

针对大数据安全的动态群密钥传输协议^{*}

刘建华¹, 邱修峰^{2,3}, 李妍⁴

(1. 中国民航飞行学院 航空工程学院, 四川 广汉 618307; 2. 北京航空航天大学 电子信息工程学院, 北京 100191; 3. 赣南师范学院 数学与计算机学院, 江西 赣州 341000; 4. 航天恒星科技有限公司, 北京 100086)

摘要: 为了确保基于大数据的群通信的安全性, 并提高通信效率和实用性, 提出了一种新的动态密钥传输协议。该协议允许任何一位群成员作为发起者分发群密钥, 整个密钥传输过程无须在线的可信中心, 且无须安全的通信信道。该协议的安全性基于 Diffie-Hellman 密钥协商协议以及线性秘密共享方案。当群成员发生变更时, 群通信发起者与其他群成员间共享的两方秘密无须更新, 能够很好地适应群成员的动态变化。该协议适用于许多基于大数据的面向群的应用。

关键词: 群密钥传输; 大数据安全; 密钥协商; 线性秘密共享; 面向群的应用

中图分类号: TP309.2

文献标志码: A

文章编号: 1001-3695(2016)05-1554-04

doi:10.3969/j.issn.1001-3695.2016.05.059

Dynamic group key transfer protocol for big data security

Liu Jianhua¹, Qiu Xiufeng^{2,3}, Li Yan⁴

(1. Aviation Engineering Institute, Civil Aviation Flight University of China, Guanghan Sichuan 618307, China; 2. School of Electronics & Information Engineering, Beihang University, Beijing 100191, China; 3. Dept. of Mathematics & Computer, Gannan Normal College, Ganzhou Jiangxi, 341000, China; 4. Space Star Technology Co., LTD, Beijing 100086, China)

Abstract: In order to meet the requirements of secure, efficient and practical group communication based on big data, this paper proposed a dynamic group key transfer protocol. The protocol allowed any group member to be an initiator to distribute a random number as a group key in the protocol. It could efficiently distributed group keys for group members without an online KGC. The Diffie-Hellman key agreement and a linear secret sharing scheme provided the security bases for our protocol. When the group members are changing, the protocol does not need to update the existing secret shared between the initiator and the other group members, thus, it is very efficient and is desirable for many group-oriented applications over big data.

Key words: group key transfer; big data security; key agreement; linear secret sharing; group-oriented application

0 引言

基于大数据的面向群的通信服务获得了快速发展, 如何确保基于大数据的安全的群通信是学术界和产业界共同关注的一个新问题^[1,2]。大数据具有数据规模大、数据类型多样、运算效率高等特点。基于的大数的群用户数量比传统的群用户数量更大, 用户的分布范围更广, 因此基于大数据的群通信安全方案与传统群组安全通信方案相比, 其对效率的要求更高。群密钥建立分为群密钥协商和群密钥传输两种类型。群密钥传输协议能够实现两个以上的通信实体获取一个共同的会话密钥, 用于之后加密等密码应用。群密钥传输的一般性安全要求包括密钥私密性、密钥的认证性、密钥的前向安全性等。与传统的群密钥传输协议相比, 适用于大数据安全的群密钥传输协议具有以下几个特点:

a) 对计算效率有更高的要求。大数据的群用户数量可能是传统群用户数量的几十倍、几百倍甚至更多, 在数量较少的

群上能够实施的群传输协议在大数据上就可能因为响应时间过长而让用户不能忍受, 甚至导致系统崩溃。

b) 各个群用户的协议实施开销差异尽可能小。特别是对于需要为其他群用户传输群密钥的实体(发起者), 要尽可能减少其相应的计算开销和存储开销, 否则协议的实施效率就会非常低下。而传统的群密钥传输协议由于用户数量相对较少, 对发起者的实施开销要求可以适当降低。

c) 信息传输的次数少。由于用户数量大、分布范围广, 如果协议中用户需要传输信息的次数过多, 将会导致响应等待时间过长。

Diffie-Hellman(DH)密钥协商协议能够为两方通信提供会话密钥^[3]。Shamir^[4]在1979年提出了群密钥分配协议。之后, 许多群密钥建立方案被提出^[2,5-8]。2010年, Harn等人^[6]提出了一种利用 (t, n) 秘密分享方案构造的群密钥传输协议, 该协议要求密钥生成中心(key generation center, KGC)以及每一位群成员计算 t 次内插多项式来加密及解密群密钥。该协

收稿日期: 2015-04-22; **修回日期:** 2015-06-08 **基金项目:** 国家自然科学基金资助项目(61272501); 中国民航飞行学院资助项目(J2013-31, Q2014-48); 国家“973”计划资助项目(2012CB315905); 中国民航大学天津市民用航空器适航与维修重点实验室开放基金资助项目; 中国民航飞行学院青年基金资助项目; 中国航天科技集团公司卫星应用研究院创新基金资助项目(2014-CXJJ-TX-10)

作者简介: 刘建华(1983-), 男, 讲师, 博士, 主要研究方向为信息安全、航空电子(ljh2583265@163.com); 邱修峰, 男, 讲师, 博士研究生, 主要研究方向为网络安全与仿真; 李妍(1988-), 女, 工程师, 硕士, 主要研究方向为卫星通信。

需要一个可信 KGC 为每一位群成员分配群密钥。2014 年 Hsu 等人^[2]利用线性秘密分享方案构造了一种适用于大数据传输安全的无须在线可信第三方的群密钥传输协议,但该协议给协议发起者 initiator 带来较大的计算量开销。

为了降低群成员的计算开销与存储开销,构造适用于大数据的群密钥建立协议,本文在 Hsu 等人^[2]的基础上设计了一个基于线性秘密共享方案的动态群密钥传输协议(dynamic group key transfer protocol based on linear secret sharing scheme, LSSS-DGKT),该协议结合了 DH 密钥协商协议及线性秘密共享方案的优点,无须在线 KGC,无需安全的通信信道,能高效地实现群成员的动态变化,有效地降低了系统实施的开销,能够适用于大数据中多种面向群的应用。

1 关于 LSSS-DGKT 协议的预备知识

LSSS-DGKT 群密钥传输协议的安全性是基于计算 Diffie-Hellman(CDH)假设和下面的线性秘密共享方案(LSSS)。

1.1 计算 Diffie-Hellman 假设

定义 1 计算 Diffie-Hellman 假设,CDH 假设^[3]。设 $G = \langle g \rangle$ 是一个阶为 q 的乘法循环群,给定 g, g^a, g^b , 其中 $a, b \in \mathbb{Z}_q^*$, 敌手在多项式时间内成功计算出 g^{ab} 是困难的。

1.2 线性秘密共享方案

在一个秘密共享方案中,由 n 个实体组成一个集合 $P = \{1, 2, \dots, n\}$ 。一个秘密 s 被分为 n 个秘密份额并分别被 n 个共享者拥有,只有拥有授权的 $t(t \leq n)$ 个用户才能够重构出该秘密 s 。当重构运算是线性运算时,称该方案是线性的。Karchmer 等人^[9]提出了一种计算单调布尔函数的线性模型。Beimel^[10]证明了设计一个线性秘密共享方案与构造一个单调张成方案是等价的。Hsu 等人^[2]利用 n 阶 Vandermonde 矩阵构造了一个线性秘密共享方案。具有较好的安全性,为了进一步提高群密钥传输的实施效率,LSSS-DGKT 群密钥传输协议被提出来。

LSSS-DGKT 密钥传输方案是在下面的线性秘密共享方案基础上构造的。设 $\bar{V} = K^n$ 是有限域 K 上一个 n 维线性空间。其中 $\text{char}(K) = p, p$ 是一个大素数。设 $\mathbf{e}_1 = (1, 0, 0, \dots, 0), \mathbf{e}_i = (0, \dots, 0, \underset{\text{第 } i \text{ 位}}{1}, 0, \dots, 0) (1 \leq i \leq n)$ 。设映射 $V^1(x, y) = x\mathbf{e}_1 + y\mathbf{e}_n, V^i(x, y) = -\mathbf{e}_{i-1} + x\mathbf{e}_i + y\mathbf{e}_n (x, y \in K, 1 < i < n), V^n(x, y) = -\mathbf{e}_{n-1} + (x+y)\mathbf{e}_n$ 。可得 n 阶矩阵

$$M_n = \begin{pmatrix} V^1(x, y_1) \\ V^2(x, y_2) \\ \dots \\ V^n(x, y_n) \end{pmatrix} = \begin{pmatrix} x & 0 & 0 & \dots & 0 & y_1 \\ -1 & x & 0 & \dots & 0 & y_2 \\ 0 & -1 & x & \dots & 0 & y_3 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & -1 & x+y_n \end{pmatrix}$$

设有 n 个秘密份额持有者 $P = \{P_1, P_2, \dots, P_n\}$ 和一个可信中心 D 。一个基于上述矩阵 M_n 的线性秘密共享方案(linear secret sharing scheme, LSSS)包括下面两个算法:

a) 两方共享秘密生成算法。

算法选择矩阵 M_n 以及一个随机向量 $r = (r_1, r_2, \dots, r_n)^T \in \bar{V}$, 并公开 r 。算法将进行如下计算得到 $s = (s_1, s_2, \dots, s_n)^T \in \bar{V}$ 。下面所有的计算是在有限域 K 上进行的。

$$M_n \cdot r = \begin{pmatrix} x & 0 & 0 & \dots & 0 & y_1 \\ -1 & x & 0 & \dots & 0 & y_2 \\ 0 & -1 & x & \dots & 0 & y_3 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & -1 & x+y_n \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ \dots \\ r_n \end{pmatrix} = \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ \dots \\ s_n \end{pmatrix}$$

然后,分别秘密地将 (x, y_i) 发送给秘密份额持有者 P_i 。

b) 秘密重构算法。

该算法将 $(x, y_1, y_2, \dots, y_n)$ 以及 $r = (r_1, r_2, \dots, r_n)^T$ 作为输入,计算内积 $(V^i(x, y_i), r) = s_i$ 然后输出秘密 $s = s_1 + s_2 + \dots + s_n$ 。

由于向量 $V(x, y_i) (1 \leq i \leq n)$ 是线性无关的,上面的方案满足如下秘密共享方案的基本要求^[9]:

(a) 任何人利用所有 n 份秘密份额 $V^i(x, y_i) (1 \leq i \leq n)$ 能够重构出秘密 s 。

(b) 在少于 n 份的秘密份额 $V^i(x, y_i) (1 \leq i \leq n)$ 的条件下,任何人不能够在多项式时间范围内重构出秘密 s 。

上面的秘密共享方案中,由于任意群成员在重构出群密钥时只需要进行最多 3 次模乘运算,记这种线性秘密共享方案为 3-LSSS。与基于 Vandermonde 矩阵线性秘密共享方案^[2]相比,3-LSSS 方案的效率更高。

2 LSSS-DGKT 协议

本文的群密钥传输方案由两部分组成:两方秘密建立阶段和群会话密钥传输阶段。假设有 n 个实体 $P = \{P_1, P_2, \dots, P_n\}$ 想要建立安全通信。可信中心 D 为每个实体安全地发送一组公私钥对: puk/prk , 满足 $\text{puk} = g^{\text{prk}} \bmod p$, 这里 $g \in \mathbb{Z}_p^*, p$ 是一个大素数。本文提出的密钥传输协议允许任何一个群成员发起群通信密钥传输协议,为叙述方便,假设发起协议的群组成员实体为 P_n 。

两方秘密建立阶段包括以下步骤:

a) 协议发起实体 P_n 随机选择 $r_n \in \mathbb{Z}_p^*$ 以及群组成员 $P = \{P_1, P_2, \dots, P_n\}$, 广播消息 $m_{nbl} = \{r_n, P = \{P_1, P_2, \dots, P_n\}, \text{puk}_n\}$ 。

b) $P = \{P_1, P_2, \dots, P_n\}$ 的每一个群成员 $P_i (1 \leq i < n)$ 选择随机数 $r_i \in \mathbb{Z}_p^*$ 并计算 $s_i = r_n r_i \text{puk}_n^{\text{prk}_i} \bmod p$, 然后计算 $\text{Auth}_i = h(s_i \| r_n)$, 发送 $\{r_i, P_i, \text{Auth}_i\}$ 给 P_n 。

c) 在收到来自各个群成员 $P_i (1 \leq i < n)$ 的消息 $\{r_i, P_i, \text{Auth}_i\}$ 后, P_n 计算 $s_i^* = r_n r_i \text{puk}_i^{\text{prk}_n} \bmod p$, 然后验证等式。如果等式 $\text{Auth}_i \stackrel{?}{=} h(s_i \| r_n)$ 成立,则 P_n 认定 $s_i = r_n r_i \text{puk}_i^{\text{prk}_n}$ 是与 $P_i (1 \leq i < n)$ 共享的两方秘密;否则, P_n 认为 $\{r_i, P_i, \text{Auth}_i\}$ 是非法的并重新发起该协议。

在会话密钥传输阶段,假设 $\bar{V} = K^n$ 是有限域 K 上的 n 维线性空间,且 $\text{char}(K) = p$ 。 $\mathbf{e}_i = (0, \dots, 0, \underset{\text{第 } i \text{ 位}}{1}, 0, \dots, 0) (1 \leq i \leq n)$ 是 $\bar{V}$ 一个基。定义映射 $V^i: K \times K \rightarrow \bar{V} (i = 1, \dots, n)$ 如下: $V^1(x, y) = x\mathbf{e}_1 + y\mathbf{e}_n, V^i(x, y) = -\mathbf{e}_{i-1} + x\mathbf{e}_i + y\mathbf{e}_n (x, y \in K, 1 < i < n), V^n(x, y) = -\mathbf{e}_{n-1} + (x+y)\mathbf{e}_n$ 。在与群成员 $P_i (1 \leq i < n)$ 建立两方共享秘密后,发起实体 P_n 需要随机选择一个群密钥,并把该密钥安全地传输给每个群成员。本文提出的协议在群密钥传输阶段的通信全部设定为开放的广播信道。会话密

钥传输阶段包括以下步骤:

a) 协议发起实体 P_n 将每个两方共享密钥分为 a_i 和 b_i 两部分, $a_i \parallel b_i = s_i$, 选择两个随机数 $x, K_C \in Z_p^*$, 其中 K_C 作为群会话密钥。然后, P_n 计算: $(a_i V^i(x, b_i), r) = K_i, U_i = K_C \oplus K_i$, $\text{Auth} = h(K_C, x, r_1, \dots, r_n, \dots, U_1, \dots, U_{n-1})$, h 是一个单向杂凑函数。 P_n 广播消息 $\{x, \text{Auth}, r_i, U_i\} (1 \leq i \leq n)$ 。

b) 对于每一个群成员, $P_i (1 \leq i \leq n)$, 利用接收到的 $\{x, \text{Auth}, r_i, U_i\}$, 计算: $(a_i V^i(x, b_i), r) = K_i$, 并恢复出群密钥 $\bar{K}_C = U_i \oplus K_i$ 。然后 P_i 计算 $h(\bar{K}_C, x, r_1, \dots, r_n, U_1, \dots, U_{n-1})$, 验证 $h(\bar{K}_C, x, r_1, \dots, r_n, U_1, \dots, U_{n-1}) \stackrel{?}{=} \text{Auth}$, 如果相等, 则 P_i 获得群会话密钥 $\bar{K}_C = K_C$; 否则, 终止协议。

在上面两个步骤成功执行后, 群成员之间建立了会话密钥 K_C 。

3 安全性分析

本章将证明本文提出的群密钥传输协议满足密钥新鲜性、密钥机密性和认证性等安全属性。密钥新鲜性是指所传输的密钥从未使用过, 能够确保已经泄露的密钥不能对现有的群通信造成安全威胁。密钥的机密性指群密钥仅能够被授权的合法用户计算出来, 非授权用户则不能计算群密钥。群密钥认证指实体能够确认是否与所声明的实体在进行群密钥传输。

本文考虑的安全模型中, 敌手分为内部敌手 $\mathcal{A}_1$ 和外部敌手 $\mathcal{A}_2$ 两种。 $\mathcal{A}_1$ 是群内部被授权获取群会话密钥的成员; $\mathcal{A}_1$ 企图恢复出 P_n 与其他成员的两方共享秘密, 或企图重构出某次不被授权获取的会话密钥。 $\mathcal{A}_2$ 是群外部未被授权获取会话密钥的实体; $\mathcal{A}_2$ 试图冒充某个群成员来获取两方共享秘密或会话密钥, 如 $\mathcal{A}_2$ 可能冒充 P_n 。本文将证明 $\mathcal{A}_2$ 不能获取两方共享秘密或会话密钥。

定理 1 本文提出的协议满足如下安全属性: 密钥新鲜性、机密性和密钥认证性。

证明 假设一个群由成员 $P = \{P_1, P_2, \dots, P_n\}$ 组成, P_n 为协议的发起者, 通过执行本文提出的协议所获取的两方共享秘密为 $S = \{s_1, s_2, \dots, s_n\}$ 。本文提出的密钥传输协议满足如下安全属性:

a) 密钥新鲜性。因为每次的会话密钥是由 P_n 选择的随机数, 并且用于恢复群密钥的计算式 $K_C = U_i \oplus K_i$ 是一个随机数与两方共享秘密的函数, 故密钥满足新鲜性。

b) 密钥机密性。密钥的私密性由 CDH 假设以及本文前面提出的线性秘密共享方案共同确保。由于两方共享秘密 $s_i = r_n r_i \text{puk}_n^{\text{prk}_i}$ 的计算是两个随机数与 $\text{puk}_n^{\text{prk}_i}$ 的乘积, 故两方共享秘密的私密性由 CDH 假设保证。利用本文前面提出的线性秘密共享方案, 发起者选择一个随机数以及群密钥, 计算出。每一个被授权的群成员 $P_i (1 \leq i < n)$, 拥有一个与发起者建立的两方共享秘密, 计算内积。故任意的授权成员能够成功地恢复出群会话密钥。对于任意的非授权敌手, 由于仅知道, 故不能获得关于以及的任何信息。所以任意非授权敌手不能获得群会话密钥。由于所提出的线性秘密共享方案是信息理论安全的, 故所提出的协议在密钥传输阶段也是信息理论安全的。

同时两方共享秘密是由两方 DH 密钥协商协议以及两个随机数得到, 然后秘密传输阶段利用两方共享秘密来建立当前

会话密钥, 即使当前会话密钥泄露, 敌手也不能获得以前的群密钥。因此, 所提出的协议满足前向保密性。

c) 密钥认证性。密钥的认证性通过杂凑函数来实现。在双方秘密建立阶段的步骤 b) 中 Auth_i 是一个单向杂凑函数, 其输入变量为共享秘密 s_i 与一个发起者 P_n 选择的随机数 r_n 。由于 s_i 仅有 P_n 与 P_i 拥有, 故非授权的敌手无法伪造 Auth_i 。此外, 在双方秘密共享阶段的步骤 b) 中, 由于 s_i 是由 P_n 与 P_i 通过加入随机数进行计算得到, 故敌手重放 $\{r_i, P_i, \text{Auth}_i\}$ 进行攻击将被发现。

在群密钥传输阶段的步骤 a) 中, Auth 是一个单向杂凑函数值, 其输入变量为群密钥 K_C 以及群成员选择的随机数等。仅有 P_n 及授权的群成员能计算出群密钥 K_C , 而非授权用户不能重构出 Auth 。由于群会话密钥是群成员与发起者之间的共享秘密的函数, 故任何的内部敌手都不能伪造群会话密钥。此外, 由于群密钥 K_C 是关于输入为每一个群成员的随机数的函数值, 任何重放的 $\{x, \text{Auth}, U_i\} (1 \leq i \leq n)$ 的攻击将被发现。

定理 2 外部敌手 $\mathcal{A}_2$ 不能获取群密钥。

证明 外部敌手 $\mathcal{A}_2$ 不知道群成员 P_i 的私钥 prk_i 以及发起者 P_n 的私钥 prk_n , 由 CDH 假设可得, $\mathcal{A}_2$ 不能计算出两方共享秘密 $s_i = r_n r_i g^{\text{prk}_n \text{prk}_i}$ 。并且, 群成员在重构群密钥 K_C 的算法中使用了本文前面提出的 3-LSSS 方案, 3-LSSS 算法的输入必须含有两方共享秘密 s_i , 任何不拥有 s_i 两方共享秘密的敌手 $\mathcal{A}_2$ 不能获取 K_i 及关于 $\sum_{i=1}^{n-1} K_i$ 的任何信息。另一方面, $\mathcal{A}_2$ 不知道发起者 P_n 的私钥 prk_n , 不能成功地冒充 P_n 获得与用户的两方共享秘密 s_i , 即敌手 $\mathcal{A}_2$ 不能通过冒充 P_n 获取群共享密钥 K_C 。

若敌手 $\mathcal{A}_2$ 利用已经泄露的群密钥进行重放攻击, 下面将证明该重放攻击不能使敌手 $\mathcal{A}_2$ 成功地与群成员建立群密钥。因为群密钥是由群成员选择的随机数以及两方共享秘密计算得到的, 如果每位群成员在每次群密钥传输协议中都选用不同的随机数, 则已经泄露的群密钥将能够被发现而不能重复使用。

定理 3 LSSS-DGKT 协议经过多次运行, 内部敌手 $\mathcal{A}_1$ 不能获取 P_i 与 P_n 的两方共享秘密 ($\mathcal{A}_1 \neq P_i, P_n$)。

证明 在密钥传输阶段的步骤 a) 中, 发起人 P_n 随机选择一个群密钥 K_C 并计算 $U_i = K_C \oplus K_i \bmod p (1 \leq i \leq n)$ 。每一个授权的用户利用计算内积 $(a_i V^i(x, b_i), r) = K_i$ 。因此任何的授权群成员 $P_i (1 \leq i < n)$ 都能够通过计算 $K_C = U_i \oplus K_i$ 重构出群密钥。并且, 根据 CDH 假设, 群成员 P_i 与发起者 P_n 之间的两方共享秘密 s_i 不能被敌手 $\mathcal{A}_1$ 获取。

一个内部敌手 $\mathcal{A}_1$ 能够获取群密钥 K_C 及 U_i , 但是 $\mathcal{A}_1$ 不能够从群密钥重构算法 $(a_i V^i(x, b_i), r) = K_i, K_C = U_i \oplus K_i$ 中获取关于 $s_i = a_i \parallel b_i$ 的任何信息, 因为这里存在两个未知参数 a_i, b_i 。同时两方共享秘密 s_i 是基于随机数对 (r_i, r_n) 以及长期私钥 $(\text{prk}_i, \text{prk}_n)$ 计算得到的, 故不能被任何内部敌手 $\mathcal{A}_1 (\mathcal{A}_1 \neq P_i, P_n)$ 获取。

4 效率分析

LSSS-DGKT 协议在进行密钥传输时无需在线 KGC, 同时也不要求群成员之间存在安全的通信信道。LSSS-DGKT 协议

允许任何群成员作为发起人发起密钥传输协议,群通信发起者与其他群成员之间通过 Diffie-Hellman 两方秘密共享协议来建立两方共享秘密。而目前大多数的基于门限秘密共享方案的密钥传输协议都要求一个可信中心来预先秘密地为每个用户分配一个共享秘密。

LSSS-DGKT 协议中,发起者与其他群成员之间通过引入随机数的 Diffie-Hellman 秘密共享协议建立两方共享秘密。添加和删除群用户无须对现有的两方共享秘密进行更新。当删除群用户时,发起者仅需要重新选择一个群通信密钥,并与群成员之间进行一次群密钥传输就能实现群用户的删除。添加用户时,发起者仅需要与新加入的用户建立两方共享秘密,然后将现有的密钥通过群密钥传输协议发送给群用户。由于该协议在增减用户时无须对现有的两方共享秘密进行更新,且除掉发起者外,其他群用户的开销很小。该协议能够高效地实现群成员的动态变化,适用于基于大数据的群通信。

在密钥传输阶段,LSSS-DGKT 协议使用了 3-LSSS 方案,这种方案与门限秘密共享方案(TSSS)^[6]相比,仅仅需要一次有限域上的内积运算,避免了 TSSS 中计算量更大的插值运算,因此具有更高的效率。并且,在重构群密钥时,3-LSSS 方案仅需要最多 3 次模乘运算,与需要 n 次模乘运算的基于 Vandermonde 矩阵线性秘密共享方案^[2]相比,3-LSSS 方案的效率更高。

记号 T_m 、 T_i 和 T_h 分别表示执行一次模乘运算、模逆运算和单向杂凑函数运算所花费的时间。与模乘和模逆运算相比,模加运算所需要的时间非常小,故下面的协议效率分析中不考虑模加运算的时间。

LSSS-DGKT 协议的计算复杂度:假设群有 n 个成员,给定两方共享秘密 $a_i \parallel b_i = s_i (i = 1, 2, \dots, n-1)$,群密钥 $K_G \in K$,随机数 x 以及随机数组成的向量 $r = (r_1, r_2, \dots, r_n)^T$,则发起者 P_n 分发群密钥 K_G 的时间复杂度为 $6(n-1)T_m + T_h$ 。每个群成员通过计算内积 $(a_i V^i(x, b_i), r) = K_i$ 恢复出群密钥,其时间复杂度为 $6T_m + T_h$ 。

在密钥传输阶段,本文提出的方案与的 Hsu 的协议^[2]及的 Harn 的协议^[6]比较如表 1 所示。

表 1 协议的计算量比较

协议	群密钥分配	群密钥恢复
LSSS-GKT 协议	$6(n-1)T_m + T_h$	$6T_m + T_h$
Hsu 的协议	$2(n-1)nT_m + T_h$	$2nT_m + T_h$
Harn 的协议	$n^2(n+1)(T_m + T_i) + T_h$	$n(n+1)n(T_m + T_i)T_h$

5 结束语

为了确保基于大数据的群通信安全,提出了一种高效的无须在线可信中心的动态群密钥传输协议 LSSS-DGKT 协议。该协议在增减群用户时无须对已有的两方共享秘密进行更新,有效地控制了群成员动态变化时的系统开销。在进行密钥分配及密钥恢复时,LSSS-DGKT 协议仅需进行一次内积运算。该协议的安全性基于 Diffie-Hellman 两方秘密共享方案及线性秘密共享方案。LSSS-DGKT 协议满足群密钥新鲜性、机密性及认证性,能够有效地抵抗内部敌手攻击和外部敌手攻击。LSSS-DGKT 协议能够适用于多种基于大数据的面向群的应用。

此外,LSSS-DGKT 协议也存在不足之处,例如,发起者的开销要大于其余的通信成员,对发起者的系统要求比较大。因此如何设计一种能够降低发起者系统开销的基于大数据的动态群密钥传输协议是今后研究的重点。

参考文献:

- [1] 冯登国,张敏,李昊. 大数据安全与隐私保护[J]. 计算机学报, 2014, 37(1): 246-258.
- [2] Hsu C F, Zeng Bing, Zhang Maoyuan. A novel group key transfer for big data security [J]. *Applied Mathematics and Computation*, 2014, 249: 436-443.
- [3] Diffie W, Hellman M E. New directions in cryptography[J]. *IEEE Trans on Information Theory*, 1976, 22(6): 644-654.
- [4] Shamir A. How to share a secret [J]. *ACM Communications*, 1979, 22(11): 612-613.
- [5] Wu Qianhong, Zhang Xinyu, Tang Ming, et al. Extended asymmetric group key agreement for dynamic groups and its applications [J]. *China Communications*, 2011, 32(7): 32-41.
- [6] Harn L, Lin C. Authenticated group key transfer protocol based on secret sharing [J]. *IEEE Trans on Comput*, 2010, 59(6): 842-846.
- [7] Hsu C F, Zeng Bing, Cheng Qi, et al. A novel group key transfer protocol, 2012/043 [R]. 2012.
- [8] Li C H, Pieprzyk J. Conference key agreement from secret sharing [C]//Proc of the 4th Australasian Conference on Information Security and Privacy. Berlin: Springer, 1999: 64-76.
- [9] Karchmer M, Wigderson A. On span programs [C]//Proc of Structure in Complexity Theory Conference. 1993: 102-111.
- [10] Beimel A. Secure schemes for secret sharing and key distribution [D]. [S. l.]: Technion-Israel Institute of Technology, Faculty of Computer Science, 1996.