MCUboot Overview

David Brown

2023-06-15

Agenda

- Overview
- History and State
- Security
- Future

MCUboot: Overview

“MCUboot is a secure bootloader for 32-bit microcontrollers.”

- Secure bootloader: includes signed root of trust, and secure upgrades.
- 32-bit microcontrollers. Constrained devices
  - Usually boot and run from flash.
  - 100s ok KiB of Flash, often 10s or 100s of KiB of SRAM.
  - Think Cortex-M, although not particular CPU-specific

Secure bootloader?

- Secure boot
  - Provided requirements met, mcuboot verifies signature of images before booting them.
  - Provides attestation information.
- Secure upgrade
  - Verifies signatures of new images.
  - Robustly swaps old and new images (several swap algorithms available).

MCUboot: History

- Began as “bootutil” in the mynewt RTOS (now an Apache project)
- Linaro ported to Zephyr, it became ‘mcuboot’ instead of just a thing inside of mynewt.
- Other platforms supported over time:
  - More RTOSes: mbed, nuttx
  - Bare metal ports: Cypress, Espressif
- More features added
  - serial and usb recovery
  - multiple images with dependencies

MCUboot: state

- Github project
  - Used for pull request, code review, issue tracking, and planning, as well as security vulnerability reporting and advisories.
  - Github provides CI with adequate resources for current project activity.
- Currently a slack instance, should probably move to TF Discord
- Several active maintainers from a few different companies
- Integrated tightly into TF-M and Zephyr, used as primary bootloader.
- Periodic releases, primarily demand driven.

MCUboot: Project Security

- Reporting:
  - Was via email to maintainers
  - Tried hackerone for a bit, not very useful
  - Have now enabled Github vulnerability reporting (this integrates nicely with rest of github)
- CVEs
  - Have been getting allocations from Zephyr
  - Reports made with Github “security advisories”
  - Docs state 90 day embargo. Not robustly kept
- Integration with other Trustedfirmware projects?

Future Work

- IETF SUIT (Software Update for IoT)
    - standardized metadata for signing and encrypting firmware images
    - Would require substantial changes to MCUboot code, possible gradual approach
- “Large write” devices
    - Many newer devices have strict requirements on writes, not yet supported for some modes (such as image swap)
    - Ideas have been discussed and some prototype code written
- Overall code cleanup
    - Code has acquired a lot of ifdefs, difficult to add new features

Questions?