分类号_____ 密 级_____

UDC _____

学 位 论 文

基于非交互式零知识证明的匿名签名方案的研究

作 者 姓 名： 王兰兰

指 导 教 师： 周福才 教授

东北大学信息科学与工程学院

申请学位级别： 硕士 学 科 类 别 ： 工学

学科专业名称： 计算机应用技术

论文提交日期： 2011 年 6 月 22 日 论文答辩日期： 2011 年 6 月 26 日

学位授予日期： 答辩委员会主席： 马宗民 教授

评 阅 人 ： 张斌 教授 赵德平 教授

东 北 大 学

2011 年 6 月

A Thesis for the Degree of Master in Computer Application Technology

Research on Anonymous Signature Scheme Based on Non-interactive Zero-knowledge Proof

by Wang Lanlan

Supervisor: Professor Zhou Fucai

Northeastern University

June 2011

独创性声明

本人声明所呈交的学位论文是在导师的指导下完成的。论文中取得的研究成果除加以标注和致谢的地方外，不包含其他人已经发表或撰写过的研究成果，也不包括本人为获得其他学位而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示诚挚的谢意。

学位论文作者签名：王[illegible]

签字日期：2011.6.27

学位论文版权使用授权书

本学位论文作者和指导教师完全了解东北大学有关保留、使用学位论文的规定：即学校有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人同意东北大学可以将学位论文的全部或部分内容编入有关数据库进行检索、交流。

作者和导师同意网上交流的时间为作者获得学位后：

半年 □　一年□　一年半☑　两年□

学位论文作者签名：王[illegible]　导师签名：[illegible]

签字日期：2011.6.27　签字日期：2011.6.27

基于非交互式零知识证明的匿名签名方案的研究

摘 要

本文对非交互式零知识证明（NIZK）理论进行了深入研究，针对实际应用中的匿名需求以及当前多数匿名签名的通信代价和计算代价较高，不支持离线状态、不能抵抗选择密文攻击等问题，提出了将非交互式证明和签名方案结合的思想，具体工作如下：

使用三素数组合阶双线性群理论构建了一个基于 BMW 模型的高效组签名方案 GSCOBG，并引进了 Groth-Sahai 等人提出的 NIZK 证明理论，解决了传统组签名通信效率低、不能抵抗选择密文攻击等问题。方案中签名的大小是一个常量而非依赖于其他系统参数。同时给出了严格的安全性证明，并将 GSCOBG 方案与已有的典型方案分别在效率和安全性方面进行了比较，结果表明该方案在这两方面均具有优势。

利用 GS 证明系统的新特性，构建了非交互式 BB+签名方案和非交互式 CL+签名方案，并对其安全性进行了严格的形式化定义和证明，方案满足匿名性（包括用户匿名性和签名者匿名性）、零知识性和不可伪造性，达到了以匿名方式进行身份认证的目的。非交互式签名方案作为匿名认证方法的原型，具有较高的应用价值，比如应用在电子商务系统中的电子拍卖系统、电子投票系统、可信计算平台等。本文将非交互式 BB+签名方案和非交互式 CL+签名方案应用于匿名电子拍卖协议中，分别构建了公开拍卖协议和封闭拍卖协议，与传统的电子拍卖相比，该方案具有较强的匿名性、可追踪性、非交互性，同时降低了通讯代价和存储代价。

研究结果表明，将非交互式零知识证明理论应用于签名方案中有助于增强传统签名方案的安全性，同时可以减少通讯代价，保证实际应用中离线的需求。

关键词：数字签名；组签名方案；承诺方案；非交互式零知识证明；电子拍卖

Reasearch on Anonymous Signature Scheme Based on Non-interactive Zero-knowledge Proof

Abstract

In this paper we make an intensive study of non-interactive zero knowledge, and in view of the anonymous requirement in practice, the inefficiency of the most anonymous signatures, and schemes without off-line mechanism or CCA secure, we proposed the combination of NIZK with the signatures. The main works are as follows:

We construct an efficient group signature scheme called GSCOBG, which is based on BMW model in the composite order bilinear groups of three primes. To solve the above problem, we use Groth-Sahai proof system. The size of signature is a constant rather than relying on other parameters, and the scheme is secure under standard model. We compare the security and the efficiency respectively with the similar group signatures, and achieve the advantage.

We construct non-interactive BB+ and CL+signature schemes and prove the schemes' security, which satisfy the properties such as anonymity(including signers and users), zero knowledge and Unforgeability and achieve identity authentication in the way of anonymity. The non-interactive signatures are used as anonymous authentication in the E-Commerce such as Electronic Auction, Electronic Voting and Trusted Computation. We apply the non-interactive BB+ and CL+ signatures in the anonymous Electronic Auction, and construct public auction protocol and sealed auction protocol. Comparing with the conventional schames, our schemes have stronger anonymity, traceability and non-interact, and reduce the communication cost.

As a result, NIZK theory is useful to improve the security and efficiency of the traditional signatures, and provide off-line support.

Keywords: digital signature; commitment; NIZK; Electronic Auction

目 录

独创性声明 I

摘 要 II

ABSTRACT III

第 1 章 绪 论 1

1.1 研究背景 1

1.2 国内外研究现状 2

1.3 本文研究内容 4

1.4 本文组织结构 5

第 2 章 非交互式零知识证明理论基础 7

2.1 数学基础 7

2.1.1 Co-GDH 群与双线性配对函数 7

2.1.2 组合阶双线性群 7

2.1.3 安全性假设难题 8

2.2 承诺方案 10

2.2.1 同态承诺方案 10

2.2.2 数字签名 11

2.4 非交互式零知识证明系统 12

2.4.1 零知识证明 12

2.4.2 非交互式零知识 14

2.5 Groth-Sahai 证明系统 17

2.5.1 GS 证明系统的建立过程 17

2.6 本章小结 19

第 3 章 基于 NIZK 的 GSCOBG 组签名方案 21

3.1 组签名的安全模型和安全性定义 21

3.1.1 安全模型 22

3.1.2 安全性定义 22

3.2 GSCOBG 方案描述 23

3.3 安全性证明及效率分析 26
3.3.1 安全性证明 26
3.3.2 效率分析与对比 33
3.4 本章小结 35
第 4 章 基于 NIZK 的 BB+/CL+匿名签名方案 37
4.1 GS 证明系统的 F-可提取性 37
4.2 非交互式 BB+签名方案 37
4.2.1 BB+签名方案 37
4.2.2 改进的 BB+签名方案 38
4.2.3 安全性证明 40
4.3 非交互式 CL+签名方案 43
4.3.1 CL+签名方案 43
4.3.2 改进的 CL+签名方案 44
4.3.3 安全性证明 44
4.4 基于非交互式签名的匿名认证 47
4.5 本章小结 49
第 5 章 基于非交互式签名的匿名电子拍卖协议 51
5.1 电子拍卖的优势及类型 51
5.2 电子拍卖协议及安全性要求 52
5.3 匿名电子拍卖协议描述 53
5.3.1 公开拍卖协议 53
5.3.2 封闭拍卖协议 56
5.4 效率分析 58
5.5 主要算法 59
5.6 本章小结 62
第 6 章 总结与展望 63
参考文献 65
致 谢 69
攻读硕士学位期间的论文项目情况 71

第1章 绪 论

1.1 研究背景

零知识证明在信息安全领域有着很广泛的应用前景，在 Goldwasser 等人提出的零知识证明中，证明者和验证者之间必须进行交互，这种证明叫做交互式零知识证明。零知识证明在于复杂性理论和密码学领域中有着重要意义。在 IP 协议框架中这一主题已经发展为计算复杂性理论的一个重要分支，它提供一种方法，证明出某些问题不是 NP 完全问题。在密码学领域，零知识证明在身份认证、证明协议的完备性方面都发挥了重要作用。传统的零知识证明具备三个特性，“交互性”使得示证者和验证者能够相互对话。“隐藏随机化”使得验证者对于示证者“掷硬币”方法是不能预测的。“计算上的困难性”，示证者在证明中设置了数学难题，使得某一问题在计算上是困难的。

零知识证明可以对某一陈述进行证明，但在零知识证明中，示证者和验证者之间必须进行多次交互，显然对任意NP语言这种证明方法不具有实用的高效性。实际应用中，协议的交互次数应当以一个安全参数的线性函数为界。传统的零知识证明很难使其交互次数以线性多项式为界。尽管对于单轮或常数轮协议的情形交互的次数很少，但需要交互就意味着示证者和验证者都必须同时在线。然而，传统的零知识证明协议都需要双方进行至少三次以上的交互并且不支持用户的离线操作，如果一个ZK证明（或论据）可以做到无需交互，那么参与方就可以用单向方式进行通信。在很多新兴的应用技术中不仅需要进行认证，而且还要求认证是非交互式的。

80 年代末，Blum 等人进一步提出了“非交互零知识证明”的概念，用一个短随机串代替交互过程并实现了零知识证明。示证者和验证者之间共享一个随机串，验证者就能以非交互式，然而却是零知识证明的方式使验证者相信示证者的某一陈述的有效性。事实上“非交互性”主要强调示证者和验证者之间交互的单向性。非交互式零知识证明(NIZK)理论的提出解决了某些实际应用的非交互需求。非交互式零知识证明的一个重要特性就是抗选择密文攻击，利用该特性可以构建抗选择密文攻击的加密、签名等一系列方案。选择密文攻击是所有普通攻击中最强的一种，在这种攻击中敌手试图通过询问和接收他所选择的密文来破译密码体制。对于一个被动敌手来说，如果消息是均匀选择的给定长度的字符串，那么该方案的破译难度相当于因子分解。这种方案经不起选择密文的攻击。而非交互式零知识证明能将这一问题解决。需要给用户发送 y 和 σ 两个串，

其中σ是关于陈述 y 的非交互式零知识证明。脱密设备只能用来输出示证者已经知道的信息，利用“脱密设备”来检验σ是否是使人信服得陈述 y 的证明。如果使人信服则输出 m，否则什么也不输出。事实上，只有在向脱密设备说明示证者知道证据时，脱密设备才输出 m。从而解决选择密文攻击问题。

数字签名技术在现代网络中具有重要作用，可以提供不可否认、抗抵赖等安全需求。传统的数字签名能够用来证实签名者的身份，并且签名具有法律效力，签名的接收者可以对签名进行验证，判断签名是否合法，以此达到认证签名者身份的目的。然而认证的同时意味着签名者身份的泄露，在实际应用中，存在着即认证身份的同时又不泄露身份的需求，这就意味着签名方案与 NIZK 理论的结合，因此本文着重于将非交互式理论应用于签名方案中，在保证匿名性的同时，也能够保证方案的效率。

组签名是一种特殊的签名方案，在一个组签名方案中，组中的任意一个成员以匿名的方式代表整个组对消息进行签名。然而传统的组签名方案的效率及安全性还有待提高，这就意味着 NIZK 理论具有更广的应用范围。

1.2 国内外研究现状

Goldwasser 等人于 1985 年提出了零知识证明[1]的概念，并指出零知识证明系统的特性“隐藏性”，“交互性”。零知识证明是一个两方的协议，示证者能够在不向验证者提供任何秘密信息的情况下，使验证者相信某个陈述是正确的。恶意的示证者不能使验证者相信一个错误的陈述，恶意的验证者不能从交互过程中获得任何额外的秘密信息。在 Goldwasser 等人提出的零知识证明中，示证者和验证者之间必须进行交互，这样的零知识证明被称为“交互零知识证明”。Fiat 等人引入一种方法[2]，将诚的实验证者的安全零知识证明转化为数字签名方案。Fiat-Shamir 启发式可以作为构造非交互式零知识证明的一般方法。但由于 Fiat-Shamir 启发式获得非交互式的代价是失去了零知识性质，“暗中证明”变成了公开验证的证明。Jakobsson 等人设计了一种技术[3]，使用 Fiat-Shamir 启发式却保持了零知识的性质。

非交互式零知识证明（NIZK）已成为密码学协议中的一个重要角色。Blum 等人首次提出了非交互零知识证明[4]的概念，并阐明如何构造一个可计算的非交互式零知识证明系统。在一个 NIZK 系统中，示证者可以以非交互方式使验证者以很高的概率确信一个陈述 $x \in L$ 是正确的，但在证明过程中不泄露任何额外的秘密信息。示证者和验证者之间共享一个随机串，用来帮助示证者计算证明，验证者验证证明。NIZK 证明系统有着广泛的应用，可以应用于抗选择密文攻击的数字签名方案[5]，抗选择密文攻击的公钥密码体制[6, 7]，及密钥分配体制。Danny 等人指出了 NIZK 的两个特性[8]统计零知识性和

计算零知识性的研究情况，统计零知识指有效的仿真器仿真证明副本，并达到用任何统计分辨器都不能区分的精度，计算零知识性强调两个证明的不可区分性。第一个可计算的 NIZK 证明系统[9]由 Blum 等人构建。Feige 等人提出了怎样构造基于陷门排列的可计算 NIZK 证明系统[10]。而对于统计 NIZK 证明一直是个难于实现的目标，DeSantis 等人使用统计 NIZK[11,12]给出了对于 NP 问题复杂性的关联。但他们提出的统计 NIZK 不适用于所有的 NP 问题。事实上在多项式机制上，对于 NP 完全问题很难构建统计零知识证明系统。Jens Groth 等人在 2006 年构建了第一个统计的 NIZK 论据系统[13]，满足电路可满足性，作为一个 UC-安全的论据系统在任何动态敌手的存在下都适用于 NP 语言，解决了 NIZK 协议中的存在的核心打开问题，与之前的证明系统相比使用了较短的公共参考串。

在后续的研究中，文献[14,15,16,17] 说明了对于所有的 NP 语言都存在对应的非交互式零知识证明。然而这些 NIZK 证明系统的效率是非常低的。由于 Dolev 等人引入选择密文攻击[18]，使得上述的 NIZK 系统都不能被实际应用。研究者们通常将 NIZK 证明归约到 NP 问题上，比如电路可满足性问题，一个简单陈述的 NP 归约会产生包含成千上万个门电路，因此 NIZK 证明会很复杂，证明效率也会降低。若想要避免一个复杂的 NP 归约，应该找到一个通用的方法去表达陈述。近年来，双线性映射在密码学领域中取得了巨大成功，研究者们利用双线性映射构建了各类安全协议。如公钥加密、数字签名、密钥协商等协议，双线性理论使得上述协议达到前所未有的安全目标。许多公钥加密协议都基于双线性群，比如 Brent 等人提出的基于身份的加密方案[19,20]，可检索的公钥加密[21,22]，同态加密方案等。

Groth 使用双线性映射构建了基于素数阶群的 NIZK 证明系统[23]，其安全性基于判定性线性假设，系统很通用，可以应用于数字签名，可验证的加密或组签名。并满足 Bellare 等人在标准模型下定义的强安全性[24]。

Jens Groth 提出一种新技术[25]用于构建 NIZK 证明或 NIZK 论据。其安全性依赖于判定性线性假设，该假设可以看做是判定性 Diffie-Hellman 假设的变种。实现了非交互式证据不可区分性。在文献[23][25]的理论基础上，Jens Groth 等人于 2007 年提出了 Groth- Sahai[26]证明系统，该系统给出了一个基于双线性映射的特殊承诺方案和一个证明系统，在这个证明系统中，用户可以产生一系列的承诺方案和 NIZK 证明，必要的时候，使用打开密钥可以将这些承诺打开，求出满足所给配对等式的承诺对象。这个系统是非交互式的，他们首先建立了高效的证据不可区分性的证明系统，然后说明了怎样将证据不可区分性证明系统转换为零知识证明系统。并给出了基于 SDA 假设[27]的实例，基于

DLIN 假设[28]的实例和基于 SXDH 假设[29]的实例。具体证明和三个不同实例的细节在文献[30]中给出。

Chandran 等人使用文献[26]中的子群判定假设 NIWI 证明构建了一个亚线性大小的环签名[31]。Groth 使用 SXDH 假设的 NIWI 和 NIZK 证明构建了一个完全匿名的组签名方案[32]。Boyen 和 Sahai 等人在文献[33,34]探索出怎样直接用双线性群来构建一个有效的 NIZK 证明方法从而取代了将其规约到 NP 问题上，并指出在任意的实例中把 NIZK 证明附属到公钥加密的语义安全上，就可以得到一个高效的非交互式可验证的密码系统。

GS 系统被广泛的应用于组签名方案，匿名投票，匿名证书等方案中，2008 年 Sarah Meiklejohn 利用 GS 中相同承诺对象可以产生不同承诺值的性质构造了一个非交互式零知识协议[35]，该协议满足比 GS 协议更强的零知识性，他利用这种性质作为 NIZK 证明系统的一个新的延伸，当然他构造的新协议也有着广泛的应用，主要应用于构建完全不可否认性的非交互式匿名认证方案。

同样也有人认为 GS 证明系统中存在错误和漏洞，E.Ghada 等人在文献[36]中指出并纠正了 GS 的 NIWI 证明中两个实例的错误，即对于协议的正常执行而言，验证者的验证等式不合理，特别指出了证明系统不能正确执行，并揭示了原因，从 NIWI 系统中构造出的 NIZK 证明也存在上述错误。另外，他们研究了 GS 系统中的三个实例，指出只有一个有实际应用的价值。并在现有的 DLIN 假设基础上做了一些改进，从对称双线性群改为非对称双线性群，从而为 GS 证明系统创造了一个新的有效的配对环境。NIZK 证明系统有着广泛的应用，文献 [31]中的环签名获得了亚线性长度的证明，而通常元素数量和陈述的长度呈线性关系。

NIZK 证明系统作为新兴的技术，已被广泛应用于当今的密码体制中，而如何构建一个高效的 NIZK 证明系统，深入挖掘 NIZK 潜在的性质，并在实际中应用成为本文的主要研究内容。

1.3 本文研究内容

本文首先描述了零知识证明和非交互式零知识证明的基本框架、国内外的研究现状及其分类；介绍了 NIZK 理论中常用的双线性配对理论以及相关数学难题；阐述了 NIZK 证明系统的基本过程及其安全性要求；然后详细介绍了 Groth 等人构建的非交互式证明系统。

本文在基本的签名方案基础上结合 Groth-Sahai 非交互式证明系统，提出了非交互式签名方案的思想。并利用非交互式证明系统构建了非交互式组签名方案 GSCOBG 及

非交互式签名方案 BB+及 CL+。非交互式组签名方案不仅可以实现组签名的基本安全性要求，还可以很好的抵抗选择明文攻击和选择密文攻击，并降低方案代价。大部分已有的使用 NIZK 思想的组签名方案，使用的是低效的 NIZK 证明，有个别组签名方案应用了 GS 证明系统，但结构复杂，效率较低。本文中的组签名基于新提出的 HIBE 加密方案，使用三个素数阶子群判定假设，在安全性方面也有了一定突破。按照组签名方案的安全性要求，本文对该方案进行了详细的安全性分析与形式化证明；证明其不仅满足组签名方案中基本的安全性要求，如签名者的匿名性、可追踪性等特性之外，还具有零知识性，抗联合攻击以及抗选择明文攻击等安全特性。同时，本文将该方案分别与传统的交互式组签名方案和非交互式组签名方案在计算代价和通信代价两方面进行对比。

本文利用非交互式证明系统将传统签名方案加以改进，构建了非交互式 BB+签名方案及非交互式 CL+签名方案，达到了匿名签名的安全性要求。并对其安全性进行了严格的形式化定义和证明，非交互式签名方案作为匿名认证方法的原型，具有较高的应用价值，比如电子商务系统中电子拍卖系统、电子投票系统、可信计算平台等。

将非交互式 BB+签名方案和非交互式 CL+签名方案应用于匿名电子拍卖协议中，分别构建了公开拍卖协议和封闭拍卖协议，与传统的电子拍卖相比，该方案具有较高的匿名性、可追踪性、非交互性，同时降低了通讯代价和存储代价。

1.4 本文组织结构

第 1 章 绪论：本章介绍了相关研究背景、国内外研究现状和本文研究的主要内容。

第 2 章 相关理论基础：本章介绍了双线性配对密码体制，SDA 难题、HSDH 难题、DLA 假设；同态承诺方案、非交互式零知识证明系统的基本构成，以及 Groth-Sahai 证明系统。

第 3 章 基于 NIZK 的组签名方案 GSCOBG：本文利用三素数组合阶双线性群及 HIBE 加密方案构建了一个基于 BMW 模型的高效组签名方案，满足 BMW 模型中定义的完全匿名性和完全可追踪性等安全性要求，并将 NIZK 的理论应用于组签名方案中，与同类方案进行安全性及效率对比。

第 4 章 基于 NIZK 的匿名签名：利用 GS 证明系统的新特性，构建了非交互式 BB+签名方案和非交互式 CL+签名方案，并对其安全性进行了严格的形式化定义和证明。

第 5 章 基于 NIZK 的匿名电子拍卖方案的研究与设计：将非交互式 BB+签名方案和非交互式 CL+签名方案应用于匿名电子拍卖协议中，分别构建了公开拍卖协议和封闭拍卖协议，并对其效率进行分析。

第 6 章 总结与展望：对本文进行总结，介绍了未来的研究方向。

第 2 章 非交互式零知识证明理论基础

本章将介绍双线性配对函数和非交互式假设难题如 SDA、DLIN、以及基于三素数阶群的假设难题。阐述了承诺方案的基本构成，以及同态承诺方案的构成和性质。主要描述了签名方案的构建机制和安全特性。此外还介绍了 NIZK 系统的基本构成和主要特性，并对典型的 NIZK 系统-GS 证明系统的结构及安全性进行阐述。

2.1 数学基础

2.1.1 Co-GDH 群与双线性配对函数

双线性配对是指两个循环群之间相对应的线性映射关系。Co-GDH 群是一个密码学的双线性映射，其建立过程如下：令 G_1 是阶为素数 p 的循环乘法群，生成元为 g；G_2 是阶为 p 的循环乘法群，生成元为 h。φ 是一个可计算的、从 G_1 群到群 G_2 的同构，即 $\varphi(g_1)=g_2$。双线性映射 $e:G_1\times G_2\to G_T$ 可从有限域上椭圆曲线中的 Weil 和 Tate 配对推导得出，该双线性映射具有下列属性：

（1）双线性：存在 $g\in G_1$、$h\in G_2$、$x,y\in Z$，使得等式 $e(g^x,h^y)=e(g,h)^{xy}$ 成立；

（2）非退化性：$e(g,h)$ 是 G_T 的生成元；

这个特性暗示了对任何 $u\in G_1$、$v_1,v_2\in G_2$，都有 $e(u,v_1v_2)=e(u,v_1)e(u,v_2)$；对任一 $u,v\in G_1$，有 $e(u,\varphi(v))=e(v,\varphi(u))$；

（3）可计算性：存在有效算法计算 $t=e(g,h)$；

当 $G_1=G_2=G$ 时，生成元 $g=h$，双线性映射就变为 $e:G\times G\to G_T$。Boneh 和 Franklin 在文献[45]中给出了上述双线性群的实例。令素数 $p=2\bmod 3$，选择一个 l 使得素数 $q=lp-1$ 并且 $p^2\neq q+1$。有限域 F_q 上的椭圆曲线上 $y^2=x^3+1$ 有 lp 个点。令该椭圆曲线上阶为 p 的子群为 G，并且 $G_T=F_{q^2}^*$。这样威尔配对就被改进为 $e:G\times G\to G_T$。从群 G_T 中随机选择一个生成元 g，随机选择 x 使得 x^3+1 是平方项，y 是平方项的底，则当 $g\neq 1$ 时 $g=(x,y)^l$ 是 G 的生成元。

2.1.2 组合阶双线性群

组合阶双线性群在文献[37]中被首次提出。2010 年 Allison Lewko 等人使用了群生成器 $\mathcal{G}$ 和一个参数生成算法重新定义了组合阶双线性群[38]，其中参数生成算法以安全参数 λ 作为输入，以双线性群 G 的描述作为输出。在他们的定义中 $\mathcal{G}$ 输出参数 $(N=p_1p_2p_3,G,G_T,e)$ 其中 p_1,p_2,p_3 是互不相同的素数，G 和 G_T 为阶为 $N=p_1p_2p_3$ 的循

环群，映射$e:G^2\to G_T$的性质描述如下：

（1）双线性：$\forall g,h\in G,a,b\in\mathbb{Z}_N,e(g^a,h^b)=e(g,h)^{ab}$

（2）非退化性：$\exists g\in G$，g 是群 G 的生成元，则$e(g,g)$是G_T的生成元

群 G 是一个阶为$N=p_1p_2p_3$的循环群，$e:G^2\to G_T$是一个双线性映射，所选的元素是对应群中的生成元。在三素数的组合阶双线性群中，令群G_{p1},G_{p2}和G_{p3}为群 G 的子群，阶分别为p_1,p_2和p_3。元素 g 是群 G 的一个生成元。因此元素$g^{p_1p_2}$是群G_{p_3}的生成元，元素$g^{p_1p_3}$是群G_{p_2}的生成元，元素$g^{p_2p_3}$是群G_{p_1}的生成元。假设$h_1\in G_{p_1}$，$h_2\in G_{p_2}$，因此，对于$\alpha_1,\alpha_2,h_1=(g^{p_2p_3})^{\alpha_1}$和$h_2=(g^{p_1p_3})^{\alpha_2}$，可以推导出$e(h_1,h_2)=e(g^{p_2p_3\alpha_1},g^{p_1p_3\alpha_2})=e(g^{\alpha_1},g^{p_3\alpha_2})^{p_1p_2p_3}=1$.

2.1.3 安全性假设难题

（1）子群判定性假设难题（SDA）

已知$n=pq$，当$r\leftarrow Z_n^*$时计算的$h=g^r$，与$r\leftarrow Z_q^*$时计算的$h=g^{pr}$在多项式时间内不可区分。形式化表述如下，其中 Pr 代表概率，A 代表敌手。

$$\Pr[(p,q,G,G_T,e,g)\leftarrow\text{BilinearSetup}(1^k);n=pq;r\leftarrow Z_n^*;h=g^r:A(n,G,G_T,e,g,h)=1]$$
$$\approx\Pr[(p,q,G,G_T,e,g)\leftarrow\text{BilinearSetup}(1^k);n=pq;r\leftarrow Z_q^*;h=g^{pr}:A(n,G,G_T,e,g,h)=1]\quad(2.1)$$

组合阶双线性群中的假设：

（2）三素数的组合阶双线性群中的子群判定假设：给定一个群的生成器$\mathcal{G}$，定义下列关系：

$$\mathbb{G}=(N=p_1p_2p_3,G,G_T,e)\xleftarrow{R}\mathcal{G},g\xleftarrow{R}G_{p_1},X_3\xleftarrow{R}G_{p_3},$$
$$D=(\mathbb{G},g,X_3),T_1\xleftarrow{R}G_{p_1p_2},T_2\xleftarrow{R}G_{p_1}\quad(2.2)$$

定义敌手$\mathcal{A}$攻破假设（2）算法优势为：

$$Adv1_{\mathcal{G},\mathcal{A}}(\lambda):=\left|\Pr[\mathcal{A}(D,T_1)=1]-\Pr[\mathcal{A}(D,T_2)=1]\right|.\quad(2.3)$$

定义 2.1 如果对于任一个多项式时间算法$\mathcal{A}$，$Adv1_{\mathcal{G},\mathcal{A}}(\lambda)$是一个关于$\lambda$可忽略函数，则生成器满足假设（2）。

（3）给定一个群生成器 $\mathcal{G}$，定义下列关系：

$$\mathbb{G}=(N=p_1p_2p_3,G,G_T,e)\xleftarrow{R}\mathcal{G},g,X_1\xleftarrow{R}G_{p_1},X_2,Y_2\xleftarrow{R}G_{p_2},X_3,Y_3\xleftarrow{R}G_{p_3},$$
$$D=(\mathbb{G},g,X_1X_2,X_3,Y_2Y_3),T_1\xleftarrow{R}G,T_2\xleftarrow{R}G_{p_1p_3}.\quad(2.4)$$

定义敌手$\mathcal{A}$攻破假设（3）的算法优势为：

$$Adv2_{\mathcal{G},\mathcal{A}}(\lambda):=\left|\Pr[\mathcal{A}(D,T_1)=1]-\Pr[\mathcal{A}(D,T_2)=1]\right|.\quad(2.5)$$

定义 2.2 如果对于任一个多项式时间算法$\mathcal{A}$，$Adv1_{\mathcal{G},\mathcal{A}}(\lambda)$是一个关于$\lambda$可忽略函数，则生成器满足假设（3）。

（4）给定一个群生成器$\mathcal{G}$，定义下列关系：

$$\mathbb{G}=(N=p_1p_2p_3,G,G_T,e)\xleftarrow{R}\mathcal{G},\alpha,s\xleftarrow{R}\mathbb{Z}_N,g\xleftarrow{R}G_{p_1},X_2,Y_2,Z_2\xleftarrow{R}G_{p_2},$$
$$X_3\xleftarrow{R}G_{p_3},D=(\mathbb{G},g,g^{\alpha}X_2,X_3,g^{s}Y_2,Z_2),T_1=e(g,g)^{\alpha s},T_2\xleftarrow{R}G_T. \quad (2.6)$$

定义敌手$\mathcal{A}$攻破假设（4）算法优势为：

$$Adv3_{\mathcal{G},\mathcal{A}}(\lambda):=\left|\Pr[\mathcal{A}(D,T_1)=1]-\Pr[\mathcal{A}(D,T_2)=1]\right|. \quad (2.7)$$

定义 2.3 如果对于任一个多项式时间算法$\mathcal{A}$，$Adv1_{\mathcal{G},\mathcal{A}}(\lambda)$是一个关于$\lambda$可忽略函数，则生成器满足假设（4）。

（5）判定性线性假设（DLA）

已知$u,v,w,u^r,v^s\in G_1$，很难区分开$z_0\leftarrow w^{r+s}$与$z_1\leftarrow G_1$。这个假设在G_2中也存在。形式化表述为：

$$\Pr[(p,G_1,G_2,e,g,h)\leftarrow BilinearSetup(1^k);r,s\leftarrow G_p;u,v,w\leftarrow G_1;b\leftarrow\{0,1\};$$
$$z_0\leftarrow w^{r+s};z_1\leftarrow G_1:A(p,G_1,G_2,G_T,e,g,h,u,v,w,u^r,v^s,z_b)=b]<1/2+v(k) \quad (2.8)$$

（6）计算性 Diffie-Hellman 难题（CDH）

已知$g^a\in G_1$和$g^b\in G_1$，其中$a,b\leftarrow Z_q^*$。想在多项式时间内计算出g^{ab}是很困难的事情，形式化的表示为，存在可忽略的函数v使得

$$\Pr[a\leftarrow Z_q^*,b\leftarrow Z_q^*,g^a,g^b:A(g,g^a,g^b)=g^{ab}]<v(k) \quad (2.9)$$

（7）强 Diffie-Hellman 难题（SDH）

若已知$(l+1)$个元素$(g,g^x,g^{x^2},...,g^{x^l})$，其中$x\in Z_p^*$为随机数。则敌手在多项式时间内能够计算出$(c,g^{1/(x+c)})\in Z_p^*\times G_p$的概率是可忽略的。形式化表述为：

$$\Pr[(p,G,G_T,e,g)\leftarrow BilinearSetup(1^k);x\leftarrow Z_p^*,c\leftarrow Z_p^*;$$
$$(A,B)\leftarrow A(p,G,G_T,e,g,g^x,g^{x^2},...,g^{x^l}):(A,B)=(c,g^{1/(x+c)})]<v(k) \quad (2.10)$$

（8）隐藏性强 Diffie-Hellman 难题（HSDH）

已知$g,g^x,u\in G_1$、$h,h^x\in G_2$和$\{g^{1/(x+c_l)},h^{c_l},u^{c_l}\}_{l=1...q}$，敌手很难在多项式时间内计算出$(g^{1/(x+c)},h^c,u^c)$，即存在可忽略的函数$v$使得

$$\Pr[(p,G_1,G_2,G_T,e,g,h)\leftarrow BilinearSetup(1^k);u\leftarrow G_1;x,\{c_l\}_{l=1...q}\leftarrow Z_p;$$
$$(A,B,C)\leftarrow A(p,G_1,G_2,G_T,e,g,g^x,h,h^x,u,\{g^{1/(x+c_l)},h^{c_l},u^{c_l}\}_{l=1...q}):$$
$$(A,B,C)=(g^{1/(x+c)},h^c,u^c)\wedge c\notin\{c_l\}_{l=1...q}]<v(k) \quad (2.11)$$

若给出元组$(A,\ B,\ C)$可以通过验证等式$e(A,BH)=e(g,h)$和$e(u,B)=e(C,h)$来判断其是否是 HSDH 元组。

（9）交互式隐藏 Diffie-Hellman 难题（IHSDH）

若已知(g,g^x,h,h^x,u)，则敌手不能在多项式时间内计算出$(g^{1/(x+c)},h^c,u^c)$。其中敌

手可以对预言 $O_x(c)$ 做出 q 次询问，返回值 $g^{1/(x+c)}$，并且敌手每一次询问时使用的 c 都不重复。形式化的表述如下，存在可忽略的函数 v 使得

$$\Pr[(p,G_1,G_2,G_T,e,g,h) \leftarrow BilinearSetup(1^k); x \leftarrow Z_p; u \leftarrow G_1;$$
$$(A,B,C) \leftarrow A^{O_x(\cdot)}(p,G_1,G_2,G_T,e,g,g^x,h,h^x,u): \exists c:(A,B,C)=(g^{1/(x+c)},h^c,u^c)] < v(k) \quad (2.12)$$

（10）k 次 Diffie-Hellman 难题（k-SDH）

已知 $\{g,g^x,g^{x^2},...,g^{x^k}\}$，其中 $x \leftarrow Z_q^*$。若存在敌手随机选择 $c \leftarrow Z_q^*$，要想在多项式时间内计算出 $g^{\frac{1}{c+x}}$ 是很困难的，即存在可忽略的函数 v 使得下式成立：

$$\Pr[x \leftarrow Z_q^*, g,g^x,g^{x^2},...,g^{x^k}: A(g,g^x,g^{x^2},...,g^{x^k},c)=g^{\frac{1}{c+x}}, c \leftarrow Z_q^*] < v(k) \quad (2.13)$$

（11）三方 Diffie-Hellman 难题（T-DH）

已知 $\{g,g^a,g^{a^2},...,g^{a^l},g^{ak},g^{a^2k},...,g^{a^lk}\}$，其中 $a,k \leftarrow Z_q^*$。若存在敌手随机选择 $r \leftarrow Z_q^*$，要想在多项式时间内计算出 $g^{a^{l+1}k} \cdot g^r$ 和 g^{ar} 是一件很困难的事情，即存在可忽略的函数 v 使得下式成立：

$$\Pr[a,k \leftarrow Z_q^*, g,g^a,g^{a^2},...,g^{a^l},g^{ak},g^{a^2k},...,g^{a^lk}:$$
$$A(g,g^a,g^{a^2},...,g^{a^l},g^{ak},g^{a^2k},...,g^{a^lk},r)=g^{a^{l+1}k}\cdot g^r \Lambda g^{ar}, r \leftarrow Z_q^*] < v(k) \quad (2.14)$$

2.2 承诺方案

承诺是隐藏签名者私有信息而使用的一种方法。在一个非交互式的承诺方案中，用户可以选择随机数用某一承诺算法对承诺对象进行承诺。承诺对象和随机数叫做承诺的打开值，承诺对象和打开值是一一对应的关系，用户将承诺发给验证者，验证者可以将打开值输入验证算法，从而验证承诺的正确性。

目前最常用的承诺方案是 Pedersen 承诺方案[39]，该方案中公钥由两个元素 $g,h \in G$ 组成，其中 G 的阶为素数 q。若希望对消息 $m \in Z_q$ 进行承诺，则选择随机数 $r \leftarrow Z_q$ 并计算 $c=g^m h^r$。离散对数问题保证了该方案在计算上具有隐藏性。

后来 Pedersen 承诺方案被扩展成对多个值进行承诺。令 G 是阶为素数 p 的循环群，$g,g_1,...,g_n$ 是 G 的随机生成元。若要对消息 $a_1,...,a_n$ 进行承诺，则选择随机数 $r \leftarrow Z_p$ 计算 $c=g^r \prod_{i\in[n]} g_i^{a_i}$，则称 $(r,a_1,...,a_n)$ 是 c 的提取信息。这个承诺方案基于离散对数问题在计算上具有完全隐藏性，很难找到关于承诺值 c 的两个不同的提取信息。并且该方案具有很好的隐藏性，无论选择什么样的 $a_1,...,a_n$ 和随机数 r 都可保证承诺值是在群中均匀分布的元素。

2.2.1 同态承诺方案

在非交互式零知识证明领域中应用的特殊承诺方案称为同态证明承诺。一个非交互

式承诺方案一般包括一个密钥生成器，用来产生承诺时所使用的密钥ck，承诺密钥ck定义了消息空间M_{ck}，随机数空间R_{ck}和承诺空间C_{ck}。对于一个高效的承诺算法$c=com(m;r)$，输入承诺密钥和随机数，能够输出承诺值c。而$(m;r)$称之为c的打开值。

承诺方案必须具备两个性质：绑定性和隐藏性。绑定性意味着对于两个不同消息，不同打开值产生相同的承诺值的概率是可忽略的。隐藏性意味着给定一个承诺值，从中得出被承诺的消息的概率是可忽略的。承诺之所以有绑定性和隐藏性，归功于绑定密钥和隐藏密钥。使用隐藏密钥和陷门信息可以将承诺值打开成任意消息。使用绑定密钥可以从承诺中提取出消息值。而隐藏密钥和绑定密钥在计算上是不可区分的。

消息空间$(M,+,0)$，随机数空间$(R,+,0)$，承诺空间$(C,\cdot,1)$是有限的阿贝尔群。同态承诺方案满足同态性：$com(m_1+m_2;r_1+r_2)=com(m_1;r_1)com(m_2;r_2)$

同态承诺具有下列性质：

（1）密钥的不可区分性：在同态承诺中绑定密钥和隐藏密钥是不可区分的。形式化表述如下：

$$\Pr[(ck,xk)\leftarrow K_{hinding}(1^k):\mathcal{A}(ck)=1]\approx\Pr[(ck,tk)\leftarrow K_{hinding}(1^k):\mathcal{A}(ck)=1] \quad (2.15)$$

（2）同态性：同态性相对于群的性质而言，两个承诺值相互运算，可以得到第三个承诺值，形式化表述如下：

$$\Pr[mode\leftarrow\{binding,hiding\};(ck,*)\leftarrow K_{mode}:\forall(m_1,r_1),(m_2,r_2)\in M\times R:$$
$$com(m_1+m_2;r_1+r_2)=com(m_1,r_1)com(m_2,r_2)]=1 \quad (2.16)$$

（3）绑定性：若签名者构建并发送了承诺值$comm$，则有且仅有一个x与其对应。若使用随机数$open$来打开承诺值$comm$，则最多存在一个承诺对象x，使得$comm=\mathrm{Commit}(x,open)$成立，即$x$与$comm$是一一绑定的。若存在敌手$A$，该性质可形式化描述为：

$$\Pr[(ck,xk)\leftarrow K_{binding}(1^k):\exists(m_1,r_1),(m_2,r_2)\in M\times R:m_1\neq m_2$$
$$\wedge com(m_1,r_1)=com(m_2,r_2)]=0 \quad (2.17)$$

（4）隐藏性：验证者只知道承诺值$comm$，而并不知道签名者是对哪个承诺对象x进行的承诺，即私有信息x被隐藏。

$$\Pr[(ck,tk)\leftarrow K_{hinding}(1^k);(m_1,m_2)\leftarrow\mathcal{A}(ck);r_1\leftarrow R;r_2\leftarrow Topen(m_1,r_1,m_2):$$
$$com(m_1,r_1)=com(m_2,r_2)\wedge m_1,m_2\in M]=1 \quad (2.18)$$

2.2.2 数字签名

一个数字签名体制由以下部分组成，一个明文消息空间M代表某字母表中串的集合；一个签名空间S表示可能的签名集合；一个签名密钥空间K代表用于生成签名的可

能密钥集合，一个认证密钥空间 K' 代表用于验证签名的可能密钥集合；一个有效的密钥生成算法 $\mathrm{Gen}: N \mapsto K \times K'$，其中 K 和 K' 分别为私钥和公钥空间；一个有效的签名算法 $\mathrm{Sign}: M \times K \mapsto S$；一个有效的验证算法 $\mathrm{Verify}: M \times S \times K' \mapsto \{\mathrm{True}, \mathrm{False}\}$。

对任意 $sk \in K$ 和任意的 $m \in M$，用 $s \leftarrow \mathrm{Sign}_{sk}(m)$ 来表示 s 是使用密钥 sk 对 m 进行的签名；对于任意的私钥 $sk \in K$，用 pk 表示与 sk 相匹配的公钥；对于 $m \in M$ 和 $s \in S$，必有 $\mathrm{Verify}_{pk}(m,s) = \mathrm{True} \| \mathrm{False}$。

密钥产生算法 Gen 的输入整数 k 规定了输出签名/验证密钥的规模长度。若密钥生成算法有效，则其运行时间必须为输入长度规模的多项式时间；输入的整数值应该是一元编码的，即这个整数的长度就是这个数本身。则这个整数 k 就是签名体制的安全参数，它定义了签名空间的大小。

安全的数字签名方案是指满足如下所述的正确性和不可伪造性。

（1）正确性：使用 Sign 算法的签名都应该被 VerifySig 算法所接受。形式化表述为：

$$\forall m \in \{0,1\}^{*} : \Pr[params_{Sig} \leftarrow SigSetup(1^{k}); (pk, sk) \leftarrow Keygen(params_{Sig}); \delta \leftarrow Sign(params_{Sig}, sk, m) : VerifySig(params_{Sig}, pk, m, \delta) = 1] = 1 \tag{2.19}$$

（2）不可伪造性：若敌手事先不知道 m 上的签名，则他输出合法的 (m, δ) 的可能性可以忽略。形式化的表述为：

$$\Pr[params_{Sig} \leftarrow SigSetup(1^{k}); (pk, sk) \leftarrow Keygen(params_{Sig}); (Q_{Sign}, m, \delta) \leftarrow \mathrm{A}(params_{Sig}, pk)^{O_{Sign}(params_{Sig}, sk, \cdot)} : VerifySig(params_{Sig}, pk, m, \delta) = 1 \wedge m \notin Q_{Sign}] < v(k) \tag{2.20}$$

2.4 非交互式零知识证明系统

零知识证明可以对某一陈述进行证明，但在零知识证明中，示证者和验证者之间必须进行多次交互，显然对任意NP语言的这种证明方法不具有实用的高效性。实际应用中，协议的交互次数应当以一个安全参数的线性函数为界。传统的零知识证明很难使其交互次数以线性多项式为界。Blum等人进一步提出了非交互零知识证明的概念，用一个短随机串代替交互过程并实现了零知识证明。但提出的证明系统效率太低，没有应用价值。本节将详细介绍非交互零知识证明系统。

2.4.1 零知识证明

零知识证明在信息安全中处于极为重要的基础性地位。自从它诞生以来特别是Goldreich等人证明任何NP语言都有一个关于成员问题的计算零知识证明以后，零知识证明便成了一个有力工具，它为多方安全计算这一几乎所有密码学任务的通用解决方案提

供了关键工具。另外，零知识协议还广泛应用于大量特定的安全协议的设计中，如身份认证、电子现金、电子投票、组签名等。所谓零知识证明，指的是示证者在证明自己身份时不泄露任何信息，验证者得不到示证者的任何私有信息，但又能有效证明对方身份的一种方法。从本质上讲，零知识证明是一种协议。所谓协议（Protocol），就是两个或两个以上的参与者为完成某项特定的任务而采取的一系列步骤，包括以下三个特征：协议自始至终是有序的过程，每一步执行必须是一次执行，在前一步没有执行完之前，后面的步骤不可能执行；协议至少需要两个参与者，一个人可以通过执行一系列的步骤完成某项任务，但它不构成协议；通过执行协议必须能够完成某项任务。

Quisquater等人描述了“阿里巴巴的故事”用来说明零知识证明的基本概念。有一个洞穴，如图3.1所示，这个洞穴只有一个入口（*a*的位置），进入洞以后（*b*的位置）有左右两道条通往洞穴底部，底部有个神奇的门，只有听到通关咒语“芝麻开门”的时候门才会打开。阿里巴巴知道这个秘密，他想向别人证明却又不想泄露秘密的咒语，他的做法如下：

（1）验证者站在*a*处等候（看不见*b*处）；

（2）阿里巴巴由*b*处走进洞穴（从左边或者右边进入）；

（3）验证者走到*b*处，并随机地要求阿里巴巴从左边或者右边的通道出来；

（4）步骤（1）到（3）的动作重复*n*次。

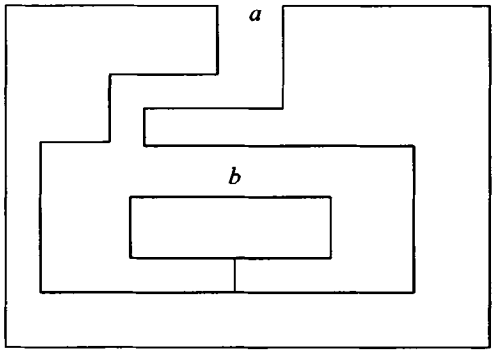

图2.1 零知识证明的故事
Fig. 2.1 The story about Zero-knowledge

在步骤（3）中，阿里巴巴有1/2的概率需要用咒语打开门，才能满足验证者的需求，如果阿里巴巴不知道通关的咒语，每次只有1/2的概率可以在步骤（2）选择正确的一边进入洞穴，不需要使用咒语就能按照验证者的要求出来，因此，如果阿里巴巴不知道通关咒语，他只有$1/2^n$的概率可以骗过验证者，当*n*足够大时，验证者被欺骗的概率为0，验证者便可以相信阿里巴巴知道打开门的咒语。

一个完整的交互式零知识证明包含承诺、挑战、和回应三个步骤，对应上述故事中

的“阿里巴巴生成自己有门的钥匙”、“验证者随机要求阿里巴巴从某一边走出来”、“阿里巴巴按照要求走出来”。而且零知识证明必须满足下列性质：

（1）完备性：若示证者真的知道秘密，则能说服验证者相信自己知道这个秘密。

（2）健壮性：若示证者不知道秘密，则有很小的概率说服验证者相信他知道这个秘密。

（3）零知识性：任何人不能从证明过程中获得任何和秘密相关的消息。

Fiat 和 Shamir 将零知识证明转成数字签名的方式，将零知识中承诺值与消息生成摘要当作挑战值，称为 Fiat-Shamir 启发式。

2.4.2 非交互式零知识

80 年代末，Blum 等人进一步提出了“非交互零知识证明”的概念，用一个短随机串代替交互过程并实现了零知识证明。示证者和验证者之间共享一个随机串，验证者就能以非交互式，然而却是零知识证明的方式使验证者相信示证者的某一陈述的有效性。事实上“非交互性”主要强调示证者和验证者之间交互的单向性。例如 P 和 V 都是数学家，前者正在周游世界，在这期间发现了新的数学定理证明，他想用 ZK 证明向后者示证这些新定理。在这种场合中的非交互式零知识（NIZK）证明就是必要的，因为 P 很可能没有固定的地址，在任何邮件到达之前就离开了。

从交互到非交互是有代价的，在证明中需要引入一个可信的第三方。读者容易验证对于非平凡语言，在没有交互和可信第三方的情况下则无法构造一个零知识证明系统，在现实世界中，为了提高效率或达到其他目的，在安全系统中通常假定了可信第三方，如 PKI。这就使得非交互零知识协议有着非常广阔的应用前景，一方面它不需要交互，另一方面，在现实中引入可信第三方是可行的。

非交互式模型中共有两个必要成分，即公共参考串和计算困难性。公共参考串是由可信第三方提供的，示证者和验证者都能够读取公共参考串，这种证明非常简单，它就是示证者发送给验证者的一次消息，然后验证者根据这条消息去决定是否可接受。

2.4.2.1 NIZK 的构成

本节中主要讨论在公共参考串模型下的非交互式证明系统，在 NIZK 证明中经常使用参考串的概念。通用参考串 CRS 是指具有绑定性承诺值的公钥，而模拟 CRS 是指具有隐藏性承诺值的公钥。基于具体不同的难题，通用参考串与模拟参考串也有具体的形式。

非交互式证明系统中包含三个多项式时间算法 *Setup* 算法、*Prove* 算法和 *VerifyProof* 算法。其中 *Setup* 算法产生一系列的公共参数，供 *Prove* 算法和 *VerifyProof* 算法使用。

示证者使用 *Prove* 算法造证明π，验证者通过验证算法 *VerifyProof* 验证证明π的正确性。

示证者想要证明陈述$x \in L$（其中 L 是 NP 语言）的正确性。也就是说，在给定 x 和多项式时间图灵机算法 M_L，示证者想要证明陈述：$\exists w$，$M_L(x,w)=1$成立。多项式时间算法描述如下：

Setup(1^k)：输入安全参数 K，系统产生公共参数 *params*。如果参数是从$\{0,1\}^{l(k)}$里随机选择的，称为公共随机串模型，如果不是完全随机选择的，则称为公共参考串模型。

Prove(*params*，*x*，*w*，M_L)：该算法将公共参数 *params*、陈述 x 和证据 w 作为输入，其中陈述 x 和证据 w 满足图灵机算法 $M_L(x,w)=1$，算法根据所满足条件给出证明π。

VerifyProof(*params*，*x*，π)：该算法将公共参数 *params*、陈述 x 和证明π作为算法的输入，如果证明π成立，输出 *yes*，反之，输出 *no*。

2.4.2.2 NIZK 的安全性

对于上述 NIZK 证明系统满足下列性质：

（1）正确性：如果诚实的示证者给出的真实的证明，验证者都能接受，那么该证明系统满足正确性。

定义 2.4. 如果一个非交互式零知识证明系统（*Setup*，*Prove*，*VerifyProof*）对于所有的语言 L 是完备的，则对于所有满足 $M_L(x,w)=1$ 的 x、w 有下式成立：

$$\Pr[params \leftarrow Setup(1^k); \pi \leftarrow \Pr ove(params,x,w,M_L): Verify\Pr oof(params,x,\pi,M_L)=1]=1 \quad (2.21)$$

（2）健壮性：指对于任何陈述$x \notin L$，多项式时间敌手 $\mathcal{A}$ 伪造的证据不被诚实的验证者接受。

定义 2.5. 一个非交互式零知识证明系统（*Setup*，*Prove*，*VerifyProof*）对于语言 L 是完备的则需满足下列条件：

$$\Pr[params \leftarrow Setup(1^k); (\pi,x,L) \leftarrow \mathcal{A}(params,x): x \notin L \wedge Verify\Pr oof(params,x,\pi,M_L)=1] \le v(k) \quad (2.22)$$

其中 $\mathcal{A}$ 是多项式时间敌手，M_L 是图灵算法，当且仅当$x \in L$时，图灵算法 M_L 接受 (x,w)。

定义 2.6. 对于一个非交互式零知识证明系统（*Setup*，*Prove*，*VerifyProof*）如果对于所有的多项式时间敌手使得 $v(k)=0$，那么该非交互式证明系统（*Setup*，*Prove*，*VerifyProof*）有完美的健壮性。

（3）证据不可区分性：如果关于某个陈述 x 存在两个论据 *w0*、*w1*，敌手不能分辨出示证者是使用哪个论据给出的证明。

定义 **2.7.** 对于所有的多项式时间敌手 $\mathcal{A}_1$，$\mathcal{A}_2$，存在一个可忽略的函数 v，同时满足下式，那么该非交互式证明系统(*Setup*，*Prove*，*VerifyProof*)对于所有语言 L 具有证据不可区分性。

$$\begin{aligned}&\Pr[params \leftarrow Setup(1^k);(x,w_1,w_2,state) \leftarrow A_1(params,x,M_L);b \leftarrow \{0,1\};\\ &\pi \leftarrow \Pr ove(params,x,w_b,M_L);b' \leftarrow A_2(state,\pi): M_L(x,w_1)=M_L(x,w_2)=1 \wedge b=b']\\ &=\frac{1}{2}+v(k)\end{aligned} \tag{2.23}$$

定义 **2.8.** 如果对于所有的敌手使得 $v(k)=0$，那么该证明系统具有完美的证据不可区分性。

（4）零知识性：验证者除了知道陈述是正确的以外，不知道其他任何信息。若一个诚实的示证者使用模拟器在没有证据 w 的情况下伪造产生一个证明，形如 $x \in L$，则它与真正的证明过程不可区分。其中模拟器必须有能力让敌手不能产生类似的证明。证明中存在两种系统，一种是零知识证明其中公共参考串仅仅被使用在单独的证明中，还有一种是相同的公共参考串被用在多个证明中。

定义 **2.9.** 如果存在 *PPT* 算法满足下列等式，那么称该非交互式系统（*Setup*，*Prove*，*VerifyProof*）是单一的零知识证明系统，且对于所有的语言 L 都成立。

对于所有的 $x \in L$ 和证据 w 都有 $M_L(x,w)=1$，以及所有的敌手 $\mathcal{A}$，下面两种算法是不可区分的：

$$\begin{aligned}&\mathrm{Re}al(k)=\{params \leftarrow Setup(1^k);\pi \leftarrow \Pr ove(params,x,w,M_L):(params,\pi)\}\\ &Sim(k)=\{(params,\pi) \leftarrow Sim\Pr oveOne(1^k,x,M_L):(params,\pi)\}\\ &b \leftarrow \mathcal{A}^{\Pr ove(params,)}\end{aligned} \tag{2.24}$$

定义 **2.10.** 如果存在 *PPT* 模拟器算法 *SimSetup*，*SimProve* 满足下式，那么该非交互式证明系统（*Setup*，*Prove*，*VerifyProof*）是一个多重的零知识证明系统，且对于所有的语言 L 都成立。对于所有 *PPT* 敌手 $\mathcal{A}$，存在可忽略的函数 v:

$$\begin{aligned}&|\Pr[params \leftarrow Setup(1^k);b \leftarrow A^{\Pr ove(params,,,M_L)}:b=1]-\Pr[(params,s) \leftarrow Setup(1^k);\\ &b \leftarrow A^{o_{Sim}(s,params,,,,M_L)}:b=1]|<v(k)\end{aligned} \tag{2.25}$$

（5）组合零知识性：首先系统模拟一些参数，这些参数和真实的参数不可区分。然后考虑诚实的示证者和模拟的示证者使用相同的模拟参数，当敌手获得模拟的陷门参数时，模拟的证明和真实的证明不可区分。

定义 **2.11.** 如果存在 *PPT* 模拟器算法 *SimSetup*，*SimProve* 满足下式，那么该非交互式证明系统（*Setup*，*Prove*，*VerifyProof*）是一个组合的零知识证明系统，且对于所有的语言 L 都成立。以下两种函数不可区分：

$$\mathrm{Re}al_{params}(k)=\{params \leftarrow Setup(1^k);params\}$$

$$Sim_{params}(k)=\{(params,s)\leftarrow SimSetup(1^k):params\} \quad (2.26)$$

对于所有的 $x\in L$ 以及有效的证据 w，以下两种函数不可区分：

$$R_{proof}(k)=\{(params,s)\leftarrow SimSetup(1^k);\pi\leftarrow Prove(params,x,w,M_L):\pi\}$$

$$Sim_{proof}(k)=\{(params,s)\leftarrow SimSetup(1^k);\pi\leftarrow SimProve(params,x,w,M_L):\pi\} \quad (2.27)$$

（6）可提取性：若给出一个已被接受的某陈述的证明并且给定一些陷门信息，则存在一个 *PPT* 抽取算法可以从陈述中提出正确的证据 w。

定义 2.12. 如果存在多项式时间抽取器（*ExtractSetup*，*Extract*）满足下式，那么该非交互式证明系统（*Setup*，*Prove*，*VerifyProof*）对于语言 L 具有可提取性。

提取参数 $\{params\leftarrow Setup(1^k):params\}\approx\{(params,t)\leftarrow ExtractSetup:params\}$，对于所有的 *PPT* 敌手 A，存在可忽略的函数 v 满足下列等式：

$$\Pr[(params,t)\leftarrow ExtractSetup;(x,\pi)\leftarrow A(params);w\leftarrow Extract(params,t,x,\pi)$$
$$:VerifyProof(params,x,\pi,M_L)=1\wedge M_L(x,w)=0]=v(k) \quad (2.28)$$

其中 $v(k)=0$，则该证明系统具有可提取性。

2.5 Groth-Sahai 证明系统

Groth 和 Sahai 于 07 年构建了一个高效的非交互式证明系统，用于证明某一承诺值满足特定的代数关系，这种代数关系称之为配对乘积等式。他们给出了一个 NIWI/NIZK 系统结构的抽象框架，同时给出了证明系统的三个实例，第一个实例使用组合阶双线性群，另外两个实例使用素数阶群，分别使用子群判定假设 SDA，判定性线性假设 DLIN 和 SXDH 假设。Groth 等人使用通用参考串模型，给出关于陈述的非交互式证据不可区分证明和非交互式零知识证明方法。

GS 证明系统基于双线性参数 (p,G_1,G_2,G_T,e,g,h)，G_1,G_2,G_T 是阶为 p 的素数阶群，g 是 G_1 的生成元，h 是 G_2 的生成元，映射 $e:G_1\times G_2\rightarrow G_T$ 是双线性映射，陈述 s 包含下述一系列值：$\{a_q\}_{q=1...Q}\leftarrow G_1$，$\{b_q\}_{q=1...Q}\leftarrow G_2$，$t\in G_T$，$\{\alpha_{q,m}\}_{q=1...Q,m=1...M}\leftarrow Z_p$，$\{\beta_{q,n}\}_{q=1...Q,n=1...N}\leftarrow Z_p$，以及群 G_1 上的承诺值 $\{C_m=a_q\prod_{m=1}^{M}x_m^{\alpha_{q,m}}\}_{m=1...M}$ 和群 G_2 上的承诺为 $\{D_n=b_q\prod_{n=1}^{N}y_n^{\beta_{q,n}}\}_{n=1...N}$，其中 C_m 是 $\{x_m\}_{m=1...M}\in G_1$ 的承诺，D_n 是 $\{y_n\}_{n=1...N}\in G_2$ 的承诺。承诺值满足下列等式 $\prod_{q=1}^{Q}e(a_q\prod_{m=1}^{M}x_m^{\alpha_{q,m}},b_q\prod_{n=1}^{N}y_n^{\beta_{q,n}})=t$。下面详细讨论 GS 系统的建立过程。

2.5.1 GS 证明系统的建立过程

令 $(R,+,\cdot,0,1)$ 为可交换环，存在模 R 的一个阿贝尔群 $(M,\cdot,1)$，对所有的 $\forall r,s\in R$、$\forall u,v\in M$，都满足 $u^{r+s}=u^ru^s\Lambda(uv)^r=u^rv^r$。GS 系统中包含五个多项式时间内的算法

(Setup,GSCommit,GSProve,GSVerify,GSExtract)，分别为初始化算法、承诺算法、证明算法和提取算法。

（1）Setup：产生公共参数包括加法群 M_1、M_2 和二者之间双线性映射 E。与此同时，需要选择 I 个元素 $u_i \in M_1$、J 个元素 $v_j \in M_2$。另外在某些实例中需要一个元素的集合 $\{\eta_h\}$。最终产生的参数包括 M_1、M_2、E、$\{u_i\}$、$\{v_j\}$、$\{\eta_h\}$。如果存在双线性映射 $e:G_1\times G_2\to G_T$，通过进行承诺可将元素 $x\in G_1$、$y\in G_2$ 进行隐藏，得到相应的承诺值 $c\in M_1$、$d\in M_1$；并且存在相应的双线性映射 $E:M_1\times M_2\to M_T$，这个映射是可交换的。承诺运算将群中元素映射到 $\tau_1:G_1\to M_1$、$\tau_2:G_2\to M_2$、$\tau_T:G_T\to M_T$；提取算子 $p_1:M_1\to G_1$、$p_2:M_2\to G_2$、$p_T:M_T\to G_T$。图 2.2 中描述了集合之间的映射关系。

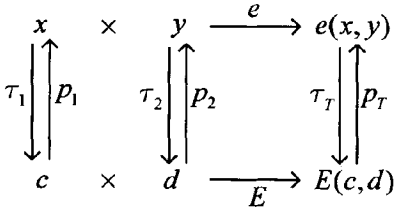

图 2.2 集合之间的映射关系

Fig 2.2 Map between the modules

并且元素之间存在关系：

$$\forall x\in G_1,\forall y\in G_2: E(\tau_1(x),\tau_2(y))=\tau_T(e(x,y)) \tag{2.29}$$

$$\forall x\in M_1,\forall y\in M_2: e(p_1(x),p_2(y))=p_T(E(x,y)) \tag{2.30}$$

（2）GSCommit：GS 系统中的承诺包含三个承诺算法：承诺初始化算法 $GSComSetup(p,G,g)$ 输出通用参考串 $params_{Com}$；承诺算法 $GSCommit(params_{Com},x,open)$ 根据承诺对象的不同产生两种计算方式，若将 $x\in G$ 承诺，则根据具体的计算方式计算出相应的承诺值 $comm$，若对随机数 $\theta\in\mathbb{Z}_p$ 承诺则需调用另外的承诺算法 $GSExpCommit(params_{Com},b,\theta,open)$，该算法将 $\theta\in\mathbb{Z}_p$，和 $b\in G$ 作为输入，输出 $(b,comm)$，其中 $comm=GSCommit(params_{Com},b^{\theta},open)$。承诺验证算法 $VerifyOpening(params_{Com},comm,x,open)$，将承诺对象和 $open$ 值作为输入，若产生相同的承诺值则输出 $accept$；对于 $(b,comm)$ 通过验证 $VerifyOpening(params_{Com},comm,b^{\theta},open)$。Groth-Sahai 证明的核心思想是构建一个承诺，其中包含模 R 的阿贝尔群 M 中若干个元素 $u_1,...u_I$，承诺使用一系列的随机数点乘群元素 u_i。如果 u_i 可以生成整个加法群，则这个承诺方案具有良好的隐藏性。若参数只产生加法群的一部分 U，则这个承诺方案只是在 M/U 集合范围内具有隐藏性。下面给出具体的承诺方法：若要对 $x\in G$ 进行承诺，则希望将 x 映射到唯一的 $x'\in M$ 上，并且希望 x'

不是 U 中的元素。对于 $u_1,...,u_I$ 是 M 中的元素，他们组成的集合 U 称为 M 的子群。因此随机选择 $r_1,...,r_I \leftarrow Z_p$，计算 $comm = x'\prod_{i=1}^{I} u_i^{r_i}$。

（3）GSProve：GS 系统中的 NIZK 证明算法将要证明的陈述和系统参数作为输入，最终输出证明。定义 $E: M_1 \times M_2 \rightarrow M_T$，$U$ 是由 $u_1,...,u_I$ 构成的 M_1 的子群；V 是由 $v_1,...,v_J$ 构成的 M_2 的子群。分别对 M_1 中的元素 x_i，M_2 中的元素 y_i 进行承诺。要证明 $c_1,...,c_Q \in M_1$ 是对 $x_1,...,x_Q$ 的承诺，即 $c_q = x_q\prod_{i=1}^{I} u_i^{r_{qi}}$；$d_1,...,d_Q \in M_2$ 是对 $y_1,...,y_Q$ 的承诺，即 $d_q = y_q\prod_{i=1}^{J} v_j^{s_{qi}}$；并且存在等式 $\prod_{q=1}^{Q} e(x_q,y_q) = t$。示证者可以计算 π_i 和 φ_i，满足下述等式：

$$\prod_{q=1}^{Q} E(c_q,d_q) = t'\prod_{i=1}^{I} E(u_i,\pi_i)\prod_{j=1}^{J} E(\varphi_j,v_j) \tag{2.31}$$

其中 t' 是 t 在 M_T 中的映射。可通过 x_i、y_i，其上的承诺值、以及打开信息计算出 π_i 和 φ_i。示证者随机选择 $r_{qi} \leftarrow Z_p$，其中 $1 \le i \le I, 1 \le q \le Q$；随机选择 $s_{qj} \leftarrow Z_p$，其中 $1 \le j \le J, 1 \le q \le Q$；随机选择 $t_{ij} \leftarrow Z_p$，其中 $1 \le i \le I, 1 \le j \le J$；随机选择 $t_h \leftarrow Z_p$，其中 $1 \le h \le H$；则

$$\pi_i = \prod_{j=1}^{J} v_j^{t_{ij}} \prod_{q=1}^{Q} d_q^{r_{qi}}，\quad \varphi_j = \prod_{i=1}^{I} u_i^{\sum_{h=1}^{H} t_h \eta_{h_{ij}}} \prod_{i=1}^{I} u_i^{-t_{ij}} \prod_{q=1}^{Q} x_q^{s_{qj}} \tag{2.32}$$

2.6 本章小结

本章首先介绍了非交互式零知识证明的数学基础，如双线性配对函数与三素数组合阶双线性群理论，介绍了 SDA 难题、DLA 难题、HSDH 等假设难题；然后阐述了零知识证明系统与非交互式零知识证明系统的基本构成与安全性，最后对典型的 NIZK 证明系统-GS 证明系统及安全特性进行了详细讨论。

第3章 基于NIZK的GSCOBG组签名方案

本章将介绍基于三素数阶群的组签名方案 GSCOBG，该方案应用了非交互式零知识证明理论，其在标准模型下证明是安全的。此外还给出了基于非交互式零知识证明的组签名定义，并对 GSCOBG 方案进行详细的安全性证明和效率分析，通过比较得出了安全性优势和效率优势。

3.1 组签名的安全模型和安全性定义

组签名是一种比较特殊的数字签名方案。在一个组签名方案中，组中的任意成员可以以匿名的方式代表整个组对消息进行签名。它在传统数字签名的基础上，加入了组管理者。组管理者决定新成员是否可以加入组，一旦成员加入组中，便可代表组对某一消息进行签名。组签名是可以公开验证的，但验证者不能从签名中得到签名者的身份。当产生纠纷时，组管理者可以将签名打开，提取出签名者的身份信息，防止组签名的伪造和抵赖。由于组签名方案的特殊性，被广泛应用于电子商务中，比如电子拍卖、电子投标、匿名投票等系统中。

按照组签名安全性的证明方法，现有的方案可以分为基于随机预言模型和基于标准模型的组签名。在基于随机预言模型（ROM）的安全性证明方面取得了很大成功，研究者们通过证明签名的不可伪造性、抗联合攻击等重要性质，说明其方案是安全的，但在实际使用中并不安全。因为使用 ROM 时假设签名的明文是均匀随机的，但在实际应用中并不存在均匀随机的明文。此后，研究者们开始避免使用 ROM，通过使用标准模型设计高效安全的组签名方案。

第一个在标准模型下证明安全的组签名模型由 Bellare 等人提出，称之为 BMW 模型[40]，该模型给出了组签名的通用结构及其应该满足的安全性要求，并在组签名过程中引入非交互式零知识证明理论。该结构满足完全匿名性和完全可提取性，组签名的其他性质可以从这两个基本性质中得到证明，而且签名密钥的长度是组内成员的对数级，签名的大小是常量。

BMW 模型的缺点之一是不能动态加入组成员，成员最大数量在初始化阶段就已经确定，而且真实的成员数量难以确定，这一点影响了组签名在很多方面的应用（比如要求动态加入成员的场合）。为了解决上述局限性，Bellare 等人于 05 年将 BMW 模型扩展为 BSZ 模型[41]，BSZ 模型中增加了成员的“加入”阶段，并增强了安全性。但每增加一个阶段，就意味着增加交互的次数，增加通信代价。

3.1.1 安全模型

本节在BMW模型的基础上介绍了组签名方案的算法构成与安全性定义，该模型中基本的安全性包含完全匿名性和完全可提取性两部分，其余的性质包含在上述两个基本性质当中，本节将BMW模型的结构进行了适当扩充。

本节定义的组签名包含六个多项式时间内算法（*Setup*，*KeyGen*，*Sign*，*Commit*，*Verify*，*Trace*）：组签名初始化算法 *Setup*，组签名密钥产生算法 *KeyGen*，组内成员签名算法 *Sign*，签名验证算法 *Verify*，签名追踪算法 *Trace*。

Setup 算法：输入安全参数，系统产生所用到的所有公共参数和随机数。

KeyGen 算法：系统将公共参数作为输入，产生三元组（*TK*，*MK* ，K_{ID} ）其中 *TK* 为追踪密钥，*MK* 为组管理者私钥，K_{ID} 是身份信息为 *ID* 的组内成员的签名密钥。

Sign 算法：该算法将组内成员 *ID* 的签名密钥和要签署的消息 *M* 作为输入，算法输出一个在消息 *M* 的签名，签名要满足特定的验证等式。

Commit 算法：该算法将上述的签名值及其满足的特定关系作为输入，产生和签名值所对应的承诺值，并将特定关系转化为配对乘积等式。

Verify 算法：该算法将公共参数、消息 *M*，签名值σ 作为输入，如验证通过输出 1，否则输出 0。

Trace 算法：该算法将组管理提取密钥 *TK*、消息 *M* 和在 *M* 上的签名值σ 作为输入，算法输出签名者标识 *ID*。

3.1.2 安全性定义

对于一个安全的组签名方案须满足下列安全性：

（1）正确性：对于诚实的组成员生成的签名，验证者必须接受签名，且通过提取算法能够从正确的签名中提取出正确的成员身份。

形式化表述如下：

$$\Pr[params \leftarrow Setup(1^{\lambda});(MK,TK,K_{ID}) \leftarrow Keygen(params);\sigma \leftarrow Sign(params,K_{ID},M):$$
$$Verify(params,m,\sigma)=1 \wedge Trace(TK,M,\sigma)=ID]=1 \qquad (3.1)$$

（2）完全匿名性：匿名性要求敌手不能算出组管理者的私钥，也不能恢复出组成员的身份信息。敌手 $\mathcal{A}$ 执行两个阶段，选择阶段和猜测阶段。在选择阶段 $\mathcal{A}$ 输入组成员的私钥 K_{ID}，公共参数 *params*，在这个阶段也可以询问提取密钥。在该阶段结束时，$\mathcal{A}$ 输出两个合法的成员标识 ID_1, ID_2 和消息 *M*，同时也产生一些状态信息用于第二阶段的攻击。在第二阶段敌手得到随机签名者在 *M* 上的一个签名值。敌手的最终目标是猜出签名值是使用哪个密钥所签署的，但猜中的概率是 *1/2*。形式化的表述如下：

$$\Pr[params \leftarrow Setup(1^{\lambda});(ID_1,ID_2,M,State) \leftarrow \mathcal{A}_1(params,K_{ID});b \leftarrow \{0,1\};$$
$$\sigma \leftarrow Sign(params,K_{ID_b},M);b' \leftarrow \mathcal{A}_2(params,K_{ID}):$$
$$Verify(params,m,\sigma)=1 \wedge b=b']=\frac{1}{2} \qquad (3.2)$$

（3）完全可提取性：完全可提取性要求组内若干个成员不能密谋伪造签名，伪造的签名不能打开提取出合法的成员身份，敌手已知提取密钥 TK，敌手根据已知密钥不能求出组管理者私钥。形式化的定义为：敌手 $\mathcal{A}$ 执行两个阶段的操作，选择阶段和猜测阶段。集合 G 中包含合法的用户 ID 与数量，选择阶段敌手通过不可靠的集合展开攻击，集合 G' 包含不可靠成员的身份标识和成员数量。猜测阶段敌手尝试伪造一个在消息 M 上的签名 σ，如果签名是合法的签名，而提取算法输出空值或提取的身份标识 $ID' \in G$，则敌手攻击成功，否则失败。形式化表述如下：

$$\Pr[params \leftarrow Setup(1^{\lambda});G' \leftarrow \mathcal{A}_1(params);\sigma' \leftarrow Sign(params,K_{ID^*},M);$$
$$ID' \leftarrow Trace(params,TK,\sigma'):Verify(params,m,\sigma')=1 \wedge ID' \in G]=0 \qquad (3.3)$$

通过以上两个重要性质可以得出如下安全特性：

（4）不可伪造性：数字签名的基本要求是签名不能被伪造，换句话说伪造的签名不能被验证算法所接受。实际上不可伪造性包含在完全可追踪性中。从完全可追踪性的第二阶段可以定义不可伪造性，在第一阶段敌手没有询问私钥，同时组管理者私钥不是状态初始化的一部分。敌手可以通过签名预言机获得选择阶段的数字签名。因此不可伪造性被成功规约到可提取性上，若敌手产生一个合法的签名 (m,σ)，并使用提取算法追踪成员身份，追踪到集合 G' 之外的用户是不可能的，实际上集合 G' 是一个空集合。

（5）抗联合攻击性：组内成员包括组管理者在内不能联合产生代表该组其他成员的签名，称之为抗联合攻击性。强抗联合攻击性可以从完全可提取性证明中获得，在询问过程中敌手没有得到组管理者的私钥。因此联合产生的签名使用提取算法输出的成员身份落在集合 G' 中，而 G' 是空集合。因此对于一个具有完全可提取性的组签名同时也具有抗联合攻击的性质。

（6）不可连接性：组签名的不可连接性来源于匿名性，而匿名性是完全匿名性的一种情况。不可连接性是指在所有的签名中，不能判断任意两个或多个签名是否来自同一签名者。匿名性和不可连接性实际上是相同的性质。

3.2 GSCOBG 方案描述

本节利用组合阶双线性群构建了一个高效组签名方案，方案中引入了 Groth 和 Sahai 提出的非交互式证明系统的思想，并结合承诺方案加以构建。首先，系统初始化过程中输出初始化参数 λ，初始化算法产生系统参数，包括承诺参数 $(u_1,...u_k) \leftarrow G_{p_1}{}^k$，

$(v_1,...v_n) \leftarrow G_{p_1}{}^n$，组管理者和组成员计算过程中所使用的随机数以及随机生成元，并为每个组成员分配一个唯一的身份信息 $ID=(x_1,...,x_k) \leftarrow \{0,1\}^k$，最终将公共信息 $PP=\{N,g,u,u_1,...u_k,v,v_1,...v_n,A=e(g,g)^{\alpha}\}$ 公布。组管理者调用密钥生成算法为每位组成员生成签名密钥 K_{ID}，并产生证明 π_1 供签名者证明自身身份。签名者调用签名算法通过输入签名密钥 K_{ID}，产生在消息 $M=(m_1,...,m_n) \leftarrow \{0,1\}^n$ 上的原始签名 $S=(S_1,S_2,S_3)$，该签名满足原始的验证等式 $e(S_1,g)e(S_2,u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(S_3,v^{r_2}\prod_{j=1}^{n}v_j^{m_j})=A$，签名者再次调用非交互式承诺算法 *Commit*，最终产生关于原始签名的非交互式零知识证明 π_2，和最终验证等式。验证者调用验证算法来验证关于最终签名的非交互式零知识证明的正确性。当验证者和签名者产生异议时，需要有组管理者调用追踪算法追踪出签名者的身份，防止非法抵赖。图 3.1 描述了整个组签名方案的框架。

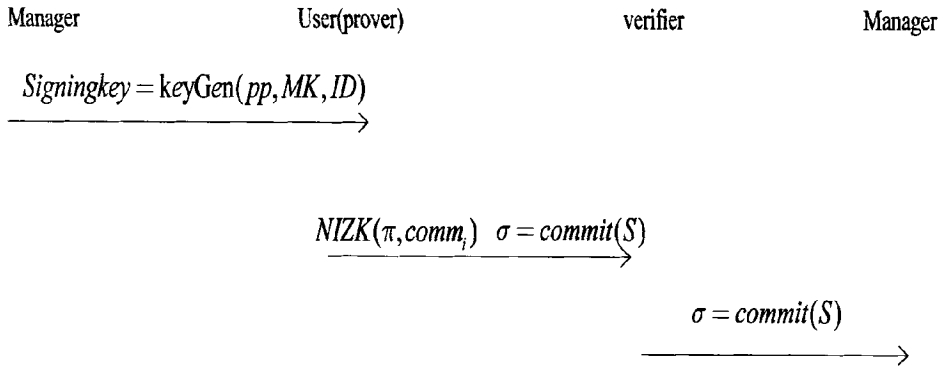

图 3.1 组签名方案

Fig. 3.1 Group signature scheme

三素数的组合阶双线性群具备特殊的性质，更方便于构造协议，相关性质介绍如下：群 G 是一个阶为 $N=p_1p_2p_3$ 的循环群，$e:G^2 \rightarrow G_T$ 是一个双线性映射，所选的元素是对应群中的生成元。在三素数组合阶双线性群中，令群 G_{p_1}，G_{p_2} 和 G_{p_3} 为群 G 的子群，各自的阶分别为 p_1,p_2 和 p_3。元素 g 是群 G 的一个生成元。根据组合阶群的正交性可以得出：元素 $g^{p_1p_2}$ 是群 G_{p_3} 的生成元，元素 $g^{p_1p_3}$ 是群 G_{p_2} 的生成元，元素 $g^{p_2p_3}$ 是群 G_{p_1} 的生成元。假设 $h_1 \in G_{p_1}$，$h_2 \in G_{p_2}$，因此，对于 α_1,α_2，$h_1=(g^{p_2p_3})^{\alpha_1}$ 和 $h_2=(g^{p_1p_3})^{\alpha_2}$，可以推导出 $e(h_1,h_2)=e(g^{p_2p_3\alpha_1},g^{p_1p_3\alpha_2})=e(g^{\alpha_1},g^{p_3\alpha_2})^{p_1p_2p_3}=1$.

初始化过程中，λ 是安全参数，身份信息 *ID* 和消息 *M* 可以分别表示成 *k* 位和 *n* 位的二进制串，要求 *k*<=*n*<<λ，组内可以支持 2^k 个成员加入。G 是一个可交换的乘法群，阶为 $N=p_1p_2p_3$，其中 p_1,p_2,p_3 为素数，$e:G\times G \rightarrow G_T$ 为双线性映射。下面将详细的描述组签名的每个算法组成及签名、验证、追踪的过程。

Setup (1^{λ})：令 G_{p_i} 是群 G 的一个子群，阶为 p_i，在群 G_{p_i} 中随机选择生成元

$g,v \leftarrow G_{p_1}{}^2$，$u \leftarrow G_{p_2}$，向量$(u_1,...u_k) \leftarrow G_{p_1}{}^k$用于承诺用户$ID$，$ID=(x_1,...,x_k) \leftarrow \{0,1\}^k$；向量$(v_1,...v_n) \leftarrow G_{p_1}{}^n$用于承诺签名消息$M$，$M=(m_1,...,m_n) \leftarrow \{0,1\}^n$，选择数$\alpha \leftarrow \mathbb{Z}_N$。因此公共参数为$PP=\{N,g,u,u_1,...u_k,v,v_1,...v_n,A=e(g,g)^\alpha\}$，，$MK=g^\alpha \leftarrow G, TK=p_2$公共参数不仅包括$pp$而且包括了$k$，$n$以及$(N,G,G_T,e)$的相关描述。

KeyGen *(PP, MK, ID)*：系统为用户分配一个唯一标识$ID=(x_1,...,x_k) \leftarrow \{0,1\}^k$，并为其产生签名密钥，选择随机数$r_1 \leftarrow \mathbb{Z}_N$，在群$G_{p_3}$中选择两个元素，$R_3,R_3^{'} \leftarrow G_{p_3}{}^2$，可以通过$G_{p_3}$中的生成元，和随机数计算产生。由所选的元素计算用户ID的签名密钥如下：$K_{ID}=\{K_1=g^\alpha(u^{r_1}\prod_{i=1}^k u_i^{x_i})^{r_1}R_3, K_2=g^{r_1}R_3^{'}\}$，并生成基于非交互式零知识的证明$\pi_1=(u^{r_1}\prod_{i=1}^k u_i^{2x_i-1})^{r_1}$，用于向验证者证实自己的身份。

Sign *(PP, K_{ID}, M)*：用户使用秘密的签名密钥$K_{ID}=(K_1,K_2) \leftarrow G^2$对消息$m$进行签名。选择随机数$s \leftarrow \mathbb{Z}_N$，得到如下签名：$S=(S_1,S_2,S_3)=(K_1(v^{r_2}\prod_{j=1}^n v_j^{m_j})^s, K_2^{-1}, g^{-s})$。而$S_1,S_2,S_3$首先满足配对乘积等式：$e(S_1,g)e(S_2,u^{r_1}\prod_{i=1}^k u_i^{x_i})e(S_3,v^{r_2}\prod_{j=1}^n v_j^{m_j})=A$

因为以上签名中包含秘密签名密钥，所以根据 Groth-Sahai 的非交互式零知识证明思想，应对私密信息进行承诺，进而使得验证者确信承诺中包含私密信息。按照非交互式零知识证明中的承诺方法，如果对群中元素$x \leftarrow G$承诺，则将x一一映射为x'选择随机数$r_1,...,r_l \leftarrow \mathbb{Z}_N{}^l$计算承诺值$comm=x'\prod_{i=1}^l u_i^{r_i}$。为了方便最后的追踪，本方案中将$x$作为$x$的一一映射$x \to x$，从而简化承诺的复杂程度，如果对$\mathbb{Z}_N$中的元素$m$承诺，则计算承诺值$comm=u^m\prod_{i=1}^l u_i^{r_i}$。

Commit *(pp, S)*：本方案中需要对上述签名过程中产生的三个签名元素S_1,S_2,S_3以及$u^{r_1}\prod_{i=1}^k u_i^{x_i}, v^{r_2}\prod_{j=1}^n v_j^{m_j}$承诺。选择$t,t_1,t_2,t_3 \in \mathbb{Z}_N$对元组$S_1,S_2,S_3,u^{ID}h_1,v^m h_2$承诺如下：$comm_1=S_1h^{t_1}, comm_2=S_2h^{t_2}, comm_3=S_3h^{t_3}, comm_4=u^{r_1}\prod_{i=1}^k u_i^{x_i}h^t, comm_5=v^{r_2}\prod_{j=1}^n v_j^{m_j}h^t$，并计算$\pi_2=g^{t_1-r_1t-st}\cdot(u^{r_1}\prod_{i=1}^k u_i^{x_i})^{t_2}\cdot(v^{r_2}\prod_{j=1}^n v_j^{m_j})^{t_3}\cdot h^{t(t_2+t_3)}$

最终签名由以下元组组成：$\sigma=(comm_1,comm_2,comm_3,comm_4,comm_5,\pi_2)$

Verify *(PP,σ)*：验证者收到签名后，利用公开参数对身份的证书和签名进行验证，如果身份和签名都满足验证等式，则验证通过，否则返回错误消息。对身份进行验证时，令$C=u^{r_1}\prod_{i=1}^k u_i^{x_i}$只需验证等式$e(C,C\prod_{i=1}^k u_i^{-1})=e(u,\pi_1)$是否成立，如果成立则继续对签名进行验证，否则返回错误消息。对消息进行验证时只需满足下列验证等式：$e(comm_1,g)e(comm_2,comm_4)e(comm_3,comm_5)=A\cdot e(h,\pi_2)$，如果等式成立则签名将被验证者所接受，否则返回错误消息。

Trace *(pp, C_i, TK)*：当签名者对$ID=x_1,...,x_k \leftarrow \{0,1\}^k$的每一位进行承诺时，选择随机数$r_{i_1},...,r_{i_k} \leftarrow (\mathbb{Z}_N)^k$，而$r=r_{i_1}+r_{i_2}+...+r_{i_k}$，计算$C_i=u^{r_{i_j}}\cdot u_i^{x_i}$使得

$C=C_1\cdot C_2...\cdot C_n=u^{r_1}\prod_{i=1}^{k}u_i^{x_i}$。当产生纠纷时，需要组管理者追踪出签名者身份，组管理者使用追踪密钥$TK=p_2$，通过计算$(C_i)^{p_2}$的值恢复出ID。若$(C_i)^{p_2}=1$（其中1为群G中的单位元），则$x_i=0$；若$(C_i)^{p_2}\neq 1$，则$x_i=1$，从而依次恢复出用户的身份信息ID所对应的二进制串。

3.3 安全性证明及效率分析

本节将对GSCOBG方案的安全性和效率进行证明和分析，并将该方案与同类的系列方案进行对比。

3.3.1 安全性证明

安全性的证明包含两部分，第一部分为BMW模型的安全性要求，包括正确性、完全匿名性、完全可提取性、不可伪造性；第二部分安全性证明是针对非交互式系统的证明，包括正确性、完备性和不可区分性的证明。两部分证明都成立的前提下，该签名方案才是安全的。

定义3.1 如果$(Setup, KeyGen, Sign, Verify)$是一个具有正确性、完全匿名性、不可伪造性，完全可追踪性的组签名方案，且对于组成员身份的证明过程$(Setup, KeyGen, Verify, Trace)$是一个具有正确性、公正性、零知识性、的非交互式证明系统，则该组签名方案是安全的。

定理3.1 $Setup(1^k)$，$KeyGen(PP, MK, ID)$，$Verify(PP,\sigma)$是一个具有正确性、不可伪造性，完全匿名性、完全可追踪性的组签名方案。

（一）正确性

正确性的证明包含三个部分：身份验证等式的正确性、最初签名验证等式的正确性、承诺以后最终签名验证等式的正确性，下面分别对以上三个等式进行验证。

证明：对于$ID=x_1,...,x_k\leftarrow\{0,1\}^k$每一位承诺时，$x_i$非0即1，若要证明对于身份的验证等式$e(C,C\prod_{i=1}^{k}u_i^{-1})=e(u,\pi_1)$成立，通过等式的左边可得：

$$\begin{aligned}
e(C,C\textstyle\prod_{i=1}^{k}u_i^{-1}) &= e(u^{r_1}\textstyle\prod_{i=1}^{k}u_i^{x_i},u^{r_1}\textstyle\prod_{i=1}^{k}u_i^{x_i-1})\\
&= e(u^{r_1},u^{r_1})e(\textstyle\prod_{i=1}^{k}u_i^{x_i},\textstyle\prod_{i=1}^{k}u_i^{x_i-1})e(u^{r_1},\textstyle\prod_{i=1}^{k}u_i^{x_i-1})e(u^{r_1},\textstyle\prod_{i=1}^{k}u_i^{x_i})\\
&= e(u^{r_1},u^{r_1})e(u^{r_1},\textstyle\prod_{i=1}^{k}u_i^{2x_i-1})\textstyle\prod_{i=1}^{k}e(u_i^{x_i},\textstyle\prod_{i=1}^{k}u_i^{x_i-1})\\
&= e(u^{r_1},u^{r_1}\textstyle\prod_{i=1}^{k}u_i^{2x_i-1})\textstyle\prod_{i=1}^{k}\prod_{j=1}^{k}e(u_i^{x_i},u_j^{x_j-1})\\
&= e(u,(u^{r_1}\textstyle\prod_{i=1}^{k}u_i^{2x_i-1})^{r_1})\textstyle\prod_{i=1}^{k}\prod_{j=1}^{k}e(u_i,u_j)^{x_i(x_j-1)}=e(u,\pi_1)
\end{aligned}\qquad(3.4)$$

签名算法中，组成员产生最初的签名包含三个签名值(S_1,S_2,S_3)，该三元组满足等式$e(S_1,g)e(S_2,u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(S_3,v^{r_2}\prod_{j=1}^{n}v_j^{m_j})=A$，若要证明该签名的验证等式成立，通过计算等式的左边可得：

$$
\begin{aligned}
&e(S_1,g)e(S_2,u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(S_3,v^{r_2}\prod_{j=1}^{n}v_j^{m_j})\\
&=e(g^{\alpha}(u^{r_1}\prod_{i=1}^{k}u_i^{x_i})^{r_1}R_3(v^{r_2}\prod_{j=1}^{n}v_j^{m_j})^{s},g)e((g^{r_1}R_3^{'})^{-1},u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(g^{-s},v^{r_2}\prod_{j=1}^{n}v_j^{m_j})\\
&=e(g^{\alpha},g)e((u^{r_1}\prod_{i=1}^{k}u_i^{x_i})^{r_1},g)e(R_3,g)e((v^{r_2}\prod_{j=1}^{n}v_j^{m_j})^{s},g)\cdot e(g^{-r_1},u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(R_3^{'-1},u^{r_1}\prod_{i=1}^{k}u_i^{x_i})\cdot e(g^{-s},v^{r_2}\prod_{j=1}^{n}v_j^{m_j})\\
&=e(g^{\alpha},g)e(u^{r_1}\prod_{i=1}^{k}u_i^{x_i},g)^{r_1}e(R_3,g)e(v^{r_2}\prod_{j=1}^{n}v_j^{m_j},g)^{s}e(g,u^{r_1}\prod_{i=1}^{k}u_i^{x_i})^{-r_1}e((R_3^{'})^{-1},u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(g,v^{r_2}\prod_{j=1}^{n}v_j^{m_j})^{-s}\\
&=e(g^{\alpha},g)e(R_3,g)e((R_3^{'})^{-1},u^{r_1}\prod_{i=1}^{k}u_i^{x_i})\\
&=e(g^{\alpha},g)=A
\end{aligned}
\tag{3.5}
$$

若元素g'是群G的一个生成元，根据组合阶双线性群的正交性，元素$g'^{p_1p_2}$是群G_{p_3}的生成元，元素$g'^{p_1p_3}$是群G_{p_2}的生成元，元素$g'^{p_2p_3}$是群G_{p_1}的生成元。假设$h_1\in G_{p_1}$，$h_2\in G_{p_2}$，因此，对于$\alpha_1,\alpha_2,h_1=(g'^{p_2p_3})^{\alpha_1}$和$h_2=(g'^{p_1p_3})^{\alpha_2}$，可以推导出$e(h_1,h_2)=e(g'^{p_2p_3\alpha_1},g'^{p_1p_3\alpha_2})=e(g^{\alpha_1},g'^{p_3\alpha_2})^{p_1p_2p_3}=1$。因此很容易得出如下结论：在配对函数中的两个元素若来自群G的任意两个不同子群G_{p_i}，则其函数值为单位元1，从而配对函数$e(R_3,g)$和$e((R_3^{'})^{-1},u^{r_1}\prod_{i=1}^{k}u_i^{x_i})$的值均为1。因此签名验证等式$e(S_1,g)e(S_2,u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(S_3,v^{r_2}\prod_{j=1}^{n}v_j^{m_j})=A$满足正确性要求。

承诺后的签名值$\sigma=(comm_1,comm_2,comm_3,comm_4,comm_5,\pi_2)$满足验证等式$e(comm_1,g)e(comm_2,comm_4)e(comm_3,comm_5)=A\cdot e(h,\pi_2)$，为证明上述等式成立，计算等式的左边可以得出：

$$
\begin{aligned}
&e(comm_1,g)e(comm_2,comm_4)e(comm_3,comm_5)\\
&=e(S_1h'^{t_1},g)e(S_2h'^{t_2},u^{r_1}\prod_{i=1}^{k}u_i^{x_i}h')e(S_3h'^{t_3},v^{r_2}\prod_{j=1}^{n}v_j^{m_j}h')\\
&=e(S_1,g)e(h'^{t_1},g)\cdot e(S_2,u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(S_2,h')e(h'^{t_2},u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(h'^{t_2},h')\cdot e(S_3,v^{r_2}\prod_{j=1}^{n}v_j^{m_j})\\
&\cdot e(h'^{t_3},v^{r_2}\prod_{j=1}^{n}v_j^{m_j})e(S_3,h')e(h'^{t_3},h')\\
&=e(g^{\alpha},g)e(h'^{t_1},g)e(S_2,h')e(h'^{t_2},u^{r_1}\prod_{i=1}^{k}u_i^{x_i})e(h'^{t_2},h')e(h'^{t_3},v^{r_2}\prod_{j=1}^{n}v_j^{m_j})e(S_3,h')e(h'^{t_3},h')\\
&=e(g^{\alpha},g)e(h,g^{t_1})e(g^{-r_1t}R_3^{'-1},h')e(h,(u^{r_1}\prod_{i=1}^{k}u_i^{x_i})^{t_2})e(h,h'^{tt_2})e(h,(v^{r_2}\prod_{j=1}^{n}v_j^{m_j})^{t_3})e(S_3^{t},h)e(h,h'^{tt_3})\\
&=e(g^{\alpha},g)e(h,g^{t_1})e(g^{-r_1t},h)e(h,(u^{r_1}\prod_{i=1}^{k}u_i^{x_i})^{t_2})e(h,h'^{tt_2})e(h,(v^{r_2}\prod_{j=1}^{n}v_j^{m_j})^{t_3})e(g^{-st},h)e(h,h'^{tt_3})\\
&=e(g^{\alpha},g)e(h,g^{t_1-r_1t-st}h^{t(t_2+t_3)}(u^{r_1}\prod_{i=1}^{k}u_i^{x_i})^{t_2}(v^{r_2}\prod_{j=1}^{n}v_j^{m_j})^{t_3})
\end{aligned}
$$

$= A \cdot e(h, \pi_2)$　　(3.6)

因此可以得出三部分验证等式都是满足正确性的，从而整个签名方案满足正确性要求。

（二）完全匿名性

引理 3.1 如果多项式时间内敌手在时间 t 内可以 ε 的概率攻破假设（2）或假设（3），则在多项式时间 t' 内，敌手 $\mathcal{A}$ 就可以以概率 $\geq \frac{\varepsilon}{2}$ 的概率攻破完全匿名性，其中 $t \approx t'$

证明：Brent Waters 等人引进了半适用性的概念，即：半适用性签名密钥 $K_1' = K_1 \cdot g_2^{\gamma z_k}, K_2' = K_2 \cdot g_2^{\gamma}, E_{k+1} = E'_{k+1} g^{\gamma z_{k+1}}, ..., E_l = E'_l g^{\gamma z_l}$，半适用性签名：$comm_4' = comm_4 \cdot g_2^{xz_s}, comm' = g' = g \cdot g_2^{x}$。使用半适用性的签名和密钥验证等式的左边可以得出 $e(g,g)^{\alpha} \cdot e(g_2, g_2)^{xy(z_k - z_s)}$，如果 $z_k = z_s$ 则验证等式是成立的。

本节将使用 *Game* 序列进行证明，其中 $Game_{Real}$ 是真实且安全的 *Game* 序列，而 $Game_{Restricted}$ 序列相对于 $Game_{Real}$ 而言，只对敌手询问做了一些限制，要求所有身份信息 ID 模 N 后的值是不相同的。q 表示敌手询问密钥的次数。k 从 *0-q* 中取值，定义 $Game_k$ 如下：$Game_k$ 类似于 $Game_{Restricted}$ 序列，只是提供给敌手的签名是半适用性的，第一个签名密钥是半适用性的，而其余签名密钥是正常的。$Game_0$ 中，所有的密钥是正常的，签名是半适用性的，在 $Game_q$ 中，签名和所有的签名密钥都是半适用性的，最后一个 *Game* 序列是 $Game_{Final}$，类似于 $Game_q$，只是产生的签名是关于随机消息的半适用性签名。

给定 $g, X_3, \mathcal{B}$ 可以使用算法 $\mathcal{A}$ 模拟 $Game_{Real}$，算法 $\mathcal{A}$ 产生身份标识 ID, ID^*，其满足 $ID \neq ID^* \bmod N$，并且 p_2 整除 $ID - ID^*$。算法 $\mathcal{B}$ 使用上述 ID 计算有效解 $a = \gcd(ID - ID^*, N)$。令 $b = \frac{N}{a}$，有 p_2 整除 a，且 $N = ab = p_1 p_2 p_3$。存在两种情况：p_1 整除 b，或 p_1 整除 a，从而有 $a = p_1 p_2$，$b = p_3$。其中之一会以概率 $\geq \frac{\varepsilon}{2}$ 的概率出现。在情况 1 中算法 $\mathcal{B}$ 会攻破假设(2)。给定的 $g, X_3, \mathcal{B}$ 能够通过验证 g^b 是否是身份标识来判定 p_1 整除 b，然后验证 T^b 是否是身份标识，如果是身份标识，则 $T \in G_{p_1}$，反之，$T \in G_{p_1 p_2}$。情况 2 中，算法 $\mathcal{B}$ 攻破假设（3），给定 $g, X_1 X_2, X_3, Y_2 Y$，算法 $\mathcal{B}$ 能够通过验证 $(X_1 X_2)^a$ 是身份标识来判定 $a = p_1 p_2$，然后验证双线性函数 $e((Y_2 Y_3)^b, T)$ 是否是身份标识，如果是，则 $T \in G_{p_1 p_3}$，否则 $T \in G$。

然而敌手攻破假设（2）的概率是可忽略的，说明 $Game_{Real}$ 序列与 $Game_{Restricted}$ 序列是不可区分的，即 $Game_{Real} Adv_{\mathcal{A}} - Game_{Restricted} Adv_{\mathcal{A}} = 0$，根据逆否命题的性质可以得出敌

手攻破完全匿名性的概率为 $Adv_A < \frac{\varepsilon}{2}$。

（三）不可伪造性

假设敌手可以攻破第二节描述的假设难题，通过得出的结论可知与假设是互相矛盾的，从而得出不可伪造性的结论。

不可伪造性的证明依赖于假设(2)(3)(4)，以下定理说明 $Game_{Restricted}$ 序列与 $Game_0$ 序列是不可区分的，$Game_0$ 序列与 $Game_{k-1}$ 序列是不可区分的，$Game_{k-1}$ 序列与 $Game_k$ 序列是不可区分的，$Game_q$ 序列与 $Game_{Final}$ 序列是不可区分的，最终得出每个序列都是不可区分的。

引理 3.2 假设如果存在算法 $\mathcal{A}$ 使得 $Game_{Restricted}Adv_A - Game_0 Adv_A = \varepsilon$，那么可以构造一个算法 $\mathcal{B}$ 以 $\geq \frac{\varepsilon}{2}$ 的概率攻破假设（2）。

证明：$\mathcal{B}$ 首先得到值 g, X_3, T 之后可以使用算法 $\mathcal{A}$ 模拟 $Game_{Restricted}$ 或 $Game_0$。$\mathcal{B}$ 算法选择随机数 $\alpha, a_1, ..., a_l, b \in Z_N$，计算 $g = g, u_i = g^{a_i}, h = g^b, i \in \{1, ..l\}$，并将公共参数 $\{N, u_1, ...u_l, g, h, e(g,g)^\alpha\}$ 发给 $\mathcal{A}$。当攻击者询问算法 $\mathcal{B}$ 对于身份 $ID = (x_1, ..., x_k)$ 的密钥时，选择随机数 $r, t, w, v_j, ..., v_l \in Z_N$，并计算：

$K_1 = g^\alpha (u^r \prod_{i=1}^{k} u_i^{x_i})^r X_3^w, K_2 = g^r X_3^t, E_{k+1} = u_{k+1}{}^r X^{v_{k+1}}, ..., E_l = u_l{}^r X_3^{v_l}$。

算法 $\mathcal{A}$ 发送给 $\mathcal{B}$ 两个消息，M_0, M_1，和一个挑战身份信息 $ID^* = (x_1^*, ..., x_k^*)$，算法 $\mathcal{B}$ 随机选择 $\beta \in \{0,1\}$，产生签名 $comm_1 = comm_{1\beta}, comm_4 = T^{a_1 x_1^* + ... a_k x_k^* + b}, g' = T$，这里将 g^s 等同于 T 中包含 G_{p_1} 中元素的部分。若 $T \in G_{p_1 p_2}$，那么签名是一个 $z_c = a_1 x_1^* + ... a_k x_k^* + b$ 的半适用性签名。已知 $z_c \bmod p_2$ 和 $a \bmod p_1$ 及 $b \bmod p_1$ 是没有关系的，所以可以被适当的分配。如果 $T \in G_{p_1}$ 这将是一个正常的签名，因此算法 $\mathcal{B}$ 可以使用算法 $\mathcal{A}$ 的输出区分 T。

引理 3.3 如果存在一个算法 $\mathcal{A}$ 满足 $Game_{k-1}Adv_A - Game_k Adv_A = \varepsilon$，那么可以构造一个算法 $\mathcal{B}$ 以 ε 的概率攻破假设（3）。

证明：算法 $\mathcal{B}$ 得到 $g, X_1X_2, X_3, Y_2Y_3, T$，选择随机数 $a_1, ..., a_l, b \in Z_N$，并设置公共参数为 $g = g, u_i = g^{a_i}, h = g^b, e(g,g)^\alpha$，并发给算法 $\mathcal{A}$ 询问身份为 ID 的第 i 位信息，$i < j$，算法 $\mathcal{B}$ 创建一个半适用性密钥，同时选择随机数 $r, t, z_{k+1}, ..., z_l \in Z_N$，并计算 $K_1 = g^\alpha (u^r \prod_{i=1}^{k} u_i^{x_i})^r (Y_2Y_3)^z, K_2 = g^r (Y_2Y_3)^t, E_{k+1} = u_{k+1}{}^r (Y_2Y_3)^{z_{k+1}}, ..., E_l = u_l{}^r (Y_2Y_3)^{z_l}$，通过 $g_2{}^\gamma = Y_2{}^t$ 产生半适应性密钥。对于 $i > j$，算法 $\mathcal{B}$ 产生正常的密钥。为了建立身份信息 $ID = (x_1, ..., x_k)$ 第 j 次密钥询问，使用随机数 $w_k, w_{j+1}, ..., w_l \in \mathbb{Z}_N$ 产生密钥，

$K_1 = g^{\alpha} T^{-z_k} X_3^{w_k}, K_2 = T, E_{k+1} = T^{a_{j+1}} X^{w_j+1}, ..., E_l = T^{a_l} X_3^{w_l}$ 算法 $\mathcal{B}$ 令 $z_k = a_1 x_1 + ... a_k x_k + b$。如果 $T \in G_{p_1 p_3}$ 则上述密钥是一个正常密钥，如果 $T \in G$ 则是一个半适用性密钥。在某一时刻，算法 $\mathcal{A}$ 发送给 $\mathcal{B}$ 两个消息 M_0, M_1，和一个挑战身份 $ID^* = (x_1^*, ..., x_k^*)$。算法 $\mathcal{B}$ 随机选择 $\beta \in \{0,1\}$，构成的签名如下：$comm_1 = comm_{1\beta}, comm_4 = (X_1 X_2)^{a_1 x_1^* + ... a_k x_k^* + b}, g' = X_1 X_2$，该签名中将 $g^s = X_1$，$z_s = a_1 x_1^* + ... a_k x_k^* + b$。

如果 $T \in G_{p_1 p_3}$，算法 $\mathcal{B}$ 能够模拟出 $Game_{k-1}$，如果 $T \in G$，$\mathcal{B}$ 能够模拟出 $Game_k$，因此 $\mathcal{B}$ 能够使用算法 $\mathcal{A}$ 的输出区分出 T。

引理 3.4. 如果存在一个算法 $\mathcal{A}$ 满足 $Game_q Adv_{\mathcal{A}} - Game_{Final} Adv_{\mathcal{A}} = \varepsilon$，那么可以构造一个算法 $\mathcal{B}$ 以 ε 的概率攻破假设（4）。

证明：算法 $\mathcal{B}$ 得到 $g, g^{\alpha} X_2, X_3, g^s Y_2, Z_2, T$，算法 $\mathcal{B}$ 选择随机数 $a_1, ..., a_l, b \in Z_N$，设置公共参数 $g = g, u_i = g^{a_i}, h = g^b, e(g,g)^{\alpha} = e(g^{\alpha} X_2, g)$ 并发给 $\mathcal{A}$，$\mathcal{A}$ 询问身份信息为 $ID = (x_1, ..., x_k)$ 的密钥，算法 $\mathcal{B}$ 产生一个半适用性密钥，选择随机数 $c, r, t, w, z, z_{k+1}, ..., z_l, w_{j+1}, ..., w_l \in \mathbb{Z}_N$，计算：

$K_1 = g^{\alpha} (u^r \prod_{i=1}^{k} u_i^{x_i})^{r_i} X_2 Z_2^c X_3^w, K_2 = g^r Z_2^{-z} X_3^t, E_{k+1} = u_{k+1}^r (Z_2)^{-z_{k+1}} X_3^{w_{j+1}}, ..., E_l = u_l^r Z_2^{-z_l} X_3^{w_l}$ 算法 $\mathcal{A}$ 发送给 $\mathcal{B}$ 两个消息，M_0, M_1，和一个挑战身份信息 $ID^* = (x_1^*, ..., x_k^*)$，算法 $\mathcal{B}$ 随机选择 $\beta \in \{0,1\}$，产生签名 $comm_1 = comm_{1\beta}, comm_4 = (g^s Y_2)^{a_1 x_1^* + ... a_k x_k^* + b}, g' = g^s Y_2$，令 $z_s = a_1 x_1^* + ... a_k x_k^* + b$，$z_s$ 的值仅关系到模 p_2，如果 $T = e(g,g)^{\alpha}$，那么该签名就是对于消息 M_{β} 的半适用性签名。如果 T 是 G_T 中的一个元素，那么该签名是对随机消息产生的签名。因此算法 $\mathcal{B}$ 使用算法 $\mathcal{A}$ 的输出区分 T。

引理 3.5. 如果假设（2）（3）和假设（4）成立，那么该签名方案是不可伪造的。

证明：如果假设（2）（3）和假设（4）成立，那么通过上述引理可以得出真实的安全 Game 序列与 $Game_{Final}$ 序列是不可区分的，这表明 β 值对于攻击者而言是隐藏的，因此，攻击者伪造一个方案的概率是一个可忽略的函数。

（四）完全可提取性

完全可提取性的证明，基于上述签名的不可伪造性，构造一个模拟器 $\mathcal{B}$ 同敌手 $\mathcal{A}$ 进行组签名 *Game* 序列的交互。

引理 3.6 如果存在一个多项式时间敌手可以以概率 ε 在时间 t 内攻破组签名的完全可提取性，则存在一个多项式时间敌手在时间 t' 内可以以概率 ε 攻破签名的不可伪造性，其中 $t \approx t'$。

证明：在初始化算法中，$\mathcal{B}$ 产生公共参数，将提取密钥 TK 发送给敌手 $\mathcal{A}$，敌手得

到提取的权限。在模拟器与敌手的交互过程中，如果敌手 $\mathcal{A}$ 询问某一用户 ID 的签名密钥，则模拟器 $\mathcal{B}$ 通过询问签名预言机获得用户 $ID=(x_1,...,x_k)$ 的签名密钥 (K_1,K_2)，然后将其发送给敌手 $\mathcal{A}$。敌手询问在身份信息为 $ID=(x_1,...,x_k)$ 的用户在消息 $M=(m_1,...,m_n)$ 上的签名，$\mathcal{B}$ 得到签名 $S=(S_1{*},S_2{*},S_3{*})$，$\mathcal{B}$ 选择随机数 $t,t_1,t_2,t_3\in\mathbb{Z}_N$ 生成最终的签名 $\sigma=(S_1{*}\cdot h^{t_1},S_2{*}\cdot h^{t_2},S_3{*}\cdot h^{t_3},u^{r_1}\prod_{i=1}^{k}u_i^{x_i}h^t,v^{r_2}\prod_{j=1}^{n}v_j^{m_j}h^t)$，该签名的合法性可以通过提取密钥 TK 来验证，看最终能否提取出组内一个合法的身份。

某一时刻，敌手 $\mathcal{A}$ 可以通过自己掌握的知识伪造用户 $ID^*=(x_1{*},...,x_k{*})$ 在消息 $M^*=(m_1{*},...,m_n{*})$ 的签名 $\sigma^*=(comm_1^*,comm_2^*,comm_3^*,comm_4^*,comm_5^*,\pi^*)$，模拟器产生 λ，其中 λ 满足 $\lambda\equiv 1(\bmod p_1)and\lambda\equiv 0(\bmod p_2p_3)$。而 $comm_1^{*\lambda},comm_2^{*\lambda},comm_3^{*\lambda},comm_4^{*\lambda},comm_5^{*\lambda},\pi^{*\lambda}$ 满足如下验证等式：

$e(comm_1^{*\lambda},g)e(comm_2^{*\lambda},comm_4^{*\lambda})e(comm_3^{*\lambda},comm_5^{*\lambda})=A\cdot e(h_3,\pi_2^{*\lambda})$ 证明敌手可以伪造出签名，而在引理 3.2-3.5 中已经证明出签名的不可伪造性，所以该结论与假设相矛盾，从而得出引理 3.6 的正确性。

定理 3.2 *Setup* (1^k)，*KeyGen (PP, MK, ID)*，*Verify (PP,*σ*)*，*Trace (pp,* σ*)*正确性、健壮性、可提取性、证据不可区分性的非交互式零知识证明系统。

正确性已经在定理 3.1 中得到证明，下面主要证明该非交互式证明系统的其他特性。

（1）健壮性：已知验证等式 $e(C,C\prod_{i=1}^{k}u_i^{-1})=e(u,\pi_1)$ 成立，通过计算可得：

$$\begin{aligned}e(C,C\prod_{i=1}^{k}u_i^{-1})&=e(u^{r_1}\prod_{i=1}^{k}u_i^{x_i},u^{r_1}\prod_{i=1}^{k}u_i^{x_i-1})\\&=e(u^{r_1},u^{r_1})e(\prod_{i=1}^{k}u_i^{x_i},\prod_{i=1}^{k}u_i^{x_i-1})e(u^{r_1},\prod_{i=1}^{k}u_i^{x_i-1})e(u^{r_1},\prod_{i=1}^{k}u_i^{x_i})\\&=e(u^{r_1},u^{r_1})e(u^{r_1},\prod_{i=1}^{k}u_i^{2x_i-1})\prod_{i=1}^{k}e(u_i^{x_i},\prod_{i=1}^{k}u_i^{x_i-1})\\&=e(u^{r_1},u^{r_1}\prod_{i=1}^{k}u_i^{2x_i-1})\prod_{i=1}^{k}\prod_{j=1}^{k}e(u_i^{x_i},u_j^{x_j-1})\\&=e(u,(u^{r_1}\prod_{i=1}^{k}u_i^{2x_i-1})^{r_1})\prod_{i=1}^{k}\prod_{j=1}^{k}e(u_i,u_j)^{x_i(x_j-1)}\end{aligned}\tag{3.7}$$

u 的阶为 p_2，所对应的配对 $e(u,(u^{r_1}\prod_{i=1}^{k}u_i^{2x_i-1})^{r_1})$ 的阶为 1 或 p_2，从验证等式 $e(C,C\prod_{i=1}^{k}u_i^{-1})=e(u,\pi_1)$ 中可以得出 $e(C,C\prod_{i=1}^{k}u_i^{-1})$ 的阶为 1 或 p_2，可以得出 $\prod_{i=1}^{k}\prod_{j=1}^{k}e(u_i,u_j)^{x_i(x_j-1)}$ 的阶为 1 或 p_2，而 u_i 是 G_{p_1} 的生成元，其阶为 p_1，因此得出 $x_i=0$ 或 $x_i=1$。

（2）证据不可区分性：当签名者对 $ID=x_1,...,x_k\leftarrow\{0,1\}^k$ 的每一位进行承诺时，选择随机数 $r_{1_1},...,r_{1_k}\leftarrow(\mathbb{Z}_N)^k$，而 $r=r_{1_1}+r_{1_2}+...+r_{1_k}$，计算 $C_i=u^{r_{1_i}}\cdot u_i^{x_i}$ 使得

$C=C_1\cdot C_2\cdot\ldots\cdot C_n=u^{r_1}\prod_{i=1}^{k}u_i^{x_i}$。对于每一位的承诺值而言，令$C_i=u^{r_i}\cdot u_i^0=u^{r_i'}\cdot u_i^1$，产生唯一的证明$\pi$，满足验证等式$e(C_i,C_iu_i^{-1})=e(u_i,(u_i^{2x_i-1}u^{r_i})^{r_i})=e(u_i,\pi)$。可以得出，相同承诺值生成的证明$\pi=(u^{r_i'}\cdot u_i^1)^{r_i'}=u^{r_ir_i'}=(u_i^{-1}u^{r_i})^{r_i}$的值是相等的，因此对于承诺值相同的不同承诺对象而言，具有证据不可区分性。

（3）可提取性：

可提取性的证明包含隐藏性和绑定性两部分现描述如下：

①隐藏性：若验证者已知承诺值为$C_i=u^{r_i}\cdot u_i^{x_i}$，并不能从中得到签名者的签名密钥以及身份信息$x$。形式化的表示如下：

$$\Pr[params_{Com}\leftarrow \mathrm{ComSetup}(1^k);C,x,x',r,r'\leftarrow A(params_{Com}):x\neq x'\wedge C=\mathrm{Commit}(x,r)\wedge C=\mathrm{Commit}(x',r')]=0 \tag{3.8}$$

在组签名方案中生成参数时，根据假设（1）可知u与$u'\notin G_{p_2}$在计算上是不可区分的。通过计算$C_i=u^{r_i}\cdot u_i^{x_i}$所得的承诺值，根据假设（1），不存在多项式时间内的敌手分辨出究竟使用的是u^{r_i}还是u'^{r_i}。若使用u'^{r_i}时使用提取密钥$TK=p_2$，C可变形为$C_i=u'^{r_i}\cdot u_i^{x_i}=u_i^{x_i'}\cdot u'^{r_i-(x_i'-x_i)/\alpha_2}$，其中$g'^{p_1p_3\alpha_2}\in G_{p_2},g'\in G,u'=u_i^{\alpha_2}$通过提取方式打开$C_i$得到的可为任意值，计算过程如下：

$$\begin{aligned}C_i^{p_2}&=(u'^{r_i}\cdot u_i^{x_i})^{p_2}=(u_i^{x_i'}\cdot u'^{r_i-(x_i'-x_i)/\alpha_2})^{\alpha_2}\\&=u_i^{x_i'\alpha_2}u'^{r_i\alpha_2-x_i'+x_i}=u_i^{x_i'\alpha_2}u_i^{r_i\alpha_2^2-x_i'\alpha_2+x_i\alpha_2}\\&=u_i^{r_i\alpha_2^2+x_i\alpha_2}=(u_i^{\alpha_2})^{r_i\alpha_2}u_i^{\alpha_2x_i}=u'^{r_i\alpha_2}u'^{x_i}\end{aligned} \tag{3.9}$$

②绑定性：承诺值C_i与承诺对象身份x_i是一一对应的，即不存在两个x_i承诺后生成相同的承诺值。形式化表述为：

$$\begin{aligned}&\{params\leftarrow ComSetup(1^k):params\}\approx\{params\leftarrow HidingSetup(1^k):params\}\\&\{params\leftarrow HidingSetup(1^k);r\leftarrow R;C=Commit(params,x_i,r):comm\}\\&=\{params\leftarrow HidingSetup(1^k);r\leftarrow R;C=Commit(params,x',r):comm\}\end{aligned} \tag{3.10}$$

使用$C_i=u^{r_i}\cdot u_i^{x_i}$构建承诺值，基于假设1，$C_i$与$x_i$值一一对应。若给出$u\in G_{p_2}$时使用提取密钥$TK=p_2$可以从$C_i$中提取出承诺对象$x_i$，即$(C_i)^{p_2}=(u^{r_i}\cdot u_i^{x_i})^{p_2}=(u_i^{p_2})^{x_i}$，如果$(C_i)^{p_2}=1$（其中1为群$G$中的单位元），则$x_i=0$；如果$(C_i)^{p_2}\neq 1$，则$x_i=1$，从而依次恢复出用户的身份信息$ID$对应的二进制串。由此可得出承诺者的身份信息。

该方案满足BMW模型中定义的完全匿名性和完全可提取性等安全性要求，除此之外，本文运用三素数阶群中的假设证明了该组签名方案的不可伪造性。表3.1是本文方案与上文提到的基于BMW模型和BSZ模型的典型组签名方案的安全性对比。

表 3.1 安全性比较
Table 3.1 Comparison of the security

安全性 \ 方案	ACHdM[42]	BW06[33]	BW07[34]	LV09[43]	WL10[44]	OURS
非交互式	×	√	√	√	×	√
匿名性	√	√	√	√	√	√
抗选择密文攻击	×	×	√	√	×	√
不可伪造性	×	×	√	×	×	√
可追踪性	√	√	√	×	×	√
不可连接性	×	×	×	√	√	×
不可区分性	×	√	×	√	√	√
成员撤销	×	×	×	√	√	×
抗选择明文攻击	√	√	√	√	√	√
正确性	√	√	√	√	√	√
零知识性	×	×	√	√	×	√

3.3.2 效率分析与对比

本节对 GSCOBG 方案的效率分别从通信次数、通信代价、计算代价三个方面进行分析，其中通信次数即组内成员与验证者之间、组内成员与管理者之前的交互次数；通信代价中包含了构建方案时所选群 G 中的元素个数和 Z_N 中的元素个数；计算代价包含了在计算过程中所进行的求幂、点乘、配对运算。

表 3.2 通信和计算代价
Table 3.2 Communication and Compution Costs

效率 \ 方案	通信次数	通信代价		计算代价		
		G 中元素	Z_N 中元素	求幂	点乘	配对
ACHdM[42]	4	20	6+1	40	5	19
BW06 [33]	3	4k+12	4k+6	k+10	2k+6	k+4
BW07[34]	3	k+22	11	2k+3	2k+12	14
VLRGS[43]	3	2k+38	k+12	k+21	k+23	22
WL[44]	4	3k+4	7	27	7	12
OURS	3	2k+14	12	k+26	4k+12	9

通信代价中主要考虑方案中所使用的群众元素个数，如图3.2所示红色线表示本章组签名方案所使用群G中元素随消息串长度的变化趋势。计算代价主要考虑幂运算个数和配对运算的个数，图3.3中红色线表示方案中配对运算个数随消息串长度的变化趋势。

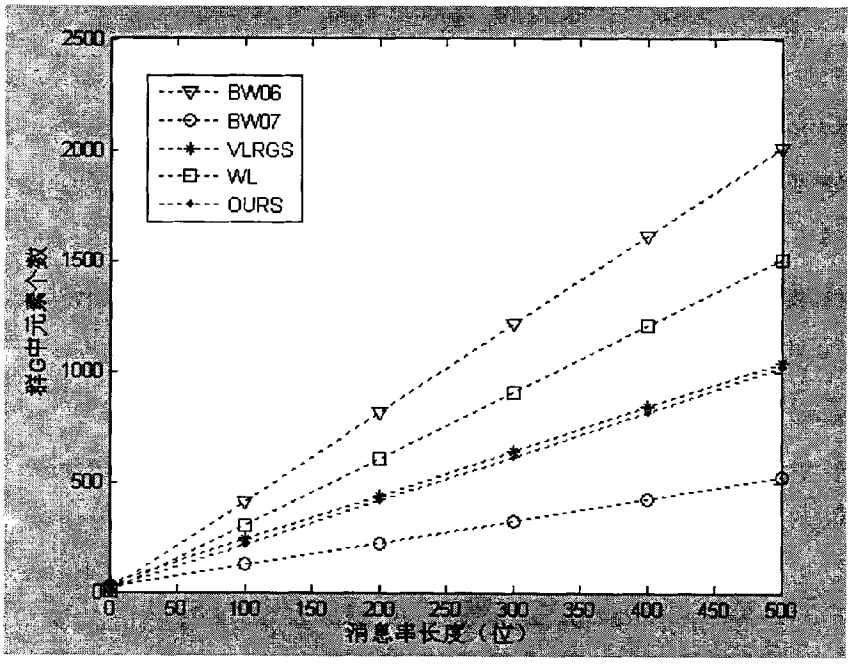

图3.2 通信代价

Fig. 3.2 Communication Costs

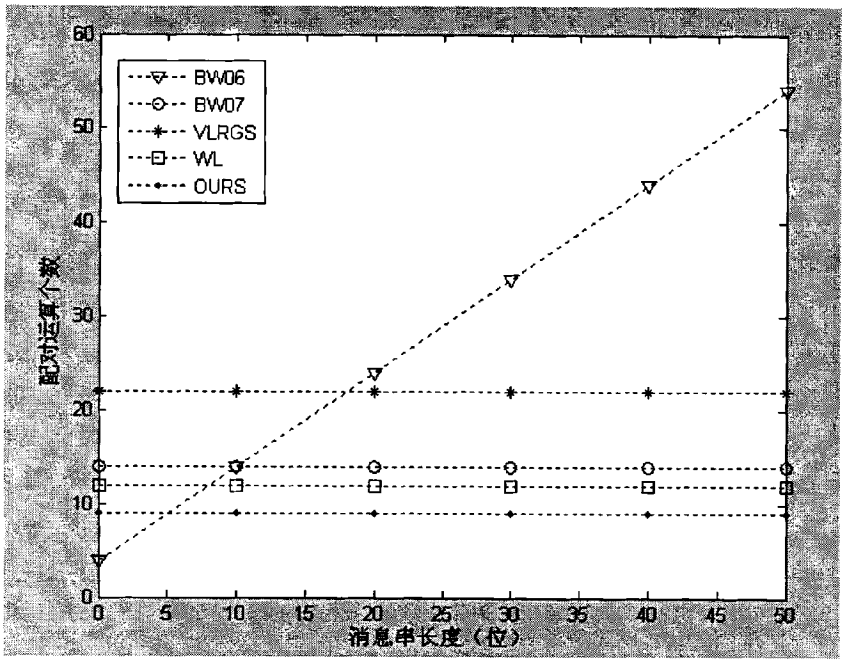

图3.3 计算代价

Fig. 3.2 Compution Costs

通过计算，本文提出的组签名方案的六个算法中共使用群中元素为 $2k+14$ 个（其中 k 为消息串的长度），随机数 12 个，幂运算 $k+26$ 个，配对运算 9 个。由于配对的计算较复杂，所以在效率分析上应该作为主要的考虑因素，通过表 3.2 的对比及图 3.3 可以发现，本方案所使用的配对个数是最少的。

此外将 GSCOBG 方案与同类的组签名方案进行了对比，通过上表可以得出以下结论：GSCOBG 方案在交互次数上优于交互式的方案[42]和[44],在通信代价上整体优于[33][43],在计算代价上优于非交互式组签名[33][34],在配对的计算量上优于方案[42][33][34][43][44]。

3.4 本章小结

本章内容是 GS 非交互式证明系统与组签名方案的结合。使用组合阶双线性群构建了一个基于 BMW 模型的高效组签名方案 GSCOBG，并在方案中引进了 Groth-Sahai 证明系统的思想。并对其安全性进行了严格的安全性证明，最后分别从安全性和效率两方面将该方案与同类组签名方案做出对比，从而得出该方案在效率和安全性上的优势。GSCOBG 方案解决了传统组签名的通信效率低、不能抵抗选择密文攻击等问题。此外，签名的大小是一个常量，不依赖于其他系统参数。同时也存在不足之处，比如没有提供完整的成员撤销机制等，在今后的工作中应该继续完善。

第4章 基于NIZK的BB+/CL+匿名签名方案

本章介绍GS证明系统的F-提取性，并利用这个重要性质构建非交互式BB+签名方案和非交互式CL+签名方案。并将非交互式方案与传统方案进行了安全性对比。匿名认证技术是近年来网络安全技术领域中涌现出的一个新领域，本章提出的非交互式签名方案可以应用到匿名认证方法中。

4.1 GS证明系统的F-可提取性

传统的知识证明系统[46]中，存在一个知识提取器，该提取器使用适当的陷门信息从一个可接受的证明中提取合法证据。在Groth-Sahai证明系统中承诺方证明的可提取性具备相同的性质。比如给定一个陷门信息，可以从承诺中提取出相关的$x_1,...,x_m$，$y_1,...,y_m$值。然而提取的这些值并不等于真实的证据值而是证据值的一部分，真实的证据值也包含承诺的打开值。对于群Z_p中的元素α，承诺须使用函数$GSExpCommit(params_{GS},h,\alpha)$，提取到的承诺对象为$g^{\alpha}$。形式化定义F-提取性如下：

定义4.1 可提取性：对于一个非交互式证明系统$(Setup,\Pr ove,VerifyProof)$，如果存在一个多项式时间内提取器$(ExtractSetup,Extract)$满足下列条件，则该系统是一个关于语言$L$的知识证明。

$$\{params \leftarrow Setup(1^k):params\} \approx \{(params,t) \leftarrow ExtractSetup:params\} \tag{4.1}$$

对于所有多项式时间内敌手$\mathcal{A}$，存在一个可忽略的函数v满足：

$$\begin{aligned}\Pr[(params,t) \leftarrow ExtractSetup;(s,\pi) \leftarrow \mathcal{A}(params);y \leftarrow Extract(params,t,s,\pi)\\ :Verify\Pr oof(params,x,\pi,M_L)=0 \vee (\exists xs.t.y=f(params,x)\\ \wedge M_L(params,s,x)=1)]=1-v(k)\end{aligned} \tag{4.2}$$

其中M_L是图灵机，当且仅当x是关于陈述$s \in L(params)$的证据时，图灵机接受$params,s,x$.

4.2 非交互式BB+签名方案

4.2.1 BB+签名方案

Ateniese等人在[46]中修改了WBB签名方案，称之为BB+方案，该方案中，系统参数为1^k，通过运行$BilinearSetup(1^k)$取得系统参数$params=(p,G_1,G_2,G_T,g,\tilde{g})$，群的阶均为素数$p$，$g$为$G_1$的生成元，$\tilde{g}$是$G_2$的生成元，双线性映射$e:G_1 \times G_2 \rightarrow G_T$。私钥为$sk \in Z_p^*$，公钥为$pk=(g,g^{sk},\tilde{g})$。

签名：签名者随机选择$r \in Z_p^*$，对消息$m \in Z_p^*$做签名，输出$\sigma=(g^r,\tilde{g}^{1/(sk+r)},\tilde{g}^{1/(m+r)})$。

验证：输入公钥 $pk=(g,g^{sk}\tilde{g})$，消息 $m\in Z_p^*$，签名 $\sigma=(A,B,C)=(g^r,\tilde{g}^{1/(sk+r)},\tilde{g}^{1/(m+r)})$，若验证等式 $e(g^{sk}A,B)=e(g,\tilde{g});e(Ag^m,C)=e(g,\tilde{g})$ 都成立，则验证通过。

4.2.2 改进的 BB+签名方案

Camenisch 等人定义了 CL 签名，同时也指出了匿名认证的结构。一个匿名认证方案包含两个协议。第一个协议叫做颁发协议 *Issue*，该协议使得用户获得在秘密消息的的签名，同时也保证消息的隐私性。用户希望获得签名者在 x 上的签名，但不希望签名者看到消息 x，用户将 x 承诺，等到承诺值 *comm*，然后将 *comm* 发给签名者，在 *Issue* 协议过后，用户获得了在 x 上的签名，但签名者不能获得关于 x 的任何信息。第二个协议叫做证明协议 *Prove*，这是一个关于在承诺对象 x 上的签名的零知识证明。示证者通过运行 *Issue* 协议获得签名对 $(x,\sigma_{pk}(x))$，同时示证者也拥有关于 x 承诺值 *comm*。验证者只知道承诺值 *comm*。示证者使用零知识理论证明他知道签名对 (x,σ) 满足 $VerifySig(pk,x,\sigma)=accept$，和 *open* 值满足 $comm=Commit(x,open)$。

Issue 协议使用两方计算协议[47]，*Prove* 协议使用零知识证明协议。同时使用了 *Pedersen* 承诺方案。这种结构在实际中是非常有用的，但 *CL* 签名方案是交互式的协议，证明需要有第三方来验证，下面本节根究 chase 等人的思想给出带有承诺签名的非交互式零知识证明，并使用 GS 证明系统中的承诺类型。

本节使用 GS 证明系统的 F-提取性和零知识性构建非交互式签名方案。GS 证明系统的一个重要特性就是 F-提取性，F-可提取性不同于完全可提取性。假设构建一个证明，x 中包含了 $a\in Z_p$，以及对 a 承诺的打开值。对于这个承诺，我们只能从证明中提取出 $b^a\in f(x)$。这样如果构建一个关于 (m,σ) 的知识证明，其中 $m\in Z_p$，并且验证通过 $VerifySig(pk,m,\sigma)=accept$，只能从证明中提取出关于 m 的函数 $F(m)$。即使不能伪造一个 (m,σ) 对，伪造一个 $(F(m),\sigma)$ 还是有可能的，因此在证明过程中需要定义签名的 F-不可伪造性，用于证明伪造一个 $(F(m),\sigma)$ 对是不可能的。本文基于 Chase 等人 P 签名的思想，将 BB+签名扩展为非交互式签名方案，传统的签名方案包含四个算法 $(Setup,Keygen,Sign,VerifySig)$，一个非交互式签名包含 $(Setup,Commit,ObtainSig,IssueSig,\mathrm{Prove},Verify\,\mathrm{Proof},EqComm\,\mathrm{Prove},VerEqComm)$ 算法。$Setup(1^k)$ 算法产生公共参数 *params*。参数包含签名方案所使用的参数和承诺方案所使用的参数。$ObtainSig(params,pk,m,comm,open)\leftrightarrow IssueSig(params,sk,comm)$ 这两个交互式算法用于在用户和颁发者之间执行签名的颁发协议。用户输入参数 $(params,pk,m,comm,open)$ 其中承诺值 $comm=Commit(params,m,open)$，用户最终得到签名 σ 作为输出。颁发者将 $(params,sk,comm)$ 作为输入，但不输出任何数据。

$\mathrm{Prove}(params, pk, m, \sigma)$ 算法输出 $(comm, \pi, open)$ 满足 $comm = Commit(params, m, open)$，其中 π 是关于签名 σ 在 m 上的知识证明。$Verify\,\mathrm{Proof}(params, pk, comm, \pi)$ 算法将 m 的承诺值 $comm$ 以及上述证明 π 作为输入，如果关于 $F(m)$ 的知识证明和在 m 上的签名合法，则接受证明，否则拒绝。

$EqComm\,\mathrm{Prove}(params, m, open, open')$ 算法将同一消息的两个承诺打开值作为输入，输出关于 $comm = Commit(m, open)$ 和 $comm' = Commit(m, open')$ 是同一个承诺对象的承诺值的证明 π。

$VerEqComm(params, comm, comm', \pi)$ 算法将两个承诺值和一个证明作为输入，如果上述证明 π 是关于 $comm$ 和 $comm'$ 是同一个承诺对象的承诺值的证明，则接受该证明。

下面将 BB+签名方案扩展成基于 GS 证明系统的非交互式匿名签名方案：

$Setup(1^k)$：首先获得系统参数 $params = (p, G_1, G_2 G_T e, g, \tilde{g}) \leftarrow BilinearSetup(1^k)$，然后产生 GS 系统的参数 $params_{GS} \leftarrow GSSetup(params)$，其中 g 为 G_1 的生成元，$\tilde{g}$ 是 G_2 的生成元，私钥为 $sk \in Z_p^*$，公钥为 $pk = (g, g^{sk}, \tilde{g})$。

$Commit(params, m, open)$：若对 m 承诺，计算 $C = GSExpCommit(params_{GS}, \tilde{g}, m, open)$ 其中 $GSExpCommit(params_{GS}, \tilde{g}, m, open) = GSCommit(params_{GS}, \tilde{g}^m, open)$。

$ObtainSig(params, pk, m, comm, open) \leftrightarrow IssueSig(params, sk, comm)$。用户和颁发者运行下列协议：

（1）用户选择随机数 $r_1, r_2 \leftarrow Z_p$。

（2）颁发者选择随机数 $r' \leftarrow Z_p$。

（3）用户和颁发者运行一个安全的两方计算协议，用户秘密的输入参数 $(r_1, r_2, m, open)$，颁发者秘密地输入参数 (r', sk)。颁发者得到得输出为 $x_1 = (sk + r_1 r')r_2$，$x_2 = (m + r_1 r')r_2$。

（4）颁发者计算 $C_1' = g^{1/x_1}$，$C_2' = g^{1/x_2}$，$C_3' = g^{r'}$，并将 (C_1', C_2', C_3') 发给用户。

（5）用户计算 $C_1 = C_1'^{r_2}$，$C_2 = C_2'^{r_2}$，$C_3 = C_3'^{r_1}$，并验证签名 (C_1, C_2, C_3) 的合法性。

$\mathrm{Prove}(params, pk, m, \sigma)$：用户计算承诺值 $R_1 = GSCommit(params_{GS}, C_1, open_1)$，$R_2 = GSCommit(params_{GS}, C_2, open_2)$，$R_3 = GSCommit(params_{GS}, C_3, open_3)$，$R_g = GSExpCommit(params_{GS}, g, m, open_4) = GSCommit(params_{GS}, g^m, open_4)$，$R_u = GSExpCommit(params_{GS}, u, m, open_5) = GSCommit(params_{GS}, u^m, open_5)$，用户输出下列证明：$\pi = NIZK\{((R_1 : C_1)(R_2 : C_2)(R_3 : C_3)(R_g : a)(R_u : b)) :$

$e(pkC_3, C_2) = e(g, \tilde{g}) \wedge e(C_3 a, C_1) = e(g, \tilde{g}) \wedge e(a, u) = e(b, g)\}$

$Verify\,\mathrm{Proof}(params, pk, comm, \pi)$：如果证明 π 是关于陈述的合法证明则接受，否则拒绝。

$EqComm\,\mathrm{Pr}ove(params,m,open,open')$：GS证明系统中用于证明两个相同承诺对象的不同承诺值 $comm=Commit(params_{GS},m,open)=GSCommit(params_{GS},g^m,open)$，$comm'=Commit(params_{GS},m,open')=GSCommit(params_{GS},g^m,open')$，使用GS证明系统产生证明 $\pi\leftarrow NIZK\{((comm:a),(comm':b)):a=b\}$。

$VerEqComm(params,comm,comm',\pi)$：用于验证上述证明的正确性。

4.2.3 安全性证明

定义4.2 若BB+是一个 F-不可伪造的签名方案（其中 F 为双射），（*Setup*, *Commit*）是一个具有完美绑定性，强计算隐藏性的承诺方案，并且（$Setup, EqCommProve, VerEqComm$）是一个非交互式的证明方案，则当非交互式签名方案满足正确性，签名者隐私性，用户隐私性，不可伪造性和零知识性五个条件时，称一个非交互式签名方案是安全的。

（一）正确性：对于诚实的用户从诚实的颁发者所获得的签名，诚实的验证者总能验证签名的有效性。形式化表示如下：

$$\forall M\in\{0,1\}^*:\Pr[params\leftarrow Setup(1^k);(pk,sk)\leftarrow KeyGen(params);\sigma\leftarrow Sign(params,sk,m);$$
$$(comm,\pi)\leftarrow Prove(params,pk,M,\sigma):Verifyproof(params,pk,comm,\pi)=1]=1 \quad (4.3)$$

证明：显然对于诚实的用户从诚实的颁发者所获得的BB+签名，诚实的验证者总能验证签名的有效性。

（二）签名者隐私性：*PPTM* 敌手不能有效区分与他交互的是真正的签名颁发者还是模拟机。形式化表述为：对于 *PPTM* 敌手 (A_1,A_2)，存在模拟算法 *SimIssue* 和可忽略的函数 v 满足：

$$\begin{aligned}&|\Pr[params\leftarrow Setup(1^k);(sk,pk)\leftarrow KeyGen(params);(M,open,state)\leftarrow A_1(params,sk);\\&comm\leftarrow Commit(params,M,open);b\leftarrow A_2(state)\leftrightarrow IssueSig(params,sk,comm):b=1]\\&-\Pr[params\leftarrow Setup(1^k);(sk,pk)\leftarrow KeyGen(params);(M,open,state)\leftarrow A_1(params,sk);\\&comm\leftarrow Commit(params,M,open);\sigma\leftarrow Sign(params,sk,M);\\&b\leftarrow A_2(state)\leftrightarrow SimIssue(params,comm,\sigma):b=1]|<v(k)\end{aligned} \quad (4.4)$$

对于敌手选择的所有 M 和 *open*，*IssueSig* 和 *SimIssue* 均能给出诚实的承诺值。

证明：定义模拟算法 $SimIssue(params,comm,\sigma)$ 来生成模拟器 *Simulator* 对两方计算协议进行模拟。在协议的进行过程中，*Simulator* 可以提取敌手的输入 $(r_1,r_2,M,open)$，然后 *SimIssue* 验证等式 $comm=Commit(params,M,open)$ 的正确性。（对于敌手选择的 M 和 *open*，算法能够保证 *IssueSig* 和 *SimIssue* 都会产生诚实的承诺值，即不存在算法对承诺的混淆情况）若验证未通过，则终止协议，否则发送给敌手 $C_1'=g^{1/x_1}$，$C_2'=g^{1/x_2}$，$C_3'=g^{r'}$。但实际上 *IssueSig* 和 *SimIssue* 不可区分，故敌手无法识别出与他交互的是

Simulator 还是真正的颁发者。一旦敌手的输入值没有通过验证，协议将终止与敌手的交互，从而保证了签名者的隐私性。

（三）用户隐私性：*PPTM* 敌手 $(\mathcal{A}_1,\mathcal{A}_2)$ 不能有效区分与他交互的是签名获得算法还是模拟机. 形式化表述为：对于 *PPTM* 敌手 $(\mathcal{A}_1,\mathcal{A}_2)$，存在模拟算法 *SimObtain* 和可忽略的函数 v 满足：

$$
\begin{aligned}
&|\Pr[params \leftarrow Setup(1^k);(pk,M,open,state) \leftarrow \mathcal{A}_1(params);comm = Commit(params,M,open);\\
&b \leftarrow \mathcal{A}_2(state) \leftrightarrow ObtainSig(params,pk,M,comm,open):b=1]\\
&-\Pr[(params,sim) \leftarrow Setup(1^k);(pk,M,open,state) \leftarrow \mathcal{A}_1(params);\\
&comm = Commit(params,M,open);b \leftarrow \mathcal{A}_2(state) \leftrightarrow SimObtain(params,pk,comm)\\
&:b=1]|<v(k)
\end{aligned}
\tag{4.5}
$$

定义模拟算法 *SimObtain*$(params,sk,comm)$ 来生成模拟器 *Simulator* 对两方计算协议进行模拟。在协议的进行过程中，*Simulator* 可以提取敌手的输入信息 sk'（此信息没有必要为合法的私钥），然后 *Simulator* 给出随机值 x 与敌手进行交互。但实际上 *SimObtain* 与 *ObtainSig* 无法区分，敌手无法识别出与他交互的是 *Simulator* 还是诚实的用户，故他无法决定是否应该中止协议的进行。即使敌手能够完成两方计算协议，他也只能得到由 *Simulator* 随机选择的 x，而非真正由私钥计算得出的证书值，从而保证了用户的隐私性。

（四）不可伪造性：若 *PPTM* 敌手没有从预言机获得消息 M 的签名或证明，则他无法伪造关于 M 的证明。具体来说，一个不可伪造的方案需满足下列条件：存在提取算法 $(ExtractSetup,Extract)$ 和双射 F 满足（1）$ExtractSetup(1^k)$ 和 $Setup(1^k)$ 不可区分（2）*PPTM* 敌手无法输出合法的证明 π，并且对于从 $F(m)$ 中提取出的 σ 满足下列条件之一：（1）σ 不是关于 M 的合法签名（2）$comm$ 不是 M 的合法承诺（3）敌手之前没有对 M 的签名预言进行过询问。形式化表述为：对于 *PPTM* 敌手 A，存在可忽略的函数 v 满足：

$$
\begin{aligned}
&\Pr[params_0 \leftarrow Setup(1^k);(params_1,td) \leftarrow ExtractSetup(1^k):\\
&b \leftarrow \{0,1\}:A(params_b)=b] < 1/2 + v(k), and\\
&\Pr[(params,td) \leftarrow ExtractSetup(1^k);(pk,sk) \leftarrow KeyGen(params);\\
&(Q_{Sign},comm,\pi) \leftarrow A(params,pk)^{O_{Sign}(params,sk,\cdot)};(y,\sigma) \leftarrow Extract(params,td,\pi,comm):\\
&VerifyProof(params,pk,comm,\pi)=accept\\
&\wedge(VerifySig(params,pk,F^{-1}(y),\sigma)=reject \vee (\forall open,comm \neq Commit(params,F^{-1}(y),open))\\
&\vee(VerifySig(params,pk,F^{-1}(y),\sigma)=accept \wedge y \notin F(Q_{Sign})))] < v(k).
\end{aligned}
\tag{4.6}
$$

预言 $O_{Sign}(params,sk,m)$ 运行 $Sign(params,sk,m)$ 并将签名结果 σ 返回给敌手，并用 Q_{Sign} 记录敌手的询问信息。

证明：考虑如下算法：$ExtractSetup(1^k)$ 输出 *params*，并调用 *GSExtractSetup* 得到可

供选择的 $params_{GS}$ 和陷门 $td=(td_1,td_2)$，(td_1,td_2) 分别用于从 G_1 和 G_2 提取 GS 承诺。由于 *GSSetup* 和 *GSExtractSetup* 的输出参数不可区分，从而得出 *ExtractSetup* 和 *Setup* 的输出参数也是不可区分的。$Extract(params,td,comm,\pi)$ 从承诺值 *comm* 和 $C_{u'}$ 中提取信息，而 C_g 包含在证明 π 中。若 *VerifyProof* 验证通过则有 $comm=C_{u'}$，定义 $F(M)=(g^m,u^m)$。

下面利用反证法证明方案的不可伪造性。假设敌手能够攻破签名方案的不可伪造性，则伪造可以攻破非随机预言下的CDH假设。但由于IBE签名方案在非随机预言的CDH假设下是安全的，故敌手无法伪造成功。具体证明过程如下：

敌手可以从 $(F(M),\sigma)$ 的提取值中输出一个证明，满足下列三个情况之一：（1）$VerifySig(params,pk,M,\sigma)=reject$，（2）*comm* 不是 M 的承诺，（3）敌手没有对 M 进行过询问。由于 *VerifyProof* 运行了一系列的配对乘积等式并且GS证明系统具有F-提取性，故情况（1）不会发生。由于 *VerifyProof* 验证了 $comm=C_{u'}$ 的正确性，故情况（2）不会发生。因此只需考虑情况（3）：提取器选取新鲜的 m 并计算 $F(M)=(g^m,u^m)$，由于方案中的 r 值是随机选取的，故证明中提取出的 $(g^r,\tilde{g}^{1/(sk+r)},\tilde{g}^{1/(m+r)},\ g^m,u^m,g^{sk})$ 满足 $m+rsk\neq m_i+r_isk$，其中 (m_i,r_i) 用于回答敌手的签名和证明询问。最终敌手根据询问输出证明 π，但由于敌手的信息是伪造的，故 $(g^m,u^m)\notin F(Q_{Sign})$，从而敌手攻破了CDH假设。但实际上方案在CDH下是安全的，故敌手无法伪造成功。

（五）零知识性：存在模拟器 $Sim=(SimSetup,SimProve,SimEqComm)$，其中 *SimSetup* 生成的参数能够以可忽略的函数 v 隐藏承诺值。具体表述为（1）*SimSetup* 与 *Setup* 生成的参数不可区分且 *SimSetup* 生成附加串 *sim*。（2）$SimProve(params,sim,sk)$ 与 $Prove(params,pk,M,\sigma)$ 无法区分，其中 *params* 由 *SimSetup* 生成。（3）$SimEqComm(params,sim,comm,comm')$ 与 $EqCommProve(params,M,open,open')$ 不可区分，其中 *params* 由 *SimSetup* 生成且 $comm=Commit(params,M,open)$，$comm'=Commit(params,M,open')$。上述过程可形式化定义为：

$$
\begin{aligned}
&|\Pr[params\leftarrow Setup(1^k);b\leftarrow A(params):b=1]\\
&-\Pr[(params,sim)\leftarrow SimSetup(1^k);b\leftarrow A(params):b=1]|<v(k),and\\
&|\Pr[(params,sim)\leftarrow SimSetup(1^k);(pk,M,\sigma,state)\leftarrow A_1(params,sim);\\
&(comm,\pi,open)\leftarrow Prove(params,pk,M,\sigma);b\leftarrow A_2(state,comm,\pi):b=1]\\
&-\Pr[(params,sim)\leftarrow SimSetup(1^k);(pk,M,\sigma,state)\leftarrow A_1(params,sim);\\
&(comm,\pi)\leftarrow SimProve(params,sim,pk);b\leftarrow A_2(state,comm,\pi):b=1]|<v(k),and\\
&|\Pr[(params,sim)\leftarrow SimSetup(1^k);(M,open,open')\leftarrow A_1(params,sim);\\
&\pi\leftarrow EqCommProve(params,M,open,open');b\leftarrow A_2(state,\pi):b=1]\\
&-\Pr[(params,sim)\leftarrow SimSetup(1^k);(M,open,open')\leftarrow A_1(params,sim);\\
&\pi\leftarrow SimEqComm(params,sim,comm,comm');b\leftarrow A_2(state,\pi):b=1]|<v(k)
\end{aligned}
\qquad (4.7)
$$

证明：考虑如下算法*SimSetup*输出参数*params*和*sim*，其中*params*与*Setup*生成的参数不可区分。*SimProve*输入*params*，*sim*和公钥$pk=(g,\tilde{g},g^{sk})$，并利用陷门*sim*来生成一个随机的伪造证明，具体过程如下：选择$s\leftarrow\mathbb{Z}_p$，计算$\sigma=g^{1/s}$，设$m=s-sk$，并计算$g^m=g^s/g^{sk},u^m=(u^s/g^{sk})^t$。接下来敌手可按照*Prove*协议并利用证据$\sigma,g^m,u^m,g^s,u^s$来生成一个伪造的证明$\pi$。根据GS证明系统的不可区分性，由伪造的证据生成的证明与真实的证据生成的证明不可区分。因此，*SimProve*和*Prove*是不可区分的，从而保证了方案的零知识性。

如表4.1所示，改进的非交互式BB+方案相对于传统的BB+签名方案增加了一些新性质，如非交互性、抗CCA、零知识性、签名者匿名性等，从而实现了以匿名方式证明自身身份的功能，即匿名认证。

表4.1 安全性比较

Table 4.1 Comparison of the security

安全性 \ 方案	BB+	非交互式BB+
匿名性	×	√
正确性	√	√
零知识性	×	√
抗CCA	×	√
非交互性	×	√
不可伪造性	√	√

4.3 非交互式CL+签名方案

4.3.1 CL+签名方案

该方案中，系统参数为1^k，通过运行*BilinearSetup*(1^k)取得系统参数$params=(p,G_1,G_2,G_T,g,\tilde{g})$，群的阶均为$p$，其中$g$为$G_1$的生成元，$\tilde{g}$是$G_2$的生成元，双线性映射$e:G_1\times G_2\rightarrow G_T$。私钥为$sk=(s,t)\in Z_p^{\ 2}$，公钥为$pk=(h^s,h^t)\in Z_p^{\ 2}$。

（1）签名：签名者随机选择$a\in G_1$，对消息$m\in Z_p^{\ *}$做出签名，输出$\delta=(a,a^t,a^{s+stm},a^m,a^{mt})$。

（2）验证：输入$\delta=(A,B,C,D,E)=(a,a^t,a^{s+stm},a^m,a^{mt})$，当且仅当下列等式$e(B,\tilde{g})=e(A,\tilde{g}^t),e(D,\tilde{g}^t)=e(E,\tilde{g}),e(C,\tilde{g})=e(A,\tilde{g}^s)e(E,\tilde{g}^s)$均成立，可以通过$\log_A D$恢复出消息$m$。

（3）随机性：对于签名 (A,B,C,D,E) 选择一个随机数 $r \in Z_p^*$ 将签名随机化，得到的 (A',B',C',D',E') 仍然满足上述验证等式。

4.3.2 改进的 CL+签名方案

改进的签名方案参数产生过程与 CL+签名的参数类似，在原有基础上嵌入了 GS 证明系统的参数产生过程。

$Setup(1^k)$：首先获得系统参数 $params=(p,G_1,G_2G_Te,g,\tilde{g}) \leftarrow BilinearSetup(1^k)$，然后产生 GS 系统的参数 $params_{GS} \leftarrow GSSetup(params)$，其中 g 为 G_1 的生成元，$\tilde{g}$ 是 G_2 的生成元，私钥为 $sk=(s,t) \in Z_p^{\ 2}$，公钥为 $pk=(h^s,h^t) \in Z_p^{\ 2}$。

$Commit(params,m,open)$：若对 m 承诺，计算 $C=GSExpCommit(params_{GS},\tilde{g},m,open)$ 其中 $GSExpCommit(params_{GS},\tilde{g},m,open)=GSCommit(params_{GS},\tilde{g}^m,open)$。

$ObtainSig(params,pk,m,comm,open) \leftrightarrow IssueSig(params,sk,comm)$。用户和颁发者运行下列协议：

（1）用户选择随机数 $r_1 \leftarrow Z_p$。

（2）颁发者选择随机数 $r_2 \leftarrow Z_p$。

（3）用户和颁发者运行一个安全的两方计算协议，用户秘密的输入参数 $(r_1,m,open)$，颁发者秘密地输入参数 (r_2,sk)。颁发者得输出为 $A=a=g^{1/(t+r_2)}$，$C=a^{s+stm}$。

（4）颁发者计算 $B=a^t$，并将 (A,B,C) 发给用户。

（5）用户计算 $D=a^m$，$E=B^m$，并验证签名 (A,B,C,D,E) 的合法性。

$\mathrm{Prove}(params,pk,m,\sigma)$：用户计算承诺值 $R_1=GSCommit(params_{GS},A,open_1)$，$R_2=GSCommit(params_{GS},B,open_2)$，$R_3=GSCommit(params_{GS},C,open_3)$，$R_4=GSCommit(params_{GS},D,open_4)$，$R_5=GSCommit(params_{GS},E,open_5)$，$R_a=GSExpCommit(params_{GS},a,m,open_6)=GSCommit(params_{GS},a^m,open_6)$，$R_u=GSExpCommit(params_{GS},u,m,open_7)=GSCommit(params_{GS},u^m,open_7)$，用户输出下列证明：$\pi=NIZK\{((R_1:A)(R_2:B)(R_3:C)(R_4:D)(R_5:E)(R_a:c)(R_u:b)):$
$e(B,\tilde{g})=e(A,\tilde{g}^t) \wedge e(D,\tilde{g}^t)=e(E,\tilde{g}) \wedge e(C,\tilde{g})=e(A,\tilde{g}^s)e(E,\tilde{g}^s) \wedge e(u,c)=e(b,a)\}$

$Verify\,\mathrm{Proof}(params,pk,comm,\pi)$：如果证明 π 是关于陈述的合法证明则接受。

$EqComm\,\mathrm{Prove}(params,m,open,open')$：GS 证明系统中用于证明两个相同承诺对象的不同承诺值 $comm=Commit(params_{GS},m,open)=GSCommit(params_{GS},a^m,open)$，$comm'=Commit(params_{GS},m,open')=GSCommit(params_{GS},u^m,open')$，使用 GS 证明系统产生证明 $\pi \leftarrow NIZK\{((comm:a),(comm':b)):a=b\}$。

$VerEqComm(params,comm,comm',\pi)$：用于验证上述证明的正确性。

4.3.3 安全性证明

下面将证明本节提出的非交互式签名方案满足正确性，签名者隐私性，用户隐私性，不可伪造性和零知识性，从而该签名方案是安全的。

（一）正确性：显然对于诚实的用户从诚实的颁发者所获得的签名，诚实的验证者总能验证签名的有效性。

（二）签名者隐私性：定义模拟算法 $SimIssue(params,comm,\sigma)$ 来生成模拟器 *Simulator* 对两方计算协议进行模拟。在协议的进行过程中，*Simulator* 可以提取敌手的输入 $(r_1,r_2,M,open)$，然后 *SimIssue* 验证等式 $comm=Commit(params,M,open)$ 的正确性。（注意到对于敌手选择的 M 和 $open$，算法能够保证 *IssueSig* 和 *SimIssue* 都会产生诚实的承诺值，即不存在算法对承诺的混淆情况）若验证未通过，则终止协议，否则发送给敌手 A,B,C。但实际上 *IssueSig* 和 *SimIssue* 不可区分，故敌手无法识别出与他交互的是 *Simulator* 还是真正的颁发者。一旦敌手的输入值没有通过验证，协议将终止与敌手的交互，从而保证了签名者的隐私性。

（三）用户隐私性：定义模拟算法 $SimObtain(params,sk,comm)$ 来成模拟器 *Simulator* 对方计算协议进行模拟。在协议的进行过程中，*Simulator* 可以提取敌手的输入信息 $g_2^{\alpha'}$（此信息没有必要为合法的私钥），然后 *Simulator* 给出随机值 x 与敌手进行交互。但实际上 *SimObtain* 与 *ObtainSig* 不可区分，敌手无法识别出与他交互的是 *Simulator* 还是诚实的用户，故他无法决定是否应该中止协议的进行。即使敌手能够完成两方计算协议，他也只能得到由 *Simulator* 随机选择的 x，而非真正由私钥计算的出的 $A=a=g^{1/(r_1+r_2)}$，$C=a^{s+stm}$，从而保证了用户的隐私性。

（四）不可伪造性：考虑如下算法：$ExtractSetup(1^k)$ 输出 *params*，并调用 *GSExtractSetup* 得到可供选择的 $params_{GS}$ 和陷门 $td=(td_1,td_2)$，(td_1,td_2) 分别用于从 G_1 和 G_2 提取 GS 承诺。由于 *GSSetup* 和 *GSExtractSetup* 的输出参数不可区分，从而得出 *ExtractSetup* 和 *Setup* 的输出参数也是不可区分的。$Extract(params,td,comm,\pi)$ 从承诺值 *comm* 和 $C_{u'}$ 中提取信息，而 C_g 包含在证明 π 中。若 *VerifyProof* 验证通过则有 $comm=C_{u'}$，定义 $F(M)=(a^m,u^m)$。

下面利用反证法证明方案的不可伪造性。假设敌手能够攻破签名方案的不可伪造性，由此伪造可以攻破非随机预言下的CDH假设。但由于IBE签名方案在非随机预言的CDH假设下是安全的，故敌手无法伪造成功。具体证明过程如下：

敌手可以从 $(F(M),\sigma)$ 的提取值中输出一个证明，满足下列三个情况之一：（1）$VerifySig(params,pk,M,\sigma)=reject$，（2）*comm* 不是 M 的承诺，（3）敌手没有对 M 进行过询问。由于 *VerifyProof* 运行了一系列的配对乘积等式并且GS证明系统具有F-可提取

性，故情况（1）不会发生。由于*VerifyProof*验证了$comm=C_{u'}$的正确性，故情况（2）不会发生。因此只需考虑情况（3）提取器选取新鲜的M并计算$F(M)=(a^m,u^m)$，由于方案中的r_1,r_2值是随机选取的，故证明中提取出的(A,B,C,D,E)满足$stm+s\neq stm_i+s$，其中m_i用于回答敌手的签名和证明询问。最终敌手根据询问输出证明π，但由于敌手的信息是伪造的，故$(a^m,u^m)\notin F(Q_{Sign})$，从而敌手攻破了CDH假设。但实际上方案在CDH下是安全的，故敌手无法伪造成功。

（五）零知识性：考虑如下算法：*SimSetup*输出参数*params*和*sim*，其中*params*与*Setup*生成的参数不可区分。*SimProve*输入*params*，*sim*和公钥$pk=(h^s,h^t)\in Z_p^2$，并利用陷门*sim*来生成一个随机的伪造证明，具体过程如下：选择$r\leftarrow\mathbb{Z}_p$，计算$\sigma=g^{1/r}$，设$m=(r-s)/st$，并计算$a^m=a^{r/st}/a^{s/st},u^m=(u^{r/st}/a^{s/st})^t$。接下来敌手可按照*Prove*协议并利用证据$\sigma,a^m,u^m,g^r,u^r$来生成一个伪造的证明$\pi$。

根据GS证明系统的不可区分性，由伪造的证据生成的证明与真实的证据生成的证明不可区分。因此，*SimProve*和*Prove*是不可区分的，从而保证了方案的零知识性。

如表4.2所示，传统的CL+签名方案除了具有正确性、不可伪造性之外还具有很好的随机性，即若将签名随机化，则随机化以后得到的签名仍然满足验证等式。改进的非交互式CL+方案仍然具备这一特性，除此之外还增加了一些新性质，如非交互性、抗CCA、零知识性、签名者匿名性等，从而实现了以匿名方式证明自身身份的功能，即匿名认证。

表4.2 安全性比较
Table 4.2 Comparison of the security

安全性 \ 方案	CL+	非交互式 CL+
匿名性	×	√
正确性	√	√
零知识性	×	√
抗 CCA	×	√
非交互性	×	√
不可伪造性	√	√
随机性	√	√

4.4 基于非交互式签名的匿名认证

传统的实名认证可以通过下列实例说明，从ATM提款机取款，必须有某个可以使用的现金卡，此外还必须输入提款密码。银行是根据用户的帐号和密码来判断取款用户的真实身份。再比如某人声称自己是张三，则可以通过查看其身份证来检验他的陈述。像这种在认证的同时鉴别了对方的实际身份的认证，被称为实名认证或非匿名认证。那么，自然考虑到能否在不确认对方具体身份的前提下进行认证，现实中这样的例子并不少见，比如无记名投票。由于没有记名，开票者不知道具体的投票者的具体身份。不过这有一个条件，无记名投票一般都是在封闭的场所进行，在默认被监管的条件下进行投票和开票。例如某俱乐部的专用停车场关卡，车主要想进入停车，就必须向关卡验证身份，以证实自己确实是俱乐部的注册会员。但车主在验证身份的同时会面临个人隐私泄露的危险。因此车主往往希望通过验证时，关卡只知道自己是会员，但个人的身份信息不会泄漏。只有在特殊情况下，如停车场内有犯罪事件发生时，通过关卡的车主的身份才能由第三方来揭示。在这两种情况下，认证机构在信任对方并赋予对方权利的时候并不知道对方的具体身份。匿名认证技术正是源自这样一种需求，被认证者要求某种权利，而同时不提供具体的个人信息。在这类匿名认证方案中，一方通过公认的第三方来向验证方证明自己身份的合法性，而不用泄漏自己的身份，且有的认证过程具有相关性，即验证者可以判断两次认证是否是由同一个用户进行的；有的则不具有非相关性，即多次认证之间是不相关的。但有时被认证者希望可以自行控制认证的相关性。

Alice向Bob证明自己拥有Carol所颁发的证书，而Bob在不联系Carol的情况下，相信Alice的证明，同时Alice提供的证明和自身的身份是不相关联的。本节提到的匿名认证要求即多次认证之间是不相关的。

如何做到上述的匿名认证呢？如果存在PKI允许示证者注册公钥，则Alice注册笔名A_B和A_C分别用于与Bob和Carol通信。其中A_B和A_C都是关于Alice私钥的承诺，根据承诺方案的不可连接性，可知A_B和A_C是不相关联的。若Alice要从Carol获得一个证书并且向Bob证实证书的存在，则Alice首先用A_C的身份与C运行颁发协议Issue，得到Carol在其私钥上的签名，之后再使用A_B的身份通过NIZK证明向Bob证明自己拥有Carol的签名。

一个非交互式的匿名认证系统包含如下算法：*Setup*算法产生系统参数，同时所有用户和机构都产生公私钥对(pk,sk)并向注册中心CA注册各自的公钥。注册阶段的结果是用户最终获得了CA所颁发的根证书C_{CA}。笔名注册算法*Pseudonym registration*。某一用户和某一机构协商为用户产生一个笔名N，用户输入他的公私钥对(pk,sk)和根证书

C_{CA}，他们的公共输出是 N，用户的秘密输出是 $aux(N)$。证书颁发算法 *Credential issue*，用户基于自身的笔名获得一个来自某一组织的证书，同时不暴漏真实身份信息。该算法中用户 U 秘密输入 (pk_U, sk_U, aux_N)，机构秘密输入自己私钥 sk_O。用户得到秘密的证书 C。证书证明算法 *Proof of possession of a credential*，该算法中，机构 O_1 通过笔名 N_1，机构 O_2 通过笔名 N_2 分别于用户建立联系，用户向机构 O_2 证明他拥有来自机构 O_1 的证书 C_1。用户秘密的输入信息 $(sk_U, pk_U, P_1, aux_{N_1}, aux_{N_2}, C_1)$。

下面利用上述的非交互式签名方案构建非交互式匿名认证系统。

初始化：首先获得系统参数 $params = (p, G_1, G_2 G_T e, g, \tilde{g}) \leftarrow BilinearSetup(1^k)$，GS 系统的参数 $params_{GS} \leftarrow GSSetup(params)$，以及承诺算法的参数。用户的私钥 sk_U 从签名方案的消息空间中选择，根据私钥以及适当的公钥算法定义用户的公钥 $pk_U = PublicKey(sk_U)$。

认证中心及其他用户包括验证者需要根据非交互式签名算法产生各自的密钥对。系统中所有用户都需要向认证中心申请一个合法证书，证书产生过程如下：

（1）用户跟 *CA* 协商一个笔名 $N_{CA} = Commit(params, sk_U, open)$，因为承诺方案具有完美绑定性，自然保证了用户身份和笔名的绑定性。

（2）用户提供一个不可否认的交互式证明，用来证明和公钥 $pk_U = PublicKey(sk_U)$ 对应的私钥的承诺值。

（3）所有用户和 *CA* 运行获得证书的协议，用户获得一个 *CA* 在其私钥的承诺值上的签名。该步骤中运行非交互式签名的 *ObtainSig* 和 *IssueSig* 算法。

笔名注册：用户通过将私钥进行承诺来形成与某一组织通信所使用的笔名 $N = Commit(params, sk_U, open)$。用户向组织 O 证明自己拥有来自认证中心 *CA* 的证书。证书合法后，示证者（用户）将 $N = Commit(params, sk_U, open)$ 发送给验证者，并用零知识证明的方式证明承诺值 N 是对自己私钥的承诺，最终用户得到与验证者协商的笔名 $aux(N) = open$。

证书颁发：用户 U 和某一组织 O 运行非交互式签名中的 *ObtainSig* 和 *IssueSig* 算法，用户的输入为 $(params, pk_O, sk_U, N, aux(N))$，组织 O 的输入为 $(params, sk_O, N)$，最终用户获得了某一组织 O 在自己私钥 sk_U 上的签名 σ 作为自身的证书 C。

证书的证明：用户拥有某一组织所颁发的证书 $C = \sigma_{O_1}(sk_U)$，他需要向验证者 O_2 证明证书 $C = \sigma_{O_1}(sk_U)$ 来自组织 O_1，倘若 O_2 与用户协商的笔名为 N，用户与颁发者运行非交互式签名方案中的证明算法，证明用户拥有来自 O_1 的证书。证明过程如下：

（1）计算证明 $(comm, \pi_1, open) \leftarrow \Pr ove(params, pk_{O_1}, sk_U, C)$

（2）计算$\pi_2 \leftarrow EqComm\,\mathrm{Pr}ove(params, sk_U, open, aux(N))$（其中$EqComm\,\mathrm{Pr}ove$是关于两个承诺值comm，$N$是针对同一承诺对象的非交互式证明）

（3）输出$(N, comm, \pi_1, \pi_2)$

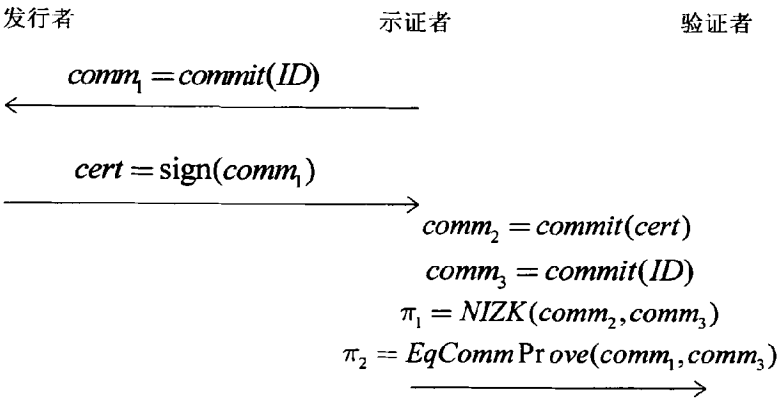

图4.1 基于非交互式签名的匿名认证
Fig. 4.1 Anonymous Credentials based on non-interactive signatures

4.5 本章小结

本章主要内容为GS证明系统与普通的签名方案的结合，本章利用GS证明系统以及相关性质，分别构建了基于BB+签名方案和CL+签名方案的非交互式签名方案。对各自的安全性进行证明，并与传统方案进行安全性对比，最后阐述了非交互式签名与匿名认证方法的关系。

第5章 基于非交互式签名的匿名电子拍卖协议

拍卖具有很好的发展前景，从个人物品拍卖到房产拍卖、工作机会拍卖、再到大型企业的投标招标方案。随着网络技术的发展和普及，越来越多的用户使用网络完成拍卖活动，从传统的拍卖方案转移到使用电子拍卖，电子拍卖方案有经济性、高效性、匿名性，而这些特性补充了传统的拍卖方案的不足。本章将非交互式签名应用到电子拍卖协议中，达到了交互次数减少，效率提高的目的。同时满足了电子拍卖协议所要求的匿名性、可追踪性、不可伪造性、可验证性、不同拍卖间的无关联性。

5.1 电子拍卖的优势及类型

电子拍卖与传统拍卖相比具有很多优势，为用户提供了很多便利条件，从而渐渐取代了传统拍卖的市场，现总结电子拍卖的优势如下：

（1）所有使用者即可以是买家也可以是卖家，都可以自由参加拍卖活动。

（2）不受时间和距离或者地点的限制或影响，投标者可以随时随地参与拍卖活动。

（3）拍卖过程统一控制管理。拍卖过程容易控制，无须较多人员参与管理，减少成本，同时避免了现场拍卖的混乱问题和人为失误造成错误判定等问题。

（4）不需要实体店面，减少了拍卖商和投标者的精力和成本。

（5）拍卖商品多样化，新商品、二手商品等任何种类任何价格的商品均可以在电子拍卖中出现。

拍卖是一种用来决定产品价格的交易，一般用于商品的价值无法客观确定的场合，例如：古董、艺术品、二手商品等。比如市场中存在一位卖者和一群买者进行交易，卖者不确定商品的价格，于是召集一些买者进行竞价，由此来确定该商品的价格。该交易中包含许多协商的规则，这些规则根据商品和参与者的实际情况来确定。

当今的网络拍卖可以大致分为两大类,即公开投标拍卖和封闭投标拍卖。在公开投标拍卖中存在两种拍卖模式分别为英式拍卖与荷兰式拍卖。封闭投标拍卖可以分为第一价格封标拍卖，和第二价格封标拍卖。

传统的英式拍卖是以公开投标的拍卖方式进行，拍卖开始时，由拍卖商负责组织整个拍卖活动，首先拍卖商为拍卖物品设定一个底价以及中标的条件或时限，每一位投标者在标单上选定一个高于目前底价的竞标价格，当拍卖物品的中标条件或时限结束时，此拍卖活动结束，同时公布中标者最终的中标价格和相关信息。该类型的拍卖广泛应用于艺术品、二手商品等。

降价拍卖也称为荷兰式拍卖，最早源于荷兰的花卉市场，拍卖的方式大多以回合的方式进行，拍卖商首先为拍卖物品设定一个最高价钱，每经过一个回合，价钱逐渐往下降，直到有竞标者表明愿意以该价格得标为止。该类型的拍卖应用于台湾花卉、农产品批发市场等。

秘密投标拍卖是采取单次投标的方式，首先拍卖商为拍卖商品设定底价以及中标条件或时限，所有竞标者投出已知封闭标单，并在封闭表单内选定一个高于底价的竞标价钱，当投标结束时，拍卖商聚集所有标单，进行开标并公布中标价格和中标者信息。其中第一价格封标拍卖是指买家提交封闭式投标并且投标价格最高者以其投标价格获得物品；第二价格封标拍卖是指买家提交封闭式投标并且投标价格第二高者以其投标价格获得物品。

5.2 电子拍卖协议及安全性要求

一般而言，在拍卖方案中存在三个参与者分别为：标的物 *objects*，即被卖商品；拍卖商 *auctioneers* 即卖方 *seller*；出价人 *bidder*，即买方 *buyers*。Kikuchi 等人提出的电子拍卖模型[48]如图 5.1 所示：

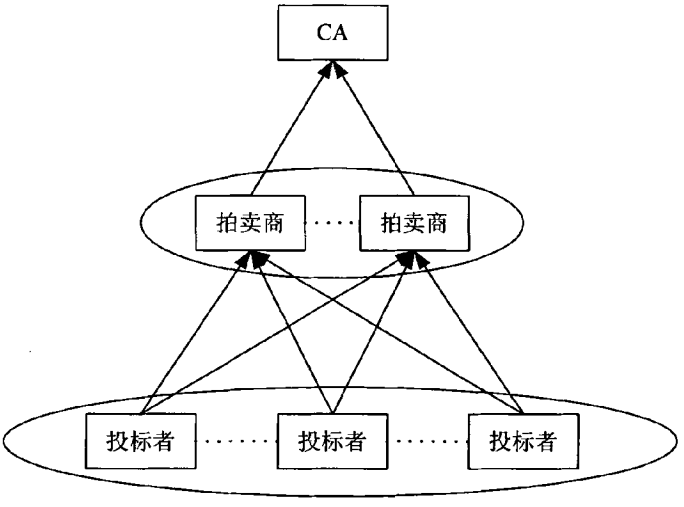

图 5.1 电子拍卖模型
Fig. 5.1 Electronic auction model

拍卖过程可以分为以下几阶段：

（1）注册阶段

电子拍卖的使用者分为投标者和拍卖商。拍卖开始之前投标者和拍卖商首先向系统的注册中心注册一个合法身份，注册中心为投标者和拍卖商分配一个唯一的身份标识，并产生相应的证书。

（2）加入阶段

在匿名电子拍卖协议中，投标者在进行投标之前须与拍卖商协商一个笔名，投标者使用笔名和拍卖商进行通讯，从而保证了拍卖活动的匿名性。

（3）投标阶段

拍卖商宣布拍卖开始后，每个合法的投标者提交他们的标价给拍卖商。实际上拍卖商得到投标者在标价上的签名，供拍卖商验证。

（4）结束投标阶段

经过一段时间后，拍卖行宣布投标结束，不再接收标价。拍卖行宣布中标者并且公开获胜标价。最后，中标者支付等量的货币获得标的物。

对于一个安全的匿名电子拍卖协议通常应该满足如下安全特性：

（1）投标者的匿名性： 不能从投标的签名上获得投标者的身份。

（2）可追踪性：在宣布中标结果后中标者不能否认他/她的投标。

（3）不可伪造性：不能以一个有效的签名伪造标书。

（4）可验证性：任何人都可以验证投标的签名 ,以确认投标人是否有效。

（5）标价保密性：只有中标价公开 ,其他标价保密。

（6）不可连接性：不同的拍卖方之间，投标者之间不能互相串通得出投标者秘密信息。

5.3 匿名电子拍卖协议描述

拍卖过程分为注册阶段、投标阶段、公布阶段等。所对应的具体算法为初始化算法、加入算法、投标算法、验证算法和打开算法。

5.3.1 公开拍卖协议

该拍卖协议适用于公开类型和封闭类型的拍卖，拍卖商和竞标者均可以自由加入，竞标者可以选择任何拍卖商，并在匿名情况下，自由参与竞标活动。公开拍卖协议分为三个子协议，分别为注册协议、加入协议、投标协议。

公开拍卖协议基于非交互式 CL+签名，系统初始化阶段，通过输入安全参数 k，得到系统参数 $BilinearSetup(1^k) \rightarrow params = (n, G, G_T, e, g)$，其中 g 是群 G 的一个生成元，n 为群 G 的阶 $n = p \cdot q$，p 和 q 为大素数，e 为双线性映射。认证中心 CA 产生私钥 (csk_1, csk_2)，其中 $csk_1 = p, csk_2 = \alpha$，$\alpha \leftarrow Z_n$，认证中心 CA 的公钥为 $(cpk_1, cpk_2) = (g^{csk_1}, g^{csk_2})$，$TKey = q$ 用于追踪用户信息。

（1）注册协议

在注册协议中，系统为用户和拍卖商分别产生唯一的身份信息 UID、PID，并分别得到由系统认证中心 CA 所颁发证书 $Cert = g^{UID \cdot csk_2 + csk_1 \cdot csk_2}$ 和 $Cert = g^{PID \cdot csk_2 + csk_1 \cdot csk_2}$，同时系统将用户身份信息和证书储存到 $regID$ 中。用户证书的合法性可通过等式 $e(Cert, g) = e(g^{UID}, cpk_2) \cdot e(cpk_1, cpk_2)$ 来验证，拍卖商证书合法性通过等式 $e(Cert, g) = e(g^{PID}, cpk_2) \cdot e(cpk_1, cpk_2)$ 验证。注册阶段流程图如图 5.4 所示。

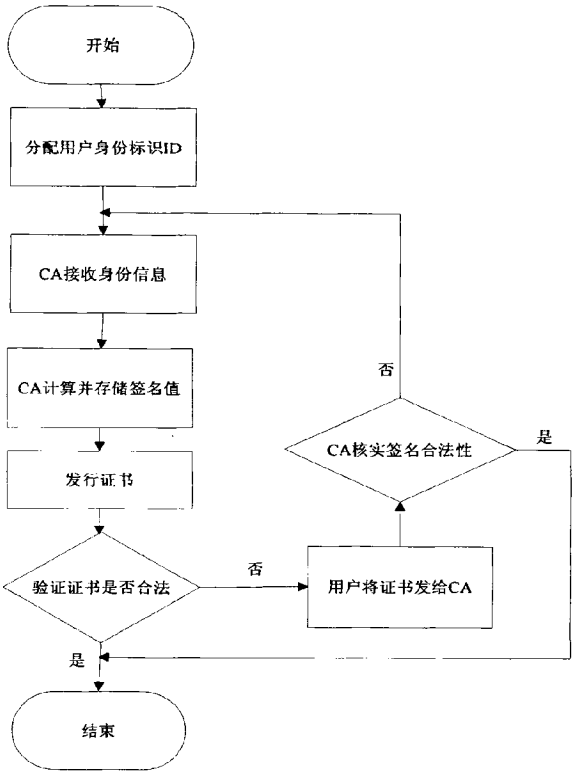

图 5.2 注册阶段流程

Fig. 5.2 Sequence diagram of the register peroid

（2）加入协议

拍卖商 Auctioneer 产生私钥对 (ask_1, ask_2) 其中 $ask_1, ask_2 \leftarrow Z_n$，公钥 $(apk_1, apk_2) = (g^{ask_1}, g^{ask_2})$。合法用户 U_i 向拍卖商注册一个笔名，用于和拍卖商通信，同时达到隐藏身份的目的。用户 U_i 和拍卖商运行两方计算协议，用户选择秘密值 usk_i，随机数 $r_1 \leftarrow Z_p$，用户输入 $(g^{UID_i}, Cert_i)$，$(r_1, usk_i, open)$ 如果证书合法则协议继续运行，否则终止。拍卖商选择随机数 $r_2 \leftarrow Z_p$ 输入 (ask_1, ask_2, r_2)，最终拍卖商得到协议的输出 $(a_1, a_3) = (g^{UID_i/(r_1+r_2)}, g^{UID_i \cdot (ask_1 + ask_1 \cdot ask_2 \cdot usk_i)/(r_1+r_2)})$，拍卖商计算 $a_2 = a_1^{ask_2}$，并将 $(a_1, a_2, a_3) = (g^{UID_i/(r_1+r_2)}, g^{UID_i/(r_1+r_2)ask_2}, g^{UID_i \cdot (ask_1 + ask_1 \cdot ask_2 \cdot usk_i)/(r_1+r_2)})$ 发给用户 U_i，同时记录相关信息。用户 U_i 收到后计算 $(a_4, a_5) = (a_1^{usk_i}, a_2^{usk_i})$，最终得到 $(a_1, a_2, a_3, a_4, a_5)$ 作为与拍卖方通

讯证书，其中a_3可作为笔名。加入阶段流程图如图5.3所示。

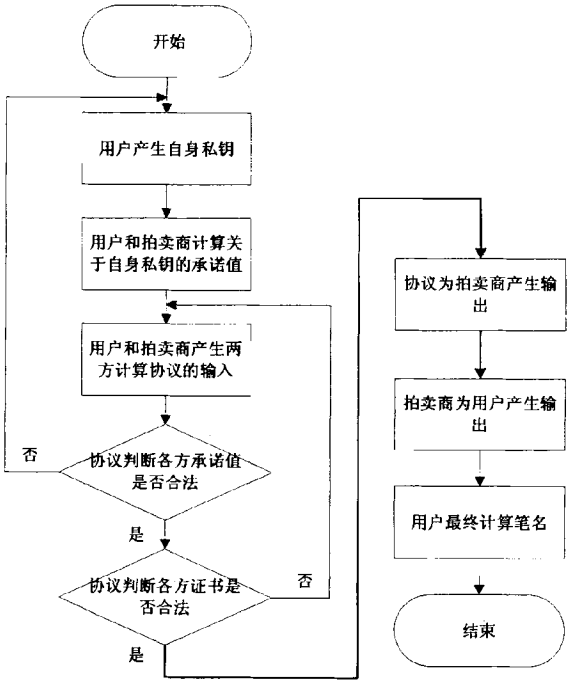

图5.3 加入阶段流程

Fig. 5.3 Sequence diagram of the joining period

（3）公开投标协议

公开投标阶段，用户U_i选择合适的价格$V=(v_1,...,v_n)\leftarrow\{0,1\}^n$，随机数$s\leftarrow Z_n^*$，$u',u_1,...u_n\leftarrow G$，计算$\sigma_1=a_5\cdot(u'\prod_{i=1}^{n}u_i^{v_i})^s$，$\sigma_2=g^s$，$\pi=u'\prod_{i=1}^{n}u_i^{v_i}$。用户产生关于投标者自身身份和投标签名的非交互式零知识证明发送给拍卖商供其验证：$\Pi_{NIZK}=\{a_1,a_2,a_3,a_4,a_5,\sigma_1,\sigma_2,V,\pi:e(a_2,g)=e(a_1,apk_2)\wedge e(a_3,g)=e(a_1,apk_1)\cdot e(a_5,apk_1)\wedge e(\sigma_1,g)=e(a_4,apk_2)\cdot e(\pi,\sigma_2)\}$发送给拍卖商，由拍卖商进行验证。拍卖商首先验证笔名证书的合法性，如果验证通过则验证投标信息的合法性，否则将消息丢弃。验证等式来自于非交互式证明包含三部分：$e(a_2,g)=e(a_1,apk_2)$，$e(a_3,g)=e(a_1,apk_1)\cdot e(a_5,apk_1)$，$e(\sigma_1,g)=e(a_4,apk_2)\cdot e(\pi,\sigma_2)$。经过一段时间后，拍卖行宣布投标结束，不再接收标价。拍卖行宣布最高价位者获胜并且公开获胜标价。公开投标阶段流程图如图5.4所示。

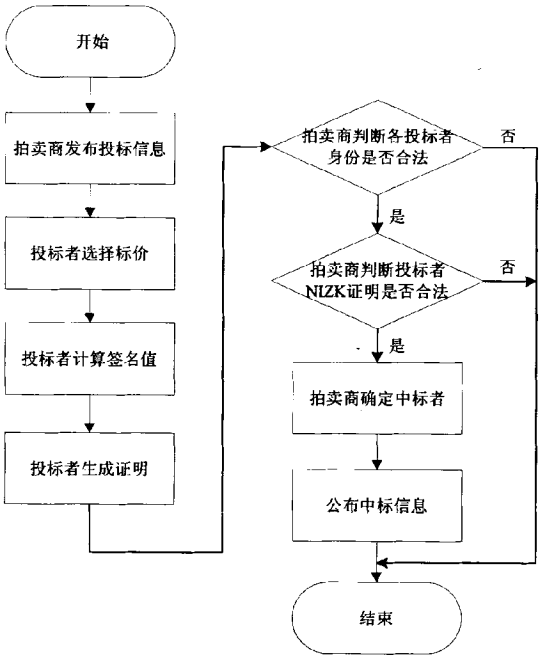

图 5.4 公开投标阶段流程

Fig. 5.4 Sequence diagram of the public bidding period

拍卖时间终止，拍卖商对用户消息验证结束，并从正确的消息中选出胜出的竞标者 U_i，并宣布拍卖结束。产生纠纷时，由认证中心 CA 使用追踪密钥 $TKey$ 追踪用户 U_i 和拍卖商的身份信息，利用组合阶群的正交性，通过如下计算方法可以追踪出用户信息：$Cert^q=(g^{UID\cdot sk_2+sk_1\cdot sk_2})^q=(g^{UID\cdot sk_2}\cdot g^{sk_1\cdot sk_2})^q=(g^{UID\cdot sk_2}\cdot g^{p\cdot sk_2})^q=(g^{q\cdot sk_2})^{UID}$。

5.3.2 封闭拍卖协议

封闭拍卖协议基于非交互式 BB+签名方案。与公开拍卖协议类似，封闭投标协议也分为三个子协议，分别为注册协议、加入协议、投标协议。系统初始化阶段，通过输入安全参数 k，得到系统参数 $BilinearSetup(1^k)\rightarrow params=(n,G,G_T,e,g)$，其中 g 是群 G 的一个生成元，n 为群 G 的阶 $n=p\cdot q$，p 和 q 为大素数，e 为双线性映射。认证中心 CA 产生私钥 (csk_1,csk_2)，其中 $csk_1=p,csk_2=\alpha$，$\alpha\leftarrow Z_n$，认证中心 CA 的公钥为 $(cpk_1,cpk_2)=(g^{csk_1},g^{csk_2})$，$TKey=q$ 用于追踪用户信息。

（1）注册协议

在注册阶段，系统为用户和拍卖商分别产生唯一的身份信息 UID、PID，并分别得到由系统认证中心 CA 所颁发证书 $Cert=g^{UID\cdot csk_2+csk_1\cdot csk_2}$ 和 $Cert=g^{PID\cdot csk_2+csk_1\cdot csk_2}$，同时系统将用户身份信息和证书储存到 $regID$ 中。用户证书的合法性可通过等式 $e(Cert,g)=e(g^{UID},cpk_2)\cdot e(cpk_1,cpk_2)$ 来验证，拍卖商证书合法性通过等式

$e(Cert,g)=e(g^{PID},cpk_2)\cdot e(cpk_1,cpk_2)$ 验证。

（2）加入协议

拍卖商 Auctioneer 产生私钥 $ask\leftarrow Z_n$，公钥 $apk=g^{ask}$。合法用户 U_i 向拍卖商注册一个笔名，用于和拍卖商通信，同时达到隐藏身份的目的。用户 U_i 和拍卖商运行一个两方计算协议，用户选择随机数 $r_1,r_2\leftarrow Z_n$，将 $(g^{UID_i},Cert_i)$，$(usk_i,r_1,r_2,open)$ 作为两方计算协议的输入，如果证书合法则协议继续运行，否则终止。拍卖商选择随机数 $r'\leftarrow Z_n$，并将 (ask,r') 作为两方计算协议的输入，最终拍卖商得到输出 $x_1=(ask+r_1r')r_2$，$x_2=(usk_i+r_1r')r_2$，g^{UID_i} 并计算 $C_1'=g^{1/x_1}$，$C_2'=g^{1/x_2}$，$C_3'=g^{r'}$，$g_2=g^{UID_i/ask}$ 将 C_1',C_2',C_3',g_2 发给用户。用户计算 $C_1=C_1'^{r_2}$，$C_2=C_2'^{r_2}$，$C_3=C_3'^{r_1}$ 作为自身的证书，g_2 为拍卖商和用户产生的笔名。

（3）封闭投标协议

在封闭投标阶段中，投标的价格不公开，拍卖商收到标书后，执行某些操作，识别出投标的价格，投标活动开始时，拍卖商为拍卖物品设定竞标条件和时限，并公开相关消息，每位投标者用上述与拍卖商协商的笔名进行投标。

用户 U_i 确定竞标价格 $v\leftarrow G_T$ 后，选择 $t,\gamma\leftarrow\mathbb{Z}_n^*$，$h,u\leftarrow G$，计算 $g_1=g^{usk_i}$，$\sigma_1,...,\sigma_6=(g_2^{usk_i}h^{\gamma},g^{\gamma},e(g_1,g_2)^t\cdot M,g_1^t,g^t,h^t)$，$\pi_1=C_1^{usk_i}$，$\pi_2=C_1^{\gamma}\cdot C_2^t$。用户产生关于投标者自身身份和投标签名的非交互式零知识证明发送给拍卖商供其验证：$\Pi_{NIZK}=\{C_1,C_2,C_3,g_1,g_2,\sigma_1,...,\sigma_6,\pi_1,\pi_2:\ e(C_1,apk_3C_3)=e(g,g)\wedge e(C_2,g_1C_3)=e(g,g)$ $\wedge e(\sigma_1,C_1)\cdot e(\sigma_6,C_2)=e(g_2,\pi_1)e(h,\pi_2)\wedge e(\sigma_4,g_2)\cdot e(\sigma_2,\sigma_6)=e(\sigma_1,\sigma_5)\}$。

拍卖商首先通过下列等式验证投标者 U_i 证书的合法性：$e(C_1,apk_3C_3)=e(g,g)\wedge e(C_2,g_1C_3)=e(g,g)$，如验证通过则继续验证竞标信息的合法性，否则将消息丢弃。验证竞标信息的合法性通过下列等式：$e(\sigma_1,C_1)\cdot e(\sigma_6,C_2)=e(g_2,\pi_1)e(h,\pi_2)\wedge e(\sigma_4,g_2)\cdot e(\sigma_2,\sigma_6)$ 若等式成立则通过 $M=\sigma_3/e(\sigma_4,g_2)$ 计算出价格，否则将消息丢弃。拍卖时间截止时，拍卖商对用户消息验证结束，并从正确的消息中选出胜出的竞标者 U_i，并宣布拍卖结束。封闭投标阶段流程图如图 5.5 所示。

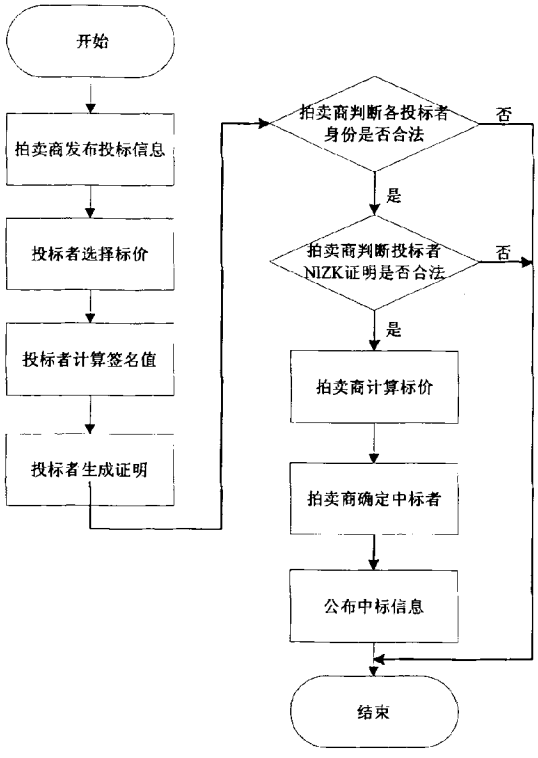

图 5.5 封闭投标阶段流程

Fig. 5.5 Sequence diagram of the sealed bidding period

5.4 效率分析

本节分别从通信代价和计算代价两方面对公开拍卖协议与封闭拍卖协议进行效率分析，各数据描述如下：

从通信角度，在协议初始化过程中，使用群 G 中元素 3 个，Z_n 中元素 3 个，注册阶段使用群 G 中元素 2 个，Z_n 中元素 2 个，配对运算 2 个。公开拍卖协议加入阶段使用群 G 中元素 7 个，Z_n 中元素 4 个；投标阶段使用群 G 中元素 $n+4$ 个（n 为标价的长度），Z_n 中元素 2 个；验证阶段使用配对运算 8 个；封闭拍卖协议加入阶段使用群 G 中元素 8 个，Z_n 中元素 6 个；投标阶段使用群 G 中元素 10 个，Z_n 中元素 3 个，配对 2 个；验证阶段使用配对 11 个。

从计算角度，在协议初始化过程中，进行 2 次幂运算；注册阶段进行 1 次点乘运算，2 次幂运算，3 次配对运算；公开拍卖加入阶段进行 3 次点乘运算，7 次幂运算；公开投标阶段进行 $n+1$ 次点乘运算，$n+3$ 次幂运算；验证阶段进行 8 次配对运算，*open* 阶段进行 1 次幂运算。封闭投标加入阶段进行点乘运算 3 次，幂运算 10 次；封闭投标阶段进行 2 次点乘运算，8 次幂运算，2 次配对运算；验证阶段进行 2 次点乘运算，11 次

配对运算。协议的计算代价和通信代价统计如表5.1所示。

表5.1 效率分析
Table 5.1 Analysis of the efficiency

方案 \ 代价		通信代价			计算代价		
		G_T中元素	群中元素数量	Z_n中元素数量	求幂	配对	点乘
公开	初始化	0	3	3	2	0	0
	注册	2	2	2	2	3	1
	加入	0	7	4	7	0	3
	投标	0	n+4	2	n+3	0	n+1
	验证	8	0	0	0	8	0
	打开	0	0	0	1	0	0
	总量	10	n+16	9	n+15	11	n+5
封闭	初始化	0	3	3	2	0	0
	注册	2	2	2	2	3	1
	加入	0	8	6	10	0	3
	投标	2	10	3	8	2	2
	验证	11	0	0	0	11	2
	打开	0	0	0	1	0	0
	总量	15	23	14	23	16	8

5.5 主要算法

在基于非交互式签名的电子拍卖协议中，认证中心、用户、以及拍卖商之间需要实现的算法有：初始化算法、注册算法、加入算法、投标算法、验证算法以及追踪算法。用伪代码描述各算法如下：

（1） 初始化算法

算法1 初始化算法

输入： $k \in Z$ (k为安全参数)

输出： $params = (n = pq, G, G_T, e, g, csk, cpk, ask, apk, usk)$

1： $params = (n, G, G_T, e, g) \leftarrow BilinearSetup(1^k)$;

2： $\alpha \leftarrow Z_n, p, q \leftarrow Z_n^*$;

3： int $csk_1 = p$, $csk_2 = \alpha$, $TKey = q$;

4： int $cpk_1 = g^{csk_1}$, $cpk_1 = g^{csk_1}$;

5：for i=1:m do $ask_i \leftarrow Z_n$；$apk_i \leftarrow Z_n$

end for

6：for i=1:n do $usk_i \leftarrow Z_n$；

end for

7：return $params=(n=pq,G,G_T,e,g,csk,cpk,ask,apk,usk)$

（2） 注册算法

算法2 注册算法

输入：$params=(n=pq,G,G_T,e,g,csk,cpk,ask,apk,usk)$

输出：$Cert_i,Cert_j$

1：for i=1:m do

$random_U \leftarrow Z_n$；

拍卖商合法证书 $Cert=g^{PID\cdot csk_2+csk_1\cdot csk_2}$；

验证等式 $e(Cert,g)=e(g^{PID},cpk_2)\cdot e(cpk_1,cpk_2)$

end for

2：for j=1:n do $UID \leftarrow Z_n$；

用户合法证书 $Cert=g^{UID\cdot csk_2+csk_1\cdot csk_2}$

验证等式 $e(Cert,g)=e(g^{UID},cpk_2)\cdot e(cpk_1,cpk_2)$

end for

3：return $Cert_i,Cert_j$；

（3） 加入算法

算法3 加入算法

输入：$params=(n=pq,G,G_T,e,g,csk,cpk,ask,apk,usk),Cert_i,Cert_j$

输出：C_i

1：用户选择随机数 $random_U \leftarrow Z_n$

2：拍卖商选择随机数 $random_P \leftarrow Z_n$

3：用户和拍卖商运行两方计算协议；

用户输入 $g^{UID_i},Cert_i,usk_i,random_U,open$,；

拍卖商输入 $ask,random_P$；

两方计算协议为拍卖商输出 x_i；

4：拍卖商利用 $x_i,ask,random_P,params$ 等值计算产生 C_i',g_i；

5：拍卖商将 C_i',g_i 发给用户；

6：用户利用$usk_i, random_U, params$计算产生证书C_i；

7：if 捕获异常；

8：then return ERROR;

9：else return C_i；

（4）公开投标算法

算法4 公开投标算法

输入：　$params=(n=pq,G,G_T,e,g,csk,cpk,ask,apk,usk),C_i$

输出：　Π_{NIZK},π

1：确定标价$V=(v_1,...,v_n)\leftarrow\{0,1\}^n$;选择$s\leftarrow Z_n^*$;$u',u_1,...u_n\leftarrow G$;

2：计算$\sigma_1=a_5\cdot(u'\prod_{i=1}^n u_i^{v_i})^s$;$\sigma_2=g^s$;$\pi=u'\prod_{i=1}^n u_i^{v_i}$;

3：将证书C_i与σ_i承诺$R_C=GSCommit(params_{GS},C_i,open_C)$;

$R_\sigma=GSCommit(params_{GS},\sigma_i,open_\sigma)$;

4：产生证明 $\Pi_{NIZK}=\{a_1,a_2,a_3,a_4,a_5,\sigma_1,\sigma_2,V,\pi:$

$e(a_2,g)=e(a_1,apk_2)\wedge e(a_3,g)=e(a_1,apk_1)\cdot e(a_5,apk_1)\wedge e(\sigma_1,g)=e(a_4,apk_2)\cdot e(\pi,\sigma_2)\}$

5：产生证明 $\pi\leftarrow NIZK\{((comm:a),(comm':b)):a=b\}$;

计算承诺$comm=Commit(params_{GS},m,open)$;

$comm'=Commit(params_{GS},m,open')$;

6：return Π_{NIZK},π；

（5）封闭投标算法

算法5 封闭投标算法

输入：　$params=(n=pq,G,G_T,e,g,csk,cpk,ask,apk,usk),C_i$

输出：　Π_{NIZK},π

1：确定标价$v\leftarrow G_T$;选择$t,\gamma\leftarrow\mathbb{Z}_n^*$，　$h,u\leftarrow G$;

2：计算$g_1=g^{usk_i},\sigma_1,...,\sigma_6=(g_2^{usk_i}h^\gamma,g^\gamma,e(g_1,g_2)^t\cdot M,g_1^t,g^t,h^t)$，　$\pi_1=C_1^{usk_i},\pi_2=C_1^\gamma\cdot C_2^t$;

3：将证书C_i与σ_i承诺$R_C=GSCommit(params_{GS},C_i,open_C)$;

$R_\sigma=GSCommit(params_{GS},\sigma_i,open_\sigma)$;

4：产生证明:

$\Pi_{NIZK}=\{C_1,C_2,C_3,g_1,g_2,\sigma_1,...,\sigma_6,\pi_1,\pi_2:\ e(C_1,apk_3C_3)=e(g,g)\wedge e(C_2,g_1C_3)=e(g,g)$

$\wedge e(\sigma_1,C_1)\cdot e(\sigma_6,C_2)=e(g_2,\pi_1)e(h,\pi_2)\wedge e(\sigma_4,g_2)\cdot e(\sigma_2,\sigma_6)=e(\sigma_1,\sigma_5)\}$

5：产生证明 $\pi\leftarrow NIZK\{((comm:a),(comm':b)):a=b\}$;

计算承诺$comm=Commit(params_{GS},m,open)$;

$comm'=Commit(params_{GS},m,open')$。

6：return Π_{NIZK}, π；

（6）验证算法

算法6 验证算法

输入：$params=(n=pq,G,G_T,e,g,csk,cpk,ask,apk,usk),C_i,\Pi_{NIZK},\pi$

输出：1/0

1：if $e(a_2,g)=e(a_1,apk_2)$ and $e(a_3,g)=e(a_1,apk_1)\cdot e(a_5,apk_1)$

and $e(\sigma_1,g)=e(a_4,apk_2)\cdot e(\pi,\sigma_2)$

2:else if $e(C_1,apk_3C_3)=e(g,g)$ and $e(C_2,g_1C_3)=e(g,g)$ and

$e(\sigma_1,C_1)\cdot e(\sigma_6,C_2)=e(g_2,\pi_1)e(h,\pi_2)$ and $e(\sigma_4,g_2)\cdot e(\sigma_2,\sigma_6)=e(\sigma_1,\sigma_5)$

2: return 1;

3: else return 0;

（7） 追踪算法

算法7 追踪算法

输入：$TKey=q$, $Cert_i$, $Cert_j$

输出：UID

1：计算$Cert^q=(g^{UID\cdot sk_2+sk_1\cdot sk_2})^q=(g^{UID\cdot sk_2}\cdot g^{sk_1\cdot sk_2})^q=(g^{UID\cdot sk_2}\cdot g^{p\cdot sk_2})^q=(g^{q\cdot sk_2})^{UID}$

2: return UID;

5.6 本章小结

本章基于第4章提出的非交互式签名方案，分别构建了匿名公开投标协议与匿名封闭投标协议，该协议与传统的投标方案相比具有完全匿名性，整个方案中交互次数最少，并具有非交互式零知识证明的特性，同时对协议的效率进行分析，并用伪代码描述相关算法。

第6章 总结与展望

在面向开放式网络环境下的电子商务中，如电子投标、电子现金、电子拍卖等系统中，匿名性具有极为重要的地位，虽然零知识证明能够解决这一安全性要求，但交互次数频繁限定了它的实际应用。而非交互式零知识证明的研究目的就是用高效的手段解决匿名性等安全问题。非交互式零知识证明的重要特点之一就是抗选择密文攻击，解决了密码学中一个棘手问题，除此之外，非交互式证明中交互的单向性也是密码学协议的一个重要应用可以为协议的参与方提供离线支持。

Groth 等人提出了强有力的非交互式 GS 证明系统，用于证明被承诺的变量所满足的特定关系。该证明系统除了具有 NIZK 证明系统的基本安全需求之外，还具有自身的特性，比如随机性、F-可提取性等，而这些特性都具有很高的应用价值。

本文对目前国内外 NIZK 证明系统进行了广泛而系统的研究，着重介绍了 NIZK 证明系统的发展、NIZK 证明系统的结构及安全特性、典型的 NIZK 证明系统如 GS 证明系统。针对当前的匿名性需求以及当前多数的匿名签名的通信代价和计算代价较高，不支持离线状态、不能抵抗选择密文攻击等问题。本文提出了基于非交互式零知识证明系统的签名方案。

主要工作如下：

（1）对国内外 NIZK 系统进行了深入而系统地研究，比较了各个 NIZK 证明系统的优缺点，同时对本文所需的基础知识进行阐述。

（2）着重对 GS 证明系统的结构及特性的研究。针对当前组签名方案通信代价和计算代价较高、不能抵抗选择密文攻击等问题，基于 GS 证明系统构及三素数的组合阶双线性群假设构建了高效安全的组签名方案 GSCOBG，该方案的签名的大小是一个常量，不依赖于其他系统参数。

（3）利用 GS 证明系统构建了非交互式 BB+签名方案，非交互式 CL+签名方案，方案作为匿名认证协议的原型，可以实现匿名身份认证的目标，非交互式签名方案具有较高的应用价值，比如可以应用在电子商务系统中电子拍卖系统、电子投票系统、可信计算平台等。

（4）从可证明安全性的角度在数学上证明了 GSCOBG 方案满足 BMW 模型所要求的基本安全特性，正确性、匿名性、可追踪性和不可伪造性。满足 NIZK 证明系统中的完备性、健壮性和不可区分性，并将安全性和同类典型方案进行了对比。证明了非交互

式签名方案的匿名性（包括用户匿名性、签名者的匿名性）零知识性、不可伪造性。

（5）将非交互式签名方案应用于匿名电子拍卖方案中，分别构建了公开拍卖协议和封闭拍卖协议，与传统的电子拍卖相比，该方案具有较强的匿名性、可追踪性、非交互性，同时降低了通讯代价和存储代价。

（6）此外对 GSCOBG 方案及非交互式签名方案进行了效率分析，并和同类的典型方案分别从通信代价和计算代价两方面进行了对比，得出效率优势。

研究结果表明，将非交互式零知识证明理论应用于签名方案中有助于增强传统签名方案的安全性（抗选择密文攻击等），同时可以减少通讯代价，提供实际应用中离线的需求。

由于时间的关系，文中也留下了以下遗憾：由于对于双线性群理论目前国内外尚未建立适当的实验环境，本文未对所提协议进行性能仿真实验。因此本文从理论上对所提出的协议进行代价分析、比较。

参考文献

1. Goldwasser S, Micali C, Rackoff. The Knowledge Complexity of Interactive Proof-Systems[M]. Annual ACM Sym-posium on Theory of Computing, 291-304, 1985
2. Fiat F and Shamir F. How to prove yourself: practical solutions of identication and signature problems[J]. Computer Science 263, 1987, 186-194
3. Jakobsson M, Sako K, and Impagliazzo R. Designated verifier proofs and their applications[J]. Computer Science 1070, 1996, 143-154
4. Blum M, Feldman P, Micali S. Non-interactive zero-knowledge and its applications[J]. Proceedings of STOC'88, 1988, 103-112
5. Bellare M, Goldwasser S. New paradigms for digital signature sandmes-sage authentication based on non-interactive zero-knowledge proofs[J]. LNCS volume435, 1990, 194-211
6. Danny D, Cynthia D, Moni N. Non-malleable cryptography[M]. Theory of Computing, 1991, 542-552
7. Naor M, Yung M. Public-key cryptosystems provably secure against chosen ciphertext attacks[M]. Annual ACM Symposium, 1990, 427-437
8. Gutfreund D and Michael B O. Increasing the Power of the Dealer in Non-interactive Zero-Knowledge Proof Systems[J]. ASIACRYPT 2000, LNCS 1976, 2000, 429–442
9. Blum M, Santis A D, Micali S, and Persiano G. Non interactive zero-knowledge[M]. SIAM Jornal of Computation, 1991, 1084–1118
10. Feige U, Lapidot D, and Shamir A. Multiple non-interactive zero knowledge proofs under general assumptions[J]. SIAM Jornal of Computation, 1999, 1–28
11. Alfredo D S, Giovanni D C, Giuseppe Persiano, and Moti Yung. Image density is complete for non-interactive-szk[J]. ICALP'98, LNCSseries, volume 1443, 1998, 784–795
12. Goldreich O, Sahai A, and Salil P V.Canstatistical zero knowledge be made non-interactive? or on the relation ship of szk and niszk[J]. CRYPTO'99, LNCSseries, volume 1666, 1999, 467–484
13. Groth J, Ostrovsky R, and Sahai A. Perfect non-interactive zero-knowledge for NP[J]. EUROCRYPT'06, LNC Sseries, volume4004, 2006, 339–358
14. Feige U, Lapidot D, and Shamir A. Multiple non-interactive zero knowledge proofs under general assumptions[M]. SIAM Journal of Computing, 1999, 1–28

15. 毛文波. 现代密码学理论与实践[M]. 北京: 电子工业出版社, 2006, 444-448
16. Kilian J and Petrank E. An efficient noninteractive zero-knowledge proof systemfor NP with general assumptions[M]. Journal of Cryptology, 1998, 1–27
17. Alfredo D S, Giovanni D C, and Persiano G. Randomness-optimal characterization of two NP proof systems[J]. RANDOM'02, LNCSseries, volume2483, 2002, 179–193
18. Dolev D, Cynthia D and Moni N. Non-malleable cryptography[M]. SIAM Journal of Computing, 2000, 391–437
19. Sahai A and Waters B. Fuzzy identity-based encryption[J]. EURO-CRYPT'05, LNCSseries, volume3494, 2005, 457–473
20. Goyal V, Pandey O, Sahai A, and Waters B. Attribute-based encryption for fine-grained access control of encrypted data[M]. In ACMCCS'06, 2006, 89–98
21. Boneh D, Crescenzo G D, Ostrovsky R, and Persiano G. Public key encryption with keyword search[J]. EUROCRYPT'04, LNCSseries, volume 3027, 2004, pages 506–522
22. Boneh D, Sahai A, and Waters B. Fully collusion resistant trait or tracing with short ciphertexts and privatekeys. EUROCRYPT'06, LNCSseries, volume 4004, 2006, 573–592
23. Groth J. Simulation-sound NIZK proofs for a practical language and constant size group signatures. Computer Science, volume4248,2006, 444–459
24. Bellare M, Haixia S, and Chong Z. Foundations of group signatures: The case of dynamic groups[J]. CT-RSA'05, LNCSseries, volume3376, 2005, 136–153
25. Groth J, Ostrovsky R, and Sahai A. Non-interactive zaps and new techniques for nizk[J]. CRYPTO'06, LNC Sseries, volume4117, 2006, 97–111
26. Groth J and Sahai A. Efficient Non-interactive Proof Systems for Bilinear Groups[J]. EUROCRYPT 2008, LNCS 4965, 2008, 415-432.
27. Boneh D, Goh E and Nissim K. Evaluating 2-DNF Formulas on Ciphertexts[J]. Springer LNCS3378, 2005, 325-341
28. Boneh D, Boyen X and Shacham H. Short group signatures[J]. CRYPTO2004, Springer LNCS3152, 2004, 41-55
29. Ateniese G, Camenisch J, Medeiros B and Hohenberger S. Practical group signatures without random oracles[R]. Cryptologye Print Archive, Report2005/385
30. Groth J and Sahai A. Efficient non-interactive proof systems for bilinear groups(full version). http://www.brics.dk/~jg/WImoduleFull.pdf
31. Chandran N, Groth J, and Sahai A. Ring signatures of sub-linear size without random oracles[J]. Computer Science, volume 4596, 2007, 423–434

32. Groth J. Fully Anonymous Group Signatures Without Random Oracles[J]. ASIACRYPT 2007, LNCS 4833, 2007, 164–180
33. Boyen X and Waters B. Compact group signatures without random oracles[J]. Computer Science, volume4004, 2006, 427–444
34. Boyen X and Waters B.Full-domain subgroup hiding and constant-size group signatures. Computer Science, volume4450, 2007, 1–15
35. Meiklejohn S. An Extension of the Groth-Sahai Proof System[R]. Brown University Masters thesis, 2009
36. Ghadafi E, Smart N P, and Warinschi B. Groth-Sahai proofs revisited[J]. Springer Berlin Heidelberg. Volume 6056/2010, 177-192
37. Boneh D, Goh E., and Nissim K. Evaluating 2-dnf formulas on ciphertexts[M]. Theory of Cryptography, volume3378, 2005, 325-342
38. Lewko A and Waters B. New Techniques for Dual System Encryption and Fully Secure HIBE with Shot Ciphertexts[J]. TCC 2010, LNCS 5978, 2010, 455-479
39. Pedersen T. Non-interactive and information-theoretic secure verifiable secret sharing[A]. In: Proceedings of CRYPTO'91[C], Santa Barbara, 1991, 129–140
40. Bellare M, Micciancio D, and Warinschi B. Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. Cryptology EUROCRYPT 2003, volume 2656, 2003, 614–29
41. Bellare M, Shi H, and Zhang C. Foundations of Group Signatures. The Case of Dynamic Groups[R]. Cryptology ePrintArchive: 2004, Report 2004/077.
42. Ateniese G, Camenisch J, Hohenberger S, and Medeiros B. Practical group signature without random oracles[R]. Cryptology ePrint Archive, 2005, Report 2005/385.
43. Libert B and Vergnaud D. Group Signatures with Verifier-Local Revocation and Backward Unlinkability in the Standard Model[J]. Computer Science, Volume 5888, 2009, 498-517
44. Lingbo W and Jianwei L. Shorter Verifier-Local Revocation Group Signature with Back ward Unlinkability[J]. Pairing 2010, LNCS6487, 2010, 136–146
45. Boneh D, Franklin M K. Identity-based encryption from the weil pairing[J]. SIAM Journal of Computing, 32(3), 2003, 586-615
46. 冯登国. 安全协议理论与实践[M]. 北京: 清华大学出版社, 2011, 125-140
47. Jarecki S, Shmatikov V. Efficient two-party secure computation on committed inputs[A]. In: Proceedings of EUROCRYPT'07[C], Barcelona, 2007, 97-114
48. Kikuchi H, Harkavy M, and Tygar J D. Multi-round Anonymous Auction Protocols[J]. IEICE transactions on Information and System, Volume. 82, 1999, 769-777

致 谢

东大两年的研究生生活中，我收获了很多，每一步的成长都为我以后步入社会打下基础。在研究生生活即将结束之际，由衷地对培养教育我的老师们以及陪伴我的同学朋友们表示感谢。

首先感谢我的导师周福才教授对我的培养。周老师对我们无微不至的关心和不倦的教诲深入我们每个人的内心，为我们创造了极为良好的学习环境和学术氛围。周老师重视基础教育，从点点滴滴的小问题入手，真正从原理上让我们明白信息安全。对于每个人的研究方向，周老师都为我们严格把关，在论文的选题、撰写、修改过程中，为我们逐字逐句的指导，没有丝毫懈怠。很幸运，我能进入网络与信息安全实验室，更幸运的是能遇见一位好导师。

感谢徐剑老师对我的关心和肯定，两年来徐老师像师兄一样指导我们的学习和工作，为我们解决学习和工作上的困难。感谢刘秀梅老师对我的指导与帮助，柳老师和蔼可亲的态度和工作的细心程度永远值得我学习。感谢李永久同学对我的支持和鼓励，让我充满生活的信心和奋斗的动力。感谢李慧师姐为我提供学习的建议以及生活上的照顾，她永远是我学习的榜样。感谢李福祥同学组织的丰富多彩的活动，为我们带来很多快乐。感谢陈晨师妹的陪伴以及工作上的支持。感谢林慕清、田静以及冶金馆 349 的其他同学给予我的帮助。感谢信息学管 422 的廖小林和关龙同学对我的照顾。感谢所有朝夕相处的同学和朋友们，我们一起度过了最为难忘的时光，认识你们是我的荣幸。

感谢我的父母和亲人们给予我的关爱、支持和包容，你们永远是我最坚实的后盾和最温暖港湾。感谢所有关心、帮助过我的人们。

感谢参加论文评审的各位专家，感谢你们在百忙之中对我的论文提出批评指正。

攻读硕士学位期间的论文项目情况

已发表或录用的论文：

1. 周福才，王兰兰，徐剑，岳虹，陈晨. 基于组合阶双线性群的组签名方案. 第六届安全协议研讨会[M], 2011,128-152
2. ZHOU Fucai, XU Jian, LI Hui, WANG Lanlan. Group Signature Based on Non-interactive Zero-Knowledge Proofs [J],China Communications,2011,8(2): 34-41(SCI 检索 检索号 000289663300005).
3. ZHOU Fucai, Wang Lanlan, XU Jian, A Group Signature in the Composite Order Bilinear Groups. 已投稿 Information Security Conference 2011,审稿中
4. 周福才，徐剑，李 慧，王兰兰. 基于非交互式零知识证明的电子现金协议，已投稿中国密码学会 2011 年会，审稿中

参与的科研项目：

国家高技术研究发展计划，项目编号：2009AA01Z122。