

基于工业物联网的分层多设备认证密钥协商协议

吴毅¹, 冯涛¹, 柳春岩², 苏春华³

(1. 兰州理工大学计算机与通信学院, 甘肃 兰州 730050;

2. 兰州理工大学经济与管理学院, 甘肃 兰州 730050;

3. 日本会津大学计算机科学系, 福岛 会津若松 965-8580)

摘要: 安全的认证密钥协商协议对确保工业物联网中设备节点及管理者身份的合法性、数据的机密性至关重要。然而, 现有的认证密钥协商协议大多数只支持设备管理者与设备节点间一对一的密钥协商, 当设备管理者需要与多个设备节点进行认证密钥协商时, 此类一对一的认证密钥协商协议需要管理员重复与目标设备节点进行认证密钥协商操作, 从而消耗大量的计算和通信资源。针对上述问题, 提出了一种分层多设备认证密钥协商协议。首先, 该协议利用椭圆曲线密码学和中国剩余定理, 保证设备管理者能够一次性建立与多个设备节点间不同的会话密钥。其次, 该协议采用对称同态加密技术, 保证边缘节点在不知道秘密信息的前提下, 帮助通信双方计算会话密钥。最后, 基于随机谕言机模型和有色 Petri 网形式化方法对所提协议进行了安全性证明, 并借助 NS-3 网络仿真软件模拟其在大规模设备接入场景下的通信过程, 以评估其可扩展性。仿真结果表明, 所提协议在保障高安全性的同时, 面对大规模设备接入仍表现出良好的可扩展性与稳定性。

关键词: 工业物联网; 认证密钥协商协议; 形式化分析; 中国剩余定理; 有色 Petri 网

中图分类号: TP393

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.2025044

Hierarchical multi-device authentication and key agreement protocol for industrial IoT

WU Yi¹, FENG Tao¹, LIU Chunyan², SU Chunhua³

1. School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China

2. School of Economics and Management, Lanzhou University of Technology, Lanzhou 730050, China

3. Division of Computer Science, University of Aizu, Fukushima 965-8580, Japan

Abstract: A secure authentication key negotiation protocol was considered to be of vital importance for ensuring the legitimacy of device nodes and managers, as well as the confidentiality of data in the Industrial Internet of

收稿日期: 2025-01-09; 修回日期: 2025-05-16

通信作者: 冯涛, fengt@lut.edu.cn

基金项目: 国家自然科学基金 (62162039); 甘肃省重点研发计划 (24YFFA016)

Foundation Items: The National Natural Science Foundation of China (62162039), The Key R&D Program of of Gansu Province (24YFFA016)

引用格式: 吴毅, 冯涛, 柳春岩, 等. 基于工业物联网的分层多设备认证密钥协商协议[J]. 网络与信息安全学报, 2025, 11(4): 86-106.

Citation Format: WU Y, FENG T, LIU C Y, et al. Hierarchical multi-device authentication and key agreement protocol for industrial IoT[J]. Chinese Journal of Network and Information Security, 2025, 11(4): 86-106.

Things. However, most existing authentication key negotiation protocols were found to support only one-to-one key negotiation between device managers and device nodes. When a device manager needed to conduct authentication key negotiation with multiple device nodes, such one-to-one protocols required the administrator to repeatedly perform authentication and key negotiation operations with each target device node, thereby consuming substantial computing and communication resources. To overcome these challenges, a hierarchical multi-device authentication and key agreement protocol was proposed for industrial IoT environments. In the proposed protocol, the Chinese Remainder Theorem was employed to allow a device manager to establish multiple session keys simultaneously with different device nodes. In addition, symmetric homomorphic encryption was utilized to enable edge nodes to participate in session key computation without gaining access to the complete session key, thereby significantly reducing the risk of key exposure even if edge nodes were compromised. The security of the proposed protocol was formally analyzed using the random oracle model in conjunction with the Colored Petri Net method. Furthermore, the communication process of the protocol under large-scale device access scenarios was simulated using the NS-3 network simulation tool to assess its scalability. Simulation results demonstrate that the proposed protocol provides a high level of security and exhibits strong scalability and stability under large-scale device access conditions.

Keywords: industrial IoT, authentication and key agreement protocol, formal analysis, Chinese remainder theorem, colored Petri net

0 引言

近年来,随着智能制造、电气自动化等技术的飞速发展,工业控制网络正逐步从传统封闭式工业以太网向互联互通的工业物联网(industrial Internet of things, IIoT)转型^[1]。IIoT借助自动化与智能化管理,显著提高了生产效率,减少了人力成本及物料浪费,因此受到世界各国的高度关注。各国相继出台了相关的战略方针^[2-5]。例如,

美国于2017年发布《工业物联网战略框架》;德国发布了《德国工业4.0战略计划实施建议》;中国将IIoT纳入“中国制造2025”战略的重要组成部分;日本则积极推动“第四次工业革命”中的IIoT应用。

随着信息技术的持续发展与工业自动化的逐步推进,IIoT架构已从传统的中心化架构演进为图1所示的基于“云-边-端”的架构^[6]。这种

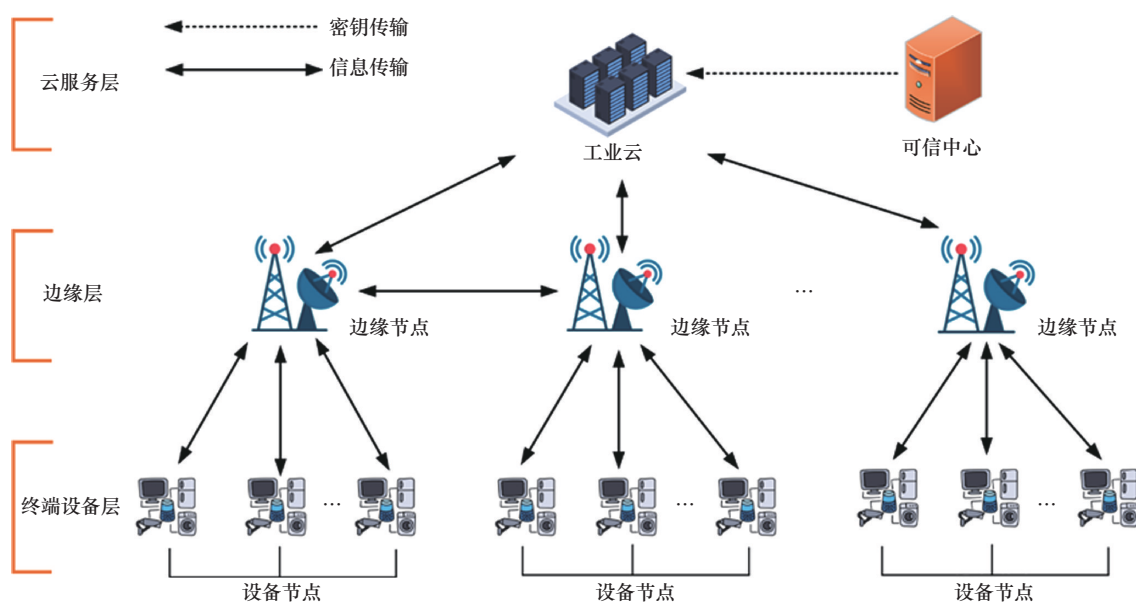


图1 基于“云-边-端”架构的IIoT

Figure 1 Industrial internet of things based on the cloud-edge-terminal architecture

“云-边-端”架构通过构建多层次协同关系，带来如下优势^[7-8]。

1) 云服务可以根据工厂需求动态分配资源，显著提高了IIoT系统的灵活性与可扩展性。

2) 云平台能够高效分析海量实时数据并支持智能决策，而边缘层处理数据可减少通信时延、实现快速响应，进而提升安全性与效率。

3) 基于“云-边-端”架构，数据可在适配的层级进行处理。终端设备层负责本地数据采集与初步处理，边缘层专注于操作指令的快速响应，云服务层则进行深度分析与全局优化。这种分工不仅优化了系统整体性能，也显著减少了网络负载。

然而，由于IIoT中传输的数据涉及工业生产、设备状态等敏感信息，数据的安全性与保密性问题逐渐成为关注焦点^[9-10]。因此，IIoT平台需要保障数据通信过程中数据的完整性、机密性与可靠性，有效防范未经授权的访问及数据篡改行为。

为保障IIoT中数据传输的安全性与机密性，部署安全的认证密钥协商（authentication and key agreement, AKA）协议是常用的解决方法之一。AKA作为一种保护通信数据安全的协议，可使两个或多个实体在不安全的网络环境中建立共享密钥，进而对后续通信进行加密。在IIoT环境下，安全的AKA协议能够有效弥补其存在的安全漏洞^[11-14]，具体体现在两个方面：一方面，IIoT环境中存在大量需要进行身份认证的设备与实体，借助AKA协议，能够确保通信双方身份的真实性；另一方面，IIoT中传输的数据往往包含敏感的商业信息与操作信息，为保护这些信息的机密性，AKA协议会采用强密码学算法来保证只有授权的设备和服务可获取密钥，以防止信息被未经授权人员窃取或篡改。

然而，随着IIoT环境中设备节点数量的持续增长，当设备管理者需要访问多台设备终端的数据时，传统的一对一AKA协议在通信与计算方面的开销显著攀升。具体而言，其主要面临三大挑战。①传统的AKA协议要求设备与管理者建立独立的通信通道，并执行单独的密钥协商。随着设备数量的增加，这种一对一模式在多设备认证场景下，需要管理员与每台设备重复进行认证操作，导致通信开销呈线性增长，进而造成网络

带宽过度占用和通信时延增加。②传统的AKA协议通常依赖复杂的加密算法，这些操作对设备的计算能力提出了较高的要求。然而，IIoT中大多数设备的资源有限，难以承受频繁的加密和解密操作带来的高额计算与能耗负担，从而导致认证效率大幅下降。③传统AKA协议依赖可信第三方服务器为设备和用户分别生成密钥和证书。面对IIoT环境中数量庞大且分布广泛的设备节点，这种中心化模式的系统可扩展性受限。综上所述，多设备认证在通信开销、计算开销、可扩展性和安全性等方面面临严峻挑战。为保障IIoT数据传输安全，本文提出一种新型的安全分层多设备AKA协议，其主要贡献如下。

1) 设计了一种基于IIoT的分层认证架构，该架构由密钥分发中心、边缘节点、设备管理者和设备节点4种实体构成。其中，密钥分发中心作为完全可信实体，主要负责边缘节点、设备管理者和设备节点的注册工作。针对工业环境中设备节点资源受限问题，该架构利用边缘节点辅助认证双方进行会话密钥协商，有效减少了通信双方的资源消耗。

2) 提出一种融合椭圆曲线密码学和中国剩余定理的多设备AKA协议。该协议不仅实现了设备管理者与多个设备节点一次性协商生成不同会话密钥，还通过引入CRT（Chinese remainder theorem）显著增强了设备节点撤销时的灵活性。

3) 在IIoT中，边缘节点被视为“诚实但好奇”的实体，其可能尝试通过计算获取设备管理者与设备节点间的会话密钥。为确保传输内容安全，本协议采用对称同态加密技术。即使边缘节点获取了加密数据，由于缺乏正确的解密密钥，也无法获取传输的具体内容。因此，即使遭受边缘节点被盗攻击，攻击者也无法从边缘节点获取通信双方协商的会话密钥。

4) 本研究采用随机谕言机（random oracle, RO）模型和有色Petri网（colored petri net, CPN）形式化分析方法对协议进行了安全性分析，结果表明该协议可实现身份认证与会话密钥协商，并能有效抵御多种攻击。为评估协议性能，本文结合PyPBC库与NS-3网络仿真工具，从密码计算开销和通信性能两个方面对协议进行综合评估。结果表明，本文所提协议在多设备密钥协商场景下

显著降低了通信开销、计算开销、能耗开销和存储开销，并在大规模设备接入环境中展现出良好的可扩展性与稳定性。

1 相关工作

为确保IIoT环境中数据传输的机密性和身份信息的隐私性，通信方之间需要执行认证密钥协商操作。针对工业环境中资源受限设备，Li等^[15]提出一种基于椭圆曲线密码学的认证协议。该协议在保证较高安全性的同时，显著减少了计算资源的消耗。针对无线传感网络的认证安全问题，Li等^[16]提出一种基于三因子的身份认证协议，旨在解决设备被盗攻击等安全问题。然而，Jiang等^[17]指出，该协议无法抵抗智能设备被盗攻击和临时密钥泄露攻击，同时缺乏前向安全性保障，且未采用标准攻击者模型对协议的安全性进行分析。Eldefrawy等^[18]提出一种基于对称密钥加密和哈希函数的轻量级密钥分发协议，用于实现安全通信。Patonico等^[19]提出一种基于身份的AKA协议，使传感器设备和中继节点能够在不泄露身份的情况下建立安全通信。但Lee等^[20]指出，该协议无法抵抗拒绝服务攻击、智能设备被盗攻击和重放攻击。Tanveer等^[21]针对无人机网络提出一种基于关联数据的认证加密（authenticated encryption with associated data, AEAD）机制和哈希函数的安全AKA协议，但此协议未能提供不可追溯性。Srinivas等^[22]提出一种新型3因素设备管理者认证协议，通过可信的云网关节点实现设备管理者与物联网设备之间的相互认证。然而，Tanveer等^[23]指出，该协议无法抵抗中间人攻击和会话密钥失窃攻击。Khan等^[24]设计了一种基于智能电网的匿名轻量级密钥协商框架PALK。然而，Chaudhry等^[25]证明PALK框架在登录和认证阶段存在设计缺陷。Xu等^[26]基于模糊提取器在用户、智能设备和控制节点之间建立了安全通道，但该协议不支持生物特征的更新。此外，Wang等^[27]提出一种多因子设备管理者身份认证模型，并利用非对称密码算法的不平衡性实现会话密钥的前向安全性。

上述协议在一对一设备认证场景下表现良好。然而，随着智能设备数量的增加，这些协议需要重复进行认证密钥协商操作，导致设备面临

巨大的计算与通信开销。因此，针对多设备认证需求，大量学者提出多设备AKA协议^[28]。Vinoth等^[29]基于中国剩余定理和对称加密提出一对多身份验证协议。但Wang等^[30]指出，此协议无法抵抗智能卡被盗攻击、重放攻击、中间人攻击，且缺乏前向安全性。Deebak等^[31]为解决医疗领域的隐私保护和身份验证挑战，结合访问控制、密钥共享和密码学技术，实现了一对多的身份验证。Cui等^[32]提出一种适用于车联网环境的协议，此协议能够实现一对多的认证及密钥协商。Ming等^[33]基于椭圆曲线密码学、中国剩余定理和哈希链提出一种一对多的AKA协议。但该协议假设网关节点是完全可信的，会话密钥是由网关节点生成的。一旦网关节点被攻破，攻击者可直接从中获取通信密钥，存在单点失效风险。Luo等^[34]提出一种基于混合区块链架构的多对多跨域认证方案，该方案允许多台设备同时与其他农业系统的多个数据服务提供商进行相互认证。

通过对现有研究的分析，发现现有基于IIoT的AKA协议存在4大问题。①现有的AKA协议大多数针对单一设备间的认证密钥交换进行设计，缺乏对多设备认证的有效支持。然而，在实际的IIoT中，往往需要对多个设备节点同时进行认证和密钥协商，导致现有协议面临高昂的计算开销与通信开销。②部分协议依赖高计算开销的密码学算法，不仅加重了系统的计算负担，还可能在资源有限的工业设备上引发性能瓶颈，影响系统的实时性与运行效率。③许多现有协议未能充分保障关键安全属性，如无法有效抵御拒绝服务攻击和智能设备被盗攻击、无法提供前向安全性。④大多数协议假设网关或边缘节点是完全可信的，并且可以安全地管理认证双方的会话密钥。然而，在实际的IIoT环境中，这些节点可能成为潜在的攻击目标，攻击者可能通过物理访问或软件漏洞获取敏感信息，从而威胁整个网络的安全。

2 预备知识

2.1 中国剩余定理

中国剩余定理^[35]是数论中的一个重要定理，用于求解一组同余方程。具体而言，设 $m_1, m_2, \dots, m_n$ 是 n 个两两互质的正整数，记 $M =$

$m_1, m_2, \dots, m_n$, 并定义 $M_i = \frac{M}{m_i}$ 。对于给定的同余方程组:

$$\begin{cases} x \equiv a_1 \pmod{m_1} \\ x \equiv a_2 \pmod{m_2} \\ \vdots \\ x \equiv a_n \pmod{m_n} \end{cases} \quad (1)$$

该方程组在模 M 下有唯一解, 其形式为:

$$x = \sum_{i=1}^n a_i M_i N_i \pmod{N} \quad (2)$$

其中, N_i 为 M_i 在模 m_i 下的乘法逆元, 即满足 $M_i N_i \equiv 1 \pmod{m_i}$ 的整数。

相较传统的密钥协商协议, 基于中国剩余定理的 AKA 协议通过将密钥分解为多个独立模数, 规避了传统协议中授权中心集中分发密钥导致的单点故障风险。该协议增强了系统的容错能力 (即使部分认证设备遭受攻击或发生故障, 系统仍能稳定运行), 从而显著提升了在复杂工业环境中的可用性、灵活性和鲁棒性。

2.2 椭圆曲线密码学

椭圆曲线密码学规定, 在有限域上的椭圆曲线循环子群中, 一个椭圆曲线密码通常由一个六元组 (q, a, b, G, n, h) 描述。其中, q 表示有限域的大小, a 和 b 表示椭圆曲线方程 $y^2 = x^3 + ax + b$ 的参数, G 表示循环子群的生成元, n 表示子群的阶数, h 表示协因子。私钥是一个随机数 $r \in \mathbb{Z}_q^*$, 公钥则由 $P = rG$ 计算得到。基于椭圆曲线离散对数问题的计算不可行性 (即从 P 和 G 推导 r 在计算上是不可行的), 椭圆曲线密码学能够有效保障私钥的安全。

在 IIoT 环境中, 由于设备节点普遍存在资源受限的情况, 采用轻量级加密算法对传输消息进行保护显得尤为重要。与传统的 RSA (Rivest-Shamir-Adleman) 加密相比, 256 位椭圆曲线密钥能够提供与 3 072 位 RSA 密钥等同的安全强度。这一特点使椭圆曲线密码学在计算效率方面具有显著优势, 同时可以有效减少设备的资源消耗。此外, 由于椭圆曲线密码学的密钥较短, 从而可以减少设备的存储资源消耗。

2.3 对称同态加密

现有的 AKA 协议通常采用对称同态加密算法^[36]或基于格密码的同态加密算法。在基于格

密码的同态加密算法中, BGV (Brakerski-Gentry-Vaikuntanathan)^[37]、BFV (Brakerski-Fan-Vercauteren)^[38]和 CKKS (Cheon-Kim-Kim-Song)^[39]是 3 种具有代表性的算法。这些算法均基于环学习同余问题的计算难解性, 涉及高维多项式运算, 并具备抗量子攻击的特性。本文利用 Microsoft SEAL 库对称同态加密算法, 以及 BGV、BFV、CKKS 这 3 种同态加密算法的计算时间开销进行了系统的对比分析, 实验结果如图 2 所示。

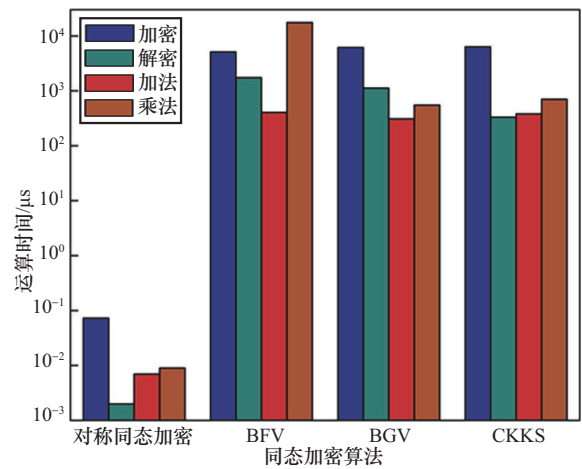


图 2 不同同态加密算法的计算时间开销对比

Figure 2 Comparison of computational time costs among different homomorphic encryption algorithms

考虑到 IIoT 环境中的终端设备普遍存在资源受限的特性, 难以承受较高的计算开销, 同时该应用场景对数据传输实时性有着严格要求, 而基于格密码的同态加密算法由于其固有的计算复杂性, 存在计算开销较大、耗时较长的问题, 难以满足实时交互需求。因此, 本文最终选择计算效率更高的对称同态加密算法作为底层加密算法。

对称同态加密算法主要由密钥生成、加密和解密 3 种算法组成。

1) 密钥生成算法。给定安全参数 (k_0, k_1, k_2) , 满足 $k_1 \ll k_2 < (k_0/2)$ 。生成私钥 $SK = (p, q, L)$ 。其中, p 和 q 为两个大素数, 满足 $|p| = |q| = k_0$; L 表示随机数, 其位长为 $|L| = k_2$ 。计算 $N = pq$, 并设置公共参数 $PP = (k_0, k_1, k_2, N)$ 。定义消息空间 M 为 $\{0, 1\}^{k_1}$ 。

2) 加密算法: 对于明文消息 $m \in M$, 使用私

钥 $SK = (p, q, L)$ 进行加密, 生成密文 c 。其形式化表示如下:

$$c = \text{HEnc}(m) = (rL + m)(1 + r'p) \bmod N \quad (3)$$

其中, $r \in \{0, 1\}^{k_2}$ 和 $r' \in \{0, 1\}^{k_0}$ 为随机数。

3) 解密算法: 使用私钥 $SK = (p, q, L)$ 解密密文 c , 得到消息。其形式化表示如下:

$$\begin{aligned} \text{HDec}(c) &= (c \bmod p) \bmod L = \\ &(rL + m) \bmod L = m \end{aligned} \quad (4)$$

在 IIoT 环境中, 网关和边缘节点设备负责工业数据的传输与处理, 但这些设备存在被滥用或恶意控制的风险。对称同态加密算法在该场景下具有 4 个关键优势: ①支持加密数据直接计算, 降低了对中间节点的信任依赖; ②即使部分设备遭受攻击或密钥泄露, 数据的安全性仍可得到有效保障; ③相比基于格密码的同态加密算法, 对称同态加密算法的计算开销更低, 适用于资源受限的 IIoT 设备; ④简化密钥管理 (仅需管理对称密钥), 减少了复杂公钥基础设施的部署成本。

3 系统模型与安全目标

3.1 系统架构

本文提出的基于 IIoT 的分层多设备 AKA 协议由密钥分发中心 (KDC)、边缘节点 (EN)、设备节点 (DN) 和设备管理员 (DM) 4 种实体构成, 系统架构如图 3 所示。

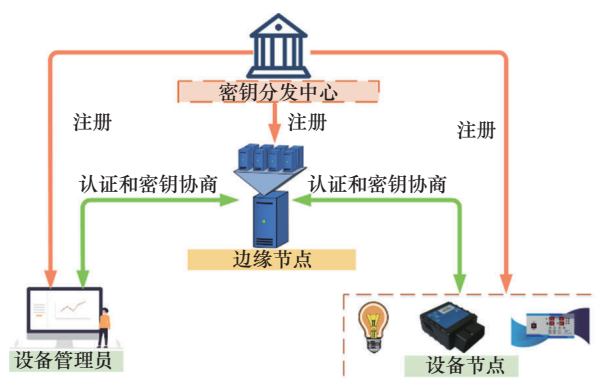


图3 系统架构

Figure 3 System architecture

1) 密钥分发中心: 该系统将“云-边-端”架构中的工业云与可信中心整合为一个完全可信的密钥分发中心, 主要负责设备管理者、边缘节点和设备节点的注册管理与密钥分发。

2) 边缘节点: 边缘节点负责管理辖区内的所有设备节点, 并协助计算设备管理者与设备节点间的会话密钥, 执行加密数据的聚合计算。

3) 设备节点: 在 IIoT 中, 设备节点负责实时监测生产状态并采集数据, 支持设备管理者远程实时获取监测信息。

4) 设备管理员: 设备管理员借助智能卡进行身份验证, 并通过边缘节点访问终端设备。

本文所提协议融合了椭圆曲线密码学、中国剩余定理和对称同态加密 3 种密码学技术, 以满足 IIoT 中设备间传输数据的安全需求。首先, 考虑到 IIoT 中的设备通常具有计算能力和能量受限的特点, 本协议选用椭圆曲线密码学作为基础加密技术。椭圆曲线密码学凭借其密钥尺寸小、计算开销低的优势, 能够显著减轻设备的计算负担。在该协议中, 椭圆曲线密码学主要用于通信实体的注册, 以及认证与密钥协商阶段的实体身份认证。其次, 基于中国剩余定理, 设备管理员通过一次计算即可生成与多个设备节点对应的会话密钥, 从而提升密钥分发的效率。最后, 设备节点采用对称同态加密技术对随机数进行加密, 并将生成的密文发送至边缘节点进行聚合。边缘节点能够在不解密的情况下直接对密文执行计算操作, 并生成聚合结果。这一机制确保了边缘节点在聚合过程中无法获取秘密信息的具体内容, 从而有效防止会话密钥的泄露。

3.2 威胁模型

本文所提协议中的敌手 $\mathcal{A}$ 拥有与 Dolev-Yao 攻击者模型^[40]和 CK 敌手模型^[41]中的敌手相同的能力。在 Dolev-Yao 攻击者模型中, 设备管理者和设备节点通常被视为不可信实体。敌手 $\mathcal{A}$ 可以窃听、拦截、修改和删除任意两个实体之间传输的消息, 伪造消息以欺骗合法实体, 冒充通信双方发送和接收消息。在该协议中, 边缘节点的核心功能是协助设备管理者和设备节点计算会话密钥, 因此协议必须确保边缘节点无法获取设备管理者和设备节点之间的会话密钥, 以保障数据的机密性和隐私性。在 CK 敌手模型中, 攻击者不仅拥有 Dolev-Yao 模型中敌手的全部能力, 还额外拥有 3 种关键能力。①会话状态泄露: 敌手 $\mathcal{A}$ 可以获取目标会话中除长期密钥外的所有内部状

态信息。②会话密钥查询：敌手 $\mathcal{A}$ 可以获取特定会话的会话密钥。③参与方破坏：敌手 $\mathcal{A}$ 可以获取被攻击参与方内部存储的所有信息。

3.3 安全目标

本文所提协议需要满足以下 4 项核心安全目标。

1) 双向认证：本文所提协议须确保任意两个通信实体在交换敏感信息前，能够通过认证机制验证彼此的身份，防止假冒攻击。

2) 匿名性与不可链接性：首先，本文所提协议须确保认证双方的身份信息得到保护，攻击者无法直接获取真实身份；其次，攻击者即使截获通信中的匿名身份信息，也无法通过分析这些信息将不同的通信会话关联到同一设备或用户，防止行为追踪。

3) 前向安全性：即使当前会话密钥泄露，攻击者也无法利用该密钥推算出历史会话密钥，从而有效降低长期密钥泄露带来的安全风险。

4) 抗边缘节点被攻击：若边缘节点被攻击者控制，攻击者无法从设备中获取到足够的信息来推算出会话密钥。

3.4 安全模型

在 RO 模型中，分别用 $\prod_{DM_i}^{S_1}$ 、 $\prod_{EN_i}^{S_2}$ 和 $\prod_{DN_j}^{S_3}$ 表示 DM_i 、 EN_i 和 DN_j 的第 S_1 、 S_2 和 S_3 个实例。这些实例可以被视为谕言机。在该模型中，敌手 $\mathcal{A}$ 能够获得系统参数和公开信道上传输的消息，并可以通过修改或者伪造消息欺骗实例。敌手 $\mathcal{A}$ 被允许访问以下谕言机，潜在地控制所有通信。

1) Hash (Type, M) 谕言机。当敌手 $\mathcal{A}$ 询问 Hash (Type, M) 谕言机时，该谕言机返回哈希列表中对应的哈希值。如果该列表中不存在对应的询问值，那么谕言机生成一个随机数发送给敌手 $\mathcal{A}$ ，并在列表中记录该键值对。

2) Execute ($\prod_{DM_i}^{S_1}, \prod_{EN_i}^{S_2}, \prod_{DN_j}^{S_3}$) 谕言机。该查询模拟协议遭受被动攻击。敌手 $\mathcal{A}$ 可以监听 $\prod_{DM_i}^{S_1}$ 、 $\prod_{EN_i}^{S_2}$ 和 $\prod_{DN_j}^{S_3}$ 之间在公开信道上传输的信息，但无法对消息内容进行篡改。

3) Send ($\prod_{DM_i}^{S_1} / \prod_{EN_i}^{S_2} / \prod_{DN_j}^{S_3}, M$) 谕言机。该查询模拟敌手 $\mathcal{A}$ 发起主动攻击。当敌手 $\mathcal{A}$ 给某实

体发送消息 M 后，实体将按照协议规则返回相应的执行结果。

4) Reveal ($\prod_{DM_i}^{S_1}, \prod_{DN_j}^{S_3}$) 谕言机。该查询模拟实体间会话密钥泄露。当敌手 $\mathcal{A}$ 发起询问时，谕言机返回 $\prod_{DM_i}^{S_1}$ 和 $\prod_{DN_j}^{S_3}$ 之间建立的会话密钥。

5) Corrupt ($\prod_{DM_i}^{S_1}, a$) 谕言机。该查询模拟 DM_i 私钥丢失场景。假设敌手 $\mathcal{A}$ 仅能获取用户的 3 个认证因子中的任意 2 个，其可以通过 3 种方式获取 DM_i 的秘密信息。① $f = 0$ ：将 DM_i 的口令 PW_i 和生物特征 Bio_i 返回给敌手 $\mathcal{A}$ 。② $f = 1$ ：将 DM_i 存储在智能卡中的数据和口令 PW_i 返回给敌手 $\mathcal{A}$ 。③ $f = 2$ ：将 DM_i 存储在智能卡中的数据和生物特征 Bio_i 返回给敌手 $\mathcal{A}$ 。

6) Test ($\prod_{DM_i}^{S_1}, \prod_{DN_j}^{S_3}$) 谕言机。当敌手 $\mathcal{A}$ 发起询问时，谕言机返回一个字符串，用于检验敌手能否根据其掌握的知识辨别该字符串是真实会话密钥还是随机选取的字符串。需要注意的是：若敌手 $\mathcal{A}$ 询问过 Reveal 或 Corrupt 谕言机，则不可再查询 Test 谕言机；而且，敌手对 Test 谕言机仅能发起一次查询。

会话密钥的语义安全性：在 RO 模型中，敌手 $\mathcal{A}$ 试图区分 Test 谕言机输出的是真实会话密钥还是随机字符串。敌手 $\mathcal{A}$ 可对 Test 谕言机发起询问，获取一个字符串。当询问结束时，敌手 $\mathcal{A}$ 猜测比特 b' ，当且仅当 $b' = b$ ，敌手 $\mathcal{A}$ 赢得游戏。敌手 $\mathcal{A}$ 攻破协议会话密钥的语义安全性优势为 $\text{Adv}(\mathcal{A}) = |2\Pr[b' = b] - 1|$ 。若对于任意概率多项式时间内的敌手 $\mathcal{A}$ 而言，优势 $\text{Adv}(\mathcal{A})$ 是可忽略的，则称所提协议的会话密钥满足语义安全性。

椭圆曲线上的计算性 Diffie-Hellman (elliptic curve computational Diffie-Hellman, ECCDH) 问题：已知 $P, aP, bP \in \mathbb{G}$, $a, b \in \mathbb{Z}_p^*$ ，求解 $abP \in \mathbb{G}$ 是困难的。

4 本文协议

本文提出的协议主要包含系统初始化、注册、登录和认证密钥协商 4 个阶段。系统初始化阶段的主要任务是生成公共参数和密钥分发中心的公私钥对。注册阶段进一步细分为边缘节点注

册、设备节点注册和设备管理员注册3个子阶段，分别完成不同实体的注册流程。登录阶段的核心功能是通过智能卡实现对设备管理员的身份认证。认证密钥协商阶段的主要目标是完成设备管理员与多个设备节点之间的双向认证，并生成双方共享的会话密钥。

本文使用的符号及其定义如表1所示。

表1 符号及其定义
Table 1 Symbols and their definitions

符号	定义
sk_x	实体 x 的私钥
PK_x	实体 x 的公钥
ID_x	实体 x 的身份信息
r_x	随机数
PID_x	设备节点的设备名
$a_x b_x$	设备节点的秘密信息
PW_x	设备管理员的口令
Bio_x	设备管理员的生物特征
Counter	计数器
K_1	同态加密密钥
K_{DN_j}	设备 j 的部分会话密钥
T_i	时间戳
$Enc(), Dec()$	对称加密, 对称解密
$HEnc(), HDec()$	同态加密, 同态解密
$H_0()$	哈希函数
$\oplus, $	异或操作, 连接符

4.1 系统初始化阶段

KDC选取椭圆曲线参数 a 和 b ，定义一个阶为素数 p 的非奇异椭圆曲线 E ， P 为群 G 的生成元。KDC随机选择一个主密钥 $sk_{KDC} \in \mathbb{Z}_p^*$ ，计算公钥 $PK_{KDC} = sk_{KDC} \cdot P$ ，定义哈希函数 $H_0: \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ ，最后公开系统参数 $\{G, P, H_0, PK_{KDC}\}$ 。

4.2 注册阶段

(1) 边缘节点注册

1) EN_i 选取随机数 $ID_{EN_i} \in \mathbb{Z}_p^*$ 作为其身份信息，通过安全信道将 ID_{EN_i} 发送至KDC。

2) KDC接收到 EN_i 发送的身份信息后，选择随机数 $r_{EN} \in \mathbb{Z}_p^*$ ，计算 EN_i 的私钥 $sk_{EN_i} = H_0(r_{EN} || sk_{KDC})$ 和公钥 $PK_{EN_i} = sk_{EN_i} \cdot P$ 。KDC通

过安全信道将 $\{sk_{EN_i}\}$ 发送给 EN_i ，并公开 PK_{EN_i} ，用于其他设备对 EN_i 身份的认证。

(2) 设备节点注册

1) DN_j 通过安全信道将注册请求发送给KDC。

2) KDC选择 N 个互素的素数作为设备名（设备节点的唯一标识） $PID_{DN_j} \in \mathbb{Z}_p, j \in [1, N]$ ，计算 $SP = \prod_{j=1}^N PID_{DN_j}$ 、 $a_j = SP/PID_{DN_j}$ 和 $a_j b_j \equiv 1 \pmod{PID_{DN_j}}$ 。KDC为 DN_j 选取对应的设备名，并计算设备节点的私钥 $sk_{DN_j} = H_0\{PID_{DN_j} || sk_{KDC}\}$ 和公钥 $PK_{DN_j} = sk_{DN_j} \cdot P$ 。KDC根据 DN_j 的地理位置信息，将其划分至距离最近的边缘节点所管理的区域，以优化计算资源和通信效率。在此过程中，KDC使用 $List_{EN_i}$ 记录 PID_{DN_j} 与 PK_{DN_j} 之间的关系，并选取 $K_{EN_i} \in \mathbb{Z}_p^*$ 作为该域的初始域密钥。最后，KDC通过安全信道将 $\{PID_{DN_j}, a_j b_j, sk_{DN_j}, K_{EN_i}\}$ 发送给相应的 DN_j ，将 $\{List_{EN_i}, K_{EN_i}\}$ 发送给相应的 EN_i 。

(3) 设备管理员注册

1) 设备管理员 DM_i 选取随机数 $r_{DM_i} \in \mathbb{Z}_p^*$ ，结合身份信息 ID_{DM_i} 和口令 PW_{DM_i} 计算 $I = H_0(PW_{DM_i} || r_{DM_i})$ ，生成注册请求。 DM_i 通过安全信道将 $\{ID_{DM_i}, I\}$ 发送至KDC。

2) KDC接收到 DM_i 发送的注册请求后，为 DM_i 分配管理权限对应的域（即指定 DM_i 可管理的边缘节点区域），并计算 $W = H_0(ID_{DM_i} || sk_{EN_i}) \oplus I$ 。KDC将 $\{W, Gen(\cdot), Rep(\cdot)\}$ 存储在智能卡中，随后将智能卡和 ID_{EN_i} 通过安全信道发送给 DM_i 。

3) DM_i 接收到智能卡后，输入自身的生物特征 Bio_{DM_i} 。系统据此计算 $Gen(Bio_{DM_i}) = (\delta_{DM_i}, \theta_{DM_i})$ 、 $E = H_0(\delta_{DM_i}) \oplus r_{DM_i}$ 和 $V = H_0(ID_{DM_i} || PW_{DM_i} || r_{DM_i})$ ，并生成计数器Counter。Counter用于记录连续认证失败的次数，若超过预设阈值 n ，智能卡将判定当前操作为非法行为，并终止认证流程。最后， DM_i 将 $\{E, V, W, Counter\}$ 存入智能卡中。

4.3 登录及目标设备节点选择阶段

1) DM_i 在智能卡中输入身份参数 ID_{DM_i} 、

PW_{DM_i} 和 Bio_{DM_i}

2) 智能卡计算 $\delta_{DM_i} = \text{Rep}(\text{Bio}'_{DM_i}, \theta'_{DM_i})$, $r_{DM_i} = E \oplus H_0(\delta'_{DM_i})$ 和 $V' = H_0(\text{ID}_{DM_i} \| PW_{DM_i} \| r_{DM_i})$, 并验证身份认证等式 $V' = V$ 是否成立。若等式不成立, 则 DM_i 身份认证失败, 计数器数值增加 (Counter = Counter + 1)。若等式成立, 则 DM_i 根据同态加密密钥生成算法 $HGen()$, 生成同态加密密钥 K_1 , 并选取随机数 $r_{SC} \in \mathbb{Z}_p^*$ 。 DM_i 计算 $x_i = H_0(\text{PID}_{DN_j})$, $y_i = H_0(r_{SC} \cdot PK_{DN_j}) \oplus K_1$, 并利用拉格朗日插值法计算 $F_i(x_i) = y_i \cdot \prod_{1 \leq j \neq i \leq n} \frac{x - x_j}{x_i - x_j}$, $F(x) = \sum_{j=1}^n F_i(x_i) = a_0 + a_1x + \dots + a_nx^n$ 。

最后, DM_i 保存消息 $M_1 = \{a_0, a_1, \dots, a_n\}$ 。

4.4 认证密钥协商阶段

1) DM_i 计算 $R_{SC} = r_{SC} \cdot P$, $K = H_0(r_{SC} \cdot PK_{EN_i})$, $D_{ID} = \text{ID}_{DM_i} \oplus K$ 和 $M_2 = W \oplus H_0(PW_{DM_i} \| r_{DM_i}) = H_0(\text{ID}_{DM_i} \| sk_{EN_i})$, 随后获取当前时间戳 T_1 , 计算认证请求消息 $M_3 = H_0(\text{ID}_{DM_i} \| K_1 \| M_1 \| T_1)$ 和 $M_4 = H_0(D_{ID} \| R_{SC} \| M_1 \| M_2 \| M_3 \| T_1)$ 。 DM_i 向 EN_i 发送 $\text{msg}_1 = \{D_{ID}, R_{SC}, M_1, M_2, M_3, M_4, T_1\}$ 以请求认证。

2) EN_i 接收到消息 $\{D_{ID}, R_{SC}, M_1, M_2, M_3, M_4, T_1\}$ 后, 首先验证时间戳 T_1 的有效性, 若时间戳 T_1 无效, 则终止认证流程。否则, EN_i 计算 $M'_4 = H_0(D_{ID} \| R_{SC} \| M_1 \| M_2 \| M_3 \| T_1)$, 判断 $M'_4 \stackrel{?}{=} M_4$ 。若相等, 则 EN_i 计算 $K' = H_0(R_{SC} \cdot sk_{EN_i})$, $\text{ID}_{DM_i}' = D_{ID} \oplus K'$ 和 $M'_2 = H_0(\text{ID}_{DM_i}' \| sk_{EN_i})$, 验证等式 $M'_2 \stackrel{?}{=} M_2$ 。若相等, 则表示 DM_i 身份正确, EN_i 对 DM_i 认证成功。 EN_i 选取随机数 $r_{EN_i} \in \mathbb{Z}_p^*$ 作为域密钥更新材料, 并获取当前时间戳 T_2 , 计算 $M_5 = \text{Enc}_{K_{EN_i}}(\text{ID}_{DM_i} \| r_{EN_i})$ 和 $M_6 = H_0(R_{SC} \| M_3 \| M_5 \| T_2)$ 。 EN_i 将 $\text{msg}_2 = \{R_{SC}, M_1, M_3, M_5, M_6, T_1, T_2\}$ 广播至域内所有的设备节点, 并更新区域密钥 $K_{EN_i} = H_0(K_{EN_i} \oplus r_{EN_i})$ 。

3) DN_j 接收到消息 $\{R_{SC}, M_1, M_3, M_5, M_6, T_1, T_2\}$ 后, 验证时间戳 T_2 的有效性, 若时间戳 T_2 无效, 则终止流程。否则, DN_j 计算 $M'_6 = H_0(R_{SC} \| M_3 \|$

$M_5 \| T_2)$, 判断 $M'_6 \stackrel{?}{=} M_6$ 。若相等, 则 DN_j 使用域密钥 K_{EN_i} 对 M_5 进行解密 ($\text{ID}_{DM_i} \| r_{EN_i}$) = $\text{Dec}_{K_{EN_i}}(M_5)$, 从而获取设备管理员的身份信息 ID_{DM_i} 和域密钥更新材料 r_{EN_i} 。 DN_j 计算 $x_i = H_0(\text{PID}_{DN_j})$, 并根据 M_1 计算 $y_i = a_0 + a_1 \cdot x_i + \dots + a_n \cdot x_i^n$, $K'_1 = y_i \oplus H_0(R_{SC} \cdot sk_{DN_j})$ 和 $M'_3 = H_0(\text{ID}_{DM_i} \| K'_1 \| M_1 \| T_1)$ 。 DN_j 判断 $M'_3 \stackrel{?}{=} M_3$, 若相等, 则表示 DN_j 对 DM_i 认证成功。 DN_j 随机选择随机数 $K_{DN_j} \in \mathbb{Z}_p^*$, 使用同态加密算法计算 $M_7 = \text{HEnc}(K_{DN_j} \cdot a_i b_i)$ 。 DN_j 获取时间戳 T_3 , 更新域密钥 $K_{EN_i} = H_0(K_{EN_i} \oplus r_{EN_i})$, 计算 $M_8 = H_0(M_7 \| K_{EN_i} \| T_3)$, 将 $\text{msg}_3 = \{M_7, M_8, T_3\}$ 发送至 EN_i 。最后, DN_j 计算与 DM_i 的会话密钥 $SK_{i-j} = K_{DN_j} \oplus \text{ID}_{DM_i}$ 。

4) EN_i 接收到消息 $\{M_7, M_8, T_3\}$ 后, 验证时间戳 T_3 的有效性, 若时间戳 T_3 无效, 则终止流程。 EN_i 计算 $M'_8 = H_0(M_7 \| K_{EN_i} \| T_3)$, 判断 $M'_8 \stackrel{?}{=} M_8$, 若相等, 则表示 EN_i 对 DN_j 认证成功。 EN_i 将接收到的 N 个 M_7 进行聚合 $M_9 = \sum_{i=1}^N M_7$ 。最后, EN_i 获取时间戳 T_4 , 计算 $M_{10} = H_0(M_9 \| K \| T_4)$, 发送 $\text{msg}_4 = \{M_9, M_{10}, T_4\}$ 至 DM_i 。

5) DM_i 接收到消息 $\{M_9, M_{10}, T_4\}$ 后, 验证时间戳 T_4 的有效性, 若时间戳 T_4 无效, 则终止认证流程。 DM_i 计算 $M'_{10} = H_0(M_9 \| K \| T_4)$, 判断 $M'_{10} \stackrel{?}{=} M_{10}$, 若相等, 则表示 DM_i 对 EN_i 认证成功。 DM_i 对消息 M_9 进行解密 $\text{HDec}(M_9) = \left(\sum_{i=1}^M K_{DN_j} \cdot a_i b_i \right)$, 则设备节点生成会话密钥集合 $SK = \sum_{i=1}^N K_{DN_j} \cdot a_i b_i$ 。 DM_i 计算与 DN_j 的会话密钥 $SK_{i-j} = (SK \bmod \text{PID}_{DN_j}) \oplus \text{ID}_{DM_i}$ 。

AKA 消息流程如图 4 所示。

4.5 设备节点撤销

当某个设备节点申请撤销时, KDC 首先删除此设备节点的 PID_{DN_j} 和 $a_j b_j$, 更新 List_{EN_i} 和 K_{EN_i} , 然后将新的 List_{EN_i} 发送给对应的 EN_i , 将

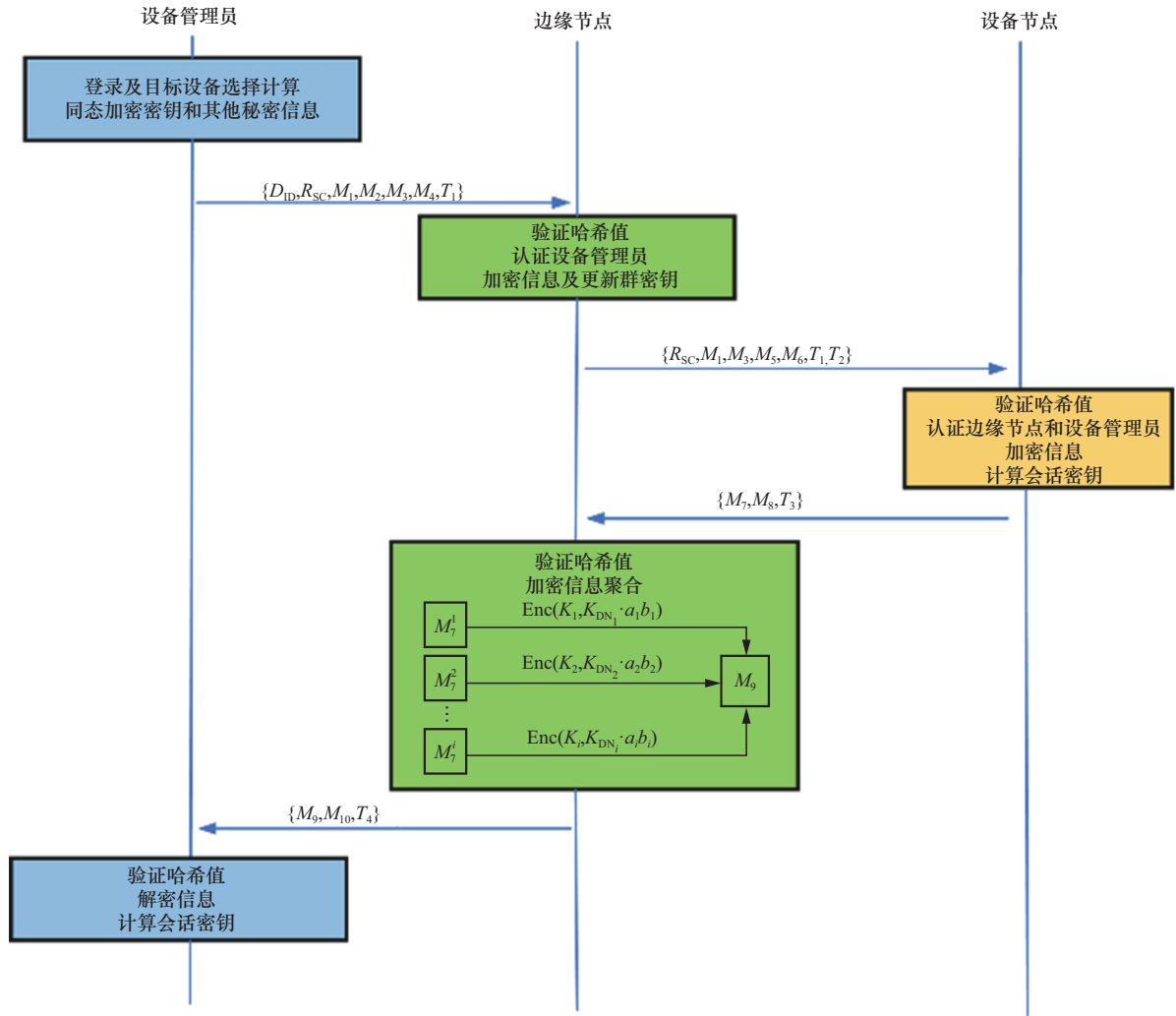


图4 AKA消息流程

Figure 4 Authentication key agreement message flow

新的 K_{EN_i} 发送给对应的 EN_i 和 DN_j 。此方法可以保证在撤销设备节点时，其他设备节点的相关参数无须修改。

4.6 设备管理员身份信息更新

1) DM_i 在智能卡中输入生物特征 Bio_{DM_i} ，计算 $\delta_{DM_i} = \text{Rep}(Bio_{DM_i}, \theta_{DM_i})$ 、 $r_{DM_i} = E \oplus H_0(\delta_{DM_i})$ 和 $V' = H_0(ID_{DM_i} || PW_{DM_i} || r_{DM_i})$ ，并验证等式 $V' = V$ 是否成立。若等式不成立，则表明 DM_i 身份信息错误，拒绝其身份信息更新请求。若等式成立，则表明身份认证成功， DM_i 输入新的口令 $PW_{DM_i}^{\text{New}}$ 和生物特征 $Bio_{DM_i}^{\text{New}}$ ，智能卡计算 $\text{Gen}(Bio_{DM_i}^{\text{New}}) = (\delta_{DM_i}^{\text{New}}, \theta_{DM_i}^{\text{New}})$ 、 $W^{\text{New}} = W \oplus H_0(PW_{DM_i} || r_{DM_i}) \oplus H_0(PW_{DM_i}^{\text{New}} || r_{DM_i})$ 、 $E^{\text{New}} = H_0(\delta_{DM_i}^{\text{New}}) \oplus r_{DM_i}$ 和 $V^{\text{New}} = H_0(ID_{DM_i} ||$

$PW_{DM_i}^{\text{New}} || r_{DM_i})$ 。智能卡使用 W^{New} 、 E^{New} 和 V^{New} 替换 W 、 E 和 V ， DM_i 身份信息更新成功。

5 安全分析

5.1 形式化安全分析

定理 假设敌手 $\mathcal{A}$ 在概率多项式时间 t 内攻破所提协议，且敌手 $\mathcal{A}$ 最多可以进行 q_e 次 Execute 询问、 q_s 次 Send 询问和 q_h 次 Hash 询问，则敌手 $\mathcal{A}$ 成功获取会话密钥的概率为：

$$\text{Adv}(\mathcal{A}) = \frac{(q_e + q_s)^2}{p} + \frac{(q_h)^2}{p} + 2\text{Adv}_{\mathcal{A}}^{\text{ECCDH}}(t') + 2\text{Adv}_{\mathcal{A}}^{\text{CPA}}(t') + 2\max\left(\frac{q_s}{|N|}, \frac{q_s}{|D|}, \delta q_s\right) \quad (5)$$

证明 本文通过定义 5 个游戏来证明所提协

议的会话密钥安全性, 分别记为 $G_k, k \in [0, 5]$ 。设 Suc_k 和 $\Pr[\text{Suc}_k]$ 分别表示游戏 G_k 中敌手 $\mathcal{A}$ 成功猜测的事件及其概率。

游戏 G_0 : 该游戏模拟真实协议环境, 所有谕言机均诚实响应敌手的询问。因此, 敌手 $\mathcal{A}$ 在该游戏中成功的概率等于其攻破真实协议的概率。由此可得:

$$\text{Adv}(\mathcal{A}) = |2\Pr(\text{Suc}_0) - 1| \quad (6)$$

游戏 G_1 : 本游戏模拟被动攻击。与游戏 G_0 的区别在于, 敌手 $\mathcal{A}$ 通过询问 $\text{Execute}(\prod_{\text{DM}_i}^{S_1}, \prod_{\text{EN}_i}^{S_2}, \prod_{\text{DN}_j}^{S_3})$ 谕言机获取协议传输的消息 $\text{msg}_1 = \{D_{\text{ID}}, R_{\text{SC}}, M_1, M_2, M_3, M_4, T_1\}$ 、 $\text{msg}_2 = \{R_{\text{SC}}, M_1, M_3, M_5, M_6, T_1, T_2\}$ 、 $\text{msg}_3 = \{M_7, M_8, T_3\}$ 和 $\text{msg}_4 = \{M_9, M_{10}, T_4\}$ 。然而, 这些消息均使用仅合法通信节点拥有的密钥 K_1 或 K 加密, 敌手 $\mathcal{A}$ 无法从中提取有效信息, 因此其攻击优势未提升。由此可得:

$$\Pr[\text{Suc}_1] = \Pr[\text{Suc}_0] \quad (7)$$

游戏 G_2 : 本游戏引入主动攻击。敌手 $\mathcal{A}$ 通过询问 $\text{Send}(\prod_{\text{DM}_i}^{S_1} / \prod_{\text{EN}_i}^{S_2} / \prod_{\text{DN}_j}^{S_3}, M)$ 和 $\text{Hash}(\text{Type}, M)$ 谕言机发起主动攻击。敌手 $\mathcal{A}$ 将伪造或者修改的消息插入协议流, 试图欺骗参与者。消息 $\text{msg}_1, \text{msg}_2, \text{msg}_3, \text{msg}_4$ 中包含时间戳和随机数等以确保随机性。根据生日悖论, 哈希查询和信息传递过程中发生碰撞的概率至多为 $\frac{(q_h)^2}{2p}$ 和 $\frac{(q_e + q_s)^2}{2p}$ 。由此可得:

$$|\Pr[\text{Suc}_2 - \text{Suc}_1]| \leq \frac{(q_e + q_s)^2}{2p} + \frac{(q_h)^2}{2p} \quad (8)$$

游戏 G_3 : 在本游戏中, 敌手 $\mathcal{A}$ 通过截获消息可以获取 $R_{\text{SC}} = r_{\text{SC}} \cdot P$, $D_{\text{ID}} = \text{ID}_{\text{DM}_i} \oplus K$ 。虽然 K 可以通过 $H_0(R_{\text{SC}} \cdot \text{sk}_{\text{CS}})$ 计算得到, 但计算 $K = H_0(r_{\text{SC}} \cdot \text{PK}_{\text{EN}}) = H_0(R_{\text{SC}} \cdot \text{sk}_{\text{EN}})$, 敌手 $\mathcal{A}$ 须知晓边缘节点的私钥 sk_{EN} 或者随机数 r_{SC} 。敌手 $\mathcal{A}$ 无法通过截获的消息直接获取密钥 K , 故敌手 $\mathcal{A}$ 在该游戏中成功的优势被归约为在多项式时间 t' 内求解 ECCDH 问题。

$$|\Pr[\text{Suc}_3 - \text{Suc}_2]| \leq \text{Adv}_{\mathcal{A}}^{\text{ECCDH}}(t') \quad (9)$$

游戏 G_4 : 在本游戏中, 敌手 $\mathcal{A}$ 截获使用密钥 K_1 进行同态加密的消息可以得到消息 $M_7 =$

$\text{HEnc}(K_{\text{DN}_j}, a_i b_i)$ 。由于加密方案满足语义安全, 敌手 $\mathcal{A}$ 的优势被归约为在多项式时间 t' 内求解选择明文攻击 (chosen-plaintext attack, CPA) 问题。由此可得:

$$|\Pr[\text{Suc}_4 - \text{Suc}_3]| \leq \text{Adv}_{\mathcal{A}}^{\text{CPA}}(t') \quad (10)$$

游戏 G_5 : 本游戏增加了 $\text{Corrput}(\prod_{\text{DM}_i}^{S_1}, a)$ 来模拟三因子认证协议的安全性。假设敌手 $\mathcal{A}$ 在任意时间最多同时获取两个身份认证因子, 则存在以下 3 种情况。

1) 敌手 $\mathcal{A}$ 通过询问 $\text{Corrput}(\prod_{\text{DM}_i}^{S_1}, 1)$ 获取 PW_{DM_i} 和智能卡中存储的 $\{\theta_{\text{DM}_i}, E, V, \text{Counter}, W\}$ 数据, 并尝试猜测生物特征 Bio_i 。经过 q_s 次询问后, 敌手 $\mathcal{A}$ 成功的概率是 $\frac{q_s}{|M|}$ 。

2) 敌手 $\mathcal{A}$ 通过询问 $\text{Corrput}(\prod_{\text{DM}_i}^{S_1}, a = 0)$ 获取 PW_{DM_i} 和生物特征 Bio_i , 并计算 $E = H_0(\delta_{\text{DM}_i}) \oplus r_{\text{DM}_i}$ 和 $V = H_0(\text{ID}_{\text{DM}_i} \| \text{PW}_{\text{DM}_i} \| r_{\text{DM}_i})$ 。但由于缺乏智能卡中的数据 r_{DM_i} , 敌手 $\mathcal{A}$ 成功的概率是可以忽略的。

3) 敌手 $\mathcal{A}$ 通过询问 $\text{Corrput}(\prod_{\text{DM}_i}^{S_1}, a = 2, 3)$ 获取生物特征 Bio_i 和智能卡数据, 尝试猜测用户口令 PW_{DM_i} 。在此场景下, 经过 q_s 次询问后, 敌手 $\mathcal{A}$ 成功的概率上界为 $\frac{q_s}{|D|}$ 。

由于上述两种情况不可能同时发生, 敌手 $\mathcal{A}$ 的总体成功概率上界为 $q_s \cdot \max\left\{\frac{1}{|M|}, \frac{1}{|D|}\right\}$ 。由此可得:

$$|\Pr[\text{Suc}_5 - \text{Suc}_4]| \leq q_s \cdot \max\left\{\frac{1}{|M|}, \frac{1}{|D|}\right\} \quad (11)$$

当询问完所有谕言机后, 由于敌手 $\mathcal{A}$ 无法获取任何与会话密钥直接相关的信息 (会话密钥由谕言机直接生成), 敌手 $\mathcal{A}$ 在区分真实会话密钥与随机字符串的游戏中不具有显著优势。因此可得:

$$\Pr[\text{Suc}_5] = \frac{1}{2} \quad (12)$$

综合上述分析, 最终可推导出:

$$\begin{aligned}
\text{Adv}(\mathcal{A}) &= 2 \left| \Pr[\text{Suc}_0] - \frac{1}{2} \right| = \\
&2 \left| \Pr[\text{Suc}_1] - \Pr[\text{Suc}_5] \right| = \\
&2 (\Pr[\text{Suc}_1] - \Pr[\text{Suc}_2] + \Pr[\text{Suc}_2] - \Pr[\text{Suc}_3] + \\
&\Pr[\text{Suc}_3] - \Pr[\text{Suc}_4] + \Pr[\text{Suc}_5] - \Pr[\text{Suc}_4]) = \\
&2 \left(\frac{(q_c + q_s)^2}{2p} + \frac{(q_h)^2}{2p} + \text{Adv}_{\mathcal{A}}^{\text{ECCDH}}(t') + \right. \\
&\text{Adv}_{\mathcal{A}}^{\text{CPA}}(t') + q_s \cdot \max \left\{ \frac{1}{|N|}, \frac{1}{|D|} \right\} \Bigg) = \\
&\frac{(q_c + q_s)^2}{p} + \frac{(q_h)^2}{p} + 2\text{Adv}_{\mathcal{A}}^{\text{ECCDH}}(t') + \\
&2\text{Adv}_{\mathcal{A}}^{\text{CPA}}(t') + 2q_s \cdot \max \left\{ \frac{1}{|N|}, \frac{1}{|D|} \right\}
\end{aligned} \tag{13}$$

5.2 非形式化安全分析

(1) 双向认证安全性

在 DM_i 注册阶段, KDC使用 EN_i 的私钥构造消息 M_1 , 确保只有持有正确私钥的边缘节点才能从消息 $\{D_{\text{ID}}, R_{\text{SC}}, M_1\}$ 中解析出密钥 K 。 EN_i 使用KDC生成的群密钥加密消息 M_5 , 确保只有该区域内成功注册的设备节点才能获取群密钥并进行解密。 DM_i 发送至 DN_j 的消息 M_3 中使用其真实身份作为秘密信息, EN_i 发送至 DM_j 的消息 M_5 中也包含 DM_i 的身份信息, DN_j 使用群密钥解密消息后, 使用 EN_i 发送的 DM_i 身份信息对消息 M_3 进行验证。

(2) 匿名性和不可链接性

DM_i 将真实 ID_{DM_i} 隐藏在 D_{ID} 中。敌手 $\mathcal{A}$ 若想获取 ID_{DM_i} , 则必须求解 $K = H_0(r_{\text{SC}} \cdot \text{PK}_{\text{CS}}) = H_0(R_{\text{SC}} \cdot \text{sk}_{\text{CS}})$ 。然而, K 的计算是基于ECCDH问题的, 故敌手 $\mathcal{A}$ 无法计算出 K 并获取 ID_{DM_i} 。并且, DM_i 通过哈希函数将设备节点的身份信息隐藏在 x_i 中, 根据哈希函数的单向性和抗碰撞性, 敌手 $\mathcal{A}$ 无法从消息 M_2 中反推出设备的真实身份。

(3) 抗智能卡被盗攻击

假设敌手 $\mathcal{A}$ 成功从设备管理员的智能卡中获取 $\{\theta_{\text{DM}_i}, E, V, \text{Counter}\}$ 。其中, $I = H_0(\text{PW}_{\text{DM}_i} \| r_{\text{DM}_i})$, $W = H_0(\text{ID}_{\text{DM}_i} \| \text{sk}_{\text{EN}_i}) \oplus I$, $E = H_0(\delta_{\text{DM}_i}) \oplus r_{\text{DM}_i}$, $V = H_0(\text{ID}_{\text{DM}_i} \| \text{PW}_{\text{DM}_i} \| r_{\text{DM}_i})$ 。由于哈希函数的单向安全性及设备管理者的生物特征唯一性, 敌手 $\mathcal{A}$ 无法通过口令猜测获取 PW_{DM_i} , 进而无法伪装成

DM_i 发起攻击。因此, 该协议能够有效抵御智能卡被盗攻击。

(4) 抗重放攻击

所提协议中所有传输的消息均包含时间戳, 且将时间戳隐藏在部分信息中。若敌手 $\mathcal{A}$ 重新发送以前截获的消息, 接收方将检测出时间戳超过规定的最大时延, 从而终止认证过程。因此, 该协议可以抵抗重放攻击。

(5) 抗边缘节点伪装攻击

由于敌手 $\mathcal{A}$ 没有 EN_i 的群密钥, 无法伪装成 EN_i 发送消息 M_5 , 因此该协议可以抵抗边缘节点伪装攻击。

(6) 抗设备节点伪装攻击

边缘节点的广播消息须使用域内群密钥加密, 而群密钥由KDC秘密分配给合法边缘节点。攻击者因未掌握群密钥, 无法解密消息 M_2 、 M_3 和 M_5 , 故无法伪装成真实设备节点发送欺骗性指令。因此, 该协议可以抗设备节点伪装攻击。

(7) 抗边缘节点被盗攻击

本文所提协议采用同态加密技术, EN_i 能够在不对加密数据进行解密的情况下协助 DM_i 和 DN_j 计算会话密钥。由于敌手 $\mathcal{A}$ 无法从边缘节点内部获取同态加密密钥, 因此该协议可以有效抵抗边缘节点被盗攻击。

(8) 前向安全性

在每次的认证密钥协商阶段, 设备管理者都会使用不同的密钥对消息进行加密, 这使得敌手 $\mathcal{A}$ 无法利用已获取的密钥对历史消息进行解密。因此, 该协议能够实现前向安全性。

(9) 抗中间人攻击

本文所提协议要求在消息传输过程中, 接收方应对发送方的身份进行验证。然而, 敌手 $\mathcal{A}$ 无法通过窃听和伪造等手段获取设备管理者、边缘节点和设备终端的私钥, 其生成的消息自然无法满足完整性与认证性要求。因此, 该协议能够抵抗中间人攻击。

(10) 抗拒绝服务攻击

本文所提协议的认证密钥协商过程始于设备管理员发起的认证请求 $\text{msg}_1 = \{D_{\text{ID}}, R_{\text{SC}}, M_1, M_2, M_3, M_4, T_1\}$ 。协议规定, 仅当请求方持有正确的认证信息 ID_{DM_i} 、 PW_{DM_i} 和 Bio_{DM_i} 时, 方可发起认证请求。同时, 为保障消息的时效性, 在消息传输过

程中采用了时间戳机制。边缘节点在接收消息时,若检测到时间戳超出规定时间窗口,则立即终止通信。这一设计可有效阻止恶意攻击者通过发送过期消息消耗系统资源,从而抵御拒绝服务攻击。

(11) 抗离线口令猜测攻击

在智能卡对设备管理者进行身份认证的过程中,需要设备管理者计算 $E = H_0(\delta_{DM_i}) \oplus r_{DM_i}$ 。由于设备管理者的生物特征 Bio_{DM_i} 具有唯一性,敌手 $\mathcal{A}$ 无法计算出 δ_{DM_i} 。因此,该协议能够抵抗离线口令猜测攻击。

(12) 抗在线口令猜测攻击

智能卡在初始化时即设定了计数器 Counter,用于记录连续输入错误的次数。当 Counter 值超过设定阈值时,系统会立即判定当前访问者为非法用户,并终止对设备管理员的身份认证流程。因此,该协议能够抵抗在线口令猜测攻击。

5.3 CPN 形式化安全分析

本文采用 CPN 形式化分析方法^[42-43]对协议及敌手进行建模,并利用状态空间验证其安全性。模型构建聚焦于协议的双向认证及会话密钥协商阶段,采用分层建模方法逐层细化协议逻辑。所提协议的顶层 CPN 模型结构如图 5 所示。在原始模型构建完成后,通过在协议中引入 Dolev-Yao 攻击者模型,进一步开展安全性分析。

首先,需要使用状态空间分析工具对协议的 CPN 模型进行一致性验证。分析表 2 的状态空间

统计结果可知,状态空间节点数与强连通节点数相等、有向弧与强连通弧的数目相等。这表明所建立的模型中不存在导致无限循环或迭代的行为,模型运行正确。针对引入攻击者的模型,利用 SML 语句对攻击者的知识库所进行查询,结果如图 6 所示。查询结果显示,攻击者的知识库所中除其原始知识外,仅包含设备节点的区域 ID 号,并未获取到任何与会话密钥相关的秘密信息。由此可证明,该协议具备安全性。

表 2 状态空间统计结果

Table 2 Statistical results of the state space

参数	原始协议	加入攻击者
状态空间节点/个	477	1 245
有向弧/条	1 032	3 204
强连通节点/个	477	1 245
强连通弧/条	1 032	3 204
死节点/个	12	24

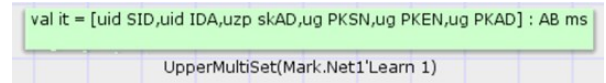


图 6 攻击者知识库所查询图

Figure 6 Attacker knowledge base query diagram

6 仿真与对比分析

6.1 安全性对比

本节将所提协议与文献[19]、[22]、[24]、

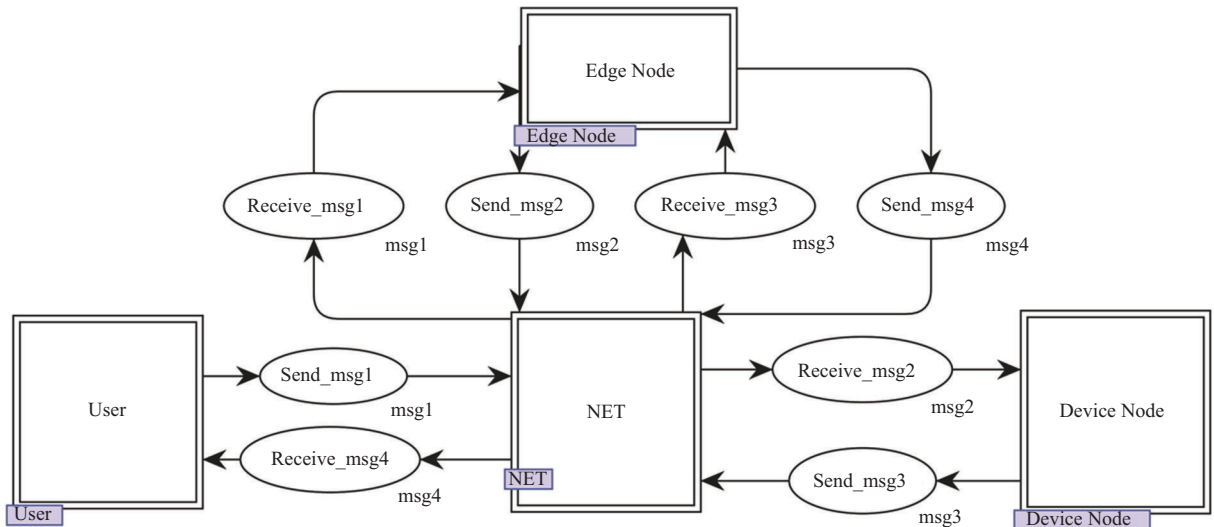


图 5 所提协议的顶层 CPN 模型结构

Figure 5 The top-level CPN model structure of the proposed protocol

[26]、[33]、[34]中的6个协议进行安全性和功能性对比分析。由表3可知,文献[19]、[22]、[24]、[26]仅支持一对一的相互认证和密钥协商,具体存在以下问题。文献[19]无法抵御拒绝服务攻击、智能设备被盗攻击和重放攻击,且未考虑设备节点的加入和撤销机制;同时,协议因缺乏用户或管理员的注册过程,其抵御口令猜测攻击的能力无法验证。文献[22]存在安全缺陷,无法抵御中间人攻击和设备伪装攻击。文献[24]未部署边缘节点或网关来协助通信双方进行密钥协商,故其抵御边缘节点伪装攻击或边缘节点被盗攻击的能力无法判定。文献[26]则存在无法抵御口令猜测攻击的安全漏洞。

相比之下,文献[33]和[34]的协议实现了一对多的双向认证和密钥协商。然而,文献[33]基于网关节点完全可信的假设,导致网关在辅助通信双方协商会话密钥时能够获取双方的会话密钥信息,因此该协议无法有效防御边缘节点被盗攻击。文献[34]通过区块链完成设备节点和服务提供商之间的身份验证,却无法提供相关安全属性。综上所述,本文所提协议能够满足所有的安全需求和功能需求。

6.2 计算开销分析

本文采用基于Ubuntu 22.04 LTS的树莓派4B

设备模拟资源受限的终端环境。同时,使用一台配备Intel Core i7-9750H处理器、16 GB内存,并运行Ubuntu 18.04 LTS操作系统的笔记本电脑,以模拟计算能力较强的边缘节点。在实验过程中,利用PyPBC库测量不同密码学算法的运算执行时间。

表4 密码学算法执行时间

Table 4 Execution time of cryptographic algorithms

符号	描述	执行时间/ms	
		非资源受限	资源受限
T_h	哈希函数运算	0.001 5	0.009 0
T_{esm}	椭圆曲线标量乘法运算	1.273 6	2.511 2
T_{esa}	椭圆曲线点加运算	0.038 8	0.054 2
T_e	AES-128加密运算	0.030 9	0.333 8
T_d	AES-128解密运算	0.005 4	0.039 4

针对协议中不同的消息参与方,本文对其计算开销展开对比分析,以评估各协议在不同角色下的计算开销。具体而言,由于文献[24]不包含边缘节点,该协议中边缘节点的计算开销为零;文献[19]、[22]、[24]和[26]本身不支持一对多认证密钥协商操作,而文献[33]和[34]具备一对多认证密钥协商能力。假设单个设备管理员需要与 n 个设备节点进行认证密钥协商操作,则不同参与方在一对多环境下的计算开销如表5所示。

表3 安全性和功能性对比

Table 3 Comparative of safety and functionality

安全属性	文献[19]	文献[22]	文献[24]	文献[26]	文献[33]	文献[34]	本文协议
双向认证	√	√	√	√	√	√	√
中间人攻击	√	×	√	√	√	√	√
匿名性	√	√	√	√	√	√	√
不可链接性	√	√	√	√	√	√	√
抗智能卡被盗攻击	—	√	√	√	√	—	√
抗重放攻击	×	√	√	√	√	√	√
抗边缘节点(网关)伪装攻击	√	√	—	√	√	√	√
抗边缘节点被盗攻击	×	√	—	√	×	—	√
前向安全性	√	√	√	√	√	√	√
抗拒绝服务攻击	×	√	√	√	√	√	√
抗离线口令猜测攻击	—	√	√	×	√	—	√
抗在线口令猜测攻击	—	×	×	×	√	—	√
设备节点撤销	—	√	—	√	√	√	√
一对多认证密钥协商	×	×	×	×	√	√	√

通过计算可得, 文献[19]的计算开销总和为 $(20T_{\text{esm}} + 4T_e + 4T_d + 38T_h)n$, 文献[22]的计算开销总和为 $(11T_{\text{esm}} + 30T_h)n$, 文献[24]的计算开销总和为 $(6T_{\text{esm}} + 18T_h + 2T_e + 2T_d + 4T_{\text{esa}})n$, 文献[26]的计算开销总和为 $(10T_{\text{esm}} + 17T_h)n$, 文献[33]的计算开销总和为 $2T_{\text{esm}} + (15 + 8n)T_h + (3 + 2n)T_{\text{esa}} + (3 + n)T_e + (2 + n)T_d$, 文献[34]的计算开销总和为 $(9n + 3)T_{\text{esm}} + (7n - 2)T_{\text{esa}} + 9nT_h$, 本文协议的计算开销总和为 $(3 + n)T_{\text{esm}} + (7n + 9)T_h + T_e + nT_d + (n^2 + 4n + 1)T_{\text{esa}}$ 。

从图7可以看出, 随着目标设备节点数量的增加, 传统的一对一认证协议在一对多认证场景下, 由于认证发起方须与每个目标设备节点分别进行认证, 其计算开销呈线性增长趋势。此外, 从图8可以发现, 文献[33]所提协议的整体计算开销低于本文协议。这主要是因为该协议假设网关为完全可信实体, 网关可直接为通信双方计算会话密钥, 无须通过其他密码学算法抵御边缘节点被盗窃攻击; 而本文协议将边缘节点设定为半可信实体, 认证双方需要借助同态加密算法确保会

话密钥信息的机密性, 由此产生了额外的计算开销。不过, 本文协议在增加少量计算开销的同时, 显著提升了协议的整体安全性。

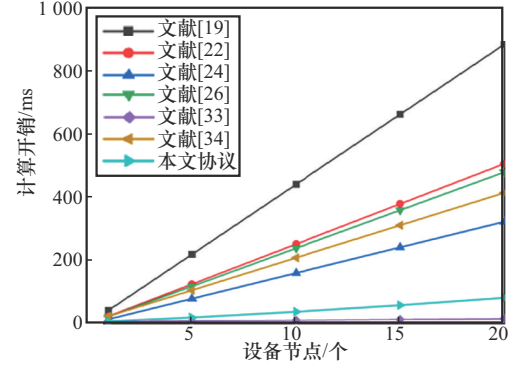


图8 不同协议的整体计算开销对比

Figure 8 Comparison of the overall computational overhead among different protocols

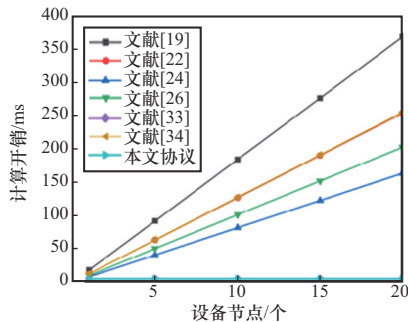
6.3 通信开销分析

通信开销是指参与者在完成认证过程中交换或传输的数据量。令 $|G|$ 表示群 G 的比特长度、 $|Z_p^*|$ 表示 Z_p^* 的比特长度、 $|E|$ 表示对称加密密

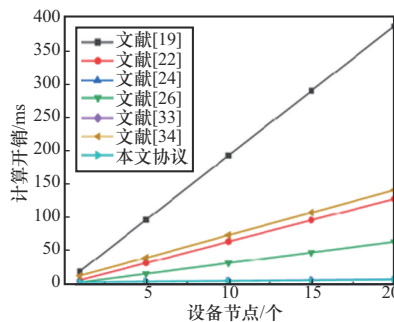
表5 计算开销对比

Table 5 Comparison of computation cost

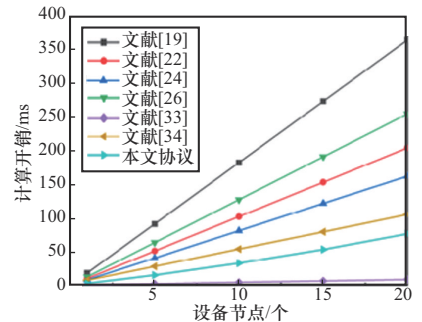
文献	管理员/设备	网关/边缘节点	设备节点
[19]	$(7T_{\text{esm}} + T_e + T_d + 12T_h)n$	$(6T_{\text{esm}} + 2T_e + 2T_d + 13T_h)n$	$(7T_{\text{esm}} + T_e + T_d + 13T_h)n$
[22]	$(5T_{\text{esm}} + 10T_h)n$	$(2T_{\text{esm}} + 12T_h)n$	$(4T_{\text{esm}} + 8T_h)n$
[24]	$(3T_{\text{esm}} + 9T_h + T_e + T_d + 3T_{\text{esa}})n$	N/A	$(3T_{\text{esm}} + 9T_h + T_e + T_d + T_{\text{esa}})n$
[26]	$(4T_{\text{esm}} + 6T_h)n$	$(4T_h + T_{\text{esm}})n$	$(7T_h + 5T_{\text{esm}})n$
[33]	$6T_h + 2T_{\text{esm}} + 2T_{\text{esa}} + T_e + T_d$	$(9 + 2n)T_h + T_{\text{esm}} + (n + 1)T_d + 2T_e + 2nT_{\text{esa}}$	$(6T_h + T_e + T_d)n$
[34]	$5nT_{\text{esm}} + 2nT_{\text{esa}} + 4nT_h$	$(2n + 2)T_{\text{esm}} + (4n - 2)T_{\text{esa}} + 3nT_h$	$(2n + 1)T_{\text{esm}} + nT_{\text{esa}} + 2nT_h$
本文协议	$2T_{\text{esm}} + T_{\text{esa}} + 4T_h$	$T_{\text{esm}} + T_e + (n + 5)T_h + 2nT_{\text{esa}}$	$(6T_h + T_d + T_{\text{esm}} + (n + 2)T_{\text{esa}})n$



(a) 管理员/用户计算开销对比



(b) 边缘节点/网关计算开销对比



(c) 设备节点计算开销对比

图7 不同通信方计算开销对比

Figure 7 Comparison of computation cost among different entities

文的比特长度、 $|ID|$ 表示实体身份的比特长度、 $|TS|$ 表示时间戳的比特长度。 $|G|$ 、 $|Z_p^*|$ 、 $|E|$ 、 $|ID|$ 和 $|TS|$ 的长度分别为160 bit、160 bit、128 bit、32 bit 和 32 bit。文献[19]、[22]、[24]、[26]、[33]和[34]所提协议的整体通信开销分别为 $(8|G| + 4|E| + 4|Z_p^*|)n$ 、 $(3|G| + 3|TS| + 9|Z_p^*|)n$ 、 $(|G| + 2|TS| + |Z_p^*| + 2|E|)n$ 、 $(6|G| + 4|TS| + 5|Z_p^*| + 4|ID|)n$ 、 $(3 + 4n)|Z_p^*| + (2 + 4n)|E| + (2 + 3n)|TS|$ 和 $5n|G| + (2n + 1)|TS| + 5n|Z_p^*| + 4n|ID|$ ，本文协议通信开销为 $2|G| + (3 + n)|TS| + (5 + 4n)|Z_p^*| + |ID| + |E|$ 。针对一对多认证密钥协商场景，不同参与方的通信开销如表6所示。通过分析图9和图10可知，文献[24]所提协议的通信开销最低，原因是其认证过程不涉及网关或边缘节点，通信交互较少，因此数据传输开销最小。其次，文献[33]中认证双方的会话密钥由网关生成，且该网关被假设为完全可信实体，因此无须针对不同设备节点分别发送认证信息，而是由网关直接广播认证消息，从而减少了数据传输量。然而，

本文协议在认证过程中需要针对不同的设备节点单独发送认证信息，以确保认证数据的完整性和安全性，由此产生了额外的通信开销。尽管如此，该协议有效提高了传输数据的安全性，避免了因完全信任网关而导致的会话密钥泄露风险。

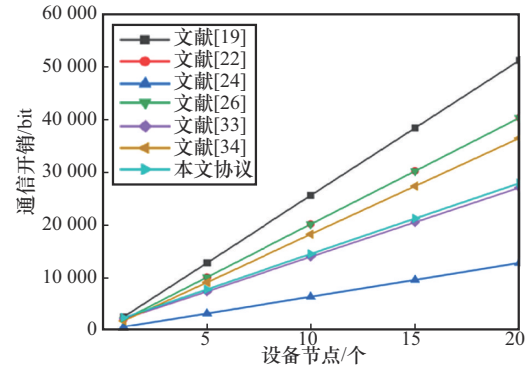


图10 不同协议整体通信开销对比

Figure 10 Comparison of the overall communication overhead among different protocols

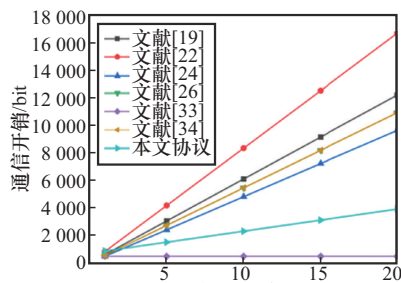
6.4 能耗对比

在IIoT环境中，设备节点通常受限于计算

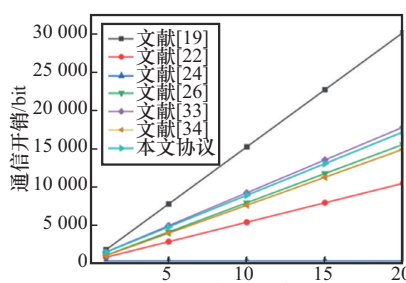
表6 通信开销对比

Table 6 Comparison of communication overhead

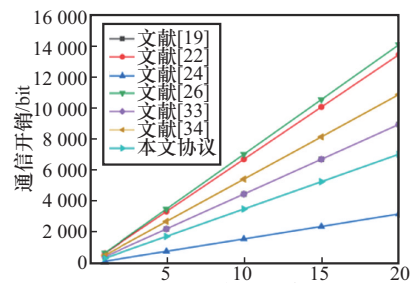
文献	管理员/用户	网关/边缘节点	设备节点
[19]	$(2 G + E + Z_p^*)n$	$(5 G + 3 E + 2 Z_p^*)n$	$(G + E + Z_p^*)n$
[22]	$(G + TS + 4 Z_p^*)n$	$(G + TS + 2 Z_p^*)n$	$(G + TS + 3 Z_p^*)n$
[24]	$(G + TS + Z_p^* + E)n$	N/A	$(TS + E)n$
[26]	$(2 G + TS + Z_p^* + ID)n$	$(2 G + 2 TS + 2 Z_p^* + 2 ID)n$	$(2 G + TS + 2 Z_p^* + ID)n$
[33]	$2 Z_p^* + E + TS $	$(2n + 1) E + (3n + 1) Z_p^* + (2n + 1) TS $	$(E + TS + Z_p^*)n$
[34]	$2n G + n TS + n Z_p^* + n ID $	$2n G + n TS + 2n Z_p^* + 2n ID $	$n G + n TS + 2n Z_p^* + n ID $
本文协议	$ G + TS + (3 + n) Z_p^* + ID $	$n G + (3n + 2) Z_p^* + n E + (2n + 1) TS $	$(TS + 2 Z_p^*)n$



(a) 管理员/用户通信开销对比



(b) 边缘节点/网关通信开销对比



(c) 设备节点通信开销对比

图9 不同通信方通信开销对比

Figure 9 Comparison of communication overhead among different entities

能力、存储空间和能量供应。因此,评估不同认证协议的适用性时,须从能耗角度展开分析,以明确其在资源受限设备上的可行性。高能耗的认证协议可能会显著缩短设备的续航时间,进而影响系统的整体运行效率。文献[44]给出了 133 MHz 的微处理器在执行不同密码学算法时的能耗:对称加解密运算为 0.002 17 mJ/次,哈希运算为 0.000 108 mJ/次,椭圆曲线标量乘法运算 8.8 mJ/次,椭圆曲线点加运算为 0.001 085 mJ/次。由图 11 可知,不同协议在认证过程中所采用的密码学运算类型和频次对整体能耗存在显著影响。部分协议频繁使用计算代价较高的椭圆曲线标量乘法运算,导致能耗大幅增加。此外,传统的一对一 AKA 协议在多设备认证场景中,需要针对每个目标设备节点单独执行认证流程,由此产生额外的计算开销和通信开销,进一步加剧了能耗。

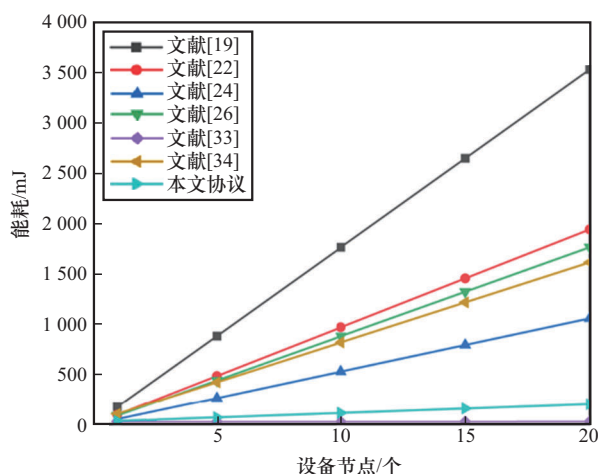


图 11 不同协议的能耗对比

Figure 11 Comparison of energy consumption among different protocols

相比之下,本文协议在保障安全性的同时,优化了计算能耗。尽管与文献[33]所提协议相比,本文协议因引入抗边缘节点被盗攻击安全机制,需要额外执行部分密码学运算,导致能耗略有增加,但这一设计有效增强了数据传输的安全性,降低了潜在的攻击风险。总体而言,本文协议适用于计算资源有限的 IIoT 环境。

6.5 存储开销对比

工业环境中的物联网设备与用户往往面临资

源受限的问题,这些设备和用户需要存储大量实时监控数据、工业控制程序及配置信息。因此,在设计 AKA 协议时,应尽可能减少会话密钥协商过程中对存储资源的占用,避免给设备带来过大的存储负担而影响其正常运行。尤其在设备计算和存储资源有限的情况下,优化存储开销对提升系统的可扩展性与稳定性至关重要。其中, $|\text{Hash}|$ 、 $|\text{ID}|$ 、 $|\text{PW}|$ 、 $|\text{Bio}|$ 、 $|\mathbb{Z}_p^*|$ 、 $|\text{Key}|$ 、 $|\mathbb{G}|$ 的存储开销分别为 160 bit、32 bit、48 bit、32 bit、160 bit、128 bit、160 bit。如图 12 所示,部分现有协议在设备注册完成后,需要存储大量的公私钥对、证书或其他密钥管理信息,导致设备存储资源消耗较大。这种设计在存储受限的设备上可能带来额外的存储负担,降低系统的可扩展性,并增加维护和密钥管理的复杂度。相比之下,本文协议在存储开销方面进行了优化。通信双方仅需存储基本的身份信息、生物特征数据或相关密钥信息,无须维护额外的公私钥对或证书管理数据。这一设计在减少存储占用的同时,降低了密钥管理的复杂性,并提高了系统的安全性。

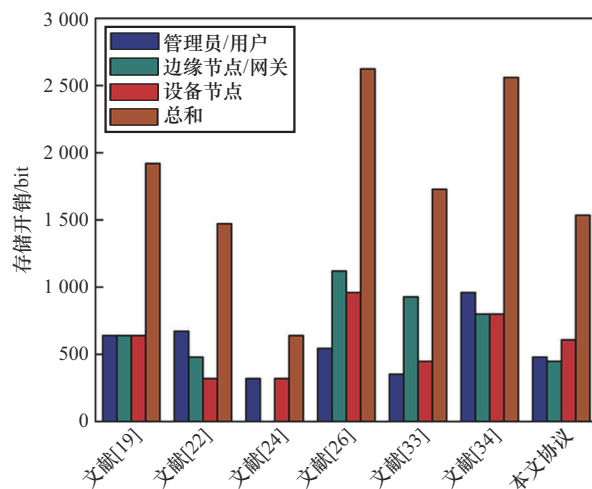


图 12 不同协议的存储开销对比

Figure 12 Comparison of storage consumption among different protocols

7 NS-3 仿真模拟

为评估本文协议在 IIoT 场景下的通信性能,本文基于 NS-3 网络仿真工具搭建了仿真实验平台。仿真场景设定为网络覆盖面积为 $2 \times 2 \text{ km}^2$ 的区域,仿真总时长为 1 800 s。为全面评估协议在多设

备接入场景下的适应性与可扩展性,实验分别设置目标设备节点为10、50、100、150和200个,并在各规模条件下开展通信性能对比分析。在实验过程中,网络实体间主要交换4类消息:① DM_i 向 EN_i 发送 $msg_1 = \{D_{ID}, R_{SC}, M_1, M_2, M_3, M_4, T_1\}$,通信开销为864 bit;② EN_i 向 DN_j 发送 $msg_2 = \{R_{SC}, M_1, M_3, M_5, M_6, T_1, T_2\}$,通信开销为832 bit;③ DN_j 向 EN_i 发送 $msg_3 = \{M_7, M_8, T_3\}$,通信开销为352 bit;④ EN_i 向 DM_i 发送 $msg_4 = \{M_9, M_{10}, T_4\}$,通信开销为352 bit。仿真环境参数配置如表7所示。通过动态调整目标设备数量,系统测量了协议在通信过程中的时延、吞吐量等关键指标,从而综合评估该协议在通信效率、系统稳定性及规模扩展性等方面的表现。

表7 仿真环境参数

Table 7 Simulation environment parameter

参数	说明
仿真平台	Ubuntu22.04 LTS
硬件平台	Intel i7-9750H, 16 GB RAM
设备管理者/个	1
边缘节点/个	1
设备节点/个	10、50、100、150、200
物理层协议	IEEE 802.11a
网络覆盖面积/ km^2	2×2
管理员通信范围/m	200
设备通信范围/m	50
模拟时间/s	1 800

7.1 端到端时延分析

端到端时延是指消息从源节点传输到目的节点所需的平均时间,是衡量方案实时性和高效性的重要指标之一,其计算公式为 $Delay = \sum_{i=1}^{N_{rx}} Delay_i / N_{rx}$ 。其中, $Delay_i$ 表示第*i*个数据包成功到达目的节点的时延, N_{rx} 表示成功接收到的数据包总数。由图13的仿真实验数据可知,随着设备节点数量的增加,端到端时延呈现显著的增长趋势。这是因为当参与通信的设备节点增多时,网络中的消息交换频率和数据包数量都会相应增加,可能引发网络拥塞和资源竞争,导致

数据包的排队等待时间增加,最终使端到端时延增加。

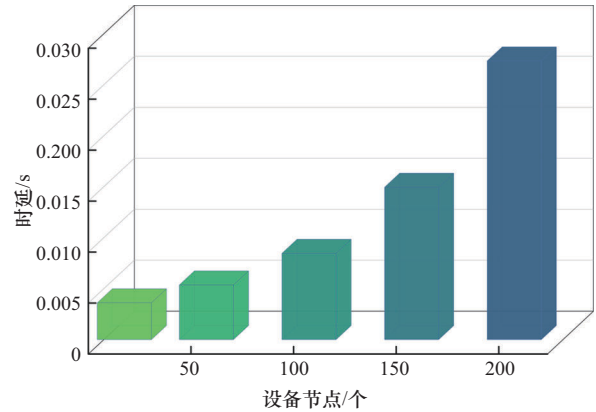


图13 端到端时延对比

Figure 13 End-to-end delay comparison

7.2 系统吞吐量分析

吞吐量是指单位时间内网络成功传输的数据量,是评价网络性能的另一项重要指标,直接体现了网络对实际生产环境中海量设备连接和高频数据传输的支撑能力,其计算公式为 $Throughput = N_{rx} \times MB/Time$ 。其中,MB表示单个数据包的大小, N_{rx} 表示仿真过程中成功接收的数据包总数,Time表示仿真试验的总持续时间。从图14的仿真结果来看,随着网络中设备节点数量的增加,系统整体的吞吐量呈现显著上升趋势。

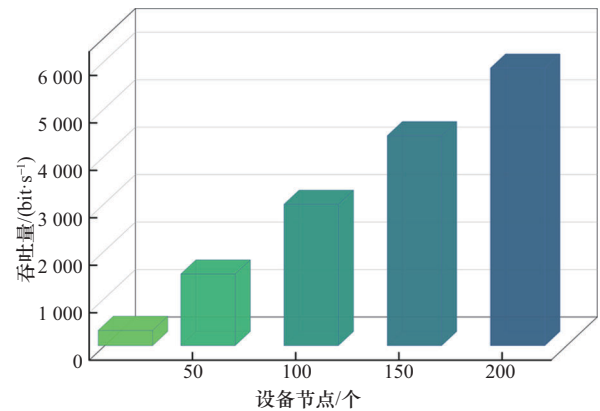


图14 系统吞吐量对比

Figure 14 System throughput comparison

7.3 数据包丢失率分析

数据包丢失率是指在仿真时间内网络中未成功传输的数据包占总发送数据包的比例,其计算公式为 $LossRate = (1 - N_{rx}/N_{tx}) \times 100\%$ 。其中,

N_{rx} 表示成功接收的数据包总数, N_{tx} 表示发送的数据包总数。从图 15 所示的仿真结果来看, 随着网络中目标设备节点数量的增加, 数据包丢失率呈现显著上升趋势。这是因为目标设备增多后, 网络通信量显著提升, 导致数据包的传输负载增加, 加剧了网络拥塞与数据碰撞, 最终增加了数据丢包率。

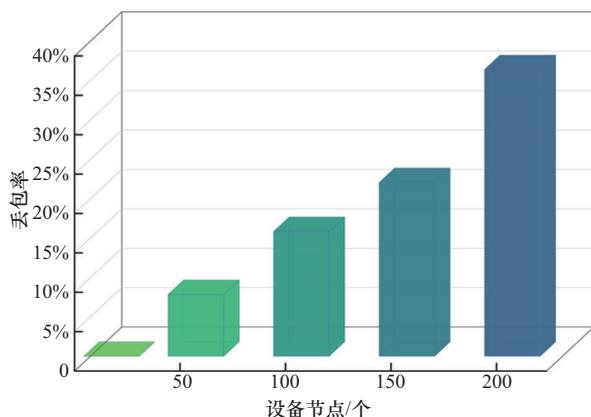


图 15 数据丢包率对比

Figure 15 Packet loss rate comparison

8 结束语

针对 IIoT 环境中一对一 AKA 协议在多设备认证场景下计算和通信开销较大的问题, 本文提出了一种安全的多设备 AKA 协议。该协议通过引入分层架构, 可有效缓解设备节点在认证密钥会话过程中因性能受限而产生的高计算开销问题。为充分利用边缘节点的强大计算能力, 本文所提协议采用同态加密技术, 确保边缘节点无法获取设备管理者与设备节点之间的会话密钥, 从而规避因边缘节点遭受攻击而导致的会话密钥泄露风险。此外, 该协议结合了椭圆曲线密码学、中国剩余定理和哈希函数等密码学技术, 在支持多设备会话密钥协商的同时, 能够保障工业控制数据传输的机密性、完整性和认证性。综合性能分析与 NS-3 仿真结果表明, 本文协议在保障高安全性的同时, 显著减少了计算开销与通信开销, 并在大规模设备接入场景中展现出良好的可扩展性。

参考文献:

[1] 陈曦, 姜亚光, 林昕, 等. 工业互联网蜜网技术综述[J]. 网络与信息

安全学报, 2024, 10(5): 23-38.

CHEN X, JIANG Y G, LIN X, et al. Overview on industrial Internet honeynet technology[J]. Chinese Journal of Network and Information Security, 2024, 10(5): 23-38.

[2] XU H S, WU J, PAN Q Q, et al. A survey on digital twin for industrial Internet of things: applications, technologies and tools[J]. IEEE Communications Surveys & Tutorials, 2023, 25(4): 2569-2598.

[3] BEHNKE I, AUSTAD H. Real-time performance of industrial IoT communication technologies: a review[J]. IEEE Internet of Things Journal, 2024, 11(5): 7399-7410.

[4] AHMED S F, BIN ALAM M S, HOQUE M, et al. Industrial Internet of things enabled technologies, challenges, and future directions[J]. Computers and Electrical Engineering, 2023, 110: 108847.

[5] CHAUDHARY S, MISHRA P K. DDoS attacks in industrial IoT: a survey[J]. Computer Networks, 2023, 236: 110015.

[6] 董裕民, 张静, 谢昌佐, 等. 云边端架构下边缘智能计算关键问题综述: 计算优化与计算卸载[J]. 电子与信息学报, 2024, 46(3): 765-776.

DONG Y M, ZHANG J, XIE C Z, et al. A survey of key issues in edge intelligent computing under cloud-edge-terminal architecture: computing optimization and computing offloading[J]. Journal of Electronics & Information Technology, 2024, 46(3): 765-776.

[7] YOUNAN M, HOUSSEIN E H, ELHOSENY M, et al. Challenges and recommended technologies for the industrial Internet of things: a comprehensive review[J]. Measurement, 2020, 151: 107198.

[8] TANGE K, DE DONNO M, FAFOUTIS X, et al. A systematic survey of Industrial Internet of things security: requirements and fog computing opportunities[J]. IEEE Communications Surveys & Tutorials, 2020, 22(4): 2489-2520.

[9] 伍高飞, 袁紫依, 孙思贤, 等. 云数据安全去重技术研究综述[J]. 密码学报, 2023, 10(6): 1099-1117.

WU G F, YUAN Z Y, SUN S X, et al. A survey on secure data deduplication techniques in cloud[J]. Journal of Cryptologic Research, 2023, 10(6): 1099-1117.

[10] PATEL C, BASHIR A K, ALI ALZUBI A, et al. EBAKE-SE: a novel ECC-based authenticated key exchange between industrial IoT devices using secure element[J]. Digital Communications and Networks, 2023, 9(2): 358-366.

[11] DIFFIE W, HELLMAN M E. New directions in cryptography[M]// Democratizing Cryptography. New York: ACM, 2022: 365-390.

[12] 郑路, 冯涛, 苏春华. 基于 CPN 的车载网络无证书匿名认证和密钥协商方案研究[J]. 通信学报, 2024, 45(6): 101-116.

ZHENG L, FENG T, SU C H. Research on certificateless anonymous authentication and key agreement scheme of vehicle network based on CPN[J]. Journal on Communications, 2024, 45(6): 101-116.

[13] OLAKANMI O O, ODEYEMI K O. Faster and efficient cloud-server-aided data de-duplication scheme with an authenticated key agreement for industrial Internet-of-things[J]. Internet of Things, 2021, 14: 100376.

[14] 胡向东, 唐玲玲. 基于轻量级梯度提升机优化的工业互联网入侵检测方法[J]. 网络与信息安全学报, 2023, 9(2): 46-55.

HU X D, TANG L L. Method on intrusion detection for industrial

- Internet based on light gradient boosting machine[J]. Chinese Journal of Network and Information Security, 2023, 9(2): 46-55.
- [15] LI X, NIU J W, BHUIYAN M Z A, et al. A robust ECC-based provable secure authentication protocol with privacy preserving for industrial Internet of things[J]. IEEE Transactions on Industrial Informatics, 2018, 14(8): 3599-3609.
- [16] LI X, PENG J Y, NIU J W, et al. A robust and energy efficient authentication protocol for industrial Internet of things[J]. IEEE Internet of Things Journal, 2018, 5(3): 1606-1615.
- [17] JIANG Q, ZHANG N, NI J B, et al. Unified biometric privacy preserving three-factor authentication and key agreement for cloud-assisted autonomous vehicles[J]. IEEE Transactions on Vehicular Technology, 2020, 69(9): 9390-9401.
- [18] ELDEFRAWY M H, PEREIRA N, GIDLUND M. Key distribution protocol for industrial Internet of things without implicit certificates[J]. IEEE Internet of Things Journal, 2019, 6(1): 906-917.
- [19] PATONICO S, BRAEKEN A, STEENHAUT K. Identity-based and anonymous key agreement protocol for fog computing resistant in the Canetti-Krawczyk security model[J]. Wireless Networks, 2023, 29(3): 1017-1029.
- [20] LEE T F, CHEN W Y. Lightweight fog computing-based authentication protocols using physically unclonable functions for Internet of medical things[J]. Journal of Information Security and Applications, 2021, 59: 102817.
- [21] TANVEER M, ZAHID A H, AHMAD M, et al. LAKE-IoD: lightweight authenticated key exchange protocol for the Internet of drone environment[J]. IEEE Access, 2020, 8: 155645-155659.
- [22] SRINIVAS J, DAS A K, WAZID M, et al. Designing secure user authentication protocol for big data collection in IoT-based intelligent transportation system[J]. IEEE Internet of Things Journal, 2021, 8(9): 7727-7744.
- [23] TANVEER M, KHAN A U, KUMAR N, et al. RAMP-IoD: a robust authenticated key management protocol for the Internet of drones[J]. IEEE Internet of Things Journal, 2022, 9(2): 1339-1353.
- [24] KHAN A A, KUMAR V, AHMAD M, et al. PALK: password-based anonymous lightweight key agreement framework for smart grid[J]. International Journal of Electrical Power & Energy Systems, 2020, 121: 106121.
- [25] CHAUDHRY S A. Correcting PALK: password-based anonymous lightweight key agreement framework for smart grid[J]. International Journal of Electrical Power & Energy Systems, 2021, 125: 106529.
- [26] XU H, HSU C, HARN L, et al. Three-factor anonymous authentication and key agreement based on fuzzy biological extraction for industrial Internet of things[J]. IEEE Transactions on Services Computing, 2023, 16(4): 3000-3013.
- [27] WANG D, CAO Y, LAM K Y, et al. Authentication and key agreement based on three factors and PUF for UAV-assisted post-disaster emergency communication[J]. IEEE Internet of Things Journal, 2024, 11(11): 20457-20472.
- [28] WANG C Y, WANG D, XU G A, et al. Efficient privacy-preserving user authentication scheme with forward secrecy for industry 4.0[J]. Science China Information Sciences, 2021, 65(1): 112301.
- [29] VINOTH R, DEBORAH L J, VIJAYAKUMAR P, et al. Secure multifactor authenticated key agreement scheme for industrial IoT[J]. IEEE Internet of Things Journal, 2021, 8(5): 3801-3811.
- [30] WANG Q X, WANG D. Understanding failures in security proofs of multi-factor authentication for mobile devices[J]. IEEE Transactions on Information Forensics and Security, 2022, 18: 597-612.
- [31] DEEBAK B D, MEMON F H, CHENG X C, et al. Seamless privacy-preservation and authentication framework for IoT-enabled smart eHealth systems[J]. Sustainable Cities and Society, 2022, 80: 103661.
- [32] CUI J, ZHANG X Y, ZHONG H, et al. Extensible conditional privacy protection authentication scheme for secure vehicular networks in a multi-cloud environment[J]. IEEE Transactions on Information Forensics and Security, 2019, 15: 1654-1667.
- [33] MING Y, YANG P F, MAHDIKHANI H, et al. A secure one-to-many authentication and key agreement scheme for industrial IoT[J]. IEEE Systems Journal, 2023, 17(2): 2225-2236.
- [34] LUO F T, HUANG R W, XIE Y Q. Hybrid blockchain-based many-to-many cross-domain authentication scheme for smart agriculture IoT networks[J]. Journal of King Saud University - Computer and Information Sciences, 2024, 36(2): 101946.
- [35] PEI D, SALOMAA A, DING C. Chinese remainder theorem: applications in computing, coding, cryptography[M]. Singapore: World Scientific, 1996.
- [36] ACAR A, AKSU H, ULUAGAC A S, et al. A survey on homomorphic encryption schemes: theory and implementation[J]. ACM Computing Surveys (Csur), 2018, 51(4): 1-35.
- [37] BRAKERSKI Z, GENTRY C, VAIKUNTANATHAN V. Fully homomorphic encryption without bootstrapping[J]. ACM Transactions on Computation Theory, 2014, 6(3): 1-36.
- [38] FAN J, VERCAUTEREN F. Somewhat practical fully homomorphic encryption[J]. Cryptology ePrint Archive, 2012.
- [39] CHEON J H, KIM A, KIM M, et al. Homomorphic encryption for arithmetic of approximate numbers[C]// Proceedings of the 23rd International Conference on the Theory and Applications of Cryptology and Information Security. Berlin: Springer, 2017: 409-437.
- [40] DOLEV D, YAO A. On the security of public key protocols[J]. IEEE Transactions on Information Theory, 1983, 29(2): 198-208.
- [41] CANETTI R, KRAWCZYK H. Analysis of key-exchange protocols and their use for building secure channels[C]// Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2001: 453-474.
- [42] JENSEN K. An introduction to the practical use of coloured petri nets[J]. Lectures on Petri Nets II: Applications: Advances in Petri Nets 3, 1998: 237-292.
- [43] 龚翔, 冯涛, 杜谨泽. 基于CPN的安全协议形式化建模及安全分析方法[J]. 通信学报, 2021, 42(9): 240-253.
- GONG X, FENG T, DU J Z. Formal modeling and security analysis method of security protocol based on CPN[J]. Journal on Communications, 2021, 42(9): 240-253.
- [44] LONG Y Y, PENG C G, TAN W J, et al. Blockchain-based anonymous authentication and key management for Internet of things with Chebyshev chaotic maps[J]. IEEE Transactions on Industrial Informatics, 2024, 20(5): 7883-7893.

[作者简介]



吴毅 (1999-), 男, 甘肃景泰人, 兰州理工大学博士生, 主要研究方向为工业物联网安全、工控协议安全。



柳春岩 (1971-), 女, 甘肃永靖人, 兰州理工大学硕士生导师, 主要研究方向为供应链管理。



冯涛 (1970-), 男, 甘肃临洮人, 博士, 兰州理工大学研究员、博士生导师, 主要研究方向为网络与信息安全、区块链、工业互联网安全。



苏春华 (1981-), 男, 广西钦州人, 博士, 日本会津大学高级副教授、博士生导师, 主要研究方向为密码分析、机器学习中的隐私保护技术和物联网安全与隐私。