文章编号:1000 - 2367(2017)01 - 0071 - 08 DOI:10.16366/j.cnki.1000 - 2367.2017.01.012

对一种无证书聚合签名方案的攻击与改进

汤鹏志[a,b], 郭红丽[a,b], 张婷婷[a,b], 陈祚松[a,b], 胡凯雨[a,b], 周庆[a,b]

(华东交通大学 a. 理学院;b. 系统工程与密码学研究所, 南昌 330013)

摘 要:无证书聚合签名方案能够有效提高签名验证阶段的效率,其存在两类攻击,在类型 I 攻击中,攻击者不知道系统主密钥和用户的部分私钥,但能替换用户的公钥;在类型 II 攻击中,攻击者知道系统主密钥和用户的部分私钥,但不能替换用户公钥. 无证书聚合签名方案只有同时能够抵抗这两类攻击,才能说明方案是安全的. 大多数无证书聚合签名方案在随机预言机模型下证明了其安全性,但是有些方案不能抵抗类型 II 攻击. 以陈提出的无证书聚合签名方案为例,给出一种适用于一些无证书聚合签名方案的对应攻击方法. 攻击者在拥有系统主密钥的情况下,根据两个有效的签名可以伪造出任意一个消息的有效签名. 在此基础上提出了一个改进的无证书聚合签名方案,并在随机预言机模型下证明了新方案针对类型 I 攻击和类型 II 类攻击是存在性不可伪造的.

关键词:无证书;聚合签名;伪造签名;随机预言机;存在性不可伪造

中图分类号:TP309.2 **文献标志码**:A

2003 年,文献[1]提出无证书公钥密码系统(certificateless PKC, CLPKC),能够有效解决密钥托管问题. CLPKC 拥有系统主密钥的密钥生成中心(key generation center, KGC),KGC 生成与用户身份对应的部分私钥,并通过安全信道将其传送给用户,用户将选取的秘密值与部分私钥结合形成私钥. 2003 年,文献[2]首次提出聚合签名,聚合签名者将任意 $n(n \geq 2)$ 个用户 $U_i(1 \leq i \leq n)$ 对 n 个不同的消息 $m_i(1 \leq i \leq n)$ 产生的签名 $\sigma_i(i = 1,2,\cdots,n)$ 压缩成一个签名 σ,验证者只需验证签名 σ 就能够同时实现对 n 个不同消息的签名验证[3]. 在签名验证阶段,降低了签名的存储空间和通信代价,提高了签名验证和传输效率[3]. 在 2007 年,文献[4]首次提出两个无证书聚合签名方案. 文献[5 - 6]首次给出无证书聚合签名的安全模型,并证明给出的聚合签名方案是不可伪造的. 近年来,国内学者深入研究聚合签名,在签名验证阶段,大大提高了验证的效率. 2012 年,文献[7]提出可证安全的常数长度无证书聚合签名方案,但是没有给出能够抵抗第 II 类攻击的可证安全性证明. 2013 年,文献[8]提出高效的可证明安全的无证书聚合签名方案,但是在证明过程中,未能对第 II 类攻击成功构造 computational Diffie - Hellman(CDH)困难问题. 2014 年,文献[9]提出的无证书聚合签名方案,不能抵抗文献[10]的攻击. 2016 年,文献[11]指出文献[8]不满足不可伪造性,提出改进方案. 分析发现,文献[11]存在漏洞,其方案不能抵抗第 II 类攻击,在拥有系统主密钥的情况下,根据两个消息的有效签名可以伪造出任意一个消息的有效签名. 针对该漏洞,本文提出一个改进的无证书聚合签名方案,在随机预言机模型下证明了方案的安全性,并给出概率分析.

1 基础知识

1.1 双线性对

双线性对是代数曲线上的 Weil 对[12]和 Tate 对[13]. 假设 l 是一个安全参数,q 是一个 l-bit 的素数,G_1 是 q 阶循环加法群,G_2 是 q 阶循环加法群,G_T 是 q 阶循环乘法群. 称 $e:G_1 \times G_2 \rightarrow G_T$ 是双线性对,若满足以下

收稿日期:2016-09-22;**修回日期**:2016-12-23.

基金项目:国家自然科学基金(11361024;11261019);江西省自然科学基金项目(20151BAB201002);江西省研究生创新专项资金项目(YC2015-S255).

作者简介:汤鹏志(1961 -),男,江西九江人,华东交通大学教授,研究方向为信息系统及其安全.

通信作者:郭红丽(1988 -),女,河南登封人,华东交通大学硕士研究生,研究方向为信息安全,E-mail:1260485833@qq.com.

性质:

(1) 双线性:对任意的 $P \in G_1, Q \in G_2, a, b \in Z_q^*$, $e(aP, bQ) = e(P, Q)^{ab}$;

(2) 非退化性:存在 $P \in G_1, Q \in G_2, e(P, Q) \neq 1$;

(3) 可计算性:对任意的 $P \in G_1, Q \in G_2$,存在有效算法计算 $e(P, Q)$.

1.2 CDH 困难问题

计算性 Diffie - Hellman (CDH) 问题:对 $\forall a, b \in Z_q^*$,给定 $P, aP, bP \in G_1$,计算 $abP \in G_1$.

1.3 无证书聚合签名的攻击类型

在无证书聚合签名方案中,密钥由两部分组成:由密钥生成中心生成系统主密钥,并通过计算,得到用户的部分私钥;用户自己生成秘密值. 这两部分形成了用户的私钥. 因此,攻击者存在两种:

类型 I 敌手(A_{I}):攻击者不知道系统主密钥和用户的部分私钥,但能替换用户的公钥.

类型 II 敌手(A_{II}):攻击者知道系统主密钥和用户的部分私钥,但不能替换用户公钥.

1.4 无证书聚合签名的安全模型

类型 I 和类型 II 的安全模型请参考文献[11].

1.5 对文献[11]方案的回顾

(1)系统建立:输入参数 k,输出参数 $p = (k, e, G, G_T, P, X, Y, P_p, H_1, H_2, H_3, H_4, H_5)$. 其中 e 为双线性对: $G \times G \to G_T$, P 为生成元. $X \in G, Y \in G$. $P_p = sP$ 为系统公钥, $s \in Z_p$ 为系统主密钥. $H_1: \{0,1\}^* \to G, H_2: \{0,1\}^* \to G, H_3: \{0,1\}^* \to G, H_4: \{0,1\}^* \to Z_p, H_5: \{0,1\}^* \to Z_p$ 为 Hash 函数.

(2)部分密钥生成:用户身份为 I_i,KGC 计算 $Q_i = H_1(I_i), D_i = sQ_i$,通过秘密信道将 D_i 给用户 I_i.

(3) 用户密钥生成:用户 I_i 随机选择 $x_i \in Z_p$,计算 $P_i = x_i P$ 作为用户的公钥. 其私钥为 $S_i = (x_i, D_i)$.

(4) 签名:给定用户 $I_i (1 \leq i \leq n)$ 和待签名消息 $M = \{m_i\}_{i=1}^n$,聚合签名者随机选取 $\alpha \in Z_p$ 作为状态信息,且公开 α. 下面对消息 m_i 的签名,计算 $T = H_2(\alpha X, P_p), W = H_3(\alpha Y, P_p)$. 随机选取 $r_i \in Z_p$,计算 $R_i = r_i P$, $h_i = H_4(m_i, I_i, P_i, T, W), g_i = H_5(m_i, I_i, P_i, T, W), V_i = h_i D_i + g_i x_i W + r_i T$,输出签名 $\sigma_i = (R_i, V_i)$.

(5) 聚合签名:对收到的签名 $(\{I_i\}_{i=1}^n, \{m_i\}_{i=1}^n, \{\sigma_i\}_{i=1}^n)$ 进行压缩操作,即聚合. 聚合者计算 $T = H_2(\alpha X, P_p), W = H_3(\alpha Y, P_p), R = \sum_{i=1}^n R_i, V = \sum_{i=1}^n V_i$,输出聚合后的签名 $\sigma = (R, V)$.

(6) 签名验证:输入 $(\{I_i\}_{i=1}^n, \{m_i\}_{i=1}^n, \sigma)$,验证者计算 $T = H_2(\alpha X, P_p), W = H_3(\alpha Y, P_p), h_i = H_4(m_i, I_i, P_i, T, W), g_i = H_5(m_i, I_i, P_i, T, W), Q_i = H_1(I_i)$,验证 $e(P, V)$ 与 $e(T, R) e(\sum_{i=1}^n h_i Q_i, P_p) e(W, \sum_{i=1}^n g_i P_i)$ 是否相等. 若两式相等,则验证通过,返回成功;否则,验证失败,返回"$\perp$".

2 对文献[11]的安全性分析

针对无证书聚合签名和第 II 类攻击的特点,本文给出了一个通用的攻击方法. 在文献[11]中,签名阶段的 $R_i = r_i P$,其不受其他条件的约束,导致 R_i 被伪造. 伪造者只需得到两个消息的有效签名,就可以伪造出任意一个消息的有效签名. 攻击者拥有系统主密钥,但不能替换用户公钥. 其攻击过程如下:

攻击者知道系统主密钥 s,因此能够计算出用户的部分私钥 $D_i = sQ_i$,其中 $Q_i = H_1(I_i)$. 因 $V_i = h_i D_i + g_i x_i W + r_i T, r_i \in Z_p, h_i = H_4(m_i, I_i, P_i, T, W), g_i = H_5(m_i, I_i, P_i, T, W)$;对于待签名消息 m'_i,签名得到 $V'_i = h'_i D_i + g'_i x_i W + T r'_i$,其中 $h'_i = H_4(m'_i, I_i, P_i, T, W), g'_i = H_5(m'_i, I_i, P_i, T, W)$. 因为 $T = H_2(\alpha X, P_p)$, $W = H_3(\alpha X, P_p)$ 且 $\alpha \in Z_p$ 是公开的,因此两次得到的 T 和 W 可以相同.

由以上等式可得: $\hat{V}_i = V_i - h_i D_i = g_i x_i W + r_i T, \hat{V}'_i = V'_i - h'_i D_i = g'_i x_i W + T r'_i$,因此 $\hat{V}_i - \hat{V}'_i = (g_i - g'_i) x_i W + (r_i - r'_i) T$ $\frac{\hat{V}_i - \hat{V}'_i}{g_i - g'_i} = x_i W + \frac{(r_i - r'_i) T}{g_i - g'_i}$. 给定一个可伪造的消息 m^*,攻击者计算 $h^* = H_4(m^*, I_i, P_i, T, W), g^* = H_5(m^*, I_i, P_i, T, W)$, 因此 $\frac{(g^* - g_i)(\hat{V}_i - \hat{V}'_i)}{g_i - g'_i} = (g^* - g_i) x_i W + \frac{(g^* - g_i)(r_i - r'_i) T}{g_i - g'_i}$.

假设

$$V_i^* = \frac{(g^* - g_i)(\hat{V}_i - \hat{V}'_i)}{g_i - g'_i} + \hat{V}_i + h^* D_i = (g^* - g_i)x_i W + \frac{(g^* - g_i)(r_i - r'_i)T}{g_i - g'_i} + \hat{V}_i + h^* D_i =$$

$$g^* x_i W + h^* D_i + \frac{(g^* - g_i)(r_i - r'_i)T}{g_i - g'_i} + r_i T = g^* x_i W + h^* D_i + [\frac{(g^* - g_i)(r_i - r'_i)T}{g_i - g'_i} + r_i]T,$$

在签名阶段,令 $r_i^* = \frac{(g^* - g_i)(r_i - r'_i)}{g_i - g'_i} + r_i, R_i^* = r_i^* P, V_i^* = \frac{(g^* - g_i)(\hat{V}_i - \hat{V}'_i)}{g_i - g'_i} + \hat{V}_i + h^* D_i$,则验证等式 $e(V_i^*, P) = e(h^* Q_i, P_p)e(W, g^* P_i)e(T, R_i^*)$ 成立. 通过以下等式验证伪造的签名成立:

$$e(V_i^*, P) = e(g^* x_i W + h^* D_i + [\frac{(g^* - g_i)(r_i - r'_i)}{g_i - g'_i} + r_i]T, P) = e(g^* x_i W, P)e(h^* D_i, P) \times$$

$$e([\frac{(g^* - g_i)(r_i - r'_i)}{g_i - g'_i} + r_i]T, P) = e(g^* x_i W, P)e(h^* s Q_i, P)e([\frac{(g^* - g_i)(r_i - r'_i)}{g_i - g'_i} +$$

$$r_i]T, P) = e(h^* Q_i, P_p)e(W, g^* P_i)e(T, R_i^*),$$

由此可见,$\sigma_i^*(R_i^*, V_i^*)$ 是 I_i 对 m^* 的有效签名,攻击成功.

3　改进的无证书聚合签名方案

为了解决以上问题,本文 R_i 嵌入到哈希函数 H_4 和 H_5 中,引入状态信息 θ. 在签名阶段,改变签名函数表达式. 改进的无证书聚合签名方案如下.

(1) 系统建立. 输入安全参数 k,系统输出参数 $p = (k, e, G, G_T, P, X, Y, P_p, H_1, H_2, H_3, H_4, H_5, H_6)$. 其中 $e: G \times G \to G_T$,表示一个双线性对映射;P 为群 G 的生成元;随机选择 $s \in Z_p$ 为系统主密钥,计算 $P_p = sP$ 作为系统公钥;哈希函数 H_1, H_2, H_3 满足 $\{0,1\}^* \to G$,H_4, H_5, H_6 满足 $\{0,1\}^* \to Z_P$,且均是强无碰撞的;$X \in G, Y \in G$.

(2) 用户部分密钥生成. 给定用户身份 I_i,PKG 计算 $Q_i = H_1(I_i)$,$D_i = sQ_i$,将 D_i 通过安全信道发送给用户 I_i.

(3) 用户密钥生成. 用户 I_i 随机选择 $x_i \in Z_P$,计算 $P_i = x_i P$,设定用户的私钥 $S_i = (x_i, D_i)$,公钥 P_i.

(4) 签名. 签名聚合者产生状态信息 $\theta \in Z_P$,签名者 I_i 按以下步骤对消息 m_i 签名,计算 $T = H_2(\theta X, P_p)$,$W = H_3(\theta Y, P_p)$;随机选择 $r_i \in Z_P$,计算 $R_i = r_i P, h_i = H_4(m_i, I_i, P, T, W, R_i, \theta), g_i = H_5(m_i, I_i, P_i, T, W, R_i, \theta), u_i = H_6(m_i, I_i, D_i, P_i, T, W, R_i, \theta)$;计算 $V_i = h_i D_i + g_i x_i W + r_i u_i T$,输出签名 $\sigma_i = (R_i, V_i, \theta)$.

(5) 聚合. 输入参数 p,选择相同状态 θ 下的 n 个签名 $(\{I_i\}_{i=1}^n, \{m_i\}_{i=1}^n, \{\sigma_i\}_{i=1}^n)$,聚合签名者计算 $R = \sum_{i=1}^n R_i, V = \sum_{i=1}^n V_i$ 输出聚合签名 $\sigma = (R, V, \theta)$.

(6) 签名验证. 输入参数 p,签名者 $\{I_i\}_{i=1}^n$ 及其对应的公钥 $\{P_i\}_{i=1}^n$,消息 $\{m_i\}_{i=1}^n$ 及聚合签名 σ. 验证者计算 $T = H_2(\theta X, P_p)$,$W = H_3(\theta Y, P_p)$,$R_i = r_i P, h_i = H_4(m_i, I_i, P_i, T, W, R_i, \theta), g_i = H_5(m_i, I_i, P_i, T, W, R_i, \theta)$,$u_i = H_6(m_i, I_i, P_i, T, W, R_i, \theta), Q_i = H_1(I_i)$. 验证等式 $e(V, P) = e(\sum_{i=1}^n h_i Q_i, P_p)e(W, \sum_{i=1}^n g_i P_i)e(T, \sum_{i=1}^n u_i R_i)$ 是否成立. 若等式成立,则验证成功;否则验证失败,返回"⊥".

4　方案分析

4.1　正确性分析

定理 1　本文的无证书聚合签名方案是正确的.

$$e(V, P) = e(\sum_{i=1}^n (h_i D_i + g_i x_i W + r_i u_i T), P) = e(\sum_{i=1}^n h_i D_i, P)e(\sum_{i=1}^n g_i x_i W, P)e(\sum_{i=1}^n r_i u_i T, P) =$$

$$e(\sum_{i=1}^n h_i Q_i, sP)e(W, \sum_{i=1}^n g_i x_i P)e(T, \sum_{i=1}^n u_i r_i P) = e(\sum_{i=1}^n h_i Q_i, P_p)e(W, \sum_{i=1}^n g_i P_i)e(T, \sum_{i=1}^n u_i R_i).$$

4.2 安全性分析

下面证明在 CDH 困难问题下,本文方案在随机预言机模型下是存在性不可伪造的.

定理 2 在随机预言模型下,存在类型 I 攻击者 A_1,在多项式时间 t 内分别进行至多 $q_d,q_p,q_m,q_{Hi}(1 \le i \le 6),q_s$ 次部分私钥、公钥询问、秘密值、$H_i(1 \le i \le 6)$、签名询问. 若 A_1 能够在 t 内以一个不可忽略的概率 ε 在游戏 G_1 中获胜,则存在一个挑战者 C,能够在 $t' \le \frac{1}{\varepsilon'}(t + \sum_{i=1}^{6} t_{Hi} + t_{PS} + t_M + t_{PK} + t_S)(1 \le i \le 6)$ 内以优势 $\varepsilon' \ge \frac{1}{q_d n}\varepsilon$ 解决 CDH 困难问题.

证明 在第 I 类攻击中,攻击者 A_1 不知道系统主密钥,但可以替换用户公钥. 挑战者 C_2 与攻击者 A_2 的交互过程如下.

系统初始化阶段 在游戏 G_1 中,C 运行系统建立算法并获得系统参数$(k,e,G,G_T,P,X,Y,P_p,H_1,H_2,H_3,H_4)$ 给 A_1,其中 $P_p = aP$,假设 I^* 为目标用户,C 维护和更新列表 $L_{H1},L_{H2},L_{H3},L_{H4},L_{H5},L_{H6},L_{PK},L_S$,各列表初始状态下为空.

询问阶段 攻击者 A_1 对挑战者 C 进行多项式有界次询问. 询问过程如下.

(1)H_1 询问:C 维持并记录列表 $L_{H_1} = \{(I_i,Q_i,D_i,\alpha_i)\}$,当 A_1 对 I_i 进行 H_1 询问时,C 先查表 L_{H_1},若(I_i,Q_i,D_i,α_i) 在 L_{H_1} 中,则返回 Q_i 给 A_1,C 随机选取 $\alpha_i \in Z_P$. 若 $I_i = I^*$,计算 $Q_i = bP,D_i =$ "$\perp$",记录$(I_i,Q_i,\perp,\perp)$ 到 L_{H_1},并返回 Q_i 给 A_1;否则计算 $Q_i = \alpha_i P,D_i = \alpha_i aP = \alpha_i P_p$,记录$(I_i,Q_i,D_i,\alpha_i)$ 到 L_{H_1},返回 Q_i 给 A_1.

(2)H_2 询问:C 维持并记录列表 $L_{H_2} = \{(\theta,X,P_p,T)\}$,当 A_1 对$(\theta X,P_p)$ 进行 H_2 询问时,C 先查表 L_{H_2},若(θ,X,P_p,T) 在 L_{H_2} 中,返回 T 给 A_1. 否则,随机选择 $\gamma_i \in Z_P$,计算 $T = \gamma_i P$,返回 T 给 A_1,记录(θ,X,P_p,T) 到 L_{H_2}.

(3)H_3 询问:C 维持并记录列表 $L_{H_3} = \{(\theta,Y,P_p,W,\beta_i)\}$,当 A_1 对$(\theta Y,P_p)$ 进行 H_3 询问时,C 先查表 L_{H_3},若(θ,Y,P_p,W,β_i) 在 L_{H_3} 中,则返回 W 给 A_1. 否则,C 随机选取 $\beta_i \in Z_P$,计算 $W = \beta_i P$,返回 W 给 A_1,记录(θ,Y,P_p,W,β_i) 到 L_{H_3}.

(4)H_4 询问:C 维持并记录列表 $L_{H_4} = \{(m_i,I_i,P_i,T,W,R_i,\theta,h_i)\}$,当 A_1 对$(m_i,I_i,P_i,T,W,R_i,\theta)$ 进行 H_4 询问时,C 先查表 L_{H_4},若$(m_i,I_i,P_i,T,W,R_i,\theta,h_i)$ 在 L_{H_4} 中,则返回 h_i 给 A_1. 否则随机选择 $h_i \in Z_P$,记录$(m_i,I_i,P_i,T,W,R_i,\theta,h_i)$ 到 L_{H_4}.

(5)H_5 询问:C 维持并记录列表 $L_{H_5} = \{(m_i,I_i,P_i,T,W,R_i,\theta,g_i)\}$,当 A_1 对$(m_i,I_i,P_i,T,W,R_i,\theta)$ 进行 H_5 询问时,C 先查表 L_{H5},若$(m_i,I_i,P_i,T,W,R_i,\theta,g_i)$ 在 L_{H_5} 中,则返回 g_i 给 A_1. 否则随机选择 $g_i \in Z_P$,记录$(m_i,I_i,P_i,T,W,R_i,\theta,g_i)$ 到 L_{H_5}. 且(θ^*,m_1^*,I_1^*) 也没有被进行签名询问. 假若聚合签名 $\sigma^* = e(R_1^*,R_2^*,\cdots,R_n^*,V^*)$,其中 $V^* = \sum_{i=1}^{n} V_i^*,R^* = \sum_{i=1}^{n} R_i^*$. 且满足 $e(V^*,P) = e(T^*,\sum_{i=1}^{n} u_i^* R_i^*)e(\sum_{i=1}^{n} h_i^* Q_i^*,P_p)e(W^*,\sum_{i=1}^{n} g_i^* P_i^*)$.

(6)H_6 询问:C 维持并记录列表 $L_{H6} = \{(m_i,I_i,P_i,T,W,R_i,\theta,u_i)\}$,当 A_1 对$(m_i,I_i,P_i,T,W,R_i,\theta)$ 进行 H_6 询问时,C 先查表 L_{H6},若$(m_i,I_i,P_i,T,W,R_i,\theta,u_i)$ 在 L_{H_6} 中,则返回 u_i 给 A_1. 否则随机选择 $u_i \in Z_P$,记录$(m_i,I_i,P_i,T,W,R_i,\theta,u_i)$ 到 L_{H_6}.

(7) 部分私钥提取询问:当 A_1 对 I_1 进行部分私钥询问时,如果 $I_i \ne I^*$,C 对 I_i 执行 H_1 询问,查询列表 L_{H_1},输出 D_i 给 A_1;若 $I_i = I^*$,C 输出"$\perp$" 给 A_1,并终止游戏.

(8) 公钥询问:C 维持并记录列表 $L_{P,K} = \{(I_i,x_i,P_i)\}$,当 C 收到 A_1 对用户 I_i 的公钥询问时,如果(I_i,x_i,P_i) 在列表 $L_{P,K}$ 中,返回 P_i 给 A_1;否则,C_1 随机选择 $x_i \in Z_P$,计算 $P_i = x_i P$,返回 P_i 给 A_1,并添加(I_i,x_i,P_i) 到列表 $L_{P,K}$.

(9) 秘密值询问:当 C 收到 A_1 对 I_i 的秘密值询问时,C 执行公钥询问访问列表 $L_{P,K}$,如果 $I_i \ne I^*$,则输出

x_i 给 A_1;否则,输出"$\perp$"给 A_1,并终止游戏(假设第 I 类型的攻击者已进行公钥替换询问,步骤可忽略).

(10) 公钥替换询问:当 C 收到 A_1 对用户(I_i,P_i)进行公钥替换询问时,C 先查找列表 $L_{P,K}$ 中的数组(I_i,x_i,P_i)(若不存在,先执行公钥询问),将 P_i 替换 A_1 为可自由选择的 P'_i,最后 C 更新列表 $L_{P,K}$,将(I_i,x_i,P_i)替换为$(I_i,\perp,P'_i)$.

(11) 签名询问:当 $I \neq I^*$ 时,按照方案中的签名方法输出 $\sigma_i = (R_i,V_i,\theta)$;当 $I_i = I^*$ 时,随机选择 $r_k \in Z_P$,$R_i = r_iP - (Tu_i)^{-1}h_iP_pQ_i$,$V_i = g_i\beta_iP_i + r_iu_iT$,输出 $\sigma_i = (R_i,V_i,\theta)$. 将$(I_i,r_i,R_i,V_i)$保存在列表 L_S 中.

攻击阶段　一轮询问结束后,攻击者 A_1 输出$(\theta^*,m^*,I^*,\sigma^*)$,其中 θ^* 为状态信息,$m^* = (m_1^*,m_2^*,\cdots,m_n^*)$,$I^* = (I_1^*,I_2^*,\cdots,I_n^*)$,$\sigma^*$ 表示$(\sigma_1^*,\sigma_1^*,\cdots,\sigma_n^*)$的聚合签名. 假设没有对目标用户 I_j^* 询问,则有

$$e(V^*,P) = e(\sum_{i=1}^{n} h_i^*Q_i^*,P_p)e(W^*,\sum_{i=1}^{n} g_i^*P_i^*)e(T^*,\sum_{i=1}^{n} u_i^*R_i^*). \tag{1}$$

由于 $P_p = aP$,$Q_j^* = bP(i=j)$,$W^* = \beta^*P$,$Q_i^* = \alpha_i^*P(i \neq j)$,$T^* = \gamma^*P$,根据(1)式可得:

$$e(h_j^*Q_j^*,P_p) = e(V^*,P)e(\sum_{i=1,i\neq j}^{n} h_i^*Q_i^*,P_p)e(-W^*,\sum_{i=1}^{n} g_i^*P_i^*)e(-T^*,\sum_{i=1}^{n} u_i^*R_i^*),$$

$$e(h_j^*bP,aP) = e(V^*,P)e(\sum_{i=1,i\neq j}^{n} h_i^*\alpha_i^*P,-P_p)e(-\beta^*P,\sum_{i=1}^{n} g_i^*P_i^*)e(-\gamma^*P,\sum_{i=1}^{n} u_i^*R_i^*),$$

则 $e(h_j^*abP,P) = e(V^* - \sum_{i=1,i\neq j}^{n} h_i^*D_i^* - \sum_{i=1}^{n}\beta^*g_i^*P_i^* - \sum_{i=1}^{n}\gamma^*u_i^*R_i^*,P)$,则 $abP = (h_j^*)^{-1}(V^* - \sum_{i=1,i\neq j}^{n} h_i^*D_i^* - \sum_{i=1}^{n}\beta^*g_i^*P_i^* - \sum_{i=1}^{n}\gamma^*u_i^*R_i^*)$ 为 CDH 困难问题的解.

下面计算 C 成功解决 CDH 困难问题实例的概率优势.

(1)E_1:C 在询问过程中,没有返回失败;

(2)E_2:A_1 能够成功伪造至少包含一个(m^*,I^*)的聚合签名;

(3)E_3:在 E_2 的前提下,至少存在 1 条记录满足 $I_i \neq I^*$.

若如果以上事件全部发生,则称 C 成功. 即求:$P_r[E_1 \wedge E_2 \wedge E_3] = P_r[E_1]P_r[E_2 \mid E_1]P_r[E_3 \mid E_1 \wedge E_2]$,只需求出该概率的一个下限即可. 对于事件 E_1,至少有一次没有对目标用户进行部分私钥询问,故 $P_r[E_1] \geq \frac{1}{q_d}$;$E_2$ 表示类型 I 攻击者 A_1 在游戏 G_1 中获胜,则 $P_r[E_2 \mid E_1] \geq \varepsilon$;根据 E_3 可知,在 n 次独立重复试验中,至少发生一次的概率为 $P_r[E_3 \mid E_1 \wedge E_2] \geq \frac{1}{n}$,则 C 成功的概率 $\varepsilon' = P_r[E_1 \wedge E_2 \wedge E_3] \geq \frac{1}{q_dn}\varepsilon$. 因此挑战者 C 以 $\varepsilon' \geq \frac{1}{q_dn}\varepsilon$ 的优势解决 CDH 困难问题. 在整个询问过程中,所用的时间包括 A_1 运行时间,C 的回答及其伪造 CDH 困难问题的时间. C 总的运行时间为 $t' \leq \frac{1}{\varepsilon'}(t + \sum_{i=1}^{6} t_{H_i} + t_{P,S} + t_M + t_{P,K} + t_S)(1 \leq i \leq 6)$ 为一个多项式界限. 其中 $t_{H_i}(1 \leq i \leq 6)$,$t_{P,S}$,t_M,$t_{P,K}$,t_S 表示 $H_i(1 \leq i \leq 6)$ 询问、部分私钥提取询问、秘密值询问、公钥替换询问、签名询问的时间.

因此 C 能够在 $t' \leq \frac{1}{\varepsilon'}(t + \sum_{i=1}^{6} t_{H_i} + t_{P,S} + t_M + t_{P,K} + t_S)(1 \leq i \leq 6)$ 内成功解决 CDH 问题从而获胜的优势为 $\varepsilon' \geq \frac{1}{q_dn}\varepsilon$.

定理 3　在随机预言模型下,针对类型 II 攻击者 A_2,在多项式时间 t 内至多进行 q_p,q_m,$q_{H_i}(1 \leq i \leq 6)$,q_s 次公钥、秘密值、$H_i(1 \leq i \leq 6)$ 和签名询问,A_2 在时间 t 内以一个不可忽略的概率 ε 在游戏 G_2 中获胜,即能够伪造出本文方案中的一个有效的签名,则存在一个挑战者 C 能够在 $t' \leq \frac{1}{\varepsilon'}(t + \sum_{i=1}^{6} t_{H_i} + t_{P,S} + t_M + t_{P,K} +$

$t_S)(1 \leq i \leq 6)$ 内成功以 $\varepsilon' \geq \frac{1}{q_d n}\varepsilon$ 的优势解决 CDH 问题.

证明 在询问过程中,将 Hash 函数 $H_i(1 \leq i \leq 6)$ 当作是随机预言机. 在类型 II 攻击中,攻击者 A_2 拥有系统主密钥,但是无法替换用户的公钥. 挑战者 C 与攻击者 A_2 的交互过程如下.

系统初始化阶段 在游戏 G_2 中,挑战者 C 输入安全参数,执行 Setup 算法,输出系统参数$(k,e,G,G_T,P,X,Y,P_p,H_1,H_2,H_3,H_4)$,其中 $P_p = \lambda P$,λ 为系统主密钥,C 发送系统参数 p 和 λ 给 A_2. 设 I^* 为目标用户,C 维护和更新列表各列表 $L_{H_1},L_{H_2},L_{H_3},L_{H_4},L_{H_4},L_{H_5},L_{H_6},L_{P,K},L_S$ 初始状态下为空.

询问阶段 攻击者 A_2 对挑战者 C 进行多项式有界次询问. 询问过程如下.

(1)H_1 询问:C 维持并记录列表 $L_{H_1} = \{(I_i,Q_i,D_i)\}$,当 A_2 对 I_i 进行 H_1 询问时,C 先查表 L_{H_1},若(I_i,Q_i,D_i) 在 L_{H_1} 中,则返回 Q_i 给 A_2;否则,C 随机选取 $Q_i \in Z_P$,计算 $D_i = \lambda Q_i$,记录(I_i,Q_i,D_i) 到 L_{H_1},返回 Q_i 给 A_2.

(2)H_2 询问:C 维持并记录列表 $L_{H_2} = \{(\theta,X,P_p,T,\beta_i)\}$,当 A_2 对$(\theta X,P_p)$ 进行 H_2 询问时,C 先查表 L_{H_2},若(θ,X,P_p,T,β_i) 在 L_{H_2} 中,则返回 T 给 A_2;否则,随机选择 $\beta_i \in Z_P$,计算 $T = \beta_i P$,记录(θ,X,P_p,T,β_i) 到 L_{H_2},返回 T 给 A_2.

(3)H_3 询问:C 维持并记录列表 $L_{H_3} = \{(\theta,Y,P_p,W)\}$,当 A_2 对$(\theta Y,P_p)$ 进行 H_3 询问时,C 先查表 L_{H_3},若(θ,Y,P_p,W) 在 L_{H_3} 中,则返回 W 给 A_2;否则,计算 $W = aP$,记录(θ,Y,P_p,W) 到 L_{H_3},返回 W 给 A_2.

(4)H_4 询问:C 维持并记录列表 $L_{H_4} = \{(m_i,I_i,P_i,T,W,R_i,\theta,h_i)\}$,当 A_2 对$(m_i,I_i,P_i,T,W,R_i,\theta)$ 进行 H_4 询问时,C 先查表 L_{H_4},若$(m_i,I_i,P_i,T,W,R_i,\theta,h_i)$ 在 L_{H_4} 中,则返回 h_i 给 A_2;否则,随机选择 $h_i \in Z_P$,记录$(m_i,I_i,P_i,T,W,R_i,\theta,h_i)$ 到 L_{H_4}.

(5)H_5 询问:C 维持并记录列表 $L_{H_5} = \{(m_i,I_i,P_i,T,W,R_i,\theta,g_i)\}$,当 A_2 对$(m_i,I_i,P_i,T,W,R_i,\theta)$ 进行 H_5 询问时,C 先查表 L_{H_5},若$(m_i,I_i,P_i,T,W,R_i,\theta,g_i)$ 在 L_{H_5} 中,则返回 g_i 给 A_2;否则随机选择 $g_i \in Z_P$,记录$(m_i,I_i,P_i,T,W,R_i,\theta,g_i)$ 到 L_{H_5},返回 g_i 给 A_2.

(6)H_6 询问:C 维持并记录列表 $L_{H_6} = \{(m_i,I_i,P_i,T,W,R_i,\theta,u_i)\}$,当 A_2 对$(m_i,I_i,P_i,T,W,R_i,\theta)$ 进行 H_6 询问时,C 先查表 L_{H_6},若$(m_i,I_i,P_i,T,W,R_i,\theta,u_i)$ 在 L_{H_6} 中,则返回 u_i 给 A_2;否则随机选择 $u_i \in Z_P$,记录$(m_i,I_i,P_i,T,W,R_i,\theta,u_i)$ 到 L_{H_6}.

(7) 公钥询问:C 维持并记录列表 $L_{P,K} = \{(I_i,x_i,P_i)\}$,当 C 收到 A_2 对用户 I_i 的公钥询问时,如果(I_i,x_i,P_i) 在列表 $L_{P,K}$ 中,返回 P_i 给 A_2. 否则,C 随机选择 $x_i \in Z_P$,若 $I_i \neq I^*$,计算 $P_i = x_i P$,返回 P_i 给 A_2,并添加(I_i,x_i,P_i) 到列表 $L_{P,K}$;若 $I_i = I^*$,计算 $P_i = bP$,返回 P_i 给 A_2,并添加$(I_i,\perp,P_i)$ 到列表 $L_{P,K}$.

(8) 秘密值询问:当 C 收到 A_2 对用户 I_i 的秘密值询问时,C 执行公钥询问访问列表 $L_{P,K}$,如果 $I_i \neq I^*$,则输出 x_i 给 A_2;否则,输出" $\perp$ ".

(9) 签名询问:当 $I_i \neq I^*$ 时,按照正常的签名方法输出 $\sigma_i = (R_i,V_i)$;当 $I_i = I^*$ 时,随机选择 $r_i \in Z_P$,$R_i = r_i P - (Tu_i)^{-1}P_i g_i W, V_i = h_i \lambda Q_i + r_i u_i T$,输出 $\sigma_i = (R_i,V_i)$. 将(I_i,r_i,R_i,V_i) 保存在列表 L_S 中.

攻击阶段 一轮询问结束后,攻击者 A_2 输出$(\theta^*,m^*,I^*,\sigma^*)$,其中 θ^* 为状态信息,$m^* = (m_1^*,m_2^*,\cdots,m_n^*)$,$I^* = (I_1^*,I_2^*,\cdots,I_n^*)$,$\sigma^*$ 表示$(\sigma_1^*,\sigma_1^*,\cdots,\sigma_n^*)$ 的聚合签名. 假设没有对目标用户 I_j^* 询问,且(θ^*,m_j^*,I_j^*) 也没有被进行签名询问. 若聚合签名 $\sigma^* = e(R_1^*,R_2^*,\cdots,R_n^*,V^*)$,其中 $V^* = \sum_{i=1}^{n} V_i^*$,$R^* = \sum_{i=1}^{n} R_i^*$ 且满足 $e(V^*,P) = e(T^*,\sum_{i=1}^{n} u_i^* R_i^*)e(\sum_{i=1}^{n} h_i^* Q_i^*,P_p)e(W^*,\sum_{i=1}^{n} g_i^* P_i^*)$,因此可以得到:

$$e(W^*,g_j^* P_j^*) = e(V^*,P)e(-T^*,\sum_{i=1}^{n} u_i^* R_i^*)e(\sum_{i=1}^{n} h_i^* Q_i^*,-P_p)e(-W^*,\sum_{i=1,i\neq j}^{n} g_i^* P_i^*). \quad (2)$$

由于 $P_p = \lambda P, W^* = aP, T^* = \beta^* P, P_i^* = bP(i=j), P_i^* = x_i^* P(i = 1,\cdots,n,且 i \neq j)$. 因此求得的困难问题 abP 过程如下:由(2)式可得,

$$e(aP,g_j^* bP) = e(V^*,P)e(-\beta^* P,\sum_{i=1}^{n} u_i^* R_i^*)e(\sum_{i=1}^{n} h_i^* Q_i^*,-\lambda P)e(-W^*,\sum_{i=1,i\neq j}^{n} g_i^* x_i^* P),$$

$$e(g_j^* abP,P) = e(V^* - \sum_{i=1}^{n}\beta^* u_i^* R_i^* - \sum_{i=1}^{n} h_i^* D_i^* - \sum_{i=1 i\neq j}^{n} W^* g_i^* x_i^*, P),$$

$abP = (g_j^*)^{-1}(V^* - \sum_{i=1}^{n}\beta^* u_i^* R_i^* - \sum_{i=1}^{n} h_i^* D_i^* - \sum_{i=1 i\neq j}^{n} W^* g_i^* x_i^*)$,作为 CDH 困难问题的解.

与定理2类似,能够计算出 C 成功的概率. E_1, E_2, E_3 参考定理2,即求:$P_r[E_1 \wedge E_2 \wedge E_3] = P_r[E_1]P_r[E_2 \mid E_1]P_r[E_3 \mid E_1 \wedge E_2]$,对于事件 E_1,至少有一次对目标用户没有进行秘密值询问,因此 $P_r[E_1] \geq \frac{1}{q_m}$;$P_r[E_2 \mid E_1] \geq \varepsilon$;$P_r[E_3 \mid E_1 \wedge E_2] \geq \frac{1}{n}$,则 C 成功的概率 $\varepsilon' = P_r[E_1 \wedge E_2 \wedge E_3] \geq \frac{1}{q_m n}\varepsilon$. 因此挑战者 C 以 $\varepsilon' \geq \frac{1}{q_m n}\varepsilon$ 的优势解决 CDH 困难问题. 在整个询问过程中,所用的时间包括 A_2 运行时间,C 的回答及其伪造 CDH 困难问题的时间. C 总的运行时间为 $t' \leq \frac{1}{\varepsilon'}(t + \sum_{i=1}^{6} t_{H_i} + t_{P,K} + t_M + t_S)(1 \leq i \leq 6)$ 为一个多项式界限. 其中,$t_{H_i}(1 \leq i \leq 6), t_{P,K}, t_M, t_S$ 分别表示 $H_i(1 \leq i \leq 6)$ 询问、公钥询问、秘密值询问、签名询问的时间.

因此 C 能够在 $t' \leq \frac{1}{\varepsilon'}(t + \sum_{i=1}^{6} t_{H_i} + t_{P,S} + t_M + t_{P,K} + t_S)(1 \leq i \leq 6)$ 内成功解决 CDH 问题从而获胜的优势为 $\varepsilon' \geq \frac{1}{q_d n}\varepsilon$.

4.3 效率分析

对比几个无证书聚合签名方案,从签名、聚合签名和聚合签名验证的计算效率方面比较,列举两种主要运算:双线性对运算(P)和群的标量乘运算(M),n 表示签名者的数量,相对于这两种运算,Hash 运算和群的加法运算计算量是可以忽略不计.

表 1　无证书聚合签名方案的效率比较

方案	签名效率	聚合签名效率	聚合签名验证效率	是否安全
文献[11]方案	$6M$	$2M$	$(2n+2)M+4P$	否
文献[14]方案	$4M$	0	$nM+4P$	是
文献[9]方案	$4M$	0	$nM+4P$	否
文献[15]方案	$4M$	$3M+5P$	$3nM+5P$	否
本文方案	$7M$	$2M$	$(2n+3)M+4P$	是

由表 1 可知,本文的方案效率比原文献效率稍低,但能抵抗针对第 I 类和第 II 类攻击,本文方案是存在性不可伪造的.

5 结束语

本文以文献[11]为例,指出攻击者根据任意两个有效的签名可以伪造出任意一个消息的有效签名,给出无证书聚合签名中针对第 II 类攻击者提出了一个通用的攻击方法,通过改变签名方案中参数和签名阶段的计算式,提出了一个无证书聚合签名方案. 在随机预言机模型下证明方案针对无证书的两类攻击是存在性不可伪造的. 基于无证书的聚合签名能够有效解决多个不同用户对签名消息的认证问题,在签名认证阶段,大大提高了认证效率.

参 考 文 献

[1] Al-Riyami S, Paterson K. Certificateless public key cryptography[C]// Advances in Cryptology-Asiacrypt'2003, LNCS 2894. Berlin: Springer-Verlag,2003:452-473.

[2] Boneh D, Gentry C, Lynn B, et al. Aggregate and verifiably encrypted signatures from bilinear maps[C]//Advances in Cryptology-Eurocrypt'2003, LNCS 2656. Berlin: Springer-Verlag,2003:416-432.

[3] 张华,温巧燕,金正平. 可证安全算法与协议[M]. 北京:科学出版社,2012:215-216.

[4] Gong Z, Long Y, Hong X, et al. Two certificateless aggregate signatures from bilinear maps[C]//8th ACIS International Conference, SNPD' 2007. Piscataway: IEEE Press,2007:188-193.

[5] Zhang L, Zhang F. A new certificateless aggregate signature scheme[J]. Computer Communications,2009,32(6):1079-1085.

[6] Zhang L, Qin B, Wu Q, et al. Efficient many-to-one authentication with certificateless signature[J]. Computer Networks,2010,54(14): 2482-2491.

[7] 陆海军,于秀源,谢琪. 可证安全的常数长度无证书聚合签名[J]. 上海交通大学学报,2012,46(2):259-263.

[8] 杜红珍,黄梅娟,温巧燕. 高效的可证明安全的无证书聚合签名方案[J]. 电子学报,2013,41(1):72-76.

[9] 明洋,赵祥模,王育民. 无证书聚合签名方案[J]. 电子科技大学学报,2014,43(2):188-193.

[10] Chen Y C,Tso R,Mambo M,et al. Certificateless aggregate signature with efficient verification[J]. Security and Communication Networks,2015, 8(13):2232-2243.

[11] 陈明. 改进的签名长度固定的无证书聚合签名方案[J]. 计算机应用研究,2016,33(1):271-280.

[12] Boneh D, Franklin M. Identity-based encryption from the Weil pairing[C]//Advances in Cryptology-Crypto' 2001, LNCS 2139. Berlin: Springer-Verlag,2001:213-229.

[13] Galbraith S,Harrison K,Soldera D. Implementing Tate pairing[C]//Algorithmic Number Theory Symposium,LNCS 2369. Berlin:Springer-Verlag,2002:324-337.

[14] 李艳平,聂好好,周彦伟,等. 新的可证明安全的无证书聚合签名方案[J]. 密码学报,2015,2(6):526-535.

[15] 喻琇瑛,何大可. 基于双线性对的聚合代理签名[J]. 中南大学学报(自然科学版),2015,46(12):4534-4541.

Attack and Improvement on a Certificateless Aggregate Signature Scheme

Tang Pengzhi[a,b], Guo Hongli[a,b], Zhang Tingting[a,b], Chen Zuosong[a,b], Hu Kaiyu[a,b], Zhou Qing[a,b]

(a. School of Science;b. Institute of Systems Engineering and Cryptography, East China Jiaotong University, Nanchang 330013, China)

Abstract: Certificateless aggregate signature scheme can improve the efficiency of the signature verification phase, and the scheme exists two types of attacks: in type I attack, the adversary cannot access the system's master key and the user's private key, but it can replace the user's public key; in type II attack, the adversary knows the system's master key and the user's private key, but it cannot replace the user's public key. A certificateless aggregate signature scheme is secure if it can resist the two types of attacks at the same time. Most of the certificateless aggregate signature schemes prove to be safe in the random oracle model, but some schemes can not resist type II adversaries. This paper makes the certificateless aggregate signature scheme proposed by Chen as an example which gives the corresponding attack method that is suitable for some certificateless aggregate signature schemes. The attacker who has system master key can forge a valid signature for any messages while knowing two valid signatures. The new scheme is proposed and proved to be existentially unforgeable for the type I and type II adversary in the random oracle model.

Keywords: certificateless; aggregate signature; forge a signature; random oracle model; existentially unforgeable