Reverse Engineering a Vulnerable RFID Reader

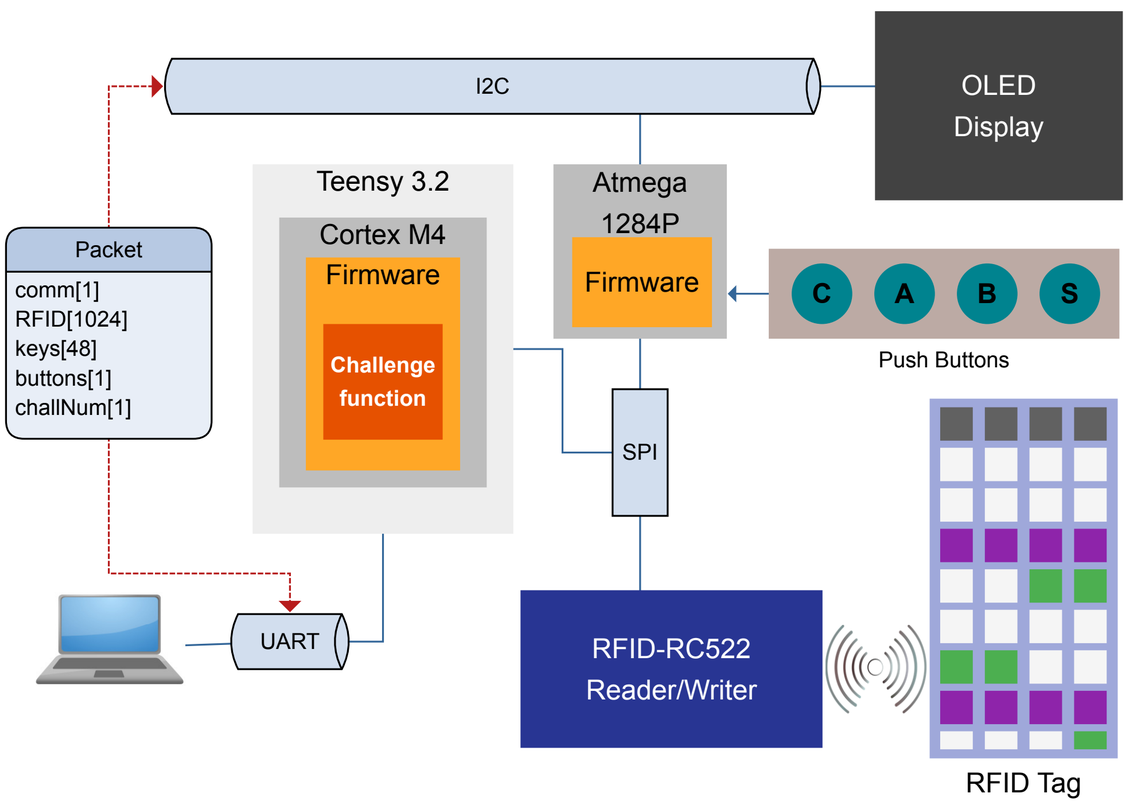
Romain Brenaget, Jérôme Blanchard, Mathieu Dolmen and Pierre Fontaine
University of Southern Brittany - Master Cybersecurity of Embedded Systems

Context

- We participated to the CSAW'19 Embedded Security Challenge.
- 5 weeks to reverse engineer firmwares of an RFID access control system.
- Hardware provided by the competition organizers.
- Challenges released periodically.

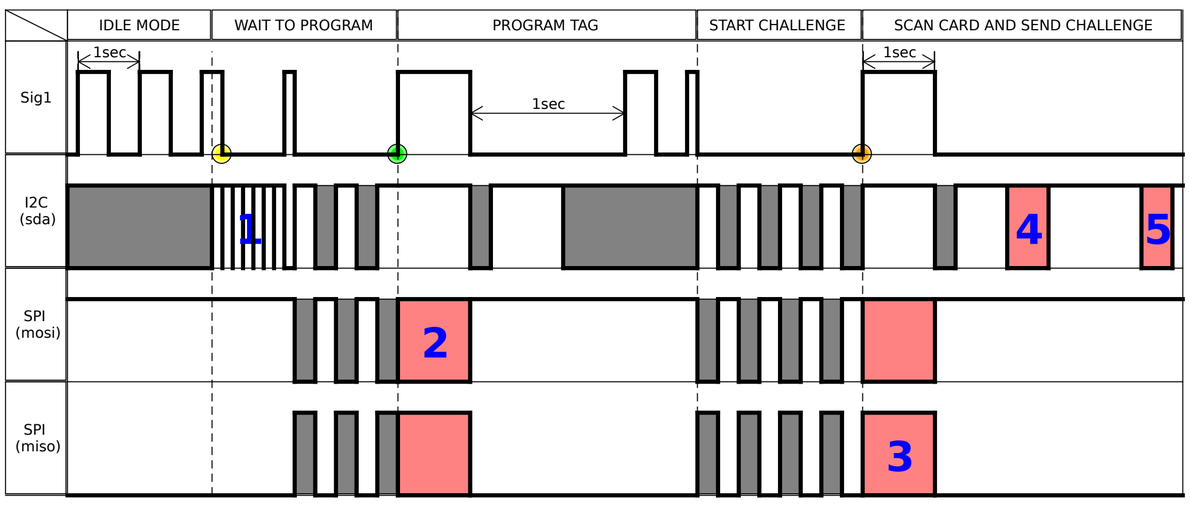
Board Architecture

- RFID Tag
  - MIFARE Classic EV1
  - EEPROM Memory (1024 bytes)
  - 16 Sectors of 4 blocs
  - **1st** block for manufacturer
- Atmega 1284P
  - 8-bit AVR (RISC-based)
  - Manage external devices
  - Connections through I2C and SPI
- Teensy 3.2
  - ARM Cortex-M4 (32-bit)
  - Runs compiled C++
  - Implements Interruption Sub-Routines



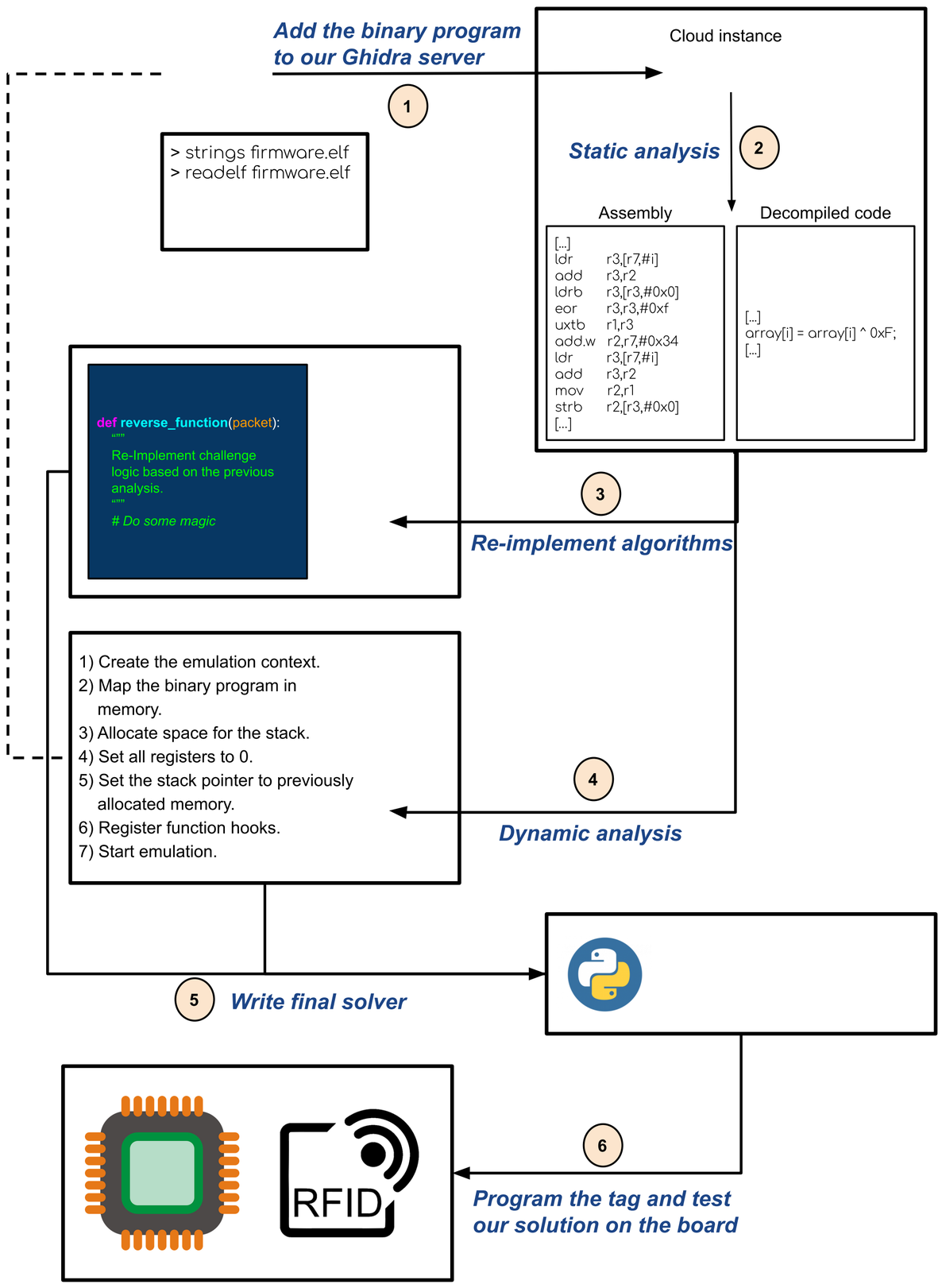
Hardware Level Analysis of a Challenge

start "sender.py" script
scan the tag to program
scan the tag for a given challenge



1. Send data tag to program from Teensy to AVR via I2C bus.
2. AVR sends tag data to RFID writer via SPI bus.
3. When a tag is scanned for a challenge, tag data are sent to AVR via SPI.
4. AVR sends tag data to Teensy via I2C.
5. After performing the challenge, Teensy sends a passphrase to AVR via I2C (the AVR is in charge of challenge validation, depending on the passphrase).

Reverse Engineering Workflow



Conclusion

- Learned a lot about embedded system architecture, ARM assembly and serial communication.
- Discovered new tools.
- Solved all the challenges.

« Mais, n'oubliez pas que ... »

ici, on est plutôt White Hat

BRETAGNE PASSEZ À L'OUEST
#passezalouest

CSAW'19