ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20230708000792
Original scientific paper

A Slice Escape Detection Model Based on Full Flow Adaptive Detection

Zhenzhen LIU, Rui ZHOU, Jingbing CHEN, Kangqian HUANG, Jingyin HUANG, Binsi CAI*, Yali GAO, Kaiguo YUAN

Abstract: The 5G power trading private network increases network flexibility and lowers building costs with the aid of 5G and Access Point Name (APN) technology. However, the private network is facing a series of security problems, such as the lack of effective isolation between slices and malicious terminal damage in slices, which result in a large consumption of slice resource failures and even slice escape attacks. To solve this problem, we propose a slice escape detection model based on full flow adaptive detection. Firstly, we improve the "six-tuple" flow table features detection technology, and creatively proposed a set of "eleven-tuple" features scheme, so as to realize the adaptive detection of intra-slice and inter-slice escape attacks. Secondly, we construct a two-level detection model based on long short-term memory network and self-attention mechanism to improve detection efficiency and reduce false alarm rate. Thirdly, we design an exception handling module to handle the abnormally detected traffic. Our model has a high detection accuracy and a low false alarm rate for the slice escape assault, according to a large number of experiments on the CIC-DDoS2019 dataset, and the detection delay complies with the requirements for online detection.

Keywords: long short-term memory; network slicing; slice escape

1 INTRODUCTION

The requirements for communication networks are diversifying as the 5G communication era arrives, ushering in a time of "all things perception", "all things intelligence", and "all things interconnected". If a service necessity is given by an independent network according to conventional network construction theories, it will put tremendous strain on the operator's operations and investments. Consequently, 5G network slicing technology was created. A 5G network can be split into numerous separate virtualized dedicated networks using network slicing technologies. While increasing the use efficiency of network resources and lowering investment in network construction, it satisfies the network requirements of various services. Guangdong Power Trading Center developed the first "5G Power Trading Private Network" in the country using 5G and Access Point Name (APN) technology, which is completely cut off from the logic of the public Internet. Based on assuring the security of the power trading private network, the private network lowers the cost of development and increases the network's flexibility.

The Slice escape assault is a new security concerns and difficulties brought on by the adoption of 5G network technologies. Assaults launched by malevolent terminals within a slice or attacks launched by other slices are referred to as "slice escape attacks." It can be seen that some original traditional security protection technologies are no longer applicable. It is essential to conduct research on the security concerns linked to network slicing technology and offer the appropriate solutions and security policies in order to secure and achieve the security of the 5G power trading private network.

Power market trading platform serves a crucial supporting function in the power industry as one of the important information systems in the energy sector. Its damage or data leakage could have a significant impact on both national security and people's interests in their livelihood. The most prevalent sort of slice escape assault is the denial-of-service attack, which can cause a number of dangers in the 5G power trading private network.

In the 5G Power Trading Private Network, malicious terminals within the slice or malicious attackers of other slices launch denial-of-service slice escape attacks against the slice using forged IP and other technologies, thus consuming resources of slicing to reduce its performance or even interrupt its normal operation. Based on the aforementioned arguments, we investigate how to carry out efficient adaptive detection of slice escape attacks of the denial-of-service variety in the 5G power trading private network.

The main contributions are as follows:

(1) We propose an adaptive detection model of intra-slice and inter-slice escape attacks in the 5G power trading private network, which improves the traditional detection system based on the "six-tuple" flow table features, and creatively expands to "eleven-tuple" features;

(2) We propose a slice escape attacks detection method using full flow adaptive detection of two-level detection method. The method is based on collaborative detection and LSTM-SA model, which integrates information entropy technology, long short-term memory models, and self-attention mechanisms;

(3) Numerous tests using the CIC-DDoS2019 dataset demonstrate that this method is capable of accurately detecting slice escape attacks in live traffic as well as forecasting attack patterns based on current network traffic values, which can improve the detection accuracy and reduce the false positive rate.

2 RELATED WORKS

Denial-of-service Slice escape assaults, which mostly happen during the slice operating phase of 5G networks, include DoS attacks, huge terminal DDoS attacks, excessive resource consumption, etc. The user data layer requests a flow table, which the SDN controller fulfils and sends to the lower layer switch. The SDN controller's flow table will get overloaded when many request signallings attempt to reach it at once, causing a service outage. The slice escape attack in the slice means that the attacker of the slice escape attack comes from the slice. A cross-slice escape attack means that the attacker of a slice escape attack comes from one or several other slices.

Regarding the security of 5G slices, Hamidreza [2] provided a thorough introduction and discussion of the security concerns raised by the combination of 5G and the

Internet of Things, with Denial of Service (DoS) and Distributed Denial of Service (DDoS) assaults being the most common. When it comes to 5G, which will deliver new services and serve as the primary communication platform for IoT, the obstacles will be greater and more difficult than those faced by earlier generations. Many IoT devices have flaws in their hardware, operating systems and design that hackers have exploited. The construction of Denial of Service (DoS) assaults and Distributed Denial of Service (DDoS) attacks, which occur with hacked IoT equipment, is one of the most typical abuses of IoT technology. Denial-of-service attacks prevent authorized users from accessing a certain service. The difference between DoS and DDoS is also in the number of equipment used in this type of attack, implying that a DoS attack originates from a single source, which is usually a server or a computer system, whereas in a DDoS attack, resources are distributed across multiple systems and can even be global. Following an examination of several DDoS defense mitigation solutions based on SDN, NFV, and MTD, Haiou Huang [3] examined the trend of DDoS attacks on the 5G network over the previous two years, concluding that with the advent of 5G and the explosion of the Internet of Things, it has become easier to launch large-scale DDoS attacks. Traditional DDoS solutions are expensive and ineffectual against a wide range of flexible and variable assaults. SDN, as a new network innovation architecture, may provide flexible network traffic control by separating network equipment's control surface from the data surface. As a result, an increasing number of academics are banking on SDN technology to investigate strategies to minimize DDoS attacks. However, after analyzing some SDN-based DDoS solutions, it was discovered that SDN development still faces many difficulties and challenges, implying that there is still a long way to go for the application of SDN in alleviating DDoS attack solutions. Slice isolation was suggested as a method of reducing Distributed Denial of Service (DDoS) assaults by Danish Sattar [4]. In this isolation mode, the host hardware resources across the slices and the host hardware resources between the slice's components are completely isolated. Isolation inside and between slices can therefore be achieved. A real-time DDoS defense mechanism based on multi-domain collaboration was also proposed by Chen Xu [5]. This architecture employs defence countermeasures at the source, as well as light-weight detection approaches such as trust management and blacklist/whitelist technologies. The vast majority of DDoS attack traffic is intercepted and terminated before it reaches the data network. Badre Bousalem [6] presents a 5G prototype for attacks detection and mitigation in sliced networks leveraging Machine Learning (ML). The 5G prototype, based on OpenAirInterface, enables the creation of network slices on demand and the dynamic management of physical resources based on user behavior, while taking into account inputs from a northbound Software Defined Network (SDN) application. They concentrate on Distributed Denial of Service (DDoS) assaults, in which one or more malicious users launch attacks on the 5G Core Network. The prototype, which is based on the ML module, can identify such assaults and then automatically builds a sinkhole-type slice with a tiny percentage of physical resources and isolates the malicious users within this slice to minimize the attackers' activity. The current academic research on 5G slice security mainly focuses on how to alleviate the security risks between slices, and pays less attention to the security risks within slices and how to detect slice escape attacks.

In the detection of distributed denial of service attacks, DDoS attack has a high impact on crashing the network resources, making the target servers unable to support the valid users. The current methods deploy Machine Learning (ML) for intrusion detection against DDoS attacks in the SDN network using the standard datasets. However, these methods suffer several drawbacks, and the used datasets do not contain the most recent attack patterns - hence, lacking in attack diversity. Elsayed [7] propose DDoSNet, an intrusion detection system against DDoS attacks in SDN environments. This method is based on Deep Learning (DL) technique, combining the Recurrent Neural Network (RNN) with autoencoder. The model is evaluated using the newly released dataset CICDDoS2019, which includes various DDoS attacks and addresses gaps in existing current datasets. The model achieves significant improvements in attack detection compared to other benchmark methods. A DDoS attack detection model based on SDN was introduced by Chen Li [8], using the OpenFlow switch flow entry information in the SDNarchitecture and creating 6 feature vectors as the detection model's input. A DDoS attack detection model based on BP neural network is built using the concept of binary categorization. In order to classify and identify the arrival flow, Tahmasebi [9] trained a classifier based on Euclidean distance using the approach of computing feature entropy and the model of queuing theory to evaluate the arrival flow. When analyzing packet samples taken from network traffic, Abdullah [10] employed a deep neural network (DNN) as a deep learning model to find DDoS attacks. The deep neural network (DNN) is proposed in this article as a deep learning model for detecting DDoS assaults on a sample of packets recorded from network traffic. DNN models may perform rapidly and accurately even with tiny samples since their structure includes feature extraction and classification procedures, as well as layers that update themselves as they are taught. Experiments using the CICDDoS2019 dataset, which contains the current DDoS attack types developed in 2019, revealed that assaults on network traffic were identified with 99.99% success and attack types were categorized with an accuracy rate of 94.57%. The high accuracy values obtained demonstrate that the deep learning model can efficiently prevent DDoS assaults. By utilizing deep convolutional neural networks (CNNs) and actual network data, Qinghe Du [11] established a comprehensive system to allow early detection of distributed denial-of-service (DDoS) attacks coordinated by botnets managing malicious devices. Zhou [12] addressed the stage long-term dependency problem when using Hidden Markov Models (HMMs), and proposed a new detection solution using a sequence-to-sequence (seq2seq) model. A multi-stage assault is a complex intrusion tactic that has been routinely utilized to breach well-defended network systems. Modern research recommends using a hidden Markov model (HMM) to detect such assaults. However, while HMMs can model the relationships and dependencies between different alerts and stages for detection, they cannot handle stage dependencies that are buried in a longer sequence of alerts. They address the problem of long-term phase dependence in this study and present a new detection method based on a sequence-to-sequence (seq2seq) model. The fundamental concept is to employ a Long Short-Term Memory (LSTM) network to encode a sequence of alerts into a latent feature vector, and then decode that vector into a sequence of projected attack phases using another LSTM. The model learns to "forget" irrelevant alarms by utilizing a Long Short-Term Memory (LSTM) network, giving it a better probability of "remembering" long-term relationships between successive phases of sequence recognition. Tan [13] proposed a DDoS attack detection and defence framework in the SDN environment using a combined machine learning algorithm based on K-Means and KNN to take advantage of the rate characteristics and asymmetric characteristics of traffic, and detect suspicious traffic determined by the detection trigger mechanism. Novaes [14] proposed a method (LSTM-FUZZY) for detecting and repelling distributed denial-of-service (DDoS) and port scanning attacks in an SDN context using long short-term memory networks. They described a modular approach for detecting and mitigating abnormalities in SDN networks in this paper. The system is composed of three components, and its actions are automated to facilitate monitoring, detection, and mitigation of assaults. They devised a new way to forecast the usual behavior of network operation in the first module, which is responsible for traffic categorization, by employing an approach of Long Short-Term Memory (LSTM) semi-supervised using IP flows. In the second module, they provided a technique for recognizing assaults using Bienaymé-Chebyshev's inequality and Fuzzy logic. At last, in the third module, they implemented automated mitigation procedures to limit the harm caused by assaults while maintaining network functioning requirements. Most of the current academic research on DDoS is based on SDN networks or related data sets in the field of network security, and there are few studies on specific slicing scenarios.

It can be seen from the above analysis that there is currently no lack of research on 5G slice security and DDoS attacks under 5G networks at home and abroad, but most of them focus on prevention and mitigation. For the isolation failure in the 5G power trading private network and the slice escape attack caused by malicious terminal attacks, it is impossible to effectively detect. The majority of DDoS attack detection research uses deep learning or machine learning techniques. Through extensive training, the classifier has the ability to detect abnormal 5G network traffic. The traditional DDoS detection method deployed in the 5G network cannot make full use of the characteristics of the OpenFlow protocol, and cannot solve the security problems existing in the 5G power trading private network and the security problems existing in the 5G power trading private network, achieve adaptive detection of slice evasion behavior between slices and within slices.

The difference in this article is reflected in:

(1) Fully studying the characteristics of 5G networks and slices, and collecting flow table information in OpenFlow switches through sFlow agents to avoid controller overload;

(2) Adaptive detection of intra-slice and inter-slice escape attacks is realized by extracting features;

(3) The collected information is processed through the first-level detection model to calculate the information entropy, and the flow that the calculation results show is abnormal and checked through the two-level detection model. The two-level detection model's design can decrease the false alarm rate while also increasing detection effectiveness.

The slice escape attack detection model can effectively detect the slice escape attack of the denial-of-service type within the slice and the slice escape attack of the denial-of-service type caused by the isolation failure when crossing slices in the 5G power trading private network.

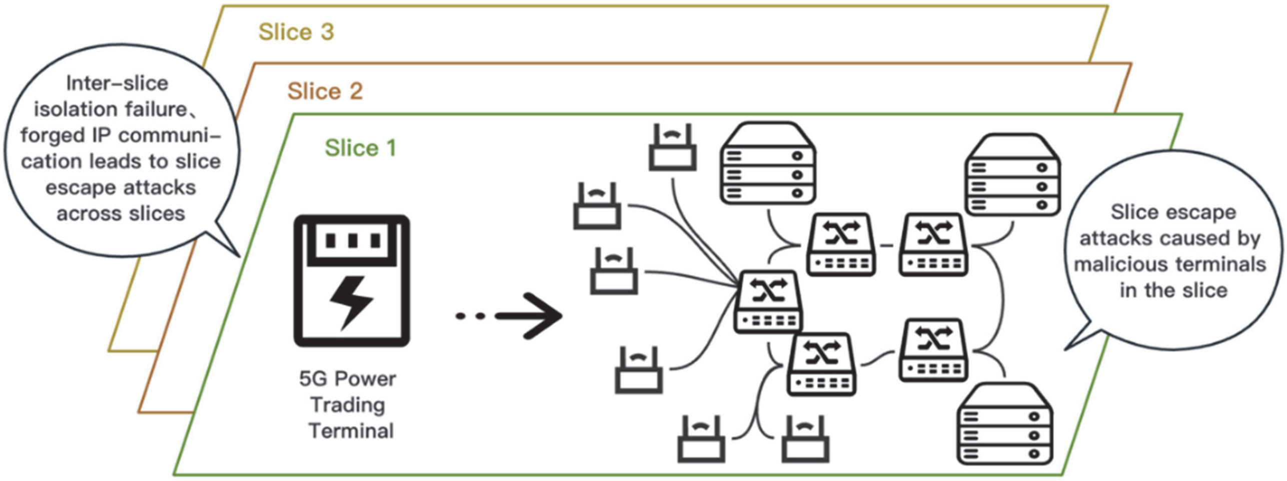
Slice 3
Slice 2
Slice 1
Inter–slice isolation failure、forged IP communi–cation leads to slice escape attacks across slices
5G Power Trading Terminal
Slice escape attacks caused by malicious terminals in the slice

Figure 1 Security issues in 5G power trading private network

3 MODEL ARCHITECTURE AND ALGORITHM DESIGN

As shown in Fig. 1, through analysis and research, we have discovered the security issues caused by the introduction of the slice concept and Software-defined networking (SDN) in the 5G network. That is:

(1) Malicious terminals in a slice may destroy the performance of the entire slice, and slice escape attacks can also be realized by excessive consumption of shared resources used to authorize access to slices;

(2) When the isolation between slices fails or malicious terminals of other slices communicate with this

slice through forged IP, malicious terminals from other slices may attack this slice, resulting in performance degradation or resource exhaustion of this slice.

The above problems exist in the 5G power trading private network and threaten power trading at all times. Therefore, we aim at the key issue of "how to effectively detect slice escape attacks such as DoS and DDoS attacks that may exist between slices and within slices in the 5G power trading private network", a deep learning-based full-flow adaptive slice escape attack detection model is proposed deployed on the SDN controller, as shown in Fig. 2, so as to realize the detection of slice escape attacks in the 5G power trading private network, and avoid the loss of power user data and property.

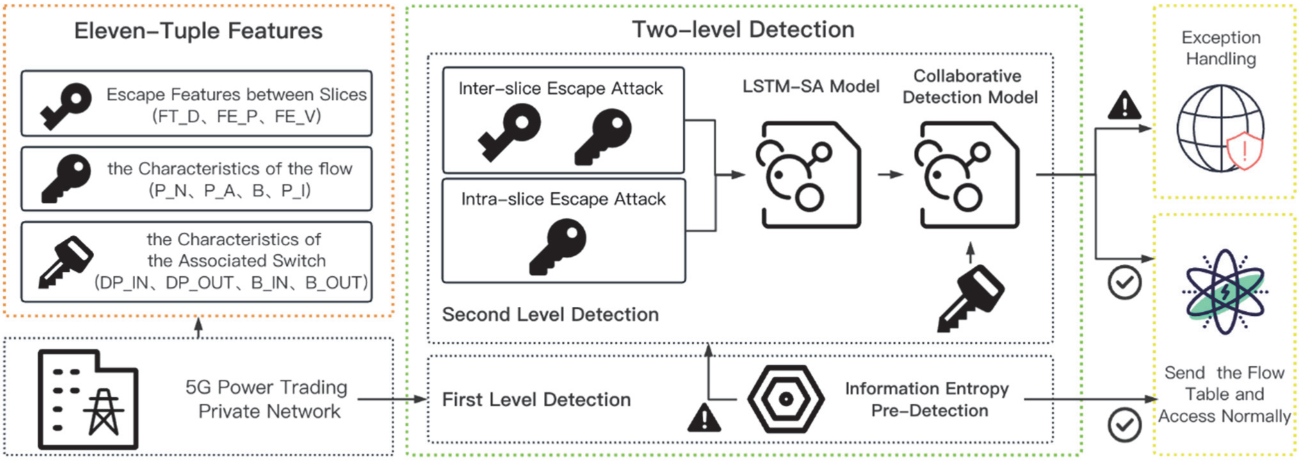
Eleven–Tuple Features
Escape Features between Slices (FT_D、FE_P、FE_V)
the Characteristics of the flow (P_N、P_A、B、P_I)
the Characteristics of the Associated Switch (DP_IN、DP_OUT、B_IN、B_OUT)
5G Power Trading Private Network
Two–level Detection
Inter–slice Escape Attack
Intra–slice Escape Attack
LSTM–SA Model
Collaborative Detection Model
Second Level Detection
First Level Detection
Information Entropy Pre–Detection
Exception Handling
Send the Flow Table and Access Normally

Figure 2 A slice escape attack detection model based on full flow adaptive detection of two-level detection method

As shown in Fig. 2, the model includes three parts: first level detection module, second level detection module and exception handling module.

Firstly, the first level detection module adopts the method of information entropy. The first-level detection of data packets is realized by using the reduction of the information entropy value in the system when a slice encounters a denial-of-service slice escape attack. For the abnormal data packets detected by the first-level detection module, they will be reviewed by the second-level detection module.

Secondly, the second-level detection module is divided into two parts: intra-slice escape attack detection and inter-slice escape attack detection. The first part is the intra-slice escape attack detection part. In this section, the sFlow agent helps gather the flow table data from the OpenFlow switch with the help of the 5G slice SDN controller of the power trading private network and extracts two sets of features from the flow table data: The first group is the characteristics of the flow, that is [P_N, P_A, B, P_I], and the second group is the characteristics of the associated switch, that is [DP_IN, DP_OUT, B_IN, B_OUT]. The probability P that this flow will be attacked by slice escape can be determined by feeding the LSTM-SA model with the characteristics of the first group of flows and datagram segments. After processing, the second set of features together with the output P of the LSTM-SA model constitute the input of the collaborative detection model. The collaborative detection module corrects the detection results by integrating the switch association features to ensure the accuracy of detection of early slice escape attacks. The second part is the inter-slice escape attack detection part. The inter-slice evasion attack detection part is different from the intra-slice evasion attack detection part in the feature extraction part: the inter-slice evasion attack detection module will extract an additional set of features from the flow table information, that is, [FT_D, FE_P, FE_V]. Attackers usually implement cross-slice evasion attacks by forging IPs and constantly changing forged addresses. As a result, the flow table of the OpenFlow switch will contain many flow entries that are not matched by data packets for a considerable amount of time.

Thirdly, the data packet will be processed according to the detection result. If it is a normal data packet, the flow table will be issued. If it is an abnormal data packet, discard the single flow. If the number of violations of the source IP address of this flow exceeds the set maximum value, the address will be blocked. The main module functions are described in detail below.

3.1 First Level Detection Module

This module mainly uses sFlow agent technology and information entropy technology.

3.1.1 sFlow Agent Technology

sFlow technology is a traffic monitoring technology that randomly samples the data flow with the device port as the basic unit. Not only can it provide complete real-time traffic information from the second layer to the fourth layer or even the entire network, but it can also adapt to traffic analysis in the environment of super large network traffic (such as greater than 10 Gbit/s); this allows users to analyze network traffic performance, trends, and problems in detail and in real time. The sFlow Agent and sFlow Collector make up the sFlow monitoring tool. The agent, which functions as a client and is integrated into network forwarding devices like switches and routers, encapsulates the data into sFlow packets by collecting interface statistics and data from the device. The sFlow Agent will deliver the sFlow packet to the selected Collector when the sFlow packet buffer is full or the sFlow packet cache time (1 second) expires. As a remote server, Collector is responsible for analyzing, summarizing, and generating traffic reports of sFlow packets. This model uses this process to obtain traffic information under the control of the SDN controller in the 5G power trading private network.

3.1.2 Information Entropy Technology

Due to the random nature of traffic in network slicing during normal operation, the information entropy in 5G slicing environment is generally maintained at a fixed value. However, a lot of attack traffic will congregate at the attacked host when a slice is subject to a denial-of-service slice escape attack. The attack traffic will, however, converge on the targeted host when a slice is subjected to a denial-of-service slice escape attempt. Therefore, by observing the change of information entropy, it can be judged whether the slicing system encounters a denial-of-service type slicing escape attack. Eq. (1) is the calculation method of Shannon information entropy.

$$H\left(x_i\right)=-\sum_{i=1}^{n}p\left(x_i\right)\log_2\left(x_i\right) \tag{1}$$

In Eq. (1), x represents a random variable and p represents a probability.

In the 5G power trading private network, the source address, destination address, number of data packets and protocol of the data packet are extracted from the SDN data plane switch flow entry to form a quadruple

$$I=\langle SrcIP, DstIP, Count, Type\rangle$$

where *SrcIP* represents the source address of the data packet, *DstIP* represents the destination address of the data packet, *Count* represents the number of data packets, and *Type* represents the transmission protocol used.

Assuming that the information entropy threshold K is specified, and that the quadruple I is subjected to the information entropy computation each time m data packets are gathered. Setting the threshold is dependent on the network's normal state's information entropy value. If the entropy value H obtained by 5 consecutive calculations is less than K, it indicates that there may be abnormal traffic in the network. To assess whether the traffic is anomalous, 300 data packets must be detected if the window size is set to 60. The reason for choosing five windows for continuous detection is to avoid accidental results and reduce the interference of unexpected events on detection information.

However, the conclusions drawn by judging information entropy are sometimes inaccurate. On the 5G power market trading platform, there are times when a specific server remains in a high-load state for an extended period of time due to business needs. Since a lot of business traffic congregates here, the slice network's randomness is reduced as a result. Therefore, the alteration in information entropy is not sufficient evidence that a denial-of-service slice escape attack occurred on the slice, and it is necessary to use deep learning methods to review suspicious traffic.

3.2 Second Level Detection Module

3.2.1 Feature Extraction

Aiming at the slice escape attacks in 5G slices, we divide them into slice escape attacks within slices and slice escape attacks between slices. Feature extraction is an important basis for building slice escape behaviour detection models. The feature extraction module will be deployed on the SDN controller of the 5G power trading platform, and will send a request to the data plane every three seconds to query the status information of all flows on the data plane.

The input features of the slice escape attack detection model, that is, the eleven-tuple features we proposed, are shown in Tab. 1.

Table 1 Input features of the slice escape attack detection model

Feature	Feature Name (Abbreviation)
the Characteristics of the Flow	The number of data packets (P_N)
	The average size of data packets (P_A)
	The number of bytes (B)
	The arrival time interval of data packets (P_I)
the Characteristics of the Associated Switch	The maximum inflow party of the number of data packets per unit time (DP_IN)
	The maximum outgoing source of the number of data packets per unit time (DP_OUT)
	The maximum inflow of bytes per unit time (B_IN)
	The maximum outflow of bytes per unit time (B_OUT)
Escape Features between Slices	Median flow table duration (FT_D)
	Percentage of IP paired flow entries (FE_P)
	The absolute value of the number of entries in a single flow entry (FE_V)

As shown in Tab. 1, a total of 11 features are used as the input of the slice escape behaviour detection model. Among them, there are four characteristics of the flow: the number of data packets, the number of bytes, the average size of data packets, and the arrival time interval of data packets. The attack traffic usually contains only one data packet and the number of bytes is significantly larger than that of the normal flow, and the arrival interval of the data packets is very short. Therefore, using the characteristics of these four flows, attack traffic can be detected. There are four switch association features: the largest inflow party of the number of data packets per unit time, the largest outflow party of the number of data packets per unit time, the largest inflow party of bytes per unit time, and the largest outflow party of bytes per unit time. When a slice escape attack occurs, the attack traffic will affect the entire link from the source address to the destination address, and the switch correlation feature can help comprehensively consider the information of the entire link to avoid false positives. And three features used to detect slice escape attacks between slices: the median flow table duration, the percentage of IP paired flow entries, and the absolute value of the number of single flow entries. The following describes in detail how the extracted features for detecting slice escape attacks between slices are used to detect slice escape attacks between slices.

When an attacker conducts cross-slice DoS, DDoS and other slice evasion behaviours to attack a slice, the IP address will be forged and the forged address will be constantly replaced. Therefore, there will be a large number of flow entries that are not matched by data packets for a long time in the flow table of the OpenFlow switch. A flow item will immediately vanish if it is not matched for more time than the "idle timeout". When a slice escape attack occurs, there will be a large number of flow entries with a duration shorter than "idle timeout" in the OpenFlow flow table, and there will be a lack of flow entries with a duration longer than "idle timeout". Therefore, the flow table's median duration is able to demonstrate how the flow and switch are doing and demonstrate whether slice escape is attacking them. At the same time, the median can not only reflect the duration of the flow entry, but also avoid the influence of abnormal maximum and minimum values, improve accuracy and reduce false alarm rate.

An enormous amount of unpaired flow entries will show up in the OpenFlow flow table when a cross-slice escape attack takes place. A flow entry can typically locate a flow entry that matches its source IP address and destination IP address. Although under normal circumstances, the OpenFlow switch uses an asymmetric routing mechanism for link load balancing may also lead to a certain degree of unpaired flow entries, but the proportion is different from that when encountering a slice escape attack. The ratio of IP paired flow entries to all flow entries may indicate whether slice escape is being used to assault the switch.

The absolute value of the number of single flow entries is added in order to overcome the insensitivity of the detection impact brought on by the ratio of IP paired flow entries to the total flow entries when the OpenFlow switch has multiple flow entries.

3.2.2 LSTM-SA Model

We use a technique known as LSTM-SA, which combines the long short-term memory model and the self-attention mechanism. This article's scenario is a supervised binary classification model with a long input sequence that consists of a feature and flow sequence inside a time period. During training, long sequences frequently experience gradient disappearance or explosion. We thus opt for the model of long-term short-term memory that can resolve the gradient disappearance and explosion issue. The long-term short-term memory model performs better in extended sequences than the regular recurrent neural network. In model training, the input of the model is eleven-dimensional features and datagram segments, and not all traffic features contribute equally to traffic detection. To focus on the features that need to be focused on, give them greater weight, gather more information, and disregard irrelevant information, use the self-attention process to determine the relative value of each feature. The combination of LSTM and self-attention mechanism can realize full traffic adaptive detection based on features and traffic sequence, and obtain the probability of slice escape attack in the traffic sequence. The overall flowchart of the LSTM-SA method is shown in Fig. 3.

As shown in Fig. 3, after the data has undergone feature analysis, a weight will be assigned to each feature through the self-attention mechanism to extract effective features of the original data, and after that, the feature weighted total will be determined. Then, after the LSTM layer processing operation, the prediction result is obtained by the activation function mapping, so as to realize the detection of the slice escape attack. The LSTM-SA detection algorithm flow is shown in Algorithm 1.

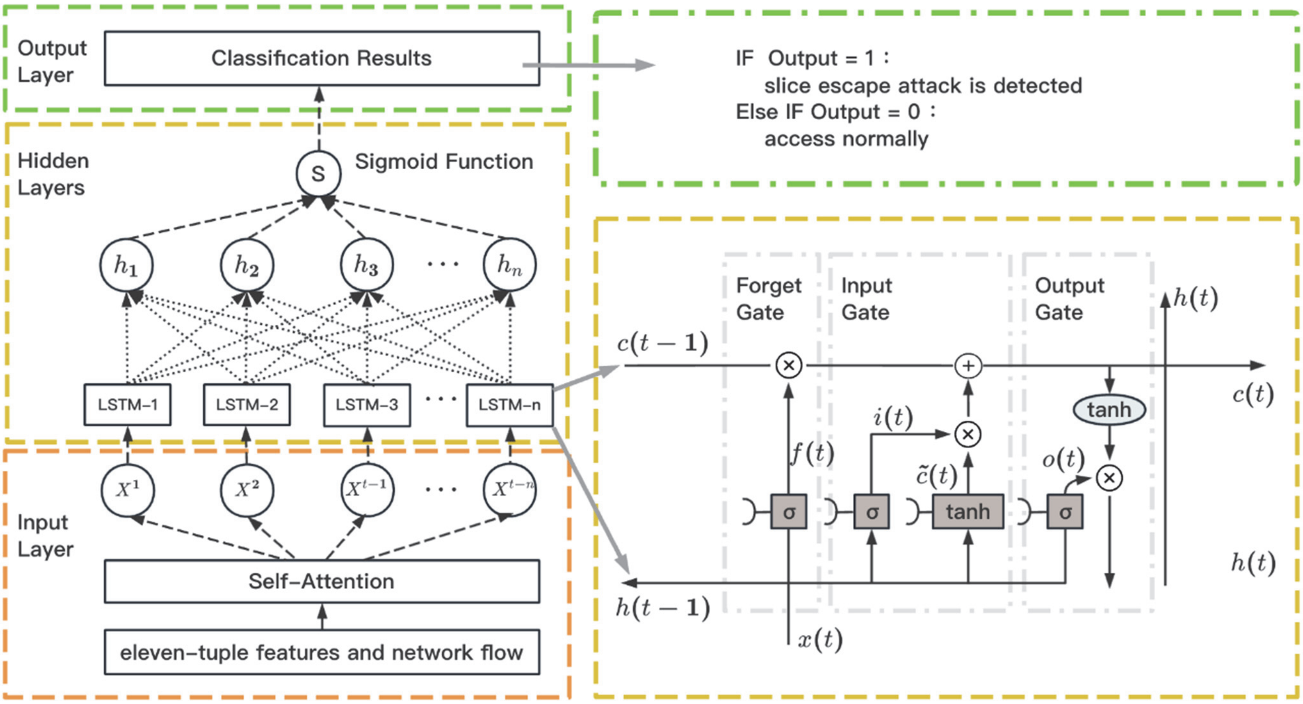

Figure 3 Overall flowchart of the LSTM-SA method

In Algorithm 1, h_t stands for hidden state, f_t represents the calculation result of the hidden state and the input of the unit in the hidden layer, i_t represents the calculation result of the input layer, C_t represents the current new state and c_t represents the old state of the cell. o_t represents the output result of the output gate.

Algorithm 1: LSTM-SA Detection Algorithm

```
INPUT: P_N,P_A, B, P_I, DP_IN, DP_OUT, B_IN, B_OUT,
       FT_D, FE_P, FE_V, X={$x_1, x_2, x_3, \ldots, x_n$}
OUTPUT: final_state
parameter initialization
R ← Attention(score(X))
input ← standardization (P_N, P_A, B, P_I, DP_IN,
  DP_OUT, B_IN,B_OUT, FT_D, FE_P, FE_V, R)
FOR epoch ← 0 to nDo:
    state ←stacked_lstm.zero_state(batch_size)
    FORi in range(len(num_steps)):
        $h_t$, state = stacked_lstm(flows[:, i], state)
        combine ← input + initial_state
        $f_t$ ← forget_layer(combine)
        candidate ← candidate_layer(combine)
        $i_t$ ← input_layer(combine)
        $C_t$ ←$c_t$ × $f_t$+ candidate × $i_t$
        $o_t$ ← output_layer(combine)
        $h_t$ ←$o_t$ × tanh($C_t$)
    final_state ← $h_t$ ←$o_t$× tanh($C_t$)
    END FOR
END FOR
```

The self-attention mechanism is to focus on the target that needs to be focused on, assign more weights, obtain more detailed information of the target, and ignore unimportant information. In model training, not all traffic features contribute equally to traffic detection. Therefore, by using the self-attention mechanism to find the relative importance of each feature.

The attention mechanism assigns an a_i to each traffic feature,

$$e_i = \sigma\left(W_h h_i + b_h\right) \tag{2}$$

$$a_i = \frac{e^{e_i}}{\sum_{K=1}^{T} e^{e_k}}, \sum_{i=1}^{T} a_i = 1 \tag{3}$$

where W and b stand for the weight and bias of the attention mechanism, whereas σ stands for the Sigmoid function. The overall flow is denoted by r, which is the weighted sum of all flow characteristics,

$$r = \sum_{i=1}^{T} a_i h_i, r \in R^{2l} \tag{4}$$

where a_i represents the weight value assigned to each traffic feature, and throughout the training process, the weight is continuously optimized, ensuring that the most crucial elements are given bigger weights.

In each LSTM unit, there are gates that perform different operations (input gate i_t, output gate O_t, forget gate f_t), in order to realize decisions such as which features and flow sequence information of previous steps are forgotten and which information is remembered. The LSTM network can capture the relationship between the input features and the flow sequence, and will not learn irrelevant or meaningless points. Forget gates prevent infinite loops by controlling the flow of information from storage blocks to storage units, thus ensuring that the network has finite memory.

The neural network's performance is significantly impacted by the activation function. ReLU activation is a preferable option. In the majority of instances, it performs superbly, and it also fixes the "gradient disappearance" issue in deep learning. This article expects to use a deeper network to improve the accuracy of detection, and will also face the problem of "gradient disappearance". As a result, the network layer's activation function is the ReLU activation function, which is expressed as:

$$g\left(z\right) = \max\left\{0, z\right\} \tag{5}$$

where z represents the calculation result of the neuron. Sigmoid is utilized as the output layer's activation function since the model ultimately has to produce a decimal value between [0, 1] to indicate the likelihood of being attacked.

The exponential function of the Sigmoid output unit is used, while the Sigmoid activation function is used as the input layer. Saturation will occur when the variable z has a very big absolute value. As a consequence, the logarithmic function and the negative log-likelihood loss function are used to reduce the exponential influence of the Sigmoid output unit:

$$l\left(\theta\right) = -\sum_{i=1}^{m} \log\left(P\left(x^i; \theta\right)\right) \tag{6}$$

where P is the slice escape detection distribution model we defined, and θ is the parameter of the distribution model. Our goal is to select a suitable guarantee that the model we construct is as precise as feasible.

3.2.3 Collaborative Detection

A major feature is to transform collaborative detection into another machine learning problem, so it is necessary to choose an appropriate machine learning model. As with OpenFlow flow table features, collaborative detection is a supervised binary classification problem with the same number of training data. Despite the smaller dimensionality, the data scale is extremely vast. Therefore, the collaborative detection method can still use the LSTM-SA model. Additionally, because technologies like Adam optimization and Dropout regularization are so adaptable, methods for collaborative detection can also use them. Although deep learning models are used, the problem inputs differ, so a new model must be trained and its hyperparameters must be readjusted to improve the model's performance.

Ensemble learning is quite similar to using various machine learning techniques to solve a problem, yet this study methodology is extremely distinct from ensemble learning. The goal of ensemble learning is to finish the learning task by assembling and merging numerous learners, integrating the individual benefits of each learner, and improving overall performance while using the same training data. However, this study introduces switch association information in the process of collaborative detection model training to enrich the dimensions of the training data instead of performing multi-model training on fixed-dimensional data.

Different from OpenFlow flow table features, switch association features cannot be used directly, because these feature data are only the name or number of the OpenFlow switch, which need to be converted before they can be used. Every Δt, the controller extracts a group of switches' flow table features, and each OpenFlow switch has a matching record of its own. The collaborative detection model will obtain the probability of slice escape attack behaviour on the switch corresponding to each record. At this time, replace the switch number in the switch association feature extracted at the same time with the probability that the switch has a slice escape attack behaviour at this time. It can be combined with the previous results to form a collaborative detection feature. Fig. 4 depicts a schematic representation of the switch S_4's distinctive format conversion.

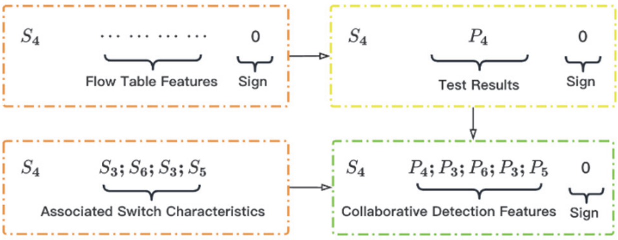
S_4
$\underbrace{\cdots\cdots\cdots\cdots}_{\text{Flow Table Features}}$
$\underbrace{0}_{\text{Sign}}$
S_4
$\underbrace{P_4}_{\text{Test Results}}$
$\underbrace{0}_{\text{Sign}}$
S_4
$\underbrace{S_3; S_6; S_3; S_5}_{\text{Associated Switch Characteristics}}$
S_4
$\underbrace{P_4; P_3; P_6; P_3; P_5}_{\text{Collaborative Detection Features}}$
$\underbrace{0}_{\text{Sign}}$

Figure 4 Schematic diagram of the characteristic format of the switch S_4

As shown in Fig. 4, S_N represents the N-th switch, and P_N represents the detection result of the N-th switch. It can be found that when combined with the detection results, the switch number S_3, S_6, S_3, S_5 in the switch association feature is replaced by the corresponding detection result P_3, P_6, P_3, P_5 and then added to the collaborative detection feature as the input of the algorithm. Since P_N is required to be the detection result within the same Δt, the order in which the training data is extracted must be kept when the detection algorithm is trained, so that it can be mapped from S_N to P_N, and collaborative detection can work effectively.

3.3 Exception Handling Module

After the traffic detection is completed, the next step is to process the attack traffic. When the attacker controls the botnet to launch a slice escape attack on the server, the attack traffic first reaches the edge switch. In the 5G power trading private network, we must determine if there is flow table information corresponding to the flow label in the detection flow space of the edge switch. If so, information will be promptly forwarded in accordance with the requirements of the flow table. If not, the flow data will be uploaded to the controller and held there until the controller determines what to do.

After the controller receives the flow information, it executes the second-level slice escape attack detection algorithm to determine whether the flow is the slice escape attack traffic. If not, proceed with regular computation and forwarding, that is, transmit the flow table with detailed routing information to the switch, and the switch will proceed with normal forwarding operations according to the flow table's rules. If affirmative, the controller gives the switch a flow table containing discard instructions, and the switch discards the slice evasion attack traffic.

4 EXPERIMENT AND RESULTS

4.1 Experimental Dataset Description

The experiment selects the CIC-DDoS2019 dataset [1] (Canadian Cyber Security Institute, 2019). According to research conducted by the Canadian Security Institute (CIC) and the Canadian Communications Security Agency (CSE), traditional classic data sets (such as KDDCUP99, NSLKDD and other data sets) lack current traffic characteristics due to the long time, lack of inclusiveness, diversity and other issues, so it is already unreliable. The CIC-DDoS2019 dataset contains different types of DDoS attacks and real traffic profiles. The full dataset contains 50,063,112 real DDoS attack samples, including 50,006,249 DDoS attacks and 56,863 benign network traffic instances. The dataset includes DDoS attack types such as UDP, SYN, DNS, LDAP, NetBIOS, NTP, MSSQL, UDP-Lag, and SNMP.

4.2 Evaluation Metrics

We employ the "Root Mean Square Error (RMSE)" index, which is extensively used in timing prediction, to assess the influence of slice escape attack detection using a full-flow adaptive detection approach. The root mean square error expresses the actual difference between the expected and actual values. The lower the estimated value, the closer the estimated amount is to the true value, and hence the better the outcome. RMSE calculation is shown in Eq. (7).

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(y_t^{'} - y_t\right)^2} \tag{7}$$

In Eq. (7), $y_t^{'}$ represents the predicted value of the t-th flow, and y_t represents the real value of the t-th flow. The experiment also measures the performance of the model through five indicators: Accuracy, Detection Rate, False Alarm Rate, Precision, and F1 Score. These indicators are calculated by four values of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). Among them, TP refers to the number of successfully judged correct samples, FP refers to the number of wrong samples judged as correct samples, TN refers to the number of successfully judged wrong samples, and FN refers to the number of correct cases judged as wrong samples. The calculations of the three indicators are shown in Eq. (8) to Eq. (12) respectively.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$DetectionRate = \frac{TP}{TP + FN} \tag{9}$$

$$FalseAlarmRate = \frac{FP}{FP + TN} \tag{10}$$

$$Precision = \frac{TP}{FP + TP} \tag{11}$$

$$F1Score = 2 * \frac{Precision \cdot DetectionRate}{Precision + DetectionRate} \tag{12}$$

4.3 Experimental Results and Analysis

In this chapter, we refer to the proposed full flow adaptive detection of two-level detection method as LSTM-SA.

4.3.1 Related Experiments of LSTM-SA

(1) Best Hidden Layers and Epochs

We run the LSTM-SA through its paces to identify the ideal number of hidden layers and epochs. The experimental data are shown in Tab. 2.

From Tab. 2, we can see that when the hidden layer is set to 2 and the epoch is set to 15, the best accuracy and prediction results have been achieved. The experiment predicts and classifies DDoS attacks with a two-layer structure of 15 epochs, and the success rate is as high as 98.56%, and the minimum RMSE is obtained at the same time. This shows that the model can accurately detect distributed denial-of-service type attacks within slices. Fundamentally, the main factor behind this high rate of successful pre-dictions is the use of LSTMs trained with the correct parameters to solve sequence pre-diction problems. This shows that with the ability of LSTM to store and recall past in-formation, the previous normal or malicious behaviour of network traffic can be effectively used to realize the detection and early warning of ongoing or upcoming malicious attacks.

Table 2 Evaluation experiments on LSTM-SA

Hidden Layers	Epoch	Training Time	Testing Time	RMSE	Accuracy
2	5	4.11	3.90	0.0709	93.68%
	15	12.08	3.16	0.0576	98.56%
	60	26.30	3.17	0.0685	95.31%
4	5	8.94	3.05	0.0800	88.49%
	15	15.89	3.24	0.0765	97.92%
	60	28.00	3.16	0.0677	92.17%
6	5	13.72	3.66	0.0668	89.13%
	15	24.52	3.89	0.0705	95.47%
	60	29.16	3.40	0.0739	91.04%

After searching and debugging, the parameters of the LSTM-SA model are shown in Tab. 3.

Table 3 Some parameters that make the LSTM-SA model achieve the best performance

Hyperparameter category	Value
Network Structure	128 × 128
Initial Learning Rate	0.005
β_1, β_2 and ε in Adam	0.9, 0.999, 10^{-8}
Dropout Probability	0.1
Dropout Random Number Seed	12

The amount of time needed for the model to detect all test data is known as detection time, also known as test time. It has a direct impact on how quickly online future models will respond when used to detect slice escape attacks in real time. Future real-time response times will be quicker the faster the detection time is. The figure demonstrates that the detection time (which is also referred to as the test time) totally meets the standards of real-time detection after being amortized to each record.

(2) Security Value

To increase the stability of the 5G network, the LSTM-SA model can be implemented in the controller of the SDN network to identify DDoS traffic. Security levels S can vary depending on the varied probabilities P of network assaults. The Eq. (13) displays the calculation of the value S.

$$S = 1 - \left(1 - ACC\right) \times P \tag{13}$$

Tab. 4 displays the security value of the LSTM-SA model under various attack probabilities.

Table 4 Security values for different attack probabilities

Probability P	1%	10%	20%	30%	50%
Security level S	99.97	99.86	99.72	99.58	99.30

It can be concluded from Tab. 4 that under different attack probabilities, the security value is higher than 99%. When the DDoS attack probability is 1%, the SDN security value can reach 99.97%, which ensures the reliability of the network.

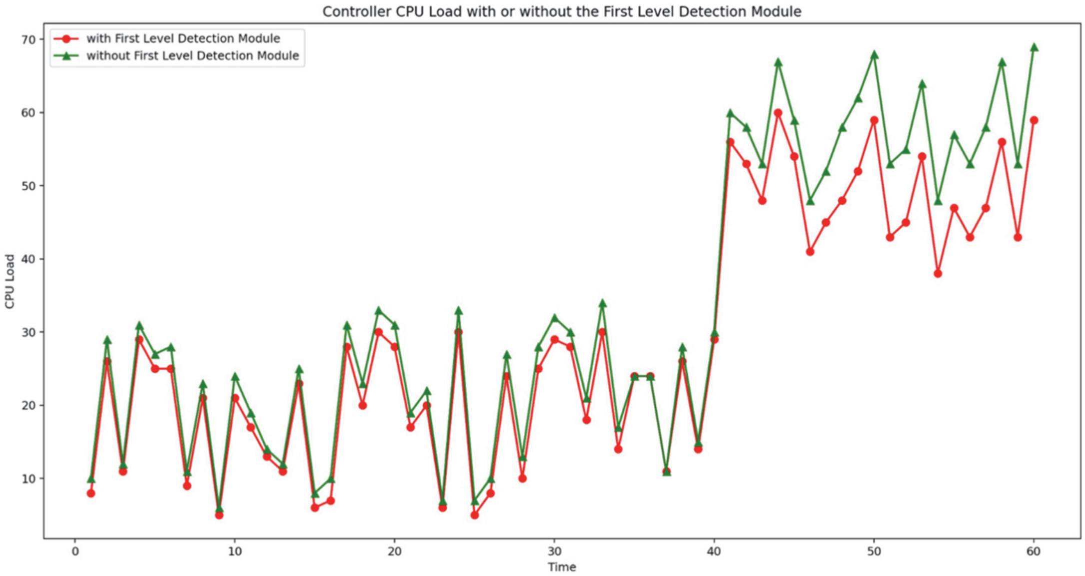

Figure 5 Controller CPU load with or without the first level detection module

(3) The Addition of the First-Level Detection Module Reduces the CPU Load

This study compares the average CPU load of the LSTM-SA without the first level detection module to the average CPU load of the two-level slice escape detection model with the first level detection module. Figure 5 shows the load on the controller CPU for 60 seconds.

In Fig. 5, we can see that a large number of data packets arrived at the 40th second, and the CPU load increased rapidly. Compared with the method without first-level detection, the average CPU load of this method is lower. When the traffic is normal, it is reduced by an average of 5%, and when the traffic is abnormal and the number of data packets increases, it is reduced by an average of 10%. The method first uses information entropy detection to identify network traffic, and when the information entropy is ab-normal, it further detects the traffic. The average CPU load of this method is lower than that of the LSTM-SA method that does not use the first-level detection. Since the calculation of information entropy is less used, it effectively prevents the CPU overload of the controller caused by the slice escape attack.

(4) The Impact of Different Feature Schemes

Combined with the six-tuple feature proposed in literature 32, we use four sets of different feature schemes for experiments, namely:

4-tuple, using the features of the flow in the feature scheme proposed in this paper;

6-tuples, using the feature scheme proposed in Ref. 32;

8-tuple, using the features of the flow and switch association feature proposed in this paper;

11-tuple, adopts the features of the flow, switch association feature and inter-slice escape feature proposed in this paper.

We compared the precision and false alarm rate under the four schemes, as shown in Fig. 6.

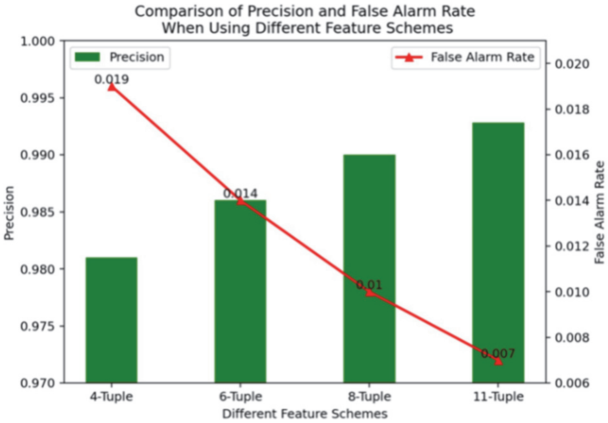

Figure 7 Comparison of precision and false alarm rate when using different feature schemes

As shown in Fig. 6, when compared to the four-tuple and six-tuple features, the model's precision and false alarm rate have both improved significantly after the addition of the switch association feature. The accuracy has been marginally enhanced with the addition of the escape feature between slices, and the false alarm rate has continued to fall. We may deduce from this that the addition of switch association features can assist the model in correcting some deviations, and the inter-slice escape feature can assist in determining inter-slice escape traffic including methods such as IP spoofing.

4.3.2 Comparative Experiments of LSTM-SA

Two models are chosen for this study, and the LSTM-SA that is put out is contrasted and examined using experiments.

Fig. 7 depicts a comparison of the three models' accuracy over various time frames.

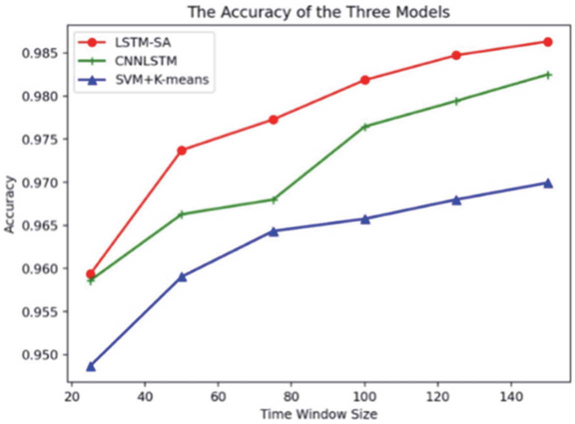

Figure 6 A comparison of the accuracy of the three models for various windows

The accuracy of the three models rises with a greater time frame for data segmentation, as seen in Fig. 7. When the window reaches 150, the accuracy of the LSTM-SA approach is higher than that of the other two algorithms, indicating that the method proposed in this study improves traffic categorization accuracy.

Fig. 8 compares the detection rates of the three models over different time frames.

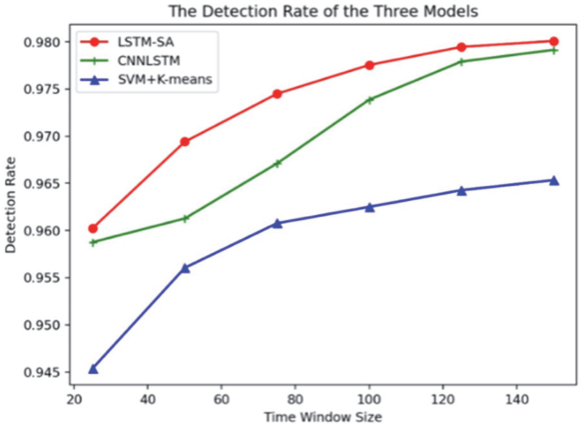

Figure 8 The detection rate indicators for the three models for various windows

As shown in Fig. 8, the performance of the three models in terms of detection rate improves as the time window for segmenting data is extended. The LSTM-SA approach suggested in this research has a detection rate of 98.0% when the window exceeds 150.

Fig. 9 shows the false alarm rate comparison of the three models under different time windows.

As shown in Fig.9, the performance of the three models in terms of false alarm rate decreases as the time frame for data segmentation grows. The method with LSTM and self-attention mechanism presented in this research performs better than conventional machine learning techniques, SVM composite K-Means algorithm, and conventional deep learning algorithm CNNLSTM when the window reaches 150, with a false alarm rate less than 1%.

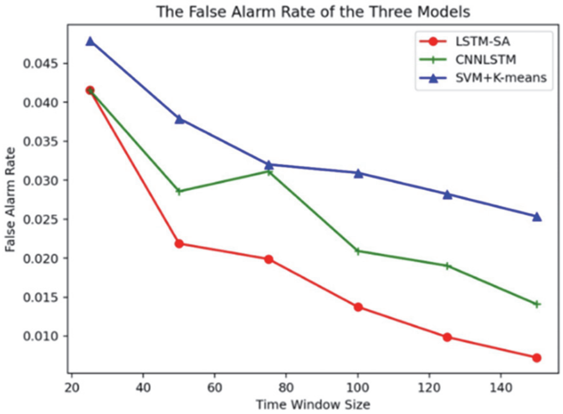

Figure 9 The three models' false alarm rate indicators for various windows

Fig. 10 shows the precision comparison of the three models under different time windows.

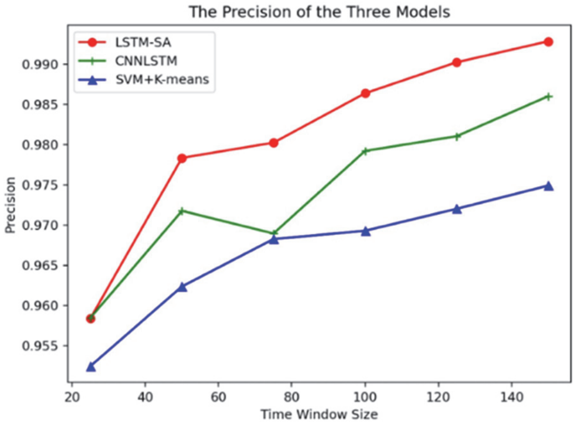

Figure 10 The three models' precision indicators for various windows

Fig. 10 shows that the performance of the three models in terms of precision increases as the time frame for data segmentation grows. The method with LSTM and self-attention mechanism presented in this research performs better than conventional machine learning techniques, SVM composite K-Means algorithm, and conventional deep learning algorithm CNNLSTM when the window reaches 150, with a precision more than 99%.

Fig. 11 shows the F1-score comparison of the three models under different time windows.

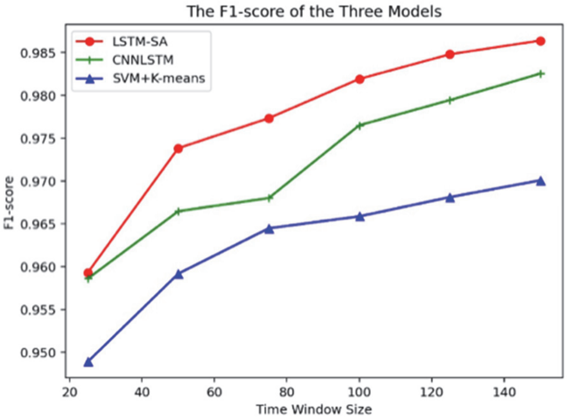

Figure 11 The three models' F1-score indicators for various windows

Fig. 11 shows that the performance of the three models in terms of F1-score in-creases as the time frame for data segmentation grows. The method with LSTM and self-attention mechanism presented in this research performs better than conventional machine learning techniques, SVM composite K-Means algorithm, and conventional deep learning algorithm CNNLSTM when the window reaches 150, with a F1-score more than 98.6%.

It can be seen from the above experiments that when the window is set to 150, the evaluation indicators of the three models have achieved the maximum value.

In summary, we have verified the effectiveness of the two-level detection module and the effectiveness of the 11-tuple feature scheme through experiments, and conducted comparative experiments on the models in five dimensions. After experimental testing and verification, the 11-element feature group scheme we proposed can correct the detection results through the switch correlation feature, and can identify the escaping traffic between slices through the escape feature between slices. Under moderate CPU load, the full-flow adaptive two-level detection approach we presented may actually increase detection accuracy and minimize false positive rate.

5 CONCLUSION

At present, the equipment scale of the 5G power trading private network is constantly expanding, and slice escape attacks are also showing an intensified trend. Aiming at the slice escape attack in the 5G power trading private network, we propose a slice escape behaviour detection scheme based on full-flow adaptive detection. Firstly, based on the "six-tuple" flow table features scheme in the previous literature, we innovatively propose a new set of "eleven-tuple" features to achieve adaptive detection of slice escape behaviours within and between slices. Secondly, we design a two-level slice escape attack detection system, which combines information entropy technology, long-term short-term memory model and self-attention mechanism, and has a high detection rate and low false alarm rate for slice escape behaviour. Thirdly, we design an exception handling module to handle the abnormally detected traffic. For slice escape assaults, the model we present has a high detection accuracy and a low false alarm rate, and the detection latency fits the online detection criteria.

Acknowledgements

The authors would like to thank the China Southern Power Grid Technological Project (Project No. GDKIXM20210107) for their support.

6 REFERENCES

[1] Canadian Institute for Cybersecurity, DDoS evaluation dataset (CICDDoS2019), 2019.

[2] Ghorbani, H., Mohammadzadeh, M. S., & Ahmadzadegan, M. H. (2020). DDoS Attacks on the IoT Network with the Emergence of 5G. *2020 International Conference on Technology and Entrepreneurship-Virtual (ICTE-V)*, 1-5. https://doi.org/10.1109/ICTE-V50708.2020.9113779

[3] Huang, H., Chu, J., & Cheng, X. (2021). Trend analysis and countermeasure research of DDoS attack under 5G network.

2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP), 153-160. https://doi.org/10.1109/CSP51677.2021.9357499

[4] Sattar, D. & Matrawy, A. (2019). Towards Secure Slicing: Using Slice Isolation to Mitigate DDoS Attacks on 5G Core Network Slices. *IEEE Conference on Communications and Network Security (CNS)*, 82-90. https://doi.org/10.1109/CNS.2019.8802852

[5] Chen, X., Chen, Y., Feng, W. et al. (2022). Real-time DDoS Defense in 5G-Enabled IoT: A Multidomain Collaboration Perspective. *IEEE Internet of Things Journal.* https://doi.org/10.1109/JIOT.2022.3218728

[6] Bousalem, B. et al. (2022). Deep learning-based approach for ddos attacks detection and mitigation in 5G and beyond mobile networks. *2022 IEEE 8th International Conference on Network Softwarization (NetSoft)*, 228-230. https://doi.org/10.1109/NetSoft54395.2022.9844053

[7] Elsayed, M. S., Le-Khac, N. A., Dev, S. et al. (2020). Ddosnet: A deep-learning model for detecting network attacks. *2020 IEEE 21st International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 391-396. https://doi.org/10.1109/WoWMoM49955.2020.00072

[8] Li, C. (2020). *Application research on SDN-based DDoS attack detection and defense methods*. Xi'an University of Science and Technology.

[9] Tahmasebi, A., Salahi, A., & Pourmina, M. A. (2021). A Novel Feature-Based DDoS Detection and Mitigation Scheme in SDN Controller Using Queueing Theory. *Wireless Personal Communications*, *4*. https://doi.org/10.1007/s11277-020-07954-3

[10] Cil, A. E., Yildiz, K., & Buldu, A. (2021). Detection of DDoS attacks with feed forward based deep neural network model. *Expert Sys-tems with Applications*, *169*, 114520. https://doi.org/10.1016/j.eswa.2020.114520

[11] Leung, K. C., Li, V. O. K., & Yang, D. (2007). An overview of packet reordering in transmission control protocol (TCP): problems, solutions, and challenges. *IEEE transactions on parallel and distributed systems*, *18*(4), 522-535. https://doi.org/10.1109/TPDS.2007.1011

[12] Zhou, P., Zhou, G., Wu, D. et al. (2021). Detecting multi-stage attacks using sequence-to-sequence model. *Computers & Security*, *105*, 102203. https://doi.org/10.1016/j.cose.2021.102203

[13] Tan, L., Pan, Y., Wu, J. et al. (2020). A new framework for DDoS attack detection and defense in SDN environment. *IEEE Access*, *8*, 161908-161919. https://doi.org/10.1109/ACCESS.2020.3021435

[14] Novaes, M. P., Carvalho, L. F., Lloret, J. et al. (2020). Long short-term memory and fuzzy logic for anomaly detection and mitigation in software-defined network environment. *IEEE Access*, *8*, 83765-83781. https://doi.org/10.1109/ACCESS.2020.2992044

[15] Li, Y. & Lu, Y. (2019). LSTM-BA: DDoS detection approach combining LSTM and Bayes. *2019 Seventh International Conference on Advanced Cloud and Big Data*, 180-185. https://doi.org/10.1109/CBD.2019.00041

[16] Yuanping, L., Hua, L., Junlan, Z. et al. (2018). Research on IPv6 attribute testing of Open Flow protocol. *Computer Engineering and Science*, 1757-1765.

[17] Elayoubi, S. E., Jemaa, S. B., Altman, Z. et al. (2019). 5G RAN slicing for verticals: Enablers and challenges. *IEEE Communications Magazine*, *57*(1), 28-34. https://doi.org/10.1109/MCOM.2018.1701319

[18] Wijethilaka, S. & Liyanage, M. (2021). Survey on network slicing for Internet of Things realization in 5G networks. *IEEE Communications Surveys & Tutorials*, *23*(2), 957-994. https://doi.org/10.1109/COMST.2021.3067807

[19] Mehr, S. Y. & Ramamurthy, B. (2019). An SVM based DDoS attack detection method for Ryu SDN controller. *Proceedings of the 15th international conference on emerging networking experiments and technologies*, 72-73. https://doi.org/10.1145/3360468.3368183

[20] Gu, Y. H., Li, K. Y., Guo, Z. Y. et al. (2019). Semi-supervised K-means DDoS detection method using hybrid feature selection algorithm. *IEEE Access*, *7*, 351-365. https://doi.org/10.1109/ACCESS.2019.2917532

[21] Chen, Z., Jiang, F., Cheng, Y. J. et al. (2018). XGBoost classifier for DDoS attack detection and analysis in SDN-based cloud. *IEEE International Conference on Big Data & Smart Computing*, 251-256. https://doi.org/10.1109/BigComp.2018.00044

[22] Dong, S. & Sarem, M. (2019). DDoS attack detection method based on improved KNN with the degree of DDoS attack in soft-ware-defined networks. *IEEE Access*, *8*, 5039-5048. https://doi.org/10.1109/ACCESS.2019.2963077

[23] Popovski, P., Trillingsgaard, K. F., Simeone, O. et al. (2018). 5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view. *IEEE Access*, *6*, 55765-55779. https://doi.org/10.1109/ACCESS.2018.2872781

[24] Barakabitze, A. A., Ahmad, A., Mijumbi, R. et al. (2020). 5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges. *Computer Networks*, *167*, 106984. https://doi.org/10.1016/j.comnet.2019.106984

[25] Sattar, D. & Matrawy, A. (2019). Towards secure slicing: Using slice isolation to mitigate DDoS attacks on 5G core network slic-es. *2019 IEEE Conference on Communications and Network Security (CNS)*, 82-90. https://doi.org/10.1109/CNS.2019.8802852

[26] Thantharate, A., Paropkari, R., Walunj, V. et al. (2019). DeepSlice: A deep learning approach towards an efficient and reliable network slicing in 5G networks. *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, 0762-0767. https://doi.org/10.1109/UEMCON47517.2019.8993066

[27] Jia, Y., Zhong, F., Alrawais, A. et al. (2020). Flowguard: An intelligent edge defense mechanism against IoT DDoS attacks. *IEEE Internet of Things Journal*, *7*(10), 9552-9562. https://doi.org/10.1109/JIOT.2020.2993782

[28] Zhang, S. (2019). An overview of network slicing for 5G. *IEEE Wireless Communications*, *26(*3), 111-117. https://doi.org/10.1109/MWC.2019.1800234

[29] Kuadey, N. A. E., Maale, G. T., Kwantwi, T. et al. (2021). DeepSecure: Detection of Distributed Denial of Service Attacks on 5G Network Slicing Deep Learning Approach. *IEEE Wireless Communications Letters*, *11*(3), 488-492. https://doi.org/10.1109/LWC.2021.3133479

[30] Onoja, D., Hitchens, M., & Shankaran, R. (2022). *DDoS Threats and Solutions for 5G-Enabled IoT Networks. Secure and Trusted Cyber Physical Systems*. Springer, Cham, 115-133. https://doi.org/10.1007/978-3-031-08270-2_5

[31] Niu, Y., Feng, G., Li, Y. et al. (2022). MTC Slice Mapping under DDoS Attack in 5G RAN. *2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, 588-591. https://doi.org/10.1109/IPEC54454.2022.9777300

[32] Braga, R., Mota, E., & Passito, A. (2010). Lightweight DDoS flooding attack detection using NOX/OpenFlow. *IEEE Local Computer Network Conference*, 408-415. https://doi.org/10.1109/LCN.2010.5735752

Contact information:

Zhenzhen LIU, PhD
Information Data Department,
Guangdong Power Exchange Center Co. Ltd.,
Guangdong 510080, China
E-mail: clz0502@163.com

Rui ZHOU, Senior Engineer
Information Data Departmnt,
Guangdong Power Exchange Center Co. Ltd.,
Guangdong 510080, China
E-mail: zhourui@gd.csg.cn

Jingbing CHEN
A Key Laboratory of Trustworthy Distributed Computing and Service,
Beijing University of Posts and Tele-communications,
Beijing 100876, China
E-mail: jingbingchen@bupt.edu.cn

Kangqian HUANG, Senior Engineer
Information Data Department,
Guangdong Power Exchange Center Co. Ltd.,
Guangdong 510080, China
E-mail: huangkangqian@gd.csg.cn

Jingyin HUANG, Engineer
Information Data Department,
Guangdong Power Exchange Center Co. Ltd.,
Guangdong 510080, China
E-mail: 8103334798@qq.com

Binsi CAI, PhD
(Corresponding author)
A Key Laboratory of Trustworthy Distributed Computing and Service,
Beijing University of Posts and Tele-communications,
Beijing 100876, China
E-mail: caibinsi@bupt.edu.cn

Yali GAO, PhD
A Key Laboratory of Trustworthy Distributed Computing and Service,
Beijing University of Posts and Tele-communications,
Beijing 100876, China
E-mail: gaoyali@bupt.edu.cn

Kaiguo YUAN
A Key Laboratory of Trustworthy Distributed Computing and Service,
Beijing University of Posts and Tele-communications,
Beijing 100876, China
E-mail: flyingdreaming@bupt.edu.cn