基于椭圆曲线的安全组播多层接入控制方案*)

徐守志　谭运猛　杨宗凯　程文青
(华中科技大学电子与信息工程系　武汉 430074)

摘　要　具有多层服务结构的组通信要求有多层接入控制的能力。已有的研究方案在密钥更新时存在开销大、时延长的缺点,使组播的服务质量难以保证。本文提出基于椭圆曲线密钥体制的斜树密钥管理方案,算法只需更新少量密钥就能满足系统的安全需求。与已有方案相比,效率有很大提高。

关键词　安全组播,多层接入控制,椭圆曲线,多层服务

Elliptic Curve Cryptosystem Based Scheme of Hierarchical Access Controlling for Secure Multicast

XU Shou-Zhi　TAN Yun-Meng　YANG Zong-Kai　CHEN Wen-Qing
(Department of Communication Science & Technology, Huazhong University of Science and Technology, Wuhan 430074)

Abstract　Multi-layer services group communications need function of hierarchical access controlling. Existing methods have disadvantages of high cost and long delay, which makes they not assure the quality of service(QoS) of multicast. An inclined tree-group key management scheme based on elliptic curve cryptosystem is presented in this paper, which needs to renew only several node keys to provide perfect secrecy. It is more efficient than previous works.

Keywords　Secure multicast, Hierarchical access control, Elliptic curve cryptosystem, Mmulti-layer services

具有多层服务结构的多媒体组播有广泛的应用前景。这种组播要求具有多层接入控制,即用户加入的等级不同,所享有的服务也不同,等级高的用户享有所有等级低的服务[1]。IETF 提出相关安全标准[2],主要包括:加入用户不能解密加入前的组播数据(后向安全);离开的用户不能获得将来的组密钥(前向安全)。如何安全高效地更新组会话密钥(SK: session key)一直是研究的难题。目前的主要成果包括基于逻辑层次密钥树(LKH: Logical Key Hierarchy)[3]和单向函数树方案(OFT: One-Way Function Tree)[4]。LKH 大多是基于 Diffe-Hallman,需要进行大量的幂指数运算。OFT 应用 Hash 函数以减小通信开销,但存在串通攻击的安全问题。而且这些成果若应用在多层服务的安全组播中,需要对多个子组分别管理,系统的存储开销、更新开销都很大,因此难以有效地应用。

鉴于此,本文提出基于椭圆曲线密钥体制的斜树密钥管理方案,简化算法设计。分析算法的安全性,并对算法的效率进行分析和仿真比较。

1　基于椭圆曲线的组共享密钥

迄今的研究表明,椭圆曲线密码体制(Elliptic Curve Crypto system,简称 ECC),除超奇异曲线和异常曲线外,其求解算法都是指数时间算法[5],能以较小的开销实现较高的安全性。

1.1　椭圆曲线 $E_p(a,b)$ 的离散对数难题

定义 1　设 $GF(p)$ 是一个特征 $p\neq 2,3$ 的有限域,$\exists a,b\in DF(p)$ 满足 $4a^3+27b^2\neq 0$,$p\in(2^t,2^{t+1})$,且 p 为一个大素数。取方程:$E: y^2=x^3+ax+b$,则 $E_p(a,b)=\{(x,y)\mid y^2=x^3+ax+b(\mathrm{mod}\ p)\}\cup O$ 按照文[5]定义的零元、逆元、加法+

*)基金项目:国家自然科学基金资助项目(90104033)。徐守志　副研究员,博士生,研究方向为网络安全、应用密码学;杨宗凯　教授,博士生导师,研究方向为现代信息网络理论及其应用和现代数字信号处理技术;谭运猛　博士,副教授,研究方向为电子支付、应用密码学。

轮询调度只能保证每个引擎所完成任务数目是基本相同的,而不能保证每个引擎的负载基本均衡。特别地,当业务过程所需的时间较长时,负载平衡调度的优点更加突出。当多引擎间任务调度出现不平衡,导致某些节点负载过重或是某个节点出现故障时,多引擎能够根据相应模型策略和实际的信息来动态调整各节点的负载,这样也可以在一定程度上解决在有大量分布式事务处理时性能问题。下一步,我们还应该继续完善负载平衡算法,使之根据不同实际应用采用不同的粒度来衡量负载指数,达到大量的简单任务时采用粗粒度来衡量,而对比较复杂的任务采用细粒度来衡量。

参 考 文 献

1　陈华平,陈国梁. 分布式动态负载平衡调度的一个通用模型,1998
2　Baker S, Moon B. Distributed Cooperative WebServers. In: Proc. of the 8th Intl. WorldWide Web Conf. 1999
3　Farrell R. Distributing the Web load. Network World, 1997
4　傅强. 一种适用于机群系统的任务动态调度方法,1999
5　Ferrari D, Zhou S. A load index for dynamic load balancing. In: 1986 Proc. of Fall Joint Computer Conf. Dallas, Texas, Nov. 1986
6　Milojicic D S, Pjevac M. LoadBalacing Survey. In: Proc. of the Autumn 1991 EurOpen Conf.
7　WestBrook J. Load Balancing for Response Time. Journal of Algorithm, 2000, 35: 1～16
8　Barbara D, Mehrotra S. INCAs: managing dynamic workflows in distributed environments. Journal of Database Management, Special Issue on Multidatabases, 1996

和数乘 $*$，组成一个 Abel $E_p(a,b)$群。

定理 在 $E_p(a,b)$上选定任一点 $M\neq O$，有：$Q=k*M\in E_p(a,b)$，$k\in(0,p)$。对于给定的点 M 和 k，可以容易计算出 Q，给定 Q 和 M 却很难计算出 k。这就是离散对数难题(ECDLP：Elliptic Curve Discrete Logarithm Problem)[5]。

引理 1 对 $P_{1,2}=n_1*n_2*G$，已知 $P_{1,2}$，n_2 和 G，计算 n_1 不可行。

这实质是基于 ECC 的密钥交换协议，U_1 将公钥 $P_1=n_1G$ 传给 U_2，U_2，计算 $P_{1,2}=n_2P_1$ 传给 U_1。依定理，U_1 通过 $P_{1,2}$恢复出 n_2 属于 ECDLP 难题。

1.2 组共享密钥

组控制者(GC：Group Controller)选择随机数 n 作为私钥，将 n 与 $GF(p)$的生成元 G 数乘得公钥 P。并为每用户 U_i 产生公私密钥对(n_i,P_i)。

定义 2 组共享密钥 $SK=\prod n_i*G(\mathrm{mod}\ p)$，$i\in[1,t]$。即嵌入所有用户私钥信息($t$ 为组用户数)。

依据以上的定理和推理，可得：

推论 1 设 $S_{x,y}p\prod n_jG(j\in[x,y])$，$S^k_{x,y}=\prod n_jG(j\in[x,y]$且 $j\neq k)$，已知 $S_{x,y}$和 $S^k_{x,y}$，计算 n_k 不可行。证明如下：

1)设 M 是 $E_P(a,b)$中一点，由 $M=mG$ 产生($m\in[1,p-1]$)，$\forall k\in Z_n$，$\exists Q=kM\in E_p(a,b)$，$E_p(a,b)$为有限循环群，故 $\exists r,s\in Z,s>r$，使 $sM-rM=(s-r)M=O$，定义满足 $n_MM=O$ 的最小整数 n_M 为周期，显然$\{O\cup iM:i\in[1,n_M]\}$构成了 $E_p(a,b)$中的一个循环子群，记为 $E^M_p(a,b)$，M 为子群的生成元。

2)选定 n 为大素数，$\forall k,m\in[1,n-1]$，$\gcd(mk,n)=1$，即$\{mk:k\in Z_n\}$为 Z_n 的满射，因此 $F:E^M_p(a,b)\to E_p(a,b)$是一个满射，两个群同构，M 为一个本原元。显然 $E^M_p(a,b)$与 $E_p(a,b)$具有同样的安全性。

3)将 U_k 的公钥 P_k 作为本原元，可构成与 $E_p(a,b)$同构的群，记为 EP_k。则 $\forall n_j\in[1,n-1](j\in[x,y])$，有 $\prod n_j(\mathrm{mod}\ n)\in[1,n-1]$，$\prod n_jG\in EP_k$，故已知 $S_{x,y}$和 $S^k_{x,y}$，计算 n_k 不可行。

推论 2 设 N 的密钥 $S_{x,y}=\prod n_jG(j\in[x,y])$，$S_{x,z}$和 $S_{z+1,y}$为 N 的左右子结点密钥。已知 $S_{x,z}$和 $S_{x,y}$，计算 $S_{z+1,y}$不可行。

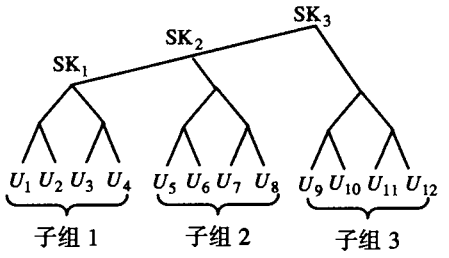
证明：令 $n_{x,y}=\prod_{nj}(j\in[x,y])$，$n_{x,z}=n_{x,y}=\prod n_j(j\in[x,y])$，$n_{z+1,y}=\prod n_j(j\in[z+1,y])$。则：$S_{x,y}=n_{x,y}=n_{x,y}*G=n_{x,z}*n_{z+1,y}*G$，即 $S_{x,y}=n_{z+1,y}*S_{x,z}$。由引理 1 知，$S_{x,z}$和 $S_{x,y}$，计算 $n_{z+1,y}$不可行，从而计算 $S_{z,y}$不可行。

2 组密钥管理协议

本节提出基于 ECC 的斜树组播密钥管理方案，简称为 HEGKM。

2.1 密钥树生成

用户按照等级分成多个子组，每个子组按满树建立逻辑二叉树，各子树按照子组的安全级别倒挂在密钥树上，逻辑顶点的安全级别最低，叶结点为用户结点。图 1 为三个子组 12 个用户的实例，子组 1 的成员级别最高，拥有三个子组密钥$\{SK_1,SK_2,SK_3\}$，而子组 3 的成员级别最低，只拥有$\{SK_3\}$。图 1 在形式上与单组播流的密钥树类似，但各子组相对独立，利于简化算法的设计。



密钥树上任意父结点的密钥值为两子结点的交换公钥(但并非交换公钥得到)，如图 1 中，$SK_1=\prod n_j*G(j\in[1,4])$，$SK_2=\prod n_j*SK_1(j\in[5,8])$。

下面简要说明组密钥更新算法的主要步骤。

2.2 单用户加入算法

当用户 U_a 请求加入，若允许，GC 按完全二叉树将加入所选择的子组[4]，为保证后向安全，其子组级别以下的 SK 都需要更新。如 U_a 加入子组 1，GC 需更新$\{SK_1,SK_2,SK_3\}$。主要步骤如下：

1)GC 选择一个大素数 n 作为组密钥更新因子，用斜树顶点的密钥加密。

2)形成并组播$\{join,sg1,ID(U_1),E_{SK_3}(n)\}$。

3)所有用户用 SK_3 解密得到更新因子，并用 $SK'=n*SK$ 更新所拥有的子组会话密钥。

4)将 U_a 接入路径上的密钥和 ID 单播给 U_a。

2.3 单用户移出算法

用户 U_a 拥有的上级结点，记为 F_a，对应的邻结点记为 $S(F_a)$。当 U_a 请求离开时，为保证前向安全，F_a 的密钥需全部更新。主要步骤如下：

1)GC 选择组密钥更新因子 n。

2)形成并组播数据包$\{leave,sg1,ID(U_a),E_{KS}(n):KS\in\{SI(F_a)\cup S(U_a)\}\}$。

3)所有在线用户可以根据$\{S(F_a)\cup S(U_a)\}$解密得到更新因子，并用 $KF'=n*KF$ 更新 F_a。

为了适应成员动态变化频繁的情况，系统采用批量更新的算法，算法可以参见文[6]，限于篇幅，本文不深入讨论。

3 更新算法的安全性分析

算法的安全性可以从两方面分析：

1)对任何加入用户，更新因子 n 由原会话密钥加密通知，故不能解密 $E_{SK}(n)$。由 ECC 的定理知，加入用户不能通过 $SK=1/n*SK'$恢复出原会话密钥，保证了后向安全。

2)由推论 1、推论 2 知，任何移出用户 U_a 不可能知道$\{S(F_a)\cup S(U_a)\}$的结点密钥，因而不能解密 $E_{KS}(n)$以获得更新因子，从而不能更新会话密钥，保证了前向安全。

4 效率分析与仿真实验

记 N_R、C_R、N_E、N_{sep}分别为更新的密钥数、组播包数、加密次数和算法的迭代步骤，这四项指标可以较全面地反映系统的效率性能。为了简化分析，设有 m 个子组，用户平均分配在各子组上，用户改变行为发生在安全级别最高的子组，此时系统的更新开销最大。

4.1 效率分析

OFT 方案与 LKH 方案类似，但采用单向函数将 C_R、N_E、N_{sep}三项指标减小一半。若采用传统方法，为每个子组建立密钥树，系统的存储开销和密钥更新开销将很大。若应

用文[1]的体系结构，可以提高系统效率。表1分析在文[1]框架下OFT、LKH的性能。而HEGKM中，当用户离开或加入，组更新只需一个组播包和一次迭代，并大大减少密钥更新量。

表1 典型密钥管理方案性能比较

指标	单用户离开				单用户加入			
	N_R	C_R	N_E	N_{step}	N_R	C_R	N_E	N_{step}
LKH	$k-1$	$2k-3$	$2k-3$	$2k-3$	k	$2k$	$2k$	$2k$
LFT	$k-2$	$k-1$	$k-1$	$k-1$	k	k	k	k
HEGKM	$m+1$	1	$k-1$	1	m	1	1	1

注：$h=\log(t/m, k=h+m)$

4.2 仿真结果

实验中，用户总数为2^{11}，子组数为2。实验采用改进的LKH[3]和OFT[4]方案，对相同的批量用户改变模式的更新效率进行比较。

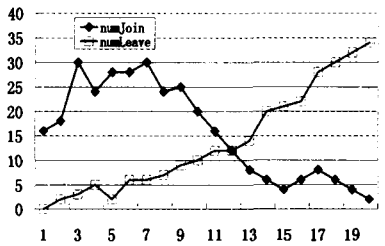


图2 用户改变模式

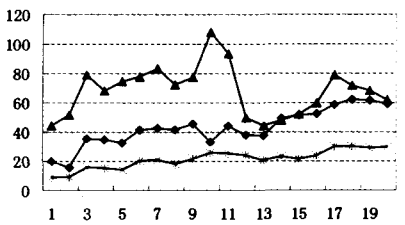


图3 时间开销比较

图2为实验中用户改变量的变化图，开始阶段加入数高于离开数，随后离开数越来越多，加入数越来越少。由于OFT将加入用户形成一族后再整体加入，因此当加入数高于离开数时，OFT的优势明显。但图3、4说明HEGKM算法的时间开销和带宽开销最小，组播带宽开销与加入用户数几乎无关。

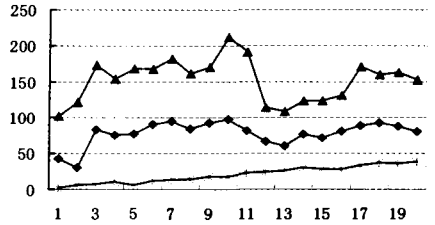


图4 带宽开销比较

结论 本文针对具有多层接入控制的安全组播，提出基于椭圆曲线密钥体制的斜树密钥管理方案，减小了系统的密钥存储开销和算法的复杂性，效率分析和仿真实验结果表明算法的效率较其它典型方案有较大提高。

参考文献

1 Sun Y, Liu K J R. Scalable hierarchical access control in secure group communications[A]. INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies[C], 2004, 2(7-11): 1296～1306

2 Wallner D, Harder E, Agee R. Key Management for Multicast: Issues and Architectures. RFC 2627, 1999

3 Pegueroles J, Rico-Novella F, Hernandez-Serrano J. Improved LKH for batch rekeying in multicast groups [A]. In: International Conference on Information Technology [C]. New York(USA): IEEE, 2003. 269～273

4 Sherman A T, McGrew D A. Key establishment in large dynamic groups using one-way function trees [J]. IEEE Trans. on Software Engineering, 2003, 29(5): 444～458

5 张方国，万育民. 超椭圆曲线密码体制的研究与进展[J]. 电子学报，2002, 30(1): 126～131

6 Yang Zongkai, Xu Shouzhi, Tan Yunmeng. An efficient key updating scheme for multicast key management [A]. Proceedings of the SPIE - The International Society for Optical Engineering [C], 2005, 5626(2): 1096～1104

（上接第86页）

3 Jensen C S, Pedersen T B. Mobile E-Services and Their Challenges to Data Warehousing. Datenbank Rundbrief, 2001, 27: 8～16

4 Jensen C S. Research Challenges in Location-Enabled M-Services. In: Proceedings of the Third International Conference on Mobile Data Management. Singapore: IEEE Computer Society Press, 2002. 3～7

5 Jensen C S, Kligys A, Pedersen T B, et al. Multidimensional Data Modeling for Location-Based Services. The International Journal on Very Large Data Bases, 2004, 13(1): 1～21

6 李建中，高宏. 一种数据仓库的多维数据模型. 软件学报，2000, 11(7): 908～917

7 Pedersen T B, Jensen C S. Multidimensional Data Modeling for Complex Data. In: Proc. of 15th ICDE, IEEE Computer Society, pp. Sydney, Australia, 1999. 336～345

8 Franconi E, Kamble A. The GMD Data Model for Multidimensional Information: A Brief Introduction. Data Warehousing and Knowledge Discovery, 5th International Conference, DaWaK 2003, Prague, Czech Republic, 2003

9 Trujillo J. The GOLD model: An Object Oriented multidimensional data model for multidimensional databases. In: Proceedings of the 16th International Conference on Advanced Information Systems Engineering(CAiSE-04), Riga, Latvia, 2004

10 Tsois A, Karayannidis N. MAC: Conceptual Data Modeling for OLAP. In: Proc. of the International Workshop on Design and Management of Data Warehouses (DMDW' 2001) Interlaken, Switzerland, 2001

11 LechtenbÖrger J, Vossen G. Multidimensional Normal Forms for Data Warehouse Design. Information System, 2003, 28(5): 415～434

12 陈明，吴国文，施伯乐. 数据仓库概念模型的设计. 小型微型计算机系统，2002, 23(12): 1453～1458

13 Tryfona N, Busborg F, Borch Christiansen J G. starER: A Conceptual Model for Data Warehouse Design. In: Proc. of the ACM 2nd Intl. Workshop on Data warehousing and OLAP(DOLAP' 99)

14 Blaschka M. FIESTA: A Framework for Schema Evolution in Multidimensional Databases: [PhD thesis, Technische]. Universitat Murchen, Germany, 2000

15 陆昌辉，邓苏，等. 基于UML的多维数据概念建模方法的研究. 系统工程与电子技术，2004, 26(4): 522～525