Defense Mechanisms against Machine Learning Modeling Attacks on Strong Physical Unclonable Functions for IOT Authentication: A Review

Nur Qamarina Mohd Noor
Advanced Informatics School
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia

Noor Azurati Ahmad
Advanced Informatics School
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia

Nurul Iman Mohd Sa'at
Advanced Informatics School
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia

Salwani Mohd Daud
Advanced Informatics School
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia

Nurazean Maarop
Advanced Informatics School
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia

Nur Syazarin Natasha Abd Aziz
Advanced Informatics School
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia

***Abstract*—Security component in IoT system are very crucial because the devices within the IoT system are exposed to numerous malicious attacks. Typical security components in IoT system performs authentication, authorization, message and content integrity check. Regarding authentication, it is normally performed using classical authentication scheme using crypto module. However, the utilization of the crypto module in IoT authentication is not feasible because of the distributed nature of the IoT system which complicates the message cipher and decipher process. Thus, the Physical Unclonable Function (PUF) is suggested to replace crypto module for IoT authentication because it only utilizes responses from set of challenges instead of cryptographic keys to authenticate devices. PUF can generate large number of challenge-response pairs (CRPs) which is good for authentication because the unpredictability is high. However, with the emergence of machine learning modeling, the CRPs now can be predicted through machine learning algorithms. Various defense mechanisms were proposed to counter machine learning modeling attacks (ML-MA). Although they were experimentally proven to be able to increase resiliency against ML-MA, they caused the generated responses to be instable and incurred high area overhead. Thus, there is a need to design the best defense mechanism which is not only resistant to ML-MA but also produces reliable responses and reduces area overhead. This paper presents an analysis on defense mechanisms against ML-MA on strong PUFs for IoT authentication.**

Keywords*—*IoT authentication; machine learning; modeling attack; Physical Unclonable Function; low area defense mechanism

I. INTRODUCTION

In todays' industrial and civil applications, there are vast number of devices that are connected in a network known as Internet of Things (IoT) [1]. Thus, many issues such as connectivity, power consumption and security have been arisen due to the implementation of IoT. The challenge in implementation of security is important for such devices because they are exposed to attacks. The designated hardware systems must be secured to avoid nullification of secure software implementation. Because of the owned and exchanged data are open for access, any information in the network must be secured to avoid the intervention of compromised data into systems. The infiltration of the malicious data can impair devices and applications especially for the applications that are highly dependent on data veracity, such as described by Barbareschi et al. [2] and Mukhopadhyay et al. [3].

For such applications, authentication becomes one of the most vital security features [3]. Traditional authentication technique for distributed system is typically based on cryptographic modules which are not feasible for implementation in IoT domain. The verifier plays a role as key manager who pre-register every device. The device is expected to use the issued cryptography key to authenticate itself [2]. This method requires message ciphering, each IoT device must comprises of at least a cryptography module, to accomplish security primitives requested by the verifier [3]. The number of devices within IoT system makes the message cipher and deciphers processes difficult to achieve because of the distributed nature of the identity verifier.

Thus, to simplify the authentication process of the IoT devices, silicon Physically Unclonable Function (PUF), has been revisited due to its ability to securely authenticate the IoT devices without requiring messages to be ciphered. The PUF generate unique responses from the set of challenges as the replacement of the cryptographic keys thereby, solve the message cipher and decipher issue. The silicon PUFs employ the unclonability and uniqueness which are produced by the manufacturing process of integrated circuits. These two features are utilized to map a set of challenges (the PUF inputs) to a set of responses (the PUF outputs), which is called challenge-response pairs (CRPs) set.

PUFs with large numbers of CRPs are defined as strong PUFs while PUFs with small numbers of CRPs are classified as weak PUFs. Strong PUFs are originated from delay-based PUFs

such as Arbiter PUF, Ring Oscillator (RO) PUF and Glitch (Anderson) PUF while weak PUFs are typically originated form SRAM PUF. The strong PUFs are suitable to be utilized as direct authentication scheme because they produce large set of CRPs thus the unpredictability is high. As for weak PUFs, they are suitable for key generation in cryptographic-based authentication scheme. However, both strong and weak PUFs are exposed to various kinds of attacks such as machine learning modeling attacks (ML-MA), side channel analysis (SCA), fault injection and physical tampering. This paper presents a comparison analysis on various defense mechanisms against ML-MA on variants of strong PUFs.

II. Background of the Problem

Strong PUFs can be directly designed to independently authenticate individual devices without the aid of any cryptographic module. There are two sequential steps to accomplish authentication scheme for devices using PUF [4] as described below:

- Enrollment: A substantial number of randomly chosen challenges is run by the verifier within the device, runs and the corresponding responses is stored in a secure database for future authentication operations [2].
- Verification: An unused challenge is selected by the verifier from the database to obtain a PUF response from the device. The device is verified by the verifier as authentic due to the unclonability property [2] if the response matches the previously recorded one.

To ensure the authentication scheme succeeds, the verifier must collect many CRPs during the enrollment so that it will have sufficient number of CRPs throughout the authentication process. The response provided by the device must be generated within short authentication period as well as be closely matched to the generated response stored by the verifier. Because of this authentication scheme makes use of large CRPs within short authentication period, the interface between verifier and device must be unrestrictedly open to allow the verification to complete faster. This makes the embedded PUFs in devices are subjected to ML-MA because the volume of CRPs can be learnt by the third party to eventually discard the unclonability property and model the strong PUFs [5].

To prevent other parties from building a model out of these PUF, various mechanisms which can be classified into three categories have been proposed. The first category is design-based where the defense mechanism is added in the architectural design of the PUFs such as adding non-linearity using XOR logics [6], [7], modifying transistor-level design [8], exploiting FPGA blocks [9], [10] and analogizing the digital components [11]. The second category is the obfuscation-based where the defense mechanism is performed masking of either challenges [12] or responses [2], [13]. The final category is the access control where the defense mechanism complicates the interface access to protect it from being openly accessed.

Although there are considerable numbers of defense mechanisms available, the best defense mechanism which provides low prediction accuracy with minimal area overhead and generates unique and reliable PUFs responses is still not achieved. All the defense mechanisms resist or at least improve the resiliency of the PUFs against the ML-MA however they come with some shortcomings. In the case of design-based defense mechanisms, adding non-linearity using XOR [6], [7], analogizing the digital components [11] and exploiting the FPGA blocks [4], [10] increase the hardware overhead.

III. PUFs: Variants and Attacks

People and objects are regarded as ‘things’ in Internet of Things. The monitoring and control of these ‘things’ is achieved using devices such as sensors and actuators via communication technologies. The services such as device monitoring, device control, and device search are also delivered in IoT system. The users are provided with applications which have user interface (UI) for controlling and monitoring the IoT system. The security component provides authentication, authorization, message and content integrity in an IoT system. However, security is not apparently highlighted as the crucial component in IoT by most of the vendors of IoT system.

Barbareschi *et al.* [2] stated that PUFs is more suitable for IoT authentication as opposed to the cryptographic algorithm because numerous devices in IoT system causes the message cipher and decipher process difficult to achieve. This is because the identity verifier must work in distributed manner, by enrolling every device. Each device is expected to use the issued key to authenticate itself. Furthermore, the generated keys must be secured on the database managed by the verifier and also on non-volatile memories (NVMs), to circumvent loss of data upon power off.

Mukhopadhyay [14] described the weaknesses of using the crypto-based authentication schemes for IoT-based light-switch system. In this system, smart bulbs with proximity tags, Bluetooth low energy (BLE) signal and a Zigbee-based Ethernet/WiFi-enabled bridge were used as the remote lighting control system to help reduce the energy consumption. The use of Zigbee as the wireless medium which is open by design caused the IoT to be susceptible to eavesdropping, jamming and message injection attacks. Due to the mentioned vulnerabilities, the MD5 hash functions were adopted as the authentication method. The MD5 hash functions were computed based on the device’s MAC address. However, the usage of MD5 hash functions as the authentication method has two significant weaknesses, namely, the secret white list token was not random (clonability) and the MAC addresses are easily recovered (predictability).

Physical Unclonability Function (PUF) on the other hand, is the physical representation of a function that makes it difficult to clone and produces an unpredictable challenge–response pair (CRP) behavior [14]. According to Pappu *et al.* [15], the PUF is ideally hard to characterize or model, but somehow its CRPs are reliably evaluated [16]. Boehm and Hofer [17] described that a PUF utilizes production variability to produce a device-specific output in a form of binary number. A PUF comprises of several components which are defined by local parameter variations. The differences in local parameter variations are known as local mismatches. These local mismatches are merged and directly read out to generate the binary output. Since these local mismatches cannot be controlled externally, a PUF cannot be

replicated thereby, it is unclonable. The properties of a PUF are described as follows [3]:

- Reliable: The generated response from the set of CRPs correctly represents the identity embedding function.
- Unclonable: It is hard to construct a procedure to reproduce the set of CRPs in a function.
- Uniqueness: It is hard to compute response from the set of CRPs.

Based on these properties, the PUFs possess several properties like MD5 hash functions, in the sense that they are one way. The fact that the PUF response is unpredictable and unclonable makes the PUF response cannot be predicted or computed thereby, it is suitable to replace the MD5 hash functions. However, PUFs come in many variants which must be evaluated to determine which variant is suitable for IoT authentication.

A. Variants of PUFs

PUFs are divided into two variants namely strong PUFs and weak PUFs as shown in Fig. 1. Maes [18] provided a definition on strong and weak PUFs. A PUF is defined as a weak PUFs if it has small challenge set. There is a weak PUFs called a physically obfuscated key (POK) that has only a single challenge. On the other hand, PUFs with a large challenge set are known as strong PUFs and their CRPs are unpredictable whereby it is not possible to build an accurate model of the PUF based on resulted CRPs.

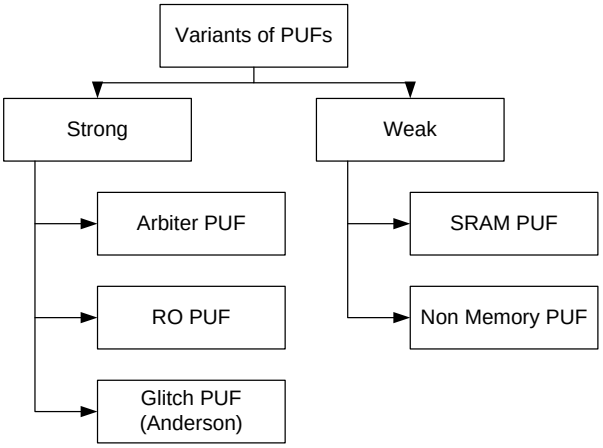

Fig. 1. Variants of PUFs.

1) Weak PUFs

Weak PUFs are typically used for storing secret keys as alternatives to non-volatile memories (NVMs) such as ROM and Flash. The characteristics of the weak PUFs were listed by Armknect *et al.* [19] and Ruehmair *et al.* [20] as follows:

- Small and fixed set of challenges where it commonly has only one challenge per PUF instance.
- The access interface to the generated responses is restricted although the adversaries may physically possess the PUF-carrying hardware.

Maes [18] described that weak PUFs are typically designed using intrinsic variations that exist in the integrated circuits. The intrinsic PUFs are cost effective because they are fabricated using standard CMOS logic parts. The first CMOS-based weak PUF was proposed by Lofstrom *et al.* [21] which utilized the threshold mismatch to identify circuits. Tuyls *et al.* [22] developed a PUF based on the capacitance sense from specially applied protective coatings. A PUF which is based on a chip-ID circuit was demonstrated by Su *et al.* [23]. The chip-ID circuit was developed based on cross-coupled devices. The ID was evaluated based on the transition of the cross coupled devices from a metastable state to a stable state which is controlled by process variation. Based on the similarity of this design to the feature of SRAM, there were numerous literatures were inspired to perform research and development of SRAM-based weak PUFs.

SRAM PUFs are developed based on the intrinsic threshold variation of the cross-coupled SRAM cells. The cells are differential in nature thereby they are sensitive to variation (uniqueness) and highly immune to common-mode noise (reliable). Because of these features, the cross-coupled SRAM cells are suitable to be design as PUF. Furthermore, the existence of SRAM in nearly all VLSI circuits makes them highly eligible as PUFs. According to Holcomb *et al.* [24], the CRPs of the SRAM PUFs are generated in the cell during the transition from the off state to an on state. The generated responses are then read out using the standard memory access mechanism [25]. There is also another method of producing CRPs which employs the small amount of data retention voltage of cells instead of the power-up state as proposed by Holcomb *et al.* [26]. Apart from SRAM PUFs, there are other types of weak PUFs were proposed which were either still based on memory or non-memory. Examples of PUFs that are based on the memory design characteristics are Flash [27], DRAM [28] and Memristors [29]. As for the non-memory PUFs, Kumar *et al.* [30] proposed the butterfly PUF which utilizes the cross-coupled latches in FPGAs while Simons *et al.* [31] developed a PUF which is based on bus keepers as an alternative to D Flip-Flop PUF. All the PUFs that were described above only have one designated way to produce the CRPs hence there will be only one challenge per PUF instance.

2) Strong PUFs

As opposed to the weak PUFs, the strong PUFs produce complex CRPs because of different kinds of intrinsic variations in the PUF. The generated responses are acquired from numerous physical components therefore a very huge number of possible challenges must be applied to the PUF. The characteristics of the strong PUFs were detailed by Brzuska *et al.* [32] and Chen *et al.* [33] as follows:

- Huge and variety set of challenges which avoid the full read-out of all CRPs, although the adversary may physically possess the PUF for ample amount of time.
- Unprotected challenge-response interface where an adversary may arbitrarily apply challenges to the strong PUF and read out the generated responses.

The strong PUFs originate from delay-based PUFs where random variations on the delay of a digital circuit are measured. Arbiter PUF which was first described by Lee *et al.* [34] is an

example of the delay-based PUFs. This type of PUF exploits the variation in the runtime delays of electrical components. The electrical signals in an Arbiter PUF begin their journey through a sequence of k stages where each stage comprises of two multiplexers [35]. The exact path for each signal is determined by k external bits which are applied as one bit per stage. The destination of the electrical signals is ended by a final latch-based arbiter element. Arbiter PUFs with k stages have 2^k challenges where each challenge produce one-bit response. The susceptibility of the Arbiter PUF to ML-MA has resulted to the development of more enhanced version of Arbiter PUF. These enhanced versions commonly utilize non-linearity in the original Arbiter PUFs to resist ML-MA [36]. The examples of the enhanced version of Arbiter PUFs are Feed-Forward Arbiter PUFs [37], [38], Lightweight PUF [39] and XOR Arbiter PUFs [5].

Another type of Strong PUFs is the ring oscillator PUF (RO PUF) which was introduced by Gassend *et al.* [40]. The ring oscillator in their design is a variant of the switch block-based delay line as proposed for the arbiter PUF. A negative feedback is applied to transform the delay circuit into an oscillator. To enable/disable the oscillation, an additional AND-gate in the loop is utilized [41]. To count the number of oscillating cycles during certain time interval, a frequency counter is connected to the oscillating signal. The counter value indicates the oscillating frequency. A simple edge detector processes the oscillating signal to ensure the counter is enabled every time a rising edge is detected [42]. The frequency of the ring oscillator is limited to half the clock because of the use of edge detector. The resulted frequency of equally implemented ring oscillators on distinct devices is considered as a PUF response [43].

There is also another kind of strong PUFs which is developed based on glitch behavior of combinatorial logic circuits. Since internal state does not exist in a pure combinatorial circuit, the input signals have total influence on the steady-state output. However, if the logical value of the input changes, transitional effect such as delays occurred whereby some time is required for the output has its steady-state value. The delays are known as glitches which is determined by the differences in time of arrival for the different logical paths from the inputs to an output signal [18]. The number, shape and occurrence of the glitches on its output signals will be instance-specific and partially random because these glitches are highly influenced by random process variations. Thus, by accurately measured these glitches, their behavior can be utilized as a PUF response.

Anderson [44] developed a glitch-based PUF construction specifically for FPGA platforms which is known as Anderson PUF. Based on the delay variations in the circuit, a custom logical circuit is implemented. The output of this logical circuit is connected to the preset signal of a flip-flop to captures the glitch in case if it occurs. The output of the circuit is treated as the single PUF response bit. By placing many of these logical circuits on an FPGA, many PUF response bits can be produced.

B. Attacks on PUFs

PUFs are exposed to various types of attacks ranging from invasive to non-invasive as depicted by Fig. 2. According to Wachsmann and Sadeghi [45], the invasive attacks require physical modification of the PUF in order to gain deeper knowledge on the PUFs implementation. This type of attack typically affects the weak PUFs because it only has one challenge thereby performing physical modification is possible. As opposed to the invasive attacks, the non-invasive attacks invisibly collect information without being physically harmful to the PUFs. This type of attack usually occurs to strong PUFs because it has huge numbers of challenges thereby data must be gathered and processed. The next section will describe the types of invasive and non-invasive attacks which are encountered by weak and strong PUFs, respectively.

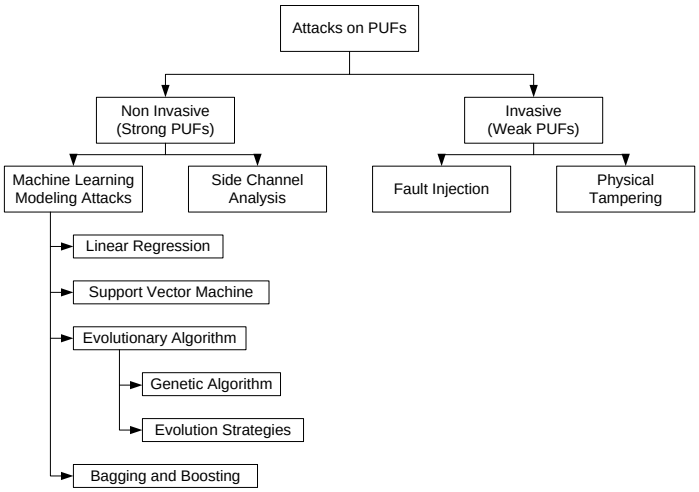

Fig. 2. Attacks on PUFs.

1) Attacks on Weak PUFs

Weak PUFs are subjected to fault injection and invasive attacks. According to Wachsmann and Sadeghi [45], the objective of the fault injection attacks is to induce erroneous behavior in a PUF through internal manipulation and if the manipulation is combined with cryptanalysis, the fault injection attacks can lead to key recovery attacks. There are many ways for the faults to be injected in a PUF such as by injecting transient faults into specific components of the PUF [46], [47] to attain the PUF response or by operating the PUF under extreme environmental conditions which produces decay effects on memory-based PUFs [48], [49]. Another fault-injection attack which is based on the decay effects in volatile memory is presented by Oren *et al.* [50]. This attack manipulates the internal structure of SRAM PUFs which makes them susceptible to cloning.

On the other hand, invasive attacks analyze the PUF hardware to gain the information on the cryptographic secrets stored in a PUF. The reverse engineering and circumvention of active protection mechanisms are the most common forms of invasive attacks. Tarnovsky [51] demonstrated the vulnerability of the algorithms and circuits that are utilized to process the PUF responses to invasive attacks through micro-probing the logic blocks, registers and the bus transfer of PUF devices. As for SRAM PUFs, they are susceptible to extreme operating conditions [52] and physical tampering [53]. As a result, the SRAM PUF hardware can be physically inspected and modified [54]. Furthermore, it was established by Helfmeier *et al.* [53] that upon gaining the response of an SRAM PUF, C_1, a focus ion

beam (FIB) can be utilized to alter the circuits of SRAM PUF, C_2 so that C_2 will have a very similar challenge/response behavior as C_1.

2) Attacks on Strong PUFs

Fault injection and invasive attacks on Weak PUFs are less applicable to Strong PUFs. The most relevant attack method for Strong PUFs is known as machine learning-based modeling attack (ML-MA) which was introduced by Ruehmair *et al.* [5]. There are three common machine learning algorithms namely logistic regression (LR), Support Vector Machine (SVM) and evolutionary algorithms (EA) such as genetic algorithm (GA) and evolution strategies (ES) that are used to perform modeling attacks. Abu-Mostafa *et al.* [55] defined LR as a supervised machine learning framework which differs from linear regression in as sense that it outputs a probability between 0 and 1 instead of produces ±1 output. As for SVM, it is a tool that utilizes the optimal margin between vectors to determine the best hyperplane and this method requires computing the distances of input vectors from the hyperplane [55]. According to Saha *et al.* [56], GA is designed to handle integer and binary string solutions by mimicking biological evolution like ES using similar concepts such as reproduction, mutation, recombination/crossover and selection. As for ES, they are utilized to generate population heuristically by adapting the generated population to certain environmental conditions [57]. The data set is randomized to avoid them from linearly separable but the resulted model must be parameterized to ensure the data set is reliable.

As for other machine learning algorithm such as Bagging and Boosting (B & B), it was first used by Vijayakumar *et al.* [7] to perform modeling attacks on strong PUFs. B & B are considered as ensemble meta-algorithm approaches as described by Schapire [58]. Ensemble learning is a technique of merging the predictions from several classifiers to generate a robust classifier. An emerging machine learning algorithm namely deep learning (DL) was also used to perform modeling attack on strong PUFs as proposed by Yashiro *et al.* [8]. They described that DL has superior performance compared to conventional machine-learning methods on a benchmark test in the field of image recognition. DL is defined by Yashiro *et al.* [8] as a multi-layer neural network where it has more than two layers. The output of a layer acts as input for the following layer. This mechanism allows the partition function to be developed which is used to classify input data in accurate and efficient manner.

According to Rührmair *et al.* [5], an adversary first collected vast numbers of CRPs from the strong PUF to perform ML-MA. Next, the adversary infer the behavior of the PUF on the unknown CRPs by combining the numerical method with the internal parametric model of the PUF. The impact of ML-MA is surprisingly massive since all strong PUFs including enhanced version of Arbiter PUFs are still vulnerable to this attack. Modeling attacks are inapplicable to weak PUFs, since they only have one challenge per PUF instance. Another attack that is associated to strong PUF is known as side channel analysis (SCA) [59]. The adversary performs SCA by observing the non-functional metrics of PUF such as the timing information and power consumption to extract information for developing ML-MA. The potential SCA on the design block for processing PUF response such as in fuzzy extractor as discussed by Merli *et al.* [60]. In general, all known SCA on PUF-based systems have some difficulty to attack the main PUF component thereby, they prefer to target the design block that is utilized to process the PUF responses, such as fuzzy extractors. Since SCA alone is difficult to be performed on PUF components, Mahmoud *et al.* [61] proposed to combine ML-MA with SCA to improve attack performance.

IV. IoT Authentication-Strong or Weak PUFs

The basic PUF-based authentication scheme as described by Gassend *et al.* [40], Devadas *et al.* [37] and Barbareschi *et al.* [2] comprises of two phases namely enrollment and verification:

- *Enrollment*: Prior to the deployment, every entity must be enrolled by the verifier. The identity (ID) of every entity is recorded by verifier during the enrollment phase. The verifier also accumulates a substantial subset of CRPs from the device's PUF. The collected challenge-response pairs are stored in the verifier's database (DB) indexed by the entity's ID.
- *Verification*: The verification phase requires the PUF challenge to be sent to the device where the device analyzes its PUF. The replied response is validated by the verifier to check whether it matches to the response it has in its database. If they match, the device is authenticated, otherwise the authentication is rejected. The used CRP is then omitted from DB.

The success of the above authentication scheme relies on the fact that the verifier must collect many CRPs during the enrolment stage so that the CRPs will not run out as emphasized by Halak *et al.* [4]. According to Maes [18], the PUF responses must reproduced within smaller intra distance to ensure that the replied response matches the stored response in the DB without possibility being predicted. Since the successful authentication relies on the large challenges and unpredictability of the responses, it is implied that the suitable type of PUFs for IoT authentication is strong PUFs.

However, the strong PUFs are exposed to the non-invasive attacks as described in Section B. The basic authentication scheme is only secured for a fully unclonable PUFs including the PUFs that are unable to be cloned through machine learning modeling. Strong PUFs which are exposed to ML-MA do not provide secure authentication because the basic protocol cannot differentiate between the real entity with the physical PUF and an adversary with a modeling clone of that PUF. Thus, to be able to provide secure authentication scheme, proper defense mechanisms for strong PUF against modeling attacks and side-channel analysis must be evaluated.

V. Comparison Analysis of Defense Mechanisms Against Attacks on PUFs for IoT Authentication

There are several techniques available as defense mechanism for strong PUFs against non-invasive attacks such as modeling attacks and side-channel analysis. The comparison between the defense mechanisms for strong PUFs against the non-invasive attacks is conducted in Table 1.

TABLE I. SUMMARY OF COMPARISON ANALYSIS OF DEFENCE MECHANISMS

Authors	Types of Strong PUF	Types of Non-invasive Attacks	Proposed Defense Mechanisms	Strengths	Weaknesses
Goa *et al.* (2013) [5]	Arbiter PUF	ML-MA • LR	Partially obfuscates challenge	Resist attack although millions of CRP are used	Instable generated response
Zheng *et al.* (2016) [11]	DR-PUF based on Analog Arbiter and Glitch PUF	SCA	Uses analog blocks for designing Arbiter PUF	Resist side channel attack	Increase run time and area overhead
Merli *et al.* (2013) [66]	RO PUF	SCA • DPA	Masks the challenge with code word	Prevent first order DPA	Instable generated response
Miao *et al.* (2016) [63]	LRR DPUF based on VLSI interconnect randomness by lithography variations	ML-MA • SVM	Augments the interconnect using cross-coupled logic network	Provide constantly low predictions	Exposed to transistor aging
Tobisch and Becker (2016) [69]	Arbiter PUF	ML-MA • LR	Noise bifurcation (obfuscates response) introduced by Yu *et al.* (2014)	Provide resistance against attacks for large PUF instances	Software model is needed on the server' side
Ye *et al.* (2015) [67]	Arbiter PUF	ML-MA	Obfuscate the logic for path segments selection using cross-coupled inverter	Stable generated responses	Incurs high area overhead
Marten Van Djik and Ruehmair (2014) [70]	Arbiter PUF	ML-MA and SCA	Pre- and post-processing to allow more complex access control	Creates unclonable CRPs	Susceptible to noises
Rührmair *et al.* (2013) [6]	Arbiter PUF	ML-MA • LR • ES	• Increasing challenge bit length • Adding non-linearity	Improves resistance against machine learning attacks	• Instable generated response • Increased hardware overhead
Barbareschi *et al.* (2015) [2]	Enhanced Anderson PUF	ML-MA	Hides PUF responses using AES	Provide better unpredictability	Incurs high area overhead
Mukhopadhyay (2016) [14]	LSPUF based on Arbiter PUF	ML-MA • ES	Fuses the access point to CRPs	Increases complexity to build ML model	Not directly secured against ML-MA
Wallrabenstein (2016) [68]	RO-PUF	ML-MA	Applies elliptic curve crypto module construct to obfuscate responses	Provide better unpredictability	Incurs high area overhead
Vijayakumar *et al.* (2016) [7]	Arbiter PUF	ML-MA • SVM • LR • B & B • ES	Non-linear XOR logic function with high cardinality / entropy	Provide better resistance against all attacks especially B & B attacks	Incurs high area overhead
Capovilla *et al.* (2015) [62]	Arbiter PUF	ML-MA and SCA	Configures gate size in accordance to arbiter elements	Generate stable responses	Exposed to transistor aging
Kumar and Burleson (2016) [64]	Feed Forward PUF	ML-MA + SCA	Reduces number of stages/loops	Exhibit better unpredictability	Induces error in CRPs
Yashiro *et al.* (2016)	Arbiter PUF	Deep Learning	Tightens the layout conditions to make P and R difficult	Has higher tolerance against ML-MA	Instable generated response
Yu *et al.* (2016) [8]	XOR Arbiter PUF	ML-MA and SCA	Employs lockdown protocol which requires server's permission to obtain new CRPs	Makes the PUF exponentially difficult to learn	Incurs high area overhead
Zhang *et al.* (2016) [71]	Enhanced Anderson PUF	ML-MA	Adds reconfigure ability to the PUF design	Provides high CRP uniqueness	Incurs high area overhead
Idriss *et al.* (2016) [15]	Arbiter PUFs	ML-MA	Hide CRP using cryptographic functions	Provide high unpredictability	Incurs high area overhead
Amsaad *et al.* (2016) [10]	RO PUF	ML-MA • SVM • GA	Exploit FPGA resources to build multi-stage structure	Improves CRP space in terms of uniqueness and reliability	Incurs high area overhead
Zalivaka *et al.* (2017) [12]	Arbiter PUF	ML-MA • SVM • LR	Obfuscates the strong challenges using FPGA resources	Decrease the ML prediction rate	Instable generated response

Based on Table 1, the defense mechanisms that were proposed can be divided into three categories namely design, obfuscation and access control. The design category consists of adding non-linearity using XOR logics [6], [7], configuring transistor's gate sizes [62], augmenting interconnects [63], tightening the layout condition [8], exploiting FPGA resources and reconfigurability [9], [10] and modify current PUF design blocks by using analog blocks [11] and reducing feed-forward stages [64]. These defense mechanisms provide resistance to machine learning modeling attacks however they come with certain shortcomings. Adding non-linearity using XOR logics, modifying the PUF design blocks using the analog blocks and exploiting FPGA resources and reconfigurability incur high area overhead. Configuring transistor's gate sizes and augmenting interconnect make the PUF to be susceptible to the transistor aging. Tightening the layout condition and reducing the feed-forward stages generate instable PUF responses.

As for the obfuscation category, the obfuscation is performed either on PUF challenges or responses. The simplest obfuscation technique for PUF challenges was proposed by Rührmair *et al.* [6] where the challenges' bitlength is increased. Goa *et al.* (2013) partially obfuscated the PUF challenges and Zalivaka *et al.* [65] segregated between strong and weak challenges, eliminated the weak challenges and obfuscated the strong challenges using FPGA resources to successfully resist the ML-MA. As a countermeasure against SCA, Merli *et al.* [66] masked the challenges with code word. This technique was able to prevent first order DPA. However, all the obfuscations on the PUF challenges generated instable response. Ye *et al.* [67] solved this issue by instead of directly obfuscating the challenges, the obfuscation was performed the logics for path segments selection for PUF challenges. However, this solution came with the cost of high area overhead. The obfuscation on the PUF responses is typically performed using the crypto module such as AES [2], [15] and elliptic curve [68]. These techniques increase the unpredictability of the PUFs, however, still they incurred high area overhead. There is also another technique of obfuscating the PUF responses which utilizes noise bifurcation [69]. The downside of this technique is that the software model must be developed and stored on the server's side.

The third category is the access control in which Djik and Rührmair [70] adding the complexity to the access control by performing pre- and post-processing of the CRPs. However, this technique makes the PUFs susceptible to noises. Mukhopadhyay [14] temporarily fused the access point to CRPs to increase the complexity to build ML model but the PUFs are still not directly secured against ML-MA. Another defense mechanism related to access control is developed by Yu *et al.* [71]. The authentication using PUFs was performed by employing lockdown which requires server's permission to obtain new CRPs. This technique makes the PUF's CRPs exponentially difficult to learn but it comes with the cost of high area overhead.

VI. Conclusion

There are several defense mechanisms against non-invasive attack particularly ML-MA on variants of strong PUFs for IoT authentication. All these defense mechanisms were claimed to provide resistance or at least improve resiliency of the strong PUFs against ML-MA. Each defense mechanism has their own strengths and weaknesses. The most apparent weakness that they exhibited is either the area overhead is high or the generated PUF responses are instable. The issues regarding PUF responses and area overhead that were incurred by these defense mechanisms must be solved to ensure that the selected variant of strong PUFs for IoT authentication is at their best performance. Thus, there is a gap in determining the most suitable variant of strong PUF that provide best defense mechanism that solve the issue of unreliable responses and high area overhead. To resolve these issues, a suitable variant of strong PUF with reliable responses and low area overhead must be developed for a quality IoT authentication.

Acknowledgment

We would like to express our gratitude to Ministry of Higher Education (MOHE Malaysia) for providing financial support (research grant Q.K130000.2538.11H85) in conducting our study. Our special thanks to Universiti Teknologi Malaysia (UTM) and specifically Advanced Informatics School (AIS) for realizing and supporting this research work.

References

[1] S. Koley and P. Ghosal, "Addressing hardware security challenges in internet of things: Recent trends and possible solutions," *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*, Beijing, 2015, pp. 517-520. DOI: 10.1109/UIC-ATC-ScalCom-CBDCom-IoP.2015.105

[2] M. Barbareschi, P. Bagnasco and A. Mazzeo, "Authenticating IoT Devices With Physically Unclonable Functions Models," *2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, Krakow, 2015, pp. 563-567. DOI: 10.1109/3PGCIC.2015.117.

[3] D. Mukhopadhyay and R.S. Chakraborty, Hardware Security: Design, Threats, and Safeguards. CRC Press, Taylor and Francis Group. 2015.

[4] B. Halak, M. Zwolinski and M.S. Mispan, "Overview of PUF-Based hardware security solutions for the internet of things. 2*016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Abu Dhabi, 2016, pp. 1-4. DOI: 10.1109/MWSCAS.2016.7870046.

[5] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, and J. Schmidhuber, "Modeling attacks on physical unclonable functions," Proceedings of the 17th ACM conference on Computer and Communications Security. Chicago, Illinois, USA, October 04 - 08, 2010, pp. 237-249. https://doi.org/10.1145/1866307.1866335

[6] U. Rührmair, J. Sölter, F. Sehnke, Xu, A. Mahmoud, V. Stoyanova, G. Dror, J. Schmidhuber, W. Borleson, "PUF Modeling Attacks on Simulated and Silicon Data," I*EEE Transactions on Information Forensics and Security*, vol. 8, no. 11, pp. 1876-1891, Nov. 2013. DOI: 10.1109/TIFS.2013.2279798.

[7] A. Vijayakumar, V.C. Patil, C.B. Prado, and S. Kundu, "Machine learning resistant strong PUF: Possible or a pipe dream?," *2016 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, McLean, VA, 2016, pp. 19-24. DOI: 10.1109/HST.2016.7495550.

[8] R. Yashiro, T. Machida, M. Iwamoto, and K. Sakiyama, "Deep-Learning-Based Security Evaluation on Authentication Systems Using Arbiter PUF," In: Ogawa K., Yoshioka K. (eds) Advances in Information and Computer Security. IWSEC 2016. Lecture Notes in Computer Science, vol 9836. Springer, Cham. https://doi.org/10.1007/978-3-319-44524-3_16.

[9] J. Zhang, Q. Wu, Y.P. Ding, Y.Q. Lv, Q. Zhou, Z.H. Xia, X. M. Sun, and X.W. Wang, "Techniques for Design and Implementation of an FPGA-Specific Physical Unclonable Function," J. Comput. Sci.

Technol. (2016) 31: 124. https://doi.org/10.1007/s11390-016-1616-8.

[10] F. Amsaad, M. Choudhury, C.R. Chaudhuri, and M. Niamat, "An Innovative Delay based Algorithm to Boost PUF Security Against Machine Learning Attacks," *2016 Annual Connecticut Conference on Industrial Electronics, Technology & Automation (CT-IETA)*, Bridgeport, CT, 2016, pp. 1-6. DOI: 10.1109/CT-IETA.2016.7868242.

[11] J.X. Zheng, T. Xu and M. Potkonjak, "Securing Embedded Systems and their IPs with Digital Reconfigurable PUFs," *2016 26th International Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, Bremen, 2016, pp. 169-176. DOI: 10.1109/PATMOS.2016.7833683.

[12] S.S. Zalivaka, A.A. Ivaniuk and Chip-Hong Chang, "FPGA implementation of modeling attack resistant arbiter PUF with enhanced reliability," *2017 18th International Symposium on Quality Electronic Design (ISQED)*, Santa Clara, CA, 2017, pp. 313-318. doi: 10.1109/ISQED.2017.7918334.

[13] T. Idriss, H. Idriss, and M. Bayoumi, "A PUF-Based Paradigm for IoT Security," *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, Reston, VA, 2016, pp. 700-705. DOI: 10.1109/WF-IoT.2016.7845456.

[14] D. Mukhopadhyay, "PUFs as Promising Tools for Security in Internet of Things," in *IEEE Design & Test*, vol. 33, no. 3, pp. 103-115, June 2016. DOI: 10.1109/MDAT.2016.2544845.

[15] R. Pappu, "Physical One-Way Functions," *Science*, 20 Sep 2002: Vol. 297, Issue 5589, pp. 2026-2030 DOI: 10.1126/science.1074376.

[16] D. Lim, J.W. Lee, B. Gassend, G.E. Suh, M. van Dijk and S. Devadas, "Extracting secret keys from integrated circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 10, pp. 1200-1205, Oct. 2005. DOI: 10.1109/TVLSI.2005.859470

[17] C. Böhm and M. Hofer, Physically Unclonable Functions in Theory and Practice. Springer-Verlag New York, 2013. DOI:10.1007/978-1-4614-5040-5.

[18] R. Maes, "Physically unclonable functions: Constructions, properties and applications," Springer-Verlag Berlin Heidelberg, 2013. (Vol. 9783642413). DOI: 10.1007/978-3-642-41395-7.

[19] F. Armknecht, R. Maes, A.R. Sadeghi, F.X. Standaert, and C. Wachsmann, "A Formalization of the Security Features of Physical Functions," *2011 IEEE Symposium on Security and Privacy*, Berkeley, CA, 2011, pp. 397-412. DOI: 10.1109/SP.2011.10.

[20] U. Rührmair, S. Devadas and F. Koushanfar, Security Based on Physical Unclonability and Disorder. In: Tehranipoor M., Wang C. (eds) Introduction to Hardware Security and Trust. Springer, New York, NY, 2013. DOI: https://doi.org/10.1007/978-1-4419-8080-9_4.

[21] K. Lofstrom, W.R. Daasch and D. Taylor, "IC identification circuit using device mismatch," *2000 IEEE International Solid-State Circuits Conference. Digest of Technical Papers (Cat. No.00CH37056)*, San Francisco, CA, USA. 2000, pp. 372-373. DOI: 10.1109/ISSCC.2000.839821

[22] P. Tuyls, G.-J. Schrijen, B. Škorić, J. van Geloven, N. Verhaegh, and R. Wolters, Read-Proof Hardware from Protective Coatings. In: Goubin L., Matsui M. (eds) Cryptographic Hardware and Embedded Systems - CHES 2006. CHES 2006. Lecture Notes in Computer Science, vol 4249. Springer, Berlin, Heidelberg. 2006. DOI: https://doi.org/10.1007/11894063_29.

[23] Y. Su, J. Holleman, and B. Otis, A1.6pJ/blt 96% stable chip-ID generating circuit using process variations. *Digest of Technical Papers – 2007 IEEE International Solid-State Circuits Conference.* [4242437] DOI: 10.1109/ISSCC.2007.373466.

[24] D.E. Holcomb, W.P. Burleson, and K. Fu, "Initial SRAM state as a fingerprint and source of true random numbers for RFID tags," Proceedings of the Conference on RFID Security. 2007, 1–12. Retrieved from http://reference.kfupm.edu.sa/content/i/n/initial_sram_state_as_a_fingerprint_and__79228.pdf.

[25] J. Guajardo, S.S. Kumar, G.-J. Schrijen, and P. Tuyls, FPGA Intrinsic PUFs and Their Use for IP Protection. Cryptographic Hardware and Embedded Systems - CHES 2007, 63–80. DOI: https://doi.org/10.1007/978-3-540-74735-2_5

[26] D.E. Holcomb, A. Rahmati, M. Salajegheh, W.P. Burleson, and K. Fu, DRV-Fingerprinting: Using Data Retention Voltage of SRAM Cells for Chip Identification. In: Hoepman JH., Verbauwhede I. (eds) Radio Frequency Identification. Security and Privacy Issues. RFIDSec 2012. Lecture Notes in Computer Science, vol 7739. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/978-3-642-36140-1_12.

[27] P. Prabhu, A. Akel, L.M. Grupp, W.K.S. Yu, G.E. Suh, E. Kan, and S. Swanson, Extracting Device Fingerprints from Flash Memory by Exploiting Physical Variations. In: McCune J.M., Balacheff B., Perrig A., Sadeghi AR., Sasse A., Beres Y. (eds) Trust and Trustworthy Computing. Trust 2011. Lecture Notes in Computer Science, vol 6740. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/978-3-642-21599-5_14.

[28] S. Rosenblatt, S. Chellappa, A. Cestero, N. Robson, T. Kirihata, and S.S Iyer, "A self-authenticating chip architecture using an intrinsic fingerprint of embedded DRAM," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 11, pp. 2934-2943, Nov. 2013. DOI: 10.1109/JSSC.2013.2282114.

[29] P. Koeberl, Ü. Kocabaş and A. R. Sadeghi, "Memristor PUFs: A new generation of memory-based Physically Unclonable Functions," *2013 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Grenoble, France, 2013, pp. 428-431. DOI: 10.7873/DATE.2013.096

[30] S. S. Kumar, J. Guajardo, R. Maes, G. J. Schrijen and P. Tuyls, "Extended abstract: The butterfly PUF protecting IP on every FPGA," *2008 IEEE International Workshop on Hardware-Oriented Security and Trust*, Anaheim, CA, 2008, pp. 67-70. DOI: 10.1109/HST.2008.4559053.

[31] P. Simons, E. van der Sluis and V. van der Leest, "Buskeeper PUFs, a promising alternative to D Flip-Flop PUFs," *2012 IEEE International Symposium on Hardware-Oriented Security and Trust*, San Francisco, CA, 2012, pp. 7-12. DOI: 10.1109/HST.2012.6224311.

[32] C. Brzuska, M. Fischlin, H. Schröder and S. Katzenbeisser, Physically uncloneable functions in the universal composition framework. In: Rogaway P. (eds) Advances in Cryptology – CRYPTO 2011. CRYPTO 2011. Lecture Notes in Computer Science, vol 6841. Springer, Berlin, Heidelberg. 2011. DOI: https://doi.org/10.1007/978-3-642-22792-9_4

[33] Q. Chen, G. Csaba, P. Lugli, U. Schlichtmann and U. Rührmair, "The Bistable Ring PUF: A new architecture for strong Physical Unclonable Functions," *2011 IEEE International Symposium on Hardware-Oriented Security and Trust*, San Diego CA, 2011, pp. 134-141. DOI: 10.1109/HST.2011.595501.

[34] 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," *2004 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No.04CH37525)*, 2004, pp. 176-179. DOI: 10.1109/VLSIC.2004.1346548.

[35] L. Lin, D. Holcomb, D. K. Krishnappa, P. Shabadi and W. Burleson, "Low-power sub-threshold design of secure physical unclonable functions," *2010 ACM/IEEE International Symposium on Low-Power Electronics and Design (ISLPED)*, Austin, TX, USA, 2010, pp. 43-48. DOI: 10.1145/1840845.1840855.

[36] S. Morozov, A. Maiti, and P. Schaumont, An Analysis of Delay Based PUF Implementations on FPGA. In: Sirisuk P., Morgan F., El-Ghazawi T., Amano H. (eds) Reconfigurable Computing: Architectures, Tools and Applications. ARC 2010. Lecture Notes in Computer Science, vol 5992. Springer, Berlin, Heidelberg, 2010. DOI: https://doi.org/10.1007/978-3-642-12133-3_37.

[37] S. Devadas, E. Suh, S. Paral, R. Sowell, T. Ziola and V. Khandelwal, "Design and Implementation of PUF-Based "Unclonable" RFID ICs for Anti-Counterfeiting and Security Applications," *2008 IEEE International Conference on RFID*, Las Vegas, NV, 2008, pp. 58-64. doi: 10.1109/RFID.2008.4519377.

[38] Y. Lao and K. K. Parhi, "Reconfigurable architectures for silicon Physical Unclonable Functions," *2011 IEEE International Conference on Electro/Information Technology*, Mankato, MN, 2011, pp. 1-7. DOI: 10.1109/EIT.2011.5978614.

[39] M. Majzoobi, F. Koushanfar and M. Potkonjak, "Lightweight secure PUFs," *2008 IEEE/ACM International Conference on Computer-Aided Design*, San Jose, CA, 2008, pp. 670-673. DOI: 10.1109/ICCAD.2008.4681648.

[40] B. Gassend, D. Lim, D. Clarke, M. van Dijk and S. Devadas, "Identification and authentication of integrated circuits," Journal Concurrency and Computation: Practice & Experience - Computer Security, Vol. 16 Issue 11, Sep. 2004 , pp 1077 – 1098. DOI: https:.//doi.org/10.1002/cpe.805.

[41] G. E. Suh and S. Devadas, "Physical Unclonable Functions for Device Authentication and Secret Key Generation," *2007 44th ACM/IEEE Design Automation Conference*, San Diego, CA, 2007, pp. 9-14.

[42] A. Maiti and P. Schaumont, "Improving the quality of a Physical Unclonable Function using configurable Ring Oscillators," *2009 International Conference on Field Programmable Logic and Applications*, Prague, 2009, pp. 703-707. DOI: 10.1109/FPL.2009.5272361.

[43] C. E. D. Yin and G. Qu, "LISA: Maximizing RO PUF's secret extraction," *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, Anaheim, CA, 2010, pp. 100-105. DOI: 10.1109/HST.2010.5513105.

[44] J. H. Anderson, "A PUF design for secure FPGA-based embedded systems," *2010 15th Asia and South Pacific Design Automation Conference (ASP-DAC)*, Taipei, 2010, pp. 1-6. DOI: 10.1109/ASPDAC.2010.5419927.

[45] C. Wachsmann and A-R Sadeghi, Physically Unclonable Functions (PUFs). Synthesis Lectures on Information Security, Privacy, and Trust. Morgan & Claypool Publishers. 2004. https://doi.org/10.2200/s00622ed1v01y201412spt012

[46] G. Selimis, M. Konijnenburg, M. Ashouei, J. Huisken, H. De Groot, V. Van Der Leest, G-J. Schrijen, M-v. Hulst, and P. Tuyls, "Evaluation of 90nm 6T-SRAM as Physical Unclonable Function for secure key generation in wireless sensor nodes," *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*, Rio de Janeiro, 2011, pp. 567-570. DOI: 10.1109/ISCAS.2011.5937628.

[47] J. Delvaux and I. Verbauwhede, "Fault Injection Modeling Attacks on 65 nm Arbiter and RO Sum PUFs via Environmental Changes," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 6, pp. 1701-1713, June 2014. DOI: 10.1109/TCSI.2013.2290845.

[48] D. E. Holcomb, W. P. Burleson and K. Fu, "Power-Up SRAM State as an Identifying Fingerprint and Source of True Random Numbers," *IEEE Transactions on Computers*, vol. 58, no. 9, pp. 1198-1210, Sept. 2009. DOI: 10.1109/TC.2008.212.

[49] S. Zeitouni, Y. Oren, C. Wachsmann, P. Koeberl and A. R. Sadeghi, "Remanence Decay Side-Channel: The PUF Case," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1106-1116, June 2016. DOI: 10.1109/TIFS.2015.2512534.

[50] Y. Oren, A.-R. Sadeghi dan C. Wachsmann, On the Effectiveness of the Remanence Decay Side-Channel to Clone Memory-Based PUFs. In: Bertoni G., Coron JS. (eds) Cryptographic Hardware and Embedded Systems - CHES 2013. CHES 2013. Lecture Notes in Computer Science, vol 8086. Springer, Berlin, Heidelberg. 2013. DOI: https://doi.org/10.1007/978-3-642-40349-1-7

[51] C. Tarnovsky, Hacking the Smartcard Chip, In Blackhat Decipher Security 2010.

[52] S. Katzenbeisser, Ü. Kocabas. V.Rožić, A-R. Sadeghi, I. Verbauwhede and C. Wachsmann. (2012). PUFs: Myth, fact or busted? A security evaluation of Physically Unclonable Functions (PUFs) cast in silicon. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). DOI: https://doi.org/10.1007/978-3-642-33027-8_17

[53] C. Helfmeier, C. Boit, D. Nedospasov and J. P. Seifert, "Cloning Physically Unclonable Functions," *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, Austin, TX, 2013, pp. 1-6. DOI: 10.1109/HST.2013.6581556.

[54] D. Nedospasov, J. P. Seifert, C. Helfmeier and C. Boit, "Invasive PUF Analysis," *2013 Workshop on Fault Diagnosis and Tolerance in Cryptography*, Santa Barbara, CA, 2013, pp. 30-38. DOI: 10.1109/FDTC.2013.19

[55] Y.S. Abu-Mostafa, M. Magdon-Ismail and H.-T. Lin, Chapter 5 - Three Learning Principles. In Learning From Data, AMLBook, 2012.

[56] I. Saha, R. R. Jeldi and R. S. Chakraborty, "Model building attacks on Physically Unclonable Functions using genetic programming," *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, Austin, TX, 2013, pp. 41-44. DOI: 10.1109/HST.2013.6581563.

[57] U. Rührmair and J. Sölter, "PUF modeling attacks: An introduction and overview," *2014 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Dresden, 2014, pp. 1-6. DOI: 10.7873/DATE.2014.361.

[58] R. E. Schapire, The Boosting Approach to Machine Learning: An Overview. In: Denison D.D., Hansen M.H., Holmes C.C., Mallick B., Yu B. (eds) Nonlinear Estimation and Classification. Lecture Notes in Statistics, vol 171. Springer, New York, NY. DOI: https://doi.org/10.1007/978-0-387-21579-2_9

[59] D. Karakoyunlu and B. Sunar, "Differential template attacks on PUF enabled cryptographic devices," *2010 IEEE International Workshop on Information Forensics and Security*, Seattle, WA, 2010, pp. 1-6. DOI: 10.1109/WIFS.2010.5711445.

[60] D. Merli, D. Schuster, F. Stumpf, and G. Sigl, Side-Channel Analysis of PUFs and Fuzzy Extractors. In: McCune J.M., Balacheff B., Perrig A., Sadeghi AR., Sasse A., Beres Y. (eds) Trust and Trustworthy Computing. Trust 2011. Lecture Notes in Computer Science, vol 6740. Springer, Berlin, Heidelberg, 2011. https://doi.org/10.1007/978-3-642-21599-5-3

[61] A. Mahmoud, U. Rührmair, M. Majzoobi and F. Koushanfar, "Combined Modeling and Side Channel Attacks on Strong PUFs.," IACR Cryptology ePrint Archive, 2013, 632.

[62] J. Capovilla, M. Cortes and G. Araujo, "Improving the Statistical Variability of Delay-based Physical Unclonable Functions," Proceedings of the 28th Symposium on Integrated Circuits and Systems Design, SBCCI '15, Salvador, Brazil — August 31 - September 04, 2015 Article No. 41. DOI: 10.1145/2800986.2801010.

[63] Jin Miao, Meng Li, S. Roy and Bei Yu, "LRR-DPUF: Learning resilient and reliable digital physical unclonable function," *2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, Austin, TX, 2016, pp. 1-8. DOI: 10.1145/2966986.2967051

[64] R. Kumar and W. Burleson, Side-Channel Assisted Modeling Attacks on Feed-Forward Arbiter PUFs Using Silicon Data. In: Mangard S., Schaumont P. (eds) Radio Frequency Identification. RFIDSec 2015.Lecture Notes in Computer Science, vol 9440. Springer,Cham.DOI: https://doi.org/10.1007/978-3-319-24837-0_4.

[65] Y. Gao, G. Li, G., H. Ma, S.F. Al-sarawi, O. Kavehei, D. Abbott and D.C. Ranasinghe, "Obfuscated challenge-response: A secure lightweight authentication mechanism for PUF-based pervasive devices," *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, Sydney, NSW, 2016, pp. 1-6. DOI: 10.1109/PERCOMW.2016.7457162.

[66] D. Merli, G. Sigl and C. Eckert, Identities for Embedded Systems Enabled by Physical Unclonable Functions. In: Fischlin M., Katzenbeisser S. (eds) Number Theory and Cryptography. Lecture Notes in Computer Science, vol 8260. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/978-3-642-42001-6_10.

[67] J. Ye, Y. Hu and X. Li, "OPUF: Obfuscation logic based physical unclonable function," *2015 IEEE 21st International On-Line Testing Symposium (IOLTS)*, Halkidiki, 2015, pp. 156-161. DOI: 10.1109/IOLTS.2015.7229850.

[68] J. R. Wallrabenstein, "Practical and Secure IoT Device Authentication Using Physical Unclonable Functions," *2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud)*, Vienna, 2016, pp. 99-106. DOI: 10.1109/FiCloud.2016.22.

[69] J. Tobisch and G.T. Becker, On the Scaling of Machine Learning Attacks on PUFs with Application to Noise Bifurcation. In: Mangard S., Schaumont P. (eds) Radio Frequency Identification. RFIDSec 2015. Lecture Notes in Computer Science, vol 9440. Springer, Cham. DOI: https://doi.org/10.1007/978-3-319-24837-0_2.

[70] M. v. Dijk and U. Rührmair, "PUF Interfaces and their Security," *2014 IEEE Computer Society Annual Symposium on VLSI*, Tampa, FL, 2014, pp. 25-28. DOI: 10.1109/ISVLSI.2014.90.

[71] M. D. Yu, M. Hiller, J. Delvaux, R. Sowell, S. Devadas and I. Verbauwhede, "A Lockdown Technique to Prevent Machine Learning on PUFs for Lightweight Authentication," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 2, no. 3, pp. 146-159, July-Sept. 1 2016. DOI: 10.1109/TMSCS.2016.2553027.