

文章编号:1001-9081(2008)07-1798-04

动态协作对等组中一种新的组密钥协商协议

章丽平^{1,2}, 崔国华², 雷建云², 许静芳²

(1. 中国地质大学(武汉)计算机学院, 武汉 430074; 2. 华中科技大学计算机学院, 武汉 430074)

(carolyn321@163.com)

摘要: 动态协作对等组自身的特征使其安全机制面临着严峻的挑战。密钥协商机制则是构建安全的动态协作对等组的核心技术。提出了一种两方 Weil 对密钥协商协议($A\text{-WGKA}_2$), 可以通过较少的步骤同时实现节点之间的密钥协商和认证。该协议具备如下性质: 前向安全性; 抵抗未知密钥共享; 部分密钥泄露的安全性; 抵抗密钥控制; 抵抗使用泄露的密钥进行假冒攻击。在 $A\text{-WGKA}_2$ 协议的基础上, 进一步提出了一个新的适用于动态协作对等组的组密钥协商协议($A\text{-WGKA}_n$)。该协议在具有较低的计算和通信开销的同时, 实现了节点之间的相互认证, 适用于动态协作对等组。

关键词: 动态协作对等组; 组密钥协商; Weil 对; 网络安全

中图分类号: TP393.08 **文献标志码:** A

Novel group key agreement protocol for dynamic collaborative peer groups

ZHANG Li-ping^{1,2}, CUI Guo-hua², LEI Jian-yun², XU Jing-fang²

(1. College of Computer Science and Technology, China University of Geosciences (Wuhan), Wuhan Hunan 430074, China;

2. College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan Hunan 430074, China)

Abstract: To achieve security in a dynamic collaborative peer group, group key agreement protocol should be provided. Two-party key agreement protocol based on the weil pairing protocol ($A\text{-WGKA}_2$) was proposed in this paper. It may establish a secret key between two nodes and authenticate each other by fewer messages. The proposed protocol has the security properties such as forward secrecy, no unknown key-share, known session key security, no key control and no key-compromise impersonation. Further, we proposed a group key agreement protocol using weil pairing referred as $A\text{-WGKA}_n$ that was adapted to the dynamic collaborative peer groups. In the $A\text{-WGKA}_n$ protocol, the $A\text{-WGKA}_2$ protocol was employed on key tree to establish and allocate group key. Therefore, it not only has low computational overhead and communication costs but also provides the node authentication.

Key words: dynamic collaborative peer group; group key agreement; Weil pairing; network security

0 引言

动态协作对等组包括一组对等的节点, 这些节点通过共同协作来完成某项共同的任务。动态协作对等组中的每个节点都是对等的, 节点之间通过协作来完成一项任务, 这就使得攻击者可以针对协作过程实施新的攻击。动态的拓扑结构和移动节点数目的变化, 则使得节点间的信任关系发生改变。此外, 节点的移动性, 使得节点可能会因漫游到敌对环境中而被俘获破坏, 所以恶意攻击不仅可能来自组外也可能来自组内。因此, 动态协作对等组的安全性面临着严峻的挑战。

1976 年, W. Diffie 和 M. Hellman 提出了 Diffie-Hellman (D-H) 协议^[1]。此后, 在此基础上, 针对动态协作对等组, 提出了大量的组密钥协商协议^[2-5], 这些协议大多是在两方 Diffie-Hellman 密钥交换协议的基础上进行的扩展。Kim 等人对 Perring 等人提出的基于树的组密钥协商协议^[6]进行了改进, 提出了 TGDH 协议^[7]。该协议有效的降低了节点在进行组密钥协商时所需的计算量和通信量。此后, Depeng 等人基于 TGDH 协议的设计思想, 将椭圆曲线 Diffie-Hellman 密钥协商协议运用到二叉逻辑密钥树结构中^[8], 进一步降低了组密钥协商所需要的通信开销、计算开销和存储开销。但是, 以上

这些协议都不能很好地实现节点之间的相互认证。2006 年, Venkata 等人提出了一种可认证的两方椭圆曲线 Diffie-Hellman ($A\text{-ECDH}_2$) 密钥协商协议^[9]。该协议采用长期密钥机制实现了节点之间的相互认证。在 $A\text{-ECDH}_2$ 协议的基础上, 文献[9]中进一步提出了基于 ABKT 结构的组密钥协商协议 $A\text{-ECDH}_n$ 。并认为该协议是一个安全的, 可认证的, 分布式组密钥协商协议, 适用于动态协作对等组。但是, 文献[9]中所提出的 $A\text{-ECDH}_n$ 协议仍存在一些尚未解决的关键性问题。例如, 在成员加入过程中, $A\text{-ECDH}_n$ 协议如何让新成员和它将要与之协商密钥的组成员之间, 事先秘密享有共同的长期密钥。显然, 在该协议中, 这是实现可认证机制的一个复杂的关键性问题。

本文提出了一种新的两方 Weil 对密钥协商协议 ($A\text{-WGKA}_2$)。该协议满足密钥协商协议所应具备的性质: 完备的前向安全性; 抵抗未知密钥共享; 部分密钥泄露的安全性; 抵抗密钥控制; 抵抗使用泄露的密钥进行假冒攻击。并能够抵抗中间人攻击。在 $A\text{-WGKA}_2$ 协议的基础上, 基于二叉逻辑密钥树结构, 进一步提出了一个适用于动态协作对等组的组密钥协商协议 ($A\text{-WGKA}_n$)。该协议在具有较低的计算开销与通信开销的同时, 实现了节点间的相互认证。

收稿日期:2008-01-14;修回日期:2008-03-13。

基金项目:国家自然科学基金资助项目(60403027);中国地质大学(武汉)优秀青年教师资助计划资助项目(CUGQNL0836)。

作者简介:章丽平(1978-),女,湖北武汉人,讲师,博士研究生,主要研究方向:密码学、网络安全;崔国华(1947-),男,湖北武汉人,教授,博士生导师,主要研究方向:密码学、信息安全。

1 基于 Weil 对的两方密钥协商协议(A-WGKA₂)

两方 Weil 对密钥协商协议(A-WGKA₂)分为初始密钥协商过程和密钥更新过程。具体过程如下。

1.1 两方 Weil 对密钥协商协议

1) 初始密钥协商过程。在动态协作对等组中,假设每个合法节点都存储了秘密信息 $Q, R \in G_1$ 。设 u, v 为两个需要进行密钥协商的合法节点,其初始密钥协商过程如下:

第 1 步 节点 u 随机的选取一个整数 $r_u \in Z_q^*$, 并单播信息 $\{r_u P, r_u Q\}$ 给节点 v 。

第 2 步 节点 v 随机的选取一个整数 $r_v \in Z_q^*$, 并单播信息 $\{r_v P, r_v Q\}$ 给节点 u 。

节点 v 接收到信息 $\{r_u P, r_u Q\}$ 后, 验证等式 $e(r_u P, Q) = e(P, r_u Q)$ 是否成立。若等式成立, 则计算 $K_v = e(r_u Q, R)^{r_v} = e(Q, R)^{r_u r_v}$ 作为节点 u, v 所协商的会话密钥 $K_{u,v} = K_v$ 进行保存。若等式不成立, 则丢弃信息 $\{r_u P, r_u Q\}$ 。同理, 节点 u 接收到信息 $\{r_v P, r_v Q\}$ 后, 验证等式 $e(r_v P, Q) = e(P, r_v Q)$ 是否成立。若等式成立, 则计算 $K_u = e(r_v Q, R)^{r_u} = e(Q, R)^{r_u r_v}$ 作为节点 u, v 所协商的会话密钥 $K_{u,v} = K_u$ 进行保存。若等式不成立, 则丢弃信息 $\{r_v P, r_v Q\}$ 。该过程结束后, 节点 u, v 协商的会话密钥为 $K_{u,v} = K_u = e(r_v Q, R)^{r_u} = e(Q, R)^{r_u r_v} = e(r_u Q, R)^{r_v} = K_v$ 。初始密钥协商之后, 节点删除秘密信息 Q 和 R 。

2) 密钥更新过程。节点 u, v 在一段时间后将进行会话密钥的更新, 具体步骤如下:

第 1 步 节点 u 随机的选取一个新的整数 $r'_u \in Z_q^*$, 并单播信息 $\{r'_u P, r'_u H(K_{u,v})P\}$ 给节点 v 。

第 2 步 节点 v 随机的选取一个新的整数 $r'_v \in Z_q^*$, 并单播信息 $\{r'_v P, r'_v H(K_{u,v})P\}$ 给节点 u 。

节点 v 接收到信息 $\{r'_u P, r'_u H(K_{u,v})P\}$ 后, 验证等式 $e(r'_u P, H(K_{u,v})P) = e(P, r'_u H(K_{u,v})P)$ 是否成立。若等式成立, 则计算 $K'_v = e(r'_u H(K_{u,v})P, H(K_{u,v})^2 P)^{r'_v} = e(P, P)^{r'_v r'_u H(K_{u,v})^3}$ 作为节点 u, v 所协商的新会话密钥 $K'_{u,v} = K'_v$ 进行保存。同时删除旧的会话密钥。若等式不成立, 则丢弃信息 $\{r'_u P, r'_u H(K_{u,v})P\}$ 。同理, 节点 u 接收到信息 $\{r'_v P, r'_v H(K_{u,v})P\}$ 后, 验证等式 $e(r'_v P, H(K_{u,v})P) = e(P, r'_v H(K_{u,v})P)$ 是否成立。若等式成立, 则计算 $K'_u = e(r'_v H(K_{u,v})P, H(K_{u,v})^2 P)^{r'_u} = e(P, P)^{r'_u r'_v H(K_{u,v})^3}$ 作为节点 u, v 的新会话密钥 $K'_{u,v} = K'_u$ 进行保存, 同时删除旧的会话密钥。若等式不成立, 则丢弃信息 $\{r'_v P, r'_v H(K_{u,v})P\}$ 。密钥更新过程完成后节点 u, v 协商的新会话密钥为 $K_{u,v} = K'_u = e(r'_v H(K_{u,v})P, H(K_{u,v})^2 P)^{r'_u} = e(P, P)^{r'_u r'_v H(K_{u,v})^3} = e(r'_u H(K_{u,v})P, H(K_{u,v})^2 P)^{r'_v} = K'_v$ 。

1.2 两方 Weil 对密钥协商协议的安全性分析

两方 Weil 对密钥协商协议的安全性是基于椭圆曲线群上的 BDH 问题假设的。该协议具有如下性质: 完备的前向安全性; 抵抗未知密钥共享; 部分密钥泄露的安全性; 抵抗密钥控制; 抵抗使用泄露的密钥进行假冒攻击。

前向安全性: 假设敌手 A 获取了节点 u 或 v 产生当前会话密钥时所用的私钥。此时, A 要想得到节点 u, v 其他次密钥协商所产生的会话密钥, 则需要知道节点 u, v 进行该密钥协商时, 任何一方的私钥以及节点 u, v 当时所共有的秘密信息或者是节点 u, v 之前所协商的共享会话密钥。然而, 节点 u, v 在进行每次密钥协商时都会重新选择新的随机数作为其私钥,

要想获取当时任何一方的私钥就只有从当时传输的信息中进行私钥的提取, 而该过程等价于解决一个 ECDLP 问题。此外, 每次密钥协商过程结束后, 节点 u, v 都会删除旧的会话密钥。所以, 即使敌手 A 获取了当时任何一方的私钥, 他在不知道节点 u, v 当时所共有的秘密信息或是节点 u, v 之前所协商的共享会话密钥时, 仍然无法获取当时的会话密钥。因此, 即使敌手 A 获取了节点 u 或 v 产生当前会话密钥时所用的私钥, 也无法获取节点 u, v 其他次密钥协商所产生的会话密钥。

部分密钥泄露的安全性: 在执行两方 Weil 对密钥协商协议的过程中, 参与密钥协商的两个节点在每一次密钥协商过程中都分别选取一个唯一的新的随机数作为自己的私钥, 用来产生当时的会话密钥。因此, 每一次密钥协商都产生一个唯一的会话密钥。此外, 密钥更新过程中, 每次当新的会话密钥产生后, 旧的会话密钥都会被删除。因此, 即使敌手知道了某些会话密钥, 并且知道旧会话密钥与新会话密钥之间的关系, 也不能由关系获取他想要的会话密钥, 从而保证了某些会话密钥的泄漏不会暴露其他会话密钥信息。

抵抗密钥控制: 两方 Weil 对密钥协商协议中, 参与密钥协商的双方节点在每次密钥协商过程中都分别选取一个随机数作为自己的私钥, 用来产生它们之间的会话密钥。显然, 该会话密钥的产生是由参与密钥协商的双方节点共同决定的。这就保证了任何单独的一方都不能控制会话密钥的产生, 也不能强迫另一方接受自己预先所选取的值作为双方所共有的会话密钥。

抵抗未知密钥共享: 假设敌手 A 想要使节点 u 相信它们之间共享了会话密钥 K , 并且使节点 v 相信该会话密钥 K 是它与节点 u 所共享的。则敌手 A 需要完成如下操作: 1) 敌手 A 需要从截获的传输信息 $r_u Q, r_v Q$ 中提取秘密 Q , 来冒充节点 v 。2) 敌手 A 需要获取节点 u, v 之间的会话密钥 $K_{u,v}$ 。3) 获取了 $K_{u,v}$ 之后, 敌手 A 需要强迫节点 v 接受他预选选取的值作为双方所共享的会话密钥。而敌手 A 想要完成任务 1 和 2 相当于解决一个 ECDLP 问题。其分析与前向安全性与部分密钥泄露的安全性分析相同。此外, 由于该协议满足抵抗密钥控制性质, 所以敌手 A 无法完成任务 3。

抵抗使用泄露的密钥进行假冒攻击: 假设敌手 A 获取了节点 u 的所有私有会话密钥。那么敌手 A 可以冒充节点 u , 但是不能冒充其他节点。因为两方 Weil 对密钥协商协议保证了某个节点私有会话密钥的泄漏不会暴露其他会话密钥的信息。

两方 Weil 对密钥协商协议可以抵抗如下攻击: 1) 敌手 A 在两方 Weil 对密钥协商协议执行的第 1 步中, 用 $\{r_t P, r_t Q'\}$ 替换 $\{r_u P, r_u Q\}$ 发送给节点 v , 并试图与节点 v 建立会话密钥。2) 敌手 A 在两方 Weil 对密钥协商协议执行的第 2 步中, 用 $\{r_s P, r_s Q'\}$ 替换 $\{r_v P, r_v Q\}$ 发送给节点 u , 并试图与节点 u 建立会话密钥。3) 敌手 A 在两方 Weil 对密钥协商协议执行的第 1 步中, 用 $\{r_t P, r_t Q'\}$ 替换 $\{r_u P, r_u Q\}$ 发送给节点 v 。同时敌手 A 在协议执行的第 2 步中, 用 $\{r_s P, r_s Q'\}$ 替换 $\{r_v P, r_v Q\}$ 发送给节点 u 。实施中间人攻击。

针对第 1 种攻击: 敌手 A 在两方 Weil 对密钥协商协议执行的第 1 步中, 用 $\{r_t P, r_t Q'\}$ 替换 $\{r_u P, r_u Q\}$ 发送给节点 v 。节点 u 在接收到节点 v 发送的信息后可以正确计算出节点 u, v 之间的共享会话密钥 $K_{u,v} = e(Q, R)^{r_u r_v}$ 。而节点 v 接收到敌手 A 发送的信息后首先验证 $e(r_t P, Q)$ 和 $e(P, r_t Q')$ 是否相等。由于敌手 A 不知道秘密 Q , 则构造这两个等式相等的概率可以

忽略不计。假设通过了验证,节点 v 计算会话密钥得到 $K_{u,v} = e(Q, R)^{r_v}$ 。此时敌手 A 要想和节点 v 建立会话密钥就需要获取 $e(Q, R)^{r_v}$,但是敌手 A 不知道秘密 R ,因此他无法完成该攻击。密钥更新过程针对该攻击的安全性分析同上。

针对第 2 种攻击:第 2 种情况跟第 1 种情况类似,分析同上。

针对第 3 种攻击:假设敌手 A 用 $\{r_u P, r_u Q'\}$ 替换 $\{r_u P, r_u Q\}$ 并用 $\{r_s P, r_s Q''\}$ 替换 $\{r_s P, r_s Q\}$ 计算会话密钥,发起中间人攻击。由于两方 Weil 对密钥协商协议的第 1 步与第 2 步的信息传输是相互独立的,所以该中间人攻击可以看成是第 2 种攻击与第 3 种攻击的联合攻击。因此,该协议能够抵抗中间人攻击。

2 基于 Weil 对的组密钥协商协议

2.1 组密钥协商协议

基于 Weil 对的组密钥协商协议的基本思想是:首先,在初始化阶段采用最大匹配算法建立二叉逻辑密钥树^[8],并在所有合法节点中预先存储该树的结构以及秘密 Q 和 R 。然后,在该二叉逻辑密钥树结构上,运用两方 Weil 对密钥协商协议进行组密钥的协商。该协议中使用到的符号如下:

n :组规模,即该组中组成员的总数。

h :密钥树的深度。

L_i :第 i 层,其中 $i \in [1, h]$ 。

$U_{(L_i,j)}$:二叉逻辑密钥树 T 中第 L_i 层中的第 j 个节点, $i \in [0, n - 1]$ 。

$PV_{(L_i,j)}$:节点 $U_{(L_i,j)}$ 的私钥。

$PB_{(L_i,j)}$:节点 $U_{(L_i,j)}$ 的公钥。

如图 1 所示,假设密钥树 T 是一棵完全二叉树。设定 L_1 层为二叉树根节点所在的最高层, L_h 层为二叉树中的最底层。每一个节点 $U_{(L_i,j)}$ 用一对二元组 (L_i, j) 唯一标识,其中 L_i 表示该节点在二叉树中所在层, j 表示该节点在 L_i 层中从左到右的节点顺序编号。每一个节点都有自己的私钥 $PV_{U_{(L_i,j)}}$ 和公钥 $PB_{U_{(L_i,j)}}$ 。节点的公钥由其私钥计算得到 $PB_{U_{(L_i,j)}} = PV_{U_{(L_i,j)}} Q$ 。

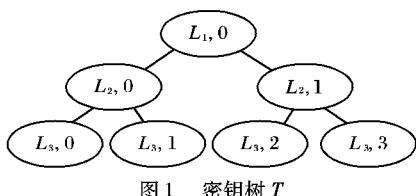


图 1 密钥树 T

密钥树 T 中有两类节点:一类是叶子节点,每一个叶子节点都与一个组成员相对应,代表该组成员。叶子节点的私钥定义如下: $PV_{U_{(L_h,j)}} = r_{(L_h,j)}$, 其中 $r_{(L_h,j)} \in Z_q^*$ 是节点 $U_{(L_h,j)}$ 随机选取的一个整数。另一类节点是拥有两个孩子节点的中间节点。它不代表任何一个组成员,而是代表一个子组。该子组中的任何子组成员都拥有该节点的信息。将中间节点的私钥作为该子组的子组密钥,其中间节点私钥的计算公式如下:

$$\begin{aligned} PV_{U_{(L_i,j)}} &= e(PB_{U_{(L_{i+1},2j)}}, R)^{PV_{U_{(L_{i+1},2j+1)}}} = \\ &e(PV_{U_{(L_{i+1},2j)}} Q, R)^{PV_{U_{(L_{i+1},2j+1)}}} = \\ &e(Q, R)^{PV_{U_{(L_{i+1},2j)}} PV_{U_{(L_{i+1},2j+1)}}} = \\ &e(PV_{U_{(L_{i+1},2j+1)}} Q, R)^{PV_{U_{(L_{i+1},2j)}}} = \\ &e(PB_{U_{(L_{i+1},2j)}}, R)^{PV_{U_{(L_{i+1},2j)}}} \end{aligned}$$

最终的组密钥即为根节点 $U_{(L_1,0)}$ 的私钥 $K = PV_{U_{(L_1,0)}}$ 。

密钥路径和兄弟路径的概念定义如下:从叶子节点 $U_{(L_h,j)}$ 到密钥树根节点的路径称为 $U_{(L_h,j)}$ 密钥路径 $KP_{U_{(L_h,j)}}$ 。 $U_{(L_h,j)}$ 拥有该路径上的所有节点信息。 $PV/KP_{U_{(L_h,j)}}$ 表示密钥路径 $KP_{U_{(L_h,j)}}$ 上的所有节点的私钥。 $PB/KP_{U_{(L_h,j)}}$ 表示密钥路径 $KP_{U_{(L_h,j)}}$ 上的所有节点的公钥。显然,所有组成员都拥有根节点 $U_{(L_1,0)}$ 的私钥,也就是最终的组密钥。叶子节点 $U_{(L_h,j)}$ 的兄弟路径由 $KP_{U_{(L_h,j)}}$ 路径上所有节点相对应的兄弟节点组成,称为 $SP_{U_{(L_h,j)}} \circ PV/SP_{U_{(L_h,j)}}$ 和 $PB/SP_{U_{(L_h,j)}}$ 则分别表示兄弟路径 $SP_{U_{(L_h,j)}}$ 上的所有节点的私钥和公钥。

在组密钥协商过程完成之后,密钥树 T 中的每个节点 $U_{(L_h,j)}$ 都将存储 $PV/KP_{U_{(L_h,j)}}$ 。在一段时间之后,采用两方 Weil 对密钥协商协议的更新算法,对组密钥进行更新。此时节点公钥的计算公式为: $PB_{U_{(L_i,j)}} = PV_{U_{(L_i,j)}} H(K_{U_{(L_i,j)}, U_{(L_i,j+1)}}) P$ (其中,当 j 为奇数时取减号,当 j 为偶数时取加号)。中间节点私钥的计算公式为 $PV_{U_{(L_i,j)}} = e(PB_{U_{(L_{i+1},2j)}})$, $H(K_{U_{(L_{i+1},2j)}, U_{(L_{i+1},2j+1)}})^2 P)^{PV_{U_{(L_{i+1},2j+1)}}}$ 。

2.2 组成员加入和退出

在成员加入过程中,需要借助 IBE 加密算法来抵抗中间人攻击。因此,初始化阶段,需要在每个合法节点中预选存储 IBE 算法中的参数。

1) 组成员加入。当合法新成员 U_{n+1} 请求加入组,此时密钥树为满二叉树,则位于该密钥树最底层的最左叶子节点负责新成员的加入。否则由密钥树倒数第二层的最左叶子节点负责新成员的加入。 $U_{(L_h,j)}$ 执行如下操作:

① 创建两个节点。

② 复制自身节点,分配节点号 $(L_{h+1}, 2j)$,并将该节点作为节点 $U_{(L_h,j)}$ 的左孩子节点。同时给新成员 U_{n+1} 分配节点号 $(L_{h+1}, 2j + 1)$,作为其右孩子节点。

③ 选取 $Q', R' \in G_1$,并随机选取一个整数 $r_{U_{(L_h,j)}} \in Z_q^*$,采用 IBE 算法,用新成员 U_{n+1} 的公钥 $ID_{U_{n+1}}$ 加密信息 $\{r_{U_{(L_h,j)}} P, r_{U_{(L_h,j)}} Q', Q', R'\}$,并单播给新成员 U_{n+1} 。新成员 U_{n+1} 在接收到组成员 $U_{(L_h,j)}$ 发送的信息后,用自己的私钥解密该信息获取 $\{r_{U_{(L_h,j)}} P, r_{U_{(L_h,j)}} Q', Q', R'\}$ 。然后,新成员 U_{n+1} 随机选取一个整数 $r_{U_{n+1}} \in Z_q^*$,计算 $K_{U_{n+1}} = e(r_{U_{(L_h,j)}} Q', R')^{r_{U_{n+1}}} = e(Q', R')^{r_{U_{(L_h,j)}} r_{U_{n+1}}}$,作为与 $U_{(L_h,j)}$ 的共享会话密钥。同时,新成员 U_{n+1} 计算 $\{r_{U_{n+1}} P, r_{U_{n+1}} Q'\}$,并将该信息单播给 $U_{(L_h,j)}$ 。 $U_{(L_h,j)}$ 在接收到 U_{n+1} 发送的信息后,验证等式 $e(r_{U_{n+1}} P, Q') = e(P, r_{U_{n+1}} Q')$ 是否成立。如果该等式成立,则计算 $K_{U_{(L_h,j)}} = e(r_{U_{n+1}} Q', R')^{r_{U_{(L_h,j)}}} = e(Q', R')^{r_{U_{n+1}} r_{U_{(L_h,j)}}}$,作为与新成员共享的会话密钥 $K_{U_{(L_h,j)}} = K_{U_{n+1}, U_{(L_h,j)}}$ 进行保存。如果等式不成立,则丢弃信息 $\{r_{U_{n+1}} P, r_{U_{n+1}} Q'\}$ 。若与新成员建立了会话密钥,则更新密钥路径 $KP_{U_{(L_{h+1},2j)}}$ 上的所有密钥,同时向全组广播更新后的公钥以及密钥树 T 的结构。

这样,组中所有其他组成员在接收到 $PB/KP_{U_{(L_{h+1},2j)}}$ 后,可以获取该密钥路径上更新后的所有私钥,从而计算出新的组密钥。一次组成员加入操作最多会使密钥树的深度加 1,因此组成员加入操作的时间复杂度为 $O(\log n)$ 。此外,负责成员加入的组成员是分布在全组中的,避免了某个组成员承担过多的操作而造成的协议性能下降。

2) 组成员离开。当有组成员 $U_{(L_i,j)}$ 退出时,它的邻节点 $U_{(L_i,j-1)}$ 执行如下操作:

①重新选取一个随机整数 $r_{U_{(L_i,j-1)}} \in Z_q^*$ 作为其新的私钥,并计算其相应的公钥。

②用其更新后的私钥和公钥替换其双亲节点的私钥和公钥。

③计算其双亲节点到根节点路径上的所有公钥。

④设置组成员 $U_{(L_i,j)}$ 的值为 -1 (在密钥树 T 中,若节点的值为 -1,则表示该节点为傀儡节点),更新密钥树 T 的结构。并将更新后的密钥树 T 的结构和更新后的所有公钥广播给全组。

2.3 基于 Weil 对的组密钥协商协议的安全性

组成员加入协议的安全性:新成员 U_{n+1} 要想成功的加入该组,就必须和负责新成员加入的某个组成员 $U_{(L_h,j)}$ 建立共享会话密钥。该会话密钥协商的安全性是基于 IBE 算法的安全性以及两方 Weil 对密钥协商协议的安全性的。IBE 算法被证明是 IND-ID-CCA 安全的^[12],从而保证了组成员 $U_{(L_h,j)}$ 发送给新成员 U_{n+1} 的秘密信息 Q', R' 只有 $U_{(L_h,j)}$ 和新成员 U_{n+1} 知道。组成员 $U_{(L_h,j)}$ 和新成员 U_{n+1} 随后采用两方 Weil 对密钥协商协议进行密钥协商和密钥更新。由两方 Weil 对密钥协商协议的安全性知,成员加入协议同样具有前向安全性;抵抗未知密钥共享;部分密钥泄露的安全性;抵抗密钥控制;抵抗使用泄露的密钥进行假冒攻击。

组成员退出协议的安全性:当有组成员退出时,成员退出协议的执行将产生一个新的组密钥。并且,每一次协议的执行所产生的新的组密钥都是唯一的。因此新的组密钥是独立于之前的所有组密钥的。此外,新的组密钥在构建过程中所涉及到的话密钥的更新过程,采用的是两方 Weil 对密钥协商协议中的更新过程。因此,退出的组成员不能够根据其所拥有的会话密钥和旧会话密钥与新会话密钥之间的关系,来获取它退出后的新的组密钥。由以上分析,组成员退出协议具有部分密钥泄露的安全性。

3 性能分析

基于 Weil 对的组密钥协商协议在密钥协商过程中,与 TGDH 协议, GDH2 协议在计算复杂度和通信复杂度方面的比较如表 1 所示。从表中可以看出本文提出的 A-WGKA_n 协议与 GDH2 协议相比,显著降低了组成员 U_n 的指数运算次数以及接收的消息数。此外, GDH2 协议中,尽管组成员 U_n 只发送了一条消息,但该消息中包含 $n-1$ 个 1024 位的密钥信息。A-WGKA_n 协议与 TGDH 协议相比,虽然它们具有相同的通信代价和计算代价,但是 A-WGKA_n 协议可以同时实现组密钥的协商和节点之间的相互认证。

表 1 基于 Weil 对的组密钥协商协议中的通信代价和计算代价

协议	计算代价	通信代价	
		发送消息数	接收消息数
TGDH	$\log n$	$\log n$	$\log n$
GDH	$U_1 - U_{n-2}$	3	2
	U_{n-1}	2	2
A-WGKA _n	U_n	1	$n-1$
	$\log n$	$\log n$	$\log n$

成员退出过程中,TGDH 协议的计算代价为 $O(\log n)$, GDH2 协议的计算代价为 n , A-WGKA_n 协议的计算代价为

$O(\log n)$ 。与 GDH2 协议相比,本协议与 TGDH 协议都有效的降低了成员退出操作所需的计算代价。在成员加入过程中,TGDH 协议的计算代价为 $O(\log n)$, GDH2 协议的计算代价为 n 。而 A-WGKA_n 协议的计算代价为 $O(\log n)$ 加上一次 IBE 的加解密计算代价。虽然与 TGDH 协议相比 A-WGKA_n 协议在成员加入过程中其计算代价略有增加,但是 A-WGKA_n 协议可以抵抗中间人攻击,实现组成员之间的相互认证,增强了动态协作对等组的安全性。

4 结语

密钥协商机制是构建安全的动态协作对等组的基础。本文提出了一种两方 Weil 对密钥协商协议(A-WGKA₂),通过较少的步骤同时实现节点之间的密钥协商和认证。在 A-WGKA₂ 协议的基础上,进一步提出了一个新的适用于动态协作对等组的组密钥协商协议(A-WGKA_n)。与其他协议相比,该协议在具有较低的计算和通信开销的同时,能够较好地实现节点之间的相互认证,适用于动态协作对等组。它的缺点是在组成员加入过程中需要采用 IBE 算法来实现节点间的认证,从而增加了一定的计算量。在今后的工作中,我们将就如何在实现节点之间相互认证的前提下进一步降低通信量和计算量的问题做进一步的研究。

参考文献:

- [1] DIFFIE W, HELLMAN M. New directions in cryptography [J]. IEEE Transactions on Information Theory, 1976, 22(6): 644 - 654.
- [2] ASOKAN N, GINZBOORG P. Key agreement in Ad Hoc networks [J]. Computer Communications, 2000, 23(17): 1627 - 1637.
- [3] ATENIESE G, STEINER M, TSUDIK G. New multiparty authentication services and key agreement protocols[J]. IEEE Journal on Selected Areas in Communications, 2000, 18(4): 628 - 639.
- [4] STEINER M, TSUDIK G, WAIDNER M. Key agreement in dynamic peer groups[J]. IEEE Transactions on Parallel and Distributed Systems, 2000, 11(8): 769 - 780.
- [5] RODEH O, BIRMAN K P, DOLEV D. Using AVL trees for fault-tolerant group key management[J]. International Journal on Information Security, 2002, 1(2): 84 - 99.
- [6] PERRIG A. Efficient collaborative key management protocols for secure autonomous group communication[C]// International Workshop on Cryptographic Techniques and Electronic Commerce. Hong Kong: City University of Hong Kong Press, 1999: 192 - 202.
- [7] KIM Y, PERRING A, TSUDIK G. Tree-based group key agreement [J]. ACM Transaction on Information and System Security, 2004, 7 (1): 60 - 96.
- [8] LI DE-PENG, SAMPALLI S. An efficient group key establishment in location-aided mobile Ad Hoc networks[C]// PE-WASUN' 05. New York: ACM Press, 2005: 57 - 64.
- [9] VENKATA C, SAIKAT C, SINGHAL M. A distributed multi-party key agreement protocol for dynamic collaborative groups using ECC [J]. Journal of Parallel and Distributed Computing, 2006, 66(7): 959 - 970.
- [10] SONG B, KIM K. Two-Pass authenticated key agreement protocol with key confirmation[C]// ROY B K, OKAMOTO E, eds. Proceeding of the Indocrypt 2000. Berlin, Heidelberg: Springer-Verlag, 2000: 237 - 249.
- [11] YAO GANG, FENG DENG-GUO. Pairwise key agreement protocols based on the weil pairing[J]. Journal of Software, 2006, 17(4): 907 - 914.
- [12] BONECH D, FRANKLIN M. Identity - based encryption from the Weil pairing [C]// Advances in Cryptology-Crypto'2001, LNCS 2139. [S. l.]: Springer-Verlag, 2001: 213 - 229.