第37卷 第17期
Vol.37 No.17

计 算 机 工 程
Computer Engineering

2011年9月
September 2011

• 安全技术 •

文章编号：1000—3428(2011)17—0124—02 文献标识码：A 中图分类号：TN918.1

一种前向安全的定向代理签名方案

周孟创，余昭平

(解放军信息工程大学电子技术学院，郑州 450004)

摘 要：基于前向安全性和有限乘法群上离散对数的难解性，提出一种新的代理签名方案。该方案不仅具有前向安全性、不可伪造性、不可否认性、匿名性，而且还具有定向性，只有指定的验证人可以验证代理签名是否有效，并分离实际签名权和代理签名权。分析结果表明，该方案的安全性较高。

关键词：前向安全；不可伪造；不可否认；定向代理签名；匿名性

Forward-secure Designated-receiver Proxy Signature Scheme

ZHOU Meng-chuang, YU Zhao-ping

(Institute of Electronic Technology, PLA Information Engineering University, Zhengzhou 450004, China)

【Abstract】Based on forward-security and the difficulty of solving discrete logarithm problems on the finite multiplicative group over the finite fields, a new proxy signature scheme is proposed. The scheme has the following characteristics: forward-security, unforgeability, undeniability, anonymous, etc. Besides the preceding characteristics, only the designated-receiver can validate the proxy signature. The new designed arithmetic successfully separates the power of proxy signature and the original signature. It analyzes the security of the scheme, and conclusion shows that the new designed scheme is more security.

【Key words】forward-security; unforgeability; undeniability; designated-receiver proxy signature; anonymous

DOI: 10.3969/j.issn.1000-3428.2011.17.041

1 概述

自文献[1]提出代理数字签名的概念后，因其在诸多方面应用前景十分广泛，研究者先后提出了许多代理数字签名方案[2-5]。文献[4]给出的代理签名方案不具有前向安全性和定向性。而文献[5]所述方案则不具有匿名性，容易泄露代理签名人的隐私。为此，本文在上述文献的基础上，提出一种前向安全的定向代理签名方案。

2 前向安全的定向代理签名方案

2.1 系统初始化

系统初始化过程如下：首先选取安全参数，选取大素数 p，$g \in GF(P)$ 为本原元，阶为 q，再设 $Hash()$ 为一个安全的单向函数。Alice、Bob、Client、CA 分别为设定的 4 个实体：原始签名人，代理签名人，指定验证人，可信任的第三方认证中心。Alice、Bob、Client 的公私钥对分别为 (x_A, y_A)、(x_B, y_B)、(x_C, y_C)。x_A、x_B 和 x_C 为各自私钥，并计算 $y_A = g^{x_A} \bmod p$、$y_B = g^{x_B} \bmod p$、$y_C = g^{x_C} \bmod p$ 为公钥，y_A、y_B、y_C 置于可信任的第三方认证中心 CA 上，并与 p、q、g、$Hash()$ 一起公开。

前向安全性理论是指代理签名方案具有时效性，具体说明可参考文献[6]。

2.2 代理授权

代理授权过程如下：

(1)Alice 想授权 Bob 自己的签名权，即 Alice 欲使 Bob 成为自己的代理签名人，同时指定签名验证人 Client。首先 Alice 生成授权书 m_w，此授权书 m_w 的内容应包括原始签名者即 Alice 本人的身份 ID_A、代理签名者即 Bob 的身份 ID_B、共同所信任的第三方 CA、指定的验证人 Client 的身份、所授权限范围、代理时间周期 1,2,…,T、代理终止时间 t、所授代理签名者的最大签名次数 k 等内容。

(2)Alice 用自己的私钥 x_A 对授权书 m_w 进行签署得到 n_w，然后将 n_w 发送给可信的第三方 CA。CA 认证了 n_w 的合法性后，根据所授代理签名者的最大签名次数 k，在系统中设定 Bob 能代表 Alice 所能行使的最大的签名次数。代理签名者 Bob 发送自己的身份信息 ID_B 给原始签名者 Alice。

(3)Alice 选取一个随机数 $\lambda \in (1, p-1)$ 作为临时密钥，计算 $y_\lambda = g^\lambda \bmod p$，$ID = Hash(\lambda, ID_B)$，$f_1 = Hash(n_w, y_\lambda, ID, y_C)$，$\delta = x_A f_1 + \lambda \cdot y_\lambda \bmod q$。然后 Alice 将 $(n_w, \delta, y_\lambda, ID)$ 发送给 Bob。

2.3 代理验证

Bob 在收到 $(n_w, \delta, y_\lambda, ID)$ 后，可通过以下步骤验证其合法性，并决定出是否接受 Alice 的委托：

(1)Bob 验证 n_w 是否合法有效，若有效，则可以根据其内容，决定是否接受 Alice 的委托请求，否则直接拒绝或者要求 Alice 重发。

(2)Bob 利用收到的 $(n_w, \delta, y_\lambda, ID)$ 以及 Client 的公钥 y_C 计算 $f_1 = Hash(n_w, y_\lambda, ID, y_C)$。然后 Bob 验证 $g^\delta = y_A^{f_1} \cdot y_\lambda^{y_\lambda} \bmod p$ 是否成立。若成立，则接受 Alice 的委托；否则要求 Alice 重新发送或直接拒绝。证明：左边=$g^\delta = g^{x_A f_1 + \lambda \cdot y_\lambda} = y_A^{f_1} \cdot (g^\lambda)^{y_\lambda}$ = 右边，得证。

(3)Bob 再计算 $\delta_0 = x_B + \delta \bmod p$，则为 Bob 的代签私钥。

作者简介：周孟创(1983－)，男，硕士研究生，主研方向：密码理论，安全协议分析；余昭平，教授

收稿日期：2011-02-11 **E-mail：**bearzmc@163.com

2.4　代理签名密钥更新

验证通过后 Bob 利用下述方法计算出自己更新后的代签私钥：

(1)假设此时签名进入第 i 个阶段，则 Bob 利用单向函数 f，使用先前密钥 $x_{B_{i-1}}$ 计算当前密钥 x_{B_i}，$x_{B_i}=(x_{B_{i-1}})^2 \bmod p$，然后立即删除密钥 $x_{B_{i-1}}$。

(2)Bob 选取随机数 $\theta_i \in (1,p-1)$，然后计算：$U_B=g^{\theta_i} \bmod p$；$V_i=x_{B_i}g^{\theta_i} \bmod p$；$f_2=Hash(i,V_i,n_w,y_\lambda,y_C)$；$x_i=\delta_0+\theta_i \cdot f_2 \bmod p$，则 x_i 为 Bob 在第 i 时刻的代理签名私钥。

至此，Alice 完成了对 Bob 授权代理签名的过程。

比较文献[6]给出的代理签名密钥进化过程，本文方案在更新密钥时，因为随机数 θ_i 的选取以及中间量 f_2 的引入，所以密钥的时效性更强、更为安全。

2.5　代理签名生成

Bob 利用上面得到的更新后的签名私钥 x_i 生成对消息 m 的代理签名，当然这个过程需要可信任的第三方 CA 的参与。

(1)Bob 利用 Alice 加密后的授权证书 n_w、Alice 的临时公钥 y_λ、验证者 Client 的公钥 y_C，Bob 的代理私钥 x_i，首先按普通数字签名的方法，得到一个普通的数字签名 $S_0=Sign(\delta_0,n_w,y_\lambda,y_C,m)$。Bob 用私钥 x_B 签署本身所生成的请求认证文书 Q_B，其内容包括 Bob 的身份 ID、数字签名 S_0、请求 CA 对 S_0 认证的消息，然后将 Q_B 发送给 CA。

(2)CA 先检测由 Bob 提供的 Q_B 的真实性与完整性，若有不符，则拒绝提供服务。否则，在系统内部搜索 Bob 可以代签的最大次数，以及 Bob 可以代理 Alice 签名的最大权限，如果都在权限范围内，则生成一个时间戳证书 T_i 发送给 Bob，同时在系统内改变相应的参数，为下次提供认证服务做好准备。

(3)Bob 收到 T_i 后，通过如下计算得到最终的代理签名：

Bob 选择 2 个随机数 $\omega_1,\omega_2 \in (1,p-1)$；计算：$M_1=(g^{\omega_1-\omega_2})^{2^{T+1-i}} \bmod p$；$M_2=(Y_C^{\omega_1})^{2^{T+1-i}} \bmod p$；$f_3=Hash(T_i,\theta_i,M_1,M_2,S_0)$；$Sign=\omega_2-x_i f_3 \bmod p$，则 $Q=(m,y_\lambda,n_w,y_C,M_1,M_2,sign)$ 即是 Bob 对 Alice 的最终代理签名。

2.6　代理签名验证

当验证者 Client 收到 Bob 对 Alice 的最终代理签名 Q，并欲验证其合法性时，需要执行以下步骤：

(1)Client 从所收到的电文 Q 中，分析出 m，代理签名 $Sign$，授权证书 n_w。

(2)验证者 Client 用 Alice 的公钥及 Bob 的公钥来验证 n_w 的合法性以及 Bob 行使代理签名权的次数是否已经到达最大值，若验证失败，则说明 Q 有效。

(3)使用私钥 y_C 来验证等式：

$$[M_1 g^{sign}(U_B^{f_2} \times y_B \times y_A^{f_1} \times y_\lambda^{y_\lambda})^{f_3}]^{x_C}=M_2 \bmod p$$

是否成立，若成立，则说明代理签名合法，否则，认为此签名非法。

3　安全性分析

3.1　前向安全性

假如攻击者已经获得了第 i 时刻代理签名人 Bob 的密钥 x_{B_i}，此时他想得到第 $j(j>i)$ 时刻的密钥，则得通过下式求 x_{B_j}：$x_{B_i}=x_{B_{i-1}}{}^2=\cdots=x_{B_j}{}^{2^{i-j}} \bmod p$，而这是一个强 RSA 假设问题，在计算上是不可行的，即无法通过 x_{B_i} 求出 x_{B_j}。并且当 x_{B_i} 泄露后，即刻宣布此时公钥私钥作废，同时启用下一时刻公钥和私钥。所以，攻击者获得的密钥不会对整个系统有任何威胁，即本协议具有前向安全性。

3.2　定向性

从等式 $[M_1 g^{sign}(U_B^{f_2} \cdot y_B \cdot y_A^{f_1} \cdot y_\lambda^{y_\lambda})^{f_3}]^{x_C}=M_2 \bmod p$ 可以看出，因为在代理签名验证阶段，要用到验证者 Client 的私钥 x_C，而其他任何人欲从 $Q=(m,y_\lambda,n_w,y_C,M_1,M_2,sign)$ 中得出 x_C，面临求解离散对数问题，这在计算上是不可行的，所以除了原始签名人指定的验证人以外，其他任何人都无法验证本代理签名。

3.3　原始签名、代理签名的不可伪造性及可区分性

通过 2.3 节步骤(2)中的证明等式可以得出：代理签名者 Bob 根据收到的委托授权请求 n_w 及 Alice 的公钥 y_A，要计算出 Alice 的私钥 x_A，这相当于求解离散对数问题，所以，Bob 无法伪造原始签名者 Alice 的签名，并且其他任何人也无法伪造出 Alice 的签名。

因为只有 Bob 知道代理签名私钥 x_i，其他任何人(包括 Alice 在内)想要伪造代理签名，相当于求解离散对数问题、大数分解问题或单向散列函数问题，假使 Alice 委托了 n 个代理签名者 $B_1,B_2,\cdots,B_n$，而每一个代理签名者在计算代理签名密钥时，都要用到私钥 x_{B_i}，所以任何人(包括 Alice 在内)都无法伪造 Bob 的代理签名。

显然，从代理签名 $Q=(m,y_\lambda,n_w,y_C,M_1,M_2,sign)$ 的格式上可以看出，Q 中明显用到了授权书 n_w，Alice 的临时公钥，Client 的临时公钥等，所以，代理签名和原始签名是可区分的。

3.4　有效性

验证代理签名有效性过程如下：

证明：由 $Sign=\omega_2-x_i f_3 \bmod p$ 得：

$$g^{Sign}g^{x_i f_3}=g^{\omega_2} \bmod p$$

$$\Rightarrow g^{Sign}g^{(x_B+x_A f_1+\lambda \cdot y_\lambda+\theta_i \cdot f_2)\, f_3}=g^{\omega_2} \bmod p$$

$$\Rightarrow [g^{sign}(U_B^{f_2} \cdot y_B \cdot y_A^{f_1} \cdot y_\lambda^{y_\lambda})^{f_3}]^{2^{T+1-i}}=(g^{\omega_2})^{2^{T+1-i}} \bmod p$$

$$\Rightarrow (M_1[g^{sign}(U_B^{f_2} \cdot y_B \cdot y_A^{f_1} \cdot y_\lambda^{y_\lambda})^{f_3}]^{2^{T+1-i}})^{x_C}=$$
$$[(g^{\omega_1-\omega_2})^{2^{T+1-i}}(g^{\omega_2})^{2^{T+1-i}}]^{x_C} \bmod p$$

$$\Rightarrow (M_1[g^{sign}(U_B^{f_2} \cdot y_B \cdot y_A^{f_1} \cdot y_\lambda^{y_\lambda})^{f_3}]^{2^{T+1-i}})^{x_C}=$$
$$(y_C^{\omega_1})^{2^{T+1-i}}=M_2 \bmod p$$

通过对上述等式的证明过程可以看出，本代理签名是有效的。与文献[7]中相比，在验证签名有效性时，引入了时间变量 i，这使代理签名具有前向安全性，在实际应用中也更加广泛。

3.5　匿名性及可追踪性

指定验证人 Client 在验证签名 $Q=(m,y_\lambda,n_w,y_C,M_1,M_2,sign)$ 时，要用到代理签名人 Bob 以 ID 的身份公开的临时公钥 U_B，因此，验证人 Client 可通过等式：

$$[M_1 g^{sign}(U_B^{f_2} \cdot y_B \cdot y_A^{f_1} \cdot y_\lambda^{y_\lambda})^{f_3}]^{x_C}=M_2 \bmod p$$

验证本代理签名是有效的，但无法确定是哪一个代理签名人所代签，即代理签名人的身份对 Bob 来说是匿名的。

可追踪性就是指当发生争议时，Client 可以发送 Bob 以 ID 的身份公开的临时公钥 U_B 给 Alice，Alice 则可以通过 Bob 所提交的别名身份 ID，然后利用保存的 λ，在数据库中查出 Bob 的真实身份 ID_B，从而解决争议，即具有可追踪性。

(下转第 145 页)