Comment
Signature scheme based on discrete logarithm without using one-way hash-function

Chan Yeob Yeun, Chris J. Mitchell and Siaw Lynn Ng

Information Security Group
Royal Holloway, University of London
Egham, Surrey TW20 0EX, UK

Email : {c.yeun,c.mitchell,phah033}@rhbnc.ac.uk

Abstract

The purpose of this Comment is to show that the discrete logarithm based signature scheme proposed by Shao in [1] is subject to homomorphism attacks, despite a claim in [1] to the contrary.

1 Introduction

ElGamal [2] and DSS [3] signature schemes are subject to homomorphism attacks. Briefly, the ElGamal signature scheme works as follows. Let p be a large prime number such that $p-1$ has a large prime factor, and let g be a primitive element modulo p. Any user A has a private key x $(1 < x < p-1)$ and a public key $y = g^x \pmod{p}$. To sign a message m, user A does the following:

(i) Randomly chooses an integer k, $1 < k < p-1$.

(ii) Computes
$$r = g^k \pmod{p}$$

(iii) Computes
$$s = (m - xr)k^{-1} \pmod{p-1}$$

(iv) Sends $\mathrm{Sig}(m) = (r, s)$ as the signature.

He and Kiesler [4] describe the following 'homomorphism attack'. Suppose that, in computing three distinct signatures, the random values k satisfy $k_3 = k_1 + k_2$. An observer of these signatures will be able to deduce this by noting that $r_3 = r_1 r_2$. This will then immediately yield the private key from

$$x = (m_1 s_2 s_3 + m_2 s_1 s_3 - m_3 s_1 s_2) \times (r_1 s_2 s_3 + r_2 s_1 s_3 - r_1 r_2 s_1 s_2)^{-1}.$$

Shao [1] introduces a variant of ElGamal which he claims is immune to this attack. However, we show that this is not the case.

2 Shao's Scheme

We now briefly describe the scheme of Shao. The system parameters known to all users are a large prime modulus p, a prime divisor q of $p-1$, and an integer g with order q, that is, $g^q = 1 \pmod p$, and $1 < g < p$. Any user A has two secret keys x_1, x_2 $(1 < m < p)$, and two public keys:

$$y_1 = g^{x_1}, \;\; y_2 = g^{x_2} \pmod p.$$

The digital signature of a message m $(1 < m < p)$ is the triple (r, s_1, s_2) such that

$$((y_1^{s_1} + m y_2^{s_2}) g^{r+m} \pmod p) \equiv r \pmod q$$

To sign a message m, user A does the following:

(i) Randomly chooses two integers k_1 and k_2, $1 < k_1, k_2 < q$.

(ii) Computes

$$r^* = g^{k_1} + m g^{k_2} \pmod p$$
$$r = r^* \pmod q$$

(iii) Computes

$$s_1 = (k_1 - r - m) x_1^{-1} \pmod q \tag{1}$$
$$s_2 = (k_2 - r - m) x_2^{-1} \pmod q \tag{2}$$

(iv) Sends $\mathrm{Sig}(m) = (r, s_1, s_2)$ as the signature.

3 The Attack

The signer cannot prevent the attackers from computing $g^{k_1}, g^{k_2} \pmod p$ from the corresponding published message m and its signature (r, s_1, s_2), since

$$g^{k_i} = g^{x_i s_i + r + m} = y_i^{s_i} g^{r+m} \pmod p, \; i = 1, 2.$$

Suppose three pairs of session keys (k_1, k_2), (k_1', k_2'), (k_1'', k_2'') were used to generate the signatures (r, s_1, s_2), (r', s_1', s_2'), (r'', s_1'', s_2'') of the messages m, m', m'' respectively as in equations (1) and (2). If $k_1 = k_1' + k_1''$, then this relation can be recognised by the attackers, as

$$g^{k_1} = g^{k_1'} g^{k_1''} \pmod p.$$

Now, we have three linear equations in x_1, k_1, k_1', k_1'' from equation (1):

$$x_1 s_1' = k_1' - r' - m' \pmod q,$$

$$x_1 s_1'' = k_1'' - r'' - m'' \pmod q,$$

and

$$x_1 s_1 = k_1 - r - m \pmod q.$$

From these equations, and knowing that $k_1 = k_1' + k_1''$, one can easily obtain the first part of private key

$$x_1 = \{(r' + r'' - r) + (m' + m'' - m)\} \times (s_1 - s_1' - s_1'')^{-1} \pmod{p}.$$

Similarly, if $k_2 = k_2' + k_2''$, then one can also easily obtain the second part of the private key

$$x_2 = \{(r' + r'' - r) + (m' + m'' - m)\} \times (s_2 - s_2' - s_2'')^{-1} \pmod{p}.$$

4 Conclusion

Contrary to Shao's claim, we have shown that Shao's scheme is vulnerable to homomorphism attacks just like all ElGamal type signature schemes.

The main justification given in [1] for the use of Shao's scheme is its resistance to homomorphism and substitution attacks. Substitution attacks can be avoided by the use of a one-way hash-function, and thus there no longer appears to be any reason to use Shao's scheme.

Although the ElGamal scheme and its variants (e.g. DSS) are subject to homomorphism attacks, such an attack being successful appears to be no more likely than finding a discrete logarithm, as long as the random integer used to construct the signature is chosen at random.

References

[1] Z. Shao. "Signature scheme based on discrete logarithm without using one-way hash-function". *Electronics Letters*, 34(11):1079–1080, 1998.

[2] T. ElGamal. "A public key cryptosystem and a signature scheme based on discrete logarithms". *IEEE Transactions on Information Theory*, 31:469–472, 1976.

[3] "The digital signature standard proposed by NIST". *Communications of the ACM*, 35(7):36–40, 1992.

[4] J. He and T. Kiesler. "Enhancing the security of ElGamal's signature scheme". *IEE Proc. Digit. Tech.*, 141(4):249–252, 1994.