CREAT. MATH. INFORM.
Volume **29** (2020), No. 2,
Pages 137 - 144

Online version at https://creative-mathematics.cunbm.utcluj.ro/
Print Edition: ISSN 1584 - 286X; Online Edition: ISSN 1843 - 441X
DOI: https://doi.org/10.37193/CMI.2020.02.04

A resistant digital signature based on elliptic curves

M. IHIA and O. KHADIR

ABSTRACT. The paper presents a new digital signature in elliptic curves. Its efficiency and security are established. The added value of this method is that it uses only a single variable as private key and the user can sign all his messages without changing it. We prove that there exists no risk for falsifying the signature or finding the secret key.

1. INTRODUCTION

In this new era of technology, security is the main focus of all branch of sciences. To secure documents or achieve any sort of identification, using the conventional method by signing on a formal paper is not suitable anymore. Hence the need to use an improved form of signature. A digital signature in computer science is equivalent to a manual signature done by a person. The party performing the signature either a person or a company, is engaged in the same manner as if he signed by hand. The concept was realisable due to asymmetric cryptography. This technique makes it possible to encrypt with a secret key and to decrypt with another key called public, both being generated beforehand. The principle of asymmetric cryptography is due to Diffie and Hellman [2]. In 1976, they suggested a method to construct a common key between two entities that will secure the communication over a public channel. Their idea was based on the difficulty of the famous discrete logarithm problem [18, p.234] [10, p.113]. The digital signature guarantees the integrity, authentication and non-repudiation of a document. There are several signatures as RSA [14] in 1978 and Rabin [13] in 1979 based on the problem of integer factorization. ElGamal [3] in 1985, Schnorr signature [15] in 1989 and DSA [4, p.4] in 1991 based on the difficulty of the resolution of the discrete logarithm problem. ElGamal signature is more expensive compared to other digital protocols. The method works modulo a prime p with an exponent witch has the same size as the prime number. This contrasts with DSA and Schnorr, which both work traditionally in a 160-subgroup or a 1024-bit modulus [10, p.453]. Schnorr and DSA are better if we desire a faster signature protocol but smaller compared by size. The recent requirement for industries are directed toward speed and low memory space. We can say safely that DSA and Schnorr signatures are the most commonly used. These signatures have known a new version after the work of Koblitz and Miller [9, 12] in 1986. The creators showed independently that elliptic curves over fields detain a suitable finite groups for pub-lic key cryptography. An elliptic curve cryptosystem offers the best security per bit among all current public-key schemes [11]. There is no sub exponential algorithm known to solve the elliptic curve discrete logarithm problem on a properly chosen curve. To create public key cryptosystems, elliptical curve cryptography remains a very effective technology. A cryptographical ECC scheme security is based on the difficulty of solving the discrete logarithms over finite group, such that a 192-bit key in ECDSA is similar to a 1024-bit key in RSA. Comparing the size of the

Received: 27.03.2019. In revised form: 02.03.2020. Accepted: 09.03.2020
2010 *Mathematics Subject Classification.* 94A60, 14H52.
Key words and phrases. *public key cryptography, elliptic curves, digital signature, discrete logarithm p problem.*
Corresponding author: O. Khadir; khadir@hotmail.com

cipher will go in favor of RSA cryptosysem. ECC is known of large output crypted messages compared to other cryptosystems. This can be seen as a disvantage when we opt for ECC as a suitable choice for small devices or fast data transmission. Since the introduction of ECC many cryptosystems have proven their efficiency trough the years. Both RSA and ECDSA are accepted as a strong signatures and used in practice. The advantages of ECDSA over RSA is that it is used more often over wireless systems for portability of limited resources electronic systems. One of the pointed disadvantages of using an ECC cryptosystems is that the crypted message size is much bigger than a conventional cryptosystem.

In this work, we describe an electronic signature scheme based on the discrete logarithm problem in elliptic curve groups. The efficiency of the method is discussed and its security analyzed. Our goal is to build a signature that resists against known attacks while reducing the number of variables in the signature function.

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

Throughout the sequel we use classical notations: $\mathbb{Z}$ is the set of integer. For every prime integer, we denote by $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ 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 $a - b$. H is used for the hash function.

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

2. Elliptic curves [5, 19]

An elliptic curve over the finite field $\mathbb{F}_q$, where q is a power of a prime number and the characteristic is not 2 or 3, is the set of solutions $(x, y) \in \mathbb{F}_q^2$ to the equation $y^2 = x^3 + ax + b$, where the discriminant $-(4a^3 + 27b^2) \neq 0$. We add to this curve a point at infinity denoted $\mathcal{O}$.

Schoof's algorithm [16] is the best method that counts the points of elliptic curves over a finite field. It works in a polynomial time.

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

(1) Identity: $P + \mathcal{O} = \mathcal{O} + P = P$ for all $P \in E(\mathbb{F}_q)$.

(2) Opposite: If $P = (x, y) \in E(\mathbb{F}_q)$ then the point $-P = (x, -y)$ is the opposite of P.

(3) Points addition: Let $P = (x_1, y_1)$ and $Q = (x_2, y_2) \in E(\mathbb{F}_q)$, where $P \neq \pm Q$. Then $P + Q = (x_3, y_3)$, where

$$x_3 = m^2 - x_1 - x_2 \qquad \text{and} \qquad y_3 = m(x_1 - x_3) - y_1$$

with $m = \dfrac{y_2 - y_1}{x_2 - x_1}$.

Otherwise $P + Q = \mathcal{O}$.

(4) Point doubling: Let $P = (x_1, y_1) \in E(\mathbb{F}_q)$, where $P \neq -P$. Then $2P = (x_3, y_3)$, where

$$x_3 = m^2 - 2x_1 \qquad \text{and} \qquad y_3 = m(x_1 - x_3) - y_1$$

with $m = \dfrac{3x_1^2 + a}{2y_1}$.

Otherwise $2\,P = \mathcal{O}$.

For more details, we refer the reader to references [1, 8, 19]. The addition is used for constructing an abelian group.

Theorem 2.1 ([6, p.287])**.** *The elliptic curve $E(\mathbb{F}_q)$, with the binary operation $+$ forms an abelian group whose identity element is the point at infinity $\mathcal{O}$.*

There are many efficient algorithms to calculate a multiple of points in elliptic curves. see for instance [10, p.615].

Definition 2.1 ([7, p.180])**.** Let p be a prime and let $P, Q \in E(\mathbb{F}_p)$. Suppose we know that there exists an integer x such that

$$Q = x\,P$$

The elliptic curve discrete logarithm problem is how to find x.

Definition 2.2 ([19, p.177])**.** 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 $H(m_1) = H(m_2)$.

2.1. **ElGamal digital signatures** [19, p.175]**.** Alice wants to sign a document. She first must establish a public key. She chooses an elliptic curve $E(\mathbb{F}_q)$ over a finite field $\mathbb{F}_q$ and a point $A \in E(\mathbb{F}_q)$, 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

$$f(R)\,B + s\,R = H(m)\,A \tag{2.1}$$

where f is a function such that $f : E(\mathbb{F}_q) \to \mathbb{Z}$ and its image must be large and have a few number of output. s and R are the unknown variables.
Alice's public key is E, $\mathbb{F}_q$, 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.

3. Our contribution

Let's assume that Alice wants to sign a document sent by Bob. She chooses an elliptic curve $E(\mathbb{F}_q)$ over a finite field $\mathbb{F}_q$. She needs two elements to calculate her signature: a point $A \in E(\mathbb{F}_q)$ such that its order is a prime number n and a secret integer k. Then she computes $B = k\ A$. The equation of the signature is

$$s\,A = H(m)\,f(R)\,B + R \tag{3.2}$$

where f is defined as the same way as in ElGamal signature. s and R are the unknown variables. Alice's public key is $E(\mathbb{F}_q)$, $\mathbb{F}_q$, f, A, and B. The only private key is k. To sign a document, Alice does the following:

(1) Calculates $H(m)$ and $H(m+k)$.
(2) Computes $R = H(m+k)\,B$.
(3) Determines $s \equiv k[H(m)\,f(R) + H(m+k)]\ [n]$.

The message signature is (m, R, s). Bob verifies the signature as follows:

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

We illustrate our method by giving the next example.

Example 3.1. To simplify the calculation, we take $f(R) = f(x, y) = x$ and $H(m) = m$. Let $E(\mathbb{F}_{5783})$ be given by the equation $y^2 \equiv x^3 + 2x + 7\ [5783]$.
$(\Delta \equiv -(4a^3 + 27b^2) \equiv 4428\ [5783])$.
The cardinality of the elliptic curve is

$$\#E(\mathbb{F}_{5783}) = 5815 = 5 \times 1163.$$

Alice chooses randomly the point $A = (3576, 1242)$ whose order is $n = 1163$ and the integer $k = 911$. She computes $B = k\,A = (1683, 4630)$.
The public and private key of Alice are $(A, B, E(\mathbb{F}_{5783}))$ and k respectively.
To sign the message $m = 725$, Alice calculates:

(1) $R = (m+k)\,B = (1437, 4977)$.
(2) $s \equiv k\,(m\,x + m + k) \equiv 965\ [1163]$.

The message signature is $(m, R, s) = (725, (1437, 4977), 965)$.
Bob verifies the signature as follows:
He computes $V = s\,A - m\,x\,B = (1437, 4977)$. He finds $V = R$. Hence the signature is valid.

3.1. **Security Analysis.** To falsify the signature of Alice, Eve has to find R and s such that the equation $V = R$ is satisfied. Suppose that Eve tries to forge Alice signature by fixing arbitrary one parameter and looking for the second:

(1) If she fixes the point R and aims to compute s, she will be faced by the discrete logarithm problem $H(m)\,f(R)\,B + R = s\,A$ for the integer s.
(2) If she chooses s, then the problem becomes an equation for R. The equation appears to be at least as complex as a discrete logarithm problem.

Moreover, we don't know any algorithm or procedure that can be used for finding R and s simultaneously.

3.2. **Attacks.**

(1) If Alice wants to sign a message m, she must use a hash function.
Suppose that Alice signs two messages m_1 and m_2 without using the hash function for calculating the point R.
The signed messages are (m_1, R_1, s_1) and (m_2, R_2, s_2), where

$$\begin{aligned} R_1 &= (m_1 + k)\,B \\ R_2 &= (m_2 + k)\,B \\ s_1 &\equiv k\,[H(m_1)\,f(R_1) + m_1 + k]\ [n] \\ s_2 &\equiv k\,[H(m_2)\,f(R_2) + m_2 + k]\ [n] \end{aligned}$$

$$s_1 - s_2 \equiv k\,[H(m_1)\,f(R_1) - H(m_2)\,f(R_2) + m_1 - m_2]\ [n] \tag{3.3}$$

The equation (3.3) becomes

$$k \equiv \frac{s_1 - s_2}{H(m_1)\, f(R_1) - H(m_2)\, f(R_2) + m_1 - m_2}\ [n]$$

Hence the importance of using a hash function to calculate the point R.

(2) If Alice Signs several messages with the same k, then

$$(S)\begin{cases} s_1 \equiv k\,[H(m_1)\, f(R_1) + H(m_1 + k)]\ [n] \\ s_2 \equiv k\,[H(m_2)\, f(R_2) + H(m_2 + k)]\ [n] \\ \vdots \\ s_r \equiv k\,[H(m_r)\, f(R_r) + H(m_r + k)]\ [n] \end{cases}$$

Since the system (S) contains $r+1$ unknown variables k and $H(m_i, k)$ for $i \in \{1,2, ..., r\}$, Eve can find many valid solutions. It is difficult to know the real solution of (S).

3.3. **Advantage.** Suppose that Alice sings two messages m_1 and m_2 with the same integer k. The signed messages are (m_1, R_1, s_1) and (m_2, R_2, s_2).
The equations for s_1 and s_2 give the following:

$$s_1 = k\,[H(m_1)\, f(R_1) + H(m_1 + k)]\ [n] \tag{3.4}$$

$$s_2 = k\,[H(m_2)\, f(R_2) + H(m_2 + k)]\ [n] \tag{3.5}$$

(1) (3.4) $-$ (3.5) implies that:

$$s_1 - s_2 = k\,[H(m_1)\, f(R_1) - H(m_2)\, f(R_2) + H(m_1 + k) - H(m_2 + k)]\ [n] \tag{3.6}$$

Put

$$\begin{aligned} s &= s_1 - s_2 \\ \alpha &= H(m_1)\, f(R_1) - H(m_2)\, f(R_2) \\ x &= H(m_1 + k) \\ y &= H(m_2 + k) \end{aligned}$$

So, the equation (3.6) becomes $s \equiv k\,(\alpha + x - y)\ [n]$, which contains three unknown variables k, x and y.

(2) $\dfrac{(3.4)}{(3.5)}$ implies that:

$$\frac{s_1}{s_2} \equiv \frac{H(m_1)\, f(R_1) + H(m_1 + k)}{H(m_2)\, f(R_2) + H(m_2 + k)}\ [n] \tag{3.7}$$

Put

$$\begin{aligned} s &= s_1\, H(m_2)\, f(R_2) - s_2\, H(m_1)\, f(R_1) \\ x &= H(m_1 + k) \\ y &= H(m_2 + k) \end{aligned}$$

So, the equation (3.7) becomes $s \equiv -s_1\, y + s_2\, x\ [n]$, which contains two unknown variables x and y.

Hence, Alice can sign two messages with the same parameter k without needing to change it.

3.4. **Particular case.**

(1) Suppose that a user signs a message m_1 with the parameter k_1 and wants to choose an another private key k_2 to sign a message m_2.
Even he takes $k_2 = -k_1$, the attacker cannot extract any information from the two messages m_1 and m_2. Indeed:
The two signatures are (m_1, R_1, s_1) and (m_2, R_2, s_2), where

$$s_1 \equiv k_1\,[H(m_1)\,f(R_1) + H(m_1 + k_1)]\ [n]$$
$$s_2 \equiv -k_1\,[H(m_2)\,f(R_2) + H(m_2 - k_1)]\ [n]$$

Eve divides the two equations, and she gets

$$\frac{s_1}{s_2} \equiv -\frac{H(m_1)\,f(R_1) + H(m_1 + k_1)}{H(m_2)\,f(R_2) + H(m_2 - k_1)} \tag{3.8}$$

Put

$$s = -s_2\,H(m_1)\,f(R_1) - s_1\,H(m_2)\,f(R_2)$$
$$x = H(m_1 + k_1)$$
$$y = H(m_2 - k_1)$$

The equation (3.8) becomes $s_2\,x + s_1\,y \equiv s\ [n]$, which contains two unknown variables x and y.

(2) Suppose now that the user does the same thing with the ElGamal signature.
The two signatures are (m_1, R_1, s_1) and (m_2, R_2, s_2), where

$$s_1 \equiv k_1^{-1}\,[H(m_1) - a\,f(R_1)\ [n]$$
$$s_2 \equiv -k_1^{-1}\,[H(m_2) - a\,f(R_2)\ [n]$$

Dividing the two equations. Eve gets

$$\frac{s_1}{s_2} \equiv -\frac{H(m_1) - a\,f(R_1)}{H(m_2) - a\,f(R_2)} \tag{3.9}$$

After developing, the equation (3.9) becomes $a \equiv \dfrac{s_1\,H(m_2) + s_2\,H(m_1)}{s_1\,f(R_2) + s_2\,f(R_1)}\ [n]$.
Hence Eve can find the private key a of the user.

3.5. **Hash functions.** In this contribution we review 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 [17].

TABLE 1. 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 requires more running time, but will enable us to keep a unique private key k for each signature.

TABLE 2. The comparison of the complexities

Complexity	ElGamal signature	Our method
Generation of keys	Both methods uses the same key generation techniques	
To sign	One multiple of point in EC One calculation of HF Two multiplications One addition One inverse modular	One multiple of point in EC Two calculations of HF Two multiplications Two additions
To verify	Three multiples of points in EC Addition of two points in EC	Two multiples of points in EC Addition of two points in EC One multiplication

3.6. **Complexity.** The two signatures above have the same complexity. Our method offers more advantages.

4. CONCLUSION

In this paper we proposed a new digital signature which is based on the discrete logarithm problem in elliptic curves. Our signature is more resistant and secure against known attacks. The only unknown variable needed to perform the signature is the message itself. We believe that our method can be used as an alternative if any previous signature is compromised. Finally, we compare our proposed solution with two of the most popular signatures, RSA and ECDSA:

- With key sizes smaller, our method provides the same level of security.
- ECDSA and our method provide faster computations and less storage space.
- In our method we can sign several messages with the same key without needing to change it, while ECDSA it must change every time. RSA is a algorithm based on factorization, so that every time RSA initialization takes two large prime number p and q.
- The length of the private and public keys is shorter in our method and ECDSA. This is explained by faster processing times, and lower demands on bandwidth and memory.
- For the three signatures, the message length must be less than the bit length otherwise algorithm will fail.
- ECDSA provides effective and compact implementations for cryptographic operations requiring smaller chips.
- ECDSA and our method are mostly suitable for machines having less memory, low bandwidth and low computing power..
- The execution time between RSA and our method was significant. This result is explained by the RSA keys which are larger than our signature keys.

REFERENCES

[1] Buchmann, J., *Introduction to cryptography*, Springer Science & Business Media, 2013

[2] Diffie, W. and Hellman, M., *New directions in cryptography*, IEEE Trans. Inform. Theory, **IT-22** (1976), No. 6, 644–654

[3] ElGamal, T., *A public key cryptosystem and a signature scheme based on discrete logarithms*, IEEE Trans. Inform. Theory, **31** (1985) , No. 4, 469–472

[4] Johnson, D., Menezes, A. and Vanstone, S., *The elliptic curve digital signature algorithm (ECDSA)*, International Journal of Information Security, **1** (2001), No. 1, 36–63

[5] Hankerson, D., Menezes, A. J. and Vanstone, S., *Guide to elliptic curve cryptography*, Computing Reviews, **46** (2005) , No. 1, 13pp.

[6] Hoffstein, J., Pipher, J. and Silverman, J. H., *An introduction to mathematical cryptography*, Springer, 2008

[7] Koblitz, N., *A course in number theory and cryptography*, Second edition, Springer-Verlag, 1994

[8] Koblitz, N., *Algebraic aspects of cryptography*, Volume 3 of Algorithms and Computation in Mathematics, Springer-Verlag 1998

[9] Koblitz, N., *Elliptic curve cryptosystems*, Math. Comp., **48** (1987), No. 177, 203–209

[10] Menezes, A. J., Van Oorschot, P. C. and Vanstone, S. A., *Handbook of applied cryptography*, CRC press, 1996

[11] Menezes, A., *Elliptic Curve Cryptosystems*, CryptoBytes, 1995, 1(2)

[12] Miller, V. S., *Use of elliptic curves in cryptography*, Conference on the Theory and Application of Cryptographic Techniques, 1985, 417–426

[13] Rabin, O. M., *Digitalized signatures and public-key functions as intractable as factorization*, Massachusetts Inst of Tech Cambridge Lab for Computer Science, 1979

[14] Rivest, R. L., Shamir, A. and Adleman, L., *A method for obtaining digital signatures and public-key cryptosystems*, Communications of the ACM, **21** (1978), No. 2, 120–126

[15] Schnorr, C.-P., *Efficient identification and signatures for smart cards*, Conference on the Theory and Application of Cryptology, Springer, 1989, 239–252

[16] Schoof, R., *Elliptic curves over finite fields and the computation of the square roots modulo p*, Math. Comp., **44** (1985), No. 170, 483-–494

[17] Sklavos, N. and Koufopavlou, O., *On the hardware implementations of the SHA-2 (256, 384, 512) hash functions*, Proceedings of the 2003, International Symposium on Circuits and Systems, IEEE, 2003, ISCAS'03., (5), V–V

[18] Stinson, D. R., *Cryptography: theory and practice*, Chapman and Hall/CRC, 2005

[19] Washington, L. C., *Elliptic curves: number theory and cryptography*, CRC press, 2008

FSTM, UNIVERSITY HASSAN II
LABORATORY OF MATHEMATICS, CRYPTOGRAPHY
MECHANICS AND NUMERICAL ANALYSIS
STR 146, 28806, MOHAMMEDIA, MOROCCO
Email address: marouane.ihia@gmail.com