一种基于双线性对的群密钥管理方案

周 健[1,2] 周贤伟[1] 孙丽艳[2]
(北京科技大学信息工程学院 北京 100081)[1] (安徽财经大学管理科学与工程学院 蚌埠 233041)[2]

摘 要 现有群密钥管理方案大都基于GDH(Group key Management Based on Diffie-Hellman)密钥交互协议,该协议限制了子树规模。针对这一问题,提出一种基于双线性对的群密钥管理方案(BPGKM,Group Key Management based on Bilinear Pairing),该方案在保证安全的前提下可扩大密钥树中子树的规模,以支持更大规模的网络,同时可使计算操作减少一半,以提高群密钥管理操作的效率。

关键词 群密钥管理,双线性对,共享密钥,安全,效率
中图法分类号 TP309.2 **文献标识码** A

Group Key Management Scheme Based on Bilinear Pairing

ZHOU Jian[1,2] ZHOU Xian-wei[1] SUN Li-yan[2]
(Department of Communication Engineering, University of Science and Technology Beijing, Beijing 100081, China)[1]
(Department of Management Science and Engineering, Anhui University of Finance and Economics, Bengbu 233041, China)[2]

Abstract Current schemes on group key management are almost based on GDH (Group key Management Based on Diffie-Hellman) protocol, and their sub tree in key tree is limit. To solve the question, this paper put forwards a new scheme based on bilinear pairing (BPGKM), which increases the children of node in key tree from two to three. Therefore, this scheme supports larger networks on the premise of protecting the network security, whose efficiency is better than current schemes with reducing half of computations on key.

Keywords Group key management, Bilinear pairing, Shared key, Security, Efficiency

1 引言

随着Ad-hoc[1,2]广泛应用于各行各业,针对Ad-hoc的群通信安全成为研究热点,网络节点不仅需要快速协商共享密钥,建立安全信道,同时需要降低网络通信负载和计算复杂度。现有的群密钥管理方案大都基于Diffie-Hellman[3]方法。如文献[4]由节点轮流给出共享密钥资料,建立共享密钥,然而该方案面对网路拓扑结构变化时,需要所有节点重新计算共享密钥。文献[5,6]对此进行了改进,提出建立密钥树协商共享密钥方案STR(Short for Skinny Tree),该方案能够有效减少由于网络拓扑变化带来的通讯负载和计算,因此成为当前较好的群密钥管理方案。然而该方案中的密钥树中,每个节点的孩子数最多只能为两个,因此该方案在面对较大的网络时,具有较大深度的密钥树,降低了群密钥管理的效率。针对这一问题,本文提出使用双线性对的群密钥管理方案,即扩展子密钥树规模,减少密钥树的深度,从而提高群密钥管理效率。

2 双线性映射

双线性映射[7]是基于身份的密码体制中非常重要的概念。双线型映射定义为:设q为一个大素数,点p为q阶加法循环群G_1的生成元,G_2为同阶的乘法循环群,$e:G_1\times G_1\rightarrow G_2$称为双线性映射(bilinear map),也称双线性配对或双线性对(bilinear pairing)。同时双线性映射具有如下一些难解问题:

• BDHP(bilinear Diffie-Hellman problem),如果$e:G_1\times G_1\rightarrow G_2$是一个标准的线性映射,$P$是$G_1$的生成元,设$a,b,c\in Z_q^*$,给定$P,aP,bP,cP$,计算$e(p,p)^{abc}\in G_2$,则多项式时间内解决该问题是不可能的。

• CDHP(computational Diffie-Hellman),如果$e:G_1\times G_1\rightarrow G_2$是一个标准的线性映射,$P$是$G_1$的生成元,设$a,b\in Z_q^*$,给定$P,aP,bP$,计算$abP\in G_1$,则多项式时间内解决该问题是不可能的。

3 群密钥管理(BPGKM)

BPGKM基于三方密钥交换协议[8],通过密钥树维护群密钥管理。拓扑变化时,执行加入、离开、合并和分离操作。

3.1 BPGKM密钥树

图1所示为密钥树,包含两类节点:叶子节点和内部节点。叶子节点对应群成员,内部节点对应共享密钥资料组成部分。每个内部节点具有两个叶子节点和一个内部节点,除

到稿日期:2010-09-14 返修日期:2010-12-13 本文受国家自然科学基金资助项目(60773074),国家863计划项目(2007AA01Z213,2009AA01Z209),安徽省高等学校自然科学研究基金项目(KJ2010B005)和安徽省高校优秀青年人才基金项目(2009SQRZ084)资助。
周 健(1979—),男,博士生,讲师,CCF会员,主要研究方向为网络安全、密钥协议;孙丽艳(1976—),女,硕士,讲师,主要研究方向为Ad Hoc网络,E-mail:yewafeng12@sina.com.cn。

第二层的内部节点具有 3 个叶子节点外，叶子节点 i 随机选择 x_i，则内部节点的 $k_i=(x_{i-2}P,x_{i-2}P)^{k_{i-1}}$，且 $k_1=(x_1P,x_2P)^{k_3}$。

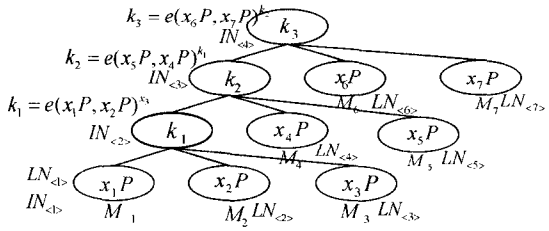
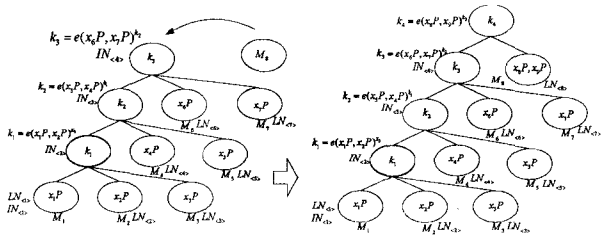


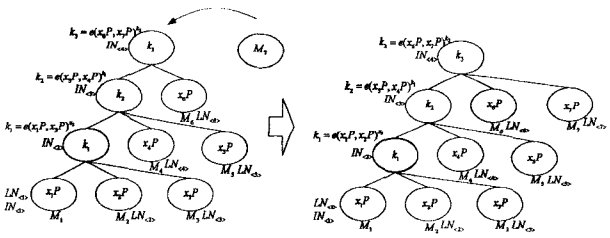
图 1　BKGKM 密钥树

3.2　加入

节点加入分为两种情况，如图 2 所示。



(a)加入的层为两个节点



(b)加入的层为三个节点

图 2　节点加入操作

(1)加入节点所在层为 2 个节点，协议步骤如下：

Step1　新加入节点 M_{2n+2} 广播加入请求消息：$M_{2n+2}\xrightarrow{x_{2n+2}P}C=\{M_1,M_2,\cdots,M_{2n+1}\}$；

Step2　每个成员更新密钥树，将新加入节点添加到密钥树中，作为一个新叶子节点；

Step3　每个成员将 k_n 移出；

Step4　节点 $LN_{\langle 2n+1\rangle}$ 重新生成共享密钥组成部分 $x'_{2n+1}P$，重新计算 $k_n=e(x'_{2n+1}P,x_{2n+2}P)^{k_{n-1}}$；

Step5　节点 $LN_{\langle 2n+1\rangle}$ 给予 M_{2n+2} 密钥资料 $x'_{2n+1}P$ 和 $k_{n-1}P$，节点 M_{2n+2} 计算 $k_n=e(x'_{2n+1}P,k_{n-1}P)^{x_{2n+2}}$；

Step6　节点 $LN_{\langle 2n+1\rangle}$ 广播更新后的密钥树 BPT，完成节点加入操作 $M_{2n+1}\xrightarrow{BPT_{\langle n\rangle}}C\cup M_{2n+2}=\{M_1,M_2,\cdots,M_{2n+2}\}$。

(2)加入节点所在层为 3 个节点，协议步骤如下：

Step1　新加入节点 M_{2n+3} 广播加入请求消息：$M_{2n+3}\xrightarrow{x_{2n+3}P,x_{2n+4}P}C=\{M_1,M_2,\cdots,M_{2n+2}\}$；

Step2　每个成员更新密钥树，将新加入节点添加到密钥树中，作为一个新叶子节点 $LN_{\langle 2n+3\rangle}$，同时建立一个新的内部节点 $IN_{\langle n+1\rangle}$；

Step3　每个成员将 k_n 移出；

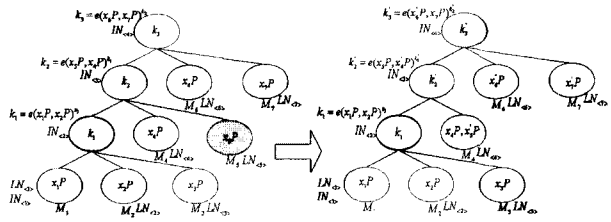
Step4　节点 $LN_{\langle 2n+1\rangle}$ 和 $LN_{\langle 2n+2\rangle}$ 重新生成共享密钥组成部分 $x'_{2n+1}P$ 和 $x'_{2n+2}P$，重新计算 $k_n=e(x'_{2n+1}P,x'_{2n+2}P)^{k_{n-1}}$；

Step5　节点 $LN_{\langle 2n+1\rangle}$ 或 $LN_{\langle 2n+2\rangle}$ 给予 M_{2n+2} 密钥资料 k'_nP，节点 M_{2n+2} 计算 $k_n=e(x_{2n+3}P,k'_nP)^{x_{2n+4}}$；

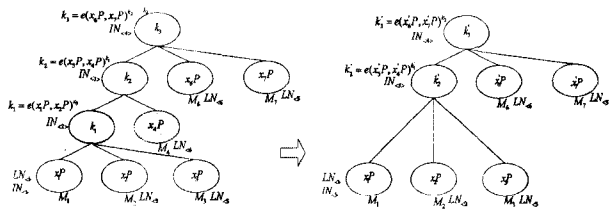
Step6　节点 $LN_{\langle 2n+1\rangle}$ 或 $LN_{\langle 2n+2\rangle}$ 广播更新后的密钥树 BPT，完成节点加入操作 $M_{2n+1}/M_{2n+2}\xrightarrow{BPT_{\langle n\rangle}}C\cup M_{2n+3}=\{M_1,M_2,\cdots,M_{2n+3}\}$。

3.3　离开

当某个节点离开网络后，也分为两种情况，如图 3 所示。



(a)离开节点具有两个兄弟节点



(b)离开节点具有一个兄弟节点

图 3　离开操作

(1)离开节点所在层为 2 个节点，协议步骤如下：

Step1　成员 M_{2i+2} 向 BPT 所有成员广播离开消息；

Step2　成员接消息后，移除层 $IN_{\langle i\rangle}$ 到 $IN_{\langle n\rangle}$ 的共享密钥；

Step3　从层 $IN_{\langle i\rangle}$ 到层 $IN_{\langle n\rangle}$ 的孩子节点重新选择共享密钥资料 $\{x'_{2i+1}P,x'_{2i+2}P,x'_{2i+3}P,\cdots,x'_{2n+2}P\}$，其中 $x'_{2i+1}P$ 和 $x'_{2i+2}P$ 都由 M_{2i+1} 产生；

Step4　成员节点 M_{2i+1} 到 M_{2n+2} 广播新的密钥资料；

Step4　M_{2i+1} 广播更新密钥树 BPT$M_{2i+1}\xrightarrow{BPT_{\langle n\rangle}}C-M_{2i+2}$；

Step5　由每个节点重新计算得到从 k'_i 到 k'_n 的共享密钥。

(2)离开节点所在层为 1 个节点，协议步骤如下：

Step1　成员 M_{2i+1} 向 BPT 中所有成员广播离开消息；

Step2　成员接到消息后，移除层 $IN_{\langle i\rangle}$ 到 $IN_{\langle n\rangle}$ 的共享密钥；

Step3　从层 $IN_{\langle i+1\rangle}$ 到层 $IN_{\langle n\rangle}$ 的孩子节点重新选择共享密钥资料 $\{x'_{2i+3}P,x'_{2i+4}P,\cdots,x'_{2n+2}P\}$；

Step4　成员节点 M_{2i+3} 到 M_{2n+2} 广播新的密钥资料；

Step5　M_{2i+3} 广播更新密钥树 BPT$M_{2i+3}\xrightarrow{BPT_{\langle n-1\rangle}}C-M_{2i+1}$；

Step6　由每个节点重新计算得到从 k'_{i+1} 到 k_n' 的共享密钥。

3.4　合并

当两个群合并时，执行合并操作，较大群作为较小群的子树，即较小群的根作为新群的根节点。设合并的群为 S_1 和 S_2，目标群为 S_D。S_1 中 BPT 的层高为 n，S_2 中 BPT 的层高为 m，且 S_1 的规模大于 S_2($n\geqslant m$)，其步骤如下：

Step1　S_1 和 S_2 合并，S_1 中成员 LN^{s1}_{2n+1} 与 S_2 中成员 LN^{s2}_1 向群中所有节点广播更新消息，S_1 作为 S_2 的子树；

Step2　成员 LN^{s2}_1 建立新层 IN_{n+1} 并共享密钥资料 $x'_{1s_1}P$，由节点 IN^{s1}_n 贡献共享密钥资料 k_nP，节点计算层 IN_{n+1} 的共享密钥资料 $k_{n+1}=e(x'_{1s_1}P,k_nP)^{x'_{1s_1}}$；

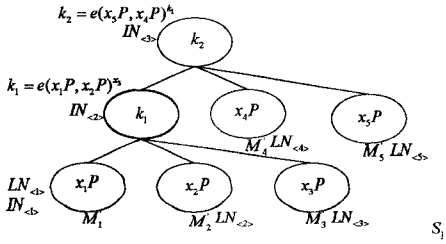
Step3　成员 LN^{s2}_1 将 k_{n+1} 广播给群 S_1 中的所有成员；

Step4　S_2 中所有成员重新生成共享密钥资料 $\{k'_{2s_2}P, k'_{3s_2}P,\cdots,k'_{ms_2}P\}$，重新计算各层的共享密钥，$k_i=e(x_{2i}P, x_{2i+1}P)^{k_{i-1}}$，且 $i\in[n+2,n+m+1]$；

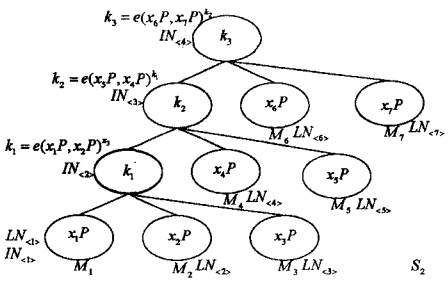
Step5　S_D 中从层 $n+1$ 到 $n+m+1$，由成员 LN_{2i} 向节点 IN_i 中的左子树成员广播该层的共享密钥 k_i；

Step6　成员节点 $LN_{2(n+m+1)}$ 向所有成员广播密钥树 $LN_{2(n+m+1)}\xrightarrow{BPT_{n+m+1}}S_D=S_1+S_2$。

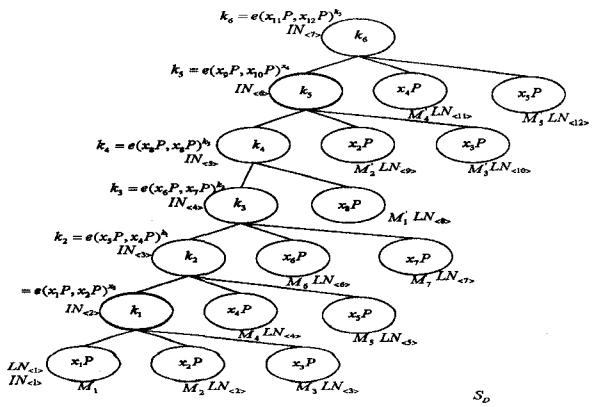
合并操作如图 4 所示。



(a)群 S_2



(b) 群 S_1

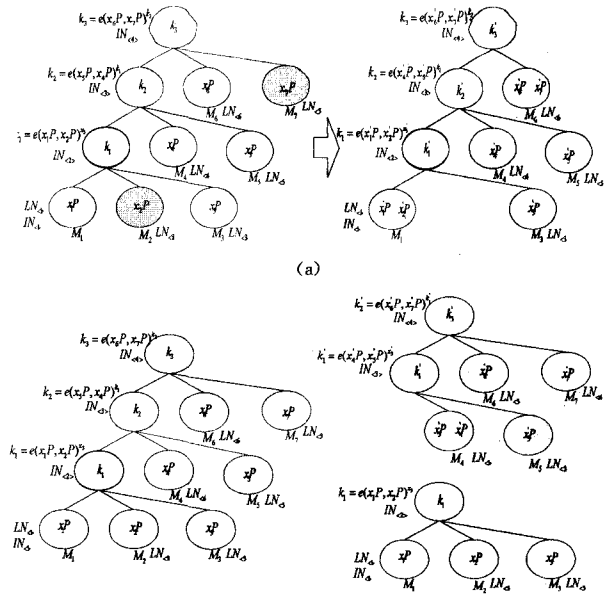


(c)群 S_D

图 4　合并操作

3.5　分离

分离操作分为两类：离开成员属不同层，如图 5(a)所示，则分离操作分解为多个离开操作；离开成员将 BTP 分为两个子树，如图 5(b)所示，包括 BTP 根节点的子树，结构不变，其成员重新选择共享密钥资料，计算各层的共享密钥，而不包括根节点的子树，BTP 各层的共享密钥不变，树结构也不变。



(a)

(b)

图 5　两种分离操作

4　性能分析

4.1　安全性

通过密钥的新鲜性、独立性保证 BPGKM 安全性。当群成员发生变化，群成员重新选择密钥资料并计算新群共享密钥，保证群共享密钥的安全性。拓扑发生变化前，群共享密钥为 $k_i=e(x_kP,x_jP)^{k_{i-1}}$，依据 BDHP 难解问题，攻击者即使获取 x_kP、x_jP 和 $k_{i-1}P$，也不能得到 x_k、x_j 和 k_{i-1}，由此也不能计算出 k_i。同理，当拓扑发生变化后，成员节点重新选择密钥资料 $x_{k'}P$、$x_{j'}P$ 和 $k'_{i-1}P$，计算更新密钥 $k'_i=e(x_{k'}P,x_{j'}P)^{k'_{i-1}}$，攻击者即使获取 $x_{k'}P$、$x_{j'}P$ 和 $k'_{i-1}P$，也不能得到 $x_{k'}$、$x_{j'}$ 和 k'_{i-1}，由此 k_i 和 k_i' 具有相同的密钥强度，即新密钥被攻破的成功概率与旧密钥一致，保证了密钥新鲜性。

证明前向安全性和后向安全性，即证明退出节点或新加入节点与攻击者从新密钥中获取的信息一样多。节点 M_{2n+1} 退出网络后，由该层重新计算密钥资料 $x'_{2n+1}P$ 和 $x'_{2n+2}P$，由于 M_{2n+1} 只有 $x_{2n+1}P$、$x_{2n+2}P$ 和 $k_{n-1}P$，而无法获取 x'_{2n+1}、x'_{2n+2} 和 k_{n-1}，由此，与攻击者获取关于密钥的信息一样多，即无法破解 k'_n，从而保证了前向安全性。同理，新加入的节点获取新的共享密钥资料 $x'_{2n+1}P$、$x'_{2n+2}P$ 和 $k'_{n-1}P$，计算新共享密钥 k'_n，由于新加入节点不能获取 x_{2n+1}、x_{2n+2} 和 k_{n-1}，由此不能得到旧密钥 $k_n=e(x_{2n+1}P,x_{2n+2}P)^{k_{n-1}}$，因此，新加入者和攻击者从 k_n 中获取的信息一样多，保证了后向安全性。

4.2　效率

由于 BPGKM 的密钥树 BPT 每个节点具有 3 个孩子，比 STR 方案多 1 个，因此更能适应较大规模的网络。同等规模的网络 N，BPGKM 的密钥树的深度 $[\frac{N-1}{2}]$ 小于 STR 的密钥树深度 N。由此 BPGKM 在执行加入、合并操作时和 STR 方案具有同样的效率，而在执行离开和分离操作时，其密钥计算操作由 $\frac{3N}{2}+2$ 降为 $\frac{3N}{4}+2$。由此 BPGKM 保持了 STR 在加入和合并上的优越性，同时提高了离开和分离的效率。

(下转第 108 页)

密同态加密方案的安全性，且与所采用的秘密同态加密方案是同等安全的。

2）end-to-end 可认证性

源节点 s_i 对感知信息密文 C_i 计算标签 $tag_i = E'(H(C_i), k_i)$；融合节点 A_j 收到密文 C_i 后计算标签 $tag_i^{A_j} = E'(H(C_i), k_{A_j})$。

若存在敌手篡改密文 C_i 为 C_i'，则基站计算 $C_i = D'(tag_i, k_i)$ 与 $C_i' = D'(tag_i^{A_j}, k_{A_j})$ 使得式(1)不成立，即融合信息密文 $C_{Agg}^{A_j}$ 无法通过验证，因此应丢弃。

若存在敌手冒充 s_i 发送密文 C_i'，则基站在接收到 $tag_i' = E'(H(C_i'), k_i)$ 后，由于基站采用与 s_i 共享的密钥无法解密 tag_i'，即融合信息密文 $C_{Agg}^{A_j}$ 无法通过验证，应丢弃。

因此，该安全数据融合协议的 end-to-end 可认证性是由协议所使用的对称加密算法来保障的。

本协议需要传输的标签数量为参与协议的源节点总数的 2 倍，但是由于在计算标签之前，协议首先对密文信息进行了散列，并采用对称加密算法加密散列值，且不存在信息交互，因此在计算量与通信量上，较文献[13]类采用的数字签名方案，依然存在明显的效率改善。

结束语 本文针对同态加密技术下的 end-to-end 可认证性保障问题，规避了多源多消息的同态签名技术与同态消息认证码技术，构造了新的安全数据融合认证方案，进一步构造了安全的数据融合协议。该协议能提供 end-to-end 的机密性与可认证性，有效解决了融合技术与安全技术在需求上的冲突。

在以后的工作中，将进一步深入探讨安全数据融合认证模型与方案，研究多源多消息的同态消息认证码技术与同态签名技术，彻底解决同态加密技术下的 end-to-end 可认证性保障问题。

参 考 文 献

[1] Akkaya K, Demirbas M, Aygun R S. The impact of data aggregation on the performance of wireless sensor networks[C]// Wireless Communication & Mobile Computing. 2008: 171-193

[2] 康健，左宪章，唐力伟，等. 无线传感器网络数据融合技术[J]. 计算机科学，2010，37(4)：31-35

[3] Ozdemir S, Xiao Y. Secure data aggregation in wireless senor networks: A comprehensive overview [J]. Computer Network, 2009, 53: 2022-2037

[4] Castelluccia C, Chan Aldar C-F, Mykletun E, et al. Efficient and provably secure aggregation of encrypted data in wireless sensor networks[J]. ACM Transactions on Sensor Networks, 2009, 5(3): 1-36

[5] Yang Y, Wang X, Zhu S, et al. SDAP: A secure hop-by-hop data aggregation protocol for sensor networks[J]. ACM Transactions on Information and System Security, 2008, 11(4): 1-43

[6] Westhoff D, Girao J, Acharya M. Concealed data aggregation for reverse multicast traffic in sensor networks: encryption key distribution, and routing adaptation[J]. IEEE Transactions on Mobile Computing, 2006, 5(10): 1417-1431

[7] Domingo-Ferrer J. A provably secure additive and multiplicative privacy homomorphism[C]// Chan A H, Gligor V D, eds. ISC 2002. vol. 2433, 2002: 471-483

[8] Peter S, Piotroeski K, Langendoerfer P. On concealed data aggregation for wireless sensor networks[C]// Proc. IEEE Consumer Communications and Networking Conference. 2007: 192-196

[9] Boneh D, Freeman D, Katz J, et al. Signing a linear subspace: signature schemes for network coding[C]// Public Key Cryptography. PKC 2009, LNCS 5443. Springer Verlag, 2009: 68-87

[10] Chan Aldar C-F, Castelluccia C. On the (im)possibility of aggregate message authentication codes[C]// ISIT 2008. Toronto, Canada, 2008: 235-249

[11] Hung-min S, Yue-hsun L, Ying-chu H, et al. An efficient and verifiable concealed data aggregation scheme in wireless sensor networks[C]// The 2008 International Conference on Embedded Software and Systems. 2008: 19-26

(上接第 60 页)

4.3 鲁棒性

BPGKM 方案密钥树中每个内部节点有 3 个孩子节点，当某节点离开时，如果所在层具有两个兄弟节点，则网络拓扑图不发生变化；同理，有新节点加入，如果所在层为一个兄弟节点，则网络拓扑图也不发生变化。而在 STR 中节点加入和退出都需要改变密钥树结构，因此相对 STR 方案，BPGKM 减少了网络间的通信负载，保证了群密钥树的稳定性。

结束语 由于现有的群密钥管理方案基于 GDH 密钥协议，密钥树具有较大深度，降低了节点的群密钥操作效率，针对这一问题，本文提出了基于双线性映射的群密钥管理方案，该方案在保证安全性的前提下增加了密钥树每层的节点数，减少了同等规模网络下的密钥树的深度，提高了密钥管理效率。相对于 STR 方案，本文方案保持了群密钥操作中加入和合并的效率，同时使得离开和分离操作的效率提高一半。本文方案由于在每层上具有较多的节点，因此比现有方案具有更为稳定的密钥树结构。由此 BPKGM 更能适用大规模网络的群安全。在下一步工作中，将进一步研究 BPKGM 在 Ad-hoc 网络中的效率。

参 考 文 献

[1] Chlamtac I, Conti M, Liu J. Mobile Ad-hoc Networking: Imperatives and Challenges[J]. Ad-hoc Networks, 2003, 1(1)

[2] Basagni S. Mobile Ad-hoc Networking[M]. IEEE Press and Wiley, 2004

[3] Diffie W, Hellman M. New Directions in Cryptography[J]. IEEE Transaction on Information Theory, 1976, 22(6): 644-654

[4] Steiner M, Tsudik G, Waidner M. Diffie-Hellman Key Distribution Extended to Group Communication[C]// Proceedings of ACM CCS. ACM Press, 1996: 31-37

[5] Kim Y, Perrig A, Tsudik G. Group Key Agreement Efficient in Communication[J]. IEEE Transactions on Computers, 2004, 53(7): 905-921

[6] Performance of Group Key Agreement Protocols [BP/OL]. http://www.cnds.jhu.edu/pub/papers/cnds-2001-5.pdf, 2001

[7] Sakai R, Ohgishi K, Kasahara M. Cryptosystems Based on Pairing[C]// Symp. on Cryptography and Information Security. Okinawa, Japan, Springer, 2000: 26-28

[8] Joux A. An One Round Protocol for Triphartite Diffie-Hellman [C]// Proceedings of ANTS. Heidelberg: Springer-Verlag, 2000: 385-394