An Interface Specification Language for Automatically Analyzing Cryptographic Protocols

Internet Society Symposium on Network and Distributed System Security

February 10-11, 1997
San Diego Princess Resort, San Diego, CA

Dr. Stephen H. Brackin
Arca Systems, Inc.
303 E. Yates St., Ithaca, NY 14850
(607) 277-8211 or (607) 277-2739
brackin@arca.com

Supported by Arca Systems, Inc., by ESC/AXS through PRISM, and by Rome Laboratory

P R I S M
Custom Command and Control

Overview of Talk

- **Problem and basic solution idea**
  - **Cryptographic protocols and network model**
  - **Protocol failure, TMN example**
  - **Belief logics**
- **Examples of what the Automatic Authentication Protocol Analyzer (AAPA) can do**
  - **Interface Specification Language (ISL) TMN specification**
  - **Terminal output, failed-goals files, other files**
  - **High points of (unnamed) commercial application**
- **Plans for the future**
  - **Thorough, but still fast and automatic, analyses**
  - **Good-guy deductions vrs. Bad-guy searches**
- **Lessons for protocol users and designers**

Cryptographic Protocols and Network Model

- **Goal: Secure communication over insecure networks**
  - **Networks, principals, and messages**
  - **No other communications**
  - **Worst case: enemy can read, or be true source, of everything**
  - **Confidentiality and authentication**
- **Tools for achieving goal**
  - **Shared or confirmable secrets**
  - **Symmetric- and public-key encryption**
  - **Effectively 1-to-1 hash functions**
  - **Timestamps, nonces, signatures, etc.**
- **Protocols**
  - **Distributed algorithms carried out by stages**
  - **Abort if something not as expected**

Protocol Failure

- **Example: TMN (Tatebayeshi-Matsuzaki-Newman) key-distribution protocol**
  - **1. A -> S: A, S, B, {SkA}Rsa(PkS)**
  - **2. S -> B: S, B, A**
  - **3. B -> S: B, S, A, {SkB}Rsa(PkS)**
  - **4. S -> A: S, A, B, {SkB}Xor(SkA)**
- **ISL notation, but more-or-less standard**
- **Published (CRYPTO '89), recommended by experts**
- **It's wrong, and has lots of company**

Belief Logics

- **Formalize authentication reasoning that assumes**
  - **Good encryption and hash functions**
  - **Correct distributions of secrets**
- **Sample deduction**
  - **If P believes only P and Q know K, and P receives an M that K decrypts to something meaningful, then P believes Q sent M -- though not necessarily recently or to P**
- **AAPA's BGNY logic**
  - **Derived from Gong-Needham-Yahalom (GNY) logic**
  - **Sending, receiving, belief, freshness, conveyence, shared secrets, possession, recognizability, trustworthiness, not-from-here checks, message extensions, feasibility constraints**
  - **Many extensions and corrections to GNY (e.g., stages, key-exchange functions, hash codes as keys)**

Sample AAPA Analysis: ISL (1)

/* Tatebayashi, Matsuzaki, Newman (TMN) Protocol */

DEFINITIONS:

PRINCIPALS: A,B,S;

PRIVATE KEYS: ^PkS;

PUBLIC KEYS: PkS;

SYMMETRIC KEYS: SkA,SkB;

ENCRYPT FUNCTIONS: Xor,Rsa;

Xor WITH ANYKEY HASINVERSE
Xor WITH ANYKEY;

Rsa WITH PkS HASINVERSE Rsa WITH ^PkS;

Sample AAPA Analysis: ISL (2)

INITIALCONDITIONS:

A Received Xor,Rsa,A,B,S,PkS,SkA;

A Believes (Fresh SkA; PublicKey S Rsa PkS;
SharedSecret A S SkA;
Trustworthy B; Trustworthy S);

B Received Xor,Rsa,A,B,S,PkS,SkB;

B Believes (Fresh SkB; PublicKey S Rsa PkS;
SharedSecret A B SkB;
Trustworthy A; Trustworthy S);

S Received Xor,Rsa,^PkS;

S Believes (PrivateKey S Rsa ^PkS; Trustworthy A;
Trustworthy B);

P R I S M
Custom Command and Control

Sample AAPA Analysis: ISL (3)

PROTOCOL:

1. A->S : A,S,B,{SkA}Rsa(PkS);

2. S ->B : S,B,A;

3. B ->S : B,S,A,{SkB}Rsa(PkS)||(SharedSecret A B SkB);

4. S ->A : S,A,B,{SkB}Xor(SkA)||(SharedSecret A B SkB);

GOALS:

1. S Possesses SkA;

3. S Possesses SkB;
S Believes SharedSecret A B SkB;

4. A Possesses SkB;
A Believes SharedSecret A B SkB;

Sample AAPA Analysis: Terminal Output

Creating theory tmn
Beginning tmn proofs
Initializing globals

Proving default goals, stage 1
Retrying failed default goals, stage 1
Proving user goals, stage 1
Proving default goals, stage 2
Proving user goals, stage 2
Proving default goals, stage 3
Retrying failed default goals, stage 3
Proving user goals, stage 3
User-goal failure, stage: 3!
Goal statement: S Believes (SharedSecret A B SkB);

Sample AAPA Analysis: .fail file (1)

/* ######### Failed default goal from stage 3: ######### */

S Believes

(B Conveyed {SkB}Rsa(PkS)||(SharedSecret A B SkB));

/* ============ Unproved subgoals: ============ */

S Believes (S Recognizes SkB);

S Believes (SharedSecret S B SkB);

S Believes

(Fresh {SkB}Rsa(PkS)||(SharedSecret A B SkB));

/* ============= Proved subgoals: ============= */

S Received {SkB}Rsa(PkS)||(SharedSecret A B SkB);

S Possesses Rsa,UNPkS;

S Believes (PrivateKey S Rsa UNPkS);

Sample AAPA Analysis: .fail file (2)

```
/* ######## Failed default goal from stage 1: ######## */
S Believes (A Conveyed {SkA}Rsa(PkS));
/* ============ Unproved subgoals: ============ */
S Believes (S Recognizes SkA);
S Believes (SharedSecret S A SkA);
S Believes
   (Fresh {SkA}Rsa(PkS)||(SharedSecret A S SkA));
/* ============= Proved subgoals: ============= */
S Received {SkA}Rsa(PkS);
S Possesses Rsa,UNPkS;
S Believes (PrivateKey S Rsa UNPkS);
```


P R I S M
Custom Command and Control

Sample AAPA Analysis: Other Files

- **Have a .thms file giving ISL versions of all theorems**
  - **In TMN case, all interesting theorems are proved subgoals**
  - **In other cases, useful for figuring out what happened**
- **Have option of producing**
  - **Higher Order Logic (HOL) theory of protocol**
  - **HOL translation of ISL input**
- **Optional outputs mainly used for debugging AAPA**

AAPA Analysis of Commercial Protocols

- **Customer requested confidentiality**
- **Protocols moderately complicated, and huge**
  - **Roughly 100 items or subitems in some messages**
  - **Most of detail irrelevant to AAPA analysis -- but which?**
  - **Biggest formally analyzed examples known to author**
- **Results of analyses of two protocols**
  - **Did not find failures in protocols**
  - **Found omissions, errors and inconsistencies in documentation**
  - **Produced basis for much better documentation**
- **AAPA additions necessary for effort**
  - **ISL abbreviation capacity (X = Y as in C)**
  - **New diagnostics for feasibility failures**

Plans for the Future

- **Belief logics vs. attack construction**
  - **Simplicity and speed vs. thoroughness and rigor**
  - **How to gain one without losing the other: Replace searches with using theorems about searches**
- **Research program**
  - **Find failed protocols in literature**
  - **Analyze them with AAPA**
  - **For failures AAPA misses, ask where belief logic allowed false beliefs during attacks**
  - **Adjust logic and repeat process; time always polynomial**
- **User interfaces**
  - **Use CAPSL (Common Authentication Protocol Specification Language) by Millen and protocol-analysis community**
  - **Make sure ISL virtues survive in CAPSL**

Lessons for Protocol Users and Designers

- **Protocol failure is a little-known, but very real problem**
  - **“Easy” design problem is actually a weak link**
  - **About half of published protocols fail -- estimate based on Lowe’s and my own experiences**
- **An AAPA analysis is worth performing**
  - **Finds common failures**
  - **Gives overview, corrects documentation, identifies information flows, and identifies trust assumptions**
  - **It’s fast and cheap**
- **A near-future AAPA could make protocol failure, for practical purposes, into a solved problem**