Trusted Boot

Components and Challanges

Adam Petz, Dr. Perry Alexander

Information and Telecommunication Technology Center
Electrical Engineering and Computer Science
The University of Kansas
ampetz@ittc.ku.edu

Formatted with the KU Beamer Class for LaTeX 2ε

How can we trust that a platform has booted into a trustworthy state?

- Hardware roots of trust
- Trusted “Measurers”
- Protocols that sequence Measurers properly
- Evidence packages that can be appraised remotely

Hardware Root of Trust

The *Trusted Platform Module (TPM)* is a cryptographic co-processor for trust.

- Endorsement Key (EK) — factory generated asymmetric key that uniquely identifies the TPM
- Platform Configuration Registers (PCRs) — protected registers for storing and extending hashes
  - Locality — Access control like OS security rings
- ...

Root of Trust for Measurement

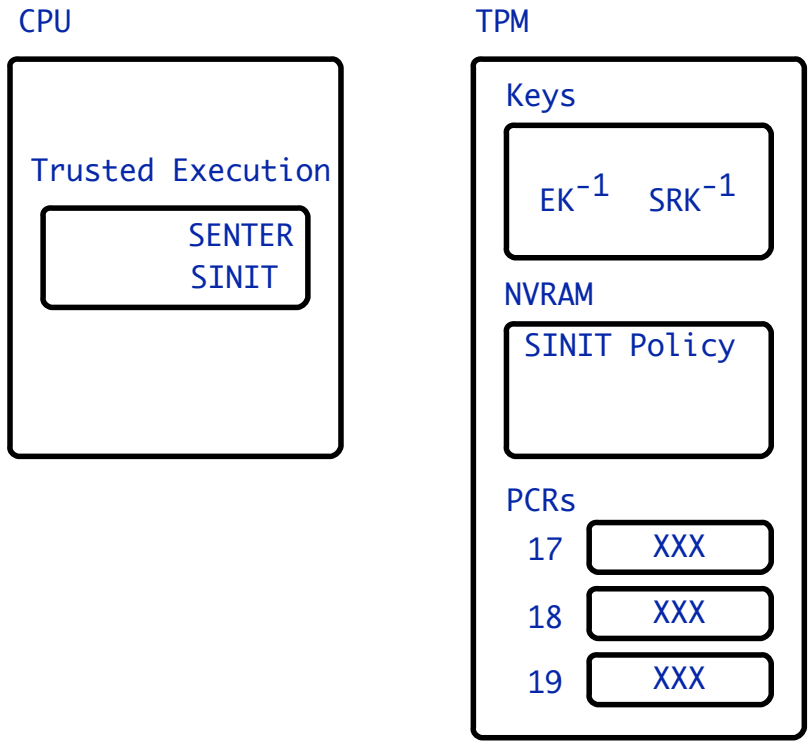
A *Root of Trust for Measurement* is trusted to take the base system measurement.

- A hash function called on an initial code base from a protected execution environment
- In the Intel TXT process the RTM is `SENTER` implemented on the processor

One Step from Roots of Trust

Roots of trust are used to build a trusted system from boot.

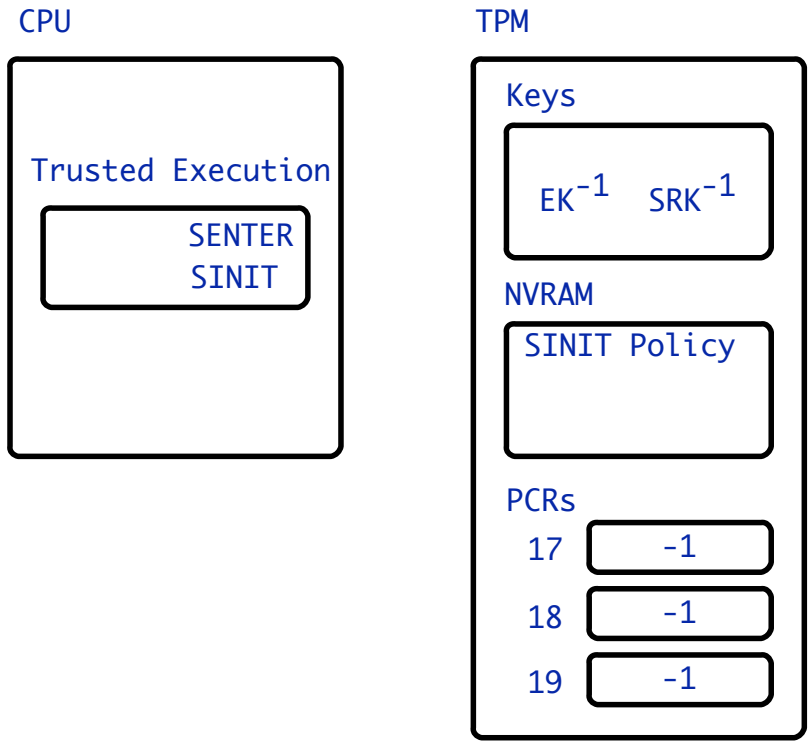
- Power-on reset



One Step from Roots of Trust

Roots of trust are used to build a trusted system from boot.

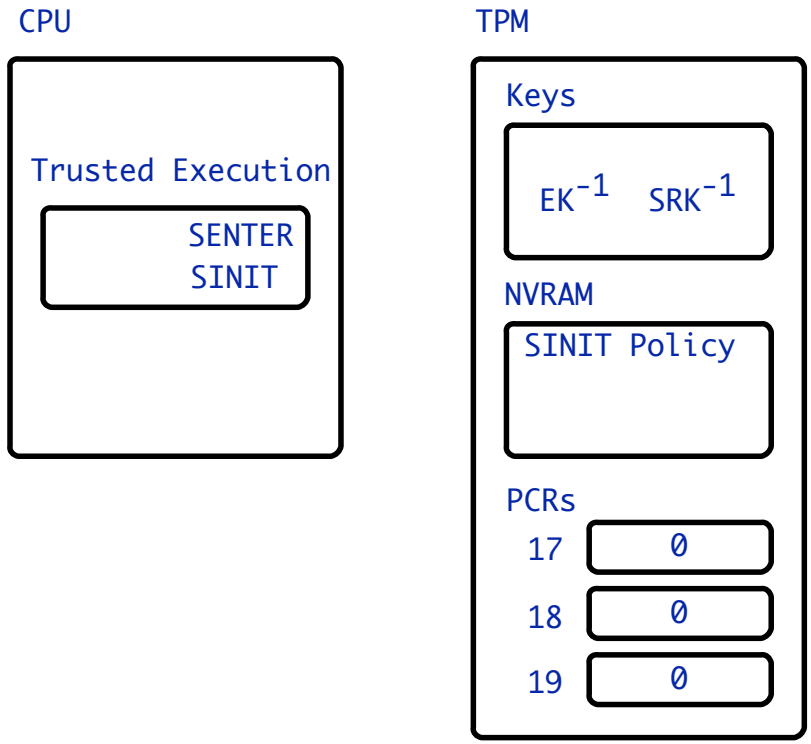
- Power-on reset
- Resettable PCRs set to -1



One Step from Roots of Trust

Roots of trust are used to build a trusted system from boot.

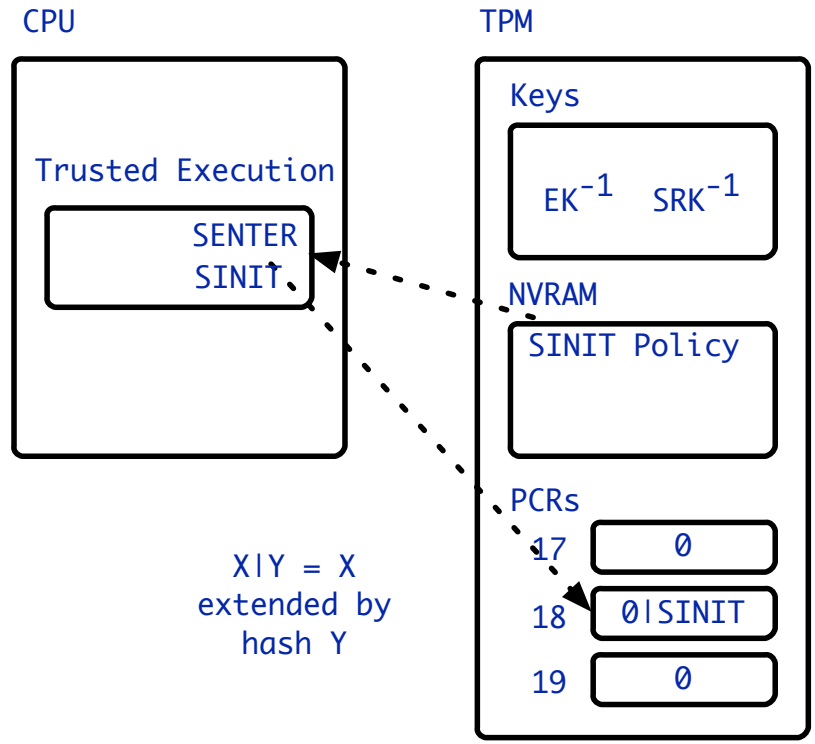
- Power-on reset
- Resettable PCRs set to -1
- `SENTER` called, resets resettable PCRs to 0



One Step from Roots of Trust

Roots of trust are used to build a trusted system from boot.

- Power-on reset
- Resettable PCRs set to -1
- `SENTER` called, resets resettable PCRs to 0
- `SENTER` measures `SINIT` policy into PCR 18



What We Know From Good PCR 18

A good value in PCR 18 tells us:

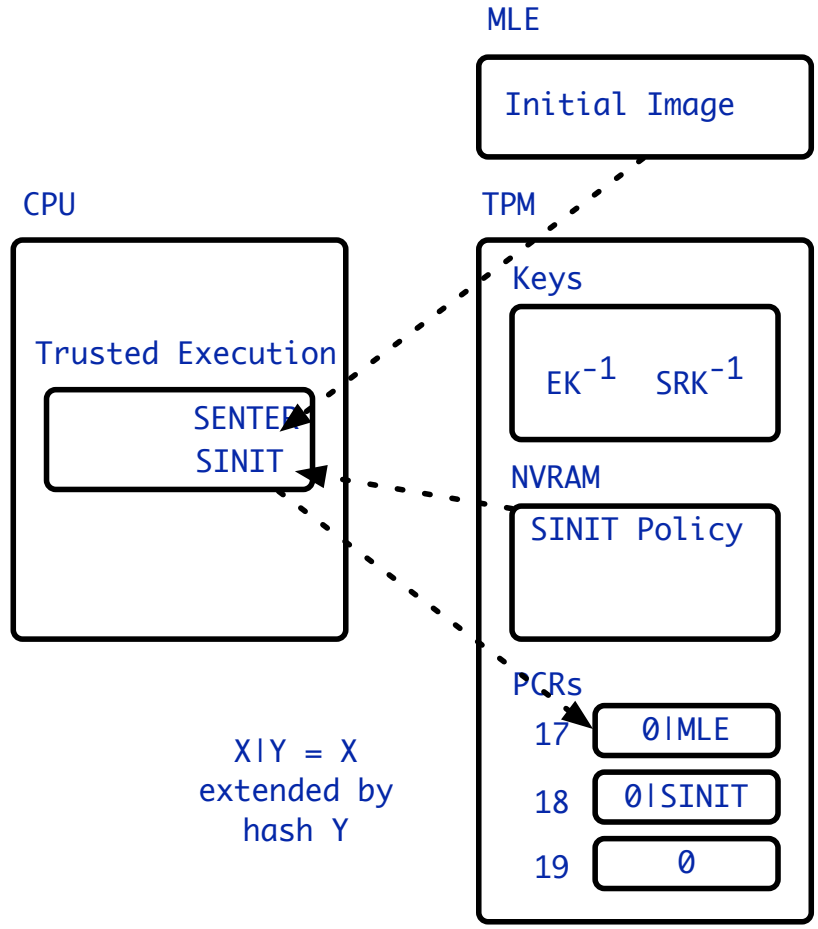
- `SENTER` was called — Resetting PCR 18 starts measurements at 0 rather than -1
- `SINIT` was measured by `SENTER` — Only `SENTER` can extend PCR 18
- `SINIT` uses the correct policy — PCR 18 is extended with `SINIT` measurement policy
- `SENTER` ran before `SINIT` was measured — $A \mid B \neq B \mid A$

Measurement $\neq$ Trust

Measurements must be appraised to determine trust.

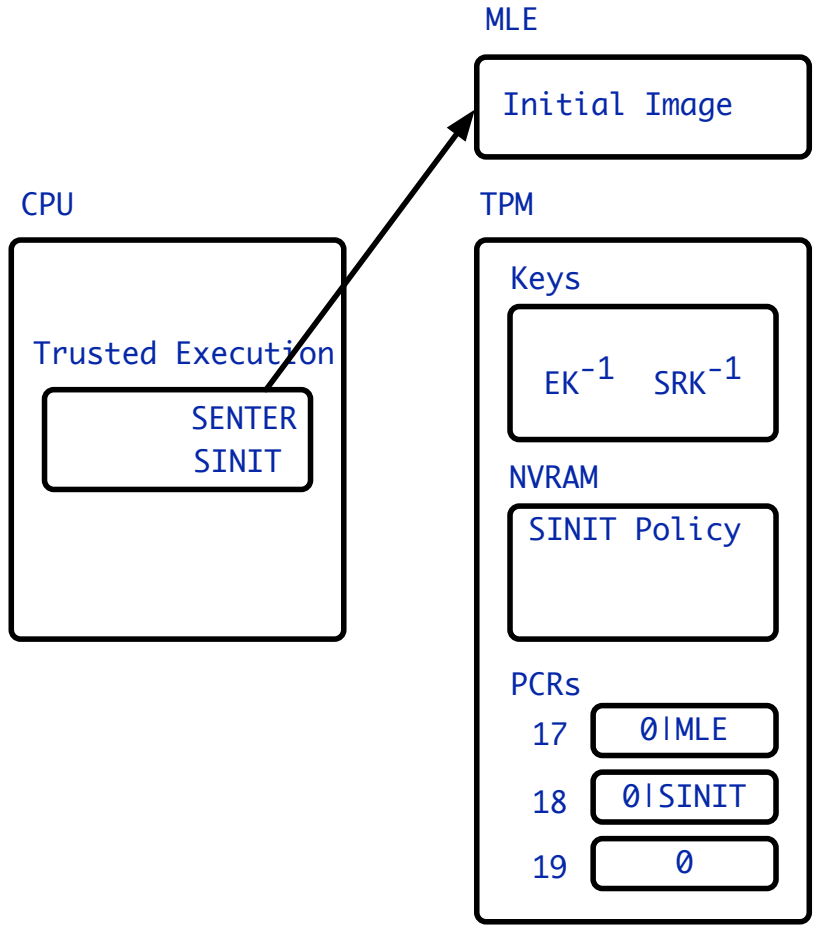
Two Steps from Roots of Trust

- `SINIT` measures the Measured Launch Environment (MLE) using measured `SINIT` policy



Two Steps from Roots of Trust

- SINIT measures the Measured Launch Environment (MLE) using measured SINIT policy
- SINIT returns control to SENTER
- SENTER invokes the MLE



Appraising a Quote

Given q of the form:

$$q = [\langle n, pcr \rangle]_{AIK^{-1}}$$

1. Signature check using *AIK* verifies authenticity
   - Signature was generated by a TPM with AIK installed
   - Appraiser must know *AIK*
2. *pcr* check verifies built from good parts in the right order
   - Compare PCR composite to known good PCR composite
   - Composite generated from desired golden PCR values
3. Nonce check guarantees freshness
   - Nonce is random and known to the appraiser
   - Sent to the target during appraisal

Verification Tasks

1. Modeling TPM commands
2. Using (crypto) properties of the TPM as axioms in larger proofs about *Remote Attestation* protocols
3. Modeling evidence packages, evidence checkers
4. Verification of Measurers (external)
5. Multi-party/Multi-realm attestations (difficult)

- The Ten Page Introduction to Trusted Computing
  `https://www.cs.ox.ac.uk/publications/publication2836-abstract.html`

- Principles of Remote Attestation
  `http://web.cs.wpi.edu/~guttman/pubs/good_attest.pdf`

- Coq Semantics for Remote Attestation Protocols:
  `https://github.com/armoredsoftware/session`

Questions?