UMR IRISA

Activity Report 2019

Team TAMIS

Threat Analysis and Mitigation for Information Security

Joint team with Inria Rennes – Bretagne Atlantique

D4 – Language and Software Engineering

CentraleSupélec · cnrs · rennes · IMT Atlantique Bretagne-Pays de la Loire École Mines-Télécom · Inria · INSA RENNES · Université Bretagne Sud ubs · UNIVERSITÉ DE RENNES 1

Table of contents

Project-Team TAMIS

Creation of the Team: 2016 January 01, updated into Project-Team: 2018 January 01

Keywords:

Computer Science and Digital Science:

A4. - Security and privacy
A4.1. - Threat analysis
A4.3. - Cryptography
A4.4. - Security of equipment and software
A4.5. - Formal methods for security

Other Research Topics and Application Domains:

B6.6. - Embedded systems

1. Team, Visitors, External Collaborators

Research Scientists

Olivier Zendra [Team leader, Inria, Researcher]
Annelie Heuser [CNRS, Researcher]
Kim Larsen [Inria, International Chair, Advanced Research Position]

Post-Doctoral Fellows

Eduard Baranov [Inria, Post-Doctoral Fellow, until Feb 2019]
Yoann Marquer [Inria, Post-Doctoral Fellow]
Matthieu Mastio [CNRS, Post-Doctoral Fellow, from Mar 2019]
Tania Richmond [Inria, Post-Doctoral Fellow]
Stefano Sebastio [Inria, Post-Doctoral Fellow]

PhD Students

Delphine Beaulaton [UBS Vannes, PhD Student, until Jul 2019]
Cassius de Oliveira Puodzius [Inria, PhD Student, from Feb 2019]
Christophe Genevey-Metat [Univ de Rennes I, PhD Student]
Alexandre Gonzalvez [IMT Atlantique, PhD Student, until Nov 2019]
Nisrine Jafri [Inria, PhD Student, until Feb 2019]
Tristan Ninet [Thales, PhD Student, until Nov 2019, granted by CIFRE]
Lamine Noureddine [Inria, PhD Student]
Duy Phuc Pham [CNRS, PhD Student, from May 2019]
Alexander Zhdanov [Inria, PhD Student, until Jun 2019]

Technical staff

Najah Ben Said [Inria, Engineer, until Jan 2019]
Ioana Domnina Cristescu [Inria, Engineer, until Jul 2019]
Cassius de Oliveira Puodzius [Inria, Engineer, until Jan 2019]
Olivier Decourbe [Inria, Engineer]
Nicolas Kiss [Inria, Engineer, from Apr 2019]
Bruno Lebon [Inria, Engineer]
Céline Minh [Inria, Engineer]

Interns and Apprentices

Yulliwas Ameur [Inria, from Mar 2019 until Sep 2019]
Agathe Cheriere [Inria, from May 2019 until Aug 2019]

Pierrick Philippe [Inria, from May 2019 until Aug 2019]
Julien Royon Chalendard [Inria, from May 2019 until Aug 2019]

Administrative Assistant
Cécile Bouton [Inria, Administrative Assistant]

Visiting Scientist
Nisrine Jafri [Inria, from Mar 2019 until Apr 2019]

2. Overall Objectives

2.1. Context

Security devices are subject to drastic security requirements and certification processes. They must be protected against potentially complex exploits that result from the combination of software and hardware attacks. As a result, a major effort is needed to develop new research techniques and approaches to characterize security issues, as well as to discover multi-layered security vulnerabilities in complex systems.

In recent years, we have witnessed two main lines of research to achieve this objective.

The first approach, often called *offensive security*, relies on engineering techniques and consists in attacking the system with our knowledge on its design and our past expertise. This is a creative approach that supports (1) checking whether a system is subject to existing vulnerabilities, i.e. classes of vulnerabilities that we already discovered on other systems, and (2) discovering new types of vulnerabilities that were not foreseen and that may depend on new technologies and/or programming paradigms. Unfortunately, this approach is limited to systems whose complexity remains manageable at the human level. This means that exploits which combine several vulnerabilities may be hard to identify. The second and more formal approach builds on formal models (also known as *formal methods*) to automatically detect vulnerabilities, or prove their absence. This is applicable to systems whose complexity is beyond human reasoning, but can only detect existing classes of vulnerabilities, i.e., those that have been previously characterized by offensive security.

2.2. Approach and motivation

The claim made by TAMIS is that *assessing security requires combining both engineering and formal techniques*.

As an example, security exploits may require combining classes of well-known vulnerabilities. The detection of such vulnerabilities can be made via formal approaches, but their successful combination requires human creativity. TAMIS's central goal is thus to demonstrably narrow the gap between the vulnerabilities found using formal verification and the issues found using systems engineering. As a second example, we point out that there are classes of attacks that exploit both the software and hardware parts of a system. Although vulnerabilities can be detected via formal methods in the software part, the impact of attacking the hardware still needs to be modeled. This is often done by observing the effect of parameter changes on the system, and capturing a model of them. To address this situation, the TAMIS team bundled resources from scalable formal verification and secure software engineering for *vulnerability analysis*, which we extend to provide methods and tools to (a) *analyze (binary) code including obfuscated malware*, and (b) *build secure systems*.

Very concrete examples better illustrate the differences and complementarity of engineering and formal techniques. First, it is well-known that formal methods can be used to detect buffer overflows. However, the definition of buffer overflows itself was made first in 1972 when the Computer Security Technology Planning study laid out the technique and claimed that over sizing could be exploited to corrupt a system. This exploit was then popularized in 1988 as one of the exploits used by the Morris worm, and only at that point systematic techniques were developed to detect it. Another example is the work we conducted in attacking smart cards. The very firsts experiments were done at the engineering level, and consisted of retrieving the key of the card in a brute force manner. Based on this knowledge, we generated user test-cases that characterize what should not happen. Later, those were used in a fully automatized model-based testing approach [18].

3. Research Program

3.1. Axis 1: Vulnerability analysis

This axis proposes different techniques to discover vulnerabilities in systems. The outcomes of this axis are (a) new techniques to discover system vulnerabilities as well as to analyze them, and (b) to understand the importance of the hardware support.

Most existing approaches used at the engineering level rely on testing and fuzzing. Such techniques consist in simulating the system for various input values, and then checking that the result conforms to a given standard. The problem being the large set of inputs to be potentially tested. Existing solutions propose to extract significant sets by mutating a finite set of inputs. Other solutions, especially concolic testing developed at Microsoft, propose to exploit symbolic executions to extract constraints on new values. We build on those existing work, and extend them with recent techniques based on dissimilarity distances and learning. We also account for the execution environment, and study techniques based on the combination of timing attacks with fuzzing techniques to discover and classify classes of behavior of the system under test.

Techniques such as model checking and static analysis have been used for verifying several types of requirements such as safety and reliability. Recently, several works have attempted to adapt model checking to the detection of security issues. It has clearly been identified that this required to work at the level of binary code. Applying formal techniques to such code requires the development of disassembly techniques to obtain a semantically well-defined model. One of the biggest issues faced with formal analysis is the state space explosion problem. This problem is amplified in our context as representations of data (such as stack content) definitively blow up the state space. We propose to use statistical model checking (SMC) of rare events to efficiently identify problematic behaviors.

We also seek to understand vulnerabilities at the architecture and hardware levels. Particularly, we evaluate vulnerabilities of the interfaces and how an adversary could use them to get access to core assets in the system. One particular mechanism to be investigated is the DMA and the so-called Trustzone. An ad-hoc technique to defend against adversarial DMA-access to memory is to keep key material exclusively in registers. This implies co-analyzing machine code and an accurate hardware model.

3.2. Axis 2: Malware analysis

Axis 1 is concerned with vulnerabilities. Such vulnerabilities can be exploited by an attacker in order to introduce malicious behaviors in a system. Another method to identify vulnerabilities is to analyze malware that exploits them. However, modern malware has a wide variety of analysis avoidance techniques. In particular, attackers obfuscate the code leading to a security exploit. For doing so, recent black hat research suggests hiding constants in program choices via polynomials. Such techniques hinder forensic analysis by making detailed analysis labor intensive and time consuming. The objective of research axis 2 is to obtain a full tool chain for malware analysis starting from (a) the observability of the malware via deobfuscation, and (b) the analysis of the resulting binary file. A complementary objective is to understand how hardware attacks can be exploited by malwares.

We first investigate obfuscation techniques. Several solutions exist to mitigate the packer problem. As an example, we try to reverse the packer and remove the environment evaluation in such a way that it performs the same actions and outputs the resulting binary for further analysis. There is a wide range of techniques to obfuscate malware, which includes flattening and virtualization. We will produce a taxonomy of both techniques and tools. We will first give a particular focus to control flow obfuscation via mixed Boolean algebra, which is highly deployed for malware obfuscation. We recently showed that a subset of them can be broken via SAT-solving and synthesis. Then, we will expand our research to other obfuscation techniques.

Once the malware code has been unpacked/deobfuscated, the resulting binary still needs to be fully understood. Advanced malware often contains multiple stages, multiple exploits and may unpack additional features based on its environment. Ensuring that one understands all interesting execution paths of a malware sample is related to enumerating all of the possible execution paths when checking a system for vulnerabilities. The main difference is that in one case we are interested in finding vulnerabilities and in the other in finding exploitative behavior that may mutate. Still, some of the techniques of Axis 1 can be helpful in analyzing malware. The main challenge for axis 2 is thus to adapt the tools and techniques to deal with binary programs as inputs, as well as the logic used to specify malware behavior, including behavior with potentially rare occurrences. Another challenge is to take mutation into account, which we plan to do by exploiting mining algorithms.

Most recent attacks against hardware are based on fault injection which dynamically modifies the semantics of the code. We demonstrated the possibility to obfuscate code using constraint solver in such a way that the code becomes intentionally hostile while hit by a laser beam. This new form of obfuscation opens a new challenge for secure devices where malicious programs can be designed and uploaded that defeat comprehensive static analysis tools or code reviews, due to their multi-semantic nature. We have shown on several products that such an attack cannot be mitigated with the current defenses embedded in Java cards. In this research, we first aim at extending the work on fault injection, then at developing new techniques to analyze such hostile code. This is done by proposing formal models of fault injection, and then reusing results from our work on obfuscation/deobfuscation.

4. Highlights of the Year

4.1. Highlights of the Year

4.1.1. Kick-off of the ANR JCJC AHMA project

The ANR JCJC project lead by Annelie Heuser was kicked-off, and a PostDoc (Matthieu Mastio) and PhD (Duy Phuc Pham) have been hired. The team already created a first platform for automated hardware malware analysis. See below and in the following.

4.1.2. New results in the TeamPlay H2020 project, coordinator

The project is coordinated by Olivier Zendra. The TeamPlay H2020 project had a successful mid-term review in October 2019, where the reviewers stressed the quality of the overall work. We TAMIS also achieved new results on security modelling in this TeamPlay project in 2019 (see in the following).

4.1.3. New software and platforms

In 2019, we continued the development of several software and platforms (hardware and software), and build up four new ones:

- E-PAC, an Evolving Packer Classifier,
- The SABR (Semantic-driven Analysis of BinaRies) platform
- Orqal, an efficient schedueler for docker images.
- A Side-channel deep learning evaluation platform,
- The AHMA (IoT malware classification through side-channel information) platform and tools.

5. New Software and Platforms

5.1. MASSE

Modular Automated Syntactic Signature Extraction

KEYWORDS: Malware - Syntactic analysis

FUNCTIONAL DESCRIPTION: The Modular Automated Syntactic Signature Extraction (MASSE) architecture is a new integrated open source client-server architecture for syntactic malware detection and analysis based on the YARA, developed with Teclib'. MASSE includes highly effective automated syntactic malware detection rule generation for the clients based on a server-side modular malware detection system. Multiple techniques are used to make MASSE effective at detecting malware while keeping it from disrupting users and hindering reverse-engineering of its malware analysis by malware creators. MASSE integrates YARA in a distributed system able to detect malware on endpoint systems using YARA, analyze malware with multiple analysis techniques, automatically generate syntactic malware detection rules, and deploy the new rules to the endpoints. The MASSE architecture is freely available to companies and institutions as a complete, modular, self-maintained antivirus solution. Using MASSE, a security department can immediately update the rule database of the whole company, stopping an infection on its tracks and preventing future ones.

- Participants: Bruno Lebon, Olivier Zendra, Alexander Zhdanov and Fabrizio Biondi
- Contact: Bruno Lebon

5.2. IoTMLT

IoT Modeling Language and tool

KEYWORDS: Internet of things - Modeling language - Cyber attack

SCIENTIFIC DESCRIPTION: We propose a framework to analyze security in IoT systems consisting of a formal languages for modeling IoT systems and of attack trees for modeling the possible attacks on the system. In our approach a malicious entity is present in the system, called the Attacker. The other IoT entities can inadvertently help the Attacker, by leaking their sensitive data. Equipped with the acquired knowledge the Attacker can then communicate with the IoT entities undetected. The attack tree provided with the model acts as a monitor: It observes the interactions the Attacker has with the system and detects when an attack is successful.

An IoT system is then analyzed using statistical model checking (SMC). The first method we use is Monte Carlo, which consists of sampling the executions of an IoT system and computing the probability of a successful attack based on the number of executions for which the attack was successful. However, the evaluation may be difficult if a successful attack is rare. We therefore propose a second SMC method, developed for rare events, called importance splitting. Both methods are proposed by Plasma, the SMC tool we use.

FUNCTIONAL DESCRIPTION: The IoT modeling language is a formal language and tool for specifying and enforcing security in IoT systems.

- Participants: Delphine Beaulaton, Ioana-Domnina Cristescu and Najah Ben Said
- Partner: Vérimag
- Contact: Delphine Beaulaton
- URL: http://iot-modeling.gforge.inria.fr

5.3. SimFI

Tool for Simulation Fault injection

KEYWORDS: Fault injection - Fault-tolerance

FUNCTIONAL DESCRIPTION: Fault injections are used to test the robust and security of systems. We have developed SimFI, a tool that can be used to simulate fault injection attacks against binary files. SimFI is lightweight utility designed to be integrated into larger environments as part of robustness testing and fault injection vulnerability detection.

- Contact: Nisrine Jafri
- URL: https://github.com/nisrine/Fault-Injection-Tool

5.4. AHMA

Automatic Malware Hardware Analysis

KEYWORDS: Side-channel - Deep learning - Malware

FUNCTIONAL DESCRIPTION: This framework is composed of several parts, each one of them taking in charge the generation and the processing of the data at different levels. Drivers have been developed to automatically control the different oscilloscopes we are working with (picoscope 6407 et infiniium keysight). We use signal processing tools on the raw data to feed a deep neural network which is in charge of classifying the observed malwares. We are using two different approaches to manage the infection of the system. The first one is to reinitialize it each time we make a measurement to ensure its integrity. We have proposed a method allowing to speed the procedure up a lot. Besides, we developed several malwares, to make our experiments in a controlled environment, to avoid the necessity of cleaning the system up after each measurement.

- Contact: Annelie Heuser

5.5. SABR

Semantic-driven Analysis of BinaRies

KEYWORDS: Malware - Semantic - Binary analysis - Unsupervised graph clustering SCDG - Machine learning

FUNCTIONAL DESCRIPTION: Toolchain for binary analysis based on different techniques for capturing binaries' semantics and performing machine learning-assisted analysis. The primary use is malware analysis for malware detection and classification, either based on supervised and unsupervised learning.

This toolchain includes modules of the former BMA toolchain, specifically the SCDG extraction.

Our approach is based on artificial intelligence. We use concolic analysis to extract behavioral signatures from binaries in a form of system call dependency graphs (SCDGs). Our software can do both supervised and unsupervised learning. The former learns the distinctive features of different malware families on a large training set in order to classify the new binaries as malware or cleanware according to their behavioural signatures. In the unsupervised learning the binaries are clustered according to their graph similarity. The toolchain is orchestrated by an experiment manager that allows to easily setup, launch and view results of all modules of the toolchain.

- Contact: Olivier Zendra

5.6. ORQAL

ORQchestration of ALgorithms

KEYWORDS: Docker - Orchestration

FUNCTIONAL DESCRIPTION: ORQAL is a simple batch scheduler for docker cluster which can be used to remotely and without overhead in scientific experiment.

- Contact: Olivier Zendra

5.7. Side-channel deep learning evaluation platform

KEYWORDS: Deep learning - Evaluation

FUNCTIONAL DESCRIPTION: Our platform is based on several software. The first software permits to train a deep neural network and evaluate it for side-channel analysis, we evaluate our neural network with guessing entropy metrics. The second software is used for programming and communicating with the target devices, but we also develop a software to communicate with the equipment and made some measurement for side-channel analysis. The last software is used to make some attack and analysis of side-channel (e.g. made Correlation Power Analysis)

- Contact: Annelie Heuser

5.8. E-PAC

Evolving-PAcker Classifier

KEYWORDS: Packer classification - Incremental learning - Clustering - Malware - Obfuscation

FUNCTIONAL DESCRIPTION: E-PAC is an Evolving packer classifier that identifies the class of the packer used in a batch of packed binaries given in input. The software has the ability to identify both known packer classes and new unseen packer classes. After each update, the evolving classifier self-updates itself with the predicted packer classes.

The software is based on a semi-supervised machine learning system composed of an offline phase and an online phase. In the offline phase, a set of features is extracted from a collection of packed binaries provided with their ground truth labels, then a density-based clustering algorithm (DBSCAN) is used to group similar packers together with respect to a distance measure. In this step, the similarity threshold is tuned in order to form the clusters that fit the best with the the set of labels provided.

In the online phase, the software reproduces the same operations of features extraction and distances calculation with the incoming packed samples, then uses a customized version of the incremental clustering algorithm DBSCAN in order to classify them, either in knowns packer classes or fom new packer classes, or provisoirely leave them unclassified (notion of noise with DBSCAN).

The clusters formed after each update serve as a baseline for the application to self-evolve.

- Contact: Lamine Noureddine

6. New Results

6.1. Results for Axis 1: Vulnerability analysis

6.1.1. New Advances on Side-channel Distinguishers

Participants: Christophe Genevey Metat, Annelie Heuser.

A Systematic Evaluation of Profiling Through Focused Feature Selection.
Profiled side-channel attacks consist of several steps one needs to take. An important, but sometimes ignored, step is a selection of the points of interest (features) within side-channel measurement traces. A large majority of the related works start the analyses with an assumption that the features are preselected. Contrary to this assumption, here, we concentrate on the feature selection step. We investigate how advanced feature selection techniques stemming from the machine learning domain can be used to improve the attack efficiency. To this end, we provide a systematic evaluation of the methods of interest. The experiments are performed on several real-world data sets containing software and hardware implementations of AES, including the random delay countermeasure. Our results show that wrapper and hybrid feature selection methods perform extremely well over a wide range of test scenarios and a number of features selected. We emphasize L1 regularization (wrapper approach) and linear support vector machine (SVM) with recursive feature elimination used after chi-square filter (Hybrid approach) that performs well in both accuracy and guessing entropy. Finally, we show that the use of appropriate feature selection techniques is more important for an attack on the high-noise data sets, including those with countermeasures, than on the low-noise ones.

[3] Make Some Noise. Unleashing the Power of Convolutional Neural Networks for Profiled Side-channel Analysis. *Profiled side-channel analysis based on deep learning, and more precisely Convolutional Neural Networks, is a paradigm showing significant potential. The results, although scarce for now, suggest that such techniques are even able to break cryptographic implementations protected with countermeasures. In this paper, we start by proposing a new Convolutional Neural Network instance able to reach high performance for a number of considered datasets. We compare*

our neural network with the one designed for a particular dataset with masking countermeasure and we show that both are good designs but also that neither can be considered as a superior to the other one. Next, we address how the addition of artificial noise to the input signal can be actually beneficial to the performance of the neural network. Such noise addition is equivalent to the regularization term in the objective function. By using this technique, we are able to reduce the number of measurements needed to reveal the secret key by orders of magnitude for both neural networks. Our new convolutional neural network instance with added noise is able to break the implementation protected with the random delay countermeasure by using only 3 traces in the attack phase. To further strengthen our experimental results, we investigate the performance with a varying number of training samples, noise levels, and epochs. Our findings show that adding noise is beneficial throughout all training set sizes and epochs.

The Curse of Class Imbalance and Conflicting Metrics with Machine Learning for Side-channel Evaluations.
We concentrate on machine learning techniques used for profiled sidechannel analysis in the presence of imbalanced data. Such scenarios are realistic and often occurring, for instance in the Hamming weight or Hamming distance leakage models. In order to deal with the imbalanced data, we use various balancing techniques and we show that most of them help in mounting successful attacks when the data is highly imbalanced. Especially, the results with the SMOTE technique are encouraging, since we observe some scenarios where it reduces the number of necessary measurements more than 8 times. Next, we provide extensive results on comparison of machine learning and side-channel metrics, where we show that machine learning metrics (and especially accuracy as the most often used one) can be extremely deceptive. This finding opens a need to revisit the previous works and their results in order to properly assess the performance of machine learning in side-channel analysis.

[5] CC Meets FIPS: A Hybrid Test Methodology for First Order Side Channel Analysis.
Common Criteria (CC) and FIPS 140-3 are two popular side channel testing methodologies. Test Vector Leakage Assessment Methodology (TVLA), a potential candidate for FIPS, can detect the presence of side-channel information in leakage measurements. However, TVLA results cannot be used to quantify side-channel vulnerability and it is an open problem to derive its relationship with side channel attack success rate (SR), i.e., a common metric for CC. In this paper, we extend the TVLA testing beyond its current scope. Precisely, we derive a concrete relationship between TVLA and signal to noise ratio (SNR). The linking of the two metrics allows direct computation of success rate (SR) from TVLA for given choice of intermediate variable and leakage model and thus unify these popular side channel detection and evaluation metrics. An end-to-end methodology is proposed, which can be easily automated, to derive attack SR starting from TVLA testing. The methodology works under both univariate and multivariate setting and is capable of quantifying any first order leakage. Detailed experiments have been provided using both simulated traces and real traces on SAKURA-GW platform. Additionally, the proposed methodology is benchmarked against previously published attacks on DPA contest v4.0 traces, followed by extension to jitter based countermeasure. The result shows that the proposed methodology provides a quick estimate of SR without performing actual attacks, thus bridging the gap between CC and FIPS.

[13] Combining sources of side-channel information.
A few papers relate that multi-channel consideration can be beneficial for side-channel analysis. However, all were conducted using classical attack techniques. In this work, we propose to explore a multi-channel approach thanks to machine/deep learning. We investigate two kinds of multi-channel combinations. Unlike previous works, we investigate the combination of EM emissions from different locations capturing data-dependent leakage information on the device. Additionally, we consider the combination of the classical leaking signals and a measure of mostly the ambient noise. The knowledge of the ambient noise (due to WiFi, GSM, ...) may help to remove it from a noisy trace. To investigate these multi-channel approaches, we describe one option of how to extend a CNN architecture which takes as input multiple channels. Our results show that multi-channel networks

are suitable for side-channel analysis. However, if one channel alone already contains enough exploitable information to reach high effectiveness, naturally, the multi-channel approach cannot improve the performance further.

6.1.2. Side-channel analysis on post-quantum cryptography

Participants: Tania Richmond, Yulliwas Ameur, Agathe Cheriere, Annelie Heuser.

In recent years, there has been a substantial amount of research on quantum computers ? machines that exploit quantum mechanical phenomena to solve mathematical problems that are difficult or intractable for conventional computers. If large-scale quantum computers are ever built, they will be able to break many of the public-key cryptosystems currently in use. This would seriously compromise the confidentiality and integrity of digital communications on the Internet and elsewhere. The goal of post-quantum cryptography (also called quantum-resistant cryptography) is to develop cryptographic systems that are secure against both quantum and classical computers, and can interoperate with existing communications protocols and networks. At present, there are several post-quantum cryptosystems that have been proposed: lattice-based, code-based, multivariate cryptosystems, hash-based signatures, and others. However, for most of these proposals, further research is needed in order to gain more confidence in their security and to improve their performance. Our interest lies in particular on the side-channel analysis and resistance of these post-quantum schemes, in particular code-based cryptosystems.

During this year, we have set up a first side-channel experiment platform suited for embedded devices running code-based cryptosystems. Using this platform we exploited vulnerabilities of the syndrome computation present in some code-based algorithms.

6.1.3. Verification of IKEv2 protocol

Participants: Tristan Ninet, Olivier Zendra.

The IKEv2 (Internet Key Exchange version 2) protocol is the authenticated key-exchange protocol used to set up secure communications in an IPsec (Internet Protocol security) architecture. IKEv2 guarantees security properties like mutual-authentication and secrecy of exchanged key. To obtain an IKEv2 implementation as secure as possible, we use model checking to verify the properties on the protocol specification, and software formal verification tools to detect implementation flaws like buffer overflows or memory leaks.

In previous analyses, IKEv2 has been shown to possess two authentication vulnerabilities that were considered not exploitable. We analyze the protocol specification using the Spin model checker, and prove that in fact the first vulnerability does not exist. In addition, we show that the second vulnerability is exploitable by designing and implementing a novel slow Denial-of-Service attack, which we name the Deviation Attack.

We propose an expression of the time at which Denial-of-Service happens, and validate it through experiment on the strongSwan implementation of IKEv2. As a counter-measure, we propose a modification of IKEv2, and use model checking to prove that the modified version is secure.

For ethical reasons we informed our country's national security agency (ANSSI) about the existence of the Deviation Attack. The security agency gave us some technical feedback as well as its approval for publishing the attack.

We then tackle formal verification directly applied to an IKEv2 source code. We already tried to analyze strongSwan using the Angr tool. However we found that the Angr was not mature yet for a program like strongSwan. We thus try other software formal verification tools and apply them to smaller and simpler source code than strongSwan: we analyze OpenSSL asn1parse using the CBMC tool and light-weight IP using the Infer tool. We find that CBMC does not scale to a large source code and that Infer does not verify the properties we want.

We explored more in-depth a formal technique and work towards the goal of verifying generic properties (absence of implementation flaws) on softwares like strongSwan.

Publications:

- [10] Model Checking the IKEv2 Protocol Using Spin
- [11] The Deviation Attack: A Novel Denial-of-Service Attack Against IKEv2

6.1.4. Software obfuscation

Participants: Alexandre Gonzalvez, Olivier Decourbe.

The limits of software obfuscation are not clear in practice. A protection based on opaque predicates can not be compatible with the control flow integrity property at low-level, due to the presence of indirect jumps in the instruction set architecture semantics. We propose a restricted instruction set architecture to overcome this limit. We argue for the adoption of restricted instruction set architecture for security-related computation. Publication:

- [9] A case against indirect jumps for secure programs

6.2. Results for Axis 2: Malware analysis

The detection of malicious programs is a fundamental step to be able to guarantee system security. Programs that exhibit malicious behavior, or *malware*, are commonly used in all sort of cyberattacks. They can be used to gain remote access on a system, spy on its users, exfiltrate and modify data, execute denial of services attacks, etc.

Significant efforts are being undertaken by software and data companies and researchers to protect systems, locate infections, and reverse damage inflicted by malware. Our contribution to malware analysis include the following fields:

6.2.1. Malware Classification and clustering

Participants: Cassius Puodzius, Stefano Sebastio, Olivier Decourbe, Annelie Heuser, Olivier Zendra.

Once malicious behavior has been located, it is essential to be able to classify the malware in its specific family to know how to disinfect the system and reverse the damage inflicted on it.

While it is rare to find an actually previously unknown malware, morphic techniques are employed by malware creators to ensure that different generations of the same malware behave differently enough than it is hard to recognize them as belonging to the same family. In particular, techniques based on the syntax of the program fails against morphic malware, since syntax can be easily changed.

To this end, semantic signatures are used to classify malware in the appropriate family. Semantic signatures capture the malware's behavior, and are thus resistant to morphic and differentiation techniques that modify the malware's syntactic signatures. We are investigating semantic signatures based on the program's System Call Dependency Graph (SCDG), which have been proven to be effective and compact enough to be used in practice. SCDGs are often extracted using a technique based on pushdown automata that is ineffective against obfuscated code; instead, we are applying concolic analysis via the `angr` engine to improve speed and coverage of the extraction.

Once a semantic signature has been extracted, it has to be compared against large database of known signatures representing the various malware families to classify it. The most efficient way to obtain this is to use a supervised machine learning classifier. In this approach, the classifier is trained with a large sample of signatures malware annotated with the appropriate information about the malware families, so that it can learn to quickly and automatically classify signatures in the appropriate family. Our work on machine learning classification focuses on using SCDGs as signatures. Since SCDGs are graphs, we are investigating and adapting algorithms for the machine learning classification of graphs, usually based on measures of shared subgraphs between different graphs. One of our analysis techniques relies on common subgraph extraction, with the idea that a malicious behavior characteristic of a malware family will yield a set of common subgraphs. Another approach relies on the Weisfeiler-Lehman graph kernel which uses the presence of nodes and their neighborhoods pattern to evaluate similarity between graphs. The presence or not of a given pattern becomes a feature in a subsequent machine learning analysis through random forest or SVM.

Moreover, we explored the impact on the malware classification of several heuristics adoptable in the SCDGs building process and graph exploration. In particular, our purpose was to:

- identify quality characteristics and evaluation metrics of binary signatures based on SCDGs (and consequently the key properties of the execution traces), that characterize signatures able to provide high-precision malware classification
- optimize the performance of the SMT solver by designing a meta-heuristic able to select the best heuristic to tackle a specific sub-class of problem, study the impact of the configuration of the SMT solver and symbolic execution framework, and understand their interdependencies with the aim of efficiently extracting SCDGs in accordance with the identified quality metrics.

By adopting a Design of Experiments approach constituted by a full factorial experiment design and an Analysis of Variance (ANOVA) we have been able to pinpoint that, considering the graph metrics and their impact on the F-score, the litmus test for the quality of an SCDG-based classifier is represented by the presence of connected components. This could be explained considering how the graph mining algorithm (gSpan) works and the adopted similarity metric based on the number of common edges between the extracted signatures and the SCDG of the sample to classify. The results of the factorial experiments show that in our context tuning the symbolic execution is a very complex problem and that the sparsity of effect principle (stating that the system is dominated by the effect of the main factors and low-order-factor interactions) does not hold. The evaluation proved that the SMT solver is the most influential positive factor also showing an ability in reducing the impact of heuristics that may need to be enabled due to resource constraints (e.g., the max number of active paths). Results suggest that the most important factors are the disjoint union (as trace combination heuristic), and the our SMT optimization (through meta-heuristics) whereas other heuristics (such as min trace size and step timeout) have less impact on the quality of the constructed SCDGs.

During this year we build a end-to-end functional toolchain for supervised learning.

Furthermore, we have extended our approach to malware classification using unsupervised clustering. Preliminary results show that we are able to classify malware according to their behavioral properties without the need of any predefined labels.

6.2.2. Packers analysis

Participants: Lamine Nourredine, Cassius Puodzius, Stefano Sebastio, Annelie Heuser, Olivier Zendra.

Packing is a widespread tool to prevent static malware detection and analysis. Detecting and classifying the packer used by a given malware sample is fundamental to being able to unpack and study the malware, whether manually or automatically. Existing works on packing detection and classification has focused on effectiveness, but does not consider the efficiency required to be part of a practical malware-analysis workflow. This work studies how to train packing detection and classification algorithms based on machine learning to be both highly effective and efficient. Initially, we create ground truths by labeling more than 280,000 samples with three different techniques. Then we perform feature selection considering the contribution and computation cost of features. Then we iterate over more than 1,500 combinations of features, scenarios, and algorithms to determine which algorithms are the most effective and efficient, finding that a reduction of 1-2% effectiveness can increase efficiency by 17-44 times. Then, we test how the best algorithms perform against malware collected after the training data to assess them against new packing techniques and versions, finding a large impact of the ground truth used on algorithm robustness. Finally, we perform an economic analysis and find simple algorithms with small feature sets to be more economical than complex algorithms with large feature sets based on uptime/training time ratio.

A limit of supervised learning is to not be able to recognize classes that were not present in the ground truth. In the work's case above, this means that packer families for which a classifier has not been trained will not be recognized. In this work, we use unsupervised learning techniques, more particularly clustering, in order to provide information about packed malware with previously unknown packing techniques. Here, we build our own dataset of packed binaries, since in the previous work, it has been shown that the construction of the ground truth was fundamental in determining the effectiveness of the packing classification process. Choosing

the right clustering algorithm with the right distance metric, dealing with different scales of features units, while being effective, efficient and robust are also majors parts of the current work.

During this year we have developed a toolchain of effective clustering of packers, in particular taking into account the possibility of evolution in packers. For this we derived and implemented new feature extraction strategies combined with incremental clustering algorithms.

6.3. (Coordination of the) H2020 TeamPlay Project, and Expression of Security Properties

Participants: Olivier Zendra, Yoann Marquer, Céline Minh, Nicolas Kiss, Annelie Heuser, Tania Richmond.

6.3.1. Overview & results

This work is done in the context of the TeamPlay EU project.

As mobile applications, the Internet of Things, and cyber-physical systems become more prevalent, so there is an increasing focus on energy efficiency of multicore computing applications. At the same time, traditional performance issues remain equally important. Increasingly, software designs need to find the best performance within some energy budget, often while also respecting real-time or other constraints, which may include security, data locality or system criticality, and while simultaneously optimising the usage of the available hardware resources.

While parallel multicore/manycore hardware can, in principle, ameliorate energy problems, and heterogeneous systems can help to find a good balance between execution time and energy usage, at present there are no effective analyses beyond user-guided simulations that can reliably predict energy usage for parallel systems, whether alone or in combination with timing information and security properties. In order to create energy-, time- and security- (ETS) efficient parallel software, programmers need to be actively engaged in decisions about energy usage, execution time and security properties rather than passively informed about their effects. This extends to design-time as well as to implementation-time and run-time.

In order to address this fundamental challenge, TeamPlay takes a radically new approach: by exploiting new and emerging ideas that allow non-functional properties to be deeply embedded within their programs, programmers can be empowered to directly treat energy ETS properties as first-class citizens in their parallel software. The concrete objectives of the TeamPlay project are:

1. To develop new mechanisms, along with their theoretical and practical underpinnings, that support direct language-level reasoning about energy usage, timing behaviour, security, etc.
2. To develop system-level coordination mechanisms that facilitate optimised resource usage for multicore hardware, combining system-level resource utilisation control during software development with efficient spatial and temporal scheduling at run-time.
3. To determine the fundamental inter-relationships between time, energy, security, etc. optimisations, to establish which optimisation approaches are most effective for which criteria, and to consequently develop multiobjective optimising compilers that can balance energy consumption against timing and other constraints.
4. To develop energy models for heterogeneous multicore architectures that are sufficiently accurate to enable high-level reasoning and optimisation during system development and at run-time.
5. To develop static and dynamic analyses that are capable of determining accurate time, energy usage and security information for code fragments in a way that can inform high-level programs, so achieving energy, time and security transparency at the source code level.
6. To integrate these models, analyses and tools into an analysis-based toolbox that is capable of reflecting accurate static and dynamic information on execution time and energy consumption to the programmer and that is capable of optimising time, energy, security and other required metrics at the whole system level.

7. To identify industrially-relevant metrics and requirements and to evaluate the effectiveness and potential of our research using these metrics and requirements.
8. To promote the adoption of advanced energy-, time- and security-aware software engineering techniques and tools among the relevant stake-holders.

Inria will exploit the results of the TeamPlay project in two main domains. First, they will strengthen and extend the research Inria has been carrying on low power and energy for embedded systems, especially for memory and wireless sensors networks. Second, they will complement in a very fitting way the research carried at Inria about security at a higher level (model checking, information theory).

The capability to express the energy and security properties at the developper level will be integrate in Inria own prototype tools, hence widening their applicability and the ease of experimentation. The use of energy properties wrt. evening of energy consumption to prevent information leakage, thus making side-channels attacks more difficult, is also a very promising path.

In addition, the methodological results pertaining to the development of embedded systems with a focus on low power and energy should also contribute to research lead at Inria in the domain of software engineering and advanced software engineering tools. Furthermore, security research lead at Inria will benefit from the security work undertaken by Inria and SIC in TeamPlay.

Overall, the project, with a strong industrial presence, will allow Inria to focus on matching concrete industrial requirements aiming at actual products, hence in providing more robust and validated results. In addition, the extra experience of working with industrial partners including SMEs will surely impact positively on Inria research methodology, making Inria research more attractive and influential, especially wrt. industry.

Finally, the results, both in terms of methodology and techniques, will also be integrated in the teaching Inria contributes to at Master level, in the areas of Embedded Systems and of Security.

The TeamPlay consortium agreement has been created by Inria, discussed with the various partners, and has been signed by all partners on 28 Feb. 2018. Inria has also distributed the partners initial share of the grant at the beginning of the project.

As WP7 (project management) leader and project coordinator, Inria was in charge of arranging general project meetings, including monthly meetings (tele-conferences), bi-annual physical meetings, boards meetings. During the first period, three exceptional physical meetings have been conducted, in addition to monthly project meetings: the kick-off meeting in Rennes from the 30th to the 31st of January 2018, the physical progress meeting has been conducted in Odense from the 26th to the 27th of June 2018, and the review in Brussels prepared the 19th of September 2018 and set the 17th of October 2018.

We have selected and set up utility tools for TeamPlay: shared notepads, mailing lists, shared calendars and collaborative repositories. We have ensured the timely production of the due deliverables. We set up the Project Advisory Board (PAB) with the aim of gathering external experts from both academia and industry, covering a wide range of domains addressed by TeamPlay. Finally, we ensured good working relationships (which can implicate conflict resolution when needed), monitored the overall progress of the project, and reported to the European Commission on technical matters and deliverables.

We also organized a tooling meeting in Hamburg in October the 30th, to discuss the relation between the tools from different partners, e.g. Idris from the University of St Andrews, the WCC compiler developed in the Hamburg University of Technology, or the coordination tool developed in the University of Amsterdam.

Measuring security, unlike measuring other more common non-functional properties like time or energy, is still very much in its infancy. For example, time is often measured in seconds (or divisions thereof), but security has no widely agreed, well-defined measurement. It is thus one goal of this project, especially for SIC and Inria, to design (necessarily novel) security measurements, and have them implemented as much as possible throughout the set of development tools.

Measuring security by only one value however seems impossible or may be meaningless. More precisely, if security could be defined overall by only one measurement, the latter would be a compound (i.e. an aggregation) of several more specialized measurement. Indeed, security encompasses many aspects of interest:

1. By allowing communications between different systems, security properties should be guaranteed in order to prevent low-level users from determining anything about high-level users activity, or in the case of public communication channels in a hostile environment, to evaluate vulnerability to intruders performing attacks on communications.
    1. *Confidentiality* (sometimes called *secrecy*) properties like non-interference (and many) variants can be described by using an information-flow policy (e.g. high- and low-level users) and studying traces of user inputs.
    2. *Vulnerability* captures how a system is sensible to attacks on communications (e.g. stealing or faking information on a public channel).
2. A *side-channel* is a way of transmitting informations (purposely or not) to another system out of the standard (intended) communication channels. *Side-channel attacks* rely on the relationship between information leaked through a side-channel and the secret data to obtain confidential (non-public) information.
    1. *Entropy* captures the uncertainty of the attacker about the secret key. The attacker must be able to extract information about the secret key through side-channel measurements, which is captured by the *attacker's remaining uncertainty* value, which can be computed by using heuristic techniques. The attacker must also be able to effectively recover the key from the extracted information, which is expressed by the *min-entropy leakage*, and refined by the *g-leakage* of a gain function.
    2. The power consumption of a cryptographic device can be analyzed to extract the secret key. This is done by using several techniques: visual examination of graphs of the current (*Simple Power Analysis*), by exploiting biases in varying power consumption (*Differential Power Analysis*), or by using the correlation coefficient between the power samples and hypotheses (*Correlation Power Analysis*).
    3. Usual security properties guarantee only the input-output behavior of a program, and not its execution time. Closing *leakage through timing* can be done by disallowing while-loops and if-commands to depend on high security data, or by padding the branches so that the external observer cannot determine which branch was taken.
    4. Finally, the correlation between the patterns of the victim's execution and the attacker's observations is formalized as a metric called the *Side-channel Vulnerability Factor*, which is refined by the *Cache Side-channel Vulnerability* for cache attacks.
3. A cryptographic scheme should be secure even if the attacker knows all details about the system, with the exception of the secret keys. In particular, the system should be secure when the attacker knows the encryption and decryption algorithms.
    1. In modern cryptography, the security level (or security strength) is given by the *work factor*, which is related to its key-length and the number of operations necessary to break a cryptographic scheme (try all possible combinations of the key). An algorithm is said to have a "security level of n bits" if the best known attack requires 2^n steps. This is a quite natural definition because symmetric algorithms with a security level of n have a key of length n bits.
    2. The relationship between cryptographic strength and security is not as straightforward in the asymmetric case. Moreover, for symmetric algorithms, a key-length of 128 bits provides an estimated long term security (i.e. several decades in the absence of quantum computer) regarding brute-force attacks. To reach an estimated long term security even with quantum computers, a key-length of 256 bits is mandatory.

Inria is implementing side-channel countermeasures (hiding) into the WCET-aware C Compiler (WCC) developed by the Hamburg University of Technology (TUHH). A research visit to TUHH was arranged with the aim at learning how to work on WCC (TUHH and WCC infrastructure, WCC developers best practices, etc.). Inria will use compiler-based techniques to prevent timing leakages and power leakages.

For instance, in a conditional branching `if` b `then` $P_1(x)$ `else` $P_2(x)$, measuring the execution time or the power profile may allow to know whether the branch P_1 or P_2 have been chosen to manipulate the value x, thus to obtain the secret value b. To prevent timing leakage, P_1 and/or P_2 can be padded (i.e. dummy instructions are added) in order to obtain the worst-case execution time in both branches.

But this does not prevent information leakage from power profile. A stronger technique, from a security point of view, could be to add a dummy variable y and duplicate the code such that $y = x$; `if` b `then` $P_1(x); P_2(y)$ `else` $P_1(Y); P_2(x)$ always performs the operations of P_1 then the operations of P_2. But the execution time is now the sum and not the worst-case of both branches, thus trading execution time to increase security.

Finally, the initialization $y = x$ can be detected, and the previous solution is still vulnerable to fault injections. Some algorithms like the Montgomery Ladder are more protected against these attacks because both variables x and y are entangled during the execution. We hope to generalize this property to a wider set of algorithms, or to automatically detect the properties required from the original code in order to transform it into a “ladderized" version with higher security level.

6.3.2. Publication

- Type-Driven Verification of Non-functional Properties [8].

7. Bilateral Contracts and Grants with Industry

7.1. Bilateral Contracts with Industry

- CISCO (http://www.cisco.com) contract (2017–2019) to work on graph analysis of malware

7.2. Bilateral Grants with Industry

- CISCO (http://www.cisco.com) one grant (2016–2019) to work on semantical analysis of malware
- Thales (https://www.thalesgroup.com) one CIFRE (2016–2019) to work on verification of communication protocols, one grant (2018–2019) to work on learning algorithms
- Oberthur Technologies (http://www.oberthur.com/) one grant (2016–2020) to work on fuzzing and fault injection

8. Partnerships and Cooperations

8.1. National Initiatives

8.1.1. ANR

- ANR MALTHY, Méthodes ALgébriques pour la vérification de modèles Temporisés et HYbrides, Thao Dang, 4 years, Inria and VISEO and CEA and VERIMAG
- ANR COGITO, Runtime Code Generation to Secure Devices, 3 years, Inria and CEA and ENSMSE and XLIM.
- ANR AHMA, Automated Hardware Malware Analysis, 3,5 years, JCJC.

8.1.2. DGA

- PhD grant for Nisrine Jafri (2016–2019),
- PhD grant for Lamine Noureddine (2017-2020)
- PhD grant for Christophe Genevey Metat (2018-2021)
- PhD grant for Cassius De Oliveira Puodzius (2019-2022)

8.2. European Initiatives

8.2.1. ENABLE-S3 (352)

Title: ENABLE-S3: European Initiative to Enable Validation for Highly Automated Safe and Secure Systems

Program: H2020

Duration: 05/2016 - 04/2019

Coordinator: Avl List Gmbh (Austria)

Partners:

Aalborg Universitet (Denmark); Airbus Defence And Space Gmbh (Germany); Ait Austrian Institute Of Technology Gmbh (Austria); Avl Deutschland Gmbh (Germany); Avl Software And Functions Gmbh (Germany); Btc Embedded Systems Ag (Germany); Cavotec Germany Gmbh (Germany); Creanex Oy(Finland); Ceske Vysoke Uceni Technicke V Praze (Czech Republic); Deutsches Zentrum Fuer Luft - Und Raumfahrt Ev (Germany); Denso Automotive Deutschland Gmbh (Germany); Dr. Steffan Datentechnik Gmbh (Austria); Danmarks Tekniske Universitet (Denmark); Evidence Srl (Italy); Stiftung Fzi Forschungszentrum Informatik Am Karlsruher Institut Fur Technologie (Germany); Gmv Aerospace And Defence Sa (Spain); Gmvis Skysoft Sa (Portugal); Politechnika Gdanska (Poland); Hella Aglaia Mobile Vision Gmbh (Germany); Ibm Ireland Limited (Ireland); Interuniversitair Micro-Electronica Centrum (Belgium); Iminds (Belgium); Institut National De Recherche Eninformatique Et Automatique (France); Instituto Superior De Engenharia Do Porto (Portugal); Instituto Tecnologico De Informatica (Spain); Ixion Industry And Aerospace Sl (Spain); Universitat Linz (Austria); Linz Center Of Mechatronics Gmbh (Austria); Magillem Design Services Sas (France); Magneti Marelli S.P.A. (Italy); Microeletronica Maser Slspain); Mdal (France); Model Engineering Solutions Gmbhgermany); Magna Steyr Engineering Ag & Co Kg (Austria); Nabto Aps (Denmark); Navtor As (Norway); Nm Robotic Gmbh (Austria); Nxp Semiconductors Germany Gmbh(Germany); Offis E.V.(Germany); Philips Medical Systems Nederland Bvnetherlands); Rohde & Schwarz Gmbh&Co Kommanditgesellschaft(Germany); Reden B.V. (Netherlands); Renault Sas (France); Rugged Tooling Oyfinland); Serva Transport Systems Gmbh(Germany); Siemens Industry Software Nvbelgium); University Of Southampton (Uk); Safetrans E.V. (Germany); Thales Alenia Space Espana, Saspain); Fundacion Tecnalia Research & Innovationspain); Thales Austria Gmbh (Austria); The Motor Insurance Repair Researchcentre (Uk); Toyota Motor Europe (Belgium); Nederlandse Organisatie Voor Toegepast Natuurwetenschappelijk Onderzoek Tno (Netherlands); Ttcontrol Gmbh (Austria); Tttech Computertechnik Ag (Austria); Technische Universiteit Eindhoven (Netherlands); Technische Universitat Darmstadt (Germany); Technische Universitaet Graz (Austria); Twt Gmbh Science & Innovation (Germany); University College Dublin, National University Of Ireland, Dublin (Ireland); Universidad De Las Palmas De Gran Canaria (Spain); Universita Degli Studi Di Modena E Reggio Emilia (Italy); Universidad Politecnica De Madrid (Spain); Valeo Autoklimatizace K.S. (Czech Republic); Valeo Comfort And Driving Assistance (France); Valeo Schalter Und Sensoren Gmbh (Germany); Kompetenzzentrum - Das Virtuelle Fahrzeug, Forschungsgesellschaft Mbh (Austria); Vires Simulationstechnologie Gmbh (Germany); Teknologian Tutkimuskeskus Vtt Oy (Finland); Tieto Finland Support Services Oy (Finland); Zilinska Univerzita V Ziline (Slovakia);

Inria contact: Olivier Zendra

The objective of ENABLE-S3 (http://www.enable-s3.eu) is to establish cost-efficient cross-domain virtual and semi-virtual V&V platforms and methods for ACPS. Advanced functional, safety and security test methods will be developed in order to significantly reduce the verification and validation

time but preserve the validity of the tests for the requested high operation range. ENABLE-S3 aspires to substitute today's physical validation and verification efforts by virtual testing and verification, coverage-oriented test selection methods and standardization. ENABLE-S3 is use-case driven; these use cases represent relevant environments and scenarios. Each of the models, methods and tools integrated into the validation platform will be applied to at least one use case (under the guidance of the V&V methodology), where they will be validated (TRL 5) and their usability demonstrated (TRL6). Representative use cases and according applications provide the base for the requirements of methods and tools, as well as for the evaluation of automated systems and respective safety. This project is industry driven and has the objective of designing new technologies for autonomous transportation, including to secure them. TAMIS tests its results on the case studies of the project.

Within ENABLE-S3, the contribution of the TAMIS team consists in in proposing a generic method to evaluate complex automotive-oriented systems for automation (perception, decision-making, etc.). The method is based on Statistical Model Checking (SMC), using specifically defined Key Performance Indicators (KPIs), as temporal properties depending on a set of identified metrics. By feeding the values of these metrics during a large number of simulations, and the properties representing the KPIs to our statistical model checker, we evaluate the probability to meet the KPIs. We applied this method to two different subsystems of an autonomous vehicles: a perception system (CMCDOT framework) and a decision-making system. We show that the methodology is suited to efficiently evaluate some critical properties of automotive systems, but also their limitations.

In 2019, in TAMIS, Olivier Zendra and Eduard Baranov were involved in this project. The project supported one postdoc in TAMIS starting in 2017.

8.2.2. TeamPlay (653)

Title: TeamPlay: Time, Energy and security Analysis for Multi/Many-core heterogeneous PLAtforms

Program: H2020

Duration: 01/2018 - 12/2020

Coordinator: Inria

Partners:

> Absint Angewandte Informatik Gmbh (Germany), Institut National De Recherche en Informatique et Automatique (France), Secure-Ic Sas (France), Sky-Watch A/S (Danemark), Syddansk Universitet (Danemark), Systhmata Ypologistikis Orashs Irida Labs Ae (Greece), Technische Universität Hamburg-Harburg (Germany), Thales Alenia Space Espana (Spain), Universiteit Van Amsterdam (Netherlands), University Of Bristol (UK), University Of St Andrews (UK)

Inria contact: Olivier Zendra

The TeamPlay (Time, Energy and security Analysis for Multi/Many-core heterogeneous PLAtforms) project federates 6 academic and 5 industrial partners and aims to develop new, formally-motivated, techniques that will allow execution time, energy usage, security, and other important non-functional properties of parallel software to be treated effectively, and as first- class citizens. We will build this into a toolbox for developing highly parallel software for low-energy systems, as required by the internet of things, cyber-physical systems etc. The TeamPlay approach will allow programs to reflect directly on their own time, energy consumption, security, etc., as well as enabling the developer to reason about both the functional and the non-functional properties of their software at the source code level. Our success will ensure significant progress on a pressing problem of major industrial importance: how to effectively manage energy consumption for parallel systems while maintaining the right balance with other important software metrics, including time, security etc. The project brings together leading industrial and academic experts in parallelism, energy modeling/ transparency, worst-case execution time analysis, non-functional property analysis, compilation,

security, and task coordination. Results will be evaluated using industrial use cases taken from the computer vision, satellites, flying drones, medical and cyber security domains. Within TeamPlay, Inria and TAMIS coordinate the whole project, while being also in charge of aspects related more specifically to security.

The permanent members of TAMIS who are involved are Olivier Zendra and Annelie Heuser.

8.2.3. SUCCESS

Title: SUCCESS: SecUre aCCESSibility for the internet of things

Program: CHIST-ERA 2015

Duration: 10/2016 - 10/2019

Coordinator: Middlesex University (UK)

Partners:

Middlesex University, School of Science and Technology (UK); Inria, TAMIS (France); Université Grenoble Alpes, Verimag (France); University of TWENTE, (Netherlands)

Inria contact: Ioana Cristescu

The objectives of the SUCCESS project is to use formal methods and verification tools with a proven track record to provide more transparency of security risks for people in given IoT scenarios. Our core scientific innovation will consist on the extension of well-known industry-strength methods. Our technological innovation will provide adequate tools to address risk assessment and adaptivity within IoT in healthcare environments and an open source repository to foster future reuse, extension and progress in this area. Our project will validate the scientific and technological innovation through pilots, one of which will be in collaboration with a hospital and will allow all stakeholders (e.g. physicians, hospital technicians, patients and relatives) to enjoy a safer system capable to appropriately handle highly sensitive information on vulnerable people while making security and privacy risks understandable and secure solutions accessible.

Within SUCCESS, the contribution of the TAMIS team consists in a framework for analyzing the security of a given IOT system, and notably whether it resists to attack. Our approach is to build a high-level model of the system, including its vulnerabilities, as well as an attacker. We represent the set of possible attacks using an attack tree. Finally, we evaluate the probability that an attack succeeds using Statistical Model Checking.

In 2019, in the TAMIS team, Delphine Beaulaton, Najah Ben Said, Ioana Cristescu and Olivier Zendra were involved in this project.

9. Dissemination

9.1. Promoting Scientific Activities

9.1.1. Scientific Events: Selection

9.1.1.1. Member of Conference Steering Committees

Olivier Zendra is a founder and a member of the Steering Committee of ICOOOLPS (International Workshop on Implementation, Compilation, Optimization of OO Languages, Programs and Systems)

9.1.1.2. Member of the Conference Program Committees

Annelie Heuser was a member of the following conference program committees: CARDIS 2019, COSADE 2019, PROOFS 2019.
Olivier Zendra was a member of the program committee of ICOOOLPS 2019.

9.1.1.3. Reviewer

Tania Richmond was a reviewer for TCHES 2019 and TCHES 2020.
Alexandre Gonzalvez was a reviewer for TCHES 2019.
Matthieu Mastio was a reviewer for DATE2020.

9.1.2. Journal

9.1.2.1. Member of the Editorial Boards

Annelie Heuser was a member of TCHES 2019.

9.1.2.2. Reviewer - Reviewing Activities

Annelie Heuser was a reviewer of several papers in Journal of Cryptographic Engineering, Transactions on Information Forensics & Security, IEEE Transactions on Circuits and Systems, Transactions on Information Forensics & Security.
Tania Richmond was a reviewer for Transactions on Information Forensics and Security (TIFS)

9.1.3. Invited Talks

Annelie Heuser was invited to present at the seminar of Paris 8 on “Mathématiques Discrètes, Workshop Workshop on Randomness and Arithmetics for Cryptography on Hardware, Codes et Cryptographie”, GDR Sécurité - Sécurité, fiabilité et test des SoC2, the Summer School on real-world crypto and privacy.
Annelie Heuser was invited to the Dagstuhl seminar on Secure Composition for Hardware Systems.
Annelie Heuser was invited to the Lorentzcenter on AI+Sec: Artificial Intelligence and Security, and SHARD: Bridging the Gap Between Software and Hardware Security.

9.1.4. Expertise

Tania Richmond reviewed for the HiPEAC Collaboration Grants 2019.
Annelie Heuser reviewed an application to receive a national application grant (inside Europe).
Olivier Zendra is a CIR expert for the MENESR.
Olivier Zendra participated to the CRHC and CRCN national juries for Inria as a member of Inria’s evaluation committee.
Olivier Zendra is a member of the editorial board and co-author of the “HiPEAC Vision 2021”

9.1.5. Research Administration

Olivier Zendra is a member of Inria’s evaluation committee.

9.2. Teaching - Supervision - Juries

9.2.1. Teaching

Christophe Genevey-Metat taught:

- Programming in Java to Licence 1, University Rennes 1,
- Introduction at Security to Master 1, University Rennes 1.

Cassius de Oliveira Puodzius taught:

- 24h of Reverse Engineering + Malware analysis for the Introduction at Security to Master1 at Rennes 1,
- 6h of Java Programming to Licence1 at ECAM Rennes.

Tania Richmond taught Physical Attacks in "Préparation à l’Agrégation" en Sciences de l’Ingénieur, ENS Rennes.

9.2.2. Supervision

- PhD in progress: Christophe Genevey Metat (Rennes 1): , October 2018, Jean-Marc Jezequel, Benoit Gerard, Annelie Heuser and Clementine Maurice
- PhD in progress : Alexandre Gonzalvez, On Obfuscation via crypto primitives, April 2016, Caroline Fontaine.
- PhD defended in 2019 : Nisrine Jafri (Rennes1), On fault Injection detection with MC of Binary code, Axel Legay and Jean-Louis Lanet.
- PhD in progress : Tristan Ninet (Rennes 1), Vérification formelle d'une implémentation de la pile protocolaire IKEv2, December 2016, Stéphanie Delaune, Romaric Maillard and Olivier Zendra
- PhD in progress: Lamine Nouredine (Rennes1); Automatic techniques for packing detection, classification and unpacking to stop malware propagation, November 2017, Stephane Ubeda, Olivier Zendra, Annelie Heuser.
- PhD in progress: Cassius de Oliveira Puodzius (Rennes1); Threat malware analysis, February 2019, Ludovic Me, Olivier Zendra, Annelie Heuser.
- PhD in progress: Duc Phuc Pham (Rennes1); Malware analysis through side-channel information, May 2019, Jean-Louis Lanet, Annelie Heuser, Olivier Zendra.

9.2.3. Juries

- Annelie Heuser was a referee for the PhD defense of Nisrine Jafri (University Rennes).
- Olivier Zendra was a member of the jury for the PhD defense of Delphine Beaulaton at University of Southern Brittany in Vannes.

10. Bibliography

Publications of the year

Doctoral Dissertations and Habilitation Theses

[1] F. DOLD. *The GNU Taler system : practical and provably secure electronic payments*, Université Rennes 1, February 2019, https://tel.archives-ouvertes.fr/tel-02138082

[2] N. JAFRI. *Formal fault injection vulnerability detection in binaries : a software process and hardware validation*, Université Rennes 1, March 2019, https://tel.archives-ouvertes.fr/tel-02385208

Articles in International Peer-Reviewed Journals

[3] J. KIM, S. PICEK, A. HEUSER, S. BHASIN, A. HANJALIC. *Make Some Noise Unleashing the Power of Convolutional Neural Networks for Profiled Side-channel Analysis*, in "IACR Transactions on Cryptographic Hardware and Embedded Systems", May 2019 [*DOI :* 10.13154/TCHES.V2019.I3.148-179], https://hal.inria.fr/hal-02010599

[4] S. PICEK, A. HEUSER, A. JOVIC, S. BHASIN, F. REGAZZONI. *The Curse of Class Imbalance and Conflicting Metrics with Machine Learning for Side-channel Evaluations*, in "IACR Transactions on Cryptographic Hardware and Embedded Systems", August 2019, vol. 2019, n° 1, pp. 1-29 [*DOI :* 10.13154/TCHES.V2019.I1.209-237], https://hal.inria.fr/hal-01935318

[5] D. B. ROY, S. BHASIN, S. GUILLEY, A. HEUSER, S. PATRANABIS, D. MUKHOPADHYAY. *CC Meets FIPS: A Hybrid Test Methodology for First Order Side Channel Analysis*, in "IEEE Transactions on Computers", March 2019, vol. 68, n° 3, pp. 347-361 [*DOI :* 10.1109/TC.2018.2875746], https://hal.inria.fr/hal-02413209

International Conferences with Proceedings

[6] M. BARBIER, A. RENZAGLIA, J. QUILBEUF, L. RUMMELHARD, A. PAIGWAR, C. LAUGIER, A. LEGAY, J. IBAÑEZ-GUZMÁN, O. SIMONIN. *Validation of Perception and Decision-Making Systems for Autonomous Driving via Statistical Model Checking*, in "IV 2019 - 30th IEEE Intelligent Vehicles Symposium", Paris, France, IEEE, June 2019, pp. 252-259 [*DOI :* 10.1109/IVS.2019.8813793], https://hal.inria.fr/hal-02127889

[7] P. BOUTILLIER, I. CRISTESCU, J. FERET. *Counters in Kappa: Semantics, Simulation, and Static Analysis*, in "ESOP 2019 - 28th European Symposium on Programming", Prague, Czech Republic, Springer, April 2019, pp. 176-204 [*DOI :* 10.1007/978-3-030-17184-1_7], https://hal.inria.fr/hal-02397876

[8] C. BROWN, A. D. BARWELL, Y. MARQUER, C. MINH, O. ZENDRA. *Type-Driven Verification of Non-functional Properties*, in "PPDP 2019 - 21st International Symposium on Principles and Practice of Declarative Programming", Porto, Portugal, ACM Press, October 2019, pp. 1-15 [*DOI :* 10.1145/3354166.3354171], https://hal.inria.fr/hal-02314723

[9] A. GONZALVEZ, R. LASHERMES. *A case against indirect jumps for secure programs*, in "SSPREW-9 2019 - 9th Software Security, Protection, and Reverse Engineering Workshop", San Juan, United States, SSPREW9 '19: Proceedings of the 9th Workshop on Software Security, Protection, and Reverse Engineering, December 2019, pp. 1-10, https://hal.archives-ouvertes.fr/hal-02382711

[10] T. NINET, A. LEGAY, R. MAILLARD, L.-M. TRAONOUEZ, O. ZENDRA. *Model Checking the IKEv2 Protocol Using Spin*, in "PST 2019 - 17th International Conference on Privacy, Security and Trust", Fredericton, Canada, August 2019, pp. 1-9, https://hal.inria.fr/hal-02062292

[11] T. NINET, A. LEGAY, R. MAILLARD, L.-M. TRAONOUEZ, O. ZENDRA. *The Deviation Attack: A Novel Denial-of-Service Attack Against IKEv2*, in "TrustCom 2019 - 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications", Rotorua, New Zealand, IEEE, August 2019, pp. 1-8, https://hal.inria.fr/hal-01980276

Conferences without Proceedings

[12] F. DÉCHELLE, B. LEBON, O. ZENDRA. *MASSE: Modular Automated Syntactic Signature Extraction*, in "RESSI 2019 - Rendez-vous de la Recherche et de l'Enseignement de la Sécurité des Systèmes d'Information", Erquy, France, May 2019, 1 p. , https://hal.inria.fr/hal-02159947

[13] C. GENEVEY-METAT, B. GÉRARD, A. HEUSER. *Combining sources of side-channel information*, in "C&ESAR 2019", Rennes, France, November 2019, https://hal.archives-ouvertes.fr/hal-02456646

Books or Proceedings Editing

[14] M. DURANTON, K. DE BOSSCHERE, B. COPPENS, C. GAMRAT, M. GRAY, H. MUNK, E. OZER, T. VARDANEGA, O. ZENDRA (editors). *The HiPEAC Vision 2019*, HiPEAC CSA, January 2019, 178 p. , https://hal.inria.fr/hal-02314184

Other Publications

[15] C. AUBERT, I. CRISTESCU. *History-Preserving Bisimulations on Reversible Calculus of Communicating Systems*, February 2019, https://arxiv.org/abs/1804.10355 - working paper or preprint, https://hal.archives-ouvertes.fr/hal-01778656

[16] S. PICEK, A. HEUSER, C. ALIPPI, F. REGAZZONI. *When Theory Meets Practice: A Framework for Robust Profiled Side-channel Analysis*, February 2019, working paper or preprint, https://hal.inria.fr/hal-02010603

[17] T. RICHMOND, A. HEUSER, B. GÉRARD. *Side-Channel Analysis of Post-Quantum Cryptography*, January 2019, 1 p. , SecDays 2019 - Security Days, Poster, https://hal.inria.fr/hal-02018859

References in notes

[18] A. SAVARY, M. FRAPPIER, M. LEUSCHEL, J. LANET. *Model-Based Robustness Testing in Event-B Using Mutation*, in "Software Engineering and Formal Methods - 13th International Conference, SEFM 2015, York, UK, September 7-11, 2015. Proceedings", R. CALINESCU, B. RUMPE (editors), Lecture Notes in Computer Science, Springer, 2015, vol. 9276, pp. 132–147, http://dx.doi.org/10.1007/978-3-319-22969-0_10