Transition Systems and Invariants

EECS 755—Software Systems Modeling

Dr. Perry Alexander

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

Presentation Outline

- Review access control modeling objectives
  - modeling platform MAC
  - modeling local access control
- Overview access control policy definition
  - design and modeling assumptions
  - platform boot policy definition
  - local policy definitions
- Overview models
  - domain and system models
  - communication model
  - theorems and status
- Identify next steps
  - runtime and moving beyond the SVP line
  - adding M&A detail

KU INFORMATION AND TELECOMMUNICATION TECHNOLOGY CENTER
The University of Kansas

Access Control Modeling Objectives

What we're about here

Reporting joint work with Geoffrey Brown, Indiana University (submitted) in which we verify two physical layer protocols.

- Biphase Mark Protocol (BMP)
- 8N1 Protocol

These protocols are used in data transmission for CDs, Ethernet, and Tokenring, etc. as well as UARTs.

- Correctness is reasonably difficult to prove due to many real-time constraints.
- Many previous formal modeling/verification efforts for these protocols.

Columns and Blocks

Trying figures next to lists

Some normal text goes here just for introduction

- Appraisal
- Measurement
- Attestation
- vTPM

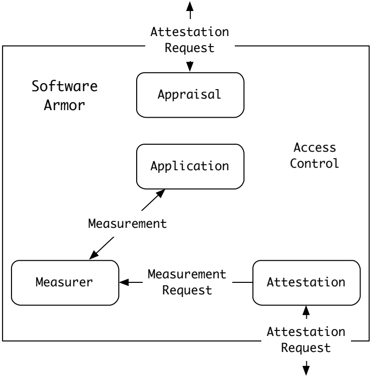
Why is this column getting higher?

Maybe it's not

Center alignment seems best.

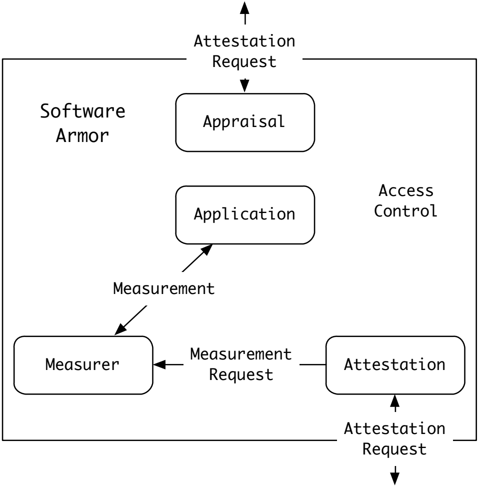
I like this for two column test and graphics

Getting higher???



Big Picture

Armor Architecture



Simple Block

Introduction to LaTeX

“Beamer is a LaTeXclass for creating presentations that are held using a projector...”

This is a definition

Proofs

Not really a proof.

1. This is a step

□

Proofs

Not really a proof.

1. This is a step
2. This is another step

□

Proofs

Not really a proof.

1. This is a step
2. This is another step
3. This is a third step
4. This is a third step
5. This is a third step
6. This is a third step

□

List with Overlays

- Item 1 followed by a pause

List with Overlays

- Item 1 followed by a pause
- Item 3 followed by a pause

List with Overlays

- Item 1 followed by a pause
- Item 2 followed by a pause
- Item 3 followed by a pause

Previous Efforts

- BMP has been verified in PVS twice and required
  - 37 invariants and 4000 individual proof directives (initially) in the one effort
  - 5 hours just to *check* the proofs in the other effort
  - A formal specification and verification of an independent real-time model in both efforts
- BMP has been verified in (the precursor to) ACL2 by J. Moore and required
  - A significant conceptual effort to fit the problem in the logic, arguably omitting some salient features of the model
  - The statement and proof of many antecedent results
  - J. Moore reports this as one of his “best ideas” in his career

Not Your Father's Theorem-Prover

The verifications are carried out in the SAL infinite-state bounded model-checker that combines SAT-solving and SMT decision procedures to *prove* safety properties about infinite-state models.

- Theorem-proving efforts took multiple engineer-months if not years to complete.
- Our initial effort in SAL consumed about *two engineer-days*. ...and we found a significant bug in a UART application note.

Parameterized Timing Constraints

SMT allows for *parameterized* proofs of correctness. The following are example constaints from the BMP verification:

```
TIME: TYPE = REAL;

TPERIOD: TIME = 16;
TSAMPLE: INTEGER = 23;

TSETTLE: {x: TIME |     0 <= x
                    AND (x + TPERIOD < TSAMPLE)
                    AND (x + TSAMPLE + 1 < 2 * TPERIOD)};

TSTABLE: TIME = TPERIOD - TSETTLE;

ERROR: {x: TIME |     (0 <= x)
                  AND (TPERIOD + TSETTLE < TSAMPLE*(1-x))
                  AND (TSAMPLE*(1+x) + (1+x) + TSETTLE < 2 * TPERIOD)};

RSAMPMAX: TIME = TSAMPLE * (1 + ERROR);
RSAMPMIN: TIME = TSAMPLE * (1 - ERROR);
RSCANMAX: TIME = 1 + ERROR;
RSCANMIN: TIME = 1 - ERROR;
```

SRI's SAL Toolset

- Parser
- Simulator
- Symbolic model-checker (BDDs)
- Witness symbolic model-checker
- Bounded model-checker
- Infinite-state bounded model-checker
- Future releases include:
  - Explicit-state model-checker
  - MDD-based symbolic model-checking

All of which are "state-of-the-art"

k-Induction

Please direct your attention to the whiteboard.

Timeout Automata[1] (Semantics)

An *explicit* real-time model.

- Vocabulary:
  - A set of state variables.
  - A *global clock*, $c \in \mathbb{R}^{0\leq}$.
  - A set of *timeout* variables T such that for $t \in T$, $t \in \mathbb{R}^{0\leq}$.
- Construct a transition system $\langle S, S^0, \rightarrow \rangle$:
  - States are mappings of all variables to values.
  - Transitions are either *time transitions* or *discrete transitions*.
    - Time transitions are enabled if the clock is less than all timeouts. Updates clock to least timeout.
    - Discrete transitions are enabled if the clock equals some timeout. Updates state variables and timeouts.

[1] B. Dutertre and M. Sorea. Timed systems in SAL. *SRI TR*, 2004.

Disjunctive Invariants

Even with *k*-induction, getting a sufficiently strong invariant is still hard! *Disjunctive invariants* help. A disjunctive invariant can be built iteratively from the counterexamples returned for the hypothesized invariant being verified.

```
t0:  THEOREM system |-
       G(   (    (phase = Settle)
             AND (rstate = tstate + 1)
             AND (rclk - tclk - TPERIOD > 0)
             AND (tclk + TPERIOD + TSTABLE - rclk > 0))
         OR
            (    (phase = Stable)
             AND (rstate = tstate + 1)
             AND (rclk - tclk - TSETTLE > 0)
             AND (tclk + TPERIOD - rclk > 0)
             AND (rdata = tdata))

                      .
                      .
                      .
```