Securing Communication Devices via Physical Unclonable Functions (PUFs)

Nicolas Sklavos

KNOSSOSnet Research Group
Technological Educational Institute of Western Greece, Greece
nsklavos@ieee.org

Abstract

In recent years, it has been more than obvious that electronic hardware devices are more than pervasive parts, in most aspects of everyday life. Although, the increased need for communications and transactions, makes both security and privacy manners a crucial factor, that has to be considered with high attention. New methodologies and approaches are developed, in order the need for high security levels, to be satisfied successfully.

Physical Unclonable Functions (PUFs) have attracted the interest of the research community the last years. PUFs basically support cryptographic primitives, in order to implement security schemes, such as key generation and storage, authentication, as well as identification.

This work carries out operation aspects of PUFs, as well as use cases, which are currently investigated by the researchers. In this paper, design approaches of PUFs are introduced, with detailed aspects of their behaviour. The security properties of the presented designs are given in detail, in order to demonstrate the security properties, introduced by the physical properties, in the most sufficient way. Comparisons of the alternative philosophies of the different designs are given.

1 Introduction

Intrinsic random physical features have been used lately at a great manner, in a great number of applications, as a useful approach for different aims and scopes. Physical(-cally) Unclonable Functions (PUFs) have been developed the last years, and new types are proposed, from time to time, regarding both their operation and construction.

A PUF can be described as a disordered physical system. Such a system can be challenged by the external environment. The PUF operation includes responses to those challenges. The responses depend on the nanoscale structural disorder in the PUFs.

Especially, PUFs have attracted lately the interest of the research community, since they are proven a very promising and trustworthy solution, especially in the areas of cryptographic hardware, since they can be used successfully for a great number of security applications.

This work proposes alternative directions and applied approaches of securing communication devices, via Physical Unclonable Functions (PUFs).

H. Reimer et al. (Eds.), *ISSE 2013 Securing Electronic Business Processes,*
DOI 10.1007/978-3-658-03371-2_22, © Springer Fachmedien Wiesbaden 2013

First, the alternative Physical Unclonable Functions (PUFs) categories are introduced by the construction point of view. The characteristics of each one of them are introduced.

The PUFs operations are presented, regarding alternative approaches of the design. The security application aspects are given in detail, regarding PUFs utilization and usage. Furthermore, the achieved security level of each one of PUFs categories is examined.

Last but not least, different aspects for the efficient implementation of PUFs are introduced. Conclusions are discussed and future directions are given.

2 PUFs from the Construction Point of View

In this section the alternative approaches for the construction of PUFs are given, as well as their certain properties.

Non Electronic PUFs: There is a number of PUFs designs which belong to this category, for which construction or/and operation is not related to electronic aspects. Although, the operation of this category is combined with other electronic parts or digital devices, mainly for the efficiency of storage. The non-electronic nature of these PUFs is related to the nature of random primitives generation. Other electronic parts are involved with the issues of processing and storage purposes.

PUFs based on Optical: Random optical procedures have been used as an alternative direction for the construction of unclonable functions. Such properties have also been used for the integration of functions with one way operation. Detailed test although, have to be performed in order the properties of the optical PUFs to be determined. It has to be mentioned that such a design approach may needs an initialization setup process regarding the laser, and possible other mechanical parts of the positioning system. Integrated designs with such or similar concepts could be found in technical literature.

PUFs and Compact Discs: Compact discs, as a laser technology product, can be used in order to support PUFs operation. In general, pits and lands of a common CD, are used as a random primitive, based on probabilistic options, which are occurred during the stage of manufacture.

PUFs Operation on Papers: Another completely different approach compared with the previous ones, is those PUFs, where their operation are based in the use of papers. They are based on scanning the random structure of a paper, which can be in a normal or in a modified version. Methodologies are introduced, in order to in order to make a connection of the document data with the paper version, based on digital signature techniques, or other methodologies like paper fingerprint.

Intrinsic PUFs: Although it is not efficient enough to define at a formal way the term of an intrinsic PUF, in order a function to belong to this category must meet at least the following specifications:

- The function has to be fully integrated in the hardware module, which may include any equipment which needed for the right operation of it.
- Any components should be fully available during the manufacturing process, of the hardware device.

PUFs and Memory Usage: This category includes PUFs which operation is based on memory elements. Alternative proposals have been published in the technical literature which contain different elements of memory storage like SRAM blocks, storage elements constructed of Flip-Flops , or data latches.

3 PUFs Operation

PUFs operation means different concepts, based on the alternative nature of the proposed designs. The concept of a Physically Obfuscated Key (POK) describes permanently the storage of a key in a physical way, instead of a digital one. Both of them have similarities, and it is efficient POKs to be constructed from PUFs. The combination of PUFs with other cryptographic primitives such as hash functions, results to controlled operation, with a number of advantages. The normal operation of a PUF can be extended to a number of additional operations which are well known are reconfiguration. As a result, the operation of the PUF is changed, concluding to a completely different design.

4 Security Applications Aspects and PUFs

The area of Physical Unclonable Functions (PUFs) has grown up at a great factor, during the last years, and it is expected for additional raise, for the forthcoming ones.

More analytically, Physical Unclonable Functions (PUFs) are widely used in a variety of aspects and especially for security applications and purposes. In this part of the paper, the alternative usages of PUFs regarding application aspects are presented in detail.

4.1 Cryptographic Engineering

One crucial aspect of security applications, where PUFs are used, has to do with the need of the key generation, for several applications of security. In this approach PUFs are used as external component of the Security System, which generates the key, which is extracted to the Security Core, as it is presented in the next Figure 1:

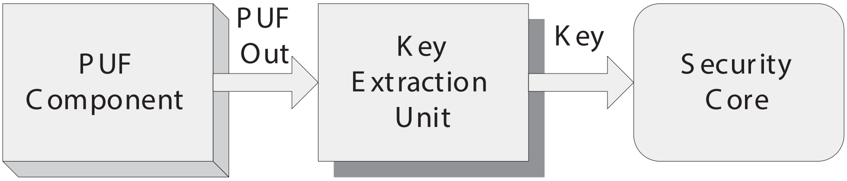

Fig. 1: PUFs Key Generation: Classical Method

Another approach for the key generation, is the integration of the PUF, inside the security primitive, with final result of this design, the hardware entangled security primitive, which is illustrated in the following Figure 2:

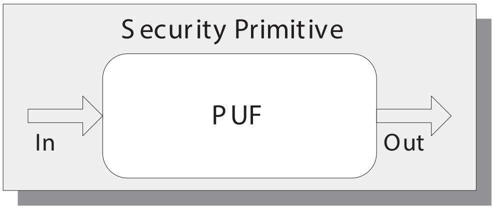

Fig. 2: PUF Integration in Security Primitive

Comparing the above presented approaches, someone can conclude to the fundamental differences between the alternative philosophies of both of them.

A cryptographic algorithm, and especially a block cipher, which operation is based on a PUF, can support cryptographic primitives based on the first approach, for which security issues may arise. The generated key is needed to be stored, as digital information, in general purpose components like memory devices, which can attract the interest of an attack attempt.

On the other hand hardware entangled security primitives are characterized as keyless approach, and fully support security, in the sense that it is not efficient for a possible attacker to gain access on information of the keys, since the data are not stored to any reachable component.

The above comparison makes clear that integrating the PUF as an internal part of the security primitive, meet the requirements of secure cryptographic systems, based on the unique properties of random physical features.

4.2 Authentication

For authentication purposes, PUFs can be alternatively used, as a module of the hardware component of each part of an established communication. Current methodologies are fundamentally based in password usages or other available approaches from software or hardware perspectives of view. Intrinsic PUFs although, can be applied alternatively.

An authentication handshake mechanism is based each time on a challenge and response scenario, which is applied for both client, as well as for server authentication as well. In most of the cases, the server takes the role of the verification part and applies a challenge-response pair, (CRP), to the system's database. This is related to a dedicated PUF component, which is being set up each time during the initialization process. The client responds to a random challenge, which has been caused by the server, which is the verifier party of the handshake protocol. The PUF is usually queried, and the response, which has been measured, is sent back to the verifier. The expected response is compared with the answered one, and in the case that these are the same, the authentication is achieved.

On the other side, server can be authenticated in a similar way of operation. In this case, there is the same initial condition and the PUF device possesses the client, while the server sends a CRP primitive, which has been chosen, in a random way. The client verifies it or not, by using it's PUF, expecting the response to be the same with the provided one.

The above described authentication protocols could be applied with additional security primitives. In such cases, a possible eavesdropper could take gain of the CRPs, which are sent during the communication, and personate a certain part of the communication. This is possible to be taken place, in the case that the challenge is used at least twice or more times, due to the fact that the attacker has the knowledge of the right response. In order to avoid such kind of attack, the CRP is proposed to be used only for one time. In different cases a new one CRP is used, for each case of authentication.

4.3 Identification of the Device

The purposes of authentication issues, analysed in detail in the previous section, make the device behaviour to the identification token. In this case, there is no need for security keys storage, which bypass the possibility an attacker to recover key data, which are stored in hardware components like RAMs. The PUFs devices, generates the cryptographic key on the fly, by making use of the intrinsic physical features of them.

Although typical components and devices could be used as PUFs, this arise security related issues, which could be under the investigation of external attackers of the system. These security issues, is a matter for further research and investigation, regarding the applied PUFs approaches.

4.4 Random Number Generation

Most of the security applications are based on random number generation, in order to ensure security in alternative means. These random generated data are basically used for the key generation or for other purposes such as salt vectors, and must not be predicted, under a possible attack. In most of the cases, pseudo random generators are used, since real random generators are not efficient to be applied.

The input of these pseudo random generators is coming from seeds with high entropy. Alternative approaches have been examined, which make use of general purpose devices, as input for high entropy data. Hardware components which are used for such purposes are GPUs, CPU Cache States, and SRAM Devices. PUFFIN project examines such approaches, for random number generation.

4.5 Safety Generation

For those applications that request keys generation, the last could be produced from any environment that includes a PUF primitive. The crucial factor in this case, has to do with those features of the environment, which could guarantee the secure generation of the keys. Following this approach, the key for the appropriate security application is generated, based on the certain hardware module. In this case, the hardware module is identified. Following that, the key maybe used in order to unlock encrypted data, which are stored on the device, or other applications that may be installed on it.

In order to achieve this, detailed schemes are applied, providing alternatives for devices identification. For example, the storage of a great number of keys and certifications, in dedicated hardware components and also in software parts could be applied. In this direction, such approaches usually need extra hardware modules, which increase the resources and the final cost of the hardware device. Although someone could suggest software solutions, which may be cheaper approaches, in order to reduce the cost. In general, software approaches, could not support strong trust directions, in order to ensure safety platforms.

Finally, today's solutions in most of the cases could attract side-channel attack attempts, or other methodologies, in order someone to gain access to the applied cryptographic data. The use of PUFs gives the benefit of embedding primitives to a hardware module, without storing cryptographic data to a separate module, or external hardware device.

4.6 Protecting IPs

It is obvious from the above section, that the physical properties of the underlying hardware are crucial aspects for the establishment of an identification scheme, which may be used for a device. As a result of this, additional applications and schemes can take place, in order to support other security or cryptographic schemes.

One very important aspect is to ensure security for IPs (Intellectual Property) protection, in software platforms. This can be achieved, by binding applications, based on hardware modules, in software platforms. With this method, illegal or not illegal software versions could be detected.

Furthermore, there are much cases, where software tools or applications have to be configured or controlled, by distance, or configurations options and primitives have to be delivered to other parties. As it has also been mentioned before, software instances, could be bind in hardware modules, and in this case software configuration information could be delivered safety, to the involved parties.

5 PUFs Security Level

Since there are several types and subtypes of PUFs available today, each one of them supports its own applications and security schemes. These different types could be divided to three main categories: a) Strong, b) Controlled and c) Weak, which have obvious differences in the sense of the security level.

5.1 Strong PUFs

With the term Strong PUFs, systems with physical environmental characteristics and with a great number of challenges and also with a complicated challenge and response behaviour are described.

Strong PUFs are widely used for a great number of applications such as key authentication processes, identification procedures and key establishment issues. This category can support cryptographic applications with high level of security, without complex arithmetic computational to be involved in the process. Electrical circuits can be used to construct Strong PUFs. Their main security features are described in the following paragraphs.

A Strong PUF is impossible to be cloned. This means in other words that it is not efficient to construct more than one system that achieves the same behaviour, for example in a challenge and response protocol. This feature, guarantees that each Strong PUF is unique from the begging of the manufacturing line process, and neither a designer nor a manufacturer could produce two PUFs with the same behaviour.

Another feature of Strong PUFs has to do with the challenge and response pairs (CRPs). It is not efficient for someone, even in the case he takes access to a PUF to measure all possible challenges and responses, in a given time period. Even in the case that someone takes full access to all challenges and responses of a PUF device, this is impossible for a long time period of days or weeks.

Finally, it is not efficient for someone to estimate a possible response for a given challenge, even in the case that some CPRs have been well known.

5.2 Controlled PUFs

Strong PUFs could be used as fundamental primitives in combination with control logic circuits. Additional logic could be used in order to control the challenges and the corresponding responses of the PUFs. In this way, challenges could be prevented from being sent to the PUF as well as responses could be read or not from other parts of the module. This strategy ensures defence from attacks or other possible deceptions.

It has to be clear that the possible outputs of a PUF, and especially a strong one, must not be read. In different case, it may be possible the behaviour of the PUF to be predicted, and the Control PUF logic to be broken.

5.3 Weak PUFs

The last category of the PUFs has to do with the weak behaviours of some of them. This means that there are cases where PUFs support a small number of challenges, and in the worst case just only one. For this reason, their response(s) has not to be given directly to the external environment.

Weak PUFs are basically used for cryptographic keys derivation, which is considered as a secret input of a security system. They can also be parts of key storage, in more efficient and secure way compared with other components, like types of ROM, which may be read. Different alternatives could be found today, in this category, like SRAM PUF, Coating and Butterfly PUF.

Strong PUFs can be used in order to construct Weak PUFs, with reducing the used number of challenges, of the available set.

Last but not least, other behaviours of the Weak PUFs have also to be considered such as error correction and stability also. A weak PUF produces and output response, which could be considered a secret key. This amount of information is been processed internal in the device. Error correction process has to been combined with high precision, since it is carried out internally on the chip. In order this to be achieved, it is possibly needed additional parts of data to be stored internally in the chip. The recipients of the produced responses, is possible to establish error correction procedures, which are allowed by Strong PUFs.

6 Implementation Aspects

Today, in technical literature there is a great number of alternative directions, regarding PUFs implementation. In order to prove their superiority and novel aspects of the proposed designs each time, a great number of advantages are figured out each time, in order the proposed design to be proven better to the compared one. Although such comparisons are based on the measurements provide for the proposed one. This concludes to a great number of measurements sets, which practically cannot be applied to all different categories of PUFs.

In order to have a fair and detailed comparison, for all different available designs, a number of parameters have to be applied, in order the security primitives as well as the practical usages of them to be examined and presented. In the following paragraphs, the most important parameters for such a comparison are considered.

- **Sample Size**: One of the basic characteristics is sample size which is considered as: the number of distinct devices, the number of challenges, as well as the number of the corresponding responses to them, as well as the measurements of each one response. Furthermore, statistical analysis has to be performed, regarding samples, in order to conclude to formal analysis each time of the examined PUF.
- **Histograms**: In most of the cases, inter- and intra- distance histograms are basically used for alternative designs comparisons, in order the uniqueness and the noise of a PUF to be determined. These histograms are mainly Gaussian, and their average is basically used. In addition, the standard deviation of each one of them is applied. For these reasons statistical approaches are considered to be used and due to the histograms' nature. Although, these are proven a good estimation for PUFs behaviour, in general, they are not sufficient enough in order to have a point of view for the factor of the entropy.
- **Entropy**: This parameter is presented as an estimation factor, in a number of PUFs responses. Alternative methods have been proposed in order to estimate and measure entropy. Although it has to be clear that these approaches are just an estimation for the factor of entropy, due to the available length of the responses, which not extend to great numbers.
- **Challenge and Response Pairs**: For the CPRs the number of the non-predictable ones is equal to a limited amount, close to a polynomial in the size of the PUF. Actually, each time and for a given size, the length, in term of bits, for the unpredictable response data is a key of great importance.
- **Implementation**: Implementation issues, regarding cost and efficiency are issues that are always related with the practical applications of the PUFs. It is obvious that such modules must have low cost. PUFs which are usually implemented in hardware modules, must also be examined in the implementation parameters by using hardware terms such as performance, speed, size, allocated resources and power consumption.
- **Influences**: PUFs usually response to non-wanted influences of the environment. It possible due to these influences to have failures in the PUFs operation, which are applied to a real application. Such influences should be examined and avoided in the term of possible, each time.
- **Tamper Evidence**: This factor is also considered as one of the important parameters of PUFs. Experimental measurements and validation are needed in order to conclude in a safe way, for a given PUF design, regarding tamper evidence. In this direction, experiments are proven of great importance, in order to result to the behaviour of the Challenge and Response Pairs, regarding PUF behaviour.

7 Conclusions & Outlook

This work proposes alternative directions of securing communication devices via Physical Unclonable Functions (PUFs). PUFs are coming on the market as part of the consumer devices. As a result of this growth, the areas the PUFs applications are introduced and presented in detail. Up today, in the field of PUFs, a great number of constructions have been set up. A comparative study of PUFs properties, for the different types of them, is essential in order to conclude to useful results, regarding the appropriate type, for each application and each time. From the implementation point of view, concrete characteristics have to be introduced and evaluated, in the applied design of PUFs. This will conclude to more advantages of the PUFs research areas. The first results of this state-of-the-art work in progress are more than promising, and the future directions are expected to achieve higher aims and scopes.

Acknowledgement

This work was partially supported by COST Action IC1204: Trustworthy Manufacturing and Utilization Of Secure Devices (TRUDEVICE).

References

[BePo09] Beckmann, N., Potkonjak, M.: Hardware-based public-key cryptography with public physically unclonable functions pp. 206-220, 2009.

[MaVe10] Maes Roel, Verbauwhede Ingrid, Physically Unclonable Functions: a Study on the State of the Art and Future Research Directions, In Towards Hardware-Intrinsic Security, D. Naccache, and A. Sadeghi (eds.), Springer, 36 pages, 2010.

[Puff13] Puffin Project: Physically Unclonable Functions Found in Standard PC Components, "INFSO-ICT-284833", Web: http://puffin.eu.org/, 2013.

[ScLe13] Schaller Andre, Leet van der Vincent, Pkysically Unclonable Functions found in Standard Components of Commercial Devices, First Workshop on Trustworthy Manufacturing and Utilization of Secure Devices, Co-located with IEEE European Test Symposium, Avignon, France, May 30-31, 2013.

[SkEf07] Sklavos Nicolas, Efstathiou Costas, SecurID Authenticator: On the Hardware Implementation Efficiency, proceedings of 14th IEEE International Conference on Electronics, Circuits and Systems (IEEE ICECS'07), Morocco, 2007.

[Skla10] Sklavos Nicolas, On the Hardware Implementation Cost of Crypto-Processors Architectures, Information Systems Security, The official journal of (ISC)2, A Taylor & Francis Group Publication, Vol. 19, Issue: 2, pp. 53-60, 2010.

[SkZh07] Sklavos Nicolas, Zhang Xinmiao: Wireless Security and Cryptography: Specifications and Implementations, CRC-Press, A Taylor and Francis Group, ISBN: 084938771X, 2007.

[TGG10] TCG, Mobile Trusted Module Specification, Version 1.0, Revision 7.02, Trusted Computing Group, Tech. Rep., 2010.