第34卷 第17期 计 算 机 工 程 2008年9月
Vol.34 No.17 Computer Engineering September 2008
·安全技术· 文章编号：1000—3428(2008)17—0162—02 文献标识码：A 中图分类号：TP309

一种具有强前向安全性的代理签名方案

杨 洁，钱海峰，李志斌
(华东师范大学计算机科学技术系，上海 200062)

摘 要：分析一种基于ElGamal的前向安全签名方案。该方案满足前向安全性，但在签名者私钥泄漏后，签名是不安全的，即不满足后向安全性，有一定的局限性。该文引入强前向安全的思想，克服了该方案的局限性，并将改进后的强前向安全签名与代理签名相结合，提出一种新的满足强前向安全定义的强前向安全代理签名方案。

关键词：前向安全；后向安全；强前向安全；代理签名

Strong Forward Security Proxy Signature Scheme

YANG Jie, QIAN Hai-feng, LI Zhi-bin
(Department of Computer Science, East China Normal University, Shanghai 200062)

【Abstract】 Forward security signature scheme satisfies forward security. However, the scheme is not secure if signature's secret key is revealed. For this reason, this paper proposes a new strong forward security proxy signature scheme combining strong forward security and proxy signature, which introduces strong forward concept and overcomes limitation of the previous scheme.

【Key words】 forward security; afterward security; strong forward security; proxy signature

1 概述

网络的迅猛发展促使数字签名技术在电子商务、电子选举等各个领域中得到广泛应用。但是数字签名的使用仍然存在很大的危险性，最大的挑战之一来自于私钥泄露带来的严重后果。基于这样的背景，1997年Anderson提出了前向安全的概念[1]。1999年Bellare和Miner在文献[2]中给出前向安全签名的正式定义。所谓前向安全签名是指将整个签名有效时间划分为T个时段，公钥在整个签名有效时间内保持不变，而私钥随着时段的推进不断更新，在每个签名时段使用当前时段的私钥产生签名。即使签名者某一时段的私钥泄漏，攻击者也无法伪造在该时段之前的有效签名，因此，在该时段之前的签名仍是有效的。前向安全的这一特点减少了由于签名私钥泄漏带来的严重后果。

基于前向安全思想，夏峰等人提出了一种基于ElGamal体制的前向安全数字签名方案[3]。本文对该方案的安全性进行了分析，发现攻击者如果得到签名者某一签名时段的签名私钥，就可以伪造本阶段及之后所有阶段的签名，这也就是前向签名中的后向安全问题。因而该方案不满足由Mike Burmeste等人提出强前向安全的定义[4]：一个签名体制无论在私钥泄漏前还是在私钥泄漏后都是安全的，则该签名体制是强前向安全的。1996年Mambo等人首次提出代理签名的概念[5]。所谓代理签名是指原始签名者将签名权委托给可靠的代理人，让代理人代表自己行使签名权力[5-6]。代理签名中也存在代理签名者私钥泄漏的问题。为克服以往的代理签名方案的不足，本文提出一种新的强前向安全代理签名方案。

2 夏方案及前向安全性分析[3]

2.1 初始参数

(1)选择一大素数 p 和随机数 sk_0，$1 < sk_0 \le p-1$，g 是 $GF(p)$ 的生成元。在此签名方案中，假设签名密钥的有效期分为 T 个时段。

(2)计算 $PK = g^{sk_0^{2^{T+1}}} / sk_0^{2^{T+1}} (\mathrm{mod}\ p)$，公开 p, g, T 和 PK。

2.2 私钥更新算法

若 $j = T+1$，则 sk_j 为空串。

若 $1 \le j < T+1$，则 $sk_{j+1} = sk_j^2 (\mathrm{mod}\ p-1)$。其中，$j$ 表示第 j 个时间段。

2.3 签名

(1)签名方选择随机数 k_1，k_2，计算 $r_1 = g^{k_1} (\mathrm{mod}\ p)$，$r_2 = sk_j g^{k_2} (\mathrm{mod}\ p)$。

(2)计算 $\delta = (h(m) - k_2 2^{T+1-j} r_1 - sk_j^{2^{T+1-j}} r_1) k_1^{-1} (\mathrm{mod}\ p-1)$。

(3)发送 (j, r_1, r_2, δ) 给签名接收者。

2.4 验证

如果 $PK^{r_1} r_1^{\delta} r_2^{2^{T+1-j} r_1} = g^{h(m)} (\mathrm{mod}\ p)$，则认为签名有效。因为：

$$PK^{r_1} r_1^{\delta} r_2^{2^{T+1-j} r_1} = g^{r_1 sk_0^{2^{T+1}}} / sk_0^{r_1 2^{T+1}} (sk_j g^{k_2})^{r_1 2^{T+1-j}} g^{k_1 \delta} (\mathrm{mod}\ p) =$$
$$g^{sk_0^{2^{T+1}} r_1 + k_2 r_1 2^{T+1-j} + h(m) - k_2 r_1 2^{T+1-j} - sk_j^{2^{T+1-j}} r_1} (\mathrm{mod}\ p) =$$
$$g^{h(m)} (\mathrm{mod}\ p)$$

2.5 安全性分析

(1)前向安全性：若 sk_j 被泄漏，令 $i < j$，根据文献[3]中对前向安全性的分析，攻击者无法伪造在私钥泄漏之前任何阶段的签名，所以该方案具有前向安全性。

(2)后向安全性：若 sk_j 被泄漏，令 $i \ge j$，攻击者可以利用 sk_j 来构造任何在本阶段及之后所有阶段的签名。因为 sk_j 被泄漏后，攻击者保持沉默，并和签名者进行同样的私钥更新，从而冒充签名者伪造签名。在计算签名 $\delta = (h(m) - k_2 2^{T+1-j} r_1 - sk_j^{2^{T+1-j}} r_1) k_1^{-1} (\mathrm{mod}\ p-1)$ 时，$sk_j^{2^{T+1-j}} = sk_i^{2^{T+1-i}} = sk_0^{2^{T+1}}$ 是一个与私钥

作者简介：杨 洁(1976－)，女，硕士研究生，主研方向：信息安全与密码学；钱海峰，副教授；李志斌，教授、博士生导师

收稿日期：2007-10-29 **E-mail：**51051201059@student.ecnu.edu.cn

进化无关的常数。攻击者随机选择 k_1'，k_2'，计算 $r_1'=g^{k_1'}(\bmod p)$，$r_2'=sk_ig^{k_2'}(\bmod p)$，$\delta'=(h(m)-k_2'2^{T+1-i}r_1'-sk_i^{2^{T+1-i}r_1'})k_1'^{-1}(\bmod p-1)$。然后将其伪造的代理签名 (i,r_1',r_2',δ') 发送给签名接收者。伪造签名能通过验证，因为：

$$PK^{r_1'}r_1'^{\delta'}r_2'^{2^{T+1-i}r_1'}=g^{r_1'sk_0^{2^{T+1}}}/sk_0^{r_1'2^{T+1}}(sk_ig^{k_2'})^{r_1'2^{T+1-i}}g^{k_1'\delta'}(\bmod p)=$$
$$g^{sk_0^{2^{T+1}}r_1'+k_2'r_1'2^{T+1-i}+h(m)-k_2'r_1'2^{T+1-i}-sk_i^{2^{T+1-i}r_1'}r_1'}(\bmod p)=$$
$$g^{h(m)}(\bmod p)$$

所以该方案不具有后向安全性。针对其存在的安全隐患，本文引入文献[4]的强前向安全思想，克服了夏方案的局限性。

3 新的强前向安全代理签名方案

3.1 系统建立

假设 A 授权 B 对消息 m 进行签名。B 先随机选择一个强安全素数 p，使得 $p=2p'q'+1$，其中，p'，q' 是 2 个大素数。选择 Z_p^* 的一个生成元 g，公布 p, g。A 随机选择一个整数 x_A，$1<x_A\le p-1$，计算 $y_A=g^{x_A}(\bmod p)$，A 得到公私钥对 (x_A,y_A)。在此签名方案中，签名密钥的有效期分为 T 个时段。B 随机选择一个整数 x_{B_0}，$1<x_{B_0}\le p-1$，计算 $y_B=x_{B_0}^{-2^{T+1}}(\bmod p)$。$B$ 得到公私钥对的一个初始值 (y_B,x_{B_0})。系统公钥为 (p,g,T,y_A,y_B)，$h:\{0,1\}^*\to Z_p^*$ 是一个密码学意义下的安全 Hash 函数。A 产生授权书 w，授权书包括 A 和 B 的身份信息、A 对 B 的代理签名授权、代理签名时限等。

3.2 代理授权

(1)A 随机选择 $k_A\in Z_{p-1}^*$，计算：

$r_A=g^{k_A}(\bmod p)$，$\delta_A=x_Ah(w,r_A)+k_A(\bmod p-1)$

然后将 (r_A,w,δ_A) 发送给 B。

(2)B 收到 (r_A,w,δ_A) 后，验证：$g^{\delta_A}\overset{?}{=}y_A^{h(w,r_A)}r_A(\bmod p)$，若成立，接收此代理权。

3.3 代理签名人私钥进化及单向函数值生成与验证

为了使签名方案具有后向安全性，需要 B 生成一单向函数值，然后用前向安全签名算法对该值进行签名，并将签名发送给 A 进行有效性验证。

(1)签名进入第 $i(1\le i\le T)$ 时段，B 用第 $i-1$ 时段的签名私钥 $x_{B_{i-1}}$ 计算第 i 时段的签名私钥：$x_{B_i}=x_{B_{i-1}}^2(\bmod p-1)$，并立刻删除 $x_{B_{i-1}}$。

(2)B 选择随机数 t_i，k_1，k_2，计算：

$y_i=g^{t_i}(\bmod p)$

$v_1=g^{k_1}(\bmod p)$，$v_2=x_{B_i}g^{k_2}(\bmod p)$

$s_i=(h(y_i,v_1,v_2)-k_22^{T+1-i}v_1)k_1^{-1}(\bmod p-1)$

发送 (i,v_1,v_2,s_i) 给 A。

(3)A 收到 (i,v_1,v_2,s_i) 后，验证：$y_B^{v_1}v_1^{s_i}v_2^{2^{T+1-i}v_1}=g^{h(y_i)}(\bmod p)$。如果等式成立。$A$ 用 ElGamal 签名算法生成 $sig_{x_A}(y_i)$，然后将 $(y_i,sig_{x_A}(y_i))$ 发送给 B。

3.4 代理密钥生成

(1)B 收到 $(y_i,sig_{x_A}(y_i))$ 后，验证签名有效性，若验证有效，则继续下一步骤。

(2)B 随机选择 $r_i\in Z_{p-1}^*$，计算：

$R_i=x_Bg^{r_i}(\bmod p)$，$\delta_i=r_ih(w,r_A)+\delta_A+t_i(\bmod p-1)$

其中，δ_i 即为 i 时段的代理签名密钥。

3.5 代理数字签名的生成及验证

代理人 B 利用第 i 时段的代理密钥 δ_i 对消息 m 的签名过程如下：

(1)B 随机选取 $k_p\in Z_{p-1}^*$，计算：

$r_p=g^{k_p}(\bmod p)$，$\delta=[h(i,m,R_i,r_p,y_i)-\delta_i2^{T+1-i}r_p]k_p^{-1}(\bmod p-1)$

然后 B 把对消息 m 的签名 $(i,w,r_A,R_i,r_p,\delta,(y_i,sig_{x_A}(y_i)))$ 发送给接收者。

(2)接收者收到 $(i,w,r_A,R_i,r_p,\delta,(y_i,sig_{x_A}(y_i)))$ 后。首先验证 $(y_i,sig_{x_A}(y_i))$，若验证通过，则继续验证等式：

$$g^{h(i,m,R_i,r_p,y_i)}=(r_p)^{\delta}[(R_iy_A)^{h(w,r_A)}r_Ay_i]^{2^{T+1-i}.r_p}.y_B^{h(w,r_A).r_p}(\bmod p)$$

若等式成立，则认可签名有效。因为：

$$\delta=[h(i,m,R_i,r_p,y_i)-\delta_i2^{T+1-i}r_p]k_p^{-1}(\bmod p-1)\Rightarrow$$
$$h(i,m,R_i,r_p,y_i)=\delta k_p+\delta_i2^{T+1-i}r_p(\bmod p-1)\Rightarrow$$
$$g^{h(i,m,R_i,r_p,y_i)}=(g)^{k_p\delta}(g)^{(r_ih(w,r_A)+\delta_A+x_i)2^{T+1-i}.r_p}(\bmod p)\Rightarrow$$
$$g^{h(i,m,R_i,r_p,y_i)}=(r_p)^{\delta}[(R_iy_A)^{h(w,r_A)}r_Ay_i]^{2^{T+1-i}.r_p}.y_B^{h(w,r_A).r_p}(\bmod p)$$

3.6 安全性分析

本文提出的方案满足代理签名安全性质：

(1)强不可伪造性：攻击者不能伪造代理签名者 B 进行签名。因为攻击者无法从等式 $\delta_i=r_ih(w,r_A)+\delta_A+t_i(\bmod p-1)$ 得到代理签名密钥 δ_i。等式中 r_i 对于攻击者来说是未知的，若是先确定 r_i 的值，则由于 i 时段 B 的私钥 x_{B_i} 是秘密的，因此从等式 $R_i=x_{B_i}g^{r_i}(\bmod p)$ 无法得到合法的 R_i。若是由 R_i 求 r_i，则必须求解离散对数难题。

此外，攻击者也无法根据签名验证等式 $g^{h(i,m,R_i,r_p,y_i)}=(r_p)^{\delta}[(R_iy_A)^{h(w,r_A)}r_A]^{2^{T+1-i}.r_p}y_B^{h(w,r_A).r_p}(\bmod p)$ 伪造合法的代理签名。假设攻击者先确定 (i,w,r_A,R_i,y_i,r_p) 的值，然后根据签名验证等式求代理签名 δ，这相当于求离散对数问题。

(2)强可识别性：根据授权书 w 和代理签名验证过程，可以确定原始签名人 A 和代理签名人 B 的身份。

(3)强不可否定性：代理签名验证时用到原始签名人 A 和代理签名人 B 的公钥，以及授权书 w，从而 A 不能否定其授权人身份，B 不能否定其代理人身份。

(4)可验证性：见 3.5 节验证过程证明。

(5)防滥用性：授权书 w 包括 A 对 B 的签名授权权限及签名时限等，能有效防止代理签名权的滥用。

(6)前向安全性：假设攻击者获得了代理签名者 B 的第 i 时段的签名密钥 x_{B_i}，他也不能伪造第 j $(j<i)$ 时段的合法代理签名。因为已知 x_{B_i} 求代理签名密钥 $\delta_j=r_jh(w,r_A)+\delta_A+t_j(\bmod p-1)$，必需先求得 x_{B_j}，这意味着攻击者要解决模合数的二次剩余问题。所以攻击者无法根据 x_{B_i} 计算出代理密钥 δ_j，从而也无法获得合法的代理签名。

假设攻击者得到了代理签名密钥 δ_i，但由 δ_i 只能得到 x_{B_i}，同上述分析，从 x_{B_i} 无法计算出有效的代理签名 δ_j。

由此可见攻击者获得 x_{B_i} 或 δ_i，都无法生成第 i 时段之前的有效代理签名。因此，该签名方案是前向安全的。

(7)后向安全性：后向安全性指即使私钥泄漏后，签名也是安全的。因为签名进入 i 时段，B 选择随机数 t_i，计算

（下转第 166 页）