Cyber-Physical Security Using System-Level PUFs

Omar Al Ibrahim, Suku Nair, *Southern Methodist University*

***Abstract*— Cyber Physical Systems (CPS) is an emerging computing paradigm that is becoming prevalent in various technologies. Achieving a trustworthy CPS requires us to build mechanisms that ensure the integrity and authenticity of these systems. Fortunately with the new advancements in semi-conductor-based technologies, in particular Physical Unclonable Functions (PUFs), we have the potential to build secure couplings between cyber and physical substrates based on intrinsic physical material. In this paper, we share some thoughts on how to utilize the PUF technology for security in CPS. Based on a composition approach, we illustrate the benefits of combining multiple PUF elements, with some inherently bias factor, into one randomly secure and strong system-level PUF.**

I. INTRODUCTION

Cyber Physical System (CPS) [1-4] is a new term that refers to the coupling of cyber elements with the physical environment. This trend in computing enriches the interactions between the cyber and physical substrates, which is governed by fine-grained temporal-spatial scales and is operated by complex, dynamic, and context-aware framework.

Over the last several years, there have been many discussions about CPS, virtually all aspects of its structure, control behavior, and communication. One of the most important topics out of all of these discussions is cyber-defense. Cyber-defense is considered of paramount importance because the nature of events in CPS usually manifest itself into some physical action, which implies that these systems need to be safeguarded against wrong control decisions, regardless whether these wrongful decisions were caused by an unexpected fault or a malicious intervention.

Several positioning papers have arguably described the challenges of cyber-defense in CPS [1,2,4]. Generally speaking, security under CPS requires deep understanding on the confluence of device proliferation, autonomy, and integration at scale, which open the door for an entirely new security paradigm: one that acknowledges the resource limitations of edge devices and the scalability challenges of high-end systems.

The advent of new semiconductor-based technologies namely Physical Unclonable Functions (PUFs) have provided powerful yet cost-effective security constructs for embedded systems. These constructs could potentially be used to secure cyber elements in CPS. In this paper, we define a new concept for PUF that is applicable to CPS. This PUF is considered strong and secure, and is achieved from the amalgamation of small elementary PUF circuits on the edge devices. We also spark some ideas on different interactions of the system-level PUF using a composition approach.

II. PHYSICAL UNCLONABLE FUNCTIONS (PUFS)

In this section, we give an overview on Physical Unclonable Functions (PUFs) before discussing system-level PUFs.

A. Overview

Physical Unclonable Functions (PUFs) [5-12] are special circuitry that makes use of variations in the manufacturing process to generate a reproducible set of unpredictable numbers. These numbers are obtained by sending a stimulus, which act as a challenge to the circuit, to obtain a corresponding number that represents the response. Every PUF circuit holds a set of challenge-response numbers that distinguish it from other PUF circuits, and yet these circuits could all be mass manufactured with the same process. In addition, PUF is considered an expedient tool for key generation because the responses are stored in the physical medium. Hence, PUF-based devices do not require tamper-proof memory to protect the secret keys.

PUFs are known to be very efficient since they can be easily manufactured using a few number of gates. Today, there exist many types of PUFs. For example, one implementation of PUFs uses delay elements and an arbiter circuit that define a race condition between two lines (Figure 1). This type of PUF, known as Silicon PUF [8,12], is available in the market and has been produced for RFID and FPGAs. Some PUFs [5,10] can also be coated onto physical objects, thereby strengthening the bond between the cyber and physical parts in a CPS.

B. Authentication protocols

As mentioned, PUF serves as a device-centric alternative for security. Several papers investigated the use of PUF to build authentication protocols. In the early works, protocols, such as the one proposed by Suh and Devedas [12], were based on simple challenge-response mechanisms that pre-load PUF responses into the system, which are then used as one-time authentication tokens. Due to scalability issues, some papers [13] suggested a probing scheme that applies an auxiliary encryption algorithm to securely acquire the responses from PUF. However, the dilemma behind these approaches comes from the key distribution factor. Alternatively in [14], a new protocol, called OCCRA (Overt-Covert Challenge-Response Authentication), was proposed to resolve these issues. OCCRA constructs an embedded sequence of challenge-response numbers that will enable authentication without additional hardware investment on key-sharing. The sequences are generated using an oracle-based mechanism to refresh the system state. Extending on this, it would thus be of interest to authenticate a response pattern from multiple PUF elements at the system-level.

978-1-4244-9538-2/11/$26.00 ©2011 IEEE

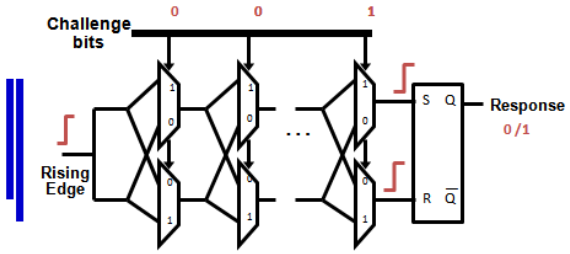

Figure 1. Sillicon PUF

III. System-Level PUF

Our idea of a system-level PUF is inspired by the long-driven discussions by the community about having an integrated cyber-defense framework for CPS. A system-level PUF is a logical PUF that describes the composite behavior of multiple PUFs, and which are then used to establish system-level properties for security. To better understand this concept, let us pounder on some of the applications.

A. Applications

- Vehicle systems: In 2005, it is estimated that about 30-90 processors were used in some automobiles for engine control, break system, and airbag deployment management. Sensors and actuators were networked to provide safety alerts, autonomous navigation were employed to provide location-based services, and telematics were used to provide a cyber-outreach to the networking infrastructure. In recent times, there has been a rampant surge with respect to counterfeit parts that could compromise on the security and safety of vehicles, especially those involving complex assemblage of interconnected systems. The embedded intelligence that we see in advanced automotive systems need to be protected through aggregative security mechanisms that will not curb the mass manufacturing process in this industry.
- Medical devices: With the new advances in computing capabilities, old generation of electromechanical instruments are being replaced by diverse arrays of high-tech medical devices. These emerging families of medical devices are often equipped with network capabilities that link to other contemporary equipments, forming systems with increasingly complex configurations. From home care monitoring and control to the operating room of the future, today's medical devices demand high-levels of system integration.
- Power grids: As global energy reserves are continuing to deplete at alarming rates, there is a necessity to incorporate renewable energy sources into the grid system. To achieve this, CPS is needed in the power grid to reduce emissions and automate operations. Accordingly, security in these environments must scale to large numbers of smart control devices.

B. PUF interactions

The applications we briefly discussed set forth the requirements for establishing security in CPS. In short, any security mechanism in CPS will have to address the inter-play of concurrency, integration, and ubiquity of the components. The PUF technology seems promising in this prospect because it couples the physical and cyber elements at the component-level in such a way that it allows us to build efficient challenge-response systems. However, we need to bring this to the system-level.

We define manifold combinations of interactions between PUF circuits. In the most basic forms, a set of PUFs can be combined using parallel interactions, series interactions, or feedback interactions.

In the parallel-interaction PUF system (Figure 2), a response pattern is obtained using a simultaneous execution of multiple PUFs. The pattern is then fed to an aggregation function that generates a system value for security. The choice for the aggregation function will depend on the randomness characteristics of the responses. That is, if the responses are fairly random and uncorrelated with respect to the input, then an XOR operation is sufficient. Otherwise, a randomness extraction step, possibly using some crypto-algorithm or permutation, is deemed necessary. There are two ways to handle the input in the parallel-interaction system. One approach is to feed the system with a synced challenge, the other approach to feed the system with parallel input combinations such that each PUF has a distinct challenge.

Another type of interaction that could be defined is a series interaction. In a series interaction, the result of the challenge from one PUF is propagated to another PUF in a cascade fashion, such that the challenge to the first PUF in the series represents the system challenge and the response of the last PUF in the series represents the system response. In contrast, the feedback interaction will loop back to a previously applied PUF to generate the system response.

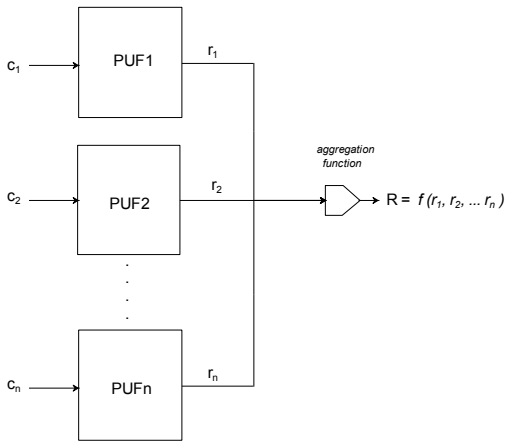

Figure 2. Composition PUF system: PUF system fed with parallel-input combinations (c_1 , c_2 , ..., c_n). The generated response R is a function of the response pattern (r_1 , r_2 , ..., r_n) that is processed and computed by the high-end system.

System-level PUFs have several advantages not present with the original PUF circuits. First, system-level PUFs provide a unique way of extracting strong random properties from multiple PUFs that could be inherently biased and vulnerable to model-building. Second, in a system-level PUF, a heterogeneous combination of genetic material including biometrics and various types of PUFs, such as Arbiter, Ring-Oscillator, and Coating PUFs, can be joined together to produce system-level characteristics.

C. Architecture

The architecture of the system-level PUF consists of a system of embedded components, each equipped with PUF circuits. Each component is also equipped with a communication module that enables wireless transmission of PUF responses and receipt of reader signals. The architecture also consists of a group of readers acting as cluster heads with the communication model limited to a challenge-response system between the reader and the components. Furthermore, all responses are processed off-chip and no processing and memory overhead are incurred on the components.

D. General authentication scheme

The back-end system is initially preloaded with challenge-response pairs for each PUF which are presumed to be collected under well-controlled conditions. The trust party, who initially has possession of the PUFs, collects challenge-response pairs that are used as one-time pads. These PUFs are coated as embedded chips on each of the components. A system value is computed for the embedded system by sending a stimulus challenge to the PUFs of the components, and then aggregating the responses using a simple function.

To verify the integrity of the components, the trusted party selects a challenge from one of the pre-stored pairs and sends it to each PUF. Each component obtains the response from PUF given the challenge as input, and transmits the response to the trusted party. The responses are collated together to obtain the system response. If the system response matches the recorded one within some acceptable threshold (i.e. Hamming distance), then the components are verified. Otherwise, the system will need to move to component-level authentication to determine which components caused the authentication to fail. Since the components are the only entities that know the challenge-response pair, the trusted party accepts the responses given the threshold is not too large to introduce false-positives.

IV. Error-Correction Techniques For PUFs

Controlling the noise-factor for the outputs produced is an open problem in system-level PUFs. One of the important factors that affect the behavior of PUF is intra-chip variation. Intra-chip variation is a noise-factor in PUFs caused by environmental changes in voltage and temperature. In a system-level PUF, when multiple PUFs are combined using the interactions, noise is accumulated in the system response. This factor is even more amplified with nested interconnections. To ensure the reproducibility of the outputs, we need to turn to error-correction techniques.

In this section, we describe several error-correction techniques that are applicable to system-level PUFs. As mentioned, intra-chip variation is an inert noise factor in PUF caused by extreme changes in environmental conditions. Several papers studied the effects of intra-chip variation on different types of PUFs. In [8], intra-chip variation was examined for Silicon PUFs, and have experimentally been demonstrated to vary roughly from 4.8 percent and 3.7 percent respectively for room and extreme temperatures [12], and when combined with extreme voltage variation, the output noise reaches up to 9 percent. Similarly, another study was conducted for the Ring-Oscillator PUF, which showed variation from 3 to 4 bits out of 128 bits (which means that the average variation is approximately 0.48 percent of the total bits).

By measuring these variations, it is possible to compute the failure rates of an authentication scheme using PUFs. Though these variation rates appear to be marginal, as indicated by the studies; notwithstanding, when constructing the system-level PUF, it is worth noting that they could accumulate and/or generate random discrepancies in the final output. To reduce the noise incurred in the final output, we enumerate a list of approaches from the open literature to tackle this problem. The error-correction techniques can be applied at various stages in the system-level PUF, and between the evaluations of the interacting PUF elements, in order to ensure stable output characteristics.

A. Traditional coding schemes

Coding is an established research and practice, especially in information redundancy, where check bits are added to the data, to allow verification, and in some cases, even correction of erroneous data. Several commonly used error-detecting and error-correcting codes could be used for PUFs to generate consistent outputs even with significant fluctuations in the environmental conditions. The general approach is as follows: In the initialization step, an error-correction syndrome is generated from the outputs of each of the PUF elements. Then, the syndrome and corresponding outputs are saved, either on-chip, off-chip, or remotely to a server.

When the PUF output is re-generated at a later time, the stored syndrome is used to correct the changes to reproduce the same output from the initialization step. Since the error-correction syndrome is likely to be a publicly known value, there will be some entropy loss to the outputs.

B. Fuzzy extractors

An elegant way to account for the entropy loss in error-correction codes is to employ a fuzzy extractor. A *fuzzy extractor* [19] couples error-correction with a randomness extraction step to generate a uniformly random and fixed response from an error-prone input. Fuzzy extractors were first introduced in biometrics to generate a cryptographically secure key from a person's fingerprint or iris scan. They are also used for non-uniform inputs such as long pass-phrases, questionnaires, handwritten signatures, and voice commands.

Fuzzy extractors are carried out in two steps. The first step, error-correction, involves the use of a non-separable syndrome which is acquired during the enrollment of PUF in the system. The parameters of the error correction code are determined by

the length of the PUF response and the number of errors that have to be corrected. The code distance is chosen to specify the error-correction capability. In the second step, a hash function is used to randomize the output to generate a key. In [21], an information-theoretic study of fuzzy extractors demonstrated that the entropy loss from the syndrome is minimal provided some randomness properties.

In [5,9] , Philips researched the application of fuzzy extractors with Coating PUFs. In their experimentations, they stored the syndrome on chip. The coating is used to generate keys from sensor measurements which are employed by the device. The coating key is transmitted to a fuzzy extractor component, from which a fixed random key is generated with the use of the syndrome bits, and is used for encryption (e.g. AES).

C. *Index-based syndrome*

In , Mandel and Devadas proposed a syndrome coding scheme that help limit the amount of information leaked by error-correction codes. The basic idea is to generate pointers to values in the PUF output sequence. These generated pointers are not directly proportional to the outputs, and unlike coding schemes, no bitwise masking is used to produce the syndrome.

However, the approach only works well for PUFs with real-valued outputs. These real-valued outputs, generated using Ring Oscillator PUFs, contain two pieces of information: a polarity bit (1 or 0) and a string of bits (i.e. a real number) that indicates the confidence-level of the polarity bit. A soft decision encoder/decoder is employed to utilize these two pieces of information to generate high code gains with reduced entropy loss. Similar to fuzzy extractors, index-based syndrome can be integrated with PUF. Though not suitable for single-bit outputs, index-based syndrome was demonstrated to significantly reduce bit errors for PUFs with real-valued outputs.

V. Fusion Of Physical Properties

After we presented some of the contemporary literature on error-correction techniques available for PUFs, and how this contributes to stabilizing the outputs, we now complete the discussions by illustrating some of the virtues of system-level PUFs.

Today, the PUF technology serves as a distinct way of bridging physical characteristics with its cyber counter-parts in CPS, and though it showed promising for physical security, current PUF implementations are far beyond cryptographically random characteristics. In fact, several works exposed weaknesses in current PUF implementations including lack of resistance to modeling attacks. In [20], a linear model was derived for delay-based PUF using certain characteristics about the inter-switch and intra-switch delay variations. Other works also described successful attacks on standard Arbiter PUFs and Feed-Forward Arbiters with one loop. In a very recent paper [21], modeling attacks targeted several classical implementations including Ring-Oscillator PUFs, and others. These attacks were based upon various existing machine learning techniques such as logistic regression and evolution strategies. Results also showed that a centralized algorithm can be devised to impersonate various types of PUFs and behaves almost indistinguishably given a scalable challenge-response subset.

Nonetheless, the PUF technology does bring a physical system with structural disorder characteristics. However, in order to leverage from these physical properties, security must go beyond a single PUF. It is for these reasons that we are pushing for the concept of a system-level PUF, a type of PUF achieved from the fusion of many PUF elements. In essence, this concept is part of a more abstract viewpoint of security, a paradigm which we call *security fusion* [22]. In the security fusion framework, strong security properties are achieved through the collation of multiple strands of primitive properties. Throughout our research, we explored new theoretical frameworks to achieve security fusion, but we believe, as we contemplate through the emerging physical technologies, that we can utilize genetic properties of PUFs and other physical substrates to the security of CPS.

VI. Conclusion

In this paper, we have introduced a new framework for security in CPS using system-level PUFs. Specifically, we have motivated for a composition approach to collate the security properties of multiple PUF elements. We outlined some of the underlying challenges in the reproducibility factor, that when elucidated, will move us closer to a realizable solution.

It would be interesting to research new fusion techniques that can be used in conjunction with PUFs to derive a single metric for scalable system-level security. We hope that such an approach will address key shortcomings of PUFs including handling output noise due to environmental changes as well as modeling attacks of the physical microstructure. Our thoughts had pinpointed us into two directions: one is to improve the physical properties of PUF by constructing an embedded PUF system from various inherently weak genetic properties; the other direction is to push for a high-end approach.

In retrospect, we realize that a composition approach at the circuit-level is limited and not fail-safe, especially for edge components that cannot carry out a complex maneuver for security, but at the same time if we were to investigate a high-end approach, the theoretical bounds and performance results should present an order-of-magnitude advantage over simple node-level verification. Currently, these questions remain open problems. Nonetheless, system-level PUFs can potentially provide a secure coupling of cyber and physical elements in order to build dependable and resilient systems.

References

[1] E. Lee, "Cyber Physical Systems: Design Challenges," in *IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC)*, 2008, pp. 363 - 369.

[2] A. Platzer, "Verification of Cyberphysical Transportation Systems," *IEEE Intelligent Systems*, vol. 24, no. 4, pp. 10-13, 2009.

[3] A. Saber and G. Venayagamoorthy, "Efficient Utilization of Renewable Energy Sources by Gridable Vehicles in Cyber-

Physical Energy Systems ," *IEEE Systems* , vol. 4, no. 3, pp. 285-294, 2010.

[4] W. Wolf, "Cyber Physical Systems," *Computer*, vol. 42, no. 3, pp. 88-89, 2009.

[5] M. Asim, J. Guajardo, S. Kumar, and P. Tuyls, "Physical unclonable function and their application to vehicle system security," in *IEEE Vehicle Technology Conference*, Barcelona, 2009, pp. 1-5.

[6] L. Bolotnyy and G. Robins, "Physically unclonable function-based security and privacy in Rfid systems," in *Fifth Annual IEEE International Conference on Pervasive Computing and Communications, 2007. PerCom '07.* , White Plains, NY, 2007, pp. 211-220.

[7] K. Frikken, M. Blanton, and M. Atallah, "Robust authentication using physically unclonable functions," in *Lecture Notes in Computer Science*.: Springer Berlin / Heidelberg, 2009, pp. 262-277.

[8] B. Gassend, "Silicon Physical Random Function," in *ACM Computer Communication Security*, 2002, pp. 148-160.

[9] J. Guajardo, S. Kumar, G. Schrijen, and P. Tuyls, "Physical unclonable functions, fpgas, and public key crypto for ip protection," in *International Symposium on Circuits and Systems (ISCAS)*, 2008, pp. 3186-3189.

[10] K. Kursawe, A. Sadeghi, D. Schellekens, B. Skoric, and P. Tuyls, "Reconfiguralbe physical unclonable functions - enabling technology for tamper-resistance storage," in *IEEE International Workshop on Hardware-Oriented Security and Trust (HOST)*, 2009, pp. 22-29.

[11] M. Majzoobi, F. Koushanfar, and M. Potkonjak, "Techniques for design and implementation of secure reconfigurable pufs," *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, vol. 2, no. 1, March 2009.

[12] G. Suh and S. Devadas, "Physical Unclonable Functions for Device Authentication and Secret Key Generation," in *ACM IEEE Design Automation Conference*, San Diego, CA, 2007, pp. 9-14.

[13] L. Kulseng, Z. Yu, Y. Wei, and Y. Guan, "Lightweight secure search protocols for low-cost Rfid systems," in *29th IEEE International Conference on Distributed Computing Systems*, 2009, pp. 40-48.

[14] Omar Al Ibrahim and Suku Nair, "OCCRA: overt-covert challenge-response authentication for device-centric primitives,"Technical Report, TR04CSE11, 2011.

[15] R. Bose and D. Ray-Chaudhuri, "On A Class of Error Correcting Binary Group Codes Information and Control," *Information and Control*, vol. 3, no. 1, pp. 68-79, March 1960.

[16] Y. Mandel and S. Devadas, "Secure and robust error correction for physical unclonable functions," *IEEE Design and Test*, pp. 48-65 , 2010.

[17] A. Hocquenghem, "Codes correcteurs d'erreurs," p. 147–156.

[18] M. Luby, M. Mitzenmacher, M. Shokrollahi, and D. Spielman, "Efficient erasure correcting codes," *IEEE Transaction on Information Theory*, vol. 47, no. 2, pp. 569-584, February 2001.

[19] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: how to generate strong keys from biometrics and other noisy data," in *Eurocrypt*, 2004, pp. 523-540.

[20] G. Hammouri, E. Ozturk, and B. Sunar, "A tamper-proof and lightweight authentication scheme," in *Pervasive and mobile computing*, 2008.

[21] U. Ruhrmair et al., "Modeling attacks on physical unclonable functions," in *Proceedings of the 17th ACM conference on computer and communications security*, 2010.

[22] K. Mahmud and S. Nair, "A Security Architecture for Nano-Sensor Networks," Southern Methodist University, Dallas, Technical Report Tech Report 06-CSE-02 , 2006.

[23] A. Jain, A. Ross, and P. Salil, "An introduction to biometric recognition," *IEEE Transactions on Circuits and Systems for Video Technology* , vol. 14, no. 1, pp. 4-20, January 2004.