计算机研究与发展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202111095
60(2): 386−397, 2023

基于区块链的工业互联网动态密钥管理

张泽林　王化群
（南京邮电大学计算机学院　南京　210023）
（zlzh_ang@163.com）

Dynamic Key Management of Industrial Internet Based on Blockchain

Zhang Zelin and Wang Huaqun
（*School of Computer Science*, *Nanjing University of Posts and Telecommunications*, *Nanjing* 210023）

Abstract　At present, the security threat of industrial Internet data is becoming more and more serious. Effective network transmission plays a key role in the data security of industrial Internet. In order to effectively adapt to the structure of the industrial Internet and achieve safe and reliable communication, a key management scheme based on blockchain dynamic nodes is proposed. In industrial communication, an effective session key needs to be established between untrusted nodes, and the traditional key agreement is realized by using a central node in the way of middleman. However, Once the central node fails, the communication of the whole communication system will fail. When the number of nodes n is small, the central node is usually used to set the key in advance. Each node needs to store $(n-1)$ keys, so the whole system needs to store $n(n-1)$ keys. Once the nodes need to be increased, the keys stored in the whole system will increase exponentially. Therefore, based on the blockchain, we use polynomials to construct communication keys, decentralize to generate shared keys, effectively resist the threat of node access to the system, and ensure effective group key negotiation. The propsosed scheme has faster processing speed than that of traditional PKI. If any node loses its key, it can effectively recover the original key with the help of other node information.

Key words　blockchain; industrial Internet; key management; secret sharing; Kate promises

摘　要　目前，工业互联网数据面临的安全威胁日益严重，有效的网络传输对工业互联网的数据安全起到关键的作用.为了能有效适应工业互联网的结构，实现安全可靠的通信，提出一种基于区块链的动态节点的密钥管理方案.工业通信中，不信任的节点之间需要建立有效的会话密钥，并且传统的借助一个中心节点采用中间人方式实现密钥协商，但是一旦中心节点失效，则会导致整个通信系统的通信失败.而当节点数n较少时，通常采用中心节点预先设置密钥的方式，每个节点均需存储$(n-1)$个密钥，那么整个系统需要存储$n(n-1)$个密钥，一旦节点需要增加，则整个系统存储的密钥就会呈指数级增长.因此，基于区块链，利用多项式来构建通信密钥，去中心化生成共享密钥，并有效抵御节点出入对系统的威胁，保证有效的组密钥协商，比传统的PKI具有更快的处理速度，且任何节点丢失密钥，可以借助其他节点信息有效地恢复原有密钥.

关键词　区块链；工业互联网；密钥管理；秘密共享；Kate承诺

中图法分类号　TP309

收稿日期：2021−11−08；**修回日期**：2022−04−15

基金项目：国家自然科学基金项目（61872192）；江苏省高等学校自然科学研究项目（19KJA310010）
This work was supported by the National Natural Science Foundation of China (61872192) and the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (19KJA310010).

通信作者：王化群（whq@njupt.edu.cn）

2020 年 4 月，“新基建”被正式提出，工业互联网技术越来越得到国家的重视，未来将成为国家经济发展的重点，也是各国发展工业的新优势.而随着工业互联网技术的发展，工业互联网数据也迅速地增长，数据安全逐渐成为工业互联网发展的重要保障[1].

本文主要针对钢铁生产中的智能设备点检系统数据传输安全，传统设备点检工作主要是通过专业人员去现场实时采集数据、分析数据，发现问题通知专业设备人员进行处理，处理完毕再进行验收反馈.点检人员操作繁琐，工作效率低；人工输入数据，工作错误率高；各部门没有有效的监督机制，容易出现遗漏；工作环境恶劣，容易出现烫伤、密闭空间窒息等安全隐患.智能点检系统通过网络传输信息，通过传感器将设备运行状态实时传输到服务器中，工作人员可以远程实时查看设备运行数据.目前，智能点检系统在某些钢铁企业得到应用，大量的信号采集，实时传输使得设备的可靠性以及生产的效率得到了显著提升，尤其是点检人员的点检效率也得到了提高，及时发现并解决故障，使设备得到更长的使用寿命，预防了设备事故的发生.

由于智能点检需要大量的传感器，传感器的数据需要进行整合分析上传到服务器，并且传感器设备之间也需要进行实时有效安全的传输数据. 比如：点检系统中涉及铁水脱硫、混铁炉、转炉冶炼、吹氩、精炼以及连铸等多个工序，工序之间的传感器设备之间需要有效地及时信息交互，才能准确地判断设备生产状态.因此，数据安全是保障炼钢点检智能化发展的重要前提.与普通互联网不同的是，一旦智能系统上的机密数据遭到泄露，可能将会导致企业失去核心竞争力，并且生产过程中的设备控制权限、状态参数如果被不法分子截获篡改，则会影响设备安全运行，导致严重的安全事故，危害国家安全、经济发展以及社会稳定.

现在的工业现场通信面对大量的节点设备，需要建立安全可靠的信息传输.密钥管理在工业互联网中便起到了重要作用[2-3].近几年，互联网中的密钥管理引起了众多研究者的兴趣.密钥管理主要分为集中式密钥管理和分布式密钥管理 2 种类型.

集中式密钥管理即为由单一中心节点负责生成、分发和更新系统中节点所用的加密密钥或者会话密钥[4-8].Sun 等人在文献 [4] 中提出了一种多组密钥管理方案，实现了分层的组通信访问控制.在文献 [5] 中，Je 等人提出一种安全组播通信的密钥树管理协议以及一种高效计算存储的密钥树结构；接着，冯力等人在文献 [6] 中提出基于单项散列函数的多级密钥管理方案，根据访问控制矩阵实现安全高效的多密级授权访问；然后，Gao 等人在文献 [7] 中提出一种基于椭圆曲线的匿名多播方案，利用多项式进行分发，具有更高的效率.

分布式密钥管理即为无单一中心节点，密钥由所有成员共同管理[9-12].Lou 等人在文献 [9] 中提出了一种基于区块链的密钥管理方案，解决通信实体之间缺乏信任的问题.Zhao 等人在文献 [10] 中设计了一种轻量级高效可恢复的密钥管理方案，用于保护医疗数据隐私信息.Lei 等人在文献 [12] 中提出一种基于车联网的组密钥管理方案，利用区块链实现了异构车辆通信系统中的分布式密钥管理方案.

集中式密钥管理方案的优点是较低的计算和传输的成本开销.但是需要一个可信第三方来充当密钥生成中心（key generation center, KGC），在注册阶段与每个用户建立成对的共享密钥. 在组通信阶段，由 KGC 首先选择生成组密钥并通过共享密钥加密后分发给每个组用户.因此，需要可信第三方 KGC 来保证通信系统的安全.但是，一旦 KGC 被攻陷，则会导致整个通信系统的崩溃.在工业通信现场，任何节点随时都有可能遭到攻击，因此去中心化的密钥管理方案是非常有必要的.

近几年，随着 5G 技术的普及，工业互联网中大多数应用程序都是将数据通过一个集中的云服务器进行存储或者处理的.然而，集中处理的服务器对于大规模系统也存在着单个中心节点被攻陷的威胁.而且对于大型应用如炼钢产业、车联网等需要低延迟网络通信来说，集中式密钥管理方案可能并不是最优的.

利用边缘计算结合区块链则可以解决通信延迟的问题.边缘节点一般具有较高的存储以及计算能力，可以给较低存储以及较低计算能力的传感器节点提供低延迟的服务，并且可以为云服务器减少计算负担[13].区块链可以保证存储数据的公开透明以及不可篡改.两者相结合可以为工业互联网提供更高的性能以及安全性[14-15].最近，新兴的边缘计算[16-17] 以及区块链技术[18] 已经得到了广泛的研究. Ning 等人在文献 [19] 中提出在车联网中部署高计算、高存储的路边单元（road side unit, RSU），实现车辆通过路边单元进行高效的信息传输.Ning 等人在文献 [20] 提出一种基于边缘计算的 5G 健康监测系统.在文献 [21] 中，Yang 等人提出通过边缘节点与云之间的相互交互来处理工业互联网的大数据流；接着，Pan 等人在文献 [22] 中设计了一种基于区块链以及智能合约的边缘

物联网框架,有效地将边缘计算以及区块链的优势整合到了物联网中.

本文通过引入区块链技术结合边缘计算,有效避免中心节点会被攻陷而产生的安全问题.通过使用二元多项式产生密钥,保证较快的处理速度和时效性,并且能够有效适应节点的出入,保证节点之间的有效通信与系统通信的安全性.

本文给出一种不同于传统的新的访问结构,使用非对称双变量多项式构建子份额,由初始元节点(一般选择使用寿命长、工作环境较好、不会轻易下线的传感器设备)共同产生多项式,而非单个中心节点产生,使每个节点有效获得安全的子份额.在有节点丢失子份额时,可通过其他节点有效恢复子份额.非对称双变量多项式中双变量中不同的阶数提供不同的阈值,一方面保证在短时间大量节点加入时多项式密钥的安全性;另一方面保证在稳定状态,即无节点出入时期,能有效恢复份额,进行有效通信.而且,在新节点加入时,节点获得的高阈值份额不能够获取其他节点的任何有效信息,但是仍能够与已加入节点进行有效通信.

本文主要贡献有3个方面:

1)提出一种基于二元多项式的去中心化密钥管理方案.节点的份额由二元多项式生成,具有较快的处理速度以及较低的存储开销,因为无需存储大量成对的共享密钥,只需要通过存储二元多项式的系数即可.初始的二元多项式由所有元节点共同生成,并分发给其他普通节点,使密钥管理具有去中心化特点.

2)通过阈值切换,有效抵御节点出入对通信系统的威胁.方案使用的是非对称二元多项式,具有2个不同阶数的变量,能够有效地进行不同阈值的切换,提高通信系统的安全性.

3)利用区块链技术以及Kate承诺,实现安全的点对点通信以及多组通信.由于区块链去中心化、不可篡改的特点,再结合Kate承诺能够保证节点之间更安全有效的信息交互.

1 相关技术

1.1 区块链

区块链[23]是一种去中心化、不可篡改的数字账本.不同于传统系统的中心化的特征,区块链能够在无信任环境下进行安全可验证的交易计算.随着区块链的快速发展,区块链在大数据、云计算等领域都得到了广泛的应用.

目前,区块链主要分为3种链,分别是公有链、私有链以及联盟链.在公有链上,各节点可以自由进出并且参与链上数据的读写;私有链必须得到授权的节点才能加入,读写权限也只能有选择地开放;联盟链是基于多个不同的机构共同管理的区块链,数据只允许不同的机构进行读写.许多商业业务根据需求选择使用不同的区块链[24-25]来改善传统信息化系统,通过区块链保护用户的密钥以及敏感隐私数据.

1.2 Kate承诺

Kate承诺是Kate等人[26]在2010年提出的多项式承诺方案.

方案主要分4个步骤:

1)*Setup*.选择一个合适的双线性对群组(p, G, G_T, e, g),其中生成元为G,配对函数$e: G\times G = G_T$,假设多项式最大阶数为t,随机选择一个私钥$sk = s$,则公钥$pk = g^s, g^{s^2}, \cdots, g^{s^t}$.将公钥公开,并销毁(遗忘)私钥.

2)*Commit*$(\varphi(x), pk)$.设要承诺的多项式为$\varphi(x) = \sum_{j=0}^{d} a_j x^j$,其中多项式阶数$d \leqslant t$,通过公钥$pk$计算承诺$C = \prod_{j=0}^{d} (g^{s^j})^{a_j}$.

3)*CreateWitness*$(\varphi(i), i, pk)$.计算目标多项式$x = i$处的多项式值$\varphi(i)$,并计算

$$\varphi(x) - \varphi(i) = (x - i)\times\omega(x),$$

设$W_i = g^{\omega(s)}$.其中W_i即为多项式$\varphi(x)$在$x = i$处的证据.

4)*VerifyEval*$(C, i, \varphi(i), W_i)$.输入$\varphi(i)$,承诺C以及证据W_i.验证等式是否成立:

$$e(C, g) = e\left(W_i, \frac{g^s}{g^i}\right)\times e(g, g)^{\varphi(i)}.$$

若等式成立,则承诺C所表示的多项式$\varphi(x)$在i处的值$\varphi(i)$是正确的.

其验证的公式推导过程为:

$$e\left(W_i, \frac{g^s}{g^i}\right)\times e(g, g)^{\varphi(i)} = e\left(g^{\omega(s)}, g^{s-i}\right)\times e(g, g)^{\varphi(i)} =$$
$$e(g, g)^{\omega(s)(s-i)+\varphi(i)} = e(g, g)^{\varphi(s)} = e(C, g).$$

1.3 Shamir秘密共享

Shamir秘密分享[27]是基于多项式的可信第三方D在$\mathbb{Z}_q$上生成的一个$t-1$阶多项式$f(x)$,并且令$f(0) = s$,其中s即为秘密并且$s\in\mathbb{Z}_q$.D生成不同的份额$f(x_i)$, $i = 1, 2, \cdots, n$,其中x_i是每个成员相应的公开信息.任意t个成员可通过拉格朗日插值法恢复秘密

值，即$s=f(0)=\sum_{i=1}^{t}f(x_i)\prod_{r=1,r\neq i}^{t}\frac{-x_r}{x_i-x_r}$.Shamir 秘密分享需要满足 2 个安全要求：1)拥有大于等于t个份额即可恢复秘密；2)拥有少于t个份额则不能够获得该秘密的任何信息.

2 系统模型与安全模型

2.1 系统模型

本系统将需要通信的设备节点分为 3 种：普通节点、元节点以及新节点，它们的关系如图 1 所示.

1)普通节点.所有分布在炼钢产业线上点检系统的传感器.对现场的设备进行检查、采集数据，并上传到数据库中.

2)元节点.从普通节点中选取的一部分(也可以是全部)作为元节点，一般选择使用寿命长、工作环境较好、不会轻易下线的传感器设备作为元节点.元节点需要互相合作生成初始的多项式，并将份额分发给其他所有节点，以达到节点之间互相通信的目的.本系统可容忍元节点为半诚实节点(honest-but-curious)(诚实且好奇的节点)，即使部分元节点离线或者被攻陷，仍无法影响其他节点持有份额信息的安全以及通信系统的正常运作.

3)新节点.需要加入点检系统的新的传感器节点.新节点的加入需要一定数量的普通节点的帮助.如果普通节点的数量不够，则全部的元节点也可以帮助新节点成功加入通信系统.

系统模型如图 1 所示.其中，低阈值份额与高阈值份额定义为：

1)低阈值份额.非对称双变量多项式中，2 个变量的阶数分别为t, n', 满足$t<n'$, 同时获得t阶与n'阶份额时，阈值由较低阶数的份额而定，即阈值为t的份额而定，此即为低阈值份额.

2)高阈值份额.与 1)同理，当且仅当获得n'阶的份额时，即为高阈值份额.

首先，选取部分或者全部的传感器节点作为元节点，多个元节点共同生成随机多项式，并将低阈值份额发送给所有其他节点.其他节点收到并计算出自己相应的完整份额，并且不可能得知不属于自己的份额信息.节点通过自己份额可计算出与其他节点的成对的共享密钥，并按照约定通过共享密钥进行点对点通信.其次，节点之间可以通过成对的共享密钥协商组密钥，并且通过区块链上摘要来验证发起者发送的组通信信息的正确性，进行安全的组通信.再次，新节点要加入时，需要一定数量的普通节点发送份额信息来帮助新节点获得部分高阈值份额，并通过发送到区块链上的承诺来验证份额信息的正确性，进而计算出可与通信系统中的部分节点进行通信的新节点自己的部分份额，而加入期间获得的部分高阈值份额保证通信系统可容忍较多节点同时加入且不影响通信系统安全.如果普通节点数量不够，可通过所有的元节点发送信息获得份额.最后，节点加入完毕后，一定数量的普通节点可帮助新节点恢复全部份额，进而新节点则成功加入通信系统与所有其他节点进行有效通信.如果普通节点数量不够，则可通过所有元节点发送信息恢复全部份额.

2.2 区块链部署结构

本文方案使用的是端—边—云的层次区块链结构，如图 2 所示，系统总共包含 3 层：设备层、边缘区块链层以及云区块链层.具体描述为：

1)设备层.设备层一般包括各种各样的基础设备节点(例如工业计算机和边缘路由器等)，本文方案中主要包含普通节点以及元节点.将设备层部署在工业产业中，执行检测、取样等相关工作.由于计算以

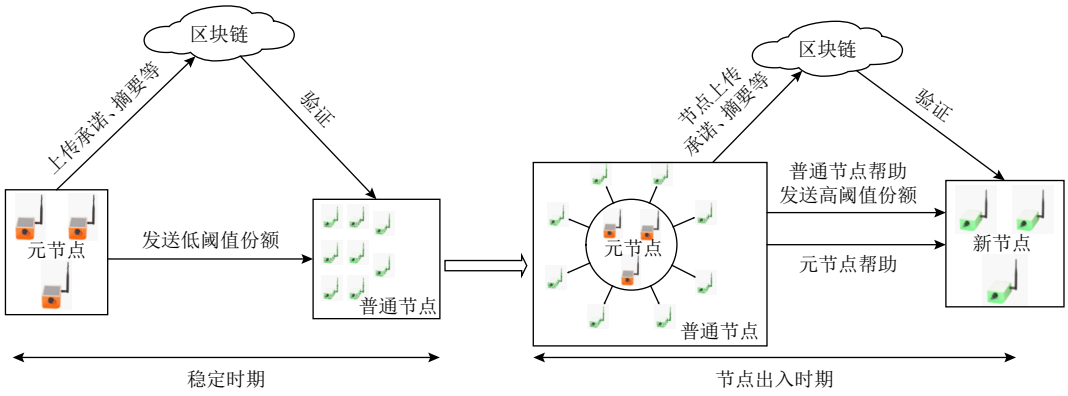

Fig. 1 Dynamic key management model of industrial network based on blockchain

图 1 基于区块链的工业互联网动态密钥管理模型

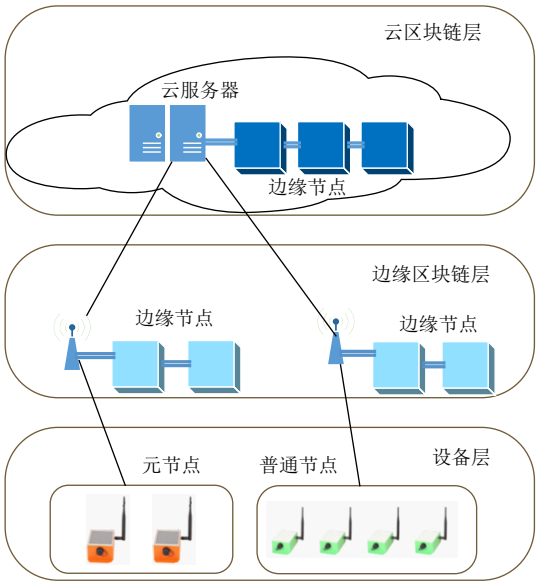

Fig. 2 Blockchain deployment structure

图 2 区块链部署结构

及存储能力有限,可将节点之间密钥协商所需要的验证数据发送到边缘层的边缘节点中.边缘层的存在使得设备区的节点查询或者区块链数据的上传变得更加快捷方便.

2)边缘区块链层.边缘区块链层是由大量的边缘节点(如基站)组成,它们一般有着较大的存储空间以及计算能力,接受设备层传来的数据,并可以低延迟地对数据进行集合、封装等相关处理.边缘计算节点通常更靠近基础设备节点,因此可有效降低通信时延,提高通信效率.

3)云区块链层.云区块链层借助其具有数据一致性以及防篡改的特点,保证信息的完整性、时序性.在本文方案中主要用于协助实现节点认证以及消息验证等功能.

在工业互联网环境中部署基于边缘计算的层次区块链,一方面,利用边缘服务器固有的较高计算能力以及存储空间,为基础设备节点提供低延迟的服务,并为云服务器分担计算负担;另一方面,区块链是可靠的分布式记账本,通过共识机制来保证事务数据的不可篡改等特性.

2.3 安全模型

本节提出的本文方案的安全需求具体描述为:

1)可防御性.任何时期,敌手攻陷低于一定数量的节点(包含部分元节点)时无法获取任何其他节点的信息.

2)可恢复性.任何时期,有节点丢失份额时都可以通过其他节点帮助恢复原有份额信息.

3)保密性.组通信时期,只有组成员知道组密钥并且知晓所有组内成员身份,组外成员无法得知密钥以及组内成员的身份.

本文方案系统的阈值切换如图 3 所示,其中 $t<n'$.由图 3 可知,敌手性质为:

1)稳定期间即无节点出入期间,敌手最多可以攻陷t个节点(包含部分元节点),串通攻击无法得知其他任何节点的份额信息.

2)节点出入期间,敌手最多可以攻陷n'个节点(包含部分元节点),串通攻击依旧无法得知其他节点的份额信息.

3)即使敌手攻陷部分元节点,仍无法得到任何其他节点的份额信息.

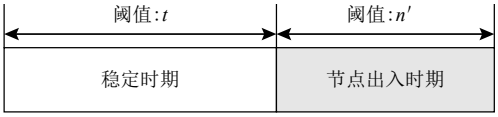

Fig. 3 Schematic diagram of threshold switching

图 3 阈值切换示意图

3 基于区块链的动态密钥管理方案

本节主要提出适应工业互联网动态节点的密钥管理方案.方案分为 4 个阶段:密钥生成、组密钥协商、节点变更、份额恢复.

1)密钥生成. 无需单个中心节点,通过初始元节点共同生成二元多项式,并分发给所有节点,节点自行构建子份额.节点可以通过子份额得到节点之间成对的共享密钥.

2)组密钥协商. 节点可以发起组通信,通过成对的共享密钥进行组密钥协商,利用区块链实现可靠的组通信.

3)节点变更. 当有节点加入时,可以容忍大量节点同时加入,不会危及到通信系统任何其他节点的密钥安全,并且能及时与部分原有节点进行有效通信.

4)份额恢复. 节点变更完毕,系统稳定以后,通过其他节点来恢复新加入节点的全部份额,保证新加入节点跟其他所有节点的有效通信.

具体结构为:令 p 是一个大素数,$\{P_i\}_{i\in[n]}$是节点成员集合,$\{P'_j\}_{j\in[m]}$是新加入的节点成员.为了阅读方便,在表 1 总结了这些符号以及对应的描述.

3.1 密钥生成

1)选取$\{P_i\}_{i\in[n]}$中部分节点$\{P_i\}_{i\in[r]}(r\leqslant n)$为元节点,然后每个元节点$P_i$随机生成一个非对称双变量多项

Table 1 Symbols and Their Descriptions

表 1 符号及其描述

符号	描述
P	普通节点
P'	新节点
n	普通节点个数
r	元节点个数
m	新节点个数
r_i	节点P_i生成的随机数
x_i	节点P_i对应的公开信息
$C_{F(x_i,y)}$	多项式$F(x_i, y)$的承诺
$W_{F(x_i,x_j)}$	多项式$F(x_i, y)$在x_j处的证据
$k_{i,j}$	P_i与P_j的共享密钥
K	组通信密钥
H	单向散列函数
R_H	摘要

式$f_i(x, y) = (a^i_{0,0} + a^i_{1,0}x + a^i_{0,1}y + a^i_{1,1}xy + a^i_{2,0}x^2 + a^i_{0,2}y^2 + \cdots + a^i_{t,n'}x^t y^{n'}) \bmod p$, 其中$x$的阶数为$t$, y的阶数为n', $a^{i,j} \in GF(p)$, 并且满足$t < n' \leqslant n$.

2)P_i根据其他节点P_j相应的公开信息x_j计算出$f_i(x_j, y)$以及$f_i(x, x_j)$, 其中满足$j \in [n]$且$j \neq i$.并将生成的子份额$\{f_i(x_j, y), f_i(x, x_j)\} i \in [n]$, $i \neq j$通过安全信道发送给相应的节点P_j.

3)P_j收到其他所有节点发送的子份额$\{f_i(x_j, y), f_i(x, x_j)\}_{i\in[n], i\neq j}$, 首先检查收到子份额是否满足$x$的阶数为$t$, y的阶数为n', 若满足, 则继续计算

$$F(x_j, y) = \sum_{i=1}^{r} f_i(x_j, y) \bmod p,$$

$$F(x, x_j) = \sum_{i=1}^{r} f_i(x, x_j) \bmod p,$$

其中$F(x_j, y)$阶数为n', $F(x, x_j)$阶数为t.并根据 Kate 承诺协议计算$F(x_j, y)$与$F(x, x_j)$的承诺$C_{F(x_j,y)}$以及$C_{F(x,x_j)}$(计算方法参考 1.2 节), 将其上传到区块链上.然后生成一个随机数r_j, 满足$r_j \neq 0$, 并将随机数r_j广播给其他成员.

4)每个节点P_i根据其他节点相应的公开信息x_j, 并通过持有的份额$F(x_i, y)$以及$F(x, x_i)$, 计算出$k_{i,j} = F(x_i, x_j)$或者$k_{j,i} = F(x_j, x_i)$. $k_{i,j}$, $k_{j,i}$即为P_i与P_j的共享密钥, 并约定如果$i < j$, 则使用$k_{i,j}$作为彼此的会话密钥; 否则使用$k_{j,i}$作为会话密钥.

3.2 组密钥协商

假设P_i要发起组密钥通信, 其中组通信成员设为节点子集$\{P_{l_j} | P_{l_j} \in P, j = 1, 2, \cdots, m\}$, 其中$1 \leqslant m \leqslant n$.

1)P_i随机生成组密钥K, 并获取每个组成员P_{l_j}的随机数$\{r_{l_j}\}_{j\in[m]}$, 根据组成员的随机数构建$R_H = H(r_{l_1}, r_{l_2}, \cdots, r_{l_m}, K)$, 其中$r_{l_j} \neq 0$, 并将$R_H$发送到区块链上.

2)P_i根据开始时其他节点广播的随机数构建$C_i = \{K, (l_1, r_{l_1}), (l_2, r_{l_2}), \cdots, (l_m, r_{l_m}), R_H\}$, 然后根据关系使用$k_{i,l_j}$或者$k_{l_j,i}$对$C_i$进行加密, 即$S_{i,l_j} = E_{k_{i,l_j} \text{or} k_{l_j,i}}(C_i)$.最后将$S_{i,l_j}$发送给相应的$P_{l_j}$.

3)P_{l_j}收到P_i发来的S_{i,l_j}后, 根据关系使用k_{i,l_j}或者$k_{l_j,i}$对S_{i,l_j}解密, 即$C_i = D_{k_{i,l_j} \text{or} k_{l_j,i}}(S_{i,l_j})$.$P_{l_j}$获取$C_i$后, 首先判断一开始收到的$(l_i, r_{l_i})$ $(i = 1, 2, \cdots, m)$与C_i中的(l_i, r_{l_i})是否相同并且满足$r_{l_i} \neq 0$; 然后判断$R_H = H(r_{l_1}, r_{l_2}, \cdots, r_{l_m}, K)$是否满足; 接着检查$R_H$与区块链上的$R_H$是否相等.若这些等式都满足, P_{l_j}则接受组密钥K作为$\{P_{l_j}\}_{j\in[m]}$的组通信密钥.

3.3 节点变更

假设同时要增加节点$\{P'_{l_j}\}_{j\in[m]}$, 其中$m \leqslant n' - t$.

当原有节点设定的阈值过大时, 即$t \geqslant \frac{n}{2}$时, 则不能够保证有足够的诚实节点帮助新节点获得份额, 必须通过元节点的帮助来恢复份额.1.3 节提到, 本系统可容忍元节点为半诚实节点.具体原因会在安全性证明中详细说明.

当原有普通节点阈值$t < \frac{n}{2}$时:

1)普通节点$\{P_i\}_{i\in[n]}$根据每个新节点P'_{l_j}计算$F(x_i, x_{l_j})$, 根据 Kate 承诺协议计算出相应证据$W_{F(x_i,x_{l_j})}$(计算方法参考 1.2 节)并发送$\{F(x_i, x_{l_j}), W_{F(x_i,x_{l_j})}\}$给$P'_{l_j}$.

2)P'_{l_j}收到每个P_i发送的信息, 根据 Kate 承诺协议通过之前发送到区块链上的承诺验证$\{C_{F(x_i,y)}, x_{l_j}, F(x_i, x_{l_j}), W_{F(x_i,x_{l_j})}\}$(验证方法参考 1.2 节), 验证成功后, 任意$t+1$个$\{F(x_i, x_{l_j})\}_{i\in[n]}$即可通过拉格朗日插值法插值获得$F(x, x_{l_j})$.

3)P'_{l_j}将$F(x, x_{l_j})$作为子份额, 根据 Kate 承诺协议计算$F(x, x_{l_j})$的承诺$C_{F(x,x_{l_j})}$, 并将其发送到区块链上.然后, P'_{l_j}可以通过构建密钥$k_{i,l_j} = F(x_i, x_{l_j})$与$P_i$进行可靠通信, 其中需要满足$i < l_j$.

当原有普通节点阈值$t \geqslant \frac{n}{2}$时:

1)所有元节点P_i根据新节点P'_{l_j}计算$f_i(x, x_{l_j})$, 并发送$f_i(x, x_{l_j})$给P'_{l_j}.

2)P'_{l_j}收到每个P_i发送的信息后, 首先检查收到多项式是否满足x的阶数为t, 若满足, 则根据所有收到的$f_i(x, x_{l_j})$计算

$$F\left(x, x_{l_j}\right)=\sum_{i=1}^{r} f_i\left(x, x_{l_j}\right) \bmod p,$$

得到$F\left(x, x_{l_j}\right)$，并将其作为子份额，根据 Kate 承诺协议计算$F\left(x, x_{l_j}\right)$的承诺$C_{F(x, x_{l_j})}$，并将$C_{F(x, x_{l_j})}$发送到区块链上.P'_{l_j}可以通过子份额构建的密钥$k_{i,l_j}=F\left(x_i, x_{l_j}\right)$与$P_i$进行可靠通信，其中需要满足$i<l_j$.

需要退出的节点P_i将随机数r_i改为0并广播给其他成员即可.

此时，短时间新节点只要不超过n'个，则所有新节点串通起来也永远无法得知关于其他节点的任何相关份额信息.如果再串通原有的t个被攻陷的普通节点，则新节点中攻陷节点不能超过$n'-t$个.并且新节点一旦加入即可与部分其他原有节点建立共享密钥，进行有效通信.

新节点加入需要$t+1$个诚实的普通节点或者所有的元节点.值得注意的是，普通节点也包含元节点，只要保证系统中存在大于阈值t的普通节点即能够帮助新节点加入，因此即使部分元节点被攻陷，也不能影响节点变更阶段.并且由于份额是由所有元节点共同生成，因此攻陷部分元节点也无法获取其他节点的份额信息.

3.4 份额恢复

节点变更完毕，所有节点$\{P_i\}_{i\leqslant[n+m]}$稳定以后，要恢复所有成员全部份额.

由于存在节点出入，故此阶段不一定存在足够的普通节点帮助新节点恢复份额，故当数量不足时，需要元节点帮助恢复新节点的份额.

当所有节点个数$|P_i|\geqslant n'+1$时：

1）每个节点P_i根据新节点P'_{l_j}计算 $F\left(x_{l_j}, x_i\right)$，根据 Kate 承诺协议计算出相应证据$W_{F(x_{l_j}, x_i)}$并发送$\{F\left(x_{l_j}, x_i\right), W_{F(x_{l_j}, x_i)}\}$给$P'_{l_j}$.

2）新节点P'_{l_j}收到P_i的信息，根据 Kate 协议验证$\left\{C_{F(x_{l_j}, x_i)}, x_{l_j}, F\left(x_{l_j}, x_i\right), W_{F(x_{l_j}, x_i)}\right\}$，验证成功后，其中任意$n'+1$个$F\left(x_{l_j}, x_i\right)$即可通过拉格朗日插值法插值获得$F\left(x_{l_j}, y\right)$作为自己的子份额，并生成一个随机数$r_{l_j}$广播给其他所有成员，其中满足$r_{l_j}\neq 0$.

当所有节点个数$|P_i|<n'+1$时：

1）每个元节点P_i根据新节点P'_{l_j}计算 $f_i\left(x_{l_j}, y\right)$，并发送给$P'_{l_j}$.

2）P'_{l_j}收到每个P_i的信息，首先检查收到的多项式是否满足y的阶数为n'，若满足，则根据所有收到的$f_i\left(x_{l_j}, y\right)$计算

$$F\left(x_{l_j}, y\right)=\sum_{i=1}^{r} f_i\left(x_{l_j}, y\right) \bmod p,$$

得到$F\left(x_{l_j}, y\right)$作为自己的子份额，生成一个随机数r_{l_j}并广播给其他成员，其中满足$r_{l_j}\neq 0$.

此时，所有节点成员都持有自己的全部子份额，可根据子份额与其他任意成员构建共享会话密钥，并进一步构建组会话密钥.

新加入的节点恢复份额需要至少$n'+1$个普通节点（包括新加入的节点）或者所有的元节点.

4 安全性分析

4.1 正确性证明

定理 1. 密钥生成阶段，每个节点得到的多项式$F(x_j, y)$与$F(x, x_j)$是多项式$F(x, y)=\sum_{i=1}^{r} f_i(x, y) \bmod p$的子份额.

证明.每个节点得到的多项式是通过$F(x_j, y)=\sum_{i=1}^{r} f_i(x_j, y) \bmod p$计算得来，是多项式$f_i(x, y)$的子份额相加的和，其中$i=1, 2, \cdots, n$.根据秘密共享的同态性质，子份额相加得来的多项式$F(x_j, y)$是$F(x, y)$的子份额. 证毕.

定理 2. 每个节点通过得到的子份额$F(x_j, y)$与$F(x, x_j)$，可以在任意一对节点P_i和P_j之间构建成对的共享密钥$k_{i,j}=F(x_i, x_j)$或者$k_{j,i}=F(x_j, x_i)$, $\forall i, j\in[1, n]$.

证明.由于每个二元多项式，$f_i(x, y)=a^i_{0,0}+a^i_{1,0}x+a^i_{0,1}y+a^i_{1,1}xy+a^i_{2,0}x^2+\cdots+a^i_{t,n'}x^t y^{n'} \bmod p$都是非对称的，其中$x$的阶数为$t$，$y$的阶数为$n'$，因此每个节点得到的子份额，$F(x_j, y)=\sum_{i=1}^{r} f_i(x_j, y) \bmod p$, $F(x, x_j)=\sum_{i=1}^{r} f_i(x, x_j) \bmod p$，也是非对称多项式，任意 2 个节点$P_i$和$P_j$都可以根据自己的份额获得$F(x_j, x_i)=\sum_{l=1}^{r} f_l(x_j, x_i) \bmod p=k_{j,i}$以及$F(x_i, x_j)=\sum_{i=1}^{r} f_i(x_i, x_j)\times \bmod p=k_{i,j}$，并按照约定，当$i<j$时，双方使用$k_{i,j}$作为会话密钥，否则使用$k_{j,i}$作为会话密钥.

每个节点得到的多项式子份额都是由所有初始元节点共同决定的，即$F(x_j, y)=\sum_{i=1}^{r} f_i(x_j, y) \bmod p$.因此，只要有一个元节点是诚实的，那么就能够保证每个节点得到的子份额是随机的，并且每个节点只能够知道自己的份额. 证毕.

定理 3. 节点变更阶段可以通过 2 种方式获得份

额：1）通过$t+1$个普通节点发送来的子份额；2）通过r个元节点发送相应的子份额.

证明.分析2种方式获得份额：

1）新节点P'_{l_j}收到普通节点P_i的子份额$F\left(x_i, x_{l_j}\right)$，并通过任意$t+1$个子份额，插值计算

$$F\left(x, x_{l_j}\right)=\sum_{i=1}^{t+1} F\left(x_i, x_{l_j}\right) \prod_{l=1, l \neq i}^{t+1} \frac{x-x_l}{x_i-x_l} \bmod p,$$

获得P'_{l_j}的子份额$F\left(x, x_{l_j}\right)$.

2）新节点P'_{l_j}收到r个元节点发来的子份额$f_i\left(x, x_{l_j}\right)$，并计算

$$F\left(x, x_{l_j}\right)=\sum_{i=1}^{r} f_i\left(x, x_{l_j}\right) \bmod p,$$

获得P'_{l_j}的子份额$F\left(x, x_{l_j}\right)$. 证毕.

定理4. 节点恢复阶段，可以通过2种方式获得份额：1）通过$n'+1$个普通节点发送来的子份额；2）通过r个元节点发送相应的子份额.

证明. 与定理3不同的是，新节点P'_{l_j}需要得到的份额是$F\left(x_{l_j}, y\right)$而非$F\left(x, x_{l_j}\right)$，因此需要$n'+1$个普通节点发送的子份额或者r个元节点发送相应的子份额. 原理与定理3相同，这里不再过多赘述. 证毕.

4.2 安全性证明

定理5. 可防御性.任何时期，敌手攻陷低于一定数量的节点，无法获取任何其他节点的信息.

证明. 对稳定时期和节点出入时期进行分析.

1）稳定时期（即无节点出入时期），可以抵御t个以下攻陷节点串通企图恢复多项式$F(x, y)$的攻击.

每个节点持有的子份额分别是，$F(x_j, y)=\sum_{i=1}^{r} f_i(x_j, y) \bmod p$，$F(x, x_j)=\sum_{i=1}^{r} f_i(x, x_j) \bmod p$，是2个不同阶数的单变量多项式，其中$x$的阶数为$t$，$y$的阶数为$n'$.因此$F(x, x_j)$可以抵御小于等于$n'$个攻陷节点企图恢复多项式$F(x, y)$的攻击，而$F(x_j, y)$可以抵御小于等于$t$个攻陷节点企图恢复多项式$F(x, y)$的攻击.因此，密钥生成阶段，通信系统的阈值由$x$，$y$中较低的阶数决定，即可以抵御小于等于$t$个攻陷节点的攻击.

2）节点出入时期，可以抵御n'以下个攻陷节点串通企图恢复多项式$F(x, y)$的攻击.

考虑最坏的情况下，节点变更阶段之前，攻陷节点的个数为t个，新加入节点中攻陷节点的个数为$n'-t$个，那么攻陷节点总共可以获得多项式$F(x_i, y)$的任意t个份额以及$F(x, x_j)$的任意$n'-t+t$个份额，由于多项式$F(x, y)$是非对称二元多项式，其中x的阶数为t，y的阶数为n'，那么刚好无法恢复其他任何诚实节点的份额.因此总共可以抵御n'个以下攻陷节点的串通攻击.

另外，发送在区块链上的多项式承诺$C_{F(x_j, y)}$是Kate承诺方案基于离散对数假设计算上的零知识，所以敌手也无法通过发送在区块链上的承诺获得任何信息.

同时，在3.3节的节点变更阶段中，当原有通过普通节点阈值$t \geqslant \frac{2}{n}$时，通过元节点获得份额，而元节点生成的$f_i(x, y)$的阶数与$F(x, y)$的阶数一样，因此也具有同样的性质. 证毕.

定理6. 可恢复性.任何时期，有节点丢失份额，都可以通过其他节点帮助恢复原有份额信息.

证明.普通节点（包括元节点）P_i丢失持有的份额$F(x_i, y)$以及$F(x, x_i)$，可通过其他普通节点P_j发送相应的份额信息$F(x_i, x_j)$以及$F(x_j, x_i)$插值计算

$$F(x_i, y)=\sum_{j=1}^{n'+1} F(x_i, x_j) \prod_{l=1, l \neq i}^{n'+1} \frac{y-x_l}{x_j-x_l} \bmod p,$$

$$F(x, x_i)=\sum_{j=1}^{t+1} F(x_j, x_i) \prod_{l=1, l \neq i}^{t+1} \frac{x-x_l}{x_j-x_l} \bmod p,$$

重新获得$F(x_i, y)$以及$F(x, x_i)$.

如果由于节点出入，导致普通节点数量不够，则可通过r个元节点P_j发送相应的子份额$f_j(x_i, y)$以及$f_j(x, x_i)$，并计算

$$F(x_i, y)=\sum_{j=1}^{r} f_j(x_i, y) \bmod p,$$

$$F(x, x_i)=\sum_{j=1}^{r} f_j(x, x_i) \bmod p,$$

重新获得$F(x_i, y)$以及$F(x, x_i)$. 证毕.

定理7. 保密性.组通信时期，只有组成员知道组密钥并且知晓所有组内成员身份，组外成员无法得知密钥以及组内成员的身份.

证明.分析内部攻击以及外部攻击：

1）内部攻击.由定理5可以知道密钥生成阶段每个节点持有的子份额可以抵御小于等于t个攻陷节点恢复秘密多项式$F(x, y)$的攻击.因此，任何不在组内的节点都无法恢复组内任何节点所持有的份额信息.由于组密钥是由发起者生成，通过与授权节点成对的共享密钥加密后传输，因此，未被授权的节点无法得知其不在组内的组密钥.而发起者传输的信息$C_i=\{K,(l_1, r_{l_1}),(l_2, r_{l_2}), \cdots,(l_m, r_{l_m}), R_H\}$，除了传输组密钥信息，还将组内成员的信息进行散列函数生成摘要$R_H=H(r_{l_1}, r_{l_2}, \cdots, r_{l_m}, K)$后上传至区块链，使得任何

组外节点都无法模仿组内成员在组中通信.

2)外部攻击.由于每个组密钥都是由发起者生成,并通过节点之间成对的通信密钥加密传输.同时将组成员信息加密后$R_H = H(r_{l_1}, r_{l_2}, \cdots, r_{l_m}, K)$上传至区块链,组内成员可以对组内的其他成员进行验证.因此,任何没有有效份额的外部节点,都无法得知其未被授权的组密钥K,也无法得知组内成员身份信息. 证毕.

定理 8. 节点变更阶段,如果原有普通节点阈值$t \leqslant \frac{n}{2}$,那么一定有至少$t+1$个诚实节点给新节点发送正确的信息$F\left(x_i, x_{l_j}\right)$,成功帮助新节点重建$F\left(x, x_{l_j}\right)$,否则只能通过元节点帮助重建份额.

证明.如果原有节点中的阈值$t < \frac{n}{2}$,也就是说原有节点中最多可以存在t个攻陷节点,那么原有节点中至少存在$n-t$个诚实节点帮助新加入的节点获得相应的份额.而当$t < \frac{n}{2}$时,那么满足$n-t > t$,故一定存在至少$t+1$个诚实节点可以成功帮助新节点重建$F\left(x, x_{l_j}\right)$.

如果原有节点中的阈值$t \geqslant \frac{n}{2}$,则不能够保证存在至少$t+1$个诚实节点可以成功帮助新节点恢复份额,因此需要通过元节点帮助恢复份额.由此可知,系统可容忍其为半诚实节点. 证毕.

5 性能评估

5.1 功能对比

表 2 将本文方案与其他文献方案从 5 个方面进行了功能对比.

Table 2 Functions Comparison of Schemes

表 2 方案的功能对比

方案	生成方式	去中心化	成对共享密钥	多组通信	阈值切换
文献 [28] 方案	一元多项式	否	否	否	否
文献 [29] 方案	一元多项式	否	是	是	否
文献 [30] 方案	一元多项式	否	否	是	否
文献 [31] 方案	一元多项式	否	否	是	是
文献 [32] 方案	对称二元多项式	是	是	是	否
本文方案	非对称二元多项式	是	是	是	是

1)从生成方式而言.目前多数方案都是基于单变量多项式来实现密钥协商的,文献 [28−30] 中要生成会话密钥都需要通过多个单变量多项式进行交互生成,文献 [31] 是通过单变量多项式的 Shamir 秘密共享来实现密钥分发.而本文方案每个节点持有并计算使用的都是确定了一个变量值的二元多项式.因此,在计算复杂度方面与其他一元多项式方案相近.

2)从去中心化而言.基于单变量多项式的方案都无法实现去中心化,由于需要通过单变量多项式进行有效的密钥协商,必须要可信第三方进行合理分配每个节点的份额.文献 [28−31] 中,需要可信第三方针对不同节点分配不同的份额信息来保证组成员能够获取相同的组密钥以及确保通信的保密性.但是一旦可信第三方被攻陷,系统中所有节点的份额信息以及密钥信息都会失效,导致整个通信系统的崩溃.而本文方案通过多个元节点共同生成二元多项式,实现去中心化,无需通过单个中心节点进行份额分配.因此,即使部分元节点被攻陷也能保证其他节点的密钥安全.

3)从成对共享密钥而言.节点之间成对的共享密钥能够增强节点协商组密钥的安全性,而基于单变量多项式的文献 [28−31] 中,节点之间不存在成对的共享密钥,只能通过可信第三方帮助生成共享会话密钥通信.而本文方案基于二元多项式,能够在节点之间高效生成成对的共享密钥进行通信,进而提高组密钥协商的安全性.

4)从多组通信而言.上述文献 [30] 需要把所有组的组员信息一次性提交给可信第三方,可信第三方才能一次生成多个组密钥发放给不同组的成员.而文献 [31] 只能通过可信第三方交互信息来获取组密钥,并且 1 轮通信只能获取单个组的组密钥.本文方案中,节点得到自己的份额,可随时通过成对的共享密钥进行安全有效的组密钥协商,自由发起多组通信.

5)从阈值切换而言.阈值的切换对于基于多项式的密钥协商方案而言是至关重要的,可以在节点出入阶段切换至不同的阈值,抵御节点出入对通信系统的威胁.从表 2 方案中,只有文献 [31] 以及本文方案可以实现在节点出入期间阈值的切换,其中本文方案基于非对称二元多项式,2 个变量可具备 2 个不同的阈值,因此在切换时具有更高的效率.

5.2 安全性对比

如表 3 所示,选出具有代表性的一元多项式文献 [31] 以及对称二元多项式文献 [32] 进行了安全性比较.

1)从可防御性而言.对于文献 [31],需要通过一个可信中心 KGC 来分发密钥,整个通信系统的安全完全依赖于一个中心节点是不安全的,敌手只需攻破一个中心节点即可攻破整个通信系统.文献 [32] 是通过多个元节点来生成一个二元多项式,有效克服了单个中心节点对系统的威胁,但是由于对称多项

Table 3 Comparison of Scheme in Safety

表 3 方案的安全性比较

方案	可防御性（阈值）	可恢复性	保密性
文献 [31] 方案	依赖中心节点	依赖中心节点	组密钥可证明安全性 会话密钥可证明安全性
文献 [32] 方案	t	任意t个节点	组密钥可证明安全性 会话密钥可证明无条件安全性
本文方案	稳定时期：t 节点出入时期：n'	最多任意 n'个节点	组密钥可证明安全性 会话密钥可证明无条件安全性

式本身的属性，即 2 个变量的阶数相同，因此在节点出入期间，如果再有恶意节点加入系统，则会导致超过阈值t的攻陷节点，那么系统的安全性就不能够得到保证.本文方案基于非对称二元多项式，即 2 个变量的阶数不同，代表 2 个不同的阈值，可以实现阈值切换，在稳定期间阈值为t，在节点出入时期，阈值提高为n’，其中$n' > t$.因此本文方案相比其他方案具有更强的可防御性.

2）从可恢复性而言.对于文献 [31]，如果某个节点密钥丢失，那么只能通过向可信中心进行重新申请，因此需要要求中心节点不可下线.而本文方案与文献 [32] 方案一样，都是可以通过系统中足够的任意普通节点发送相关的份额来恢复丢失的密钥，因此可有效抵御由于部分节点下线导致无法恢复密钥问题的威胁.本文方案由于持有的份额$F(x_j, y)$与$F(x, x_j)$阶数不同，因此可能需要更多的普通节点来恢复密钥.由于初始时设有元节点，因此也可以通过所有的元节点来恢复丢失的密钥.

3）从保密性而言.文献 [31] 方案虽然使用了 Shamir 秘密共享，但是在秘密重建阶段所使用的份额依旧是基于计算困难问题假设的，因此其组密钥以及会话密钥都是可证明安全的.而本文方案与文献 [32] 方案构建的会话密钥是根据初始的分发份额产生的，只有节点被攻陷达到阈值数量时才能攻破密钥，因此每个节点的会话密钥是无条件安全的.

5.3 效率分析

如表 4 所示，将本文方案与文献 [31−32] 方案在计算开销方面进行了比较.表 4 中T_{SC}表示执行对称加密操作所需的时间，T_{f^l}表示计算一个阶数为l的多项式会话密钥所需要的时间，T_{LP^l} 表示一次通过l个点进行插值计算所需要的时间，T_M表示执行一次椭圆曲线上的倍点运算所需要的时间，T_{gh}表示执行一次散列函数所需要的时间.T_{LP^t}表示一次通过 t 个点进行插值计算所需要的时间，T_{LP^n} 表示一次通过 n 个点进行插值计算所需要的时间，$T_{f^{n'}}$表示计算一个阶数为 n'的多项式会话密钥所需要的时间，T_{f^n}表示计算一个阶数为 n 的多项式会话密钥所需要的时间.由表 4 可知，安全性不如本文方案的文献 [32] 方案在组密钥协商阶段计算开销与本文方案相近.而文献 [31] 方案则在组密钥协商阶段的计算开销远远大于本文方案.由于涉及到不同阶数的多项式插值计算，无法直观地了解方案之间的效率关系，接下来将通过实验分析来证明.

Table 4 Comparison of Computational Cost

表 4 计算开销比较

方案	组密钥协商阶段	节点变更阶段
文献 [31] 方案	$(4n+4)T_{gh}+nT_{f^n}+2T_{LP^n}$	$5T_M+2T_{gh}$
文献 [32] 方案	$(n+1)(T_{f^t}+T_{SC})+2T_{gh}$	$tT_{f^t}+T_{LP^t}$
本文方案	$(n+1)\left(T_{f^{n'}}+T_{SC}\right)+2T_{gh}$	$(t+n')T_{f^{t'}}+T_{LP^{t'}}+T_{LP^{n'}}$

注：$t < n' \leqslant n$.

5.4 性能分析

本文使用 codahale 库在装有 Intel Core i7-6700K 4.00 GHz 处理器和 8GB 内存的 Windows 系统上进行仿真实验.

在通信开销方面，文献 [31] 方案如果需要与其他节点构建会话密钥，那么需要预先协商并存储与不同节点的会话密钥，这需要消耗非常大的存储开销.而本文方案以及文献 [32] 方案都只需要存储多项式的系数，利用秘密共享来生成会话密钥，本文方案和文献 [31−32] 方案相比，文献 [31] 方案有着明显的通信开销方面的优势.在计算开销方面，通过仿真实验，如图 4 和图 5 所示，将本文方案与文献 [31] 方案以及文献 [32] 方案分别在组密钥协商阶段以及节点变更阶段中密钥生成所消耗的时间进行了对比.设定文献 [32] 方案以及本文方案所需要的阈值$t = n/2$, $n' = 3n/4$, 例如$n = 100$, $t = 50$, $n' = 75$.由图 4 可知，本文方案在组密钥协商阶段的效率与文献 [32] 方案基本持平，并且远高于文献 [31] 方案.由图 5 可知，本文方案在节点变更阶段密钥生成的效率低于文献 [32] 方案，但仍然高于文献 [31] 方案.

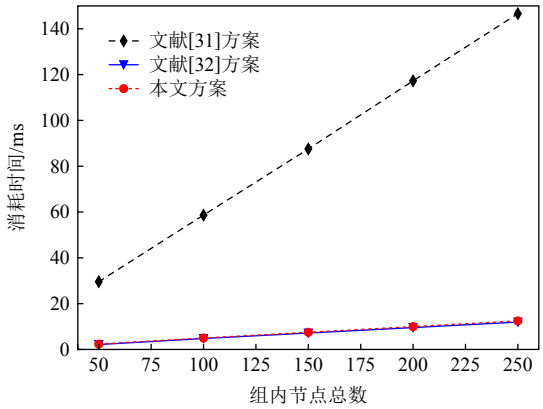

Fig. 4 Time comsumption comparison in group key agreement phase

图 4 组密钥协商阶段所消耗的时间对比

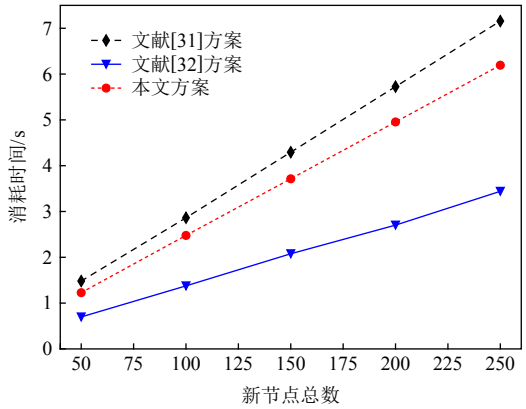

Fig. 5 Time comsumption comparison of key generation in node alternation phase

图 5 节点变更阶段密钥生成所消耗的时间对比

由于本文方案比文献 [32] 方案具有更高的安全性, 涉及到了更高阶数的插值运算, 因此效率上略低于文献 [32] 方案, 但安全性与效率上均高于文献 [31] 方案, 因此本文方案在工业互联网动态节点密钥管理场景中具备明显的优势.

6 结 语

从工业互联网的安全可靠的信息传输出发, 确保节点之间能够安全有效地通信.本文结合区块链技术以及秘密共享技术, 提出一种基于区块链的工业互联网动态密钥管理方案.基于非对称二元多项式生成密钥, 通过阈值切换, 有效抵御节点出入对通信系统的威胁, 具有较低的成本开销.此外, 通过区块链以及 Kate 承诺实现安全的点对点通信以及多组通信. 最后, 通过具体的安全性分析表明, 本文提出的基于区块链的工业互联网动态密钥管理方案是安全的, 并且通过仿真实验和对比分析表明, 本文方案是高效的.接下来, 我们将对如何实现更高效率的安全密钥管理协议作进一步研究.

作者贡献声明: 张泽林提出算法思路、完成实验并撰写论文; 王化群指导研究方案、设计并修改论文.

参 考 文 献

[1] Zhang Yinghui, Deng R H, Zheng Dong, et al. Efficient and robust certificateless signature for data crowdsensing in cloud-assisted industrial IoT[J]. IEEE Transactions on Industrial Informatics, 2019, 15(9): 5099−5108

[2] Yousefpoor M S, Barati H. Dynamic key management algorithms in wireless sensor networks: A survey[J]. Computer Communications, 2019, 134: 52−69

[3] Messai M L, Seba H. A survey of key management schemes in multi-phase wireless sensor networks[J]. Computer Networks, 2016, 105: 60−74

[4] Sun Yan, Liu K R. Hierarchical group access control for secure multicast communications[J]. IEEE/ACM Transactions on Networking, 2007, 15(6): 1514−1526

[5] Je D H, Lee J S, Park Y, et al. Computation-and-storage-efficient key tree management protocol for secure multicast communications[J]. Computer Communications, 2010, 33(2): 136−148

[6] Feng Li, Deng Guoqing, Yu Bin. Multi-level key management scheme for multi-level removable storage devices[J]. Journal of Information Security Research, 2018, 4(4): 329−335 (in Chinese)
(冯力, 邓国庆, 郁滨. 一种多密级移动存储设备多级密钥管理方案[J]. 信息安全研究, 2018, 4(4): 329−335)

[7] Gao Ronghai, Zeng Jiwen, Deng Lunzhi. Efficient certificateless anonymous multi-receiver encryption scheme without bilinear parings[J]. Mathematical Problems in Engineering, 2018, 2018(9): 1−13

[8] Lin J, Huang K, Lai Feipei, et al. Secure and efficient group key management with shared key derivation[J]. Computer Standards & Interfaces, 2009, 31(1): 192−208

[9] Lou Junjun, Zhang Qichao, Qi Zhuyun, et al. A blockchain-based key management scheme for named data networking[C] //Proc of the 1st IEEE Int Conf on Hot Information-Centric Networking (HotICN). Piscataway, NJ : IEEE, 2018, 141–146

[10] Zhao Huawei, Bai Peidong, Peng Yun, et al. Efficient key management scheme for health blockchain[J]. CAAI Transactions on Intelligence Technology, 2018, 3(2): 114−118

[11] Vijayakumar P, Chang V, Deborah L J, et al. Key management and key distribution for secure group communication in mobile and cloud network[J]. Future Generations Computer Systems, 2018, 84: 123−125

[12] Lei Ao, Ogah C, Asuquo P, et al. A secure key management scheme for heterogeneous secure vehicular communication systems[J]. ZTE

Communications, 2016, 14(3): 21−31

[13] Yu Yao, Liu Shumei, Guo Lei, et al. CrowdR-FBC: A distributed fog-blockchains for mobile crowdsourcing reputation management[J]. IEEE Internet of Things Journal, 2020, 7(9): 8722−8735

[14] Wu Bo, Xu Ke, Li Qi, et al. Toward blockchain powered trusted collaborative services for edge-centric networks[J]. IEEE Network, 2020, 34(2): 30−36

[15] Jin Tao, Zhang Liwan, Zhang Chen, et al. Research and design of cloud edge collaboration based on blockchain[J]. Journal of Information Security Research, 2021, 7(4): 310−318 (in Chinese)
(金韬, 庄丽婉, 张晨, 等. 基于区块链的云边协同系统研究与设计[J]. 信息安全研究, 2021, 7(4): 310−318)

[16] Xie Yong, Wu Libing, Zhang Yubo, et al. Anonymous mutual authentication and keyagreement protocol in multiserver architecture for VANETs[J]. Journal of Computer Research and Development, 2016, 53(10): 2323−2333 (in Chinese)
(谢永, 吴黎兵, 张宇波, 等. 面向车联网的多服务器架构的匿名双向认证与密钥协商协议[J]. 计算机研究与发展, 2016, 53(10): 2323−2333)

[17] Yu Yao, Li Fuliang, Liu Shumei, et al. Reliable fog-based crowdsourcing: A temporal-spatial task allocation approach[J]. IEEE Internet of Things Journal, 2020, 7(5): 3968−3976

[18] Belotti M, Bozic N, Secci S. A vademecum on blockchain technologies: when, which, and how[J]. IEEE Communications Surveys and Tutorials, 2015, 21(4): 3796−3838

[19] Ning Zhaoling, Zhang Kaiyuan, Wang Xiaojie, et al. Intelligent edge computing in Internet of vehicles: A joint computation offloading and caching solution[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 22(4): 2212−2225

[20] Ning Zhaolong, Dong Peiran, Wang Xiaojie, et al. Mobile edge computing enabled 5G health monitoring for Internet of medical things: A decentralized game theoretic approach[J]. IEEE Journal on Selected Areas in Communications, 2021, 39(2): 463−478

[21] Yang Bo, Cao Xuelin, Li Xiangfang, et al. Mobile-edge-computing-based hierarchical machine learning tasks distribution for IoT[J]. IEEE Internet of Things Journal, 2020, 7(3): 2169−2180

[22] Pan Jianlin, Wang Jianyu, Hester A, et al. Edgechain: An edge-IoT framework and prototype based on blockchain and smart contracts[J]. IEEE Internet of Things Journal, 2019, 6(3): 4719−4732

[23] Liu Aodi, Du Xuehui, Wang Na, et al. Block chain technology and its research progress in the field of information security[J]. Journal of Software, 2018, 29(7): 2092−2115 (in Chinese)
(刘敖迪, 杜学绘, 王娜, 等. 区块链技术及其在信息安全领域的研究进展[J]. 软件学报, 2018, 29(7): 2092−2115)

[24] Asayag A, Cohen G, Grayevsky I, et al. Helix: A scalable and fair consensus algorithm[C] //Proc of the 26th IEEE Int Conf on Network Protocols (ICNP). Piscataway, NJ: IEEE, 2018: 863−885

[25] Cheng R, Zhang Fan, Kos J, et al. Ekiden: A platform for confidentiality preserving, trustworthy, and performant smart contracts[C] //Proc of the 4th IEEE European Symp on Security and Privacy (EuroS&P). Piscataway, NJ: IEEE, 2019: 185−200

[26] Kate A, Zaverucha G M, Goldberg I. Constant-size commitments to polynomials and their applications[C] //Proc of the 16th Int Conf on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2010: 177−194

[27] Shamir A. How to share a secret[J]. Communications of the ACM, 1979, 24(11): 612−613

[28] Cheng Qi, Hsu C, Xia Zhe, et al. Fast multivariate-polynomial-based membership authentication and key establishment for secure group communications in WSN[J]. IEEE Access, 2020, 8: 71833−71839

[29] Albakri A, Harn L. Non-interactive group key pre-distribution scheme(GKPS) for end-to-end routing in wireless sensor networks[J]. IEEE Access, 2019, 7: 31615−31623

[30] Ching-Fang H, Lein H, Bing Z. UMKESS: User-oriented multi-group key establishments using secret sharing[J]. Wireless Networks, 2020, 26(1): 421−430

[31] Wei Lu, Cui Jie, Xu Yan, et al. Secure and lightweight conditional privacy-preserving authentication for securing traffic emergency messages in VANETs[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 1681−1695

[32] Harn L, Hsu C, Li Bohan. Centralized group key establishment protocol without a mutually trusted third party[J]. Mobile Networks and Applications, 2018, 23(5): 1132−1140

Zhang Zelin, born in 1998. Master candidate. His main research interests include cryptography and information security.

张泽林, 1998 年生.硕士研究生.主要研究方向为密码学与信息安全.

Wang Huaqun, born in 1974. PhD, professor. His main research interests include applied cryptography, blockchain and cloud computing security.

王化群, 1974 年生.博士, 教授.主要研究方向为应用密码学、区块链、云计算安全.