

文章编号:1001-9081(2005)10-2272-04

一种时间相关的分析安全协议的扩展逻辑

赵华伟¹, 李大兴¹, 秦 静²

(1. 山东大学 网络信息安全研究所, 山东 济南 250100;

2. 山东大学 数学与系统科学学院, 山东 济南 250100)

(zhuav@163.com)

摘 要:在对 CS 逻辑进行研究的基础上,提出了 CS 逻辑的扩展逻辑。该扩展逻辑对 CS 逻辑中存在的一些缺陷进行了修改和扩展,使其不仅可以分析公钥协议,还可以分析对称密钥协议。最后对一个协议实例进行了有效的形式化分析。

关键词:CS 逻辑;认证协议;形式化分析

中图分类号:TP309 **文献标识码:**A

Time-dependent extension logic of secure protocols

ZHAO Hua-wei¹, LI Da-xing¹, QIN Jing²

(1. Institute of Network Security, Shandong University, Jinan Shandong 250100, China;

2. School of Mathematics and System Science, Shandong University, Jinan Shandong 250100, China)

Abstract: CS logic added the time into logic structure, so it could be used to analyze the security of time-dependent secrets of timed-release public key cryptographic protocols. An extension of CS logic was proposed in this paper, which corrected some defects of CS logic, and extended the logic to analyze the symmetrical key cryptographic protocols. Finally a good formal analysis of a concrete protocol was given.

Key words: CS logic; authentication protocol; formal analysis

0 引言

安全协议的执行是由消息在主体之间传播实现的。由于消息传播有先后顺序,因此主体依据消息所产生的知识和信仰有时间上的次序。根据这个时序,文献[1]中提出了一种将时间与逻辑结构相结合的逻辑——CS 逻辑,用来分析公钥协议的安全性。与著名的安全协议形式化分析工具类 BAN 逻辑^[2]相比,CS 逻辑将时间因素加入到逻辑分析中,因此在分析与时间相关的安全性问题时,具有明显优势;此外 CS 逻辑对信仰和知识都能进行推理,不仅可以分析协议的认证性,而且可以分析协议的保密性。

然而 CS 逻辑中的若干推理公理存在着与现实不符,或表述不清的缺陷,如:加密消息获取公理违背了公钥是公开的性质;解密公理对于签名和解密过程没有进行区分;还有的公理不能简明清晰地表达出消息中蕴含的逻辑关系,等等。我们在对 CS 逻辑进行细心研究后,对其公理进行了适当的改进和摒弃,在此基础上,又增加了若干公理,使扩展后的 CS 逻辑不仅可以分析公钥协议,还可以分析对称密钥协议。

1 CS 逻辑简介

1.1 CS 逻辑的符号含义

惯例上常用 Σ 和 Ψ 来表示任意主体, ENT 来表示所有主体; k_Σ 和 k_Σ^{-1} 来分别主体表示 Σ 的公钥和私钥; $e(x, k_\Sigma)$ 来表示加密消息; $d(x, k_\Sigma^{-1})$ 来表示用私钥进行的运算; $K_{\Sigma, t}\phi$ 表示主体 Σ 在 t 时知道表达式 ϕ ; $L_{\Sigma, t}x$ 表示主体 Σ 在 t 时知道并且可以有意识地重新产生消息 x ; $B_{\Sigma, t}\phi$ 表示主体 Σ 在 t 时相信表

达式 ϕ ; $S(\Sigma, t, x)$ 意味着在 t 时主体 Σ 发送了消息 x ; $R(\Sigma, t, x)$ 意味着在 t 时主体 Σ 接收了消息 x 。

1.2 CS 逻辑的推理规则

整个 CS 逻辑只有两种基本推理规则:1) $R1$ 为 MP 规则(modus ponens):由 p 和 $p \rightarrow q$ 可以推导出 q ;2) $R2$ 为 Nec 规则(necessitation):由 $\vdash p$ 可以推导出 $B_{\Sigma, t}p$ 和 $K_{\Sigma, t}p$ 来。

从这两个基本推理规则又得出五个推理规则:

$R3$: from $(p \wedge q)$ infer p

$R4$: from p and q infer $(p \wedge q)$

$R5$: from p infer $(p \vee q)$

$R6$: from $\neg(\neg p)$ infer p

$R7$: from (from p infer q) infer $(p \rightarrow q)$

下面给出几个重要的公理:

$A1$ 知识与信仰公理

1) $K(B)_{\Sigma, t}p \wedge K(B)_{\Sigma, t}(p \rightarrow q) \rightarrow K(B)_{\Sigma, t}q$

2) $\exists t \exists p(K_{\Sigma, t}p \rightarrow p)$

3) $K(L)_{i, t}x \rightarrow \forall t' \geq t K(L)_{i, t'}x$

4) $B_{i, t}x \rightarrow \forall t' \geq t B_{i, t'}x$

其中 2) 描述了知识与信仰的区别,即知识表示了知道的东西就是正确的,而信仰不涉及正确与否。3) 和 4) 表示了主体的知识与信仰,一旦获得,便不会丢失。

$A2$ 发送公理

$\exists t, \exists x(S(\Sigma, t, x) \rightarrow L_{\Sigma, t}x \wedge \exists i, i \in ENT \setminus \{\Sigma\}, \exists t' > t, R(i, t', x))$

即如果有主体 Σ 在 t 时刻发送了一条消息,那么在此之后一定有某个主体 i 接收了这条消息。

收稿日期:2005-04-21;修订日期:2005-06-28 基金项目:国家 973 规划项目(G1999035802)

作者简介:赵华伟(1977-),男,山东聊城人,博士研究生,主要研究方向:信息安全; 李大兴(1963-),男,教授,博士生导师,主要研究方向:信息安全; 秦静(1960-),女,教授,主要研究方向:密码学、安全协议。

A3 接收公理

$$\exists t, \exists x(R(\Sigma, t, x) \rightarrow L_{\Sigma, t}x \wedge \exists i, i \in ENT \setminus \{\Sigma\}, \\ \exists t' < t, S(i, t', x))$$

即如果主体 Σ 在 t 时刻收到一个消息,那么在此之前,一定另有某个主体 i 发送了这消息。

A4 密钥公理

$$1) \exists t, \exists x, \exists i, i \in \{ENT\},$$

$$(L_{i, t}x \wedge L_{i, t}k_{\Sigma} \rightarrow L_{i, t}(e(x, k_{\Sigma})))$$

$$2) \exists t, \exists x, \exists i, i \in \{ENT\},$$

$$(L_{i, t}x \wedge L_{i, t}k_{\Sigma}^{-1} \rightarrow L_{i, t}(d(x, k_{\Sigma}^{-1})))$$

$$3) \forall t(\forall i, i \in \{ENT\}, L_{i, t}k_i^{-1} \wedge \forall j,$$

$$j \in ENT \setminus \{i\} \neg L_{j, t}k_i^{-1})$$

$$4) \exists t, \exists x(\exists i, i \in \{ENT\}, L_{i, t}d(x, k_{\Sigma}^{-1}) \rightarrow L_{\Sigma, t}x)$$

即如果主体 i 在 t 时知道消息 x 并且知道有关密钥,则在 t 时, i 知道对 x 进行密钥运算的结果;主体的私钥只有他自己知道。

A5 密文公理

$$1) \exists t, \exists x, \exists i, i \in \{ENT\}(\neg L_{i, t}k_{\Sigma} \wedge \forall t', \\ t' < t \neg L_{i, t'}e(x, k_{\Sigma})) \wedge \neg (\exists y(R(i, t, y) \wedge \\ C(y, e(x, k_{\Sigma})))) \rightarrow \neg L_{i, t}(e(x, k_{\Sigma}))$$

$$2) \exists t, \exists x, \exists i, i \in \{ENT\}(\neg L_{i, t}k_{\Sigma}^{-1} \wedge \forall t', \\ t' < t \neg L_{i, t'}d(x, k_{\Sigma}^{-1})) \wedge \neg (\exists y(R(i, t, y) \wedge \\ C(y, d(x, k_{\Sigma}^{-1})))) \rightarrow \neg L_{i, t}(d(x, k_{\Sigma}^{-1}))$$

1) 说明如果主体 i 在 t 时刻没有获得公钥 k_{Σ} , 并且在小于 t 时刻的时间内没有获得密文 $e(x, k_{\Sigma})$, 且没有获得包含密文 $e(x, k_{\Sigma})$ 的消息, 那么在 t 时刻, 主体 i 不知道密文 $e(x, k_{\Sigma})$; 2) 同理。

1.3 CS 推理规则中的不足

1) 描述缺陷

CS 逻辑将解密运算和签名运算混为一谈。虽然两种运算都采用私钥, 但作用的对象不同。前者作用于密文, 且只对 $m = e(x, k_{\Sigma})$ 用 k_{Σ}^{-1} 解密有效; 后者可对于任何有意义的消息签名。所以应将两者区分开来。

2) 公理缺陷

首先, 公钥具有公开性, 任何主体可获得任何公钥, 进而产生 $e(x, k_{\Sigma})$, 而密文公理(1) 违背了公钥的公开性。再次, 由于签名和解密的混淆, 使得密文公理(2) 表述不清, 因为将 $d(x, k_{\Sigma}^{-1})$ 作为对 x 的解密时, 该公理没有实际意义。

未对知识和信仰与时间相关的不可知性进行描述。

未对与发送和接收密文有关的知识 and 信仰进行描述, 这使得 CS 逻辑对于一些信念和知识的推理十分冗长。比如因为没有签名消息的推理公理, 对于 $d(x, k_{\Sigma}^{-1})$ 这样一条消息, 要经过若干步推理才能得出主体 Σ 曾经发出过该消息。

3) 其他缺陷

没有考虑敌手 Eve 的知识和信仰。在形式化分析中, 考察敌手能够从协议中得到什么样的知识与信仰, 可以更清晰全面地分析协议的安全性。

2 对于 CS 逻辑已有的改进

文献[3]对 CS 协议进行了一些改进, 这些改进包括: 时间单调性规则的改进、包含规则的改进、发送和接收规则的改进、消息获取规则的改进等等, 使其可以分析 Timed-Release

协议^[4]的正确性。但是改进后的逻辑公理仍有若干不合理的地方:

1) 发送规则和密钥相结合的推理公理 I5(b2):

$$S(\Sigma, t, e(x, K_i)) \rightarrow B_{i, t'} \exists j \in ENT \setminus \{\Sigma, i\} \neg K(j, t', x)$$

该结论只有在消息 x 是新鲜的情况下才成立, 而逻辑中并没有给出 x 是新鲜的声明。

2) 时间单调性规则中的时间段的拆分规则 I2(c2):

$$\forall tn \leq t \leq ts, K(B)_{i, t}x \rightarrow \forall tn \leq t' \leq tm, K(B)_{i, t'}x \wedge \\ \forall tm \leq t'' \leq ts, K(B)_{i, t''}x$$

该规则是有歧义的。因为若 tn 为协议的开始时刻, ts 为协议的结束时刻, 该逻辑的描述为: 主体 i 在协议运行地 t 时知道消息 x , 那么主体 i 在协议中的任意时刻都知道消息 x 。这显然是不准确的。

3 扩展 CS 逻辑

我们在 CS 逻辑的基础之上不仅对一些公理进行了改进, 使其更加准确的反映公钥协议的安全性, 而且增加了一些逻辑符号和推理公理, 使得 CS 逻辑还可以反映对称密钥协议的安全性。

首先增加几个逻辑符号:

$x = x_1 \parallel x_2 \parallel \dots \parallel x_n$, 表示 x 由消息 x_1, x_2, \dots, x_n 串联组成。

$\oplus x$ 表示在一轮协议中消息 x 是保密的。

$\oplus_t x$ 表示消息 x 在时段 $[t_0, t]$ 中是保密的, 其中 t_0 表示协议运行的起始时间。

$\otimes x$ 表示 x 是公开的。

$Pub(x)$ 表示消息 x 中公开的部分, 若在 t 时有 $\otimes x_i$, 则 $x_i \in Pub_i(x)$ 。

$Fresh(x)$ 表示消息 x 是新鲜的, 即 x 在本轮协议之前没有出现过。

Eve 表示敌手, 是一个特殊的主体。敌手 Eve 可以控制整个网络中消息的传播, 可以认为传播中的消息均被 Eve 截获, 再由 Eve 转发给接收者。

Σ 和 Ψ 来表示任意合法主体, ENT 来表示包括敌手 Eve 在内的所有主体。

还需要对协议中消息的传播标注几个时间点。标记规则为: $ti(m_i)ti + 1$, 表示发送消息时间若为 ti , 则接收者收到消息的时间为 $ti + 1$ 。此外, 敌手 Eve 作为一个特殊主体在 $ti < t \leq ti + 1$ 时间内能够得到消息 m_i 。对于一个合法主体来说, m_{i+1} 消息的发送时间等于消息 m_i 的接收时间。

此外, 我们在协议中隐含用正确密钥进行的解密操作, 而将 $d(x, k_{\Sigma}^{-1})$ 明确定义为签名数据, 即 $d(x, k_{\Sigma}^{-1}) = (x, f_{k_{\Sigma}^{-1}}(x))$, 其中 x 是消息, $f_{k_{\Sigma}^{-1}}(x)$ 是签名值。我们认为签名数据中包含签名值和被签名的明文消息两部分, 这对于大部分的签名运算(签密除外)是合理的。

对 A1 的扩展:

$$1) K(B)_{\Sigma, t}p \wedge K(B)_{\Sigma, t}(q) \rightarrow K(B)_{\Sigma, t}(p \wedge q)$$

$$2) K_{\Sigma, t}\phi_{t'} \rightarrow t' \mid t' > t, K_{\Sigma, t'}\phi_{t'}$$

$$3) \neg K(L/B)_{\Sigma, t}x \rightarrow \forall t' < t \neg K(L/B)_{\Sigma, t'}x$$

2) 指出主体 Σ 在 t 时知道发生在 t' 时的表达式 ϕ , 则在 $t' > t$ 时, Σ 也知道发生在 t' 时的 ϕ ; 3) 指出若主体 Σ 在 t 时不相信或者不知道某消息, 则在 t 时以前均不相信或不知道该

消息。

对 A2 的修改和扩展:

$$1) \exists t, \exists x(S(\Sigma, t, x) \rightarrow L_{\Sigma, t}x \wedge L_{Eve, t+1}x)$$

敌手 *Eve* 控制着整个网络,发出的消息 *Eve* 均可得到。但若 *Eve* 将该消息删除,则接收者收不到该消息。所以主体 Σ 发出消息 x 后,所能确定的是 Σ 和 *Eve* 可以拥有消息 x 。

2) 发送签名消息

$$\exists t, \exists x(S(\Sigma, t, d(x, k_{\Sigma}^{-1})) \rightarrow L_{\Sigma, t}x \wedge L_{Eve, t+1}x)$$

同理 1)。

3) 发送加密消息

$$\exists t, \exists x(S(\Sigma, t, e(x, k_B) \wedge fresh(x)) \rightarrow \forall i, \\ i \in \{ENT/\Sigma, B\} \vdash L_{i, t+1}x)$$

当加密消息使用公钥 k_B 加密时,且 x 是新鲜的,那么在 $t+1$ 时刻,除了 Σ 和 B 外,没有其他主体(包括 *Eve*) 可以有意识的重新产生 x 。

$$\exists t, \exists x(S(\Sigma, t, e(x, k_{\Sigma B}) \wedge fresh(x) \wedge \oplus_i k_{\Sigma B}) \rightarrow \forall i, \\ i \in \{ENT/\Sigma, B\} \vdash L_{i, t+1}x)$$

当 $k_{\Sigma B}$ 是 Σ 和 B 的保密共享对称密钥,如果主体 Σ 在 t 时刻发送了加密消息 $e(x, k_{\Sigma B})$,且 x 是新鲜的,那么在 $t+1$ 时刻,除了 Σ 和 B 外,没有其他主体(包括 *Eve*) 可以有意识的重新产生 x 。

对 A3 的修改和扩展:

$$1) \exists t, \exists x(R(\Sigma, t, x) \rightarrow L_{\Sigma, t}x \wedge L_{Eve, t}x \wedge \exists i, \\ i \in \{ENT/\Sigma\} \exists t', t' < tS(i, t', x))$$

当主体 Σ 在 t 时收到一个消息 x ,则它在 t 时拥有 x ,且敌手 *Eve* 在 t 时拥有 x ,在 t 时以前一定有一个主体 i (包括敌手 *Eve*) 发送了该消息。

注:“ t 时以前”没有时间下界,包括本轮协议之前产生的消息。以下同。

2) 用私钥签名的消息

$$\exists t, \exists x(R(\Sigma, t, d(x, k_B^{-1})) \rightarrow L_{\Sigma, t}x \wedge L_{Eve, t}x, \\ \exists t' \mid t' < t, S(B, t', x))$$

当主体 Σ 在 t 时收到一个签名消息 $d(x, k_B^{-1})$,则它在 t 时拥有 x ,且敌手 *Eve* 在 t 时拥有 x ,在 t 时以前主体 B 发送了该消息。

3) 用公钥加密的消息

$$\exists t, \exists x, \exists i \mid i \in \{ENT/\Sigma\}, \\ R(\Sigma, t, e(x, k_i) \rightarrow \exists t' \mid t' < t, S(i, t', e(x, k_i)))$$

即当 Σ 在 t 时收到一个别人的公钥加密的消息,则在 t 时以前一定有一个主体 i (包括敌手 *Eve*) 发送了该消息。

$$\exists t, \exists x(R(\Sigma, t, e(x, k_{\Sigma})) \rightarrow L_{\Sigma, t}x \wedge \exists t' \mid 0 < t' < t, \\ \exists i \mid i \in \{ENT/\Sigma\} S(i, t', e(x, k_{\Sigma})))$$

即当 Σ 在 t 时收到一个自己公钥加密的消息,则 Σ 拥有该消息,且在 t 时以前一定有一个主体 i (包括敌手 *Eve*) 发送了该消息。

4) 用对称密钥加密的消息

$$\exists t, \exists x(R(\Sigma, t, e(x, k_{\Sigma B})) \wedge \oplus_i k_{\Sigma B} \rightarrow L_{\Sigma, t}x \wedge \\ \exists t' \mid t' < t, S(B, t', x))$$

即当 $k_{\Sigma B}$ 是 Σ 的对称密钥且没有被泄漏,如果主体 Σ 在 t 时接收到该加密消息,那么除了 Σ 和 B 外,没有任何主体能有意识的产生 x 。且在时间 t 前的某一时刻 t' ,主体 B 发送了该消息。

对密钥公理 A4 - d 的重新解释:

密钥公理 A4 - d 在文献[1] 中解释为私钥 k_{Σ}^{-1} 拥有者 Σ 知道用 k_{Σ}^{-1} 解密的密文。在完善 CS 逻辑中我们将表达式 $d(x, k_{\Sigma}^{-1})$ 定义为签名数据,因此该公理被重新解释为:签名者知道被签名的明文。

此外,由于密文公理 A5 用来推理加密消息和签名消息的获取,而该公理有前述的缺陷,所以将其摒弃。扩展后的逻辑公理完全可以推理与加密消息和签名消息有关的知识 and 信仰。

添加的公理:

新鲜性公理

消息 $x = x_1 \parallel x_2 \parallel \dots \parallel x_n$, 且 $1 \leq i \leq n$, 则:

$$fresh(x_i) \rightarrow fresh(x)$$

$$K_{\Sigma, t}(fresh(x)) \rightarrow \forall t' \mid t' > t, K_{\Sigma, t'}(fresh(x))$$

即若在 t 时刻, Σ 知道 x 是新鲜的,则在本轮协议的 $t' > t$ 时刻, Σ 知道 x 是新鲜的。

保密性公理

令 $x = x_1 \parallel x_2 \parallel \dots \parallel x_n$, 且 $1 \leq i \leq n$

在公钥加密的情况下:

$$K_{\Sigma, t}(R(\Sigma, t, e(x, k_{\Sigma})) \wedge x_i \notin pub_t(x) \wedge fresh(x)) \rightarrow \\ K_{\Sigma, t}(\oplus_i x_i)$$

即当 Σ 收到一个公钥加密的消息 x ,若 Σ 知道 x 是新鲜的,且 x_i 不属于 x 中公开的部分,则 Σ 知道 x_i 是保密的。

在 k 是对称密钥的情况下:

$$K_{\Sigma, t}(R(\Sigma, t, \{x\}_k) \wedge x_i \notin pub_t(x) \wedge fresh(x) \wedge \oplus_i k) \rightarrow \\ K_{\Sigma, t}(\oplus_i x_i)$$

即当 Σ 收到一个对称密钥 k 加密的消息 x ,若 Σ 知道 k 是保密的, x 是新鲜的,且 x_i 不属于 x 中公开的部分,则 Σ 知道 x_i 是保密的。

拥有对称密钥的判定公理:

若 $t' > t$, 则有:

$$K_{B, t'}(S(B, t, \{M\}_{k_{ab}})) \wedge K_{B, t'}(fresh(M)) \wedge \\ K_{B, t'}(L_{A, t'}(M)) \wedge K_{B, t'}(\oplus_{t'} k_{ab}) \rightarrow K_{B, t'}(L_{A, t'}(k_{ab}))$$

即,若 B 在 t 时发送了消息 $\{M\}_{k_{ab}}$,且在 $t' > t$ 时 B 知道 k_{ab} 是保密的、 M 是新鲜的、 A 拥有 M ,则 B 认为 A 拥有对称钥 k_{ab} 。

4 实例分析

4.1 Needham-Schroeder 对称密钥协议实例

$$\begin{array}{ll} 1) A \rightarrow S: A, B, Na & (M_1) \\ 2) S \rightarrow A: \{Na, B, k_{ab}, \{k_{ab}, A\}_{k_{bs}}\}_{k_{as}} & \{M_2\}_{k_{as}} \\ 3) A \rightarrow B: \{k_{ab}, A\}_{k_{bs}} & \{M_3\}_{k_{bs}} \\ 4) B \rightarrow A: \{N_b\}_{k_{ab}} & \{M_4\}_{k_{ab}} \\ 5) A \rightarrow B: \{N_b - 1\}_{k_{ab}} & \{M_5\}_{k_{ab}} \end{array}$$

为了下面分析时书写方便,将 5 个消息分别用 M_1, \dots, M_5 来代替。

4.2 协议分析

扩展逻辑对协议的分析包括以下几步:

- 1) 首先对协议中消息的流动进行时间标注。
- 2) 提出该协议的假设和前提。
- 3) 形式化说明协议将达到的目标。
- 4) 运用 CS 扩展逻辑中的规则和公理、前提和假设开始

推理,验证协议是否达成其最终目标。

第一步:时间标注如图 1。

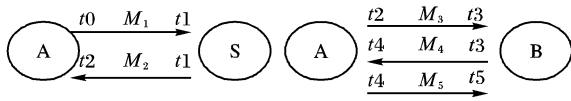


图 1 协议消息的时序

将协议中合法主体收发消息的时间顺序如图 1 排序。

第二步:前提和假设条件。

1) 有关密钥的假设:

a1: $\oplus k_{as}, \oplus k_{bs}, L_{A,0}(k_{as}), L_{B,0}(k_{bs}), L_{s,0}(k_{as}, k_{bs})$

即:协议主体 A、B 和可信中心 S 在协议运行开始时知道自己的对称密钥,且这些密钥是保密的。

2) 有关随机数的假设:

a2: $K_{A,0}(\text{fresh}(Na)), K_{B,0}(\text{fresh}(Nb))$

即:两个随机数都是本轮产生的,且主体 A 和 B 知道 S 产生新鲜的 k_{ab} 。

3) 其他假设:

a3: $L_{A,0}(\otimes A, \otimes B, \otimes Na), L_{B,0}(\otimes A, \otimes B)$

即:标识符 A 和 B 都是公开的。

第三步:协议的目标。

1) $K_{A,t_{end}}(\oplus_{t_{end}} k_{ab} \wedge L_{B,t_{end}} k_{ab})$

2) $K_{B,t_{end}}(\oplus_{t_{end}} k_{ab} \wedge L_{A,t_{end}} k_{ab})$

即在协议结束时, A 相信对称密钥 k_{ab} 是保密的,且相信 B 拥有 k_{ab} ; B 相信 k_{ab} 是保密的,且相信 A 拥有 k_{ab} 。这样 A 和 B 就认证了对方,并通过 k_{ab} 发送加密消息。

第四步:逻辑推理。

证明:

1) 如果在第二步主体 A 收到了消息,则下面的结论为真:

$K_{A,2}(R(A, t2, e(M_2, k_{as}))), K_{A,2}(\text{fresh}(Na)) \rightarrow K_{A,2}(\text{fresh}(M_2))$ (新鲜性公理), $K_{A,2}(k_{ab} \notin \text{pub}_2(M_2))$ 。那么根据公理 A1 - e 和保密性公理,有下面的结论成立:

$$K_{A,2}(\oplus_2 k_{ab}) \quad (1)$$

2) 如果在第三步主体 B 收到了消息,则下面的推理为真:

$K_{B,3}(R(B, t3, e(M_3, k_{bs}))), K_{B,3}(k_{ab} \notin \text{pub}_3(M_3))$, 但是由于不能判断消息 M_3 是否是新鲜的,所以此时无法利用保密性公理来得出结论:

$$K_{B,3}(\oplus_3 k_{ab}) \quad (2)$$

而从下面的推理可知,没有式(2),我们得不出协议目标。现在假设式(2)成立,可以继续分析协议。

3) 在第四步,主体 B 发送了消息 $\{M_4\}_{k_{ab}}$, 则有 $K_{B,3}(S(B, t3, \{N_b\}_{k_{ab}}))$, 加上前提假设 a2, 根据公理 A2 - c, 则可以下面的结论成立:

$$K_{B,3}(\forall i \mid i \in \{ENT/B, A\}, \neg L_{i,4}(Nb)) \quad (3)$$

此外,若主体 A 在第四步得到了消息 $\{M_4\}_{k_{ab}}$, 则 $K_{A,4}(R(A, t4, \{M_4\}_{k_{ab}}))$ 为真。但从前提假设可以看出, A 并不知道 Nb 是否是新鲜的,不能确认 Nb 是否被重放过,且 A 不知道在 $[t2, t4]$ 时间区间 k_{ab} 是否被泄露,所以根据保密性定理,推不出结论: $K_{A,4}(\oplus_{t4} Nb)$

4) 在第五步,若主体 B 接收到了消息 $\{M_5\}_{k_{ab}}$, 则有 $K_{B,4}(R(B, t5, \{M_5\}_{k_{ab}}))$; 根据 2) 中假设式(2): $K_{B,3}(\oplus_3 k_{ab})$ 成立; 根据前提假设 a2 和新鲜性公理有:

$$K_{B,3}(\text{fresh}(Nb)) \rightarrow K_{B,4}(\text{fresh}(Nb)) \quad (4)$$

假设 B 知道在 $[t1, t5]$ 时间区间 S 没有泄露 k_{ab} , 在 $[t3, t5]$ 时间区间 A 均没有泄露 k_{ab} , 且在 $[t4, t5]$ 时间区间 A 没有泄露 Nb , 则有 $K_{B,4}(\oplus_4 k_{ab})$ 和 $K_{B,4}(Nb \notin \text{pub}_4(M_4))$ 。那么根据公理 A1 - e 和保密性公理,下面的结论成立:

$$K_{B,4}(\oplus_4 Nb) \quad (5)$$

此外,由式(3)和公理 A1 - f 可知 $K_{B,4}(\forall i \mid i \in \{ENT/B, A\}, \neg L_{i,4}(Nb))$, 结合 $K_{B,4}(R(B, t5, \{Nb - 1\}_{k_{ab}}))$ 和公理 A1 - g、A3 - d, 我们可以分析:在 $t4$ 时,除了主体 A 和 B, 没有别的主体拥有 Nb , 但是在 $t5$ 时, B 却收到了 $Nb - 1$, 所以 B 知道在 $t4$ 时刻, A 拥有 Nb 则下列结论成立:

$$K_{B,4}(L_{A,4} Nb) \quad (6)$$

最后假设 B 知道在 $[t1, t5]$ 时间区间 S 没有泄露 k_{ab} , 在 $[t3, t5]$ 时间区间 A 没有泄露 k_{ab} , 由式(4)、式(6)、 $K_{B,4}(S(B, t3, \{N_b\}_{k_{ab}}))$, 根据拥有对称密钥的判定定理,下面的结论成立:

$$K_{B,4}(L_{A,4} k_{ab}) \quad (7)$$

通过利用扩展 CS 逻辑对 Needham-Schroeder 对称密钥协议实例的分析,得出该协议只能达到部分认证性,即不能让 A 确认 B; 而在 B 确认 A 时,还需要这样一些假设: 1) $\{M_3\}_{k_b}$ 应该是新鲜的,式(2)才成立。2) S 在 $[t1, t5]$ 时间区间没有泄露 k_{ab} , 且 A 在 $[t3, t5]$ 时间区间没有泄露 k_{ab} , 这样才能保证 $K_{B,4}(\oplus_4 k_{ab})$ 和式(7)成立。通过这两个假设,根据公理 A1 - e 我们才能得到协议目标 ii 。

对于式(5): $K_{B,4}(\oplus_4 Nb)$, 我们可以看出它和协议目标没有关系,因此为式(5)所做的假设“在 $[t4, t5]$ 时间区间内 Nb 是保密的”与协议目标无关。这对在实现该认证协议时,应该重点对协议中的哪些消息实施保密有指导意义。此外,这个结论为下面的想法提供了理论依据:既然 Nb 不需要保密,是否可以用具有数据完整性服务的单项函数来代替对称密钥加密来实现认证协议?我们将进一步进行研究。

5 结语

CS 扩展逻辑不仅能更好地反映主体知识和信仰与时间之间的关系,而且可以分析公钥和对称密钥两种协议。通过对 Needham-Schroeder 对称密钥协议实例的分析可以看出,扩展 CS 逻辑的一大优点在于能够挖掘出与时间相关的假设条件,这些假设条件是协议正确执行所必须的,但通常被人们所忽略掉。此外 CS 扩展逻辑分析信息的保密性,还可以指出协议中哪些信息的保密是至关重要的,哪些是不重要的,对协议的实施是采用什么样的算法具有指导意义。

参考文献:

- [1] COFFEY T, SAIDHA P. Logic for verifying public-key cryptographic protocols[J]. IEEE Proc. Computers and Digital Techniques, 1997, 144(1): 28 - 32.
- [2] BURROWS M, ABADI M, NEEDHAM R. A logic of authentication[R]. Research Report 39, Digital Systems Research Center, 1989. 17 - 24.
- [3] 范红, 冯登国. 一种分析 Timed-Release 公钥协议的扩展逻辑[J]. 计算机学报, 2003, 7: 832 - 838.
- [4] RONALD L, SHAMIR RA, WAGNER DA. Time-lock puzzles and timed-release cryptographic protocol[R]. Mit Laboratory for Computer Science, 1996.