2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W) | 978-1-6654-3950-3/21/$31.00 ©2021 IEEE | DOI: 10.1109/DSN-W52860.2021.00030

Sensitive Instruction Detection Based on the Context of IoT Sensors

Yucheng Wang*†, Xuejun Li*, Peiyang Jia*†, Yiyu Yang†, and He Wang*‡
*School of Cyber Engineering, Xidian University, Xi 'an, China
†National Computer Network Intrusion Protection Center, University of Chinese Academy of Sciences, Beijing, China
‡Corresponding author, Email: hewang@xidian.edu.cn

***Abstract*—With the development of Internet of Things (IoT) technology and the increasing popularity of smart devices, smart homes' IoT devices are becoming more and more intelligent. However, with the access of sensor technology and the gradual increase in the number of devices, some security issues will inevitably occur in smart homes, such as ensuring that the instructions currently executed by the device are legal instructions issued by legitimate users. To solve such challenges, we propose a framework for detecting sensitive instructions. At the same time, we collected a large amount of automation strategy data in smart homes. We used machine learning to extract the sensor cooperative work context characteristics during the execution of sensitive instructions. Finally, We used the test set data to verify and evaluate the performance of our framework. Our evaluation results show that for some sensitive instructions, the goal of being able to intercept high-threat instructions actively has been achieved. The accuracy of the attack detection framework model has also reached more than 89.23%, and some devices even exceed 95%.**

Keywords—IOT, Machine learning, Contextual features, Attack detection

I. Introduction

In recent years, the rapid development of IoT technology has enabled real smart homes to enter people's lives. According to market forecasts, the world's smart home economy will reach 63.2 billion US dollars in 2025[1]. With the increase in IoT devices, related security issues have also attracted widespread attention[2][3]. For example, the malware Mirai used more than 600,000 vulnerable IoT devices to launch attacks, causing a large number of websites[4][5] to disrupt services. At the same time, attackers can attack the sensor in many ways [6], they can use the sensor to leak sensitive information, and even deceive the smart gateway by modifying the parameters of the temperature sensor.

In order to deal with these new threats and challenges, this paper proposes an intrusion detection (IDS) framework. Our framework is based on the sensor context characteristics in the actual activity scenario when the user's instruction is executed. In the design of the framework, the framework can obtain the status of different sensors in real-time and then judge whether the current sensitive command is initiated by a legitimate user according to the sensor context characteristics of the smart home device.

This article mainly introduces the design of an intrusion detection framework based on sensor context characteristics. Simultaneously, Samsung and Xiaomi's two IoT devices are used as objects, and the communication module for real-time communication with IoT sensor devices is designed. It also obtained all the instruction sets of Xiaomi IoT manufacturers' devices, and conducted a survey and analysis of the threat level of the device instructions through a questionnaire survey, and finally used the smart home automation strategy data set to evaluate and optimize the intrusion detection framework. Our evaluation shows that the intrusion detection framework can detect some attacks against IoT devices. The accuracy rates of related models generated by multiple major smart home devices have reached more than 89.23%, and some even exceed 95%.

Contributions: In summary, the contribution of this article has three main aspects.

(1) We designed an intrusion detection framework based on sensor context characteristics, intercepting sensor-based attacks and security threats generated when smart home devices work together.

(2) We evaluated and optimized the intrusion detection framework using the automated command data set. The evaluation shows that the accuracy rate of the frame detection attack has reached more than 89.23%.

(3) Using questionnaire surveys, we divided the threat levels of instructions in the smart home and obtained a high-threat and sensitive instruction data set.

Organizational structure: The rest of the paper's organizational structure is as follows: We outline the attacks and existing solutions against smart home IoT devices in Section II. In Section III, we discussed the threat model and the scope of the research problem. In Section IV, we introduced the detailed design of the intrusion detection framework, including different components. In Section V, the effects of the model framework are discussed by analyzing various performance indicators. Section VI discussed the deficiencies in the work process and the issues to be explored later. At the same time, we discussed in Section VII the close work that others have achieved and concluded this article at the end.

II. Background

This section will introduce the previous work based on IoT security and the necessary background for IoT work.

978-1-6654-3950-3/21/$31.00 ©2021 IEEE
DOI 10.1109/DSN-W52860.2021.00030

A. Previous research

At present, the security of the IoT homes system and equipment has been extensively studied. Existing Internet of Things security research mainly focuses on traditional security issues in the Internet of Things environment, such as equipment or protocol flaws [7], malicious applications [8] , side channels [9] , Platform issues [10][11] and sensor interaction [16].

Regarding device defects, Sivaraman et al.[8] showed that attackers could use malicious mobile applications to carry out multi-step attacks to compromise the local home network from the Internet. Yuan et al.[13] used static analysis and NLP techniques to resolve the inconsistency between application descriptions and their functions. Also, Celik et al.[14] proposed a static pollution analysis tool called SAINT to track sensitive data flows in IoT applications. They [15] also introduced Soteria to detect safety and security violations in IoT applications or IoT environments. Besides, Wang et al.[17] have proved that log information can be used to monitor malicious behavior on the SmartThings platform.

Regarding applications, Pandita et al.[18] compared the results of NLP and static analysis to assess the risks of Android applications. Some researchers use machine learning techniques to assess the risk of malicious Android applications. For example, Jing et al.[19] used support vector machines to give application risk scores based on users' trusted applications. In terms of sensor interaction, HoMonit[20] uses event fingerprints based on network traffic to detect misbehaving smart applications fraud.

B. Smart home IoT platform

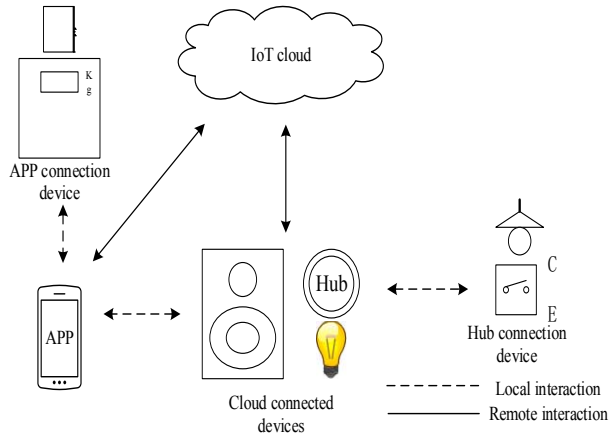

Fig. 1. Smart home IoT platform system architecture

Almost all IoT platforms follow a similar system architecture. As shown in Figure 1, the smart home system comprises four parts: terminal equipment, mobile apps, IoT cloud, and communication network. Terminal devices include smart bulbs, smart door locks, and sensors with different functions. The IoT cloud is generally responsible for storing, managing, and analyzing the terminal device's data using its perception. The mobile app is responsible for functions such as controlling equipment and formulating equipment automation execution strategies. Generally, there are two paths for control equipment. When the mobile app and the terminal device are in the same local area network, some manufacturers support the direct communication between the mobile app and the terminal device; when the mobile app and the terminal device are not in the same local area network, The mobile app needs first to send the instructions to control the device to the IoT cloud. Then the cloud performs the command verification and forwarding to achieve the purpose of controlling the device.

C. Trigger-Action platform

At present, with the development of traditional IOT technology and the addition of sensor devices, more and more IoT vendors allow users to customize automation strategy instructions. That is, the Trigger-Action strategy, when the home environment or equipment meets the Trigger conditions, the Action control instruction is executed.

Due to the convenience of the Smart Home, many Web services automation platforms, such as Home Assistant, IFTTT, and Huginn, have begun to be used to bridge the gap between physical and digital processes. These platforms allow users to write rules to connect the events and actions of IoT devices with those of digital services. For example, when a user receives an email, one direction will turn on the indicator light; when the user's window at home is opened, and there is no one at home, the platform sends the user a message that the window is opened.

The emergence of these platforms and manufacturers to support user-defined instructions has also brought new threats and challenges.

D. Sensitive instruction sensor context

In order to study the needs of smart home security, we need to define a noun: sensitive command sensor context feature. Sensitive instructions are high-threat instructions. When these instructions are issued by an attacker, they will cause direct or indirect harm and loss to legitimate users. For example, opening a window command may cause burglary[21].

The term IoT sensor context has frequently appeared in the field of the IoT recently. In our idea, we describe the definition of sensor context by identifying the information of different sensors that work together in the same scene. Most of the previous context studies are based on the control flow context [22], and data flow context [23] in the IoT devices' program. And our context is based on a feature in the smart home system of the IoT, that is, the interrelationship of devices. When a scene occurs, there will always be multiple intelligent devices and sensors participating.

The sensitive command sensor context feature means that when a sensitive command is executed, the parameters of various sensors in the current environment have a certain correlation. The associated state of different sensors in each scene participating in the same active location is called the sensitive command sensor context feature.

III. Threat Model and Problem Scope

This chapter introduces the security threat model studied in the thesis and the scope of related issues of the security model.

A. Threat model

There is a significant difference between the IoT system and the traditional network system. Various devices in the IoT system can interact with the surrounding environment. When multiple sensors detect changes in the surrounding environment, they will trigger pre-set commands.

However, this will also bring new problems. For example, previous research found that a malicious attacker used the SmartThings SmartApp program to insert malicious code to forge the value of the fire smoke sensor so that the gateway would automatically execute the command "If a fire occurs, open the back door and give an early warning." Afterward, the attacker carried out the user's house's theft, causing the user's loss. One of our goals is to increase the threshold of the above-mentioned sensor-based attacks by providing contextual feature support for sensor collaborative work.

At the same time, our other goal is to improve the safety factor of the execution of sensitive instructions. For example, in Figure 2, when the indoor temperature is lower than the outdoor temperature, the thermostat will start heating; when the thermostat heating makes the indoor temperature too high, the window will be opened. This not only causes a waste of resources but also increases the threat of indoor theft. Therefore, it is also essential to detect and research smart home devices' security threats due to collaborative work.

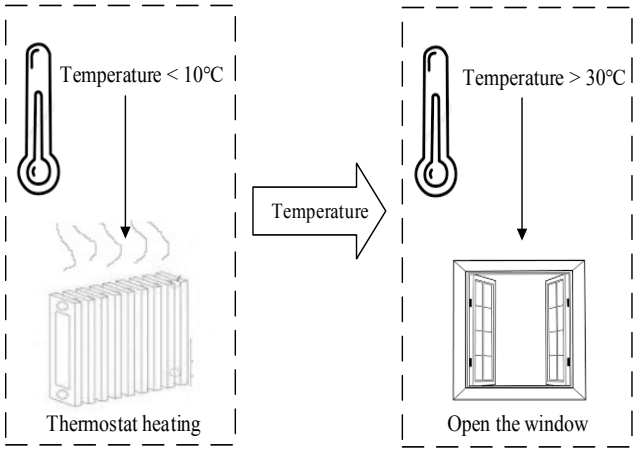

Fig. 2. An example of physical interaction between applications

B. Problem scope

Our design goal is to discover and analyze problem scenarios based on sensors' collaborative work and the device itself on the IoT platform. Therefore, attacks that do not use sensors are not within the scope of our research. For example, we do not investigate problems caused by device or platform vulnerabilities. Attacks against protocol flaws and denial of service (DoS) behaviors and the issue of sensitive information leakage[24] are also beyond our scope.

IV. Contextual Attack Detection Model for IoT Sensors

As shown in Figure 3: Our IoT sensor context model is divided into four parts: sensitive instruction detector, sensor data collector, sensor context feature memory, and instruction judger.

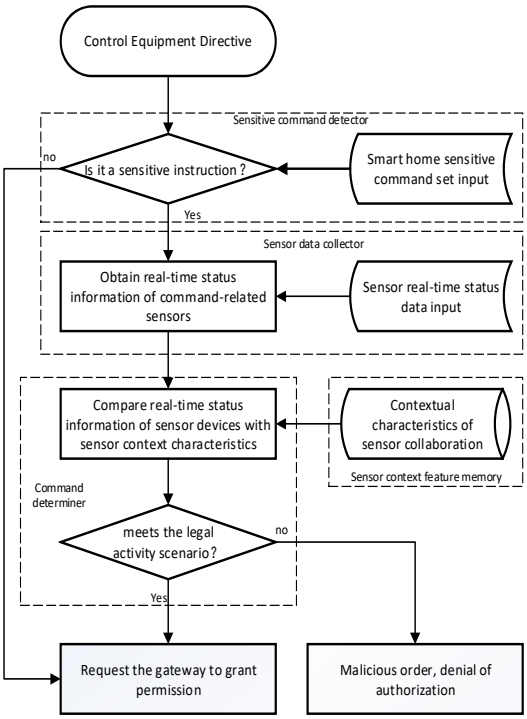

Fig. 3. IDS model framework

A. Sensitive command detector

The sensitive command detector function is to make the first judgment on all the IoT devices' commands, that is, whether they are highly threat-sensitive commands.

First, we need to obtain a complete smart home instruction set. We choose the Xiaomi device to extract the instruction set. Because Xiaomi's smart home equipment covers a wide range, it can increase the integrity of the command data set. Through the reverse analysis of the firmware of the Xiaomi gateway device, we found that all instructions are stored at the address 0x0x102F80 (a function + an instruction) specified in the firmware. Subsequently, we classified the devices involved in the communication instruction set into nine categories in Table I.

We distributed a questionnaire to 340 IoT smart home users. In the investigation, the instructions in the smart home are divided into two categories: control instructions and status acquisition instructions.

Each category of equipment in Table I was investigated for problems, as shown in Table II. We[27] refer to the Grading Standard for General Safety Certification of Smart Home Devices for the classification standard of high, medium, and low threat levels in the questionnaire drafted and released by China Mobile Smart Home Operation Center in July 2020. Therefore, we have also divided the threat levels of the instructions in the smart home. For example, when the execution of a smart home control command may cause the user account password to be easily blasted, sensitive in-formation leakage, etc., then the command is considered to be a high-threat command; when the execution of a control instruction may cause information leakage caused by physical attacks, then the command is considered to

be a medium threat instruction; when the control instruction execution may cause improper user data collection management mechanism, insufficient physical protection capabilities, etc., then the instruction is considered a low-threat instruction.

TABLE. I. The main equipment and classification of IoT smart home

1. Alarms (smoke and fire alarms, flood sensor alarms, combustible gas detection alarms)
2. Kitchen appliances (smart rice cooker, smart dishwasher, smart oven, refrigerator, etc.)
3. Entertainment appliances such as TVs, stereos
4. Air conditioner, thermostat
5. Curtains, blinds
6. Lamp
7. Smart door locks, doors and windows
8. Smart vacuum cleaner and smart lawn mower
9. Security camera

TABLE. II. Description of problems related to device instructions

[Equipment type 5] Curtains, blinds Q1: Please select the statement that you think is correct. ( ) • The control instructions on this type of equipment are highly threatening. • The control instructions for this type of equipment are low threat. • The control instructions on this type of equipment are non-threatening. Q2: Please select the statement that you think is correct. ( ) • Instructions on obtaining the status of such devices are highly threatening. • The status acquisition instructions for such devices are low threat. • The state acquisition instructions on this type of equipment are non-threatening.

As shown in Figure 4, The results of the questionnaire survey show that 85.29% of users believe that the control instructions of the smart home system pose a greater security threat than the status acquisition instructions. Simultaneously, among all the devices owned or expected to own by the surveyed users, up to 91.18% of them are included in the list of devices listed by us. This shows that our research tar-gets' equipment list is rich in categories, sufficient in quantity, and widely covered.

At the same time, it can be seen from the data in Table III that more than 50% of users of alarms, kitchen equipment, air conditioners, curtains and blinds, lights, window smart door locks, and security cameras consider them to be high-threat instructions. Window smart door locks and security cameras account for 94% of the high threats. We defined the instructions that accounted for more than 50% of the survey results' high threats as sensitive instructions.

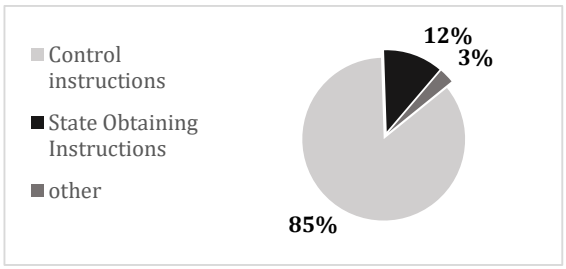

Fig. 4. Threat investigation statistics of different instruction categories

TABLE. III. Threat situation of control instructions for smart home devices

Equipment category	High threat	Low threat	No threat
Alarm equipment	70.59%	26.47%	2.94%
Kitchen equipment	67.65%	32.35%	0%
TV audio equipment	26.47%	73.54%	0%
Air conditioning equipment	52.94%	44.12%	2.94%
Curtain blinds equipment	55.88%	41.18%	2.94%
Lighting equipment	64.71%	26.47%	8.82%
Window equipment	94.12%	5.88%	0%
Sweeping robot equipment	41.18%	52.94%	5.88%
Security camera equipment	94.12%	5.88%	0%

B. Sensor data collector

The sensor data collector's function is to collect the data of the relevant sensors in real-time during the execution of the instruction request. Both Xiaomi and Samsung have the characteristics of early involvement in the industry, relatively complete ecology, and certain influence, so it is more convincing to choose them.

1) Xiaomi

We reversely analyzed Xiaomi APK and smart gateway devices' firmware and analyzed and decrypted their communication data packets. Through these technologies and the developer mode provided by Xiaomi Gateway, the data collection function of Xiaomi sensors is realized.

For the specific implementation process, we unpack and decompile the APK file of Xiaomi Smart Home and then use the data characteristics obtained by analyzing the data packet (such as fixed port number, data packet header, etc.) to locate the key function. Finally, use the key function to reverse the so library to obtain the MD5 and AES_CBC encryption algorithms used in the communication process. Therefore, when we construct the sensor request module's data packet, we use these encryption algorithms to process the data and then use SOCKET communication to send the data to different sensor devices. The device will return the corresponding sensor data.

2) SmartThings

We use Home Assistant as a bridge. Home Assistant is an open-source smart Home system developed based on Python. It supports many smart home device brands and can easily realize the voice control and automation of devices. Its most prominent feature is that it allows devices from different manufacturers to

work together, such as turning on TP-Link's bulb when Samsung's sensors detect that someone is coming back.

We built our Home assistant system on the laboratory server and then deployed the full set of SmartThings smart home equipment we purchased on the Home assistant. The home assistant provides and opens many APIs for developers, including APIs for obtaining status. Only need to obtain the long-term access token in the background management in advance, can we send data packets to the API at any time to query the real-time status data of different sensors bound.

3) Real-time communication

The communication module for acquiring sensor data based on Xiaomi and Samsung devices is our sensor data collector's main body. After it obtains the associated sensor data, it performs simple processing. The data returned by the sensor is mainly divided into two types: logic-oriented discrete values and data-oriented continuous values. For the convenience of calculation, we process all data into unified data in JSON format.

After the sensor data collector obtains the real-time relevant sensor data of the sensitive instruction, the sensitive education enters the next part of the framework's context feature memory.

C. Command sensor context feature memory

Sensor context feature memory is the critical point in our work. We study the classification algorithms in machine learning, such as KNN, support vector machine, Naive Bayes, and decision tree, and find that these algorithms have their advantages and disadvantages. To select a suitable algorithm, we need to analyze the collected data. First of all, there is a vast disparity in the ratio of positive and negative samples in our data set. Secondly, our data include continuous and discrete data, and the classification problem is a classification problem. Therefore, considering various factors, we finally choose decision tree algorithm to realize the construction of the model.

1) Credibility of data

We explain the data set used in machine learning to ensure that the sensor context features we finally obtain are convincing. Our data is the automation strategy instructions in the smart home, as shown in Table IV.

TABLE. IV. Customized automation strategy

• If someone goes home and it is afternoon or later, turn on the lights in the living room.
• When the indoor temperature is too high and there are people in the house, turn on the air conditioning cooling mode.

Data sources are mainly from two aspects. On the one hand, They are the official websites of well-known IoT vendors. They support customization and officially have custom strategy development platforms. On the other hand, web service platforms are derived from automated strategies, such as HA, IFTTT, Zapier, etc. Among them, IFTTT products have attracted 19 million consumers in 140 countries/regions, provide services for 90 million active connections, and support more than 650 devices and services to use IFTTT to secure connections. The widespread use of IFTTT proves that the automation strategy we get from the website is convincing.

We obtained a total of 804 original valid data. Normally, machine learning requires a lot of data to learn a better model. Our data sheet is relatively small in terms of the amount of data. The reason for the small amount of data is that both equipment manufacturers and automation platforms will merge the repeated automation strategy data into one. For example, in IFTTT, for each strategy, the official statistics on the number of users of it, as shown in Figure 5.

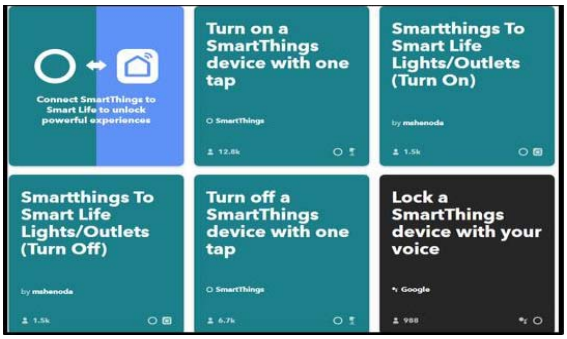

Fig. 5. User usage of different strategie

Therefore, each piece of valid data will generate a large amount of data when multiplied by the number of users of these single pieces of data. In the paper's experiment, we rationally expanded the data set based on the original data to reach a scale suitable for machine learning.

2) Data set context characteristics

Our sample sources are various automated strategy scenarios used by users, resulting in most samples being positive samples. In the machine learning method of uneven samples, there are two under-sampling methods and over-sampling to improve the uneven data set. Combined with the actual situation, we choose the oversampling method to improve the data set.

We studied different machine learning classification algorithms and identified decision trees as our machine learning algorithm. The advantage of the decision tree algorithm is that it is suitable for learning from small sample data sets, is ideal for numerical data and discrete data, and can also obtain the weights of feature attributes. The decision tree builds a classification or regression model with a tree structure. It keeps the decision tree growing by continuously splitting the data set into smaller subsets and finally grows into a tree with decision nodes (including root nodes and internal nodes) and leaf nodes. Generally, decision trees involve three standard methods: information gain, gain ratio, and Gini impurity. We use these features to generate a decision tree.

Different smart home devices correspond to different sensor characteristics, so we need to obtain the sensor context characteristics of all devices. Figure 6 is the feature weight diagram of the decision tree model of whether to open the window. This is a representative model we selected from many device models. From the figure, there are a total of nine features that affect the opening of windows, among which the weight coefficients are smoke sensor, combustible gas sensor, user voice command,

smart door lock status, temperature sensor, air quality detector, outdoor weather, Motion sensor, specific time. From the weight ratio perspective, smoke sensors, combustible gas sensors, user voice commands, and smart door lock status account for a larger proportion and contribute more to the resulting judgment.

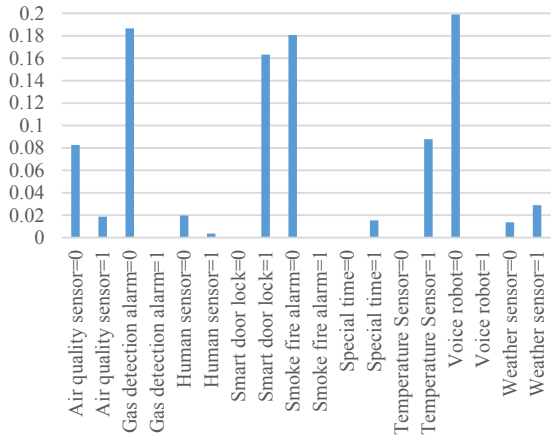

Fig. 6. Window related attribute feature weight map

3) Feature memory

We have established a corresponding decision tree model for equipment related to high-threat instructions. The context features and feature weights of the sensors are calculated and stored, which constitute the context features related to our smart home devices.

D. Command determiner

This summary focuses on the sensitive instruction determiner. We take the trained decision tree model as the core and use real-time relevant sensor data of the device's sensitive instructions as the decision tree model's input. If the output result of the model shows that the real-time data is consistent with the current sensor context characteristics, we believe that the sensitive instructions comply with the legal activity scenario, and the device instructions are allowed to be executed; On the contrary, we believe that the sensitive instruction does not meet the lawful activity scenario and reject the execution of the order.

V. EVALUATION

In this chapter, we will evaluate the generated decision tree models for windows air conditioners , lights, curtains , TV audio, and other entertainment appliances. The evaluation indicators use accuracy, recall, precision, false positive, and false negative.

As shown in Table V，TP stands for true cases, TN stands for true negative cases, FP stands for false-positive cases, and FN stands for false-negative cases.

Accuracy refers to the proportion of correct results that our model predicts. It is defined as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

TABLE. V. Confusion matrix

		Forecast	
		1	0
The actual situation	1	Real Example (TP): ·The real situation: Tor ·Machine learning model prediction results: Tor	False negative (FN): ·The real situation: Tor ·Machine learning model prediction results: no-Tor
	0	False positive case (FP): ·The real situation: no-Tor ·Machine learning model prediction results: Tor	True negative example (TN): ·The real situation: no-Tor ·Machine learning model prediction results: no-Tor

Recall rate, the recall rate represents the proportion of correctly identified data as a positive category among all positive category samples. Defined as follows：

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

Precision (Precision) The precision rate represents the proportion of the positive category in the sample identified as the positive category. It is defined as follows:

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

The false-positive rate (FPR) indicates that the machine model prediction is a positive sample. Still, the actual data is not a positive sample, which accounts for the ratio of total prediction errors. It is defined as follows:

$$FPR = \frac{FP}{FP+TN} \quad (4)$$

The false-negative rate (FNR) can be understood as all data whose true situation is a positive sample, and how many are predicted to be a negative sample (negative sample prediction error), and the definition is given as follows:

$$FNR = \frac{FN}{TP+FN} \quad (5)$$

In the process of machine learning, we divide the data set by 7:3 into the data set and then use the cross-validation method to obtain the results, as shown in Table VI. For different device models, the accuracy of the training set is greater than the accuracy of the test set, indicating that cross-validation has successfully avoided the occurrence of over-fitting. The accuracy of different models has reached more than 89.23%, and kitchen appliances' accuracy even reached 96.43%. The reason is that the eigenvalue types of kitchen appliances are relatively simple, and the data fitting is better. The false alarm rate of all equipment models is almost 0, which means that almost all abnormal data can be accurately identified; At the same time, the underreporting rate is also below 6.67%, indicating that for some normal users' legitimate operations, we are less likely to be misjudged as malicious commands. So it can be seen that the overall model training effect is better.

TABLE. VI. Smart home device model effect

Equipment model	Training set accuracy	Test set accuracy	Recall rate	Accuracy	False alarm rate	False negative rate
window	0.9901	0.9385	0.93694	0.9905	0.0526	0.0631
Air conditioning	1.0	0.9481	0.9333	1.0	0.0	0.0667
light	0.9075	0.8923	0.9375	1.0	0.0	0.0625
Curtains, blinds	0.9796	0.9545	0.9412	1.0	0.0	0.0588
TV, stereo	1.0	0.9473	0.9444	1.0	0.0	0.0556
Kitchen appliances	1.0	0.9643	0.9630	1.0	0.0	0.0370

Door. For door locks, fingerprints, passwords, etc., are required to open outside in the locked state. Regardless of the opening method, its safety is guaranteed by the door lock's corresponding module. And what we are concerned about is the threat situation caused by various linkages. When the smart home equipment meets a certain legal activity scenario, the equipment automatically executes the corresponding function. Therefore, door locks are not within the scope of our research.

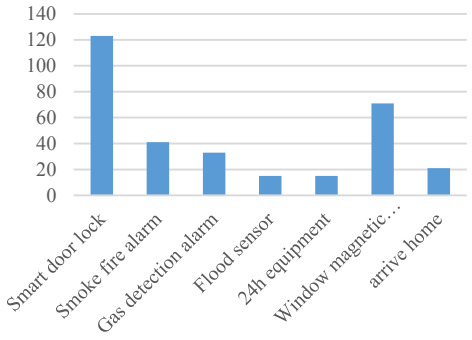

Fig. 7. Camera warning statistics

Security camera. Home security camera, its working mode is to monitor the home 24 hours, and automatically alarm when some critical situations occur. The camera's early warning function is more important, so for the smart home camera, our focus is on which scenes the camera should be linked to realize the part of a warning to the user. We researched the collected automated strategy instructions and counted 319 strategy instructions related to camera warning, as summarized in Figure 7. Statistics have found that users will be alerted when the doors and windows are opened in most cases, and the smoke and fire sensors, water sensors, and combustible gas detectors in the home find abnormalities. Therefore, for the paper's attack detection model, when these situations in Figure 7 occur, we will take the initiative to send an alarm to the user to improve the user's family's anti-risk ability.

Alarm. The role of smoke and fire sensors, water sensors, and combustible gas detectors in the smart home user system is as the Trigger in Trigger-Action, that is, the condition. Therefore, it is not within the scope of the security research object of the smart home equipment system we are discussing.

VI. Discussion

Our thesis's results have achieved the detection of some attacks on smart home IoT devices, and the detection accuracy of different device models can reach more than 89.23%. However, the work of the paper also has some shortcomings. The biggest problem is that we have fewer data types, and we need to obtain more automated strategy instruction data to test and optimize our contextual attack detection model framework. Simultaneously, our model framework is currently only successfully deployed on the devices of two IoT manufacturers, Xiaomi and Samsung, and subsequent research on other manufacturers' machines is needed to expand the application scope of the contextual attack detection model framework. We will also increase the system overhead experiment and further study the attack detection system.

VII. Related Work

In the previous research work on IoT security, some researches are more relevant to our work. The initial research work on IoT security focused on IoT cloud security, APP security, device security, and communication protocol security. Later, as many sensor devices appeared in the smart home equipment system, these sensors can report various environmental attributes and physical event status. Therefore, safety-related research work based on IoT sensors slowly appeared.

The work-related to us first is that Jia proposed a runtime authorization mechanism [12], which uses contextual information to ensure sensitive operations' correctness. They first proposed the concept of contextual information in the IoT. Amit et al.[25] conducted research on smartphones and found that many of the sensors on mobile phones are related to users' actual operating scenarios. They proposed a framework model called 6thSense, which successfully used the correlation between the sensor information on the mobile phone to determine whether the current device has a sensor-based threat. Their work belongs to the research of using sensors in smartphones and has not been applied to the field of IoT devices.

The most relevant to our work is Birnbach et al.[26], who proposed a method that can use sensors associated with a single sensor to verify whether a single sensor is attacked in a smart home IoT scenario. For example, use accelerometers, light sensors, pressure sensors, etc., to verify the status reported by door and window sensors. Because when the door opening action occurs, the door and window sensors, accelerometers, light sensors, and pressure sensors will respond to the opening act. This is the correlation of these sensors. For the first time, Birnbach et al. combined sensors, the IoT, and contextual information to do related research. However, its disadvantage is that the attack detection occurs after the attack event is completed, which means that although sensor-based attacks can be detected, they cannot be prevented. Simultaneously, many of the sensors they use are not deployed in the homes of current smart furniture users, such as accelerometers. Therefore, it cannot be directly promoted to a large number of smart home users.

In short, our thesis's research work, for the first time, uses the context of sensor collaborative work to detect the instructions of the control device in the IoT smart home system and realizes that it can solve the security problems of some IoT devices.

CONCLUSION

In recent years, with the popularization of smart home devices, more and more devices have flooded into thousands of households. But it also brings a lot of security problems. This article has researched IoT devices. First, crawlers are used to obtain a large number of automated instruction strategies. Then machine learning is used to extract the scene features of the collaborative work context of sensors in the IoT. Finally, use sensor characteristics to establish a unified IoT device attack detection model framework. Our framework can detect security issues that occur during the execution of some sensitive instructions. We used automated instruction strategies to evaluate the frame model, and the accuracy rate reached over 89.23%. The accuracy rate of the curtain blinds related instruction model even reached 95.45%, and the false-negative rate was also below 6.67%.

ACKNOWLEDGMENT

This paper is thanks to the 340 smart home users who responded to the questionnaire. And this work was supported by the National Key Research and Development Program of China(2018YFB0804701) and The National Natural Science Foundation of China (No.U1836210)。

REFERENCES

[1] A. Mehra. (2020, Nov.) Home automation system market worth 63.2 bill ion usd by 2025. [Online]. Available: http://www.marketsandmarkets.com/PressReleases/home-automation-control-systems.asp

[2] A. Hesseldahl. (2015, Apr.) A hacker's-eye view of the internet of things . [Online]. Available: https://www.vox.com/2015/4/7/11561182/a-hackers-eye-view-of-the-internet-of-things

[3] B. Schneier, "The internet of things is wildly insecure-and often unpatchable,"Schneier on Security, vol. 6, 2014.

[4] Nicky Woolf. (2016, Oct.) Ddos attack that disrupted internet was large st of its kind in history, experts say. [Online]. Available: https://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet

[5] Wikipedia. (2016, Aug.) Mirai (malware). [Online]. Available: https://en .wikipedia.org/wiki/Mirai_(malware).

[6] A. S. Uluagac, V. Subramanian, and R. Beyah, "Sensory channel threatsto cyber physical systems: A wake-up call," in2014 IEEE Conference onCommunications and Network Security. IEEE, 2014, pp. 301–309.

[7] J. Chen, W. Diao, Q. Zhao, C. Zuo, Z. Lin, X. Wang, W. C. Lau, M. Sun,R. Yang, and K. Zhang, "Iotfuzzer: Discovering memory corruptions iniot through app-based fuzzing." in NDSS, 2018.

[8] L. Wei, W. Luo, J. Weng, Y. Zhong, X. Zhang, and Z. Yan, "Machinelearning-based malicious application detection of android,"IEEE Access,vol. 5, pp. 25 591–25 601, 2017.

[9] D. R. Gnad, J. Krautter, and M. B. Tahoori, "Leaky noise: New side-channel attack vectors in mixed-signal iot devices,"IACR Transactionson Cryptographic Hardware and Embedded Systems, pp. 305–339, 2019.

[10] E. Fernandes, A. Rahmati, J. Jung, and A. Prakash, "Decoupled-ifttt:Constraining privilege in trigger-action platforms for the internet ofthings,"arXiv preprint arXiv:1707.00405, 2017.

[11] Y. J. Jia, Q. A. Chen, S. Wang, A. Rahmati, E. Fernandes, Z. M. Mao,A. Prakash, and S. Unviersity, "Contexlot: Towards providing contextualintegrity to appified iot platforms." inNDSS, vol. 2, no. 2, 2017, pp. 2–2.

[12] C. Nandi and M. D. Ernst, "Automatic trigger generation for rule-based smart homes," inProceedings of the 2016 ACM Workshop onProgramming Languages and Analysis for Security, 2016, pp. 97–102.

[13] Y. Tian, N. Zhang, Y.-H. Lin, X. Wang, B. Ur, X. Guo, and P. Tague,"Smartauth: User-centered authorization for the internet of things," in26th{USENIX}Security Symposium ({USENIX}Security 17), 2017, pp.361–378.

[14] Z. B. Celik, L. Babun, A. K. Sikder, H. Aksu, G. Tan, P. McDaniel,and A. S. Uluagac, "Sensitive information tracking in commodity iot," in27th{USENIX}Security Symposium ({USENIX}Security 18), 2018, pp.1687–1704.

[15] Z. B. Celik, P. McDaniel, and G. Tan, "Soteria: Automated iot safetyand security analysis," in2018{USENIX}Annual Technical Conference({USENIX}{ATC}18), 2018, pp. 147–158.

[16] E. Fernandes, J. Paupore, A. Rahmati, D. Simionato, M. Conti, andA. Prakash, "Flowfence: Practical data protection for emerging iot application frameworks," in25th{USENIX}security symposium ({USENIX}Security 16), 2016, pp. 531–548.

[17] Q. Wang, W. U. Hassan, A. Bates, and C. Gunter, "Fear and logging inthe internet of things," inNetwork and Distributed Systems Symposium,2018.

[18] R. Pandita, X. Xiao, W. Yang, W. Enck, and T. Xie, "{WHYPER}:Towards automating risk assessment of mobile applications," in22nd{USENIX}Security Symposium ({USENIX}Security 13), 2013, pp.527–542.

[19] Y. Jing, G.-J. Ahn, Z. Zhao, and H. Hu, "Riskmon: Continuous andautomated risk assessment of mobile applications," inProceedings ofthe 4th ACM Conference on Data and Application Security and Privacy,2014, pp. 99–110.

[20] W. Zhang, Y. Meng, Y. Liu, X. Zhang, Y. Zhang, and H. Zhu, "Homonit:Monitoring smart home apps from encrypted traffic," inProceedings ofthe 2018 ACM SIGSAC Conference on Computer and CommunicationsSecurity, 2018, pp. 1074–1088.

[21] T. Budd, "Burglary of domestic dwellings: Findings from the britishcrime survey," 1999.

[22] A. Agarwal, S. Dawson, D. McKee, P. Eugster, M. Tancreti, and V. Sundaram, "Detecting abnormalities in iot program executions throughcontrol-flow-based features," in2017 IEEE/ACM Second InternationalConference on Internet-of-Things Design and Implementation (IoTDI).IEEE, 2017, pp. 339–340.

[23] N. Boltz, M. Walter, and R. Heinrich, "Context-based confidentialityanalysis for industrial iot," in2020 46th Euromicro Conference onSoftware Engineering and Advanced Applications (SEAA). IEEE, 2020,pp. 589–596.

[24] E. Fernandes, A. Rahmati, J. Jung, and A. Prakash, "Decentralized actionintegrity for trigger-action iot platforms," inProceedings 2018 Networkand Distributed System Security Symposium, 2018.

[25] Z. B. Celik, G. Tan, and P. D. McDaniel, "Iotguard: Dynamic enforce-ment of security and safety policy in commodity iot." inNDSS, 2019.

[26] S. Birnbach and S. Eberz, "Peeves: Physical event verification in smarth omes," 2019.

[27] C Mobile. (2020, Dec.) Smart Home. [Online]. Available: https://open.home.10086.cn/openhomePortal/pages/homepage/bulletin/bulletinDetail/?id=20201221000002