

适用于多方协议的可否认认证

陈勇, 鲁龙, 曾晟珂, 何明星

(西华大学计算机与软件工程学院, 四川 成都 610039)

摘要: 可否认认证不仅可以像数字签名那样实现消息认证, 且非公开可验证性使其可以满足许多签名不能满足的需求, 如隐私保护等。研究了如何在多方协议中实现完全可否认认证, 利用承诺方案提出了一个适用于多方协议的可否认认证。该方案作为一种一般化的方法, 能够为大多数多方协议实现可否认认证, 如可否认的群密钥协商协议等, 形式化了方案的敌手模型并严格证明了方案的安全性, 为研究可否认认证提供了一种新思路。

关键词: 可否认认证; 多方协议; 承诺; 安全性

中图分类号: TP393

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2016.00059

Deniable authentication for multi-party protocol

CHEN Yong, LU Long, ZENG Sheng-ke, HE Ming-xing

(School of Computer and Software Engineering, Xihua University, Chengdu 610039, China)

Abstract: Deniable authentication can authenticate messages like digital signatures, and its non-public verification can meet many requirements that cannot be met by signatures, such as privacy preservation. How to achieve deniable authentication with multi-party protocol was the study aim. A deniable authentication for multi-party protocol using string commitment scheme was proposed. As a generic method, it can pass the most of protocol to deniable protocol, such as the deniable group key agreement protocol. The adversarial model was formalized and the security was proved strictly. The research content provide a new idea for studying deniable authentication.

Key words: deniable authentication, multi-party protocol, commitment, security

1 引言

数字签名作为密码学的一种基本原语, 其不可伪造性以及不可否认性受到学术界的广泛关注。由于签名过程的不可抵赖性, 数字签名算法不适用于保护用户隐私的认证。为了保护认证过

程中用户的隐私, 可否认认证的概念被提出。可否认认证协议在保证消息认证性的同时, 其认证副本并不能证明协议的参与者参与了认证, 即不具有公开验证性, 这在电子购物、互联网会议等方面有广泛的应用前景。

1998年, Dwork等^[1]首次正式提出可否认认证

收稿日期: 2015-04-17; 修回日期: 2015-05-25; 通信作者: 陈勇, chenrong.sclz@gmail.com

基金项目: 国家自然科学基金资助项目 (No.61402376, No.U1433130); 四川省高校重点实验室开放课题基金资助项目 (No.SZJJ2014-078); 国家科技支撑计划基金资助项目 (No.2011BAH26B04); 教育部春晖计划基金资助项目 (No.Z2014045)

Foundation Items: The National Natural Science Foundation of China (No.61402376, No.U1433130), The Open Research Foundation of Key Laboratory of Colleges and Universities in Sichuan Province (No.SZJJ2014-078), National Science and Technology Support Program (No.2011BAH26B04), Chunhui Project of the Ministry of Education of China (No.Z2014045)

的概念,并利用时间约束机制实现了并发环境中的可否认认证。之后一系列的可否认协议被提出^[2-4]。交互式的认证主要面临并发会话攻击。认证的可否认性要在并发环境中成立,一般采用时间约束机制^[1],然而,这种假设太强。通过使用 Pass 在文献[5]中提出的非编程预言机, Jiang 等^[6]提出一个不使用时间约束的在并发环境中完全可否认的认证协议。Raimondo 等在文献[7]中利用明文可意识性 (plaintext awareness) 证明了 SKEME 协议满足完全可否认性。Jiang^[8]利用“时限加密”,在可认证的密钥交换协议中实现了可否认性。

为了提高认证协议的效率,很多学者关注认证协议的交互轮数,提出了一系列非交互式可否认的认证协议^[9-11]。由于非交互性,这些协议自然满足并发环境,但却仅实现“半可否认性”^[5]。例如, Susilo 在文献[9]中利用 Chameleon Hash 函数实现认证的可否认性,在文献[9]方案中,参与方只能否认通信的内容,却不能否认其参与了认证。Li 等^[11]也提出了一种非交互式的、基于身份的可否认认证协议。该协议也仅达到了半可否认性,因为只有消息接收方才能仿真发送方发的认证副本。即对于某次认证副本,发送方和接收方中至少有一方不能否认参与了认证。可见,非交互式的可否认认证虽然认证效率较高,但是应用限制大。

在这些方案中,研究者研究的模型大多都是一对一的,即一对消息的发送者与接收者。在一些实际应用中,协议需要多方用户的参与。在多方协议中引入可否认认证有以下研究工作。2002年, Naor^[12]将可否认认证与环签名的特性相结合,提出了可否认的环认证,消息的接收者能够确认消息确实来自环中的成员,但不能确认具体是哪一个成员,同时,第三方却不能确认消息来自于环成员。另一个将可否认认证引入到多方环境的协议是可否认的群密钥协商协议,可否认的群密钥协商协议是指多个参与者通过协商得到一个共同的会话密钥,在协商结束后,任何实体都不能证明协商的参与者曾经参与了协议。2006年, Bohli 等^[13]利用 Schnorr 签名算法^[14]首次提出一个可否认的群密钥协商协议,并在 Random 模型中证明协议的安全性。2010年, Zhang 等^[15]改

进了文献[14]中方案的通信效率,且安全性证明不需要随机预言机;2010年, Zhang 等^[16]给出一个构造可否认群密钥协商协议的一般构造方法,该方法可以将无认证的群密钥协商协议转化成可否认的群密钥协商协议。2014年, Steinwandt 等^[17]提出一个构造器可以将一个普通的两轮次的群密钥协商协议转化为一个三轮次的可否认的群密钥协商协议。2016年, Chen 等^[18]利用 DB 协议^[19]提出一个可否认的群密钥协商协议,进一步改进了协议的通信效率。

目前,虽然已有一系列关于可否认的群密钥协商协议的研究,但其可否认性都依赖于最终的会话密钥协商,无法简单地将其扩展到普通的多方协议,如非对称的群密钥协商协议^[20]等。因此,本文旨在通过秘密承诺^[23]的思想设计一个适用于广播信道(如 ad hoc 网络等)中多方协议的可否认认证方案(MDA, deniable authentication for multi-party protocol)。与以往的多方可否认认证(如可否认的群密钥协商)方案不同,本文方案不再关心消息的具体内容,因此,具有很好的扩展性。另外,本文定义了 MDA 方案的敌手模型,并对方案的安全性给出了证明。

2 基础知识

2.1 困难性问题

本文的安全性主要是基于判定性 Diffie-Hellman (DDH) 假设。令 G 为阶是素数 q 的乘法群, g 是群 G 的生成元。

DDH 问题: 给定 g 、 g^a 、 g^b 和 g^c , 其中, $a, b, c \in \mathbb{Z}_q^*$ 是未知的, 判断 g^{ab} 和 g^c 是否相等。

DDH 假设: 设 Δ_{DDH} 是一个多项式时间的算法, 用

$$\text{Adv}_{\text{DDH}} = \left| \Pr \left[D_{\text{DDH}}(g, g^a, g^b, g^c) = 1 \right] - \Pr \left[D_{\text{DDH}}(g, g^a, g^b, g^{ab}) = 1 \right] \right|$$

表示 Δ_{DDH} 成功解决 DDH 问题的优势, DDH 假设认为 Adv_{DDH} 是可以忽略不计的。

2.2 多方计算协议

在只存在被动敌手的公共信道上, 给出多方协议^[21]的一般定义。令 $P_1, \dots, P_n$ 表示信道中的 n 个对等用户, q 表示系统的安全参数。假设协

议共执行 l 个轮次, 则多方协议的基本通信模型如下。

1) 参与方 P_i 选择 x_i 作为自己的输入, 同时选择一个随机值 r_i 。

2) 令 $t=1$, 参与方 P_i 依次执行以下步骤, 直到 $t \geq l$ 。

① P_i 产生消息 $\{m_{i,j,t} | j=1, \dots, n, j \neq i\}$, 其中, $m_{i,j,t}$ 表示在本轮次将要发送给参与方 P_j 的消息。

② P_i 收到来自各个参与方的消息 $\{m_{j,i,t} | j=1, \dots, n, j \neq i\}$ 。

③ 令 $t=t+1$ 。

3) 参与方 P_i 产生输出 o_i 。

在只存在被动敌手的情况下, 多方协议只需满足私密性, 即当存在敌手监听整个信道时, 参与方 P_i 所产生的输出 o_i 只能被它自己所了解。然而, 当信道中存在主动敌手时, 主动敌手可以插入、删除、篡改任何消息。此时, 协议不仅需要满足私密性, 还必须满足认证性。

私密性: 参与方 P_i 产生的输出 o_i 只能被参与方自己所了解, 而对于任何第三方来说, o_i 在参数空间内是均匀分布的。

认证性: 在多项式时间内, 任何敌手都不能以不可忽略的概率伪装成其他诚实参与方参与并完成协议。

最常见的多方协议的例子便是群密钥协商协议。

2.3 字符串承诺方案

承诺方案作为一种基础性密码工具, 常被用在构造零知识证明^[22]或可否认认证中。在承诺方案中, 承诺者将承诺值和秘密值绑定在一起, 承诺结束后, 承诺者打开秘密值并验证。安全的承诺方案必须满足私密性和绑定性。字符串承诺方案^[22]一般由以下几个算法组成。

1) 初始化阶段 $\text{Gen}(1^k)$: 可信方运行初始化算法生成系统参数 (p, g, h) 。

2) 承诺阶段 $\text{Commit}(p, g, h, v)$: 承诺者随机地选择 s , 用 v 表示要承诺的串, 计算 $c = g^s h^v$ 。

3) 打开阶段 $\text{Open}(p, g, h, v, s, c)$: 承诺者公开 (s, v) 。

4) 验证阶段 $\text{Verify}(p, g, h, v, s, c)$: 验证者验证等式 $c = g^s h^v$ 是否成立。

私密性: 在承诺值被打开之前, 承诺值保持

私密性。

绑定性: 如果敌手能够破坏协议的绑定性, 即能够输出一个 $(s', v') \neq (s, v)$, 使 $c = g^{s'} h^{v'}$, 则该敌手一定能够解决离散对数问题 (g, g^x) 。令 $h = g^x$, 有 $g^{s'} h^{v'} = g^s h^v$, 则 $g^{s'-s} = h^{v-v'}$, 可以计算 $x = \frac{s'-s}{v-v'}$ 。因此, 该承诺方案满足绑定性。

3 模型

3.1 相关符号

首先, 给出方案所用的主要符号, 如表 1 所示。

符号	描述
k	安全参数
l	多方计算协议的通信轮次
π_i^l	用户 P_i 的第 l_i 个会话实例
sid_i^l	会话实例 π_i^l 的会话 ID
pid_i^l	参与实例 π_i^l 的所用用户的集合, 其中, $P_i \in pid_i^l$
$stat_i^l$	用户 P_i 在执行实例 π_i^l 过程中的内部状态信息
$MAC(\cdot)$	一个不可伪造的消息认证编码
H	密钥学散列函数, $H: \{0,1\}^* \rightarrow G$
$\text{negl}(k)$	基本安全参数 k 的可忽略函数
m_i^t	用户 P_i 在参与多方计算协议时第 t 轮次所广播的消息
n	参与实例 π_i^l 的用户的数量

3.2 系统模型

本方案的基本思想是: 在协议开始初, 参与者承诺一个消息认证码密钥; 然后, 参与者利用私钥对消息认证码密钥进行签名; 接着, 参与者利用消息认证码密钥对多方协议的通信进行认证, 这里需要认证编码函数 MAC ; 在协议结束后, 参与者公开自己的消息认证码密钥, 其他参与者验证多方协议消息的合法性。

因此, MDA 方案包括 6 个算法, 下面对各个算法进行介绍。

1) $\text{KeyGen}(1^k) \rightarrow (sk_i, pk_i)$: 可信方 T 利用该算法为用户 P_i 生成公私钥对 (sk_i, pk_i) , 并通过安全信道发送给用户 P_i 。

2) $\text{Commit}(i, l_i, pk_i, pk_2, \dots, pk_n) \rightarrow C_i$: 用户 P_i 随机地选择 s_i 作为自己的消息认证码密钥, 并利用所有输入的公钥对 s_i 做出承诺。

3) $\text{Authenticate}(i, l_i, sk_i, C_1, C_2, \dots, C_n) \rightarrow A_i$: 用

户 P_i 利用私钥对所有的消息认证码密钥的承诺进行认证。

4) $\text{Communicate}(i, l_i, s_i, m_i') \rightarrow \delta_i'$: 用户 P_i 利用消息认证码密钥 s_i 对通信的消息 m_i' 进行认证。

5) $\text{Open}(i, l_i, C_i) \rightarrow s_i$: 用户 P_i 打开消息认证码密钥的承诺。

6) $\text{Verify}(\{C_j, s_j\}_{j=1}^n, \{A_j\}_{j=1}^n, \{\delta_j'\}_{j=1}^n) \rightarrow 0 \text{ or } 1$: 用户 P_i 对各个参与者的消息认证码进行验证, 若任何一个验证不通过, 则拒绝接受, 输出 0; 否则, 输出 1。

需要注意的是, **Communicate** 阶段不需要在 **Commit** 阶段和 **Authenticate** 阶段之后, 可以在这 2 个阶段之前或之间。为了达到较高的通信效率, **Communicate** 阶段一般与 **Commit** 阶段和 **Authenticate** 阶段同时进行。

3.3 安全模型

MDA 方案使每个用户确信收到的消息的确来自诚实用户, 且发送者对消息的认证是可否认的, 因此, 其安全性包括完整性、健壮性、可否认性。

1) 完整性。如果所有的参与者都诚实地执行协议, 则 $P_j \in \text{pid}$ 接受消息 m_i' 来自用户 P_i 。

2) 健壮性。方案的健壮性意味着任何用户都不能伪装成诚实用户 P_i 对消息 m_i' 进行认证。接下来, 通过一个游戏来定义方案的健壮性。

初始化阶段: 挑战者 X 运行 $\text{Setup}(1^k)$ 生成系统参数, 并将公开的系统参数发送给敌手 A。

训练阶段: 挑战者 X 回答敌手 A 的一系列查询。

$\text{Commit}(i, l_i, M)$: 挑战者 X 对消息认证码做出承诺, 并返回给敌手 A。

$\text{Authenticate}(i, l_i, M)$: 如果集合 pid_i^l 中的所有用户都完成了对消息认证码的承诺, 挑战者利用对消息认证码的承诺进行可模拟的签名认证, 并返回给敌手 A。

$\text{Communicate}(i, l_i, M)$: 挑战者利用消息认证码对消息进行认证。

$\text{Open}(i, l_i, M)$: 如果多方协议已经完成, 挑战者向敌手返回用户的消息认证码和承诺值。

$\text{Corrupt}(i)$: 敌手请求腐蚀用户 P_i , 挑战者将用户 P_i 的私钥和内部数据返回给敌手 A。此外, 用户 P_i 将来的动作全部由敌手 A 控制。

在定义游戏的挑战阶段前, 先定义会话实例 π_i^l 的新鲜性。如果没有以下情况发生: ①用户 P_i 被 $\text{Corrupt}(i)$ 查询; ②一个 $\text{Open}(i, l_i)$ 被敌手请求; ③ pid_i^l 中超过 $|\text{pid}_i^l| - 2$ 个用户被 Corrupt 查询, 其中, $|\text{pid}_i^l|$ 表示 pid_i^l 中成员的个数, 则认为会话实例 π_i^l 是满足新鲜性的。

$\text{Challenge}(i, l_i, s_i, v_i)$: 敌手 A 向挑战者 X 提交承诺值 (s_i, v_i) , 如果承诺值通过验证, 挑战者返回 1; 否则, 返回 0。

Challenge 查询仅被允许执行一次, 且会话实例 π_i^l 必须满足新鲜性。如果挑战者 X 返回 1, 则说明敌手赢得了游戏, 用 Succ_A 表示敌手赢得游戏的事件。如果敌手赢得游戏的概率是可忽略不计的, 即 $\Pr[\text{Succ}_A] \approx \text{negl}(k)$, 则认为方案是满足健壮性的。

3) 可否认性。方案的可否认性是指协议的通信副本可以由敌手独自构造出来。即存在一个仅了解敌手知识的模拟者 Σ , Σ 不能访问用户的私钥, 除非用户被腐蚀, 构造的通信副本的分布与真实通信的分布是一致的。正式地说, 用 View_Σ 表示模拟者 Σ 模拟的通信副本, 用 View_R 表示真实执行过程的副本, 令 Δ 表示某个多项式时间的区分者, 给出以下可否认的定义。如果对于任何的敌手 A 和所有的区分者 Δ , 总存在一个模拟者 Σ , 使 $|\Pr[\Delta(\text{View}_\Sigma)=1] - \Pr[\Delta(\text{View}_R)=1]| \approx \text{negl}(k)$, 则认为协议是满足可否认的。

4 协议的实现

本节给出 MDA 方案的具体实现, 方案的灵感来源于文献[1]的零知识证明。下面给出方案的详细过程。

假设每个用户 P_i 都拥有一对公私钥对 $(sk_i, pk_i) = (\alpha_i, g^{\alpha_i})$ 。在承诺阶段, 用户 P_i 随机地选择 s_i , 作为多方协议的消息认证码, 并对其做出承诺; 在认证阶段, 用户 P_i 利用可模拟的签名算法对承诺阶段的承诺进行签名认证; 在通信阶段, 用户 P_i 利用消息认证码对多方协议的消息进行认证; 在承诺打开阶段, 用户 P_i 公开自己的承诺值; 在最后的验证阶段, 用户 P_i 利用通信阶段打开的消息认证码对承诺阶段的承诺进行验证, 同时对用户的身份进行验证, 并检查

消息是否合法。

Setup : k 为系统的安全参数; G 是一个阶为素数 q 的乘法群; g 是群 G 的生成元; H 是一个密码学散列函数 $\{0,1\}^* \rightarrow G$ 。

KeyGen : 用户 P_i 随机地选择 $\alpha_i \leftarrow Z_q$, 令用户 P_i 的私钥为 $sk_i = \alpha_i$ 、公钥为 $pk_i = g^{\alpha_i}$ 。

Commit : 用户 P_i 随机地选择 $s_i \leftarrow Z_q$ 作为本次会话的消息认证码密钥; 另外, 随机地选择 $v_i \leftarrow Z_q$ 作为本次会话的临时私钥, 计算 $V_i = g^{v_i}$; 然后, 依次计算 $C_j = g^{s_i} (pk_j)^{v_i}$, 其中, $j \in \{1, \dots, n\}$; 接着计算 $c_i = H(C_1 \parallel C_2 \parallel \dots \parallel C_n \parallel g^{s_i} \parallel g^{v_i})$; 最后, 广播消息 $(sid, pid, P_i, c_i, V_i)$ 。

Authenticate : 当用户 P_i 收到 $\{sid, pid, P_j, c_j, V_j\}_{j \in \{1, \dots, i-1, i+1, \dots, n\}}$ 后, 依次计算 $A_j = g^{s_i} (V_j)^{\alpha_i}$, 其中, $j \in \{1, \dots, n\}$; 接着计算 $a_i = H(A_1 \parallel \dots \parallel A_n \parallel c_1 \parallel \dots \parallel c_n \parallel V_1 \parallel \dots \parallel V_n)$; 最后广播消息 (sid, pid, P_i, a_i) 。

Communicate : 用户 P_i 计算 $\delta'_i = MAC_{s_i}(m'_i)$ 并广播 $(sid, pid, P_i, \delta'_i, m'_i)$ 。

Open : 用户 P_i 确认是否收到以上各个阶段的全部消息, 如果是, 广播消息 $(sid, pid, P_i, s_i, v_i)$; 否则, 等待。

Verify : 用户 P_i 依次验证 $V_j = g^{v_j}$ 是否成立, 其中, $j \in \{1, \dots, n\}$, 如果不成立, 则中止; 否则, 继续。依次计算 $C_j^l = g^{s_j} (pk_l)^{v_j}$, 其中, $l \in \{1, \dots, n\}$, 接着计算 $c_j' = H(C_j^l \parallel \dots \parallel C_j^n \parallel g^{s_j} \parallel g^{v_j})$ 。验证 $c_j = c_j'$, 如果不成立, 则终止; 否则, 继续。采用这种方式对所有的用户 $P_j \in pid$ 进行验证, 如果存在任何一个不成立的验证, 则中止协议; 否则, 继续。依次计算 $A_j^l = g^{s_j} (pk_j)^{v_l}$, 其中, $l \in \{1, \dots, n\}$, 接着计算 $a_j^l = H(A_j^l \parallel \dots \parallel A_j^n \parallel c_1 \parallel \dots \parallel c_n \parallel V_1 \parallel \dots \parallel V_n)$, 验证 $a_j = a_j^l$, 如果不成立, 则终止; 否则, 继续。采用这种方式对所有用户进行 $P_j \in pid$ 进行验证; 如果存在任何一个不成立的验证, 则中止协议; 否则, 继续。依次验证用户 P_j 的消息 m'_j 是否合法, 即 $\delta'_j = MAC_{s_j}(m'_j)$, 如果存在不合法, 则中止。如果以上验证均成立, 用户 P_i 接受所收到的消息来自于各个用户。

5 安全性证明

本节对方案的安全性进行证明, 很容易理解方案是满足完整性的, 所以, 接下来分别证明方案的可否认性和健壮性。

5.1 可否认性

MDA 方案的可否认性使参与者可以否认曾经参与了多方协议消息的认证, 即参与者可以否认曾参与了多方协议。本文通过构造一个模拟者 Σ , 在不允许模拟者 Σ 访问未腐蚀用户私钥的情况下, 显示模拟者生成的通信副本与真实执行过程的通信副本不可区分, 从而证明方案是满足可否认性的。

定理 1 MDA 方案满足可否认性, 如果 H 是一个随机预言机。

证明 构造模拟者 Σ , 它的输入仅包括用户公钥等公共参数以及被腐蚀用户的私钥。用 A_d 表示破坏方案可否认性的敌手。在模拟过程中, 模拟者 Σ 必须回答敌手的一系列询问, 如 Commit、Authenticate、Communicate、Open 以及 Corrupt。用 $View_\Sigma$ 表示在模拟过程中的通信副本, 用 $View_R$ 表示真实协议的执行副本。可否认性显示对于任何敌手 A_d 和所有的区分者 Δ , 都存在一个模拟者 Σ , 使 $View_\Sigma$ 和 $View_R$ 是不可区分的。

接下来, 构造模拟者 Σ 来回答 A_d 的一系列查询。

Commit(i, l_i, M): 模拟者 Σ 随机地选择 $v_i \leftarrow Z_q$ 和 $s_i \leftarrow Z_q$, 依次计算 $C_j = g^{s_i} (pk_j)^{v_i}$ 与 $V_i = g^{v_i}$, 其中, $j \in \{1, \dots, n\}$, 并计算 $c_i = H(C_1 \parallel C_2 \parallel \dots \parallel C_n \parallel g^{s_i} \parallel g^{v_i})$; 最后, 更新内部信息 $stat_i^l = \{s_i, v_i, V_i, \{c_j\}\}$, 广播消息 $(sid, pid, P_i, c_i, V_i)$ 。

Authenticate(i, l_i, M): 模拟者 Σ 检查用户 P_i 是否被腐蚀: 1) 如果用户 P_i 被腐蚀, 则模拟者 Σ 利用用户 P_i 的私钥 α_i 依次计算 $A_j = g^{s_i} (V_j)^{\alpha_i}$, 再计算 $a_i = H(A_1 \parallel \dots \parallel A_n \parallel c_1 \parallel \dots \parallel c_n \parallel V_1 \parallel \dots \parallel V_n)$; 2) 如果用户 P_i 没有被腐蚀, 模拟者 Σ 从 $\{stat_j^l\}$ 中取出 v_i , 再计算 $A_j = g^{s_i} (pk_j)^{v_i}$, 接着计算 $a_i = H(A_1 \parallel \dots \parallel A_n \parallel c_1 \parallel \dots \parallel c_n \parallel V_1 \parallel \dots \parallel V_n)$ 。最后, 模拟者 Σ 广播消息 (sid, pid, P_i, a_i) 。

Communicate(i, l_i, M): 模拟者 Σ 利用 s_i 认证消息 m'_i , 即计算 $\delta'_i = MAC_{s_i}(m'_i)$, 广播消息

$(sid, pid, P_i, \delta'_i, m'_i)$ 。

Open (i, l_i) : 模拟者 Σ 公开承诺值, 即广播消息 $(sid, pid, P_i, s_i, v_i)$ 。

Corrupt (i) : 模拟者 Σ 获取用户 P_i 的长期私钥, 同时标记出被腐蚀的用户 P_i 。

显然, 模拟者 Σ 能够在不引起任何区别的情况下成功地回答敌手 A_d 的所有查询, 所以, 该方案实现了可否认性。

5.2 健壮性

方案的健壮性是指任何敌手, 在不了解用户 P_i 私钥的前提下, 都不能伪装成用户 P_i 参与协议。本文显示, 如果一个敌手能够破坏方案的健壮性, 则该敌手一定能以不可忽略的概率解决 DDH 困难性问题。

定理 2 如果 DDH 困难性假设成立, 且消息认证编码函数 $MAC(\cdot)$ 是安全的, H 是一个随机预言机, 那么方案是满足健壮性的。

证明 用 A 表示攻击方案健壮性的敌手, 如果 A 能以不可忽略的概率攻破方案的健壮性, 则挑战者 X 就可以利用敌手 A 来解决 DDH 困难性问题。挑战者 X 被给出实例 (g, g^a, g^b, g^c) , 判断 g^{ab} 与 g^c 是否相等。

接下来, 说明挑战者 X 如何利用敌手 A 来解决困难性问题。

用 q_{com} 、 q_{auth} 、 q_{open} 、 q_{cor} 分别表示 Commit 查询、Authenticate 查询、Open 查询以及 Corrupt 查询的次数。用 $Collision$ 表示 H 发生碰撞的事件, 其中, $\Pr[Collision] \leq \frac{(q_{com} + q_{auth})^2}{2^q}$ 。

系统建立阶段: 挑战者 X 被要求解决 (g, g^a, g^b, g^c) 的 DDH 困难性问题。在开始阶段, 挑战者 X 设置用户 P_i 的公钥为 $pk_i = g^a$, 用户 P_i 的私钥 $sk_i = a$ 对于挑战者 X 来说是未知的。接下来, 挑战者 X 随机地选择 $\alpha_i \leftarrow Z_q^*$, 其中, $i \in \{1, \dots, i^* - 1, i^* + 1, \dots, n\}$, 令用户 P_i 的私钥为 $sk_i = \alpha_i$ 、公钥为 $pk_i = g^{\alpha_i}$ 。另外, 挑战者 X 令 $l_i \in \{1, \dots, q_{com}\}$ 。最后, 挑战者公布方案的系统参数 $(g, pk_1, pk_2, \dots, pk_n, MAC, H)$ 。

接下来, 敌手开始进行游戏, 首先是训练阶段, 具体如下。

H 查询: 挑战者 X 维持一个初始化为空的列

表 $list_H$ 。当输入消息 m_i 时, 挑战者 X 检查 $list_H$ 里是否存在 (m_i, θ_i) 。如果存在, 则返回 θ_i ; 否则, 挑战者 X 随机地选择 θ_i , 同时将 (m_i, θ_i) 存入到 $list_H$ 中并返回 θ_i 。

Commit (i, l_i, M) : 1) 如果 $(i, l_i) = (i^*, l_i^*)$, 挑战者 X 令 $V_i = g^b$, 即 $v_i = b$ 但未知, 再随机地选择 s_i , 依次计算 $C_j = g^{s_i}(V_i)^{\alpha_j}$, 其中, $j \in \{1, \dots, i-1, i+1, \dots, n\}$, 另计算 $C_i = g^{s_i} \cdot g^c$, $c_i = H(C_1 \parallel C_2 \parallel \dots \parallel C_n \parallel g^{s_i} \parallel V_i)$ 。2) 否则, 挑战者 X 随机地选择 s_i 和 v_i , 依次计算 $C_j = g^{s_i}(pk_j)^{v_i}$, 再计算 $V_i = g^{v_i}$ 和 $c_i = H(C_1 \parallel C_2 \parallel \dots \parallel C_n \parallel g^{s_i} \parallel g^{v_i})$, 完成后, 挑战者 X 返回 (c_i, V_i) , 更新 $stat_i^l = \{s_i, v_i, V_i, \{C_j\}\}$ 。

Authenticate (i, l_i, M) : 1) 如果 $(i, l_i) = (i^*, l_i^*)$, 挑战者 X 计算 $A_i = g^{s_i} \cdot g^c$, 再从 $\{stat_j^l\}$ 中取出 $\{v_j\}$ 计算 $A_j = g^{s_i}(pk_i)^{v_j}$, 其中, $j \in \{1, \dots, i-1, i+1, \dots, n\}$, 再计算 $a_i = H(A_i \parallel \dots \parallel A_n \parallel c_1 \parallel \dots \parallel c_n \parallel V_1 \parallel \dots \parallel V_n)$; 2) 如果 $i = i^*$ 但 $l_i \neq l_i^*$, 对于所有参与者 $\{P_j\}$, 挑战者 X 检查 P_j 是否被腐蚀, 如果未腐蚀, 挑战者 X 从 $\{stat_j^l\}$ 中取出 $\{v_j\}$ 计算 $A_j = g^{s_i}(pk_i)^{v_j}$, 否则, 挑战者 X 随机地选择 $\hat{A}_j$, 最后, 访问 H 查询得到 $a_i = H(A_i \parallel \dots \parallel A_n \parallel c_1 \parallel \dots \parallel c_n \parallel V_1 \parallel \dots \parallel V_n)$; 3) 如果 $i = i^*$ 且 $l_i = l_i^*$, 挑战者 X 直接计算 $A_j = g^{s_i}(V_j)^{\alpha_j}$, 其中, $j \in \{1, \dots, n\}$; 再计算 $a_i = H(A_i \parallel \dots \parallel A_n \parallel c_1 \parallel \dots \parallel c_n \parallel V_1 \parallel \dots \parallel V_n)$, 最后, 挑战者 X 返回 (a_i) 。

Communicate (i, l_i, M) : 挑战者 X 计算 $\delta'_i = MAC_{s_i}(M)$, 并返回 (δ'_i, M) 。

Open (i, l_i) : 如果 $(i, l_i) = (i^*, l_i^*)$, 挑战者 X 终止游戏, 用 $Event_1$ 表示; 否则, 挑战者 X 检查 P_i 是否被腐蚀, 如果未被腐蚀, 挑战者 X 返回 (s_i, v_i) , 如果被腐蚀, 敌手向挑战者 X 提交 (s_i, v_i) , 挑战者计算 $A_i = g^{s_i}(pk_j)^{v_i}$, 并用 A_i 替换 $list_H$ 中的 $\hat{A}_i$ 。

Corrupt (i) : 如果 $i = i^*$, 挑战者 X 中止游戏, 用 $Event_2$ 表示; 否则, 挑战者 X 返回私钥 α_i 和内部数据。

经过一系统的训练之后, 敌手 A 提交挑战查

询, 挑战部分如下。

Challenge(i, l_i, s'_i, v'_i): 如果 $(i, l_i) \neq (i^*, l_i^*)$, 挑战者中止游戏, 用 $Event_3$ 表示该事件; 否则, 敌手 A 向挑战者 X 提交自己的结果 (s'_i, v'_i) 。如果 (s'_i, v'_i) 成功地通过验证, 则认为敌手 A 赢得了游戏。显然, 如果敌手成功地赢得游戏, 则挑战者 X 输出 1; 否则, 挑战者 X 随机抛一块硬币, 并根据硬币来返回 0 或者 1。用 $\Pr[Succ_A]$ 表示敌手 A 赢得游戏的概率, 用 Adv_{DDH} 表示挑战者成功解决 DDH 的优势, 则 $Adv_{DDH} = \Pr[Succ_A] \cdot \Pr[\neg Event_1] \cdot \Pr[\neg Event_2] \cdot \Pr[\neg Event_3] \cdot \Pr[\neg Collision]$, 将等式两边同时除以 $\Pr[\neg Event_1] \cdot \Pr[\neg Event_2] \cdot \Pr[\neg Event_3] \cdot \Pr[\neg Collision]$, 得到

$$\Pr[Succ_A] = \frac{\Pr[Adv_{DDH}]}{\Pr[\neg Event_1] \cdot \Pr[\neg Event_2] \cdot \Pr[\neg Event_3] \cdot \Pr[\neg Collision]}$$

由于 Adv_{DDH} 是可以忽略不计的概率, 而 $\Pr[\neg Event_1] \cdot \Pr[\neg Event_2] \cdot \Pr[\neg Event_3] \cdot \Pr[\neg Collision]$ 是多项式可计算的, 所以, $\Pr[Succ_A]$ 是可忽略不计的, 即敌手 A 赢得游戏的概率是可以忽略不计的, 协议的健壮性得证。

6 结束语

本文提出了一个适用于多方协议的可否认认证方案, MDA 协议可以有效地防止敌手对通信的消息进行修改或插入, 从而冒充合法用户去欺骗其他诚实用户。同时, 区别于数字签名的不可否认性, 即公开可验证性, 可否认协议的通信副本不能证明用户曾经参与了协议, 有效地保护了参与者的隐私。本文在前人的基础上, 形式化了方案的敌手模型, 并证明了方案满足可否认性和健壮性。相比于可否认的群密钥协商协议等具体方案, 本文方案具有很好的扩展性, 能够被应用到多数多方协议中实现可否认认证。

参考文献:

[1] DWORK C, NAOR M, SAHAI A. Concurrent zero-knowledge[J]. Journal of the ACM, 2004, 51(6):851-98.

[2] DENG X, LEE C H, ZHU H. Deniable authentication protocols[J]. IEEE Computers and Digital Techniques, 2001, 148(2):101-104.

[3] LEE W B, WU C C, TSAUR W J. A novel deniable authentication protocol using generalized ElGamal signature scheme[J]. Information Sciences, 2007, 177(6):1376-1381.

[4] KAR J. ID-based deniable authentication protocol based on diffie-hellman problem on elliptic curve[J]. International Journal of Network Security, 2013, 15(5):357-364.

[5] PASS R. On deniability in the common reference string and random oracle model[C]//Advances in Cryptology, Berlin. c2003:316-337.

[6] JIANG S. Deniable authentication on the Internet[M]. Berlin: Springer, 2008:298-312.

[7] RAIMONDO M D, GENNARO R, KRAWCZYK H. Deniable authentication and key exchange[C]//ACM Conference on Computer and Communications Security. c2006:400-409.

[8] JIANG S. Timed encryption with application to deniable key exchange[J]. Theoretical Computer Science, 2014(560): 172-189.

[9] SUSILO W, MU Y. Deniable ring authentication revisited[C]//Applied Cryptography and Network Security, Berlin. c2004: 149-163.

[10] WANG L, ZHANG G, MA C. ID-based deniable ring authentication with constant-size signature[J]. Frontiers of Computer Science in China, 2008, 2(1):106-12.

[11] LI F, XIONG P, JIN C. Identity-based deniable authentication for ad hoc networks[J]. Computing, 2014, 96(9): 843-853.

[12] NAOR M. Deniable ring authentication[C]//Advances in Cryptology, Berlin. c2002:481-498.

[13] BOHLI J M, STEINWANDT R. Deniable group key agreement[C]//Progress in Cryptology, Berlin. c2006:298-311.

[14] SCHNORR C P. Efficient identification and signatures for smart cards[C]//Advances in Cryptology, Berlin. c1989:239-252.

[15] ZHANG Y, WANG K, LI B. A deniable group key establishment protocol in the standard model[C]//Information Security, Practice and Experience, Berlin. c2010: 308-323.

[16] 张雅哲, 徐海霞, 李宝. 可否认群密钥协商协议的一般化构造方式[J]. 通信学报, 2011, 32(3):143-149.

ZHANG Y Z, XU H X, LI B. But denied group key agreement protocol way of generalized structure[J]. Journal of Communications, 2011, 32(3): 143-149.

[17] STEINWANDT R, CORONA A S. Scalable attribute-based group key establishment: from passive to active and deniable[J]. Applicable Algebra in Engineering, Communication and Computing, 2014, 25(1/2):1-20.

[18] 陈勇, 何明星, 曾晟珂, 等. 两轮次的可否认的群密钥协商协议[J]. 密码学报, 2016, 3(2):137-146.

CHEN Y, HE M X, ZENG S K, et al. Two rounds of deniable group key agreement protocol[J]. Journal of Password, 2016, 3(2): 137-146.

[19] DUTTA R, BARUA R. Provably secure constant round contribu-

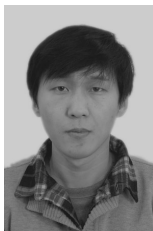
tory group key agreement in dynamic setting[J]. IEEE Transactions on Information Theory, 2008, 54(5):2007-2025.

- [20] CHEN Y, HE M, ZENG S, et al. Universally composable asymmetric group key agreement protocol[C]//The 10th International Conference on Information, Communications and Signal Processing(ICICS), IEEE. c2015.
- [21] CANETTI R. Security and composition of multiparty cryptographic protocols[J]. Journal of Cryptology, 2000, 13(1):143-202.
- [22] FEIGE U, FIAT A, SHAMIR A. Zero-knowledge proofs of identity[J]. Journal of Cryptology, 1988, 1(2):77-94.
- [23] CHAUM D, HEIJST E V, PFITZMANN B. Cryptographically strong undeniable signatures, unconditionally secure for the signer[C]//The 11th Annual International Cryptology Conference on Advances in Cryptology. c1991:470-484.

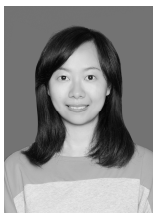
作者简介:



陈勇(1991-), 男, 四川阆中人, 西华大学硕士生, 主要研究方向为密码与信息安全。



鲁龙(1989-), 男, 山东临沐人, 西华大学硕士生, 主要研究方向为密码与信息安全。



曾晟珂(1982-), 女, 四川成都人, 博士, 西华大学讲师, 主要研究方向为密码与信息安全。



何明星(1964-), 男, 四川巴中人, 博士, 西华大学教授, 主要研究方向为密码与信息安全。