

Suricata IDS Detection

Suricata is running

```
Ubuntu Server [Running] - Oracle VirtualBox
File Machine View Input Devices Help
lab@labserver:~$ sudo suricata -c /etc/suricata/suricata.yaml -i enp0s3
i: suricata: This is Suricata version 7.0.3 RELEASE running in SYSTEM mode
i: threads: Threads created -> W: 1 FM: 1 FR: 1   Engine started.
```

After this, I ran aggressive + os detection + scripts.

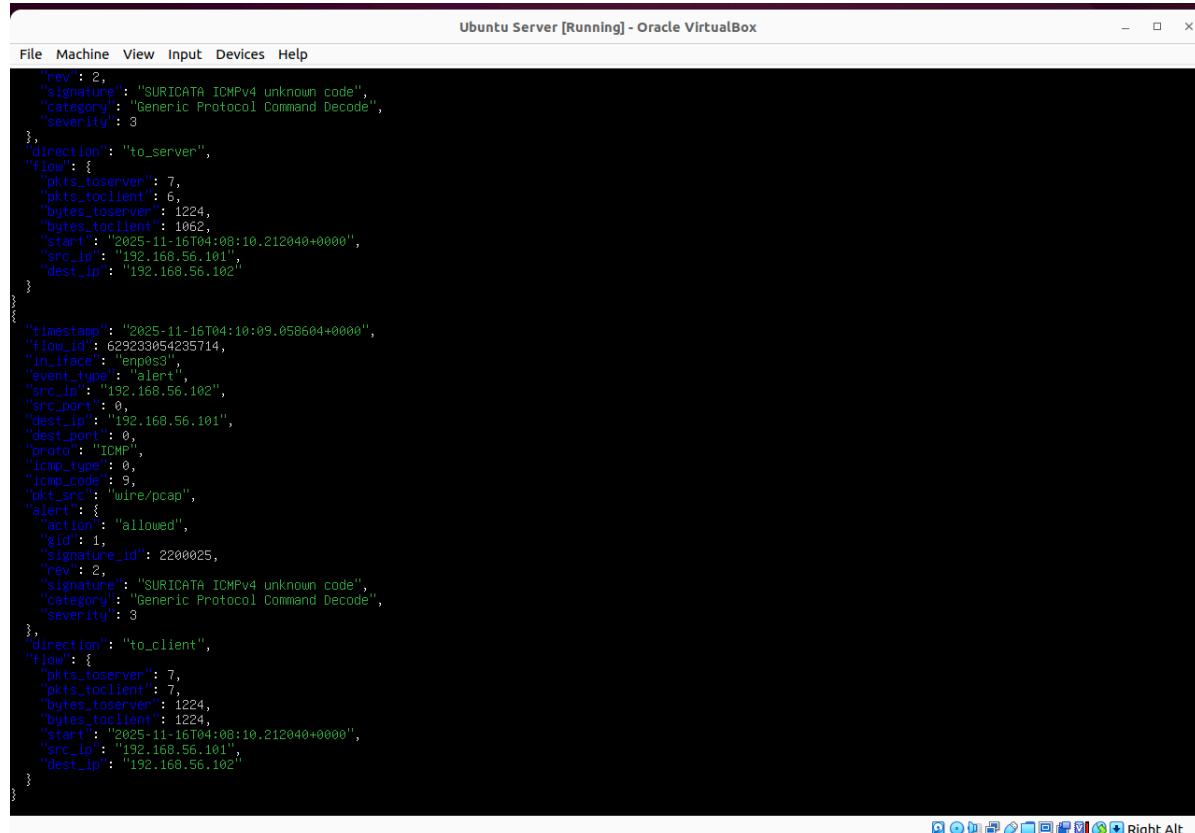
nmap -sS -sV -A 192.168.56.102

```
(kali㉿kali)-[~]
$ nmap -sS -sV -A 192.168.56.102
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-15 23:10 EST
mass_dns: warning: Unable to open /etc/resolv.conf. Try using --system-dns or specify valid servers with --dns-servers: No such file or directory (2)
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.56.102
Host is up (0.00066s latency).
All 1000 scanned ports on 192.168.56.102 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 08:00:27:71:49:A4 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

TRACEROUTE
HOP RTT      ADDRESS
1  0.66 ms  192.168.56.102

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.25 seconds
```

And, I got the result from suricata



```
Ubuntu Server [Running] - Oracle VirtualBox
File Machine View Input Devices Help
{
  "rev": 2,
  "signature": "SURICATA ICMPv4 unknown code",
  "category": "Generic Protocol Command Decode",
  "severity": 3
}
  "direction": "to_server",
  "flow": {
    "pkts_toserver": 7,
    "pkts_toclient": 6,
    "bytes_toserver": 1224,
    "bytes_toclient": 1062,
    "start": "2025-11-16T04:08:10.212040+0000",
    "src_ip": "192.168.56.101",
    "dest_ip": "192.168.56.102"
  }
}

"timestamp": "2025-11-16T04:10:09.058604+0000",
"flow_id": 629233054235714,
"in_iface": "enp0s3",
"event_type": "alert",
"src_ip": "192.168.56.102",
"src_port": 0,
"dest_ip": "192.168.56.101",
"dest_port": 0,
"proto": "ICMP",
"icmp_type": 0,
"icmp_code": 9,
"okr_src": "wire/pcap",
"alert": {
  "action": "allowed",
  "gid": 1,
  "signature_id": 2200025,
  "rev": 2,
  "signature": "SURICATA ICMPv4 unknown code",
  "category": "Generic Protocol Command Decode",
  "severity": 3
}
  "direction": "to_client",
  "flow": {
    "pkts_toserver": 7,
    "pkts_toclient": 7,
    "bytes_toserver": 1224,
    "bytes_toclient": 1224,
    "start": "2025-11-16T04:08:10.212040+0000",
    "src_ip": "192.168.56.101",
    "dest_ip": "192.168.56.102"
}
}
```

which told me that

event_type: "alert"

src_ip: "192.168.56.101" — Kali (attacker)

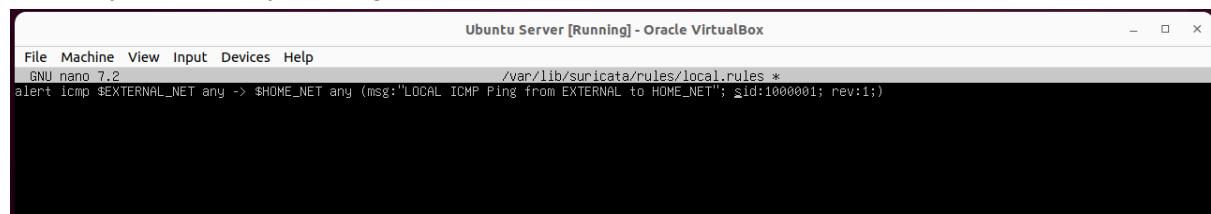
dest_ip: "192.168.56.102" — Ubuntu server (victim)

proto: "ICMP" — Nmap sending ICMP packets.

Signature(attack type/name): "SURICATA ICMPv4 unknown code" — suricata detecting unusual ICMP behavior

Next, I wanted to try a custom local rule setup. So, I wrote my own suricata rule and trigger it on purpose.

I built my own rule by naming local.rules.



```
Ubuntu Server [Running] - Oracle VirtualBox
File Machine View Input Devices Help
GNU nano 7.2
/var/lib/suricata/rules/local.rules *
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"LOCAL ICMP Ping from EXTERNAL to HOME_NET"; $id:1000001; rev:1;)
```

It's a simple rule that when Kali pings Ubuntu server, it fires an alert.

I moved the local.rules to /var/lib/suricata/rules from /etc/suricata/rules to make it work.

Then, added to the configuration

Ubuntu Server [Running] - Oracle VirtualBox

```
File Machine View Input Devices Help
GNU nano 7.2
# /etc/suricata/suricata.yaml *

##

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

rule-files:
  - suricata.rules
  - local.rules

##
## Auxiliary configuration files.
##

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

##
## Include other configs
##

# Includes: Files included here will be handled as if they were in-lined
# in this configuration file. Files with relative pathnames will be
# searched for in the same directory as this configuration file. You may
# use absolute pathnames too.
#include:
#  - include1.yaml
#  - include2.yaml
```

Then, I tested the config.

```
lab@labserver:~$ sudo suricata -T -c /etc/suricata/suricata.yaml -v
Notice: suricata: This is Suricata version 7.0.3 RELEASE running in SYSTEM mode
Info: cpu: CPUs/cores online: 1
Info: suricata: Running suricata under test mode
Info: suricata: Setting engine mode to IDS mode by default
Info: exception-policy: master exception-policy set to: auto
Info: logopenfile: fast output device (regular) initialized: fast.log
Info: logopenfile: eve-log output device (regular) initialized: eve.json
Info: logopenfile: stats output device (regular) initialized: stats.log
Info: detect: 2 rule files processed, 46344 rules successfully loaded, 0 rules failed, 0
Info: threshold-config: Threshold config parsed: 0 rule(s) found
Info: detect: 46347 signatures processed. 971 are IP-only rules, 4422 are inspecting packet payload, 40723 inspect application layer, 108 are decoder event only
Notice: suricata: Configuration provided was successfully loaded. Exiting.
lab@labserver:~$
```

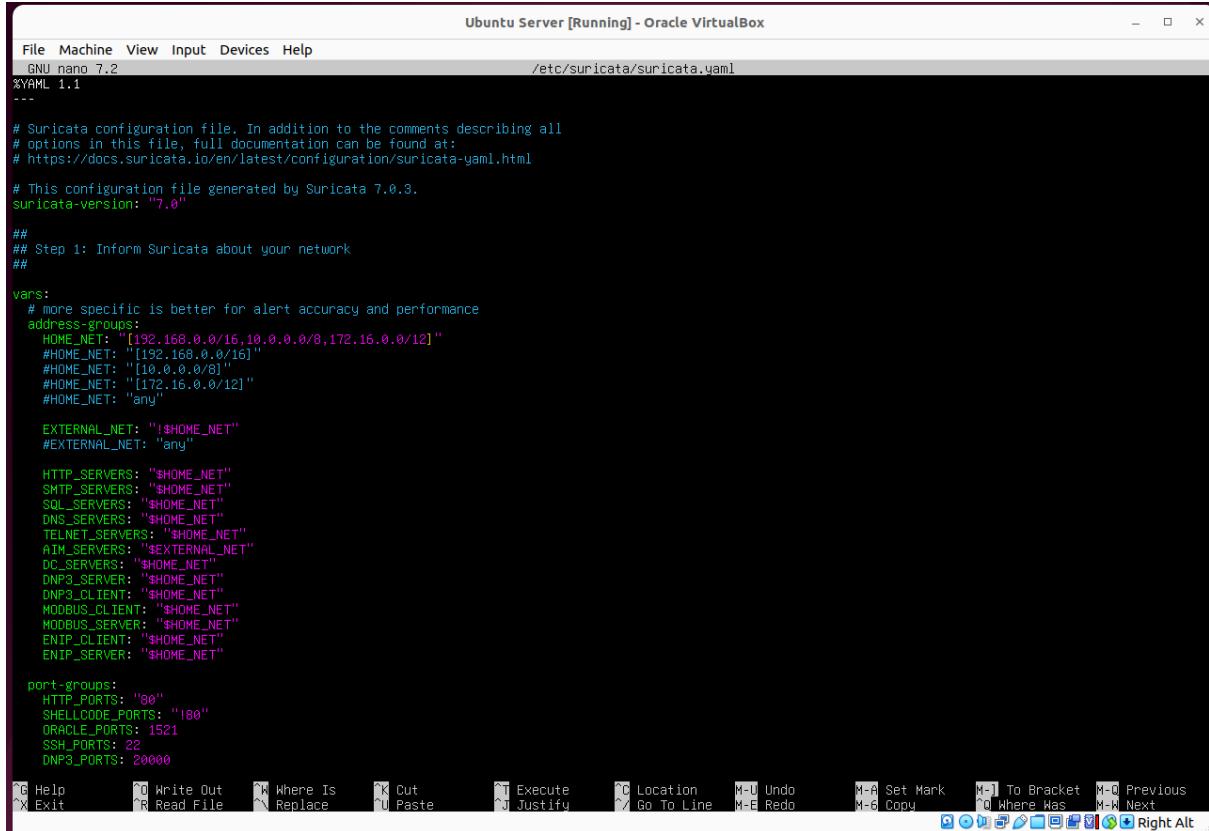
From Kali, I pinged the suricata server,
ping -c 4 192.168.56.102

```
(kali㉿kali)-[~]
$ ping -c 4 192.168.56.102
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.
64 bytes from 192.168.56.102: icmp_seq=1 ttl=64 time=0.558 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=0.543 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.509 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=0.522 ms

--- 192.168.56.102 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3073ms
rtt min/avg/max/mdev = 0.509/0.533/0.558/0.018 ms

(kali㉿kali)-[~]
$
```

However, suricata didn't get any alerts of pinging, so I checked the configuration file. In the file, under address-group,



```
Ubuntu Server [Running] - Oracle VirtualBox
File Machine View Input Devices Help
GNU nano 7.2
/etc/suricata/suricata.yaml
---  

# Suricata configuration file. In addition to the comments describing all  

# options in this file, full documentation can be found at:  

# https://docs.suricata.io/en/latest/configuration/suricata-yaml.html  

# This configuration file generated by Suricata 7.0.3.  

suricata-version: "7.0"  

##  

## Step 1: Inform Suricata about your network  

##  

vars:  

# more specific is better for alert accuracy and performance  

address-groups:  

  HOME_NET: "[192.168.0.0/16,10.0.0.0/8,172.16.0.0/12]"  

  #HOME_NET: "[192.168.0.0/16]"  

  #HOME_NET: "[10.0.0.0/8]"  

  #HOME_NET: "[172.16.0.0/12]"  

  #HOME_NET: "any"  

  EXTERNAL_NET: "[$HOME_NET]"  

  #EXTERNAL_NET: "any"  

  HTTP_SERVERS: "$HOME_NET"  

  SMTP_SERVERS: "$HOME_NET"  

  SQL_SERVERS: "$HOME_NET"  

  DNS_SERVERS: "$HOME_NET"  

  TELNET_SERVERS: "$HOME_NET"  

  AIM_SERVERS: "$EXTERNAL_NET"  

  DC_SERVERS: "$HOME_NET"  

  DNP3_SERVER: "$HOME_NET"  

  DNP3_CLIENT: "$HOME_NET"  

  MODBUS_CLIENT: "$HOME_NET"  

  MODBUS_SERVER: "$HOME_NET"  

  ENIP_CLIENT: "$HOME_NET"  

  ENIP_SERVER: "$HOME_NET"  

port-groups:  

  HTTP_PORTS: "80"  

  SHELLCODE_PORTS: "!80"  

  ORACLE_PORTS: 1521  

  SSH_PORTS: 22  

  DNP3_PORTS: 20000  

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark M-J To Bracket M-Q Previous  

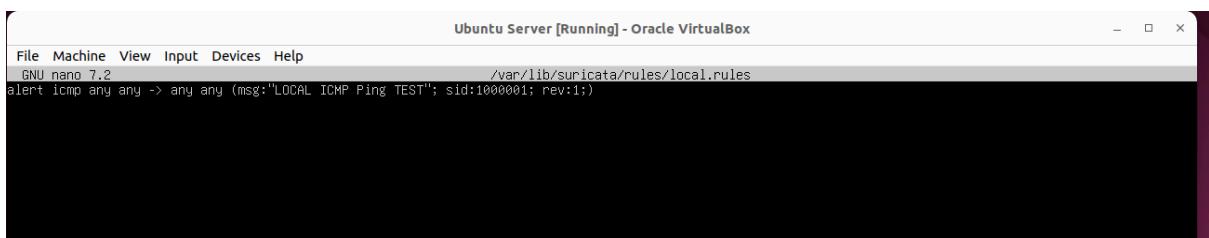
^X Exit ^R Read File ^U Replace ^P Paste ^J Justify ^V Go To Line M-E Redo M-G Copy ^D Where Was M-W Next ^I Right Alt
```

HOME_NET covered all 192.168.x.x/16, and EXTERNAL_NET covered anything not in HOME_NET. This part showed the address definitions were the cause of my rule not firing. After I checked this, I realized Kali and Ubuntu server both are in HOME_NET, while my rule only matched EXTERNAL_NET → HOME_NET ICMP. So, my ICMP ping did not fall into the category, and the rule never triggered.

I evaluated three correct solutions for this problem.

1. Simplifying the rule by changing to any
2. Redefine HOME_NET and EXTERNAL_NET.
HOME_NET: 192.168.56.102(Ubuntu), EXTERNAL_NET: 192.168.56.101(Kali)
3. Restrict HOME_NET to the Lab Subnet
Before: 192.168.0.0/16 —> After: 192.168.56.0/24

I chose the first one.



```
Ubuntu Server [Running] - Oracle VirtualBox
File Machine View Input Devices Help
GNU nano 7.2
/var/lib/suricata/rules/local.rules
alert icmp any any -> any any (msg:"LOCAL ICMP Ping TEST"; sid:1000001; rev:1);
```

Then I pinged four times, finally suricata showed the alert.

Suricata

```
Ubuntu Server [Running] - Oracle VirtualBox
File Machine View Input Devices Help
lab@labserver:~$ sudo jq 'select(.event_type=="alert")' /var/log/suricata/eve.json | tail -n 20
{
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 1000001,
    "rev": 1,
    "signature": "LOCAL ICMP Ping TEST",
    "category": "",
    "severity": 3
  },
  "direction": "to_client",
  "flow": {
    "pkts_toserver": 1,
    "pkts_toclient": 1,
    "bytes_toserver": 98,
    "bytes_toclient": 98,
    "start": "2025-11-16T05:27:15.405663+0000",
    "src_ip": "192.168.56.101",
    "dest_ip": "192.168.56.102"
  }
}
lab@labserver:~$
```

After restarting Suricata and pinging again, the alert triggered successfully and appeared in eve.json. This confirmed that Suricata detected the custom rule.

I successfully completed IDS detection using both built-in Suricata signatures and a custom ICMP rule. I also identified and resolved an issue related to Suricata's HOME_NET and EXTERNAL_NET variables, demonstrating full understanding of Suricata's rule-matching logic and rule writing.