Physically Related Functions: A New Paradigm for Light-weight Key-Exchange

Durba Chatterjee¹, Harishma Boyapally¹, Sikhar Patranabis², Urbi Chatterjee³, Debdeep Mukhopadhyay¹, Aritra Hazra¹
1. Indian Institute of Technology Kharagpur, 2. ETH Zurich, 3. Indian Institute of Technology Kanpur

ABSTRACT
In this paper, we propose a novel concept named Physically Related Function (PReF) which are devices with hardware roots of trust. It enables secure key-exchange with no pre-established/embedded secret keys. This work is motivated by the need to perform key-exchange between lightweight resource-constrained devices. We present a proof-of-concept realization of our contributions in hardware using FPGAs.

Keywords: Boolean functions, Key-exchange protocol, Physically Related Functions

1 INTRODUCTION
The most straightforward way of achieving secure key-exchange is via standard cryptographic techniques, where an initially exchanged secret key allows the devices to encrypt all ensuing communication. However, these devices are far too resource-constrained to either support the heavy protection mechanisms against invasive and semi-invasive attacks [12] associated with secure key-storage (as in symmetric-key primitives) or does not have provisions for renewal of certificates (as in asymmetric-key primitives). This is a scenario frequently encountered today with the widespread advent of Internet of Things (IoT) and Cyber-Physical Systems (CPS). Recently, Amazon, Uber, and UPS and other major corporations have announced plans to launch commercial autonomous drone operations [4, 7]. The widespread deployment of such technologies would potentially involve millions of devices communicating with each other. This makes it essential to design light-weight protocols for secure communication that can scale to a large number of devices.

In this paper, we introduce novel hardware primitives called Physically Related Functions (abbrev. PReFs). We present a PReF-based on-the-fly key-exchange scheme, without the need to store any secret key or the need to contact any trusted third party. We present a proof-of-concept (PoC) realization of PReFs in hardware using 84 separate Xilinx Artix 7 FPGAs.

2 PHYSICALLY RELATED FUNCTIONS
In this section, we present a brief description of PReFs and a PReF-based key-exchange scheme. A pair of PReFs (DA, DB) physically implement the functions (fA, fB) with input space X and output space Y such that there exists a specific subset of inputs XAB ⊆ X such that the output behaviors of the functions fA and fB on each input in XAB are correlated with respect to some distance metric (eg. Hamming Distance (HD)). On the other hand, for any x ∈ X \ XAB any probabilistic poly-time bounded algorithm that only has access to an implementation of fA and does not not have (even blackbox) access to an implementation of fB cannot distinguish fB(x) from random. This is defined as the pseudorandomness property of PReFs.

DAB: Round 1
(1) Computes yA = fA(x), A = E(r) + yA
(2) Sends (x, A) to DB

DB: Round 1
(1) Computes yB = fB(x), r’ = D(A + yB)

Figure 1: Basic Key Exchange

PReF-based Key Exchange Scheme
Now, we develop a key-exchange scheme for secure communication between two devices, each embodying a PReF instance. Our protocol precludes the usage of long-term secure key storage, as is usually the case with a large class of key-exchange protocols based on traditional cryptographic approaches [1, 10]. In other words, it avoids not only the need for dedicated key storage but also the associated countermeasures for preventing potential physical attacks (both invasive and non-invasive) targeting such key storage. It is also superior to the key-exchange schemes based on alternative primitives (such as Physically Unclonable Functions (PUFs)) which are asymmetric in nature requiring one (or both) device(s) to perform complex computations or require trusted third party during the protocol run. So, in a way, our scheme achieves the best of both worlds, especially in the context of lightweight resource-constrained devices.

Protocol Description. Let (DA, DB) be a pair of PReFs as described previously, such that X = {0, 1}^n and Y = {0, 1}^m. These devices form a PReF pair over the input subset XAB ⊂ X such that for any x ∈ XAB, HD(f(x), q(x)) ≤ δ. Let (E, D) be the encoding and decoding algorithms of an (n, k, δ) linear error correction code (ECC).

We present a basic PReF-based key exchange protocol as described in Fig. 1, that requires no computational resources beyond evaluating PReF outputs. It enables a key exchange between two PReF devices DA and DB with the unique "related" input set XAB. The protocol involves a single round of communication between the devices and is considerably light-weight given that it only uses error-correcting codes in addition to PReFs.

Theoretical Implications. This protocol also has some interesting theoretical implications about the computational power of PReFs. It is well-known that no computationally secure key-exchange protocol (even multi-round) can be based in a black-box manner on purely symmetric-key cryptographic primitives such as pseudorandom functions or symmetric-key encryption [8]. The fact that we can bypass this impossibility result using only PReFs and no additional cryptographic primitives/trusted parties indicates that PReFs are, in fact, more powerful than simple symmetric-key cryptoprimitives. This makes PReFs an interesting object of study from a cryptographic standpoint.
3 REALIZING PREFS IN HARDWARE

In this section, we establish the feasibility of embedding “related” functions into physical devices, i.e., the feasibility of realizing PReFs in hardware. We show that the notion of correlation with respect to Hamming distance (HD) allows us to obtain a set of “related” inputs for any pair of random Boolean functions.

Correlation Analysis of Boolean Functions. In Boolean theory, the cross-correlation function [11] is used to study the cryptographic properties of Boolean functions. The cross-correlation function for two Boolean functions \( f : X \to Y \) and \( g : X \to Y \) where \( X = \{0, 1\}^n \) and \( Y = \{0, 1\} \) calculates the correlation between its outputs over the complete input set \( X \). It is given by:

\[
C_{f,g} = \frac{1}{|X|} \sum_{x \in X} (-1)^{f(x) \oplus g(x)}
\]

and its value lies in \([-1, 1]\). Now, if the functions \( f \) and \( g \) are chosen uniformly at random from the space of all \( n \)-bit Boolean functions, their outputs will be statistically uncorrelated. Hence, \( C_{f,g} = 0 \).

For this work, we have exploited the fact that even if \( f \) and \( g \) are statistically uncorrelated, there exist some inputs for which both have the same outputs. We can split \( X \) into two disjoint subsets \( X_0 \) and \( X_1 \) such that \( f(x) = g(x) \) \( \forall x \in X_0 \) and \( f(x) \neq g(x) \) \( \forall x \in X_1 \). Therefore, we can say that \( f \) and \( g \) are related over the subset \( X_0 \). For a more generic analysis, consider the functions \( f_A, f_B : X \to Y \), where \( X = \{0, 1\}^n \) and \( Y = \{0, 1\}^m \). To find the input subset over which \( f_A \) and \( f_B \) are related, we split the input set into disjoint subsets \( \{X_0, X_1, \ldots, X_n\} \), such that for any input belonging to subset \( X_i \), the HD between the function outputs is \( i \). Note that the HD is calculated over the \( m \)-bit response.

Let \( f_A(x)[i] \) denote the \( i \)th bit of the output of \( f_A \) for an input \( x \) and let \( q \) be the probability with which \( f_A(x)[i] \oplus f_B(x)[i] = 1 \) occurs, for any \( i \in \{1, \ldots, m\} \). The probability that \( \text{HD}(f_A(x), f_B(x)) \) takes the value \( j \in \{0, m\} \) can be given as:

\[
Pr[\text{HD}(f_A(x), f_B(x)) = j] = \binom{m}{j} q^j (1 - q)^{m-j}.
\]

From the above equation, it is evident that the frequency distribution calculated using the HD follows a Binomial distribution.

Let \( e_{AB} \) denote the probability with which \( \text{HD}(f_A(x), f_B(x)) \leq \delta \) holds. We can calculate \( e_{AB} \) as:

\[
e_{AB} = Pr[\text{HD}(f_A(x), f_B(x)) \leq \delta] = \sum_{j=0}^{\delta} \binom{m}{j} q^j (1 - q)^{m-j}.
\]

Then the size of the input subset \( X_{AB} \subseteq X \), over which the outputs of \( f_A \) and \( f_B \) have HD almost \( \delta \) is given as:

\[
|X_{AB}| = e_{AB} |X|.
\]

A pair of functions \( f_A \) and \( f_B \) are said to be related if \( X_{AB} \neq \emptyset \) and \( e_{AB} > 0 \). Thus with this notion of output-correlation, we can obtain “related” inputs for any pair of random Boolean functions.

The overview of the correlation analysis for a pair of PReFs is illustrated in Fig. 2.

Sampling Related Inputs for PReF Devices. Equipped with this analysis, our aim is to realize PReFs using hardware devices. For this, we simply design Boolean functions as hardware circuits and rely on the internal variability of every device to introduce unpredictable yet repetitive randomness in the circuit behaviour. The next step is to identify the inputs over which the functions produce “related” outputs. Since the functionality is fixed and unpredictable, we resort to a machine-learning and SAT [6] based reverse-engineering approach described as follows. The idea is to generate accurate machine-learning models of each PReF instance in setup phase prior to deployment, and use these models to obtain “related” challenges. The steps are described as follows:

Related Input Set Identification. We use some of the well-known learning techniques such as Logistic Regression (LR) [9], Support Vector Machine (SVM) [5], and Random Forest (RF) [2] algorithms to build, for each elementary component of PReF (hardware-embedded function or the Strong PUF embedded in hardware device), a corresponding mathematical model that closely approximates the input-output behaviour of the component. Since the modelling is expected to be done at setup (i.e., at the device manufacturing stage), we crucially exploit the knowledge of the circuit-internals, in particular the input-output behaviour of each intermediate stage of the circuit, to build an overall model with close to 100% accuracy. After building the mathematical models \( M_{f_A} \) and \( M_{f_B} \) approximating the hardware-embedded functions \( f_A \) and \( f_B \) respectively. The next step is to formulate the problem of identifying the corresponding related set of inputs \( X_{AB} \) into a Boolean satisfiability problem in terms of the model parameters for pair \((M_{f_A}, M_{f_B})\).

Mathematical Formulation of Models. Depending on the chosen machine learning algorithm, we formulate a mathematical expression for the output of functions (corresponding to each component) in terms of the obtained model parameters and the input.

Note that the resultant mathematical expression depends not only on the implemented circuit but the instance-specific system.
We observe that the probability that two devices establish the same inputs over which a pair of related devices can communicate. We observe that the probability that two devices establish the same key is 100%, for all the 1000 inputs. We use the same inputs to find the probability (FPR) that an illegitimate device (not related over the chosen input subset) can successfully exchange the key with a legitimate device. The FPR of the protocol is observed to range from $2^{-90}$ to $2^{-110}$ for all the 1000 inputs.

Next, we find the true positive rate (TPR) and false positive rate (FPR) of protocol. To calculate TPR, we randomly choose 1000 inputs and calculate HD. We observe that the probability that two devices establish the same key is 100%, for all the 1000 inputs. We use the same inputs to find the probability (FPR) that an illegitimate device (not related over the chosen input subset) can successfully exchange the key with a legitimate device. The FPR of the protocol is observed to range from $2^{-90}$ to $2^{-110}$ for all the 1000 inputs.

5 CONCLUSION AND FUTURE WORK

In this paper, we initiated the study of Cryptophasia in Hardware using a novel concept called Physically Related Functions (PReFs) that have not been studied before to the best of our knowledge. We demonstrated how PReFs can be used to establish a key-exchange scheme with no pre-established secure channels and no secure storage for cryptographic keys. We established the feasibility of our proposal via concrete prototype implementations and extensive experimental implementations on Artix-7 FPGAs. As a future work, we will build authenticated key-exchange schemes for a network of resource-constrained devices, ensuring secure communication between any two devices.

REFERENCES


