The TPM as Controlled State Repository: Modelling Trusted Platform Modules

Ariel Segall
Joshua Guttman

October 2, 2008

The Trusted Platform Module

- Small trusted hardware chip
- Relied upon to store keys & measurements
- Limited command set
- Predictable behavior (we hope!)

Controlled State Repositories

Controlled state repository:

- Stores data
- Modifies data in restricted patterns
- Interaction results indicate current state

The TPM is a controlled state repository!

PCR_Reset Role

$\leftarrow$ PCR_Reset(p, *mask*, *locality*)

$\Downarrow$

$\mathsf{E}_{\mathsf{PCR\_Reset}}(p, \mathit{mask}, \mathit{locality}, \_, \_)$

$\Downarrow$

$\rightarrow$ PCR_ResetResult(p)

PCR_Reset Rules (1/2)

PCR_attrs(p, *attrs*),
PCR_values(p, *pcrs*),
$\rightarrow$
$\mathrm{E}_{\mathrm{PCR\_Reset}}$($p$, *mask*, *locality*, *newpcrs*, *attrs*)

when

Resettable(*attrs*, *mask*),
Rst_loc(*attrs*, *mask*, *locality*),
Defaults(*mask*, *newvalues*),
Vecs_differ_only(*pcrs*, *newpcrs*, *mask*, *newvalues*)

PCR_Reset Rules (2/2)

$E_{\text{PCR\_Reset}}(p, \mathit{mask}, \mathit{locality}, \mathit{newpcrs}, \mathit{attrs})$

$\rightarrow$

$\text{PCR\_attrs}(p, \mathit{attrs})$,
$\text{PCR\_values}(p, \mathit{newpcrs})$

State Effects from PCR_Reset

PCR_attrs(p, *attrs*),
PCR_values(p, *pcrs*),
$\longrightarrow$
...
$\longrightarrow$
PCR_attrs(p, *attrs*),
PCR_values(p, *newpcrs*)

Advantages of modelling the TPM

- Simplicity
  - >700 page specification
  - <40 page model (partial)
- Comprehensibility
  - Abstract away details
  - Focus on external effects
- Analysis
  - What we can rely on
  - What the TPM relies on

Analysis with the model

What guarantees does the TPM provide?

- State before command? After?
- State invariants?
- Possible past or future behavior?
- Authorization of this command or others?

The spec makes lots of invisible assumptions!

Going forward

- Implement rules in Maude
- Extend model
- Integrate with protocols & ALIS
- Provide model to other researchers
- Model other controlled state repositories

Modelling a Controlled State Repository

- Each CSR is one principal
- Interactions (commands) are strand space protocols
- State changes are multiset rewriting rules
- Event nodes in strands trigger rules
- A rule affects only one principal
- All failures look the same (for now)