Space RadSim

Binary-Agnostic Fault Injection to Evaluate Cosmic Radiation Impact on Exploit Mitigation Techniques in Space

Johannes Willbold, Tobias Cloosters, Simon Wörner, Felix Buchmann,
Moritz Schloegel, Lucas Davi, Thorsten Holz

RUHR UNIVERSITÄT BOCHUM
RUB
CISPA HELMHOLTZ-ZENTRUM FÜR INFORMATIONSSICHERHEIT
Arizona State University
UNIVERSITÄT DUISBURG ESSEN

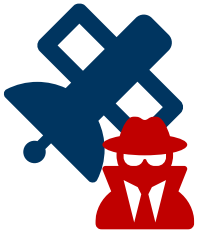
Motivation

Broad application spectrum

2025: ~2600
2023: 2136

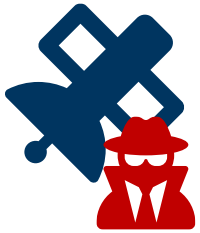
2010: 81

Increasing number of satellites



Interesting to attackers

Previous Research



Small satellites often vulnerable [1]

Lacking access control

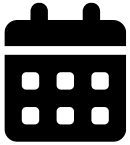
Existing research

No exploit mitigations

No Research

[1] Johannes Willbold, Moritz Schloegel, Manuel Vögele, Maximilian Gerhardt, Thorsten Holz, and Ali Abbasi.
Space Odyssey: An Experimental Software Security Analysis of Satellites. In IEEE Symposium on Security and Privacy (S&P), 2023

Hardware Survey



2020-2023
4 Years

Launch Dates

Only unique designs

Counting Method

Open Source Intelligence (OSINT)

Data Source

Year	Conventional CPUs							Radiation-hardened CPUs				No data
	ARM-C M	ARM-C A	AVR32	ARM9	AVR8	Other	∑	MSP430	LEON3	PicoSkyFT	∑	
2020	10	7	2	4	2	-	25 (51%)	1	-	1	2 (4%)	22 (45%)
2021	8	1	2	3	3	4	21 (42%)	1	-	-	1 (2%)	28 (56%)
2022	18	5	4	1	1	10	39 (30%)	4	4	-	8 (6%)	85 (64%)
2023	21	2	2	1	1	6	33 (22%)	-	-	-	0 (0%)	117 (78%)
∑	**57**	**15**	**10**	**9**	**7**	**20**	**118 (31%)**	**6**	**4**	**1**	**11 (3%)**	**252 (66%)**

Available Exploit Mitigations

Platform	Compiler	Stack Canary	Safe Stack	Shadow Stack	CFI	FSan
ARM Cortex-M	GCC	✓	✗	✗	✗	✗
ARM Cortex-M	LLVM	✓	✗	✗	✓	✓
ARM Cortex-A	GCC	✓	✗	✗	✗	✗
ARM Cortex-A	LLVM	✓	✓	✗	✓	✓
AVR32	GCC	?	✗	✗	✗	✗
AVR32	LLVM	–	–	–	–	–

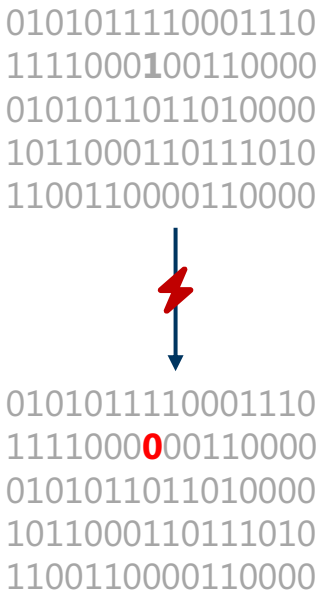
Stack Canaries (all)

Stack Canaries (strong)

Control Flow Integrity (CFI)

Function Sanitization

Single Event Upsets



Code Section Bitflips

```
02 48          LDR    R0, =5
03 49          LDR    R1, =7
00 eb 01 02    ADD    R2, R0, R1
1a 60          STR    R2, [R3]
fe e7          B      endless
```

```
02 48          LDR    R0, =5
03 49          LDR    R1, =7
00 eb 01 02    ADD    R2, R0, R1
1a 61          STR    R2, [R3, 0x10]
fe e7          B      endless
```

Data Section Bitflips

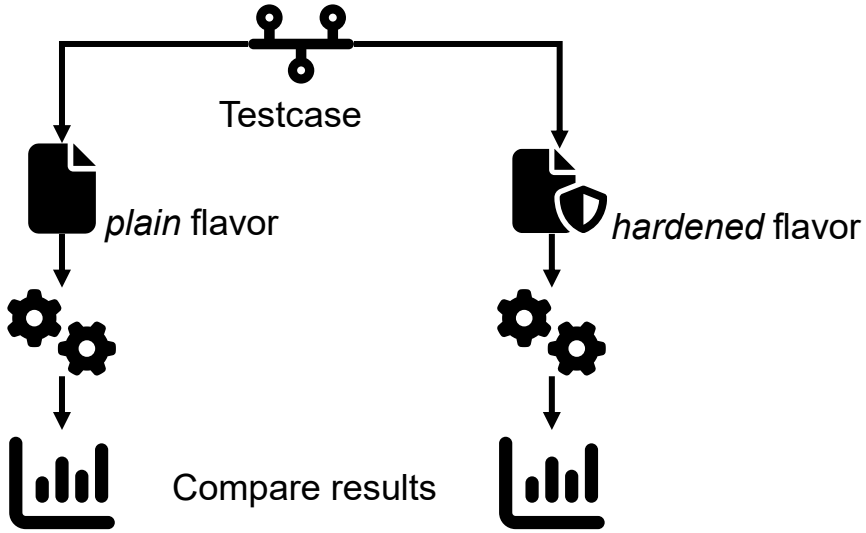
Stack Entry	No Error	Silent Error	Hard Error
Return Address		!	×
Saved Regs.	✓	!	×
Stack Canary			×
Local Vars	✓	!	×

Fault Injection (FI) Approach

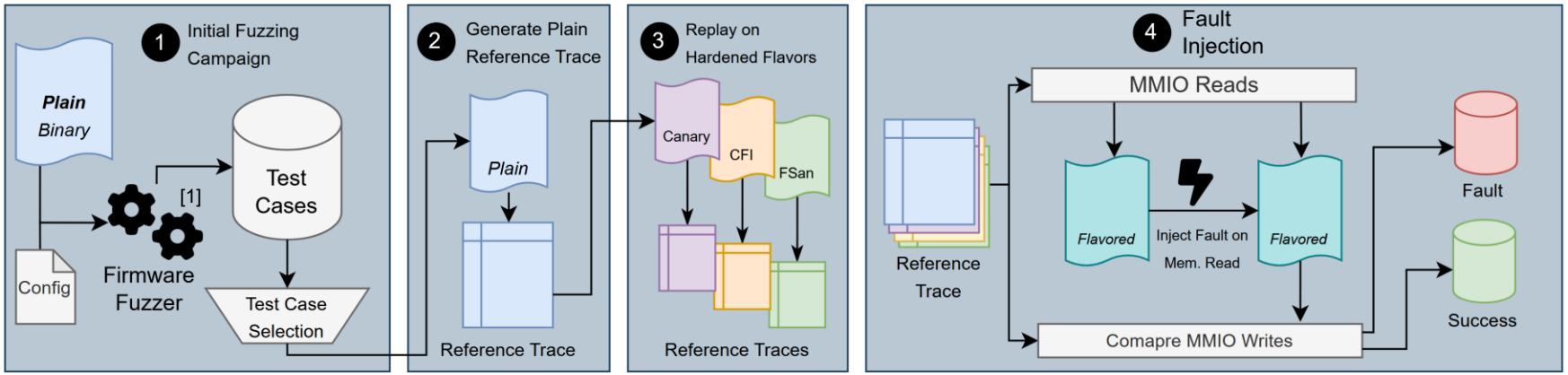
Existing Solutions:

- Pre-defined PC for FI
  - PCs break when hardening
- Probability for FI
  - Introduces randomness

Space RadSim:

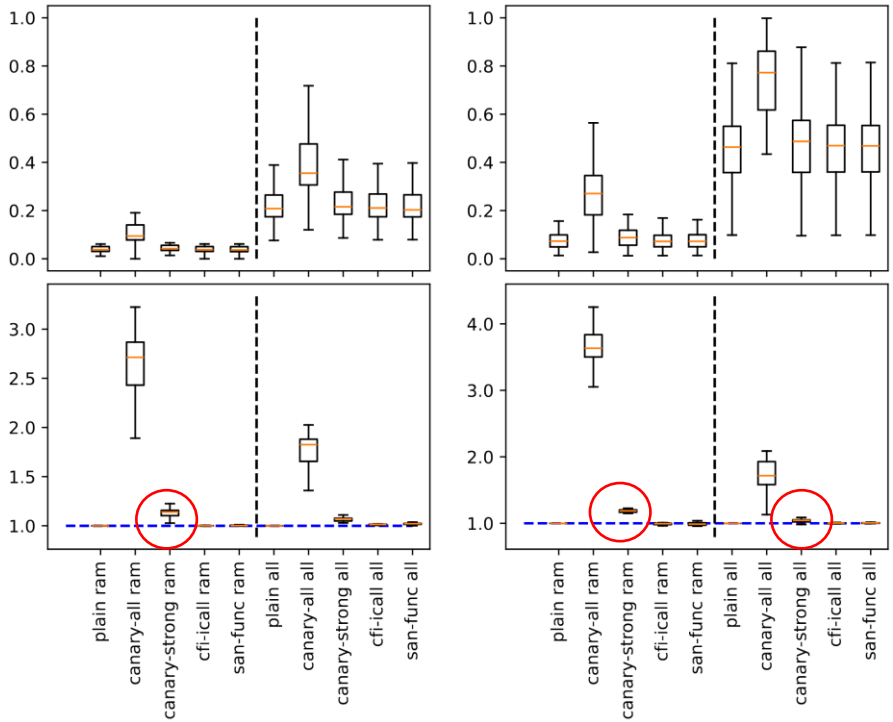


RadSim Design Overview



[1] Tobias Scharnowski, Simon Wörner, Felix Buchmann, Nils Bars, Moritz Schloegel, and Thorsten Holz. Hoedur: Embedded Firmware Fuzzing using Multi-stream Inputs. In USENIX Security Symposium, 2023.

Results



Stack Canaries (all):

- Increase crash prob. by 2.7/3.6x
- 23/44% code size increase
- Only for reference

Stack Canaries (strong):

- Crash prob.: 14/19% increase
- 3/3% code size increase

Conclusion

Survey of Applicable Mitigations:

- Extensive hardware survey
- Surveyed compiler-level support for mitigations

Binary-Agnostic Fault Injection Engine:

- Check every possible bitflip
- Stack canaries degrade failure probability beyond code-size increase

Results:

- Stack canaries degrade failure probability beyond code-size increase

Open Source Code:

CISPA-SysSec/space-radsim

Visit our poster for Q&A!

Johannes Willbold – johannes.willbold@rub.de