Read-Proof Hardware from Protective Coatings

Pim Tuyls, Geert-Jan Schrijen, Boris Škorić,
Jan van Geloven, Nynke Verhaegh, Rob Wolters

Philips Research Laboratories, The Netherlands

Abstract. In cryptography it is assumed that adversaries only have black box access to the secret keys of honest parties. In real life, however, the black box approach is not sufficient because attackers have access to many physical means that enable them to derive information on the secret keys. In order to limit the attacker's ability to read out secret information, the concept of Algorithmic Tamper Proof (ATP) security is needed as put forth by Gennaro, Lysyanskaya, Malkin, Micali and Rabin. An essential component to achieve ATP security is *read-proof* hardware. In this paper, we develop an implementation of read-proof hardware that is resistant against invasive attacks. The construction is based on a hardware and a cryptographic part. The hardware consists of a protective coating that contains a lot of randomness. By performing measurements on the coating a *fingerprint* is derived. The cryptographic part consists of a Fuzzy Extractor that turns this fingerprint into a secure key. Hence no key is present in the non-volatile memory of the device. It is only constructed at the time when needed, and deleted afterwards. A practical implementation of the hardware and the cryptographic part is given. Finally, experimental evidence is given that an invasive attack on an IC equipped with this coating, reveals only a small amount of information on the key.

1 Introduction

Secure key storage is an important problem from a theoretical point of view as well as from a practical point of view. Recently, the theory of this topic started to develop in [1]. In the traditional cryptographic setting the attacker has only *black box* access to the secret information (keys) of the honest parties. In [1] this assumption was removed and the impact on the algorithmic aspects was investigated. It was observed that this problem is highly non-trivial and that in the most general setting no security can be guaranteed. The authors introduce the notion of *Algorithmic Tamper Proof* (ATP) security and show that this can only be achieved if the device has *read-proof hardware* together with a self-destructing capability and some hardwired data which can not be tampered with (Tamper Proof Hardware).

Read-proof hardware is hardware from which an enemy can not read any information on the data stored in it. Tamper-proof hardware contains data that can not be changed by an attacker. Clearly, to approach the black-box setting of cryptography as closely as possible, the (secret) keys have to be stored in read-proof hardware while public information such as algorithms and public keys have to be stored in tamper-proof hardware.

In this paper, we focus on the practical implementation of *read-proof* hardware. An attempt to translate the theoretical definition of read-proof hardware into a practical realisation shows that the theoretical definition has a rich variety of practical aspects. More specifically, it has been shown that there are many practical ways for *reading* out information from storage media, and read-proof hardware has to be resistant against all those methods. At a high level one can distinguish between *invasive physical attacks* [2], side channel attacks [3], and *fault induction attacks* [4]. An invasive physical attack is defined as an attack where the enemy physically breaks into the device by modifying its structure. A non-invasive physical attack is one where the attacker performs physical measurements without modifications to the device's structure. If the memory is not protected, a non-invasive physical attack (*e.g.* optical scrutiny) suffices to read out the memory. If the memory is covered with a protective layer, the attacker may attack invasively, *e.g.* by chemically etching away the layer, drilling a hole, or using a Focused Ion Beam (FIB), and then applying a microprobe. Once an attacker is able to open up a device and investigate its memory (EEPROM, ROM) he can (with reasonable efforts) obtain the keys. One of the main reasons that this readout is possible, originates from the fact that the key is stored in digital form as a string of zeros and ones. Since the state of a physical system representing a zero is distinguishable from the state representing a one, the key bits are observable.

We develop *read-proof* hardware resistant against invasive physical attacks, and non-invasive optical attacks. In order to make read-proof hardware, we build further on the idea of Physical Unclonable Functions introduced in [11] and further extended in [17]. A Physical Unclonable Function consists of a physical object that is inherently unclonable (since it contains many uncontrollable parameters during production). When a stimulus (usually called *challenge*) is applied to the object, it reacts with a *response* that can be measured. This challenge-response behaviour characterizes the structure completely. Furthermore the structure is tamper-evident, meaning that if the structure is physically damaged (by an attack), its challenge-response behaviour changes noticeably. Our solution for read-proof hardware is built on *coating* PUFs which can easily be integrated with an IC. In contrast to the usual setting of PUFs, where it is assumed that there is a huge number of challenge-response pairs, we only require one challenge-response pair. It is clear however how our construction is extended to many challenge-response pairs.

Read-Proof hardware in general and our construction in particular can be applied for secure key storage in Smart-Cards, SIM-Cards, TPMs (Trusted Platform Modules), DRM (Digital Rights Management) systems and in RFID tags [16].

1.1 Model

In our model, we build an IC equipped with read-proof hardware and ordinary memory (ROM or EEPROM). The secret key K of the cryptographic algorithm is extracted from the read-proof hardware only at the point in time when needed. All other required cryptographic components (algorithms, public keys) are stored in tamper-proof hardware and can not be changed by an attacker (but can be read) [1]. The enroller of the IC is considered to be trustworthy. He has a private key sk with which he certifies the data in the IC. The attacker can get hold of the device when it is in the field and can apply physical methods (invasive and non-invasive) to investigate the device and try to retrieve information on the secret key K.

We consider an adversary who has access to optical and invasive methods,

- Optical inspection equipment to look into memory cells (ROM).
- Etching methods (*e.g.* chemical) to remove protective layers.
- Focused Ion Beam to make holes in protective layers and allow for probing (of *e.g.* buses, memory).

1.2 Contributions

We have the following contributions:

- We state the requirements for practical read-proof hardware. Additionally we derive principles to satisfy these requirements. The main idea is not to store a key in digital form in a memory, but to extract it from an unclonable physical structure only at the point in time when needed. In this way the *time* that the *digital* key is present in the device (and hence susceptible to attack) is minimized.
- We describe a Coating PUF in detail (both the physics and the measurement circuit) and argue that it is opaque and chemically inert.
- It is shown how a Coating PUF has to be integrated with an IC and the required cryptographic primitives to meet the abovementioned goals. In particular, we present a new information reconciliation protocol on analog data to derive a unique fingerprint from the coating in a reliable way.
- Experimental evidence is given which shows that protection against invasive attacks is indeed obtained.
- Finally, when the read-proof coating hardware is combined with tamper-proof data and with a self-destruction capability, our solution additionally provides protection against *fault attacks*. This statement is based on the analysis performed in [1].

[1] In this paper, we do not develop a hardware solution for tamper-proof hardware.

1.3 Related Work

Since invasive attacks are sometimes performed by carefully removing protective layers of the IC (*e.g.* by etching), the smart-card industry is working on protective layers and coatings that are difficult to remove (*i.e.* removing the layer implies removing part of the IC, which renders the IC unusable). Additionally, sensors are sometimes built into the IC to check for the presence of the protective layer. If removal is detected, the IC will stop functioning and hence prevent an attacker from learning its secrets through playing games with the device. Although such coatings make life more difficult for the attacker, it turns out that in practice an attacker can often still successfully remove a coating (and possibly fool the sensors) and get access to the ICs interior. This is especially the case when the attacker has access to Focused Ion Beam (FIB) equipment, which makes it possible to reconnect wires in the interior of an IC [20]. The FIB is used to influence the (yes/no) signal that indicates the presence of the protective coating.

A more secure form of protective coatings, which has the potential to protect even against these sophisticated attacks, is the 'active coating' that was first introduced in [13] and further investigated in [14]. Our solution extends this work from the hardware point of view as well as from the cryptographic and design point of view. Additionally, we provide experimental data that show that our coating also provides protection against FIB attacks.

Another technology that is used to protect sensitive information stored in a memory is *memory encryption* [21]. This technology protects information from being exposed to an attacker who gets access to the memory. However, a key is still needed to encrypt and decrypt that information. The problem is then reduced to the secure storage of that secret key.

2 Read-Proof Hardware: Design and Requirements

2.1 Hardware requirements

In order to protect stored keys against invasive physical attacks, we propose that *no key shall be stored in digital form in the memory of a device*. Since there is no digital key in the memory, it can not be directly attacked. Instead, we propose to generate the key K only at the time when it is needed. The key is extracted from a *tamper evident* physical structure, integrated with the IC, by applying a challenge, measuring the response and carrying out the reconstruction phase of the helper data algorithm. In our case we extract the key from the protective coating, which behaves like a PUF (see Section 3). Additionally, we assume that the device has some memory where the public information (algorithms, public keys) is stored in a tamper proof way. Furthermore it has registers/RAM for storage of the key K at the time when needed. In order to be resistant against physical attacks, such a physical structure has to meet the following requirements:

1. 'Inscrutability' including 'opaqueness'. Measurements (both destructive and non-destructive) must not reveal accurate information about the composition of the physical structure.

2. The structure has to be *unclonable.* This requires two properties.
   - Physical unclonability. It should be hard to make a physical copy, even given accurate knowledge of the structure's composition.
   - Mathematical unclonability. It should be hard to construct a mathematical model that has a non-negligible probability of correctly predicting responses, even given accurate knowledge of the structure's composition.
3. The structure has to be tamper evident. Physical damage should significantly change the challenge-response behaviour of the structure.

Additionally, in order to be practically feasible, the following properties are required.

- It has to be easy to challenge the structure and to measure its response.
- It has to be cheap and easy to integrate the structure in an IC.
- From a robustness point of view, it should additionally have excellent mechanical and chemical properties, so that it cannot be detached from the IC (without causing damage to the coating and the IC).

2.2 Required Cryptographic Primitives

As mentioned before, the key is extracted from measurements on the coating. Since measurements on a physical structure are inherently noisy, the responses of such a structure can not be directly used as a secret key. This implies that we need a helper data algorithm/fuzzy extractor [10, 8] for reconstruction of the secret keys. A fuzzy extractor consists of a pair of algorithms (G, W) and two phases: an *enrolment* and a *reconstruction* phase. We will use the following notation: x denotes the measurement value of a response during the enrolment phase, while y denotes the corresponding value during the reconstruction phase. During enrolment, the key K is created for the first time. The helper data algorithm $W(.,.)$ is used during the enrolment phase and creates the helper data w based on the measurement value x during enrolment and the randomly chosen key K. The algorithm $G(.,.)$ is used during the key reconstruction phase for reconstruction of the key K as follows: $K = G(y, w)$.

As a second primitive, we need a standard signature scheme SS: $(\mathrm{SK}_g, \mathrm{Sign}, V)$, where SK_g is the secret-key generation algorithm, Sign the signing algorithm and V the verification algorithm. The enroller runs SK_g and obtains a secret-public key pair (sk, pk). (This is a one-time action). The public key pk is hard-wired in each IC (*i.e.* tamper-proof memory). With the secret key sk, the enroller signs the helper data w and $P(K)$ (where P is a one-way function). The signatures $\sigma(w)$ and $\sigma(P(K))$ are then stored [2] in EEPROM memory of the IC together with the helper data w.

[2] Instead of storing $\sigma(P(K))$, it is more secure to store $\sigma(P(K), \tilde{x})$ where $\tilde{x}$ is additional unpredictable key material that is obtained from the PUF (if necessary derived from the response of a second challenge). We have chosen not to include this in the notation throughout the paper for the sake of transparency.

2.3 Procedure for Generation and Reconstruction

Creation and reconstruction of the secret key is done as follows. First, the global statistical properties (noise level etc) of the behavior of the physical structure are determined. In particular, the entropy of the output of the physical structure is estimated and the secrecy capacity $C_S = \mathbf{I}(X;Y)$ (mutual information) of the channel describing the noisy observation is estimated [3]. This can be done using the methods described in [18]. These parameters determine the choice of an appropriate helper data algorithm/fuzzy extractor (G, W).
Enrollment This phase consists of two steps.

1. Generation of a key $K \in \{0,1\}^k$ and helper data w by running the enrolment phase of the helper data/Fuzzy Extractor pair (G, W) on the measurement outcome $X : (K, w) \leftarrow \text{Enrollment}(X)$.
2. The IC interprets K as a private key and generates the corresponding public key $P(K)$. Then the IC outputs $(w, P(K))$. The enroller signs these data and stores the signatures $\sigma(w)$, $\sigma(P(K))$ in the IC's EEPROM.[4]

Reconstruction The IC performs the following steps.

1. It retrieves $w, \sigma(w)$ from EEPROM and checks the signature $\sigma(w)$ by running V on w and $\sigma(w)$. If the signature is not ok, the IC shuts down permanently. Otherwise, it continues.
2. The IC challenges its physical structure and obtains the measurement value y (note that typically $y \neq x$ due to noise).
3. The data w and y are processed by the helper data algorithm G. This yields the key $K' \leftarrow G(y, w)$.
4. The IC computes $P(K')$. Then it runs V on $P(K')$ and $\sigma(P(K))$ using the public key pk. If the signature is ok, the IC proceeds and K can be used as a private key. Otherwise, the IC shuts down permanently.

3 Physical Unclonable Functions

In this section, we describe the physical component of read-proof hardware. Opaque physical systems that are produced by an uncontrollable production process, *i.e.* one that contains uncontrollable randomness, turn out to be good candidates for PUFs.

3.1 Coating PUFs

Coating PUFs are PUFs in the form of a protective coating that covers an IC. The coating consists of a matrix material which is doped with random dielectric

[3] This is a one-time event that is performed during a pre-processing step.

[4] Alternatively, K is used as a symmetric key. The IC outputs K and the enroller stores $\sigma(P(K))$ in the EEPROM. The circuit that outputs K is destroyed after this procedure.

particles. By random dielectric particles we mean several kinds of particles of random size, shape and location with a relative dielectric constant ϵ_r differing from the dielectric constant of the coating matrix. This is depicted in Fig. 1.

We used a mixture of TiO_2 and TiN particles in a matrix of aluminophosphate. This composition of the coating gives it the following properties. (i) The TiN-particles absorb light (from infrared up to ultraviolet) and hence make the coating opaque. Moreover they are conductive and very hard. (ii) The TiO_2-particles also absorb UV-light. (iii) The aluminophosphate matrix is very hard and chemically relatively inert. From this material the coating gets its protection against chemical substances. We note that the coating can be easily sprayed on top of the IC.

The top metal layer of the IC contains an array of sensors that are used to measure the local capacitance values of the coating. An example of a comb-shaped sensor structure is depicted in Fig. 2. Sufficient randomness in the measured capacitance values is obtained only if the dielectric particles are not much bigger than the distance between the sensor parts. The measurement circuit is integrated on the IC, so the measurements are done from within the IC. The measured capacitance values form the responses of this system and are protected against inspection from outside by the coating. Measuring the Coating PUF from the outside gives different capacitance results since the measurements are very sensitive to the precise locations of the dielectric particles. It is derived from the entropy formula in [5], that a coating PUF contains 6.6 bits of entropy per sensor.

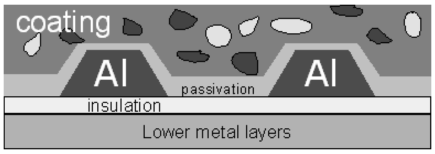

Fig. 1. Schematic cross-section of a Coating PUF IC. The upper metal layer contains aluminium sensor structures (Al) that are used to measure the local capacitance of the coating.

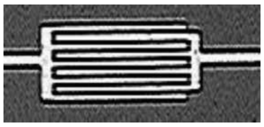

Fig. 2. Top-view microscope image of a single comb-shaped sensor structure (aluminum) in the top metal layer of the IC.

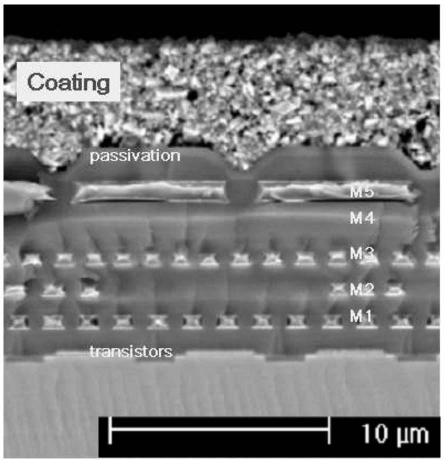

Fig. 3. Cross-sectional microscope image of a coating PUF IC. The sensors are located in metal layer 5 (M5).

4 Robust Fingerprint Extraction: Information Reconciliation

In this section, we describe the algorithmic part of our architecture. In order to derive secure keys from a physical source two steps are typically needed: Information Reconciliation and Privacy Amplification. The Information Reconciliation phase is basically an error correction step. The Privacy Amplification step guarantees that the extracted key is highly secure [6]. In this Section, we focus on the Information Reconciliation procedure. Since the capacitances obtained from a measurement are analog values we present an Information Reconciliation protocol for the analog case. This leads to a unique digital fingerprint that characterizes the coating.

4.1 Measurement Method

We have developed an on-chip measurement circuit that measures capacitance values at several sensors. The measurement principle is based on a period-modulated oscillator circuit, similar to Smartec's commercially available Universal Transducer Interface (UTI) [15], in which the oscillating frequency depends on the capacity at the sensor. A multiplexer circuit allows for the selection of one of several sensors. In order to derive measurement results that are insensitive to temperature and supply voltage variations, a 'three signal technique' is used (see also [15]). Based on this technique, we calculate a relative capacitance value at sensor i as follows:

$$\frac{C_i - C_0}{C_{\text{ref}} - C_0}. \tag{1}$$

Here, C_i with $i = 1, \ldots, M$, is a counter value that corresponds to the number of clock cycles that has occurred within 1024 oscillations of the measurement circuit when the i-th sensor is selected (note that M is the number of capacitance sensors). This counter value is related to the capacitance of the i-th sensor since this capacitance determines the oscillation frequency of the measurement circuit. The value C_0 is a reference counter value that is measured when no sensor is connected to the measurement circuit. Hence, the difference $C_i - C_0$ is proportional to the capacitance of the coating directly above the i-th sensor. The C_{ref} is a counter value from a (pre-defined) reference sensor. By taking the quotient (1) we remove temperature and voltage fluctuations.

4.2 Fingerprint Extraction: Information Reconciliation on Analog Data

In order to use the coating as a source of cryptographic keys, we start with an information reconciliation phase to derive a unique fingerprint $K \in \{0,1\}^k$ from the noisy measurements of the coating. In order to extract highly secure keys, it is advantageous to have the distribution of those fingerprints as close to the uniform random distribution on $\{0,1\}^k$ as possible. In order to extract noise-robust and highly random fingerprints at the same time from the analog coating measurements, we first apply a histogram equalisation to the analog data, making the distribution almost uniform. Then, the 'helper data' are defined in the transformed domain.

Notation and assumptions. We define the i.i.d. real stochastic variables $F_i := C_i - C_0$ and $F_{\rm ref} := C_{\rm ref} - C_0$, which are a property of the coating alone. Numerical instances of F_i are denoted as f_i.

The randomized manufacturing process of the coating gives rise to a probability distribution $\rho(F_i)$ for a capacitance value F_i at location i. Note that ρ is the 'true' capacitance distribution, *i.e.* without any noise. We incorporate temperature effects by postulating that F_i represents the true capacitance at a fixed reference temperature T_0. For any different temperature T, the capacitance changes to $F_i \cdot m(T)$, where m is a function satisfying $m(T_0) = 1$.

The distribution ρ has an average μ and a standard deviation σ. We assume that ρ is public knowledge and hence available to attackers. In order to equalize the distribution ρ, we define the cumulative distribution function q as

$$q(f) = \int_0^f \mathrm{d}x \rho(x). \tag{2}$$

Note that the stochastic variable $q(F) \in [0, 1]$ is uniformly distributed. A noisy capacitance measurement at temperature T and location i results in a stochastic variable F_i', $F_i' = F_i m(T) + N_i$, where the noise N_i is independent of T, i and F_i and also independent of previous measurements. We assume that N_i is gaussian with zero mean and fixed variance $\sigma_{\rm N} \ll \mu$.

In order to deal with the noise, we define quantisation intervals as follows. The f-axis is divided into L equiprobable parts with boundaries at t_j, $j = 0, \ldots, L$. The boundaries are placed according to $t_j = q^{-1}(j/L)$. Here q^{-1} is the inverse function of q.

Enrolment. Enrolment occurs under tightly controlled circumstances. The temperature is T_0. For each IC the following steps are performed.

- The capacitance values f_i for $i = 1, \ldots, M$ and $f_{\rm ref}$ are measured. The value $f_{\rm ref}$ is stored in the IC for later use as a normalising factor.
- For each capacitance f_i ($i = 1, \ldots, M$) the quantised value $I_i \in \{0, \ldots, L-1\}$ is determined, $I_i = \lfloor L\, q(f_i) \rfloor$.
- Helper data W_i is computed as follows, $W_i = I_i + 1/2 - Lq(f_i)$. The helper data $\{W_i\}$ is stored in the EEPROM of the IC.
- From the set $\{I_i\}$ a codeword in an error-correcting code is created as follows. We will assume that L has the form $L = 2^a$. In this case it is advantageous to assign to each quantised value $I_i \in \{0, \ldots, L-1\}$ a code word from a binary Gray code. The Gray code has the nice property that the Hamming distance between two neighbouring code words equals one. In this way a measurement error $I_i' = I_i \pm 1$ has the effect of flipping only a single bit in the code word.

By concatenating the Gray codes from all the sensors a string X is obtained of length $n = M \log L$. A secret $K \in \{0,1\}^k$ is randomly generated. Then, using the 'XOR-trick' as described in [9, 16] a codeword $c_K \in \{0,1\}^n$ of an error-correcting code $\mathcal{C}$ is computed. Further helper data w called 'conversion data' are derived that map X onto c_K. The conversion data w are stored in the IC's EEPROM.
- The total set of helper data that has to be signed and stored in EEPROM is given by, $(\{W_i\}, w, f_{\text{ref}})$.

Key Reconstruction. At a later time, the IC reconstructs the key from noisy capacitance measurements combined with the enrolment/helper data. The temperature is not controlled.

- The IC measures noisy values $f'_i, i = 1, \ldots, M$ and f'_{ref} and looks up the values f_{ref}, $\{W_i\}$ and w from memory.
- For each $i = 1, \ldots, M$ the IC computes a reconstruction of I_i as follows,

$$I'_i = \left\lfloor Lq(f_{\text{ref}} \frac{f'_i}{f'_{\text{ref}}}) + W_i \right\rfloor . \tag{3}$$

- From the values I'_i the IC constructs a string Y by concatenating Gray codes in the same way as was done during enrolment. Then it applies the mapping w to Y. Finally it performs the decoding step of the 'XOR-trick' (for details see the extended version). This yields the secret key K, provided that the number of measurement errors does not exceed the correction capacity of the error-correcting code $\mathcal{C}$.

Properties of the method. The helper data method described above has the following properties (for details we refer the reader to the extended version of this paper).

- The noise in I'_i is linear in L, leading to a practical bound on the number of quantization intervals. To reduce the probability p_E of a quantization error to 10%, we need $L < 8.8$ in our experimental ICs.
- The maximum length of a secret key extracted from the coating is $M \log L \cdot [1 - h(p_E)]$.
- As long as the attacker does not have better knowledge of ρ than the manufacturer, the helper data $\{W_i\}$ do not leak any information about the key K.

5 Experimental Results

We have produced a batch of ICs containing the coating and the measurement circuit of Section 3. The top metal layer of the IC contains 31 sensor structures. Each sensor structure has a capacitor area of $120 \times 120\ \mu m^2$. The top of the ICs is covered with a coating. The coating consists of a mono-aluminum phosphate matrix that is doped with TiN and TiO_2 particles.

5.1 Capacitance measurements

We have measured the capacitances from 36 different ICs. On each IC, one of the 31 sensors is used as a reference sensor which leads to the value C_{ref}. The C_0 value comes from an internal measurement in which the measurement circuit is not connected to a sensor. The measurements at the 30 remaining sensors form the C_i values. We compute the stabilized capacitance value B_i of sensor i as follows:

$$B_i = f_{\mathrm{ref}} \frac{f_i' - (\frac{1}{M} \sum_{i=1}^{M} f_i')}{f_{\mathrm{ref}}'} \tag{4}$$

Note that this method differs slightly from Eq. (1). In Eq. (4) we subtract the average of f_i' over the IC in order to compensate for unwanted coating thickness variations that are caused by the manufacturing process.

Fig. 4 shows the B_i measurements [5] of 30 sensors, measured at 6 different ICs. In the extended version of the paper, we show the influence of temperature variations on the values of f_i' and B_i.

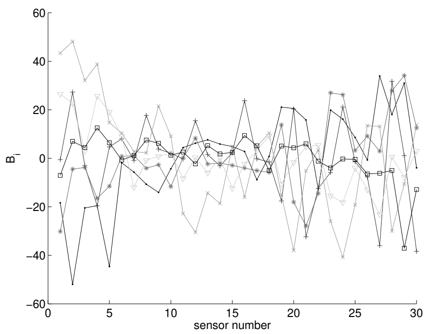

Fig. 4. Measured stabilized capacitance values B_i at 30 sensors of 6 different ICs.

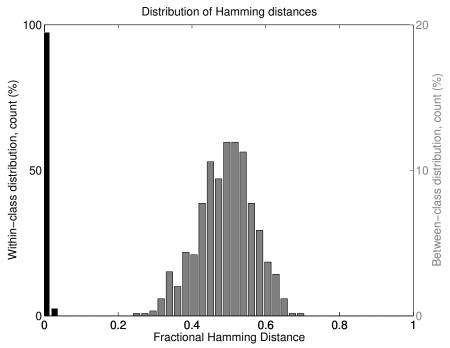

Fig. 5. Histogram of fractional hamming distances between fingerprints derived from the same IC (within class) and between fingerprints derived from different ICs (between-class).

The capacitance measurements show an average within class standard deviation of of $\sigma_N = 0.97$ and an average between class standard deviation of $\sigma_{B_i} = 18.8$. In our practical setup we derive 3 bits per sensor, which gives the best results w.r.t. robustness.

[5] Note that B_i is dimensionless since f_i is the difference between two counter values (see section 4.1). Measurements of similar coating and sensor structures with a Hewlett Packard 4192 impedance analyzer show that the average capacitance value is around 0.18 pF (*i.e.* corresponding to $B_i = 0$ in Fig. 4).

5.2 Fingerprints

By way of example, we show key extraction from our experimental data according to the method of Section 4.2 . First the distribution ρ was estimated empirically by measuring all 30 sensors on 36 ICs. The interval $q(f) \in [0, 1]$ was divided into $L = 2^3 = 8$ intervals. We used a Gray code to make a 3-bit encoding of each integer I_i. In this way we derived fingerprints of 90 bits. Histograms of the fractional Hamming distances between the extracted fingerprints for both the within- and between-class distribution are shown in Fig. 5. The between-class distribution is centered around a fractional Hamming distance of 0.5, which means that the fingerprints derived from 2 different ICs will on average differ in 50% of the bits.

It turns out that bit strings derived from the same IC (within-class distribution) have fewer than 4 errors. Hence, an error-correcting code that corrects 4/90 of all bits is suitable in this case. Using an optimal error correcting code (*i.e.* one that achieves maximal key length), one would get a key length of approximately $k = 66.4$ bits. In practice one can *e.g.* use a BCH code which turns 63 bits of the 90 into a key of 45 bits. The remaining bits can be turned into additional key material with a second error-correcting code. The practical choice of the error-correcting code has to be optimized. This is not the subject of this paper.

5.3 Attack Detection

Physical attacks in which the coating is damaged are detected from the capacitance measurements. A well-known method for getting access to internal circuit lines of an IC, is by making a hole through the IC with a Focused Ion Beam (FIB). Afterwards the hole is filled with metal such that a surface contact is created. This can be used by the attacker for easy access to an internal line (*e.g.* for eavesdropping on a signal). In Section 5.3, we show the effect of a FIB attack with gallium particles.

A FIB was used to create two holes in one of the Coating PUF ICs by shooting gallium particles on two areas of size $100\mu m$ x $100\mu m$ and depth of around $1.5\mu m$ in a coating of thickness $6\mu m$, see Fig. 6.

Fig. 7 shows the effect of the FIB attack on the measured capacitances f_i'. After the FIB attack, several sensors measure a significant change in capacitance value. This is due to the fact that ions are implanted into the coating, which changes its behaviour non-locally. The derived fingerprint after the FIB attack differs in 14 of the 90 bits. Table 1 summarizes the direct effect of Gallium FIB and Argon beam attacks on a single sensor.

6 Security of the Coating: Experimental evidence

Since the coating is opaque, optically looking into the digital memory is very hard without damaging the coating. Furthermore, since the coating is tough and

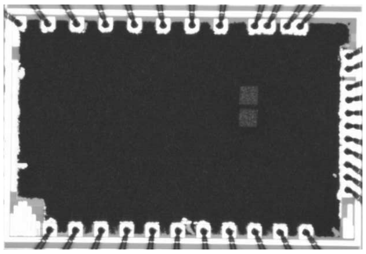

Fig. 6. Top view of a Coating PUF IC in which two holes have been shot with a Gallium FIB.

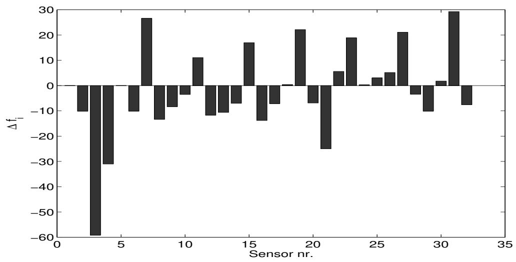

Fig. 7. Differences in capacitance f_i' between measurements of Coating PUF IC 89 before and after the Gallium FIB attack.

Beam type	Hit area	Depth	Δf
Gallium	$100\mu m$ x $100\mu m$	$1.5\mu m$	-40
Gallium	$15\mu m$ x $15\mu m$	$4\mu m$	-34
Argon	$100\mu m$ x $100\mu m$	$1.5\mu m$	-28

Table 1. Change of capacitance measured by the sensor lying under the area of impact of the beam.

chemically inert, it is very hard to remove mechanically or chemically. Next, we discuss some more advanced attacks and show the resistance of the coating against these attacks.

6.1 Impact of FIB Attack on the Keys

We discuss an attack, where the attacker first uses a FIB to make a hole in the coating. Then, he makes the IC start the key reconstruction phase described in Section 2.3. During the reconstruction phase, he uses his micro-probe(s) to retrieve the key bits. We denote the measurement values after the FIB-attack by a random variable Z and the key extracted after the FIB-attack by K'. During step 4 of the reconstruction phase, the IC checks whether the extracted key K' is correct by running the algorithm V on $P(K')$ and $\sigma(P(K))$. If the signature is not ok, the device is destructed. Hence, the attacker gets the information that the extracted key K' is incorrect. We assume furthermore that the attacker can capture the noisy measurement Z by using his microprobe [6] (note that this is a worst case assumption). It is natural to investigate how much uncertainty there still remains about the original key K.

In the extended version, we construct a model that represents the FIB damage as an additional bit error rate ϵ on top of the already present bit error rate α due to measurement noise, with $\epsilon > \alpha$. This effectively leads to a noisy channel $X \to Z$ with combined error rate $\chi = \alpha(1-\epsilon)+\epsilon(1-\alpha)$ as seen by the attacker.

[6] Since he also gets the helper data w from the ROM, this implies that he can reconstruct K'.

The amount of uncertainty he has about K can be expressed as a number N_c of 'candidate' keys, which turns out to be of order

$$N_c = \mathcal{O}\left(2^{n(h(\chi)-h(R\alpha))}\right), \tag{5}$$

where R is a constant larger than 1 and the function h is defined as $h(p) = -p\log p - (1-p)\log(1-p)$. With the ICs that we have, the parameters α, $R\alpha$ are given by $\alpha = 1/30$, $R\alpha = 4/90$. The values for ϵ range from $\epsilon = 8/90$ to $\epsilon = 14/90$. Therefore we take an average value $\epsilon = 11/90$. In practice one would like to have a key of length 128 bits. Given these error rates that would require $n = 174$ (then $\mathbf{I}(X;Y) = 128$). Substituting this value of n into Eq. (5), we obtain $N_c = 2^{51}$.

7 Conclusions and Future Work

In this paper we have given an implementation of read-proof hardware. The main idea is: "thou shalt not store secret keys in digital memory". The key should be derived from a protective coating containing a lot of randomness. The key is obtained from capacitance measurements on the coating. In order to extract the key from the measurement values, we have developed a secure helper data algorithm that is implemented on the IC. We have provided experimental evidence that our construction is secure against invasive physical attacks such as attacks with a Focused Ion Beam.

One of the main open questions that remains is the resistance of this construction against side-channel attacks. In order to thwart those attacks, the cryptographic part has to be implemented in a side channel resistant way (which can be done with existing methods). Currently, it is being investigated whether the measurement circuit itself is susceptible to side-channel attacks such as Electromagnetic Analysis, Power Analysis and Timing analysis. Although no leakage has been reported yet, countermeasures against leakage of the measurement circuit are being considered.

Another open question is to investigate whether this technique can also be applied at the back of the IC to provide protection against backside attacks.

References

1. R. Gennaro, A. Lysyanskaya, T. Malkin, S. Micali and T. Rabin, *Algorithmic Tamper-Proof Security: Theoretical Foundations for Security against Hardware Tampering*, In Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004, Cambridge, MA, USA, February 19-21, volume 2951 of LNCS, pages 258-277, Springer-Verlag.
2. R. Anderson and M. Kuhn, *Low Cost Attacks on Tamper Resistant Devices*, In M. Lemmas et al., editor, Proceedinggs of Security Protocols, 5th International Workshop, volume 1361 of Lecture Notes in Computer Science, pages 125-136, Paris, France, April 1997, Springer-Verlag.

3. P.C. Kocher, J. Jaffe, B. Jun, *Differential Power Analysis*, Proceedings of the 19th International Conference on Cryptology, Advances in Cryptology, volume 1666 of LNCS, pages 388-397, 1999, Springer Verlag.
4. E. Biham and A. Shamir, *Differential Fault Analysis of Secret Key Crypto Systems* Advances in Cryptology, Crypto 97.
5. B. Škorić, S. Maubach, T. Kevenaar, P. Tuyls, *Information-theoretic analysis of coating PUFs*, http://eprint.iacr.org/2006/101, accepted for publication in the Journal of Applied Physics.
6. C.H. Bennett, G. Brassard, C. Crepeau, and U. Maurer, *Generalized Privacy Amplification*, In IEEE Transactions on Information Theory, vol 41, 6, pages 1915-1923, 1995.
7. H. Bar-El, *Known Attacks Against Smartcards*, Discretix Technologies Ltd. http://www.infosecwriters.com/text_resources/pdf/Known_Attacks_Against_Smartcards.pdf
8. Y. Dodis and M. Reyzin and A. Smith, *Fuzzy Extractors: How to generate strong keys from biometrics and other noisy data*, In C. Cachin and J. Camenisch Editors, Proceedings of Eurocrypt 2004, volume 3027 of Lecture Notes in Computer Science, pages 523-540, Springer-Verlag
9. A. Juels and M. Wattenberg, *A fuzzy commitment scheme*, 6th ACM Conference on Computer and Communication Security, pp.28-36, 1999.
10. J.P. Linnartz, P. Tuyls, *New Shielding Functions to Enhance Privacy and Prevent Misuse of Biometric Templates*, AVBPA 2003, LNCS 2688, pp.393-402.
11. R. Pappu, *Physical One-way functions*, Ph.D. thesis, MIT, 2001.
12. R. Pappu, B. Recht, J. Taylor, N. Gershenfeld, *Physical One-way functions*, Science Vol.297, 2002, pp.2026-2030.
13. R. Posch, *Protecting Devices by Active Coating*, Journal of Universal Computer Science, vol.4 no.7, 1998.
14. G.A. Kamendje, R. Posch, *Intrusion aware CMOS Random Pattern Generator for Cryptographic Applications*, In Peter Rossler and Andreas Dorderlein Editors, Proceedings of Austrochip 2001, Vienna, Austria, 12 October 2001 ISBN 3-9501517-0-2.
15. Smartec, *Universal Transducer Interface evaluation board*, Specifications v3.0, http://www.smartec.nl/pdf/Dsuti.pdf .
16. P. Tuyls, L. Batina, *RFID tags for Anti-Counterfeiting*, RSA 2006 conference, San Jose, USA, Feb. 13-17, 2006.
17. P. Tuyls, B. Škorić, *Secret Key Generation from Classical Physics*, In Mukherjee et al. editors, AmIware, Hardware Technology Drivers of Ambient Intelligence, Philips Research Book Series, Kluwer, pages, 421-447,2005.
18. T. Ignatenko, G.J. Schrijen, B. Škorić, P. Tuyls, F. Willems, *Estimating the Secrecy-Rate of Physical Uncloneable Functions with the Context-Tree Weighting Method*, accepted at ISIT 2006
19. M. Witteman, *Smart card security analysis*, IPA Spring Days on Security, Kapellerput, Heeze, April 18-20, 2001. http://www.win.tue.nl/ipa/archive/springdays2001/witteman.ppt
20. M. Witteman, *Advances in Smartcard Security*, Information Security Bulletin, July 2002, pp.11-22. http://www.riscure.com/articles/ISB0707MW.pdf
21. J. Yang, L. Gao, Y. Zhang, *Improving Memory Encryption Performance in Secure Processors*, IEEE. Trans. Computers, vol 53, 5, 1-11, 2005.