

可证明安全 - 1. 前言

钱宸

网络空间安全学院
山东大学

2025/09/29

Contents

1. 背景

2. 密码学的形式化标记

3. Needham Schnroeder 密钥交换

背景

密码学的形式化标记

形式化标记

- A, B: 用户 A, B

形式化标记

- A, B : 用户 A, B
- $A \rightarrow B$: 用户 A 向用户 B 发送消息

形式化标记

- A, B : 用户 A, B
- $A \rightarrow B$: 用户 A 向用户 B 发送消息
- $pk(A), sk(A)$: 用户 A 的公钥和私钥

形式化标记

- A, B : 用户 A, B
- $A \rightarrow B$: 用户 A 向用户 B 发送消息
- $\text{pk}(A), \text{sk}(A)$: 用户 A 的公钥和私钥
- $\llbracket \text{pk}(A) \rrbracket_x^a$: 对消息 x 使用用户 A 的公钥进行加密

形式化标记

- A, B : 用户 A, B
- $A \rightarrow B$: 用户 A 向用户 B 发送消息
- $\text{pk}(A), \text{sk}(A)$: 用户 A 的公钥和私钥
- $\llbracket \text{pk}(A) \rrbracket_x^a$: 对消息 x 使用用户 A 的公钥进行加密
- $\langle x, y \rangle$: 有序对 (x, y) , 表示元素之间的简单连接

Needham Schnroeder 密钥交换

密钥交换 (简略) [Needham and Schroeder, 1978]

1. $A \rightarrow B : \llbracket \langle A, N_A \rangle \rrbracket_{pk(B)}^a$
2. $B \rightarrow A : \llbracket \langle B, N_B \rangle \rrbracket_{pk(A)}^a$
3. $A \rightarrow B : \llbracket N_b \rrbracket_{pk(B)}^a$

协议执行

1. 用户 A 向用户 B 发送加密消息 $\llbracket \langle A, N_A \rangle \rrbracket_{pk(B)}^a$, 其中 N_A 是用户 A 生成的随机数.

密钥交换 (简略) [Needham and Schroeder, 1978]

1. $A \rightarrow B: \llbracket \langle A, N_A \rangle \rrbracket_{pk(B)}^a$
2. $B \rightarrow A: \llbracket \langle B, N_B \rangle \rrbracket_{pk(A)}^a$
3. $A \rightarrow B: \llbracket N_b \rrbracket_{pk(B)}^a$

协议执行

1. 用户 A 向用户 B 发送加密消息 $\llbracket \langle A, N_A \rangle \rrbracket_{pk(B)}^a$, 其中 N_A 是用户 A 生成的随机数.
2. 用户 B 向用户 A 发送加密消息 $\llbracket \langle N_A, N_B \rangle \rrbracket_{pk(A)}^a$, 其中 N_B 是用户 B 生成的随机数.

密钥交换 (简略) [Needham and Schroeder, 1978]

1. $A \rightarrow B: \llbracket \langle A, N_A \rangle \rrbracket_{pk(B)}^a$
2. $B \rightarrow A: \llbracket \langle B, N_B \rangle \rrbracket_{pk(A)}^a$
3. $A \rightarrow B: \llbracket N_B \rrbracket_{pk(B)}^a$

协议执行

1. 用户 A 向用户 B 发送加密消息 $\llbracket \langle A, N_A \rangle \rrbracket_{pk(B)}^a$, 其中 N_A 是用户 A 生成的随机数.
2. 用户 B 向用户 A 发送加密消息 $\llbracket \langle N_A, N_B \rangle \rrbracket_{pk(A)}^a$, 其中 N_B 是用户 B 生成的随机数.
3. 用户 A 向用户 B 发送加密的消息 $\llbracket N_B \rrbracket_{pk(B)}^a$, 以证明自己是通信的发起者.

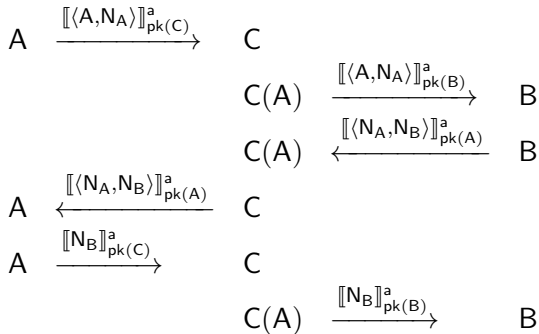
更严格的交互说明

$$\begin{array}{ccc} A & \xrightarrow{\llbracket \langle A, N_A \rangle \rrbracket_{pk(B)}^a} & \xrightarrow{\llbracket \langle x, y \rangle \rrbracket_{pk(B)}^a} B \\ A & \xleftarrow{\llbracket \langle N_A, z \rangle \rrbracket_{pk(A)}^a} & \xleftarrow{\llbracket \langle y, N_B \rangle \rrbracket_{pk(x)}^a} B \\ A & \xrightarrow{\llbracket z \rrbracket_{pk(B)}^a} & \xrightarrow{\llbracket N_B \rrbracket_{pk(B)}^a} B \end{array}$$

并行运行与攻击

- 协议实际运行在并行环境中, 可能会有多个实例同时进行.
- 用变量 x, y, z 来指代接收到, 但是无法验证的消息.
- 在并行环境中, 是否存在攻击?

Lowe's 中间人攻击 [Lowe, 1998]



Lowe's 中间人攻击

- C 对 A 扮演 C 自己, 而对 B 扮演 A.
- B 以为自己在与 A 通信, 实际上却是在与 C 通信.

[Needham and Schroeder, 1978]

该论文实际提出的时候要求: 任意参与者都诚实执行协议. 安全性保障针对诚实但好奇的敌手.

- 因此 [Lowe, 1998] 的攻击严格意义上并不是一个针对 NS 协议的攻击.
- 然而, NS 协议中的假设在如今的环境中被普遍认为是不成立的.

Lowe 的修复



- Lowe 通过在第二个消息中加入 B 的身份消息, 修复了 NS 协议.
- 第二轮消息从 $[[\langle N_A, N_B \rangle]]_{pk(A)}^a$ 变更为 $[[\langle N_A, \langle N_B, B \rangle \rangle]]_{pk(A)}^a$.
- 往后我们称修复后的协议为 NSL 协议.

Thank you!

`chen.qian@sdu.edu.cn`

`https://qianchen92.github.io/`

References

-  Lowe, G. (1998).
Casper: A compiler for the analysis of security protocols.
J. Comput. Secur., 6(1-2):53–84.
-  Needham, R. M. and Schroeder, M. D. (1978).
Using encryption for authentication in large networks of computers.
Commun. ACM, 21(12):993–999.