前向安全的聚集签名研究和改进

王琴竹

（运城学院 公共计算机教学部，山西 运城 044000）

摘 要：针对 Di Ma 等人提出的前向安全的顺序聚集签名方案，提出了一个新的改进方案。该方案使每个时段能够对多个消息产生多个签名，同时把签名聚集算法分成两个独立的算法——签名生成算法和聚集算法。新的方案能够满足实时性和并发性要求，还能够满足一些特殊无线传感器网络的需求。

关键词：前向安全；聚集签名；并发性；co-CDH

中图分类号：TN91 **文献标识码**：A **文章编号**：1674-7720(2010)20-0063-03

Study and improvement of forward-secure aggregation signatures

WANG Qin Zhu

(Public Computer Teaching Department, Yuncheng University, Yuncheng 044000, China)

Abstract: In this paper, Di Ma, who firstly advanced by Di Ma called forward secure sequential aggregate (FssAgg) signature, we proposed a modified schemederived from FssAgg signature scheme and called it as forward——secure aggregation of signatures. This newly scheme separates the sign-and-aggregate algorithminto two dependent algorithms and signs messages noboundary per time period. It can be useful in some scenario in wireless sensor network.supplying real-time and better concurrency.

Key words: forward-secure; aggregate signature; concurrency; co-CDH

通常情况下，人们需要管理多个用户对不同消息生成的多个签名。例如，证书链需要包含 CA 中心颁发的对不同证书生成的多个签名。这时聚集签名方案应运而生，它能够聚集由不同用户对不同消息生成的多个签名为单个短签名。

2007 年，Di Ma 提出了几个前向安全的顺序聚集签名方案[1,2](FssAgg)，主要适用于无线传感器网络。他根据无线传感器网络的特点，强调了此网络中数据完整性和可认证性的必要性，并分析了一般技术运用到无线传感器网络中存在的两个问题：传感器的密钥泄漏问题以及存储和通信的限制问题。他构造的前向安全的顺序聚集认证方案不仅能够解决密钥泄漏问题，而且减少了算法的存储空间，提高了通信资源的利用率。但该方案也有不足之处，每个时段只能产生一个签名，有一定的局限性。

本文对 Di Ma 等人提出的前向安全的顺序聚集签名方案进行了改进，改进后的方案使每个时段能够对多个消息产生多个签名；同时，把签名聚集算法分成了两个独立的算法——签名生成算法和聚集算法，该方案能够满足实时性和并发性要求，并证明了签名的前向安全性。

1 Di Ma 的签名方案

Di Ma 等人[3]提出了前向安全的聚集签名(简称 FssAgg)，FssAgg 方案的聚集算法是对一个人的多个签名的聚集：

$$\sigma_{1,n}=\prod_{i=1}^{n}\sigma_i \tag{1}$$

FssAgg 由 4 个算法构成，分别是 FssAgg.Kg、FssAgg.Asign、FssAgg.Avf 和 FssAgg.Update，具体的方案描述如下：

(1)FssAgg.Kg：密钥生成算法，两个输入参数分别是安全性参数 $k\in N$ 和总的时段数 T，返回 T 个公私钥对 $(x_i,pk_i)(i=1,\cdots,T,pk_i=g_2^{x_i})$，其中 $(x_i,pk_i)(i=1,\cdots,T)$ 表示第 i 时段的公私钥对。

(2)FssAgg.ASign：签名产生和聚集算法，根据标准的签名算法，返回第 i 时段消息 m 的签名 σ；输入参数是前一时段的聚集签名(signature-so-far)$\sigma_{1,i-1}$、当前时段的

密钥 x_i 和给定的消息 m_i，签名人首先对消息 m_i 产生一个 BLS 签名 $\sigma_i=H^{x_i}(index||m_i)$($index$ 表示消息 m_i 的存储位置，用来表示消息的顺序性)；接着签名人将签名 σ_i 和前一时段的聚集签名 σ_{i-1} 产生新的聚集签名 $\sigma_{1,i}=\sigma_i\cdot\sigma_{i-1}$。

(3)FssAgg.Avf：签名验证算法，根据聚集签名 $\sigma_{1,i}$ 和一组消息以及公钥础 pk(包括所有时段的公钥)，验证聚集签名：$e(\sigma_{1,n})=\prod_{i=1}^{n}e(h_i,pk_i)$。如果验证通过，则表示签名有效，否则签名无效。

(4)FssAgg.Update：密钥更新算法，是一个单向性的函数，根据当前时段的密钥 x_i，返回下一时段的密钥 x_{i+1}，并及时删除密钥 x_i。

2 改进后的签名方案

2.1 形式化定义

改进的 FssAgg 签名方案[2] 由 5 个算法构成，FSA=(FSA.Kg，FSA.Sign，FSA.Agg，FSA.Avf，FSA.Update)，每个时段签名人可以对多个消息产生多个签名，任何人都可以参与聚集的过程。

2.2 方案描述

根据参考文献[1]，基于一般的聚集签名方案(如 BLS 方案)，使用全域的哈希函数 $H_1(\cdot):\{0,1\}^*\rightarrow G_1$。密钥生成算法为每个签名人随机选取 $x\in Z_q$，并生成公钥 $v=g_2^x$。签名人根据自己的私钥 x 和公钥 $V\in G_2$，计算消息 m 的哈希值 $h=H_1(m)$，然后产生签名 $\sigma=h^x$。为了验证签名的合法性，验证者首先计算哈希值 $h=H_1(m)$，并判断等式 $e(\sigma,g_2)=e(h,v)$ 是否成立。验证算法的计算复杂度等于 2 个双线性映射的复杂度。下面主要对聚集算法和验证算法进行改造。

在第 i 时段，为了产生当前时段的聚集签名，签名人需要前一时段的聚集签名 A_{i-1} 和当前时段产生的 r 个 BLS 签名，计算：

$$A_i=A_{i-1}\cdot\prod_{k=1}^{r}\sigma_{i,k} \tag{2}$$

其中 $\sigma_{i,k}$ 是消息 $m_{i,k}$ 的签名，A_i 的长度等于单个 BLS 签名；任何人都可以调用聚集算法。

在第 i 时段，验证算法根据消息组 $((m_{1,1},\cdots,m_{1,r}),\cdots,(m_{i,1},\ldots,m_{i,r}))$，公钥 pk 和聚集签名 A_i，判断下列等式是否成立，一次验证所有消息的签名的合法性。

$$e(A_i,g_2)\equiv\prod_{j=1}^{i}\prod_{k=1}^{r}e(H_1(j||pk_j||m_{j,k}),pk_j) \tag{3}$$

2.3 FSA 签名方案的五个算法

(1)FSA.Kg：签名人随机选取 $X_0\in Z_p$ 并产生公私钥对 $(x_i,pk_i)(i=1,\cdots,T)$：$x_i=H(x_{i-1})$，$pk_i=g_2^{x_i}$。初始化的密钥为 x_0，公钥为 $(pk_1,\cdots,pk_T)=(g_2^{x_1},\cdots,g_2^{x_T})$，可信方为所有的传感器结点产生上述的公私钥对。而数据接收端只获得公钥。

(2)FSA.Sign：给定消息 $m_{i,k}$，签名人根据当前的密钥 x_i 产生 BLS 签名：$\sigma_{i,k}=H_1^{x_i}(i||pk_i||m_{i,k})$，其中 i 表示消息 $m_{i,k}$ 所在的时段，标记 i 体现了消息的时间顺序，而一般的 BGLS 聚集签名不能够表示出消息的时间顺序。

(3)FSA.Agg：任何人根据当前时段的一组 BLS 签名 $(\sigma_{i,1},\cdots,\sigma_{i,k})$ 和前一时段的聚集签名 A_{i-1}，通过式(2)计算当前时段的聚集签名。

(4)FSA.Avf：验证者根据公钥 pk 判断聚集签名 A_i 是否满足等式(3)。

(5)FSA.Update：签名人利用哈希函数 H 的单向性从前一时段的密钥计算出当前时段的密钥 $x_i=H(x_{i-1})$，并及时删除前一时段的密钥。

3 安全性证明

FSA 签名方案的安全性是基于双线性群上计算 co-CDH 假设困难[4]，下面给出 FSA 签名方案的安全性定理并证明该定理。

定理：假设 (G_1,G_2) 是基于 co-Diffie-Hellman 双线性群对 (t',ξ') 安全的，其中每个群的阶数为素数 P，不同的生成元群 G_1 和 G_2，存在同态映射从 G_2 映射到 G_1，以及双线性映射 $e:G_1\times G_2\rightarrow G_T$。则本方案基于 (G_1,G_2) 在选择密钥攻击模型下和前向安全攻击模型下 (t,qH,qs,T,ξ) 是安全的，其中 t 和 e 满足 $t'=t+o(qH+qs)$，$\xi'=\xi/T$。

证明：假设存在一个攻击者 A 能够以概率 ξ 成功伪造聚集签名。构造一个模拟器 B 调用攻击游戏来计算群域 (G_1,G_2) 下的 co-CDH 问题。给定一个挑战公钥 cpk，模拟器 B 和攻击者 A 按照以下步骤交互：

(1)建立阶段：模拟器 B 首先选择某个时段 $t(1<t<T)$，希望攻击者 A 能够在第 t 时段伪造签名，最后 B 能够得到 $h^{x_t}\in G_1$(给定 h 让 B 伪造其签名)。B 设置 cpk 为第 t 时段的公钥：$cpk=pk_tg_2^{\alpha}\in G_2$，其中 α 是在 Z_p 中的随机数，pk_t 是真正的公钥。给定一个消息，预言机 o_{cpk} 将返回公钥 cpk 下的签名。攻击者 A 按照下面的方式产生其他时段的信息：①B 产生 $t-1$ 个独立的 BLS 公私钥对 $(pk_i,x_i),(i=1,\cdots,t-1)$，作为前 $t-1$ 时段的公私钥对；②B 随机产生一对 BLS 公私钥作为第 $t+1$ 时段的公私钥，然后利用密钥生成函数生成剩下的 $T-t-1$ 个公私钥对。B 提供给 A 公钥 $pk=\{pk_1,\cdots,pk_{t-1},cpk,pk_{t+1},\cdots,pk_T\}$ 和总的时段数 T。

(2)哈希询问：在任何时段，A 可以询问随机预言机 H_1。为了回答 A 的询问，B 建立一个哈希列表 $(m^{(i)},\omega^{(i)},\alpha^{(i)},c^{(i)})$，初始化为空，当 A 询问的消息不在 t 时段时，B 调用签名方案的哈希函数 $H_1(m)$ 产生消息的哈希值并返回；当消息在第 t 时段时，且 A 询问随机预言机消息 $m\in\{0,1\}^*$ 的哈希值时，模拟器 B 按照下面方式返回哈希值：

①若 m 已经被询问过，且存在于哈希列表中的某元组(m,ω,α,c)，则模拟器 B 直接返回哈希值：$H_1(m)=\omega\in G_1$。

②否则，B 随机选取 $c\in\{0,1\}$，其概率 $Pr[c=0]=1/(qs+N)$。

③模拟器 B 随机选取$\beta\in Z_p$。若 $c=0$，则 B 计算 $\omega\leftarrow h\cdot\Psi(g_2)^{\beta}\in G_1$；若 $c=1$，则 B 计算 $\omega\leftarrow\Psi(g_2)^{\beta}\in G_1$。

④模拟器 B 把元组(m,ω,α,c)增加到哈希列表中，同时返回攻击者 A 的哈希值为 $H_1(m)=\omega$。

(3)签名询问：一般情况下，A 可以询问任何时段任何消息的签名，但也存在一个限制，即不允许 A 询问之前时段的签名。在第 i 时段，当 A 询问签名时，选择消息组$(m_{i,1},\cdots,m_{i,r})$，若 $i\neq t$，B 根据自己的公私钥对消息组$(m_{i,1},\cdots,m_{i,r})$返回一组 BLS 签名$(\sigma_{i,1},\cdots,\sigma_{i,r})$（其中$\sigma_{i,k}=H_1^{x_i}(i\|pk_i\|m_{i,k})$；若 $i=t$，按照下面步骤执行：

①模拟器 B 调用上述算法产生消息 m 的哈希值(m,ω,α,c)，若 $c=0$，则 B 宣告失败并终止。

②若 $c=1$，由于 $\omega=\Psi(g_2)^{\beta}\in G_1$，令 $\sigma=\Psi(pk)^{\beta}\cdot\Psi(g_2)^{\alpha\beta}\in G_1$，因此满足等式 $\sigma=\omega^{x_t+\alpha}$，即 σ 是消息 m 在公钥 $cpk=cpk\cdot g_2^{\alpha}=g_2^{x_t+\alpha}$ 下的合法签名。最后，模拟器 B 返回攻击者 A 签名 σ。

(4)攻击阶段：当 A 决定进入攻击阶段时，A 询问 B 某个时段 b 的私钥。若 $b=t$，则停止（这种情况下，B 宣告失败）。若 $b>t$，则 B 提供给攻击者 A 第 b 时段的密钥。

(5)签名输出阶段：最后，当 B 宣告失败时，A 也失败；否则，A 返回公钥 pk 下对消息组$((m_{l,1},\cdots,m_{l,r}),\cdots,(m_{t,1},\cdots,m_{t,r}))$的签名 A_t，且存在消息 $m_{t,r}$ 攻击者没有询问签名预言机。模拟器 B 询问了哈希预言机消息 $m_{t,k}$ $(1\leqslant k\leqslant r)$的哈希值，得到 r 个哈希值元组(m,ω,β,c)。

当 $c_r=0,c_k=1(1\leqslant k\leqslant r-1)$时，模拟器 B 继续执行；否则 B 宣告失败并停止。当 $c_k=0$ 时，则 $\omega_k=h\cdot\Psi(g_2)^{\beta_k}$。当 $c_k=1$ 时，$k\leqslant r-1$，则 $\omega_k=\Psi(g_2)^{\beta_k}$。同时，聚集签名 A_t 必须满足签名验证等式(3)，第 $i(i\neq t)$时段的签名 $\sigma_{i,k}(1\leqslant k\leqslant r)$是合法的签名，第 t 时段的签名 $\sigma_{t,k}(k\neq r)$满足下列等式：

$$e(\sigma_{t,k},g_2)=e(\Psi(pk_t)^{\beta_{t,k}},g_2)=e(\Psi(pk_t),g_2)^{\beta_{t,k}}$$
$$=e(\Psi(g_2),pk_t)^{\beta_{t,k}}=e(\Psi(g_2)^{\beta_{t,k}},pk_t)=e(\omega_{t,k},pk_t)\quad(4)$$

因此，$\sigma_{t,k}(k\neq r)$是消息 $\sigma_{t,k}$ 的合法签名。然后，模拟器 B 计算 $\sigma_{t,r}$ 的值：

$$\sigma_{t,r}=A_t\cdot\prod_{i=1}^{t-1}\prod_{k=1}^{r}\sigma_{i,k}^{-1}\cdot\prod_{k=1}^{r-1}\sigma_{t,k}^{-1}\quad(5)$$

最后，将 $\sigma_{t,r}$ 代入双线性映射：

$$e(\sigma_{t,r},g_2)=e(A_t\cdot\prod_{i=1}^{t-1}\prod_{k=1}^{r}\sigma_{i,k}^{-1}\cdot\prod_{k=1}^{r-1}\sigma_{t,k}^{-1})$$
$$=e(A_t,g_2)\cdot e(\prod_{i=1}^{t-1}\prod_{k=1}^{r}\sigma_{i,k}^{-1}\cdot\prod_{k=1}^{r-1}\sigma_{t,k}^{-1},g_2)$$
$$=\prod_{i=1}^{t-1}\prod_{k=1}^{r}e(H_1(i\|pk_i\|m_{i,k}),pk_i)\cdot\prod_{k=1}^{r}e(\omega_k,cpk)\cdot$$
$$e(\prod_{i=1}^{t-1}\prod_{k=1}^{r}\sigma_{i,k}^{-1}\cdot\prod_{k=1}^{r-1}\sigma_{t,k}^{-1},g_2)=e(w_k,cpk)\quad(6)$$

显然，$\sigma_{t,r}$ 是消息在公钥 $cpk=pk_t\cdot g_2^{a}=g_2^{x_t+\alpha}$ 下合法的 co-CDH 签名，其哈希值为 $\omega_r=h\cdot\Psi(g_2)^{\beta_r}$，B 可以返回伪造签名 h^{x_t}：

$$h^{x_t}\leftarrow\sigma_{t,r}\cdot(\Psi(pk_t)^{\beta_r}\cdot h^{\alpha}\cdot\Psi(g_2)^{\alpha\beta})^{-1}\quad(7)$$

以上过程描述了模拟器 B 伪造签名的过程。显然，B 在群(G_1,G_2)下以概率 ξ' 解决了 co-CDH 问题。

综合上述，与 Di Ma 的签名方案相比，新方案继承了 FssAgg^m 方案[3]的主要特点：在产生签名之前，把每个时段的多个消息聚集成一个大消息，这样密钥的更新频率完全可以基于密钥泄漏的概率来计算。同时，新方案主要有两个改进的方面：一方面，在每个时段里，多个消息可以聚集成若干个大消息，然后对这些大消息产生相应的签名。因此，可以把每个时段分成若干个时间点，中间传感器相对及时地从末端传感器节点接收签名，这样满足了消息签名的实时性要求。另一方面，独立的签名算法和聚集算法允许不同的主体同时参与调用签名算法和聚集算法，满足这两个算法的并发性，提高了整个方案的效率。

参考文献

[1] MA D, TSUDIK G. Forward-secure sequential aggregate authentication. In Proceedings of IEEE Symposiumon Security and Privacy, California, May 20-23, 2007.

[2] MA D. Practical forward secure sequential aggregate signatures. In Proceedings of ACM ASIACCS'08, Tokyo, March 18-20, 2008. New York: ACM, 2008.

[3] BONEH D, LYNN B, SHACHAM H. Short signatures from the Weil pairing. In Proceedings of the Advances in Cryptology. ASIACRYPT'0 1, Australia, December9-1 3, 2001. Springer-Verlag. 2001.

(收稿日期：2010-05-23)

作者简介：

王琴竹，女，1976年生，硕士研究生，讲师，主要研究方向：数据挖掘技术。