ARCHIE: A QEMU-Based Framework for Architecture-Independent Evaluation of Faults

Florian Hauschild, Kathrin Garb, Lukas Auer, Bodo Selmke, Johannes Obermaier

September 17, 2021

Outline

Just Another Fault Tool ...?

ARCHIE

Experimental Verification

Summary

Just Another Fault Tool ...?

Just Another Fault Tool ...?

Fault Analysis in Practice

Laser fault injection into embedded devices [1]

Task: Find "fault candidates", test binary

However:

- No source code available
- Binary varies depending on compiler settings

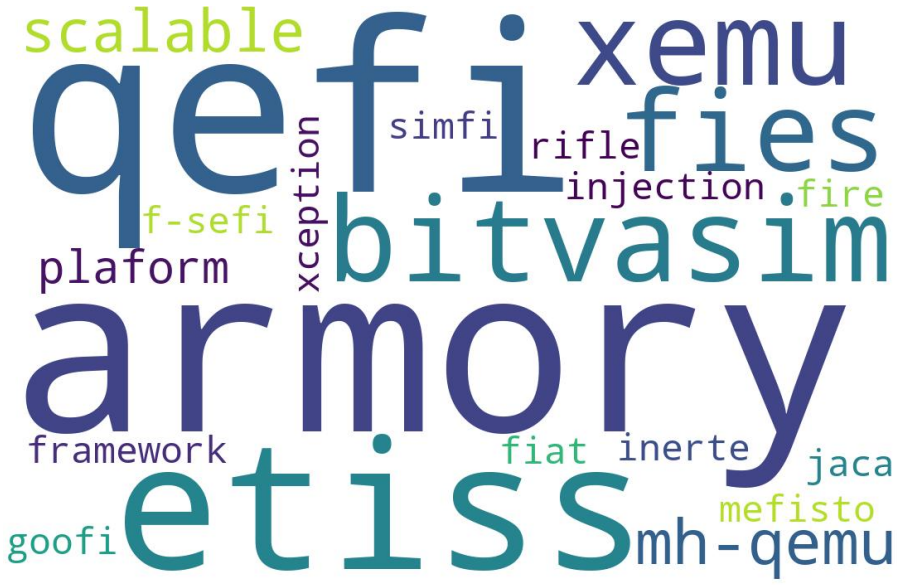
⇒ Manual examination of binary needed

Question: Is there an easier way to do this?

[1] K. Garb, J. Obermaier: Temporary Laser Fault Injection into Flash Memory. IEEE IOLTS 2020

Just Another Fault Tool ...?

The Zoo of Fault Simulation Tools



However:

- Architecture-dependent
- Only specific fault types and models
- Source code mostly non-available
- Old QEMU version

Just Another Fault Tool ...?

Requirements for a Fault Tool

- Support of different architectures
- Support of different fault types
  - Transient and permanent faults
  - Instruction and data faults
  - In registers, RAM, and flash memory
- Automated, parallelized execution
- Open source

$\Rightarrow$ ARCHitecture-Independent Evaluation (ARCHIE) of faults

ARCHIE

ARCHIE

QEMU

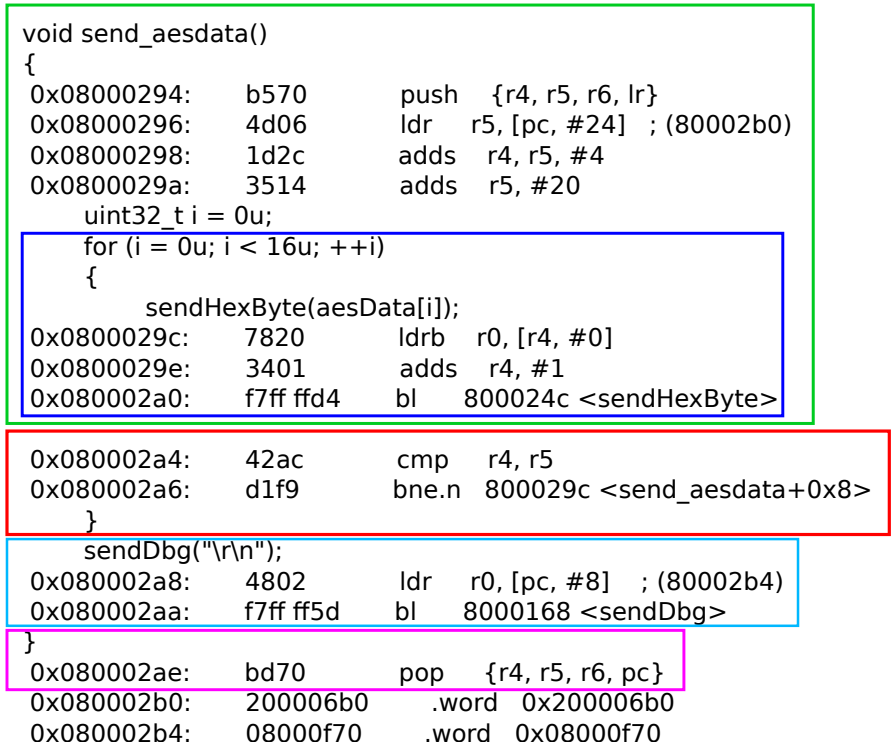
- Generic and open source machine emulator and virtualizer
- Quick execution time
- Support of 22 different architectures, e.g.,
  - ARM
  - RISC-V
  - x86_64
- Easy to add new machines

⇒ Fulfills requirements

ARCHIE

Tiny Code Generator (TCG)

- Execution of guest instructions
- Division of guest instructions into basic blocks
- Translation of basic block into host instructions



ARCHIE

TCG Plugin Interface

TCG plugin interface (since QEMU 4.2, November 2019)
⇒ API for monitoring TCG and read access to low-level guest instructions

Fault injection plugin:

- Minimal extension of TCG plugin interface
- Read and write to guest memory and registers
- Logging of guest behavior

ARCHIE

Experiment, Campaign, Golden Run

Fault
Low-level alteration of a physical state, e.g., one or multiple bit flips

Experiment
One or multiple faults

Campaign
Multiple experiments that follow a high-level goal of testing for failure

Golden Run
binary execution without faults

ARCHIE

Configuration

QEMU configuration: binary, machine

Fault configuration:

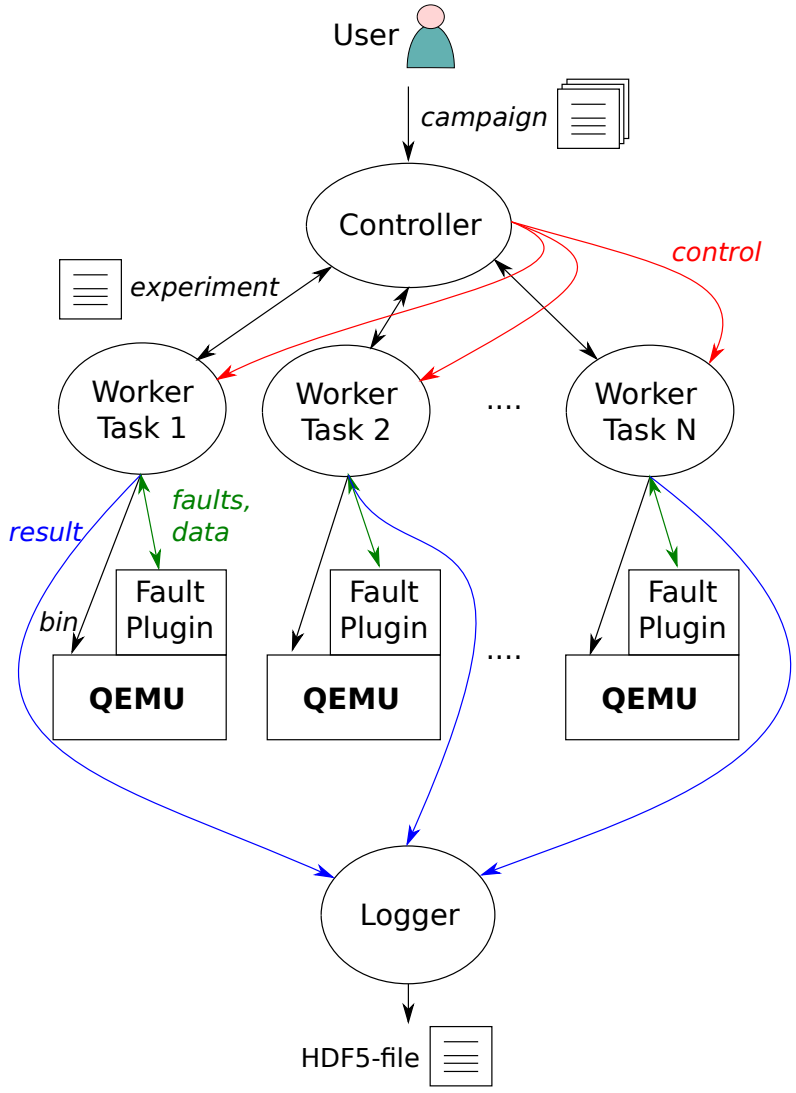
- address, type
- model, mask, lifespan
- trigger address and counter

Output:

- difference to golden run
- translation blocks
- memory access

ARCHIE

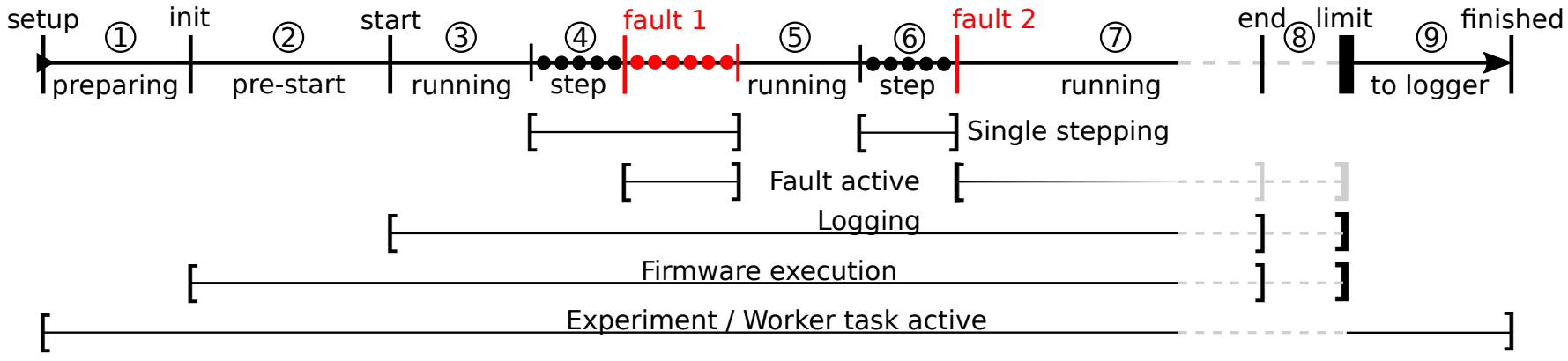
Overview of ARCHIE



Fraunhofer AISEC

ARCHIE

Execution of an Experiment



Fraunhofer AISEC

Experimental Verification

Experimental Verification

Fault Injection into AES

Laser Fault Injection (LFI) experiment [2]

- target: ARM Cortex-M0 (STM32F0discovery)
- diagonal fault attack on AES S-Box [3]

  $\Rightarrow$ 5 unique faulty ciphertexts

Analysis with ARCHIE

- campaign: transient 1-bit data faults in flash memory ("set1")
- 588 faults leading to exploitable unique faulty ciphertexts

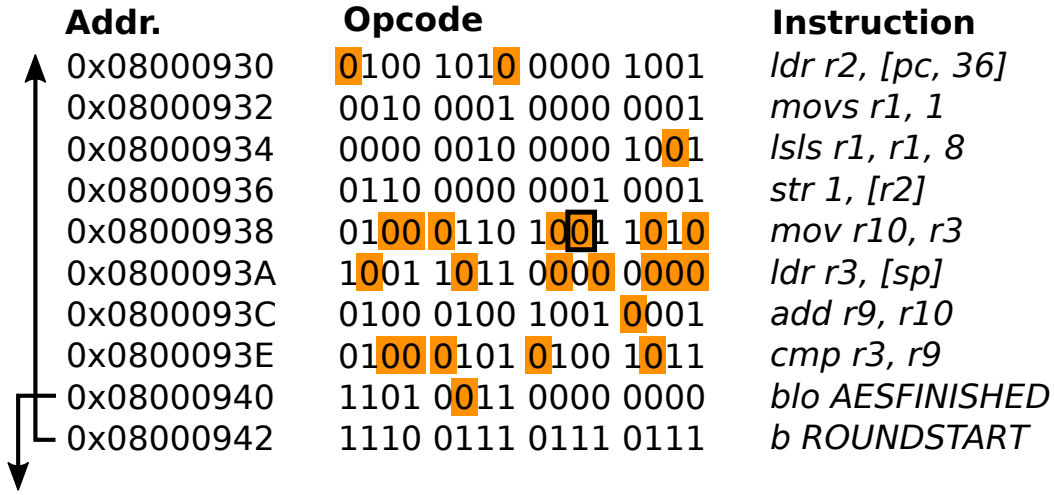
[2] K. Garb, J. Obermaier: Temporary Laser Fault Injection into Flash Memory. IEEE IOLTS 2020

[3] Saha et al., IACR Cryptology ePrint Archive, vol. 2009, p. 581, 2009.

Experimental Verification

Control Flow Manipulation of AES

- LFI attack to skip last round of AES [4] ⇒ 1 fault
- ARCHIE ⇒ 24 exploitable faults
- transient 1-bit instruction faults in flash memory (“set1”)



[4] K. Garb, J. Obermaier: Temporary Laser Fault Injection into Flash Memory. IEEE IOLTS 2020

Experimental Verification

Performance

Measurement was taken with 24,794 faults into TinyAES

Device	OS	CPU	RAM	Storage	Cores / Threads	Worker Tasks	Exec. time
Dell Poweredge Rack Server	Debian 11 via QEMU/KVM	AMD EPYC 7502	192 GB	SAS 7200rpm	32 / 64	64	**6 min 50 s**
Lenovo P52	Ubuntu 20.04	Intel Core i7-8750H	48 GB	SSD	6 / 12	12	**23 min 12 s**
Lenovo P15v	Debian 11 via VirtualBox on Windows 10	Intel Core i7-10750H	32 GB	SSD	6 / 12	12	**24 min 32 s**
Fujitsu Server TX140 S2	Debian 11 via QEMU/KVM	Intel Xeon E3-1230	28 GB	SSD	4 / 8	8	**33 min 12 s**
Supermicro Server	Ubuntu 20.04 via ESXI 6.7	Intel Xeon Silver 4108	16 GB	NAS (HDD)	8 / 8	8	**38 min 41 s**
Supermicro Server	Ubuntu 20.04 via ESXI 6.7	2 x Intel Xeon E5645	32 GB	SAN (HDD)	8 / 9	9	**42 min 21 s**
Cloud Server	Debian 10	Generic Intel Skylake	8 GB	SSD	3 / 3	3	**46 min 44 s**
Lenovo T470s	Debian 10	Intel Core i5-6200U	20 GB	SSD	2 / 4	4	**80 min 19 s**

Experimental Verification

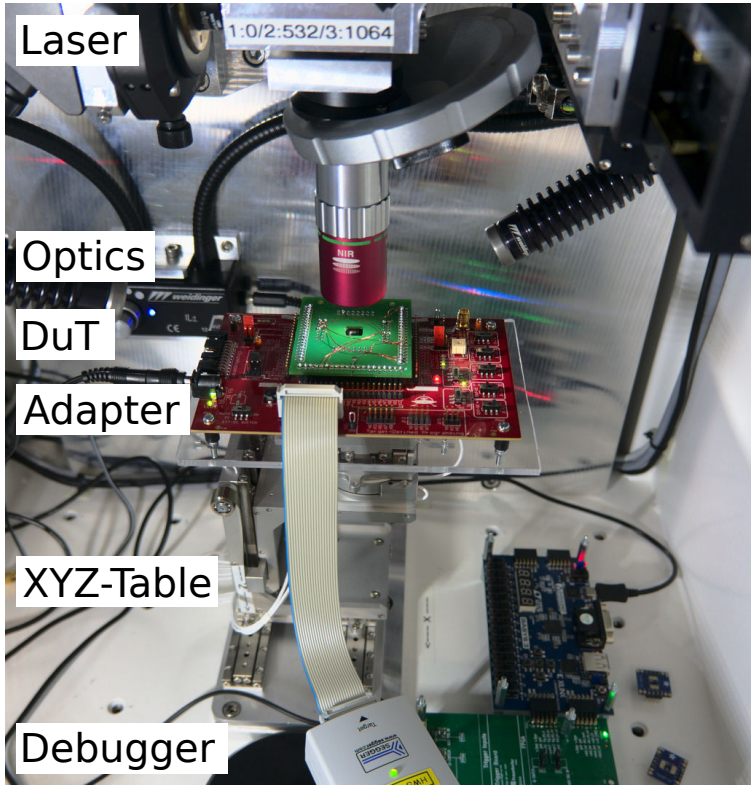
Secure Bootloader

Testing with ARCHIE

- secure bootloader implementation
- test for permanent 1-bit faults

Verification through LFI

- Infineon XMC1400 (ARM Cortex-M0)
- backside attack on SRAM



Experimental Verification

Firmware Testing on RISC-V

- RISC-V processor: 32-bit RV32IMC
- S-Box faulting:
  526 unique exploitable ciphertexts
- Control flow manipulation of AES
  only 2 possible 1-bit faults (instead of 24 for ARM)
- Secure bootloader: identical vulnerability

Summary

Summary

- ARCHIE supports all guest architectures of QEMU

  $\Rightarrow$ Verification for ARM and RISC-V architectures
- Binary testing (AES, bootloader)

  $\Rightarrow$ Large amount of different fault types

  $\Rightarrow$ Identification of new fault candidates
- ARCHIE is available on GitHub:
  `https://github.com/Fraunhofer-AISEC/archie`

Future work

- Further runtime improvements
- Add support for multiprocessor guests

Contact Information

Florian Hauschild, Kathrin Garb, Lukas Auer, Bodo Selmke, Johannes Obermaier

Fraunhofer-Institute for
Applied and Integrated Security (AISEC)

Address: Lichtenbergstr. 11
85748 Garching (near Munich)
Germany
Internet: https://www.aisec.fraunhofer.de

E-Mail: florian.hauschild@tum.de
kathrin.garb@aisec.fraunhofer.de