A Component Vulnerability Matching Approach for IoT Firmware

Bo Yu, Yongyi Zhang, Runhao Liu†, Zhoushi Sheng
College of Computer
National University of Defense Technology
Changsha Hunan China
{yubo0615, zhangyongyi, runhaoliu, shengzhoushi12}@nudt.edu.cn

ABSTRACT

Component vulnerability matching offers an approach for discovering vulnerabilities existing in IoT firmware. In this work, A component composition analysis and reliability assessment (C2ARA) is developed to improve the component vulnerability matching. The C2ARA method employs a knowledge graph for discovering the components and their relationships from the extracted file system of the firmware. The key to the proposed method is to discover vulnerabilities from the component composition extracted from IoT firmware file systems, rather than only the information provided by CVE databases and firmware vendor. The results of the experiment with a large-scale dataset demonstrate the effectiveness of the C2ARA method.

CCS CONCEPTS

• Security and privacy • Software and application security

KEYWORDS

Vulnerability matching, IoT firmware, Reliability assessment, Component composition

ACM Reference format:

Bo Yu, Yongyi Zhang, Runhao Liu, Zhoushi Sheng. 2022. A Component Vulnerability Matching Approach for IoT Firmware. In *Proceedings of 6th Asia-Pacific Workshop on Networking (APNet 2022). ACM, New York, Fuzhou, China,* 2 pages. https://doi.org/10.1145/3542637.354264

1 Introduction

Nowadays, billions of IoT (Internet-of-Thing) devices are connected to the Internet. Generally, firmware is the software part of IoT devices. From the point of software supply chain, the components of a firmware include system kernel, application, system library and so on. Therefore, the vulnerabilities of IoT firmware exist not only in network applications [1], but also in its system kernel, library components and so on. Component vulnerability-related events are frequently exposed, such as Treck TCP/IP, OpenSSL and log4j vulnerabilities. From the perspective of firmware component composition analysis, a component vulnerability matching approach is need to discovery and repair vulnerability of IoT firmware in depth. However, the concealment of the relationship among the system kernel, application component and library component of the firmware makes it difficult to discovery all vulnerabilities [2]. The existing technology [3,4] of vulnerability discovery is mainly based on the information from CVE databases and firmware vendors. There is no existing method to discovery component vulnerabilities, which cause many vulnerabilities are missing during repairing process.

To solve the problem, this paper proposes a component composition analysis and reliability assessment (C2ARA) approach for IoT firmware. In this method, we design the component composition analysis and component vulnerability matching process by establishing a knowledge graph model among firmware, components and vulnerabilities. Then we provide a reliability evaluation module to assess the severity of the vulnerability. The key of C2ARA is to mine the component relationship among firmware components, and match the vulnerabilities based on the reference relationship of components.

APNet '22, July 1–2, 2022, Fuzhou, China

ACM ISBN 978-1-4503-9748-3/22/07.
https://doi.org/10.1145/3542637.354264

2 Design and Implementation

C2ARA method consists of several parts, including a knowledge graph model, vulnerabilities matching and reliability assessment module. The knowledge graph contains the relationship among target, component, and vulnerability. Vulnerabilities matching and reliability assessment module discover the component relationship and evaluate the reliability of the vulnerabilities.

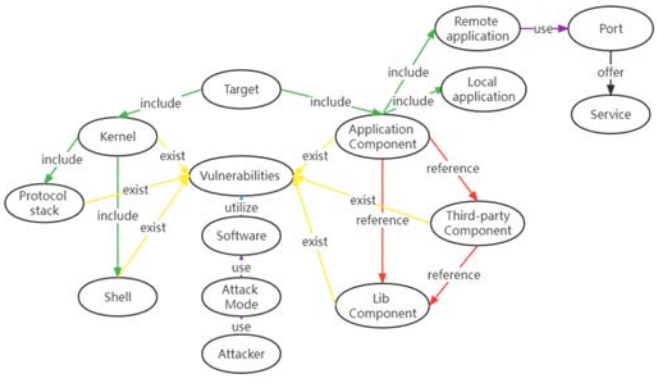

Figure 1: Knowledge Graph Model of C2ARA method

2.1 Knowledge Graph

Figure 1 shows the knowledge graph model. A firmware includes a kernel and several application components. Vulnerabilities exist not only application component, but also in the system kernel, system shell and system library. The application components reference third-party library components and system library components. Third-party library components reference system library components. Also, vulnerability exists not only in third-party library, but also in system library components. Additionally, application components can be divided into two categories, including remote service applications and local applications. An attacker can invade the firmware by exploiting vulnerabilities of remote service applications. By using the relationship among these components, the model helps to discover the existing vulnerabilities of all components in IoT firmware.

2.2 Matching and Reliability Evaluation

Component analysis mainly identifies components in the firmware and their reference relationship. Firstly, this approach identifies application components, third-party library components and system library components separately. TrID tool is integrated to distinguish application components and system library components. For discovering the relationship among components, this part then uses Yara and objdump tool to identify third-party library components and system library components referenced by the application components respectively, and then extracts the reference relationship between the application components and library components. Vulnerabilities Matching for IoT firmware mainly uses results of component analysis to match the corresponding vulnerabilities. Firstly, this approach processes the firmware analysis results to obtain the name and version of each component. Then, this approach uses the information to match the CVE database.

A reliability evaluation module is designed to evaluate the reliability of vulnerabilities and express the component relationship with a tree structure. Theoretically, the deeper the node where the component is located and the more the number of the next-level components is, the worse the vulnerability reliability of the component is. Therefore, considering a component with depth h and the number of next-level components n(n>0), the reliability value of a vulnerability can be expressed as $K = \frac{1}{h} * \frac{1}{n}$. Specially, if a component with depth h has no next-level component, the reliability value of its vulnerability can be expressed as $K = \frac{1}{h} * 2$.

3 Evaluation

We run our system on an Ubuntu server two CPU cores and 15GB memory. The dataset in our experiment consists of 100 vendors, including several types such as routers, cameras, wireless bridge, wireless controller, wireless access point and so on. The evaluation results are as follows.

Result 1. We count all components and vulnerabilities distributions for 100 vendors of firmware. Busybox, usbmount and libgcc_s.so.1 appear most in applications, third-party library and system library components of the 100 firmwares, respectively. Table 1 describes the top 5 vulnerabilities according the number of the three kinds of components. CVE-2018-20679, CVE-2019-5747 and CVE-2014-3620 appear most in the 100 firmwares.

Table 1: Distribution of vulnerabilities in firmware components

	Application	Third-part library	System library
1	CVE-2018-20679	CVE-2019-5747	CVE-2014-3620
2	CVE-2019-5747	CVE-2018-20679	CVE-2016-5420
3	CVE-2017-16544	CVE-2015-9261	CVE-2013-1944
4	CVE-2015-9261	CVE-2017-16544	CVE-2015-3153
5	CVE-2016-2147	CVE-2016-2147	CVE-2014-3613

Result 2. We calculate the reliability for 100 vendors of firmware. Table 2 describes the vulnerabilities with the top 5 reliability scores of application components, third-party components and library components in firmware ASUS 380.70_HGG-FINAL, in which the depth of library components is 2. CVE-2015-3153, CVE-2015-0290 and CVE-2014-6272 are vulnerabilities with highest reliability score.

Table 2: Distribution of vulnerabilities in components

	Application	Score	Third-part library	Score	System library	Score
1	CVE-2015-3153	0.111	CVE-2015-0290	0.031	CVE-2014-6272	0.167
2	CVE-2013-1944	0.111	CVE-2018-0734	0.031	CVE-2016-10196	0.167
3	CVE-2016-5420	0.111	CVE-2017-13080	0.031	CVE-2016-10195	0.167
4	CVE-2014-3613	0.111	CVE-2017-8798	0.031	CVE-2015-7497	0.083
5	CVE-2016-7141	0.111	CVE-2017-1000494	0.031	CVE-2013-2877	0.083

4 CONCLUSION

In this paper, we present C2ARA method to discover vulnerabilities of IoT firmware from application components and library components. The C2ARA method employs a knowledge graph to describe the relationship of different components in IoT firmware. Based on an evaluation algorithm, we evaluate the reliability of the vulnerabilities. We conduct an experiment with a large-scale dataset on 100 firmware of different brands. Experimental results indicate the effectiveness of the C2ARA method.

ACKNOWLEDGMENTS

The work was supported by the Natural Science Foundation of China (61902416, 61902412).

REFERENCES

[1] David Y, Partush N, and Yahav E. 2018. Firmup: Precise static detection of common vulnerabilities in firmware. ACM SIGPLAN Notices, 53(2), 392-404.
[2] Hou J, Li T, and Chang C. 2017. Research for vulnerability detection of embedded system firmware. Procedia Computer Science, 107, 814-818.
[3] Yao Y, Zhou W, Jia Y, et al. Identifying privilege separation vulnerabilities in IoT firmware with symbolic execution[C]//European Symposium on Research in Computer Security. Springer, Cham, 2019: 638-657.
[4] He D, Yu X, Li T, et al. Firmware Vulnerabilities Homology Detection Based on Clonal Selection Algorithm for IoT Devices[J]. IEEE Internet of Things Journal, 2022.