

# Building A Firewall And Intrusion Detection System Dased Network Security System Using Opnsense Tools

Jalal Sami Majid

Department of Artificial Intelligence, Faculty of Engineering, Shahid Chamran University of Ahvaz, Ahvaz, Iran

## Article Info

### Article history:

Received February 2, 2025

Revised March 15, 2025

Accepted March 18, 2025

### Keywords:

Firewall

Intersion DetectionSystem(IDS)

OPNSense Tools

Security

## ABSTRACT

Computer networks are a crucial element in the evolution of information technology, because all aspects in the realm of information technology require computer networks as a medium of communication between users of the technology. OPNSense will act as a link between the internet and the Ubuntu client, which will serve as a firewall and Detection Instruction System (IDS) provider. Implementing a Firewall and Intrusion Detection System (IDS) with OPNSense Tools can be an effective solution for server security and preventing unauthorised attacks. During scanning, DDOS testing, and sniffing, the system can record attack logs, send attack notifications, protect against attacks, and test URL filters on websites. According to the proposed system topology, Virtual OPNSense will connect to the internet via Virtual Kali Linux via a NAT Network adapter, Virtual OPNSense will forward data to the server via a configured Host Only Network network adapter, and the server will include a Web server service. In Virtual OPNSense, the Firewall, Intrusion Detection System, and Webfilter will all be configured. Firewalls and intrusion detection systems (IDS) will keep servers safe by preventing attacks and recording attack logs. Then Kali Linux will run port scanning and DDoS attacks. The results of the current study were concluded as a series of tests from Building Systems, Firewall, and IDS-Based Network Security. OPNSense Tools can prevent clients from accessing specific predetermined websites, monitor via the log file menu, and block DoS attacks, but it cannot record attack logs

## Corresponding Author:

Hayder Naser Khraibet

Department of Computer Science, Shatt Al-Arab University College, Basra, Iraq

Email: hayderkhraibet@sa-uc.edu.iq

## 1. INTRODUCTION

The use of computer networks to search for information and communicate has experienced rapid development at this time. Computer networks are a crucial element in the evolution of information technology because all aspects of the realm of information technology require computer networks as a medium of communication between users of the technology [1]. Computer networks also act as access channels to various websites connected to a web server in a computer network. However, the use of this network cannot be separated from potential threats that may arise from other computer network users. Therefore, we need a system that can prevent attacks on the integrity of computer networks [2]. Many server defence systems still depend manually on the administrator, making system integrity dependent on the administrator's availability and speed in responding to disturbances that occur [3]. If the disturbance has succeeded in causing the server to go down or the network to malfunction, the administrator will no longer be able to access the system remotely.

So administrators cannot quickly restore the network. Administrators need a system that can provide optimal information about threats that occur and can be resolved quickly. This will speed up the process of troubleshooting and system recovery [4]. Suricata is an IDS that can detect threat attack activities on a network assisted by existing rules. The way Suricata works is that when there is an attack, Suricata will check existing packages/attacks through the rules created. When an attack is detected, Suricata will create a log of the attacks carried out [5].

By default, the security system on web servers in the network is still very dependent on administration, so the security of a server is closely related to an administrator's responsibility for possible disturbances. In this situation, the current system can cause difficulties for administrators when a serious disruption of the web server occurs, resulting in downtime or inability of network connections, which in turn can slow down the server recovery process. Therefore, an administrator needs a system that can help monitor and provide immediate information when interference or threats to the web server are detected. The system is also expected to be able to take preventative action against identified disturbances or threats. Firewall: Firewall is software that blocks entry through public networks. It can even be a Modem or Router that incorporates ACL policy between the two networks [6]. The definition of Firewall presented by Oloyede (2021) [9] describes the Firewall as a point between two or more networks, which can be a single component or a set of components, through which all traffic passes, thus allowing control, authentication, and a record of all traffic. A firewall is not only used to protect a private or public network; its applications can also be used in a corporate environment to separate work groups or subnets. For Ingham (1994) [10], Firewalls can be classified into three categories: packet filters, state filters, and application gateways. IDS/IPS: An intrusion detection system (Intrusion Detection System - IDS) has a mechanism whose main objective is to detect suspicious, inappropriate, or incorrect activities or attempted intrusions in computer networks, becoming an essential element in a corporate environment. The IDS works as an alarm system against intrusions and can perform detection based on some types of knowledge, such as signatures or behavior, with signature-based being the most common [9]. • - Knowledge-based (also called signatures) analyzes activity for known attack or intrusion patterns. • - Behavior-based: detects deviations from normality in the behavior of users or groups of users. This type of detection has the main advantage of being able to detect new forms of attacks and intrusions. IDS systems that capture traffic only for analysis operate in passive mode, meaning it is not possible to manage packet traffic on the network. According to Korcak et al. (2014) [11], Wireless sensor networks have been the subject of study for some time, mainly due to technological innovations introduced by the advance in micro-electro-mechanical systems, wireless communications, and digital electronics [14]. In the last decade there has been widespread fast and sophisticated technical networks the wireless different was the use of various techniques and are local networks of the various threats to the growing increasingly been used encryption and that the protection is being from entering the local network and the protection of user and tampering with the computer and find out local area networks has become a major tool to many companies and factories [15].

## 2. OBJECTIVES

- i. How do you create effective rules to capture the threat of server attacks with Suricata in the form of a log file?
- ii. What is the impact of implementing Suricata rules on the web server where IDS is implemented?
- iii. What are the results of attack testing using the Web Penetration Testing method with Skipfish and Dirbuster tools?
- iv. What are the results of testing attacks using the DDoS method with Slowloris?

According to the objective of this work, there are currently several unified threat management solutions for cybersecurity, which can be commercial as well as Open Source, with Open-Source solutions being the focus of the work. It is considered that the increase in cyber threats to organizations and the evolution of attack techniques make it difficult for cyber-attack defense mechanisms to monitor network communications. Are OPNsense solutions suitable for detecting cyberattacks efficiently without influencing network performance?

## 3. RELATED WORKS

In the field of network security, several studies have explored the implementation of firewalls and intrusion detection systems (IDS) to enhance cybersecurity measures. Previous research has highlighted the importance of robust security frameworks that combine real-time monitoring, deep packet inspection, and behavioral analysis to detect and mitigate cyber threats effectively.

One such study by Oloyede et al. (2021) describes firewalls as essential security components that act as intermediaries between networks, allowing controlled access while blocking unauthorized traffic. Another study by Korcak et al. (2014) emphasizes the role of Intrusion Detection and Prevention Systems (IDPS) in Wi-Fi networks, providing real-time threat detection through anomaly-based and signature-based techniques.

Additionally, He (2021) investigates the application of firewall technology in securing computer networks. The study suggests that modern firewalls integrated with IDS/IPS provide enhanced protection against cyber-attacks. Stephani et al. (2020) discuss the deployment of Suricata IDS in web server security, showcasing its efficiency in logging and detecting unauthorized access attempts.

This study builds upon these prior works by integrating OPNSense firewall tools with Suricata IDS for enhanced cybersecurity. By leveraging advanced detection mechanisms and real-time monitoring, this research aims to provide a comprehensive solution for mitigating network security threats.

#### 4. METHODOLOGY:

##### 1. Implementation and Testing

**a. Observation:** Collecting data by direct observation is a way of collecting data using the eyes without the help of other standard tools for this purpose. In this method, the OPNSense Operating System records the results of attacks on servers that have been provided with security by Suricata.

##### 2. Research Methods.

According to Baskerville et al. (2016) [11], action research as a research method is founded on the assumption that theory and practice can be closely integrated with learning from the results of planned interventions after a detailed diagnosis of the problem context. The stages of the Action Research Method can be seen in Figure 1.

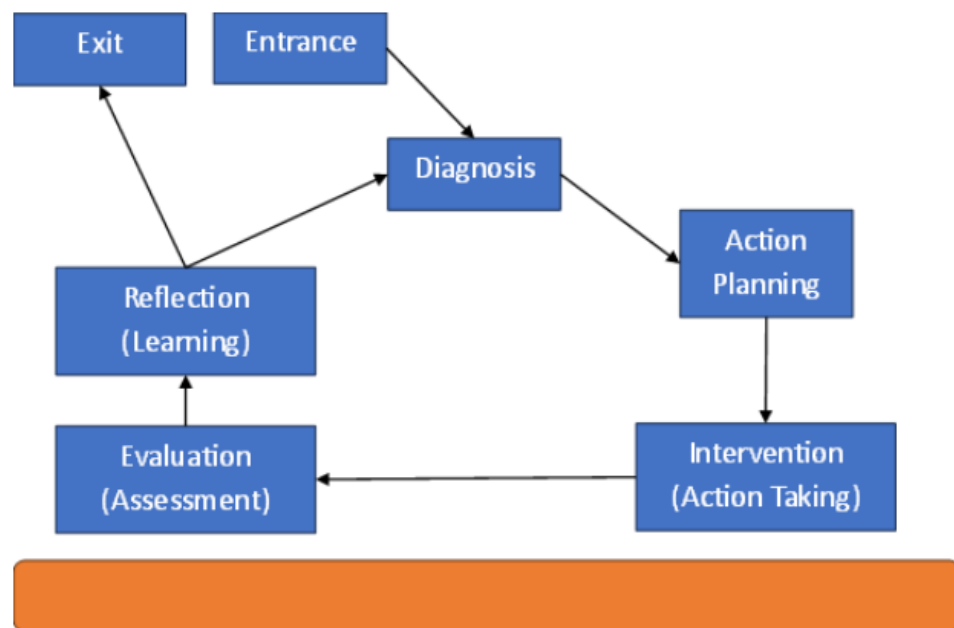


Figure 1: Action Research Stages

The steps for setting up a firewall and intrusion detection system using OPNSense tools. The work steps include setting up the OPNSense adapter, installing OPNSense, configuring the network on Kali Linux, OPNSense, and Ubuntu, configuring the web server on Ubuntu, configuring WebFilter and Suricata as intrusion detection software, and testing port scanning and denial of service (DoS) (Figure 2). Denial of Service (DoS) Testing Denial of Service testing is carried out using the LOIC tool which is installed in Kali Linux for Web services on Ubuntu 21.04. The steps are as follows:

- To carry out Denial of Service (DoS) testing using the LOIC tool, in the IP column, enter the IP Address of the website that will be attacked. Then click “Lock on”
- Then enter the port and type of attack, whether TCP, UDP, or HTTP, then set how often the tool will hit the server. Then Start the Attack.

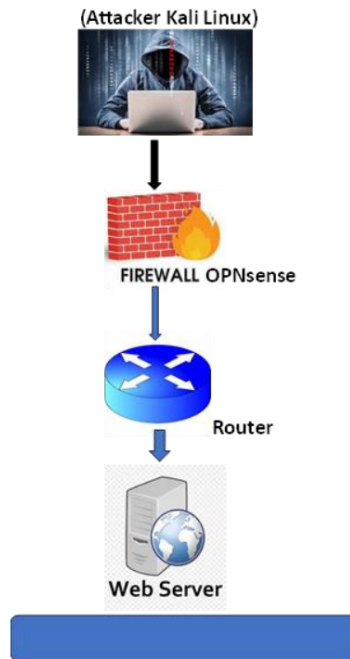


Figure 2. Suricata IDS System Design Scheme

1. Diagnosis: At this stage, the author identifies problems regarding attacks on the web server.
2. Action Planning: At this stage, the author understands what tools are needed, namely:
  - a. The OPNsense software, which already has the Suricata package functions as an Intrusion Detection System, will detect attacks that will come to the web server that has been provided. Before installing OPNsense, create a Virtual NAT Network as the adapter that OPNsense will use to connect to the internet with the network, then check “Supports DHCP”. After that, create a Virtual adapter Host Only network adapter for OPNsense to connect to the Server. With IP adapter.
  - b. The suricata rules file is installed in the Intrusion Detection System software; the suricata rules file functions according to the rules that will be installed to detect attacks on the web server.
  - c. Attacker Laptop.
3. Intervention (Action Taking): At this stage, researchers also begin to carry out attacks using the tools that have been prepared and configure the Suricata IDS that has been prepared. After completion, the examiner will write down how the Suricata System works to detect attacks.
4. Evaluation (Assessment): In this stage, the author collects the results of the attack in the Suricata log and evaluates whether Suricata can detect the attack.
5. Reflection (Learning): At this stage, the author gets the results from testing the attack and whether Suricata can detect the attack. Rules are also added here to increase the security of the suricata.

## 5. RESULTS AND DISCUSSION

At the analysis stage, the emphasis is on the search process being intensified and focused on software. The analysis found that there were weaknesses in the system for detecting attacks on the network. Web servers require a security system that can protect against all kinds of attacks and intrusion or scanning attempts by third parties. The design stage applies the design of several things needed in the previous stage as a configuration of the application/system [12].

Figure 2 is an IDS design scheme where the attacker is on the outside of the Web Server. The attack can enter via the internet and then enter the router, which will then be checked by the Suricata IDS system. This attack will check the IDS system in two ways, The first is signature-based, namely by matching network

traffic with a database containing attacks and infiltration methods that are often carried out by attackers. Anomaly-based detection is the second method, namely by comparing attack patterns that frequently occur with the attack patterns that are being monitored. The IDS used is Suricata on the Kali Linux operating system, which aims to protect the real server, the client server, and the network below. Suricata requires a package or library to build Suricata. Apart from that, a package is needed for Suricata rules because the IDS will work according to the rules created. The rules here are very important for Suricata in the form of scripts that can recognize intrusion actions that are occurring on the network where the IPS system is installed. IPS uses a firewall to block packets that comply with the rules created

In building a network or server, the first step is to determine what form of topology will be used. The following is the form of the topology that will be built:

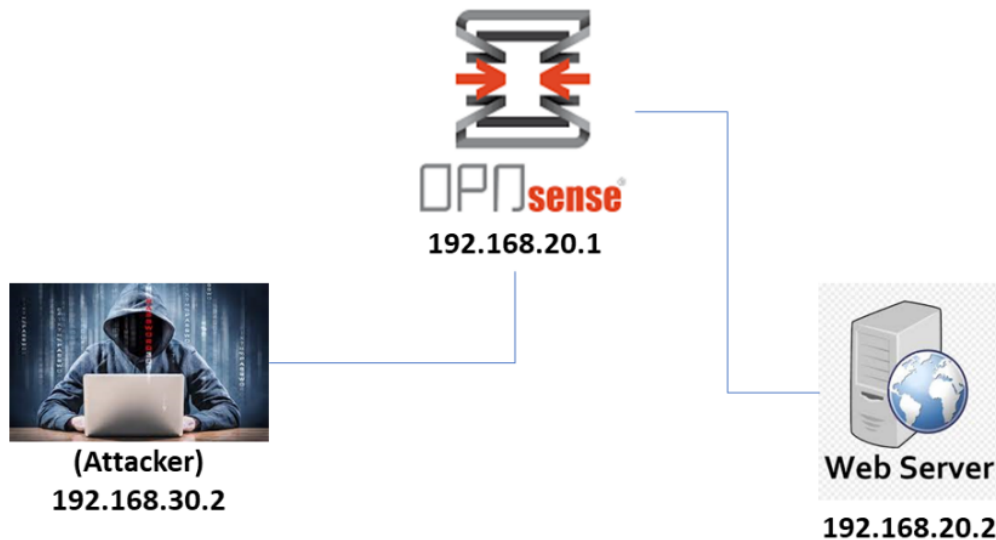


Figure 3: IDS Network Security Topology Using Suricata on a Web Server

From the topology above, it can be explained that 3 important components are needed here, namely OPNsense as a router and firewall, which will detect attacks originating from Kali Linux. Kali Linux functions as an attacker, where the IP address of Kali Linux is 192.168.30.2 and the IP address of OPNsense is 192.168.20.1. Also, the object that will be attacked is the Web Server, where the IP Address of the Web Server is 192.168.20.2. 2 testing methods will be used and have 3 tools, namely, the DDoS method with the Slowloris tool and the Web Penetration Testing method with the Dirbuster and Skipfish tools.

The flowchart (Figure 4) above explains how the Suricata IDS system works as a whole. Data packets that go to the server are checked first by Suricata. The data packet is then matched with Suricata rules. If the data packet is indicated as an attack, Suricata will create an alert and log file [13]. OPNsense provides rules that can regulate how Suricata can work. Users can set Suricata rules to block or allow attacks that enter the Web Server. For the LAN interface, there are 2 rules used, namely allowing internal DNS, while the other one blocks external DNS. What this means is that Suricata allows access via internal DNS and blocks all types of access originating from outside DNS. The WAN interface only has one rule, which blocks all types of attacks originating from the WAN interface. After setting up the existing rules, continue testing using Kali Linux with the Dirbuster, Skipfish, and Slowloris tools. To find out that the system being built can run as designed. The following tests are required:

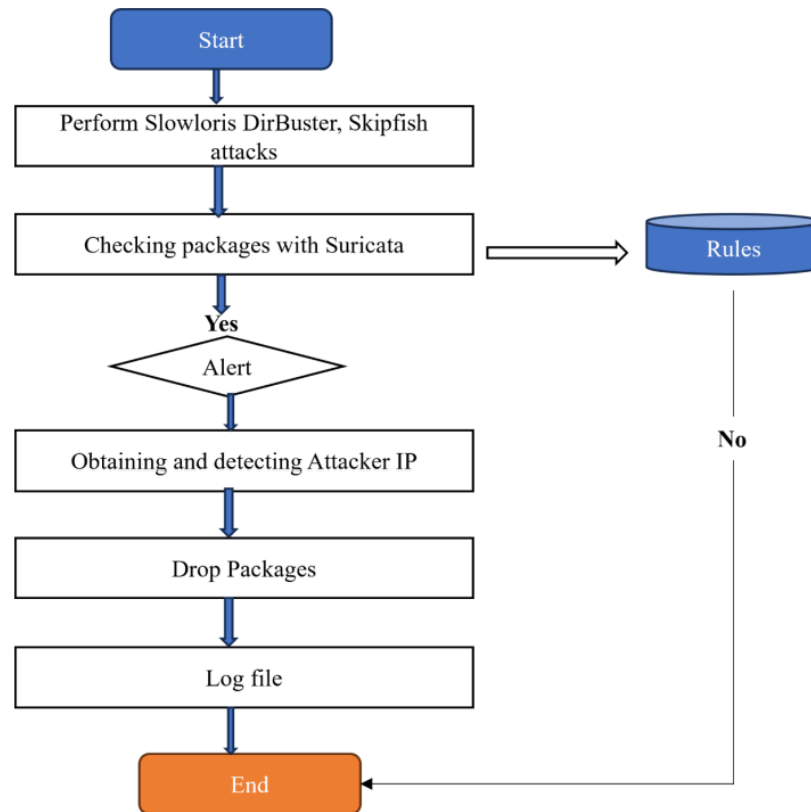


Figure 4. Suricata IDS System Flowchart

#### a. Testing using the Slowloris tool

1. Carry out testing using Kali Linux. Using the DDoS method with the *Slowloris* tool
2. First download the *slowloris* tool at <https://www.kitploit.com/2016/11/slowloris-low-bandwidth-dos-tool.html?m=0>.
3. Open the terminal, then type “*perl slowloris.pl*”
4. And, then type the command, i.e., the address of the server or web server that will be attacked
5. After that, through the OPNsense firewall, get a log of attacks that attacked the web server.
6. In the Suricata in OPNsense, you will know whether an attack has occurred or not through the available alerts and log files

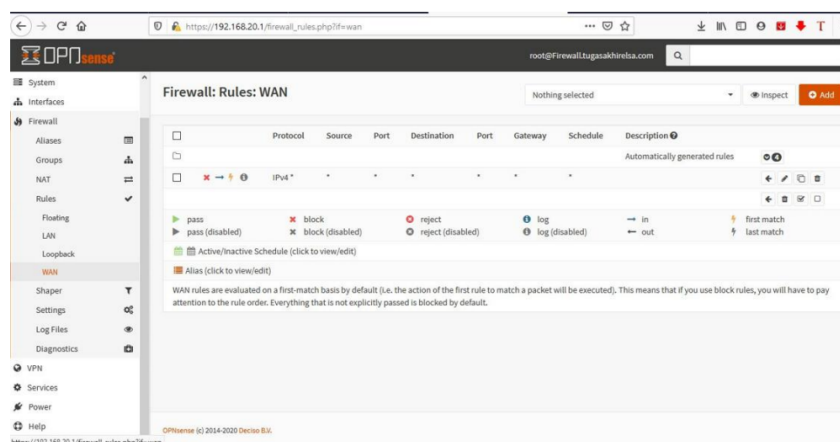


Figure 5. Setting rules on the LAN interface

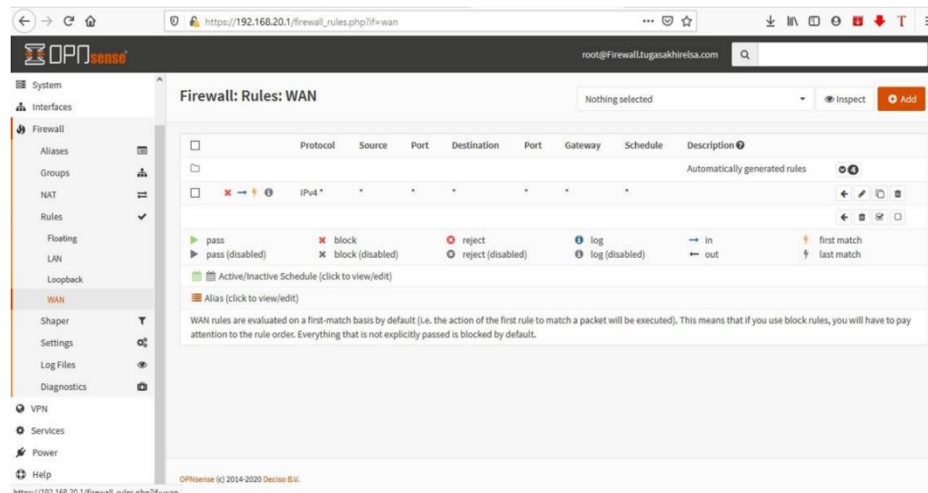


Figure 6: Set rules on the WAN interface

## b. Testing using the Dirbuster tool

1. Open the dirbuster tool on the Kali Linux operating system.
2. Fill in the IP of the web server that will be attacked.
3. Browse and select the wordlist file (usually located in /usr/share/dirbuster/wordlists) that you want to use to brute force.
4. After that, through the OPNsense firewall, get a log of attacks that attacked the web server.
5. In the suricata in OPNsense, you will know whether an attack has occurred or not through the available alerts and log files.

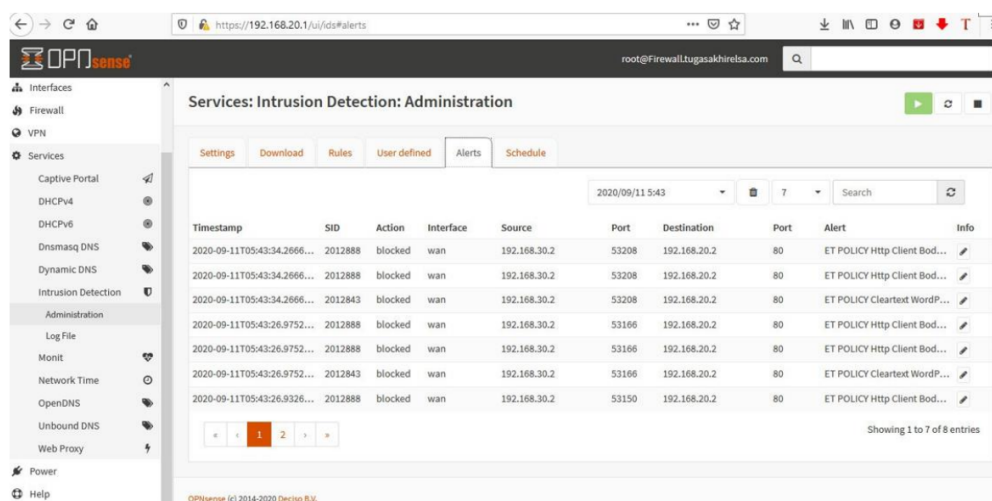
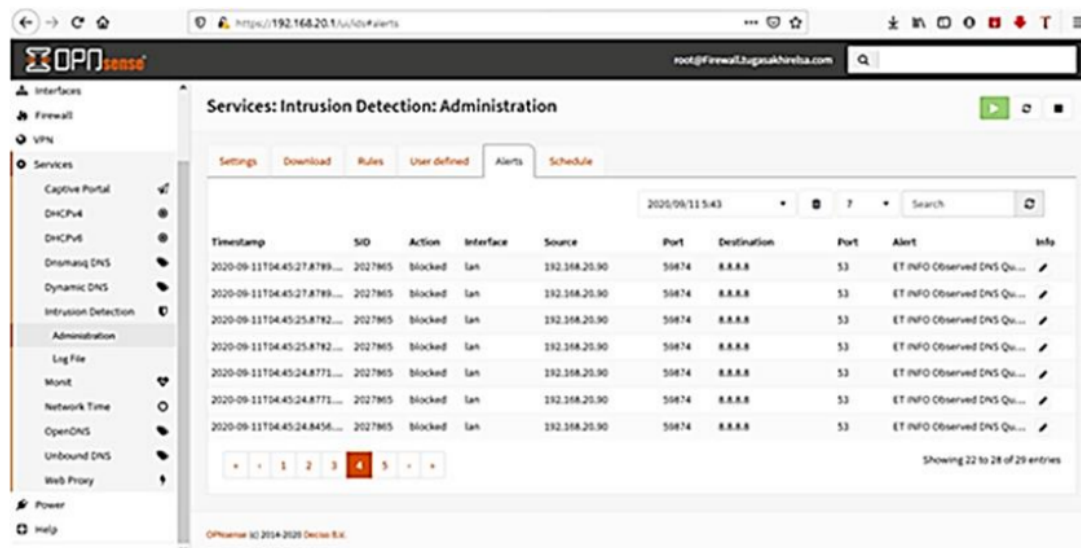


Figure 7: Alert results from WAN interfaces blocked by Suricata

## c. Testing using the Skipfish tool

1. Open the terminal then type skipfish -h.
2. After that, create a new folder to accommodate the information that skipfish is trying to hack from the web server.
3. After scanning, open index page which is already in the folder created previously. There is various information from the destination web server.



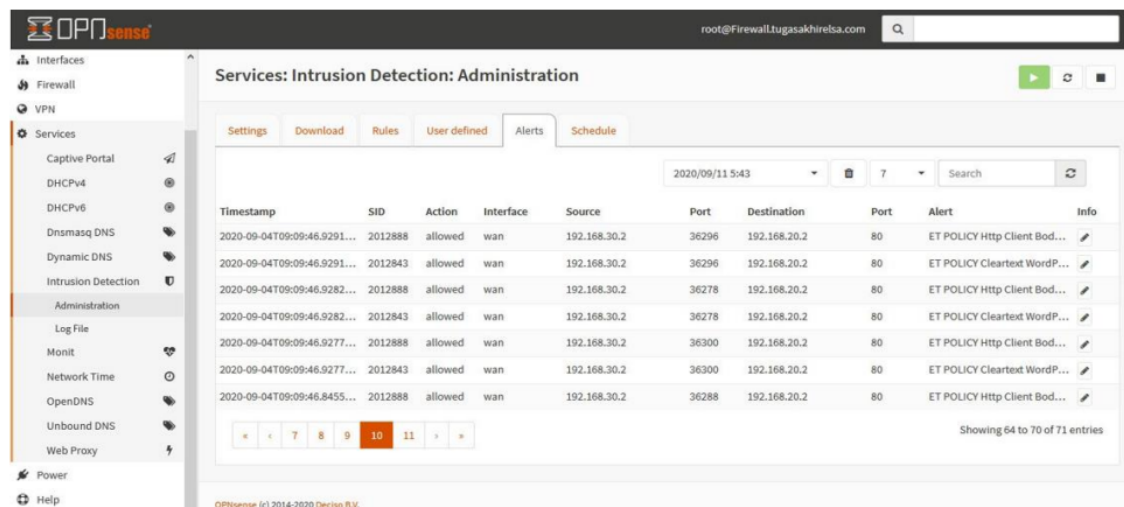


The screenshot shows the Opnsense web interface with the 'Alerts' tab selected under 'Services: Intrusion Detection: Administration'. The table displays 22 entries of blocked traffic from LAN interfaces. The columns are: Timestamp, SID, Action, Interface, Source, Port, Destination, Port, Alert, and Info. The 'Action' column shows 'blocked' for all entries, and the 'Interface' column shows 'lan'.

Timestamp	SID	Action	Interface	Source	Port	Destination	Port	Alert	Info
2020-09-11T04:45:27.879...	2027965	blocked	lan	192.168.20.90	59874	8.8.8.8	53	ET INFO Observed DNS Qu...	/
2020-09-11T04:45:27.879...	2027965	blocked	lan	192.168.20.90	59874	8.8.8.8	53	ET INFO Observed DNS Qu...	/
2020-09-11T04:45:25.8792...	2027965	blocked	lan	192.168.20.90	59874	8.8.8.8	53	ET INFO Observed DNS Qu...	/
2020-09-11T04:45:25.8792...	2027965	blocked	lan	192.168.20.90	59874	8.8.8.8	53	ET INFO Observed DNS Qu...	/
2020-09-11T04:45:24.8771...	2027965	blocked	lan	192.168.20.90	59874	8.8.8.8	53	ET INFO Observed DNS Qu...	/
2020-09-11T04:45:24.8771...	2027965	blocked	lan	192.168.20.90	59874	8.8.8.8	53	ET INFO Observed DNS Qu...	/
2020-09-11T04:45:24.8456...	2027965	blocked	lan	192.168.20.90	59874	8.8.8.8	53	ET INFO Observed DNS Qu...	/

**Figure 8: Alert results from LAN interfaces blocked by Suricata**

It can be seen that Suricata can detect and block attacks that enter the Web Server according to the rules that have been created. Here, the alerts are recorded according to the rules that have been created. The image below is proof that Suricata has blocked attacks originating from the WAN interface or via the WAN interface. Attacks originating from outside the LAN DNS interface will immediately block Suricata.



The screenshot shows the Opnsense web interface with the 'Alerts' tab selected under 'Services: Intrusion Detection: Administration'. The table displays 64 entries of allowed traffic from WAN interfaces. The columns are: Timestamp, SID, Action, Interface, Source, Port, Destination, Port, Alert, and Info. The 'Action' column shows 'allowed' for all entries, and the 'Interface' column shows 'wan'.

Timestamp	SID	Action	Interface	Source	Port	Destination	Port	Alert	Info
2020-09-04T09:09:46.9291...	2012888	allowed	wan	192.168.30.2	36296	192.168.20.2	80	ET POLICY Http Client Bod...	/
2020-09-04T09:09:46.9291...	2012843	allowed	wan	192.168.30.2	36296	192.168.20.2	80	ET POLICY Cleartext WordP...	/
2020-09-04T09:09:46.9282...	2012888	allowed	wan	192.168.30.2	36278	192.168.20.2	80	ET POLICY Http Client Bod...	/
2020-09-04T09:09:46.9282...	2012843	allowed	wan	192.168.30.2	36278	192.168.20.2	80	ET POLICY Cleartext WordP...	/
2020-09-04T09:09:46.9277...	2012888	allowed	wan	192.168.30.2	36300	192.168.20.2	80	ET POLICY Http Client Bod...	/
2020-09-04T09:09:46.9277...	2012843	allowed	wan	192.168.30.2	36300	192.168.20.2	80	ET POLICY Cleartext WordP...	/
2020-09-04T09:09:46.8455...	2012888	allowed	wan	192.168.30.2	36288	192.168.20.2	80	ET POLICY Http Client Bod...	/

**Figure 9: Alert results from the WAN interface permitted by Suricata**

Suricata can detect and allow attacks to enter the Web Server. If the rule is set to "Pass" or "Allow"



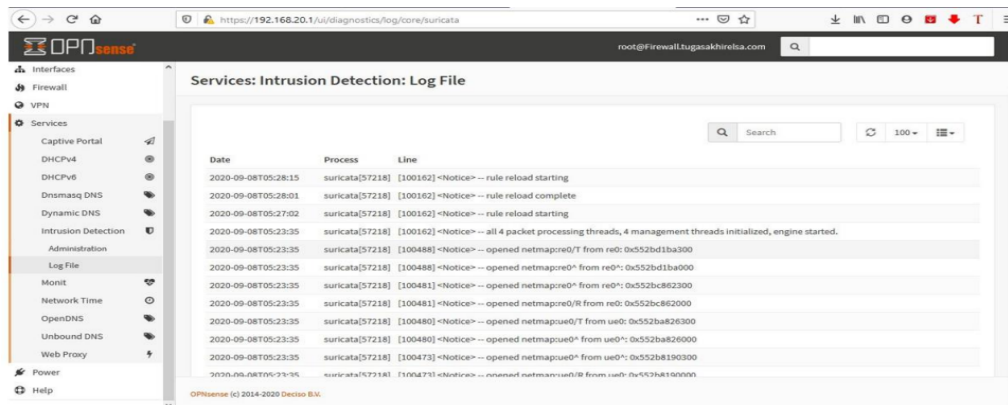


Figure 10: Results from Suricata IDS Log File

Suricata will also receive several Log Files, where in this Log File the user knows what series of events have occurred and been carried out by Suricata.



Figure 11: Continuation of Suricata IDS Log File

Continuation of the Log File results received by Suricata. In this log file, the log or activity of an attack that attacks the Web Server is recorded. In this log file, it is known what Suricata rules were running to detect attacks and show the security breach attempts that were made. Suricata can carry out detection according to the rules that have been created, and the final result of IDS testing using Suricata is to get Log Files and Alerts from attacks that enter the Web Server.

## 6. CONCLUSION

As technology has advanced, internal networks and the Internet have become popular. This has led to difficulties with unauthorised network intrusions, necessitating the addition of systems by organisers to manage information technology security flaws. Intrusion detection system: Software or a tool called an IDS (Intrusion Detection System) alerts the user to problems when unusual activity occurs within the system and helps to keep it safe. IDS's primary goal is to stop and identify activities like port scanning and probing that compromise system security. IDS software also distinguishes between internal and external attacks. In some cases, intrusion detection systems can detect and respond to attacks. Blocking malicious traffic by preventing the source IP address from accessing the network. Suricata-based IDS can monitor web server traffic and store detection and prevention results if an intruder enters the web server, as well as determine whether there is suspicious activity in the Suricata log. By integrating Suricata with the OPNsense firewall, intruder anomalies on the web server

can be identified and stopped. Utilising dir buster and skip fish tools, along with slow loris from the DDos method applied to test attacks on Intrusion Detection System (IDS) Suricata, the Information Technology Department of our university can help provide information regarding the detection of web scanning attacks through the implementation of an Intrusion Detection System (IDS) using Suricata on a web server. Suricata also lacks shared object rules, in contrast to other intrusion detection software. One could say that intrusion detection systems, or IDS, are a common tool in use today. Having an IDS is both necessary and practical for businesses in the current era of rapidly developing digital technology to safeguard their operations against potential threats within the system.

## REFERENCES

- [1] L. Chen, "Application of Computer Network Communication Technology in Production and Life," *J. Phys. Conf. Ser.*, vol. 1744, p. 032161, 2021.
- [2] J. Rogers and I. Tchakov, "Computer Networks as the Embodiment of Social Networks," *Int. J. Actor-Network Theory Technol. Innov.*, vol. 6, pp. 1–25, 2014.
- [3] G. Symon, "Information and communication technologies and the network organization: A critical analysis," *J. Occup. Organ. Psychol.*, vol. 73, pp. 389–414, 2000.
- [4] A. Anggraeni, J. Ginting, and S. Ikhwan, "Implementation of intrusion prevention system (IPS) to analysis triad CIA on network security attacks on web server," *J. Infotel*, vol. 14, 2022.
- [5] S. Tirumala, H. Sathu, and A. Sarrafzadeh, "Free and open source intrusion detection systems: A study," in *Proc. ICMLC*, 2015.
- [6] X. He, "Research on Computer Network Security Based on Firewall Technology," *J. Phys. Conf. Ser.*, vol. 1744, p. 042037, 2021.
- [7] A. O. Oloyede, N. Yekini, A. Akinwale, and O. Ojo, "Firewall Approach To Computer Network Security: Functional Viewpoint," *Int. J. Adv. Netw. Appl.*, vol. 13, pp. 4993–5000, 2021.
- [8] K. Ingham and S. Forrest, "Network Firewalls," *IEEE Commun. Mag.*, vol. 32, 1994.
- [9] M. Korcak, J. Lamer, and F. Jakab, "Intrusion Prevention/Intrusion Detection System (IPS/IDS) for Wi-Fi Networks," *Int. J. Comput. Netw. Commun.*, vol. 6, pp. 77–89, 2014.
- [10] E. Stephani, F. Nova, and E. Asri, "Implementation and Analysis of IDS (Intrusion Detection System) Network Security Using Suricata on a Web Server," *Sci. J. Inf. Syst. Technol.*, vol. 1, no. 2, pp. 67–74, 2020.
- [11] R. Baskerville and T. Wood-Harper, "A critical perspective on action research as a method for information systems research," in *Advances in Information Systems Research, Education and Practice*, Cham: Springer, 2016, pp. 169–190. doi: 10.1007/978-3-319-29269-4\_7.
- [12] Z. Zhou, Z. Chen, T. Zhou, and X. Guan, "The study on network intrusion detection system of Snort," in *Proceedings of the International Conference on Network and Distributed Systems (ICNDS)*, 2010, pp. 142–146. doi: 10.1109/ICNDS.2010.5479341.
- [13] S. Hassan and M. Daneshwar, "Anomaly-based network intrusion detection system using deep intelligent technique," *Polytechnic Journal*, vol. 12, pp. 100–113, 2023. doi: 10.25156/ptj.v12n2y2022.pp100-113.
- [14] L. A. Shihab, "Study and evaluation of wireless sensor networks performance," *Webology*, vol. 19, no. 1, Jan. 2022.
- [15] L. A. W. Shihab, "Wireless LAN security and management," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 2, no. 1, pp. 45–50, Oct. 2012.

## APPENDIX: SYSTEM TEST

To evaluate the effectiveness of the proposed firewall and IDS-based network security system, a series of practical tests were conducted. These tests included firewall rule validation, intrusion detection, and network attack simulations using penetration testing tools.

1. **Denial of Service (DoS) Testing**
  - Conducted using the LOIC tool in Kali Linux.
  - The attack targeted the web server to test its resilience against excessive traffic.
  - OPNSense firewall logged the attack, and Suricata IDS detected and blocked unauthorized requests.
2. **DirBuster Testing**
  - The DirBuster tool was used to perform brute-force directory enumeration.
  - The web server's security was tested against unauthorized access attempts.
  - Suricata IDS generated alerts for detected brute-force attempts, confirming successful detection.
3. **Skipfish Testing**
  - The Skipfish tool was utilized for web vulnerability scanning.
  - The tool attempted to exploit security loopholes within the web server.

- OPNSense firewall and Suricata IDS effectively blocked malicious requests and logged attempted intrusions.
- 4. **Slowloris Attack Simulation**
  - A Slowloris attack was conducted to exhaust server connections.
  - Suricata IDS successfully identified and mitigated the attack by blocking connections exceeding the allowed threshold.

These tests confirm the system's ability to detect, log, and prevent unauthorized access and cyber-attacks, thereby reinforcing its effectiveness in real-world network security applications.