Security Detection and Research of Intelligent Hardware System

Ge Guojian[1, a] Xue Qingshu[1, b]

[1]Department of Computer Science and Engineering, Shanghai JiaoTong University, Shanghai, China

[a]geguojian123@163.com, [b] xue-qsh@cs.sjtu.edu.cn

Keywords: Intelligent hardware; simulation; dynamic analysis; static analysis; vulnerability mining

Abstract. With the rapid development of Internet of things technology, intelligent hardware equipment has become an important part of our lives. It widely exists in the smart router, network cameras, smart home and smart wearable devices. By analyzing the common features of intelligent hardware devices, our research object is the intelligent hardware devices based on embedded Linux system in this paper. Through the method of software simulation, we can reduce the hardware cost and make it possible for the intelligent hardware security analysis at scale. By combing static and dynamic analysis method, serious vulnerability is found. In the stage of static analysis, we collect the file information as many as possible, which can make dynamic analysis more effective and accurately. In the dynamic analysis stage, we mine system vulnerabilities of the firmware. Finally, testing the actual application of the 24 manufacturers of 1945 firmwares, we found 365 vulnerabilities in the 96 firmwares. The results of the test show the effectiveness of the proposed method.

Introduction

With the rapid development of the Internet of things, intelligent hardware equipment has become an important part of our lives, it is widely used in the router, network cameras, smart home and smart wearable devices.

However, attacks against smart devices continue to emerge in recent years, the security threat is gradually highlights, and is becoming a new topic in the field of network security.

The security of intelligent hardware is also very important. Home wireless router is the first line of defence between user equipments (computer, mobile phone and so on) and the Internet. if an attacker control the router, he can fully enter the user's network, and network traffic will be monitor. If a home network camera was attacked, the user's privacy information will be disclosure to the attacker.

Some achievements have been made in the research of intelligent hardware security. For example, Costin A get a lot of firmware in the Internet through the crawler, then automatic decompress, finally analyze whether there is a back door or private key leaks [1]. Zaddach J proposed Avatar system, which can be combined with the specific hardware implementation of the embedded firmware dynamic security analysis [2]. Basnight Z modify the firmware of the programmable logic controller to achieve the purpose of attack [3]. Daming D put forward the dynamic analysis technology of the Linux embedded system firmware by using the known vulnerabilities [4]. Costin A put forward the method of automatic analysis of embedded system firmware by using dynamic and static methods to analyze the possible security problems of web [5].

In summary, we make the following main contributions:

- By improving the support of NVRAM technology, the QEMU [6] can support much more firmware simulation then before.
- By using the static analysis process to collect information and the dynamic vulnerability mining, we can improve the analysis efficiency.
- The method is tested on 1945firmware applications. We have found 365 serious vulnerabilities, some vulnerabilities are discovered for the first time.

Framework design

Intelligent hardware system firmware is the software or data embedded in the hardware device, which provide the most intuitive, the lowest level of hardware control system. Early firmware was stored in ROM, which can not be modified later. With the development of technology, the firmware should be update and modify easily because of the constantly update the hardware environment. Modern equipment firmware is stored in erasable programmable read-only memory (EPROM) or flash.

In this paper, we propose a complete software simulation system, which do not rely on specific hardware equipment. The system is consisted of four parts: file system extraction, simulation initialization, static analysis and information collection, dynamic analysis and verification.

We extract the root file system from the firmware first. Then combining the precompiled Linux kernel and root file system, we use the QEMU to simulate it. At last, based on the static and dynamic analysis method ,we mining the security problems of the firmware. The framework of the system is shown in Fig. 1.

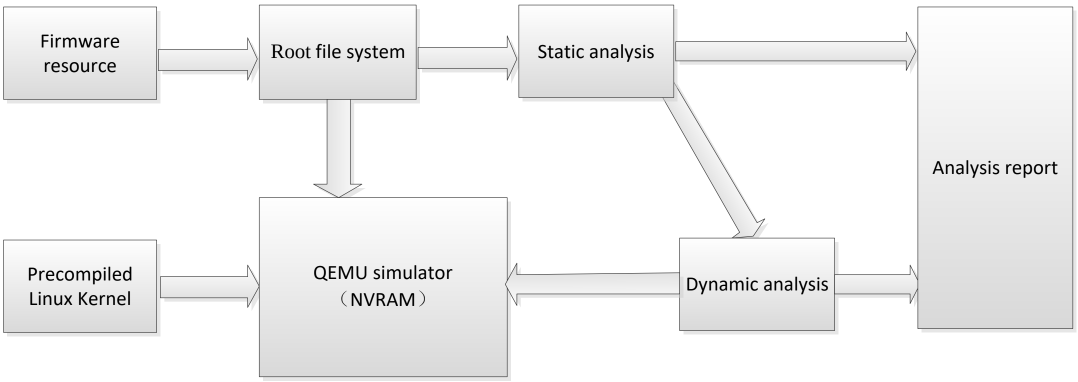

Fig.1 Framework of the system

In the system, we use the API function of binwalk to extract the key contents (such as: /bin, /sbin,etc.) from firmware. In order to reduce the error during the extration, we also use jefferson and sasquatch tool to extract the JFFS2 and SquashFS file system. We can extract most firmware which contain root file system.

By analysing the root file system, we find that the most used hardware archittecture: the 32 bits big-endian MIPS、 32 bits little-endian MIPS and 32 little-endian ARM. In the system, we only simulate this three kind architecture using the cross compilation environment compile Linux kernel.

In the process of static analysis, we collect the information of the root file system as much as possible. Some sensitive information collection (such as weak password in the root file system) is one of the most import part of the system. We also use some open source tool (such as RIPS) to mining the vulnerability of PHP.

The dynamic analysis is the key part of the system. We mining different kind vulnerabilities using different methods or tools. The main vulnerability includes authentication bypass、XSS、CSRF、sql injection and command execution. In this process, we use w3af、Arachni and IronWASP to automatic mining the vulnerability.

Experimental results and analysis

We have selected 1945 firmware from 24 different vendors in the open network resources, including router, cameras,etc.

We perform static analysis on all the firmware, whose root file system can be extracted firstly. We dynamic mining vulnerabilities in simulator by using the information, which is collected in the static analysis process then. At last, we give the detailed analysis of vulnerabilities in two products.

Using the previous extraction method, root file system can be extract in 852 firmware, accounting for 43.80%. We also find 365 serious vulnerabilities in 96 different firmware. The detail information is shown in Table 1.

Table 1 Vulnerability classification statistics

Vulnerabiliy type	issues	Firmware number
Authentication bypass	48	21
Command execution	37	16
XSS	189	47
CSRF	84	24
DNS rebinding	7	7
total	365	96（unique）

As part of the firmware is not the latest version, in which the vulnerability may has been remedied. We will give a detailed analysis of the causes of vulnerability in two products, while the vulnerabilies was submitted to the third security platform wooyun.

Phicomm K1 router is widely used. By using the system to analysis the latest firmware (version:V21.4.1.0), we find several 0day vulnerabilities. The first one is command execution vulnerability. We can run arbitrarily shell by visiting diagnose.asp in special format, which is caused by bad using network diagnosis function of ping and tracert. The second is system configuration file download vulnerability unauthorized. By using the static analysis of sitemap, we find that /cgi-bin/ directory can be access without logging on. When we access the ExportSettings.sh file, the simulator will return the contents of the config.dat,which contains all the router configuration information, including the adminsitrator account and password. The third vulnerability is CSRF. Arachni report router exists CSRF vulnerability. We find that when one administrator has logged on already, others can access file with administrator authority without logging on. The last problem is found manual. We find some sensitive information in the config.dat, one ftp address, ftp username and ftp password. We access the ftp server using the username and password. After decrypting the file in the server, we find the router is uploading our access domain name and ip to the server.

Ali Blink router is one smart home product. By using the system above, we find two kind velnerabilities. The first one is that we can reset the administrator password by accessing the protomimacol.csp file in certain format without logging on. The second is that we can access the machine in local area network by accessing the protocol.csp in certain format without logging on.

Summary

We use the QEMU to simulate the intelligent hardware firmaware in this paper, which can reduce the threshold and the cost of hardware effectively. By using the system, we can mining the vulnerability of intelligent hardware devices in large scale. Although this method can find some security issues, the whole process of analysis still need a lot of manual involvement. How to decrease the manual involvement will be our next work.

References

[1] Costin A, Zaddach J, Francillon A, and Balzarotti D. A large-scale analysis of the security of embedded firmwares //Proceedings of the 23rd USENIX Security Symposium. San Diego, USA ,2014:95–110

[2]Zaddach J, Bruno L, Francillon A, and Balzarotti D. Avatar: a framework to support dynamic security analysis of embedded systems’ firmwares//Proceedings of the 2014 Network and Distributed System Security Symposium. San Diego, USA, 2014: 23–26

[3]Basnight Z，Butts J， Lopez J， Dube T. Firmware modification attacks on programmable logic controllers. International Journal of Critical Infrastructure Protection, 2013,6(2):76 － 84

[4]Chen Daming D, Egele Manuel, Woo Maverick, and Brumley David. Towards automated dynamic analysis for Linux-based embedded firmware// Proceedings of the ISOC Network and Distributed System Security Symposium (NDSS). San Diego, USA, 2016

[5]Costin A, Zarras A, and Francillon A. Automated dynamic firmware analysis at scale: A case study on embedded web interfaces. 2015,arXiv, (arXiv:1511.03609)

[6]Bellard F. QEMU, a fast and portable dynamic translator// Proceedings of the USENIX 2005 Annual Technical Conference. Anaheim, USA, 2005: 41–46