**Intrusion Detection and Log Analysis Using Suricata and the ELK Stack**

**Project Report**

Submitted by

**Mohammed Ramees T**

rameesthamarath@gmail.com

**RedTeam Hacker Academy**

**CICSA (Certified IT Infrastructure and Cyber SOC Analyst)**

**In Cyber Security**

Under the Guidance of

**Mr. Gopu Nair**

A red text on a black background

AI-generated content may be incorrect.

**Table of Contents**

1 Introduction

 1.1 Background . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

 1.2 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

 1.3 Project Overview . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

 1.4 Objectives . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .2

2 Environment and Tools

 2.1 System Environment . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . . . 3

 2.2 Tools and Technologies Used . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3

  2.2.1 Suricata – Intrusion Detection System (IDS) . . . . . . . . . . . . . . . .. . . . . . 4

  2.2.2 Working of Suricata . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .4

  2.2.3 ELK stack (Elasticsearch, Logstash, and Kibana) . . . . . . . . . . . . .. . . . .. 5

3 Implementation

 3.1 Install Suricata . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. 7

  3.1.1 Create Suricata Rules . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. 8

  3.1.2 Configure Suricata . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

 3.2 Install Elasticsearch . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

 3.3 Install Logstash . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . . 10

  3.3.3 Logstash Pipeline Configuration . . . . . . . . . . . . . . . . . . . . . . . .. . .. . . . . 11

 3.4 Install Kibana . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. .. . . . . 11

 3.5 Integration Steps . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12

 3.6 Install OpenSSH Server (if not already installed) . . . . . . . . . . . . . . . . . . . . 12

 3.7 Attacks Performed . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . . 12

 3.8 Detection . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . 14

 3.9 Kibana Log Overview . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15

4 Results and Observations

 4.1 Project Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17

 4.2 Conclusion and Future Scope . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . 17

1

**1. Introduction**

**1.1 Background**

In today’s cybersecurity landscape, organizations face increasing threats such as unauthorized access, brute-force login attempts, and data exfiltration. These attacks often go undetected without proper monitoring and analysis tools in place. Intrusion Detection Systems (IDS) like Suricata offer real-time detection of such malicious activities by inspecting network traffic against predefined rule sets.

However, raw alert data is difficult to manage at scale. That’s where the ELK Stack (Elasticsearch, Logstash, Kibana) plays a crucial role — by collecting, indexing, and visualizing logs in a human-readable format. Together, these tools provide an efficient and scalable Network Security Monitoring (NSM) solution.

**1.2 Motivation**

The primary motivation for this project is to gain hands-on experience with industry-standard cybersecurity tools. As part of my learning in the Certified Information Cyber Security Analyst (CICSA) course, I aimed to:

* Understand how real-time traffic inspection works.
* Detect attacks like brute force login attempts.
* Visualize threats for better incident response.

The goal is to replicate the functionality of a Security Operations Center (SOC) environment using open-source tools and simulated attacks.

**1.3 Objectives**

* Install and configure Suricata IDS on a Ubuntu system to monitor incoming network traffic.
* Simulate an SSH brute force attack using Hydra to generate real-world detection scenarios.
* Create a custom Suricata rule to trigger alerts based on repeated failed SSH login attempts.
* Use Filebeat and Logstash to collect and process Suricata-generated logs.
* Visualize detected attacks in Kibana using real-time dashboards.
* Gain hands-on experience with ELK Stack as a log management and analysis platform.
* Understand the workflow of a SOC in detecting and analysing security incidents.

2

**2. Environment and Tools**

**2.1 Environment Setup**

Host OS: Windows 11

• Virtualization Platform: VirtualBox 7.1.6

• Guest Virtual Machines:

– Kali Linux (Attacker Machine)

– Ubuntu 22.04 (Suricata and ELK Stack)

**2.2 Tools Used**

|  |  |
| --- | --- |
| **Tool** | **Description** |
| * Suricata | Network IDS/IPS |
| * Elasticsearch | Log storage and search engine |
| * Logstash | Data collection and transformation tool |
| * Kibana | Data visualization and a dashboard tool |

3

**2.2.1 Suricata**



Suricata is a powerful, open-source network threat detection engine that functions as both an Intrusion Detection System (IDS) and an Intrusion Prevention System (IPS). Suricata stands as a high-performance, open-source network analysis, and threat detection software widely adopted by both private and public organizations.

In simple terms, Suricata is like a digital security guard for computer networks. It carefully watches the traffic moving through the network, looking for any signs of suspicious or harmful activity. It helps keep the network safe by identifying and alerting to potential threats, making it a valuable tool for organizations to protect their digital assets.

  
Figure 1: Suricata Working

**2.2.2 Working of Suricata**

Suricata monitors network traffic using signature and anomaly detection to identify threats. It provides alerts and logs to administrators and can block threats when in Intrusion Prevention System (IPS) mode, enhancing network security.

4

* **Packet Capture & Decoding:** Suricata captures network packets and decodes their content to understand communication protocols.
* **Signature-Based Detection**: It identifies known threats by comparing traffic against a database of malicious patterns, much like antivirus software.
* **Anomaly-Based Detection**: Suricata also detects unknown or emerging threats by identifying unusual deviations from normal network behaviour.
* **Logging & Alerting**: When suspicious activity is found, it generates detailed logs and alerts for administrators to investigate.
* **Blocking (IPS Mode):** In Intrusion Prevention System (IPS) mode, Suricata can actively block malicious traffic in real-time, providing an additional layer of security.
* **Deployment Flexibility**: It can be deployed at various points in the network, including the perimeter, internal segments, or in the cloud, based on specific security needs.
* **Configurable Rule Sets**: Administrators can configure Suricata's detection and response capabilities through customizable rule sets.
* **Regular Updates**: Suricata requires regular updates to its signature databases and software to remain effective against evolving threats.

**2.2.3 ELK stack (Elasticsearch, Logstash, and Kibana)**

The ELK stack (Elasticsearch, Logstash, and Kibana) is a popular open-source suite for centralized logging, search, and data visualization. It's widely used for monitoring, troubleshooting, and gaining insights from large volumes of data, especially logs

A diagram of a data processing process

AI-generated content may be incorrect.

Figure 2: ELK Stack

5

**Logstash**

Logstash ingests data from various sources (logs, databases, sensors, etc.), transforms it (e.g., parses, filters, enriches), and then sends it to a "stash."

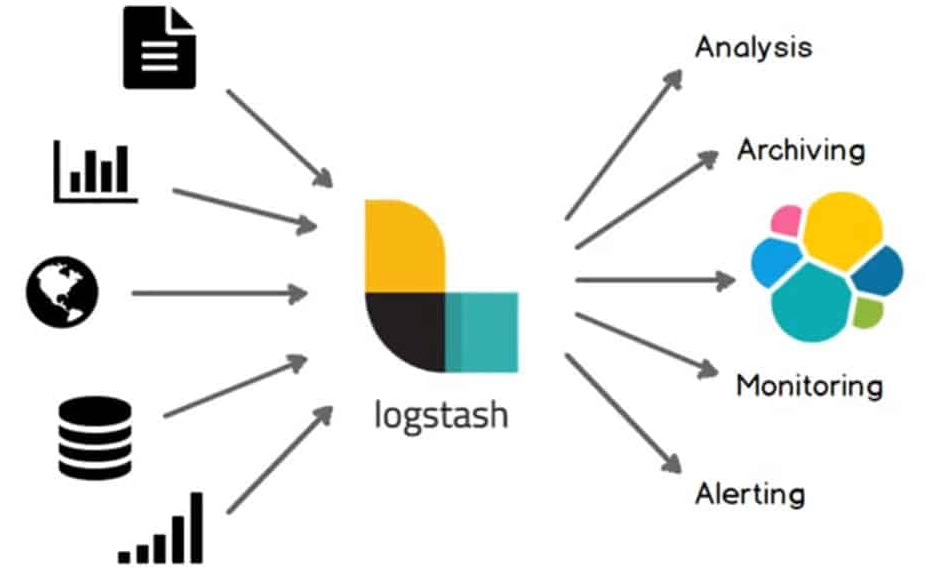


Figure 3: Logstash

**Elasticsearch**

Elasticsearch stores the processed data (often in JSON format) in a way that makes it incredibly fast to search, analyse, and aggregate, even across massive datasets. It uses an "inverted index" for quick lookups.

**Kibana**

Kibana connects to Elasticsearch and allows you to create interactive dashboards, charts, graphs, and maps to visualize your data. This makes complex data easy to understand and helps identify trends, anomalies, and insights in real-time.

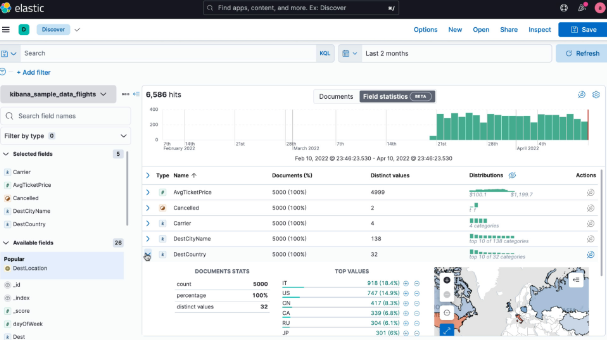


Figure 4: Demo Kibana Dashboard

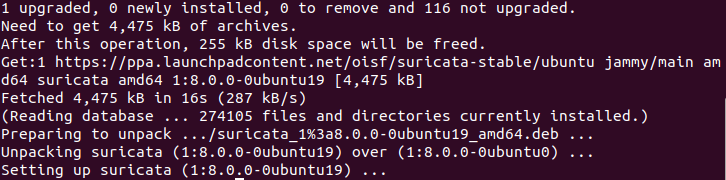
6

**3. Implementation**

**3.1 Install Suricata**

Execute the given commands to setup and install the latest Stable Suricata on Ubuntu.

sudo apt-get install software-properties-common  
sudo add-apt-repository ppa:oisf/suricata-stable;  
sudo apt-get update  
sudo apt-get install suricata

  
Figure 5: Install Suricata

Once Suricata is installed, let’s now check if its running already by using the following command.  
 sudo systemctl status surictata

A computer screen with white text

AI-generated content may be incorrect.  
Figure 6: Status of Suricata

As we can see that Suricata is already running  
Now that Suricata is successfully installed, and its service is running successfully, let’s explore its

7

configuration files located in the /etc/suricata/ directory.   
A screenshot of a computer screen

AI-generated content may be incorrect.

Figure 7: Suricata Directory

Here's a shortened explanation of the four essential Suricata configuration files:

* **classification.config**: Categorizes detected events (e.g., "attempted-dos") to organize Suricata alerts by their nature.
* **reference.config**: Provides external reference information (e.g., about malware, known threats) to add context to alerts.
* **suricata. yaml**: The **main configuration file** for Suricata, controlling all aspects of its behaviour, from network settings to rule management and output.
* **threshold.config**: Manages alert volume by setting conditions to suppress or trigger alerts, preventing overload based on event frequency.

**3.1.1 Create Suricata Rules**

Add a custom brute force detection rule to /etc/suricata/rules/local.rules:

alert ssh any any -> any any (msg:"SSH Brute Force Attempt"; flow:to\_server,established; content:"SSH"; threshold:type both, track by\_src, count 5, seconds 60; classtype:attempted-admin; sid:1000001; rev:1;)

Figure 8: Rule for Detecting SSH Brute Force Attempt

**3.1.2 Configure Suricata**

Edit the Suricata configuration file at /etc/suricata/suricata.yaml

Make sure the rule file is included in suricata.yaml:

  
Figure 9: Updated Rule File Name  
8

Set the output path in /etc/suricata/suricata.yaml

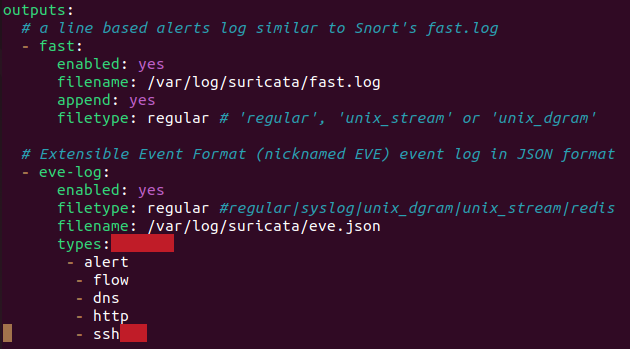


Figure 10: Log Path in Yaml File

Test Configuration

sudo suricata -T -c /etc/suricata/suricata.yaml -v

A computer screen with text

AI-generated content may be incorrect.  
Figure 11: Rule Status Check

**Now we can start Suricata.**

sudo systemctl start suricata

9

**3.2 Install Elasticsearch**

sudo apt install elasticsearch  
  
A computer screen with white text

AI-generated content may be incorrect.

Figure 12: Install Elasticsearch

sudo systemctl start elasticsearch

**3.3 Install Logstash**

sudo apt install logstash  
  
A computer screen with white text

AI-generated content may be incorrect.

Figure 13: Install Elasticsearch

10

**3.3.3 Logstash Pipeline Configuration**

Create a file /etc/logstash/conf.d/suricata.conf:A computer screen shot of a program

AI-generated content may be incorrect.

Figure 14: Logstash Pipeline Configuration commands

**Start Logstash**

sudo systemctl start logstash

**3.4 Install Kibana**

sudo apt install kibana  
sudo systemctl start kibana

A screenshot of a computer program

AI-generated content may be incorrect.  
Figure 15: Install Kibana

Access the Kibana interface

Open the browser and access: <http://localhost:5601>

11

**3.5 Integration Steps**

* Suricata logs alerts to /var/log/suricata/eve.json
* Logstash reads and parses eve.json into structured data
* Data is indexed into Elasticsearch
* Kibana is used to create visual dashboards based on the alert data

**3.6 Install OpenSSH Server (if not already installed):**

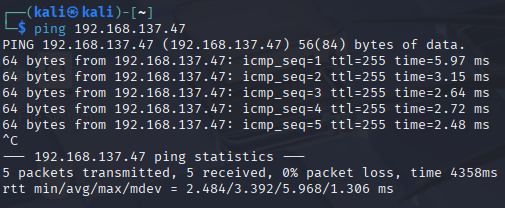
sudo apt update  
sudo apt install openssh-server  
  
A screenshot of a computer program

AI-generated content may be incorrect.

Figure 16: OpenSSH Server

**Enable and Start the SSH Service**  
sudo systemctl enable ssh  
sudo systemctl start ssh

**3.7 Attacks Performed**  
  
**ICMP Ping**

  
Figure 17: Performing ping check on the device connection

12

**Nmap Port Scan**

A computer screen shot of a computer

AI-generated content may be incorrect.

Figure 18: Nmap scanning for open ports

**SSH Brute Force**

Run Hydra Brute Force Attack

**From Kali Linux:**

hydra -l ubuntu -P /usr/share/wordlists/rockyou.txt ssh://192.168.137.47

A screen shot of a computer

AI-generated content may be incorrect.

Figure 19: Attacker performs SSH brute force attack

13

**3.8 Detection**  
Suricata analyses SSH traffic and logs repeated attempts based on custom rule.

**Alert**

The rule triggers an alert "SSH Brute Force Attempt".

**Check the Suricata log detection in Ubuntu**

tail -f /var/log/suricata/eve.json

A computer screen shot of white text

AI-generated content may be incorrect.  
Figure 20: Logs in eve.json after brute force attack from kali.

grep -a “Possible SSH Brute Force Attack” /var/log/suricata/fast.log or eve.json  
  
A computer screen shot of white text

AI-generated content may be incorrect.  
Figure 21: Rule alert  
  
14

**3.9 Kibana Log Overview**

Discover Tab: Search using:

alert.signature: "\*Brute\*

Time Filter: Set to "Last 15 minutes" or a relevant custom range

A screenshot of a computer

AI-generated content may be incorrect.

Figure 22: ELK Log analysis

* Kibana is the web interface used to analyze and visualize log data from Elasticsearch.
* An index pattern (e.g., filebeat-\*) must be created to access Suricata logs.
* **The Discover tab** allows real-time viewing of raw logs with timestamp, source IP, and alert signature details.
* **Kibana Query Language** (KQL) is used to filter and search specific log entries (e.g., by alert type or severity).
* The **Visualize** section helps create charts and graphs to summarize log data.
* **Dashboards** combine multiple visualizations to provide an overview of network threats and activities.

15

Brute Force Detection Dashboard :  
Kibana → Dashboard → Create New Dashboard.  
  
**Logs Over Time**: Shows the number of Suricata alerts across a timeline. Sudden spikes indicate possible brute-force activity.

A screenshot of a graph

AI-generated content may be incorrect.

Figure 23: Logs Over Timeline

**Top Source IPs**: Displays IP addresses generating the most alerts, helping identify the main attacker sources.

A pie chart with numbers and a number on it

AI-generated content may be incorrect.  
Figure 24: Top Source IPs pie chart  
  
16

**4. Results and Observations**

**4.1 Project Summary**

This project focuses on detecting network-based attacks using Suricata IDS integrated with the ELK Stack (Elasticsearch, Logstash, Kibana) in a virtual lab environment. The setup includes an Ubuntu system as the monitored target and a Kali Linux machine to simulate attacks.

An SSH brute-force attack was performed using Hydra from the attacker machine. Suricata was configured to detect these login attempts and generate alerts. Logs were forwarded to ELK via Filebeat, parsed in Logstash, and visualized in Kibana.

**To analyze the attack:**

* A custom Suricata rule was added to detect brute-force behavior.
* Two Kibana visualizations were created: alerts over time and top source IPs
* A dashboard was built to monitor and analyze the attack in real time.

This project demonstrates how open-source tools can be used to simulate and detect cyber threats, providing insights for building a basic intrusion detection system.

**4.2 Conclusion and Future Scope**

This project demonstrated how open-source tools like Suricata IDS and the ELK Stack can be effectively used to detect and analyze network attacks, specifically focusing on SSH brute-force attempts. By simulating an attack from a Kali Linux machine and monitoring logs on an Ubuntu system, the project successfully showed real-time detection, alert generation, and visualization using Kibana dashboards.

In real-life scenarios, such a setup can help organizations improve their security visibility, detect unauthorized access attempts, and perform log-based threat analysis. It provides a cost-effective foundation for building a basic Security Operations Center (SOC).

**Future Scope**

* Integrate tools like Zeek for deeper traffic inspection and protocol analysis.
* Expand detection capabilities to cover attacks like DNS tunneling, port scanning, and web-based intrusions.
* Automate alerts and actions using SIEM rules or scripts.
* Enhance scalability for multi-host or enterprise-level monitoring.
* This project lays the groundwork for building more advanced, real-world intrusion detection and response systems using open-source technologies.

17