F. Javier Thayer

Models for Protocols
Why Security Protocols?

Interlocuteurs (e.g., buyers and sellers) are rarely
– Eavesdropping capabilities
– Cryptographic skill
– Computational resources

Aggression?

Money, Fun, Prestige, Boredom, Mility

Motivation

characteristics such as

Adversaries with unknown or unspecified
Potentially enormous numbers of agents

+
Goals for this Presentation

- Conclusion
  - Tolerance for protocol failure:
    - Application to pure authentication protocol.
    - Bundle Random Variables.
  - Stochastic models
    - Qualify formal
  - Formal correctness proof techniques
    - Propose and discuss threat models.
    - Real protocols are horrendous.

Why?
- What does protocol correctness mean?
  - Models, Strand space model.
  - What’s a security protocol. Examples.
Types of Protocols

- Goals: Manage key longevity, revocation etc.
  - Agents: Programs.
- Key management protocols:
  - Goal: Establish session encryption key.
  - Agents: Programs.
- Key exchange protocols:
  - Goal: Establish key.
  - Agents: Programs.
- Identity of agents:
  - Goal: Establish identity to mutually establish identity
  - Agents: Individuals or programs.
- Authentication protocols:
  - Goal: Reliable communication.
  - Agents: Application programs.
  - Networking protocols such as TCP.
Protocol Analysis

- Hostile attacks
- Failures
- Timing delays, communication failures, agent
- What can go wrong?
  - Interlocutors attempt to prove they possess a secret or have received a message
  - Use primitives such as cryptography or hash

How do security protocols work?
- Methods of authenticating interlocutors
- Exchange of session keys for encryption
- Goals of security protocols
Approaches

- Allows formulation of probabilistic statements.
  - Kolmogorov, A. (1933)

- Techniques applicable to a variety of protocols.
  - Explicitly formulate security goals.

- What is authentication, secrecy?
  - J. T. May, JCS 1999

- Transactions in protocols (J. Gutman, J. Herzog, F. Strnad, 1998)
  - Models and graphical structure of interactions
  - Burroughs, Abdali, Needham (1989)
  - Dolev, Yao, Karp (1982)

Classic works based on logical formalisms

Many security protocols are in use currently.

Approaches
Only A has private key $K_{-1}$, $N_1$, $N_2$ can be used to compose a session key.

$A$ proves to $B$ that she received $N_2$.

Only $B$ has private key $K_{-1}$.

$B$ proves to $A$ that he received nonce $N_1$.

Authenticated Example

Needham-Schroeder public key cryptography.
Protocol "Flaw"
Example: Otway-Rees

and $S$ is long-term shared key between $A$. Key $K_{AVS}$ is mutually authenticate, distribute session.

Goal: Mutually authenticate, distribute session.
<table>
<thead>
<tr>
<th>Node</th>
<th>Fresh Value</th>
<th>Initiator</th>
<th>Responder</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N^q_1$</td>
<td>${q_{N^b}^N} -$</td>
<td>${q_{N^a}^N} +$</td>
<td>${q_{N^b}^N} -$</td>
</tr>
</tbody>
</table>

**Needham-Schroeder (Modified)**

**Parametric Form for Strands I**
<table>
<thead>
<tr>
<th>Node Schema</th>
<th>Fresh Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiator</td>
<td></td>
</tr>
<tr>
<td>Responder</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td></td>
</tr>
</tbody>
</table>

**Parametric Form for Strands II**

<table>
<thead>
<tr>
<th>Node Schema</th>
<th>Fresh Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiator</td>
<td></td>
</tr>
<tr>
<td>Responder</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Node Schema</th>
<th>Fresh Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiator</td>
<td></td>
</tr>
<tr>
<td>Responder</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td></td>
</tr>
</tbody>
</table>

**Node Schema**

<table>
<thead>
<tr>
<th>Node Schema</th>
<th>Fresh Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiator</td>
<td></td>
</tr>
<tr>
<td>Responder</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td></td>
</tr>
</tbody>
</table>

**Parametric Form for Strands II**

<table>
<thead>
<tr>
<th>Node Schema</th>
<th>Fresh Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiator</td>
<td></td>
</tr>
<tr>
<td>Responder</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td></td>
</tr>
</tbody>
</table>

**Node Schema**

<table>
<thead>
<tr>
<th>Node Schema</th>
<th>Fresh Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiator</td>
<td></td>
</tr>
<tr>
<td>Responder</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td></td>
</tr>
</tbody>
</table>

**Parametric Form for Strands II**

<table>
<thead>
<tr>
<th>Node Schema</th>
<th>Fresh Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiator</td>
<td></td>
</tr>
<tr>
<td>Responder</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td></td>
</tr>
</tbody>
</table>
The assumptions of the general protocol are as follows:

- Regular principals have only know what they send, they are sent.
- Adversary cannot cause messages to be received before.
- Adversary cannot violate laws of physics:
  - Lots of guessing.
  - Apply relations in cryptography.
- Encrypt or decrypt messages using keys known from the start.
- Cardinality:
  - Cardinal and repeating other messages or distinct.
  - Replace messages by concatenating, decomposing.
- Intercept and examine all messages.
- Adversary can communicate medium controlled by hostile agent.
on a strand

\[ \text{in} \quad \text{immediately precedes} \quad \text{in} \]

\[ \text{m} \quad \text{m} \]

\[ \text{in} \quad \text{in} \quad \text{in} \quad \text{in} \quad \text{in} \quad \text{in} \]

\[ \text{in} \quad \text{in} \quad \text{in} \quad \text{in} \quad \text{in} \quad \text{in} \]

\[ \text{two kinds of arrows} \]

\[ \forall \quad \forall \]

\[ \text{run} \]

Strand: One principal's experience of one

- Nodes \( N \) partitioned into strands \( X \).

\[ \forall \text{strand space over} \quad A \]

\[ \text{transmission of} \quad t \quad \text{reception of} \quad t \quad \text{is} \quad t \]

\[ \forall \text{possible messages in a protocol.} \]

Strand Spaces
algebra.

- Prevent failures based on protocol structure,
  which cannot be linked to $\mathcal{A}$.
- Protocol failures may exploit relations in $\mathcal{A}$.

For $\mathcal{A}$ over $\mathcal{A}$, then the same protocol property fails.

- A protocol property fails for strand spaces $\mathcal{A}$.

\[ \forall : \mathcal{A} \to \mathcal{A}. \]

Any encryption algebra $\mathcal{A}$ has a free “cover” titles of the form $\mathcal{A} \{ p \} = \mathcal{R} \{ q \}$.

Real encryption algebra has many non-trivial identity.

\[ \forall : \mathcal{R} \times \mathcal{A} \to \mathcal{A} \]

Encryption concatenation $\mathcal{E} : \mathcal{A} \times \mathcal{A} \to \mathcal{A} \]

Message Algebra
- Encryption, decryption
- Concatenation, decomposition, repetition,

strands for
pentractor: In simplest model, represented by short

- Usually given in parametric form.

server)
roles in protocol (e.g., initiator, responder,
protocol
protocol

kinds of regular (non-pentractor) strands de-

strands
A bundle is a finite subgraph $\mathcal{G}$ of $(\mathcal{N},\rightarrow)$.
A Bundle
A Uniqueness Property

\[ a_N \neq a_N \]

Assuming:

\[ \forall x (K^x \{ B_N \}) + \]

Find:

\[ \forall x (K^x \{ B \land \neg N \}) - \]

Given an NSL bundle:

\[ \forall x (K^x \{ \neg N \}) + \]

\[ \forall x (K^x \{ B \land \neg N \}) - \]
Other Examples of Flaws

Infiltrated Otway-Rees Bundle:
Summary

- General solution of corectness problem (assuming fitness using authentication tests.
- Emphasizes importance of protocol specifications.
- Suggests singular situations/flows.
- Semantics of protocol correctness.
- Geometric representation useful.

Follow from experience of one principle.

Strand Spaces: Emphasizes global properties that
exists

• means the test shows this regular node

J. Gutman, F. J. Thayer, TCS 2001

Authentication Test
Limitations

- No relations between cryptography and the protocol.
- Potentially useful for taking signatures.
- Adversary is not allowed to make lots of guesses.
- No account of collisions or nonces.
Bundle-valued Random Variables

- Assumptions on random variables
  - \((\mathcal{Y}, A, \mathbb{P})\) is only constrained by independence
  - \(B \colon \mathcal{Y} \to \mathcal{X}\) is measurable

  - For
    - Message delays and behavior of the penetrators
    - Encapsulates choice of nonces, interlocutors

- \((\mathcal{Y}, A, \mathbb{P})\) probability space
  - \(\mathcal{Y}\) is set of bundles, \(A\) is a complete separable
    - Metric space
    - Finite labeled graph

- Bundle is "countably specified" object
  - View bundle as first class object

Current work (J. Cutherman, F. J. Thayer, L. Zuck)
Protocol Correctness: Possible

- Given bounds on size and some stochastic aspects, what can be said about a specific bundle?
  - \( B(m) \) is correct within tolerance.
  - True in asymptotic sense.
  - \( B(m) \) is correct almost surely tractable probability.
  - Requires bundles of infinite size.
  - \( B(m) \) is correct almost surely.

Meanings

Protocol Correctness: Possible
Behavior of Adversary

- Adversary sees into future.
- $Z$ is set of forged messages.
- $R, W$ stochastically independent.

$Z \xleftarrow{} R \times W$

Diagram

- Other random input $R \xleftarrow{} Z$
- Regular messages, $W \xleftarrow{} Z$
- Depends on
- Some random variable $Z \xleftarrow{} Z$
Pure Authentication Protocol

- Number of adversary messages
  △ Number of regular nonces
  □ Number of regular messages
- Depends on size of bundle. Maximum of
  cure.
- Carter-Wegman hashing is unconditionally safe.

mates for likelihood of protocol failure.

Using Carter-Wegman hashes, obtain upper esti-

Adversary wins if makes one correct guess.

adversary.

are hash guesses of adversary.

Protocol of theoretical interest (Bellare-Rogaway)

+$
An Authentication Protocol

Hash functions $f$ is chosen independently of $p$.
Nonces are independently uniformly chosen.

$$f[q_N \oplus V]$$

$$f[q_N \oplus q_N \oplus p \oplus B]$$

$B \leftarrow q_N$
Conclusion

- Requires quality random number generation.

- Limits key longevity.

- Protocol tolerance: Probability of failure under conditions of bundle size.

- Contrast to cryptographic aspects.

- Protocol correctness close to "social aspects" of

- Suggests protocol quality standard.