基于区块链的匿名问卷系统

毛子旗 刘百祥

（复旦大学计算机科学技术学院上海市智能信息处理重点实验室 上海 200433）

（上海市区块链工程技术研究中心复旦-众安区块链与信息安全联合实验室 上海 200433）

摘 要 在传统的问卷系统中，用户需要经过认证才能填写问卷，然而这可能造成隐私信息的泄露。同时传统的问卷系统可能存在一个用户多次填写的问题。针对上述缺陷，基于非交互式零知识证明、消息摘要算法以及区块链提出一种新的问卷系统，该系统在保护用户隐私信息的同时保证一个用户只能提交一次问卷。系统将问卷调查的流程用智能合约实现，保证系统全部流程的公开透明可信。系统在以太坊上进行实际部署测试，消耗的 gas 值在可接受范围，可以解决传统问卷系统中的问题。

关键词 区块链 零知识证明 隐私保护 问卷系统 智能合约

中图分类号 TP3 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2024.06.003

ANONYMOUS QUESTIONNAIRE SYSTEM BASED ON BLOCKCHAIN

Mao Ziqi Liu Baixiang

(*Shanghai Key Laboratory of Intelligent Information Processing*, *School of Computer Science*, *Fudan University*, *Shanghai 200433*, *China*)

(*Fudan-Zhongan Joint Laboratory of Blockchain and Information Security*, *Shanghai Engineering Research Center of Blockchain*, *Shanghai 200433*, *China*)

Abstract In traditional questionnaire systems, users are usually required to be authenticated before they can fill in the questionnaire, which may cause the disclosure of private information. At the same time, the traditional questionnaire system may have a problem that users fill in repeatedly. Aimed at the traditional system defects, a new system based on non-interactive zero-knowledge proof, message digest algorithm and blockchain is proposed. This system protected user's privacy data while ensuring that a user could only submit it once. The system implemented the questionnaire survey process with smart contracts to ensure openness, transparency and credibility. The system was actually deployed on Ethereum and the consumed gas was acceptable. The results show that the problems in the traditional questionnaire system can be solved.

Keywords Blockchain Zero-knowledge proof Privacy protection Questionnaire system Smart contract

0 引 言

网络问卷调查一直是一种比较普遍、方便的调研方法。对于调查方，只需要在互联网上发布感兴趣的问题，被调查方可以异步地填写问卷并提交。相对于传统的线下调查，具有方便、快速、成本低、可并行进行、自统计等优点。经过多年研究，网络问卷调查系统有很大发展，可以适用于各种调查情形，基本满足调查方需要。但是对于被调查方而言，还是存在着身份暴露、问卷信息被非法利用等问题。

为了避免被调查方的身份信息被泄露，同时保证填写问卷的用户只能一人一填，不可重复填写问卷，本文结合区块链的去中心化网络以及零知识证明提出一种新的网络问卷系统，本系统在保护被调查方身份信息同时避免被调查方重复填写问卷。该系统的问卷

收稿日期：2020 - 11 - 10。国家自然科学基金项目（61672166）；国家重点研发计划项目（2019YFB2101703）。**毛子旗**，硕士生，主研领域：区块链，网络安全。**刘百祥**，工程师。

发起、填写问卷、提交问卷合法性检查、问卷结果统计均由智能合约实现,确保了所有过程的公开性、合法性。

问卷调查包括设计问卷、实施调查、整理问卷三个阶段。设计问卷主要是调查者根据关系的问题设计调查问题,这部分不是本文关心的。实施调查主要包括确定调查对象,通过网络分发问卷,确认被调查者的资质。整理问卷主要是问卷内容的整理,包括剔除不合法的问卷内容。利用网络的开放性、自由性以及不受时空限制,网络调查可以在保证成本低的前提下方便高效地进行。但是,由于网络的开放性,存在着代理人攻击、分布式拒绝服务攻击(DDoS)等问题,在调查的实施以及调查结果的整理与统计上都存在着巨大的挑战。

网络调查可以分成匿名的以及非匿名的,匿名的网络调查应该保证除了被调查者能够将一份已经提交的问卷与自己的身份信息关联,其他任何人,包括调查发起人,也不能将一份已提交问卷与被调查者关联在一起。非匿名的网络调查需要被调查者在填写问卷前经过身份认证,同时在填写问卷时也需要出示相应的认证通过的证明,如用户 id,或进行身份认证时的真实姓名。

当前的网络调查出于各种考虑,会要求用户提前注册个人信息或者使用第三方认证的方式登入系统,随后填写问卷并提交。在用户提交问卷时,系统会将填写的问卷内容与用户身份进行绑定。对于这种能够将问卷内容与个人身份信息绑定在一起的可能,用户在回答一些隐私性强,涉及敏感话题的问题时,会采用随大潮的态度,不敢表达自己的观点,造成最终统计数据出现偏差。又或者用户放弃调查,造成被调查用户样本少,数据不能真实反映现状。

1 相关研究

问卷调查与投票系统有很多相似的地方,本文提出的方案可以处理更复杂的数据,因此更适用于问卷调查这样的场景。在文献[1-2]提出的电子投票模型,投票用户只能对候选人进行选择,不能处理非选项问题。对于非选项问题,用户会填写不定大小的信息。本文提出的方法,可以处理任意格式的数据,即适用于问卷调查,也可以用于电子投票。

根据使用密码学技术的不同,当前的电子投票方案可以分成四大类:基于全同态的电子投票方案[4-6];基于环签名、盲签名的电子投票方案[7-8];使用混合网络的电子投票方案[9-11];基于秘密共享的投票方案[14-15]。这些方案都存在各自的不足,基于全同态的电子投票方案的计算复杂度太高,不能有效地应用在实际当中。基于环签名、盲签名的电子投票方案需要假设存在一个可信的中心化的签名机构。使用混合网络的电子投票方案计算过于复杂,使用效率低,实际中难以使用。使用秘密共享的方案存在着内部欺诈的风险,降低了方案的安全性。

电子投票系统强调完备性、健壮性、匿名性、不可重投性、合法性、公平性、可验证性[13],其中公平性是指在全部用户投完票之前,不能获悉投票结果,即不能实时获取结果。本文提出的方案结合零知识证明与区块链,在只进行一轮的零知识证明前提下,尽可能降低计算量,实现最大匿名、唯一性以及全流程的透明。

2 背景知识

2.1 区块链与智能合约

区块链技术是指在一个开放、不可信的网络中,使所有参与节点能够共同维护一个公开、可信、不可篡改的公共账本[20]。系统节点可以在该账本中记录一些信息,以实现电子交易、信息共享等功能。一个区块链系统根据新节点的加入是否需要经过认证可以分成两种:许可链与非许可链。本文讨论的问题都是在非许可链上。非许可链也被叫做公链,公链上的数据可以被任何节点访问,节点的加入与退出不受限制。系统中的节点数量也不受节点限制,最少可以只有一个节点,最多没有上限。许可链包括联盟链与私有链。联盟链一般由多个组织共同维护,节点数量有限,系统中的节点加入需要经过授权认证,多个节点共同维护公共账本。私有链通常只由一个组织或公司内部的参与方进行维护,系统中不存在不可信的节点,私有链主要用于系统内部的自我审计与记录保存,确保历史记录不会被篡改。

比特币最早出现在文献[3]中,比特币仅支持非图灵完备的脚本语言,用于其支持的比特币的转账交易,交易的消息字段也可以用来保存少量信息,将信息记录在比特币中,可以很好地保护信息不可被篡改,同时证明信息的时效。随后出现的以太坊[12],有一个支持图灵完备的虚拟机 EVM(Ethereum Virtual Machine),用户可以在 EVM 上运行事先部署的程序,程序也被叫做智能合约。相对比特币,以太坊支持更加复杂的业务,应用范围不仅局限于电子货币的交易转账,也能应

用于物联网中[21]。本文在以太坊上实现了系统并进行实验。

2.2　零知识证明

在密码学中,零知识证明[16]是一种能够让一方证明者(Prover)向另外一方验证者(Verifier)证明某命题的正确性,同时这个过程不会揭露任何其他额外信息的方法。一个零知识证明系统需要满足以下 3 个属性:

1) 可靠性(Soundness)。证明者无法欺骗验证者。如果该命题不成立,则不管证明者采取何种手段,验证者相信证明者的概率都非常低。

2) 完备性(Completeness)。如果证明者知道如何证明某命题成立,那么证明者有非常高的概率使验证者相信其知道该命题成立。

3) 零知识性(Zero-knowledge)。在验证者验证的过程中,验证者不能获得任何除了命题成立以外的信息。

零知识证明系统可以分成交互式和非交互式[17],交互式零知识证明系统需要验证者与证明者之间进行多次交互。后者不需要进行多次交互,证明者只需要将其生成的证明公布出来,任何验证者都可以进行验证。针对区块链系统,非交互式零知识证明更有优势,证明者可以在链下构建证据,将证据上传到区块链中永久保存,区块链中的其他节点在证据上传以后可以随时验证。

2.3　zk-SNARKs 与 ZoKrates

zk-SNARKs 是简明非交互式零知识证明 Zero-knowledge Succinct Non-interactive Arguments of Knowledge 的简称[19],其在非交互式零知识证明的基础上做了一些优化,最重要的两个改进是减少了证明的大小,同时缩短了验证时间。区块链对于每个区块的大小有严格的上限要求,如果证明者生成的证明过大,不能够被打包进交易中,即使能够被打包进交易中,一个区块能包含的交易数量也会减少,减低了系统吞吐量。零知识证明在区块链中的应用大多是采用链下证明,链上验证,受限于链上计算的效率,验证证明的过程不能过于复杂,zk-SNARKs 相对一般的交互式零知识证明验证更快,更好地适应区块链的应用场景。

zk-SNARKs 算法大致可以分成三个部分:1) 将需要证明的命题转换成一个证明多项式相等的问题,即 $t(x)h(x)=w(x)v(x)$;2) 选取随机数进行验证该多项式成立,证明多项式相等可以对其进行展开合并,但是该过程计算量大,只要挑选的随机数足够随机,安全性也能有保障;3) 同态隐藏,即输入的计算和输出的计算保持同态。

非交互式零知识证明 Z 由三个多项式时间算法组成,分别是 Setup、Prove、Verify。Setup 是由可信第三方生成公共参数;Prove 是证明者生成证明;Verify 则是验证者验证证明的过程。三个算法的具体描述如下:

1) $Setup(1^n)$:由一个可信第三方执行初始化算法,算法输入参数 n 表示系统的安全参数,算法输出为系统的公共参数 $params$,具体过程如下:

选定随机数 r 和 α,可信第三方计算下列数据:

$$E(r^0),E(r^1),\cdots,E(r^n) \tag{1}$$

$$E(\alpha r^0),E(\alpha r^1),\cdots,E(\alpha r^n) \tag{2}$$

$$E(t(r)),E(\alpha t(r)) \tag{3}$$

$$E(v_0(r),\cdots,E(v_m(r))),E(\alpha v_0(r),\cdots,E(\alpha v_m(r))) \tag{4}$$

$$E(w_0(r),\cdots,E(w_m(r))),E(\alpha w_0(r),\cdots,E(\alpha w_m(r))) \tag{5}$$

式中:$E(x)$ 表示对有限群的生成元 g 的 x 次方。上述参数作为公共参数 $params$ 公开。

2) $Prove(params,w,x,M)$:证明者执行生成证明算法,该算法的输入包括初始化算法中生成的 $params$,以及证据 w、命题 x、图灵机算法 M,输出是证明 π,具体计算过程如下:

定义 $v_{\text{free}}(x)=\sum a_k v_k(x)$,计算如式(6) - 式(9)所示。

$$V_{\text{free}}=E(v_{\text{free}}(r)) \tag{6}$$

$$V'_{\text{free}}=E(\alpha v_{\text{free}}(r)) \tag{7}$$

$$W=E(w(r)),W'=E(\alpha w(r)) \tag{8}$$

$$H=E(h(r)),H'=E(\alpha h(r)) \tag{9}$$

式中:$v(x)$、$w(x)$、$h(x)$ 是命题转化成的多项式中的因子,将上述 V_{free}、V'_{free}、W、W'、H、H' 作为证明输出。

3) $Verify(params,x,M,\pi)$:验证者执行验证证明算法,算法输入包括公共参数 $params$、命题 x、图灵机算法 M,以及由证明者提供的证明,算法输出 true/false,分别表示接受或拒绝证明者提供的证明。具体是计算式(10) - 式(13)是否成立,式(10) - 式(12)验证 V_{free}、W、H 是否正确,式(13)验证 V_{free} 与 W 的计算是否采用一致的参数。都成立返回 true,否则返回 false。

$$e(V'_{\text{free}},g)=e(V_{\text{free}},g^{\alpha}) \tag{10}$$

$$e(W',E(1))=e(W,E(\alpha)) \tag{11}$$

$$e(H',E(1))=e(H,E(\alpha)) \tag{12}$$

$$e(E(v_0(r))E(v_{\text{free}}(r)),E(w_0(s))W)=e(H,E(t(r))) \tag{13}$$

ZoKrates[18]是一个用于帮助开发者在以太坊上应用 zk-SNARKs 的工具包。可以帮助开发者快速生成公共参数,生成证明,以及生成用于验证证明的智能合约源码。本文实验使用该工具包实现系统功能。

3 匿名的网络问卷调查系统

3.1 系统角色

本文将系统中的角色根据需要完成的任务分成 4 种,一个用户可以兼任多个角色。下面详细介绍了每个角色在系统中承担的任务。

1)区块链节点。匿名的网络问卷调查系统建立在一个区块链系统上,区块链的稳定运行需要大量的节点参与。

2)系统管理员。系统管理员需要执行 $Setup(1^n)$ 算法,生成 zk-SNARKs 的公共参数,公共参数生成以后,需要上传至区块链,供证明者与验证者使用。本文提出的方法,需要将证明的验证过程写成智能合约,部署在区块链上。证明的验证过程对不同的问卷调查都是统一的,因此这个过程只需在链上部署一次。

3)调查发起人。一份问卷的发起人,问卷发起人在发起一个问卷时需要确认这个问卷的 *id* 与 *nonce*,参与问卷调查的用户公钥集合,问卷题目,这些信息需要事先通过合约调用接口发送至区块链上保存。问卷 *id* 与 *nonce* 用于唯一标识这次问卷调查,被调查用户使用该值生成针对这次调查的唯一 *token*。问卷的 *id* 用于表示问卷的编号,为了增加随机性,引入 *nonce* 随机数,使得标识问卷的信息具有不可预测性。*token* 在系统中表示用户的填写完问卷的内容,防止用户重复提交。本文提出的方案支持对公钥集合进行更新,发起人可以根据需要更新公钥集合。公钥集合的更新不对已经完成调查的用户产生影响,对于那些已经生成证明,但是没有将证明上传的用户可能产生影响。

4)被调查用户。被调查用户需要在填写问卷调查前需要向调查发起人确认身份,审核通过后,发起人会将用户提交的公钥添加到用户公钥集合中。在调查发起以后,用户可以从智能合约中获取到所有的公钥,如果之前提交的公钥在其中,表示可以参与调查问卷。为了隐藏自己的身份,用户需要选取合适数量的公钥组成混淆公钥集合,将该集合作为输入,执行 $Prove(params, w, x, M)$ 算法生成证明。

3.2 实施过程

本文设计的方案中,问卷调查的实施、问卷结果的统计均运行在区块链上,使用 ZoKrates-0.6.1 工具包构建零知识证明,消息摘要算法使用 SHA-256 算法。方案的具体实施总共分成 9 个部分,图 1 展示了具体流程,下面是对流程的详细介绍。

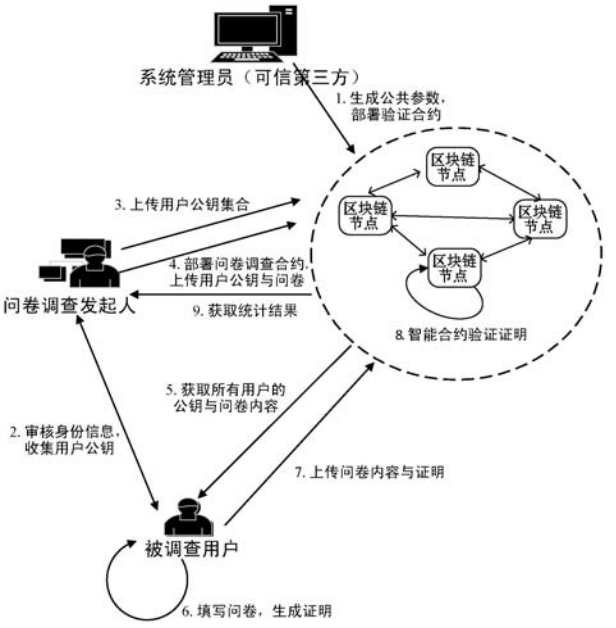

图 1 系统实施过程

1)系统管理员执行 $Setup(1^n)$ 算法,生成系统的公共参数。系统管理员需要将公共参数上传至公开位置供后续过程使用,同时将验证合约部署至区块链

2)调查发起人与被调查用户进行交互,审核被调查用户的身份信息,以及用于这次调查的公钥等信息。被调查用户可以使用直接的以太坊账号作为自己的公钥,也可以使用新生成的公钥,必须保证公钥在这次调查中唯一,且公钥对应的私钥只有被调查用户知道,这是用户身份不被泄露的一大保证。

3)调查发起人在收集了众多公钥信息后,将被调查用户的公钥集合 S_{all} 上传到智能合约中保存,这些公钥会被用于之后的验证问卷结果的合法性。

4)调查发起人发布问卷,为了保证问卷的公开,不可篡改,系统要求调查发起人将问卷内容上传到智能合约中,如果发起人不希望用户填写的问卷结果被公开,可以一同上传一个加密公钥,被调查用户可以使用该公钥加密填写的问卷答案,保证除了问卷发起人,其他用户不能解密该问卷内容。同时需要发布该问卷的 *id* 与 *nonce*,*id* 与 *nonce* 唯一标识该问卷,能够保证用户为当前问卷生成证明不会被用于其他问卷调查。

5)被调查人从智能合约获取当前的公钥集合 S_{all},问卷题目,以及问卷 *id* 与 *nonce*。拥有公钥集合中任意一个公钥对应的私钥都可以生成有效的证明,参与该问卷调查。

6)用户填写问卷,根据需要进行选择是否对填写

的内容进行加密。在填写完问卷后需要根据匿名性要求选取混淆公钥集合 S_{mix},该集合满足 $S_{mix} \subseteq S_{all}$。执行生成证明算法 $Prove(params, w, x, M)$,其中输入参数证据 w 包括 S_{mix}、sk、$token$、id、$nonce$。

$$M = \{token, params \mid \exists w = (S_{mix}, sk, token, id, nonce),$$

$$\text{s.t.} \quad F(sk) = pk \ \&pk \in S_{mix} \& \tag{14}$$

$$token = hash(sk \ \| id \| \ nonce)\} \tag{15}$$

7) 用户将构建的证明、填写的问卷内容,以及部分证据一同上传至调查合约,上传的证据包括 sk、$token$、id、$nonce$。

8) 调查合约先检查提交的 $token$ 与 S_{mix} 是否有效,通过后调用验证合约验证用户提交的证明是否能通过验证。验证合约执行 $Verify(params, x, M, \pi)$ 算法,包括以下检查:

$$F(sk) == pk \text{ 且 } pk \in S_{mix} \tag{16}$$

$$token == hash(sk \ \| id \| \ nonce) \tag{17}$$

验证通过后,验证合约返回 true,问卷合约将 $token$ 添加到 $token$ 集合 S_{token} 中,并且保存该用户提交的问卷内容。验证流程如下。

步骤 1　检查 S_{mix} 中公钥是否有序,是则继续,否则返回 false。

步骤 2　令 left := 0,right := len(S_{mix}) − 1。

步骤 3　若 left > right 返回 false。

步骤 4　令 mid := (left + right)/2,pk $\in S_{mix}$ = [mid]。

步骤 5　计算 $token == hash(sk \ \| id \| \ nonce)$ 与 $F(sk) == pk$,若都成立,返回 true。

步骤 6　若 $F(sk) < pk$,令 left := mid + 1,跳转至 3。

步骤 7　若 $F(sk) > pk$,令 right := mid − 1,跳转至 3。

步骤 8　返回 false。

9) 调查发起人从智能合约处获取调查统计结果。

3.3　系统特性

本文提出的方法可以随时更新被调查用户集合。因为用户在构建证明时选择了公钥集合中的一部分作为输入,当然也可以选择集合中全部的公钥,因此,集合中增加元素不影响已有的证明的有效性。另外一种情况是调查发起人删除集合中的公钥,对于已经构建并验证通过的证明,这也是没有影响的,但是对于那些还没有验证通过,并且包含了被删除公钥的证明会有影响,这部分证明不再是有效的。系统允许更新用户公钥集合,是其他方案不具备的一个特性,本系统可以更灵活使用。系统对用户身份信息的隐藏不是通过对用户身份信息或填写的问卷内容进行隐藏,而是通过切断问卷内容与填写问卷的用户之间的关系。因此本文提出的方法可以处理任意格式的数据,不局限于文献[1-2]中提出的只能从已有选项中选择,在本文提出的方案中,用户可以添加任意长度的文本数据。本文的方案只需要一轮零知识证明,即用户在填写问卷时生成证明,在用户提交之后,不需要再与系统做交互,减少了用户的工作量,提高了系统的可用性。

4　系统实现

本文提出的方案均在以太坊上实现,通过以太坊提供的 web3.js 库部署与调用智能合约,如图 2 所示。针对大部分不熟悉区块链的用户,系统提供 Web 页面发起问卷,以及提交问卷。被调查用户生成证明的过程需要输入用户私钥,在 Web 页面上进行可能造成用户私钥泄露,因此系统提供 Ubuntu 环境下的可执行文件用于离线生成证明。用户在本地生成证明后将证明通过 Web 页面调用智能合约进行验证。系统界面如图 3 - 图 8 所示。

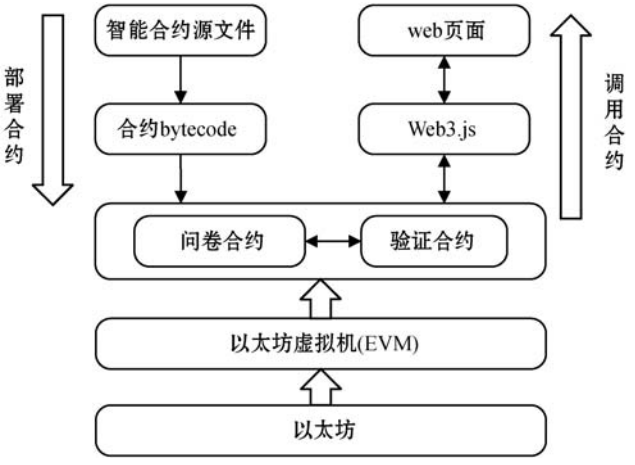

图 2　部署与调用合约

图 3　调查发起人登录页面

图 4　问卷内容与参与调查用户公钥设置

图 5 查看被调查用户公钥集合

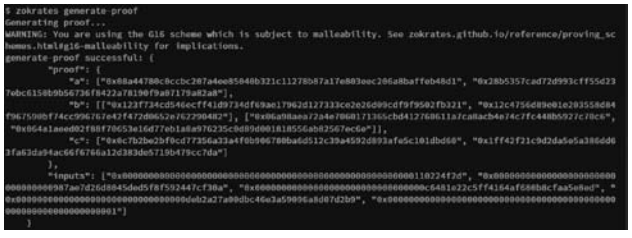

图 6 用户本地生成证明

图 7 提交证明与问卷

图 8 查看问卷结果

调查发起者的流程包括:登录问卷系统;收集用户公钥信息;发起问卷调查,确认问卷题目,参与调查的用户公钥集合;获取问卷统计结果。

被调查用户的流程包括:查看问卷题目;本地生成证明;提交填写的问卷回答与证明;查看是否提交成功,被统计在内。

5 可用性分析

用户生成证明时通过混淆公钥集合 S_{mix} 来混淆自己的公钥,集合中公钥数量越多意味着更多的验证工作。验证证明的工作由以太坊上智能合约完成,因此验证需要支付一定的以太坊费用(gas),在以太坊上对实际消耗的 gas 进行了测算,结果如图 9 所示,在 S_{mix} 包含 100 个公钥时,消耗 176 345 gas。以太坊主网上单个区块的可提供消耗的 gas 为 12 500 000,每个区块可以包含最少 70 次验证交易。按照每 gas 的单价为 10gwei 设置,一笔验证证明的交易需要 0.001 8 个 Ether,当前 2020 年价值为 2.34 美元。未来的工作将系统构建在其他链上计算能力更强的区块链上,进一步提高系统的处理能力。

同时,zk-SNARKs 的公共参数(验证 key,证明 key),生成的证明大小都与混淆公钥集合 S_{mix} 大小有关,我们将结果进行了测算,结果如表 1。生成的证明大小只有几 KB 大小,用户调用验证合约时不会造成很大的 gas 消耗。zk-SNARKs 的公共参数基本不随集合大小变化,本文实现的系统中有 22 MB,该参数由 zk-SNARKs 初始化算法生成,只需在系统启动时生成一次,用户也只需下载一次,对存储资源要求不高。

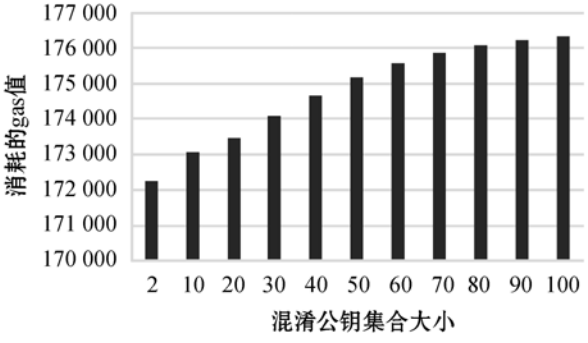

图 9 混淆公钥集合大小与消耗 gas 关系

表 1 混淆公钥集合大小与证明大小关系 单位:B

混淆公钥集合大小	zk-SNARKs 公共参数		证明 π 大小
	证明密钥	验证密钥	
2	22 973 656	2 206	1 375
10	22 975 256	3 518	1 967
20	22 976 536	5 158	2 707
30	22 977 816	6 798	3 447

续表 1

混淆公钥集合大小	zk-SNARKs 公共参数		证明 π 大小
	证明密钥	验证密钥	
40	22 979 096	8 438	4 187
50	22 980 376	10 078	4 927
60	22 981 656	11 718	5 667
70	22 982 936	13 358	6 407
80	22 984 216	14 998	7 147
90	22 985 496	16 638	7 887
100	22 986 776	18 278	8 627

6　结　语

本文主要关注了网络调查的匿名性问题,针对当前网络调查中出现的匿名性不高的问题,用非交互式零知识证明隐藏用户的身份,同时使用 SHA-256 算法生成用户针对当前问卷的唯一 token,避免一个用户多次提交问卷。系统将问卷分发、填写、统计过程放在区块链上,由智能合约执行,保证了问卷过程公开透明。从实验测算的数据中可以看出,本文提出的方案,在支持多达 100 个混淆公钥也不会造成很大的花销,对用户的身份信息做了很好的隐藏。通过合理选取混淆公钥集合的元素,可以在各种不同用户数量的场景下隐藏用户的身份信息。

参　考　文　献

[1] McCorry P, Shahandashti S F, Hao F. A smart contract for boardroom voting with maximum voter privacy[C]//International Conference on Financial Cryptography and Data Security,2017:357 – 375.

[2] 颜春辉,游林. 基于区块链的安全投票系统设计与实现[J]. 通信技术,2018,51(8):1979 – 1989.

[3] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system [EB/OL]. [2020 – 04 – 25]. https://bitcoin. org/bitcoin. pdf.

[4] Chillotti I, Gama N, Georgieva M, et al. A homomorphic LWE based E-voting scheme[C]//Post-Quantum Cryptography,2016:245 – 265.

[5] Peng K, Aditya R, Boyd C, et al. Multiplicative homomorphic E-voting[C]//International Conference on Cryptology in India,2004:61 – 72.

[6] Fan X Y, Wu T, Zheng Q H, et al. HSE-Voting: A secure high-efficiency electronic voting scheme based on homomorphic signcryption[J]. Future Generation Computer Systems, 2020,111:754 – 762.

[7] Kumar M, Katti C P, Saxena P C. A secure anonymous E-voting system using identity-based blind signature scheme [C]//International Conference on Information Systems Security,2017:29 – 49.

[8] Wu Y. An E-voting system based on blockchain and ring signature[D]. Birmingham:University of Birmingham,2017.

[9] Islam N, Alam K M R, Tamura S, et al. A new E-voting scheme based on revised simplified verifiable re-encryption mixnet[C]//International Conference on Networking, Systems and Security,2017:12 – 20.

[10] Aditya R, Lee B, Boyd C, et al. An efficient mixnet-based voting scheme providing receipt-freeness[C]//International Conference on Trust, Privacy and Security in Digital Business,2004:152 – 161.

[11] Lee B, Boyd C, Dawson E, et al. Providing receipt-freeness in mixnet-based voting protocols[C]//International Conference on Information Security and Cryptology,2003:245 – 258

[12] Buterin V. A next-generation smart contract and decentralized application platform[BE/OL]. [2020 – 04 – 25]. https://the-blockchain. com/docs/Ethereum_white_paper-a_next_generation_smart_contract_and_decentralized_application_platform-vitalik-buterin. pdf.

[13] Gritzalis D A. Principles and requirements for a secure E-voting system[J]. Computers Security,2002,21(6):539 – 556.

[14] Liu Y N, Zhao Q Y. E-voting scheme using secret sharing and K-anonymity[J]. World Wide Web,2019,22(4):1657 – 1667.

[15] Gutub A, Al-Juaid N, Khan E. Counting-based secret sharing technique for multimedia applications[J]. Multimedia Tools and Applications,2019,78(5):5591 – 5619.

[16] Goldwasser S, Micali S, Rackoff C. The knowledge complexity of interactive proof systems[J]. SIAM Journal on Computing,1989,18(1):186 – 208.

[17] Santis A, Persiano G. Zero-knowledge proofs of knowledge without interaction[C]//33rd Annual Symposium on Foundations of Computer Science,1992:427 – 436.

[18] Eberhardt J. ZoKrates—A toolbox for zkSNARKs on Ethereum [EB/OL]. [2020 – 04 – 25]. https://github. com/zokrates/zokrates.

[19] Sasson E B, Chiesa A, Garman C, et al. Zerocash: Decentralized anonymous payments from bitcoin[C]//IEEE Symposium on Security and Privacy,2014:459 – 474.

[20] 张亮,刘百祥,张如意,等. 区块链技术综述[J]. 计算机工程,2019,45(5):1 – 12.

[21] Augusto L, Costa R, Ferreira J, et al. An application of Ethereum smart contracts and IoT to logistics[C]//International Young Engineers Forum,2019:1 – 7.