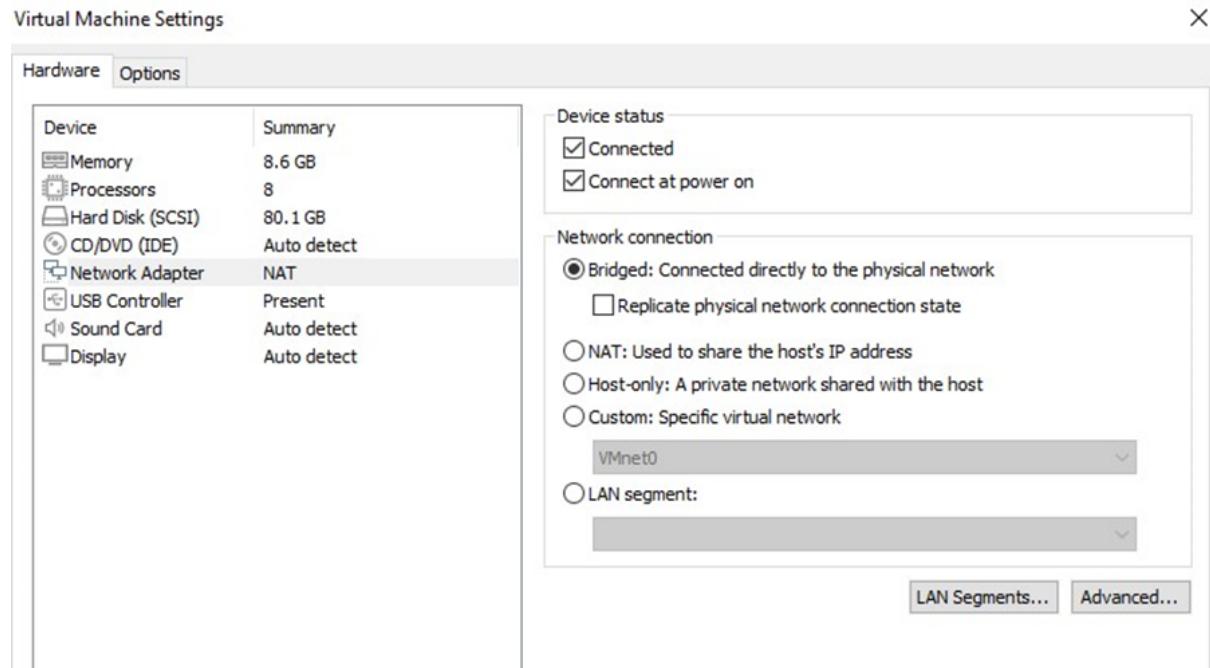


Introduction to Network Intrusion Detection with Snort

This lab focuses on installing and running Snort, identifying the home network, creating a simple rule, and configuring Snort to alert on specific types of attacks.

1) Network Setting in VirtualBox

Configure the network setting in Kali Linux as Bridged Adapter:



Make sure you have connection to the Internet: open a terminal and type the following command:

```
ping 8.8.8.8
```

A terminal window showing the output of a ping command. The prompt is '(kali㉿kali)-[~]'. The command '\$ ping 8.8.8.8' is entered, followed by three successful echo requests to Google's public DNS server. The final line shows the ping statistics: 3 packets transmitted, 3 received, 0% packet loss, time 2005ms, with round-trip times ranging from 39.054 to 49.637 ms.

```
(kali㉿kali)-[~]
$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=112 time=49.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=112 time=39.1 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=112 time=45.9 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 39.054/44.863/49.637/4.382 ms
```

2) Identify the Network IP address

Check Kali Linux IP address:

```
ifconfig
```

```
(kali㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.128 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::3db2:696b:fece:ac65 prefixlen 64 scopeid 0x20<link>
            ether 00:0c:29:65:ce:0c txqueuelen 1000 (Ethernet)
                RX packets 10 bytes 1760 (1.7 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 41 bytes 5230 (5.1 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 4 bytes 240 (240.0 B)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 4 bytes 240 (240.0 B)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

To Identify the network IP address, check the netmask for your IP address.

In the above screenshot, the netmask is 255.255.255.0

Kali IP Address: **192.168.1.128**

Netmask: 255.255.255.0

Network IP Address: 192.168.1.0/24

If you have the netmask 255.255.255.0 you need to add /24 at the end of your network IP address.

Each octet (255) is 8 bits. So, 8+8+8=24.

The network IP address will be used in the next step as the network we will be protecting.

3) Installing and testing Snort NIDS 1

To install Snort on Kali Linux, use the following commands:

sudo apt-get update

sudo apt-get install snort

```
(kali㉿kali)-[~]
$ snort --version

      -> Snort! <-
o"_)~ Version 2.9.7.0 GRE (Build 149)
     By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
     Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
     Copyright (C) 1998-2013 Sourcefire, Inc., et al.
     Using libpcap version 1.10.1 (with TPACKET_V3)
     Using PCRE version: 8.39 2016-06-14
     Using ZLIB version: 1.2.11
```

```
Snort successfully validated the configuration!
Snort exiting
```

4) Browsing Snort Rules

Change the directory to the rules' location, cd /etc/snort/rules/

List all the files in the directory ls

All Snort rules files are stored in this location.

```
(kali㉿kali)-[~]
└─$ cd /etc/snort/rules/
(kali㉿kali)-[/etc/snort/rules]
└─$ ls
attack-responses.rules      community-web-dos.rules    policy.rules
backdoor.rules               community-web-iis.rules   pop2.rules
bad-traffic.rules            community-web-misc.rules  pop3.rules
chat.rules                  community-web-php.rules  porn.rules
community-bot.rules          ddos.rules                 rpc.rules
community-deleted.rules     deleted.rules              rservices.rules
community-dos.rules          dns.rules                 scan.rules
community-exploit.rules     dos.rules                 shellcode.rules
community-ftp.rules          experimental.rules      smtp.rules
community-game.rules         exploit.rules             snmp.rules
```

Listing all the available rules in Snort rules directory.

```
(kali㉿kali)-[/etc/snort/rules]
└─$ ls
attack-responses.rules      community-web-dos.rules    policy.rules
backdoor.rules               community-web-iis.rules   pop2.rules
bad-traffic.rules            community-web-misc.rules  pop3.rules
chat.rules                  community-web-php.rules  porn.rules
community-bot.rules          ddos.rules                 rpc.rules
community-deleted.rules     deleted.rules              rservices.rules
community-dos.rules          dns.rules                 scan.rules
community-exploit.rules     dos.rules                 shellcode.rules
community-ftp.rules          experimental.rules      smtp.rules
community-game.rules         exploit.rules             snmp.rules
community-icmp.rules         finger.rules              sql.rules
community-imap.rules         ftp.rules                 telnet.rules
community-inappropriate.rules icmp-info.rules        tftp.rules
community-mail-client.rules  icmp.rules                virus.rules
community-misc.rules          imap.rules              web-attacks.rules
community-nntp.rules          info.rules              web-cgi.rules
community-oracle.rules        local.rules             web-client.rules
community-policy.rules       misc.rules              web-coldfusion.rules
community-sip.rules           multimedia.rules       web-frontpage.rules
community-smtp.rules          mysql.rules             web-iis.rules
community-sql-injection.rules netbios.rules          web-misc.rules
community-virus.rules         nntp.rules             web-php.rules
community-web-attacks.rules  oracle.rules            x11.rules
community-web-cgi.rules       other-ids.rules
community-web-client.rules   p2p.rules
```

Question 2: Pick any rule in “web-attacks.rules” and explain what that rule does. You can display all the rules in “web-attacks.rules” by typing the following command:

```
sudo mousepad /etc/snort/rules/web-attacks.rules
```

```
/code
```

Action	Protocol	Source IP	Source Port	Direction	Destination IP	Destination Port
Alert	TCP	\$EXTERNAL_NET	Any	->	\$HTTP_SERVERS	\$HTTP_PORTS

This rule will display an alert if packets came from any external port to the http ports

6) Create a simple rule

Let's create a Snort rule to detect any ping to 8.8.8.8 (Google DNS IP). Note that "ping" is using ICMP protocol.

```
alert icmp $HOME_NET any -> 8.8.8.8 any (msg:"My Goooooogle Ping";
classtype:bad-unknown; sid:999; rev:1;)
```

7) Adding Snort Rule

Now, we'll add our simple Snort rule created in the previous stem into "icmp.rules" file that contains all rules related to ICMP protocol.

Open the icmp.rules file

```
sudo mousepad /etc/snort/rules/icmp.rules
```

8) Running Snort NIDS and Detecting ICMP request (ping)

In this step, you need to open two command Terminals in Kali Linux. One to run Snort and the other one to run the ping command.

To run Snort, type this command but change the network IP address to yours

```
(kali㉿kali)-[~/etc/snort/rules] $ sudo snort -h 192.168.1.128 -A console -c /etc/snort/snort.conf
Running in IDS mode
```

- h to set the home network (\$HOME_NET)
- A console: Sends alerts to the console window.
- c Indicates which Snort configuration file to use

In the second terminal type the following ping command:

```
(kali㉿kali)-[~] $ ping 8.8.8.8 -c 2
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=112 time=52.3 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=112 time=38.3 ms

— 8.8.8.8 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 38.298/45.321/52.345/7.023 ms
```

```
Preprocessor object. Snort version 1.1 <built in>, GPLv2+ (2013-07-11)
Commencing packet processing (pid=69647)
12/30-13:52:01.880158 [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.1.128 → 8.8.8.8
12/30-13:52:01.880158 [**] [1:999:1] My Gooooogle Ping [**] [Classification: Potentially Bad Traffic] [Priority: 2] {ICMP} 192.168.1.128 → 8.8.8.8
12/30-13:52:01.880158 [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.1.128 → 8.8.8.8
12/30-13:52:01.932492 [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 8.8.8.8 → 192.168.1.128
12/30-13:52:02.882388 [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.1.128 → 8.8.8.8
12/30-13:52:02.882388 [**] [1:999:1] My Gooooogle Ping [**] [Classification: Potentially Bad Traffic] [Priority: 2] {ICMP} 192.168.1.128 → 8.8.8.8
12/30-13:52:02.882388 [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.1.128 → 8.8.8.8
12/30-13:52:02.920634 [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 8.8.8.8 → 192.168.1.128
```

As shown in the above screenshot, Snort has detected ICMP request (ping) according to the direction we have specified in Step 6 (from our HOME_NET to 8.8.8.8 (Google DNS IP) \$HOME_NET any -> 8.8.8.8):

11) Disable / Enable Snort Rules

Snort uses a configuration file called "snort.conf".

This file allows you to disable or enable rules before running Snort.

To edit the configuration file, run the following command:

```
sudo mousepad /etc/snort/snort.conf
```

The screenshot shows a terminal window with the file 'snort.conf' open. The file path is indicated as '/etc/snort'. The code in the file is as follows:

```
531 #####
532
533 # unified2
534 # Recommended for most installs
535 # output unified2: filename merged.log, limit 128, nostamp, mpls_event_types, vlan_event_types
536 output unified2: filename snort.log, limit 128, nostamp, mpls_event_types, vlan_event_types
537
538 # Additional configuration for specific types of installs
539 # output alert_unified2: filename snort.alert, limit 128, nostamp
540 # output log_unified2: filename snort.log, limit 128, nostamp
541
542 # syslog
543 # output alert_syslog: LOG_AUTH LOG_ALERT
544
545 # pcap
546 # output log_tcpdump: tcpdump.log
547
548 # metadata reference data. do not modify these lines
549 include classification.config
550 include reference.config
551
552
553 #####
554 # Step #7: Customize your rule set
555 # For more information, see Snort Manual, Writing Snort Rules
556 #
557 # NOTE: All categories are enabled in this conf file
558 #####
```

Any line (rule) begins with # is ignored (disabled), while the remaining rules are enabled (active).

Conclusion

In conclusion, the Snort lab provided hands-on experience in installing, configuring, and utilizing Snort for network intrusion detection. Through this lab, I gained the knowledge and skills to identify and protect my network, create custom rules, and effectively utilize Snort's detection capabilities. The acquired expertise will play a crucial role in enhancing network security by proactively detecting and responding to potential intrusions and attacks.