引用格式:王庆璇,汪定. 无线传感网络中基于非抗窜扰智能卡的双因素认证协议[J]. 信息对抗技术,2022, 1(2):16-37. [WANG Qingxuan, WANG Ding. Non-tamper resistance smart-card based multi-factor authentication in wireless sensor networks[J]. Information Countermeasure Technology, 2022, 1(2):16-37. (in Chinese)]

无线传感网络中基于非抗窜扰智能卡的双因素认证协议

王庆璇[1,2],汪 定[1,2*]

(1. 南开大学网络空间安全学院,天津 300350; 2. 天津市网络与数据安全技术重点实验室(南开大学),天津 300350)

摘 要 智能卡是最为常见的密码设备之一,因其抗窜扰特性,智能卡常常被用于电子商务、医疗健康以及物联网等高安全需求的领域中为安全提供服务。随着侧信道攻击、逆向工程等技术的发展,研究表明智能卡内保存的参数可恢复,使其不再具有抗窜扰特性,因而,基于非抗窜扰智能卡假设的多因素认证协议设计得到了广泛的关注。为此,研究了在无线传感网络中典型的多因素协议,指出其不能抵抗离线口令猜测攻击和中间人攻击、无法实现双向认证,以及不能抵抗离线口令猜测攻击、无法实现用户匿名性等问题。为克服这些缺陷,在非抗窜扰智能卡假设下,结合哈希链技术提出了一类面向无线传感器网络的双因素认证协议,并在随机预言机模型中给出了严格的安全证明。与现有无线传感网络环境下多因素认证协议相比,该协议在保持较低计算开销的同时,实现了更高的安全性,适于资源受限的无线传感器网络环境。

关键词 双因素认证协议;无线传感器网络;非抗窜扰智能卡;离线口令猜测攻击;匿名性

中图分类号 TP 311 **文献标志码** A **文章编号** 2097-163X(2022)02-0016-22

DOI 10.12399/j.issn.2097-163x.2022.02.002

Non-tamper resistance smart-card based two-factor authentication in wireless sensor networks

WANG Qingxuan[1,2], WANG Ding[1,2*]

(1. College of Cyber Science, Nankai University, Tianjin 300350, China;
2. National Engineering Laboratory of Mobile Network Security (Nankai University), Tianjin 300350, China)

Abstract Smart-card is one of the most common cryptographic devices. Due to its tamper resistance characteristics, it has been widely used in many security-critical areas, such as e-commerce, medical health, and Internet of Things (IoT). However, with the development of side-channel attacks and reverse engineering, research shows that the parameters in the smart-card can be extracted, and the smart-card has no longer tamper resistance. Therefore, how to design multi-factor authentication protocols based on the non-tamper resistance smart-card has got a lot of attention. This paper studied two typical multi-factor authentication protocols in wireless sensor networks, pointing out that one cannot resist offline password guess-

收稿日期:2022-03-09 **修回日期**:2022-04-14

通信作者:汪定,E-mail:wangding@nankai.edu.cn

作者简介:王庆璇(1995—),男,博士生,主要研究方向为应用密码学、多因素身份认证协议; 汪定(1985—),男,教授,博士研究生导师,研究方向为数字身份安全

基金项目:国家自然科学基金资助项目(62172240);京津冀基础研究合作专项项目(S22ZX08013)

ing attacks and man-in-the-middle attacks, and fails to achieve mutual autentication, while the other cannot resist offline password-guessing attacks and fails to provide anonymity. In order to overcome these weaknesses, combined with the Hash-chain technique, a new non-tamper resistance smart-card based two-factor authentication scheme was proposed and formally proved secure under the random oracle model. Compared with other related multi-factor authentication schemes in the wireless sensor network environment, the proposed scheme could achieve higher security while maintaining lower computation cost. Consequently, it could be suitable for resource constrained wireless sensor network environment.

Keywords　two-factor authentication protocol; wireless sensor networks; non-tamper resistance smart-card; offline password guessing attack; user anonymity

0　引言

物联网(internet of things, IoT)由各种不同功能的物理感知设备和控制器组成，这些设备通过无线传感网络(wireless sensor networks, WSNs)相互连接并提供各种各样的服务，如实时监控、数据获取和分析[1]等。基于物联网的服务近年来迅猛发展，到 2020 年物联网设备的数量已接近 700 亿[2]。物联网应用广泛分布在人们的日常生活中，如医疗健康、智能家居、电子商务等；同样，物联网在工业领域也做出了巨大的贡献，如工业物联网(industrial IoT)或工业 4.0 等[3-4]。在这些应用中，无论是工业物联网还是医疗物联网，都是安全攸关的基础设施。然而，IoT 设备收集到的数据往往通过公开信道传输，这就意味着敌手有机会窃取或篡改公开信道中的数据。同时，未经授权的用户可能会通过非法访问传感设备来获取 IoT 设备收集的实时数据。这些情况都为无线传感网络中的安全和隐私保护带来了挑战。

身份认证是实现无线传感网络安全和隐私保护的一种有效手段，它通常包含 3 个基本要素：所知、所有和所是。所知，即个人所知道或掌握的知识，如口令等；所有，即可用以证明个人身份的物品，如身份证、信用卡等；所是，即个人所具有的生物特征，如指纹、虹膜等。运用 2 种及以上要素进行用户身份认证的协议被称为多因素认证协议。如图 1 所示，WSNs 环境下多因素身份认证协议一般包含 3 类实体：用户(user)、网关节点(gateway node, GWN)以及大量传感节点(sensor node, SN)。面向无线传感网络的认证协议面临着两方面的挑战：一方面，传感节点是资源受限设备[5]，无法承担复杂密码算法带来的计算和存储开销；另一方面，WSNs 环境的开放性和应用的敏感性对应用在其中的身份认证协议提出了一系列严苛的要求[6-7]，如抗节点捕获攻击、抗智能卡丢失攻击等，需要实现用户匿名性和前向安全性。

近 30 年关于认证协议的研究显示，设计一个安全高效的单因素口令认证协议是困难的[8-10]；近 20 年关于多因素认证协议的艰难探索表明，设计安全高效的多因素认证协议更具有挑战性[11-13]。在多因素认证协议研究初期，大多数基于智能卡的多因素认证协议采用智能卡抗窜扰假设[14-16]，即存储在智能卡中的数据是安全的。然而随着侧信道技术的发展，智能卡内的数据可以被敌手提取出来[17-18]。侧信道攻击是一种密码分析攻击，通过侧信道攻击，敌手可以利用密码系统实现的物理环境(如能耗[19]、电磁[20]、温度[21])来恢复一些数据以实现窃取秘密。尽管目前已有一些侧信道攻击的防御方法，如掩蔽[22]和改组[23]已经被应用于安全产品中，但仍有新的智能卡攻击方法不断出现。例如 2019 年，Carbone 等[24]介绍了一种攻击方法，能够攻击在配备了遮蔽模数、指数等经典侧信道攻击防御方法的处理器上实现的 RSA 算法；2021 年，Roche 等[25]成功地克隆了谷歌安全产品 Titan 的合法密钥。上述事实均表明，存储在安全产品(例如，智能卡或硬件设备)中的秘密参数不再是无条件安全的。因此，在智能卡非抗篡改假设下设计协议是必要和

现实的。此外，在 WSNs 环境下，传感节点处于无人监管的状态，易被敌手捕获。综上，WSNs 中的多因素身份认证协议设计面临智能卡参数泄露、传感节点捕获等新型风险。这意味着 WSNs 环境下，基于非抗窜扰智能卡假设的多因素认证协议具有更高的设计难度。

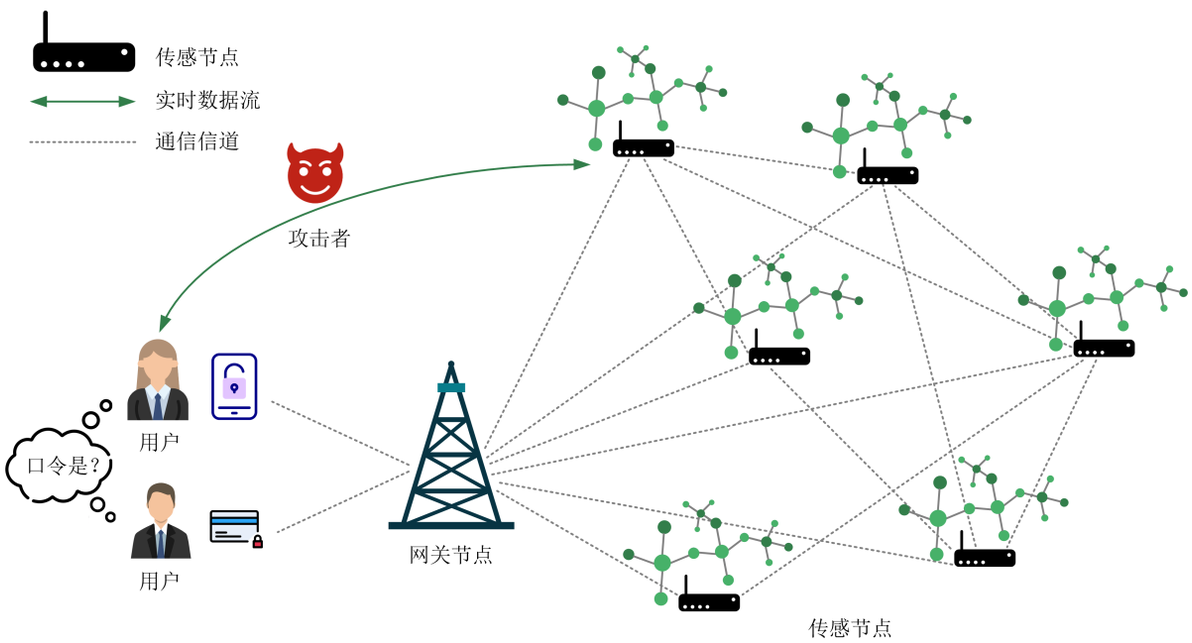

图 1 无线传感网络环境中的用户身份认证

Fig. 1 User authentication in the wireless sensor networks

1991 年，由 Chang 和 Wu 提出了第 1 个基于智能卡的口令认证协议[26]，这是在多因素认证领域内的一项开创性工作。在这之后，大量相关工作[27-29]被提出，其中 Das 等[28]在 2009 年首次提出了 WSNs 环境下基于智能卡的口令认证协议。2011 年前，基于智能卡的协议设计往往是在智能卡抗窜扰假设（即保存在智能卡中的参数是安全的）下进行的；2011 年以后，考虑到侧信道攻击，设计者们认为该假设已不足以刻画敌手的真实能力[7]。因此，智能卡非抗窜扰假设逐渐被采用。2013 年，Xue 等[30]在智能卡非抗窜扰假设的基础上提出了一种基于临时凭证的身份认证机制：即 GWN 借助口令认证向每个用户和传感节点分发临时认证凭据。随后，在此方案基础上，许多改进工作[31-33]被提出。但是，这些方案都无法实现前向安全性。2019 年，Yang 等[34]提出了一个基于动态认证凭据的认证密钥协商协议，声称此协议可以实现完美的前向安全性。然而，本文将证明该机制易遭受离线口令猜测攻击，且不能提供有效的用户匿名性。

2021 年，Li 等[35]提出了一种适用于医疗物联网的认证密钥协商机制，并指出现存的大多数认证密钥协商机制存在以下 3 个问题：(1) 不能抵抗常见的已知攻击，如节点捕获攻击、中间人攻击和传感节点仿冒攻击等；(2) 现有协议在注册阶段往往需要在安全的信道上进行；(3) 多数协议不具备轻量性，不能很好地适应资源限制的物联网设备。Li 等[35]称其协议可以解决上述 3 个问题，然而我们发现 Li 等的机制中仍然存在安全缺陷，如无法抵抗离线口令猜测攻击和中间人攻击，且没有实现双向认证。

在多因素身份认证协议领域，一直存在着设计者与攻击者之间的较量，对该领域研究的节奏明显呈现出一种“提出—攻击—再提出—再攻击”的循环态势。由于 WSNs 环境下的多因素身份认证协议面临着比通用环境中更严峻的挑战，因此摆在设计者面前的是一系列更为严苛的安全需求。为了具体刻画这些安全需求，Wang 等[7]提出了 12 条 WSNs 环境下的协议评价指标。该套评价指标有助于本文指出新近提出的 WSNs 环境下多因素身份认证协议的设计

缺陷，避免在设计新协议时引入“老生常谈”的安全缺陷。此外，由于应用在无线传感网络中的传感设备受到计算和存储资源不足的约束，一些设计者试图仅使用对称加密和异或运算等轻量性操作设计多因素身份认证协议[34-36]。然而 Ma 等[37]指出：在智能卡非抗窜扰假设下的多因素身份认证协议中，公钥密码学是实现前向安全性以及抵抗离线口令猜测攻击不可或缺的组件。进一步，王晨宇等[38]通过严格证明的方式指出了公钥密码学是实现用户匿名性的必要条件。

基于这些研究，王晨宇等[39]提出了一个基于椭圆曲线(ECC)的面向 WSNs 环境的多因素认证协议。经分析，该协议满足 Wang 等[7]提出的 12 条 WSNs 环境下的协议设计评价指标(详见 1.2 节)。尽管该协议总体运行效率较高，但仍需要在资源有限的物联网设备上运行计算开销较大的模幂运算。这也提出了一个挑战性的问题：如何在不违背 Ma 等[37]协议设计原则的前提下，设计一个具备多因素安全的协议使其在满足 Wang 等[7]12 条协议评价标准的同时进一步提升效率，减少物联网设备的计算负担。

为了解决这一问题，本文对王晨宇等[39]协议进行了分析，发现其协议主要在传感节点端通过使用 ECC 来实现前向安全性(包含计算开销较大的椭圆曲线点乘运算)。因此，解决上述问题的关键在于：如何在不影响前向安全性的条件下，在传感节点端减少或不使用公钥密码运算。本文在对 Yang 等[32]协议进行分析时，受到其动态认证凭据(其本质是哈希链技术)的启发，将 Ma 等协议设计原则中所必须的公钥密码学组件转移至计算资源相互对充分的 GWN 处，在 SN 端使用开销较小的哈希链技术来实现前向安全性，从而成功地解决了上述问题。

1　攻击者模型、评价指标及哈希链技术

本节介绍 WSNs 环境下多因素身份认证协议中潜在的安全模型(即敌手能力)、使用的评价指标以及哈希链技术。本文所使用到的符号及其含义如表 1 所示。

表 1　本文中使用的符号及其含义

Tab. 1　Symbols used in this article and their definitions

符号	含义
GWN	网关节点
U_i	第 i 个用户
S_j	第 j 个传感节点
I_{ID_i}	用户 U_i 的用户名
P_{PW_i}	用户 U_i 的口令
$X_{\mathrm{GWN}\text{-}S_j}$	传感节点 S_j 的密钥
S_{SID_j}	传感节点 S_j 的标识
X_{GWN}	网关 GWN 的长期私钥
a_{I}	由 GWN 和实体 I 共享的动态认证凭据
a_{I}^{gw}	由 GWN 存储的动态认证凭据
$H(\cdot)$	哈希函数
$\oplus$	按位异或运算
$\|$	比特连接操作
$U_i \rightarrow S_j : M$	U_i 将消息 M 通过公开信道发送给 S_j
$U_i \Rightarrow S_j : M$	U_i 将消息 M 通过安全信道发送给 S_j

1.1　攻击者模型

本文主要考虑在被广泛接受的安全模型[7,12]中使用的敌手能力，WSNs 环境下多因素身份认证协议的攻击者具备如下能力：(1) 根据标准的 Dolev-Yao 威胁模型[40]，攻击者对用户 U_i 与网关节点 GWN 与传感节点 S_j 之间的公开信道享有完全的掌控权，即能够任意窃听、拦截或修改在公开信道中传输的消息(A1)。(2) 攻击者能够以离线枚举的方式，遍历用户口令空间 D_{PW} 和身份标识空间 D_{ID} 中的所有元素(A2)。有两方面原因使得这一假设是合理的：一是身份标识不是秘密且通常易被获取[41]；二是为了方便记忆，用户倾向于选择低熵的口令[42]。(3) 通过恶意的读卡器，攻击者可以获取用户键入的口令 P_{PW_i}；通过侧信道攻击[43]，攻击者可以获取用户丢失的智能卡中存储的认证参数(A3)。但是这两种情况不能同时发生，否则将没有协议能够防御此类攻击者的侵害。(4) 当判断协议是否具备前向安全性时，可以假设攻击者已经持有 GWN 的长期私钥[7](A4)。(5) 由于传感设备通常部署在无人监管的环境中，攻击者能够获取存储在传感节点中的数据[7](A5)。本文采用的安全模型假设网关节点 GWN 是可信的，它不会冒充用户 U_i 或传感节点 S_j，且存储在其数据库中的数据不会被攻击者得到。

1.2 评价指标

Wang 等[7]针对 WSNs 环境下多因素身份认证提出了一套由 12 条具体要求组成的协议评价指标(如表 2 所示),此套评价指标基于 Wang-Wang[12]针对通用环境下双因素认证协议的评价指标提出,该通用环境下的协议评价指标被广泛引用并评价为"一个重大的突破"[44]。因此,本文将这一指标作为协议设计的组成模块。

表 2 Wang 等提出的 WSNs 环境下多因素协议设计评价指标

Tab. 2 Evaluation criteria for multi-factor authentication schemes in WSNs proposed by Wang et al.

指标	含义
C1	无口令验证表:在 GWN 等除去用户智能卡以外的位置中,都不应该直接存储用户口令或包含用户口令的值
C2	口令易用性:用户能够自由地选择口令
C3	口令无暴露风险:即使是网关节点的特权管理员也无法提取或计算用户的口令
C4	免疫智能卡丢失攻击:敌手无法通过受害者智能卡中的信息来获得其口令甚至冒充受害者
C5	免疫已知攻击:协议能够免疫离线口令猜测攻击、重放攻击、中间人攻击、节点捕获攻击等
C6	健全的修复机制:支持在不改变用户名的情况下注销丢失的智能卡;对于传感节点,应支持新增节点
C7	提供密钥协商:协议应为传感节点和用户之间建立共享的会话密钥
C8	无时钟同步机制:为了防止去同步攻击[45]并降低传输时延的影响,消息的发送与接收方不需要维持时钟同步
C9	即时拼写检测:智能卡应及时检测并提示用户由于输入错误导致的认证失败,以免浪费资源
C10	双向认证:参与通信的各方之间应双向认证
C11	用户匿名性:协议应提供用户匿名性,且实现用户行为的不可追跟踪性
C12	前向安全性:获取了 GWN 长期私钥的攻击者无法计算此前使用过的会话密钥

1.3 哈希链技术

哈希链是一种基本的密码学方法,最早由 Lamport[46]提出,被用来防止用户口令被公开信道上的敌手窃取。由于哈希链技术计算开销较低、应用成本小,其可被用在一次性口令机制、微支付机制、以及射频识别(RFID)认证机制中[47]。长度为 N 的哈希链表示为:

$$H^N(s)=H(H(\cdots H(s)\cdots)) \tag{1}$$

式中,$H(\cdot)$表示单向哈希函数,s 表示初始种子值。由于哈希函数本身具备的单向性,即已知 $H^N(s)$时,无法计算 $H^{N-1}(s)$;而已知 $H^{N-1}(s)$时,可以验证 $H^N(s)$的正确性。

2 代表性协议及安全性分析

本文首先对 2 个代表性多因素认证协议进行分析,即 Li 等[35]在 2021 年提出的面向医疗物联网的认证与密钥协商协议(Li 等协议)和 Yang 等[34]提出的适用于工业物联网的基于动态认证凭据框架的认证密钥协商协议(Yang 等协议)。限于篇幅,在回顾上述 2 个协议时仅保留了与文本内容关联较为密切的部分。

2.1 Li 等[35]协议

Li 等[35]指出协议的注册阶段不应依赖安全信道的假设,并在提出协议时进行了相应的设计,本文通过指出该协议在注册阶段存在中间人攻击来说明该假设的不合理性。

2.1.1 预部署阶段

该阶段是由网络管理员完成的离线阶段。管理员为每个传感节点 $S_j(1\leqslant j\leqslant m)$选取唯一身份标识 S_{SID_j} 和安全密钥 $X_{GWN\text{-}S_j}$。其中,$X_{GWN\text{-}S_j}$ 由 GWN 和 S_j 共享。然后管理员为每个 GWN 分配主密钥 X_{GWN}。

2.1.2 注册阶段

本阶段包含 2 个子阶段:用户注册阶段和传感节点注册阶段。用户注册阶段开展于用户(U_i)和 GWN 之间并且由 U_i 发起通信;传感节点注册阶段开展于 S_j 和 GWN 之间,此过程在预部署阶段完成后立即执行。此协议定义在 Z_p^* 上,GWN 选取一个大素数 p 和 Z_p^* 的生成元 g,并将 $\{p,g\}$作为系统参数进行广播。之后,GWN 选择一个随机数 x 作为其私钥,计算 $y=g^x \bmod p$ 作为其公钥。

用户注册阶段过程如下：

(1) U_i 选取用户名 I_{ID_i} 和口令 P_{PW_i}，并产生 2 个随机数 $a, r_i \in Z_p^*$。接下来 U_i 计算认证信息 $(A_1, A_2) = (g^a, y^a) \bmod p$、掩盖的用户名 $M_{I_i} = I_{\mathrm{ID}_i} \oplus H(T_1 \| A_1 \| A_2 \| '000')$、消息 $M_{\mathrm{PI}_i} = H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i) \oplus H(T_1 \| A_1 \| A_2)$ 以及验证信息 $V_{U_i} = H(T_1 \| A_1 \| A_2 \| H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i) \| I_{\mathrm{ID}_i})$，其中，$T_1$ 是当前的时间戳。然后 U_i 通过公开信道将消息 $\{V_{U_i}, M_{\mathrm{PI}_i}, M_{I_i}, A_1, T_1\}$ 传输给 GWN。

(2) 在收到 U_i 发送来的注册信息之后，GWN 首先通过判断 $|T_1 - T_c| < \Delta T$ 来确认消息的新鲜性。如果 T_1 是过时的消息，GWN 将会拒绝此请求；否则，GWN 将分别计算 $A_2 = A_1^x = g^{ax} \bmod p$，$H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i)^* = M_{\mathrm{PI}_i} \oplus H(T_1 \| A_1 \| A_2)$ 和 $I_{\mathrm{ID}_i}^* = M_{I_i} \oplus H(T_1 \| A_1 \| A_2 \| '000')$。然后，GWN 检查 $V_{U_i} = H(T_1 \| A_1 \| A_2 \| H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i)^* \| I_{\mathrm{ID}_i}^*)$ 是否成立。若成立，GWN 计算 $\omega_i = H(A_1 \| X_{\mathrm{GWN}}) \oplus H(T_2 \| A_1 \| A_2)$ 和验证信息 $V_G = H(T_2 \| A_1 \| A_2 \| H(A_1 \| X_{\mathrm{GWN}}))$。若上述验证失败，则 GWN 终止用户注册请求。最后，GWN 发送信息 $\{\omega_i, V_G, T_2\}$ 给 U_i。

(3) U_i 收到智能卡 SC 和消息 $\{\omega_i, V_G, T_2\}$ 后首先通过判断 $|T_2 - T_c| < \Delta T$ 来确认消息的新鲜性。如果 T_2 是过时的消息，则 U_i 拒绝处理。否则，U_i 恢复 $H(A_1 \| X_{\mathrm{GWN}})^* = \omega_i \oplus H(T_2 \| A_1 \| A_2)$ 并且检查收到的验证信息 $V_G = H(T_2 \| A_1 \| A_2 \| H(A_1 \| X_{\mathrm{GWN}})^*)$。如果成立，$U_i$ 计算秘密信息 (f_i, g_i, z_i, k_i)，其中，$f_i = H(H(A_1 \| X_{\mathrm{GWN}})^* \| I_{\mathrm{ID}_i})$，$g_i = H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| A_1)$，$z_i = f_i \oplus H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| g_i \| A_1)$ 和 $k_i = H(A_1 \| X_{\mathrm{GWN}})^* \oplus H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| z_i \| A_1)$，并在智能卡 SC 中存储 r_i, z_i, g_i, k_i, A_1，之后向 GWN 发送确认信息 CI。

(4) 在收到确认信息 CI 之后，GWN 清除了 U_i 的数据(如 I_{ID_i})并且完成了用户的注册。

传感节点注册阶段过程如下：

(1) S_j 选取 2 个随机值 $b, r_j \in Z_p^*$ 并且计算验证信息 $(B_1 = g^b, B_2 = y^b = g^{bx})$、$M_{\mathrm{SID}_j} = S_{\mathrm{SID}_j} \oplus H(B_1 \| B_2 \| T_1)$、$r_j$ 的掩盖信息 $M_{R_j} = r_j \oplus X_{\mathrm{GWN\text{-}S}_j}$ 以及验证信息 $V_{S_j} = H(X_{\mathrm{GWN\text{-}S}_j} \| B_1 \| B_2 \| r_j \| S_{\mathrm{SID}_j} \| T_1)$。其中，$T_1$ 是当前的时间戳。最后 S_j 通过公开信道将消息 $\{V_{S_j}, M_{\mathrm{SID}_j}, M_{R_j}, B_1, T_1\}$ 传输给 GWN。

(2) 在收到 S_j 发送来的注册信息之后，GWN 首先通过判断 $|T_1 - T_c| < \Delta T$ 来确认消息的新鲜性。若 T_1 是过时的消息，则 GWN 拒绝此请求；否则，GWN 计算 $B_2 = B_1^x = g^{bx} \bmod p$，$S_{\mathrm{SID}_j}^* = M_{\mathrm{SID}_j} \oplus H(B_1 \| B_2 \| T_1)$ 并且在记录列表中搜索可用的 S_{SID_j}。如果计算出来的 $S_{\mathrm{SID}_j}^*$ 不在此列表中，GWN 将会丢弃此注册请求；当 $S_{\mathrm{SID}_j}^*$ 为合法时，GWN 计算 $r_j^* = M_{R_j} \oplus X_{\mathrm{GWN\text{-}S}_j}$ 并检查等式 $V_{S_j} = H(X_{\mathrm{GWN\text{-}S}_j} \| B_1 \| B_2 \| r_j^* \| S_{\mathrm{SID}_j}^* \| T_1)$ 是否成立。若不成立，GWN 拒绝注册请求；若成立，GWN 计算秘密信息 $l_j = H(X_{\mathrm{GWN}} \| S_{\mathrm{SID}_j}^*)$、掩盖信息 $m_j = l_j \oplus H(X_{\mathrm{GWN\text{-}S}_j} \| B_1 \| B_2 \| T_2)$ 和验证信息 $V_{n_j} = H(l_j \| X_{\mathrm{GWN\text{-}S}_j} \| S_{\mathrm{SID}_j}^* \| B_1 \| B_2 \| T_2)$。最后，GWN 发送信息 $\{m_j, V_{n_j}, T_2\}$ 给对应的传感节点 S_j。

(3) S_j 收到 $\{m_j, V_{n_j}, T_2\}$ 后首先通过判断 $|T_2 - T_c| < \Delta T$ 来确认消息的新鲜性。若 T_2 已过期，则 U_i 拒绝处理此条消息。否则，S_j 恢复秘密消息 $l_j^* = m_j \oplus H(X_{\mathrm{GWN\text{-}S}_j} \| B_1 \| B_2 \| T_2)$，检查验证信息 $V_{n_j} = H(l_j^* \| X_{\mathrm{GWN\text{-}S}_j} \| S_{\mathrm{SID}_j} \| B_1 \| B_2 \| T_2)$ 是否成立。若成立，S_j 存储 l_j^*，删除 $X_{\mathrm{GWN\text{-}S}_j}$，之后向 GWN 发送确认信息 CI。

(4) 在收到确认信息 CI 之后，GWN 在其内存中清除了 S_j 的数据(如，S_{SID_j} 和 $X_{\mathrm{GWN\text{-}S}_j}$)。

2.1.3　登录阶段

(1) 用户 U_i 将智能卡插入终端媒介，输入在注册阶段确定的 $I_{\mathrm{ID}_i}^*$ 和 $P_{\mathrm{PW}_i}^*$。

(2) 智能卡 SC 计算并验证 $g_i = H(P_{\mathrm{PW}_i}^* \| I_{\mathrm{ID}_i}^* \| A_1)$ 是否正确，如果验证通过，SC 便可确保 U_i 是一个合法的用户。

2.1.4　认证与会话密钥协商阶段

本阶段，U_i，S_j 和 GWN 会认证彼此的身份并且通过以下步骤在 U_i 和 S_j 之间建立会话密钥 S_{SK}。

(1) SC 验证过 U_i 的身份之后，SC 计算 $f_i^* = z_i \oplus H(P_{\mathrm{PW}_i}^* \| I_{\mathrm{ID}_i}^* \| g_i \| A_1)$、$H(A_1 \| X_{\mathrm{GWN}}) = k_i \oplus H(P_{\mathrm{PW}_i}^* \| I_{\mathrm{ID}_i}^* \| z_i \| A_1)$，参数 $M_{\mathrm{ID}_i} = I_{\mathrm{ID}_i}^* \oplus H(H(A_1 \| X_{\mathrm{GWN}}) \| T_1)$ 和 $M_{K_i} = K_i \oplus H(M_{\mathrm{ID}_i} \| f_i^* \| T_1)$ 以及验证信息 $V_i = H(M_{\mathrm{ID}_i} \| M_{K_i} \| K_i \| I_{\mathrm{ID}_i}^* \| T_1)$。其中，$T_1$ 是当前的时间戳，K_i 是智

能卡选择的随机数。最后，U_i 发送$\{M_{ID_i}, M_{K_i}, V_i, A_1, T_1\}$给 GWN。

(2) 当从 U_i 处收到消息$\{M_{ID_i}, M_{K_i}, V_i, A_1, T_1\}$后，GWN 首先通过判断$|T_1-T_c|<\Delta T$ 成立来确定消息的新鲜性。若不成立，GWN 拒绝此登录请求；如成立，GWN 计算 $H(A_1 \| X_{GWN})$，$I_{ID_i}^* = M_{ID_i} \oplus H(H(A_1 \| X_{GWN}) \| T_1)$，$f_i^* = H(H(A_1 \| X_{GWN})^* \| I_{ID_i}^*)$，$K_i^* = M_{K_i} \oplus H(M_{ID_i} \| f_i^* \| T_1)$并且检查等式 $V_i = H(M_{ID_i} \| M_{K_i} \| K_i^* \| I_{ID_i}^* \| T_1)$是否成立，若成立，GWN 计算秘密信息 $l_j = H(X_{GWN\text{-}S_j} \| S_{SID_j})$，$M_{ID_{GWN}} = I_{ID_i}^* \oplus H(M_{ID_i} \| l_j \| T_1 \| T_2)$，$M_{K_{GWN}} = K_i^* \oplus H(l_j \| T_1 \| T_2)$以及验证信息 $V_{GWN} = H(l_j \| K_i^* \| I_{ID_i}^* \| T_1 \| T_2)$，最后 GWN 发送消息$\{M_{ID_{GWN}}, M_{K_{GWN}}, V_{GWN}, M_{ID_i}, T_1, T_2\}$给传感节点 S_j。

(3) 当从 GWN 处收到消息$\{M_{ID_{GWN}}, M_{K_{GWN}}, V_{GWN}, M_{ID_i}, T_1, T_2\}$后，$S_j$ 首先检查 T_2 的新鲜性。如果 T_2 是新鲜的 S_j，计算 $I_{ID_i}^* = M_{ID_{GWN}} \oplus H(M_{ID_i} \| l_j \| T_1 \| T_2)$，$K_i^* = M_{K_{GWN}} \oplus H(l_j \| T_1 \| T_2)$，然后检查 $V_{GWN} = H(l_j \| K_i^* \| I_{ID_i}^* \| T_1 \| T_2)$是否成立。若不成立，$S_j$ 终止此会话；若成立，S_j 选取随机值 K_j 并计算 $K_{SK_{ij}} = H((K_i^* \oplus K_j) \| I_{ID_i}^* \| S_{SID_j})$，$M_{SID_j} = S_{SID_j} \oplus H(K_i^* \| I_{ID_i}^* \| T_3)$，$M_{K_j} = K_j \oplus H(K_i^* \| I_{ID_i}^* \| T_2 \| T_3)$和验证信息 $V_j = H(K_j \| K_i^* \| I_{ID_i}^* \| S_{SID_j} \| T_2 \| T_3)$，最后 S_j 发送$\{M_{SID_j}, M_{K_j}, V_j, T_2, T_3\}$给 U_i。

(4) 收到 S_j 发来的消息后，U_i 检验$|T_3-T_c|<\Delta T$ 消息的新鲜性，如果不具备新鲜性，则 U_i 拒绝此消息。否则，U_i 计算 $K_j^* = M_{K_j} \oplus H(K_i \| I_{ID_i} \| T_2 \| T_3)$，$S_{SID_j}^* = M_{SID_j} \oplus H(K_i \| I_{ID_i} \| T_3)$，并检验 $V_j = H(K_j^* \| K_i \| I_{ID_i} \| S_{SID_j}^* \| T_2 \| T_3)$是否成立。若不成立，$U_i$ 终止会话；若成立，用户 U_i 计算与传感节点 S_j 的共享会话密钥 $K_{SK_{ij}} = H((K_i \oplus K_j^*) \| I_{ID_i} \| S_{SID_j}^*)$。

2.2 Li 等[35]协议安全性分析

对 Li 等[35]协议进行密码分析，结果表明，在抗已知攻击性方面，Li 等[33]协议存在离线口令猜测攻击、中间人攻击；在重要安全属性方面，Li 等[35]协议没有实现双向认证。

2.2.1 离线口令猜测攻击 I

在非抗窜扰智能卡的假设下，敌手可以通过侧信道攻击技术获得用户智能卡内保存的秘密信息$\{r_i, Z_i, g_i, k_i, A_1, g, y, p\}$，按照如下步骤发动离线口令猜测攻击：

步骤 1 敌手从用户身份空间 D_{ID} 和口令空间 D_{PW} 选择一对$(I_{ID_i}^*, P_{PW_i}^*)$作为猜测；

步骤 2 敌手计算 $g_i^* = H(P_{PW_i}^* \| I_{ID_i}^* \| A_1)$，其中 A_1 从智能卡中获得；

步骤 3 敌手将计算出的 g_i^* 与从智能卡中提取的 g_i 进行比较，若 $g_i^* \neq g_i$，则敌手跳转至步骤 1 直至 $g_i^* = g_i$ 成立。

实际上，为了便于用户使用，系统一般会允许其自行选择口令与用户名，这使得本文对敌手能力的假设 A2 是符合实际的。此外，用户选择的用户名和口令往往是低熵的，因此，敌手可以在多项式时间内以离线的方式，穷举空间$|D_{ID} \times D_{PW}|$(空间大小通常为 $2^{20} \times 2^{20}$[42])中所有的(I_{ID_i}, P_{PW_i})对。需要指出的是，不管系统中口令每个数字或字母、特殊符号的编码占几个字节(比如，某口令包含 8 个字符，通常占 8 个字节)，口令的有效猜测空间都是既定的，与编码方式无关。敌手发动此攻击需要的运行时间为 $O(T_H \times |D_{ID}| \times |D_{PW}|)$，其中$|D_{ID}|$表示用户名空间中用户名数量，$|D_{PW}|$表示口令空间中的口令数量，$T_H$ 表示执行一次 Hash 操作需要的时间。由于$|D_{ID}|$与$|D_{PW}|$是有限的，因此对口令认证协议进行离线口令猜测攻击是有实际意义的。

2.2.2 离线口令猜测攻击 II

除 2.2.1 节中展示的方式外，敌手还可以使用另一种方式对 Li 等[35]协议进行离线口令猜测攻击。类似地，敌手在获得智能卡中的信息$\{r_i, Z_i, g_i, k_i, A_1, g, y, p\}$之后，通过对公开信道的窃听，获取了用户 U_i 的登录请求$\{M_{ID}, M_{K_i}, V_i, A_1, T_1\}$，然后通过以下步骤开展对 U_i 的离线口令猜测攻击：

步骤 1 敌手从用户身份空间 D_{ID} 和口令空间 D_{PW} 选择一对$(I_{ID_i}^*, P_{PW_i}^*)$作为猜测；

步骤 2 敌手计算 $f_i^* = z_i \oplus H(P_{PW_i}^* \| I_{ID_i}^* \| g_i \| A_1)$，其中，$z_i, g_i, A_1$ 为敌手从智能卡内提取；

步骤 3 敌手计算 $K_i^* = M_{K_i} \oplus H(M_{ID_i} \| f_i^* \| T_1)$，其中参数 M_{K_i}, M_{ID_i}, T_1 通过被动监听信道获得；

步骤 4　敌手计算 $V_i^* = H(M_{\mathrm{ID}_i} \| M_{K_i} \| K_i^* \| I_{\mathrm{ID}_i}^* \| T_1)$，其中 $M_{\mathrm{ID}_i}, M_{K_i}, T_1$ 来自敌手截获的登录请求，K_i^* 来自步骤 3；

步骤 5　敌手将计算得出的 V_i^* 与从信道中截获的 V_i 相对比。若 $V_i^* \neq V_i$ 则敌手跳转至步骤 1 直至等式 $V_i^* = V_i$ 成立。

同 2.2.1 节中的攻击相似，敌手发动此攻击所需的运行时间为 $O(3T_H \times |D_{\mathrm{ID}}| \times |D_{\mathrm{PW}}|)$。在此攻击过程中，忽略敌手采用的时间开销较少的异或运算。

2.2.3　中间人攻击

Li 等[35]声称其协议的注册阶段可以在公开信道上进行。他们认为，用户及传感节点的注册阶段不应依赖于安全信道。且他们将此作为所提方案优于众多现存方案之处。然而事实并非如此，在 Li 等协议的用户注册阶段存在着严重的设计缺陷，这使得敌手可以发动中间人攻击，具体流程如下：

步骤 1　用户输入 $I_{\mathrm{ID}_i}, P_{\mathrm{PW}_i}$ 并产生随机数 $a, r_i \in Z_p^*$，接着 U_i 计算认证信息 $(A_1, M_2) = (g^a, M^a) \bmod p$。其中，$M = g^m \bmod p$ 为敌手的公钥。其通过欺骗等手段使用户 U_i 误将 M 当作是 GWN 的公钥（其真实公钥为 $y = g^x \bmod p$），之后 U_i 计算参数 $M_{I_i} = I_{\mathrm{ID}_i} \oplus H(T_1 \| A_1 \| M_2 \| '000')$，$M_{\mathrm{PI}_i} = H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i \oplus H(T_1 \| A_1 \| M_2)$ 和 $V_{U_i} = H(T_1 \| A_1 \| M_2 \| H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i) \| I_{\mathrm{ID}_i})$；

步骤 2　$U_i \rightarrow$ 敌手：$\{V_{U_i}, M_{\mathrm{PI}_i}, M_{I_i}, A_1, T_1\}$；

步骤 3　敌手收到 U_i 发送的消息后，计算 $M_2 = A_1^m = g^{am} \bmod p$，$H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i)^* = M_{\mathrm{PI}_i} \oplus H(T_1 \| A_1 \| M_2)$，$I_{\mathrm{ID}_i}^* = M_{I_i} \oplus H(T_1 \| A_1 \| M_2 \| '000')$。至此，$U_i$ 的信息已被敌手获取。接下来敌手假冒 U_i 向服务器完成注册过程。敌手计算认证消息 $(M, A_2^A) = (g^m, y^m) \bmod p$，$M_{I_i}^A = I_{\mathrm{ID}_i}^* \oplus H(T_1^A \| M \| A_2^A \| '000')$，和 $M_{\mathrm{PI}_i}^A = H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i)^*$ 以及参数 $V_{U_i}^A = H(T_1^A \| M \| A_2^A \| H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i)^* \| I_{\mathrm{ID}_i}^*)$；

步骤 4　敌手 $\rightarrow$ GWN：$\{V_{U_i}^A, M_{\mathrm{PI}_i}^A, M_{I_i}^A, M, T_1^A\}$；

步骤 5　GWN 首先验证 T_1^A 的新鲜性，接下来计算参数 $A_2^A = M^x = g^{mx} \bmod p$，$H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i)^* = M_{\mathrm{PI}_i}^A \oplus H(T_1^A \| M \| A_2^A)$，$I_{\mathrm{ID}_i}^* = M_{I_i}^A \oplus H(T_1^A \| M \| A_2^A \| '000')$ 和 $V_{U_i}^* = H(T_1^* \| M \| A_2^A \| H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| r_i)^* \| I_{\mathrm{ID}_i}^*)$。至此，敌手在保持受害者 ID 和 PW 不变的情况下“帮助”U_i 完成了注册。接着 GWN 计算 $\omega_i = H(M \| X_{\mathrm{GWN}}) \oplus H(T_2 \| M \| A_2^A)$，计算 $V_G = H(T_2 \| M \| A_2^A \| H(M \| X_{\mathrm{GWN}}))$；

步骤 6　GWN $\rightarrow$ 敌手：$\{\omega_i, V_G, T_2\}$；

步骤 7　敌手计算参数 $H(M \| X_{\mathrm{GWN}})^* = \omega_i \oplus H(T_2 \| M \| A_2^A)$，以及验证消息 $V_G^A = H(T_2^A \| A_1 \| M_2 \| H(M \| X_{\mathrm{GWN}})^*)$，$\omega_i^A = H(M \| X_{\mathrm{GWN}})^* \oplus H(T_2^A \| A_1 \| M_2)$；

步骤 8　敌手 $\rightarrow U_i$：$\{\omega_i^A, V_G^A, T_2^A\}$；

步骤 9　在收到消息之后，U_i 计算 $H(M \| X_{\mathrm{GWN}})^* = \omega_i^A \oplus H(T_2^A \| A_1 \| M_2)$，检查 $V_G = H(T_2 \| A_1 \| M_2 \| H(M \| X_{\mathrm{GWN}})^*)$ 是否成立。若成立，计算 $g_i = H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| A_1)$，$f_i = H(H(M \| X_{\mathrm{GWN}})^* \| I_{\mathrm{ID}_i})$，$Z_i = f_i \oplus H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| g_i \| A_1)$，$k_i = H(M \| X_{\mathrm{GWN}})^* \oplus H(P_{\mathrm{PW}_i} \| I_{\mathrm{ID}_i} \| Z_i \| A_1)$。最后，$U_i$ 在智能卡 SC 中存储参数 g_i, r_i, Z_i, k_i, A_1。

通过上述攻击步骤，敌手可以在用户 U_i 完全不知情的情况下发动中间人攻击。$H(A_1 \| X_{\mathrm{GWN}})^*$ 为 GWN 在用户登录阶段用于验证用户身份的主要参数。在敌手发动中间人攻击之后，前述参数变为 $H(M \| X_{\mathrm{GWN}})$。但由于 U_i 无法得知 X_{GWN} 的值，因此无法验证 $H(A_1 \| X_{\mathrm{GWN}})$ 中的 A_1 是否遭到替换。

值得注意的是，Li 等[35]协议存在中间人攻击的原因是敌手可以欺骗用户 U_i 使其将敌手的公钥 M 当作是 GWN 的公钥，这是由用户缺少对 GWN 的认证导致的。通常情况下，有 2 种方法可以防止此攻击：一种是通过数字签名、公钥证书等技术使用户能够认证公钥持有者的身份，但这往往伴随着较大的计算开销，难以在计算资源受限的 IoT 环境下应用；另一种是在注册阶段，由 GWN 将其公钥预置（即通过安全信道）在用户的设备或智能卡中。这种方法在多因素认证方案设计中较为常见[12,38]。实际上，在 Li 等[35]协议中存在一个预部署阶段，在此阶段中网络管理员为传感节点 S_j 部署身份标识 S_{SID_j} 和安全密钥 $X_{\mathrm{GWN\text{-}S}_j}$，这也是一种通过安全信道通信的方式。因此，在注册阶段不依赖安全信

道的假设没有实际意义。

2.2.4 双向认证缺失

双向认证是身份认证协议中的一个重要属性,该属性要求通信的双方均需要验证对方的身份。然而在 Li 等[35]协议中却没有实现此必要属性。首先,通过等式 $V_i=H(M_{\mathrm{ID}_i}\|M_{K_i}\|K_i^*\|I_{\mathrm{ID}_i}^*\|T_1)$是否成立来验证 U_i 的身份,但 U_i 没有验证 GWN 的身份;其次,S_j 通过等式 $V_{\mathrm{GWN}}=H(l_j\|K_i^*\|I_{\mathrm{ID}_i}^*\|T_1\|T_2)$是否成立来验证 GWN 的身份,但是 GWN 没有验证 S_j 的身份。需要注意的是,再次强调双向认证这一属性的必要性,根据我们对众多身份认证协议的分析经验,协议的每个参与方都应在进行任何密码学计算之前确认对方身份,否则该通信实体将会面临被当作预言机(注:当一个主体无意地为攻击者执行了一个密码运算时,该主体就被攻击者当作了预言机,或者称该主体为敌手提供了预言机服务。例如,中间人攻击是典型的预言机服务,产生此攻击的原因是主体未对消息来源进行充分的身份认证。)的风险。

2.3 Yang 等[34]协议

Yang 等[32]协议介绍了一种动态更新的认证机制,通过该机制可以实现完美的前向安全性。本文通过分析指出:由于未使用公钥密码技术,该协议存在离线口令猜测攻击并且没有真正实现用户匿名性。但 Yang 等[32]实现前向安全性的方法给予了本文启发。

2.3.1 用户注册阶段

U_i 和 GWN 通过安全的信道传输注册信息,具体步骤如下:

步骤 1 用户 U_i 选择用户名 I_{ID_i} 和认证密钥 $K_{\mathrm{lk}_{\mathrm{ID}_i}}$(实际上在原文中作者用此来指代用户的口令 P_{PW_i},因此在后文中我们使用 P_{PW_i} 来代替这一参数),随后 U_i 在密钥空间 K 中随机选取随机数 r_{ID_i} 并计算 $l_{\mathrm{PID}_i}=H(r_{\mathrm{ID}_i}\|P_{\mathrm{PW}_i})$,$m_{rg}=\{I_{\mathrm{ID}_i},l_{\mathrm{PID}_i}\}$;

步骤 2 $U_i\Rightarrow\mathrm{GWN}:\{m_{rg}\}$;

步骤 3 网关节点 GWN 收到来自 U_i 的注册请求后,首先在密钥空间 K 中选择随机数$(r'_{\mathrm{ID}_i},r_{k,gw})$,计算 $T_{I_i}=H(r'_{\mathrm{ID}_i}\|I_{\mathrm{ID}_i})$、$a_{\mathrm{ID}_i}^{gw}=H(T_{I_i}\|r_{k,gw})$ 和 $l_{\mathrm{sid}_i}=l_{\mathrm{PID}_i}\oplus a_{\mathrm{ID}_i}^{gw}$,接着设置 $K_\psi=\varphi$,最后 GWN 在数据库中存储相关参数 $\{T_{I_i},a_{\mathrm{ID}_i}^{gw},K_\psi\}$;

步骤 4 $\mathrm{GWN}\Rightarrow U_i:\{l_{\mathrm{sid}_i}\}$;

步骤 5 用户 U_i 在设备中保存$\{T_{I_i},l_{\mathrm{sid}_i},r_{\mathrm{ID}_i}\}$。

2.3.2 工业物联网设备注册阶段

步骤 1 对于设备 S_j,GWN 首先选择一个独特的设备名 s_{sid_j} 和一个随机数 $r'_{k,gw}$,之后计算动态认证凭据 $a_{\mathrm{sid}_j}=H(s_{\mathrm{sid}_j}\|r'_{k,gw})$,设置 $a_{\mathrm{sid}_j,1}^{gw}=a_{\mathrm{sid}_j,2}^{gw}=a_{\mathrm{sid}_j}$ 且 $a_{\mathrm{sid}_j}^{gw}=a_{\mathrm{sid}_j,1}^{gw}\|a_{\mathrm{sid}_j,2}^{gw}$,最后 GWN 存储$\{s_{\mathrm{sid}_j},a_{\mathrm{sid}_j}\}$;

步骤 2 $\mathrm{GWN}\Rightarrow S_j:\{s_{\mathrm{sid}_j},a_{\mathrm{sid}_j}\}$;

步骤 3 S_j 在设备中存储参数$\{s_{\mathrm{sid}_j},a_{\mathrm{sid}_j}\}$。

2.3.3 认证阶段

步骤 1 用户 U_i 输入用户名 I_{ID_i} 和口令 P_{PW_i} 后,首先计算 $l_{\mathrm{PID}_i}=H(r_{\mathrm{ID}_i}\|P_{\mathrm{PW}_i})$,$a_{\mathrm{ID}_i}=l_{\mathrm{SID}_i}\oplus l_{\mathrm{PID}_i}$。然后,$U_i$ 在密钥空间 K 中随机选取随机临时会话密钥材料 $K_{\mathrm{ID}_i}^s$ 和随机数 N_{ID_i}。接着,U_i 计算 $K_{\mathrm{ID}_i,1}^e=H(a_{\mathrm{ID}_i}\|N_{\mathrm{ID}_i}\|1)$,$K_{\mathrm{ID}_i,2}^e=H(a_{\mathrm{ID}_i}\|N_{\mathrm{ID}_i}\|2)$,$C_{\mathrm{ID}_i,1}=K_{\mathrm{ID}_i,1}^e\oplus s_{\mathrm{sid}_j}$,$C_{\mathrm{ID}_i,2}=K_{\mathrm{ID}_i,2}^e\oplus K_{\mathrm{ID}_i}^s$。最后,$U_i$ 计算 $T_1=N_{\mathrm{ID}_i}\|T_{I_i}\|C_{\mathrm{ID}_i,1}\|C_{\mathrm{ID}_i,2}$,$A_{\mathrm{ID}_i,1}=H(a_{\mathrm{ID}_i}\|T_1)$,$m_1=T_1\|A_{\mathrm{ID}_i,1}$;

步骤 2 $U_i\rightarrow\mathrm{GWN}:\{m_1\}$;

步骤 3 GWN 接收到来自 U_i 的消息$\{m_1\}$后,首先从密钥空间 K 中选择随机数 N_{gw}。然后,GWN 计算认证消息 $A_{\mathrm{ID}_i,1}^{gw,1}=H(a_{\mathrm{ID}_i}^{gw}\|T_1)$,$A_{\mathrm{ID}_i,1}^{gw,2}=H(H(K_\psi\|a_{\mathrm{ID}_i}^{gw})\|T_1)$。接着,GWN 会对收到的 $A_{\mathrm{ID}_i,1}$ 进行验证,当 $A_{\mathrm{ID}_i,1}^{gw,1}\neq A_{\mathrm{ID}_i,1}$ 且 $A_{\mathrm{ID}_i,1}^{gw,2}\neq A_{\mathrm{ID}_i,1}$ 时,GWN 拒绝 U_i 的登录请求。如果 $A_{\mathrm{ID}_i,1}^{gw,1}=A_{\mathrm{ID}_i,1}$ 则 $x=1$;如果 $A_{\mathrm{ID}_i,1}^{gw,2}=A_{\mathrm{ID}_i,1}$ 则 $x=2$,同时 $a_{\mathrm{ID}_i}^{gw}=H(K_\psi\|a_{\mathrm{ID}_i}^{gw})$且 GWN 从密钥空间 K 中选取随机密钥 K_ψ,然后 GWN 存储参数$\{a_{\mathrm{ID}_i}^{gw},K_\psi\}$,如果存储失败,则 GWN 同样拒绝 U_i 的登录请求;

步骤 4 GWN 计算 $K_{\mathrm{ID}_i,1}^e=H(a_{\mathrm{ID}_i}^{gw}\|N_{\mathrm{ID}_i}\|1)$,$K_{\mathrm{ID}_i,2}^e=H(a_{\mathrm{ID}_i}^{gw}\|N_{\mathrm{ID}_i}\|2)$。然后 GWN 从收到的消息中恢复 U_i 通信目标的设备编号 $s_{\mathrm{sid}_j}=K_{\mathrm{ID}_i}^e\oplus C_{\mathrm{ID}_i,1}$、临时会话密钥材料 $K_{\mathrm{ID}_i}^s=C_{\mathrm{ID}_i,2}\oplus K_{\mathrm{ID}_i,2}^e$,$T_2=N_{\mathrm{ID}_i}\|T_{I_i}\|N_{gw}$。最后 GWN 计算 $A_{gw,1}=H(a_{\mathrm{sid}_j,1}^{gw}\|s_{\mathrm{sid}_j}\|T_2)$、$A_{gw,2}=H(a_{\mathrm{sid}_j}^{gw},2\|s_{\mathrm{sid}_j}\|T_2)$和 $m_2=A_{gw,1}\|A_{gw,2}\|T_2$;

步骤 5　GWN→S_j：$\{m_2\}$。

2.4　Yang 等[34]协议安全性分析

2.4.1　离线口令猜测攻击

假设攻击者通过侧信道攻击技术获得用户设备内的敏感信息$\{T_{I_i}, l_{\mathrm{sid}_i}, r_{\mathrm{ID}_i}\}$，并通过对公开信道的窃听，获取用户 U_i 的登录请求$\{M_1\}$，然后通过以下步骤开展对 U_i 的离线口令猜测攻击：

步骤 1　敌手从用户口令空间 D_{PW} 选择一个 $P^*_{\mathrm{PW}_i}$ 作为猜测；

步骤 2　敌手计算 $l^*_{\mathrm{PID}_i}=H(r_{\mathrm{ID}_i}\|P^*_{\mathrm{PW}_i})$，其中 r_{ID_i} 从用户的设备中获得；

步骤 3　敌手计算 $a^*_{\mathrm{ID}_i}=l_{\mathrm{SID}_i}\oplus l^*_{\mathrm{PID}_i}$，其中 l_{SID_i} 从用户的设备中获得；

步骤 4　敌手计算 $A^*_{\mathrm{ID}_i,1}=H(a^*_{\mathrm{ID}_i}\|T_1)$，其中 T_1 来自公开信道中获得的消息 M_1；

步骤 5　敌手比较 $A^*_{\mathrm{ID}_i,1}$ 与 M_1 中提取的 $A_{\mathrm{ID}_i,1}$。若 $A^*_{\mathrm{ID}_i,1}\neq A_{\mathrm{ID}_i,1}$ 则 A 跳转至步骤 1，直至等式 $A^*_{\mathrm{ID}_i,1}=A_{\mathrm{ID}_i,1}$ 成立。

该攻击与 2.2.3 节描述的攻击一致，敌手发动此攻击所需的运行时间为 $O(2T_{\mathrm{H}}\cdot|D_{\mathrm{PW}}|)$。

2.4.2　匿名性失效

尽管 Yang 等[34]协议考虑到了基本层次的匿名性，即由 GWN 选择随机数并计算 $T_{I_i}=H(r'_{\mathrm{ID}_i}\|I_{\mathrm{ID}_i})$来保护用户的$\mathrm{ID}_i$，但攻击者仍可追踪用户行为，对用户的隐私构成威胁。用户向网关节点 GWN 发送登录请求消息 $M_1=T_1\|A_{\mathrm{ID}_i,1}$，GWN 在收到的登录请求并经过一系列的验证后，向传感节点 S_j 发送消息 $M_2=A_{gw,1}\|A_{gw,2}\|T_2$。其中，$T_1$ 与 T_2 中都包含参数 T_{I_i}，T_{I_i} 的计算包含一个由 GWN 选取的随机数 r_i，它用于掩盖用户的真实身份 I_{ID_i}。长期保存在 GWN 数据库及用户的设备中，与用户 U_i 直接相关且固定不变。因而敌手可通过跟踪固定参数 T_{I_i} 确定用户 U_i 与 GWN 的交互行为。故该协议无法真正实现用户匿名性。

3　非抗窜扰智能卡的双因素身份认证协议

通过对 Li 等[35]协议和 Yang 等[34]协议的分析发现：已知攻击如离线口令猜测攻击、中间人攻击等（表 3）的存在；身份认证中的重要属性如用户匿名性、双向认证等的缺失，依旧是身份认证协议设计面临的主要安全威胁。本节在总结 Li 等[35]和 Yang 等[34]协议设计缺陷的基础上，提出了一类 WSNs 环境下基于非抗窜扰智能卡的双因素身份认证协议。此协议能有效抵抗已知攻击，且具备 Li 等[35]和 Yang 等[34]协议中缺失的安全属性。所提协议的主要流程如图 2 所示，由 8 个阶段组成，分别是：系统初始化阶段、传感节点注册阶段、用户注册阶段、用户登录阶段、认证与密钥协商阶段、口令更改阶段、传感节点增加阶段以及恢复同步阶段。

表 3　需要应对的已知攻击

Tab. 3　Known attacks to be considered

攻击名称	描述	应对策略
离线口令猜测攻击	敌手在口令空间中以离线的方式选取可能的用户口令，并通过计算包含该口令的哈希值对猜测的结果进行检验	对于 2.2.1 节中的Ⅰ型口令猜测攻击，产生原因是：为了实现即时拼写检测等功能，在智能卡中保留了与口令的相关哈希值，通过使用“模糊验证”的技术，敌手无法验证精确的用户口令即可防止此类攻击 对于 2.2.2 节中的Ⅱ型口令猜测攻击，产生原因是：示证主体将与口令间接相关的值包含在发送给验证主体的消息中，敌手通过截获此消息来验证口令猜测的结果，通过使用公钥密码原语，将共享秘密包含在验证消息中可有效防止
在线口令猜测攻击	敌手在口令空间中选取可能的用户口令，通过与 GWN 进行交互并根据能否成功登录来判断猜测结果	通过“Honeywords”技术检测并限制敌手的交互次数

续表3

攻击名称	描述	应对策略
智能卡丢失攻击	敌手可能以窃取或拾取的方式获得受害者的智能卡，通过侧信道技术提取卡内的安全参数。然后敌手使用这些参数猜测受害者的口令、冒充用户身份；或者在不引起受害者注意的情况下归还其智能卡，等到下次用户正常登录时再伺机发动攻击	遗失的智能卡会为敌手提供离线口令猜测攻击的有效途径，因此需要有效防止口令猜测攻击
节点捕获攻击	敌手可以捕获有限数量的传感节点 S_j，并获取其中保存的安全参数，之后再使用获取到的信息发起攻击，破坏用户匿名性等	在 GWN 与 S_j 的认证消息中不包含任何用户的信息
重放攻击	敌手可以收集用户和 GWN 之间的过往通信记录，当下一个认证会话开始时敌手可以替换通信数据中的全部或部分内容	使用随会话变化的随机数来保持消息的新鲜性
中间人攻击	也称为平行会话攻击，敌手可以介入通信双方，先与一方进行交互；再用得到的信息与另一方交互	在进行密码学运算之前首先验证消息来源的身份，以免被敌手当作预言机
去同步攻击	对于存在时钟同步机制的协议，敌手可以通过阻断关键消息的传输从而使通信双方失去同步，进一步发动拒绝服务攻击	不使用时钟同步机制

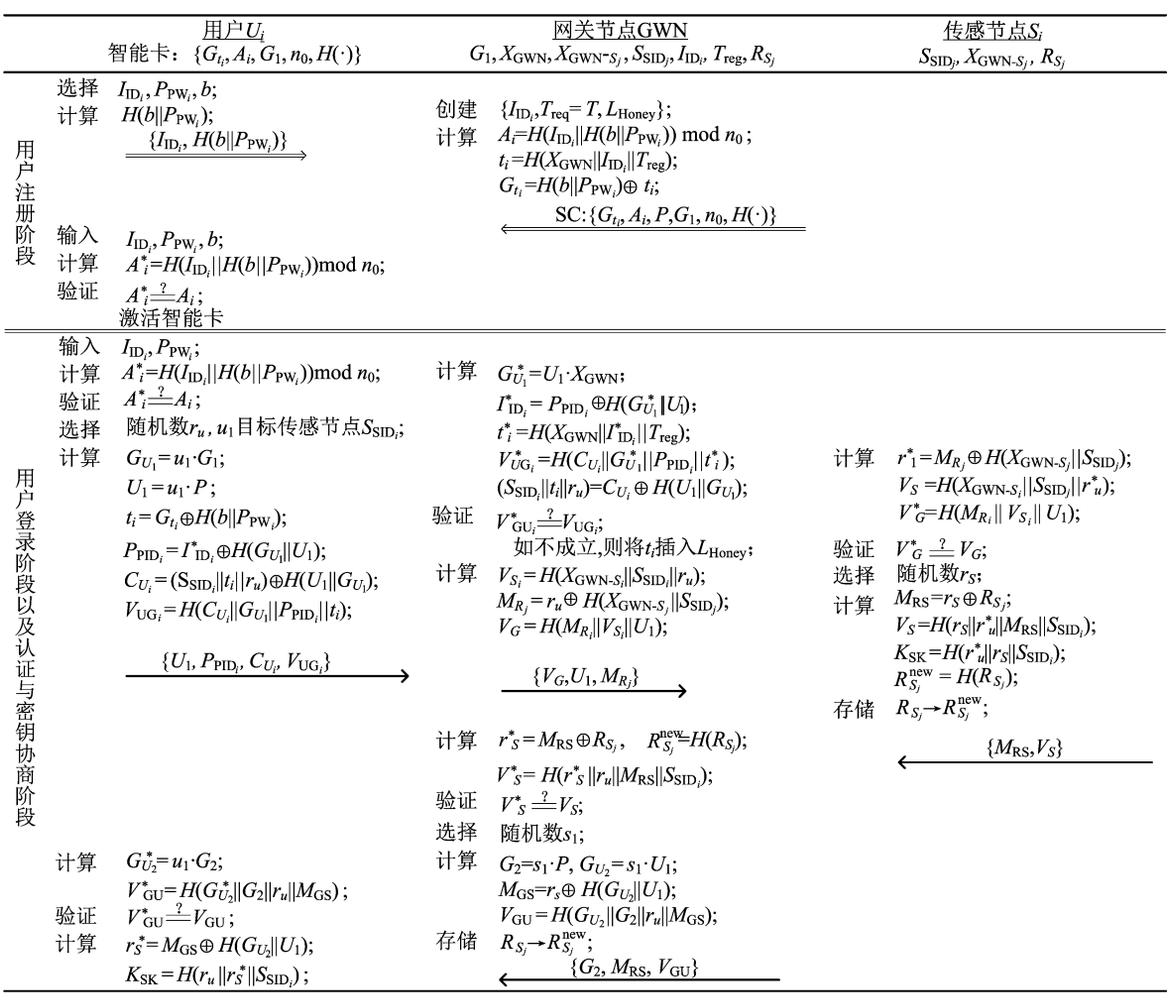

图 2 本文提出协议的主要流程

Fig. 2 The main processes of the proposed scheme

3.1 基本设计思想

对 Li 等[35]和 Yang 等[34]协议的密码分析对本文设计并提出改进的协议具有重要的指导作用。首先，Li 等[33]提出了一个全过程在公开信道上完成的身份认证方案，以此来改善目前大部分协议在注册阶段需依赖安全信道的现状。然而，本文发现该协议在注册阶段存在中间人攻击。这是由于在公开网络环境中的用户无法正确分辨 GWN 与敌手的公钥。此外，Li 等[33]协议在注册阶段之前还存在一个预部署阶段，网络管理员在该阶段中为待注册的传感节点 S_j 分配标签 S_{SID_j} 和密钥 $X_{\mathrm{GWN\text{-}S}_j}$，此过程实际上是在安全信道的环境中进行的。结合上述两点，依赖安全信道的假设是合理且必要的，因此本文提出的改进协议的注册阶段仍然在安全信道中进行。

其次，Yang 等[34]考虑 IoT 设备资源限制的特点，在他们的工作中刻意避免了使用公钥密码原语。由于 Wang 等[38]通过证明的形式指出公钥密码学是实现用户匿名性的必要条件，Yang 等[34]的这一做法引起了本文的关注，分析发现该协议存在用户匿名性缺失的问题。尽管如此，Yang 等[34]通过动态更新参数来实现前向安全性的做法给本文协议的设计提供了一些启迪，故本文在 S_j 端使用哈希链来实现前向安全性。

用户匿名性是身份认证协议不可缺少的重要属性，尽管使用公钥密码原语会在一定程度上影响协议的运行效率，但相较于保障协议的用户匿名性所获得的优势，在不影响协议可用性的前提下，做出适当性能上的牺牲是可取的。本文采用椭圆曲线 Diffie-Hellman(ECDH)密钥交换技术来提供用户匿名性，并且为了降低传感设备的计算开销，所有 ECDH 的计算都在计算资源相对宽松的用户端与 GWN 端运行。最后，离线口令猜测攻击一直以来是口令认证协议设计所面对的难题，本文分析的两个代表性协议都存在这种攻击，这一事实更加证明了这一观点。为解决这一问题，本文采用了 Wang 等[12]提出的“fuzzy-verifier(模糊验证)＋Honeywords(诱饵口令)”方法。

为了进一步确保改进协议的安全性，本文根据 Wang 等[7]提出的 12 条协议评价指标，逐条提出对应的设计思路。具体来说，在 GWN 的数据库中只保存用户的用户名(I_{ID_i})和对应的注册时间，不保存任何与其口令直接或间接相关的值(C1)。由于 GWN 中没有保存与口令相关的值，因此即使是 GWN 的特权管理员也无法获取用户口令(C3)；用户可以自行选择所使用的口令(C2)；使用模糊验证机制以及公钥密码原语可以有效地防止表 3 中提到的 2 种类型的离线口令猜测攻击，从而抵抗智能卡丢失攻击(C4)；表 3 中给出的常见的已知攻击及其相应的应对策略应落实到协议设计中(C5)；协议应具备口令更改阶段和传感节点动态增加阶段(C6)；认证阶段以用户和传感节点成功协商出一个共享会话密钥为结束的标志(C7)。使用随机数来保障消息的新鲜性，不依赖于时间同步机制(C8)；在用户键入 ID 和 PW 后，首先由智能卡检测正确后再进行后续步骤(C9)；用户与 GWN 实现双向认证，GWN 与 S_j 之间实现双向认证(C10)；采用公钥密码原语隐藏用户的真实身份标识，为了防止潜在的节点捕获攻击，GWN 不会将任何与用户有关的信息发送给 S_j(C11)；采用 ECDH 密钥交换协议和哈希链技术来保护用于构建会话密钥的关键参数，且会话密钥中包含随会话更新的随机数(C12)。

3.2 系统初始化阶段

本协议定义在有限域 F_p 上，选取 F_p 上的一条椭圆曲线 $E_p(a,b): y^2=x^3+ax+b(\mathrm{mod}\ p)$。其中，$a,b\in F_p$ 且满足 $4a^3+27b\neq 0(\mathrm{mod}\ p)$。哈希函数由 $H(\cdot):\{0,1\}^*\rightarrow\{0,1\}^l$ 表示，其中，l 表示哈希函数输出的字节长度。系统管理员选择一个常数 n_0 用于模糊验证，n_0 的值和口令空间的大小有关，通常在区间 $[2^4,2^8]$ 中。令(G_1，X_{GWN})表示 GWN 的公私钥对，其中，$G_1=X_{\mathrm{GWN}}\cdot P$，$P$ 为椭圆曲线上的一点。

3.3 传感器节点注册阶段

步骤 1　GWN 为传感节点 S_j 选择唯一的标识 S_{SID_j} 以及随机数 $X_{\mathrm{GWN\text{-}S}_j}$ 和 R_{S_j}，其中 $X_{\mathrm{GWN\text{-}S}_j}$ 作为 S_j 的密钥；

步骤 2　GWN 在数据库中存储与 S_j 相关的参数 S_{SID_j}，$X_{\mathrm{GWN\text{-}S}_j}$ 和 R_{S_j}；

步骤 3　GWN⇒S_{SN_j}:{ S_{SID_j}，$X_{\mathrm{GWN\text{-}S}_j}$，$R_{S_j}$ }；

步骤 4　S_j 存储 S_{SID_j}，$X_{\mathrm{GWN\text{-}S}_j}$ 和 R_{S_j}。

3.4 用户注册阶段

步骤 1　用户 U_i 选择用户名 I_{ID_i}，口令 P_{PW_i} 和一个随机的字符串 b，并计算 $H(b\,\|\,P_{\mathrm{PW}_i})$；

步骤 2 $U_i \Rightarrow$GWN:$\{I_{\mathrm{ID}_i}, H(b \| P_{\mathrm{PW}_i})\}$;

步骤 3 GWN 计算 $A_i = H(I_{\mathrm{ID}_i} \| H(b \| P_{\mathrm{PW}_i})) \bmod n_0$。然后在数据库中创建一个包含 $\{I_{\mathrm{ID}_i}, T_{\mathrm{reg}} = T, L_{\mathrm{Honey}}\}$ 的条目,分别用来存储用户 U_i 的用户名 I_{ID_i},注册时间 $T_{\mathrm{reg}} = T$ 以及用于检测口令猜测的列表 L_{Honey}。接下来 GWN 计算 U_i 的关键认证参数 $t_i = H(X_{\mathrm{GWN}} \| I_{\mathrm{ID}_i} \| T_{\mathrm{reg}})$,$G_{t_i} = H(b \| P_{\mathrm{PW}_i}) \oplus t_i$。最后 GWN 将参数 $\{G_{t_i}, A_i, P, G_1, n_0, H(\cdot)\}$ 写入智能卡;

步骤 4 GWN$\Rightarrow U_i$:智能卡$\{G_{t_i}, A_i, P, G_1, n_0, H(\cdot)\}$;

步骤 5 用户 U_i 收到智能卡后输入 I_{ID_i},P_{PW_i} 和字符串 b 并计算 $A_i^* = H(I_{\mathrm{ID}_i} \| H(b \| P_{\mathrm{PW}_i})) \bmod n_0$。再比较计算出的 A_i^* 与卡内存储的 A_i 是否相等,若相等则智能卡被激活并将 b 存储进卡中。

3.5 用户登录阶段

步骤 1 用户 U_i 输入 $I_{\mathrm{ID}_i}^*$ 和 $P_{\mathrm{PW}_i}^*$,智能卡计算 $A_i^* = H(I_{\mathrm{ID}_i}^* \| H(b \| P_{\mathrm{PW}_i}^*)) \bmod n_0$,并检查等式 $A_i^* \stackrel{?}{=} A_i$ 是否成立。如果不成立则终止登录进程,如果成立则继续;

步骤 2 智能卡生成随机数 r_u 和 u_1。然后选择目标传感器的标识 S_{SID_j}。接下来,智能卡首先计算 $G_{U_1} = u_1 \cdot G_1$,$U_1 = u_1 \cdot P$。然后计算 $t_i = G_{t_i} \oplus H(b \| P_{\mathrm{PW}_i})$,$P_{\mathrm{PID}_i} = I_{\mathrm{ID}_i} \oplus H(G_{U_1} \| U_1)$,$C_{U_i} = (S_{\mathrm{SID}_j} \| t_i \| r_u) \oplus H(U_1 \| G_{U_1})$ 和认证参数 $V_{\mathrm{UG}_i} = H(C_{U_i} \| G_{U_1} \| P_{\mathrm{PID}_i} \| t_i)$;

步骤 3 $U_i \rightarrow$GWN:$\{U_1, P_{\mathrm{PID}_i}, C_{U_i}, V_{\mathrm{UG}_i}\}$。

3.6 认证与密钥协商阶段

步骤 1 GWN 在收到 U_i 发送的消息 $\{U_1, P_{\mathrm{PID}_i}, C_{U_i}, V_{\mathrm{UG}_i}\}$ 后,首先计算 $G_{U_1}^* = U_1 \cdot X_{\mathrm{GWN}}$ 和 $I_{\mathrm{ID}_i}^* = P_{\mathrm{PID}_i} \oplus H(G_{U_1}^* \| U_1)$,然后根据 $I_{\mathrm{ID}_i}^*$ 在数据库中取出对应的用户注册时间 T_{reg}。若 $I_{\mathrm{ID}_i}^*$ 不在数据库中则终止登录进程;如找到对应的条目则计算 $t_i^* = H(X_{\mathrm{GWN}} \| I_{\mathrm{ID}_i}^* \| T_{\mathrm{reg}})$,$V_{\mathrm{UG}_i}^* = H(C_{U_i} \| G_{U_1}^* \| P_{\mathrm{PID}_i} \| t_i^*)$ 和参数 $(S_{\mathrm{SID}_j} \| t_i \| r_u) = C_{U_i} \oplus H(U_1 \| G_{U_1}^*)$ 并将计算出来的 $V_{\mathrm{UG}_i}^*$ 与收到的 V_{UG_i} 进行比较,若两个值不相等则 GWN 将 t_i 的值插入 L_{Honey} 随后结束登录进程。需要注意的是当 L_{Honey} 中错误的 t_i 的个数超过一定数量时系统将暂时冻结此账户,直到 U_i 重新注册;若 $V_{\mathrm{UG}_i}^* = V_{\mathrm{UG}_i}$,GWN 认证用户的身份并根据用户 U_i 选择的 S_{SID_j} 计算 $V_{S_j} = H(X_{\mathrm{GWN\text{-}S}_j} \| S_{\mathrm{SID}_j} \| r_u)$,$M_{R_j} = r_u \oplus H(X_{\mathrm{GWN\text{-}S}_j} \| S_{\mathrm{SID}_j})$,和 $V_{\mathrm{GS}} = H(M_{R_j} \| V_{S_j} \| U_1)$;

步骤 2 GWN$\rightarrow S_j$:$\{U_1, M_{R_j}, V_G\}$;

步骤 3 S_j 计算 $r_u^* = M_{R_j} \oplus H(X_{\mathrm{GWN\text{-}S}_j} \| S_{\mathrm{SID}_j})$,$V_{S_j} = H(X_{\mathrm{GWN\text{-}S}_j} \| S_{\mathrm{SID}_j} \| r_u^*)$ 和验证参数 $V_G^* = H(M_{R_j} \| V_{S_j} \| U_1)$ 并比较 V_G^* 与 V_G 是否相等。若不相等,则终止会话;否则,S_j 验证 GWN 的身份。然后 S_j 选择随机数 r_S;

步骤 4 S_j 计算 $M_{\mathrm{RS}} = r_S \oplus R_{S_j}$,$R_{S_j}^{\mathrm{new}} = H(R_{S_j})$,$V_S = H(r_S \| r_u^* \| M_{\mathrm{RS}} \| S_{\mathrm{SID}_j})$ 和会话密钥 $S_{\mathrm{SK}} = H(r_u^* \| r_S \| S_{\mathrm{SID}_j})$;

步骤 5 S_j 存储 $R_{S_j}^{\mathrm{new}}$ 并使用其替换 R_{S_j};

步骤 6 $S_j \rightarrow$GWN:$\{V_S, M_{\mathrm{RS}}\}$;

步骤 7 GWN 计算 $r_S = M_{\mathrm{RS}} \oplus R_{S_j}$,$R_{S_j}^{\mathrm{new}} = H(R_{S_j})$ 和 $V_S^* = H(r_S^* \| r_u \| M_{\mathrm{RS}} \| S_{\mathrm{SID}_j})$,然后检查计算出来的 V_S^* 与接收到的 V_S 是否相等,若不相等,则终止会话;否则,GWN 认证 S_j 的身份。然后选取随机数 s_1 并计算 $G_2 = s_1 \cdot P$,$G_{U_2} = s_1 \cdot U_1$,$M_{\mathrm{GS}} = r_S \cdot H(G_{U_2} \| U_1)$ 以及 $V_{\mathrm{GU}} = H(G_{U_2} \| G_2 \| r_u \| M_{\mathrm{GS}})$。存储 $R_{S_j}^{\mathrm{new}}$ 并使用其替换 R_{S_j};

步骤 8 GWN$\rightarrow U_i$:$\{G_2, M_{\mathrm{GS}}, V_{\mathrm{GU}}\}$;

步骤 9 U_i 计算 $G_{U_2}^* = U_1 \cdot G_2$ 和 $V_{\mathrm{GU}}^* = H(G_{U_2}^* \| G_2 \| r_u \| M_{\mathrm{GS}})$,并比较 V_{GU}^* 与 V_{GU} 是否相等。若不相等,则终止会话;否则,U_i 认证 GWN 的身份;

步骤 10 U_i 计算 $r_S^* = M_{\mathrm{GS}} \oplus H(G_{U_2} \| U_1)$ 和会话密钥 $S_{\mathrm{SK}} = H(r_u \| r_S^* \| S_{\mathrm{SID}_j})$。

3.7 口令更改阶段

步骤 1 U_i 将其智能卡插入读卡器并输入 $I_{\mathrm{ID}_i}^*$ 和 $P_{\mathrm{PW}_i}^*$;

步骤 2 智能卡计算 $A_i^* = H(I_{\mathrm{ID}_i} \| H(b \| P_{\mathrm{PW}_i})) \bmod n_0$,然后比较 A_i^* 与卡内存储的 A_i,若不相等,则终止会话;否则,智能卡要求 U_i 输入新的口令;

步骤 3 U_i 输入新的口令 $P_{\mathrm{PW}_i}^{\mathrm{new}}$ 和新的字符串 b^{new}。然后智能卡计算参数 $G_{t_i}^{\mathrm{new}} = G_{t_i} \oplus H(b \| P_{\mathrm{PW}_i}^*) \oplus H(b^{\mathrm{new}} \| P_{\mathrm{PW}_i}^{\mathrm{new}})$,$A_i^{\mathrm{new}} = H(I_{\mathrm{ID}_i} \| H(b^{\mathrm{new}} \| P_{\mathrm{PW}_i}^{\mathrm{new}})) \bmod n_0$,然后使用 $G_{t_i}^{\mathrm{new}}$,A_i^{new} 和 b^{new} 替换卡中相应参数。

3.8 传感器节点增加阶段

步骤 1 GWN 为新的传感节点 S_{new} 选择唯

一的标识 $S_{SID_{new}}$ 以及一个新的密钥 $X_{GWN\text{-}S_{new}}$ 和随机数 $R_{S_{new}}$；

步骤 2　GWN 在数据库中存储 $S_{SID_{new}}$，$X_{GWN\text{-}S_{new}}$ 以及 $R_{S_{new}}$；

步骤 3　GWN $\Rightarrow S_{new}$：$\{S_{SID_{new}}, X_{GWN\text{-}S_{new}}, R_{S_{new}}\}$；

步骤 4　S_{new} 存储 $S_{SID_{new}}$，$X_{GWN\text{-}S_{new}}$ 和 $R_{S_{new}}$。

3.9　恢复同步阶段

本协议采用了哈希链技术来保障传感节点 S_j 和 GWN 之间通信的前向安全性（即每次通信完成后 S_j 会更新参数 $R_{S_j}^{new}=H(R_{S_j})$）。虽然哈希链技术能够减少传感设备的计算开销，但是 GWN 和 S_j 之间存在失去同步的可能性，如通信中断、敌手阻断（发起去同步攻击）等。一旦此种情况发生则 S_j 无法通过 GWN 的认证。因此，通过本阶段系统能够抵抗去同步攻击并及时恢复通信，具体步骤如下：

步骤 1　系统管理员使用一个管理员账户 U_{ad} 登录系统输入 $I_{ID_{ad}}$ 和 $P_{PW_{ad}}$；

步骤 2　U_{ad} 输入发生去同步的传感设备号 S_{SID_j}，并发送正常登录请求；

步骤 3　GWN 收到消息 $\{V_S, M_{RS}\}$，由于传感节点 S_j 和 GWN 之间失去同步，因此通信会在此时中断；

步骤 4　GWN 从数据库中取出参数 R_{S_j}，然后计算 $R_{S_j}^*=H(R_{S_j})$，$r_S^*=M_{RS}\oplus R_{S_j}^*$ 和 $V_S^*=H(r_S^*\|r_{ad}\|M_{RS}\|S_{SID_j})$。其中 r_{ad} 是由系统管理员 U_{ad} 在登录时选取的随机数，因此是已知的；

步骤 5　U_{ad} 比较计算出来的 V_S^* 与接收到的 V_S 是否相等，若相等则在 GWN 的数据库中保存 $R'_{S_j}=H(R_{S_j}^*)$；否则重复步骤 4 直到等式 $V_S^*=V_S$ 成立后保存 $R'_{S_j}=H(R_{S_j}^*)$至 GWN 的数据库。

值得注意的是，本阶段虽然能够恢复传感节点 S_j 和 GWN 之间的同步，但是无法检测传感节点 S_j 和 GWN 之间失去同步现象的发生。然而在实际应用中失去同步的现象不难检测，例如通过分析系统日志的记录，如果某一个传感节点在短时间内发生多次无法通信的现象则可以推测其遭受了去同步攻击。

4　安全模型及形式化证明

4.1　安全模型

为了刻画非抗窜扰智能卡假设下，WSNs 环境中多因素认证协议面对的敌手能力，本文根据 BPR2000 模型[48]和 Wang 等[12]提出的安全模型做出如下定义：

（1）协议参与方。本文提出协议的参与方包括 3 类通信实体：即用户 $U\in$User，传感节点 $S\in$SN，以及网关节点 $G\in$GWN。使用 U^i 表示第 i 个用户实体；使用 S^j 表示第 j 个传感节点；使用 G^k 表示第 k 个网关节点。另外，使用 $I\in$User$\cup S_{SN}\cup$GWN 泛指协议参与方中的任意一个实体。

（2）口令与长期密钥。在注册阶段，系统会为每一个参与方准备必要的长期私钥和公开参数。网关节点 GWN 持有一对长期的公私钥对 (p_k,s_k)和向量$\langle S_{SID_{SN^j}}, X_{SN^j}\rangle$，其中 $S_{SID_{SN^j}}$ 表示第 j 个传感节点的标识，X_{SN^j} 表示 GWN 为 S_{SN^j} 生成的长期秘密；传感节点持有一个身份标识 $S_{SID_{SN}}$ 和一个长期秘密 X_{SN^j}；每一个用户 U 持有一个身份标识 I_{ID_U} 和一个口令 P_{PW_U}，其中，P_{PW_U} 可以看作是从一个服从 Zipf 分布[49]的小容量字典 D 中选取的一个密钥参数。$|D|$表示字典容量，它是与系统安全参数独立的常数。

（3）接受。一个实体完成了协议的运行并且生成了会话密钥，此状态被称为是接受状态。

（4）查询。敌手通过预言机查询的方式与协议的各个参与方进行交互，预言机查询刻画了在真实攻击中敌手的能力，主要有以下几种：

① Execute(U^i,G^k,S^j)。此预言机查询模拟了敌手的被动攻击，此查询的输出为协议正常运行的情况下在 U^i，G^k 和 S^j 3 个通信实体之间进行传输的数据。

② Send(I,m)。此预言机查询模拟了敌手的主动攻击。在此查询中敌手发送消息 m 给实体 I。此查询的输出为实体 I 根据协议的流程对消息 m 做出的响应。

③ Test(I)。此查询不是用于刻画敌手的能力，而是为证明过程服务的。当敌手对实体 I 进行此查询时，系统会在$\{0,1\}$中随机选取一个值 c。若 $c=1$，则实体 I 的会话密钥 S_{SK} 将作为查询输出发送给敌手；若 $c=0$，则系统选择一个与会话密钥 bit 长度相同的随机字符串作为查询输出发送给敌手。

④ Corrupt(U,1)。根据 1.1 节中敌手的能力 A3，敌手可以通过恶意的读卡器获取用户的口令，此查询用于模拟敌手的此项能力。此查询的

输出为用户 U 的口令 PW_U。

⑤ Corrupt(U,2)。在敌手的能力 A3 中,敌手还可以通过侧信道技术来获取保存在智能卡内的安全参数,此查询用于模拟敌手的此项能力。故此查询的输出为 U 的安全参数。

(5) 匹配会话。对于一组通信实体 U^i,G^k 和 S^j,使用 s_{sid} 和 p_{pid} 来表示会话标识和伙伴标识。我们称 U^i 和 S^j 是匹配会话当且仅当:① U^i 和 S^j 两个实体都处于接受状态。② 会话标识相等 $s_{\mathrm{sid}_{U^i}}=s_{\mathrm{sid}_{S^j}}=s_{\mathrm{sid}}$。③ 伙伴标识相等 $p_{\mathrm{pid}_{u^i}}=G^k=p_{\mathrm{pid}_{S^j}}$。

(6) 新鲜性。对于通信实体 I 来说,如果满足以下 2 种情况则可以称其为新鲜的:① I 处于被接受的状态并且计算出了会话密钥;② 在游戏开始之前,至多对 U 进行一种 Corrupt 查询。

(7) 语义安全性。认证协议的一个主要目的是保护会话密钥 S_{SK} 不被敌手获得,为了说明提出协议的安全性,我们需要实际衡量敌手能以多大的概率成功攻破协议 $\mathscr{P}$。为了区分真实的会话密钥和一个随机选取的等长字符串,敌手可以进行多项式次数的预言机查询从而不断增加自己的优势。经过上述过程后,敌手通过猜测 Test() 查询中的 c 来检验自己的训练成果。令事件 $S_{\mathrm{Succ}}(\mathscr{A})$ 表示敌手猜测的 $c'=c$。敌手取胜的优势定义如下:

$$\mathrm{Adv}_P^{\mathrm{AKE}}(\mathscr{A})=2P_r[S_{\mathrm{Succ}}(\mathscr{A})]-1=2P_r[c'=c]-1 \tag{1}$$

对于一个安全的口令认证协议,在线口令猜测攻击应当是敌手攻破协议所采取的最佳策略[12][50]。因此,如果对任意的多项式时间敌手 $\mathscr{A}$,存在一个可以忽略的函数 ε 使得:

$$\mathrm{Adv}_{P,D}^{\mathrm{AKE}}(A)\leqslant c'\cdot q_{\mathrm{send}}^{s'}+\varepsilon \tag{2}$$

则称本文提出的协议是语义安全的,其中 $\mathscr{D}$ 是服从 Zipf 分布[41]的口令空间,c' 和 s' 是 Zipf 参数,q_{send} 表示敌手进行的主动攻击次数。

4.2 形式化证明

本节的证明过程将通过序列游戏的方式开展,证明过程将展示敌手 $\mathscr{A}$ 无法区分参与协议的通信实体的会话密钥 S_{SK} 与随机选取的等长字符串,即敌手无法拥有比仅发起在线猜测攻击更多的优势。为了简单起见,我们暂时不考虑前向安全性的目标。

在证明开始之前,首先引入证明所需要的困难问题,椭圆曲线计算 Diffie-Hellman(ECCDH) 假设:E_p 为有限域 F_p 上的一条椭圆曲线,P 是 E_p 上一个阶为大素数 n 的点,G 是由 P 生成的循环加法群,且 $G\in F_p$。对一个多项式时间敌手 $\mathscr{A}$ 来说,有概率:

$$\mathrm{Adv}_{P,G}^{\mathrm{ECCDH}}(A)=P_r[\mathscr{A}(P,a\cdot P,b\cdot P)=ab\cdot P]\leqslant\varepsilon \tag{3}$$

其中,ε 是一个可忽略的值,a,b 是 2 个随机数。

定理 1 令 $\mathscr{P}$ 表示本文所提出的协议,$\mathscr{D}$ 表示服从 Zipf 分布[41]的口令空间。令 $\mathscr{A}$ 为一个多项式时间敌手,它将在时间 t 内进行 q_{send} 次 Send() 查询;q_{exe} 次 Execute() 查询以及 q_h 次哈希查询。若对于这样的敌手有:

$$\mathrm{Adv}_P^{\mathrm{AKE}}(\mathscr{A})=2P_r[S_{\mathrm{Succ}_6}]-1+2(P_r[S_{\mathrm{Succ}_0}]-P_r[S_{\mathrm{Succ}_7}])\leqslant c'\cdot q_{\mathrm{send}}^{s'}+\frac{(q_{\mathrm{send}}+q_{\mathrm{exe}})^2}{p}+\frac{q_h^2+6q_{\mathrm{send}}}{2^l}+6q_h\mathrm{Adv}_{P,G}^{\mathrm{ECCDH}}(t) \tag{4}$$

则称本文所提出的协议是语义安全的。

证明 令 $\mathscr{A}$ 表示一个企图攻击本文所提出协议的敌手。利用 $\mathscr{A}$ 来构建攻击本文所使用的密码学原语(如哈希函数和 ECCDH 假设)的算法,将本文的安全性规约到 ECCDH 假设上。如果存在能攻击本文所提协议语义安全的敌手,则存在能够攻击 ECCDH 假设的敌手。通过一系列连续的仿真游戏来证明定理 1,在每个游戏中,首先定义如下的事件:

(1) S_{Succ_n}。$\mathscr{A}$ 成功地猜测出 Test() 查询中由预言机选择的随机值 c。

(2) $\mathrm{AskPara}_n(A_{\mathrm{AskP}_n})$。$\mathscr{A}$ 通过哈希查询 $b\parallel P_{\mathrm{PW}_i}$ 或者 $X_{\mathrm{GWN}}\parallel I_{\mathrm{ID}_i}\parallel T_{\mathrm{reg}}$ 从而成功计算出 t_i。

(3) $\mathrm{AskAuth}_n(A_{\mathrm{AskA}_n})$。$\mathscr{A}$ 正确地计算出 t_i 并通过哈希查询正确地计算出验证因子 $\{V_{\mathrm{UG}_i},V_G,V_S,V_{\mathrm{GU}}\}$。

(4) $\mathrm{AskH}_n(A_{\mathrm{AskH}_n})$。$\mathscr{A}$ 通过哈希查询 $G_{U_1}\parallel U_1$ 或者 $G_{U_2}\parallel G_2$ 从而正确地计算出 r_u 或 r_S。

下面分别讨论仿真游戏 $\mathrm{Game}_0\sim\mathrm{Game}_8$。

(1) Game_0。此游戏对应随机预言机模型下的真实攻击,因此根据定义有:

$$\mathrm{Adv}_P^{\mathrm{AKE}}(A)=2P_r[S_{\mathrm{Succ}_0}]-1 \tag{5}$$

(2) Game_1。在此游戏中,模拟哈希查询 H

和之后在Game_6中出现的H'。系统维护 2 个列表Λ_H和Λ_A。其中Λ_H用于记录所有的哈希查询，对于以x作为输入的查询$H(x)$，预言机首先检查列表Λ_H中是否含有x，若存在则预言机将以$y=H(x)$作为输出；若不存在，则预言机选择一个随机数y作为输出，并将(x,y)存储到列表Λ_H中。列表Λ_A用于保存敌手$\mathcal{A}$进行的哈希查询。此外，我们还将其他依据真实敌手能力的预言机查询如 Send()，Execute()以及 Corrupt()等引入此游戏中。因此，此游戏与真实的攻击(Game_0)并无差别：

$$|P_r[S_{\text{Succ}_1}]-P_r[S_{\text{Succ}_0}]|=0 \tag{6}$$

(3) Game_2。在此游戏中，所有的预言机保持与Game_1中的一致，且为了方便后续分析，我们需要去除掉Game_1中碰撞发生的可能性，碰撞包含以下 2 个方面：

① 哈希输出中存在的碰撞；

② 通过信道传输的数据($\{U_1,P_{\text{PID}_i},C_{U_i},V_{\text{UG}_i}\}$，$\{U_1,M_{R_j},V_G\}$，$\{V_S,M_{\text{RS}}\}$，$\{G_2,M_{\text{GS}},V_{\text{GU}}\}$)中存在的碰撞。

以上两种情况都可以由生日碰撞得出：

$$|P_r[S_{\text{Succ}_2}]-P_r[S_{\text{Succ}_1}]|\leqslant\frac{(q_{\text{send}}+q_{\text{exe}})^2}{2p}+\frac{q_h^2}{2^{l+1}} \tag{7}$$

式中，l表示哈希函数H的输出长度，q_{send}，q_{exe}及q_h分别表示敌手进行 Send()，Execute()及哈希查询的次数。

(4) Game_3。在此游戏中，如果敌手幸运地直接猜中(不通过哈希查询的方式)协议中的验证因子$\{V_{\text{UG}_i},V_G,V_S,V_{\text{GU}}\}$，则终止此游戏的进行。由于在$\text{Game}_2$中已经排除了哈希碰撞的可能性，故$\text{Game}_2$和$\text{Game}_3$是不可区分的，除非敌手可以直接猜中上述验证因子，可得：

$$|P_r[S_{\text{Succ}_3}]-P_r[S_{\text{Succ}_2}]|\leqslant\frac{q_{\text{send}}}{2^l} \tag{8}$$

(5) Game_4。在此游戏中，如果敌手幸运地猜中(即不通过对应的查询)正确的双因素认证元t_i，则终止此游戏。由于$t_i=H(X_{\text{GWN}}\parallel I^*_{\text{ID}_i}\parallel T_{\text{reg}})$是一个哈希值，且在$\text{Game}_2$中已经排除了碰撞的可能性，故除非敌手直接猜中$t_i$的值，否则$\text{Game}_4$和$\text{Game}_3$是不可区分的，可得：

$$|P_r[S_{\text{Succ}_4}]-P_r[S_{\text{Succ}_3}]|\leqslant\frac{q_{\text{send}}}{2^l} \tag{9}$$

(6) Game_5。在此游戏中，若敌手能够正确地计算出(即通过对应的哈希查询，并在列表Λ_A中检验)双因素认证元t_i，则终止此游戏。

$$|P_r[S_{\text{Succ}_5}]-P_r[S_{\text{Succ}_4}]|\leqslant P_r[A_{\text{AskP}_5}] \tag{10}$$

在先前的游戏中已经讨论了敌手直接猜出正确t_i的情况，因此事件AskPara_5可以由以下 2 种情况导致：

① 敌手通过 Corrupt(U,1)查询的帮助，成功计算出正确的t_i，我们定义此事件为$\text{AskPara}_5\text{WithCorrupt}_1(A_{\text{AskP}_5,\text{WC}_1})$；

② 敌手通过 Corrupt(U,2)查询的帮助，成功计算出正确的t_i，我们定义此事件为$\text{AskPara}_5\text{WithCorrupt}_2(A_{\text{AskP}_5,\text{WC}_2})$。

首先来确定事件$\text{AskPara}_5\text{WithCorrupt}_1$发生的概率$P_r[A_{\text{AskP}_5,\text{WC}_1}]$。通过 Corrupt($U$,1)查询敌手可以获得用户的口令$\text{PW}_U$，由于没有其他的参数此时敌手的成功率与直接猜测相同：

$$P_r[A_{\text{AskP}_5,\text{WC}_1}]\leqslant\frac{q_{\text{send}}}{2^l} \tag{11}$$

接下来确定事件$\text{AskPara}_5\text{WithCorrupt}_2$发生的概率$P_r[A_{\text{AskP}_5,\text{WC}_2}]$。当进行 Corrupt($U$,2)查询时，敌手可以获得保存在智能卡中的安全参数$\{G_{t_i},A_i,P,G_1,n_0,H(\cdot),b\}$。此时，敌手为了获得认证元$t_i$，可以随机地选择从用户的角度或者服务器的角度进行攻击，因此，有$P_r[U]=P_r[S]=1/2$。如果敌手选择从服务器端进行攻击，无法获得认证元t_i。因此，此时敌手只有从用户的角度，通过离线猜测的方式，在服从 Zipf 分布的口令字典中，选取出正确的口令才有可能计算出正确的t_i：

$$\begin{aligned}P_r[A_{\text{AskP}_5,\text{WC}_2}]=&\\P_r[A_{\text{AskP}_5,\text{WC}_2}\mid U]\cdot P_r[U]+&\\P_r[A_{\text{AskP}_5,\text{WC}_2}\mid S]\cdot P_r[S]=&\\P_r[A_{\text{AskP}_5,\text{WC}_2}U]\cdot P_r[U]+&\\P_r[A_{\text{AskP}_5,\text{WC}_2}S]\cdot P_r[S]\leqslant&\\\frac{1}{2}c'\cdot q_{\text{send}}^{s'}+\frac{1}{2}\cdot 0=\frac{1}{2}c'\cdot q_{\text{send}}^{s'}&\end{aligned} \tag{12}$$

根据式(10)～(12)的结果，可得出：

$$|P_r[S_{\text{Succ}_5}]-P_r[S_{\text{Succ}_4}]|\leqslant\frac{1}{2}c'\cdot q_{\text{send}}^{s'}+\frac{q_{\text{send}}}{2^l} \tag{13}$$

(7) Game_6。在此游戏中，如果敌手能够通过正确地计算出(即通过对应的哈希查询，并在

列表 Λ_A 中检验)$\{V_{UG_i}, V_G, V_S, V_{GU}\}$,则终止此游戏,在事件 $AskH_6$ 不发生的情况下,$Game_6$ 与 $Game_5$ 具备完美的不可区分性:

$$|P_r[S_{Succ_6}]-P_r[S_{Succ_5}]|\leqslant P_r[A_{AskA_6}] \quad (14)$$

$$|P_r[A_{AskP_6}]-P_r[A_{AskP_6}]|\leqslant P_r[A_{AskA_6}] \quad (15)$$

(8) $Game_7$。在此游戏中,使用私有的哈希函数 H'来替换协议中使用的 H。由于敌手并不了解使用的 H',则有:

$$P_r[S_{Succ_7}]=\frac{1}{2} \quad (16)$$

且在事件 $AskH_7$ 不发生的情况下,$Game_7$ 与 $Game_6$ 具备完美的不可区分性:

$$|P_r[S_{Succ_7}]-P_r[S_{Succ_6}]|\leqslant P_r[A_{AskH_7}] \quad (17)$$

(9) $Game_8$。在此游戏中,我们使用 Diffie-Hellman 密钥交换算法问题的随机自归约性来模拟运行。由于此游戏中的参数计算与 $Game_7$ 中完全相同,因此有:

$$P_r[A_{AskH_8}]=P_r[A_{AskH_7}] \quad (18)$$

需要注意的是,$AskH_8$ 说明敌手已经发起了对(ECCDH(G_2,U_1)$\|U_1$)或(ECCDH(G_1,U_1)$\|U_1$)的 H 查询,并且敌手会在多项式时间 t 内完成此项查询,通过在列表 Λ_A 中随机地进行选择,我们可以概率 $1/q_h$ 得到真实的 G_{U_1} 或者 U_{S_1},因此:

$$P_r[A_{AskH_8}]\leqslant q_h\,\mathrm{Adv}_{P,G}^{ECCDH}(t) \quad (19)$$

由式(5)~(9)的结果,可得:

$$|P_r[S_{Succ_4}]-P_r[S_{Succ_0}]|\leqslant \frac{2q_{send}}{2^l}+\frac{(q_{send}+q_{exe})^2}{2p}+\frac{q_h^2}{2^{l+1}}$$

由式(10)~(17)的结果,可得:

$$|P_r[S_{Succ_7}]-P_r[S_{Succ_4}]|\leqslant P_r[A_{AskP_5}]+P_r[A_{AskA_6}]+P_r[A_{AskH_7}]$$

因此,综合可得:

$$\mathrm{Adv}_P^{AKE}(A)=2P_r[S_{Succ_7}]-1+2(P_r[S_{Succ_0}]-P_r[S_{Succ_7}])\leqslant c'\cdot q_{send}^{s'}+\frac{(q_{send}+q_{exe})^2}{p}+\frac{q_h^2+6q_{send}}{2^l}+6q_h\,\mathrm{Adv}_{P,G}^{ECCDH}(t) \quad (20)$$

至此,定理 1 证毕。

5 启发式安全性分析

由于 ROM 模型的局限性,导致一些现实攻击难以刻画。为了更为全面地分析本文提出协议的安全性。本节采用启发式分析的方式,对本文提出的 WSNs 环境下双因素认证协议进行安全性分析。

5.1 抵抗 2 种离线口令猜测攻击

离线口令猜测攻击是基于口令的身份认证协议无法绕过的攻击。该攻击的本质是敌手可以通过获取包含受害者口令的哈希值(如 2.2.1 节离线口令猜测攻击Ⅰ中的 $g_i^*=H(P_{PW_i}^*\|I_{ID_i}^*\|A_1)$)或与口令间接相关的哈希值(如 3.2.2 节离线口令猜测攻击Ⅱ中的 $V_i^*=H(M_{ID_i}\|M_{K_i}\|K_i^*\|I_{ID_i}^*\|T_1)$,其中 K_i^* 可以由用户口令导出),并利用哈希函数的抗碰撞属性验证猜测的口令是否正确。

本文提出的协议中,由于存在参数 $A_i=H(I_{ID_i}\|H(b\|P_{PW_i}))\bmod n_0$ 的存在,因此敌手能够通过验证的备选($I_{ID_i}^*$,$P_{PW_i}^*$)对有 $|D_{ID}\times D_{PW}|/n_0$ 个。为了得到精确的口令,敌手只能通过与服务器的交互来排除。然而为了防止这一事件的发生,本文引入了“Honeywords”技术。具体来说,当服务器收到登录请求后首先会计算 $I_{ID_i}^*=P_{PID_i}\oplus H(G_{U_1}\|U_1)$,然后根据得到的 $I_{ID_i}^*$ 来确定对应的 T_{reg},即敌手提供正确的 I_{ID_i} 是成功登录的第一步。接着,若验证消息 $V_{UG_i}^*$ 与 $H(C_{U_i}\|G_{U_1}^*\|P_{PID_i}\|t_i^*)$相等,则意味着敌手需要持有精确的用户口令 P_{PW_i}。只有提供精确的 P_{PW_i},敌手才能计算出正确的 $t_i=G_{t_i}\oplus H(b\|P_{PW_i})$。因此,一个正确的 I_{ID_i} 和一次失败的 V_{UG_i} 验证意味着:目前的登录者以非精确的 $P_{PW_i}^*$ 通过了智能卡的验证。这种情况有 2 种潜在的可能:(1) 合法的用户由于拼写错误碰巧输入了非精确的口令;(2) 有敌手在进行口令猜测攻击。通过计算可知,情况(1) 发生的概率为 $P_{r,typo}=n_0/|D_{PW}|$(为了给出一个较为直观的印象,采用文献[12]中使用的数据,$n_0=2^8$,口令空间 $|D_{PW}|=2^{20}$。因此 $P_{r,typo}=1/2^{12}$),由于 GWN 会将错误的 t_i 添加到 L_{Honey} 中,一个逐渐增长的 L_{Honey} 列表能不断加强 GWN 对口令猜测攻击发生的信心。假设 L_{Honey} 中包含 n 个 t_i,则 GWN 有 $P_{r,guess}=1-P_{r,typo}^n=1-1/2^{12n}$ 的概率确定有攻击者在进行口令猜测。因此本文提出的协议能够抵抗 2.2.1 节中的Ⅰ型离线口令猜测攻击。

对于Ⅱ型离线口令猜测攻击，由于用以证明用户合法身份的关键验证参数通常由用户的口令加以保护，如本文提出协议的 $t_i=G_{t_i}\oplus H(b\|P_{\mathrm{PW}_i})$。为了使 GWN 验证用户的身份，这样的值必然会包含在用户发送的登录请求中，如 $V_{\mathrm{UG}_i}=H(C_{U_i}\|G_{U_1}\|P_{\mathrm{PID}_i}\|t_i)$。敌手猜测的口令可以由构造的 t_i^* 体现，但是由于本文使用了公钥密码原语构建用户的登录信息，即 $G_{U_1}=u_1\cdot G_1$。只有持有私钥 u_1 的用户才能够重构验证参数 V_{UG_i}。因此，本文提出的协议能够抵抗 2.2.2 节中的Ⅱ型离线口令猜测攻击。

5.2　用户匿名性

用户匿名性要求用户的身份无法被除 GWN 以外的通信实体所知，且无法区分实体间的通信数据是否属于同一实体发出。本文提出的协议为用户提供随会话而改变的伪装身份标签，如 $P_{\mathrm{PID}_i}=I_{\mathrm{ID}_i}^*\cdot H(G_{U_1}\|U_1)$。由于每次登录时 U_i 都要选择一个随机数 u_1，因此每次登录的 P_{PID_i} 都不同。此外，由用户 U_i 发送的登录消息 $\{U_1,P_{\mathrm{PID}_i},C_{U_i},V_{\mathrm{UG}_i}\}$ 中都包含 U_1 和 G_{U_1} 这样的随会话而改变的参数，因此敌手也无法通过 2.4.2 节中提出的方法来建立用户与消息之间的联系。由于本文提出的协议采用公钥密码原语来构建用户登录信息，敌手无法区分在公开信道中截获的消息是否属于同一实体。

5.3　前向安全性

本文基于椭圆曲线 Diffie-Hellman 密钥交换算法的基本思想保障了会话密钥的安全建立。在本文提出的协议中会话密钥为 $K_{\mathrm{SK}}=h(r_u\|r_S\|S_{\mathrm{SID}_j})$，其中 r_u 与 r_S 分别为用户与传感节点选择的随机数。在用户与 GWN 的通信过程中，这些随机数受到共享秘密 $G_{U_1}=u_1\cdot X_{\mathrm{GWN}}\cdot P$ 和 $U_{S_1}=u_1\cdot s_1\cdot P$ 的保护，如 $C_{U_i}=(S_{\mathrm{SID}_j}\|t_i\|r_u)\oplus H(U_1\|G_{U_1})$ 和 $M_{\mathrm{RS}}=r_S\oplus H(U_{S_1}\|S_1)$；在传感节点与 GWN 的通信过程中，$r_S$ 受到哈希值 R_{S_j} 的保护，如计算 $M_{\mathrm{RS}}=r_S\oplus R_{S_j}$。由于本文使用了哈希链的技术，每当传感节点在发送消息后会使用 $R_{S_j}^{\mathrm{new}}=H(R_{S_j})$ 来替换参数 R_{S_j}，基于哈希函数的单向性，即使敌手捕获对应的传感节点并从中提取出参数 $R_{S_j}^{\mathrm{new}}$，由于无法计算出 $R_{S_j}^{\mathrm{new}}$ 的原像 R_{S_j}，故无法通过计算获得共享会话密钥 S_{SK}。此外，随机数 r_S 和 r_u 是随会话而改变的，即使 GWN 和 S_j 的私钥被泄露，敌手也无法计算出先前的会话密钥。因此，本文提出的协议能够实现前向安全性。

5.4　去同步攻击

本文在实现前向安全性时引入了单向哈希链的技术，由于敌手可以阻断、窃听和篡改公开信道上传输的消息，因此敌手可以在传感节点 S_j 完成对参数 R_{S_j} 的更新后阻断消息 $\{V_S,M_{\mathrm{RS}}\}$。当 GWN 与 S_j 再次通信时，S_j 使用 $R_{S_j}^{\mathrm{new}}$ 计算 $M_{\mathrm{RS}}=r_S\oplus R_{S_j}^{\mathrm{new}}$，然而 GWN 受之前敌手的干扰未及时更新 R_{S_j}，故无法计算出正确的 r_S。为了解决此问题，本文提出的协议中由传感节点 S_j 首先更新 R_{S_j}，当 GWN 与 S_j 就参数 R_{S_j} 失去同步后，GWN 只需要进行有限次的自循环（即重复计算 $R_{S_j}^{\mathrm{new}}=H(R_{S_j})$）后便可恢复同步，因此本文提出的协议能够抵抗去同步攻击。

5.5　双向认证

本文实现的双向认证分为 2 个方面：(1) 用户与网关节点之间的双向认证；(2) 网关节点和传感节点之间的双向认证。对于网关节点和用户之间的双向认证，GWN 计算验证参数 $V_{\mathrm{UG}_i}^*=H(C_{U_i}\|G_{U_1}^*\|P_{\mathrm{PID}_i}\|t_i^*)$ 并检查 $V_{\mathrm{UG}_i}^*$ 与 V_{UG_i} 是否相等，若相等则 GWN 认证了 U_i 的身份；U_i 计算 $V_{\mathrm{GU}}^*=H(G_{U_2}\|G_2\|r_u\|M_{\mathrm{GS}})$，并比较 V_{GU}^* 与 V_{GU} 是否相等，如相等则 U_i 认证了 GWN 的身份。对于网关节点和传感节点之间的双向认证，S_j 计算 $V_G^*=H(M_{R_j}\|V_{S_j}^*\|U_1)$ 并比较 V_G^* 与 V_G 是否相等。若相等则 S_j 认证 GWN 的身份；GWN 计算 $V_S^*=H(r_S^*\|r_u\|M_{\mathrm{RS}}\|S_{\mathrm{SID}_j})$ 并比较 V_S^* 与 V_S 是否相等，若相等，则 GWN 认证 S_j 的身份。

5.6　传感节点捕获攻击

由于传感节点长期部署在无人监管的环境中，其保存的安全参数易被敌手获取。因此，为了阻止敌手通过捕获传感节点从而获取与用户相关的敏感数据，在本文提出的协议中 GWN 仅将 U_i 选择的随机数 r_u 发送给传感节点 S_j，即使敌手捕获了用户选择的传感节点，也无法计算出代表用户身份的 t_i。因此，本文提出的协议能抵抗传感节点捕获攻击。

6　性能与安全性对比

本节将近期提出的 WSNs 环境下多因素身份认证协议如 Li 等[35]、Yang 等[34]、Wu 等[52]、

Srinivas 等[53]、Ali 等[54]、Vinoth 等[55]以及王晨宇等[39]协议，与本文提出的改进协议进行对比。对比主要从协议的安全性和运行效率 2 个方面进行，结果如表 4 所示。

表 4 中关于协议性能的对比分析主要包含 3 个部分：(1) 计算量。其中 T_H 表示哈希运算所花费的时间，T_S 表示对称加解密所花费的时间，T_P 表示椭圆曲线点乘运算所花费的时间，T_B 表示与生物特征相关的运算(如指纹识别等)时间。除此之外，由于异或运算以及 bit 连接等运算时间消耗较短，故在本环节中我们忽略其所需要的时间。(2) 通信量。主要指用户、传感节点及网关之间通过公开信道传递的消息大小。(3) 存储量。表示用户、网关、传感节点在本地存储的数据大小。为了便于比较，本文假设哈希函数的输出、随机数、口令、身份标识等参数的长度为 128 bit；公平起见，本文在相同的安全强度下对基于对称密码算法的系统与基于 ECC 的系统进行比较：假设对称加密算法的密钥为 128 bit；基于 ECC 的系统中私钥长度为 256 bit，公钥长度为 512 bit。通过表 4 的对比结果可以看出，虽然采用公钥密码原语会在一定程度上影响协议的运行效率以及增加设备的通信和存储负担，但由于在协议的设计过程中往往需要寻求协议安全性与可用性的平衡，因此在实现诸如前向安全性、用户匿名性等重要安全属性时，可以选择牺牲一些可用性从而换取更高的安全性。

表 4 中关于协议安全性的对比显示，不使用公钥密码原语会造成某些安全属性的缺失，如 C5 抗已知攻击性和 C11 用户匿名性。对比结果表明，本文提出的协议不仅填补了前文提到的 Li 等[35]协议和 Yang 等[34]协议中存在安全漏洞，且满足 Wang 等[7]提出的所有协议评价指标。特别地，表 4 显示由王晨宇等[39]提出的协议在安全性层面与本文协议相同，均满足 12 条协议评价指标。从运行效率层面看，本文协议采用的公钥密码运算总数与之相同均为 6 次，但在资源限制的传感设备端，本文仅仅使用了 6 次哈希运算，相比于王晨宇等[36]协议的 4 次哈希运算和 2 次椭圆曲线点乘运算，本文协议具有更少的计算开销，因此更适用于资源限制无线传感网络环境。

表 4 WSNs 环境下相关认证协议的性能比较

Tab. 4 Performance comparison among relevant authentication schemes in WSNs

协议	计算量			通信量/bit	存储量/bit	Wang 等[7]协议评价指标											
	用户端	网关	传感器			C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
Wu 等[52]	$8T_H$	$10T_H$	$3T_H$	2 533	3 328	√	√	×	×	×	×	√	×	×	√	√	×
Srinivas 等[53]	$8T_H+T_B$	$6T_H$	$13T_H$	2 304	3 200	√	√	×	×	√	×	√	×	×	√	×	×
Ali 等[54]	$6T_H+2T_S+T_B$	$6T_H+2T_S$	$5T_H+T_S$	1 792	1 152	√	√	×	×	×	×	√	×	×	√	×	×
Yang 等[34]	$10T_H$	$19T_H$	$8T_H$	5 376	2 304	√	√	√	×	×	√	√	√	×	√	×	√
Vinoth 等[55]	$9T_H+T_B+T_S$	$6T_H+3T_S$	T_H+2T_S	3 040	4 536	√	√	√	×	×	√	√	×	√	√	×	×
Li 等[35]	$10T_H$	$9T_H$	$7T_H$	2 520	2 312	√	√	√	×	×	√	√	×	√	×	×	×
王晨宇等[39]	$5T_H+T_B+3T_P$	$9T_H+T_P$	$4T_H+2T_P$	2 944	2 688	√	√	√	√	√	√	√	√	√	√	√	√
Chaudhry 等[56]	T_B+17T_H	$13T_H$	$8T_H$	2 560	3 200	√	√	√	×	×	×	√	×	√	√	√	×
Wazid 等[57]	T_B+16T_H	$8T_H$	$8T_H$	1 664	3 456	√	√	√	×	×	√	√	×	√	√	√	×
本文方法	$9T_H+3T_P$	$11T_H+3T_P$	$6T_H$	2 688	3 200	√	√	√	√	√	√	√	√	√	√	√	√

7 结束语

WSNs 环境下，基于非抗窜扰智能卡假设，设计安全高效的多因素身份认证协议是近年来的研究热点。本文分析了两个 WSNs 环境下典型的多因素身份认证协议，分别指出了这两个协议存在多种已知攻击以及缺失了多种重要的安全属性。在阐明这些安全缺陷产生的根本原因的

基础上，综合考虑了由 Wang 等[7]提出的 12 条针对 WSNs 环境的多因素协议评价指标，以及 7 种常见的已知攻击，提出了一个改进的双因素认证协议。ROM 模型下的形式化证明显示了本文协议的安全性。然而，本文设计的协议仅面向单网关，设计面向多网关的无线传感网络环境下基于非抗窜扰智能卡的多因素身份认证协议是本文未来的研究方向。希望本文的工作能为未来的 WSNs 环境下基于非抗窜扰智能卡假设的多因素身份认证协议设计提供更多的灵感。

参 考 文 献

[1] DAS A K, WAZID M, KUMAR N, et al. Biometrics-based privacy-preserving user authentication scheme for cloud-based industrial internet of things deployment[J]. IEEE Internet of Things Journal, 2018, 5(6): 4900-4913.

[2] KHAN W Z, REHMAN M H, ZANGOTI H M, et al. Industrial internet of things: recent advances, enabling technologies and open challenges[J]. Computers & Electrical Engineering, 2020, 81: 106522.

[3] MENG Z, WU Z, MUVIANTO C, et al. A data-oriented M2M messaging mechanism for industrial IoT applications[J]. IEEE Internet of Things Journal, 2017, 4(1): 236-246.

[4] YIN C, XI J, SUN R, et al. Location privacy protection based on differential privacy strategy for big data in industrial internet of things[J]. IEEE Transactions on Industrial Informatics, 2017, 14(8): 3628-3636.

[5] 程庆丰，李钰汀，李兴华，等. 面向边缘计算环境的密码技术研究综述[J]. 计算机科学，2020，47(11)：10-18.
CHENG Qingfeng, LI Yuting, LI Xinghua, et al. Research on application of cryptography technology for edge computing environment[J]. Computer Science, 2020, 47(11): 10-18. (in Chinese)

[6] WAZID M, DAS A K, ODELU V, et al. Secure remote user authenticated key establishment protocol for smart home environment[J]. IEEE Transactions on Dependable and Secure Computing, 2017, 17(2): 391-406.

[7] WANG D, LI W, WANG P. Measuring two-factor authentication schemes for real-time data access in industrial wireless sensor networks[J]. IEEE Transactions on Industrial Informatics, 2018, 14(9): 4081-4092.

[8] SHINAND S H, KOBARA K. Security analysis of password-authenticated key retrieval[J]. IEEE Computer Architecture Letters, 2017, 14(5): 573-576.

[9] 罗敏，孙艾颖，阴晓光，等. 基于身份的双服务器口令保护协议[J]. 密码学报，2020，7(6)：839-852.
LUO Min, SUN Aiyin, YIN Xiaoguang, et al. Dual-server identity-based password protection scheme[J]. Journal of Cryptologic Research, 2020, 7(6): 839-852. (in Chinese)

[10] 魏福山，马建峰，李光松，等. 标准模型下高效的三方口令认证密钥交换协议[J]. 软件学报，2016，27(9)：2389-2399.
WEI Fushan, MA Jianfeng, LI Guangsong, et al. Efficient three-party password-based authenticated key exchangeprotocol in the standard model[J]. Journal of Software, 2016, 27(9): 2389-2399. (in Chinese)

[11] HUANG X, CHEN X, LI J, et al. Further observations on smart-card-based password-authenticated key agreement in distributed systems[J]. IEEE Transactions on Parallel and Distributed Systems, 2013, 25(7): 1767-1775.

[12] WANG D, WANG P. Two birds with one stone: two-factor authentication with security beyond conventional bound[J]. IEEE Transactions on Dependable and Secure Computing, 2018, 15(4): 708-722.

[13] KUMARI S, KHAN M K, ATIQUZZAMAN M. User authentication schemes for wireless sensor networks: a review[J]. Ad Hoc Networks, 2015, 27: 159-194.

[14] DAS M L, SAXENA A, GULATI V P. A dynamic ID-based remote user authentication scheme[J]. IEEE Transactions on Consumer Electronics, 2004, 50(2): 629-631.

[15] WANG Y, LIU J, XIAO F, et al. A more efficient and secure dynamic ID-based remote user authentication scheme[J]. Computer Communications, 2009, 32(4): 583-585.

[16] KHAN M K, KIM S K, ALGHATHBAR K. Cryptanalysis and security enhancement of a more efficient & secure dynamic ID-based remote user authentication scheme[J]. Computer Communications, 2011, 34(3): 305-309.

[17] MESSERGES T S, DABBISH E A, SLOAN R H. Examining smart-card security under the threat of power analysis attacks[J]. IEEE Transactions on Computers, 2002, 51(5): 541-552.

[18] NOHL K, EVANS D, STARBUG S, et al. Reverse-engineering a cryptographic RFID tag[C]// Proceedings of the 17th USENIX Security Symposium. San Jose, USA: [s. n.], 2008: 28-37.

[19] PAUL C KOCHER, JOSHUA JAFFE, BENJAMIN

JUN. Differential Power Analysis[C]// Proceedings of CRYPTO. [S. l. : s. n.], 1999: 388-397.

[20] KARINE G, CHRISTOPHE M, FRANCIS O. Electromagnetic analysis: concrete results[C]// Proceedings of CHES. [S. l: s. n.], 2001: 251-261.

[21] RAMYA J M, DEVENDRA R, AANJHAN R, et al. Thermal covert channels on culti-core platforms[C]// Proceedings of USENIX Security Symposium. [S. l. : s. n.], 2015: 865-880.

[22] CORON J S. Resistance against differential power analysis for elliptic curve cryptosystems[C]//Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems. Heidelberg, Germany: Springer, 1999: 292-302.

[23] MANGARD S, OSWALD E, POPP T. Power analysis attacks: revealing the secrets of smart cards[M]. Berlin: Springer Science & Business Media, 2008.

[24] CARBONE M, CONIN V, CORNELIE M A, et al. Deep learning to evaluate secure RSA implementations [J]. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2019(2): 132-161.

[25] ROCHE T, LOMNÉV, MUTSCHLER C, et al. A side journey to titan [C]//Proceedings of the 30th USENIX Security Symposium (USENIX Security 21). [S. l. : s. n.], 2021: 231-248.

[26] CHANG C, WU T. Remote password authentication with smart cards[J]. IEE Proceedings-E, 1991, 138 (3): 165-168.

[27] LIAO I E, LEE C C, WANG M S. A password authentication scheme over insecure networks[J]. Journal of Computer and System Sciences, 2006, 72(4): 727-740.

[28] DAS M L. Two-factor user authentication in wireless sensor networks[J]. IEEE Transactions on Wireless Communications, 2009, 8(3): 1086-1090.

[29] HE D, GAO Y, CHAN S, et al. An enhanced two-factor user authentication scheme in wireless sensor networks[J]. Ad Hoc & Sensor Wireless Networks, 2010, 10(4): 361-371.

[30] XUE K, MA C, HONG P, et al. A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks[J]. Journal of Network and Computer Applications, 2013, 36(1): 316-323.

[31] DAS A K. A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks[J]. Peer-to-Peer Networking and Applications, 2016, 9(1): 223-244.

[32] JIANG Q, MA J, WEI F, et al. An untraceable temporal-credential-based two-factor authentication scheme using ECC for wireless sensor networks[J]. Journal of Network and Computer Applications, 2016, 76: 37-48.

[33] LI X, NIU J, KUMARI S, et al. A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments[J]. Journal of Network and Computer Applications, 2018, 103: 194-204.

[34] YANG Z, HE J, TIAN Y, et al. Faster authenticated key agreement with perfect forward secrecy for industrial internet-of-things[J]. IEEE Transactions on Industrial Informatics, 2019, 16(10): 6584-6596.

[35] LI J, SU Z, GUO D, et al. PSL-MAAKA: provably-secure and lightweight mutual authentication and key agreement protocol for fully public channels in internet of medical things[J]. IEEE Internet of Things Journal, 2021, 8(17): 13183-13195.

[36] 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.

[37] MA C G, WANG D, ZHAO S D. Security flaws in two improved remote user authentication schemes using smart cards[J]. International Journal of Communication Systems, 2014, 27(10): 2215-2227.

[38] WANG D, WANG P. On the anonymity of two-factor authentication schemes for wireless sensor networks: attacks, principle and solutions[J]. Computer Networks, 2014, 73: 41-57.

[39] 王晨宇，汪定，王菲菲，等. 面向多网关的无线传感器网络多因素认证协议[J]. 计算机学报，2020，43(4)：683-700.
WANG Chenyu, WANG Ding, WANG Feifei, et al. Multi factor user authentication scheme for multi gateway wireless sensor networks[J]. Chinese Journal of Computers, 2020, 43(4): 683-700. (in Chinese)

[40] DOLEV D, YAO A. On the security of public key protocols[C]//Proceedings of the 22nd Annual Symposium on Foundations of Computer Science. Nashville, USA: IEEE, 1981: 350-357.

[41] BONNEAU J, JUST M, MATTHEWS G. "What is in a name?"[C\]//Proceedings of Financial Cryptography and Data Security 2010. Canary Islands, Spain: [s. n.], 2010: 98-113.

[42] BONNEAU J. The science of guessing: analyzing an anonymized corpus of 70 million passwords[C]// Proceedings of IEEE Symposium on Security and Privacy. San Francisco, USA: IEEE, 2012: 538-552.

[43] KIM T H, KIM C, PARK I. Side channel analysis attacks using AM demodulation on commercial smart cards with SEED[J]. Journal of Systems and Software, 2012, 85(12): 2899-2908.

[44] HUSSAIN K, JHANJHI N Z, MATIURRAHMAN H, et al. Using a systematic framework to critically analyze proposed smart card based two factor authentication schemes[J]. Journal of King Saud University Computer and Information Sciences, 2021, 33(4): 417-425.

[45] WANG D, HE D, WANG P, et al. Anonymous two-factor authentication in distributed systems: certain goals are beyond attainment[J]. IEEE Transactions on Dependable and Secure Computing, 2014, 12(4): 428-442.

[46] LAMPORT L. Password authentication with insecure communication[J]. Communications of the ACM, 1981, 24(11): 770-772.

[47] SYAMSUDDIN I, DILLON T, CHANG E, et al. A survey of RFID authentication protocols based on hash-chain method[C]//Proceedings of the 3rd International Conference on Convergence and Hybrid Information Technology. Busan, Korea: IEEE, 2008: 559-564.

[48] BELLARE M, POINTCHEVAL D, ROGAWAY P. Authenticated key exchange secure against dictionary attacks[C]//Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Germany: [s. n.], 2000: 139-155.

[49] WANG D, WANG P. On the implications of Zipf's law in passwords[C]// Proceedings of the 21st European Symposium on Research in Computer Security. Heraklion, Greece: Springer, 2016: 111-131.

[50] BRESSON E, CHEVASSUT O, POINTCHEVAL D. Security proofs for an efficient password-based key exchange[C]//Proceedings of the 10th ACM Conference on Computer and Communications Security. Washington D C, USA: [s. n.], 2003: 241-250.

[51] WANG D, CHENG H, WANG P, et al. Zipf's law in passwords[J]. IEEE Transactions on Information Forensics and Security, 2017, 12(11): 2776-2791.

[52] WU F, XU L, KUMARI S, et al. An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment[J]. Journal of Network and Computer Applications, 2017, 89: 72-85.

[53] SRINIVAS J, MUKHOPADHYAY S, MISHRA D. Secure and efficient user authentication scheme for multi-gateway wireless sensor networks[J]. Ad Hoc Networks, 2017, 54: 147-169.

[54] ALI R, PAL A K, KUMARI S, et al. A secure user authentication and key-agreement scheme using wireless sensor networks for agriculture monitoring[J]. Future Generation Computer Systems, 2018, 84: 200-215.

[55] VINOTH R, DEBORAH L J, VIJAYAKUMAR P, et al. Secure multi-factor authenticated key agreement scheme for industrial IoT[J]. IEEE Internet of Things Journal, 2020, 8(5): 3801-3811.

[56] CHAUDHRY S A, IRSHAD A, YAHYA K, et al. Rotating behind privacy: an improved lightweight authentication scheme for cloud-based IoT environment[J]. ACM Transactions on Internet Technology, 2021, 21(3): 1-19.

[57] WAZID M, DAS A K, BHAT V, et al. LAM-CIoT: lightweight authentication mechanism in cloud-based IoT environment[J]. Journal of Network and Computer Applications, 2020, 150: 1-16.

责任编辑　董　莉