**Interim Progress Report (IPR)**

**A Signature-based approach towards Intrusion Detection in Encrypted Network Traffic with Zero Trust and Blockchain Integration**

|  |  |
| --- | --- |
| **Name:** | **Farhan Tauqeer** |
| **Student ID:** | **23034210** |
| **Course:** | **7COM1070 Cyber Security Masters Project** |
| **Supervisor:** | **Dr. Deepthi Ratnayake** |
| **Date:** | **3rd March, 2025** |

**Contents**

[1 Introduction and Overview 4](#_Toc191856104)

[1.1 Project Background 4](#_Toc191856105)

[1.2 Research Question 4](#_Toc191856106)

[1.3 Aim 4](#_Toc191856107)

[1.4 Research Objectives 5](#_Toc191856108)

[1.4.1 Investigate the Challenges in Intrusion Detection for Encrypted Traffic 5](#_Toc191856109)

[1.4.2 Analyze the Role of Zero Trust in Enhancing NIDS Effectiveness 5](#_Toc191856110)

[1.4.3 Evaluate the Impact of Blockchain for Secure IDS Logging 5](#_Toc191856111)

[1.5 Developmental Objectives 5](#_Toc191856112)

[1.5.1 Setup of Controlled Lab Environment 5](#_Toc191856113)

[1.5.2 Implement a Dynamic Signature-Based Detection Mechanism 6](#_Toc191856114)

[1.5.3 Integrate Zero Trust Security into NIDS 6](#_Toc191856115)

[1.5.4 Develop a Blockchain-Secured Logging System for IDS Events 6](#_Toc191856116)

[1.6 Evaluation Objectives 6](#_Toc191856117)

[1.6.1 Validate Detection Accuracy of the Proposed IDS Model 6](#_Toc191856118)

[1.6.2 Assess the Performance of Zero Trust Security Policies 6](#_Toc191856119)

[1.6.3 Measure the Efficiency and Scalability of Blockchain-Based Logging 6](#_Toc191856120)

[1.7 Ethical Considerations 6](#_Toc191856121)

[1.7.1 Minimization of Data Collection 7](#_Toc191856122)

[1.7.2 Exclusion of Personally Identifiable Information (PII) 7](#_Toc191856123)

[1.7.3 Compliance with Research Ethics Standards 7](#_Toc191856124)

[1.8 Legal Considerations 7](#_Toc191856125)

[1.8.1 Data Protection and Privacy Regulations 7](#_Toc191856126)

[1.8.2 Cyber Security and Compliance Standards 7](#_Toc191856127)

[1.8.3 Legal Risk Mitigation Strategies 7](#_Toc191856128)

[1.9 Professional Considerations 8](#_Toc191856129)

[1.9.1 Adherence to Industry Standards 8](#_Toc191856130)

[1.9.2 Secure Development Practices 8](#_Toc191856131)

[1.10 Social Considerations 8](#_Toc191856132)

[2 Progress To Date 8](#_Toc191856133)

[2.1 Literature Review 8](#_Toc191856134)

[2.2 System Design and Architecture 11](#_Toc191856135)

[2.3 Network Setup 12](#_Toc191856136)

[2.4 Components of ZT-NIDS 12](#_Toc191856137)

[2.4.1 Traffic Collection 12](#_Toc191856138)

[2.4.2 Intrusion Detection 12](#_Toc191856139)

[2.4.3 Zero Trust Controls 13](#_Toc191856140)

[2.5 Implementation Progress 13](#_Toc191856141)

[2.5.1 Completed Milestones 13](#_Toc191856142)

[2.5.2 In Progress 13](#_Toc191856143)

[3 Planned Work 13](#_Toc191856144)

[3.1 Remaining Tasks 13](#_Toc191856145)

[3.2 Project Evaluation 14](#_Toc191856146)

[3.2.1 Metrics for Success 14](#_Toc191856147)

[3.2.2 Comparison 14](#_Toc191856148)

[4 Conclusion 14](#_Toc191856149)

[5 References (generated using Mendeley) 15](#_Toc191856150)

[6 Appendices (Supporting Evidence) 16](#_Toc191856151)

[6.1 Appendix A: SNORT Validation 16](#_Toc191856152)

[6.2 Appendix B: Feature Extraction 18](#_Toc191856153)

[6.3 Appendix C: Gantt Chart 22](#_Toc191856154)

# Introduction and Overview

## Project Background

With the rapid expansion of digital communication and cloud-based infrastructures, ensuring network security has become increasingly challenging. The widespread adoption of encryption protocols, such as Transport Layer Security (TLS) 1.3, has significantly enhanced privacy and data security by preventing unauthorized interception of sensitive communications. However, this increased encryption poses a significant challenge for Intrusion Detection Systems (IDS), as deep packet inspection (DPI)-based techniques are no longer effective in analyzing encrypted network traffic (Panchal, Snehkunj, and Panchal, 2024). Attackers are exploiting encryption to evade detection, embedding malicious payloads within TLS-encrypted sessions, rendering traditional signature-based IDS ineffective (Papadogiannaki, Tsirantonakis, and Ioannidis, 2022). This necessitates the development of alternative detection mechanisms that do not require decryption of traffic but can still identify threats hidden within encrypted communication channels.

To address the challenges associated with traditional perimeter-based security models, Zero Trust Architecture (ZTA) has emerged as a modern cybersecurity paradigm. Zero Trust follows the fundamental principle of "Never Trust, Always Verify", if both external and internal entities pose security risks and should undergo continuous authentication, monitoring, and least-privilege access enforcement (Syed, 2022). Unlike conventional network security models, which implicitly trust devices and users within a secured perimeter, Zero Trust enforces granular access controls and verifies all entities attempting to interact with network resources (Gautam et al., 2024).

In the context of intrusion detection system, Zero trust principles enables:

* Real time verification of users, devices and applications.
* Least-privilege access.
* Continuous monitoring and threat detection.

Another major challenge in Intrusion Detection Systems is ensuring the integrity and security of incident logs. In traditional IDS architectures, attackers who gain unauthorized access to network infrastructure often attempt to tamper with security logs, effectively erasing forensic evidence of their activity (Meng et al., 2018). This undermines the ability of security teams to conduct post-incident analysis and respond effectively to breaches.

To address this, blockchain technology is being incorporated into the ZT-NIDS framework as a tamper-proof logging mechanism. Blockchain is a distributed ledger technology that ensures immutable and transparent record-keeping, making it ideal for securing intrusion detection logs (Alevizos et al., 2022). Data Integrity and Immutability, Decentralized Storage and Auditability are some of the key advantages of blockchain backed Intrusion Detection System (Meng et al., 2018).

One of the most effective ways to detect cyber threats is through signature-based intrusion detection, where known attack patterns are matched against real-time network traffic. However, the challenge with encrypted traffic is that conventional deep packet inspection (DPI) approaches are ineffective since they rely on payload visibility (Papadogiannaki and Ioannidis, 2021). To overcome this limitation, this project utilizes a metadata-driven approach where signatures are generated based on encrypted traffic features rather than plaintext content.

To create an effective intrusion detection model, a Signature Language Generator will be designed using the VMSP (Vertical Mining of Sequential Patterns) algorithm. This method identifies common patterns and recurring attributes within encrypted malicious network traffic, allowing the IDS to generate attack signatures dynamically (Panchal et al., 2024). The generated signatures will be stored in a structured signature database to facilitate efficient real-time detection.This signature-based approach, when combined with Zero Trust enforcement and blockchain-backed logging, enhances the ability of ZT-NIDS to detect and mitigate cyber threats within encrypted communication channels without compromising user privacy.

## Research Question

How can a signature-based Network Intrusion Detection System (NIDS) effectively detect threats in encrypted network traffic?

## Aim

This project aims to develop a Network Intrusion Detection System (NIDS) that enhances threat detection in encrypted network traffic by integrating dynamic signature-based detection, Zero Trust security enforcement, and blockchain-secured logging to improve detection accuracy, prevent lateral movement, and ensure forensic integrity.

This project **improves** upon the research presented in **"Network Intrusion Detection in Encrypted Traffic"** by(Papadogiannaki, Tsirantonakis and Ioannidis, 2022)By enhancing metadata-based intrusion detection by incorporating more packet features to be able to detect different malware families with dynamic signature updates, Zero Trust security enforcement, and blockchain-secured logging, thereby addressing key limitations such as limited threat detection scope, static signatures, high false positives, and lack of automated threat mitigation.

The project offers the following contributions in Cyber Security Research:

1. A working prototype of NIDS that detects intrusions in real-time encrypted traffic and dynamically updates the attack signatures.
2. Integration of Zero Trust and blockchain-secured logging in an IDS environment.
3. Scientific comparison of generated signatures against traditional IDS in encrypted network traffic.

## Research Objectives

### Investigate the Challenges in Intrusion Detection for Encrypted Traffic

* Conduct a systematic literature review on existing traditional signature-based IDS models and their effectiveness in encrypted environments.
* Identify key limitations of traditional NIDS, including false positives, static signatures, and limited adaptability to encrypted threats.
* Compare metadata-based detection techniques, such as packet size sequences, TLS fingerprinting, and inter-packet timing analysis.

### Analyze the Role of Zero Trust in Enhancing NIDS Effectiveness

* Examine how Zero Trust security policies can limit lateral movement.
* Study the feasibility of dynamic access control based on real-time threat detection.
* Compare Zero Trust-based IDS models against traditional perimeter-based security solutions.

### Evaluate the Impact of Blockchain for Secure IDS Logging

* Assess existing blockchain-based security logging frameworks for their effectiveness in ensuring tamper-proof forensic records.
* Benchmark blockchain-based logging performance against traditional logging mechanisms.

## Developmental Objectives

### Setup of Controlled Lab Environment

* Configure a virtualised test environment using multiple Linux-based virtual machines (VMs) to simulate real-world network traffic.
* Deploy essential open-source security tools such as scapy, Tshark, and Cisco Joy for traffic analysis and feature extraction.
* Simulate both legitimate and malicious encrypted traffic scenarios to validate IDS performance.

### Implement a Dynamic Signature-Based Detection Mechanism

* Develop a signature extraction and update model to automatically generate signatures based on packet metadata using basic pattern mining algorithms.
* Design a mechanism for continuous signature updates from live traffic analysis.
* Validate the efficiency of the system in detecting encrypted attack patterns.

### Integrate Zero Trust Security into NIDS

* Develop role-based access control (RBAC) mechanisms to prevent unauthorised lateral movement.
* Automate access restriction rules, based on threat detection events.

### Develop a Blockchain-Secured Logging System for IDS Events

* Design a Hyperledger-based logging system to ensure tamper-proof security event storage.

## Evaluation Objectives

### Validate Detection Accuracy of the Proposed IDS Model

* Test the developed IDS against real-world dataset CIC-IDS2017.
* Measure detection accuracy, false positives, and false negatives for encrypted and non-encrypted traffic.
* Compare results against traditional Intrusion Detection Systems like SNORT or Suricata.

### Assess the Performance of Zero Trust Security Policies

* Analyze how Zero Trust mechanisms restrict lateral movement in simulated attack scenarios.
* Compare response times between traditional NIDS vs. Zero Trust-enhanced NIDS.

### Measure the Efficiency and Scalability of Blockchain-Based Logging

* Compare blockchain-based logs with traditional SQL and NoSQL security logging solutions.

## Ethical Considerations

The project involves analysing **network metadata** for threat detection, raising ethical concerns regarding **user privacy, consent, and data protection**. Since **Transport Layer Security (TLS) 1.3 and other encryption standards** are designed to preserve confidentiality (Panchal, Snehkunj and Panchal, 2024). Intrusion detection must be implemented in a way that **does not compromise legitimate user privacy.**

To address ethical concerns, the project adheres to the following principles:

### Minimization of Data Collection

The system will focus exclusively on metadata-based analysis (e.g., packet size, timing patterns, and TLS fingerprints) rather than decrypting or inspecting packet payloads. This approach aligns with privacy-by-design principles outlined in ISO/IEC 29100.

### Exclusion of Personally Identifiable Information (PII)

No PII or user-specific content will be processed, ensuring compliance with privacy standards such as ISO/IEC 27701:2019, which extends ISO/IEC 27001 for privacy information management.

### Compliance with Research Ethics Standards

The research methodology follows ethical cybersecurity research frameworks, such as the Menlo Report (of Homeland Security, Directorate and Security Division, 2012), ensuring responsible and lawful handling of network security data.

## Legal Considerations

The development and testing of an intrusion detection system must comply with international, regional, and national data protection laws, including:

### Data Protection and Privacy Regulations

* **General Data Protection Regulation (GDPR, EU 2016/679):** The system ensures lawful data processing without user profiling or behavioral tracking, aligning with Article 25 (Data Protection by Design and by Default) and Article 32 (Security of Processing).
* **UK Data Protection Act (2018):** The project adheres to UK-specific data security and privacy mandates, particularly those concerning automated decision-making and cybersecurity risk assessments.

### Cyber Security and Compliance Standards

* **ISO/IEC 27001:2022 (Information Security Management):** Ensures the system adheres to internationally recognised security controls and risk management practices.
* **NIST Special Publication 800-207 (Zero Trust Architecture)**: The project aligns with Zero Trust principles, enforcing strict access control and continuous monitoring of security threats.
* **Computer Misuse Act (UK, 1990)**: The system is designed to detect threats without engaging in unauthorised network surveillance or hacking activities.

### Legal Risk Mitigation Strategies

* **Explicit Authorization:** Testing will be conducted only on authorised networks, ensuring that the system does not violate legal restrictions on traffic interception.
* **Security and Transparency:** All findings will be reported by responsible disclosure policies, ensuring that vulnerabilities are not exploited for malicious purposes.

## Professional Considerations

The development of the NIDS system aligns with recognised professional and ethical standards.

### Adherence to Industry Standards

* **ACM Code of Ethics (Association for Computing Machinery):** The system adheres to Principle 1.2 (Avoid Harm), Principle 1.6 (Respect Privacy), and Principle 2.9 (Design and Implement Secure Systems).
* **BCS Code of Conduct (British Computer Society):** The project aligns with public interest, professional competence, and integrity in security research.

### Secure Development Practices

Security testing will be conducted in controlled environments to ensure system effectiveness without introducing new vulnerabilities.

## Social Considerations

* The system will incorporate adaptive tuning mechanisms to minimise unnecessary alerts and ensure high detection accuracy.
* Ensures tamper-proof forensic audit trails, aligned with ISO/IEC 2022 for digital evidence management**.**
* The project contributes to public knowledge on encrypted traffic security, supporting initiatives such as NIST’s Cybersecurity Framework (CSF) and ENISA's Threat Landscape Reports.
* The system will respect fundamental privacy rights, ensuring that security measures do not infringe upon user freedoms (UN Declaration of Human Rights, Article 12).

# Progress To Date

## Literature Review

The rapid evolution of cyber threats, particularly in encrypted network environments, has necessitated significant advancements in **Network Intrusion Detection Systems (NIDS).** Traditional intrusion detection methods have predominantly relied on **signature-based detection**, leveraging tools such as Snort and Suricata, which employ **deep packet inspection (DPI)** to identify malicious activity. (EFE and ABACI, 2022). However, with the widespread adoption of **TLS 1.3, QUIC, and other encryption protocols**, these methods are becoming increasingly ineffective as they cannot inspect packet contents. Consequently, **meta-data-based detection** approaches, which analyse **packet size sequences, inter-packet timing, and TLS fingerprinting**, have emerged as viable alternatives. (Papadogiannaki, Tsirantonakis and Ioannidis, 2022). This review critically examines the evolution of NIDS models, the effectiveness of Zero Trust security frameworks, and the potential of **AI-driven** and **blockchain-enabled security enhancements**, laying the foundation for the research direction of this project.

Early research on **NIDS architectures** introduced the **NetSTAT model**, which employed a **state-transition analysis** approach to detect network intrusions based on predefined attack sequences. (Vigna and Kemmerer, 1998). While effective in recognising known attack patterns, this model exhibited **limited adaptability to zero-day threats** due to its reliance on static rule sets. Subsequent advancements led to the development of **hybrid intrusion detection systems**, which combined **signature-based and anomaly-based detection** techniques to improve detection rates. (Kumar, Gupta and Arora, 2021). These models demonstrated enhanced accuracy but struggled with **false positives**, particularly when analysing **encrypted network traffic** where **payload visibility is restricted**.

To overcome these limitations, researchers have explored machine learning (ML) and deep learning (DL) models for automated anomaly detection. (Bivens et al., 2002) Pioneered the application of **neural networks** to detect **denial-of-service (DoS), distributed denial-of-service (DDoS), and port scanning attacks**, demonstrating that **self-organising maps (SOMs) and multi-layer perceptron (MLP) models** could effectively classify network anomalies. However, **high computational overhead and lack of interpretability** posed challenges for real-time deployment in high-speed networks. More recent studies by (Panchal, Snehkunj and Panchal, 2024) Have shown that **convolutional neural networks (CNNs), bi-directional extended short-term memory networks (BiLSTMs), and autoencoders** significantly enhance NIDS performance by **dynamically adapting to evolving attack vectors**. Despite these improvements, **false favourable rates remain high**, leading to operational inefficiencies and security fatigue among network defenders. (Oliveira *et al.*, 2021).

The growing **adoption of Zero Trust Security Models (ZTA)** has introduced a **fundamental shift** in network defense strategies. Traditional perimeter-based security models, which assume that **internal network traffic is inherently trusted**, leave organisations vulnerable to **insider threats and lateral movement attacks**. (Alalmaie, 2023)Propose integrating **Zero Trust principles** into NIDS, implementing **continuous authentication, micro-segmentation, and automated threat response mechanisms** to enhance security posture. Zero Trust architecture enables dynamic access control by leveraging real-time security telemetry, **preventing attackers from escalating privileges or moving laterally within compromised networks**.

Another significant advancement in **NIDS security frameworks** is integrating **blockchain technology for secure logging and forensics**. Conventional logging mechanisms are prone to **tampering**, allowing adversaries to modify or delete records to **erase traces of their activities**. (Alevizos *et al.*, 2022) Introduced a **blockchain-enabled intrusion detection system** utilising **Hyperledger Fabric** to **maintain immutable security event logs**. By embedding **smart contracts**, blockchain-enhanced NIDS ensures **forensic integrity, regulatory compliance, and automated policy enforcement**. This approach mitigates the risks associated with **log manipulation and unauthorised data alterations**, providing a **transparent and verifiable intrusion record**.

Despite these advancements, several **critical gaps remain in the field of NIDS for encrypted traffic**. Many existing models rely on **static signature databases**, making them ineffective against **zero-day attacks and sophisticated evasion techniques.** (Papadogiannaki and Ioannidis, 2021). This research addresses these challenges by integrating **dynamic signature updates**, ensuring the system can **continuously learn and adapt to new threats**. Additionally, most **metadata-based encrypted traffic detection techniques** focus primarily on **packet payload size sequences**, limiting their effectiveness in **distinguishing between benign and malicious encrypted traffic**. By incorporating **TLS handshake analysis and inter-flow timing patterns**, this project aims to **improve detection accuracy while maintaining privacy-preserving security measures**.

A significant limitation of traditional IDS implementations is their **passive nature**, where they focus primarily on detection without **real-time mitigation capabilities**. (Alalmaie, Nanda and He, 2023) By **integrating Zero Trust principles**, an IDS can actively **block malicious traffic upon detection**, reducing **attacker dwell time and limiting potential damage**. Furthermore, **blockchain-secured logging** ensures **compliance with industry regulations and forensic integrity**, a critical aspect for organisations handling **sensitive and classified information.** (Alevizos *et al.*, 2022).

The increasing reliance on **cloud computing and IoT-driven infrastructures** introduces additional challenges for **NIDS scalability**. The **high volume of network traffic and resource constraints in IoT environments** necessitate the development of **lightweight, efficient intrusion detection models** capable of operating in **real-time with minimal computational overhead.** (Madiba and Velempini, 2023). This research aims to **optimise signature-matching algorithms and enhance metadata-based detection techniques**, ensuring **NIDS solutions remain scalable and effective** in high-speed, resource-constrained environments.

In conclusion, the **literature highlights a clear evolution in NIDS methodologies**, transitioning from **static rule-based detection to AI-driven, Zero Trust-secured, and blockchain-enhanced security frameworks**. However, **significant challenges remain**, particularly in **encrypted traffic analysis, adaptive threat response, and scalability**. This project builds upon existing research by developing a **comprehensive NIDS framework** that integrates **dynamic signature updates, Zero Trust enforcement, and blockchain-secured logging**. By addressing these gaps, this research contributes to creating **a next-generation intrusion detection system** that is **adaptive, resilient, and effective against modern cyber threats**, particularly in **encrypted network environments where traditional IDS models fail**.

Table Summary of Literature Review

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Research Area** | **Key Findings** | **Limitations** | **Addressed in this Research** | **Citations** |
| Signature-Based IDS | Effective for known attack patterns (e.g., Snort, Suricata) | Limited against encrypted traffic and zero-day threats | Meta-data-based detection for encrypted traffic | (EFE and ABACI, 2022) |
| Anomaly-Based IDS | ML/DL models (CNNs, BiLSTM) enhance adaptability | High false positive, heavy | Lightweight Pattern Mining Algorithms for detection | (Panchal, Snehkunj and Panchal, 2024) |
| Zero Trust Security | Prevents lateral movement and unauthorised access | Requires continuous authentication and segmentation | Implements Zero Trust access control for dynamic threat mitigation | (Alalmaie, 2023) |
| Blockchain-Enhanced IDS | Ensures tamper-proof forensic logs | Computational overhead in real-time applications | Hyperledger-based logging for forensic integrity | (Alevizos *et al.*, 2022) |
| Encrypted Traffic Analysis | Metadata-based IDS (packet size, inter-packet timing) improves detection | Limited detection scope | Enhances encrypted traffic detection with the addition of more features | (Papadogiannaki, Tsirantonakis and Ioannidis, 2022) |
| Scalability in Cloud & IoT | High-speed networks require lightweight IDS solutions | Resource constraints limit effectiveness | Optimised signature-matching for efficient performance | (Madiba and Velempini, 2023) |

## System Design and Architecture

The Network Intrusion Detection System (NIDS) is developed within a Linux-based virtualised environment using open-source security tools and custom intrusion detection algorithms. The architecture incorporates signature-based detection, flow-based anomaly detection, and zero-trust security principles to enhance network security in encrypted environments.

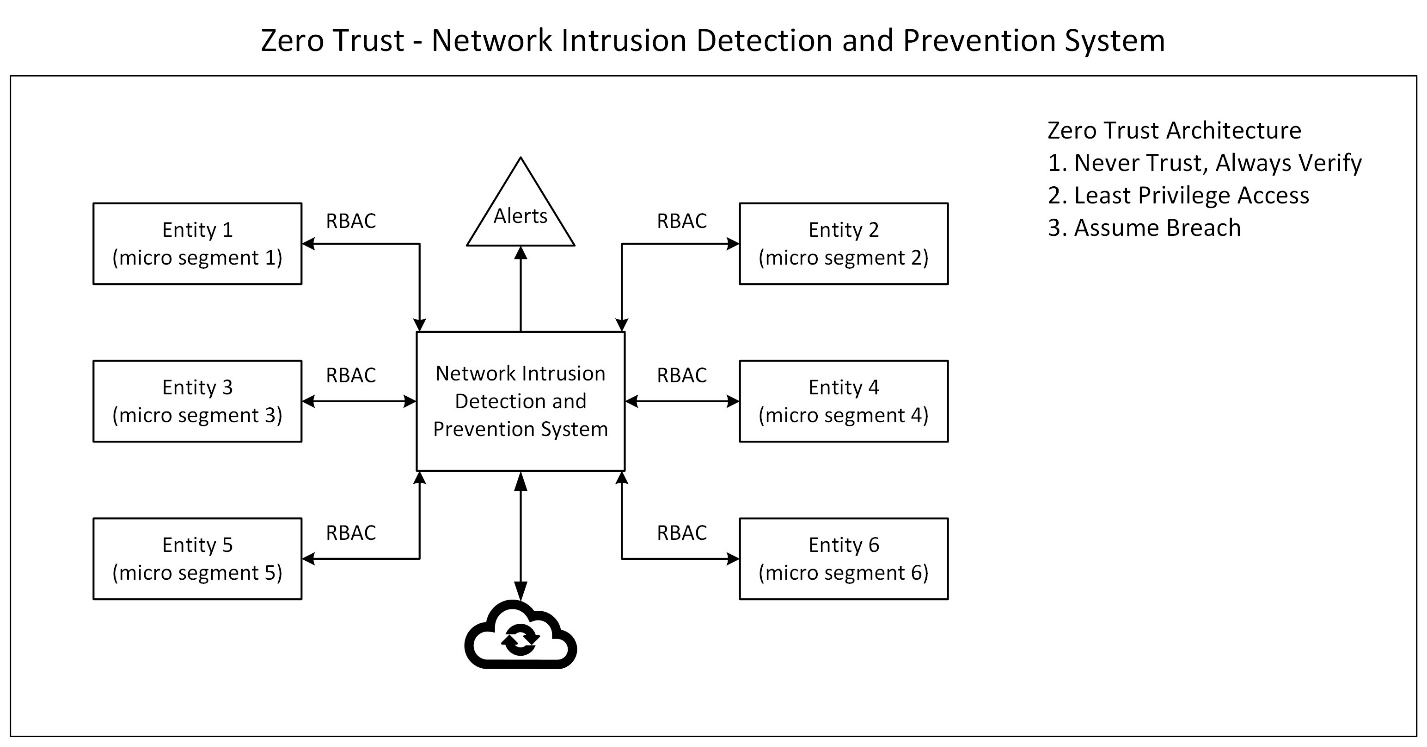


Figure Overview of Zero Trust Intrusion Detection and Prevention System

## Network Setup

The virtualised testbed consists of multiple Ubuntu-based Virtual Machines (VMs) simulating a real-world network environment. The infrastructure includes:

* **Attack Simulation Machine:** Generate network traffic, including encrypted malicious activity.
* **ZT-NIDS Server:** Hosts Signature-based Detection Mechanisms.
* **Normal Ubuntu User:** Mimics regular user activity.

## Components of ZT-NIDS

The system consists of multiple functional components designed to detect and mitigate cyber threats in encrypted traffic environments.

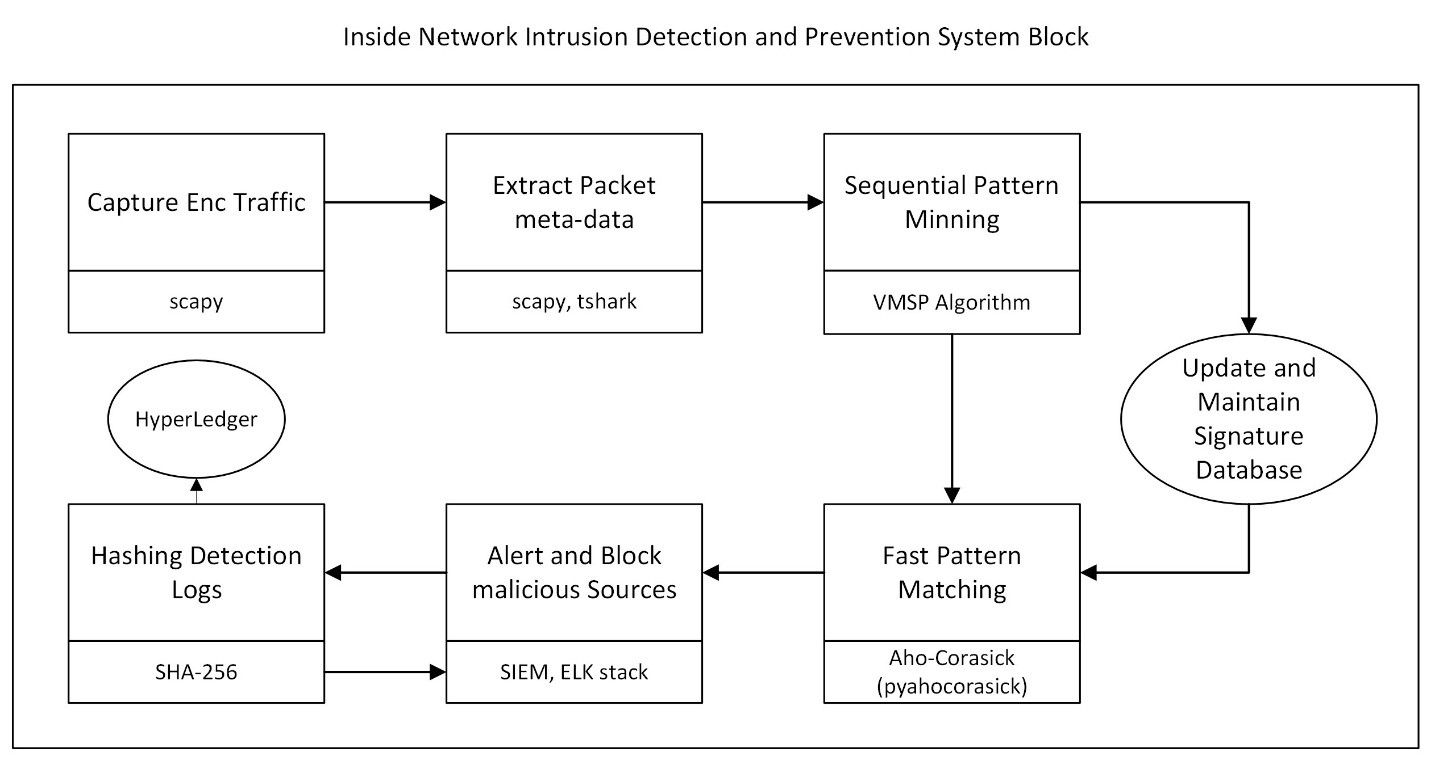


Figure Inside NIDS Block

### Traffic Collection

* **Custom Packet Sniffer (Python - Scapy):** Captures raw network packets for further analysis.
* **Tshark (Wireshark CLI):** Performs deep network traffic analysis, extracting packet-level metadata for anomaly detection and forensic investigations.

### Intrusion Detection

* **Feature Extraction:** Utilizing **Cisco Joy** to extract useful feature**s** from PCAP files captured via Tshark.
* **Signature Generation:** Leveraging **pattern mining algorithms** (non-AI) to identify patterns in packet metadata. Dynamically building a signature knowledge base based on extracted features from Cisco Joy**.**
* **Pattern Matching:** Implementing the **Aho-Corasick algorithm** for efficient real-time pattern matching. Detecting matches between live network traffic and generated signatures for rapid threat identification.
* **Alerts and Visualization:** Integrating ELK Stack (Elasticsearch, Logstash, Kibana) for storing, indexing, and visualising detection logs.
* **Hashing and Decentralization:** Applying **SHA-256 hashing** on detection logs to ensure data integrity. Uploading hashed logs onto a **secure Hyperledger blockchain** to maintain tamper-proof records and comply with confidentiality and integrity standards.

### Zero Trust Controls

* **Role-Based Access Control (RBAC) via FreeIPA:** Implements strict authentication and authorisation policies.
* **Multi-Factor Authentication (MFA):** Ensures identity verification for critical network access.
* **Dynamic Firewall Rule Enforcement (iptables):** Dynamically blocks suspicious connections based on real-time threat intelligence.

## Implementation Progress

### Completed Milestones

* Literature Review
* Virtualized Test Environment
* Traffic Capture and Analysis
* Installation and configuration of SNORT for research and comparison

### In Progress

* Dynamic Signature Generation
* Pattern Matching
* Alerts and Logging
* Hashing

# Planned Work

## Remaining Tasks

Table Remaining Tasks with completion target

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Target Completion** |
| Dynamic Signature Generation | Selection and implementation of the Pattern mining algorithm | Week 8 |
| Pattern Matching | Implementation of the Aho-Corasick Algorithm for fast string matching | Week 9 |
| Alerts and Logging | Secure Logging and alert integrity | Week 10 |
| Expand Attack Simulation | Include encrypted malware traffic | Week 11 |
| Performance Testing and Benchmarking | Evaluate efficiency and detection rates | Week 12 |
| Prepare Final Report | Documentation and submission | Week 13-14 |

## Project Evaluation

### Metrics for Success

* **True Positive Rate (TPR) / Detection Rate:** Percentage of actual attacks correctly identified.
* **False Positive Rate (FPR):** Percentage of regular traffic incorrectly flagged as an attack.
* **False Negative Rate (FNR):** Percentage of undetected attacks.
* **Precision & Recall:** Helps determine the system’s ability to detect attacks while minimising false positives correctly.

### Comparison

Comparison of metrics mentioned above with the popular NIDS (SNORT, Suricata) and the NIDS introduced by (Papadogiannaki, Tsirantonakis and Ioannidis, 2022).

# Conclusion

This research investigates the feasibility of a signature-based Network Intrusion Detection System (NIDS) for encrypted network traffic by integrating Zero Trust security principles and blockchain-secured logging mechanisms. Traditional intrusion detection techniques, primarily reliant on deep packet inspection, have proven inadequate due to the widespread adoption of encryption protocols such as TLS 1.3. To address these challenges, this project introduces a Zero Trust-enabled NIDS that leverages metadata-based anomaly detection, dynamic signature updates, and secure forensic logging through blockchain technology.

The initial phases of this research have provided valuable insights into the limitations of conventional IDS models, particularly in detecting encrypted threats without decrypting network traffic. The literature review has established a foundation for designing a more effective, scalable, and privacy-preserving intrusion detection approach. A controlled lab environment has been configured, where network traffic is analysed, signature-based detection mechanisms are implemented, and blockchain-secured logging is integrated for tamper-proof auditing.

Progress thus far includes system architecture design, virtualised environment setup, initial traffic collection testing, and Zero Trust access control policy implementation. The remaining work focuses on refining dynamic signature generation, optimising anomaly detection algorithms, and evaluating the system’s effectiveness against real-world datasets. Performance metrics such as detection accuracy, false favourable rates, and computational efficiency will be rigorously assessed to validate the proposed model against industry-standard IDS solutions.

By addressing critical gaps in existing IDS models, this project contributes to the broader field of cybersecurity by enhancing threat detection capabilities in encrypted environments while maintaining compliance with privacy and ethical standards. Future work will focus on expanding the system’s adaptability to evolving attack patterns and improving real-time response mechanisms. The outcomes of this research are expected to provide a robust, scalable, and industry-relevant NIDS framework capable of mitigating modern cyber threats in encrypted network traffic scenarios.

# References (generated using Mendeley)

Alalmaie, A. (2023) *Zero Trust with Guaranteed Accuracy Architecture Implementation for Intrusion Detection Systems (ZTA-IDS)*.

Alalmaie, A.Z., Nanda, P. and He, T.X. (no date) *ZT-NIDS: Zero Trust-Network Intrusion Detection System Validation based on Attack Simulations*. Available at: https://orcid.org/0000-0001-8962-540X.

Alevizos, L. *et al.* (2022) ‘Blockchain-Enabled Intrusion Detection and Prevention System of APTs Within Zero Trust Architecture’, *IEEE Access*, 10, pp. 89270–89288. Available at: https://doi.org/10.1109/ACCESS.2022.3200165.

Almajed, H.N. and Almogren, A.S. (2019) ‘SE-Enc: A Secure and Efficient Encoding Scheme Using Elliptic Curve Cryptography’, *IEEE Access*, 7, pp. 175865–175878. Available at: https://doi.org/10.1109/ACCESS.2019.2957943.

Bivens, A. *et al.* (no date) *NETWORK-BASED INTRUSION DETECTION USING NEURAL NETWORKS*.

EFE, A. and ABACI, İ.N. (2022) ‘Comparison of the Host Based Intrusion Detection Systems and Network Based Intrusion Detection Systems’, *Celal Bayar Üniversitesi Fen Bilimleri Dergisi*, 18(1), pp. 23–32. Available at: https://doi.org/10.18466/cbayarfbe.832533.

Kumar, S., Gupta, S. and Arora, S. (2021) ‘Research Trends in Network-Based Intrusion Detection Systems: A Review’, *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., pp. 157761–157779. Available at: https://doi.org/10.1109/ACCESS.2021.3129775.

Madiba, M.S. and Velempini, M. (2023) ‘Evaluation of Intrusion Detection System for the Distributed Denial of Service Attack on Internet of Things in Fog Computing Environment’, in *IEEE AFRICON Conference*. Institute of Electrical and Electronics Engineers Inc. Available at: https://doi.org/10.1109/AFRICON55910.2023.10293548.

‘Network Traffic Analysis for Intrusion with Zero-Trust’ (2024) *International Research Journal of Modernization in Engineering Technology and Science* [Preprint]. Available at: https://doi.org/10.56726/irjmets56206.

Of Homeland Security, D., Directorate, T.S. and Security Division, C. (2012) *The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research*.

Oliveira, N. *et al.* (2021) ‘Intelligent cyber attack detection and classification for network-based intrusion detection systems’, *Applied Sciences (Switzerland)*, 11(4), pp. 1–21. Available at: https://doi.org/10.3390/app11041674.

Panchal, R.K., Snehkunj, R. and Panchal, V. V (2024) ‘A Survey on Network-based Intrusion Detection System using Learning Techniques’, in. Institute of Electrical and Electronics Engineers (IEEE), pp. 740–747. Available at: https://doi.org/10.1109/icipcn63822.2024.00128.

Papadogiannaki, E. and Ioannidis, S. (2021) ‘Acceleration of intrusion detection in encrypted network traffic using heterogeneous hardware’, *Sensors (Switzerland)*, 21(4), pp. 1–21. Available at: https://doi.org/10.3390/s21041140.

Papadogiannaki, E., Tsirantonakis, G. and Ioannidis, S. (2022) ‘Network Intrusion Detection in Encrypted Traffic’, in *5th IEEE Conference on Dependable and Secure Computing, DSC 2022 and SECSOC 2022 Workshop, PASS4IoT 2022 Workshop SICSA International Paper/Poster Competition in Cybersecurity*. Institute of Electrical and Electronics Engineers Inc. Available at: https://doi.org/10.1109/DSC54232.2022.9888942.

Vigna, G. and Kemmerer, R.A. (no date) *NetSTAT: A Network-based Intrusion Detection Approach*.

# Appendices (Supporting Evidence)

## Appendix A: SNORT Validation

**Overview**

As part of the literature review, I have implemented and analysed SNORT, a renowned NIDS. SNORT is a signature-based (like an antivirus) NIDS. It focuses on detecting known attack patterns. It is a lightweight NIDS.   
   
SNORT has a built-in rule or signatures database of known attacks, allowing users to add additional custom rules. SNORT initially listens to the network traffic via the network adapter. It checks whether the real-time network traffic matches rules or signatures defined in the rule database. If it finds a match, an alert has been generated which is logged in /var/log/snort/snort.log, can be seen on console using –A console flag and can be sent to a SIEM like Splunk for further analysis.

**Lab Environment**

The lab environment has been set up using VMs in Oracle Virtual Box.   
1. Ubuntu VM (SNORT server/also mimics a victim) - IP: 192.168.56.102

2. Kali VM (attacker) - IP: 192.168.56.105

3. Ubuntu VM (mimics any user in a network) - IP: 192.168.56.1

I have performed and detected the following attacks Via SNORT.   
**1. RECON using PING**

Regular Ubuntu VM (192.168.56.1) pings Kali VM (192.168.56.105) and the SNORT server (192.168.56.102) detects it and generates alerts.

A screenshot of a computer

AI-generated content may be incorrect.

**2. RECON: Port Scanning**

Attacker VM (IP: 192.168.56.105) is doing a port scan on the regular Ubuntu user (IP: 192.168.56.1) and the SNORT server (IP: 192.168.56.102) detects it, generates relevant alerts.

A screenshot of a computer

AI-generated content may be incorrect.

**3. ssh BruteForce Attempt:**

I have defined a custom rule for attack responses. Rules files in /etc/snort/rules, as there is no predefined rule in SNORT to detect the SSH bruteforce.

**Rule:**   
Alert tcp any any -> any 22 (msg: "ET POLICY Possible SSH Brute Force Attack"; flow:to\_server, established; threshold: type threshold, track by\_src, count 5, seconds 30; sid:2001219; rev:5;)

**Breakdown:**

Template

alert <protocol> <source IP> <source port> -> <destination IP> <destination port> (msg:"<Custom Alert Message>"; flow:<direction>; content:"<string\_to\_match>"; threshold:type threshold, track by\_src, count <number>, seconds <time>; sid:<unique\_id>; rev:<revision\_number>;)

If five or more SSH connection attempts are made from the same source IP (by\_src) within 30 seconds, trigger an alert.

**Scenario:**

Using hydra, Kali VM (192.168.56.105) does SSH brute-force attack on the SNORT server (192.168.56.102).  SNORT server detects it and generates the alerts.

A screenshot of a computer

AI-generated content may be incorrect.

## Appendix B: Feature Extraction

**Feature Extraction**

**Joy** is an open-source network traffic analysis tool developed by **Cisco**. It is designed to extract detailed features from network packets, focusing on **flow-based analysis** rather than deep packet inspection. Joy is widely used for **network forensics, anomaly detection, and cybersecurity research**.

The joy tool provides multiple options for **network traffic feature extraction**. Below is a categorised list of commonly used options

|  |  |
| --- | --- |
| Dist=1 | Captures byte distribution in packets |
| Entropy=1 | Measures randomness in packet data |
| Hd=1 | Extracts header-only data (excludes payload) |
| Dns=1 | Extracts DNS metadata |
| Ssh=1 | Capture ssh features |
| Tls=1 | Extracts TLS handshake metadata |
| Bidir=1 | Bidirectional flow tracking for inter-arrival times of packets |

The features have been extracted from the pcap file using the following command:

joy bidir=1 tls=1 dist=1 output=output.json < input.pcap 

And a piece from the output (well formatted) JSON file is

Joy Configuration:

{

  "version": "4.5.0",

  "interface": "none",

  "promisc": 0,

  "output": "none",

  "num\_pkts": 50,

  "bidir": 1,

  "dist": 1,

  "tls": 1

}

Sample Flow Analysis

{   
"source\_ip": "192.168.56.105",   
"dest\_ip": "192.168.56.100",   
"protocol": 17,   
"source\_port": 68,   
"dest\_port": 67,   
"bytes\_out": 282,   
"num\_pkts\_out": 1,   
"bytes\_in": 548,   
"num\_pkts\_in": 1,   
"time\_start": 1740261914.018305,   
"time\_end": 1740261914.030945,   
"packets": [    
 {"b": 282, "dir": ">", "ipt": 0},   
 {"b": 548, "dir": "<", "ipt": 12}   
],   
"byte\_dist\_mean": 11.2,   
"byte\_dist\_std": 29.04,   
"ttl\_out": 64,   
"ttl\_in": 255   
}

Observations:

* This appears to be **DHCP traffic (UDP port 67/68)**.
* **TTL difference** (64 → 255) suggests that it is a **DHCP request from a client (192.168.56.105) to a DHCP server (192.168.56.100)**.
* **1 outgoing packet (282 bytes), one incoming packet (548 bytes).**
* **Not malicious**, just network configuration traffic.

Suspicious TCP Traffic:

 {   
 "source\_ip": "192.168.56.105",   
 "dest\_ip": "192.168.56.102",   
 "protocol": 6,   
 "source\_port": 33217,   
 "dest\_port": 8888,   
 "bytes\_out": 0,   
 "num\_pkts\_out": 1,   
 "bytes\_in": 0,   
 "num\_pkts\_in": 1,   
 "time\_start": 1740261933.858171,   
 "time\_end": 1740261933.858171,   
 "tcp": {    
 "flags": "S",   
 "first\_window\_size": 1024,   
 "options": [{"mss": 1460}]    
 }   
}

Observations (Possible Port scan):

* The **source (192.168.56.105) sends SYN packets to multiple ports on 192.168.56.102 (e.g., 8888, 993, 135, 256, 445, 21, 139, 110).**
* **No data is transferred (bytes\_out = 0, bytes\_in = 0)** → Could be an **active scanning attempt**.
* **Ports targeted:**
* 8888 → Often used for web apps.
* 993 → IMAP over SSL (Email access).
* 135 → Windows RPC.
* 256 → Unassigned.
* 445 → SMB file sharing (Common in exploits).
* 21 → FTP.
* 139 → NetBIOS.
* 110 → POP3 email.

## Appendix C: Gantt Chart

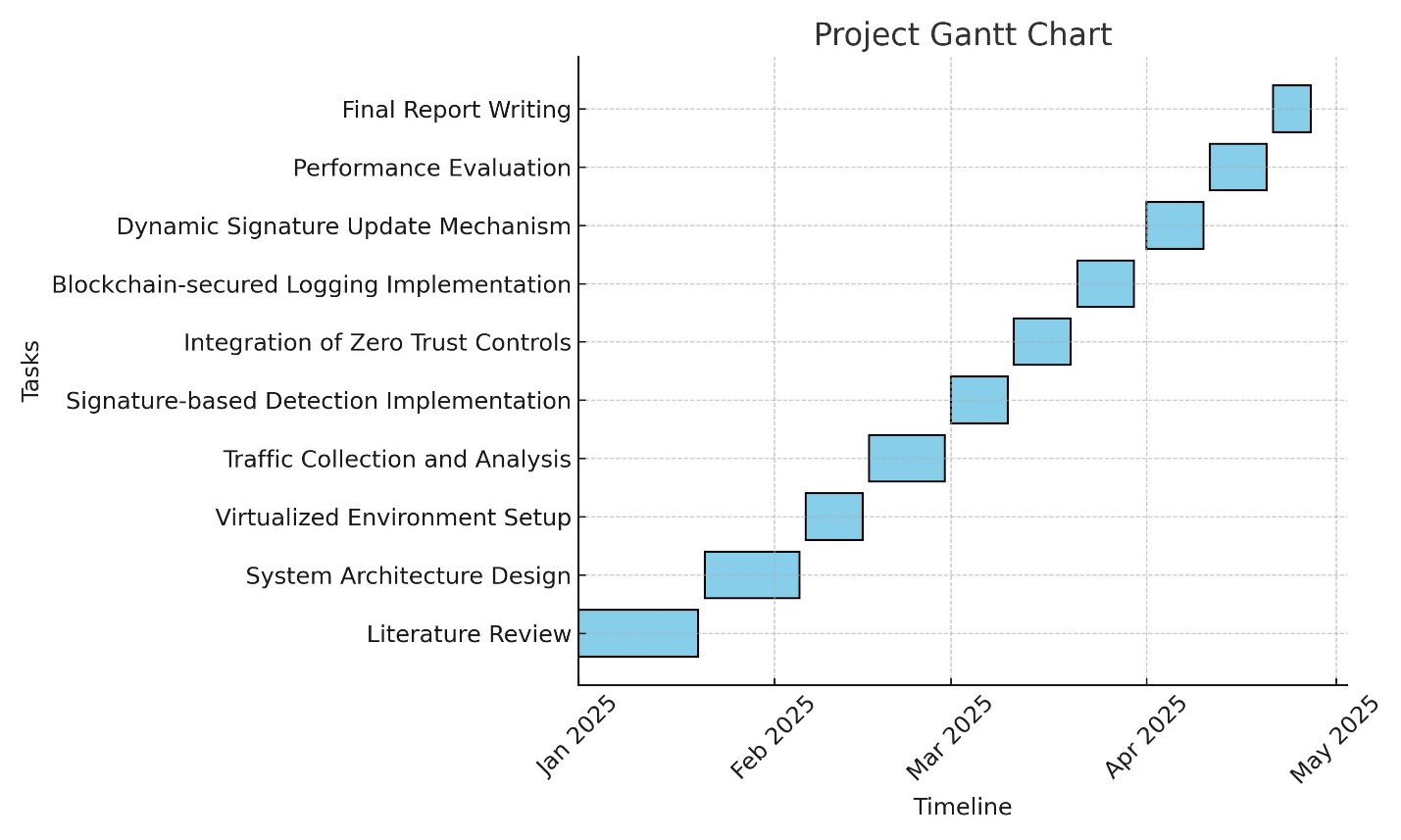


Figure Project Gantt Chart