Activity Report 2018

Team TAMIS

Threat Analysis and Mitigation for Information Security

Joint team with Inria Rennes – Bretagne Atlantique

D4 – Language and Software Engineering

Table of contents

1. Team, Visitors, External Collaborators **1**
2. Overall Objectives **2**
2.1. Context 2
2.2. Approach and motivation 2
3. Research Program **3**
3.1. Axis 1: Vulnerability analysis 3
3.2. Axis 2: Malware analysis 4
3.3. Axis 3: Building a secure network stack 4
4. Application Domains **4**
4.1. System analysis 4
4.2. Cybersecurity 5
5. Highlights of the Year **5**
6. New Software and Platforms **5**
6.1. GNUnet 5
6.2. PLASMA Lab 6
6.3. Taler 6
6.4. SimFI 7
6.5. DaD 7
6.6. MASSE 7
6.7. BMA 8
6.8. PEPAC 8
6.9. Arml 8
6.10. IoTMLT 8
7. New Results **9**
7.1. Results for Axis 1: Vulnerability analysis 9
7.1.1. Statistical Model Checking of Incomplete Stochastic Systems 9
7.1.2. A Language for Analyzing Security of IOT Systems 9
7.1.3. Verification of IKEv2 protocol 10
7.1.4. Combining Software-based and Hardware-based Fault Injection Approaches 11
7.1.5. Side-channel analysis on post-quantum cryptography 11
7.1.6. New Advances on Side-channel Distinguishers 12
7.2. Results for Axis 2: Malware analysis 13
7.2.1. Malware Detection 14
7.2.2. Malware Deobfuscation 14
7.2.3. Malware Classification and clustering 15
7.2.4. Papers 17
7.3. Other research results 17
7.3.1. ContAv: a Tool to Assess Availability of Container-Based Systems 17
7.3.2. (Coordination of the) TeamPlay Project, and Expression of Security Properties 18
8. Bilateral Contracts and Grants with Industry **21**
8.1. Bilateral Contracts with Industry 21
8.2. Bilateral Grants with Industry 21
9. Partnerships and Cooperations **21**
9.1. National Initiatives 21
9.1.1. ANR 21
9.1.2. DGA 21
9.1.3. Autres 21
9.2. European Initiatives 22
9.2.1.1. ACANTO (028) 22

9.2.1.2. ENABLE-S3 (352) 22
9.2.1.3. TeamPlay (653) 24
9.2.1.4. SUCCESS 24
10. Dissemination **25**
10.1. Promoting Scientific Activities 25
10.1.1. Scientific Events Selection 25
10.1.1.1. Member of Conference Steering Committees 25
10.1.1.2. Chair of Conference Program Committees 25
10.1.1.3. Member of the Conference Program Committees 25
10.1.1.4. Reviewer 25
10.1.2. Journal 25
10.1.3. Scientific Expertise 25
10.2. Teaching - Supervision - Juries 26
10.2.1. Teaching 26
10.2.2. Supervision 26
10.2.3. Juries 26
11. Bibliography **26**

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

Axel Legay [Team leader until 12 Oct. 2018, Inria, Researcher, until 26 Nov 2018, HDR]
Olivier Zendra [Team leader since 12 Oct 2018, Inria, Researcher]
Annelie Heuser [CNRS, Researcher]
Jean-Louis Lanet [Inria, Senior Researcher, until Apr 2018, HDR]
Fabrizio Biondi [Centrale-Supelec, Researcher, "Chaire Malware"]
Kim Larsen [Inria, International Chair, Advanced Research Position]

Post-Doctoral Fellows

Najah Ben Said [Inria]
Eduard Baranov [Inria, from May 2018]
Ludovic Claudepierre [Inria, until Apr 2018]
Ioana Domnina Cristescu [Inria, from Feb 2018]
Yoann Marquer [Inria, from Jul 2018]
Stefano Sebastio [Inria, from Feb 2018]
Tania Richmond [Inria]

PhD Students

Sebanjila Bukasa [Inria, until Apr 2018]
Delphine Beaulaton [UBS Vannes]
Olivier Decourbe [Inria]
Florian Dold [Inria, until Oct 2018]
Christophe Genevey-Metat [Inria, from Oct 2018]
Alexandre Gonzalvez [IMT Atlantique]
Nisrine Jafri [Inria]
Ruta Moussaileb [IMT Atlantique, until Apr 2018]
Tristan Ninet [Thales]
Lamine Noureddine [Inria]
Leopold Ouairy [Inria, until Apr 2018]
Aurelien Palisse [Inria, until Apr 2018]
Emmanuel Tacheau [CISCO, until Sep 2018]
Alexander Zhdanov [Inria]

Technical staff

Jeffrey Paul Burdges [Inria, until Feb 2018]
Sébastien Campion [Inria]
Cassius de Oliveira Puodzius [Inria, from Feb 2018]
Thomas Given-Wilson [Inria]
Bruno Lebon [Inria]
Celine Minh [Inria, from May 2018]
Laurent Morin [Univ de Rennes I, until Sep 2018]
Jean Quilbeuf [Inria, until Sep 2018]
Louis-Marie Traonouez [Inria, until Jul 2018]

Interns

Philippe Charton [Inria, from Feb 2018 until Aug 2018]
Ilham Dami [Centrale-Supélec, from May 2018 until Aug 2018]
Felix Grunbauer [Inria, from Feb 2018 until Jun 2018]
Mickael Lebreton [Inria, from May 2018 until Aug 2018]
Dylan Marinho [Inria, from May 2018 until Jul 2018]

Administrative Assistant

Cecile Bouton [Inria]

Visiting Scientists

Shiraj Arora [PhD student from IIT Hyderabad, India, from Apr 2018 until Jun 2018]
Abdelhak Mesbah [PhD student from Université de Boumerdes, Algeria, Feb 2018]

External Collaborators

Francois-Renaud Escriva [DGA]
Sebastien Josse [DGA]
Colas Le Guernic [DGA]

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 [39].

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.

3.3. Axis 3: Building a secure network stack

Christian Grothoff, who leads this axis, got a position in Bern in 2017. This axis followed him, although TAMIS still held during 2018 expertise and members to finish ongoing work with the team.

4. Application Domains

4.1. System analysis

The work performed in Axes 1 and 2 and the methods developed there are applicable to the domain of system analysis, both wrt. program analysis and hardware analysis.

4.2. Cybersecurity

The work done in the axes above aims at improving cybersecurity, be it via vulnerability analyses, malware analyses and the development of safer networking mechanisms.

5. Highlights of the Year

5.1. Highlights of the Year

Change of team leader

Participants: Olivier Zendra, Axel Legay

Olivier Zendra was appointed team leader instead of Axel Legay on 12 Oct 2018.

"Chaire Analyse de Menaces" (Threat Analysis)

Participants: Fabrizio Biondi

Fabrizio Biondi resigned from Centrale Supelec and from the "Chaire Analyse de Menaces" (Threat Analysis) on 31 Dec 2018.

TeamPlay H2020 project, coordinated by Olivier Zendra

Participants: Olivier Zendra, Cécile Bouton, Yoann Marquer, Céline Minh, Tania Richmond

Launch on Jan 2018 of the TeamPlay (https://www.teamplay-h2020.eu) H2020 project (that had been submitted 25 April 2017), about the integration of nonfunctional properties in programs. TAMIS is in charge of security properties.

6. New Software and Platforms

6.1. GNUnet

KEYWORD: Distributed networks
SCIENTIFIC DESCRIPTION: The GNUnet project seeks to answer the question what a modern Internet architecture should look like for a society that care about security and privacy. We are considering all layers of the existing well-known Internet, but are also providing new and higher-level abstractions (such as voting protocols, Byzantine consensus, etc.) that are today solved in application-specific ways. Research questions include the desired functionality of the overall stack, protocol design for the various layers as well as implementation considerations, i.e. how to implement the design securely.
FUNCTIONAL DESCRIPTION: GNUnet is a framework for secure peer-to-peer networking that does not use any centralized or otherwise trusted services. Our high-level goal is to provide a strong free software foundation for a global network that provides security and in particular respects privacy.

GNUnet started with an idea for anonymous censorship-resistant file-sharing, but has grown to incorporate other applications as well as many generic building blocks for secure networking applications. In particular, GNUnet now includes the GNU Name System, a privacy-preserving, decentralized public key infrastructure.

- Participants: Alvaro Garcia Recuero, Florian Dold, Gabor Toth, Hans Grothoff, Jeffrey Paul Burdges and Sree Hrsha Totakura
- Partner: The GNU Project
- Contact: Sébastien Campion
- URL: https://gnunet.org/

6.2. PLASMA Lab

KEYWORDS: Energy - Statistics - Security - Runtime Analysis - Model Checker - Statistical - Model Checking - Aeronautics - Distributed systems
SCIENTIFIC DESCRIPTION: Statistical model checking (SMC) is a fast emerging technology for industrial scale verification and optimisation problems. SMC only requires an executable semantics and is not constrained by decidability. Therefore we can easily apply it to different modelling languages and logics. We have implemented in PLASMA Lab several advanced SMC algorithms that combine formal methods with statistical tests, which include techniques for rare events estimation and non-deterministic models. PLASMA Lab comes with a simulator plugin that allows to verify LLVM code.
FUNCTIONAL DESCRIPTION: PLASMA Lab is a compact, efficient and flexible platform for statistical model checking of stochastic models. PLASMA Lab includes simulators for PRISM models (Reactives Modules Language-RML) and Biological models. It also provides plugins that interface external simulators in order to support Matlab/Simulink, SytemC and LLVM . PLASMA Lab can be extended with new plugins to support other external simulators, and PLASMA Lab API can be used to embed the tool in other softwares. PLASMA Lab provide fast SMC algorithms, including advanced techniques for rare events simulation and nondeterministic models. These algorithms are designed in a distributed architecture to run large number of simulations on several computers, either on a local area network or grid. PLASMA Lab is implemented in Java with efficient data structures and low memory consumption.
NEWS OF THE YEAR: In 2018 Tania Richmond and Louis-Marie Traonouez have extended PLASMA Lab to propose statistical model checking analysis of discrete time Markov chains with unknown values (qDTMC). We have defined a new logic, called qBLTL, that extends the semantics of BLTL properties to take care of the unknown information in the path of the qDTMC. We have also adapted the model checking algorithm of probabilistic model checking of incomplete models to perform a three hypotheses test and provide bounds on the probability of errors of this test.

- Participants: Jean Quilbeuf, Louis-Marie Traonouez, Tania Richmond, Sean Sedwards, Benoît Boyer, Kevin Corre, Matthieu Simonin and Axel Legay
- Contact: Tania Richmond
- URL: https://project.inria.fr/plasma-lab/

6.3. Taler

GNU Taler
KEYWORD: Privacy
SCIENTIFIC DESCRIPTION: Taler is a Chaum-style digital payment system that enables anonymous payments while ensuring that entities that receive payments are auditable. In Taler, customers can never defraud anyone, merchants can only fail to deliver the merchandise to the customer, and payment service providers can be fully audited. All parties receive cryptographic evidence for all transactions, still, each party only receives the minimum information required to execute transactions. Enforcement of honest behavior is timely, and is at least as strict as with legacy credit card payment systems that do not provide for privacy.

The key technical contribution underpinning Taler is a new refresh protocol which allows fractional payments and refunds while maintaining untraceability of the customer and unlinkability of transactions. The refresh protocol combines an efficient cut-and-choose mechanism with a link step to ensure that refreshing is not abused for transactional payments.

We argue that Taler provides a secure digital currency for modern liberal societies as it is a flexible, libre and efficient protocol and adequately balances the state's need for monetary control with the citizen's needs for private economic activity.

FUNCTIONAL DESCRIPTION: Taler is a new electronic payment system. It includes an electronic wallet for customers, a payment backend for merchants and the main payment service provider logic called the exchange. Taler offers Chaum-style anonymous payments for citizens, and income-transparency for taxability.

- Participants: Florian Dold, Gabor Toth, Hans Grothoff, Jeffrey Paul Burdges and Marcello Stanisci
- Partner: The GNU Project
- Contact: Sébastien Campion
- URL: http://taler.net/

6.4. 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

6.5. DaD

Data-aware Defense
KEYWORD: Ransomware
FUNCTIONAL DESCRIPTION: DaD is a ransomware countermeasure based on a file system minifilter driver. It is a proof of concept and in its present condition cannot be used as a replacement of the existing antivirus solutions. DaD detects randomness of the data by monitoring the write operations on the file system. We monitor all the userland threads, and also the whole file system (i.e., not restricted to Documents). It blocks the threads that exceed a specific threshold. The malicious thread is not killed, we only block its next I/O operations.

- Participants: Aurélien Palisse and Jean-Louis Lanet
- Contact: Aurélien Palisse

6.6. 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

6.7. BMA

Behavioral Malware Analysis
KEYWORDS: Artificial intelligence - Malware - Automatic Learning - Concolic Execution
FUNCTIONAL DESCRIPTION: 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.

- Participants: Stefano Sebastio, Cassius De Oliveira Puodzius, Lamine Noureddine, Sébastien Campion, Jean Quilbeuf, Eduard Baranov and Thomas Given-Wilson
- Partner: Cisco
- Contact: Sébastien Campion
- URL: https://team.inria.fr/tamis/

6.8. PEPAC

PE PAcker Classifier. Version 1.4
KEYWORDS: Packer classification - Packer detection - Entropy - Machine learning - Feature selection - Portable Executable file - Obfuscation - Malware
FUNCTIONAL DESCRIPTION: This program takes a number of PE binary files and runs many packer detection and classification techniques on them, including YARA rules, PEiD rules, hash lists, and ML classifiers. The results are outputted to screen and dumped to disk on .json form.

This program is meant as a convenient way to compare the effectiveness of ML packer classifiers, but can also be used to detect and classify packing techniques in given binaries.

- Participants: Lamine Noureddine and Fabrizio Biondi
- Partner: Cisco
- Contact: Lamine Noureddine
- Publication: Effective, Efficient, and Robust Packing Detection and Classification

6.9. Arml

ARM to RML translator
KEYWORDS: Binary translation - ARM - RML
FUNCTIONAL DESCRIPTION: ArmL is an ARM to RML translator tool. ArmL tool takes as input an ARM executable binary, it produces as output a RML model.

- Contact: Nisrine Jafri

6.10. 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

7. New Results

7.1. Results for Axis 1: Vulnerability analysis

7.1.1. Statistical Model Checking of Incomplete Stochastic Systems

Participants: Tania Richmond, Louis-Marie Traonouez, Axel Legay.

We proposed a statistical analysis of stochastic systems with incomplete information. These incomplete systems are modelled using discrete time Markov chains with unknowns (qDTMC), and the required behaviour was formalized using qBLTL logic. By doing both quantitative and qualitative analysis of such systems using statistical model checking, we also proposed refinement on the qDTMCs. These refined qDTMCs depict a decrease in the probability of unknown behaviour in the system. The algorithms for both qualitative and quantitative analysis of qDTMC were implemented in the tool Plasma Lab. We demonstrated the working of these algorithms on a case study of a network with unknown information. We plan to extend this work to analyse the behaviour of other stochastic models like Markov decision processes and abstract Markov chains, with incomplete information.

This work has been accepted and presented to a conference this year [10].

[10] We study incomplete stochastic systems that are missing some parts of their design, or are lacking information about some components. It is interesting to get early analysis results of the requirements of these systems, in order to adequately refine their design. In previous works, models for incomplete systems are analysed using model checking techniques for three-valued temporal logics. In this paper, we propose statistical model checking algorithms for these logics. We illustrate our approach on a case-study of a network system that is refined after the analysis of early designs.

7.1.2. A Language for Analyzing Security of IOT Systems

Participants: Delphine Beaulaton, Najah Ben Said, Ioana Cristescu, Axel Legay, Jean Quilbeuf.

We propose a model-based security language of Internet of Things (IoT) systems that enables users to create models of their IoT systems and to make analysis of the likelihoods of cyber-attacks to occur and succeed. The modeling language describes the interactions between different entities, that can either be humans or “Things” (i.e, hardware, sensors, software tools, ..). A malicious entity is present in the system, called the Attacker, and it carries out attacks against the system. 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. For instance, an attacker can launch a phishing attack via email, only if it knows the email address of the target.

Another feature of our modeling language is that security failures are modeled as a sequence of simpler steps, in the spirit of *attack trees*. As their name suggests, attacks are modeled as trees, where the leaves represent elementary steps needed for the attack, and the root represents a successful attack. The internal nodes are of two types, indicating whether all the sub-goals (an AND node) or one of the sub-goals (an OR node) must be achieved in order to accomplish the main goal. The attack tree provided with the IoT system 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 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 also use a second SMC method, developed for *rare events*, called *importance splitting*.

To implement this we rely on *BIP*, a heterogeneous component-based model for which an execution engine is developed and maintained. The IoT model is translated into a BIP model and the attack tree into a BIP monitor. The two form a BIP system. The execution engine of BIP produce executions which are the input of Plasma Lab, the model checker developped in TAMIS. We have extended Plasma Lab with a plugin that interacts with the BIP execution engine.

The tools are available at http://iot-modeling.gforge.inria.fr/. This work has been published in two conference papers [20], [23]. A third paper was submitted in November [29], and is currently under review.

[20] In this paper we propose our security-based modeling language for IoT systems. The modeling language has two important features: (i) vulnerabilities are explicitly represented and (ii) interactions are allowed or denied based on the information stored on the IoT devices. An IoT system is transformed in BIP, a component-based modeling language, in which can execute the system and perform security analysis. To illustrate the features of our language, we model a use-case based on a Smart Hospital and inspired by industrial scenarios.

[23] In this paper we revisit the security-based modeling language for IoT systems. We focus here on the BIP models obtained from the original IoT systems. The BIP execution and analysis framework provides several methods to analyse a BIP model, and we discuss how these methods can be lifted on the original IoT systems. We also model a new use-case based on Amazon Smart Home.

[29] Attack trees are graphical representations of the different scenarios that can lead to a security failure. In this paper we extend our security-based framework for modeling IoT systems in two ways: (i) attack trees are defined alongside the model to detect and prevent security risks in the system and (ii) the language supports probabilistic models. A successful attack can be a *rare event* in the execution of a well designed system. When rare, such attacks are hard to detect with usual model checking techniques. Hence, we use *importance splitting* as a statistical model checking technique for rare events.

7.1.3. Verification of IKEv2 protocol

Participants: Tristan Ninet, Olivier Zendra, Louis-Marie Traonouez, Axel Legay.

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

7.1.4. Combining Software-based and Hardware-based Fault Injection Approaches

Participants: Nisrine Jafri, Annelie Heuser, Jean-Louis Lanet, Axel Legay, Thomas Given-Wilson.

Software-based and hardware-based approaches have both been used to detect fault injection vulnerabilities. Software-based approaches can provide broad and rapid coverage as it was shown in the previous publications [36], [37], [38] , but may not correlate with genuine hardware vulnerabilities. Hardware-based approaches are indisputable in their results, but rely upon expensive expert knowledge and manual testing.

This work bridges software-based and hardware-based fault injection vulnerability detection by contrasting results of both approaches. To our knowledge no research where done trying to bridge the software-based and hardware-based approach to detect fault injection vulnerabilities the way it is done in this work.

Using both the software-based and hardware-based approaches showed that:

- Software-based approaches detect genuine fault injection vulnerabilities.
- Software-based approaches yield false-positive results.
- Software-based approaches did *not* yield false-negative results.
- Not all software-based vulnerabilities can be reproduced in hardware.
- Hardware-based EMP approaches do *not* have a simple fault model.
- There is a coincidence between software-based and hardware-based approaches.
- Combining software-based and hardware-based approaches yields a vastly more efficient method to detect genuine fault injection vulnerabilities.

This work implemented both the SimFI tool and the ArmL tool.

7.1.5. Side-channel analysis on post-quantum cryptography

Participants: Annelie Heuser, Tania Richmond.

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. We first focus on code-based cryptography and then extend our analysis to find common vulnerabilities between different families of post-quantum crypto systems.

We started by a survey on cryptanalysis against code-based cryptography [13], that includes algebraic and side-channel attacks. Code-based cryptography reveals sensitive data mainly in the syndrome decoding. We investigate the syndrome computation from a side-channel point of view. There are different methods that can be used depending on the underlying code. We explore vulnerabilities of each one in order to propose a guideline for designers and developers. This work was presented at CryptArchi 2018 and Journées Codes et Cryptographie 2018.

[13] Nowadays public-key cryptography is based on number theory problems, such as computing the discrete logarithm on an elliptic curve or factoring big integers. Even though these problems are considered difficult to solve with the help of a classic computer, they can be solved in polynomial time on a quantum computer. Which is why the research community proposed alternative solutions that are quantum resistant. The process of finding adequate post-quantum cryptographic schemes has moved to the next level, right after NIST's announcement for post-quantum standardization.

One of the oldest quantum resistant proposition goes back to McEliece in 1978, who proposed a public-key cryptosystem based on coding theory. It benefits of really efficient algorithms as well as strong mathematical backgrounds. Nonetheless, its security has been challenged many times and several variants were cryptanalyzed. However, some versions are still unbroken.

In this paper, we propose to give a short background on coding theory in order to present some of the main flawless in the protocols. We analyze the existing side-channel attacks and give some recommendations on how to securely implement the most suitable variants. We also detail some structural attacks and potential drawback for new variants.

7.1.6. New Advances on Side-channel Distinguishers

Participants: Christophe Genevey Metat, Annelie Heuser, Tania Richmond.

[17] *On the Performance of Deep Learning for Side-channel Analysis* We answer the question whether convolutional neural networks are more suitable for SCA scenarios than some other machine learning techniques, and if yes, in what situations. Our results point that convolutional neural networks indeed outperforms machine learning in several scenarios when considering accuracy. Still, often there is no compelling reason to use such a complex technique. In fact, if comparing techniques without extra steps like preprocessing, we see an obvious advantage for convolutional neural networks only when the level of noise is small, and the number of measurements and features is high. The other tested settings show that simpler machine learning techniques, for a significantly lower computational cost, perform similar or even better. The experiments with the guessing entropy metric indicate that simpler methods like Random forest or XGBoost perform better than convolutional neural networks for the datasets we investigated. Finally, we conduct a small experiment that opens the question whether convolutional neural networks are actually the best choice in side-channel analysis context since there seems to be no advantage in preserving the topology of measurements.

[8] *The Curse of Class Imbalance and Conflicting Metrics with Machine Learning for Side-channel Evaluations* We concentrate on machine learning techniques used for profiled side-channel 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.

[35] *When Theory Meets Practice: A Framework for Robust Profiled Side-channel Analysis* Profiled side-channel attacks are the most powerful attacks and they consist of two steps. The adversary first

builds a leakage model, using a device similar to the target one, then it exploits this leakage model to extract the secret information from the victim's device. These attacks can be seen as a classification problem, where the adversary needs to decide to what class (corresponding to the secret key) the traces collected from the victim's devices belong to. For a number of years, the research community studied profiled attacks and proposed numerous improvements. Despite a large number of empirical works, a framework with strong theoretical foundations to address profiled side-channel attacks is still missing.

In this paper, we propose a framework capable of modeling and evaluating all profiled analysis attacks. This framework is based on the expectation estimation problem that has strong theoretical foundations. Next, we quantify the effects of perturbations injected at different points in our framework through robustness analysis where the perturbations represent sources of uncertainty associated with measurements, non-optimal classifiers, and methods. Finally, we experimentally validate our framework using publicly available traces, different classifiers, and performance metrics.

[33] *Make Some Noise: Unleashing the Power of Convolutional Neural Networks for Profiled Side-channel Analysis* Profiled side-channel attacks 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 that is able to reach high performance for a number of considered datasets. Additionally, for a dataset protected with the random delay countermeasure, our neural network is able to break the implementation by using only 2 traces in the attack phase. We compare our neural network with the one designed for a particular dataset with masking countermeasure and we show how both are good designs but also how 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 improve the number of measurement needed to reveal the secret key by orders of magnitude in certain scenarios for both neural networks. To strengthen our experimental results, we experiment with a number of datasets which differ in the levels of noise (and type of countermeasure) where we show the viability of our approaches.

[9] *On the optimality and practicability of mutual information analysis in some scenarios* The best possible side-channel attack maximizes the success rate and would correspond to a maximum likelihood (ML) distinguisher if the leakage probabilities were totally known or accurately estimated in a profiling phase. When profiling is unavailable, however, it is not clear whether Mutual Information Analysis (MIA), Correlation Power Analysis (CPA), or Linear Regression Analysis (LRA) would be the most successful in a given scenario. In this paper, we show that MIA coincides with the maximum likelihood expression when leakage probabilities are replaced by online estimated probabilities. Moreover, we show that the calculation of MIA is lighter that the computation of the maximum likelihood. We then exhibit two case-studies where MIA outperforms CPA. One case is when the leakage model is known but the noise is not Gaussian. The second case is when the leakage model is partially unknown and the noise is Gaussian. In the latter scenario MIA is more efficient than LRA of any order.

7.2. 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:

7.2.1. Malware Detection

Participants: Olivier Decourbe, Annelie Heuser, Jean-Louis Lanet, Olivier Zendra, Cassius Puodzius, Stefano Sebastio, Lamine Nourredine, Jean Quilbeuf, Eduard Baranov, Thomas Given-Wilson, Fabrizio Biondi, Axel Legay, Alexander Zhdanov.

Given a file or data stream, the malware detection problem consists of understanding if the file or data stream contain traces of malicious behavior. For binary executable files in particular, this requires extracting a signature of the file, so it can be compared against signatures of known clean and malicious files to determine whether the file is malicious. Binary file signatures can be divided in *syntactic* and *semantic*.

Syntactic signatures are based on properties of the file itself, like its length, hash, number and entropy of the executable and data sections, and so on. While syntactic signatures are computationally cheap to extract from binaries, it is also easy for malware creators to deploy *obfuscation* techniques that change the file's syntactic properties, hence widely mutating the signature and preventing its use for malware detection.

Semantic signatures instead are based on the binary's behavior and interactions with the system, hence are more effective at characterizing malicious files. However, they are more expensive to extract, requiring behavioral analysis and reverse-engineering of the binary. Since behavior is much harder to change than syntactic properties, against these signatures obfuscation is used to harden the file against reverse-engineering and preventing the analysis of the behavior, instead of changing it directly.

In both cases, *malware deofbuscation* is necessary to extract signatures containing actuable information that can be used to characterize the binaries as clean or malicious. Once the signatures are available, *malware classification* techniques, usually based on machine learning, are used to automatically determine whether binaries are clean or malicious starting from their signatures. Our contributions on these fields are described in the next sections.

7.2.2. Malware Deobfuscation

Participants: Olivier Decourbe, Lamine Nourredine, Annelie Heuser, Nisrine Jafri, Jean-Louis Lanet, Jean Quilbeuf, Axel Legay, Fabrizio Biondi.

Given a file (usually a portable executable binary or a document supporting script macros), deobfuscation refers to the preparation of the file for the purposes of further analysis. Obfuscation techniques are specifically developed by malware creators to hinder detection reverse engineering of malicious behavior. Some of these techniques include:

Packing Packing refers to the transformation of the malware code in a compressed version to be dynamically decompressed into memory and executed from there at runtime. Packing techniques are particularly effective against static analysis, since it is very difficult to determine statically the content of the unpacked memory to be executed, particularly if packing is used multiple times. The compressed code can also be encrypted, with the key being generated in a different part of the code and used by the unpacking procedure, or even transmitted remotely from a command and control (C&C) server.

- **1. Packing Detection and Classification**

  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.

– **2. Packing clustering** 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.

  This work is still in progress ...

- **Control Flow Flattening** This technique aims to hinder the reconstruction of the control flow of the malware. The malware's operation are divided into basic blocks, and a dispatcher function is created that calls the blocks in the correct order to execute the malicious behavior. Each block after its execution returns control to the dispatcher, so the control flow is flattened to two levels: the dispatcher above and all the basic blocks below.

  To prevent reverse engineering of the dispatcher, it is often implemented with a cryptographic hash function. A more advanced variant of this techniques embed a full virtual machine with a randomly generated instruction set, a virtual program counted, and a virtual stack in the code, and uses the machine's interpreter as the dispatcher.

  Virtualization is a very effective technique to prevent reverse engineering. To contrast it, we are implementing state-of-the-art devirtualization algorithms in `angr` , allowing it to detect and ignore the virtual machine code and retrieving the obfuscated program logic. Again, we plan to contribute our improvements to the main `angr` branch, thus helping the whole security community fighting virtualized malware.

- **Opaque Constants and Conditionals** Reversing packing and control flow flattening techniques requires understanding of the constants and conditionals in the program, hence many techniques are deployed to obfuscate them and make them unreadable by reverse engineering techniques. Such techniques are used e.g. to obfuscate the decryption keys of packed encrypted code and the conditionals in the control flow.

  We have proven the efficiency of dynamic synthesis in retrieving opaque constant and conditionals, compared to the state-of-the-art approach of using SMT (Satisfiability Modulo Theories) solvers, when the input space of the opaque function is small enough. We are developing techniques based on fragmenting and analyzing by brute force the input space of opaque conditionals, and SMT constraints in general, to be integrated in SMT solvers to improve their effectiveness.

7.2.3. Malware Classification and clustering

Participants: Annelie Heuser, Nisrine Jafri, Jean-Louis Lanet, Cassius Puodzius, Stefano Sebastio, Olivier Decourbe, Eduard Baranov, Jean Quilbeuf, Thomas Given-Wilson, Axel Legay, Fabrizio Biondi.

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.

Preliminary experiments show the promising results of our approach by considering the F-score in the classification of the malware families. Further investigation are needed in particular by using a larger dataset. For this purpose we established an academic collaboration with VirusTotal for helping us to build a ground truth for the family name.

One fundamental issue for supervised learning is the trustworthiness of the settled ground truth. In the scenario of malware classification, it is common to have great disagreement in the labeling of the very same malware sample (e.g. family attributed by different anti-malware vendors). Therefore, unsupervised learning on malware datasets by clustering based on the similarities of their SCDGs allows to overcome this problem.

We have put in place a platform for malware analysis, using dedicated hardware provided by Cisco. This platform is now fully operational and receives a daily feed of suspicious binaries for analysis. Furthermore, we developed tools for maintaining our datasets of cleanware and malware binaries, run existing syntactic analysis on them. Our toolchain is able to extract SCDGs from malwares and cleanwares and apply our classification techniques on the SCDGs.

7.2.4. Papers

This section gathers papers that are results common to all sections above pertaining to Axis 2.

- Efficient Extraction of Malware Signatures Through System Calls and Symbolic Execution: An Experience Report [28]

The ramping up use of network connected devices is providing hackers more incentives and opportunities to design and spread new security threats. Usually, malware analysts employ a mix of automated tools and human expertise to study the behavior of suspicious binaries and design suitable countermeasures. The analysis techniques adopted by automated tools include symbolic execution.Symbolic execution envisages the exploration of all the possible execution paths of the binary without neither concretizing the values of the variables nor dynamically executing the code (i.e., the binary is analyzed statically). Instead, all the values are represented symbolically. Progressing in the code exploration, constraints on symbolic variables are built and system calls tracked. A satisfiability-modulo-theory (SMT) checker is in charge of verifying the satisfiability of the collected symbolic constraints and thus the validity of an execution path. Unfortunately, while widely considered promising, this approach suffers from high resource consumption. Therefore, optimizing the constraint solver and tuning the features controlling symbolic execution is of fundamental importance to effectively adopting the technique. In this paper, we identify the metrics characterizing the quality of binary signatures expressed as system call dependency graphs extracted from a malware database. Then, we pinpoint some optimizations allowing to extract better binary signatures and thus to outperform the vanilla version of symbolic analysis tools in terms of malware classification and exploitation of the available resources.

7.3. Other research results

7.3.1. ContAv: a Tool to Assess Availability of Container-Based Systems

Participant: Stefano Sebastio.

This work was the result of a collaboration with former members of XRCI (Xerox Research Centre India): Rahul Ghosh, Avantika Gupta and Tridib Mukherjee.

[18] (C) The momentum gained by the microservice-oriented architecture is fostering the diffusion of operating system containers. Existing studies mainly focus on the performance of containerized services to demonstrate their low resource footprints. However, availability analysis of densely deployed container-based solutions is less visited due to difficulties in collecting failure artifacts. This is especially true when the containers are combined with virtual machines to achieve a higher security level. Inspired by Google's Kubernetes architecture, in this paper, we propose ContAv, an open-source distributed statistical model checker to assess availability of systems built on containers and virtual machines. The availability analysis is based on novel state-space and non-state-space models designed by us and that are automatically built and customized by the tool. By means of a graphical interface, ContAv allows domain experts to easily parameterize the system, to compare different configurations and to perform sensitivity analysis. Moreover, through a simple Java API, system architects can design and characterize the system behavior with a failure response and migration service.

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

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

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 $y = x; \texttt{if } b \texttt{ then } P_1(x); P_2(y) \texttt{ else } P_1(Y); P_2(x)$ 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 “Montgomerised" version with higher security level.

8. Bilateral Contracts and Grants with Industry

8.1. Bilateral Contracts with Industry

- CISCO (http://www.cisco.com) contract (2017–2022) to work on graph analysis of malware

8.2. 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

9. Partnerships and Cooperations

9.1. National Initiatives

9.1.1. 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 (42month),
- ANR JCJC CNRS.

9.1.2. DGA

- PhD grant for Nisrine Jafri (2016–2019),
- PhD grant for Aurélien Palisse (2016–2019),
- PhD grant for Alexandre Gonzalves (2016–2019),
- PhD grant for Olivier Decourbe (2017–2020),
- PhD grant for Alexandre Zdhanov (2017–2020)
- PhD grant for Christophe Genevey Metat (2019-2022)

9.1.3. Autres

- INS2I JCJC grant for Annelie Heuser

9.2. European Initiatives

9.2.1. FP7 & H2020 Projects

9.2.1.1. ACANTO (028)

Title: ACANTO: A CyberphusicAl social NeTwOrk using robot friends

Program: H2020

Duration: February 2015 - July 2018

Coordinator: Universita di Trento

Partners:

Atos Spain (Spain), Envitel Tecnologia Y Control S.A. (Spain), Foundation for Research and Technology Hellas (Greece), Servicio Madrileno Delud (Spain), Siemens Aktiengesellschaft Oesterreich (Austria), Telecom Italia S.P.A (Italy), Universita' Degli Studi di Siena (Italy), Universita Degli Studi di Trento (Italy), University of Northumbria At Newcastle. (United Kingdom)

Inria contact: Axel Legay

'Despite its recognised benefits, most older adults do not engage in a regular physical activity. The ACANTO project proposes a friendly robot walker (the FriWalk) that will abate a some of the most important barriers to this healthy behaviour. The FriWalk revisits the notion of robotic walking assistants and evolves it towards an activity vehicle. The execution of a programme of physical training is embedded within familiar and compelling every-day activities. The FriWalk operates as a personal trainer triggering the user actions and monitoring their impact on the physical and mental well-being. It offers cognitive and emotional support for navigation pinpointing risk situations in the environment and understanding the social context. It supports coordinated motion with other FriWalks for group activities. The FriWalk combines low cost and advanced features, thanks to its reliance on a cloud of services that increase its computing power and interconnect it to other assisted living devices. Very innovative is its ability to collect observations on the user preferred behaviours, which are consolidated in a user profile and used for recommendation of future activities. In this way, the FriWalk operates as a gateway toward a CyberPhysical Social Network (CPSN), which is an important contribution of the project. The CPSN is at the basis of a recommendation system in which users' profiles are created, combined into 'circles' and matched with the opportunity offered by the environment to generate recommendations for activities to be executed with the FriWalk support. The permanent connection between users and CPSN is secured by the FriPad, a tablet with a specifically designed user interface. The CPSN creates a community of users, relatives and therapists, who can enter prescriptions on the user and receive information on her/his state. Users are involved in a large number in all the phases of the system development and an extensive validation is carried out at the end.'

9.2.1.2. 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.

Olivier Zendra, Jean Quilbeuf, Jean-Louis Lanet and Axel Legay and were involved in this project. The project supports one postdoc in TAMIS starting in 2017.

9.2.1.3. 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.

9.2.1.4. 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 the TAMIS team, Delphine Beaulaton, Najah Ben Said, Ioana Cristescu, Axel Legay and Jean Quilbeuf are involved in this project.

10. Dissemination

10.1. Promoting Scientific Activities

10.1.1. Scientific Events Selection

10.1.1.1. 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)

10.1.1.2. Chair of Conference Program Committees

- Olivier Zendra was co-chair of the Program Committee and the Organizing Committee of the 13th Workshop on Implementation, Compilation, Optimization of Object-Oriented Languages, Programs and Systems (ICOOOLPS 2018)

10.1.1.3. Member of the Conference Program Committees

- Stefano Sebastio was a PC member of IEEE SOCA 2018 and ICORES 2019
- Annelie Heuser was PC member of TCHES 2018, CARDIS 2018, PROOFS 2018, KANGACRYPT 2018.

10.1.1.4. Reviewer

- Stefano Sebastio was a reviewer for ICORES 2019, IEEE SOCA 2018, CRiSIS 2018, COORDINATION 2018, MeTRiD satellite workshop of ETAPS 2018

10.1.2. Journal

10.1.2.1. Reviewer - Reviewing Activities

- Stefano Sebastio was a reviewer for EJOR (European Journal of Operational Research), OptimLett (Optimization Letters), JCST (Journal of Computer Science and Technology), IJCC (International Journal of Cloud Computing), IJDSN (International Journal of Distributed Sensor Networks)

10.1.3. Scientific Expertise

- 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 participated to a MCF recruiting committee for IUT de Vannes.
- Olivier Zendra is a member of the editorial board and co-author of the "HiPEAC 2019 Vision"
- Olivier Zendra is a member of Inria's evaluation committee.

10.2. Teaching - Supervision - Juries

10.2.1. Teaching

- Eduard Baranov: Master Méthodes d'analyse de risques, M2, Université de Bretagne Sud, France
- Tania Richmond: ENS Ker Lan.

10.2.2. Supervision

- PhD in progress: Christophe Genevey Metat (Rennes 1): , October 2018, Jean-Marc Jezequel, Benoit Gerard, Annelie Heuser and Clementine Maurice
- PhD in progress : Olivier Descourbe, On Code Obfuscation, October 2016, Axel Legay and Fabrizio Biondi.
- PhD in progress : Alexandre Gonsalvez, On Obfuscation via crypto primitives, April 2016, Axel Legay and Caroline Fontaine.
- PhD in progress : Nisrine Jafri (Rennes1), On fault Injection detection with MC of Binary code, December 2015, Axel Legay and Jean-Louis Lanet.
- PhD in progress : Routa Moussaileb, From Data Signature to Behavior Analysis, 2017, Nora Cuppens and Jean-Louis Lanet
- PhD in progress : Tristan Ninet (Rennes 1), Vérification formelle d'une implémentation de la pile protocolaire IKEv2, December 2016, Axel Legay, Romaric Maillard and Olivier Zendra
- PhD in progress: Lamine Nouredine (Rennes1); Developing new packing detection techniques to stop malware propagation, November 2017, Axel Legay and Annelie Heuser.
- PhD in progress : Aurélien Palisse, Observabilité de codes hostiles, 2015, Jean-Louis Lanet
- PhD in progress: Emmanuel Tacheau (Rennes1); Analyse et détection de malwares au moyen de méthodes d'analyse symbolique, September 2017, Axel Legay, Fabrizio Biondi, Alain Fiocco.
- PhD in progress : Aurélien Trulla, Caractérisation de malware Android par suivi de flux d'information et nouvelles techniques d'évasion, 2016, Valerie Viet Triem Tong and Jean-Louis Lanet
- PhD in progress: Alexander Zhdanov (Rennes 1): Modular Automated Syntactic Signature Extraction (MASSE), December 2017, Axel Legay, Fabrizio Biondi, François Déchelle and Olivier Zendra.

10.2.3. Juries

- Annelie Heuser was a referee for the PhD defense of Eleonora Cagli (CEA - Commissariat à l'Energie atomique et aux Energies alternatives, Grenoble)
- Annelie Heuser was a referee for the PhD defense of Damien Marion (Telecom ParisTech, CIFRE with Secure-IC)

11. Bibliography

Publications of the year

Articles in International Peer-Reviewed Journals

[1] F. BIONDI, M. A. ENESCU, T. GIVEN-WILSON, A. LEGAY, L. NOUREDDINE, V. VERMA. *Effective, Efficient, and Robust Packing Detection and Classification*, in "Computers and Security", 2018, pp. 1-15, https://hal.inria.fr/hal-01967597

[2] J. DUCHENE, C. LE GUERNIC, E. ALATA, V. NICOMETTE, M. KAÂNICHE. *State of the art of network protocol reverse engineering tools*, in "Journal of Computer Virology and Hacking Techniques", February 2018, vol. 14, n° 1, pp. 53-68 [*DOI :* 10.1007/s11416-016-0289-8], https://hal.inria.fr/hal-01496958

[3] J. L. FIADEIRO, A. LOPES, B. DELAHAYE, A. LEGAY. *Dynamic networks of heterogeneous timed machines*, in "Mathematical Structures in Computer Science", June 2018, vol. 28, n° 06, pp. 800 - 855 [*DOI :* 10.1017/S0960129517000135], https://hal.archives-ouvertes.fr/hal-01917079

[4] T. GIVEN-WILSON, A. HEUSER, N. JAFRI, A. LEGAY. *An automated and scalable formal process for detecting fault injection vulnerabilities in binaries*, in "Concurrency and Computation: Practice and Experience", September 2018, pp. 1-12 [*DOI :* 10.1002/CPE.4794], https://hal.inria.fr/hal-01960940

[5] T. GIVEN-WILSON, A. LEGAY. *On the Expressiveness of Joining and Splitting*, in "Journal in honour of Bernhard Steffen's 60th", November 2018, https://hal.inria.fr/hal-01955922

[6] T. GIVEN-WILSON, A. LEGAY, S. SEDWARDS, O. ZENDRA. *Group Abstraction for Assisted Navigation of Social Activities in Intelligent Environments*, in "Journal of Reliable Intelligent Environments", May 2018, vol. 4, n° 2, pp. 107–120 [*DOI :* 10.1007/s40860-018-0058-1], https://hal.inria.fr/hal-01629137

[7] A. NOURI, B. L. MEDIOUNI, M. BOZGA, J. COMBAZ, S. BENSALEM, A. LEGAY. *Performance Evaluation of Stochastic Real-Time Systems with the SBIP Framework*, in "International Journal of Critical Computer-Based Systems", 2018, pp. 1-33, https://hal.archives-ouvertes.fr/hal-01898426

[8] 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", November 2018, vol. 2019, n° 1, pp. 1-29 [*DOI :* 10.13154/TCHES.V2019.I1.209-237], https://hal.inria.fr/hal-01935318

[9] È. DE CHÈRISEY, S. GUILLEY, A. HEUSER, O. RIOUL. *On the optimality and practicability of mutual information analysis in some scenarios*, in "Cryptography and Communications - Discrete Structures, Boolean Functions and Sequences ", January 2018, vol. 10, n° 1, pp. 101 - 121 [*DOI :* 10.1007/s12095-017-0241-x], https://hal.inria.fr/hal-01935303

International Conferences with Proceedings

[10] S. ARORA, A. LEGAY, T. RICHMOND, L.-M. TRAONOUEZ. *Statistical Model Checking of Incomplete Stochastic Systems*, in "ISoLA 2018 - International Symposium on Leveraging Applications of Formal Methods", Limassol, Cyprus, LNCS, Springer, November 2018, vol. 11245, pp. 354-371 [*DOI :* 10.1007/978-3-030-03421-4_23], https://hal.inria.fr/hal-02011309

[11] F. BIONDI, T. GIVEN-WILSON, A. LEGAY, C. PUODZIUS, J. QUILBEUF. *Tutorial: an Overview of Malware Detection and Evasion Techniques*, in "ISoLA 2018 - 8th International Symposium On Leveraging Applications of Formal Methods, Verification and Validation", Limassol, Cyprus, October 2018, pp. 1-23, https://hal.inria.fr/hal-01964222

[12] S. K. BUKASA, R. LASHERMES, J.-L. LANET, A. LEGAY. *Let's shock our IoT's heart: ARMv7-M under (fault) attacks*, in "ARES 2018 - 13th International Conference on Availability, Reliability and Security", Hambourg, Germany, ACM Press, August 2018, pp. 1-6 [*DOI :* 10.1145/3230833.3230842], https://hal.inria.fr/hal-01950842

[13] V. DRAGOI, T. RICHMOND, D. BUCERZAN, A. LEGAY. *Survey on cryptanalysis of code-based cryptography: From theoretical to physical attacks*, in "ICCCC 2018 - 7th International Conference on Computers Communications and Control", Oradea, Romania, IEEE, May 2018, pp. 215-223 [*DOI :* 10.1109/ICCCC.2018.8390461], https://hal.inria.fr/hal-02011334

[14] K. DRIRA, F. OQUENDO, A. LEGAY, T. BATISTA. *Editorial Message Track on Software-intensive Systems-of-Systems (SiSoS) of the 33rd ACM/SIGAPP Symposium On Applied Computing (SAC 2018)*, in "SAC 2018 - The 33rd ACM/SIGAPP Symposium On Applied Computing", Pau, France, April 2018, pp. 1-3, https://hal.laas.fr/hal-01666389

[15] J. DUCHENE, E. ALATA, V. NICOMETTE, M. KAÂNICHE, C. LE GUERNIC. *Specification-Based Protocol Obfuscation*, in "DSN 2018 - 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks", Luxembourg City, Luxembourg, IEEE, June 2018, pp. 1-12, https://arxiv.org/abs/1807.09464 [*DOI :* 10.1109/DSN.2018.00056], https://hal.inria.fr/hal-01848573

[16] S. PICEK, A. HEUSER, A. JOVIC, K. KNEZEVIC, T. RICHMOND. *Improving Side-Channel Analysis through Semi-Supervised Learning*, in "17th Smart Card Research and Advanced Application Conference (CARDIS 2018)", Montpellier, France, November 2018, https://hal.inria.fr/hal-02011351

[17] S. PICEK, I. P. SAMIOTIS, A. HEUSER, J. KIM, S. BHASIN, A. LEGAY. *On the Performance of Convolutional Neural Networks for Side-channel Analysis*, in "SPACE 2018 - International Conference on Security, Privacy, and Applied Cryptography Engineering", Kanpur, India, LNCS, Springer, December 2018, vol. 11348, pp. 157-176, https://hal.inria.fr/hal-02010591

[18] S. SEBASTIO, R. GHOSH, A. GUPTA, T. MUKHERJEE. *ContAv: a Tool to Assess Availability of Container-Based Systems*, in "SOCA 2018 - 11th IEEE International Conference on Service Oriented Computing and Applications", Paris, France, November 2018, pp. 1-8, https://hal.inria.fr/hal-01954455

National Conferences with Proceedings

[19] C. LE GUERNIC, F. KHOURBIGA. *Taint-Based Return Oriented Programming*, in "SSTIC 2018 - Symposium sur la sécurité des technologies de l'information et des communications", Rennes, France, June 2018, pp. 1-30, https://hal.inria.fr/hal-01848575

Conferences without Proceedings

[20] D. BEAULATON, N. BEN SAID, I. CRISTESCU, R. FLEURQUIN, A. LEGAY, J. QUILBEUF, S. SADOU. *A Language for Analyzing Security of IOT Systems*, in "SoSE 2018 - 13th Annual Conference on System of Systems Engineering", Paris, France, IEEE, June 2018, pp. 37-44 [*DOI :* 10.1109/SYSOSE.2018.8428704], https://hal.inria.fr/hal-01960860

[21] B. L. MEDIOUNI, A. NOURI, M. BOZGA, M. DELLABANI, A. LEGAY, S. BENSALEM. *SBIP 2.0: Statistical Model Checking Stochastic Real-time Systems*, in "ATVA 2018 - 16th International Symposium Automated Technology for Verification and Analysis", Los Angeles, CA, United States, October 2018, pp. 1-6, https://hal.archives-ouvertes.fr/hal-01888538

[22] J. QUILBEUF, M. BARBIER, L. RUMMELHARD, C. LAUGIER, A. LEGAY, B. BAUDOUIN, T. GENEVOIS, J. IBAÑEZ-GUZMÁN, O. SIMONIN. *Statistical Model Checking Applied on Perception and Decision-making Systems for Autonomous Driving*, in "PPNIV 2018 - 10th Workshop on Planning, Perception and Navigation for Intelligent Vehicles", Madrid, Spain, October 2018, pp. 1-8, https://hal.inria.fr/hal-01888556

Scientific Books (or Scientific Book chapters)

[23] D. BEAULATON, I. CRISTESCU, A. LEGAY, J. QUILBEUF. *A Modeling Language for Security Threats of IoT Systems*, in "Formal Methods for Industrial Critical Systems - 23rd International Conference, FMICS 2018", LNCS, Springer, August 2018, vol. 11119, pp. 258-268 [*DOI :* 10.1007/978-3-030-00244-2_17], https://hal.inria.fr/hal-01962080

[24] T. GIVEN-WILSON, N. JAFRI, A. LEGAY. *The State of Fault Injection Vulnerability Detection*, in "Verification and Evaluation of Computer and Communication Systems", August 2018, pp. 3-21 [*DOI :* 10.1007/978-3-030-00359-3_1], https://hal.inria.fr/hal-01960915

Books or Proceedings Editing

[25] N. CUPPENS-BOULAHIA, F. CUPPENS, J.-L. LANET, A. LEGAY, J. GARCIA-ALFARO (editors). *Risks and security of internet and systems : 12th international conference, CRiSIS 2017, Dinard, France, September 19-21, 2017, revised selected papers*, Lecture Notes in Computer Science, Springer, 2018, vol. 10694, 269 p. , https://hal.archives-ouvertes.fr/hal-01865019

Scientific Popularization

[26] H. LE BOUDER, A. PALISSE. *Quand les malwares se mettent à la cryptographie*, in "Interstices", February 2018, https://hal.inria.fr/hal-01827607

Other Publications

[27] C. AUBERT, I. CRISTESCU. *History-Preserving Bisimulations on Reversible Calculus of Communicating Systems*, April 2018, https://arxiv.org/abs/1804.10355 - working paper or preprint, https://hal.archives-ouvertes.fr/hal-01778656

[28] E. BARANOV, F. BIONDI, O. DECOURBE, T. GIVEN-WILSON, A. LEGAY, C. PUODZIUS, J. QUILBEUF, S. SEBASTIO. *Efficient Extraction of Malware Signatures Through System Calls and Symbolic Execution: An Experience Report*, December 2018, working paper or preprint [*DOI :* 10.1145/NNNNNNN.NNNNNNN], https://hal.inria.fr/hal-01954483

[29] D. BEAULATON, N. BEN SAID, I. CRISTESCU, A. LEGAY, J. QUILBEUF. *Security Enforcement in IoT Systems using Attack Trees*, December 2018, working paper or preprint, https://hal.inria.fr/hal-01962089

[30] F. BIONDI, T. GIVEN-WILSON, A. LEGAY. *Universal Optimality of Apollonian Cell Encoders*, February 2018, working paper or preprint, https://hal.inria.fr/hal-01571226

[31] F. BIONDI, Y. KAWAMOTO, A. LEGAY, L.-M. TRAONOUEZ. *Hybrid Statistical Estimation of Mutual Information and its Application to Information Flow*, September 2018, working paper or preprint, https://hal.inria.fr/hal-01629033

[32] T. GIVEN-WILSON, N. JAFRI, A. LEGAY. *Bridging Software-Based and Hardware-Based Fault Injection Vulnerability Detection*, December 2018, working paper or preprint, https://hal.inria.fr/hal-01961008

[33] 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*, February 2019, working paper or preprint, https://hal.inria.fr/hal-02010599

[34] T. NINET, A. LEGAY, R. MAILLARD, L.-M. TRAONOUEZ, O. ZENDRA. *The Deviation Attack: A Novel Denial-of-Service Attack Against IKEv2*, 2018, working paper or preprint, https://hal.inria.fr/hal-01980276

[35] 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

References in notes

[36] T. GIVEN-WILSON, A. HEUSER, N. JAFRI, J.-L. LANET, A. LEGAY. *An Automated and Scalable Formal Process for Detecting Fault Injection Vulnerabilities in Binaries*, November 2017, working paper or preprint, https://hal.inria.fr/hal-01629135

[37] T. GIVEN-WILSON, N. JAFRI, J.-L. LANET, A. LEGAY. *An Automated Formal Process for Detecting Fault Injection Vulnerabilities in Binaries and Case Study on PRESENT – Extended Version*, April 2017, working paper or preprint, https://hal.inria.fr/hal-01400283

[38] T. GIVEN-WILSON, N. JAFRI, J.-L. LANET, A. LEGAY. *An Automated Formal Process for Detecting Fault Injection Vulnerabilities in Binaries and Case Study on PRESENT*, in "2017 IEEE Trustcom/BigDataSE/ICESS", Sydney, Australia, August 2017, pp. 293 - 300 [*DOI :* 10.1109/TRUSTCOM/BIGDATASE/ICESS.2017.250], https://hal.inria.fr/hal-01629098

[39] 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