DOI:10.3969/j.issn.1001-0505.2020.05.018

5G 网络中 D2D 安全动态群组认证和密钥协商协议

程贤兵[1] 蒋 睿[1] 裴 蓓[2] 吴松洋[2]

([1] 东南大学网络空间安全学院,南京 210096)
([2] 信息网络安全公安部重点实验室,上海 200031)

摘要:为了解决5G网络中设备到设备(D2D)的群组通信中常见的安全威胁,提出了确保D2D安全通信的动态群组认证和密钥协商(DG-AKA)协议方案.结果表明:该方案基于CDH假设难题实现了安全的认证,使得非法用户无法伪造签名;基于MDBDH假设难题并结合安全认证过程实现了安全的密钥协商,使得非法用户或核心网络无法获取共享会话密钥,保证了密钥的安全性,解决了密钥托管问题;结合认证和密钥协商过程实现了安全的动态群组成员管理以保证群组前向和后向安全,当群组成员被撤销或新成员加入时,无需重新执行全部协议,即能安全地更新会话密钥.安全性分析证明了DG-AKA协议方案满足所有安全性目标,效率分析则表明了该方案具有和现有方案同等数量级的运行效率.

关键词:D2D通信;认证;密钥协商;动态群组
中图分类号:TN918.4 **文献标志码**:A **文章编号**:1001-0505(2020)05-0918-11

Dynamic group authentication and key agreement protocol for D2D secure communication in 5G networks

Cheng Xianbing[1] Jiang Rui[1] Pei Bei[2] Wu Songyang[2]

([1]School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China)
([2]Key Lab of Information Network Security, Ministry of Public Security, Shanghai 200031, China)

Abstract: To solve common secure challenges in device-to-device (D2D) group communication in 5G networks, a dynamic group-authentication and key agreement (DG-AKA) protocol scheme for ensuring D2D secure communication was proposed. The results show that the security authentication can be implemented based on the Diffie-Hellman(CDH) hypothesis problem, so illegal users cannot forge signatures. Based on the MDBDH hypothesis problem and the security authentication process, a secure key agreement can be implemented, preventing illegal users or the core network to obtain the shared session key, thus the security of the key is ensured and the key escrow problem is solved. Combined with the authentication and key agreement process, a secure dynamic group member management can be implemented to ensure the group forward and backward secrecy. When group members are revoked or new members join the system, the session key is safely updated without re-executing the entire protocol. The security analysis proves that DG-AKA protocol scheme meets all safety objectives, and the efficiency analysis shows that the scheme has the same order of operation efficiency as the existing schemes.

Key words: device-to-device (D2D) communication; authentication; key agreement; dynamic group

收稿日期:2020-01-15. **作者简介**:程贤兵(1995—),男,硕士生;蒋睿(联系人),男,博士,副教授,R.Jiang@seu.edu.cn.
基金项目:国家自然科学基金资助项目(61372103)、江苏省自然科学基金资助项目(SBK2020020282)、信息网络安全公安部重点实验室开放课题资助项目(C19607)、江苏省计算机网络技术重点实验室资助项目.
引用本文:程贤兵,蒋睿,裴蓓,等.5G网络中D2D安全动态群组认证和密钥协商协议[J].东南大学学报(自然科学版),2020,50(5):918-928. DOI:10.3969/j.issn.1001-0505.2020.05.018.

为了适应无线通信服务的多样化和普遍接入的发展趋势,5G 无线网络通过与 LTE-A、WLAN 以及其他无线接入技术相结合,成为一种高密度的异构网络. 高密度 5G 异构网络在增加网络容量方面起着重要作用. 然而,复杂的宏蜂窝和微蜂窝网络之间的相互干扰限制了通信容量的增加. D2D 通信作为流量卸载技术,可以直接在临近设备之间进行通信,从而减轻基站承载网络流量的负担. 文献[1-6]表明 D2D 通信能够大幅度提高频谱利用率,具有较低的通信时延,以及适应更加复杂通信环境(如应急通信、物联网通信等)的能力. IEEE 802.11、802.15 标准为 D2D 通信提供了许多协议,例如 Wi-Fi、LTE 和 Bluethooth. 然而这些协议使用在不同的无线设备中,使得 D2D 通信面临严重的安全威胁.

在许多 D2D 通信场景中安全是十分重要的,如移动支付、无线局域网中个人医疗信息的传输、车联网中的车辆信息以及智能家居等. 而身份认证和密钥协商方案可以帮助 D2D 用户建立安全的通信渠道. 因此研究适用于 D2D 通信的安全认证和密钥协商协议具有十分重要的意义.

近年来,越来越多的研究[7-19]集中在 D2D 通信的安全认证和密钥协商协议上,其中文献[7-15]主要解决 2 个用户间的 D2D 安全认证和密钥协商问题. 文献[7]提出了一个基于异或的安全密钥分发方案,然而基于异或的密钥很容易被提取,所以该方案不能保证安全的 D2D 通信. 文献[8]提出了一个通过 WiFi 直连的共享密钥建立方案,该方案通过 Diffie-Hellman 密钥交换机制保证安全的密钥分发,然而该方案并没有实现真正的相互认证过程,容易遭到消息篡改或假冒等攻击. 文献[9]提出了一个不需要核心网络参与的安全密钥交换方案,该方案能够抵御中间人攻击. 文献[10]提出了一个适用于漫游场景的 D2D 通用身份认证和密钥协商协议,然而该方案要求访问网络(VN)大量参与 D2D 通信过程. 文献[11]提出了一种基于属性的 D2D 通信方案,该方案将可信协商过程建模为一个 0/1 背包问题,然后基于同态加密技术进行安全计算,以确保 D2D 通信的安全性. 文献[12]提出了一种轻量级的基于无证书广义签密技术的安全感知 D2D 辅助数据传输协议,该协议适用于医疗健康系统,以满足对敏感信息保护的更高要求. 文献[13]提出了一种安全的数据共享协议方案,该方案借助于基站(eNB)在 D2D 用户之间实现了相互认证和安全的数据传输,然而 eNB 的过度参与导致整个系统存在局限性和单点故障. 文献[14-15]提出了 2 个实现安全和匿名的 D2D 群组通信协议,但是这 2 个协议仅仅使用 2 个用户之间群组信息的匿名性来解决安全的 D2D 通信问题,并不适用于安全的 D2D 群组通信.

在 D2D 通信中,研究群组通信的安全认证和密钥协商协议更具必要性. 群组通信可以帮助用户建立基于群组的服务,如多人游戏、多人聊天和智能家居等. 文献[16]提出了一个适用于 D2D 群组通信的认证和密钥协商方案(简称方案 1),该方案可以在不需要核心网络的情况下,进行群组密钥协商. 然而,由于该方案在密钥协商过程中需要同时传递用户公钥以及由私钥生成的签名,因此攻击者可以通过替换用户公钥,然后篡改协商信息并利用自己的私钥签名,最后发送给接收方,以此达到欺骗的目的. 文献[17]提出了一个安全的 D2D 群组认证和密钥协商方案(简称方案 2),该方案利用多项式来分发密钥,每个用户利用自己的身份信息来计算多项式参数,然后将参数代入多项式进行计算来提取会话密钥. 然而,在群组成员动态管理的过程中,基于多项式的密钥更新方式并不能保证安全的会话密钥更新,离开的用户仍然能够计算出新的会话密钥. 文献[18]提出了 2 个具有隐私保护的认证和密钥协商方案 PPAKA-HMAC 和 PPAKA-IBS(简称方案 3),这 2 个方案均适用于安全的 D2D 群组通信,但是在动态群组成员管理的过程中,必须依赖于安全信道,导致方案的实用性降低. 文献[19]提出了一种适用于医疗物联网中 D2D 通信的群组密钥协商方案(简称方案 4),该方案使用秘密共享的方式分发密钥,但该方案不能实现动态的群组成员管理,在群组成员发生变化时,不能及时更新会话密钥,无法保证群组前向和后向安全.

目前已有的 D2D 动态群组认证和密钥协商协议方案仍存在一些问题. 其中,方案 1 没有实现安全的身份认证,攻击者能够伪造认证消息从而欺骗用户. 其次,在密钥托管问题上,方案 2 和方案 4 的会话密钥是由核心网络生成和管理的,一旦核心网络被攻击,用户数据将被泄露. 最后,在群组成员动态管理上,方案 2、方案 3 和方案 4 没有真正实现安全动态群组管理,密钥更新过程极易被攻击或需要通过安全信道实现.

因此,为了解决上述安全问题,本文提出一个 5G 网络中 D2D 安全动态群组认证和密钥协商(DG-AKA)协议方案. 在 DG-AKA 协议方案中,

D2D 群组成员在服务网络(SN)的协助下,进行相互认证并协商出一个共享会话密钥,从而实现安全的认证和密钥协商,并且可实现群组成员的动态管理.

1 预备知识

1.1 定义

定义 1(双线性对)[20] 定义 G_0 和 G_1 是有着相同素数阶 q 的乘法循环群,映射 $e:G_0\times G_0\to G_1$ 称为双线性对,满足如下性质:

1) 双线性. $\forall g_0, g_1 \in G_0$, $\forall a, b \in \mathbf{Z}_q^*$, 有 $e(g_0^a, g_1^b) = e(g_0^b, g_1^a) = e(g_0, g_1)^{ab}$.

2) 非退化性. $\exists g_0, g_1 \in G_0 \backslash \{1\}$, $e(g_0, g_1) \neq 1$.

3) 可计算性. $\forall g_0, g_1 \in G_0$, $e(g_0, g_1)$ 可以在群 G_1 中得到有效的计算.

定义 2(计算 Diffie-Hellman(CDH)假设)[21]

设 G_0 是一个素数阶 q 的乘法循环群,g 是生成元. 随机选择 $a, b \in \mathbf{Z}_q^*$,并公开 $g^a, g^b \in G_0$,以此计算 g^{ab} 是困难的.

对于任何多项式时间敌手 A,将其针对 CDH 问题的优势定义为 $\varepsilon_1 = |\Pr[A(g, g^a, g^b) = g^{ab}]|$,当 ε_1 在一个多项式时间内可忽略时,CDH 假设成立.

定义 3(判定双线性 Diffie-Hellman(DBDH)假设)[22] 定义 $e:G_0\times G_0\to G_1$ 是一个双线性对,g 是群 G_0 中一个生成元,且随机选择 $a, b, c, d \in \mathbf{Z}_q^*$,并公开 $g^a, g^b, g^c \in G_0$,以此区分 $(g, g^a, g^b, g^c, e(g,g)^{abc})$ 和 $(g, g^a, g^b, g^c, e(g,g)^d)$ 是困难的.

设 F 为 0-1 函数,它对于区分上述 2 个元组的优势为 $\varepsilon_2 = |\Pr[F(e(g,g)^{abc}) = 0] - \Pr[F(e(g,g)^d) = 0]|$,当 ε_2 在一个多项式时间内可忽略时,DBDH 假设成立.

定义 4(变性的判定双线性 Diffie-Hellman(MDBDH)假设)[21] 定义 $e:G_0\times G_0\to G_1$ 是一个双线性对,g 是群 G_0 中一个生成元,且随机选择 $a, b, c, d, s \in \mathbf{Z}_q^*$ 并公开 $g^a, g^b, g^c, g^s, g^{sa}, g^{sb}, g^{sc} \in G_0$,以此区分 $(g, g^a, g^b, g^c, g^s, g^{sa}, g^{sb}, g^{sc}, e(g,g)^{abc})$ 和 $(g, g^a, g^b, g^c, g^s, g^{sa}, g^{sb}, g^{sc}, e(g,g)^d)$ 是困难的.

设 F 为 0-1 函数,它对于区分上述 2 个元组的优势为 $\varepsilon_3 = |\Pr[F(e(g,g)^{abc}) = 0] - \Pr[F(e(g,g)^d) = 0]|$,当 ε_3 在一个多项式时间内可忽略时,MDBDH 假设成立.

1.2 群组前向和后向安全

定义 5(群组前向安全) 对于一个已存在的群组,当有新用户加入并建立新的群组后,新加入的成员无法获取原群组的会话密钥. 即对于任何多项式时间敌手 A,定义 $\varepsilon_4 = \mathrm{Adv}_A^{\text{DG-AKA-fs}}(t, q_E, q_S)$ 为敌手在执行 q_E 次 Execute 询问和 q_S 次 Send 询问后获得加入群组并获得会话密钥的最大优势. 当 ε_4 在一个多项式时间 t 内可忽略时,则满足群组前向安全.

定义 6(群组后向安全) 对于一个已存在的群组,当有成员被撤销,未撤销的成员构建新的群组,并更新会话密钥,离开的用户无法获取新的会话密钥. 即对于任何多项式时间敌手 A,定义 $\varepsilon_5 = \mathrm{Adv}_A^{\text{DG-AKA-bs}}(t, q_E, q_S)$ 为敌手在执行 q_E 次 Execute 询问和 q_S 次 Send 询问后获得更新会话密钥的最大优势. 当 ε_5 在一个多项式时间 t 内可忽略时,则满足群组后向安全.

2 DG-AKA 协议方案

2.1 系统模型

在 DG-AKA 协议方案中有 D2D 用户和 SN 两种不同的实体,其系统通信模型如图 1 所示. 为便于理解,只给出了 3 个用户(U_1、U_2、U_3)之间的系统模型,但它同样适用于 3 个以上用户的场景. D2D 通信中用户与用户之间直接进行相互认证和密钥协商,从而得到 D2D 群组会话密钥. SN 是一个用于用户身份管理、密钥管理以及 D2D 通信管理的机构,它负责为用户生成临时身份和公/私钥对,并帮助用户相互之间建立 D2D 通信会话.

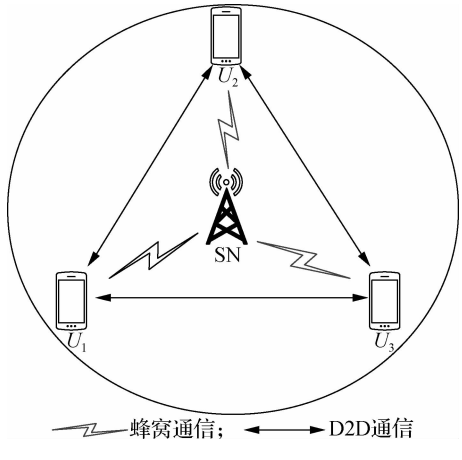

图 1 系统模型

2.2 安全模型

1) 参与者. 每个用户 U_i 有一个真实的身份 Q_i,其中 $i = 1, 2, \cdots, n$. 假设有一组用户 $G = \{U_1,$

$U_2,\cdots,U_n\}$.

在安全模型中,允许每个用户 $U_i \in G$ 可以与不同用户之间多次执行协议. 用 $\Pi_{U_i}^{\pi}$ 表示用户 U_i 第 π 次执行协议的实例.

2) 初始化. 在这个阶段,每个用户 $U_i \in G$ 获得一个公私钥对(PK_i,SK_i)和一个唯一的临时身份 T_i. 每个用户均可以获得系统参数和 D2D 会话列表 $L=\{T_1,T_2,\cdots,T_n\}$.

3) 敌手模型. 本方案的安全性和敌手的能力有关,假设一个多项式时间敌手 A 完全控制通信信道,并且可以询问任何实例. A 可以自适应地询问以下预言机:

① Extract(T_U). 该预言机允许敌手获取与用户 U 临时身份 T_U 相关的长期密钥,其中,$T_U \notin L$.

② Execute(L^*). 该预言机对被动攻击进行建模,允许敌手窃听用户集合 L^* 中用户之间的协议执行过程,并记下它们诚实执行协议的全部记录.

③ Send($\Pi_{U_i}^{\pi}$,M). 该预言机对主动攻击进行建模,它允许敌手发送消息 M 给实例 $\Pi_{U_i}^{\pi}$. 在接收到来自敌手的询问后,该预言机将返回由 $\Pi_{U_i}^{\pi}$ 生成的回复.

④ Corrupt(U_i). 该预言机对揭示长期密钥攻击进行建模,它允许敌手向 U_i 发出询问,当 U_i 接收到询问后,该预言机返回 U_i 的长期密钥.

在敌手模型中,敌手 A 可以通过 2 种攻击方式获得优势: ①敌手伪造认证消息或冒充合法用户;② 获取会话密钥且不修改协议中的任何消息. 因此,敌手 A 获得的优势为 $\mathrm{Adv}_A^{\text{DG-AKA}}(t,q_{\mathrm{E}},q_{\mathrm{S}}) = \mathrm{Pr}_A[\mathrm{Forge}] + \mathrm{Pr}_A[\sim\mathrm{Forge}]$,其中 $\mathrm{Adv}_A^{\text{DG-AKA}}(t,q_{\mathrm{E}},q_{\mathrm{S}})$表示敌手在多项式时间 t 内发出 q_{E} 次 Execute 询问和 q_{S} 次 Send 询问后赢得的优势,$\mathrm{Pr}_A[\mathrm{Forge}]$ 表示敌手成功伪造认证消息的概率,$\mathrm{Pr}_A[\sim\mathrm{Forge}]$ 表示敌手在不修改任何消息的情况下成功获取会话密钥的概率.

2.3　具体方案

本文提出的 DG-AKA 协议方案有系统初始化、用户注册、安全 D2D 发现、会话请求、会话建立、动态群组成员管理 6 个阶段.

2.3.1　系统初始化

SN 选择 2 个拥有相同素数阶 q 的乘法循环群 G_0、G_1 及一个 G_0 的生成元 g,设 $e:G_0\times G_0\to G_1$ 是一个双线性对,$H_1:\{0,1\}^*\to\mathbf{Z}_q^*$ 和 $H_2:\{0,1\}^*\to G_0$ 是 2 个安全的单向 Hash 函数. 随后 SN 选择一个随机数 $\alpha\in\mathbf{Z}_q^*$ 作为系统的主密钥,计算 $K_{\text{pub}}=g^{\alpha}$ 作为系统的公钥. 最后发布系统参数$\{G_0,G_1,e,H_1,H_2,g,q,K_{\text{pub}}\}$.

2.3.2　用户注册

用户向 SN 发送身份信息 Q_i 请求注册. 在接收到请求信息后,SN 为用户生成一个临时身份 $T_i\in\{0,1\}^*$,然后计算 $\mathrm{PK}_i=H_2(T_i)$ 作为用户的公钥,并计算用户私钥 $\mathrm{SK}_i=\mathrm{PK}_i^{\alpha}$. 最后 SN 通过安全信道向用户 U_i 发送临时身份 T_i 和密钥对(PK_i,SK_i).

2.3.3　安全 D2D 发现

用户注册完成后,利用各自的临时身份进行 D2D 发现. 首先用户 U_i 选择一个随机数 $r_i\in\mathbf{Z}_q^*$,计算 $V_i=g^{r_i}$,并生成请求信息 $M_i=s_{\text{req}}\parallel T_i\parallel V_i$,其中 s_{req}表示对特定 D2D 服务的请求,随后计算签名 $\sigma_i=\mathrm{PK}_i^{r_i}\mathrm{SK}_i^{H_1(M_i)}$,最后将$\{M_i,\sigma_i\}$广播出去. 在接收到用户 U_i 的广播消息后,临近用户验证等式 $e(\sigma_i,g)=e(V_iK_{\text{pub}}^{H_1(M_i)},\mathrm{PK}_i)$ 是否成立,若等式成立,则将 T_i 加入到 D2D 会话列表,并保存 PK_i. 假设有 n 个 D2D 用户 U_1、U_2、…、U_n 通过上述 D2D 发现过程发现彼此,则每个用户最终获得 D2D 会话列表 $L=\{T_1,T_2,\cdots,T_n\}$.

2.3.4　会话请求

用户 U_i 在获得 D2D 会话列表后,向 SN 发送 D2D 会话请求信息 $L_i^{\text{req}}=\{T_1,T_2,\cdots,T_n\}$. 当 SN 接收到所有的请求后,检查请求信息中用户身份是否为合法注册身份,若不是则拒绝 D2D 会话请求. 如果验证成功,SN 为所有合法用户创建一个 D2D 群组会话,选择一个随机数 $r_{\text{sid}}\in\mathbf{Z}_q^*$ 作为会话标识. 随后 SN 根据用户身份创建一个环形群组结构 $R_0=\{T_1,T_2,\cdots,T_i,\cdots,T_n\}$,其中每个用户 U_i 都有一对邻居 U_{i-1} 和 U_{i+1},$U_0=U_n$,$U_{n+1}=U_1$. 最后 SN 将$\{r_{\text{sid}},R_0\}$广播出去.

2.3.5　会话建立

用户接收到来自 SN 的会话响应消息后,进行 2 轮相互认证和密钥协商过程,得到共享会话密钥.

1) 首先用户 U_i 生成一个随机的秘密数 $x_i\in\mathbf{Z}_q^*$,并计算 $V_i^1=P_i=g^{x_i}$,生成第 1 次协商信息 $M_i^1=r_{\text{sid}}\parallel P_i\parallel T_i\parallel 0$. 随后用户 U_i 为消息 M_i^1 生成一个签名 $\sigma_i^1=\mathrm{PK}_i^{x_i}\mathrm{SK}_i^{h_i^1}$,其中 $h_i^1=H_1(M_i^1)$. 最后用户 U_i 将$\{M_i^1,\sigma_i^1\}$发送给用户 U_{i-1}、U_{i-2}和 U_{i+1},其中 $U_0=U_n$,$U_{-1}=U_{n-1}$,$U_{n+1}=U_1$.

2) 用户 U_i 接收到 U_j 发送来的第 1 次协商信息,其中 $j=i-1,i+1,i+2$,执行以下步骤:

① 验证 M_j^1 中的 r_{sid}是否和用户 U_i 持有的 r_{sid} 相同. 若相同,则继续;否则终止.

② 验证 T_j 是否为合法注册身份,以及序列参数是否为“0”. 若是,则继续;否则终止.

③ 当接收到用户 U_{i-1}、U_{i+1}和 U_{i+2}的消息后,计算 $h_j^1 = H_1(M_j^1)$.

④验证等式 $e(\sigma_j^1, g) = e(V_j^1 K_{\text{pub}}^{h_j^1}, \text{PK}_j)$是否成立,若等式成立,则继续;否则,向验证不通过的用户发出重新连接请求.

⑤ 认证成功后,用户 U_i 首先计算 $Y_i = e(P_{i+1}, (P_{i+2}/P_{i-1})^{x_i})$,然后选择一个随机数 $k_i \in \mathbf{Z}_q^*$,并计算 $V_i^2 = g^{k_i}$,生成第 2 次协商信息 $M_i^2 = r_{\text{sid}} \parallel Y_i \parallel T_i \parallel V_i^2 \parallel 1$. 随后用户 U_i 为消息 M_i^2 生成一个签名 $\sigma_i^2 = \text{PK}_i^{k_i} \text{SK}_i^{h_i^2}$,其中 $h_i^2 = H_1(M_i^2)$. 最后 U_i 将$\{M_i^2, \sigma_i^2\}$广播出去.

3) 用户 U_i 接收到 U_w 发送来的第 2 次协商信息,其中 $w = 1, 2, \cdots, i-1, i+1, \cdots, n$,执行以下步骤:

① 验证 M_w^1 中的 r_{sid}是否和用户 U_i 持有的 r_{sid} 相同. 若相同,则继续;否则终止.

② 验证 T_w 是否为合法注册身份,以及序列参数是否为“1”. 若是,则继续;否则终止.

③ 当接收到所有用户的消息后,计算 $h_w^2 = H_1(M_w^2)$.

④ 验证等式 $e(\sigma_w, g) = e(V_w^2 K_{\text{pub}}^{h_w^2}, \text{PK}_w)$是否成立,若等式成立, 则继续;否则,向验证不通过的用户发出重新连接请求.

⑤ 认证成功后,U_i 计算会话密钥 K_i 的过程为

$$k_{i-1,i,i+1} = e(P_{i+1}, P_{i-1}^{x_i})$$
$$k_{i,i+1,i+2} = k_{i-1,i,i+1} Y_i$$
$$\vdots$$
$$k_{i-2,i-1,i} = k_{i-3,i-2,i-1} Y_{i-2}$$
$$K_i = k_{1,2,3} k_{2,3,4} \cdots k_{n,1,2} =$$
$$e(g,g)^{x_1 x_2 x_3 + x_2 x_3 x_4 + \cdots + x_n x_1 x_2}$$

4) 在上述会话建立过程中,若用户 U_i 未接收到全部协商信息,则将未发送协商信息的用户身份 T 加入到失败信息列表 L_i^{fail} 中,并发送给 SN. 若 SN 接收到多个来自不同用户的失败信息,则将失败信息列表中对应身份的用户从 D2D 群组中移出,并重新生成会话标识和构建新的环形群组结构,然后发送给正常用户,最后群组中的用户再执行会话建立过程.

2.3.6 动态群组成员管理

群组中成员可能会发生变动,如用户撤销或加入. 为了确保群组内的通信安全,需要对群组会话密钥进行更新. 如果每次变动都重新进行一轮完整的协议过程,将增加通信开销和计算开销,降低用户的使用体验,因此实现动态的群组成员管理方案十分必要. 具体过程如下.

1) 用户撤销管理

当一个或多个用户被撤销时,设 $G^- = \{U_{l_1}, U_{l_2}, \cdots, U_{l_m}\}$为被撤销的用户集合,其中$\{l_1, l_2, \cdots, l_m\} \subset \{1, 2, \cdots, n\}$. 设 $G' = \{U_{l_1-2}, U_{l_1-1}, U_{l_1+1}, \cdots, U_{l_m-2}, U_{l_m-1}, U_{l_m+1}\}$为与撤销用户相邻的用户集合. SN 构建新群组结构 $R_{\text{new}}^- = \{T_{l_{m+1}}, T_{l_{m+2}}, \cdots, T_{l_n}\}$,其对应的用户集合为 G_{new}^-,其中$\{l_{m+1}, l_{m+2}, \cdots, l_n\} \subset \{1, 2, \cdots, n\}$. SN 将 R_{new}^-发送给群组成员,然后群组成员执行以下过程来更新会话密钥.

① 用户 $U_{i_1} \in G'$,选择一个新的随机秘密值 $x_{i_1} \in \mathbf{Z}_q^*$,计算 $P_{i_1} = g^{x_{i_1}}$,然后按照会话建立第 1 轮的过程生成第 1 次协商信息和签名$\{M_{i_1}^1, \sigma_{i_1}^1\}$,并依照新的群组结构 R_{new}^-发送给相邻的 3 个用户.

② 当用户 $U_{i_2} \in G_{\text{new}}^-$接收到第 1 次协商信息后,若验证签名通过,则按照会话建立第 2 轮的过程生成第 2 次协商信息和签名$\{M_{i_2}^2, \sigma_{i_2}^2\}$,然后广播出去.

③ 当用户集合 G_{new}^-中的用户接收到第 2 次协商信息后,首先验证其签名,若验证通过,则更新会话密钥为

$$K^- = e(g,g)^{x_{l_{m+1}} x_{l_{m+2}} x_{l_{m+3}} + x_{l_{m+2}} x_{l_{m+3}} x_{l_{m+4}} + \cdots + x_{l_n} x_{l_{m+1}} x_{l_{m+2}}}$$

2) 用户加入管理

当只有一个用户加入群组时,新加入的用户 U_{n+1}与原群组中的用户 U_n 执行以下过程来更新会话密钥:

① 用户 U_{n+1}与用户 U_n 执行 2 个用户之间的 Diffie-Hellman 密钥交换协议,从而更新会话密钥为

$$K^+ = g^{x_{n+1} H_1(K_n^G)}$$

式中,x_{n+1}为 U_{n+1}选择的随机秘密值;K_n^G 表示原群组的会话密钥.

② 用户 U_n 利用原群组会话密钥加密新的会话密钥 K^+并广播出去. 群组成员接收到消息后解密即可获得更新后的会话密钥.

当多个用户加入群组时,设 $G^+ = \{U_{n+1}, U_{n+2}, \cdots, U_{n+m}\}$为新加入的用户集合. SN 为用户构建群组结构 $R_{\text{new}}^+ = \{T_n, T_{n+1}, T_{n+2}, \cdots, T_{n+m}\}$,其对应的用户集合为 $G_{\text{new}}^+ = \{U_n, U_{n+1}, \cdots, U_{n+m}\}$. SN 将 R_{new}^+广播出去,然后群组成员执行以下过程

来更新会话密钥:

① 用户 $U_{i_3} \in G_{\text{new}}^{+}$选择一个随机秘密值 $x_{i_3} \in \mathbf{Z}_q^*$,计算 $P_{i_3} = g^{x_{i3}}$,其中 $x_n = H_1(K_n^G)$. 然后按照会话建立第 1 轮的过程生成第 1 次协商信息和签名 $\{M_{i_3}^1, \sigma_{i_3}^1\}$,并依照群组结构 R_{new}^{+}发送给相邻的 3 个用户.

② 当用户 $U_{i_3} \in G_{\text{new}}^{+}$接收到第 1 次协商信息后,若验证签名通过,则按照会话建立第 2 轮中的方式生成第 2 次协商信息和签名 $\{M_{i_3}^2, \sigma_{i_3}^2\}$,然后广播出去.

③ 用户集合 G^{+}中的用户在接收到第 2 次协商信息后,首先验证其签名,若验证通过,则更新会话密钥为

$$K^{+} = e(g,g)^{x_n x_{n+1} x_{n+2} + \cdots + x_{n+m-1} x_{n+m} x_n + x_{n+m} x_n x_{n+1}}$$

④ 用户 U_n 利用原群组会话密钥加密新的会话密钥 K^{+}并广播出去. 原群组成员接收到消息后解密即可获得更新后的会话密钥.

3　DG-AKA 协议方案的安全性分析

3.1　安全认证

定理 1　DG-AKA 协议方案可以实现安全认证.

证明　用户相互认证的过程中,发送方 U_i 发送协商信息和签名 $\{M_i, \sigma_i\}$,其中 $\sigma_i = \text{PK}_i^{x_i} \text{SK}_i^{H_1(M_i)}$,$V_i = g^{x_i}$,$x_i$ 为用户选择的随机秘密数. 接收方在接收到消息后,通过验证等式 $e(\sigma_i, g) = e(V_i K_{\text{pub}}^{H_1(M_i)}, \text{PK}_i)$来确认发送方 U_i 的身份. 其正确性如下所示:

$$\begin{aligned} e(\sigma_i, g) &= e(\text{PK}_i^{x_i} \text{SK}_i^{H_1(M_i)}, g) = \\ &\quad e(\text{PK}_i^{x_i}, g) e(\text{SK}_i^{H_1(M_i)}, g) = \\ &\quad e(\text{PK}_i, g^{x_i}) e(\text{PK}_i, g^{\alpha H_1(M_i)}) = \\ &\quad e(\text{PK}_i, V_i) e(\text{PK}_i, g^{\alpha H_1(M_i)}) = \\ &\quad e(V_i K_{\text{pub}}^{H_1(M_i)}, \text{PK}_i) \end{aligned}$$

接下来证明本文认证过程的安全性,可以通过引理 1 推出.

引理 1　在随机预言机模型下,若存在一个敌手 A 能够在多项式时间 t 内进行 q_{H_1} 和 q_{H_2}次对散列函数 H_1 和 H_2 的询问、q_{E} 次 Extract 询问和 q_{S} 次 Send 询问且以 ε 的优势赢得游戏,则存在一个算法,能够在多项式时间 t'内以 $\varepsilon' \geqslant \varepsilon(1 - 1/q_{\text{E}})/q$ 的优势解决 CDH 问题.

证明　首先根据 A 构造一个算法 C,它把 A 作为子程序并扮演 A 的挑战者. C 接收一个随机的 CDH 问题实例(g, g^a, g^b),$a, b \in \mathbf{Z}_q^*$,$g \in G_0$,其目标是计算 g^{ab}.

1) 初始化. C 设定 $K_{\text{pub}} = g^a$,a 是系统的主密钥,且 C 不知道 a. C 生成系统参数$\{G_0, G_1, e, H_1, H_2, g, q, K_{\text{pub}}\}$并发送给 A.

2) 阶段 1. A 接收到系统参数后,向 C 进行如下询问和挑战.

① H_2 询问. A 向 C 发出 $H_2(T^*)$询问,T^* 为被询问的用户身份,C 返回一个公钥 $\text{PK}^* = g^b$. 对于所有的 H_2 询问,C 选择一个随机数 $r_i \in \mathbf{Z}_q^*$,并将$\langle T_i, r_i, \text{PK}_i \rangle$插入到表 L_2 中,然后把 $\text{PK}_i = g^{r_i}$发送给 A.

② Extract 询问. A 向 C 发出 Extract(T_i)询问,如果 $\text{PK}_i = \text{PK}^*$,那么 C 输出 FAIL,然后终止模拟程序. 否则,C 在表 L_2 中找出一个三元组$\langle T_i, r_i, \text{PK}_i \rangle$,然后发送一个私钥 $K_{\text{pub}}^{r_i} = g^{ar_i} = g^{r_i a} = \text{PK}_i^a$ 给 A.

③ H_1 询问. A 向 C 发出 $H_1(M_i)$询问,M_i 为协商信息. C 返回一个随机数 $h_i \in \mathbf{Z}_q^*$,并发送给 A.

④ Send 询问. A 向 C 发出 Send(T_i)询问,C 选择一个随机数 $x_i \in \mathbf{Z}_q^*$,并计算 g^{x_i},然后将元组$\langle T_i, g^{x_i} \rangle$加入表 L_1 中. C 在表 L_2 找一个三元组$\langle T_i, r_i, \text{PK}_i \rangle$,然后计算 $\sigma_i = \text{PK}^{x_i} g^{ar_i h_i} = \text{PK}^{x_i} \text{SK}_i^{h_i}$,并将$\langle T_i, g^{x_i}, \sigma_i \rangle$发送给 A.

⑤ 伪造. 如果 A 发出 Corrupt(T^*)询问,那么 C 输出 FAIL,然后终止模拟程序. 否则按如下方式伪造出一个新的有效元组$\langle T^*, g^{x^*}, h, \sigma^* \rangle$,其中 x^* 为随机秘密数,h 为消息的哈希值,σ^* 为签名:任取 $x^* \in \mathbf{Z}_q^*$,计算 g^{x^*},然后选取满足等式 $e(g, \sigma^*) = e(\text{PK}_{T^*}, g^{x^*} K_{\text{pub}}^h)$的 σ^*,生成一个元组$\langle T^*, g^{x^*}, h, \sigma^* \rangle$,其中 PK_{T^*}为用户 T^*的公钥.

如果 A 能以不可忽略的优势 ε 在时间 t 内产生相应的元组赢得游戏,C 可以重复 A 的行为,通过控制随机预言机 H_1 的输出,在时间 $2t$ 内输出$\langle T^*, g^{x^*}, h, \sigma^* \rangle$和$\langle T^*, g^{x^*}, h', \sigma' \rangle$,其中,$h'$和 σ'分别表示预言机第 2 次输出的协商信息的哈希值和签名,$h \neq h'$. 显然,$g^{ab} = (\sigma^*/\sigma')^{(h-h')^{-1}}$,$C$ 便解决了 CDH 问题.

下面分析 C 的优势:

1) 如果事件 E 发生,C 模拟失败,E 表示 A 对目标身份 T^* 执行了 Extract 询问. 显然有 $\Pr(\bar{E}) = 1 - 1/q_{\text{E}}$.

2) 敌手在伪造签名时,需要在 G_0 中寻找满足等式 $e(g, \sigma^*) = e(\text{PK}_{T^*}, K_{\text{pub}}^h)$的 σ^*. 此概率不大

于 $1/q$.

综上所述,若算法 C 在模拟过程中未终止,并且敌手以不可忽略的优势 ε 赢得游戏,则算法 C 输出 CDH 困难问题的有效解优势是 $\varepsilon' \geqslant \varepsilon(1-1/q_{\mathrm{E}})/q$. 这显然与定义 2 相矛盾. 故定理 1 得证.

3.2 安全的密钥协商与托管

定理 2 基于随机预言机模型,将 $\mathrm{Adv}_A^{\mathrm{DG\text{-}AKA}}(t, q_{\mathrm{E}}, q_{\mathrm{S}})$ 定义为敌手 A 针对 DG-AKA 协议方案密钥协商过程的最大优势,$\mathrm{Adv}_{G_0,G_1,e}^{\mathrm{DBDH}}(t)$ 定义为敌手 A 针对 DBDH 假设难题的最大优势,$\mathrm{Adv}_{\Gamma}^{\mathrm{Forge}}(t)$ 定义为针对认证方案 Γ 的任意伪造算法在多项式时间 t 内的最大优势. 令 $\varepsilon_0 = \mathrm{Adv}_{G_0,G_1,e}^{\mathrm{MDBDH}}(t)$,则敌手 A 在多项式时间 t 内发出 q_{E} 次 Execute 询问和 q_{S} 次 Send 询问后,通过伪造签名或计算会话密钥来获得的优势有

$$\mathrm{Adv}_A^{\mathrm{DG\text{-}AKA}}(t, q_{\mathrm{E}}, q_{\mathrm{S}}) \leqslant 2nq_{\mathrm{E}}\,\mathrm{Adv}_{G_0,G_1,e}^{\mathrm{DBDH}}(t) + \mathrm{Adv}_{\Gamma}^{\mathrm{Forge}}(t) = 2nq_{\mathrm{E}}\varepsilon_0 + \varepsilon'$$

证明 首先,假设敌手 A 可以通过自适应的假冒攻击来赢得优势,那么就可以构造一个算法 C 来生成有效的消息元组$\langle T, g^x, h, \sigma\rangle$,其中 T 表示被伪造用户的身份,x 为用户选择的随机秘密数,h 为消息的哈希值,σ 为伪造后的签名. C 诚实地生成系统参数,然后模拟 A 的所有预言机询问,如果 A 发出 Corrupt(T)询问,那么 C 输出 FAIL,然后终止模拟程序. 否则,A 伪造出一个新的有效消息元组$\langle T, g^x, h, \sigma\rangle$,那么 C 生成有效的消息元组$\langle T, g^x, h, \sigma\rangle$. C 成功的概率满足 $\Pr_A[\mathrm{Forge}] \leqslant \mathrm{Adv}_{C,\Gamma}^{\mathrm{Forge}}(t) \leqslant \mathrm{Adv}_{\Gamma}^{\mathrm{Forge}}(t)$,在 3.1 节中已经证明了针对认证方案 Γ 的任意伪造算法在一个多项式时间 t 内有可以忽略的优势 $\mathrm{Adv}_{\Gamma}^{\mathrm{Forge}}(t) = \varepsilon'$.

接下来假设敌手 A 可以在不更改任何消息的情况下赢得优势. 根据文献[21]可得, MDBDH 和 DBDH 在计算上等效,即 $\mathrm{Adv}_{G_0,G_1,e}^{\mathrm{MDBDH}}(t) = \mathrm{Adv}_{G_0,G_1,e}^{\mathrm{DBDH}}(t)$. 首先考虑 A 只发出一次 Execute 询问 Execute$(T_1, T_2, \cdots, T_n)$,然后扩展到发出多次 Execute 询问的情形. n 是 A 选择的用户数,给出公共参数 S 和协商过程生成的参数集合 R 如下所示:

$$S = [(G_0, G_1, e) \leftarrow \mathrm{IG}_{\mathrm{BDH}}(1^k); g \leftarrow G_0; \alpha \leftarrow \mathbf{Z}_q^*; K_{\mathrm{pub}} = g^{\alpha}; \mathrm{PK}_1, \mathrm{PK}_2, \cdots, \mathrm{PK}_n \leftarrow G_0; \mathrm{SK}_1 = \mathrm{PK}_1^{\alpha}, \mathrm{SK}_2 = \mathrm{PK}_2^{\alpha}, \cdots, \mathrm{SK}_n = \mathrm{PK}_n^{\alpha}; (G_1, G_2, e, g, K_{\mathrm{pub}})]$$

$$R = [x_1, x_2, \cdots, x_n, h_1, h_2, \cdots, h_n \leftarrow \mathbf{Z}_q^*; P_1 = g^{x_1}, P_2 = g^{x_2}, \cdots, P_n = g^{x_n}; \sigma_1 = \mathrm{PK}_1^{x_1}\mathrm{SK}_1^{h_1}, \sigma_2 = \mathrm{PK}_2^{x_1}\mathrm{SK}_2^{h_1}, \cdots, \sigma_n = \mathrm{PK}_n^{x_n}\mathrm{SK}_n^{h_n}; Y_1 = \frac{e(P_2, P_3^{x_1})}{e(P_2, P_n^{x_1})}, Y_2 = \frac{e(P_3, P_4^{x_1})}{e(P_3, P_n^{x_1})}, \cdots, Y_n = \frac{e(P_1, P_2^{x_n})}{e(P_1, P_{n-1}^{x_n})}; I = \langle P_1, P_2, \cdots, P_n; \sigma_1, \sigma_2, \cdots, \sigma_n; Y_1, Y_2, \cdots, Y_n\rangle; k_{n,1,2} = e(P_2, P_n^{x_1}), k_{1,2,3} = k_{n,1,2}Y_1, \cdots, k_{n-1,n,1} = k_{n-2,n-1,n}Y_{n-1}; K = k_{1,2,3}k_{2,3,4}\cdots k_{n,1,2}: (I, K)]$$

式中,$\mathrm{IG}_{\mathrm{BDH}}$为一个安全参数为 1^k 的多项式时间算法,它在多项式时间内输出 2 个相同素数阶 q 的乘法循环群 G_0 和 G_1 以及一个双线性对 $e: G_0 \times G_0 \rightarrow G_1$; K 为真实的共享会话密钥.

定义如下参数分布 F_1:

$$F_1 = [r_{n,1,2}, x_1, x_2 \cdots, x_n \leftarrow \mathbf{Z}_q^*; P_1 = g^{x_1}, P_2 = g^{x_2}, \cdots, P_n = g^{x_2}; \sigma_1 = \mathrm{PK}_1^{x_1}\mathrm{SK}_1^{h_1}, \sigma_2 = \mathrm{PK}_2^{x_2}\mathrm{SK}_2^{h_2}, \cdots, \sigma_n = \mathrm{PK}_n^{x_n}\mathrm{SK}_n^{h_n}; Y_1 = \frac{e(P_2, P_3^{x_1})}{e(g,g)^{r_{n,1,2}}}, Y_2 = \frac{e(P_3, P_4^{x_2})}{e(P_3, P_1^{x_2})}, \cdots, Y_n = \frac{e(g,g)^{r_{n,1,2}}}{e(P_1, P_{n-1}^{x_n})}; I = \langle P_1, P_2, \cdots, P_n; \sigma_1, \sigma_2, \cdots, \sigma_n; Y_1, Y_2, \cdots, Y_n\rangle; k_{n,1,2} = e(P_2, P_n^{r_{n,1,2}}), k_{1,2,3} = k_{n,1,2}Y_1, \cdots, k_{n-1,n,1} = k_{n-2,n-1,n}Y_{n-1}; K = k_{1,2,3}k_{2,3,4}\cdots k_{n,1,2}: (I, K)]$$

同样的方式可得参数分布 F_n,即

$$F_n = [r_{n,1,2}, r_{1,2,3}\cdots, r_{n-1,n,1}, x_1, x_2 \cdots, x_n \leftarrow \mathbf{Z}_q^*; P_1 = g^{x_1}, P_2 = g^{x_2}, \cdots, P_n = g^{x_n}; \sigma_1 = \mathrm{PK}_1^{x_1}\mathrm{SK}_1^{h_1}, \sigma_2 = \mathrm{PK}_2^{x_2}\mathrm{SK}_2^{h_2}, \cdots, \sigma_n = \mathrm{PK}_n^{x_n}\mathrm{SK}_n^{h_n}; Y_1 = \frac{e(g,g)^{r_{1,2,3}}}{e(g,g)^{r_{n,1,2}}}, Y_2 = \frac{e(g,g)^{r_{2,3,4}}}{e(g,g)^{r_{1,2,3}}}, \cdots, Y_n = \frac{e(g,g)^{r_{n,1,2}}}{e(g,g)^{r_{n,n-1,1}}}; I = \langle P_1, P_2 \cdots, P_n; \sigma_1, \sigma_2, \cdots, \sigma_n; Y_1, Y_2, \cdots, Y_n\rangle; k_{n,1,2} = e(P_2, P_n^{r_{n,1,2}}), k_{1,2,3} = k_{n,1,2}Y_1, \cdots, k_{n-1,n,1} = k_{n-2,n-1,n}Y_{n-1}; K = k_{1,2,3}k_{2,3,4}\cdots k_{n,1,2}: (I, K)]$$

敌手 A 通过多次发出 Corrupt 和 H_1 询问, 可以获得全部长期密钥 SK_i 和协商信息的哈希值 h_i,从而计算 $\mathrm{PK}_i^{x_i} = \sigma_i/\mathrm{SK}_i^{h_i}$. 根据 F_1 可知,敌手 A 需要区分 $e(g,g)^{x_n x_1 x_2}$ 和 $e(g,g)^{r_{n,1,2}}$,这满足 MDBDH 问题. 根据 MDBDH 假设,在时间 t 内运行的任何区分算法 A 都有

$$|\Pr[(I,K) \leftarrow R; A(I,K) = 1] - \Pr[(I,K) \leftarrow F_1: A(I,K) = 1]| \leqslant \varepsilon_0$$

同理有

$$|\Pr[(I,K) \leftarrow F_1: A(I,K) = 1] - \Pr[(I,K) \leftarrow F_2: A(I,K) = 1]| \leqslant \varepsilon_0$$

$$\vdots$$

$$|\Pr[(I,K) \leftarrow F_{n-1}: A(I,K) = 1] - \Pr[(I,K) \leftarrow F_n: A(I,K) = 1]| \leqslant \varepsilon_0$$

令 $e(g,g) = f$,根据以下 n 个等式可知,随机数 $r_{1,2,3}, r_{2,3,4}, \cdots, r_{n,1,2}$的值受 I 的约束:

$$\log_f Y_1 = r_{1,2,3} - r_{n,1,2}$$
$$\log_f Y_2 = r_{2,3,4} - r_{1,2,3}$$
$$\vdots$$
$$\log_f Y_n = r_{n,1,2} - r_{n-1,n,1}$$

因此该方程组是线性无关的. 在 F_n 中,敌手计算的会话密钥为 $K_{F_n} = e(g,g)^{r_{1,2,3}+r_{2,3,4}+\cdots+r_{n,1,2}}$,方程两边取对数,可以得到$\log_f K_{F_n} = r_{1,2,3} + r_{2,3,4} + \cdots + r_{n,1,2}$,显然它独立于 I. 这意味着对于任何敌手 A,以下等式是成立的:

$$\Pr[(I,K) \leftarrow F_n : A(I,K) = 1] =$$
$$\Pr[I \leftarrow F_n; K \leftarrow K_R : A(I,K) = 1]$$

这表明敌手 A 无法通过 F_n 将会话密钥 K_{F_n}与相同长度的随机数 K_R 区分开.

同理,在 MDBDH 假设下,时间 t 内运行的任何算法都有

$$|\Pr[(I,K) \leftarrow F_n : A(I,K) = 1] - \Pr[I \leftarrow F_{n-1}; K \leftarrow K_R : A(I,K) = 1]| \leqslant \varepsilon_0$$
$$\vdots$$
$$|\Pr[I \leftarrow F_1; K \leftarrow K_R : A(I,K) = 1] - \Pr[I \leftarrow R; K \leftarrow K_R : A(I,K) = 1]| \leqslant \varepsilon_0$$

这表明,当敌手 A 截取它所选子集的所有消息时,区分真实会话密钥 K 与 K_R 的优势是 $2n\varepsilon_0$,即

$$|\Pr[I \leftarrow R; K \leftarrow R : A(T,K) = 1] - \Pr[I \leftarrow R; K \leftarrow K_R : A(I,K) = 1]| \leqslant 2n\varepsilon_0$$

由于 $\varepsilon_0 = \mathrm{Adv}^{\mathrm{MDBDH}}_{G_0,G_1,e}(t) = \mathrm{Adv}^{\mathrm{DBDH}}_{G_0,G_1,e}(t)$,因此 $\Pr_A[\sim \mathrm{Forge}] \leqslant 2n\mathrm{Adv}^{\mathrm{DBDH}}_{G_0,G_1,e}(t)$ 成立. 因此下式亦成立:

$$\mathrm{Adv}^{\mathrm{DG\text{-}AKA}}_A(t,1,q_S) \leqslant 2n\,\mathrm{Adv}^{\mathrm{DBDH}}_{G_0,G_1,e}(t) + \mathrm{Adv}^{\mathrm{Forge}}_\Gamma(t)$$

最终,在执行 q_E 次 Execute 询问和 q_S 次 Send 询问后,敌手 A 针对 DG-AKA 协议方案的最大优势为

$$\mathrm{Adv}^{\mathrm{DG\text{-}AKA}}_A(t,q_E,q_S) \leqslant 2nq_E\,\mathrm{Adv}^{\mathrm{DBDH}}_{G_0,G_1,e}(t) + \mathrm{Adv}^{\mathrm{Forge}}_\Gamma(t) = 2nq_E\varepsilon_0 + \varepsilon'$$

因敌手 A 针对 DG-AKA 协议方案的最大优势 $\mathrm{Adv}^{\mathrm{DG\text{-}AKA}}_A(t,q_E,q_S) \leqslant 2nq_E\varepsilon_0 + \varepsilon'$在多项式时间内是可忽略的,由此定理 2 得证. 定理 2 证明了 DG-AKA 协议方案能实现安全的密钥协商.

定理 3　DG-AKA 协议方案能够解决密钥托管问题.

证明　在密钥协商的过程中,密钥中的所有组件均由群组成员共同生成. 在定理 2 中已证明,即使攻击者获取了群组成员的长期私钥,它们也无法计算出正确的会话密钥. 而 SN 作为用户密钥生成和管理中心,只能获取用户的长期私钥,因此由定理 2 可知,SN 无法计算出共享会话密钥. 综上可知,DG-AKA 协议方案能解决密钥托管问题.

3.3　安全的动态群组成员管理

定理 4　DG-AKA 协议方案能实现安全的动态群组成员管理.

证明　动态群组成员管理分为以下 3 种情况.

1)有一个或多个用户被撤销时,可将群组划分为 2 部分,即与离开用户相邻的 G'中的群组成员以及其他未离开的群组成员. 在更新协议中对 G'中成员的秘密值进行了更新,撤销的用户无法获取这些秘密值,并且生成的新会话密钥不再含有撤销用户的任何密钥组件成分. 因此由定理 2 可知,在执行 q_E 次 Execute 询问和 q_S 次 Send 询问后,设 m 为未离开群组的用户数,敌手 A 针对 DG-AKA 协议方案用户撤销管理方案的最大优势为

$$\mathrm{Adv}^{\mathrm{DG\text{-}AKA\text{-}bs}}_A(t,q_E,q_S) \leqslant 2mq_E\mathrm{Adv}^{\mathrm{DBDH}}_{G_0,G_1,e}(t) + \mathrm{Adv}^{\mathrm{Forge}}_\Gamma(t) = 2mq_E\varepsilon_0 + \varepsilon'$$

式中,$\varepsilon_0 = \mathrm{Adv}^{\mathrm{DBDH}}_{G_0,G_1,e}(t)$,$\varepsilon' = \mathrm{Adv}^{\mathrm{Forge}}_\Gamma(t)$. 由于 $\mathrm{Adv}^{\mathrm{DG\text{-}AKA\text{-}bs}}_A(t,q_E,q_S) \leqslant 2mq_E\varepsilon_0 + \varepsilon'$在多项式时间内是可忽略的,因此根据定义 6,本方案在上述情况下满足群组后向安全.

2)只有一个用户加入群组,原群组中的一个成员与新加入的成员执行 2 个用户之间的 Diffie-Hellman 密钥交换协议,得到新的会话密钥,然后利用原群组会话密钥加密发送给原群组中的其他成员,从而更新群组会话密钥. 由定义 2 可知,群组外的用户仅根据 $g^{x_{n+1}}$和 $g^{H_1(K_n^G)}$ 无法计算出新的会话密钥 $g^{x_{n+1}H_1(K_n^G)}$. 在更新密钥的过程中,只有原群组中的一个成员参与了密钥的生成过程,并且该成员使用了新的秘密值. 因此,由定理 2 可知,在执行 q_E 次 Execute 询问和 q_S 次 Send 询问后,设 n 为原群组的用户数,敌手 A 针对 DG-AKA 协议方案中单个用户加入管理方案的最大优势为

$$\mathrm{Adv}^{\mathrm{DG\text{-}AKA\text{-}fs}}_A(t,q_E,q_S) \leqslant 2nq_E\,\mathrm{Adv}^{\mathrm{DBDH}}_{G_0,G_1,e}(t) + \mathrm{Adv}^{\mathrm{Forge}}_\Gamma(t) = 2nq_E\varepsilon_0 + \varepsilon'$$

因为 $\mathrm{Adv}^{\mathrm{DG\text{-}AKA\text{-}fs}}_A(t,q_E,q_S) \leqslant 2nq_E\varepsilon_0 + \varepsilon'$在多项式时间内是可忽略的,因此根据定义 5,该方案在上述情况下满足群组前向安全. 证毕.

3)有多个用户加入群组时,原群组中的一个成员与新加入成员之间执行 DG-AKA 协议方案得到会话密钥,然后利用原群组会话密钥加密发送给原群组中的其他成员,从而更新群组会话密钥. 由定理 2 可知,群组外的用户无法计算出新的会话密

钥. 同第 2 种情况,原群组中的一个成员使用新的秘密值来参与密钥的生成. 因此,由定理 2 可知,在执行 q_E 次 Execute 询问和 q_S 次 Send 询问后,设 u 为新加入群组的用户数,敌手 A 针对 DG-AKA 协议方案中多个用户加入管理方案的最大优势为

$$\mathrm{Adv}_A^{\text{DG-AKA-fs}}(t,q_E,q_S)\leqslant 2(u+1)q_E\mathrm{Adv}_{G_0,G_1,e}^{\text{DBDH}}(t)+\mathrm{Adv}_{\Gamma}^{\text{Forge}}(t)=2(u+1)q_E\varepsilon_0+\varepsilon'$$

因为 $\mathrm{Adv}_A^{\text{DG-AKA-fs}}(t,q_E,q_S)\leqslant 2(u+1)q_E\varepsilon_0+\varepsilon'$ 在多项式时间内是可忽略的,因此根据定义 5,该方案在上述情况下满足群组前向安全. 证毕.

3.4 安全性能比较

表 1 为方案 1、方案 2、方案 3 和 DG-AKA 协议方案安全性能的综合比较,由表可知,DG-AKA 协议方案满足所有安全性目标,相对于其他方案在安全性上更具优势.

表 1 安全性能比较

方案	抵抗伪造攻击	安全密钥协商	抵抗密钥托管问题	前向安全	后向安全
方案 1	×	×	√	√	√
方案 2	√	√	×	×	×
方案 3	√	√	√	×	×
本文方案	√	√	√	√	√

注:√表示满足安全要求;×表示不满足安全要求.

4 DG-AKA 协议方案的效率分析

4.1 计算开销

本节主要分析 SN 和用户的计算开销,DG-AKA 协议方案中 SN 负责用户注册和会话请求,用户则参与了协议的全部过程. 表 2 为 DG-AKA 协议方案不同阶段计算开销分析结果. 其中,T_R 表示取随机数运算,T_E 表示指数运算,T_H 表示哈希运算,T_M 表示乘法运算,T_P 表示双线性对运算.

表 2 DG-AKA 协议方案的计算开销

阶段	计算开销	计算复杂度
系统初始化	T_R+T_E	$O(1)$
用户注册	$n(T_H+T_E)$	$O(n)$
会话请求	T_R	$O(1)$
会话建立第 1 轮	$T_H+T_R+2T_E$	$O(1)$
会话建立第 2 轮	$4T_H+5T_E+T_M+7T_P$	$O(1)$
密钥生成	$(n-1)T_H+nT_E+2(n-1)T_M+nT_P$	$O(n)$

如表 2 所示,系统初始化阶段的复杂度为 $O(1)$. 在用户注册阶段,SN 要为 n 个用户生成临时身份和公/私钥对,因此总复杂度为 $O(n)$. 在会话请求阶段,SN 只需一次取随机数运算,复杂度为 $O(1)$. 在会话建立阶段第 1 轮和第 2 轮中,计算复杂度均为 $O(1)$. 而在密钥生成阶段,计算复杂度和用户数量成正比,此阶段的计算复杂度为 $O(n)$.

表 3 为方案 1、方案 2、方案 3 和 DG-AKA 协议方案计算开销的对比. 其中,方案 1 的计算量要低于 DG-AKA 协议方案,由于方案 1 中没有 SN 的参与,便没有用户与 SN 之间通信产生的计算开销,因此相比方案 1,DG-AKA 协议方案的计算开销较大. 但方案 1 没有实现安全认证,容易遭受伪造攻击. 同时,方案 2 的开销要大于 DG-AKA 协议方案,因为方案 2 只有哈希运算的计算量与 DG-AKA 协议方案相同,在指数、乘法、双线性对和取随机数运算上计算量均大于 DG-AKA 协议方案. 方案 3 在哈希运算和双线性对运算上计算量与 DG-AKA 协议方案相当,而乘法运算计算量比 DG-AKA 协议方案大,但在取随机数运算和指数运算上又略低于 DG-AKA 协议方案,因此 2 个方案的计算开销相当.

表 3 各方案计算开销

方案	计算开销
方案 1	$2T_R+4T_H+(2n+2)T_M+(n+3)T_P$
方案 2	$(2n+2)T_R+4nT_E+(2n+4)T_H+(5n+3)T_M+(3n+3)T_P$
方案 3	$3T_R+(n+10)T_E+(2n+6)T_H+5nT_M+(n+2)T_P$
本文方案	$(n+3)T_R+(2n+7)T_E+(2n+4)T_H+(2n-1)T_M+(n+7)T_P$

4.2 通信开销

为了分析通信开销,首先定义协议中每个参数的大小. 设 Q 和 T 的大小均为 16 B,r_{sid} 为 8 B,其余参数的大小均设置为 20 B. 将通信开销分为 2 类,即 SN 与用户之间的通信和用户与用户之间的通信. 在用户注册阶段,每个用户与 SN 的通信开销为 72 B,总开销为 $72n$ B. 在会话请求阶段,每个用户与 SN 的通信开销为 $(32n+8)$ B,总开销为 $n(32n+8)$ B. 在会话建立第 1 轮阶段,每个用户要与其他 3 个用户进行通信,通信开销为 194 B,总开销为 $194n$ B. 在会话建立第 2 轮阶段,每个用户都要将消息广播出去,通信开销为 $64(n-1)$ B,总开销为 $64n(n-1)$ B.

如图 2 所示,DG-AKA 协议方案的通信开销高于方案 1,因为方案 1 中没有 SN 的参与,只有用户与用户之间的通信开销. 但方案 1 没有实现安全认证,容易遭受伪造攻击. 同时,DG-AKA 协议方案的通信开销略高于方案 2,因为方案 2 的会话密钥是由 SN 生成和管理的,减少了用户与用户之间协商产生的通信开销. 但方案 2 不能解决密钥托管问题,并且没有实现安全的动态群组成员管理,不

能保证群组前向和后向安全. 最后,方案 3 和 DG-AKA 协议方案的通信开销相当,因为 2 个方案中均有 SN 与用户之间的通信以及用户与用户之间的通信,且这些通信过程的通信量相当. 但方案 3 没有实现安全的动态群组成员管理,需要借助安全信道.

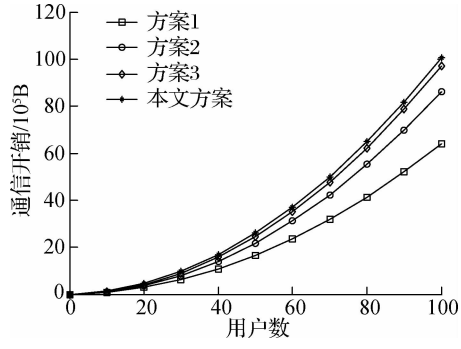

图 2　不同方案的通信开销对比

4.3　时间开销

本节通过仿真实验对运行 DG-AKA 协议方案的时间性能进行了测试. 测试中用到的参数 Q 和 T 的大小设为 16 B,r_{sid}为 8 B,其余参数的大小均设置为 20 B.

首先对协议的注册、会话请求和会话建立阶段进行了测试. 如图 3 所示,在用户注册和会话建立阶段,协议的执行时间随用户数的增加呈线性增长. 在会话建立阶段,需要执行大量的乘法、指数、哈希、双线性对和取随机数运算,并且在通信过程中每个用户需要与多个用户进行通信,因此会话建立阶段是整个协议中最耗时的部分,当用户数为 5 和 100 时,执行时间分别为 82.4 和 1 597.5 ms. 在用户注册阶段,只需少量的哈希和指数运算,当用户数为 5 和 100 时,执行时间分别为 16.5 和 296.7 ms. 在会话请求阶段,SN 只做一次取随机数运算,而对用户的合法性检验也只需对身份做简单的比较运算,因此该阶段执行时间是整个协议中最少的,当用户数为 5 和 100 时,执行时间分别为0.3 和 5.7 ms.

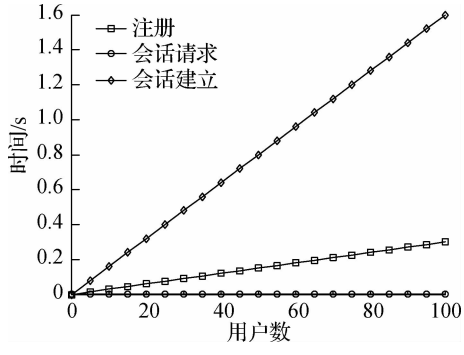

图 3　协议各阶段执行时间

为了分析不同协议的执行时间情况,本文对方案 1、方案 2、方案 3 和 DG-AKA 协议方案总的执行时间进行了对比(见图 4),其中总的执行时间包括注册、会话建立和会话请求 3 个阶段. 如图 4 所示,随着用户数的增加,不同方案均呈线性增长. 其中 DG-AKA 协议方案总的执行时间大于方案 1,因为 DG-AKA 协议方案中用户与 SN 通信需要额外的执行时间. 但方案 1 容易遭受伪造攻击,无法保证安全认证. 同时,DG-AKA 协议方案总的执行时间低于方案 2,因为方案 2 在会话建立阶段需要更多的指数、乘法、双线性对和取随机数运算,因此 DG-AKA 协议方案的执行时间更低,并且方案 2 没有实现安全动态群组管理. 最后,DG-AKA 协议方案总的执行时间略高于方案 3,DG-AKA 协议方案中每个用户要与相邻的 3 个用户通信,增加了一次通信和计算开销,但方案 3 没有实现真正的安全动态群组管理,需要借助安全信道来传递更新的密钥组件.

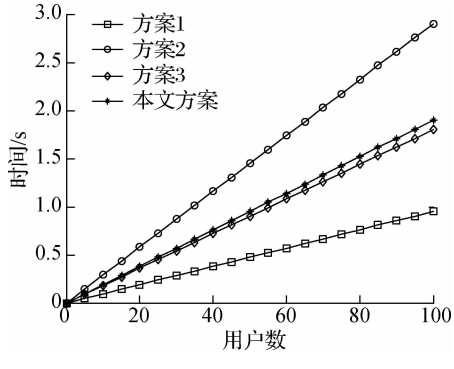

图 4　不同方案的总执行时间对比

5　结论

1) 基于 CDH 假设难题,实现了适用于 D2D 群组通信的安全认证过程. 群组用户利用临时身份进行相互认证,而非法用户无法伪造认证消息,从而能够抵抗伪造攻击.

2) 基于 MDBDH 假设难题,实现了一种安全的密钥协商过程. 密钥协商时的密钥组件均由群组成员生成,而 SN 无法获取密钥组件,从而解决了密钥托管问题. 通过结合安全的认证过程,DG-AKA 协议方案确保了会话密钥的安全性.

3) 在群组成员管理中,当有用户加入时,新加入的成员与原群组中的一个成员执行 DG-AKA 协议,而无需知道原群组成员的密钥组件,保证了新

密钥的安全性以及群组前向安全;当有用户被撤销时,与撤销用户相邻的用户更新其密钥组件,而撤销用户的密钥组件不再参与新密钥的生成,保证了新密钥的安全性以及群组后向安全.

参考文献(References)

[1] Doppler K, Rinne M, Wijting C, et al. Device-to-device communication as an underlay to LTE-advanced networks[J]. *IEEE Communications Magazine*, 2009, **47**(12): 42 - 49. DOI: 10.1109/mcom.2009.5350367.

[2] Doppler K, Yu C H, Ribeiro C B, et al. Mode selection for device-to-device communication underlaying an LTE-advanced network[C]//2010 *IEEE Wireless Communication and Networking Conference*. Sydney, Australia, 2010: 1 - 6. DOI: 10.1109/wcnc.2010.5506248.

[3] Lei L, Zhong Z D, Lin C, et al. Operator controlled device-to-device communications in LTE-advanced networks[J]. *IEEE Wireless Communications*, 2012, **19**(3): 96 - 104. DOI: 10.1109/mwc.2012.6231164.

[4] Wu X Z, Tavildar S, Shakkottai S, et al. FlashLinQ: A synchronous distributed scheduler for peer-to-peer ad hoc networks[J]. *ACM Transactions on Networking*, 2013, **21**(4): 1215 - 1228. DOI: 10.1109/tnet.2013.2264633.

[5] Gamage A T, Liang H, Zhang R, et al. Device-to-device communication underlaying converged heterogeneous networks[J]. *IEEE Wireless Communications*, 2014, **21**(6): 98 - 107. DOI: 10.1109/mwc.2014.7000977.

[6] Janis P, Yu C H, Doppler K, et al. Device-to-device communication underlaying cellular communications systems[J]. *International Journal of Communications, Network and System Sciences*, 2009, **2**(3): 169 - 178. DOI: 10.4236/ijcns.2009.23019.

[7] Alam M, Yang D, Rodriguez J, et al. Secure device-to-device communication in LTE-A[J]. *IEEE Communications Magazine*, 2014, **52**(4): 66 - 73.

[8] Shen W L, Hong W S, Cao X H, et al. Secure key establishment for Device-to-Device communications[C]//2014 *IEEE Global Communications Conference*. Austin, TX, USA, 2014: 336 - 340. DOI: 10.1109/glocom.2014.7036830.

[9] Sedidi R, Kumar A. Key exchange protocols for secure device-to-device (D2D) communication in 5G[C]//2016 *Wireless Days* (*WD*). Toulouse, France, 2016: 1 - 6. DOI: 10.1109/wd.2016.7461477.

[10] Wang M J, Yan Z, Niemi V. UAKA-D2D: Universal authentication and key agreement protocol in D2D communications[J]. *Mobile Networks and Applications*, 2017, **22**(3): 510 - 525. DOI: 10.1007/s11036-017-0870-5.

[11] Guo J J, Ma J F, Li X H, et al. An attribute-based trust negotiation protocol for D2D communication in smart city balancing trust and privacy[J]. *Information Science and Engineering*, 2017, **33**(4): 1007 - 1023. DOI: 10.6688/JISE.2017.33.4.10.

[12] Zhang A Q, Wang L, Ye X R, et al. Light-weight and robust security-aware D2D-assist data transmission protocol for mobile-health systems[J]. *IEEE Transactions on Information Forensics and Security*, 2017, **12**(3): 662 - 675. DOI: 10.1109/tifs.2016.2631950.

[13] Zhang A Q, Chen J X, Hu R Q, et al. SeDS: Secure data sharing strategy for D2D communication in LTE-advanced networks[J]. *IEEE Transactions on Vehicular Technology*, 2016, **65**(4): 2659 - 2672. DOI: 10.1109/tvt.2015.2416002.

[14] Hsu R H, Lee J, Quek T Q S, et al. GRAAD: Group anonymous and accountable D2D communication in mobile networks[J]. *IEEE Transactions on Information Forensics and Security*, 2018, **13**(2): 449 - 464. DOI: 10.1109/tifs.2017.2756567.

[15] Hsu R H, Lee J. Group anonymous D2D communication with end-to-end security in LTE-A[C]//2015 *IEEE Conference on Communications and Network Security* (*CNS*). Florence, Italy, 2015: 451 - 459. DOI: 10.1109/cns.2015.7346857.

[16] Tan H W, Song Y Z, Xuan S C, et al. Secure D2D group authentication employing smartphone sensor behavior analysis[J]. *Symmetry*, 2019, **11**(8): 969. DOI: 10.3390/sym11080969.

[17] Wang M J, Yan Z. Privacy-preserving authentication and key agreement protocols for D2D group communications[J]. *IEEE Transactions on Industrial Informatics*, 2018, **14**(8): 3637 - 3647. DOI: 10.1109/tii.2017.2778090.

[18] Wang L J, Tian Y L, Zhang D, et al. Constant-round authenticated and dynamic group key agreement protocol for D2D group communications[J]. *Information Sciences*, 2019, **503**: 61 - 71. DOI: 10.1016/j.ins.2019.06.067.

[19] Mustafa U, Philip N. Group-based key exchange for medical IoT device-to-device communication (D2D) combining secret sharing and physical layer key exchange[C]//2019 *IEEE 12th International Conference on Global Security, Safety and Sustainability* (*ICGS3*). London, United Kingdom, 2019: 42 - 48. DOI: 10.1109/icgs3.2019.8688022.

[20] Steinfeld R, Bull L, Wang H X, et al. Universal designated-verifier signatures[M]//*Advances in Cryptology-ASIACRYPT* 2003. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, **2894**: 523 - 542. DOI: 10.1007/978-3-540-40061-5_33.

[21] Choi K Y, Hwang J Y, Lee D H. Efficient ID-based group key agreement with bilinear maps[M]//*Public Key Cryptography-PKC* 2004. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, **2947**: 130 - 144. DOI: 10.1007/978-3-540-24632-9_10.

[22] Wang J M, Lang B. An efficient KP-ABE scheme for content protection in Information-Centric Networking[C]//2016 *IEEE Symposium on Computers and Communication* (*ISCC*). Messina, Italy, 2016: 830 - 837. DOI: 10.1109/iscc.2016.7543839.