Module 3 Assignment: Progress Report 1

Done by: **D. James Chrishan**

Introduction

This project involves setting up a home lab environment for Elastic Stack Security Information and Event Management (SIEM) using Elastic Cloud and a Kali Linux VM. The primary objective is to gain hands-on experience in configuring and managing a SIEM system, generating and analyzing security events, and creating actionable alerts and dashboards. This project is designed to enhance practical skills in security monitoring and incident response, providing a robust learning opportunity for aspiring security analysts and engineers.

Chosen SDLC Model

The **Iterative Model** of the Software Development Life Cycle (SDLC) is selected for this project. This model emphasizes incremental development, allowing each phase of the projects setting up infrastructure, configuring agents, generating events, and analyzing results—to be developed and refined in cycles.

Justification for Selection

The Iterative Model is ideal for this project because:

- It supports gradual learning and adaptation, enabling iterative improvements to configurations and analysis techniques.
- The project tasks, such as agent configuration and event generation, can be implemented and tested in stages, ensuring flexibility in resolving challenges.
- It aligns well with the exploratory and experimental nature of setting up a lab environment.

This approach ensures continuous enhancement of the SIEM system while allowing scope for experimentation and optimization at every step.

System Analysis and Design

Functional Requirements

1. **Log Collection and Forwarding**:
    - Configure Elastic Agent to collect security logs (e.g., process logs, network logs) from the Kali VM and forward them to the Elastic SIEM.
2. **Event Analysis**:
    - Enable querying of collected security events (e.g., Nmap scans, login attempts) in the SIEM.
3. **Visualization**:
    - Create dashboards to visualize security event patterns over time.
4. **Alerting**:
    - Implement alerting mechanisms to detect specific security events like Nmap scans.
5. **Data Integration**:
    - Support integration with Elastic Cloud services for centralized monitoring.

Non-Functional Requirements

1. **Performance**:
    - Ensure low latency in log forwarding and event querying.
2. **Scalability**:
    - Support additional log sources or integrations in the future.
3. **Usability**:
    - Provide an intuitive SIEM interface for querying, analysis, and visualization.
4. **Security**:

    - Secure communication between the agent and Elastic Cloud using encryption.
5. **Reliability**:
    - Maintain consistent log collection and data integrity.

System Architecture

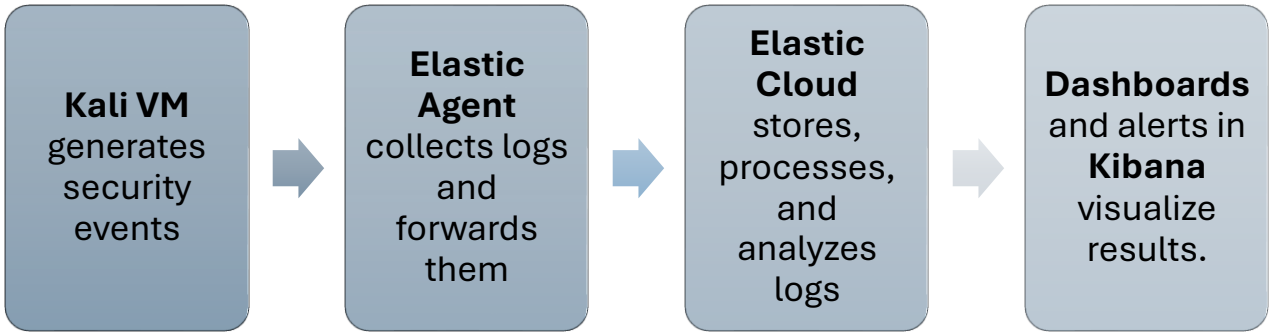
Overview

- The architecture follows a centralized data collection and analysis model.
- Security logs are collected from endpoints (Kali Linux VM) via Elastic Agent and transmitted to the Elastic Cloud for analysis.

Components

1. **Kali Linux VM**:
    - Acts as the endpoint generating logs.
    - Runs Elastic Agent for data forwarding.
2. **Elastic Cloud**:
    - Hosts Elasticsearch, Kibana, and other SIEM components.
3. **Elastic Agent**:
    - Collects logs and forwards them to Elastic Cloud.

Data Flow Diagram (DFD)



Data Model

Field	Description	Example
@timestamp	Timestamp of the event	2024-12-07T12:34:56Z
host.name	Name of the host machine	kali-vm
event.action	Action of the event	nmap_scan
process.args	Command-line arguments	sudo nmap -sT
source.ip	Source IP address	192.168.1.100
destination.ip	Destination IP address	192.168.1.101

Database Schema

- **Elasticsearch Index**: Stores log data from Elastic Agent.
- Fields: *timestamp*, host.name, event.action, process.args, source.ip, destination.ip, etc.
- Quarriable and searchable by Kibana for analysis and visualization.

This structured approach ensures efficient log collection, processing, and analysis, aligning with project objectives.

Coding and Implementation

1. Chosen Programming Languages and Frameworks:

- **Languages:** Bash commands for setting up and verifying configurations.
  - ❖ **Note:** Python can be used for automation; however, due to the project's objective of creating a SIEM tool with minimal technical knowledge, Bash commands and readily available tools like Elastic Agent and Nmap were utilized instead.
- **Frameworks/Tools:**
  - **Elastic Stack:** Elasticsearch and Kibana for log management and visualization.
  - **Kali Linux Tools:** For generating security events.
  - **Elastic Agent:** Installed on the Linux VM to forward logs to Elastic SIEM.
  - **Nmap:** Used for generating network scan events.

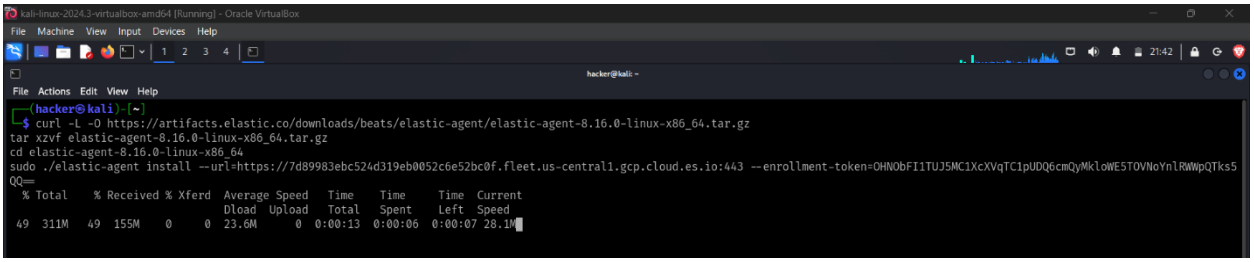
The reliance on Bash commands and Elastic Agent ensures streamlined configuration without the need for additional programming. Elastic Stack provides robust visualization and analytics for security monitoring.

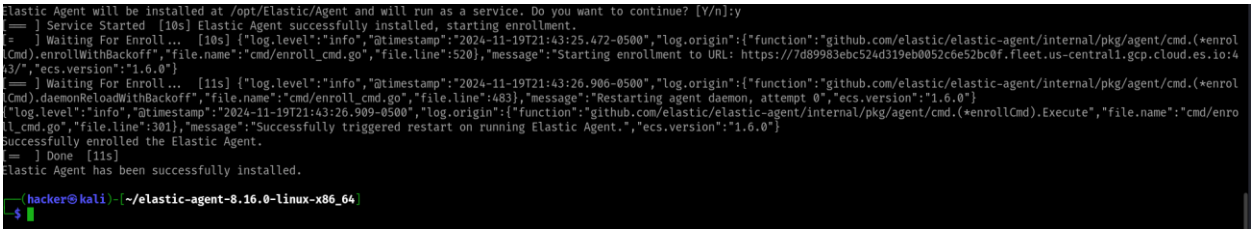
2. Examples of Key Code Snippets or Modules:

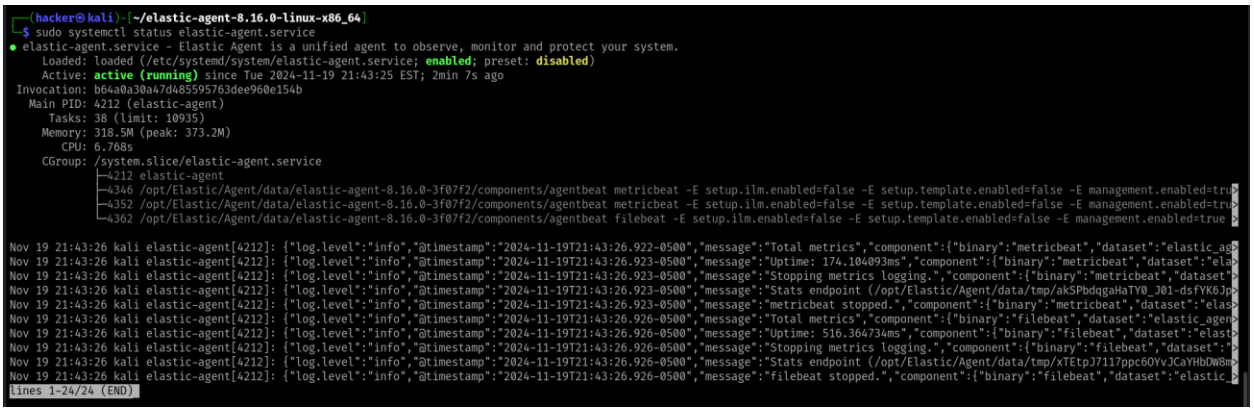
a) Installing Elastic Agent on the Kali VM:

bash

```
#Command to download and install Elastic Agent

curl -L -0 https://artifacts.elastic.co/downloads/beats/elastic-
agent/elastic-agent-8.16.0-linux-x86_64.tar.gz
tar xzvf elastic-agent-8.16.0-linux-x86_64.tar.gz
cd elastic-agent-8.16.0-linux-x86_64
sudo ./elastic-agent install --
url=https://7d89983ebc524d319eb0052c6e52bc0f.fleet.us-
central1.gcp.cloud.es.io:443 -- enrollment-
token=OHNObFI1TUJ5MC1XcXVqTC1pUDQ6cmQyMkloWE5TOVNoYnlRWWpQTks5
QQ=
```







b) Generating Security Events Using Nmap:

bash

#Basic Nmap scan
```
sudo nmap 10.0.2.15
```

#Comprehensive scan of all ports with version detection
```
sudo nmap -A -p- 10.0.2.15
```

#TCP Connect scan
```
sudo nmap -sT 10.0.2.15
```

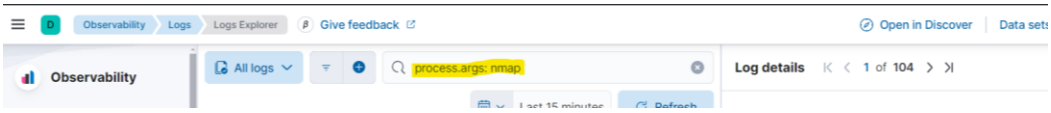
#TCP Connect scan
```
sudo nmap -sT 10.0.2.15
```

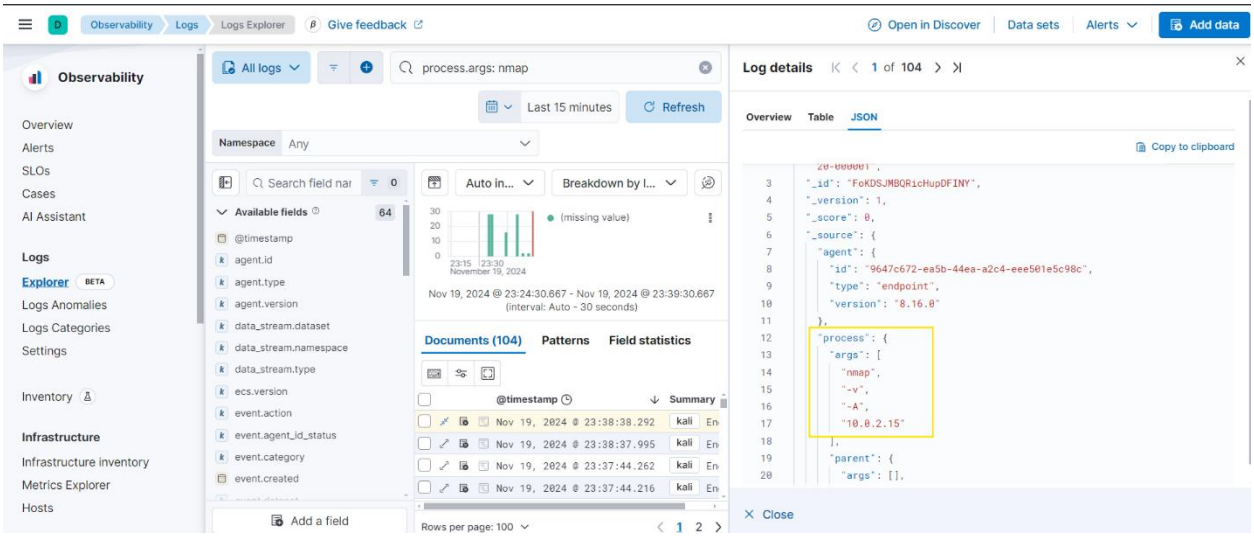
```
┌──(hacker㉿kali)-[~/elastic-agent-8.16.0-linux-x86_64]
└─$ sudo nmap 10.0.2.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-19 21:50 EST
Nmap scan report for 10.0.2.15
Host is up (0.0000080s latency).
All 1000 scanned ports on 10.0.2.15 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds

┌──(hacker㉿kali)-[~/elastic-agent-8.16.0-linux-x86_64]
└─$ sudo nmap -A -p- 10.0.2.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-19 21:54 EST
Nmap scan report for 10.0.2.15
Host is up (0.000064s latency).
All 65535 scanned ports on 10.0.2.15 are in ignored states.
Not shown: 65535 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

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 4.24 seconds

┌──(hacker㉿kali)-[~/elastic-agent-8.16.0-linux-x86_64]
└─$ sudo nmap -sT 10.0.2.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-19 21:56 EST
Nmap scan report for 10.0.2.15
Host is up (0.00024s latency).
All 1000 scanned ports on 10.0.2.15 are in ignored states.
Not shown: 1000 closed tcp ports (conn-refused)

Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
```

d) Querying Logs in Elastic SIEM:

- To filter Nmap-related logs:

Process.args: nmap





Conclusion

1. Summary of Key Achievements and Progress

Through this project, significant strides were made in setting up an Elastic SIEM integrated with a Kali Linux VM for effective cybersecurity monitoring, particularly with a focus on accessibility for users with less technical knowledge. The key achievements include:

1. **Environment Setup**: Successfully configured Elastic SIEM on the Elastic Cloud and deployed a functional Kali Linux VM for log forwarding and threat simulation, emphasizing a user-friendly approach with clear documentation for replication.
2. **Elastic Agent Integration**: Configured the Elastic Agent on the Kali VM to forward logs seamlessly to Elastic SIEM, enabling real-time security event monitoring with straightforward implementation steps.
3. **Threat Simulation**: Generated realistic attack scenarios using Kali Linux tools like Nmap, providing meaningful data for analysis and validation of SIEM functionality, even for users new to these tools.
4. **Dashboard and Alerting**: Designed intuitive dashboards to visualize trends in security events and created alerts for specific incidents, making monitoring accessible and actionable.
5. **Documentation**: Developed a detailed, step-by-step guide to ensure the setup is replicable and accessible for broader audiences, including those with minimal technical expertise.

These achievements highlight the feasibility of creating a hands-on lab environment for cybersecurity training that is both cost-effective and suitable for users with varying levels of technical knowledge.

2. Reflection on Challenges Faced and How They Were Addressed

1. **Configuration Issues**: Challenges arose during Elastic Agent setup, particularly in ensuring seamless log forwarding from the Kali VM. These were resolved by cross-referencing Elastic documentation and simplifying the instructions for easier understanding.
2. **Delayed Log Ingestion**: Some delays were encountered in logs appearing within the Elastic SIEM. Patience and reconfiguration of the data ingestion pipeline mitigated these issues.

3. **Complexity for Beginners**: The steep learning curve for Kali Linux and Elastic SIEM was a barrier, but providing simplified steps and user-friendly documentation reduced this challenge, making the setup approachable for less experienced users.
4. **Integration Documentation**: Limited resources on integrating Elastic SIEM with Kali Linux required referencing multiple sources and applying creative problem-solving to bridge gaps. Simplifying the integration steps has made the process more accessible.
5. **Free Version Constraints**: The use of Elastic's free trial posed challenges, requiring an extension to complete the project. While this is a limitation in a small-scale setup, it is negligible in enterprise scenarios where full licenses are standard.
6. **Time Constraints**: Adhering to the project timeline while ensuring quality was challenging but achieved through careful planning and prioritization.

Overall, the project demonstrates the potential of combining Elastic SIEM with Kali Linux to create a practical and accessible lab environment for cybersecurity training. It underscores the importance of simplifying processes to enable broader adoption by users with less technical knowledge, while future improvements could focus on advanced integration techniques and scaling solutions for enterprise use.