ______    -*> Barnyard2 <*-
 / ,,_  \   Version 2.1.13 (Build 327)
 |o"  )~|   By Ian Firns (SecurixLive): http://www.securixlive.com/
 + '''' +   (C) Copyright 2008-2013 Ian Firns <firnsy@securixlive.com>

Processing 1 files...
Opened spool file '/var/log/snort/snort.u2.1632984196'
09/30-13:45:03.375829  [**] [3:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {TCP} 192.168.0.252:55903 -> 192.168.100.1:21
09/30-13:45:03.375863  [**] [3:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {TCP} 192.168.0.252:55902 -> 192.168.100.1:21
09/30-13:45:03.375872  [**] [3:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {TCP} 192.168.0.252:55904 -> 192.168.100.1:21
09/30-13:45:03.375880  [**] [3:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {TCP} 192.168.0.252:55905 -> 192.168.100.1:21
09/30-13:45:03.876539  [**] [3:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {TCP} 192.168.0.252:55902 -> 192.168.100.1:21
09/30-13:45:03.876561  [**] [3:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {TCP} 192.168.0.252:55904 -> 192.168.100.1:21
09/30-13:45:03.876570  [**] [3:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {TCP} 192.168.0.252:55903 -> 192.168.100.1:21
09/30-13:45:03.876579  [**] [3:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {TCP} 192.168.0.252:55905 -> 192.168.100.1:21
```

Figure 12.Log File with Namesnort.u2.1632984196

Figure 12shows the information contained in the log file with the name snort.u2.1632984196, in the file there is information on DDoS attacks that were successfully detected by snort.

b) Attack Log File with File Name snort.u2.1632991771.
The opened file is an attack log file stored in the snort directory with the file name snort.u2.1632991771with the output as Figure 13.

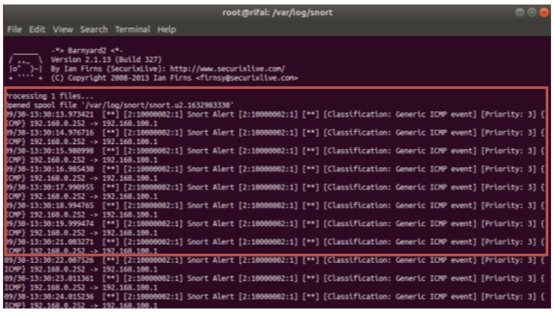


```
root@rifal: /var/log/snort
File  Edit  View  Search  Terminal  Help

  ______    -*> Barnyard2 <*-
 / ,,_  \   Version 2.1.13 (Build 327)
 |o"  )~|   By Ian Firns (SecurixLive): http://www.securixlive.com/
 + '''' +   (C) Copyright 2008-2013 Ian Firns <firnsy@securixlive.com>

Processing 1 files...
Opened spool file '/var/log/snort/snort.u2.1632991771'
09/30-15:50:38.096238  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56620 -> 192.168.100.1:67
09/30-15:50:38.096239  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56618 -> 192.168.100.1:67
09/30-15:50:38.096239  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56621 -> 192.168.100.1:67
09/30-15:50:38.096239  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56619 -> 192.168.100.1:67
09/30-15:50:38.105756  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56621 -> 192.168.100.1:67
09/30-15:50:38.105757  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56619 -> 192.168.100.1:67
09/30-15:50:38.105757  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56618 -> 192.168.100.1:67
09/30-15:50:38.105757  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56620 -> 192.168.100.1:67
09/30-15:50:38.114834  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56619 -> 192.168.100.1:67
09/30-15:50:38.114834  [**] [4:10000004:1] Snort Alert [4:10000004:1] [**] [Classification: Detection of a Denial of Service At
tack] [Priority: 2] {UDP} 192.168.100.252:56621 -> 192.168.100.1:67
```

Figure 13.Log File with Namesnort.u2.1632991771

Figure 13shows the information contained in the log file with the name snort.u2.1632991771, in the file there is information on DDoS attacks that were successfully detected by snort.

c) Attack Log File with File Namesnort.u2.1632983330.
The opened file is an attack log file stored in the snort directory with the file name snort.u2.1632983330with the output as Figure 14.



```
root@rifal: /var/log/snort
File  Edit  View  Search  Terminal  Help

  ______    -*> Barnyard2 <*-
 / ,,_  \   Version 2.1.13 (Build 327)
 |o"  )~|   By Ian Firns (SecurixLive): http://www.securixlive.com/
 + '''' +   (C) Copyright 2008-2013 Ian Firns <firnsy@securixlive.com>

Processing 1 files...
Opened spool file '/var/log/snort/snort.u2.1632983330'
09/30-13:30:13.973421  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:14.976716  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:15.980998  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:16.985430  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:17.990955  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:18.994765  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:19.999474  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:21.003271  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:22.007526  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:23.011361  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
09/30-13:30:24.015236  [**] [2:10000002:1] Snort Alert [2:10000002:1] [**] [Classification: Generic ICMP event] [Priority: 3] {
ICMP} 192.168.0.252 -> 192.168.100.1
```

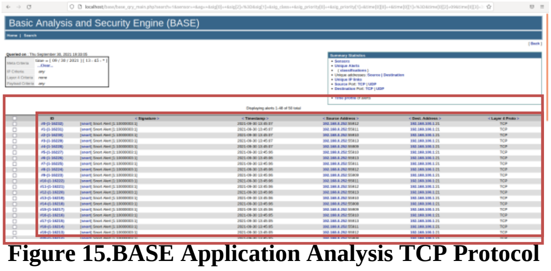
Figure 14.Log File with Namesnort.u2.1632983330

Figure 14shows the information contained in the log file with the name snort.u2.1632983330, in the file there is information on DDoS attacks that have been successfully detected by snort.

3.Basic Analysis and Security Engine (BASE)

The display on the BASE application is directly distinguished based on the type of attack detected by snort, while the log display for each attack is as follows:

a) SYN Flooding Attack Log
Investigation of the DDoS SYN Flooding attack log using the BASE application can be seen as Figure 15.



Basic Analysis and Security Engine (BASE)

Home | Search

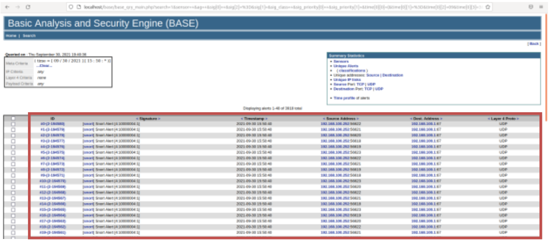
Displaying alerts 1-48 of 50 total

< Signature > | < Timestamp > | < Source Address > | < Dest. Address > | < Layer 4 Proto >

Figure 15.BASE Application Analysis TCP Protocol Attack

Figure 15shows the results of the information from the SYN Flooding attack log that was successfully detected by the IDS system, the information that was successfully obtained is displayed in tabular form. The table shows information in the form of ID, Signature, Time Stamp, Source Address and Layer Protocol containing the simulated DDoS SYN Flooding attack log.

b) UDP Flooding Attack Log
Investigation of the UDPFlooding DDoS attack log using the BASE application can be seen as Figure 16.



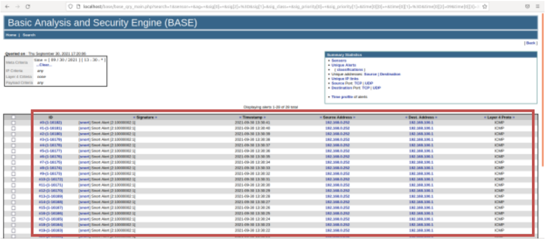
Basic Analysis and Security Engine (BASE)

Home | Search

Figure 16.BASE Application Analysis UDP Protocol Attack

Figure 16shows the results of the information from the UDP Flooding attack log that was successfully detected by the IDS system, the information that was successfully obtained is displayed in tabular form. The table shows information in the form of ID, Signature, Time Stamp, Source Address and Layer Protocol containing the simulated DDoS UDP Flooding attack log.

c) Smurf Attack Log
Investigation of the DDoS Smurf Attack attack log using the BASE application can be seen as Figure 17.



Basic Analysis and Security Engine (BASE)

Home | Search

Figure 17.BASE Application Analysis of ICMP Protocol Attacks

Figure 17shows the results of the information from the Smurf attack log that was successfully detected by the IDS system, the information that was successfully obtained is displayed in tabular form. The table shows information in the form of ID, Signature, Time Stamp, Source Address and Layer Protocol containing the simulated DDoS Smurf attack log.

2.2.3 Analysis

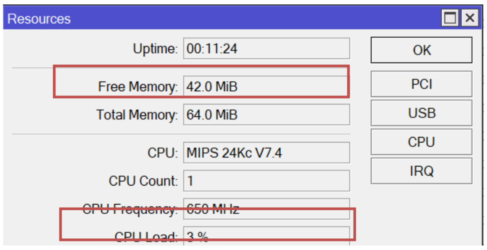
The next process is the analysis process, which is analyzing the digital evidence that has been found based on the previous process. There are several analyzes carried out by the author, some of the analyzes referred to are as follows:

1. Analysis of Research Tools

   The results of the attack log analysis that have been obtained are in the form of simulated DDoS attack log information. The results of the information obtained from several tools have several differences, the difference in the results of this information is only seen in the form of information displayed, in barnyard2 it produces very complete information to display the total attack packets that have been detected, the Wireshark application produces information that is quite complete but the time information difficult to understand, and while the BASE application also produces fairly complete information in the attack log, to find out the number of attacks that have been detected it is necessary to search for attacks based on the type and time of attack.

2. Router Device Analysis

   The state of the router device before the DDoS attack simulation is carried out as shown in Figure 18.

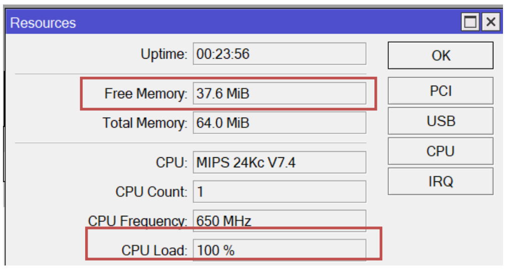


Uptime: 00:11:24
Free Memory: 42.0 MiB
Total Memory: 64.0 MiB
CPU: MIPS 24Kc V7.4
CPU Count: 1
CPU Frequency: 650 MHz
CPU Load: 3 %

Figure 18.State of Router Before DDoS Attack

Figure 18 shows the state of the router before the DDoS attack which shows the amount of Free Memory as much as 42.0 MB, and also the amount of CPU Load as much as 3%.

Then the simulation of a DDoS attack on the router device produces an impact as shown in Figure 19.



Uptime: 00:23:56
Free Memory: 37.6 MiB
Total Memory: 64.0 MiB
CPU: MIPS 24Kc V7.4
CPU Count: 1
CPU Frequency: 650 MHz
CPU Load: 100 %

Figure 19.Router State During DDoS Attack

Figure 19 shows the results of the impact of a simulated DDoS attack on the router device which shows that the amount of Free Memory is reduced, before the attack the amount of Free Memory is 42.0 MB then when the attack is carried out its capacity decreases to 37.6 MB, and also the amount CPU Load before the attack by 3% increased dramatically to 100% and this can cause the network to be disrupted.

2.2.4 Reporting

Reporting is the last stage in the NIST method, the reporting stage aims to present reports in the form of data obtained during the investigation and analysis process, these results will be used as evidence of DDoS attacks on router network devices. Table 1 contains information about DDoS attacks that have been obtained based on the investigation and analysis process.

Table1.DDoS Attack Information based on investigation and Analysis Process

Attack Type / Protocol	Timestamp	Source Address	Dest. Address	Port	Number of Attack Packets
Syn Flooding / TCP	30/09/2021 13:45	192.168.0.252	192.168.100.1	21	50
UDP Flooding / UDP	30/09/2021 15:50	192.168.100.252	192.168.100.1	67	1306
Smurf Attack / ICMP	30/09/2021 13:30	192.168.0.252	192.168.100.1	-	28

Based on Table 1, information is obtained that the detected DDoS attacks are Syn Flooding attacks with TCP protocol, UDP Flooding attacks with UDP protocol, and Smurf Attack attacks with ICMP protocol. The source of the attack comes from the IP addresses 192.168.0.252 and 192.168.100.252 with the destination IP address 192.168.100.1 with different ports and multiple attack packets. The results of the attack information were obtained from several tools used by the author in the analysis process. The results of this report can be used as evidence of a DDoS attack on a router network device.

2.3 Mitigation

Mitigations are steps or a series of activities carried out to reduce risk or prevent the occurrence of an event. One process that can be done to mitigate DDoS attacks on routers can be done by increasing security on the router, increasing security can be done in two ways, namely increasing security in terms of software and hardware[25]. The explanation of each aspect is as described below:

1. Improved Security Using Software

   a) Using Firewall Filter

   Firewall Filter aims to filter incoming and outgoing data packets on the router network device, this can prevent the router from experiencing a down condition. The Firewall Filter script is as follows:

   1) Drop Syn Flood Attack

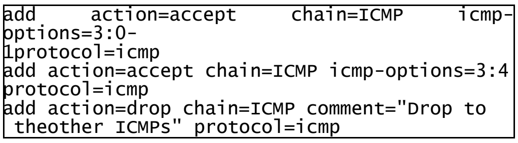
   The Firewall Filter script for Drop Syn Flood attacks is as follows:

```
add action=add-src-to-address-list address
list=syn_flooder
    address-list-timeout=30m chain=input
comment="Drop Syn-Flood IP " connection-
limit=30,3protocol=tcp tcp-flags=syn
add action=drop chain=input src-address-
list=syn_flooder
```

   2) Drop ICMP Flood Attack

   The Firewall Filter script to Drop ICMP Flood attacks is as follows:

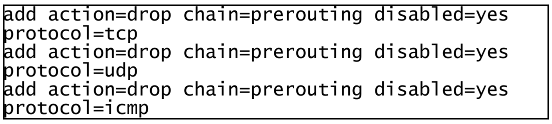
```
add action=jump chain=input comment="ICMP
input,output,forward Flow" jump-
target=ICMP protocol=icmp
add action=jump chain=output jump-target=ICMP
protocol=icmp
add     action=jump     chain=forward     jump-
target=ICMP
protocol=icmp
add action=accept chain=ICMP comment="Allow
NormalICMP Action" icmp-options=8:0
limit= 1,5:packetprotocol=icmp
add action=accept chain=ICMP icmp-options=0:0
protocol=icmp
add     action=accept     chain=ICMP     icmp-
options=11:0
protocol=icmp
```



```
add     action=accept     chain=ICMP     icmp-
options=3:0-
1protocol=icmp
add action=accept chain=ICMP icmp-options=3:4
protocol=icmp
add action=drop chain=ICMP comment="Drop to
 theother ICMPs" protocol=icmp
```

b) Using Firewall Raw

Firewall Raw can be used to block IP Addresses that are suspected of being attackers. The RAW firewall allows us to choose to skip or drop data packets prior to connection tracking, thus saving CPU load. The following Raw Firewall script is written on the CLI terminal to block the attacker's IP.



```
add action=drop chain=prerouting disabled=yes
protocol=tcp
add action=drop chain=prerouting disabled=yes
protocol=udp
add action=drop chain=prerouting disabled=yes
protocol=icmp
```

2. Improved Security Using Hardware

Hardware firewalls can be used to enhance router network security. These devices can intercept data packets and traffic requests before connecting to a network server. Physical tool-based firewalls such as these can ensure malicious traffic from outside the network is stopped before network endpoints are exposed to risk of attack.

3. CONCLUSION

Based on the results of the previous description, it can be concluded that the IDS system built using snort 100% can detect DDoS attacks aimed at router network devices. Based on the analysis process, it turns out that a DDoS attack can make the CPU Load on the router increase up to 100% and the Free Memory capacity before the attack is 40.0 MB reduced to 37.6 MB in a short time. The information obtained in the form of the time of attack, the source of the attack, the purpose of the attack, what attack was carried out, through what port and protocol the attack was carried out and the number of attack packets, this was used as evidence that a DDoS attack had occurred on the router. Further research is recommended to test other types of attacks, add detection rules, and use tools and supporting software with the latest versions, this aims to further improve security on router devices because over time the types of attacks will continue to grow.

4. REFERENCES

[1] F. Ridho, A. Yudhana, and I. Riadi, “Forensic Analysis of Routers to Detect Distributed Danial of Service (DDoS) Attacks in Real Time,” vol. 2, no. 1, pp. 111–116, 2016, [Online]. Available: http://ars.ilkom.unsri.ac.id.

[2] Nexusguard, “Threat Report Distributed Denial of Service (DDoS) Q1 2020,” *Aust. Cyber Secur. Cent.*, 2020, [Online]. Available: https://blog.nexusguard.com/threat-report/ddos-threat-report-2020-q1.

[3] T. L. Report *et al.*, “AWS Shield,” pp. 1–9, 2020.

[4] F. Ridho, A. Yudhana, and I. Riadi, “Implementation of Logs in Router Forensics against Faizin's Distributed Denial of Service (DDoS) Attacks,” *J. TIMES*, vol. 2, no. Desember, pp. 652–657, 2017, doi: 10.1109/ETFA.2003.1248760.

[5] I. Riadi, A. Fadlil, and M. N. Hafizh, “Analysis of Evidence of Address Resolution Protocol Spoofing Attacks using the National Institute of Standard Technology Method,” *Edumatic J. Pendidik. Inform.*, vol. 4, no. 1, pp. 21–29, 2020, doi: 10.29408/edumatic.v4i1.2046.

[6] R. Umar, I. Riadi, and G. Maulana, “A Comparative Study of Forensic Tools for WhatsApp Analysis using NIST Measurements,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 12, pp. 69–75, 2017, doi: 10.14569/ijacsa.2017.081210.

[7] B. Mardiyanto, T. Indriyani, and I. M. Suartana, “Honeypot Analysis and Implementation in Detecting Distributed Denial-Of-Services (DDOS) Attacks on Wireless Networks,” *Integer J.*, vol. 1, no. 2, pp. 32–42, 2016.

[8] A. R. Caesarano and I. Riadi, “Network Forensics for Detecting SQL Injection Attacks using NIST Method,” *Int. J. Cyber-Security Digit. Forensics*, vol. 7, no. 4, pp. 436–443, 2018.

[9] Kristono and I. Riadi, “Simulation for Data Security Improvement in Exploited,” *Int. J. Comput. Sci. Inf. Secur. (IJCSIS)*, vol. 16, no. 5, pp. 6–15, 2018.

[10] M. Purwoko and H. Hilal, “Analysis of the Application of Nftables Firewall as a Server Security System in Virtualization Machines,” *J. Telekomun. dan Komput.*, vol. 9, pp. 1–22, 2019, doi: 10.22441/incomtech.v9i1.5676.

[11] N. Al-munawar and A. Sediyono, “Characteristics of Computer Power Consumption with Changes in the Level of Distributed Denial of Service (Ddos) attacks,” *Semin. Nas. Cendekiawan Ke 3*, pp. 141–147, 2017.

[12] M. Siddik Hasibuan, “The Syn Flooding Threat Analysis Model in Networks,” *J. Teknovasi*, vol. 05, pp. 2540–8389, 2018.

[13] F. Ridho, “Forensic Analysis of Routers Against Distributed Denial of Service (DDoS) Attacks,” Universitas Ahmad Dahan, 2018.

[14] J. Chris, J. Sihombing, D. P. Kartikasari, and A. Bhawiyuga, “Implementation of Distributed Denial of Service (DDoS) Attack Detection and Mitigation Systems using SVM Classifier on Software-Defined Network (SDN) Architecture,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 10, pp. 9608–9613, 2019.

[15] A. H. Hambali and S. Nurmiati, “Implementation of Intrusion Detection System (IDS) on PC Server Security Against Data Flooding Attacks,” *Sainstech J. Penelit. dan Pengkaj. Sains dan Teknol.*, vol. 28, no. 1, pp. 35–43, 2018, doi: 10.37277/stch.v28i1.267.

[16] M. Q. Syahputra, D. R. Akbi, and D. Risqiwati, “DDoS Attack Detection and Mitigation in Software Defined Networks Using Decision Tree Algorithms,” *J. Repos.*, vol. 2, no. 11, p. 1491, 2020, doi: 10.22219/repositor.v2i11.795.

[17] H. E. Wahanani, B. Nugroho, and G. I. Prakoso, “Analysis of Smurfs and Ping of Death Attacks Using the Support Vector Machine (Svm) Method,” *Anal. Smurfs Attack And Ping Death With Method. Support Vector Mach. (Svm)*, 2016.

[18] Lukman and M. Suci, “Comparative Analysis of Snort and Suricata Performance as Intrusion Detection System

in Detecting Syn Flood Attacks on Apache Web Server," *J. Teknol. Inf.*, vol. XV, no. 2, pp. 6–15, 2020.

[19] W. W. Purba and R. Efendi, "Design and analysis of computer network security system using SNORT," *Aiti*, vol. 17, no. 2, pp. 143–158, 2021, doi: 10.24246/aiti.v17i2.143-158.

[20] V. Prisscilya and T. Santoso, "Implementation Of Network Security Using Intrusion," *J. Inf. Technol.*, vol. 6, pp. 1–8, 2021.

[21] M. Suyuti Ma'sum, M. Azhar Irwansyah, and H. Priyanto, "Comparative Analysis of Network Security Systems Using Snort and Netfilter," *J. Sist. dan Teknol. Inf.*, vol. 5, no. 1, pp. 56–60, 2017.

[22] M. Zulfadhilah, Yudi Prayudi, and I. Riadi, "Cyber Profiling using Log Analysis and K-Means Clustering A Case Study Higher Education in Indonesia," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 7, pp. 430–435, 2016, [Online]. Available: http://thesai.org/Downloads/Volume7No7/Paper_59-Cyber_Profiling_Using_Log_Analysis_And_K_Means_Clustering.pdf.

[23] L. Arsada and H. Pembahasan, "Application of the NIST Method for Analysis of Denial of Service (DoS) Attacks on Internet of Things (IoT) Devices," *J. Ilm. KOMPUTASI*, vol. 20, pp. 275–281, 2021.

[24] Firmansyah, A. Fadlil, and R. Umar, "Identification of Forensic Evidence for Virtual Router Networks Using the NIST Method," *Resti*, vol. 1, no. 1, pp. 19–25, 2017.

[25] B. Jaya, Y. Yuhandri, and S. Sumijan, "Improved Mikrotik Router Security Against Denial of Service (DoS) Attacks," *J. Sistim Inf. dan Teknol.*, vol. 2, pp. 115–123, 2020, doi: 10.37034/jsisfotek.v2i4.32.