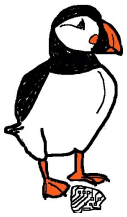

PUFFIN

Physically unclonable functions found in standard PC components

Project number: 284833
FP7-ICT-2011-C

D4.3

Periodic Report M1-18

Due date of deliverable: 31. July 2013
Actual submission date: 3. October 2013

WP contributing to the deliverable: WP4

Start date of project: 1. February 2012 Duration: 3 years

Coordinator:
Technische Universiteit Eindhoven
Email: coordinator@puffin.eu.org
www.puffin.eu.org

Revision 1.0

Project co-funded by the European Commission within the 7th Framework Programme		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission services)	
RE	Restricted to a group specified by the consortium (including the Commission services)	
CO	Confidential, only for members of the consortium (including the Commission services)	

Periodic Report M1-18

All partners

3. October 2013
Revision 1.0

The work described in this report has in part been supported by the Commission of the European Communities through the FP7 program under project number 284833.

Abstract

This document is the first periodic report, covering activity in PUFFIN for M1 – 18.

Keywords: WP4, first overall report

Contents

Chapter 1

Publishable summary

"Physically unclonable functions found in standard PC components" (PUFFIN) is a 3-year STREP in ICT-2011.9.2 "High-Tech Research Intensive SMEs in FET research". PUFFIN started on 1 February 2012. Its objective is to identify, qualify and use "Physically Unclonable Functions" (PUFs) in standard PC components.

PUFs are used to uniquely identify electronic components and to protect valuable objects against counterfeiting. They allow creating a root of trust in a hardware system through generating device-unique "fingerprints" and deriving secret keys from the underlying physical properties of the silicon. Today they are typically found in specially designed hardware components and result from the silicon properties of individual transistors. They exist in many forms, among which are the so-called SRAM PUFs.

The goal of the PUFFIN project is to study and show the existence of SRAM PUFs and other types of PUFs in standard PCs, laptops, mobile phones and consumer electronics. This was never attempted before. The mere existence of physical properties that depend on a component and are reproducible is only the first step to guarantee appropriate robustness, reliability and randomness properties for use as secret keys or trust anchors in mass-market applications. By uncovering the security properties of PUFs in standard components such as graphical processing units, central processing units and PCI connectors, this project will provide the first intrinsic and long-wanted basis for security in everyone's most common computing platforms: standard PCs and similar hardware. This new root of trust in turn adds security for mass-market applications, replacing or complementing the role of a trusted platform module and enabling security for applications such as broadcast applications, content protection for the gaming industry and secure day-to-day transactions for everyone. The results of the project will allow for the first time an a-priori open platform, the most difficult element to secure in an information-technology system today, to inherit security properties from its own identity and its intrinsic physical properties.

The activities in the PUFFIN project are organized in 3 work packages:

- WP1: Exploration
- WP2: Analysis and qualification
- WP3: Use cases

The following sections present the planned work for the work packages and the work actually carried out.

1.1 WP1: Exploration

The Exploration work package searches for new ways to physically identify PCs and other commodity hardware. It focuses on standard PCs and handheld devices as they actually exist today and in the foreseeable future. The goal is not to modify components to make them easy to identify; the goal is to find identifiers that are already intrinsic in PCs and mobile devices. WP1

- attempts to read out the uninitialized memory of various GPU types;
- attempts to read out the uninitialized memory of various CPU types;
- makes a preliminary assessment of the quality of the so obtained data;
- attempts to find identifying properties of mobile devices such as smart phones that are hard to clone;
- considers PUFs on FPGAs as a potential add-on to resort to if the data above is insufficient.

During the first period WP1 developed a range of new hardware and software tools for measuring undocumented behavior of several different types of chips. WP1 successfully measured power-on SRAM state from a range of chips, including smartphone CPUs, smaller embedded CPUs, and large GPUs. WP1 carried out a preliminary assessment indicating that most of this data was of reasonable quality, justifying forwarding the data to WP2 for further analysis.

1.2 WP2: Analysis and qualification

WP2 is the natural follow-up to WP1 in this project:

- WP2 starts with the development of statistical analysis tools and of mathematical and probabilistic models for the qualification of potential PUF data found and extracted in WP1.
- WP2 then proceeds to run the tools, perform such analysis and qualification on the different potential PUF instances identified in WP1, starting with GPU SRAM PUFs and CPU PUFs and continuing as new PUF instances are discovered throughout WP1.
- Finally, the analysis carried out leads to useful insights and recommendations for the choice of security parameters required for protocols and security architectures built upon these PUFs in WP3.

Work performed during period 1 in WP2 can be divided into two main parts: development of new methodologies for evaluating PUF behavior and analysis of PUF measurements (from WP1). The PUFFIN project has developed two new methodologies for evaluating PUF behavior. These methodologies, focusing on reliability modeling and extractable entropy calculation respectively, have been described in two scientific papers. These papers have both been accepted for publication at security workshops during 2013.

In regard to the analysis of PUF measurements from WP1 several samples of 5 different microprocessor types, 1 GPU type, 1 tablet, and 1 computer development platform (Pandaboard) have been evaluated. Results from this analysis can be used during the project as an indication of which PUFs are suitable for deployment in the use cases of WP3.

1.3 WP3: Use cases

WP3 is devoted to applications of and use cases for the PUFs investigated in WP1 and WP2. In particular, WP3 considers different ways for increasing the trust in computer systems through the use of PUFs found therein:

- WP3 develops hardware-entangled cryptographic primitives that draw their security directly on physical assumptions of the underlying PUF; an integral part of this task is to develop error correction schemes specifically tailored towards the error characteristics of the PUFs identified in WP1.
- WP3 investigates to which extent the PUFs developed in WP1 can be used to implement low-cost alternatives to Trusted Platform Modules for key storage and attestation.
- Finally, WP3 considers the use of PUFs in software security mechanisms that either guarantee the integrity of software or allow to securely bind software to a particular hardware platform; both problems naturally occur in solutions for anti-counterfeiting.

The security of the developed techniques is analyzed and eventually a proof-of-concept implementation will be built.

During the first period of the PUFFIN project WP3 identified and collected possible use cases for intrinsic PUFs. A finer selection extracted potential use cases, which form the basis for future research in WP3.

A novel Helper Data scheme was proposed, covering task 1 (Hardware-based trust establishment). The approach is based on a modified Code Offset Method and takes into account the non-uniformly distributed PUF measurements in intrinsic PUFs. PUF-based software security (Task 2) was covered by two contributions. First, a secure boot architecture was implemented on a system-on-a-chip platform exploiting a hardware fingerprint extracted from the on-board SRAM. Intertwining the platform's boot loader with the extracted fingerprint makes lightweight hardware-software binding possible. The second contribution is the implementation of a secure pseudo-random number generator (PRNG) in wide-spread microchips by exploiting the noise inherent to PUF measurements. Security evaluations will be considered in the second half of the project.

1.4 Public web page

Up to date project information, the research deliverables, and details on the upcoming PUFFIN workshop can be found on the project web page at http://puffin.eu.org/.

Chapter 2

Project objectives, work progress and achievements

More and more information is stored on PCs and other consumer devices, and distributed with the intention to be accessible by a limited set of people. This information varies from confidential information on company intranets through personal information on social networks to sensitive information such as health-care or tax-related data in government databases. In addition, the increased availability of broadband connections for end users has triggered an invasion of commercial content such as ebooks, music, software, online gaming, and video on demand. Furthermore, embedded systems are increasingly used to perform security-critical tasks. For example, smartphones are used as payment systems, serve as (trusted) personal mobile access points, and consequently store a lot of privacy sensitive information.

Commercial incentives or legal requirements often demand that protection mechanisms are in place to prevent files from being accessed by unauthorized people or copied and further distributed. Traditional techniques include simple access control or encryption. The problem with these solutions is that they depend on the secrecy of a single piece of information such as a decryption key. Stressing a real-world analogy, anyone who possesses a door key, or a photograph of the key from a proper angle, can easily build a clone of that key, and later use the clone to open the door. Similarly, anyone who sees a cryptographic key can immediately copy it, and later use a copy of the key to decrypt data protected by the key and to forge data encrypted by the key. Security requires keeping door keys and cryptographic keys hidden from attackers. These keys have to be stored somewhere on the PC or communication device; as soon as this location is discovered and the key can be extracted, the protection mechanism is broken.

In an arms race with hackers and pirates, refuge has been sought in hardware-based solutions for key management. *It is now common wisdom in the IT security domain that a hardware-based foundation is necessary for secure and trustworthy systems.* For instance, smart cards are used to store cryptographic keys in set-top boxes (STBs) for television, and Trusted Platform Modules (TPMs) were introduced on personal computers (especially laptops) in order to provide a root of trust; TPMs have also been specified for mobile phones. Unfortunately, the introduction of hardware components induces extra cost and complexity. At this moment there is still a large number of PCs that do not have a TPM and it is likely that this will not change in the near future. It is therefore pressing to search for other roots of trust. *The goal of PUFFIN is to show that unique physical features that are intrinsically*

available in current PCs and other consumer devices can be used as an alternative foundation of trust.

To increase the security of hardware components and embedded systems, some research projects have explored the use of unique physical hardware properties through Physically Unclonable Functions (PUFs). A PUF is a physical characteristic that acts as a noisy function: it receives an input (called a challenge) and responds with a noisy physical output (called a response); the PUF produces slightly different outputs for the same input. In order to obtain a noiseless output, error correction (in the form of fuzzy extractors) can be utilized.

PUFs provide intrinsic security: A PUF is physically unclonable—even after thoroughly inspecting the PUF, an attacker cannot build another device that responds to inputs in the same way, because doing so is an infeasible (or too expensive) engineering challenge. PUFs therefore do not need to be kept hidden from attackers. A PUF can be temporarily loaned to someone, and then returned to the owner, who verifies that the PUF responds to various inputs in the same way as before; the owner is then sure that the PUF has not been replaced with another one, is sure that the borrower does not have a copy of the PUF, and is sure that the borrower no longer can produce PUF responses to yet unknown challenges. The latter is due to the PUF security property of unpredictability: without physical access to the PUF, an attacker cannot predict the PUF response to a previously unused challenge.

Over the past ten years PUFs have been built from a wide variety of technologies, including magnetic stripes, coatings added to chips, and glass bubbles that refract light. For example, dedicated hardware read-out circuits using intrinsic SRAM variations can generate reliable secret keys. SRAM in existing field-programmable gate arrays (FPGAs) can be used as PUFs, allowing copy protection for designs loaded into FPGAs. In each case the PUF output is dictated by uncontrollable random variations in the PUF manufacturing process. At the forefront of research and development in this area is a European SME: Intrinsic-ID; the same SME is, not by coincidence, the initiator and leader of this proposal.

Common applications of PUFs are in the domain of key storage, authentication and anti-counterfeiting. PUFs can provide security for key storage due to their unpredictability: cryptographic keys do not need to be stored directly in memory; rather, keys can be derived from PUF responses 'on the fly' whenever a cryptographic operation needs to be performed. This reduces the time period during which keys are stored in memory, thereby minimizing exposure to attackers. PUF unclonability can be used to authenticate devices in a secure fashion: only an authentic device can answer a random unused challenge with the correct response. The latter is a key ingredient to prevent counterfeiting of hardware components.

So far, in all application scenarios considered for PUFs, a dedicated PUF has to be built and added to the system. This results in a complex system design, longer time to market and increased cost of a product. However, since PUFs can, for instance, be built from memory modules, it may be possible to use the unique physical properties of components that *are already available* in PCs, smartphones, etc. as PUFs. At the start of the PUFFIN project this direction was completely unexplored, and it was open where the research would lead.

The PUFFIN project aims at a breakthrough in PUF usability, demonstrating that standard mobile devices, PCs, laptops and consumer electronics devices are already full of components that can be combined into a high-security PUF. Devices do not have to be modified; system integrators do not have to purchase add-on PUFs, which is particularly important for small mobile devices having high cost constraints; users do not have to carry around key rings full of dongles. This breakthrough will enable applications ranging from anti-counterfeiting and access control to content protection and high-security encryption. Our long-term vision is that

users on the Internet will no longer have to worry about attackers stealing their credentials and privileges through software bugs or Trojans; security will be guaranteed by intrinsic properties of the devices.

Even for PCs equipped with a TPM it may be advantageous to use existing PC components as a PUF, to add an extra layer of security. For example it can help to make sure that the TPM chip is indeed part of the specific PC, and not moved to another PC. Further, PUFs can enhance the security of TPM-based secure storage against physical attacks, which are not yet considered when designing TPM chips. In the future, PUFs may become the new root of trust in PCs. Set-top boxes, game consoles or other embedded systems need similar protection as PCs, as witnessed by recent attacks against game consoles and smart phones. At the same time, their architecture is—due to cost issues—increasingly composed of standard components that share many similarities with PC parts. For this reason, PUFFIN will focus (besides PCs) on mobile computation platforms, such as smart phones.

> *It is the goal of PUFFIN to show that security can be obtained from properties that are inherent to the components that constitute a modern computer or mobile computation platform.*

In order to realize this vision of intrinsic security, PUFFIN strives to achieve three goals during the duration of the project:

- First, PUFFIN investigates the potential use of standard PC components and embedded devices as PUFs. The main research challenge addressed is to find ways to adequately access the PUFs. For example, one of the big problems with memory PUFs is to get access to the raw memory, before it is overwritten or initialized. A second challenge is to provide methods to securely integrate the extraction of PUF data and to make the data available in a live system (running an operating system and applications on top), to the actual application that needs it.

- Second, PUFFIN analyzes the identified PUFs in terms of reliability, uniqueness, and unclonability. PUFFIN will not include expensive chip-fabrication experiments but will investigate whether programmable devices could emulate the behavior of the PUF. To this end, novel theoretical models need to be developed to measure PUF robustness and randomness properties. These models will be applied to the identified PUFs in order to identify the most reliable and secure ones. The overall goal is to give recommendations to choose the most suitable PUF for a given application scenario and determine its operational parameters.

- Third, PUFFIN works towards the goal of using PUFs to provide intrinsic security for current platforms. In particular, PUFFIN attempts to investigate to which extent PUFs can be used as 'root of trust' for applications running on a PC platform or a smartphone. As a common basis for the applications, PUFFIN investigates how intrinsic PUFs can be used as core components in the construction of cryptographic primitives and protocols. The security of the developed protocols will rest on physical assumptions on the PUF rather than on computational assumptions, commonly made in cryptography. In a second phase, PUFFIN considers applications of the developed PUF-based primitives and protocols. For one, we consider applications from the domain of data protection, where data should be bound to a particular platform in an nontransferable

manner. Further, we consider ways to identify counterfeit hardware parts and provide methods to prevent software infringement by binding software to the platform it runs on. PUFFIN also investigates to which extent PUFs can replace or enhance hardware security components such as TPMs or MTMs (Mobile Trusted Modules).

2.1 Project objectives for the period

The following subsections list the particular project objectives for the first reporting period.

2.1.1 WP1: Exploration

WP1 has three tasks that overlap this period. Task 1.1, "Identification of potential PUFs in PCs" (M0 to M24), endeavors to find identifying digital fingerprints in as many of a PC's components as possible. Specific targets identified in the Description of Work included CPU caches and SRAM on GPUs. Task 1.2, "Identification of potential PUFs in mobile devices" (M6 to M36), asks analogous questions regarding components of mobile devices such as smartphones and tablets. Task 1.4, "PUFs for the future" (M12 to M36), considers non-SRAM technologies such as phase-change memory and SSDs.

WP1 had three milestones for this period: MS11 (measurements from GPUs), MS12 (measurements from CPUs), and MS13 (measurements from mobile devices).

2.1.2 WP2: Analysis and qualification

There are two main objectives for WP2 in the PUFFIN project: - Creation of mathematical models and development of statistical analysis tools, which can be used to evaluate PUF measurements focusing on reliability and uniqueness. In this context reliability is the stability with which a PUF response can be reproduced under varying ambient conditions, over time, etc. Uniqueness indicates that the probability that two PUFs have closely resembling properties is exponentially small. In order for PUFs to be usable for security implementations they should both be reliable and unique.

- Evaluation of the PUF measurements from WP1 based on the two main properties of PUFs mentioned above, reliability and uniqueness. Based on this evaluation can be assessed which PUFs from WP1 are suitable for deployment in use case implementations of WP3.

2.1.3 WP3: Use cases

The main goals of Work Package 3 (WP3) is the collection of use cases for intrinsic PUFs, the development of protocols and implementations based on the identified use cases and the security evaluation of the proposed protocols and implementations. Thus, the objectives of Work Package 3 are three-fold.

Task 1 — Hardware-based trust establishment — aims at the development of protocols and architectures, which exploit the hardware properties of the underlying PUF instances to be secure. In particular, this task considers cryptographic building blocks, such as random number generators, block and stream ciphers, authentication and key exchange protocols, which are based on intrinsic PUFs rather then on complexity-theoretic hardness assumptions. Additionally, in task 1 new Helper Data schemes and Fuzzy Extractor approaches are investigated, which are specialized on the peculiarities of intrinsic PUFs. Finally, task 1 is

devoted to explore the possibilities to replace traditional Trusted Platform Modules (TPMs) by intrinsic PUFs, such that PUFs instances can be used for key storage or platform authentication.

Task 2 — PUF-based software security — aims at the protection of software against modification or unauthorized duplication. Thus, in task 2 approached to intertwine PUF challenges and responses with a given binary are explored. The main challenge of task 2 is to prevent attackers to passively listen to PUF challenges and responses transmitted between the PUF instance and the software instance on a virtualization layer. The attacker would be able to emulate the PUF instance and break and thus break its security.

Task 3 — Security evaluation and implementation — aims at evaluation the security properties of the results from task 2. To do so, formal evaluation of the proposed approaches is performed. In the case of cryptographic primitives these tests will be conducted by means of mathematical proofs. As all three objectives span the whole PUFFIN project, the first period is devoted to a subset of the listed tasks. In particular, WP3 focused on task 1 and 2 during the first period of the project. Task 1 is covered by a novel Helper Data scheme based on a modified Code Offset method. Task 2 is covered by a) a lightweight secure boot architecture for commodity hardware using intrinsic PUFs and b) a secure pseudo-random number generator for microchips, which exploits the noise inherent to PUF measurements.

In the second period of the PUFFIN project more sophisticated approaches will be explored, using the results of the first period as a basis. Furthermore, the most challenging objective of developing a solution to the passive listening attack will be approached.

2.2 Work progress and achievements during the period

In general the PUFFIN progress exceeded the initial plans. WP1 made progress on identifying PUFs on a significantly larger number of platforms useful for mobile devices, WP2 not only analyzed all WP1 data in a timely manner but developed new methodologies for evaluating PUF behavior, and WP3 made solid progress towards implementing two use cases.

2.2.1 WP1: Exploration

WP1 quickly identified microcontrollers as an attractive initial target for collecting digital fingerprints. Microcontrollers are low-cost low-power CPUs used in a wide range of embedded applications; they are of interest as stepping-stones towards more complex CPUs, and they are also of interest as security devices per se. WP1 successfully accessed uninitialized SRAM data from the following microcontroller CPUs, reaching milestone MS12: Microchip PIC16F1825, STMicro STM32F100R8, Texas Instruments MSP430F5308, and Atmel ATMega328p.

WP1 also began two directions of investigating larger processors. First, as a prototype of a mobile device, WP1 selected the PandaBoard (ES) development board; this board contains the same TI OMAP4460 (dual-core Cortex-A9) CPU used in many smartphones. WP1 successfully accessed uninitialized SRAM data from L3 OCM RAM on the PandaBoard, reaching milestone MS13. Second, building upon the preliminary GPU results reported in the Description of Work, WP1 developed a new GPU SRAM readout tool for NVIDIA GPUs, and successfully accessed uninitialized SRAM data from four GPUs on two GTX 295 graphics cards, reaching milestone MS11.

The multi-pronged success in collecting SRAM data, from more devices than initially anticipated and with more phenomena worthy of study, warranted delaying the start of Task

1.4 in favor of a continuing focus on SRAM. The same success means that PUFFIN already has several test and demonstration platforms, without having to resort to add-on FPGAs.

Statement on the use of resources

Here is an overview stating the planned and actual person months spent on WP1. For further explanations of the deviations see section 2.2.4.

TUE		IID		KUL		TUD		total	
planned	actual	planned	actual	planned	actual	planned	actual	planned	actual
5 PM	1.5 PM	5 PM	3.7 PM	8 PM	8 PM	1.5 PM	1.5 PM	19.5 PM	14.2 PM

2.2.2 WP2: Analysis and qualification

The PUFFIN project has developed two new methodologies for evaluating PUF behavior in WP2, which have both been accepted for publication at security workshops during 2013.

The first new method focuses on analyzing PUF reliability. The work introduces a new reliability model taking an observed heterogeneous nature of PUF cells into account. A substantial experimental validation has demonstrated that the new predicted distributions from this model describe the empirically observed data statistics almost perfectly, even considering sensitivity to operational temperature. This will allow to study PUF failure behavior in full detail, including the average and the worst case probabilities.

Besides for reliability, also a new methodology for evaluating the uniqueness of PUFs has been developed within PUFFIN. The aim of this work was to develop and implement a new methodology for accurately estimating the entropy of PUFs. This novel method estimates the extractable entropy by calculating the mutual information between enrollment and reconstruction measurements.

Furthermore, all (mature) measurement sets from WP1 have been analyzed in WP2. These sets contain several samples of 5 different microprocessor types, 1 GPU type, 1 tablet, and 1 computer development platform (PandaBoard). In deliverable D2.1 an overview can be found of the performance of these devices in several different PUF tests. The most important conclusion that can be drawn from these results is that PUF behavior can be found in the SRAMs of many different commercially available platforms. Most of the SRAMs that have been measured show promising results and therefore are suitable for use in PUF implementations; however, the amount of pre-processing required on the data will vary between the platforms.

Statement on the use of resources

Here is an overview stating the planned and actual person months spent on WP2. For further explanations of the deviations see section 2.2.4.

TUE		IID		KUL		TUD		total	
planned	actual	planned	actual	planned	actual	planned	actual	planned	actual
1.5 PM	1.6 PM	6 PM	4 PM	4 PM	4 PM	1.5 PM	1.5 PM	13 PM	11.1 PM

2.2.3 WP3: Use cases

During the first period we were able to achieve the following points:

Task 1: We were able to create a new Helper Data scheme, which takes into account the peculiarities of intrinsic PUF measurements. This way, a new variant of so-called Zero Secrecy Leakage of the Helper Data was achieved, even though – especially in the case of intrinsic PUFs – the PUF measurements are not uniformly distributed.

Task 2: Based on the research of WP1, we identified and extracted a PUF instance on a modern system-on-a-chip platform, which features wide-spread ARM microprocessors found in today's mobile devices, such as smart phones. In WP3 we utilized this intrinsic PUF to extract and reconstruct a cryptographic key unique to single instances of this platform. By modifying the platform's bootloader, we were able to bind the bootloader to the extracted key and thus to the hardware itself, enabling a lightweight secure boot approach, which does not need additional hardware modifications (such as TPM chips).

During the analysis of different microchips towards their PUF characteristics, we realized that for some microchips it is possible to exploit the noise inherent to every PUF measurement to provide a secure pseudo-random number generator. Random numbers are essential for almost all cryptographic primitives (secret keys in block ciphers, secret keys in stream ciphers, secret keys in public-key signature systems, and so on), and the quality of this randomness is critical for security. However, in the case of lightweight embedded devices, sources of secure random numbers are rare. We succeeded in exploiting the entropy from the noise of the PUF measurements to extract a high-quality seed for a pseudo-random number generator on such devices. Thus, we were able to use the inherent noise in PUF measurements to strengthen an important cryptographic building block.

Statement on the use of resources

Here is an overview stating the planned and actual person months spent on WP3. For further explanations of the deviations see section 2.2.4.

TUE		IID		KUL		TUD		total	
planned	actual	planned	actual	planned	actual	planned	actual	planned	actual
1 PM	0.5 PM	6 PM	4 PM	5 PM	5 PM	15 PM	13 PM	27 PM	22.5 PM

2.2.4 Statements on the use of the resources

While the planned research advanced according to plan, there are some differences between the planned hours and the hours actually worked on the PUFFIN project. The differences in paid PMs are reflected in the financial statements. The following paragraphs give justifications for the deviations and explains how the deviations from the plan were handled.

TUE

TUE spent significantly less paid time in the first reporting period because Lange was on sabbatical and the postdoc (Ruben Niederhagen) was not available in M13 but only in M20. Bernstein is leading WP1 as a volunteer without financial support by PUFFIN.

IID

IID has spent some PMs less than planned during period 1. This has to do with delay in finishing other European projects during the first 6 months of the PUFFIN project. Due to this delay, IID did not have as many resources available for PUFFIN as originally planned. This is no longer an issue and IID will make sure to make up for the lost time during period 2 of the PUFFIN project.

KUL

There were no significant deviations between planned and actual PMs.

TUD

Since the main effort of TUD is in WP3, which builds on the results of WP1 and WP2, we decided to shift 2 PMs towards the second half of the project to exploit the results of WP1.

Chapter 3

Project management during period I

As this is a small-scale project with four partners that have a long-standing collaboration (in part while some of the key researchers had other affiliations), the project uses a lean governing culture. The Project Management Board (PMB) consists of one senior representative per partner complemented with all the WP leaders. At the moment these are the following:

- Project Manager: Tanja Lange, TUE.
- Scientific Manager: Pim Tuyls, IID.
- WP1 leader: Daniel J. Bernstein, TUE.
- WP2 leader: Vincent van der Leest, IID.
- WP3 leader: Stefan Katzenbeisser, TUD.
- WP1 co-leader: Frederic Vercauteren, KUL.

3.1 Scientific and Industrial Advisory Board (SIAB)

The SIAB is chaired by the Scientific Manager and consists of the Project Manager as well as a senior researcher from each of the four partners complemented by several senior researchers from European industry and academia. The following experts form the SIAB of PUFFIN:

- David Naccache, École normale supérieure, France.
- Christof Paar, Ruhr-Universität Bochum, Germany.
- Jean-Jacques Quisquater, Université catholique de Louvain, Belgium.
- Claire Vishik, Intel, UK.

In the first reporting period we had one meeting with the SIAB board on July 8, 2013 in London at the Intel offices. The minutes of the meeting are available on the review page.

3.2 Communication and coordination activities

Coordination is mainly done by email via the general mailing list `all@puffin.eu.org` which reaches everybody involved with PUFFIN. During the first reporting period the PMB had frequent phone conferences and in-person meetings to coordinate the ongoing research.

3.2.1 List of project meetings, dates and venues

For all the meetings mentioned below all partners were represented.

2012-04-13 Kick off meeting at TUE. Initial distribution of work and responsibilities. First meeting of persons working on project. Discussion of list of devices to study. Presentation of webpage, templates, and svn repository.

2012-05-03 Phone conference. List of available devices compiled. Discussion of naming and file conventions. Plan to make overview of protocols based on PUFs.

2012-05-14 Phone conference. Presentation of naming and file conventions. Will have separate repository for PUF data.

2012-06-11 Phone conference. Discussion about available hardware and some purchases. Collection of list of available scripts for PUF analysis. Overview of PUF-based protocols almost done.

2012-07-18 Phone conference. Problems in accessing SRAM on mobile phones. Plan to purchase development boards with open bootloaders to access ARM chips. More references for PUF-based protocols. TUD has hired a PhD student.

2012-08-20 Phone conference. Planning of next in-person meeting. Randomness generation is big problem in practice. Devices which are not stable enough to give a good PUF can be helpful for generating randomness; this is particularly interesting for small devices (see USENIX and Crypto papers). Discussion about interesting use cases.

2012-09-13 Meeting at KUL. Discussion of press release. Members of SIAB board. Overview of human resources for project. Overview of platforms considered so far. Entropy analysis of Physical Unclonable Functions. Discussion on the suitability for PUFFIN applications and use cases. Progress is steady, so fewer general meetings needed; partners organize some research visits.

2013-02-19 Meeting at TUD. Discussion of dates and location of PUFFIN workshop and SIAB meeting. Progress on analyzing several microprocessors, strong patterns for PICs, demonstration of hardware. Discussion of future work for WP1: postpone work on SD cards in favor of more diversity in small processors. Analysis and qualification of data obtained so far; new tool for entropy estimates. Use cases to consider further: secure boot, authentication in games, remote attestation, and random number generation.

2013-05-06 Meeting at IID. Progress on deliverables, planning of SIAB meeting. More plans to investigate small devices, chance to get access to more GPUs. New results on

entropy analysis. Work on bootloader for PandaBoard. First papers got accepted, some more under preparation. Several new open questions. Part of the afternoon was used as a research retreat to work together.

2013-07-03 Phone conference. Preparation of SIAB meeting. Status of deliverables, coordination of presentations.

3.3 Impact of possible deviations from the planned milestones and deliverables

As shown in Tables 3.6.1 and 3.6.2 the PUFFIN project is progressing smoothly, following the plan in all essential respects.

Most partners had a slightly slower than planned start in hiring additional personnel which led to some initial delays in achieving the milestones but have now caught up.

TUE had some budget problems in obtaining access to more GPUs and hired their postdoc 7 months later than planned, but has overcome these problems by now. The coordinator (herself at TUE) ensured that work in WP1 was not impacted by these deviations.

3.4 Development of the project website

The project website has been established in M1. WP leaders can update the sections on their WP. Once the deliverables have been accepted by the reviewers they will be available for download on the website.

Scientific papers resulting from the PUFFIN project will be made available if copyright allows this.

3.5 Statement on the use of resources

Here is an overview stating the planned and actual person months spent on management. Deviations from planned efforts are minimal.

TUE		IID		KUL		TUD		total	
planned	actual	planned	actual	planned	actual	planned	actual	planned	actual
0.7 PM	0.7 PM	1 PM	0.75 PM	0 PM	0 PM	0 PM	0 PM	1.7 PM	1.45 PM

3.6 Deliverables and milestones tables

Table 3.6.1: Deliverables

Deliverable number	Deliverable name	Version	WP number	Lead beneficiary	Nature	Dissemination level	Delivery date from Annex I	Actual/forecast delivery date	Status	Comments
D1.1	Scientific contribution of WP1, part 1	1.0	1	1	R	PU	M18	M20	✓	
D1.2	Scientific contribution of WP1, part 2		1	1	R	PU	M36	M36		
D2.1	Scientific contribution of WP2, part 1	1.0	2	2	R	PU	M18	M20	✓	
D2.2	Scientific contribution of WP2, part 2		2	2	R	PU	M36	M36		
D3.1	Scientific contribution of WP3, part 1	1.0	3	4	R	PU	M18	M20	✓	
D3.2	Scientific contribution of WP3, part 2		3	4	R	PU	M36	M36		
D4.1	Project website and internal ICT communication infrastructure	*	4	1	O	PU	M3	M1	✓	
D4.2	Project dissemination plan	*	4	1	R	PU	M6	M8	✓	
D4.3	First report according to the Grant Agreement	1.0	4	1	R	PU	M18	M21	✓	
D4.4	Second report according to the Grant Agreement		4	1	R	PU	M36	M36		
D4.5	Final report for the use and dissemination of foreground		4	1	R	PU	M30	M30		
D4.6	Report on IPR		4	1	R	PU	M36	M36		

Standard codes for the nature of the deliverable: R = Report O = Other
Standard codes for dissemination level: PU = Public

*: Web site and D4.2 updated continuously. Current version of D4.2 is from M20.

Table 3.6.2: Milestones

Milestone number	Milestone name	WP number	Lead beneficiary	Delivery date from Annex I	Achieved	Actual/forecast delivery date	Comments
MS11	Measurements from GPUs	1	1	M4	✓	M1	*
MS12	Measurements from CPUs	1	1	M10	✓	M10	
MS13	Measurements from mobile devices	1	3	M10	✓	M15	
MS14	Integration of FPGA SRAM PUFs	1	1	M27		M27	
MS21	Mathematical models are created and statistical analysis tool is developed	2	2	M10	✓	M8	
MS22	Potential GPU PUFs discovered in WP1 are either qualified or disqualified	2	2	M10	✓	M1	*
MS23	Potential CPU PUFs and mobile-device PUFs discovered in WP1 are either qualified or disqualified	2	2	M16	✓	M13	
MS31	Preliminary analysis of high-level requirements for PUFs	3	4	M10	✓	M5	
MS32	Initial evaluation of PC PUF suitability	3	4	M18	✓	M16	
MS33	Decision on protocols and security architectures to be implemented as proof of concept	3	4	M20		M20	
MS41	Elements of ICT infrastructure usable	4	1	M1	✓	M1	
MS42	Kick-off meeting	4	1	M2	✓	M3	
MS43	Website up and running	4	1	M3	✓	M1	
MS44	Initial project dissemination plan completed	4	1	M4	✓	M6	
MS45	SIAB established	4	1	M6	✓	M8	
MS46	First meeting of the SIAB	4	1	M18	✓	M18	
MS47	Second meeting of the SIAB	4	1	M30		M30	
MS48	Workshop	4	1	M20		M22	

Milestones are managed internally to keep the project on track.

* Multiple rounds of GPU measurements and analyses, most recently in M19 and M20.