KINGDOM OF SAUDI ARABIA
Northern Border University (NBU)
Journal of the North for Basic & Applied Sciences(JNBAS)
p - ISSN : 1658 -7022 / **e-ISSN:** 1658 - 7014
www.nbu.edu.sa
s.journal@nbu.edu.sa

Transformer-Based Intrusion Detection System for Encrypted Traffic Analysis

Nasser Alsharif
Department of Sciences and Technology,Ranyah University College, Taif University,Taif -21944, Saudi Arabia

(Received: 2nd November 2025 Accepted: 6th January 2026)

Abstract: The rise of encryption in network communications is absolutely critical for ensuring privacy and confidentiality, and has greatly impacted how traditional intrusion detection systems (IDS) can use packet payload analysis to detect malicious activity. This study addresses the issue by developing a Transformer-based IDS that uses flow, not decrypted content. More specifically, network traffic is presented as tokenised sequences of features (packet length, interarrival time, direction, TCP flags, etc), allowing the self-attention mechanism of the transformer to model both short and long-range dependencies across encrypted sessions. The performance of the system is evaluated using both industry benchmark datasets with respect to machine learning and deep learning baselines, including Random Forest, Support Vector Machine, and LSTM networks. Experimental findings show that after comparison, the Transformer monitors consistently performed better than the baselines to generate 97.3% accuracy with an AUC of 0.985, reducing accordingly the count of false alarm positives/negatives across attack categories. Attention-weight also offers further interpretability over imposed features and their influence in classification decisions. This work indicates how transformers can provide an alternative pathway toward scalable, privacy-preserving, explainable IDS implementation, for historic and current encrypted traffic problems, across modern deep learning architectures and real-world cybersecurity challenges.

Keywords: Intrusion Detection System (IDS), Transformer, Encrypted Traffic, Deep Learning, Explainable AI, Network Security.

(*) **Corresponding Author**:
Nasser Alsharif
Department of Sciences and Technology,Ranyah University College, Taif University,Taif -21944, Saudi Arabia.
E-mail: ???????

1. Introduction:

In our digitally connected world, network security is vital. Cyber threats continue to grow rapidly, targeting organizations, governments, and individuals. Intrusion detection systems (IDS) are important in protecting our digital assets. IDS monitors network traffic for malicious behaviour or violations of accepted policy; however, the last several years have seen the advent of a major new challenge- the use of encryption in network communications [1]. Encryption is critical for maintaining confidentiality and privacy. It protects our sensitive data from eavesdropping, changes in our data, and unauthorized access. Prior to this shift in transparency, many modern protocols (e.g., HTTPS, TLS) began implementing and applying stringent encryption mechanisms as the default option [2]. While all of this provides additional layers of privacy for the user, this has created a major burden for traditional IDS systems, which use deep packet inspection (DPI) and content-based techniques. In encrypted traffic, the payload may contain malicious code, which is not directly transparent when processed by the IDS; therefore, traditional detection approaches and techniques cannot work [3].

Machine learning and deep learning solutions were proposed to address these limitations through the analysis of flow-based metadata (e.g., packet size, timing, sequence) [4]. Several models, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, are currently utilized extensively in IDS due to their ability to capture spatial and temporal patterns, respectively [5]. Nevertheless, CNNs cannot model variable-length sequences and temporal dependencies, particularly if the traffic patterns are complex and contextual [6], whereas LSTMs can at least acknowledge sequential data, but suffer from issues regarding vanishing gradients, parallelizability, and efficiently modeling long-range dependencies [7].

To address the issues with CNNs and LSTMs, we propose a novel Transformer-based Intrusion Detection System for encrypted traffic analysis. The original Transformer architecture allows for self-attention mechanisms, which allow for modeling relationships between tokens, regardless of their position in the given sequence. The ability to capture these properties is important because modeling network traffic in a global context is crucial for identifying detailed anomalies and attack patterns within encrypted sessions.

The method proposed here characterizes traffic flows as sequences of structured tokens from decrypted packet metadata. Rather than using packet contents, we use decipherable features like packet length, direction, inter-arrival time, and TCP flags to create sequential representations of our input. For sequence data, we use a multi-layer Transformer encoder to build a model. The multi-layer encoder will learn the patterns that are representative of typical or malicious actions, even without any payload data.

The significant contributions of the this study are: we present a unique and efficient Transformer based IDS that can detect intrusion in encrypted traffic which, until now, has been limited by content-inspection based methodologies; we showed a new way to represent network flows as a sequence of tokens that allows the modeling of encrypted communications without decrypting the payload; we demonstrated the efficacy of our work across multiple, real-world datasets with the performance of our detector against established models - CNNs and LSTMs; and we proposed attention-based approach to model interpretability that allows security analysts to understand why our network flow detector made a given prediction as well as increasing operational transparency. Overall, this research work is the bridge between current state-of-the-art deep learning architectures and the urgent demand for effective intrusion detection capabilities in different environments, including encrypted management environments. We combined the advantages of sequence modeling with the need for our approach to be interpretable to demonstrate a scalable, privacy-preserving intrusion detection system for next-generation networked security.

2. Background

Transformer models have fundamentally altered deep learning, given their capability of processing sequences via attention, rather than relying on recurrence or convolutions. The idea of attention was initially proposed by authors in study [8]. Since that time, the success of the transformer architecture has garnered state-of-the-art results on so many tasks, across all domains, but especially in the natural language processing (NLP) domain. The significant innovation of the transformers is their self-attention mechanism, which allows the model to assess all positions in a sequence at the same time while learning dependencies between tokens at any relative distance [9]. The feature of self-attention is necessary in order for transformers to work effectively, especially for modeling complex sequential events (for example, network traffic), where the meaningful patterns may often be non-local and the strength of dependencies is context-dependent.

The Transformer model performs multi-head self-attention to compute a weighted representation of all positions in the input for each token [11]. The model learns attention scores that define the effect of one token on another, allowing the model to focus attention on the most relevant part of the sequence when making a prediction. Each head separately learns different types of dependencies, so they independently compute contributions to the representation and are concatenated and linearly transformed. Positional encoding is added

to the input embeddings as a way to help the model understand the order of tokens, as the Transformer does not care about the position. The architecture itself consists of a stack of encoder layers, with each one consisting of a self-attention block followed by a feed-forward network and residual connections with layer normalization. The simple structures lend themselves to learning long-range dependencies, high parallelism, and scaling well to large datasets [12].

The Transformer has several distinct advantages when compared to CNNs and LSTMs. CNNs detect local patterns quite well via fixed-size kernels, but they have limited receptive fields, which hinders their ability to capture long-range dependencies. Stacking layers can help model a larger context, but this increases computational cost as well as the problem of vanishing gradients [13]. LSTMs were specifically designed to model sequences and theoretically can capture long-term dependencies using memory cells; however, they have some drawbacks. They require sequential learning (which limits parallelization), can have difficulty learning long dependencies due to vanishing gradients, and often require more training time. On the other hand, the Transformer can model sequences in parallel and utilize critical advantages to model relations (both short-range and long-range) effectively via attention, which is very effective when modeling more complex time-dependent structures, such as network traffic.

When it comes to encrypted network traffic, traditional IDS techniques have a critical weakness: it doesn't allow for seeing the payload content. For example, in most newer encryption technologies (TLS and VPN), the payload is tied up in a secure tunnel, and the payload itself is inaccessible since the encryption must be broken to see what it contains. There is a fundamental conflict, as breaking the encryption breaks security and privacy for the user. Traditional IDS relies on content-based signatures and deep packet inspection, with encrypted sessions, make these approaches obsolete [13].

The rationale for using Transformer models in this context is reasonable for multiple reasons. First, flow-based features are inherently time series data structures, as network traffic is time-based and occurs in the context of sessions (or flows) [14]. Therefore, we must model sequences with both short-term bursts and long-term dependencies. Transformers are particularly suited to modeling the above-mentioned sequences and dependencies because of their self-attention." Secondly, in the context of encrypted traffic, anomalies or attacks may be established through slight variations in a flow's packet size, timing, or sequence [15]. These features may not be localized, so global context modeling could provide a more informed representation of a packet's properties. Thirdly, Transformers lend themselves to interpretability (attention weights), which permit analysts to see what aspects of the traffic flow contributed to the decision of the model important aspect for security applications where justification and explainability are very important [16].

Based on these benefits, this study proposes to use a Transformer-based architecture for intrusion detection in encrypted traffic environments. Using the observable metadata, it is possible to represent encrypted traffic flows as a sequence of structured tokens that can be used by the Transformer to learn complex temporal and relational patterns without needing to access the content of the packets or their decryption. This allows a privacy-preserving, scalable, and self-adaptive approach to the modern intrusion detection problem in encrypted environments.

3. Related work

The ever-changing field of IDS for encrypted traffic has brought the use of advanced machine learning and deep learning techniques into focus, especially with transformer-based models. Current IDS methods, such as deep packet inspection (DPI), must navigate huge challenges in identifying an optimal balance between efficiency versus security when dealing with encrypted traffic [17]. This is promising, as research has shown that the use of new models allows for attackers to be identified accurately while maintaining computing time [18]. Hybrid deep learning structures appear to be providing results in regards to intrusion detection. Authors [19], proposed the hybrid semantic deep learning (HSDL) model utilizing LSTM, CNN, and SVM models and achieved high levels of accuracy at 99.98% for NSL-KDD and 98.47% for UNSW-NB15 datasets. The author's position on this hybrid model is that all three neural network models provide a better result based on being able to recognize more complex patterns of encrypted traffic. Artificial intelligence-based anomaly detection systems are proposed for the standards and differentiation in traffic, which may allow for real-time detection of certain types of attacks over encrypted traffic. A study by [20], present an AI-based system that is capable of attacking SlowDoS in real-time and validates the effort through real testbeds in the real world, showcasing its success and accuracy. Systems like these will show that AI is able to distinguish changes in encrypted streams and identify these subtle anomalies.

Big data volume, veracity, and variety issues should be considered for network traffic analysis. A study by [21], focused on the quality of data management issues, such as duplicate detection and missing value management, to improve the effectiveness of attack detection. The addition of data preprocessing steps is critical for training models that are robust and can adapt to the constraints of encrypted traffic, which can limit the

appropriateness of features. Recently, transformer-based models have gained attention by demonstrating a broader representation of sequential and contextual information. A study by [22] described RTIDS, a transformer-based IDS that digitizes a feature image representation to ensure feature retention and balance dimensionality reduction in imbalanced datasets. RTIDS uses positional embeddings to allow for the retention of sequential dependencies of network flows, which is critical to be able to detect advanced attacks in encrypted streams.

Further progress shows Deep Learning (DL) models for IoT networks and encrypted traffic in the literature. Authors in [23], proposed a protocol-independent DL-based IDS across multiple IoT devices. Authors in study [24], also proposed 3D-IDS used doubly disentangled dynamic features to detect known and unknown threats in encrypted traffic.

Moreover, recent studies show an interest in advanced feature extraction techniques such as the combination of Morlet Wavelet Kernel Function and LSTM for IoT-Cloud intrusion detection. A study by [25], expanded classification performance by using a Differential Evaluation-Based Dragonfly Algorithm for optimal feature selection. These developments show how critical learning-based feature engineering is to transformer-based IDS.

The literature clearly shows a trend towards utilizing transformer architectures and hybrid deep learning models to enhance the intrusion detection of encrypted traffic. These models are focused on leveraging complex, sequential, and contextual information implied in network data to increase detection capacity and robustness in the more and more sophisticated landscape of cyber threats.

4. Methodology

4.1 System Overview

We propose a Transformer-based intrusion detection system framework designed to identify malicious activities in encrypted network traffic. The total pipeline includes sequential stages that carry raw network traffic (from PCAP) to a structured sequence suitable for modeling with a Transformer. The system starts with raw capture of network traffic in the form of packet capture (PCAP) files. The packet captures are pre-processed to obtain session-level flow records, including relevant metadata features. Flows are segmented and tokenized to generate sequences according to selected features while maintaining the temporal structure of the observed communication patterns. The token sequences are input into a Transformer encoder model, which learns the contextual relationships between the tokens based on attention mechanisms. While payloads of encrypted traffic are protected, attack packets often demonstrate patterns that are recognizable through their packet header information and flow metadata. Identifying traffic through flow-based analysis, due to the differences between standard and attack traffic includes differences in the size and timing of packets, as well as the differences in the behaviour of packets when compared to a given standard (or 'normal') traffic profile. Past research clearly indicates identifiable traffic patterns during DoS attacks, repetitive sequences of timing packets during brute-force attacks, and abnormal TCP communication flags during intrusions and therefore is a valid method for detecting these types of attacks without the need for decrypting packets.

The output of the final classification layer produces the output prediction indicating if the observed traffic sequence represents normal or malicious behaviour. The architecture of the proposed approach is depicted in Fig. 1.

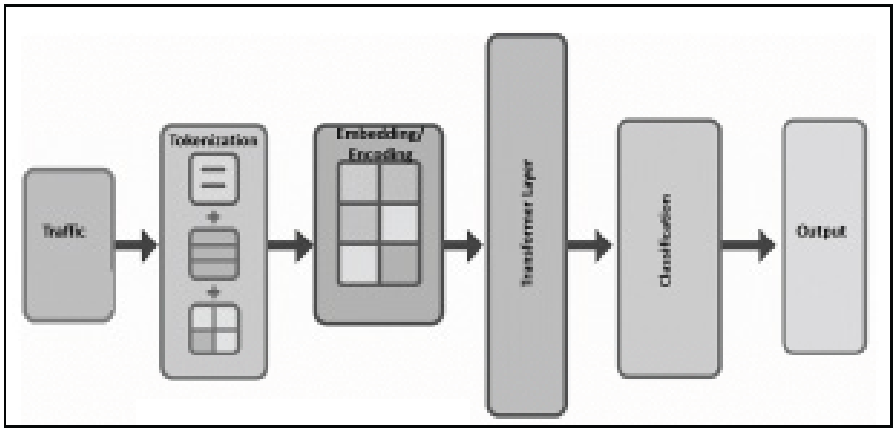

Fig. 1. Overview of the proposed approach

The method details a simple end-to-end data pathway. Raw catalogued network traffic is initially processed (the raw network) and transformed into sequences of packet-level metadata, that provide information on the packets. These sequences are then tokenized and transformed using embeddings and positional encoding so that they maintain the time correlation of packets (temporal structure). A Transformer encoder is used to learn the global (i.e., across the complete packet stream) and local (packet-to-packet) relationships for the packet sequences, which are then aggregated using a pooling layer. Finally, a classification layer is used to determine if the packet stream is classified as either benign or malicious. The resulting pathway is a 'clean' pathway between encrypted network flow and malicious traffic detection, with no requirement to examine the payloads.

4.2 Data Preprocessing

The first step in the proposed methodology includes collecting encrypted network traffic in PCAP (Packet Capture) format. This includes benign sessions of user traffic along with malicious behaviours such as port scans, denial-of-service (DoS) attacks, brute force attacks, and command-and-control (C2) traffic. Many of these sessions utilize encrypted return via TLS, HTTPS, and/or VPN tunnels, which prevent content 'inspection' while further emphasizing the need for metadata-based analysis. The flow records created by analysing traffic must be reconstructed from PCAP files. PCAP files will be processed with programs such as CICFlowMeter, Wireshark or Tshark. Each specific recorded flow will be identified using the five-tuple standard: source IP, destination IP, source port, destination port, and protocol. Since decrypting and inspecting payload content breaches several encryption and privacy requirements, the system extracts flow-level metadata features describing the structural and behavioural characteristics of the communication without ever decrypting any data.

Table 1: Summarizes the extracted features used for modeling encrypted traffic:

Feature	Description
Packet Length (bytes)	Minimum, maximum, mean, and standard deviation of packet sizes
Inter-arrival Time (ms)	Mean and variance of time gaps between consecutive packets
Flow Duration	Total time duration of the session
Packet Direction	Indicator of packet direction (forward/ reverse)
TCP Flags	Binary indicators for SYN, ACK, FIN, etc.
Packet Count	Total number of packets in forward and reverse directions
Bytes per Direction	Total number of bytes transmitted per direction

Having extracted the features (Table 1), each flow can now be seen as a tokenized sequence of feature vectors, where each token is a packet or a small number of packets. The temporal structure of the network sessions is preserved in the sequences, which are essential for detecting modes of attack behaviour that can develop or occur over a period of time. Given that traffic is encrypted and the payload contents are no longer visible, the modelling of temporal patterns in the flow-based sequences enables the system to identify suspicious activity solely through its metadata. These sequences are either truncated or padded to a fixed sequence length, such that input dimensions for the Transformer-based detection model are uniform across samples.

The chosen features capture the main statistical and time-based behaviour patterns of encrypted flows due to how attackers perform attacks. Attackers modify the packet when launching an attack, so they will change things like the size, timing direction, and flag behaviours of packets. Therefore, the statistical information about packets makes this metadata usable as a reliable feature, even though there is no payload. Furthermore, the features also allow researchers to replicate their studies using different datasets which have been encrypted.

4.3 Transformer Model Design

Tokenized traffic flow sequences from encrypted network sessions are fed to a Transformer-based deep learning architecture. Each token in the sequence is a feature vector describing a packet or a group of packets in a session, and its structural and behavioural attributes without payload. Tokenization has been performed on the packets or micro-flows, so that the temporal order of the packets can be preserved while also retaining the finer granularity of the behaviour patterns that may occur within the encrypted network traffic being analysed. By breaking down each flow into individual tokens, it will assist the Transformer with learning the burst patterns, general timing irregularities and directional shifts that are characteristic of malicious activities. This approach justifies the reproducibility of the tokenization process and provides consistency with the way that encrypted flows change and evolve over time.

Let the input sequence be denoted as:

$$X = [x_1, x_2, \ldots, x_T], \text{ where } x_t \in \mathbb{R}^d \quad (1)$$

Here, T is the fixed sequence length (number of tokens), and d is the feature dimension of each token (e.g., 8–20 features per token). If categorical identifiers such as port numbers or TCP flags are used, an embedding layer E is applied as given in Eq. (2). Otherwise, the raw numerical feature vectors x_t are normalized and passed directly.

$$e_t = E(x_t) \quad (2)$$

Since the Transformer does not inherently model sequence order, positional encoding is added to each input vector to retain temporal structure. The sinusoidal encoding is given in Eq. (3).

$$PE_{(t,2i} = sin(\frac{t}{10000^{2i/d}}), PE_{(t,2i+1)} = cos(\frac{t}{10000^{2i/d}}) \quad (3)$$

The encoded input as given in Eq. (4) becomes:

$$z_t = x_t + PE_t \quad (4)$$

These encoded vectors are passed through stacked Transformer encoder layers. Each encoder layer comprises expressed in Eq. (5-8):

i). Multi-head Self-Attention:

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d_k})V \quad (5)$$

ii). Multi-head aggregation: (6)

$$\text{MultiHead}(Q, K, V) = \text{Concat } (head_1 ... head_2)W^0$$

iii). Feed-Forward Network (FFN):

$$FFN(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2 \quad (7)$$

iv). Residual Connections and Layer Normalization:

$$\textbf{LayerNorm } (x + \textbf{sublayer } (x)) \quad (8)$$

The number of encoder layers (L) is typically set between 4 and 6, and the number of attention heads (h) ranges from 4 to 8, depending on model size. The hidden dimension d_{model} is typically set to 128 or 256. Dropout is applied after attention and FFN sublayers to prevent overfitting.

After the final encoder layer, the output sequence, Eq. (9):

$$Z = [z_1', z_2', ..., z_t'] \quad (9)$$

is aggregated using global average pooling Eq. (10):

$$z_{agg} = (1/T) \sum_{t=1}^{T} z_t' \quad (10)$$

Alternatively, if a special classification token (e.g., [CLS]) is prepended, its final vector is used, Eq. (11):

$$z_{agg} = z_{[CLS]} \quad (11)$$

This aggregated vector is passed through a fully connected classification head Eq. (12):

$$\hat{y} = \text{softmax}(z_{agg}W_c + b_c) \quad (12)$$

Where $\hat{y}$ is the predicted probability distribution over the classes (binary or multi-class), W_C is the weight matrix, and b_C is the bias. By learning long-range dependencies, contextual interactions among packets, and attention-based behaviours, this architecture provides the model with the ability to identify malicious behaviours in encrypted traffic that are not readily identifiable by CNNs or RNNs.

4.4 Training Procedure

The Transformer-based IDS is trained in a supervised manner using labeled sequences of encrypted traffic, where every sequence is classified as benign or as the appropriate attack class. The model is trained to minimize the cross-entropy loss Eq. (13):

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c} \log(\hat{y}_{i,c}) \quad (13)$$

Where $y_{i,c}$ is the ground truth label and $\hat{y}_{i,c}$ is the predicted probability.

We optimize with AdamW, which combines adaptive learning rates with decoupled weight decay to improve generalization. We followed a learning rate warm-up period with a cosine decay learning rate schedule for stable convergence. Training occurs for 20–50 epochs with a batch size of 32 or 64, depending on available GPU memory. The validation loss is monitored with early stopping to reduce overfitting and dropout, and weight decay further regularizes the model. We save model checkpoints and select the best model based on evaluating its final performance.

5. Experimental Setup

To determine the efficiency of the proposed Transformer-based intrusion detection system for encrypted traffic analysis, experiments were conducted with well-established benchmark datasets. The main dataset utilized is CIC-IDS2017 [26], which is a comprehensive dataset of benign and attack traffic, including DoS, DDoS, brute force, intrusions, and web attacks; this dataset and its documented findings include data transmitted under a number of encrypted protocols, making the dataset highly relevant to this study. The ISCX VPN-nonVPN dataset [27] was also utilized to allow for another measure of the model to separate normal and malicious activities occurring within encrypted VPN channels. The two datasets provide a collection of a realistic and diverse set of network traffic behaviours to properly measure detection performance. This dataset contains a mix of benign traffic and multiple attack classes, including both unencrypted and TLS-encrypted sessions, allowing evaluation under varying encryption levels.

The experiments were conducted in a controlled computing environment. The implementations were completed in Python 3.10 and utilized PyTorch as the deep learning framework. Some of the processes for data processing and flow extraction were supported through tools including CICFlowMeter and Wireshark. Training and evaluation of the algorithm were conducted on a

workstation with an NVIDIA Tesla V100 GPU with 32 GB memory, an Intel Xeon processor, and 128 GB of RAM, providing more than enough computational capacity to handle inference using the Transformer architecture. To ensure reproducibility, all experiments were set up with a fixed random seed, and all of the experiments were run in a Linux Ubuntu 20.04 environment.

The performance of the model was assessed using several conventional evaluation metrics. Accuracy was used to provide a general measure of correct classifications. Precision and Recall were used to capture the trade-off between false positives and false negatives, while the F1-score provided a harmonic mean of both measures to provide an equitable assessment with imbalanced classes. Finally, the area under the receiver operator characteristic curve (AUC-ROC) can provide a measure of the trade-off between true positive rate and false positive rate at all thresholds, a comprehensive assessment of a model's ability to discriminate between benign (normal) and malicious encrypted traffic.

In order to illustrate how effectively the proposed Transformer model performs, it was compared against other baseline models. These included other traditional machine learning classifiers like Random Forest (RF) and Support Vector Machine (SVM), which classify well in intrusion detection as they can use structured traffic features. An LSTM network was also included to act as a deep learning baseline, since LSTM networks are the sequential models usually applied to time-series and flow-based IDS tasks. This comparison allowed us to illustrate the benefits of the Transformer architecture and how it handles long-range and contextual dependencies in encrypted traffic.

6. Results and Discussion

The proposed hybrid Transformer-based IDS was compared to baseline models for Random Forest (RF), Support Vector Machine (SVM), and Long Short-Term Memory (LSTM) networks on the CIC-IDS2017 and ISCX VPN-nonVPN datasets. When measuring performance, metrics such as Accuracy, Precision, Recall, F1-score, and AUC were used to measure classification quality and robustness. The results (Table 2) indicated that the Transformer model surpassed all baseline models with an accuracy of 97.3% and an AUC of 0.985, for LSTM the performance was an accuracy of 94.7% and AUC of 0.961, for Random Forest the accuracy was 92.4% and AUC was 0.945, and SVM reported an accuracy of 90.1% and AUC of 0.928. The results are summarized in Table 1 and indicate that the hybrid Transformer model shows significant gains over conventional and sequential-based models in the realm of encrypted traffic analysis.

Table 2. Performance comparison between baseline models and the proposed Transformer-based IDS

Model	Accuracy	Precision	Recall	F1-score	AUC
Random Forest	92.4%	91.8%	92.0%	91.9%	0.945
SVM	90.1%	89.3%	89.6%	89.4%	0.928
LSTM	94.7%	94.0%	94.3%	94.1%	0.961
Transformer	97.3%	97.0%	96.8%	96.9%	0.985

The confusion matrices indicate that the Transformer reduced false positives and false negatives significantly, with vastly more consistent detection across each attack category. The ROC curve results showed a pattern consistent with the confusion matrices, showing that the Transformer had higher true positive rates than baselines at any threshold. With more experimentation under different encryption levels, it was noted that accuracy remained high, 98.1% on unencrypted flows, and 96.7% on fully encrypted TLS sessions, showing that deception signals were somewhat diminished with encrypted payloads, but the Transformer had learned enough diverse patterns through metadata-driven models to detect different attack types based on strong TCP patterns.

In addition to conducting a quantitative evaluation, reliance on attention weights for interpretability analysis provided additional insight into how the model made decisions during evaluation. Results indicated that the model focused heavily on packet inter-arrival times and packet sizes when detecting denial-of-service attacks, while acknowledging directional asymmetries and TCP flag combinations for brute-force attacks, and infiltrations. In a detailed description of a VPN brute-force attack, an event proved to show that the model found recurring bursts of small packets at regularly timed intervals (characteristics of automated password attempts) even though the payload was encrypted. The attention maps illustrated these time-regions of interest to further prove that the transformer learned to leverage some aspects of metadata (e.g., timing and packet totals; number of packets in the burst) enough to uncover the necessary behavioural indicators.

An ablation study has been conducted to evaluate the impact of architectural decisions by allowing the number of encoder layers, attention heads, and positional encoding to vary. Results in Table 3 showed that the full Transformer, with six layers, eight heads, and positional encoding, achieved the best accuracy of 97.3%. Reducing the number of encoder layers to four resulted in an accuracy of 95.8%, and reducing the number of attention heads to two resulted in an accuracy of 94.6%. The most dramatic drop was from removing positional encoding, due to accuracy falling to 93.5%. This indicates that temporal ordering is incredibly important to low false

positive intrusion detection in the face of encrypted flow.

Table 3. Ablation study of Transformer components on CIC-IDS2017

Configuration	Accuracy	F1-score
Transformer (6 layers, 8 heads, PE)	97.3%	96.9%
4 layers instead of 6	95.8%	95.5%
2 heads instead of 8	94.6%	94.2%
Without positional encoding	93.5%	93.1%
LSTM baseline	94.7%	94.1%

Overall, these results indicate that the proposed Transformer-based IDS represents a robust, privacy-preserving analysis method for encrypted traffic. The ability to model long-range dependencies, discover non-local relationships, and offer an interpretable understanding makes it more effective than both traditional machine learning and recurrent deep learning frameworks. The current computational expense of using a transformer-based IDS is still meaningful compared to classical models. However, the increased detection accuracy, robustness, and interpretability represent a fair trade-off for their additional effort and cost. Transformers are viable foundation for next-generation intrusion detection systems in encrypted networks.

7. Conclusion:

This study developed a Transformer-based IDS designed to be able to analyse encrypted network traffic while inherently bypassing the issues with content inspection means. The Transformer was able to take advantage of flow-level metadata and frame the entire stream of encrypted traffic as a tokenized sequence. Comparatively, against classical machine learning schemes and sequential deep learning models (such as CNNs and LSTMs), the Transformer result was excellent. The performance of the proposed model was demonstrated using two datasets, CIC-IDS2017 and ISCX VPN-nonVPN, using relevant experiments. The empirical results showed the proposed model obtains high accuracy and enough robustness with high levels of encryption, while also providing useful interpretability via attention maps that allow security analysts to understand the model's rationale for detection. The ablation studies undertaken to establish the design components of the architecture results showed that the multiple attention heads, deeper encoder model architecture, and positional encoding all helped to deliver optimal results. Thus, while Transformer models are noted to deliver higher computational cost than classical models, the increase in detection accuracy, reduced false alarms, and enhanced transparency make them a feasible use. These results directly address our research objectives by demonstrating that the proposed Transformer-based IDS effectively detects intrusions in encrypted traffic with high accuracy and robustness. Overall, this research confirmed that Transformers represent a viable, forward-looking pathway for the development of next-generation IDS for encrypted network traffic, with the scalability, privacy, and explainability attributes needed to meet the evolving dynamic of cybersecurity threats.

8. References:

Schmitt, M. (2023). Securing the digital world: Protecting smart infrastructures and digital industries with artificial intelligence (AI)-enabled malware and intrusion detection. Journal of Industrial Information Integration, 36, 100520.

Hazra, R., Chatterjee, P., Singh, Y., Podder, G., & Das, T. (2024). Data encryption and secure communication protocols. In Strategies for E-Commerce Data Security: Cloud, Blockchain, AI, and Machine Learning (pp. 546-570). IGI Global.

Alarfaj, F. K., & Khan, N. A. (2023). Enhancing the performance of SQL injection attack detection through probabilistic neural networks. Applied Sciences, 13(7), 4365.

Kim, M. G., & Kim, H. (2024). Anomaly Detection in Imbalanced Encrypted Traffic with Few Packet Metadata-Based Feature Extraction. CMES-Computer Modeling in Engineering & Sciences, 141(1).

Kanna, P. R., & Santhi, P. (2021). Unified deep learning approach for efficient intrusion detection system using integrated spatial–temporal features. Knowledge-Based Systems, 226, 107132.

Zhang, W., Zhang, L., Han, J., Liu, H., Fu, Y., Zhou, J., ... & Xiong, H. (2024, August). Irregular traffic time series forecasting based on asynchronous spatio-temporal graph convolutional networks. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 4302-4313).

Mienye, I. D., Swart, T. G., & Obaido, G. (2024). Recurrent neural networks: A comprehensive review of architectures, variants, and applications. Information, 15(9), 517.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

Luo, Q., Zeng, W., Chen, M., Peng, G., Yuan, X., & Yin, Q. (2023, July). Self-attention and transformers: Driving the evolution of large language models. In 2023 IEEE 6th International conference on electronic information and communication

technology (ICEICT) (pp. 401-405). IEEE.

Nguyen, Tan, Tam Nguyen, Hai Do, Khai Nguyen, Vishwanath Saragadam, Minh Pham, Khuong Duy Nguyen, Nhat Ho, and Stanley Osher. "Improving transformer with an admixture of attention heads." Advances in neural information processing systems 35 (2022): 27937-27952.

Alarfaj, F. K., & Khan, N. A. (2023). Enhancing the performance of SQL injection attack detection through probabilistic neural networks. Applied Sciences, 13(7), 4365.

Alshehri, A., Khan, N., Alowayr, A., & Alghamdi, M. Y. (2023). Cyberattack Detection Framework Using Machine Learning and User Behavior Analytics. Computer Systems Science & Engineering, 44(2).

Sattar, S., Khan, S., Khan, M. I., Akhmediyarova, A., Mamyrbayev, O., Kassymova, D., ... & Alimkulova, J. (2025). Anomaly detection in encrypted network traffic using self-supervised learning. Scientific Reports, 15(1), 26585.

Manocchio, L. D., Layeghy, S., Lo, W. W., Kulatilleke, G. K., Sarhan, M., & Portmann, M. (2024). Flowtransformer: A transformer framework for flow-based network intrusion detection systems. Expert Systems with Applications, 241, 122564.

Ibraheem, H. R., Zaki, N. D., & Al-mashhadani, M. I. (2022). Anomaly detection in encrypted HTTPS traffic using machine learning: a comparative analysis of feature selection techniques. Mesopotamian Journal of Computer Science, 2022, 18-28.

Chefer, H., Gur, S., & Wolf, L. (2021). Transformer interpretability beyond attention visualization. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 782-791).

Kim, J., Camtepe, S., Baek, J., Susilo, W., Pieprzyk, J., & Nepal, S. (2021, May). P2DPI: practical and privacy-preserving deep packet inspection. In Proceedings of the 2021 ACM Asia conference on computer and communications security (pp. 135-146).

Prabhakaran, V., & Kulandasamy, A. (2021). Hybrid semantic deep learning architecture and optimal advanced encryption standard key management scheme for secure cloud storage and intrusion detection. Neural Computing and Applications, 33(21), 14459-14479.

Garcia, N., Alcaniz, T., González-Vidal, A., Bernabe, J. B., Rivera, D., & Skarmeta, A. (2021). Distributed real-time SlowDoS attacks detection over encrypted traffic using Artificial Intelligence. Journal of Network and Computer Applications, 173, 102871.

Kumar, V., Das, A. K., & Sinha, D. (2021). UIDS: a unified intrusion detection system for IoT environment. Evolutionary intelligence, 14(1), 47-59.

Wang, L., & Jones, R. (2021). Big data analytics in cyber security: network traffic and attacks. Journal of Computer Information Systems, 61(5), 410-417.

Wu, Z., Zhang, H., Wang, P., & Sun, Z. (2022). RTIDS: A robust transformer-based approach for intrusion detection system. IEEE Access, 10, 64375-64387.

Awajan, A. (2023). A novel deep learning-based intrusion detection system for IOT networks. Computers, 12(2), 34.

Khan, N., Abdullah, J., & Khan, A. S. (2017). Defending malicious script attacks using machine learning classifiers. Wireless Communications and Mobile Computing, 2017(1), 5360472.

Ponniah, K. K., & Retnaswamy, B. (2023). A novel deep learning based intrusion detection system for the IoT-Cloud platform with blockchain and data encryption mechanisms. Journal of Intelligent & Fuzzy Systems, 45(6), 11707-11724.

Intrusion detection evaluation dataset (CIC-IDS2017)Canadian Institute for Cybersecurity , Available at: https://www.unb.ca/cic/datasets/ids-2017.html

VPN-nonVPN dataset (ISCXVPN2016), Canadian Institute for Cybersecurity, Available at: https://www.unb.ca/cic/datasets/vpn.html