

TASK 4: Network Intrusion Detection System

Step 1: Install Suricata (Recommended)

Command: sudo apt update

```
sudo apt install suricata -y
```

Verify installation:

Command: suricata --version

```
(kali㉿kali)-[~]
$ suricata --version
suricata: unrecognized option '--version'
Suricata 7.0.10
USAGE: suricata [OPTIONS] [BPF FILTER]

      -c <path>          : path to configuration file
      -T                  : test configuration file (use with -c)
      -i <dev or ip>     : run in pcap live mode
      -F <bpf filter file> : bpf filter file
      -r <path>          : run in pcap file/offline mode
      -q <qid[:qid]>     : run in inline nfqueue mode (use colon to specify a range of queues)
      -s <path>          : path to signature file loaded in addition to suricata.yaml settings (optional)
      -S <path>          : path to signature file loaded exclusively (optional)
      -l <dir>           : default log directory
      -D                  : run as daemon
      -k [all|none]       : force checksum check (all) or disabled it (none)
      -V                  : display Suricata version
      -v                  : be more verbose (use multiple times to increase verbosity)
--list-app-layer-protos   : list supported app layer protocols
--list-keywords[=all|csv|<kword>] : list keywords implemented by the engine
--list-runmodes            : list supported runmodes
--runmode <runmode_id>    : specific runmode modification the engine should run. The argument supplied should be the id for the runmode obtained by running --list-runmodes
--engine-analysis          : print reports on analysis of different sections in the engine and exit.
                           Please have a look at the conf parameter engine-analysis on what reports can be printed
--pidfile <file>          : write pid to this file
--init-errors-fatal        : enable fatal failure on signature init error
--disable-detection        : disable detection engine
--dump-config              : show the running configuration
--dump-features             : display provided features
```

Step 2: Identify Your Network Interface

Run this to find your active interface (e.g., eth0, wlan0):

Command: 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:29:5e:b4:13 brd ff:ff:ff:ff:ff:ff
  inet 192.168.213.128/24 brd 192.168.213.255 scope global dynamic noprefixroute eth0
    valid_lft 1346sec preferred_lft 1346sec
    inet6 fe80::bac5:2857:f12c:57cc/64 scope link noprefixroute
      valid_lft forever preferred_lft forever
```

Note down the correct interface for monitoring.

Step 3: Configure Suricata

Edit the config file:

Command: sudo nano /etc/suricata/suricata.yaml

```
##  
## Step 3: Configure common capture settings  
##  
## See "Advanced Capture Options" below for more options, including Netmap  
## and PF_RING.  
##  
  
# Linux high speed capture support  
af-packet:  
- Interface: eth0  
  # Number of receive threads. "auto" uses the number of cores  
  #threads: auto  
  # Default clusterid. AF_PACKET will load balance packets based on flow.  
  cluster-id: 99
```

Step 4: Enable and Test Suricata Rules

Suricata uses .rules files (like Snort). Make sure rules are enabled.

You can download or update rule sets:

Command: sudo suricata-update

```
[(kali㉿kali)-~]$ sudo suricata-update  
5/7/2025 -- 02:27:11 - <Info> -- Using data-directory /var/lib/suricata.  
5/7/2025 -- 02:27:11 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml  
5/7/2025 -- 02:27:11 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.  
5/7/2025 -- 02:27:11 - <Info> -- Found Suricata version 7.0.10 at /usr/bin/suricata.  
5/7/2025 -- 02:27:11 - <Info> -- Loading /etc/suricata/suricata.yaml  
5/7/2025 -- 02:27:11 - <Info> -- Disabling rules for protocol pgsql  
5/7/2025 -- 02:27:11 - <Info> -- Disabling rules for protocol modbus  
5/7/2025 -- 02:27:11 - <Info> -- Disabling rules for protocol dnp3  
5/7/2025 -- 02:27:11 - <Info> -- Disabling rules for protocol enip  
5/7/2025 -- 02:27:11 - <Info> -- No sources configured, will use Emerging Threats Open  
5/7/2025 -- 02:27:11 - <Info> -- Fetching https://rules.emergingthreats.net/open/suricata-7.0.10/emerging.rules.tar.gz.  
100% - 4961765/4961765  
5/7/2025 -- 02:27:19 - <Info> -- Done.  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/app-layer-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/decoder-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/dhcp-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/dnp3-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/dns-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/files.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/http2-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/http-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/ipsec-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/kerberos-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/modbus-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/mqtt-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/nfs-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/ntp-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/quic-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/rfb-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/smb-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/smtp-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/ssh-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/stream-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Loading distribution rule file /etc/suricata/rules/tls-events.rules  
5/7/2025 -- 02:27:20 - <Info> -- Ignoring file f625293e2432dbf07497d06349de6f0b/rules/emerging-deleted.rules  
5/7/2025 -- 02:27:22 - <Info> -- Loaded 59707 rules.  
5/7/2025 -- 02:27:22 - <Info> -- Disabled 13 rules.  
5/7/2025 -- 02:27:22 - <Info> -- Enabled 0 rules.  
5/7/2025 -- 02:27:22 - <Info> -- Modified 0 rules.  
5/7/2025 -- 02:27:22 - <Info> -- Dropped 0 rules.  
5/7/2025 -- 02:27:23 - <Info> -- Enabled 136 rules for flowbit dependencies.  
5/7/2025 -- 02:27:23 - <Info> -- Backing up current rules.  
5/7/2025 -- 02:27:23 - <Info> -- Writing rules to /var/lib/suricata/rules/suricata.rules: total: 59707; enabled: 44113; added: 59707; remov
```

You'll find default rules in:

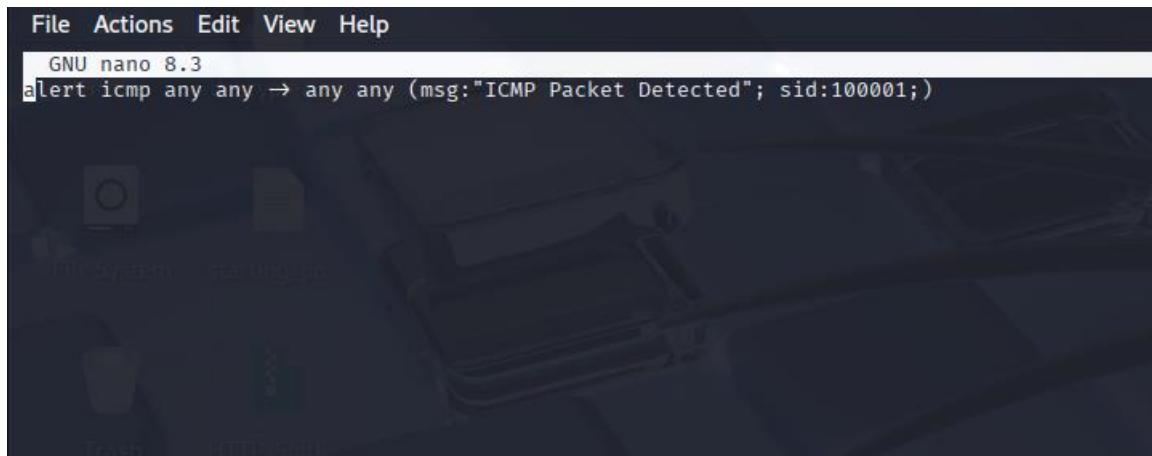
Command: /etc/suricata/rules/

Example rule format (to detect ICMP ping):

```
alert icmp any any -> any any (msg:"ICMP Packet Detected"; sid:100001;)
```

Add this to a custom rule file:

Command: sudo nano /etc/suricata/rules/custom.rules



```
File Actions Edit View Help
GNU nano 8.3
alert icmp any any → any any (msg:"ICMP Packet Detected"; sid:100001;)
```

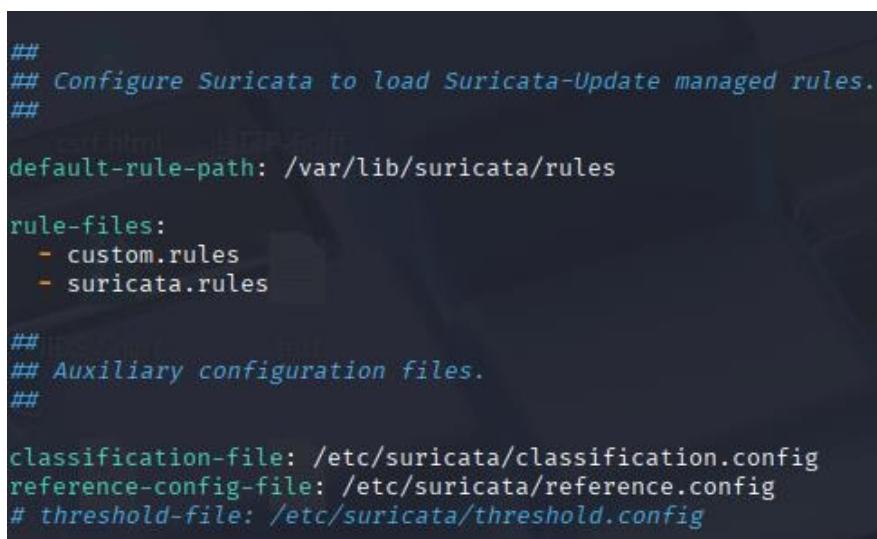
And add the rule above.

Now include that in your config:

Command: sudo nano /etc/suricata/suricata.yaml

And add this under rule-files:

- custom.rules



```
## Configure Suricata to load Suricata-Update managed rules.
##

default-rule-path: /var/lib/suricata/rules

rule-files:
  - custom.rules
  - suricata.rules

## Auxiliary configuration files.
##

classification-file: /etc/suricata/classification.config
reference-config-file: /etc/suricata/reference.config
# threshold-file: /etc/suricata/threshold.config
```

Step 5: Start Suricata in Live Mode

Now run Suricata in **IDS mode**:

Command: sudo suricata -c /etc/suricata/suricata.yaml -i eth0

This will monitor traffic **live**.

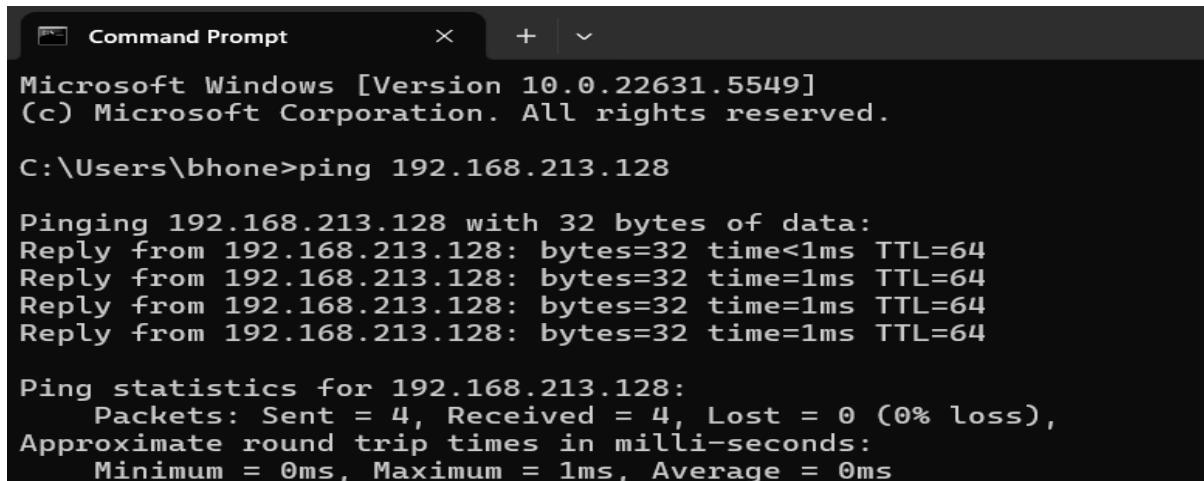
```
(kali㉿kali)-[~]
└─$ sudo suricata -c /etc/suricata/suricata.yaml -i eth0
i: suricata: This is Suricata version 7.0.10 RELEASE running in SYSTEM mode
W: af-packet: eth0: AF_PACKET tpacket-v3 is recommended for non-inline operation
i: threads: Threads created → W: 4 FM: 1 FR: 1   Engine started.
```

Step 6: Simulate Attacks / Scan

From another machine in same network:

Command: ping <Kali-IP>

```
nmap <Kali-IP>
```



```
Microsoft Windows [Version 10.0.22631.5549]
(c) Microsoft Corporation. All rights reserved.

C:\Users\bhone>ping 192.168.213.128

Pinging 192.168.213.128 with 32 bytes of data:
Reply from 192.168.213.128: bytes=32 time<1ms TTL=64
Reply from 192.168.213.128: bytes=32 time=1ms TTL=64
Reply from 192.168.213.128: bytes=32 time=1ms TTL=64
Reply from 192.168.213.128: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.213.128:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

From Kali you can simulate attack too:

Command: curl <http://example.com>

```
(kali㉿kali)-[~]
└─$ curl http://example.com
<!doctype html>
<html>
<head>
<title>Example Domain</title>
<meta charset="utf-8" />
<meta http-equiv="Content-type" content="text/html; charset=utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<style type="text/css">
body {
    background-color: #f0f0f2;
    margin: 0;
    padding: 0;
    font-family: -apple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "Helvetica Neue", Helvetica, Arial, sans-serif;
}
div {
    width: 600px;
    margin: 5em auto;
    padding: 2em;
    background-color: #fdfdff;
    border-radius: 0.5em;
    box-shadow: 2px 3px 7px 2px rgba(0,0,0,0.02);
}
a:link, a:visited {
    color: #38488f;
    text-decoration: none;
}
@media (max-width: 700px) {
    div {
        margin: 0 auto;
        width: auto;
    }
}
</style>
</head>
<body>
<div>
<h1>Example Domain</h1>
```

Step 7: View Alerts

Alerts will be saved in:

Command: sudo cat /var/log/suricata/fast.log

```
(kali㉿kali)-[~]
└─$ sudo cat /var/log/suricata/fast.log
[sudo] password for kali:
07/05/2025-02:52:21.271709 [**] [1:100001:0] ICMP Packet Detected [**] [Classification: (null)] [Priority: 3] {ICMP} 192.168.213.1:8 → 192.168.213.128:0
07/05/2025-02:52:21.271762 [**] [1:100001:0] ICMP Packet Detected [**] [Classification: (null)] [Priority: 3] {ICMP} 192.168.213.128:0 → 192.168.213.1:0
```

You can also open in tail:

Command: sudo tail -f /var/log/suricata/fast.log

```
(kali㉿kali)-[~]
└─$ sudo tail -f /var/log/suricata/fast.log
07/05/2025-02:52:21.271709 [**] [1:100001:0] ICMP Packet Detected [**] [Classification: (null)] [Priority: 3] {ICMP} 192.168.213.1:8 → 192.168.213.128:0
07/05/2025-02:52:21.271762 [**] [1:100001:0] ICMP Packet Detected [**] [Classification: (null)] [Priority: 3] {ICMP} 192.168.213.128:0 → 192.168.213.1:0
```

Step 8: Make Suricata Run at Boot

If you want Suricata to always start:

Command: sudo systemctl enable suricata

sudo systemctl start suricata

```
(kali㉿kali)-[~]
└─$ sudo systemctl enable suricata
Synchronizing state of suricata.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable suricata
Created symlink '/etc/systemd/system/multi-user.target.wants/suricata.service' → '/usr/lib/systemd/system/suricata.service'.

(kali㉿kali)-[~]
└─$ sudo systemctl start suricata
```

✓ Final Report Structure:

Section	Details
Tool Used	Suricata
Interface Monitored	eth0 (example)
Custom Rule	ICMP detection
Test Performed	ping, nmap, curl
Logs Captured	/var/log/suricata/fast.log
Mitigation Plan	Block offending IPs, firewall integration
Future Plan	Integrate with Kibana, automate alerts

Step 9 : Goal: Block IPs that trigger a Suricata alert.

Step 1: Create an ipset list

```
sudo ipset create blacklist hash:ip
```

Step 2: Add IPTables Rule to Block Blacklisted IPs

Command: sudo iptables -I INPUT -m set --match-set blacklist src -j DROP

Step 3: Create a Response Script

This script adds attacker IPs from Suricata alerts (fast.log) to the blacklist.

Command: sudo nano /usr/local/bin/suricata-ipblocker.sh

Paste this:

```
#!/bin/bash

LOG_FILE="/var/log/suricata/fast.log"

tail -Fn0 "$LOG_FILE" | \
while read line; do
    echo "$line" | grep -oP '\d+\.\d+\.\d+\.\d+' | while read ip; do
        if ! ipset test blacklist $ip &>/dev/null; then
            echo "[+] Blocking IP: $ip"
            ipset add blacklist $ip
        fi
    done
done
```

Step 4: Make Script Executable and Run It

Command: sudo chmod +x /usr/local/bin/suricata-ipblocker.sh

Command: sudo /usr/local/bin/suricata-ipblocker.sh

This script monitors Suricata logs and auto-bans detected attacker IPs.

Custom Rule Set Template

```
# File: /etc/suricata/rules/custom.rules
```

```
alert icmp any any -> any any (msg:"ICMP Ping Detected"; sid:100001; rev:1;)
```

```
alert http any any -> any any (msg:"HTTP Traffic Detected"; sid:100002; rev:1;)  
alert dns any any -> any any (msg:"DNS Request Detected"; sid:100003; rev:1;)
```

1. Tool Used

- Suricata 7.0.10

2. Setup

- Kali Linux with Suricata installed.
- Network interface: eth0
- Rule path: /etc/suricata/rules/

3. Rules Implemented

- ICMP alert rule (`sid:100001`)
- HTTP alert rule (`sid:100002`)
- DNS alert rule (`sid:100003`)

4. Detection Example

- Used `ping` to generate ICMP traffic.
- Alert confirmed in `/var/log/suricata/fast.log` .

5. Response Mechanism

- Auto-blocking IPs that triggered alerts using `ipset` and `iptables` .
- Script used: `/usr/local/bin/suricata-ipblocker.sh`
- Verified that repeated alerts from same IP were blocked.

6. Reflection

- Suricata effectively detected traffic types.
- Adding auto-blocking improves defense but should be reviewed to avoid false positives.