International Congress of Information and Communication Technology (ICICT 2017)

Research for Vulnerability Detection of Embedded System Firmware

Jin-bing Hou[a*], Tong Li[a], Cheng Chang[a]

[a]*Academy of Armored Force Engineering, NO.21 DuJiakan, Fengtai District, Beijing 100072, P.R.China*
* Corresponding author：painthought@163.com Tel.: 13121511377

Abstract

Because the embedded devices are widely used in the life, the industry, the national defense, the security of embedded system becomes especially important. The embedded systems firmware with the characteristics of strong concealment, difficulties of detecting, long maintenance cycle, the big hazard, has big threat to society security, industry security, national security. The paper summarizes the characteristics, current situation of vulnerability of embedded system firmware, and introduces the vulnerability detection briefly.

Keywords: embedded system; firmware; vulnerability; detection

1. Introduction

With the development of hardware devices and communication technology, embedded devices have already get into various realm, such as life, industry, national defense, medical treatment etc. Mobile phone, vehicle-mounted system, router, smart home, UVA, PLC etc are familiar embedded devices. The expansion of Internet of Things makes the application of embedded devices more and more extensive, and more equipment connect into network. So the security problems of embedded system get more and more attention. In the numerous security problem of embedded system, the problems that firmware brings are very obvious.

Firmware is stored in the permanent storage device of binary process. Due to limitations of technology, in the original the firmware is stored in ROM/FLASH which has the function permanent storage devices. Later with development of integrated circuit manufacturing process, now the firmware can also be stored in the other which has the function of permanent devices, such as SOC/FPGA chips and other micro controllers that have internal storage. In all sorts of microprocessor as the core of electronic system, upper software through firmware can effectively use various call interface provided by the hardware[1]. The firmware is the necessary constitute part in the electronic

system. Due to the great significance of firmware in the field of security, more attention paid to the studies of firmware security work. On the one hand, for improving competitiveness of own product in the aspect of security, firmware manufacturer dedicated to the firmware defects analysis. On the other hand, some security researchers analyze the firmware code, for the purpose to test whether the firmware is embedded malicious code, ensuring the security of the device firmware. Part of this analysis the firmware code, known as hacker, then they found vulnerabilities that exist in the firmware through the analysis and use of these holes enrich their attack means, to achieve the hidden malicious purpose. The current firmware security studies although has made many achievements in many ways, but there are still some inevitable limitations[2]: First, the security of firmware source to a great extent, determines the security of the firmware itself. Second, because the firmware code targeted, its security analysis technology lack of universality. Third, firmware analysis exist some problems, like low coverage of code, high false alarm rate of vulnerability.

Embedded firmware that stored in ROM storage is the firmware applied to embedded devices. The security hidden dangers of firmware also exist in embedded system firmware. Due to large-scale promotion of embedded devices and the restriction of hardware, embedded system with high authority and long vulnerability repaired period, the security situation of embedded system firmware became more rigorous. The underlying hardware is the core element of embedded system, and one of the most basic security layers. Firmware The firmware with very high authority is a key part of the upper application and the underlying hardware connecting. Only ensure the security of embedded firmware code, then can provide the most basic guarantee of security and solve the security problem of embedded system fundamentally. If attacker designed vulnerability or backdoor in the embedded system code, they can get particular operation authority easily. Furthermore, if vulnerabilities in the embedded system firmware code are not cleared and controlled in a timely manner, an attacker may take action to trigger these threats if necessary, causing fatalities behavior such as denial of service or self-destruction of embedded devices, resulting in paralysis of all equipment and serious economic losses and security disasters. The embedded system firmware vulnerability detection, can early detect vulnerability early, timely remove and control vulnerability in order to do a good job ahead of response measures to minimize the impact of security issues. The research of embedded firmware vulnerability detection technology can find out the more effective and comprehensive method to detect the vulnerability of embedded firmware. The more vulnerability is found, the sooner the embedded device security can get more protection.

2. Firmware vulnerability status

The CIH virus that broke out in 1998 was an early exploit of firmware vulnerabilities, and the CIH virus rewrote the BIOS firmware online, causing the BIOS firmware program to crash and fail to load the system. To ensure that the security of the device firmware has become an important issue to protect the security of all types of systems, and attracts more and more people to invest in its research. At this stage of the firmware code research results are very worth learning and learning.

In order to prevent unauthorized access to resources, ARM microprocessor with different permissions differentiate the operating mode. Embedded Linux kernel program works in ARM's USR mode, and the user program works in SVC mode. The kernel module can operate the hardware register. Yang Shide[3] designed the use of watchdog can modify the kernel module, dynamic loading into the embedded Linux system, through the user mode call to trigger.

The D-Link router, due to its peer-to-peer firmware library, discovered that the D-Link DIR-645 router detected a "cgibin"[4] vulnerability with its family of routers. The vulnerability could allow an unauthenticated attacker to get a remote control of the router by passing an extra cookie to the program stack overflow. Some Cisco routers have an untested test interface on the TCP32764 port, allowing an attacker to access the device's LAN-side interface. The vulnerability could result in an unauthenticated remote attacker gaining access to the root level of the device and executing arbitrary commands. Cui A[5], who use the HP printer firmware on-line remote upgrade feature, find use vulnerabilities, and completely control of an HP printer.

In 2007, Israel launched the radar by deactivating the built-in Trojan switch in the Syrian radar chip. The Israeli Air Force to avoid the Syrian air defense command system, and successfully attacked a Syrian nuclear reactor. The

2010 Stuxnet[6] virus attacked the Siemens S7 300 and S7 400 series industrial control systems at the Bushehr nuclear plant in Iran, causing serious equipment problems and delaying Tehran's nuclear program for two years.

3. Status of Firmware Vulnerability Detection

3.1 Fuzzy Testing

Fuzzy[7] test is a defect-based injection testing technology that uses a large number of semi-valid data as input to the application. The program appears as an anomaly as a symbol to identify possible application vulnerabilities. The so-called semi-effective data refers to the application program. The necessary identification part of the test case and most of the data is valid so that the program under test considers it to be a valid data, while the rest of the data is invalid. This error can cause the application to crash or trigger a corresponding security vulnerability.

Because most embedded devices, especially industrial systems, are almost undefended, for this problem, Shang Wenli[8] has established a fuzzy detection-based industrial embedded device vulnerability detection model, including the industrial protocol state model, unknown vulnerabilities mining, known vulnerability scanning, vulnerability identification model, security assessment and analysis and monitoring and control of six parts. She used PLC as an example to analyze the loopholes in embedded devices. The petrochemical liquid level control system was used to describe the attack process of industrial virus exploit and the detection and analysis of this process was demonstrated. A new method is proposed in the theory of vulnerability detection.

DAI Zhonghua[9] put forward a fuzzy analysis method based on fuzzy analysis of embedded device firmware. This method proposed the corresponding security rules according to the attack, and then put forward the concept of the risk weight from the analysis result, and designed the set of fuzzy test cases associated with the weight. He used this method to excavate the firmware vulnerability of embedded devices, successfully found the firmware of multiple embedded devices vulnerabilities, and mining vulnerability is significantly higher than the traditional fuzzy test method. The vulnerability mining method utilizes the stain analysis technique to improve the coverage and hit rate of the test case set, and performs well on the actual embedded device firmware.

3.2 Behavioral Analysis

Behavior analysis is based on the input of the behavior of the code and the results of the code analysis, and then determine whether there is malicious code behavior or vulnerability.

Zhang Cuiyan[10] proposed a method of formalizing the malicious code of the firmware code. This method is based on the fact that the firmware code has a strong correlation with the hardware, and describes the malicious behavior of the firmware code according to the users' wishes, and uses the multi-path method to analyze the malicious behavior of the firmware code. Compared with the traditional dynamic analysis method, this method has higher coverage and overcomes the shortcomings of the static analysis method for indirect jump instruction processing. However, the detection method is based on the user's willingness library, which needs abundant and complete user's willingness database to achieve the desired detection result. In the multi-path analysis process, the circular code has obvious influence on the coverage.

Hu Chaojian[11] on the embedded firmware of the back door were analyzed. Since the backdoor is semantically legal, its behavior is obviously malicious and difficult to detect. The malicious behavior of the backdoor is defined, the state is divided, the detection model is established, and the behavior of the firmware is analyzed to determine whether the malicious behavior is malicious or not. He took the HC 900 as an example of the back door scan, successfully found the back door. But the author did not lead to the collapse of the embedded system to detect the firmware vulnerability.

3.3 Homology Analysis

Depending on the production requirements, different brands of devices may run the same or similar firmware and contain the same third-party libraries, and the same brand of device may have vulnerabilities caused by several different third-party firmware libraries. Homology analysis is the analysis of the embedded device firmware and

known vulnerabilities of the third-party firmware library of homology, which found that the embedded device firmware vulnerabilities.

Costin A[12], who built a good scalability, high accuracy of the embedded firmware detection platform. The platform is based on powerful cloud computing to achieve a wide range of static firmware vulnerabilities on-line detection. The platform is based on the analysis of 32,000 firmware and 1700000 files, can be found in the embedded device firmware in the shared credentials, shared self-signed certificate and other documents stored in the remote login password. The platform is highly efficient and accurate, but the workload is huge and the scale of the project is huge.

Li Deng[13] pointed out that the same family of equipment generally have similar, homogeneous third-party firmware library. He can be based on third-party firmware library vulnerability has been detected in other embedded devices with the same family, the existence of the same vulnerability. The D-Link family of routers as example, Lee Deng to detect the same family of routers 63.15% contain D-Link DIR-645 router has been found loopholes. This method can find the vulnerability accurately and quickly, but the premise is that the homologous matching algorithm can match the third-party homologous library fast and accurately.

3.4 Symbol Execution

Symbols[14] in the implementation of the use of symbols as input to the program, by analyzing the current implementation of the instruction to simulate the implementation of the semantics of the instruction. Symbolic expressions are used during symbol execution to represent the values of variables, registers, and memory associated with the input symbol. Analyze the existence of the vulnerability based on the output. Symbol execution has two biggest features: 1, the code coverage is high; 2, does not produce false positives.

Drew[15] Davidson pointed out that the use of symbolic execution to detect embedded system firmware vulnerabilities, and based on the implementation of selected symbols to establish an embedded firmware vulnerability detection system called FIE. The system tested for good coverage and hit rate. The system detects 99 open source firmware project libraries using 13 different MSP30 modules and successfully detects 21 vulnerabilities in the firmware library. The system is only in the small, simple firmware vulnerability detection project running well, but in large projects Its performance is not very satisfactory.

4. Conclusions

With the development of integrated circuit technology and Internet technology, all aspects of life will be used in a variety of embedded devices. Embedded devices in the convenience of people's lives, at the same time, its security issues have become particularly prominent. Industrial production of industrial control equipment, defense equipment in the weapons system, automotive vehicle systems, smart home in everyday life and wearable equipment, are used in embedded devices. So large to national security, small to personal information, are closely related to the security of embedded devices. Embedded system firmware security is an important and difficult point in embedded system security.

In this paper, some firmware vulnerabilities were reviewed and summarized, some of the embedded system firmware vulnerability detection technology, and a brief analysis of these detection techniques. There are many problems in the development of embedded device firmware vulnerability detection technology, but there are still many problems, such as the coverage and hit rate of test cases in fuzzy test, the problem of homologous library matching in homology analysis, and the classification and detection of malicious behavior in behavior analysis. With the widespread use of embedded devices in the life and the increasingly important position in security, there will be a lot of technical problems to attract more researchers to study and explore.

References

1. Scott Borg. Securing the Supply Chain for Electronic Equipment: A Strategy and Framework, *2009 Internet Security Alliance report*, 2009
2. Lekang Yang. Research on Analysis for Firmware Code Malicious Behavior, Shandong Universuty,2013.

3. Shide Yang. Study on discovery technology against ARM‑based embedded system vulnerability, *Modern Electronics Technique*, Vol 38, No.18, pp. 57-59, Sep 2015

4. Shaohua Wu. Reveal Home Router Zero Day Vulnerabilities Mining Technology, *Electronic Industry Press*, 2015.

5. Cui A, Costello M. When Firmware Modifications Attack: A Case Study of Embedded Exploitation. *Network and Distributed System Security Symposium, California*, USA:NDSS Press, Vol 42, No.4, pp. 134-141, 2013.

6. Farwell J P, Rohozinski R. Stuxnet and the future of cyber war, *Survival*, 53(1), pp. 23-40, 2011.

7. Honghui Lee. Fuzz Testing Technology Research, *Chinese Science: Information Science*, Vol 44, No.10, pp. 1305-1322, 2014.

8. Wenli Shang. Study on the Vulnerability Analysi Method for Industrial Embedded Devices, *Automation Instrument*, Vol 36, No.10, pp.63-67, Oct, 2015.

9. Farwell J P, Rohozinski R. Stuxnet and the future of cyber war.*Survival*,2011,53(1):23-40.

10. Zhonghua Dai. A Fuzzing Test Method for Embedded Device Firmware Based on Taint Analysis, *JOURNAL OF SICHUAN UNIVERSITY*, Vol 48, No. 2, pp.125-131, Mar, 2016.

11. Chaojian Hu. Backdoor detection in embedded system firmware without file system, *Journal on Communications*, Vol 34, No. 8, pp. 140-145, Aug, 2013.

12. Costin A. Zaddach J, Francillon A. A large-scale analysis of the security of embedded firmwares, *USENIX Security Symposium. San Diego,USA:USENIX Association.* pp. 95-110, 2014.

13. Deng Lee. Firmware Vulnerability Detection in Embedded Device Based on Homology Analysis, *Computer Engineering*, Apr, 2016.

14. Pingping Lu. the Fuzzing Testing Technology Based on Hybrid Symbolic Execution, *Computer Application Research*, 2014.

15. Drew Davidson. Benjamin Moench, FIE on Firmware: Finding Vulnerabilities in Embedded Systems using Symbolic Execution, *USENIX Security Symposium. San Diego,USA:USENIX Association*. pp. 463-478, 2013