A Survey on Detecting Interactions among Different Devices/Apps in IoT

Yicheng Zhen[1], Yeonjoon Lee[2]
[1]Major in Bio Artificial Intelligence, Department of Computer Science and Engineering, Hanyang University
[2]Department of Computer Science and Engineering, Hanyang University

IoT 분야의 다양한 기기/앱 간 상호작용 검출에 관한 연구동향

전이정[1], 이연준[2]
[1]한양대학교 컴퓨터공학과 바이오인공지능융합전공 석사과정
[2]한양대학교 컴퓨터공학과 교수

zyc0928@hanyang.ac.kr, yeonjoonlee@hanyang.ac.kr

Abstract

With the recent advances in communication technology and Internet of Things (IoT) infrastructure, home automation systems have emerged as a new paradigm for providing users with convenient smart home services. The IoT ecosystem has merged digital systems with the physical world, dramatically changing the way people live and work. However, at the same time, security remains one of the most significant research issues in IoT, as the deployment and application of high-availability systems come with various security risks that cause serious threats to users. Among them, the security issues arising from the interaction among devices/applications should not be underestimated. Attackers can exploit interactions among devices/applications to hack into the user's home. In this paper, we present a survey of research on detecting various types of interactions among devices/applications in IoT.

1. Introduction

The Internet of Things (IoT) is a new way of connecting objects to the Internet and gathering data from sensors, enabling the remote control of appliances, machines, and other areas, such as buildings, vehicles, and healthcare [1-2]. IoT systems are becoming ubiquitous in various cyber-physical infrastructures [3]. IoT has dramatically changed home automation due to the exponential growth of IoT devices [4]. While IoT technologies can offer many conveniences, community and user concern about the security and privacy of smart home environments are rapidly increasing [5].

In particular, safety and security issues resulting from interactions among devices/applications are of great concern. For instance, as shown in Figure 1, an interaction between a vacuum and a door can create a security problem when two apps run simultaneously on the same device.

In this paper, we explain the different types of interactions among different devices/applications and the potential dangers when such interactions occur. Additionally, we present methods for detecting these interactions in IoT environments.

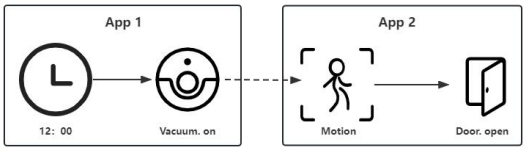

Figure 1: An example of security problem

2. Background

In the IoT environments, there are two types of interactions that are used in association with IoT applications and devices: cyberspace interaction and physical interaction.

2.1 Cyberspace interaction

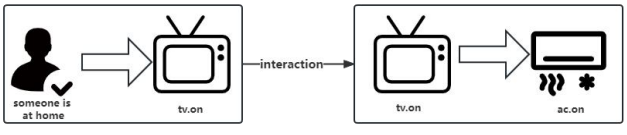

Figure 2: An example of cyberspace interaction

Cyberspace interaction refers to the interaction of different applications with each other through common devices or system events in cyberspace [6]. In multi-application IoT

systems, cyberspace interactions can lead to insecure and unsafe states. For instance, as shown in Figure 2, one app turns on the TV when someone is at home, and another app turns on the air conditioner when the TV is on. These two apps interact through tv. on events on the same device and share a common device attribute on the same IoT platform.

2.2 Physical interaction

Physical interaction, on the other hand, occurs when IoT devices/applications affect the physical environment, and changes in the physical environment (e.g., changes in temperature, humidity, brightness, etc.) may trigger the behavior of other IoT devices [6]. For instance, as shown in Figure 3, a heater raises the room temperature, and then another device opens the window after detecting the increased temperature. In this case, the physical quantity of temperature connects the heater to the temperature sensor, creating a physical interaction. An attacker can use a heater to raise the room temperature and then automate the rules to detect the temperature increase and open the window [7]. Physical interactions can put users in insecure and unexpected situations, and they can be exploited by advanced attackers to launch IoT attacks.

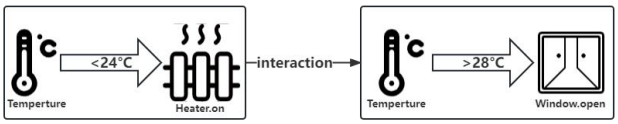

Figure 3: An example of physical interactions

3. Method

In this section, we will introduce two systems, IoTGUARD and IoTSAFE.

Both systems are dynamic solutions that enforce security policies at runtime; IoTGUARD mainly accounts for cyberspace interactions in multi-application IoT environments, while IoTSAFE aims to capture real physical interactions.

3.1 IoTGUARD

IoTGUARD [8] is a system for the dynamic enforcement of security and safety policies in commodity IoT devices. It is designed to monitor and analyze interactions between different applications and devices in real-time and enforce policies based on the detected interactions to protect users from unsafe and insecure device states. It consists of three components: Code Instrumentor, Data Collector, and Security Service. First, application information is collected at runtime by inserting additional logic into the application's source code. Then, Data Collector stores the application information in a dynamic model that reflects the actions taken by the application at runtime. Finally, Security Service checks whether the application's actions pass the policy and actions that violate the policy are blocked or allowed for the application, depending on the response of the security service.

Celik et al. evaluate IoTGUARD on 35 SmartThings IoT and 30 IFTTT trigger-action apps. The results show that policy enforcement leads to an average runtime overhead of 17.3% for a single application and 19.8% for five interactive applications.

3.2 IoTSAFE

IoTSAFE [6] is a dynamic safety and security policy enforcement system for multi-application IoT environments that protects users from unsafe and insecure IoT device interactions in a preventative manner. It is designed to capture real physical interactions between IoT devices. It first generates static interaction graphs by extracting the trigger conditions and corresponding actions through the code analysis module. Then, it uses the application configuration and room information to generate test cases to further simulate the user's real-world environment. Based on the generated cases, it identifies device physical interactions by performing dynamic tests on device/condition constraints. It then uses the physical model to predict future states based on runtime events and updates the model using temporal interaction data collected during dynamic testing and online data from untested or newly added devices. IoTSAFE uses the policy management application to set user-defined policies and performs violation identification and mitigation through the control server, monitors device runtime information through the data collection, runtime prediction, and violation detection modules, and compares current/predicted conditions to user-defined policies to mitigate risk conditions.

Ding et al. implement the prototype of IoTSAFE on the SmartThings platform. And they evaluate IoTSAFE in a simulated smart home environment. IoTSAFE identifies 39 real physical interactions from 21 applications.

4. Conclusion

This paper discussed the different types of interactions among different devices/applications and introduced the roles and main principles of IoTGUARD and IoTSAFE. The experimental results from Celik et al. and Ding et al. demonstrate that both systems can detect interactions among different devices/applications. The differences between the

<TABLE 1>: Differences between the two methods

	Intermediate Representation	Environment Analysis	Enforcement
IoTGUARD	Dynamic model	No	Runtime Policy Enforcement
IoTSAFE	Physical model	Yes	Safety and Security Policy Enforcement

two methods are presented in TABLE 1.

ACKNOWLEDGMENT

This work was partly supported by Institute of Information & communications Technology Planning &Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2022-00155885, Artificial Intelligence Convergence Innovation Human Resources Development (Hanyang University ERICA)) and Institute of Information & communications

Technology Planning& Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2020-0-01343, Artificial Intelligence Convergence Research Center(Hanyang University ERICA)).

Reference

[1] Paul Comitz and Aaron Kersch. Aviation analytics a nd the Internet of Things. In Integrated Communica tions Navigation and Surveillance, pages 2A1‐1, Herndon, VA, USA, 2016. IEEE.

[2] MinghaoWang, Tianqing Zhu, Tao Zhang, Jun Zhan g, Shui Yu, andWanlei Zhou. Security and privacy in 6G networks: New areas and new challenges. Di gital Communications and Networks, 6(3):281‐291, 2020

[3] Pradeep, Pavana, and Krishna Kant. "Conflict detect ion and resolution in IoT systems: a survey." IoT 3. 1 (2022): 191-218.

[4] Ban, Xinbo. Protect Smart Homes from Inter-Rule Vulnerabilities: Large-Scale Testbed, Static and Dyn amic Techniques. Diss. SWINBURNE UNIVERSIT Y OF TECHNOLOGY, 2023.

[5] Babun, Leonardo, et al. "A survey on IoT platform s: Communication, security, and privacy perspective s." Computer Networks 192 (2021): 108040.

[6] Ding, Wenbo, Hongxin Hu, and Long Cheng. "IoT Safe: Enforcing safety and security policy with real IoT physical interaction discovery." the 28th Netw ork and Distributed System Security Symposium (N DSS 2021). 2021.

[7] Wang, Zhibo, et al. "A Survey on IoT-enabled Ho me Automation Systems: Attacks and Defenses." IE EE Communications Surveys & Tutorials (2022).

[8] Celik, Z. Berkay, Gang Tan, and Patrick D. McDan iel. "IoTGuard: Dynamic Enforcement of Security a nd Safety Policy in Commodity IoT." NDSS. 2019.