Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2024.053437

ARTICLE

Encrypted Cyberattack Detection System over Encrypted IoT Traffic Based on Statistical Intelligence

Il Hwan Ji[1], Ju Hyeon Lee[1], Seungho Jeon[2] and Jung Taek Seo[2,*]

[1]Department of Information Security, Gachon University, Seongnam, 13120, Republic of Korea

[2]Department of Computer Engineering (Smart Security), Gachon University, Seongnam, 13120, Republic of Korea

*Corresponding Author: Jung Taek Seo. Email: seojt@gachon.ac.kr

Received: 30 April 2024 Accepted: 13 August 2024 Published: 27 September 2024

ABSTRACT

In the early days of IoT's introduction, it was challenging to introduce encryption communication due to the lack of performance of each component, such as computing resources like CPUs and batteries, to encrypt and decrypt data. Because IoT is applied and utilized in many important fields, a cyberattack on IoT can result in astronomical financial and human casualties. For this reason, the application of encrypted communication to IoT has been required, and the application of encrypted communication to IoT has become possible due to improvements in the computing performance of IoT devices and the development of lightweight cryptography. The application of encrypted communication in IoT has made it possible to use encrypted communication channels to launch cyberattacks. The approach of extracting evidence of an attack based on the primary information of a network packet is no longer valid because critical information, such as the payload in a network packet, is encrypted by encrypted communication. For this reason, technology that can detect cyberattacks over encrypted network traffic occurring in IoT environments is required. Therefore, this research proposes an encrypted cyberattack detection system for the IoT (ECDS-IoT) that derives valid features for cyberattack detection from the cryptographic network traffic generated in the IoT environment and performs cyberattack detection based on the derived features. ECDS-IoT identifies identifiable information from encrypted traffic collected in IoT environments and extracts statistics-based features through statistical analysis of identifiable information. ECDS-IoT understands information about normal data by learning only statistical features extracted from normal data. ECDS-IoT detects cyberattacks based only on the normal data information it has trained. To evaluate the cyberattack detection performance of the proposed ECDS-IoT in this research, ECDS-IoT used CICIoT2023, a dataset containing encrypted traffic generated by normal and seven categories of cyberattacks in the IoT environment and experimented with cyberattack detection on encrypted traffic using Autoencoder, RNN, GRU, LSTM, BiLSTM, and AE-LSTM algorithms. As a result of evaluating the performance of cyberattack detection for encrypted traffic, ECDS-IoT achieved high performance such as accuracy 0.99739, precision 0.99154, recall 1.0, F1 score 0.99575, and ROC_AUC 0.99822 when using the AE-LSTM algorithm. As shown by the cyberattack detection results of ECDS-IoT, it is possible to detect most cyberattacks through encrypted traffic. By applying ECDS-IoT to IoT, it can effectively detect cyberattacks concealed in encrypted traffic, promoting the efficient operation of IoT and preventing financial and human damage caused by cyberattacks.

KEYWORDS

IoT cybersecurity; IoT encrypted traffic; IoT cyberattack detection

Nomenclature

IoT	Internet of Things
ECDS-IoT	Encrypted Cyberattack Detection System for IoT
IIoT	Industrial Internet of Things
SSL	Secure Sockets Layer
TLS	Transport Layer Security
HTTPS	Hypertext Transfer Protocol Secure
DPI	Deep Packet Inspection
ICS	Industrial Control System
AI	Artificial Intelligence
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
TPR	True Positive Rate
FPR	False Positive Rate

1 Introduction

The Internet of Things (IoT) is a technology that aims to make the real world more intelligent by connecting things without human intervention [1]. IoT has been introduced and used in many fields, such as smart cities, smart factories, smart farms, energy management, healthcare, smart homes, smart transportation infrastructure, aviation/space, marine ships, defense, and construction [2]. IoT provides better services by connecting multiple devices using wired and wireless communication technology [3]. In the early days of IoT's introduction, it was challenging to introduce encryption communication due to the lack of performance of each component, such as computing resources such as CPUs and batteries, to encrypt and decrypt data [4]. Plain text-based communication is a security weakness, making it vulnerable to various cyberattacks such as reply attacks, sniffing, snooping, and spoofing. Since IoT is applied and utilized in many vital areas, cyberattacks on IoT can cause astronomical financial damage and casualties [5]. For this reason, the application of encryption communication to IoT has been required, and due to the improvement of computing performance of IoT devices and the development of lightweight cryptography, the application of encryption communication to IoT has become possible. Currently, encrypted communication based on secure sockets layer (SSL)/transport layer security (TLS) is essentially applied to most IoT, and encrypted communication has been used in a majority of IoT [6].

Although encrypted communication has made it possible to respond to cyberattacks caused by vulnerabilities in plaintext communication, attackers also perform cyberattacks through protected communication channels. One survey explained that attacks through encrypted channels continued to increase from 57% in 2020 to 80% in 2021, and more than 85.9% of attacks in 2023 were made using encrypted channels [7,8].

Detection research has been actively conducted to detect and respond to cyberattacks on IoT in the existing plaintext communication environment. Cai et al. [9] proposed a CapBad anomaly detector based on payloads in network packets to detect cyberattacks on the Industrial Internet of Things (IIoT). CapBad models industrial control protocol packets automatically learns the payload characteristics of the packets, and detects anomalies based on the learned information. As a result

of the performance evaluation of CapBad, high performance of ROC_AUC 0.974 was derived. Kim et al. [10] proposed autoencoder-based payload analog detection (APAD) for anomaly detection for IIoT. After network collection, the framework derived identifiable information, including payload as features, and preprocessed it. After that, cyberattacks are detected through an autoencoder-based anomaly detector. As a result of APAD's performance evaluation, a high accuracy performance of 0.944 and recall of 0.983 were derived. Like the previously presented research, most IoT target cyberattack detection research conducted deep packet inspection (DPI) on network packets to identify critical information and use it as a feature or use direct or indirect statistical information of payload as a feature. However, extracting evidence of an attack based on crucial information from a network packet is no longer valid because critical information, such as the payload in a network packet, is encrypted by encrypted communication [11]. For this reason, a technology capable of detecting cyberattacks over encrypted network traffic occurring in IoT environments is required.

This research proposes the Encrypted Cyberattack Detection System for IoT (ECDS-IoT), which derives valid features through statistical analysis of encrypted traffic and performs cyberattack detection in encrypted network traffic in the IoT environment. The proposed ECDS-IoT consists of a statistics-based feature extractor, a feature preprocessor, and a cyberattack detector. The statistics-based feature extractor derives effective features for cyberattack detection through statistical analysis of identifiable information and identifies information in encrypted traffic. The feature preprocessor performs feature selection, normalization, and missing value removal on the features derived by the statistics-based feature extractor for efficient training of the cyberattack detector and detection of cyberattacks. Cyberattack Detector learns only normal encryption traffic data and detects cyberattacks by classifying input data into normal encryption traffic and encryption traffic generated by cyberattacks. In this paper, CICIoT2023 [12], a dataset including TLS-based normal encryption traffic collected in IoT environments and encryption traffic generated by seven categories of cyberattacks, is employed to evaluate the cyberattack detection performance of ECDS-IoT. As a result of the experiment, high performance is derived for the Cyberattack Detector designed based on long short-term memory based autoencoder (AE-LSTM), such as accuracy 0.99739, precision 0.99154, recall 1.0, F1 score 0.99575, and ROC_AUC 0.99812. The contribution of this paper is as follows:

- It proposes ECDS-IoT, which has effective feature derivation, feature preprocessing, and cyberattack detection procedures for cyberattack detection for encrypted traffic generated in IoT environments.
- ECDS-IoT proposed in this research is trained and validated using CICIoT 2023, a dataset containing positive encryption traffic generated in IoT environments and encryption traffic generated by seven categories of cyberattacks, and performs successful cyberattack detection by deriving high performance such as accuracy 0.99739, precision 0.99154, recall 1.0, F1 score 0.99575, ROC_AUC 0.99812.
- ECDS-IoT derives similar or higher performance in cyberattack detection compared to existing research on detecting cyberattacks based on crucial information, such as payloads of network packets.

The remainder of this paper is organized as follows. Section 2 explains the overview of IoT and possible types of cyberattacks on IoT and analyzes AI-based cyberattack detection technology for IoT and existing research related to cyberattack detection technology over encrypted traffic. Section 3 presents ECDS-IoT for cyberattack detection over encrypted traffic generated in IoT environments. Section 4 conducts a performance evaluation on ECDS-IoT and performs a comparative analysis with

existing research. Section 5 presents and discusses the limitations of our research. Section 6 presents the conclusions of this research and future work directions.

2 Background and Related works

Sections 2.1 and 2.2 of this section present an overview of IoT and the types of cyberattacks that can occur against IoT targets, respectively. Section 2.3 analyzes the research on AI-based cyberattack detection technology for IoT to detect cyberattacks on IoT. Section 2.4 analyzes the research on cyberattack detection technology over encrypted traffic.

2.1 Internet of Things (IoT) Overview

IoT refers to a technology network that enables physical objects to exchange information and interact with each other or with a central server through the Internet. IoT is built around IoT devices, physical devices such as sensors and actuators that can be managed, and management systems such as central servers to collect information on the site and control on-site devices [13]. IoT is applied to each infrastructure to analyze the state of the environment and infrastructure based on the collected data and automatically deliver necessary commands to field devices based on the analyzed information, thereby automating the process and increasing productivity. Most IoT devices support wireless communication, so there are relatively few restrictions on the field application of IoT devices. For this reason, IoT is applied to various fields such as smart cities, smart factories, smart farms, energy management, healthcare, smart homes, smart transportation infrastructure, aviation/space, marine ships, defense, and construction fields to provide better services [3].

Fig. 1 shows the IoT architecture. IoT architecture is composed of a device layer, communication layer, IoT gateway, IoT platform, and application.

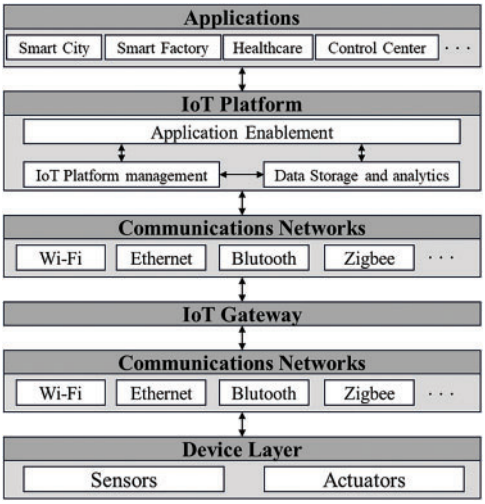


Figure 1: IoT architecture [13]

The device layer includes sensors and actuators located at the site. Sensors are hardware devices such as temperature/humidity measurement sensors, speed measurement sensors, and brightness measurement sensors that measure data generated in the surrounding environment or other systems. An actuator is a hardware device such as a motor, switch, or current control device that performs operations based on user or management system commands. In the device layer, Sensors and Actuators

required in each IoT application environment, such as smart cities, smart factories, and energy management facilities, exist.

The communication layer is a network for data transmission and reception between the device layer, IoT gateway, and IoT platform. The communication layer is responsible for forwarding and routing network packets and supports protocols such as IPv4, IPv6, and 6LoWPAN. IoT devices use wireless communication standards such as Zigbee, Z-Wave, Bluetooth, Wi-Fi, LoraWan, LTE-M, Sigfox, and MQTT [13]. In the past, IoT has relied on unencrypted communication protocols to accommodate the limitations of low computing resources and to ensure data availability. However, as cyber threats to IoT increase, encrypted communication based on SSL, TLS, and Datagram Transport Layer Security (DTLS) protocols is becoming essential to counter them [14]. More than 62% of IoT information exchanges are made through TLS/SSL-based encryption communication [6].

IoT gateway can be introduced to solve different communication networks, data communication protocols, and data format problems of different IoT devices. The IoT gateway can provide an efficient communication environment by providing a common communication network, data communication protocol, and data format within a network.

IoT platform provides three functions: device management and connection, data storage and analysis, and application support, allowing the sensor's collected data to be monitored and the actuator to be controlled through the application.

As IoT is applied and used in various fields, cyberattacks on IoT cause economic damage, and cyberattacks on IIoT existing in the industrial control system (ICS) can cause economic damage and casualties.

2.2 IoT Cyberattacks Type

IoT is being applied and utilized in various fields for efficiency and automation. However, various security vulnerabilities exist due to the absence of security measures, a lack of computing resources in IoT devices, and increased connectivity with various devices. In addition, it has vulnerabilities, such as physical access to IoT devices. For this reason, it has a different cyberattack vector from the general IT environment and various security vulnerabilities so that various cyber threats can occur. Table 1 shows the types of cyberattacks that can occur by component in the IoT.

Table 1: Cyberattacks type on IoT [15]

Component	Cyberattacks type
IoT devices	Sybil attack Buffer overflow attack Blueborne attack Rolling code attack Brute force attack
IoT Gateway and internal network of the gateway	MITM attack DNS poisoning attack Wormhole attack Replay attack Injection attack

(Continued)

Table 1 (continued)

Component	Cyberattacks type
Control devices and the servers	Back doors Exploits attacks SQL injection Weak authentication DDoS attacks Malicious applications

Table 1 presents possible cyberattacks on IoT devices, the IoT gateway, the internal network of the gateway, control devices, and the servers in IoT. These cyberattacks are caused by various security vulnerabilities, such as the absence of access control means, lack of data encryption, and weak authentication. As listed in Table 1, various cyber threats to IoT cause much damage, such as economic and casualties. If encryption communication is applied to prevent cyberattacks, MITM attacks such as sniffing and spoofing and some cyberattacks, including reply attacks, may be prevented. If an IoT device using encrypted communication is privileged, hijacked, or infected with malware, most of the cyberattacks presented in Table 1 can occur through the encrypted channel. Due to the nature of IoT, it is connected to many devices, but it is challenging to manage the security of all devices, so the cyberattacks mentioned above are more likely to occur.

For this reason, cybersecurity technology is needed to detect and respond to encrypted cyberattacks in advance by monitoring encrypted network traffic in IoT.

2.3 AI-Based Cyberattack Detection Technology for IoT

Various research has been conducted to detect cyberattacks on IoT. The existing rule-based detection method for network traffic targets and the signature-based detection method have disadvantages. The rule-based detection method has a problem: the rule update cycle to respond to rapidly changing attacks cannot keep up with the speed of change in the attack. The signature-based detection method is disadvantageous because it cannot detect unknown attacks [16,17]. Anomaly detection research using artificial intelligence (AI) is conducted to compensate for the weaknesses of these existing anomaly behavior detection methods and effectively detect network traffic generated in IoT environments.

Chang et al. [18] proposed the Hierarchical Animal Detection Framework for IoT (HADIoT) for cybersecurity for IoT, which consists of various heterogeneous IoT devices. The operation method of the framework is that the local edge server preprocesses and normalizes data by referring to the data pattern unique to each device. The framework performs cyberattack detection on the local area based on data type, payload, protocol, and port information. After that, the local edge server delivers the processed data to the cloud server, and the cloud server performs global cyberattack detection that requires higher computing capacity based on the received data. In this research, performance verification of HADIoT was performed using the ISCX 2012 [19] dataset, and high cyberattack detection performance such as a True Positive Rate (TPR) of 0.9812 and False Positive Rate (FPR) of 0.0453 was derived.

Liu et al. [20] proposed a payload-based anomaly detection framework that can be distributed to IoT edge devices to detect cyberattacks on IoT. This research identified payloads in network packets

and preprocessed them in a form that can be learned by machine learning and deep learning algorithms and used them as features. In this research, a CNN-LSTM-based cyberattack detection model was designed. For the CNN-LSTM-based anomaly detection model, the high performance of F1 score 0.9732 was derived from verifying the cyberattack detection performance using the CICIDS 2017 [21] and ISCX 2012 [19] dataset.

Alanazi et al. [22] proposed machine learning-based anomaly detection methods such as decision tree (DT), support vector machine (SVM), K-nearest neighbors (KNN), and linear discrete analysis (LDA) to detect anomalies for large-scale IoT ecosystems such as IIoT. The proposed anomaly-based IDS has three stages: preprocessing, feature selection, and anomaly classification. This research employed minimum redundancy, maximum relevance (MRR), and neighborhood components analysis (NCA) to reduce the data dimension and improve detection performance. Normal and anomaly data are classified based on DT, SVM, KNN, and LDA algorithms in the anomaly classification stage. X-IIoTID [23] is a network traffic dataset collected in the IIoT environment to verify the anomaly detection method. As a result of the verification, an accuracy 0.9958, sensitivity 0.9959, specificity 0.9958, and F1 score 0.9959 were derived.

Tomar et al. [24] proposed an anomaly detection method based on VGG-16 and VGG-19 models to detect possible cyberattacks such as infusion attacks, man-in-the-middle attacks, information gathering, malware attacks, and DDoS/Dos attacks on IoT. In this research, the Edge-IIoT [25] dataset, including normal network traffic collected from IoT testbeds and network traffic generated by cyberattacks, was employed to evaluate the cyberattack detection performance of the proposed cyberattack detection method. As a result of the verification, the anomaly detection performance of the VGG-19 model resulted in an accuracy of 0.99, and the classification accuracy for 15 cyberattacks was 0.948.

However, due to the development of IoT devices, they have sufficient computing resources, and the application of cryptographic communication to IoT is increasing as cyberattacks using vulnerabilities in plaintext communication increase. X-IIoTID [23] was utilized to verify the work of Alanazi et al. [22], which uses statistical values of network traffic and includes information from payloads. Chang et al. [18,20,24] detected cyberattacks based on information about payloads and messages in network packets. Since the related research presented above detects cyberattacks based on important information, such as payloads of network packets, it is not easy to apply to the encrypted communication environment. For this reason, research on cyberattack detection technologies over encrypted traffic occurring in IoT environments should be conducted.

2.4 Cyberattack Detection Technology over Encrypted Traffic

As various security threats arise due to vulnerabilities in the existing plaintext communication method, most Internet communication is currently encrypted. However, cyberattacks through encrypted channels continued to increase from 57% in 2020 to 80% in 2021, with more than 85% of attacks performed in 2022 through encrypted channels [7,8]. Cyberattacks through encrypted channels must be detected and responded to, but because critical data in network packets are encrypted, conventional anomaly detection methods using features extracted through Deep Packet Inspection (DPI) cannot be applied. Accordingly, research is actively conducted to detect cyberattacks based on features that can be extracted from encryption traffic.

Chao [26] conducted anomaly detection research for network traffic encrypted by SSL and TLS communication channels. The researchers entered the encrypted network traffic into Zeek IDS [27], an open-source intrusion detection system and network analysis framework, for real-time network

traffic analysis. Since then, the researchers have derived the information of the conn.log (connection information), x509.log (certificate information), and ssl.log (SSL/TLS information) files generated by analyzing the network traffic encrypted by Zeek IDS as features. This research used the CTU-Malware-Captures [28] dataset to verify the proposed methodology. This dataset includes encrypted normal traffic collected from the Internet environment and dozens of encrypted malware traffic. An anomaly detection was performed based on log information extracted from the dataset using a machine learning algorithm, LightGBM-based anomaly detection model. The anomaly detection model performed binary classification with normal data and Malware data for encrypted traffic and derived an accuracy performance 0.9409 and an F1 score 0.9222.

Niu et al. [29] conducted anomaly detection research targeting network traffic encrypted by SSL and TLS communication channels. In this research, a new dataset was created by mixing the MTA dataset [30], MCFP [31], and CTU-13 [32], which are datasets containing cryptographic network traffic. After that, the encrypted traffic data set was inputted into the Zeek IDS to generate conn.log, x509.log, and ssl.log, and the information was used as a feature. In this research, an anomaly detection model based on Improved Adaptive Random Forests (IARF), a machine learning algorithm, was proposed to detect anomalies in the encrypted traffic. The anomaly detection model performed binary classification with normal data and Malware data for encrypted traffic and derived the performance of precision 0.9966, recall 0.9967, and F1 score 0.9966.

Alzighaibi [33] conducted anomaly detection research targeting network traffic encrypted by the hypertext transfer protocol secure (HTTPS) communication channel. This research employed a CIRA-CIC-DoHBrew-2020 [34] dataset derived using DNS-over-HTTPS Analyzer (DoHlyzer), a capture and statistics-based feature extraction tool for HTTP traffic. The dataset contains https traffic generated by multiple applications and HTTPS encryption traffic caused by cyberattacks. The feature of this dataset contains non-encrypted data in HTTPS traffic and statistical numerical values based on it. In this research, binary classification was performed on the stacking algorithm-based anomaly detection model combining Random Forest (RF) and DT using the extracted statistics-based features, and account 0.999 and F1 score 0.999 were derived.

Bahlali et al. [35] conducted anomaly detection research on HTTPS, SSH, and TLS encrypted traffic. This research extracted statistics-based features from the encrypted traffic dataset, including HTTPS, TLS, and secure shell (SSH), using CICFlowmeter [36], an open source for capturing and statistics-based feature extraction for network traffic. This research used UNSW-NB15 [37] and CSE-CIC-IDS2018 [38] datasets with normal and anomaly HTTPS, TLS, and SSH encrypted traffic to evaluate anomaly detection performance. This research devised a fully-supervised autoencoder architecture with custom reconstruction loss (DAE-CRL) to model network traffic effectively, and an anomaly detection model based on the architecture was developed. As a result of the experiment, an average accuracy 0.942 and an average F1 score 0.9482 were derived.

Zhao et al. [39] conducted anomaly detection research to detect malicious code behavior performed in an encrypted communication environment. In this research, statistics-based features were derived using a flow statistical classifier to extract statistical values for network traffic encrypted by TLS. In this research, binary classification was performed on the anomaly detection model based on the ensemble learning algorithm using the extracted statistics-based features, and high cyberattack detection performance, such as TRP of 0.95 and FPR of 0.08, was derived.

Ferriyan et al. [40] proposed a malicious traffic detection technique over TLS-encrypted traffic based on Word2Vec. In this research, packet-level features were derived from TLS-encrypted network traffic. The research derived the TLS/SSL version (version), the list of cipher suites in the client hello

or server hello (cipher), the extension length of the client hello or server hello (ext_len), the elliptic curves of the client hello (elliptic_curves), the point formats of the elliptic curves (ec_point_formats), and the payload length of each packet (len), in words, that can be identified from the network packets generated during the TLS handshake. Each word was tokenized by applying Word2Vec technology to the derived words, and the tokens were input to malicious traffic classification models based on LSTM and BiLSTM algorithms to detect malicious traffic. To validate the method proposed in this research used CTU-Malware-Capture and Jason Stroschein's public GitHub malware dataset, which contains malicious encrypted network packets caused by Zeus, Cobalt, and Trickbot, and detected an average classification performance F1 score 0.891.

Although various types of cyberattack detection research are conducted on cryptographic traffic, consideration of the IoT environment and generated cyberattacks is not included, so it is unclear whether it applies to the IoT environment. As the application of encrypted communication for cybersecurity is increasing in the IoT environment, there is a possibility that a cyberattack performed hidden in encrypted communication may occur. For this reason, research for cyberattack detection over encrypted traffic occurring in IoT environments should be conducted.

Table 2 provides an overview of the contributions and limitations of related work on AI-Based Cyberattack Detection Technology for IoT and Cyberattack Detection Technology over Encrypted Traffic.

Table 2: Summary of related research

Objective	Ref.	Contribution	Limitation
Detect cyberattacks in IoT environments	[18]	• Proposed framework for detecting cyberattacks on IoT based on data type, payload, protocol, and port information	• Unable to detect new types of cyberattacks • Unable to detect cyberattacks on encrypted traffic
	[20]	• Payload-based cyberattack detection method that can be deployed on machine learning-based IoT edge devices	• Unable to detect cyberattacks on encrypted traffic
	[22]	• Proposed a cyberattack detection system for IoT using statistical-based information, including payload information based on machine learning algorithm	• Unable to detect new types of cyberattacks • Unable to detect cyberattacks on encrypted traffic

(Continued)

Table 2 (continued)

Objective	Ref.	Contribution	Limitation
	[24]	• Proposed method for detection technique for cyberattacks and malware in IoT based on VGG-16 and VGG-19 using payload information	• Unable to detect new types of cyberattacks • Unable to detect cyberattacks on encrypted traffic
Detect cyberattacks carried out via encrypted traffic	[26]	• Proposed a log information-based feature extraction method and LightGBM-based cyberattack detection system over encrypted traffic	• Unable to detect new types of cyberattacks
	[29]	• Proposed a log information-based feature extraction method and IARF-based cyberattack detection system over encrypted traffic	• Unable to detect new types of cyberattacks
	[33]	• Proposed a cyber-attack detection system over encrypted traffic based on a statistical-based encrypted traffic feature extraction method using DoHlyzer and a stacking algorithm	• Difficulty detecting sophisticated payload forgery attacks
	[35]	• Proposed a cyber-attack detection system over encrypted traffic based on a statistical-based encrypted traffic feature extraction method using CICFlowmeter and a DAE-CRL flamework	• Difficulty detecting sophisticated payload forgery attacks

(Continued)

Table 2 (continued)

Objective	Ref.	Contribution	Limitation
	[39]	• Proposed flow statistical classifier for statistical feature extraction for TLS network traffic • Propose a framework for cyberattack detection over encrypted traffic utilizing statistical features based on ensemble learning	• Difficulty detecting sophisticated payload forgery attacks • Unable to detect new types of cyberattacks
	[40]	• Proposed a packet-level feature extraction scheme and Word2Vec-based word embedding scheme for TLS network traffic • Proposed a malicious traffic detection method over encrypted traffic using LSTM and BiLSTM-based packet-level features	• Difficulty detecting sophisticated payload forgery attacks

3 ECDS-IoT (Encrypted Cyberattack Detection System for IoT) Structure

Section 3 describes a cyberattack detection system targeting encrypted traffic in an AI-based IoT environment. In some fields where IoT is applied, such as industrial control systems (ICS) and weapon systems, data availability is critical because a single command or network packet can play an essential role in operation. If a firewall or intrusion prevention system (IPS) is applied to such an environment, normal data can be falsely detected as cyberattack data, and the data may be blocked, significantly negatively impacting operations. For this reason, it is necessary to detect cyberattacks on IoT and enable managers to take appropriate action. The proposed technique detects cyberattacks occurring in IoT environments based on learning information about normal encryption traffic occurring in IoT communication environments. As the frequency of cyberattacks on the IoT environment increases and cyberattacks performed hidden in encrypted communication occur due to the application of encrypted communication, it is necessary to apply a system to detect encrypted cyberattacks. Therefore, developing and applying an Encrypted Cyberattack Detection System for IoT (ECDS-IoT) that extracts valid features from encrypted traffic and detects cyberattacks based on them is necessary. Section 3.1 provides an overview of the detection of cyberattacks targeting the encryption traffic in the IoT environment. Section 3.2 describes extracting essential information for learning AI models and anomaly detection from encryption traffic. Section 3.3 describes feature preprocessing for features extracted from encrypted traffic to increase cyberattack detection efficiency and performance. Section 3.4 introduces how to detect encrypted traffic caused by cyberattacks in encrypted traffic that occur in a normal state.

3.1 Overview

Fig. 2 shows the overview of ECDS-IoT, which detects cyberattacks over encrypted traffic generated in IoT environments based on the AI proposed in this research. ECDS-IoT consists of statistics-based feature extraction, feature preprocessing, and cyberattack detection steps to detect encrypted traffic in IoT environments. ECDS-IoT extracts statistics-based features of metadata and traffic in addition to key data that are encrypted and unidentifiable within encrypted traffic collected in IoT environments. After that, the input features are preprocessed to increase the learning and anomaly detection accuracy and efficiency of the AI-based cyberattack detection model. The preprocessed features are entered into the cyberattack detector based on AI algorithms. The cyberattack detector classifies the input encrypted traffic data as normal and cyberattack data.

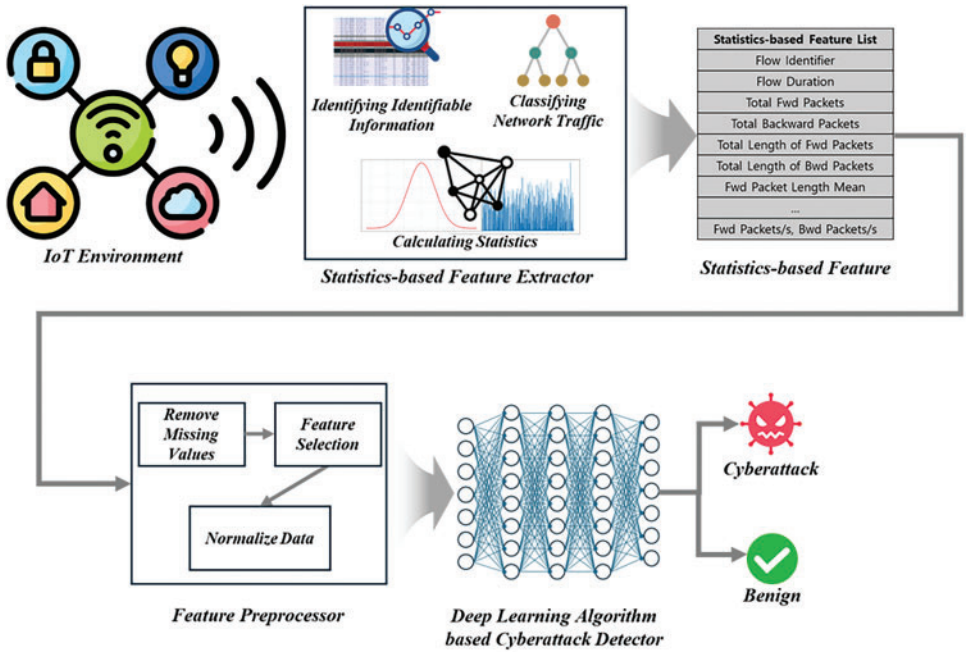


Figure 2: Overview of encrypted cyberattack detection system for IoT (ECDS-IoT)

3.2 Statistics-Based Feature Extraction

Statistics-based feature extraction refers to a statistical method used to summarize and describe the main features and patterns of data. Unlike non-encrypted traffic, encrypted traffic does not contain information that can be intuitively identified without decryption. However, due to the general network environment in which the key employed to encrypt data is unknown, cyberattack detection must rely on identifiable information and statistical features in network traffic without decryption. In the research on cyberattack detection over encrypted traffic, there are methods for extracting logs from encrypted traffic and detecting cyberattacks based on them. In addition, there is a method to derive packet-level features, which are unencrypted information in encrypted packets, and detect cyberattacks based on them. Finally, methods exist to extract statistical information from encrypted network traffic as features and detect cyberattacks. The method of cyberattack detection through log collection requires a separate device to collect logs, and it may be challenging to derive meaningful features depending on the log generation criteria. The packet-level feature-based cyberattack detection method may have a very low detection rate for some cyberattacks because there is very little identifiable information in

encrypted packets, making it difficult to construct a meaningful feature set. For this reason, a statistics-based feature is required based on identifiable information, excluding encrypted data within the encrypted network traffic. These extracted statistics-based features encapsulate the behavior patterns of encrypted traffic flows, facilitating model training and validation for anomaly detection without compromising encryption, which is a powerful approach for network security in encrypted packet environments. The procedure for deriving these statistics-based features is as follows: First, statistics-based feature extractors collect raw network traffic data, including encrypted network traffic, and identify identifiable information such as source IP, destination IP, source port, destination port, port number, and protocol. The network traffic is then classified based on the information identified earlier. After that, basic data correction tasks such as duplicate packet removal, timestamp-based alignment, and packet filtering are performed on the classified network traffic. Statistical analysis then generates features such as flow duration (total and average), packet count (total and average), byte count (total and average), packet size statistics (average, standard deviation, minimum, maximum), time statistics between arrivals (average, standard deviation), protocol distribution (e.g., ratio of TLS, SSH), and port distribution (emphasizing source and destination port usage patterns). Since these characteristics are not information in the packet payload, they do not provide direct evidence of cyberattacks. However, these statistics-based features can be used to train or verify anomaly detection models by providing abstract information about network flows. The method of extracting some statistics-based features from encrypted network traffic is shown in Algorithm 1.

Algorithm 1: Statistics-based feature extraction

Input:

P: Set of $\boldsymbol{k}$ network packets. Each packet p_t includes {*source_IP*, *destination_IP*, *source_port*, *destination_port*, *port_number*, *protocol*, *packet_size*, *timestamp*}

Output:

F: Dictionary containing sets of statistical features for each of the $\boldsymbol{n}$ flows, where key is the flow identifier f, and value is the statistical features ϕ_f of that flow.

Step 1. Initialization

$F \leftarrow$ *empty set*;

Step 2. Packet Classification

For $t = 1$ to k **do**

$f_{id} \leftarrow (p_t.source\_IP,\ p_t.destination\_IP,\ p_t.source\_port,\ p_t.destination\_port,\ p_t.protocol)$;

If $f_{id} \notin F$:

$F_{f_{id}} \leftarrow \{packet_{size} \leftarrow$ empty set, $timestamp \leftarrow emptyset, count = 0\}$;

else

$F_{f_{id}}.packet\_size \cup p_t.packet\_size$;

$F_{f_{id}}.timestamp \cup p_t.timestamp$;

$F_{f_{id}}.count \leftarrow F_{f_{id}}.count + 1$;

Step 3. Statistical Calculation

For $j = 1$ to n **do**

$\tau \leftarrow \{\tau_z | \tau_z \leftarrow F_j.timestamp_z, 1 \le z \le |F_j.timestamp_z|\}$ *with* $\tau_z \le \tau_z + 1$;

$D_j \leftarrow \tau_{last} - \tau_{first}$;

$N_j \leftarrow F_j.count$;

$B_j \leftarrow \sum(F_j.packet\_size)$;

(Continued)

Algorithm 1 (continued)

$\lambda_j \leftarrow \frac{D_j}{N_j}$;

$\mu_{size} \leftarrow \frac{1}{N_j}\sum(F_j.packet\_size)$;

$\sigma^2{}_{size} \leftarrow \frac{1}{N_j}\sum((F_j.packet\_size - \mu_{size})^2)$;

$IAT_j \leftarrow [\tau_j - \tau_{j-1}]$;

$\mu_{IAT} \leftarrow \frac{1}{N_j}\sum(IAT_j)$;

$\sigma_{IAT} \leftarrow \frac{1}{N_j}\sum(IAT_j - \mu_{IAT})^2$;

$\phi_j \leftarrow (D_j, N_j, B_j, \lambda_j, \mu_{size}, \sigma^2{}_{size}, \mu_{IAT}, \sigma_{IAT})$;

$F_j \leftarrow \phi_j$;

Step 4. Return Results

Return F

As can be seen in Algorithm 1, the collected network packets are classified based on the f_{id} (*source_IP*, *destination_IP*, *source_port*, *destination_port*, *port_number*, *protocol*) of each packet to generate each packet set, $F_{f_{id}}$. Thereafter, the total *packet_size*, *timestamp*, and the number of packets of $F_{f_{id}}$ are calculated. To extract the statistics-based features for each $F_{f_{id}}$, τ, which is an ascending order sorting result for the timestamp of $F_{f_{id}}$ is derived. IAT, which is the transmission time of a packet, and D, which is the connection duration of $F_{f_{id}}$, are derived. And N, which is the total number of packets of $F_{f_{id}}$, and B, which is the sum of the total packet sizes of $F_{f_{id}}$, is derived. And λ, which is the number of packets per unit time of $F_{f_{id}}$, is derived. Thereafter, based on the previously derived information, statistics-based information such as μ_{size}, which is the average of the packet size in $F_{f_{id}}$, $\sigma^2{}_{size}$, which is the variance of the packet size in $F_{f_{id}}$, and μ_{IAT}, which is the IAT average of $F_{f_{id}}$, and σ_{IAT}, which is the IAT variance of $F_{f_{id}}$ is derived. Based on this statistics-based information derivation method, more statistics-based information can be derived by applying it to the forward packet set and the reverse packet set, and based on this information, statistics-based feature extraction is possible.

3.3 Feature Preprocessing

Data preprocessing is applied to clean data, normalize data, and filter a subset of features. The feature preprocessing step is very important because noise in the data can degrade performance [41]. Most datasets contain noise data and missing data and may contain data that may adversely affect model training and cyberattack detection. In addition, preprocessing should be performed because only specific data types can be input to learn an anomaly detection model and anomaly detection. Data preprocessing enables AI models to train data efficiently and detect cyberattacks. Data preprocessing methods include the removal of missing values, feature selection, and normalization.

Feature selection is an essential process for identifying and selecting the most relevant and informative features from the set of available features. Using Feature Selection can improve the efficiency and effectiveness of machine learning and deep learning by reducing computational complexity and the risk of overfitting. It also improves the interpretability of the model, making it easier to understand and explain why certain features contribute to the detection of cyberattacks. Finally, it helps alleviate the curse of dimensionality that can arise from work with high-dimensional data [42].

Removal of missing values refers to a method of filling in missing values in data. For missing values, the sample size of the data becomes smaller than intended, and consequently, the reliability of the research results is impaired. In addition, biased results can be generated when deducing populations based on these samples [43], which can compromise the reliability of the data. For this reason, removing missing values is an essential process of feature preprocessing. There is a method of replacing missing values with the average of data categories in which missing values exist. In addition, there are methods of replacing missing values with random values and replacing values derived by predicting missing values with missing values [44].

Normalization means standardizing different data categories of each feature within a dataset. The different data categories of each feature in the dataset can cause a decrease in the performance of learning and cyberattack detection of the AI model and a decrease in efficiency, so normalization is an essential step. As a type of normalization, there is min-max normalization, which converts the range of features to a consistent range between 0 and 1. In addition, z-score normalization converts the range of different features to a consistent range between −1 and 1.

3.4 Cyberattack Detection

The cyberattack detector detects a cyberattack based on information identifiable in encryption traffic and statistical analysis information about it. The cyberattack detector learns only the statistics-based feature extracted from normal encryption traffic. The learned cyberattack detector performs cyberattack detection for encryption traffic based on normal encryption traffic information. The supervised learning-based anomaly detection system has the advantage of being able to classify attack types within an attack and a relatively high anomaly detection rate. However, it is very difficult to collect datasets to be used for learning by performing actual cyberattacks on most IT infrastructure environments, including IoT environments, to develop a cyberattack detection system [45]. In addition, the supervised learning-based cyberattack detection system has the disadvantage of being unable to detect and classify new cyberattacks other than the learned cyberattack type. For this reason, cyberattack detectors use a method of learning only normal data collected in the IoT environment. The cyberattack detector learns to derive the smallest error possible for normal data. If such cyberattack data is input to the cyberattack detector, a relatively large error value is derived by the statistics-based feature information of the cyberattack data different from the normal data, and if the derived error value is greater than the set Threshold, the data is detected as cyberattack data.

4 Evaluation

Section 4 evaluates the ECDS-IoT proposed in Section 3 using a dataset containing cryptographic traffic collected in the IoT environment. Section 4.1 describes the dataset containing encrypted traffic collected from the IoT environment used in the experiment, the feature extraction method for encrypted traffic, and feature preprocessing. Section 4.2 presents the experimental environment and model structure. Section 4.3 describes various evaluation indicators to evaluate the performance of ECDS-IoT. Section 4.4 verifies the cyberattack detection performance of ECDS-IoT. In addition, this section presents answers to the following research questions (RQ):

- (RQ 1) How does ECDS-IoT understand and learn normal data flow in encrypted traffic?
- (RQ 2) How can ECDS-IoT classify encrypted traffic into normal traffic and traffic from cyberattacks?
- (RQ 3) Does ECDS-IoT maintain its detection performance compared to the cyberattack detection model targeting non-encrypted traffic that occurs in the existing IoT environment?

4.1 Dataset Description

The proposed ECDS-IoT was learned and validated in this research using CICIoT2023 [12]. CICIoT2023 consists of a normal network packet collected in an IoT environment consisting of 105 IoT devices that communicate based on Z-wave, Zigbee, and Wi-Fi, and a network packet caused by cyberattacks of seven types (DDoS, DoS, Recon, Web-based, Brute Force, Spoofing, and Mirai). This dataset includes encryption network packets generated by TLS 1.2 based encryption communication. ECDS-IoT performs the statistics-based feature extraction presented in Section 3.2 to derive a feature for detecting a cyberattack in the encrypted traffic target. ECDS-IoT used a CICFlowmeter for statistics-based feature extraction. CICFlowMeter is an open source designed to extract statistics-based features from raw packet data [36]. CICFlowMeter captures statistics for different traffic flows, including packet count, byte count, duration, and packet transmission time statistics [46]. CICFlowmeter derives the data size distribution within the encrypted flow by calculating statistics such as mean, median, and standard deviation for the previous function. In addition, it generates features such as average packet length, packet length change, and entropy by considering the packet length distribution. CICFlowmeter generated 82 features in total. Missing values were replaced with zeros to remove missing values for the 82 features generated by the CICFlowmeter. These features include statistical and metadata-based features such as IP addresses, MAC addresses, and port numbers. These features were excluded from the dataset because it was limited to the environment from which the dataset was extracted and had a negative aspect in that it considered the generalization of the system. In addition, non-correlated features were excluded from the dataset by performing a correlation analysis between features. Fig. 3 shows 66 features in the data range used in this experiment after feature selection among the extracted statistical-based features.

Fig. 3 indicates that the statistics-based feature derived through statistical analysis based on information that can identify encrypted traffic using CICFlowmeter has various data categories. Different data categories for each feature are likely to have a negative effect, such as increasing the amount of computation and blurring the importance of each feature when learning and detecting anomalies in the cyberattack detection model. To solve this problem, the feature preprocessor used Max-Abs scaling to unify the range of data between −1 and 1 for all features. The equation for deriving the Max-Abs calculated value $x_i^{'}$ for feature X with n data is the same as Eq. (1) [47]:

$$x_i^{'} = \frac{x_i}{max(|x_1|, |x_2|, |x_3|, \ldots, |x_n|)} \tag{1}$$

In Eq. (1), x_i means all values of X, $|x_i|$ means the absolute value of x_i, and is the of $max(|x_1|, |x_2|, |x_3|, \ldots, |x_n|)$ means the largest value of x_i.

ECDS-IoT used 19,331 train data (normal data: 19,331) and 14,578 test data (normal: 10,119, backdoor_Malware: 498, browserhijacking: 500, commandinjection: 500, DDoS-SlowLoris: 500, DDoS-SlowLoris: 500, DictionalyBruteForce: 500, DNS_Spoofing: 500, Uploading_attack: 500, sqlinjection: 501, XSS: 460) for learning and validation. For the verification and testing of ECDS-IoT, cyberattack data was inserted by selecting some sections within the normal data section.

4.2 Experimental Setup

In this section, we describe the experimental environment used to experiment with the performance of the proposed ECDS-IoT.

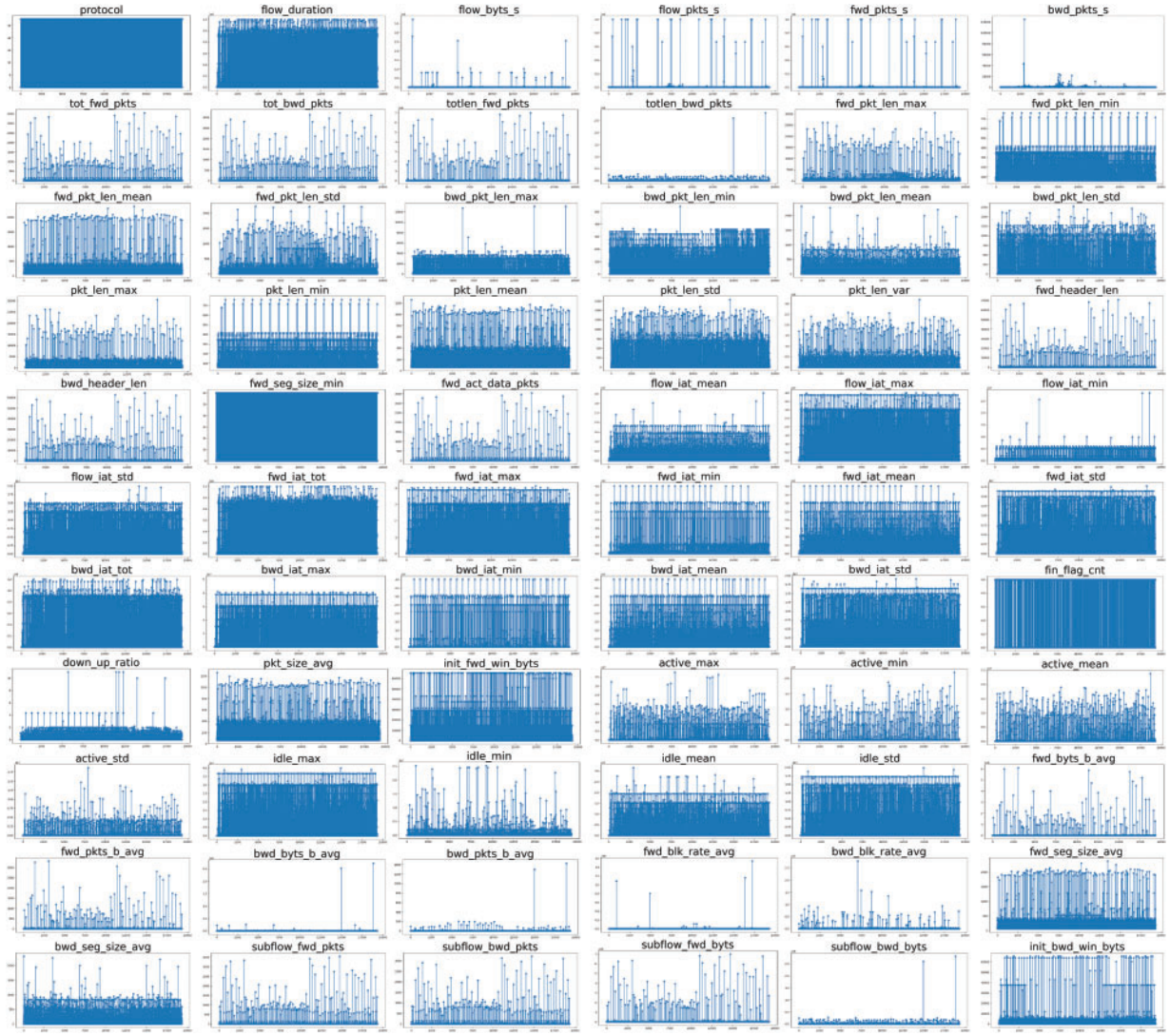


Figure 3: Visualization result of observation data of each feature

Computational environment. This experiment used the Ubuntu 20.04.6 LTS operating system and was conducted in the Intel(R) Xeon(R) Silver 4216 CPU @ 2.10 GHz, 64 GB RAM, Tesla V100S PCIe 32 GB GPU, and python3 development environment.

Hyper-parameter settings. In this experiment, this study used autoencoder, recurrent neural network (RNN), gated recurrent unit (GRU), long short-term memory (LSTM), bidirectional LSTM (BiLSTM), and LSTM-based autoencoder (AE-LSTM) algorithms to implement a cyberattack detector in ECDS-IoT that detects cyberattacks based only on information from learned normal data. Autoencoder was designed as an encoder and decoder consisting of a dense layer with 66,64,32,16,8 nodes, respectively. The RNN model was designed as an RNN layer with 66,64,32 nodes and a dense layer with 32 nodes. The GRU model was designed as 3 GRU layers with 66 nodes and a dense layer with 66 nodes. The LSTM model was designed as 3 LSTM layers with 66 nodes and dense layers with 66 nodes. The BiLSTM model was designed as 1 BiLSTM layer with 66 nodes and a dense layer with

66 nodes. AE-LSTM was designed with an Encoder and Decoder consisting of LSTM layers with 66, 64, and 32 nodes, respectively. The Autoencoder used ReLU as the activation function of each layer and tanh as the layer's activation function for the rest of the models. This study used Adam as the optimizer for all models and applied a learning rate of 0.001.

4.3 Assessment Indicators

In this experiment, accuracy(2), recall(3), precision(4), F1 score(5), and receiver operating characteristic area under the curve (ROC-AUC) were used as performance measurement indicators of the anomaly detection model. Accuracy defines the ratio of the number of correctly classified data to the total number of data. It is an evaluation index for whether the detection system correctly classifies normal data and anomaly data as anomaly data in the collected data. Recall is the ratio of the number of anomalies divided by the total number of intrusions, and it is an evaluation index that evaluates how well the data to be detected was found. Precision is an evaluation index that evaluates the ratio of actual anomaly data among data predicted as anomaly data in the system. The F1 score is an evaluation index employed to measure the model's overall accuracy. ROC_AUC is an evaluation index representing the rate of change of TPR and FPR. The evaluation indicators are calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$F1\ score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{5}$$

4.4 ECDS-IoT Performance Evaluation

This section evaluates the detection performance for each cyberattack detection algorithm presented in Section 4.2. All algorithms learned only the data derived from normal encrypted traffic, and the cyberattack detection performance was verified using the test data set. Fig. 4 shows the learning and validation loss of each cyberattack detection model. Each model was learned by dividing 19,331 normal data (train data) into train data and validation data. Train data and validation data were split at a 9:1 ratio. Each model trains train data every epoch and derives the validation loss of the model based on validation data. Each model was trained in a way that reduces validation loss by modifying the weight of the deep learning node. The epoch of all models was set to 1000, and early stopping was applied to stop model training at the point when validation loss no longer decreased.

In Fig. 4, the blue line represents train loss, and the orange line depicts validation loss. Fig. 4 indicates that all models were trained to have a validation loss close to zero. The validation loss of 0.0015 for the Autoencoder model, 0.0009108 for the RNN model, 0.0006489 for the GRU model, 0.0005192 for the LSTM model, 0.0009203 for the BiLSTM model, and 0.0000271 for the AE-LSTM model.

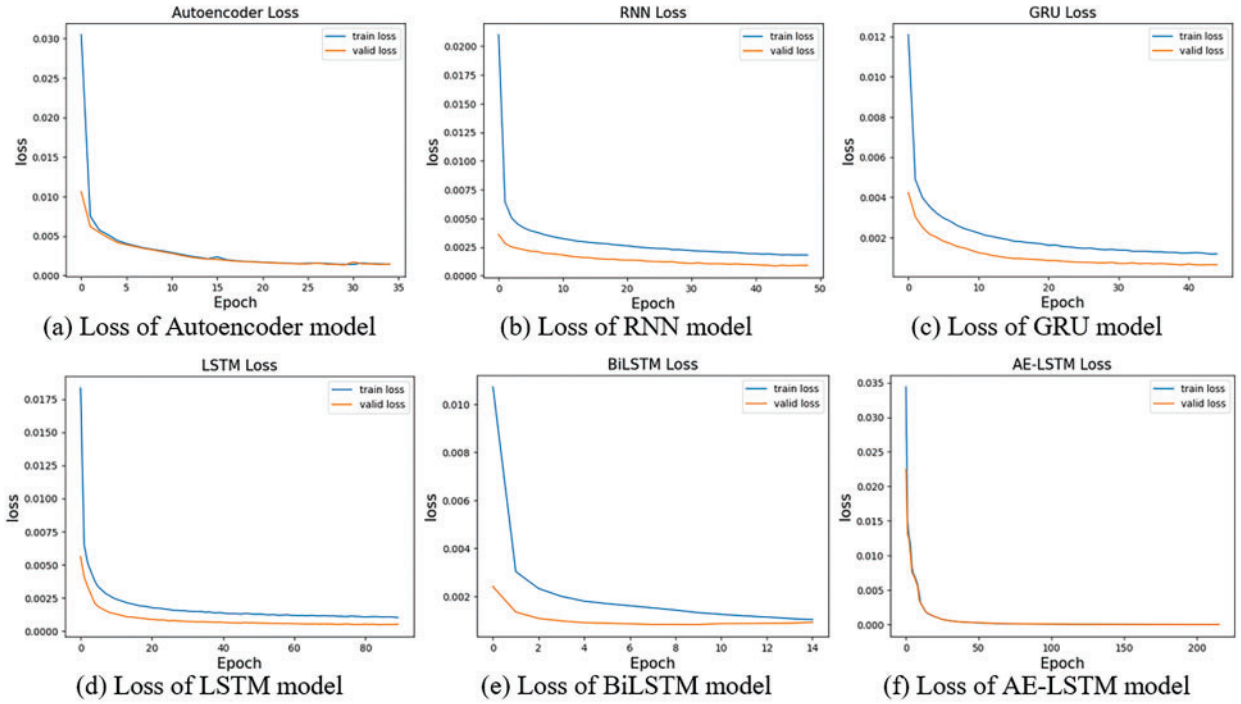


Figure 4: Loss of each cyberattack detection model

Table 3 explains how ECDS-IoT understands and trains about the flow of encrypted traffic.

Table 3: Answers to research question 1

RQ 1	How does ECDS-IoT understand and learn the flow of normal data in encrypted traffic?
	As mentioned in Section 4.1, this study derived a statistics-based feature for the encrypted traffic dataset. It derived identifiable features with or without packet encryption, such as the number of transmission packets per hour, the maximum transmission time of the packet, and the maximum active time before the flow went idle. The packet's length is changed by being padded by performing packet encryption, but the degree of padding is different for each type of packet, and thus, the corresponding information can be used as a weak characteristic of the packet. For this reason, this research derived features such as maximum packet length, minimum packet length, and packet length average. Each model for cyberattack detection understands only the data flow and characteristics of normal data by learning the range of data, trends, and correlation information for each feature for normal traffic in the dataset shown in Fig. 3. The cyberattack detection model learned in this way detects data that differs more than a certain level from the information of the learned normal data as cyberattack data.

Figs. 5–10 are the results of cyberattack detection on test datasets containing cryptographic traffic using Autoencoder, RNN, GRU, LSTM, BiLSTM, and AE-LSTM-based ECDS-IoT, respectively.

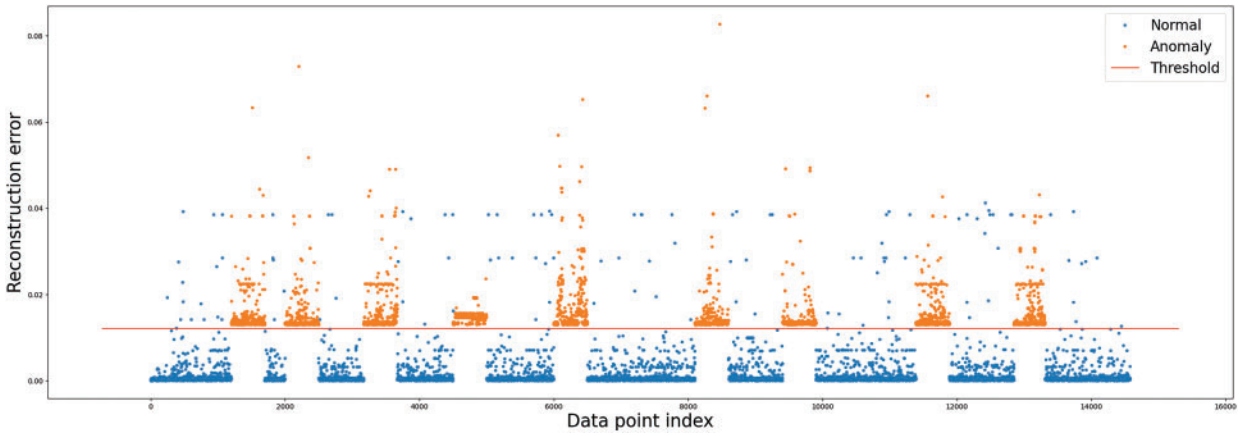


Figure 5: ECDS-IoT (Autoencoder) cyberattack detection

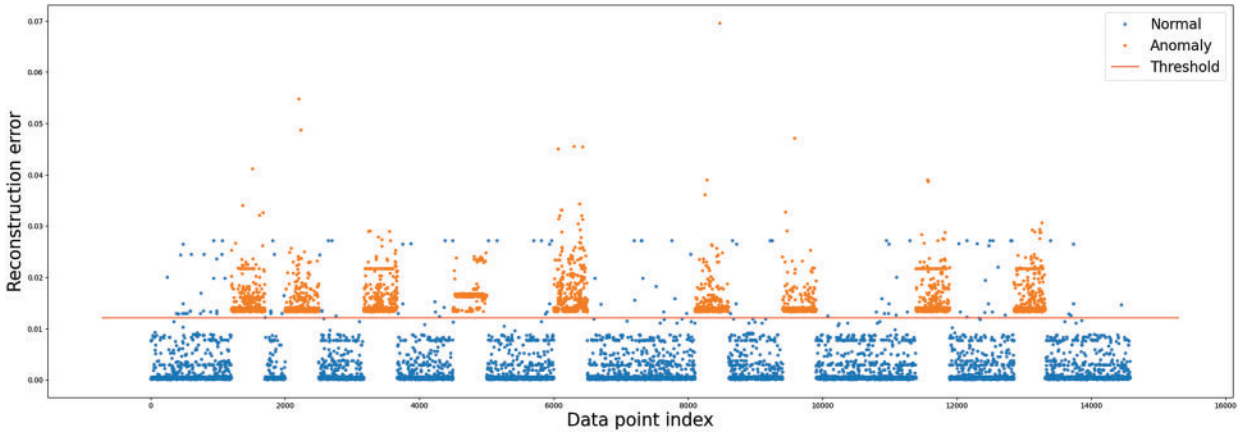


Figure 6: ECDS-IoT (RNN) cyberattack detection

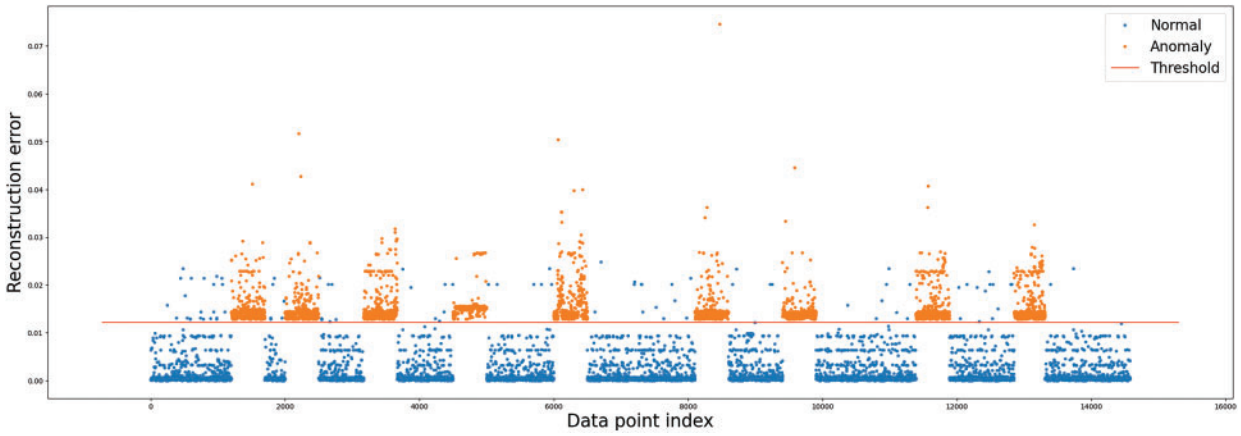


Figure 7: ECDS-IoT (GRU) cyberattack detection

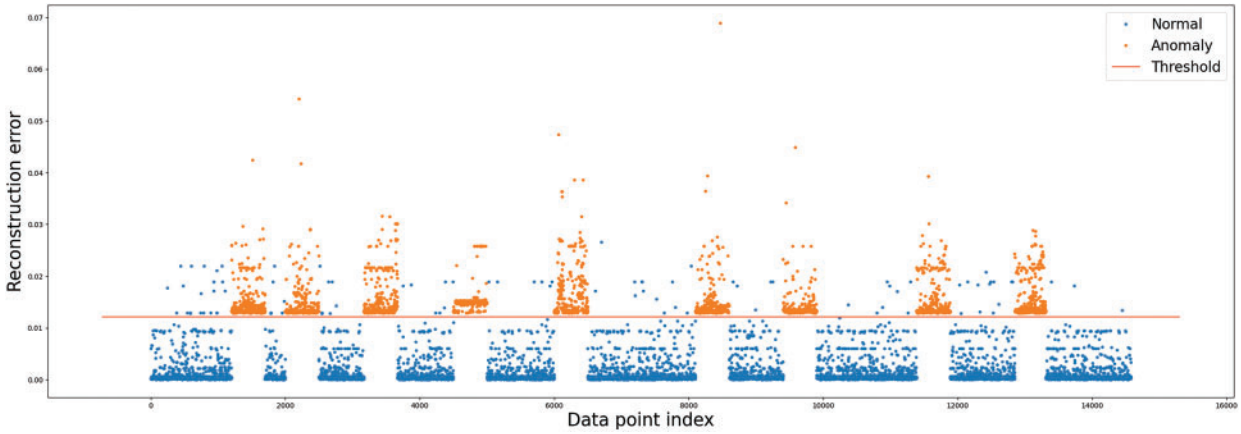


Figure 8: ECDS-IoT (LSTM) cyberattack detection

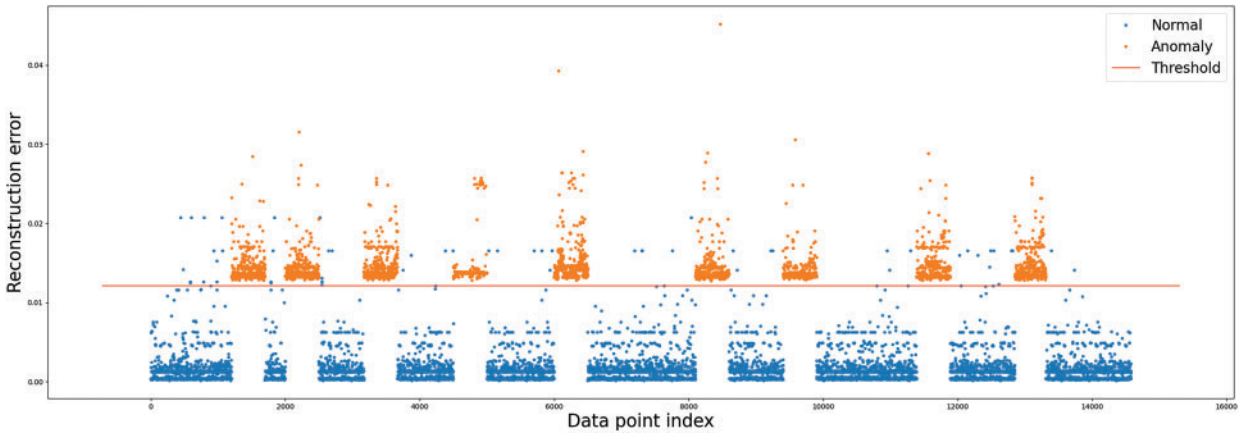


Figure 9: ECDS-IoT (BiLSTM) cyberattack detection

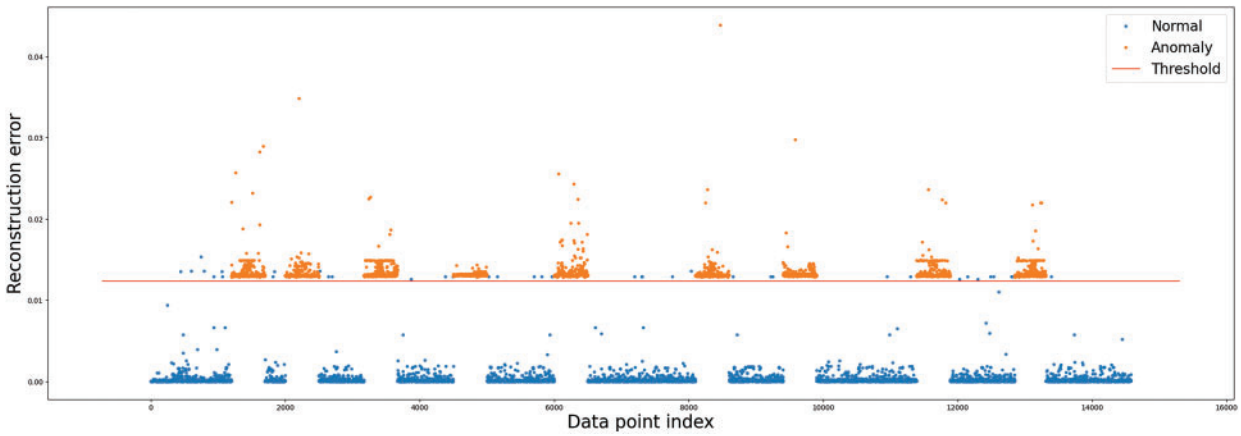


Figure 10: ECDS-IoT (AE-LSTM) cyberattack detection

The blue point in Figs. 5–10 represents normal data, and the orange point depicts cyberattack data. As can be seen in Figs. 5–10, the detection results of all cyberattack detectors show a large difference so that normal data (blue point) and cyberattack data (orange point) can be identified. The red straight line represents a threshold for anomaly behavior detection. Each figure shows Autoencoder, RNN, GRU, LSTM, BiLSTM model, and AE-LSTM model classified normal data (blue point) and cyberattacks (orange point) based on threshold values. The threshold value was set to the value when the precision and recall of the anomaly detection model were the same. High precision means that false detections of normal data are low, and high recall means that false detections of anomaly data are low [48]. Since these two performance indicators are usually in a trade-off relationship, in this research, we derived thresholds in the same way as above to consider both performances without bias [49]. ECDS-IoT detects data as cyberattack data when the error between the expected and actual output values is higher than the Threshold for each model.

Figs. 5–10 show that the autoencoder, RNN, GRU, LSTM, and BiLSTM models have more false positives that misclassified normal data as cyberattack data based on Threshold than AE-LSTM models. This means that the AE-LSTM model learned the characteristics of normal data better than other models. As a result of the detection of cyberattack data, it can be confirmed that Autoencoder, RNN, GRU, LSTM, BiLSTM, and AE-LSTM models all detected all cyberattack data based on Threshold. This means all models are well-trained to derive small error values for normal data, and large error values are derived for unlearned cyberattack data. The low number of detections that fail to detect cyberattacks in cyberattack detection is a huge advantage. Table 4 shows the cyberattack detection performance of Autoencoder, RNN, GRU, LSTM, BiLSTM, and AE-LSTM.

Table 4: Cyberattack detection performance metrics for each algorithm

Model algorithm	Accuracy	Precision	Recall	F1 score	ROC_AUC
Autoencoder	0.99259	0.97635	**1.0**	0.98803	0.99466
RNN	0.99224	0.97528	**1.0**	0.98748	0.99441
GRU	0.99368	0.97978	**1.0**	0.98978	0.99545
LSTM	0.99375	0.98	**1.0**	0.98989	0.99550
BiLSTM	0.99588	0.98672	**1.0**	0.99331	0.99703
AE-LSTM	**0.99739**	**0.99154**	**1.0**	**0.99575**	**0.99812**

Fig. 11 is a visualization of the performance of each algorithm by performance indicator to compare the cyberattack detection performance of Autoencoder, RNN, GRU, LSTM, BiLSTM, and AE-LSTM.

Fig. 12 visually shows the results of the ROC_AUC performance indicator of each cyberattack detection model.

Table 4 and Figs. 11 and 12 indicate that the AE-LSTM model derived the highest performance on all evaluation metrics. All models derived performance of 0.99924 accuracy, 0.98672 precision, 0.98748 F1 score, and 0.99441 ROC_AUC, and all models derived recall 1.0, confirming that all encrypted traffic caused by cyberattacks was detected.

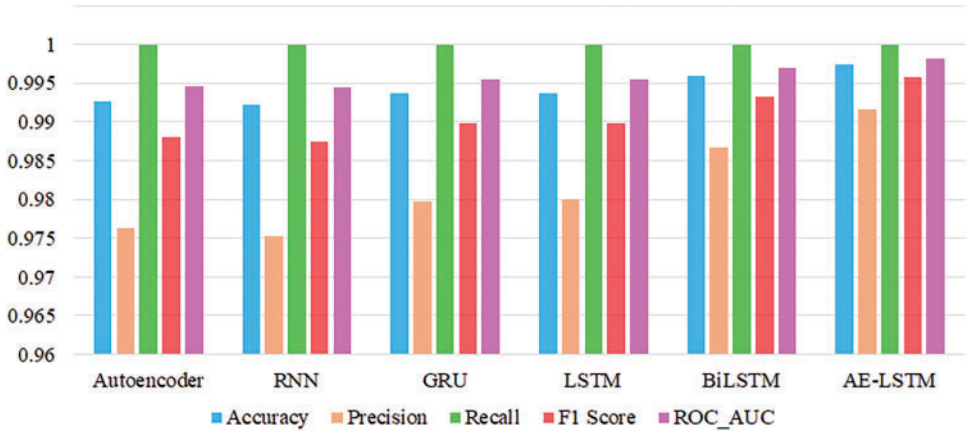
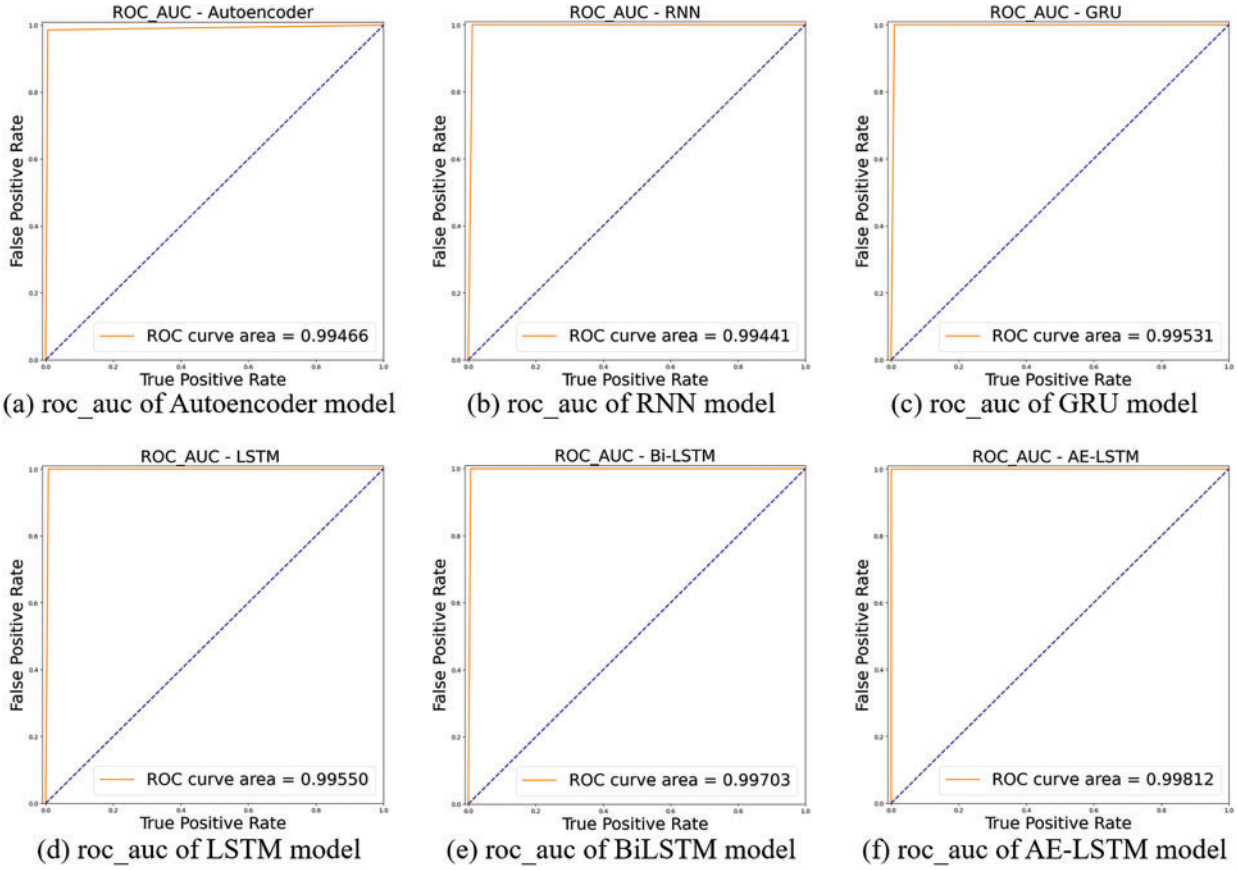


Figure 11: Cyberattack detection performance comparison



(a) roc_auc of Autoencoder model (b) roc_auc of RNN model (c) roc_auc of GRU model

(d) roc_auc of LSTM model (e) roc_auc of BiLSTM model (f) roc_auc of AE-LSTM model

Figure 12: ROC_AUC of each cyberattack detection model

ECDS-IoT is learned to minimize the error of the predicted output value and the actual output value based on information on learned normal data, such as the range of data by feature, correlation by feature, and flow of normal data. If cyberattack data with characteristics different from normal data are entered into these models, a relatively large error is derived and detected as a cyberattack.

Fig. 13 compares the normal distribution of features in which the normal distribution of statistics-based features extracted from normal traffic and the normal distribution of statistics-based features extracted from encrypted traffic caused by cyberattacks are similar. Fig. 14 compares the normal distribution of features in which the normal distribution of statistics-based features extracted from normal traffic and the normal distribution of statistics-based features extracted from encrypted traffic caused by cyberattacks are very different.

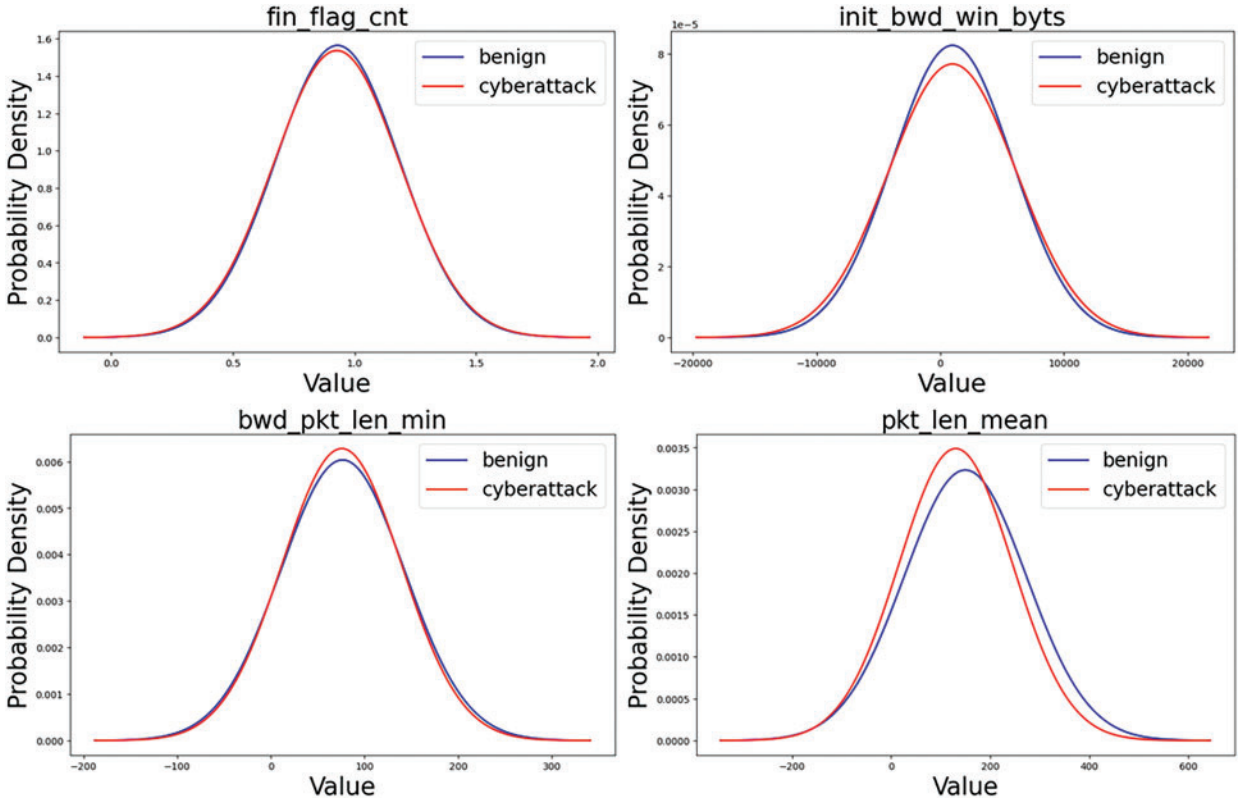


Figure 13: Features that match the benign normal distribution and the cyberattack normal distribution (partial)

The blue line in Figs. 13 and 14 represents the normal distribution of normal data features, and the red line represents the normal distribution of attack data features. As can be seen in Fig. 13, among the statistics-based features extracted based on statistical analysis in encrypted traffic for the learning and verification of ECDS-IoT, there is a characteristic that the range of feature values of normal data and cyberattack data is similar. On the other hand, as shown in Fig. 14, there are features with different averages of the normal distribution of feature values and features with significantly different probability densities.

Table 5 explains how ECDS-IoT classifies normal traffic and cyber attack traffic based on the flow information of encrypted traffic.

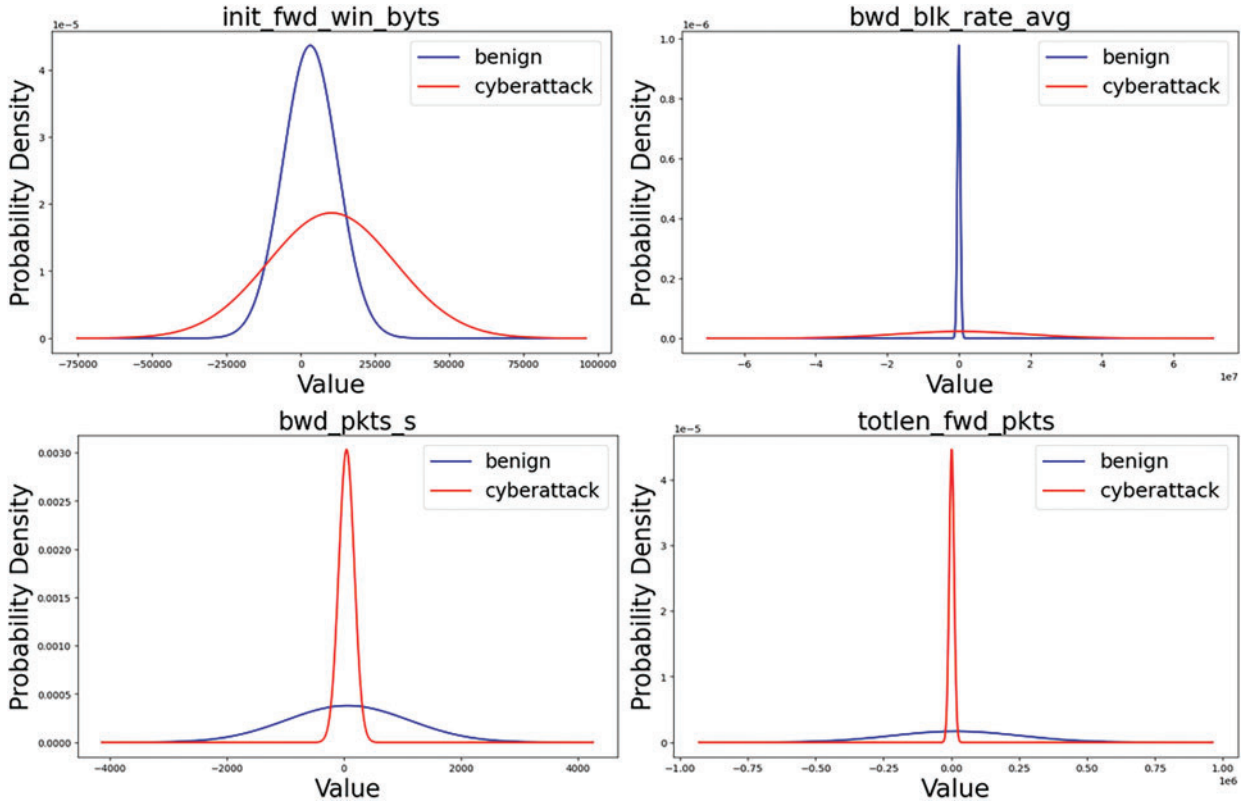


Figure 14: Features that do not match the benign normal distribution and the cyberattack normal distribution (partial)

Table 5: Answers to research question 2

RQ 2	How can ECDS-IoT classify encrypted traffic into normal traffic and traffic from cyberattacks?
	As shown in Figs. 13 and 14, the categories and flows of normal and cyberattack data features in the dataset vary. Some features of normal and cyberattack data show the same data pattern, while others do not. The results of comparing normal distributions with only one feature are not available as a basis for cyberattack detection, but several features with different normal distributions can be used as a basis for cyberattack detection. Since the correlation between features also affects cyberattack detection, different normal distributions of data in each feature can be used as a basis for cyberattack detection.

4.5 Comparative Study

In this section, we compare and analyze ECDS-IoT and other research. Since there is no existing research on the detection of cyberattacks targeting the IoT environment, this study compares ECDS-IoT with research of IoT cyberattack detection or anomaly detection over encrypted traffic. It compares cyberattack detection algorithms, detection performance (accuracy, F1 score), whether

cyberattacks targeting IoT are detected, and whether cyberattacks targeting encrypted traffic are detected. Table 6 compares the existing research and the proposed model.

Table 6: Comparison of existing models, including ours

Reference	Method	Accuracy	F1 score	IoT cyberattack detection	Detect cyberattacks on encrypted traffic
Liu et al. [20]	CNN-LSTM	0.9987	0.9992	O	X
Alanazi et al. [22]	Decision Tree	0.9958	0.9959	O	X
Tomar et al. [24]	VGG-19	0.99	–	O	X
Chao [26]	LightGBM	0.9409	0.9222	X	O
Alzighaibi [33]	Stacking	0.999	0.999	X	O
Bahlali et al. [35]	Deep autoencoder	0.9730	0.9732	X	O
Ferriyan et al. [40]	LSTM	–	0.891 (Average)	X	O
Ours (ECDS-IoT)	**AE-LSTM**	0.9973	0.9957	O	O

This research proposed ECDS-IoT to detect cyberattacks in encrypted traffic in IoT environments. There is no related work on detecting cyberattacks over encrypted traffic in IoT environments. For this reason, we compared the proposed ECDS-IoT to research for IoT Cyberattack detection (CNN-LSTM model [20], Decision Tree model [22], and VGG-16 model [24]), and research for detecting cyberattacks on encrypted traffic (LightGBM model [26], Stacking model [33], Deep Autoencoder [35] and LSTM [40]), as shown in Table 6.

Research related to IoT Cyberattack detection (CNN-LSTM model [18], Decision Tree model [20], and VGG-16 model [22]) has derived high cyberattack detection rates, such as 0.9987 accuracy and 0.9959 F1 score, by detecting cyberattacks targeting IoT environmental network traffic. This research aims to classify and detect various cyberattacks that occur in IoT environments, but because they use payload-based data to detect cyberattacks, they are challenging to apply to encrypted communication environments. Research related to cyberattack detection over encrypted traffic (LightGBM model [26], Stacking model [33], Deep Autoencoder [35], and LSTM [40]) showed high cyberattack detection performance, such as 0.999 accounting and 0.999 F1 score. However, the cyberattack detection method using packet-level features showed relatively low cyberattack detection and classification performance [40]. This is presumed to be due to the low amount of information identifiable in the encrypted packet, so the detection rate of malicious traffic by some types of malicious code is close to zero. Research related to cyberattack detection over encrypted traffic aim to detect cyberattacks carried out by hiding in encrypted traffic. However, they do not include consideration of IoT environments and cyberattacks in IoT, so it is unclear whether they are applicable to IoT environments. The ECDS-IoT presented in this research is designed to detect cyberattacks performed by being hidden from encryption traffic generated in IoT environments. As a result of evaluating the performance of cyberattack detection on encrypted traffic generated in the IoT environment, our proposed ECDS-IoT derived high cyberattack detection characteristics such as 0.9973 accuracy and 0.9957 F1 score. The cyberattack detection

performance of ECDS-IoT is comparable to or better than that of cyberattack detection research on unencrypted traffic in existing IoT environments. This means that ECDS-IoT's cyberattack detection performance on encrypted traffic is not inferior to existing research that detects cyberattacks based on plain network packet-level features. It also outperforms existing research on detecting cyberattacks on encrypted traffic. For these reasons, ECDS-IoT can effectively detect cyberattacks with encrypted communications when applied to IoT environments.

Table 7 summarizes the performance comparison results of ECDS-IoT with research of IoT cyberattack detection and research on anomaly detection over encrypted traffic.

Table 7: Answers to research question 3

RQ 3	Does ECDS-IoT maintain its detection performance compared to the cyberattack detection model targeting non-encrypted traffic in the existing IoT environment?
	Our proposed ECDS-IoT derives a level of F1 score similar to that of high accuracy as a result of comparison with the research of non-encrypted traffic cyberattack detection in the IoT environment. In addition, most of the performances are similar to or higher than those suggested in previous research when compared to the cyberattack detection model aimed at detecting cyberattacks over encrypted traffic. As a result of the derivation of such cyberattack detection performance, ECDS-IoT, which performs cyberattack detection over encrypted traffic generated in the IoT environment, shows that it does not fall off in terms of cyberattack detection performance of existing research that performs cyberattack detection for non-encrypted traffic in IoT environments, and it also shows that the cyberattack detection performance over encrypted traffic is not lagging behind.

5 Limitations

This research proposed ECDS-IoT to detect cyberattacks over encrypted traffic in IoT environments. ECDS-IoT has effectively detected cyberattacks over encrypted traffic in the IoT environment and achieved high performance, but some limitations exist. This section discusses these limitations:

- ECDS-IoT derived high performance on datasets containing encrypted traffic collected in IoT environments. However, the effectiveness of cyberattack detection in real-world IoT environments, where new IoT devices are introduced frequently, and network flows vary depending on operational purposes, remains uncertain.
- In this research, only the cyberattack detection performance of ECDS-IoT was discussed. When ECDS-IoT is applied to actual IoT environments, it can have negative effects, such as reducing the operability of IoT environments due to increased computational overhead for feature extraction and cyberattack detection in encrypted traffic.
- ECDS-IoT extracts statistical features from encrypted traffic and uses them to detect cyberattacks. However, it fails to account for potential limitations or biases in the feature extraction process.

The dataset employed in this research to evaluate the performance of ECDS-IoT in detecting cyberattacks over encrypted traffic includes encrypted traffic generated by 33 cyberattacks in seven categories on an IoT testbed consisting of 105 IoT devices. The IoT environment performs communication utilizing TCP, UDP, SSL, and TLS communication protocols, including network traffic. The IoT

environment includes more devices, and various communication protocols are applied and utilized. It is necessary to include more IoT devices and use data collected in different environments with different communication protocols to address the limitations presented in this section. In addition, an IoT testbed should be built, and research should be conducted to apply the cyberattack detection system over encrypted traffic to the testbed. The problems that can occur in the IoT environment, such as increased computational overhead for feature extraction and cyberattack detection in encrypted traffic, can be identified.

6 Conclusions

Research on cyberattack detection technology based on AI is actively conducted for IoT cybersecurity. However, most existing studies use a method of extracting features based on the primary information of plaintext network packets and detecting cyberattacks using them. As cyberattacks using IoT's plaintext communication vulnerabilities increase, encrypted communication methods are applied to many IoT environments. Based on the application of the encrypted communication method, significant information on network packets and others is encrypted. For this reason, it has become difficult to apply a cyberattack detection method for IoT that detects cyberattacks based on crucial information such as payload in network packets. In addition, research that has performed the purpose of cyberattack detection over encrypted traffic is not considered for IoT environments and cyberattacks that occur, making it challenging to apply them to IoT environments. Therefore, this research proposes ECDS-IoT, a cyberattack detection system over encrypted traffic in IoT environments. ECDS-IoT determines identifiable information on encrypted traffic collected in the IoT environment and extracts statistics-based features through statistical analysis of identifiable information. ECDS-IoT understands information about normal data by learning only statistics-based features extracted from normal data. Based on this learning information, data showing a specific difference from normal data is detected as cyberattack data. This research uses CICIoT 2023, which includes normal encryption network packets and cyberattack encryption network packets collected from IoT environments that communicate based on Z-wave, Zigbee, and Wi-Fi, to evaluate the cyberattack detection performance of ECDS-IoT. It derives statistics-based features using a CICFlowmeter for the dataset. Afterward, a cyberattack detection model is implemented utilizing Autoencoder, RNN, GRU, LSTM, BiLSTM, and AE-LSTM algorithms. This study evaluates the performance of the cyberattack detection model using statistics-based features and derives 0.99739 accuracy, 0.99154 precision, 1.0 recall, and 0.99575 F1 score from the AE-LSTM-based cyberattack detection model.

Future works will develop the necessary technologies for the advancement and field application of ECDS-IoT by collecting real-time network traffic for encrypted communication IoT and conducting cyberattack detection research on real-time collected encrypted traffic.

Acknowledgement: None.

Funding Statement: This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2021-0-00493, 5G Massive Next Generation Cyber Attack Deception Technology Development).

Author Contributions: Conceptualization, Il Hwan Ji; methodology, Il Hwan Ji and Ju Hyeon Lee; experiments and validation, Il Hwan Ji and Seungho Jeon; writing, Il Hwan Ji; writing—review

and editing, Ju Hyeon Lee, Seungho Jeon, and Jung Taek Seo. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: All the data used and analyzed is available in the manuscript.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Mohamed M. A comparative study on Internet of things (IoT): frameworks, tools, applications and future directions. J Intell Syst Internet Things. 2020;1(1):13–39.
2. Khanna A, Kaur S. Internet of things (IoT), applications and challenges: a comprehensive review. Wirel Pers Commun. 2020;114:1687–762. doi:10.1007/s11277-020-07446-4.
3. Dargaoui S, Azrour M, El Allaoui A, Amounas F, Guezzaz A, Attou H, et al. An overview of the security challenges in IoT environment. Adv Technol Smart Environ Energy. 2023;151–60. doi:10.1007/978-3-031-25662-2_13.
4. Rana M, Mamun Q, Islam R. Lightweight cryptography in IoT networks: a survey. Future Gener Comput Syst. 2022;129:77–89. doi:10.1016/j.future.2021.11.011.
5. Djenna A, Harous S, Saidouni DE. Internet of things meet internet of threats: new concern cyber security issues of critical cyber infrastructure. Appl Sci. 2021;11(10):4580. doi:10.3390/app11104580
6. Zscaler. Zscaler ThreatLabz 2023 Enterprise IoT & OT Threat Report 2023. Available from: https://www.zscaler.com/resources/2023-threatlabz-enterprise-iot-ot-threat-report. [Accessed 2024].
7. Zscaler. Spoiler: New ThreatLabz Report Reveals over 85% of Attacks Are Encrypted. 2022. Available from: https://www.zscaler.com/blogs/security-research/2022-encrypted-attacks-report. [Accessed 2024].
8. Zscaler. Zscaler ThreatLabz 2023 State of Encrypted Attacks Report 2023. Available from: https://www.zscaler.com/resources/2023-threatlabz-state-of-encrypted-attacks-report. [Accessed 2024].
9. Cai J, Wang Q, Luo J, Liu Y, Liao L. Capbad: content-agnostic, payload-based anomaly detector for industrial control protocols. IEEE Internet Things J. 2021;9(14):12542–54. doi:10.1109/JIOT.2021.3138534.
10. Kim S, Jo W, Shon T. APAD: autoencoder-based payload anomaly detection for industrial IoE. Appl Soft Comput. 2020;88:106017. doi:10.1016/j.asoc.2019.106017.
11. Wang W, Zhu M, Zeng X, Ye X, Sheng Y. Malware traffic classification using convolutional neural network for representation learning. In: 2017 International Conference on Information Networking (ICOIN), 2017; Da Nang, Vietnam: IEEE. doi:10.1109/ICOIN.2017.7899588.
12. Neto ECP, Dadkhah S, Ferreira R, Zohourian A, Lu R, Ghorbani AA. CICIoT2023: a real-time dataset and benchmark for large-scale attacks in IoT environment. Sensors. 2023;23(13):5941. doi:10.20944/preprints202305.0443.v1.
13. Domínguez-Bolaño T, Campos O, Barral V, Escudero CJ, García-Naya JA. An overview of IoT architectures, technologies, and existing open-source projects. Int Things. 2022;20:100626. doi:10.1016/j.iot.2022.100626.
14. Jaloudi S. Communication protocols of an industrial internet of things environment: a comparative study. Fut Internet. 2019;11(3):66. doi:10.3390/fi11030066.
15. Mann P, Tyagi N, Gautam S, Rana A. Classification of various types of attacks in IoT environment. In: 12th International Conference on Computational Intelligence and Communication Networks (CICN), 2020; Bhimtal, India: IEEE. doi:10.1109/CICN49253.2020.9242592.

16. Bacon M. New Mirai Variant Attacks Apache Struts Vulnerability. https://www.techtarget.com/searchsecurity/news/252448779/New-Mirai-variant-attacks-Apache-Struts-vulnerability. [Accessed 2018].
17. Kolias C, Kambourakis G, Stavrou A, Voas J. DDoS in the IoT: mirai and other botnets. Computer. 2017;50:80–4. doi:10.1109/MC.2017.201.
18. Chang H, Feng J, Duan C. HADIoT: a hierarchical anomaly detection framework for IoT. IEEE Access. 2020;8:154530–9. doi:10.1109/ACCESS.2020.3017763.
19. UNB. Intrusion detection evaluation dataset (ISCXIDS2012). Available from: https://www.unb.ca/cic/datasets/ids.html. [Accessed 2024].
20. Liu J, Song X, Zhou Y, Peng X, Zhang Y, Liu P, et al. Deep anomaly detection in packet payload. Neurocomputing. 2022;485:205–18. doi:10.1016/j.neucom.2021.01.146.
21. UNB. Intrusion detection evaluation dataset (CIC-IDS2017). Available from: https://www.unb.ca/cic/datasets/ids-2017.html. [Accessed 2024].
22. Alanazi R, Aljuhani A. Anomaly detection for industrial internet of things cyberattacks. Comput Syst Sci Eng. 2023;44(3):2361–78. doi:10.32604/csse.2023.026712.
23. Al-Hawawreh M, Sitnikova E, Aboutorab N. X-IIoTID: a connectivity-agnostic and device-agnostic intrusion data set for industrial Internet of Things. IEEE Internet Things J. 2021;9(5):3962–77. doi:10.1109/JIOT.2021.3102056.
24. Tomar K, Bisht K, Joshi K, Katarya R. Cyber attack detection in IoT using deep learning techniques. In: 6th International Conference on Information Systems and Computer Networks (ISCON), 2023; Mathura, India: IEEE. doi:10.1109/ISCON57294.2023.10111990.
25. Ferrag MA, Friha O, Hamouda D, Maglaras L, Janicke H. Edge-IIoTset: a new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. IEEE Access. 2022;10:40281–306. doi:10.1109/ACCESS.2022.3165809.
26. Chao D. A mining policy based malicious encrypted traffic detection scheme. In: Proceedings of the 2020 9th International Conference on Computing and Pattern Recognition, 2020; Xiamen, China. p. 130–5. doi:10.1145/3436369.3436479.
27. Tiwari A, Saraswat S, Dixit U, Pandey S. Refinements in Zeek intrusion detection system. In: 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), 2022; Coimbatore, India: IEEE. doi:10.1109/ICACCS54159.2022.9785047.
28. Lab S. CTU-malware-capture-botnet. Available from: https://www.stratosphereips.org/datasets-malware. [Accessed 2024].
29. Niu Z, Xue J, Qu D, Wang Y, Zheng J, Zhu H. A novel approach based on adaptive online analysis of encrypted traffic for identifying Malware in IIoT. Inform Sci. 2022;601:162–74. doi:10.1016/j.ins.2022.04.018.
30. Duncan DB. Malware traffic analysis. Available from: https://www.malware-traffic-analysis.net/. [Accessed 2024].
31. Garcia S, Zunino A, Campo M. Malware capture faculity project. Available from: https://mcfp.weebly.com/. [Accessed 2024].
32. Lab S. The CTU-13 dataset. Available from: https://www.stratosphereips.org/datasets-ctu13. [Accessed 2024].
33. Alzighaibi A. Detection of DoH traffic tunnels using deep learning for encrypted traffic classification. Computers. 2023;12(3):47. doi:10.3390/computers12030047.
34. MontazeriShatoori M, Davidson L, Kaur G, Lashkari AH. Detection of DoH tunnels using time-series classification of encrypted traffic. In: 2020 IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cyber Science and Technology

Congress (DASC/PiCom/CBDCom/CyberSciTech), 2020; Calgary, Canada: IEEE. doi:10.1109/DASC-PICom-CBDCom-CyberSciTech49142.2020.00026.

35. Bahlali AR, Bachir A, Cheriet A. Malicious encrypted network traffic detection using deep auto-encoder with a custom reconstruction loss. In: 2023 International Symposium on Networks, Computers and Communications (ISNCC), 2023; Doha, Qatar: IEEE. doi:10.1109/ISNCC58260.2023.10323710.
36. Lashkari AH, Gil GD, Mamun MSI, Ghorbani AA. Characterization of tor traffic using time based features. In: Proceedings of the 3rd International Conference on Information Systems Security and Privacy, 2017; Porto, Portugal: SciTePress. vol. 1. doi:10.5220/0006105602530262.
37. Moustafa N, Slay J. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS), 2015; Canberra, Australia: IEEE. doi:10.1109/MilCIS.2015.7348942.
38. Sharafaldin I, Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018), 2018; Funchal, Portugal. p. 108–16. doi:10.5220/0006639801080116.
39. Zhao C, Li S, Wu X, Han W, Tian Z, Chen M. A novel malware encrypted traffic detection framework based on ensemble learning. In: 2021 IEEE Sixth International Conference on Data Science in Cyberspace (DSC), 2021; Shenzhen, China: IEEE. doi:10.1109/DSC53577.2021.00097.
40. Ferriyan A, Thamrin AH, Takeda K, Murai J. Encrypted malicious traffic detection based on Word2Vec. Electronics. 2022;11(5):679. doi:10.3390/electronics11050679.
41. Hnamte V, Najar AA, Nhung-Nguyen H, Hussain J, Sugali MN. DDoS attack detection and mitigation using deep neural network in SDN environment. Comput Secur. 2024;138:103661. doi:10.1016/j.cose.2023.103661.
42. Chahid I, Elmiad AK, Badaoui M. Data preprocessing for machine learning applications in healthcare: a review. In: 2023 14th International Conference on Intelligent Systems: Theories and Applications (SITA), 2023; Casablanca, Morocco: IEEE. doi:10.1109/SITA60746.2023.10373591.
43. Kwak SK, Kim JH. Statistical data preparation: management of missing values and outliers. Korean J Anesthesiol. 2017;70(4):407. doi:10.4097/kjae.2017.70.4.407.
44. Hutcheson G. Missing Data: data replacement and imputation. J Model Manag. 2012;7(2):221–33.
45. He K, Kim DD, Asghar MR. Adversarial machine learning for network intrusion detection systems: a comprehensive survey. IEEE Commun Surveys & Tut. 2023;25(1):538–66. doi:10.1109/COMST.2022.3233793.
46. Draper-Gil G, Lashkari AH, Mamun MSI, Ghorbani AA. Characterization of encrypted and vpn traffic using time-related. In: Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP), 2016; Rome, Italy. doi:10.5220/0005740704070414.
47. Yang N-C, Sung K-L. Non-intrusive load classification and recognition using soft-voting ensemble learning algorithm with decision tree, K-nearest neighbor algorithm and multilayer perceptron. IEEE Access. 2023;11:94506–20. doi:10.1109/ACCESS.2023.3311641.
48. Al Razib M, Javeed D, Khan MT, Alkanhel R, Muthanna MSA. Cyber threats detection in smart environments using SDN-enabled DNN-LSTM hybrid framework. IEEE Access. 2022;10:53015–26. doi:10.1109/ACCESS.2022.3172304.
49. Sun X, Wang H. Adjusting the precision-recall trade-off with align-and-predict decoding for grammatical error correction. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022; Dublin, Ireland: Association for Computational Linguistics. vol. 2, p. 686–93. doi:10.18653/v1/2022.acl-short.77