Hardware Security Assessment Methodology Applied on an Embedded Application: A Case Study on a Medical IoT

Zahra Kazemi, David Hély, Vincent Beroulle

LCIS Laboratory

Univ. Grenoble Alpes, Grenoble INP, F-26000 Valence, France

Abstract

- The wide usage and high accessibility of embedded IoTs have created major concerns for the manufacturers and enterprises in the hardware security domain. Embedded software developers, often lack the knowledge to consider the hardware-based threats and their effects on the important assets. To overcome such challenges, we develop an easy to use and low cost hardware security assessment framework, against major physical attacks (e.g. clock, voltage, or EM fault injection attacks). It can assist the software developers to detect their system vulnerabilities and to protect the important assets. This work can also guide on implementing software-level countermeasures, which can reduce the effects of the physical attack's risks to an acceptable level. As a case study, we apply our approach on an IoT medical application named "SecPump" that models an infusion pump in the hospitals. This study mimics a real experimental evaluation process and highlights the potential risks of ignoring the physical attacks.

Introduction

- Various physical attacks such as **Side-Channel Analysis** (SCA) and **Fault Injection Attacks** (FIAs) can impose a severe security risk to an IoT system. Below is an overview of the main physical attacks:

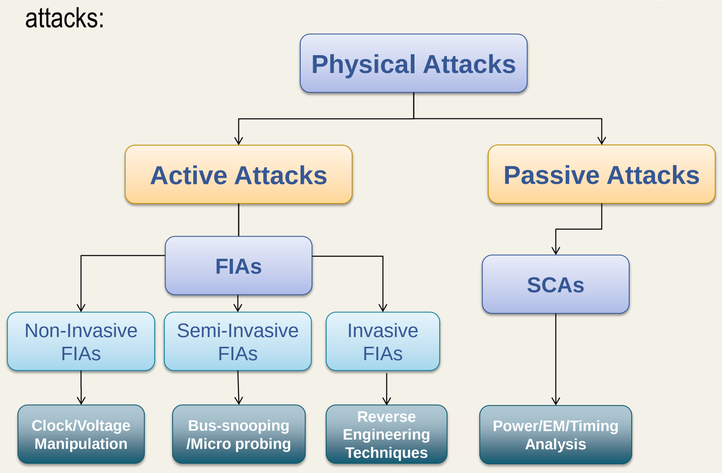

Figure.1 Taxonomy of different physical attacks against embedded systems

- We only consider the non-invasive attacks in our platform that do not require expensive equipment and can be performed by only accessible interfaces.
- Our evaluation platform includes power SCA and non-invasive FIAs and reveals their effect as a key step to secure IoT devices.

Methodology

- Mostly an embedded software is composed of different modules and therefore the hardware security assessment of the entire application is a puzzling task without applying a proper strategy.
- We propose a systematic approach for hardware security assessment of an IoT application that includes different sequential steps:

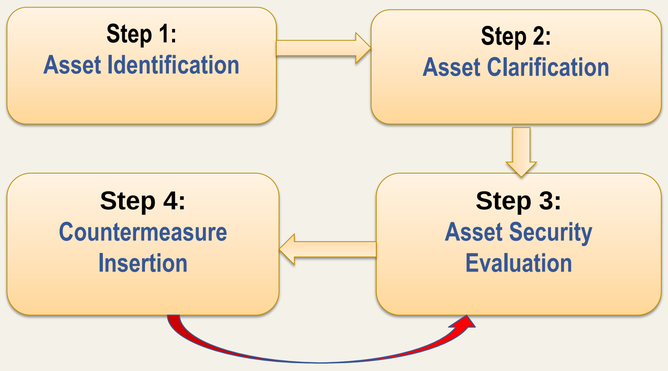

Figure.2. Software Security Evaluation Flow[1]

- We identify the important assets and the potential vulnerability of the underlying application that can affect the credential information. This can be the data/control flow integrity and the availability of the critical services when targeted by the SCAs and FIAs.
- Corresponding to each asset, an evaluation scenario with a predefined set of parameters is applied.

Analysis

- We developed our open hardware evaluation platform "HackmyMCU" (Fig.3) focusing on practical non-invasive attacks:

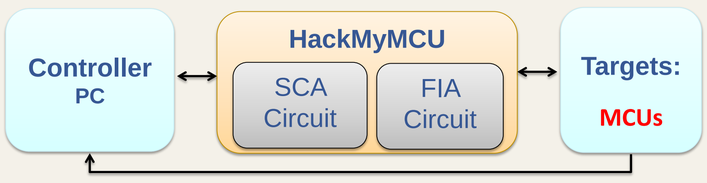

Figure.3 HackMyMCU Platform [2,3]

Fig.4 presents our framework that can evaluate a running software on a MCU-based system against fault attack vulnerabilities. It has the following characteristics:

- Low cost and easy to use platform
- High precision for the software security evaluation
- Configuration interface to adjust the fault parameters
- Analyzer interface to generate a report

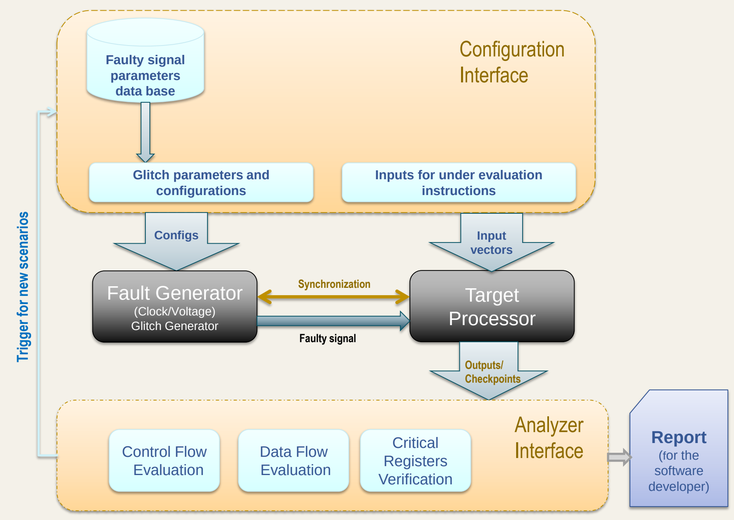

Figure.4 Framework of a Clock and Voltage Fault Injection Platform [4]

- In the analyser interface, we categorize the evaluation of the software commands into two groups:

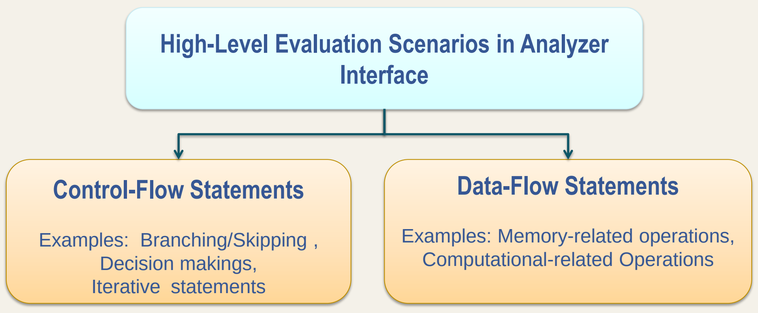

Figure 5. Evaluation approaches in the Analyzer Interface

Case Study

- To make an experimental hardware security assessment, we apply our approach on a medical IoT device named "SecPump" (Fig.6).
- These infusion pumps are widely used in hospitals to deliver doses of drugs to the patients and also to monitor their health status

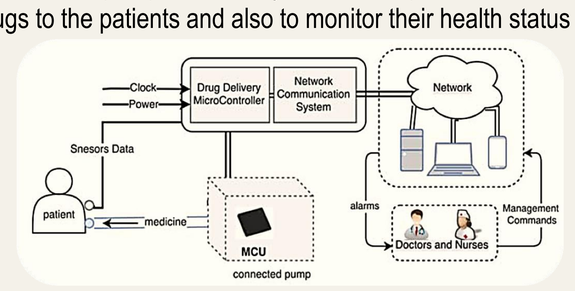

Figure.6. SecPump physical architecture

- **Step 1:** Fig. 7 shows the assortment of the critical assets for the reliable and proper functioning of the SecPump.

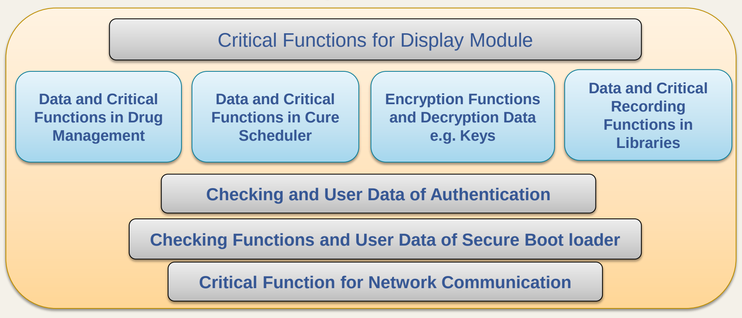

Figure.7. Critical assets in SecPump [1]

- **Step 2:** Table 1 presents the different asset categories of the SecPump device.

Table 1. Classification of important assets in the SecPump

Category	Module	Assets Examples	Threat
Confidentiality-related assets	Authentication	Username and passwords	SCA
	Encryption	Encryption keys	FIA - SCA
Integrity-related assets	Boot loader	The Program Counter register	FIA
	Authentication	Control flow of password checking instruction	FIA
	Clock Scheduler and Drug Management	Data flow in setting the values (registers in memory)	FIA
	Encryption	Control flow in sequential instructions	FIA
Availability-related assets	Network	Essential functions to maintain the connection with the network	FIA
	Display	All the necessary functions	FIA

- **Step 3:** Utilizing our hardware platform, we evaluate the SecPump application against SCA and FIA (some results next section).
- **Step 4** : Adding proper software-level countermeasures:
  - Bit-slicing countermeasure against SCA and FIA for the Encryption Module [6].
  - A new compiler-based countermeasure named "BackFlow" to avoid the manipulation of the control flows [7].

Results

- Here, we provide some examples of the discovered vulnerabilities against clock glitching FIA and Power SCA.
- . **Evaluation of the Encryption Module:** Fig. 8, shows a CPA attack by performing high-side (VDD) power measurements, using the implemented evaluation platform.

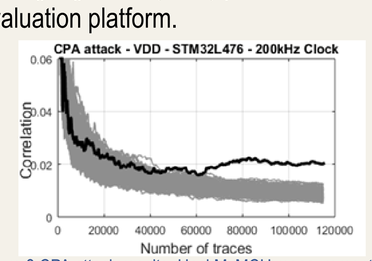

Figure 8.CPA attack results, HackMyMCU measurementsV [2]

Fig 9 shows the impact of fault injection on the AES algorithm in the SecPump application.

- On the vertical axis, we show the number of affected bytes of the state register of the AES (-1 shows reset/hang of the MCU).
- The horizontal axis contains all 410 clock cycles of the AES operation under attack.

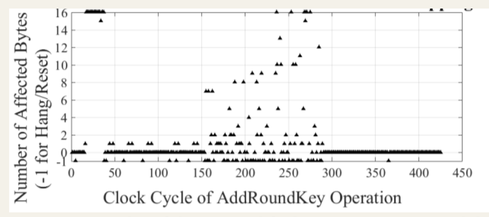

Figure 9. cartography of the injected faults on Encryption module [3]

- **Evaluation of the Authentication Module:**

Our goal was to evaluate single and double password authentications under FI and to compare them.

- Single Step Authentication:

Fig.10, presents an evaluation example of single conditional branch statements by first setting the condition to a state which leads to a known result, and then we insert CP1 and CP2 and monitor the consequence.

Fig.11 , the CP1 shows the glitch width and glitch delay that cause the password checking step to be skipped.

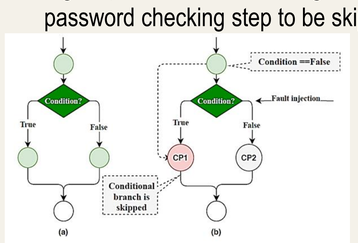

Figure 10. Control Flow Evaluation: Single Conditional Branch [5]

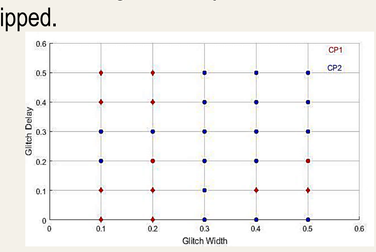

Figure 11.. The Glitch map for single step authentication [5]

- Double Step Authentication:

Fig. 12 and 13 depict the combination of parameters that cause CP2 to be set which shows a successful attack.

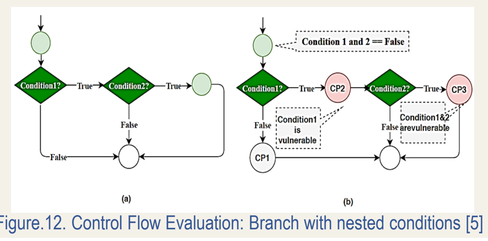

Figure.12. Control Flow Evaluation: Branch with nested conditions [5]

Figure 13. The Glitch Map of Nested Decision Making (a) Single Glitch Induction, (b) Double Glitch Induction after One Clock Cycle, (c)) Double Glitch Induction after Two Clock Cycle [5]

Conclusion and Future Work

In this work, we developed a practical strategy to evaluate an embedded application against non-invasive hardware attacks. Then, we analyzed the impacts of the injected faults in a critical medical application to identify the potential risks. To advance our study, we will extend our work as following:

- Improving the configurator interface in order to cover more attack scenarios
- Improving the analyzer interface in order to properly inspect the fault effects

The advanced interface can result in finding the most effective fault injection setup, which aids on properly evaluating any high level instruction.

References

1. Zahra Kazemi, Cyril Bresch, Mahdi Fazeli, David Hely, and Vincent Beroulle; " A Systematic Approach for Hardware Security Assessment of Secured IoT Applications". Trudevice2020, DateConference, Grenoble,France, March 2020.
2. Z. Kazemi, A. Papadimitriou, D. Hely, M. Fazeli, and V. Beroulle, "Hardware Security Evaluation Platform for MCU - based Connected Devices : Application to healthcare IoT," IVSW:3nd Int. Verif. Secur. Work.,CostaBrava,Spain, July 1-4, 2018.
3. Zarha Kazemi, Athanasios Papadimitriou, Ehsan Aerabi, Mosabbah Mushir Ahmed, David Hely and Vincent Beroulle, "A low Cost and Practical Fault Injection Framework for Security Assessment of the Cyber-Physical Systems" IVSW:4rd Int. Verif. Secur. Work.,Rhodos Greece, July 1-4, 2019.
4. Kazemi, Z.; Hely, D.; Fazeli, M.; Beroulle, V. A Review on Evaluation and Configuration of Fault Injection Attack Instruments to Design Attack Resistant MCU-Based IoT Applications. Electronics 2020, 9, 1153.
5. Zahra Kazemi, Mahdi Fazeli, David Hely, and Vincent Beroulle; "Hardware Security Vulnerability Assessment to Identify the Potential Risks in A Critical Embedded Application", 26th IEEE International Symposium on On- Line Testing and Robust System Design, Napoli, Italy, July 12-15, 2020.
6. E. Aerabi, A. Papadimitriou, and D. Hely, "On a Side Channel and Fault Attack Concurrent Countermeasure Methodology for MCU-based Byte-sliced Cipher Implementations," 2019 IEEE 25th International Symposium on On-Line Testing and Robust System Design, IOLTS 2019, pp. 103–108, 2019, doi: 10.1109/IOLTS.2019.8854372.
7. C. Bresch, R. Lysecky, and D. Hely, "BackFlow: Backward Edge Control Flow Enforcement for Low End ARM Microcontrollers," in 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, Mar. 2020, pp. 1606–1609, doi: 10.23919/DATE48585.2020.9116396.

Acknowledgements

This work is carried out under the SERENE-IoT project, a project labeled within the framework of PENTA, the EUREKA Cluster for Application and Technology Research in Europe on NanoElectronics.