Logic for Computer Security Protocols

Ante Derek

Outline

• Last lecture
  • Floyd-Hoare logic of programs
  • BAN logic
• Today
  • Compositional Logic for Proving Security Properties of Protocols

Intuition

• Reason about local information
  • I chose a new number
  • I sent it out encrypted
  • I received it decrypted
  • Therefore: someone decrypted it
• Incorporate knowledge about protocol
  • Protocol: Server only sends m if it received m'
  • If server not corrupt and I receive m signed by server, then server received m'

Intuition: Picture

Consumption

Private Data

Honest Principals, Attacker

Example: Challenge-Response

A

m, A

n, sigB (m, n, A)

sigA (m, n, B)

B

Alice: if Bob is honest, then:
  • only Bob can generate his signature [protocol independent]
  • if Bob generates a signature of the form sigB(m, n, A),
    - he sends it as part of msg2 of the protocol and
    - he must have received msg1 from Alice [protocol specific]
  • Alice deduces: Received (B, msg1) ∧ Sent (B, msg2)

Formalizing the Approach

• Language for protocol description
  • Arrows-and-messages are informal.
• Protocol Semantics
  • How does the protocol execute?
• Protocol logic
  • Stating security properties.
• Proof system
  • Formally proving security properties.
Cords

- "protocol programming language"
- A protocol is described by specifying a "program" for each role
  - Server = [receive x; new n; send (x, n)]

- Building blocks
  - Terms
    - names, nonces, keys, encryption, ...
  - Actions
    - send, receive, pattern match, ...

Terms

\[ t ::= c \quad \text{constant term} \]
\[ x \quad \text{variable} \]
\[ N \quad \text{name} \]
\[ K \quad \text{key} \]
\[ t, t \quad \text{tupling} \]
\[ \text{sig}_K \{ t \} \quad \text{signature} \]
\[ \text{enc}_K \{ t \} \quad \text{encryption} \]

Example: \( x, \text{sig}_B \{ m, x, A \} \) is a term

Actions

- send \( t \): send a term \( t \)
- receive \( x \): receive a term into variable \( x \)
- match \( t/p(x) \): match term \( t \) against \( p(x) \)

- A Cord is just a sequence of actions
- Notation:
  - we often omit match actions
  - receive \( \text{sig}_B \{ A, n \} \) = receive \( x \); match \( x/\text{sig}_B \{ A, n \} \)

Challenge-Response as Cords

\[ \text{InitCR} (A, X) = \begin{cases} 
\text{new } m; \\
\text{send } A, X, \{ m, A \}; \hspace{1cm} \\
\text{receive } X, A, \{ x, \text{sig}_X \{ m, x, A \} \}; \\
\text{send } A, X, \text{sig}_A \{ m, x, X \}; 
\end{cases} \]

\[ \text{RespCR} (B) = \begin{cases} 
\text{receive } Y, B, \{ y, Y \}; \\
\text{new } n; \\
\text{send } B, Y, \{ n, \text{sig}_B \{ y, n, Y \} \}; \\
\text{receive } Y, B, \text{sig}_Y \{ y, n, B \}; 
\end{cases} \]

Cord Spaces

- Cord space is a multiset of cords
- Cords may react
  - via communication
  - via internal actions
- Sample reaction steps:
  - Communication:
    - \( [ S; \text{send } t; S'] \circ [ T; \text{receive } x; T'] \Rightarrow [ S; S'] \circ [ T; T'(t/x) ] \)
  - Matching:
    - \( [ S; \text{match } p(t)/p(x); S'] \Rightarrow [ S; S'(t/x) ] \)

Execution Model

- Initial configuration
  - Protocol is a finite set of roles
  - Set of principals and keys
  - Assignment of \( \geq 1 \) role to each principal
- Run
  - A new \( x \), send \( \{ x \}_A \)
  - B receive \( \{ x \}_B \), receive \( \{ z \}_B \)
  - C new \( z \), send \( \{ z \}_C \)

Position in run
Logical assertions

- Modal operator
  \[ [\text{actions}] P \phi \] - after actions, \( P \) reasons \( \phi \)
- Predicates in \( \phi \)
  - \( \text{Send}(X, m) \) - principal \( X \) sent message \( m \)
  - \( \text{Receive}(X, m) \) - principal \( X \) received message \( m \)
  - \( \text{Verify}(X, m) \) - \( X \) verified signature \( m \)
  - \( \text{Has}(X, m) \) - \( X \) created \( m \) or received msg containing \( m \) and has keys to extract \( m \) from msg
  - \( \text{Honest}(X) \) - \( X \) follows rules of protocol

Formulas true at a position in run

- Action formulas
  \[ a ::= \text{Send}(P, m) \mid \text{Receive}(P, m) \mid \text{New}(P, t) \]
  \[ \mid \text{Decrypt}(P, t) \mid \text{Verify}(P, t) \]
- Formulas
  \[ \phi ::= a \mid \text{Has}(P, t) \mid \text{Fresh}(P, t) \mid \text{Honest}(N) \]
  \[ \mid \text{Contains}(t_1, t_2) \mid \neg \phi \mid \phi_1 \land \phi_2 \mid \exists x \phi \]
  \[ \mid \Box \phi \mid \Diamond \phi \]
- Example
  \[ \text{After}(a, b) = \Diamond (b \land \Box \Diamond a) \]

Semantics

- Protocol \( Q \)
  \[ \text{Defines set of roles (e.g., initiator, responder)} \]
  \[ \text{Run } R \text{ of } Q \text{ is sequence of actions by principals following rules, plus attacker} \]
- Satisfaction
  \[ Q, R \models [\text{actions}] P \phi \]
  \[ \text{Some role of } P \text{ in } R \text{ does exactly actions and } \phi \text{ is true in state after actions completed} \]
  \[ Q \models [\text{actions}] P \phi \]
  \[ Q, R \models [\text{actions}] P \phi \text{ for all runs } R \text{ of } Q \]

Security Properties

- Authentication for Initiator
  \[ \text{CR } \models [\text{InitCR}(A, B)] A \text{ Honest}(B) \Rightarrow \]
  \[ \text{ActionsInOrder(} \]
  \[ \text{Send}(A, \{A, B, m\}), \]
  \[ \text{Receive}(B, \{A, B, m\}), \]
  \[ \text{Send}(B, \{B, A, \{n, \text{sig}_B \{m, n, A\}\}\}), \]
  \[ \text{Receive}(A, \{B, A, \{n, \text{sig}_B \{m, n, A\}\}\})\]
  \[ ) \]

Security Properties

- Shared secret
  \[ \text{NS } \models [\text{InitNS}(A, B)] A \text{ Honest}(B) \Rightarrow \]
  \[ (\text{Has}(X, m) \Rightarrow X=A \land X=B) \]

Proof System

- Goal: formally prove properties
- Axioms
  - Simple formulas provable by hand
- Inference rules
  - Proof steps
- Theorem
  - Formula obtained from axioms by application of inference rules
Sample axioms about actions

- New data
  - $[\text{new } x]_p \cdot \text{Has}(P,x)$
  - $[\text{new } x]_p \cdot \text{Has}(Y,x) \supset Y=P$

- Actions
  - $[\text{send } m]_p \cdot \text{Send}(P,m)$

- Knowledge
  - $[\text{receive } m]_p \cdot \text{Has}(P,m)$

- Verify
  - $[\text{match } x/\text{sig}_X(m)]_p \cdot \text{Verify}(P,m)$

Reasoning about knowledge

- Pairing
  - $\text{Has}(X, (m,n)) \supset \text{Has}(X, m) \land \text{Has}(X, n)$

- Encryption
  - $\text{Has}(X, \text{enc}_c(m)) \land \text{Has}(X, K^{-1}) \supset \text{Has}(X, m)$

Encryption and signature

- Public key encryption
  - $\text{Honest}(X) \land \Diamond \text{Decrypt}(Y, \text{enc}_X(m)) \supset X=Y$

- Signature
  - $\text{Honest}(X) \land \Diamond \text{Verify}(Y, \text{sig}_X(m)) \supset$
    $\exists m’(\Diamond \text{Send}(X, m’) \land \text{Contains}(m’, \text{sig}_X(m)))$

Sample inference rules

- Preservation rules
  - $[\text{actions }]_p \cdot \text{Has}(X, t)$
    $[\text{actions; action }]_p \cdot \text{Has}(X, t)$

- Generic rules
  - $[\text{actions }]_p \cdot \phi$
    $[\text{actions }]_p \cdot \phi$
    $[\text{actions }]_p \cdot \phi \land \phi$

Bidding conventions (motivation)

- Blackwood response to 4NT
  - $5\spadesuit: 0$ or $4$ aces
  - $5\heartsuit: 1$ ace
  - $5\diamondsuit: 2$ aces
  - $5\clubsuit: 3$ aces

- Reasoning
  - If my partner is following Blackwood,
    then if she bid $5\spadesuit$, she must have $2$ aces

Honesty rule (rule scheme)

$\forall \text{roles } R \text{ of } Q. \forall \text{ initial segments } A \subseteq R.$
$Q \vdash [A]_x \phi$
$Q \vdash \text{Honest}(X) \supset \phi$

- This is a finitary rule:
  - Typical protocol has 2-3 roles
  - Typical role has 1-3 receives
  - Only need to consider $A$ waiting to receive
Honesty rule (example use)

\[ \forall \text{roles } R \text{ of } Q, \forall \text{ initial segments } A \subseteq R. \]
\[ Q \vdash [A] \phi \]
\[ Q \vdash \text{Honest}(X) \supset \phi \]

- Example use:
  - If Y receives a message from X, and \text{Honest}(X) \supset (\text{Sent}(X,m) \supset \text{Received}(X,m'))
  then Y can conclude \text{Honest}(X) \supset \text{Received}(X,m'))

Correctness of CR

\[ \text{InitCR}(A, X) = \{ \]
new m;
send A, X, \{m, A\};
receive X, A, \{x, sig_x(m, x, A)\};
send A, X, \{x, \text{sig}_A(m, x, X)\};
\]
\[ \text{RespCR}(B) = \{ \]
new n;
send B, Y, \{n, \text{sig}_B(y, n, Y)\};
receive Y, B, \{n, \text{sig}_B(y, n, B)\};
\]

Correctness of CR – step 1

1. A reasons about it's own actions
\[ CR \vdash [\text{InitCR}(A, B)]_A \sim \text{Verify}(A, \text{sig}_B(m, n, A)) \]

Correctness of CR – step 2

2. Properties of signatures
\[ CR \vdash [\text{InitCR}(A, B)]_A \supset \text{Honest}(B) \supset \text{ActionsInOrder}(\]
Send(A, \{A,B,m\}),
Receive(B, \{A,B,m\}),
Send(B, \{B,A,n, \text{sig}_B(m, n, A)\}),
Receive(A, \{B,A,n, \text{sig}_B(m, n, A)\})\]

Correctness of CR – Honesty

Honesty invariant
\[ CR \vdash \text{Honest}(X) \land \sim \text{Send}(X, m') \land \text{Contains}(m', \text{sig}_y(x, x, Y)) \land \sim \text{New}(X, y) \supset m = X, Y, \{x, \text{sig}_A(y, x, Y)\} \land \text{Receive}(X, \{Y, X, \{y, Y\}\}) \]

Correctness of CR – step 3

3. Use Honesty rule
\[ CR \vdash [\text{InitCR}(A, B)]_A \supset \text{Honest}(B) \supset \text{Receive}(B, \{A,B,m\}) \]
Correctness of CR - step 4

4. Use properties of nonces for temporal ordering

\[ CR \vdash [\text{InitCR}(A, B)] A \text{ Honest(B)} \supset \text{Auth} \]

Complete proof

We have a proof. So what?

- Soundness Theorem:
  - if \( Q \vdash \phi \) then \( Q \models \phi \)
  - If \( \phi \) is a theorem then \( \phi \) is a valid formula

- \( \phi \) holds in any step in any run of protocol \( Q \)
  - Unbounded number of participants
  - Dolev-Yao intruder

Correctness of WCR - step 1

Weak Challenge-Response

Correctness of WCR - step 2

Properties of signatures

\[ CR \vdash [\text{InitCR}(A, B)] A \text{ Honest(B)} \supset \exists m' (\Diamond \text{Send}(B, m') \land \text{Contains}(m', \text{sig}_B (m, n, A))) \]
Correctness of WCR - Honesty

\[
\text{InitWCR}(A, X) = \{
\begin{align*}
\text{new } m; \\
\text{send } A, X, (m); \\
\text{receive } X, A, (x, \text{sig}_X(m, x)); \\
\text{send } A, X, \text{sig}_X(m, x);
\end{align*}
\}
\]

Honesty invariant

\[
\text{CR} \vdash \text{Honest}(X) \land \\
\Diamond \text{Send}(X, m) \land \text{Contains}(m, \text{sig}_X(y, x)) \land \neg \Diamond \text{New}(X, y) \Rightarrow \\
m = X, (x, \text{sig}_X(y, x)) \land \Diamond \text{Receive}(X, (Z, X, (y, Z)))
\]

Correctness of WCR - step 3

\[
\text{RespWCR}(B) = \{
\begin{align*}
\text{receive } Y, B, (y); \\
\text{new } n; \\
\text{send } B, Y, (n, \text{sig}_B(y, n)); \\
\text{receive } Y, B, \text{sig}_Y(y, n);
\end{align*}
\}
\]

3. Use Honesty rule

\[
\text{WCR} \vdash [\text{InitWCR}(A, B)] A \text{Honest}(B) \Rightarrow \\
\Diamond \text{Receive}(B, (Z, B, m)),
\]

Result

- WCR does not have the strong authentication property for the initiator
- Counterexample
  - Intruder can forge senders and receivers identity in first two messages
    - \( A \rightarrow X(B), m \)
    - \( X(C) \rightarrow B, m \)
    - \( B \rightarrow X(C), n, \text{sig}_B(m, n) \)
    - \( X(B) \rightarrow A, n, \text{sig}_B(m, n) \)

Benchmarks

- Can prove authentication for CR
- Proof fails for WCR
- Can prove repaired NSL protocol
- Proof fails for original NS protocol
- Proof fails for a variant of GDOI protocol (C. Meadows, D. Pavlovic)

Extensions

- Add Diffie-Hellman primitive
  - Can prove authentication and secrecy for key exchange protocols (STS, ISO-97898-3)
- Add symmetric encryption and hashing
  - Can prove authentication for ISO-9798-2, SKID3

Derivation system

- Protocol derivation
  - Build security protocols by combining parts from standard sub-protocols
- Proof of correctness
  - Prove protocols correct using logic that follows steps of derivation
- Reuse proofs
ISO-9798-3 Key Exchange

- Authentication
  - Do we need to prove it from scratch?
- Shared secret: $g^{ab}$

Abstract challenge response

- Free variables $m$ and $n$ instead of nonces
- Modal form: $\phi [\text{actions}] \psi
- precondition: $\text{Fresh}(A, m)$
- actions: $[\text{InitACR}]_A$
- postcondition: $\text{Honest}(B) \supset \text{Authentication}$
- Secrecy is proved from properties of Diffie-Hellman

Parallel protocol composition

- Assume that agents run both CR and NSL using same public/private keys
  - Is authentication property preserved?
- Honesty rule is only protocol specific step in the proof system
  - Properties are preserved if the new protocol satisfies honesty invariants

Combining protocols

- $\Gamma \vdash \text{Honest}(X) \supset \ldots$
- $\Gamma' \vdash \text{CRAuthentication}$
- $\Gamma \cup \Gamma' \vdash \text{CRAuthentication} \land \text{NSLAuthentication}$
- $\Gamma \cup \Gamma' \vdash \text{CR} \land \text{NSL}
- $\Gamma \cup \Gamma' \vdash \text{CRAuthentication} \land \text{NSLAuthentication}$

Current work

- Formalize protocol refinements and transformations
- Automate proofs