Formal Analysis of Security Protocols: Protocol Composition Logic

Ante Đerek

Outline

- Introduction
  - Security protocols
  - Formal analysis methods
- Protocol Composition Logic
  - Intuition
  - Definitions: protocols, logic, proof system
  - Composition theorems

Security Protocols

- Distributed programs
  - Insecure communication channels
  - Cryptography used to achieve goals
- Mission critical
  - SSL: protecting credit card information
  - Kerberos: identity verification and authorization
  - WEP: protecting wireless networks
- Subtle
  - Attack may combine data from many sessions
  - Modeling cryptographic primitives is not easy
  - Many broken protocols and incorrect “proofs”

Complexity: Cryptography Is Hard

"A private-key signature scheme is secure if for every probabilistic polynomial-time oracle machine M, every polynomial p and all sufficiently large n, it holds that the probability of M producing a pair (a, b) such that V(a, b)=1 and a is different from all strings for which the signature has been requested, is less than 1/p(n). Probability is taken over the coin tosses of key-generation, signing and verification algorithms, as well as over the coin tosses of machine M."

[Oded Goldreich, Foundations of Cryptography]

- Assume it away
  - Perfect cryptography, symbolic model (PCL)
- Hide it behind…
  - …correspondence theorems
  - …ideal functionalities
  - …logical axioms and proof rules (CPCL)

Complexity: Goals Not Always Clear

"Two services are provided to bring the IEEE 802.11 functionality in line with wired local area network (LAN) assumptions: authentication and confidentiality. Authentication is used instead of the wired media physical connection. Privacy Data confidentiality is used to provide the confidential aspects of closed wired media."

[IEEE Standard 802.11i]

- Exactly specify the security requirements
  - Properties holding in face of an attack (CPCL)
  - Simulation relation
  - Set of undesirable runs (PCL)
- Use a formal language (PCL, CPCL)
  - Unambiguous

Complexity: Attackers Are Powerful

"This document makes several traditional assumptions, including that attackers have substantial computational resources and cannot obtain secret information from sources outside the protocol. Attackers are assumed to have the ability to capture, modify, delete, replay, and otherwise tamper with messages sent over the communication channel."

[The SSL Protocol Version 3.0, Internet Draft]

- Case-by-case hand proofs…
  - …demand considerable effort and skill
  - …are error-prone
- Formal methods
  - Model checking
  - Machine verifiable proofs (PCL)

Complexity: Protocols Are Complex

"IEEE 802.11 defines two authentication methods: Open System authentication and Shared Key authentication. Open System authentication admits any STA to the DS. Shared Key authentication relies on WEP to demonstrate knowledge of a WEP encryption key. The IEEE 802.11 authentication mechanism also allows definition of new authentication methods. An RSNA also supports authentication based on IEEE 802.1X, or preshared keys (PSKs). IEEE 802.1X authentication utilizes the EAP to authenticate STAs and the AS with one another."

[IEEE Standard 802.11i – Page 25/190]

- Security properties to not compose in general
  - Components may degrade each other's security
- Divide and conquer
  - Universal composability
  - Assume-guarantee (PCL)

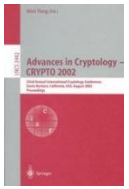
Formal Methods in Protocol Analysis

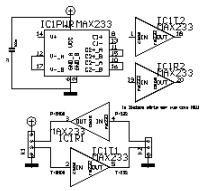
- Symbolic model
  - Ad hoc methods [Strands]
  - Model checking [Murphi]
  - Theorem proving [Isabelle]
- Complexity theoretic model
  - Passive adversary [AR2002]
  - Correspondence theorems [MW2004]
  - Simulation framework [BPW2003, Can2001]

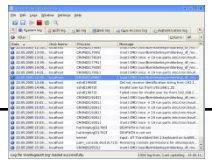
Protocol Composition Logic: Outline

- Intuition
- Basic components
  - Protocol programming language
  - Logic, syntax and semantics
  - Proof system, soundness
- Composition theorems

Intuition







Alice's reasoning based on…

…actions she performed: I have received $SIG_B\{m, n, A\}$, therefore …

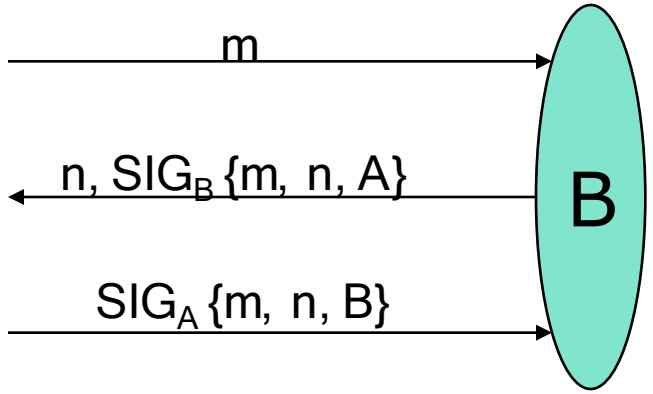
…cryptographic properties: … Bob must have produced $SIG_B\{m, n, A\}$, therefore …

…protocol specification: … Bob must have received m, thinking it was from A, …

…information theory, concurrency, first order logic: … Bob is authenticated

Protocol Programming Language

```
Init(A, B) = [
    new m;
    send (A, B, m);
    receive (B, A, n, r);
    verify r, (m, n, A), B;
    s := sign (m, n, B), A;
    send (A, B, s);
    ]A
```



Terms: constants, variables, pairing

Internal actions: nonces, signing, matching, …

Communication steps: sending and receiving

Execution Model

- A protocol is a set of roles
  - Initiator, Responder, Server
- Initial configuration
  - Each honest party assigned one or more roles
  - Arbitrary attacker program
- Execution
  - Programs react
  - Communication and propagation via substitution
- A run
  - Sequence of actions

Protocol Logic

$\forall\ \forall\ X.\ (Has(X, m) \supset X = A)$

[
send (A, B, m);
receive (B, A, n, r);
verify r, (m, n, A), B;
s := sign (m, n, B), A;
send (A, B, s);
]A

$Honest(B) \supset (Send(A, (A, B, m)) < Send(B, (B, A, SIG_B\{m, n, A\}) \wedge$
$Send(B, (B, A, Sig\_B\{m, n, A\}) < Receive(A, (B, A, SIG_B\{m, n, A\}))$

If, starting from a state where m is secret, A completes the protocol steps, then, in the resulting state, the authentication property holds, assuming that B is honest.

Actions: Send, Receive, Verify, …

Knowledge: Has, Computes, …

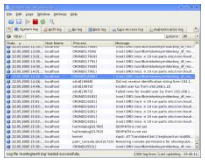
Temporal ordering operator

Honesty assumption

Modal operator

Proof System

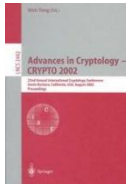
- A set of axioms and proof rules
- Sample axioms:



AA1 $[\text{receive m}]_X \text{ Receive}(X, m)$

AN2 $[\text{new x}]_X \forall Y \text{ Has}(Y, m) \supset Y = X$

TUP $\text{Has}(A, (m,n)) \supset \text{Has}(A, m) \wedge \text{Has}(A, n)$

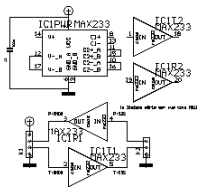


SIG $\text{Verify}(X, \text{SIG}_Y\{m\}) \wedge \text{Honest}(Y) \supset \exists Y' \text{ Sign}(Y', m)$

ENC $\text{Honest}(X) \wedge \text{Decrypt}(Y, \text{ENC}_X\{m\}) \supset X=Y$

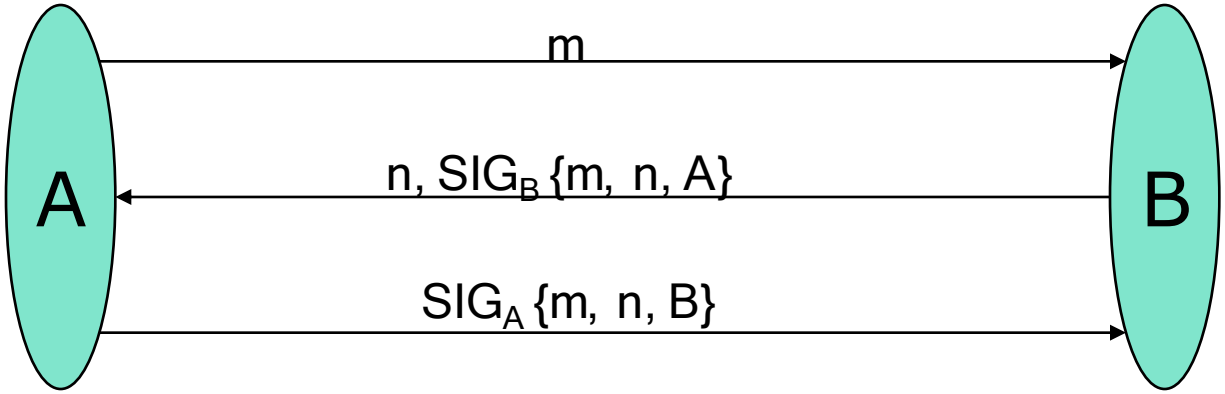
Protocol Invariants – Honesty Rule

- Honesty rule: induction over protocol steps
  - Resulting formulas hold in all states
- Invariants: properties preserved by all steps



$$\textbf{HON}\ \frac{[\,]_X\,\varphi \qquad \forall B \in \mathrm{ProtocolSteps}(Q).\ \varphi\,[B]_X\,\varphi}{Q \mid\!\!- \mathrm{Honest}(X) \supset \varphi}$$

Example



Formal proof stages:

$[\text{Init}]_A\ \text{Receive}(A, (B, A, \text{SIG}_B\{m, n, A\})) \wedge \text{Verify}(A, \text{SIG}_B\{m, n, A\})$

$[\text{Init}]_A\ \text{Sign}(B, (m, n, A))$

$[\text{Init}]_A\ \text{Honest}(B) \supset \text{Receive}(B, (A, B, m)) < \text{Send}(B, (n, \text{SIG}_B\{m, n, A\}))$

$[\text{Init}]_A\ \text{Honest}(B) \supset$ *authentication*

Soundness Theorem

Theorem: Every statement provable in the proof system is a valid formula of the logic.

- Proof by induction
- Hence any provable property holds…
  - …in all runs
  - …regardless of the attacker actions
  - …for any number of participants and sessions

Protocol Composition

- Composition goals
  - Ensure noninterference (parallel composition)
  - Accumulate properties (sequential composition)
- Composition theorems: specify sufficient conditions for non interference
  - Intuition: honesty rule is the only protocol specific proof step, hence it is sufficient to check that protocols satisfy each other's invariants
- Combining components: sequencing rule

$$\textbf{SEQ}\ \frac{\varphi\ [P\ ]_X\ \psi \qquad \psi\ [P]_X\ \alpha}{\varphi\ [P;\ P']_X\ \alpha}$$

PCL: Summary

- Formal proofs sound in the symbolic model
  - No explicit reasoning about the attacker
  - Authentication, secrecy and other properties
  - Encryption, signatures, hash, Diffie-Hellman
  - Composition theorems
- Case studies
  - IKE family [DDMP2003, DDMP2004]
  - 802.11i [HSDDM2005]
  - Kerberos [RDDMS2006]
  - Contract signing protocols [BDDMT2005]
  - GDOI [MP2004]
- Tool support
  - Implementation in Isabelle [Stanford]
  - Protocol Derivation Assistant [Kestrel]

There is more

- Computational PCL
  - Attempt to carry over results to the complexity theoretic model
- Secrecy framework
  - Challenge: Secrecy properties not inductive by nature
- …

Questions?