PS- LTL for Constraint- based Security Protocol Analysis

Sandro Etalle
with Ricardo Corin & Ari Saptawijaya
University of Twente

Outline

- security protocols
- constraint-based verification of security protocols
- contribution PS-LTL
  - specification of security properties
  - verification

The Context

- One of the best protocol verifiers:
- Millen&Shmatikov (CCS'01)
  - Bounded sessions
  - Constraint solving
    - Unbounded message space
  - Perfect crypto, Dolev-Yao

Roles and Protocols

Message 1. $a \rightarrow b : (a, n_a)$
Message 2. $b \rightarrow a : \{(n_a, k_{st})\}_{k_{lt}}$
Message 3. $a \rightarrow b : \{n_a\}_{k_{st}}$

BAN Concrete Andrew Secure RPC

- Roles: sequences of send & recv actions

$$\begin{aligned} \mathrm{init}(A, B, N_A, K_{lt}, K_{st}) &= \langle\ \langle A : (A, N_A) \triangleright B\rangle\ \langle A : \{(N_A, K_{st})\}_{K_{lt}} \triangleleft B\rangle \\ &\quad \langle A : \{N_A\}_{K_{st}} \triangleright B\rangle\ \rangle \\ \mathrm{resp}(A, B, N_A, K_{lt}, K_{st}) &= \langle\ \langle B : (A, N_A) \triangleleft A\rangle\ \langle B : \{(N_A, K_{st})\}_{K_{lt}} \triangleright A\rangle \\ &\quad \langle B : \{N_A\}_{K_{st}} \triangleleft A\rangle\ \rangle \end{aligned}$$

- variables start with uppercase
- Scenario: set of semi-instantiated roles

$$S = \{\mathrm{init}(a, B, na, k_{lt}, K_{st}), \mathrm{resp}(a, b, N_A, k_{lt}, k_{st})\}$$

For IK={}, only solution
N_A-> na, K_{st}-> k_{st}

for IK=k_{lt}, many: e.g.
K_{st},N_A->a

each solution is a
concrete trace:
the symbolic trace
stands for many (∞?)
concrete traces

is a **solution** if intruder can produce r
sing knowledge Kσ.

- symbolic trace contains ***variables***

$$= \langle\ \langle a : (a, na) \triangleright B\rangle\ \langle b : (a, \quad) \triangleleft a\rangle\ \langle b : \{(\quad , k_{st})\}_{k_{lt}} \triangleright a\rangle \\ \langle a : \{(na, \quad t)\}_{k_{lt}} \triangleleft b\rangle \langle a : \{na\}_{K_{st}} \triangleright B\rangle\ \langle b : \{ \quad \}_{k_{st}} \triangleleft a\rangle\ \rangle$$

- and has an associated ***constraint store***:

$$\begin{aligned} cs(\nu, IK) = \{\ & (a, N_A) : IK \cup \{(a, na)\}, \\ & \{(na, K_{st})\}_{k_{lt}} : IK \cup \{(a, na), \{(N_A, k_{st})\}_{k_{lt}}\}, \\ & \{N_A\}_{k_{st}} : IK \cup \{(a, na), \{(N_A, k_{st})\}_{k_{lt}}, \{na\}_{K_{st}} \\ \}\ & \end{aligned}$$

Problem

- Encoding properties in MS
  - **Secrecy**: add to scenario a special role
  - **Authentication**: construct scenario in which a role has no corresponding party
- Problem
  - indirect
  - built-in
  - inflexible

The Solution: $\mathcal{PS}$- LTL

- Based on LTL with past operators
  - similar to NPATRL
- Syntax
  - $p(d_1,\ldots,d_n)$, learn(m)
  - $Y\varphi$, $\varphi_1 S \varphi_2$, $O\varphi$ (= true S φ) , $H\varphi$ (= $\neg O \neg \varphi$)
  - $\neg\varphi$, $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$, $\exists v.\varphi$, $\forall v.\varphi$

UT – seminar 20 12 2006

Decorating Protocols

$$\begin{aligned}
\text{init}(A,B,N_A,K_{lt},K_{st}) = \langle\ & \langle A:(A,N_A) \triangleright B\rangle \langle A:\{(N_A,K_{st})\}_{K_{lt}} \triangleleft B\rangle \\
& \mathbf{run}(\mathbf{A},\mathbf{B},\mathbf{initiator},\mathbf{N_A},\mathbf{K_{lt}},\mathbf{K_{st}}) \\
& \langle A:\{N_A\}_{K_{st}} \triangleright B\rangle \\
& \mathbf{end}(\mathbf{A},\mathbf{B},\mathbf{initiator},\mathbf{N_A},\mathbf{K_{lt}},\mathbf{K_{st}})\ \rangle \\
\text{resp}(A,B,N_A,K_{lt},K_{st}) = \langle\ & \langle B:(A,N_A) \triangleleft A\rangle \\
& \mathbf{run}(\mathbf{B},\mathbf{A},\mathbf{responder},\mathbf{N_A},\mathbf{K_{lt}},\mathbf{K_{st}}) \\
& \langle B:\{(N_A,K_{st})\}_{K_{lt}} \triangleright A\rangle \\
& \langle B:\{N_A\}_{K_{st}} \triangleleft A\rangle \\
& \mathbf{end}(\mathbf{B},\mathbf{A},\mathbf{responder},\mathbf{N_A},\mathbf{K_{lt}},\mathbf{K_{st}})\ \rangle
\end{aligned}$$

UT – seminar 20 12 2006

Properties

- Aliveness
- Non-injective agreement

$$\forall A,B,D1,D2,D3.end(A,B,responder,D1,D2,D3) \rightarrow \\ O\ run(B,A,initiator,D1,D2,D3)$$

- Freshness
- Standard secrecy
- Perfect forward secrecy (DH key agreement):
  - disclosure of long term-keys does not compromise secrecy of earlier exchanged short-term keys

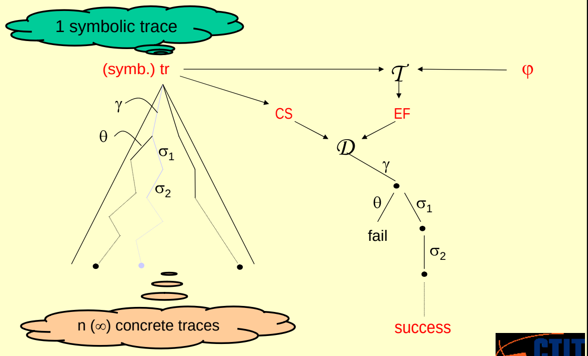
UT – seminar 20 12 2006

Challenge

- Checking a formula φ on a SYMBOLIC trace tr
  - given tr, find σ s.t
  - trσ is a valid concretization of tr
    - (the associated CSσ is solvable)
  - σ falsifies φ
    - We are looking for an attack
- Idea: φ guides the concretization of tr

UT – seminar 20 12 2006

Sketch



UT – seminar 20 12 2006

how we do it

- positive equalities: (tweaked) unification
- positive constraints: MS procedure
- negative constraints:
  - safe approximation
- negative equalities: syntactic check

UT – seminar 20 12 2006

Conclusions

University of Twente
The Netherlands

- Separate the properties from the spec.
- Constraint solving for protocol engineering
  - (we used it successfully in three large case studies: WSN protocol, OSA/Parlay auth protocol and DRM protocol)
- Among the simplest D-Y like approaches
- PS-LTL clarifies the difference between model and requirements

CTIT