IEEE Copyright Notice

Machine Learning based IoT Edge Node Security Attack and Countermeasures

Vishalini R. Laguduva, Sheikh Ariful Islam, Sathyanarayanan Aakur, Srinivas Katkoori and Robert Karam

Department of Computer Science and Engineering

University of South Florida

Tampa, FL 33620

{vishalini, sheikhariful, saakur, katkoori, rkaram}@mail.usf.edu

Abstract—Advances in technology have enabled tremendous progress in the development of a highly connected ecosystem of ubiquitous computing devices collectively called the Internet of Things (IoT). Ensuring the security of IoT devices is a high priority due to the sensitive nature of the collected data. Physically Unclonable Functions (PUFs) have emerged as critical hardware primitive for ensuring the security of IoT nodes. Malicious modeling of PUF architectures has proven to be difficult due to the inherently stochastic nature of PUF architectures. Extant approaches to malicious PUF modeling assume that *a priori* knowledge and physical access to the PUF architecture is available for malicious attack on the IoT node. However, many IoT networks make the underlying assumption that the PUF architecture is sufficiently tamper-proof, both physically and mathematically. In this work, we show that knowledge of the underlying PUF structure is not necessary to clone a PUF. We present a novel non-invasive, architecture independent, machine learning attack for strong PUF designs with a cloning accuracy of 93.5% and improvements of up to 48.31% over an alternative, two-stage brute force attack model. We also propose a machine-learning based countermeasure, *discriminator*, which can distinguish cloned PUF devices and authentic PUFs with an average accuracy of 96.01%. The proposed discriminator can be used for rapidly authenticating millions of IoT nodes remotely from the cloud server.

Fig. 1. Typical IoT architecture is illustrated here. The dashed lines indicate IoT device authentication protocols. IoT device enrollment is done by populating and authenticating CRPs "K" times.

I. INTRODUCTION AND MOTIVATION

Evolution of technology has resulted in a highly connected ecosystem of ubiquitous computing devices that work together seamlessly to collect, process and analyze large amounts of data to aid in human-centric decision making. Collectively called the Internet of Things (IoT), the collection of wearable devices, sensors and embedded systems (to name a few) have enabled automated decision making for improving quality of life. Given the highly integrated nature of IoT devices, adversarial attacks can lead to high levels of security and trust issues. Ensuring the security of IoT devices is a high priority due to the sensitive nature of the collected data [1], [2]. However, this comes with it a set of challenges: (1) IoT devices are typically resource-constrained, thus requiring high energy efficient security protocols; (2) their highly distributed nature can provide easy physical access to the node and (3) the highly connected nature of IoT framework requires fast and secure security protocols.

Traditional approaches to cryptography, while effective, have not proven to be sufficiently lightweight and fast for IoT device authentication. Thus, hardware-based security protocols have emerged as viable alternatives for IoT device registration and authentication. Recent efforts have shifted to leveraging the inherent randomness induced in silicon devices during the manufacturing process as the secret key, opposed to the traditional binary key stored in silicon devices, which can be susceptible to physical attacks. Such approaches, called Physically Unclonable Functions (PUFs), have helped provide a higher level of security against direct physical attacks. This alleviates the need for costly physical protection measures. PUFs have become increasingly popular and have been used for IoT device authentication [3]–[7] and other security tasks [8], [9].

Silicon-based PUF devices [8] are easily fabricated, physical structures that leverage the stochastic nature of the manufacturing process to create physically unclonable, unique identifiers for each manufactured unit. This typically results in a one-way function. Given an electronic stimulus, the response of a PUF device is an unpredictable, repeatable function. This response identifies each device with a unique signature. This interaction is termed as the Challenge-Response Pair (CRP), where the challenge is the external stimulus and the PUF's reaction is termed as the response.

Using PUFs for IoT security protocols typically involves an initial enrollment phase and an authentication protocol during the actual data exchange. Figure 1 illustrates the typical

architecture of an IoT network and the generic enrollment protocol. A typical IoT network consists of remote, resource-constrained data nodes $(N_1, N_2, N_3 \ldots N_k)$ connected to static server nodes $(S_1, S_2, S_3 \ldots S_n)$ that transfer the acquired data to the cloud using routers $(R_1, R_2, R_3 \ldots R_m)$. The data is transmitted from the routers to the cloud using a network gateway. IoT edge nodes can range from simple sensors to complex systems with processor, memory, communication etc. Strong PUFs implemented in complex IoT nodes are subject to attacks, which is the focus of this work. When a data node is added to the IoT network, an enrollment phase is executed to create a CRP database for the PUF within the data node. This database of CRPs is used in the authentication phase when two nodes corresponding to the same server node want to communicate. The common server node authenticates both data nodes, generates security key pairs and helps secure key sharing.

Security Assumptions: Following the protocols established in [10], extant IoT networks using PUF authentication [3]–[7] make the following underlying assumptions: (1) cloning a PUF architecture, either physically or mathematically is a difficult problem, (2) an adversary has unrestricted physical access to the communication channel, (3) the challenge-response characteristics of the PUF within the data IoT node is an implicit property and is not accessible to an adversary and (4) the attacker can obtain access to the database of CRPs through malicious software attacks, though explicit knowledge of the secret keys. Given these security assumptions, the goal of the adversary becomes straightforward: it must be able to spoof the server nodes into accepting a malicious node on behalf of the original data nodes without actual possession of the node in question. Any physical intrusions can compromise the integrity of the PUF and hence render the attack futile. The underlying stochastic nature of PUFs and the above constraints lend itself to a solid security protocol that can be hard to breach. However, advances in machine learning have led to a vast majority of the non-invasive attacks explored. Machine learning-based approaches can be characterized by the application of a learned mathematical model on a collected subset of valid CRPs. The curation of such data is typically assumed to be an eavesdropping protocol. Prior works, especially the pioneering work of Rühmair et al [11], have shown great success in cloning PUFs, gaining cloning accuracy of up to 99.99%. Such success does come with a caveat - the underlying architecture must be known *a priori*, either through invasive physical intrusions or explicit architecture knowledge.

Today's IoT nodes are designed such that they are tamper-proof [12] [13] which makes it difficult or impossible for micro-probing. Even if the attacker is successful in micro-probing, given the myriad of PUF architectures in literature, extracting information on the underlying PUF architecture is extremely difficult. Hence, earlier ML-based PUF attacks with the assumption of knowing underlying architecture are either not practical or extremely difficult to stage. In this work, we present, for the first time, an ML-based attack that does not require PUF architecture information. We also present a countermeasure for this attack that can be effectively used to remotely evaluate an IoT node's trust level.

To overcome such limitations, we focus on an architecture independent attack, that assumes no prior knowledge of the PUF architecture in the system. We show that observed CRPs are sufficient to improve cloning accuracy of a strong PUF irrespective of the underlying architecture. The attack can simulate PUF network without knowing underlying PUF architecture. To evaluate the effectiveness of our approach, we compare against a brute force attack model (Section III) that leverages the current advances in PUF-architecture cloning. We leverage architecture-specific-cloning [11] through a cascaded framework of (1) PUF architecture identification, (2) employ architecture specific cloning models, and (3) evaluate the prediction accuracy of the model by combining the architecture classification accuracy and the cloning accuracy in a harmonic mean.

Inspired from the pioneering work of Goodfellow *et al* [14] on Generative Adversarial Networks (GANs), we propose a machine learning-based defense, a *discriminator*, to identify the possibility of cloning using any ML-based attack non-invasive attack. Extant countermeasures [15], [16] to ML-based cloning have focused on creating complex, cloning resistant PUF architecture. As we enter into a more realizable IoT ecosystem, complex PUF architectures may not be suitable for lightweight IoT systems. Hence, we propose a lightweight, probabilistic identification of cloning through machine learning. To the best of the authors' knowledge, this is the first such framework for the non-invasive attack of PUF-based IoT network authentication schemes and a proposed mechanism to differentiate original PUFs from cloned ones. In short, our paper makes the following novel contributions:

- propose a non-invasive, architecture independent cloning attack on string PUFs,
- show that a brute force attack on strong PUFs to identify the PUF architecture for cloning is increasingly complex and hence not trivial for feasible cloning, and
- propose a probabilistic, discriminator model to bolster the security of the CRP protocol by identifying possible instances of cloning attacks.

The rest of this paper is organized as follows. We briefly review extant machine learning attacks on PUFs and corresponding countermeasures in Section II. We describe and evaluate a baseline, brute-force approach in Section III, followed by a description of the proposed attack and discriminator approach in Section IV. We present our empirical evaluation of the proposed approach in Section V and conclude with a discussion on the proposed approach in Section VI.

II. BACKGROUND AND RELATED WORK

In this section, we briefly summarize extant work on machine-learning based attack and prevention techniques in the strong PUF design.

Strong PUFs: A strong PUF can support a large number of complex CRPs with physical access to the PUF for a query such that an attacker cannot generate correct response given finite resources and time [17]–[19]. While a weak PUF has only few CRPs which makes it difficult for the attack

Fig. 2. The proposed attack model on oblivious PUFs architecture. The brute force attack has an additional PUF architecture detection process as indicated by the block in red.

PUF Model	PUF Classification Rate (%)	Cloning Rate (%)
APUF	81.49%	77.42%
3 XOR APUF	76.53%	72.71%
4 XOR APUF	65.01%	61.76%
5 XOR APUF	63.57%	60.39%
6 XOR APUF	61.31%	58.25%
LW 3 XOR APUF	76.91%	73.05%
LW 4 XOR APUF	65.37%	62.10%
LW 5 XOR APUF	59.32%	56.33%

and prediction techniques, hence in this paper we consider strong PUF. The number of CRPs of strong PUFs can grow exponentially depending on the number of module blocks available for generating responses for a large number of corresponding challenges. Error due to noise in the response of PUF can be minimized using helper data [20], [21]. For a detailed analysis of constructions and description of strong PUFs, we refer the reader to [17].

ML Attacks on PUFs: Rührmair *et al.* [11] proposed an ML-based attack on strong PUFs based on a predictive model. The authors were able to clone the functionality of the underlying PUF given the PUF model by evaluating model parameters using LR with RProp and ES. Though the method was quite successful in cloning, the attacker needs to know the underlying PUF architecture and the corresponding signature function. While it is reasonable to assume that CRPs can be obtained by eavesdropping or other interfaces [18], it is not always possible to ascertain the underlying PUF model without physical access to the PUF. There have also been other approaches such as PAC [22] and hybrid methods [23] that have successfully cloned PUFs using a combination of ML and invasive techniques.

ML resistant PUFs: The linear additive behavior of Arbiter PUF (APUF) has made it an ideal target for ML attack. Hence, higher non-linearity in a given PUF architecture can improve the uniqueness and randomness with increased defense against modeling attack. Other approaches to ML resistant PUFs have been randomized challenges [24], obfuscation [15], sub-string based challenges [16].

III. BRUTE FORCE ATTACK ON STRONG PUFS

The success of the proposed models by Rühmair et al [11] allow us to successfully clone strong PUF models with a prediction accuracy of 99.9%. However, to use it in a non-invasive manner, we would first need to identify the underlying PUF architecture, as the approaches in [11] require intimate knowledge of the PUF architecture such as PUF type, number of stages and number of XOR gates, to name a few.

To address this, we propose the use of a machine learning model to identify the PUF architecture through observation of

the challenge-response pairs, as illustrated in Figure 2. One major assumption in this approach is that there exists a subset of challenges $\tilde{C} \in C$ that is valid for all PUF architectures in a given network, where C is the collection of all valid CRPs. Given the number of PUF architectures and their use for authentication, this is not an unreasonable assumption.

A. Identifying PUF architectures

Given the set of challenges $\tilde{C}$, we can observe the set of valid responses R_{c_i} for each PUF architecture $c_i \in C_{puf}$, where C_{puf} is the set of all known PUF architectures described in Section II. Hence, the objective of the classification is to learn a function f_c which maximizes the probability

$$\arg\max_{\tilde{C}_i \in \tilde{C}} P(c_i|\tilde{C}_i, R_{c_i}) \qquad (1)$$

where the objective is the find the PUF architecture c_i given the challenge $\tilde{C}_i$, and the subsequent response R_{c_i}. We use the following machine learning models as the basis for the function $f_c(\cdot)$: logistic regression, artificial neural network, and random forests.

B. Empirical Evaluation

We evaluate the performance of the proposed brute force attack to identify the architecture of eight (8) common strong PUF architectures. We use a fixed number of randomly sampled 100 CRPs for evaluation for each PUF architecture for a total of 800 CRPs. We report average results from 5 different runs, with the test set sampled randomly each time. We curate a set of $100,000$ CRPs for training the classification model.

As can be seen from Table I, identifying the PUF architecture from an observed set of CRPs is not a trivial task. Even with 100% cloning accuracy for a given PUF architecture, identifying the said architecture requires a large set of CRPs for training a model. The maximum performance that we were able to obtain was using the logistic regression model, which took 100 iterations to converge resulting in the maximum classification rate for Arbiter PUF architecture. There was a large confusion among different design variations of each PUF type. The prediction rate for XOR PUFs decreased as the complexity of the architecture increased.

While the average cloning accuracy can be as high as 77.42% (for the Arbiter PUF), the numbers can be misleading in practice. The performance of the two-stage attack model is rather low, considering the practical gap between the intra

and inter Hamming Distance of PUF CRPs, this prediction rate cannot be considered to be successful cloning.

IV. ARCHITECTURE INDEPENDENT PUF MODELING

In this section, we describe our proposed approach for a PUF-independent attack model on various PUF architectures by exploiting the CRP authentication protocol. We begin with a discussion on the use of machine learning models to capture the underlying correlation between challenge-response pairs to model the randomness unique to a given PUF architecture. We then follow with a discussion on defending against such attacks using complementary machine learning models.

A. Attack Model

Each PUF is made unique through a digital signature characterized by its response to a given challenge. This signature is representative of the randomness encoded in its state due to manufacturing variations and other physical disorders. In order to compromise the integrity of the CRP protocol, one has to model this randomness to generate a response representative of the PUF's signature. There are two approaches to this problem: a model-based solution and a model-agnostic solution. The model-based solution, explored in [11], attempted to capture this randomness through modeling the characteristics of a PUF using domain knowledge (PUF architecture) and characteristics (delay model, thermal response characteristics, etc.). Thus, the attack consists of regression of the model's parameters.

We, however, consider an architecture independent approach to the solution by disregarding the need for a characteristic equation for the PUF. We postulate that the challenge and subsequent response of any given PUF is representative of its characteristic function. Thus, modeling the dependency between the various features of a given challenge along with the target response allows us to capture the randomness of a given PUF architecture. To this end, we use several approaches to capture the dependency between the challenge and response pairs of various PUF architectures. Since the underlying dependency is not known to be linear or non-linear, we explore several different machine learning models that characterize the dependency with a linear decision boundary (logistic regression) or with a non-linear decision boundary (random forest and artificial neural networks).

The attack model consists of learning the optimal function that maps the given n-bit challenge $C = c_1, c_2, \ldots, c_n$ to an appropriate output response $R \in \{-1, 1\}$ with a probability $p(R|C)$. The objective of the attack model is to learn the function $f : C \rightarrow R$ such that the difference between the generated and actual response of the PUF is minimized. Hence the best attack model is characterized by the search for the optimal function f given by

$$\underset{(C_s, R_s)}{\arg\min} E[(\hat{f}(C) - f(C))^2] \qquad (2)$$

where $\hat{f}(C)$ is the characteristic function of the given PUF architecture and (C_s, R_s) represents the space of all known challenge-response pairs obtained through the eavesdropping protocol. We search for the optimal function $f(C)$ through the characteristic equation of the different machine learning

Fig. 3. ML-based discriminator model to ascertain a PUFs integrity

models defined above. For example, in a logistic regression model, f is defined as

$$f = \arg\max(\sigma(R \times d(\vec{w}, C))) \qquad (3)$$

where $\vec{w}$ is a learned vector that represents the decision boundary (d) for the logistic regression model and σ is the logistic function.

B. Discriminator Model

The modeling of the internal randomness of a given PUF architecture puts the integrity of the CRP-based authentication into question. Hence, it becomes critical that we are able to differentiate between the original PUF and an adversarial attack, such as one described in Section IV-A. To this end, we introduce a mathematical model that is able to discriminate between an original and a cloned PUF called the *discriminator model*, as illustrated in figure 3. The discriminator decides whether each instance of the response belongs to the actual PUF or a malicious attacker. As seen in figure 3, the discriminator model takes in the response of the original PUF along with the response of the PUF cloned with several ML attacks as the input to predict whether the PUF is an original or a cloned and returns the probabilities. The cloned part of the response is shown in red. The output of this discriminator is a single scalar value $D(C)$, indicative of an adversarial attack. The value $D(C)$ is a probability function that maps a given response (R) to the distribution belonging to either the original PUF ($\hat{f}(C)$) or an attacker ($f(C)$) for a given n-bit challenge C. Hence, the optimal discriminator model is given by

$$D^\star(C, R) = \frac{p(\hat{f}(C)}{p(f(C)) + p(\hat{f}(C))} \qquad (4)$$

where $D^\star(C, R)$ is a mathematical model that maps the response R for a given challenge (C) into the probability space of either the original PUF ($\hat{f}(.)$) or the attack model ($f(.)$). Again, we explore the use of well-known machine learning models as the basis for our discriminator mathematical model.

The search space for the optimal discriminator is similarly characterized by the optimization function defined in Equation 2. However, the search is represented by the discriminator to distinguish between the original PUF's response and an adversarial attack.

C. Search for Optimal Attack-Discriminator Model

The search space for the optimal attack model and discriminator model is defined by the optimizer functions defined in Equation 2 and its subsequent adaptation for the discriminator, respectively. We employ a simple grid search algorithm to find

PUF Model	Cloning Error (%)	Discriminator Error (%)	Cloning Time
APUF	6.50%	12.66%	0.002 sec
3 XOR APUF	8.20%	1.18%	70:85 sec
4 XOR APUF	10.70%	4.03%	1:38 min
5 XOR APUF	9.00%	3.84%	62:48 min
6 XOR APUF*	10.70%	0.50%	240 min
LW 3 XOR APUF	12.00%	8.66%	1:59 sec
LW 4 XOR APUF	12.50%	5.22%	30:58 min
LW 5 XOR APUF*	17.00%	3.69%	180 min
Average	10.83%	4.97%	

*Note that in the literature [11] [9], the maximum number of XORs used is 6. It is known that 6 XORs is sufficient to give a strong PUF.

the optimal attack model ($f(.)$) from a given set of possible models (F). The attack models space, F comprises of all transformation functions that satisfy the condition $f : C \rightarrow R$. We restrict the search space to the given three machine learning models: Logistic Regression (LR), Random Forest (RF), and Neural Network (NN). We also ensure that the optimal discriminator is chosen from a set of discriminative function $G(.) \in G_s$, where G_s is the collection of all discriminative functions that optimize the probability function defined in Equation 4. Again, we restrict the search space to the three aforementioned models. While the grid search suffers from the curse of dimensionality and does not scale to large search spaces of F and G_s, limiting the number of plausible functions allows us to exhaustively search for the optimal discriminator for a given attack model and a target PUF. Additionally, the grid search is a reasonable approach given that it can be embarrassingly parallel.

V. EXPERIMENTAL RESULTS AND DISCUSSION

Following experimental setup by [11], we report the upper bound of attacker and discriminator accuracy in a supervised setting. We consider three strong PUF architectures (Arbiter, XOR and Lightweight), while each of them contains three stages (64, 128, and 256) and the number of XOR is limited to (3, 4, and 5) for both XOR and Lightweight PUFs. This gives us a total of 24 different strong PUF architectures for validating the efficacy of the proposed method.

The average cloning error, the discriminator error and the cloning time is shown in Table II. We report average results over 5 different runs for each PUF architecture. From Table II, we can see that on average, a strong PUF can be cloned with a cloning error of 10.83% irrespective of its underlying architecture of the PUF. The discriminator error shows that the obtained response is either from the original PUF or from the cloned PUF with an error of 4.97%. The aging of the PUF [25] affects the delay characteristic which produces a different pattern of the responses compared to the compromised node. It can seen that the cloning time is reasonable, particularly given the complexity and stochastic nature of the considered PUFs. We discuss the performance of our approach on different PUF architectures in detail below.

Arbiter PUF: As seen from Figures 4(a) and 5(a), modeling attack on the exact number of stages for arbiter PUF can be accurately guessed (93.5%) for a combination of NN(CM) and LR(DM). However, with a single ML model (RF), the discrim-

inator prediction accuracy improves to (94.4%) compared with combined NN(CM) and LR(DM).

XOR PUF: Figures 4(b) and 5(b) show the ML models performance for three (3) XOR PUF. While the combined NN(CM) and NN(DM) models are capable of capturing the PUF parameters 91.8% of the time, we are able to make use of combined NN(CM) and LR(DM) to get discriminator accuracy up to 98.8%. It can be seen from Figures 4(c) and 5(c) that the single NN model performance is markedly higher (89.3%) in modeling attack compared to all other combinations for four (4) XOR PUF. Similarly, the combined NN(CM) and LR(DM) outperforms others in discriminator prediction with 95.9% accuracy. As shown in Figures 4(d) and 5(d), our trained single ML model (LR) notably improves modeling attack accuracy about 91.0% for five (5) XOR PUF. The discriminator accuracy improves up to 96.2% for combined RF(CM) and NN(DM). Finally, for six (6) XOR PUF in Figures 4(e) and 5(e), the combined LR(CM) and RF(DM) can achieve up to 89.3% and 100% for modeling the attack and discriminator, respectively.

Lightweight PUF (LPUF): For LPUF, we evaluate our method on 3-, 4-, and 5-XOR PUF. With supervised experiment for LPUF with three (3) XOR, the single LR model can achieve modeling accuracy up to 88.0% while that for discriminator, the single RF outperforms all other combinations by 91.7% as shown in Figures 4(f) and 5(f). Figures 4(g) and 5(g) show modeling and prediction accuracy for LPUF with four (4) XOR. In this case, the single NN improves cloning accuracy by 87.5% and the combined LR(CM) and RF(DM) improves the discriminator performance by 94.8%. Finally, we apply the proposed method for LPUF with five (5) XOR as shown in Figures 4(h) and 5(h). For a single NN model, the cloning accuracy is 83.0% and the discriminator prediction can improve by 96.3% the combined RF(CM) and LR(DM).

VI. CONCLUSION

In this work, we introduced an efficient architecture-independent machine learning based approach for cloning strong PUFs. We also introduce a novel discriminator model to identify cloned and original PUFs with a high degree of confidence. We also introduce a search-based approach for identifying the optimal discriminator model for a given cloned PUF using three common ML models. Extensive experiments show the efficacy of the proposed approach. For future work, we will extend this method for control PUFs and explore ensemble meta-algorithms.

REFERENCES

[1] N. Cam-Winget, A. Sadeghi, and Y. Jin. Can IoT be secured: Emerging challenges in connecting the unconnected. In *Proceedings of the 53rd Annual Design Automation Conference*, page 122. ACM, 2016.

[2] S. Ray, S. Bhunia, Y. Jin, and M. Tehranipoor. Security validation in IoT space. In *2016 IEEE 34th VLSI Test Symposium (VTS)*, pages 1–1. IEEE, 2016.

[3] U. Chatterjee, R. S. Chakraborty, and D. Mukhopadhyay. A PUF-based secure communication protocol for IoT. *ACM Transactions on Embedded Computing Systems (TECS)*, 16(3):67, 2017.

[4] U. Chatterjee, V. Govindan, R. Sadhukhan, D. Mukhopadhyay, R. S. Chakraborty, D. Mahata, and M. M. Prabhu. Building PUF based Authentication and Key Exchange Protocol for IoT without Explicit CRPs in Verifier Database. *IEEE Transactions on Dependable and Secure Computing*, 2018.

Fig. 4. Comparison of cloning and discriminator accuracy for different PUFs architecture with combinations of ML models. Single bar represents average accuracy for 64, 128, and 256 stages. Along X-axis, X(Y) defines X model is used for Y task where Y can be cloning (CM) or discriminator (DM).

Fig. 5. Comparison of cloning and discriminator accuracy for different PUFs architecture under single ML model. Single bar represents average accuracy for 64, 128, and 256 stages. Along X-axis, X(Y) defines only X model is used for Y where Y defines both cloning (CM) and discriminator (DM) tasks.

[5] M. N. Aman, S. Taneja, B. Sikdar, K. C. Chua, and M. Alioto. Token-based security for the Internet of Things with dynamic energy-quality tradeoff. *IEEE Internet of Things Journal*, 2018.

[6] M. N. Aman, K. C. Chua, and B. Sikdar. Hardware Primitives-Based Security Protocols for the Internet of Things. In *Cryptographic Security Solutions for the Internet of Things*, pages 117–141. IGI Global, 2019.

[7] A. Braeken. PUF Based Authentication Protocol for IoT. *Symmetry*, 10(8):352, 2018.

[8] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld. Physical One-Way Functions. *Science*, 297(5589):2026–2030, 2002.

[9] U. Rührmair. Oblivious transfer based on physical unclonable functions. In Alessandro Acquisti, Sean W. Smith, and Ahmad-Reza Sadeghi, editors, *Trust and Trustworthy Computing*, pages 430–440, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[10] R. Ostrovsky, A. Scafuro, I. Visconti, and A. Wadia. Universally composable secure computation with (malicious) physically uncloneable functions. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 702–718. Springer, 2013.

[11] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, and J. Schmidhuber. Modeling Attacks on Physical Unclonable Functions. In *Proceedings of the 17th ACM Conference on Computer and Communications Security*, CCS '10, pages 237–249, New York, NY, USA, 2010. ACM.

[12] Y. Ishai, M. Prabhakaran, A. Sahai, and D. Wagner. Private circuits II: keeping secrets in tamperable circuits. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 308–327. Springer, 2006.

[13] K. Yang, D. Forte, and M. Tehranipoor. Protecting endpoint devices in IoT supply chain. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pages 351–356. IEEE Press, 2015.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, Bing Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[15] M. S. Mispan, B. Halak, and M. Zwolinski. Lightweight obfuscation techniques for modeling attacks resistant PUFs. In *2017 IEEE 2nd International Verification and Security Workshop (IVSW)*, pages 19–24, July 2017.

[16] M. Rostami, M. Majzoobi, F. Koushanfar, D. S. Wallach, and S. Devadas. Robust and Reverse-Engineering Resilient PUF Authentication and Key-Exchange by Substring Matching. *IEEE Transactions on Emerging Topics in Computing*, 2(1):37–49, March 2014.

[17] C. Herder, M. D. Yu, F. Koushanfar, and S. Devadas. Physical Unclonable Functions and Applications: A Tutorial. *Proceedings of the IEEE*, 102(8):1126–1141, Aug 2014.

[18] U. Rührmair and D. E. Holcomb. PUFs at a glance. In *2014 Design, Automation Test in Europe Conference Exhibition (DATE)*, pages 1–6, March 2014.

[19] S. A. Islam and S. Katkoori. High-level synthesis of key based obfuscated RTL datapaths. In *2018 19th International Symposium on Quality Electronic Design (ISQED)*, pages 407–412, March 2018.

[20] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In Christian Cachin and Jan L. Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004*, pages 523–540, Berlin, Heidelberg, 2004.

[21] S. A. Islam, L. K. Sah, and S. Katkoori. Empirical word-level analysis of arithmetic module architectures for hardware trojan susceptibility. In *2018 Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*, pages 109–114, Dec 2018.

[22] F. Ganji, S. Tajik, F. Fler, and J. P. Seifert. Strong machine learning attack against pufs with no mathematical model. Cryptology ePrint Archive, Report 2016/606, 2016. https://eprint.iacr.org/2016/606.

[23] U. Rührmair, X. Xu, J. Sölter, A. Mahmoud, F. Koushanfar, and W. Burleson. Power and Timing Side Channels for PUFs and their Efficient Exploitation. Cryptology ePrint Archive, Report 2013/851, 2013. https://eprint.iacr.org/2013/851.

[24] J. Ye, Y. Hu, and X. Li. RPUF: Physical Unclonable Function with Randomized Challenge to resist modeling attack. In *2016 IEEE Asian Hardware-Oriented Security and Trust (AsianHOST)*, pages 1–6, Dec 2016.

[25] S. Meguerdichian and M. Potkonjak. Device aging-based physically unclonable functions. In *2011 48th ACM/EDAC/IEEE Design Automation Conference (DAC)*, pages 288–289. IEEE, 2011.