Protocol Derivation Assistant

Dusko Pavlovic
and Matthias Anlauff
Kestrel Institute, Palo Alto CA
Problem

- Networks are complex
  - Regulated by protocols
- Protocols are complex
  - Require incremental approach
Problem

... but

- security is not preserved under
  - refinement
  - composition
Solution

- annotate processes by properties
  - “distributed hoare logic”
- develop
  - processes and
  - properties in parallel
Protocol Derivation System

Protocols
- components
- refinements
- transformations

Proofs
- axioms
- proof rules
- proof transformations

• derivation patterns

“truth is just another security property”
Protocol Derivation Assistant

- protocol development
- instance checking
- code generation
- property specification
- distributed reasoning
- certificate generation

- library, taxonomy
- peer-to-peer exchange
Outline

1. Protocols
2. Protocol Derivations
3. Protocol Derivation Assistant
Running example: GDoI
Running example: GDoI
Running example: GDoI

- IPSec [IETF RFC 2400-2410]

- ISAKMP [IETF RFC 2408]
  - distributions: http://web.mit.edu/network/isakmp/
  - implementations: DoD, Cisco

- GDoI [IETF RFC 3547, July 2003]
  - doc.: http://www.networksorcery.com/enp/protocol/gdoi.htm
  - carefully designed: seven internet drafts
  - formally verified and corrected

analyzed in joint work with Cathy Meadows
Group Domain of Interpretation
Group Domain of Interpretation


Alice
Group Domain of Interpretation
Group Domain of Interpretation

$\text{GDoI}[A, B](H, S)$
Group Domain of Interpretation

GDoI[A,B](H,S)
Group Domain of Interpretation

\[ \text{GDol}[A,B](H,S) \]
Group Domain of Interpretation

GDoI[A,B](H,S)
Group Domain of Interpretation
Group Domain of Interpretation

\[ A \nu \times \nu B \]

\[ \text{chal}[A,B], \text{resp}[A,B], \text{resp}[B,A] \]

key

\[ \text{fresh}, \text{fresh}, \text{fresh} \]
Group Domain of Interpretation
Group Domain of Interpretation

\[\nu x, x, H_{AB}(x) \]

\[\nu y, \text{resp}[A,B], \text{chal}[B,A], \text{resp}[B,A], \text{key}, \text{fresh} \]
Group Domain of Interpretation
Group Domain of Interpretation

\[ \sum^z = C^z, S^z(x, y) \]
Group Domain of Interpretation

\[ \sum^Z = C^Z, S^Z(x, y) \]
Group Domain of Interpretation

\[ \Sigma^2 = C^2, S^2(x, y) \]
Group Domain of Interpretation

\[ \nu x, H^{AB}(x) \]
\[ \nu y, H^{BA}(x, y) \]
\[ H^{AB}(y, \sum^{A'), \sum^{A'}) - k, H^{BA}(k, x, \sum^{B'), \sum^{B'}) \]

\[ \sum^{Z} = C^{Z}, S^{Z}(x, y) \]
Group Domain of Interpretation

\[ \nu_k A \equiv B \]

\[ \nu \nu x \equiv \nu y \]

\[ \nu x, H^{AB}(x) \rightarrow \nu y, H^{BA}(x,y) \rightarrow H^{AB}(y) \rightarrow k H^{BA}(k,x) \rightarrow \]

\[ \nu \nu x \equiv \nu y \rightarrow x, x, H^{AB}(x) \rightarrow y, H^{BA}(x,y) \rightarrow H^{AB}(y) \rightarrow k H^{BA}(k,x) \rightarrow \]

\[ \nu x \rightarrow x, S^{A^'}(x,y) \rightarrow \nu y, S^{B^'}(x,y) \rightarrow \]
Group Domain of Interpretation
Group Domain of Interpretation
Group Domain of Interpretation
Group Domain of Interpretation

\[ \nu x \quad x \quad H^{BA}(x) \quad y \quad H^{AB}(y) \]

\[ A \quad B \quad A \quad B \]
Group Domain of Interpretation

\[ \nu, H^{AB}(x, y) \]
\[ \nu y, H^{BA}(x, y) \]
\[ H^{AB}(y) \]
\[ k, H^{BA}(k, x) \]
Group Domain of Interpretation

\[
\nu_x A
\]

\[
\nu_y A
\]

\[
\nu_x B
\]

\[
\nu_y B
\]

\[
\nu_x H^{AB}(x)
\]

\[
\nu_y H^{BA}(x,y)
\]

\[
H^{AB}(y)
\]

\[
H^{BA}(k,x)
\]

\[
S'(x,y)
\]
Group Domain of Interpretation
Outline

1. Protocols
2. Protocol Derivations
3. Protocol Derivation Assistant
1. Protocols
2. Protocol Derivations
3. Protocol Derivation Assistant
PDA Architecture

Graphical User Interface

Graphical Editor

Derivation Browser

Protocol Derivation Engine

Syntax Checker

Protocol Instance Engine

Protocol Composition & Refinement Engine

Protocol Library Manager

Code Generator
PDA Architecture

Graphical User Interface

- Graphical Editor
- Derivation Browser

Protocol Derivation Engine

- Syntax Checker
- Protocol Instance Engine
- Protocol Composition & Refinement Engine

Security Derivation Engine

- Protocol Library Manager
- Code Generator

Derivation

Browser
Outline

1. Protocols
2. Protocol Derivations
3. Protocol Derivation Assistant
PDA DEMO

Deriving attacks on GDol.
GDolv2 proposal.
Summary

- PDA supports:
  - protocol specifications
    - process representations
    - property axioms
    - distributed reasoning
  - composition and refinement of distributed processes
  - evolving taxonomies of protocols and properties
  - rudimentary code generation

- to do:
  - automate property derivations
  - code and certificate generation
  - integrate other tools
  - add crypto
Papers

- An encapsulated authentication logic for reasoning about key distribution protocols
  - with I. Cervesato and C. Meadows, submitted

- Deriving, attacking and defending GDOI
  - with C. Meadows, Proceedings of ESORICS 2004 (Springer LNCS)

- A derivational system and compositional logic for security protocols

- Abstraction and refinement in protocol derivation
Papers

- Secure protocol composition
  - with A. Datta and A. Derek and J. Mitchell, Proceedings of MFPS 2003 (ELNCS); ext. abstract in FMCS 2003 (ACM)
- Derivation system for security protocols and its logical formalization
- Compositional logic for protocol correctness
- Composition and refinement of behavioral specifications
  - with D. Smith, ASE 2002 (IEEE)

www.kestrel.edu/users/pavlovic/
PDA web site

www.kestrel.edu/software/pda/