ACFA: Secure Runtime Auditing and Guaranteed Device Healing via Active Control Flow Attestation

Adam Caulfield[*], Norrathep Rattanavipanon[+], Ivan De Oliveira Nunes[*] | [*]Rochester Institute of Technology; [+]Prince of Songkla University, Phuket Campus

Remote Microcontroller Units (MCUs)

Resource-constrained MCUs are deployed in a wide range of modern systems but lack system security features to prevent exploits.

Secure Auditing of Remote MCU's

Can we ***remotely audit*** the behavior of a remotely deployed (and potentially compromised) MCU?

Security requirements of runtime auditing:

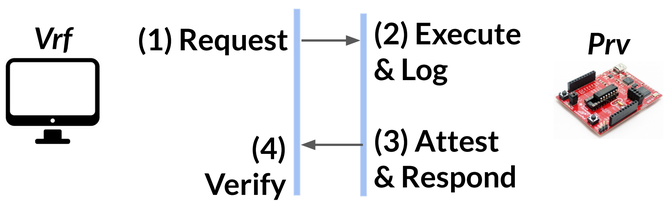
1. Generate authentic/accurate evidence of the exact runtime behavior.
2. Reliably deliver evidence for analysis.
3. Remotely remediate detected compromises.

Control Flow Attestation (CFA)

A device operator (Vrf) requests reliable evidence of a remote MCU's (Prv) behavior.

Prv responds with proof of the code and all control flow transfers that executed.

Vrf determines valid or malicious behavior.



From Attestation to Auditing

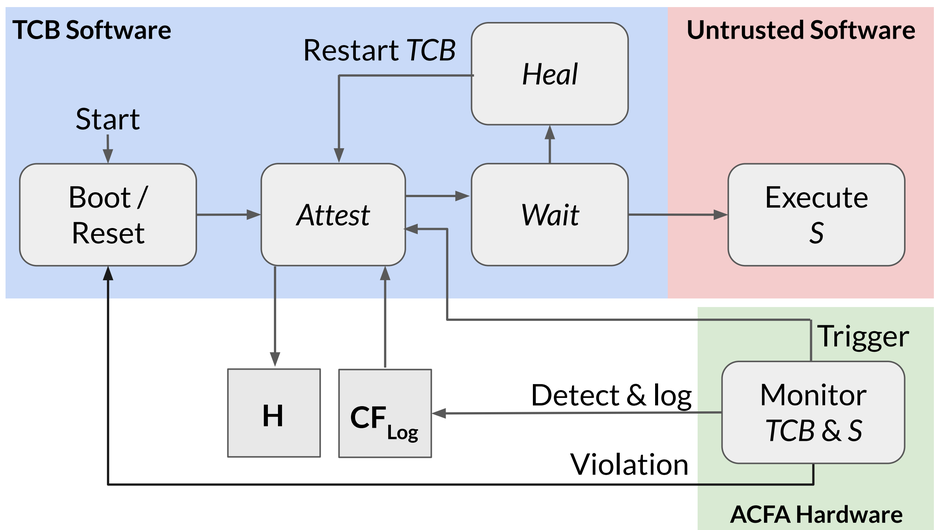
What if a compromised Prv ignores requests from Vrf?
Prior CFA: Nothing!
Why: Absence of response => compromise => enough for attestation goal.

But: Absence of response *prevents auditing* - cannot analyze evidence!
Additionally: Intervention to remedy Prv compromise must be physical.

Our Work: ACFA - Active Control Flow Attestation

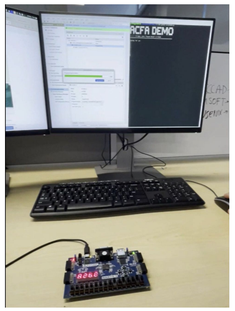
- Reliable communication of evidence implemented within ACFA's TCB
  - Hardware-protected *active* generation of evidence
  - Trigger-based attestation and transmission of evidence to Vrf.
  - ACFA TCB (within Prv) waits for authenticated approval of evidence from Vrf (retransmitting evidence periodically) before resuming untrusted execution.

- Active Remediation (in case of compromise detection):
  - Vrf-specified healing action is guaranteed to execute on Prv.

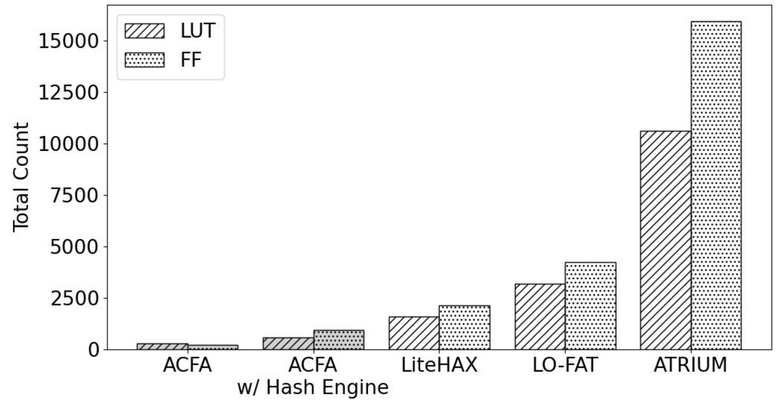


Prototype and Evaluation

Deployed on a Basys3 prototyping board equipped with an Artix-7 FPGA.



- ACFA hardware requires 275 Look-up tables (LUTs) and 202 Flip-Flop registers (FFs)
- No runtime overhead to record control flow transfers.



Resources

See our paper (USENIX Security '23) here:

ACFA open-source prototype available here:

ARTIFACT EVALUATED usenix ASSOCIATION AVAILABLE
ARTIFACT EVALUATED usenix ASSOCIATION FUNCTIONAL
ARTIFACT EVALUATED usenix ASSOCIATION REPRODUCED

Contact information (email): ac7717@rit.edu