Sorting Attacks Resilient Authentication Protocol for CMOS Image Sensor Based PUF

Chandan Kumar, Mahendra Rathor, Urbi Chatterjee

***Abstract*—Physically Unclonable Functions (PUFs) have emerged as a viable and cost-effective method for device authentication and key generation. Recently, CMOS image sensors have been exploited as PUF for hardware fingerprinting in mobile devices. As CMOS image sensors are readily available in modern devices such as smartphones, laptops etc., it eliminates the need for additional hardware for implementing a PUF structure. In ISIC2014, an authentication protocol has been proposed to generate PUF signatures using a CMOS image sensor by leveraging the fixed pattern noise (FPN) of certain pixel values. This makes the PUF candidate an interesting target for adversarial attacks. In this work, we testify that a simple sorting attack and a win-rate (WR) based sorting attack can be launched in this architecture to predict the PUF response for given a challenge. We also propose a modified authentication protocol as a countermeasure to make it resilient against simple sorting and WR sorting attacks.**

***Index Terms*—CMOS Image Sensor, PUF, Hardware Security, Sorting Attack.**

I. Introduction

In the current decade, the number of subscriptions to cellphone devices has escalated worldwide because of their wide usability in modern-day life [1]. However, the growth of the cellphone industry has also opened up opportunities for black marketing for adversaries. An adversary may introduce counterfeit, refurbished mobile phones in the supply chain [2] to earn illegal revenue or to sabotage brand value. Because of these counterfeited and refurbished mobile phones, the global market share of genuine mobile phones has been massively affected. Hence, it is vital to ensure the authenticity of cellphone devices and discern their fake counterparts to protect both the revenue and brand value of the original vendors. Physically unclonable functions (PUFs) have emerged as a promising solution to provide a unique signature of each manufactured chip or device. In the CMOS image sensor-based PUF, an inherent imperfection in the image sensor manufacturing process is leveraged to generate unique signatures.

In ISIC2014, Cao et al. [3] employed a CMOS image sensor-based PUF for smartphone identification. In this approach, fixed pattern noise (FPN) present in an image sensor was exploited to generate a reliable and unique signature for the identification of smartphones. The term FPN [4] is defined as the variations in output values of pixels under uniform illumination. These pixel output variations across the sensor incur due to mismatch in the device and interconnect parameters. In 2021, Yamada et. al. [5] first highlighted two adversarial attacks viz. simple sorting and column FPN attacks on CMOS image sensor PUF. In simple sorting attack list of pixel output order is generated by sorting raw pixel output using collected CRPS, whereas a column FPN attack uses the column FPN property of the raw pixels to facilitate the attack. In the column FPN attack, a win-rate (WR) function is used to compute the win-rate of the pixels at known challenges (addresses) [5]. A win-rate indicates how many times a pixel output value is greater than the other pixel values among the known CRPs. Further, a column average of the win rate of the pixels in each column is computed. Next, these column averages become the basis of sorting. However, in this paper, we have shown that a much simpler attack that simply uses win-rate function can be applied to perform the sorting among the known addresses. This simple win-rate based sorting attack is capable to provide similar accuracy as the column FPN attack. The sorting attacks make the device authentication using CMOS image sensor-based PUF weak. Hence, it demands attention to develop a strong device authentication protocol using CMOS PUF that is resilient against both sorting attacks.

Our major contributions to this paper are as follows:

- We testified that the existing CIS PUF-based authentication protocol is susceptible to a simple sorting attack and win-rate based sorting attack. We implemented both simple and win-rate based sorting attacks on existing CIS PUF (ISIC2014) and estimated the prediction accuracy to be 86% and 87.5% respectively.
- We propose a new authentication protocol using CIS PUF that incorporates the comparison of the pixel value at a particular challenge address with all the neighbour pixels. It is followed by a modulo-XOR operation based expansion of bit-vectors, to generate robust CRPs. Thus the proposed authentication protocol eliminates the transitive relations among the address pairs. And it can successfully offer a robust countermeasure against the simple sorting and win-rate based attacks.
- We analyse our proposed scheme against an adversarial model that may target the XOR operation and parity bits to determine the comparative relation among the pixels for sorting.We found, in this attack model, the brute-force effort of predicting a response bit is in $\mathcal{O}(2^g)$ where g is the size of expansion of the bit-vector.

II. Background on CMOS Image Sensor PUF

We first present the circuit design and operations of 3T-active pixel sensor (APS), then we illustrate the CMOS image

C. Kumar and U. Chatterjee are with department of computer science engineering at Indian Institute of Technology Kanpur, 208016, India (e-mail: ck80152@gmail.com, urbic@cse.iitk.ac.in). M. Rathor is with software innovation centre at Indian Institute of Technology (BHU) Varanasi, 221005, India (e-mail: mahendra.chr@itbhu.ac.in).

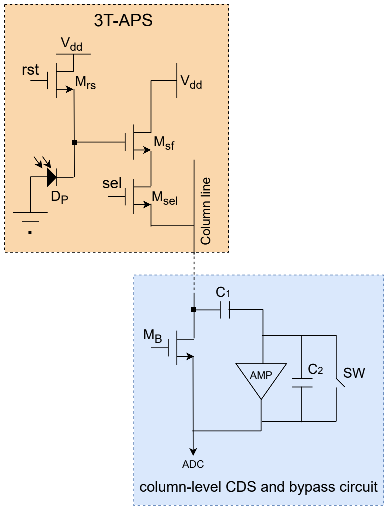


Fig. 1. A 3T-APS pixel and column readout & CDS design of CMOS image sensor [3]

sensor design and how it can be leveraged to act as a PUF.

A. Circuit Design and Operations of a 3T CMOS Image Sensor

A typical 3T-APS [6] and column readout & bypass circuit is depicted in Fig. 1. A pixel cell is composed of the following: (i) a photo-diode D_P (ii) a select transistor M_{sel} (iii) a reset transistor M_{rs} and (iv) a source follower readout transistor M_{sf}. Upon switching the M_{rs} on, the reset voltage V_{DP} of the photo-diode D_P is given as follows [3]:

$$V_{DP} = V_{dd} - V_{th,rs} + V_{kt} \tag{1}$$

where, $V_{th,rs}$ and V_{kt} indicate the reset transistor's threshold voltage and the thermal noise, respectively, and V_{dd} is supply voltage. The voltage V_{kt} contributes to the main random noise due to the reset operation.

Further, once M_{rs} is switched off, V_{DP} discharges due to flow of the photo-current I_p under the incident light (forming a dark current). Next, the select transistor M_{sel} is switched on after an exposure period t. The following equation is used to calculate the pixel's output voltage V_O [3]:

$$V_O = V_{dd} - V_{th,rs} + V_{kt} - V_{th,sf} - \frac{I_p \times t}{C_{DP}} \tag{2}$$

where, C_{DP} and $V_{th,sf}$ represent the photo-diode junction capacitance and threshold voltage of M_{sf} respectively. It is noteworthy that the following factors contribute to the variations in the pixel output values: (i) differences in photo-diode size (ii) variations in threshold voltages of M_{RS} and M_{SF} (iii) capacitance. The variation in pixel output values is also referred to as FPN.

Further, a column-level correlated double sampling (CDS) and bypass circuit for pixel value readout is shown in Fig. 1. This column readout circuit reads the pixel output value and feeds it to an analog-to-digital converter (ADC) to obtain the corresponding digital value. The readout circuit can be enabled to act in two different modes: (i) regular sensing mode and (ii) PUF mode. A particular mode is enabled using a CDS circuit and a bypass transistor M_B. In the regular sensing mode, the bypass transistor is turned off and the CDS circuit is enabled to read the pixel voltage. Since the FPN adversely impacts the image quality, the CDS is employed as a noise-cancelling circuit to suppress FPN. Next, we discuss how to use the column readout circuit in the PUF mode and generate challenge-response pairs (CRPs) for device authentication.

B. CMOS Image Sensor as A PUF

To use a CMOS image sensor as a PUF, the desirable impact of FPN for the formation of random and unique PUF response needs to be retained. Since the CDS can decrease the randomness by diminishing the required effect of FPN, it is bypassed through an additionally inserted parallel bypass transistor M_B as shown in Fig. 1. Then, the output voltage of pixel during reset (V_{rst}) is directly measured. The V_{rst} can be calculated by subtracting the threshold voltage $V_{th,sf}$ of M_{sf} from (1) [3]. The V_{rst} value of every pixel is scanned out to read a complete image and stored in the memory in digital form. Thus, we obtain a "reset image (RI)". The uniqueness in the pattern of each pixel array is obtained due to the variations in V_{rst}. The variation in V_{rst} is contributed by the variations in $V_{th,rs}$ and $V_{th,sf}$.

The challenge-response pair (CRP) generation algorithm [3] for the CMOS image sensor PUF-based authentication is explained as follows. The main crux is to compare the reset voltages of two pixels to obtain each response bit. For a challenge(address) Ch, the pixel value X_{Ch} in the image "RI" is obtained to provide a stable response bit. An n-bit challenge Ch is used to initialize a linear feedback shift register (LFSR) counter. The n-bit LFSR counter generates another challenge (address) Ch' by shifting Ch in N $(N < 2_{n-1})$ clock cycles. Further, the X'_{Ch} pixel value is retrieved and compared with X_{Ch}. The output bit is either 0 or 1 depending on whether the pixel value is larger. The produced bit is regarded as stable if the absolute value of X_{Ch} - $X_{Ch'}$ is greater than a preset threshold X_Z. If this absolute value is less than or equal to X_Z, a new stable pair of pixels will be formed by changing the LFSR's content by one clock cycle. This process is performed until the whole pixel array is traversed. However, the above-mentioned device authentication protocol using COMS image sensor PUF is vulnerable to simple sorting and win-rate based sorting attacks. In the next section, we discuss them in detail.

III. Sorting Attack on CMOS Image Sensor PUF

In this section, we describe two adversarial attacks that are capable of predicting the PUF response given a challenge with considerable accuracy. In this attack model, we assume that the attacker knows the CRP generation algorithm and is capable of eavesdrop some CRPs of a CMOS image sensor-based PUF to conduct the simple sorting attack and win-rate based sorting attack. In the CMOS image sensor PUF, two pixels are compared to obtain a PUF response bit and the same pixels can contribute to producing multiple responses. Due to this reason, the attacker can exploit the same pixel

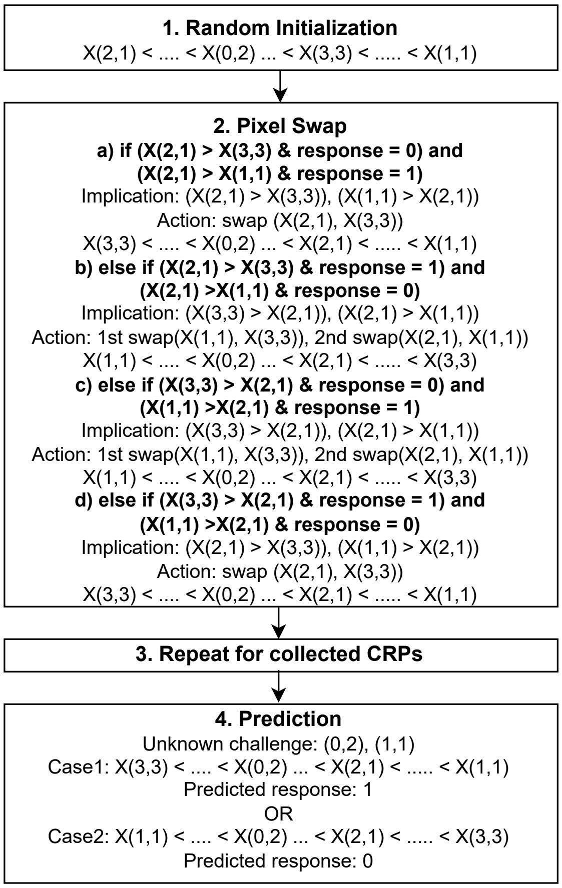

1. Random Initialization
X(2,1) < .... < X(0,2) ... < X(3,3) < ..... < X(1,1)
2. Pixel Swap
a) if (X(2,1) > X(3,3) & response = 0) and
(X(2,1) > X(1,1) & response = 1)
Implication: (X(2,1) > X(3,3)), (X(1,1) > X(2,1))
Action: swap (X(2,1), X(3,3))
X(3,3) < .... < X(0,2) ... < X(2,1) < ..... < X(1,1)
b) else if (X(2,1) > X(3,3) & response = 1) and
(X(2,1) >X(1,1) & response = 0)
Implication: (X(3,3) > X(2,1)), (X(2,1) > X(1,1))
Action: 1st swap(X(1,1), X(3,3)), 2nd swap(X(2,1), X(1,1))
X(1,1) < .... < X(0,2) ... < X(2,1) < ..... < X(3,3)
c) else if (X(3,3) > X(2,1) & response = 0) and
(X(1,1) >X(2,1) & response = 1)
Implication: (X(3,3) > X(2,1)), (X(2,1) > X(1,1))
Action: 1st swap(X(1,1), X(3,3)), 2nd swap(X(2,1), X(1,1))
X(1,1) < .... < X(0,2) ... < X(2,1) < ..... < X(3,3)
d) else if (X(3,3) > X(2,1) & response = 1) and
(X(1,1) >X(2,1) & response = 0)
Implication: (X(2,1) > X(3,3)), (X(1,1) > X(2,1))
Action: swap (X(2,1), X(3,3))
X(3,3) < .... < X(0,2) ... < X(2,1) < ..... < X(1,1)
3. Repeat for collected CRPs
4. Prediction
Unknown challenge: (0,2), (1,1)
Case1: X(3,3) < .... < X(0,2) ... < X(2,1) < ..... < X(1,1)
Predicted response: 1
OR
Case2: X(1,1) < .... < X(0,2) ... < X(2,1) < ..... < X(3,3)
Predicted response: 0

Fig. 2. Procedure for simple sorting attack

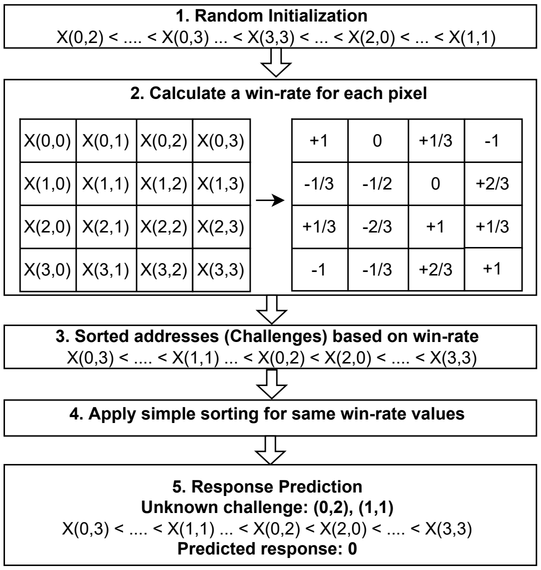
to obtain or relate multiple responses and the PUF becomes susceptible to sorting attacks.

A. Simple Sorting Attack

A simple sorting attack exploits the transitive relation among the known CRPs. In this attack, pixel addresses (challenges) are sorted in order of the pixel output values as illustrated in Fig. 2. First, the attacker randomizes the order of the list of pixel outputs $X_{i,j}$. Now, the attacker searches CRPs for transitive pairings in collected CRPs. For example, for the challenge $X(2,1)$ and $X(3,3)$, if the response is 0, it implies that $X(2,1) > X(3,3)$ is true. Further, for the challenge $X(2,1)$ and $X(1,1)$, if the response is 1, it implies that $X(2,1) > X(1,1)$ is false. As $X(2,1)$ is common in both CRPs, this leads to the conclusion that $X(3,3) < X(2,1) < X(1,1)$. Hence these pixel addresses are swapped in the list to be in the order and the pair $X(3,3)$, $X(1,1)$ with response 1 is added to the list of known CRPs. This process is repeated for all collected CRPs until the pixel output list is sorted. It enables the attacker in guessing an unknown response to a fresh challenge based on the ordered list of raw pixel variants.

B. Win-Rate based Sorting Attack

In this attack, the pixel addresses are sorted based on the value of the win rate among the known CRPs. A win rate of a pixel is defined as $\frac{(W-L)}{T}$, where W and L indicate the number of times the pixel value is greater (win situation) and lesser respectively than other pixels values in the collected CRPs. Whereas, T indicates the number of times the respective pixel address is used in the collected CRPs. Thereby, in the collected CRPs, pixel challenges (addresses) are swapped to sort them based on their win rates. Fig. 3 depicts the process for the scenario when the number of pixels is 4x4. Let us assume that the collected CRPs use $X(2,0)$ three times, with two responses of 0 and one of 1. This implies that the $X(2,0)$ is two times greater (i.e. W=2) and one time lesser (i.e. L=1). Hence, the win rate is calculated to be $\frac{(2-1)}{3} = \frac{+1}{3}$ and labeled on the $X(2,0)$. Once the win rate for all the challenges (pixel addresses) in the collected CRPs is computed, the addresses are sorted in the increasing order of their win rates. Here, a pixel having a higher win-rate indicates that the pixel value is greater than most of the other pixel values in the collected CRPs. Further, as indicated in Fig. 3, a simple sorting may be applied to sort the pixel addresses with same win rate. Thus, the obtained sorted array of pixel addresses helps to predict the PUF responses for unknown challenges and breaks the PUF. This vulnerability of the CMOS image sensor PUF motivated us to propose a new CRP generation algorithm to mitigate both the sorting attacks and make device authentication stronger.


1. Random Initialization
X(0,2) < .... < X(0,3) ... < X(3,3) < ... < X(2,0) < ... < X(1,1)
2. Calculate a win-rate for each pixel
3. Sorted addresses (Challenges) based on win-rate
X(0,3) < .... < X(1,1) ... < X(0,2) < X(2,0) < .... < X(3,3)
4. Apply simple sorting for same win-rate values
5. Response Prediction
Unknown challenge: (0,2), (1,1)
X(0,3) < .... < X(1,1) ... < X(0,2) < X(2,0) < .... < X(3,3)
Predicted response: 0

X(0,0)	X(0,1)	X(0,2)	X(0,3)
X(1,0)	X(1,1)	X(1,2)	X(1,3)
X(2,0)	X(2,1)	X(2,2)	X(2,3)
X(3,0)	X(3,1)	X(3,2)	X(3,3)

+1	0	+1/3	-1
-1/3	-1/2	0	+2/3
+1/3	-2/3	+1	+1/3
-1	-1/3	+2/3	+1

Fig. 3. Win-rate based sorting attack

IV. Proposed Countermeasure and Analysis

In this section, we first discuss the proposed CRP generation algorithm in detail, and then we establish how the proposed countermeasure is capable of mitigating both the sorting attacks. Further, we also present a potential attack model that could target our proposed PUF scheme and then discuss a resilience against it. Fig. 4 highlights the proposed CRP generation scheme. It consists of two phases:

1. Pre-processing phase: The following steps are performed on the pixel array as pre-processing phase before

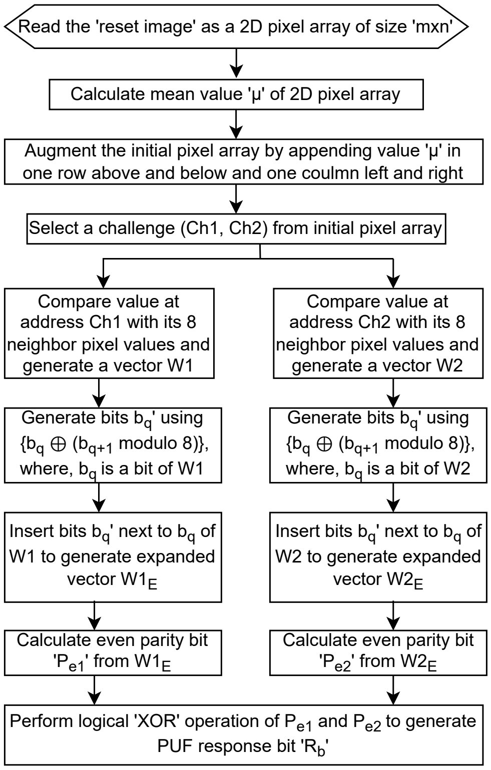

Read the 'reset image' as a 2D pixel array of size 'mxn'
Calculate mean value 'µ' of 2D pixel array
Augment the initial pixel array by appending value 'µ' in one row above and below and one coulmn left and right
Select a challenge (Ch1, Ch2) from initial pixel array
Compare value at address Ch1 with its 8 neighbor pixel values and generate a vector W1
Compare value at address Ch2 with its 8 neighbor pixel values and generate a vector W2
Generate bits b_q' using {b_q ⊕ (b_q+1 modulo 8)}, where, b_q is a bit of W1
Generate bits b_q' using {b_q ⊕ (b_q+1 modulo 8)}, where, b_q is a bit of W2
Insert bits b_q' next to b_q of W1 to generate expanded vector W1_E
Insert bits b_q' next to b_q of W2 to generate expanded vector W2_E
Calculate even parity bit 'P_e1' from W1_E
Calculate even parity bit 'P_e2' from W2_E
Perform logical 'XOR' operation of P_e1 and P_e2 to generate PUF response bit 'R_b'

Fig. 4. Proposed CRP generation procedure for CMOS image sensor PUF

forming the CRP database.

- For every pixel of the CMOS image sensor, the pixel output at reset i.e. V_{rst} is readout. Thus, the entire $2D$ pixel array is scanned and, after being digitized, it is stored in the memory as a "reset image". The $2D$ pixel array of size $m \times n$ is shown below:

$$\begin{bmatrix} X(0,0) & X(0,1) & ... & X(0,n-1) \\ X(1,0) & X(1,1) & ... & X(1,n-1) \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ X(m-1,0) & X(m-1,1) & ... & X(m-1,n-1) \end{bmatrix}$$

- Next, we augment the pixel matrix $m \times n$ by appending one row above and below and one column left and right. The size of new augmented pixel matrix is $(m+2) \times (n+2)$.
- A mean 'μ' of pixel output values is calculated and inserted in the appended rows and columns.

2. CRP generation phase: Once we construct the $(m+2) \times (n+2)$ pixel array, the following steps are taken in order to generate the CRPs.

- From the initial pixel array of size $m \times n$ (without augmentation), two pixel addresses viz. $Ch1$ and $Ch2$ are chosen as a challenge. The augmentation is only performed in order to consider the boarder values of the of initial pixel array in our CRP generation algorithm.
- Around both of the addresses $Ch1$ and $Ch2$, we select separate 3×3 matrices. The below matrices highlight the chosen pixel addresses $Ch1$ and $Ch2$ at the centre and their neighbour pixel addresses.

$$\begin{bmatrix} X(i-1,j-1) & X(i-1,j) & X(i-1,j+1) \\ X(i,j-1) & \textbf{Ch1(i,j)} & X(i,j+1) \\ X(i+1,j-1) & X(i+1,j) & X(i+1,j+1) \end{bmatrix}$$

$$\begin{bmatrix} X(l-1,k-1) & X(l-1,k) & X(l-1,k+1) \\ X(l,k-1) & \textbf{Ch2(l,k)} & X(l,k+1) \\ X(l+1,k-1) & X(l+1,k) & X(l+1,k+1) \end{bmatrix}$$

Where, (i,j) and (l,k) are the indices of the addresses $Ch1$ and $Ch2$ respectively.

- The value X_{Ch1} at the addresses $Ch1$ is compared with its each neighbour pixel value shown in above matrix. If the value X_{Ch1} is greater than its neighbour values, then we collect the result of comparison in the variable b_q to be 1, otherwise 0; where q varies from 1 to 8 for the eight neighbour pixels. A similar process is performed for another address i.e. $Ch2$. This leads to two 8-bit vectors viz. $W1$ and $W2$ corresponding to the address pair $Ch1$ and $Ch2$ of the challenge.
- Now, both vectors $W1$ and $W2$ are expanded to 16 bits using the following function:

$$W_E = \&_{q=1}^{8}[b_q \& (b_q \ \oplus (b_{q+1} \ mod \ 8))] \qquad (1)$$

Where, W_E indicates the expanded 16-bit vector, '&' indicates the concatenation operator and '$\oplus$' indicates the XOR operation. Fig. 5 depicts the above discussed process of 8 to 16 bits expansion expressed in (3). Applying XOR operation in (3) ensures that the even/odd parity remain unaffected (same for pre and post expansion), simultaneously fulfilling the purpose of expansion to 16 bits to enhance security. Let us assume that $W1_E$ and $W2_E$ are the expanded 16-bit vectors of $W1$ and $W2$.

- Next, we compute the even parity bits P_{e1} and P_{e2} from $W1_E$ and $W2_E$ respectively (if the number of ones are even then the parity bit is 1, otherwise 0).
- The response bit R_b corresponding to the given challenge is computed using the following equation:

$$R_b = P_{e1} \oplus P_{e2} \qquad (2)$$

Applying XOR operation ensures that the probability of getting the response bit to be '1' or '0' is around $\frac{1}{2}$.

Further, the above steps are repeated to obtain multiple CRPs using the proposed approach for a CIS based PUF.

Intuition behind the resilience: The sorting attack on the existing CIS PUFs exploits the transitive relations among the pixel addresses as the response bit generation is based on the direct comparison of two addresses. Hence it directly leads to sorting of the addresses. However, in our case, the attacker needs to know total 16 comparative relations to be able to predict a response bit for a given challenge. Secondly, the

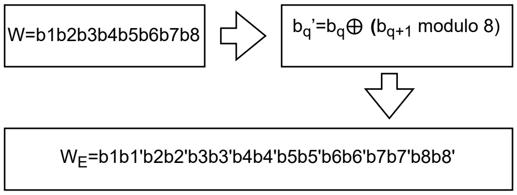

W=b1b2b3b4b5b6b7b8
b_q'=b_q⊕ (b_{q+1} modulo 8)
W_E=b1b1'b2b2'b3b3'b4b4'b5b5'b6b6'b7b7'b8b8'

Fig. 5. Proposed procedure for expansion of vector W form 8 to 16 bits, where q varies from 1 to 8

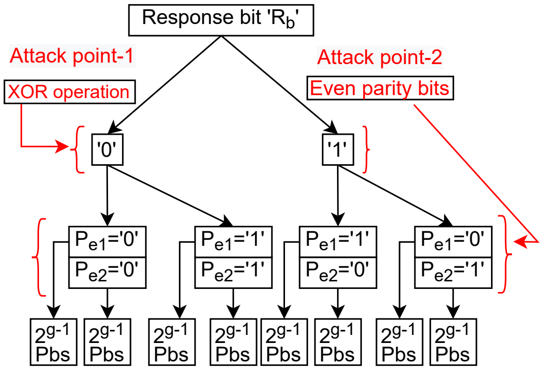


Fig. 6. A potential attack model on proposed PUF, where Pbs indicates the total possibilities

two addresses in the challenge are not directly compared. Hence, by simply knowing one CRP, the attacker cannot guess the comparative relation between the two addresses in the respective challenge. A further improvement in the attack can still be developed to build transitive relationship as shown in Fig. 6. The attacker may try to perform a brute force analysis to find out the even parity bits and in-turn the 16 comparative relations of the address pair (in a challenge) with their respective neighbour pixels. In order to do so, the following attack points may be identified by the attacker:

- **Attack point-**1**:** The attacker may target the XOR operation to infer the various possible even parity bits P_{e1} and P_{e2} viz. $(0,0)$, $(0,1)$, $(1,0)$ and $(1,1)$.
- **Attack point-2:** The attacker may explore all possible combinations of $W1_E$ and $W2_E$ corresponding to even parity. For the size of, for example, g bits, the attacker needs to explore 2^{g-1} possibilities of even/odd parity.

However, the proposed PUF is resilient against the above mentioned attack scenario. We analyse the resilience in terms of brute force effort required in determining the comparative relations to perform the sorting, from an attacker's perspective. We quantify the brute force effort with respect to one CRP in terms of a parameter B_f given below:

$$B_f = 2^{4(g-1)} \tag{3}$$

Where, g denotes the size of the expanded vectors $W1_E$ and $W2_E$ and 2^{g-1} is the total number of possibilities for the parity bit P_{e1} or P_{e2} being even or odd. As shown in Fig. 6, the P_{e1} and P_{e2} can have four combinations to determine the response bit R_b.

However, to predict the response bit for an unknown challenge, the attacker needs to find 16 new comparative relations. In order to do so, the attacker needs to collect multiple CRPs. Let us assume that the collection of y number of total CRPs can provide the 16 comparative relations. Hence, the brute force effort is enhanced y times which is given as follows.

$$B_f = y \times 2^{4(g-1)} \tag{4}$$

It is noteworthy that the proposed approach is flexible towards exponential expansion in the size of the bit-vectors $W1$ and $W2$ at the cost of some extra XOR operations.

References

[1] T. world in 2014: ICT facts and figures, "International telecommunication union, tech. rep., april 2014. [online]," https://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2014-e.pdf.

[2] D. Barboza, "In china, knockoff cellphones are a hit, april 2009. [online]. available," https://www.nytimes.com/2009/04/28/technology/28cell.html.

[3] Y. Cao, S. S. Zalivaka, L. Zhang, C.-H. Chang, and S. Chen, "Cmos image sensor based physical unclonable function for smart phone security applications," in *2014 International Symposium on Integrated Circuits (ISIC)*, 2014, pp. 392–395.

[4] S.-C. Zhang, B.-Y. Dong, and J.-T. Xu, "Analysis of fixed pattern noise in cmos photodiode active pixel sensor," vol. 18, pp. 798–801, 12 2005.

[5] H. Yamada, S. Okura, M. Shirahata, and T. Fujino, "Modeling attacks against device authentication using cmos image sensor puf," *IEICE Electronics Express*, vol. 18, no. 7, pp. 20 210 058–20 210 058, 2021.

[6] J. Ohta, "Smart cmos image sensors and applications.london, new york: Crc press," 11 2007.