# Cybersecurity Lab: Threat Detection

**Overview**

The lab consists of three **Kali Linux virtual machines (VMs)** running on **VMware Workstation**, simulating a **SOC Analyst system, a Client/Victim system, and an Attacker system**.

**Features**

* **Threat Simulation**: Use tools like Metasploit, Nmap, and hping3 to generate attack traffic.
* **Network Monitoring**: Utilize Suricata, snort and Wireshark for packet inspection.
* **Log Aggregation**: Configure Splunk or ELK to collect and visualize security logs.

The lab is designed for threat detection and log analysis

**Virtual Machine Architecture**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | VM Name | OS | IP Address | Purpose | | SOC Analyst | Kali Linux | 192.168.22.137/24 | Analyses logs, trains AI model | | Client/Victim | Kali Linux | 192.168.22.136/24 | Runs services (Web, SSH, FTP), captures logs | | Hacker/Attacker | Kali Linux | 192.168.22.134/24 | Generates attacks (Nmap, Metasploit, DoS) | |

**SOC Analyst VM (192.168.22.137)**

**Installed Splunk**

wget -O splunk.deb 'https://download.splunk.com/products/splunk/releases/latest/linux/splunk-latest-linux-2.6-amd64.deb'

sudo dpkg -i splunk.deb

sudo /opt/splunk/bin/splunk enable boot-start

sudo /opt/splunk/bin/splunk start

**Installed Suricata**

sudo apt update && sudo apt install -y suricata

sudo systemctl enable --now suricata

**Installed Python for AI-based Threat Detection**

sudo apt install -y python3-pip

pip3 install scikit-learn pandas numpy matplotlib joblib scapy

**Configured Splunk to Receive Logs**

sudo /opt/splunk/bin/splunk add udp 514 -sourcetype syslog

sudo /opt/splunk/bin/splunk restart

**Client/Victim VM (192.168.22.136)**

**Installed and Configured Suricata**

sudo apt update && sudo apt install -y suricata

sudo systemctl enable --now suricata

**Configured Log Forwarding to SOC VM**

Edited /etc/rsyslog.conf and add:

\*.\* @192.168.22.137:514

Restarted rsyslog:

sudo systemctl restart rsyslog

**Install Web, SSH, and FTP Services**

sudo apt install -y apache2 openssh-server vsftpd

sudo systemctl enable --now apache2 ssh vsftpd

**Attacker VM (192.168.22.134)**

**Install Nmap, Metasploit, and hping3**

sudo apt install -y nmap metasploit-framework hping3

**Simulating Attacks**

**Port Scanning with Nmap**

nmap -sS -p- 192.168.22.136

**Brute Force SSH with Metasploit**

msfconsole -q

use auxiliary/scanner/ssh/ssh\_login

set RHOSTS 192.168.22.136

set USERPASS\_FILE /usr/share/wordlists/rockyou.txt

run

**DoS Attack with hping3**

hping3 -S --flood -p 80 192.168.22.136

**Threat Simulation and AI-based Analysis**

1. **Simulate Attacks** from the Attacker VM to the Client VM.
2. **Capture Logs** using Suricata and forward them to the SOC VM.
3. **Analyze Logs** in Splunk or ELK.
4. **Train AI Model** using Python:

from sklearn.ensemble import IsolationForest

import pandas as pd

# Load log data

df = pd.read\_csv('logs.csv')

# Train AI model

model = IsolationForest(n\_estimators=100).fit(df)

# Save model

import joblib

joblib.dump(model, 'ai\_model.pkl')

**Tools and Technologies**

* Security Monitoring: Suricata, Snort, Wireshark
* Threat Simulation: Metasploit, Nmap, hping3
* Log Aggregation: Splunk Free / ELK Stack
* AI Model Training: Python (Scikit-learn, Pandas, NumPy, Matplotlib, Scapy)

**Attack Scenarios**

* Port Scanning with Nmap
* Brute-force SSH Attack with Metasploit
* Denial-of-Service (DoS) using hping3