

Indian Institute of Technology, Roorkee



Project Report **Simulation and analysis of network attack using Snort**

Project Guide

Dr. Anjali Sardana

Asst. Proffessor

Electronics and Computer Science Department

Project Group

Amol Deshmukh 11536008

Ankit Jain 11536003

Ankit Raj Garg 11535006

Lokesh Agrawal 11536014

Mitul Shah 11535027

Mayank Gupta 11536015

Subrata Biswas 11535028

Contents

1	Objective	1
2	Motivation	1
3	Introduction	1
3.1	What Is an Intrusion-Detection System (IDS)?	1
3.2	What Is an Intrusion-Prevention System (IPS)?	2
4	Attacks	2
4.1	Types Of Attacks	2
4.1.1	ICMP Flood/Ping Flood:	2
4.1.2	SYN Flood:	2
4.1.3	Fraggle Attack:	3
4.1.4	Land Attack:	3
5	SOFTWARE/TOOLS USED	3
5.1	Snort	3
5.2	HPING3	4
5.3	Wireshark	4
5.4	VMWare	4
6	IMPLEMENTATION	5
6.1	PING FLOOD	5
7	SYN FLOOD	7
7.1	FRAGGLE ATTACK	9
7.2	LAND ATTACK	11
8	CONCLUSION:	14
	References	15

Abstract

The project aims at developing a mechanism which detects various kinds of attacks on the network using Snort as a detection tool. A user in a network is prone to be attacked by other malicious users. It is not uncommon to hear about virus attacks and wasting up of bandwidth with useless packets. As a part of the Intrusion Detection Systems course project, it gave us an opportunity to learn about the possible attacks from the network and understand the working of antivirus and firewall which detect them. We are making a basic model which can be used for a set of attacks.

1 Objective

The project aims at developing a mechanism which detects various kinds of attacks from the network using Snort as a detection tool.

2 Motivation

A user in a network is prone to be attacked by other malicious users. It is not uncommon to hear about virus attacks and wasting up of bandwidth with useless packets. As a part of the Intrusion Detection Systems course project, we wanted to learn about the possible attacks from the network and understand the working of antivirus and firewall which detect them. We are making a basic model which can be used for a set of attacks.

3 Introduction

The focus of this project is to produce a mechanism using a Network Intrusion Detection System (NIDS) to monitor a network and find types of malicious activities. The possible threats are a part of the system specifications. The implementation has been done using Snort IDS for detecting the threat. It is a tool monitoring the traffic on a network interface. It can perform real-time traffic analysis and packet logging on Internet Protocol (IP) networks. We are using the alerts generated by Snort identified threats.

3.1 What Is an Intrusion-Detection System (IDS)?

An intrusion-detection system (IDS) can be defined as the tools, methods, and resources to help identify, assess, and report unauthorized or unapproved network activity. An IDS does not actually detect intrusionsit detects activity in traffic that may or may not be an intrusion.

3.2 What Is an Intrusion-Prevention System (IPS)?

It is still early in the development of intrusion-prevention systems (IPSs), but generally an IPS sits inline on the network and monitors it, and when an event occurs, it takes action based on prescribed rules. This is unlike IDSs, which do not sit inline and are passive.

4 Attacks

A denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a computer resource unavailable to its intended users. Saturating the target machine with external communications requests, such that it cannot respond to legitimate traffic, or responds so slowly as to be rendered effectively unavailable, comes under this category of attack. The implementation involves consumption of its resources so that it can no longer provide its intended service or obstruction of the communication media between the intended users and the victim so that they can no longer communicate adequately. According to the United States Computer Emergency Readiness Team (US-CERT), the symptoms of denial-of-service attacks include: Unusually slow network performance (opening files or accessing web sites) Unavailability of a particular web site Inability to access any web site Dramatic increase in the number of spam emails received

Among the various possible network threats, the ones dealt in this project are as follows:

4.1 Types Of Attacks

4.1.1 ICMP Flood/Ping Flood:

Ping is a utility in which data packets are forwarded for checking the quality of a link or for verifying the connection of a machine to the Internet. In the ping program one 64-byte datagram per second is sent. Ping flooding occurs when number of pings received per unit time is much higher than the normal value. It emits ICMP echo requests at the highest possible frequency. The processing of such an amount of ICMP requests/replies might cause an extreme CPU load and thus clog the network bandwidth which can eventually lead to exhaustion of the bandwidth.

4.1.2 SYN Flood:

To establish a TCP connection between two nodes (say node A and node B), there is an exchange a series of messages between them, which runs like the following:

1. Node A requests a connection by sending a SYN (synchronize) message to Node B.

2. Node B acknowledges this request by sending SYN-ACK back to Node A.
3. Node A responds with an ACK, and the connection is established.
4. This is the TCP three-way handshake and is mandatory for every connection established using the TCP protocol.

A SYN flood occurs when a host sends a burst of TCP/SYN packets, often with a forged sender address as clients. Each of these packets is handled by the server like a connection request, causing the server to spawn a half-open connection, by sending back a TCP/SYN-ACK packet, and waiting for a TCP/ACK packet in response from the sender address. However, because the sender address is forged, the response to SYN-ACK never comes. These half-open connections saturate the number of available connections the server is able to make, keeping it from responding to legitimate requests until after the attack ends.

4.1.3 Fraggle Attack:

A much more insidious attack is generated by sending spoofed UDP packets to port 7 (echo) or port 19 (chargen). This attack, called the fraggle attack, is generated by an attacker who sets both the source and destination ports to one or the other of these on two hosts that the attacker is targeting. An unending series of packets will be transmitted between the two systems as they ping-pong back and forth. Best practices generally recommend that these ports, and others with little or no reason to be exposed on the Internet, be blocked at the perimeter. Hosts usually also have no need to expose these ports and generally should disable them.

4.1.4 Land Attack:

Another similar attack (which most modern systems guard against) is the so-called land attack. In this attack, the source and destination IP addresses are both set to the victims address, and the port is set to UDP 7 or 19, as in the fraggle attack. Of course, a packet of this nature is always synthetically generated, since no host will send a packet to itself over the network but would instead route the data internally. On a vulnerable host, this packet will cause it to ping-pong with itself, causing CPU utilization to zoom up to 100 percent

5 SOFTWARE/TOOLS USED

5.1 Snort

An Intrusion Detection System (IDS) is a software tool used to monitor network and/or system activities for malicious activities or policy violations or unauthorized access to a

computer system or network. An IDS is characterized by the location and the type of its sensor. The engine behavior and its way to alert and log an event are also key settings to consider. Alerts are generated when attacks are detected. The alerts generated could also be false. The terminologies used for the validness of the alert are:

- True Positive: A legitimate attack which triggers IDS to produce an alarm.
- False Positive: An event signaling IDS to produce an alarm without any attack.
- False Negative: A failure of IDS to detect an actual attack.
- True Negative: When no attack has taken place and no alarm is raised.

Snort is a free and open source network intrusion prevention system (NIPS) and network intrusion detection system (NIDS). Snorts open source network-based intrusion detection system (NIDS) has the ability to perform real-time traffic analysis and packet logging on Internet Protocol (IP) networks. It can be tailored as per individual requirement. Snort provides commands to the user to specify attack signatures in the form of rules.

The software formed our basis for generation of alerts, in which we disabled all prewritten rules and wrote our rules as a separate entity.

5.2 HPING3

This software helps in generating SYN flood attack as it has the capability of setting individual flags while sending the TCP packets. This command also has the ability of sending packets at a very high rate which can even consume the entire bandwidth in some scenarios.

5.3 Wireshark

In order to check whether the firewall is effective against the worm attack, we use wireshark to see the retransmissions when the destination note has updated its IP tables to drop the malicious packets.

5.4 VMWare

VMWare is used to create virtual environment for performing intrusion detection experiments. This network includes attacking host, normal and detecting host.

6 IMPLEMENTATION

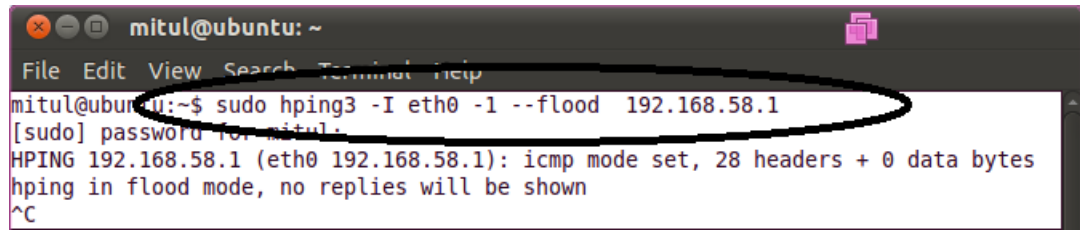
In this project, the above mentioned attacks are being detected in real time. It basically involves a 3-tier implementation:

1. Generating an attack
2. Detection of attack with Snort
3. Alerting the administrator

Syntax of a Snort rule: action $\langle \text{src-ip} \rangle \langle \text{src-port} \rangle \rightarrow \langle \text{dst-ip} \rangle \langle \text{dst-port} \rangle$ (options)

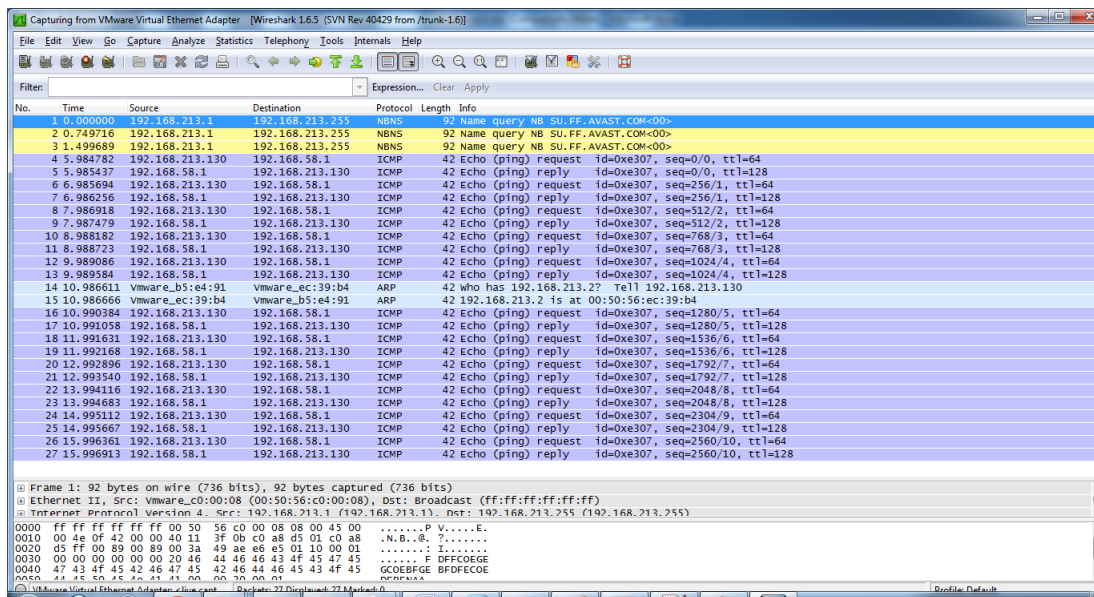
6.1 PING FLOOD

1. Generation: Using the command: `hping3 -i1 -1 --flood $\langle \text{dst-ip} \rangle$`



```
mitul@ubuntu: ~  
File Edit View Search Terminal Help  
mitul@ubuntu:~$ sudo hping3 -I eth0 -1 --flood 192.168.58.1  
[sudo] password for mitul:  
HPING 192.168.58.1 (eth0 192.168.58.1): icmp mode set, 28 headers + 0 data bytes  
hping in flood mode, no replies will be shown  
^C
```

2. Detection: In the snort based rule we utilized the count in time feature where it counts all the ICMP packets which have been retrieved from that IP for a specified amount of time. Thus if the count is more than a threshold in a specified time are classified as a ping flood attack.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.213.1	192.168.213.255	NBNS	92	Name query NB SU.FF.AVAST.COM<00>
2	0.749716	192.168.213.1	192.168.213.255	NBNS	92	Name query NB SU.FF.AVAST.COM<00>
3	1.499689	192.168.213.1	192.168.213.255	NBNS	92	Name query NB SU.FF.AVAST.COM<00>
4	5.984782	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=0/0, ttl=64
5	5.985437	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=0/0, ttl=128
6	6.985694	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=256/1, ttl=64
7	6.986256	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=256/1, ttl=128
8	7.986918	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=512/2, ttl=64
9	7.987479	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=512/2, ttl=128
10	8.988182	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=768/3, ttl=64
11	8.988723	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=768/3, ttl=128
12	9.989086	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=1024/4, ttl=64
13	9.989584	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=1024/4, ttl=128
14	10.986611	Vmware_b5:e4:91	Vmware_ec:39:b4	ARP	42	Who has 192.168.213.2? Tell 192.168.213.130
15	10.986666	Vmware_ec:39:b4	Vmware_b5:e4:91	ARP	42	192.168.213.2 is at 00:50:56:ec:39:b4
16	10.990384	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=1280/5, ttl=64
17	10.991058	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=1280/5, ttl=128
18	11.991631	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=1536/6, ttl=64
19	11.992168	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=1536/6, ttl=128
20	12.992896	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=1792/7, ttl=64
21	12.993540	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=1792/7, ttl=128
22	13.994116	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=2048/8, ttl=64
23	13.994683	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=2048/8, ttl=128
24	14.995112	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=2304/9, ttl=64
25	14.995667	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=2304/9, ttl=128
26	15.996361	192.168.213.130	192.168.58.1	ICMP	42	Echo (ping) request id=0xe307, seq=2560/10, ttl=64
27	15.996913	192.168.58.1	192.168.213.130	ICMP	42	Echo (ping) reply id=0xe307, seq=2560/10, ttl=128

3. Snort rule: alert icmp any any → any any (itype:8; threshold: type threshold, track by-src, count 20, seconds 3; msg:"Ping flood attack detected!";sid:100121;)

```

C:\Windows\system32\cmd.exe - snort -il -l c:\snort\log -c c:\Snort\rules\myrules.rules -K ascii
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Mitul>cd\
C:\>cd Snort\bin\
C:\Snort\bin>snort -il -l c:\snort\log -c c:\Snort\rules\myrules.rules -K ascii
Running in IDS mode

==== Initializing Snort ====
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "c:\Snort\rules\myrules.rules"
Tagged Packet Limit: 256
Log directory = c:\snort\log

*****
Initializing rule chains...
WARNING: c:\Snort\rules\myrules.rules(6) threshold <in rule> is deprecated; use
detection_filter instead.

4 Snort rules read
4 detection rules
  
```

4. Alert Generated: 04/12-20:37:13.784560 [**] [1:100121:0] Ping flood attack detected! [**] [Priority: 0] {ICMP} 172.27.19.41 → 172.27.20.50

```

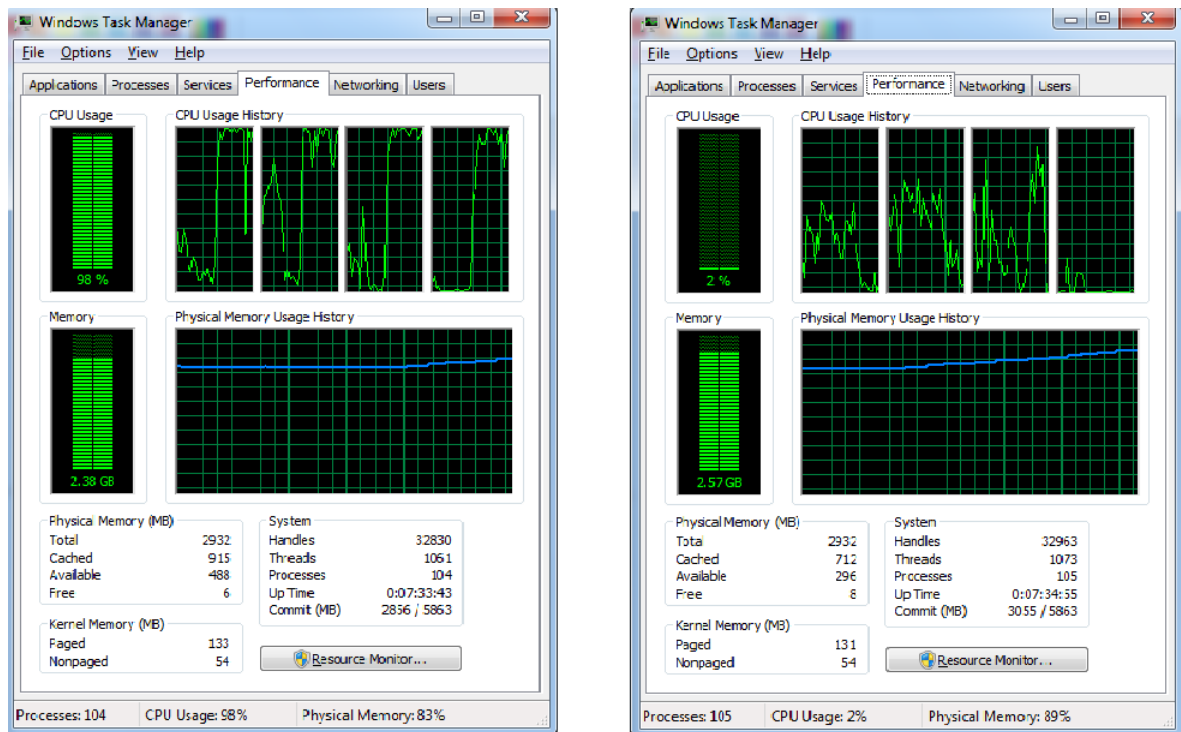
ICMP_ECHO.ids - Notepad
File Edit Format View Help
[**] Ping Flood Attack! [**]
04/11-03:48:30.652364 192.168.213.130 -> 192.168.58.1
ICMP TTL:64 TOS:0x0 ID:46923 IpLen:20 DgmLen:28
Type:8 Code:0 ID:37127 Seq:25089 ECHO
+++++
+==+

[**] Ping Flood Attack! [**]
04/11-03:48:30.654017 192.168.213.130 -> 192.168.58.1
ICMP TTL:64 TOS:0x0 ID:8595 IpLen:20 DgmLen:28
Type:8 Code:0 ID:37127 Seq:26113 ECHO
+++++
+==+

[**] Ping Flood Attack! [**]
04/11-03:48:30.654446 192.168.213.130 -> 192.168.58.1
ICMP TTL:64 TOS:0x0 ID:58337 IpLen:20 DgmLen:28
Type:8 Code:0 ID:37127 Seq:27137 ECHO
+++++
+==+

[**] Ping Flood Attack! [**]
04/11-03:48:30.654980 192.168.213.130 -> 192.168.58.1
ICMP TTL:64 TOS:0x0 ID:49832 IpLen:20 DgmLen:28
Type:8 Code:0 ID:37127 Seq:28161 ECHO
  
```


5. Performance



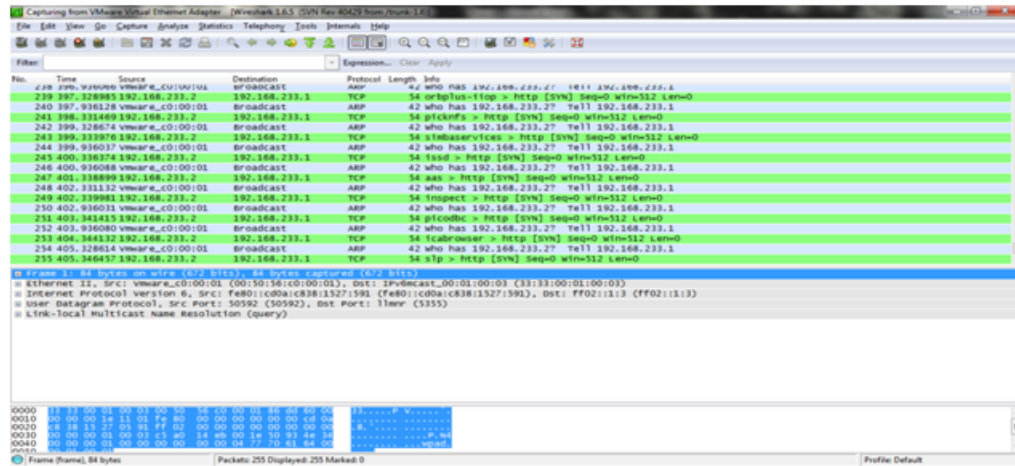
7 SYN FLOOD

1. Generation:

```
mitul@ubuntu: ~  
File Edit View Search Terminal Help  
mitul@ubuntu:~$ sudo hping3 -I eth0 --flood -a 10.0.0.1 -S 192.168.58.1 -p 80  
HPING 192.168.58.1 (eth0 192.168.58.1): S set, 40 headers + 0 data bytes  
hping in flood mode, no replies will be shown  
^Z
```

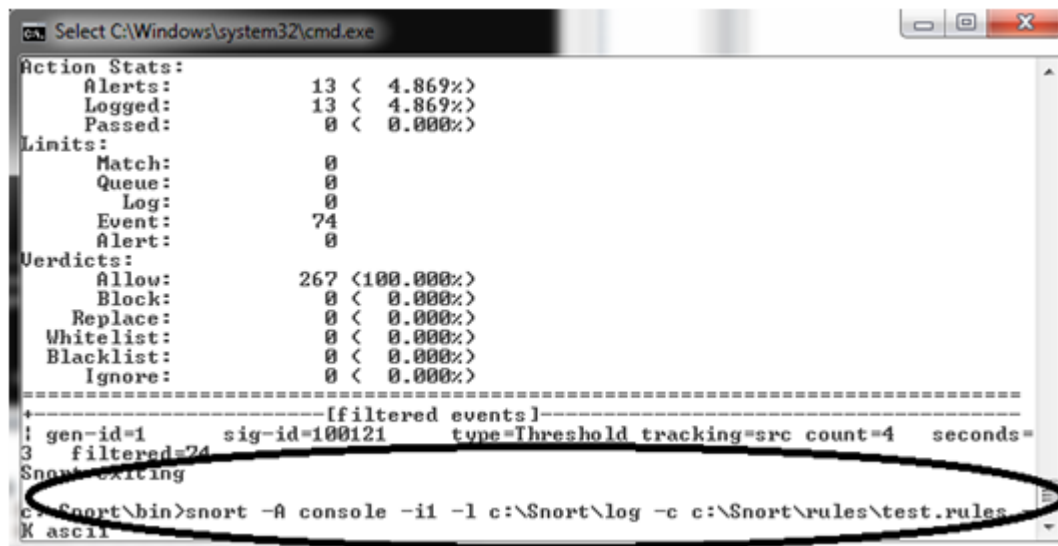
Using the command: `hping3 -I eth0 --flood -a <dst-ip> -S <src-ip> -p <port-no>`

2. Detection:



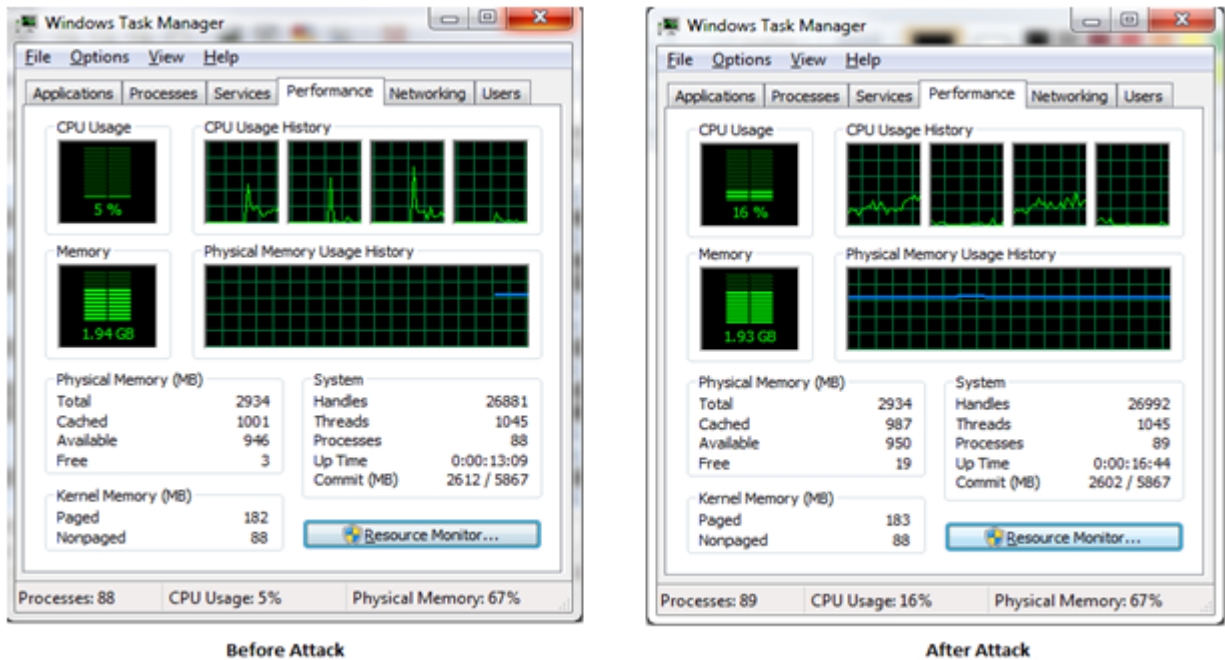
For the detection of SYN flood attack, we have used the count in time feature to count all the TCP packets with their S flag (SYN flag) set which have been received from a particular IP within a specified amount of time. Thus, if too many SYN packets are received within a stipulated time interval, it is classified as a ping flood attack.

3. Snort rule:



alert tcp any any → any 80 (flag:S; threshold: type threshold, track by-dst,count 5000, seconds 1; msg:"DoS SYN flood attack detected!"; sid:12121; rev:1;)

4. Alert Generated:



[**] DoS SYN attack detected! [**] 04/10-16:18:09.408675 192.168.58.1:1412 → 192.168.58.1:7
TCP TTL:64 TOS:0x0 ID:903 IpLen:20 DgmLen:40 ***** Seq: 0x4D71FAB2 Ack:
0x33D4ACC8 Win: 0x200 TcpLen: 20

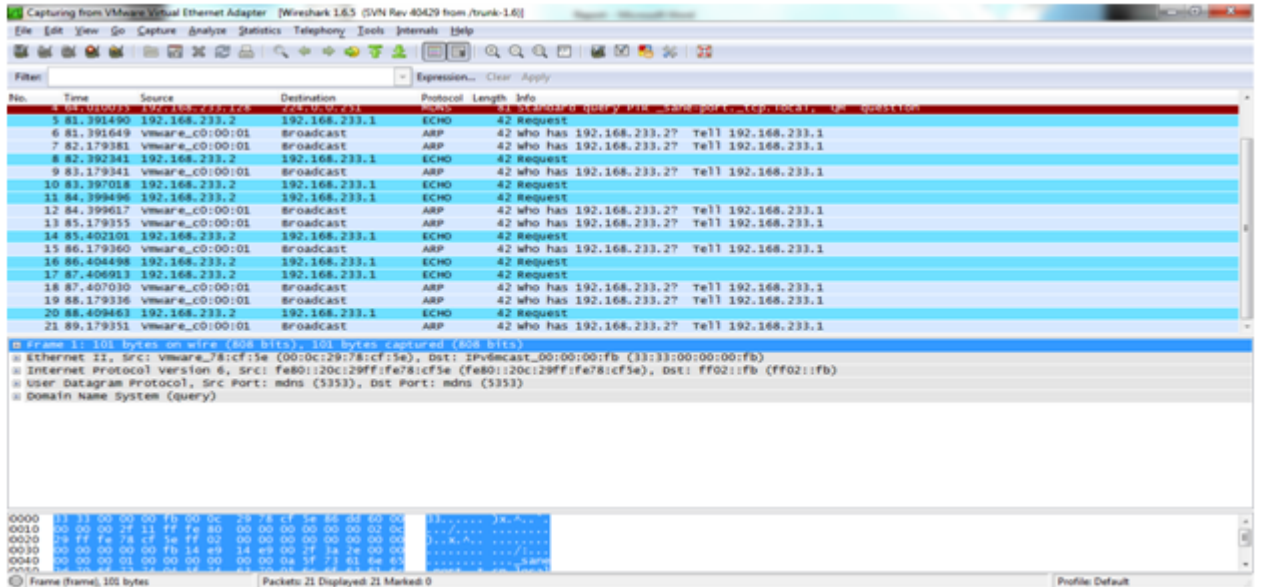
7.1 FRAGGLE ATTACK

1. Generation:

```
mitul@ubuntu: ~  
File Edit View Search Terminal Help  
mitul@ubuntu:~$ sudo hping3 -I eth0 --flood -a 10.0.0.1 192.168.58.1 -p 7  
HPING 192.168.58.1 (eth0 192.168.58.1): NO FLAGS are set, 40 headers + 0 data byte  
S
```

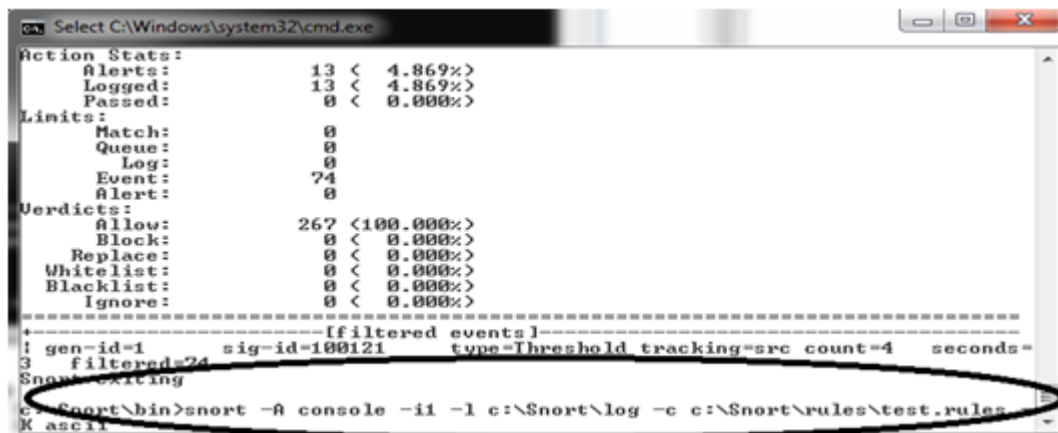
Using the command: `hping3 i< interface-no > -a < spoofed-ip > < dst-ip > -p < 7 or 19 >`

2. Detection:



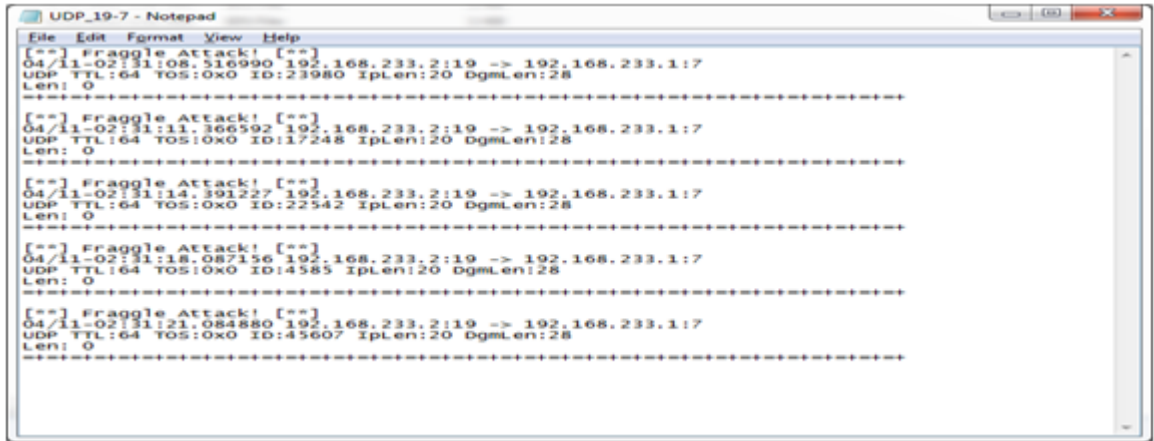
For the detection of fraggle attack, we have used the count in time feature to count all UDP packets to port 7 (echo) or port 19 (chargen) which have been received from a particular IP within a specified amount of time. Thus, if too many packets are received within a stipulated time interval, it is classified as a fraggle attack.

3. Snort rule:



alert udp any [7,19] → any any (msg:"Dos Fraggle Attack!"; sid: 122555; gid: 100000; detection-filter: track by-src, count 100, seconds 1;)

4. Alert Generated:



```
UDP_19-7 - Notepad
File Edit Format View Help
[**] Fraggle Attack! [**]
04/11-02:31:08.516990 192.168.233.2:19 -> 192.168.233.1:7
UDP TTL:64 TOS:0x0 ID:23980 IpLen:20 DgmLen:28
Len: 0

[**] Fraggle Attack! [**]
04/11-02:31:11.366592 192.168.233.2:19 -> 192.168.233.1:7
UDP TTL:64 TOS:0x0 ID:17248 IpLen:20 DgmLen:28
Len: 0

[**] Fraggle Attack! [**]
04/11-02:31:14.391227 192.168.233.2:19 -> 192.168.233.1:7
UDP TTL:64 TOS:0x0 ID:22542 IpLen:20 DgmLen:28
Len: 0

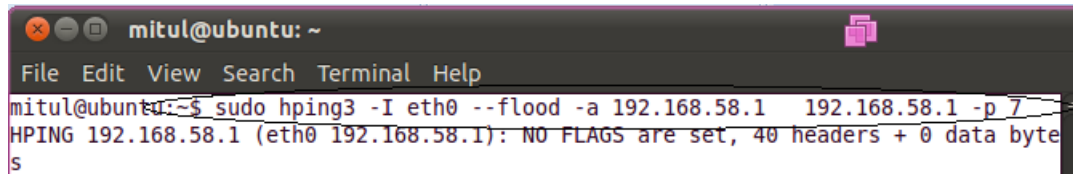
[**] Fraggle Attack! [**]
04/11-02:31:18.087156 192.168.233.2:19 -> 192.168.233.1:7
UDP TTL:64 TOS:0x0 ID:4585 IpLen:20 DgmLen:28
Len: 0

[**] Fraggle Attack! [**]
04/11-02:31:21.084880 192.168.233.2:19 -> 192.168.233.1:7
UDP TTL:64 TOS:0x0 ID:45607 IpLen:20 DgmLen:28
Len: 0
```

[**] DoS Fraggle attack! [**] 04/10-16:18:09.408675 192.168.58.1:1412 → 192.168.58.1:7
TCP TTL:64 TOS:0x0 ID:903 IpLen:20 DgmLen:40 ***** Seq: 0x4D71FAB2 Ack:
0x33D4ACC8 Win: 0x200 TcpLen: 20

7.2 LAND ATTACK

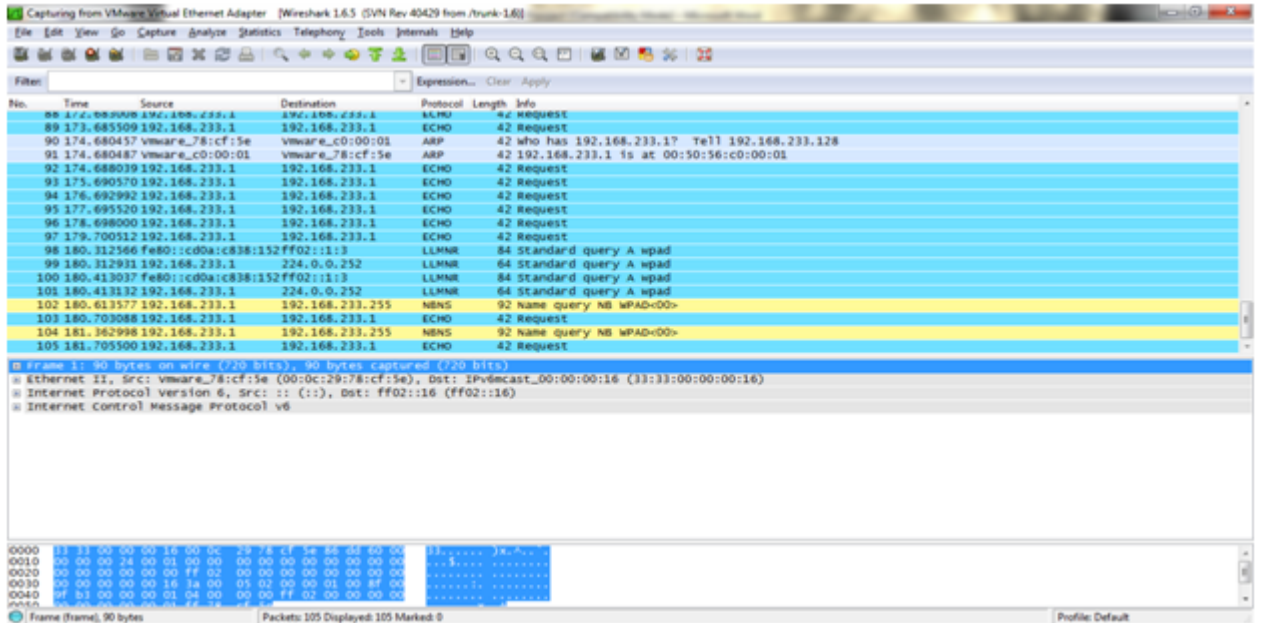
1. Generation:



```
mitul@ubuntu: ~
File Edit View Search Terminal Help
mitul@ubuntu:~$ sudo hping3 -I eth0 --flood -a 192.168.58.1 192.168.58.1 -p 7
HPING 192.168.58.1 (eth0 192.168.58.1): NO FLAGS are set, 40 headers + 0 data byte
S
```

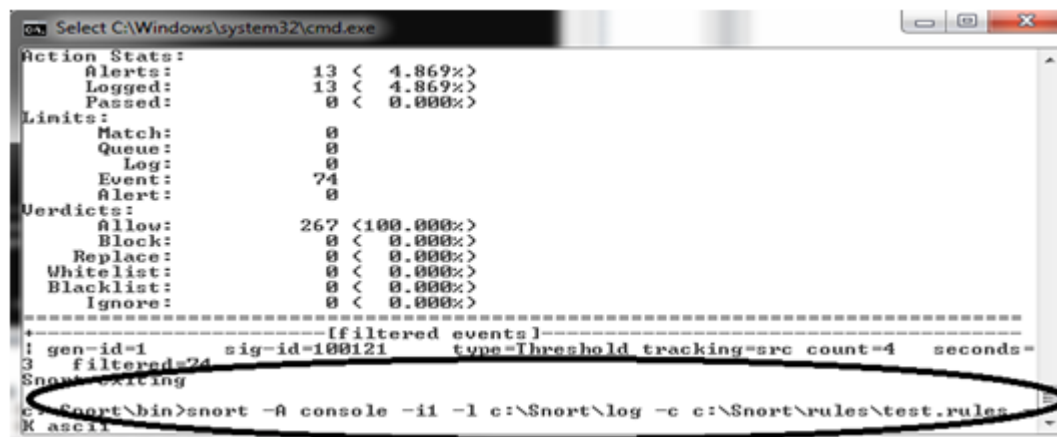
Using the command: `hping3 i< interface-no > -a < source-ip > < dst-ip > -p < 7 or 19 >`

2. Detection:



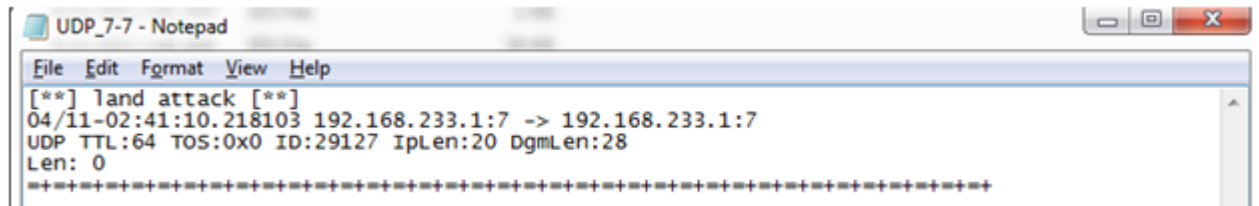
For the detection of Land attack, we have used the count in time feature to count all UDP packets to port 7 (echo) or port 19 (chargen) which have been received from a same source and destination IP address and both have to be same as host ip address. Thus, if too many packets are received within a stipulated time interval, it is classified as a Land attack.

3. Snort rule:



alert udp any [7,19] → any any (msg:"Land Attack"; sameip; sid: 100001; gid: 100001; detection-filter: track by-src, count 1, seconds 1;)

4. Alert Generated:



11/07-20:38:09.468017 [
[122:1:0] [
[Priority:3] PROTO:255 172.24.1.122 → 172.27.20.50

8 CONCLUSION:

The use of snort forms an effective way for detecting the intrusions from the network and thus can be used to make more robust network services. Our rules are only detect dos attack .IDS checks that any port receiving more than specified threshold packets will lead to generation of the alert message. We studied various tools like snort, latex and wireshark.

References

- [1] snort 2.9.5.2 <http://www.snort.org>
- [2] Wikipedia <http://en.wikipedia.org/>
- [3] Hping manual <http://www.hping.org>
- [4] Snort for dummies
- [5] Dos Rules <http://cvs.snort.org/viewcvs.cgi/snort/rules/dos.rules>
- [6] Snort Documentation <http://www.snort.com/docs>
- [7] Attacks <http://www.pcflank.com/expl-d.htm>
- [8] Writing Snort Rules <http://manual.snort.org/node27.html>
- [9] Tools <http://www.sectools.org/>
- [10] Hping ManualD <http://wiki.hping.org/94>