·区块链技术专题·　　文章编号：1000-3428(2019)11-0016-08　　文献标志码：A　　中图分类号：TP391

基于智能合约的无线 Mesh 网络安全架构

张　彬[1]，广　晖[2]，陈　熹[1]

（1. 数学工程与先进计算国家重点实验室，郑州 450001；2. 郑州大学 体育学院，郑州 450044）

摘　要： 针对无线 Mesh 网络安全性，结合区块链技术，提出基于智能合约的安全架构（SCSA）。在 Mesh 路由器节点上建立区块链，部署 3 类智能合约对节点公钥进行分级管理，通过这 3 类智能合约实现节点公钥更新与撤销以及用户访问认证，同时采用椭圆曲线 Diffie-Hellman 和对称加密实现节点间的安全通信。实验结果表明，SCSA 可有效抵抗克隆或伪造节点攻击、密钥泄露攻击及拒绝服务攻击，同时保证了网络通信的安全性和高效性。

关键词： 无线 Mesh 网络；区块链；智能合约；密钥管理；认证；椭圆曲线 Diffie-Hellman

开放科学（资源服务）标志码（OSID）：

中文引用格式： 张彬，广晖，陈熹. 基于智能合约的无线 Mesh 网络安全架构[J]. 计算机工程，2019，45(11)：16-23，31.

英文引用格式： ZHANG Bin，GUANG Hui，CHEN Xi. Smart contract-based security architecture for wireless Mesh network[J]. Computer Engineering，2019，45(11)：16-23，31.

Smart Contract-based Security Architecture for Wireless Mesh Network

ZHANG Bin[1]，GUANG Hui[2]，CHEN Xi[1]

（1. State Key Laboratory of Mathematical Engineering and Advanced Computing，Zhengzhou 450001，China；
2. Physical Education College，Zhengzhou University，Zhengzhou 450044，China）

【Abstract】 To improve the security of Wireless Mesh Network（WMN），this paper proposes a Smart Contract-based Security Architecture（SCSA）that integrates blockchain technology. The proposed architecture builds blockchains on routing nodes. Three types of smart contracts are deployed to provide hierarchical management for public keys of nodes，including public key updating，public key cancellation and user access authentication. Also，the Elliptic Curve Diffie-Hellman（ECDH）and symmetric encryption are used to implement secure communication between nodes. Experimental results show that SCSA can efficiently resist node clone attacks or node forgery attacks，key compromise attacks and denial of service attacks，ensuring the security and efficiency of network communication.

【Key words】 Wireless Mesh Network（WMN）；blockchain；smart contract；key management；authentication；Elliptic Curve Diffie-Hellman（ECDH）

DOI： 10.19678/j.issn.1000-3428.0054940

0　概述

无线 Mesh 网络（Wireless Mesh Network，WMN）为在较大区域内的无线宽带网络连接提供了解决方案[1-2]。WMN 是由 Mesh 路由器和 Mesh 客户端组成的动态自组织、自配置的多跳无线网络。Mesh 路由器是静止的或具有较小的移动性，可以使用任何无线技术（如 Wi-Fi、Wi-Max 等）互连构成骨干网，并作为路由设备或接入点。Mesh 客户端通常是笔记本电脑、手机等便携设备，可以自由移动，也可以处于静止状态。在 Mesh 网络中，Mesh 客户端还可以协作转发其他节点传输的信息，扩大网络覆盖范围。整个 Mesh 网络通过作为网关的 Mesh 路由器经有线链路连接到互联网，WMN 具有自组织、自配置、自我修复等特性，这意味着网络以即插即用的方式运行，可以为用户提供低成本的高速网络服务。WMN 部署方便，且具有移动性、灵活性、高可靠性、高覆盖率等特点。因此，WMN 得到越来越广泛的应用，特别是需要进行网络通信的野外复杂环境。WMN 的应用领域包括远程医疗、安全监控系统、城市交通以及社区和企业网络等[3]。WMN 被业界认为是未来无线互联网的重要组网方式之一，但是其

基金项目： 国家重点研发计划项目“网络空间拟态防御技术机制研究”（2016YFB0800101）。

作者简介： 张　彬（1994—），男，硕士研究生，主研方向为区块链安全技术；广　晖、陈　熹，讲师。

收稿日期： 2019-05-17　　**修回日期：** 2019-07-05　**E-mail：** 1453265428@qq.com

计算机工程 www.ecice06.com

也面临较大的安全风险,主要包括无线信道容易遭受窃听、篡改等攻击;移动设备易丢失或落入敌手,恶意攻击不仅来自网络外部,也可能来自内部被控节点;Mesh 路由器通常部署在室外,而且由于成本控制的因素,因此通常得不到较好的物理保护。

WMN 中主要需具备 3 类安全功能:用户终端访问安全性,即用户验证;骨干设备安全性,即骨干设备间的相互验证;数据加密,即安全地交换数据和控制消息[4]。近年来的研究发现大多数基于身份密码或证书的方案都需要中心化的认证服务器或者 CA,易受到单点故障和拒绝服务攻击。本文将区块链技术和智能合约应用于 WMN,使用以太坊智能合约对 Mesh 路由器和 Mesh 客户端的公钥进行管理实现终端访问控制和安全通信,进而提出基于智能合约的安全架构(Smart Contract-based Security Architecture, SCSA)。

1　相关理论与研究

1.1　ECDH 算法

椭圆曲线 Diffie-Hellman(Elliptic Curve Diffie-Hellman,ECDH)算法使用椭圆曲线密码的 Diffie-Hellman 协议的变体。在该算法中,参与方各自具有一个椭圆曲线公私钥对,能够通过不安全的信道协商出一个共享会话密钥。该共享会话密钥通常作为对称加密密钥对之后的通信过程进行加密。

通过以下示例对密钥创建过程进行说明。假设 Alice 想要与 Bob 建立共享密钥。首先,他们要确立公共参数(p,a,b,G,n),其中,p 为大素数,a、b 为椭圆曲线方程参数,G 为椭圆曲线在 F_p 下加法本原元,也称为基点,n 为 G 的阶。另外,每个参与方都必须有一个椭圆曲线公私钥对,由私钥 d(在区间$[1,n-1]$中随机选择的整数)和表示一个点的公钥 Q($Q = d \cdot G$)组成。假设 Alice 的公私钥对为(d_A,Q_A),Bob 的公私钥对为(d_B,Q_B)。在执行计算前,每个参与方都必须知道对方的公钥。

Alice 计算点$(x_A,y_A) = d_A \cdot Q_B$,Bob 计算点$(x_B,y_B) = d_B \cdot Q_A$,那么共享秘密为 x_A(相应点的 x 坐标)。大多数基于 ECDH 的标准化协议使用一些基于散列的密钥导出函数,从而由 x_A 得到一个对称密钥。双方计算的共享密钥是相等的,因为 $d_A \cdot Q_B = d_A \cdot d_B \cdot G = d_B \cdot d_A \cdot G = d_B \cdot Q_A$。Alice 最初公开的有关自己的唯一信息是公钥,因此除 Alice 之外的任何一方都不能确定 Alice 的私钥,除非该方能解决椭圆曲线离散对数问题。同样,Bob 的私钥也是安全的。除 Alice 和 Bob 以外的任何一方都不能计算出该共享密钥,除非该方能解决椭圆曲线 Diffie-Hellman 问题。

1.2　区块链技术

一般认为区块链是一种按照时间顺序将数据区块以链条的方式组合形成的特定数据结构,并且是一种以密码学方式保证其不可篡改和不可伪造的去中心化的分布式共享总账系统[5]。从数据的角度来看,区块链是一种不可能被更改的分布式数据库。从技术的角度来看,区块链并不是一项单一的技术创新,而是 P2P 网络技术[6]、非对称加密技术[7]、共识机制[8]、链上脚本[9]等多种技术深度整合后实现的分布式账本技术[10]。区块链技术具有以下特性:

1)分布式控制:区块链是一种分权去中心化的方案,整个系统没有中央权威机构,所有节点之间的地位都是平等的,系统内所有节点通过共识协议进行点对点的安全通信。

2)数据透明和可追溯性:系统中执行的每个事务(交易)的完整副本都存储在区块链中,并且对所有节点公开。

3)分布式共识:事务由网络的所有节点而不是中央实体维护验证,并且这些节点都是平等的。

4)安全性:系统内每个节点都有一份完整的数据备份,并且利用哈希算法对区块中的数据进行完整性保护,以链式连接区块。改变区块或区块内数据非常困难,比特币中只有超过全网系统 51% 的算力才可能修改数据。因此,区块链是防篡改且不易伪造的,不会被恶意行为者操纵。

区块链的链式结构(如图 1 所示)包括交易、区块、链 3 个基本概念。由于使用区块链的比特币是一种电子货币,因此通过交易来描述区块链上的基本操作,交易是对账本添加转账记录的操作。区块用于记录一段时间内的全部交易操作以及产生的结果,区块按照时间顺序依次产生并链接到前一区块后,这样就构成一条单链,即区块链。在区块链上,后一区块会记录前一区块的哈希值,从而确保区块的合法性以及一致性。

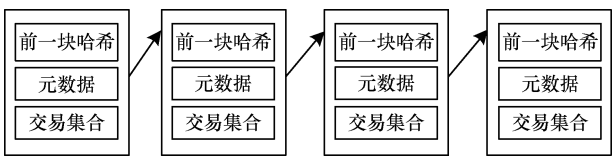


图 1　区块链的链式结构

1.3　以太坊和智能合约

比特币开发商 Vitalik Buterin 在 2013 年设计以太坊(Ethereum)[11],他希望建立一个平台来促进区块链上分布式应用程序的开发。因此,以太坊是一个全新的开放的区块链平台,允许任何人在平台中建立和使用通过区块链运行的去中心化应用,开启了区块链 2.0 时代,其也是区块链 2.0 时代的典型代表。以太坊拥有自己的加密货币 ether 和用于支

计算机工程 www.ecice06.com

付计算和交易费的内部货币 gas。智能合约是区块链上以计算机程序的方式缔结和运行的各种合约，用以开发分布式应用。在以太坊中智能合约可以使用称为 Solidity 的内置图灵完备语言进行编程开发。图灵完备语言是指假设在有足够的时间和空间情况下，可以解决任何计算问题的编程语言。图灵完备性也是以太坊区别于比特币成为区块链 2.0 开端的主要原因之一。以太坊使用工作量证明（Proof of Work，PoW）作为其共识机制，但很快转向权益证明（Proof of Stake，PoS）共识机制。以太坊目前使用的工作证明算法的基本构建是一种称为 Ethash 的记忆硬件哈希算法。创建块的时间约为12 s，明显低于其他区块链系统的块创建时间。但是，较低的块创建时间会导致较大的“孤儿块”比率，因此系统使用 GHOST 协议[12]将“最重”的链视为主要区块链。在这种情况下，“最重”的链也包括“孤儿块”。

1.4 相关研究

国内外学者对于 WMN 安全性已有较多研究成果，提出多种解决方案，但这些方案都有其优势与不足，并且适用于不同的网络模型或应用环境。目前，WMN 的安全性问题并没有得到完全解决，还需要进一步研究。

文献[13-14]使用椭圆曲线密码体制（Elliptic Curve Cryptography，ECC）提出节点之间相互认证的方法。文献[15]利用密码学中基于身份的盲签名思想，提出一种具有隐私保护功能的高效无线 Mesh 网络匿名切换认证方案。该方案使用盲签名技术，客户端利用假名进行切换认证，实现对用户的真实身份信息与运动轨迹的隐藏，并对客户端的隐私数据进行有效保护。文献[16]提出一种高效的切换认证机制。该机制提供了更高级别的安全性，从而更好地保护客户隐私，包括信任模型、不同类型的票证和椭圆曲线密码体制，计算成本较低，适用于运算能力有限的移动设备。文献[17]设计一种基于身份的代理群签名方案。该方案在 WMN 分层代理体系结构上应用代理群签名，结合基于身份的群签名方式实现匿名认证，具有较高的计算效率。

但上述研究都假设存在可信服务器。文献[18]提出结合 EC-IBC 与 ThSS 的分布式认证协议。该协议提供了密钥的协作生成方式，而不需要任何可信的权威，可以为 WMN 提供灵活、完全分布式的密钥管理，但其密钥建立的延时较高，仅适应于具有较少节点的小型 Mesh 网络，因为当节点数目较大时密钥建立可能会失败。

文献[19-20]提出基于区块链技术的分布式 PKI 系统 Certcoin，其核心思想是通过公共总账来记录用户证书，以公开的方式将用户身份与证书公钥相关联，从而实现去中心化的 PKI 建设，任何用户都可查询证书签发过程，解决传统 PKI 系统所面临的证书透明度及 CA 单点故障问题。针对证书颁发机构发布攻击性的证书且证书颁发机构对其颁发的证书不透明的情况，在以太坊区块链上使用 Web 信任模型和智能合约，文献[21]设计基于智能合约的 PKI 机制 SCPKI。文献[22]设计在区块链系统下的多方密钥协商协议。该协议去除了可信中心，从而实现去中心化的密钥协商。文献[23-24]提出基于区块链的公钥分发系统，采用改进的 PBFT 共识，解决公钥分发的中间人攻击问题。但这些方案均未解决 WMN 的认证问题。

2 基于智能合约的 WMN 安全架构

WMN 由 Mesh 路由节点和 Mesh 客户端节点组成（如图 2 所示），所有的路由节点组成区块链网络。WMN 分成若干区域，这在实际应用中相当普遍，比如一个工业园区由多个企业或者一个企业由多个部门共同建立 WMN 网络，划分区域的目的是便于分级管理。在每个区域，选择较可靠（固定且不易被攻击）的路由节点作为管理节点。在每个区域有一个管理节点（可以有多个管理节点），在该管理节点上部署一个名为管理合约的智能合约，用于管理所在区域路由节点的公钥，并实现公钥更新和公钥撤销功能。在每个普通路由节点上部署智能合约，用于管理客户端合约。所有节点共同建立一个联盟合约，用于记录所有管理节点地址及其上的智能合约地址，便于拓展 WMN。

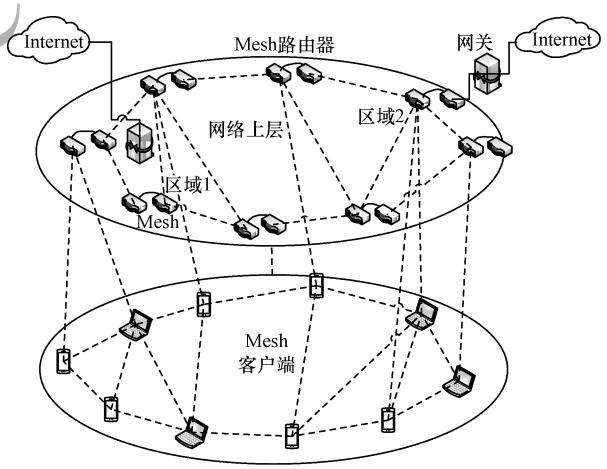


图 2 无线 Mesh 网络结构

2.1 网络模型

在 WMN 网络模型中，Mesh 路由器具有一定的计算能力、存储能力以及足够的电源，可以作为区块链网络的节点。网关具有足够的计算能力、存储能

力,以充当区块链的记账节点(共识“挖矿”节点),并且做出以下假设:

1)每个路由器和客户端节点都有一个 *ID* 标识。

2)Mesh 路由器具有检测机制[25]。

2.2　初始化

在无线 Mesh 网络建立前,离线私钥生成器(Private Key Generator,PKG)确定整个无线 Mesh 网络系统 ECC 基本公共参数,为系统所有 Mesh 路由器 R_j(客户端 U_k)生成唯一标识 ID_j(ID_k),并为区域生成标识 *Area*。未来新加入用户也要向离线 PKG 申请唯一标识 *ID*。本文方案中的离线 PKG 仅用于初始化过程确定 ECC 基本公共参数,并不保存任何密钥信息。每个路由器和客户端 U_k 分别生成自己的区块链账户 $Addr_j$ 和 $Addr_k$,公私钥对(PK_j,SK_j)和(PK_k,SK_k)可能有不同用途的多对公私钥对,本文为简化,只有一对公私钥对。每个区域选取本区域一个路由节点 R_l 作为自己的管理节点 *M*,管理账户 $Addr_M = Addr_l$ 对应的公私钥对为(PK_M,SK_M),公钥更新时间间隔为 *T*(根据安全情况和节点数量设置)。

2.3　管理注册

假设在每个区域内有一个管理节点,管理节点本身也是路由节点之一,管理节点账户的私钥必须得到严格保护,因为管理节点账户私钥泄露会使整个区域的网络安全性得不到保障。如表 1 所示,管理节点账户通过 *manager_register*( )函数将管理节点标识 *managerID*、管理节点公钥 *managerPK*、签名 *sig_manager*、管理节点地址 *managerAddr*、公钥更新间隔 *T_renew* 写入管理合约,并将管理节点所在区域路由节点数量 *node_num* 置为 0,以此完成管理节点的注册和管理合约的部署。其他节点可通过调用管理合约 *M* 以获得管理节点信息,验证管理节点签名 *sig_manager* 以确认管理合约的真实性。普通 Mesh 路由器使用上述类似的过程部署客户端合约,以及执行后续的添加客户端公钥等操作。

表 1　管理合约初始化相关数据结构和函数

数据项/函数	功能
struct manager	管理节点结构体
managerID	管理节点 *ID*
managerPK	管理节点公钥
sig_manager	管理节点账户私钥对 *managerID*、*managerPK* 签名,用于验证管理节点身份
T_renew	管理节点所在区域公钥更新时间间隔
managerAddr	管理节点账户地址
manager public M	管理节点 *M*,节点 *M* 公开
node_num	管理节点所在区域节点数量
manager_register( )	管理节点注册函数及合约构造函数

2.4　节点注册与访问认证

2.4.1　Mesh 路由节点注册

当新的 Mesh 路由节点加入网络时,可通过如下方式完成注册,其中使用的相关数据结构和函数见表 2:

1)Mesh 路由节点将信息提交给管理节点,管理节点首先审查提交的信息,确定该节点未注册过(*register_ack* 为 0),然后调用管理合约 *M_Add*( )函数,将 *routerID* 和 *routerPk* 写入合约,设定过期时间 $outtime = now + T_renew$,将 *register_ack* 和 *manager_ack* 置 1(表示节点注册且管理节点确认其加入网络),*ack* 标记为已注册,*route_num* 增加 1,*userID*[*routerID*]记录 *router* 的地址,通过合约将节点直接添加到网络,并将节点加入的信息公告全网(区块链网)。

2)Mesh 路由节点 *A* 调用管理合约 *Add*( )添加信息,但此时节点 *A* 还没加入到网络中,其属性 *ack* 标记为正在注册,*route_num* 和 *manager_ack* 不变。节点 *A* 注册消息公告全网,其他节点(包括管理节点)收到新节点注册的信息,核查其身份,管理节点调用管理合约的 *Rec*( ),*manager_ack* 置 1;普通节点调用管理合约 *Rec*( )向节点 *A* 的 *REC* 列表添加自己的地址,确认节点数目 *rec_num* 增加 1,多数普通节点调用管理合约 *Rec*( )使节点 *A* 确认节点数目 *rec_num* 超过总节点数 *route_num* 的 2/3,从而节点 *A* 成功加入网络,并且广播节点 *A* 的加入信息。*trustlevel* 是一个节点在网络中的可信程度(满值为 100%),$trustlevel = \frac{rec_num}{route_num} \times 100\%$ 表示节点在网络中被认可的程度,在通信过程中选择 *trustlevel* 高的节点作为中继节点转发消息。

表 2　管理合约路由节点注册相关数据结构和函数

数据项/函数	功能
ACK{*registering*,*registed*,*malcious*,*expired*}	用于说明节点公钥属性(正在注册,已注册,可疑,过期)
struct router	路由节点结构
routerID	路由 *ID* 编号
routerPk	路由公钥
manager_ack	管理节点确认,值为 1 时说明管理节点同意节点加入网络
node_contract_addr	Mesh 路由节点客户端管理合约地址
register_ack	确认节点注册
REC	确认节点地址列表
rec_num	确认节点数量
ack	节点公钥属性
trustlevel	可信级别
outtime	公钥过期时间
userID[*ID*]	路由 *ID* 对应的地址
M_Add( )	管理节点添加路由节点公钥函数
Add( )	路由节点申请注册函数
Rec( )	确认路由节点注册申请函数
ViewPK( )	查看公钥及其属性函数

2.4.2　客户端节点注册和访问认证

客户端 *C* 连上附近的路由 *R* 时,告知路由身份 *ID*,路由 *R* 调用管理节点智能合约 *ViewPK*( )检查

计算机工程 www.ecice06.com

该节点是否注册,若已注册,则将自己的公钥和时间戳用客户端公钥加密后传给客户端,正确的客户端获得路由公钥,认证通过加入网络;否则,路由发消息询问客户端身份 *ID* 的证明,验证 *ID* 并取得客户端公钥,调用客户端合约 *R_Add*( )(见表 3)使 *clientPk* 写入自己的合约,将注册时间 *Time* 记为网络时间,*R_Add*( )函数调用管理合约添加 *userID*[*ID*]为路由合约地址,以便通过管理合约查看客户端公钥。

表 3 客户端合约添加客户端节点相关数据结构和函数

数据项/函数	功能
struct client	客户端结构
clientPk	客户端公钥
time	客户端注册时间
R_Add( )	路由节点添加客户端用户函数

2.5 网络安全通信

假设网络中的两节点 U_i、U_j 相互通信,U_i 为客户端时,通过注册和访问认证后加入网络获得所属路由 R_i 公钥,通过 ECDH 算法获得会话密钥,与路由加密通信获得目的节点 U_j 公钥;U_i 路由节点直接调用管理节点智能合约 *ViewPK*( )函数获得网络节点 U_j 公钥,通过 ECDH 算法获得会话密钥 $K_{i,j}$。U_i 与 U_j 间的通信数据包为:$E_{K_{k_1,k_2}}(E_{K_{i,j}}(msg)\parallel U_{k_2\text{-next}}\parallel U_j\parallel U_i)$,其中,$E$ 为对称加密算法,K_{k_1,k_2} 为两中继节点的会话密钥,$U_{k_2\text{-next}}$ 为 U_{k_2} 的下一跳节点,*msg* 为实际消息数据。当中继节点 U_k 为客户端时,客户端会在向下一跳发包前向所连路由 R_l 发送带有下列数据包的消息询问下一跳公钥:$E_{K_{k,l}}(E_{K_{i,j}}(msg)\parallel U_{k_2\text{-next}}\parallel U_j)$。

2.6 公钥更新

公钥更新步骤具体如下:

1)路由节点更新公钥:每个路由都有一个公钥计时器,当自己的公钥快过期时(公钥更新至少需要一个区块生成时间,建议留出 3 倍的区块生成时间),在路由空闲时调用管理合约 *renew*( )函数更新公钥,并将 *outtime* 更新,*ack* 重新设为已注册状态(路由节点只有当公钥为已注册或过期状态时才更新公钥),并可以通过调用 *Time*( )函数实时查看任意节点是否过期。

2)客户端更新公钥:当客户端 *C* 连接到网络后(客户端已注册),所连接的路由 *R* 都会调用 *C* 所属路由 *R*′的客户端合约 *Time*( )函数检查客户端 *C* 的公钥注册时间,如果更新间隔 *time-Now*(*Now* 为网络当前时间)接近 *T_renew*(由于公钥更新至少需要一个区块生成时间,因此更新间隔应小于 *T_renew*),通知客户端更新公钥以保证安全,然后客户端 *C* 生成新的公私钥对,将新公钥 PK_{new} 及数据包 $newPK=E_{K_{C,R'}}(PK_{\text{new}}\parallel ID_C)$ 加密传输给路由 *R*,路由 *R* 会将公钥保存在本地。*R* 将 *newPK* 包传输给路由 *R*′通知其更新 *C* 公钥,由于 $K_{C,R'}$(由 *C* 的旧公钥生成)只有 *C*、*R*′双方知道,因此该公私钥更新由 *C* 发起,*R*′调用客户端合约 *renew*( )函数更新 *C* 公钥。等待 *R* 查询到 ID_C 已更新的公钥时,通知 *C* 更新公钥成功。需要注意的是客户端更新公钥应选在客户端刚接入网络时,验证和更新同时进行,避免影响客户端通信。

2.7 公钥撤销

每个路由节点都有检测机制,检测周围节点是否存在可疑行为。如表 4 所示,如果节点 *A* 监测到一个相邻节点 *B* 的可疑行为,则通过管理合约 *accuse*( )函数对 *B* 节点进行指控。当一个节点 *B* 被管理节点指控时,*router*[$Addr_A$]直接被删除,*node_num* 减小 1,节点公钥被撤销(网络中其他节点不能查到该节点公钥,该节点被孤立,相当于从网络中删除该节点);而被普通路由 *A* 指控时,*B* 被加入可疑节点 *malcious_node* 列表,其 *Acc* 增加 *A* 的地址,指控节点数量 *Acc_num* 增加 1。当 *B* 受到路由节点指控数增多,*Acc_num* 超过 *route_num*/3 时,认为该节点确实可疑,节点 *ack* 值被标记为可疑,其他与其通信的节点通过 *ViewPK*( )函数查看其公钥时就会注意到节点可疑情况;当一个节点的指控节点数量超过 2/3 时,确认该节点不可信,撤销其公钥,并将可疑列表的 *Rev* 置为 1。

表 4 管理合约公钥撤销相关数据结构和函数

数据项/函数	功能
struct malcious_node	可疑节点结构体
ID	可疑节点 *ID*
Acc	指控者列表
Acc_num	指控者数量
Rev	撤销确认
Accuse( )	指控可疑节点函数

2.8 多区域联盟合约

初始的多个多区域管理节点共同建立一个联盟合约,如表 5 所示,类似于管理合约,只是该合约去掉 *struct manager* 结构,因为所有管理节点权力等同,合约初始化时将初始所有管理节点的路由信息 *manager_router*[$Addr_i$](*Addr*$_i$ 为各管理节点的地址)写入合约。当两个区域节点通信时,先确定目的节点所在区域的管理节点,查看联盟合约确定目的节点的管理合约地址,调用该合约 *ViewPK*( )得到目的节点公钥,然后进行安全通信。

计算机工程 www.ecice06.com

表 5　联盟合约相关数据结构和函数

数据项/函数	功能
struct manager_router	管理节点结构
ID	管理路由节点 *ID* 编号
Area	管理节点区域标记
manage_contract_addr	管理所属节点下的合约地址
REC	确认该节点的节点地址列表
rec_num	确认节点数量
ack	节点属性
managerID[*ID*]	路由 *ID* 对应的地址
Add_manager( )	添加管理节点函数
Rec_manager( )	管理节点确认函数

联盟合约增强了无线 Mesh 网络拓展性,可以在原网络基础上加入新的区域拓展网络。新加入区域按系统 *ECC* 公共参数初始化本区域节点,区域管理节点构建自己的管理合约,调用联盟合约 *Add_manager*( )函数(类似于管理合约 *Add*( )函数)以注册到联盟合约,之后等待其他管理节点审查调用类似于 *Rec*( )的方法完成确认。在至少 2/3 管理节点确认后,该管理节点加入联盟合约,新区域加入到 WMN 网络中。

3　安全性分析

本文方案建立在以太坊私有链上,其安全性分析具体如下:

1)克隆或伪造节点攻击。本文方案建立在以太坊私有链上,在一定程度上防止了外部设备加入。若外部攻击者获知私有链信息,创建账号,并实施克隆或伪造节点攻击时,由于所有注册的节点都可在区块链上查到,无法重复使用相同的 *ID*,而伪造节点注册无法得到管理节点和其他节点的验证,因此无法成功注册到网络。

2)内部节点失控。每个路由都会实时检测周围节点的流量等信息,一旦发现攻击可以指控撤销该节点。撤销方案的安全性取决于安全比率,当安全比率为 2/3 时,只要网络内超过 2/3 路由节点可信,该网络就是安全的。只有网络合法节点可以发起指控,区块链的防篡改且不易伪造的性质保证了撤销不会被对手伪造。整个 WMN 网络节点实行公钥分级管理,客户端公钥由所属路由器节点管理,普通路由节点由所属管理节点控制,所有管理节点共同维护整个网络,但不能操作其他管理节点下的节点公钥。该机制使得任何一个节点的失控都不至于影响全网络,普通路由节点失控只会影响所控制的客户端,单一管理节点失控也不会影响所管理节点机构内的其他节点。

3)密钥泄露。网络中最危险的情况就是密钥泄露,一旦管理节点账户地址私钥泄露,被敌手获取,敌手可以随意增加恶意节点、删除该管理节点所维护的节点。因此本文给出两种模式预防密钥泄露:(1)离线管理节点模式,管理节点只负责网络初始建立,网络建立后,管理节点不再起作用,节点公钥撤销和添加新节点都需要网络所有节点确认,这样敌手很难拿到管理节点地址私钥;(2)多管理节点模式,在管理合约注册时注册多个管理节点而不是单一管理节点,只有在所有管理节点都允许时,才能对节点撤销和确认注册。这样避免了由单一节点失控(宕机或被攻击等)导致整个网络失控的情况发生。建议同时采用以上两种模式,充分发挥区块链的去中心化、分布式作用。

4)单点故障/拒绝服务攻击。本文方案基于区块链技术,所有数据在每个节点中都有备份,而且由于基于安全哈希函数的 merkle 树保证了数据的完整性,因此单节点(包括 CA)故障不影响系统运行。区块链抗拒绝服务攻击,区块链的操作需要全节点验证才能写入区块中,对系统发起拒绝服务攻击是无效的。抵抗拒绝服务攻击的能力与节点规模相关,节点越多,抵抗能力越强。

5)区块链安全性。本文方案采用私有链而不是公有链,主要是为了抵抗单点故障和拒绝服务攻击,假定系统中存在故障节点(不能提供服务,无法与之通信),但不存在拜占庭节点(恶意节点/故障节点)。私有链和联盟链是有权限的区块链,是不完全去中心的,不像公有链节点可随时任意加入和退出,在私有链中节点加入是受管理中心控制的,节点可信,不会产生 51% 攻击。

6)安全通信。通信消息经过基于 ECDH 生成的对称秘钥加密,保证了数据机密性、隐私性。该方案中 ECDH 是防中间人攻击的,因为区块链保证了公钥的不可篡改性和防伪造性。当确保了公钥正确性后,ECDH 算法保证了私钥不可泄露,以及共享密钥只能为通信双方已知。

4　实验结果与分析

本文利用以太坊模拟路由节点,搭建区块链网络进行性能分析。实验在处理器为 Intel© Core™ i7-6700@3.4 GHz Ubuntu 16.04 操作系统上完成。使用以太坊 go-ethereum[26] 搭建以太坊私有链,配置创世区块,其中难度为公有链的 1/65 000(公有链难度

计算机工程 www.ecice06.com

约为 170 亿),在实际使用中根据设备性能和安全性需要合理配置创世区块难度。

初始创建 10 个节点,使用 remix-ide[27] 编写智能合约。将管理合约部署在任意一个节点(相当于管理节点,本文实验将节点 1 作为管理节点),调用合约添加其他节点(见图 3)。

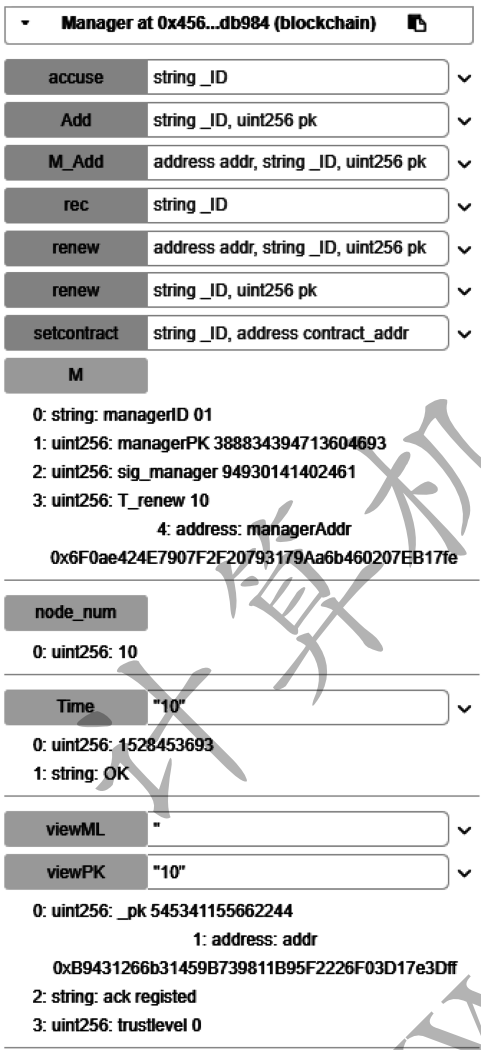


图 3 remix-ide 网页端管理合约界面及添加 10 个节点的操作

可以看出管理合约的基本函数,*M* 为管理节点基本信息,*node_num* 为 10,说明网络中 10 个节点已添加。在节点 1(管理节点)指控撤销节点 10(见图 4),显示了查看节点 10 的指控者列表只有 01,节点数量变为 9,并查看了 2 号节点公钥。当不同节点(非管理节点)指控节点 9 时(见图 5),4 个节点指控后,节点数量没变,9 号公钥状态变为可疑(见图 5(a));7 个节点指控后,节点数量变为 8,查询 9 号节点公钥变为空(None)(见图 5(b))。节点 2 更新自己的公钥(见图 6),与图 3 比较可以看出公钥已被更新,从而证明本文方案可通过管理合约注册添加节点,更新、指控、撤销公钥,查询公钥信息等基本操作。

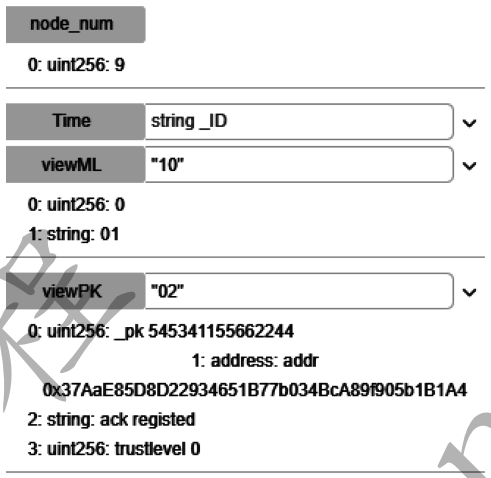


图 4 撤销节点 10 公钥的操作

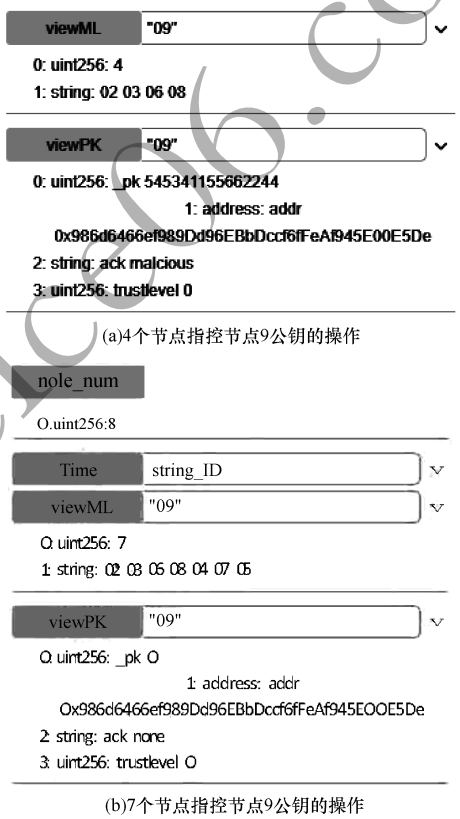


图 5 撤销节点 9 公钥的操作

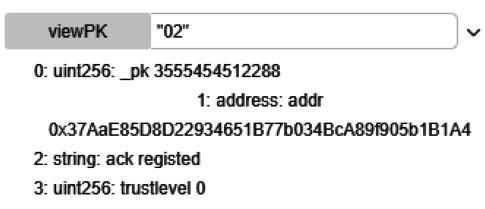


图 6 更新节点 2 公钥的操作

计算机工程 www.ecice06.com

通过实验分析更新、指控撤销的时间发现,这些时间完全依赖于“挖矿”时间,即交易达成共识,被“矿工”写入到区块的时间。这是由于添加、更新、撤销、指控等修改数据的操作都要通过交易(事务)来修改、添加区块中的数据。“挖矿”时间在以太坊公有链中需要 12 s,私有链需要的时间稍短一些(5 s 或者更短,这主要取决于节点数量和私有链安全配置),而在本文实验配置中平均需要 1.58 s,当节点数增加到 20 时,时间几乎没有变化,而难度增大到公有链的 1/4 000,时间增大到 5.8 s。在节点计算能力没有明显增强时,节点数量对“挖矿”时间影响较小,而创世区块难度直接影响“挖矿”时间。所以,应折中考虑安全和效率因素,合理配置区块链。以上操作可能会导致长时间的延迟。在某些情况下,可能不足以等待这么长时间。例如,节点发现可疑情况撤销公钥时,攻击者可以在公钥撤销的操作达成共识前攻击网络。但是本文方案中的路由节点可以即时获取任意已注册节点的公钥,因为查询操作不需要使用交易,每个节点储存了整个区块链副本,而通信中 ECDH 和对称加密耗时较小,所以本文方案安全通信效率较高。在一般情况下,只在网络空闲时间更新公钥,更新的高延迟不影响实际网络的使用。

5　结束语

本文针对 WMN,提出基于智能合约的安全架构(SCSA)。在 SCSA 中,将 Mesh 路由充当区块链节点构建区块链私有链网络,在区块链网络上部署了 3 类智能合约:联盟合约,管理合约和客户端合约。联盟合约管理不同机构区域的节点,管理合约实现对系统内公钥设定有效期、更新、撤销操作,客户端合约管理路由节点下的客户端。3 类合约实现了对 WMN 的分级管理,单一节点失控仅会影响小范围内的节点,从而增强网络安全性。由于区块链分布式存储数据,智能合约能够实时访问数据,因此利用 ECDH 密钥交换算法实现了高效可靠的安全通信。另外,联盟合约也增强了网络的拓展性。下一步将降低区块链共识中公钥更新、撤销等操作的延时,并通过各区域公钥参数的独立设置,实现跨自治域的认证和通信。

参考文献

[1] 柴远波,郑晶晶. 无线 Mesh 网络应用技术[M]. 北京:电子工业出版社,2015.

[2] AKYILDIZA I F, WANG Xudong, WANG Weilin. Wireless Mesh networks: a survey [J]. Computer Networks,2005,47(4):445-487.

[3] SIDDIQUI M S, CHOONG S V. Security issues in wireless Mesh networks [C]//Proceedings of International Conference on Multimedia and Ubiquitous Engineering. Washington D. C., USA: IEEE Computer Society,2007:717-722.

[4] MARTIGNON F,PARIS S,CAPONE A. DSA-Mesh: a distributed security architecture for wireless mesh networks[J]. Security and Communication Networks, 2011,4(3):242-256.

[5] 朱建明,高胜,段美姣. 区块链技术与应用[M]. 北京:机械工业出版社,2018.

[6] DONET J A D, PÉREZ-SOLA C. The bitcoin P2P network[C]//Proceedings of International Conference on Financial Cryptography and Data Security. Berlin, Germany: Springer,2014:87-102.

[7] STALLINGS W. Cryptography and network security: principles and practice [J]. International Annals of Criminology,2003,46(4):121-136.

[8] MATTILA J. The blockchain phenomenon——the disruptive potential of distributed consensus architectures [EB/OL]. [2018-10-12]. https://ideas. repec. org/p/rif/wpaper/38. html.

[9] PETERS G W, PANAYI E. Understanding modern banking ledgers through blockchain technologies: future of transaction processing and smart contracts on the internet of money [M]. Berlin, Germany: Springer, 2016:239-278.

[10] EVANS D. Economic aspects of bitcoin and other decentralized public-ledger currency platforms [J]. SSRN Electronic Journal,2014,685:203-231.

[11] WOOD G. Ethereum: a secure decentralized generalised transaction ledger [EB/OL]. [2018-10-12]. http://gavwood. com/paper. pdf.

[12] SOMPOLINSKY Y, ZOHAR A. Secure high-rate transaction processing in bitcoin [C]//Proceedings of International Conference on Financial Cryptography and Data Security. Berlin, Germany: Springer, 2015: 507-527.

[13] CHATTERJEE K, DE A, GUPTA D. Mutual authentication protocol using hyperelliptic curve cryptosystem in constrained devices [J]. International Journal of Network Security,2013,15(1):9-15.

[14] SHI Wenbo, GONG Peng. A new user authentication protocol for wireless sensor networks using elliptic curves cryptography [J]. International Journal of Distributed Sensor Networks,2013,9(4):1-10.

[15] 许力,王栋城,苏彬庭,等. 基于身份盲签名的无线 Mesh 网络匿名切换认证方案[J]. 工程科学与技术,2018,50(2):148-153.

(下转第 31 页)

计算机工程 www.ecice06.com

(上接第 23 页)

[16] LAI Y M, CHENG P J, LEE C C, et al. A new ticket-based authentication mechanism for fast handover in mesh network[J]. PloS One, 2016, 11(5): 1-7.

[17] GAO Tianhan, MIAO Qidi, GUO Nan. Anonymous authentication scheme based on identity-based proxy group signature for wireless Mesh network [J]. EURASIP Journal on Wireless Communications and Networking, 2016(1): 193.

[18] ALTOP D K, BINGÖL M A, LEVI A, et al. DKEM: secure and efficient distributed key establishment protocol for wireless Mesh networks [J]. Ad Hoc Networks, 2017, 54: 53-68.

[19] NAKAMOTO S. Bitcoin: a peer-to-peer electronic cash system [EB/OL]. [2018-10-12]. http://bitcoin.org/bitcoin.pdf.

[20] FROMKNECHT C, VELICANU D, YAKOUBOV S. Certcoin: a namecoin based decentralized authentication system[EB/OL]. [2018-10-12]. http://courses.csail.mit.edu/6.857/2014/files/19-fromknecht-velicann-yakoubovcertcoin.pdf.

[21] FROMKNECHT C, VELICANU D, YAKOUBOV S. A decentralized public key infrastructure with identity retention[EB/OL]. [2018-10-12]. https://eprint.iacr.org/2014/803.pdf.

[22] AL-BASSAM M. SCPKI: a smart contract-based PKI and identity system[C]//Proceedings of ACM Workshop on Blockchain, Cryptocurrencies and Contracts. New York, USA: ACM Press, 2017: 35-40.

[23] 唐春明,高隆. 区块链系统下的多方密钥协商协议[J]. 信息网络安全,2017(12):17-21.

[24] 胡逸飞,熊焰,黄文超. 基于区块链审计的公钥分发方案[J]. 计算机工程,2019,45(5):29-34.

[25] ZHANG Yongguang, LEE Wenke, HUANG Yian. Intrusion detection techniques for mobile wireless networks[J]. Wireless Networks, 2003, 9(5): 545-556.

[26] Go-ethereum[EB/OL]. [2018-10-12]. https://github.com/ethereum/Go-ethereum.

[27] Remix-ide[EB/OL]. [2018-10-12]. https://github.com/ethereum/remix-ide.

编辑　陆燕菲

计算机工程 www.ecice06.com