TASK – 1 Basic Network Sniffer

What is a Network Packet Sniffer?

A network packet sniffer is a software tool that intercepts and logs network traffic passing through a specific network interface. It captures packets at the data link layer and can analyze various protocol headers to extract information such as sources and destination IP addresses, ports, and packet payloads. Packet sniffers are commonly used by network administrators, security professionals and developers for tasks such as network troubleshooting, performance monitoring, and security analysis.

Getting Started with Python and Scapy

Scapy is a powerful Python library for packet manipulation and network analysis. It provides a high-level interface for crafting and dissecting network packets, making it an ideal choice for building network utilities such as packet sniffers. Before we begin, ensure that you have Python and Scapy installed on your system. You can install Scapy using pip:

pip install scapy

Developing the Packet Sniffer

We'll start by writing a Python script that captures TCP packets on a specified network interface and logs relevant information to a file.

Conclusion

In this guide, we've developed a simple network packet sniffer using Python and Scapy. While the example focuses on capturing TCP packets, you can extend the functionality to support other protocols and perform more advanced analysis. Packet sniffers are powerful tools for understanding network behavior and diagnosing network issues, making them invaluable in various networking scenarion.

TASK – 2 Phishing Awareness Training

What is Phishing?

Phishing is a cyberattack where attackers trick people into giving away sensitive information such as:

✅ **Usernames & Passwords**
✅ **Credit Card Details**
✅ **Social Security Numbers**
✅ **Bank Account Information**

Attackers disguise themselves as trusted entities (banks, social media, government agencies) to steal information or install malware.

Types of Phishing Attacks

🔹 **Email Phishing** – Fake emails impersonating trusted entities
🔹 **Spear Phishing** – Targeted attacks on individuals or companies
🔹 **Smishing (SMS Phishing)** – Phishing via text messages
🔹 **Vishing (Voice Phishing)** – Phone-based scams
🔹 **Website Spoofing** – Fake websites stealing credentials

How to Identify Phishing Attempts?

🔍 Suspicious sender email addresses
🔍 Generic greetings (“Dear Customer” instead of your name)
🔍 Urgent requests for personal information
🔍 Poor grammar and spelling mistakes
🔍 Mismatched or shortened URLs
🔍 Unexpected attachments or links

Best Practices to Stay Safe

✅ **Verify Links** – Hover over links before clicking
✅ **Check Email Headers** – Look for mismatched sender details
✅ **Enable Multi-Factor Authentication (MFA)**
✅ **Never Share Sensitive Information** via email or phone
✅ **Use Security Software & Updates**
✅ **Report Phishing** – To IT, Google Safe Browsing, or Anti-Phishing organizations

TASK – 3 Network Intrusion Detection System

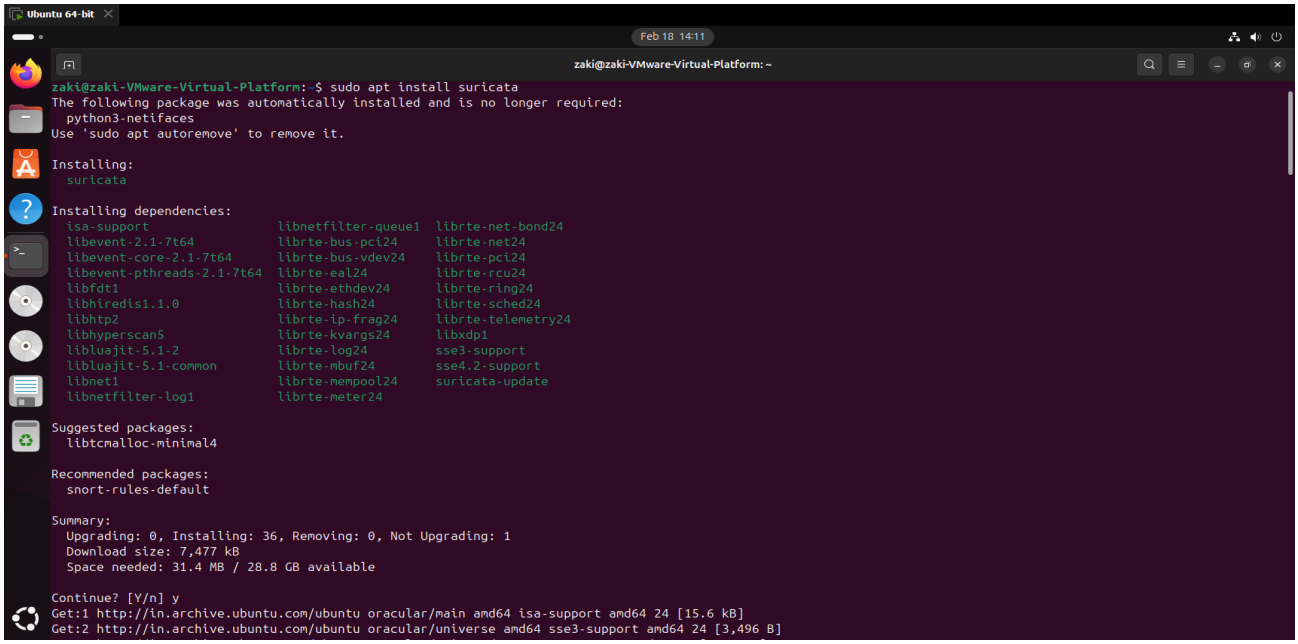
What is Suricata?

Suricata is an open-source, high-performance, and multi-threaded network intrusion detection system (NIDS), intrusion prevention system (IPS), and network security monitoring (NSM) tool. It analyzes network traffic in real-time to detect and prevent cyber threats.

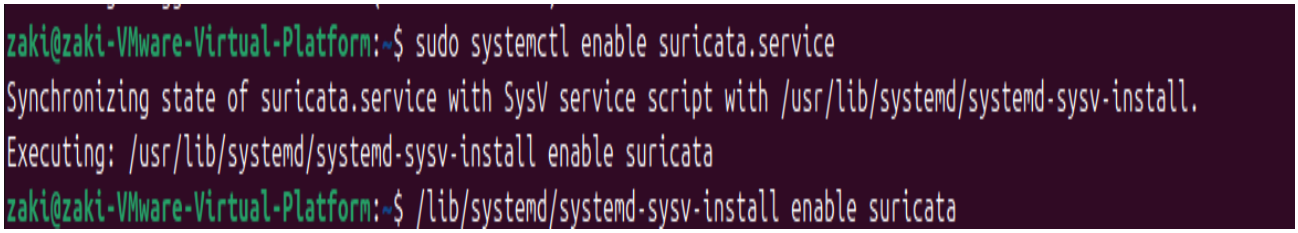
Suricata is an advanced open-source network threat detection engine that provides Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) functions. It enables organizations to monitor network traffic in real-time, identify potential security threats, and respond effectively. In this guide, we will walk you through the installation and initial configuration of Suricata on an Ubuntu system.

Step 1 — Installing Suricata

Install Suricata: Use the following command to install the Suricata package:



Enable the Suricata Service: To ensure Suricata runs on startup, enable the service using:



Stop the Suricata Service: Before editing the configuration file, stop the service:

Step 2 — Configuring Suricata For The First Time

Open the Configuration File: Use your preferred text editor to open the Suricata configuration file:

```
zaki@zaki-VMware-Virtual-Platform:~$ sudo nano /etc/suricata/suricata.yaml
```

Step 3 — Updating Suricata Rulesets

Run Suricata Update: Fetch the latest ruleset for your Suricata server with:

```
zaki@zaki-VMware-Virtual-Platform:~$ sudo suricata-update
18/2/2025 -- 14:15:29 - <Info> -- Using data-directory /var/lib/suricata.
18/2/2025 -- 14:15:29 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
18/2/2025 -- 14:15:29 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
18/2/2025 -- 14:15:29 - <Info> -- Found Suricata version 7.0.6 at /usr/bin/suricata.
18/2/2025 -- 14:15:29 - <Info> -- Loading /etc/suricata/suricata.yaml
18/2/2025 -- 14:15:29 - <Info> -- Disabling rules for protocol pgsql
18/2/2025 -- 14:15:29 - <Info> -- Disabling rules for protocol modbus
18/2/2025 -- 14:15:29 - <Info> -- Disabling rules for protocol dnp3
18/2/2025 -- 14:15:29 - <Info> -- Disabling rules for protocol enip
18/2/2025 -- 14:15:29 - <Info> -- No sources configured, will use Emerging Threats Open
18/2/2025 -- 14:15:29 - <Info> -- Fetching https://rules.emergingthreats.net/open/suricata-7.0.6/emerging.rules.tar.gz.
 100% - 4764792/4764792
```

Adding Ruleset Providers: To list available rule sources, use: Sudo suricata-update list-sources

```
zaki@zaki-VMware-Virtual-Platform:~$ sudo suricata-update list-sources
18/2/2025 -- 14:16:02 - <Info> -- Using data-directory /var/lib/suricata.
18/2/2025 -- 14:16:02 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
18/2/2025 -- 14:16:02 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
18/2/2025 -- 14:16:02 - <Info> -- Found Suricata version 7.0.6 at /usr/bin/suricata.
18/2/2025 -- 14:16:02 - <Warning> -- Source index does not exist, will use bundled one.
18/2/2025 -- 14:16:02 - <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
  Summary: Commodity malware rules
  License: MIT
```

To enable additional rulesets, for example, the tgreen/hunting ruleset, run:
sudo suricata-update enable-source tgreen/hunting

```
zaki@zaki-VMware-Virtual-Platform:~$ sudo suricata-update enable-source tgreen/hunting
18/2/2025 -- 14:16:20 - <Info> -- Using data-directory /var/lib/suricata.
18/2/2025 -- 14:16:20 - <Info> -- Using Suricata configuration /etc/suricata/suricata.yaml
18/2/2025 -- 14:16:20 - <Info> -- Using /etc/suricata/rules for Suricata provided rules.
18/2/2025 -- 14:16:20 - <Info> -- Found Suricata version 7.0.6 at /usr/bin/suricata.
18/2/2025 -- 14:16:20 - <Warning> -- Source index does not exist, will use bundled one.
18/2/2025 -- 14:16:20 - <Warning> -- Please run suricata-update update-sources.
18/2/2025 -- 14:16:20 - <Info> -- Creating directory /var/lib/suricata/update/sources
18/2/2025 -- 14:16:20 - <Info> -- Enabling default source et/open
18/2/2025 -- 14:16:20 - <Info> -- Source tgreen/hunting enabled
```

Step 4 — Validating Suricata's Configuration

Test the Configuration: Validate your configuration changes by running:

This command will check the validity of the configuration and any included rules.

After completing these steps, you will have Suricata installed and configured on your Ubuntu system. You can now proceed to start the Suricata service and monitor your network traffic. If you have any further questions or need assistance with specific configurations, feel free to ask!