D3.1 New Methodologies for Security Evaluation

Project number:	238811
Project acronym:	**UNIQUE**
Project title:	Foundations for Forgery-Resistant Security Hardware
Start date of the project:	01.09.2009
Duration:	30 months

Deliverable type:	Report
Deliverable reference number:	238811/ D3.1 / v1.0
Deliverable title:	New Methodologies for Security Evaluation
WP contributing to the deliverable:	WP3
Due date:	2011-05-31 (M21)
Actual submission date:	2011-05-31 (M21)

Responsible organisation:	Intel
Authors:	Intel (Patrick Koeberl), TU Darmstadt (Christian Wachsmann), Thales (Jean-Christophe Courrège, Anne-Sophie Rivemale, Jérôme Quévremont), Intrinsic ID (Vincent van der Leest, Ilze Eichhorn)
Abstract:	This document presents novel methodologies to assess the security of PUFs (Physically Unclonable Functions).
Keywords:	PUF, security evaluation, attacks, Common Criteria

Dissemination level:	Public
Revision:	v1.0

Instrument:	STREP
Thematic Priority:	ICT

Table of Contents

1 Introduction

Physically Unclonable Functions (PUFs) are promising new security primitives which exhibit a range of security properties that make them attractive and useful for a variety applications. Product-focused security evaluation methodologies such the Common Criteria [4], when used to evaluate PUF-based systems, will need to be aware of these as well as any specific vulnerabilities that PUFs might introduce. The goal of this deliverable is to develop methodologies to evaluate PUF security and identify those areas in the Common Criteria that need refining in order to accommodate evaluations of PUF-based systems. Additionally, we recognize that today's complex global supply chains put a new premium on security assurance in which product-focused evaluation methodologies are necessary but not sufficient. With this in mind we outline areas where PUF-based technologies can increase supply chain assurance.

1.1 Outline

Section 2 of this deliverable presents attack methods which are focused on Physically Unclonable Functions. In Section 3 we define the attacker's tools and capabilities in order to develop an attacker model. Logically Reconfigurable PUFs (LR-PUFs) represent a new security primitive developed within the UNIQUE project. Section 4 performs a lightweight security evaluation of LR-PUFs and suggests mitigations and design rules which should be considered in any PUF-based implementation.

Evaluation gaps which were indentified during the baseline Common Criteria Evaluation of UNIQUE Task 3.1 are presented in Section 5 and an overview of how PUFs can benefit ICT supply chain security is given in Section 6. The UNIQUE security evaluation test plan is detailed in Section 7 and identifies those tests that will be carried out on the physical prototypes developed during the UNIQUE project.

2 PUF-Specific Attacks

Since their introduction in the last decade interest in Physically Unclonable Functions has grown steadily as the practical relevance of this new security primitive for security applications is recognised. The introduction of a new security primitive brings new opportunities to the table but also potentially new vulnerabilities and unexpected behaviours. In this section we review a number of attacks specific to PUFs which have the potential to materially affect the security properties of any system building on the PUF primitive. For an overview the PUF primitive and constructions that are referred to in this section please refer to Unique Deliverable D2.1 [44].

2.1 Fault Attacks on PUFs and Fuzzy Extractors

Fault attacks aim to force erroneous behaviour in a device by manipulating it in some way in order to inject a fault. Faults may be injected in many ways, for example by operating the device in extreme environmental conditions or by injecting a transient fault into a device transistor by means of targeted optical methods. The implications of fault attacks specific to PUFs and fuzzy extractors should be carefully considered. Many PUF applications require error correction of the noisy PUF outputs, typically by employing a fuzzy extractor. Attempts to force the PUF outside its normal operating envelope by varying supply voltage or ambient temperature will change the PUF noise characteristics beyond the capabilities of the error correction solution. In the first instance such an attack would result in a Denial of Service for downstream functions reliant on the corrected PUF response. However, since error correction units are likely to exhibit data-dependent behaviour, a fault attack on the PUF could cause unintended leakage on a fuzzy extractor side channel.

Fuzzy extractors have the convenient property that the associated helper data need not be private. However, the security guarantees of traditional fuzzy extractors only hold for those cases where the helper data cannot be modified by the adversary, as pointed out in [2]. Thus a fault attack on the helper data must be considered. Fuzzy extractors which are capable of coping with modified helper data are termed *robust fuzzy extractors*. An example is presented in [17].

2.2 Side Channel Attacks on PUFs and Fuzzy Extractors

Side channel attacks are hardware attacks that aim to extract secret data (e.g., a cryptographic key) from an electronic component. Hereby, the adversary observes the behaviour (e.g., the power consumption, electro-magnetic radiation, and/or timing behaviour) of the component while it is using the secret data to be extracted. Since the behaviour of the component is typically dependent on the data processed, it can leak information on this data.

Research on side channel attacks against PUFs and fuzzy extractors has been started only recently and there are only a few published results. Karakoyunlu et al. [18] and Merli et al. [26] show side channel attacks on typical fuzzy extractor implementations. Moreover, Merli et al. [26] theoretically discuss potential side channel attacks on different PUF types.

PUFs are typically used in combination with fuzzy extractors, which eliminate the noise (error correction) and enhance the entropy (privacy amplification) of the PUF responses. In most PUF use cases, the plain PUF responses (i.e., before error correction/privacy amplification) must be secret. Otherwise it may be possible to perform model building attacks and to clone the PUF (see Section 4). All side channel attacks on PUFs and fuzzy extractors that have been shown so far aim to extract the plain PUF response from the fuzzy extractor and are independent of the underlying PUF construction.

Fuzzy extractors used with PUFs are typically implemented using offset coding and BCH or RS error correcting codes. Standard implementations of these mechanisms usually show input-dependent behaviour and hence, are vulnerable to side channel attacks.

Karakoyunlu et al. [18] point out that most implementations of BCH and RS decoders skip the error correction process in case they did not detect an error in the codeword, which can be exploited by a simple power analysis attack to extract the plain PUF response. They show for a software implementation of a fuzzy extractor on a low power microcontroller that in case of BCH codes only one single power trace is needed, while in case of RS codes they need 2^{m-1} traces, where m is the length of the symbols of the RS code. To prevent simple power analysis attacks, Karakoyunlu et al. [18] propose to eliminate all data dependent control flow in the fuzzy extractor implementation. However, they show that even with this change a more sophisticated side channel attack, i.e., a differential template attack, is possible, which exploits the fact that the helper data of the fuzzy extractor can be chosen externally, i.e. by the adversary.

Merli et al. [26] show an attack against the privacy amplification part of a fuzzy extractor, which is usually implemented based on a lightweight cryptographic hash function. In their case, they attacked an FPGA implementation of a fuzzy extractor using the Toeplitz universal hash function. Similar to the attack by Karakoyunlu et al. [18], the adversary must also be able to choose the helper data input to the fuzzy extractor. Moreover, Merli et al. [26] theoretically discuss potential side channel attacks on the PUF itself. For instance they claim that the outputs of arbiter and ring oscillator PUFs[1] can be extracted by observing the electromagnetic emission of the arbiters that are activated when the PUF is challenged. The difficulty of this approach is locating the arbiters on the ASIC or FPGA chip, which could become feasible in the near future by using advanced electromagnetic cartography methods. Further, Merli et al. [26] point out that ring oscillator PUFs can also be attacked by analyzing the electromagnetic emission of the counter and/or comparator that is used to determine which of the ring oscillators was faster, i.e., the response bit of the PUF. A potential countermeasure may be to obfuscate the counter and the arbiters in the chip making them hard to detect. Attacks based on analyzing the characteristic frequencies of the ring oscillators themselves may be more practical.

Research on side channel attacks on PUFs and fuzzy extractors has just started. While fuzzy extractors are algorithms, it seems that standard side channel attack techniques as well as standard protection mechanisms can be applied to them

[1] Note that ring oscillators show a frequency-specific characteristic behaviour upon activation.

without major difficulties. However, since PUFs typically are analogue circuits, performing standard side channel attacks against them and hardening them against side channel attacks may not be straightforward.

2.3 Modelling Attacks on Delay-Based PUFs

Delay-based PUFs were introduced in [9] and comprise the arbiter and ring oscillator PUF variants. The linear construction of the delay circuit at the core of both of these PUF types can be described via an additive linear delay model allowing the PUF behaviour to be modelled, a fact that was recognised early in the development of delay-based PUFs. Model-building attacks collect a subset of challenge-response pairs from the overall challenge-response space and derive a mathematical model of the challenge-response behaviour of the PUF from this subset, i.e., a formula that allows computing a numerical approximation of the PUF response for a given PUF challenge. A number of mitigations have been proposed, all based on inserting non-linearities into the delay circuit (see, e.g., [21], [24], [23]) However, all have been shown to be vulnerable to modelling attacks in [32] which leverages machine learning techniques based upon Logistic Regression and Evolution Strategies.

The results in [32] show that the non-linear arbiter variant introduced in [23] (termed a "lightweight secure PUF") exhibits the most resistance to modelling attacks with an attack time measured in months for a 128-bit challenge configuration and 100,000 Challenge-Response Pairs (CRPs). The relative ease with which modelling attacks have been developed for delay-based PUFs has implications for any system embedding this PUF variant as a security primitive. Modelling attacks can be mitigated by ensuring the attacker cannot easily challenge the PUF or access the raw PUF responses. The controlled PUF concept was introduced in [10]. Here chosen challenge attacks are prevented by placing a hash function on the PUF input. A hash function is also placed on the PUF output to prevent the raw PUF responses from being accessed. Clearly this doesn't address the fundamental weakness of delay-based PUFs to modelling attacks and therefore invasive attacks to collect raw CRPs are still possible.

3 Attacker Model

In this section we develop an attacker model based on the attacker's tools and capabilities. We distinguish between three attacker types: A non-invasive attacker, a semi-invasive attacker and an invasive attacker.

3.1 Non-invasive Attacker

We define non-invasive attacks to be those which do not violate the physical integrity of the device package, i.e., the attacks that are mounted outside the external physical boundary of the device. Examples of non-invasive attacks are those exploiting side channels such as power [19], timing [20] and electromagnetic emissions [29]. For example, fault attacks attempt to inject a fault leading to an adverse result such as leakage of a cryptographic secret. Fault injection attack methods include attacks on the device's power supply or clock, such as introducing transients or over- and under-voltage conditions, or forcing a temperature excursion outside the normal operating range. Other non-invasive attack methods include exploiting flaws in the protocols used to communicate with the device or using factory test and debug interfaces in unintended ways.

The equipment requirements for mounting a non-invasive attack are relatively modest: A logic analyzer and mid-range DSO (Digital Sampling Oscilloscope) with high-impedance FET probes represent the most demanding of these and would be met by any well-equipped university laboratory.

3.2 Invasive Attacker

Invasive attacks expose the die surface by removing the device package and passivation[2] layer with the intention of probing signals of interest, performing circuit modifications or reverse engineering the device functionality. Accessing device features of interest by the backside is also possible. In this case the die substrate is mechanically thinned and the passivation layer is left intact. Invasive attacks require specialized or even bespoke equipment and skills normally associated with semiconductor failure analysis laboratories and can be considered out of reach for all but well-resourced attackers. Nevertheless, it is possible to outsource some of the required operations – a fact that must be taken into consideration when considering the likelihood of such attacks. The removal of protective features such as the passivation layer or substrate thinning reduces the integrity of the die, thus invasive attacks are destructive in nature.

The options available to an invasive attacker can broadly be classified into three methods:
Probing, circuit editing, and reverse engineering. Mechanical probing attacks attempt to monitor a circuit node by placing a probe in mechanical contact with the feature of interest and monitoring it during circuit operation. The probe can

[2] The passivation layer is a protective coating which is formed or deposited on the IC surface (front side) in order to protect the underlying structures from oxidation damage.

also be used to inject a signal into the circuit. In a deep-submicron technology node it is possible that the feature of interest is too small to contact (a fine metal interconnect) and is obscured by higher interconnect layers. In these cases it may be possible to use a Focussed Ion Beam (FIB) to deposit a probe pad which is then connected to the feature of interest. The probe pad dimensions ensure a reliable electrical contact. This technique can also be used from the backside on a thinned substrate. Mechanical probes place a capacitive load on the signal being measured and as a result the bandwidth of the measurement will be restricted.

Circuit editing involves using a FIB to modify circuit functionality. This is achieved by cutting existing metal interconnect and depositing new metal in order to form new circuit connections. A complex circuit edit can take many hours of FIB time with no guarantee of a successful outcome.

In order successfully mount a probing attack or circuit edit operation the attacker must have some knowledge of where to direct his efforts. Insider attacks where the attacker has access to design documentation and the associated CAD database are not considered here. Regular structures on the die such as memories and data paths are recognizable and can inform the attack strategy. Most PUF variants have regular structures such as SRAM, ring oscillator and arbiter PUFs. For deep submicron technology nodes, near-IR imaging through the substrate is required since front-side imaging at optical wavelengths will be hindered by interconnect layers and layer planarization. Once the macro structures of interest have been indentified, depending on feature size, a Scanning Electron Microscope (SEM)[3] can be used to examine items of interest, for example, a specific transistor or metal interconnect. Reverse engineering the complete circuit netlist is a possibility and is a commercially available service. The process is destructive, requiring one device sample per process layer, and uses automated image processing techniques to extract a complete transistor- or gate-level netlist [41].

3.3 Semi-invasive Attacker

The semi-invasive attack classification was introduced in [37]. In a semi-invasive attack the device packaging is removed in order to expose the die, leaving the passivation layer on the device front-side intact, ruling out front-side mechanical probing and FIB operations. In [37], several techniques are introduced utilizing low cost equipment such as laser pointers, photographic flash guns and UV light sources. These low-end techniques do not scale to modern technology nodes therefore we extend the semi-invasive definition to to include high-end non-contact probing technologies and some level of mechanical substrate thinning in order to increase the efficiency of the non-contact probing technologies.

Contactless probing technologies capable of measuring a time-varying waveform on an internal device node include electron-beam probing [22] and those based on photonic emission. E-beam probers are based on Scanning Electron Microscope (SEM) technology and are capable of probing through passivation

[3] Note that SEMs are typically incorporated into the FIB equipment and termed a dual beam FIB.

layers to lower interconnect layers to a limited extent. The technique has been extended to allow probing of active devices through the device substrate [34]. E-beam probers have largely been superseded by techniques based on optical phenomena due to the bandwidth limitations placed on the measured signal.

Optical probing techniques such as Time Resolved Emission (TRE) [25] and Laser Voltage Probing (LVP) [47] enable high bandwidth measurements with picosecond accuracy. Although these techniques use light in the near-IR wavelength, they have been shown to scale with modern technology nodes by using Solid Immersion Lens (SIL) techniques to overcome spatial resolution constraints.

4 LR-PUF Security Evaluation

Logically Reconfigurable PUFs (LR-PUFs) are introduced in UNIQUE Deliverable D1.2 [43]. In this section we evaluate LR-PUF security in the following manner: First, the security assumptions of the LR-PUF specification are restated. The LR-PUF architecture is then examined in the context of two use cases and possible attacks are identified and organised into attack trees. Finally, mitigations and design rules are suggested.

An LR-PUF is a PUF whose challenge/response behaviour depends on both the physical properties of the PUF and the logical state maintained by a control logic unit. Ideally, an LR-PUF should resemble a *physically* reconfigurable PUF. This implies that it should be infeasible for an adversary to predict the response to a challenge of an LR-PUF for some state, even if he knows the responses to this challenge of the *same* LR-PUF but for *other* (e.g., old) states. Here, we must distinguish between the case where the adversary aims to predict the responses of the LR-PUF for the current state (e.g., to forge a PUF response in an authentication protocol) or for a previous LR-PUF state (e.g., to recover an old key bound to the previous LR-PUF state). Moreover, in most applications of reconfigurable PUFs, it must be infeasible to set the state of an LR-PUF to a specific value, which would allow resetting the LR-PUF to a previous state and may help the adversary to predict LR-PUF responses.

The architecture of a generic LR-PUF is shown in Figure 1. The control logic maintains a state S and provides an algorithm $\text{query}_S()$ for querying and rcnf() for reconfiguring the LR-PUF. The algorithm $\text{query}_S()$ consists of an input transformation function $\text{mapin}_S()$ and an output transformation function $\text{mapout}_S()$: $\text{query}_S(x)$ transforms c into w, evaluates $y \leftarrow \text{PUF}(w)$, and returns $r \leftarrow \text{mapout}_S(y)$. The algorithm implementing rcnf() reconfigures the LR-PUF by changing the current state S to a new independent state $S' \leftarrow$ rcnf().

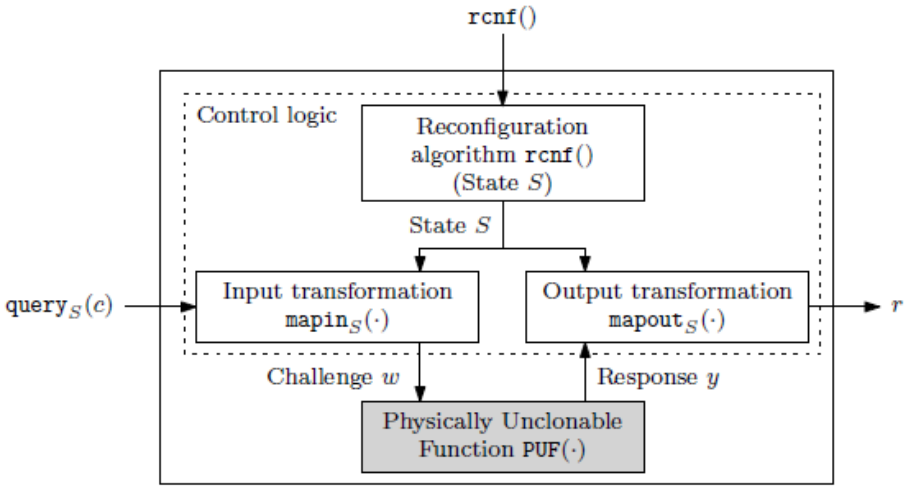

Figure 1 Generic LR-PUF Construction

4.1 Security Assumptions

The LR-PUF architecture and protocol design of [43] follow best practices of modern cryptography, proving the security of the LR-PUF construction by means of a formal security model. As part of this process assumptions must be made on physical aspects of the LR-PUF: First, the physical PUF underpinning the LR-PUF is assumed to be physically unclonable and unpredictable. The algorithms mapin(), mapout(), and rcnf(), are publicly known. Finally it is assumed that the adversary knows the current and previous states *S* of the LR-PUF but cannot set the LR-PUF state to a value chosen by the attacker.

4.2 UNIQUE Use Case Analysis

UNIQUE Deliverable D1.2 [43] defines two use cases building on the LR-PUF construction which are briefly outlined in the following. The first use case involves a token-based access control system that supports recyclable LR-PUF enabled tokens. This builds upon existing PUF-based authentication schemes such as that presented in [38] by introducing the possibility of token recycling while at the same time preserving privacy. The second use case targets a hardware-software binding application where the LR-PUF is used as a secure key storage mechanism. Here the LR-PUF enables keys to be updated while preventing old keys from being reused, for example to prevent downgrading of software to a previous version which may have known security vulnerabilities.

An architecture which in principle can support both cases is shown in Figure 2. This differs from the generic LR-PUF construction of Figure 1 in that the input transformation function $mapin_S()$ is not used. A hash function is used for the output function $mapout_S()$. Since all known silicon PUF implementations exhibit noisy outputs, a fuzzy extractor (FE) or error correction function is required on the raw PUF response *r'*. The fuzzy extractor uses the helper data *W* to reconstruct a noise free PUF response *id* from *r'*. A state update function updates the state *S* in response to the reconfiguration command rcnf(). The fuzzy extractor and helper data *W* are assumed to be public. Further, the hash function and state update functions are assumed to be publicly known.

For the recyclable access token use case this architecture poses some challenges with regard to the resource requirements of the fuzzy extractor and helper data storage. In practice, a more lightweight error correction scheme at the token side is required and as of the date of this document work is ongoing in UNIQUE Work Package 2 to develop a suitable solution. With this in mind the following security analysis will be based on the architecture depicted in Figure 2 for both use cases.

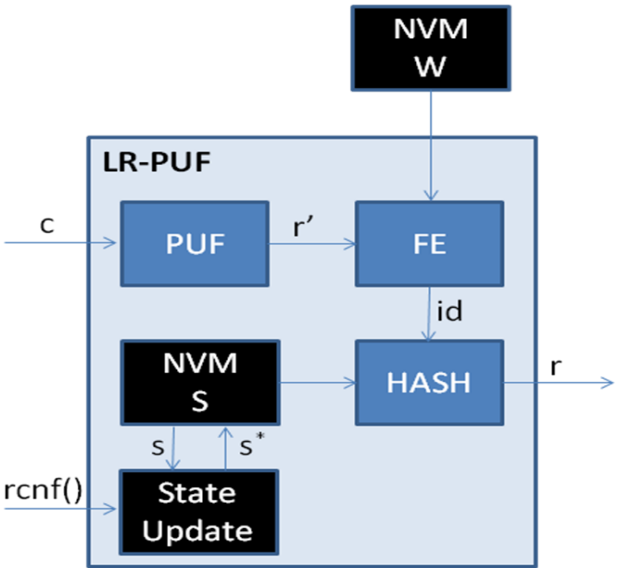

Figure 2 LR-PUF architecture

We distinguish between two attacks on the LR-PUF architecture of Figure 2. The first is an LR-PUF cloning attack, where the behaviour of an existing LR-PUF is replicated in a separate instance. Since the state *S*, hash function, fuzzy extractor and helper data *W* are assumed to be publicly known, such an attack requires that the underlying PUF is cloned and that the remaining LR-PUF functionality is replicated in the new LR-PUF instance. The attack tree[4] for a cloning attack is shown in Figure 3.

Note that the feasibility of a cloning attack is dependent on the system embedding the LR-PUF. In the recyclable access token use case, the LR-PUF is not intended to be embedded within a higher-level system. In this context, the cloning attack is contingent only on a successful execution of the attack tree. The HW-SW binding use case in contrast requires that the LR-PUF is embedded within a higher-level system. The feasibility of the cloning attack thus depends not only on the LR-PUF attack tree, but also on the properties of the higher-level system.

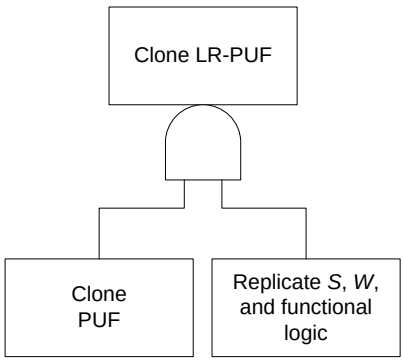

Figure 3 LR-PUF cloning attack

[4] Attack trees describe the possible attack paths for a given attack. Subpaths of the overall tree may be combined with the logical operators AND and OR.

The second attack is a state modification attack where a previously used LR-PUF state *S* is written to the NVM state storage of a pre-existing LR-PUF instance. The aim of such an attack might be to impersonate a previous user in the recyclable token use case, or to enable a software downgrade attack in the HW-SW binding use case. An attack tree for the state modification attack is shown in Figure 4. The state *S* can be modified in one of two ways: First, circuitry which is external to the NVM memory cell array can be subverted. The other is an attack directly on the cell array.

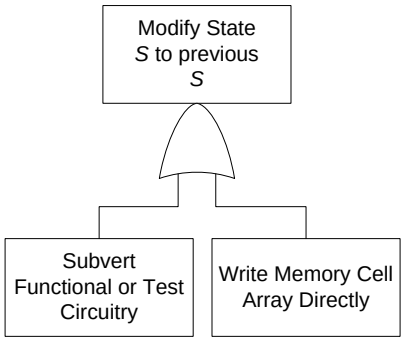

Figure 4 State modification attack

4.2.1 PUF Cloning Attacks

The resistance of the PUF primitive to cloning attacks underpins the security of the LR-PUF construction. We will review the main PUF properties here. There have been a number of attempts in the literature to formalise these properties in terms of a formal PUF security definition with varying degrees of success. In some cases PUF properties have been simply assumed, for example the assumption that tampering with the PUF significantly changes its challenge/response behaviour. Recently, a promising security definition for PUFs has been provided in [1], focusing on three PUF properties: Robustness, unclonability and unpredictability. The first of these, robustness, is a prerequisite for any stable and efficient PUF-based system and will not be considered further here.

The second, unclonability is arguably the most important property that PUFs bring to the table and one that cannot be achieved using traditional cryptographic techniques. We distinguish between two types of unclonability: *Physical* unclonability and *mathematical* unclonability. A PUF is physically unclonable if a physical copy of the PUF with similar[5] challenge/response behaviour cannot be made, even by the manufacturer. In practice this property holds for all known silicon PUFs. A PUF is mathematically unclonable if it is not possible to construct a mathematical procedure which models the original PUF behaviour up to some small error. None of the known silicon PUFs is mathematically unclonable. For example, memory based PUFs such as those based on SRAM can be cloned by exhaustive readout of the post power-up data,

[5] Note that due to the noisiness of PUF responses, it is not even straightforward to define what is meant by two PUFs having a "similar challenge/response" behaviour.

while all variants of delay-based PUFs such as the arbiter and ring oscillator PUF have been shown to be vulnerable to model building attacks [32], which lead to mathematical clones. It should be noted that [1] restricts its notion of unclonability to *physical* unclonability, excluding *mathematical* unclonability from the PUF security definition. In this security evaluation we must include the possibility of mathematical clones.

The third property, unpredictability, refers to the difficulty of predicting the response to a random challenge from previously observed challenge/response pairs. Delay-based PUFs [11], such as the arbiter PUF, become increasingly easier to predict as an adversary learns more challenge/response pairs (CRPs). Memory-based PUFs [12], which effectively possess only a single CRP, can be considered unpredictable since predicting the power-up state of each memory storage element would require a knowledge of the physical implementation to a level of detail which is infeasible in practice.

	Physically Unclonable	*Mathematically Unclonable*	*Unpredictable*
Memory-based PUFs	Yes	No	Yes
Delay-based PUFs	Yes	No	No

Table 1 Properties of interest for the two silicon PUF groupings

For the two main silicon PUF groupings, Table 1 shows the properties of interest. Both groupings show a vulnerability to mathematical cloning attacks, as a result these must be considered valid attack paths in any security analysis. The attack tree for a PUF cloning attack is shown in Figure 5.

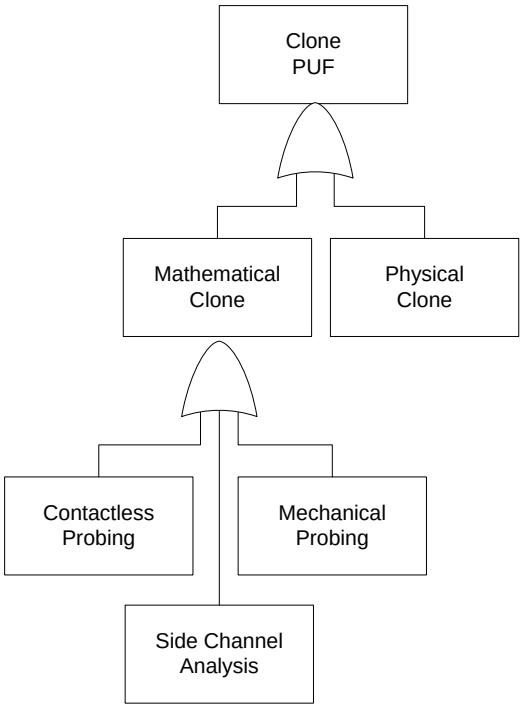

Figure 5 PUF cloning attack

The attack tree comprises four paths leading to a PUF cloning attack. Creating a physical clone of the PUF is considered infeasible since this would imply a level of control over the manufacturing process which is not achievable in any practical sense. We will now consider the paths leading to a mathematical clone. The creation of a mathematical clone requires that the raw PUF response(s) *r'* or alternatively that the corrected PUF response(s) *id* are captured.

Non-invasive attack methods using side channel analysis on the PUF or fuzzy extractor are described in Section 2.2. Research in this area is at an early stage and it remains to be seen whether the proposed attacks are feasible in practice. The operating context of the LR-PUF is important here, a side channel attack against a stand-alone recyclable access token will be more feasible than an attack on an LR-PUF embedded within a large System-on-Chip.

A prerequisite to an invasive attack involving mechanical probing of *r'* or *id* would be to assess whether the available probe bandwidth is sufficient to capture the signals of interest. The attack would proceed by identifying the *r'* or *id* signal lines and connecting them to probe pads. Both the probe pads and connection to the bus lines are achieved using FIB mill and metal deposition operations. The number of required probe pads might range from one for an attack on *r'* of a serial arbiter PUF implementation to perhaps 128 for an attack on *id*. Mechanical probing setups typically have 8 or fewer probes necessitating that requiring an attack to be repeated to build up the complete measurement set.

Attackers with access to contactless probing equipment can use a semi-invasive methodology to obtain the data of interest. As an example TRE might be used to capture the raw PUF responses *r'* of an arbiter PUF while challenges are applied

repeatedly. In this way sufficient CRPs can be collected to mount a PUF modelling attack. In effect this is an optical side channel attack on the PUF. In particular the recyclable access token as defined in [43] does not specify at what rate challenges can be applied to the token simplifying this attack.

4.2.2 State Modification Attacks

The state storage requirement of the LR-PUF construction will be met by a non-volatile memory (NVM) technology which is multiple-time programmable (MTP) in order to support reconfiguration. It is likely that the NVM implementation will be a floating-gate type memory of which EEPROM and flash are typical examples. These technologies require extra wafer processing steps during fabrication which increase cost. As an alternative, NVM technologies which can be implemented in standard CMOS processes [14] have become available; these are collectively known as 'logic-NVM' in the industry and achieve this flexibility at the expense of cell density. The relatively small state storage requirements of the LR-PUF make these technologies a good fit.

Floating-gate memories encode data by the presence or absence of charge in an electrically isolated polysilicon gate. From a security perspective this has advantages, since there is no physical change to the device that can be detected to determine the presence of encoded data.
Failure analysis methodologies to examine floating gate memories at the cell level exist, for example using Scanning Probe Microscopy [27], [6] or voltage contrast [13]. These methodologies are intended to read or locate cell failures and are destructive, requiring frontside or backside die de-processing down to the floating gate level. Writing at the cell level in order to write arbitrary data is unlikely for this reason.

The semi-invasive attacks on floating gate memories introduced in [37] are of interest since they include optical modification attacks on EEPROM and flash memory cells with low equipment costs. However, it must be noted that these were demonstrated on isolated memory cells such as security fuses where the optical resolution of the equipment was not a factor. Using these techniques to write a memory cell array to an arbitrary value in a modern deep-submicron technology is likely to be extremely difficult.

An adversary wishing to read or write a floating gate memory using invasive means might attempt to subvert existing logic, for example the read/write control circuitry or test structures such as BIST (Built-In Self Test) and scan circuitry. The attack tree for writing NVM based on floating gate technology is shown in Figure 6.

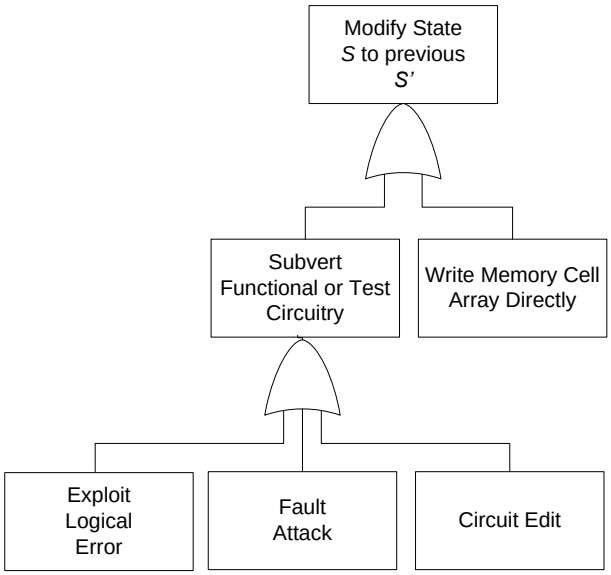

Figure 6 State modification attack tree

The attack tree comprises four paths leading to a state modification attack. Writing the memory cell array directly is not considered a viable attack method for the reasons discussed above. A non-invasive attacker might attempt to exploit a logical error, for example a protocol flaw in order to modify the state. Fault attacks on the functional or test circuitry are conceivable but challenging since a specific state *S* must be written. Finally, circuit editing techniques might be used to modify existing circuitry to achieve the required aim. For example a minor circuit edit to the scan circuitry used for production test in combination with the application of suitable vectors on the scan test interface might be sufficient to force the required write.

4.3 Mitigations and Design Rules

Any implementation of the LR-PUF construct must be cognizant of the attack paths detailed above and implement mitigations if appropriate. A non-exhaustive list of mitigations is presented in the following.

4.3.1 Non-invasive Attacks

Logical attacks can be mitigated to some extent by employing formal design methodologies although these should not be considered a panacea. Closing the verification gap between high-level design specifications and the implementation can help. Formal analysis tools are gaining momentum in the IC design community, where tools employing formal methods [39] are employed to evaluate the equivalence of hardware representations at differing levels of abstraction. The possibility of moving seamlessly from a formal security algorithm or protocol specification to a hardware friendly design representation is attractive and could be based on a similar approach. Nevertheless, some level of assurance that the security algorithm or protocol specification itself is correct is required, which can only be realistically provided through an extensive peer-review process.

Non-invasive fault attacks can be mitigated by taking steps to ensure that injected single-bit errors cannot force undesired behaviour. Environmental monitors on power supplies, temperature and clock inputs should disable the device when an excursion is detected such as an introduced clock glitch.

Side-channel attacks can be mitigated by removing or minimizing data dependent behaviour. For power and electro-magnetic side channel attacks implementing critical logic in an asynchronous dual-rail logic style [31] can eliminate data dependent power transients (and as a consequence electro-magnetic emanations).

4.3.2 Semi-invasive Attacks

Mitigating against contactless probing methods via the die backside is a difficult proposition. Since heavily doped silicon is highly absorptive in the near-IR wavelengths [7], substrate doping has been proposed as a mitigation for optical non-contact probing. The effectiveness of this approach is questionable since mechanical thinning of the substrate will restore optical transmission to adequate levels.

4.3.3 Invasive Attacks

Steps should be taken to increase the cost and complexity of a mechanical probing attack. Critical signals should be routed on the lowest metal layers in order to prevent easy access from the front side. Parallel data paths are preferred over serial since an attacker must implement many probe pads in order to monitor a bus. Circuits should be designed in such a way that a single FIB edit does not result in undesired behaviour.

4.3.4 General Mitigations

Invasive and semi-invasive attacks require knowledge of the target circuit. The attacker's task is made more difficult if the features of interest do not exhibit a regular structure. This can be achieved by employing flat rather than hierarchical layout techniques (also termed 'glue logic' in the smart card industry). Some features will enforce a regular structure, for example memories such as SRAM and most PUF variants.

Careful attention should be given to the test circuitry on the device since, e.g., scan chains and BIST can be subverted to mount attacks. Attacks combining minor circuit edits with test circuitry have been demonstrated and shown to be effective [41]. Removing test capability from critical circuits is a possibility but runs counter to the requirements of manufacturing testing which aims for maximum test coverage.

Finally, the technology node itself can be considered a mitigation against invasive and semi-invasive attacks. Smaller feature sizes and a higher number of metal layers require the attacker to use more sophisticated and costly equipment and may for a time make an attack uneconomic.

5 Common Criteria Evaluation Gaps

The Common Criteria [4] is an international standard for security evaluation and has been successfully used during the last 15 years to evaluate a wide range of IT security products. The CC enables comparisons between the results of independent security evaluations. It achieves this by providing a common set of requirements for the security functionality of IT products and for assurance measures applied to these IT products during a security evaluation. These IT products may be implemented in hardware, firmware or software. The CC community is organized around working groups focused on particular application areas that promote and share common interpretations of the standard. For instance, the smart card community has developed supporting documentation to harmonize the evaluation process across different evaluation laboratories.

The application of a 'CC evaluation to a new type of product and/or technology will necessitate that the main actors share their understanding and interpretation of the CC in order to ensure a common acceptance of the results of the evaluation across all participating countries (through CCRA[6] recognition). PUF technology will not be an exception to the above mentioned rules. If we want to be able to ensure the security level of PUF-based products using CC evaluation and moreover be sure that the results will be recognised all over the world, it will be necessary to ensure that people are talking the same language.

In the previous chapters we presented PUF-specific evaluation methods and use case security analysis. This chapter will highlight CC which improvements will be necessary to ensure a correct evaluation of the PUF's characteristics.

5.1 Security Functional Requirements

The first question to be solved by the CC community is "how can we specify PUFs?". Indeed CC part 2 defines Security Functional Components (SFCs). These SFCs are the basis for the Security Functional Requirements (SFRs) expressed in a Protection Profile (PP) or a Security Target (ST). These requirements describe the desired security behaviour expected of a Target of Evaluation (TOE) and are intended to meet the security objectives as stated in a PP or an ST. These requirements describe security properties that users can detect by direct interaction (i.e. inputs, outputs) with the TOE or by the TOE's response to stimulus.

Security functional components express security requirements intended to counter threats in the assumed operating environment of the TOE and/or cover any identified organisational security policies and assumptions.

The main security aspect addressed by PUF technology is the uniqueness of the challenge-response pairs. This intrinsic characteristic enables the complete identification of each individual TOE sample. Compared to common identification technology (such as the one used in smart card products) the identification of

[6] CCRA Common Criteria Recognition Arrangement. An agreement signed by member countries stating the conditions for participation in the Common Criteria.

the TOE is not based on an asset stored inside the TOE during the manufacturing phase (e.g., an identifier) but on intrinsic characteristics of TOE. The PUF can be seen as a kind of fingerprint of the product. It is therefore necessary to specify this through a new set of Security Functional Requirements.
Here we deal with the main limitation of the CC part 2 requirements. Indeed, even if the CC part 2 catalogue were to contain a class dealing with identification and authentication mechanisms, these families (from the FIA[7] class) focus on users and not on the TOE itself. Here the security problem highlighted in the previous chapter demonstrates that we need to have a way in order to identify genuine TOEs from potential counterfeit ones. This specific functionality could be specified in an extension to the Common Criteria part 2 Security Functional Requirements.

5.2 Security Assurance Requirements

No evaluation gap has been identified in the Security Assurance Requirements.
Nevertheless at least for the AVA[8] and ATE[9] tasks, some common agreement must be stated.
Regarding ATE activity, it will be useful to specify which kind of tests must be applied in order to validate the expected PUFs' properties (robustness, unclonability and unpredictability).

Regarding the vulnerability assessment class AVA, Section 2 of this document lists PUF-specific evaluation methods. These methods correspond to some intrinsic PUF vulnerabilities that must be taken into account by the evaluator for any vulnerability analysis.
Common interpretation / ranking methods must also be discussed in order to unambiguously state the resistance of TOE against these specific attacks.

We think that most of the attack methods identified for smart card products or similar devices will probably also apply to PUF-based TOEs; therefore attack scenarios could be assessed according to the Joint Interpretation Library (JIL) method for the AVA_VAN (vulnerability analysis) category.

According to the JIL, two phases are distinguished: "identification" to set up and define the attack then "exploitation" for the repetition of the attack. The attack potential score is obtained by summing up five factors for both identification and exploitation phases (simplified table, refer to [5] for all details):

[7] FIA : Identification and authentication. Families in this class address the requirements for functions to establish and verify a claimed user identity.
[8] AVA : Vulnerability Assessment : the purpose of the vulnerability assessment activity is to determine the exploitability of flows or weaknesses in the TOE in the operational environment.
[9] ATE : Tests : The goal of this activity is to determine whether the TOE behaves as described in the Security Target and as specified in the evaluation evidences (described in the ADV class)

	Identification	Exploitation
Elapsed time		
< one hour	0	0
< one day	1	3
< one week	2	4
< one month	3	5
> one month	5	8
not practical	*	*
Expertise		
Layman	0	0
Proficient	2	2
Expert	5	4
Multiple Expert	7	6
Knowledge of the TOE		
Public	0	0
Restricted	2	2
Sensitive	4	3
Critical	6	5
Very critical hardware design	9	NA
Access to TOE		
< 10 samples	0	0
< 100 samples	2	4
> 100 samples	3	6
Not practical	*	*
Equipment		
None	0	0
Standard	1	2
Specialized	3	4
Bespoke	5	6
Multiple Bespoke	7	8

Depending upon the desired level of security, the desired AVA_VAN level implies that the TOE be resistant at a level higher than each attack scenario minimal score, according to:

AVA_VAN level:	Required for:	TOE resistant to attackers with a attack potential of:
AVA_VAN.2	EAL2, EAL3	16-20
AVA_VAN.3	EAL4	21-24
AVA_VAN.4	**EAL5**	**25-30**
AVA_VAN.5	EAL4+, EAL5+, EAL6, EAL7	31 and above.

In UNIQUE, we target EAL5; therefore our reference is AVA_VAN.4, which means an attack potential higher than 24 for both identification and exploitation phases.

As an illustration, a possible rating[10] of the attack potential for the "extraction of secrets by electromagnetic analysis" attack scenario would be:

	Identification		**Exploitation**	
Elapsed time	< one month	3	< one month	5
Expertise	Expert	5	Expert	4
Knowledge of the TOE	Public (based on commercial parts available on the market)	0	Public (based on commercial parts available on the market)	0
Access to TOE	< 10 samples	0	< 10 samples	0
Equipment	Specialized (advanced university lab)	3	Specialized (advanced university lab)	4
Intermediate score		11		13
Total score	**24**			

[10] This rating is for illustration only and does not anticipate evaluations to be performed in task T3.5.

6 Securing the ICT Supply Chain

Today's pervasive computing environments put a new premium on security assurance. Security assurance can be viewed as a metric that provides a confidence level that an implemented product will perform as intended, even in the presence of adversaries.
Determining the assurance level of an ICT product is a challenging task. Security solutions are composed of many individual protections, which are often interdependent and exist at differing abstraction levels, spanning silicon, algorithms, protocols and systems as well as including non-technological aspects such as operational policies and economics. In addition the complexities of today's globalised, segmented and specialised supply chains add a new dimension to the assurance problem.

The basic ICT supply chain is shown in Figure 7. In many cases products are designed with some third party IP which can be hardware or software. The products are typically manufactured offshore with inputs from a variety of suppliers and brokers which may also be geographically dispersed. Products enter the distribution network and are delivered to the end-user. The product then enters the support phase which may involve product updates that originate from yet another supplier. This presents many opportunities for attack: third party IP may introduce malware or vulnerabilities into the product during the design phase, counterfeit or substandard components may be included during manufacture and malware might be introduced by updates during product support. The distribution network may be complex involving a hierarchy of distributors into which cloned, counterfeit or otherwise subverted product might be inserted. A number of high profile incidents highlighting the importance of supply chain security have been reported. In 2008 the New York Times reported that 3,500 counterfeit Cisco network routers were intercepted by the F.B.I, bought in part by U.S. military agencies, military contractors and electric power companies [40]. In another incident involving the U.S. Air Force, microprocessors for its F-15 flight control computer were procured from a broker and were found to be counterfeit [28].

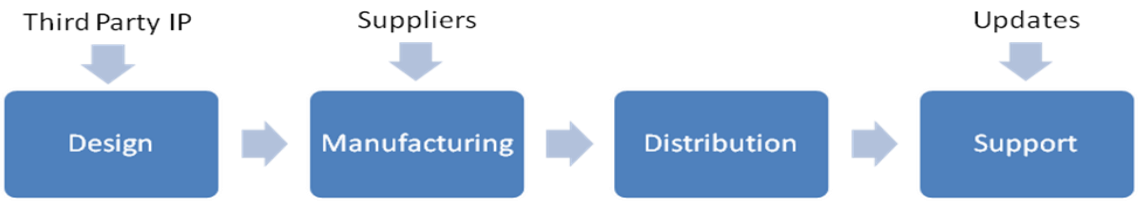

Figure 7 ICT Supply Chain

Supply chain integrity is a key factor influencing the overall assurance level for a product.
Product security evaluation methodologies such as the Federal Information Processing Standard 140-2 [8] evaluate whether products meet a set of defined security requirements or in the case of the Common Criteria compare product performance against a set of claimed capabilities to a prescribed assurance level. FIPS140-2 focuses on security requirements for cryptographic modules and is as

a result quite specialised. Supply chain integrity is not a FIPS140-2 requirement. The Common Criteria on the other hand recognises that any assurance measure for a product must hold up through the distribution and support phases of the product life-cycle. The assurance class *Class ALC: Life-cycle support* [3] consists of a number of families concerned with the procedures used for the delivery of the product to the consumer (ALC_DEL: Delivery) and for flaw remediation (ALC_FLR: Flaw Remediation), a fact that testifies to the flexible nature and broad scope of the Common Criteria framework. However, the guidelines given are too generic to form the basis of a set of best practices and approaches. The supply chain security risks associated with manufacturing are not considered in the framework.

Mitigating the risks of supply chain compromises will require a combination of evaluation methodologies, policies, incentives and novel technological solutions. The need for supply chain standards, best practices and approaches that can accommodate fast paced innovation, diversely sourced components and a globalised marketplace is recognised. Efforts are underway in the International Standards Organisation to develop ISO 27036, a multi-part standard offering guidance on the evaluation and mitigation of security risks associated with supplier relationships, with one part targeting ICT supply chain risk management.

Technologies that can complement and improve existing processes and practices are required. Physically Unclonable Functions (PUFs) have generated a lot of interest in the security community as a promising new security primitive. There is no single technology solution to improving the security of the supply chain, nevertheless technologies that can detect compromises as early as possible and which can be integrated into existing business processes at a reasonable cost are desirable. In the next sections we will explore how PUFs offer opportunities for increasing supply chain assurance in two areas, semiconductor anti-counterfeiting and traceability.

6.1 Semiconductor Anti-Counterfeiting

Complex semiconductors provide much of the enabling technology for the diverse range of electronic equipment in evidence today in the consumer and professional markets. Semiconductor counterfeiting not only affects the business revenues of the companies being targeted but has negative implications for critical infrastructure and public safety. An increase in customs and law enforcement seizures and reported customer complaints[11] reveal that the problem is a growing one.

Remarked devices account for the bulk of counterfeits detected [36]. In a typical remarking attack a device's product markings are misrepresented by replacing the original markings with markings indicating a higher specification and hence more valuable part. Such a device, if embedded in an electronic system, may fail in the field when subjected to a different operational environment than the original part was designed for. The risk of counterfeit product entering the

[11] Joint Statement of the 14th Meeting of the World Semiconductor Council (WSC), Seoul, May 27th, 2010

supply chain increases when devices suffer supply shortfalls or have production terminated by the manufacturer. In extreme cases, where the product life cycle includes a lengthy validation and certification phase, devices may be obsolete by the time the product enters the field. In such cases purchasing managers may be forced to procure devices from non-certified sources.

Current practice for detecting counterfeit semiconductors includes visual checking, electrical testing, and reliability testing which can require significant investments in expertise, equipment, and time. Additionally, best practices have been developed in industry worldwide to combat counterfeiting in many of its variants. Although the current approaches improve the situation significantly, they do not provide extensive technical means to detect counterfeiting. Such methods cannot guarantee the provenance or performance of a device and in many cases it may only be feasible to perform testing on a sample of devices, for example when tests are destructive. However, new approaches in this area are beginning to emerge. Since the introduction of SEMI T20-1109 in 2009 [35] standardised methods providing device traceability and authentication have been defined, however these are serialisation mechanisms based on the generation of unpredictable, random codes and are intended to be applied at the device package and higher levels of packaging.
Authentication mechanisms which operate at the silicon rather than packaging level are an attractive proposition, particularly if they utilise intrinsic characteristics of the silicon rather than a serialisation mechanism. Physically Unclonable Functions (PUFs) enable a class of applications in which identifiers are inseparably bound to hardware instances. An overview of known PUF constructions is contained in UNIQUE Deliverable D2.1 [44].

The success of an anti-counterfeiting scheme will be dependent on a number of factors. It must be generic enough to apply to semiconductor products from a wide range of manufacturers, it should use existing production infrastructure where possible, the impact on production cycle time should be negligible and its cost should not be prohibitive. Most importantly the technology must raise the cost of counterfeiting to uneconomic levels. This last point is the motivation for using PUFs in this context. Although PUFs cannot be considered unclonable for the reasons which were discussed in Section 4.2.1, the cost of a cloning attack is likely to be prohibitive for a typical device remarker.

A general usage model for a PUF based anti-counterfeiting technology would involve the semiconductor manufacturer integrating a PUF into the device and registering the PUFs unique identifier bound together with other device information in a publically available on-line database. An equipment manufacturer would query the device and the database at some point during production and verify that the device's specification is as expected. In this way tampering with the device package is mitigated. More sophisticated attacks such as a PUF cloning attack while not technically impossible are mitigated for economic reasons since they imply a high-end reverse engineering and fabrication capability. Practical concerns with the above usage model such how to query the PUF in a production environment can be addressed with existing standards such as the JTAG Test Access Port [15].

6.2 Traceability

The ability to trace the path components and products take through the supply chain is a key enabler of supply chain assurance. It is for example of benefit to an electronic equipment manufacturer to have visibility into the history of a component in terms of which distributors or brokers it has passed through. A component with an incomplete history or other irregularity can be rejected. Radio Frequency Identification (RFID) tags allow an item to be uniquely identified by radio communication between the tag and a reader and are small enough to be to be embedded into higher level component packaging or perhaps the component itself. Note that the PUF usage model described in the previous section is not suitable here since it requires that the semiconductor device embedding the PUF is powered during querying, something that is only practical when it is has been assembled into the final product.

RFID tags can be classified as active and passive. Active tags have an on-board power source in the form of a battery. Passive tags have no on-board power source and derive their power from the incident radio signal of the interrogating reader. As a result passive tags are typically computationally and memory constrained and possess at most a few thousand gates devoted to basic operations. Security functionality if available at all is likely to be restricted to symmetric cryptography such as a hashing unit. The lack of security functionality poses challenging security and privacy risks [46],[16]. For example a simple cloning attack may involve simply copying one tag's contents to another.

Any deployment of RFID to facilitate traceability in the supply chain is likely to be of the passive type for cost reasons and for it to be credible mitigations to the security shortcomings must be found. One approach to the problem is to use the security properties of PUFs to prevent cloning attacks by implementing PUF-based authentication and identification schemes which are achievable within the power and area budgets of a typical passive RFID tag. Several approaches have been proposed [30], [42], [33] and one commercially available PUF-enabled RFID tag is available on the market [45].

7 UNIQUE Security Evaluation Test Plan

The previous chapters present novel methodologies that are very specific to PUF features. Here, we present "classical" tests that will be performed on the PUFs that are designed within UNIQUE.
Although these "classical" tests are not fully in the scope of this document, we feel it is valuable to present them at this stage as this provides information about the kind of result demonstration we are planning at the end of the project.

We stress that the word "classical" does not mean that these tests are easy to drive – it rather means that these tests have already been used for other kinds of ICs and can provide meaningful assessment of PUF security properties.
As five different PUF technologies are prototyped in the same ASIC, these tests are expected to provide valuable comparisons between the different solutions. The tests listed below have been selected based on the expected relevancy for the designed PUFs.

7.1 PUF Characterisation

A first category of necessary tests will focus on the main expected characteristic of the PUFs: The unique and reproducible signature generation. They are mainly linked to the PUF qualification but can also have impact on the security of the PUF[12]. Indeed the ASIC will enable behavioural comparisons of most of known silicon PUFs based on the same process implementation.

Intrinsic ID (IID) has built up expertise in characterising PUF technology, so IID will take lead on the PUF characterisation. The main PUF qualification tests are presented in Table 2.
In order to be relevant, these tests must be carried out on a large number of samples to obtain relevant statistical characteristics.
In addition to the classical tests described above, it will be interesting to enlarge the excursion of the environmental conditions out of the normal expected conditions for the technology.

Remarks about ageing test:

- For ageing tests, some endurance tests on memories (such as repetitive writing of same value to the same address) could also be interesting in order to measure their impact on the PUF response. Indeed this kind of feature is known to have deep impact on memories (even with some cases of failure) and may thus modify the post-power up data.
- We are planning to perform ageing tests on all PUF types included in the UNIQUE ASIC. However, it is not yet clear whether we will have sufficient time within the project to run these tests. Furthermore, ageing models are not well defined for each PUF type. Due to these two reasons it is possible that the outcome of our ageing tests is only preliminary.

[12] If a PUF is not reliable, the device containing the PUF will not work properly. For instance, in the case of a user's VPN device being out of order, (s)he will presumably revert to clear text transmission which is a security concern.

Repeated Startup Test	Repeatedly measure PUF responses at room temperature to evaluate noise between measurements
Temperature Cycle Test	Measure PUF responses at different ambient temperatures
Voltage Variation Test	Measure PUF responses at different core voltages
Voltage Ramp Up Test	For memory based PUFs: Measure startup values when the memory is powered up with different power-up times (ramps)
Voltage Dip Test	For memory based PUFs: Measure startup values when the memory is subjected to short power dips of varying lengths
Data Retention Voltage Test	For memory based PUFs: Store a pattern of ones (0xFF bytes) into the memory, temporarily lower the core voltage and then measure the PUF response at the normal voltage level
Ageing Test	Measure PUF responses on a weekly basis on ICs that are kept at high temperature and increased voltage for a long period

Table 2 PUF qualification tests

7.2 Penetration tests

Penetration tests can be performed to assess the vulnerability and resistance of PUFs to non-invasive, semi-invasive and fully invasive attacks. These tests are standard when evaluating secure ICs, such as smart cards.

7.2.1 PUF tamper evidence

Does the addition of a front-side (or back-side) probe to a PUF circuit node change the PUF response in a way that is detectable? The PUF literature makes a lot of statements about PUFs being tamper evident, it would be valuable to get some quantitative data.

Indeed, even though direct physical measurements of the PUF itself will likely modify its behaviour (for example mechanical probing of the delay circuit nodes in the arbiter PUF), in most cases, direct measurement of the PUF circuit itself is not required. For example on ring oscillator PUFs, the oscillating signal will be buffered before being propagated to a counter. It therefore is possible to pick the output signal of the ring oscillator at the buffer or counter without modifying its frequency.

We aim to identify for some of the implemented PUFs which kind of signal can be probed with lower risk of modifying the PUF and try to measure this signal on the ASIC. We will target one or two representative signals to be probed through front-side or back-side probing. This will require access to the layout of the ASIC and at least four samples (plastic package or raw die) to get access to front-side or back-side. An additional requirement will be the capability to put the test

board (or at least an extender board) on the probe station. Some issues are still to be addressed within WP4 to ensure that the boards are compatible with the test equipment.

These tests will be managed by Thales.

7.2.2 Semi-invasive attacks

Can the system be compromised by inducing faults into the PUF? Fault injection techniques are widely used during security evaluation. They have been shown to be very efficient at retrieving secret keys and/or bypassing security mechanisms such as conditional access.

For PUF-based security functions two aspects can be assessed:

1. Retrieving the start up values of a given memory-based PUF: Fault injection can permit setting or resetting of memory nodes. This technique combined with safe error techniques can allow the retrieval of the initial value of a PUF if an attacker is able to independently set or reset each memory cell involved in the PUF.
   We will challenge some of the memories on the ASIC to check if we can force their values and then investigate how to use this method to retrieve portions of the PUF. The attack path will involve characterization of laser fault injections on SRAM in order to see if we can force cells (or bit reads) to a fixed value. If this attack succeeds, by injecting faults during PUF access and comparing the output with and without fault injection, it could be possible to retrieve the initial values of the memory. If the PUF response is not modified the read value and initial one will be equivalent, so the initial value will be identical to the forced value. If the response changes the value in memory has been modified. The main limitation of the attack is due to the presence of the fuzzy extractor.
   This test requires two ASIC samples with back-side access and the layout database of the ASIC.

2. Ring oscillator-based attacks: Fault injection techniques can be used in order to modify/characterize ring oscillator frequencies and as a result change the normal behaviour of RO-based PUFs. We will investigate the effect of light perturbation and electromagnetic injection techniques on the oscillators to see how these techniques could be used in order to predict/modify the PUF responses.

These tests will be led by Thales.

7.2.3 Side channels analysis

Do PUFs exhibit any interesting signatures? Simply determining the presence of a PUF is potentially useful.

Several PUFs and/or functions can be simultaneously activated in the ASIC. The main goal of this attack will be to assess if it is possible to identify which PUF is activated through side channel techniques.

In UNIQUE, the ASIC is a PUF characterization vehicle i.e. it implements a variety of PUF types but not the required supporting functionality. A full PUF-based system will also require additional functionality (fuzzy extractors) that would possibly ease the detection (see below). In this test, we will only observe the ASIC part in order to search for characteristic signatures that could be used in order to detect the presence/activation of a PUF.
This test will be led by Thales.

7.2.4 Power and EM signature of PUFs

Ring oscillator PUFs may be vulnerable to harmonic analysis, i.e., deducing the operating frequencies of the oscillators (or a subset of the oscillators).

We will try to retrieve which oscillator is activated through the characterization of its signature. This attack could lead to the leakage of the full raw data set of Ring Oscillator PUFs. It is an additional step to the initial characterization presented above.

This test will be performed by Thales.

7.2.5 Fuzzy extractors

These are the key to most PUF-based systems. Are fuzzy extractors vulnerable to side channel analysis? Some of them are computationally expensive so we can expect them to have interesting power signatures during operation.

Fuzzy extractors will not be implemented in the ASIC but in the FPGA. So it will be necessary to plan how to access to power measurement on the FPGA.

The fuzzy extractors are not yet detailed and at this point this is not possible to define the tests that can be applied. This test category will be further investigated in task T3.4.

7.2.6 Access to raw PUF data

What are the consequences of an attacker gaining access to the raw PUF data, e.g., via an invasive attack?

The goal of this test will be to better understand how an attacker can use raw PUF data in order to attack a PUF-based system. The UNQIUE hardware architecture combines a PUF characterization ASIC with a FPGA allowing for convenient access to raw PUF data for this work. Note that in a PUF-based commercial product the PUF and its supporting functionality would be integrated within an IC; access to raw PUF data in this scenario might require an invasive attack, raising the difficulty for the attacker significantly.

8 References

[1] Frederik Armknecht, Roel Maes, Ahmad-Reza Sadeghi, Francois-Xavier Standaert, and Christian Wachsmann. A formal foundation for the security features of physical functions. In *32nd IEEE Symposium on Security and Privacy (IEEE S&P 2011)*. IEEE Computer Society, May 2011.

[2] Xavier Boyen. Reusable cryptographic fuzzy extractors. In *ACM CCS 2004, ACM*, pages 82–91. ACM Press, 2004.

[3] Common Criteria for Information Technology Security Evaluation - Part 3: Security Assurance Components. http://www.commoncriteriaportal.org/-files/ccfiles/CCPART3V3.1R3.pdf.

[4] Common Criteria Portal. http://www.commoncriteriaportal.org/, 2011.

[5] Common Criteria. Application of Attack Potential to Smartcards, . Common Criteria supporting Document, version 2.7, revision 1.

[6] C. DeNardi, R. Desplats, P. Perdu, J-L. Gauffier, and C. GuÃ©rin. Descrambling and data reading techniques for flash-eeprom memories. application to smart cards. *Microelectronics and Reliability*, 46(9-11):1569 – 1574, 2006. Proceedings of the 17th European Symposium on Reliability of Electron Devices, Failure Physics and Analysis. Wuppertal, Germany 3rd-6th October 2006.

[7] R. A. Falk. Near ir absorption in heavily doped silicon - an empirical approach. *(ISTFA) International Symposium for Testing and Failure Analysis*, 2000.

[8] Federal Information Processing Standard 140-2 - Security Requirements for Cryptographic Modules. http://csrc.nist.gov/groups/STM/cmvp/-standards.html#02.

[9] Blaise Gassend, Dwaine Clarke, Marten van Dijk, and Srinivas Devadas. Silicon physical random functions. pages 148–160.

[10] Blaise Gassend, Dwaine Clarke, Marten van Dijk, and Srinivas Devadas. Controlled physical random functions. In *ACSAC '02: Proceedings of the 18th Annual Computer Security Applications Conference*, page 149, Washington, DC, USA, 2002. IEEE Computer Society.

[11] Blaise Gassend, Dwaine Clarke, Marten van Dijk, and Srinivas Devadas. Silicon physical random functions. In *ACM Conference on Computer and Communications Security*, pages 148–160, New York, NY, USA, 2002. ACM Press.

[12] Jorge Guajardo, Sandeep S. Kumar, Geert Jan Schrijen, and Pim Tuyls. FPGA intrinsic PUFs and their use for IP protection. In *Cryptographic Hardware and Embedded Systems Workshop*, volume 4727 of *LNCS*, pages 63–80, September 2007.

[13] Ray Haythornthwaite, Jochonia Nxumalo, and Michael W. Phaneuf. Use of the focused ion beam to locate failure sites within electrically erasable read only memory microcircuits. *Journal of Vacuum Science Technology A: Vacuum, Surfaces, and Films*, 22(3):902 –907, may 2004.

[14] A. Horch, Bin Wang, M. Niset, T.J. Hu, T. Gilliland, and T. Humes. Building true logic lnvm with automotive-level reliability. In *Non-Volatile Memory Technology Symposium, 2008. NVMTS 2008. 9th Annual*, pages 1 –5, nov. 2008.

[15] IEEE. 1149.1-1990 - IEEE Standard Test Access Port and Boundary-Scan Architecture. http://standards.ieee.org/findstds/standard/1149.1-1990.html.

[16] Ari Juels. RFID security and privacy: A research survey. *Journal of Selected Areas in Communication*, 24(2):381–395, February 2006.

[17] Bhavana Kanukurthi and Leonid Reyzin. Key agreement from close secrets over unsecured channels. In *Proceedings of the 28th Annual International Conference on Advances in Cryptology: the Theory and Applications of Cryptographic Techniques*, EUROCRYPT '09, pages 206–223, Berlin, Heidelberg, 2009. Springer-Verlag.

[18] D. Karakoyunlu and B. Sunar. Differential template attacks on puf enabled cryptographic devices. In *Information Forensics and Security (WIFS), 2010 IEEE International Workshop on*, pages 1 –6, dec. 2010.

[19] Paul Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. pages 388–397. Springer-Verlag, 1999.

[20] Paul C. Kocher. Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems. In *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '96, pages 104–113, London, UK, UK, 1996. Springer-Verlag.

[21] J.W. Lee, Daihyun Lim, B. Gassend, G.E. Suh, M. van Dijk, and S. Devadas. A technique to build a secret key in integrated circuits for identification and authentication applications. In *VLSI Circuits, 2004. Digest of Technical Papers. 2004 Symposium on*, pages 176 – 179, june 2004.

[22] W.T. Lee. Engineering a device for electron-beam probing. *Design Test of Computers, IEEE*, 6(3):36 –42, 44–49, jun 1989.

[23] M. Majzoobi, F. Koushanfar, and M. Potkonjak. Lightweight secure pufs. In *Computer-Aided Design, 2008. ICCAD 2008. IEEE/ACM International Conference on*, pages 670 –673, nov. 2008.

[24] Mehrdad Majzoobi, Farinaz Koushanfar, and Miodrag Potkonjak. Techniques for design and implementation of secure reconfigurable pufs. *ACM Trans. Reconfigurable Technol. Syst.*, 2:5:1–5:33, March 2009.

[25] M. K. Mc Manus, J. A. Kash, S. E. Steen, S. Polonsky, J. C. Tsang, D. R. Knebel, and W. Huott. Pica: Backside failure analysis of cmos circuits using picosecond imaging circuit analysis. *Microelectronics Reliability*, 40(8-10):1353 – 1358, 2000. Reliability of Electron Devices, Failure Physics and Analysis.

[26] Dominik Merli, Dieter Schuster, Frederic Stumpf, and Georg Sigl. Side-channel analysis of pufs and fuzzy extractors. In *4th International Conference on Trust and Trustworthy Computing (Trust 2011)*, Lecture Notes in Computer Science, Pittsburgh, PA USA, June 2011. Springer-Verlag. accepted for publication.

[27] Christophe De Nardi, Romain Desplats, Phillippe Perdu, Felix Beaudoin, and Jean Luc Gauffier. Eeprom failure analysis methodology: Can programmed charges be measured directly by electrical techniques of scanning probe microscopy? In *31st International Symposium for Testing and Failure Analysis*, pages 256–261, 2005.

[28] United States Govenment Accountability Office. Defense supplier base: Dod should leverage ongoing initiatives in developing its program to mitigate risk of counterfeit parts. GAO-10-389, March 2010.

[29] Jean-Jacques Quisquater and David Samyde. Electromagnetic analysis (ema): Measures and counter-measures for smart cards. In *Proceedings of the International Conference on Research in Smart Cards: Smart Card Programming and Security*, E-SMART '01, pages 200–210, London, UK, UK, 2001. Springer-Verlag.

[30] Damith C. Ranasinghe, Daniel W. Engels, and Peter H. Cole. Security and privacy: Modest proposals for low-cost rfid systems. Auto-ID Labs Research Workshop, September 2004.
[31] Simon Moore Ross, Simon Moore, Ross Anderson, Paul Cunningham, Robert Mullins, and George Taylor. Improving smart card security using self-timed circuits. In *Technologyâ€□, Fourth AciD-WG Workshop, Grenoble, ISBN*, pages 211–218, 2002.
[32] Ulrich Rührmair, Frank Sehnke, Jan Sölter, Gideon Dror, Srinivas Devadas, and J "urgen Schmidhuber. Modeling attacks on physical unclonable functions. In *CCS 2010: Proceedings of the 17th ACM conference on Computer and communications security*, pages 237–249, New York, NY, USA, 2010. ACM.
[33] Ahmad-Reza Sadeghi, Ivan Visconti, and Christian Wachsmann. Puf-enhanced rfid security and privacy. 2nd Workshop on Secure Component and System Identification (SECSI'10), April 26–27 2010.
[34] R. Schlangen, R. Leihkauf, U. Kerst, C. Boit, R. Jain, T. Malik, K. Wilsher, T. Lundquist, and B. Kruger. Backside e-beam probing on nano scale devices. In *Test Conference, 2007. ITC 2007. IEEE International*, pages 1 – 9, oct. 2007.
[35] SEMI. Semi t20-1109 specification for authentication of semiconductors and related products, 2009. http://www.semi.org/.
[36] Semiconductor Industry Association. http://www.sia-online.org/cs/-anticounterfeiting.
[37] Sergei Skorobogatov. Semi-invasive attacks — a new approach to hardware security analysis. Technical Report UCAM-CL-TR-630, University of Cambridge, 15 JJ Thomson Avenue, Cambridge CB03 0FD, UK, April 2005.
[38] G. Edward Suh and Srinivas Devadas. Physical unclonable functions for device authentication and secret key generation. In *Design Automation Conference*, pages 9–14, New York, NY, USA, 2007. ACM Press.
[39] Synopsys formal equivalence checker. http://www.synopsys.com/Tools/-Verification/FormalEquivalence/Pages/default.aspx.
[40] New York Times. F.b.i. says the military had bogus computer gear. http://www.nytimes.com/2008/05/09/technology/09cisco.html, May 2008.
[41] Randy Torrance and Dick James. The state-of-the-art in ic reverse engineering. In *Proceedings of the 11th International Workshop on Cryptographic Hardware and Embedded Systems*, CHES '09, pages 363–381, Berlin, Heidelberg, 2009. Springer-Verlag.
[42] Pim Tuyls and Lejla Batina. RFID-tags for anti-counterfeiting. In *The Cryptographers' Track at the RSA Conference 2006, San Jose, CA, USA, February 13–17, 2005, Proceedings*, volume 3860 of *Lecture Notes on Computer Science (LNCS)*, pages 115–131. Springer Verlag, 2006.
[43] UNIQUE project. Deliverable D1.2: Security architectures, protocol design and evaluation principles for anti-counterfeiting/anti-tampering solutions, 2010.
[44] UNIQUE project. Deliverable D2.1: Description and design of intrinsic hardware security, crypto and hardware entangled crypto components, 2010.
[45] Verayo, Inc. Verayo website — product page. http://www.verayo.com/-product/products.html, February 2011.
[46] Stephen A. Weis, Sanjay E. Sarma, Ronald L. Rivest, and Daniel W. Engels. Security and privacy aspects of low-cost radio frequency

identification systems. In *Proc. of PerCom*, volume 2802 of *LNCS*, pages 50–59. Springer, 2003.

[47] Wai Mun Yee, M. Paniccia, T. Eiles, and V. Rao. Laser voltage probe (lvp): a novel optical probing technology for flip-chip packaged microprocessors. In *Physical and Failure Analysis of Integrated Circuits, 1999. Proceedings of the 1999 7th International Symposium on the*, pages 15 –20, 1999.

9 Glossary

A

AES	Advanced Encryption Standard
ALU	Arithmetic Logic Unit
ASIC	Application-Specific Integrated Circuit

B

BIST	Built-In-Self-Test

C

CMOS	Complementary Metal Oxide Semiconductor
CPUF	Controlled Physical Unclonable Function
CRP	Challenge Response Pair

D

DES	Data Encryption Standard
DH	Diffie-Hellman
DPM	Direct Part Marking
DRAM	Dynamic Random Access Memory
DRM	Digital Right Management
DSA	Digital Signature Algorithm

E

ECB	Electronic Code Book Mode
ECDSA	Elliptic Curve DSA
ECRYPT	European Network of Excellence in Cryptology
EDA	Electronic Design Automation
EEPROM	Electrically Erasable Programmable ROM

F

FIB	Focused Ion Beam
FPGA	Field Programmable Gate Array

I

IC	Integrated Circuit
ICT	Information and Communications Technology
IKE	Internet Key Exchange
IP	Intellectual Property
IPSec	Internet Protocol Security
JIL	Joint Interpretation Library

K

KEM	Key Encapsulation Mechanism
KDF	Key Derivation Function

L

LVP	Laser Voltage Probing

M

MD5	Message Digest Algorithm 5

N

NESSIE	New European Schemes for Signatures, Integrity and Encryption
NIST	National Institute of Standard and Technology
NVRAM	Non-Volatile Random-Access Memory

O

OAEP	Optimal Asymmetric Encryption Padding
OEM	Original Equipment Manufacturer
OFB	Output Feedback Mode
OTP	One-Time Programmable

P

PCB	Printed Circuit Board
PLD	Programmable Logic Device
PLL	Phase-Locked Loop
PRF	Pseudo-Random Function
PSS	Probabilistic Signature Scheme
PUF	Physically Unclonable Function

R

RAM	Random Access Memory
RFID	Radio-Frequency Identification
ROM	Read-Only Memory

S

SEM	Scanning Electron Microscope
SIL	Solid Immersion Lens
SIA	Semiconductor Industry Association
SRAM	Static Random Access Memory
STM	Scanning Tunnelling Microscopy

T

TRE	Time Resolved Emission
TSMC	Taiwan Semiconductor Manufacturing Company