RESEARCH ARTICLE OPEN ACCESS

Research Paper on Security in Embedded System

Prof. Sudhir Morey [1], Prof. Sangpal Sarkate [2]
Computer Department
Shri Sai Polytechnic, Chandrapur
India

ABSTRACT
The underlying hardware, hardware implementations of these software applications, embedded systems, and hardware devices were considered to be secure and out of reach of these attacks. However, during the previous few years, it has been demonstrated that novel attacks against the hardware and embedded systems can also be mounted. Not only viruses, but worms and Trojan horses have been developed for them, and they have also been demonstrated to be effective. Whereas a lot of research has already been done in the area of security of general purpose computers and software applications, hardware and embedded systems security is a relatively new and emerging area of research. This chapter provides details of various types of existing attacks against hardware devices and embedded systems, analyzes existing design methodologies for their vulnerability to new types of attacks, and along the way describes solutions and countermeasures against them for the design and development of secure systems.

Keywords :- OEMs, RSA, EMR, TEMPEST

I. INTRODUCTION

A few years ago almost all electronic equipment was built using analog components and devices. However, after the advent of microprocessors and microcontrollers majority of electronic equipment developed today uses digital components for design implementation. Embedded systems are finding their use in diverse applications ranging from complicated defense systems to home gadgets. Smart cards, debit and credit cards, DVD players, cell phones and PDAs are just a few examples of embedded systems that we use in our daily lives and fighter aircrafts, weapons systems, etc. are a few of many places where embedded systems are part of a bigger system.

With this increased usage of embedded systems in our daily lives, it is not unusual that bad guys and criminals try to take advantage of weak links in their security. Specially, the embedded systems used in financial institutions, battlefield equipment, fighter planes and industrial and nuclear plants may become targets of attack due to the importance of functions performed by them. Therefore, it is essential that these systems and the components used in them are highly dependable and their security is not compromised.

Currently, embedded system is becoming a main solution to most specific tasks because of its high stability, economic power consumption, portability and numerous useful. As a result, embedded system was used as a tiny computer to process many applications. Nowadays, many new applications are developed using web- based technologies, which users can access from anywhere through the Internet. However, data communicated between users and web-based applications may be revealed by anyone in the Internet, so secured communication is needed for web-based applications.

II. BACKGROUND

Embedded systems security is a new and emerging area of research. It is meeting point of many disciplines such as electronics, logic design, embedded systems, signal processing and cryptography. It is closely related to the area of information and software systems security because software is an integral component of any embedded system.

First microprocessor was developed around 1971 and later innovations in this field resulted in the development of computer systems and embedded devices. Software is an integral component of the both. In particular, every desktop computer carries a critical piece of software called the operating system. It manages the hardware resources and makes it possible for an end user to operate the computer. Other software applications in a computer run on top of the operating system.

It was the software component of digital systems which was first subjected to different types of security threats and attacks and many security incidents were reported against different operating systems and software applications. This started in 1970s and continues to date. However, embedded systems security gained importance in 1990s, specially, after side channel attacks were shown to be successful against smart cards. Later, emergence of networked embedded systems highlighted this area of research as the embedded devices could now be subjected to remote attacks.

Many of the methods and techniques used in the attacks against software applications can also be used against embedded devices, specially, in the firmware component. However, a few considerations involving the security of an embedded system are different from those of a general purpose digital system. To get a better perspective, it The security industry has focused largely on portable storage devices for the consumer electronics industry. The basic premise here is that users want security capabilities to travel with the device, such as with a USB thumb drive. This approach lets users protect their data on any system, whether it's on an office or home PC, an Internet kiosk, or a public computer. Software applications and data are password-protected using industry-defined security protocols, which often are targeted by Internet hackers. Portable data devices are also highly susceptible to theft. Once stolen and the security encryption defeated, the fully intact data can be accessed, loaded onto a PC or the Internet, sold, or worse data and prevent IP theft. Security requirements can vary for these applications. They can be as simple as ensuring that the correct storage product is in the host, or as intricate as tying the software IP and application data directly to the storage device.

Current Challenges in Maintaining the Security of Embedded Systems

Unlike standard PCs, embedded systems are designed to perform a designated set of tasks. These devices are typically designed to minimize the processing cycles and reduce the memory usage, as there are no extra processing resources available. Considering this, the security solutions developed for PCs will not solve the issues of would help to look at the traits of embedded systems security that are different from those of software security.

III. SECURITY ISSUES IN EMBEDDED SYSTEMS

Embedded systems traditionally have had very limited security options. Indeed, fitting a robust set of security features into such a small mechanical footprint can be challenging. Storage components, processing power, battery life, time-to-market, and overall cost concerns have prevented most security features from being implemented. Overcoming these design challenges has become crucial to embedded systems designers in light of the growing threat of security breaches as more systems are shared or attached to networks and new regulations are adopted that make security mandatory.

This imposes a number of challenges for embedded systems security, some of them are:

A. *Irregular security updates-*

Most of the embedded systems are not upgraded regularly for security updates. Once the embedded device is deployed, it keeps running on the software that it came with for years and even decades. If the device needs a remote software update, a capability needs to be designed into the device to allow security updates since the embedded operating system may not have automated capabilities to allow easy firmware updates that ensure embedded security.

B. *Attack replication-*

As embedded devices are mass-produced, the same version of devices have the same design and built as other devices in the lot. Considering this, there will be millions of identical embedded devices. If someone is able to successfully hack any of the devices from the lot, the attack can be easily replicated across the rest of the devices.

C. *Dependability*

Many critical aspects such as utility grids, transportation infrastructure, and communication systems are controlled by embedded systems. The modern society relies upon several facilities, many of them, in turn, rely on embedded devices. Cyber attacks would lead to an interruption in the functioning of embedded systems, which may have some catastrophic consequences.

D. Device life cycle

Embedded devices have a much longer lifespan as compared to PCs. One can easily spot embedded devices in the field that are a decade old, still running on the same system. So, when a manufacturer plans to develop an embedded system, they need to consider the potential threats that may arise in the next two decades. On top of developing a system that is secure against current threats, manufacturers need to match the security requirements of the future, which is a great challenge in itself.

E. Industrial protocols

Embedded systems follow some set of industrial protocols that are not protected or recognized by enterprise security tools. Enterprise intrusion detection system and firewalls can save the organizations from enterprise specific threats, but are not capable of providing security against industrial protocol attacks.

F. Remote Deployment

Numerous embedded devices are deployed in the field, outside the enterprise security perimeter. Therefore, these remote or mobile devices may be directly connected to the internet, without the security layers provided in the corporate environment.

All the above-mentioned challenges need to be addressed during the embedded device design and development, considering both hardware and firmware aspects. Only if the embedded device is secure, it will be able to run the intended tasks.

G. Wrapping Up

The question isn't if an embedded device is secure, the question is if an embedded device is secure enough. Different embedded devices require a different level of security, depending on the function it carries out.

The level of embedded security needs to be considered in the early phase of device design. Instead of relying on the enterprise security tools, embedded devices should come with a security system, so they can stand up against threats even outside the enterprise security perimeter.

IV. TYPES OF ATTACKS

Attacks on embedded systems can be broadly categorized as:

- Design and algorithmic attacks
- Side channel attacks

As the name suggests, a design and algorithmic attack exploits weakness inherent in the design and algorithm of the embedded system whereas a side channel attack attempts to exploit weakness in the implementation of the design. It is pertinent to point out that the bug may be left un-intentionally which is seldom the case or intentionally by the designer(s) involved at various stages of the implementation of the design level. Such a bug exists in the embedded system in the shape of tangible hardware and is commonly known as hardware In the case of side channel attacks, the attacker treats the embedded system as a black box and analyzes the inside of the system by feeding it various types of inputs and then observing the behavior of the system and its output. These types of attacks are normally used to extract some secret information from the embedded system.

A. Hardware trojan horses

Consider the case of standalone embedded systems i.e. an embedded system that is not part of any network. As the embedded device does not interact with any external network, it may be thought that no attacks can be mounted against the device. However, it is still possible for a malicious design engineer to leave a malignant hole i.e. a Trojan horse in the system. For example, a design engineer could program an embedded device to run correctly for all operations except, say, #2600th, or program it in such a way to behave erratically after certain number of operations or under a certain critical condition. If the device is part of a critical safety system in, say, an industrial process plant, the consequences may have a devastating effect on the plant operation. The result may be degraded performance, partial shutdown of the process or even complete failure of the entire plant.

B. *Side channel attacks*

Design and algorithmic attacks discussed above usually implant a Trojan horse in the system so that the system can perform a certain hidden action on a trigger. To be more elaborate, a hardware Trojan horse may, for example, be used to send all the records and data of the system to an unauthorized entity over a covert channel or it may be used to allow remote control of the system by an unauthorized entity. For these attacks to be effective, a certain malicious circuitry is usually part of the entire digital system, be it an embedded system based upon microcontrollers and microprocessors or be it an integrated circuit.

Side channel attacks, on the other hand, are usually used to extract some secret information stored inside a digital system. The digital system is treated as a black box and is subjected to various tests by applying different sets of stimuli to its input and noting the output behavior against every input. By comparing the output results against various inputs, an attacker tries to infer the design of the digital system and secret information stored inside it. In other words, side channel attacks exploit weakness of the implementation of the algorithm as compared to algorithmic attacks which exploit weakness in the algorithm itself.

There are four broad categories of side channel exploits:

- Time analysis
- Error analysis
- Power analysis
- Electromagnetic Radiation Analysis

A. *Time analysis*

In a side channel attack based on time analysis, the attacker tries to infer protected information by comparing time delays in processing of various forms of information.

For example, take the case of implementation of RSA (Rivest, Shamir, Adleman) public key encryption algorithm in a hardware security module. An attacker can encrypt a few thousand plain text samples and note the time it takes each time. With the analysis of this timing information, the attacker can infer the private key stored in the hardware module. Schmeh (2003) proclaims that in the case of a smart card, only a few hours are needed to extract the key.

To further elaborate the timing attack, consider the RSA decryption operations to compute a plaintext message *m* from cipher text *c* using private key (*d*,*n*) where *d* is the secret exponent and *n* is the modulus. The computation required to extract message *m* is given below:

$m = c^d \bmod n$

An attacker can get samples of cipher text *c* by passively eavesdropping on a target system and *n* can be inferred from the public key (*e*,*n*). By making timing measurements on multiple decrypted computations of the form given above, the perpetrator of the attack can then infer the secret exponent *d*, thereby enabling him or her to find the secret key (*d*,*n*). For timing analysis to work, the attacker must also know the encryption algorithm being used by the victim.

B. *Error Analysis*

Error analysis attack is also referred to as fault analysis attack. It was pioneered by Boneh, DeMillo and Lipton (1997) and later developed by Biham and Shamir (1997). In an error analysis side channel attack, the hardware module under attack is falsely activated or damaged and output pattern for a given input is obtained. For example, a smart card is damaged either mechanically or through heat. Output for the same input from a healthy module is also obtained. By comparing the correct and false results, the private key can be reconstructed.

Boneh et al. developed mathematical model of the fault analysis attack based upon transient faults

faults which occur only for a short duration of time and then are gone. For example, flipping of a single bit in a hardware module for a few micro seconds is an example of transient fault. The attacker can also induce the transient faults into a system. Effectiveness of the attack is dependent upon the implementation of the crypto system. For example, for an RSA implementation based upon Chinese remainder theorem, Boneh et al. showed that the modulus can be factored with a very high probability by using a single faulty RSA signature.

C. *Power analysis*

In this type of side channel attack, the attacker feeds different inputs to the embedded system and then observes the power consumed. The attacker then draws conclusions about the stored information by measuring and comparing fluctuations in power consumption. For example, DES key embedded in hardware security module can be inferred after about 100,000 encryption operations (Schmeh, 2003).

Like other side channel attacks, the power analysis attack can either be a simple power analysis attack or differential power analysis (DPA) attack. In the case of simple power analysis, the attacker draws conclusion about the operation being performed by observing the amount of power being consumed. For example, different instructions of a microprocessor take different amounts of time to execute and hence consume different amounts of power while they are being executed. Similarly, different stages of an encryption algorithm take different amount of time and power to execute. Some stages may be more computationally extensive and hence require more amount of power to execute and some other stages may require less amount of power to execute. As a result, by observing the power being consumed at a particular instant of time, the attacker can infer information about the stage of the encryption algorithm being executed and also the data upon which the operation is being performed. In other words, a simple power analysis can reveal the sequence of instructions being executed by the system under attack. Cryptosystems in which the path of execution depends upon the type of data being processed can be broken using this knowledge gained about the sequence of instructions being executed.

D. Electromagnetic radiation analysis

Even if an embedded system does not house a Trojan horse, or is not prone to timing, power or error analysis, it is still possible to breach its security by other means. In an attack based on electromagnetic radiation (EMR) analysis, the attacker captures and reconstructs the signal leaked through the electromagnetic radiation from the target equipment.

It is well known that the US government has been well aware of attacks based on analysis of electromagnetic radiation since 1950s and that display screen of video display units could be reconstructed after capturing their EMR. Standards were developed for the protection against this attack and were called TEMPEST which is an acronym for Telecommunications Electronics Material Protected from Emanating Spurious Transmissions. Partial details of TEMPEST are also available at the Internet. TEMPEST certification for private organizations is very expensive and therefore another standard called ZONE has been developed which is less secure but costs less than TEMPEST. There are three classes of TEMPEST standard: class 1 has the highest level of security and is available only to US government and approved contractors; class 2 is less secure and is again meant for use of US government; class 3 is available for general and commercial purposes.

EMR analysis attack is particularly dangerous against digital signals. If the digital signals of data being processed by a device can be reconstructed remotely, this can reveal the data. For example, hard disk stores information in binary form and when the data is read from or written to the hard disk, the digital signals are generated during these operations. If these signals are strong enough to be reconstructed remotely, the data being read from or written to the hard disk can be seen by the attacker.

V. FUTURE RESEARCH DIRECTIONS

As embedded systems security is an emerging area of research, the hardware security research community is expected to discover and develop new forms of attacks on the hardware devices and embedded systems. This will further fuel the research on the countermeasures and protection schemes against these attacks. As a result, there may be a paradigm shift in the design of hardware devices and embedded systems particularly those used in defense applications. For example, design and implementation of an IC may go through fundamental changes from an abstract level description to semiconductor level fabrication so as to incorporate new security measures. For this to happen, the research community needs to come up with reliable and robust techniques which can be implemented at every layer of abstraction.

Embedded systems security shares a few common traits with software security. Therefore, in these common areas existing security techniques and methods may be applied.

However, a few of the traits of embedded systems security are different from those of software security. Therefore, new algorithms and techniques will need to be developed for these areas. For example, research community will need to build secure operating systems which can be deployed in embedded systems which are usually resource constrained. This will include implementing security in all the critical functions performed by an operating system.

In many instances, attacks on embedded devices are possible only if physical security of the device is compromised. For example, in the case of side channel attacks on smart card, the attacker first needs to get a copy of the smart card which she intends to attack. Therefore, new security techniques will need to be developed which can prevent physical tampering of the device and ensure that the information stored in it remains secure.

Embedded devices are usually limiting in resources and most of the existing cryptographic algorithms are computation intensive. Implementation of security with the heavy algorithms results in performance degradation. Therefore, lightweight cryptographic algorithms and protocols are needed which are tailored to run on embedded devices with limited resources.

VI. CONCLUSION

Embedded systems are finding widespread uses in every sphere of our lives and their security has become an important research issue. In this chapter, we have discussed the background and current state of research on the threats and attacks being developed against embedded systems. The hardware attacks can be mounted at any of the layers of abstraction involved in the fabrication of the device with varying degrees of success. We have also discussed various countermeasures against these attacks.

REFERENCES

[1] Alkabani, Y., & Koushanfar, F. (2008, July). Extended abstract: Designer’s hardware trojan horse. In Proceedings of EEE International Workshop on Hardware-Oriented Security and Trust, HOST 2008, (pp. 82-83). Washington, DC: IEEE Computer Society.

[2] Biham, E., & Shamir, A. (1997). Differential fault analysis of secret key cryptosystems. In Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology, 1294, (pp. 513-525). London, UK: Springer- Verlag.

[3] Boneh, D., DeMillo, R. A., & Lipton, R. J. (1997). On the importance of checking cryptographic protocols for faults. In Proceedings of the 16th Annual International Conference on Theory and Application of Cryptographic Techniques, (pp. 37-51). Berlin, Germany: Springer-Verlag.

[4] Brier, E., Clavier, C., & Oliver, F. (2004). Correlation power analysis with a leakage model. In Proceedings of Cryptographic Hardware and Embedded Systems –LNCS 3156, (pp. 135-152). Springer-Verlag.

[5] Clark, J., Leblanc, S., & Knight, S. (in press). Compromise through USB-based hardware trojan horse device. International Journal of Future Generation Computer Systems, 27(5). Elsevier B.V.

[6] De Mulder, E., Buysschaert, P., Ors, S. B., Delmotte, P., Preneel, B., Vandenbosch, G., & Verbauwhede,

[7] Electromagnetic analysis attack on an FPGA implementation of an elliptic curve cryptosystem. In Proceedings of the IEEE International Conference on Computer as a Tool. EUROCON 2005, (pp. 1879-1882).

[8] Farrell, N. (2007, November). Seagate hard drives turn into spy machines. The Inquirer.

[9] Han, Y., Zou, X., Liu, Z., & Chen, Y. (2008). Efficient PA attacks on AES hardware implementations.

[10] International Journal of Communications, Network and System Sciences, 1, 1-103.

[11] IET. (2005). The celebrated maroochy water attack. Computing & Control Engineering Journal, 16(6), 24-25.

[12] Jin, Y., & Makris, Y. (2008). Hardware trojan detection using path delay fingerprint. In Proceedings of IEEE International Workshop on Hardware-Oriented Security and Trust, HOST 2008, (pp. 51- 57). Washington, DC: IEEE Computer Society.