WEEK 1

• Introduction to NIDS and Snort

What is NIDS (Network Intrusion Detection System)?

A **Network Intrusion Detection System (NIDS)** is a **security tool** that monitors network traffic in real-time to detect:

- Malicious activity
- Suspicious behavior
- Policy violations
- Attack signatures (e.g., port scans, malware, exploits)

How it works:

- It captures packets from a network interface.
- It analyzes them using rules, heuristics, or machine learning.
- If suspicious activity is detected, it logs or alerts the administrator.

What is Snort?

Snort is one of the most widely used **open-source NIDS tools**, developed by **Martin Roesch** and now maintained by **Cisco**.

Snort can function as:

- A **Packet Sniffer** (like Wireshark)
- A **Packet Logger** (stores network traffic)
- A **Real-time Intrusion Detection System**

Snort Features:

Feature	**Description**
Signature-based Detection	Detects known threats using pre-defined rule sets
Protocol Analysis	Inspects network protocols (TCP, UDP, ICMP, etc.)
Logging & Alerting	Logs suspicious activity or sends alerts
Custom Rule Creation	Users can write their own detection rules
Real-time Traffic Analysis	Works on live traffic from network interfaces

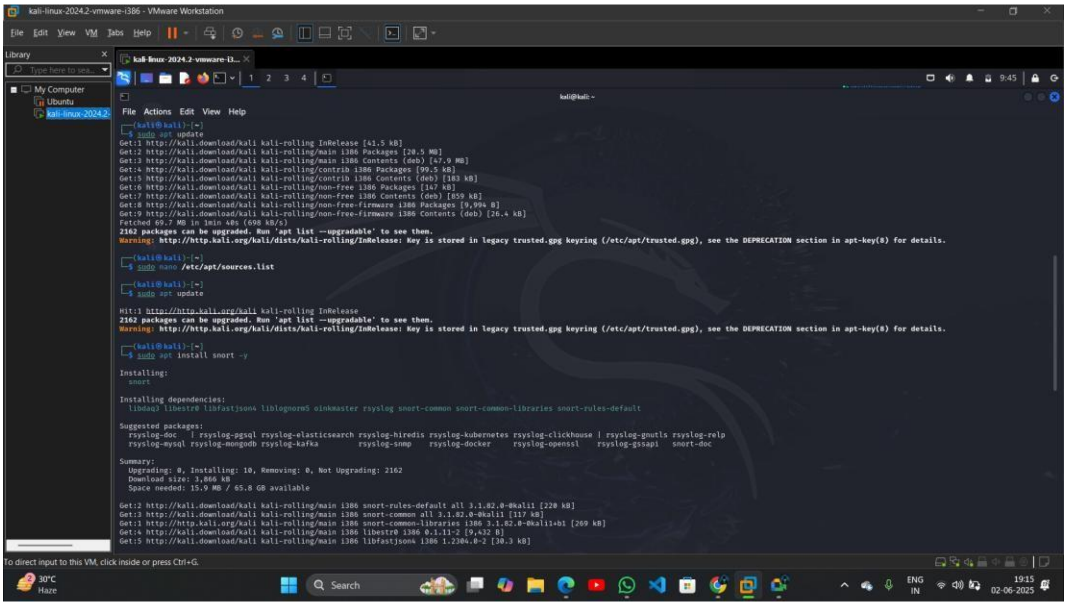
- Install Linux (Ubuntu/Kali)

Done

- Install and verify Snort

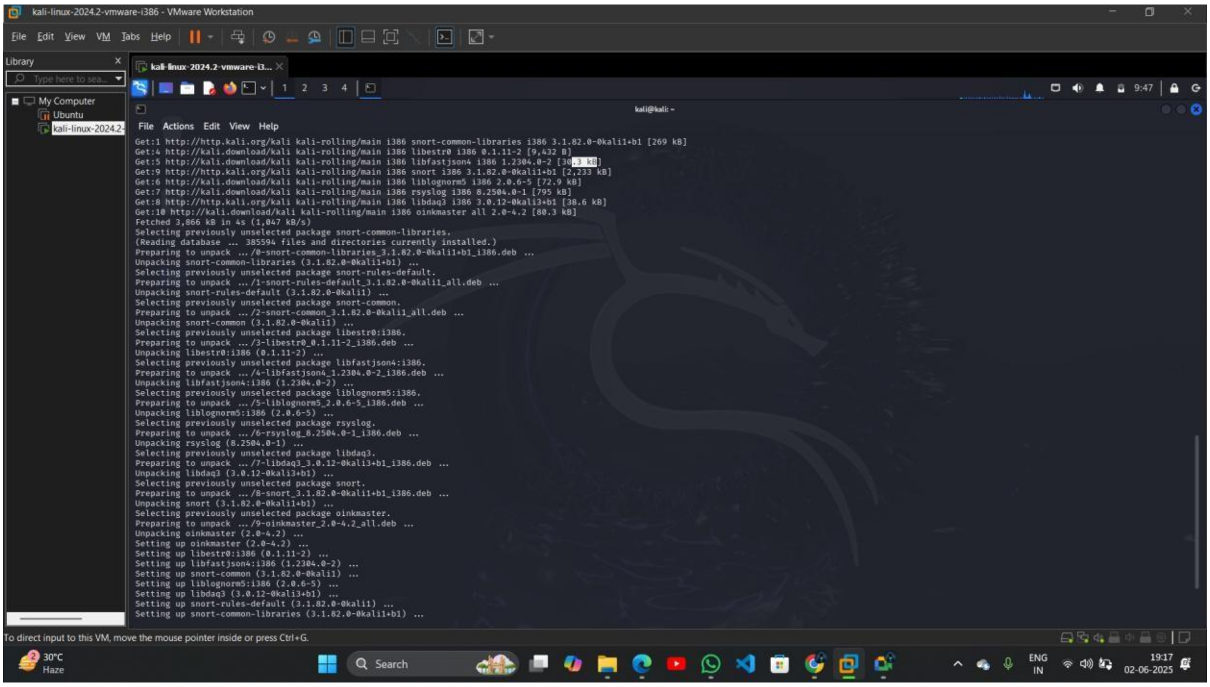
1. sudo apt update

2. sudo apt install snort


kali-linux-2024.2-vmware-i386 - VMware Workstation
File Edit View VM Tabs Help
Library
My Computer
Ubuntu
kali-linux-2024.2
kali@kali: ~
File Actions Edit View Help
30°C
Haze
Search
19:15
02-06-2025

```
┌──(kali㉿kali)-[~]
└─$ sudo apt update
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Get:2 http://kali.download/kali kali-rolling/main i386 Packages [20.5 MB]
Get:3 http://kali.download/kali kali-rolling/main i386 Contents (deb) [47.9 MB]
Get:4 http://kali.download/kali kali-rolling/contrib i386 Packages [99.5 kB]
Get:5 http://kali.download/kali kali-rolling/contrib i386 Contents (deb) [183 kB]
Get:6 http://kali.download/kali kali-rolling/non-free i386 Packages [147 kB]
Get:7 http://kali.download/kali kali-rolling/non-free i386 Contents (deb) [859 kB]
Get:8 http://kali.download/kali kali-rolling/non-free-firmware i386 Packages [9,994 B]
Get:9 http://kali.download/kali kali-rolling/non-free-firmware i386 Contents (deb) [26.4 kB]
Fetched 69.7 MB in 1min 40s (698 kB/s)
2162 packages can be upgraded. Run 'apt list --upgradable' to see them.
Warning: http://http.kali.org/kali/dists/kali-rolling/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.

┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/apt/sources.list

┌──(kali㉿kali)-[~]
└─$ sudo apt update

Hit:1 http://http.kali.org/kali kali-rolling InRelease
2162 packages can be upgraded. Run 'apt list --upgradable' to see them.
Warning: http://http.kali.org/kali/dists/kali-rolling/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.

┌──(kali㉿kali)-[~]
└─$ sudo apt install snort -y

Installing:
  snort

Installing dependencies:
  libdaq3 libestr0 libfastjson4 liblognorm5 oinkmaster rsyslog snort-common snort-common-libraries snort-rules-default

Suggested packages:
  rsyslog-doc   | rsyslog-pgsql rsyslog-elasticsearch rsyslog-hiredis rsyslog-kubernetes rsyslog-clickhouse | rsyslog-gnutls rsyslog-relp
  rsyslog-mysql rsyslog-mongodb rsyslog-kafka         rsyslog-snmp    rsyslog-docker     rsyslog-openssl    rsyslog-gssapi   snort-doc

Summary:
  Upgrading: 0, Installing: 10, Removing: 0, Not Upgrading: 2162
  Download size: 3,866 kB
  Space needed: 15.9 MB / 65.8 GB available

Get:2 http://kali.download/kali kali-rolling/main i386 snort-rules-default all 3.1.82.0-0kali1 [220 kB]
Get:3 http://kali.download/kali kali-rolling/main i386 snort-common all 3.1.82.0-0kali1 [117 kB]
Get:1 http://http.kali.org/kali kali-rolling/main i386 snort-common-libraries i386 3.1.82.0-0kali1+b1 [269 kB]
Get:4 http://kali.download/kali kali-rolling/main i386 libestr0 i386 0.1.11-2 [9,432 B]
Get:5 http://kali.download/kali kali-rolling/main i386 libfastjson4 i386 1.2304.0-2 [30.3 kB]
```

To direct input to this VM, click inside or press Ctrl+G.

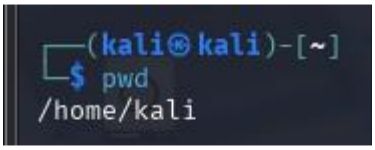

kali-linux-2024.2-vmware-i386 - VMware Workstation
File Edit View VM Tabs Help
Library
My Computer
Ubuntu
kali-linux-2024.2
kali@kali: ~
File Actions Edit View Help
30°C
Haze
Search
19:17
02-06-2025

```
Get:1 http://http.kali.org/kali kali-rolling/main i386 snort-common-libraries i386 3.1.82.0-0kali1+b1 [269 kB]
Get:4 http://kali.download/kali kali-rolling/main i386 libestr0 i386 0.1.11-2 [9,432 B]
Get:5 http://kali.download/kali kali-rolling/main i386 libfastjson4 i386 1.2304.0-2 [30.3 kB]
Get:9 http://http.kali.org/kali kali-rolling/main i386 snort i386 3.1.82.0-0kali1+b1 [2,233 kB]
Get:6 http://kali.download/kali kali-rolling/main i386 liblognorm5 i386 2.0.6-5 [72.9 kB]
Get:7 http://kali.download/kali kali-rolling/main i386 rsyslog i386 8.2504.0-1 [795 kB]
Get:8 http://http.kali.org/kali kali-rolling/main i386 libdaq3 i386 3.0.12-0kali3+b1 [38.6 kB]
Get:10 http://kali.download/kali kali-rolling/main i386 oinkmaster all 2.0-4.2 [80.3 kB]
Fetched 3,866 kB in 4s (1,047 kB/s)
Selecting previously unselected package snort-common-libraries.
(Reading database ... 385594 files and directories currently installed.)
Preparing to unpack .../0-snort-common-libraries_3.1.82.0-0kali1+b1_i386.deb ...
Unpacking snort-common-libraries (3.1.82.0-0kali1+b1) ...
Selecting previously unselected package snort-rules-default.
Preparing to unpack .../1-snort-rules-default_3.1.82.0-0kali1_all.deb ...
Unpacking snort-rules-default (3.1.82.0-0kali1) ...
Selecting previously unselected package snort-common.
Preparing to unpack .../2-snort-common_3.1.82.0-0kali1_all.deb ...
Unpacking snort-common (3.1.82.0-0kali1) ...
Selecting previously unselected package libestr0:i386.
Preparing to unpack .../3-libestr0_0.1.11-2_i386.deb ...
Unpacking libestr0:i386 (0.1.11-2) ...
Selecting previously unselected package libfastjson4:i386.
Preparing to unpack .../4-libfastjson4_1.2304.0-2_i386.deb ...
Unpacking libfastjson4:i386 (1.2304.0-2) ...
Selecting previously unselected package liblognorm5:i386.
Preparing to unpack .../5-liblognorm5_2.0.6-5_i386.deb ...
Unpacking liblognorm5:i386 (2.0.6-5) ...
Selecting previously unselected package rsyslog.
Preparing to unpack .../6-rsyslog_8.2504.0-1_i386.deb ...
Unpacking rsyslog (8.2504.0-1) ...
Selecting previously unselected package libdaq3.
Preparing to unpack .../7-libdaq3_3.0.12-0kali3+b1_i386.deb ...
Unpacking libdaq3 (3.0.12-0kali3+b1) ...
Selecting previously unselected package snort.
Preparing to unpack .../8-snort_3.1.82.0-0kali1+b1_i386.deb ...
Unpacking snort (3.1.82.0-0kali1+b1) ...
Selecting previously unselected package oinkmaster.
Preparing to unpack .../9-oinkmaster_2.0-4.2_all.deb ...
Unpacking oinkmaster (2.0-4.2) ...
Setting up oinkmaster (2.0-4.2) ...
Setting up libestr0:i386 (0.1.11-2) ...
Setting up libfastjson4:i386 (1.2304.0-2) ...
Setting up snort-common (3.1.82.0-0kali1) ...
Setting up liblognorm5:i386 (2.0.6-5) ...
Setting up libdaq3 (3.0.12-0kali3+b1) ...
Setting up snort-rules-default (3.1.82.0-0kali1) ...
Setting up snort-common-libraries (3.1.82.0-0kali1+b1) ...
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

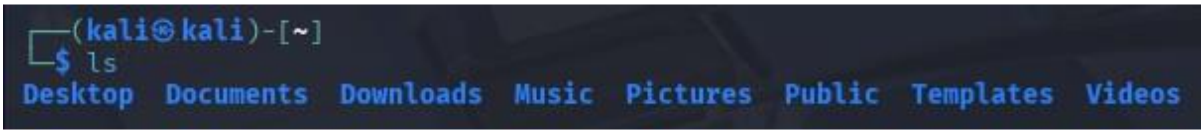
• Basic Linux command-line navigation

1. pwd: Show current working directory



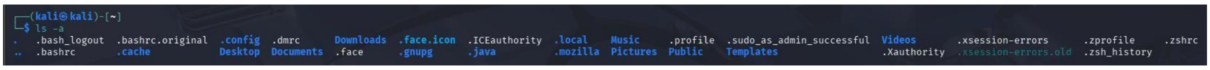
```
┌──(kali㉿kali)-[~]
└─$ pwd
/home/kali
```

2. ls: List files in current folder



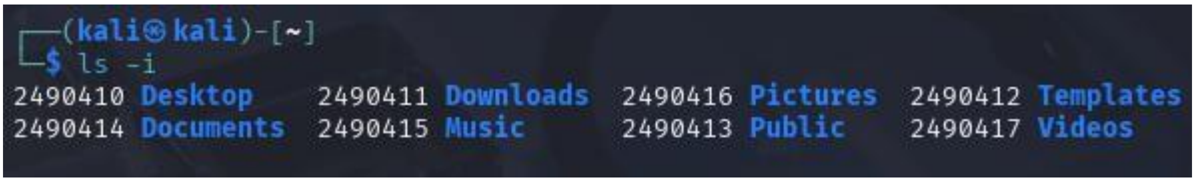
```
┌──(kali㉿kali)-[~]
└─$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
```

3. ls -a: Show hidden files



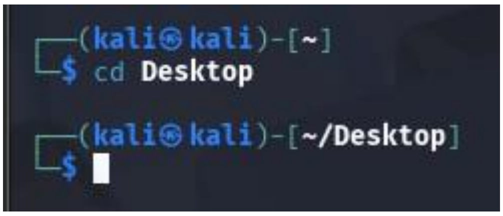
```
┌──(kali㉿kali)-[~]
└─$ ls -a
.   .bash_logout  .bashrc.original  .config  .dmrc      Downloads  .face.icon  .ICEauthority  .local    Music     .profile  .sudo_as_admin_successful  Videos       .xsession-errors      .zprofile     .zshrc
..  .bashrc       .cache            Desktop  Documents  .face      .gnupg      .java          .mozilla  Pictures  Public    Templates                  .Xauthority  .xsession-errors.old  .zsh_history
```

4. ls -l: Show detailed file list



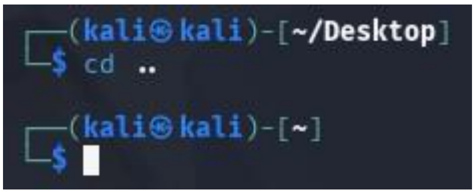
```
┌──(kali㉿kali)-[~]
└─$ ls -i
2490410 Desktop    2490411 Downloads  2490416 Pictures  2490412 Templates
2490414 Documents  2490415 Music      2490413 Public    2490417 Videos
```

5. cd folder/: Change directory



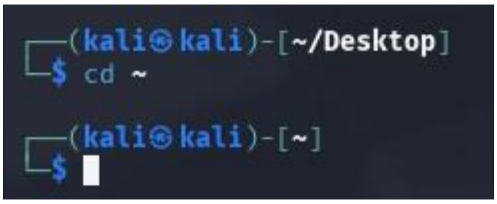
```
┌──(kali㉿kali)-[~]
└─$ cd Desktop

┌──(kali㉿kali)-[~/Desktop]
└─$ █
```

6. cd .. : Go up one directory



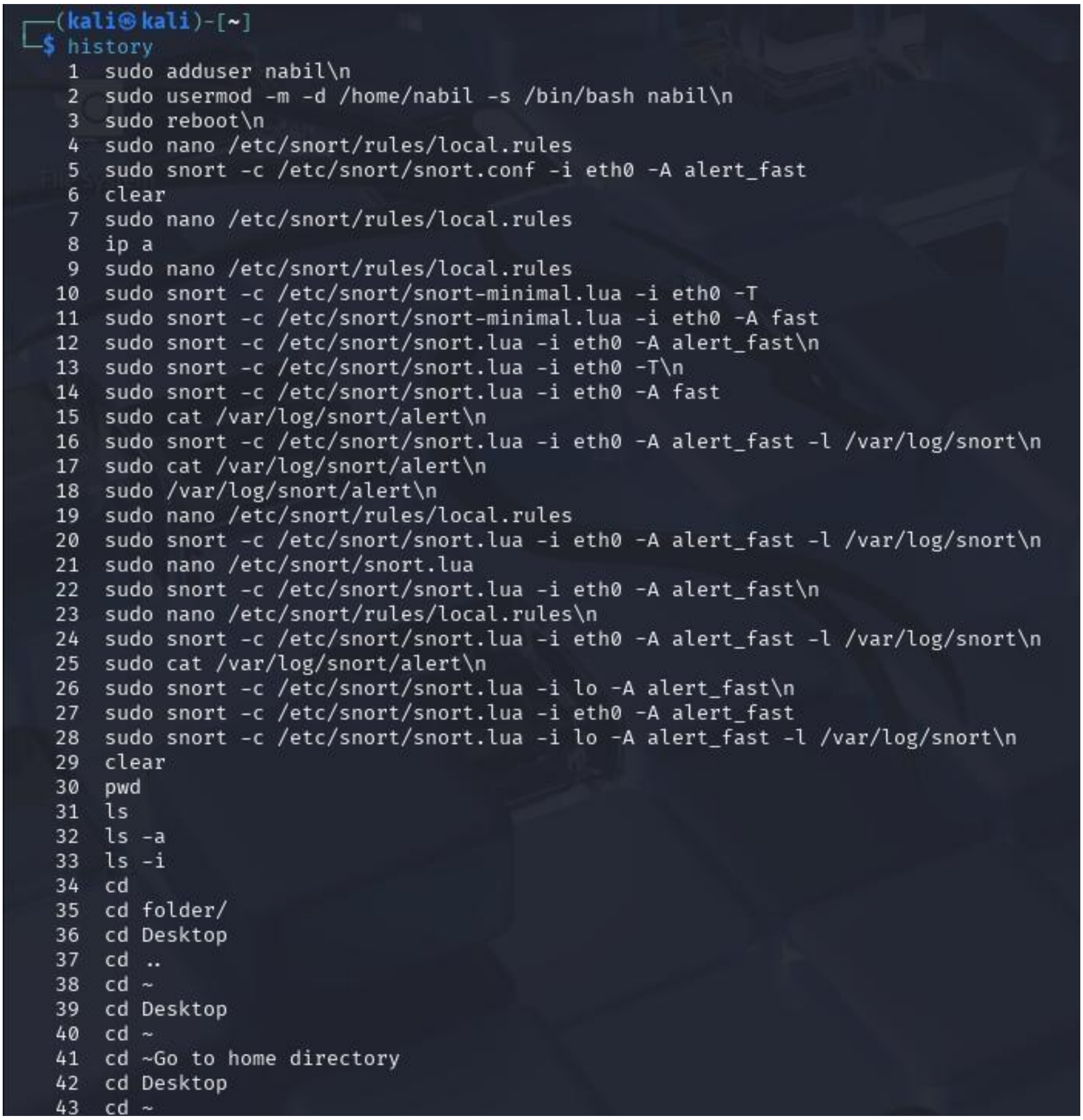
```
┌──(kali㉿kali)-[~/Desktop]
└─$ cd ..

┌──(kali㉿kali)-[~]
└─$ █
```

7. cd ~ : Go to home directory



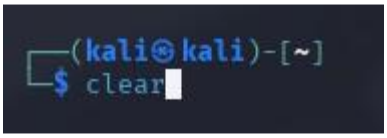
```
┌──(kali㉿kali)-[~/Desktop]
└─$ cd ~

┌──(kali㉿kali)-[~]
└─$ █
```

8. history: Show command history

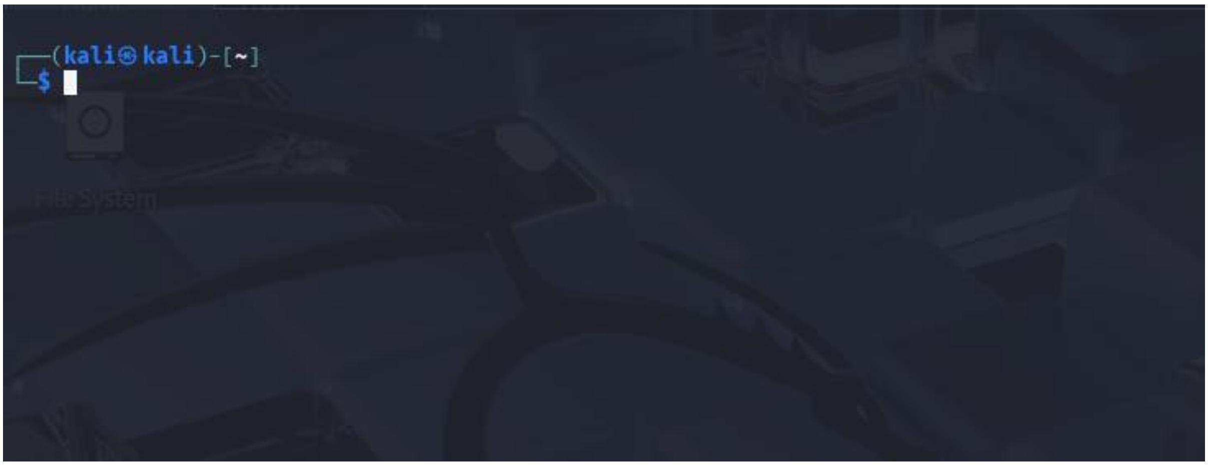


```
┌──(kali㉿kali)-[~]
└─$ history
    1  sudo adduser nabil\n
    2  sudo usermod -m -d /home/nabil -s /bin/bash nabil\n
    3  sudo reboot\n
    4  sudo nano /etc/snort/rules/local.rules
    5  sudo snort -c /etc/snort/snort.conf -i eth0 -A alert_fast
    6  clear
    7  sudo nano /etc/snort/rules/local.rules
    8  ip a
    9  sudo nano /etc/snort/rules/local.rules
   10  sudo snort -c /etc/snort/snort-minimal.lua -i eth0 -T
   11  sudo snort -c /etc/snort/snort-minimal.lua -i eth0 -A fast
   12  sudo snort -c /etc/snort/snort.lua -i eth0 -A alert_fast\n
   13  sudo snort -c /etc/snort/snort.lua -i eth0 -T\n
   14  sudo snort -c /etc/snort/snort.lua -i eth0 -A fast
   15  sudo cat /var/log/snort/alert\n
   16  sudo snort -c /etc/snort/snort.lua -i eth0 -A alert_fast -l /var/log/snort\n
   17  sudo cat /var/log/snort/alert\n
   18  sudo /var/log/snort/alert\n
   19  sudo nano /etc/snort/rules/local.rules
   20  sudo snort -c /etc/snort/snort.lua -i eth0 -A alert_fast -l /var/log/snort\n
   21  sudo nano /etc/snort/snort.lua
   22  sudo snort -c /etc/snort/snort.lua -i eth0 -A alert_fast\n
   23  sudo nano /etc/snort/rules/local.rules\n
   24  sudo snort -c /etc/snort/snort.lua -i eth0 -A alert_fast -l /var/log/snort\n
   25  sudo cat /var/log/snort/alert\n
   26  sudo snort -c /etc/snort/snort.lua -i lo -A alert_fast\n
   27  sudo snort -c /etc/snort/snort.lua -i eth0 -A alert_fast
   28  sudo snort -c /etc/snort/snort.lua -i lo -A alert_fast -l /var/log/snort\n
   29  clear
   30  pwd
   31  ls
   32  ls -a
   33  ls -i
   34  cd
   35  cd folder/
   36  cd Desktop
   37  cd ..
   38  cd ~
   39  cd Desktop
   40  cd ~
   41  cd ~Go to home directory
   42  cd Desktop
   43  cd ~
```

9. clear: Clear terminal screen

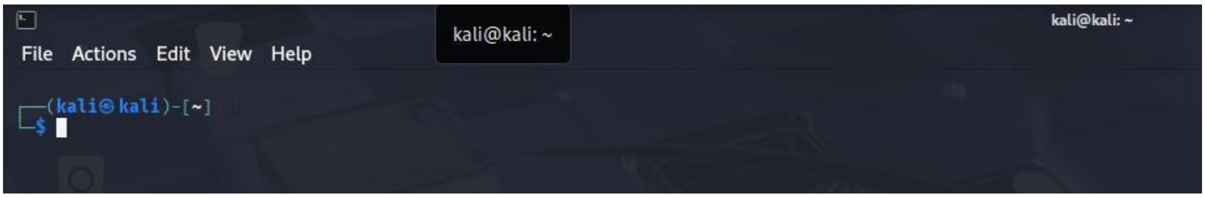


```
┌──(kali㉿kali)-[~]
└─$ clear
```



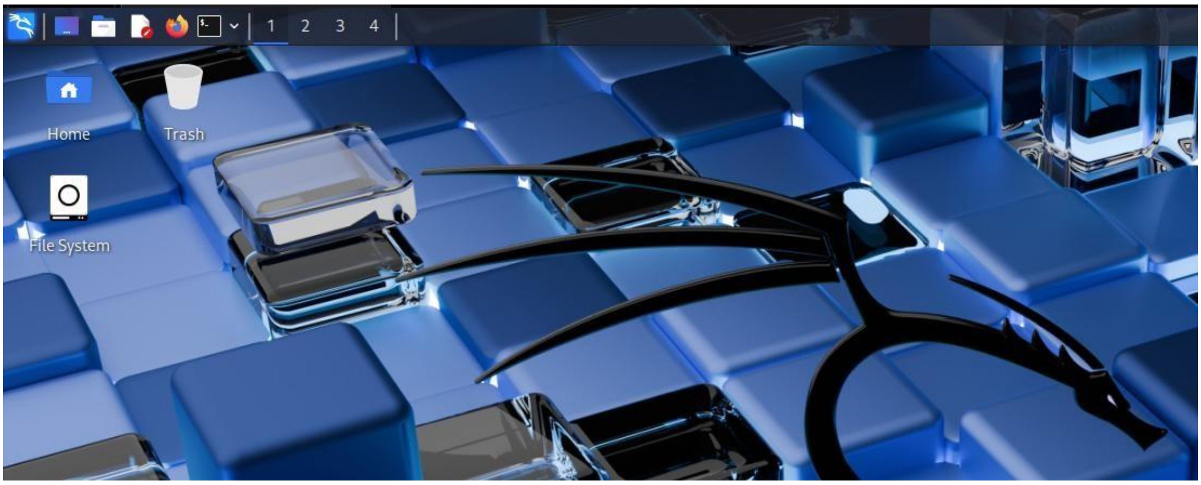
```
┌──(kali㉿kali)-[~]
└─$ 
```

10. exit: Exit terminal session



```
┌──(kali㉿kali)-[~]
└─$ 
```

After enter the command ‘exit’ the terminal shutdown and we see desktop interface



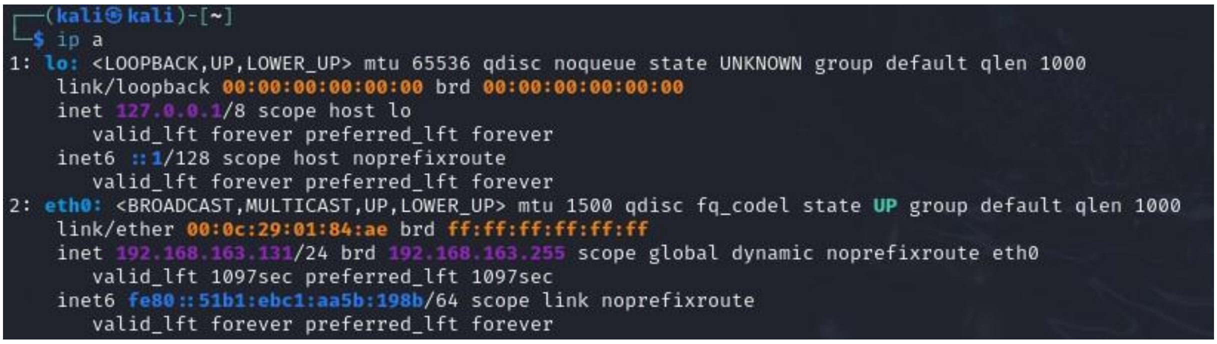
WEEK 2

- Identify active network interface

Commands:

Method 1: Using ip a (recommended)

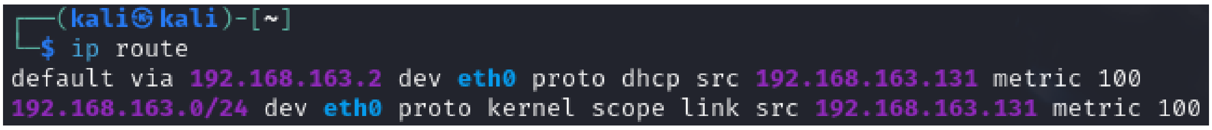
ip a



```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:01:84:ae brd ff:ff:ff:ff:ff:ff
    inet 192.168.163.131/24 brd 192.168.163.255 scope global dynamic noprefixroute eth0
       valid_lft 1097sec preferred_lft 1097sec
    inet6 fe80::51b1:ebc1:aa5b:198b/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Method 2: Using ip route

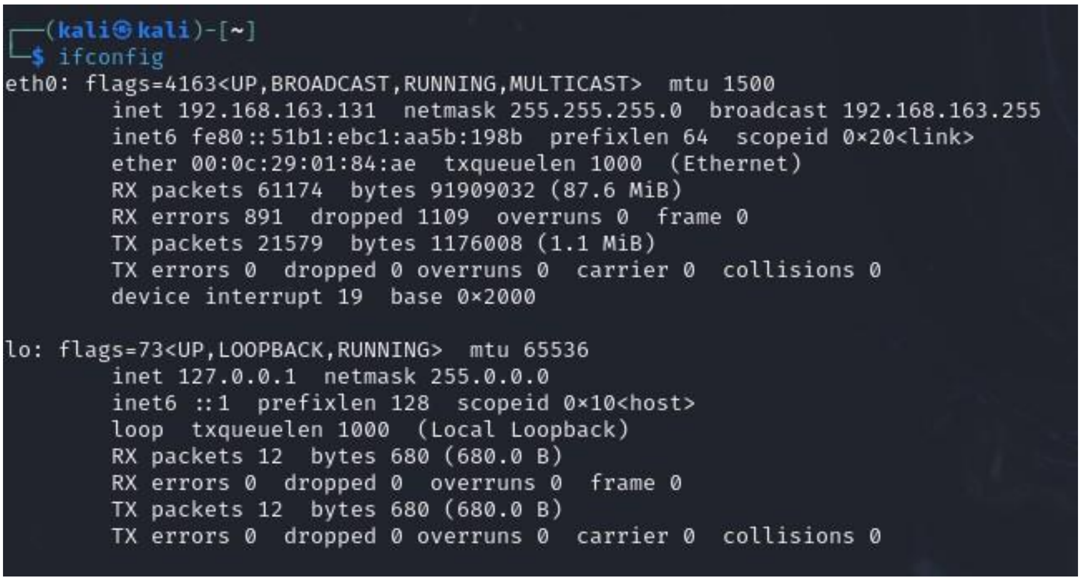
ip route



```
┌──(kali㉿kali)-[~]
└─$ ip route
default via 192.168.163.2 dev eth0 proto dhcp src 192.168.163.131 metric 100
192.168.163.0/24 dev eth0 proto kernel scope link src 192.168.163.131 metric 100
```

Method 3: Using ifconfig (legacy)

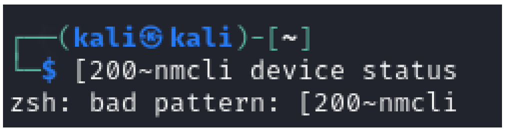
Ifconfig



```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.163.131  netmask 255.255.255.0  broadcast 192.168.163.255
        inet6 fe80::51b1:ebc1:aa5b:198b  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:01:84:ae  txqueuelen 1000  (Ethernet)
        RX packets 61174  bytes 91909032 (87.6 MiB)
        RX errors 891  dropped 1109  overruns 0  frame 0
        TX packets 21579  bytes 1176008 (1.1 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 19  base 0x2000

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 680 (680.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12  bytes 680 (680.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Method 4: With nmcli (if using NetworkManager)

nmcli device status



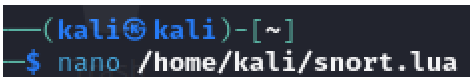
```
┌──(kali㉿kali)-[~]
└─$ [200~nmcli device status
zsh: bad pattern: [200~nmcli
```

• Configure Snort with monitored IP range

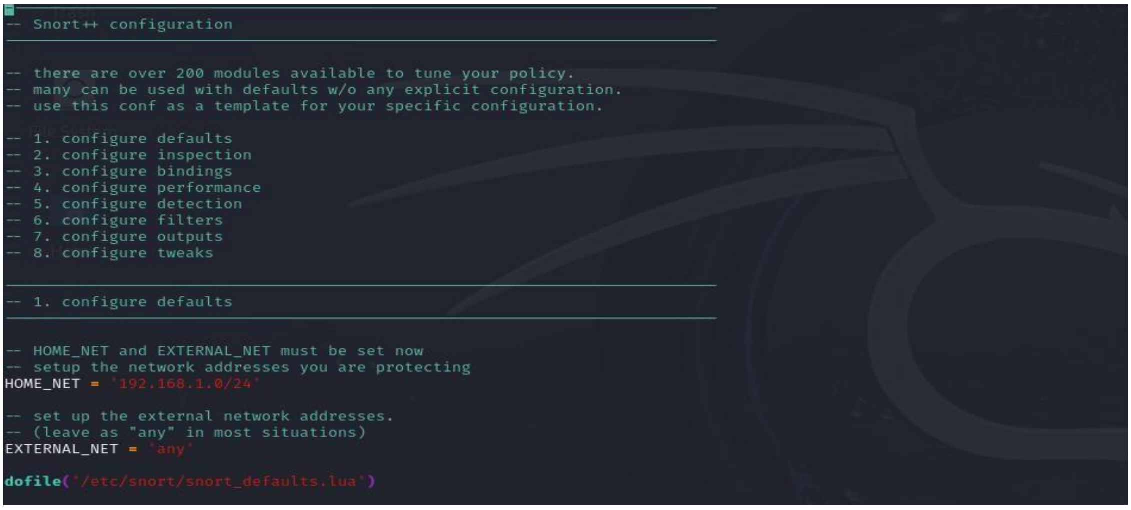
1. Define your monitored IP range in your Lua config

Open your Snort config file (/home/kali/snort.lua):

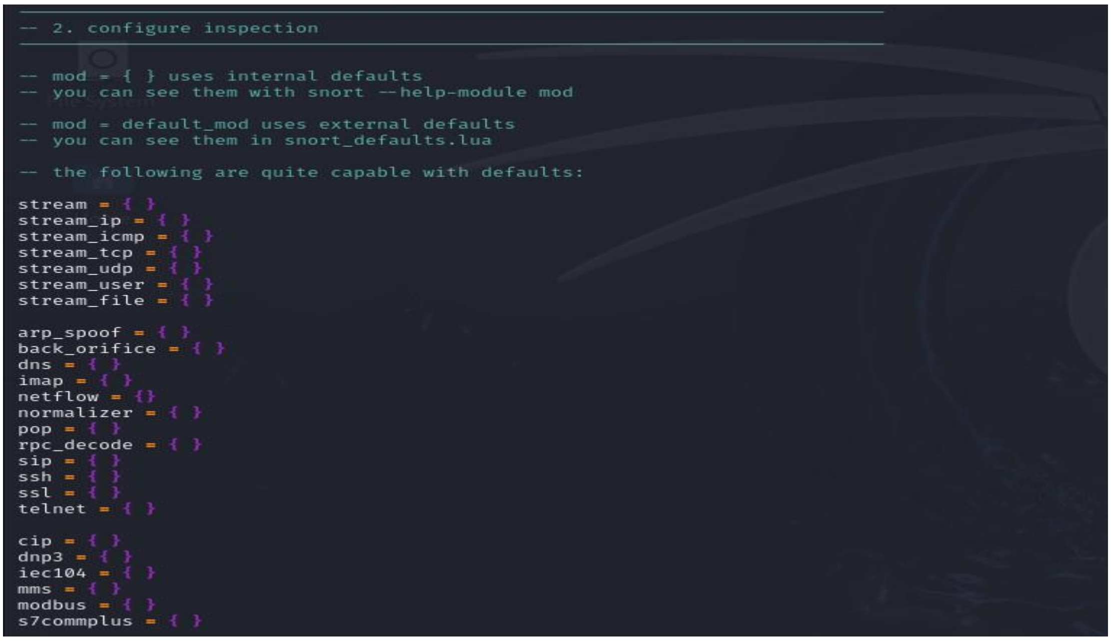
nano /home/kali/snort.lua



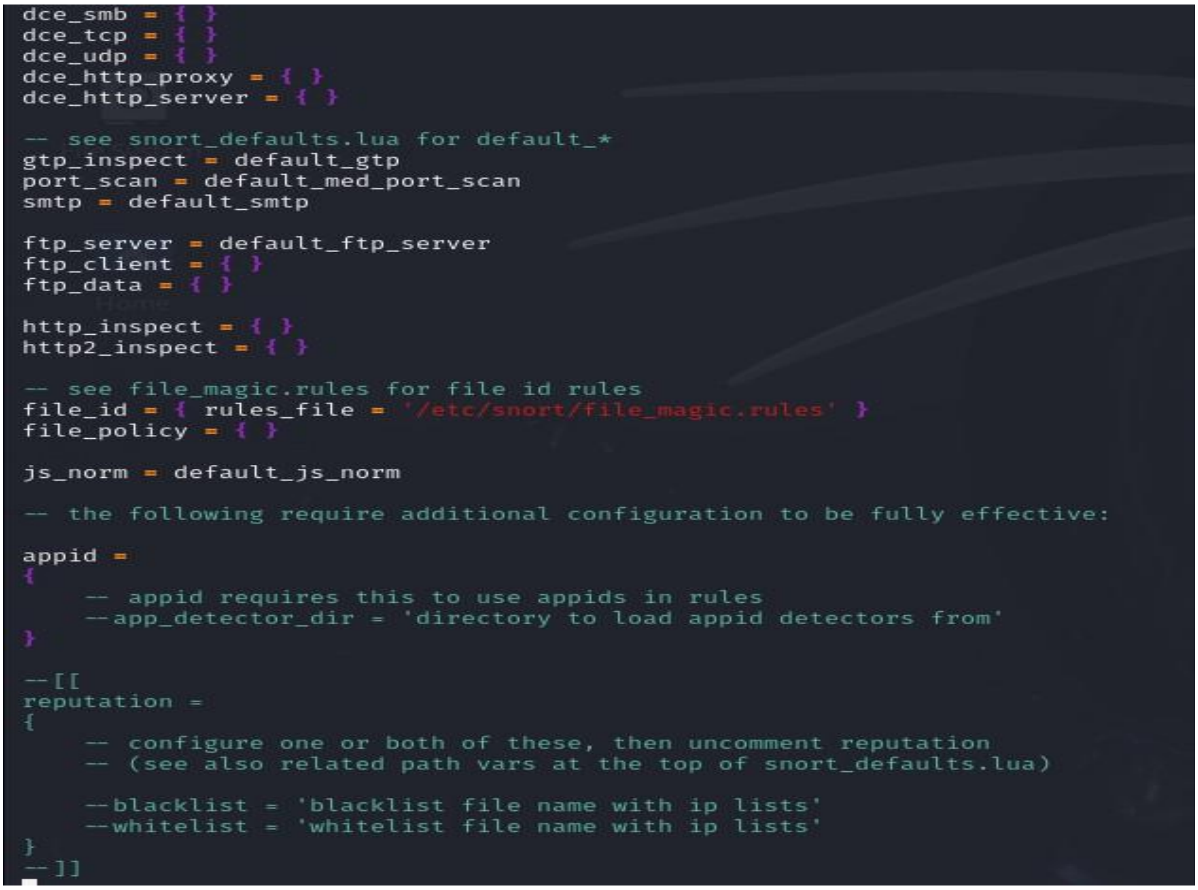
```
──(kali㉿kali)-[~]
─$ nano /home/kali/snort.lua
```



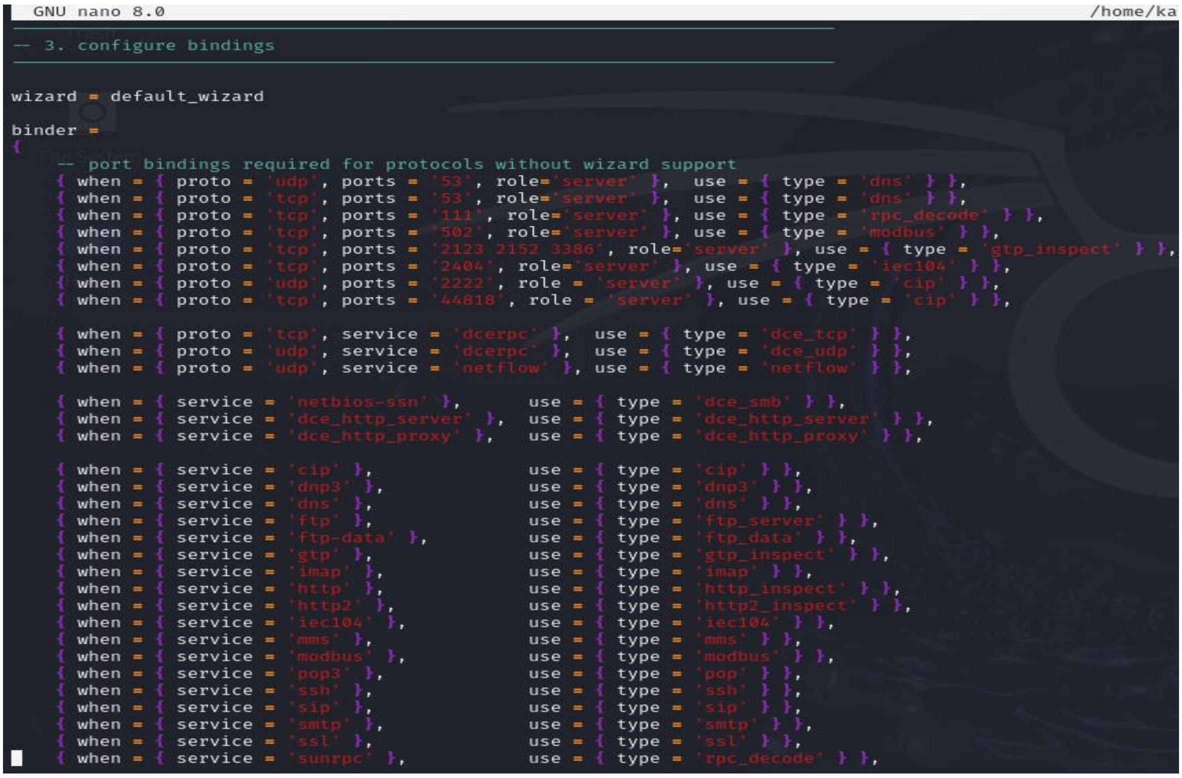
```
---------------------------------------------------------------------------
-- Snort++ configuration
---------------------------------------------------------------------------

-- there are over 200 modules available to tune your policy.
-- many can be used with defaults w/o any explicit configuration.
-- use this conf as a template for your specific configuration.

-- 1. configure defaults
-- 2. configure inspection
-- 3. configure bindings
-- 4. configure performance
-- 5. configure detection
-- 6. configure filters
-- 7. configure outputs
-- 8. configure tweaks

---------------------------------------------------------------------------
-- 1. configure defaults
---------------------------------------------------------------------------

-- HOME_NET and EXTERNAL_NET must be set now
-- setup the network addresses you are protecting
HOME_NET = '192.168.1.0/24'

-- set up the external network addresses.
-- (leave as "any" in most situations)
EXTERNAL_NET = 'any'

dofile('/etc/snort/snort_defaults.lua')
```



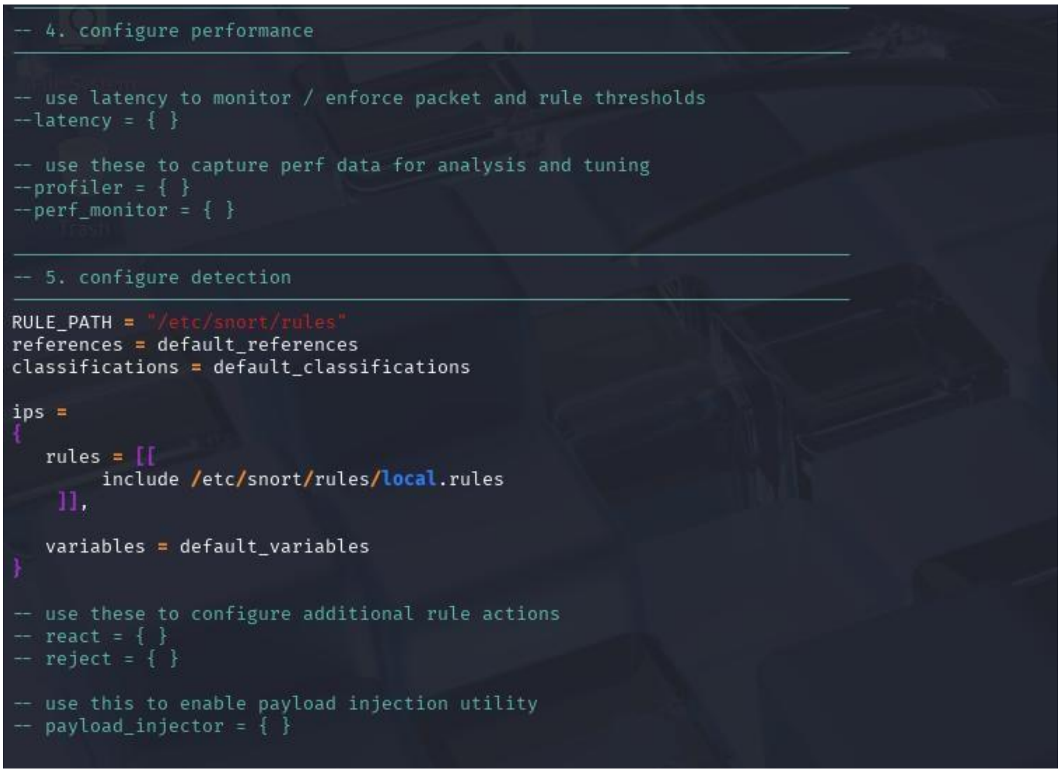
```
---------------------------------------------------------------------------
-- 2. configure inspection
---------------------------------------------------------------------------

-- mod = { } uses internal defaults
-- you can see them with snort --help-module mod

-- mod = default_mod uses external defaults
-- you can see them in snort_defaults.lua

-- the following are quite capable with defaults:

stream = { }
stream_ip = { }
stream_icmp = { }
stream_tcp = { }
stream_udp = { }
stream_user = { }
stream_file = { }

arp_spoof = { }
back_orifice = { }
dns = { }
imap = { }
netflow = {}
normalizer = { }
pop = { }
rpc_decode = { }
sip = { }
ssh = { }
ssl = { }
telnet = { }

cip = { }
dnp3 = { }
iec104 = { }
mms = { }
modbus = { }
s7commplus = { }
```



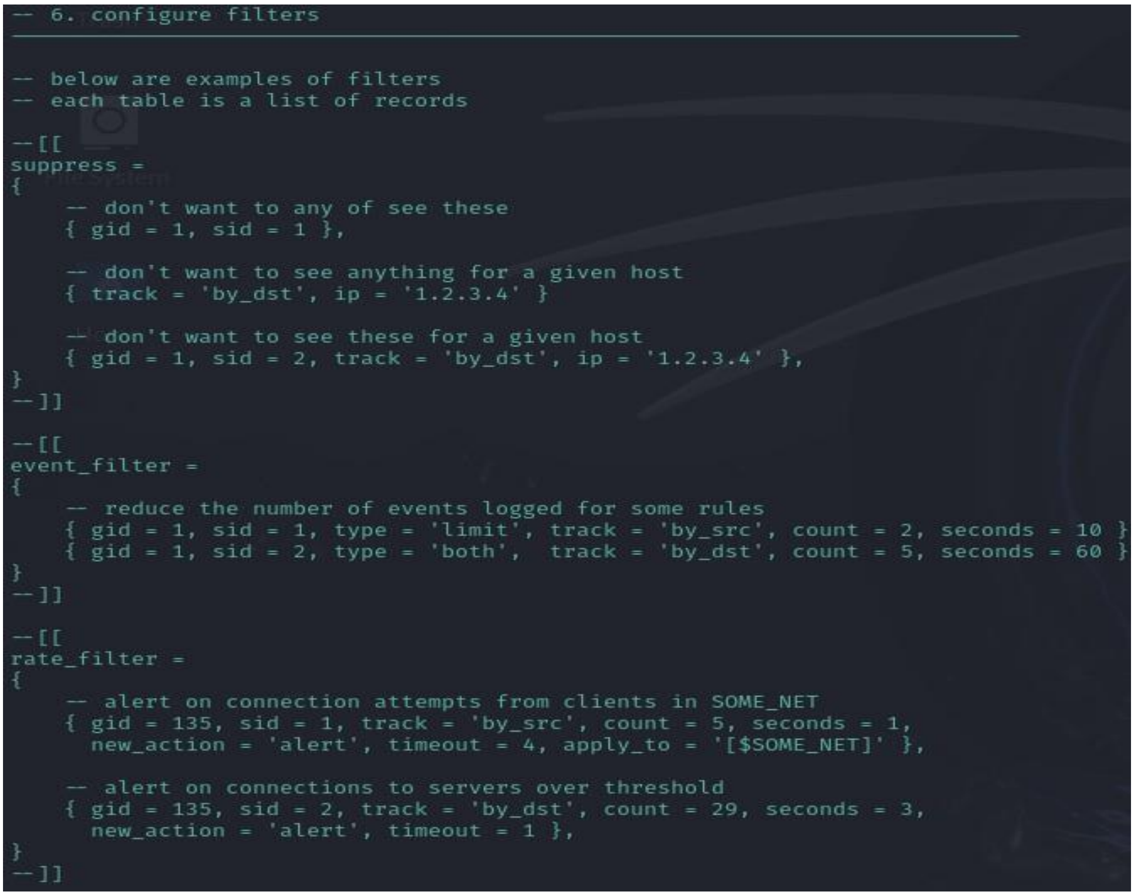
```
dce_smb = { }
dce_tcp = { }
dce_udp = { }
dce_http_proxy = { }
dce_http_server = { }

-- see snort_defaults.lua for default_*
gtp_inspect = default_gtp
port_scan = default_med_port_scan
smtp = default_smtp

ftp_server = default_ftp_server
ftp_client = { }
ftp_data = { }

http_inspect = { }
http2_inspect = { }

-- see file_magic.rules for file id rules
file_id = { rules_file = '/etc/snort/file_magic.rules' }
file_policy = { }

js_norm = default_js_norm

-- the following require additional configuration to be fully effective:

appid =
{
    -- appid requires this to use appids in rules
    --app_detector_dir = 'directory to load appid detectors from'
}

--[[
reputation =
{
    -- configure one or both of these, then uncomment reputation
    -- (see also related path vars at the top of snort_defaults.lua)

    --blacklist = 'blacklist file name with ip lists'
    --whitelist = 'whitelist file name with ip lists'
}
--]]
```



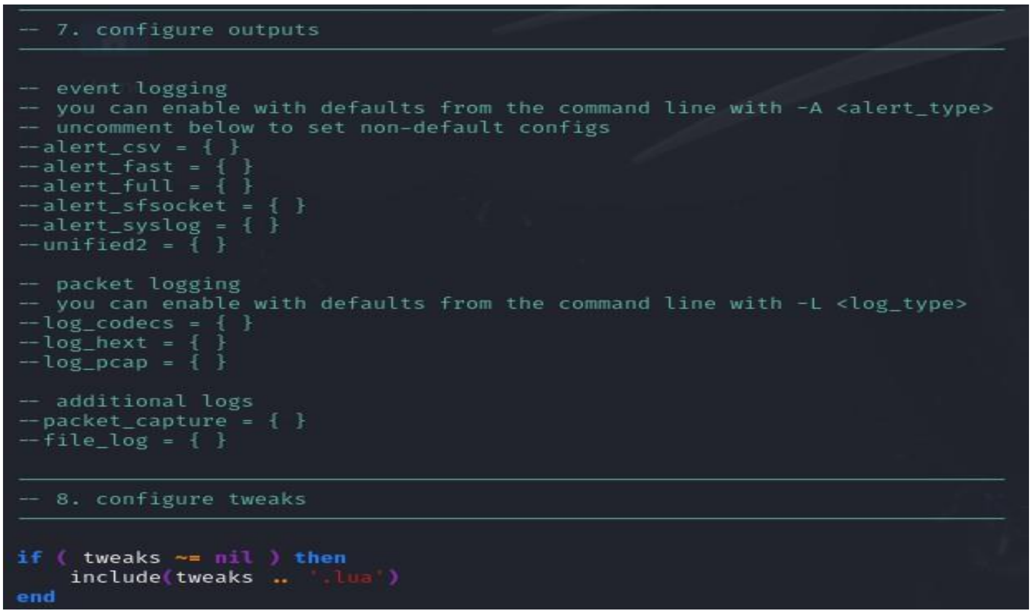
```
  GNU nano 8.0                                                                    /home/ka
--------------------------------------------------------------------------------
-- 3. configure bindings
--------------------------------------------------------------------------------

wizard = default_wizard

binder =
{
    -- port bindings required for protocols without wizard support
    { when = { proto = 'udp', ports = '53', role='server' },  use = { type = 'dns' } },
    { when = { proto = 'tcp', ports = '53', role='server' },  use = { type = 'dns' } },
    { when = { proto = 'tcp', ports = '111', role='server' }, use = { type = 'rpc_decode' } },
    { when = { proto = 'tcp', ports = '502', role='server' }, use = { type = 'modbus' } },
    { when = { proto = 'tcp', ports = '2123 2152 3386', role='server' }, use = { type = 'gtp_inspect' } },
    { when = { proto = 'tcp', ports = '2404', role='server' }, use = { type = 'iec104' } },
    { when = { proto = 'udp', ports = '2222', role = 'server' }, use = { type = 'cip' } },
    { when = { proto = 'tcp', ports = '44818', role = 'server' }, use = { type = 'cip' } },

    { when = { proto = 'tcp', service = 'dcerpc' },  use = { type = 'dce_tcp' } },
    { when = { proto = 'udp', service = 'dcerpc' },  use = { type = 'dce_udp' } },
    { when = { proto = 'udp', service = 'netflow' }, use = { type = 'netflow' } },

    { when = { service = 'netbios-ssn' },      use = { type = 'dce_smb' } },
    { when = { service = 'dce_http_server' }, use = { type = 'dce_http_server' } },
    { when = { service = 'dce_http_proxy' },  use = { type = 'dce_http_proxy' } },

    { when = { service = 'cip' },             use = { type = 'cip' } },
    { when = { service = 'dnp3' },            use = { type = 'dnp3' } },
    { when = { service = 'dns' },             use = { type = 'dns' } },
    { when = { service = 'ftp' },             use = { type = 'ftp_server' } },
    { when = { service = 'ftp-data' },        use = { type = 'ftp_data' } },
    { when = { service = 'gtp' },             use = { type = 'gtp_inspect' } },
    { when = { service = 'imap' },            use = { type = 'imap' } },
    { when = { service = 'http' },            use = { type = 'http_inspect' } },
    { when = { service = 'http2' },           use = { type = 'http2_inspect' } },
    { when = { service = 'iec104' },          use = { type = 'iec104' } },
    { when = { service = 'mms' },             use = { type = 'mms' } },
    { when = { service = 'modbus' },          use = { type = 'modbus' } },
    { when = { service = 'pop3' },            use = { type = 'pop' } },
    { when = { service = 'ssh' },             use = { type = 'ssh' } },
    { when = { service = 'sip' },             use = { type = 'sip' } },
    { when = { service = 'smtp' },            use = { type = 'smtp' } },
    { when = { service = 'ssl' },             use = { type = 'ssl' } },
    { when = { service = 'sunrpc' },          use = { type = 'rpc_decode' } },
```



```
---------------------------------------------------------------------------
-- 4. configure performance
---------------------------------------------------------------------------

-- use latency to monitor / enforce packet and rule thresholds
--latency = { }

-- use these to capture perf data for analysis and tuning
--profiler = { }
--perf_monitor = { }

---------------------------------------------------------------------------
-- 5. configure detection
---------------------------------------------------------------------------
RULE_PATH = "/etc/snort/rules"
references = default_references
classifications = default_classifications

ips =
{
   rules = [[
        include /etc/snort/rules/local.rules
    ]],

   variables = default_variables
}

-- use these to configure additional rule actions
-- react = { }
-- reject = { }

-- use this to enable payload injection utility
-- payload_injector = { }
```



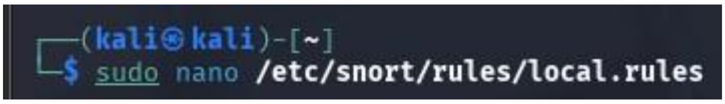
```
-- 6. configure filters
---------------------------------------------------------------------------

-- below are examples of filters
-- each table is a list of records

--[[
suppress =
{
    -- don't want to any of see these
    { gid = 1, sid = 1 },

    -- don't want to see anything for a given host
    { track = 'by_dst', ip = '1.2.3.4' }

    -- don't want to see these for a given host
    { gid = 1, sid = 2, track = 'by_dst', ip = '1.2.3.4' },
}
--]]

--[[
event_filter =
{
    -- reduce the number of events logged for some rules
    { gid = 1, sid = 1, type = 'limit', track = 'by_src', count = 2, seconds = 10 }
    { gid = 1, sid = 2, type = 'both',  track = 'by_dst', count = 5, seconds = 60 }
}
--]]

--[[
rate_filter =
{
    -- alert on connection attempts from clients in SOME_NET
    { gid = 135, sid = 1, track = 'by_src', count = 5, seconds = 1,
      new_action = 'alert', timeout = 4, apply_to = '[$SOME_NET]' },

    -- alert on connections to servers over threshold
    { gid = 135, sid = 2, track = 'by_dst', count = 29, seconds = 3,
      new_action = 'alert', timeout = 1 },
}
--]]
```



```
---------------------------------------------------------------------------
-- 7. configure outputs
---------------------------------------------------------------------------

-- event logging
-- you can enable with defaults from the command line with -A <alert_type>
-- uncomment below to set non-default configs
--alert_csv = { }
--alert_fast = { }
--alert_full = { }
--alert_sfsocket = { }
--alert_syslog = { }
--unified2 = { }

-- packet logging
-- you can enable with defaults from the command line with -L <log_type>
--log_codecs = { }
--log_hext = { }
--log_pcap = { }

-- additional logs
--packet_capture = { }
--file_log = { }

---------------------------------------------------------------------------
-- 8. configure tweaks
---------------------------------------------------------------------------

if ( tweaks ~= nil ) then
    include(tweaks .. '.lua')
end
```

Edit local.rules file:

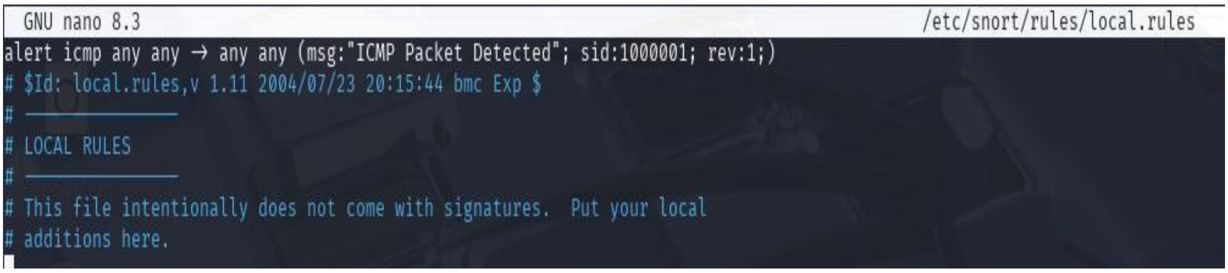
sudo nano /etc/snort/rules/local.rules



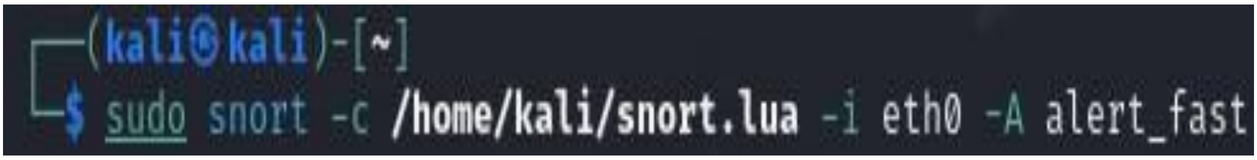
```
┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/snort/rules/local.rules
```

Add this rule to detect all ICMP packets:

alert icmp any any -> any any (msg:"ICMP Packet Detected"; sid:1000001; rev:1;)

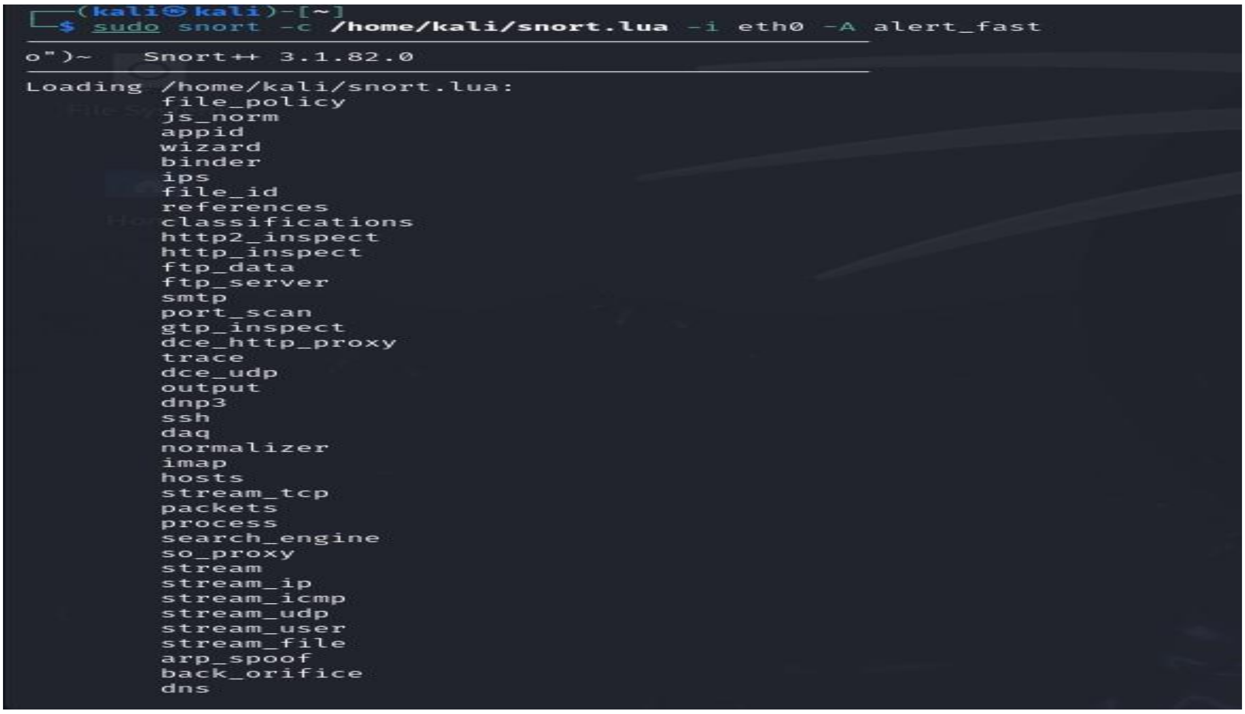


```
  GNU nano 8.3                                              /etc/snort/rules/local.rules
alert icmp any any → any any (msg:"ICMP Packet Detected"; sid:1000001; rev:1;)
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# ----------------
# LOCAL RULES
# ----------------
# This file intentionally does not come with signatures.  Put your local
# additions here.
```

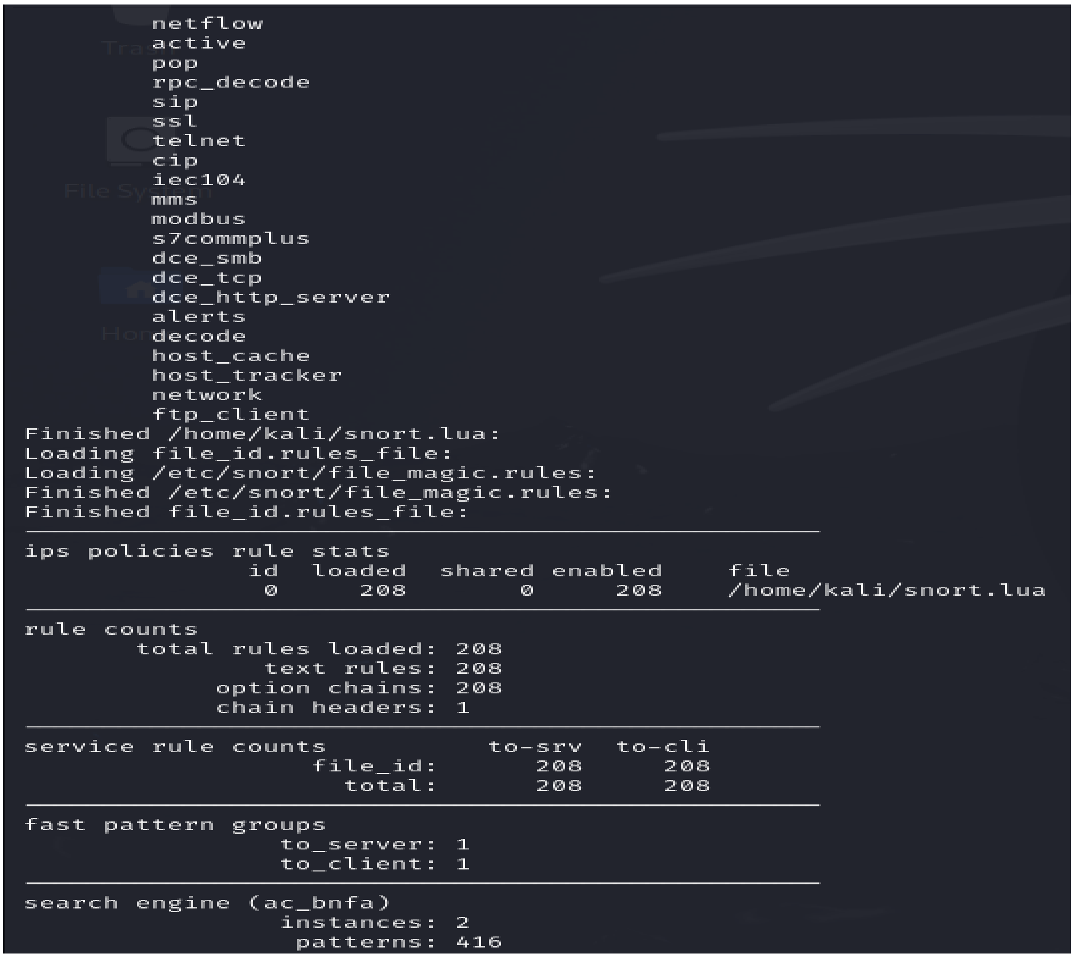


```
┌──(kali㉿kali)-[~]
└─$ sudo snort -c /home/kali/snort.lua -i eth0 -A alert_fast
```

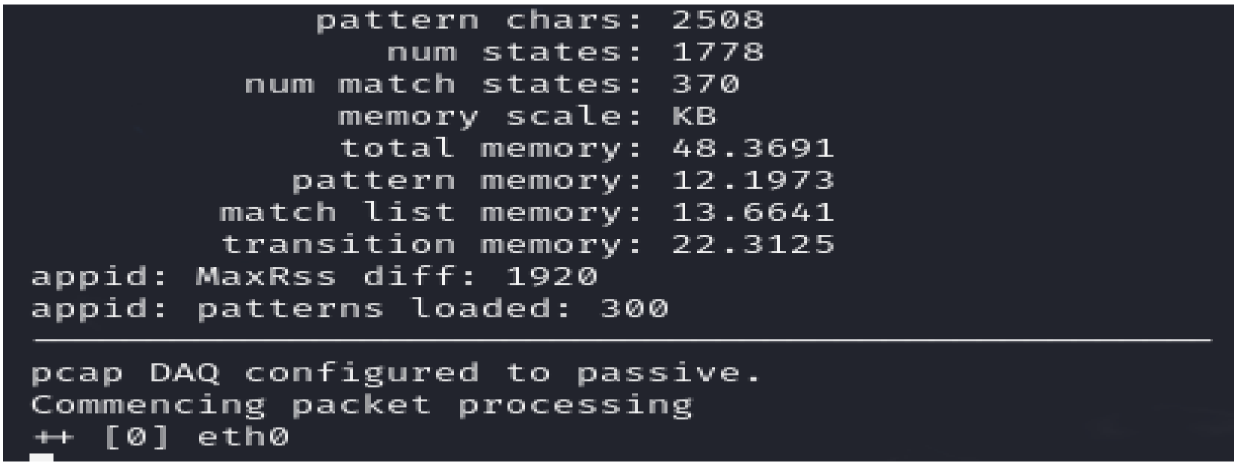
sudo snort -c /etc/snort/snort.lua -i eth0 -A alert_fast



```
┌──(kali㉿kali)-[~]
└─$ sudo snort -c /home/kali/snort.lua -i eth0 -A alert_fast
--------------------------------------------------
o")~   Snort++ 3.1.82.0
--------------------------------------------------
Loading /home/kali/snort.lua:
        file_policy
        js_norm
        appid
        wizard
        binder
        ips
        file_id
        references
        classifications
        http2_inspect
        http_inspect
        ftp_data
        ftp_server
        smtp
        port_scan
        gtp_inspect
        dce_http_proxy
        trace
        dce_udp
        output
        dnp3
        ssh
        daq
        normalizer
        imap
        hosts
        stream_tcp
        packets
        process
        search_engine
        so_proxy
        stream
        stream_ip
        stream_icmp
        stream_udp
        stream_user
        stream_file
        arp_spoof
        back_orifice
        dns
```



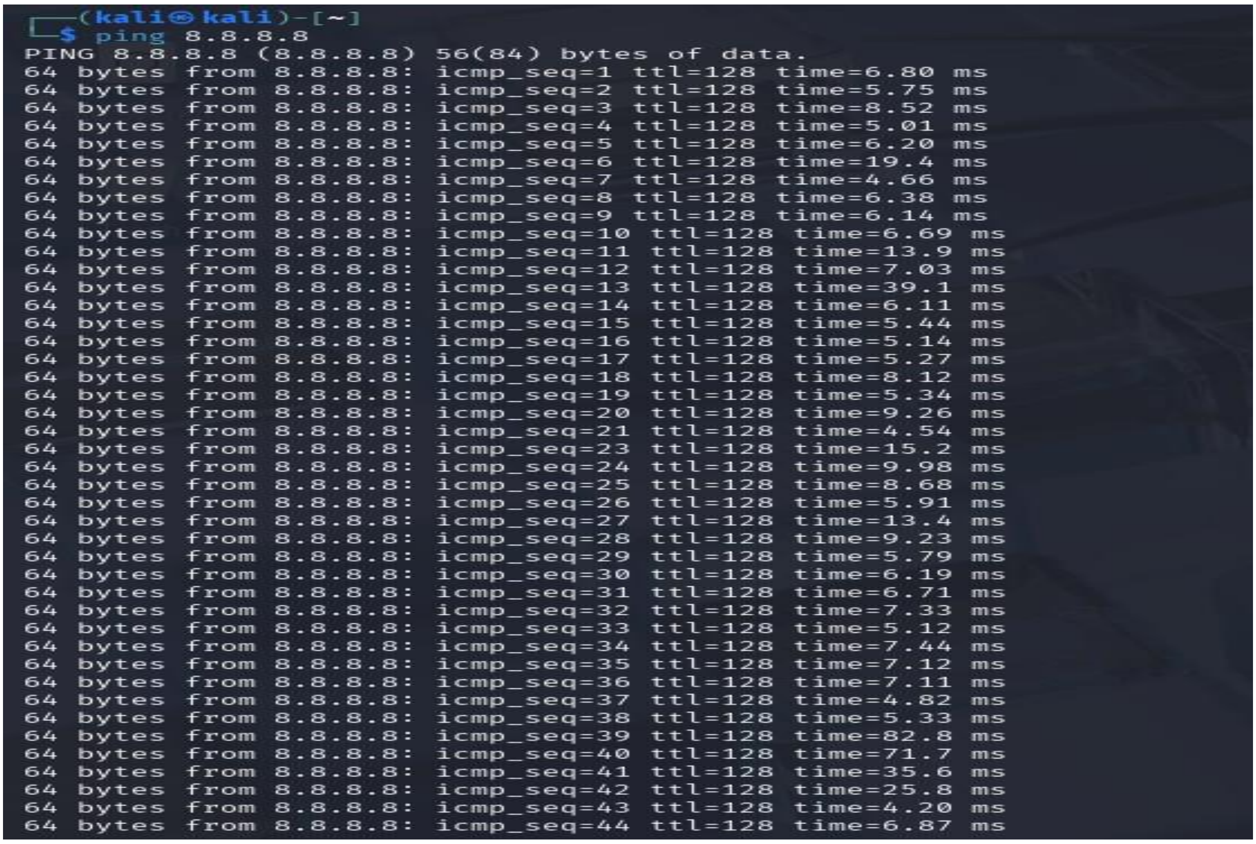
```
        netflow
        active
        pop
        rpc_decode
        sip
        ssl
        telnet
        cip
        iec104
        mms
        modbus
        s7commplus
        dce_smb
        dce_tcp
        dce_http_server
        alerts
        decode
        host_cache
        host_tracker
        network
        ftp_client
Finished /home/kali/snort.lua:
Loading file_id.rules_file:
Loading /etc/snort/file_magic.rules:
Finished /etc/snort/file_magic.rules:
Finished file_id.rules_file:
--------------------------------------------------
ips policies rule stats
              id  loaded  shared enabled    file
               0     208       0     208    /home/kali/snort.lua
--------------------------------------------------
rule counts
       total rules loaded: 208
               text rules: 208
            option chains: 208
            chain headers: 1
--------------------------------------------------
service rule counts          to-srv  to-cli
                  file_id:      208     208
                    total:      208     208
--------------------------------------------------
fast pattern groups
                to_server: 1
                to_client: 1
--------------------------------------------------
search engine (ac_bnfa)
                instances: 2
                 patterns: 416
```



```
             pattern chars: 2508
                num states: 1778
          num match states: 370
              memory scale: KB
              total memory: 48.3691
            pattern memory: 12.1973
         match list memory: 13.6641
         transition memory: 22.3125
appid: MaxRss diff: 1920
appid: patterns loaded: 300
--------------------------------------------------
pcap DAQ configured to passive.
Commencing packet processing
++ [0] eth0
```

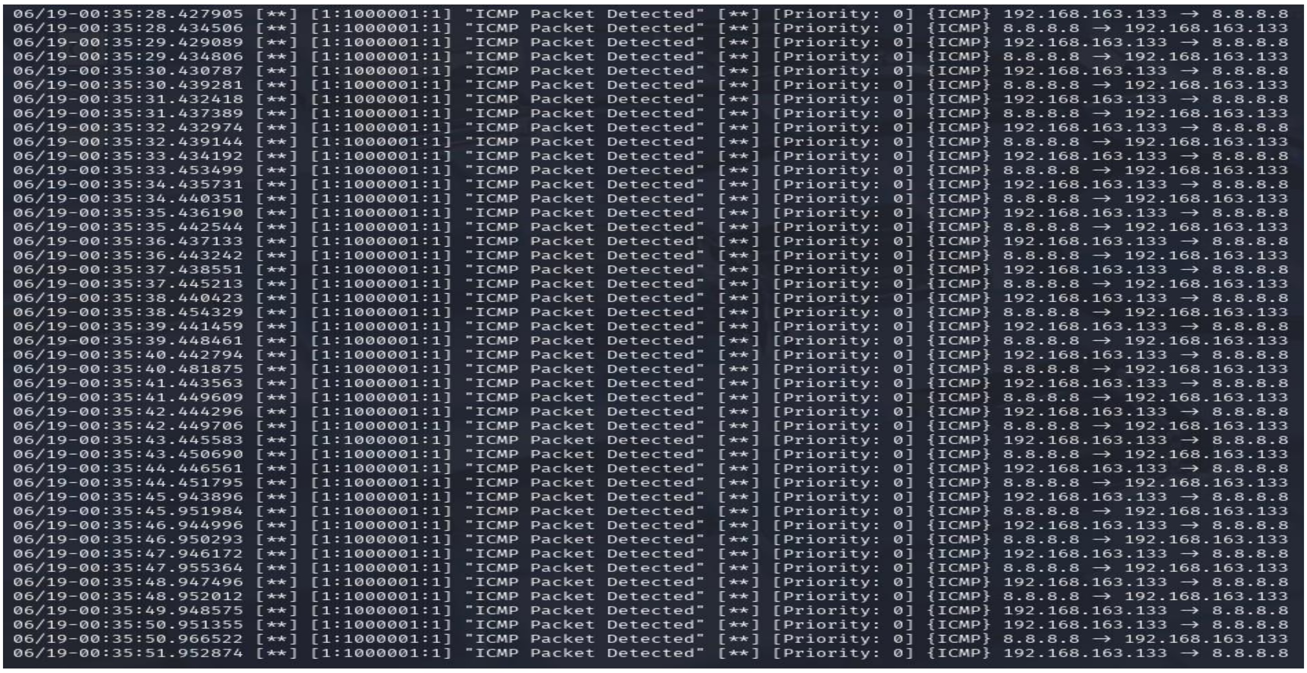
Now open new terminal tab and ping:

Ping 8.8.8.8



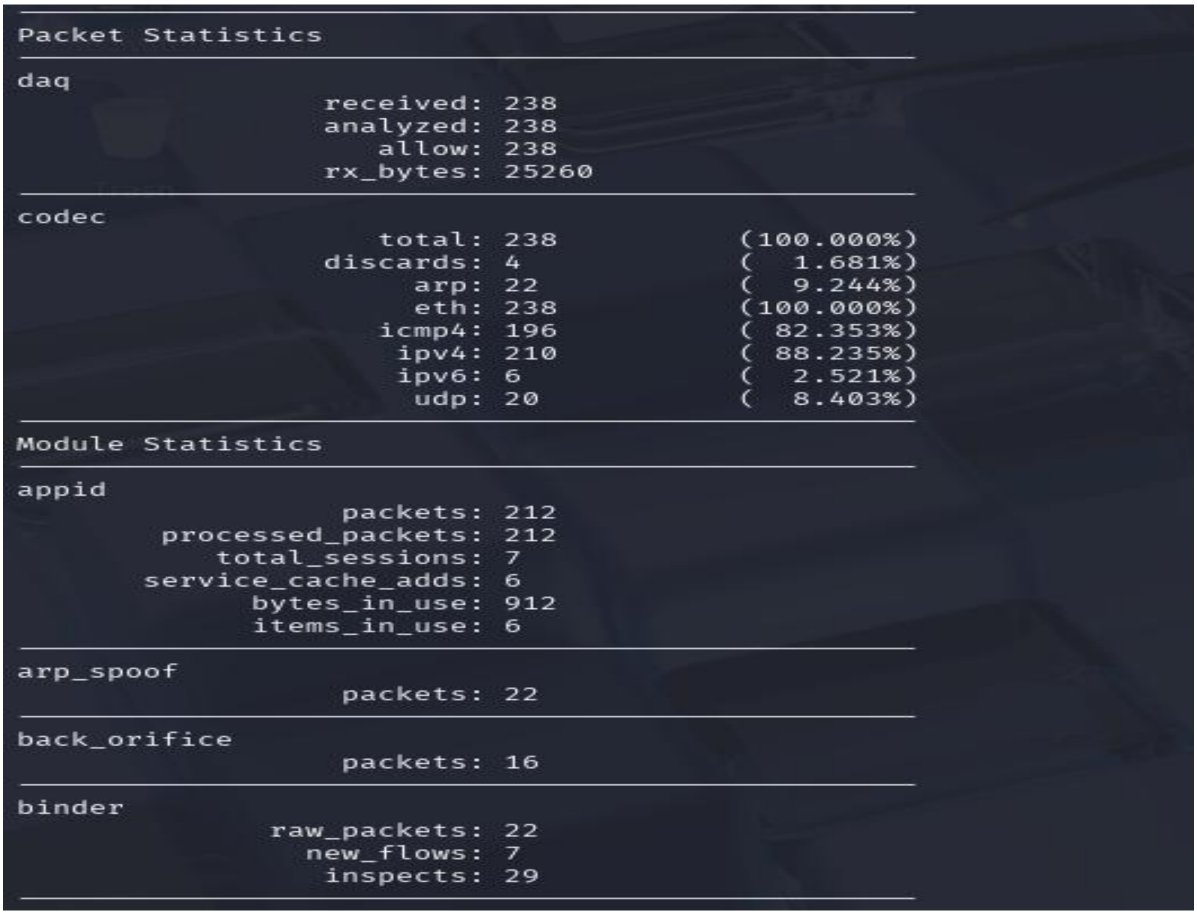
```
┌──(kali㉿kali)-[~]
└─$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=6.80 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=5.75 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=128 time=8.52 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=128 time=5.01 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=128 time=6.20 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=128 time=19.4 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=128 time=4.66 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=128 time=6.38 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=128 time=6.14 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=128 time=6.69 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=128 time=13.9 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=128 time=7.03 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=128 time=39.1 ms
64 bytes from 8.8.8.8: icmp_seq=14 ttl=128 time=6.11 ms
64 bytes from 8.8.8.8: icmp_seq=15 ttl=128 time=5.44 ms
64 bytes from 8.8.8.8: icmp_seq=16 ttl=128 time=5.14 ms
64 bytes from 8.8.8.8: icmp_seq=17 ttl=128 time=5.27 ms
64 bytes from 8.8.8.8: icmp_seq=18 ttl=128 time=8.12 ms
64 bytes from 8.8.8.8: icmp_seq=19 ttl=128 time=5.34 ms
64 bytes from 8.8.8.8: icmp_seq=20 ttl=128 time=9.26 ms
64 bytes from 8.8.8.8: icmp_seq=21 ttl=128 time=4.54 ms
64 bytes from 8.8.8.8: icmp_seq=23 ttl=128 time=15.2 ms
64 bytes from 8.8.8.8: icmp_seq=24 ttl=128 time=9.98 ms
64 bytes from 8.8.8.8: icmp_seq=25 ttl=128 time=8.68 ms
64 bytes from 8.8.8.8: icmp_seq=26 ttl=128 time=5.91 ms
64 bytes from 8.8.8.8: icmp_seq=27 ttl=128 time=13.4 ms
64 bytes from 8.8.8.8: icmp_seq=28 ttl=128 time=9.23 ms
64 bytes from 8.8.8.8: icmp_seq=29 ttl=128 time=5.79 ms
64 bytes from 8.8.8.8: icmp_seq=30 ttl=128 time=6.19 ms
64 bytes from 8.8.8.8: icmp_seq=31 ttl=128 time=6.71 ms
64 bytes from 8.8.8.8: icmp_seq=32 ttl=128 time=7.33 ms
64 bytes from 8.8.8.8: icmp_seq=33 ttl=128 time=5.12 ms
64 bytes from 8.8.8.8: icmp_seq=34 ttl=128 time=7.44 ms
64 bytes from 8.8.8.8: icmp_seq=35 ttl=128 time=7.12 ms
64 bytes from 8.8.8.8: icmp_seq=36 ttl=128 time=7.11 ms
64 bytes from 8.8.8.8: icmp_seq=37 ttl=128 time=4.82 ms
64 bytes from 8.8.8.8: icmp_seq=38 ttl=128 time=5.33 ms
64 bytes from 8.8.8.8: icmp_seq=39 ttl=128 time=82.8 ms
64 bytes from 8.8.8.8: icmp_seq=40 ttl=128 time=71.7 ms
64 bytes from 8.8.8.8: icmp_seq=41 ttl=128 time=35.6 ms
64 bytes from 8.8.8.8: icmp_seq=42 ttl=128 time=25.8 ms
64 bytes from 8.8.8.8: icmp_seq=43 ttl=128 time=4.20 ms
64 bytes from 8.8.8.8: icmp_seq=44 ttl=128 time=6.87 ms
```

After Pinging, Go back where you started snort you’ll detect Packets:

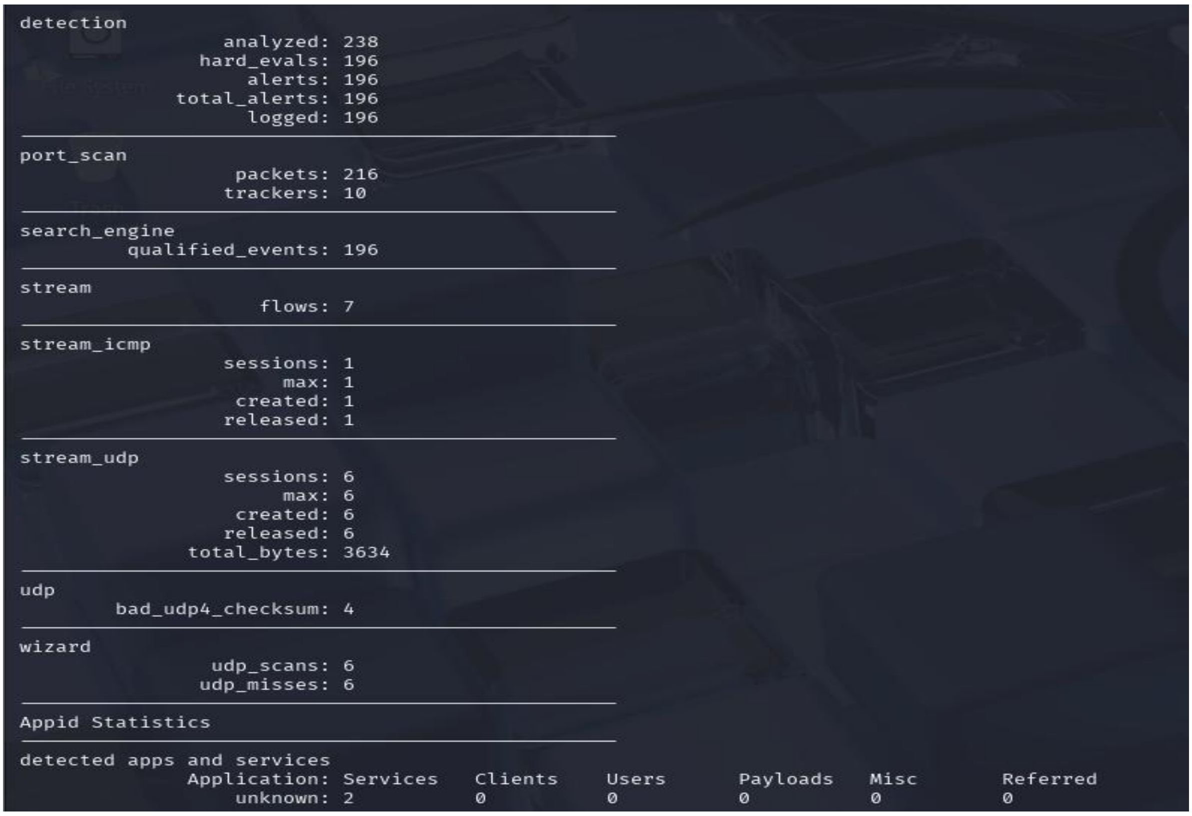


```
06/19-00:35:28.427905 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:28.434506 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:29.429089 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:29.434806 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:30.430787 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:30.439281 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:31.432418 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:31.437389 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:32.432974 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:32.439144 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:33.434192 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:33.453499 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:34.435731 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:34.440351 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:35.436190 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:35.442544 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:36.437133 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:36.443242 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:37.438551 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:37.445213 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:38.440423 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:38.454329 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:39.441459 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:39.448461 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:40.442794 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:40.481875 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:41.443563 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:41.449609 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:42.444296 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:42.449706 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:43.445583 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:43.450690 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:44.446561 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:44.451795 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:45.943896 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:45.951984 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:46.944996 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:46.950293 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:47.946172 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:47.955364 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:48.947496 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:48.952012 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:49.948575 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:50.951355 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:50.966522 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:51.952874 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
```

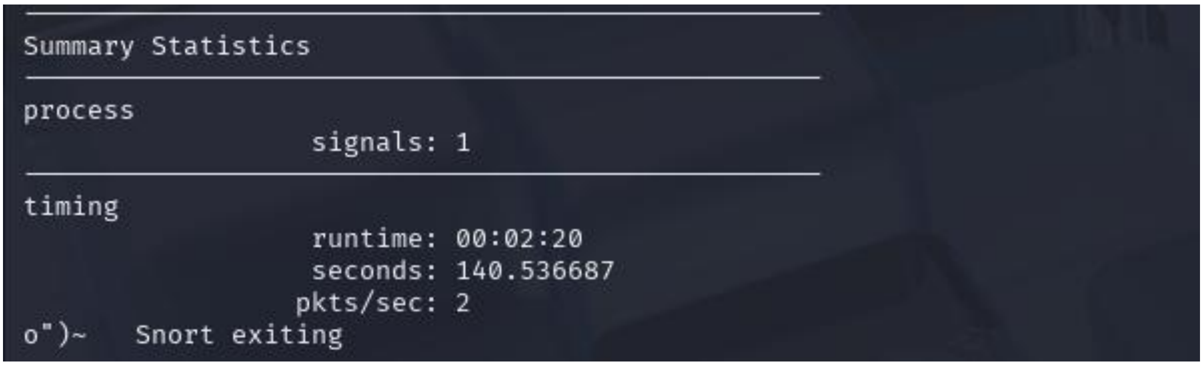
After killing or stopping the process:



```
--------------------------------------------------
Packet Statistics
--------------------------------------------------
daq
                 received: 238
                 analyzed: 238
                    allow: 238
                 rx_bytes: 25260
--------------------------------------------------
codec
                    total: 238          (100.000%)
                 discards: 4            (  1.681%)
                      arp: 22           (  9.244%)
                      eth: 238          (100.000%)
                    icmp4: 196          ( 82.353%)
                     ipv4: 210          ( 88.235%)
                     ipv6: 6            (  2.521%)
                      udp: 20           (  8.403%)
--------------------------------------------------
Module Statistics
--------------------------------------------------
appid
                  packets: 212
        processed_packets: 212
           total_sessions: 7
       service_cache_adds: 6
             bytes_in_use: 912
             items_in_use: 6
--------------------------------------------------
arp_spoof
                  packets: 22
--------------------------------------------------
back_orifice
                  packets: 16
--------------------------------------------------
binder
              raw_packets: 22
                new_flows: 7
                 inspects: 29
--------------------------------------------------
```



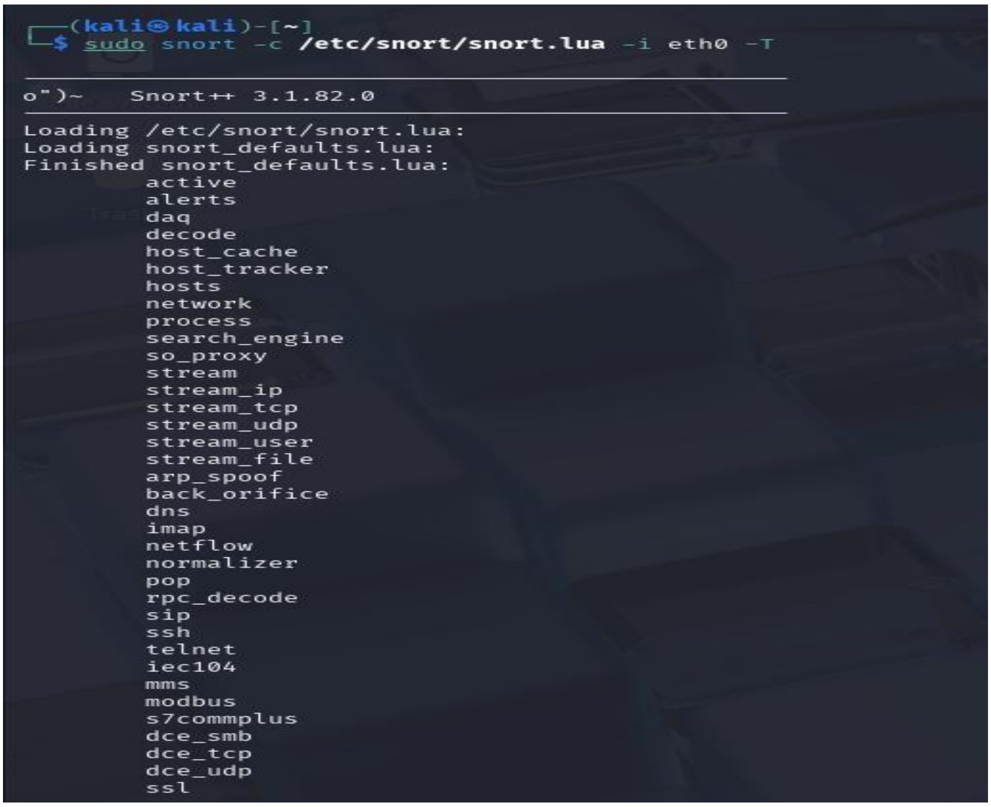
```
detection
                 analyzed: 238
               hard_evals: 196
                   alerts: 196
             total_alerts: 196
                   logged: 196
--------------------------------------------------
port_scan
                  packets: 216
                 trackers: 10
--------------------------------------------------
search_engine
         qualified_events: 196
--------------------------------------------------
stream
                    flows: 7
--------------------------------------------------
stream_icmp
                 sessions: 1
                      max: 1
                  created: 1
                 released: 1
--------------------------------------------------
stream_udp
                 sessions: 6
                      max: 6
                  created: 6
                 released: 6
              total_bytes: 3634
--------------------------------------------------
udp
        bad_udp4_checksum: 4
--------------------------------------------------
wizard
                udp_scans: 6
               udp_misses: 6
--------------------------------------------------
Appid Statistics
--------------------------------------------------
detected apps and services
              Application: Services   Clients    Users      Payloads   Misc       Referred
                  unknown: 2          0          0          0          0          0
```



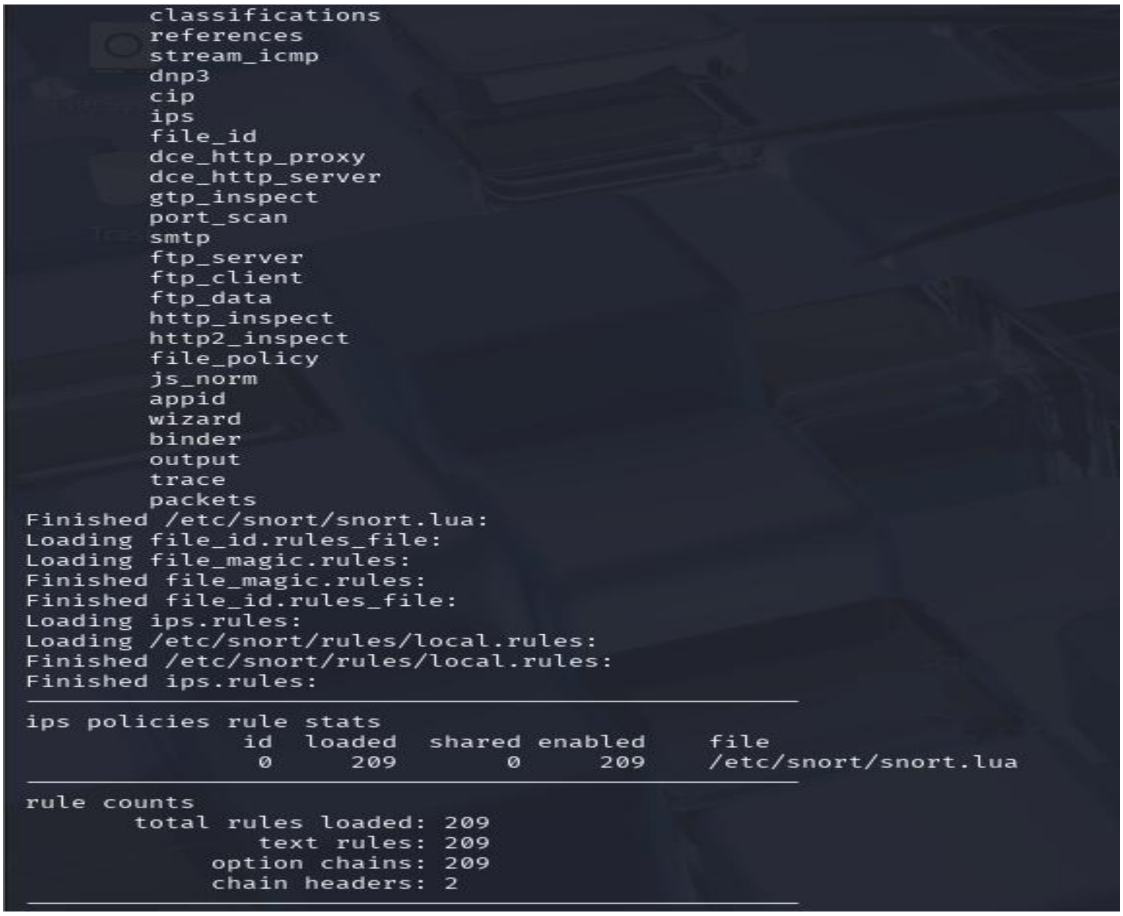
```
--------------------------------------------------
Summary Statistics
--------------------------------------------------
process
                  signals: 1
--------------------------------------------------
timing
                  runtime: 00:02:20
                  seconds: 140.536687
                 pkts/sec: 2
o")~   Snort exiting
```

• Run Snort in detection mode

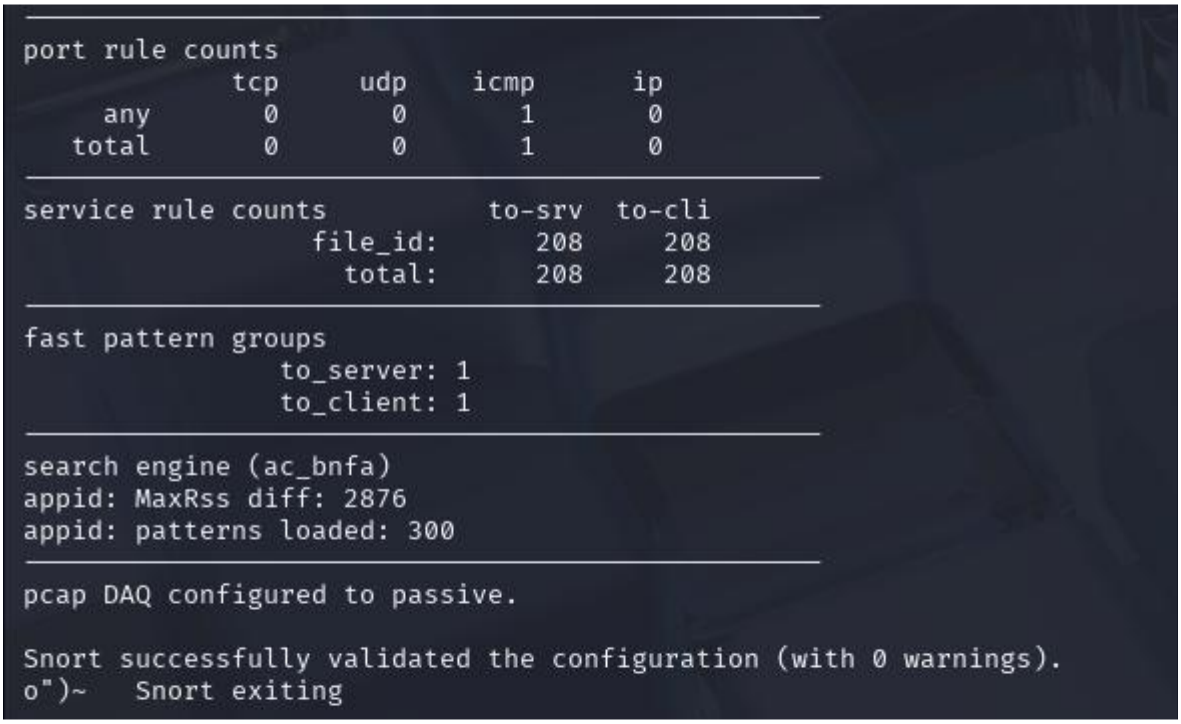
sudo snort -c /etc/snort/snort.lua -i eth0 -T



```
┌──(kali㉿kali)-[~]
└─$ sudo snort -c /etc/snort/snort.lua -i eth0 -T
--------------------------------------------------
o")~   Snort++ 3.1.82.0
--------------------------------------------------
Loading /etc/snort/snort.lua:
Loading snort_defaults.lua:
Finished snort_defaults.lua:
        active
        alerts
        daq
        decode
        host_cache
        host_tracker
        hosts
        network
        process
        search_engine
        so_proxy
        stream
        stream_ip
        stream_tcp
        stream_udp
        stream_user
        stream_file
        arp_spoof
        back_orifice
        dns
        imap
        netflow
        normalizer
        pop
        rpc_decode
        sip
        ssh
        telnet
        iec104
        mms
        modbus
        s7commplus
        dce_smb
        dce_tcp
        dce_udp
        ssl
```



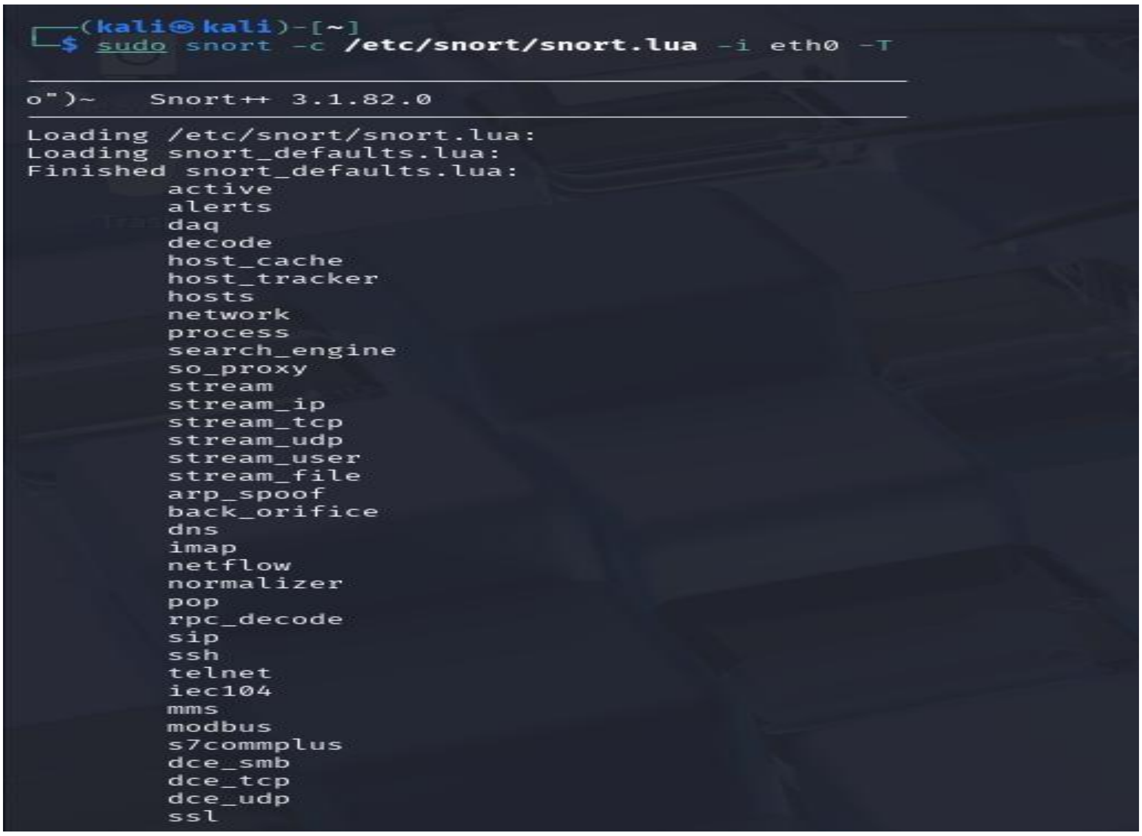
```
        classifications
        references
        stream_icmp
        dnp3
        cip
        ips
        file_id
        dce_http_proxy
        dce_http_server
        gtp_inspect
        port_scan
        smtp
        ftp_server
        ftp_client
        ftp_data
        http_inspect
        http2_inspect
        file_policy
        js_norm
        appid
        wizard
        binder
        output
        trace
        packets
Finished /etc/snort/snort.lua:
Loading file_id.rules_file:
Loading file_magic.rules:
Finished file_magic.rules:
Finished file_id.rules_file:
Loading ips.rules:
Loading /etc/snort/rules/local.rules:
Finished /etc/snort/rules/local.rules:
Finished ips.rules:
--------------------------------------------------
ips policies rule stats
              id  loaded  shared enabled    file
               0     209       0     209    /etc/snort/snort.lua
--------------------------------------------------
rule counts
       total rules loaded: 209
               text rules: 209
            option chains: 209
            chain headers: 2
--------------------------------------------------
```



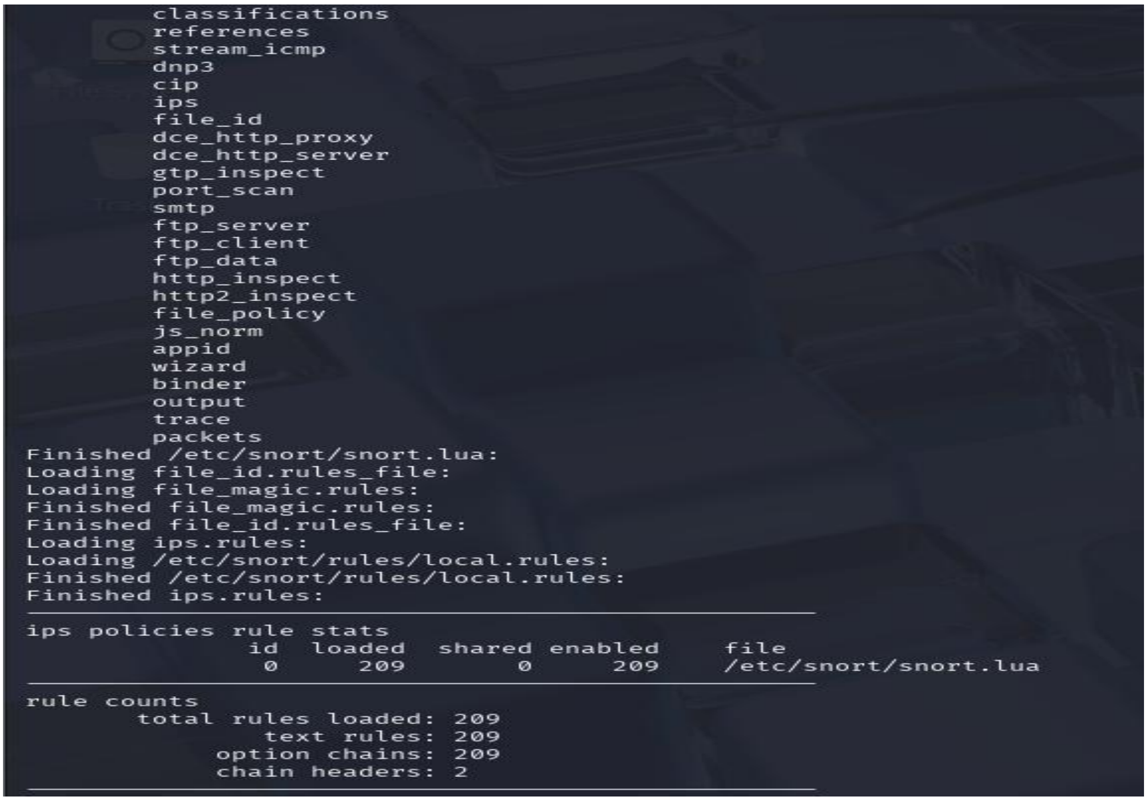
```
--------------------------------------------------
port rule counts
             tcp     udp    icmp      ip
     any       0       0       1       0
   total       0       0       1       0
--------------------------------------------------
service rule counts          to-srv  to-cli
                  file_id:      208     208
                    total:      208     208
--------------------------------------------------
fast pattern groups
                to_server: 1
                to_client: 1
--------------------------------------------------
search engine (ac_bnfa)
appid: MaxRss diff: 2876
appid: patterns loaded: 300
--------------------------------------------------
pcap DAQ configured to passive.

Snort successfully validated the configuration (with 0 warnings).
o")~   Snort exiting
```

• Monitor live traffic and alerts

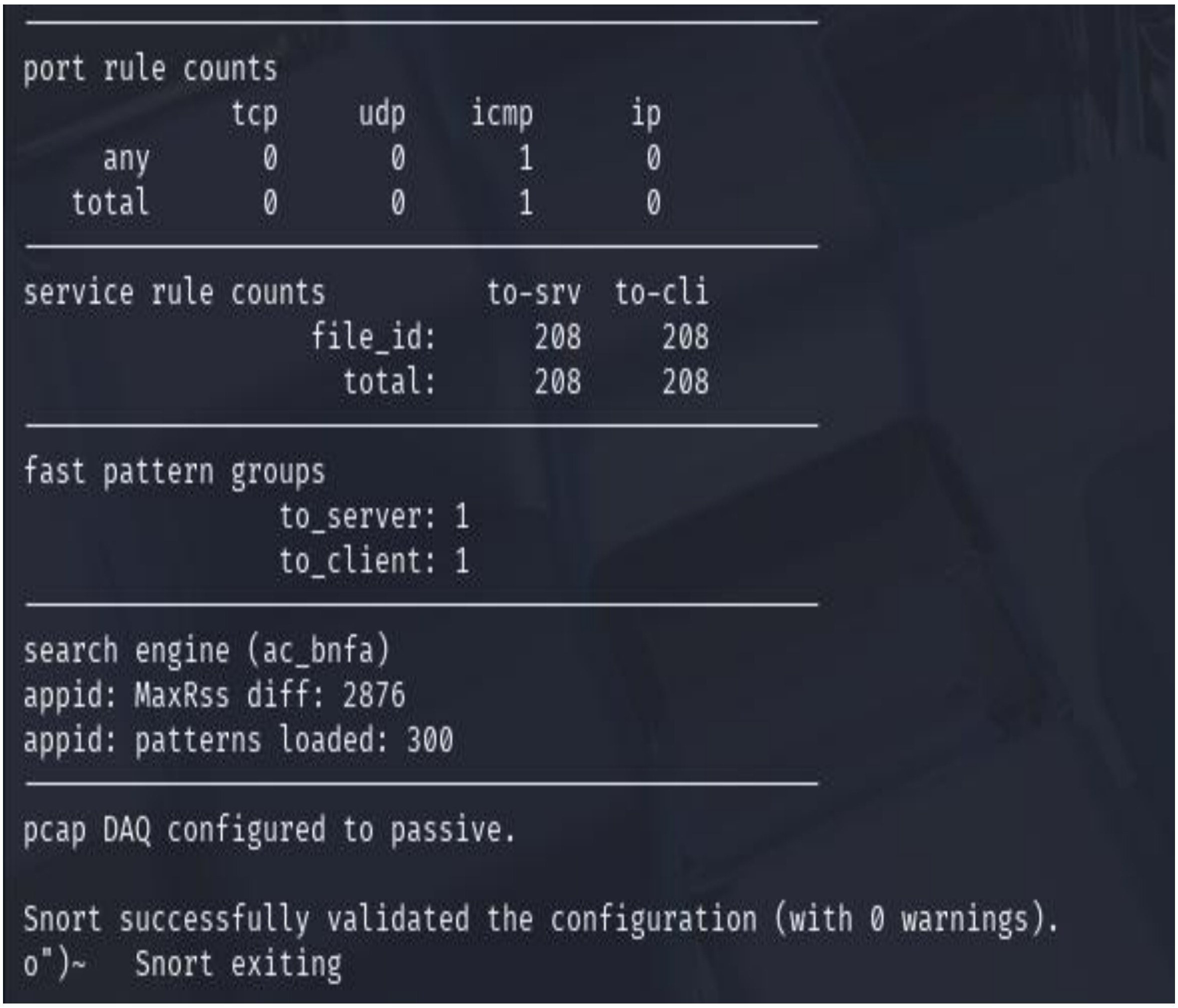
sudo snort -c /etc/snort/snort.lua -i eth0 -T



```
┌──(kali㉿kali)-[~]
└─$ sudo snort -c /etc/snort/snort.lua -i eth0 -T
--------------------------------------------------
o")~   Snort++ 3.1.82.0
--------------------------------------------------
Loading /etc/snort/snort.lua:
Loading snort_defaults.lua:
Finished snort_defaults.lua:
        active
        alerts
        daq
        decode
        host_cache
        host_tracker
        hosts
        network
        process
        search_engine
        so_proxy
        stream
        stream_ip
        stream_tcp
        stream_udp
        stream_user
        stream_file
        arp_spoof
        back_orifice
        dns
        imap
        netflow
        normalizer
        pop
        rpc_decode
        sip
        ssh
        telnet
        iec104
        mms
        modbus
        s7commplus
        dce_smb
        dce_tcp
        dce_udp
        ssl
```



```
                classifications
                references
                stream_icmp
                dnp3
                cip
                ips
                file_id
                dce_http_proxy
                dce_http_server
                gtp_inspect
                port_scan
                smtp
                ftp_server
                ftp_client
                ftp_data
                http_inspect
                http2_inspect
                file_policy
                js_norm
                appid
                wizard
                binder
                output
                trace
                packets
Finished /etc/snort/snort.lua:
Loading file_id.rules_file:
Loading file_magic.rules:
Finished file_magic.rules:
Finished file_id.rules_file:
Loading ips.rules:
Loading /etc/snort/rules/local.rules:
Finished /etc/snort/rules/local.rules:
Finished ips.rules:
--------------------------------------------------
ips policies rule stats
              id  loaded  shared enabled    file
               0     209       0     209    /etc/snort/snort.lua
--------------------------------------------------
rule counts
       total rules loaded: 209
               text rules: 209
            option chains: 209
            chain headers: 2
--------------------------------------------------
```



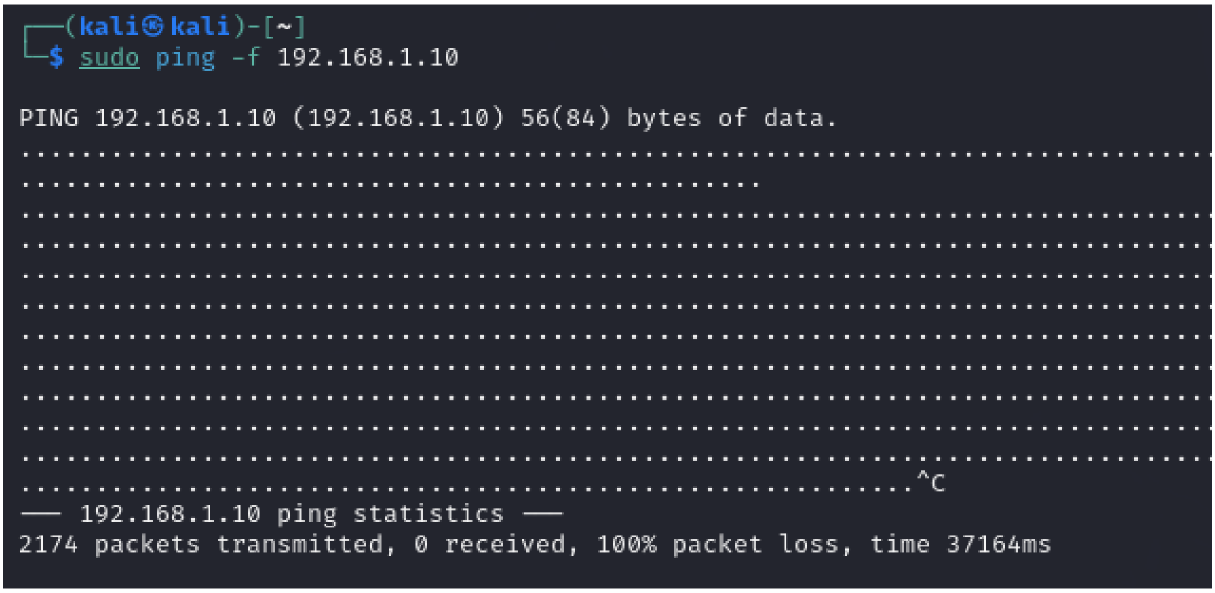
```
--------------------------------------------------
port rule counts
             tcp     udp    icmp      ip
     any       0       0       1       0
   total       0       0       1       0
--------------------------------------------------
service rule counts          to-srv  to-cli
                  file_id:      208     208
                    total:      208     208
--------------------------------------------------
fast pattern groups
                to_server: 1
                to_client: 1
--------------------------------------------------
search engine (ac_bnfa)
appid: MaxRss diff: 2876
appid: patterns loaded: 300
--------------------------------------------------
pcap DAQ configured to passive.

Snort successfully validated the configuration (with 0 warnings).
o")~   Snort exiting
```

Week 3

• Simulate attacks (e.g., ping flood)

1. Basic Ping Flood (Using ping)

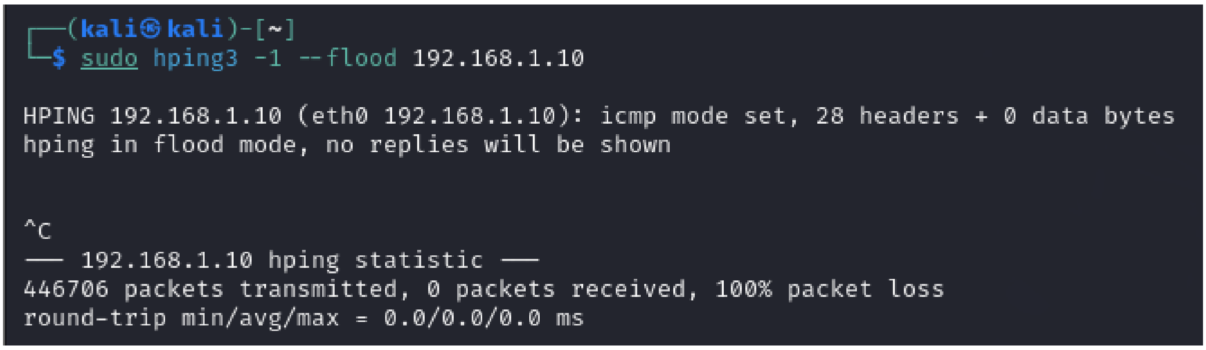
sudo ping -f 192.168.1.10



```
┌──(kali㉿kali)-[~]
└─$ sudo ping -f 192.168.1.10

PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
...............................................................................
.................................................
...............................................................................
...............................................................................
...............................................................................
...............................................................................
...............................................................................
...............................................................................
...............................................................................
...............................................................................
...............................................................................
............................................................^C
--- 192.168.1.10 ping statistics ---
2174 packets transmitted, 0 received, 100% packet loss, time 37164ms
```

2. Advanced Ping Flood (Using hping3)

sudo hping3 -1 --flood 192.168.1.10

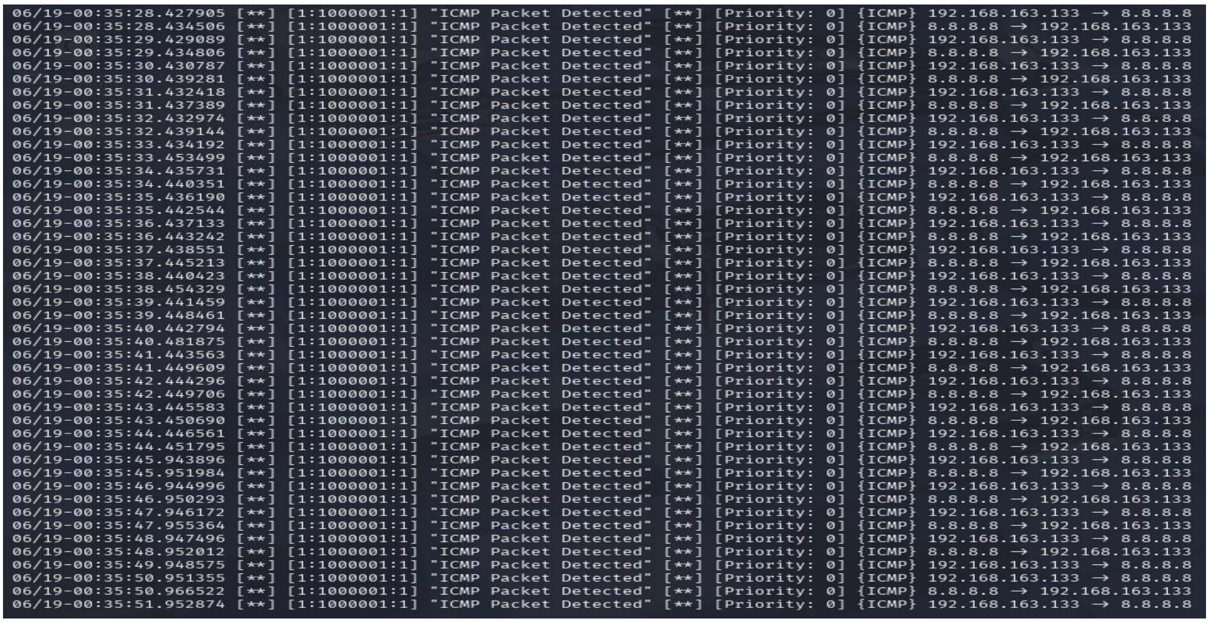


```
┌──(kali㉿kali)-[~]
└─$ sudo hping3 -1 --flood 192.168.1.10

HPING 192.168.1.10 (eth0 192.168.1.10): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown


^C
--- 192.168.1.10 hping statistic ---
446706 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Use this for learning:

- How firewalls react
- How to detect ICMP floods
- How to create signatures for IDS/IPS (e.g., Snort or Suricata)

• Observe Snort alerts



```
06/19-00:35:28.427905 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:28.434506 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:29.429089 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:29.434806 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:30.430787 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:30.439281 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:31.432418 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:31.437389 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:32.432974 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:32.439144 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:33.434192 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:33.453499 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:34.435731 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:34.440351 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:35.436190 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:35.442544 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:36.437133 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:36.443242 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:37.438551 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:37.445213 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:38.440423 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:38.454329 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:39.441459 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:39.448461 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:40.442794 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:40.481875 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:41.443563 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:41.449609 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:42.444296 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:42.449706 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:43.445583 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:43.450690 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:44.446561 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:44.451795 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:45.943896 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:45.951984 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:46.944996 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:46.950293 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:47.946172 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:47.955364 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:48.947496 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:48.952012 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:49.948575 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:50.951355 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
06/19-00:35:50.966522 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 8.8.8.8 → 192.168.163.133
06/19-00:35:51.952874 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 → 8.8.8.8
```

• Understand Snort alert formats

Example in our case:

06/19-00:35:28.427905 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 -> 8.8.8.8

Field	Description
Timestamp	06/19-00:35:28.427905 – The date and time when the alert was triggered. Format: MM/DD-HH:MM:SS.milliseconds.
Alert Markers	[**] – Visual separators to distinguish alert sections.
Rule Metadata	[1:1000001:1] – This consists of: • 1: Generator ID (GID), indicates Snort itself triggered the alert. • 1000001: Signature ID (SID), uniquely identifies the rule. • 1: Rule revision number. | | **Message** | "ICMP Packet Detected" – The alert message defined in the msg: field of the rule. | | **Priority** | [Priority: 0] – Indicates severity (0 = lowest). Priority is set manually in the rule or inferred. | | **Protocol** | {ICMP} – The detected packet's protocol. | | **Source → Destination** | 192.168.163.133 -> 8.8.8.8 – The IP addresses of the packet's origin and destination. |

• Review alert logs

Review in our case:

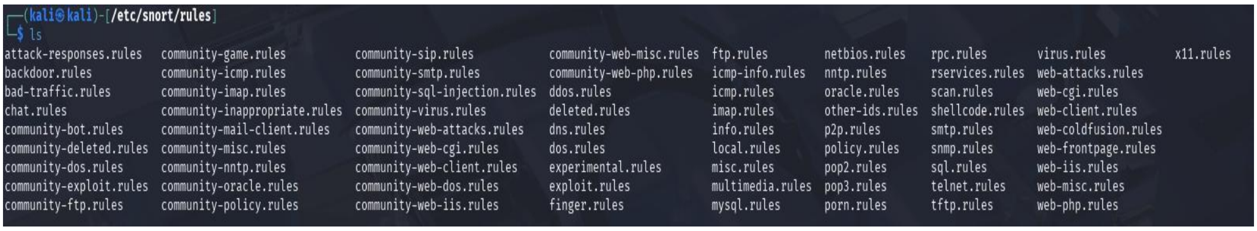
06/19-00:35:28.427905 [**] [1:1000001:1] "ICMP Packet Detected" [**] [Priority: 0] {ICMP} 192.168.163.133 -> 8.8.8.8

This log tells you:

- A Snort rule was triggered by **ICMP traffic**
- The alert was logged on **June 19 at 00:35:28**
- The **source IP** was your machine (192.168.163.133)
- The **destination** was Google DNS (8.8.8.8)

WEEK 4

• Explore default Snort rules and structure



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

• Learn rule components (actions, protocols, etc.)

Rule:

alert icmp any any -> any any (msg:"ICMP Packet Detected"; sid:1000001; rev:1;)

Component	Value	Description
Action	alert	Tells Snort to generate an alert when this rule matches traffic
Protocol	icmp	Matches ICMP packets (used in ping, traceroute, etc.)
Source IP	any	Matches traffic from any source IP
Source Port	any	ICMP doesn't use ports, but format requires this
Direction	->	Matches traffic from source to destination
Destination IP	any	Matches traffic to any destination IP
Destination Port	any	Placeholder, ICMP does not use ports
Options	(msg:"ICMP Packet Detected"; sid:1000001; rev:1;)	Rule-specific metadata and message

• Prepare a basic report with screenshots on configuration and alerts

The screenshots for the following tasks have already been included in their relative sections in this report:

1. Identifying active network interface
2. Configuring Snort with the monitored IP range.
3. Running Snort in detection mode.
4. Monitoring live traffic and alerts.
5. Simulating attacks such as ping and ping flood.
6. Observing Snort alerts.
7. Understanding Snort alert formats.
8. Reviewing Snort alert logs.

What I got to learn?

Using Snort as an Intrusion Detection System (IDS), I was able to obtain hands-on experience in network security monitoring. Important lessons learnt include:

1. Setting up and installing Snort on a Linux computer.
2. Finding and keeping an eye on active network interfaces.
3. Creating and evaluating unique Snort rules with appropriate syntax and organisation.
4. Using traffic simulation (ICMP, ping flood, etc.) to set off alarms.
5. Examining log files and Snort alert analysis.
6. Using key Linux commands for log analysis and configuration.

My knowledge of network traffic analysis and real-time intrusion detection has improved as a result of this experience.

Summary – Month 1

It was found for Month 1 that installation, configuration, and testing of Snort were undertaken successfully. One custom ICMP detection rule was created, and alerts were generated using the ping and hping3 tools. Snort was started in detection mode, and the logs were reviewed for verification of rule efficacy.

While understanding and explaining the rule structure and alert format, screenshots were provided for all major steps, including rule setup, configuration, traffic simulation, and alert output.

All Month 1 milestones were achieved, thus setting a firm base for further developments and testing of the IDS.