SECURITY IN EMBEDDED SYSTEMS

Jesús Lizarraga, Roberto Uribeetxeberria, Urko Zurutuza, Miguel Fernández
Computer Science Department
Mondragon University, Spain
{jlizarraga,ruribeetxeberria,uzurutuza,mfernandez}@eps.mondragon.edu

ABSTRACT

Security has traditionally been a subject of intensive research in the area of computing and networking. However, security of embedded systems is often ignored during the design and development period of the product, thus leaving many devices vulnerable to attacks. The growing number of embedded systems today (mobile phones, pay-tv devices, household appliances, home automation products, industrial monitoring, control systems, etc.) is subjected to an increasing number of threats as the hacker community is starting to pay attention to these systems. On the other hand, the implementation of security measures is not easy due to the constraints on resources of this kind of devices.

KEYWORDS

Embedded systems, security, hacker attacks, security evaluation.

1. INTRODUCTION

An embedded system is a special-purpose computer system, which is completely encapsulated in the device it controls. An embedded system has specific requirements and performs pre-defined tasks, unlike a general-purpose personal computer. Examples of embedded systems are: mobile phones, network equipment, control devices for automobiles, household appliances, monitoring and control systems for industrial automation.

The security of this type of systems is a pending subject and this can soon become a problem, even bigger than the lack of security of current desktop computers. One of the reasons for this lack of security is the constraints of the hardware devices when implementing security measures. Another reason is the cost of security; manufacturers try to reduce production costs to obtain a market advantage for price sensitive products.

However, attacks and exploits on embedded systems are starting to get the attention of the hacker community. There are more and more exploits against PDAs and "mobile phone hacking tools" available on the net. There is also a reported mobile phone attack that would create a "Denial of Service" on 911 emergency service [1]. Today, due to the advances in technology, lower cost of products and easier access to the information on the net, attacks on embedded systems are becoming increasingly common.

2. ATTACKERS AND LEVEL OF PROTECTION

IBM proposes the following classification of possible attackers based on their expected abilities and attack strengths [2]: Class I (clever outsiders), class II (knowledgeable insiders), class III (funded organization).

There is an emerging group below the "class I" formed by those individuals just intelligent enough to read a document that describes an attack and follow step by step the instructions. This is a low skilled group but in the long-term can become a very numerous and dangerous group. "Class III" attackers belong usually to Government Agencies, organized crime groups or big companies.

Some embedded systems may incorporate security measures that will make more difficult to accomplish an attack. To classify the level of protection of a product Weinhart et al. [3] specified six levels of difficulty to succeed in attacks: *None, Intent, Common tools, Unusual tools, Special tools, In laboratory*.

It is obvious that the higher the level of security the bigger the cost of the product. For this reason the manufacturer should carry out a risk analysis to determine what the cost of a successful attack against his product will be and what class of attacker he must protect the product from. Once he knows the possible loss he must identify the candidate security measures for implementation and their cost.

3. ATTACKS ON EMBEDDED SYSTEMS

It is possible to classify the attacks based on their final goal, functional objective and on the method used to execute them. Our classification is based on Ravi's [4] taxonomy of attacks. At the top level, attacks are classified into four main categories based on the final goal of the attack [5]: Cloning, Theft-of-Service, Spoofing and Feature unlocking.

The second level of classification is the functional objective of the attack [4]. Here we would distinguish between attacks against privacy (the goal of these attacks will be gaining knowledge of sensitive information manipulated, stored or communicated by an embedded system); attacks against integrity (these attacks will try to change data or code within an embedded system); attacks against availability (a.k.a "Denial of Service" attack, these attacks disrupt the normal operation of the system).

The third level of classification is based on the method used to execute the attack. These methods are grouped into three categories [4]: Physical attacks, Side-channel attacks and software attacks.

4. COUNTERMEASURES TO AVOID ATTACKS

As stated previously, the security of embedded systems is often not considered during the design phase of a new product. Nevertheless, there are occasions where security is a concern during the development lifecycle of the product. In this case, developers must face important challenges due to the constraints in this kind of systems such as important limitation in processing, storage and battery life. There is an important research activity developing technologies for protecting embedded systems against some of the attacks described before.

To avoid physical tampering, there are mechanisms that offer: resistance, evidence, detection and response [5]. Weingart [6] describes various tamper mechanisms ranging from cheap and easy ones to extremely costly and complex ones. A mechanism against microprobing attacks involves the use of processors that encrypt all information sent on global buses [7], but this solution will cause important performance overheads. To prevent timing attacks there are techniques that suggest the addition of random delays that would increase the number of measurements required [8]. Buffer overflows are probably the most common type of vulnerability exploited by software attacks. To keep this type of problem from happening regulating the accesses of various software components to different portions of the system during different stages of execution is recommended [4]. It is also suggested to remove all unnecessary functionality and use compiler optimization to obfuscate easily identifiable code [5].

5. SECURITY EVALUATION

When a manufacturer wants to lunch a new product he will need to know how secure is his system and whether it meets the security objectives. To achieve this it is necessary to perform some kind of security evaluation. Security testing is about making sure that the countermeasures present in a device work correctly and all the security requirements are fulfill.

The most common approach to the evaluation process is to perform a suite of tests that represents known exploits. As new exploits and attacks arise it is necessary to maintain an exploits library up to date. It is important to mention that with this security testing method we will only be certain that the product meets a specific level of security today but we will not be sure about meeting that level in six months. For this reason it is important to complete security evaluations periodically.

There are numerous standards that address the requirements of security. Among them the most widely used is the Common Criteria ISO 15408 [9]. We start by establishing the security objectives for the product. Then we identify threats. There are two types of threats. Theoretical threats are usually found by universities and vendors. Active threats (a.k.a. "In the Wild threats") are exploits actually in use. Next, we identify countermeasures to protect against each threat and meet out security objective [1].

Unfortunately, current techniques are either too expensive, involve too much human subjectivity, or both.

6. CONCLUSION AND FURTHER WORK

There is a lack of security on present embedded systems. Security is not usually taken into account during the design phase of the product and it is difficult to implement once the product is completed. Even in those cases where security has been a concern from the beginning, the developer must face important hardware constraints to include security measures. Security should be integrated into the product during the conceptual design phase and should be taken into account for every part of the design.

There is an increasing number of security threats over embedded systems and various hacker attacks that jeopardise the commercial viability of new products or that can endanger the correct operation of existing ones. As 100% security does not exists an attacker having enough time, resources and motivation could always break into any system. For this reason, manufacturers must secure their products against specific threats trying to achieve a balance between the cost of security implementation and the benefits obtained.

However the implementation of security measures is not enough. It is also necessary to verify the effectiveness of these measures and even check for gaps or hidden threats. The product must be regularly monitored and updated in order to have the greatest effect possible against attacks. To accomplish this the definition of new security evaluation methodologies that take into consideration the changing nature of security and will assure that the product will remain secure is necessary.

REFERENCES

[1] Ricci, L. and McGinness, L., 2004. *Embedded System Security. White Paper*. Applied Data Systems, Columbia, USA.

[2] Abraham, D. et al, 1991. Transaction Security System. *IBM Systems Journal*, Vol 30, No 2, pp 206-229.

[3] Weinhart, S. et al. 1993. An Evaluation System for the Physical Security of Computing Systems. *Sixth Annual Computer Security Application Conference*.

[4] Ravi, S. et al. 2004. Tamper Resistance Mechanisms for Secure Embedded Systems. *In Proceedings of the International Conference of VLSI Design.* pp 605-611.

[5] Grand, J., 2004. Practical Secure Hardware Design for Embedded Systems. *In Proceedings of the 2004 Embedded Systems Conference.*

[6] Weingart, S. 2000. Physical Security Devices for Computer Subsystems: A Survey of Attacks and Defenses. *Workshop on Cryptografic Hardware and Embedded Systems.*

[7] Kuhn, M. 1997. *The TrustNo1 Cryptoprocessor Concept*. CS555 Report, Purdue University.

[8] Kommerling, O., Kuhn, M. 1999. Design principles for tamper resistant smartcards processors. In proceedings USENIX Workshop on Smartcard Technology.

[9] NIST. Common Criteria http://csrc.nist.gov/cc