维普资讯 http://www.cqvip.com

基于 LUC 一般访问结构上的秘密共享方案及其安全性*)

庞辽军[1] 李慧贤[2] 王育民[1]
(西安电子科技大学综合业务网国家重点实验室 西安 710071)[1]
(大连理工大学计算机科学与工程系 大连 116024)[2]

摘 要 提出了一个基于 LUC 公钥算法的一般访问结构上的秘密共享方案。它使用参与者的私钥作为各自的秘密份额,分发者无需进行秘密份额的分配。在秘密重构过程中,每个合作的参与者只需提交一个伪份额而不必暴露其私钥,同时方案提供了预防和检测欺骗的能力。该方案可以用来共享任意多个秘密,而不必更新各参与者的秘密份额。方案的安全性是基于 LUC 算法以及 Shamir 门限方案的安全性。

关键词 秘密共享,访问结构,安全性

Secret Sharing Scheme with General Access Structures Based on LUC and its Security

PANG Liao-Jun[1] LI Hui-Xian[2] WANG Yu-Min[1]
(State Key Lab. of Integrated Service Networks, Xidian Univ., Xi'an 710071)[1]
(School of Electronic and Information Engineering, Dalian Univ. of Tech., Dalian 116024)[2]

Abstract Based on the LUC public-key algorithm, a new secret sharing scheme with general access structures is proposed in this paper. In this scheme, each participant's private-key is used as his secret shadow and the secret dealer doesn't have to distribute any secret shadow. In the recovery phase, each cooperative participant only needs to submit a pseudo-share instead of his private-key and anyone is allowed to check whether a cooperative participant provides the true information or not immediately. The secret shadows do not need to be changed when sharing multiple secrets. The security of this scheme is based on the security of the LUC algorithm and Shamir's threshold scheme.

Keywords Secret sharing, Access structure, Security

1 引言

秘密共享是现代密码学领域中一个非常重要的分支,也是信息安全方向一个重要的研究内容。秘密共享方案主要由秘密分发算法和秘密重构算法构成。在执行秘密分发算法时,分发者将秘密分割为若干份额,并将其安全分发给各个参与者,使得访问结构[1]中任一授权子集[1]的参与者可以合作重构该秘密,而非授权子集中的参与者合作不能得到秘密的任何信息。秘密共享的概念最早是由 Shamir[2] 和 Blakley[3] 提出的,它是一个(t, n)门限方案,任意 t 个参与者的集合都是授权子集。Banaloh 等[4]指出了门限方案仅适合门限访问结构的局限性,并提出一般访问结构上的秘密共享方案,可以参考文[5]来获得进一步的了解。现有大多数秘密共享方案都具有这样的特点:一是各参与者的秘密份额都是由秘密分发者产生,秘密分发者掌握着所有参与者的秘密份额。这使得秘密分发者需要保存大量的秘密信息,增加了秘密分发者的存储复杂度,而且会使得秘密分发者成为攻击者所攻击的目标;二是在秘密分发者和各参与者之间需要一条安全信道,利用该信道进行秘密份额的分发。维护一条安全信道会提高系统的代价和复杂度,而且会引入新的攻击点。如果攻击者攻破了该信道,那么他就可以获得任意参与者的秘密份额,从而可以任意地恢复共享的秘密。这些特点或多或少会影响秘密共享方案的实际应用。比如,当参与者和秘密分发者不可能存在安全信道时,这些方案也将不再有用。

本文提出了一个基于 LUC 密码体制[6,7]的一般访问结构上的秘密共享方案,它使用参与者的私钥作为其秘密份额,秘密分发者和各参与者之间可以以明文形式相互通信,不需要维护安全信道。在秘密重构过程中,每个合作的参与者只需提交一个由秘密份额计算的伪份额,而且任何人都可以立即判断每个合作的参与者是否进行了欺骗。方案的安全性是基于所使用的 LUC 密码体制和 Shamir 的(t,n)门限方案的安全性。

2 LUC 密码体制简介

LUC 是新西兰学者 P. Smith 等[6]提出的双钥密码体制。LUC 密码体制是采用 Lucas 数列来实现消息的加密和解密。本小节将对其做一简单的介绍。

2.1 Lucas 数列

Lucas 数列可以定义为:

定义 1 选两个非负整数 P 和 Q,构成二次式 $x^2-Px+Q=0$,其根为 α,β

$$\alpha,\beta=\frac{P\pm\sqrt{D}}{2} \tag{1}$$

其中 D 是方程的判别式,即 $D=P^2-4Q$。如果选 P 和 Q,使

*)国家 973 项目资助课题(G1999035805);"十五"军事通信技术预研项目资助课题(Y1010122)。庞辽军 博士生,主要研究方向为电子商务中的安全理论与技术;李慧贤 博士生,主要研究领域为电子商务中的安全理论与技术;王育民 博士生导师,主要研究方向为信息论、密码、编码。

维普资讯 http://www.cqvip.com

$D\neq 0$，则 Lucas 数列可定义为：

$$U_n(P,Q)=\frac{\alpha^n-\beta^n}{\alpha-\beta} \qquad n\geqslant 0 \tag{2}$$

$$V_n(P,Q)=\alpha^n+\beta^n \qquad n\geqslant 0 \tag{3}$$

LUC 公钥体制仅对 $V_n(P,Q)$ 序列感兴趣。有关 $V_n(P,Q)$ 的性质及证明可以参见文[6，7]，这里仅给出本文所用到的性质。

性质 1[6] 设 a,b 为任意正整数，则有 $V_{ab}(P,1)=V_a(V_b(P,1),1)$。

性质 2 设 a,b 为任意正整数，则有 $V_b(V_a(P,1),1)=V_a(V_b(P,1),1)$。

证明：由性质 1 可以得到，$V_b(V_a(P,1),1)=V_{ba}(P,1)=V_{ab}(P,1)=V_a(V_b(P,1),1)$。 □

2.2 LUC 密码体制

令 $N=pq$，为两个奇素数之积；选一个整数 e，使 $(e,\phi(N))=1$，这里 $\phi(N)$ 是欧拉函数；并由式 $ed(1 \bmod \phi(N)$ 确定出另一整数 d。构造 LUC 体制可以简单地表示如下：

公钥： N,e；

私钥： d（陷门信息 p,q）；

明文： P 为小于 N 的某个整数；

密文： $C=V_e(P,1) \bmod N$；

解密： $P=V_d(C,1) \bmod N$。

有关 LUC 密码体制的正确性、安全性证明以及密钥的选取请参考文[6,7]，这里不再赘述。

3 方案构成

系统由 n 个参与者和一位为各参与者所信赖的秘密分发者组成。不失一般性，设 $P=\{P_1,P_2,\cdots,P_n\}$ 是 n 个参与者的集合。设 N 为两个足够大的奇素数 p 和 q 之积，即 $N=pq$；系统中每个参与者 P_i 的 LUC 公钥和私钥分别为 $\{N,e_i\}$ 和 d_i；秘密分发者的 LUC 公钥和私钥分别为 $\{N,e_d\}$ 和 d_d；令 Q 是一个随机选取的且大于 N 的素数；假设单调访问结构为 $\Gamma=[\gamma_1,\gamma_2,\cdots,\gamma_t]$；秘密分发者从 $[1,Q-1]$ 中选取 t 个不同的随机数 $d_1,d_2,\cdots,d_t$ 分别标识访问结构 Γ 中的每一个授权子集。秘密分发者将所有的公开信息向系统中所有成员广播，如 $N,Q,d_1,d_2,\cdots,d_t,\{N,e_1\},\{N,e_2\},\cdots,\{N,e_n\}$ 和 $\{N,e_d\}$ 等。由于素数 p 和 q 不再有用，予以销毁。

3.1 秘密分发过程

秘密分发者可以利用下面的算法将秘密 $s\in Z_Q$ 分发给这 n 个参与者 $P_1,P_2,\cdots,P_n$，使得访问结构 $\Gamma=\{\gamma_1,\gamma_2,\cdots,\gamma_t\}$ 中的每一个授权子集 γ_j 的参与者可以合作来重构该秘密，而非授权子集的参与者合作不能得到该秘密的任何信息。

(1) 从 $[N^{1/2},N-1]$ 中随机选取一个整数 r；

(2) 对于每一个参与者 $P_i(i=1,2,\cdots,n)$，秘密分发者与其进行以下交互过程：

(2.1) 将 r 送给参与者 P_i；

(2.2) 参与者 P_i 利用自己的私钥对 r 进行解密，即计算 $V_{d_i}(r,1) \bmod N$，并将结果送给秘密分发者；

(2.3) 秘密分发者可以利用 P_i 的公钥来验证 $V_{d_i} \bmod N$ 的正确性，即验证 $r=V_{e_i}(V_{d_i}(r,1),1) \bmod N$ 是否成立。如果不成立，说明参与者 P_i 没有诚实地给出自己计算的 $V_{d_i}(r,1) \bmod N$，或者可能消息在传送过程中出错。这时，秘密分发者可以向 P_i 发送一个抱怨信息，并要求其进行重发，直到验证通过，或者进行其它相应的出错处理。如果 $r=V_{e_i}(V_{d_i}(r,1),1) \bmod N$ 成立，秘密分发者利用自己的私钥对 $V_{d_i}(r,1) \bmod N$ 进行解密，得到 $V_{d_d}(V_{d_i}(r,1),1) \bmod N$。

(3)从 $[1,Q-1]$ 随机选取一个整数 b，构造一次多项式：$f(x)=s+bx \bmod Q$，并计算 $f(1)$。

(4)对于访问结构 Γ 中的每一个授权子集 $\gamma_j=\{P_{1j},P_{2j},\cdots,P_{dj}\}$ $(1\leqslant j\leqslant t)$，计算

$H_j=f(d_j)\oplus(V_{d_d}(V_{d_{1j}}(r,1),1) \bmod N)\oplus(V_{d_d}V_{d_{2j}}(r,1),1) \bmod N)\oplus\cdots\oplus(V_{d_d}(V_{d_{dj}}(r,1),1) \bmod N)$。

(5) 向系统中所有成员广播关于秘密 s 的公开信息：

$\text{Message}(s)=(r,V_{d_d} \bmod N,f(1),H_1,H_2,\cdots,H_t)$。

3.2 秘密重构过程

任何授权子集 γ_j 中的参与者合作，利用他们的私钥及公开信息 Message(s) 可以恢复秘密 s。不失一般性，我们选取授权子集 $\gamma_j=\{P_{1j},P_{2j},\cdots,P_{dj}\}$ 为例来说明该过程。秘密重构过程如下：

(1)每个合作的参与者 P_{ij} 利用自己的私钥计算关于秘密 s 的伪份额 $V_{d_{ij}}(V_{d_d}(r,1),1) \bmod N$，并将其提交给指定的秘密计算者。秘密计算者可以通过验证 $V_{d_d}(r,1)=V_{e_{ij}}(V_{d_d}(r,1),1),1) \bmod N$ 是否成立来验证参与者 P_{ij} 所提交的伪份额。如果成立，那么 P_{ij} 所提交的伪份额是正确的，接着执行下面的第(2)步；否则，P_{ij} 没有诚实地给出自己的伪份额，或者可能消息在传送过程中出错，这时秘密计算者可以向 P_{ij} 发送一个抱怨信息，并要求其进行重发，直到验证通过，或者进行其它相应的出错处理。

(2)由性质 2 可知，$V_{d_{ij}}(V_{d_d}(r,1),1)=V_{d_d}(V_{d_{ij}}(r,1),1)$。这时，秘密计算者可以通过下面的计算获得关于授权子集 γ_j 的信息 $f(d_j)$：

$$f(d_j)=H_j\oplus(V_{d_d}(V_{d_{1j}}(r,1),1) \bmod N)\oplus(V_{d_d}(V_{d_{2j}}(r,1),1) \bmod N)\oplus\cdots\oplus(V_{d_d}(V_{d_{dj}}(r,1),1) \bmod N)$$

(3)秘密生成者利用两个点：$(1,f(1)),(d_j,f(d_j))$ 和 Lagrange 插值法[2]重构多项式 $f(x)$：

$$f(x)=f(1)\frac{x-d_j}{1-d_j}+f(d_j)\frac{x-1}{d_j-1}=\frac{xf(1)-xf(d_j)-d_jf(1)+f(d_j)}{1-d_j}$$

$$=x(f(1)-f(d_j))(1-d_j)^{-1}+(f(d_j)-d_jf(1))(1-d_j)^{-1} \bmod Q \tag{4}$$

(4)计算秘密为 $s=f(0)$。

4 分析和讨论

本文所提方案的安全性是基于 LUC 公钥算法和 Shamir 的门限方案的安全性。

(1)本文所提出的方案利用了 LUC 密码体制的性质，以参与者的私钥作为秘密份额，秘密分发者不需与各参与者进行秘密通信即可使一群参与者共享任意的秘密。秘密分发者和各参与者之间的所有通信可以以明文形式进行。因此，该方案对于秘密分发者与参与者之间不存在安全通信信道的场合尤为有用。

(2)如果某个参与者 P_i 想进行欺骗，他可以在秘密分发过程的第(2.2)步计算 $V_{d_i}(r,1) \bmod N$ 时，或在秘密重构过程的第(1)步计算伪份额 $V_{d_i}(V_{d_d}(r,1),1) \bmod N$ 时进行欺骗。但是，由于 r 和 $V_{d_d}(r,1)$ 都是已知的信息，任何人都可以利用 P_i 的公钥进行验证，发现这种欺骗。

(3)系统外的攻击者可以设法推导出各参与者的私钥来

维普资讯 http://www.cqvip.com

对本方案进行攻击。他可以在秘密分发过程的第(2.2)步中，或在秘密重构过程的第(1)步中，根据各参与者 P_i 提交的信息 $V_{d_i}(r,1) \bmod N$ 或 $V_{d_i}(V_{d_d}(r,1),1) \bmod N$ 来推导出参与者 P_i 的私钥，即参与者 P_i 的秘密份额。由于 LUC 密码体制的安全性，攻击者的这种攻击无法奏效。同样道理，攻击者通过推导秘密分发者的私钥以进行模仿秘密分发者的攻击也是不可能的。

(4)由 Lagrange 插值多项式的性质[2]可知，在秘密重构过程中，要重构一次多项式 $f(x)$ 需要知道两个满足 $Y_i = f(X_i)$ 的不同的点 (X_i, Y_i)。利用公开信息只可以得到一个点 $(1, f(1))$。对于授权子集 γ_j 来说，其参与者的合作可以很容易地得到另外一个点 $(d_j, f(d_j))$；而非授权子集上参与者的合作不能得到这样的点。如果一个非授权子集 $\overline{\gamma_j}$ 上的参与者或者攻击者想要得到点 $(d_j, f(d_j))$，需要设法得到 γ_j 上任何一个参与者 P_{ij} 所提供的子秘密 $V_{d_{ij}}(V_{d_d}(r,1),1) \bmod N$，否则只能仅凭猜测来得到点 $(d_j, f(d_j))$。而要对 $f(d_j)$ 进行猜测，其成功的概率仅为 $1/Q$。当 Q 足够大时，这种攻击的成功概率接近于 0。

(5)该方案也是一个多重秘密共享方案[8]，可使一群参与者利用他们各自的私钥共享任意多个秘密而不必更新他们的私钥。为了共享多个秘密 $s_1, s_2, \cdots, s_k \in Z_Q$，秘密分发者在进行秘密分发时，只需在秘密分发过程的第(1)步为每个秘密 $s_i (i=1,2,\cdots,k)$ 随机选取一个唯一的整数 r_i。由 LUC 密码体制的安全性可知，在秘密的分发和重构过程中，每个参与者 P_i 的私钥不会被其它参与者或系统外的任何人计算出来。而且，即使知道某个参与者 P_i 关于若干个秘密的伪份额，也不可能计算出他的关于其它秘密的伪份额。这也是本文方案的一个优点，即秘密分发者可以动态地在 n 个参与者中共享任意秘密。

(6)值得注意的是，尽管 LUC 和 RSA[9]通常是可替换的[6]，但是在本文的方案中不可以使用 RSA 来替换 LUC，这种替换会导致方案安全性的降低。这是因为在 RSA 体制中，数字签字的乘积是相应消息之积的数字签字，这使得 RSA 会受到公共模以及称之为自适应选择消息伪造的密码攻击[6]；而 LUC 不具有这样的乘积性质，因而不受这些攻击[6]。这也从另一个侧面显示了本文方案的安全性。

(7)最后，需要讨论一下方案的性能。在 Shamir 的门限方案[2]及其基于 Shamir 的方案中，如文[1,8]等，最耗时的操作为多项式插值计算。一般来说，这些方案的性能主要决定于多项式插值操作。多项式的次数越高，插值运算越复杂。现有基于 Shamir 的方案中，采用的 Lagrange 插值多项式大多是高次多项式。目前有效的多项式插值运算的算法复杂度也仅为 $O(n\log^2 n)$[1]，因而这些方案效率不是很高。而在本文方案中，所采用的 Lagrange 插值多项式仅为一次多项式，多项式的构造仅需要 3 次乘(除)法和 3 次加(减)法，见式(4)。可见，本文方案比起其它基于 Shamir 的方案更为有效，而且容易实现。

结论 提出了一个基于 LUC 的一般访问结构上的秘密共享方案。参与者的秘密份额为他们的私钥，即使秘密分发者也不能获得每个参与者的秘密份额；秘密分发者和各参与者间所有的消息都可以以明文形式发送，不需要在他们之间维护一条安全信道，这对于秘密分发者与参与者之间不存在安全通信信道时的场合尤为有用；秘密重构过程中，任何人可以立即检验每个合作的参与者是否进行了欺骗。利用该方案可以共享任意多个秘密，而不必修改参与者的秘密份额。只要 LUC 算法和 Shamir 的门限方案具有足够强的安全性，本文方案就具有很强的防欺诈、防攻击能力。

参 考 文 献

1 Hwang R J, Chang C C. An on-line secret sharing scheme for multi-secrets[J]. Computer Communications, 1998, 21(13): 1170～1176

2 Shamir A. How to share a secret [J]. Communications of the ACM, 1979, 22(11): 612～613

3 Blakley G. Safeguarding cryptographic keys [A]. In: Proc. AFIPS 1979 Natl Conf [C], New York: AFIPS press, 1979. 313～317

4 Benaloh J, Leichter J. Ceneralized secret sharing and monotone functions [A]. In: Advances in Cryptology-Crypto'88 [C], Berlin: Springer-Verlag, 1990. 27～35

5 Wang S J. Direct construction of a secret in generalized group-oriented cryptography [J]. Computer Standards & Interface, 2004, 26(5): 455～460

6 Smith P. LUC public-key encryption: A secure alternative to RSA [J]. Dr. Dobb's Journal, 1993, 18(1): 44～49

7 王育民，刘建伟. 通信网的安全——理论与技术[M]. 西安：西安电子科技大学出版社，1999

8 Yang Chou-Chen, Chang Ting-Yi, Hwang Min-Shiang. A (t, n) multi- secret sharing scheme [J]. Applied Mathematics and Computation, 2004, 151(2): 483～490

9 Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public key cryptosystem [J]. Communication of ACM, 1978, 21(2): 120～126

(上接第 97 页)

初始化串口，本系统选用的是 UART0，以便在系统启动时可通过超级终端查看由串口打印输出的启动信息，方便检查运行情况和调试，

```
#define ULCON0 ( * (volatile unsigned * ) 0x3FFD000)
#define UCON0 ( * (volatile unsigned * ) 0x3FFD004)
#define UBRDIV0 ( * (volatile unsigned * ) 0x3FFD014)
void InitUART(int Baudrate)
{
ULCON0 = 0x03;//8 位数据，1 位停止，无校验
UCON0 = 0x09;//当有数据读/写 UART
             //读/写 Buffer 时产生中断请求
UBRDIV0 = Baudrate; //通过函数参数设
                    //置波特率
}
```

程序跳转，控制权交给 uCLinux 操作系统。在本系统中，将 uCLinux 内核烧写至 Flash 起始地址为 0x8000 的存储空间中。通过 Flash 拷贝，两次内存映射，内核起始地址被映射到 0x8000 的 SDRAM 中，因此代码为

```
LDR  R0,=0x8000
MOV  PC,R0
```

结束语 本系统的基本功能已设计完成，并在实验室内部测试成功，可替代原服务器充当网关。下一步的设计内容主要是为系统增加 IDE 硬盘接口使其可作为 ftp 服务器，并在网络部分加入防火墙、流量控制等扩展功能。

参 考 文 献

1 李驹光. ARM 应用系统开发详解. 清华大学出版社，2004

2 庞继勇，李维英，王竞. 网络通信处理器 S3C4510B 的网口驱动设计. 单片机与嵌入式系统应用，2004

3 Samsung 公司. S3C4510B 芯片手册