END-TO-END NETWORK SIEM PIPELINE USING ZEEK, FILEBEAT AND THE ELK STACK

OVERVIEW

In this project, I set up a **Security Information and Event Management (SIEM)** pipeline using **Zeek, Filebeat**, and the **ELK Stack (Elasticsearch, Logstash, Kibana)**.The objective of this project was to capture live network traffic, convert it into structured security logs, centralise those logs, and perform real-time analysis and visualisation.
This setup closely resembles how SIEM systems are deployed in SOC (Security Operations Center) and DFIR (Digital Forensics & Incident Response) environments.

1. INTRODUCTION

Security monitoring and incident detection are critical components of modern networked environments. Organisations rely on **Security Information and Event Management (SIEM)** systems to collect, correlate, and analyse logs from multiple sources in real time.

In this project, I implemented a practical **SIEM pipeline using Zeek for network traffic analysis, Filebeat for log forwarding, and the ELK Stack for centralised storage and visualisation.**
The goal was to gain hands-on experience with SOC-style tooling and demonstrate how live network traffic can be transformed into actionable security intelligence.

1.1 PLATFORM AND ARCHITECTURE

This project was implemented on **Kali Linux running on ARM architecture.**
Some security tools provide different binaries or container images depending on system architecture (ARM vs x86_64). Wherever required, ARM-compatible images or repositories were used. The SIEM design, data flow, and analysis logic remain identical to x86-based deployments, only the installation sources differ due to hardware compatibility.

1.2 Why Live Log Capture and Analysis is Important

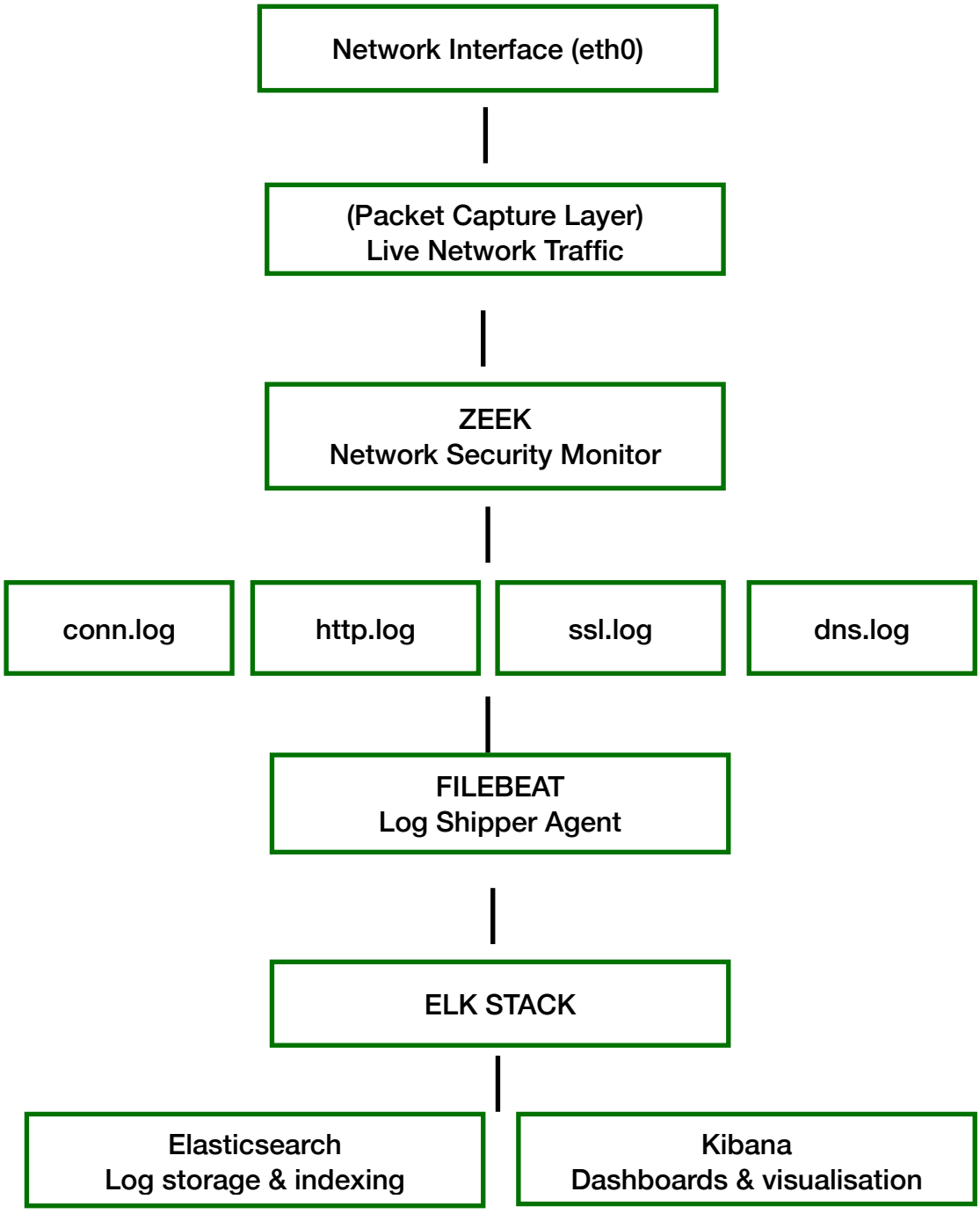
- It provides real-time visibility into attacks & suspicious behaviour.
- It enables faster incident detection & response.
- It is helpful in detecting anomalies (e.g., unusual traffic, brute force attempts), and supports threat hunting and forensic analysis.
- It provides centralised logging from multiple endpoints.

1.3 Log sources and Data Analysed

The SIEM continuously collects and analyses the following types of log data to support monitoring, threat detection and forensic investigation:

- HTTP traffic
- DNS queries
- Connections (TCP/UDP)
- SSL/TLS logs
- File modifications
- Authentication and authorisation attempts
- System errors and application status codes

These logs provide high-value network metadata that can be correlated to detect attacks, misconfigurations, or abnormal behaviour.



2. COMPONENTS

2.1 ZEEK (Network Traffic Analysis)

Zeek is a passive, open-source network traffic analyser commonly used as a Network Security Monitor (NSM). It observes live network traffic and generates detailed, structured logs describing network connections and application-layer activities. Zeek helps security teams investigate suspicious or malicious behaviour and also supports network performance monitoring and troubleshooting. It is used as the primary network traffic analysis and monitoring tool in this project.

Zeek was managed using **zeekctl,** which provides centralised control over zeek processes and logging.
Running zeek through zeekctl ensures that all logs are generated in a consistent and structured directory hierarchy **(/opt/zeek/logs/),** rather than being **written to the current working directory.** This makes log management, rotation and integration with Filebeat more reliable.

Key logs generated by zeek include:

- **conn.log** - Network connection metadata (TCP/UDP)
- **http.log** - HTTP request and response details
- **dns.log** - DNS queries and responses
- **ssl.log** - TLS/SSL session metadata
- **files.log** - File transfer information
- **weird.log** - Unusual or malformed network behaviour

Key Features & Functions:

- Generates detailed logs of network activity
- Records connection, DNS, HTTP, SSL, and other protocol data
- Produces structured logs in TSV or JSON format
- Enables security monitoring and network investigations

2.2 FILEBEAT (Log Shipper)

Filebeat is a lightweight log shipper that is part of the Elastic (ELK) Stack. It runs as an agent on the source system, monitors specified log files, and forwards log data to Elasticsearch or Logstash. Filebeat tracks its reading position to ensure reliable and continuous log delivery without data loss.

Key Features & Functions:

- Collects and forwards logs in real time
- Lightweight with minimal system impact
- Ensures reliable log delivery even after interruptions
- Integrates seamlessly with the ELK Stack

2.3 ELK Stack (log storage and Visualisation)

The ELK Stack is a centralised log management and analytics platform used to collect, store, search, and visualize large volumes of log data. It enables real-time monitoring, analysis, and correlation of logs for security, troubleshooting, and operational insights.

Key Components & Functions:

- **Elasticsearch**: Stores and indexes log data, enabling fast search and analytics.
- **Logstash**: Processes, parses, and enrich logs before storage (optional).
- **Kibana**: Provides dashboards and visualisations for log analysis and monitoring.

The ELK Stack was deployed using **Docker containers**, which simplifies deployment, improves isolation, and makes the setup portable across systems.
Docker was used only for the ELK Stack, while Zeek and Filebeat run directly on the host system.

2.4 Overall Data Flow

1. Zeek captures live network traffic from the network interface.
2. Zeek generates structured logs in a centralised directory.
3. Filebeat monitors these logs and forwards them to Elasticsearch.
4. Elasticsearch indexes the data.
5. Kibana is used to visualise, search, and analyse the logs in real time.

5. INSTALLATION AND SETUP STEPS

5.1 ZEEK

Zeek was installed on Kali Linux using an **ARM-compatible package repository** to ensure compatibility with the system architecture.

Step 1: Add the Zeek package repository

echo 'deb https://download.opensuse.org/repositories/security:/zeek/xUbuntu_22.04/ /' \| sudo tee /etc/apt/sources.list.d/security:zeek.list

```
┌──(niniatick㉿kaliKali)-[~]
└─$ echo 'deb https://download.opensuse.org/repositories/security:/zeek/xUbuntu_22.04/ /' \| sudo tee /etc/apt/sources.list.d/security:zeek.list
```

Step 2: Import the repository signing key

curl -fsSL https://download.opensuse.org/repositories/security:zeek/xUbuntu_22.04/Release.key \| gpg --dearmor \| sudo tee /etc/apt/trusted.gpg.d/security_zeek.gpg > /dev/null

```
┌──(niniatick㉿kaliKali)-[~]
└─$ curl -fsSL https://download.opensuse.org/repositories/security:zeek/xUbuntu_22.04/Release.key \| gpg --dearmor | sudo tee /etc/apt/trusted.gpg.d/security_zeek.gpg > /dev/null
```

Step 3: Update the package list

sudo apt update

```
┌──(niniatick㉿kaliKali)-[~]
└─$ sudo apt update
```

Step 4: Install Zeek

sudo apt install zeek-7.0

```
┌──(niniatick㉿kaliKali)-[~]
└─$ sudo apt install zeek-7.0
```

Step 5: Navigate to the Zeek binary directory

After installation, Zeek binaries **were located under the following directory:**

cd /opt/zeek/bin

Zeek was executed from this directory to ensure the correct environment, policy scripts, and configuration files were used.

Step 6: Initialise and deploy Zeek using zeekctl

sudo ./zeekctl deploy

zeekctl was used to manage Zeek processes and ensure logs were generated in a structured and centralised directory.

5.2 ELK STACK

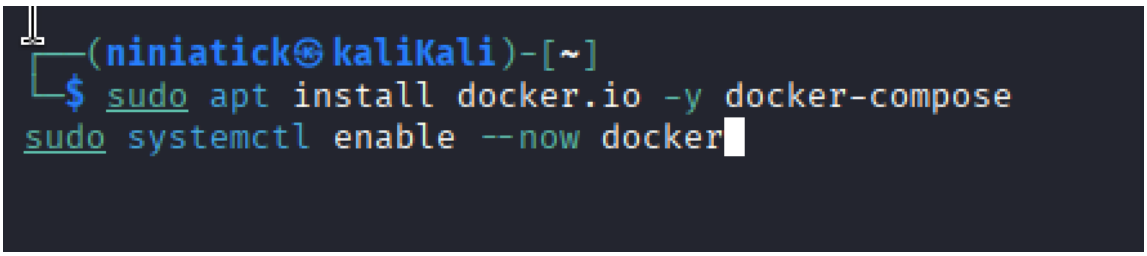
The ELK Stack was deployed using **Docker** and **Docker Compose** to ensure isolation, consistency, and reproducibility. Elasticsearch security features are **enabled by default (Elastic Stack 8.x)**, requiring authentication for access to Elasticsearch and Kibana.

Step 1: Update the system

sudo apt update

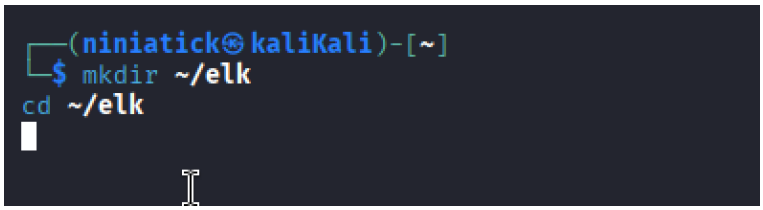
Step 2: Install Docker and Docker Compose

sudo apt install docker.io -y docker-compose
sudo systemctl enable --now docker



Step 2: Create ELK working directory

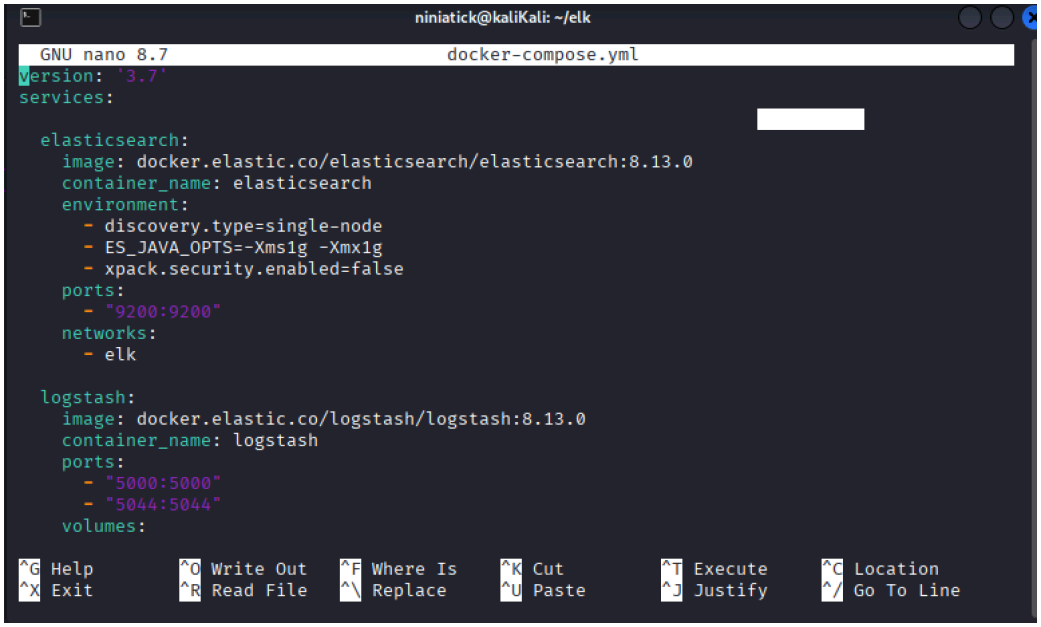
mkdir ~/elk
cd ~/elk



This directory stores **ELK configuration files.**

Step 3: Create Docker Compose configuration

nano docker-compose.yml



Save and Exit
CTRL + O, ENTER, CTRL + X

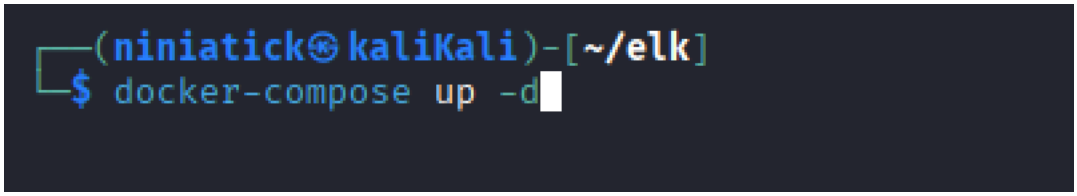
Step 4: Create Logstash pipeline

nano logstash.conf

Save and Exit
CTRL + O, ENTER, CTRL + X

Step 5: Deploy ELK Stack

docker-compose up -d



This command:

- Automatically downloads ELK Docker images
- Creates and **starts Elasticsearch, Logstash, and Kibana containers**

Step 6: Verify Running Containers

docker ps

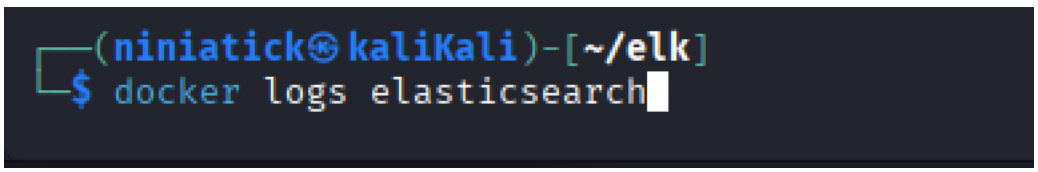
The running status of all ELK containers is shown.

Step 7: Retrieve Elasticsearch Credentials

Elastic Stack 8.x automatically generates credentials during first startup.

To retrieve the generated password:

docker logs elasticsearch



Step 8: Access Kibana

http://localhost:5601

Login using:

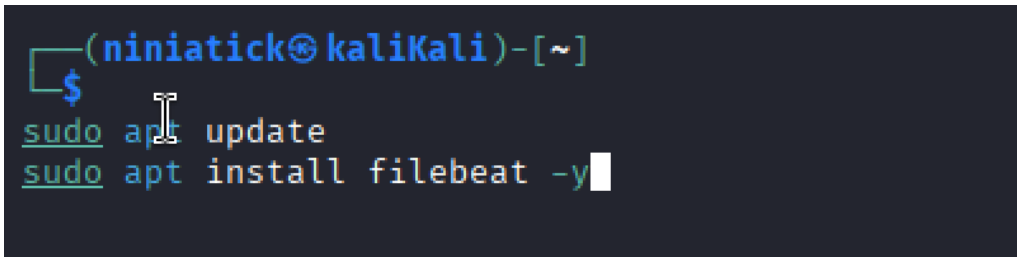
- **Username: elastic**
- **Password: generated or reset password**

5.3 FILEBEAT

Filebeat was installed directly on the host system using the system package manager.

Step 1: Update and Install Filebeat

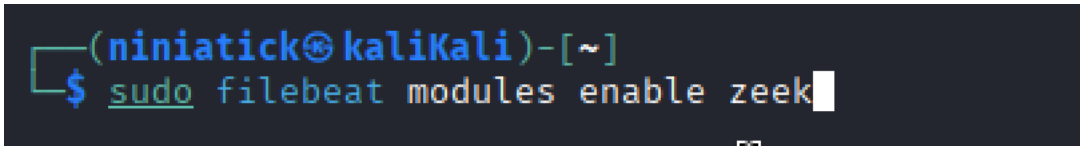
sudo apt update
sudo apt install filebeat -y



Step 2: Enable the Zeek Module

Filebeat provides a built-in Zeek module that understands Zeek log formats and parses them into structured fields.

sudo filebeat modules enable zeek



This creates a symlink in modules.d so Filebeat knows to load it.

Step 3: Configure Zeek Log Path

Edit the Zeek module configuration:

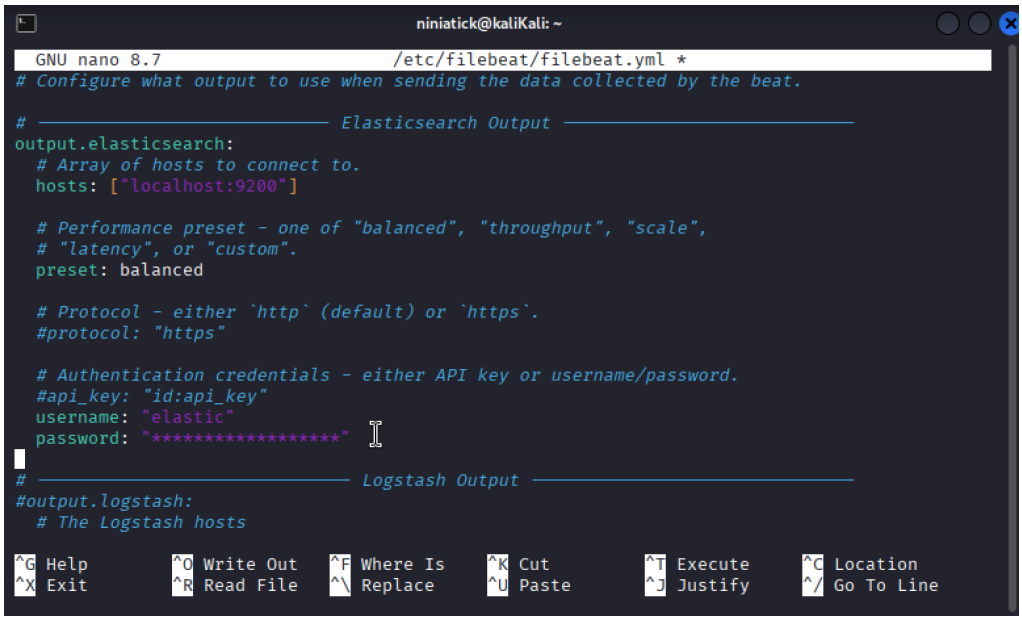
sudo nano /etc/filebeat/modules.d/zeek.yml

Save and Exit
CTRL + O, ENTER, CTRL + X

Step 4: Configure Elasticsearch Output and Authentication

sudo nano /etc/filebeat/filebeat.yml

Edit the main Filebeat configuration file:



Save and Exit
CTRL + O, ENTER, CTRL + X

The hosts field defines the Elasticsearch server and port where Filebeat sends logs, while the **username** and **password** authenticate Filebeat to ensure only authorised access to Elasticsearch for secure log ingestion.

6. INTEGRATION AND DATA VALIDATION

To Verify the complete SIEM pipeline from network capture to log visualisation.

Step 1: Start Zeek

cd /opt/zeek/bin
sudo ./zeekctl deploy

```
┌──(niniatick㉿kaliKali)-[/opt/zeek/bin]
└─$ sudo ./zeekctl deploy

[sudo] password for niniatick:
checking configurations ...
installing ...
removing old policies in /opt/zeek/spool/installed-scripts-do-not-touch/site ...
removing old policies in /opt/zeek/spool/installed-scripts-do-not-touch/auto ...
creating policy directories ...
installing site policies ...
generating standalone-layout.zeek ...
generating local-networks.zeek ...
generating zeekctl-config.zeek ...
generating zeekctl-config.sh ...
stopping ...
stopping zeek ...
starting ...
starting zeek ...
```

Zeek begins capturing live network traffic
Logs are generated in */opt/zeek/logs/current/*
zeekctl ensures logs are stored in a structured directory

Step 2: To verify and view the logs generated from live network traffic capture

cd /opt/zeek/logs

```
┌──(root㉿kaliKali)-[/opt/zeek/bin]
└─# cd /opt/zeek/logs

┌──(root㉿kaliKali)-[/opt/zeek/logs]
└─# ls
2025-12-11  2025-12-15  current

┌──(root㉿kaliKali)-[/opt/zeek/logs]
└─# cd /opt/zeek/logs/2025-12-15

┌──(root㉿kaliKali)-[/opt/zeek/logs/2025-12-15]
└─# ls
capture_loss.13:57:48-14:00:00.log.gz  dns.13:56:59-14:00:00.log.gz            ntp.14:01:57-15:00:00.log.gz           stats.13:56:48-14:00:00.log.gz
capture_loss.14:00:00-15:00:00.log.gz  dns.14:00:00-15:00:00.log.gz            packet_filter.13:56:48-14:00:00.log.gz stats.14:00:00-15:00:00.log.gz
conn-summary.13:56:59-14:00:00.log.gz  files.14:36:43-15:00:00.log.gz          quic.13:58:14-14:00:00.log.gz          telemetry.13:56:48-14:00:00.log.gz
conn-summary.14:00:00-15:00:00.log.gz  http.14:36:43-15:00:00.log.gz           quic.14:00:00-15:00:00.log.gz          telemetry.14:00:00-15:00:00.log.gz
conn.13:56:59-14:00:00.log.gz          known_hosts.13:57:13-14:00:00.log.gz    software.14:36:43-15:00:00.log.gz      weird.14:01:57-15:00:00.log.gz
conn.14:00:00-15:00:00.log.gz          known_services.13:57:23-14:00:00.log.gz ssl.13:57:13-14:00:00.log.gz           x509.14:03:42-15:00:00.log.gz
dhcp.14:02:07-15:00:00.log.gz          loaded_scripts.13:56:48-14:00:00.log.gz ssl.14:00:00-15:00:00.log.gz
```

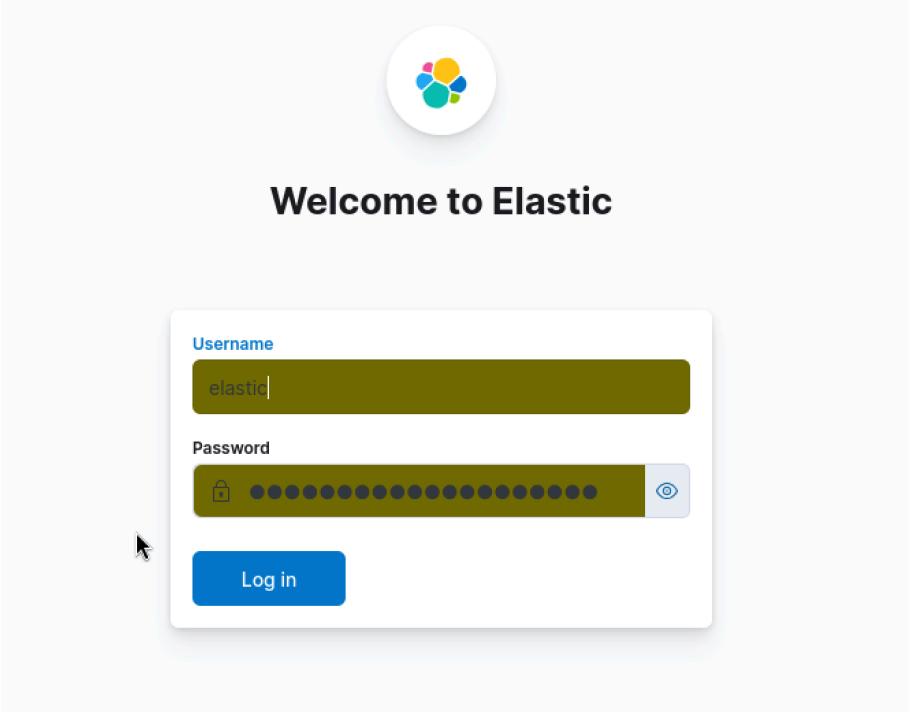
Step 3: Start ELK and Kibana

Sudo docker start elasticsearch
Sudo docker start kibana

```
┌──(niniatick㉿kaliKali)-[~/elk]
└─$ sudo docker start elasticsearch

[sudo] password for niniatick:
elasticsearch

┌──(niniatick㉿kaliKali)-[~/elk]
└─$ sudo docker start kibana

kibana
```

Step 3: Access Kibana

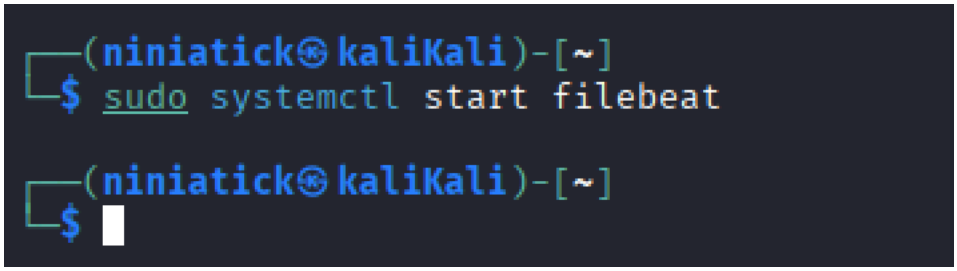
http://localhost:5601



Log in using elastic user and the generated password

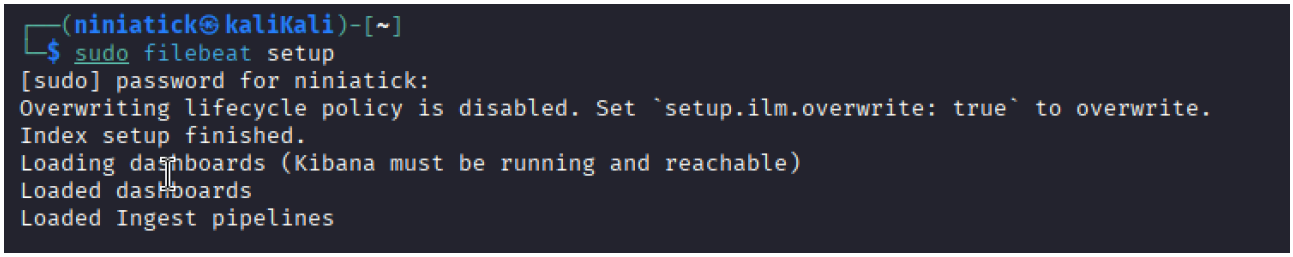
Step 4: Start Filebeat Service

sudo systemctl start filebeat



Step 5: Load Dashboards

sudo filebeat setup



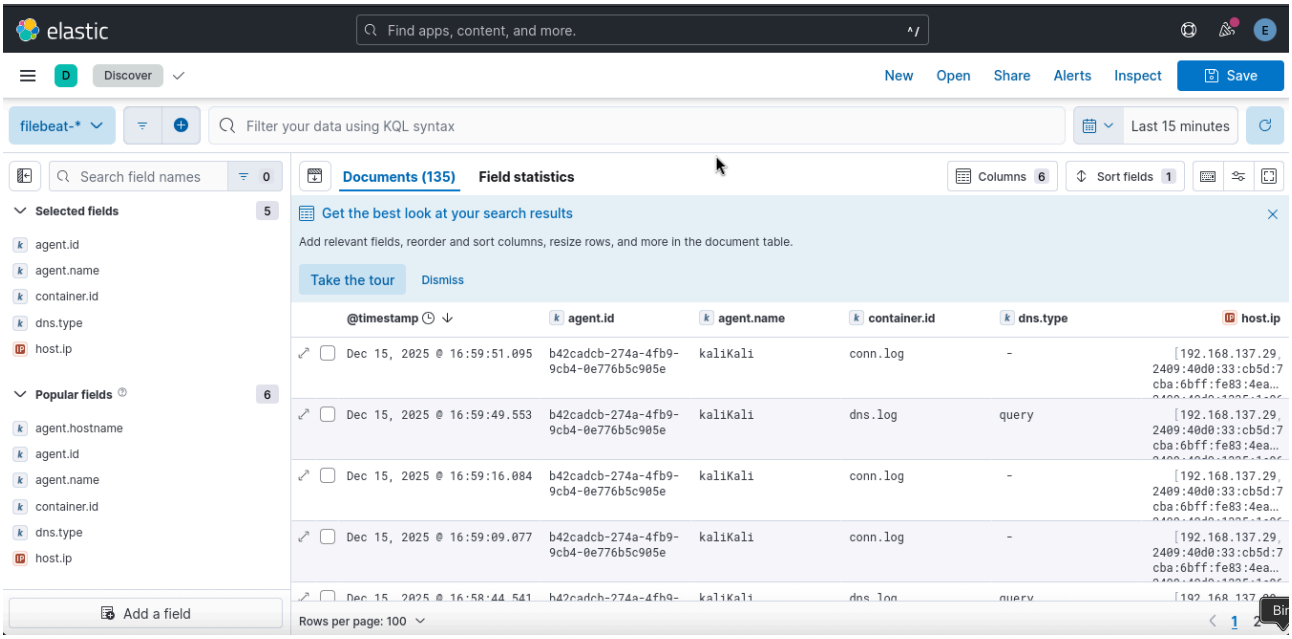
Filebeat monitors the Zeek log directory

Logs are forwarded directly to Elasticsearch

Loads prebuilt Kibana dashboards for Zeek logs

Step 6: Analyse Logs in Kibana
Open Discover in Kibana

Select the relevant index pattern (e.g., filebeat-*)



Search, filter, and visualize Zeek-generated network logs

7. Conclusion

Setting up a **Zeek → Filebeat → ELK** SIEM pipeline provides a powerful, real-time log analysis environment.
This infrastructure allows security teams to capture **live network traffic, ship logs efficiently**, **visualise patterns, detect threats, and perform detailed forensic investigations.**

It is a highly scalable and industry-standard architecture for **SOC analysts, DFIR teams,** and **cybersecurity researchers.**