TTShuffle：一种区块链中基于两层洗牌的隐私保护机制 *

程其玲[1,2]，金 瑜[1,2†]

(1. 武汉科技大学 计算机科学与技术学院，武汉 430065; 2. 湖北省智能信息处理与实时工业系统重点实验室，武汉 430065)

摘 要：区块链隐私保护技术中的去中心化混币机制 CoinJoin 和 CoinShuffle，分别因为代理节点不可信和节点需要层层传递加密而存在安全或效率低的问题，因此提出了 TTShuffle，一种两层的隐私保护机制。首先通过将所有混币节点分成不同的组，通过层层加密在组内进行第一层洗牌，保护组内交易的敏感数据；然后由每个组派一个节点作为本组的代表，进行第二层的组间洗牌，保护所有组的敏感数据得到最后完整的交易内容。通过安全和实验分析表明，TTShuffle 中 40 个混币参与者混币时间低于 10s，相比 CoinShuffle 机制提高了混币的效率，减少了混币时间，也确保了混币过程的隐私安全。

关键词：区块链；混币机制；隐私保护；洗牌；两层

中图分类号：TP399 doi: 10.19734/j.issn.1001-3695.2019.11.0653

Ttshuffle: privacy protection mechanism based on two-tier shuffling in blockchain

Cheng Qiling[1,2], Jin Yu[1,2†]

(1. *Dept. of Computer Science & Technology*, *Wuhan University of Science & Technology*, *Wuhan* 430065, *China*; 2. *Hubei Province Key Laboratory of Intelligent Information Processing & Real-time Industrial*, *Wuhan* 430065, *China*)

Abstract: In the block chain privacy protection technology, the decentralized coin mixing mechanism, CoinJoin and CoinShuffle, has the problem of low security or efficiency because the agent node cannot be trusted and the node need to passes encryption layer by layer, respectively. Therefore, this paper proposes TTShuffle, a two-tier privacy protection mechanism. Firstly, all mixed currency nodes are divided into different groups, and the first layer is shuffled in the group through layers of encryption to protect the sensitive data of intra-group transactions; Then, each group sends a node as the representative of the group, and carries out the second layer of inter-group shuffling to protect the sensitive data of all groups to obtain the final complete transaction content. Through security and experimental analysis, it is shown that the mixing time of the 40 participants in TTShuffle is less than 10s, which improves the mixing efficiency, reduces the mixing time and ensures the privacy security of the mixing process compared with the CoinShuffle mechanism.

Key words: blockchain; mixing mechanism; privacy protection; shuffle; two tier

0 引言

自比特币[1]产生后，具有去中心化、去信任化等特点的区块链技术得到了研究者的广泛关注[2]，但随着区块链技术的不断发展和广泛应用，其面临的隐私泄露问题也越来越突出。因为在区块链中，其上所有交易记录在全网都是透明的，其中的敏感数据(如金额，地址等)会有泄露的风险。目前，混币机制是解决该问题典型的方法[3,4,5]，其中 CoinJoin[6]和 CoinShuffle[7]机制是典型的去中心化的混币机制。CoinJoin 技术原理是将原来一笔简单交易变成由多笔交易组合成新的交易，并寻找一个代理节点来完成这一工作。该方案的最大优点是实现简单，每次混币过程仅由一个代理节点完成，不会对区块链原有的共识机制产生影响，但是仍存在缺陷：代理节点可能不可信，因为通常选择混币发起者为代理节点，如果代理节点是恶意的，就会泄露其他节点的隐私或者可能中断服务，造成系统无法正常工作[8]。针对此缺陷，出现了一种去中心化的混币协议 CoinShuffle 。CoinShuffle 方案在 CoinJoin 基础上设计一种输出地址洗牌机制，能够在不需要第三方的条件下完成混币过程，还能保证混币参与方不知道其他交易方的对应关系。但 Coinshuffle 存在明显不足：效率低，参与交易的所有节点都参与混币服务，混币过程冗长，时间复杂度高[9]。

基于此，本文提出了一种基于两层洗牌的隐私保护机制(two-tier shuffle，TTShuffle)。TTShuffle 结合了 CoinJoin 和 CoinShuffle 方案的优势，通过分组的形式在组内完成第一层洗牌；洗牌后每组的最后一个节点再次进行第二层洗牌将所有参与节点的输出地址组合起来。两层洗牌的分工合作提高了混币的效率，也确保了混币过程的隐私保护。

1 相关工作

1.1 区块链交易隐私

区块链技术中存储交易信息的全局账本是公开的，任何加入区块链网络中的节点都可以获取完整的数据，且区块链中每一笔交易的输入地址都是来自于前一笔交易的输出地址，通过分析全局账本中的交易记录，攻击者有可能对用户的隐私进行威胁。目前已经有很多研究通过分析区块链交易，推测区块链用户的隐私信息。例如，基于 Reid 和 Harrigan[10]通过对公布的账户进行数据分析，能够获取公布的比特币地址的资金金额、来源以及流向。Ron 和 Shamir[11]通过研究比特币交易的统计数据，发现了 364 个单笔交易大于 50000BTC 的这些大额交易的一些交易规律，发现这种大额交易会采用多种方式将资金分散到不同账号。Meiklejohn 等人[12]使用启

收稿日期：2019-11-13；**修回日期：**2019-12-15 **基金项目：**国家自然科学基金资助项目(61602351，61502359)

作者简介：程其玲(1995-)，女，湖北红安人，硕士研究生，主要研究方向为区块链隐私保护；金瑜(1973-)，女(通信作者)，湖北武汉人，副教授，博士，主要研究方向为云计算、对等计算和信任模型(jinyu@wust.edu.cn).

发式的聚类分析技术分析区块链中的交易数据，能够识别出属于统一用户的不同地址等。

针对区块链的隐私问题，出现了一种在不改变交易结果的前提下改变交易过程的混币机制[13]。根据混币过程中有无第三方节点参与，将现有的混币机制分为基于中心节点的混币方法和去中心化的混币机制。

a) 基于中心节点的混币机制的核心特点是混币过程主要由第三方节点执行。若参与混币的用户想从输入地址向输出地址转账，在基于中心节点的混币机制中会先向第三方转账，过段时间第三方再向输出地址转账，从而实现了将资金从输入地址向输出地址转移的目的。这个混币机制，一定程度上让攻击者无法知道转移给输出地址的资金是从哪个地址转移过来的，隐藏了输入地址和输出地址之间的关系。但是完全依赖于第三方节点存在严重的安全问题等缺陷，为了尽量保证第三方的可信度，Bonneau J 等人[14]提出了改进的中心化混币机制—Mixcoin。若第三方存在违规行为，用户可以通过审计功能公布签名数据举报第三方，第三方将会失去信誉，但是缺点是此类方案没有从根源上解除第三方对信息泄露的威胁。Valenta L 等人[15]在 Mixcoin 的基础上使用盲签名技术进一步优化，设计了 Blindcoin 方案，能够使第三方节点在进行混币过程中，无法获取所有交易双方的真实信息，从而避免第三方信息泄露风险。但是使用盲签名技术会增加混币过程的计算量。ShenTu Q C[16]等人提出了一种更高效的盲签名混币方案，使用椭圆曲线加密算法提升计算效率。

b) 去中心化混币机制的核心特点是混币过程不需要第三方节点节点执行。最早出现的去中心化混币方案是 Gregory Maxwell 提出 CoinJoin 机制,核心思想是通过将多个交易合并成一个交易，隐藏输入地址与输出地址之间的链接关系，形成多输入多输出交易，即将比特币中原来的多笔单输入-单输出的交易封装成一笔多输入-多输出的交易，使用代理节点(如混币交易的发起者)来完成这一工作。因此攻击者无法直接看出每笔输入地址对应的输出地址的关系。整个方案避免了中心化混币机制的一些隐私泄露问题，但是 CoinJoin 机制过于依赖代理节点,因此仍不可避免中心化混币的一些威胁，如果该节点不诚实就会泄露其他节点的隐私，或者进行违规操作使系统无法正常工作，存在安全隐患。

针对 CoinJoin 机制的缺陷，Ruffing T 等人[7]提出了一种完全去中心化的比特币混币协议—CoinShuffle，在 CoinJoin 的基础上增加了将输出地址洗牌的机制，使混币参与者无法得到自己以外的交易地址链接关系。在 CoinShuffle 中参与者 i 获取消息后解密并将自己的输出地址 m_i 通过所有参与者的公钥，一层一层加密得到 $Enc(ek_{i+1},...,Enc(ek_n,m_i))$ 后随机插入到解密列表中发送给参与者 $i+1$，参与者 $i+1$ 重复前面的解密加密操作发送给下一个参与者，直到最后一个参与者 n 得到完整的输出地址列表，并将输出地址列表并广播给其他参与者验证签名。当参与者获得每个参与者的签名，每个参与者就能够单独创建混币交易的完整签名版本，交易就成立了，反之进入责备阶段。

CoinShuffle 很好的解决了 CoinJoin 中的隐私安全缺陷，只有参与者自己知道自己的地址，在消息传递过程中无法获取其他参与者的输出地址与输入地址之间的链接关系，但是此类方案也是存在明显不足：效率低，参与交易的所有节点需要等待前一个参与者洗牌结束才能进行下一步洗牌，当参与混币操作的人数过多时，混币过程冗长，时间复杂度高。

根据以上分析，TTShuffle 根据 CoinJoin 和 CoinShuffle 机制的优势，通过分组的两层洗牌，能够有效解决 CoinJoin 和 CoinShuffle 机制中的缺点。

2　TTShuffle

TTShuffle 是在 CoinJoin 和 CoinShuffle 机制的基础上提出的，结合 CoinJoin 中的代理以及 CoinShuffle 机制的洗牌，设计了两层洗牌机制。

2.1　机制概述

区块链中基于两层洗牌的隐私保护机制主要部分大致可以分为四个阶段(见 2.2 节详述)。一组用户共同创建一个混合交易，每个用户都可以单独验证自己的交易是否包含在整个交易中。如果协议未成功运行，将进入额外的责备阶段，识别和排除至少一个行为不端的参与者，其他参与者可以在没有行为不端的参与者的情况下再次运行机制。

首先，参与者公布他们的输入地址，同时广播其生成短暂的密钥 (ek,dk) 对中的公钥加密密钥 ek。其次参与人员协商分组，确保所有的参与者都知道对应的小组。分好组后每个小组从第一个参与人员开始洗牌，使用一开始获取的其他节点的加密密钥 ek 将自己的输出地址层层加密发送给小组中下一个节点，当每个小组中最后一个参与节点将自己的输出地址随机插入解密后的输出地址列表后，第二层洗牌开始。第一组的最后一个节点将自己小组完整的输出地址列表通过加密密钥层层加密后发送给第二组的最后一个节点，第二组的最后一个节点同样将自己组的完整输出地址列表层层加密随机插入打乱后的解密消息中并发送给第三组最后一个节点，依此类推，当最后一个组的最后一个节点将自己组的完整输出地址列表随机插入打乱后的输出地址中后，广播最后的输出地址列表。最后，参与者检查自己的输出地址是否在列表中并签名，若输出地址丢失则事务无效并且参与者进入责备阶段并找出违规者。

2.2　详细描述

在下文中，详细描述了每个阶段。假设每一个混币的参与者在某个比特币地址上持有相同数目的货币，并以该地址作为混合事务中的输入地址之一，并用与其输入地址对应的签名密钥进行签名，在整个混币过程中，规定每个阶段的操作都带有签名的消息传递，用以在后期的操作中方便查找，并使用 $\delta_{a,b}$ 来表示节点 a 在阶段 b 的签名。

算法主要过程包含密钥的生成，消息体的构造发送，AES 的 CBC 模式的消息加密，解密，ECDSA 签名，验证等这几个部分。

阶段一　Anncouncement:参与混币的用户 $i\in\{1,2,3,...,N\}$ 公布其输入地址 vk_i，同时每个节点生成一个新的短暂的加密解密密钥对 (ek_i,dk_i)，并将其生成的加密密钥 ek_i 和签名　广播给所有参与人员，τ 表示新会话标识符，sk_i 表示成员 i 的签名密钥。在其他节点收到 i 广播的消息后，通过获取的输入地址 vk_i 验证签名 $\delta_{i,1}$，即 $Verify(vk_i,\delta_{i,1})$，来验证用户 i 的输入地址 vk_i 上是否有足够的比特币来进行混币交易，若没有，用户 i 将进入责备阶段。

阶段二　Shuffling1:如图 1 所示。参与人员通过协商将 N 个参与人员分为多个小组(本文中分了 5 组，每组 5 个成员)。N 个成员都分好组后从每个小组的第一个成员开始进行输出地址第一层洗牌操作。如图 1 中第一组成员{A1,A2,A3,A4,A5}，从 A1 开始，A1 将自己的输出地址 A1’通过获取的小组其他成员的 ek_i 层层加密得到 $m_1=Enc(ek_2,Enc(ek_3,Enc(ek_4,Enc(ek_5,A1'))))$，将　和签名 $\delta_{1,2}=Sig(sk_i,(m_1,2,\tau))$ 发送给小组的第二个成员 A2，A2 收到消息后先使用自己的解密密钥 dk_2 解密获取的消息中的 m_1 得到 $c1=Dec(dk_2,m_1)$，同时将自己的输出地址 A2’同样的使用剩余的参与者 A3、A4、A5 的加密密钥 ek_i 层层加密得到 $c2(ChiperText)=Enc(ek_3,Enc(ek_4,Enc(ek_5,A2')))$ (加密结构如图 2 所

示)，然后将 $c2$ 和 $c1$ 随机打乱后形成 m_2 (Group Node Message，其数据结构如图 3)，连着签名 $\delta_{2,2}=Sig(sk_1,(m_2,2,\tau))$ 发送给小组的第三个成员 A3，依此类推，当小组的最后一个成员 A5 获取到消息后使用解密密钥解密 dk_5 得到输出地址列表，然后将输出地址列表中的地址随机打乱的同时插入自己的输出地址 A5’后得到完整的输出地址列表 $OutA=\{A2',A3',A4',A5',A1'\}$ (其他小组同样操作)。

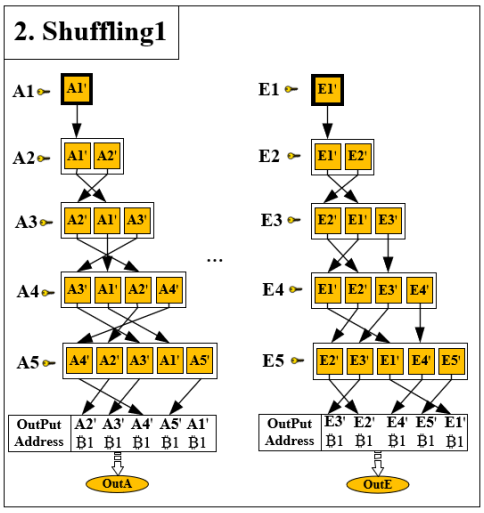

图 1　TTShuffle 机制中阶段二的一层洗牌过程

Fig. 1　The first layer shuffle of stage 2 in the ttshuffle mechanism

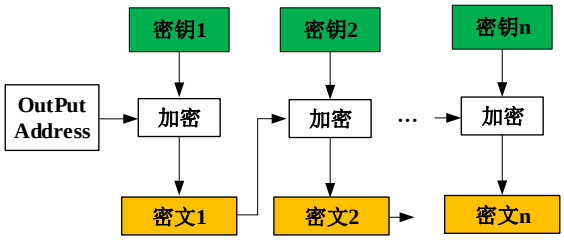

图 2　加密结构

Fig. 2　Encryption structure

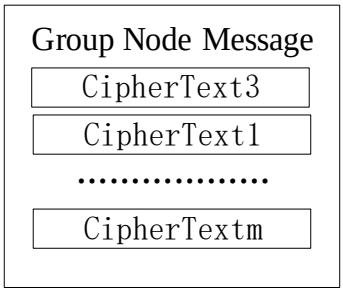

图 3　阶段二中 Group Node Message 结构

Fig. 3　The Group Node Message structure in stage 2

所以在此阶段每个节点 $i+1$ 都会验证前一个节点 i 发送的签名 $\delta_{i,2}=Sig(sk_i,(m_i,2,\tau))$，验证不通过会进入责备阶段并寻找出操作违规的节点并重新进行一层洗牌。

阶段三　Shuffling2:如图 4 所示。当 M 个小组的最后一个节点一层洗牌都完成后，第 j 组 $j\in\{1,2,3,...,M\}$ 的最后一个节点，此处的 ek_j 、 sk_j 均表示第 j 组的最后节点的加秘密钥、签名密钥，使用其他小组 $k>j$ 的最后一个节点的加密密钥 ek_k 来创建其输出地址列表的多层加密，进行第二层洗牌(洗牌流程见算法 1，其中一层洗牌流程同二层洗牌流程相同)。从第一个小组的最后一个节点开始层层加密自己的输出地址列表 Out_1 得到 $mList1=Enc(ek_2,...,Enc(ek_{M-1},Enc(ek_M,Out_1)))$ 和签名 $\delta_{1,3}=Sig(sk_1,(mList1,3,\tau))$ 发送给第二组的最后节点，第二组的最后节点收到消息后解密得到 $c1$，然后将自己的输出地址列表 Out_2 层层加密得到 $c2=Enc(ek_3,...,Enc(ek_{M-1},Enc(ek_M,Out_2)))$ 然后将 $c1$ 和 $c1$ 随机打乱得到 $mList2$ (Lsat Node MessageList 结构同图 3 的 Group Node Message 结构相同)连着签名 $\delta_{2,3}=Sig(sk_2,(mList2,3,\tau))$ 发送给第三组的最后节点。

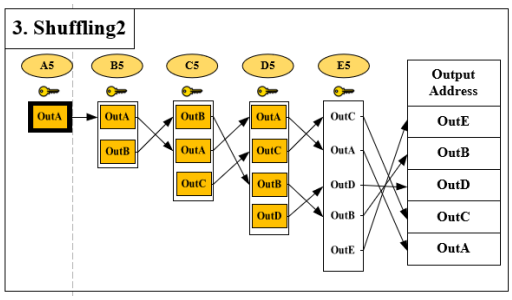

图 4　TTShuffle 中阶段三的二层洗牌过程

Fig. 4　The second layer shuffle of stage 3 in the ttshuffle mechanism

依此类推，后面的第 k 组的最后节点都希望获取其前面小组的最后节点 $k-1$ 的密文。第 k 组的最后节点接收到消息后，从消息体中通过解密去除最外层的一层加密，然后将自己输出列表 Out_k 通过其他剩余小组的最后节点的加密密钥层层加密，随机插入至打乱的解密后的列表中得到 $mListk$ ，再将 $mListk$ 和签名 $\delta_{k,3}=Sig(sk_k,(mListk,3,\tau))$ 送给第 $k+1$ 组的最后节点。每一个小组最后节点都按照协议操作，第 M 组的最后节点将会产生一个完整的输出地址列表 AddressLists，然后最后节点广播 AddressLists 给所有混币参与者进行验证签名。

算法 1　洗牌流程

输入：mList1　　// mList1 表示获取的加密消息

输出：mList2　　// mList2 表示发送的加密消息

//有 n 个小组，此处是第 i 组最后节点收到第 i-1 组最后节点的消息

1) mList1={$m_1,m_2,m_3,\ldots,m_{i-1}$};

2) 使用解密密钥 dk_i 对输入 mList1 数组进行解密，对每一个 j∈{0,…,i-2},有 sData[j]←AESdecrypt(m_j,dk_i,dk_i);

3) Out←Out_i; // Out_i 表示第 i 组的输出地址列表

4) 使用每一个 k∈{i+1,…,n}组的最后节点的公钥层层加密 Out,得到 Out←AESencrypt(Out, ek_k, ek_k);

5) 将前面解密的数组 sData 和加密的密文 Out 重新排列组合成一个新的数组列表，得到 mList2.add(sData [randam]||Out);

6) 返回 mList2。

阶段四　Verification:如图 5 所示。混币参与者收到最后小组的最后节点的广播后，会验证自己的输出地址是否在输出地址列表中，如果存在，则使用其输入地址的签名密钥进行签名并广播。每个参与者接收到所有参与节点签名后，就可以单独创建一份完整的混合交易，此时交易有效，可以提交给比特币网络。若不在，系统将进入责备阶段，查找操作失误节点。

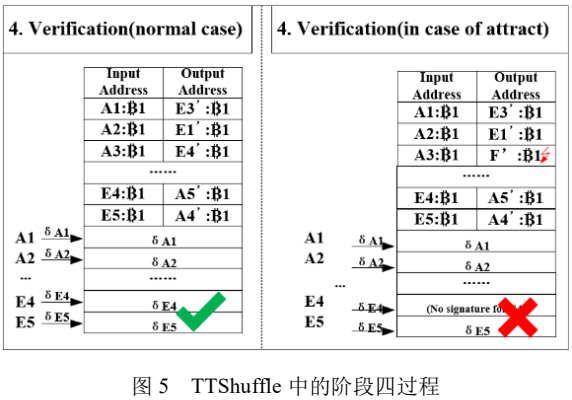

图 5　TTShuffle 中的阶段四过程

Fig. 5　The stage 4 in the ttshuffle mechanism

责备阶段　Blame:在 TTShuffle 的四个步骤中，每一步中的参与者都是会监督是否有违规现象。为了能够快速查找出混币过程中出现违规现象是出现在哪一阶段，采用了 CoinShuffle 的方法，在每一阶段，操作者都会有签名，如果出现了违规现象，参与的诚实节点会通告并进入责备阶段，然后查找出行为不端者，随后将违规者剔除。在混合协议中主要讨论三种情况：

a) 在 Announcement 阶段出现行为不端者。在此阶段主要是参与节点的加密密钥以及输入地址的广播，需要确保每一个参加混币的节点参与其中。若其中操作违规者，发现者可以通过签名查找出违规者并广播违规情况进行处理。

b) 参与者在 Shuffling1 阶段未正确执操作。此阶段主要是小组内的洗牌操作，违规操作包括发现错误的加密密钥 ek、层层加密地址不正确等。若出现违规操作，发现违规的节点会广播接收到的消息以及解密密钥 dk 进入责备阶段，所有参与的小组节点验证并剔除违规节点的输出地址。

c) 参与者在 Shuffling2 阶段未正确执行。一般出现这种现象是出现在每个小组的最后节点的二层洗牌之间，同 Shuffling1 阶段一样，发现问题的最后节点会公布短暂的解密密钥 dk 以及收到的消息进行公布广播，参与的最后节点一起验证查找出操作违规的节点。

3　分析

3.1　安全性分析

为了确保参与混币的节点不会泄露其他节点隐私信息，TTShuffle 机制中加密部分仍是沿用了 CoinShuffle 机制中的 AES-CBC 加密，确保了混币参与者除了知道自己的输出地址外，并不知道其他参与者的输出地址是什么，从而确保了参与者无法将其他参与者的输入地址与输出地址对应。

TTShuffle 机制中的阶段二是小组内部的第一层洗牌，因为输出地址的层层加密，每个参与者获取的消息只能使用自己的解密密钥解开消息的最外层，无法获取最内层的输出地址信息，直到小组最后一个参与者解密得到本小组的所有输出地址，但是无法知道输出地址都是谁的。阶段三的第二层加密是每个小组的最后节点间进行的组间洗牌，主要是为了将所有参与混币的所有输出地址整合成一个大的交易单，层层加密和随机打乱，再次增加了输出地址的保密性。

TTShuffle 机制通过分组两层洗牌，保证了混币参与者的输入地址和输出地址的不可连接性，同时也避免了 CoinJoin 机制的单点失效。

3.2　实验分析

3.2.1 实验环境

操作系统：64 位 windows10；编程语言：java；编程工具：Eclipse, jdk1.8.0_101；JXTA 版本：2.4；JXTA CMS 版本：2.4.1；加密 API：bcprov-jdk14。

本机制通过搭建 P2P(Perr-to-Peer)网络(比特币中的底层 P2P 网络)，建立多个节点并加入同一个 PeerGroup 中。节点间的加密主要还是采用 AES 的 CBC 模式，节点的签名主要是在 secp256k1 椭圆曲线上使用 ECDSA128 位的安全级别实现签名。

3.2.2 时间分析

本文机制包括四个阶段，为了测试 TTShuffle 机制的性能情况，记录了多个实验结果。

实验 1　固定每个小组有 5 个参与节点，通过增加小组数来实现处理多个混币参与者。通过递增小组数，即从 2、3、4、5、6、7、8 个小组组来统计 10、15、20、25、30、35、40 个混币用户在 TTShuffle 机制中完成所需要的时间和 CoinShuffle 机制运行所需要时间的对比，如图 6 所示。

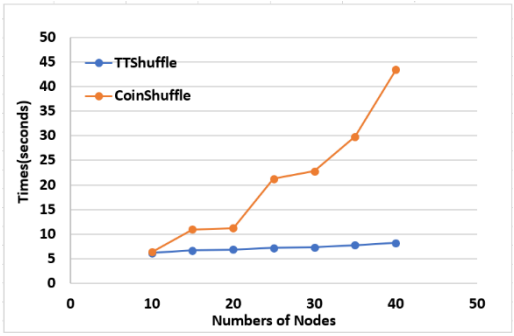

图 6　混币机制运行时间对比图

Fig. 6　The running time comparison diagram of the coin mixing mechanism

在图 6 中，CoinShuffle 在有 40 个参与者参与混币的情况下，大约需要 40s。在 TTShuffle 实验中，当固定每个小组的成员为 5 个节点时，随着总的混币参与人数的递增，40 名混币参与者在 TTShuffle 机制中完成混币所需时间不到 10s，且随着普通混币节点的增多，时间递增幅度不大，因为本实验中的混币节点数量是通过小组的数量来控制的，TTShuffle 通过分组合作，多个小组内的第一次洗牌是可以同时进行的，随着参与人数每次增加 5 个，每次第二层洗牌就多一个小组最后节点参与，所以处理时间随参与者线性增加。本方案相比 CoinShuffle 机制在混币效率上有很大的提升。

实验 2　进行混币用户的平均时间的比较，在实验 1 的基础下，计算分担到每个参与节点的平均混币时间。如图 7 是随着参与混币的用户数量递增时 TTShuffle 机制中节点所需平均时间同 CoinShuflle 机制中的节点平均时间的对比图。

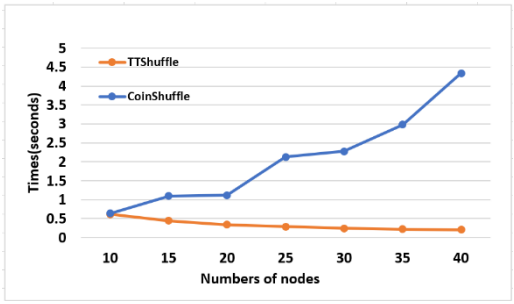

图 7　TTShuffle 机制中节点平均时间对比图

Fig. 7　Comparison diagram of node mean time in ttshuffle mechanism

在图 7 中，随着混币参与者的递增，在 CoinShuffle 机制中节点的平均时间明显高于 TTShuffle 机制中节点的平均时间，在平均时间上，CoinShuffle 机制是递增的，本文的 TTShuffle 机制是递减的。其原因是随着参与者的递增，CoinShuffle 机制中每个参与者都需要比前一个参与者多包含一层密文的密文向量；TTShuffle 在本实验中随着参与人数每 5 个的递增，也只是在第二次洗牌中多了一个节点，总时间并没有增加很多，所以每个参与者的平均计算时间反而会减少。在有 40 个参与者的情况下，平均计算时间只占总时间的 2.5%。

结合图 6 和 7 可以直观的看出，本文 TTShuflle 机制时间开销比 CoinShuffle 机制小很多，且参与混币的人数越多，前者的混币时间优势越明显。实验结果证明了 TTShuffle 机制的可行性，即使在参与人数众多的情况下也是如此。

4　结束语

4.1　总结

本文提出了一种区块链中基于两层洗牌的隐私保护机制，为了解决目前区块链中比较成功的混币机制 CoinJoin 机制中的不可信代理节点以及 CoinShuffle 机制中的混币时间过长

问题，采用了混币用户分组的模式，将混币操作分散出去，通过组内的洗牌以及组间的洗牌，既保证了混币过程中的隐私安全，也实现了混币的效率的提升。通过实验对比，本文提出的一种区块链中基于两层洗牌的隐私保护机制相对 CoinShuffle 机制而言，在不影响交易过程的隐私保护的情况下，显著减少了混币过程时间。

4.2 展望

虽然本文解决了部分 CoinJoin 机制和 CoinShuffle 机制中的缺陷，但由于时间和水平的关系，机制还有以下缺陷。

a) 由于 TTShuffle 机制中小组的最后一个节点相对小组其他节点而言，多出了部分工作量，所以应该需要设立一定的奖罚机制来促进 TTShuffle 机制的完整性，并为区块链的隐私保护服务，使其更完善。

b) 本文中的责备机制目前只研究了节点间的违规操作，在此阶段，可以深入研究怎么快速查找出违规节点，使 TTShuffle 机制在进入责备阶段时，能够快速找出问题并重新回到正轨。

参考文献：

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system [EB/OL]. (2008) [2019-11-13]. https://bitcoin. org/bitcoin. pdf.

[2] 袁勇, 王飞跃. 区块链技术发展现状与展望 [J]. 自动化学报, 2016, 42 (4): 481−494. (Yuan Yong, Wang Feiyue. Blockchain: The State of the Art and Future Trends [J]. Acta Automatica Sinica, 2016, 42 (4): 481−494.)

[3] Rivest R L, Shamir A, Tauman Y. How to Leak a Secret [C]// Proc of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology. Berlin: Springer-Verlag, 2001: 552-565.

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

[5] Gentry C. Fully homomorphic encryption using ideal lattices [C]// Proc of the 41st Annual ACM Symposium on Theory of Computing (STOC) . USA: Bethesda, 2009: 169-178.

[6] Maxwell G. CoinJoin: Bitcoin privacy for the real world [EB/OL]. (2013) [2019-11-13]. https://bitcointalk. org/index. php?topic=279249. 0.

[7] Ruffing T, Moreno-Sanchez P, Kate A. Coinshuffle: Practical decentralized coin mixing for bitcoin [C]// Proc of the 19th European Symposium on Research in Computer Security. USA: Springer-Verlag, 2014: 345-364.

[8] 祝烈煌, 董慧, 沈蒙. 区块链交易数据隐私保护机制 [J]. 大数据, 2018 (1): 46-56. (Zhu Liehuang, Dong Hui, Shen Meng. Privacy protection mechanism for blockchain transaction data [J]. Big Data Research, 2018 (1): 46-56.)

[9] 付烁, 徐海霞, 李佩丽, *et al.* 数字货币的匿名性研究 [J]. 计算机学报, 2019 (05): 1045-1062. (Fu Shuo, Xu Haixia, Li Peili, *et al.* A Survey on Anonymity of Digital Currency [J]. Chinese Journal of Computers, 2019, 42 (05): 1045-1062.)

[10] Reid F, Harrigan M. An analysis of anonymity in the bitcoin system [C]// Proc of the 3rd IEEE Int Conf on Privacy, Security, Risk and Trust. USA: IEEE Press, 2011: 1318-1326.

[11] Ron D, Shamir A. Quantitative analysis of the full bitcoin transaction graph [C]// Proc of the 17th International Conference on Financial Cryptography and Data Security. Berlin: Springer, 2013: 6-24.

[12] Meiklejohn S, Pomarole M, Jordan G, *et al.* A fistful of bitcoins: characterizing payments among men with no names [C]// Proc of conference on Internet measurement conference. New York: ACM Press, 2013: 127-140.

[13] Chaum D. Untraceable electronic mail, return addresses and digital pseudonyms [J]. Communications of the ACM, 1981, 24 (2): 84-90.

[14] Bonneau J, Narayanan A, Miller A, *et al.* Mixcoin: Anonymity for Bitcoin with accountable mixes [C]// Proc of the 18th Int Conf on Financial Cryptography and Data Security. Berlin: Springer, 2014: 486-504.

[15] Valenta L, Rowan B. Blindcoin: Blinded, Accountable Mixes for Bitcoin [C]// Proc of International Conference on Financial Cryptography and Data Security. Berlin: Springer, 2015: 112-126.

[16] Shentu Qingchun, Yu Jianping. A blind-mixing scheme for Bitcoin based on an elliptic curve cryptography blind digital signature algorithm [EB/OL]. (2015) [2019-11-13]. https://arxiv.xilesou.top/ftp/arxiv/papers/1510/1510. 05833. pdf.