# 一个多方认证邮件协议的分析与改进

高悦翔[1,2] 彭代渊[1]
(西南交通大学信息安全与国家计算网格实验室 成都 610031)[1]
(四川师范大学计算机科学学院 成都 610068)[2]

**摘 要** 多方认证邮件协议被广泛用于在多方网络环境中传递具有保密性、不可否认性、公平性、无排斥性以及时限性性的电子邮件。指出了一个典型的多方认证邮件协议存在不满足公平性、可追究性以及个别不诚实参与方行为导致整个协议执行失败等安全隐患。基于签密方案,对该协议进行了改进,并利用Kailar逻辑对改进后的协议的安全属性进行了分析。研究结果表明,该协议能够满足保密性、不可否认性及公平性等要求,并具有抗篡改、重放、合谋等攻击的特点。

**关键词** 多方认证邮件协议,可追究性,公平性,签密,Kailar逻辑
**中图法分类号** TP393.08 **文献标识码** A

## Improvement and Formal Analysis of a Fair Multi-party Certified Mail Protocol

GAO Yue-xiang[1,2] PENG Dai-yuan[1]
(Information Security and National Computing Grid Laboratory, Southwest Jiaotong University, Chengdu 610031, China)[1]
(School of Computer Science, Sichuan Normal University, Chengdu 610068, China)[2]

**Abstract** Fair multi-party certified mail protocol is a value service to deliver important data over internet with guaranteed receipt for each successful delivery. Therefore, the protocol needs to be confidential, and meets non-repudiation, fairness, exclusion and timeliness. This paper pointed out potential security concern for a current protocol, and improved the protocol based on signcryption scheme. The analysis by Kailar logic shows that the protocol can achieve the non-repudiation, fairness. Furthermore, it has advantage of resisting the attacks such as distort, replay and conspiracy.

**Keywords** Multi-party certified e-mail protocol, Non-repudiation, Fairness, Signcryption, Kailar logic

## 1 引言

随着电子商务、电子政务的迅速开展,尤其是多方活动涉入后,对多方公平交换协议的需求日益迫切。与普通三方协议相比,多方公平交换协议中参与协议的实体数量的增加导致协议执行步骤增加,执行流程分支复杂化,出现两方协议中不存在的异常,例如参与方之间的相互欺骗、干扰、排斥等。因此,多方公平交换协议还需要满足无排斥性(Exclusion)[1]、时限性[2]等更多的性质。目前在多方公平交换协议方面存在不少研究工作,如文献[3]中首次对多方公平交换协议进行了分类,提出了环状拓扑结构以及矩阵拓扑结构的协议。Kremer等[4]和Markowitch等[5]首先提出了支持在线可信第三方和离线可信第三方的多方不可否认协议,这两个协议都是一对多的关系,即一方发送相同的消息给多个接收者。2003年,Kremer等[6]改进了由他本人提出的协议,为协议增加了机密性。国内学者韩志耕等[7]通过在SVO逻辑中添加时间表达式和时间演算分析的方法分析了Kremer改进后的协议,并指出该协议仍然存在不满足时限性的缺点。同时,文献[8]基于时间段概念、双重群加密技术以及证据链技术提出了一个公平的多方不可否认协议。

作为多方公平交换协议中的一个重要分支,多方认证邮件协议也必须具备可追究性、公平性、时限性、无排斥性等特点,其结构往往也是一对多的结构,这一类协议近年来也得到了许多学者的关注。其中,Ferrer-Gomila等提出了一个使用在线可信第三方的协议[9],但Zhou[10]指出该协议存在一系列的安全问题,同时Zhou等改进了该协议[11],使协议满足了更多的安全属性。为了提高多方认证邮件协议的执行效率,王彩凤等[12]提出了一种签密方案,并在此基础上设计了一个一对多的多方认证邮件协议。该签密方案能有效地减少协议执行中的运算量,其协议的执行效率也较高。虽然目前存在不少对多方认证邮件协议的研究,但这类协议至今仍然没有一个成熟通用的模型。

本文通过分析文献[12]中的协议执行过程发现,该协议具有部分冗余,同时并不完全满足公平性、可追究性,存在多个接收方之间的合谋,使得个别不诚实参与方的行为可能导致整个协议执行失败的潜在安全隐患。在此基础上,本文对该协议进行了扩展改进,弥补了其安全隐患,并利用Kailar逻辑对改进后的协议进行了形式化证明。

到稿日期:2012-01-05 返修日期:2012-04-18
高悦翔(1975—),男,博士生,讲师,主要研究方向为信息安全、密码协议形式化分析,E-mail:gyx415@163.com;彭代渊(1955—),男,教授,博士生导师,主要研究方向为密码学、信息安全、编码理论及扩频序列设计。

## 2 对一个多方认证邮件协议的分析

文献[12]中提出了一种新的、可用于多方认证邮件协议的签密方案，该方案可在一个逻辑步骤内同时实现签名和加密，从而有效地减少协议执行过程中的运算量。在该方案的基础上，提出了一个多方认证邮件协议。该协议的设计目标是能抵御常见的篡改和重放攻击，并能减少对可信第三方的信赖程度，保证邮件在多方环境中的机密性、公平性和不可否认性等属性。以下首先介绍文献[12]中提出的协议。

### 2.1 协议步骤

协议由 Exchange、Recover 和 Abort 3 个子协议构成。其中协议使用的签密方案及所使用的符号含义参见文献[12]。

1)Exchange 子协议

若协议的所有参与方都是诚实的，则只要执行 Exchange 子协议即可完成邮件的传递。

a) $A \Rightarrow R_i: m1$

b) $R_i \rightarrow A: Sig_{R_i}(m1)$，其中 $R_i \in R, i \in \{1, \cdots, n\}$

c) $A \Rightarrow R': C_i = E_{k_i}(m)$，其中 $i \in \{1, \cdots, |R'|\}$

d) $R'' \rightarrow A: Sig_{R_i}(m)$，其中 $R'' \in R'$

2)Recover 子协议

若 $R_i$ 在给 A 发送签名后没有收到 $E_{k_i}(m)$，$R_i$ 可以使用如下的 Recover 子协议请求 TTP 的帮助。

a) $R_i \rightarrow TTP: Sig_{R_i}(m1), m1$，其中 $R_i \in R'$

if *aborted*=true or *recovered*=true then finished

else *recovered*=true

b) $TTP \rightarrow R_i: m$

c) $TTP \rightarrow A: Sig_{TTP}(Sig_{R_i}(m))$

若 $A$ 没有收到 $R_i$ 发送的证据 $Sig_{R_i}(m)$，则 $A$ 执行 Recover 子协议请求 TTP 的帮助。

a) $A \rightarrow TTP: Sig_{R_i}(m1), m1$，其中 $R_i \in R'$

if *aborted*=true or *recovered*=true then finished

else *recovered*=true

b) $TTP \rightarrow R_i: m$

c) $TTP \rightarrow A: Sig_{TTP}(Sig_{R_i}(m))$

3)Abor 子协议

若在 Exchange 子协议中 $A$ 一直无法收到消息 d)而无法获得 $Sig_{R_i}(m1)$，$A$ 可以执行 Abor 子协议向 TTP 申请中止协议。

a) $A \rightarrow TTP: m1$

if *aborted*=true or *recovered*=true then finished

else *aborted*=true

b) $TTP \rightarrow A: Sig_{TTP}(abort)$

c) $TTP \rightarrow R_i: Sig_{TTP}(abort)$

### 2.2 协议存在的缺陷

文献[12]设计的协议由于使用了签密方案，使得协议在安全性不被降低的情况下提高了协议的执行效率。但仔细观察协议的执行可以发现，该协议依然存在如下安全隐患。

1)由于该协议是设计用于多方环境下的邮件传递，协议执行环境中可能存在多个 TTP，而协议执行过程中并未协商使用哪一个 TTP，因此可能存在双方选择不同 TTP 的情况。此时协议参与方在执行 Recover 和 Abort 子协议时可能无法获得相应的有效证据。即使参与方依次遍历检索 TTP，也可能由于额外的时间开销导致邮件的时效性无法满足。

2)如文献[12]所述，$A$ 验证 $R_i$ 收到密文的证据是 $Sig_{R_i}(m1)$，根据签密方案的特性，在发生纠纷时，$A$ 可以提交 $E_{k_i}(m)$，恢复出 $m$。故对 $A$ 而言，$R_i$ 收到密文的证据可以视为 $R_i$ 收到 $m$ 的证据。因此 Exchange 子协议中的 d)消息可以省略，直接利用 $Sig_{R_i}(m1)$作为收方不可否认证据。同时，$A$ 发起的 Recover 子协议的目的是获得 $Sig_{R_i}(m)$，而此时 $A$ 必然已经获得 $Sig_{R_i}(m1)$。如前所述，实质上 $A$ 已经拥有收方不可否认证据，故该子协议为冗余子协议。

3)在协议中，$A$ 没有指明消息的接收群体，任意接收方 $R_i$ 可以和任意未参与协议的 $R_j$ 合谋，在正常执行完协议后将 $m$ 转发给 $R_j$，而此时 $A$ 无法得到 $R_j$ 的收方不可否认证据。即 $R_j$ 可以申明自己参与协议并获得 $m$，而 $A$ 没有收方不可否认证据 $Sig_{R_i}(m1)$，也无法通过发起 Recover 子协议来获取该证据。故在这种情况下，$A$ 是不公平的。

4)在 Recover 子协议中，只要 TTP 判断 *aborted*=true 或者 *recovered*=true 就会终止协议，若一个接收方 $R_i$ 发起 Recover 子协议导致 *recovered*=true，之后其他 $R_j$ 想要发起 Recover 子协议来获得证据都会由于 *recovered*=true 而失败，这对于其他参与方是不公平的。在该子协议中 TTP 发送给 $R_i$ 的消息 $m$ 是明文传输而未作任何处理，故 $R_i$ 只能获得 $m$ 而不能确认 $m$ 是来自于 TTP 或 $A$，因此 $A$ 存在否认发送过 $m$ 的可能，即协议此时不能满足可追究性，同时，攻击者此时也可截获明文传输的 $m$，从而破坏消息的保密性。

5)在 Exchange 子协议中，若 $A$ 已获得部分参与方 $R_i$ 的证据 $Sig_{R_i}(m1)$，$A$ 将会发送 $C_i$ 给 $R_i$，这意味着部分 $R_i$ 已经成功执行协议，获得相应的信息。而对于另一部分 $R_j$($R_j \in (R-R_i)$)，$A$ 由于无法收到 $Sig_{R_j}(m1)$而发起 Abort 子协议。由 Abort 子协议的执行可知，TTP 会向所有的接收方和 $A$ 发送 $Sig_{TTP}(abort)$，即对所有参与方而言，协议放弃执行，个别不诚实参与方的行为导致了整个协议的执行失败，对于前期已经获得 $C_i$ 的参与方而言，这显然是不公平的。

## 3 一个改进的多方认证邮件协议

针对 2.2 节中分析的问题，本文对文献[12]中的协议做出以下改进：

1)在 $A$ 给 $R_i$ 发送的消息中加入 TTP 的身份标识，以防止双方选择不同 TTP 的安全隐患。

2)省略 Exchange 子协议中的消息 d)以及由 $A$ 所发起的 Recover 子协议。

3)在 $A$ 给 $R_i$ 发送的消息中加入接收方集合 $R$ 的标识，用 $R_i$ 的公钥对其加密，以防止参与协议 $R_i$ 和未参与协议的 $R_j$ 合谋获得邮件 $m$。

4)在 Recover 子协议中，删除对 *aborted* 和 *recovered* 的判断，取而代之的是考察发起该子协议的 $R_i$ 的身份，对不同状态采用不同的处理方法，以避免出现其他接收方无法执行 Recover 子协议而导致的不满足公平性的缺陷。同时在 TTP 将 $m$ 发送给 $R_i$ 时用 $R_i$ 的公钥加密，以防止消息的泄露；且要求 TTP 在发送发放不可否认证据的时候对其进行签名。

5)在 Abort 子协议中，单独考察 $R_i$ 的状态，对于已经成功执行完协议或者已经执行了 Recover 子协议的 $R_i$ 不发送 $Sig_{TTP}(abort)$，以避免个别不诚实参与方的行为导致整个协

议的执行失败而破坏公平性的情况。

综上，本文对协议进行如下改进。协议依然采用文献[12]提出的签密方案，同样由 Exchange、Recover 和 Abort 3 个子协议构成。

1)Exchange 子协议

a) $A \Rightarrow R: m1, TTP, PR, h(m1, TTP, PR)$

其中，$PR = P_{R1}(R_1), P_{R2}(R_2), \cdots, P_{Rn}(R_n)$，$h()$为哈希函数。

b) $R_i \rightarrow A: Sig_{Ri}(m1), Sig_{Ri}(TTP, PR)$

其中，$R_i \in R, i \in \{1, \cdots, n\}$

c) $A \Rightarrow R': C_i, h(E_{ki}(m))$

其中，$C_i = E_{ki}(m), i \in \{1, \cdots, |R'|\}$

$A$ 广播 $Sig_A(m1), TTP, PR, h(Sig_A(m1), TTP, PR)$给 $R$，其中用接收方的公钥分别加密各自的身份标识。接收方 $R_i$ 在收到 $A$ 广播的消息后，检查 $m1$ 及 $h(Sig_A(m1), TTP, PR)$是否正确。若正确，对 $m1$ 签名并发送给 $A$。$A$ 检查 $R_i$ 的签名后，对正确回复了 $Sig_{Ri}(m1)$的参与方 $R_i$ 发送 $C_i$。如没有出现异常情况，协议正常结束。

2)Recover 子协议

若部分 $R_i$ 在给 $A$ 发送签名后没有收到 $C_i$，$R_i$ 可以使用如下的 Recover 子协议请求 TTP 的帮助。

a) $R_i \rightarrow TTP: Sig_{Ri}(m1), A, PR, h(Sig_{Ri}(m1), A, PR)$

其中，$R_i \in R$

if ($R_i \in R\_aborted \wedge R_i \notin R\_recovered$) then

b) $TTP \rightarrow R_i: Sig_{TTP}(Aborted, A, R_i)$

else

b) $TTP \rightarrow R_i: Sig_{TTP}(P_{Ri}(m), Sig_A(m1))$

c) $TTP \rightarrow A: Sig_{TTP}(Sig_{Ri}(m1))$

d) $TTP$: appends $R_i$ into $R\_recovered$

$R_i$ 将 $Sig_{Ri}(m1), A, R, h(Sig_{Ri}(m1), A, R)$发送给 TTP，TTP 首先检查 $R_i$ 的签名是否正确，然后检查 $R_i$ 的状态。若 $R_i$ 执行过 Abort 子协议且没有执行过 Recover 子协议，TTP 将 $Aborted\ token = Sig_{TTP}(Aborted, A, R_i, T)$发送给 $R_i$。否则，TTP 利用 $m1$ 恢复出 $m$，将 $m$ 用 $R_i$ 的公钥加密并签名后发送给 $R_i$ 并同时将 $Sig_{Ri}(m1)$签名后发送给 $A$。最后将 $R_i$ 加入 $R\_recovered$ 中，以表示 $R_i$ 曾经执行过 Recover 子协议。

3)Abort 子协议

若在 Exchange 子协议中 $A$ 一直无法收到 $Sig_{Ri}(m1)$，$A$ 可以执行 Abort 子协议，向 TTP 申请中止协议。

a) $A \rightarrow TTP: Sig_A(m1), P_T(R''), H(Sig_A(m1), P_T(R''))$

其中，$R''$表示 $A$ 没有收到回应的接收方集合。

For (all $R_i \in R''$)

if ($R_i \in R''\_recovered$) then

b) $TTP \rightarrow A: Sig_{TTP}(Sig_{Ri}(m1))$

c) $TTP \rightarrow R_i: Sig_{TTP}(P_{Ri}(m), Sig_A(m1))$

else

b) $TTP \rightarrow A: Sig_{TTP}(Aborted, A, R_i)$

c) $TTP \rightarrow R_i: Sig_{TTP}(Aborted, A, R_i)$

d) TTP: appends $R_i$ into $R''\_Aborted$

$A$ 向 TTP 发送 $m1, P_T(R''), H(m1, P_T(R''))$，表示需要中止协议执行。TTP 检查 $m1$ 后，对所有的 $R_i \in R''$分别检查其状态。如果 $R_i \in R''\_recovered$，TTP 向 $A$ 发送收方不可否认证据 $Sig_{Ri}(m1)$，并同时向 $R_i$ 发送发放不可否认证据 $P_{Ri}(m), Sig_A(m1)$。否则，TTP 向 $A$ 和 $R_i$ 发送协议中止 $Aborted\ token = Sig_{TTP}(Aborted, A, R_i)$，表示 $A$ 和 $R_i$ 之间的协议已取消。

## 4 对改进后协议的形式化分析

虽然安全协议的形式化分析方法有很多，但目前还不存在专门用于多方不可否认协议的形式化分析方法。Kailar 提出了一种安全协议的逻辑分析方法[13]，其扩展了信念逻辑的分析范围，用于分析电子商务协议的可追究性。文献[14]针对 Kailar 逻辑存在的缺陷进行了扩展，以下利用该方法对改进后的协议中的各参与方的目标分别进行形式化分析，最后利用各方的分析结果来考察整个协议的执行目标是否达到。限于篇幅，分析中使用的符号及公理系统参见文献[14]。

### 4.1 协议分析的准备

1) 协议分析的准备

① 列出初始拥有集合：

$O_A^0 = \{K_A^{-1}, K_A, K_{Ri}, K_{ttp}, m\}, O_{Ri}^0 = \{K_{Ri}^{-1}, K_A, K_{Ri}, K_{ttp}\}$

② 初始假设集合

a)基本假设

B1 $A$ CanProve $PK(R_i, K_{Ri})$

B2 $R_i$ CanProve $PK(A, K_A)$

B3 $A, R_i$ CanProve $PK(TTP, K_{ttp})$

b)可信假设

T1 $R_i$ CanProve $TTP$ Controls($A$ $Aborted$)

T2 $A$ CanProve $TTP$ Controls($R_i$ $Aborted$)

T3 $A$ Claims $m1 \wedge A$ Claims $E_{ki}(m) \Rightarrow A$ Claims $m$

T4 $TTP$ Claims $Sig_A(m1) \Rightarrow TTP\ ver(Sig_A(m1)) = \text{true}$

T5 $TTP$ Claims $Sig_{Ri}(m1) \Rightarrow$ TTP $ver(Sig_{Ri}(m1)) = \text{true}$

T6 $A, R_i$ CanProve TTP CanProve $X \Rightarrow A, R_i$ CanProve $X$

c)协议理解假设

C1 $R_i$ CanProve $TTP$ Claims($Aborted, A, R_i$) $\Rightarrow R_i$ CanProve $TTP$ Claims($A$ $Aborted$)

C2 $A$ CanProve $TTP$ Claims($Aborted, A, R_i$) $\Rightarrow A$ CanProve $TTP$ Claims($R_i$ $Aborted$)

C3 $TTP$ Claims $P_{Ri}(m) \Rightarrow TTP\ ver(m1) = \text{true}$

其中，$TTP\ ver(m1) = \text{true}$ 意味着 TTP 能够从 $m1$ 中恢复出 $m$。

C4 $TTP\ ver(Sig_A(m1)) = \text{true} \wedge TTP\ ver(m1) = \text{true} \Rightarrow TTP$ CanProve $A$ Claims $m$

C5 $R_i$ Claims $m1 \Rightarrow R_i$ Has $m$

C6 $TTP ver(Sig_{Ri}(m1)) = \text{true} \Rightarrow R_i$ Claims $m1$

③ 列举 EOO 和 EOR

$EOO = Sig_A(m1) \wedge Sig_A(E_{ki}(m)) \vee$

$EOO = Sig_{TTP}(Aborted, A, R_i) \vee$

$EOO = Sig_{TTP}(P_{Ri}(m), Sig_A(m1))$

$EOR = Sig_{Ri}(m1) \vee EOR = Sig_{TTP}(Aborted, A, R_i) \vee$

$EOR = Sig_{TTP}(Sig_{Ri}(m1))$

### 4.2 不可否认性分析

1) 列举不可否认目标

$R_i$ CanProve($A$ Claims $m$) $\vee R_i$ CanProve ($A$ $Aborted$)

$A$ CanProve($R_i$ Has $m$) $\vee$ $A$ CanProve($R_i$ *Aborted*)

2)分析 EOO 与 EOR 的设计是否符合可追究性要求

① 假设 $EOO \in O_{Ri}$，即：

$(Sig_A(m1) \wedge Sig_A(E_{ki}(m))) \in O_{Ri} \vee Sig_{TTP}(Aborted, A, R_i) \in O_{Ri} \vee Sig_{TTP}(P_{Ri}(m), Sig_A(m1)) \in O_{Ri}$

a) 当 $Sig_{TTP}(Aborted, A, R_i) \in O_{Ri}$ 时，

$R_i$ Has $Sig_{TTP}(Aborted, A, R_i)$ (1)

由签名公理、式(1)、B3 可得

$R_i$ CanProve *TTP* Claims($Aborted, A, R_i$) (2)

由式(2)、C1 可得

$R_i$ CanProve *TTP* Claims ($A$ *Aborted*) (3)

由管辖公理、式(3)、T1 可得

$R_i$ CanProve($A$ *Aborted*) (4)

由式(4)可知此时满足不可否认目标。

b) 当$(Sig_A(m1) \wedge Sig_A(E_{ki}(m))) \in O_{Ri}$时，

$R_i$ Has $Sig_A(m1) \wedge R_i$ Has $Sig_A(E_{ki}(m))$ (5)

由签名公理、B2、式(5)可得

$R_i$ CanProve $A$ Claims $m1$ (6)

由签名公理、B2、式(5)可得

$R_i$ CanProve $A$ Claims $E_{ki}(m)$ (7)

由连接公理，式(6)、式(7)、T3 可得

$R_i$ CanProve($A$ Claims $m$) (8)

由式(8)可知此时满足不可否认目标。

c) 当 $Sig_{TTP}(P_{Ri}(m), Sig_A(m1)) \in O_{Ri}$ 时为

$R_i$ Has $Sig_{TTP}(P_{Ri}(m), Sig_A(m1))$ (9)

由式(9)、B3 可得

$R_i$ CanProve *TTP* Claims $P_{Ri}(m), Sig_A(m1)$ (10)

由式(10)、T4、T6、C3、C4 可得

$R_i$ CanProve $A$ Claims $m$ (11)

由式(11)可知此时满足不可否认目标。

② 假设 $EOR \in O_A$，即：

$Sig_{Ri}(m1) \in O_A \vee Sig_{TTP}(Aborted, A, R_i) \in O_A \vee Sig_{TTP}(Sig_{Ri}(m1)) \in O_A$

a) 当 $Sig_{TTP}(Aborted, A, R_i) \in O_A$ 时，

$A$ Has $Sig_{TTP}(Aborted, A, R_i)$ (12)

由签名公理、式(1)、B3 可得

$A$ CanProve *TTP* Claims($Aborted, A, R_i$) (13)

由式(13)、C2 可得

$A$ CanProve *TTP* Claims($R_i$ *Aborted*) (14)

由管辖公理、式(14)、T2 可得

$A$ CanProve($R_i$ *Aborted*) (15)

由式(15)可知此时满足不可否认目标。

b) 当 $Sig_{Ri}(m1) \in O_A$ 时，

$A$ Has $Sig_{Ri}(m1)$ (16)

由签名公理、式(1)、B1 可得

$A$ CanProve($R_i$ Claims $m1$) (17)

由式(17)、C5 可得

$A$ CanProve($R_i$ Has $m$) (18)

由式(18)可知此时满足不可否认目标。

c) 当 $Sig_{TTP}(Sig_{Ri}(m1)) \in O_A$ 时，

$A$ Has $Sig_{TTP}(Sig_{Ri}(m1))$ (19)

由签名公理、式(19)、B3 可得

$A$ CanProve(*TTP* Claims $Sig_{Ri}(m1)$) (20)

由式(20)、T5、C6 可得

$A$ CanProve($R_i$ Claims $m1$) (21)

由式(21)、C5 可得

$A$ CanProve($R_i$ Has $m$) (22)

由式(22)可知此时满足不可否认目标。

综上所述，EOO 和 EOR 的设计能够满足可追究性目标。以下考察协议是否能够获得 EOO 和 EOR。

3) 分析协议是否达到可追究性目标

如前所述，EOO 为：

$Sig_A(m1) \wedge Sig_A(E_{ki}(m)) \vee Sig_{TTP}(Aborted, A, R_i) \vee Sig_{TTP}(P_{Ri}(m), Sig_A(m1))$

EOR 为：

$Sig_{Ri}(m1) \vee Sig_{TTP}(Aborted, A, R_i) \vee Sig_{TTP}(Sig_{Ri}(m1))$

对应于协议的不同执行状况，对 $R_i$ 而言有：$Sig_A(m1) \in O_{Ri}^1$，其中 $O_{Ri}^1$ 表示 $R_i$ 在 Exchange 子协议中第一步所拥有的知识集合。$Sig_A(E_{ki}(m)) \in O_{Ri}^2$，故 $Sig_A(m1) \wedge Sig_A(E_{ki}(m)) \in O_{Ri}$，即 $R_i$ 总能获得 $Sig_A(m1) \wedge Sig_A(E_{ki}(m))$。

在 Abort 子协议的执行过程中，$R_i$ 获得 $Sig_{TTP}(Aborted, A, R_i)$ 或 $Sig_{TTP}(P_{Ri}(m), Sig_A(m1))$。同时，在 Recover 子协议的执行过程中，$R_i$ 获得 $Sig_{TTP}(Aborted, A, R_i)$ 或 $Sig_{TTP}(P_{Ri}(m), Sig_A(m1))$。总之，在各子协议中，$R_i$ 均能获得 EOO。同理，$A$ 在 3 个子协议的执行中，也能获得 EOR。可见，通过协议的不同执行流程，参与方总能获得各自的不可否认证据，因此协议能够达到可追究性目标。

### 4.3 公平性分析

协议达到公平性目标等价于下述 3 个命题成立。

**命题 1** $Sig_{TTP}(Aborted, A, R_i) \in O_{R_i}$ 当且仅当 $Sig_{TTP}(Aborted, A, R_i) \in O_A$。

证明：由于 $Sig_{TTP}(Aborted, A, R_i)$ 在 Recover 子协议或 Abort 子协议执行中，由 TTP 同时向 $A$ 和 $R_i$ 发出，故在弹性信道的条件下，$A$ 和 $R_i$ 总可以接收到 EOO 和 EOR。此时，命题 1 成立。

**命题 2** $Sig_{TTP}(P_{Ri}(m), Sig_A(m1)) \in O_{R_i}$ 当且仅当 $Sig_{TTP}(Sig_{Ri}(m1)) \in O_A$。

证明同命题 1。

**命题 3** $Sig_A(m1) \wedge Sig_A(E_{ki}(m)) \in O_{R_i}$ 当且仅当 $Sig_{Ri}(m1) \in O_A$。

证明：由协议执行过程可知，如果协议顺利执行 Exchange 子协议而结束，$A$ 发送 $Sig_A(E_{ki}(m))$ 的前提条件是 $A$ 能够收到部分合法的 $Sig_{Ri}(m1)$。而 $R_i$ 发送 $Sig_{Ri}(m1)$ 的前提条件为收到有效的 $Sig_A(m1)$。那么存在：

$R_i$ receives $Sig_A(m1) \wedge Sig_A(E_{ki}(m)) \Rightarrow A$ receives $Sig_{Ri}(m1)$ (23)

$A$ receives $Sig_{Ri}(m1) \Rightarrow R_i$ receives $Sig_A(m1) \wedge Sig_A(E_{ki}(m))$ (24)

由式(23)、式(24)知，命题 3 成立。

若协议在执行 Exchange 子协议时出现异常，需要执行 Abort 子协议或 Recover 子协议，此时参与方无法正常获得相关证据，协议的公平性目标等同于命题 1 和命题 2，其证明如

(下转第 97 页)

[13] Kiyomoto, Andrew S. A Security Protocol Compiler Generating C Source Codes[C]// International Conference on Information Security and Assurance. Busan, Korea, April 2008:20-25

[14] Zhang Huan-guo, Wang Zhang-yi. Cryptography Introduction [M]. Wuhan: Wuhan University Press, 2009:23-25

[15] Sun Sheng-he, Lu Zhe-ming, Niu Xia-mu. Digital Watermarking Technology and Application[M]. Beijing: Science press, 2004: 36-37

[16] Wang C X. A security architecture for survivability mechanisms [D]. University of Virginia, Department of Computer Science, 2000

[17] Collberg C, Thomborson C. Watermarking, Tamper-Proofing, and Obfuscation-Tools for Software Protection[J]. IEEE Transactions on Software Engineering, 2002, 28(6): 735-746

[18] Zhu W, Thomborson C D, Wang Fei-yue. Applications of Homomorphic Functions to Software Obfuscation[C]// WISI. 2006: 152-153

[19] Han Xiao-xi, Wang Gui-lin. An Attack to Multisignature Schemes Based on Discrete Logarithm[J]. Chinese Journal of Computers, 2004, 8: 1147-1152

[20] Horne B, Matheson L, Sheehan C, et al. Dynamic self-checking techniques for improved tamper resistance[C]// Proc. 1st ACM Workshop on Digital Rights Management (DRM2001). Springer, 2002: 141-159

(上接第 61 页)

前所述。

综合命题 1—命题 3 的分析可知，协议能够满足公平性。

## 5 与改进前协议的比较

本文对文献[12]提出的协议进行了改进，方案主要是在原协议的交换轮次中增加了部分交换项，改进了原协议执行中的部分判断条件，同时删除了原协议中不影响协议安全属性的冗余消息及冗余子协议。通过 Kailar 逻辑证明了改进后的协议弥补了原协议存在的安全隐患。以下通过对协议的整体执行流程以及 3 个子协议的流程进行比较，分析改进后的协议与原协议的执行效率。

在协议的整体执行流程方面，改进后的协议并没有增加原协议的执行步骤，同时删除了由 $A$ 发起的 Recover 子协议，故在协议执行流程方面，改进后的协议更为精简。在 3 个子协议中，改进后的协议增加的数据项包括：

1)在 Exchange 子协议的 b)中增加了一个签名运算，在 c)中增加了一个 Hash 函数运算；

2)在 Recover 子协议中的 a)中增加了 PR 等数据项，说明协议发起者是合法参与方，c)中增加了一个加密操作以满足邮件的保密性；

3)在 Abort 子协议中的 a)中增加了 PR 和 Hash 等数据项，说明协议发起者是合法参与方，同时在 b)和 c)中若 $R_i \in R''\_recovered$，则对签名内容进行更改。

以上增加的数据项对于协议安全属性的保证是必须的，且并未对原协议的执行效率和复杂度造成影响。同时由于删除了原协议的 Exchange 子协议中的 d)消息和由 $A$ 发起的 Recover 子协议，因此改进后的协议并没有降低原协议执行效率。

**结束语** 本文分析了文献[12]提出的一个基于离线第三方的多方认证邮件协议，指出了该协议存在的安全隐患，并对原协议进行了改进。通过与文献[12]的比较，可以看出本文对协议所做的改进并没有降低原协议的执行效率，但消除了原协议存在的安全隐患。然后利用 Kailar 逻辑对改进后的协议进行了形式化分析。研究结果表明，在保证执行效率的基础上，改进后的协议具有多方认证邮件协议所必需的保密性、不可否认性及公平性的特性，而且具有抗篡改、重放、合谋等攻击的特点。

## 参 考 文 献

[1] González-Deleito N, Markowitch O. Exclusions and related trust relationships in multi-party fair exchange protocols [J]. Electronic Commerce Research and Applications, 2007, 6(3): 343-357

[2] Kremer S, Markowitch O, Zhou Jian-ying. An intensive survey of fair non-repudiation protocols [J]. Computer communications, 2002, 25(17): 1606-1621

[3] Franklin M, Tsudik G. Secure Group Barter: Multi-Party Fair Exchange With Semi-Trusted Neutral Parties [A]// Process of Financial Cryptography'98 [C]. LNCS1465. Anguilla, Springer, 1998: 90-102

[4] Kremer S, Markowitch O. A multi-party non-repudiation protocol [C]// The 15th International Conference on Information Security. Beijing, China, IFIP World Computer Congress, 2000: 271-280

[5] Markowitch O, Kremer S. A multi-party optimistic non-repudiation protocol [C]// Information Security and Cryptology ICISC 2000: Third International Conference. Seoul, Korea: Spinger-Verlag, 2001: 109-122

[6] Kremer S, Markowitch O. Fair Multi-Party Non-Repudiation protocols [J]. International Journal of Information Security, 2003, 1(4): 223-235

[7] 韩志耕，罗军舟. 一个公平的多方不可否认协议[J]. 计算机学报，2008，31(10)：1705-1715

[8] 韩志耕，罗军舟. 多方不可否认协议时限性分析与改进 [J]. 电子学报，2009，37(2)：377-381

[9] Ferrer-Gomila J L, PayerasCapell M, Huguetrotger L. A realistic protocol for multi-party certified electronic mail [C]// Proceedings of Information Security Conference. Sao Paulo: Springer, 2002: 210-219

[10] Zhou J. On the security of a multi-party certified e-mail protocol [C]// Proceedings of 2004 International Conference on Information and Communications Security. Malaga: Springer, 2004: 40-52

[11] Zhou J, Onieva J, Lopez J. Optimized multi-party certified email protocols [J]. Information Management and Computer Security, 2005, 13(5): 350-366

[12] 王彩芬，贾爱库，刘军龙，等. 基于签密的多方认证邮件协议[J]. 电子学报，2005，33(11)：2070-2073

[13] Kailar R. Accountability in electronic commerce protocols [J]. IEEE Trans. on Software Engineering, 1996, 22(5): 313-328

[14] 卿斯汉. 一种电子商务协议形式化分析方法[J]. 软件学报，2005，16(10)：1757-1765