Information & Security

Marouane Ihia & Omar Khadir
vol. 42 (2019): 117-126
https://doi.org/10.11610/isij.4207

Published since 1998

ISSN 0861-5160 (print), ISSN 1314-2119 (online)

Research Article

Amelioration of ElGamal Digital Signature Schemes

***Marouane Ihia, Omar Khadir* (✉)**

Laboratory of Mathematics, Cryptography, Mechanics and Numerical Analysis, Faculty of Science and Technology, University Hassan II of Casablanca, Morocco
www.univh2c.ma

ABSTRACT:

In this paper, we propose a new way to use ElGamal signature that will allow us to conserve the private key used for signing. This approach is mainly to prevent Known-Messages-Attack against ElGamal signature and its variants. This work is an amelioration of ElGamal digital signature in the group multiplicative, the elliptic curves and their variations.

ARTICLE INFO:

RECEIVED: 08 mar 2018
REVISED: 10 Nov 2018
ONLINE: 12 FEB 2019

KEYWORDS:

Public Key Cryptography, Elliptic Curves, Discrete Logarithm Problem, ElGamal digital signature

Introduction

Most of the historians consider that the public key cryptography appears for the first time in 1976 when Diffie and Hellman published their seminar paper "New directions in cryptography."[2] A method to obtain a common key between two partners who communicate over a public insecure channel was described. It was based on the hardness of the famous discrete logarithm problem. In 1978 Rivest, Shamir and Adleman proposed an algorithm to encrypt and decrypt confidential messages.[11] The technical security relies mainly on hard mathematical problems in number theory and particularly on integer factorization. Until now, their

✉ E-mail: khadir@hotmail.com

discovery is considered as the most important and practical way to protect data. In 1985, ElGamal presented a cryptosystem inspired by the work of Diffie and Hellman.[3] He also suggested a remarkable and secure signature protocol. In 1986, Koblitz[5] and Miller[9] independently showed that elliptic curves over fields offers suitable finite groups for public key cryptography. The new advantage is that, with comparative security level, private and public keys produced by the mean of elliptic curves has size less than that needed in the conventional public key cryptography.[7]

In the last decades, several problems have arisen such as data integrity, user identification[10] and digital signatures.[4] In 1999 Koblitz showed how to apply the same idea of Diffie and Hellman to elliptic curves in order to produce a common key.[6]

In this work we propose an amelioration of ElGamal signature in the group multiplicative, the elliptic curves and their variations.

Our technique uses the randomness of hash functions to reinforce the signature's security. By using the hash of the message and the private key, we guarantee the safety of our hidden parameters. We analyze the security and assess its complexity.

The paper is organized as follows: the second section is a reminder of the groups defined using elliptic curves. The third section contains our contribution. We conclude in the fourth section.

Throughout the sequel we use classical notations: $\mathbb{Z}$, $\mathbb{R}$ are respectively the sets of non-negative integers and real numbers. For every prime integer, we denote by $Fp = Z|pZ$ the field of modular integers with p elements. Let a,b,c be three integers we write $a \equiv b\ [c]$ if c divides the difference $a - b$ and $a = b \bmod c$ if the remainder in the division of b by c. We use the notation $\#S$ to designate the cardinality of a set S.

Let us start by a recalling of the elliptic curve groups construction.

Elliptic Curves

Definition 1. An elliptic curve over $\mathbb{R}$ is the set of points that verified the equation $y^2 = x^3 + ax + b$, where the discriminant $-(4a^3 + 27b^2)$ is not equal to zero. We add to this curve a point at infinity noted $\boldsymbol{O}$.

We recall the additive operation on points over an elliptic curve.

Let $E(F_p)$: $y^2 = x^3 + ax + b$ be an elliptic curve. The sum of two points is defined as follows:

$P = (x_P, y_P)$ and $Q = (x_Q, y_Q) \in E(F_p)$: with P and $Q \neq \boldsymbol{O}$.

We have $-P = (x_P, -y_P)$ and P + Q = R such that:

- If $Q = -P$, then $P + Q = \boldsymbol{O}$
- If $Q \neq -P$, then $P + Q = (x_R, y_R)$
  with $x_R = m^2 - x_P - x_Q$ and $y_R = m(x_P - x_R) - y_R$

where the slope m is given by

$$\text{m} = \begin{cases} \frac{y_Q - y_P}{x_Q - x_P} & if\ x_P \neq x_Q \\ \frac{3x_P^2 + a}{2y_P} & \text{otherwise} \end{cases}$$

Note that in this sum, we never use coefficient b to calculate the coordinates R. For more details, we refer the reader to the works of Buchmann,[1] Koblitz,[6] and Menezes, Van Oorschot, and Vanstone.[8]

Theorem 1.[6] The set $E(\mathbb{R})$, with the binary operation $+$ forms an abelian group whose identity element is the point at infinity $\boldsymbol{O}$.

Definition 2. An elliptic curve over the finite field F_p, where the characteristic is not 2 or 3, is the set of solutions $(x, y) \in F_p^2$ to the equation $y^2 \equiv x^3 + ax + b\ [p]$, where the discriminant $-(4a^3 + 27b^2) \not\equiv 0[p]$. We add to this curve a point at infinity denoted $\boldsymbol{O}$.

The relation giving the sum of two points belonging to elliptic curves over $\mathbb{R}$ remains valid modulo p.

Theorem 2.[5] The set $(E(F_p), +)$ with the binary operation $+$ forms an abelian group whose identity element is $\boldsymbol{O}$.

Since 1985, from Schoof's work, we have:

Theorem 3.[12] The order of elliptic curve that is defined over a finite field $E(F_p)$ is calculated by a deterministic algorithm in a polynomial time.

Definition 3.[14]. A cryptographic hash function H, is a function such that the image of any element of long length gives a result having a smaller fixed length, and should have the following properties:

1. Given a message m, the calculation of $H(m)$ can be done very quickly.
2. H is preimage resistant: Given y, it is computationally infeasible to find m with $H(m) = y$.
3. H is strongly collision-free: It is computationally infeasible to find m_1 and m_2 with $m_1 \neq m_2$ and $H(m_1) = H(m_2)$.

ElGamal Digital Signature in F_p [3]

In 1985 ElGamal proposed a signature based on the difficulty of solving the discrete logarithm problem. He described his scheme as follows:

Let m be a document to be signed, where $0 \leq m \leq p - 1$. To sign a document, Alice chooses a primitive root α, a secret key x and calculates $y \equiv \alpha^x\ [p]$.

The equation of signature is:

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

The signature for m is the pair (r, s), where $0 \leq r, s < p - 1$.

To sign m, Alice does the following:

1. Chooses a random number k, such that $2 \leq k < p - 1$ and
$$gcd(k, p - 1) = 1$$
2. Computes
$$r \equiv \alpha^k [p]$$
3. Calculates
$$s \equiv \frac{H(m) - xr}{k} [p - 1]$$

To verify the validity of signature, Bob replaces m, r, and s in equation (1).

ElGamal Digital Signature over Elliptic Curves[14]

Alice wants to sign a document. She first must establish a public key. She chooses an elliptic curve E over a finite field F_q and a point $A \in E(F_q)$, the order of A is a large prime n. Alice also chooses a secret integer a and computes $B = a A$.

The equation of the signature is

$$f(R) B + s R = H(m) A \qquad (2)$$

where f is a function such that $f: E(F_q) \to \mathbb{Z}$. f must verify that: only a small number of inputs give the same output which is a large number.

Alice's public key is E, F_q, f, A, and B. The only private key is a.

To sign a document, Alice does the following:

1. Calculates $H(m)$
2. Chooses a random integer k co-prime with n and computes $R = k A$.
3. Determines $s \equiv k^{-1}(H(m) - a f(R)) [n]$.

The signed message is (m, R, s).

Bob verifies the signature as follows:

1. Downloads Alice's public key.
2. Computes $V_1 = f(R) B + s R$ and $V_2 = H(m) A$.
3. If $V_1 = V_2$, he declares the signature valid.

Our contribution

Amelioration of the digital ElGamal scheme in F_p

It is known that we cannot sign two messages with the same key using the ElGamal signature,[3] and it is a must to change the parameter k after each signature. Indeed: Suppose that Alice signs two messages m_1 and m_2 with the same random number. She provides (r_1, s_1) and (r_2, s_2) which verifies :

$$(S_1)\begin{cases} r_1 \equiv \alpha^k[p] \\ s_1 \equiv \dfrac{H(m_1) - xr_1}{k}[p-1] \end{cases}$$

$$(S_2)\begin{cases} r_2 \equiv \alpha^k[p]\ \equiv r_1 \\ s_2 \equiv \dfrac{H(m_2) - xr_2}{k} \equiv \dfrac{H(m_2) - xr_1}{k}[p-1] \end{cases}$$

If Eve, knows the two messages m_1 and m_2 and intercepts the two signatures (r_1, s_1) and (r_2, s_2) then she can do:

$$s_1 - s_2 \equiv \frac{H(m_1) - H(m_2)}{k}[p-1]$$

This means

$$k \equiv \frac{H(m_1) - H(m_2)}{s_1 - s_2}[p-1]$$

We suppose that $s_1 - s_2$ is invertible modulo $p - 1$.

After the calculation of k, Eve will obtain the private key of Alice x by one of the two systems (S_1) and (S_2).

Now we describe our amelioration of the signature. We do not change the step of key generation.

To sign a document m, Alice does as follows:

1. Chooses a random number k such that
$$\gcd(H(m\ +\ k), p-1) =\ 1$$
2. Computes
$$r \equiv \alpha^{H(m+k)}[p]$$
3. Computes
$$s \equiv \frac{H(m) - xr}{H(m+k)}[p-1]$$

The validation step of signature stays the same, Bob replaces m, r, and s in equation (1).

The added value in our amelioration is that we can sign as many messages as we want with the same parameter k, and we do not need to change it. Indeed:

Suppose Alice signs two messages m_1 and m_2 with the same random number using our method. She provides (r_1, s_1) and (r_2, s_2) which verfies:

$$(S_1)\begin{cases} r_1 \equiv \alpha^{H(m_1+k)}[p] \\ s_1 \equiv \dfrac{H(m_1) - xr_1}{H(m_1+k)}[p-1] \end{cases}$$

$$(S_2)\begin{cases} r_2 \equiv \alpha^{H(m_2+k)}[p] \\ s_2 \equiv \dfrac{H(m_2) - xr_2}{H(m_2+k)}[p-1] \end{cases}$$

Eve can do the following:

$$s_1 - s_2 \equiv \frac{H(m_1)-xr_1}{H(m_1+k)} - \frac{H(m_2)-xr_2}{H(m_2+k)}[p-1] \qquad (3)$$

Equation (3) contains three unknown variables x, $H(m_1+k)$, $H(m_2+k)$ and it is difficult to find the secret parameters k and x.

Remark 1. Note that Alice does not need to choose a parameter k to sign, she can simply use her secret key x.

Our method is not confined to ElGamal digital signature alone, but to all its variations.

Menezes, van Oorschot and Vanstone say in their book handbook of applied cryptography [8] that the signing equation can be written as $u \equiv xv + kw\ [p-1]$, where $u = H(m)$, $v = r$ and $w = s$ $(H(m) \equiv xr + ks\ [p-1])$.

They also say that other signing equations can be obtained by permitting u, v and w to take on the values s, r and $H(m)$ in different orders.

So, if we applied our amelioration over the six variations announced in their book, we find the following results:

Table 1. Variations of the ElGamal signing equation.

	u	v	w	**Signing equation** $[p-1]$	Verification $[p]$
1	$H(m)$	r	s	$H(m) \equiv xr + H(m+k)s$	$\alpha^{H(m)} \equiv (\alpha^x)^r r^s$
2	$H(m)$	s	r	$H(m) \equiv xs + H(m+k)r$	$\alpha^{H(m)} \equiv (\alpha^x)^s r^r$
3	s	r	$H(m)$	$s \equiv xr + H(m+k)H(m)$	$\alpha^s \equiv (\alpha^x)^r r^{H(m)}$
4	s	$H(m)$	r	$s \equiv xH(m) + H(m+k)r$	$\alpha^s \equiv (\alpha^x)^{H(m)} r^r$
5	r	s	$H(m)$	$r \equiv xs + H(m+k)H(m)$	$\alpha^r \equiv (\alpha^x)^s r^{H(m)}$
6	r	$H(m)$	s	$r \equiv xH(m) + H(m+k)s$	$\alpha^r \equiv (\alpha^x)^{H(m)} r^s$

Note that in the six cases Alice can sign several messages with the same key, or as we said in remark 1 only with the secret key x.

Remark 2. If we use the signature equation (1) or (6) in Table 1, we need to make sure that $gcd(H(m+k), p-1) = 1$. For the other singing equations finding the inverse of $H(m+k)$ modulo $p-1$ isn't required to compute s.

Hash functions

In this amelioration we propose four hash functions of the SHA family. We leave the choice open depending on the user. A hash function is defined mainly by three aspects: Message size, Message digest size and security.[13]

Table 2. Secure Hash Functions.

Hash Functions Terms (bits)	SHA-1	SHA-2 (256)	SHA-2 (384)	SHA-2 (512)
Message Size	$< 2^{64}$	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$
Message Digest	160	256	384	512
Security	80	128	192	256

In our method the key size is fixed and pseudo-random unlike ElGamal who takes randomly a number k. If we suppose that the running time of digital signature of ElGamal is T, finding an appropriate key will take $T + Time(Hash(message + k))$. Proposing this new way for generating the private key require more running time, but will enable us to keep a unique private key k for each signature.

Amelioration of the digital ElGamal scheme in elliptic curves

The problem to sign two messages with same key did accompany ElGamal digital signature even over elliptic curves. Indeed:

Suppose that the user sign two messages m_1 and m_2 with the same key k .
The two signatures are $(m_1,\ R_1,\ s_1)$ and $(m_2,\ R_2,\ s_2)$ where

$$s_1 \equiv\ k^{-1}(H(m_1) - a\, f(R))\ [n]$$

$$s_2 \equiv\ k^{-1}(H(m_2) - a\, f(R))\ [n]$$

Subtracting the two equations. Eve gets

$$s_1 - s_2 \equiv\ k^{-1}(H(m_1) - H(m_2))\ [n].$$

This means

$$k \equiv \frac{H(m_1) - H(m_2)}{s_1 - s_2}\, [n]$$

Once Eve knows k, she can find a.

Let us now apply our amelioration in this signature. We do not change the step of key generation.

To sign a document Alice does as follows:

1. Calculates $H(m)$.
2. Chooses a random integer k and computes $R = H(m + k)\, A$. We suppose that $H(m\ +\ k)\ <\ n$
3. Determines

$$s \equiv H(m+k)^{-1}(H(m) - a\, f(R))\ [n]$$

The validation step of signature stays the same, Bob replaces R and s in equation (2).

Now we suppose that Alice signs two messages m_1 and m_2 with the same random number using our method. She provides (r_1, s_1) and (r_2, s_2) which verifies

$$s_1 \equiv H(m_1+k)^{-1}[H(m_1) - a\, f(R_1)]\ [n]$$
$$s_2 \equiv H(m_2+k)^{-1}[H(m_2) - a\, f(R_1)]\ [n]$$

Subtracting the two equations. Eve gets

$$s_2 - s_1 \equiv H(m_1+k)^{-1}[H(m_1) - a\, f(R_1)] - H(m_2+k)^{-1}[H(m_2) - a\, f(R_1)]\ [n]$$

Which contains three unknown variables $H(m_1 + k)$, $H(m_2 + k)$ and a.

Hence, we have solved the problem of signing several messages with the same key. Notice that our amelioration remains valid over all variations of ElGamal digital signature in elliptic curves.

Remark 3. Note that in elliptic curves Alice can also sign only with the secret key a and she does not need to choose any other parameter.

Security analysis

In F_p :

Suppose that Alice signs several messages $m_1, m_2, \cdots, m_t$ using $k_1, k_2, \cdots, k_t$ respectively. So, the system of equations is:

$$\begin{cases} s_1 \equiv \dfrac{H(m_1) - xr}{k_1}[p-1] \\ s_2 \equiv \dfrac{H(m_2) - xr}{k_2}[p-1] \\ \vdots \\ s_t \equiv \dfrac{H(m_t) - xr}{k_t}[p-1] \end{cases}$$

Suppose now that the user selects a parameter k and signs several messages $m_1, m_2, \cdots, m_t$ using our method. He finds the following system :

$$\begin{cases} s_1 \equiv \dfrac{H(m_1) - xr}{H(m_1 + k)}[p-1] \\ s_2 \equiv \dfrac{H(m_2) - xr}{H(m_2 + k)}[p-1] \\ \vdots \\ s_t \equiv \dfrac{H(m_t) - xr}{H(m_t + k)}[p-1] \end{cases}$$

If we put

$$\begin{cases} k_1 \equiv H(m_1 + k) \\ k_2 \equiv H(m_2 + k) \\ \vdots \\ k_t \equiv H(m_t + k) \end{cases}$$

The system (S_4) becomes exactly (S_3). Hence the two signatures have the same security.

In elliptic curves:

We do the same thing as the case above.
ElGamal method:

$$(S_5)\begin{cases} s_1 \equiv \ k_1^{-1}(H(m_1) - a\,f(R_1))[n] \\ s_2 \equiv \ k_2^{-1}(H(m_2) - a\,f(R_2))[n] \\ \vdots \\ s_t \equiv \ k_t^{-1}(H(m_t) - a\,f(R_t))[n] \end{cases}$$

And if we use the same change of variables like in (*), the system (S_6) becomes exactly (S_5). Hence the two methods have the same security.

Complexity

As we can see, in the method either in the group multiplicative or elliptic curves, the complexity of ElGamal digital signature is similar to our method. The only difference is the running time to compute $H(m + k)$ is added to the signing equation. Hence, based on the hash function used the complexity might differ.

Conclusion

In this work we presented a new way to sign a message using ElGamal digital signature and its variations in group multiplicative and elliptic curves. Our approach secured ElGamal signature against Known-Messages-Attack. We analysed its security and showed that by using only one private key, we reinforced ElGamal signature and its variants.

Acknowledgements

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

References

[1] Johannes Buchmann, *Introduction to Cryptography* (Springer Science & Business Media, 2013).

[2] Whitfield Diffie and Martin E. Hellman, "New Directions in Cryptography," *IEEE transactions on Information Theory* 22, no. 6 (1976): 644-654.

[3] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE Transactions on Information Theory* 31, no. 4 (1985): 469-472.

[4] Don Johnson, Alfred Menezes, and Scott Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)," *International Journal of Information Security* 1, no. 1 (2001): 36-63.

[5] Neal Koblitz, "Elliptic Curve Cryptosystems," *Mathematics of Computation* 48, no. 177 (1987): 203-209.

[6] Neal Koblitz, "Algebraic Aspects of Cryptography," Algorithms and Computation in Mathematics, vol. 3 (Berlin: Springer-Verlag, 1998).

[7] Alfred Menezes, "Elliptic Curve Cryptosystems," *CryptoBytes* 1, no. 2 (1995).

[8] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, *Handbook of Applied Cryptography* (Boca Raton, FL: CRC press, 1996).

[9] Victor S. Miller, "Use of Elliptic Curves in Cryptography," in *Advances in Cryptology – CRYPTO'85 Proceedings*, edited by Hugh C. Williams (Berlin, Heidelberg: Springer, 1985), 417-426.

[10] P. Vasudeva Reddy and M. Padmavathamma, "An Authenticated Key Exchange Protocol in Elliptic Curve Cryptography," *Journal of Discrete Mathematical Sciences and Cryptography* 10, no. 5 (2007): 697-705.

[11] Ronald Linn Rivest, Adi Shamir, and Leonard Max Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM* 21, no. 2 (1978): 120-126.

[12] René Schoof, "Elliptic Curves Over Finite Fields and the Computation of the Square Roots mod *p*," Mathematics of Computation, 44, no. 170 (1985): 483-494, https://doi.org/10.1090/S0025-5718-1985-0777280-6.

[13] Nicolas Sklavos and Odysseas G. Koufopavlou, "On the Hardware Implementations of the SHA-2 (256, 384, 512) Hash Functions," in Proceedings of the 2003 International Symposium on Circuits and Systems, ISCAS'03, Bangkok, 2003, pp. V-V, https://doi.org/10.1109/ISCAS.2003.1206214.

[14] Lawrence C. Washington, *Elliptic Curves: Number Theory and Cryptography*, 2nd edition (Boca Raton, FL: CRC press, 2008).

About the Authors

The authors are with the Laboratory of Mathematics, Cryptography, Mechanics and Numerical Analysis, Faculty of Science and Technology, University Hassan II of Casablanca, Morocco. They can be reached respectively at marouane.ihia@gmail.com and khadir@hotmail.com.