一种验证非否认协议的新方法

周 勇　　朱梧槚
(南京航空航天大学信息科学与技术学院　南京　210016)

摘　要：为了描述非否认协议中的各种不确定因素，在 Kailar 逻辑系统中引入了表示缺省信息的否定词，以及相应的推理机制。提出了安全协议验证的新方法，主要特点是：可以直接对协议的动态运行过程进行推理；推理具有非单调性；避免过多的理想化假设；可以分析含有多个子协议的非否认协议，以及协议的可追究性和公平性。文中以一种基于离线 TTP 方式的非否认协议为例，验证了该协议在运行一次时具有可追究性，但多次运行时存在攻击。

关键词：非否认协议；Kailar 逻辑；协议验证
中图分类号：TP309　　**文献标识码**：A　　**文章编号**：1009-5896(2007)10-2493-05

A New Verification Method for Non-repudiation Protocol

Zhou Yong　　Zhu Wu-jia
(*College of Information Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing* 210016, *China*)

Abstract: For the description of the nondeterministic factors in the non-repudiation protocols, the Kailar logic system extended with the default negation and the corresponding reasoning mechanism is introduced. The extended system can be used to verify security protocols and it has several main characteristics. Firstly, the method can reason not only for the results but also for the dynamic procedure of the protocol run. Secondly, the reasoning is nonmonotonic. Thirdly, the ideal assumptions of the protocols can be reduced. Fourthly, the accountability and fairness of the security protocols with some sub-protocols can be analyzed. As an example, a non-repudiation protocol with offline TTP was verified. The protocol has accountability during one protocol run and gets the attack in the repeated runs.

Key words: Non-repudiation protocol; Kailar logic; Protocol verification

1　引言

在电子商务等领域，以密码学为基础的安全协议对于保障通信各方的安全和利益起着重要的作用。非否认协议是安全协议的一种，除了需要满足安全性、保密性之外、还要求通信各方对自己的行为负责，满足可追究性。

非否认协议根据可信第三方(TTP)参与的方式可以分为两种：在线TTP型协议[1, 2]和离线TTP型协议[3–5]。前者是指每次协议运行都要求TTP参与，协议的表述形式比较简单，但TTP的负担较重，协议的效率也相对较低。后者是指在正常情况下，通信双方直接对话，无需TTP的参与，只有在通信意外中断后才要求TTP进行协调。

离线TTP型协议一般分为 3 个子协议，对该类协议的验证通常涉及协议的不同运行过程和状态。基于逻辑系统对非否认协议进行表示和验证是一种常用的方法。Kailar针对电子商务协议提出一个逻辑系统[6]，可以对协议的可追究性进行验证。周典萃等指出该逻辑系统存在着不能分析公平性以及无法处理密文等问题[7]，并进行了改进。另外，如果直接运用Kailar逻辑系统分析协议，则需要对协议结束后各方掌握的证据进行人工分析，给出各种理想化假设，无法对协议运行过程进行自动处理。特别是分析运行状态不确定的协议，需要更多的人工参与。

本文对 Kailar 逻辑系统进行扩充，引入表示缺省信息的否定词，以及表示环境假设的谓词，给出了新的逻辑推理机制和相应的协议验证方法。该方法可以自动对协议的运行环境进行分析，对协议的运行过程进行推理。

2　新的逻辑系统

2.1 基本符号

Kailar 逻辑系统中的语法符号涉及主体(Principals)、消息(Message)和构件(statement)。本文使用的构件有以下几种：

$\mathrm{E}(x,y)$：用密钥 x 对消息 y 加密后得到的密文；

P CanProve x：主体 P 能够证明 x；

2006-03-29 收到，2007-01-26 改回
国家自然科学基金(60573070)资助课题

P Received x：主体 P 收到消息 x；

P Said x：主体 P 说过消息 x；

P Has x：主体 P 拥有消息 x；

x Authenticates P：主体 P 对 x 负责。

在 Kailar 逻辑中有“强证明”构件 P CanProve x 和“弱证明”构件 P CanProve x to Q。前者表示主体 P 可对所有主体证明 x。文献[8]中指出，关于“强证明”构件的结论很容易推广到“弱证明”构件。所以本文只采用“强证明”构件。

为了形式化描述通道的状态和主体发送消息的动作，本文引入以下两种新的构件。

ChannelBroken(t)：在 t 时刻通信通道阻塞(消息无法被传递)；

P Sends $<Q,t,x>$：P 在时刻 t 向 Q 发送消息 x。

2.2 推理规则与公理

基于上述构件的公理模式表示如下：

A1 P CanProve x $\wedge$ P CanProve y $\rightarrow$ P CanProve (x,y)；

A2 P Received E(K^{-1},x) $\wedge$ P CanProve (K Authenticates Q)$\rightarrow$ P CanProve (Q Said x)；

A3 P Received $(x_1,x_2,\cdots,x_n)$ $\rightarrow$ P Received x_i；

A4 P Said $(x_1,x_2,\cdots,x_n)$ $\rightarrow$ P Said x_i；

A5 P CanProve (Q Said E(K,x)) $\wedge$ P CanProve (Q Has K) $\rightarrow$ P CanProve；(K Authenticaticates Q)

A6 P Received x $\rightarrow$ P Has x；

A7 P Sends $<Q,t,x>$ $\wedge\sim$ ChannelBroken(t)。 $\rightarrow$ Q Received x

其中 A1 到 A4 为 Kailar 系统中的，在原系统中表述为推理规则，这里给出的是相应的公理模式，在推理过程中可对主体和消息等进行代入。

A1 为连接规则，表示可以将被证明的公式进行组合。A2 为签名规则，表示如果主体 P 收到经 Q 签名的消息 x，则 P 能证明 Q 曾经说过 x。A3 和 A4 表示可以对收到的或说过的消息进行分割。由于本文主要讨论协议的可追究性和公平性，没有采用 Kailar 逻辑中的信任规则和强弱证明转换规则。A5 是文献[9]中提出的密文理解规则，用于分析加密消息的来源。

为了对协议过程进行推理，本文提出了 A6 和 A7 规则。A6 表示一旦 P 收到消息 x，则 P 就拥有 x。A7 表示如果主体 P 在时刻 t 向 Q 发送消息 x，并且在 t 时刻通信通道没有阻塞的话，则主体 Q 能够接收到消息 x。

A7 规则中含有否定词 $\sim$，表示缺省信息。其含义为：如果构件 H 未被证明，则 $\sim$H 为真，否则 $\sim$H 为假。将不含 $\sim$ 的构件 H 称为正构件，将 $\sim$H 称为负构件。如果 $\Phi\rightarrow\Psi$ 是规则，令 $\Phi^+=\{\mathrm{H}\mid$H 是正构件且为 Φ 中的一个合取项$\}$，$\Phi^-=\{\mathrm{H}\mid\sim$H 为 Φ 中的一个合取项$\}$。

由于新系统中允许出现否定词，其推理过程和 Kailar 逻辑系统不同。推理的基础是上述 7 条公理与对具体协议进行形式化后所得的协议规则。

定义 1 设 Π 是一个不含 $\sim$ 的构件集合，Σ 是具体协议规则的集合，如果 Π 是满足以下条件(1)，条件(2)，条件(3)的极小集合，则称之为 Σ 的一个稳定集。

(1) 如果 H 是构件且 $\mathrm{H}\in\Sigma$，则 $\mathrm{H}\in\Pi$；

(2) 对所有公理和 Σ 中的协议规则 $\Phi\rightarrow\Psi$，如果 $\Phi^+\subseteq\Pi$，$\Phi^-\cap\Pi=\varnothing$，则 $\Psi\in\Pi$；

(3) 如果 (P CanProve Φ) $\in\Pi$，$\Phi\rightarrow\Psi$ 是公理或协议规则，则 (P CanProve Ψ) $\in\Pi$。

稳定集表示协议验证的可能结果。由于缺省否定的存在，稳定集具有如下性质。

命题 1 一个规则集 Σ 可能存在多个稳定集，也可能不存在稳定集。

证明 令 $\Sigma_1=\{\sim(P$ CanProve $x)\rightarrow Q$ CanProve x, $\sim(Q$ CanProve $x)\rightarrow P$ CanProve x$\}$，则 Σ_1 有两个稳定集，分别含有 P CanProve x 和 Q CanProve x。又令 $\Sigma_2=\{\sim(P$ CanProve $x)\rightarrow P$ CanProve x$\}$，则 Σ_2 不存在稳定集。 证毕。

命题 2 存在构件集合 Π_1,Π_2 和规则集合 Σ_1，Σ_2，使得 Π_1 是 Σ_1 的稳定，Π_2 是 Σ_2 的稳定集且 $\Sigma_1\subseteq\Sigma_2$，但 $\Pi_1\subseteq\Pi_2$ 不成立。

证明 令 $\Sigma_1=\{\sim(P$ Received $x)\rightarrow P$ Send $<Q,t,y>\}$，$\Sigma_2=\Sigma_1\cup\{P$ Received $x\}$。设 Π_1 和 Π_2 分别是 Σ_1 和 Σ_2 的稳定集，则 P Sends $<Q,t,y>\in\Pi_1$，P Sends $<Q,t,y>\notin\Pi_2$，即 $\Pi_1\subseteq\Pi_2$ 不成立。 证毕

命题 1 表明协议中的不确定信息会导致不确定的推理结果。命题 2 表明基于稳定集的推理具有非单调性。在 Kailar 逻辑系统中，上述两个命题不成立。

2.3 安全协议的动态验证方法

基于上述逻辑系统进行协议验证的主要步骤如下：

(1) 建立协议运行的初始假设，如加密机制、通信通道情况等；

(2) 将具体的协议转化为相应的逻辑规则 Σ (Σ 中允许含有 $\sim$)；

(3) 将需要验证的目标进行形式化描述；

(4) 进行协议的推理、验证、分析。

3 一种非否认协议的形式描述与验证

3.1 一种离线 TTP 方式的非否认协议

非否认协议可分为在线TTP方式和离线TTP方式。下面以Kremer等提出的离线TTP型协议(以下简称KM协议)[4]为例进行形式化描述和验证。下面是协议中使用的简记符号。

f_{EOOC}, f_{EORC}，…：用于表明消息含义的标记；

$C=\mathrm{E}(K,M)$：用密钥 K 加密的消息 M；

$L=\mathrm{H}(M,K)$：消息 M 和密钥 K 的 Hash 函数；

$\mathrm{Sig}_X(Y)$：X 对消息 Y 的签名；

$\mathrm{EOOC}=\mathrm{Sig}_A(f_{\mathrm{EOOC}},B,\mathrm{TTP},L,\mathrm{H}(C))$：发送方对发送密文 C 的证据；

$\mathrm{EORC}=\mathrm{Sig}_B(f_{\mathrm{EORC}},A,\mathrm{TTP},L,\mathrm{H}(C))$：接收方对接收密文 C 的证据；

$\mathrm{SUBK}=\mathrm{Sig}_A(f_{\mathrm{SUB}},B,L,\mathrm{E}_{\mathrm{TTP}}(K))$：发送方将密钥 K 提交给 TTP 的证据；

$\mathrm{EOOK}=\mathrm{Sig}_A(f_{\mathrm{EOOK}},B,L,K)$：发送方对发送密钥 K 的不可否认证据；

$\mathrm{EORK}=\mathrm{Sig}_B(f_{\mathrm{EORK}},A,L,K)$：接收方对发送密钥 K 的不可否认证据；

$\mathrm{RecX}=\mathrm{Sig}_X(f_{\mathrm{RecX}},Y,L)$：$X$ 对 Recovery 子协议的请求（X 可以是 A 或 B）；

$\mathrm{ConK}=\mathrm{Sig}_{\mathrm{TTP}}(f_{\mathrm{ConK}},A,B,L,K)$：TTP 对密钥确认的证据；

$\mathrm{ConA}=\mathrm{Sig}_{\mathrm{TTP}}(f_{\mathrm{ConA}},A,B,L)$：TTP 对 Abort 确认的证据；

$\mathrm{Abort}=\mathrm{Sig}_A(f_{\mathrm{Abort}},B,L)$：发送方对执行 Abort 子协议的请求；

$\mathrm{Rec\_app}=(f_{\mathrm{RECX}},f_{\mathrm{SUB}},Y,L,\mathrm{H}(C),\mathrm{E}_{\mathrm{TTP}}(K),\mathrm{RecX},\mathrm{SUBK},\mathrm{EORC},\mathrm{EOOC})$：申请 Recovery 子协议所需信息；

KM 协议包含 3 个子协议：Main 子协议、Abort 子协议和 Resolve 子协议。

Main 子协议：

m1　$A\rightarrow B: f_{\mathrm{EOOC}},f_{\mathrm{SUB}},A,B,\mathrm{TTP},L,C,\mathrm{E}_{\mathrm{TTP}}(K),$；EOOC,SUBK

m2　$B\rightarrow A: f_{\mathrm{EORC}},A,\mathrm{TTP},L,\mathrm{EORC}$；IF A times out THEN abort；

m3　$A\rightarrow B: f_{\mathrm{EOOK}},B,L,K,\mathrm{EOOK}$；IF B times out THEN recovery；

m4　$B\rightarrow A: f_{\mathrm{EORK}},A,L,\mathrm{EORK}$；IF A times out THEN recovery。

Abort 子协议：

a1　$A\rightarrow \mathrm{TTP}: f_{\mathrm{Abort}},L,B,\mathrm{Abort}$；IF resolved OR aborted THEN stop ELSE aborted=true；

a2　$\mathrm{TTP}\rightarrow A: f_{\mathrm{ConA}},A,B,L,\mathrm{ConA}$；

a3　$\mathrm{TTP}\rightarrow B: f_{\mathrm{ConA}},A,B,L,\mathrm{ConA}$。

Recovery 子协议

r1　$X\rightarrow \mathrm{TTP}: f_{\mathrm{RecX}},f_{\mathrm{SUB}},Y,L,\mathrm{H}(C),\mathrm{E}_{\mathrm{TTP}}(K),\mathrm{RecX},$ SUBK,EORC,EOOC；IF aborted OR Recovered THEN stop ELSE Recovered=true；

r2　$\mathrm{TTP}\rightarrow A: f_{\mathrm{ConK}},A,B,L,K,\mathrm{ConK},\mathrm{EORC}$；

r3　$\mathrm{TTP}\rightarrow B: f_{\mathrm{ConK}},A,B,L,K,\mathrm{ConK}$。

3.2　KM 协议的形式化描述与验证

假定 KM 协议运行的通信环境满足以下条件：(1)完美的密码机制。(2)数字签名不能被伪造。(3)参与通信的主体为 A，B 和 TTP。(4) TTP 是可信的，对 A 和 B 的回应是及时可靠的。

在 KM 协议中允许 A 与 B 之间的通道阻塞，也允许 A 和 B 随时放弃协议的运行。以下先考虑协议只运行一次的情形。用集合 Σ_0 表示 KM 协议的初始假设，Σ_0 中包含以下内容：

(1)通道的状态　将主协议 4 个步骤的运行时刻分别记为 m_1，m_2，m_3 和 m_4，如果 $\mathrm{ChannelBroken}(m_i)\in\Sigma_0$，则表示在 m_i 时刻通道阻塞，否则表示通道在 m_i 时刻是畅通的。另外，假设 TTP 与各主体的通信通道是畅通的。

(2)主体的意愿　如果 $\mathrm{GiveUp}(P,m_i)\in\Sigma_0$，则表示主体 P 在 m_i 时刻放弃协议的继续运行。

利用逻辑构件可以直接将 KM 协议转化为以下规则：

P1　A Sends $<B,m_1,(f_{\mathrm{EOOC}},f_{\mathrm{SUB}},A,B,\mathrm{TTP},L,C,\mathrm{E}_{\mathrm{TTP}}(K),\mathrm{EOOC},\mathrm{SUBK})>$；

P2　B Received EOOC $\wedge$ $\sim\mathrm{GiveUp}(B,m_2)$ $\rightarrow B$ Sends $<A,(f_{\mathrm{EORC}},A,\mathrm{TTP},L,\mathrm{EORC})>$；

P3　A Received EORC $\wedge$ $\sim\mathrm{GiveUp}(A,m_3)$ $\rightarrow A$ Sends $<B,(f_{\mathrm{EOOK}},B,L,K,\mathrm{EOOK})>$；

P4　B Received EOOK $\wedge$ $\sim\mathrm{GiveUp}(B,m_4)$ $\rightarrow B$ Sends $<A,(f_{\mathrm{EORK}},A,L,\mathrm{EORK})>$；

P5　$\sim(A\,\mathrm{Received\ EORC})\rightarrow A$ Sends $<\mathrm{TTP},(f_{\mathrm{Abort}},L,B,\mathrm{Abort})>$；

P6　TTP Received $(L,A,B,\mathrm{Abort})\wedge\sim\mathrm{Abort}\wedge\sim\mathrm{Recovered}\rightarrow$ TTP Sends $<A,(f_{\mathrm{ConA}},A,B,L,\mathrm{ConA})>$；

P7　TTP Received $(L,A,B,\mathrm{Abort})\wedge\sim\mathrm{Abort}\wedge\sim\mathrm{Recovered}\rightarrow$ TTP Sends $<B,(f_{\mathrm{ConA}},A,B,L,\mathrm{ConA})>$；

P8　TTP Received $(L,A,B,\mathrm{Abort})\wedge\sim\mathrm{Abort}\wedge\sim\mathrm{Recovered}\rightarrow\mathrm{Aborted}$；

P9　B Sends $<A,t,\mathrm{EORC}>\wedge\sim(B$ Received EOOK$)\rightarrow B$ Sends $<\mathrm{TTP},\mathrm{Rec\_app}>$；

P10　A Sends $<B,t,\mathrm{EOOK}>\wedge\sim(A$ Received EORK$)\rightarrow A$ Sends $<\mathrm{TTP},\mathrm{Rec\_app}>$；

P11　TTP Received $\mathrm{Rec\_app}\wedge\sim\mathrm{Abort}\wedge\sim\mathrm{Recovered}\rightarrow$ TTP Sends $<A,(f_{\mathrm{ConK}},A,B,L,K,\mathrm{ConK},\mathrm{EORC})>$；

P12　TTP Received $\mathrm{Rec\_app}\wedge\sim\mathrm{Abort}\wedge\sim\mathrm{Recovered}\rightarrow$ TTP Sends $<B,(f_{\mathrm{ConK}},A,B,L,K,\mathrm{ConK})>$；

P13 TTP Received Rec_app $\wedge$ $\sim$ Abort $\wedge$ $\sim$ Recovered $\rightarrow$ Recovered 。

其中 P1–P4 分别对 m1–m4, 在通信期间如果通道不畅或有主体放弃，则 P4 条件无法满足，代表 Main 子协议不能顺利完成。P5–P8 对应 a1–a3，表示 A 申请 Abort 子协议，TTP 给予回应。P9–P10 对应 r1,表示由 A 或 B 申请 Recovery 子协议。P11–P13 对应 r2 和 r3。

KM 协议的验证目标是：

G1 B CanProve (A Said M) ；

G2 A CanProve (B Received M) 。

G1 表示主体 B 对主体 A 的可追究性，G2 表示 A 对 B 的可追究性。

命题 3 设 Σ_0 是任意的 KM 初始假设集合，$\Sigma=\Sigma_0\cup\{P1,P2,\cdots,P13\}$，则 Σ 的稳定集中要么同时含有 G1 和 G2，要么同时不含有 G1 和 G2。

证明 Σ_0 所有可能的取值可划分为 5 类情形：(1) ChannelBroken(m_1) $\in\Sigma_0$ 。(2) ChannelBroken(m_2) $\in\Sigma_0$ 或 GiveUp(B,m_2) $\in\Sigma_0$ 。(3) ChannelBroken(m_3) $\in\Sigma_0$ 或 GiveUp(A,m_3) $\in\Sigma_0$ 。(4) ChannelBroken(m_4) $\in\Sigma_0$ 或 GiveUp (B,m_4) $\in\Sigma_0$ 。(5) $\Sigma_0=\varnothing$ 。

对于情形(1)，A7，P2 的条件不满足，P1，P5–P8 的条件满足，Σ 的稳定集中含有 Aborted , 不含 G1 和 G2。情形(2)时，P1,P2,P9,P11–13 的条件满足，P3 的条件不满足，Σ 的稳定集中含有 Recovered , G1 和 G2。情形(3)和情形(4)同样可得 Σ 的稳定集中含有 Recovered , G1 和 G2。情形(5)中 P1–P4 的条件为真，P5–P13 条件不满足，Σ 的稳定集中含有 G1 和 G2。情形(5)对应着 KM 协议如下的理想化假设：通道畅通并且通信方不会人为中断协议的运行。总之，5 种情形下命题 3 的结论都成立。 证毕

命题 3 表明 KM 协议在上述通信环境假设下具有可追究性。

3.3 KM 协议的一个漏洞

如果 KM 协议参与主体只有 A，B 和 TTP，并且协议只运行一次，可以证明以下结论。

命题 4 设 Σ_0 是任意的 KM 初始假设集合，$\Sigma=\Sigma_0\cup\{P1,P2,\cdots,P13\}$，如果 Σ 的稳定集中含有 B Received M ，那么也一定含有 A CanProve (B Received M) 。

证明 类似命题 3，对 Σ_0 的 5 种情形分别讨论，除情形(1) 的稳定集中不含 B Received M 和 A CanProve(B Received M) 。其它 4 种情况都同时含有这两个构件。证毕。

一旦允许 KM 协议重复运行，参与主体也不限于 A 和 B(可由 A 或 B 进行冒名代替)，则命题 2 不再成立。

命题 5 设 Σ_0 是 KM 初始假设集合，$\Sigma=\Sigma_0\cup\{P1,P2,\cdots,P13\}$，如果允许对 P1–P13 中的主体 A 和 B 以及相应签名进行代入，则存在 Σ_0，使得 Σ 的稳定集中含有 B Received M ，但不含有 A CanProve (B Received M) 。

证明 设通信主体为 A，B，B', TTP，其中 B' 可以由 B 本身充当。当 $\Sigma_0=\{\text{GiveUp}(B,m_2)\}$，P1–P13 中的 A 用 B' 替换时，可得 Σ 的稳定集中含有 B Received M 和 B' CanProve (B Received M), 但不含有 A CanProve(B Received M) 。 证毕

命题 5 表明，主体 B 可以进行重放攻击。经过进一步分析可知，如果允许对 P1–P13 中的主体和消息进行代入，则相当于允许多主体重复运行协议，在这种情形下容易产生重放攻击。在 KM 协议中，如果将 SUBK 中的 $E_{TTP}(K)$ 换成 K，则能够避免上述攻击。

4 与相关工作的比较

周典萃等针对 Kailar 逻辑系统存在的问题进行了改进[7,9]，使新的系统可以分析协议的公平性，并能够对密文进行推理。本文的工作采用了该文的密文理解规则，也克服了 Kailar逻辑不能分析加密消息的缺陷。与文献[9]的区别主要有以下几点：

(1)本文的推理具有非单调性。随着环境和时间的变化，协议的推理结果可能会减少。例如，当主体 P 向主体 Q 发送消息 M，若通信条件正常，可以推出 Q 能收到消息 M，但如果通信通道发生故障，就无法推出 Q 收到消息 M 。

(2)本文的方法可以将环境状况以及主体的意愿加入推理规则中，使得转化后的规则更能反映协议的动态运行过程，减少在推理过程中的人为干预。

(3)针对协议的公平性，文献[9]按照协议的运行步骤建立单调上升(相对于包含关系)的公式集合序列，再判断 EOO 和 EOR 是否属于同一层次的集合来讨论。本文的方法是基于各种不同的环境和主体意愿来讨论相应的稳定集中是否同时含有 EOO 和 EOR 。在特殊情况下，当环境等前提恰好与协议的运行步骤相吻合时，协议分析的过程便退化为文献[9]中的方法(此时的推理也是单调的)。

Li等改进了NCP协议中的缺陷[10]，并利用扩展的Kailar逻辑系统进行了验证。该文的扩展主要体现在对密文和Hash函数的分析。本文的方法也采用了类似的密文理解规则。

Kremer等在分析了已有的离线TTP型协议后提出了新的协议[4]，但没有对该协议进行形式化分析。在一定的理想化假设下，该协议具有可追究性和公平性。但如果将其中一部分理想化假设改为不确定的初始环境，则有可能并生攻击。

5 结束语

本文针对非否认协议运行过程的验证问题，对 Kailar 逻辑系统进行了扩充，增加了表示缺省信息的否定词，建立了新的推理机制和协议验证方法。新方法可以减少协议理想化假设，增强系统的表达能力，推理具有非单调性。文中以一个离线 TTP 型的非否认协议为例，形式化地验证了该协议

在运行一次时具有可追究性，但多次运行时存在攻击。这也说明了不恰当的理想化假设会掩盖着一些可能发生的攻击。

参 考 文 献

[1] Abadi M, Glew N, and Horne B. Certified email with a light on-line trusted third party: design and implementation [C].Proceedings of the Eleventh International World Wide Web Conference.Honolulu, Hawaii, USA. 2002: 387–395.

[2] Abadi M and Blanchet B. Analyzing security protocols with secrecy types and logic programs[J]. *Journal of the ACM*, 2005, 52(1): 102–146.

[3] Gurgens S. On the security of fair non-repudiation protocols[J]. *International Journal of Information Security*, 2005, 4(4): 253–262.

[4] Kremer S and Markowitch O. Optimistic non-repudiable information exchange[C]. 21st Symp.on Information Theory in the Benelux.Wassenaar. The Netherlands. 2000: 139–146.

[5] Kremer S, Markowitch O, and Zhou J. An intensive survey of non-repudiation protocols[J]. *Computer Communications*, 2002, 25(17): 1606–1621.

[6] Kailar R. Accountability in electronic commerce protocols[J]. *IEEE Trans. on Software Eng.* 1996, 22(5): 313–328.

[7] 周典萃，卿斯汉，周展飞. Kailar 逻辑的缺陷. 软件学报. 1999, 10(12): 1238–1245.
Zhou Dian-cui, Qing Si-han, and Zhou Zhan-fei. Limitations of Kailar logic. *Journal of Software*, 1999，10(12): 1238–1245.

[8] 卿斯汉. 安全协议. 北京：清华大学出版社. 2005: 190–210.
Qing Si-han. Security protocols. Beijing: Tsinghua University Press. 2005: 190–210.

[9] 周典萃，卿斯汉，周展飞. 一种分析电子商务协议的新工具. 软件学报. 2001, 12(9): 1318–1328.
Zhou Dian-cui, Qing Si-han, and Zhou Zhan-fei. A new approach for the analys is of electronic commerce protocols. *Journal of Software*, 2001,12(9): 1318–1328.

[10] Li L, Zhang H, and Wang L. An improved non-repudiation protocol and its securty analysis. *Wuhan University Journal of Natural Sciences*, 2004, 9(3): 288–292.

周　勇：　男，1975 年生，讲师，博士生，研究方向为人工智能、信息安全.

朱梧槚：　男，1935 年生，教授，研究方向为数理逻辑.