



Embedded

SP2026 • 2026-01-28

eCTF Design Wrapup

Announcements

- Design Concept Finished
 - Rough sketch finished last meeting on Saturday
- Fault Injection Confirmed!
 - Simple loop increment skip working on MSPM0 target board!
- Design Doc Due Friday
 - Let us know if you would like to help and we can find a topic for you
 - Ideally write the section by tomorrow night so we can edit
- 397 Forms in progress
 - If you are registered but don't have an MSPM0 board, let us know
 - Conversely, if you aren't doing design, try to return your board



Design Outline

SR 1 - Unprovisioned Should not work, 2 - PIN, 3 - Valid Files

KEYS

per group: ^{encrypted by pin hash} Content key, ^{W.C} Transfer key, ^{W.C} Transfer metadata key
 Per device: Pin salt, Pin hash

List ← Pin

- hash w/ salt
- check
- ↳ commit
- ↳ return
- ↳ otherwise
- ↳ delay penalty

Read

- check corresponding permission
- read content key + decrypt
- store file

write

- write file

inter. pull

- ... Pin
- start process
- get (no is r/group)
- receive w/ what you have
- get metadata

inter push

- send chars for each file
- verify
- ↳ send for correct
- ↳ penalty for wrong

Transfer pull

- decrypt extend metadata
- verify content if we have content key
- ↳ write to flash

Push

- send one metadata

Send UART (^{char} buf, len)

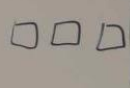
for (i=0; i<len; i++)

write byte (^{char} buf[i])

Send UART ("hi")

i = 100000

buf++



rodata
...
text



Projects Overview

Attack Projects

- SCA
- FI
- Automation (scripting)

Design Projects

- Rainbow
- Compiler Mitigations
- MPU + ECC
- List + Read + Write
- Interrogate + Listen + Receive



SCA Project

- Explore power analysis against crypto algorithms
- AES: wolfcrypt
- Chacha20: wolfcrypt, monocipher



FI Project

- Perform glitch attacks



Automation Project

- Perform automated security requirement checks
- Not sophisticated attacks, but should find common requirement misunderstandings



Rainbow Project

- Perform automated analysis for fault injection attacks



Compiler Mitigations

- Stack zeroing
- Control flow integrity/shadow stack



MPU, ECC

- Enforce memory permissions
- Deter fault injection with ECC on flash and ram



List + Read + Write

- Implement these functionalities from eCTF Spec
- More beginner friendly



Interrogate + Listen + Receive

- Implement these functionalities from eCTF Spec
- More advanced due to the crypto protocol



Next Meetings

2026-01-31 • This Saturday

- First subteam meeting
- Also: Nikhil will present AEAD



ctf.sigpwny.com

Meeting content can be found at
sigpwny.com/meetings.

