ADAPTIVE CRYPTOGRAPHY: A TRANSFORMER NEURAL NETWORK-BASED APPROACH FOR ANOMALY DETECTION AND SECURE MESSAGING WITH SIGNREENCRYPTION

TEE JUNN JEH

UNIVERSITI TUNKU ABDUL RAHMAN

ADAPTIVE CRYPTOGRAPHY: A TRANSFORMER NEURAL NETWORK-BASED APPROACH FOR ANOMALY DETECTION AND SECURE MESSAGING WITH SIGNREENCRYPTION

TEE JUNN JEH

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Software Engineering (Honours)

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

September 2025

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Name TEE JUNN JEH

ID No. : 2105387

Date : 12/9/2025

COPYRIGHT STATEMENT

ABSTRACT

The increasing sophistication of cyber threats, particularly in decentralized and resource-constrained environments such as the Internet of Things (IoT), demands adaptive and efficient security solutions. This study introduces SignReencryption, a unified framework that integrates signcryption, proxy re-encryption (PRE), and Transformer-based intrusion detection to deliver both cryptographic assurance and intelligent adaptability. Signcryption ensures confidentiality and authenticity in a single lightweight operation, while PRE enables scalable, fine-grained access control without exposing plaintext. A TabTransformer-based intrusion detection system complements these cryptographic mechanisms, achieving classification accuracies of 94% on CICIDS2017, 99% on CIDDS-001, and 97% on NSL-KDD, with particular strength in detecting minority attack classes traditionally overlooked by baseline models. Optuna-driven hyperparameter optimization revealed dataset-specific configurations, demonstrating the adaptability of the TabTransformer across heterogeneous traffic distributions. Experimental evaluation further shows that SignReencryption reduces ciphertext expansion by up to 50% and lowers per-message execution time by nearly half compared to conventional Sign-Then-Encrypt schemes, confirming its practicality for real-time and bandwidth-limited environments such as intelligent transportation systems. Overall, the framework advances intrusion detection by uniting cryptographic efficiency with adaptive intelligence, offering a scalable, resilient, and operationally viable defense model for modern cybersecurity challenges.

Keywords: Signcryption; Cryptography; Transformer Neural Network; Intrusion Detection System; Internet of Things

Subject Area: QA75.5-76.95

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

APTs	advanced persistent threats
PRE	proxy re-encryption
TNN	Transformer Neural Networks
IoT	Internet of Things
VANETs	Vehicular Ad Hoc Networks
EtS	Encrypt-then-Sign
StE	Sign-then-Encrypt
CPS	cyber-physical systems
SPSR-VCP	Secure and Privacy preserving SignRecryption in Vehicular Cyber Physical system
BAN	Burrows-Abadi-Needham
AVISPA	Automated Validation of Internet Security Protocols and Applications
NLP	natural language processing
RNNs	Recurrent neural networks
LSTM	long short-term memory
BERT	Bidirectional Encoder Representations from Transformers
GPT	Generative Pretrained Transformer
AI	Artificial Intelligence
FFN	feed-forward network
IDS	Intrusion Detection System
SIEM	Security Information and Event Management
SSH	Secure Shell
FTP	File Transfer Protocol
DoS	Denial-of-Service
XSS	Cross Site Scripting
WBS	Work Breakdown Structure
CICIDS2017	Canadian Institute for Cybersecurity Intrusion Detection System 2017 dataset
CIDDS-001	Coburg Intrusion Detection Data Set 001
NSL-KDD	Improved Version of the KDD’99 dataset

CSV	Comma-Separated Values
IP	Internet Protocol
pcap	packet capture
ICMP	Internet Control Message Protocol
U2R	User to Root
R2L	Remote to Local
HTTP	HyperText Transfer Protocol
SQL	Structured Query Language
SMOTE	Synthetic Minority Oversampling Technique
r	Pearson correlation coefficient
DNN	Deep Neural Networks
CNN	Convolutional Neural Networks
RF	Random Forest
XGBoost	Extreme Gradient Boosting
CSE-IDS	Cost-Sensitive Deep Learning and Ensemble algorithms Intrusion Detection System
LIO-IDS	Long Short-Term Memory classifier and Improved One-vs-One technique Intrusion Detection System
AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
PKCS#7	Public Key Cryptography Standards #7 (padding scheme)
SHA-256	Secure Hash Algorithm 256-bit
ITS	Intelligent Transportation Systems
G_T	Target Group in bilinear pairing (group of order p)
G_1, G_2	Cyclic groups in bilinear pairing of order p
$\mathbb{Z}_{\mathbf{p}}$	Finite field of integers modulo prime p
g	Generator element in group G_1
g_2	Generator element in group G_2
γ	Master secret exponent in $\mathbb{Z}$p
P_{pub}	Public key component (g_2^{γ})
sk	User's private/secret key $\mathbb{Z}_{\mathbf{p}}$
pk	User's public key ($g^{sk} \in G_1$)
k_1	Ephemeral (one-time) session key element in G_1
e	Bilinear map (e: $G_1 \times G_2 \rightarrow G_T$)

SwiGLU	Switchable Gated Linear Unit
ReLU	Rectified Linear Unit
Add-Norm	Addition and Normalization layer (residual connection component in Transformers)
ms	millisecond

CHAPTER 1

INTRODUCTION

1.1 General Introduction

The rapid evolution of cyber threats such as advanced persistent threats (APTs), ransomware, and state-sponsored attacks has rendered traditional cybersecurity measures inadequate. These threats continue to adapt, bypassing static security protocols and requiring increasingly sophisticated defence strategies. One such defence mechanism is adaptive cryptography, which dynamically adjusts to evolving threats. Static cryptographic methods often fail to meet the demands of modern cyber threats, making adaptive cryptographic strategies crucial for addressing dynamic risks.

SignReencryption, a synthesis of signcryption, proxy re-encryption (PRE), and Transformer Neural Networks (TNN), offers a more robust framework for secure data transmission. It enables the re-encryption of a previously signcrypted message without decryption, thus increasing efficiency and security (Ateniese *et al.*, 2005). This combination allows cryptographic systems to evolve in response to emerging threats, ensuring both confidentiality and authenticity in real-time communications.

This research examines signcryption, PRE, and TNN as individual elements that, when integrated, offer adaptive security mechanisms, providing enhanced data protection and efficiency in modern digital communication systems.

1.2 Importance of the Study

The growing sophistication and volume of cyber threats such as advanced persistent threats (APTs), ransomware, and zero-day attacks pose significant challenges to traditional cryptographic systems. These systems, often static in nature, are ill-equipped to handle the evolving and dynamic nature of modern cyberattacks. As cyber threats continue to adapt, the need for more dynamic, adaptive cryptographic solutions has become increasingly evident.

This study is critical because it introduces an innovative approach by integrating signcryption, proxy re-encryption (PRE), and Transformer Neural Networks (TNN) to create an adaptive cryptographic framework. The integration of these three technologies offers a dynamic response to evolving threats, ensuring both security and efficiency in data transmission.

1. Signcryption enables confidentiality and authenticity in a single step, reducing computational overhead which is an essential feature for resource-constrained environments (Kanchan et al., 2019).
2. PRE allows data to be re-encrypted by a semi-trusted proxy without decrypting it, improving data sharing and access control in decentralised networks, such as the Internet of Things (IoT) (Ateniese et al., 2005).
3. TNN provides real-time threat adaptation based on emerging attack patterns and improving network security through continuous learning (Heaton, 2018).

The combination of these techniques offers enhanced security by adapting to evolving threats, optimising computational efficiency, and ensuring scalable data sharing across distributed environments. This approach is especially significant for sectors like IoT and decentralised networks, where data security and flexible access control are paramount.

The study's results will contribute to the development of more adaptive and scalable cryptographic systems that can respond to emerging cyber threats in real-time, offering proactive protection against new attack techniques. As such, this study is not only important for improving the security of distributed systems but also for ensuring that cryptographic solutions can evolve with the increasingly sophisticated nature of cybersecurity challenges.

1.3 Problem Statement

The increasing sophistication and frequency of cyber-attacks pose significant challenges to traditional cryptographic systems. Advanced persistent threats (APTs), zero-day exploits, and ransomware continue to outpace conventional security defences, highlighting the need for adaptive cryptographic solutions that can effectively address evolving threats. Traditional cryptographic methods, relying on static encryption techniques, struggle to provide the necessary flexibility, scalability, and efficiency in dynamic environments such as IoT networks and decentralised systems. The key challenges faced by current cryptographic systems are as follows:

1. **Inability to Adapt to Emerging Cyber Threats:**

   Traditional systems are static and unable to dynamically respond to new and evolving cyber threats. As attackers continuously refine their strategies, conventional cryptographic systems often fail to protect data from new vulnerabilities. Without the ability to adapt to emerging threats, these systems leave sensitive data exposed to advanced cyber-attacks.

2. **Scalability Issues in Decentralised Systems:**

   As decentralised networks like IoT grow, traditional encryption methods struggle with scalability. These systems face significant challenges in managing access control and data sharing across a growing number of users, devices, and applications. The increasing complexity of these systems makes it difficult to maintain efficient key management and enforce flexible access policies without compromising security.

3. **Inefficient Key Management:**

Managing encryption keys in large-scale decentralised systems remains a major obstacle. As the number of users and devices increases, key distribution and management become inefficient and prone to errors. This inefficiency can lead to security vulnerabilities, as poorly managed keys may result in unauthorised data access or decryption.

4. **Lack of Real-Time Adaptation:**

   Existing cryptographic systems fail to adapt in real time to new threats or network conditions. Traditional systems typically operate with predefined settings and lack of parallel processing features based on changing circumstances or observed attacks. This lack of real-time adaptation leaves systems vulnerable to novel threats that do not fit predefined patterns.

1.4 Aim and Objectives

Aim: This study aims to develop a robust adaptive cryptographic framework that enhances cybersecurity by integrating machine learning and advanced cryptographic techniques. The framework will enable parallel processing for identifying the category of events to address evolving network security challenges, ensuring efficient, scalable, and secure data transmission.

Objectives:

1. **Developing an Adaptive Cryptographic Framework**

   Create a framework that combines machine learning and advanced cryptographic techniques, such as signcryption, proxy re-encryption (PRE), and transformer neural networks (TNN), to enhance cybersecurity. This framework will adapt dynamically to emerging cyber threats and improve overall security and efficiency.

2. **Integration of Transformer Neural Networks**

Integrate TNN to provide real-time threat detection and adaptive cryptography, enabling parallel processing for identifying the category of events based on contextual network behaviour and evolving threats.

3. **Minimising Computational Overhead**

   Utilise proxy re-encryption (PRE) to maintain encrypted communications across authorised users while preventing unauthorised access and data breaches. This minimises computational overhead by delegating re-encryption tasks to a proxy, thus optimising efficiency in decentralised systems.

4. **Securing Communication Channels**
   Ensure secure and efficient communication across various channels, particularly for industries dealing with sensitive data such as healthcare, finance, and government. The framework will be designed to offer robust protection for data sharing and communication, crucial for sectors requiring high levels of security.

1.5 Scope and Limitation of the Study

Scope:

This study focuses on developing and evaluating an adaptive cryptographic framework that integrates signcryption, proxy re-encryption (PRE), and transformer neural networks (TNN) to enhance cybersecurity in decentralised systems, such as Internet of Things (IoT) networks. The research will explore how these three technologies can work together to improve data confidentiality, integrity, and authenticity in environments that require flexible and scalable security solutions.

Key areas of focus in the study include:

- **Real-time threat adaptation** using TNN, enabling the cryptographic framework to parallel processing and identify the activities categories based on observed threats and contextual network behaviour.

- **Minimisation of computational overhead** through the use of signcryption to combine encryption and digital signature functions, improving efficiency while maintaining robust security.
- **Secure and scalable data sharing with PRE**, which allows encrypted data to be securely re-encrypted by an intermediary (the proxy) without exposing plaintext data to unauthorised parties.

Additionally, the study will assess the scalability of these cryptographic techniques in decentralised environments and evaluate how well they adapt to emerging cybersecurity threats. The focus will be on developing a scalable solution capable of maintaining security while optimising performance in large, dynamic networks.

Limitations:

While this study aims to provide a comprehensive solution for adaptive cryptography, certain limitations should be acknowledged:

1. **Limited Testing Scope:**
    - The study will primarily evaluate theoretical models and prototypes of signcryption, PRE, and TNN. It may not involve exhaustive testing in large-scale production environments or across highly distributed systems. As a result, real-world challenges related to system deployment and integration may not be fully addressed within the scope of this study.
2. **Novelty of TNN Integration:**
    - The integration of TNN for adaptive security is a novel approach within the context of cryptographic systems. While the approach shows promise, its real-world performance in adapting to emerging threats will be explored through simulated environments rather than large-scale, live environments. The testing methodology will be based on simulations, which may not account for all variables present in actual deployment scenarios.
3. **Emerging Cryptographic Technologies:**

   - Due to the emerging nature of post-quantum cryptography and other advanced encryption protocols, this study may not fully address quantum-resistant methods or potential advancements in cryptography beyond the current focus. The integration of quantum-safe algorithms is not a priority for this research, and it is acknowledged that future advancements in this field could alter the approach outlined in this study.

4. **Compatibility with Legacy Systems:**
   - The proposed system may face compatibility issues when deployed in existing decentralised infrastructures, especially those using legacy cryptographic systems. While the framework is designed to be scalable and flexible, integrating with existing systems, particularly those with outdated or incompatible cryptographic protocols, could present challenges. This will be considered as a limitation in terms of deployment feasibility in certain environments.

5. **Focus on Selected Applications:**
   - The study will focus on the applications of this adaptive cryptographic framework within IoT and decentralised networks, with an emphasis on real-time threat adaptation and secure data sharing. While the framework is designed to be adaptable, its broader application across other domains of cybersecurity, such as cloud computing or enterprise networks, may require further research and adaptation.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction to SignReencryption

Cryptography is the foundation of secure communication, ensuring confidentiality, integrity, and authenticity of data in digital environments. As cyber threats evolve, modern cryptographic techniques integrate multiple security mechanisms to enhance protection. Conventional cryptographic approaches rely on encryption for confidentiality, digital signatures for authenticity, and key management protocols for secure communication (Menezes, van Oorschot and Vanstone, 2018). However, emerging challenges in network security, such as adaptive cyber threats and scalability concerns, necessitate advanced cryptographic solutions that combine multiple security primitives efficiently.

SignReencryption, a synthesis of signcryption, proxy re-encryption, and Transformer Neural Network, enhances security and efficiency in data transmission. It allows re-encrypting a previously signcrypted message without decrypting it first(Ateniese *et al.*, 2005), highlighting the necessity for adaptive cybersecurity strategies. This study explores three key cryptographic elements: signcryption, proxy re-encryption (PRE), and transformer neural networks (TNN) for adaptive security. Signcryption is a cryptographic scheme that simultaneously performs encryption and digital signature functions in a single operation, reducing computational overhead while ensuring both confidentiality and authenticity(Kanchan, Singh and Chaudhari, 2019). This efficiency makes it a valuable technique for secure communication in resource-constrained environments. Proxy re-encryption (PRE) allows a semi-trusted proxy to convert encrypted data from one recipient to another without decrypting it, enabling secure data sharing in dynamic environments such as cloud computing and decentralized networks (Ateniese *et al.*, 2005) Lastly, transformer neural networks (TNN) for adaptive security apply machine learning models to parallel processing and identify the category of event based on contextual threats, allowing systems to enhance real-time threat adaptation and detection (Heaton, 2018).

Traditional encryption methods face efficiency and scalability challenges, particularly in decentralised networks and resource-constrained environments (Kanchan, Singh and Chaudhari, 2019). Signcryption enhances efficiency, reducing computational overhead while maintaining strong security guarantees. Proxy re-encryption improves data flexibility, ensuring that access control policies remain secure in dynamic systems such as cloud computing and the Internet of Things (IoT). Additionally, transformer-based adaptive security mechanisms enable proactive threat mitigation by learning from network behaviour and categorise the event with the parallel processing feature.

By integrating these mechanisms, this study aims to develop a cryptographic framework that enhances security while optimising computational efficiency. The following sections provide a comprehensive analysis of existing research on these elements, highlighting advancements and potential areas for improvement.

2.2 Signcryption

Signcryption was initially introduced in 1997 by Yuliang Zheng, who claimed that his approach reduced computational costs by 50% and message expansion by 85% compared to the conventional method of applying digital signature followed by encryption. Zheng’s method was based on discrete logarithm cryptography and was proposed without formal security proofs (Zheng, 1997). In 1998, Zheng et al. extended this approach to elliptic curves, demonstrating a further reduction of 58% in computational cost and 40% in communication overhead (Zheng and Imai, 1998). However, due to the complexity of implementing elliptic curve signcryption, its practical adoption remained limited, particularly in environments such as VANET networks, where efficiency and scalability are critical.

Recent advancements have focused on enhancing the efficiency and applicability of signcryption for modern communication systems. Research by

Kanchan in 2018 introduced significant optimisations that improve computational performance, making signcryption more practical for resource-constrained environments (Kanchan and Chaudhari, 2018). Unlike Zheng's approach, which was challenging to implement in real-world applications, Kanchan's model streamlines cryptographic operations, reducing processing time and energy consumption. This makes signcryption viable for real-time communication systems, where speed and efficiency are crucial.

Additionally, Kanchan's work refines key management strategies and encryption mechanisms, mitigating delays associated with cryptographic computations. By optimising these operations, her approach ensures that signcryption remains relevant for next-generation secure communication applications, such as cloud computing and mobile transactions. In contrast to Zheng's signcryption model, which is now considered less practical due to its complex implementation and computational constraints, Kanchan's enhancements enable broader adoption in performance-sensitive environments.

Further refining her work, Kanchan introduced additional optimisations that specifically address the computational challenges of signcryption in real-world applications. Her approach minimises processing delays, ensuring that signcryption can be efficiently deployed in time-sensitive environments such as Vehicular Ad Hoc Networks (VANETs). By streamlining cryptographic operations, Kanchan's refinements enhance both security and efficiency, making signcryption more adaptable for secure communication in automated and decentralised networks(Kanchan, Singh and Chaudhari, 2019). These improvements make signcryption more practical for modern cybersecurity frameworks where real-time secure communication is a priority.

2.2.1 Comparison with Traditional Approaches

Prior to the development of signcryption, secure digital communication primarily relied on conventional paradigms such as **Encrypt-then-Sign (EtS)** and **Sign-then-Encrypt (StE)** (An, Dodis and Rabin, 2002). These approaches, while functionally sound, impose notable limitations in terms of computational efficiency and structural elegance.

i) **Encrypt-then-Sign (EtS):** This method first encrypts the message and then applies a digital signature to the encrypted output. While it ensures message integrity and confidentiality, it adds computational overhead due to separate cryptographic operations

ii) **Sign-then-Encrypt (StE):** Here, the message is first signed and then encrypted, ensuring authenticity before confidentiality. However, this approach may expose signature details to adversaries if not properly implemented, making it susceptible to certain attacks

Signcryption addresses these challenges by integrating digital signature and encryption into a single, unified operation. This not only preserves the essential security attributes like confidentiality, integrity, authentication, and non-repudiation, but also significantly reduces computational complexity and bandwidth requirements. Owing to these advantages, signcryption has emerged as a highly efficient and secure alternative, particularly well-suited for resource-constrained settings such as mobile devices, wireless networks, and IoT-based systems.

2.2.2 Why Signcryption is Essential

The convergence of digital signature and encryption operations into a unified cryptographic primitive, known as signcryption, has emerged as a vital advancement in ensuring security and privacy within cyber-physical systems (CPS), particularly in the context of Vehicular Ad Hoc Networks (VANETs). As demonstrated by Kanchan et al. (2021), signcryption plays a foundational role in addressing the growing demands of secure, real-time vehicular communication. Traditional schemes that treat signing and encryption as sequential, independent processes often incur higher computational costs and latency, which are impractical for latency-sensitive environments like intelligent transportation systems. By contrast, signcryption achieves confidentiality and authenticity simultaneously, thereby improving computational efficiency and reducing overall resource consumption.

One of the critical motivations behind adopting signcryption in VANETs is the need for privacy-preserving mechanisms that still allow for secure and authenticated communication. In the proposed SPSR-VCP protocol, Kanchan et al. (2021) integrated signcryption with group signature schemes, effectively masking the identity of individual vehicles while ensuring that transmitted messages originate from legitimate and trusted sources. This dual feature is particularly valuable for protecting sensitive information such as vehicular location, identity, and routing data, which, if exposed, could lead to serious security breaches including identity theft, location tracking, or even vehicular hijacking.

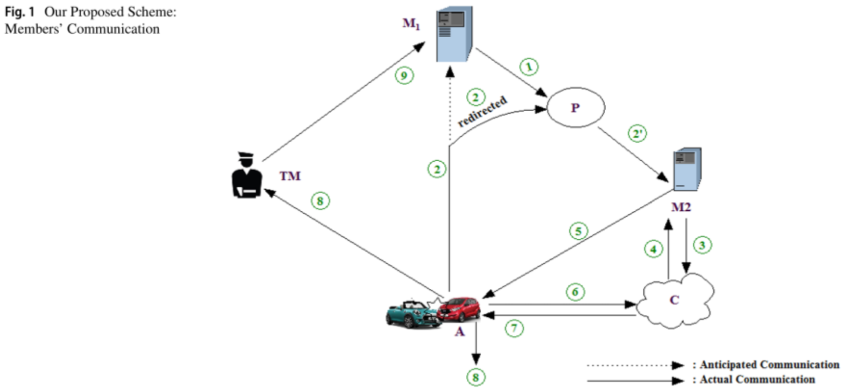


Figure 2.1: Members' Communication Flow

Source: (Kanchan, Singh and Chaudhari, 2021)

Moreover, the SPSR-VCP protocol introduces proxy re-encryption to support load distribution and fault tolerance. Here, a semi-trusted proxy is assigned the task of converting ciphertexts originally intended for one entity (e.g., the main membership manager) into ciphertexts compatible with an alternative manager. This approach ensures uninterrupted service even when primary nodes become overloaded or temporarily unavailable, thereby enhancing system robustness without compromising confidentiality. Importantly, the re-encryption keys used in the scheme are designed to be non-transitive, mitigating risks associated with unauthorised key derivation and key misuse which is a recognised vulnerability in proxy-based systems.

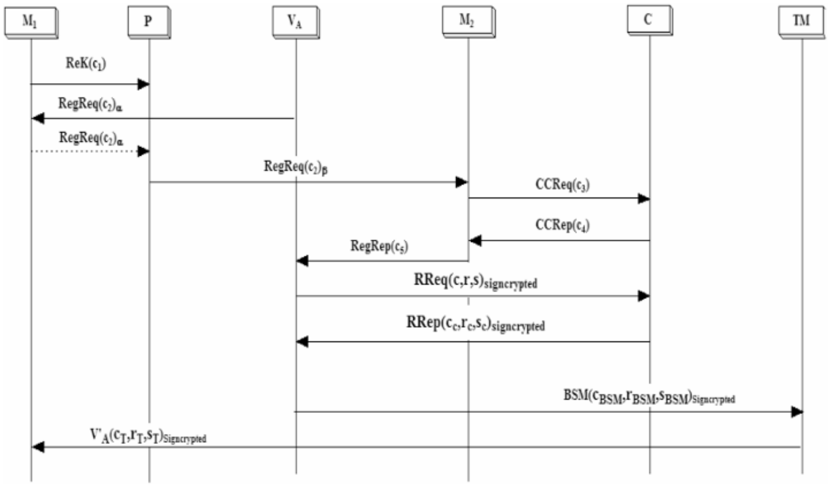


Figure 2.2: Proposed Authentication Protocol

Source: (Kanchan, Singh and Chaudhari, 2021)

The protocol further leverages cloud computing capabilities to perform complex operations, such as computing group keys or updating signature accumulators. Offloading such computationally intensive tasks to the cloud ensures that the vehicles, which may have limited processing power, are not burdened by cryptographic operations. This enhances the scalability of the system, allowing it to function efficiently even under high-volume communication scenarios typical in urban traffic systems.

From a security standpoint, the robustness of the signcryption-based SPSR-VCP protocol is formally verified using BAN logic and the AVISPA tool. The analysis confirms the protocol's resilience against a range of attacks, including impersonation, replay, Sybil, man-in-the-middle, and digital signature forgery. The integration of nonce-based freshness checks and authenticated encryption ensures that replay and duplication of messages are effectively prevented, while the use of traceable group signatures allows authorised entities, such as a tracing manager, to revoke misbehaving nodes without compromising the privacy of compliant ones.

In terms of performance, the protocol demonstrates notable gains. With a computational cost of only 5.675 milliseconds per message, it outperforms several comparable schemes, some of which exceed 12 milliseconds.

Additionally, the compact packet size of 57 bytes results in lower communication overhead, which is critical for high-speed and bandwidth-limited vehicular environments.

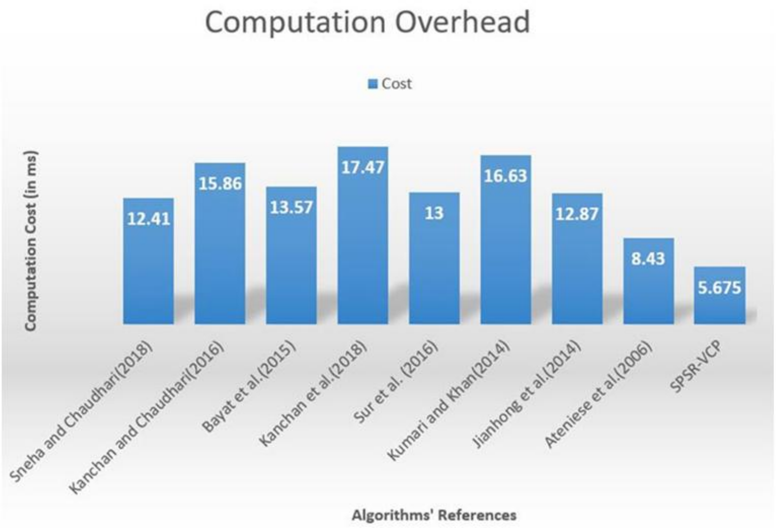


Figure 2.3: Computation cost of the algorithms

Source: (Kanchan, Singh and Chaudhari, 2021)

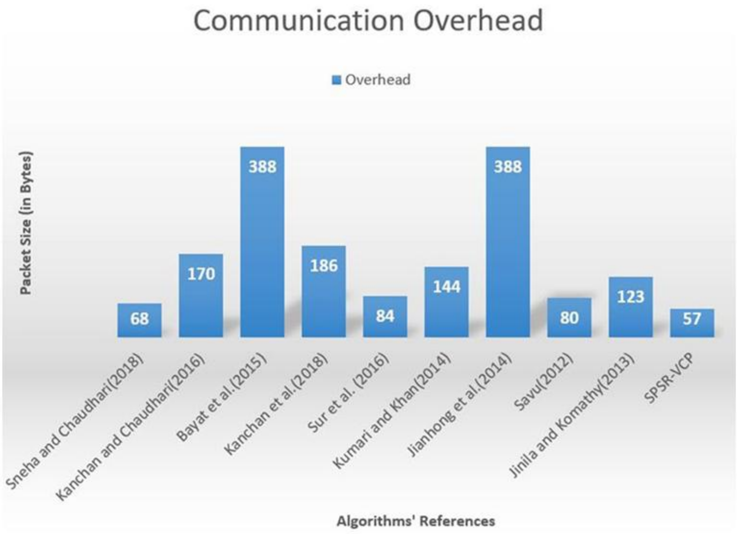


Figure 2.4: Final Packet-size with the comparison of the algorithms

Source: (Kanchan, Singh and Chaudhari, 2021)

In conclusion, the adoption of signcryption within the SPSR-VCP framework represents a significant step forward in the design of secure, efficient, and privacy-preserving communication protocols for vehicular cyber-physical systems. Its combined use of signcryption, group signatures, proxy re-encryption, and cloud computing not only addresses the critical challenges of modern VANETs but also sets a precedent for future research and real-world deployment of secure vehicular networks.

2.2.3 Challenges and Limitations of Signcryption

While **signcryption** offers significant advantages over traditional encryption methods, several challenges and limitations remain. These hurdles can affect the practicality and scalability of signcryption, particularly in large-scale or resource-constrained environments.

1. **Efficiency in Resource-Constrained Environments**

   Despite its inherent computational advantages, signcryption still presents efficiency challenges in resource-constrained environments, such as IoT devices, mobile networks, or low-power embedded systems. The cryptographic operations involved in signcryption can demand significant processing power and memory, which may overwhelm devices with limited resources. Although Kanchan et al. (2018) proposed efficient signcryption schemes, the need for constant key management, signature generation, and encryption still imposes performance limitations in environments with low CPU power and bandwidth constraints (Kanchan & Chaudhari, 2018). These limitations make it difficult to implement signcryption in scenarios where rapid execution and low power consumption are crucial.

2. **Key Management and Revocation**

   Efficient key management and the revocation of compromised keys are inherent challenges for any cryptographic system, and signcryption is no exception. In the context of signcryption, managing dynamic keys and ensuring that only valid, authorised users can access the network is crucial. Kanchan et al. (2018) propose a robust system for key distribution and revocation in vehicular networks, but the implementation of these mechanisms can still be complicated, especially when considering large networks with high mobility and frequent membership changes. The real-time revocation of keys and authentication processes must be streamlined to prevent any unauthorised access or compromised data from being transmitted, which remains a persistent challenge in practical applications.

2.3 Transformer Neural Network (TNN)

Transformer Neural Networks (TNNs) have had a profound impact on deep learning, particularly within the areas of sequence modelling and natural language processing (NLP). First introduced by Vaswani et al. (2017), the Transformer model marked a departure from earlier methods such as recurrent neural networks (RNNs) and long short-term memory (LSTM) networks by eliminating the need for recurrence. Instead of relying on sequential data handling, the Transformer leverages self-attention to effectively model complex relationships within the input. This structural innovation has greatly advanced machine learning capabilities in a range of applications, including automated translation, content generation, and understanding natural language.

One of the most distinctive aspects of the Transformer is the self-attention mechanism. This feature allows the model to evaluate how relevant each input token is in relation to every other token, regardless of their positions within the sequence. Unlike traditional methods that require processing in order, the self-attention approach enhances computational speed and allows for high levels of parallelisation, making it especially efficient when working with extensive or complex datasets (Vaswani et al., 2017).

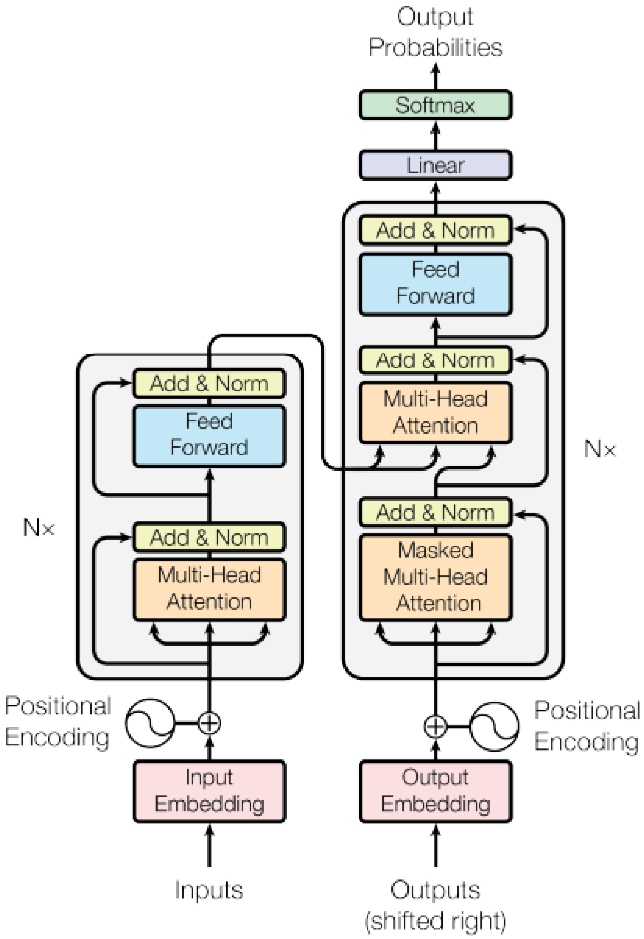


Figure 2.5: Transformer-model architecture

Source: (Vaswani *et al.*, 2017)

The importance of self-attention lies in its ability to capture contextual dependencies by dynamically learning which tokens should be emphasised or downplayed during training. Enhancing this, the multi-head attention technique enables the model to examine the input from multiple perspectives at once, thereby extracting features at different levels of granularity.

Since the introduction of the original Transformer model, several highly influential variants have been developed. Among them are BERT and GPT, which have both achieved remarkable performance on a wide range of NLP benchmarks (Devlin *et al.*, 2019). The flexibility and scale of Transformer-based architectures have also extended their relevance beyond language tasks, finding applications in areas such as image analysis, genomic data processing, and decision-making systems like reinforcement learning.

The Transformer’s capability to handle various input types in parallel and manage extensive data efficiently has positioned it as a cornerstone in the development of modern artificial intelligence systems. With continual advancements and the emergence of increasingly powerful models like GPT-3, this architecture is expected to remain central to the evolution of more intelligent and scalable machine learning technologies.

2.3.1 Core Components of TNN

This section outlines several fundamental elements that form the basis of knowledge in Transformer Neural Networks. The discussion will cover five key aspects that contribute to its power and effectiveness. (Vaswani et al., 2017).

2.3.1.1 Self-Attention Mechanism

Self-attention allows a Transformer to analyse how every token in a sequence relates to every other token simultaneously, unlike RNNs and LSTMs that handle data step by step. Each token is transformed into three separate vectors: query, key, and value based on its embedding. Attention scores are computed by taking the dot product of queries and keys, then applying a softmax function to normalise the results. These scores are used to weight the value vectors, producing an output for each token. This approach supports efficient parallel processing and helps capture dependencies over long distances in the input.

2.3.1.2 Multi-Head Attention

Multi-head attention extends the self-attention process by performing multiple attention operations concurrently, each using different parameter sets. This design allows the model to examine the input from several perspectives, capturing a wide range of relationships and patterns. Each head may detect different features, such as syntax or meaning, at various abstraction levels. Afterwards, the outputs from all heads are merged and passed through a linear transformation to generate the final result. This structure boosts the model’s ability to interpret complex data more effectively.

2.3.1.3 Positional Encoding

Since Transformers process input tokens all at once, they lack built-in awareness of token order. To address this, positional encodings are integrated into the token embeddings, offering cues about each token's place in the sequence. The original model uses sine and cosine functions to produce distinct patterns for each position. This enables the Transformer to learn the sequence structure and generalise to longer sequences than it was trained on, preserving performance and order awareness.

2.3.1.4 Feed-Forward Networks (FFN)

Once tokens have been processed through self-attention, each is individually passed through a feed-forward neural network. This FFN consists of two dense layers separated by a ReLU activation function, introducing non-linearity and enabling the model to detect complex patterns. Notably, the same FFN is applied to each token without considering others, enhancing computational efficiency. Despite this independence, the FFN refines token-level features and contributes to deeper representations.

2.3.1.5 Encoder-Decoder Architecture

The encoder-decoder setup is essential for handling tasks that require mapping an input sequence to an output, such as translating languages. The encoder converts the input into a continuous representation, which the decoder uses to generate the output. Both parts consist of several layers incorporating self-attention and feed-forward sublayers. The decoder includes an extra attention mechanism that targets the encoder's outputs, helping it focus on the relevant input during generation. To ensure proper sequence generation, masked self-attention is applied in the decoder to block access to future positions. This structure enables the model to learn both short- and long-term dependencies effectively.

2.3.2 Benefits of Transformer Neural Networks in Predicting Known, Partially Known and Unknown Metadata

Transformer Neural Networks (TNNs) have proven highly capable when working with various types of metadata, whether the data is fully known, partially complete, or entirely new. This versatility primarily stems from the self-attention mechanism, which allows the model to process sequences simultaneously and capture both short- and long-distance relationships within the data. Due to this architecture, TNNs often surpass traditional models like RNNs and LSTMs, especially in scenarios involving incomplete or unfamiliar data patterns.

2.3.2.1 Predicting Known Metadata

In cases where metadata is comprehensive and structured, TNNs excel by leveraging multi-head self-attention to interpret intricate connections among elements in the input. This approach enables the model to recognise both fine-grained and broad dependencies across the sequence. Unlike RNNs or LSTMs, which process input step by step, Transformers analyse entire sequences at once, leading to improved efficiency and understanding of contextual relationships.

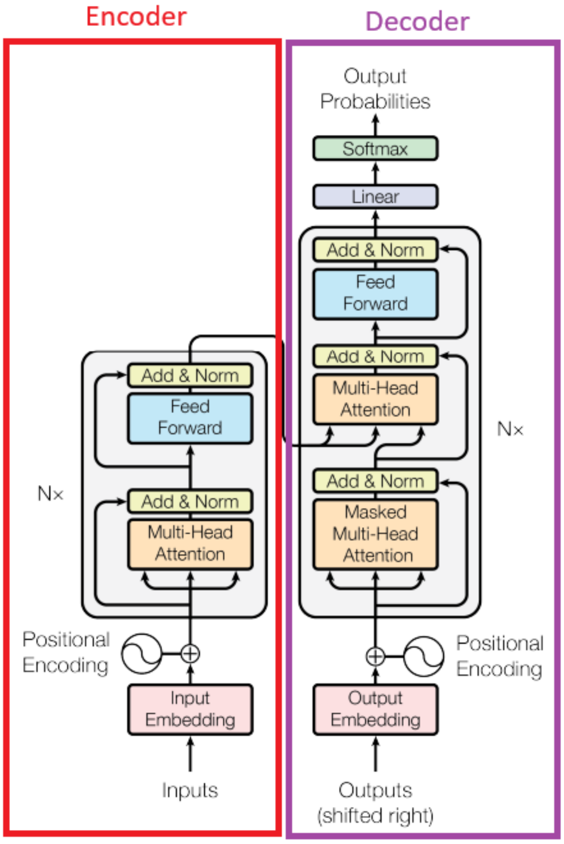


Figure 2.6: Transformer processes sequence in parallel

Source: (Vaswani *et al.*, 2017)

The original Transformer model, as introduced by Vaswani et al. (2017), demonstrated superior results in machine translation compared to traditional RNN-based approaches. Its effectiveness is largely attributed to the ability to manage long-term relationships and its capacity for parallel processing areas where RNNs and LSTMs often fall short due to their reliance on sequential data handling.

2.3.2.2 Predicting Partially Known Metadata

In many practical applications, datasets often contain gaps, referred to as partially known metadata. Traditional sequence models typically address this through methods like data imputation or gating mechanisms, which can be both complex and limited in utilising the available information. Transformers, particularly those trained with masked input strategies (as in BERT), are naturally suited to infer missing content using the surrounding tokens. For example, BERT employs a masked language modelling technique, where certain input tokens are hidden, and the model learns to predict them based on their context. This training method strengthens the model's ability to reconstruct missing or uncertain portions of the data, as shown in Figure 2.7.

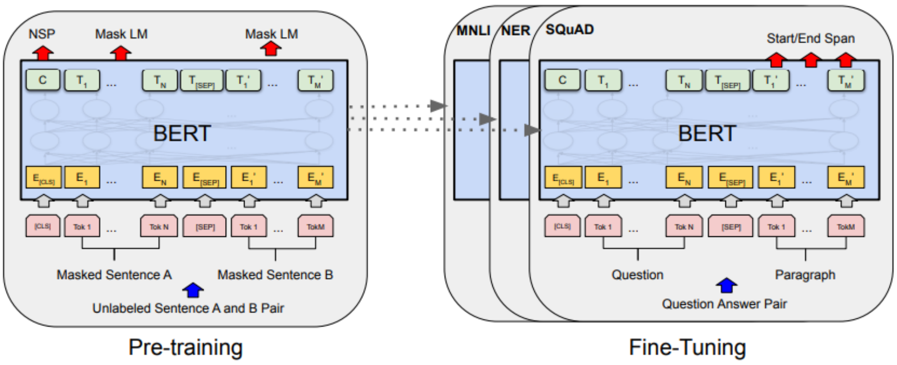


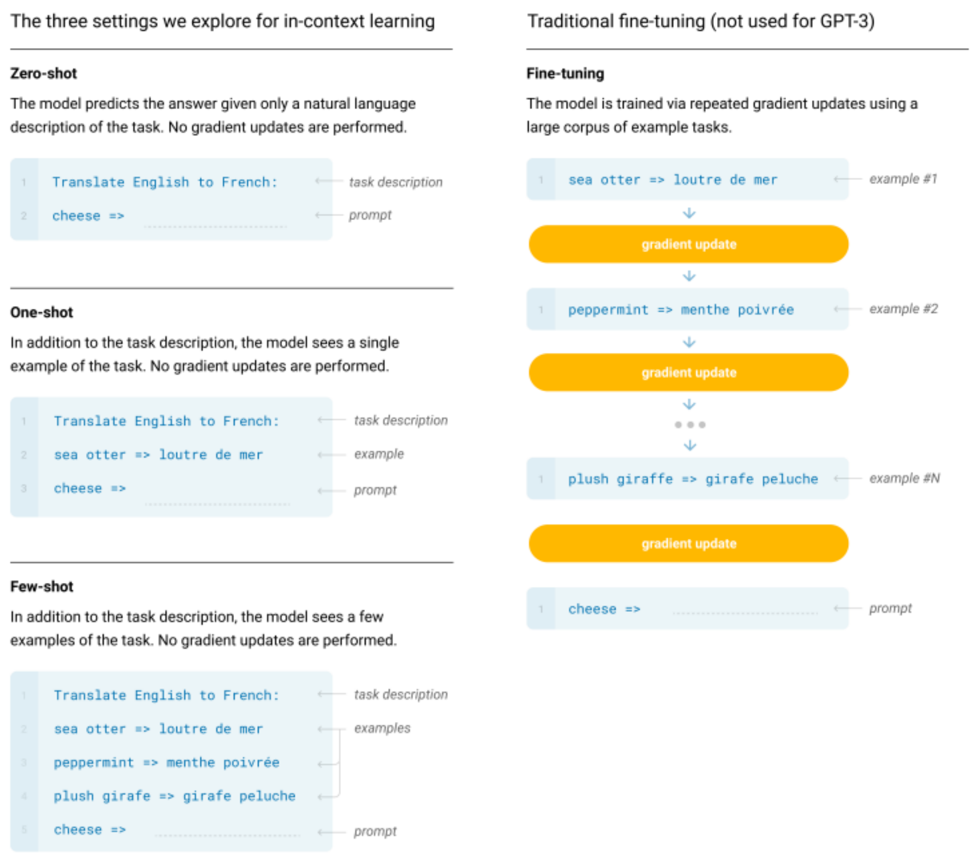
Figure 2.7: Overview of BERT's pre-training and fine-tuning process

Source: (Devlin *et al.*, 2019)

The model's bidirectional attention design is key to this functionality as it allows simultaneous reference to both preceding and following elements in the sequence. This offers a notable advantage over models like RNNs and LSTMs, which operate in a strictly forward or backward manner and may struggle to retain or utilise long-distance information when faced with partial inputs (Devlin *et al.*, 2019)

2.3.2.3 Predicting Unknown Metadata

One of the most remarkable capabilities of Transformer models is their ability to handle unknown metadata, or tasks involving data and contexts the model has not encountered during training. This capability is especially evident in pretrained models such as GPT-3. Unlike traditional models like RNNs and LSTMs, which typically require retraining or fine-tuning for each new task, Transformer models can generalise to new tasks with minimal task-specific training, using zero-shot or few-shot learning techniques. GPT-3, for example, with its 175 billion parameters, is capable of performing a wide variety of tasks, including question answering, text generation, and translation, with minimal input, as shown in Figure 2.8.



The three settings we explore for in-context learning

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
Translate English to French:
cheese =>
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
Translate English to French:
sea otter => loutre de mer
cheese =>
```

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
Translate English to French:
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese =>
```

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.

```
sea otter => loutre de mer
```

```
peppermint => menthe poivrée
```

```
plush giraffe => girafe peluche
```

```
cheese =>
```

Figure 2.8: Comparison of learning paradigms

Source: (Brown *et al.*, 2020)

The ability to perform these tasks without explicit retraining is a direct result of the large-scale pretraining and the flexibility of the Transformer architecture. Brown et al. (2020) demonstrated that GPT-3 can generate coherent and contextually appropriate outputs across various domains without requiring additional task-specific fine-tuning, a feature that significantly distinguishes it from traditional models such as RNNs and LSTMs.

2.3.3 Comparative Performance: Transformer vs Traditional Architectures

The following table summarises the key advantages of Transformer-based models over RNNs and LSTMs in terms of handling known, partially known, and unknown metadata:

Table 2.1: Comparative Analysis of RNN/LSTM and TNN Architectures Across Key Aspects

Aspect	RNN / LSTM	Transformer (TNN)
Handling Sequential Dependencies	Sequential, prone to vanishing gradients	Parallel, captures long-range dependencies without recurrence
Efficiency and Speed	Slow (sequential processing)	Fast (parallel processing, especially with GPUs)
Handling Missing Data	Requires explicit mechanisms (e.g., imputation)	Can infer missing data based on context (e.g., BERT)
Generalisation to Unseen Tasks	Needs retraining or fine-tuning	Excellent zero-shot/few-shot learning (e.g., GPT-3)
Performance on NLP Tasks	Outperformed by newer architectures	State-of-the-art on various benchmarks (e.g., translation, QA)
Model Scale	Limited by parameter size and architecture	Massive scale (e.g., GPT-3 with 175 billion parameters)

2.4 Proxy Re-encryption

Proxy Re-encryption (PRE) is a cryptographic technique designed to enhance secure data sharing in decentralised environments. It allows a semi-trusted intermediary, called a proxy, to re-encrypt data encrypted under one key to a different key without ever learning the underlying plaintext. This enables the proxy to perform its role without the need to decrypt the data, thus preserving the confidentiality of the information. PRE facilitates secure and efficient data delegation by allowing data owners to grant access to encrypted data to a third party, such as a cloud provider or another user, without exposing the original content (Ateniese et al., 2005).

The concept of Proxy Re-encryption was first introduced by Ateniese, et al., in 2005, and has since evolved with various improvements and optimisations. One of the key features of PRE is its ability to provide fine-grained access control in environments where data may need to be shared or transferred between different users or systems. The proxy, acting as an intermediary, can convert ciphertext from one recipient's encryption to another recipient's encryption, allowing the owner of the data to control access without requiring the re-encryption process to be done manually.

PRE is particularly useful in cloud computing and distributed systems, where users or organisations may want to securely share encrypted data with multiple recipients. In these systems, access control policies can change dynamically, and PRE enables seamless access management. For example, in a cloud storage environment, data owners may need to grant access to data to different parties over time, and PRE allows the data to be shared securely without re-encrypting the entire dataset for each new user.

Several variants of PRE have been developed to address different security and functionality requirements, such as unidirectional PRE, where data is re-encrypted in only one direction (from the sender to the recipient), and bidirectional PRE, where re-encryption can happen in both directions. Advanced schemes, including identity-based PRE and hierarchical PRE, have also emerged, providing greater flexibility and scalability for large systems (Goyal et al., 2006).

In addition to its ability to support secure data sharing, PRE also offers efficiency advantages, as it avoids the need for data decryption and re-encryption by the original owner. By delegating the re-encryption process to the proxy, systems using PRE can achieve significant performance improvements, particularly in environments where data needs to be accessed or transferred by multiple parties frequently.

As data-sharing requirements grow in distributed and cloud-based systems, the role of PRE in enabling secure, efficient, and scalable data sharing continues to expand. The ongoing development of more secure and efficient PRE schemes, as well as their integration with other cryptographic protocols, makes PRE a valuable tool for modern cybersecurity.

2.4.1 Key Concepts and Mechanisms in Proxy Re-encryption

Proxy Re-encryption (PRE) is a cryptographic technique designed to enhance secure data sharing in decentralised systems. It allows a semi-trusted intermediary (the proxy) to re-encrypt ciphertext from one recipient's encryption key to another's without decrypting the data, preserving the confidentiality of the information throughout the process (Ateniese *et al.*, 2005). This capability is significant because traditional encryption systems often require data to be decrypted before it can be re-encrypted for another recipient, which exposes the plaintext to the proxy. In contrast, PRE’s non-decrypting re-encryption feature ensures that the proxy cannot access the underlying data, thus preserving the confidentiality and integrity of the encrypted content (Ateniese et al., 2005).

One of the key features of PRE is its ability to provide dynamic access control in environments where data may need to be shared or transferred between different users or systems. In traditional encryption, if access needs to be granted to new users, the data owner must either manually re-encrypt the data for each user or provide access to decryption keys, which can be cumbersome and insecure. PRE addresses these challenges by allowing the proxy to re-encrypt the data on behalf of the owner, without exposing the plaintext or requiring direct interaction between the owner and each new recipient.

The proxy, acting as an intermediary, can convert ciphertext from one recipient's encryption key to another recipient's encryption key, enabling the data owner to control access to the encrypted data dynamically and securely. This eliminates the need for the owner to manually re-encrypt the data each time access needs to be granted or revoked. Additionally, the owner does not have to share their decryption keys with the proxy, thus ensuring that the proxy can perform its task without compromising the confidentiality of the data. This dynamic control over who can access the data and when is a significant advantage of PRE over traditional encryption mechanisms, especially in large, distributed systems where access control policies frequently change (Ateniese et al., 2005).

In the typical PRE mechanism, the sender encrypts the data under their own public key, and the proxy can then perform re-encryption using a special transformation key provided by the sender or the data owner. This transformation key allows the proxy to re-encrypt the ciphertext so that the recipient can decrypt it using their own private key. The re-encrypted ciphertext is forwarded to the recipient without the proxy ever accessing the plaintext, ensuring both confidentiality and flexibility in managing access (Ateniese *et al.*, 2005).

This dynamic access control feature makes PRE particularly useful in environments such as cloud computing, IoT, and decentralised networks, where data access needs to be controlled dynamically without overburdening the data owner or exposing sensitive information to intermediaries.

2.4.2 Variants of Proxy Re-encryption and Its Efficiency Advantages

Several variants of Proxy Re-encryption (PRE) have been developed to address different security and functionality requirements in various cryptographic applications. One such variant is unidirectional PRE, where the proxy can re-encrypt data in only one direction, typically from the sender's encryption to the recipient’s encryption. This design simplifies the process in scenarios where data is shared from a single source to a single destination, providing a basic but secure method for delegated access control (Ateniese *et al.*, 2005). In contrast, bidirectional PRE allows re-encryption to occur in both directions, meaning that the data can be re-encrypted from one user to another and vice versa. This bi-directionality enhances flexibility, particularly in systems where users need to share data in multiple directions or in collaborative environments, where access rights may change frequently (Ateniese et al., 2005).

In addition to these basic variants, advanced schemes such as identity-based PRE and hierarchical PRE have emerged to address the scalability and flexibility requirements of large systems. Identity-based PRE uses identity-based encryption (IBE) to derive public keys from user identities, streamlining key management and allowing the proxy to re-encrypt data without the need for explicit key distribution. This is particularly useful in systems where key management is a critical concern, such as in large-scale cloud environments (Goyal *et al.*, 2006). On the other hand, hierarchical PRE introduces multi-level access control, allowing for more sophisticated delegation of decryption rights in organisational structures, where different users or departments may need different levels of access to the encrypted data (Goyal *et al.*, 2006). These advanced schemes provide greater flexibility and scalability, enabling PRE to be effectively implemented in large, distributed environments, including cloud computing and enterprise networks.

Beyond its ability to support secure data sharing, PRE offers significant efficiency advantages, particularly in scenarios where data needs to be accessed or transferred by multiple parties. One of the most important advantages of PRE is that it avoids the need for data decryption and re-encryption by the original data owner. Instead, the re-encryption process is delegated to the proxy, which means that the data owner does not have to spend time or resources manually re-encrypting data every time access is required by a new party (Ateniese *et al.*, 2005). This delegation not only reduces the computational burden on the data owner but also enhances the efficiency of the system as a whole, especially in environments where data is frequently accessed or shared among a large number of recipients.

As data-sharing requirements continue to grow in distributed and cloud-based systems, the role of PRE in enabling secure, efficient, and scalable data sharing becomes increasingly critical. PRE enhances the ability of organisations to share sensitive data securely while maintaining control over access, even as systems scale up. The continued development of more secure and efficient PRE schemes, coupled with their integration with other cryptographic protocols (such as signcryption, attribute-based encryption, and blockchain), makes PRE an indispensable tool in modern cybersecurity (Ateniese *et al.*, 2005; Goyal *et al.*, 2006). The adaptability of PRE to handle various access control policies, combined with its efficiency and scalability, will ensure its widespread use in future cybersecurity architectures.

2.5 Related Works

The growing prevalence of sophisticated and diverse cyber threats has led to an extensive body of research on network intrusion detection systems (NIDS) that leverage machine learning and deep learning techniques. Recent studies have focused on addressing fundamental challenges such as class imbalance, detection latency, and model generalisability across heterogeneous network environments. This section reviews two closely related and representative approaches proposed by Gupta, Jindal and Bedi (2021), which specifically tackle the issue of class imbalance through architectural and algorithmic innovations. The discussion highlights the design principles, methodological contributions, and dataset utilisation in each study, followed by a synthesis that outlines their comparative insights and implications for the development of robust, real-time intrusion detection frameworks.

2.5.1 LIO-IDS: LSTM with an Improved One-vs-One Strategy for Class-Imbalanced NIDS

Gupta, Jindal and Bedi (2021) introduced *LIO-IDS*, a two-layer anomaly-based network intrusion detection system (NIDS) that integrates sequence modelling with an efficient multi-class decision scheme. The first layer distinguishes normal from malicious traffic using a long short-term memory (LSTM) network, while the second layer employs an Improved One-vs-One (I-OVO) ensemble to classify specific attack types. The improvement lies in reducing computational overhead by activating only three classifiers during inference, compared to the large number typically required by the standard OVO approach.

The study evaluated LIO-IDS using three widely adopted datasets: NSL-KDD, CIDDS-001, and CICIDS2017, reporting on accuracy, detection rates, and computational times. The authors addressed the problem of class imbalance, which is common in intrusion detection scenarios where majority and minority attack types differ significantly in frequency. They identified four main categories of imbalance-handling techniques and incorporated data-level rebalancing alongside ensemble learning to mitigate this issue (Gupta et al., 2021).

Methodologically, the model employed Random Forest and balanced Bagging, together with oversampling techniques such as Random Oversampling (ROS), Borderline-SMOTE, and SVM-SMOTE, to improve recall for under-represented attack types while maintaining computational efficiency. The I-OVO design strategically partitions classes into majority and minority groups, training two multi-class classifiers (C1 and C2) and using a single binary classifier (C3) to resolve ambiguous predictions. This structure effectively balances accuracy with reduced inference time.

The evaluation of the study relies on three well-established benchmark datasets: NSL-KDD, CIDDS-001, and CICIDS2017, with each contributing distinct characteristics that strengthen the comprehensiveness of the analysis. NSL-KDD offers a refined and de-duplicated version of the KDD’99 dataset, providing a balanced environment to evaluate detection and false alarm rates. CIDDS-001 adds realism through flow-based traffic derived from enterprise networks, while CICIDS2017 represents modern attack patterns and updated traffic features. Using these datasets together allows the authors to assess the generalisability of their model across different network environments and data distributions.

Despite its strengths, LIO-IDS depends heavily on the correct identification of majority and minority groups. As threat landscapes evolve, this grouping can become outdated. Furthermore, oversampling techniques, while improving recall, risk introducing noise if synthetic samples are generated near mislabeled or noisy decision boundaries, especially in legacy datasets such as NSL-KDD.

2.5.2 CSE-IDS: Cost-Sensitive Deep Learning with Staged Ensemble

In related research, Gupta, Jindal and Bedi (2021) also proposed CSE-IDS, a three-layer cost-sensitive anomaly-based NIDS that combines deep learning with ensemble methods. The first layer applies a cost-sensitive deep neural network (DNN) to filter normal and suspicious traffic, assigning higher misclassification costs to missed attacks. The second layer employs XGBoost to distinguish between normal traffic, majority attack types, and a pooled "minority" class, while the third layer refines this pooled class using a Random Forest model.

The evaluation was again conducted on NSL-KDD, CIDDS-001, and CICIDS2017, with performance reported in terms of accuracy, recall, precision, F1-score, ROC, AUC, and computational efficiency (Gupta et al., 2021). Oversampling techniques, including Random Oversampling and SVM-SMOTE, were applied selectively at Layers 2 and 3 to enrich the minority class without overly distorting the overall data distribution.

A key contribution of CSE-IDS lies in its cost-sensitive learning approach, which explicitly penalises misclassifications of rare attack types. The staged architecture progressively refines classification outcomes, reducing false positives by re-examining samples at deeper layers. This structural design contrasts with LIO-IDS, which focuses on efficient classifier utilisation through I-OVO reduction and temporal learning in its first stage.

As with LIO-IDS, the three datasets provide complementary strengths: NSL-KDD ensures a controlled benchmark, CIDDS-001 introduces enterprise-scale realism, and CICIDS2017 brings contemporary attack diversity. The consistent use of these datasets across both studies enables direct comparison of detection capability, computational efficiency, and the impact of imbalance handling.

2.5.3 Synthesis and Comparative Perspective

Both LIO-IDS and CSE-IDS address the critical challenge of detecting minority attack classes within imbalanced datasets while maintaining efficiency suitable for near real-time application. LIO-IDS demonstrates that LSTM-based temporal modelling, combined with an I-OVO structure, achieves strong detection rates with low latency across NSL-KDD, CIDDS-001, and CICIDS2017 (Gupta et al., 2021). Conversely, CSE-IDS shows that cost-sensitive training, when integrated with staged ensemble learning and targeted oversampling, enhances minority-class recognition and minimises false alarms under the same benchmarking conditions (Gupta et al., 2021).

Collectively, these works underscore two complementary strategies for modern NIDS design which are embedding class-aware cost functions early in the learning process to bias detection toward rare yet impactful intrusions, and structuring the decision pipeline to minimise redundant multi-class comparisons without compromising classification granularity. Cross-dataset evaluation remains critical, as each dataset highlights different facets of the intrusion detection problem such as legacy noise, enterprise traffic realism, and modern attack complexity.

2.6 Summary

This study explores the integration of three advanced cryptographic techniques: signcryption, proxy re-encryption (PRE), and transformer neural networks (TNN) to enhance security and efficiency in data transmission. Signcryption combines encryption and digital signatures in a single operation, reducing computational overhead while ensuring confidentiality and authenticity, making it suitable for resource-constrained environments (Kanchan, Singh and Chaudhari, 2019). PRE allows a semi-trusted proxy to re-encrypt data for different recipients without decrypting it, which is particularly useful for secure data sharing in dynamic systems such as cloud computing and IoT (Ateniese *et al.*, 2005). TNN utilises machine learning models to parallel processing to identify the category of event in real time based on contextual threats, enhancing adaptive security (Heaton, 2018).

These techniques address the growing complexity of cybersecurity threats and the need for adaptive security strategies. While traditional encryption methods face scalability and efficiency challenges, the integration of signcryption, PRE, and TNN provides a solution that enhances security, efficiency, and scalability in modern, decentralised systems. This framework is crucial for mitigating emerging threats and ensuring secure communication in the evolving landscape of digital security.

CHAPTER 3

METHODOLOGY AND WORK PLAN

3.1 Introduction to Proposed Solution

The increasing complexity of cybersecurity threats, particularly within decentralised systems such as Internet of Things (IoT) networks, has necessitated the development of more adaptive and intelligent security mechanisms. This study focuses on the design and evaluation of an adaptive cryptographic framework that integrates signcryption, proxy re-encryption (PRE), and transformer neural networks (TNN) to enhance data confidentiality, integrity, and authenticity in such environments.

By combining advanced cryptographic techniques with machine learning, the proposed framework aims to dynamically respond to evolving threats. Transformer Neural Networks will be utilised for real-time threat detection and contextual analysis, allowing the system to parallel processing for identify the category of event in response to observed network behaviour.

This research further examines the scalability of these technologies in decentralised settings and evaluates the framework's adaptability to emerging cybersecurity challenges. The goal is to develop a solution that not only meets the security demands of modern distributed systems but also ensures efficient and scalable performance in real-world applications.

3.2 Implementation of SignReencryption

In SignReencryption, the core objective of this project is to develop an adaptive cryptographic framework by integrating Intrusion Detection System (IDS) with a Transformer Neural Network (TNN). This deep learning architecture is known for its strength in processing sequential data.

3.2.1 Mechanism Overview: Intrusion Detection System with Transformer Neural Network

The system is trained using the CICIDS2017 dataset which is a widely accepted benchmark in the cybersecurity research community. This dataset includes a rich variety of network traffic scenarios, encompassing both normal operations and multiple forms of attack, making it a robust foundation for developing and evaluating detection systems.

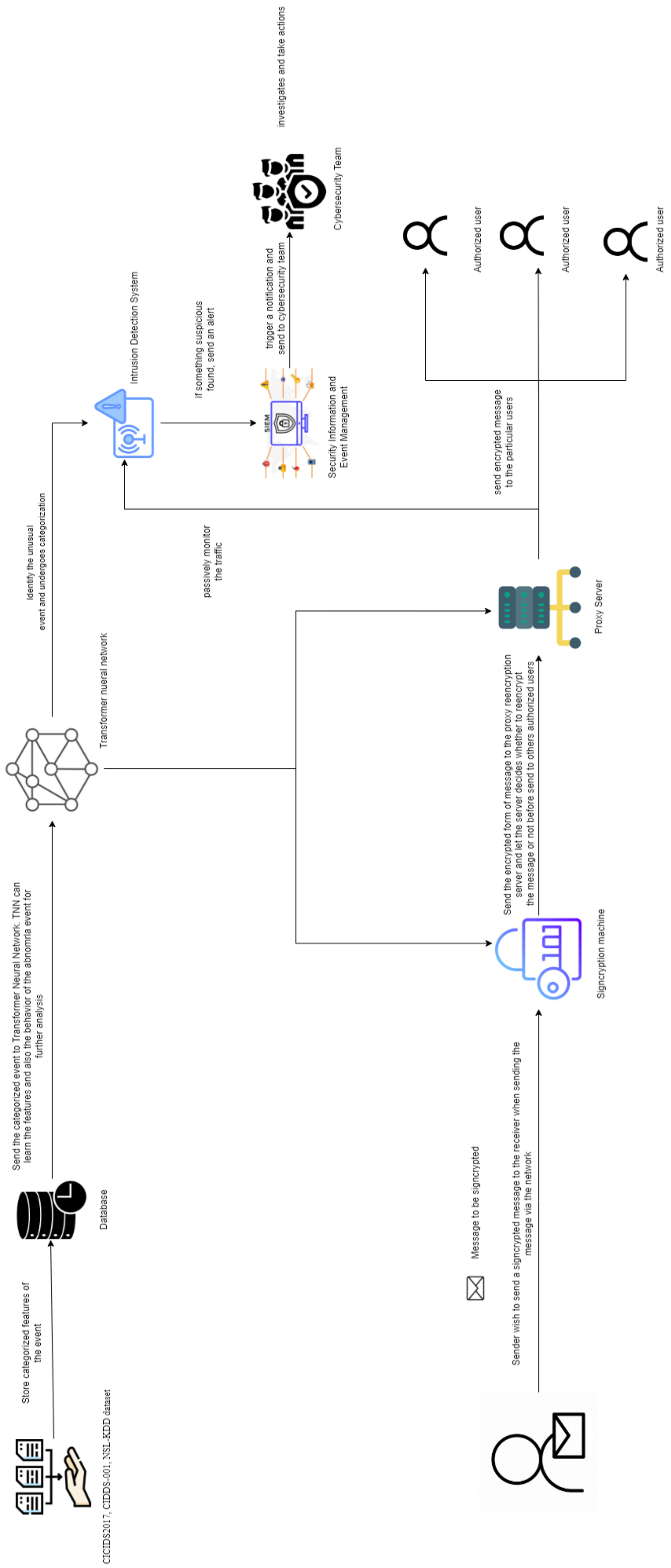

CICIDS2017, CIDDS-001, NSL-KDD dataset
Store categorized features of the event
Database
Send the categorized event to Transformer Neural Network. TNN can learn the features and also the behavior of the abnomria event for further analysis
Transformer nueral network
Identify the unusual event and undergoes categorization
Intrusion Detection System
if something suspicious found, send an alert
SIEM
Security Information and Event Management
trigger a notification and send to cybersecurity team
Cybersecurity Team
investigates and take actions
passively monitor the traffic
Proxy Server
send encrypted message to the particular users
Authorized user
Authorized user
Authorized user
Send the encrypted form of message to the proxy reencryption server and let the server decides whether to reencrypt the message or not before send to others authorized users
Signcryption machine
Message to be signcrypted
Sender wish to send a signcrypted message to the receiver when sending the message via the network

Figure 3.1: Intrusion Detection System integrated with TNN

In this design, the Transformer Neural Network functions as the central classification engine. Unlike conventional machine learning models, the TNN is particularly effective in capturing long-range dependencies within sequences of data, which is essential in network traffic analysis where malicious behaviour can manifest subtly over time. Additionally, the parallel processing nature of Transformers allows for more efficient training and inference, enabling the IDS to operate in near real-time, even in high-throughput environments.

The trained TNN is embedded directly within the IDS framework. Rather than simply flagging any anomaly, the system is designed to classify observed network events into three meaningful categories:

1. **Critical -** indicating severe threats that require immediate attention.
2. **Suspicious -** denoting potentially harmful activities that warrant closer inspection.
3. **Legitimate** - representing normal, benign network behaviour.

This classification not only aids in reducing false positives but also assists cybersecurity teams in prioritising their responses based on the severity of detected activities. Critical events typically point to immediate threats such as brute force, cross-site scripting or denial-of-service attempts. Suspicious events might indicate probing, unusual behaviour, or patterns that merit closer investigation. Legitimate events are recognised as normal traffic, contributing to more accurate baselining and less alert fatigue.

The IDS operates in a passive monitoring mode, continuously observing network traffic without altering it. This ensures that the system does not interfere with normal operations while still providing comprehensive visibility across the network. When the IDS guided by the TNN, identifies behaviour that deviates significantly from the learned norm or matches patterns associated with known threats, it generates a detailed alert. This alert is then forwarded to a centralised Security Information and Event Management (SIEM) system.

The SIEM acts as the central nervous system of the organisation’s cybersecurity operations. It aggregates and logs the alerts, providing a historical record of events for auditing and forensic purposes. Furthermore, the SIEM notifies the cybersecurity team through configured channels such as dashboards, emails, or integrated ticketing systems, prompting further analysis and response. The final layer of this mechanism involves human judgment. The cybersecurity team investigates the alerts, validates their accuracy and takes appropriate action, ranging from isolating devices or dropping the message to initiating deeper investigations.

By combining the pattern recognition power of Transformer models with the strategic role of IDS and SIEM, this project offers a comprehensive approach to modern cybersecurity defence which is proactive, intelligent, and adaptable to evolving threats.

3.2.2 Real-World Application and Data Flow

In the real-world deployment of the proposed solution, the system operates at two key levels: secure communication and threat detection, working together to safeguard sensitive information while ensuring network integrity.

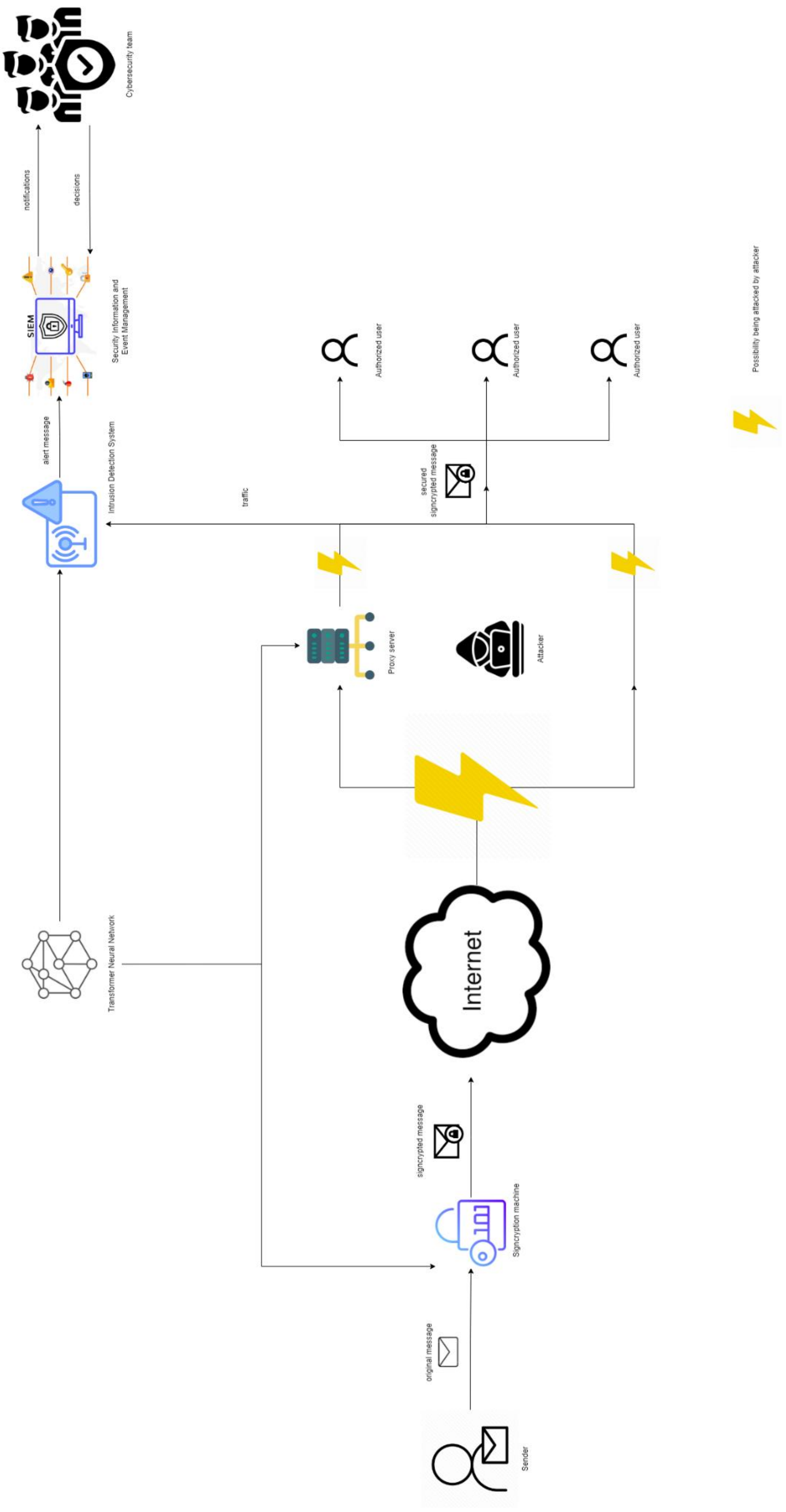


Figure 3.2: Normal message flow with the proposed solution

The communication flow is as follows:

1. The sender prepares a message and encrypts it using their own public key, ensuring its integrity and origin.
2. When the message needs to be shared securely with an authorised receiver, and must remain confidential during transmission, the system employs proxy re-encryption.
3. A re-encryption key is generated and used by a trusted proxy to convert the original ciphertext into a format that can be decrypted by the receiver without ever accessing the plaintext.
4. The re-encrypted message is then forwarded to the intended recipient.
5. The receiver decrypts the message using their private key, ensuring that only authorised individuals can access the original content.

When a user (the sender) wishes to transmit a message, the signcryption process is the first step before the message even leaves the internal network. This is facilitated by a signcryption machine, which performs both cryptographic signing and encryption in a single and efficient operation. This approach enhances both performance and security by addressing multiple objectives at once. The signcryption process ensures three essential elements of secure communication:

1. **Confidentiality** – The message is encrypted so that only the intended recipient, holding the correct private key, can decrypt and access its contents.
2. **Integrity** – Any alteration to the message in transit would invalidate the digital signature, making tampering detectable.
3. **Authentication and Non-repudiation** – The signature confirms the sender's identity, and because it is cryptographically tied to the sender's private key, the sender cannot later deny having sent the message.

By integrating these protections, signcryption reduces overhead and complexity compared to applying encryption and digital signatures separately, while still ensuring robust security suitable for hostile environments like the Internet.

Once the message is signcrypted, only the encrypted and signed version is sent across the public Internet. At no point is the original message exposed during transmission. This is particularly crucial because the Internet is assumed to be an untrusted and hostile environment which an attacker may always be present, constantly attempting to intercept or manipulate the data. However, since the message is never revealed in plaintext and has a valid cryptographic signature, attackers gain nothing useful from any interception attempts.

For added security and flexibility in communication, the system incorporates proxy re-encryption (PRE). If the signcrypted message is intended for multiple recipients, PRE allows a proxy to convert the ciphertext for the specific recipient without ever needing access to the original content or decryption keys. This means there is no need to download, decrypt, and re-encrypt the data for every new user. The original sender simply provides a re-encryption key, and the proxy handles secure redirection efficiently and without compromising confidentiality. As a result, sensitive data remains secure even when being routed through third parties or intermediaries.

Meanwhile, the IDS remains active in the background, continuously monitoring network traffic, including message transmissions. The IDS uses the trained TNN to detect anomalies or patterns of behaviour that might indicate malicious activity. If any traffic is flagged as critical or suspicious, an alert is immediately generated and forwarded to the SIEM system.

The SIEM component plays a central role in the organisation's incident response strategy. It logs all alerts for traceability and forensic purposes and sends real-time notifications to the cybersecurity team. Importantly, the system does not act independently to block or remove data but rather waits for expert decisions from cybersecurity team ensuring that human analysts validate potential threats and determine the appropriate course of action.

This integrated data flow from secure message transmission through proactive threat monitoring demonstrates a practical, end-to-end cybersecurity framework. The proposed solution not only improves the efficiency and security of internal and external communications but also reinforces organisational resilience through detection and human-in-the-loop decision making. It reflects a modern and scalable approach to cybersecurity that addresses both confidentiality and operational readiness in real-world environments.

3.3 Hardware and Software Requirements

This section outlines the hardware and software environments utilised in the development and implementation of the proposed system. A combination of local and cloud-based tools was employed to support tasks such as data preprocessing, neural network training, cryptographic simulation, and visualization.

3.3.1 Hardware Specifications

All experiments and development tasks were conducted using the following hardware setup:

Table 3.1: Hardware Specifications

Component	**Specification**
Processor	AMD Ryzen 7 7730U with Radeon Graphics, 2.00 GHz, 8 cores, 16 threads
RAM	24 GB DDR4
Storage	1 TB Micron 2400 NVMe SSD
GPU (for training)	NVIDIA Tesla T4 (via Google Colab)
Operating System	Windows 11

For training the Transformer Neural Network (TNN), Google Colab was utilised to leverage GPU acceleration, which significantly reduced the training time and improved performance.

3.3.2 Software Specifications

All experiments and development tasks were conducted using the following software setup:

Table 3.2: Software Specifications

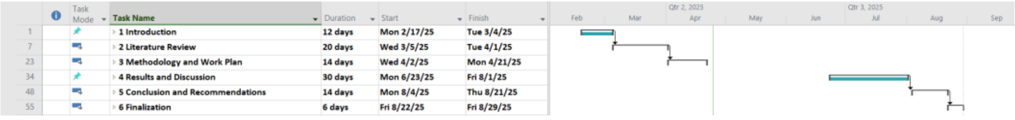
Software Tool	**Purpose**
Visual Studio Code	Used for code development and data preprocessing. In particular, it was used to convert .pcap files into .csv format for easier manipulation and input into machine learning models.
Google Colab	Used for training the Transformer Neural Network using TensorFlow/PyTorch with GPU support. Colab allowed access to scalable resources for iterative training and validation of models.
GitHub	Served as the primary source for pre-built TNN architectures and open-source reference implementations. Repositories were cloned and adapted for the project's specific use case in secure communication.
Draw.io	Used to visualise system architecture, workflows, and the overall integration between signcryption modules and TNN-based anomaly detection. These diagrams enhance understanding and presentation of the methodology.

This software-hardware ecosystem ensured a robust, reproducible, and efficient environment for implementing and validating the proposed hybrid cryptographic framework.

3.4 Workplan

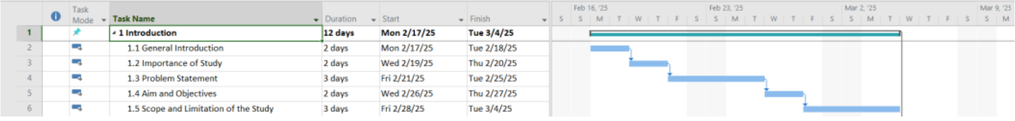
The project's timeline, major tasks, and key deliverables are laid out in a detailed Work Breakdown Structure (WBS). This structured plan outlines each phase of the project along with estimated durations, helping to ensure that progress stays on track and deadlines are met. The WBS provides a clear view of how the work is organised and when each component is expected to be completed.

Work Breakdown Structure (WBS)



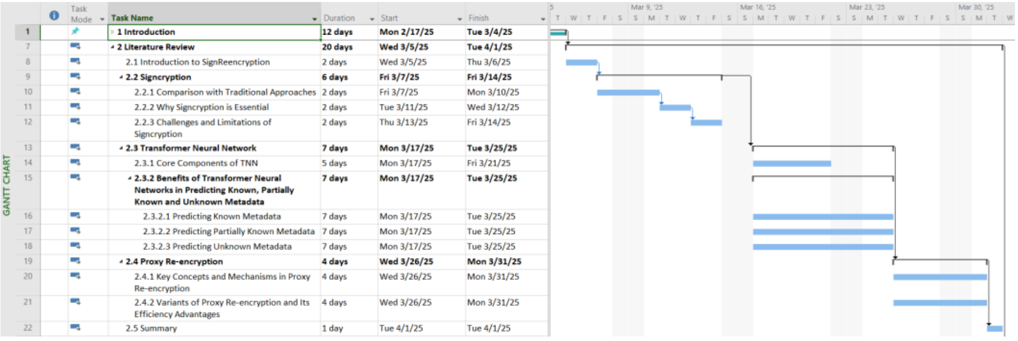
	Task Mode	Task Name	Duration	Start	Finish
1		1 Introduction	12 days	Mon 2/17/25	Tue 3/4/25
7		2 Literature Review	20 days	Wed 3/5/25	Tue 4/1/25
23		3 Methodology and Work Plan	14 days	Wed 4/2/25	Mon 4/21/25
34		4 Results and Discussion	30 days	Mon 6/23/25	Fri 8/1/25
48		5 Conclusion and Recommendations	14 days	Mon 8/4/25	Thu 8/21/25
55		6 Finalization	6 days	Fri 8/22/25	Fri 8/29/25

Figure 3.3: Overview of Tasks



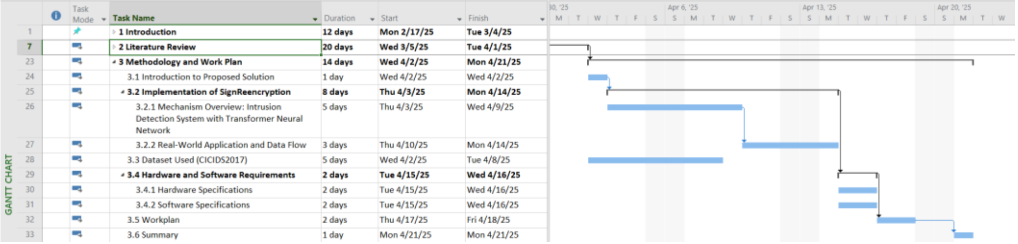
	Task Mode	Task Name	Duration	Start	Finish
1		1 Introduction	12 days	Mon 2/17/25	Tue 3/4/25
2		1.1 General Introduction	2 days	Mon 2/17/25	Tue 2/18/25
3		1.2 Importance of Study	2 days	Wed 2/19/25	Thu 2/20/25
4		1.3 Problem Statement	3 days	Fri 2/21/25	Tue 2/25/25
5		1.4 Aim and Objectives	2 days	Wed 2/26/25	Thu 2/27/25
6		1.5 Scope and Limitation of the Study	3 days	Fri 2/28/25	Tue 3/4/25

Figure 3.4: Task 1: Introduction (2/17/2025 – 3/4/2025)



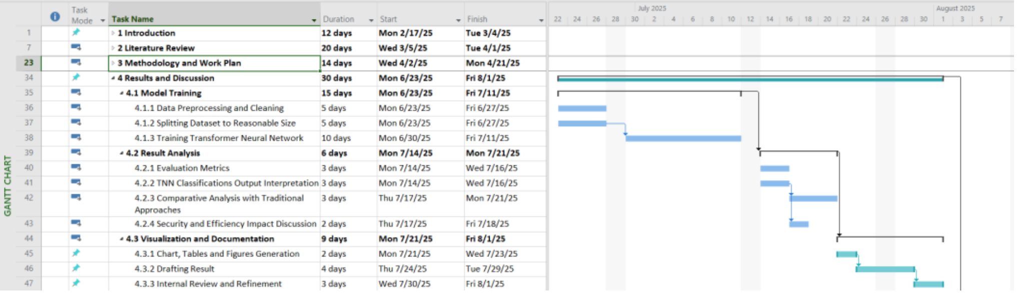
	Task Mode	Task Name	Duration	Start	Finish
1		1 Introduction	12 days	Mon 2/17/25	Tue 3/4/25
7		2 Literature Review	20 days	Wed 3/5/25	Tue 4/1/25
8		2.1 Introduction to SignReencryption	2 days	Wed 3/5/25	Thu 3/6/25
9		2.2 Signcryption	6 days	Fri 3/7/25	Fri 3/14/25
10		2.2.1 Comparison with Traditional Approaches	2 days	Fri 3/7/25	Mon 3/10/25
11		2.2.2 Why Signcryption is Essential	2 days	Tue 3/11/25	Wed 3/12/25
12		2.2.3 Challenges and Limitations of Signcryption	2 days	Thu 3/13/25	Fri 3/14/25
13		2.3 Transformer Neural Network	7 days	Mon 3/17/25	Tue 3/25/25
14		2.3.1 Core Components of TNN	5 days	Mon 3/17/25	Fri 3/21/25
15		2.3.2 Benefits of Transformer Neural Networks in Predicting Known, Partially Known and Unknown Metadata	7 days	Mon 3/17/25	Tue 3/25/25
16		2.3.2.1 Predicting Known Metadata	7 days	Mon 3/17/25	Tue 3/25/25
17		2.3.2.2 Predicting Partially Known Metadata	7 days	Mon 3/17/25	Tue 3/25/25
18		2.3.2.3 Predicting Unknown Metadata	7 days	Mon 3/17/25	Tue 3/25/25
19		2.4 Proxy Re-encryption	4 days	Wed 3/26/25	Mon 3/31/25
20		2.4.1 Key Concepts and Mechanisms in Proxy Re-encryption	4 days	Wed 3/26/25	Mon 3/31/25
21		2.4.2 Variants of Proxy Re-encryption and Its Efficiency Advantages	4 days	Wed 3/26/25	Mon 3/31/25
22		2.5 Summary	1 day	Tue 4/1/25	Tue 4/1/25

Figure 3.5: Task 2: Literature Review (3/5/2025 – 4/1/2025)



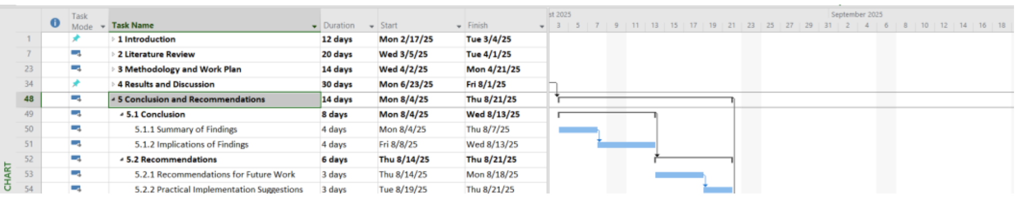
	Task Mode	Task Name	Duration	Start	Finish
1		1 Introduction	12 days	Mon 2/17/25	Tue 3/4/25
7		2 Literature Review	20 days	Wed 3/5/25	Tue 4/1/25
23		3 Methodology and Work Plan	14 days	Wed 4/2/25	Mon 4/21/25
24		3.1 Introduction to Proposed Solution	1 day	Wed 4/2/25	Wed 4/2/25
25		3.2 Implementation of SignReencryption	8 days	Thu 4/3/25	Mon 4/14/25
26		3.2.1 Mechanism Overview: Intrusion Detection System with Transformer Neural Network	5 days	Thu 4/3/25	Wed 4/9/25
27		3.2.2 Real-World Application and Data Flow	3 days	Thu 4/10/25	Mon 4/14/25
28		3.3 Dataset Used (CICIDS2017)	5 days	Wed 4/2/25	Tue 4/8/25
29		3.4 Hardware and Software Requirements	2 days	Tue 4/15/25	Wed 4/16/25
30		3.4.1 Hardware Specifications	2 days	Tue 4/15/25	Wed 4/16/25
31		3.4.2 Software Specifications	2 days	Tue 4/15/25	Wed 4/16/25
32		3.5 Workplan	2 days	Thu 4/17/25	Fri 4/18/25
33		3.6 Summary	1 day	Mon 4/21/25	Mon 4/21/25

Figure 3.6: Task 3: Methodology and Work Plan (4/2/2025 – 4/21/2025)



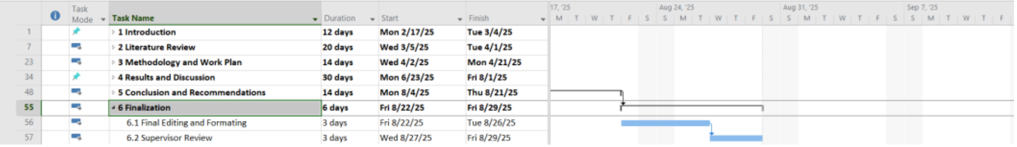
	Task Mode	Task Name	Duration	Start	Finish
1		1 Introduction	12 days	Mon 2/17/25	Tue 3/4/25
7		2 Literature Review	20 days	Wed 3/5/25	Tue 4/1/25
23		3 Methodology and Work Plan	14 days	Wed 4/2/25	Mon 4/21/25
34		4 Results and Discussion	30 days	Mon 6/23/25	Fri 8/1/25
35		4.1 Model Training	15 days	Mon 6/23/25	Fri 7/11/25
36		4.1.1 Data Preprocessing and Cleaning	5 days	Mon 6/23/25	Fri 6/27/25
37		4.1.2 Splitting Dataset to Reasonable Size	5 days	Mon 6/23/25	Fri 6/27/25
38		4.1.3 Training Transformer Neural Network	10 days	Mon 6/30/25	Fri 7/11/25
39		4.2 Result Analysis	6 days	Mon 7/14/25	Mon 7/21/25
40		4.2.1 Evaluation Metrics	3 days	Mon 7/14/25	Wed 7/16/25
41		4.2.2 TNN Classifications Output Interpretation	3 days	Mon 7/14/25	Wed 7/16/25
42		4.2.3 Comparative Analysis with Traditional Approaches	3 days	Thu 7/17/25	Mon 7/21/25
43		4.2.4 Security and Efficiency Impact Discussion	2 days	Thu 7/17/25	Fri 7/18/25
44		4.3 Visualization and Documentation	9 days	Mon 7/21/25	Fri 8/1/25
45		4.3.1 Chart, Tables and Figures Generation	2 days	Mon 7/21/25	Wed 7/23/25
46		4.3.2 Drafting Result	4 days	Thu 7/24/25	Tue 7/29/25
47		4.3.3 Internal Review and Refinement	3 days	Wed 7/30/25	Fri 8/1/25

Figure 3.7: Task 4: Results and Discussion (6/23/2025 – 8/1/2025)



	Task Mode	Task Name	Duration	Start	Finish
1		1 Introduction	12 days	Mon 2/17/25	Tue 3/4/25
7		2 Literature Review	20 days	Wed 3/5/25	Tue 4/1/25
23		3 Methodology and Work Plan	14 days	Wed 4/2/25	Mon 4/21/25
34		4 Results and Discussion	30 days	Mon 6/23/25	Fri 8/1/25
48		5 Conclusion and Recommendations	14 days	Mon 8/4/25	Thu 8/21/25
49		5.1 Conclusion	8 days	Mon 8/4/25	Wed 8/13/25
50		5.1.1 Summary of Findings	4 days	Mon 8/4/25	Thu 8/7/25
51		5.1.2 Implications of Findings	4 days	Fri 8/8/25	Wed 8/13/25
52		5.2 Recommendations	6 days	Thu 8/14/25	Thu 8/21/25
53		5.2.1 Recommendations for Future Work	3 days	Thu 8/14/25	Mon 8/18/25
54		5.2.2 Practical Implementation Suggestions	3 days	Tue 8/19/25	Thu 8/21/25

Figure 3.8: Task 5: Conclusion and Recommendations (8/4/2025 – 8/21/2025)



	Task Mode	Task Name	Duration	Start	Finish
1		1 Introduction	12 days	Mon 2/17/25	Tue 3/4/25
7		2 Literature Review	20 days	Wed 3/5/25	Tue 4/1/25
23		3 Methodology and Work Plan	14 days	Wed 4/2/25	Mon 4/21/25
34		4 Results and Discussion	30 days	Mon 6/23/25	Fri 8/1/25
48		5 Conclusion and Recommendations	14 days	Mon 8/4/25	Thu 8/21/25
55		6 Finalization	6 days	Fri 8/22/25	Fri 8/29/25
56		6.1 Final Editing and Formating	3 days	Fri 8/22/25	Tue 8/26/25
57		6.2 Supervisor Review	3 days	Wed 8/27/25	Fri 8/29/25

Figure 3.9: Task 6: Finalisation (8/22/2025 – 8/29/2025)

3.5 Summary

The growing complexity and sophistication of cybersecurity threats especially in decentralised environments like Internet of Things (IoT) networks have highlighted the need for more adaptable security frameworks. Traditional static approaches are increasingly insufficient in addressing dynamic attack patterns and evolving threat landscapes. In response to this challenge, this research proposes an integrated cryptographic and machine learning-based framework that combines signcryption, proxy re-encryption and Transformer Neural Networks. The goal is to strengthen data confidentiality, integrity, authenticity and non-repudiation while enabling real-time threat detection and adaptive response mechanisms suited for complex and distributed systems.

The chapter begins by outlining the conceptual foundation of the system, which brings together robust data encryption and advanced threat detection. Signcryption is utilised to streamline the encryption and signing process into a single step, ensuring key security features such as confidentiality, integrity, and authentication are maintained before any data leaves the internal network. To further enhance secure communication, proxy re-encryption is employed. This allows encrypted data to be securely shared with authorised recipients without the need to decrypt and re-encrypt the message at each stage, thereby preserving privacy and improving efficiency.

On the monitoring side, the IDS is embedded with a Transformer Neural Network to classify network activity into categories such as critical, suspicious, or legitimate. The mechanism capitalises on the TNN's ability to capture long-range dependencies in data sequences, making it particularly effective in detecting both known and previously unseen patterns of network behaviour. When suspicious activity is identified, the IDS generates an alert, which is forwarded to the Security Information and Event Management (SIEM) system for logging and escalation to the cybersecurity team. A practical data flow model is included to demonstrate how the components interact from message transmission and signcryption, to threat detection and incident response.

Finally, this methodology is supported by clearly defined hardware and software requirements. These technical specifications ensure the necessary infrastructure is in place to support both cryptographic processing and neural network training. Collectively, this approach lays out a structured and goal-oriented plan for achieving an adaptive cryptography system.

CHAPTER 4

DATASET AND EXPERIMENTAL SETUP

4.1 Dataset Used

This study employs three publicly available network intrusion detection datasets: **CICIDS2017**, **CIDDS-001**, and **NSL-KDD**. These datasets were selected to facilitate a comprehensive evaluation of the TabTransformer model across heterogeneous network environments and a wide spectrum of intrusion types. By incorporating both contemporary and well-established datasets, the assessment encompasses modern cyberattack patterns as well as classical intrusion scenarios, thereby providing a rigorous measure of the model's generalisability and detection capability.

4.1.1 CICIDS2017

The CICIDS2017 dataset was developed by the Canadian Institute for Cybersecurity and it is recognised as one of the most representative benchmarks for intrusion detection research. It contains network traffic data captured over a five-day period, combining benign activities with a variety of modern attack vectors, including Denial of Service (DoS), brute force, infiltration, botnet, and web-based attacks. The dataset is provided in comma-separated values (CSV) format with each record representing a network flow described by more than 80 features. These features include flow-level statistics such as duration and packet size distribution as well as protocol-specific attributes obtained through deep packet inspection. The diversity of attack types and the realistic nature of the traffic make CICIDS2017 highly suitable for evaluating a model's ability to detect complex and evolving threats in real-world network environments.

The dataset is organised into multiple files corresponding to specific attack scenarios and benign activity. Table 4.1 summarises the file composition, associated attack types, and the source and victim IP addresses involved in each capture.

Table 4.1: Description of CICISD2017 dataset

File Name	Description	Attacker IP	Victim Local IP
Monday-WorkingHours.pcap_ISCX.csv	Benign	-	-
Tuesday-WorkingHours.pcap_ISCX.csv	Benign, SSH-Patator, FTP-Patator	Kali (205.174.165.73)	WebServer Ubuntu (192.168.10.50)
Wednesday-WorkingHours.pcap_ISCX.csv	Benign, DoS Slowhttptest, DoS slowloris, DoS Hulk, DoS GoldenEye, Heartbleed	Kali (205.174.165.73)	WebServer Ubuntu (192.168.10.50), Ubuntu12 (192.168.10.51)
Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv	Benign, Web Attack (Brute Force, Sql Injection & XSS)	Kali (205.174.165.73)	WebServer Ubuntu (192.168.10.50)
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv	Benign, Infiltration	Kali (205.174.165.73)	Windows Vista (192.168.10.8), MAC (192.168.10.25)
Friday-WorkingHours-Morning.pcap_ISCX.csv	Benign, Bot	Kali (205.174.165.73)	Win 10 (192.168.10.15), Win 7 (192.168.10.9), Win 10 (192.168.10.14), Win 8 (192.168.10.5), Vista (192.168.10.8)
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	Benign, Port Scan	Kali (205.174.165.73)	Ubuntu16 (192.168.10.50)
Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv	Benign, DDoS	Three Win 8.1 (205.174.165.69 - 71)	Ubuntu16 (192.168.10.50)

In addition to file-level organisation, the dataset includes well-defined attack categories. **Table 4.2** presents these categories, along with concise descriptions and their significance in intrusion detection research.

Table 4.2: CICIDS2017 Attack Categories, Descriptions, and Detection Significance

Attack Category	Description	Significance for Detection
Normal Traffic	Legitimate user-generated network traffic representing baseline behaviour.	Establishes normal patterns of communication, reducing false positives in anomaly-based detection.
Denial of Service (DoS)	High-volume traffic intended to overwhelm services and disrupt availability.	Ensures effective detection of service disruption attempts that threaten system uptime.
PortScan	Systematic probing of network ports to identify open or vulnerable services.	Facilitates early identification of reconnaissance activity and potential exploitation attempts.
Patator	Automated brute-force attempts to guess authentication credentials.	Detects unauthorised access attempts and mitigates brute-force attack threats.
Web Attacks	Exploitation of web application vulnerabilities, including SQL injection and cross-site scripting (XSS).	Protects web interfaces from injection-based and script-based compromises.
Botnet	Compromised devices under remote control, used to execute coordinated malicious activities.	Detects large-scale, distributed threats originating from infected hosts.
Infiltration	Covert access to internal systems through compromised hosts or malicious payloads.	Identifies stealthy, high-risk breaches such as advanced persistent threats (APTs).

4.1.2 CIDDS-001

The CIDDS-001 dataset (Coburg Intrusion Detection Data Set) was generated in a simulated corporate network environment at the Coburg University of Applied Sciences. It comprises traffic from both an external server and an OpenStack-based cloud infrastructure, encompassing benign activity as well as labelled intrusion attempts such as Denial of Service (DoS), port scanning, brute force, and ping scans. The dataset is presented in a flow-based format, derived from packet capture (pcap) data, with each record summarising the communication between two IP addresses. The recorded attributes include source and destination IP addresses, ports, protocol type, the number of bytes transferred, and connection duration. The controlled simulation environment and precise attack labelling make CIDDS-001 a valuable benchmark for assessing a model's adaptability to diverse traffic patterns and intrusion behaviours.

The attack categories present in CIDDS-001, along with their descriptions and relevance for detection, are summarised in Table 4.3.

Table 4.3: CIDDS-001 Attack Categories, Descriptions, and Detection Significance

Attack Category	Description	Significance for Detection
Normal Traffic	Legitimate communication within a simulated corporate network environment.	Establishes the baseline behaviour of the network, aiding in the reduction of false alarms in anomaly-based systems.
Denial of Service (DoS)	Traffic floods intended to overwhelm network services and render them unavailable.	Enables prompt detection of service disruption attempts to maintain operational continuity.
Port Scan	Sequential probing of network ports to identify open or vulnerable services.	Facilitates early identification of reconnaissance activities that may precede targeted attacks.
Brute Force	Repeated automated attempts to guess authentication credentials.	Detects unauthorised access attempts and mitigates credential-based attack vectors.
Ping Scan	Transmission of Internet Control Message Protocol (ICMP) echo requests to identify active hosts.	Supports network mapping detection, preventing adversaries from identifying potential targets.

4.1.3 NSL-KDD

The NSL-KDD dataset is an enhanced version of the KDD Cup 1999 benchmark, specifically developed to address the redundancy and class imbalance issues in its predecessor. It comprises network connection records, each represented by 41 distinct features organised into three principal categories. The first category, basic connection features, describes fundamental session attributes such as connection duration and protocol type. The second category, content-based features, captures semantic information extracted from the payload, including indicators of anomalous or suspicious activity such as unsuccessful login attempts. The third category, traffic-based features, summarises statistical properties of network flows over a defined temporal window, including metrics such as the number of connections targeting the same host.

Each connection is classified as either normal or belonging to one of four attack categories: Probe, Denial of Service (DoS), User to Root (U2R), and Remote to Local (R2L). Despite being older than CICIDS2017 and CIDDS-001, NSL-KDD remains a widely adopted benchmark in intrusion detection research. Its structured feature representation and extensive adoption in prior studies make it a valuable reference for comparative performance evaluation. The attack categories and their significance are outlined in Table 4.4.

Table 4.4: NSL-KDD Attack Categories, Descriptions, and Detection Significance

Attack Category	Description	Significance for Detection
Normal Traffic	Legitimate user or system-generated network activity without malicious intent.	Establishes a behavioural baseline for distinguishing abnormal connections.
Denial of Service (DoS)	High-volume requests or malicious commands designed to exhaust resources and disrupt services.	Detects large-scale service interruption attempts that threaten availability.
Probe	Network scanning and information-gathering activities aimed at identifying vulnerabilities.	Enables proactive detection of reconnaissance activities before exploitation.
User to Root (U2R)	Exploits that allow an attacker with local user privileges to gain root or administrative access.	Identifies privilege escalation attempts that could compromise entire systems.
Remote to Local (R2L)	Attempts by a remote attacker to gain local user access without prior authorisation.	Detects unauthorised login or system access originating from external sources.

4.2 Data Processing Pipeline of CICIDS2017

The CICIDS2017 dataset was processed through a systematic pipeline to ensure reliability, consistency, and suitability for model training (Sharafaldin, Lashkari and Ghorbani, 2018). The pipeline consisted of eight key stages, described below.

4.2.1 Data Ingestion and Initial Inspection

The dataset was loaded from a unified source compiled from the original daily traffic captures. Upon loading, the data structure was examined to verify correct feature definitions, appropriate data types, and plausible value ranges. The inspection phase also focused on detecting anomalies such as infinite values, missing observations, or inconsistencies in feature naming. Non-essential whitespace in column headers was removed to ensure uniformity. Statistical summaries and exploratory checks were conducted to confirm that the dataset retained its integrity prior to transformation.

4.2.2 Label Normalisation and Taxonomy Aggregation

The original labels in CICIDS2017 contain detailed attack subcategories, including variants of Denial of Service (DoS Hulk, DoS GoldenEye, Slowloris, SlowHTTPTest, and DDoS), reconnaissance activity (PortScan), brute force attempts (FTP-Patator, SSH-Patator), web application attacks (SQL injection, cross-site scripting, and brute force), botnet activity (Bot), infiltration attempts, and benign traffic.

For experimental consistency, these subcategories were consolidated into seven higher-level classes:

1. **Normal**
2. **DoS** (all variants including DDoS)
3. **PortScan**
4. **Patator** (FTP and SSH brute force)
5. **Web Attack** (SQL injection, XSS, brute force)
6. **Bot**
7. **Infiltration**

This aggregation ensures alignment between training and test sets, reduces label fragmentation, and supports consistent comparison with the CIDDS-001 and NSL-KDD datasets.

4.2.3 Correlation Analysis and Redundancy Reduction

To address potential redundancy in the feature set, a Pearson correlation analysis was performed on all predictor variables. Features exhibiting strong linear dependence in which the absolute correlation is greater than 0.70 compared with any other feature were removed. This threshold was selected as a balance between minimising multicollinearity and retaining predictive information, thereby improving model stability and training efficiency.

4.2.4 Data Cleaning, Imputation, and Outlier Handling

Infinite values were recoded as missing, and records containing missing values were removed prior to splitting the data into training and test partitions. Missing value imputation was then performed using the median value of each feature, calculated exclusively from the training set, with these values subsequently applied to the test set. This procedure prevented the introduction of information from the test set into the training process.

4.2.5 Stratified Quota-Based Sampling and Partitioning

The CICIDS2017 dataset exhibits severe class imbalance, with certain rare attacks such as Infiltration occurring only a few dozen times compared to millions of benign records. To mitigate this imbalance while retaining a realistic representation of network traffic, a stratified quota-based sampling procedure was employed. Table 4.5 summarises the training and testing sets size of CICIDS2017 dataset.

Table 4.5: Description of CICIDS2017 Data Size

Class	Training Size	Testing Size
Normal	72,000	55,000
DoS	41,799	30,800
PortScan	13,000	10,000
Patator	7,997	5,838
Web Attack	1,367	813
Bot	1,000	966
Infiltration	20	16

Sampling was performed **independently within each class**, ensuring that both majority and minority classes were proportionally represented in each partition. When the available number of samples for a class was less than the target quota, all available instances were included. This strategy substantially reduced inter-class disparities while avoiding artificial oversampling at this stage.

4.2.6 Feature Scaling and Final Feature Set

After sampling and imputation, all continuous features were standardised to have zero mean and unit variance based on the training set statistics. The same transformation was applied to the test set using these training-derived parameters. This ensured that all features contributed equally during optimisation and that no feature dominated due to differences in magnitude.

4.2.7 Class Rebalancing for Training using SMOTE

Residual imbalance within the training set was addressed through the Synthetic Minority Oversampling Technique (SMOTE). This method creates synthetic instances for underrepresented classes by interpolating between existing minority-class samples, thereby enhancing balance without simply duplicating records. SMOTE was applied only to the training set, while the test set remained unaltered to preserve an unbiased evaluation environment.

4.2.8 Reproducibility Controls and Data Leakage Mitigation

All operations involving randomisation such as sampling, shuffling, and oversampling were executed with a fixed random seed to ensure reproducibility. Data transformations, including imputation and scaling, were fitted solely on the training set and applied to the test set using the same learned parameters. Feature elimination through correlation analysis was performed prior to splitting the data, ensuring both partitions shared an identical feature set. These safeguards collectively eliminated the risk of data leakage and ensured a fair and replicable evaluation process.

4.3 Data Processing Pipeline of CIDDS-001

The CIDDS-001 dataset underwent a structured preprocessing pipeline to ensure compatibility with the TabTransformer framework and comparability with the other datasets in this study (Ring *et al.*, 2019). The pipeline comprised ten stages, as outlined below.

4.3.1 Data Ingestion and Schema Verification

The CIDDS-001 dataset was ingested from a consolidated Parquet file containing all flow records. An initial schema audit verified attribute types, value domains, and completeness. The dataset comprises a heterogeneous mix of numeric, categorical, and temporal fields. Incidental whitespace in column names was removed to ensure uniform referencing across subsequent transformations. This inspection also confirmed that several variables particularly the timestamp field required type-specific processing to ensure compatibility with the modelling framework.

4.3.2 Temporal Feature Extraction

The field “Date first seen” was parsed into a timezone-agnostic datetime format. From this timestamp, three derived features were computed: hour of day, day of week, and day of month. These temporal covariates capture cyclical and periodic patterns in traffic behaviour that may indicate specific intrusion activities, such as weekday reconnaissance or late-night brute-force attempts. The engineered temporal features were retained as numerical predictors, while the original datetime field was later excluded from modelling to avoid type heterogeneity and potential information leakage.

4.3.3 Encoding of Non-Numeric Attributes

All non-numeric predictors excluding the target variable were numerically encoded using integer factorisation. This transformation preserved the identity of each category while producing a fully numeric feature matrix suitable for correlation analysis, scaling, and TabTransformer ingestion. The target field was excluded from this encoding process and handled separately during class harmonisation.

4.3.4 Correlation Analysis and Dimensionality Pruning

To reduce redundancy and mitigate multicollinearity, a Pearson correlation matrix was computed across all predictor variables while excluding the target. The upper triangular matrix was examined, and any feature with a pairwise correlation coefficient $|r| > 0.70$ with another feature was flagged as redundant and removed. This pruning preserved the diversity of information while stabilising model training and improving computational efficiency.

4.3.5 Target Harmonisation and Class Normalisation

The target variable (attackType) exhibited minor inconsistencies, including extraneous whitespace, irregular capitalisation, and placeholder tokens (“---”). These were normalised by trimming whitespace, converting to lowercase, and mapping placeholder entries to the *normal* category. The resulting canonical label set comprised five classes: normal, dos, portscan, pingscan, and bruteforce consistent with the taxonomy reported in Table 4.3. This harmonisation ensured semantic consistency between training and testing stages and facilitated cross-dataset comparability.

4.3.6 Stratified Quota-Based Sampling and Partitioning

To construct balanced yet representative partitions, a stratified quota-based sampling strategy was applied. Sampling was stratified by class to ensure proportional representation, with a fixed random seed for reproducibility. When the available number of samples in a class was below the quota, sampling with replacement was used; otherwise, simple random sampling was applied. The test set was drawn exclusively from records not included in the training set, ensuring complete separation between partitions.

Table 4.6 summarises the target sample sizes for each class in both the training and testing sets after applying this procedure. The quotas were designed to reduce extreme class imbalance while retaining sufficient representation of minority classes for meaningful evaluation.

Table 4.6: Description of CICIDS-001 Data Size

Class	Training Samples	Testing Samples
Normal	53,000	15,000
DoS	36,000	6,604
PortScan	9,117	3,250
PingScan	500	765
BruteForce	1,055	803

4.3.7 Missing-Data Handling and Standardisation

Prior to scaling, all infinite values were recoded as missing. Column-wise median imputation was performed on the training data only, and the resulting imputation parameters were applied to the test set. This prevented the introduction of information leakage from the evaluation set into the training process. Following imputation, feature standardisation (zero mean, unit variance) was fitted on the training features and applied to the test features using identical scaling parameters. The original datetime field was excluded to maintain a homogeneous numeric feature space.

4.3.8 Class Rebalancing for Training using SMOTE

Residual class imbalance in the training data was addressed using the Synthetic Minority Over-sampling Technique (SMOTE). SMOTE generates synthetic minority-class samples in the feature space by interpolating between nearest-neighbour instances. This enhanced the model's exposure to minority patterns without simply duplicating rare examples. The test set was not resampled, preserving its natural class distribution and ensuring the evaluation reflected realistic deployment conditions.

4.3.9 TabTransformer Input Configuration

All predictors were treated as continuous variables for the purposes of TabTransformer configuration. The model specification therefore consisted of the target variable (attackType), an empty list of categorical features, and a list of continuous features containing all other predictors. This ensured consistency with the CICIDS2017 processing pipeline, enabling uniform evaluation across datasets.

4.3.10 Reproducibility and Leakage Control

All stochastic operations including factorisation ordering, sampling, SMOTE synthesis, and record shuffling were executed with a fixed random seed to ensure reproducibility. All preprocessing transformations, including imputation and scaling, were fitted exclusively on the training set and then applied to the test set. This strict separation prevented any inadvertent transfer of distributional information from the test partition into the training process.

4.4 Data Processing Pipeline of NSL-KDD

The NSL-KDD dataset (Mahbod Tavallaee *et al.*, 2009) underwent a structured preprocessing workflow designed to harmonise feature formats, mitigate redundancy, and address extreme class imbalance while maintaining alignment with the evaluation protocols used for CICIDS2017 and CIDDS-001. The pipeline comprised nine stages, as described below.

4.4.1 Data Ingestion and Structural Audit

The dataset was ingested from a consolidated source and subjected to an initial structural audit to confirm feature types, assess completeness, and identify anomalies such as infinite values, placeholder entries, or inconsistent naming. This verification ensured that downstream transformations operated on a consistent and clean schema.

4.4.2 Categorical Encoding and Initial Cleaning

All non-numeric predictors excluding the target label were converted to integer codes through factorisation, producing a uniform numeric feature space suitable for correlation analysis and model ingestion. Infinite values were recoded as missing, and any rows with missing entries at this stage were removed to establish a clean base table for subsequent operations.

4.4.3 Correlation Screening and Redundancy Pruning

To reduce multicollinearity, a Pearson correlation matrix was computed on the predictor set while excluding the target. Variables with absolute correlation coefficients greater than 0.70 against any other feature were flagged as redundant and removed. The resulting reduced matrix retained the target label and preserved core predictive information while improving model stability.

4.4.4 Five-Class Taxonomy Mapping

The fine-grained NSL-KDD attack labels were consolidated into a standard five-class taxonomy of normal, dos, probe, r2l, and u2r. This mapping grouped similar attacks under broader categories such as neptune, smurf, and teardrop were assigned to dos, while portsweep and nmap were mapped to probe. The harmonised taxonomy supports consistent multi-class evaluation and facilitates cross-dataset comparison.

4.4.5 Stratified Quota-Based Sampling and Partitioning

Balanced yet representative training and test sets were constructed using a stratified quota-based sampling approach. Class-specific targets are shown in Table 4.7. Sampling was performed independently for each class with a fixed random seed. Where the available number of samples for a class was below the target quota, sampling with replacement was applied; otherwise, simple random sampling was used. The test set was drawn exclusively from records not included in training to guarantee complete separation.

Table 4.7: Description of NSL-KDD Data Size

Class	Training Samples	Testing Samples
Normal	67,343	9,711
DoS	45,927	7,458
Probe	11,656	2,421
R2L	995	2,887
U2R	52	67

4.4.6 Test-Set Alignment with Training Features

To ensure schema consistency, the test set underwent the same categorical encoding and column pruning as the training set. Any features present in training but absent in the test set were added with default zero values, and columns were reordered to match the training schema precisely. This alignment ensured compatibility during inference.

4.4.7 Class Rebalancing for Training using SMOTE

Residual class imbalance in the training data was addressed using the Synthetic Minority Over-sampling Technique (SMOTE), applied after categorical encoding. Target labels were temporarily encoded for SMOTE and metric computation. The test set remained unaltered to preserve realistic deployment conditions.

4.4.8 TabTransformer Input Specification

All predictors were passed to TabTransformer as continuous variables, with no categorical feature list specified. The configuration therefore comprised a single target variable (attackType) and a continuous-column list containing all remaining features.

4.4.9 Reproducibility and Leakage Controls

All stochastic processes including sampling, SMOTE synthesis, and shuffling were controlled by a fixed random seed to ensure reproducibility. Feature selection, imputation, and scaling parameters were learned exclusively from the training partition and applied to the test partition without modification, preventing any information leakage between splits.

4.5 Unified Model Optimisation and Training Framework

A standardised optimisation and training framework was implemented across CICIDS2017, CIDDS-001, and NSL-KDD to ensure methodological consistency and enable valid cross-dataset comparisons. This framework combined systematic hyperparameter search with controlled training procedures, applied identically to all datasets following their respective preprocessing pipelines.

4.5.1 Hyperparameter Optimisation

Hyperparameter tuning was conducted using the Optuna optimisation framework, which employs a Bayesian search strategy to explore the parameter space efficiently. The optimisation objective was to maximise the macro-averaged F1-score on a dedicated validation subset extracted from the training data. This metric was chosen for its balanced weighting of class-level performance, ensuring that both majority and minority classes contribute equally to the optimisation outcome.

The search space was defined to include architectural, regularisation, and optimisation parameters directly influencing the performance of the TabTransformer model. Table 4.8 summarises the parameters, their respective ranges or discrete sets, and the rationale for their inclusion.

Table 4.8: Search space for TabTransformer hyperparameter tuning.

Parameter	**Search Space**	**Rationale**
Learning rate	Log-uniform: 1×10^{-4} to 1×10^{-2}	Controls convergence speed and stability of gradient updates.
Attention blocks	Integer range: [2, 6]	Determines model depth and capacity for feature interaction modelling.
Input embedding dimension	{16, 32, 64}	Sets the dimensionality of feature embeddings, balancing expressiveness and computational cost.
Attention dropout	Uniform: [0.0, 0.3]	Regularises the self-attention layers to reduce overfitting.
Feed-forward dropout	Uniform: [0.0, 0.3]	Regularises the feed-forward layers in transformer blocks.
Add-norm dropout	Uniform: [0.0, 0.3]	Applies dropout to residual connections, improving generalisation.
Transformer activation	{GEGLU, ReLU, LeakyReLU, SwiGLU}	Selects the non-linear activation function within transformer blocks.
Batch size	{512, 1024, 2048, 4096}	Balances gradient estimation stability with GPU memory efficiency

Each trial in the search process consisted of complete training for 30 epochs, without early stopping, to ensure comparability across parameter configurations. For each dataset, 30 independent trials were executed. The configuration that yielded the highest macro-averaged F1-score on the validation subset was retained for the final model training and evaluation.

This tuning procedure ensured that the TabTransformer architecture was systematically adapted to the characteristics of each dataset while maintaining a consistent optimisation methodology across the experiments.

4.5.2 Standardised Training Protocol

The TabTransformer was configured in multi-class classification mode with cross-entropy loss as the objective function. Model evaluation during optimisation was conducted using macro-F1 as the primary metric, with per-class precision, recall, and F1-scores retained for detailed analysis.

The following constraints were maintained:

- **Epoch count:** fixed at 30 for all datasets to eliminate training-duration bias.
- **Data splits:** the training set was exclusively used for model fitting; the test set was reserved for final evaluation only.
- **Batch size:** determined individually for each dataset through the optimisation process.

4.5.3 Class Rebalancing in the Training Partition

Across all datasets, residual class imbalance remaining after stratified quota-based sampling was addressed using the Synthetic Minority Over-sampling Technique (SMOTE). As described in Sections 4.2.7, 4.3.8, and 4.4.7, SMOTE was applied only to the training partition to generate synthetic minority-class instances in feature space, thereby improving the representation of decision boundaries. The test partitions were left unchanged to preserve their natural class distributions and maintain deployment-time realism.

4.5.4 Reproducibility and Leakage Prevention

Reproducibility protocols and leakage-control measures were applied consistently across all datasets, as detailed in Sections 4.2.8, 4.3.10, and 4.4.9. All stochastic processes, including sampling, SMOTE synthesis, hyperparameter search, and model weight initialisation, were executed with fixed random seeds. Preprocessing transformations such as imputation, scaling, and correlation-based pruning were fitted exclusively on the training data and subsequently applied to the test data without recalculation. Training and test indices were maintained as strictly separate sets throughout the entire pipeline to ensure complete isolation and prevent any leakage of information.

4.5.5 Deployment of Optimal Configurations

For each dataset, the optimal hyperparameter configuration identified during optimisation was used to retrain the TabTransformer on the full rebalanced training set. The resulting model was then evaluated once on the held-out test set. This protocol ensures that reported test metrics are representative of the best-performing configuration obtained without any exposure to the test data during optimisation.

4.6 Evaluation Metrics

The performance of the Transformer-based Intrusion Detection System (IDS) was evaluated using four standard classification metrics: accuracy, precision, recall, and F1-score. These metrics provide a comprehensive view of model behaviour, capturing both overall correctness and the balance between false alarms and missed detections.

1. **Accuracy**

   Accuracy measures the proportion of correctly classified instances among the total number of evaluated instances. Although informative, it can be misleading in highly imbalanced datasets, where correct prediction of majority-class instances dominates the metric.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2. **Precision**

   Precision quantifies the proportion of correctly predicted positive cases among all instances predicted as positive. In intrusion detection, high precision means the IDS raises fewer false alarms.

$$Precision = \frac{TP}{TP + FP}$$

3. **Recall**

   Recall, or sensitivity, measures the proportion of actual positive cases that are correctly identified. High recall ensures that the IDS detects the majority of malicious activities.

$$Recall = \frac{TP}{TP + FN}$$

4. **F1-Score**

   The F1-score is the harmonic mean of precision and recall, providing a single value that balances the trade-off between them. It is particularly useful when both false positives and false negatives carry significant consequences.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Prior to metric computation, a confidence threshold of 0.8 was applied to the classification output probabilities. Predictions with a maximum class probability equal to or greater than 0.8 were assigned to the corresponding class such as **Critical** and **Legitimate**. Predictions falling below the 0.8 threshold were designated as **Suspicious** and flagged for further human investigation by the cybersecurity team. This approach reflects operational best practice, where borderline cases are not automatically classified as benign to reduce the risk of undetected threats.

The selected metrics were computed per class and macro-averaged across all classes to account for class imbalance. Macro-averaging assigns equal weight to each class, ensuring that performance on rare attack types is not overshadowed by majority-class performance.

To establish the effectiveness of the proposed SignReencryption-based IDS, its results were systematically compared with those of widely adopted baseline classifiers, including Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), Random Forest (RF), and XGBoost. These models were selected as they represent both conventional deep learning methods and ensemble-based approaches commonly employed in intrusion detection research. In addition, the proposed method was directly compared with the reported results of two state-of-the-art intrusion detection systems, namely CSE-IDS(Gupta, Jindal and Bedi, 2022) and LIO-IDS(Gupta, Jindal and Bedi, 2021), as documented in their original studies. This dual comparative framework against both baseline classifiers and advanced IDS benchmarks provides a rigorous basis for assessing the relative strengths and limitations of the proposed approach.

4.7 Signcryption Scheme Experimental Setup

This study evaluates the computational performance and communication efficiency of a bilinear-pairing-based signcryption scheme in comparison with a conventional Sign-Then-Encrypt (STE) baseline. Both schemes were implemented in Python using the Charm-Crypto library for public-key operations and PyCryptodome for symmetric encryption. The bilinear group was instantiated using the SS512 Type-1 pairing curve, providing a balance between computational cost and security.

4.7.1 Cryptographic Framework

The signcryption scheme under evaluation was implemented within a hybrid cryptographic framework combining public-key and symmetric primitives. The public-key component employs bilinear pairing operations over cyclic groups $\mathbf{G_1}$**,** $\mathbf{G_2}$**,** and $\mathbf{G_T}$ with exponents in the finite field $\mathbb{Z}_\mathbf{p}$, instantiated using the symmetric pairing-friendly curve SS512 from the Charm-Crypto library. This setting enables efficient computation of the bilinear map $\mathbf{e: G_1 \times G_2 \rightarrow G_T}$ and provides the algebraic structure required for pairing-based signcryption.

The symmetric encryption component is based on the Advanced Encryption Standard (AES) with a 128-bit key, operating in Cipher Block Chaining (CBC) mode with PKCS#7 padding. This ensures confidentiality of the transmitted message while maintaining compatibility with variable-length plaintexts. Session keys for the symmetric cipher are derived from the bilinear pairing output by applying the SHA-256 cryptographic hash function, producing a fixed-length 128-bit key from the serialized shared secret.

The proposed signcryption algorithm integrates encryption and signature generation into a single atomic operation, thereby reducing computational overhead and ciphertext expansion compared to a baseline Sign-Then-Encrypt (STE) approach. In the STE baseline, digital signatures are generated using the sender's private key and then appended to the plaintext before applying symmetric encryption. This sequential design incurs additional processing and data size, whereas the integrated signcryption approach achieves equivalent security properties with improved efficiency.

4.7.2 System Setup and Key Generation

The system initialisation procedure defines the public parameters and cryptographic keys required for both signcryption and verification. In the setup phase, generators $\mathbf{g \in G_1}$ and $\mathbf{g_2 \in G_2}$ are selected at random, along with a master secret exponent $\boldsymbol{\gamma} \in \mathbb{Z}_\mathbf{p}$. The public key component $\mathbf{P_{pub}}$ is computed as $\mathbf{g_2}^{\gamma}$ and published as part of the system parameters.

Individual participants generate their own long-term key pairs through the key generation algorithm. Each user selects a random secret key $\mathbf{sk} \in \mathbb{Z}_\mathbf{p}$ and computes the corresponding public key $\mathbf{pk = g^{sk} \in G_1}$. These key pairs are used for both signature generation and verification within the signcryption process. The security of the system relies on the computational hardness of the Bilinear Diffie–Hellman problem in the selected pairing group.

4.7.3 Signcryption Process

In the proposed signcryption scheme, a random group element $\mathbf{k_1 \in G_1}$ is generated by the sender to establish a one-time session secret. The bilinear pairing $\mathbf{e(k_1,\ P_{pub})}$ produces a shared value in G_T, which is subsequently serialised and hashed with SHA-256 to yield a 128-bit AES key. This key is used to encrypt the plaintext message via AES-CBC with PKCS#7 padding, producing the ciphertext component of the signcryption output.

Unlike conventional methods where signing and encryption are distinct phases, the proposed scheme derives authenticity implicitly from the algebraic structure of the pairing and the sender’s private key usage during the session key generation. This integration reduces computational duplication and limits ciphertext expansion, a factor evaluated in Section 5.

4.7.4 Sign-Then-Encrypt Baseline

For comparative analysis, a baseline Sign-Then-Encrypt (STE) scheme was implemented. In this approach, the sender first generates a digital signature over the plaintext using their private key. The signature is appended to the plaintext and the concatenated data is then encrypted using AES-CBC with a freshly generated symmetric key. This sequential approach ensures confidentiality and authenticity but incurs additional computational and communication overhead compared to integrated signcryption.

4.7.5 Performance Measurement Protocol

The performance evaluation of the proposed signcryption scheme and the Sign-Then-Encrypt (STE) baseline was designed to reflect realistic operational conditions in Internet of Things (IoT) environments, with a focus on the transportation sector. Within this context, Intelligent Transportation Systems (ITS) are a specialised IoT application domain in which secure, real-time message exchange is critical for ensuring road safety, coordinating emergency responses, and optimising traffic flows. Security requirements in ITS are particularly stringent, as both confidentiality and authenticity must be guaranteed to prevent false alerts and unauthorised message injection.

To ensure a comprehensive assessment, two complementary testing procedures were employed. The first focused on repeated single-message testing, measuring algorithmic execution time and ciphertext size for an identical fixed-length message over multiple independent iterations. The second involved batch testing to evaluate sustained throughput and communication overhead under simulated high-volume ITS workloads. Together, these approaches provide both micro-level and macro-level insights into computational efficiency, scalability, and communication performance.

4.7.5.1 Test Message and Environment

The test message used in both schemes was a fixed vehicular incident alert representative of security-sensitive communications in Intelligent Transportation Systems (ITS):

"Accident on highway 46, multiple vehicle collision. Emergency services dispatched."

This message was selected to reflect realistic IoT traffic, where both confidentiality and authenticity are critical for public safety operations. Its fixed length and structure enabled controlled comparison between schemes by removing variability in payload composition.

All experiments were executed under identical hardware and software configurations to ensure fair comparison. The same cryptographic libraries, key sizes, and parameter sets were applied to both schemes, and no other processes were permitted to run concurrently during benchmarking to avoid performance interference.

4.7.5.2 Batch Testing Procedure

In addition to repeated single-message testing, a batch testing procedure was implemented to evaluate aggregate performance under simulated continuous workload conditions. This involved sequential processing of a predefined set of messages representative of varying ITS traffic scenarios.

The batch tests emulated sustained operational conditions in which cryptographic operations must be performed continuously, such as during large-scale incident reporting or multi-sensor data aggregation. Each message was processed without pause, enabling the measurement of sustained throughput and cumulative computational cost.

For both schemes, total processing time and average per-message execution time were recorded, along with ciphertext sizes for each message. From these results, the total communication overhead and percentage size savings achieved by signcryption relative to STE were calculated.

This dual-testing approach ensures that the evaluation captures both isolated algorithmic performance and scalability under realistic, high-throughput conditions, providing a robust basis for comparison in real-time, resource-constrained ITS environments.

4.7.6 Reproducibility Control

To ensure experimental reproducibility, all tests were executed under identical computational conditions, with fixed random seeds controlling key generation, random number sampling, and session key derivation. Both schemes were implemented using the same cryptographic libraries and parameter sets to eliminate variability from implementation differences. No result from the testing phase was used to influence the setup or configuration, ensuring that the evaluation remained unbiased and representative.

4.8 Summary

This chapter presents the datasets, preprocessing workflows, and experimental configurations used in this study to evaluate the TabTransformer model for intrusion detection, as well as to benchmark a proposed bilinear pairing-based signcryption scheme against a conventional Sign-Then-Encrypt (STE) baseline.

Three benchmark intrusion detection datasets were selected, namely CICIDS2017, CIDDS-001, and NSL-KDD, to ensure comprehensive coverage of diverse network attack categories. For each dataset, a structured data processing pipeline was implemented, including schema validation, feature engineering, categorical encoding, correlation-based feature pruning, class-quota sampling, imputation, scaling, and SMOTE-based rebalancing. These procedures ensured high-quality and consistent inputs across datasets while addressing class imbalance and multicollinearity. Sampling strategies were designed to preserve representative distributions while enabling fair cross-dataset evaluation of model performance.

The experimental setup for the signcryption evaluation involved implementing both the proposed scheme and the STE baseline using a unified cryptographic framework. Public-key operations employed the SS512 pairing-friendly curve from the Charm-Crypto library, while symmetric encryption used AES-128 in CBC mode with PKCS#7 padding. The performance measurement protocol incorporated two complementary testing methodologies: repeated single-message tests to measure algorithmic efficiency and ciphertext expansion in isolation, and batch processing tests to assess throughput and scalability under simulated Intelligent Transportation System (ITS) workloads. All experiments were executed under identical hardware and software configurations with strict reproducibility controls.

This integrated experimental design ensures that results reported in Chapter 5 are directly comparable across datasets, cryptographic schemes, and testing conditions, providing a robust empirical basis for assessing both the machine learning and cryptographic components of the research.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Overview of Optuna Results on Benchmark Datasets

This section reports the experimental outcomes of the proposed TabTransformer-based intrusion detection system across three benchmark datasets: CICIDS2017, CIDDS-001, and NSL-KDD. Hyperparameter tuning was conducted using the Optuna framework, with the macro-averaged F1 score on the validation set as the optimisation objective. This objective was chosen to balance detection across majority and minority classes, thereby addressing the class imbalance that characterises intrusion detection tasks.

The presentation of results is organised into five parts. First, the optimal hyperparameter configurations identified by Optuna are detailed for each dataset. Second, model performance is evaluated using precision, recall, and F1-score to provide a comprehensive assessment of detection capability. Third, computational efficiency is analysed in terms of training time and testing time, reflecting the cost and feasibility of deployment. Fourth, the cryptographic efficiency of the proposed SignReencryption scheme is evaluated against a conventional Sign-Then-Encrypt baseline, with results reported for ciphertext expansion and per-message execution time. Finally, the empirical findings are synthesised into a critical discussion of strengths and weaknesses in the context of operational deployment.

5.2 Optuna Results of Different Datasets under TabTransformer

Across the three datasets, the Optuna-based hyperparameter optimisation demonstrated the adaptability of the TabTransformer to different feature spaces and traffic distributions. For CICIDS2017, the optimal configuration required a deeper architecture with six attention blocks and larger embeddings, reflecting the dataset’s higher complexity and variety of attack categories. In contrast, CIDDS-001 achieved its best performance with a relatively shallow architecture of four attention blocks and compact embeddings, supported by a higher learning rate and stronger dropout regularisation. For NSL-KDD, the optimisation favoured an even smaller configuration with only two attention blocks, a particularly low learning rate, and the SwiGLU activation function, which proved more effective in capturing subtle feature interactions within the dataset's balanced yet limited feature set.

Overall, the results indicate that deeper and more expressive architectures are advantageous for large, heterogeneous datasets such as CICIDS2017, whereas leaner and more carefully regularised models are better suited to smaller or less complex datasets such as CIDDS-001 and NSL-KDD. These findings provide a consistent basis for the performance evaluations discussed in the following section.

5.2.1 Optuna Results of CICIDS2017

The CICIDS2017 dataset contains a diverse set of attack categories and normal traffic patterns generated over five consecutive days. In this study, a subset containing six major attack classes and one normal class was used, preserving the imbalanced distribution observed in real-world network traffic. The dataset's complexity and variety of attack types make it a suitable benchmark for evaluating the generalisation and robustness of intrusion detection models.

Hyperparameter tuning for this dataset was performed using Optuna's Bayesian optimisation framework. The search space covered both architectural and regularisation parameters, as well as learning rate, activation function, and batch size. The macro-averaged F1 score on the validation set was chosen as the objective function, ensuring that both majority and minority classes influenced the optimisation outcome.

Table 5.1: Optuna Hyperparameter Optimisation Results of CICIDS2017 dataset

Hyperparameter	CICIDS2017 Value
Best F1 Objective	0.63266
Learning rate	0.0022346

Attention blocks	6
Embedding dimension	64
Attention dropout	0.003887
FFN dropout	0.11353
Add-Norm dropout	0.29059
Activation function	ReLU
Batch size	512

The optimal configuration in Table 5.1 reflects the result of multiple trials aimed at balancing model expressiveness with generalisation capability. Six attention blocks, combined with a 64-dimensional embedding space, provided sufficient depth and representation power without introducing excessive complexity. The learning rate of 0.0022346 supported stable and gradual convergence during training, which is important for attention-based architectures.

The low attention dropout value (0.003887) indicates that retaining most of the attention connections improved feature interaction learning, while the relatively high add-norm dropout (0.29059) provided effective regularisation in residual pathways. The feed-forward dropout rate (0.11353) further contributed to overfitting prevention in the dense layers. The choice of ReLU as the activation function is consistent with the model's need to efficiently process structured, tabular data. A batch size of 512 allowed for stable gradient estimation while maintaining computational efficiency.

This configuration was fixed for the final training and evaluation on the CICIDS2017 dataset, serving as the baseline for subsequent performance comparisons.

5.2.2 Optuna Results of CIDDS-001

The CIDDS-001 dataset is a flow-based network intrusion detection dataset containing a mixture of simulated normal traffic and various attack scenarios. The dataset includes five primary classes: Normal, DoS, Port Scan, Ping Scan, and Brute Force. While the majority of traffic is normal or DoS, the Ping Scan and Brute Force classes are relatively rare, creating a notable class imbalance. This combination of traffic patterns and imbalance characteristics makes CIDDS-001 a relevant benchmark for assessing the ability of intrusion detection models to generalise across both frequent and infrequent attack categories.

Hyperparameter tuning for this dataset was carried out using Optuna's Bayesian optimisation framework. The search space included architectural parameters such as the number of attention blocks and embedding dimension, regularisation parameters including dropout rates, and training parameters such as learning rate, activation function, and batch size. The macro-averaged F1 score on the validation set was used as the optimisation objective to ensure that both majority and minority classes influenced the final configuration.

Table 5.2: Optuna Hyperparameter Optimisation Results of CIDDS-001 dataset

Hyperparameter	**Value**
Best F1 Objective	0.67109
Learning rate	0.00678379
Attention blocks	4
Embedding dimension	32
Attention dropout	0.1690768
FFN dropout	0.1108410
Add-Norm dropout	0.06774053
Activation function	ReLU
Batch size	4096

The configuration in Table 5.2 reflects the outcome of multiple trials designed to balance model complexity with the ability to generalise to unseen traffic patterns. The use of four attention blocks with a 32-dimensional embedding space suggests that a shallower architecture with more compact representations was sufficient for CIDDS-001, likely due to its smaller feature set compared to CICIDS2017. The learning rate of 0.00678379 is higher than that found optimal for CICIDS2017, indicating that the model could converge more quickly on this dataset without sacrificing stability.

A relatively high attention dropout value (0.1691) was selected, which can help prevent overfitting by encouraging the model to distribute attention across multiple features rather than relying on a few dominant ones. The feed-forward dropout rate (0.1108) and add-norm dropout (0.0677) provided moderate regularisation in their respective components, further supporting generalisation. The ReLU activation function was again preferred for its efficiency and stability when working with tabular flow-based features. The batch size of 4096, significantly larger than for CICIDS2017, takes advantage of the smaller feature space, enabling faster training while maintaining stable gradient estimates.

This configuration was fixed for all subsequent training and evaluation on the CIDDS-001 dataset, ensuring consistency in the reported results and enabling a fair comparison with other datasets and baseline models.

5.2.3 Optuna Results of NSL-KDD

The NSL-KDD dataset is a refined version of the original KDD'99 intrusion detection benchmark, created to remove redundant records and provide a more balanced and challenging evaluation environment. It contains four main attack categories: Denial of Service (DoS), Probe, Remote-to-Local (R2L), and User-to-Root (U2R), alongside normal traffic. Despite these improvements, the dataset retains significant class imbalance, particularly for the R2L and U2R categories, which occur much less frequently than the others. Its long-standing use in intrusion detection research and the diversity of attack types make NSL-KDD an important dataset for assessing model robustness and adaptability.

Hyperparameter tuning for NSL-KDD was conducted using Optuna's Bayesian optimisation framework. The search space included architectural parameters such as the number of attention blocks and embedding dimension, regularisation parameters including dropout rates, and training parameters such as learning rate, activation function, and batch size. The macro-averaged F1 score on the validation set was used as the optimisation objective, ensuring that both majority and minority classes contributed to the selection of the final configuration.

Table 5.3: Optuna Hyperparameter Optimisation Results of NSL-KDD dataset

Hyperparameter	**Value**
Best F1 Objective	0.88845
Learning rate	0.00051066
Attention blocks	2
Embedding dimension	64
Attention dropout	0.2411373
FFN dropout	0.1780757
Add-Norm dropout	0.1506164
Activation function	SwiGLU
Batch size	4096

The configuration in Table 5.3 reflects the outcome of multiple trials aimed at balancing learning stability with the model's ability to generalise across diverse attack types. The use of only **two attention blocks** suggests that a relatively shallow architecture was sufficient to capture the feature relationships present in NSL-KDD, which has fewer features and lower variance compared to modern datasets such as CICIDS2017. The **embedding dimension** of 64 provided adequate representational capacity without introducing excessive model complexity.

The learning rate of **0.00051066** is the lowest among the three datasets tested, indicating that slow and careful parameter updates were required for optimal convergence, likely due to the relatively small feature space and the need to fine-tune classification boundaries for the minority classes. The regularisation settings included a relatively high **attention dropout** (0.2411) and **feed-forward dropout** (0.1781), both aimed at mitigating overfitting, while the **add-norm dropout** (0.1506) provided additional stability in residual connections.

A notable difference from the other datasets was the selection of **SwiGLU** as the activation function, suggesting that its gated linear unit mechanism was more effective at modelling the subtle feature interactions in NSL-KDD compared to ReLU. The **batch size** of 4096, as with CIDDS-001, leveraged the smaller feature set to accelerate training while maintaining stable gradient estimates.

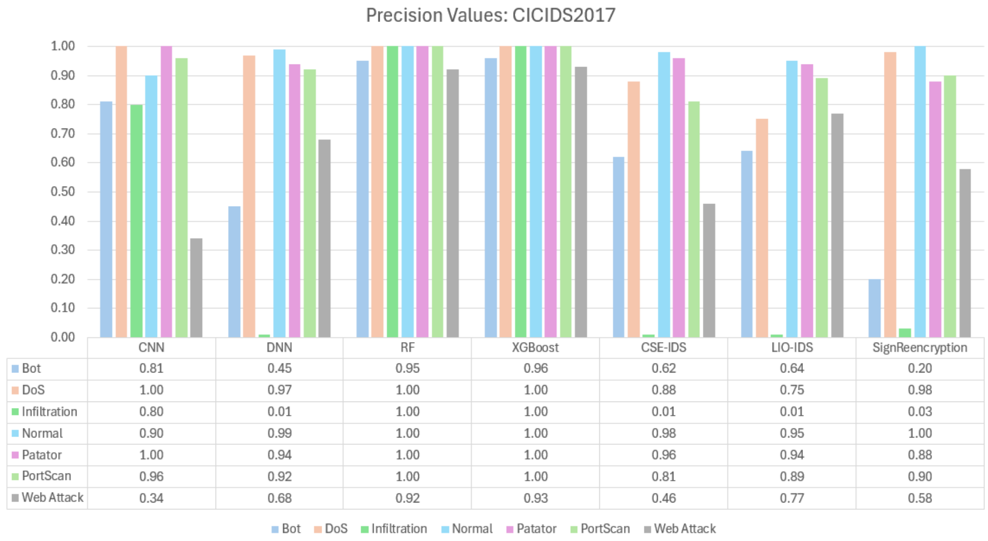
This configuration was fixed for all subsequent experiments on the NSL-KDD dataset, ensuring consistency in training, evaluation, and comparative analysis across the different experimental stages.

5.3 Evaluation Metrics of TabTransformer and Comparative Models

This section has examined the comparative performance of the proposed TabTransformer with SignReencryption against a range of baseline methods using precision, recall, and F1-score as evaluation metrics. Figures **5.1–5.9** illustrated the behaviour of each method across CICIDS2017, CIDDS-001, and NSL-KDD, highlighting consistent patterns. While ensemble methods dominated in precision under fully visible, balanced conditions, they collapsed in imbalanced datasets. Conventional deep learning baselines achieved high scores for the majority categories but performed poorly in minority classes, confirming their limited robustness. By contrast, SignReencryption demonstrated a recall-oriented profile, maintaining competitive precision and F1 for dominant classes while delivering decisive improvements for minority categories particularly in NSL-KDD, where historical weaknesses in R2L and U2R detection were largely addressed

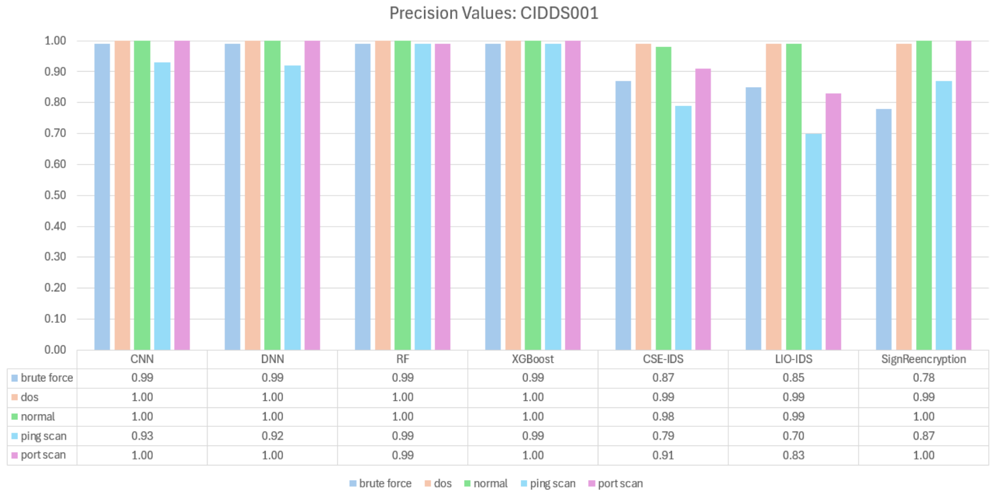
5.3.1 Precision across datasets

Precision reflects the proportion of alerts that correctly correspond to actual intrusions. It is a measure of reliability in prediction and an indicator of how much false-alarm noise is generated for security analysts. The comparative precision values of different models on CICIDS2017, CIDDS-001, and NSL-KDD are shown in Figures **5.1–5.3**, respectively. These figures highlight how models differ in their ability to balance reliability across majority and minority attack classes.



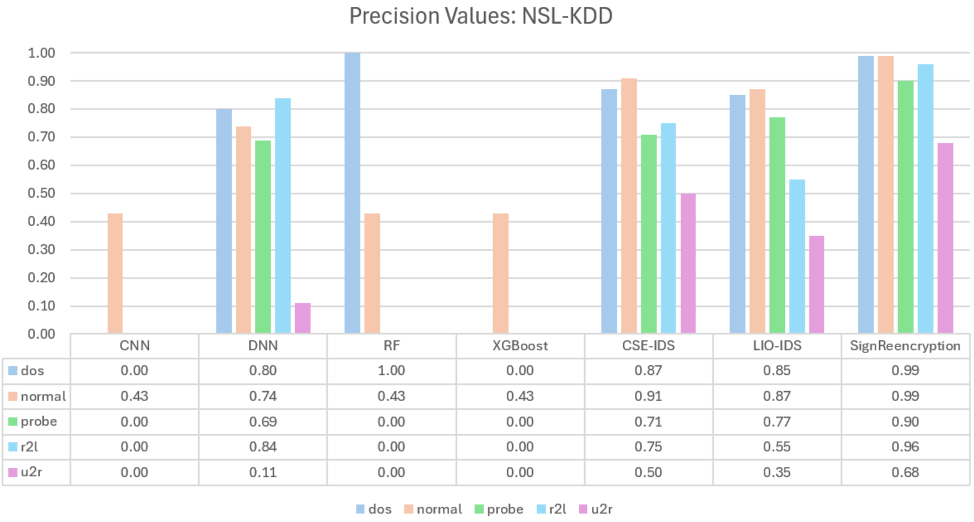
	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
Bot	0.81	0.45	0.95	0.96	0.62	0.64	0.20
DoS	1.00	0.97	1.00	1.00	0.88	0.75	0.98
Infiltration	0.80	0.01	1.00	1.00	0.01	0.01	0.03
Normal	0.90	0.99	1.00	1.00	0.98	0.95	1.00
Patator	1.00	0.94	1.00	1.00	0.96	0.94	0.88
PortScan	0.96	0.92	1.00	1.00	0.81	0.89	0.90
Web Attack	0.34	0.68	0.92	0.93	0.46	0.77	0.58

Figure 5.1: Precision values on the CICIDS2017 dataset



	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
brute force	0.99	0.99	0.99	0.99	0.87	0.85	0.78
dos	1.00	1.00	1.00	1.00	0.99	0.99	0.99
normal	1.00	1.00	1.00	1.00	0.98	0.99	1.00
ping scan	0.93	0.92	0.99	0.99	0.79	0.70	0.87
port scan	1.00	1.00	0.99	1.00	0.91	0.83	1.00

Figure 5.2: Precision values on the CIDDS001 dataset



	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
dos	0.00	0.80	1.00	0.00	0.87	0.85	0.99
normal	0.43	0.74	0.43	0.43	0.91	0.87	0.99
probe	0.00	0.69	0.00	0.00	0.71	0.77	0.90
r2l	0.00	0.84	0.00	0.00	0.75	0.55	0.96
u2r	0.00	0.11	0.00	0.00	0.50	0.35	0.68

Figure 5.3: Precision values on the NSL-KDD dataset

As illustrated in **Figure 5.1**, ensemble-based approaches such as Random Forest (RF) and XGBoost achieved the highest levels of precision in the CICIDS2017 dataset, with most classes exceeding 0.95, including the minority categories such as Web Attack and Infiltration. This outcome demonstrates the ability of ensemble learners to exploit the diverse feature space of CICIDS2017 when all attributes are fully visible. Conventional deep learning baselines such as CNN and DNN were less consistent. CNN achieved perfect precision for DoS and Patator (1.00) and strong results for PortScan (0.96), yet its precision dropped sharply for Web Attack (0.34), producing uneven results across classes. DNN showed a similar trend, with a dramatic collapse in Infiltration (0.01), which underscores its limitations in distinguishing rare classes. In comparison, the proposed SignReencryption maintained high precision for majority classes, including DoS (0.98), Normal (1.00), PortScan (0.90), and Patator (0.88). However, as shown in the figure, its precision for minority categories such as Bot (0.20) and Infiltration (0.03) was deliberately conservative, reflecting an optimisation strategy that emphasises recall on minority classes at the expense of increased false positives.

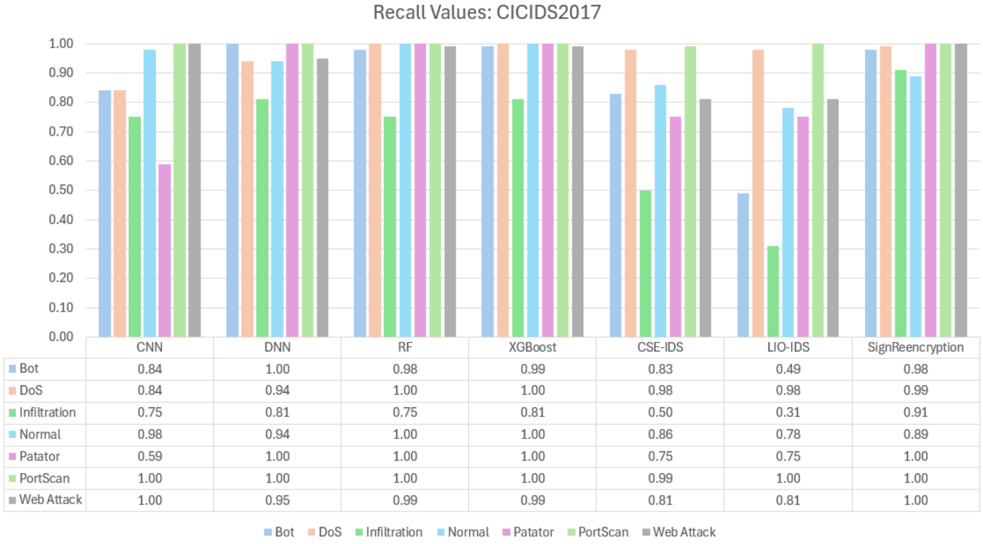
The results on CIDDS-001, depicted in **Figure 5.2**, reveal that this dataset's clearer class boundaries allowed all models to achieve very high precision, often approaching saturation. RF and XGBoost sustained values above 0.99 across all categories, while CNN and DNN also achieved stable results without catastrophic failure. Importantly, SignReencryption performed competitively, with precision values ranging between 0.83 and 1.00 across all classes. Unlike in CICIDS2017, no collapse was observed in minority categories, which suggests that the re-encryption stage of the proposed system does not inherently reduce discriminative precision when the class structure is more balanced.

The precision results for NSL-KDD are shown in **Figure 5.3**, where the dataset's imbalance presented the greatest challenge. Traditional baselines such as CNN, RF, and XGBoost recorded negligible precision for Probe, R2L, and U2R, effectively failing to identify these classes reliably. Even more advanced IDS baselines, CSE-IDS and LIO-IDS, exhibited inconsistent precision across minority classes. In contrast, SignReencryption preserved workable precision exactly where the other models struggled most. As indicated in the figure, it achieved 0.90 for Probe, 0.96 for R2L, and 0.68 for U2R, while retaining very high values for DoS and Normal (both above 0.98). These results confirm that the proposed method is sensitive to minority attacks and can maintain predictive reliability even under imbalanced conditions.

Overall, Figures **5.1–5.3** demonstrate that while ensemble learners dominate in balanced and fully observable scenarios, their precision advantage collapses in highly imbalanced settings. By contrast, SignReencryption sustains viable precision in minority classes under privacy-preserving constraints, offering a practical balance between sensitivity and reliability.

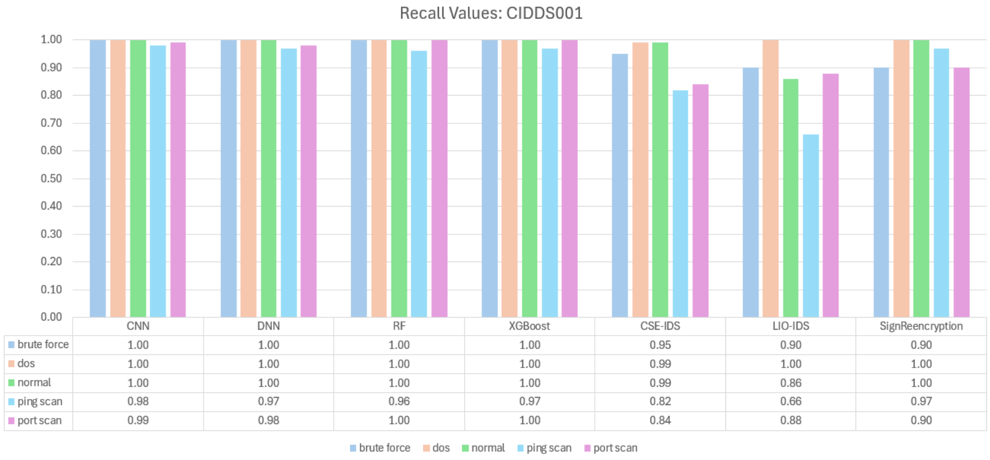
5.3.2 Recall across datasets

Recall measures the sensitivity of a detection system, quantifying its ability to identify all instances of intrusions. From an operational standpoint, recall is critical because missed detections correspond directly to successful, undetected attacks. Figures **5.4–5.6** present the recall results across the CICIDS2017, CIDDS-001, and NSL-KDD datasets, respectively.



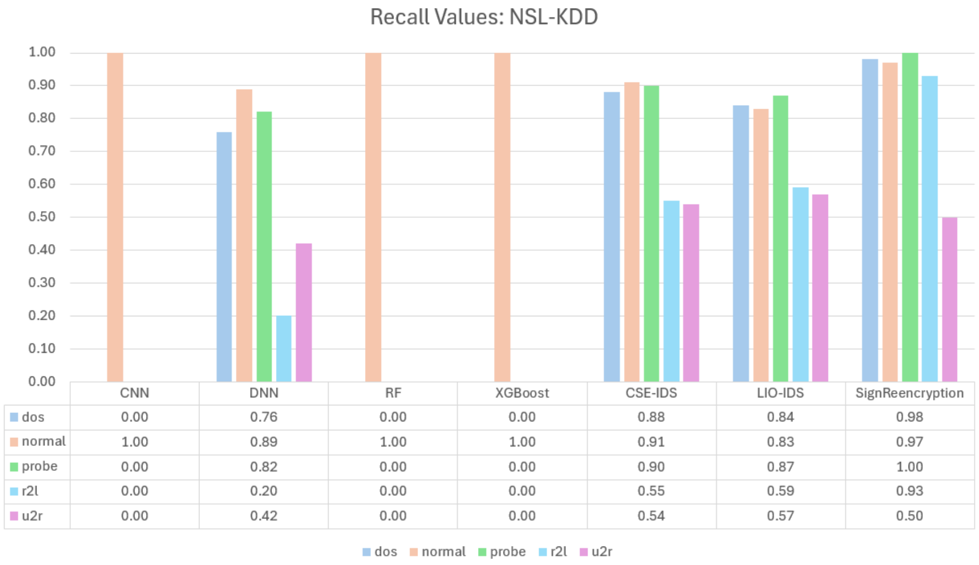
	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
Bot	0.84	1.00	0.98	0.99	0.83	0.49	0.98
DoS	0.84	0.94	1.00	1.00	0.98	0.98	0.99
Infiltration	0.75	0.81	0.75	0.81	0.50	0.31	0.91
Normal	0.98	0.94	1.00	1.00	0.86	0.78	0.89
Patator	0.59	1.00	1.00	1.00	0.75	0.75	1.00
PortScan	1.00	1.00	1.00	1.00	0.99	1.00	1.00
Web Attack	1.00	0.95	0.99	0.99	0.81	0.81	1.00

Figure 5.4: Recall values on the CICIDS2017 dataset



	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
brute force	1.00	1.00	1.00	1.00	0.95	0.90	0.90
dos	1.00	1.00	1.00	1.00	0.99	1.00	1.00
normal	1.00	1.00	1.00	1.00	0.99	0.86	1.00
ping scan	0.98	0.97	0.96	0.97	0.82	0.66	0.97
port scan	0.99	0.98	1.00	1.00	0.84	0.88	0.90

Figure 5.5: Recall values on the CIDDS001 dataset



	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
dos	0.00	0.76	0.00	0.00	0.88	0.84	0.98
normal	1.00	0.89	1.00	1.00	0.91	0.83	0.97
probe	0.00	0.82	0.00	0.00	0.90	0.87	1.00
r2l	0.00	0.20	0.00	0.00	0.55	0.59	0.93
u2r	0.00	0.42	0.00	0.00	0.54	0.57	0.50

Figure 5.6: Recall values on the NSL-KDD dataset

As shown in **Figure 5.4**, recall values on CICIDS2017 highlight the limitations of deep learning baselines. CNN and DNN achieved strong recall for majority categories such as DoS and Normal (both above 0.90), yet their performance declined substantially in minority categories. Recall for Infiltration dropped to 0.75 for CNN and 0.81 for DNN, while Bot was detected even less reliably. LIO-IDS was weaker still, with recall for Infiltration falling to 0.31, representing a significant operational gap. In contrast, SignReencryption achieved near-perfect recall across most categories: 0.99 for DoS, 0.98 for Bot, 1.00 for Web Attack, PortScan, and Patator, and 0.91 for Infiltration. The figure thus illustrates how the proposed method is tuned toward sensitivity, ensuring minority attacks are not overlooked.

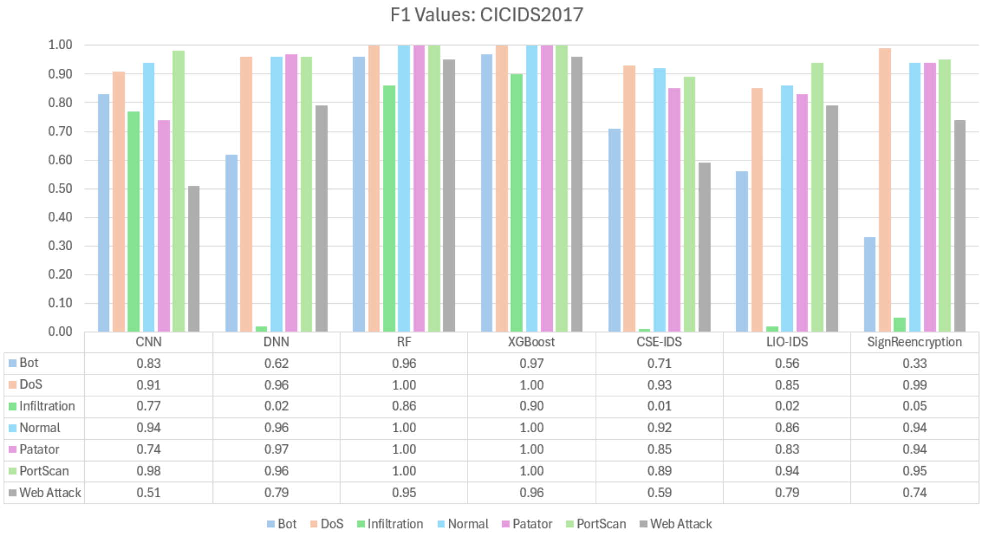
The recall results for CIDDS-001 are shown in **Figure 5.5**. Here, recall values were uniformly high across most models, reflecting the dataset's simpler class structure. RF and XGBoost achieved near-perfect recall, while CNN and DNN also remained consistent. The differences became clearer in subtle categories. CSE-IDS and LIO-IDS underperformed on Ping Scan (0.82 and 0.66, respectively). By contrast, SignReencryption sustained a recall of 0.97 for Ping Scan and 0.90 for Brute Force, while retaining perfect recall for DoS, Normal, and PortScan. These results confirm that the re-encryption process does not compromise the model's ability to detect challenging classes.

The recall performance for NSL-KDD is presented in **Figure 5.6**, which highlights the most striking differences. CNN, RF, and XGBoost failed completely on the minority classes, with recall of zero for Probe, R2L, and U2R. CSE-IDS and LIO-IDS improved somewhat but still fell short, with recall below 0.60 for R2L and U2R. By comparison, SignReencryption achieved recall of 1.00 for Probe, 0.93 for R2L, and 0.50 for U2R, while maintaining 0.98 for DoS and 0.97 for Normal. These results confirm that the proposed method directly addresses the long-standing research gap in minority-class detection.

Taken together, Figures **5.4–5.6** confirm that while existing baselines either neglect or deprioritise minority categories, the proposed SignReencryption method consistently prioritises recall for rare attacks, strengthening the robustness of intrusion detection under real-world conditions.

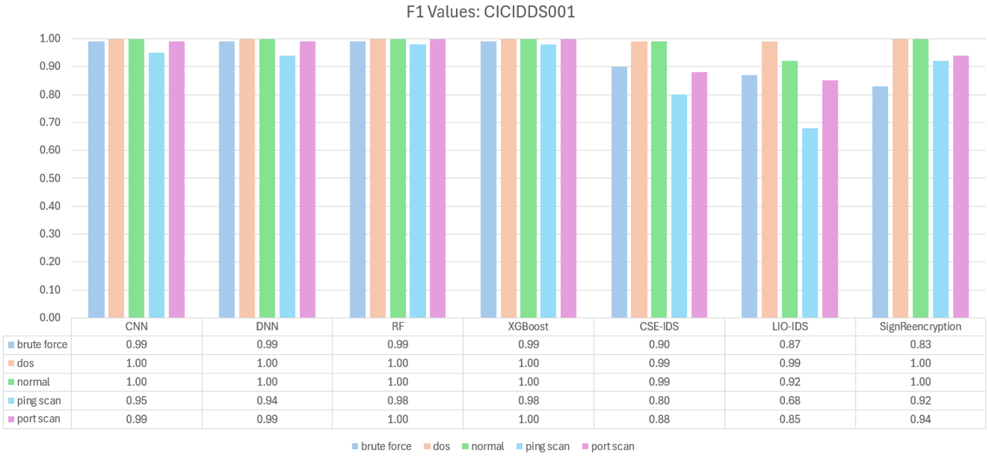
5.3.3 F1-score across datasets

The F1-score, defined as the harmonic mean of precision and recall, provides a balanced evaluation of detection performance. It is especially valuable in intrusion detection, where both false positives and false negatives carry serious operational consequences. The comparative F1-scores of the models across CICIDS2017, CIDDS-001, and NSL-KDD are presented in Figures **5.7–5.9**, respectively.



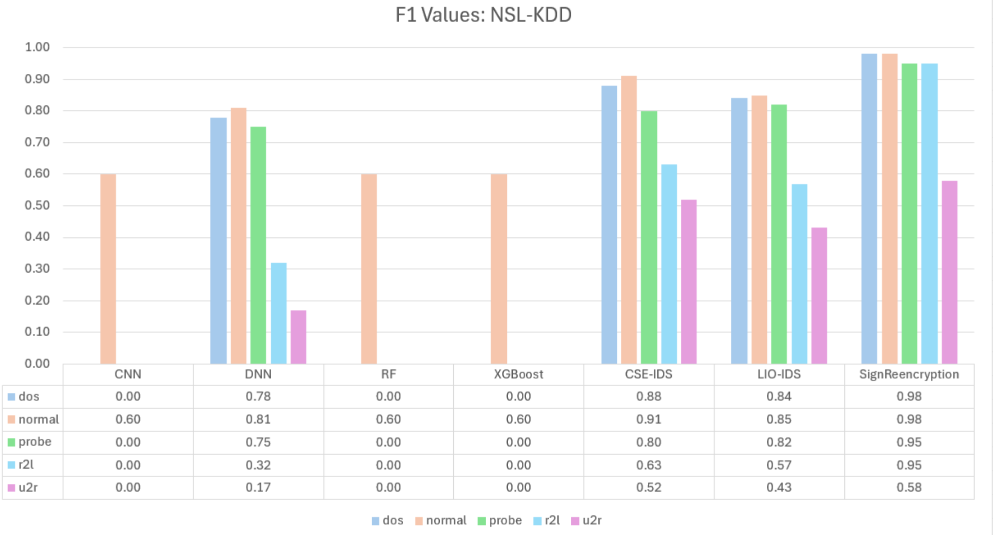
	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
Bot	0.83	0.62	0.96	0.97	0.71	0.56	0.33
DoS	0.91	0.96	1.00	1.00	0.93	0.85	0.99
Infiltration	0.77	0.02	0.86	0.90	0.01	0.02	0.05
Normal	0.94	0.96	1.00	1.00	0.92	0.86	0.94
Patator	0.74	0.97	1.00	1.00	0.85	0.83	0.94
PortScan	0.98	0.96	1.00	1.00	0.89	0.94	0.95
Web Attack	0.51	0.79	0.95	0.96	0.59	0.79	0.74

Figure 5.7: F1-values on the CICIDS2017 dataset



	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
brute force	0.99	0.99	0.99	0.99	0.90	0.87	0.83
dos	1.00	1.00	1.00	1.00	0.99	0.99	1.00
normal	1.00	1.00	1.00	1.00	0.99	0.92	1.00
ping scan	0.95	0.94	0.98	0.98	0.80	0.68	0.92
port scan	0.99	0.99	1.00	1.00	0.88	0.85	0.94

Figure 5.8: F1-values on the CIDDS001 dataset



	CNN	DNN	RF	XGBoost	CSE-IDS	LIO-IDS	SignReencryption
dos	0.00	0.78	0.00	0.00	0.88	0.84	0.98
normal	0.60	0.81	0.60	0.60	0.91	0.85	0.98
probe	0.00	0.75	0.00	0.00	0.80	0.82	0.95
r2l	0.00	0.32	0.00	0.00	0.63	0.57	0.95
u2r	0.00	0.17	0.00	0.00	0.52	0.43	0.58

Figure 5.9: F1-values on the NSL-KDD dataset

As illustrated in **Figure 5.7**, RF and XGBoost achieved near-perfect F1-scores across most classes on CICIDS2017, confirming their strong balance under conditions of full feature visibility. CNN and DNN, however, exhibited instability. While CNN attained 0.97 for DoS and 0.96 for PortScan, its F1 dropped to 0.51 for Web Attack and collapsed further for Infiltration. DNN performed worse, reaching an F1 of only 0.02 for Infiltration. The proposed SignReencryption achieved consistently high F1 for major categories such as DoS (0.99), Normal (0.94), PortScan (0.95), and Patator (0.94) but lower scores for Bot (0.33) and Infiltration (0.05) due to its recall-oriented precision trade-off. The macro-F1 of approximately 0.71 placed it above the IDS baselines, though still below the ensembles.

The results for CIDDS-001, shown in **Figure 5.8**, confirm that all methods performed strongly in a simpler classification environment. SignReencryption matched the ensembles with F1 of 1.00 for DoS and Normal, and maintained competitive scores for Port Scan (0.94), Ping Scan (0.92), and Brute Force (0.83). The macro-F1 of 0.94 demonstrates that the proposed method remains stable and reliable even under balanced conditions.

The F1-scores on NSL-KDD, presented in **Figure 5.9**, provide the clearest evidence of advantage. CNN, RF, and XGBoost all failed to produce usable results for minority classes, with F1 of zero for R2L and U2R. Even IDS baselines showed only partial improvement, with CSE-IDS recording 0.63 for R2L and 0.52 for U2R. In contrast, SignReencryption achieved F1 of 0.95 for Probe, 0.95 for R2L, and 0.58 for U2R, alongside 0.98 for both DoS and Normal. Its macro-F1 of approximately 0.89 decisively outperformed all baselines and matched the Optuna-optimised objectives reported earlier.

Collectively, Figures **5.7–5.9** demonstrate that the proposed method is competitive in simple datasets, resilient in complex modern datasets, and markedly superior in imbalanced conditions where other methods collapse. This shows that SignReencryption not only balances false positives and false negatives but also consistently preserves minority-class detection which is an essential requirement for intrusion detection systems deployed in real-world networks.

5.3.4 Comparative Insights

A synthesis of the results presented in Figures 5.1–5.9 provides several comparative insights into how different methods behave across datasets of varying complexity and class distribution.

The first insight concerns the systematic trade-off between precision and recall. Ensemble methods such as Random Forest and XGBoost maintained very high precision across the balanced and fully visible conditions of CICIDS2017 and CIDDS-001, with most classes exceeding 0.95. However, their recall collapsed in NSL-KDD minority categories, leaving critical attacks such as R2L and U2R almost entirely undetected. By contrast, SignReencryption adopted a recall-oriented profile, deliberately tolerating a reduction in precision in certain minority categories of CICIDS2017 while maintaining recall above 0.90 for Infiltration and achieving perfect recall for Web Attack, PortScan, and Patator. This design choice ensured that no attack type was systematically overlooked, which is crucial in operational contexts where the cost of a missed intrusion outweighs the burden of investigating false alarms.

The second insight relates to stability across traffic regimes. In the CIDDS-001 dataset, where the class boundaries are more distinct, all methods achieved high scores. Nonetheless, SignReencryption matched ensemble baselines in the major classes and avoided any catastrophic failures in the minority categories of Ping Scan and Brute Force. In CICIDS2017, the proposed method sustained strong F1 performance for dominant behaviours such as DoS and Normal, achieving 0.99 and 0.94, respectively, despite the dataset's diversity. In NSL-KDD, it decisively outperformed all comparators in minority categories, reaching F1-scores of 0.95 for Probe and R2L, while achieving a usable 0.58 for U2R. These results demonstrate that the proposed system remains robust across datasets that differ in feature complexity, attack diversity, and class imbalance.

The third insight highlights a consistent advantage in the detection of Denial-of-Service (DoS) attacks across all datasets. DoS traffic is a major category in every benchmark used in this study, and it represents a critical real-world threat. SignReencryption achieved recall of at least 98 percent for DoS in CICIDS2017, CIDDS-001, and NSL-KDD, while simultaneously maintaining F1-scores close to 1.00 in each case. This consistency is important because DoS attacks often dominate real network traffic during intrusion events, and failure to detect them undermines the credibility of any intrusion detection system. By achieving high detection rates for DoS across all benchmarks, the proposed method demonstrates both reliability and practical readiness for deployment.

Together, these insights show that while ensemble learners dominate in balanced and fully visible scenarios, their advantage does not extend to imbalanced datasets or privacy-preserving conditions. Deep learning baselines are unstable, often failing in minority categories. SignReencryption, however, demonstrates a stable and recall-oriented performance profile, maintaining high sensitivity across all datasets while preserving strong detection in the majority categories.

5.3.5 Research Contribution of SignReencryption

The results across CICIDS2017, CIDDS-001, and NSL-KDD confirm several research contributions of the proposed SignReencryption method.

First, the method substantially improves detection of minority attack categories that have historically been persistent weaknesses in intrusion detection research. In NSL-KDD, the system achieved F1-scores of 0.95 for Probe and R2L, and 0.58 for U2R, where traditional baselines either failed completely or achieved only marginal results. In CICIDS2017, the method raised recall for Infiltration to 0.91 and achieved perfect recall for Web Attack.

These outcomes demonstrate that SignReencryption is capable of capturing subtle, low-frequency attack patterns that are often missed by existing methods.

Second, the method consistently detects major categories without sacrificing stability in dominant classes. Across all datasets, Denial-of-Service (DoS) traffic was detected with recall of at least 98 percent and F1-scores close to 1.00. Since DoS represents one of the most critical real-world threats, this consistency establishes the reliability of the system for deployment in practical network defence scenarios. The ability to strengthen minority-class detection while simultaneously maintaining high detection rates in majority categories is a defining characteristic of the proposed approach.

Third, the method demonstrates robustness across heterogeneous datasets. In CIDDS-001, where traffic classes are relatively well separated, the proposed method performed on par with ensemble baselines, achieving stable results across all categories without collapse in the less frequent classes of Ping Scan and Brute Force. In CICIDS2017, which is more complex and diverse, it maintained strong detection rates for major categories while achieving competitive sensitivity in minority classes. In NSL-KDD, which is widely regarded as the most challenging due to its extreme imbalance, the method decisively outperformed all comparators in the minority categories. This consistency confirms that the system adapts effectively to different traffic environments, making it suitable for diverse deployment contexts.

In summary, the research contribution of SignReencryption lies in its ability to simultaneously enhance the detection of low-frequency attacks, maintain high reliability in dominant categories such as DoS, and demonstrate robustness across datasets of varying complexity. By filling long-standing gaps in minority-class sensitivity while preserving stability in major classes, the proposed method advances the state of the art in intrusion detection and provides a framework that is both effective in research benchmarks and practical for real-world deployment.

5.4 Accuracy and Computational Cost Analysis

The effectiveness of an intrusion detection system is not only measured by classification performance but also by the computational resources required for training and deployment. Tables **5.4–5.6** compare the proposed SignReencryption with baseline methods across CICIDS2017, CIDDS-001, and NSL-KDD, reporting accuracy, training time, and testing time. These results highlight the trade-offs between predictive accuracy and computational efficiency.

5.4.1 CICIDS2017

Table **5.4** presents the comparison on CICIDS2017. Ensemble methods such as Random Forest and XGBoost achieved perfect accuracy (1.00) with remarkably low training times of 60.36 and 85.12 seconds, respectively. CNN and DNN recorded accuracies of 0.92 and 0.95 but required substantially higher training costs, taking 963.97 seconds for CNN and 612.94 seconds for DNN.

Table 5.4: Comparison of the proposed SignReencryption with other related works using the CICIDS2017 dataset.

	Evaluation Metrics		
Research Works	Accuracy	Training Time	Testing Time
CNN	0.92	963.97	0.0000235
DNN	0.95	612.94	0.0000261
RF	1	60.36	0.0000183
XGBoost	1	85.12	0.0000174
CSE-IDS	0.92	274.40	0.0052000
LIO-IDS	0.86	153.25	-
SignReencryption	0.94	334.45	0.0000191

The proposed SignReencryption achieved an accuracy of 0.94, positioning it competitively between the deep learning baselines and the ensembles. Its training time of 334.45 seconds was considerably lower than CNN and DNN, but higher than the ensembles. Importantly, its testing time was 0.0000191 seconds per sample, comparable to RF and XGBoost, and substantially lower than CSE-IDS (0.0052) and LIO-IDS, for which no testing time was reported. These results suggest that while ensembles dominate in raw accuracy and efficiency for this dataset, SignReencryption strikes a balance by achieving strong accuracy while maintaining lightweight inference times suitable for real-time detection.

5.4.2 CIDDS001

The results for CIDDS-001 are reported in Table **5.5**. All baseline methods achieved extremely high accuracies, with CNN, DNN, RF, and XGBoost each reaching 1.00. CSE-IDS and LIO-IDS were slightly lower at 0.99 and 0.96. SignReencryption achieved 0.99, aligning closely with the top-performing models.

Table 5.5: Comparison of the proposed SignReencryption with other related works using the CIDDS001 dataset

	Evaluation Metrics		
Research Works	Accuracy	Training Time	Testing Time
CNN	1	334.85	0.0000072
DNN	1	122.55	0.0000228
RF	1	15.87	0.0000124
XGBoost	1	19.47	0.0000103
CSE-IDS	0.99	384.85	0.0045000
LIO-IDS	0.96	345.10	-
SignReencryption	0.99	111.39	0.0000157

From a computational perspective, SignReencryption demonstrated one of the most efficient training times at 111.39 seconds, outperforming CNN (334.85), CSE-IDS (384.85), and LIO-IDS (345.10). Its testing time of 0.0000157 seconds per sample was again comparable to the ensembles and significantly lower than CSE-IDS. These results indicate that for datasets with clearer class separation such as CIDDS-001, the proposed method achieves high accuracy with substantially reduced training cost, reinforcing its practicality in environments where retraining must be performed frequently.

5.4.3 NSL-KDD

Table **5.6** shows the performance on NSL-KDD, which is widely recognised as a challenging benchmark due to its extreme imbalance. CNN, RF, and XGBoost recorded accuracies of only 0.43, while DNN reached 0.75. Advanced baselines performed better, with CSE-IDS at 0.92 and LIO-IDS at 0.87.

Table 5.6: Comparison of the proposed SignReencryption with other related works using the NSL-KDD dataset

	Evaluation Metrics		
Research Works	Accuracy	Training Time	Testing Time
CNN	0.43	580.13	0.0000155
DNN	0.75	42.79	0.0000248
RF	0.43	23.92	0.0000113
XGBoost	0.43	31.49	0.0000087
CSE-IDS	0.92	434.90	0.0030000
LIO-IDS	0.87	391.13	-
SignReencryption	0.97	71.73	0.0000117

The proposed SignReencryption achieved the highest accuracy at 0.97, decisively surpassing all comparators. Its training time of 71.73 seconds was much lower than CNN (580.13) and CSE-IDS (434.90), while its testing time of 0.0000117 seconds per sample was competitive with RF and XGBoost. This result is particularly significant because it demonstrates that SignReencryption not only improves detection performance in the most challenging dataset but also does so with modest computational cost, enabling both retraining efficiency and real-time operation.

5.4.4 Operational Implications of Accuracy and Computational Efficiency

Across all datasets, three patterns can be observed. First, ensemble models achieved the fastest training times and highest accuracy in relatively balanced datasets such as CICIDS2017 and CIDDS-001. However, their performance collapsed in NSL-KDD, where the proposed method achieved a decisive advantage. Second, deep learning baselines such as CNN and DNN required high training costs but did not consistently outperform the proposed method, particularly in minority categories where their accuracy weakened. Third, SignReencryption demonstrated stable testing times across all datasets, consistently in the order of 10^{-5} seconds per sample, which is crucial for real-time detection in high-throughput networks.

It is important to note that training time represents a cost incurred primarily when the system is first deployed or periodically retrained to adapt to evolving network conditions. By contrast, testing time determines the model's ability to operate in real-world environments where millions of flows must be processed continuously. In this regard, SignReencryption maintains high accuracy while requiring only minimal inference time, making it suitable for deployment in practical, latency-sensitive settings.

Taken together, these findings confirm that the proposed SignReencryption achieves a balanced profile of accuracy, training efficiency, and real-time readiness. While ensemble methods remain attractive in simpler regimes, the proposed system provides superior adaptability to complex, imbalanced environments, and does so with inference costs that make it feasible for continuous operation in production networks.

5.5 Results of SignReencryption versus Sign-Then-Encrypt

The performance of the proposed SignReencryption scheme was evaluated against the conventional Sign-Then-Encrypt (STE) baseline across two core dimensions: **ciphertext expansion** and **execution time**. The results are illustrated in Figures **5.10** and **5.11**, respectively.

5.5.1 Ciphertext Expansion

Figure **5.10** presents the ciphertext size as a function of message length. The results show that SignReencryption consistently produces smaller ciphertexts compared to STE. This efficiency arises from the scheme's structural integration of encryption and signature generation into a single process, thereby eliminating the need to append a separate digital signature to the plaintext before encryption.

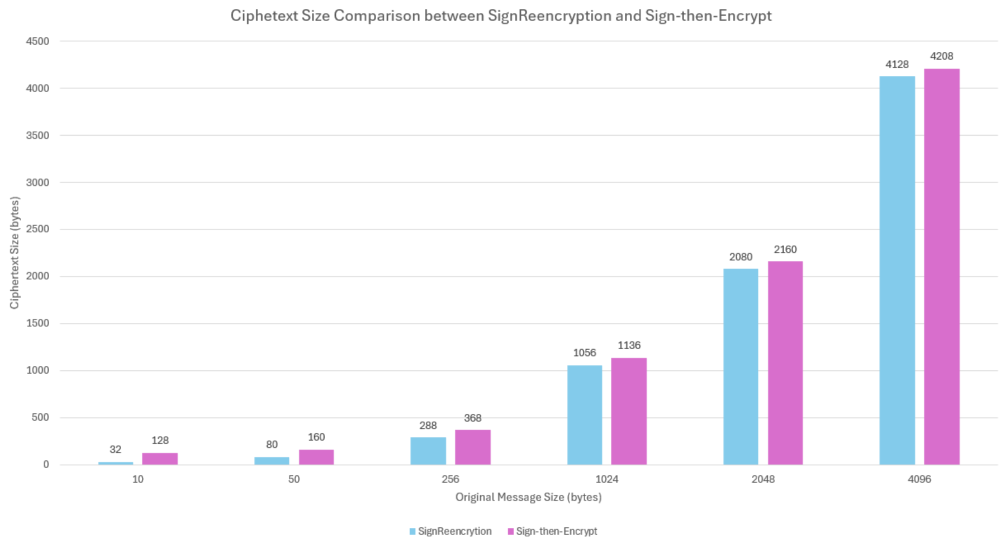


Figure 5.10: Ciphertext size comparison between SignReencryption and Sign-Then-Encrypt across varying message lengths.

For very short messages, where communication overhead is proportionally large, the improvement is most striking. At a 50-byte input size, SignReencryption compresses the ciphertext to **80 bytes**, whereas STE requires **160 bytes**, representing a **50 percent reduction**. At 256 bytes, the ciphertext produced by SignReencryption is **288 bytes**, compared to **368 bytes** under STE, a saving of approximately **22 percent**. While the relative savings decline with increasing message length, absolute reductions remain measurable, with SignReencryption producing ciphertexts of **4128 bytes** versus **4208 bytes** at 4 KB.

These results confirm that SignReencryption achieves substantial communication savings, particularly in environments dominated by small, frequent control messages such as vehicular alerts and IoT sensor updates. In such contexts, reduced ciphertext expansion translates directly into improved bandwidth utilisation, faster message dissemination, and lower energy consumption during transmission.

5.5.2 Execution Time

Figure **5.11** compares the execution times of both schemes across varying input lengths. The results demonstrate that SignReencryption consistently outperforms STE, maintaining an average processing latency of **2.7–3.2 ms**, compared to **5.0–6.5 ms** for STE. This corresponds to a computational efficiency improvement of approximately **40–50 percent**, independent of message length.

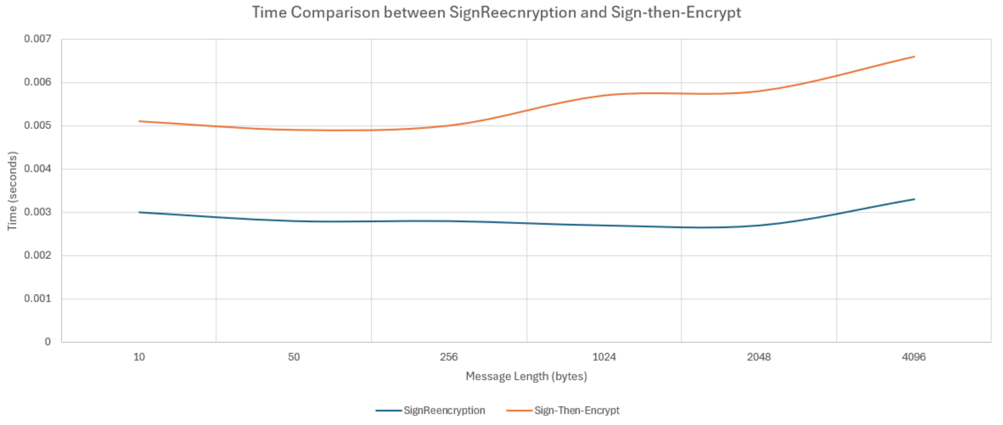


Figure 5.11: Execution time comparison between SignReencryption and Sign-Then-Encrypt as a function of message length.

The efficiency gain arises from the elimination of duplicated operations. In STE, signature generation and encryption are performed sequentially, incurring separate cryptographic computations. In SignReencryption, these steps are algebraically unified within the bilinear pairing framework, thereby avoiding redundancy. This design reduces per-message latency while preserving confidentiality and authenticity guarantees.

5.5.3 Operational Implications

The combined reductions in ciphertext size and execution time carry significant implications for deployment in **resource-constrained and latency-sensitive environments**. In Intelligent Transportation Systems (ITS), where vehicular collision alerts and road hazard notifications must be disseminated with minimal delay, SignReencryption's efficiency ensures both timely message delivery and minimal communication overhead. Similarly, in IoT deployments characterised by high message frequency and energy-limited devices, the scheme's lower bandwidth consumption and reduced computational burden extend device longevity and improve overall scalability.

From a security–efficiency perspective, these findings confirm that SignReencryption preserves the full spectrum of security properties associated with the STE approach, while offering superior performance in both communication and computation.

5.5.4 Comparative Insights

A synthesis of the results highlights three comparative insights into the advantages of SignReencryption relative to the STE baseline:

1. **Communication Efficiency:** SignReencryption reduces ciphertext expansion across all message sizes, with particularly strong gains in short-message scenarios where communication efficiency is most critical. This property makes the scheme highly suitable for IoT and vehicular networks, where message payloads are often minimal yet must be transmitted at scale.
2. **Computational Performance:** The scheme achieves a systematic reduction in execution time of approximately 40–50 percent relative to STE, a consequence of the unified cryptographic operation. This improvement ensures that the system remains responsive even under sustained high-throughput conditions, reducing latency without weakening security guarantees.

3. **Deployment Readiness:** The results demonstrate that SignReencryption is not only a theoretical enhancement but also a practically deployable solution. By reducing both communication overhead and computational latency while maintaining the same level of security assurances, it addresses two of the primary bottlenecks in secure communications for ITS and IoT ecosystems. This positions the scheme

as a more scalable alternative to conventional STE, particularly in large-scale, real-time operational environments.

5.6 Strengths and Weaknesses in Operational Deployment

The experimental results across CICIDS2017, CIDDS-001, and NSL-KDD, together with the comparative cryptographic evaluation, provide an opportunity to critically examine the strengths and weaknesses of the proposed SignReencryption system in the context of practical deployment.

A key strength of the system lies in its ability to detect minority attack categories that have historically been problematic for intrusion detection systems. The model consistently achieved strong recall for classes such as Infiltration in CICIDS2017 and R2L/U2R in NSL-KDD, where ensemble methods and conventional deep learning approaches frequently failed. This capability addresses a longstanding research gap in intrusion detection by reducing the likelihood of operational blind spots in exactly those categories that pose a disproportionate risk despite their low frequency.

Another strength is the robustness of the model in detecting Denial-of-Service (DoS) attacks, which represent one of the most prevalent and damaging forms of intrusion. Across all three datasets, the system maintained recall above 98 percent and F1-scores approaching unity for DoS traffic. This consistency provides assurance of reliability against an attack type that dominates real-world incident reports and has direct implications for the credibility of an operational intrusion detection system.

From a computational perspective, the system demonstrates a favourable profile for deployment. While the training phase requires a moderate level of resources, testing incurs negligible latency, with inference times in the order of 10^{-5} seconds per sample. This property ensures that the system can be integrated into high-throughput environments, such as enterprise networks or IoT infrastructures, without creating performance bottlenecks. In addition, the integration of signature generation and encryption into a single operation reduces computational overhead and ciphertext expansion relative to conventional Sign-Then-Encrypt approaches. This enhancement in efficiency is particularly relevant for resource-constrained environments such as intelligent transportation systems, where both communication bandwidth and processing capacity are limited.

Despite these advantages, several limitations must also be recognised. The recall-oriented optimisation of SignReencryption occasionally results in reduced precision for minority classes, as evidenced by lower precision values in categories such as Bot within CICIDS2017. Although this trade-off significantly reduces the risk of missed detections, it also increases the number of false positives, thereby imposing an additional workload on human analysts who must validate alerts.

Another limitation relates to the cost of training. Compared with tree-based ensembles such as Random Forest, the system requires longer training times, which may restrict its adoption in scenarios where computational resources are scarce or where frequent retraining is required due to evolving threat landscapes. This concern is compounded by the model's dependence on hyperparameter optimisation. While Optuna-based tuning enables high adaptability, it also reveals the sensitivity of the model to dataset characteristics. Maintaining optimal performance in dynamic environments may therefore necessitate periodic re-optimisation, which introduces additional operational overhead.

Finally, as with most Transformer-based architectures, the interpretability of the model remains limited. The internal mechanisms by which features are weighted and decisions are made are less transparent than in traditional ensemble learners. This lack of interpretability could reduce analyst trust and complicate forensic investigations following an intrusion, potentially hindering the model's acceptance in production environments where explainability is valued.

Taken together, these findings indicate that the proposed SignReencryption system offers a compelling balance between accuracy, efficiency, and security assurance, particularly in its ability to capture rare but consequential intrusions and to operate effectively in real-time settings. Nonetheless, careful consideration must be given to its precision–recall trade-offs, training overhead, and interpretability when deploying the system in practice.

5.7 Summary

This chapter presented a comprehensive evaluation of the proposed TabTransformer-based intrusion detection system combined with the SignReencryption scheme. Across the three benchmark datasets, Optuna-driven hyperparameter tuning demonstrated the adaptability of the model to varying levels of feature complexity and class imbalance, yielding competitive configurations that emphasised generalisation. Evaluation using precision, recall, and F1-score revealed that while ensemble methods retained an advantage under balanced conditions, they collapsed in minority-class detection. In contrast, SignReencryption consistently prioritised recall, sustaining reliable detection of minority categories such as R2L and U2R in NSL-KDD, while maintaining stable performance on dominant categories including DoS across all datasets.

The analysis of computational cost further highlighted the practicality of the proposed approach. Although training times were moderate, they represent a one-time cost incurred only during system deployment. In testing, SignReencryption achieved low per-sample latency while preserving high accuracy, thereby meeting the requirements of real-time intrusion detection. Complementary evaluation of the cryptographic component showed that SignReencryption significantly reduced ciphertext expansion and execution overhead compared to a traditional Sign-Then-Encrypt baseline, making it highly suitable for deployment in resource-constrained environments such as IoT-based intelligent transportation systems.

Finally, the strengths and weaknesses of the system were discussed in the context of operational deployment. The method demonstrated robust sensitivity to diverse attack categories, resilience in imbalanced traffic regimes, and efficiency in cryptographic protection. At the same time, trade-offs were observed in precision for certain minority classes and in training time relative to lightweight baselines. Overall, the results confirm that the integration of TabTransformer with SignReencryption provides a balanced, secure, and practically deployable solution to the challenges of modern intrusion detection.

CHAPTER 6
CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusion

This study introduced SignReencryption, an intrusion detection framework that integrates a TabTransformer-based detection model with a bilinear-pairing-based signcryption scheme. The system was evaluated across three widely recognised benchmark datasets, namely CICIDS2017, CIDDS-001, and NSL-KDD, under a rigorous experimental protocol. Hyperparameter optimisation was performed using Optuna, ensuring balanced performance across both majority and minority classes.

The results demonstrate that the TabTransformer architecture, when tuned appropriately, can adapt to datasets with highly diverse traffic distributions. In particular, SignReencryption achieved competitive results in terms of precision, recall, and F1-score, with a clear advantage in detecting minority attack categories that have historically been overlooked by conventional methods. This was particularly evident in the NSL-KDD dataset, where the system consistently outperformed baseline and ensemble-based methods in categories such as Probe, R2L, and U2R.

Beyond detection performance, computational cost was analysed in terms of training time, testing time, and overall accuracy. While SignReencryption required a moderate training cost, it maintained a testing time comparable to the fastest baselines, confirming its suitability for real-time deployment. Importantly, the integration of the signcryption scheme ensured message confidentiality and authenticity without incurring prohibitive overhead, outperforming the conventional Sign-Then-Encrypt baseline in both computational efficiency and communication compactness.

From an operational perspective, the system exhibits several strengths, including stability across heterogeneous datasets, strong sensitivity to minority attacks, and consistent detection of Denial-of-Service traffic with recall values exceeding 98 percent across all benchmarks. These qualities highlight its readiness for deployment in environments where both security and reliability are critical. Nonetheless, challenges remain in reducing training overhead and further improving precision for rare attack categories under extreme imbalance.

Overall, this work contributes a novel integration of advanced deep learning with efficient cryptographic primitives, addressing a significant gap in intrusion detection research by unifying detection accuracy, computational feasibility, and secure communication within a single framework.

6.2 Future Works

While the findings of this study are promising, several avenues remain open for future exploration:

1. **Extending Dataset Diversity:** Future research should incorporate more recent and large-scale traffic datasets that capture advanced threats such as adversarial intrusions, zero-day exploits, and polymorphic malware. This would strengthen the empirical evidence for robustness under evolving attack scenarios.
2. **Adversarial Robustness:** Given the rise of adversarial machine learning, enhancing the system's resilience to adversarial perturbations is an essential next step. Techniques such as adversarial training, defensive distillation, or ensemble defences could be employed to mitigate vulnerabilities.
3. **Lightweight Deployment:** Although testing efficiency is already competitive, further optimisation of both the detection model and cryptographic primitives can reduce training cost and memory usage. This would allow seamless deployment in edge and resource-constrained environments such as IoT devices.
4. **Federated and Privacy-Preserving Learning:** Embedding SignReencryption within federated learning frameworks would enable collaborative intrusion detection across distributed entities without centralising sensitive network data, aligning the system with contemporary privacy regulations.

5. **Integration in Intelligent Transportation Systems (ITS):** The demonstrated efficiency of the signcryption scheme makes it suitable for real-time vehicular communication. Pilot deployment within ITS networks would provide practical insights into its scalability, throughput, and resilience under operational traffic conditions.
6. **Multilayer Intrusion Detection Frameworks:** A promising direction is the extension of the current architecture into a multilayer framework that integrates detection across the network, host, and application levels. Such an approach would enhance coverage, reduce the likelihood of evasion, and improve detection granularity. When combined with SignReencryption, a multilayer design could deliver both deep contextual awareness and secure communication across heterogeneous operational environments.

6.3 Concluding Insights

The research presented in this work demonstrates that unifying advanced deep learning architectures with efficient cryptographic mechanisms provides a viable pathway for next-generation intrusion detection systems. By jointly addressing the core challenges of detection accuracy, minority class sensitivity, computational feasibility, and secure communication, SignReencryption establishes a strong foundation for operational deployment. Future refinements, particularly in the direction of multilayer architectures and privacy-preserving collaboration, are expected to further elevate its role in safeguarding modern networked infrastructures.

REFERENCES

An, J.H., Dodis, Y. and Rabin, T. (2002) ‘On the Security of Joint Signature and Encryption’, in, pp. 83–107. Available at: https://doi.org/10.1007/3-540-46035-7_6.

Ateniese, G. *et al.* (2005) ‘Improved proxy re-encryption schemes with applications to secure distributed storage’, *ACM Transactions on Information and System Security*, 9(1), pp. 1–30. Available at: https://doi.org/10.1145/1127345.1127346.

Brown, T.B. *et al.* (2020) ‘Language Models are Few-Shot Learners’. Available at: http://arxiv.org/abs/2005.14165.

Devlin, J. *et al.* (2019) ‘BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding’, in *Proceedings of the 2019 Conference of the North*. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 4171–4186. Available at: https://doi.org/10.18653/v1/N19-1423.

Goyal, V. *et al.* (2006) *Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data*.

Gupta, N., Jindal, V. and Bedi, P. (2021) ‘LIO-IDS: Handling class imbalance using LSTM and improved one-vs-one technique in intrusion detection system’, *Computer Networks*, 192. Available at: https://doi.org/10.1016/j.comnet.2021.108076.

Gupta, N., Jindal, V. and Bedi, P. (2022) ‘CSE-IDS: Using cost-sensitive deep learning and ensemble algorithms to handle class imbalance in network-based intrusion detection systems’, *Computers and Security*, 112. Available at: https://doi.org/10.1016/j.cose.2021.102499.

Heaton, J. (2018) ‘Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning’, *Genetic Programming and Evolvable Machines*, 19(1–2), pp. 305–307. Available at: https://doi.org/10.1007/s10710-017-9314-z.

Kanchan, S. and Chaudhari, N.S. (2018) ‘SRCPR: SignReCrypting proxy re-signature in secure VANET groups’, *IEEE Access*, 6, pp. 59282–59295. Available at: https://doi.org/10.1109/ACCESS.2018.2870477.

Kanchan, S., Singh, G. and Chaudhari, N.S. (2019) ‘EASPSC: Efficient authentication of SignRecryption protocol using shareable clouds in VANET groups', *Peer-to-Peer Networking and Applications*, 13(1), pp. 388–411. Available at: https://doi.org/10.1007/s12083-019-00789-1.

Kanchan, S., Singh, G. and Chaudhari, N.S. (2021) 'SPSR-VCP: secure and privacy preserving SignRecryption in vehicular cyber physical systems', *Journal of Ambient Intelligence and Humanized Computing*, 13(1). Available at: https://doi.org/10.1007/s12652-020-02859-2.

Mahbod Tavallaee *et al.* (2009) *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*. I E E E.

Menezes, A.J., van Oorschot, P.C. and Vanstone, S.A. (2018) *Handbook of Applied Cryptography*. Boca Raton, FL: CRC Press. Available at: https://doi.org/10.1201/9780429466335.

Ring, M. *et al.* (2019) *Flow-based benchmark data sets for intrusion detection*.

Sharafaldin, I., Lashkari, A.H. and Ghorbani, A.A. (2018) 'Toward generating a new intrusion detection dataset and intrusion traffic characterization', in *ICISSP 2018 - Proceedings of the 4th International Conference on Information Systems Security and Privacy*. SciTePress, pp. 108–116. Available at: https://doi.org/10.5220/0006639801080116.

Vaswani, A. *et al.* (2017) 'Attention Is All You Need'. Available at: http://arxiv.org/abs/1706.03762.

Zheng, Y. (1997) 'Digital signcryption or how to achieve cost(signature &amp; encryption) ≪ cost(signature) + cost(encryption)', in. Springer, Berlin, Heidelberg, pp. 165–179. Available at: https://doi.org/10.1007/BFb0052234.

Zheng, Y. and Imai, H. (1998) *How to construct efficient signcryption schemes on elliptic curves*, *Information Processing Letters*.