A Project Report On

**INTRUSION DETECTION SYSTEM**

Submitted in partial fulfillment of the requirement for the award of the degree

MASTER OF SCIENCE (CS & CL)

from

Marwadi University

Academic Year 2025 – 26

**Ananthagiri sindhura (92400565071)**

**Abin Philip Varghese (92400565024)**

**Internal Guide**

**Prof. Dr. Pankaj Mudholkar**



Rajkot-Morbi Road, At &PO: Gauridad, Rajkot 360 003. Gujarat. India.



**Faculty of Computer Applications (FoCA)**

****

**This is to certify that the project work entitled**

**INTRUSION DETECTION SYSTEM**

**submitted in partial fulfillment of the requirement for**

**the award of the degree of**

**Master of Science (CS & CL)**

**of the**

**Marwadi University**

**is a result of the bonafide work carried out by**

**Ananthagiri sindhura (92400565071)**

**Abin Philip Varghese (92400565024)**

**during the academic year 2025 – 2026**

**Faculty Guide HOD Dean**

**DECLARATION**

We hereby declare that this project work entitled Intrusion Detection System is a record done by us.

We also declare that the matter embodied in this project is genuine work done by us and has not been submitted whether to this University or to any other University / Institute for the fulfillment of the requirement of any course of study.

Place:

Date:

Ananthagiri sindhura (92400565071) Signature:\_\_\_\_\_\_\_\_\_\_\_

Abin Philip Varghese (92400565024) Signature:\_\_\_\_\_\_\_\_\_\_\_

**ACKNOWLEDGEMENT**

It is indeed a great pleasure to express our thanks and gratitude to all those who helped us. No serious and lasting achievement or success one can ever achieve without the help of friendly guidance and co-operation of so many people involved in the work.

We are very thankful to our guide **Dr. Pankaj Mudholkar,** the person who makes us to follow the right steps during our project work. We express our deep sense of gratitude to for his guidance, suggestions and expertise at every stage. A part from that his/her valuable and expertise suggestion during documentation of our report indeed help us a lot.

Thanks to our friend and colleague who have been a source of inspiration and motivation that helped to us during our project work.

We are heartily thankful to the Dean of our department **Prof (Dr.) R. Sridaran** sir and HoD **Dr. Sunil Bajeja** sir for giving us an opportunity to work over this project and for their end-less and great support to all other people who directly or indirectly supported and help us to fulfil our task.

Ananthagiri sindhura (92400565071) Signature:\_\_\_\_\_\_\_\_\_\_\_

Abin Philip Varghese (92400565024) Signature:\_\_\_\_\_\_\_\_\_\_\_

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Chapters** | **Particulars** | **Page No.** |
| 1 | **Introduction**   * 1. Objective of the New System   2. Problem Definition   3. Core Components   4. Project Profile   5. Assumptions and Constraints   6. Advantages and Limitations of the Proposed System | 8-18 |
| 2 | **Requirement Determination & Analysis**   * 1. Requirement Determination   2. Targeted Users   3. Details of tools and techniques used / implemented   4. Advantages and Limitations of the used security tools | 19-23 |
| 3 | **System Design**   * 1. Flow chart   2. Dataset details | 24-25 |
| 4 | **Development**  4.1 Script details / Source code  4.2. Screen Shots / UI Design of simulation (if applicable)  4.3. Test reports | 26-36 |
| 5 | **Proposed Enhancements** | 37-48 |
| 6 | **Conclusion** | 39 |
| 7 | **Bibliography** | 40 |

**Table index**

|  |  |  |  |
| --- | --- | --- | --- |
| Table No. | | Title | Page No. |
| Table 1.3.1 | | Comparison of core components | 16 |
| Table 2.2.1 | Primary target users | | 20 |
| Table 2.3.1 | Tools | | 21 |
| Table 2.4.1 | Advantages of security tools | | 22 |
| Table 2.4.2 | Limitations of security tools | | 23 |
| Table 3.2.1 | Datasets | | 25 |
| Table 4.3.1 | Test reports | | 36 |

**Figure Index**

|  |  |  |
| --- | --- | --- |
| **Fig No.** | **Title** | **Page No.** |
| Fig 1.1 | Network based IDS | 9 |
| Fig 1.2 | Host based IDS | 10 |
| Fig 1.3 | Centralized control | 12 |
| Fig 1.4 | Distributed control strategy | 12 |
| Fig 1.5 | Fully distributed control | 13 |
| Fig 3.1.1 | Flow chart | 24 |
| Fig 4.1.1 | Source code | 26 |
| Fig 4.1.2 | Premium dashboard | 27 |
| Fig 4.1.3 | Alert system | 28 |
| Fig 4.1.4 | Alert system output | 28 |
| Fig 4.1.5 | Network capture | 29 |
| Fig 4.1.6 | Network capture output | 29 |
| Fig 4.1.7 | Logging | 30 |
| Fig 4.1.8 | Logging output | 30 |
| Fig 4.1.9 | ML Detection | 31 |
| Fig 4.1.10 | ML Detection output | 31 |
| Fig 4.1.11 | Threat intelligence | 32 |
| Fig 4.1.12 | Threat intelligence output | 32 |
| Fig 4.2.1 | Screen shots | 33 |
| Fig 4.2.2 | Deep detection login | 33 |
| Fig 4.2.3 | Deep detection dashboard | 34 |
| Fig 4.2.4 | Deep detection live monitoring | 34 |
| Fig 4.2.5 | Deep detection alerts output | 35 |

**Chapter 1**

**Introduction**

**Intrusion Detection System (IDS)**

An Intrusion Detection System (IDS) is a cybersecurity mechanism designed to monitor, detect, and alert administrators about unauthorized or suspicious activities occurring within a computer system or network. It acts as a digital watchdog, helping organizations identify potential threats before they cause harm.

However, we must differentiate between IDS and IPS (Intrusion Prevention System). The first one is a software that automates the process of intrusion detection; the latter is an intrusion prevention software, which aims to prevent possible attacks. One therefore works in a reactive and informative way, while the IPS reduces the risk of compromising an environment.

**Key Features of IDS:**

* Monitoring: Continuously observes network traffic or system logs.
* Detection: Identifies anomalies, known attack patterns, or policy violations.
* Alerting: Sends real-time notifications to security teams for quick response.
* Logging: Records events for forensic analysis and future reference.

**Types of IDS:**

1. Network-Based IDSs

The majority of commercial intrusion detection systems are network-based. These IDSs detect attacks by capturing and analyzing network packets. Listening on a network segment or switch, one network-based IDS can monitor the network traffic affecting multiple hosts that are connected to the network segment, thereby protecting those hosts.

Network-based IDSs often consist of a set of single-purpose sensors or hosts placed at various points in a network. These units monitor

network traffic, performing local analysis of that traffic and reporting attacks to a central management console. As the sensors are limited to running the IDS, they can be more easily secured against attack. Many of these sensors are designed to run in “stealth” mode, in order to make it more difficult for an attacker to determine their presence and location.

|  |  |
| --- | --- |
|  |  |
|  |  |

Advantages of Network-Based IDSs:

|  |  |
| --- | --- |
|  |  |
|  |  |

* A few well-placed network-based IDSs can monitor a large network.
* The deployment of network-based IDSs has little impact upon an existing network. Network-based IDSs are usually passive devices that listen on a network wire without
* interfering with the normal operation of a network. Thus, it is usually easy to retrofit a network to include network-based IDSs with minimal effort.
* Network-based IDSs can be made very secure against attack and even made invisible to many attackers.

Disadvantages of Network-Based IDSs:

* Network-based IDSs may have difficulty processing all packets in a large or busy network and, therefore, may fail to recognize an attack launched during periods of high traffic. Some vendors are attempting to solve this problem by implementing IDSs completely in hardware, which is much faster. The need to analyze packets quickly also forces vendors to both detect fewer attacks and also detect attacks with as little computing resource as possible which can reduce detection effectiveness.
* Network-based IDSs cannot analyze encrypted information. This problem is increasing as more organizations (and attackers) use virtual private networks.

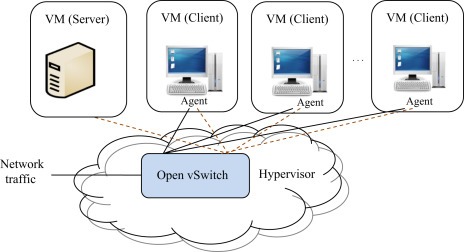


Fig:1.1 (Network based IDS)

2. Host-Based IDSs

Host-based IDSs operate on information collected from within an individual computer system. (Note that application-based IDSs are actually a subset of host-based IDSs.) This vantage point allows host-based IDSs to analyze activities with great reliability and precision, determining exactly which processes and users are involved in a particular attack on the operating system. Furthermore, unlike network -based IDSs, host-based IDSs can “see” the outcome of an attempted attack, as they can directly access and monitor the data files and system processes usually targeted by attacks.

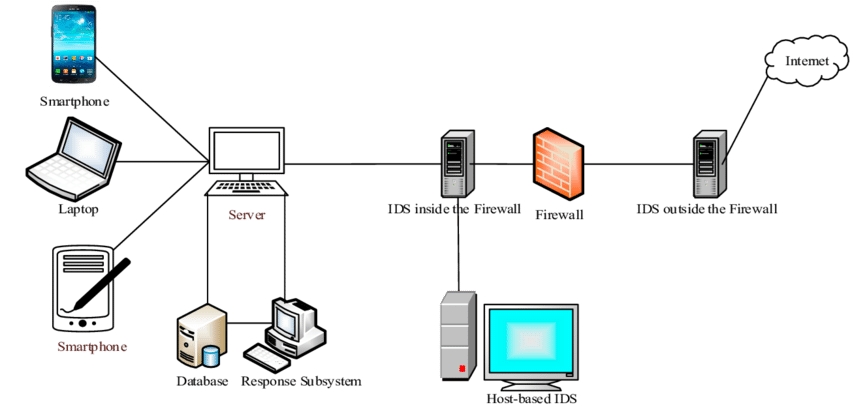


Fig 1.2 (Host-based IDS)

Advantages:

* Host-based IDSs, with their ability to monitor events local to a host, can detect attacks that cannot be seen by a network-based IDS.
* Host-based IDSs can often operate in an environment in which network traffic is encrypted, when the host-based information sources are generated before data is encrypted and/or after the data is decrypted at the destination host
* Host-based IDSs are unaffected by switched networks.
* When Host-based IDSs operate on OS audit trails, they can help detect Trojan Horse or other attacks that involve software integrity breaches. These appear as inconsistencies in process execution.

Disadvantages:

* Host-based IDSs are harder to manage, as information must be conFigd and managed for every host monitored.
* Since at least the information sources (and sometimes part of the analysis engines) for host-based IDSs reside on the host targeted by attacks, the IDS may be attacked and disabled as part of the attack.
* Host-based IDSs are not well suited for detecting network scans or other such surveillance that targets an entire network, because the IDS only sees those network packets received by its host.
* Host-based IDSs use the computing resources of the hosts they are monitoring, therefore inflicting a performance cost on the monitored systems.

3. Application-Based IDSs

Application-based IDSs are a special subset of host-based IDSs that analyze the events transpiring within a software application. The most common information sources used by application-based IDSs are the application’s transaction log files.

The ability to interface with the application directly, with significant domain or application-specific knowledge included in the analysis engine, allows application-based IDSs to detect suspicious behavior due to authorized

users exceeding their authorization. This is because such problems are more likely to appear in the interaction between the user, the data, and the application.

Advantages:

* Application-based IDSs can monitor the interaction between user and application, which often allows them to trace unauthorized activity to individual users.
* Application-based IDSs can often work in encrypted environments, since they interface with the application at transaction endpoints, where information is presented to users in unencrypted form.

Disadvantages:

* Application-based IDSs may be more vulnerable than host-based IDSs to attacks as the applications logs are not as well-protected as the operating system audit trails used for host-based IDSs.
* As Application-based IDSs often monitor events at the user level of abstraction, they usually cannot detect Trojan Horse or other such software tampering attacks. Therefore, it is advisable to use an application-based IDS in combination with Host-based and/or Network-

based IDSs.

**Control Strategy:**

Control Strategy describes how the elements of an IDS is controlled, and furthermore, how the input and output of the IDS is managed.

1.Centralized

Under centralized control strategies, all monitoring, detection and reporting is controlled directly from a central location.

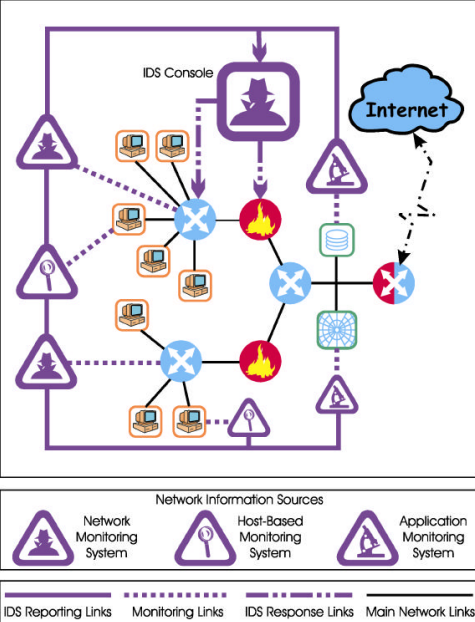


Fig 1.3: Centralized Control

2.Partially Distributed:

Monitoring and detection is controlled from a local control

node, with hierarchical reporting to one or more central location(s).

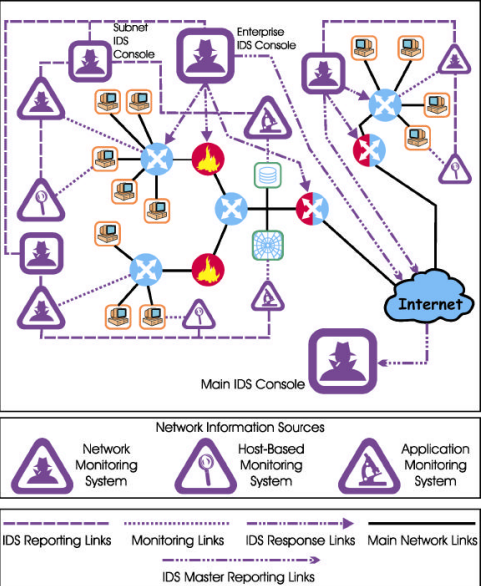


Fig 1.4: Distributed Control Strategy

**3.**Fully Distributed:

Monitoring and detection is done using an agent-based approach, where response decisions are made at the point of analysis.

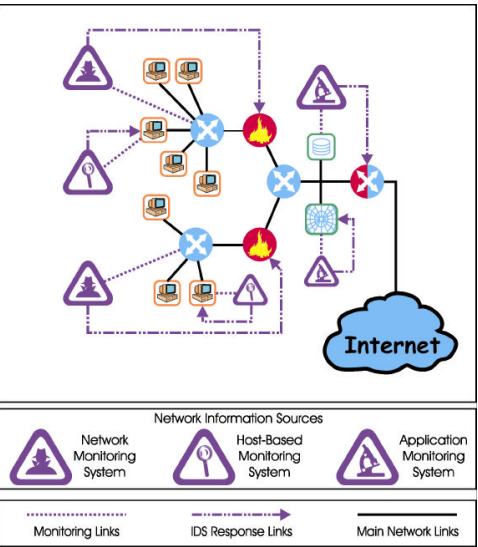


Fig 1.5: Fully Distributed (Agent-Based) Control

Detection Techniques:

* Signature-Based: Matches known attack patterns (like antivirus).
* Anomaly-Based: Flags unusual behavior that deviates from normal usage.

Importance of IDS:

* Enhances security posture by detecting threats early.
* Helps prevent data breaches and system compromise.
* Supports compliance with cybersecurity regulations.

**1.1 Objective of the New System**

The **primary objective** of an **Intrusion Detection System (IDS)** is to safeguard digital infrastructure by continuously monitoring network traffic and system activities to detect suspicious behavior that may indicate a security breach, malware infection, or unauthorized access attempt. In today’s threat landscape, where cyberattacks are increasingly sophisticated, IDS plays a vital role in early detection and response.

Key Objectives:

**1.Real-Time Detection of Known and Unknown Attacks**  
IDS is designed to identify both signature-based threats (known attack patterns) and anomaly-based threats (unusual behavior). This ensures that even zero-day attacks or novel intrusion techniques can be flagged promptly.

**2.Minimizing False Positives and False Negatives**  
A well-tuned IDS aims to reduce false alarms (false positives) that can overwhelm administrators, while also ensuring that genuine threats are not missed (false negatives). This balance improves trust and efficiency in the system.

**3.Providing Actionable Alerts to Administrators**  
When a potential threat is detected, the IDS generates detailed alerts that include the nature of the threat, affected systems, and recommended actions. This enables quick decision-making and incident response.

**4.Enhancing Overall Network Security Posture**  
By continuously monitoring and analyzing traffic, IDS helps organizations maintain visibility into their network, identify vulnerabilities, and strengthen their defense mechanisms. It complements other security tools like firewalls and antivirus software.

**5.Supporting Forensic Analysis and Compliance**  
IDS logs and reports can be used for post-incident investigations and to meet regulatory requirements related to data protection and cybersecurity.

**6.Adaptive Learning and Scalability**  
Advanced IDS solutions incorporate machine learning to adapt to evolving threats and can scale across large enterprise networks without compromising performance.

**1.2 Problem Definition**

In digital landscape, organizations face an ever-growing array of cyber threats that are more sophisticated, stealthy, and damaging than ever before. Traditional security measures such as firewalls and antivirus software, while essential, are no longer sufficient to detect and respond to advanced attacks in real time.

Key Challenges in Modern Networks:

1. **Malware, Phishing, and Denial-of-Service (DoS) Attacks**  
   These attacks can cripple systems, steal sensitive data, and disrupt operations. Attackers often use deceptive techniques to bypass conventional security tools.
2. **Insider Threats**  
   Employees or trusted users may intentionally or unintentionally compromise systems. These threats are difficult to detect using perimeter-based defenses.
3. **Lack of Real-Time Visibility**  
   Without continuous monitoring, organizations may remain unaware of ongoing intrusions until damage is done. Delayed detection leads to increased recovery costs and reputational harm.
4. **Zero-Day and Polymorphic Attacks**  
   These are attacks that exploit unknown vulnerabilities or change their code to evade detection. Signature-based tools cannot identify them without prior knowledge.

Core Problem:

The absence of a **robust, intelligent, and adaptive detection mechanism** leaves networks vulnerable to exploitation. Static defenses cannot keep pace with dynamic threats, and manual monitoring is impractical for large-scale environments.

Why IDS Is Needed:

An **Intrusion Detection System (IDS)** addresses these challenges by:

* **Analyzing traffic intelligently** using rule-based and anomaly-based methods.
* **Detecting both known and unknown threats** through pattern recognition and behavioral analysis.
* **Providing actionable alerts** to administrators without disrupting normal operations.
* **Enhancing situational awareness** and enabling faster incident response.

In summary, the problem lies in the growing complexity of cyber threats and the limitations of traditional security tools. IDS offers a proactive solution to detect intrusions before they escalate, making it a critical component of modern cybersecurity infrastructure.

**1.3 Core Components**

An Intrusion Detection System (IDS) is composed of several interdependent components that work together to monitor, analyze, and report suspicious activities. Each component plays a critical role in ensuring accurate detection and timely response.

|  |  |  |
| --- | --- | --- |
| **Component** | | **Function** |
| Sensor/Agent | | Captures raw data from network traffic or host  system logs. |
| Detection Engine | | Analyzes data using signature-based or anomaly-  based techniques. |
| Signature Database | | Stores known attack patterns for comparison during  analysis. |
| Alert System | | Generates real-time notifications when suspicious  activity is detected. |
| Logging Module | Records events and alerts for forensic analysis and  auditing. | |

|  |  |
| --- | --- |
| Management Console | Provides a user interface for configuration,  monitoring, and reporting. |

Table: 1.3.1 Core components

These components collectively enable the IDS to detect intrusions efficiently, reduce false alarms, and support security teams in responding to threats effectively.

**1.4 Project Profile**

Technology Stack

* Programming Language: Python (for scripting and ML)
* Tools: Snort (signature-based IDS), Wireshark (packet analysis), optional ML libraries (Scikit-learn, TensorFlow)

Scope of the Project

* Real-time monitoring of incoming and outgoing network traffic.
* Detection of both known (signature-based) and unknown (anomaly-based) threats.
* Logging of suspicious activities for future analysis.
* Alerting mechanisms to notify administrators instantly.
* Optional integration with Security Information and Event Management (SIEM) tools for centralized monitoring.

Target Users

* Network administrators
* IT security teams
* Enterprise-level infrastructure managers

Development Phases

1. Requirement Analysis – Identify system needs and threat models.
2. System Design – Define architecture, data flow, and detection logic.
3. Implementation – Develop and conFig IDS components.
4. Testing and Evaluation – Validate detection accuracy and performance.
5. Deployment and Maintenance – Install in live environments and update regularly.

This profile outlines a comprehensive and scalable IDS solution tailored for modern enterprise networks.

**1.5 Assumptions and Constraints**

**Assumptions**

* The IDS will be deployed in a controlled and monitored network environment.
* Users interacting with the system possess basic cybersecurity knowledge.
* Network traffic is accessible and unencrypted for effective monitoring.
* The signature database is regularly updated to include the latest threats.

**Constraints**

* IDS is a passive system—it detects but does not prevent attacks.
* High network traffic may lead to performance degradation or delayed detection.
* Encrypted traffic (e.g., HTTPS) limits visibility into payloads, reducing detection accuracy.
* False positives may occur, requiring manual verification and tuning.
* IDS can be resource-intensive, especially in large-scale or high-speed networks.

Understanding these assumptions and constraints is essential for designing a realistic and effective IDS solution.

**1.6 Advantages and Limitations of the Proposed System**

**Advantages**

1. Early Threat Detection  
   IDS identifies suspicious activities before they escalate, allowing timely intervention.
2. Real-Time Alerts  
   Immediate notifications enable faster incident response and containment.
3. Improved Network Visibility  
   Continuous monitoring provides insights into traffic patterns and potential vulnerabilities.
4. Compliance Support  
   Helps organizations meet regulatory requirements by maintaining logs and audit trails.
5. Scalability and Customization  
   IDS can be tailored to specific environments and scaled across multiple systems.

**Limitations**

1. False Positives  
   Excessive alerts may overwhelm security teams and reduce system credibility.
2. No Prevention Capability  
   IDS only detects threats—it does not block or mitigate them directly.
3. Regular Updates Required  
   Signature databases and detection models must be updated frequently to remain effective.
4. Performance Impact  
   High traffic volumes can slow down detection and analysis processes.
5. Limited Visibility into Encrypted Traffic  
   IDS may not inspect encrypted payloads, reducing its effectiveness against certain threats.

**Chapter 2:**

**Requirement Determination & Analysis**

**2.1 Requirement Determination**

The development of an Intrusion Detection System (IDS) requires a clear understanding of both functional and non-functional requirements to ensure the system meets its intended goals.

Functional Requirements

1. Traffic Monitoring: The IDS must continuously monitor network packets or system logs.
2. Threat Detection: It should identify known attack signatures and anomalous behavior.
3. Alert Generation: Real-time alerts must be triggered upon detection of suspicious activity.
4. Logging and Reporting: All events should be logged for forensic analysis and compliance.
5. Rule Management: Administrators should be able to define and update detection rules.

Non-Functional Requirements

1. Performance: The system must operate efficiently under high traffic loads.
2. Scalability: It should support expansion across multiple nodes or networks.
3. Reliability: IDS must maintain consistent uptime and accurate detection.
4. Security: The IDS itself must be protected against tampering or exploitation.
5. Usability: A user-friendly interface should allow easy configuration and monitoring.

These requirements form the foundation for designing a robust and

effective IDS tailored to modern cybersecurity needs.

**2.2Targeted Users**

An IDS is designed to serve a range of users who are responsible for maintaining the security and integrity of digital systems.

**Primary Targeted Users**

Primary targeted users of Intrusion Detection Systems (IDS) are network security professionals, IT administrators, and organizations seeking to protect their networks and data from cyber threats and unauthorized access.

|  |  |
| --- | --- |
| **User Role** | **Responsibilities** |
| Network  Administrators | Monitor traffic, conFig IDS rules, respond to  alerts. |
| Security  Analysts | Investigate incidents, perform threat analysis, and fine-  Tune detection mechanisms |
| System  administrators | Ensure host-level security and integrate IDS with other tools |
| Compliance  officers | Use IDS logs to verify adherence to regulatory standards |
| Incident  Response  teams | Act on IDS alerts to contain and remediate threats |

Table: 2.2.1 Primary targeted users

**Why Are These Users Targeted**

* **Need for Real-Time Monitoring:**IDS offers continuous surveillance necessary for swift identification of threats.
* **Early Threat Detection**: Early alerts help organizations stop attacks

before they escalate, minimizing damage.

* **Compliance Requirements:**Many industries must deploy IDS to meet legal and regulatory standards.
* **Incident Response:**Security teams employ IDS to accelerate their response to detected threats and streamline forensic investigations.

Organizational Use Cases

* Enterprises: Protect sensitive data and intellectual property.
* Government Agencies: Secure critical infrastructure and classified networks.
* Educational Institutions: Monitor academic networks and prevent misuse.
* Healthcare Providers: Safeguard patient data and comply with HIPAA regulations.

The IDS must be adaptable to different user needs, offering both technical depth and operational simplicity.

* 1. **Details of Tools and Techniques Used / Implemented**

Modern IDS implementations leverage a combination of open-source tools and advanced detection techniques to ensure comprehensive threat coverage.

**Tools Used**

* Snort
* ManageEngine Event Log Analyzer & Log360

|  |  |
| --- | --- |
| **Tools** | **Purpose** |
| Snort | Signature-based detection using predefined rules. |
| Suricata | Multi-threaded IDS/IPS with protocol analysis and file extraction. |
| Zeek(bro) | Behavioral analysis and network traffic scripting. |
| Wireshark | Packet capture and manual inspection |
| Scikit-learn /  tensorFlow | Machine learning-based anomaly detection |

Table: 2.3.1 Tools

Techniques Implemented

* **Signature-Based Detection**: Matches traffic against known attack patterns.
* **Anomaly-Based Detection**: Flags deviations from normal behavior using statistical models.
* **Hybrid Detection**: Combines both methods for improved accuracy.
* **Protocol Analysis**: Examines application-layer protocols (e.g., HTTP, DNS) for misuse.

Machine Learning Models

* Decision Trees
* Random Forest
* Support Vector Machines (SVM)
* Neural Networks

These tools and techniques enable the IDS to detect a wide range of threats with high precision and adaptability.

**2.4 Advantages and Limitations of the Used Security Tools**

Each IDS tool offers unique strengths and faces certain limitations. Understanding these helps in selecting the right tool for specific environments.

**Advantages**

|  |  |
| --- | --- |
| Tool | Advantages |
| Snort | Lightweight, customizable, large community support |
| Suricata | High performance, multi-threading, deep packet inspection |
| Zeek | Flexible scripting, excellent for behavioral analysis |
| Wireshark | Powerful foe manual inspection and protocol-level debugging |
| ML Models | Capable of detecting unknown threats and adapting over time |

Table: 2.4.1 Advantages of used security tools

**Limitations**

|  |  |
| --- | --- |
| **Tools** | **Limitations** |
| Snort | High false positives, limited anomaly detection |
| Suricata | Resources-intensive, complex configuration |
| Zeek | Steep learning curve, not ideal for real-time alerting |
| Wireshark | Manual analysis only, not scalable for large networks |
| ML Models | Requires large datasets, risk of overfitting, high computational cost |

Table: 2.4.2 Limitations of used security tools

**Chapter 3**

**3.System Design**

The system design of an ids outlines how data flows through various modules-from traffic capture to threat detection and alerting. It integrates both signature-based and anomaly-based techniques to ensure comprehensive security coverage.

**3.1 Flow chart**

Normal data

Belongs to known attack class

New attack identified

Is testing threshold lies in normal threshold range

Project testing data into subspace representing attack behavior

Is testing threshold lies in normal threshold range

Calculate threshold for testing data

Project testing data into subspace representing normal behavior

Provide testing data vector

Fig 3.1.1 (F)

**Module Functions**

* Sensor: Captures real-time traffic using tools like tcpdump or Wireshark.
* Preprocessing: Cleans and formats data for analysis.
* Detection Engine: Applies rules (signatures) or ML models (anomaly detection).
* Classification: Labels threats and determines severity.
* Alerting: Sends notifications and logs incidents.

**3.2 Dataset Details**

**Purpose**

IDS datasets are used to train and evaluate detection models. They contain labelled examples of both normal and malicious behavior.

Recommended Datasets

|  |  |
| --- | --- |
| Dataset | Highlights |
| NSL-KDD | Improved version of KDD99; reduced redundancy and better evaluation metrics |
| CICIDS2017 | Realistic modern traffic; includes DoS, DDoS, Botnet, Brute Force, etc. |
| BoTNetIoT-L01 | IoT-focused; includes Mirai and Gafgyt botnet attacks |

|  |  |
| --- | --- |
| RPL-IDS | Behavior-based dataset for IoT networks using Contiki-NG simulation |

Table- 3.2.1 Datasets

**Dataset Features**

* Protocols: HTTP, FTP, SSH, DNS, etc.
* Attack Types: Brute Force, DoS, DDoS, Heartbleed, Botnet, Infiltration.
* Labels: Each record is tagged as benign or malicious.
* Format: CSV or PCAP files with features like IP addresses, ports, timestamps, packet sizes.

**Chapter 4**

**Development**

**4.1. Source code**

**a)Source code**

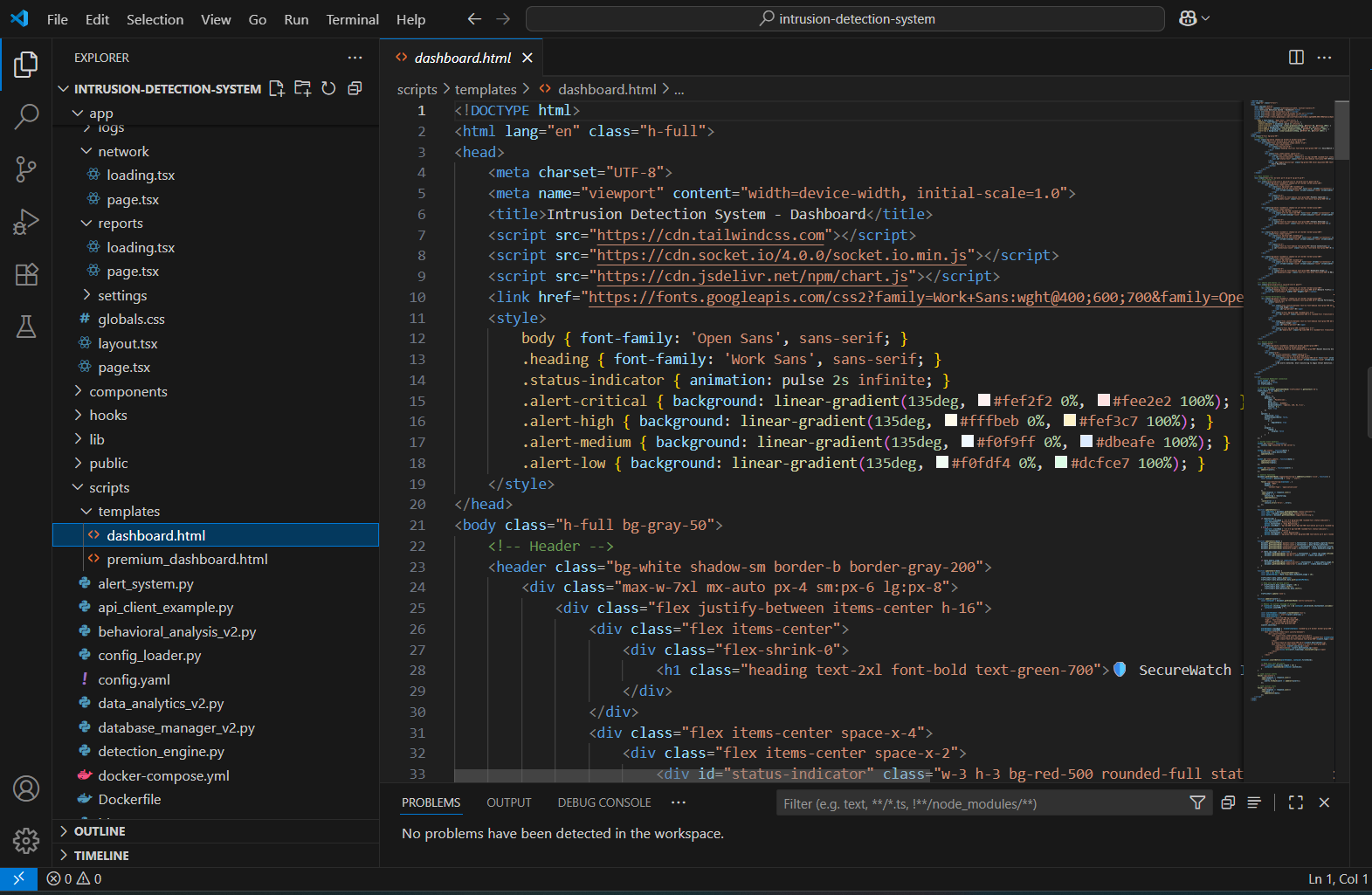


Fig:4.1.1 source code

dashboard.html Script Details

* **File Location**: scripts/templates/dashboard.html
* **Purpose**:  
  This template is for the primary dashboard view of the IDS, likely used for displaying real-time alerts and system statuses to users.

**b) Premium Dashboard**

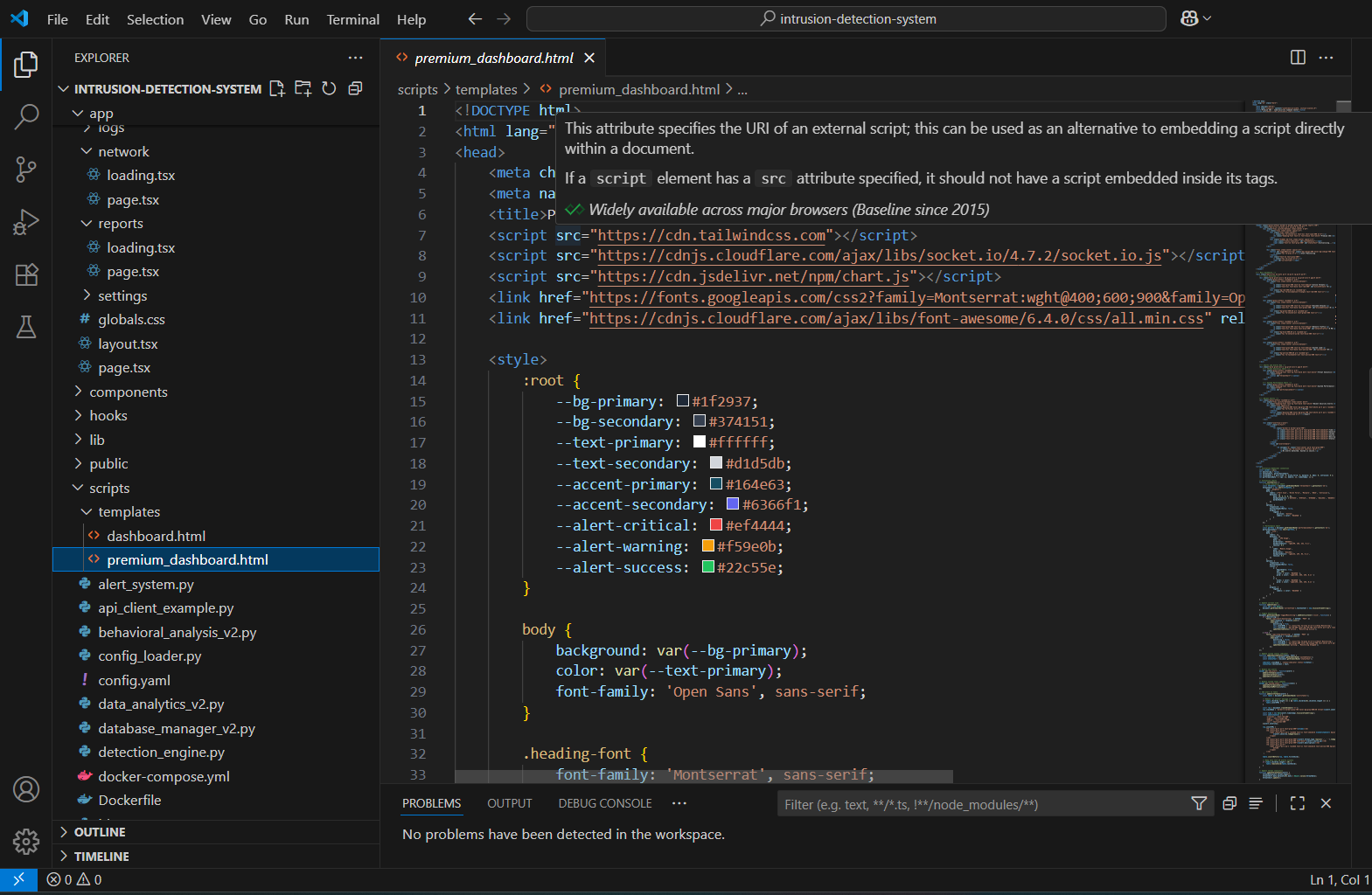


Fig: 4.1.2 premium dashboard

* **Purpose**  
  This is a premium or advanced view/dashboard, likely designed for enterprise users with enhanced customization and visuals.

**c) Alert System**

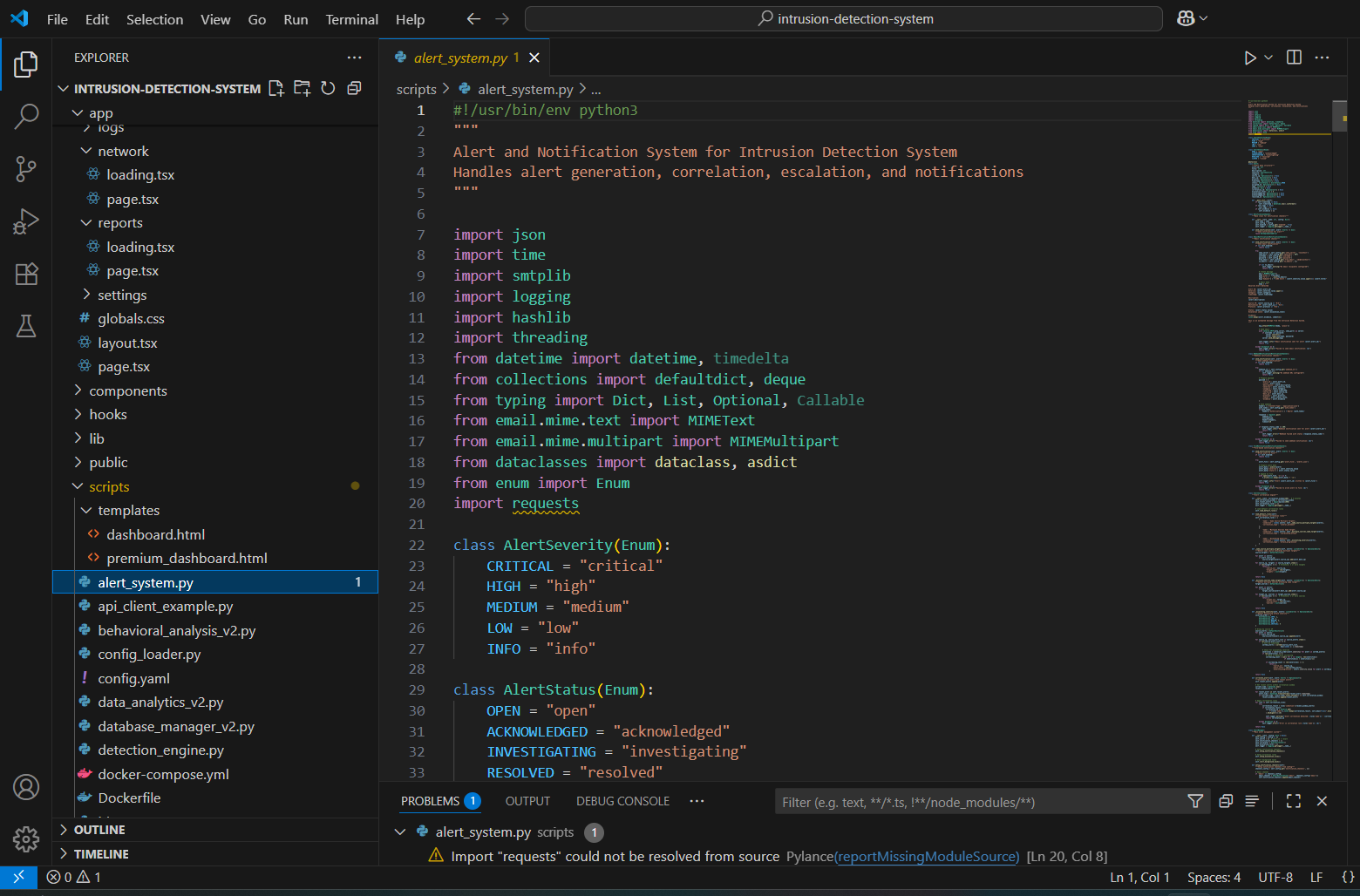


Fig:4.1.3 Alert system

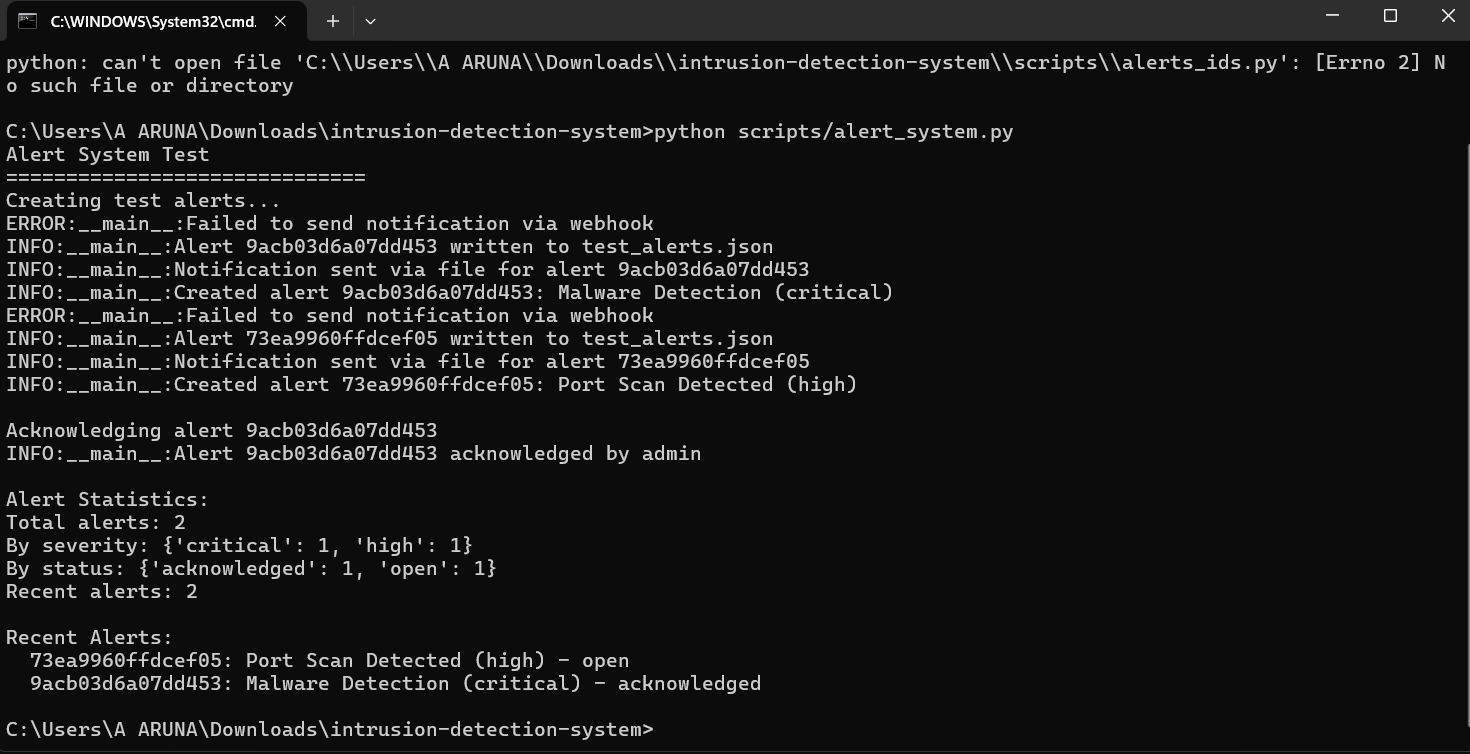
****

Fig:4.1.4 Alert system report

**Purpose**

* Manages the lifecycle of alerts: generation, escalation, correlation, notification, and resolution.
* Ensures that detected security incidents are properly flagged with a severity and status, and communicated to operators or automated response systems using various channels (like email, logs, or APIs).

**d)Network capture:**

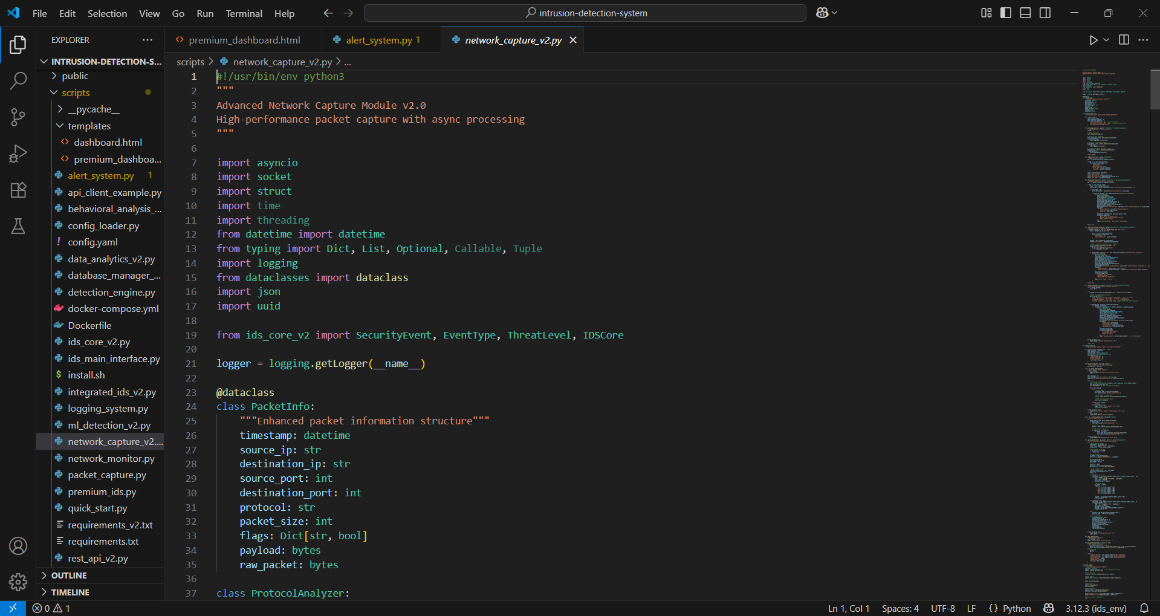
****

Fig:4.1.5 Network capture

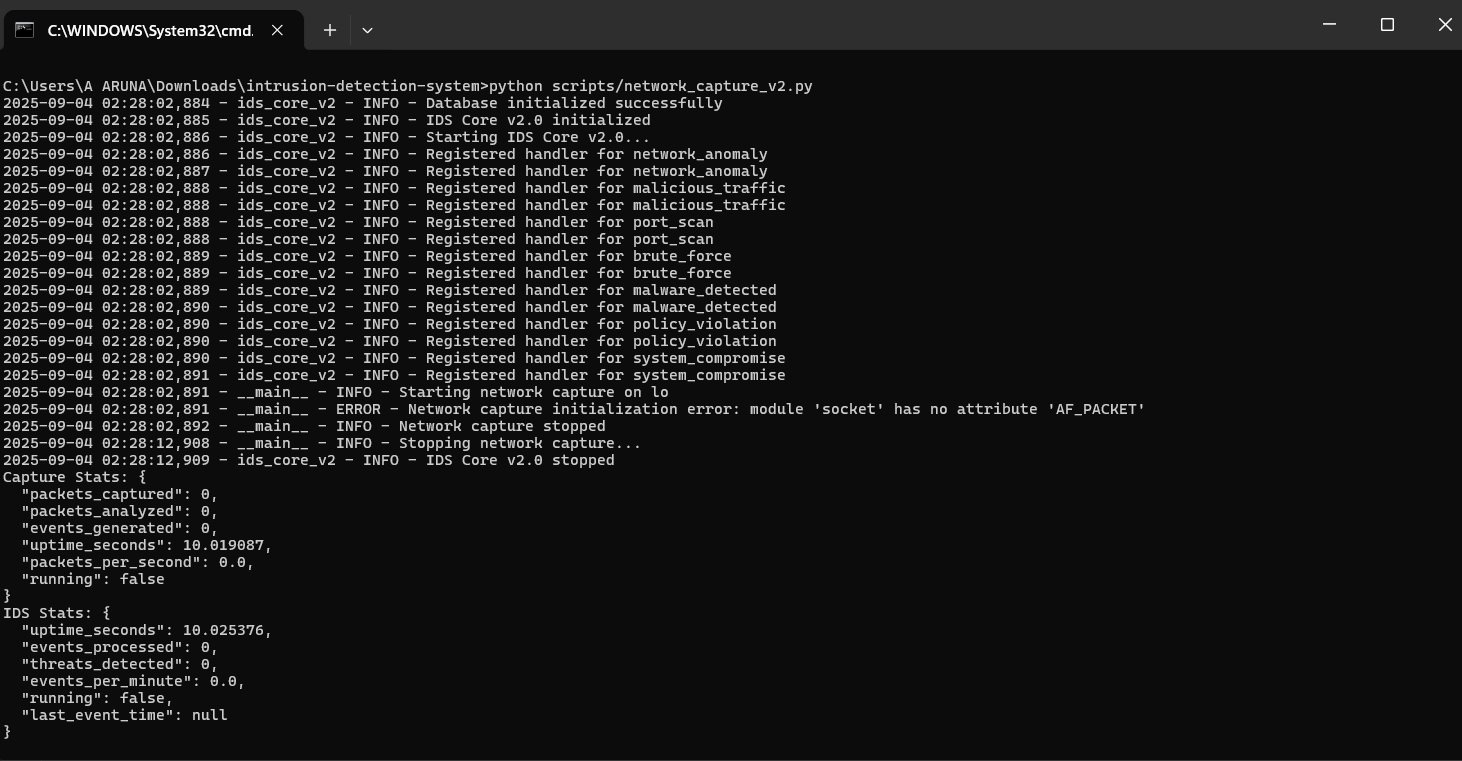


Fig:4.1.6 Network capture report

**Purpose:**

* Captures and analyzes network traffic in real-time to detect attacks.

**e) Logging:**

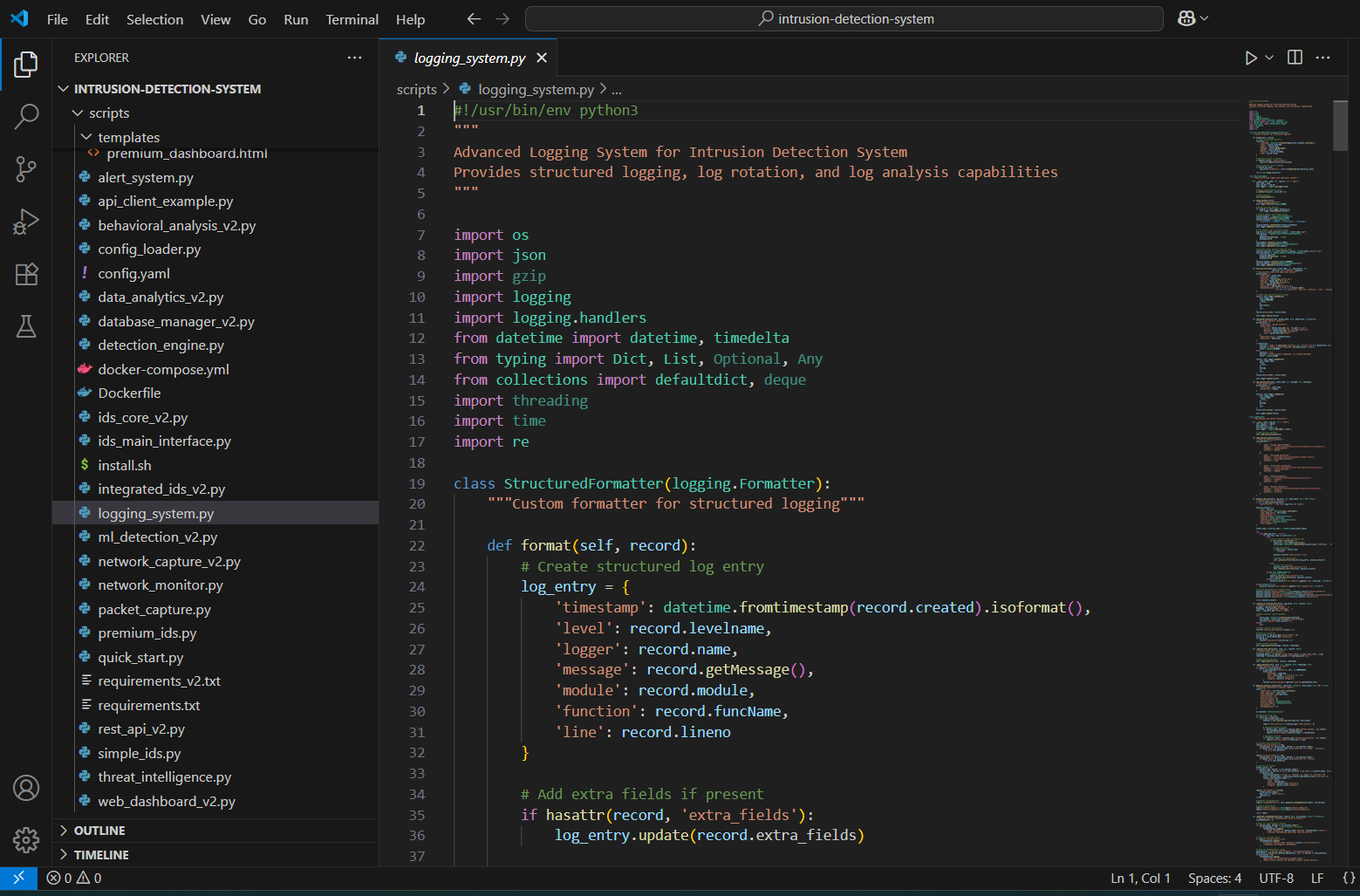


Fig:4.1.7 Logging system

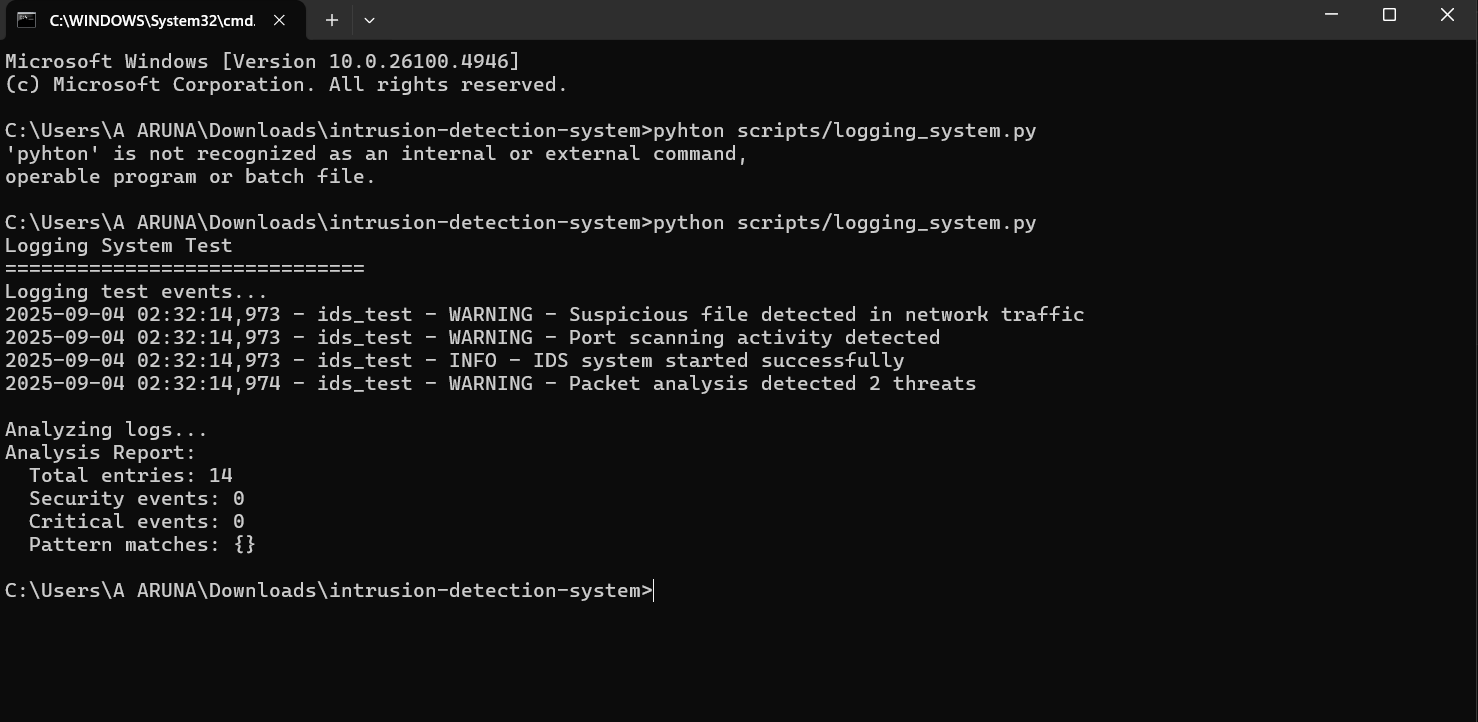
****

Fig:4.1.8 Logging system report

**Purpose:**

* To record suspicious activities, IDS status, and detected threats into logs.

**f) Machine learning detection(ml):**

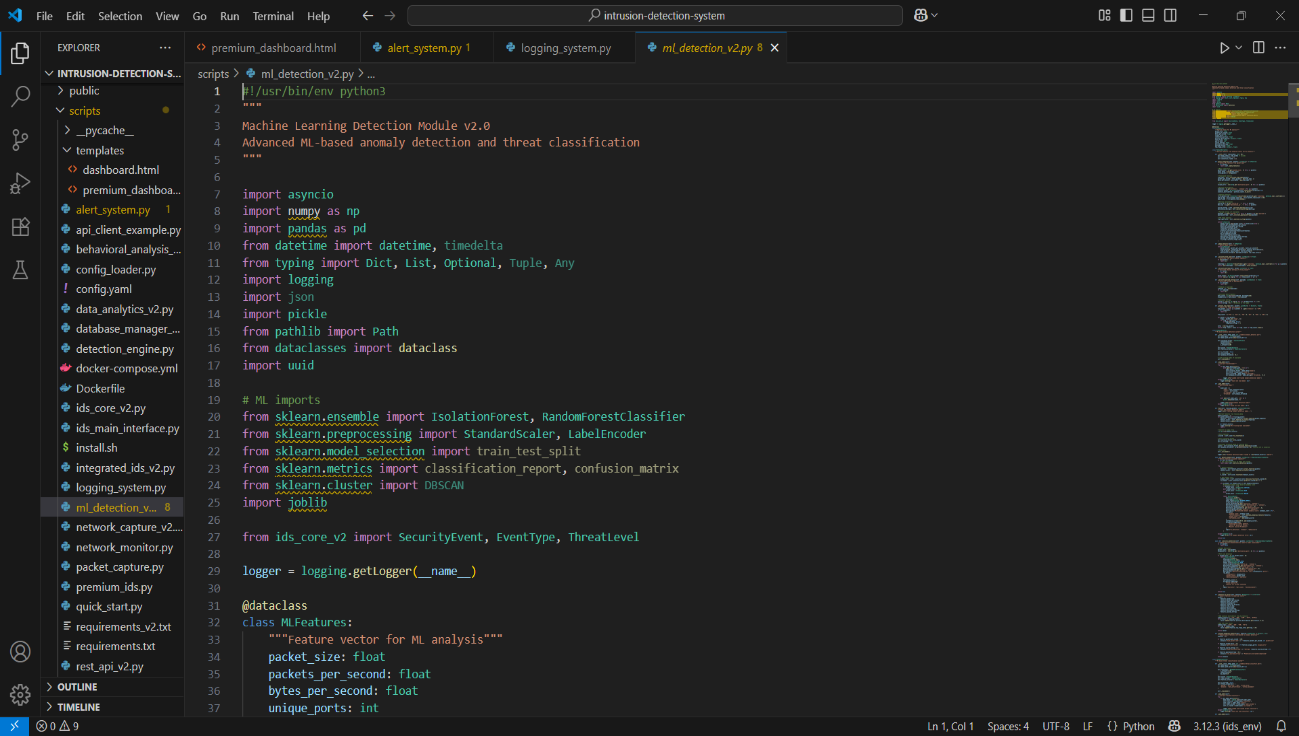
****

Fig:4.1.9 ML detection

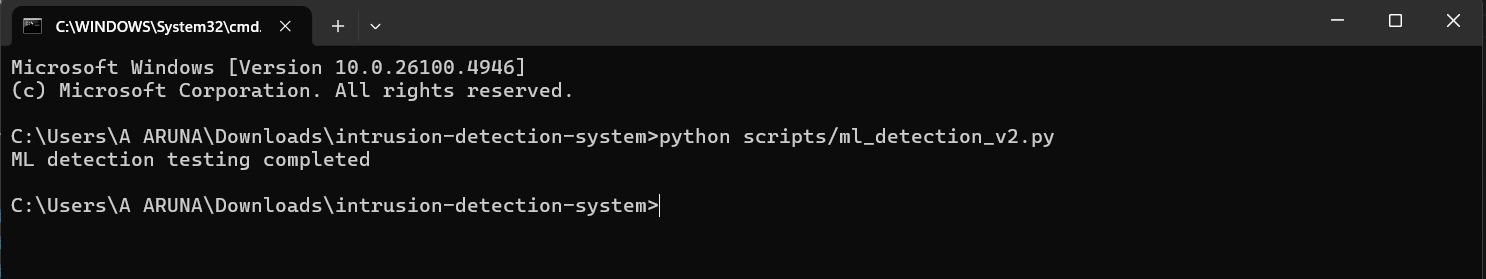


Fig:4.1.10 ML detection report

**Purpose:**

* Uses ML algorithms (IsolationForest, RandomForest, DBSCAN, etc.) to detect anomalies and classify threats.

**g) Threat intelligence:**

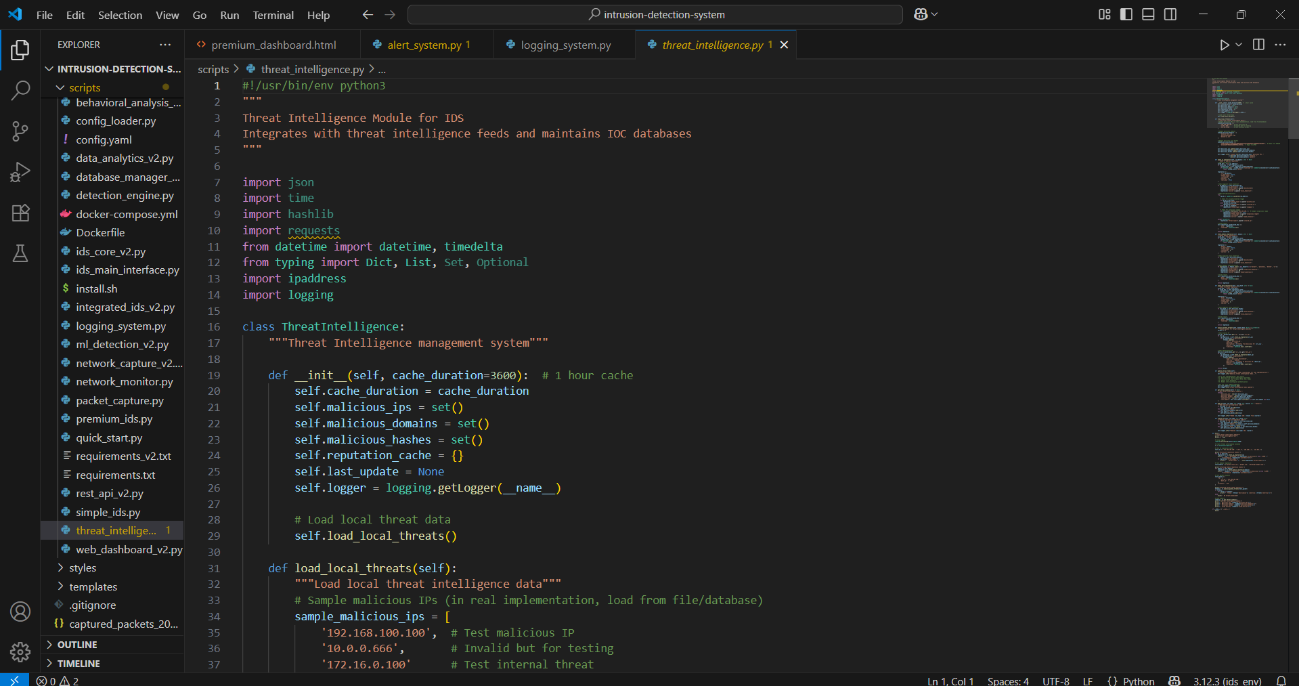
****

Fig:4.1.11 Threat intelligence

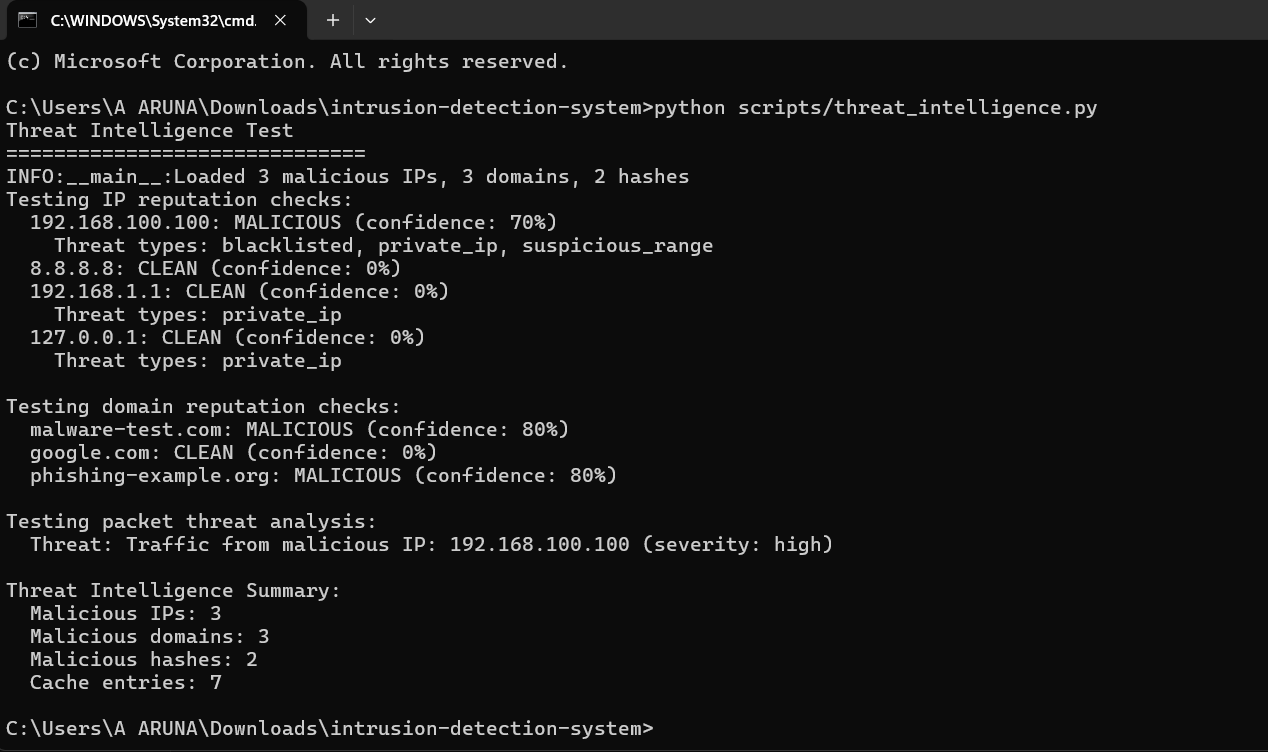
****

Fig:4.1.12 Threat intelligence report

**Purpose:**

* threat intelligence to detect and report malicious IPs, domains, and traffic.

**4.2. Screen Shots**

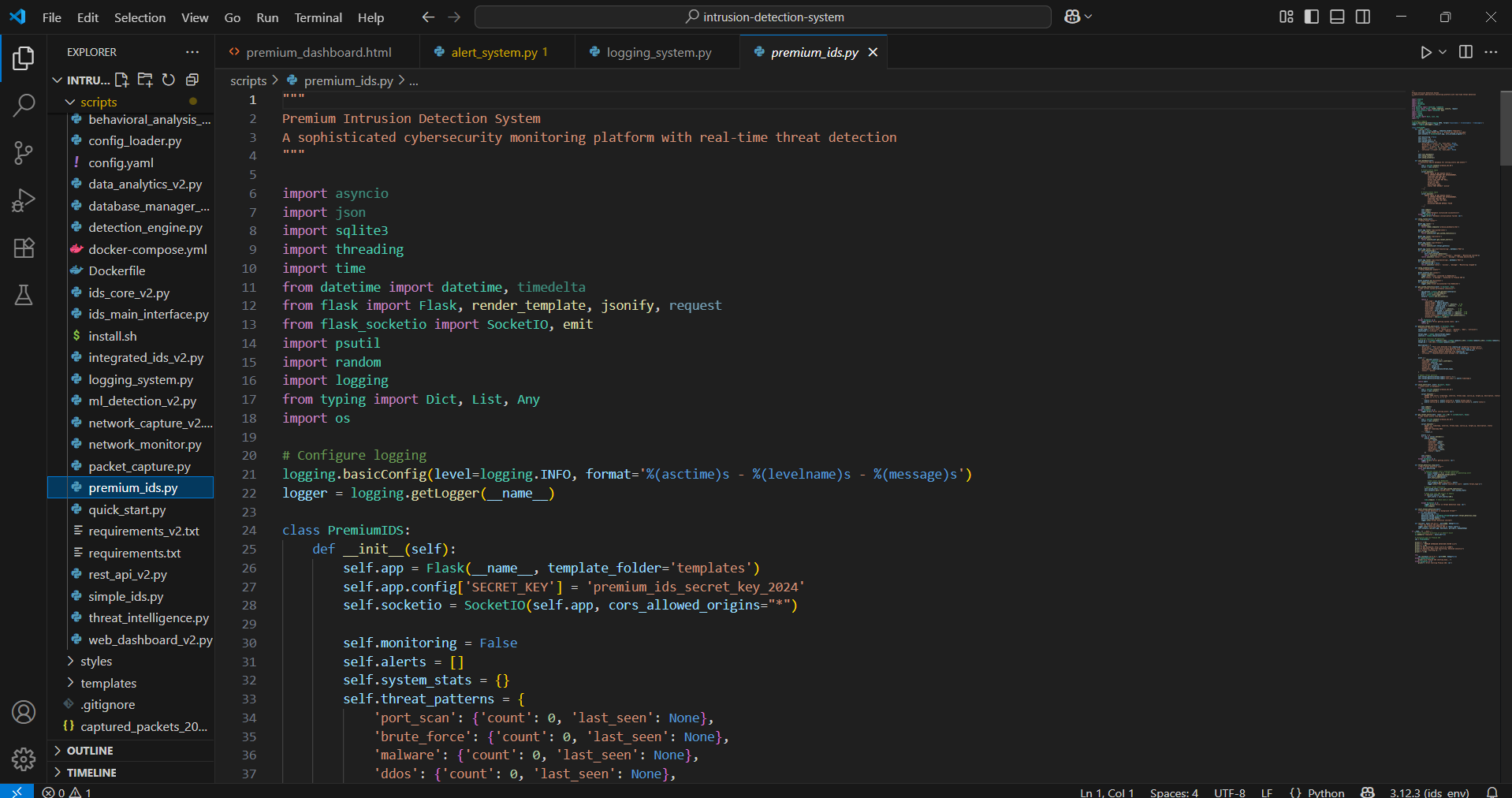
****

Fig:4.2.1 Premium IDS

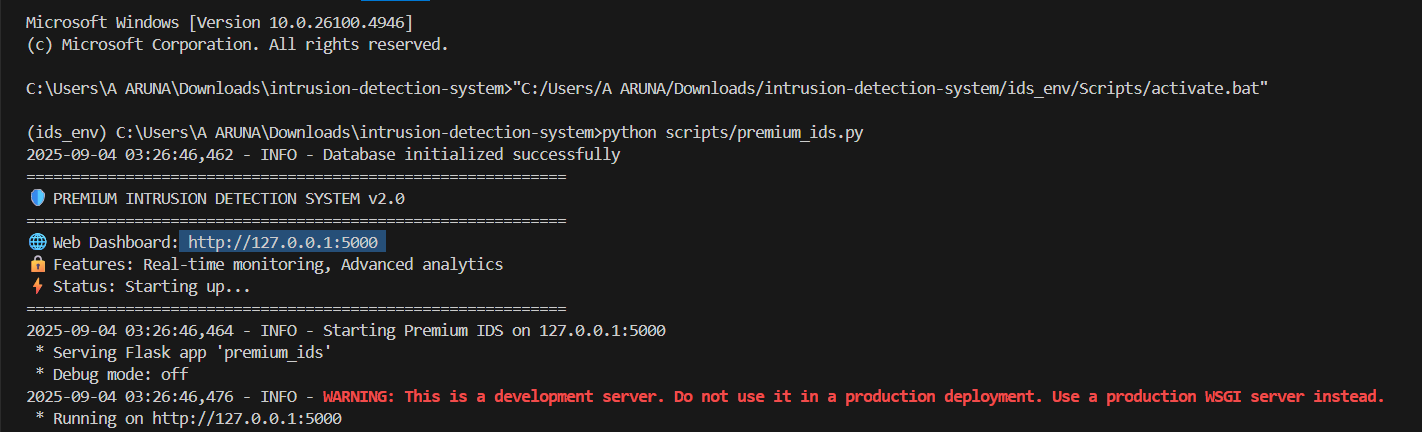


Fig:4.2.2 Premium IDS report

**Deep detect IDS:**

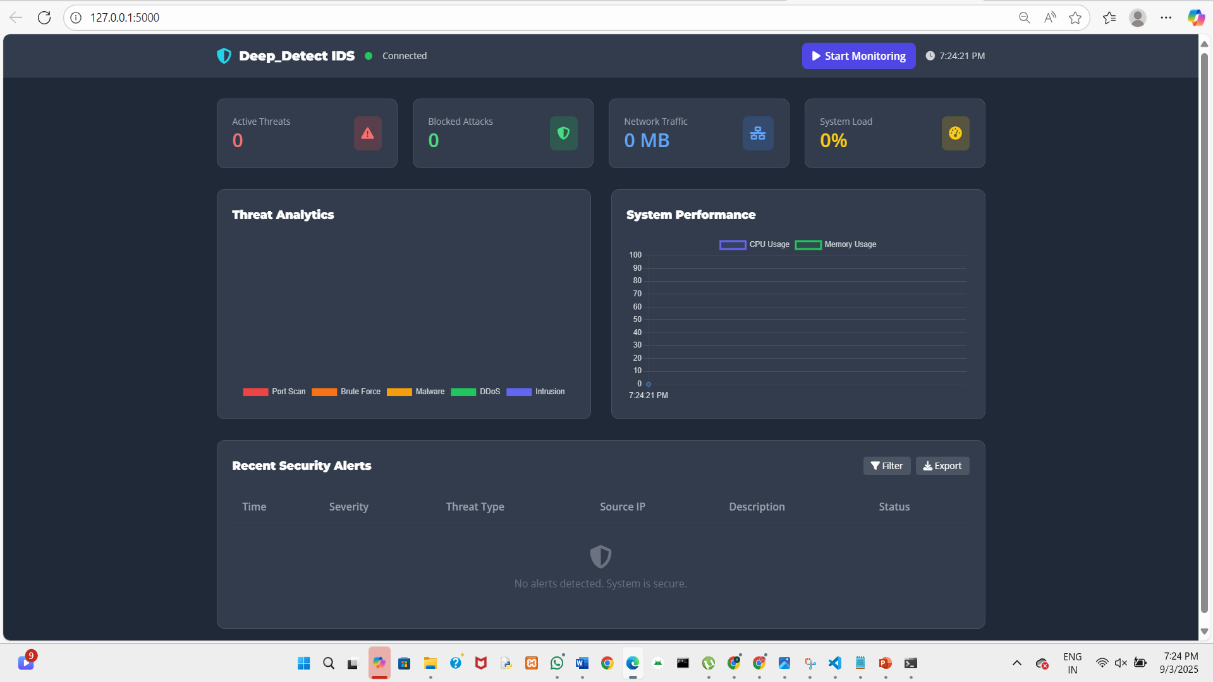
****

Fig:4.2.3 Web dashboard

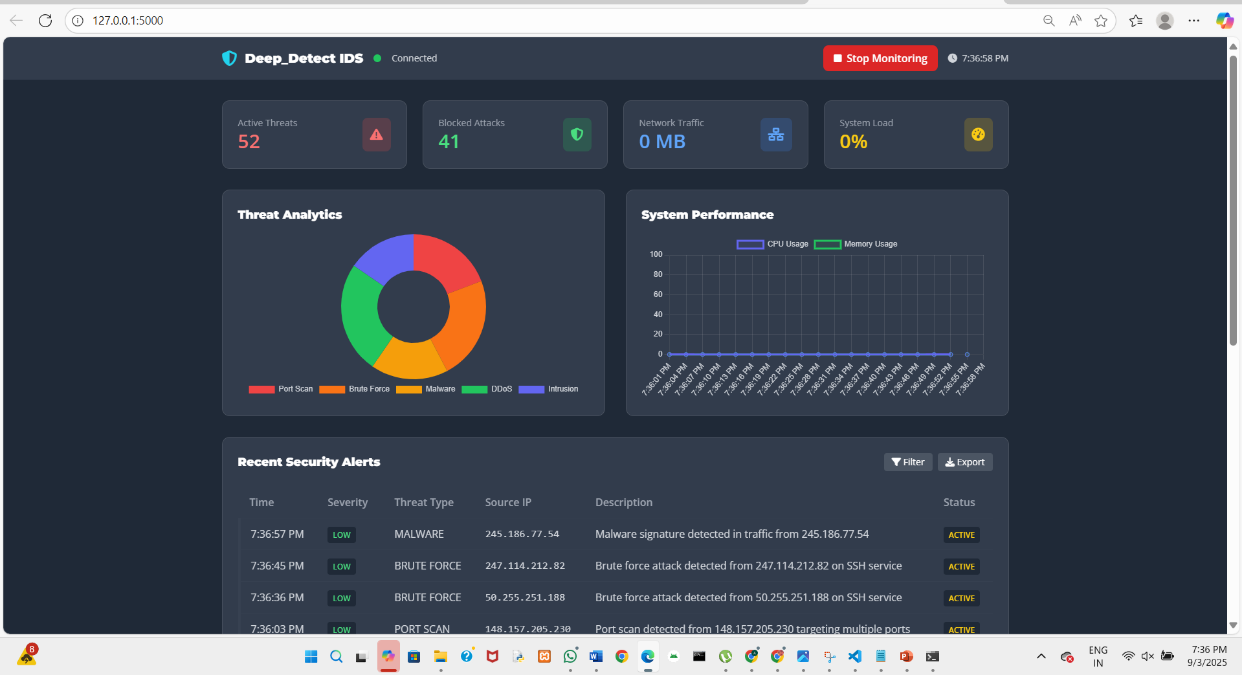
****

Fig:4.2.4 Web dashboard report

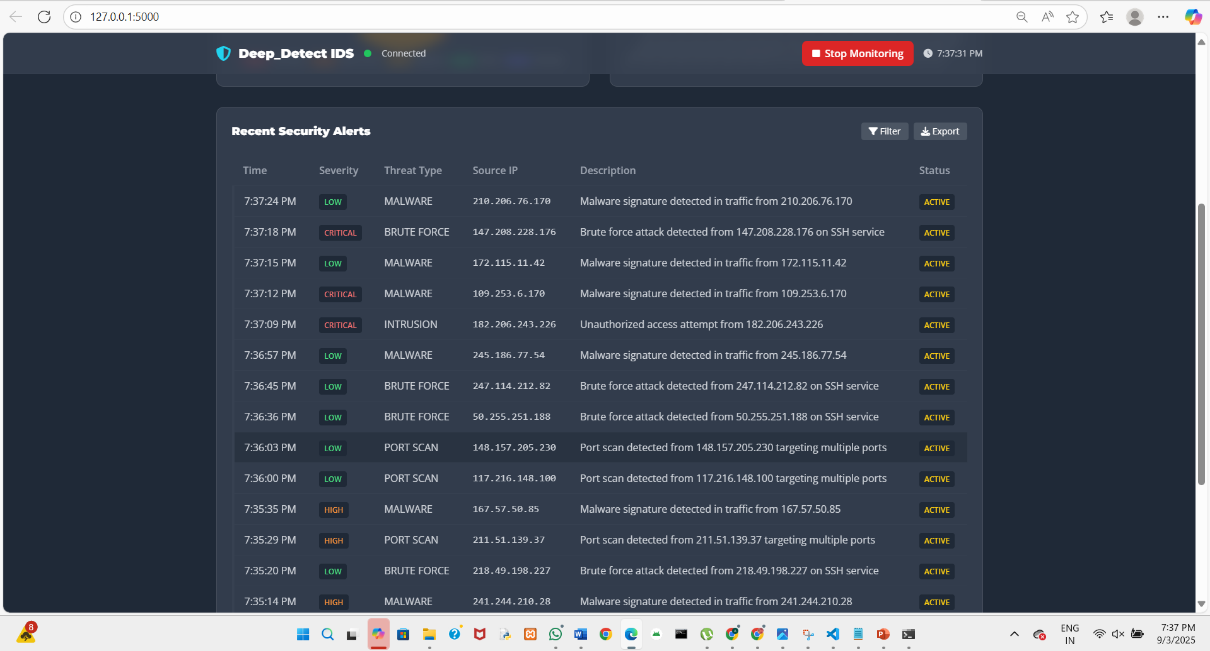
****

Fig:4.2.5 alerts

|  |  |  |  |
| --- | --- | --- | --- |
| Test Parameter | Description | Result (Sample) | Remarks |
| Detection Rate | Ability to correctly identify intrusions | 92% | Good, but can be improved further |
| False Positives | Normal traffic incorrectly flagged as attack | 5% | Needs optimization to reduce unnecessary alerts |
| False Negatives | Malicious traffic not detected | 3% | Acceptable, but dangerous in critical systems |
| System Performance | CPU, memory, and network load impact | +12% CPU under heavy load | Within limits, but requires monitoring |
| Response Time | Time taken to detect and generate alert | 1.2 seconds | Fast detection suitable for real-time alerts |
| Scalability | Handling of high-volume network traffic | Stable up to 10 Gbps | Effective for enterprise-scale networks |
| Evasion Resistance | Ability to detect encrypted/fragmented/stealth traffic | Partially effective | Needs improvement against advanced evasion |
| Usability | Ease of configuration, reporting, and monitoring | Moderate | Requires skilled administrator |

**4.3. Test reports**

Table:4.3.1

**Chapter: 5**

**Proposed Enhancements**

* 1. **Integration of Artificial Intelligence (AI) and Machine Learning (ML)**

**Current Limitation:** Traditional IDS generate many false positives/negatives and struggle with new attack patterns.

**Benefits:** AI/ML improves detection accuracy, adapts to evolving threats, and helps identify zero-day attacks.

**Challenges:** Requires large datasets, high computational resources, and faces explainability issues.

**Proposed Enhancement:** Incorporate AI/ML-driven models that continuously learn from network traffic and adapt in real time.

**5.2 Hybrid Detection Approach (Signature + Anomaly-Based IDS)**

**Current Limitation:** Signature-based IDS misses unknown attacks, while anomaly-based IDS creates too many false alarms.

**Benefits:** A hybrid approach balances detection accuracy and reduces false positives.

**Challenges**: More complex to design, requires tuning, and may increase processing overhead.

**Proposed Enhancement:** Combine signature-based methods for known threats with anomaly-based detection for unknown ones.

**5.3 Real-Time Threat Intelligence Integration**

**Current Limitation:** Traditional IDS rely on static signatures, making them outdated against emerging threats.

**Benefits:** Provides timely updates on malware, ransomware, and global attack

campaigns.

**Challenges:** Requires trusted threat intelligence sources and continuous updates to avoid overload.

**Proposed Enhancement:** Integrate IDS with live global threat intelligence feeds for proactive detection.

**5.4 Cloud-Based and Distributed IDS**

**Current Limitation:** On-premise IDS cannot efficiently monitor cloud or IoT environments.

**Benefits:** Scalable, centralized monitoring for hybrid and distributed networks.

**Challenges:** Data privacy issues, latency concerns, and difficulty in integrating with legacy systems.

**Proposed Enhancement:** Deploy cloud-based IDS with distributed sensors to monitor large-scale and IoT networks.

**5.5 Automated Incident Response**

**Current Limitation:** IDS mostly provide alerts, requiring manual intervention, which delays response.

**Benefits:** Faster response to threats, reduced workload for analysts, and minimized damage.

**Challenges:** Risk of false triggers leading to unnecessary disruptions.

**Proposed Enhancement:** Integrate IDS with SOAR tools to enable automated blocking and isolation of threats.

**5.6 Encrypted Traffic Analysis**

**Current Limitation:** Attackers hide malicious traffic within SSL/TLS encryption, making detection difficult.

**Benefits:** Provides visibility into encrypted traffic without losing security.

**Challenges:** High computational cost, potential legal/privacy issues, and risk of slowing networks.

**Proposed Enhancement:** Develop advanced techniques to securely inspect encrypted traffic.

**Chapter 6**

**Conclusion**

* IDS are essential for monitoring networks and systems to identify malicious or policy-violating activity and alert responders, often integrating with SIEM for broader visibility
* Modern IDS combine signature-based and anomaly-based techniques, increasingly enhanced with machine learning, to improve detection across on-prem, cloud, and hybrid environments
* While IDS improves detection and compliance posture, it does not inherently block threats; pairing with IPS or broader controls is common to enable prevention and automated response
* Key challenges remain: tuning to reduce false positives and negatives, scaling to high-throughput networks, and adapting to encrypted traffic and evolving attacker tactics
* Effective IDS programs rely on good data sources, thoughtful deployment points, continuous rule and model tuning, and integration with incident response workflows to translate alerts into action

**Chapter 7**

**Bibliography**

**Offline References**

[1] S. Northcutt and J. Novak, *Network Intrusion Detection*, 3rd ed. Indianapolis, IN, USA: Sams Publishing, 2002.

[2] R. Bace, *Intrusion Detection*. Indianapolis, IN, USA: Macmillan Technical Publishing, 2000.

[3] E. Amoroso, *Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Traps, Trace Back, and Response*. Hoboken, NJ, USA: Intrusion.Net Books, 1999.

[4] B. Mukherjee, L. T. Heberlein, and K. N. Levitt, “Network intrusion detection,” *IEEE Network*, vol. 8, no. 3, pp. 26–41, May/Jun. 1994.

[5] C. Kruegel, F. Valeur, and G. Vigna, *Intrusion Detection and Correlation: Challenges and Solutions*. New York, NY, USA: Springer Science & Business Media, 2005.

**Online References**

[6] K. Scarfone and P. Mell, *Guide to Intrusion Detection and Prevention Systems (IDPS)*, NIST Special Publication 800-94, 2007. [Online]. Available: [https://nvlpubs.nist.gov](https://nvlpubs.nist.gov?utm_source=chatgpt.com)

[7] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung, “Intrusion detection system: A comprehensive review,” *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 16–24, Jan. 2013, doi: 10.1016/j.jnca.2012.09.004.

[8] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, “Anomaly-based network intrusion detection: Techniques, systems and challenges,” *Comput. Secur.*, vol. 28, no. 1–2, pp. 18–28, Feb.–Mar. 2009, doi: 10.1016/j.cose.2008.08.003.

[9] OWASP Foundation, “Intrusion Detection Systems (IDS),” OWASP, 2021. [Online]. Available: [https://owasp.org](https://owasp.org?utm_source=chatgpt.com)