A NEW SIGNATURE PROTOCOL BASED ON RSA AND ELGAMAL SCHEME

J. Ettanfouhi and O. Khadir

Laboratory of Mathematics, Cryptography and Mechanics, Fstm,
University Hassan II of Casablanca, Morocco

ABSTRACT

In this paper, we present a new signature scheme based on factoring and discrete logarithm problems. Derived from a variant of ElGamal signature protocol and the RSA algorithm, this method can be seen as an alternative protocol if known systems are broken.

KEYWORDS

Factoring, DLP, PKC, ElGamal signature scheme, RSA.

MSC: 94*A*60

1. INTRODUCTION

In 1977, Rivest, Shamir, and Adleman[6] described the famous RSA algorithm which is based on the presumed difficulty of factoring large integers. In 1985, ElGamal [2] proposed a signature digital protocol that uses the hardness of the discrete logarithm problem[5 p.116, 7 p.213, 8 p. 228]. Since then, many similar schemes were elaborated and published[1,3].

Among them, a new variant was conceived in 2010 by the second author[4].In this work, we apply a combination of the new variant of Elgamal and RSA algorithm to build a secure digital signature. The efficiency of the method is discussed and its security analyzed.

The paper is organised as follows: In section 2, we describe the basic ElGamal digital signature algorithm and its variant. Section 3 is devoted to our new digital signature method. We end with the conclusion in section 4.

In the paper, we will respect ElGamal work notations [3]. $\mathbb{N}$, $\mathbb{Z}$ are respectively the sets of integers and non-negative integers. For every positive integer n, we denote by $\mathbb{Z}/n\mathbb{Z}$ the finite ring of modular integers and by $(\mathbb{Z}/n\mathbb{Z})^*$ the multiplicative group of its invertible elements. Let a, b, c be three integers. The GCD of a and b is written as $gcd(a,b)$. We write $a \equiv b \quad 0.1cm[c]$ if c divides $a-b$, and $a = b \bmod c$ if a is the rest in the division of b by c. The bit length of n is the number of bits in its binary model, with n an integer .We start by presenting the basic ElGamal digital signature algorithm and its variant:

DOI: 10.5121/ijitmc.2016.4302

2. ELGAMAL SIGNATURE SCHEME

In this section we recall ElGamal signature scheme[2] and its variant[4].

1. Alice chooses three numbers:
- p , a large prime integer.
- α , a primitive root of the finite multiplicative group $(\mathbb{Z}/p\mathbb{Z})^*$
- x , a random element of $\{1,2,..., p-1\}$

2. She computes $y = \alpha^x \mod p$. Alice's public key is (p, α, y), and x is her private key.

3. To sign the document m , Alice must solve the problem:

$$\alpha^m \equiv y^r r^s \ [p] \tag{1}$$

where r, s are the unknown variables.
Alice fixes arbitrary r to be $r = \alpha^k \mod p$, where k is chosen randomly and invertible modulo $p-1$. Equation (1) is then equivalent to:

$$m \equiv xr + ks \quad [p-1] \tag{2}$$

Since Alice has the secret key x , and as the number k is invertible modulo $p-1$, she calculates the other unknown variable s by

$$s \equiv \frac{m - xr}{k} \quad [p-1] \tag{3}$$

4. Bob can verify the signature by checking if congruence (1) is valid for the variables r and s given by Alice.

3. VARIANT OF ELGAMAL SIGNATURE SCHEME

We present a variant of ElGamal digital signature system.

This variant **Error! Reference source not found.** is based on the equation:

$$\alpha^t \equiv y^r r^s s^m \ [p] \tag{4}$$

r, s, t are the unknown parameters, and (p, α, y) are Alice public keys. p is an integer (a large prime). α is a primitive root of $\mathbb{Z}_p^*$. y is calculated by $y = \alpha^x \mod p$. x is a random element of {1,2,...,p-1}.

Let $m = h(M)$, where h is a hash function, and M the message to be signed by Alice.
To give the solution (3), she fixes randomly r as $r \equiv \alpha^k mod\ p$, and s to be $s \equiv \alpha^l\ mod\ p$, where k, l are selected arbitrary in {1,2,...,p-1}.

Equation (3) is then equivalent to:

$$t \equiv rx + ks + lm[p-1] \tag{5}$$

as Alice recognize the values of r, s, k, l, m, x, she is able to calculate the last unknown variable t.
Bob verify the signature by verifying the congruence (4).
this system does not use the extended Euclidean algorithm for calculating $k^{-1}\ mod\ (p-1)$.
We clarify the scheme by the example given by the creator of this alternative[4].

3.1 EXAMPLE

Let (p, α, y) be Alice public keys where: $p = 509$, $\alpha = 2$ and $y = 482$. We assert that we are not confident if using a small value of α does not abate the protocol. The private key is $x = 281$. Suppose that Alice wants to generate a signature for the document M for which $m \equiv h(M) \equiv 432[508]$ with the exponents $k = 208$ and $l = 386$ are randomly taken. She computes $r \equiv \alpha^k \equiv 2^{208} \equiv 332[p]$, $s \equiv \alpha^l \equiv 2^{386} \equiv 39[p]$ and $t \equiv rx + ks + lm \equiv 440[p-1]$.

Bob or anyone can verify the relation $\alpha^t \equiv y^r r^s s^m [p]$. Indeed, we find that $\alpha \equiv 436[p]$ and $y^r r^s s^m \equiv 436[p]$.

4. OUR PROTOCOL

4.1 DESCRIPTION

In this section, we describe our new digital signature. The protocol is based simultaneously on two hard problems.

We assume first that h is a public secure hash function like SHA1[5 p.348, 7 p.242, 8 p.133].

We suppose that Alice public keys are (P, α, y, e) where:
- $P = 2pq + 1$, p, q are three primes.
- α, a primitive root of the multiplicative group $(\mathbb{Z}/p\mathbb{Z})^*$.
- $y = \alpha^x\ mod\ P$, where x is the private key of Alice, which is randomly taken in $\{1,2,..., P-1\}$.

- Element e is the public exponent in the RSA cryptosystem.

We propose the following protocol:

If Alice wants to sign the message M, she must give a solution for the modular equation:

$$\alpha^t \equiv y^{r^e}(r^e)^{(s^e)}(s^e)^m [P] \tag{6}$$

where $m = h(M) \mod p$, and r, s, t are unknown.

To solve equation (5), Alice starts by putting:

$$r' \equiv r^e \quad [P-1] \tag{7}$$

$$s' \equiv s^e \quad [P-1] \tag{8}$$

Equation (5) becomes:

$$\alpha^t \equiv y^{r'} r'^s s'^m \quad [P] \tag{9}$$

Alice uses the new variant of Elgamal algorithm[4] to solve equation (9) and to get the values of r', s' and t.

Then with her RSA private key she solves equations (7) and (8). The cupel r and s is her signature for the message M.

Bob or anybody can check that the signature is valid by replacing r, s and t in relation (5).

4.2 EXAMPLE

Let us illustrate the method by the following example.

Suppose that Alice's public key is: $P = 2*167*313+1 = 104543$, $\alpha = 5$, $y = 23292$, $e = 7$. The private keys for RSA and ElGamal systems are respectively: $x = 9502$, $d = 7399$.

Assume that $m = h(M) = 12345$ is the hashed message that she likes to sign.

If she takes Randomly $k = 845$ and $l = 2561$.

She will find from equation 8 that $r' = 17744$, $s' = 31839$.

Relation 4 implies $t = 57764$.

Alice uses (6) and (7) to obtain:

$$r \equiv r'^{d} \quad [P-1] \equiv 75282$$
$$s \equiv s'^{d} \quad [P-1] \equiv 19005.$$

To verify Bob puts $A = \alpha^{t} \bmod P = 62833$, $B = y^{r^{e} \bmod P-1} \bmod P = 79849$, $C = (r^{e} \bmod P-1)^{(s^{e} \bmod P-1)} \bmod P = 83421$ and $D = (s^{e} \bmod P-1)^{m} \bmod P = 212997$, and checks if $A = B * C * D \bmod P$.

4.3 SECURITY ANALYSIS

Now that we have presented the protocol, we will discuss some possible attacks. Assume that Oscar is Alice's opponent.

ATTACK 1: If the attacker try to imitate the computation made by Alice, he can find r and s, but to find t he needs the value of the private key x to solve equation 4.

ATTACK 2: Suppose Oscar is capable to solve the discrete logarithm problem [2]. He cannot calculate r and s from equation (7) and (8) he will be confronted to the factorisation of a large composite modulus [5,8].

ATTACK 3: Suppose Oscar is capable to solve RSA equations (7) and (8). Oscar cannot get t from equation (9) since x is Alice's secret key. If he tries to get t from equation (4), he will be stopped by the discrete logarithm problem.

4.4 COMPLEXITY OF OUR ALGORITHM

As in [1], let T_{exp}, T_{mult} and T_h be appropriately the time to calculate an exponentiation, a multiplication and hash function of a document M. We neglect the time needed for modular substraction, additions, comparisons and apply the conversion $T_{exp} = 240T_{mult}$.

4.4.1 SIGNATURE COMPLEXITY

To sign the message M, Alice must compute the six parameters:
$m = h(M)\ mod \quad P$, $r' \equiv \alpha^{k} \quad [P]$, $s' \equiv \alpha^{l} \quad [P]$, $r \equiv r'^{d} \quad [P-1]$, $s \equiv s'^{d} \quad [P-1]$, $t \equiv xr' + ks' + lm \ [P-1]$.
Alice needs to perform four modular exponentiations, three modular multiplications and one hash function computation. So the global required time is :

$$T_1 = 4T_{exp} + 3T_{mult} + T_h = 963T_{mult} + T_h$$

4.4.2 VERIFICATION COMPLEXITY

Bob should calculate 4 exponentiations, 2 multiplications and one hash function. So the global required time is :

$$T_2 = 4T_{exp} + 2T_{mult} + T_h = 962T_{mult} + T_h$$

5. CONCLUSION

In this work, we proposed a new signature protocol that can be an alternative if old systems are broken. Our method is based simultanyously on RSA cryptosystem and DLP.

ACKNOWLEDGEMENTS

This work is supported by the MMS e-orientation project.

REFERENCES

[1] R. R. Ahmad, E. S. Ismail,and N. M. F. Tahat , A new digital signature scheme based on factoring and discrete logarithms , J. of Mathematics and Statistics (4): (2008), pp. .

[2] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithm problem, IEEE Trans. Info. Theory, IT-31, (1985), pp. .

[3] L.C. Guillou, J.J. Quisquater, A Paradoxial Identity-based SIgnature Scheme Resulting from Zero-Knowledge, Advances in cryptography, LNCS 403, (1990) pp. .

[4] O. Khadir, New variant of ElGamal signature scheme, Int. J. Contemp. Math. Sciences, Vol. 5, no. 34, (2010), pp. .

[5] A. J. Menezes, P. C. van Oorschot and S. A. Vanstone, Handbook of applied cryptography, CRC Press, Boca Raton, Florida, 1997.

[6] R. Rivest, A. Shamir and L. Adeleman, A method for obtaining digital signatures and public key cryptosystems, Communication of the ACM, Vol. no 21, (1978), pp. .

[7] J. Buchmann, Introduction to Cryptography,(Second Edition), Springer 2000.

[8] D. R. Stinson, Cryptography, theory and practice, second Edition, Chapman & Hall/CRC, 2006.

Authors

Jaouad Ettanfouhi holds an engineer degree in Computer Science from the University of Hassan II of Casablanca (2011). Member of the laboratory of Mathematics, Cryptography and Mechanics, he is preparing a thesis in public key cryptography.

Dr Omar Khadir received his Ph.D. degree in Computer Science from the University of Rouen, France (1994). Co-founder of the Laboratory of Mathematics, Cryptography and Mechanics at the University of Hassan II Casablanca, Morocco, where he is a professor in the Department of Mathematics. He teaches cryptography for graduate students preparing a degree in computer science. His current research interests include public key cryptography, digital signature, primality, factorisation of large integers and more generally, all subjects connected to the information technology.