A Real-time Detection Framework for Abnormal Devices in the Power Internet of Things

Weiwei Miu\textsuperscript{1}, Zhen Zhang\textsuperscript{1}, Xiaochao Wang\textsuperscript{2,3}, Jixin Hou\textsuperscript{2,3}, Yunan Sun\textsuperscript{2,3}, Qing Liu\textsuperscript{2,3} and Zeng Zeng\textsuperscript{1}

\textsuperscript{1} State Grid Jiangsu Electric Power Company Information and Communication Branch, Nanjing 210024, China
\textsuperscript{2} NARI Group Corporation (State Grid Electric Power Research Institute), Nanjing 211106, China;
\textsuperscript{3} NARI Technology Co., Ltd., Nanjing 211106, China
Email: wangxiaochao@sgepri.sgcc.com.cn

Abstract. Power Internet of Things [1] is the application of Internet of Things in smart grid. By effectively integrating power system and communication infrastructure resources, the information level of power system and the utilization rate of existing resources are improved. Thus, important technical support is provided for power grid generation, transmission, transformation, distribution and electricity consumption. The Power Internet of Things contains a large number of terminal equipment, which involves the data interaction frequency of 100 million levels per second. Therefore, how to avoid the access of invalid data is an important topic in the construction process of power Internet of Things. In response to the detection requirements of abnormal devices, this paper, based on existing Intelligent IoT System architecture, designs a real-time detection framework that integrates a two-level control mechanism, which is universal, extensible, and supports multiple protocols, using the data interaction specifications formulated by the State Grid Corporation as the basic detection basis. As a result, the framework realizes the effective interception of abnormal devices in power Internet of Things.

1. Introduction
The concept of power Internet of Things was first proposed by The State Council, and the State Grid Corporation of China has carried out a lot of technical research in this field. According to the definition given by State Grid Corporation of China, electric IoT can connect power users and their equipment, power grid enterprises and their equipment, power generation enterprises and their equipment, suppliers and their equipment, as well as people and things, generate shared data, and serve users, power grid, power generation, suppliers and government society.

In 2019, State Grid Corporation of China proposed 57 tasks in the construction plan of ubiquitous power Internet of Things, and started 27 key tasks in that year, including "construction of Intelligent IoT System [2]". The Intelligent IoT system is the source of data collection for the power Internet of Things and the basis for the realization of various applications. It mainly includes IoT management platform [3], edge IoT agent and standardized access of various types of terminals. Downward, the system addresses the needs of ubiquitous IoT and in-depth perception in fields such as power transmission and transformation, distribution network, customer side, and supply chain, and realizes unified IoT management and standardized terminal access. Going forward, the system provides open...
and standard services for enterprise middle stations, forms a cross-professional data sharing ecosystem, and gives full play to the value of data assets.

IoT management platform plays the role of data channel in the Intelligent IoT System. Equipment of various manufacturers sends data to the IOT management platform and then forwards it to the business system. In the actual application scenarios of the existing Intelligent IoT System, there are many types of devices connected to the IoT management platform, different manufacturers, and complex connection logic. In the interaction between massive devices and the IoT management platform, message format errors and abnormal sudden increase of reports are very likely to occur.

The existing verification methods for abnormal messages of power Internet of Things are simple. The current IoT management platform in Jiangsu province uses a data-flow-based blacklist method. This method puts the device ID exceeding the traffic threshold into a blacklist list, and no more data is received in the future. In view of the above blacklist mechanism, the following shortcomings are summarized:

1. Single statistical dimension, only based on data source dimension statistics, can not detect specific middleware, application and other abnormal traffic conditions;
2. It does not have the ability to detect abnormal format. When the data flow is within the threshold range, the IOT management platform cannot detect the data in the wrong format even if the device reports it for a long time;
3. The application scenario is fixed. The blacklist mechanism must be deployed along with the platform as a module in the IoT management platform, and cannot be applied to other platforms or business processes, lacking portability and universality.

2. Research Methods

This paper proposes a real-time detection framework of abnormal messages for Power Internet of Things [4], which adopts a two-level control mechanism of detection rule making and execution, which realizes the input, analysis, transformation and execution of rules by writing application programs. In response to complex message format and different communication protocol, the framework has real-time detection capabilities with custom rules, multi-protocol scalability, and flexible configuration support. Besides, the framework will not be restricted by both IoT management platform and business application, it can be independently deployed and applied, and from a more comprehensive and flexible perspective, realize the detection and interception of abnormal devices in the power Internet of Things.

3. Research on Real-time Message Detection Framework Based on Power Internet of Things

3.1. Introduction to Intelligent IoT System

Intelligent IoT System is a complete set of application architecture including the whole business applied to power IoT system, which is divided into four levels of “cloud, pipe, edge and terminal” [5] from top to bottom. As shown in figure 1, "terminal" represents the terminal device at the last level of Intelligent IoT System, responsible for data collection. "Edge" represents the smart device agent, which is responsible for summarizing and forwarding the data reported by "terminal" into the IoT management platform. "pipe" represents the network channel of data transmission, including LTE power wireless private network, public APN, etc., and the communication protocols used include MQTT, Http, Coap, etc. "Cloud" stands for IoT management platform and business system. IoT management platform realizes data collection and forwarding, and finally submit the data to the uppermost business system.
Edge computing framework is a underlying application running on edge agent device, which can support various functions of edge agent and terminal device. The edge computing framework is equivalent to the first "gate" for device data to enter the Intelligent IoT System. If the data sent by the device does not conform to the specifications or the amount of data is abnormal, it should be intercepted in the edge computing framework to prevent it from entering the IoT management platform and upper applications.

3.2 Data Interaction Specification

The comprehensive construction of Intelligent IoT System requires unified access and control of electrical equipment within the whole business scope on the IOT management platform. Data generated by equipment of different manufacturers, categories and functions will be collected to the data center of provinces and cities to achieve business functions. This requires the establishment of universal and unified specification within the scope of the power industry to ensure non-blocking transmission and realize data sharing. In the Intelligent IoT System, the specifications for interaction between equipment and business system include cloud-edge interaction specifications and object-model specifications.

The cloud-edge interaction specification is a protocol specification that defines the interaction mode between the agent device and the IOT management platform, including the device management, container management, application management and business interaction of the IoT management platform to the edge device. All terminal devices and business systems planned to be connected to the Intelligent IoT System must comply with the format specified in the cloud-edge interaction specification for data transmission.

The specification of object-model [6] has unified requirements for data transmission formats of devices connected to different business systems. It mainly classifies devices of the same type as the same product and formulates a data communication model for them as a data transmission specification for devices of the same type to access the Intelligent IoT System. Devices of this type can only send and receive data according to the properties specified in the object-model.

3.3 Real-time Detection Framework

In this paper, the abnormal data real-time detection framework based on power Internet of Things is proposed. According to message verification methods and implementation methods, corresponding
verification rules are formulated for different message types and detection dimensions, and the detection framework is divided into two parts: rule making client and rule execution client, as shown in figure 2. According to the cloud-edge interaction specification and object-model specification, the rule-making client is connected with the business system and the IoT management platform, responsible for the specification analysis and verification rule formulation, and has the ability to deliver the rules to the execution client.

As the rule execution client of the whole detection framework, edge computing framework is mainly responsible for formulating specific verification methods and interception methods according to verification rules, executing interception actions, and reporting processing results to business system and IoT management platform.

3.4. Rule Making Client
In the framework for real-time detection of abnormal devices described in this paper, the rule-making client is connected with the business system and the IoT management platform, and its main responsibility is to complete rule parsing and delivery through preset interaction specifications or manually configured verification rules. The verification rules include data format verification and flow verification, as shown in figure 3.

3.4.1. Data Format Verification Policy. Rule-making client of the real-time detection system provides functions to complete format-based entry of key points for verification. This function first parses the corporate specification or the data interaction format customized by the equipment manufacturer, then enters the format features, required message segments, and required fields in the specification into the system in the form of characters, and finally parses into an array type with sequence. Among them, the required content is used as the array element corresponding to the index, and the index value represents the order. A custom data interaction specification for a business data is shown in figure 4.
The business data contains 14 lines of content, and the form of the corresponding first-level format verification rule is shown in figure 5.

As you can see from figure 4, the starting character in this custom data interaction specification is "{" (see Line 1 of figure 4), and the ending character is "}" (see line 14 of figure 4), where "type": "analog" and so on represent key-value characters, and the separator character between the key-value characters is ",",(for example, at the end of line 2 of figure 4). Also, lines 3-12 in figure 4 represent a key-value pair character containing a structure, and lines 13 represent a key-value pair character, so the array on the left in figure 5 represents the first-level format verification rule. The array can be expressed as "[K1:V1, K2:V2, K3:V3]", which contains seven array elements, as shown in figure 5. The index values of the array elements are 0,1,2... 6. Where, the key-value array element "K1:V1" corresponds to the key-value pair characters in line 2 of figure 4, the key-value array element "K2:V2" corresponds to the key-value pair characters in line 3-12 of figure 4, and the key-value array element "K3:V3" corresponds to the key-value pair characters in line 13 of figure 4. Because the values in the key-value array element "K2:V2" contain structs (where the value V2 is of the form {V2}), the key-value array element "K2:V2" can be used as a target key-value array element and reference the second key-value template.

By referring to the second key-value template, the key-value array element "K2:V2" can be divided into target key-value array element "K2", target key-value join array element ",", target value start array element "{" target value array element V2 "and target value end array element "}" . Based on figure 4, it can be seen that the target value array element "V2" contains two key-value pair characters, namely, the key-value pair characters in line 5 in figure 4 and the key-value pair characters in line 6-11 in figure 4. As shown in the middle array in figure 5, the key-value array elements corresponding to the two key-value pair characters can be represented as "K21:V21" and "K22:V22", respectively. For the key-value array element "K21:V21", its corresponding key-value pair character is the key-value pair in line 5 of figure 4, which does not contain structure, so the first key-value template can be referred to. The key-value array element "K22:V22" contains structs, so the second key-value template can continue to be referenced as long as the number of levels does not exceed the preset threshold. And so on, to complete the message format of the array conversion.

![Figure 4. Interactive format example.](image_url)
The verification process is shown in figure 6. Firstly, the data that the device connects to the IoT management platform through the edge frame must comply with the cloud-edge interaction specification. After the inspection of the cloud-edge interaction specification, the message content inside the data is checked to determine whether it conforms to the object model specification.

Figure 5. Rule array style.

The verification mechanism based on the cloud-edge interaction specification includes two parts: the verification key point and the verification method. The key points of verification include topic format and interactive message format. The verification method is mainly based on the topic and the fields contained in the packet and their sequence.

3.4.1.1. Topic Format Verification. Topic-based publish and subscribe mode is used to communicate between the side device and IoT management platform. The Topic format based on MQTT protocol is specified in the specification of cloud-side interaction as '/ {v1}/edgeId/_topicType/TopicCommand'. The '{v1}' field is used to distinguish between different interaction specifications and the updated version of the interaction format. The {edgeId} field can be used to uniquely identify a device. The {TopicType} field specifies the message Type; The {TopicCommand} field specifies the Command type. The rule-making client converts the above topic format into an array form with index value, as shown in figure 7, and delivers the array to the edge frame to complete the rule-making and distribution.

Figure 6. Equipment data verification process.
3.4.1.2. Verify Exchanged Packets. According to the cloud-edge interaction specification, the request messages exchanged between the device and the IoT management platform are in JSON format, and the request messages and response messages have strict verification fields to ensure the validity of the interaction messages. However, the existing verification operation is only carried out within the IoT management platform. Even the message with wrong format has entered the Intelligent IoT System and participated in data transmission, analysis and processing, resulting in the occupation of resources.

During packet verification, the device first verifies the packet format, whether the device message is reported in JSON format. The JSON format must have the following characteristics: Data is represented in key-value pairs. Key value pairs are separated by commas; Curly braces store objects; Square brackets denote arrays.

JSON format is a widely used message format. The business system or platform only needs to send the verification instruction in JSON format to the edge computing framework, and the edge computing framework calls the method in its own development language to realize the verification of the access data in JSON format.

After the device message passes the verification of the protocol format, the device receives further verification of the message content. Input the fields in table 1 at the rule making client, and deliver the required fields to the rule execution client in the form of array in a manner similar to the topic format test in the previous article.

| Field   | Meaning                                                                 |
|---------|-------------------------------------------------------------------------|
| Mid     | Identifies the mapping between request packets and reply packets         |
| Timestamp | The timestamp of the message sent                                        |
| Expire  | The relative time of packet expiration                                   |
| Type    | message type                                                            |

The edge framework parses the verification rules, compares and queries the data reported by access devices, checks whether the data complies with the cloud-edge interaction specifications, makes control decisions, and reports the results.

3.4.1.3. Verify the Format of Object-model. The "object-model" in the Power Internet of Things is the digital representation of the attributes, topological relations and collection amount of various entity terminals (such as electricity meters, intelligent switches, temperature and humidity sensors, etc.) in the power industry Internet of Things system. From three dimensions of attributes, messages, and services, object model describes the information of model entity, the information and services provided. It is used to standardize various applications of edge IoT agents in Intelligent IoT system, the data of IoT management platform Exchange method, the data association relationship between equipment and business application system, etc. Through model identifier, model description, static
attribute, dynamic attribute, message and service, the object-model conducts standardized modeling of terminal equipment, forming a complete description of the data information and mutual relationship of one or several types of transmission, transmission, distribution and distribution business terminal equipment, and supporting the transmission of terminal equipment data information in the Power Internet of Things.

For the verification of the object-model, whether the device message format conforms to the object-model specification is detected first. The verification method is similar to that of the cloud-edge interaction specification. The key detection points in the object model specification are stored in an array and delivered to the edge computing framework, which completes the verification operation. As shown in figure 8, the object-model information of each type of device is saved in the detection framework, and the attribute fields are resolved and summarized in the form of array. When a device is connected to the IoT management platform through the edge computing framework, the edge framework reads the attribute array of the device's counterpart model from the detection framework, compares and verifies the data reported by the device, and then makes control decisions and reports the results.

**Figure 8.** Process of object model attribute field detection.

3.4.2. **Flow-based Verification Policy.** In the actual operation environment of the Power Internet of Things, there are a large number of devices and a large number of message types. It is inevitable that some devices frequently report due to configuration errors or abnormal device status within a certain period of time, which will cause great pressure on the interface and network resources. As invalid data occupies resources, real data cannot be correctly accessed or transmitted, thus affecting the normal use of service systems.

Flow statistics in the Power Internet of Things should be carried out from three dimensions: message data source, message middleware of message temporary storage, and applications of different businesses. These three dimensions respectively correspond to the device ID in actual operation, the topic in the message middleware, and the application on the device. The real-time detection framework collects flow statistics based on the three dimensions of device ID, topic, and application, so as to comprehensively identify abnormal behaviors of devices. When the real-time traffic exceeds the preset threshold, the corresponding device ID, topic, or application is recorded in the permission control list of the detection framework. As shown in figure 9, the real-time detection framework collects and stores the permission control list reported by the edge framework, forming edge side screening conditions and fault diagnosis basis for business systems.

**Figure 9.** Detection flow based on traffic statistics.
4. Conclusion
With promotion of Intelligent IoT System, IoT architecture covering the whole field in power industry is gradually forming. With further increase in device number and message type, there are higher requirements for real-time detection ability of abnormal devices. Compared with the blacklist mechanism from existing IOT management platform, this protocol extensible abnormal message detection framework, based on Power Internet of Things, has a larger detection range, more flexible deployment, as well as a better adaptability to the extension of protocol and business in later stage. In this paper, framework described still needs to be manually configured in the method of rule entry system, which has certain limitations. In later version, automatic identification function is planned to be imported into upgrade protocol, which can ensure the accuracy of rule entry and improve the overall detection efficiency.

Acknowledgement
This project was supported by State Grid Jiangsu Electric Power Company Information and Communication Research and Application of Key Technologies for Mass Connection and Analysis Capability Optimization of IOT Management Platform (Project Number: JF2021009)

References
[1] Li G J, Cheng L, Wang Z H 2021 Preliminary study on standard framework of Power Internet of Things Electric Power Automation Equipment 41: 1-9.
[2] Zhai W H, Fan B X, Liu Y 2021 Research on the construction and application of business scenario system based on power smart IoT system Electric. Engineering 9: 35-136,139.
[3] Guo X Y, Wu K, Chen X Z 2021 Implementation of power IOT management platform under new infrastructure construction J. Electric. Engineering 11: 187-188.
[4] Shao Y 2009 Web-based remote real-time monitoring framework Journal of Computer Applications B06: 296-298.
[5] Wei X, Pan Z G and Wang B 2020 Review on virtual power plant resource aggregation and collaborative regulation using cloud-tube-edge-end architecture Journal of Global Energy Interconnection 6: 539-551.
[6] Sha M Q, Liu C, Chen L 2021 Design and implementation of intelligent facility management Platform based on object model Electronic Technology & Software Engineering 2: 170-171.