Heimdall

2406 - Artificial Intelligence system for Threat Hunting and Detection

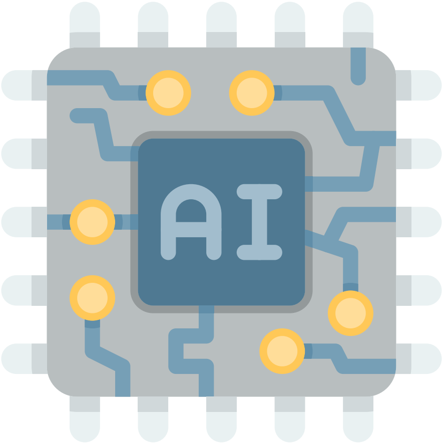
Total Slides: 24

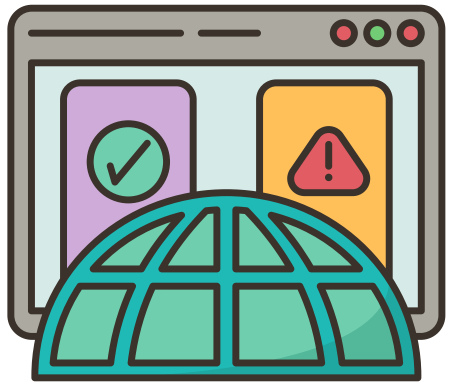
Group Members :

Mubahil Ahmad - 211037
Ahsan Ahmed - 211059
Syed Ali Zain - 211113

Supervised by:

Dr. Zunera Jalil





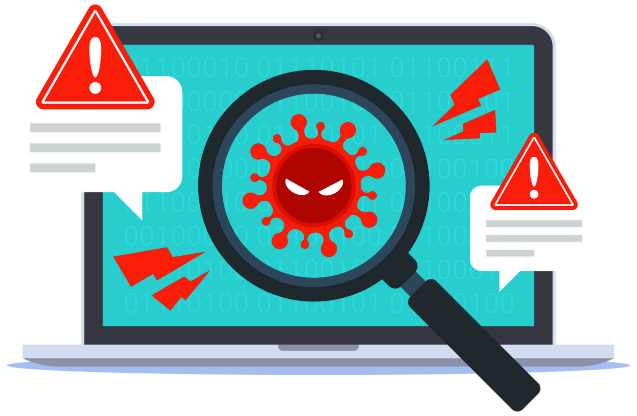
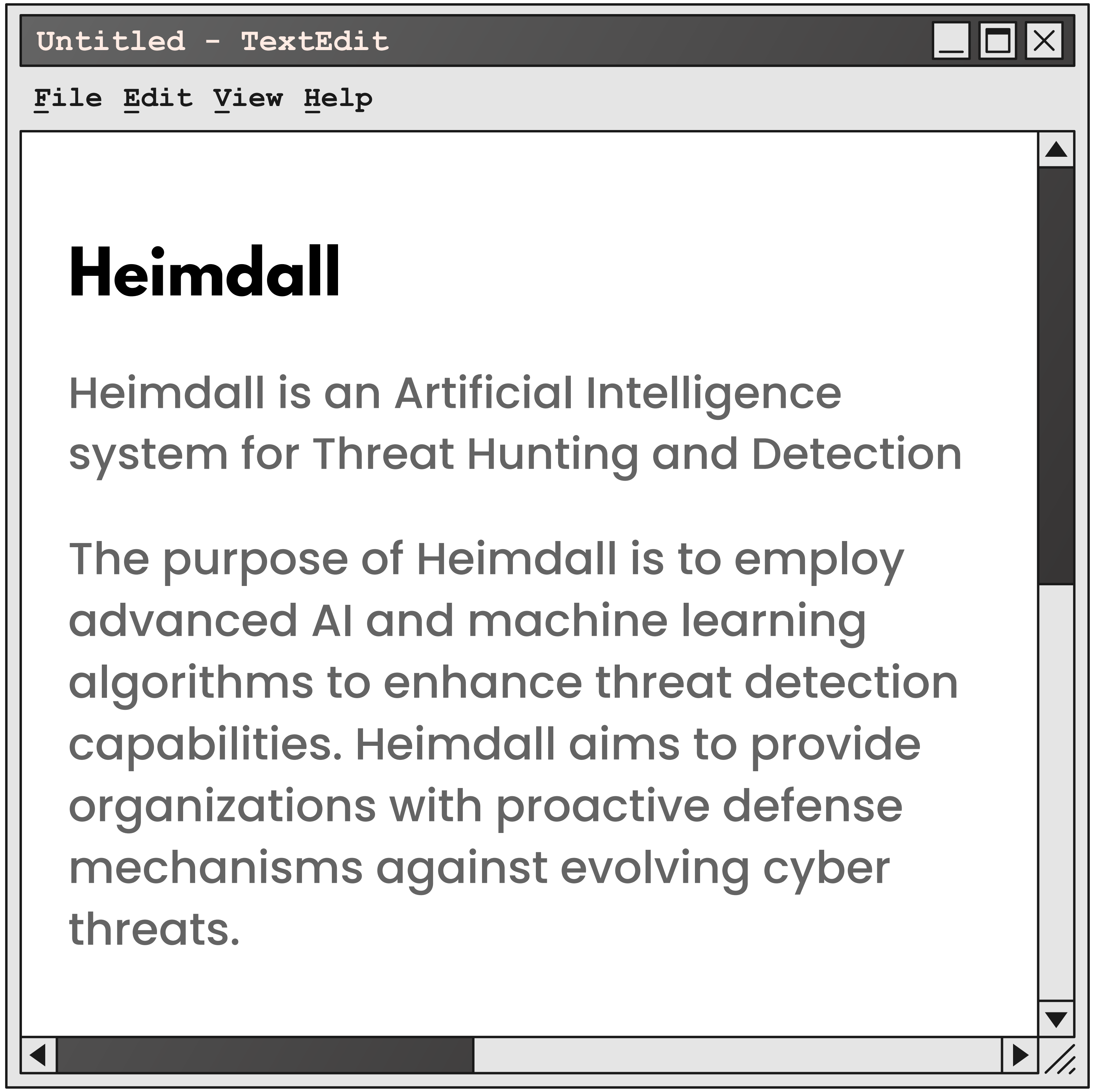


Table of Content

1	Introduction
2	Problem Defination
3	Existing Solutions
4	Proposed Solution
5	Design and Implementation
6	Future Work and Commercialization
7	Demo

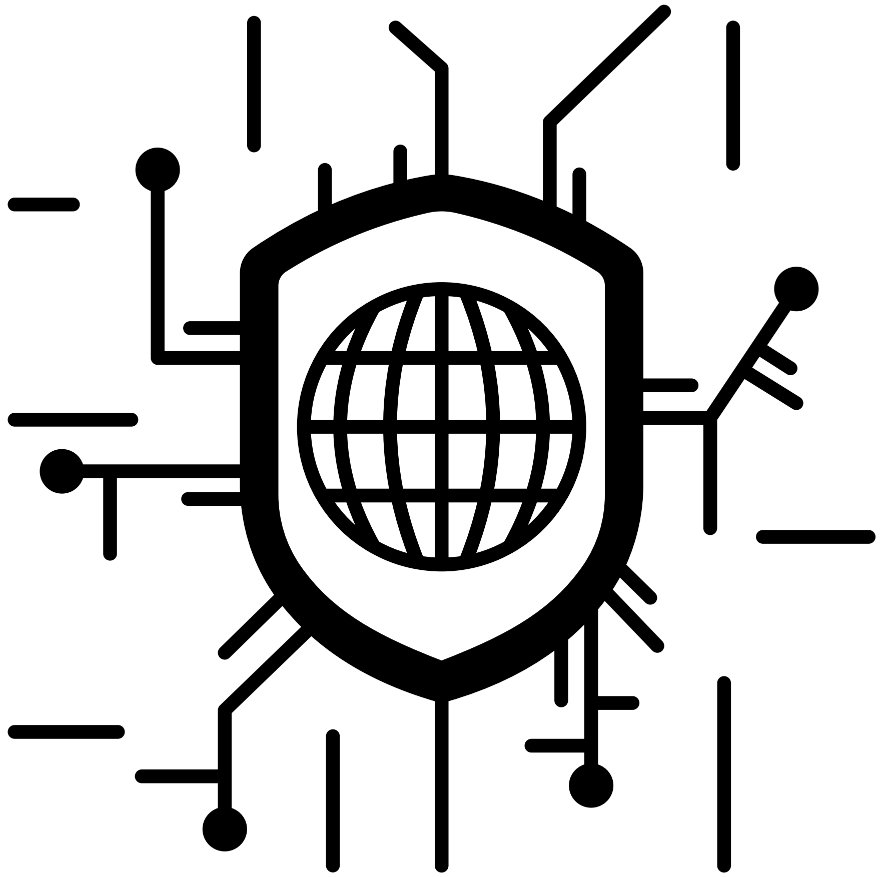
Introduction



Heimdall

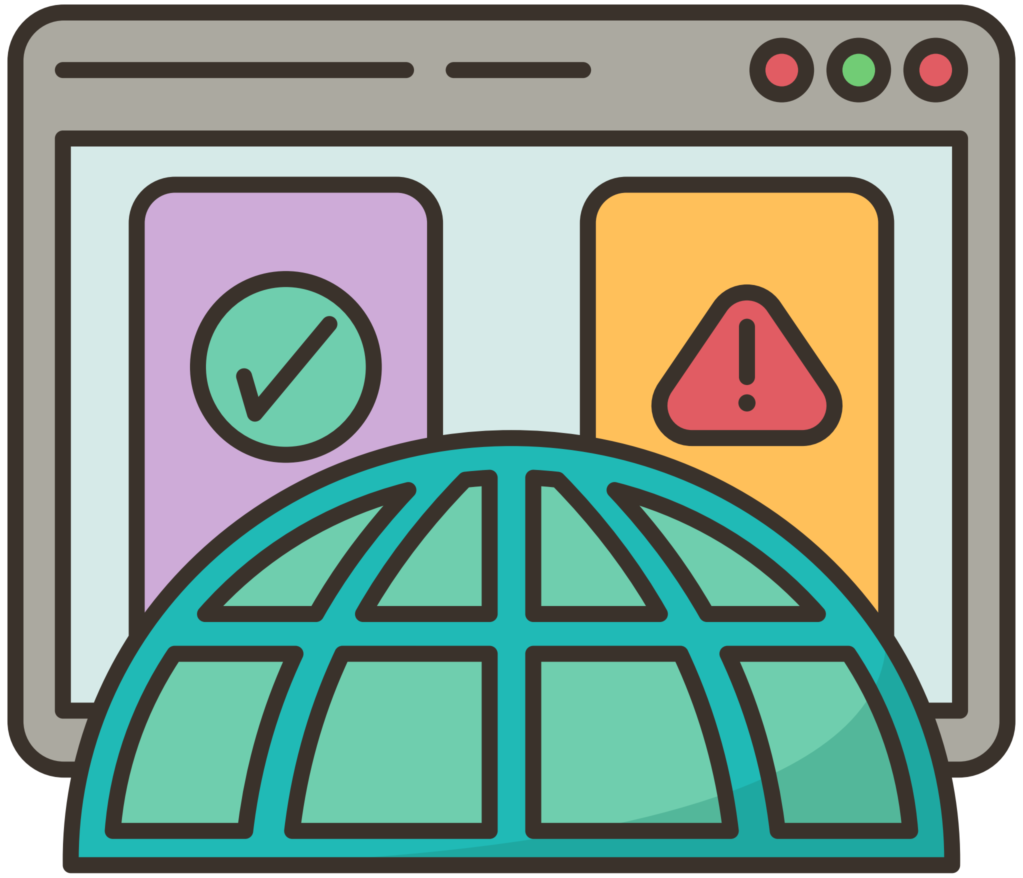
Heimdall is an Artificial Intelligence system for Threat Hunting and Detection

The purpose of Heimdall is to employ advanced AI and machine learning algorithms to enhance threat detection capabilities. Heimdall aims to provide organizations with proactive defense mechanisms against evolving cyber threats.



PROBLEM DEFINITION

PROBLEM DEFINITION



What problem is your FYP addressing?

Traditional intrusion detection systems (IDS) rely on static, signature-based rules, making them ineffective against novel cyber threats like zero-day attacks, polymorphic malware, and concept drift in network behavior.

Why is this problem significant?

Modern network environments are dynamic, distributed, and high-volume. Static IDS cannot adapt in real-time, leading to false positives/negatives and increased security risks. There's a pressing need for adaptive, intelligent solutions that can evolve with network behavior.

2. EXISTING SOLUTIONS

EXISTING SOLUTIONS

Features	Fidelis Elevate	Carbon Black	Stealth Watch	Vectra AI	Proposed Solution
AI-Based Threat Detection	✓		✓	✓	✓
Real-Time Threat Response			✓		✓
Adaptive Learning			✓		✓
Seamless Integration		✓		✓	✓
High Scalability	✓	✓	✓		✓
Low Resource Usage		✓		✓	✓
Zero-Day Vulnerability Identification					✓

Feature Missing in Most Solutions	Why It Matters
Adaptive Learning	Lacks adaptive decision-making over time without manual intervention.
Zero-Day Vulnerability Detection	Static models struggle to detect novel, previously unseen attacks.
Low Resource Usage + Real-Time Adaptability	Most commercial tools are too heavy for small-scale or educational setups.
All-in-One Integration	Features like Kafka-based streaming + Prometheus + retraining are absent.

HOW HEIMDALL FILLS THE GAPS:

AI-Based Threat Detection	✓
Real-Time Threat Response	✓
Adaptive Learning	✓
Seamless Integration (Docker)	✓
High Scalability with lightweight architecture	✓
Zero-Day Threat Identification through anomaly-based ML	✓

PROPOSED SOLUTION

PROPOSED SOLUTION: HEIMDALL

Heimdall is a real-time, dual model Threat Hunting and Detection system designed to:

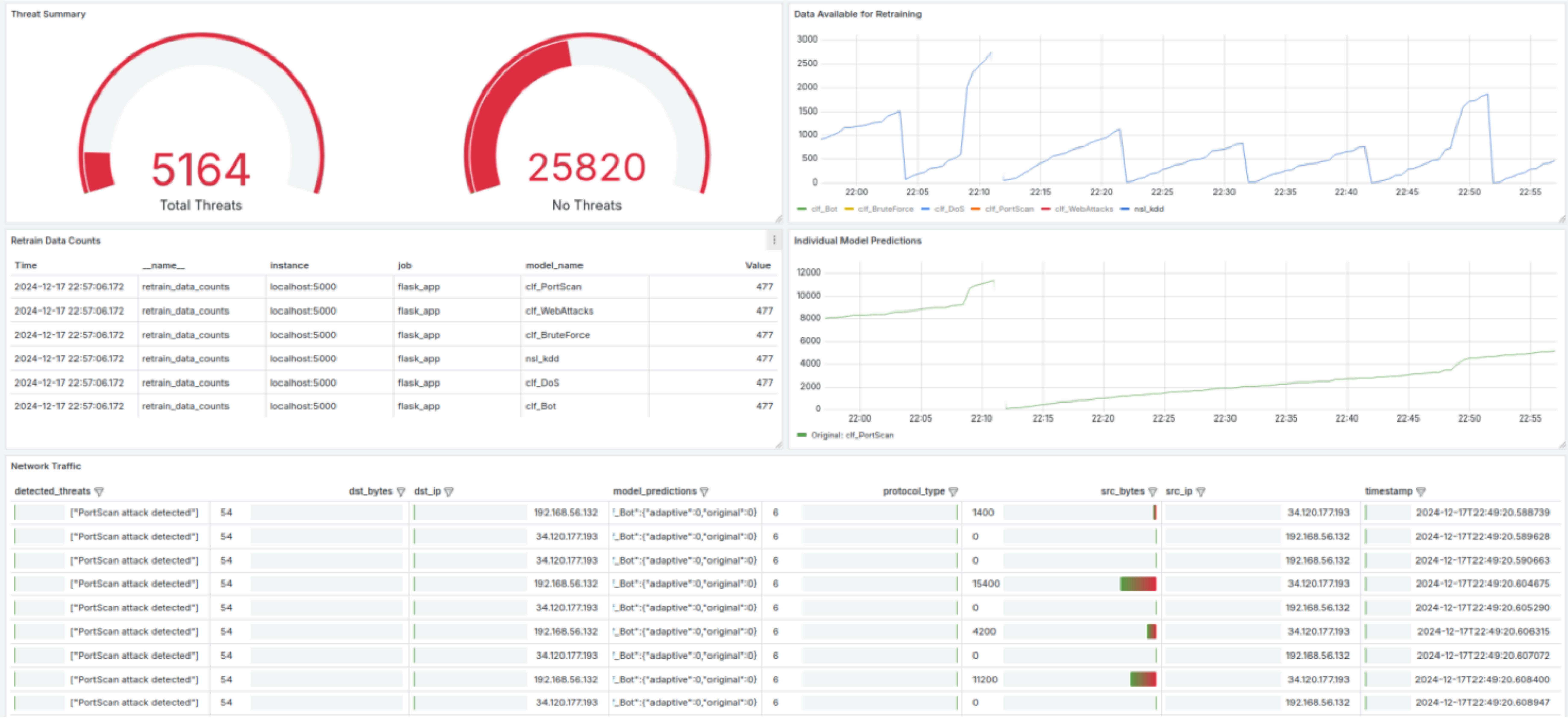
- Capture live network traffic using Scapy
- Stream packet features via Apache Kafka
- Detect threats using supervised and un-supervised ML models
- Visualize results on a Grafana dashboard

PROPOSED SOLUTION: HEIMDALL

Core Features:

- Real-time classification of network packets
- Adaptive retraining (automatic & manual) to handle concept drift
- Prometheus + Grafana integration for system metrics
- Lightweight and Docker-ready deployment

PROPOSED SOLUTION: HEIMDALL



5164

Total Threats

25820

No Threats

Retrain Data Counts

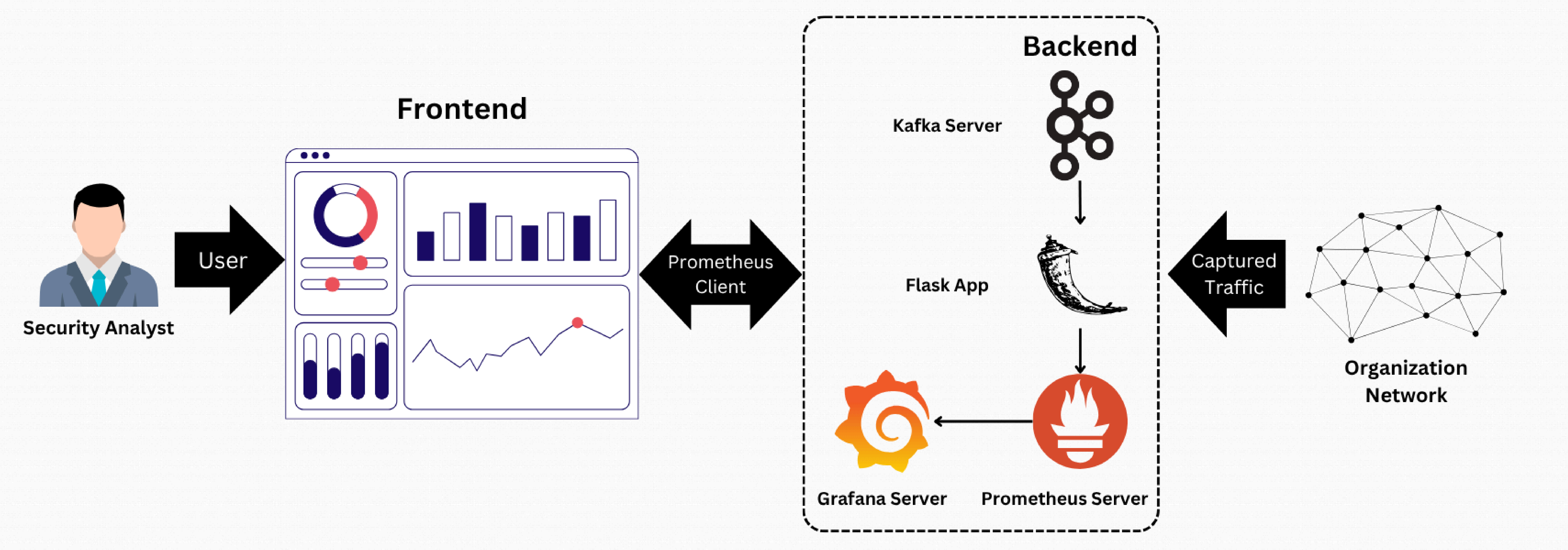
Time	__name__	instance	job	model_name	Value
2024-12-17 22:57:06.172	retrain_data_counts	localhost:5000	flask_app	clf_PortScan	477
2024-12-17 22:57:06.172	retrain_data_counts	localhost:5000	flask_app	clf_WebAttacks	477
2024-12-17 22:57:06.172	retrain_data_counts	localhost:5000	flask_app	clf_BruteForce	477
2024-12-17 22:57:06.172	retrain_data_counts	localhost:5000	flask_app	nsl_kdd	477
2024-12-17 22:57:06.172	retrain_data_counts	localhost:5000	flask_app	clf_DoS	477
2024-12-17 22:57:06.172	retrain_data_counts	localhost:5000	flask_app	clf_Bot	477

Network Traffic

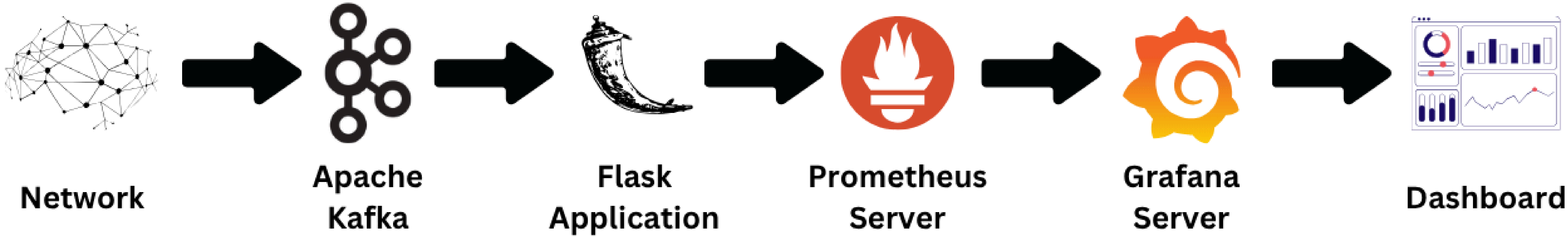
detected_threats	dst_bytes	dst_ip	model_predictions	protocol_type	src_bytes	src_ip	timestamp
["PortScan attack detected"]	54	192.168.56.132	'_Bot":{"adaptive":0,"original":0}	6	1400	34.120.177.193	2024-12-17T22:49:20.588739
["PortScan attack detected"]	54	34.120.177.193	'_Bot":{"adaptive":0,"original":0}	6	0	192.168.56.132	2024-12-17T22:49:20.589628
["PortScan attack detected"]	54	34.120.177.193	'_Bot":{"adaptive":0,"original":0}	6	0	192.168.56.132	2024-12-17T22:49:20.590663
["PortScan attack detected"]	54	192.168.56.132	'_Bot":{"adaptive":0,"original":0}	6	15400	34.120.177.193	2024-12-17T22:49:20.604675
["PortScan attack detected"]	54	34.120.177.193	'_Bot":{"adaptive":0,"original":0}	6	0	192.168.56.132	2024-12-17T22:49:20.605290
["PortScan attack detected"]	54	192.168.56.132	'_Bot":{"adaptive":0,"original":0}	6	4200	34.120.177.193	2024-12-17T22:49:20.606315
["PortScan attack detected"]	54	34.120.177.193	'_Bot":{"adaptive":0,"original":0}	6	0	192.168.56.132	2024-12-17T22:49:20.607072
["PortScan attack detected"]	54	192.168.56.132	'_Bot":{"adaptive":0,"original":0}	6	11200	34.120.177.193	2024-12-17T22:49:20.608400
["PortScan attack detected"]	54	34.120.177.193	'_Bot":{"adaptive":0,"original":0}	6	0	192.168.56.132	2024-12-17T22:49:20.608947

DESIGN & IMPLEMENTATION

HIGH-LEVEL SYSTEM ARCHITECTURE



PROCESS WORKFLOW



Network

Apache Kafka

Flask Application

Prometheus Server

Grafana Server

Dashboard

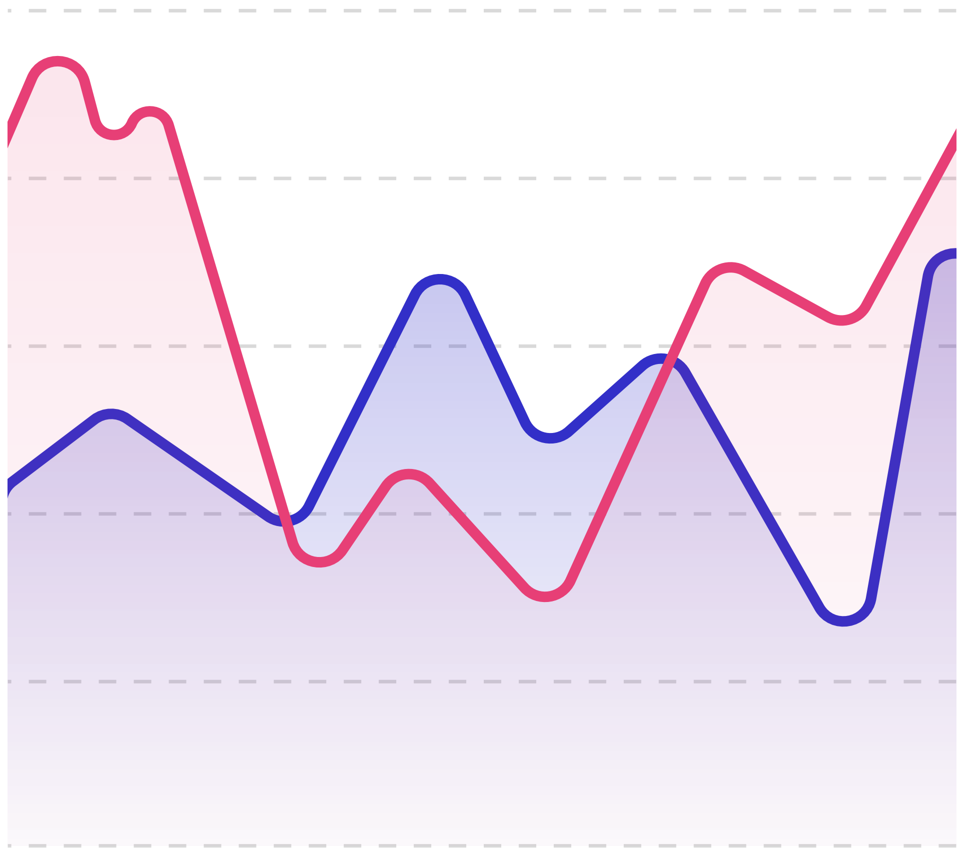
Tools & Technologies Used

ML Models

Random Forest,
Decision Tree

Streaming

Apache Kafka
(Producer & Consumer)



Dashboard and Monitoring

Prometheus for Monitoring + Grafana Dashboard

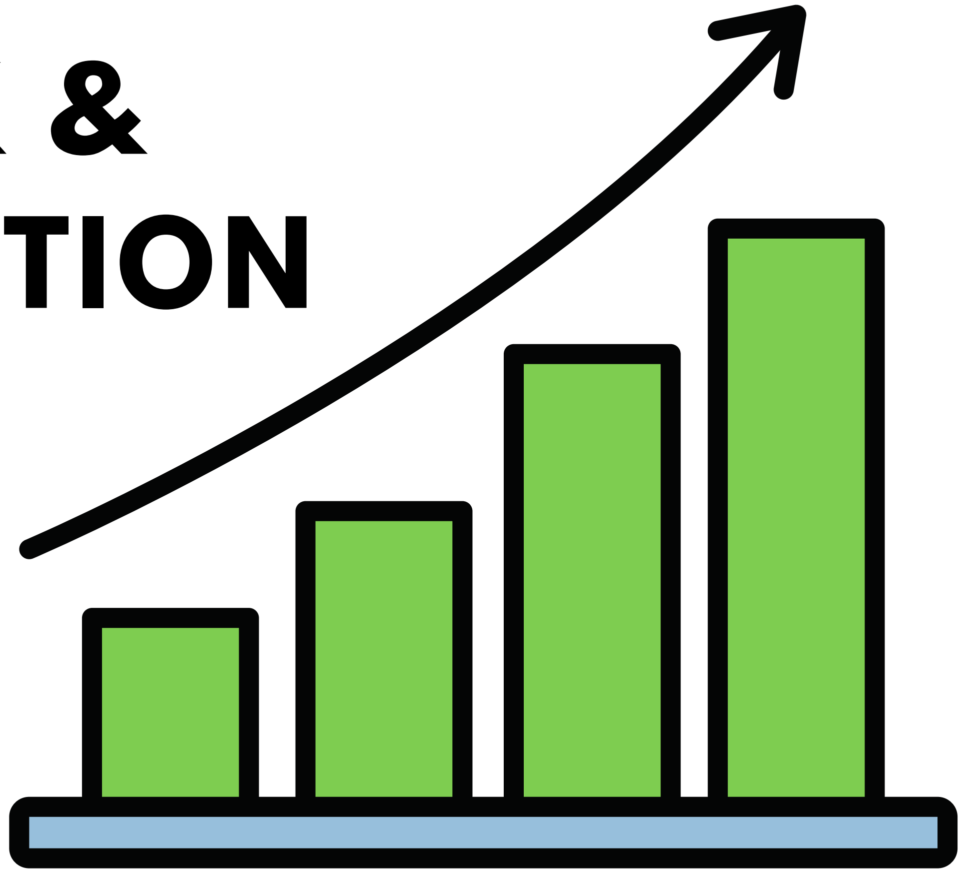
Programming

Python (Scapy,
joblib, Flask, pandas)

Deployment

Docker Compose

FUTURE WORK & COMMERCIALIZATION



Future Work

Expanded Detection Capabilities

- Add support for encrypted traffic inspection
- Improve detection of insider threats, APT attacks, and fileless malware

Smart Retraining Enhancements

- Implement incremental learning
- Enable user feedback loop for supervised labeling

Scalability & Distribution

- Multi-node Kafka setup for distributed networks
- Cross platform compatibility

Security & Access Control

- Role-based access to dashboard
- Secure REST API with OAuth2 or JWT

Commercialization

Target Market

- Small/Medium Enterprises (SMEs)
- Academic Institutions
- Managed Security Service Providers (MSSPs)
- Startups needing budget-friendly IDS

Business Models

- Open-source core + paid add-ons
- B2B Consulting – offer integration/custom deployment packages

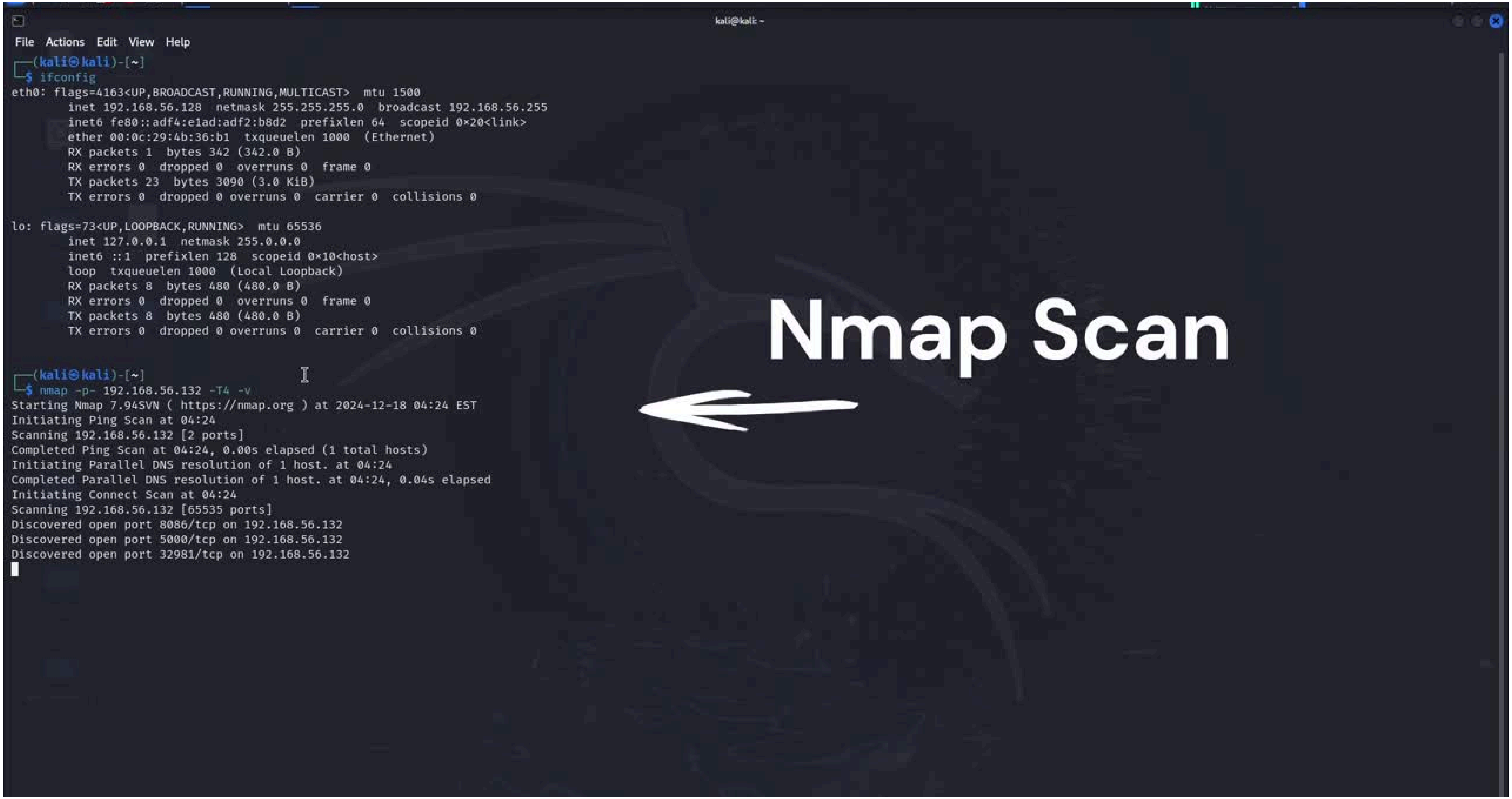
Why It's Market-Ready

- Lightweight & modular
- Dockerized for easy deployment
- Real-time + adaptive = rare combo in SME-level IDS tools
- Prometheus integration = easy to manage at scale

Productization

- Build installer / GUI
- Package as a plug-and-play appliance
- Obtain security certification (ISO 27001-compatible design)

Demo



```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.128  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::adf4:e1ad:adf2:b8d2  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:4b:36:b1  txqueuelen 1000  (Ethernet)
        RX packets 1  bytes 342 (342.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 23  bytes 3090 (3.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 8  bytes 480 (480.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8  bytes 480 (480.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0


┌──(kali㉿kali)-[~]
└─$ nmap -p- 192.168.56.132 -T4 -v
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-18 04:24 EST
Initiating Ping Scan at 04:24
Scanning 192.168.56.132 [2 ports]
Completed Ping Scan at 04:24, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 04:24
Completed Parallel DNS resolution of 1 host. at 04:24, 0.04s elapsed
Initiating Connect Scan at 04:24
Scanning 192.168.56.132 [65535 ports]
Discovered open port 8086/tcp on 192.168.56.132
Discovered open port 5000/tcp on 192.168.56.132
Discovered open port 32981/tcp on 192.168.56.132
```

ANY QUESTIONS?

THANK YOU.