Weighted Group Decision Making Using Multi-identity Physical Unclonable Functions

Lake Bu and Michel A. Kinsy
Adaptive and Secure Computing Systems Laboratory
Department of Electrical and Computer Engineering, Boston University
Email: {bulake, mkinsy}@bu.edu

***Abstract*—To enable next-generation distributed and connected computing systems, we must address the context-aware chip authentication challenge. An important remaining gap in the design of these systems is the enabling of multi-personality authentication to support applications or schemes requiring a single device to own manifold legitimate identities. In this work, we propose a Multi-identity Physical Unclonable Function (Mi-PUF) assisted weighted group decision making scheme. The Mi-PUF approach enables individual devices to be authenticated and associated with multiple identities in order to hold different number of ballots. Hence, devices with higher impact in a decision making network will have more weight than the less influential ones. Besides the introduction of the scheme, the design and FPGA implementation details of the Mi-PUF are explored and presented.**

***Keywords*-weighted decision making; context-aware; PUF; multi-identity.**

I. Introduction

A physical unclonable function (PUF) is a piece of hardware that produces unpredictable responses upon challenges due to its manufacturing variations [1]. It provides an inexpensive and device-integrated approach for device authentication and identification. Besides the common use of PUFs to authenticate a single device as a single entity, there is a need to perform more fine-grained context-aware authentications. For instance, a device should be able to hold multiple identities and use whichever is appropriate for the given context.

One such context is group decision making using threshold secret sharing. At the beginning, each party is given a ballot uniquely linked to its identity (ID). The ballots are then collected and counted. If the number of collected ballots reaches a certain threshold, the authorization token or message can be reconstructed and the group can carry out a collective decision [2]. This problem is particularly interesting in the hierarchical group setting where parties may need to hold different numbers of ballots to demonstrate their overall influence on the group decision [3]. Due to the mathematical underpinning of the problem, a device may be required to verify with more than one ID, in order to be given more than one ballot.

To address the issue, we propose a weighted decision making scheme assisted by Multi-identity PUF (Mi-PUF). The Mi-PUF enables a single device to be authenticated and identified with multiple legal identities. With this property, depending on a device's influential or trustworthy level, the decision making scheme can respectively compute and link multiple ballots to those IDs owned by one device. Thus, the final decision can be achieved in a weighted manner.

The major contributions of this paper are:

1) It proposes a Mi-PUF assisted group decision making scheme integrating device authentication, identification, and weighted voting in one protocol;
2) It introduces a design of Mi-PUF compatible with both weak and strong delay PUFs;
3) It explores the key details of Mi-PUF's FPGA implementation using the Xilinx's Vivado IDE 2018.2.

II. Group Decision Making Based on Threshold Secret Sharing

Group decision making or voting based on threshold secret sharing was first proposed by Berry Schoenmakers [2] and then studied by a number of researchers [3], [4], [5]. Briefly speaking, the general concept consists of two stages: 1) each voter is authenticated and identified with a unique ID by a verifier, who splits an authentication token into multiple ballots according to the voters' IDs and distributes them; 2) any voter that is supportive of the decision submits its ID and ballot. If the number of ballots reaches a certain threshold, the authorization token can be reconstructed to grant the decision. The following notations are defined first:

- A: the authorization token for the final decision;
- D_i: the ID of the i^{th} voter;
- β_i: the ballot assigned to the i^{th} voter;
- t: the minimum number of ballots needed to reconstruct the authorization token;
- $\oplus, \bigoplus, \prod$: the addition, cumulative sum, and product operators in finite fields;
- $GF(2^b)$: finite field with 2^b elements.

Protocol II.1. The two-stage group decision making protocol is as follows. The ballot computing and authorization token reconstruction equations are from the t-threshold secret sharing (TSS) scheme. It was first introduced by Shamir [6] in 1979 and spawned many variations [7].

1) **Ballot Distribution:**
   a) A dealer holding the authorization token A authenticates all the n voters/devices (e.g. by PUF), so that each of them is linked to a legitimate ID D_i;
   b) The dealer computes the ballot for each voter with:
   $$\beta_i = a_0 \oplus a_1 D_i \oplus a_2 D_i^2 \oplus \cdots \oplus A D_i^{t-1}, \quad (1)$$
   where A, β_i, $D_i \in GF(2^b)$, and all other coefficients can be arbitrarily chosen. The dealer distributes the ballots to all the n voters. It can be seen that each voter's ballot is closely associated with its ID.

2) **Group Decision Making (voting):**
   a) If there are at least t voters supporting the decision, they will submit their IDs and ballots. Then with the Lagrange interpolation formula:
   $$A = \bigoplus_{i=0}^{t-1} \frac{\beta_i}{\prod_{j=0,j\neq i}^{t-1}(D_i \oplus D_j)}, \quad (2)$$
   the authorization token A can be reconstructed in order to grant the permission to the final decision.
   b) If there are less than t ballots, then A remains unknown and the decision proposal is denied. ■

Remark II.1. In a multiple-choice style group decision making scheme, for each choice, a device can be given a ballot generated from that choice's authentication token.

III. Weighted Group Decision Making

It is notable that due to the mathematical nature of the threshold secret sharing and the conventional PUF architecture, each device can be authenticated for only one identity and thus can have only one ballot (equal influence) associated with it. However, there exists quite a few situations requiring weighted group decision making, such as the scenario mentioned in Section I or cases in medical IoT networks. Medical devices such as ECG and EEG machines, oximeters, or blood glucose meters etc., should affect differently the group decision in a given patient emergency. Therefore, we propose a Multi-identity PUF (Mi-PUF), so that for an individual device holding a Mi-PUF, it can be authenticated and associated with multiple legal identities. This property therefore enables the devices/voters to demonstrate various influence levels in a decision making.

A. Mi-PUFs for Authentication and Identification

For a conventional PUF, no matter how many challenge and response pairs (CRPs) it has, it can only be identified and authenticated for one identity. However, a Mi-PUF can be identified and authenticated for multiple identities through an extra input called identity selection *Iden-Sel*. In addition, although a Mi-PUF remains a single piece of hardware, with different *Iden-Sel* inputs, each identity has a distinct CRP behavior from another. We introduce the new notations:

- CHL_k: the k^{th} challenge to a Mi-PUF;
- D_{i_j}: the j^{th} identity of the Mi-PUF indexed by i;
- $|ID|$: the total number of a Mi-PUF's identities;
- $|id|$: the number of identities a verifier can verify on a Mi-PUF, $|id| \leq |ID|$;
- RSP_{k_j}: the response of CHL_k under identity D_{i_j} of the Mi-PUF;
- β_{i_j}: the ballot assigned to D_{i_j};
- p: the impact level of a device, where $1 \leq p \leq t$.

Protocol III.1. The Mi-PUF's authentication procedure for multiple identities on a single device is as follows:

1) Before a Mi-PUF is delivered to its owner D_i, the manufacturer will challenge it with multiple challenges $\{CHL_0, CHL_1, \cdots, CHL_k, \cdots\}$ for $|ID|$ rounds under identities $\{D_{i_0}, D_{i_1}, \cdots, D_{i_{|ID-1|}}\}$. The responses are stored under each identity's CRP set;
2) A verifier acquires $|id|$ number of CRP sets from the manufacturer in a trusted way, where $|id| \leq |ID|$;
3) When the owner claims to be D_{i_j}, it needs to be verified by taking the verifier's CHL_k to its Mi-PUF;
4) When the Mi-PUF returns a response to the verifier, this response will be compared with the pre-stored RSP_{k_j} under identity D_{i_j} to check its validity. If it matches with the verifier's record, then the Mi-PUF owner is legitimately associated with D_{i_j}. The verifier can repeat the steps above to validate other identities as well.

B. Weighted Group Decision Making Assisted by Mi-PUF

Protocol III.2. The proposed 3-step scheme is as follows:

1) **The Impact Level of Voters:**
   a) Each device/voter is pre-determined to have an impact level p. The smaller is p, the higher the device weight will be on the group decision. A device's p is defined as the minimum number of such devices needed to reconstruct the authorization token A.
   b) Therefore, for a device with impact level p, the number of identities and corresponding ballots it can possess are both $\lceil t/p \rceil$, where t is the decision making threshold. $p = 1$ means a single device having the authority to make the group decision. A device with $p = t$ means it would take at least t such low influence items to make a decision.
2) **Ballot Distribution:**
   a) The verifier first authenticates each device by the Mi-PUF authentication Protocol III.1, so that each can be associated with $|id|$ legitimate identities;
   b) The verifier also works as a dealer. It holds the authorization token A to be split to the devices/voters. The ballot for each voter can be computed with:
   $$\beta_{i_j} = a_0 \oplus a_1 D_{i_j} \oplus a_2 D_{i_j}^2 \oplus \cdots \oplus A D_{i_j}^{t-1}, \quad (3)$$
   where β_{i_j} is the ballot computed based on the device D_i's identity D_{i_j}. Different numbers of ballots are then sent out to voters based on their $|id|$.
3) **Group Decision Making:**
   a) During group decision making, each supportive device will submit its $|id|$ number of IDs and ballots. Devices with a larger $|id|$ will obviously influence the result more than the ones with a smaller $|id|$. If there are at least t ballots, then the authorization tag can be reconstructed by [Eq. 2].
   b) If there are less than t number of ballots, then A remains zero-knowledge to all the devices and the decision proposal is denied. ■

IV. Multi-identity PUF (Mi-PUF)

In the previous section the weighted group decision making has been introduced to support more flexible and

sophisticated voting scenarios. In this section we focus on the design and implementation of the Mi-PUF primitive. We propose two types of Mi-PUFs based on the ring oscillator (RO) PUFs and the Arbiter PUFs [1] respectively. In their basic elements, three functional blocks are integrated: the ID box, strict avalanche criterion (SAC) network, and first order Reed-Muller (FORM) encoder. Fig. 1 and 2 illustrate the basic element of the RO-based and Arbiter-based Mi-PUFs. The ID Box is installed between every two neighboring stages. The SAC and FORM encoder transform the *Iden-Sel* for security purposes.

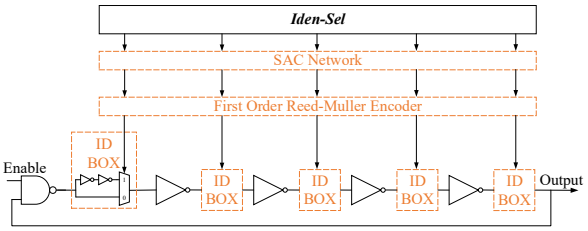

Figure 1: The upgraded RO for the RO-based Mi-PUF.

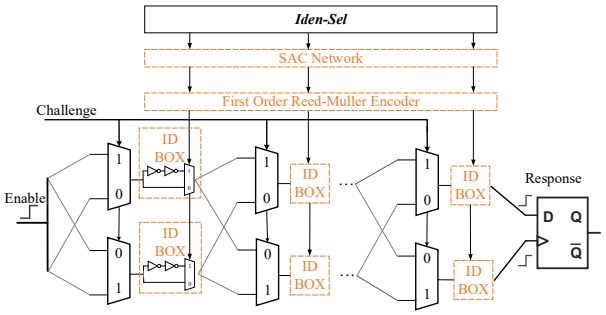

Figure 2: The upgraded MUX chain for the Arbiter-based Mi-PUF.

A. ID Box

The ID box manifests the personality variation of Mi-PUF by altering the PUF circuit. It consists of two inverters and one 2-to-1 MUX as shown in Fig. 1 and 2. Each ID box takes in one bit of the transformed *Iden-Sel* input, and will affect the timing but not the value of the propagated signal. With different inputs to the ID Box, the RO- and Arbiter-based Mi-PUFs will function under various timing characteristics, thus behaving differently under the change of *Iden-Sel*.

If more identities are needed (especially for weak PUFs such as RO PUF), the ID box can further evolve in at least two ways as shown in Fig. 3: 1) by fitting in multiple ID boxes between two stages; 2) by adding more choices of timing routes to the MUX. Both points are able to provide a great number of additional circuit delay characteristics to the Mi-PUF.

B. FPGA Implementation of the ID Box

If the routing and placement of the ID boxes are automatically carried out by the synthesis tools, then the delay bias produced by the automation will dominate over the LUTs' intrinsic delay, resulting in the reduction of the PUF's

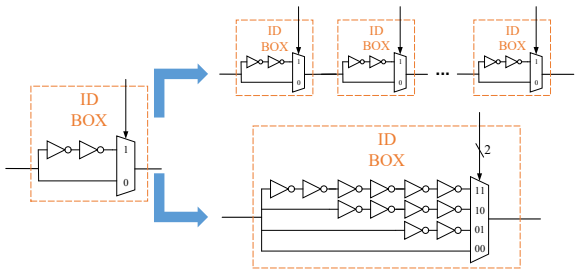

Figure 3: In the basic design of an ID Box, each box possesses two identities, and so the five ID Boxes in Fig. 1 provide 2^5 identities in total. Suppose it is replaced by h 2-to-1 ID boxes or one h-to-1 ID box, then the number of identities will increase to 2^{5h}.

uniqueness. Namely different Mi-PUF will tend to have the same CRPs. Therefore, all the elements of the ID box will need to be configured identically, including:

1) Placement: which SLICE (logic slice) and BEL (basic element) a LUT or DFF will be placed into;
2) Pins: which pins a LUT will be using;
3) Route: the path connecting two or more BELs.

First, the LUT placement can be set in the *.xdc* constraint file through the following two commands.

```
# Placing each cell into a fixed SLICE
set_property LOC SLICE_XxYy [get_cells
    cell_name]

# Placing each cell into a LUT in that SLICE
set_property BEL bel_index [get_cells
    cell_name]
```

It is notable that the "x" and "y" coordinates in the "set_property LOC SLICE_XxYy" command are absolute. It determines which slice on FPGA this element belongs to. However, the "bel_index" in the "set_property BEL bel_index" command is relative. It determines which LUT in a slice is selected to place this element. Moreover, according to the Vivado constraints manual [8], different pins of a LUT are manufactured with different speeds. For example, LUT pins indexed by A6 and A5 are faster than A1 $\sim$ A4. Therefore, the same pin of an element in all ID Boxes has to be locked identically:

```
# Lock the pin I0 to A5 of a cell
set_property LOCK_PINS {I0:A5} [get_cells
    cell_name]
```

Finally, each LUT's route to the next needs to be fixed. For the ID box in the RO PUF-based Mi-PUF, the designer needs to ensure the ID boxes in all ROs are identically routed. This is because each RO is more of a standalone system and it does not require symmetric design. However, in an Arbiter-based Mi-PUF the routing is more complicated, since each stage is connected to the next in a staggered manner, so the two routes need to have as much symmetry as possible. Fig. 4 (part of Fig. 2) is an example requiring symmetric routing.

The identical routing path between the two stages cannot be "created" by simply editing the *.xdc* constraints. It has

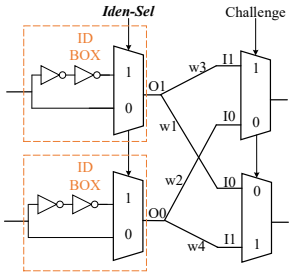

Figure 4: To eliminate the delay difference created from biased routing, the routing of w1 should be the same as w2, and w3 the same as w4. I1 of the upper MUX should be the same pin as the I1 of the lower MUX, similarly to I0.

to be explored using the "Assign Routing Mode" in Vivado, which is an FPGA editing mode allowing user-defined configurations in post implementation. Since from one cell to another there are only limited resources (paths) for routing, designers will need to first discover and fix the best BEL, pin, and route for a cell within these limited resources, and then generalize it using the following command or macro:

```
set_property FIXED_ROUTE [get_nets wirename]
```

Fig. 5 shows that {w1, w3} and {w2, w4} in Fig. 4 are symmetrically routed to the next stage.

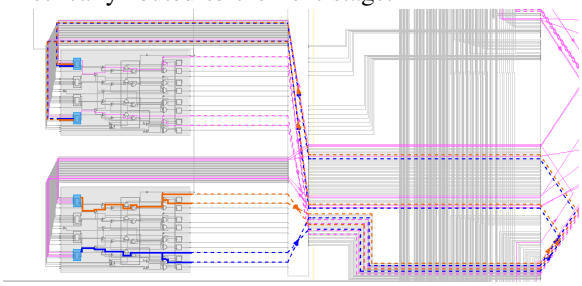

Figure 5: w1 and w3 are in blue, and w2, w4 in orange.

C. Strict Avalanche Criterion Network

On receiving an input of *Iden-Sel*, it will be first transformed by a Strict Avalanche Criterion (SAC) network, which is also used in the design of lightweight secure PUFs [9]. In a SAC network, whenever a single input bit is flipped, each output bit should have a probability of 0.5 to flip. The introduction of the SAC network is to increase the unpredictability and diversity a Mi-PUF circuit generated by a given *Iden-Sel*. Thus every two *Iden-Sel* inputs with a small Hamming distance will not result in similar circuits. This helps to prevent learning attacks across different identities.

D. First Order Reed-Muller Encoder

Encoding schemes using error control codes [10] are usually adopted to address the issue of side channel attacks on PUFs [11]. In the Mi-PUF design the first order Reed-Muller (FORM) encoder is utilized to address this issue. An N-bit FORM codeword can be generated by check matrix $M = [\frac{M_1}{M_0}]$, where M_1 is a single row of all 1's, and the columns of M_0 consist of all different vectors of $\lceil \log_2 N \rceil$ bits. One important attribution of FORM codes is to ensure equal weights and uniformity of 1's and 0's in all its outputs (the vectors of all 1's and all 0's are excluded). In this way there are always half of ID boxes turned on in the slower route (port 1 of the MUX in the ID Box), and half of them in the faster route (port 0 of the MUX), which makes the power analysis on different personalities harder for attackers. The price to pay is the encoding redundancy.

V. Design and Implementation Evaluation

In this section, we evaluate the design and implementation of Mi-PUF in terms of (1) response sizes, (2) uniformity, (3) uniqueness of the identities using intra-board/chip and inter-board/chip Hamming distances, and finally (4) the hardware overhead.

A. $|CRP|$ and $|RSP|$ Set Sizes

For a five-stage RO PUF based Mi-PUF (cf. Fig. 1), it is fair to assume that there are m ROs in an RO group and r of such groups. Since the number of identities will be $|ID| = 2^5$, the total number of CRPs is:

$$|CRP| = 2^5 \cdot \binom{m}{2}, \tag{4}$$

which is also the total number of unique responses $|RSP|$.

For an Arbiter PUF based Mi-PUF in Fig. 1, the assumption is c MUX pairs in a MUX chain and r of such chains. The number of identities is $|ID| = 2^c$ and the total number of CRPs and unique responses are:

$$|CRP| = 2^c \cdot 2^c = 2^{2c}, \tag{5}$$

In [12], Gassend et al. proposed a controlled PUF (CPUF) which introduced the concept of a personality input to increase the range of the conventional PUF. The differences between a CPUF and a Mi-PUF are:

1) A CPUF does not change the conventional PUF's structure. It remains the same piece of hardware while using hash to combine the personality input and challenge into a new CHL. As for the Mi-PUF, each identity is linked to a unique PUF circuit and signal route.
2) A CPUF does not increase the number of unique responses, while Mi-PUF does. Mi-PUF also provides more $|CRP|$ than the CPUF under the same settings.

Table I shows the differences among the proposed Mi-PUF, conventional PUF, and CPUF on the $|CRP|$ size and unique response set size $|RSP|$.

Table I: $|CRP|$ and $|RSP|$ Comparison

PUF Type	Proposed Mi-PUF		conventional PUF		CPUF	
	$\|CRP\|$	$\|RSP\|$	$\|CRP\|$	$\|RSP\|$	$\|CRP\|$	$\|RSP\|$
RO	$2^5 \cdot \binom{m}{2}$	$2^5 \cdot \binom{m}{2}$	$\binom{m}{2}$	$\binom{m}{2}$	$a \cdot \binom{m}{2}$	$\binom{m}{2}$
Arbiter	2^{2c}	2^{2c}	2^c	2^c	$a \cdot 2^c$	2^c

[I] a the constant is the number of personalities.
[II] The proposed Mi-PUF has the largest $|CRP|$ and $|RSP|$, CPUF the second, and the conventional PUF the smallest. Posessing more responses can help the PUF be more resistant to modeling attacks.

B. Uniformity

Under the uniformity testing, we examine the PUF's responses in terms of their bit vector balance between 0's and 1's. The ideal is 50-50 ratio. For a given response, uniformity is calculated by:

$$\text{Uniformity} = \frac{1}{r}\sum_{r}^{j=1} RSP(j) \cdot 100\%$$

where $RSP(j)$ is the j^{th} bit of the response. We examine 32 identities' uniformity in each of the five Mi-PUF sizes with $r \in \{8, 16, 32, 96, 128\}$.

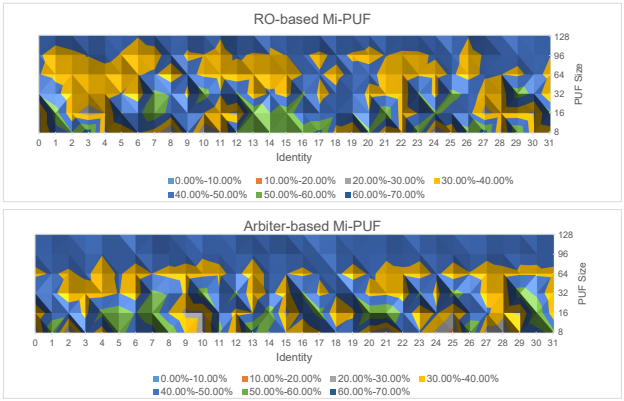

Figure 6: The uniformity of both RO- and Arbiter-based Mi-PUFs are around 35% to 50% across the 32 identities. The larger is the PUF size, the better the uniformity.

C. Uniqueness

Uniqueness is one of the most important parameters of a PUF and is evaluated using the average *Hamming Distances* (HD) of the PUF's responses. We provide two types of uniqueness for the Mi-PUF's evaluation: 1) the "PUF uniqueness" across 12 FPGA boards (inter-board uniqueness under the same *Iden-Sel*), and 2) the "identity uniqueness" across $|id| = 32$ identities on one FPGA (intra-board uniqueness under different *Iden-Sel*). Both are calculated based on the average Hamming Distance (HD) of their responses under the same CHL:

$$\text{Uniqueness} = \frac{2}{l(l-1)}\sum_{i=1}^{l-1}\sum_{j=i+1}^{l}\frac{HD(RSP_i, RSP_j)}{r}\cdot 100\%$$

where l is the number of boards for the inter-board uniqueness, and number of identities for the intra-board.

Table II: Uniqueness Evaluation

	Inter-board		Intra-board	
	RO-based	Arbiter-based	RO-based	Arbiter-based
Without Hash	33.81%	19.13%	25.40 %	23.09 %
With Hash	49.27%	49.31%	49.30%	49.29%

[I] It can be seen that even without hashing the Mi-PUF response, the identically placed and routed Mi-PUFs already achieve an acceptable uniqueness (50% being the ideal).

D. Hardware Overhead

The Mi-PUF uses more resources than the conventional PUF because of its ID Boxes. Table III shows the hardware overhead (LUT utilization and on-chip power) of the elements of the Mi-PUF over the conventional (CNV) PUF.

Table III: Overhead Evaluation

	RO-based			Arbiter-based		
	CNV	Mi-PUF	Overhead	CNV	Mi-PUF	Overhead
LUT	118	88	34.1%	136	546	300.1%
Power (W)	0.654	0.952	45.7%	1.789	4.086	128.4%

[I] The basic element of an RO-based Mi-PUF is two ROs (with ID Boxes) and one counter, while for Arbiter-based Mi-PUF, it is a MUX chain (with ID Boxes). We only look into LUT utilization, since ID Boxes do not consume any additional flip-flops.

[II] The RO-based Mi-PUF has less overhead because each ID Box is a small component compared to the counter. For the Arbiter PUF where a single stage consumes two LUTs, the ID Box becomes a relatively large add-on.

VI. Conclusion

We propose a Multi-identity PUF (Mi-PUF) assisted group decision making scheme. With the Mi-PUF, different devices/parties in a decision making network can (1) be authenticated with multiple identities, and (2) make group hierarchy based and context-aware decisions. The design and implementation details of Mi-PUF on FPGA are presented.

Acknowledgment: This research is partially supported by the NSF grant (No. CNS-1745808).

References

[1] I. Papakonstantinou and N. Sklavos, "Physical unclonable functions (pufs) design technologies: Advantages and trade offs," in *Computer and Network Security Essentials*. Springer, 2018, pp. 427–442.

[2] B. Schoenmakers, "A simple publicly verifiable secret sharing scheme and its application to electronic voting," in *Annual International Cryptology Conference*, 1999.

[3] S. Iftene, "General secret sharing based on the chinese remainder theorem with applications in e-voting," *Electronic Notes in Theoretical Computer Science*, 2007.

[4] A. Neff, "A verifiable secret shuffle and its application to e-voting," in *Proceedings of the 8th ACM conference on Computer and Communications Security*. ACM, 2001.

[5] C.-C. Chang, C.-C. Lin, T. H. N. Le, and H. B. Le, "A probabilistic visual secret sharing scheme for grayscale images with voting strategy," in *Electronic Commerce and Security, International Symposium on*, 2008.

[6] A. Shamir, "How to share a secret," *Communications of the ACM*, 1979.

[7] L. Bu, H. D. Nguyen, and M. A. Kinsy, "Rasss: A perfidy-aware protocol for designing trustworthy distributed systems," *2017 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, 2017.

[8] Xilinx, "Vivado design suite user guide - using constraints," 2015.

[9] M. Majzoobi, F. Koushanfar, and M. Potkonjak, "Lightweight secure pufs," *Computer-Aided Design*, 2008.

[10] L. Bu, M. Mark, and M. A. Kinsy, "A short survey at the intersection of reliability and security in processor architecture designs," *IEEE Computer Society Annual Symposium on VLSI*, 2018.

[11] A. Mahmoud *et al.*, "Combined modeling and side channel attacks on strong pufs," *IACR Cryptology Archive*, 2013.

[12] B. Gassend, M. V. Dijk, D. Clarke, E. Torlak, S. Devadas, and P. Tuyls, "Controlled physical random functions and applications," *ACM Transactions on Information and System Security (TISSEC)*, 2008.