XOR Compositions of Physically Related Functions

Harishma Boyapally*†, Sikhar Patranabis‡§‖, Debdeep Mukhopadhyay*¶
*Indian Institute of Technology Kharagpur, India ‡IBM Research India
E-mail: †harishmasko@gmail.com, §sikhar.patranabis@ibm.com, ¶debdeep@cse.iitkgp.ernet.in
‖Work done wihile the author was affiliated with ETH Zürich, Switzerland and Visa Research, USA.

***Abstract*—Physically related functions (PReFs) are hardware primitives proposed to establish key-exchange between resource-constrained devices with no pre-established secrets. In this paper, we introduce XOR composition of PReFs to eliminate the requirement of revealing the complete functionality of the hardware primitive during the setup phase, which is a prerequisite to setup PReFs. We evaluate the quality of XOR_PReF design by implementing them on Artix-7 FPGAs.**

I. Introduction

Over the past several decades, a plethora of cryptographic solutions, both symmetric-key and asymmetric-key, have been proposed in the literature for securing data and preserving its privacy. Essentially all traditional cryptographic solutions crucially rely on some form of secure storage of "secret" keys on the user's device using trusted platforms. This leads to a question - *how practical is it to assume the existence of such secure storage?* In particular, given the widespread advent of invasive and non-invasive implementation-level attacks [1, 2] on cryptographic implementations, such an assumption might not always be realistic, especially on low-end consumer devices such as encountered in the Internet of Things.

A pair of physically related functions (PReFs) are two devices, for which a unique input set called the "related" input set exists such that they output binary strings that are close to each other with respect to some distance metric. For any input that is not in the related input set, the output of the devices are not close. Additionally, over this unique related input set, any third PReF devices' outputs are not close with either of the former PReF devices. It is mathematically shown in [3], that PReFs can be sampled from devices that implement Boolean functions. In fact, they used PUFs to realize PReFs, where two PUF devices output responses that differ in few bit positions for a subset of challenges. These challenges can be made public, as long as they do not reveal the functionalities of the PUFs. They provided a simple key-exchange scheme between two PReF-enable resource-constrained nodes.

In this work, we explore new properties of PReFs to design superior and practically deployable schemes. We propose XOR composition of PReFs, to eliminate the inherent requirement of revealing the complete truth table of PReFs to generate the related input subset. We experimentally validate the quality of our improved construction, by implementing it on Artix-7 FPGAs.

Notations. Below are some notations that are used in the following sections. For $a, b \in \mathbb{N}$ such that $a \geq b$, we denote by $[a, b]$ the set of integers lying between a and b (both inclusive). We refer to $\lambda \in \mathbb{N}$ as the security parameter, and denote by $\mathsf{poly}(\lambda)$ and $\mathsf{negl}(\lambda)$ any generic (unspecified) polynomial function and negligible function in λ, respectively. Note that a function $f : \mathbb{N} \rightarrow \mathbb{N}$ is said to be negligible in λ if for every positive polynomial p, $f(\lambda) < 1/p(\lambda)$ when λ is sufficiently large.

II. Physically Related Functions

Below, we formalize a unique case of PReFs.

Definition II.1. Physically Related Function (PReF). Let $F : \{f_i : X \rightarrow Y\}$ be a family of functions where X and Y are arbitrary sets. D_i and D_j be two devices that are functionally modeled as f_i and f_j respectively. $(\mathsf{D}_i, \mathsf{D}_j)$ is said to constitute a PReF pair if there exists a unique input set $X_{i,j} \subset X$, such that

1) $f_i(x) = f_j(x)$ for all $x \in X_{i,j}$, and
2) D_i, D_j are conditionally pseudorandom as per Definition II.2.

Definition II.2. Conditional Pseudorandomness. Let $F : \{f_i : X \rightarrow Y\}$ be a family of functions where X and Y are arbitrary sets. For $k \in \mathbb{N}$ and $i \in [1, k]$, let there be an input set $X_{0,i}$ such that $f_0(x) = f_i(x)$ for all $x \in X_{0,i}$. The devices $\mathsf{D}_0 \ldots \mathsf{D}_k$ are functionally modelled as $f_0, \ldots f_k$ respectively such that $(\mathsf{D}_0, \mathsf{D}_i)$ forms a PReF pair, for all $k \in [1, k]$. Let $\mathcal{A}$ be a probabilistic poly-time adversary who has oracle access to the devices $\{\mathsf{D}_1, \ldots, \mathsf{D}_k\}$. Then we say that D_0 is conditionally pseudorandom if $\mathcal{A}$ cannot distinguish between $\mathsf{D}_0(x')$ and a value y that is chosen randomly from Y with more than negligible advantage subject to the restriction that

- if $x' \in X \setminus \left(\cup_{i=1}^{k} X_{0,i} \right)$, then $\mathcal{A}$ is not allowed to issue an oracle query of the form $\mathsf{D}_0(x')$ and
- if $x' \in X_{0,i}$, then $\mathcal{A}$ is not allowed to issue oracle queries of the form $\mathsf{D}_0(x')$ or $\mathsf{D}_i(x')$.

Note that the restrictions are obvious since otherwise $\mathcal{A}$ can trivially win the distinguishing game.

Generating and Distributing Related Inputs for PReFs. Let the inputs to Alice and Bob be the devices D_A and D_B. They form a PReF pair over the input set $X_{\mathrm{A,B}} \subset X$, that are functionally modeled as f_A and f_B. To compute this input set, the designer should learn the complete functionality of both the devices. If the functions f_A and f_B are known, then suitable machine learning algorithms can be used to mathematically model these functions. These models are then be fed to a SAT solver to obtain the inputs over which both the devices generate the same output. Here, there is a dependency on a trusted third party or setup who gains access to the mathematical models and generated the related inputs. The entities

with the PReF devices should trust that this setup process does not behave maliciously after gaining the knowledge of PReF functionalities and destroys this information after the related inputs are distributed. However, in the next section, we discuss how to generate and distribute these related inputs, without revealing the complete functionalities of the PReF devices.

A. Security Properties for PReFs

Here, we introduce two security properties for PReFs and to prove the security of PReF based protocols. Let the devices $(\mathsf{D}, \mathsf{D}')$, with input and output spaces X and Y respectively, form a pair of PReFs over the related input set $X' \in X$. Let $\mathcal{A}$ be an adversary who has access to neither the devices nor their functionalities. We informally present the *relation hiding* and *universality* properties of the PReFs as follows.

- **Relation Hiding**: It should be difficult for $\mathcal{A}$ to generate the related input set X' over which the outputs of two PReFs match, without learning the mathematical models or complete input/output behaviour of both the devices.
- **Universality**: It should be difficult for $\mathcal{A}$ to distinguish between $\mathsf{D}(x)$ and y where $x \xleftarrow{R} X'$ and $y \xleftarrow{R} Y$.

We formalize these properties below.

Definition II.3. Relation Hiding. Let $(\mathsf{D}, \mathsf{D}')$ be a PReF pair that are functionally modeled as $f : X \to Y$ and $f' : X \to Y$ respectively. The related input set is denoted as $X' \subseteq X$. For any security parameter λ, and for any probabilistic algorithm $\mathcal{A}$ that is $\mathsf{poly}(\lambda)$-time bounded and has oracle-access to f, f', define the event $\mathsf{E}_{\mathcal{A},b}(\lambda)$ for $b \in \{0,1\}$ as:

$$\mathsf{E}_{\mathcal{A},b}(\lambda) : \mathcal{A}^{f(\cdot),f'(\cdot)}(1^\lambda, x, f(x))$$

where x is uniformly randomly sampled as:

- $x \leftarrow X'$ when $b = 0$.
- $x \leftarrow X \setminus X'$ when $b = 1$,

subject to the restrictions that, $\mathcal{A}$ can make either the query $f(x)$ or the query $f'(x)$. $\mathcal{A}$ can make queries of the form $f(x'), f'(x')$, where $x' \neq x$ after the event $\mathsf{E}_{\mathcal{A},b}(\lambda)$.
The relation between f and f' is said to be *hidden*, if for any security parameter λ, for any input x satisfying the requirements above and for any such PPT algorithm $\mathcal{A}$, we have

$$\left|\mathsf{Pr}[\mathsf{E}_{\mathcal{A},0}(\lambda) = 1] - \mathsf{Pr}[\mathsf{E}_{\mathcal{A},1}(\lambda) = 1]\right| \leq \mathsf{negl}(\lambda)$$

Definition II.4. Universality. A PReF device instance D functionally modeled as $f : X \to Y$ is said to follow *universality* property if for *any* subset of inputs $X' \subseteq X$, the distribution $\left\{f(x)\right\}_{x \in X'}$ is statistically close to the uniform distribution $\{\mathrm{U}\}_Y$ over Y. In other words, for any subset $X' \subseteq X$ and for any $y \in Y$, we have

$$\sum_{x \in X'} \left|\mathsf{Pr}[f(x) = y] - \mathsf{Pr}[\mathrm{U}_Y = y]\right| \leq \mathsf{negl}(\lambda)$$

It should be noted that breaking the *universality* property of PReFs, in turn breaks the *relation hiding* property.

III. Improved PReF Construction

In this section, we present an improved architecture for PReFs we call as XOR_PReFs . We begin with the construction of XOR_PReFs followed by the methodology to find the related inputs. We obtain these inputs without revealing the secret XOR_PReF functionalities.

A. XOR_PReF: A New Construction

Let the devices $\mathsf{D}_1, \mathsf{D}'_1, \mathsf{D}_2, \mathsf{D}'_2$ physically implement the functions f_1, f'_1, f_2, f'_2 respectively over the input and output spaces $X = \{0,1\}^m$ and $Y = \{0,1\}^n$. Now, consider two devices D (ref. to Fig. 1) and D' defined as:

$$\begin{aligned} \mathsf{D} &: \mathsf{D}_1 \oplus \mathsf{D}_2, \\ \mathsf{D}' &: \mathsf{D}'_1 \oplus \mathsf{D}'_2 \end{aligned} \tag{1}$$

Then they physically implement the functions f and f' such that for any $(x_1, x_2) \in X \times X$:

$$\begin{aligned} f(x_1, x_2) &= f_1(x_1) \oplus f_2(x_2), \\ f'(x_1, x_2) &= f'_1(x_1) \oplus f'_2(x_2) \end{aligned} \tag{2}$$

Let $X_1 \subset X$ and $X_2 \subset X$ be the related input sets for the pairs $(\mathsf{D}_1, \mathsf{D}'_1)$ and $(\mathsf{D}_2, \mathsf{D}'_2)$ respectively. Then as per Definition II.1 for any input (u, v) where $u \in X_1$ and $v \in X_2$:

$$\mathsf{D}(u, v) = \mathsf{D}'(u, v)$$

In this way, we can say that the device pair $(\mathsf{D}, \mathsf{D}')$ forms a PReF pair for the related input set $X_1 \times X_2$. We formally define XOR_PReFs below.

Definition III.1. XOR Physically Related Function (XOR_PReF). Let $F : \{f_i : X \times X \to Y\}$, $G : \{g_i : X \to Y\}$ and $H : \{h_i : X \to Y\}$ be families of functions, where $X = \{0,1\}^m$ and $Y = \{0,1\}^n$. Let there be two devices D_i and D_j that physically implement the distinct functions $f_i, f_j \leftarrow F \times F$ such that for any $(u, v) \in X \times X$

$$f_i(u, v) = g_i(u) \oplus h_i(v)$$

and

$$f_j(u, v) = g_j(u) \oplus h_j(v)$$

Then, we say that the devices D_i and D_j form an XOR_PReF pair if

1) There exists a unique subset $X_{i,j} \subset X \times X$ such that for any $(u, v) \in X_{i,j}$:

   $$g_i(u) = g_j(u)$$

   and

   $$h_i(v) = h_j(v)$$

2) The devices D_i and D_j are *conditionally pseudorandom*, as per Definition II.2.
3) The pair $(\mathsf{D}_i, \mathsf{D}_j)$ follows *relation hiding* property as per Definition II.3. Note that, the adversary is restricted from outputting inputs of the form (u, v') or (u', v) if it has the knowledge about inputs $(u, v), (u', v') \in X_{i,j}$ during query phase.
4) Both D_i and D_j follow *universality* property as per Definition II.4

Next, we present a mechanism to generate the related inputs for XOR_PReFs.

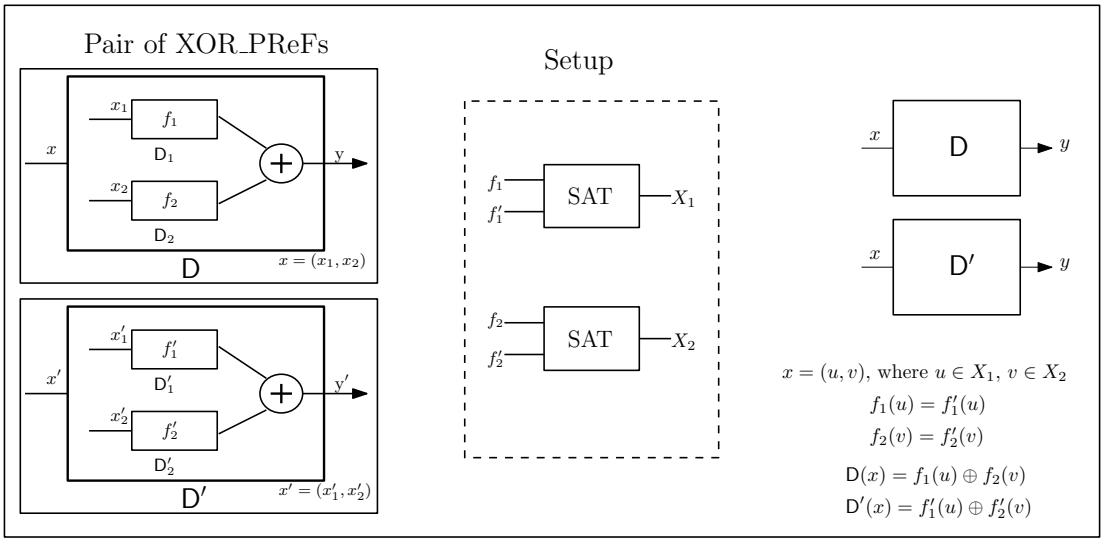

Figure 1: D and D' are two XOR_PReF devices that physically implement the functions $f = f_1 \oplus f_2$ and $f' = f_1' \oplus f_2'$, with input space $X \times X$ and output space Y where $X = \{0,1\}^m$ and $Y = \{0,1\}^n$. $X_1 \subset X$ is the related input set for the PReF pair $(\mathsf{D}_1, \mathsf{D}_1')$ and $X_2 \subset X$ is the related input set for the PReF pair $(\mathsf{D}_2, \mathsf{D}_2')$. For any $u \in X_1$ and $v \in X_2$, the input to XOR_PReF is $x = (u, v)$ and the output is $y \in Y$ such that $\mathsf{D}(x) = \mathsf{D}'(x) = y$.

B. Generating Related Inputs for a pair of XOR_PReFs

In this section, we generate the related inputs for a pair of XOR_PReFs while preserving the privacy of their functionalities, with the help of two non-colluding semi-honest third parties TP_1 and TP_2.

Let $(\mathsf{D}, \mathsf{D}')$ be a pair of XOR_PReFs that physically implement the functions $f = f_1 \oplus f_2$ and $f' = f_1' \oplus f_2'$ respectively. We denote the input and output spaces for functions f, f' as $X \times X$ and Y. For $i \in \{1, 2\}$, the inputs to TP_i are the functions f_i and f_i'. TP_i generates the related input set $X_i \subset X$ such that for any $x \in X_i$

$$f_i(x) = f_i'(x) \tag{3}$$

Then the related input set for the pair $(\mathsf{D}, \mathsf{D}')$ is defined as

$$\langle X_1, X_2 \rangle = \Big\{ (u, v) \mid u \in X_1, v \in X_2 \Big\} \tag{4}$$

such that for any $(u, v) \in \langle X_1, X_2 \rangle$

$$\mathsf{D}(u, v) = \mathsf{D}'(u, v) \tag{5}$$

as per Eq. 2 and Eq. 3.

Here, we make the following observations:

- Since TP_1 learns only (f_1, f_1') and the functions (f_2, f_2') are conditionally pseudorandom, it is difficult for TP_1 to compute the final output of D or D' over inputs that are not in the related input set. Similar argument can be made for TP_2.
- If the functions f_2, f_2' follow the *relation hiding* property, then it is hard for TP_1 to find their related input set X_2. Therefore, TP_1 cannot learn the final output of D and D' over related inputs. Similar argument can be made for TP_2.
- Related inputs can be made public if (f_1, f_2) and (f_1', f_2') follow *universality* property.

In this way we generate the related input set for a pair of XOR_PReFs without revealing their complete functionalities.

IV. Experimental Results

In this section, we implement a prototype system for evaluating our XOR_PReF based commitment scheme. For our XOR_PReF implementation, we use the PReF design proposed in [3].

A. Hardware realization of the XOR_PReF construction

PUFs are used for proof-of-concept (POC) realization of PReFs in [3]. They used the PUF design presented in [4], which takes a 64-bit binary input and outputs a 128-bit binary response. For POC realization of XOR_PReFs, we use the same PUF design to implement the internal PReFs as shown in Fig. 1. We implemented our XOR_PReF design on 20 Artix-7 FPGAs to realize 10 XOR_PReF pairs. We evaluate the quality of XOR_PReF using PUF metrics: uniqueness and uniformity. Uniqueness is the inter-distance between two PUFs for any random challenge. Uniformity of a PUF denotes the distribution of $0'$s and $1'$s in the output space. The ideal value for both is 50%. For the 20 XOR_PReFs, the observed uniqueness is [44-46]% and uniformity is [41-45]%, over 20K randomly generated inputs. Below, we experimentally prove that the XOR_PReF design follows the relation-hiding and universality properties as defined in Def.II.3 and Def. II.4.

Relation-hiding. To prove this property, we show that the probability with which a pair of XOR_PReF devices produce the same output for a *randomly* chosen 64-bit input is negligible. Let y and y' denote the $m-$bit binary outputs of the XOR_PReF pair. Then, for all $i \in [1, n]$, let the probability with which $y[i] \neq y'[i]$ be p. In [3], the authors presented the

Table I: The probability$(1-\mathsf{p})$ with which each output bit matches for XOR_PReF pairs **1**-**10**.

1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
0.592	0.536	0.578	0.537	0.583	0.591	0.541	0.532	0.579	0.536

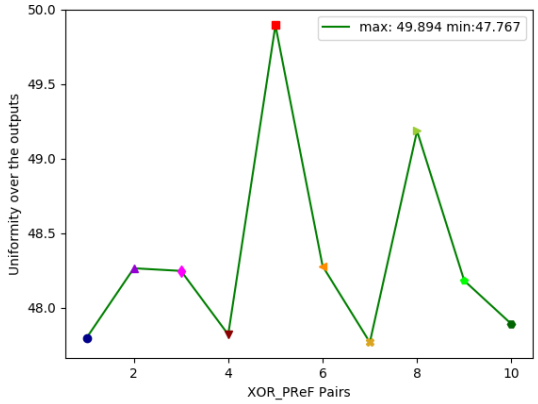

Figure 2: Uniformity of 10 XOR_PReF pairs.

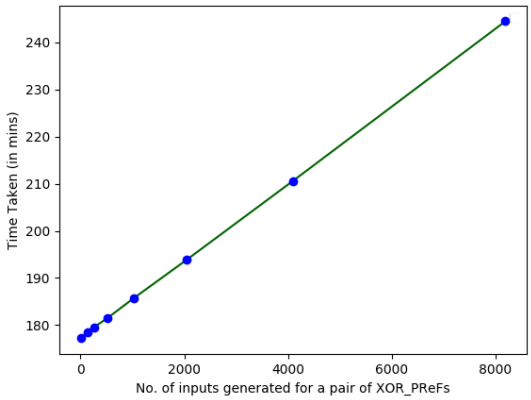

Figure 3: Cumulative graph representing the time taken to generate inputs for a pair of XOR_PReFs.

probability with which the Hamming distance (HD) between the outputs of two PReF devices is j by

$$\mathsf{Pr}[\mathrm{HD}(y,y')=j]=\sum_{k=0}^{j}\binom{n}{k}\mathsf{p}^k(1-\mathsf{p})^{n-k}. \quad (6)$$

From here, we can calculate the probability with which all the bits match as

$$\mathsf{Pr}[y=y']=n(1-\mathsf{p})^n. \quad (7)$$

The ideal value of $p=0.5$, since the outputs are uniform and are independent of each other. We calculated the probability with which each output bit mismatches for the 10 XOR_PReF pairs using 20K challenges as presented in Tab.I. Then the probability with which XOR_PReF 2 (as shown in Tab.I) outputs same 128-bit string for a randomly chosen 64-bit string is atmost 2^{-109} (see Eq.7), which is negligible. Therefore, the adversary who has no knowledge of the XOR_PReF functionalities can break the *relation hiding* property only with negligible probability.

Universality. For each XOR_PReF pair $(\mathsf{D}_1,\mathsf{D}_2)$,we generate the input subset X', such that theoutputs of both devices is same. The XOR_PReF pair is said to follow *universality*, if the outputs are uniform, given X' as input. To prove this experimentally, we generated the set X' of size 100K for each of the 10 XOR_PReF pairs. We measured the uniformity of XOR_PReFs as described Fig. 2 over these inputs. We can observe that the uniformity lies between 47.76 and 49.894. Thus, we can say that the XOR_PReF pairs follow *universality* property.

V. Conclusion

In this paper, we introduced XOR compositions of PReFs called as XOR_PReFs, that eliminate the requirement to reveal the complete functionality of the PReF devices to generate the related input sets. We proposed two security properties for PReFs, that can be used to prove the security of PReF-based protocols. We validated the quality of our construction by implementing them on Artix-7 FPGA boards and show that they match with the theoretical assumptions.

As future work, we will build XOR_PReF-based oblivious transfer protocols for resource-constrained devices, that do not require secure storage or physical transfer of the hardware devices.

References

[1] S. P. Skorobogatov, "Semi-invasive attacks: a new approach to hardware security analysis," Ph.D. dissertation, University of Cambridge, UK, 2005.

[2] M. N. I. Khan and S. Ghosh, "Comprehensive study of security and privacy of emerging non-volatile memories," *CoRR*, vol. abs/2105.06401, 2021.

[3] D. Chatterjee, H. Boyapally, S. Patranabis, U. Chatterjee, D. Mukhopadhyay, and A. Hazra, "Physically related functions: A new paradigm for light-weight key-exchange," *IACR Cryptol. ePrint Arch.*, vol. 2021, p. 389, 2021. [Online]. Available: https://eprint.iacr.org/2021/389

[4] U. Chatterjee, D. P. Sahoo, D. Mukhopadhyay, and R. S. Chakraborty, "Trustworthy proofs for sensor data using FPGA based physically unclonable functions," in *2018 Design, Automation & Test in Europe Conference & Exhibition, DATE 2018, Dresden, Germany, March 19-23, 2018*, J. Madsen and A. K. Coskun, Eds. IEEE, 2018, pp. 1504–1507. [Online]. Available: https://doi.org/10.23919/DATE.2018.8342252