Cryptanalysis on an Algorithm for Efficient Digital Signatures

Fuw-Yi Yang

Department of Computer Science and Information Engineering
Chaoyang University of Technology, Taichung County 41349, Taiwan
yangfy@ms7.hinet.net

Abstract. The total computation of the generation and verification of personnel identification or digital signature is heavy. For many schemes of them, the total computation is not less than hundreds of modular multiplications. Efficient schemes of personnel identification and digital signature were proposed, which require no more than 10 modular multiplications on generation and verification of challenge-response or digital signature. However, the schemes are weak in security. The paper will show that by interception of a transcript of communications between the prover and verifier, the private key of the prover is revealed.

1 Introduction

In daily life we write our name down on a document to give a handwritten signature on the content of that document. The digital counterpart to a handwritten signature is digital signature. It is an important primitive operation in public key cryptosystems. Many applications in information security require digital signatures, entity authentication, data integrity, and proof of non-repudiation, for example.

Since the emergence of public key cryptography, many schemes of digital signature have been proposed, such as RSA-based signature schemes [1, 2, 3] and DL-based (Discrete Logarithm) signature schemes [4, 5, 6]. Generally, a signer (prover) signed on a message (document) and then sent receiver (verifier) the digital signature (may be along with the message). In the setting of public key cryptosystems, heavy computation is required in signature generation, verification, or both. The following table summarizes computational cost of signature generation and signature

verification, and other information. Two signature schemes are listed in different columns, one is RSA-based and the other one is DL-based. To quantify computation, MM is used to denote the costly operation of modular multiplication.

	RSA signature scheme $\|n\| = \|d\| = 1024$, $e = 3$	Schnorr signature scheme $\|p\| = \|y\| = 1024$, $\|q\| = 160$, $g = 2$
Signature size	1024 bits	320 bits
Size of private key	1024 bits	160 bits
Size of public key	1024 bits (only n counted)	2208 bits (only p, y, and q counted)
Signer's computations	1536 MMs $(1.5 \cdot \|d\|)$	240 MMs $(1.5 \cdot \|q\|)$
Verifier's computations	2 MMs	480 MMs
Total computation (Signer + Verifier)	1538 MMs	720 MMs

The signer/verifier may be a host computer, a mobile computer, a pocket device or a smart card. Usually the latter three entities are powered by battery, which implies that they have limited processing capability. In order to reduce the processing burden of signer/verifier, many methods have been developed. Short public keys (*e.g.* $e = 3$) are used in RSA-based signature schemes to verify signatures efficiently; many researchers [5, 7, 8, 9] have devoted to reduce the complexity of online computation during the signing phase. However the total computation (computational cost of generation and verification signature) still requires hundreds of MMs or more.

Using diophantine solutions for the Pythagorean triplets, the work in [10] proposes efficient schemes of personnel identification and digital signature (Method 2). Let's call it M2-scheme. The schemes' total computation is less than 10 MMs. The extremely low computational requirement seems attractive especially for portable systems.

However, the paper will show that weakness exists in M2-scheme. By the transcript of communications between a prover and a verifier, the prover's private key can be calculated.

2 M2-scheme reviewed

Two operation modes are permitted in [10]. The section only reviews the trusted party mode. Let n be the product of large primes p and q. Their bit lengths may be $|p| = |q| = 512$, for example. In order to register at a trusted party, the user should show his/her credentials to the party. From the supplied credentials, the trusted party generates an identity string $I(s)$ according to a fixed and widely published format, *e.g.* $I(s) \in Z_n^*$.

In the trusted party mode, the M2-scheme has designed such that the modular product of public key (Puk) and modulo square of private key (Prk) equals one, *i.e.* $(Puk) \cdot (Prk)^2 = 1 \bmod n$. In the finite group Z_n^*, it may be failure to find the square roots of $I(s)$. Then a short random string (s_r) is concatenated with $I(s)$ such that the resultant string is in the set of quadratic residue modulo n, namely, $I(s) \parallel s_r \in QR_n$. Now, the user's public key is the concatenation of $I(s)$ and s_r, and private key is the reciprocal of the square root of Puk modulo n.

A digital signature scheme can be converted from a personnel identification scheme using the Fiat-Shamir transformation [11]. The following only examines the personnel identification scheme. Assume that a prover wants to prove his identity to a verifier. On receipt of verifier's challenge string c, the prover computes strings p_1 and p_2 as response. The strings are created using the challenge string c, a random number u, and the prover's private key Prk. After receiving prover's response, the verifier will verify whether the response is valid. A valid response confirms verifier the identity of the prover. The detailed interactions between the prover and verifier are as follows.

M2-scheme: *Prover*(Prk)		M2-scheme: *Verifier*(Puk)
// $Prk = 1 / (Puk)^{1/2} \bmod n$	$\xleftarrow{c}$	// $Puk = I(s) \parallel s_r$ $c \leftarrow Z_n^*$
$u \leftarrow Z_n^*$, $v \leftarrow c / (2 \cdot u) \bmod n$ $p_1 \leftarrow u + v \bmod n$ $p_2 \leftarrow Prk \cdot (u^2 - v^2) \bmod n$	$\xrightarrow{p_1, p_2}$	$A = Puk \cdot p_2^2 \bmod n$ $B = p_1^2 \cdot (p_1^2 - 2 \cdot c) \bmod n$ If $A = B$ then accept else reject

3 Cryptanalysis of the M2-scheme

Let n be the product of large primes p and q. It is hard to find square roots of $x \in QR_n$. However, the congruence $x^2 + k \cdot y^2 = m \bmod n$ is solved by the probabilistic algorithm in [12] with polynomial time, without the knowledge of the factorization of the modulus n. Based on the probabilistic algorithm, the prover's private key can be computed. The details are as follows.

1. Obtain the transcript (c, p_1, p_2) of the communications between the prover and verifier.
2. Compute the quantity $m = Puk \cdot {p_2}^2 + c^2 / 2 = u^4 + v^4 \bmod n$.
3. Apply the probabilistic algorithm to the congruence $u^4 + v^4 = m \bmod n$. The values u^2 and $v^2 \bmod n$ are computed in polynomial time.
4. The prover's private key is solvable, *i.e.* $Prk = p_2 / (u^2 - v^2) \bmod n$.

4 Conclusion

This paper has shown that M2-scheme is weak against active attack. The prover's private key is computable using the information leaked from the transcript of challenge-response.

References

1. R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, Vol. 21, No. 2, pp. 120-126, 1978.
2. M. Bellare and P. Rogaway, "The exact security of digital signature: How to sign with RSA and Rabin," *Advances in Cryptology-EUROCRYPT'96*, LNCS 1070, pp. 399-416, 1996.
3. J. S. Coron, "On the exact security of full-domain-hash," *Advances in Cryptology-CRYPTO'00*, LNCS 1880, pp. 229-235, 2000.
4. T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inform. Theory*, IT-31, (4), pp. 469-472, 1985.
5. C. P. Schnorr, "Efficient signature generation for smart cards," *Journal of Cryptology*, Vol. 4, pp. 161-174, 1991.

6. D. Pointcheval and J. Stern, “Security arguments for digital signatures and blind signatures,” *Journal of Cryptology*, Vol. 13, No. 3, pp. 361-396, 2000.
7. G. Poupard and J. Stern, “Security analysis of a practical ‘on the fly’ authentication and signature generation,” *Advances in Cryptology-EUROCRYPT’98*, LNCS 1403, pp. 422-436, 1998.
8. G. Poupard and J. Stern, “On the fly signatures based on factoring,” *Proceedings of the 6th ACM Conference on computer and communications security* (*CCS*), pp. 48-57, 1999.
9. T. Okamoto, M. Tada and A. Miyaji, ”Efficient ‘on the fly’ signature schemes based on integer factoring,” *Proceedings of the 2nd International Conference on Cryptology in India, INDOCRYPT’01*, LNCS 2247, pp. 275-286, 2001.
10. D. Ramesh, “A twin algorithm for efficient generation of digital signatures,” *Proceedings of the 2nd International Conference on Cryptology in India, INDOCRYPT’01*, LNCS 2247, pp. 267-274, 2001.
11. A. Fiat and A. Shamir, “How to prove yourself: practical solutions to identification and signature problems,” *Advances in Cryptology-CRYPTO’86*, LNCS 263, pp. 186-194, 1986.
12. J. M. Pollard and C. P. Schnorr, “An efficient solution of congruence $x^2 + ky^2 = m \pmod{n}$, ” *IEEE Transactions on Information Theory*, IT-33, No. 5, pp. 702-709, 1987.