IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 31 January 2024, accepted 4 March 2024, date of publication 18 March 2024, date of current version 22 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3376413

RESEARCH ARTICLE

A Human-in-the-Loop Anomaly Detection Architecture for Big Traffic Data of Cellular Network

SHENGLONG LIU, YUXIAO XIA, AND DI WANG
Big Data Center, State Grid Corporation of China, Beijing 100012, China

Corresponding author: Di Wang (leeyz0925@vip.qq.com)

This work was supported by the Science and Technology Project of Big Data Center of State Grid Corporation of China under Grant SGSJ0000HGJS2310029.

ABSTRACT In the era of mobile big data, smart mobile devices have become an integral part of our daily life, which brings many benefits to the digital society. However, their popularity and relatively lax security make them vulnerable to various cyber threats. Traditional network traffic analysis techniques utilizing pattern matching and regular expressions matching algorithms are becoming insufficient for mobile big data. Network traffic anomaly detection is an effective method to replace traditional methods. Network traffic anomaly detection can solve many new challenges brought by future network and protect the security of network. In this article, we propose a streaming network framework for mobile big data, referred to as SNMDF, which provides massive data traffic collection, processing, analysis, and updating functions, to cope with the tremendous amount of data traffic. In particular, by analyzing the specific characteristics of anomaly traffic data from flow and user behavior, our proposed SNMDF demonstrates its capability to offer real data-based advice to address new challenges for future wireless networks from the viewpoints of operators. Tested by real mobile big data, SNMDF has proven its efficiency and reliability. Furthermore, SNMDF is accessed for the digital twin of the space Internet, which validates that it can be generalized to other environments with massive data traffic or big data.

INDEX TERMS Cyber threats, network traffic, network security, big data, SNMDF.

I. INTRODUCTION

With the development of AI and mobile communication technologies, 5G networks facilitate the appearance of new mobile applications (e.g., live video, short-form video, VR and AR). New technologies enhance the quality of service and user experience, that makes users more reliant on mobile networks. On the flip side, it also makes smart mobile devices and systems more vulnerable to attackers. Attackers can gain unauthorized access to user information and computing resources by attacking mobile devices. The quality management of mobile networks and the protection of user information become an important issue. The main method used to detect malicious network attacks is network traffic-based suspicious behavior identification [1], [2]. Currently, there are two main types of research in the analysis of malicious attacks on network traffic. The first is a signature-based method that takes the information from the network traffic to discover signatures [3] of malicious attacks. The second is an anomaly-based method that employs anomaly-checking techniques [4] in network traffic logs. However, as every mobile phone generates network traffic data in real-time, this results in a huge amount of network traffic data being generated on the Internet Service Provider(ISP)'s backbone network. The massive traffic data becomes a huge challenge to ISPs in identifying malicious applications on the network. Firstly, there is a competing relationship between detection time and detection accuracy. To complete detection rapidly, the number of features and algorithm complexity is limited. It is necessary to extract deep statistical information about the network traffic to improve the accuracy (e.g., flow duration, number of

The associate editor coordinating the review of this manuscript and approving it for publication was Qilian Liang.

requests, frequency of requests, etc.), however, this inevitably increases the completion time. Secondly, internet traffic is complex and difficult to track. The traffic data generated by all the applications/websites are mixed, and the traffic sequences are transmitted randomly and alternately, which results in a matching difficulty between applications/websites and their traffic [5]. Therefore, it is very difficult to measure an application/website traffic integrally. Thirdly, network traffic anomaly detection represents an inherently imbalanced classification problem. In order to validate the effectiveness of the detection algorithm, there must be an amount of labeled anomalous traffic samples to cover different types of intrusions and attacks. Whereas, the labeled anomalous traffic sample is a rarity, which represents a very small percentage of the entire dataset especially in the massive amount of network traffic data from ISPs. In addition, future space internet is difficult to defend. The long-distance of space transmission and the limited computing and storage capabilities affected by the light-speed delay effect and lost frames and packets make it challenging to apply the current high reliability and high strength encrypted transmission method of space communication transmission like terrestrial communication. This leads to the fact that the existing defense methods for terrestrial networks are not fully applicable to the defense of space networks.

To solve the four above challenges, we propose a human-in-the-loop architecture based on network traffic for efficient and fast detection of malicious applications on the network backbone of ISPs. The detection architecture is designed to face the small sample size and the complex change in the current real network. Deep Packet Inspection (DPI) data is recognized as the most effective data for traffic classification, traffic control, and anomaly detection by ISPs [6], [7]. The human-in-the-loop anomaly detection architecture in this paper uses DPI data as the training data and application scenario. Due to the large size of DPI data, the detection architecture is divided into two phases (offline and online). We train the anomaly detection model offline and then deploy the model to a distributed online streaming system. Furthermore, the architecture extends the anomaly detection algorithm to a ''human in the loop'' system that is used to iteratively improve the accuracy of the algorithm through continuous human verification of the results. It is absolutely necessary to update the anomaly detection model with the latest data.

The contributions of our work are summarized as follows:

1. We propose a Streaming Network Malicious Application Detection framework (SNMDF) that can handle massive amounts of traffic data in real time and performs well in real-world network traffic environments. This framework uses traditional anomaly detection algorithms differently from existing deep learning frameworks. The deep learning framework requires a large amount of instance data. The network traffic data has inherent problems such as uneven features and abnormal sample shortage. The traditional algorithm selected is more suitable for large-scale network traffic modeling and parallelization. Through the actual network traffic environment experiment, the feasibility of machine learning algorithm in the era of big data network traffic is verified.

2. Based on the URL field in DPI data, a multi-model fusion anomaly detection method for mass data is provided. The method uses a machine learning approach to discover malicious application messages on the limited DPI data and ensures improved efficiency and accuracy simultaneously. Moreover, the method can also be used as a defense measure for the virtual world, and continually improve in the digital twin environment.

3. SNMDF was applied to a 5-day mobile Internet traffic dataset from a province in China with detailed statistical analysis on malicious messages. The method can detect performance characters of malicious applications in real network traffic. This is the first time that long-time traffic statistics characteristics of network abnormal traffic are given for real carrier-grade massive network traffic. The analysis results provide helpful support to ISPs for traffic governance, diagnostic monitoring, and security protection.

II. RELATED WORK

A. NETWORK TRAFFIC MANAGEMENT

As the size of wireless network traffic grows continuously, network traffic management becomes a hot research topic in the field of wireless communications. Network traffic management can help network managers achieve intelligent control of network traffic, classification, and anomaly detection. The work in [8] proposes an evolutionary game-based distributed approach to optimal configuration selection for upLink-downLink (UL-DL), which minimizes interference and maximizes system throughput. The authors of [9] designed an intelligent control algorithm for network traffic. The method is based on label-free learning in edge clouds by exploiting the limited computational and storage resources. The algorithm can evaluate the value of data uploaded by edge cloud devices. The research in [10] presents a new learning-based approach to wireless resource management. The core idea is to consider the input and output of a resource allocation algorithm as an unknown non-linear mapping and to approximate it using a deep neural network (DNN). Liu et al. [11] proposed a big data analysis framework to analyze user click requests, mobile terminals and behavior in ISPs' network traffic, which expands the scope of ISPs' traffic experience. Alzahrani and Hong [12] use computational intelligence and a majority vote-based ensemble approach to propose a system for detecting known and unknown Distributed Denial of Service (DDoS) Attacks. Their proposed system applies two different intrusion detection approaches anomaly-based distributed artificial neural networks(ANNs) and signature-based approach. Chen et al. [13] divide each network traffic into successive message segments. Each segment is represented as relatively independent content transmitted between the server and the

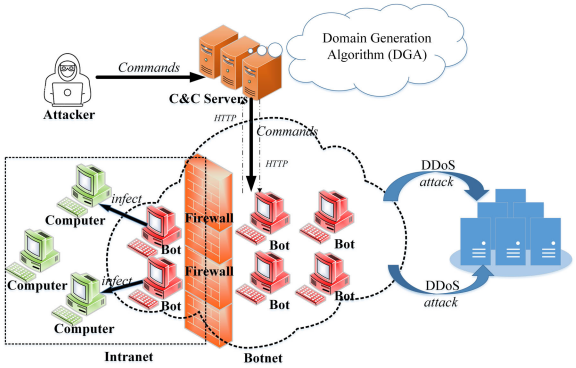


FIGURE 1. Network intrusion attacks using HTTP.

client. The sequence characteristics of different source applications are discovered through data analysis of the message sequences and then classify encrypted applications' traffic according to the sequence characteristics. Dusi et al. [14] propose a statistical traffic classification mechanism that uses packet size, arrival orders and packet intervals to train the production of offline ''fingerprints'' to classify HTTP or SSH.

B. THE SECURITY OF NETWORKS

The development of communication technology, the Internet and artificial intelligence significantly changes the traffic model and service architecture, which makes network security with new challenges and attracts more public attention [15]. Oh et al. [16] proposed a signature intrusion detection method for IoT systems using multiple pattern-matching algorithms. The accuracy of the method can achieve 81% to 90% on different experimental datasets. Anthi et al. propose a three-layer intrusion detection system (IDS) that uses a supervised approach to detect a range of popular cyber attacks on the Internet of Things. In another study, Ye et al [17] used a self-encoder to reconstruct the original input data to extract new deep features. They perform variable-weighted feature selection of deep features to achieve compression while reducing the bias and computational complexity of machine learning models. This method can achieve good results on WiFi network intrusion detection data. Aminanto et al. [18] use an ant colony clustering approach to find communication patterns to discover the characteristics of botnet traffic. Azmoodeh et al. [19] extracted code graphs of executable files for manipulation and then trained convolutional neural networks (CNNs), which were eventually used to identify malicious applications. Goldenberg and Wool [20] modeled the network traffic behavior of the Modbus protocol to perform intrusion detection, which can achieve an accuracy of 65% to 99%. Aminanto et al. [18] proposed a deep feature extraction and selection model using a deep self-encoder that combines the auto-encoder with a supervised classification algorithm. It can improve the detection rate of attacks to 99.918%.

III. PROBLEM STATEMENT

With the development of network applications, network traffic becomes complex and diverse, including web traffic, video traffic, application traffic, etc. HTTP (Hypertext Transfer Protocol) is a data transfer protocol for transmitting World Wide Web documents over the Internet, which functions as a request–response protocol between Web browsers and World Wide Web servers. Nowadays HTTP has become one of the most widely used network protocols on the Internet, for both web browsers and mobile apps to request data. Meanwhile, it also become a widely used network protocol for cyber attacks, network intrusion and malware. Network attacks, intrusions, malware, and other anomalies are upgraded with the development of technology iteratively. They become more automated and intelligent by simulating the behavior of normal traffic. A variety of automatic URL generation algorithms are used by anomalies to constantly change URL addresses that can evade network detecting and blocking. For example, the C&C attack (command-and-control attack), as shown in Fig.1, is one of the most destructive malicious cyber-attacks and has a continuous increase over the past decade. Attackers take control of a large number of bot servers through phishing email spoofing, executing malicious code, or exploiting browser security plug-in vulnerabilities. They establish an HTTP-based communication connection with the bot servers through C&C servers and execute the commands issued by the attacker. The bot server can even be hosted inside the corporate network so that attackers can break firewalls and infect internal hosts. C&C servers deploy the Domain Generation Algorithm (DAG), which can generate numerous domains to evade discovery and blocking. Hackers can use a botnet to initiate data theft, DDoS attacks [21], APT attacks and other malicious acts. Therefore, the variable HTTP domains can be treated as the main traffic characteristic that distinguishes C&C servers from normal servers. The goal of this paper is to save manual detection costs by using URLs as the main feature of traffic analysis. We discover hidden malicious behaviors in traffic, develop an anomaly detection algorithm with good real-time performance, parallelize the algorithm, and apply it to massive amounts of real network traffic anomaly detection. We also present a manual verification architecture to continuously increase the number of anomalous samples, feed them back to the algorithm for training, and continuously expand the algorithm's feature selection to discover more traffic features that can detect malicious behavior.

IV. PROPOSED FRAMEWORK

A. DATA

Most previous studies mainly used data collected by company edge devices or by university campuses [13], which is not able to reflect the real state of the network. Simultaneously, some aged public datasets are not suitable for today's network traffic analysis. To cope with the limitation, we consider a more generic application scenario. When Mobile users

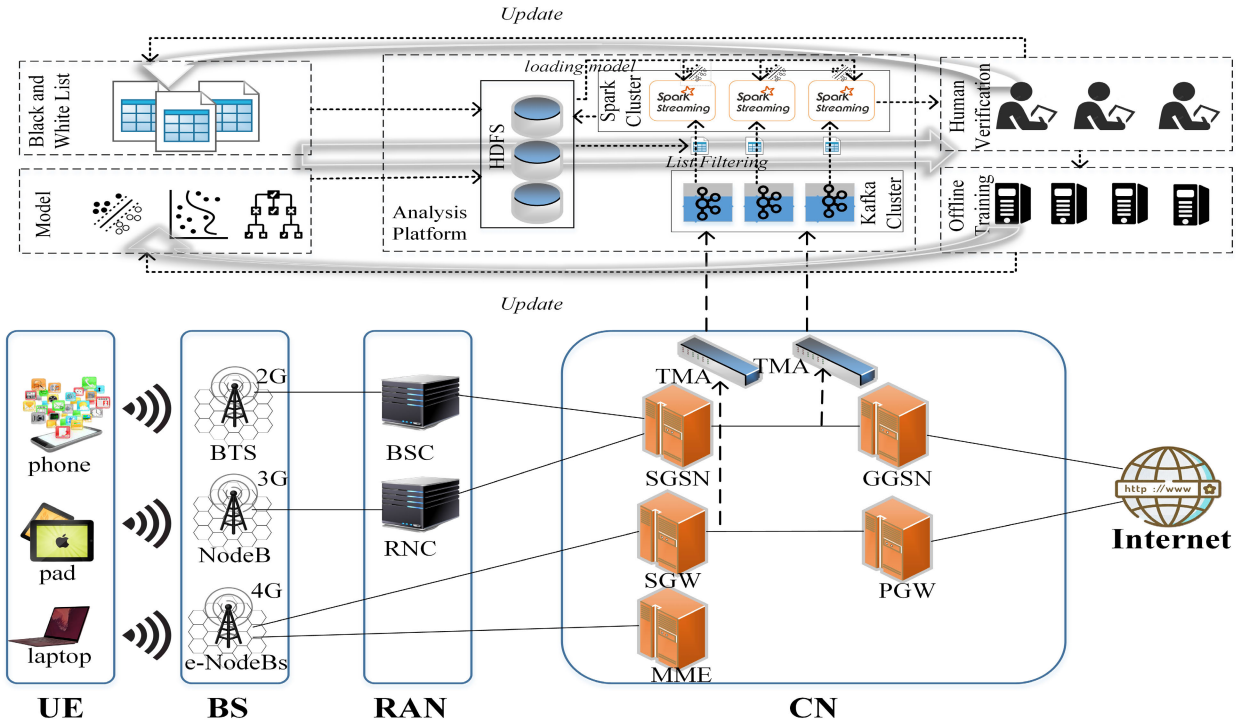


FIGURE 2. System architecture.

access the Internet, a large amount of traffic data is generated in the ISP core network. We capture these data traffic as experimental data and application scenarios. Our data is based on 2G/3G/4G mobile data provided by an operator that serves over 50% of China's mobile internet users. We mirror all uplink and downlink packets from mobile users, capturing HTTP packets and combining them into streams based on 5 components (IP source address, IP destination address, source port number, destination port number, transport protocol). We extract over 90 valuable features from stream data, including user ID, start time and end time of each flow, ID of base station user connected with, uplink and downlink bytes, number of uplink and downlink packets, and so forth. In particular, flows over the HTTP contain the user's access host and URL (Uniform Resource Locator), which is a link to the web page and is an indirect representation of the contents of the request. This gives us the opportunity to crawl and analyze the URLs visited by users to understand their browsing behavior. To protect personal privacy, private user information in the data is replaced by hashed numbers that can be used to identify the user without affecting the effectiveness of our analysis.

B. FRAMEWORK

In this section, we present the architecture of our proposed network traffic monitoring and analysis system. The system is a Human-in-the-Loop Architecture, which takes feedback of human certification results into a black and white list and supports continual model updating by offline training. As shown in Fig.2, user devices (including smartphones, tablets, laptops, etc.) link to the Internet by accessing base stations in the mobile network, generating large amounts of network traffic data. There are multiple networks in China's mobile network (2G, 3G, 4G, 5G). However, due to the limited number of users on the 5G network, we only analyze the user data on the 2G, 3G and 4G networks. In 2G and 3G networks, a Base Transceiver Station (BTS) or Node B transmits the user requests to the Base Station Controller (BSC) or Radio Network Controller (RNC). The controller (BSC/RNC) then sends the network traffic to the Serving GPRS Support Node (SGSN), which is connected to the Gateway GPRS Support Node (GGSN) via the Gn interface for packet routing and data forwarding. The gateway GPRS support node converts GPRS packets from the SGSN into a suitable packet data protocol for transmission to an external network, such as the Internet. In 4G networks, evolved Node B (eNodeB) is responsible for the connection establishment between the terminal (UE) and the mobile management entity (MME). The user's network data traffic is routed through the Serving GateWay (SGW) and Packet Data Network GateWay (PDN) gateways to the Internet.

Traffic Monitoring and Analysis (TMA) equipment is in charge of deep packet inspection, flow record composition, and basic packet statistics. TMA is deployed in the Core Network (CN) to collect traffic data generated by User Equipment. Two kinds of input are received from the traffic monitor via Gigabit Ethernet: flow record frames and specific mirrored packets. With these inputs, the TMA

performs three functions: extracting flow records from flow record frames, generating application-layer session records, and uploading records to the Data Store. Network traffic is obtained in mirroring mode, which does not affect actual service transmission, and can be adapted to various application scenarios, such as power big data networks and enterprise internal networks. To enable efficient real-time traffic anomaly detection, the collected traffic is sent to a Kafka cluster for processing. Kafka provides a message queue, which allows producers to send the message to a specific topic partition order and then consumers view the messages in orders. Kafka clusters handle massive amounts of high-speed traffic data with low processing latency and server failure performance. It also ensures efficient storage and secure transmission of traffic data. Kafka clusters are a common real-time processing framework that can receive data from multiple links within the core network (Kafka clusters can act as data processing centers to receive data from multiple links in multiple provinces and cities). High-speed and massive traffic data is written by the Kafka cluster to a specified queue and sent to Spark streaming. Spark streaming receives network traffic data as the consumer of the message. Spark Streaming is a major library that leverages fast scheduling capabilities to perform streaming data analysis. Incoming streaming network traffic is encapsulated by Spark Streaming into mini-batches in specified periods. For each mini-batch, Spark Streaming first loads the black-and-white list from the Hadoop Distributed File System (HDFS) into the cluster memory and filters the URLs. After that, anomalous traffic will be identified by Spark Streaming by loading a model that is trained offline. The accuracy of the offline trained prediction model will constantly change with the network traffic. To continuously maintain the accuracy of the training model, it is necessary to constantly update the pre-trained model. The detected anomalous traffic is sent to humans for validation, allowing the model to be updated to include human judgment results. The final validation results are used as the standard set for the model's offline training model. At a fixed interval, all pre-trained models are updated and saved to HDFS for Spark Streaming to load.

C. ANOMALY DETECTION METHOD

Offline training uses manually authenticated anomaly data and historically accumulated black and white list data as the training dataset. Since in the network of malicious behavior discovery is an inherent problem of data imbalance (anomalous data is extremely rare), the extreme imbalance of the data sample will have an impact on the final training effect of the model. Therefore, this paper uses a sampling technique to mitigate the problem of data imbalance. Firstly, we reduce the sample size of the normal traffic sample. Secondly, we only sample the URLs of the top-ranking domains [22]. Then we update malicious traffic data after a manual verification to the blacklist continuously, accumulating additional anomalous samples for category data balancing. As the traffic data captured in the operator's backbone network is mixed with multiple users and applications, the data traffic records are mixed and non-sequential. So, in order to meet the real-time requirements, we follow two principles in feature selection:

1) fully reflect the user behavior.

2) the statistics can be calculated on a separate traffic record, without the need to remember the correlation linkage.

Based on the above principle, each sample consists of two types of features: URL features and traffic behavior features.

1) URL FEATURES

Generally, URLs are the most important feature of network traffic data in terms of user behavior. URLs consist of various types of characters containing fixed structural patterns and keywords that normally reflect the nature and type of resources accessible on the remote server. We extract the semantic and statistical characteristics of URLs, including URL character length, domain character length, whether they contain IP addresses, whether they are top-level domains, number of meta characters, number of consonant characters, number containing special characters, etc. (as shown in Table1), with the aim of discovering the structural model of normal URLs.

2) TRAFFIC BEHAVIOUR FEATURES

Traffic behaviour characteristics are a wealth of information about networks, applications, and users extracted by TMA devices from mobile network data traffic. The TMA device captures raw packets on the 10 Gb/s link of the core network, synthesizes them into stream records based on TCP/IP five-tuple groups and generates application layer session records. For each stream record, more than 90 characteristics can be generated, including user ID, current time, location, number of upstream and downstream bytes, device type, protocol type, URL, etc. We select the user's access time, access duration, http_version, object_type, content_length, etc. (as shown in Table1) as the traffic behavior input features for our algorithm. We choose a more efficient inference model that is suitable for parallel loading in the Spark framework rather than a deep learning model to satisfy real-time network anomaly detection under high-speed and massive network traffic. To solve the problem of unbalanced data, we use integrated learning algorithms by combining multiple classifiers to achieve good results. For the base classifier, we use GBDT, RandomForest, DecisionTree, and logistic regression [23]. Due to the characteristics of network traffic, such as uneven characteristics, small anomaly sample size and large extreme value, deep learning algorithms are not suitable for network traffic anomaly detection. GBDT, RandomForest, and DecisionTree have relatively few parameters to tune and are less expensive regarding computational resources [24]. These algorithms can split the data set into subsets to parallelize the algorithm. Logistic regression divides each iterative process into independent computational steps to achieve algorithmic parallelism. All

TABLE 1. URL character and traffic behavior characteristics.

Character classification	Name	Description
URL character	URL String length	The number of characters in the user request URL after removing the first part of the protocol after the first part of the protocol, e.g.http://example.com/S985.apk has a character length of 17
	Domain String length	Length of domain in URL
	IP exist	Whether URL contains IP address or not e.g. http://219.238.7.67/files/s1.apk contains IP address
	TLD or not	The domain of URL is TLD e.g. ".com",".org",".net",".cn" etc.
	Number of vowel	The number of vowel in URL
	Number of consonant	The number of consonant in URL
	Number of special character	The number of special character like '%','#','$', e.g. http://example.com/E6\%83\%B2.apk 2 special character
	Protocol	Protocol type
	Port number	16-bit unsigned integer e.g. "8080"
Traffic behavior characteristics	Start time	Flow start time
	End time	Flow end time
	http_version	Defines a version of the HTTP protocol
	object_type	Identifies the content of HTTP, including text or pictures, videos, applications, etc.
	content_length	Indicate the size of entity-body
	Response Time	Link establishment response delay
	ACK Time	Link establishment confirmation delay

the above models are implemented in the spark mllib [25] library for parallel computing and are more suitable for the Spark framework. In algorithm integration, we design a voting mechanism with adjustable vote thresholds, which can be automatically adjusted by the threshold value of the manual verification results. Although the analytical model does not use a deep learning model, we introduce a validation session with manual participation. This approach can satisfy real-time requirements through fast model training updates and real-time inference, while reducing the target range and improving the efficiency of manual validation. The algorithm is feasible through the analysis of the framework in the network environment of a real operator, and provides a theoretical reference that the traditional anomaly detection algorithm is capable of applying on mass data.

D. KEY POINT OF THE FRAMEWORK

As shown Fig.2, the overall architecture of the framework is designed to collect, store, process, manage the data traffic of 2G/3G/4G networks and detect the abnormality. To operate in a secure, stable and efficient production environment, our SNMDF analysis framework needs to meet the following characteristics:

1) PERFORMANCE

Based on Apache Spark streaming and Kafka, SNMDF can process the analytic job of mobile big data in a streaming way. The SNMDF framework supports traffic customization in different regimes, automatic model and black and white list updates, job schedules automatically by the cluster, allocating computing resources to running jobs according to their importance.

2) PORTABILITY

In our case, we mainly use the SNMDF framework for network anomalous behavior traffic detection. However, our architecture and applications work independently of stored data and specific analytics. Therefore, the framework can be used for other high real-time mass data analysis application scenarios.

3) SCALABILITY

The storage and computing power of the SNMDF framework can easily be increased or decreased by adding or removing machines or computing resources from the cluster. Through detection, it is found that the traditional manual verification of a network flow takes 1-2 seconds. In the face of terabyte-level massive network data, the efficiency of manual verification is extremely low. Our framework uses server cluster distributed storage and parallel computing to greatly reduce the detection time and achieve millisecond-level detection of a single network flow, with an efficiency of 10-20 times that of a single computer, and which is much higher than the traditional manual verification efficiency. Additionally, the redundancy and security of applications and data can be improved by centralized horizontal scaling.

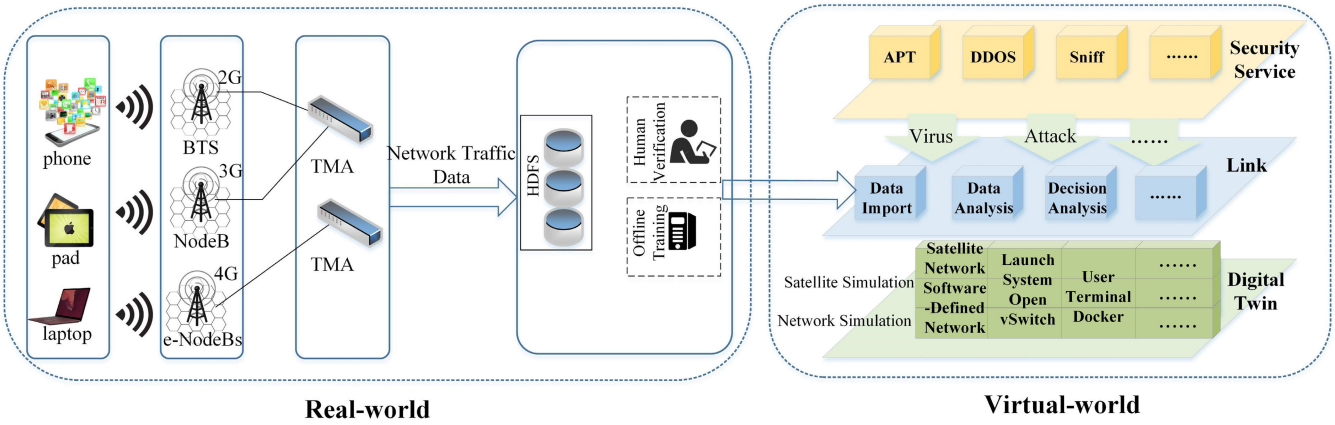


FIGURE 3. Overall framework of the experimental platform.

4) AVAILABILITY

The SNMDF framework can be continuously adapted as the complexity of the network environment changes, adjusting model parameters and model selection through manual validation and manual discovery. For the telecoms-grade application environment, manual verification is necessary to ensure quality of service. The SNMDF framework utilizes manual verification sufficiently, balancing the verification efficiency and recognition accuracy.

V. EXPERIMENTS AND ANALYSIS

The experimental platform consists of two core units. One unit is deployed into the core network of a largescale cellular network to monitor network traffic. The other is accessed into the hyper large scientific Infrastructure based on digital twin for the space internet. The real world and the virtual world influence and promote each other, as shown in Fig.3. To reduce the space Internet network security risk, we in the protection of the existing network security protection system based on the normal and effective operation, from the global perspective, proposed for the space Internet device architecture, and named "Space Spider" [26], which can realize all-element ground simulation, technical verification, attack and defense exercises of the space Internet. The technical architecture of the Space Spider is composed of a digital twin layer, link layer, and security service layer, which ultimately forms a highly integrated space Internet attack and defense simulation and verification platform, and it provides strong support for the sustainable development of the space Internet. The digital twin layer provides infrastructure support for the safety and security services of the entire space Internet hyper large scientific infrastructure "Space Spider", including digital space, the overall network environment and the network shield. The digital twin of the overall network environment aims to unify the dispatch of available resources in different network nodes. The network defense adopts the active defense architecture based on deploying traps in the system to deceive. The traps are set based on anomalies detected in normal network traffic. The link layer is the interface between Space Spider's security services and the overall infrastructure of the space Internet, including data import, data processing and equivalence verification. The data import layer collects data through built-in devices, such as network traffic data monitored in the real world. The security service layer is suitable for scientific research, competition, real-world confrontation and emergency exercises.

The real-world confrontation can support a variety of real-world security policies. The intrusion exercise supports simulated intrusion attacks on specific scenarios and executes attacks from various entry points within a limited time frame. In our experiment, we collected mobile traffic data between March 9th (Monday) and March 13th (Friday), 2021 from a western city in China. We collected flows of 257683 users, they generated 695.21 TB traffic and 73.87×10^{10} flows in total. Each user was online for 2.2 hours on average in 5 days (the duration between the start and end time of a flow is defined as online duration of user who generated this flow), generated 10.75 MB traffic and 10.36×10^3 flows in average per day. With these 5-day data sets, the proposed model is examined to detect anomaly traffic, and then the anomaly traffic is confirmed by experts. Finally, we confirmed 59.04 GB of abnormal traffic that involved 102961 users from the 5-day dataset.

A. CHARACTERISTICS OVERVIEW OF ANOMALY TRAFFIC DATA

To have an overview of the daily feature of anomaly traffic, we analyze a time-series graph of flow metrics (number of flows, flow bytes, the amount of users and online duration) with 5 minutes time granularity in 5 days, and obtained a periodogram of the time series, as shown in Fig.4-7.

1) NUMBER OF FLOWS

As can be seen in Fig.4, Number of flows shows obvious daily cycle characteristics. The network traffic is gradually

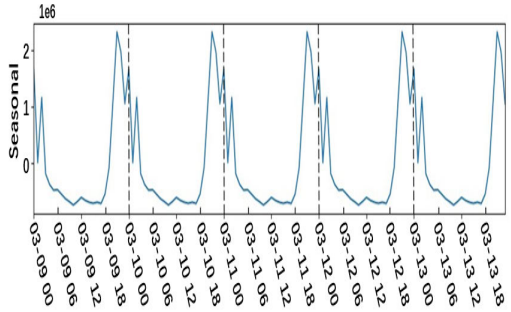


FIGURE 4. Periodogram of the number of flows.

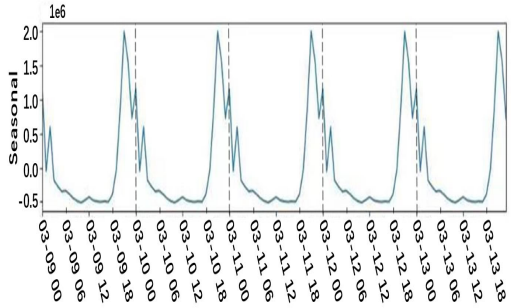


FIGURE 5. Periodogram of the flow bytes.

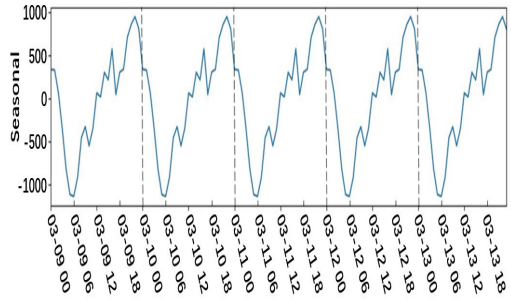


FIGURE 6. Periodogram of the amount of users.

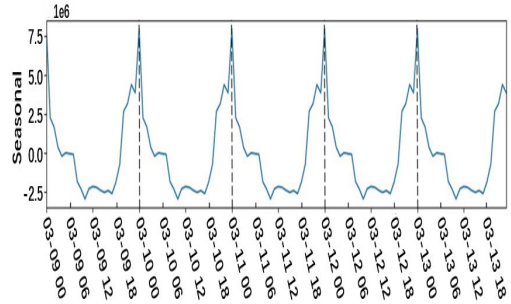


FIGURE 7. Periodogram of the online duration.

high from 2:00 am to 5:00 am and reaches its peak around 5:00 am. In addition, there were two peaks in the evening hours of March 11 and March 12. In Fig.4, we found that from March 9th to March 13th, from 22:00 to 5:00 the next morning, the number of traffic was at least 2×10^6.

2) FLOW BYTES

By observing the number of stream bytes (Fig5), we found that the Flow bytes also exhibit the characteristics of a daily cycle. There were still two peaks on the evenings of March 11 and March 12. From the figure, we can see that the Trend and Periodogram distribution of Flow bytes is similar to the Number of flows, but the peak value of Flow bytes reaches 6.8×10^9 Byte.

3) THE AMOUNT OF USERS

In the time series distribution diagram of the number of users from March 9th to March 13th (Fig.6), we can see the number of abnormal users has a minimum value around 5:00 am every day. After that, it began to gradually increase. After 22:00, the number of abnormal users was significantly more than the number during the day.

4) ONLINE DURATION

From the time series chart and cycle chart of duration, we can see that the online duration of abnormal users reached a peak in the early morning of March 11 due to the increase in the total number of infected users on March 11 within 5 days. In addition, it can be seen from Fig.7 that when the evening time of each day comes, the duration value of abnormal users starts to rise, and it is almost 0 in the daytime.

Through the time-series distribution of the four typical traffic indicators of abnormal user number of flows, flow bytes, the amount of users, and online duration, it can be observed that there are infected people with abnormal users even during the day, but the numbers have always remained at a low level during the day. After 22:00, these numbers have increased significantly. Between 2:00-5:00, although the number of abnormal users is not the peak in these 5 days. However, the traffic accessing abnormal network domain names within this period is the highest value. The traffic in the evening hours is significantly more than in the daytime hours. This means that the high frequency of abnormal flow behavior tends to occur more at nighttime than at other times. Since people follow obvious patterns in their daily behaviors [27] (for example: wake up at 7 am, go to work before 9 am, eat lunch at 12 o'clock, go home after 5 pm and go to bed at 11 pm, etc.). Studies have shown that network traffic characteristics (user-generated network traffic data) are closely related to users' daily habits and follow daily laws. The traffic characteristics found in this paper are significantly different from the overall network traffic characteristics. Because the abnormal network traffic is relatively small compared to the overall network, the abnormal network traffic is hidden in the overall network traffic and is difficult to detect. Therefore, according to the statistical analysis of the time series of abnormal network traffic in this paper, it is recommended that network managers should strengthen

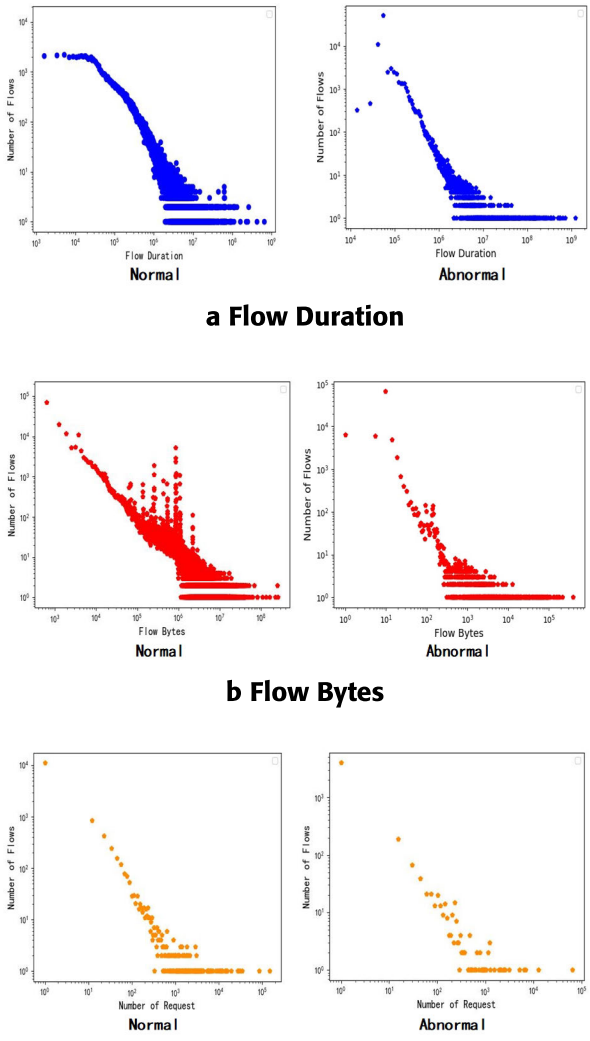


FIGURE 8. The distribution of normal and abnormal traffic within 5 days.

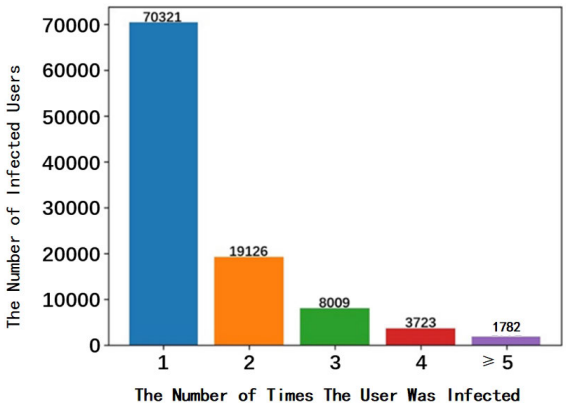


FIGURE 9. Abnormal users.

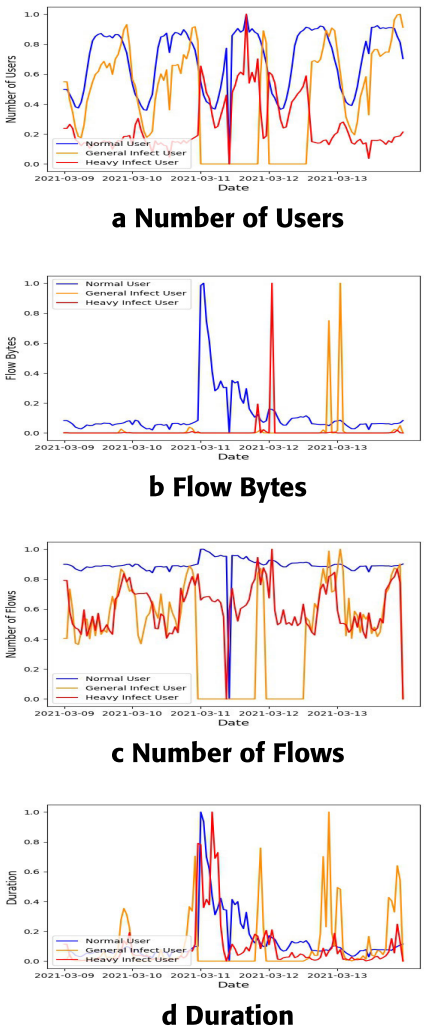


FIGURE 10. Comparison of abnormal user and normal user traffic.

the monitoring and management of abnormal traffic between 22:00 and 5:00 in the evening. In this way, users can avoid network security accidents caused by the relaxation of the monitoring level in the evening.

B. THE FREQUENCY DISTRIBUTION

To capture the statistical characteristics of the actual anomaly traffic in the mobile network, the comparative distribution of the flow duration, flow bytes and frequency of the number of requests between normal and abnormal traffic is plotted in Fig.8. It can be noticed from the results that 80 percent of the flows are less than 0.58 KB, 80 percent of the flow duration is less than 5.463 milliseconds, and 80 percent of the requests are less than 19 times. Compared with the stream bit of normal users (80% of the streams are less than 3 KB), the stream bytes of abnormal users are much smaller than that of normal users. This means that most abnormal users generate less data traffic every day, while a small number of abnormal users generate a lot of data traffic in the network. This is more helpful to distinguish abnormal and normal users

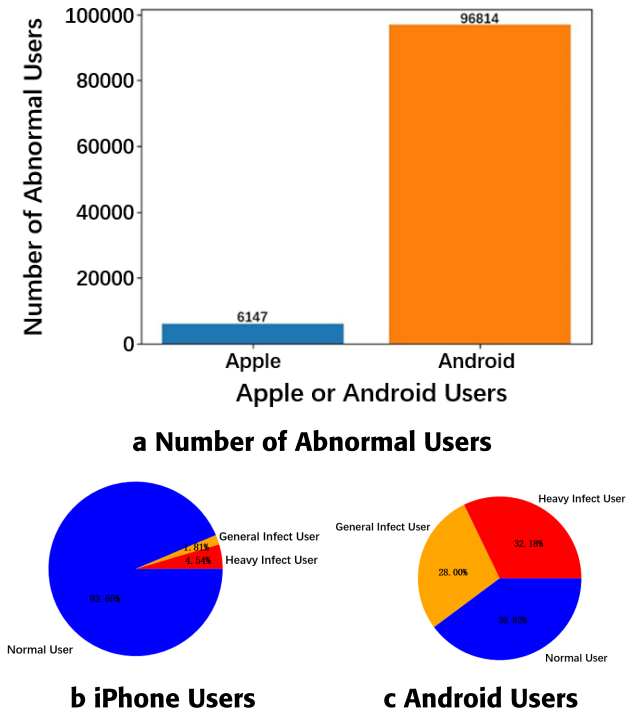


FIGURE 11. iPhone or android users.

by focusing on the size of data traffic consumed, to better protect the network environment.

In order to understand the different patterns of users in our dataset, we analyze and compare the flow metrics of heavy and general infect users, which help us understand the difference between the above two groups of users and the factors that heavy users emerge. As shown in Fig.9, we define general infect users as the mobile users producing anomaly traffic one time in 5 days, and others are heavy users.

It can be seen from Fig.10 of the normalized four indicators that the largest difference between the traffic-infected user and the normal user in the generated flow bytes and the number of generated flows occurred between 0:00 to 5:00 in the morning. General infected users also peaked in the evening and early morning. Heavy infected users peaked in the early morning of the 12th. In addition, the online duration of general infected users continued from the evening to the early morning of the next day, and for heavily infected users, the peak fluctuation of online duration was similar with the peak fluctuations of general infected users. All of them had a high peak in the early hours of the morning. This means that the longer users stay online at midnight, the more likely they are to be infected.

In other words, we can quickly detect whether it is infected by observing his/her data traffic usage behavior at midnight.

C. EQUIPMENT USER ANALYSIS

At present, mobile terminal devices mainly include two categories: iPhone and Android. To discover the susceptibility of different devices, we extracted different device types of devices of infected users and analyzed the abnormal traffic characteristics of different devices.

Firstly, as can be seen from Fig.11a, we found 5.97% of the abnormal users are using iPhone, while most of the rest are Android devices. Secondly, the number of all users using iPhone and Android devices is 96814 and 160869 respectively. Only 1.81% of iPhone users were generally infected. The heavily infected iPhone users accounted for 4.54%. It can be seen that fewer iPhone users are infected in Fig.11b. However, Android users are different from iPhone users. The general infect Android users accounted for 28.00%, and the heavy infect Android users accounted for 32.18% in Fig.11c. IPhone users with exceptions accounted for 6.32% and Android users with exceptions accounted for 60.18%. This means that Android users are more susceptible to infection. Therefore, we should further strengthen the monitoring and anti-infection technology for Android users. So as to prevent more users from being infected.

VI. CONCLUSION

In this work, we propose a Streaming Network Malicious Application Detection Framework (SNMDF). The framework is based on Apache Spark Streaming and Kafka and is capable of processing and streaming analytics work on mobile big data. It also supports traffic customization under different systems, automatic models and black and white list updates, automatic cluster scheduling of jobs, and allocating computing resources to running jobs according to their importance. The storage and computing power of the SNMDF framework can be easily increased or decreased by adding or removing machines or computing resources in the cluster. In addition, the redundancy and security of applications and data can be improved by centralized horizontal scaling.

SNMDF is applied to a 5-day mobile Internet traffic dataset in a province in China to conduct a detailed statistical analysis of malicious messages. We found that the number of anomalous users increased significantly from midnight to the early hours of the next morning and that Internet usage was significantly higher at night than during the day. The traffic to the abnormal network domain name between 22:00-5:00 in the evening is the maximum value. Internet traffic is significantly higher at night than during the day. This means that the high frequency of abnormal infection behavior tends to occur at night and not at other times. Since abnormal network traffic is relatively small relative to the overall network, abnormal network traffic is hidden in the overall network traffic and is difficult to detect. According to the statistical analysis of the time series of abnormal network traffic in this paper, we recommend that network administrators strengthen the monitoring and management of abnormal traffic between 22:00-5:00 in the evening. In this way, users can further avoid network security accidents caused by relaxing the monitoring level at night. Thereby, the security and reliability of the user's online environment are improved.

REFERENCES

[1] K. A. P. da Costa, J. P. Papa, C. O. Lisboa, R. Munoz, and V. H. C. de Albuquerque, "Internet of Things: A survey on machine learning-based intrusion detection approaches," *Comput. Netw.*, vol. 151, pp. 147–157, Mar. 2019.

[2] N. Moustafa, B. Turnbull, and K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4815–4830, Jun. 2019.

[3] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in cloud," *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 42–57, Jan. 2013.

[4] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *J. Comput. Sci.*, vol. 25, pp. 152–160, Mar. 2018.

[5] C. Fang, J. Liu, and Z. Lei, "Parallelized user clicks recognition from massive HTTP data based on dependency graph model," *China Commun.*, vol. 11, no. 12, pp. 13–25, Dec. 2014.

[6] J. Yang, Y. Qiao, X. Zhang, H. He, F. Liu, and G. Cheng, "Characterizing user behavior in mobile Internet," *IEEE Trans. Emerg. Topics Comput.*, vol. 3, no. 1, pp. 95–106, Mar. 2015.

[7] Y. Jin, N. Duffield, A. Gerber, P. Haffner, W.-L. Hsu, G. Jacobson, S. Sen, S. Venkataraman, and Z.-L. Zhang, "Characterizing data usage patterns in a large cellular network," in *Proc. ACM SIGCOMM Workshop Cellular Netw., Oper., Challenges, Future Design*, Aug. 2012, pp. 7–12.

[8] C.-C. Chao, C.-H. Lee, H.-Y. Wei, C.-Y. Wang, and W.-T. Chen, "Distributed dynamic-TDD resource allocation in femtocell networks using evolutionary game," in *Proc. IEEE 26th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Aug. 2015, pp. 1157–1162.

[9] M. Chen, Y. Hao, K. Lin, Z. Yuan, and L. Hu, "Label-less learning for traffic control in an edge network," *IEEE Netw.*, vol. 32, no. 6, pp. 8–14, Nov. 2018.

[10] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for wireless resource management," in *Proc. IEEE 18th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2017, pp. 1–6.

[11] J. Liu, F. Liu, and N. Ansari, "Monitoring and analyzing big traffic data of a large-scale cellular network with Hadoop," *IEEE Netw.*, vol. 28, no. 4, pp. 32–39, Jul. 2014.

[12] S. Alzahrani and L. Hong, "Detection of distributed denial of service (DDoS) attacks using artificial intelligence on cloud," in *Proc. IEEE World Congr. Services (SERVICES)*, Jul. 2018, pp. 35–36.

[13] W. Chen, F. Lyu, F. Wu, P. Yang, G. Xue, and M. Li, "Sequential message characterization for early classification of encrypted Internet traffic," *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3746–3760, Apr. 2021.

[14] M. Dusi, M. Crotti, F. Gringoli, and L. Salgarelli, "Tunnel hunter: Detecting application-layer tunnels with statistical fingerprinting," *Comput. Netw.*, vol. 53, no. 1, pp. 81–97, Jan. 2009.

[15] A. Singh and B. B. Gupta, "Distributed denial-of-service (DDoS) attacks and defense mechanisms in various web-enabled computing platforms: Issues, challenges, and future research directions," *Int. J. Semantic Web Inf. Syst.*, vol. 18, no. 1, pp. 1–43, Apr. 2022.

[16] D. Oh, D. Kim, and W. Ro, "A malicious pattern detection engine for embedded security systems in the Internet of Things," *Sensors*, vol. 14, no. 12, pp. 24188–24211, Dec. 2014.

[17] H. Ye, G. Ye Li, and B.-H. Fred Juang, "Bilinear convolutional auto-encoder based pilot-free end-to-end communication systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

[18] M. E. Aminanto, R. Choi, H. C. Tanuwidjaja, P. D. Yoo, and K. Kim, "Deep abstraction and weighted feature selection for Wi-Fi impersonation detection," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 3, pp. 621–636, Mar. 2018.

[19] A. Azmoodeh, A. Dehghantanha, and K. R. Choo, "Robust malware detection for Internet of (Battlefield) things devices using deep eigenspace learning," *IEEE Trans. Sustain. Comput.*, vol. 4, no. 1, pp. 88–95, Jan. 2019.

[20] N. Goldenberg and A. Wool, "Accurate modeling of modbus/TCP for intrusion detection in SCADA systems," *Int. J. Crit. Infrastruct. Protection*, vol. 6, no. 2, pp. 63–75, Jun. 2013.

[21] I. Cvitic, D. Perakovic, B. B. Gupta, and K. R. Choo, "Boosting-based DDoS detection in Internet of Things systems," *IEEE Internet Things J.*, vol. 9, no. 3, pp. 2109–2123, Feb. 2022.

[22] V. Le Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczynski, and W. Joosen, "TRANCO: A research-oriented top sites ranking hardened against manipulation," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2019, pp. 1–16.

[23] H. Peng, D. Liang, and C. Choi, "Evaluating parallel logistic regression models," in *Proc. IEEE Int. Conf. Big Data*, Oct. 2013, pp. 119–126.

[24] R. Sharma and N. Sharma, "Attacks on resource-constrained IoT devices and security solutions," *Int. J. Softw. Sci. Comput. Intell.*, vol. 14, pp. 1–21, Jan. 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:252796630

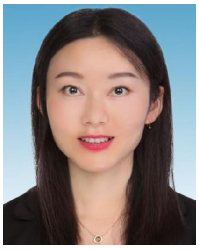
[25] T. Lin and C. Jiang, "Load forecasting of power SCADA based on spark MLlib," in *Proc. Int. Conf. Modeling, Simulation Optim. Technol. Appl. (MSOTA)*, 2016, pp. 480–484.

[26] J. Li, L. Zhang, Q. Hong, Y. Yu, and L. Zhai, "Space spider: A hyper large scientific infrastructure based on digital twin for the space internet," in *Proc. 1st Workshop Digit. Twin Edge AI Ind. (IoT)*, Oct. 2022, pp. 31–36.

[27] R. K. Polaganga and Q. Liang, "Self-similarity and modeling of LTE/LTE–A data traffic," *Measurement*, vol. 75, pp. 218–229, Nov. 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263224115003991

SHENGLONG LIU was born in 1988. He received the master's degree from Beihang University, in 2016. His research interest includes network and data security.

YUXIAO XIA was born in 1990. She received the master's degree from the University of Chinese Academy of Sciences, in 2015. Her research interest includes technology of computer application.



DI WANG was born in 1987. She received the master's degree from Xinjiang University, in 2020. Her research interests include electric power networks and data security.