Hardware Attacks
Tamper Resistance, Tamper Response and Tamper Evidence

Maurice Aarts

TU/e - Eindhoven University of Technology, Eindhoven, NL,
m.a.p.aarts@student.tue.nl,
WWW home page: http://maurice.aarts.info/

Abstract. This article gives a brief introduction into the technologies used for securing embedded devices such as smart cards, microchips and FPGAs against hardware based side channel attacks. The main focus of this article is on smart cards, as they are usually the target of such attacks and are often especially designed with such attacks in mind. However most, if not all, of the technologies and scenarios hold for any type of embedded device or -system. Many of the hardware attacks make use of the algorithms and software programs that are used on the embedded device, so there is also a short explanation of how the software introduces possibilities to attackers. The article first gives a set of attack categories, which is followed by an overview of countermeasures through tamper resistance, tamper response and tamper evidence.

Keywords: tamper resistance, tamper response, tamper evidence, responsiveness, embedded device, system, smart card, microchip, FPGA

1 Introduction

Embedded devices are often susceptible to two types of attacks, namely attacks on the physical hardware, and theoretical attacks on the software and algorithms used. Hardware attacks are used to either brute-force a known algorithm, or to perform side channel attacks on an embedded device. Both hardware based brute-forcing and (software-) attacks depending only on the algorithm used are not in the scope of this paper.

Side channel attacks have been deeply studied for years to ensure the tamper resistance of embedded implementations[1]. This has led to numerous improvements to the physical security features of embedded devices and the implemented algorithms. Smart cards and other similar embedded devices are typically considered to be tamper resistant, which means that the intended functionality and data held within such a device should not be undermined by tampering[2]. Such devices often embed cryptographic applications and their associated private keys on the embedded device itself, allowing the device to ensure the authenticity of the user and to enforce the confidentiality of the stored data[1]. The security of

practically any cryptographic application depends on the fact that the opponents have no access to some secret key data[3].

> No system will ever be 100% secure.
> "Secure" can simply be defined as when the time and money required to break the product is greater then the benefits to be derived from the effort. Given enough determination, time, and resources, an attacker can break any system. - Joe Grand [4]

The level of protection provided against well-equipped attackers is frequently overestimated; application designers must consider that skilled attackers can use various semiconductor testing equipment and other tools to extract large amounts of information from smart cards and other embedded devices, including cryptographic keys. Both invasive and non-invasive hardware attacks are possible and play an important role in the security of embedded devices[3]. This paper looks into the methods that can be used to attack a smart card, what can be done to make embedded devices more resistant to tampering, and increasing the tamper-evidence so that tampering may be detected.

The structure of the article is as follows. Section 2 gives a brief insight into what an attacker is willing to do with respect to the available resources. In section 3 we provide an overview of the most popular types of hardware attacks on embedded devices, namely sections 3.1 and 3.2 give a more in-depth view of invasive attacks and non-invasive attacks respectively. Section 4 covers how embedded devices can be made more resistant to tampering, and section 5 shows what can be done if tampering is detected by the device. Section 6 provides a number of additional ways of detecting tampering; and finally section 7 contains our conclusions.

2 Attackers

Today, due to advances in technology, lower cost of products and easier access by the public to once-specialized tools, attacks against hardware are becoming more prevalent. There are basically three classes of attackers, depending on their expected abilities and strengths. The classification[4] shown[1] in Table 1 is based on Abraham et al's *Transaction Security System*[5] and is an industry standard for describing attackers in an academic fashion. Additionally we will describe four main security threat classes, which are defined by J. Grand[4] as follows:

- Interception (or eavesdropping) - Gaining access to protected information without opening the product. A silent interceptor may leave no traces by which the interception can be readily detected.

[1] The table is self-explanatory and should be sufficient in this context. See J. Grand's *Protecting your crown jewels: an introduction to embedded security for hardware-based products* [4] for additional information on this topic.

Table 1. Comparison of each attacker class against available resources

Resource	**Class I**	**Class II**	**Class III**	**Class III**
Category	Script-kiddie	Academic	Organized Crime	Government
Time	Limited	Moderate	Large	Large
Budget	<$1000	$10k - $100k	>$100k	Unknown
Creativity	Varies	High	Varies	Varies
Detectability	High	High	Low	Low
Goal	Challenge/Prestige	Publicity	Money	Varies
Number	Many	Moderate	Few	Unknown
Organized?	No	No	Yes	Yes
Releases information?	Yes	Yes	Varies	No

- Interruption (or fault generation) - An asset of a product becomes unavailable, unusable, or removed. An example is malicious destruction of a hardware device, intentional erasure of program or data contents, or a Denial-of-Service network attack. Fault generation, which consists of intentionally provoking malfunctions, which may lead to the bypassing of certain security measures, also falls into this class.
- Modification style - Tampering with an asset of a product. Modification is typically an invasive technique for both hardware, such as circuit modifications or micro-probing, and software/firmware, such as changing the values of data or altering a program so that it performs a different computation.
- Fabrication style - Creating counterfeit assets in a product or system. Fabrication can come in many forms, including adding data into a device, inserting spurious transactions into a bus or interface, or a Man-in-the-Middle attack on a network. Sometimes these additions can be detected as forgeries, but if skillfully done, they may be indistinguishable from the real thing.

Attackers usually exploit a targeted system in order either to make a copy of (a part of) the technology, to bypass a service such as copy protection or a payment system, to spoof user authentication, or for privilege escalation and feature unlocking. These attacks fall into one of three categories. The attack can be a *focused attack*, in which the attacker takes the time to sit down and plan the attack without a high risk of being discovered while actually doing it. If there is a strict time constraint, the attack is referred to as a *Lunchtime attack* meaning that the attacker has anywhere from a few seconds to a few hours to do the attack. Such attacks are usually riskier than focused attacks. The third type of attack is an *insider attack*. Insider attacks are attacks that are done by someone that was in the development and supply chain of the device.

3 Attack overview

The type of attack that an attacker choses to do depends a lot on the goal of the attack and which class of attacker he is. The type of attack also depends on

exactly which kind of embedded device it is, and which security methods were implemented during the devices construction.

3.1 Invasive attacks

Invasive attacks are the simplest way to learn a large amount of information about an embedded device, however most invasive attacks require very expensive equipment while non-invasive attacks can often be done with tools that are available to an advanced hobby enthusiast[3]. Additionally invasive attacks tend to destroy the packaging and in some cases the entire device. This means that invasive attacks are only feasible in situations where destruction of the device doesn't really matter, or where the damage to the device can be reconstructed so that there is little to no evidence of the attack. Below we give a classification of the most common types of invasive attacks. Note that this list is incomplete and that an attack can fall in multiple categories.

Probe attacks - The purpose of a probe attack is to directly attach a conductor to the circuit being protected so that information can be obtained from and changes can be injected into the system under attack. Attack probes can be either passive or active and may not actually be a physical object. *Passive probes* are simple oscilloscope or logic analyzer leads that are attached to the embedded device and are set to record the information at that point of the circuit. Passive probes are often terminated in active circuitry, which gives them a very high input impedance which in turn may help to avoid detection or interference with the circuit being attacked[6].

Probe attacks[3] are also commonly used as the first step for more advanced attacks. Once an attacker has probes in place they can then attempt to do a number of different attacks such as timing attacks[7], cache-based attacks[8], power monitoring attacks[3, 7, 2, 9] such as simple power analysis (SPA) and differential power analysis(DPA), and differential fault analysis (DFA)[1, 2, 9] attacks.

Machining methods - Another invasive attack on a smart card or embedded device is to simply cut away parts of the chip, piece by piece until the attacker understands the construction of the device. Often integrated circuits are packaged in a cover or other tamper resistant coating thus ensuring that a probe attack cannot be done. By machining the chip and removing the cover and coatings, it becomes possible to reach the actual circuit and proceed using a probe attack. Machining can be done manually usually with the attacker using a knife of other tool to remove material from the device. Mechanical machining is the automated process of removing material from a chip. Even though mechanical machining is usually faster and more precise than manual machining, mechanical machining is often less accurate than manual machining as there is little to no feedback and often too much material is removed. Extremely precise machining can also be done using either (demineralized/deionized/pure-) water or a laser.

Water machining has the advantage of being extremely precise and is difficult to detect if pure water is used as it is non-conductive, however, water machining equipment is usually very large and generally only available to some class II en class III attackers. Laser machining has most of the same advantages as water machining, in that the laser is non-conductive and thus hard to detect. The equipment for laser machining is generally also smaller than for water machining, but a large disadvantage of laser machining is the heat that is generated by the laser. The last general type of machining is by using chemicals. Chemical machining is similar to water machining, except that instead of water corrosive chemicals are used to quickly and efficiently dissolve the material. The biggest disadvantage of chemical machining is that the chemical agents are often conductive and thus they are easier to detect and may even cause unintended short circuits[6].

Shaped charge technology - A shaped charge is an explosive charge shaped to focus the effect of the explosive's energy. Using tiny explosives, it is possible to penetrate an integrated circuit so quickly that circuits that detect intrusions can be disabled before they have a chance to respond. As the explosions cause the cuts to be done at hypersonic speeds of up to over 7 km/s there is almost no time for the circuit to signal its alarms. One disadvantage of this method is the fact that it is purely destructive and relatively inaccurate.

Glitching - Changing the inputs of a microchip in an unexpected way can cause the chip to *glitch*, which means that the chip starts doing erratic operations. Glitching can be caused by changing the input voltage (Vcc) thus causing instructions to be misinterpreted and circuitry to fail. Doing so at the correct moment can cause advantages to the attacker such as memory not getting cleared or instructions being garbled. A similar effect can be achieved by lengthening and shortening the clock pulses going to the IC. The timings in the chip desynchronize and erratic behavior results. A third way of introducing glitches is through electromagnetic interference, as such fields can cause disruptions in diodes and transistor circuits[3, 6]. Note that glitching can also be caused by environmental factors and thus it is not strictly an invasive attack.

Scanning electron microscopes - Class III attackers that have access to scanning electron microscopes can use their equipment to read and possibly write bits to ROMs or RAM on a molecular level. This technique requires that the chips' surface is exposed, but once exposed the scanning electron microscope can access and read almost any part of the chip to obtain and possibly modify the secrets stored there.

3.2 Non-invasive attacks

Non-invasive attacks are often more sophisticated in their design than invasive attacks, and their implementation often depends on tiny design vulnerabilities in

the embedded device. Non-invasive attacks require detailed knowledge of both the processor and software used, in contrast to an invasive attack where the attacker can simply probe the logic to see what does what. A large amount of work might be necessary to first design a non-invasive attack, but once such a technique has become available for a specific device and software version, it can often be reproduced reliably within seconds on another device of the same type[3].

Energy and Radiation attacks - Energy and radiation attacks can be used to 'lock' or 'freeze' certain parts of a circuit into a certain state. Energy and radiation attacks can be done both with (invasive) and without (non-invasive) actual contact and usually require close access to the device. One such attack, called *Radiation imprinting* is the process of radiating parts of the IC, such as the CMOS RAM, such that the values of the bits are 'burned' into the memory. This means that a normal clear or write operation will not change the value of the bits in that ROM. This allows an attacker to read the ROM at a later moment without having to worry about the data accidentally being lost. Similarly *Temperature imprinting* is a method that literally 'freezes' the bits in ROM so that they can be read minutes or even hours after power has been removed from the chip. An IR laser can be used to read and write to the cells of a ROM or RAM. Silicon is transparent to infrared frequencies, so it is possible to read or write a bit value by focusing an IR laser beam on a certain location on the chip without requiring it to be machined or otherwise invaded.[3, 6]

Imaging technologies - Almost any imaging technology available can be used to make images of a chip. Microscopes with recording devices, X-ray equipment, ultrasound, and other tomographic equipment. These devices can help an attacker visualize the internals of a chip without needing to physically open or tamper with the device.

Software attacks - Software attacks are attacks done by simply communicating with the embedded device over the normal channels, and attempting to learn more about the device by exploiting security vulnerabilities in the software[9].

Fault generation techniques - Fault generation techniques usually use external environmental factors to cause glitches and other malfunctions in the embedded device. This is basically a combination of both Glitching and Energy/Radiation style attacks and can be used in combination with either software-based attacks or probe-based attacks.

4 Tamper resistance

Tamper resistance relies on restricting physical access to the smart card or embedded device, such that the only interaction has to be done through the software

embedded on the device. Of all security methods, tamper resistant security is usually the easiest to apply, as tamper resistant systems usually take the so-called bank vault approach and ensconce the microchip in a protective cover that protects it against invasive attacks[6].

There are many different ways to restrict physical access to an embedded device. Below we have a list of such methods[2], each with a brief description of what the method details and the types of attacks it helps protect against.

"Bank vault technology" - By simply making the embedded device too big or heavy to steal can significantly decrease the probability of an attacker stealing the device. The device can also be permanently attacked to an object such that the embedded device is destroyed before it can be detached from the object. Note that this is not very convenient for portable devices and thus other technologies have been developed.

Hard Barriers - An actual hard physical barrier surrounding the device. Materials such as steel, ceramics, hard plastics and cement or brick can help prevent invasive tampering, and may also prevent theft in combination with the technology above. An example of a hard physical barrier is shown in Figure 1.

Fig. 1. Insecure and tamper resistant chips[9]

Metal Shielding - Enclosing the device in a metallic cage helps protect it against electromagnetic fields, and embedding layers of metal in the circuit board help obfuscate which traces in the board are causing the magnetic field[9].

[2] This list is based mainly on Weingart's work in *Physical Security Devices for Computer Subsystems: A survey of Attacks and Defenses* [6]

Insulator based substrates - Silicon becomes transparent to infrared radiation, so in order to prevent against IR laser attacks it is possible to replace the majority of the silicon in the device with a material that is not transparent to IR lasers or other frequencies that enable imaging of the circuits. Some examples of such materials are SiMOX (Silicon/Metal Oxide) and SOS (Silicon-on-Sapphire). Using an insulator based substrate in combination with advanced passivation gives the highest level of passive, single-chip, protection. Note that material machining techniques can still disable this type of security by removing or thinning the substrates to a thickness where the material is too thin to block IR light and allows imaging attacks to take place.

Semiconductor Topography Design - By designing the chip in a certain way, it is possible to ensure that the layers required for functionality surround the layers that need to be kept secret. This ensures that the secret areas cannot be exposed without removing or damaging the functional layers that are required to read the secret data. This technique can be used against pico-probing, scanning electron microscopes and the various machining techniques.

5 Tamper response

Whereas tamper resistant systems used a bank vault approach, tamper response systems are more like a burglar alarm. These systems specialize in detecting an intrusion, and if such a detection takes place the chip will instantly attempt to stop the attacker from learning anything else about the system. Such responses can vary from simply sounding an alarm, to clearing the ROMs, to destroying the physical device itself.

Tamper response technology consists of two important parts, the first is detection of an attack, and the second is the actual response if an attack is detected. Detection of an attack can be done by installing sensors on the embedded device. In Steve Weingart's paper *Physical Security Devices for Computer Subsystems: A survey of Attacks and Defenses*[6], he describes a complete list of sensors that can be used to detect a multitude of attacks. The exact shape and type of sensor depends on what it is built to detect, but regardless of the type of sensor it gives an output when an attack is detected. Such an output is caught by the logic that handles the response part of the tamper-response module. These mechanisms fall mainly into four groups:

- Switches - devices that detect mechanical movement.
- Sensors - devices that detect an environmental change.
- Circuitry - wires and/or fiber-optics that are wrapped around and throughout the embedded device. These materials are used to detect a break, puncture or attempted modification of the wrapper[4].
- Electronic - detection and monitoring of changes in frequencies, clock pulses or voltages leading in and out of the chip[7].

The circuitry that handles the output of the tamper-response sensors is usually used to ensure that an attacker cannot obtain the secret data on the device. Often an attack is detected before the attacker has finished obtaining all the necessary data from the device, and in such cases it is essential that the device attempts to keep the attacker from obtaining the rest of the data. In most embedded devices and smart cards, the secrets are stored in either RAM or ROM memory modules. While RAM is relatively easy to clear during an attack, ROM is significantly harder.

The simplest way to erase the secrets in RAM is to do a *RAM Power Drop*. This means that power to the RAM modules is removed which effectively clears the contents.

A slightly more difficult way to clear RAM (or ROM) is by doing a *RAM Overwrite* (or *ROM Overwrite* respectively). A RAM overwrite repeatedly overwrites the memory module with all zeros and all ones alternatively. This process ensures that there is no residual information left that could be caused by imprinting, but it requires power and time to do the actual overwriting. This method is most accepted by governmental standards, but its success cannot be guaranteed in attack scenarios as a reliable source of power is needed while it is overwriting the memory modules.

The third and most effective way of ensuring that an attacker does not obtain the secrets on the device is by completely destroying the device itself it an attack is detected. Physical destruction of the device can be done by shorting certain parts of the circuit and thus rendering the device inoperable. It can be done with little to no violence, and in some cases may not even be detectable until the attacker notices that the device ceased functioning.

6 Tamper evidence

Tamper evident systems are designed to ensure that if a break-in occurs that evidence of the break-in is left behind. These systems do not protect against the attack itself, but only prove that an attack occurred after the fact. Tamper evident systems often use chemical or mechanical means to show evidence that an attack has taken place. As tamper evident systems themselves do not activate an alarm or otherwise notify the owner that a break-in attempt has occurred, it is important for an effective audit policy to be established and adhered to that visually checks the device frequently to ensure that there is no evidence of an attack[6]. As such tamper evident systems are often combined with a tamper response system to alert the owner of an attack, and to prove that an attack indeed took place.

As with the tamper resistance techniques there are a large number of different possibilities to ensure that tampering becomes evident. Again we will enumerate a number of possible methods. This list is incomplete as new materials are developed daily that can be used as a tamper evident layer. The use of cutting-edge materials can also help ensure that an attacker cannot easily replicate the material and reconstruct the tamper-evidence layer.

Brittle Packages - The most trivial way of proving that a device has been tampered with is by sealing it in a brittle package. Once an attacker attempts to open or penetrate the enclosure the brittle package shatters and cannot be repaired. Such packages are difficult to reconstruct and thus the attacker leaves evidence of the attack.

Crazed Aluminum and Polished Packages - The package is made from aluminum or other similar material, which has been heated (usually above 1000 degrees F.) and quenched. This heat treating causes a myriad of shallow, web-like cracks to appear on the surface. These cracks, like a fingerprint, are unique to each piece. The case can be photographed and subsequently audited using the photograph and optical comparison devices[6]. A polished package is an aluminum package that has been polished such that there are no cracks or marks evident. If on inspection there are such markings, it is evident that the package has been tampered with.

Bleeding Paint - Paint of one color is mixed with micro-balloons containing paint of a contrasting color. If the painted surface is damaged by the attacker the other color bleeds onto the surface and is easy to identify as having been tampered with.

Holographic Tape - The surface of tape, with a very firm adhesive, is printed with a holographic image similar to the kind used on credit cards. This kind of tape is moderately difficult to forge, and it is constructed so that attempts to remove it will damage it (the tape may be scored to promote tearing when removal is attempted)[6].

7 Conclusions and Future Work

The security of embedded devices is a game of cat and mouse played on the cutting-edge of technology. The developers of embedded devices must stay ahead of the attackers in the race to keep their devices secure. Devices can never be 100% secure and so it is always a cost/benefit consideration to see if the time and money required to break the security is sufficient to keep attackers at bay. As we saw earlier, there is not a single countermeasure that works against all possible attacks. This means that an embedded device that has multiple layers of tamper resistance, tamper evidence and tamper response features will be better protected against attacks than a device that uses a single feature. Note however that a single vulnerability is enough to break the entire system. A smart card may be suggested to be 99% 'secure' against all kinds of hardware attacks, but if the software on the device contains a vulnerability, the security is still flawed.

As this technology is being developed at such a rapid pace this article can only give a guideline of possible attacks and possible countermeasures to such attacks. In the future new techniques will be developed that may or may not

fall within the categories covered in this paper. However, hopefully this article gave some insights to the security of embedded devices and the technological possibilities. While tamper responsiveness, -evidence and -resistance are not really new subjects, there are still numerous subjects that are open for research. Methods of detecting intrusions and perhaps non-destructive methods of keeping the attacker from learning anything from the system when an attack has been detected. Smart cards are still gaining in popularity and as long as embedded devices are used to store sensitive data, there will remain a necessity for new and improved security techniques.

References

1. F. Amiel, K. Villegas, B. Feix, and L. Marcel, "Passive and active combined attacks: Combining fault attacks and side channel analysis," in *Fault Diagnosis and Tolerance in Cryptography, 2007. FDTC 2007. Workshop on*, pp. 92 –102, sept. 2007.
2. K. Markantonakis, M. Tunstall, G. Hancke, I. Askoxylakis, and K. Mayes, "Attacking smart card systems: Theory and practice," *Information Security Technical Report*, vol. 14, no. 2, pp. 46 – 56, 2009. Smart Card Applications and Security.
3. M. Kuhn and O. Kmmerling, "Physical security of smartcards," *Information Security Technical Report*, vol. 4, no. 2, pp. 28 – 41, 1999.
4. J. Grand, "Protecting your crown jewels: an introduction to embedded security for hardware-based products," *Computer Fraud & Security*, vol. 2005, no. 10, pp. 13 – 20, 2005.
5. D. G. Abraham, G. M. Dolan, G. P. Double, and J. V. Stevens, "Transaction security system," *IBM Systems Journal*, vol. 30, no. 2, pp. 206 –229, 1991.
6. S. Weingart, "Physical security devices for computer subsystems: A survey of attacks and defenses," in *Cryptographic Hardware and Embedded Systems CHES 2000* (. Ko and C. Paar, eds.), vol. 1965 of *Lecture Notes in Computer Science*, pp. 45–68, Springer Berlin / Heidelberg, 2000. 10.1007/3-540-44499-8_24.
7. W. Rankl, "Overview about attacks on smart cards," *Information Security Technical Report*, vol. 8, no. 1, pp. 67 – 84, 2003.
8. D. Page, "Defending against cache-based side-channel attacks," *Information Security Technical Report*, vol. 8, no. 1, pp. 30 – 44, 2003.
9. X. Leng, "Smart card applications and security," *Information Security Technical Report*, vol. 14, no. 2, pp. 36 – 45, 2009. Smart Card Applications and Security.