一种基于消息分割的一次签名方案①

One – Time Signature Based On Message Division

彭维平　李子臣　刘　辉　（河南理工大学 计算机科学与技术学院 河南 焦作 454003）

摘　要: 一次签名依赖于没有陷门的单向函数达到了很高的安全性,在在线/离线签名、带前向安全性的数字签名以及广播授权协议等各个领域获得了广泛的应用。分析表明现存的一次签名方案签名太长且实现过程相对复杂,在一定程度上限制了其广泛应用。为了克服这些缺陷,在保证签名安全性的前提下,本文提出一种改进的基于消息处理的一次签名方案,利用不等式对消息进行分割,缩短签名的长度,使一次签名实现起来更加高效。通过对此改进的一次签名方案的安全性和性能进行分析,证明此方案是可行的。

关键词: 一次签名 单向函数 数据压缩 不等式 消息分割

1　引言

Rabin[1] 和 Lamport[2] 分别于 1978 年和 1979 年提出了一次签名的思想。一次签名是一种特殊的数字签名方案,其基本思想是利用任何单向函数对消息进行签名。这种签名方案在对一条消息签名时,安全性特别高,而对多条消息签名时是不安全的。二十多年来,许多专家、学者在两者的理论基础上提出了许多改进的方案,使得一次签名方案进一步完善,在在线/离线签名[3]、带前向安全性的数字签名[4]、广播授权协议[5]等各个领域获得了广泛的应用。一次签名相对于普通签名的最主要优势就在于签名的安全性依赖于没有陷门的单向函数,并且可以利用 SHA – 1[8] 等快速哈稀函数来实现。

现存的一次签名方案有两个主要的缺陷:一是签名太长,二是实现过程相对复杂。因而在一定程度上限制了其推广应用。为了克服这些缺陷,在保证签名安全性的前提下,本文提出了一种改进的基于消息处理的一次签名方案,利用不等式对消息进行分割,缩短签名的长度,使一次签名实现起来更加高效。通过对此改进的一次签名方案的安全性和效率进行分析,证明此方案是可行的。

2　签名方案缺陷分析

在 Lamport[2] 提出一次签名方案后,许多人针对他的方案进行了一些改进,大多数人的方案中签名方案的签名长度和密钥的个数都取决于等式 $2b \le C_t^k$ (其中 b 表示消息的位数,t 表示公开的密钥数,k 表示应用于签名的私钥的个数)。下面给出一种这样的方案,包括三个算法:密钥生成、签名和验证。

假设要对一条 b 位长度的消息 m 进行签名。选择合适的整数 t,k 使其符合关系 $2b \le C_t^k$,用 T 表示集合 {1,2,3,...,t} ,T_k 表示含 k 个元素的 T 的子集,S 函数表示从 {1,2,3,...,2b – 1} 到的一一映射,所以 S(m) 能够标识唯一一个 Tk 的 k 个元素的子集。让 f 是一个操作在长度为 a(a 是一个安全的参数)比特的字符串上的单向函数。

(1) 密钥的产生:用密钥产生器随机选择 t 个长度为 a 的字符串 s_i (i = 1,2,...t),把 s_i 当作私钥。计算公钥 $v_i = f(s_i)$。

(2) 签名过程:把 m 看作是 0 到 2b – 1 的一个整数,并且计算 $S(m) = \{i_1, \ldots, i_k\} \in T_k$, $S_{i_1}, \ldots, S_{i_k}$ 就是消息对 m 的签名。

(3) 验证过程:验证签名 $(s'_1, \ldots, s'_k)$ 是否合法,验

① 基金项目:国家自然科学基金资助项目(90104032);河南省科技攻关项目(05424220047);河南理工大学校青年基金资助项目(Q2006 – 57)

证者只需要重新把 m 看做一个 0 到 2b −1 之间的一个整数，计算出 $\{i_1,\ldots,i_k\}$ 作为 T_k 的第 m 个含 k 个元素的子集。验证是否，$f(s_1')=v_{i_1},\ldots,f(s_t')=v_{i_t}$，若结果成立就是合法的签名，否则就是伪造签名。

从以上方案可以看出，由于 m、t、k 必须符合关系式 $2b\le C_t^k$，故消息 m 的长度和 t、k 的长度成正比，即消息 m 越长，签名的长度和密钥的长度也越长。另一方面，针对 S 函数，有很多人设计了不同的 S 函数，但实现起来过于复杂。为了尽可能的使签名的长度比较短，所用到密钥比较少，并且，不再涉及到复杂的 S 函数，我们提出以下这个改进的一次签名方案。

3　改进的一次签名方案描述

为了尽可能的减小签名的长度，首先对长为 b 比特的消息 m 进行压缩。目前有很多数据压缩方案，为了使压缩的效率更高，在此使用散列函数对数据进行压缩处理，从而使任意长度的消息被压缩成相同长度的消息；另一方面，为了进一步减少密钥的数量，舍弃原方案中必须用到的 s 函数。

本文所提出的方案依赖于不等式 $b'\le 2^1+\ldots+2^n$（表示压缩后的数据的长度）。将压缩后的数据 m′分割成 $m_1'\cdots m_n'$ 共 n 部分，每部分的数据长度分别是 2^1bit、2^2bit、…、2^bit，签名时把每一部分对应一个整数，则一条消息签名后的长度就是 n，而公钥的长度就是。

具体方案描述如下：

3.1 消息压缩

对要签名的消息，特别是比较大的消息，签名之前使用 SHA −1 等散列函数对其进行压缩处理，使不同长度的消息压缩成相同长度的消息。假如对消息 m 进行压缩，输出结果为 m′（m′是一条 160 比特的消息）。则：

$$m'=hash(m)$$

3.2 消息分割

根据 $160\le 2^1+2^2+\ldots.+2^n$ 求出最小的 n，将消息分割成 n 个部分，每一部分的长度是 2^i（i =1,2…n）比特，其中如果第 n 部分的比特数小于 2^n，则在其后面补 0，使其成为 2^n 比特的数据。从而得到：

3.3 产生密钥

利用密钥生成器随机生成 2^n-1 个长度为 l（l 是一个安全的参数）的随机字符串，$s_1,\ s_2\cdots,\ s_{2^n-1}$，作为私钥，利用单向函数 f 求出公钥 v_i。

$$v_i=f(s_i)$$

3.4 签名

根据二进制与十进制之间的转换关系，计算每个 m_i'（i=1，2，… t）对应的十进制整数 j_i（i=1，2，… t）。则 s_{ji} 就是对 m_i' 的签名，也就是对原消息 m_i 的签名。

3.5 验证

假设验证者需要对签名，s_1'，s_2'，…，s_n'，进行验证签名的合法性，验证者必须和签名者一样对消息 m 进行同样的压缩和分割，得到 $m'=m_1'm_2'\ldots m_n'$，并计算各个的值，如果 $v_{j_i'}=f(s_i')$ 就是合法的签名，否则验证失败。

4　方案分析

4.1 安全性分析

我们提出的方案基于不等式 $b'\le 2^1+\ldots+2^n$ 对消息的分割。由于把消息分割成了 2^1bit、2^2bit、…、2^bit，对任何 n −1，第 i 部分表示的数值范围比第 j 部分要小，所以把消息分成 n 部分后，前 n −1 部分对应的十进制整数都在第 n 部分表示的范围之内，一个消息唯一标明了 0 至 2^n-1 之间 n 个可重复整数的有序序列，对消息 m 的签名 $s_{j_i}(i=1,2,\ldots,n)$，第三方不可能伪造一个不同的消息，使其和 m 有相同的签名。

另外，本文的一次签名方案仍然建立在单向函数的基础上，私钥是随机选择的，由单向函数的性质可知，根据公钥是不能够推算出私钥的。这就保证第三方不能以原始签名者的名义伪造一个有效的签名。

4.2 性能分析

本方案中利用了散列函数 SHA −1，使任何长度的消息被处理成 160 比特长的消息。对 160 比特的消息进行签名比对未处理的消息签名其长度要短的多，特别是对本文提出的消息分割理论提供了方便，更进一步缩短了签名。例如：对一个 6 比特的消息进行签名，在据传统方案中，根据 $2^6\le C_8^4$，签名长度是 4，公钥数是 8。而利用本文提出的方案 $6\le 2^1+2^2$，签名长度是 2，公钥数是 3，签名长度更短。

另一方面，签名时只需要计算分割后的各部分对应的十进制的整数值，比使用复杂的一一映射函数效率要高，大大简化了一次签名方案。（下转第 74 页）

(上接第76页)

5 结束语

本文通过对现存的一次签名方案分析,针对其签名太长和实现复杂的的缺点,从消息处理的角度,提出一种基于不等式$b' \le 2^1 + \ldots + 2^n$对消息进行特殊分割的一次签名方案,有效克服了现存方案中签名太长和实现复杂的缺点;同时本方案依然建立在单向函数的安全性基础上,相比于以往提出的基于单向函数的签名方案,签名更短,并且舍弃了原来方案中必须用到的一一映射函数,实现起来更加高效。不过,由于验证者和签名者一样必须对消息进行压缩和分割,并计算分割后各部分的值,从而使验证过程要比以往的方案复杂。

参考文献

1 Rabin M O. Digital signatures. Foundations of Secure Communication, Academic Press, 1978. 155 – 168.

2 Lamport L. Constructing digital signature from a one way function. Technical Report CSL – 98, SRI International, 1979.

3 Even S, Goldreich O and Micali S. On – line/off – line digital signatures, Journal of Cryptology, 1996(9): 35 – 67.

4 Abdalla M and Reyzin L. A new forward – secure digital signature scheme, Advancein Cryptology – Asicarypt' 00, LNCS, 1976, 2000: 116 – 129.

5 Perrig A. The BiBa one – time signature and broa – dcast authenticalation. Eighth ACM Conference on Computerand Comunication Security, ACM, 2001, 28 – 37.

6 Ming – Hsin Chang, s Yi – Shiung Yeh. Improving Lamport one – time signatueres Scheme. Applied Mathematic and Computation, 2005, 167: 118 – 124.

7 Bicakci K, Tsudik G, Tung B. How to construct optimal one – timesignatures. Computer Networks, 2003, 43: 339 – 349.

8 Dobbertin H, Bosselaers A and Preneel B. RIPED – 160: A strengthened version of RIPEMD. In D Gollmann, Fast Software Enerption. Third International Workshop Proceedings. Spring – Verlag, 1996.