



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: Content key, Transfer key, Transfer metadata key

Per device: Pin salt, Pin hash

W,C

W,C

i = 10000 0

buff

~ ~ ~

□ □ □

rodata  
text

LIST & PIN

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

RPOD

- check corresponding permission
- read content key + decipher
- save file

WRITE

- write file
- inter. NULL
- ... Pin
- start process
- get (has w/group)
- respond w/ what you have
- get metadata

Transfer NULL

- decrypt content using transfer metadata key
- verify content if we have content key
- └ write to flash

Push

- inter push
- send chars for each file
- verify
- └ send for correct
- └ penalty for wrong

Send one metadata



# 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](http://ctf.sigpwny.com)

Meeting content can be found at  
[sigpwny.com/meetings](http://sigpwny.com/meetings).

