密码学报 ISSN 2095-7025 CN 10-1195/TN
Journal of Cryptologic Research, 2018, 5(5): 556–567

E-mail: jcr@cacrnet.org.cn
http://www.jcr.cacrnet.org.cn
Tel/Fax: +86-10-82789618

区块链技术专刊

基于区块链的高效公平多方合同签署协议*

高 莹[1], 吴进喜[2]

1. 北京航空航天大学 网络空间安全学院, 北京 100191
2. 北京航空航天大学 数学与系统科学学院, 北京 100191

通信作者: 吴进喜, E-mail: mathwjx@163.com

摘 要: 随着数字货币发展的广泛应用, 区块链作为其核心支撑迅速成为关注的焦点. 由于区块链可充当一个去中心化的可信第三方, 因此在设计电子合同签署协议时, 可引入区块链来保证其公平性. 现有的基于区块链的合同签署协议大多只适用于两方的合同签署, 当考虑扩展为多方合同签署时, 由于签署方需要对每个签名进行验证, 验证工作量极大. 因此, 设计一个简单又高效的多方公平合同签署协议成为电子商务安全亟待解决的关键问题. 可验证加密签名 (verifiably encrypted signature, VES) 能够有效地保证互联网上交易过程的公平性, 很自然成为合同签署协议构造的一种常规技术手段. 聚合签名能够聚合多个签名为一个签名, 从而提高签名和验证的效率. 本文结合可验证加密签名和聚合签名, 提出无证书的聚合可验证加密签名方案 (CLAVES), 并给出一个具体的构造方案并证明其安全性. 利用该方案设计出基于区块链的多方合同签署协议. 该协议分两个阶段. (1) 在区块链链下阶段, 签署方执行 CLAVES 方案, 生成各自的 CLAVES 签名. 若所有的 CLAVES 签名验证通过, 则进入区块链链上阶段, 否则协议停止. (2) 在区块链链上阶段, 利用基于罚金的公平交换协议 (claim or refund, COR) 思想, 签署方在区块链上公平地交换各自的秘密值, 最后通过计算, 可提取出各方对合同的普通数字签名, 从而完成合同的签署. 通过分析和与已有的多方合同签署协议对比表明, 本文提出的多方合同签署协议具有高效性和公平性.

关键词: 区块链; 公平合同签署协议; 聚合签名; 高效; 隐私保护

中图分类号: TP309 **文献标识码:** A DOI: 10.13868/j.cnki.jcr.000265

中文引用格式: 高莹, 吴进喜. 基于区块链的高效公平多方合同签署协议[J]. 密码学报, 2018, 5(5): 556–567.

英文引用格式: GAO Y, WU J X. Efficient multi-party fair contract signing protocol based on blockchains[J]. Journal of Cryptologic Research, 2018, 5(5): 556–567.

Efficient Multi-party Fair Contract Signing Protocol Based on Blockchains

GAO Ying[1], WU Jin-Xi[2]

1. School of Cyber Science and Technology, Beihang University, Beijing 100191, China
2. School of Mathematics and Systems Sciences, Beihang University, Beijing 100191, China

Corresponding author: WU Jin-Xi, E-mail: mathwjx@163.com

* 基金项目: 国家重点研发计划 "现代服务业共性关键技术研发及应用示范" 重点专项 (2017YFB1400700)
Foundation: National Key Research and Development Program of China "Common Key Technology R&D and Application Pilot for Modern Service Industry"(2017YFB1400700)
收稿日期: 2018-08-02 定稿日期: 2018-09-21

Abstract: With the wide application of digital currency, blockchain as its core support has quickly become the focus of attention. Since the blockchain can act as a decentralized trusted third party (TTP), it is introduced to ensure fairness in the design of contract signing protocols. However, most of the existing protocols only apply to contract signing between two parties. When they are extended to multi-party contract signing protocols, the participants' workload becomes enormous because every signature needs to be verified. Therefore, designing a simple and efficient multi-party fair contract signing protocol is a key issue for e-commerce. The verifiably encrypted signature (VES) can effectively guarantee the fairness of the transaction process on the Internet and can be naturally used to design contract signing protocols. Aggregate signature can aggregate multiple signatures into one signature, thereby improving the efficiency of verification. Based on certificateless aggregate signature scheme and verifiably signature scheme, this paper proposes a certificateless aggregate verifiably signature scheme (CLAVES). We present a concrete construction of the scheme, and prove its security. Then we use this scheme to design a multi-party fair contract signing protocol based on blockchain. The proposed protocol is divided into two phases. (1) In the off-blockchain phase, the signing party executes the CLAVES scheme and generates their respective CLAVES signatures. If the verification of CLAVES signatures is valid, then protocol enters the on-blockchain phase. Otherwise, the protocol is terminated. (2) In the on-blockchain phase, using the penalty-based fair exchange protocol (claim or refund, COR), the signatories exchange their secret values fairly on the blockchain, and finally, they can extract the ordinary signatures by some computation. Thereby completing the signing of the contract. Through analysis and comparison with the existing multi-party contract signing protocols, the multi-party contract signing protocol proposed in this paper is efficient and fair.

Key words: blockchain; fair contract signing protocol; aggregate signature; efficiency; privacy preserving

1 引言

随着信息技术的发展, 如今签署方可通过网络交换各自的签名来完成合同签署. 由于网络的异步性, 合同签署过程也是异步, 这必然会带来一定程度的不公平性. 公平合同签署协议是公平交换协议的实际应用, 旨在保障签署方公平地交换合同的数字签名. 公平交换协议按有无第三方参与可分为两类: 第一类无需第三方参与. 比如基于逐步交换协议的思想 [1], 签署方需要逐步释放秘密值. 这类协议不仅存在不公平性且需要昂贵的通信费用. 另外一类依赖于可信第三方保证公平性 [2–4], 又可细分为在线的与离线的两类. 基于在线 (半) 可信第三方的协议 [2,3] 需要第三方始终在线, 所以在多用户系统中, 该协议的效率较低. 基于离线可信第三方的协议 [4] 可避免在线第三方的缺陷, 该协议的第三方只在签署方出现争端时, 参与解决争端问题, 但是该类协议的第三方是中心化的, 存在为了牟利与签署方合谋或泄露敏感信息的情况, 从而存在不公平性和隐私泄露的问题.

区块链是一种基于共识算法的技术, 使得在完全不信任的节点之间建立起信任机制 [5]. 因此, 区块链在功能上可以提供一个无中心化的可信第三方. 文献 [6–9] 基于区块链技术设计公平交换协议, 其核心思想是依赖罚金机制, 即诚实方可获得奖励, 而不诚实方将会受到罚金惩罚. 文献 [10] 利用公平交换协议设计电子投票方案, 利用罚金机制迫使投票者诚实地投票, 否则将会受到惩罚. 不仅如此, 区块链利用地址作为假名身份, 一定程度上保障用户身份隐私. 鉴于区块链的这两个特性, 基于区块链技术的公平合同签署协议的构造成为研究热点 [11–15].

值得注意的是, 基于区块链的公平合同签署协议在签署方出现争端时, 区块链上的节点都能提取出普通签名 (为了与可验证加密签名区别开, 本文将经典签名称为普通签名), 又会造成隐私泄露问题. 文献 [12] 构造了一种盲的可验证加密签名 (blind verifiably encrypted signature, BVES) 算法, 并提出一种基于公开区块链的隐私保护公平合同签署协议来解决隐私泄露问题. 签署方首先在区块链链下进行协商秘密因子, 用来盲化合同与身份信息; 然后在区块链链上实现公平交换 BVES 签名, 从而实现双方公平

合同签署协议. 在区块链上需要基于以太坊的智能合约或基于超级账本的链码来实现对 BVES 签名的验证. 文献 [13] 在文献 [12] 的基础上, 从实用性的角度出发, 设计了专门用于签署合同的数字货币 Contract Coin.

可验证加密签名 (verifiably encrypted signature, VES) 是保证合同签署公平性的重要工具. 文献 [4] 首次提出 VES 方案的思想, 该方案参与方包含: 签名者、验证者和仲裁者. 签名者利用仲裁者的公钥对普通签名进行加密, 由此生成 VES 签名. 验证者可以通过仲裁者的公钥来验证 VES 签名的有效性, 但是无法提取出普通签名. 当出现争端时, 验证者可以要求仲裁者来提取普通签名. 文献 [16] 基于短签名方案构造出一个 VES 方案, 由此开始, VES 方案得到广泛的关注. 因此, 研究人员相继提出不同的 VES 方案 [12,13,17,18]. 其中文献 [17] 基于离散对数, 提出一个实用的 VES 签名方案, 后续被文献 [15] 应用于设计公平合同签署协议. 文献 [18] 则是基于格理论, 提出一个新的 VES 签名方案.

在物联网中, 聚合签名的广泛应用, 使其迅速成为研究热点. 无证书聚合签名与传统基于 PKI 的聚合签名相比较, 既不需要管理公钥证书, 又不存在密钥托管的问题. 因此, 无证书聚合签名方案更具实用性. 文献 [19] 首次提出基于无证书的聚合签名方案来解决传统基于 PKI 聚合签名 [20] 中证书管理和密钥托管的问题. 随后基于高效和安全两个特性展开了大量的研究 [21–26].

文献 [21] 提出了一个新的无证书聚合签名方案, 并对其安全性进行证明, 但是该聚合签名方案的签名长度依赖于用户数量. 文献 [22] 构造的一个无证书聚合签名方案被指出是可以普遍伪造的 [23]. 随后, 基于聚合签名长度固定的想法, 文献 [24] 提出一个签名长度固定的聚合签名方案, 随即被指出该方案不能抵抗类型 II 敌手的攻击 [25]. 在此基础上, 文献 [26] 对该方案进行改进, 并提出一个新的签名长度固定的无证书聚合签名方案. 因此, 构造一个签名长度固定的无证书聚合签名仍旧是一个难点.

1.1 本文工作

本文的主要贡献如下:

(1) 将无证书聚合签名 (certificateless aggregate signature, CLAS) 方案 [16] 和 VES 方案结合起来, 提出无证书的聚合可验证加密签名 (certificateless aggregate VES, CLAVES) 方案, 并给出其安全模型;

(2) 基于具体的 CLAS 方案 [21], 构造出一个具体的 CLAVES 方案, 并证明了其安全性;

(3) 基于具体的 CLAVES 方案构造了基于区块链的多方合同签署协议, 该协议设计简单高效且满足公平性.

1.2 相关工作

现有公开工作中, 签署方大多数是两方. 当扩展为多方合同签署时, 若各签署方都需要对每个 (BVES)VES 签名进行验证, 则 (矿工) 签署方的验证工作量极大. 因此, 设计一个简单又高效的多方公平合同签署协议成为一个严峻的挑战.

文献 [15] 设计了一种基于区块链的三方合同签署协议, 各签署方首先在区块链链下协商共享一个门限公钥, 用于构造 VES 签名 [17]. 当成功验证所有签署方的 VES 签名之后, 在区块链链上实现公平交换秘密因子. 最后通过 VES 签名和秘密因子计算得出三方的普通数字签名. 该协议通过区块链链上的公平交换协议来保证公平性, 以及区块链链下的可验证加密签名方案保障签署方隐私. 由于在区块链链下阶段需要验证每一方 VES 签名, 随着签署方的人数增多, 计算开销将增大, 并且在区块链链上阶段实现 deposit 交易 [8] 过程较为复杂.

文献 [14] 在文献 [12] 的基础上, 以增强协议的隐私性为目标, 引入混币技术来设计公平合同签署协议, 并且可支持多方进行合同签署. 该协议分为两个阶段. 区块链链下阶段, 各签署方通过共享多个秘密因子, 来盲化合同及身份信息. 在区块链链上阶段, 签署方与各自选定的第三方进行合同签署, 即公平地交换签署方的 BVES 签名和第三方的临时密钥. 虽然该协议可拓展为多方合同签署, 但是在区块链链上阶段, 矿工的工作量随着签署方的增加而增大, 由此签署方所要支付的交易费也随之增多.

本文提出一个安全的具体的无证书的聚合可验证签名方案 (CLAVES), 利用这个方案设计了基于区块链的多方合同签署协议. 将签名和大量的签名验证工作放在区块链链下, 仅当所有的签名通过验证后才进入区块链链上, 使得在区块链链上阶段仅用来公平交换秘密值, 矿工只需要增加验证公私钥对是否匹配,

从而计算开销不随签署方人数的增多而增大. 同时, 通过分析和与已有的多方合同签署协议对比表明, 本文提出的多方合同签署协议简单高效且具有公平性.

2 预备知识

本节介绍预备知识. 在2.1节介绍双线性对的定义, 在2.2节介绍一种基于罚金的公平交换协议, 在2.3节介绍 CLAS 方案及其安全性模型, 在2.4节介绍 VES 方案及其安全模型.

2.1 双线性对

设 G_1 是加法循环群, P 为生成元. G_2 是乘法循环群, 且它们都是 q 阶. 将映射 $e: G_1 \times G_1 \to G_2$ 称为双线性对, 如果满足以下性质:

(1) 双线性. 对任意 $P, Q \in G_1$, $a, b \in Z_q^*$, 有 $e(aP, bQ) = e(P, Q)^{ab}$.

(2) 非退化性. 存在 $P, Q \in G_1$, 使得 $e(P, Q) \neq 1$.

(3) 可计算性. 对所有的 $P, Q \in G_1$, 存在一个有效的多项式时间算法来计算 $e(P, Q)$.

2.2 基于罚金的公平交换协议 (COR 协议)

本小节介绍在区块链上用于设计公平交换协议的功能函数 (claim or refund, COR) [8].

设 π 是布尔函数, COR 由一些交易构成. 其功能是确保接收方 Q 在时间 τ 内释放秘密值 w, 满足 $\pi(w) = 1$, 发送方 P 将会发 qB 给接收方 Q. 具体分为以下三个过程:

(1) Deposit phase

发送方 P 生成一个 deposit 交易, 价值为 qB. 其输入脚本验证为: 要么需要 P 和 Q 的签名; 要么需要 Q 的签名以及秘密值 w, 满足 $\pi(w) = 1$. 该交易暂时保密.

发送方 P 将 deposit 交易作为输入用于生成 refund 交易, 其时间锁为 τ. 然后将该交易发送给 Q, 并让 Q 对交易签名. 最后 P 也对该交易签名.

发送方 P 将 deposit 交易广播全网, 并记录在区块链上. 此时存款阶段完成.

(2) Claim phase

接收方 Q 将 deposit 交易作为输入用于生成 claim 交易. 在时间 τ 内提供自己的签名以及秘密值 w, 满足 $\pi(w) = 1$, 通过输出脚本的验证, 拿到 qB.

(3) Refund phase

若在时间 τ 后, 接收方 Q 没有广播有效的 claim 交易. 发送方通过广播 refund 交易拿回 qB.

综上, 将该 COR 协议简单记为: $\mathrm{P} \xrightarrow[q,\tau]{\pi} \mathrm{Q}$.

2.3 CLAS 方案

本节简要介绍 CLAS 方案及其安全模型 [19].

2.3.1 CLAS 方案定义

一个 CLAS 方案包含以下 6 个算法:

(1) Setup: 安全参数为 k, KGC 选取主密钥 λ, 公开系统参数 params.

(2) PartialPrivateKeyExtract: 输入用户身份信息 ID_i, 系统参数 params 和主密钥 λ, 输出部分私钥 D_i.

(3) UserKeyGen: 输入用户身份 ID_i, 随机选取 x_i, 输出用户的秘密值和公钥对 (x_i, P_i).

(4) Sign: 输入系统参数 params, 消息 M_i, 状态信息 Δ, 用户身份信息 ID_i, 私钥 (D_i, x_i) 及其公钥 P_i, 输出签名 σ_i.

(5) Aggregate: 聚合者执行该算法. 输入 n 个身份-消息-公钥-签名对 $(\mathrm{ID}_i, M_i, P_i, \sigma_i)$, 输出聚合签名 σ.

(6) AggregateVerify: 输入系统参数 params, 状态信息 Δ, 聚合签名为 σ, 用户身份信息 ID_i 及其公钥 P_i. 若聚合签名有效, 输出 true, 否则输出 false.

2.3.2 CLAS 方案的安全模型

与无证书的公钥密码体制相同, 文献 [21] 给出两种类型的敌手 $A \in \{A_{\mathrm{I}}, A_{\mathrm{II}}\}$, 其中敌手 A_{I} 是一个恶意的用户; 敌手 A_{II} 是一个恶意的密钥生成中心 (Key Generation Center, KGC). 具体如下:

类型 I: 敌手 A_{I} 没有主密钥以及用户的部分私钥, 但是他有能力随意替换任何用户的公钥.

类型 II: 敌手 A_{II} 知道主密钥以及用户的部分私钥, 但是他既不知道用户的秘密值, 也不能随意替换任何用户的公钥.

攻击者 A_{I} 和 A_{II} 分别和挑战者 C 进行模拟游戏 Game1 和 Game2, 具体游戏过程参考文献 [21]. 对于未进行签名询问的消息, 若攻击者 A_{I} 能输出一个在自己替换新的公钥下的有效消息签名, 则 A_{I} 攻击成功; 若攻击者 A_{II} 能输出一个有效的消息签名对, 则 A_{II} 攻击成功.

定义 1 若多项式有界的两类敌手 A_{I} 和 A_{II} 分别在 Game1 和 Game2 中攻击成功的概率都可以忽略, 则 CLAS 方案在适应性选择消息攻击下是存在不可伪造的 [21].

2.4 VES 签名方案

本节简要介绍 VES 方案及其安全属性 [19].

2.4.1 VES 方案定义

一个 VES 方案有三个参与者: 签名者, 验证者以及仲裁者. 方案具体包含以下八个算法:

Setup: 输入安全系数 k, 输出系统参数 params.

KeyGen: 输入系统参数 params, 输出签名者公私钥对 (PK, SK).

AdjKeyGen: 输入系统参数 params, 输出仲裁者的公私钥对 (APK, ASK).

Sign: 输入签名者的私钥 SK 和消息 m, 输出签名者的普通签名 σ.

Verify: 输入签名者的普通签名 σ、公钥 PK 和消息 m. 若普通签名验证正确, 则输出 true. 否则, 输出 false.

VESigCreate: 输入公钥 PK、消息 m 以及仲裁者的公钥 APK, 输出对消息 m 的可验证加密签名 ν.

VESigVerify: 输入公钥 PK、消息 m、仲裁者的公钥 APK 以及可验证加密签名 ν. 若签名 ν 是对应公钥 PK 和消息 m 的可验证加密签名, 则输出 true. 否则, 输出 false.

Adjudicate: 输入公钥 PK、对消息 m 的可验证加密签名 ν 以及仲裁者的公私钥对 (APK, ASK), 输出消息 m 在公钥 PK 下的普通签名 σ.

2.4.2 VES 方案的安全属性

VES 方案的正确性需满足如下两点:

(1) 由 VESigCreate 算法生成的可验证加密签名 ν 必须通过 VESigVerify 算法的验证;

(2) 仲裁者通过 Adjudicate 算法可以有效地从可验证加密签名 ν 中提取出普通签名 σ, 且该签名必能通过 Verify 算法的验证.

除了正确性, VES 方案还需满足以下三个安全性质:

(1) 不可伪造性: 攻击者无法伪造他人的可验证加密签名;

(2) 不透明性: 除了仲裁者外, 攻击者不能从可验证加密签名中提取普通签名;

(3) 可提取性: 仲裁者总能从一个有效的可验证加密签名中提取出一个有效的普通签名.

3 CLAVES 方案

本节基于2.3节 CLAS 方案和2.4节 VES 方案, 在3.1节给出 CLAVES 方案的基本定义. 在3.2节基于文献 [21] 的 CLAS 方案, 给出一类具体的 CLAVES 方案, 并在3.3节给出具体方案的安全性证明.

3.1 CLAVES 方案的定义

该方案由以下八个算法组成:

Setup: 安全参数为 k, KGC 选取主密钥 λ, 公开系统参数 params.

PartialPrivateKeyExtract: 输入用户身份信息 ID_i, 系统参数 params 和主密钥 λ, 输出部分私钥 D_i.

UserKeyGen: 输入用户身份 ID_i, 随机选取 x_i, 输出用户的秘密值和公钥对 (x_i, P_i).

AdjKeyGen: 输入系统参数 params, 输出仲裁者的私钥对 (APK, ASK).

VESSign: 输入系统参数 params, 消息 M_i, 仲裁者公钥 APK, 状态信息 Δ, 用户身份信息 ID_i , 私钥 (D_i, x_i) 及其公钥 P_i, 输出可验证加密签名 σ_i.

Aggregate: 聚合者执行该算法. 输入 n 个身份-消息-公钥-签名对 $(\mathrm{ID}_i, M_i, P_i, \sigma_i)$, 输出聚合签名 σ.

AggregateVerify: 输入系统参数 params, 状态信息 Δ, 仲裁者公钥 APK, 聚合签名为 σ, 用户身份信息 ID_i 及其公钥 P_i. 若聚合签名有效, 输出 true, 否则输出 false.

Adjudicate: 输入用户公钥 P_i、对消息 M_i 的可验证加密签名 σ_i 以及仲裁者的私钥 ASK, 输出消息 M_i 在公钥 P_i 下的普通签名 v_i.

定义 2 在 CLAVES 方案中, 如果其在适应性选择消息攻击下是存在不可伪造的且满足 VES 方案的三个安全性质, 则称该 CLAVES 方案是安全的.

3.2 一类 CLAVES 方案的具体构造

本节利用各签署方共享的公钥对普通签名进行加密, 构造出一个具体的 CLAVES 签名方案. 该方案包括以下 7 个算法:

Setup: 设安全参数为 k, KGC 选定一个循环加法群 G_1, P 为其生成元. G_2 是一个循环乘法群且 $|G_1| = |G_2| = q$, 其中 q 为素数. 定义 $e: G_1 \times G_1 \to G_2$ 为一个双线性对, 随机选取 $\lambda \in Z_q^*$ 作为系统主密钥, 计算 $P_T = \lambda P$, 并定义三个密码哈希函数 $H_1: \{0,1\}^* \to G_1$, $H_2: \{0,1\}^* \to G_1$ 以及 $H_3: \{0,1\}^* \to G_1$. 则系统公开参数为 $\mathrm{params} = (G_1, G_2, e, P, P_T, H_1, H_2, H_3)$, 消息空间为 $M = \{0,1\}^*$.

PartialPrivateKeyExtract: 给定用户身份 $\mathrm{ID}_i \in \{0,1\}^*$, KGC 生成用户的部分私钥:

(1) 计算 $Q_i = H_1(\mathrm{ID}_i)$.

(2) 输出部分私钥 $D_i = \lambda Q_i$.

UserKeyGen: 用户 U_i 随机选取 $x_i \in Z_q^*$ 作为另一部分私钥, 计算公钥 $P_i = x_i P$.

PreSignAgree: 在该算法步骤, 用户生成临时公私钥对并共享一个公钥用于可验证加密签名算法. 具体实现如下:

(1) U_i 随机选取 $x_i' \in Z_q^*$ 作为秘密值, 计算临时公钥 $P_i' = x_i' P$.

(2) U_i 随机选取 $k_i \in Z_q^*$, 对临时公钥承诺 $C_i = \mathrm{Commit}(P_i', k_i)$, 并发送给其他用户.

(3) 当所有用户都收到 C_i, U_i 打开承诺, 得到 P_i'.

(4) 直到所有用户都打开其承诺, 计算共享公钥 $P_{\mathrm{pub}} = \sum_{i=1}^{n} P_i'$, 其私钥为 $x_{\mathrm{pub}} = \sum_{i=1}^{n} x_i'$.

VESSign: 给定消息 $M_i \in M$, 私钥 (D_i, x_i), 用户身份 ID_i 及其公钥 P_i, 状态信息 Δ(选取系统参数的一些元素), 生成签名如下:

(1) 随机选取 $r_i, b_i \in Z_q^*$, 计算 $R_i = r_i P$, $B_i = b_i P$.

(2) 计算 $w = H_2(\Delta)$, $S_i = H_3(\Delta || M_i || \mathrm{ID}_i || P_i || R_i || B_i)$.

(3) 计算 $\mathrm{VES}_i = D_i + x_i w + r_i S_i + b_i P_{\mathrm{pub}}$.

(4) 输出消息 M_i 的签名 $\sigma_i' = (R_i, B_i, \mathrm{VES}_i)$.

Aggregate: 给定 n 个签名-消息对 $(M_1, \sigma_1' = (R_1, B_1, \mathrm{VES}_1)), \cdots, (M_n, \sigma_n' = (R_n, B_n, \mathrm{VES}_n))$, 聚合签名器计算 $\mathrm{VES} = \sum_{i=1}^{n} \mathrm{VES}_i$, 则聚合签名为 $\sigma' = (R_1, \cdots, R_n, B_1, \cdots, B_n, \mathrm{VES})$.

AggregateVerify: 输入消息 $M_1, \cdots, M_n$, 共同的状态信息 Δ 以及在这 n 个消息上的聚合签名 σ', 验证者计算如下:

(1) 计算 $w = H_2(\Delta)$,$Q_i = H_1(\mathrm{ID}_i)$ 以及 $S_i = H_3(\Delta || M_i || \mathrm{ID}_i || P_i || R_i || B_i)$, 其中 $1 \le i \le n$.

(2) 验证 $e(\mathrm{VES}, P) \stackrel{?}{=} e(P_T, \sum_{i=1}^{n} Q_i) e(w, \sum_{i=1}^{n} P_i) \prod_{i=1}^{n} e(S_i, R_i) e(P_{\mathrm{pub}}, \sum_{i=1}^{n} B_i)$.

(3) 若 (2) 中的等式成立, 输出 true. 否则, 输出 false.

3.3 安全性证明

以下给出3.2中方案的安全性证明.

(1) 正确性.

首先验证 $e(\mathrm{VES}_i, P) = e(Q_i, P_T)e(w, P_i)e(S_i, R_i)e(B_i, P_{\mathrm{pub}})$.

$$\begin{aligned} e(\mathrm{VES}_i, P) &= e(D_i + x_i w + r_i S_i + b_i P_{\mathrm{pub}}, P) \\ &= e(D_i, P)e(x_i w, P)e(r_i S_i, P)e(b_i P_{\mathrm{pub}}, P) \\ &= e(Q_i, P_T)e(w, P_i)e(S_i, R_i)e(B_i, P_{\mathrm{pub}}) \end{aligned}$$

故可验证签名正确;

然后验证 $e(\mathrm{VES}, P) = e(\sum_{i=1}^n Q_i, P_T)e(w, \sum_{i=1}^n P_i)\prod_{i=1}^n e(S_i, R_i)e(P_{\mathrm{pub}}, \sum_{i=1}^n B_i)$.

$$\begin{aligned} e(\mathrm{VES}, P) &= e(\sum_i D_i + \sum_i x_i w + \sum_i r_i S_i + \sum_i P_{\mathrm{pub}}, P) \\ &= e(\sum_i D_i, P)e(\sum_i x_i w, P)e(\sum_i r_i S_i, P)e(\sum_i P_{\mathrm{pub}}, P) \\ &= e(\sum_{i=1}^n Q_i, P_T)e(w, \sum_{i=1}^n P_i)\prod_{i=1}^n e(S_i, R_i)e(P_{\mathrm{pub}}, \sum_{i=1}^n B_i) \end{aligned}$$

故聚合的可验证加密签名是正确的.

(2) 抗类型 I 攻击的不可伪造性.

与文献 [21] 类似, 只需要在如下阶段进行修改:

♦ setup 阶段: 把共享公钥 P_{pub} 发送给敌手 A_I.

♦ H_3 询问阶段: 增加元素 B_j, 将 $(\Delta_j, M_j, \mathrm{ID}_j, P_j, R_j, B_j, S_j, \gamma_j)$ 添加到 H_3^{list} 中.

♦ Sign 询问阶段:

(1)$c_i = 0$ 时, 随机选取 b_i, 则 $\mathrm{VES}_i = \beta_i P_i + r_i\gamma_i P_T + b_i P_{\mathrm{pub}}$;

(2)$c_i = 1$ 时, 随机选取 b_i, 则 $\mathrm{VES}_i = \alpha_i P_T + \beta_i P_i + \gamma_i R_i + b_i P_{\mathrm{pub}}$.

♦ Forgery 阶段: 将聚合签名修改为 $\sigma'^* = (R_1^*, \cdots, R_n^*, B_1^*, \cdots, B_n^*, \mathrm{VES}^*)$.

(3) 抗类型 II 攻击的不可伪造性.

同样地, 只需在如下阶段进行修改:

♦ setup 阶段: 把共享公钥 P_{pub} 发送给敌手 A_{II}.

♦ H_3 询问阶段: 增加元素 B_j, 将 $(\Delta_j, M_j, \mathrm{ID}_j, P_j, R_j, B_j, S_j, \gamma_j)$ 添加到 H_3^{list} 中.

♦ Sign 询问阶段:

(1)$c_i = 0$ 时, 随机选取 b_i, 则 $\mathrm{VES}_i = V_i + b_i P_{\mathrm{pub}}$;

(2)$c_i = 1$ 时, 随机选取 b_i, 则 $\mathrm{VES}_i = V_i + b_i P_{\mathrm{pub}}$.

♦ Forgery 阶段: 将聚合签名修改为 $\sigma'^* = (R_1^*, \cdots, R_n^*, B_1^*, \cdots, B_n^*, \mathrm{VES}^*)$.

(4) 可验证加密签名的不可伪造性.

定理 1 若 CLAS 方案中的普通签名不可伪造, 则 3.2 节 CLAVES 方案中的 VES 签名不可伪造.

证明: 记 CLAVES 方案和 CLAS 方案的攻击者分别为 $\mathcal{A}$ 和 $\mathcal{B}$, 同时 $\mathcal{B}$ 还作为 CLAVES 方案的挑战者.

假设攻击者 $\mathcal{A}$ 能伪造消息 m^* 的 VES 签名 $\sigma_i'^* = (\mathrm{VES}_i'^*, R_i^*, B_i^*)$, 则挑战者 $\mathcal{B}$ 可以计算出 $v_i^* = \mathrm{VES}_i'^* - x_{\mathrm{pub}} B_i^*$, 从而得到消息 m^* 的普通签名. 这与 CLAS 方案中的不可伪造性矛盾.

故, 定理得证. □

(5) 可验证加密签名的不透明性.

引理 1 (计算 Diffie-Hellman 问题 (CDH)) 给定 $P, aP, bP \in G_1$, 计算 $abP \in G_1$ [28].

定理 2 CLAVES 方案中的 VES 签名是不透明的.

证明: 假设攻击者 $\mathcal{A}$ 在不知道私钥值 x_{pub} 时, 能够从 $\sigma_i' = (R_i, B_i, \text{VES}_i)$ 提取出普通签名 $\sigma_i = (R_i, V_i)$. 这个问题可规约为解决 CDH 困难问题, 即:
已知 P,$B_i = b_iP$,$P_{\text{pub}} = x_iP$, 攻击者 $\mathcal{A}$ 可以计算出 $\text{VES}_i - V_i = b_iP_{\text{pub}} = b_ix_iP$.
这与引理 1 矛盾. 故, 定理 2 得证. □

(6) 可验证签名的可提取性.

本文的仲裁者是 n 个签署方, 当所有签署方都诚实地签署合同, 各签署方均可在区块链上获取私钥 $x_{\text{pub}} = \sum_{i=1}^{n} x_i'$, 从而提取出普通签名 $V_i = \text{VES}_i - x_{\text{pub}}B_i$.

4 多方合同签署协议的构造

在本节, 我们给出一类多方合同签署协议的构造过程. 一般地, 根据各签署方的需求, 协议可采用不同的合同信息. 为了方便介绍, 我们在本节采用相同的合同信息. 本协议包含区块链链下和区块链链上两个阶段. 在区块链链下, 各签署方各自生成 VES 签名, 从而可聚合成一个 CLAVES 签名. 由于该阶段不在区块链上进行, 合同内容和 VES 签名等敏感信息都不会公开, 从而保证协议的隐私性; 在区块链链上, 各签署方公平地交换秘密值, 保证协议公平性. 具体过程如下:

区块链链下: 在本阶段, 各签署方执行 CLAVES 方案, 生成 CLAVES 签名. 若验证通过, 则进入区块链链上阶段. 否则, 协议停止. 具体包括:

(1) 签署方 $\{U_i\}_{i=1}^{n}$ 对合同信息 M 达成一致.

(2) 各签署方执行 CLAVES 方案, 对合同信息 M 生成 CLAVES 签名.

(3) 若 CLAVES 签名通过验证, 则进入区块链链上阶段. 否则, 协议终止.

区块链链上: 在本阶段, 各签署方在区块链链上公平地交换秘密值, 执行以下算法:

PreCondition: 各签署方协商好释放秘密值的顺序, 并指定保证金交易 T_{ji}. 如图 1 所示, 交易 T_{1i} 作为输入用于生成 deposit 交易, 迫使签署方 U_n 生成 claim 交易, 诚实地释放秘密值 x_n 使得 $x_nP = x_n'P$, 否则 U_i 在 τ_n 后, 生成 refund 交易, 拿回保证金. 交易 T_{2i} 同理, 具体区块链脚本内容可参考文献 [6].

(1) 各签署方协商好释放秘密值的顺序, 不妨设为 $U_1 < \cdots < U_n$.

(2) 对于 $i \neq n$, 签署方 U_i 指定交易 T_{1i}, 价值为 dB.

(3) 对于 $i \in \{2, \cdots, n\}$, 签署方 U_i 指定交易 T_{2i}, 价值为 $(i-1)d$B.

(4) 协商交易时间锁 $\tau_1 < \tau_2 < \cdots < \tau_{n-1} < \tau_n$.

Deposit: 各签署方交纳保证金, 执行 COR 协议.

(1) 对于任意 $i \neq n$, 签署方 U_i 将交易 T_{1i} 作为输入用于生成 deposit 交易, 并广播全网, 即

$$U_i \xrightarrow[d,\tau_n]{x_1', x_2', \cdots, x_n'} U_n \tag{1}$$

(2) 按降序, 从 n 到 2 开始: 签署方 U_i 首先确认之前所有的 deposit 交易在区块链上, 然后将交易 T_{2i} 作为输入用于生成 deposit 交易, 并广播全网, 即:

$$U_i \xrightarrow[(i-1)d,\tau_{i-1}]{x_1', x_2', \cdots, x_{i-1}'} U_{i-1} \tag{2}$$

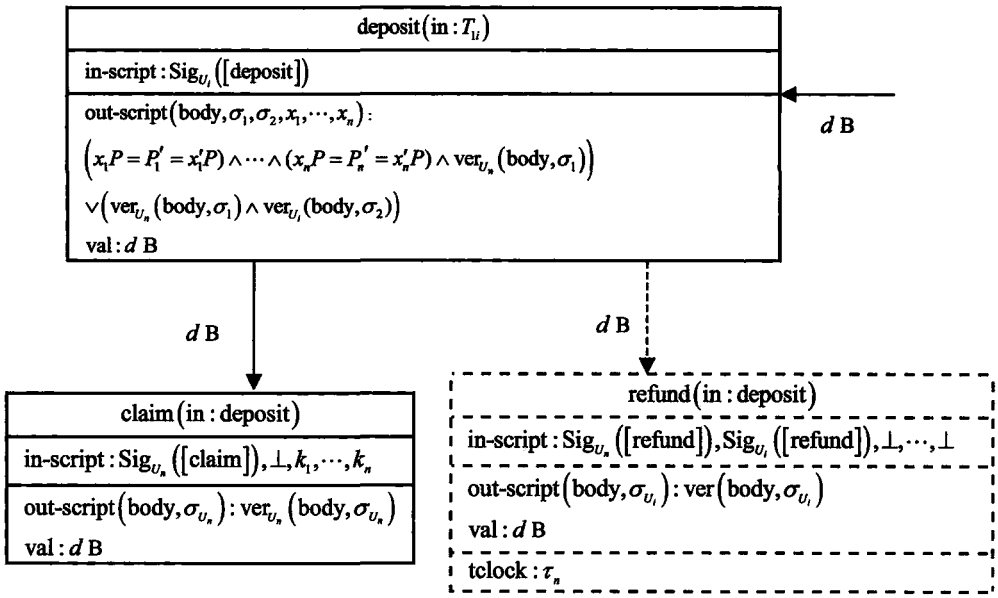

图 1 交易 T_{1i}
Figure 1 Transaction T_{1i}

Claim: 签署方释放秘密值, 拿回保证金.

(1) 对于 $1 \leqslant i < n$, 签署方 U_i 提供签名及秘密值 $x_1', \cdots, x_i'$, 广播 claim 交易, 拿到 U_{i+1} 给的 dB.

(2) 对于 $i \neq n$, 签署方 U_n 提供签名及秘密值 $x_1', \cdots, x_n'$, 广播 claim 交易, 拿到 U_i 给的 dB.

Refund: 若有签署方 U_i 没有广播有效的 claim 交易, 则在其之前的各签署方得到 dB 奖励.

(1) 签署方 U_1 没有广播有效的 claim 交易, 那么签署方均生成 refund 交易拿回保证金, 协议停止.

(2) 对于 $i \neq 1$, 若签署方 U_i 第一个没有广播有效的 claim 交易, 则 $U_{i+1}, \cdots, U_n$ 生成 refund 交易, 拿回保证金. 而 U_i 失去交纳的保证金 $(i-1)d$B, 奖励给 $U_1, \cdots, U_{i-1}$ 每人 dB.

Extract: 各签署方通过计算 $x_{\mathrm{pub}} = \sum_{i=1}^{n} x_i'$, 并提取出普通签名 $V_i = \mathrm{VES}_i - x_{\mathrm{pub}} B_i$.

5 多方合同签署协议的构造的安全性和高效性

本节对第 4 节提出的多方合同签署协议的安全性和高效性进行分析. 在 5.1 节证明协议满足公平性, 在 5.2 节分析协议的高效性.

5.1 安全性

定理 3 第 4 节构造的多方合同签署协议满足公平性.

证明: 考虑各签署方的诚实与否, 分以下两种情况:

(1) 各签署方都诚实.
这种情况意味着签署方都成功地签署合同, 他们分别可以在区块链链下得到各方的 CLAVES 签名、在区块链链上得到各方的秘密值, 最后通过计算, 可提取各方对合同的普通数字签名.

(2) 有签署方不诚实.
假设 U_i 是第一个没有释放秘密值的签署方, 则他将会受到惩罚, 失去其交纳的保证金 $(i-1)d$B. 由于在其之前的签署方都诚实地释放秘密值, 所以每人得到奖励 dB.

根据上面的讨论得知, 签署方要么都能获得各方的签名, 要么需要付出代价获得对方的秘密值, 要么什么都得不到, 所以该协议满足公平性. □

5.2 高效性

在本小节, 我们将本文协议与三方的合同签署协议 [15] 和多方合同签署协议 [14] 进行效率对比. 如表1所示, 其中 p 表示双线性对运算时间, s 表示在群 G_1 上标量乘运算时间, h 表示在 $\{0,1\}^* \to G_1$ 上 hash 运算时间.

表 1 本文协议与已有协议的效率比较

Table 1 Efficiency comparison of the proposed protocol and the existing protocols

协议	签署方	签名验证次数	签名验证开销
文献 [15]	3	6	*
本文	3	1	*
文献 [14]	n	n	$2np+ns$
本文	n	1	$(n+4)p+(2n+1)h$

从表1可以观察得到:

(1) 本文的协议与文献 [15] 相比, 签名验证只需 1 次, 而不需要签署方重复验证. 由于文献 [15] 是基于离散对数的 VES 方案, 与本文所用的基于双线性对的签名方案不同, 因此表 1 中签名验证开销没有进行比较.

(2) 本文的协议与文献 [14] 相比, 签名验证只需一次, 而不需要矿工进行 n 次验证. 在计算开销上, 由于双线性运算的计算开销较大, 因此随着 n 的增大, 本文在验证开销上远低于文献 [14].

6 总结

面向多方合同签署协议的高效性设计, 本文引入聚合签名技术, 提出一类通用的 CLAVES 方案构造方法, 即可根据已有的 CLAS 方案构造出 CLAVES 方案. 由于所选用的 CLAS 方案的签名长度依赖于签署方数量, 所以理论上可以构造出更高效的 CLAVES 方案. 但是基于签名长度固定的 CLAS 方案, 大部分存在安全性问题. 因此, 权衡之下本文选用一种安全的 CLAS 方案进行构造.

结合 CLAVES 方案以及区块链技术, 本文给出一个新的公平合同签署协议. 该协议支持多方进行合同签署, 不仅满足公平性和高效性, 还保障签署方的隐私.

注意到在利用 CLAVES 方案设计公平合同签署协议时, 用户可以采用不同的方案来生成仲裁者的公钥. 本文直接利用所有用户公钥的加法运算生成共享公钥, 因此在实现智能合约时较为简单, 但理论上存在更简单的构造方法, 有待将来进一步研究.

References

[1] BLUM M. How to exchange (secret) keys[J]. ACM Transactions on Computer Systems, 1983, 1(2): 175–193. [DOI: 10.1145/357360.357368]

[2] FRANKLIN M K, REITER M K. Fair exchange with a semi-trusted third party[C]. In: ACM Conference on Computer & Communications Security. New York, ACM, 1999: 1–5. [DOI: 10.1145/266420.266424]

[3] ZHOU J Y, GOLLMANN D. An efficient non-repudiation protocol[C]. In: Proceedings of 10th Computer Security Foundations Workshop, IEEE, 1997: 126–132. [DOI: 10.1109/CSFW.1997.596801]

[4] ASOKAN N, SCHUNTER M, WAIDNER M. Optimistic protocols for fair exchange[C]. In: Proceedings of ACM Conference on Computer and Communications Security. New York, ACM, 1997: 7–17. [DOI: 10.1145/266420.266426]

[5] NAKAMOTO S. Bitcoin: A peer-to-peer electronic cash system[EB/OL]. https://bitcoin.org/bitcoin.pdf. 2017.

[6] ANDRYCHOWICZ M, DZIEMBOWSKI S, MALINOWSKI D, et al. Fair two-party computations via Bitcoin deposits[C]. In: Financial Cryptography and Data Security. Springer Berlin Heidelberg, 2014: 105–121. [DOI: 10.1007/978-3-662-44774-1_8]

[7] KUMARESAN R, BENTOV I. How to use Bitcoin to incentivize correct computations[C]. In: Proceedings of ACM SIGSAC Conference on Computer and Communications Security. New York, ACM, 2014: 30–41. [DOI: 10.1145/2660267.2660380]

[8] BENTOV I, KUMARESAN R. How to use Bitcoin to design fair protocols[C]. In: Advances in Cryptology—CRYPTO 2014. Springer Berlin Heidelberg, 2014: 421–439. [DOI:10.1007/978-3-662-44381-1_24]

[9] LIU J, LI W T, KARAME G O, et al. Towards fairness of cryptocurrency payments[DB/OL]. https://arxiv.org/pdf/1609.07256v3.pdf. 2017-01-28.

[10] ZHAO Z C, CHAN T H H. How to vote privately using Bitcoin[C]. In: Information and Communications Security—ICICS 2015. Springer Cham, 2015: 82–96. [DOI: 10.1007/978-3-319-29814-6_8]

[11] WAN Z G, DENG R H, LEE D. Electronic contract signing without using trusted third party[C]. In: Network and System Security—NSS 2015. Springer Cham, 2015: 386–394. [DOI: 10.1007/978-3-319-25645-0_27]

[12] TIAN H B, HE J J, FU L Q. A privacy preserving fair contract signing protocol based on public block chains[J]. Journal of Cryptologic Research, 2017, 4(2): 187–198. [DOI: 10.13868/j.cnki.jcr.000173]
田海博, 何杰杰, 付利青. 基于公开区块链的隐私保护公平合同签署协议 [J]. 密码学报, 2017, 4(2): 187–198. [DOI: 10.13868/j.cnki.jcr.000173]

[13] TIAN H B, HE J J, FU L Q. Contract coin: Toward practical contract signing on blockchain[C]. In: Information Security Practice and Experience—ISPEC 2017. Springer Cham, 2017: 43–61. [DOI: 10.1007/978-3-319-72359-4_3]

[14] WU J X, GAO Y, ZHANG Z Y, et al. A multi-party privacy preserving fair contract signing protocol based on blockchains[J]. Journal of Cyber Security, 2018, 3(3): 8–16. [DOI: 10.19363/j.cnki.cn10-1380/tn.2018.05.02]
吴进喜, 高莹, 张宗洋, 等. 基于区块链的多方隐私保护公平合同签署协议 [J]. 信息安全学报, 2018, 3(3): 8–16. [DOI: 10.19363/j.cnki.cn10-1380/tn.2018.05.02]

[15] HUANG H, LI K C, CHEN X F. A fair three-party contract singing protocol based on blockchain[C]. In: Cyberspace Safety and Security—CSS 2017. Springer Cham, 2017: 72–85. [DOI: 10.1007/978-3-319-69471-9_6]

[16] BONEH D, GENTRY C, LYNN B, et al. Aggregate and verifiably encrypted signatures from bilinear maps[C]. In: Advance in Cryptology—EUROCRYPT 2003. Springer Berlin Heidelberg, 2003: 416–432. [DOI: 10.1007/3-540-39200-9_26]

[17] SHAO Z H, GAO Y P. Practical verifiably encrypted signatures based on discrete logarithms[J]. Security & Communication Networks, 2016, 9(18): 5996–6003. [DOI: 10.1002/sec.1751]

[18] Zhang Y H, Hu Y P. A new verifiably encrypted signature scheme from Lattices[J]. Journal of Computer Research and Development, 2017, 54(2): 305–312. [DOI: 10.7544/issn1000-1239.2017.20150887]
张彦华, 胡予濮. 新的基于格的可验证加密签名方案 [J]. 计算机研究与发展, 2017, 54(2): 305–312. [DOI: 10.7544/issn1000-1239.2017.20150887]

[19] GONG Z, LONG Y, HONG X, et al. Two certificateless aggregate signatures from bilinear maps[J]. Journal of Information Science & Engineering, 2008, 26(6): 2093–2106. [DOI: 10.1109/SNPD.2007.132]

[20] WEN Y L, MA J F. An aggregate signature scheme with constant pairing operations[C]. In: Proceedings of International Conference on Computer Science and Software Engineering. IEEE, 2008: 830–833. [DOI: 10.1109/CSSE.2008.941]

[21] ZHANG L, ZHANG F T. A new certificateless aggregate signature scheme[J]. Computer Communications, 2009, 32(6): 1079–1085. [DOI: 10.1016/j.comcom.2008.12.042]

[22] XIONG H, WU Q H, CHEN Z. Strong security enabled certificateless aggregate signatures applicable to mobile computation[C]. In: Proceedings of Third International Conference on Intelligent Networking and Collaborative Systems. IEEE, 2012: 92–99. [DOI: 10.1109/INCoS.2011.151]

[23] KHAN M K, HE D. Cryptanalysis of a certificateless aggregate signature scheme for mobile computation[J]. Applied Mathematics & Information Sciences, 2013, 7(4): 1383–1386. [DOI: 10.12785/amis/070416]

[24] MING Y, ZHAO X M, WANG Y M. Certificateless aggregate signature scheme[J]. Journal of University of Electronic Science & Technology of China, 2014, 43(2): 188–193. [DOI: 10.3969/j.issn.1001-0548.2014.02.005]
明洋, 赵祥模, 王育民. 无证书聚合签名方案 [J]. 电子科技大学学报, 2014, 43(2): 188–193. [DOI: 10.3969/j.issn.1001-0548.2014.02.005]

[25] ZHANG Y L, LI C Y, WANG C F, et al. Security analysis and improvements of certificateless aggregate signature schemes[J]. Journal of Electronics & Information Technology, 2015, 37(8): 1994–1999. [DOI: 10.11999/JEIT141635]

张玉磊, 李臣意, 王彩芬等. 无证书聚合签名方案的安全性分析和改进 [J]. 电子与信息学报, 2015, 37(8): 1994–1999. [DOI: 10.11999/JEIT141635]

[26] DU H Z, WEN Q Y. Attack and improvement of a certificateless aggregate signature scheme[J]. Acta Scientiarum Naturalium Universitatis Sunyatseni, 2017, 56(1): 77–84. [DOI: 10.13471/j.cnki.acta.snus.2017.01.013]
杜红珍, 温巧燕. 无证书聚合签名方案的攻击与改进 [J]. 中山大学学报 (自然科学版), 2017, 56(1): 77–84. [DOI: 10.13471/j.cnki.acta.snus.2017.01.013]

[27] ZHANG F G, SAFAVI-NAINI R, SUSILO W. Efficient verifiably encrypted signature and partially blind signature from bilinear pairings[C]. In: Progress in Cryptology—INDOCRYPT 2003. Springer Berlin Heidelberg, 2003: 191–204. [DOI: 10.1007/978-3-540-24582-7_14]

[28] BONEH D, FRANKLIN M. Identity-based encryption from the Weil pairing[J]. SIAM Journal on Computing, 2001, 32(3): 213–229. [DOI: 10.1137/S0097539701398521]

作者信息

高莹 (1977–), 湖北大悟人, 副教授, 硕士生导师. 主要研究领域为密码学、区块链等.
gaoying@buaa.edu.cn

吴进喜 (1994–), 福建莆田人, 硕士. 主要研究领域为密码学、区块链等.
mathwjx@163.com