Let’s talk about Turtles

- Roots of Trust
- Trust Anchors
IDevID considerations document

- This document is about the quality of the turtles
  - How do they get there?
  - Can they be trusted?
    - How much?
    - **For what?** (Is the risk mitigation appropriate to the user’s threat model?)

- Three fundamental ways to provision initial roots of trust.

- Ultimately, the software update trust anchor **rules everything**.
Roots of Trust

• How are they provisioned?
  − What would be involved in compromising that process?
    • assume: bribery, kidnapping, might be used
  − How can we qualify the different processes?
    • Not every process is appropriate for every end use.

• NDAs abound, but Supply Chain considerations mean some of these things need to get through anyway
Goals of this document

- Enumerate the reasonable, and maybe some less reasonable ways to provision and secure keys, and give them names.
- Not just the most secure way, because it is not always worth it.
The document so far

- **Trust Anchor**
  - a thing a device uses to verify an external entity’s identity

- **IDevID**
  - a thing a device uses to prove an identity to an external entity
  - ways of provisioning key pair
Industry Consultations

- secdispatch said to take this to industry people to get their feedback
- two public presentations on this, and four private discussions
- yet to get any feedback!
- everyone busy due to pandemic, but still persuing feedback.
Public Key Infrastructure

- using “subordinate” rather than “intermediate”
- self-signed certificate is a PKI of level “one”
  - not counting from zero
- intermediate used in bridge CA use
- see https://fpki.idmanagement.gov/tools/fpkigraph/

- This document about the shapes of these things.
- Recovery and Resilience
- How are private keys kept safe?
Properties of PKI

- initial-enclave-location:
- initial-enclave-integrity-key:
- initial-enclave-privacy-key:
- first-stage-initialization:
- first-second-stage-gap:
- identity-pki-level:
- identity-time-limits-per-subordinate:
- identity-number-per-subordinate:
- identity-anchor-storage:
- pki-level:
- pki-algorithms:
- pki-level-locked:
- pki-breadth:
- pki-lock-policy:
- pki-anchor-storage:

- many attributes shown on left
- not at all complete!

- How to deal with level of secret splitting?
  - business continuity vs risk of counterfeit
Intended vs Unintended Business Continuity

- Use Shamir Secret Sharing on PKI keys
  - 4 out of 7 pieces
    - generally n of k
- how to distribute pieces?
- do they reconstruct the PKI private key,
  - or do they just restruct the HSM secret that unlocks the private key?

More pieces => more resiliency to “bus events”

higher threshold => more resistance to corruption, bribery, extortion?

If operations are spread across continents, should key pieces too?

HSMs are great, but expensive, and one needs two or three vs a bootable CDrom and any PC?
Confidentiality of IDevID private key..

Non-Disclosure Agreement

Firmware TPM

Hardware TPM

Supply Chain Security Audit

Silicon Root Of Trust
Adding layer of indirection...

Auditor: Returns Normative Description

Supply Chain Security Audit
Audit Model

Recognize:
Posessor of Bootloader software update key wins all battles.

• However >pubkey< is provisioned determines in-system risk of entire system.
  - This is the bottom turtle, “Mack”, and he’d better not burp.

• Even more critical: how is the private key that can sign code kept?

![Diagram with stages of bootloader and WebPKI trust anchors]