The 5th International Symposium on Mobile Internet Security

MobiSec 2021

October 7 – 9, 2021
Jeju Oriental Hotel,
Jeju Island, South Korea

Organized by

KIISC Research Group on 5G Security

Hosted by

Korea Institute of Information Security and Cryptology (KIISC)

Sponsored by

Huawei Korea

Electronics and Telecommunications Research Institute (ETRI)

MemTwin: Consecutive Memory-Recording enhanced with FPGA for IoT Memory Inspection

Hye Lim Jung, Gi-Choel Choi, and Ki-Woong Park*
SysCore Laboratory, Sejong University, Seoul, Korea
hyello13@gmail.com, woongbak@sejong.ac.kr

Abstract

The IoT platform environment consists of hundreds of thousands of distributed IoT devices and is applied to individuals, businesses, and national facilities to provide services. This makes it difficult to determine the cause, and analyze the failure, of the IoT system when an external attack occurs. Because of this, an analyst performs an in-depth analysis of the IoT system and monitors it through various methods. However, it is difficult to identify the time of an attack in many IoT devices; therefore, this study proposes the *MemTwin* method to solve this problem. *MemTwin* performs a high-speed memory dump on IoT devices and archives the collected memory as a single image to confirm the IoT memory at the time the analyst wants. *MemTwin* achieves this through IoT memory, which can process FPGA -based parallel circuits and performs compression storage operations by extracting only data differences between the memories generated between each dump, thereby solving storage problems that occur in the existing memory dump process. In addition, by using a memory recovery mechanism, a user of the IoT platform can sequentially analyze the memory in depth for IoT device management.

Keywords: IoT, Memory Dump, Data archiving

1 Introduction

The IoT platform environment consists of hundreds of thousands of distributed IoT devices and is applied to individuals, businesses, and national facilities to provide services. According to the development of IoT technology, IoT platforms can be applied to smart factories, smart buildings, smart grid systems, companies, institutions, and national infrastructure beyond the scope of personal use, such as smart plugs, smart bulbs, and smart door locks [9, 10] The threat to the IoT platform is expanding beyond the scope of individuals to companies and countries. As a result, security monitoring has become an essential solution for maintaining a secure IoT platform [15]. However, the IoT platform, which constitutes the core infrastructure, is composed of complex forms, such as interconnections between IoT devices, data management systems, and firmware [11, 6]. To prevent threats, the Smart Control Center for monitoring IoT platforms is performing checks for IoT devices and system threats that makeup the IoT platform. However, the IoT platform configuration has a complex configuration, such as data generated through the operation of the sensor information sharing and applications collected between the devices that make up the platform [15, 5, 14]. Therefore, even if a problem occurs in one IoT device, it can have an adverse effect on the entire platform [15, 14] . This makes it difficult to determine the cause and analyze the failure of the IoT system when an external attack occurs. Therefore, the application of a systematically advanced monitoring solution for IoT devices is required as an essential element for threat analysis in the IoT platform.

H. Kim, F. Palmieri, T.-Y. Youn (eds.), *The 5th International Symposium on Mobile Internet Security (MobiSec'21)*, Jeju Island, Republic of Korea, October 7-9, 2021, Article No. 2, pp. 1-8

*Corresponding author: Department of Computer and Information Security, Sejong University, Seoul 05006, South Korea

To build an advanced IoT platform security monitoring environment, it is necessary to address the resource problems of IoT devices, increase the risk of cyberattacks targeting IoT devices that are developing in real time, and the internal operation failure of IoT devices operating at a distance. Considering these issues, it is necessary to apply a light solution that can be continuously updated to IoT devices for a safe monitoring environment; however, IoT devices are produced by numerous sensors through various vendors. This makes it difficult to solve the problem in a short period of time and apply the solution that can prevent the threat of all IoT devices.

To minimize the threat that occurs in this environment, the memory, which is a basic component of the IoT system, is monitored in real time rather than applying an independent solution to each IoT device. The necessity of configuring the IoT platform through memory data analysis to detect abnormal signs and find the cause of the remote IoT device in real time is increased.

However, memory dump operation for memory data collection is accompanied by a suspension phenomenon (a system-wide pause phenomenon) to minimize the variation in memory data generated during the operation. As the memory capacity required for memory dump operation increases, the suspension development time increases proportionally. This impedes the service availability of normal IoT devices. In addition, IoT devices often have low computing resources because of their characteristics, and the suspension phenomenon caused by memory dump operations can have a greater impact on IoT devices than normal computing systems. For example, because the size of memory information collected through memory dump operation is proportional to the size of the memory, the available storage space of the IoT device in operation is continuously consumed because the space for temporarily storing the memory dump data is required before sending it to the monitoring system. Therefore, the proposed mechanism minimizes the suspension phenomenon by performing the replication of the entire memory data to one clock through the parallelized memory dump operation based on FPGA to solve these limitations. *MemTwin* performs a high-speed memory dump on IoT devices and archives the collected memory as a single image to confirm the IoT memory at the time the analyst wants. Accordingly, *MemTwin* can minimize the consumption of available storage capacity of an IoT device by extracting only essential parts, where data is changed by system operation according to time flow, by sequentially storing the memory dump data and the delta (real time change) values that have been performed by XOR operation before one step. In addition, XOR calculation is performed on memory data collected in reverse order, using a delta value based on the memory data at a specific point, where memory data are collected to obtain original memory data in a previous step. This provides a mechanism for in-depth monitoring and memory analysis of IoT devices operating in real time. In section 2, this study describes related studies and the advantages of the proposed mechanism. Section 3 describes the structure and workflow of the proposed mechanism. Section 4 presents the conclusions of this study.

2 Related Work

The *MemTwin* mechanism proposes a method to efficiently monitor various threats that may occur in an IoT environment. In this section, we explain the research on monitoring in the IoT environment and explain the advantages of the proposed mechanism. Many studies have been conducted regarding threats in the IoT platform environment [**?**, 7, 13], this study highlights some of them. A monitoring study [8] analyzed threats that may occur in IoT, generated logs at a point where threats may occur, and analyzed them on the server. The study [12] which monitored IoT through blockchain used a private block chain architecture to monitor the integrity of IoT devices and restore device configuration settings . In addition, the study [7] which monitored IoT network information was a more systematic study because it used plug-ins for each protocol. Commercialized IoT monitoring platforms [2, 4, 1, 3] collect sensor

information of IoT, application, and device log information to perform monitoring.
Research that monitored the response to threats from IoT has been conducted as well. The proposed mechanism collects memory information for monitoring IoT, therefore, it is possible to confirm the overall situation of IoT, such as applications, networks, and sensor devices. In addition, the *MemTwin* mechanism performs a suspension phenomenon that may occur during the process of performing a memory dump with an FPGA-based parallel circuit, thereby performing a memory dump more efficiently. Because memory information can be continuously collected by reducing suspense, the memory of the IoT can be recorded as a video. Accordingly, the analyst can recover and reproduce the IoT memory at the point in time when analysis is requested in the event of an IoT failure. The analyst can obtain information recorded in the memory of the log, and information that was recorded in the situation at the time. *MemTwin* collects memory continuously because memory dumps are performed in parallel and collected more quickly. The collected memory dump data archives only the change value of the previous memory through the XOR. In the archiving process, if the difference between the memories is not large, it can be efficiently compressed, to enable the memory archiving of the IoT to be performed to a minimum.

3 System Architecture of *MemTwin*

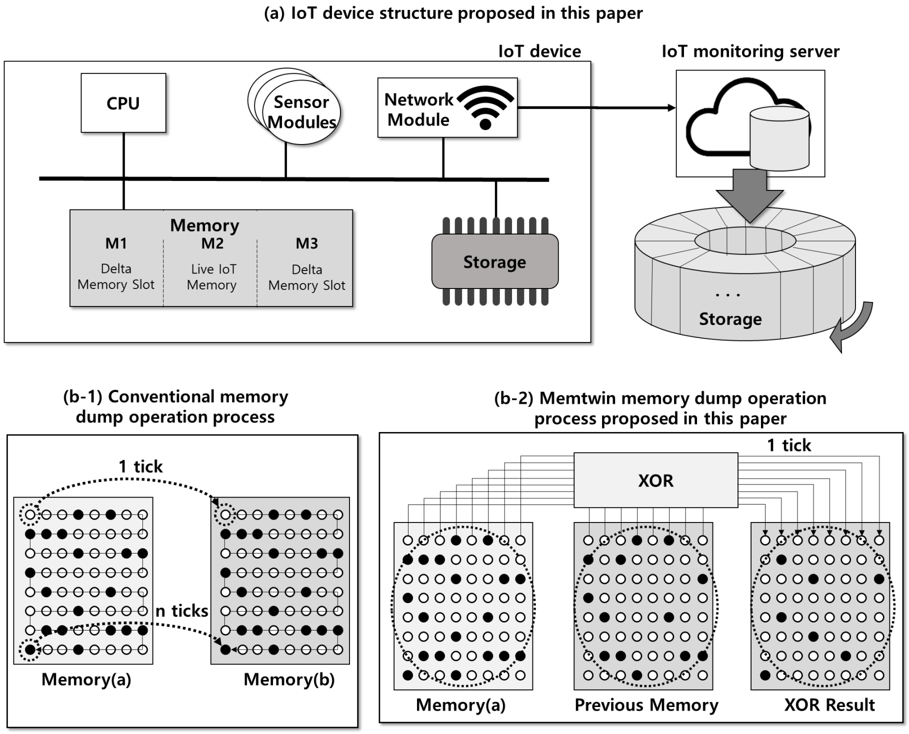

Figure 1: *MemTwin*'s IoT Device Structure and the Memory Structure Concept Applying FPGA-based Parallel Processing Circuit.

In this section, we explain the structure and workflow of the *MemTwin* mechanism proposed in this study. The mechanism-driven environment proposed in this study is similar to a general system with CPU, memory, and storage, as shown in Figure 1 (a). To utilize the technology presented in this study, the memory of IoT devices is divided into delta memory slot and live IoT memory. The live memory slot is a section where systems such as firmware operate; the delta memory slot is a temporary storage space consisting of memory where the backed-up data is temporarily stored. The delta memory slot is a space where delta data generated by XOR calculations during the memory dumping process is stored. Figure 1b compares the process of collecting and computing the real-time memory continuous collection technique of the low-lying mode of this mechanism, and the IoT memory reproduction technology according to the parallel circuit with the existing memory dump operation process. The latter dumps the memory cell in memory (a) to memory (b), and each cell moves in one second. However, according to the parallel processing memory dump operation process proposed in this mechanism, the cell moves and calculates from memory (a) to memory (b), and the entire dump is possible by parallel circuits in one second. Thus, in this mechanism, this process allows relatively low-latency situations in which IoT devices suspend or delay computational performance for dumps even during computational performance. This allows IoT device memory dump collection to be collected more quickly and continuously.

The *MemTwin* mechanism performs high-speed memory dump operations for collecting, monitoring, and in-depth analysis of memory information from IoT devices in real time. In the IoT memory dump process, *MemTwin* utilizes the memory of the IoT device consisting of the parallel circuit of the FPGA base to accomplish the dump. This way, the system suspension phenomenon generated in the memory dump process can be minimized. The detailed memory dump process of *MemTwin* is described in subsection 3.1. In addition, the *MemTwin* mechanism proposed in this study solves the storage resource consumption problem of existing memory dump data by securing light memory dump data, collecting only the changed part, and comparing memory dump data at the present time and memory dump data of the previous period. *MemTwin* can reverse the acquisition of memory dump data at a specific point in time when analysis is required by using a delta measurement value that has information about the memory part changed, owing to the operation of the system during the memory dump process. By recovering and analyzing IoT memory states operating at that point, we propose a mechanism that can efficiently monitor time series-type system data and detect abnormal signs. The process of restoring memory situations at a particular point using delta data is described in section 3.2.

3.1 Memory Snapshot Workflow of *MemTwin* Mechanism

The *MemTwin* mechanism for memory data collection and monitoring of IoT devices performs high-speed memory dump using FPGA-based parallel circuits. The mechanism works in an IoT device environment consisting of a CPU, memory, and storage. Figure 2 shows the process of dumping in IoT memory. The memory of the IoT device is categorized into the live memory slot(M2), backup memory slot (M1), and delta memory slot (M3), which has a memory structure including an FPGA-based parallel processing circuit. All data located in the backup memory slot (M1) and the delta memory slot (M3) in *MemTwin* are initialized to zero. For the XOR operation (2 in Figure 2) of *MemTwin*, the delta memory slot selector (1 in Figure 2) selects M3 on the even number of tn and M1 on the odd number of tn. The XOR operation proceeds in the delta memory slot and the selected live memory slot (M2). The results are stored in the delta memory slot, which is not selected by the selector. The delta value stored in the delta memory slot is stored through the run-length encoding (3 in Figure 2) process, through the *MemTwin*. The run-length encoding process (3 in Figure 2) is a process for minimizing data in an IoT device environment with insufficient resources. In the memory XOR result, only the changed part of the memory is extracted. The smaller the changed part, the larger the amount of duplicated data. Therefore, the size of the data can be minimized through the run-length encoding (3 in Figure 2) process.

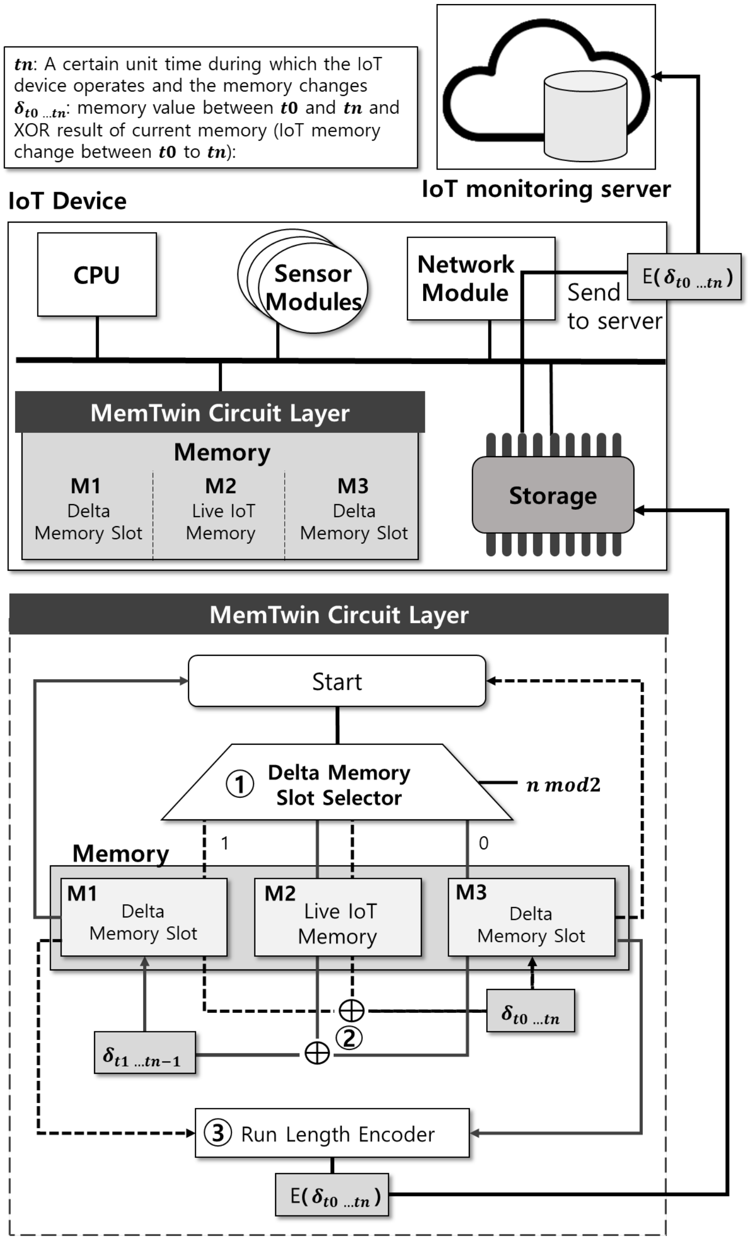

Figure 2: Memory dump operation process of *MemTwin* in IoT device environment

3.2 Memory Restoration Workflow of *MemTwin* Mechanism

In the *MemTwin* mechanism environment, when a user wants to recover memory data at a specific point in time of the IoT device through a monitoring environment, the user restores the memory dump data stored on the IoT device to a delta-memory value through a run-length decoding process. The re-stored delta value performs live memory data and XOR operations currently used in the system, and the results of the XOR operations can be restored to the dump data at the previous point of the most recent dumped memory. If the XOR operation is repeated to delta data generated in the restoration process at specific times, the user can restore the memory dump data of the point in time the user wants. In the process of performing the function of the IoT apparatus, when memory analysis is required, the user can perform the memory dump more rapidly through the dump process proposed in this mechanism without

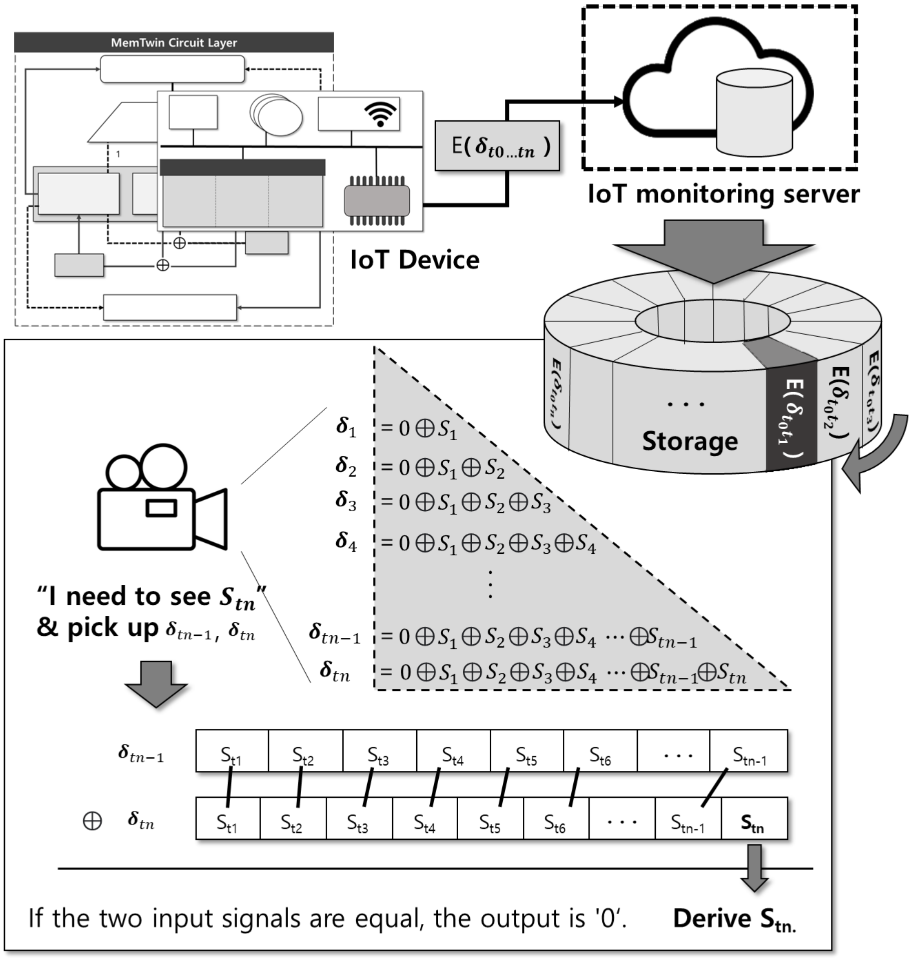
MemTwin Circuit Layer
E($\delta_{t0...tn}$)
IoT monitoring server
IoT Device
Storage
E($\delta_{t_0t_1}$)
E($\delta_{t_0t_2}$)
E($\delta_{t_0t_3}$)
E($\delta_{t_0t_n}$)
"I need to see S_{tn}" & pick up δ_{tn-1}, δ_{tn}
$\delta_1 = 0 \oplus S_1$
$\delta_2 = 0 \oplus S_1 \oplus S_2$
$\delta_3 = 0 \oplus S_1 \oplus S_2 \oplus S_3$
$\delta_4 = 0 \oplus S_1 \oplus S_2 \oplus S_3 \oplus S_4$
$\delta_{tn-1} = 0 \oplus S_1 \oplus S_2 \oplus S_3 \oplus S_4 \cdots \oplus S_{tn-1}$
$\delta_{tn} = 0 \oplus S_1 \oplus S_2 \oplus S_3 \oplus S_4 \cdots \oplus S_{tn-1} \oplus S_{tn}$
δ_{tn-1} St1 St2 St3 St4 St5 St6 ... Stn-1
$\oplus$ δ_{tn} St1 St2 St3 St4 St5 St6 ... Stn-1 Stn
If the two input signals are equal, the output is '0'.
Derive Stn.

Figure 3: Memory data recovery operation process at a specific point in *MemTwin*

the pause of the IoT function and calculation without much delay. *MemTwin* can reduce the capacity by continuously dumping memory at a specific period, analyzing the changes in the previous memory dump and the current memory dump in time series units, and storing the changed values using a run-length encoding-based compression algorithm. For a user to obtain memory data at a specific point in time through the memory recovery process of *MemTwin*, only the memory change value and the immediately preceding point in time can be obtained through calculation. The memory dump process of the IoT device is continuously and rapidly recorded throughout the process; and thus, the data can be re-stored in the memory at the time when abnormal signs occur during the IoT in-depth analysis process. Therefore, the IoT device user can grasp the problem more quickly. In addition, when an IoT platform user wants to remotely perform in-depth analysis of the target IoT, by applying the *MemTwin* mechanism, the initial memory dump data, and a small amount of memory dump data from which the memory dump is changed afterwards are transmitted to the server. Thus, the delay is minimized. *MemTwin* can be restored to a specific point in time in IoT memory through a high-speed memory dump and archiving process through an FPGA-based parallel processing circuit, which can improve efficiency and economic efficiency in IoT system analysis. Figure 3 shows the IoT memory restoration process of *MemTwin*. Delta data are stored in the mode in which the XOR operation is in real-time IoT memory dump and the previous XOR result value, is added to the previous result. Thus, it is possible to shorten the operation in the memory restoration process at a point in time when analysis is required. The analyst can perform the memory restoration with a simple calculation by performing the XOR operation with the Delta data of the point in time when it tries to restore without using the whole XOR data of the immediately preceding point of time. The server can perform XOR for restoration by decompressing the memory at the desired point in time through the run-length decoder. As the memory dump is performed faster through the *MemTwin*, it is possible to continuously record the operation process of the IoT. This enables IoT administrators to perform problem identification faster.

4 Conclusion

In this study, we proposed a mechanism that can perform more efficiently, the memory collection of IoT to monitor the problems and cyberattacks that may occur in IoT devices. The mechanism proposed is for the replication of the entire memory data to one clock through an FPGA-based parallelized memory dump operation to reduce the suspension phenomenon that may occur during memory dump operations. The memory dump data is collected through the mechanism, and the continuous memory dump data is restored to the time of the failure during the IoT accident analysis process. In addition, the mechanism performs XOR operation on the dumped memory data to archive only the difference value to collect only the changed part, and continuous memory collection is restored to the memory at the time of the problem. Because the XOR operation saves only the changed value of the memory dump data, it is possible to perform more efficient compression through the run length encoder. Through the mechanism proposed in this study, IoT analysts can play IoT memory as a video.

Acknowledgments

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. 2019-0-00426,Development of active kill-switch and biomarker based defense system for life-threatening IoT medical devices, 20%) and National Research Foundation of Korea (NRF) (NO.NRF-2020R1A2C4002737, Intelligent Incident Investigation Platform for IoT CERT, 80%)

References

[1] Aws iot core, https://aws.amazon.com/iot-core/.

[2] Azure iot, https://azure.microsoft.com/en-gb/overview/iot/.

[3] Google cloud iot core, https://cloud.google.com/iot-core.

[4] Ibm watson iot platform, https://internetofthings.ibmcloud.com/.

[5] M. Al-Kuwari, A. Ramadan, Y. Ismael, L. Al-Sughair, A. Gastli, and M. Benammar. Smart-home automation using iot-based sensing and monitoring platform. In *Proc. of the 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG'18), Doha, Qatar*, pages 1–6. IEEE, June 2018.

[6] A. Alsalemi, Y. Al Homsi, M. Al Disi, I. Ahmed, F. Bensaali, A. Amira, and G. Alinier. Real-time communication network using firebase cloud iot platform for ecmo simulation. In *Proc. of the 2017 IEEE International Conference on iThings (iThings'17) and IEEE Green Computing (GreenCom'17) and IEEE Cyber, Physical and Social Computing (CPSCom'17) and IEEE Smart Data (SmartData'17), Exeter, UK*, pages 178–182. IEEE, 2017.

[7] V. Casola, A. De Benedictis, A. Riccio, D. Rivera, W. Mallouli, and E. M. de Oca. A security monitoring system for internet of things. *Internet of Things*, 7:100080, 2019.

[8] S.-K. Choi, C.-H. Yang, and J. Kwak. System hardening and security monitoring for iot devices to mitigate iot security vulnerabilities and threats. *KSII Transactions on Internet and Information Systems (TIIS)*, 12(2):906–918, 2018.

[9] F.-J. Ferrández-Pastor, H. Mora, A. Jimeno-Morenilla, and B. Volckaert. Deployment of iot edge and fog computing technologies to develop smart building services. *Sustainability*, 10(11):3832, 2018.

[10] J. Jung, J. Cho, and B. Lee. A secure platform for iot devices based on arm platform security architecture. In *Proc. of the 2020 14th International Conference on Ubiquitous Information Management and Communication (IMCOM'20), Taichung, Taiwan*, pages 1–4. IEEE, February 2020.

[11] D.-W. Kim, J.-Y. Choi, and K.-H. Han. Medical device safety management using cybersecurity risk analysis. *IEEE Access*, 8:115370–115382, June 2020.

[12] K. Košt'ál, P. Helebrandt, M. Belluš, M. Ries, and I. Kotuliak. Management and monitoring of iot devices using blockchain. *Sensors*, 19(4):856, 2019.

[13] B. K. Mohanta, D. Jena, U. Satapathy, and S. Patnaik. Survey on iot security: Challenges and solution using machine learning, artificial intelligence and blockchain technology. *Internet of Things*, 11:100227, 2020.

[14] P. M. Santos, J. G. Rodrigues, S. B. Cruz, T. Lourenço, P. M. d'Orey, Y. Luis, C. Rocha, S. Sousa, S. Crisóstomo, C. Queirós, et al. Portolivinglab: An iot-based sensing platform for smart cities. *IEEE Internet of Things Journal*, 5(2):523–532, January 2018.

[15] Q. Wang, P. Datta, W. Yang, S. Liu, A. Bates, and C. A. Gunter. Charting the attack surface of trigger-action iot platforms. In *Proc. of the 2019 ACM SIGSAC conference on computer and communications security (CCS'19), London, UK*, pages 1439–1453. ACM, November 2019.