proxy

- What we are building
- Hardware integrity
- Software integrity
- Device authenticity check

What we are building

- Hardware × Software
- Non-custodial wallet for digital identity and digital assets
    - Focus on ease of use and safe defaults
    - User can’t “hold it wrong”
- Wearable hardware wallet (NFC, BLE, SE)
    - Companion to the mobile software wallet
    - Co-signer in multisig transactions
    - Can participate in wallet recovery

🚧

Unreleased product, everything here can change

Components

- The pieces:
    - MCU + BLE
    - secure element (SE) + NFC
    - sensors: capacitive, force, fingerprint [opt.]
- Secure element
    - keeps user secrets
    - component auth codes (more below)
    - protection from physical and side-channel attacks
    - provides TRNG
- Extremely constrained form factor (power, size)

System perspective

Hardware integrity (1)

- PUF in MCU and FP components
- PUF: physical uncloneable function
  - derived from physical irregularities of the silicon
  - device-unique and uncloneable
  - immutable
  - key does not need to be programmed
  - key does not exist anywhere when power is off
- Used to key-wrap and derive other keys, and to authenticate components (in conjunction with the SE)

Hardware integrity (2)

- At mfg time, write PUF auth code into the SE
    - one-time-write
    - can be read freely, since auth codes themselves are not keys (need original silicon to recover the key)
- At runtime, use auth code to reconstruct key on demand
    - derived keys for actual use
    - minimize time the reconstructed key exists in memory
- Permanently bond together MCU + SE + FP components

Hardware integrity (3)

- MCU ⇢ key-wrapped keys:
    - Encrypt bus comms
        - MCU ⇄ SE (secure channel)
        - MCU ⇄ FP (image capture data)
    - Encrypt sensitive data in flash (FP templates)
- FP ⇢ derived keys:
    - Encrypt bus comms
- Swapping out any component breaks its comms
- Can be used as part of a composite authenticity check

Software integrity

- Bootloader only accepts firmware images signed by Proxy
    - two image slots
    - automatically revert invalid images
    - image downgrade protection
- MCU debugger interface
- MCU memory protection of bootloader region
- SE applets
    - field-upgradable (except for applets storing user data and system authenticity info)
    - verification of load file signatures on install (gp)

Device authenticity check (🚧)

- At manufacture time, register SE generated key and hash of PUF auth codes generated on device
- Mobile app presents challenge over NFC, reads back a cryptogram that can be verified by `proxy.com` if device was manufactured by Proxy

“Do better” list

- Secure code integrity checks
    - MCU signature check must rely on code running on the MCU; subject to glitch attacks and silicon vendor bugs
    - participate in device authenticity check
- Transparent encryption of ext. memory reads/writes
    - currently done “manually” by MCU, only some data
    - cannot use with DMA controller
- Physical tamper evidence
- Authenticity check using WebBluetooth / WebNFC from browser

proxy

@proxy ~

@simonratner ~