一个安全的基于身份的密钥分发解决方案①

A Secure Identity-Based Key Issuing Solution

邓　光　(中国科学院研究生院 北京 100039)

鲁士文　(中国科学院 计算技术研究所 北京 100080)

摘　要： 本文针对基于身份密码体制中存在的密钥托管问题,提出一个安全的基于身份的密钥分发协议。首先介绍 Lee.B 提出的一个解决方案,对该方案的利弊加以分析; 然后在此基础上提出改进协议。在该协议中,用户的私钥由多个可信实体共同参与并以串行方式生成。这个方案不但有效解决单个可信实体的密钥托管隐患,还减少了用户与可信实体间的交互次数,提高了系统效率。最后对 TC 的可信性进行了讨论。

关键词： 基于身份密码体制 密钥托管 网络安全

1　引言

在基于身份的公钥密码体制中,通常采用由用户用自己的身份信息和系统参数生成公钥,并提交身份信息给密钥生成中心(KGC),KGC 生成用户私钥的密钥生成方案。但这样的密钥生成方法存在一个本身固有的密钥托管 (KeyEscrow)问题,即用户必须无条件信任 KGC。由于 KGC 可以依据用户的公开身份信息计算出系统内任意一个用户的私钥,继而可以伪造用户的有效签名,或者解密用户的加密信息。第三方无法察觉到 KGC 是否有欺骗行为,一旦用户对自己身份对应的签名有争议,或者用户的加密信息被泄漏,系统没有有效的办法保证用户的合法权益。因此,密钥托管问题妨碍了基于身份密码体制的广泛应用。

近年来,几种 IBE(Identity-Based Encryption) 密钥托管问题的解决方案相继提出,基本的解决思路分为两类:一类是密钥的生成过程加入用户的保密信息,例如,Gentry[1] 提出的基于证书的加密方案,Al-Riyami 和 Paterson 提出的省略证书的公钥加密方案[2]等;另一类是密钥生成由多方参与,例如 Chen 等人提出的 n 个 KGC 分别生成用户部分私钥的密钥分发方案[3];LEE.B 提出的由 KGC 和多个 KPA 的共同参与的串行密钥分发方案[4]等。几种方案各有利弊,现有的加入用户保密信息的 IBE 方案中,加密方必须要验证接收方的公钥,因此失去了 IBE 的优势,本文着重探讨多方共同参与的 IBE 密钥生成方案。在 chen 的方案中[3],由于单个 KGC 生成的私钥信息没有加密,任何截获所有部分私钥的人都能够算出用户的私钥。LEE.B 的方案[4]克服了 chen 方案的缺陷,但是也存在漏洞[5]。本文在 LEE.B 方案的基础上进行了改进,针对密钥托管问题,提出一个新的高效安全的基于身份密钥分发方案。

1　Lee.B的解决方案

在文献[4]中,Lee.B 提出一个解决方案。在该方案中,设置多个 KPA(Key Privacy Authority),用户的私钥是在 KGC 和所有 KPA 的共同参与下生成,方案描述如下:

1.1 系统参数设置

KGC 执行如下步骤产生系统参数:

步骤 1. KGC 定义两个 q 阶子群 G_1 和 G_2,:$(G_1,+)$和$(G_2,*)$;一个双线性映射 e: $G_1XG_1\rightarrow G_2$。

步骤 2. KGC 随机选取 $S_0\in Z_q^*$作为自己的主密钥,并公开 $P_0=s_0 P$,其中 $P_0\in G_1$ 为生成元。

步骤 3. KGC 定义 hash 函数 H_1:{0,1}*→G1,H_2: G_2: →{0,1}l,l 为明文长度,H_3:$G_2\rightarrow Z_q^*$。KGC 的公开参数是(G_1,G_2,e,H_1,H_2),明文空间是 M={0,1}l,主密钥为 $S_0\in Z_q^*$。系统公钥生成过程如下:

每一个 KPA_i (i=1,2,3,…,n)分别随机选择 $Si\in Z_q^*$(模为 q 的有限域)作为各自的子密钥,对应的公钥

① 收稿时间:2008-09-23

为 Pi=si P。系统公钥 Y=$s_0s_1s_2\cdots s_n$ P。令 Y_0’ = P。, Y_i’ =s_i Y_i −1’,n 个 KPA 依次计算 Y_1’,Y_2’,…,Yn’, 则 Y=Yn’。

1.2 用户密钥生成阶段

设有用户 Alice,字符串 ID 表明 Alice 的身份。用户 Alice 随机选取私钥 x,并计算盲因子 X=xP。Alice 将 X 和 ID 传给 KGC, 提出生成密钥请求。KGC 通过以下步骤生成用户 Alice 的部分盲私钥:

(1)KGC 验证用户身份。

(2)KGC 计算用户公钥 $Q_{ID}=H_I$(ID, KGC, KPA_1, …,KPA_n)。

(3)KGC 计算用户的部分盲私钥 Q_0’ =H_3 ($e(s_0X,P_0)$)s_0Q_{ID} 。

(4)KGC 计算对 Q_0’的签名 $Sig_0(Q_0')= s_0Q_0$’。

(5)KGC 将 Q_0’和 $Sig_0(Q_0')$传给用户 Alice。Alice 依次将(ID,X,Q_{i-1}’, $Sig_{i-1}(Q_{i-1}')$)传给 KPA_i(i==1,2,3,…n),KPA_i(i==1,2,3,…n)执行以下步骤:

(1)检查等式 $e(Sig_{i-1}(Q_{i-1}'),P) = e (Q_{i-1}',P_{i-1})$ 是否成立。

(2)计算 Qi’=H3(e(siX,Pi))siQi−1’, Sigi (Qi’) =siQi’。

(3)将 Qi’和 Sigi(Qi’)返回给用户 Alice。

最后用户 Alice 得到 $Q_n' = H_3(e(s_nX, P_n))s_nQ_{n-1}$’。用户 Alice 计算自己的私钥 $D_{ID}=Q_n'$ /$H_2(e(P_0,P_0)x)\cdots H_2(e(P_n,P_n)x)= s_0s_1,\cdots,s_n\ Q_{ID}$。用户可以检查等式 $e(D_{ID},P) = e(Q_{ID},Y)$是否成立以验证私钥 D_{ID} 的正确性。

1.3 用户密钥应用

产生的用户公私钥对适用于任何建立在双线性配对上的基于身份密码体制,如加密方案和签名方案等。比如在 Bonen-Franklin 的加密方案[6]中,对于明文 m, 加密者计算密文 C=(U,V)=(rP,m ⊕ $H_2(e(Q_{ID},Y)')$) , 解密者 Alice 计算 V ⊕ $H_2(e(D_{ID},U))$=m,恢复明文。

1.4 Lee3 方案分析

相对于 Chen 等人提出的多个 KGC 分别独立生成用户部分私钥的密钥分发方案[3],Lee.B 的方案可以防止攻击者分别攻击每一个 KGC 和 KPA_i(i=l,2,3,…,n):由于含有用户的盲因子 X,即使攻击者截获所有的 Q_i’(1=0,1,2,…,n),也不能够计算出用户的私钥。但是在 LEE.B 的方案中,存在着以下漏洞和不足[5]:

(1)KGC 可以随机选取 x_I,用 $X_I=x_IP$ 冒充用户 Alice 的盲因子,然后计算 $Q_o'' = H_3(e(s_0X_1,P_0))s_0Q_{ID}$。与 KGC 合作的攻击者 (字符串 ID_1 表示其身份),可以用(ID_1,X_I,Q_0’’, $Sig_0(Q_0'')$)向 KPA_i(i=1,2,3,…,n)提出计算私钥请求。即使 KPA_i 验证用户身份,由于无法验证 Q_0’和用户 ID 的对应关系,KPA_i 也还是无法分辨出假冒者。KGC 和合作攻击者可以联合得到 Q_n’,计算出 Alice 的私钥 $D_{ID}= Q_n'$ /$H_2(e(P_0,P_0))\cdots H_2(e(P_n,P_n))= s_0s_1,\cdots,s_n\ Q_{ID}$。

(2)用户需要向所有的 KPA_i(i=1,2,3,…,n)证明自己的身份,这在实际应用中会大大增加系统的负担。

LEE.B 方案并没有真正的解决密钥托管问题。针对以上漏洞,本文提出改进方案。

2 改进的密钥分发方案

2.1 基本思想

根据以上对 Lee.B 方案的分析,本文首先在改进方案中加入了 KPA_i(i=1,2,3,…,n)验证传输的部分私钥信息和用户 ID 对应关系的方法。为了减少用户和 n 个 KPA_i 之间身份认证的次数,本文设定用户的私钥生成参数(ID,X,Q_{i-1}’, $Sig_{i-1}(Q_{i-1}')$)直接在 KGC 和 KPA_i(i=1,2,3,…,n)之间依次传递,而不经由用户 Alice。这样的话,则对于 KPA_i(i=1,2,3,…,n)来说,KGC 和所有的 KPA_j(j<i)有可能会联合生成伪造的用户盲因子 X’。这是因为用户 Alice 的 ID 是公开参数,KPA_i 即使能够验证 X 与用户 ID 的对应关系,也还是无法证明 X 就是合法用户 Alice 生成的。所以 Alice 需要向所有的 KPA_i(i=1,2,3,…,n)证明自己的身份,并将盲因子 X 传递给 KPA_i。

为了解决以上问题,本文在系统中加入了一个可信任的认证机构 TC(Trust Center),用户 Alice 可以将盲因子 X 通过安全信道传递给 TC,TC 对 X 加上自己的签名,然后将某个时间段内所有申请生成私钥用户的盲因子一次性传给 KPA_i(i= 1,2,3,…,n),这将大大减少系统中用户身份认证的次数。本文的密钥分发方案描述如下:

2.2 系统设置

KGC 执行如下步骤产生系统参数:

(1) KGC 定义两个 q 阶子群 G_1 和 G_2,:(G_l,+)和(G_2,*);一个双线性映射:e:　$G_1XG_1\rightarrow G_2$。

(2) KGC 随机选取 $S_0 \in Z_q^*$作为自己的主密钥,并公开 $P_0=s_0 P$,其中 $P \in G_1$ 为生成元。

(3) 定义 hash 函数 H_l:{0,l}*→G_1,H_2: G_2:→{0,l}l,l 为明文长度,H_3:G_2→Z_q*。

KGC 的公开参数是(G_1,G_2 ,e,H_1,H_2),明文空间是 M={0,l}l,主密钥为 $s_0 \in Z_q^*$。系统公钥生成过程如下:

每一个 KPA_i (i=l,2,3,…,n)分别随机选择 $s_i \in Z_q^*$作为各自的子密钥,对应的公钥为 $P_i=s_i P$。系统公钥 $Y=s_0s_1s_2\cdots s_n P= s_1s_2\cdots s_n P_n$。令 $Y_0' = P_0$。, $Y_i' =s_i * Y_{i-1}'$,n 个 KPA 依次计算 Y_l',Y_2' ,…,Yn',则 Y=Yn'。

TC 随机选取 $s_{TC} \in Z_q^*$作为自己的主密钥,并公开 $P_{TC}=s_{TC}P$。

2.3 用户密钥生成阶段

设有用户 Alice,字符串 ID 表明 Alice 的身份。Alice 随机选取私密信息 x,并计算盲因子 X=xP。Alice 向 TC 认证身份并将 ID 和 X 传给 TC。TC 生成签名 $Sig_{TC}(Alice)=s_{TC}H1(ID||X)$。

Alice 将 X 和 ID 传给 KGC,提出生成密钥请求。KGC 通

过以下步骤生成 Alice 的部分盲私钥:

(1)KGC 验证用户身份。

(2)KGC 计算用户公钥 $Q_{ID}=H_l(ID, KGC,KPA_1, \cdots,KPA_n)$。

(3)KGC 计算用户的部分盲私钥 $Q_0' =H_3(e(s_0X,P_0))s_0Q_{ID}$ 。

(4)KGC 计算对 Q_0' 的签名 $Sig_0(Q_0') = s_0Q_0'$。

(5)KGC 将(ID,X,Q_0', $Sig_0(Q_0')$)传给 KPA_1。

KPA_i(i==1,2,3,…n) 依次执行以下步骤:

(1) 验证等式 $e(Sig_{i-1}(Q_{i-1}'),P) = e(Q_{i-1}',P_{i-1})$ 是否成立。

(2) 从 TC 得到 $Sig_{TC}(Alice)$, 验证等式 $e(Sig_{TC}(Alice),p) =e(H_l(ID||X),P_{TC})$是否成立。

(3)计算 $Q_i' =H_3(e(s_iX,P_i))s_iQ_{i-l}'$, $Sig_i(Q_i') = s_iQ_i'$。

(4)当 i=n 时,将 Q_i'和 $Sig_i(Q_i')$返回给用户 Alice,否则将(ID,X,Q_i' , $Sig_i(Q_i')$)传给 KPA_i+1;。

最后 Alice 得到 $Q_n' = H_3(e(s_nX,P_n))snQ_{n-1}'$。Alice 计算自己的私钥 $D_{ID}= Q_n' /H_2(e(P_0,P_0)x)\cdots H_2(e(P_n,P_n)x)= s_0s_1\cdots s_n Q_{ID}$。Alice 可以检查等式 $e(D_{ID},P)=e(Q_{ID},Y)$是否成立以验证私钥 D_{ID} 的正确性。

产生的用户公私钥对适用于大多数建立在双线性配对上的基于身份密码体制,如加密方案和签名方案等。本文把产生的用户公私钥对应用于 Boneh — Franklin 的基于身份加密方案[6],得到的加密和解密过程如下:

2.4 加密

对于明文 m 和给定的目标 ID,加密者执行以下步骤:

(1)计算 $Q_{ID}=H_l(ID,KGC,KPA_1,\cdots,KPA_n)$

(2)随机选择 $r \in Z_q^*$

(3)计算密文 $C=(U,V)= (rP,m \oplus H_2(e(D_{ID},Y)'))$,并传给 ID 对应的用户 Alice。

2.5 解密

接收到密文 C=(U,V)后,用户 Alice 用自己的私钥 D_{ID} 计算 $V \oplus H_2(e(D_{ID},U))=m$,恢复出明文。

本文把 2.3 产生的用户公私钥对应用于 Cha-Cheon 的基于身份签名方案[7],得到的签名和验证过程如下:

2.6 签名

对于明文 m,用户 Alice 用其私钥 D_{ID} 产生签名的步骤如下:

(1)随机选择 $r \in Z_q^*$。

(2)计算 $u=rQ_{ID}$,$h=H_3(m,U)$,$V=(r+h)D_{ID}$。此处 H_3 定义为 H_3:{0,l}* × G_l→Z_q*。

(3)用户 Alice 产生的签名为 σ =(U,V)。

2.7 验证

接收到签名 σ =(U,V)后,验证者执行以下步骤:

(1)计算 $Q_{ID}=H_l((ID,KGC,KPA_1,\cdots,KPA_n)$。

(2)验证等式 $\hat{e}(P,V)= \hat{e}(Y,U+hQ_{ID})$是否成立,等式成立,则接受签名,否则拒绝签名。

2.8 方案分析

(1)与 Lee.B 的方案类似,由于用户的私钥由 KGC 和 n 个 KPA 计算生成,因此单个可信实体的密钥托管问题得以解决。只有知道用户选择的秘密信息的合法

用户,才能解析出在生成过程中盲化的用户私钥。

(2)本方案比 Lee 的方案改进之处在于:对于 $KPA_i(i==1,2,3,\cdots n)$来说,本方案能够抵御 KGC 和所 $KPA_j(j<i)$的合谋攻击。假设用户 Alice 向 KGC 提出生成私钥请求。KGC 随机选择 x',生成伪造的用户盲因子 X'=x'P。KGC 和所有的 $KPA_j(j<i)$合谋,KPA_i-1 将$(ID,X',Q_{i-1}',Sig_{i-1}(Q_{i-1}'))$传给 KPA_i,KPA_i 从 TC 得 $Sig_{TC}(ID)$,验证 $e(Sig_{TC}(ID),P)=e(H_1(ID||X),P)\neq e(H_1(ID||X'),P_{TC})$从而会发现异常。

(3)另外,从效率角度考虑,TC 的加入可以减少用户和 $KPA_i(i==1,2,3,\cdots n)$之间的身份认证次数。假设在某个时间段 Tl 内有 m 个用户申请生成私钥,如果没有 TC,则所需的用户身份认证次数为 n*m 次;加入 TC 后,则省去了用户和 $KPA_i(i==1,2,3,\cdots n)$之间的身份认证,TC 会将时间段 Tl 内申请私钥的 n 个用户的盲因子加上自己的签名,一次性传给 KPA_i。这个过程所需的身份认证次数为 n+2m 次。其中 m 为用户向 TC 认证的次数,n 为 $KPA_i(i==1,2,3,\cdots n)$向 TC 认证次数。m 为用户向 KGC 认证的次数。

3 小结

密钥托管最难解决的问题是如何有效地阻止用户的欺诈行为,即逃脱托管机构的跟踪。这也是密钥托管的目的。但作为用户来讲,KGC 是否绝对可靠是基于身份的公钥密码体制的一个关键问题。

本协议在实现时,用户必须向 TC 认证身份并将用户自己的 ID 和盲因子 X 传给 TC,TC 根据这些信息生成相应的签名。与此同时,用户也将 X 和 ID 传给 KGC,提出生成密钥请求。以后 KPA 验证来自于 TC 的签名后才能产生的正确的用户公私钥。从这一过程可以看出,由于加入了可信任的认证机构 TC(Trust Center),本协议解决了 KGC 和所有的 KPA 联合作弊的隐患。TC 的可信性是可以保证的,因为在上面的过程中,可以知道在 TC 中生成的签名必须要 KPA 的验证,而密钥是在 KGC 中生成。任何 CA(Certificate Authority)在验证托管证书的真实性时,都不需要 TC 的介入。当然,如果能够实现法律职能部门的介入,既可以在必要时帮助国家司法和安全等部门获取原始明文信息,又可以在用户丢失,损坏自己的密钥后恢复密文,同时也能加强 TC 的可信性。由此可见本方案的确能够抵御 KGC 和多个 KPA 的合谋攻击,同时通过引入可信机构 TC,减少了方案中用户身份认证的次数,提高了系统效率。

参考文献

1 Gentry C. Certificate-based Encryption and the Certificate Revocation problem. Proceedings of EUROCRPYT 2003, LNCS 2656, Springer-Verlag: 272 – 293.

2 Al-Piyami SS, Peterson KG. Certificateless public key cryptography. Proceedings of Asiacrypt LNCS3-43, Springer-Verlag, 2003:452 – 474.

3 Chen L, Harrison K, Soldera D, Smart N P. Applications of Multiple Trust Authorities in Pairing Based Crypto systems. Proceedings of the International Conference on Infrastructure Security, 2002:260 – 275.

4 Lee B, Boyd C, Dawson E, Kim K, Yang J, Yoo S. Secure key issuing in ID-based cryptography. Proceedings of the second workshop on Australasian Information Security, Data Mining and Web Intelligence, and Software Internationalisation, 2004:69 – 74.

5 Xu C, Zhou J, Qin Z. A note on secure key issuing in ID-based cryptography. http://eprint.iacr.org/2005/180.

6 Boneh D, Franklin M. Identity-based encryption from the Weil Pairing.Advances in Cryptology-Crypto. 01. -LNCS 2139. J Kilian ed, Berlin Springer- Verlag. 2001: 213 – 239.

7 Cha J, Cheon J. An identity-based signature from Diffie-Hellman groups. Public Key Cryptography Proceedings of PKC2003, LNCS2567, Berlin: SPringer-Verlag, 2003:18 – 30.