Universal Forgeability of a Forward-Secure Blind Signature Scheme Proposed by Duc et al.

Lihua Liu† Zhengjun Cao‡

†Department of Mathematics, Shanghai Jiaotong University.

‡ Center of Information Security, Academy of Mathematics and Systems Science,

Chinese Academy of Sciences, Beijing, P.R. China. 100080

Abstract Duc et al. proposed a forward-secure blind signature scheme in [1]. They claimed that the scheme is constructed from the provably secure Okamoto-Guilou-Quisquater blind signature scheme. But we recently found that their scheme is insecure. In the paper, we show the scheme is universally forgeable by a simple and direct attack.

Keywords Blind signature, Universal forgeability.

1 Introduction

Some public key cryptosystems, such as RSA, Rabin[2] and so on, can be used to sign digital signatures. Without the private key, no one can forge a legal signature. Therefore, digital signatures are widely used to prove the integrity of data and the identity of signee. However, in some applications, such as electronic cash systems or anonymous electronic voting systems, in order to protect the privacy of users, the anonymity property is necessary. Hence, in 1982, Chaum invented a blind signature scheme[3], which not only achieves the unforgeability property but also achieves the unlinkability property. The protocol is briefly described as below. When a requester sends a blind message to request his signature from the signee, the signee signs the blind message and sends the result to the requester. Then, the requester can obtain the signature of the chosen message from performing the unblinding function. The signature can be verified, but the signee can not link the relationship between the blind message and the signature of the chosen message. A secure blind signature scheme must satisfy the unforgeability property and the unlinkability property.

Clearly, the ability to sign must be available to the signer only. In practice, it is very difficult to guarantee that secret keys cannot be compromised since many implementation and

0‡ Corresponding author's e-mail address: zjcamss@hotmail.com

administration errors can be exploited. To relax the problem, an intuitive solution is to use many secret keys each valid only within a short period of time and preferably keeps the public key unchanged over its lifetime. Such strategy is called key evolution.

The notion of forward secrecy was introduced by Anderson[4]. Duc et al. proposed a forward-secure blind signature scheme in [1]. They claimed that the scheme is constructed from the provably secure Okamoto-Guilou-Quisquater blind signature scheme. But we recently found that their scheme is insecure. In the paper, we show the scheme is universally forgeable by a simple and direct attack.

2 Review of the blind signature scheme

In this section, we review the forward-secure blind signature scheme. It consists of five algorithms:

$$< FBSIG.Setup, FBSIG.Update, FBSIG.Signer, FBSIAG.User, FBSIG.Verify > .$$

algorithm FBSIG. Setup (k)
Generate randomly two safe primes p and q of length $k/2$ bits
$N \leftarrow pq$
$\phi(N) \leftarrow (p-1)(q-1)$
Generate a random number λ such that it is co-prime with $\phi(N)$
Choose a from Z_N^* of order greater than λ
Choose $r_0 \in_R Z_\lambda^*, s_0, e \in_R Z_N^*$
$V \leftarrow a^{-r_0} s_0^{-\lambda} \ \ mod\ N$
$f_1 \leftarrow a^e \ \ mod\ N$
$v_1 \leftarrow V^2 a^e \quad \mathrm{mod} \quad N$
$l \leftarrow (2r_0 - e) \div \lambda$
$r_1 \leftarrow (2r_0 - e) \ \ mod\ \lambda$
$s_1 \leftarrow a^l s_0^2 \ \ mod\ N$
Erase p, q, e, r_0, s_0, and $\phi(N)$
$SK_1 \leftarrow (1, r_1, s_1, v_1, f_1)$
$PK \leftarrow (N, a, V, \lambda)$
RETURN (PK, SK_1)

algorithm FBSIG.Update (SK_i)
$(i, r_i, s_i, v_i, f_i) \leftarrow SK_i$
Choose $e \in_R Z_N^*$
$v_{i+1} \leftarrow v_i^2 a^e \ \ mod\ N$
$f_{i+1} \leftarrow f_i^2 a^e \ \ mod\ N$
$l \leftarrow (2r_i - e) \div \lambda$
$r_{i+1} \leftarrow (2r_i - e) \ \ mod\ \lambda$
$s_{i+1} \leftarrow a^l s_i^2 \ \ mod\ N$

$SK_{i+1} \leftarrow (i+1, r_{i+1}, s_{i+1}, v_{i+1}, f_{i+1})$
Erase SK_i, e, l
RETURN (SK_{i+1})

Note that, i, v_i, f_i of SK_i are not secret anyway. We prefer to keep PK unchanged to avoid confusion because if public key is changed, we need to perform public key revocation. The signature issuing protocol is given as follows:

algorithm FBSIG.Signer (SK_i)	algorithm FBSIG.User (PK, m)
On Error RETURN 'incomplete'	On Error RETURN $\perp$
$(i, N, \lambda, a, r_i, s_i, f_i) \leftarrow SK_i$ Choose $t \in_R Z_\lambda^*$ Choose $u \in_R Z_N^*$ $x \leftarrow a^t u^\lambda \ \ mod\ N$ Send x to FBSIG.User	
	Get x from FBSIG.Signer $(N, \lambda, a, V) \leftarrow PK$ Choose blinding factors $\alpha, \gamma \in_R Z_\lambda^*$ and $\beta \in_R Z_N^*$ $x' \leftarrow x a^\alpha \beta^\lambda v_i^\gamma \ \ mod\ N$ $c' \leftarrow H(i \parallel f_i \parallel m \parallel x')$ $c \leftarrow (c' - \gamma) \ \ mod\ \lambda$ Send c to FBSIG.Signer
Get c from FBSIG.User $y \leftarrow (t + c r_i) \ \ mod\ \lambda$ $\omega \leftarrow (t + c r_i) \div \lambda$ $z \leftarrow a^\omega u s_i^c \ \ mod\ N$ Send y, z to FBSIG.User	
	Get y, z from FBSIG.Signer $y' \leftarrow (y + \alpha) \ \ mod\ \lambda$ $\omega' \leftarrow (y + \alpha) \div \lambda$ $\omega'' \leftarrow (c' - c) \div \lambda$ $z' \leftarrow a^{\omega'} v_i^{-\omega''} z \beta \ \ mod\ N$ $\sigma(m) \leftarrow (f_i, c', y', z',)$
RETURN 'complete'	RETURN $(i, \sigma(m))$

(We denotes $\div$ by a division operation which gives the result as the quotient of the division (i.e., if $a = qb + r$ then $a \div b = q$).)

algorithm FBSIG.Verify $(m, i, \sigma(m), PK)$

$$(N, \lambda, a, V) \leftarrow PK$$
$$(f_i, c', y', z') \leftarrow \sigma(m)$$
$$v_i \leftarrow V^{2^i} f_i \ \ mod \ N$$
$$x'' \leftarrow a^{y'} z'^{\lambda} v_i^{c'} \ \ mod \ N$$

If $c' = H(i \parallel f_i \parallel m \parallel x'')$ then RETURN 'accept' else RETURN 'reject'.

Correctness:

$$\begin{aligned} x'' &= a^{y'} z'^{\lambda} v_i^{c'} = a^{y'} z'^{\lambda} (V^{2^i} f_i)^{c'} \\ &= a^{y'} (a^{\omega'} v_i^{-\omega''} z\beta)^{\lambda} v_i^{c'} = a^{y' + \omega' \lambda} (a^{\omega} u s_i^{c})^{\lambda} \beta^{\lambda} v_i^{c' - \omega'' \lambda} \\ &= a^{y + \omega\lambda + \alpha} u^{\lambda} s_i^{c\lambda} \beta^{\lambda} v_i^{c' - \omega''\lambda} = a^{t + cr_i + \alpha} u^{\lambda} s_i^{c\lambda} \beta^{\lambda} v_i^{c' - \omega''\lambda} \\ &= x a^{\alpha} \beta^{\lambda} v_i^{c' - c - \omega''\lambda} = x a^{\alpha} \beta^{\lambda} v_i^{\gamma} = x' \ \ (mod \ N) \end{aligned}$$

3 Universal forgeability

In this section, we present a simple and direct attack on the scheme. It shows that the scheme is universally forgeable.

Given public keys (N, λ, a, V) of signer and an arbitrary massage m, Adversary only needs to choose three random numbers α, β, γ, and computes

$$f_i = V^{-2^i} a^{\alpha} \ mod \ N, \qquad z' = \gamma \ mod \ N$$

$$c' = H(i \parallel f_i \parallel m \parallel a^{\beta}\gamma^{\lambda}), \qquad y' = \beta - \alpha c' \ \ mod \ \lambda$$

Hence, he obtains a valid blind signature $\sigma(m) = (f_i, c', y', z')$ for m.

Correctness:

$$\begin{aligned} x'' &= a^{y'} z'^{\lambda} (V^{2^i} f_i)^{c'} \\ &= a^{\beta - \alpha c'} \gamma^{\lambda} (V^{2^i} V^{-2^i} a^{\alpha})^{c'} \\ &= a^{\beta} \gamma^{\lambda} \ \ (\ mod \ N) \end{aligned}$$

In fact, the challenge in the scheme doesn't work. Adversary can easily shun it. This is a serious design error.

4 Conclusion

In this paper, we presented a simple and direct attack on a forward-secure blind signature scheme. Our results show that the scheme is universally forgeable.

References

[1] Dang Nguyen Duc, Iung Hee Cheon, Kwangjo Kim. A forward-secure blind signature scheme based on the strong RSA Assumption. Information and Communications Secureity'2003. Springer-Verlag, 2003. ICICS 2003, pp. 11-21.

[2] M. O. Rabin. Digitalized signatures and public key functions as intractable as factorization. Techbical Report, MIL/LCS/TR212, MIT Lab. Computer Science, Cambridge, Mass., January 1979.

[3] D.Chaum. Blind signature for untraceable payments. Advances in Cryptology: Crypto'82, pp. 199-203, 1982.

[4] Ross Anderson. Two remarks on public key cryptography. Invited Lecture, Fourth Annual Conference on Computer and Communications Security. ACM, 1997.