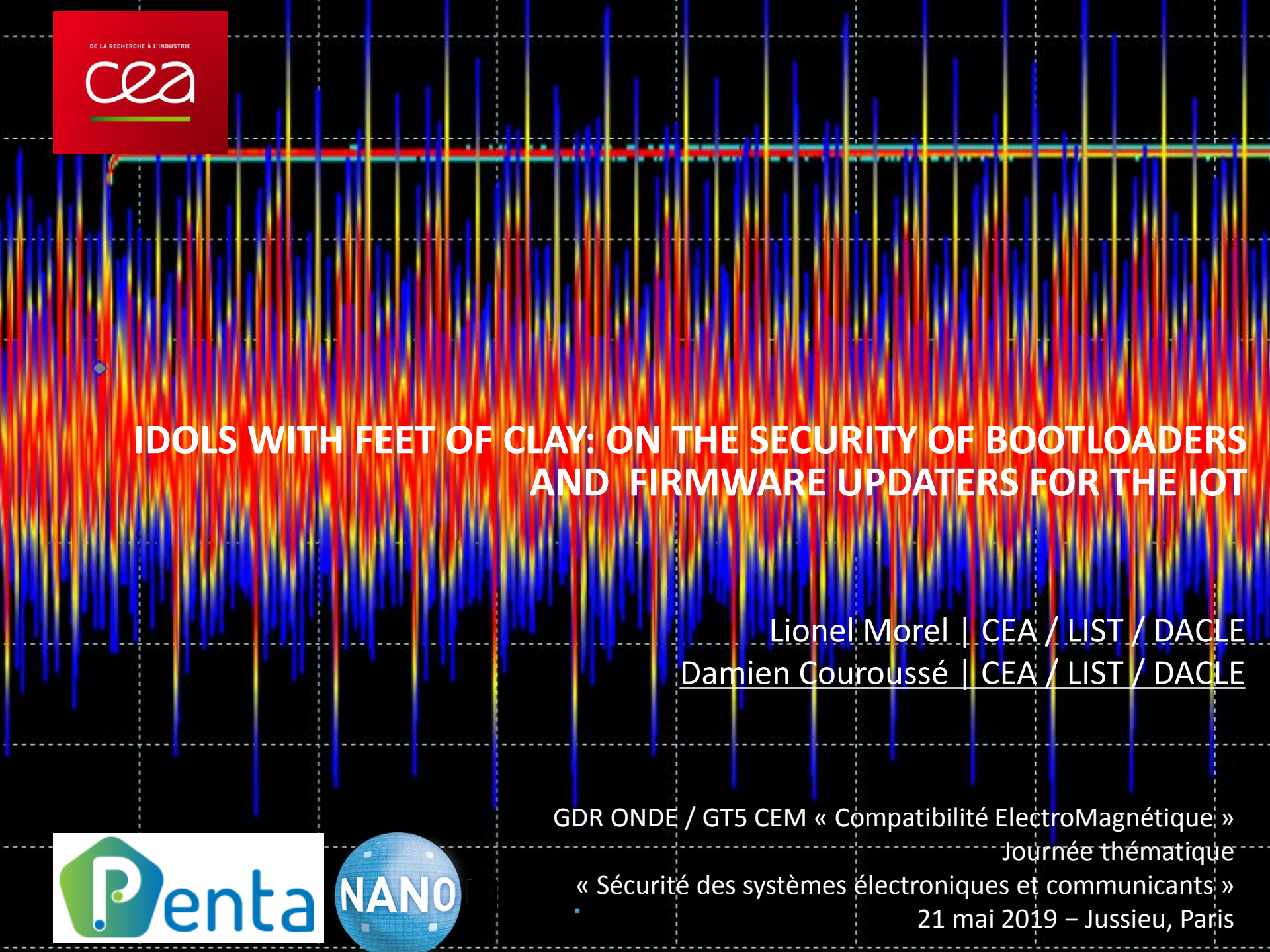


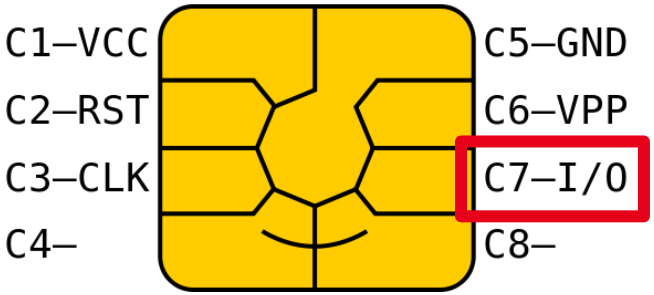
IDOLS WITH FEET OF CLAY: ON THE SECURITY OF BOOTLOADERS AND FIRMWARE UPDATERS FOR THE IOT

Lionel Morel | CEA / LIST / DACLE
Damien Couroussé | CEA / LIST / DACLE

GDR ONDE / GT5 CEM « Compatibilité ElectroMagnétique »
Journée thématique
« Sécurité des systèmes électroniques et communicants »
21 mai 2019 – Jussieu, Paris

Secure Element

- The HW and SW architecture is carried out by one main provider – and possibly a few sub-contractors
- Limited connectivity and communication capabilities
- Logical attacks are considered, but are not the main threat.
- **Considered secured**; security evaluated by expensive certification processes before market deployment
- **Impacts of a security breach: mostly limited** to the exploitation of the data stored in the component.



IoT device

- Integration of many HW and SW components, mostly issued by (untrusted) third parties.
- Lots of communication and sensing capabilities
- **Known to be unsecured**; lots of potential security vulnerabilities, certification is still an open topic and available schemes (e.g. CSPN) are not widely adopted.
- Impacts of a security breach:
  - Device level: usually low. On-device data have low value.
  - **Network/infrastructure level: high**. The device can be used as a stepping stone to attack other systems.
  - **Societal level: high**. Discredits the use of technology.

BFUs – BOOTLOADERS AND FIRMWARE UPDATERS

Bootloader ?

- Everything that comes between the system reset/startup and the startup of the 'User Application'.
- Also supports the capability to upgrade parts or all of its *firmware*
  - The bootloader component may not be included in this understanding of *firmware*
- Achille's heel of the whole system: if you control the boot process or the upgrade process, you control the ~~world~~ platform.

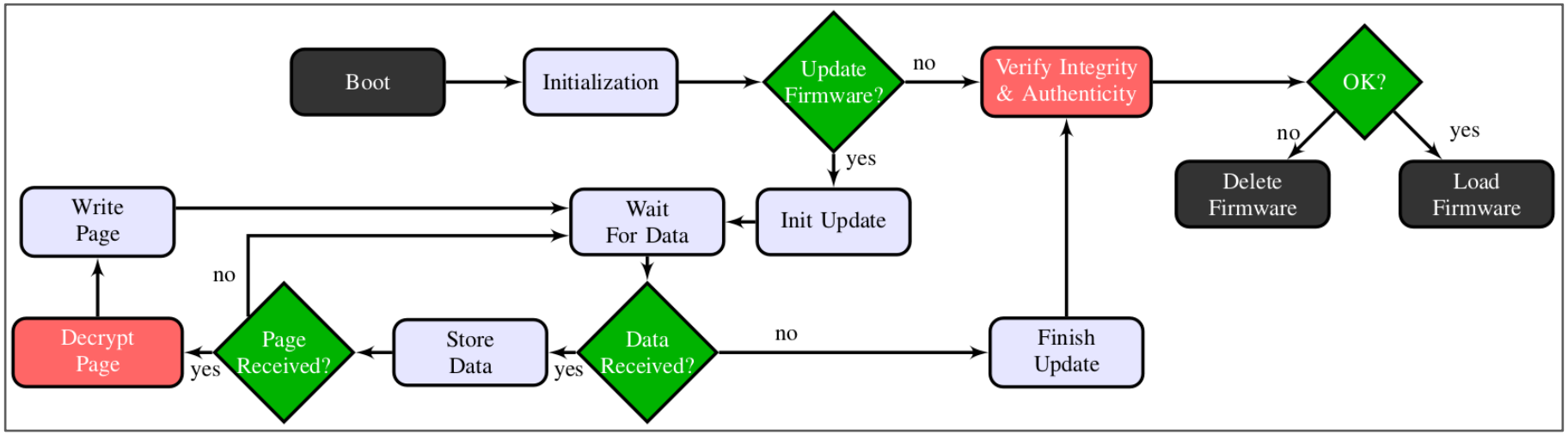
Security properties to support

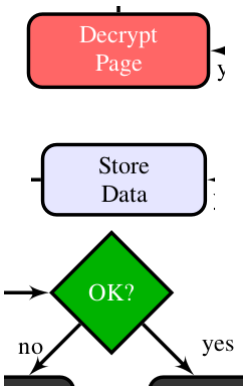
- **Confidentiality** → encryption functions, usually symmetric
- **Integrity**
  - Of the device → requires hardware support ("*anti tampering*")
  - Of the firmare → CRC, hash functions, MAC, digital signature
- **Authenticity** → MAC, digital signature

Our credo: BFUs provide a good case to study the security of Embedded/IoT systems

- Logical security: exploits of buffer overflows, ROPs, memory dumps, etc.
- Hardware security, mainly side-channel and fault-injection attacks, reverse engineering
- BFUs integrate cryptography
- But you can target all the glue code around the crypto components!
- A good case study to demonstrate the scalability of analysis tools

GENERIC ARCHITECTURE OF A BFU





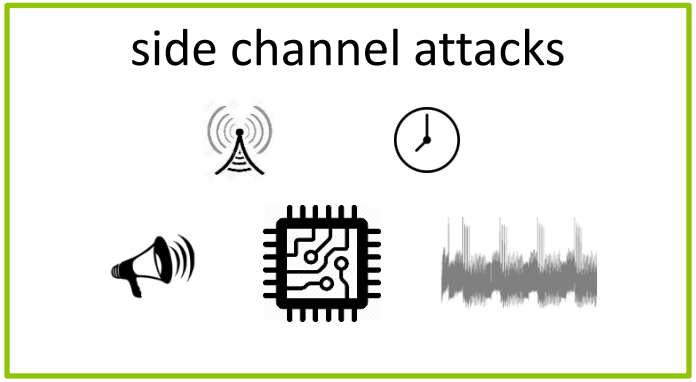
Cryptographic functions: implemented in SW, dedicated HW IPs, or SW+specific processor instructions

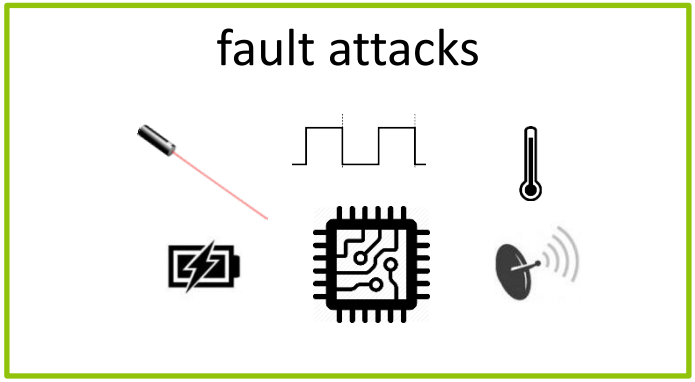
"System" components: implemented in SW, mostly HW-dependant and/or supported by dedicated HW (e.g. DMA for data movement)

Control logic: implemented in SW

One of the major threats against secure embedded systems

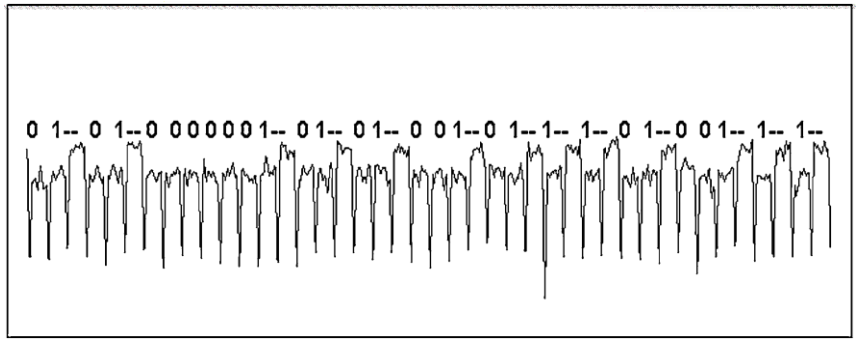
- The only effective classes of attacks against crypto-systems
- Relevant in many cases against cyber-physical systems: bootloaders, firmware upgrade, etc.





- In practice, logical attacks will be sufficient if the target is unprotected (e.g. typical IoT devices): buffer overflows, ROP, protocol vulnerabilities, read after free, etc.
- In practice, all high security products embed countermeasures against side-channel and fault injection attacks. E.g. Smart Cards, payTV, military-grade devices.
  - Using a combination of hardware *and* software countermeasures
- Side-channel and fault injection benches are getting really affordable

EXPLOITATION OF SIDE-CHANNEL INFORMATION LEAKAGE

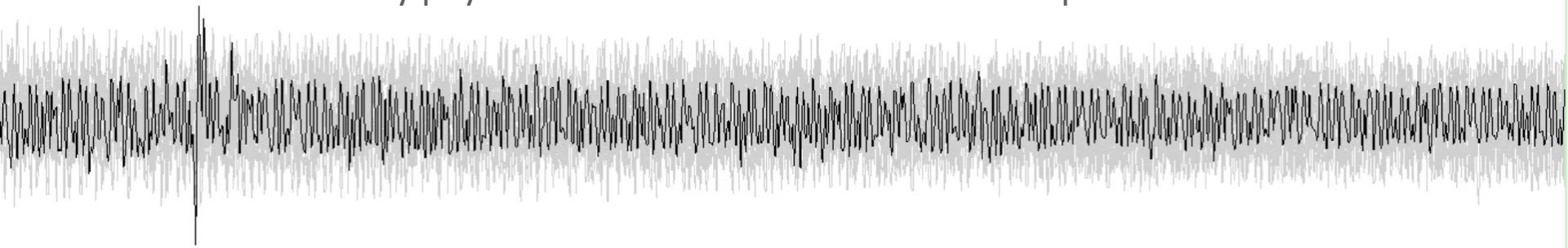


Simple power analysis (SPA)

SPA leaks from an RSA implementation

P. Kocher, J. Jaffe, B. Jun, and P. Rohatgi, 'Introduction to differential power analysis', Journal of Cryptographic Engineering, vol. 1, no. 1, pp. 5–27, 2011.

Correlation Power/EM Analysis (CPA/CEMA) – Can be generalised to any physical observation of the secured computation

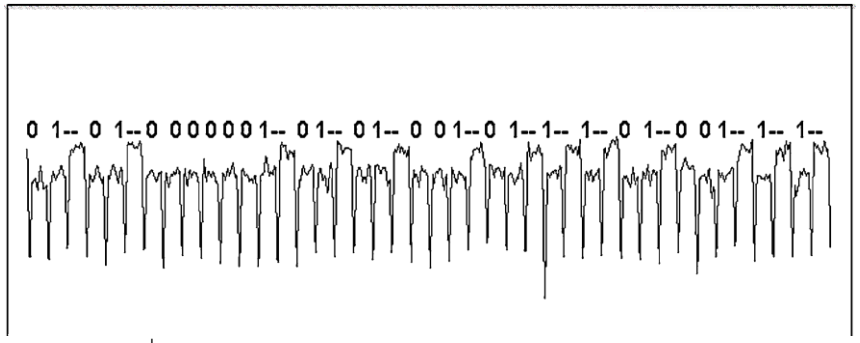


Key found!

- AES, unprotected implementation
- EM traces
- Attack on the output of the 1st SBOX

After the encryption of 4240 Bytes of data!

EXPLOITATION OF SIDE-CHANNEL INFORMATION LEAKAGE

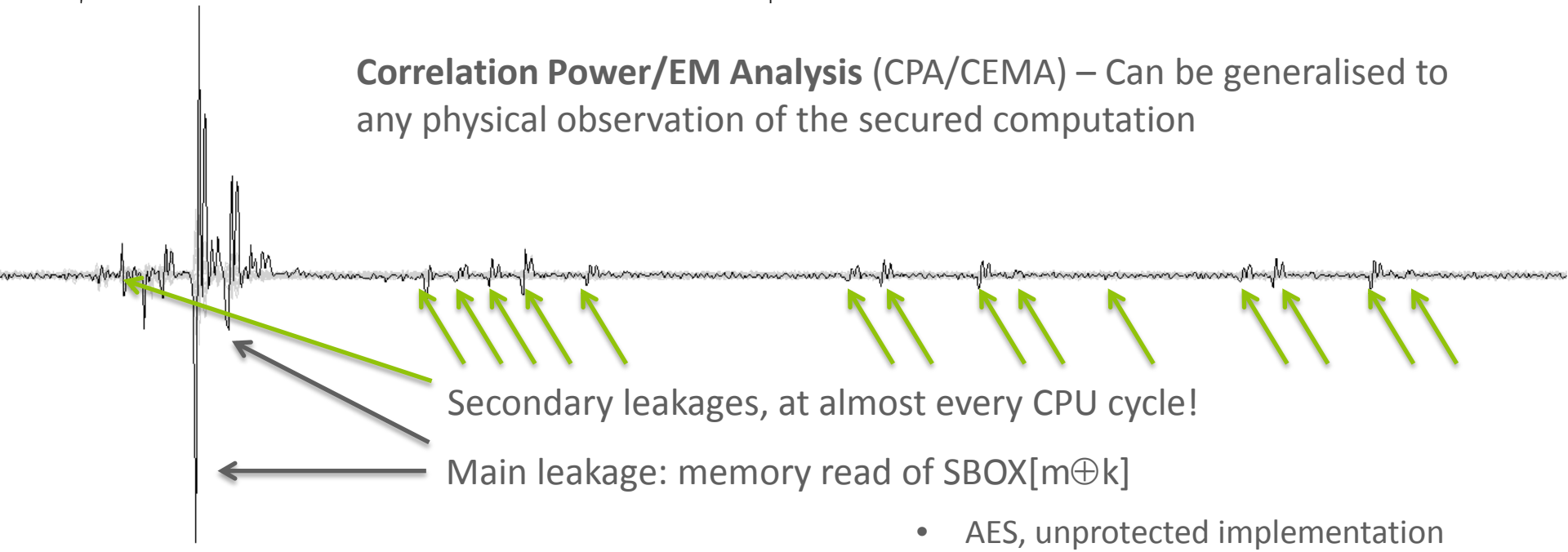


Simple power analysis (SPA)

SPA leaks from an RSA implementation

P. Kocher, J. Jaffe, B. Jun, and P. Rohatgi, 'Introduction to differential power analysis', Journal of Cryptographic Engineering, vol. 1, no. 1, pp. 5–27, 2011.

Correlation Power/EM Analysis (CPA/CEMA) – Can be generalised to any physical observation of the secured computation

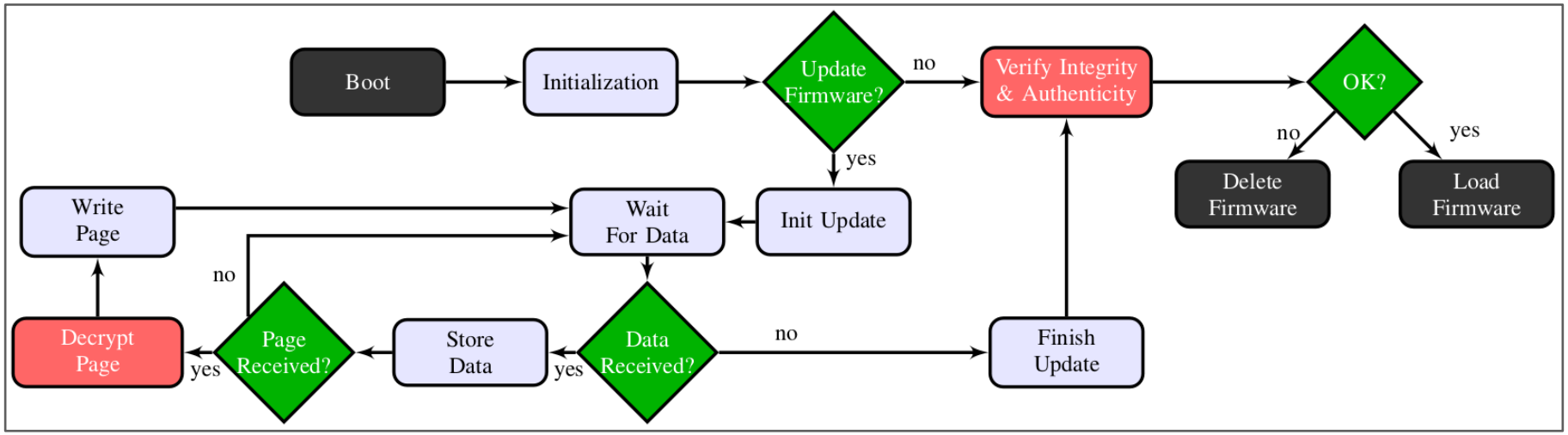


Secondary leakages, at almost every CPU cycle!

Main leakage: memory read of SBOX[m⊕k]

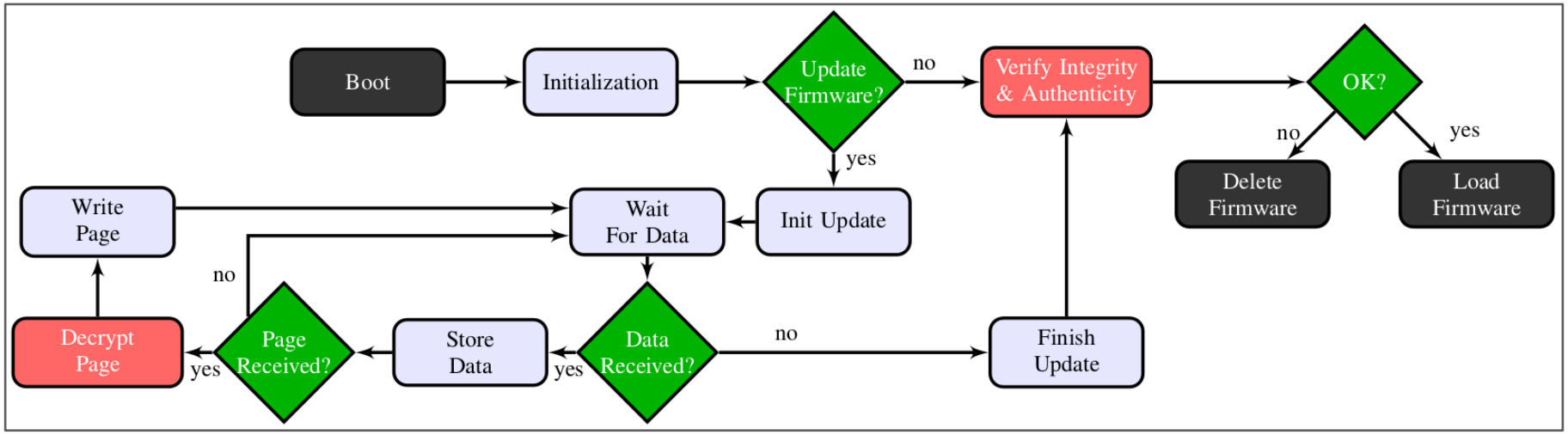
- AES, unprotected implementation
- EM traces
- Attack on the output of the 1st SBOX

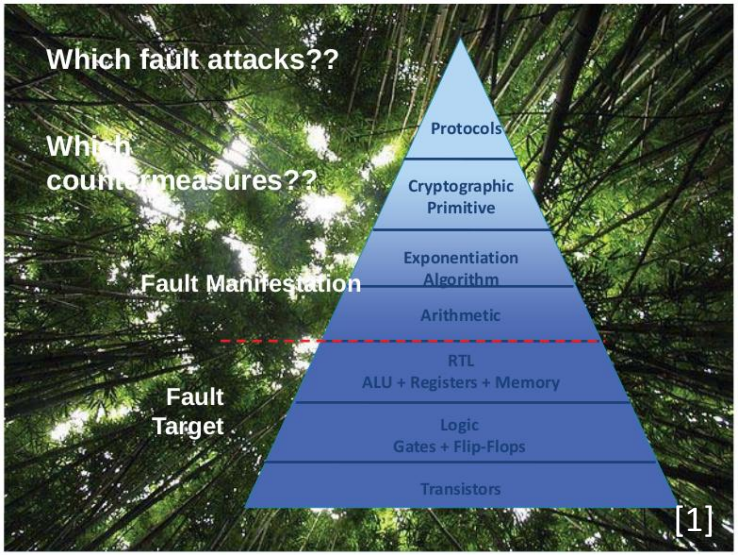
SIDE-CHANNEL ATTACKS: APPLICATION TO BFUS



1. Inspection of single side-channel traces
   - Reverse-engineering, e.g., identification of the program structure
2. CPA/CEMA
   - Recovery of secret data, e.g. cipher keys
   - Reverse-engineering of lookup tables

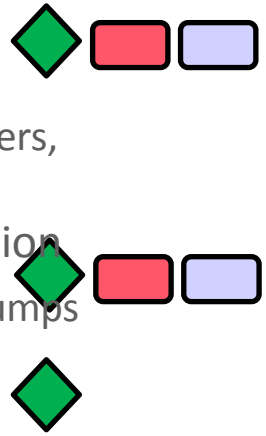
FAULT INJECTION ATTACKS: APPLICATION TO BFUS





Fault models, at the Instruction Set Architecture (ISA) level:

1. Data alteration, down to the bit level.
   - ROM / RAM, processor registers
   - Bit flip, bit stuck-at
   - Typically: modification of loop counters, crypto data, opcode corruption.
2. Instruction skip, instruction modification
   - Typically: NOP execution, arbitrary jumps
3. Modification of the control flow, e.g., test inversion



[1] I. Polian, M. Joye, I. Verbauwhede, M. Witteman, and J. Heyszl, 'Controlled fault injection: wishful thinking, thoughtful engineering or just luck ?', FDTC, 2017.

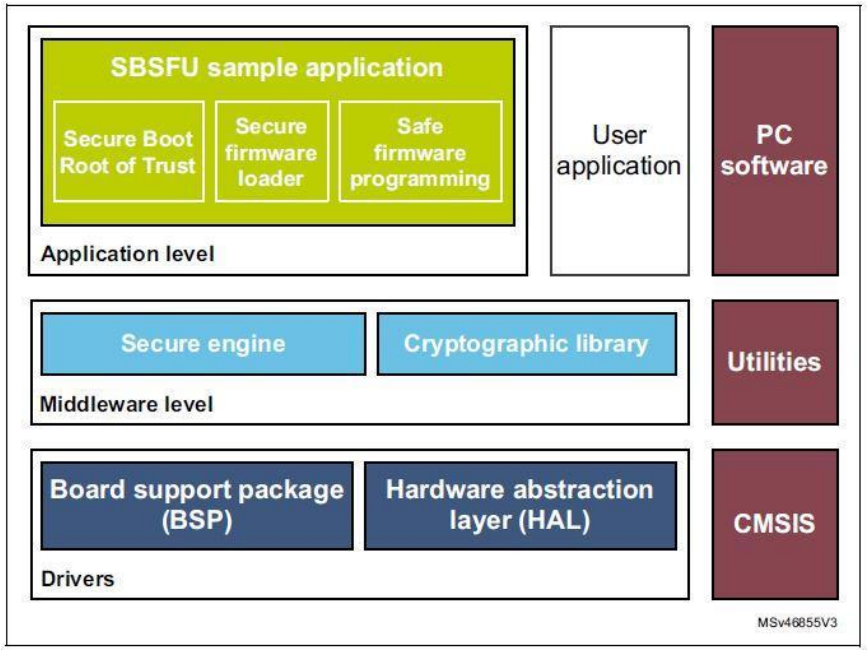
SECURED BOOTLOADERS

IoT security: 2 types of product families

1. Integrates AES-256 → clearly not enough
2. Secured bootloaders → Atmel, MicroChip, STMicroelectronics, etc.

Secured bootloader: provides a secured Chain-of-Trust (CoT) encompassing a full boot sequence.

- The SW boot component can be considered as part of the product. Immutable in memory, usually not upgradable.
- Provides a Secure Enclave
  - Secured storage with limited capacity. Usually only for a few encryption keys.
  - Secured execution context with limited processing capability
  - Strong isolation from the User / Application execution domain.
- Only the User/Client app is upgradable
- Securing the User / Application execution domain is still up to the application developer



Example: X-CUBE-SBSFU

Many interesting public initiatives in the RISC-V ecosystem.

e.g. wolfBoot
https://www.wolfssl.com/products/wolfboot

IRT NANOELEC / PULSE – PROJET CLAPS

- WP1 – Développement d'outils pour **l'analyse de robustesse** en présence d'attaques par injection de fautes,
- WP2 – Développement d'**outils** pour **l'application automatisée de contre-mesures**,
- WP3 – Mise en œuvre de mécanismes **d'identification de comportements sous attaque** dans le but de prendre des mesures préventives ou correctives,
- WP4 – Confrontation d'une analyse de sécurité à une caractérisation de sécurité en grandeur nature sur un **banc d'attaque**,
- WP5 – Démonstration du passage à l'échelle des outils et des moyens de caractérisation de sécurité sur un **cas d'usage applicatif**

  ➔ **Bootloader/FU**

JOURNÉE THÉMATIQUE
SUR LES ATTAQUES PAR INJECTION DE FAUTES

23 mai 2019 – Minatec, Grenoble – https://jaif2019.github.io

Session #1. Injection de fautes
- **Injection de fautes par médium EM : modèle et implications -** Philippe Maurine (LIRMM)
- **On-the-fly laser-induced corruption of the firmware stored into the flash memory of a 32-bit microcontroller -** Brice Colombier (Univ. Saint-Étienne)
- **How modern System-on-Chips are vulnerable to fault attacks -** Ronan Lashermes (INRIA) and Thomas Trouchkine (ANSSI)

Session #2. Architectures matérielles robustes
- **Analyse de fautes au niveau RTL -** Vincent Beroulle (LCIS Valence)
- **IntrinSec: an intrinsically secure RISC V processor -** Olivier Savry (CEA)

Session #3. Questions ouvertes sur la sécurité des systèmes
- **Fault attacks: What practical exploits on IoT? -** Éric Vétillard (NXP)
- **Concevoir des applications robustes à l'injection de fautes (projet CLAPs) -** Laurent Mounier et Marie-Laure Potet (VERIMAG)

Session #4. Protections logicielles
- **Compilation de contre-mesures -** François de Ferrière (STMicroelectronics)
- **Sécurisation automatisée des boucles à la compilation -** Julien Proy (INVIA)

Session #5. Analyse de code
- **Techniques d'analyse statique pour détecter des vulnérabilités sécuritaires lors d'une revue de code -** David Féliot (CEA Grenoble)
- **Évaluation sécuritaire de code binaire soumis à des attaques en faute -** Jean-Baptiste Bréjon (LIP6)

IDOLS WITH FEET OF CLAY: ON THE SECURITY OF BOOTLOADERS AND FIRMWARE UPDATERS FOR THE IOT

Lionel Morel | CEA / LIST / DACLE
Damien Couroussé | CEA / LIST / DACLE

damien.courousse@cea.fr

leti
Centre de Grenoble
17 rue des Martyrs
38054 Grenoble Cedex

list
Centre de Saclay
Nano-Innov PC 172
91191 Gif sur Yvette Cedex