Hardware Attacks Against Smart Cards

Tommy Persson Anton Nilsson
Email: {tompe015, antni790}@student.liu.se
Supervisor: Anna Vapen, {*x07annva@ida.liu.se*}
Project Report for Information Security Course
Linköpings universitet, Sweden

Abstract

In this report we discuss several hardware attacks against smart cards. We describe a number of successful attacks against cards such as pay-TV cards and pre-paid phone cards and how these attacks has since been mitigated.

We have also looked into a class of attacks called Optical Fault Induction Attacks and described how they work and what a designer can do to prevent them.

1. Introduction

Smart cards are becoming evermore popular as a means of authenticating and identifying users in a number of systems. Among these systems are the SIM-cards used in mobile phones, credit and debit cards as well as pay-tv cards. These systems obviously require a certain degree of security; it should not be easy to steal the keys contained in a persons SIM-card or credit card.

We begin by describing what we mean by hardware attacks in section two, providing an introduction to the most common tools used. Later in section three we describe a number of successful hardware attacks in the past. In section four we provide details about the methods used to mitigate these attacks and others.

Finally in section five we describe a single somewhat new class of hardware attacks in more detail and as well as a mitigation technique.

1.1 A Short Description Of Smart Cards

Before we go any further it would probably be prudent to describe exactly what a smart card is. A smart card is quite simply a small chip typically embedded in larger plastic cards, such as a credit card. There are two types of smart cards: memory cards and processor cards. The memory cards are only capable of storing fixed data, although they may contain small security circuits for preventing writes to read-only data. Processor cards on the other hand are full-fledged microprocessors capable of for example executing real-time cryptographic functions.

Since a number of years smart cards also come in contact-less form, which communicate wirelessly. Contact-less processor cards are typically limited to a range of a few centimeters while memory cards might function up to a meter. [5]

1.2 Problems

The questions we asked ourselves before writing this report were:

- What kinds of hardware attacks against smart cards are there?
- Which of these attacks are still possible?
- How can they be mitigated?

2. Background

Before diving into the hardware attacks, there are a number of terms and details relating to the writing of this report that we need to clarify. We do this in the following sections.

2.1 Terminology

By hardware attacks we refer to attacks that mainly concern the physical aspect of attacking smart cards. We are not interested in logical attacks where the protocols used in a smart card are attacked.

To help evaluation of tamper resistant devices IBM proposed the following classification of attackers [1], which we will refer to in our report.

- Class 1 – 'clever outsiders' – are clever individuals that may lack detailed knowledge of the system. They may also lack access to more advanced equipment and may often try to use an existing flaw in the system than to create a new one.
- Class 2 – 'knowledgeable insiders' – have more knowledge of the system and have experience working with it. Their knowledge of the entire system may vary

but potentially have access to most of it. They have access to more sophisticated equipment.

- Class 3 – 'funded organizations' – are organizations or groups able to hire teams of highly skilled personnel capable of in-depth analysis of the systems as well as designing attacks. They also have access to the most advanced tools. Class 2 attackers may be part of their attack teams.

2.2 Method

To be able to write this report we conducted a literary study of the subject. We have read a number of articles before piecing together the report you are enjoying.

3. Hardware Attacks

What kind of security was available on the earlier, the smart cards that preceded those that are sold today, cards? What kinds of attacks have been performed on them? What was used in these attacks? Which type of attacker (Class 1, 2 or 3 as described above) may perform the attacks? If for example, Class 1 is stated to be able to perform the attack below, it means that Class I attackers and up can perform it. These questions we will try to answer in this section of the report.

3.1 Pay-TV Cards

Early hacks, from the time when smart cards were fairly new on the market (1980s and early 1990s), against smartcards were often directed at the protocols used, some of these hacks could be considered as hardware attacks. Early hacks against the cards used for pay-TV were such attacks. The hacker would buy a pay-TV card, which at the time of purchase was "clean", meaning there were no restrictions on what channels the customer was allowed to see. When a customer buys a subscription to pay-TV the supplier often gives the customer more channels than the subscription covers during an introduction period. When this period ends the supplier sends a message to the customers' card, "locking" the channels the customer doesn't pay for. However if the customer, or hacker, was to place a device between the pay-TV card and the decoder that discards all messages to the card then the hacker can cancel his/her pay-TV subscription and the suppliers cannot "lock" the pay-TV card. This attack could be mounted by a Class 1 attacker since it does not require so much from the attacker [1] [2].

Once the producers of smartcards had filled the holes in the protocols, hackers turned to more hardware oriented attacks. Probing the cards for information turned out to be rather easy since methods of breaching the protection were somewhat well known. The protection often consisted of a layer of epoxy, which could be removed by small amounts of nitric acid, and a layer of glass which simply could be removed over the parts of the card the hacker wished to examine. The equipment needed for these probing attacks seems to be rather easy to obtain if the hacker has enough funds to buy them, perhaps buy them second hand from factories. The equipment needed was microscopes with micromanipulators attached. The micromanipulators were used to guide the probes to the processor's bus, which was the target of most attacks, on the chips surface. Some of these pieces of equipment even came with a built in laser which could be used to make holes in the glass protection layer making it even easier. The attacker would place his/her probes on the processor's bus and record the traffic; this would give a trace with both data and code of the operations performed. There used to be a standard where the checksum of the memory was computed right after each reset. This was very helpful to the hacker since it would give the hacker all of the content of the memory on the card if he/she could access the processor's bus. Later on, when the card producers had introduced a multiple key and algorithm defence (read more about this in the mitigation techniques section of this report). A new kind of attack was discovered, it involves breaking the instruction decoder on the card so that it cannot handle jump-instructions. This can be achieved by placing a ground probe on certain places on the card. Resulting in the repeated execution of the instruction that is on the connection where the next instruction awaits execution. This way the hacker can read all of the card's memory by listening to the bus, even the keys and algorithms that aren't in use at the moment. This kind of attacks could be performed by Class 1 attackers if they have access to the right equipment. The equipment is not too hard to come by but it may lean the attack type more to Class 2 attackers [1] [2].

When cryptographic processors were introduced on the cards to prevent the hackers they were still unsuccessful. This due to the fact that the cryptographic processor calculated the current key to decrypt the video stream and then passed it on to the cards CPU which was located on another part of the chip. Hackers were able to listen to the wiring between the CPU and the cryptographic processor and thereby see the decryption key. In newer cards the cryptographic processor is a part of the CPU, since it doesn't require more than a few thousand gates this is no problem. It is still possible for a hacker to manually analyse and reconstruct the circuit but it takes a lot of effort. The attacker would need to etch away layer after layer of the chip. After each layer the attacker would need to make electron micrographs after each layer and use image processing software to make a 3D map of the circuit. However, there is an easier way. There are companies specialised in reverse engineering, the hacker can simply hire them to reconstruct the chip, the companies does a lot

of this for legitimate companies who wish to find out if their competitors are using solutions they have the patent for. Class 2 or Class 3 attackers would be the ones that perform this kind of attack [1] [2].

3.2 Pre-paid Phone Cards

Early pre-paid phone cards had the same weakness as the early pay-TV cards and similar attacks could be mounted against these. Cards could be prevented from decrementing the tokens, that show how much time is left on the pre-paid card, by hand, resulting in unlimited call-time. These were often rather simple attacks which could easily be performed by a Class 1 attacker [1] [2].

3.3 General Attacks

Smartcards store their crypto keys and value counters in EEPROM, when changing these values the card uses an external power source. This external power source was received through a dedicated connection on the early cards, so to prevent cancellation of cards or decrementation of values on cards the hacker could simply put a piece of sticky tape over the dedicated connection and the values in the EEPROM could not be changed. This is one way of performing the first attack against pay-TV cards was mentioned in that section (3.2). It was a very simple attack which could be performed by a Class 1 attacker [1].

Another attack that the earlier cards were vulnerable to was an attack that slowed down the execution on the card, possibly so much that the execution was single-stepped. The slow down could be achieved by resting the card after each instruction (1st instruction, reset. 1st instruction, 2nd instruction, reset…). This type of attack could be mounted by Class 2 attackers, it requires more knowledge about the system than a Class 1 attacker would have [1].

Some smartcards have test circuitry that is used in production to see if the card works properly, when it is used the content of the cards memory can be read. The test circuitry is severed from the rest of the chip after this test but it can be put back in use by simply bridging the severed connections. This attack could be mounted by a Class 1 attacker [1].

Some cards have a protective mesh covering the card and sensors connected to it, explained further in the mitigation section below. This kind of defence is not easily broken but it can be done. With the help of a focused ion beam workstation the hacker can drill a hole in the protective mesh while the card is powered down. The hole is then filled with some kind of isolative material through which the hacker drills another hole. This hole is filled with a conductive metal so the hacker can use a probe on it and read out the information that travels on the line it is connected to. Focused ion beam workstations are not very easy to come by but many universities and companies have them and the hacker may be able to rent time on them from these. This attack would be mounted by Class 2 attackers since it requires rather sophisticated, not to mention expensive, equipment [1] [2] [3].

4. Mitigation Techniques

The defence against the early "Remove the programming voltage" attack against smartcards is rather simple. There is no specific connection for programming voltage; instead a voltage multiplier circuit is used. The defence isn't perfect, far from it, since the multiplier circuit could be destroyed by the hacker. Further defence would be to check to see if the value that was supposed to be changed actually was. In the case of pre-paid phone cards the number of tokens, after the instruction to decrement them has been executed, would be compared to the actual number of tokens left [1] [2].

One of the pay-TV card industry's first defences against hardware attacks against the actual smartcard was to store several keys and/or algorithms in the card's memory, so that when a key was found by hackers the pay-TV suppliers could simply send a command to the cards to start using the next key and/or algorithm [1].

Another important thing to keep in mind regarding smart card security is the logistics of it. Although many of the weaknesses we have described have since been fixed in newer cards, many old revisions are still circulating among users. One of the authors own SIM-card is approaching the ten-year mark and is undoubtedly susceptible to a lot of these attacks.

4.1 Tampering Detection

As a response to the "slow down" attacks discussed above the smartcard manufacturers installed a clock-frequency detector that would trigger a reset or perhaps a freeze of the card. However as with all detection systems this is vulnerable to false alarms, perhaps the clock fluctuates wildly at start-up causing a reset of the card. Many such defences aren't activated by the manufacturers; it is left to the people/companies that write software to the cards to decide if they are to be used [1] [2].

Other sensors that have been used on smartcards are: light sensors, temperature sensors and power supply sensors. As previously stated all such sensors come at a cost, they are somewhat unreliable and cards using sensors need to be configured to be rather insensitive which defeats its own purpose or they would be affected by false alarms [1] [2].

Another detection defence technique is to use a protective mesh to cover the surface of the card. The mesh is connected to sensors as well as ground and power. Any

attempt to breach the mesh would bread the circuit and the sensors would detect it which would lead to protective measures. Such as erasing the memory of the card, freezing the card or even setting of an explosive charge [1].

5. Optical Fault Induction Attacks

A somewhat new class of attacks we have not yet mentioned are optical fault induction attacks. Our previously mentioned attacks can be divided into two classes; invasive and non-invasive. While attacks like glitching or instruction injecting requires physical contact to the circuit, and often a large capital investment in equipment, optical fault induction which requires only that the circuit itself is exposed, is more accurately described as semi-invasive. [4]

Using devices such as X-rays, lasers and UV-light, transistors may be made to conduct at will, allowing the contents of individual bits in SRAM cells on the chips to be altered. No large investments are needed as evidenced by [4], where a simple photoflash lamp was used successfully. Also used was a small laser pointer together with an ordinary microscope. [4]

An example of an attack that can be carried out with this method is to introduce faults in calculation of an RSA signature. Often the $S = h(m)^d \pmod{pq}$ calculation in RSA is carried out first mod p and then mod q, because this is much faster. If the card calculates an incorrect S_p with a correct mod p calculation but a incorrect mod q calculation, the value p is given by $p = gcd(pq, S_p - h(m))$. Another possible, and more general, attack is to interfere with jump instructions. If an attacker can cause the code to follow conditional branches incorrectly he could for example reduce the number of rounds in a block cipher arbitrarily, making it easy to get the key. This attack could be mounted by a Class 2 or 3 attacker due to the extensive knowledge of the components in the card that is required. [4]

5.1 Mitigating Optical Fault Induction Attacks

Traditional mitigation techniques such as using metal shielding or encrypting the bus may make an attack harder but are not enough to completely stop an attacker as IR-light may penetrate the shielding and the bus may be avoided in the favor of individual registers.

The strategy suggested by Sergei in [4] is to use self-timed dual-rail logic. Self-timed, or asynchronous, circuits are of interest when a design grows complex enough that the cost of clocking is driven too high. These kinds of circuits instead signal when they are ready to receive data or done with their current computation.

In dual-rail logic no single line is used to transmit a 0 or 1, instead two wires are used. These wires may signal a 0 with 'LH', a 1 with 'HL' and 'LL' may be the “ready” state. Errors may pop up, resulting in an unwanted 'HH' state which will propagate and finally lock up the circuit. The strategy is to use this 'HH' state as a real error signal, that can be triggered by tamper sensors [4].

6. Conclusion

We refer back to our original problem statement: What attacks are there? Which of them are still possible? How may they be mitigated?

While we have discussed only a few attacks, we have described mitigation techniques for each of them. Those attacks that we have read about but have declined to mention have also all had some form of fix. We can conclude that none of the attacks should be feasible on a card produced today. More attacks are undoubtedly going to be discovered in the future and the possible mitigation techniques remain to be seen.

7. Similar work

Other articles reporting the state of smartcard security apart from our references include the following:

- Sanchez-Reillo, R., "Achieving security in Integrated Circuit Card applications: reality or desire?," *Security Technology, 2001 IEEE 35th International Carnahan Conference on* , vol., no., pp.197-201, Oct 2001
- Gary McGraw and Edward Felten, “How Secure Are Smart Cards?,” *Securing Java*, Chapter 8, Section 5, Jan 1999

8. References

1. Anderson R. 2001, Security Engineering: A guide to Building Dependable Distributed Systems, Wiley
2. Anderson R. & Kuhn M. 1996, Second USENIX Workshop on Electronic Commerce Proceedings, USENIX
3. Witteman M. 2002, Information Security Bulletin July, Information Security Bulletin
4. Skorobogatov S. & Anderson R. 2003, Optical Fault Induction Attacks, University of Cambridge
5. Wolfgang R. 1997, Smart card handbook, Wiley