

文章编号:1001-9081(2018)05-1372-05

DOI:10.11772/j.issn.1001-9081.2017102413

## 抗合谋攻击能力可调的有状态组密钥更新协议

敖丽<sup>1</sup>, 刘璟<sup>1\*</sup>, 姚绍文<sup>1</sup>, 武楠<sup>2</sup>

(1. 云南大学 软件学院, 昆明 650500; 2. 云南大学 信息学院, 昆明 650500)

(\*通信作者电子邮箱 liujing@ynu.edu.cn)

**摘要:**逻辑密钥分层(LKH)协议已经被证明在抗完全合谋攻击时,它通信开销的下界是  $O(\log n)$ ,但是在一些资源受限或者商业应用场景中,用户仍然要求通信开销低于  $O(\log n)$ 。虽然,有状态的完全排外子树(SECS)协议具有常量通信开销的特性,却只能抵抗单用户攻击。考虑用户愿意牺牲一定安全性来降低通信开销的情况,利用LKH协议的完全抗合谋攻击特性和SECS协议具有常量通信开销的优势,设计并实现了一种混合的组密钥更新协议(H-SECS)。H-SECS协议根据应用场景的安全级别来配置子组数目,在通信开销和抗合谋攻击能力之间做一个最优的权衡。理论分析及仿真实验表明,与LKH协议和SECS协议相比,H-SECS协议的通信开销可以在  $O(1)$  和  $O(\log n)$  区间进行调控。

**关键词:**有状态;组密钥更新;逻辑密钥分层协议;抗合谋攻击

**中图分类号:**TP393.08      **文献标志码:**A

### Stateful group rekeying scheme with tunable collusion resistance

AO Li<sup>1</sup>, LIU Jing<sup>1\*</sup>, YAO Shaowen<sup>1</sup>, WU Nan<sup>2</sup>

(1. School of Software, Yunnan University, Kunming Yunnan 650500, China;

2. School of Information Science and Engineering, Yunnan University, Kunming Yunnan 650500, China)

**Abstract:** Logical Key Hierarchy (LKH) protocol has been proved that  $O(\log n)$  is the lower bound of the communication complexity when resisting complete collusion attacks. However, in some resource-constrained or commercial application environments, user still require the communication overhead below  $O(\log n)$ . Although Stateful Exclusive Complete Subtree (SECS) protocol has the characteristic of constant communication overhead, but it can only resist single-user attacks. Considering the willingness of users to sacrifice some security to reduce communication overhead, based on LKH which has the characteristic of strict confidentiality, and combined with SECS which has constant communication overhead, a Hybrid Stateful Exclusive Complete Subtree (H-SECS) was designed and implemented. The number of subgroups was configured by H-SECS according to the security level of application scenario to make an optimal tradeoff between communication overhead and collusion resistance ability. Theoretical analysis and simulation results show that, compared with LKH protocol and SECS protocol, the communication overhead of H-SECS can be regulated in the ranges between  $O(1)$  and  $O(\log n)$ .

**Key words:** stateful; group rekeying; Logical Key Hierarchy (LKH) protocol; collusion resistance

## 0 引言

组密钥更新作为组播安全研究中的关键,用于解决成员动态变化(加入或离开)时,组控制器(Group Controller, GC)如何高效和安全地分发一个新的组密钥(Group Key, GK)给合法用户的问题,并保证密钥更新过程中的前向安全性和后向安全性<sup>[1]</sup>。在有状态的组密钥更新协议<sup>[2-4]</sup>中,大多数协议是基于逻辑密钥分层(Logical Key Hierarchy, LKH)协议发展而来的。最早的LKH协议是由Wong等<sup>[5]</sup>和Wallner等<sup>[6]</sup>分别提出。文献[7]中证明了LKH协议在抗完全合谋攻击时,通信开销的下界是  $O(\log n)$ ,  $n$  为组规模大小。所以,不折中安全性和其他性能(存储开销、密钥更新效率),使通信开销低于  $O(\log n)$  是不可能的。抗合谋攻击主要针对组密

钥管理中安全威胁提出的系统需求<sup>[8]</sup>。在一些高度敏感的数据中,抗完全合谋攻击是一种较强的安全性要求,但是,在一些资源受限(无线传感器网络<sup>[9-10]</sup>)或者商业应用(付费电视<sup>[11]</sup>、即付即看(Pay Per View, PPV)<sup>[12]</sup>)场景中,对安全性要求相对较弱,而对通信开销更为敏感,组成员频繁的加入(离开),会给GC带来巨大的通信开销,所以,为了节省通信开销,这些应用场景中的用户愿意容忍一定级别的安全性,通过牺牲抗合谋攻击能力来降低通信开销。

基于此,Fan等<sup>[11]</sup>提出了一种混合的组密钥更新协议(Hybrid Structuring Of Receivers, HySOR)。HySOR协议在LORE(Linear Ordering Of Receivers)协议的基础上结合LKH协议,实现了通信开销和抗合谋攻击之间的权衡。但是,LORE是基于双哈希链设计的协议,HySOR协议的计算复杂

收稿日期:2017-10-12;修回日期:2017-11-25;录用日期:2017-12-04。      基金项目:国家自然科学基金资助项目(61363084);云南大学第四批中青年骨干教师基金资助项目(XT412003);云南大学师资队伍建设基金资助项目(XT412001)。

**作者简介:**敖丽(1993—),女,云南曲靖人,硕士研究生,主要研究方向:信息安全、密码学; 刘璟(1972—),男,四川绵阳人,副教授,博士,主要研究方向:信息安全、计算机网络安全、密码学; 姚绍文(1966—),男,湖南永顺人,教授,博士生导师,博士,主要研究方向:信息安全、分布式计算; 武楠(1989—),女,河南南阳人,硕士研究生,主要研究方向:信息安全、密码学。

度随着组规模  $n$  呈线性变化。在文献[13]中,利用多密钥分发(Multicast Key Distribution, MKD)协议和 LKH 协议构成的混合方案中,MKD 仍然是基于双哈希链的协议。Liu 等<sup>[14]</sup>基于“子集-覆盖”框架提出了排外子集覆盖框架(Exclusive Complete Subtree, ECS),在此框架上,构造一个混合的无状态组密钥更新协议,该协议适合于资源受限或者商业应用场景。

本文在文献[14]的基础上,针对有状态的组播通信,设计并实现了一种混合的组密钥更新协议(Hybrid Stateful Exclusive Complete Subtree, H-SECS)。H-SECS 协议是基于 LKH 协议和有状态的完全排外子树(Stateful Exclusive Complete Subtree, SECS)协议设计的,可以在 LKH 协议和 SECS 协议之间进行调整,达到抗合谋攻击能力和通信开销之间的权衡。H-SECS 协议的一个极端是 LKH 协议,具有完全抗合谋攻击的特点;另外一个极端是 SECS 协议,具有常量通信开销但是只能抵抗一个用户攻击。本文对 H-SECS 协议的抗合谋攻击能力、通信开销进行了仿真实验分析,结果证明了 H-SECS 协议的通信开销和抗合谋攻击能力可以在 LKH 与 SECS 之间进行调控。

## 1 SECS 协议

Liu 等<sup>[14]</sup>基于排外密钥<sup>[15-16]</sup>设计了 SECS 协议。SECS 是一种有状态的组密钥更新协议,通信开销为  $O(1)$ ,但是只能抵抗单用户攻击。SECS 协议主要包括两个算法:个人密钥分配算法和组密钥更新算法。

### 1.1 个人密钥分配算法

个人密钥分配算法满足正确性和密钥的不可区分性,所以用 3 个伪随机生成器(Pseudo-Random Number Generator, PRNG)<sup>[17]</sup> ( $G_L, G_R, G_M$ ) 来生成密钥。根据组播中  $n$  个成员组成的集合  $U = \{u_1, u_2, \dots, u_n\}$ ,经过  $(G_L, G_R, G_M)$  自顶向下生成一棵平衡二叉树。二叉树的叶子节点与成员  $u$  直接相连。在平衡二叉树中,密钥的分配如下:任意中间节点  $i$ ,有一个种子节点  $S_i$ ,则  $S_{2i} = G_L(S_i)$  代表  $i$  节点的左孩子( $2i$ )的种子; $S_{2i+1} = G_R(S_i)$  代表  $i$  节点的右孩子( $2i+1$ )的种子; $K_i = G_M(S_i)$  代表  $i$  节点的排外密钥。将二叉树中从叶子节点回溯到根节点所经过路径上的排外密钥称为路径密钥。成员  $u$  的排外密钥集为  $u$  的路径密钥的集合;用  $I_u$  表示 GC 为  $u$  分配的个人密钥集,则  $I_u$  是所有路径密钥上的兄弟密钥。根据排外密钥的定义, $u$  不能计算出它的路径密钥,但是, $u$  可以推导出这棵密钥二叉树上其余的密钥。

### 1.2 组密钥更新算法

定义  $t$  时刻合法成员的集合为  $S^t$  ( $S^t \subset U$ ),撤销者集合  $R^t$  ( $R^t = U \setminus S^t$ )。 $S^{(t+1)}$  表示  $(t+1)$  时刻成员动态变化后的集合; $GK^t$  表示当前的组密钥; $GK^{(t+1)}$  表示更新过的组密钥;“ $\Rightarrow$ ”表示“组播”; $E_K(text)$  表示用密钥  $K$  加密  $text$ 。对于 SECS 协议,成员的加入(离开),组密钥都需要进行更新。

1) 成员加入。当成员  $u'$  在  $t$  时刻加入集合  $U$  时,有两种方案。第一种方案是 GC 通过安全渠道给合法用户广播密文块,即:GC  $\Rightarrow S^{(t+1)} : E_{GK^t}(GK^{(t+1)})$ ,只有知道  $GK^t$  的用户可以对密文块进行解密,得到  $GK^{(t+1)}$ 。本文为了保证信息的机密性,采用了哈希函数的消息认证码(Hash-based Message Authentication Code, HMAC)函数<sup>[18]</sup>,使得用于组通信的密钥和用于加密的密钥不是同一个密钥,本文后面提到的  $GK$  是

已经处理过,用于加密的  $GK$ 。第二种方案是通过一个公开的单向哈希函数(SHA-1)<sup>[19]</sup> 来更新  $GK^{(t+1)}$ ,合法成员在收到 GC 广播的更新消息时,直接通过哈希函数计算出新的组密钥  $GK^{(t+1)}$ 。

2) 成员离开。假设  $t$  时刻离开的成员集合是  $L = \{i_1, i_2, \dots, i_m\}$ ,GC 首先计算出离开成员的排外密钥  $\{K_{i_1}, K_{i_2}, \dots, K_{i_m}\}$ ,然后将这些排外密钥和  $GK^t$  异或后作为会话密钥来加密  $GK^{(t+1)}$ ,GC 将密文广播给合法成员,密文结构如下:

$$GC \Rightarrow S^{(t+1)} : \langle i_1, i_2, \dots, i_m, E_{K_{i_1} \oplus K_{i_2} \oplus \dots \oplus K_{i_m} \oplus GK^t}(GK^{(t+1)}) \rangle$$

用  $C$  表示密文块, $D_K(C)$  表示用  $K$  来解密密文块  $C$ 。当用户  $u \in S^{(t+1)}$  接收到更新消息  $\langle i_1, i_2, \dots, i_m, C \rangle$ ,根据排外密钥的定义, $u$  根据密钥集  $I_u$  中推导出排外密钥  $K_j$ ;所有的排外密钥进行异或运算; $K_{i_1} \oplus K_{i_2} \oplus \dots \oplus K_{i_m} \oplus GK^t$  计算出新的会话密钥。最后解密得到组密钥  $GK^{(t+1)}$ ,解密结构如下: $D_{K_{i_1} \oplus K_{i_2} \oplus \dots \oplus K_{i_m} \oplus GK^t}(C)$ 。

## 2 H-SECS 协议

结合 LKH 协议抗完全合谋攻击的特点和 SECS 协议具有常量通信开销的优势,设计并实现了一种混合的组密钥管理协议(H-SECS)。该协议的密钥管理是一个两层的树型结构,为了描述方便,第一层称为 LKH 协议树;第二层称为子组树(Division Tree, DT),DT 沿用 SECS 协议树的设计规则。H-SECS 协议可以分为成员初始化、成员加入和成员离开 3 个阶段。表 1 列出 H-SECS 协议中用到的部分符号。

表 1 H-SECS 协议中符号说明

Tab. 1 Symbol definition of H-SECS protocol

符号	说明	符号	说明
$D_i$	第 $i$ 子组	$I_u$	$u$ 在 $DT_i$ 中的个人密钥集
$m$	子组成员数目	$KEK$	密钥加密密钥
$u_j^{D_i}$	$D_i$ 子组里的第 $j$ 成员	$E_K(text)$	用 $K$ 加密 $text$
$DT_i$	$D_i$ 子组的 SECS 树	$D_K(C)$	用 $K$ 来解密密文 $C$

### 2.1 成员初始化

假设  $n$  个成员建立组通信。首先,GC 建立一棵 LKH 协议树,包含  $d$  个叶子节点,树中每个节点对应一个密钥,树的叶子节点与子组一一对应。然后,在每个子组里建立一棵 SECS 协议树,简称 DT。DT 中包含  $m$  ( $m = n/d$ ) 个叶子节点,其中,如果  $n$  不能整除  $d$ ,则前几个子组里 DT 的叶子节点为  $m$ ,最后一个子组叶子节点不足  $m$ 。DT 中每个叶子节点与组成员一一对应。在建树的过程中,采用哈夫曼编码为树的每个节点进行编号,规定从根节点开始,左分支为 0,右分支为 1,根节点的编号为 0。对于 DT 中的每个成员,在叶子节点编号的前面加上子组号  $D_i$  (Division),其中  $D_i$  的数值范围是  $0 \sim 2^{32} - 1$ 。

由于 H-SECS 协议的密钥管理是一个两层的树型结构,所以在 DT 中的每个成员存储两组密钥:一组是所在子组对应的 LKH 协议树上的辅助密钥;另外一组是成员对应的 DT 中的私人密钥。辅助密钥是子组  $D_i$  对应的 LKH 协议树上叶子节点到根节点路径上所有节点对应的密钥;私人密钥是成员  $u$  所对应的叶子节点到  $DT_i$  树上根节点路径上所有节点对应的兄弟节点的密钥,其中,LKH 协议树中根节点的密钥为  $GK$ ,组通信中所有合法成员共享  $GK$ 。如图 1 所示,H-SECS 结

构图中共有15个成员。分为5个子组: $D_0, D_1, D_2, D_3, D_4$ ; 每个子组中包含一棵DT, 分别为 $DT_0, DT_1, DT_2, DT_3, DT_4$ ; DT的叶子节点对应3个成员 $u_1, u_2, u_3$ 。组密钥 $CK$ 是 $K_1, u_2^{D_1}$ 表示 $D_1$ 子组中的 $u_2$ 用户, 它的ID号是11001, 第一位的1代表子组号, 后面是成员对应的叶子节点的编号。 $u_2^{D_1}$ 用户有两组密钥: 一组是LKH协议树上的辅助密钥 $\{K_1, K_2, K_4, K_9\}$ , 辅助密钥被该 $D_1$ 子组里成员共享; 一组是 $u_2^{D_1}$ 对应的私人密钥 $\{SK_3, SK_4\}$ 。

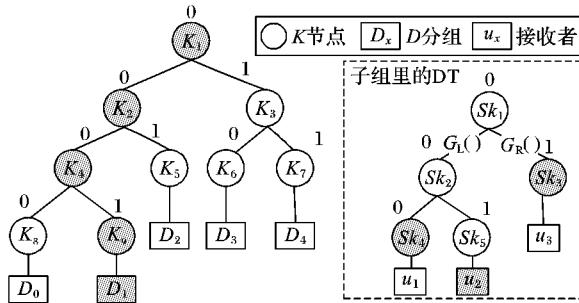


图1 H-SECS结构示例图  
Fig. 1 Schematic of H-SECS protocol

## 2.2 成员加入

加入组播的成员有两种: 一种是新加入组播中的成员; 另一种是由网络故障或者其他原因被撤销的成员, 重新请求加入组播通信。新成员请求加入组通信时, 步骤如下:

1) GC为加入成员在整个密钥树中分配对应位置。

重新请求加入组播通信的成员, GC为这个成员分配到以前的节点位置, 并且将节点对应的身份、私有密钥、辅助密钥分配给该成员。如果子组有成员加入, 那么这个子组变为活跃的子组; 否则, 称为不活跃的子组。对于新加入组播的成员, 为了使通信开销小并且提高抗合谋攻击能力, GC随机选择LKH协议树中叶子节点对应的子组 $D_i$ 时, 有以下两种情况:

①如果选择的 $D_i$ 是对的LKH协议树中最小深度的叶子节点, 则将这个叶子节点分裂成两个节点并且新创建一个子组, 子组中包含一棵 $DT_i$ 树。其中, 左孩子节点与 $D_i$ 相连, 右孩子节点与新创建的 $DT_i$ 树相连。将新成员插入到这个新的子树 $DT_i$ 中, 并且将插入成员所在的子组标记为活跃的子组。

②如果选择的 $D_i$ 对应的不是LKH协议树中最小深度的节点, 为了保证子组中对应的 $DT_i$ 树平衡, GC从所选子组中对应 $DT_i$ 树中深度最小的叶子节点中随机选择一个叶子节点, 将这个叶子节点进行分裂, 左边的节点与原来的成员相关联, 右边的节点对应的是新插入成员的位置。如果选择到的 $D_i$ 为不活跃的子组, 由于新成员的加入, 子组 $D_i$ 变成活跃子组。图2是在图1中加入一个成员, GC选择到 $D_3$ 子组后H-SECS结构图对应的LKH协议树变化。此时,  $D_3$ 子组被标记为活跃的子组,  $D_3$ 子组下面对应 $DT_3$ 树,  $DT_3$ 树只有一个节点, 这个节点与新插入的成员对应。

2) GC生成 $CK^{t+1}$ 并且广播密钥更新消息。

假设 $t$ 时刻要求加入组播通信的节点集合用 $J$ 表示,  $S^{t+1} (S^{t+1} = S^t \cup J)$ 表示加入成员后合法成员的集合。为了保证后向安全性, 需要更新加入成员所影响的节点密钥并且分发私人密钥给加入的新成员。

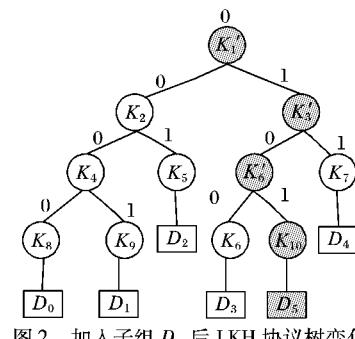


图2 加入子组 $D_5$ 后LKH协议树变化  
Fig. 2 LKH tree view of joining subgroup  $D_5$

组密钥更新步骤如下:

①为活跃子组生成密文块。为了保证信息的机密性, GC需要为活跃子组中的成员分发两种密文块: 逻辑密文块和SECS对应的密文块。

发送逻辑密文块 GC首先统计每个活跃子组对应的辅助密钥, 然后使用活跃子组对应的LKH协议树中儿子节点加密父节点的方式生成密钥加密密钥(Key Encryption Key, KEK); 最后, 将加密节点的身份ID、被加密节点的身份ID和对应的KEK封装成逻辑密文块(Logical Key Hierarchy Cipher Text Block, LKHCB)发送给组播中合法成员。

发送SECS协议对应的密文块 GC使用以前的组密钥 $CK$ 来加密活跃子组对应的LKH协议树上的叶子节点密钥, 即:  $E_{CK}(K_i)$ , 然后将 $E_{CK}(K_i)$ 发给合法成员。在图2中, 活跃的子组为 $D_5$ , 对应LKH协议树上所影响节点的密钥是 $\{K_1, K_3, K_6\}$ , 本文用 $\{K_1', K_3', K_6'\}$ 表示已经更新过的密钥。则密钥的更新产生密文块LKHCB为:

$$LKHCB_1 = \{0100, 010, E_{K_6'}(K_6')\}$$

$$LKHCB_2 = \{0101, 010, E_{K_{10}}(K_6')\}$$

$$LKHCB_3 = \{010, 01, E_{K_6'}(K_3')\}$$

$$LKHCB_4 = \{011, 01, E_{K_7}(K_3')\}$$

$$LKHCB_5 = \{00, 0, E_{K_2}(K_1')\}$$

$$LKHCB_6 = \{01, 0, E_{K_3'}(K_1')\}$$

$K_1'$ 为更新过后的组密钥 $CK^{(t+1)}$ 。GC将逻辑密文块信息和SECS对应的密文信息封装成密文(Cipher Text, CT), 发送给活跃子组中合法成员。密文结构如下:

$$GC \Rightarrow S^{(t+1)}: CT = \{LKHCB_1, LKHCB_2, LKHCB_3, LKHCB_4, LKHCB_5, LKHCB_6, E_{CK}(K_{10})\}$$

②为不活跃子组生成密文块。对于不活跃子组中的成员, GC发送的密文遵循SECS协议成员加入的密文更新规则。GC使用以前的组密钥 $CK$ 来加密更新过的组密钥, 即:  $E_{CK}(CK^{(t+1)})$ , 然后将 $E_{CK}(CK^{(t+1)})$ 发给不活跃子组中成员。

3) 成员解密文更新消息。

组播里合法成员到密文CT。对于非活跃组里的成员, 利用其知道的 $CK$ , 直接解密出 $CK^{(t+1)}$ 。对于活跃子组的成员, 根据其所知的 $CK$ , 解密出对应的LKH协议树叶子节点更新过的密钥, 依次根据接收到的密文块解密出 $CK^{(t+1)}$ 。假设在图2中, 成员 $u_2^{D_5}$ 解密, 则解密信息如下: 根据 $CK$ 密钥 $K_{10}$ ; 根据 $K_{10}$ , 解密密文块 $LKHCB_2$ , 获取 $K_6'$ ; 根据 $K_6'$ , 解密 $LKHCB_3$ , 获取 $K_3'$ ; 根据 $K_3'$ 解密 $LKHCB_6$ , 获取到 $CK^{(t+1)}$ , 即 $K_1'$ 。

### 2.3 成员离开

假设  $t$  时刻离开的成员用集合  $L = \{i_1, i_2, \dots, i_m\}$  表示。为了保证前向安全性,需要更新离开成员所影响的 H-SECS 结构树中的密钥。如果成员  $u_j^{D_i}$  离开时,则更新处理步骤如下:

1) 根据退出成员的 ID 定位到退出成员所在子组里的位置,删除  $DT_i$  树中代表该成员的叶子节点,并且将这个节点标记为不可用的节点,由于成员的离开,这个子组变成不活跃的子组。如果  $DT_i$  树中的所有叶子节点都为不可用的节点,则删除 LKH 协议树中代表  $DT_i$  组的叶子节点。GC 对 LKH 协议树进行调整,使 LKH 协议树保持最优状态。图 3 所示是图 1 中 LKH 协议树删除组  $D_1$  后调整后的图。

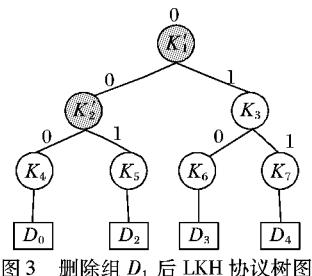


图 3 删去子组  $D_1$  后 LKH 协议树图

Fig. 3 LKH protocol tree after deleting subgroup  $D_1$

2) 成员退出密钥更新可以分为两个步骤:

更新子组对应 LKH 协议树上影响节点的密钥,过程遵循 LKH 协议的密钥更新过程。更新退出成员所在  $DT_i$  树上密钥,这个过程遵循 SECS 密钥更新规则,步骤如下:

① 建立 Steiner 树<sup>[13]</sup>。对  $(U \setminus L)$  建立 Steiner 树,用  $ST(U \setminus L)$  代表 Steiner 树。假设图 4 中是某个子组中离开成员集为  $L = \{u_1, u_2, u_3\}$ ,  $ST(U \setminus L)$  树是由虚线相连的节点组成;

② 查找覆盖子树。对比活跃的  $DT_i$  树和  $ST(U \setminus L)$  树,覆盖子树为当前组的  $DT_i$  树除去当前组的  $ST(U \setminus L)$ ,即  $DT_i - ST(U \setminus L)$ 。图 4 中覆盖子树由实线相连的节点组成。

③ 更新与分发密钥树中相关节点的密钥。覆盖子树所对应的排外密钥与组密钥异或后形成会话密钥,用会话密钥来加密  $DT_i$  树中对应 LKH 协议树的叶子节点  $K_\omega$ 。将排外密钥的 ID, 加密形成的密文封装成密文块 (Division Cipher Text Block, DCB), 其结构如下:

$$DCB = \langle i_1, i_2, \dots, i_m, E_{Sk_1 \oplus Sk_2 \oplus \dots \oplus Sk_m \oplus GK^t}(K_\omega) \rangle$$

将活跃子组形成的密文块 DCB 和对应的子组号  $D_i$  以及形成的逻辑密文块封装成密文 (Division Cipher Text, DCT)。GC 将 DCT 发给组播通信中合法成员。

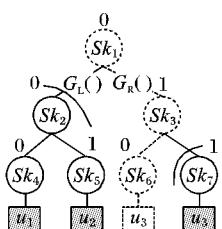


图 4 子组里的完全排外子树图

Fig. 4 Complete exclusion subtree in the subgroup

3) 组成员解密密文更新消息。

组播里合法成员接收到密文 DCT。对于活跃子组里的成

员,组成员根据 LKH 协议树的密钥检索出密文块,获取相关的密钥,最终解密出  $GK^{(t+1)}$ 。而对于不活跃子组里的合法成员,解密  $GK^{(t+1)}$  有以下两个步骤:首先对 DCB 解密,根据排外密钥的定义,对于每个  $i_j (j = 1, 2, \dots, m)$ ,  $u$  可以从个人密钥集  $I_u$  中推导出  $Sk_{ij}$ , 根据这些  $Sk_{ij}$  计算出 DCB 的会话密钥,即  $Sk_1 \oplus Sk_2 \oplus \dots \oplus Sk_m \oplus GK^t$ ; 然后解密出对应 LKH 协议树上新更新过的叶子节点的密钥  $K_\omega$ , 解密结构为:  $D_{Sk_1 \oplus Sk_2 \oplus \dots \oplus Sk_m \oplus GK^t}(C)$ ; 最后对逻辑密文块解密,根据 LKH 协议树的解密规则来依次解密。最终,获取到组密钥  $GK^{(t+1)}$ 。

### 2.4 抗合谋攻击能力分析

对于 SECS 协议,两个恶意的撤销用户可能就会破坏组的前向安全性,而在 H-SECS 协议中,  $k$  个合谋者成功攻破组密钥的条件是当且仅当  $k$  个合谋者中至少有两个合谋者被分在同一个子组中。用  $P_k$  表示  $k$  个合谋者中至少有两个合谋者被分配在同一个子组中的概率,根据文献[11]有式(1):

$$P_k = \begin{cases} 1, & k > d \\ 1 - \frac{d!(n-k)!(n/d)^k}{n!(d-k)!}, & k \leq d \end{cases} \quad (1)$$

其中:  $n$  是组规模,  $d$  是子组数目。在给定  $n$  和  $k$  时,  $P_k$  随着  $d$  的增大单调递减, 即:  $d$  越大, 至少有两个合谋者被分配在同一个子组中的概率  $P_k$  就越小。在组规模一定时,  $d$  越大, 安全性越好, 但是  $d$  的增大, 会导致带宽的增大。本文提出的 H-SECS 方案中, 用户可以根据不同的安全性要求, 调整  $d$  的大小, 在合谋攻击和通信开销之间进行一个最优化的权衡: 当  $d = 1$  时, H-SECS 协议变成 SECS 协议, 此时每个成员加入(离开)的通信开销是  $O(1)$ , 但是只能抵抗一个用户攻击; 当  $d = n$  时, H-SECS 协议变成 LKH 协议, 此时每个成员加入(离开)的通信开销是  $O(\log n)$ 。

### 3 性能仿真实验

在 Microsoft Visual Studio 2013 环境下,利用 OpenSSL(版本号:1.0.1t)加密库中的高级加密标准 128 位 (Advanced Encryption Standard-128 bit, AES-128) 和安全散列算法 256 位 (Secure Hash Algorithm-256 bit, SHA-256) 来对 H-SECS 方案进行仿真。在 H-SECS 协议中,成员的加入(离开)时,组密钥更新是影响通信开销的主要因素;而对于抗合谋攻击能力则是由子组数目决定;首先,对子组数目和  $P_k$  之间变化进行仿真;然后,在加入(离开)成员不同时,对 LKH、SECS、H-SECS 三种协议的通信开销进行仿真实验。

在仿真实验中,参数的配置  $k = 5; n = 32768; P_5$  代表 5 个合谋者至少有两个合谋者被分在同一个子组中的概率。 $P_5 = 0$  代表完全抗合谋攻击;  $P_5 = 1$  代表只能抵抗单用户攻击。图 5 是子组数目  $d$  与  $P_5$  之间的关系图,可以看出当子组  $d$  逐渐增大时, H-SEC 协议的抗合谋攻击能力逐渐增强。当  $d = n$  时, H-SECS 协议是完全抗合谋攻击的,此时, H-SECS 协议变成 LKH 协议。当  $d = 1$  时, H-SECS 协议变成 SECS 协议,安全性最弱。

测试加入(离开)成员不同时,LKH、SECS 和 H-SECS 三种方案的通信开销。为了更好地体现成员变化的随机性,对于加入(离开)成员数目不同时,本文均进行 20 次计算,然后取算数平均值作为最终的测试结果。测试中的  $P_k, d, k, n$  如 2.4 节所述。测试参数配置如表 2 所示:其中  $P_5 = 0.1476$

( $d = 64, m = 512$ ) 和  $P_5 = 0.0094$  ( $d = 1024, m = 32$ ) 代表 H-SECS 协议的两个中间点。成员加入测试的结果如图 6 所示, 成员离开测试的结果如图 7 所示。

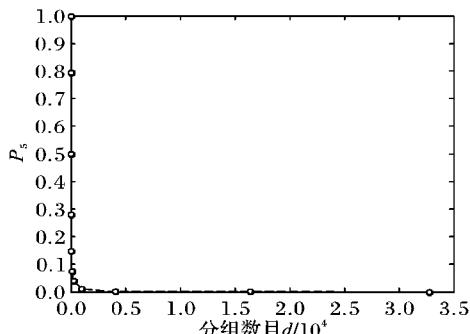


图 5 H-SECS 协议中  $P_5$  与子组的关系

Fig. 5 Relationship between  $P_5$  and subgroups in H-SECS protocol

表 2 测试参数配置说明

Tab. 2 Test parameter configuration instruction

对象	$d$	$P_5$	对象	$d$	$P_5$
SECS	1	1	H-SECS2	1 024	0.0094
H-SECS1	64	0.1476	LKH	32 768	0

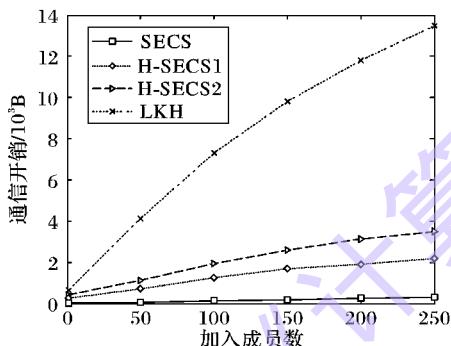


图 6 三种协议中成员加入通信开销的对比

Fig. 6 Communication overhead comparision of three protocols when users join

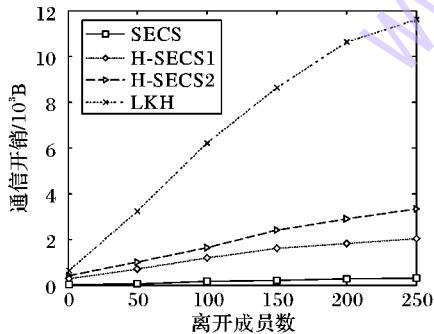


图 7 三种协议中成员离开通信开销的对比

Fig. 7 Communication overhead comparision of three protocols when users depart

由图可以看出,无论是成员加入还是成员离开,LKH 协议的通信开销最大,而 SECS 协议的通信开销最小。对于 H-SEC 协议的通信开销可以随着子组数目  $d$  在 SECS 协议和 LKH 协议之间进行调控,即:可以在  $O(1)$  和  $O(\log n)$  之间进行调控。

仿真实验中,子组数目  $d$  越大,H-SEC 协议抗合谋攻击的能力就越强,通信开销就越接近 LKH 方案的通信开销;而子组数目  $d$  越小,H-SEC 协议抗合谋攻击能力就越弱,通信开销

就越接近 SECS 方案的通信开销。所以,根据不同应用场景的安全级别,H-SEC 协议可以通过设置子组的数目,将抗合谋攻击能力在 SECS 协议和 LKH 协议之间进行调整,最终达到抗合谋攻击能力和通信开销之间的权衡。

#### 4 结语

本文基于 SECS 协议和 LKH 协议提出并实现了一种有状态的组密钥更新协议(H-SECS)。从理论分析和仿真结果中可以看出该协议的通信开销可以在 LKH 协议和 SECS 协议之间进行调整。因为在频繁组密钥更新的场景中,LKH 协议所产生的通信开销也是不容小觑的,所以在权衡安全性的条件下,H-SECS 协议可以在抗合谋攻击能力和通信开销上作出最优的权衡,实现通信开销在 SECS 协议和 LKH 协议通信开销之间的调整。

#### 参考文献 (References)

- [1] WU Y, LIU J, HOU J, et al. A stateful multicast key distribution protocol based on identity-based encryption [C]// Proceedings of the 2017 IEEE/ACIS 16th International Conference on Computer and Information Science. Piscataway, NJ: IEEE, 2017: 19–24.
- [2] SHERMAN A T, MCGREW D A. Key establishment in large dynamic groups using one-way function trees [J]. IEEE Transactions on Software Engineering, 2003, 29(5): 444–458.
- [3] LIU J, YANG B. Collusion-resistant multicast key distribution based on homomorphic one-way function trees [J]. IEEE Transactions on Information Forensics and Security, 2011, 6(3): 980–991.
- [4] XU J, LI L, LU S, et al. A novel batch-based LKH tree balanced algorithm for group key management [J]. Science China Information Sciences, 2017, 60(10): 108–301.
- [5] WONG C K, GOUDA M, LAM S S. Secure group communications using key graphs [J]. IEEE/ACM Transactions on Networking, 2000, 8(1): 16–30.
- [6] WALLNER D, HARDER E, AGEE R. Key management for multicast: issues and architectures [EB/OL]. [2017-06-20]. <http://www.rfc-editor.org/in-notes/pdfs/rfc2627.txt.pdf>.
- [7] MICCIANICIO D, PANJWANI S. Optimal communication complexity of generic multicast key distribution [J]. IEEE/ACM Transactions on Networking, 2004, 16(4): 803–813.
- [8] DU W, HE M, X. Self-healing key distribution with revocation and resistance to the collusion attack in wireless sensor networks [C]// Proceedings of the 2nd International Conference on Provable Security. Berlin: Springer-Verlag, 2008: 345–359.
- [9] PERRING A, STANKOVIĆ J C, WAGNER D. Security in wireless sensor networks [J]. Communications of the ACM, 2004, 47(6): 53–57.
- [10] 王洁, 卢建朱, 曾小飞. 可及时确定受攻击节点的无线传感器网络数据聚合方案 [J]. 计算机应用, 2016, 36(9): 2432–2437. (WANG J, LU J Z, ZENG X F. Data aggregation scheme for wireless sensor network to timely determine compromised nodes [J]. Journal of Computer Applications, 2016, 36(9): 2432–2437.)
- [11] FAN J, JUDGE P, AMMAR H M. HySOR: group key management with collusion-scalability tradeoffs using a hybrid structuring of receivers [C]// Proceedings of the 11th International Conference on Computer Communications and Networks. Washington, DC: IEEE Computer Society, 2002: 196–201.

(下转第 1382 页)

方案,通过利用模糊系统的特性,主要解决了不同用户的外包数据的共享性问题,提高了用户外包数据的实用价值。下一步的工作是探索能够适用于本方案的具体实例环境,构造出真正意义能够实际应用的全同态加密方案。

#### 参考文献 (References)

- [1] MEZGHANI K, AYADI F. Factors explaining IS managers attitudes toward cloud computing adoption [J]. International Journal of Technology and Human Interaction, 2016, 12(1): 1–20.
- [2] GENTRY C. Fully homomorphic encryption using ideal lattices [C]// Proceedings of the 41st Annual ACM Symposium on Theory of Computing. New York: ACM, 2009: 169–178.
- [3] DUCAS L, MICCIANCIO D. FHEW: Bootstrapping homomorphic encryption in less than a second [C]// EUROCRYPT 2015: Proceedings of the 2015 Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2015: 617–640.
- [4] BRAKERSKI Z, PERLMAN R. Lattice-based fully dynamic multi-key FHE with short ciphertext [C]// CRYPTO 2016: Proceedings of the 36th Annual International Cryptology Conference. Berlin: Springer, 2016: 190–213.
- [5] NUIDA K, KUROSAWA K. Fully homomorphic encryption over integers for non-binary message spaces [C]// EUROCRYPT 2015: Proceedings of the 2015 Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2015: 537–555.
- [6] GENTRY C, HALEVI S. Implementing gentry's fully homomorphic encryption scheme [C]// Proceedings of the 30th Annual International Conference on Theory and Applications of Cryptographic Techniques: Advances in Cryptology. Berlin: Springer, 2011: 129–148.
- [7] BRAKERSKI Z, VAIKUNTANATHAN V. Efficient fully homomorphic encryption from (Standard) LWE [C]// Proceedings of the 2011 IEEE 52nd Annual Symposium on Foundations of Computer
- [8] GENTRY C, SALAHAI A, WATERS B. Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based [C]// Proceedings of the 33rd Annual Cryptology Conference. Berlin: Springer, 2013: 75–92.
- [9] SAHAI A, WATERS B. Fuzzy identity-based encryption [C]// Proceedings of the 24th Annual International Conference on Theory and Applications of Cryptographic Techniques. Berlin: Springer-Verlag, 2005: 457–473.
- [10] REGEV O. On lattice, learning with errors, random linear codes, and cryptography [C]// Proceedings of the 37th Annual ACM Symposium on Theory of Computing. New York: ACM, 2005: 84–93.
- [11] BRAKERSKI Z. Fully homomorphic encryption without modulus switching from Classical GapSVP [C]// Proceedings of the 32nd Annual Cryptology Conference on Advances in Cryptology. New York: ACM, 2012: 868–886.
- [12] LAI J, DENG R H, GUAN C, et al. Attribute-based encryption encryption with verifiable outsourced decryption [J]. IEEE Transactions on Information Forensics & Security, 2013, 8(8): 1343–1354.
- [13] TROMER E, VAIKUNTANATHAN V. On-the-fly multiply computation on the cloud via multikey fully homomorphic encryption [C]// Proceedings of the 44th ACM Symposium on Theory of Computing. New York: ACM, 2012: 1219–1234.

This work is partially supported by the National Cryptography Development Fund of China (MMJJ20170112), the Natural Science Foundation of Shaanxi Province (2016JQ6037).

**BAI Ping**, born in 1990, M. S. candidate. His research interest include cryptology.

**ZHANG Wei**, born in 1976, Ph. D., professor. Her research interests include cryptology, information security.

(上接第 1376 页)

- [12] 姬东耀,王育民. 公平有效的 Web 视频服务即付即看协议设计与分析[J]. 西安电子科技大学学报(自然科学版),2001,8(4):425–429. (JI D Y, WANG Y M. Analysis and design of fair and efficient pay-per-view protocols for Web-based video service [J]. Journal of Xidian University (Natural Science Edition), 2001, 8(4): 425–429.)
- [13] LIU J, HUANG Q, YANG B, et al. Efficient multicast key distribution using HOWP-based dynamic group access structures [J]. IEEE Transactions on Computers, 2013, 62(8): 1656–1672.
- [14] LIU J, LIU M, WANG C J, et al. Group rekeying in the exclusive subset-cover framework [J]. Theoretical Computer Science, 2017, 678: 63–77.
- [15] FIAT A, NAOR M. Broadcast encryption [C]// Proceedings of the 13th Annual International Conference on Advances in Cryptology. New York: Springer-Verlag, 1993: 480–491.
- [16] KIM H, HONG M S, YOON H, et al. Secure group communication with multiplicative one-way functions [C]// Proceedings of the 2005 International Conference on Information Technology: Coding and Computing. Washington, DC: IEEE Computer Society, 2005: 685–690.
- [17] CANETTI R, GARAY J, ITKIS G, et al. Multicast security: taxonomy and some efficient constructions [C]// Proceedings of the 1999 Conference on Computer Communications. Piscataway,

- NJ: IEEE, 1999: 708–716.
- [18] KRAWCZYK H, BELLARE M, CANETTI R. HMAC: keyed-hashing for message authentication [EB/OL]. [2017-06-20]. <http://ikamr.asp24.no/kdrs/pdf-copies/org.python.library.000000825.pdf>.
- [19] WANG X, YIN Y L, YU H. Finding collisions in the full SHA-1 [C]// Proceedings of the 25th Annual International Conference on Advances in Cryptology. Berlin: Springer-Verlag, 2005: 17–36.

This work is partially supported by the National Natural Science Foundation of China (61363084), the Foundation for the 4th Batch of the Middle-aged and Youth Key Teachers of Yunnan University (XT412003), the Faculty Team Construction Foundation of Yunnan University (XT412001).

**AO Li**, born in 1993, M. S. candidate. Her research interests include information security, cryptography.

**LIU Jing**, born in 1972, Ph. D., associate professor. His research interests include information security, computer network security, cryptography.

**YAO Shaowen**, born in 1966, Ph. D., professor. His research interests include information security, distributed computing.

**WU Nan**, born in 1989, M. S. candidate. Her research interests include information security, cryptography.