Hybrid Side-Channel/Machine-Learning Attacks on PUFs: A New Threat?

Xiaolin Xu, Wayne Burleson
Department of Electrical and Computer Engineering
University of Massachusetts Amherst
Amherst, Massachusetts, 01002
Email: {xu, burleson}@ecs.umass.edu

***Abstract*—Machine Learning (ML) is a well-studied strategy in modeling Physical Unclonable Functions (PUFs) but reaches its limits while applied on instances of high complexity. To address this issue, side-channel attacks have recently been combined with modeling techniques to make attacks more efficient [25][26]. In this work, we present an overview and survey of these so-called "hybrid modeling and side-channel attacks" on PUFs, as well as of classical side channel techniques for PUFs. A taxonomy is proposed based on the characteristics of different side-channel attacks. The practical reach of some published side-channel attacks is discussed. Both challenges and opportunities for PUF attackers are introduced. Countermeasures against some certain side-channel attacks are also analyzed. To better understand the side-channel attacks on PUFs, three different methodologies of implementing side-channel attacks are compared. At the end of this paper, we bring forward some open problems for this research area.**

I. Introduction

Physical Unclonable Functions (PUFs) are now a class of well known security primitives, based on which various security protocols have been proposed [1] [3] [5] [6] [4] [7] [8]. A PUF works by digesting challenges and gather corresponding material imperfection and uncertainties for a unique identification (ID), which is hard to control and reproduce. Arbiter PUF is a well-known PUF example, which is depicted as Figure 1. The main component of an Arbiter PUF is two delay chains built with n delay cells based on $2-1$ MUX. A challenge vector composed by C_1, C_2 ... C_n is applied as the enable signal onto each MUX. Thus, a pulse applied at the beginning stage gathers the process variation from each delay cells, while passing through the circuit. The delay mismatch between two delay chains are then converted into the timing difference between T_A and T_B. A latch based arbiter digitizes the response into "1" or "0" by judging which is the first arrival.

As a security solution, resistance against malicious attack is an important metric in evaluating a good PUF. Because for a cryptographic system built on PUFs, modeling PUFs behavior means means knowing the inner security messages. In this case, exploring applicable PUF attack methods becomes an interesting topic. Moreover, studying the mechanism of different PUF attacks and proposing countermeasures are indirectly helpful to develop more secure PUF based protocols. For PUFs based on different mechanisms, a number of attacks are demonstrated and published [30] [10] [13] [9] [11] [12]. Among all the verified attack methods on PUFs, Machine Learning (ML) modeling strategy is a special one, which is mostly implemented in attacking the so-called Strong PUFs: a class of PUFs which digest a challenge vector and produce a corresponding response bit.

F978-3-9815370-2-4/DATE14/©2014 EDAA

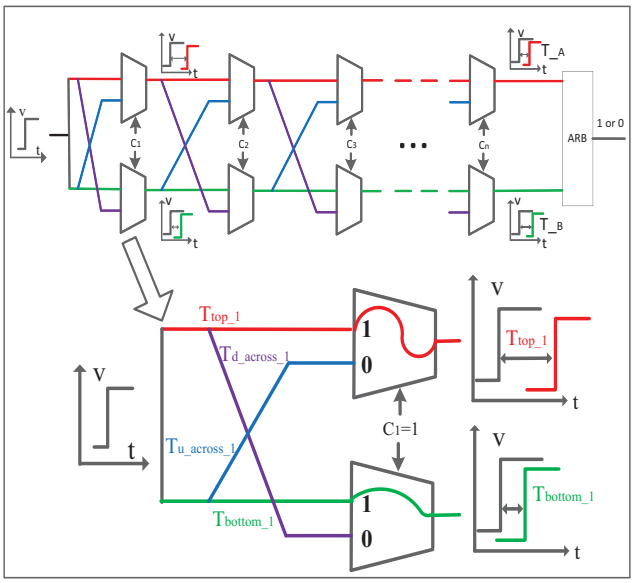

Fig. 1. Basic Arbiter PUF

Strong PUFs mainly include Arbiter PUF [23], XOR Arbiter PUF [23], Lightweight PUF [10], Ring Oscillator PUF [23] and Feed-forward Arbiter PUF [23]. The feature of public CRPs accessability makes Strong PUFs flexible, but also renders them vulnerable to ML modeling attacks [21] [22] [25], which digests a number of known challenge and response pairs (CRPs) to train a mathematic model. A completed model can then be utilized to to mimic the behavior of PUFs: predicting the responses for unknown challenge vectors [21] [22]. The commonly used modeling algorithm includes Support Vector Machines (SVMs) [30], Logistic Regression (LR) [21] [22] and Evolution Strategies (ES) [35] [36]. Take Arbiter PUF in Figure 1 as an example, in ML modeling attacks, all the silicon (delay) mismatch like T_{top_i}, T_{bottom_i}, $T_{u\_across\_i}$, $T_{d\_across\_i}$ (u *means up and* d *means down*) can be modeled with a

set of training (known) CRPs. While more known CRPs are applied, more precisely the inner silicon feature of PUF would be characterized.

Though ML modeling is demonstrated as effective in attacking some Strong PUFs, it is also concluded as reaching the limits when applied on Lightweight PUFs or XOR PUFs with bit-lengths of 256 or more and with 6 XORs or more [21] [22]. In this context, another conventional method: side-channel attack is combined with ML modeling method to address the issue of exponentially increasing computational workload [25]. The new hybrid side-channel attack is a special class of approaches which adapts some well-known strategies to the PUF case. Side-channel attack alone is an effective method to gain information from the physical implementation of a cryptosystem. The well-known means include power side-channel [14], timing side-channel attacks [16], electromagnetic attacks [17] and differential fault analysis [11], etc. Since the main purpose of combining side-channel attacks on PUFs is to decrease the workload of building ML models, the latter is still the key applicable solution on the Strong PUFs.

The rest of this paper is organized as follows: Section II describes the ML model and basic side-channel attack algorithms on PUFs. Section III describes some details of different side-channel attack methods and categorizes them according to their characteristics. Section IV analyzes the potential challenges of implementing PUF side-channel attacks as well as some opportunities, some countermeasures against side-channel attacks are also recalled in this section. Different methodologies like simulation, test chips and FPGA are compared in section V. Some open problems of side-channel attacks on PUFs are propsed in section VI while conclusion is summarized in section VII.

II. Threat Model and High-Level Attack Approaches on PUFs

Proposed as security primitives, PUFs work by digesting challenges and produce corresponding inner silicon signature. Due to the constant silicon features and rule-based challenge supply, PUF can be described into mathematical models, which reveal the complicated correlation between challenge and responses. Thus, if given a number of known CRPs, the built model of PUF can be used to predict unknown ones.

A. ML Model and Algorithms

According to [30], the inner silicon mismatches on PUFs can be modeled with known CRPs. Take the basic Arbiter PUF in Figure 1 as an example, since the final response o is determined by Arbiter to tell the first arrival pulse. If denote the four delay parameters of the i_{th} delay cell as: T_{top_i}, T_{bottom_i}, $T_{u\_across\_i}$, $T_{d\_across\_i}$, and challenge vector as: $C = C_1C_2...C_n$, we can model the delay of $(i+1)_{th}$ delay cell as (A for top and B for bottom):

$$\begin{aligned} D_A(i+1) =& ((1+C_{i+1})/2)(T_{top\_i+1} + D_A(i)) + \\ & ((1-C_{i+1})/2)(T_{u\_across\_i+1} + D_B(i)) \\ D_B(i+1) =& ((1+C_{i+1})/2)(T_{bottom\_i+1} + D_B(i)) + \\ & ((1-C_{i+1})/2)(T_{d\_across\_i+1} + D_A(i)) \end{aligned} \quad (1)$$

Please note that in Equation 1, the challenge bit C should be pre-processed following $C_i = 1 - 2 * C_i$. With the mathematic model of each delay cell, delay difference between each pair of delay cells can be derived as $\Delta_{i+1} = C_{i+1} * \Delta_i + \alpha_{i+1}C_{i+1} + \beta_{i+1}$, in which:

$$\alpha_i = (T_{top\_i} - T_{bottom\_i} + T_{d\_across\_i} - T_{u\_across\_i})/2 \quad (2)$$

$$\beta_i = (T_{top\_i} - T_{bottom\_i} - T_{d\_across\_i} + T_{u\_across\_i})/2 \quad (3)$$

Since the final signal delay $T_A = D_A(n)$ and $T_B = D_B(n)$ (accumulative signal delay from the starting stage to Arbiter) would determine final response o, following [30], we can denote the final time difference as $T_A - T_B =< \mathbf{p}, \mathbf{d} >$, in which $\mathbf{p}$ and $\mathbf{d}$ denote the parity vector from challenge and constant delay of PUF circuitry (for brevity, we do not give the explicit definitions here but refer the readers to [21] [22]).

The first modeling attacks on Arbiter PUF is demonstrated in [30], in which SVMs was utilized to do the binary classification with known CRPs. In such a model, an Arbiter PUF of N bit-length is viewed as a linear classifier dealing with each $N - bit$ challenge vector. An advanced modeling attacks on PUFs was proposed in [21] and [22], which applied Logistic Regression (LR) [34], Evolution Strategies (ES) [35] [36] and SVMs respectively for comparison. It is concluded that the LR framework achieved the best performance in modeling PUFs. Following the response pre-processing $R = 1 - 2 * R$, each challenge vector $C = C_1C_2...C_n$ obtains a probability p to characterize how likely the response will be a 1 or -1. The purpose of machine learning with a set of CRPs is to maximally re-producing these CRPs with the built PUF model.

B. Side-Channel Attacks

The ML modeling attacks have been demonstrated as reaching the limits while dealing with complicated XOR PUF and Lightweight PUFs [21] [22]. To further help the MLmodeling attacks, additional side-channels have been proposed to enhance model building [25][26]. Side-channel attacks is a class of attack, which extracts information from the physical implementation of a hardware system (usually a cryptosystem). Instead of brute force or theoretical weakness in the algorithms, side-channel explores the weaknesses from the practical implementation of a system. The demonstrated side-channel information mainly includes timing information, power consumption, electromagnetic (EM) leaks and sound, etc. From the implementation strategies, some side-channel attacks require technical knowledge of the internal operation of the system on which the cryptography is implemented, while others such as differential power analysis are effective as black-box attacks. Moreover, power and timing side-channel attacks are mainly based on statistical methods. For a complicated encryption function as PUF, it is infeasible to directly model its behavior based on side-channel information alone. However, knowing such information would greatly decrease the workload [25].

Taking power side-channel as an example, the main purpose of implementing power side-channel attacks is extracting varied power consumption derived by inner operations. Some well-known strategies include simple power analysis (SPA), differential power analysis (DPA) [14] and high-order differential power analysis. For a hardware implementation with power supply V_{DD}, its power consumption P can be expressed as: $P = V_{DD} * I$, where I stands for the real-time current.

Thus, by observing and collecting the current trace, the power consumption P, which is closely related with the inner digital keys, can be deduced.

III. Overview of Existing Hybrid Side-channel Attack on PUFs

Since the combination of side-channel attacks with ML is for decreasing workload of building PUF models, it is meaningful to discuss the two methods respectively. In this paper, we categorize the side-channel attacks on PUFs as: 1) passive attacks; 2) active attacks; 3) semi-invasive attacks; 4) hybrid attacks.

A. Passive Attacks on PUF

1) Power Side-Channel: Passive attacks is the main member of side-channel attacks, which only passively observe and collect information from the target instead of changing it. If necessary, the attackers possibly run the hardware to execute a specific behavior to extract some wanted information like timing, power consumption or electromagnetic leaks. As we described above, passive attacks like power side-channel and timing side-channel are based on statistical methods. So, subsequent data analysis and related tools are needed, like MatLab. The most relevant published passive attacks on PUF is by Mahmoud et al. [25], in which power side-channel attack is implemented to extract the sub-responses information of XOR Arbiter PUFs and Lightweight PUFs.

The basic mechanism of XOR Arbiter PUFs and Lightweight PUFs is encoding sub-response patterns with XOR function into a public response, to keep the original secret message from being known by outside world. The basic component of the above-mentioned two PUFs is an Arbiter PUF, which employs a latch as arbiter to determine the response. According to [25], the latch based arbiter consumes more power while generating a "1" response than a "0" one. Thus, the power consumption of whole XOR/Lightweight PUFs increases while more "1" sub-responses are generated. Thus, with power tracking method, power side-channel collects the current trace of whole PUF, with statistical processing tools, the current trace can be transformed into power consumption (the area below each current trace).

If we denote the charged XOR gate by the "0 to 1" response as a active one, the amount of extra power consumption is linearly proportional with the number of active XOR gates. So, with the power consumption information of each public response, it is feasible to deduce the proportion of "1" in sub-responses, thus greatly improves the success of guessing it.

2) Timing Side-Channel: Timing information is another common used side-channel parameter. Usually, a timing side-channel attack is simply implemented by observing variations in how long it takes to perform cryptographic operations. With timing side-channel information it might be possible to determine the entire secret key. Such attacks involve statistical analysis of timing measurements, and have been demonstrated across different areas [16] [27] [28].

Even though there is no published timing side-channel attacks on PUFs yet, we can predict some foreseeable strategies. Similar to power side-channel attacks described above, the purpose of collecting timing side-channel information is also providing additional information about the sub-response bits. Take Lightweight PUF with N sub-PUFs as a example, the XOR function at output network is actually composed by several basic XOR gates. Due to the process variations between each XOR gate, we can assume that different sub-response patterns would have different timing signatures (propagation delay). Thus, by sweeping the frequency of PUF circuitry, it is feasible to categorize public responses, and then refer the sub-response patterns.

B. Active Attacks on PUF

Different from passive attacks, active attacks attempts to alter system resources or affect their normal operation, which is mostly used in network attacks [32]. By manipulating the target or its environment outside of its normal behavior, the change of systems performance is observed and collected by attackers. Through analyzing the modified input-output pairs, inner working mechanism can be learned. Common used methods in active attacks include "fault injection", which uncovers cryptographic keys of system; changing program flow to break the integrity checks; etc.

A recently published active attack on PUF is by Delvaux and Verbauwhede [12]. In this literature, a PUF repeatability model is built based on the short-term reliability of the PUF as affected by CMOS (and interconnect) noise sources. Here, it is worth to distinguish the concept of noise from that of variability. For a normal functional electronic circuits, neither variability nor noise is desired. However, PUF is a special class of security primitives which measure the process variability. Thus, noise becomes the ideal "fault injection" candidate.

For a PUF circuits, noise mainly comes from temperature variations, supply voltage variation, etc. All of the noise source would reduce the CRPs repeatability of a PUF circuit. In [12], the presence of noise to characterize the variability relevant is explored for response bit generation. Based on the built repeatability PUF model, the fraction R of the responses which evaluates to "1" for a certain CRP is evaluated. By held $Repeatability\ Measurements$, the authors get the probability distribution function (PDF) of R. Statistical methods like $Least\ Mean\ Square\ (LMS)\ Method$ and $Differential\ Measurements\ Method$ are proposed to analyze the model. It is demonstrated that response repeatability can be exploited as a side channel for modeling strong PUFs.

C. Semi-invasive Attacks on PUF

Semi-invasive attacks stands for the attacks which physically break the cover of a implementation, without effecting its function. In [17] [29], Dominik et al. held a semi-invasive attack on Ring Oscillator (RO) PUFs while not damaging the underlying PUF structure. For a RO PUF, the process variations on die are transformed into operating frequency of each subcomponent, a counter choosing one pair of ROs is utilized for a 1-bit response. In Merli's work, RO PUFs implemented on $Xilinx\ Spartan\ XC3S200$ is decapsulated, on-die EM measurements is held to collect the frequency of each RO as side-channel information. With the complete list frequency range of all RO components determined, the whole RO PUF was modeled successfully. It is concluded in this

paper that no significant influence will be made on the physical structure after such a semi-invasive attack.

D. Hybrid Attacks on PUF

Even though demonstrated as an effective attack approach, pure ML modeling attacks reached the limit when attacking XOR Arbiter PUF or Lightweight PUF of 256-bit lengths of 256 or more and with 6 XORs or more [21] [22] [25]. To address this issue, a hybrid attack on PUF combining power side-channel attacks with conventional ML modeling attacks is proposed by Mahmoud et al. in [25] and Rührmair et al. in [26]. It is stated that power side-channel is of little value if taken by itself, but strongly improves an attackers capacity if suitably combined with modeling techniques.

In [25], good sub-response patterns like "all-1" and "all-0" are filtered out as side-channel information. Through this, the whole XOR Arbiter PUF and Lightweight PUF are decomposed into individual Arbiter PUFs. While enough CRPs are collected for each individual Arbtier PUF, current ML technique is capable to model them. The entire XOR Arbiter PUF and Lightweight PUF are attacked as long as each sub-PUF is learned. While in [26], Rührmair et al. present a more advanced machine learning (ML) techniques is adapted to efficiently exploit the side-channel information, with good error tolerance.

IV. Challenges and Opportunities for Attackers

Due to the complexity of PUFs and strict requirements for implementing side-channel attacks, there exit some challenges for PUF attackers. Meanwhile, the availability of variety of side-channel strategies also present opportunities for this research.

A. Challenges

1) Difficulty of Practical Implementation: In section III, we list some published side-channel attacks. In this section, we will analyze the challenges for these attackers. From the definition of side-channel attacks, the extracted information is usually of a subtle fraction compared with the entire measured value. This appears as a potential challenge for side-channel attacks on PUFs, because if measuring such subtle parameter is infeasible, the side-channel attack becomes worthless.

Take the power side-channel as an example, in [25], the CRPs used by Rührmair et al. are mainly from SPICE simulations. Even though with simulation it is not hard for one to precisely extract out the extra power consumptions of XOR gates. In practical implementations, power trace collected from PUFs on FPGA or ASIC chips would mix with a lot of noise. In this context, there is no significant difference between all the power trace, thus it is infeasible to deduce the sub-responses behind them. There are two reasons for this problem: 1): in real silicon PUFs, XOR tree usually consumes no more than 5% silicon resource of the whole design, charged XOR gates consume much less power than the whole circuits; 2): environmental and measurement noise have a great impact on the extracted power trace;

2) Countermeasures: Proposing various attack methods not only validates the quality of a PUF protocol, but also motivates people to look for countermeasures to perfect it. The conventional countermeasures against side-channel attacks mainly include: 1)Decrease the information leakage, like balance the processing of values; 2)Increase noise to circuit operation, this would cover the subtle changes.

Against the published side-channel attacks on PUFs, numbers of corresponding countermeasures were also developed. A method against power side-channel attack for XOR Arbiter PUFs and Lightweight PUFs is proposed in [25]. In response to the different power consumption between "1" and "0" responses, two crossed arbiters are employed at the end of each Arbiter PUF. With the improved Arbiter PUF, a constant number of "1" and "0" responses are produced, thus equal amount of power is consumed.

Two countermeasure examples are discussed in [11], against the semi-invasive side-channel attacks on RO PUFs. The first method modifies the RO components, which combines the so-called "non-overlapping" and "parallel comparison" ideas. By measuring a small number of oscillators in parallel and repeat such measurement on several groups (no overlap between any two groups), no repeated frequency information of a certain RO is leaked. The second method focuses on reducing the width of the measuring counters to decrease the leaked electro-magnetic radiation. An asynchronous ripple counters is employed where only the first flip-flop is clocked by the RO signal and all others follow asynchronously. The results demonstrate that compared with synchronous counters, ripple counters had less emanation.

B. Opportunities

Even though there would be challenges for attackers to implement side-channel attacks on PUFs, some opportunities also exist for this study. From the introduction above, it is not hard for one to note that, all the verified side-channel attacks have been proposed with corresponding countermeasures, which would better protect PUFs. However, built on complex inner variations, different PUFs would rely on varied silicon characteristics, it is difficult to bring forward a comprehensive countermeasure. As discussed in [33], all the proposed countermeasures are targeting some certain side-channel attacks respectively. Thus, it possibly renders a well protected PUF which is immune to power side-channel attacks, vulnerable to timing side-channel attacks, etc. Moreover, for now there is no obvious scheme which could protect PUF primitives against all of the existing side-channel attacks.

V. Methodologies

In this section, we list and compare different methodologies for verifying side-channel attacks. The most commonly utilized methodologies are: SPICE simulation, test chips and FPGA.

A. Simulation

Compared with other platforms, simulation is a convenient way to verify ideas. The well known tool is SPICE, with which one can precisely extract the wanted signals like power consumption [25] [26], timing difference, etc. Moreover, it is

TABLE I. ACHIEVEMENTS OF THE HYBRID ATTACKS

Methodologies	Variations	Cost	Flexibility	Duration
Simulation	library model	$	✓✓✓	+
Test Chips	silicon	$$$	✓	+++
FPGA	silicon	$$	✓✓	++

feasible to mimic the practical operation of PUFs by injecting some noise, such as temperature and supply voltage variations.

However, only validating side-channel attacks with data from simulation may renders the attack impractical. As we discussed in section IV, due to the existence of environmental and measurement noise, the successful power side-channel attacks on XOR Arbiter PUFs and Lightweight PUFs can not be directly applied onto real silicon PUF instances.

B. Test Chips

Test chips is a good way to validate different PUF-based security protocols and attack methods. Unlike simulation and FPGA, with ASIC test chips, people can optimize the circuitry and reserve some output pins for wanted signals. In [13], a class of 64-stage Arbiter PUFs realized in 65nm CMOS technology is learned with ML modeling attacks. A good 90% prediction is achieved from a training set of merely 500 CRPs. The shortcomings of test chips mainly lies in its cost, and a long-term duration is required to wait for the fabrication.

C. FPGA

FPGAs are important components for a considerable number of embedded systems, thus are the desirable platforms to implement PUFs. While their properties facilitate high performance, fast prototyping and hardware updates even after market launch, theft of Intellectual Property (IP) poses a serious threat to FPGA systems. The use of Physical Unclonable Functions (PUFs) in FPGAs seems to be a promising solution for IP protection [4] [31]. Moreover, based on the reconfigurability feature of FPGAs, side-channel attacks can be quickly verified and optimized. All these related features of the three metrologies are summarized in Table I.

VI. OPEN PROBLEMS

- As we discussed in section IV, for the proposed power side-channel attacks on PUFs, it still requires a feasible method to practically extracting subtle power consumption. Thus, looking for a applicable noise-removal strategy becomes interesting and necessary.
- According to the presented opportunities for PUF attackers, for existing PUF primitives, it will be a good research topic to explore a comprehensive countermeasure against the well-learned side-channel attacks. And undoubtedly, strategies against some certain side-channel attack is the theoretical basis and needed to be studied, too.
- For some proposed combined side-channel attacks on PUFs [25], the useable CRPs for ML are exponentially decreasing from XOR PUFs and Lightweight PUFs with more subcomponents. This strategy makes the useful CRPs to be rare, wasting a lot of known CRPs. Thus, better ML modeling algorithms as the one proposed in [26] are welcome to make this method more applicable.

VII. CONCLUSION

In this work, we studied some proposed side-channel attacks on PUFs, a tentative taxonomy is given according to their characteristics. Based on the recently published combined side-channel attacks on PUF, we point out that some challenges would make side-channel on PUFs impractical. We also present some opportunities in this research due to the diversity of applicable side-channel strategies.

REFERENCES

[1] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas: *Silicon physical random functions*, in *CCS*, 2002, pp. 148−160.

[2] K. Lofstrom, W. R. Daasch and, D. Taylor: *IC identification circuit using device mismatch*, in *IEEE International Solid-State Circuits Conference (ISSCC)*, 2002, pp. 372−373.

[3] Ravikanth Pappu, Ben Recht, Jason Taylor, Neil Gershenfeld: *Physical One-Way Functions*, Science, vol. 297, pp. 2026-2030, 20 September 2002.

[4] Jorge Guajardo, Sandeep S. Kumar, Geert Jan Schrijen, Pim Tuyls: *FPGA Intrinsic PUFs and Their Use for IP Protection*, in CHES 2007: 63-80

[5] Sandeep S. Kumar, Jorge Guajardo, Roel Maes, Geert Jan Schrijen, Pim Tuyls: *The Butterfly PUF: Protecting IP on every FPGA*, HOST 2008: 67-70

[6] Pim Tuyls, Geert Jan Schrijen, Boris Skoric, Jan van Geloven, Nynke Verhaegh, Rob Wolters: *Read-Proof Hardware from Protective Coatings*, CHES 2006: 369-383

[7] Marten van Dijk: *System and method of reliable forward secret key sharing with physical random functions*. US Patent No. 7,653,197, October 2004.

[8] Christina Bruzska, Marc Fischlin, Heike Schroder, Stefan Katzenbeisser: *Physically Unclonable Functions in the Universal Composition Framework*. CRYPTO 2011.

[9] Erdinc ztrk, Ghaith Hammouri, Berk Sunar: *Towards robust low cost authentication for pervasive devices.* In PerCom, pages 170-178. IEEE Computer Society, 2008.

[10] Mehrdad Majzoobi, Farinaz Koushanfar, Miodrag Potkonjak: *Lightweight Secure PUFs.* IC-CAD 2008: 607-673.

[11] Jeroen Delvaux, Ingrid Verbauwhede: *Side channel modeling attacks on 65nm arbiter PUFs exploiting CMOS device noise*. HOST 2013.

[12] Jeroen Delvaux, Ingrid Verbauwhede: *Fault Injection Modeling Attacks on 65nm Arbiter and RO Sum PUFs via Environmental Changes.* IACR Cryptology ePrint Archive, Report 2013/619.

[13] Gabriel Hospodar, Roel Maes, Ingrid Verbauwhede: Machine learning attacks on 65nm Arbiter PUFs: Accurate modeling poses strict bounds on usability. WIFS 2012: 37-42

[14] Paul C. Kocher, Joshua Jaffe, Benjamin Jun, *Differential Power Analysis*, Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology, p.388-397, August 15-19, 1999

[15] Daihyun Lim: *Extracting Secret Keys from Integrated Circuits*. MSc Thesis, MIT, 2004.

[16] P. C. Kocher: *Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems*, in Proc. CRYPTO',96, vol. LNCS 1109, pp.104-113 1996

[17] Dominik Merli, Dieter Schuster, Frederic Stumpf, Georg Sigl: *Semi-invasive EM attack on FPGA RO PUFs and countermeasures*. ACM Workshop on Embedded Systems Security (WESS'11), 2011.

[18] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld. Physical one-way functions. *Science*, 297(5589):2026, 2002.

[19] Ulrich Rührmair, Srinivas Devadas, Farinaz Koushanfar: *Security based on Physical Unclonability and Disorder*. In M. Tehranipoor and C. Wang (Editors): "Introduction to Hardware Security and Trust". Springer, 2011.

[20] Ulrich Rührmair, Christian Jaeger, Michael Algasiner: *An Attack on PUF-based Session Key Exchange, and a Hardware-based Countermeasure: Erasable PUFs.* Financial Cryptography and Data Security 2011.

[21] Ulrich Rührmair, Frank Sehnke, Jan Sölter, Gideon Dror, Srinivas Devadas, Jürgen Schmidhuber: *Modeling Attacks on Physical Unclonable Functions*. ACM Conference on Computer and Communications Security, 2010.

[22] Ulrich Rührmair, Jan Sölter, Frank Sehnke, Xiaolin Xu, Ahmed Mahmoud, Vera Stoyanova, Gideon Dror, Jürgen Schmidhuber, Wayne Burleson, Srinivas Devadas: *PUF Modeling Attacks on Simulated and Silicon Data*. IEEE Transactions on Information Forensics and Security (IEEE T-IFS), 2013.

[23] G. Edward Suh and Srinivas Devadas: *Physical unclonable functions for device authentication and secret key generation*. In Proceedings of the 44th Design Automation Conference, pages 9-14. IEEE, 2007.

[24] Blaise Gassend, Daihyun Lim, Dwaine Clarke, Marten v. Dijk, Srinivas Devadas: *Identification and authentication of integrated circuits* Concurrency and Computation: Practice & Experience, pp. 1077 - 1098, Volume 16, Issue 11, September 2004.

[25] Ahmed Mahmoud, Ulrich Rührmair, Mehrdad Majzoobi, Farinaz Koushanfar: *Combined Modeling and Side Channel Attacks on Strong PUFs*. IACR Cryptology ePrint Archive 2013: 632 (2013)

[26] Ulrich Rührmair, Xiaolin Xu, Jan Sölter, Ahmed Mahmoud, Farinaz Koushanfar, Wayne Burleson: *Power and Timing Side Channels for PUFs and their Efficient Exploitation*. IACR Cryptology ePrint Archive 2013: 851 (2013)

[27] D. J. Bernstein. *Cache-timing attacks on AES. Technical Report*, 37 pages, April 2005. Available at: http://cr.yp.to/antiforgery/cachetiming-20050414.pdf

[28] David Brumley , Dan Boneh, *Remote timing attacks are practical*, Proceedings of the 12th conference on USENIX Security Symposium, p.1-1, August 04-08, 2003, Washington, DC

[29] D. Merli, D. Schuster, F. Stumpf, and G. Sigl. *Side-channel analysis of pufs and fuzzy extractors*. In 4th International Conference on Trust and Trustworthy Computing (TRUST2011), Pittsburgh, PA, USA, June 2011. Springer.

[30] Daihyun Lim: *Extracting Secret Keys from Integrated Circuits*. MSc Thesis, MIT, 2004.

[31] E. Simpson and P. Schaumont: *Offline Hardware/Software Authentication for Reconfigurable Platforms*. In L. Goubin and M. Matsui, editors, Cryptographic Hardware and Embedded Systems - CHES 2006, volume 4249 of LNCS, pages 311-323. Springer, October 10-13, 2006.

[32] M. Balliu and I. Mastroeni: *A weakest precondition approach to active attacks analysis*. In PLAS 09: Proc. of the ACM SIGPLAN Fourth Workshop on Programming Languages and Analysis for Security, pages 5971. ACM, 2009.

[33] Francesco Regazzoni, Thomas Eisenbarth, Johann Groschädl, Luca Breveglieri, Paolo Ienne, Israel Koren, and Christof Paar: *Power Attacks Resistance of Cryptographic S-Boxes with Added Error Detection Circuits. In DFT*, pages 508-516. IEEE Computer Society, September 26-28 2007. Rome, Italy.

[34] C.M. Bishop et al. *Pattern recognition and machine learning*. Springer New York:, 2006.

[35] T. Bäck. *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford University Press, USA, 1996.

[36] H.P.P. Schwefel. *Evolution and Optimum Seeking: The Sixth Generation*. John Wiley & Sons, Inc. New York, NY, USA, 1993.