Task 3 : Network Intrusion Detection System

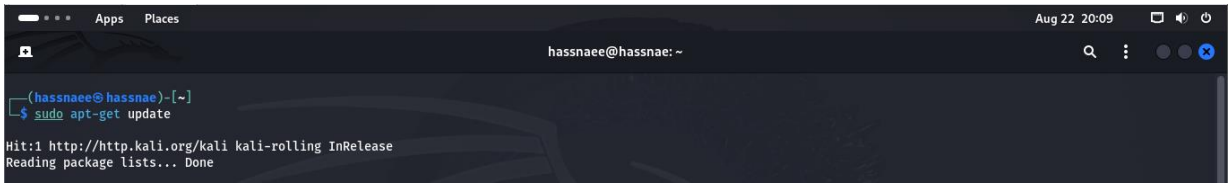
1. Introduction:

This report describes in detail the process of installation, configuration and deployment of Suricata, an open-source Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) We will go through each step, from running commands to changing settings, posting screenshots and suggested locations.

Installation of Suricata:

The firste step is to update and installe the suricate using this command:

- Sudo apt-get update
- Sudo apt-get install suricate



2. Updating the Emerging Threats Open Ruleset

To update the ruleset, run the following command:

- Sudo suricata-update

```
┌──(hassnaee㉿hassnae)-[~]
└─$ sudo systemctl enable suricata.service
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.
```

This command fetches and installs the latest version of the ruleset into the default directory (/var/lib/Suricata/rules/).

Now that the package is installed, enable the Suricata.Service so that it will run when our system restarts. We can use the systemctl command to enable it:

```
┌──(hassnaee㉿hassnae)-[~]
└─$ sudo systemctl stop suricata.service
```

3. Configuration of Suricata

Suricata can include a Community ID field in its JSON output to make it easier to match individual event records to records in datasets generated by other tools.

To enable the option, we open /etc/suricata/suricata.yaml using nano editor:

```
┌──(hassnaee㉿hassnae)-[~]
└─$ sudo nano /etc/suricata/suricata.yaml
```

Determining Which Network Interface(s) To Use:

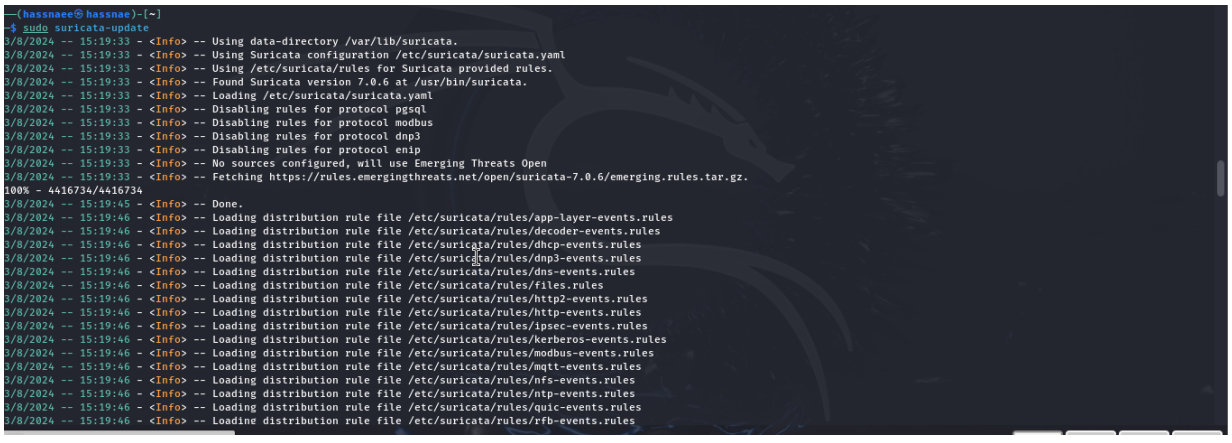
To determine the device name of our default network interface, we can use the ip command as follows:

- ip -p -j route show default

The dev line indicates the default device. In this example output, the device is the highlighted eth0 interface.
Now we can edit Suricata's configuration and verify or change the interface name.
We Open the /etc/suricata/suricata.yaml configuration file using nano or your preferred editor:

A command like the following will notify the Suricata process to reload its rulesets, without restarting the process:

4. Updating Suricata Rulesets

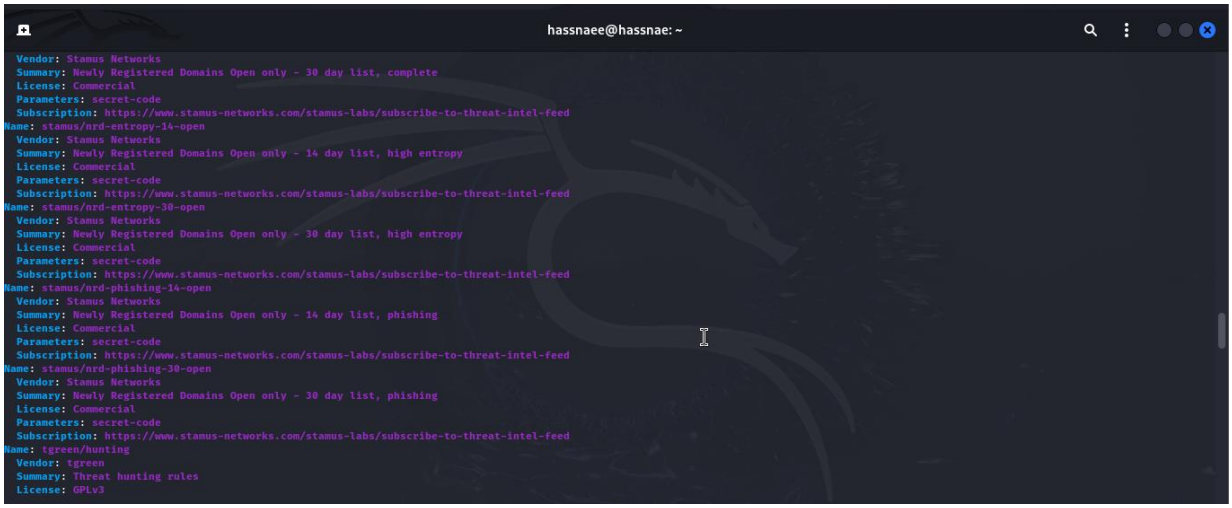


We can list the default set of rule providers using the list-sources flag to suricata-update like this:
Using the following commande:

- Sudo suricata-update list-sources

```
23/8/2024 -- 15:19:56 - <Info> -- Dropped 0 rules.
23/8/2024 -- 15:19:56 - <Info> -- Enabled 136 rules for flowbit dependencies.
23/8/2024 -- 15:19:56 - <Info> -- Backing up current rules.
23/8/2024 -- 15:19:57 - <Info> -- Writing rules to /var/lib/suricata/rules/suricata.rules: total: 52097; enabled: 39663; added: 52097; removed 0; modified: 0
23/8/2024 -- 15:19:57 - <Info> -- Writing /var/lib/suricata/rules/classification.config
23/8/2024 -- 15:19:59 - <Info> -- Testing with suricata -T.
23/8/2024 -- 15:21:13 - <Info> -- Done.

┌──(hassnaee㉿hassnae)-[~]
└─$ sudo suricata-update list-sources
23/8/2024 -- 15:21:35 - <Info> -- Using data-directory /var/lib/suricata.
23/8/2024 -- 15:21:35 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
23/8/2024 -- 15:21:35 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
23/8/2024 -- 15:21:35 - <Info> -- Found Suricata version 7.0.6 at /usr/bin/suricata.
23/8/2024 -- 15:21:35 - <Warning> -- Source index does not exist, will use bundled one.
23/8/2024 -- 15:21:35 - <Warning> -- Please run suricata-update update-sources.
Name: et/open
  Vendor: Proofpoint
  Summary: Emerging Threats Open Ruleset
  License: MIT
Name: et/pro
  Vendor: Proofpoint
  Summary: Emerging Threats Pro Ruleset
  License: Commercial
  Replaces: et/open
  Parameters: secret-code
  Subscription: https://www.proofpoint.com/us/threat-insight/et-pro-ruleset
Name: etnetera/aggressive
  Vendor: Etnetera a.s.
  Summary: Etnetera aggressive IP blacklist
  License: MIT
Name: malsilo/win-malware
  Vendor: malsilo
```



to include the tgreen/hunting ruleset, we could enable it using the following command:

- sudo suricata-update enable-source tgreen/hunting

```
┌──(hassnaee㉿hassnae)-[~]
└─$ sudo suricata-update enable-source tgreen/hunting
23/8/2024 -- 15:21:52 - <Info> -- Using data-directory /var/lib/suricata.
23/8/2024 -- 15:21:52 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
23/8/2024 -- 15:21:52 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
23/8/2024 -- 15:21:52 - <Info> -- Found Suricata version 7.0.6 at /usr/bin/suricata.
23/8/2024 -- 15:21:52 - <Warning> -- Source index does not exist, will use bundled one.
23/8/2024 -- 15:21:52 - <Warning> -- Please run suricata-update update-sources.
23/8/2024 -- 15:21:52 - <Info> -- Creating directory /var/lib/suricata/update/sources
23/8/2024 -- 15:21:52 - <Info> -- Enabling default source et/open
23/8/2024 -- 15:21:52 - <Info> -- Source tgreen/hunting enabled

┌──(hassnaee㉿hassnae)-[~]
└─$ sudo suricata -T -c /etc/suricata/suricata.yaml -v
Notice: suricata: This is Suricata version 7.0.6 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: 1 rule files processed. 39663 rules successfully loaded, 0 rules failed, 0
Info: threshold-config: Threshold config parsed: 0 rule(s) found
Info: detect: 39666 signatures processed. 1169 are IP-only rules, 4112 are inspecting packet payload, 34176 inspect application layer, 108 are decoder event only
Notice: suricata: Configuration provided was successfully loaded. Exiting.
```

5. Running Suricata

Start the Suricata Service:

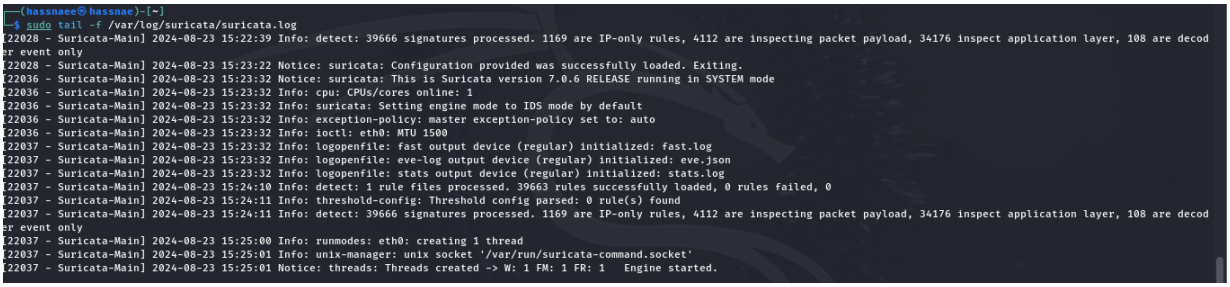
Now we can start the Suricata server. Run the following systemctl command:

- sudo systemctl start suricata-service

we can examine the status of the service using the systemctl status command:

- sudo systemctl status suricata. service

we can use the tail command to watch for a specific message in Suricata's logs that indicates it has finished starting:



6. Testing Suricata Rules

Execute a Test Request

To test Suricata's ability to detect suspicious traffic, we use the curl command to send an HTTP request that triggers a predefined rule:

Examining /var/log/suricata/fast.log

To check for a log entry in /var/log/suricata/fast.log that corresponds to our curl request we use the grep command.
Using the 2100498-rule identifier from the QuickStart documentation, search for entries that match it using the following command:

- Sudo grep 2100498 /var/log/suricata/fast/log

Examining /var/log/suricata/eve.log

Suricata also logs events to /var/log/suricata/eve.log (nicknamed the EVE log) using JSON to format entries.
The Suricata documentation recommends using the jq utility to read and filter the entries in this file.
We Install jq if we don't have it on our system using the following apt command:

- Sudo apt install jq

```
┌──(hassnaee㉿hassnae)-[~]
└─$ sudo apt install jq
jq is already the newest version (1.7.1-3).
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1065
```

Then, run:

```
┌──(hassnaee㉿hassnae)-[~]
└─$ sudo jq 'select(.alert .signature_id==2100498)' /var/log/suricata/eve.json
{
  "timestamp": "2024-08-23T15:55:40.133189+0200",
  "flow_id": 919486000414787,
  "in_iface": "eth0",
  "event_type": "alert",
  "src_ip": "18.154.22.58",
  "src_port": 80,
  "dest_ip": "192.168.1.108",
  "dest_port": 34108,
  "proto": "TCP",
  "pkt_src": "wire/pcap",
  "community_id": "1:mQGQi8eQmJYVOYULzRVslBAmPr8=",
  "tx_id": 0,
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 2100498,
    "rev": 7,
    "signature": "GPL ATTACK_RESPONSE id check returned root",
    "category": "Potentially Bad Traffic",
    "severity": 2,
    "metadata": {
      "created_at": [
        "2010_09_23"
      ],
      "updated_at": [
        "2019_07_26"
      ]
    }
  },
  "http": {
    "hostname": "testmynids.org",
```

```
hassnaee@hassnae: ~
      "2019_07_26"
    ]
  }
},
"http": {
  "hostname": "testmynids.org",
  "url": "/uid/index.html",
  "http_user_agent": "curl/8.7.1",
  "http_content_type": "text/html",
  "http_method": "GET",
  "protocol": "HTTP/1.1",
  "status": 200,
  "length": 39
},
"files": [
  {
    "filename": "/uid/index.html",
    "gaps": false,
    "state": "CLOSED",
    "stored": false,
    "size": 39,
    "tx_id": 0
  }
],
"app_proto": "http",
"direction": "to_client",
"flow": {
  "pkts_toserver": 5,
  "pkts_toclient": 4,
  "bytes_toserver": 429,
  "bytes_toclient": 810,
  "start": "2024-08-23T15:55:39.934980+0200",
  "src_ip": "192.168.1.108",
  "dest_ip": "18.154.22.58",
```

```
  "http_content_type": "text/html",
  "http_method": "GET",
  "protocol": "HTTP/1.1",
  "status": 200,
  "length": 39
},
"files": [
  {
    "filename": "/uid/index.html",
    "gaps": false,
    "state": "CLOSED",
    "stored": false,
    "size": 39,
    "tx_id": 0
  }
],
"app_proto": "http",
"direction": "to_client",
"flow": {
  "pkts_toserver": 5,
  "pkts_toclient": 4,
  "bytes_toserver": 429,
  "bytes_toclient": 810,
  "start": "2024-08-23T15:55:39.934980+0200",
  "src_ip": "192.168.1.108",
  "dest_ip": "18.154.22.58",
  "src_port": 34108,
  "dest_port": 80
}
}
```

This output confirms that Suricata detected the traffic and generated an alert based on the specified rule.