第33卷 第23期 计 算 机 工 程 2007年12月
Vol.33 No.23 Computer Engineering December 2007
·安全技术· 文章编号：1000—3428(2007)23—0163—04 文献标识码：A 中图分类号：TP309

一种分析安全协议的新逻辑

刘英杰，姚正安
（中山大学数学与计算科学学院，广州 510275）

摘 要：提出了一种分析安全协议的新逻辑，既能有效地分析认证协议的认证性，又能分析电子商务协议的可追究性和公平性。该方法对认证协议的分析，不需要协议理想化，避免了因理想化而导致的各类问题。能够有效地分析电子商务协议的可追究性和公平性，用于分析实用协议。分析过程简单直观，便于实现机器自动验证。

关键词：安全协议；逻辑；可追究性；公平性；自动验证

New Logic for Analyzing Security Protocols

LIU Ying-jie, YAO Zheng-an
(School of Mathematics & Computational Science, Sun Yat-sen University, Guangzhou 510275)

【Abstract】 This paper presents a new logic which can be used to analyze security protocols. There is no necessity to idealize protocols when analyzing authentication protocols, which can avoid analysis errors caused by informal idealization. The new logic can be used to analyze accountability and fairness in electronic commerce protocols including the real-world protocols. The process of analyzing protocols is concise and can be implemented automatically.

【Key words】 security protocols; logic; accountability; fairness; automatic verification

自1989年由Burrows, Abadi和Needham提出的BAN逻辑[1]以来，用形式化方法分析安全协议逐渐成为当今的一大热点。许多学者对BAN逻辑进行了扩展和改进，其中SVO逻辑[2]标志着BAN类逻辑的成熟。但是，BAN类逻辑并不能分析电子商务协议的安全属性[3]。文献[3]提出了一种新的逻辑分析方法，记为Kailar逻辑。Kailar逻辑主要是分析电子商务协议的可追究性，但是Kailar逻辑同样存在着一些缺陷。Kailar逻辑通过构件canprove来证明可追究性，并没有用到信仰逻辑中的构件believe，从而 Kailar逻辑中的规则无法用BAN类逻辑中的语义来解释，即无法证明逻辑本身的正确性。Kailar逻辑也不能分析安全电子商务协议的公平性；在解释和分析协议语句时，也只能解释和分析那些签名的明文消息，而不能处理签名的加密消息和hash值。而现实中的电子商务协议并不只是通过签名明文来进行交易的，Kailar逻辑并不能分析现实中的实用协议。针对Kailar逻辑存在的缺陷，很多学者也在其基础上进行了改进[4~6]。但是文献[4~6]依然存在着缺陷，都无法同时分析认证协议的认证性和电子商务协议的可追究性和公平性。

为了解决以上文献中存在的问题，本文提出了一种分析安全协议的新逻辑。我们仍然引用 Kailar 逻辑中的 canprove 构件，为方便起见，采用文献[6]中的符号“ $\succ$ ”表示 canprove，但对它进行了新的定义。另外，将新逻辑映射到文献[2]提出的语义理论上来，自然地将“ $\succ$ ”(canprove)和“ $\mid\equiv$ ”(believe)构件结合在一起，从而一方面保证了逻辑本身的正确性，另一方面又可以分析认证协议的认证性。

1 一种分析安全协议的新逻辑

1.1 基本符号

主要介绍本文引入的符号，其他的符号含义与文献[7]相同，因篇幅有限，不再详细说明。

以下说明中的 X 指的是公式，M 是消息。其中公式和消息的定义与文献[7]的相同。

(1) $P \succ X \ \ to\ V$：主体 P 通过向第三方 V 发送非秘密性的消息使得 V 相信公式 X。非秘密性的消息一般对具体的安全协议有着具体的定义，比如，在支付协议中，买方不希望第三方知道自己的定单，这样，定单对用户来说就是秘密性的消息。

(2) $P \mid\equiv X$ ：主体 P 相信公式 X 是真的。

(3) $P \mid\neq X$ ：主体 P 不相信公式 X。

(4) $P \mid\sim M$：主体 P 发过消息 M，但不一定对 M 负责，只有在本轮协议中发过 M 才对其负责。

(5) $P \mid\approx M$ ：主体 P 最近发过消息 M，即在本轮协议中主体 P 发过消息 M。

(6) $Link(N_p)$ ：用于联系一个响应与一个请求。当主体生成临时值 N_p 时，即公式 $Link(N_p)$ 加入到这个主体的信念集合中。以后 $Link(N_p)$ 只允许收到一次包含 N_p 的消息，一旦收到这样的消息，该主体就从其信念集合中将 $Link(N_p)$ 删掉。

(7) $P \overset{M}{\leftarrow} Q$ ：主体 P 主动从主体 Q 处取到消息 M。

(8) $p \mapsto M$: 主体 P 生成消息 M。

(9) $match(M, h(M))$ ： $h(M)$ 为 M 的 hash 值。

1.2 两个定义

定义 1 拥有集合 O_p^i :该集合表示当协议的第 $i\ (1 \le i \le n)$

作者简介：刘英杰(1981－)，男，硕士研究生，主研方向：信息安全，安全协议的形式化分析与设计；姚正安，教授、博士生导师
收稿日期：2007-01-25 **E-mail：**liuyingjie1114@163.com

条语句执行完后，主体 P 的拥有集合，记为 O_P^i。在协议开始之前，主体 P 的初始拥有集合记为 O_P^0，它包含环境分配给主体 P 的密钥和 P 能证明的公式。当协议经过 n 步运行终止时，用 O_p 记主体 P 的最终拥有集合，即 $O_P = O_P^n$。显然，如果 $P \ni M$，那么 $M \in O_P^i$。

定义 2 信念集合 B_p^i :该集合表示当协议的第 $i\,(1 \le i \le n)$ 条语句执行完后，主体 P 的信念集合为 B_p^i。在协议开始之前，主体 P 的初始信念集合记为 B_P^0，它包括主体 P 相信环境分配的密钥的新鲜性和针对具体的协议主体 P 的一些具体的可信假设。当协议经过 n 步运行终止时，用 B_P 记主体 P 的最终信念集合，即 $B_P^n = B_P$。显然，如果 $P \models X$，那么 $X \in B_p^i$。如果 $P \not\models X$，那么 $X \notin B_p^i$。反之亦然。

1.3 两个集合的生成规则

1.3.1 O_P^i 的生成规则

当协议执行完第 i 条语句时，主体 P 的拥有集合由 O_P^{i-1} 到 O_P^i (i=1,2,⋯,n)的改变，遵循以下规则：

(1)如果协议的第 i 条语句为 $P \to Q : M$，不妨记 M 的形式为 $M = (\{M'\}_{K'}, \{M''\}_{K''}, \cdots)$。其中，$\{M'\}_K, \{M''\}_{K''} \notin O_P^{i-1}, K', K'' \in O_P^{i-1}$，则 $O_p^i = O_P^{i-1} \cup \{M, M', M'', \cdots\}$。

(2)如果协议的第 i 条语句为 $Q \to P : M$，或者 $P \overset{M}{\leftarrow} Q$，那么 $O_p^i = O_p^{i-1} \cup \{M\}$。

(3)如果协议的第 i 条语句为 $Q \to R : M$，其中 $R \ne P$，那么 $O_p^i = O_p^{i-1}$。

(4)如果 $(M_1, M_2) \in O_p^i$，则 $M_1 \in O_p^i, M_2 \in O_p^i$；反之，如果 $M_1 \in O_p^i, M_2 \in O_p^i$，则 $(M_1, M_2) \in O_p^i$。

(5)如果 $\{M\}_K \in O_p^i$且$K \in O_p^i$，那么 $M \in O_p^i$。反之，如果 $M \in O_p^i$ 且 $K \in O_P^i$，则 $\{M\}_K \in O_p^i$。

(6)如果 $M \in O_p^i$，则 $F(M) \in O_p^i$。特别是，$h(M) \in O_p^i$。

1.3.2 B_p^i 的生成规则

当协议执行完第 i 条语句时，主体 P 的信念集合由 B_P^{i-1} 到 B_P^i (i=1,2,⋯,n)的改变，遵循以下规则:

(1)如果协议的第 i 条语句为 $P \to Q : M$，若 P 是转发消息 M，则 $B_p^i = B_p^{i-1}$。若 P 生成消息 M，则 $B_p^i = B_p^{i-1} \cup \{\#(M)\}$。如果 P 生成随机数 N_P，则 $B_p^i = B_p^{i-1} \cup \{\#(N_p), Link(N_p)\}$。

(2)如果协议的第 i 条语句为 $Q \to R : M$。其中，$R \ne P$，则 $B_p^i = B_p^{i-1}$。

(3)如果协议的第 i 条语句为 $Q \to P : M$，则根据以下规则推理 B_p^i ：

1) $P \ni \{M\}_K \wedge P \ni K \wedge P \models \#(K) \wedge P \models Q \ni K \Rightarrow P \models Q \mid\sim M \wedge Q \ni M$ ，即 $B_P^i = B_P^{i-1} \cup \{Q \mid\sim M\}$。 $P \ni \{M\}_{K_q^{-1}} \wedge P \ni K_q \Rightarrow P \models Q \mid\sim M$，即 $B_P^i = B_P^{i-1} \cup \{Q \mid\sim M\}$。

2) $P \models Q \mid\sim M \wedge P \models \#(M) \Rightarrow P \models Q \mid\approx M$，即 $B_P^i = B_P^{i-1} \cup \{Q \mid\approx M\}$。

3) $\#(M_i) \Rightarrow \#(M_1, M_2, \cdots, M_n)$，即 $B_P^i = B_P^{i-1} \cup \{\#(M_1, M_2, \cdots, M_n)\}$。

4) $\#(X_1, X_2, \cdots, X_n) \Rightarrow \#(F(X_1, X_2, \cdots, X_n))$ $B_P^i = B_P^{i-1} \cup \{\#F(X_1, X_2, \cdots, X_n)\}$。

5) $P \models X \wedge P \models Y \Rightarrow P \models (X \wedge Y)$，即 $B_P^i = B_P^{i-1} \cup \{X \wedge Y\}$。

6) $P \models \#(K) \wedge P \models Link(N_P) \wedge P \ni K \wedge f(N_P)$ in M $\wedge$ m in M $\wedge P \ni \{M\}_K \wedge M \notin O_P^{i-1}) \Rightarrow P \models \#(m) \wedge P \not\models Link(N_P)$ ，即 $B_P^i = (B_P^{i-1} - Link(N_P)) \cup \{\#(m)\}$。

7) $Q \mid\sim (M_1, M_2, \cdots, M_n) \Rightarrow Q \mid\sim M_j$, $j = 1,2,\cdots,n$ ，即 $B_P^i = B_P^{i-1} \cup \{Q \mid\sim M_j\}$; $Q \mid\approx (M_1, M_2, \cdots, M_n) \Rightarrow Q \mid\approx M_i$, $j = 1,2,\cdots,n$ ，即 $B_P^i = B_P^{i-1} \cup \{Q \mid\approx M_j\}$。

8) $P \models Q \mid\approx \{M\}_K \wedge P \ni K \wedge Q \ni K \Rightarrow P \models \{Q \ni M\}$，即 $B_P^i = B_P^{i-1} \cup \{Q \ni M\}$。

9) $P \models (Q \ni M) \wedge P \models Q \mid\approx M \Rightarrow P \models \{Q \mid\approx \#(M)\}$，即 $B_P^i = B_P^{i-1} \cup \{\#(M)\}$。

1.4 基本规则

(1)MP 规则(model ponens)：由|- φ 和 $\varphi \Rightarrow \psi$，则可以推出 φ。

(2)Nec 规则：由|- φ，可推导出 $P \models \varphi$。

限于篇幅，没有详细解释以下每条规则的含义。

(1) R_1 (签名规则)：

$$P \ni \{X\}_{K_q^{-1}} \wedge P \succ \overset{k_q}{\rightarrow} Q \wedge P \models (V \ni X \Rightarrow V \models \#(X)) \Rightarrow P \succ Q \to X$$

(2) R_2 (Hash 规则)：

$$P \succ Q \to h(X) \wedge P \succ match(X, h(X)) \Rightarrow P \succ Q \to X$$

(3) R_3 (密文理解规则)：

$$P \succ Q \to \{X\}_K \wedge P \succ Q \ni K \Rightarrow P \succ Q \to X$$

(4) R_4 (管辖规则)：

$$P \succ Q \mid\Rightarrow X \wedge P \succ Q \to X \Rightarrow P \succ X$$

(5) R_5 (连接规则)：

$$P \succ Q \to (X, Y) \Rightarrow P \succ Q \to X \wedge P \succ Q \to Y$$
$$P \succ Q \to X \wedge P \succ Q \to Y \Rightarrow P \succ Q \to (X, Y)$$
$$P \succ X \wedge P \succ Y \Rightarrow P \succ (X \wedge Y)$$
$$P \succ (X \wedge Y) \Rightarrow P \succ X \wedge P \succ Y$$

(6) R_6 (消息含义规则)：

$$P \succ Q \mid\approx X \equiv P \succ Q \to X$$

(7) R_7 (推理规则)：

$$P \succ X \wedge (X \Rightarrow Y) \Rightarrow P \succ Y$$
$$P \models X \wedge P \models (X \Rightarrow Y) \Rightarrow P \models Y$$

(8) R_8 (信仰与可证明性关系规则)：

$$P \succ X \quad to \quad P \Leftrightarrow P \models X$$
$$P \succ X \; to \; V \Rightarrow V \models X$$

1.5 协议分析的具体步骤

1.5.1 认证协议的分析步骤

(1)列出所有主体的初始拥有集合和初始信念集合。

(2)分析协议目的，列出协议的最终目标。

(3)从协议的第 1 条语句开始，对协议的每一条语句进行动态分析，逐步更新 O_p^i和B_P^i，$i = 1,2,\cdots,n$。一般是先更新 O_p^i 后，再更新 B_p^i。当协议结束时，通过检查拥有集合判断是否达到了拥有目标，通过检查信念集合判断是否达到了信念目标。

1.5.2 电子商务协议的分析步骤

(1)列出所有主体的初始拥有集合和初始信念集合。

(2)分析协议目的，列出协议的目标，包括可追究性目标和公平性目标。

(3)可追究性分析

1)列出发方不可否认证据 EOO，收方不可否认证据 EOR 和传递方不可否认证据 EOD。在协议中，EOO, EOR 和 EOD 是协议设计者明确定义了的。

2)假设（EOO, EOD）$\in O_q$, （EOR, EOD）$\in Op$和（EOO, EOR）$\in O_A$，是否能够推出 $G_{Pi} \in B_p$, $G_{Qi} \in B_q$ 和 $G_{Ai} \in B_a$。其中，G_{Ji} 为主体J的信念目标，即证明某个主体对某个公式负有责任。J=P, Q, A，

$i=1,2,\cdots,n$。

3)分析协议是否达到可追究性目标，即协议结束后是否满足 $(EOO,EOD)\in O_q$，$(EOR,EOD)\in O_p$ 和 $(EOO,EOR)\in O_a$。

(4)公平性分析

协议满足公平性等价于对于任何第 i 条可中断的协议语句，$(EOO,EOD)\in O_q^{i-1}$ 当且仅当 $(EOR,EOD)\in O_p^{i-1}$。

2 实例分析

下面通过分析NSSK认证协议[8]和iKP协议[9]来看笔者的逻辑是如何推理的。

2.1 NSSK 协议的分析

NSSK 协议是 1978 年 Needham 和 Schroeder 提出的，现有的许多协议都是在这个协议的基础上修改而来的。协议具体描述如下：

(1) $A\to S:A,B,N_a$

(2) $S\to A:\{N_a,B,K_{ab},\{K_{ab},A\}_{K_{bs}}\}_{K_{as}}$

(3) $A\to B:\{K_{ab},A\}_{K_{bs}}$

(4) $B\to A:\{N_b\}_{K_{ab}}$

(5) $A\to B:\{N_b-1\}_{K_{ab}}$

协议的分析过程如下：

(1)列出初始拥有集合和初始信念集合

$O_a^0=\{K_{as}\}$，$B_a^0=\{\#(K_{as})\}$，$O_b^0=\{K_{bs}\}$，$B_b^0=\{\#(K_{bs})\}$

(2)列出协议认证目标

$G_{A1}:K_{ab}\in O_a$，$G_{A2}:\#(K_{ab})\in B_a$，$G_{A3}:(K_{ab}\in O_b)\in B_a$，$G_{A4}:(\#(K_{ab})\in B_b)\in B_a$。

$G_{B1}:K_{ab}\in O_b$，$G_{B2}:\#(K_{ab})\in B_b$，$G_{B3}:(K_{ab}\in O_a)\in B_b$，$G_{B4}:(\#(K_{ab})\in B_a)\in B_b$。

(3)对协议进行动态分析

由语句(1)以及拥有集合和信念集合的生成规则，得

$O_a^1=\{K_{as},N_a,A,B\}$，$B_a^1=\{\#(K_{as}),\#(N_a),Link(N_a)\}$

由语句(2)，容易得到 $O_a^2=O_a^1\cup\{K_{ab},\{K_{ab},A\}_{K_{ab}}\}$。由 $A|\equiv\#(K_{as})$，$A|\equiv Link(N_a)$，$A\ni K_{as}$，N_a *in* $\{N_a,B,K_{ab},\{K_{ab},A\}_{K_{bs}}\}_{K_{as}}$，从而由信念生成规则可知，$P|\equiv\#(K_{ab})$，即

$B_a^2=\{\#(K_{as}),\#(K_{ab}),\#(N_a),\#\{K_{ab},A\}_{K_{bs}}\}$

由语句(3)知 A 只是转发收到的消息，由生成规则得

$O_a^3=O_a^2,B_a^3=B_a^2$

由语句(4)得 $O_a^4=O_a^3\cup\{N_b\}$。

因为在 B_a^3 中并没有对应的 *Link* 请求，所以 A 不认为 $\{N_b\}_{K_{ab}}$ 为新鲜的，这样也就无法判断是否是 B 发送过来的，即协议不满足 G_{A3} 和 G_{A4}。

同样对主体 B 进行分析，当 B 收到 $\{K_{ab},A\}_{bs}$ 时，由于 B_b^2 中并没有 *Link* 请求存在，因此无法判断 K_{ab} 的新鲜性，即协议不满足 G_{B2} 和 G_{B4}。

文献[7]中指出的两种对 NSSK 协议的攻击说明了笔者分析结果的正确性。

2.2 电子商务协议 3KP 协议的分析

*i*KP 协议(*i*=1,2,3)是 IBM 公司提出的一组公开的基于信用卡/账号的安全电子支付协议，3KP 协议的安全性和复杂性最大。在本文中用新逻辑只分析 3KP 协议。3KP 协议的基本模型如下:

$Initiate:B\to S:SALT_B,ID_B$

$Invoice:S\to B:Clear,\{h(Common)\}_{K_S^{-1}}$

$Payment:B\to S:\{Slip\}_{K_A},\{\{Slip\}_{K_A},h(Common)\}_{K_B^{-1}}$

$Auth-Request:S\to A:Clear,h(SALT_B,DESC),\{Slip\}_{K_A},\{h(Common)\}_{K_S^{-1}},\{\{Slip\}_{K_A},h(Common)\}_{K_B^{-1}}$

$Auth-Request:A\to S:RESPCODE,\{RESPCODE,h(Common)\}_{K_A^{-1}}$

$Confirm:S\to B:RESPCODE,\{RESPCODE,h(Common)\}_{K_A^{-1}},\{h(Common)\}_{K_S^{-1}},V/VC$

协议中符号的详细含义参见文献[9]，限于篇幅，这里省略。分析过程如下：

2.2.1 可追究性分析

(1)列出初始拥有集合和初始信念集合。

$O_B^0=\{K_B,K_B^{-1},K_S,K_A,DESC\}$

$B_B^0=\{\#(K_B),\#(K_B^{-1}),\#(K_A),B\succ\xrightarrow{K_S}S,B\succ\xrightarrow{K_A}A\}$

$O_S^0=\{K_S,K_S^{-1},K_B,K_A,DESC\}$

$B_S^0=\{\#(K_S),\#(K_S^{-1}),\#(K_A),S\succ\xrightarrow{K_B}B,S\succ\xrightarrow{K_A}A\}$

$O_A^0=\{K_A,K_A^{-1},K_B,K_S\}$

$B_A^0=\{\#(K_A),\#(K_A^{-1}),A\succ\xrightarrow{K_B}B,A\succ\xrightarrow{K_S}S\}$

(2)列出 *EOO*, *EOR* 和 *EOD*。

$EOO=\{\{Slip\}_{K_A},h(Common)\}_{K_B^{-1}}$

$EOR=\{h(Common)\}_{K_S^{-1}}$

$EOD=\{RESPCODE,h(Common)\}_{K_A^{-1}}$

(3)列出可追究性目标。

$G_{B1}:B\succ S\to(B,S,Price,DATE)$

$G_{B2}:B\succ A\to(A,B,Price,DATE)$

$G_{B3}:B\succ S\to(B,S,DESC,DATE)$

$G_{S1}:S\succ B\to(B,S,Price,DATE)$

$G_{S2}:S\succ A\to(A,S,Price,DATE)$

$G_{S3}:S\succ B\to(B,S,DESC,DATE)$

$G_{A1}:A\succ B\to(A,B,Price,DATE)$

$G_{A2}:A\succ S\to(A,S,Price,DATE)$

(4)分析 *EOO*,*EOR* 和 *EOD* 的设计是否符合可追究性要求。

假定（*EOO*，*EOD*）$\in O_S$，即

$\{\{Slip\}_{K_A},h(Common)\}_{K_B^{-1}}$

$\{RESPCODE,h(Common)\}_{K_A^{-1}}\in O_S$

因为 $\{\{Slip\}_{K_A},h(Common)\}_{K_B^{-1}}\in O_S$，$K_B\in O_S^0\subset O_S$，所以由拥有集合生成规则，得

$$h(Common)\in O_S \tag{1}$$

由 Initiate 语句，知 $SALT_B\in O_S^1\subset O_S$，$ID_B\in O_S^1\subset O_S$。又 $DESC\in O_S^0\subset O_S$，$V\in O_S^2\subset O_S$，$VC\in O_S^2\subset O_S$，从而有

$$Common\in O_S \tag{2}$$

由式(1)和式(2)，得

$$(Common,h(Common))\in O_S \tag{3}$$

显然由式(3)，得

$$S\succ macth(Common,h(Common)) \tag{4}$$

由 $DATE\in O_S^2,\#(DATE)\in B_S^2\subset B_S$，$DATE\in Common$，运用信念生成规则，得

$$\#(h(Common))\in B_S \tag{5}$$

因为 *DATE* 为时戳，显然有 $V\ni DATE\Rightarrow V|\equiv\#(DATE)$，从而 $V\ni Common\Rightarrow V|\equiv\#(Common)$，进一步有

$$V\ni h(Common)\Rightarrow V|\equiv\#(h(Common)) \tag{6}$$

因为 $(S \succ \overset{K_B}{\rightarrow} B) \in B_S^0 \subset B_S$，运用签名规则，得

$$S \succ B \rightarrow h(Common) \quad (7)$$

由式(4)和式(7)，运用 Hash 规则，得 $S \succ B \rightarrow Common$。由连接规则，得 $S \succ B \rightarrow (B, S, \mathrm{Price}, DATE)$，由 NEC 规则，知 $G_{S1} \in B_S$。从而达到 G_{S1} 目标。

根据以上思路，可以推得 $G_{S2} \in B_S$，$G_{S3} \in B_S$。同样，假设 $(EOR, EOD) \in O_B$，可推得 $G_{B1}, G_{B2}, G_{B3} \in B_B$。假设 $(EOO, EOR) \in O_A$，可得 $G_{A1}, G_{A2} \in B_A$。限于篇幅，详细推理步骤省去。

(5)分析协议能否达到可追究性目标。即协议结束后是否有 $(EOO, EOD) \in O_S$，$(EOR, EOD) \in O_B$，$(EOO, EOR) \in O_A$。

因为

$$\{\{Slip\}_{K_A}, h(Common)\}_{K_B^{-1}} \in O_S^3 \subset O_S$$

$$\{RESPCODE, h(Common)\}_{K_A^{-1}} \in O_S^5 \subset O_S$$

所以有 $(EOO, EOD) \in O_S$。

因为

$$\{h(Common)\}_{K_S^{-1}} \in O_B^2 \subset O_B$$

$$\{RESPCODE, h(Common)\}_{K_A^{-1}} \in O_B^6 = O_S$$

所以有 $(EOR, EOD) \in O_B$。

因为

$$\{\{Slip\}_{K_A}, h(Common)\}_{K_N^{-1}} \in O_A^4 \subset O_A$$

$$\{h(Common)\}_{K_B^{-1}} \in O_A^4 \subset O_A$$

所以 $(EOO, EOR) \in O_A$，协议达到可追究性目标。

2.2.2 公平性分析

即证明：

$(EOO, EOD) \in O_S^{i-1}$ 当且仅当

$(EOR, EOD) \in O_B^{i-1}, i = 1,2,3,4,5,6$

(1)分析 O_S^5

$$O_S^5 = O_S^4 \cup \{RESPCODE, \mathrm{h(Common)}\}_{K_A^{-1}}$$

$$O_S^4 = O_S^3 = O_S^2 \cup \{\{Slip\}_{K_A}, h(Common)\}_{K_B^{-1}}$$

因此，$O_S^5 = O_S^2 \cup (EOO, EOD)$，即 $(EOO, EOD) \in O_S^5$。

(2)现在分析 O_B^5

$$O_B^5 = O_B^3 = O_B^2 \cup \{Slip\}_{K_A}, h(Common)\}_{K_B^{-1}}$$

$$= O_B^1 \cup \{Clear, \{h(Common)\}_{K_S^{-1}}\} \cup \{\{Slip\}_{K_A}, h(Common)\}_{K_S^{-1}}$$

可见，$(EOR, EOD) \notin O_B^5$。因此 3KP 协议不满足公平性。

3 与其他逻辑的比较

表 1 对现有的逻辑和新的逻辑进行了比较。

表 1 现有逻辑和新逻辑在功能上的比较

	认证性	可追究性	公平性	实用协议
BAN 类逻辑	√			
Kailar 逻辑		√		
Zhou 逻辑		√	√	
KN 逻辑	√	√		
KP 逻辑		√		√
新逻辑	√	√	√	√

表 1 中“√”表示逻辑能够分析安全协议的属性。文献[6]提出的逻辑称为 Zhou 逻辑，文献[4]提出的逻辑称为 KN 逻辑，文献[5]提出的逻辑称为 KP 逻辑。与其他逻辑相比较，新的逻辑在功能上有了进一步的扩展和增强。同时，由实例分析的过程可以看出，新逻辑同样具有简单、直观的特点。另外由于在逻辑中定义了拥有集合和信念集合两个概念，并且一直贯穿于协议分析的始终，从而为实现安全协议的自动验证提供了有力的工具。总体来说，与现有的逻辑相比，本文有以下优点：

(1)分析认证协议时，无须协议的理想化过程。能够形式化地引入初始假设；同时增加了新鲜性子消息规则，从而能够更好地抵御重放攻击。

(2)分析电子商务协议时，不仅增加了密文理解规则，而且增加了 Hash 规则，同时对签名规则做了进一步完善，从而能够更广泛地分析电子商务协议，尤其是实用协议。

(3)不仅能够形式化的分析电子商务协议的可追究性，同时能够有效地分析公平性。

(4)易于实现安全协议的机器自动证明。

4 结束语

本文提出了一种能够同时分析认证协议以及实用电子商务协议的新逻辑，并对 NSSK 协议和 3KP 协议运用新逻辑进行了形式化分析，分析得出 NSSK 协议不满足认证属性，3KP 协议满足可追究性，但是不满足公平性。

以后的研究工作将集中在完成一个基于这个新的逻辑验证模型的安全协议的机器自动验证系统，并探讨如何将这个新的逻辑扩展到安全协议的设计领域，使得扩展后的逻辑不仅能够验证安全协议的正确性，同时能够指导如何设计出更为安全的协议。

参考文献

1 Burrows M, Abadi M, Needham R. A Logic of Authentication[Z]. Digital Systems Research Center, Pao Alto, California, 1989.

2 Syverson P, Oorschot P. On Unifying Some Cryptographic Protocol Logics[C]//Proc. of the IEEE Computer Society Symp. on Security and Privacy. 1994: 346-347.

3 Kailar R. Accountability in Electronic Commerce Protocols[J]. IEEE Transactions on Software Engineering, 1996, 2(5): 313-328.

4 Kessler V, Neumann H. A Sound Logic for Analyzing Electronic, Commerce Protocols[C]//Proceedings of ESORICS'98. 1998.

5 Kungpisdan S, Permpoontanalarp Y. Practical Reasoning About Accountability in Electronic Commerce Protocols[C]//Proceedings of the 4th International Conference on Information Security and Cryptology, Seoul, Korea. 2001-12-06.

6 周典萃, 卿斯汉, 周展飞. 一种分析电子商务协议的新工具[J]. 软件学报, 2001, 12(9): 1318-1328.

7 卿斯汉. 安全协议[M]. 北京: 清华大学出版社, 2005.

8 Needham R, Schroeder M. Using Encryption for Authentication in Large Networks of Computers[J]. Communications of the ACM, 1978, 21(12): 993-999.

9 Bellare M, Garay J A, Hauser R, et al. Design, Implementation and Deployment of the iKP Secure Electronic Payment System[J]. IEEE Journal of Selected Areas in Communications, 2000, 18(4): 611-627.