

Verification of Cryptographic Protocols

Part 1: The Dolev-Yao Model and “Ping-Pong” Protocols

Heiko Stamer

University of Kassel
Department of Mathematics/Computer Science
Heinrich-Plett-Straße 40, D-34132 Kassel
stamer@theory.informatik.uni-kassel.de
76F7 3011 329D 27DB 8D7C 3F97 4F58 4EB8 FB2B E14F

Seminar: Theoretische Informatik, April 2005

1 Introduction

2 Dolev-Yao Model

3 “Ping-Pong” Protocols

Cryptographic protocols are sequential exchanges of messages between participating entities (principals, parties) whose purpose is to establish *protection or security goals*.

Protection goals are required properties of the application/system:

- Authentication of principals
- Confidentiality/Secrecy and integrity of messages or distributed keys
- Robustness, non-repudiation, anonymity, ...

Notation for protocol steps: $A \rightarrow B : m$

Regular (honest) principals and names: A, B, C, \dots

Impersonator or Adversary: I, Z

Nonces: n_A, n_B, \dots (fresh/random numbers)

Cryptographic keys: $K_{AB}, PK_A, SK_A, \dots$

Primitives: $\{\cdot\}_K$ (encryption), $[\cdot, \cdot]$ (pairing), ...

Flawed protocol: Needham-Schroeder Public-Key Protocol, 1978

Protection goal: Authentication of the two principals A and B

Short version without
authentication server:

1. $A \rightarrow B : \{n_A, A\}_{PK_B}$
2. $B \rightarrow A : \{n_A, n_B\}_{PK_A}$
3. $A \rightarrow B : \{n_B\}_{PK_B}$

Impersonator executes two
concurrent sessions 1 and 2:

- 1.(1) $A \rightarrow I : \{n_A, A\}_{PK_I}$
- 1.(2) $I(A) \rightarrow B : \{n_A, A\}_{PK_B}$
- 2.(2) $B \rightarrow I(A) : \{n_A, n_B\}_{PK_A}$
- 2.(1) $I \rightarrow A : \{n_A, n_B\}_{PK_A}$
- 3.(1) $A \rightarrow I : \{n_B\}_{PK_I}$
- 3.(2) $I(A) \rightarrow B : \{n_B\}_{PK_B}$

The attack was found **only recently in 1995** by Gavin Lowe.

Fixed protocol: Needham-Schroeder-Lowe, 1995

Protection goal: Authentication of the two principals A and B

1. $A \rightarrow B$: $\{n_A, A\}_{PK_B}$
2. $B \rightarrow A$: $\{n_A, n_B, B\}_{PK_A}$
3. $A \rightarrow B$: $\{n_B\}_{PK_B}$

Proof of correctness (Lowe, 1995): semi-automatic technique

- 1 For a “small system” it was proven with the help of a model checker (Failures Divergence Refinement Checker for CSP).
- 2 The full proof (arbitrarily-sized system) was done by hand.

Flawed protocol: Modified Needham-Schroeder-Lowe

Protection goal: Authentication of the two principals A and B

Impersonator executes two concurrent sessions 1 and 2:

1. $A \rightarrow B$: $\{A, n_A\}_{PK_B}$
2. $B \rightarrow A$: $\{[n_A, n_B], B\}_{PK_A}$
3. $A \rightarrow B$: $\{n_B\}_{PK_B}$

- 1.(1) $I(A) \rightarrow B$: $\{A, I\}_{PK_B}$
- 2.(1) $B \rightarrow I(A)$: $\{[I, n_B], B\}_{PK_A}$
- 1.(2) $I \rightarrow A$: $\{I, [n_B, B]\}_{PK_A}$
- 2.(2) $A \rightarrow I$: $\{[[n_B, B], n_A], A\}_{PK_I}$
- 3.(1) $I(A) \rightarrow B$: $\{n_B\}_{PK_B}$

Attack assumptions:

- 1 Associativity of the pairing function $[\cdot, \cdot]$
(This algebraic property often holds!)
- 2 Type confusions: Names/pairs typed as nonces

This attack was first mentioned in 2001 by Millen and Shmatikov.

Flawed protocol: Wired Equivalent Privacy Protocol (WEP)

Protection goals: Confidentiality and integrity of the transferred messages between the principals A and B

$$1. \quad A \rightarrow B \quad : \quad v, ([m, \text{CRC}(m)] \oplus \text{RC4}(v, K_{AB}))$$

Properties of \oplus (XOR):

$$x \oplus (y \oplus z) = (x \oplus y) \oplus z \quad (1)$$

$$x \oplus y = y \oplus x \quad (2)$$

Attack assumptions:

$$\text{CRC}(x \oplus y) = \text{CRC}(x) \oplus \text{CRC}(y) \quad (3)$$

$$[x_1, y_1] \oplus [x_2, y_2] = [x_1 \oplus x_2, y_1 \oplus y_2] \quad (4)$$

Attack: A malicious adversary can apply controlled modifications d to a protected message m .

$$\begin{aligned}
 ([m, \text{CRC}(m)] \oplus \text{RC4}(v, K_{AB})) \oplus [d, \text{CRC}(d)] & \stackrel{(1),(2)}{=} \text{RC4}(v, K_{AB}) \oplus ([m, \text{CRC}(m)] \oplus [d, \text{CRC}(d)]) \\
 & \stackrel{(4)}{=} \text{RC4}(v, K_{AB}) \oplus ([m \oplus d, \text{CRC}(m) \oplus \text{CRC}(d)]) \\
 & \stackrel{(3)}{=} \text{RC4}(v, K_{AB}) \oplus ([m \oplus d, \text{CRC}(m \oplus d)])
 \end{aligned}$$

This attack was discovered 2001 by Borisov, Goldberg, and Wagner.

Other flawed protocols: Otway-Rees, Kerberos, Wide-mouthed Frog, Yahalom, CCITT X.509, SSL 2.0/3.0, ...

Conclusion and motivation for a formal verification of protocols:

- Cryptographic protocols are subject to very subtle flaws
- Behavior of the adversary is hard to predict
- Algebraic properties of primitives have a strong influence

Well-known techniques for protocol validation/verification:

- 1** Reachability analysis (e.g. Interrogator, NRL), model checking (e.g. FDR, Mur φ), constraint solving, specification languages (e.g. CAPSL, HLPSL)
- 2** Approaches based on modal logic and deduction
 - BAN (Burrows, Abadi, Needham), GNY (Gong, Needham, Yahalom), . . .
 - Drawbacks: originally only usable for authentication, needs some kind of “unnatural” idealization, no real “security proof”
- 3** Methods based on state machines (protocol traces)
- 4** Using algebras to reason about knowledge of participants

D. Dolev, A.C. Yao: On the Security of Public Key Protocols
IEEE Transactions on Information Theory, 29(2), pp. 198–208, 1983

- First paper that provides a formal model:
 - Precise language for the description of protocols
 - Execution model, e.g. behavior and capabilities of the adversary
 - Formal specification of the desired protection goals
- Protocol restrictions are mostly on honest participants and the protection goal, i.e. the adversarial model is still quite general
- Interesting theoretical results:
 - Simple security characterization for a subclass of protocols (Cascade Protocols)
 - Security problem for the more general class of Name-Stamp Protocols is decidable in polynomial time

Example (Secrecy violated)

- | | | | | |
|----|-------------------|---------------------|----------------|----------------|
| | 1.(1) | $A \rightarrow Z/B$ | : | $\{m\}_{PK_B}$ |
| 1. | $A \rightarrow B$ | : | $\{m\}_{PK_B}$ | |
| 2. | $B \rightarrow A$ | : | $\{m\}_{PK_A}$ | |
| | 1.(2) | $Z \rightarrow B$ | : | $\{m\}_{PK_B}$ |
| | 2.(2) | $B \rightarrow Z$ | : | $\{m\}_{PK_Z}$ |
| | 2.(1) | $Z \rightarrow A$ | : | $\{m\}_{PK_A}$ |

Example (Secrecy kept)

- | | | | |
|----|-------------------|---|-------------------|
| 1. | $A \rightarrow B$ | : | $\{m, A\}_{PK_B}$ |
| 2. | $B \rightarrow A$ | : | $\{m, B\}_{PK_A}$ |

Example (Secrecy violated)

- | | | | | |
|----|-------------------|---------------------|------------------------------|--|
| | 1.(1) | $A \rightarrow B$ | : | $\{\{m\}_{PK_B}, A\}_{PK_B}$ |
| | 2.(1) | $B \rightarrow Z/A$ | : | $\{\{m\}_{PK_A}, B\}_{PK_A}$ |
| 1. | $A \rightarrow B$ | : | $\{\{m\}_{PK_B}, A\}_{PK_B}$ | |
| 2. | $B \rightarrow A$ | : | $\{\{m\}_{PK_A}, B\}_{PK_A}$ | |
| | 1.(2) | $Z \rightarrow A$ | : | $\{\{\{m\}_{PK_A}, B\}_{PK_A}, Z\}_{PK_A}$ |
| | 2.(2) | $A \rightarrow Z$ | : | $\{\{\{m\}_{PK_A}, B\}_{PK_Z}, A\}_{PK_Z}$ |
| | 1.(3) | $Z \rightarrow A$ | : | $\{\{m\}_{PK_A}, Z\}_{PK_A}$ |
| | 2.(3) | $A \rightarrow Z$ | : | $\{\{m\}_{PK_Z}, A\}_{PK_Z}$ |

- 1 Perfect public key cryptosystem (E_X, D_X) and infrastructure:
 - One-way functions E_X are unbreakable; only X knows D_X
 - Symmetry of the functions, i.e. $\forall X : E_X D_X = D_X E_X = 1$
 - Secure public directory contains (X, E_X) for all X
- 2 Two-party protocols (no trusted third party necessary)
 - Honest parties are stateless (“Ping-Pong” Protocols)
- 3 Uniform format of the protocol messages
- 4 Behavior of the adversary (active saboteur):
 - He can obtain and intercept any message passing through the network, i.e. he acts as the network.
 - He is a legitimate user of the network (or has corrupted some participants), and thus he can initiate concurrent protocol instances with any other user.
 - He can maintain state information, i.e. record all transmitted messages or protocol sessions.
- 5 Protection goal: Secrecy of the input for honest principals

- Very simple protocols (less practical relevance?)
- Dolev and Yao distinguish between two classes of protocols:

Cascade Protocols are protocols in which the users can apply only the public key **encryption-decryption operations** (E_X, D_X) to form messages; several layers of these operators may be applied.

Name-Stamp Protocols are protocols in which the users are additionally allowed to **append** (i_X), **delete** (d), and **check** (d_X) **names** in messages. Thus Name-Stamp Protocols are a generalization of the above Cascade Protocols.

Let Σ be finite alphabet of operator symbols and Σ^* be the set of all strings over Σ including the empty word λ .

Operators for Cascade Protocols: $\Sigma = \{E_X, D_X \mid X \text{ is a user name}\}$

Operators for Name-Stamp Protocols:

$$\Sigma = \{d\} \cup \{E_X, D_X, i_X, d_X \mid X \text{ is a user name}\}$$

Based on the properties of the public key cryptosystem and the name-stamps we have the following reduction/rewriting system:

$$E_X D_X \rightarrow \lambda$$

$$D_X E_X \rightarrow \lambda$$

$$d_X i_X \rightarrow \lambda$$

$$d i_X \rightarrow \lambda$$

Observation: If $a, b, c \in \Sigma$, $ab \rightarrow \lambda$ and $bc \rightarrow \lambda$ then $a = c$, i.e. the reduction process has the Church-Rosser property.

Reduced form (normal form) of $\alpha \in \Sigma^*$ is denoted by $\bar{\alpha}$.

Restricted operators: $\Sigma_X = \{d, D_X\} \cup \{E_Y, i_Y, d_Y \mid Y \text{ is a user name}\}$

Definition (Dolev, Even, and Karp)

A "Ping-Pong" protocol $P(S, R)$ between the users S and R is a sequence of strings $\alpha_1, \alpha_2, \dots, \alpha_\ell$, such that $\alpha_i \in \Sigma_S^*$ if i is odd and $\alpha_i \in \Sigma_R^*$ otherwise.

Operator-words of the adversary: $\Delta = (\Sigma_Z \cup \{\alpha_i(X, Y) \mid 1 \leq i \leq \ell, X \text{ and } Y \text{ are different users}\})^*$

Definition

Let $\alpha_1(S, R)$ be the first operator-word of $P(S, R)$ and Z an adversary. P is **insecure**, if there exists a string $\gamma \in \Delta$ such that $\overline{\gamma\alpha_1} = \lambda$.

It is sufficient to consider $\overline{\gamma\alpha_1} = \lambda$ instead of $\overline{\gamma\alpha_i\alpha_{i-1}\cdots\alpha_1} = \lambda$.
Further it is sufficient to consider only one possible adversary Z .

Example (Secrecy violated)

1. $A \rightarrow B$: $\{\{m\}_{PK_B}, A\}_{PK_B}$ $\alpha_1(A, B) = E_{B i_A} E_B$
2. $B \rightarrow A$: $\{\{m\}_{PK_A}, B\}_{PK_A}$ $\alpha_2(A, B) = E_{A i_B} E_A D_B d_A D_B$

$$\gamma' = \underbrace{D_Z d_A D_Z}_Z \underbrace{E_Z i_A E_Z D_A d_Z D_A}_{A(3)} \underbrace{E_{A i_Z} d D_Z d_A D_Z}_{Z(3)} \underbrace{E_Z i_A E_Z D_A d_Z D_A}_{A(2)} \underbrace{E_{A i_Z}}_{Z(2)}$$

$$\overline{\gamma' \alpha_2 \alpha_1} = \lambda$$

$$\gamma = \gamma' \alpha_2 \quad \overline{\gamma \alpha_1} = \lambda$$

Theorem (Dolev, Even, and Karp)

For "Ping-Pong" protocols there exists a security checking algorithm whose input are the generic cancellation rules and the protocol. The time-complexity is $O(n^3)$, where n is the length of the input.

Generic construction: $O(n^3)$ -time and $O(n^3)$ -space

Let L be the context-free language of all possible protocol traces of $P(S, R)$ that can be reduced by the cancellation rules to the empty word λ :

$$L = \{\xi \in \text{TRACES}_{P(S,R)} \mid \bar{\xi} = \lambda\}$$

Further let L_Z be the regular language of all possible attacks:

$$L_Z = \{\psi\alpha_1(S, R) \mid \psi \in \Delta\}$$

Essentially, the security problem of $P(S, R)$ is the question whether the intersection of a regular language and a context-free language is non-empty, i.e. $L \cap L_Z \stackrel{?}{=} \emptyset$.

Remark

The algorithm of Dolev, Even, and Karp has time-complexity $O(n^3 m)$ and space-complexity $O(n^2 m)$, where m is the length of the context-free grammar G such that $L = L(G)$ and n is the number of states of the non-deterministic finite automaton A such that $L_Z = L(A)$.

[Low96] Gavin Lowe.

Breaking and fixing the Needham-Schroeder public-key protocol using FDR.
TACAs '96: Proceedings of the Second International Workshop on Tools and Algorithms
for Construction and Analysis of Systems, pp. 147–166, 1996.

[DY83] Danny Dolev and Andrew C. Yao.

On the security of public key protocols.
IEEE Transactions on Information Theory, 29(2):198–208, 1983.

[DEK82] Danny Dolev, Shimon Even, and Richard M. Karp.

On the security of ping-pong protocols.
Information and Control, 55(1–3):57–68, 1982.

[CDL05] Véronique Cortier, Stéphanie Delaune, and Pascal Lafourcade.

A survey of algebraic properties used in cryptographic protocols.
Journal of Computer Security, 2005. To appear.

[Mic05] Daniele Micciancio.

Advanced cryptography: Symbolic methods for security analysis.
<http://www.cse.ucsd.edu/classes/sp05/cse208/index.html>