Verified Boot and firmware updates

How to do them securely and openly

Michał Kopeć

whoami

- Michał Kopeć (original author)
  - Firmware Engineer at 3mdeb since 2021
  - Develops Dasharo for laptops, network appliances and other platforms
  - Uses Arch btw
- Michał Żygowski (presenter)
  - Firmware Engineer at 3mdeb since 2017
  - Develops Dasharo
  - open-source, HW and security features enthusiast

CC-SA-4.0 | Michał Kopeć

Agenda

- Verified Boot
  - in Chrome
  - in Dasharo
- Problem statement
- Boot Guard
- CBFS verification
- Secure firmware updates

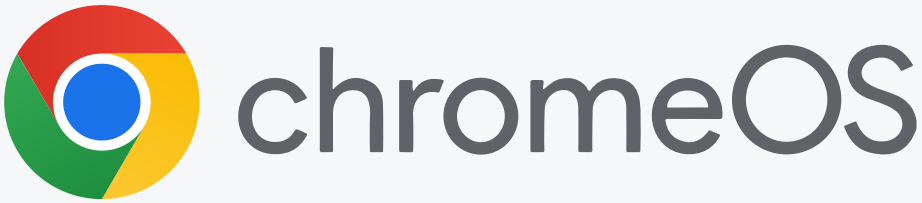
CC-SA-4.0 | Michał Kopeć

Verified boot

- Verified Boot: Cryptographic verification of the boot process to ensure only code from a trusted source (e.g. device vendor) can run
- coreboot supports Vboot, Google's version of verified boot scheme
- Vboot has:
  - A root of trust
  - Verification of subsequent firmware stages
  - Signatures and public tools for self-signing
  - A special A/B update mechanism
  - Support for re-ownership
- Dasharo uses Vboot

CC-SA-4.0 | Michał Kopeć

Verified Boot in Chrome devices



- Write protection: WP# pin on the BIOS chip
  - WP# pin is controlled by Cr50 security chip (newer devices) or physical screw (older devices)
- Updates: An OS service handles the A/B update scheme
  - One slot is updated, and when confirmed bootable, the other slot is updated too
- Write protected portion of the flash is **not** updated, and serves as recovery
  - WP region contains the initial bootblock and verifies the A/B slots
  - When both slots fail or recovery is manually requested (e.g. by keyboard key), the firmware boots from the recovery partition

CC-SA-4.0 | Michał Kopeć

Verified Boot in Dasharo

- Write protection is provided by chipset (typically, can be also SPI WP#)
  - Optional feature, but enabled by default for most platforms
  - Protects against software attacks
- Updates are handled by Dasharo Tools Suite
  - Capsule Update and fwupd support is WIP
- Most updates also need to update the bootblock, so they require protection to be disabled for updates - obviously suboptimal
- OS secure boot is handled by UEFI SB

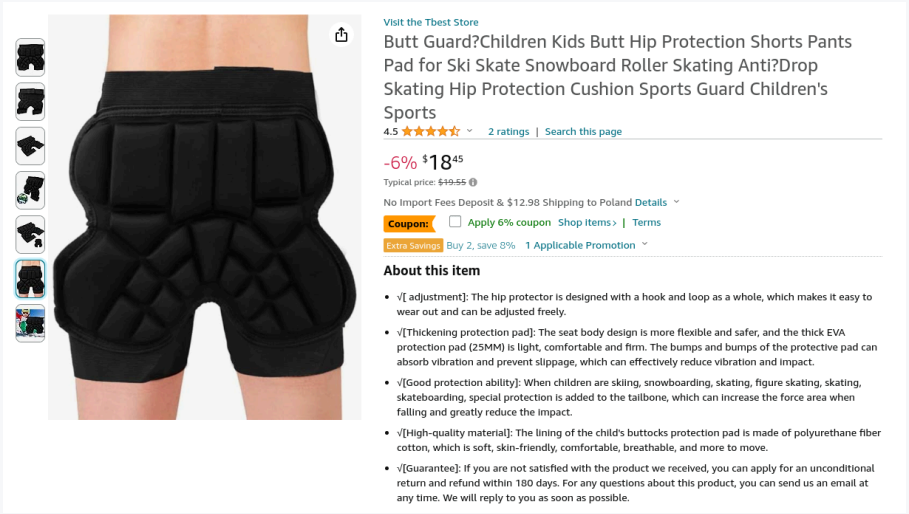
CC-SA-4.0 | Michał Kopeć

Problem

- How do we improve Dasharo verified boot while staying secure and without taking control away from users?
  - Security: A guarantee that only firmware from a trusted vendor is run
  - Control: Ability to self-sign, to inspect and replace firmware components
- There are other verified boot schemes

Image source: https://picryl.com/media/question-mark-note-man-people-55a8e2
Image license: Creative Commons CC0 1.0 Universal Public Domain Dedication

Intel Boot Guard



Source: Amazon

- The most widely known option
- Verifies and measures the initial bootblock
- Ensures FW authenticity using keys fused to the chipset
- Different profiles with different features enabled
    - Verified boot is always enabled in all profiles
- **Ties a platform to a specific firmware vendor, forever!**

Boot Guard cons

Antu selinux CC BY-SA 3.0

- Requires more blobs in firmware
  - Boot Guard ACM
- Fusing removes some owner control
- Profile 3 does not ensure initial boot block authenticity
  - Only helps us establish a root of trust for measurement
- Other profiles take away owner control completely

CBFS Verification

- A relatively new coreboot feature
- Cryptographically verifies components of the coreboot image
- According to documentation:

  > This only makes sense if you use some out-of-band mechanism to guarantee the integrity of the bootblock itself, such as Intel Boot Guard or flash write-protection.

  - Same applies to vboot's recovery region, which is ultimately trusted

- Depends on some other mechanism for signing (e.g. Boot Guard)
- Can effectively replace the portions of Vboot that Dasharo uses

CC-SA-4.0 | Michał Kopeć

Firmware updates

- Chrome Vboot: A+B+RO
  - A and B slots for updates, RO slot contains IBB and verification code
  - Anti-rollback using TPM
  - Vboot aware OS service manages slot updates
- Dasharo:
  - Initial idea was to leave RO untouched and only update A/B slots
  - In practice, most updates introduce breaking changes that require updating the bootblock
  - Dasharo Tools Suite handles updates, does the entire update in one operation
  - So we don't use the A/B feature
  - Capsule Updates may help here

CC-SA-4.0 | Michał Kopeć

Other firmware update mechanisms

The Flash Wallpaper by kelso CC BY-SA 3.0

- Flashrom plugin in fwupd
  - Updates the BIOS region only
  - Is not Vboot aware
- UEFI Capsule Update
  - Most widely used in proprietary UEFIs
  - Can be made aware of which regions to update and which to preserve, verify signatures, handle disabling flash protections
  - WIP for Dasharo

Other firmware update mechanisms

- Intel BIOS Guard
  - Hooks into the SMM flash update handler to call an ACM, which authorizes flash writes
  - Can bypass chipset flash protections and Top Swap (configurable with MFIT/FIT/FITC in flash descriptor straps)
  - Can also handle EC updates
  - Proprietary feature with proprietary tooling
- Top Swap
  - Redundant bootblock feature
  - Configurable from 64KB up to 4M/8MB (maximum depending on CPU/SoC/chipset family) in flash descriptor straps (MFIT/FITC/FIT)
  - Potential integration with vboot?

CC-SA-4.0 | Michał Kopeć

Q&A