Effectiveness of Wireshark Tool for Detecting Attacks and Vulnerabilities in Network Traffic

Nawal A. L. Mabsali(✉), Hothefa Jassim, and Joseph Mani

Modern College of Business and Science, Muscat, Sultanate of Oman
{20212797,hothefa.shaker,drjosephmani}@mcbs.edu.om

Abstract. Due to the popularity of using the technology, network security plays a crucial role recently which supports to establish strong systems that work against cyberattacks. Furthermore, the term "network vulnerabilities" refers to the flaws in the network which attackers exploit to break security and steal critical data. To discover the weaknesses of the network, the attackers use the mechanism of open port scanning to reach the systems and data, therefore the administrator should configure the network correctly and close any open ports. Monitoring network traffic is very important, so the developer focus to design analyzing tools that employ to inspect transmitted packets over the network to trace anomalous activities. Wireshark is one of the most well-known packets analyzing tool that is used to monitor the packets as well as used for examining the protocols. Moreover, the type of attack can be determined from the statistic report which generated by Wireshark tool. For instance, if the attacker sends *syn* packets to a target device, Wireshark will show the detail of *syn* packets. Practically, when *tcp syn* requests are flooded to any device, there will be a huge impact on the device's resources like consuming the bandwidth which affects system performance at the end. This paper presents a penetration test to lunch *syn* flood attack by sending a huge number of *syn* packets from Kali Linux machine to three targeted machines which are Windows 8.1, Windows 10 and Metasploitable. The test includes three scenarios, the first one focus on flooding *syn* packets by using the real source *ip* address of the attacker machine while the second scenario relays on sending *syn* packets by utilizing a spoof source *ip* address. The final scenario depends on using a random source *ip* address to flood *syn* packets. The Wireshark tool will be run in Kali machine to capture the packets and generate detailed reports. The results of captured data will be recorded to make analysis and list the capabilities of this tool.

Keywords: Wireshark tool · Syn flood attack · Vmware · Hping3

1 Introduction

The cybersecurity field becomes very important, therefore there're a lot of tools existing currently that are used to enhance the security of the systems and support detecting attacks. The increasing use of technology led to expanding cybercriminals in several areas to exploit sensitive information, which is relevant to the systems, users and because of that the security topic considers a significant field. Moreover, the developers start

N. Bacanin and H. Shaker (Eds.): ICIITB 2022, ACSR 104, pp. 114–135, 2023.
https://doi.org/10.2991/978-94-6463-110-4_10

building and designing appropriate tools that utilize to monitor and control capturing data, to use for management, as well as for mitigating the issues [9]. In general, the idea is not relying on upon send a message through the internet from a source and receiving the same at the destination side, but it's depending on making a secure channel [17], so existing of security tools becomes a significant task to be highlighted especially in recent days.

Traffic analysis is a practice of collecting information about the user activity on the network. A packet sniffing tool is required in this case to collect traffic patterns and identify the users who are attempting to access the internet and using the bandwidth. The used tool is called Wireshark, and it is quite useful because it has features that make analyzing networks quick and easy. A network administrator can use the data gathered from a live network to manage it more accurately and to examine collected statistics for analysis [21].

Observing and analyzing network traffic is a basic task to be considered, for it required to make a further tracing of anomalous actions which can be detected by security tools. This is needed to protect the network and look for any indications of possible attacks or suspicious behavior. In this case, Wireshark can be used for packet sniffing to examine captured data [15].

Network security is a matter of prime importance, and the techniques include penetration testing which allows organizations to implement regular quality control and detect threats as well as the system's vulnerabilities. Normally, vulnerability assessment is conducted as per multiple processes such as making the setup of a case study and specifying the used tool. The next process is implementing the test to identify the weaknesses and following that analysis stage to create a plan to handle issues and use reasonable solutions to decrease the impact of any future attacks. After that, reporting the case to record the outcomes and deliver the analysis's data to the appropriate authorities. The last step is the recovery stage which enforces security practices for mitigating the vulnerabilities [17].

Many attacks can be detected by Wireshark, syn flood attack is one example of the well-known threats which can be captured by this tool. It's the most frequent and harmful form of distributed denial of service attack that has resulted in server resource unavailability. This kind of attack depends on using the TCP three-way handshaking of TCP session setup process to waste resources on the targeted machine and make it inaccessible by sending many packets per second [5].

Three-way handshaking establishes connection reliability in TCP communication approach between client and server [10] that used three types of packets. Once the sender needs to create a connection with the receiver, syn request will be transmitted to the receiver side. Syn and Acknowledgment (Syn-Ack) packet sent back to the sender from the receiver and the connection information will be saved in a block known as transmission control block (TCB) and that state will be named by half-open connection (HOC). Finally, the sender will send (Ack) packet to the receiver for creating the connection [20][23][5].

Wireshark is a packet analyzer tool that supports to provides more details to the administrator such as time of transmission, source and destination ip addresses, and protocol information. This data may be useful for evaluating the incidents or addressing network device issues.

Whenever there is a conversation over the network between two or more devices, Wireshark can capture the packets and display statistics detail in its interface as it uses a graphical user interface (GUI). Based on generated reports, the investigators can catch malicious activities and decide the proper policies and security countermeasures needed to protect the network.

Wireshark is used by a security analyst to determine if the network traffic represents a suspicious behavior, identify the nature of the attack, and show the attacker ip addresses, and the attack's origin. To come up of proper set up rules on a firewall to prevent ip addresses from which the malicious traffic came. Although using Wireshark is legitimate, it could be criminal if someone intend to track the network without being legally allowed to do [23].

The main objective of this project is to investigate the network traffic by using the Wireshark packet analyzer tool in order to assess the tool's capabilities for discovering vulnerabilities and detecting syn flood attacks. The research depends on launching syn flood attack by kali Linux machine to create a denial of service with the support of using hping3 command. The packets will be flood to three nodes which are Windows 8.1, Windows 10 and metasploitable. Basically, the experiment focuses on performing a penetration test to target an open port number 139/tcp netbios-ssn service. The study includes various scenarios that help to give clear thoughts to generate the results of analysis and identify the effectiveness of the Wireshark tool.

This paper is structured as follows: Sect. 2 provides an overview of Wireshark tool and shows the previous work conducted which is related to this domain (literature review). Besides, Sect. 3 demonstrates the methodology that includes a brief idea about syn flood attack, software & hardware used for the experiment, an overview of hping3, and displays detail of three scenarios. After the observation stage, the results and analysis will be described in Sect. 4. The last part of this paper specifies the conclusion and future work which will be listed in Sect. 5.

2 Literature Review

Basically, the attacker can use packet analyzer tools to track the conversation in network traffic between the client and server side to make cybercrimes and get unauthorized access to the systems. On the other hand, the administrator takes advantage of these tools for diagnosing the network packets to generate assessment reports that can be supported to implement legitimate responses for any malicious actions [9].

2.1 Related Work

This section presents some penetration testing conducted by previous research projects that use to examine the flow of packets in the network and identify the system weaknesses and vulnerabilities. Also, some of the research studies are evaluating the efficiency of the Wireshark tool.

(P. Goyal and A. Goyal, 2017) [9] were conducting a comprehensive study to compare tcpdump and the Wireshark tool. Essentially, Wireshark is a packet sniffing tool that helps to detect attacks, denial of service attack (DoS) is one example in this case. In addition, the tool can work like an intrusion detection system (IDS) to discover the security breaches of various protocols. Both tools are available in open source as packet sniffing to enhance the collection of raw data that is being transmitted, also packet information can be saved for future use. Tcpdump tool allows dumping the packets in raw pattern with less analysis, whereas Wireshark enhances graphical interface that enable filtering choice. For monitoring the speed of packet captured, it displayed that many of packets dropped when using tcpdump tool for approximate 2 to 3 percent. On the other hand, it showed that Wireshark is faster in term of packet capturing. Some features of Wireshark make this tool more efficient such as ability of distinguish between the different protocols and this feature is absent in tcpdump. Another feature is Wireshark has more capability to analyze the packets more than tcpdump tool.

Based on paper (S. Sandhya et al., 2017) [19] discussed about penetration testing to discover the vulnerabilities by using Wireshark that assist the tester to ensure the network security and find out the correct measurements to prevent the attacks activities. Because web applications mostly exploit by the attackers, so the organizations can monitor the transmitted data to identify security breaches and analyze the packets while performing user authentication process that shall give comprehensible detailed about the user identity. Motivation of the research focus on analyze vulnerable website to check the proficiency of Wireshark to capture credential data after user login process performed. The final result displayed in this tool, to show the username and password which entered by the user through (HTTP) protocol, so in this case the system can be controlled by reporting the issue and implement proper solution.

Another article (P. Navabud and C. Chen, 2018) [16] covered the results of practicing HTTP as well as HTTPs protocols with the support of using Wireshark tool. HTTPs provides high level of security to protect data between web browser and server by applying additional encryption mechanism such as SSL/TLS. While accessing the web mail “http://stmail.nptu.edu.tw“, it will request for credential information. Once the user provides the detail, the tool will record the data and filter the option of ‘HTTP’ to list the sensitive information. On the other hand, the Wireshark can’t capture data which send by using HTTPs protocol.

Searching for open ports is one of the main targets for any attacker to gain access to the system and steal critical data. However, the attacker can use null scan method through ‘nmap’ by sending TCP packets to victim without including flags. The result of this kind of scan will determine if the port is open or close by waiting the response from the destination. Once there is no response, so the attacker recognizes that the port is open but if there’s RST response it indicates that the port is closed. (G. Bagyalakshmi et al., 2018) [2] performed a study on network vulnerabilities to catch abnormal activities to implement security management and ensure forensic analysis. Examining the packets and finding out vulnerable ports detected by Wireshark that assist to generate the required information.

The existence of various attacks catches the attention of the developers to improve tools that are working against such activities. (R. Banu et al., 2019) [3] pinpointed in their research about Xmas attacks that can be detected by Wireshark and Snort tools. Also, the comparison of efficiency was highlighted in the research as well. Snort sends false alarms and the attack's nature is not specified in the detection process which led to a low detection rate. On the other side, using Wireshark will not give a warning in case of any threats located but it can generate detailed data about abnormal actions for making a decision based on statistics and graphs produced. Moreover, it is applicable to different platforms for execution purposes. Researchers introduced other methods that were used to cover the drawbacks of Wireshark and Snort which is called 'MONOSEK' as its software supports session and packet analysis as well as packet inspection.

(V. Dang et al., 2018) [5] described syn flood attacks in detail and presented the proposed solution "SDN-Based SYN Proxy" to mitigate such attacks. For the testing process, the researchers used Wireshark for the experiment to check the effectiveness of their solution by recording packet flow. The server was flooded with a massive number of Syn packets during a SYN flood exploit, that frequently utilised faked source IP addresses. Syn requests consumed CPU resources and network bandwidth [7]. The server generates several HOCs while being unknowing of the attack, and before these HOCs reach to time out, the machine's resources are rapidly wasted as a result of being consumed with useless TCBs [5].

As per the research paper of (H. Iqbal and S. Naaz, 2019) [11] specified the importance of using the Wireshark tool to evaluate the network performance and detect several types of attacks. Sometimes, outside attacks try to make the web server down by transmitting fake ARP reply packets or penetrating the network with malicious activity to become an aspect of a botnet that generates service interruptions. In this research, the authors showed the efficiency of Wireshark to detect such attacks as DoS attack, DNS attacks, ARP poisoning and the countermeasures specified to prevent the attacks. The final result showed that Wireshark is a powerful tool used to keep track of network activity.

(G. Jain and Anubha, 2021) [12] conducted a study to determine how Snort and Wireshark work together to detect and analyze malicious practices. However, Wireshark has the ability to show if the packets are encrypted, which will display as per captured traffic, even though it is less accurate for detecting intrusions. Wireshark enables to detect numerous cyberattacks, including DoS and DDOS. Snort captures real-time internet traffic and matches it to established rules; if no match is achieved, alert messages are sent to the user. This approach makes use of a router to connect to the internet and ensure consistent data packet delivery. It creates a log file that contains all of the live packets were recorded. Then, log file of Snort tool will be transferred to Wireshark to analyze the collected network packets.

(Charles, AS Joseph, and P. Kalavathi, 2018) [4] focused on their paper about making a quality evaluation of RPL routing protocol for IPv6 with support of the Wireshark tool and Cooja Simulator. This protocol is a proactive protocol used for the wireless network (low power and loosy networks) that's a popular protocol used in IoT technology. Packet analysis is one of the major fundamental fields in network security to make a comprehensive study of protocol patterns. The role of Wireshark is to enhance the study of analyzing the protocol.

(Musa, Ahmad, 2020) [14] proposed approach to capture peer-to-peer traffic over the network and apply inspection with the assistance of Wireshark tool as a part of forensic analysis. According to the analysis's findings, the suggested method was effective in tracking the threats' sources on the internet and providing valid digital evidence that may be used in forensic investigations.

(Acosta, Jaime C., and Daniel E. Krych, 2021) [1] performed three practical software reverse-engineering experiments to understand the pattern of Remote access trojans (RATs) malware in the network. First of all, Wireshark used to examine malware traffic seen between C2 server as well as the Bot. Another tool required to remove malware from a hard drive by inspecting the drive through the Volatility tool. Ghidra tool utilize for Reverse-engineering the C2 server and Bot's communication. Based on the experiment demonstrated in this article, it concludes that Wireshark can catch the traffic of RATs malware to give detailed information.

An article by (Dodiya, Bindu, and Umesh Kumar Singh, 2022) [6] collected Indicators of Compromise (IoC) in the network for malicious activities by gathering raw data over the wire and studying the pattern of the activities. Authors depend on the Wireshark analyzer tool to classify the signatures of the attacks, so it will provide accurate details to system administrator. This tool has capabilities to display some data of (IoC) such as Domain name, MAC address, File Hashes, Hostname and Host ip address.

(X. Guo and X. Gao, 2022) [13] proposed method to detect syn flood attack. "hping3" tool was used in the experiment to flood syn packets to a target device, and the flow of spoof packets monitor by Wireshark to display packets information and ip addresses was used. The administrator will take immediate response to block source ip address and disable open ports [24] as a task of the incident response matrix.

3 Methodology

3.1 Hardware Requirements and Parameters

Basically, this section specifies the hardware and software requirements that needed for implementing the experiment and evaluating the analyzed reports generated by Wireshark tool based on the attack detection process. VMware workstation utilizes for running Kali Linux, Windows 8.1, Windows 10 and metasploitable machines. Wireshark tool runs on Kali Linux machine to detect the packets of the attack and monitor the simulation results to extract useful information. Besides, draw.io application will be used for making diagram for detection process and draw scheme of testing process.

The following, Table 1 displays the required specifications of hardware and software which will be used for the experiment.

Table 1. Specifications of hardware and software

	Operating system	Memory	Hard disk	Processor (CPU)
VMware Workstation	VMware® Workstation 16 Pro 16.2.4 build-20089737			
To run Attacker Machine	Linux-Debian 10.x64-bit	8 GB	80 GB	4
To run Victim Machines	Windows 10 and later x64	1.4 GB	60 GB	1
	Windows 8.x x64	1.5 GB	60 GB	1
	Linux-Ubuntu	1 GB	8 GB	1
To run Wireshark Tool	Linux-Debian 10.x64-bit (Kali Linux)			

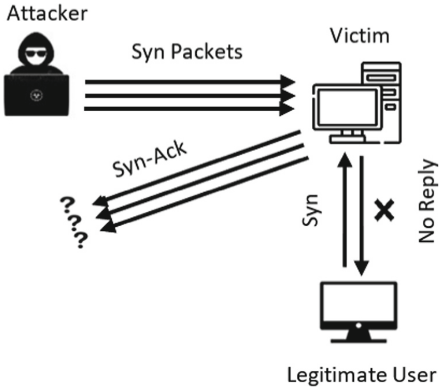


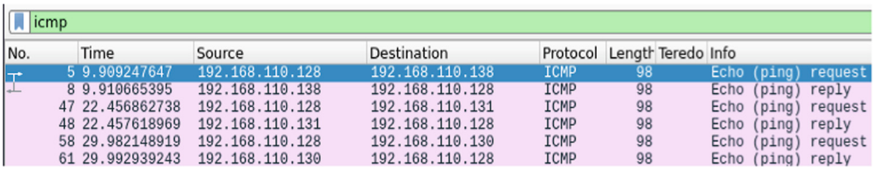
Fig. 1. Syn flood attack

3.2 TCP Syn Flood Attack

Recently, there're various of attacks available which target the systems and computers in the network in order to gain access and make use of sensitive data while other attacks impact the network resources to make it busy and unavailable to legematic users that lead to denial of service [8] such example is TCP syn flood attack. Attacker transmits huge number of syn packets for each second [5] to the victim without waiting for the response of (Syn-Ack) packets, therefore when usual user needs to establish TCP connection, syn packet will send to server but no replay will happen due to unavailability of the server (Fig. 1).

Table 2. Ip addresses for used machines

Machine	IP address
Kali Linux (attacker)	192.168.110.128
Windows 8.1 (Target 1)	192.168.110.138
Windows 10 (Target 2)	192.168.110.131
Metasploitable (Target 3)	192.168.110.130



icmp

No.	Time	Source	Destination	Protocol	Length	Teredo	Info
5	9.909247647	192.168.110.128	192.168.110.138	ICMP	98		Echo (ping) request
8	9.910665395	192.168.110.138	192.168.110.128	ICMP	98		Echo (ping) reply
47	22.456862738	192.168.110.128	192.168.110.131	ICMP	98		Echo (ping) request
48	22.457618969	192.168.110.131	192.168.110.128	ICMP	98		Echo (ping) reply
58	29.982148919	192.168.110.128	192.168.110.130	ICMP	98		Echo (ping) request
61	29.992939243	192.168.110.130	192.168.110.128	ICMP	98		Echo (ping) reply

Fig. 2. ICMP request and reply in wireshark

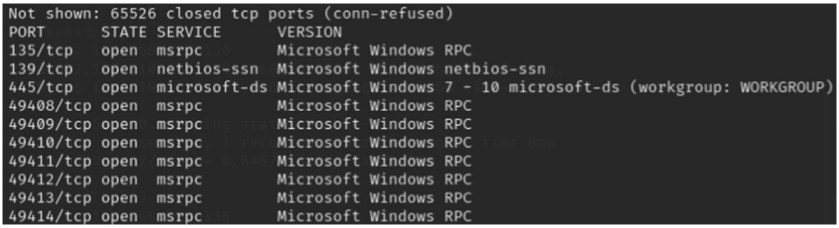
3.3 TCP Syn Flood Attack Implementation

To execute the experiment and examine the effectiveness of Wireshark tool for detecting the attack traffic in network, the case required to perform TCP syn flood attack in VMware workstation by using Kali Linux as attacker or source machine and monitor the flow of syn packets by Wireshark. This case relays on flood syn spoofing packets to three nodes in the network.

Generally, for enhancing the implementation process, three basic requirements needed as initial steps such as displaying IP addresses of source machine (Kali Linux) and destination machines that need to be used in the experiment to flood syn packets. Table 2 displays Kali machine IP address as well as IP addresses for Windows 8.1, Windows 10, and metasploitable.

The second initial requirement is examining the connectivity of the attacker machine (Kali Linux) with target machines in the network by using ping command once ping successful the syn flood packet can be transmitted. ICMP request messages send to specific devices and reply massages show to ensure the reachability of attacker machine to other machines. Figure 2 lists the success result of ping to view icmp request along with response and time that establish between attacker machine and victims.

Third requirement is scanning for open ports to find and exploit the services on the network of targeted resources. In order to find open ports for all victim machines, nmap command used to generate scan report and list all vulnerable port numbers. For about 65,535 ports are available but not all of these are used, some of them used for specific services that so-called well-known ports and the number of well-known ports is 1024. Here, the attacker takes advantage of open ports to use it as a door for compromising the system and consume the resources.



```
Not shown: 65526 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
49408/tcp open  msrpc        Microsoft Windows RPC
49409/tcp open  msrpc        Microsoft Windows RPC
49410/tcp open  msrpc        Microsoft Windows RPC
49411/tcp open  msrpc        Microsoft Windows RPC
49412/tcp open  msrpc        Microsoft Windows RPC
49413/tcp open  msrpc        Microsoft Windows RPC
49414/tcp open  msrpc        Microsoft Windows RPC
```

Fig. 3. Output of open port scanning

To run nmap, it required to scan all numbers of ports from 0 to 65535 to identify the vulnerable ports and specify the version of each service. The following part clarify the command used along with arguments.

Command:

nmap -sV -p 0-65535 <target ip address>

Arguments:

- -sV: Display service and version
- -p: Represents the port
- 0-65535: Specify range of ports
- <target ip address>: 192.168.110.138, 192.168.110.131, 192.168.110.130

The next Fig. 3 list the detail of open ports, service names as well as the versions which needed as basic requirement for implementation.

Generated reports for all victims show that port number 139/tcp (netbios-ssn) is open and it will be considered as target port in this experiment. The purpose of netbios-ssn is for session service that allow tcp netbios connections to be created through port 139 for windows machines as well as for other machines which running Samba (SMB). That's used for connection-oriented to allow file sharing services over network, so the TCP connections establish NetBIOS sessions [22].

In general, the testing process shows in Fig. 4 to illustrate executed steps:

3.3.1 Hping3 Tool for Lunching Attack

In order to lunch syn flood attack, hping3 tool utilized to send huge number of TCP syn packets that called denial of service attack (DoS) which consume a lot of resources and reduce the performance of devices, so the processors as well as network utilizations will impact. Hping3 command supports to specify the parameters based on examiner needs that enhance the analyzing process for various situations. To execute the command, it should access the root in kali Linux.

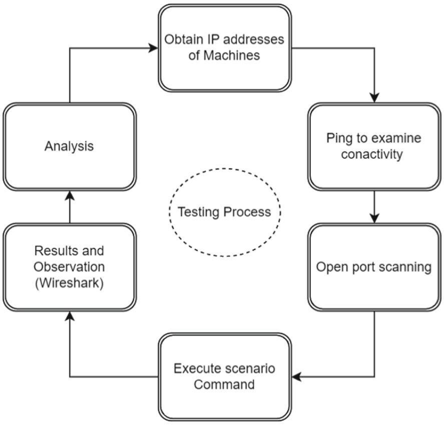

Obtain IP addresses of Machines
Ping to examine conactivity
Open port scanning
Execute scenario Command
Results and Observation (Wireshark)
Analysis
Testing Process

Fig. 4. General scheme of testing process for the scenarios

3.3.2 Experiment Scenarios

This section pinpoints the pen testing to include four machines, one of them as attacker machine and the rest as victims. The simulation specifies the flow of tcp syn packets with the respective of different three scenarios and the results analysis recorded as per Wireshark packet analyzer tool.

Scenario 1: To flood huge number of syn packets to three victims' machines with **real source ip address** (attacker ip) and use port number 139 for the service tcp/netbios-ssn. Here, the attacker provides details of targeted machine.

Simulation Scenario Setup: The command used along with arguments will be identified in the following.

Command:

```
hping3 -S --flood <target ip address> -p 139
```

Arguments:

- -S: Represents 'Syn' packets
- --flood: flag to ignore replies (Syn-Ack) and flood a lot of packets
- <target ip address>: 192.168.110.138, 192.168.110.131, 192.168.110.130
- -p 139: target port number 139 for service netbios-ssn

Running command of Scenario 1 show in following part and the original ip address of attacker machine will be used in this case and ip addresses of all victim machines will be specified.

```
  ┌──(root㉿kali)-[/home/kali]
  └─# hping3 -S --flood 192.168.110.138 -p 139
       HPING  192.168.110.138  (eth0  192.168.110.138):  S
       set, 40   headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.138 hping statistic ---
       557089 packets  transmitted,  0  packets  received,
       100% packet loss
       round-trip min/avg/max = 0.0/0.0/0.0 ms
  ┌──(root㉿kali)-[/home/kali]
  └─# hping3 -S --flood 192.168.110.131 -p 139
       HPING  192.168.110.131  (eth0  192.168.110.131):  S
       set, 40   headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.131 hping statistic ---
       1007175 packets  transmitted,  0  packets  received,
       100% packet loss
       round-trip min/avg/max = 0.0/0.0/0.0 ms

  ┌──(root㉿kali)-[/home/kali]
  └─# hping3 -S --flood 192.168.110.130 -p 139
       HPING  192.168.110.130  (eth0  192.168.110.130):  S
       set, 40   headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.130 hping statistic ---
       250875  packets  transmitted,  0  packets  received,
       100% packet loss
       round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Scenario 2: Flood a lot of syn packets to victim machines to consume resources by using **fake source ip address** to target port number 139 and observe the packets flow through Wireshark tool. In this case, the fake ip address for attacker machine will be (192.168.110.150). Usually, the attacker uses this mechanism to hide source identity and to challenge the detection systems, so the work of administrator will be more difficult to stop such malicious threats.

Simulation Scenario Setup: The following part displays the command used and show the arguments as well:

Command:

hping3 -S <target ip address> --spoof <fake source ip address> -p 139 --flood

Arguments:

- -S: Represents ‘Syn’ packets
- <target ip address>: 192.168.110.138, 192.168.110.131, 192.168.110.130
- --spoof: is flag for fake IP address
- <fake source ip address>: 192.168.110.150
- -p 139: target port number 139 for service netbios-ssn
- --flood: flag to ignore replies (Syn-Ack) and flood a lot of packets

The next running command for Scenario 2 state the needed arguments and parameters.

```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S 192.168.110.138 --spoof 192.168.110.150 -p
   139 --flood
   HPING 192.168.110.138 (eth0 192.168.110.138): S set, 40
   headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.138 hping statistic ---
485550 packets transmitted, 0 packets received, 100%
packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S 192.168.110.131 --spoof 192.168.110.150 -p
   139 --flood
   HPING 192.168.110.131 (eth0 192.168.110.131): S set, 40
   headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.131 hping statistic ---
825183 packets transmitted, 0 packets received, 100%
packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S 192.168.110.130 --spoof 192.168.110.150 -p
   139 --flood
   HPING 192.168.110.130 (eth0 192.168.110.130): S set, 40
   headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.130 hping statistic ---
758283 packets transmitted, 0 packets received, 100%
packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Scenario 3: Sending massive number of syn packets to target devices by using **random ip address** for the source machine to exploit port number 139 service. By this method the attacker will hide the identity, so the random source ip generated and it will show different ip addresses (several source locations). This will create obstacles for forensic investigators once, they collect the evidence and make investigation.

Simulation Scenario Setup: The following part, illustrate the command required for this Scenario as well as the arguments detail.

Command:

hping3 -S <target ip address> --rand-source -p 139 --flood

Arguments:

- -S: Represents ‘Syn’ packets
- <target ip address>: 192.168.110.138, 192.168.110.131, 192.168.110.130

- --rand-source: to generate random ip addresses for the source
- -p 139: target port number 139 for service netbios-ssn
- --flood: flag to ignore replies (Syn-Ack) and flood a lot of packets

Scenario 3 running command clarify in the following.

```
 ┌──(root㉿kali)-[/home/kali]
 └─# hping3 -S 192.168.110.138 --rand-source -p 139 --flood
    HPING 192.168.110.138 (eth0 192.168.110.138): S set, 40
    headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.138 hping statistic ---
    897623 packets transmitted, 0 packets received, 100%
    packet loss
   round-trip min/avg/max = 0.0/0.0/0.0 ms
 ┌──(root㉿kali)-[/home/kali]
 └─# hping3 -S 192.168.110.131 --rand-source -p 139 --flood
    HPING 192.168.110.131 (eth0 192.168.110.131): S set, 40
    headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.131 hping statistic ---
    1450581 packets transmitted, 0 packets received, 100%
    packet loss
   round-trip min/avg/max = 0.0/0.0/0.0 ms
 ┌──(root㉿kali)-[/home/kali]
 └─# hping3 -S 192.168.110.130 --rand-source -p 139 --flood
    HPING 192.168.110.130 (eth0 192.168.110.130): S set, 40
    headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.110.130 hping statistic ---
    1141731 packets transmitted, 0 packets received, 100%
    packet loss
   round-trip min/avg/max = 0.0/0.0/0.0 ms
```

3.4 TCP Syn Flood Attack Detection Using Wireshark Tool

Wireshark tool is packet and protocol analyzer that support to capture the live flow of data over the network also the researchers can employ this tool to get accurate graphs representation since it has graphical user interface (GUI) feature. Some data can be recorded by Wireshark like network characteristics, bandwidth utilization and detecting malicious actions (syn flood attack) that performed by intruders [18]. More than 400 protocols can be decoded [4] to enhance the flexibility [13], as well as more than 750 protocols can be handled. Also, new protocols included in this tool to be applicable with the various versions. Wireshark can be supported by 20 different platforms. Moreover, both promiscuous and non-promiscuous mode can be maintained by Wireshark [4].

For enhancing evaluation and detection process this tool support the following functions:

- **Collection stage:** As first step this tool will gather raw data in form of binary from the network once the attack performed. When the network interface is shifting to promiscuous mode, then this mode will enable network card to monitor all flowed traffic [18].
- **Conversion Stage:** After that, the collected binary data will be transferred into understandable format that will support for analyzing stage. General data about the packets as well as statistics and graphs are presenting in readable forms [18].
- **Analyzing Stage:** The last stage is evaluating and analyzing captured packets by examiners based on the final represented data which extracted from the tool [18].

Additional features to be considered for Wireshark tool which makes assessment and evaluation more flexible:

- **Supporting protocols:** Because of this feature, so several attacks can be detected which deal with various protocols such as tcp which used for flooding syn packets. As per the studies, more than 1000 protocols supported by Wireshark including DHCP, IP, bitTorrent and DNS3. Moreover, additional protocols take into account to be added whenever there're updates in this open-source tool [18].
- **User friendly:** The interface and layout structure of the tool assist the examiners to identify and analyze the packets like color coding of the protocols as well as graph illustration.

Using GUI instead of command line makes this tool more popular for conducting studies. Another crucial feature of using Wireshark is ability of capturing real time packets on the network and record it to make data analyzing offline that support analyzer to make a decision of incident response [18].

- **Supporting Operating Systems:** It compatible with various types of operating systems like Mac, Linux and windows [18].
- **Security Issues:** Determining malicious activities by analyzing the packets that send by attacker and define the proper security solutions for such issues. Besides to that, the examiner can set immediate action once the spoofing packets detected [18].

The following Fig. 5 displays detailed diagram for detection process, based on the demonstrated experiment and performed scenarios. Starting from executing the attack through Kali and ending to documentation stage. Wireshark will be run in Kali to capture and monitor the packets. While collecting the raw data the tool will convert it to readable format to present it in analyzing stage. System specialist can gather the report of statistic for any incident during investigation process. In any future cases, the recorded document will be beneficial to make incident response for the attacks.

After executing the command for all three scenarios (Real, Spoof, Random ip addresses for the source), the following data can be captured by Wireshark. Figures 6, 7 and 8 are displaying in the interface of the tool for implemented scenarios in sequence.

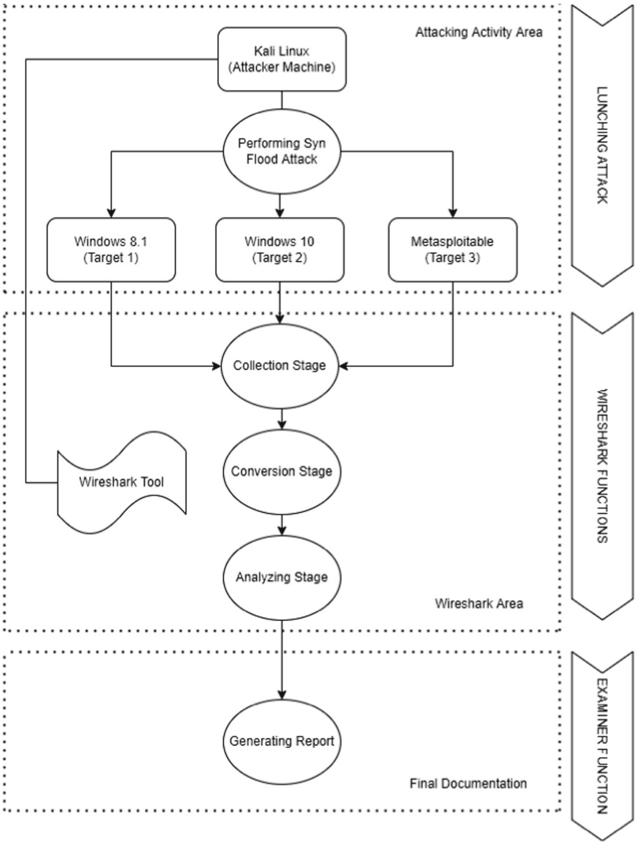
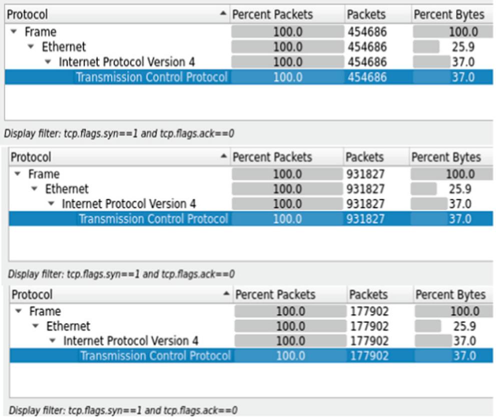


Fig. 5. Wireshark detection layered diagram



Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	454686	100.0
Ethernet	100.0	454686	25.9
Internet Protocol Version 4	100.0	454686	37.0
Transmission Control Protocol	100.0	454686	37.0

Display filter: tcp.flags.syn==1 and tcp.flags.ack==0

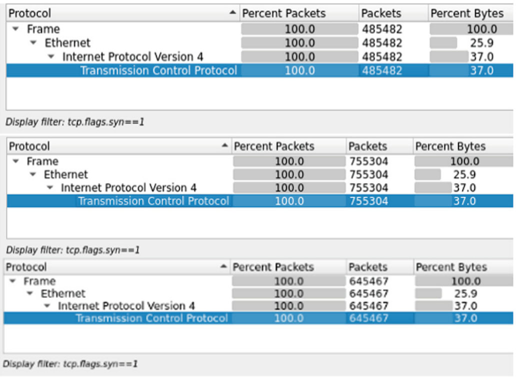
Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	931827	100.0
Ethernet	100.0	931827	25.9
Internet Protocol Version 4	100.0	931827	37.0
Transmission Control Protocol	100.0	931827	37.0

Display filter: tcp.flags.syn==1 and tcp.flags.ack==0

Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	177902	100.0
Ethernet	100.0	177902	25.9
Internet Protocol Version 4	100.0	177902	37.0
Transmission Control Protocol	100.0	177902	37.0

Display filter: tcp.flags.syn==1 and tcp.flags.ack==0

Fig. 6. Wireshark (protocol statistics) scenario 1



Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	485482	100.0
Ethernet	100.0	485482	25.9
Internet Protocol Version 4	100.0	485482	37.0
Transmission Control Protocol	100.0	485482	37.0

Display filter: tcp.flags.syn==1

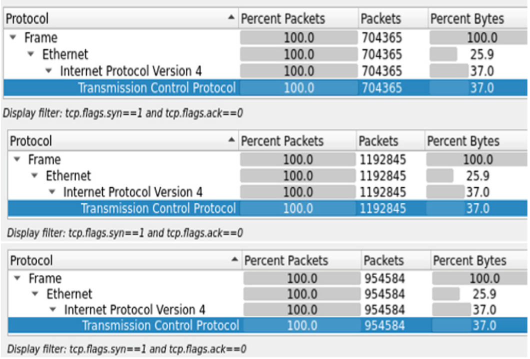
Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	755304	100.0
Ethernet	100.0	755304	25.9
Internet Protocol Version 4	100.0	755304	37.0
Transmission Control Protocol	100.0	755304	37.0

Display filter: tcp.flags.syn==1

Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	645467	100.0
Ethernet	100.0	645467	25.9
Internet Protocol Version 4	100.0	645467	37.0
Transmission Control Protocol	100.0	645467	37.0

Display filter: tcp.flags.syn==1

Fig. 7. Wireshark (protocol statistics) scenario 2



Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	704365	100.0
Ethernet	100.0	704365	25.9
Internet Protocol Version 4	100.0	704365	37.0
Transmission Control Protocol	100.0	704365	37.0

Display filter: tcp.flags.syn==1 and tcp.flags.ack==0

Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	1192845	100.0
Ethernet	100.0	1192845	25.9
Internet Protocol Version 4	100.0	1192845	37.0
Transmission Control Protocol	100.0	1192845	37.0

Display filter: tcp.flags.syn==1 and tcp.flags.ack==0

Protocol	Percent Packets	Packets	Percent Bytes
Frame	100.0	954584	100.0
Ethernet	100.0	954584	25.9
Internet Protocol Version 4	100.0	954584	37.0
Transmission Control Protocol	100.0	954584	37.0

Display filter: tcp.flags.syn==1 and tcp.flags.ack==0

Fig. 8. Wireshark (protocol statistics) scenario 3

4 Results Analysis

Section (3.3.2) examined three different Scenarios for executing the attack, while this section will be illustrated the results after analyzing the scenarios by Wireshark tool to assess the efficient of this tool and determine the accuracy of the tool to deal with vulnerable packets.

4.1 Measurements and Observations

Results for Scenario 1 is summarized as Table 3:

Results for Scenario 2 is summarized as Table 4:

Results for Scenario 3 is summarized as Table 5:

Tables 3, 4 and 5 show collected data that extracted as per the output displays in Wireshark for infected machines as well as some data gathered from Kali machine during lunching syn attack stage. Table 3 presents packets of scenario 1 for using real source ip and the number of received and loss packets recorded, while Table 4 shows received and loss packets for scenario 2 which test flooding syn packets with the use of fake source

Table 3. Percentage of packets (real source ip)

Machine Name	*Transmitted as per (Kali)*	*Received as per (Wireshark)*	*Percentage of Received Packets*	*Percentage of Packets Loss*
Windows 8.1 (T1)	557089	454686	81.6%	18.4%
Windows 10 (T2)	1007175	931827	92.5%	7.5%
Metasploitable (T3)	250875	177902	70.9%	29.1%

Table 4. Percentage of packets (fake source ip)

Machine Name	*Transmitted as per (Kali)*	*Received as per (Wireshark)*	*Percentage of Received Packets*	*Percentage of Packets Loss*
Windows 8.1 (T1)	485550	485482	99.9%	0.1%
Windows 10 (T2)	825183	755304	91.5%	8.5%
Metasploitable (T3)	758283	645467	85.1%	14.9%

Table 5. Percentage of packets (random source ip)

Machine Name	*Transmitted as per (Kali)*	*Received as per (Wireshark)*	*Percentage of Received Packets*	*Percentage of Packets Loss*
Windows 8.1 (T1)	897623	704365	78.5%	21.5%
Windows 10 (T2)	1450581	1192845	82.2%	17.8%
Metasploitable (T3)	1141731	954584	83.6%	16.4%

ip. Moreover, Table 5 indicates received and loss packets based on using random source ip. Table 6 summarized all three scenarios to identify calculating percentage of packet ratio that collected from Wireshark.

Syn flood attack impact the delivery of packets. This can be seen in all three scenarios, the packets will not receive perfectly at receiver side, therefore there's different percentage in the rate of packet loss.

The following Fig. 9 displays the overall packet ratio for three scenarios.

The detailed evaluation and analysis demonstrate in following:

For scenario 1, the attacker used real ip to exploit the victims, so there will be dedicated infection perimeter from one source to a destination to flood the packets. Windows 10 achieve highest percentage of packet ratio which gives sign that it affected

Table 6. General packet ratio for all scenarios

Machine Name	Scenario 1	Scenario 2	Scenario 3
Windows 8.1 (T1)	81.6%	99.9%	78.5%
Windows 10 (T2)	92.5%	91.5%	82.2%
Metasploitable (T3)	70.9%	85.1%	83.6%

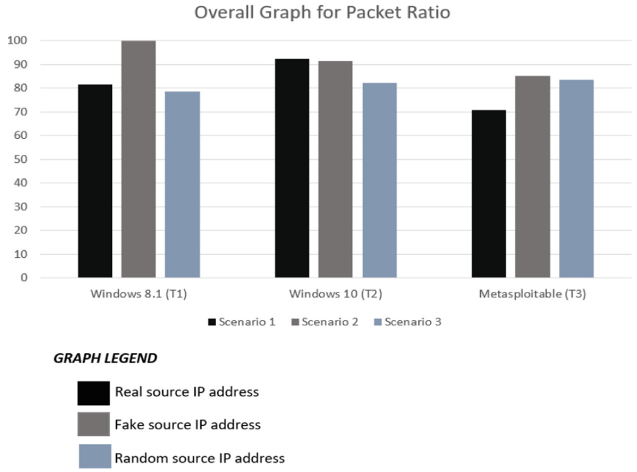


Fig. 9. Overall packet ratio for all scenarios

with syn packets more than other nodes and the percentage of packet loss decrease to be (7.5%) compare with the rest machines. The reason could be there's low latency and high bandwidth in the network that enable the machine to receive more packets. Even though, the attacker will consume the bandwidth and affect machine performance at end. In other side, the percentage of dropped packets grow in metasploitable to reach (29.1%). In general, Wireshark tool capable to show the significant information about violate machine, because real ip used in this case.

Scenario 2 relays on using spoof ip address to send syn packets, so the identity of the attacker will be hidden and not detected by Wireshark. General observation and analysis show that packet ratio rate is raised in this scenario compared to scenario 1 and scenario 3. As it's noticed there's a significant increase in packet ratio that received by Windows 8.1 machine to be (99.9%) compare with other victim machines. One factor could be increase in the bandwidth and throughput of the machine that make network connectivity good. Because of the attack, this machine will be infected by flood packets which lead to consume the resources. The results show packet loss percentages for Windows 10 and metasploitable as (8.5% and 14.9%) progressively. One reason is due to a lot of traffic arrived at these machines that lead to extent peak time, so network traffic will reach to maximum limit which cause packets loss and slow down the performance.

One of the major factors that cause packet loss is congestion that happened due to flooding huge number of syn packets as DoS. In scenario 3, the attacker used multiple random ip addresses that create botnet cluster to target one machine at time, so there will be more network congestion, and this lead to discard a lot of packets (as it's observed the percentage of loss packet in scenario 3 has dramatic increase and at the same time there's decrease in packet delivery). Because of network congestion, the bandwidth will be consumed rapidly. Practically, this will not affect attacker machine due to using of *flood* flag that ignore any response (Syn-Ack) from victim, but significant impact will be occurred in target machines. Both, Windows 10 and metasploitable machines achieved convergent percentage in packet ratio that equal to (82.2% and 83.6%) in sequence.

4.2 Wireshark Tool Capabilities

This section discusses Wireshark's capabilities and effectiveness as determined based on the results of a penetration test that was conducted in this paper for three scenarios. Wireshark is a packet analysis tool that can assist investigators in gathering digital evidence to identify illegal activity. This tool is more widely used that's because of it's features and characteristics, some examples of features are ability to filter the protocols, provide coding colors to distinguish between different forms of traffic and using GUI to display captured packets. On the other hand, there're some weaknesses and limitations of using Wireshark that can be observed in this test. More detail will be expressed in the following:

Strengths:

- Ability to detect syn packets that transmitted from attacker device to victim. SYN data was displayed on the tool's UI.
- Wireshark is user friendly tool that uses GUI and provide filtering option to determine precise requirements as needed.
- Show graphs and statistical information of the collected packets to enhance analyzing process.
- Displaying details of attacking machine such as ip address and port number.
- The ability to capture packets online and preserve them for offline analysis.
- Once the administrator monitors the traffic, the abnormal activities will be displayed by this tool, so it can be considered as Intrusion Detection System (IDS).
- Support cybersecurity specialist to implement the necessary security controls/polices.
- Applicable with different platforms of operating systems as well as new protocols can be added in this tool whenever there's updates.

Limitations/Weaknesses:

- Unable to send notification in case there's malicious activity detected, so the system specialist needs to access the tool and capture the traffic in order to get awareness about the attacks.
- Wireshark is an open-source tool that is accessible for free on the internet to everyone, including attackers.

- Unable to identify ip address of attacker machine while using random or fake ip (Reliability).
- Can be consider as vulnerable tool if the user utilizes it for illegal actions.
- Ineffective to prevent the attacks/threats.

5 Conclusion and Future Work

Network vulnerabilities can be described as flaws and weaknesses in the network that can be exploited by attackers. In addition, administrator should be aware about these weaknesses to make immediate action and to protect the data, devices and entire network. Basically, network security will support to ensure the implementation of the rules and policies in order to provide a high level of security that works against unethical activities. Security tools become very critical task for any organization, therefore new tools are constantly being developed to prevent network attacks.

This study analyzes network traffic by launching a tcp syn flood attack on different network nodes and Wireshark tool used to track packets flow. Furthermore, the implementation of attack carried out in three distinct scenarios which are using real source ip address, spoof source ip address and random source ip address. As protection mechanism, intrusion prevention system (IPS) needs to be used in any machine to stop the traffic once the attack has been identified as well as it required to turn on a firewall to filter network traffic.

Based on the findings, it was determined that Wireshark is a powerful packet analysis tool that can record syn packets. According to data statistics produced by Wireshark, there is a high percentage of packet ratio which gives indicator that; the attacker takes advantage of the machines' weaknesses and able to waste performance. Furthermore, Wireshark able to display attacker machine's information which make it easier for the administrator to block the ip address. Also, protocol statistics and graph representation can be generated by this tool. Wireshark provides the feature of saving ". pcap" file for each scenario, so additional evaluations can be performed offline as needed.

In contrast, additional improvements required to apply to this tool which help to reduce drawbacks. For instance, apply restriction on use, so attacker can't utilize it for illegal purposes. Might be upgraded to enhance extra security features like delivering alert to system administrator for infected network/systems. Also, more information should be specified by Wireshark whenever there's spoof or random source ip address, therefore the reliability of this tool will be increased.

References

1. Acosta, J.C., Krych, D.E.: Hands-on cybersecurity studies: uncovering and decoding malware communications-initial analysis with wireshark and volatility. Tech. rep., US Army Combat Capabilities Development Command Army Research Lab- oratory Adelphi (2021)
2. Bagyalakshmi, G., Rajkumar, G., Arunkumar, N., Easwaran, M., Narasimhan, K., Elamaran, V., Solarte, M., Hern´andez, I., Ramirez-Gonzalez, G.: Network vulnerability analysis on brain signal/image databases using nmap and wireshark tools. Ieee Access 6, 57144–57151 (2018). doi: https://doi.org/10.1109/access.2018.2872775

3. Banu, R., Jyothi, T., Amulya, M., Anju, K., Raju, A., Kashyap, S.N.: Monosek–a network packet processing system for analysis & detection of tcp xmas attack using pattern analysis. In: 2019 International Conference on Intelligent Computing and Control Systems (ICCS). pp. 952–956. IEEE (2019). doi: https://doi.org/10.1109/iccs45141.2019.9065325
4. Charles, A.J., Kalavathi, P.: Qos measurement of rpl using cooja simulator and wireshark network analyser. International Journal of Computer Sciences and Engineering 6(4), 283–291 (2018)
5. Dang, V.T., Huong, T.T., Thanh, N.H., Nam, P.N., Thanh, N.N., Marshall, A.:Sdn-based syn proxy—a solution to enhance performance of attack mitigation under tcp syn flood. The Computer Journal 62(4), 518–534 (2019). doi: https://doi.org/10.1093/comjnl/bxy117
6. Dodiya, B., Singh, U.K.: Malicious traffic analysis using wireshark by collection of indicators of compromise. International Journal of Computer Applications 975, 8887
7. Fidele, K.A., Syafei, W.A., et al.: Denial of service (dos) attack identification and analyse using sniffing technique in the network environment. In: E3S Web of Conferences. vol. 202, p. 15003. EDP Sciences (2020)
8. Goldschmidt, P., Kuˇcera, J.: Defense against syn flood dos attacksˇ using network-based mitigation techniques. In: 2021 IFIP/IEEE International Symposium on Integrated Network Management (IM). pp. 772–777. IEEE (2021)
9. Goyal, P., Goyal, A.: Comparative study of two most popular packet sniffing tools-tcpdump and wireshark. In: 2017 9th International Conference on Computational Intelligence and Communication Networks (CICN). pp. 77–81. IEEE (2017). doi: https://doi.org/10.1109/cicn.2017.8319360
10. Guo, X., Gao, X.: A syn flood attack detection method based on hierarchical multihead self-attention mechanism. Security & Communication Networks (2022). doi: https://doi.org/10.1155/2022/8515836
11. Iqbal, H., Naaz, S.: Wireshark as a tool for detection of various lan attacks. Inter-national Journal of Computer Science and Engineering 7(05), 833–837 (2019)
12. Jain, G., et al.: Application of snort and wireshark in network traffic analysis. In: IOP Conference Series: Materials Science and Engineering. vol. 1119, p. 012007. IOP Publishing (2021). doi: https://doi.org/10.1088/1757-899x/1119/1/012007
13. Kaur, G., Bhatia, N.: Wireshark–packet capture tool (2018)
14. Musa, A.: Forensic analysis of peer-to-peer network traffic with wireshark. SLU Journal of Science and Technology 1(2), 92–99 (2020)
15. Musa, A., Abubakar, A., Gimba, U.A., Rasheed, R.A.: An investigation into peer-to-peer network security using wireshark. In: 2019 15th International Conference on Electronics, Computer and Computation (ICECCO). pp. 1–6. IEEE (2019). doi: https://doi.org/10.1109/icecco48375.2019.9043236
16. Navabud, P., Chen, C.L.: Analyzing the web mail using wireshark. In: 2018 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD). pp. 1237–1239. IEEE (2018). doi: https://doi.org/10.1109/fskd.2018.8686871
17. Pandey, R., Jyothindar, V., Chopra, U.K.: Vulnerability assessment and penetration testing: a portable solution implementation. In: 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN). pp. 398–402. IEEE (2020). doi: https://doi.org/10.1109/cicn49253.2020.9242640
18. Sanders, C.: Practical Packet Analysis, 3E: Using Wireshark to Solve Real-World Network Problems. No Starch Press (2017)
19. Sandhya, S., Purkayastha, S., Joshua, E., Deep, A.: Assessment of website security by penetration testing using wireshark. In: 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS). pp. 1–4. IEEE (2017). doi: https://doi.org/10.1109/icaccs.2017.8014711

20. Sasi, G., Thanapal, P., Balaji, V., Babu, G.V., Elamaran, V.: A handy approach for teaching and learning computer networks using wireshark. In: 2020 Fourth International Conference on Inventive Systems and Control (ICISC). pp. 456–461. IEEE (2020). doi: https://doi.org/10.1109/icisc47916.2020.9171197
21. Siswanto, A., Syukur, A., Kadir, E.A., et al.: Network traffic monitoring and analysis using packet sniffer. In: 2019 International Conference on Advanced Communication Technologies and Networking (CommNet). pp. 1–4. IEEE (2019). doi: https://doi.org/10.1109/commnet.2019.8742369
22. Steve Gibson, "Gibson Research Corporation" grc.com. https://www.grc.com/port_139.htm
23. Tanner, N.H.: Wireshark (2019)
24. Wahid, A., Firdaus, M.E., Parenreng, J.M.: The implementation of wireshark and iptables firewall collaboration to improve traffic security on network systems. Internet of Things and Artificial Intelligence Journal 1(4), 249–264 (2021)