维普资讯 http://www.cqvip.com

公平非否认协议的有限状态分析*)

董荣胜　陈大伟　郭云川　古天龙
(桂林电子工业学院计算机系　桂林 541004)

摘　要　本文针对公平非否认协议给出了一种基于有限状态自动机的分析模型，并使用 SPIN 模型检测工具，对 Zhou-Gollmann 非否认协议进行了分析，结果发现该协议不满足公平性和机密性，为此对该协议进行了改进。

关键词　模型检测，SPIN，非否认性，公平性，机密性

Finite-state Analysis of a Fair Non-repudiation Protocol

DONG Rong-Sheng　CHEN Da-Wei　GUO Yun-Chuan　GU Tian-Long
(Department of Computer, Guilin University of Electronic Technology, Guilin 541004)

Abstract　A model based on finite state automaton is proposed for a fair non-repudiation protocol. The Zhou-Gollmann fair non-repudiation protocol is analyzed with using model checker SPIN. The results show that the protocol isn't satisfied with fairness and confidentiality. Finally, some improvements are given for this protocol.

Keywords　Model checking, SPIN, Nonrepudiation, Fairness, Confidentiality

1　引言

随着 Internet 的迅速发展，电子交易已经变得越来越普遍。如交易的是合同、标书、订单和支票等重要数据的时候，首先考虑安全性问题。然而在保证信息安全的同时，如何防止交易双方否认发送或者接收了消息，或者说要证明一个指定的行为发生过，那么就需要引入非否认服务。非否认协议保护了正在交易的实体，不能否认任何已经发生的事件或行为，在协议执行中将产生、收集、维护证据，以解决争端。在保证非否认的同时，非否认协议还必须保证在执行的任何阶段，参与协议的任一方都不占有优势，也即需要满足公平性。

密码协议在设计出来之后，需要从多方面对其各种性质进行验证。形式化验证以其严谨、简洁的特点而成为验证协议的重要方法，但主要集中在认证和密钥交换协议上。非否认协议不同于认证和密钥交换协议，交易实体双方互不信任，而认证和密钥交换协议仅考虑实体以外存在入侵者。这意味着，这种协议需要满足不同的性质和需求，且不能完全采用认证和密钥交换协议的建模和分析方法。本文基于模型检测技术，给出了一种针对该类协议的形式化模型，将协议中的主体分为诚实主体和不诚实主体，分别表示为有限状态机形式。建立了 Zhou-Gollmann 公平非否认协议[1]的有限状态机模型，将该模型转换为 SPIN 模型检测工具的描述语言(Promela)，对其进行分析，结果发现该协议不具备公平性和机密性。最后对该协议进行了改进，使其满足公平性和机密性。

2　公平非否认协议

从目前已有的公平非否认协议来看，按协议中交换证据的方式不同，可以分为两大类：一类是无需可信第三方(Trusted Third Party, TTP)的非否认协议，也称为概率协议(probabilistic protocols)，它是采用逐步释放消息和逐步请求的方法，协议参与方同步将所要交换的信息透漏给对方。由于逐步交换，获得全部证据的概率每轮都在增长，但这类协议的参与方还是存在一个小的概率欺骗另一方而处于优势地位。另外这种协议要求通信双方严格的时间同步和相同的计算能力，且交互次数相当频繁，现有的 Internet 是较难实现的。第二类是使用可信第三方的非否认协议，它允许参与方异步的交换信息，TTP 的参与可以作为信息的传递、仲裁纠纷，但需要完成较大的计算量，这容易形成协议通信或计算的瓶颈。因此目前的研究重点是如何减少协议对可信第三方的依赖程度。

基于可信第三方，Zhou 和 Gollmann 提出了一种带有 Online TTP 的公平非否认协议，简称 ZG 协议，保证了公平性，并使 TTP 的使用最小化。

2.1　ZG 协议

ZG 协议的思想是将要交换的关键数据(消息)划分为两部分，委托(commitment)C 和密钥 K。C 在 A 和 B 间交换，密钥由第三方寄存。A 和 B 必须从 TTP 中取得确认密钥，作为非否认证据的部分。协议描述中需要符号如下：

A：非否认交换的发起者；B：非否认交换的接收者；TTP：对公众提供网络服务访问的在线可信第三方；$sS_A(X)$：主体 A 用私钥对 X 的数字签名，不可伪造，不可否认；(M：由 A 发送给 B 的消息(合同、订单等)；$C = eK(M)$：消息 M 的委托(Commitment)，用 K 对 M 加密所得；K：用来对 M 加密的对称密钥，由 A 给出；$L = H(M, K)$：连接一次协议运行所有消息的唯一标识符，M 和 K 的 hash 而得；$NRO = sS_A(f_{NRO}, B, L, C)$：消息 M 的非否认发起证据；$NRR = sS_B(f_{NRR}, A, L, C)$：消息 M 的非否认接收证据；$sub\_K = sS_A(f_{SUB}, B, L, K)$：密钥 K 的提交证据；$con\_K = s_{TTP}(f_{CON}, A, B, L, K)$：TTP 发出的密钥 K 的确认证据；$f_{NRO}$，$f_{NRR}$，$f_{SUB}$，$f_{CON}$：分别是表示各个证据的标记。

协议如下：

1. $A \rightarrow B: f_{NRO}, B, L, C, NRO$
2. $B \rightarrow A: f_{NRR}, A, L, NRR$
3. $A \rightarrow TTP: f_{SUB}, B, L, K, sub\_K$
4. $B \leftrightarrow TTP: f_{CON}, A, B, L, K, con\_K$

*)本文得到广西自然科学基金项目(编号：桂科自 0229051)资助。

维普资讯 http://www.cqvip.com

5. $A \leftrightarrow TTP: f_{ON}, A, B, L, K, con\_K$

协议中发起者和接收者的身份都被包括在签名的消息中，来保护他们是合法被使用，并且不可被伪造。首先，A 使用密钥 K 将要发送的重要消息 M 加密获得 C，发送消息 1 来初始协议；B 接收消息 1 后，保存 A 的数字签名 NRO 作为发起非否认证据，日后出现争议时使用，同时发送消息 2；A 收到消息 2 后，保存 B 的数字签名 NRR 作为接收非否认证据，日后出现争议时使用，然后消息 3 中，A 发送密钥 K 的拷贝版本给 TTP；TTP 在一次（L 来标识）协议会话中仅仅接收来自一个实体的提交，在 sub_K 中检测是否 A 的签名有效，确定无误之后，TTP 生成确认证据 con_K，连同标记 f_{ON}，发起和接收者身份 A 和 B，标识协议唯一会话的 L 和密钥 K，一同放到 TTP 的公共目录中，对外只读可访问；在消息 4,5 中，“↔”是“FTP”操作，A 和 B 在 TTP 的公共目录中下载消息。A 将获得这次会话的消息确认证据 con_K，证明会话密钥 K 已被 B 接收；而 B 将获得密钥 K，对 C 解密获得重要消息 M，而 con_K 连同之前的 NRO 构成完整的非否认证据。

2.2 争议解决

如果协议运行在消息 3 以前出现问题，协议终止，任何一方都不能获得完全的证据，争议不会产生。而在消息 4,5 中，双方都会努力在 TTP 上获得证据，而不想在之后的争议中失败。在 ZG 协议中，假设在 TTP 和每个实体之间，协议要很好地运行，需要一个弹性的通信通道（resilient channel），即在弹性通道中所有发送的消息，在耽搁有限的时间后，最终消息一定会发送成功。这就保证了 A 和 B 最后一定能获得确认证据 con_K。

在争议中，如果 A 声明已经成功地发送了消息给 B，而 B 否认接收了消息，裁判要求她提供这个消息和这个消息的接收非否认证据。非否认证据由 B 提供的接收证据 NRR 和 TTP 提供的确认证据 con_K 两部分组成。如果所有 A 提供的消息都是正确的，裁判就宣布 A 是正确的。同样如果 B 声明接收了 A 的消息，而 A 否认曾经发送的消息，那么 B 必须出示非否认证据 NRO 和 con_K 以及消息 M，所有消息都正确，就裁定 B 获胜。

3 ZG 协议的形式化建模

Zhou 等[2]首先使用 SVO 逻辑对 ZG 协议进行了验证，推理得出证据的有效性。之后，Schneider[3] 使用进程演算 CSP 证明了非否认协议的正确性。Sigrid 等[4]提出一种新的方法，使用 APA（asynchronous product automata）和 SHVT（simple homomorphism verification tool）对 ZG 协议进行了验证。模型检测对认证和密钥交换协议的分析已经取得了很好的效果，但在非否认协议上的应用却很少，本文的分析方法是首先建立协议主体的有限状态自动机模型，然后应用模型检测器 SPIN 对其分析。

认证协议和非否认协议有根本的不同，通常处理认证协议最主要的是引入入侵者模型，而执行协议的主体都被认为是严格按照协议来执行，也即，成功的建模入侵者是找出协议缺陷的关键。而在公平非否认协议中，在验证公平性的时候，由于是考虑两个主体是否公平，它们的利益是相互冲突的，因此并不需要对入侵者建模。但要考虑 A 和 B，两个协议参与主体之间可能存在欺骗行为。诚实主体通常按给定的序列来执行，如果执行序列被改变就可能导致协议不满足公平性，这就是本文提出的对协议进行规约和验证的方法。文[5]提出了一种基于通信有限状态机结合 SMV 模型检测工具，对电子商务原子性验证的一个形式化模型。本文在其基础上，给出了对非否认协议验证的有限状态机形式化模型。

3.1 协议主体的建模

协议的正常执行通常被看作是消息交换的线性队列，在都是诚实主体情况下，以预先定义的序列来执行。而本文采用的方法是，对主体 A 和 B 不给任何预定义的序列，主体可以选择任意不同的执行序列，以找到可能的攻击来欺骗诚实的主体。将严格按照协议规定的序列执行的主体称为诚实主体，而不按照协议规定，任意发送消息，任意终止协议运行，或改变执行顺序的主体称为不诚实主体或叫恶意主体。A 和 B 就分为诚实和恶意主体来建模，而 TTP（可信第三方）被假设一定是诚实的，否则协议一定不公平。

通过以上分析，协议主体可定义为通信有限状态自动机，消息的不同执行序列就是各个主体状态转换的异步结合。

定义 1 一个通信有限状态自动机（简称 CFSM）是一个五元组：$C = (S, N, M, \delta, S_0)$，其中：(1) S 是有限状态集；(2) $N \in names$，$names$ 是所有协议主体（principal）名称集合；(3) M 是消息集，即 $M = \{!\ m$ 或者 $?\ m \mid m \in M\}$，公式中“$!\ m$”表示发送消息 m，而“$?\ m$”表示接收消息 m；(4) $S_0 \in S$，是初始状态；(5) δ 是状态转移函数，$\delta: S \times M \to S$。

协议中的每个诚实主体都可定义成这样的一个有限自动机，而不诚实主体需要定义成非确定的有限自动机。

定义 2 具有 ε 动作的非确定通信有限状态自动机是一个五元组：$C = (S, N, M, \delta, S_0)$，其中 S, N, M, S_0 的含义与定义 1 相同，而转移函数 $\delta: S \times (M \cup \{\in\}) \to >2^S$（$S$ 的一切子集集合）。

一个恶意主体在不接收正确的 M 中的消息（输入为 ε）时也能做状态转换，图 1 给出 A 和 B 的有限状态机模型。

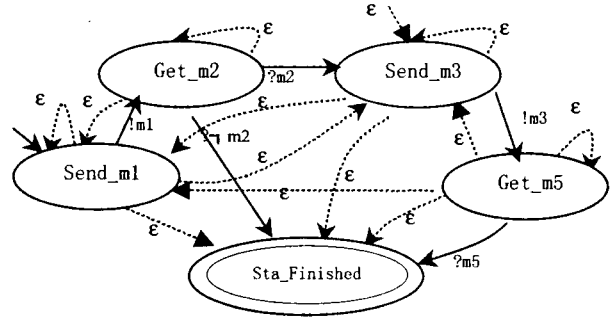

图 1 主体 A 的有限自动机

主体 A 分为五种状态：$Send\_m1$，Get_m2，$Send\_m3$，Get_m5，$Sta\_Finished$。图中“$!\ m1$”表示发送消息 1，“$?\ m2$”表示接收消息 2，“ε”表示主体的不确定行为，也即是恶意主体的执行路径。

对于诚实主体 A，按图中的实线顺序执行，初始状态是 $Send\_m1$，首先发送消息 1 给 B，在状态 Get_m2 时，如收到 B 的消息 2，就转到 $Send\_m3$ 状态，如未收到消息 2，直接结束协议运行，双方都没有获得足够证据，不会出现争议。这里在 A 和 B 之间的通信通道是不可靠信道（unreliable channel），消息可能丢失。所以消息 1 和 2 可能发送不成功，A 可在 Get_m2 状态直接终止协议。而在 A 和 TTP 之间的信道是弹性信道（resilient channel），消息最终一定会到达，这样在状

维普资讯 http://www.cqvip.com

态 *Send_m3*，*A* 发送消息 3 给 *TTP* 和状态 *Get_m5*，从 *TTP* 取得消息 5，保证了消息最后一定会达到目的地，主体 *A* 最后获得全部证据。

而当主体 *A* 是恶意主体时，就按照实线和虚线结合的所有可能序列和状态转换来执行。图 2 中为了简化去掉了部分无用的状态转换，如 *Get_m2* 到 *Get_m5*，初始状态是 *Get_m2* 或 *Get_m5* 等，这些状态转换对协议的执行结果不可能产生影响。恶意主体可在任何状态下终止协议，初始状态也不一定是 *Send_m1*，执行到后面的状态时也可跳转到协议的初始状态，总之恶意主体的执行是不确定性的。

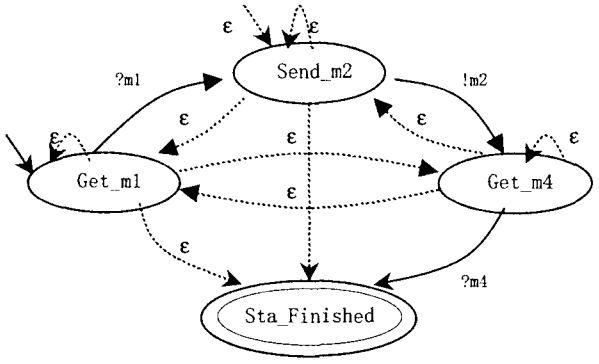

图 2　主体 *B* 的有限自动机

主体 *B* 的建模同 *A* 的相似，也分为诚实主体和恶意主体。但需注意的是，恶意主体目的是欺骗对方的诚实主体，而使自己在协议中能获得利益。如果双方都为恶意主体，协议执行就没有意义，所以假设一次协议会话中最多只能有一个主体是不诚实的。

由于可信第三方 TTP 假设一定是诚实的，因此其状态模型相对简单，在此就不给出，图 3 给出整个协议的通信模型。

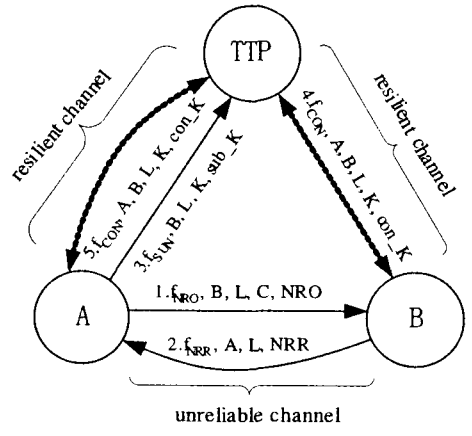

图 3　ZG 协议的通信模型

由图可知，消息 1，2 是在不可靠信道传送，消息 3，4，5 在弹性通道传送。意味着，消息 1，2 在不可靠信道中，可能随时丢失或被无限期延迟，对此通道的建模是使其进入的消息随机的丢失。而弹性通道中，消息最终都会发送到目的地，所以就建模成正常的通道，但对于要求时限性时就需要引入一个整型变量 count_time 来计数，在小于某个值时，消息一定要发送。

3.2　协议性质的 LTL 描述

公平非否认协议属于电子商务协议的一种，不但应当具备安全认证协议的性质，还需要具备非否认性、公平性、机密性等安全性质[6]。而协议被设计出来，首先要验证的是其有效性，下面给出各个性质的定义和线性时态逻辑（LTL）描述（本文对认证协议的相关性质不做介绍）：

定义 3（非否认性或不可抵赖性（Nonrepudiation））　要求发送方对已发出的信息不可抵赖，接收方对已接收的信息也无法否认。

由定义 3 可知，非否认包括两层意思：发起非否认（Repudiation of Origin）和接收非否认（Repudiation of Receipt）。由于 ZG 中提供的非否认证据 *NRR*、*NRO*、*con_K* 都是经过主体签名的，一般认为一个主体不能否认经其私有密钥签名的数据，因此不用再对非否认性证明。

定义 4（有效性或可行性（Viability））　如果双方都遵守协议且不中断，则协议确保交换成功，即 *B* 可收到 *A* 发送的消息 *M* 和证据，而 *A* 可得到 *B* 合法的签名收据。

有效性的验证实际上就是，*A* 和 *B* 都是诚实主体的情况下，协议正常运行并且结束，双方都能获得自己所需要的信息，LTL 表示如下：

P1：<>((*A_NRR*&&*A_con_K*)&&(*B_NRO*&&*B_con_K*&&*B_M*))

定义 5（公平性（Fairness））　是指在协议执行完成之后，收发方要么彼此都获得自己想要得到的数据，要么均收不到任何有价值的信息，而且在协议进行中，即使某方有意中断协议，也不会比对方处于有利地位。

公平性是非否认协议最重要的性质，如果至少一个实体是诚实的，那么或者两个实体收到所期望的非否认证据，或者他们中没有一个能获得。对 *A* 的公平性，如果 *B* 是不诚实的，*B* 收到了发起非否认证据，那么 *A* 最终一定能收到她的接收非否认证据，表示如下：

P2：[]((*B_NRO*&&*B_con_K*) → <> (*A_NRR*&&*A_con_K*))

如果 *A* 是不诚实的，*A* 收到了接收非否认证据，那么 *B* 最终一定能收到他的发起非否认证据，表示如下：

P3：[]((*A_NRR*&&*A_con_K*) → <> .(*B_NRO*&&*B_con_K*))

定义 6（机密性（Confidentiality））　保证需要保密的协议消息内容，在传送过程中不被非法窃取，即使入侵者截获了消息，依然无法读取消息中的关键信息。

对于 ZG 协议，就是即使入侵者在信道上截获消息，他依旧无法解读消息 M，也即消息只有 A 和 B 知道，连 TTP 也不知道。显然，机密性是针对协议双方之外的任何主体而保密的，所以在验证机密性时需要引入入侵者模型。本文建立了一个被动攻击者模型，它只负责在信道中截取和转发消息，以增长知识，自己本身不作为主体参与协议运行，协议的发起和接收者可以是诚实主体或恶意主体。机密性的 LTL 表示如下：

P4：[]!（*know_C*&&*know_K*）

4　ZG 协议的 SPIN 分析与改进

SPIN[7,8]（Simple Promela INterpreter）适合于并行系统，尤其是协议一致性的辅助分析验证工具。它以 PROMELA（PROcess Meta LAnguage）为输入语言，可以对网络协议设计中规格的逻辑一致性进行检验。检测一个有限状态系统是否满足线性时态逻辑（LTL）公式及其它一些性质。

维普资讯 http://www.cqvip.com

将上述的模型和 LTL 描述的性质，转换为相应的 Promela 语言，每个主体定义不同的进程，各个进程交叉异步执行，穷举所有可能的状态空间，如遇到不满足性质的状态转换，SPIN 输出错误，并且给出导致性质不满足的执行序列。结合不同的主体模型，对 P1～P4 性质分别进行了验证，结果发现 ZG 协议满足有效性，但不满足公平性和机密性。

(1)公平性不满足　当 A 是不诚实主体，B 是诚实主体，A 可能使得自己获得足够的证据，而 B 没有获得，而使得在以后的争议中获得优势。执行序列如下：

1. $A \rightarrow TTP$: f_{SUB}, B, L, K, sub_K
2. $A \leftrightarrow TTP$: f_{CON}, A, B, L, K, con_K
3. $A \rightarrow B$: f_{NRO}, B, L, C, NRO
4. $B \rightarrow A$: f_{NRR}, A, L, NRR
5. *Protocol aborted*

恶意主体 A 开始执行消息 3，而不是消息 1。TTP 接收之后，并不具备辨别诚实和恶意主体的能力，也不知道 1，2 消息没有执行过，检测消息 3 正确后，就公布密钥 K。而 A 立刻执行消息 5，从 TTP 获得确认证据 con_K。这时 B 并不知道已经执行了上述操作，就不会从 TTP 读取密钥 K 和证据。TTP 在一定的时间之后会取消发布的信息，A 仍然使用前面消息 M，唯一标识符 L，给 B 发送消息 1。B 接收后回复消息 2，A 收到消息 2 取得 NRR 后终止协议，B 执行消息 4 发现 TTP 并没有发布密钥 K。最后 A 获得了 NRR 和 con_k，而 B 只拥有 NRO，协议因此不公平。

经分析导致此攻击的原因是，TTP 接收消息 3 后并不能判断消息 1，2 是否执行过，因此不知道 A 是恶意的主体，解决的方法是在消息 3 中加入 NRR，由于 NRR 是 B 签名，无法伪造，要想获得必定消息 1，2 已被执行过。

(2)机密性不满足　被动攻击者监控所有的信道，只从信道中监听消息，而不删除或篡改。当 A 和 B 都是诚实主体时候，协议正常执行，攻击者就可最终获得重要的消息 M，如下：

1. $A \rightarrow B$: f_{NRO}, B, L, C, NRO
   1: $\rightarrow I$: f_{NRO}, B, L, C, NRO
2. $B \rightarrow A$: f_{NRR}, A, L, NRR
3. $A \rightarrow TTP$: f_{SUB}, B, L, K, sub_K
   3: $\rightarrow I$: f_{SUB}, B, L, K, sub_K

攻击者只需监控 A-B 信道和 A-TTP 信道就可获取消息 1 和 3，而取得 C 和密钥 K，因为 K 是对称密钥，C 是由 K 加密，同样解密也是使用 K，这样攻击者就得到了机密消息 $M = dK(C)$。即使攻击者不能截获消息 3，也可在 TTP 上读取密钥 K，因为 TTP 是对外公开提供 FTP 服务的。

机密性不满足的主要原因是密钥 K 是对称密钥，加密和解密都是一样的，而 C 和 K 一定会被攻击者截取。解决的思路是，即使攻击者截取了所有的消息也无法解密获得消息 M，这就需要引入非对称密钥体制。假设 A，B 和 TTP 每个都拥有一个密钥对——公钥和私钥，公钥通过证书公开，而私钥只有自己知道。私钥可以用来作为协议证据的签名来使用，也同样可以解密由公钥加密的消息。首先，发起者 A 使用接收者 B 的公钥加密消息 M，然后再以密钥 K 重新加密这个结果；而在消息 3 中，使用 TTP 的公钥加密密钥 K 发送。这样攻击者由于无法知道 B 和 TTP 的私钥，从而截获的消息也不能解密。整个改进的协议如下：

1. $A \rightarrow B$: f_{NRO}, B, L, C, NRO
2. $B \rightarrow A$: f_{NRR}, A, L, NRR
3. $A \rightarrow TTP$: f_{SUB}, B, L, $eP_{TTP}(K)$, NRR, sub_K
4. $B \leftrightarrow TTP$: f_{CON}, A, B, L, K, con_K
5. $A \leftrightarrow TTP$: f_{CON}, A, B, L, K, con_K

其中 $C = eK[eP_B(M)]$。

结论　本文发现的公平性缺陷同 Dimitrios Petropoulos 等人[9]在 ZG 协议中指出的混乱攻击(shuffle attack)类似，区别是文[9]讨论的是带有时限性的 ZG 协议。Dimitrios 等人直观地给出了攻击，但未用形式化的分析方法。而机密性问题已被多篇文献所指出[9～11]。就解决方法而言，Kim 等人[10]提出了对要发送的消息 M 使用 Diffie-Hellman 密钥交换保证了机密性，而 Dimitrios Petropoulos 等[11]采用的是非对称密钥方法，私钥即可以签名也用来解密公钥加密的消息，在保证机密性的同时没有增加密钥管理的花销。本文给出的改进协议借鉴了 Dimitrios 等人的方法。

Zhou-Gollmann 协议是典型的带有 online 可信第三方(TTP)的非否认协议，其它的非否认协议还包括，不带有 TTP、$inline$ TTP 和 $offline$ TTP 的非否认协议，这类协议同样需要满足上述的性质。本文基于有限状态机和模型检测工具 SPIN，对 ZG 公平非否认协议进行了分析。该研究思路亦可分析其它非否认协议。另外像电子合同签订协议、挂号电子邮件协议等，同样带有不诚实主体参加的协议，它们都需要满足公平性的要求，本文的方法也可起到借鉴的作用。

参考文献

1 Zhou J, Gollmann D. A fair non-repudiation protocol. In: IEEE Computer Society Symposium on Research in Security and Privacy, 1996. 55～61

2 Zhou J, Gollmann D. Towards verification of non-repudiation protocols. In: Proc. of 1998 Intl. Refinement Workshop and Formal Methods Pacific, 1998. 370～380

3 Schneider S. Verifying authentication protocols with CSP. In: IEEE Computer Security Foundations Workshop, IEEE, 1997

4 Sigrid, Rudolph C. Security Analysis of (Un-) Fair Non-repudiation Protocols. In: Formal Aspects of Security (FASec'02) Springer Verlag, LNCS, 2002, 2629: 97～114

5 董荣胜，郭云川，古天龙. 电子商务协议原子性的模型检验分析方法. 计算机科学，2005，32(4)：184～186

6 卿斯汉. 电子商务协议中的可信第三方角色. 软件学报，2003，14(11)：1936～1943

7 Maggi P, Sisto R. Using SPIN to Verify Security Properties of Cryptographic Protocols. In: SPIN'2002 Workshop, 2002

8 邵晨曦，胡香冬，等. 密码协议的 SPIN 建模和验证. 电子学报，2002，12(12A)：2099～2101

9 Petropoulos D, Kotzanikolaou P. Some more improvements on a fair non-repudiation protocol. Journal of Internet Technology, 2002, 4(4)

10 Kim K, Park, Baek J. Improving Fairness and Privacy of Zhou-Gollmann's Fair Non-repudiation Protocol. In: Proc. of ICPP Workshop on Security(IWSEC), IEEE Computer Society, 1999. 140～145

11 Kremer S, Markowitch O, Zhou Jianying. An intensive survey of non-repudiation protocols. Computer Communications, 2002, 25(17): 1606～1621