Hindawi
Scientific Programming
Volume 2020, Article ID 8872482, 9 pages
https://doi.org/10.1155/2020/8872482

Research Article

The Abnormal Detection for Network Traffic of Power IoT Based on Device Portrait

Jiaxuan Fei,[1,2] **Qigui Yao**,[1,2] **Mingliang Chen**,[3] **Xiangqun Wang**,[1,2] **and Jie Fan**[1,2]

[1]*Global Energy Interconnection Research Institute Co., Ltd., Nanjing, China*
[2]*State Grid Key Laboratory of Information & Network Security, Nanjing, China*
[3]*State Grid Jiangxi Electric Power Co., Ltd., Ganzhou, JiangXi, China*

Correspondence should be addressed to Jiaxuan Fei; 444965979@qq.com

Received 2 September 2020; Revised 8 October 2020; Accepted 10 November 2020; Published 24 November 2020

Academic Editor: Ting Yang

The construction of power Internet of things is an important development direction for power grid enterprises. Although power Internet of things is a kind of network, it is denser than the ordinary Internet of things points and more complex equipment types, so it has higher requirements for network security protection. At the same time, due to the special information perception and transmission mode in the Internet of things, the information transmitted in the network is easy to be stolen and resold, and traditional security measures can no longer meet the security protection requirements of the new Internet of things devices. To solve the privacy leakage and security attack caused by the illegal intrusion in the network, this paper proposes to construct a device portrait for terminal devices in the power Internet of things and detect abnormal traffic in the network based on device portrait. By collecting traffic data in the network environment, various network traffic characteristics are extracted, and abnormal traffic is analyzed and identified by the machine learning algorithm. By collecting the traffic data in the network environment, the features are extracted from the physical layer, network layer, and application layer of the message, and the device portrait is generated by a machine learning algorithm. According to the established attack mode, the corresponding traffic characteristics are analyzed, and the detection of abnormal traffic is achieved by comparing the attack traffic characteristics with the device portrait. The experimental results show that the accuracy of this method is more than 90%.

1. Introduction

Power IoT (Internet of things) is to apply the Internet of things technology in the smart grid business, in power generation, transmission, substation, power distribution, and utilization, and so, on each link, the comprehensive deployment has edge information awareness, calculation ability, and management ability to execute the terminal device, effectively integrate power system infrastructure and communications infrastructure resources, and promote the operation of the enterprise operating the whole process of the whole scene perception, information fusion, and intelligent decision support, to raise the efficiency of utilization of electric power system existing infrastructure for the grid all the chain management to provide important technical support [1, 2]. With the wide application of Internet of things technology in the SGC (State Grid Co., Ltd.) to adapt to the traditional industry and the trend of the Internet to accelerate convergence, the content associated terminal will increase by geometric series, through the sensor technology, communication technology, and computer technology to terminal access networks; it puts forward higher requirements on network security protection. At the same time, due to the particularity of information perception and transmission mode in the Internet of things, the transmission information of the Internet of things is easy to be stolen and replay. Traditional security measures can no longer meet the security protection requirements of the new Internet of things devices. Therefore, how to realize the security of the ubiquitous power Internet of things and build a full-scene security protection system that adapts to the ubiquitous power Internet of things has become an urgent problem to be solved by The State Grid Company.

The power Internet of things architecture consists of four logical levels from bottom to top: sensing, transport, platform, and application [3]. The sensor layer collects raw data from smart meters, sensors, handheld terminals, cameras, PCS, and other smart devices, which are the source of all data of power grid companies. The network layer transmits the data collected by the perception layer to the platform layer securely and reliably through the network communication technologies such as power communication network and wireless private network. The platform layer stores, clusters, and analyzes data to provide data support for the application layer. The application layer provides data to users for service [4–7]. In the perception layer, the power system's devices owned limited computing capacity and less storage. So, the traditional authentication method is not suitable, which will bring serious security problems. Communication methods and network protocols are complex, which makes network security protection more difficult. In the application layer, the interface of security responsibility is not clear, and there is a management gap. Power Internet of things lacks complete network security protection standards for power utilities. However, with the development of new services and the change of security situation, the extensive access of a large number of terminal equipment and users in the ubiquitous power Internet of things environment increases the network exposure surface, which brings severe challenges to the protection system characterized by boundary isolation. In addition, the ubiquitous power Internet of things gives birth to a large number of new business models, business interaction is more complex, and more flexible and accurate security policies and protection measures are urgently required.

Therefore, this paper takes terminal devices in the power Internet of things as the object, extracts features based on the traffic generated by their information exchange and constructs device portraits. Device portrait is the tagging of information, which presents the overall state of the device by describing the characteristics of a series of individuals. Through the device portrait, we can clearly and intuitively see the various feature dimensions of all the devices. The network anomaly detection technology can identify the attack behavior of illegal devices according to the device portrait and inform the system to intercept and deal with it in time. The information interaction between networks is carried by network traffic, and the behavioral characteristics of network attacks will naturally be reflected in the network traffic generated. Therefore, device portrait constructed according to device traffic is an efficient and real-time anomaly detection technology.

The simulation results show that the average accuracy of abnormal behavior detection can reach more than 90%. The rest of the article is structured as follows. We begin with an overview of the work related to flow anomaly detection in Section 2. In Section 3, we will introduce the main work of the scheme. In addition, in section 4, we introduce the simulation results and finally summarize them in Section 5.

2. Related Work

The premise of network anomaly detection technology is to understand the abnormal behavior of the network attack. Abnormal activities can be divided into three types: point exception, context exception, and collective exception. According to the attacker's objectives and activities, the attack is classified as the following four types: DoS (denial of service) attack, probing attacks, the user to the root (U2R), and remote to the user (R2U). In network anomaly detection technology, the representation of abnormal behavior is usually divided into two kinds: fraction and binary tag. According to [8], the current network anomaly detection technologies can be divided into four categories: based on classification, typical classification technologies include support vector machine, Bayesian network, and neural network. Based on statistical theory, chi-square test statistics are taken as the standard. Typical statistical theory-based methods include hybrid model, signal processing technology, and principal component analysis.

Since Denning first put forward the network intrusion detection model in 1980s, scholars at different times have put forward many network abnormal behavior detection methods with their own advantages by using the latest technology. At present, abnormal behavior detection methods mainly include four kinds: the most basic method based on statistical analysis, the method based on feature rules based on comparison and matching of data features and feature base, the method based on data mining with automatic analysis capability, and the most common method based on machine learning. Because machine learning algorithms can detect unknown patterns effectively, network anomaly detection based on machine learning has become the focus of research in recent years, for example, network abnormal behavior detection based on clustering, naive Bayes, or decision tree.

There are many algorithms in machine learning, and they have their own advantages and disadvantages. Therefore, scholars further study and innovate on the basis of previous studies, fuse multiple algorithms or improve them to design a new model, effectively make use of their respective advantages and avoid disadvantages to improve the effect of network anomaly detection. B. Senthilnayaki et al. [9] combined a genetic algorithm and support vector machine algorithm to propose a network anomaly analysis method, and the results showed that the detection accuracy of partial features extracted with genetic algorithm was higher than that of the support vector machine model trained with all features. Chang et al. [10] proposed a network anomaly detection method based on RF and SVM to address the low detection rate of network anomaly detection and found that the combination of feature extraction and machine learning could improve the detection rate. Gao et al. [11] proposed an adaptive integration model. By adjusting the proportion of training data, setting up multiple decision trees, and constructing MultiTree algorithm, the comparison test proved that the detection accuracy was improved, and it was found that the quality of data features was an important factor determining the detection effect.

Naseer et al. [12] studied the applicability of deep learning in network anomaly detection and implemented anomaly detection models based on different depth neural network structures, including convolutional neural network, autoencoder, and regression neural network. Tavoli [13] proposed a new intrusion detection method based on MLP neural network in view of the shortcoming that traditional anomaly detection methods cannot detect unknown anomalies in the network with high speed and complexity. Experimental results show that this method is superior to other methods in reducing false positives. Yong [14] proposed an intrusion detection algorithm based on the convolutional neural network. This network model has higher accuracy and detection rate than classical BP neural network, SVM algorithm, and deep learning algorithm DBN, which improves the classification accuracy of intrusion detection recognition. Zhang et al. [15] proposed an intrusion detection method based on deep learning, which uses a deep automatic encoder to compress unimportant features, extract key features, and build a model, and it uses the NSL-KDD data set to conduct tests to quickly and accurately identify attacks.

3. The Design of Scheme Architecture

The overall framework of this paper is mainly divided into two parts, as shown in Figure 1: the construction of interrupt device portrait and the detection of abnormal network access behavior of devices under specific attack scenarios.

Taking all the traffic generated by the terminal equipment as the object of study and comprehensively considering the physical layer, network traffic, and protocol behavior characteristics of the terminal equipment, the portrait of the terminal equipment was established. Based on the established device portrait and combined with the specific attack scenario, analyze and detect whether the network access behavior of the terminal is abnormal. The specific implementation route is shown in Figure 1.

3.1. Device Portrait. Based on the device data, utilizing tagging and taking the device as a unit, the data model is established to analyze the device data and extract the labels of each dimension of the device. The label set of the device is the device portrait constructed. Equipment data interpretation can be from two aspects: from the perspective of physical view, different devices have different electronic components, in which the emission of the electromagnetic wave from different devices also are different. These characteristics include carrier frequency offset of the baseband's steady-state responses, synchronization signal correlation value, baseband I/O of offset, signal demodulation signal amplitude and phase error, etc. The value of these characteristics is unique to the different devices to be treated as the fingerprints of the devices. On the other hand, it starts from the network layer, which is mainly reflected in the network traffic generated by devices during the running time. Due to different functions and applications, the performance on the TCP/IP protocol stack is also different. According to these two types of characteristics, useful data can be extracted from them to form a tag set using tagging. Then, based on the machine learning algorithm, exclusive portraits can be constructed for different devices to identify legitimate and illegal users.

Portrait refers to the digitization and labeling of information. A series of features that can represent the device are logically combined in a certain way to form a proprietary portrait of the device. The core work of the portrait is to find as many features that can represent the device as possible and as much as possible, to fuse multidimensional features, and to generate a device portrait by establishing a mathematical model to analyze device data.

3.2. System Model. Device portrait is a way to present the multidimensional description of device features. In this paper, the portrait of the terminal device contains two core contents, such as basic attributes and access behavior attributes, as shown in Figure 2. Basic properties include the terminal's IP, MAC address, and machine name. Access behavior attribute is composed of physical characteristics of a terminal device, network traffic characteristics, and protocol behavior characteristics.

Basic properties can be obtained from network traffic packets. The physical layer characteristics of access behavior attributes include transient signal fingerprint characteristic vector of individual characteristics of communication devices.

The characteristics of the physical layer include the constellation locus of BPSK, QPSK, OQPSK, and MSK modulation signals in the receiving end baseband and the time domain and frequency domain characteristics of the time domain waveform. The constellation trajectory chart is a means to measure the emitted signal itself and its change law. The nonlinear response of the transmitter amplifier, the response of the filter, and other linear and nonlinear interference factors will be reflected in the changing trajectory of the constellation trajectory map. A constellation chart provides a more comprehensive measure of the characteristics of the received signal. After receiving the oversampled baseband signal, the receiver can preprocess the signal simply. Preprocessing is mainly to normalize the energy of the signal. After preprocessing the received signal, the signal is sent to the I/Q two-channel delay device. The delayer can choose the same and different delays for both I/Q channels. The choice of I/Q two-channel signal delay is mainly determined by the judgment of signal modulation. After that, the system performs differential processing on the signal, and the stable and clear constellation trajectory diagram can be drawn on the complex plane.

Network traffic features include target IP address distribution (number), magnitude distribution (mean and variance) of upstream and downstream traffic, duration distribution (mean and variance) of upstream and downstream traffic, and network flow order. The distribution of the target IP address, the size distribution of upstream and downstream traffic, and the duration distribution of upstream and downstream traffic can be extracted from the

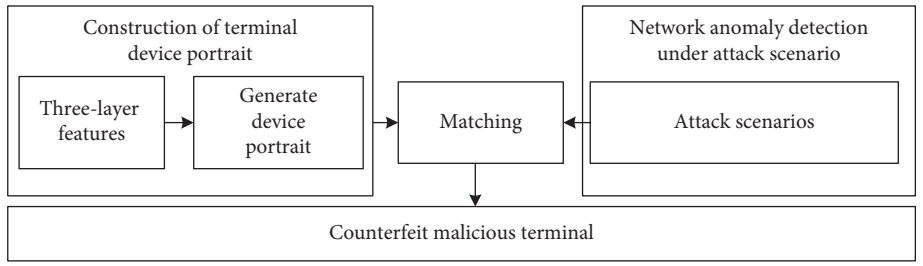

Figure 1: Implementation roadmap of abnormal network access behavior detection based on device portrait.

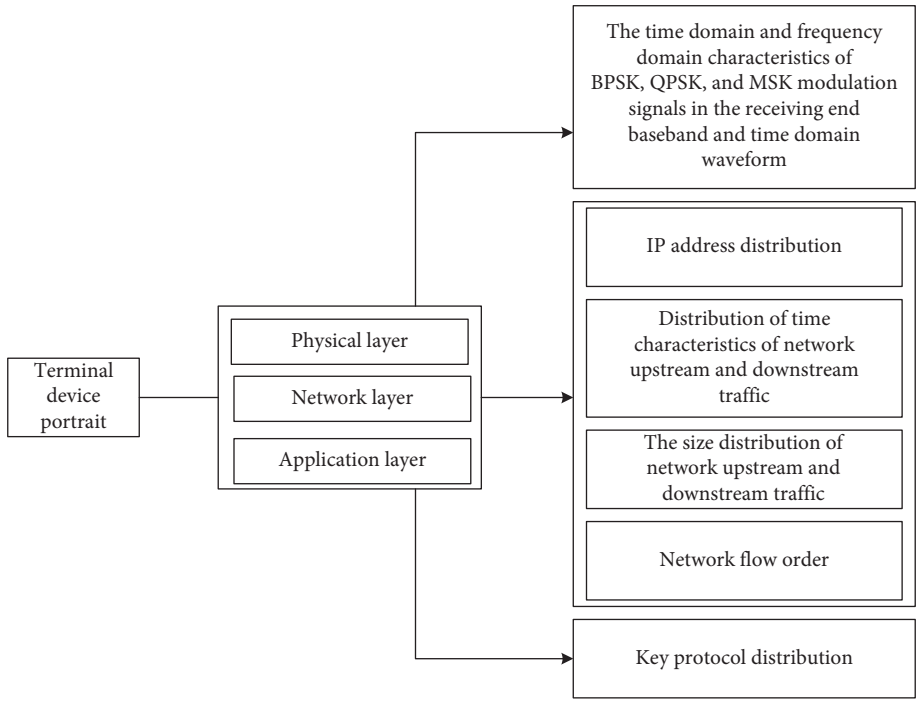

Figure 2: Portrait content of terminal equipment.

network flow order and calculated statistically. Statistics of such characteristics is convenient for rapid detection. There are also Timestamp field, MSS, WScale, SYN, FIN, ACK, PSH, URG, and RST in the TCP header. Version number, IHL, TTL, DF, TOS, Protocol field value, source port number (Sport), destination port number (Dport), and so forth are shown in Tables 1 and 2.

The characteristics of the application layer include protocol key fields, writing all the protocol keywords that appear into the document, and reflecting the protocol characteristics in the form of files, without specific analysis of the business type. The characteristics of the service layer can be determined according to the parallel parsing results of the protocol message.

3.3. Abnormal Network Access Behavior Detection of Devices under Specific Attack Scenarios. Time range T can be set for the construction of the device portrait, and the eigenvalue in time T is the current portrait of the device. Based on device portrait, abnormal network access behavior of terminal devices of power Internet of things can be further detected.

Table 1: TCP and IP header field value.

TCP	Timestamp, MSS, WScale, SYN, FIN, ACK, PSH, URG, RST
IP	Version, TOS, IHL, TTL, DF, Protocol, Dport, Sport

Table 2: Application layer protocol.

TCP-based	http, https, smtp, ssh, ftp, lpd rtsp, telnet, raw
UDP-based	snmp, onvif, dns, ntp, mdns, ssdp, icmp, igmpv3, nfs, dhcp, tftp, pop

Specifically, this project intends to analyze the portrait of terminal devices under specific attack scenarios and determine abnormal network access behaviors, to realize the fast and accurate detection of forgery and malicious terminals. The specific process is as follows:

Step 1. Establish attack modes. Referring to the network attack chain model, the process of the attack is summarized and the attack mode is established. Because the network attack chain model is mainly aimed at Internet attacks and the whole service is ubiquitous in the power Internet of things, there are special attacks, such as fake terminals and then access to background applications. Therefore, this project considers specific attack scenarios and plans to establish a common attack mode for the power Internet of things. Specifically, in addition to the normal stages of investigation and weaponization included in the network attack chain model, special stages such as terminal forgery and abnormal execution of normal instructions will be added, to model such special attacks as using forgery terminals to access specific services and then destroying business systems through normal instructions.

There are various types of network attacks, and there are corresponding attacks for each level. It is not practical to use a network anomaly detection model for all attacks. TCP-IP architecture is the infrastructure of today's Internet; many network attacks are aimed at TCP, IP layer, affecting the normal process of the target host. In this paper, SYN denial of service attack, TCP port scanning, and IP attack are selected to verify the detection performance of the system, among them the SYN denial of service attack. In the three-time handshake process based on TCP connection, the attacker sends SYN request message to the target, and the target host will assign resources after receiving the message, send the response message, and wait for the attacker to reply. The attacker does not respond to the target, making the target in a waiting state. Through a large number of SYN request packets, the resource of the target host can be exhausted. TCP port scanning: when the corresponding port of the target host receives the TCP connection request, if the port is open, the TCP ACK message is sent back to establish the connection, and if the port is not open, the TCP RST message is sent to inform the sender that the port is not open. By sending requests to all ports of the target in turn, detecting the received response can tell which ports of the target are open for the next attack. The IP address of the target host is "192.168.2.102". The target port is traversed and selected in (1,65535) to parse the received message. If the TCP layer flags = 18, the port is open; otherwise, it is closed. Sharding IP packet attack: when an IP packet is too large, it will be shared by the IP layer. The flag MF of the sharded packet tells the receiver that the packet has been sharded. The target host will receive these packets into the cache waiting for the arrival of subsequent packets and merge them. Sending a large number of sharding messages to the target artificially will deplete the cache resources of the target host. The IP address of the target host is "192.168.2.102", the target port is 80, the source address is any available IP in the network segment, and the source port is any port in the interval. Each IP address will be traversed to send five different ids, and each ID will send five IP messages with different slice offsets.

Step 2. Analyze the traffic characteristics of the established attack mode. For each attack step in the established attack mode, the corresponding traffic characteristics are analyzed. The flow characteristics analyzed include physical layer characteristics, such as time domain and frequency characteristics of the communication channel. Network traffic characteristics and network flow order of attack traffic are constructed. Protocol layer characteristics build the attack traffic behavior model diagram.

Step 3. Compare and distinguish attack traffic characteristics and device portraits based on the cluster analysis algorithm and the nearest neighbor set selection. If similar traffic characteristics are found, the terminal is judged to be a counterfeit or malicious terminal.

Clustering technology is a widely applied unsupervised machine learning algorithm, which combines the k-means algorithm and improved collaborative filtering algorithm to realize network anomaly detection technology based on device image. In the process of dividing a set of objects into different classes, the same class of data should have similar characteristics, and the data characteristics in different classes should be highly different. Data sets containing normal and abnormal data form clusters of various sizes, and the smaller and sparse ones are generally considered as abnormal. After the completion of clustering, the similarity between the observed data and each cluster center is calculated, and the cluster with a high similarity is selected as the nearest neighbor set. If the similarity between the abnormal cluster and the normal cluster is higher than that between the abnormal cluster and the normal cluster, or the similarity between the normal cluster and the abnormal cluster is less than the established threshold, it is considered as an anomaly. The detailed process of clustering is as follows:

(1) Randomly select K objects from n objects as the initial clustering center

(2) Calculate the difference between all objects and the central object according to the mean value of each cluster object, and divide the elements into the cluster with the lowest difference

(3) The mean value of each cluster with changes was calculated again

(4) Repeat steps b and c until no change occurs in each cluster

(5) Output results

In the process of selecting the nearest neighbor, the similarity is calculated through the weighted idea of various similarity balance factors [12], and the selected similarity measurement factors mainly include feature matrix

similarity and feature similarity of device portrait; then, the similarity calculation formula in this section is

$$\sin(u, v) = \alpha \sin_r(u, v) + \beta \sin_e(u, v), \quad (1)$$

where $\sin_r(u, v)$ is used to describe the similarity of different features between the device u to be observed and the existing device; $\sin_e(u, v)$ is used to describe the similarity of the image of the device to be observed and the existing device; and α, β in turn are used to describe the corresponding weights. After the similarity is obtained, the results with the largest similarity are taken as the nearest neighbor. The following is the calculation of equipment feature similarity. Assuming there are m features, the m features of the observation equipment u and the existing equipment v are used and described by Q_u and Q_m in turn. The characteristic similarity of equipment v and u is the similarity between Q_u and Q_m. The calculation formula is as follows:

$$\sin_r(u, v) = \frac{|Q_u \cap Q_m|}{|Q_u \bigcup Q_m|}, \quad (2)$$

where $Q_u \cap Q_m$ is used to describe the characteristic quantities of the same value shared by devices u and v . $Q_u \bigcup Q_m$ is used to describe the number of uncharacteristic rights in common. The device portrait similarity is described by the Pearson correlation coefficient. The Pearson correlation coefficient is valued in the range of [−1, 1]. The higher the absolute value of the Pearson correlation coefficient, the stronger the correlation. When the Pearson correlation coefficient is 1, the similarity is the highest and positively correlated. When the Pearson coefficient is −1, the correlation is considered to be completely negative. When the Pearson correlation coefficient is 0, no relationship is considered. The similarity between u and v is measured by Pearson's correlation coefficient as follows:

$$\sin_e(u, v) = \frac{\sum (r_u - \overline{r_u})(r_v - \overline{r_v})}{\sqrt{(r_u - \overline{r_u})^2} \sqrt{(r_v - \overline{r_v})^2}}, \quad (3)$$

where r_u is used to describe the characteristic value of equipment u and $\overline{r_u}$ is the average value of the characteristics of multiple traffic flows of the equipment. r_v is the existing cluster, namely, the single eigenvalue of the device portrait of a certain device, and $\overline{r_v}$ is the average characteristic value in the device portrait.

According to the characteristics of the collaborative filtering algorithm, the device portrait similarity is optimized to prevent the nearest neighbor from being found in the whole object space. Since $r_u - \overline{r_i}$ and $r_v - \overline{r_j}$ are not defined as nonnegative, there are

$$(r_u - \overline{r_u})(r_v - \overline{r_v}) \le 4[\max(\overline{r_u}, R - \overline{r_u}, \overline{r_v}, R - \overline{r_v})], \quad (4)$$

where R is used to describe the value of the most similar feature.

In particular, specific attack stages can be detected according to different matching results. If the physical layer fingerprint features are abnormal and the network and business contents are normal, it indicates that the attacker may change the core equipment into his equipment to prepare for the subsequent attack. If the physical layer fingerprint characteristics are normal, but the network traffic characteristics are abnormal, then the business content is normal, indicating that the attacker is contacting the background server, receiving instructions or updating the attack code, and so forth. To prepare for the attack. If the physical layer and network traffic characteristics are normal and the business layer characteristics are abnormal, it indicates that the attacker is carrying out a business attack to complete the attack preparation.

4. Experimental Analysis

4.1. Simulation Experiment Environment. The experiment was conducted on the Pycharm open-source platform, which can build various machine learning algorithms, including the application of the clustering algorithm and the collaborative filtering algorithm. Build a recommendation engine with tools provided by the platform.

In this paper, the bypass monitoring traffic method is adopted to implement network data traffic through bypass monitoring which includes bypass detection, data acquisition, network analysis, and information extraction. Specific scheme of data acquisition based on bypass detection: set up bypass monitoring on the data switch, regularly collect data traffic packets of devices accessing network services and internal resources, and extract network protocol and business information related to terminal devices. Its architecture block diagram is shown in Figure 3.

By mirroring on the network switch port, bypass monitoring is set to filter the detected network packets according to the corresponding MAC address, to obtain the packets related to the device terminal. In data communication, the data packets generated by each network activity of the terminal equipment can be regarded as the process of data interaction between two nodes. This passive device traffic acquisition method is less aggressive and invasive, has no restrictions on the type of device, and is easy to operate in practice, which is also the reason why passive device traffic acquisition is selected in this paper.

There is another problem that needs to be considered. The terminal equipment will show different network behaviors with the change of the execution task, so the network traffic generated by the equipment is of great uncertainty, which will affect the accuracy of our portrait generation. Faced with this problem, we divide the entire running process of the terminal device into two parts: the startup phase and the service phase. The process from power-on to complete the hardware and software configuration in the startup stage: the service stage is the stage in which the device performs functional tasks after completing various configurations. In equipment beginning to perform a task, its network behavior is highly susceptible to the influence of network environment, such as the network administrator to configure the network environment changes and the communication terminal entity flow changes, and this kind of change in a certain period of time is needed to show some

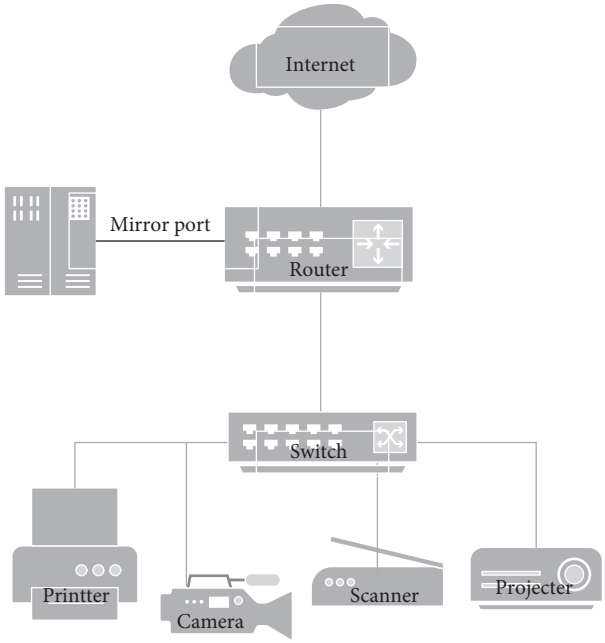

Figure 3: Data acquisition architecture diagram.

kind of rule and has the characteristics of uniqueness, but this time cycle length is not fixed. For example, the periodic behavior model of access point devices can be analyzed by analyzing individual data packets, which is not feasible in practical operation. We can use the characteristics of the startup phase to solve this problem. Due to its single function and limited resources, many configurations of intelligent devices in the Internet of things will be fixed in the hardware or software settings, that is, the configuration in the startup stage. The transition from power failure to the normal state of the device has a strict loader, which provides us with a stable flow window starting time. Launch complex Linux operating system. For example, firstly, the basic input-output system (BIOS) is checked to confirm the hardware status is normal; secondly, load the disk, and then load the master boot record (MBR) of file system's; Thirdly, read the content of the code from the MBRboot, which is the bootstrap program. The bootstrap program contains the code to load the system's kernel, start the initialization process, perform the script from different levels, etc.; Finally, the whole booting procedure is completed by the bootstrap program of the operating system loads. The startup process takes a relatively fixed time, so the time window experienced in the startup stage is relatively stable. In addition, most of the installation systems of smart devices in the Internet of things are not heavyweight systems, so the startup time is about 3 minutes or less. Therefore, we choose to obtain the traffic packets during the startup period of the device to construct the device portrait.

In the attack module, this paper constructs a system consisting of two hosts, making one host the attacker and the other the target host to receive the attack. The data acquisition module collects attack data and traffic generated by normal communication, collects and saves network attack data and normal data together, and analyzes packets. Feature extraction module is responsible for feature extraction and processing, changing network data from unordered to TCP connection, and writing code to calculate the feature value of each connection from these fields. After feature extraction, feature processing should be carried out to make features meet the requirements of the machine learning algorithm. The three attacks generated in this article are all based on the Scapy library. Scapy is used to set the field value of the network protocol, which can generate the required packets and form the network attack.

4.2. Determine the Weight Value of Similarity. The weights α and β in the similarity calculation of attack flow and device portrait are in line with $\alpha + \beta = 1$. According to the actual data and the least square fitting method in linear programming, α and β are adjusted to obtain the optimal result. The following are three different cases of $\alpha = 0.65, \beta = 0.35$, $\alpha = 0.55, \beta = 0.45$, and $\alpha = 0.35, \beta = 0.65$ to compare the accuracy rate and recall rate. The success rate is the proportion of the total weight of the successful verification match of the attack device. The recall rate is the proportion of the number of successful matches in the candidate set to the actual number of received matches in the experiment.

It can be seen from Table 3 that, in the case of increase, the accuracy rate gradually increases and is the highest in the case of 0.65. By synthesizing all the elements, the similarity analysis in cluster analysis takes time.

4.3. Comparison of Different Algorithms. After determining the parameters of cluster analysis, the device portrait is constructed by comparing different machine learning algorithms. In the experiment, the selected machine learning algorithms include logistic regression, decision tree, and random forest. The algorithm which is most consistent with device portrait and anomaly detection is selected. The results are as in Table 4.

As can be seen from Table 4, the cluster analysis algorithm has the highest accuracy of 91.2% in TCP port scanning, while the random forest algorithm has the lowest accuracy of 88.7%. In IP sharding attack, the accuracy rate reaches 90.9% in the clustering algorithm and 80.3% in the decision tree. In the SYN denial-of-service attack, the logistic regression algorithm had 90.5% accuracy, and the second highest was cluster analysis, with a difference of only 0.2%.

According to the results in Table 4, among the above four algorithms, the device portrait constructed by the clustering analysis algorithm has a high accuracy rate for the identification of device anomalies in the network, meeting the requirements of the system.

4.4. The Influence of Features on Anomaly Detection. The device portraits constructed by different features are also different. To observe the influence of device portraits on anomaly detection, different features are deleted, respectively, to observe the difference of experimental results.

Figures 4–6, correspond to the influence of device portrait on detection results in three attack modes: TCP Port

Table 3: Weight analysis results.

Weighted value	Recall (%)	Precision (%)
$\alpha = 0.65, \beta = 0.35$	86.2	91.2
$\alpha = 0.55, \beta = 0.45$	89.3	75.9
$\alpha = 0.35, \beta = 0.65$	61.2	53.1

Table 4: Comparison of the accuracy of different attack types and different algorithms.

Algorithm	TCP port scan (%)	SYN denial (%)	IP sharding attack (%)
Cluster analysis	91.2	90.3	90.9
Logistic regression	89.5	90.5	90.5
Decision tree	89.1	88.3	80.3
Random forest	88.7	78.5	85.8

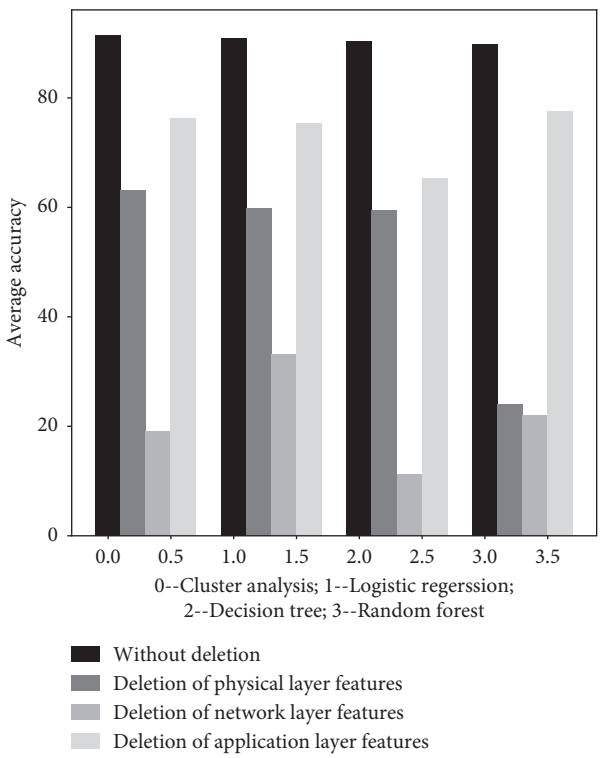

Figure 4: Influence of device portrait on port scanning.

Scan, SYN denial Attack, and IP sharding attack. The vertical axis represents accuracy, and the horizontal axis represents four algorithms from left to right: clustering, logistic regression, decision tree, and random forest. The first cylinder represents the result before deleting the feature, the second represents the accuracy of deleting the physical layer feature, the third represents only deleting the network layer feature, and the fourth represents the result of deleting the application layer feature.

As can be seen from Figures 4–6, after the deletion of network layer features, the system performance degrades significantly, followed by the physical layer and finally the application layer. It can be seen from the above that the

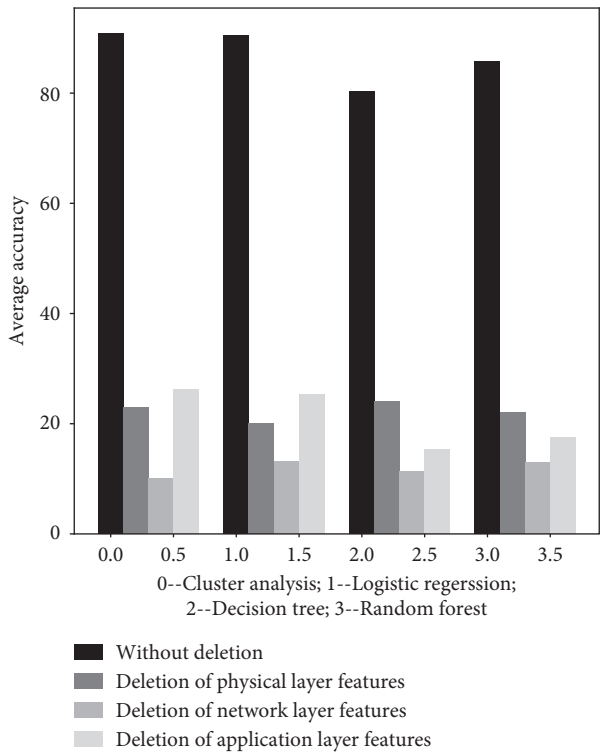

Figure 5: Influence of device portrait on SYN denial attack.

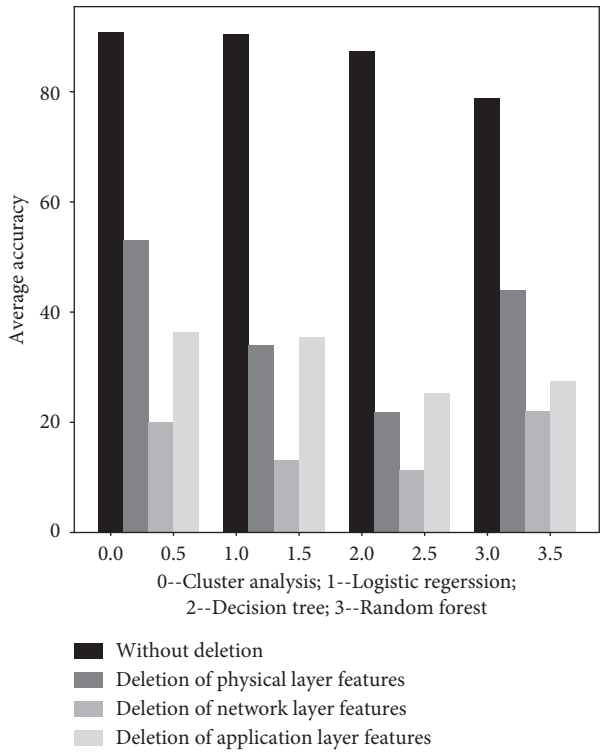

Figure 6: Influence of device portrait on IP sharding attack.

detection of abnormal network traffic by the machine learning algorithms is largely dependent on the perfection of eigenvalue selection. If some key features are removed, the

performance of the model will be significantly reduced. For the SYN denial attack, the performance of the four algorithms declines after removing several features, and the characteristics have the most obvious impact on the SYN denial attack. After the TCP port scanning attack is removed, clustering analysis and logistic regression can maintain high detection performance, while other algorithms decline. After the SYN denial of service attack was removed, the four algorithms all showed a certain degree of decline, and the random forest and cluster analysis performed slightly better. In the future upgrade, to enable the system to make more accurate judgment of network attacks, feature extraction needs to be increased and improved.

Device portrait is a description of a series of features of the device. The accuracy of device portrait to detect abnormal network traffic largely depends on the integrity of the portrait, that is, the integrity and accuracy of selected feature values. If some key features are removed, the monitoring performance of the system will be greatly reduced.

5. Conclusion

In this paper, we propose to construct device portraits for terminal devices in the power Internet of things and detect abnormal traffic in the network according to device portraits, to protect the security of the Internet of things to a certain extent. The experimental results show that the accuracy of this method is more than 90%. Due to resource constraints, we collected limited terminal equipment and traffic data and were unable to conduct large-scale testing. In the following work, we will collect more brushes and constantly improve the construction of the device portrait.

Data Availability

The experimental data source of this paper and the actual production and operation data of State Grid Corporation of China are only available on the company's internal network.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by Science and Technology Project of State Grid Corporation of China (Grant no. 5700–201958466A-0-0-00): "End-to-End Security Threat Analysis and Accurate Protection of Ubiquitous Power Internet of Things."

References

[1] C. Tian, *Application of Homomorphic Encryption in Block Chain Data Security of the Internet of Things Network Security Technology and Application*, vol. 3, pp. 34–36, 2018.

[2] J Liang, D. E. N. G. Yurong, L. Guo et al., "Research and application of remote monitoring for power transmission and transformation facilities based on satellite Internet of things," *Electric Power Construction*, vol. 34, no. 9, pp. 6–9, 2013.

[3] L. Ling, L. Shancang, and Z. Shanshan, "QoS-aware scheduling of services-oriented internet of things," *IEEE Transactions onIndustrial Informatics*, vol. 10, no. 2, pp. 1497–1505, 2014.

[4] W. Zou, J. Chen, X. Weng et al., "The security analysis and countermeasure of power Internet of things," *Electric Power Information and Communication Technology*, vol. 12, no. 8, pp. 121–125, 2014.

[5] J. Lü, W. Luan, R. Liu et al., "Architecture of distribution Internet of things based on widespread sensing & software defined technology," *Power System Technology*, vol. 42, no. 10, pp. 3108–3115, 2018.

[6] C. Wu, *Security Basis of Internet of Things*, pp. 55-56, Science Press, Beijing, China, 2013.

[7] R. Roman, J. Y. Zhou, and J. Lopez, "On the features andchallenges of security and privacy in distributed Internet ofthings," *Computer Networks*, vol. 57, no. 10, pp. 2266–2279, 2013.

[8] M. Ahmed, A. Naser Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, 2016.

[9] B. Senthilnayaki, K. Venkatalakshmi, and A. Kannan, "Intrusion detection using optimal genetic feature selection and SVM based classifier," in *Proceedings of the 3rd International Conference on Signal Processing, Communication and Networking (ICSCN)*, pp. 1–4, IEEE, Chennai, India, March 2015.

[10] Y. Chang, W. Li, and Z. Yang, "Network intrusion detection based on random forest and support vector machine,"vol. 1, pp. 635–638, in *Proceedings of the IEEE international conference on computational science and engineering (CSE) and IEEE international conference on embedded and ubiquitous computing (EUC)*, vol. 1, pp. 635–638, Institute of Electrical and Electronics Engineers, Guangzhou, China, July 2017.

[11] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, "An adaptive ensemble machine learning model for intrusion detection," *Institute of Electrical and Electronics Engineers*, vol. 7, pp. 82512–82521, 2019.

[12] S. Naseer, Y. Saleem, S. Khalid et al., "Enhanced network anomaly detection based on deep neural networks," *Institute of Electrical and Electronics Engineers*, vol. 6, pp. 48231–48246, 2018.

[13] R. Tavoli, "Providing a method to reduce the false alarm rate in network intrusion detection systems using the multilayer perceptron technique and backpropagation algorithm," in *Proceedings of the 5th Conference on Knowledge Based Engineering and Innovation (KBEI)*, IEEE, Tehran, Iran, March 2019.

[14] L. Yong and Z. Bo, *An Intrusion Detection Model Based on Multi-Scale CNN[C]//2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference*, pp. 214–218, ITNEC), Chengdu, China, 2019.

[15] C. Zhang, F. Ruan, L. Yin et al., "A deep learning approach for network intrusion detection based on NSL-KDD dataset," in *Proceedings of theIEEE 13th international conference on anti-counterfeiting, security, and identification (ASID)*, pp. 41–45, IEEE, Xiamen, China, July 2019.