A Modification of ECDSA to Avoid the Rho Method Attack

Amira Zahra, Kiki Ariyanti Sugeng*

Department of Mathematics, Faculty of Mathematics and Natural Sciences
Universitas Indonesia, Depok 16424, Indonesia
**Corresponding author. Email: kiki@sci.ui.ac.id*

ABSTRACT

Elliptic Curve Digital Signature Algorithm (ECDSA) is a digital signature algorithm that utilizes an elliptic curve. ECDSA consists of three steps, which are key generation, signature generation, and verification algorithm. ECDSA is used on Bitcoin transactions to generate the user's public key, private key, and signature, and also to verify a Bitcoin user's signature. There are some researches on ECDSA weak randomness which can be exploited by attackers to reveal the user's private key, and causes thefts of the user's money. ECDSA weak randomness is generating a random number that is not cryptographically secure. Some modifications of ECDSA to overcome this problem have been done, such as generating the digital signature by using two private keys. Although those modified algorithms overcome ECDSA weak randomness exploitations, it is not resistant to the Rho method attack which can solve elliptic curve discrete logarithm problem (ECDLP). In case ECDLP can be solved, the user's private key can be revealed. Therefore, in this paper, we modify an ECDSA algorithm that overcomes the exploitation of ECDSA weak randomness and is also resistant to the Rho method attack by using three private keys.

Keywords: *ECDLP, ECDSA, ECDSA weak randomness, Rho method attack.*

1. INTRODUCTION

Elliptic Curve Digital Signature Algorithm (ECDSA) is a digital signature algorithm which utilizes an elliptic curve. ECDSA consists of three steps, which are key generation, signature generation, and verification algorithm [1]. ECDSA is utilized in the digital signature scheme in Bitcoin transactions [2]. In 2013, Schneider presented a complete computational process to show there is a potential private key leakage on public blockchain [3]. Giechaskiel, Cremers, and Rasmussen analyze ECDSA weak randomness. ECDSA weak randomness is generating a random number that is not cryptographically secure. Since reusing random numbers is not cryptographically secure, it belongs to ECDSA weak randomness. ECDSA weak randomness exploitations can reveal a user's private key [4]. The leakage of user's private key leads to money theft on Bitcoin wallet [5].

To avoid the exploitation of ECDSA weak randomness, a deterministic algorithm called RFC 6979 is used to determine a random number which depends on the transaction document and the user's private key. However, according to Wang, et.al. (2020), by using RFC 6979, the possibility of reusing random numbers still exists. It implies that ECDSA weak randomness exploitations cannot be avoided completely [4]. Therefore, it is needed to modify the ECDSA algorithm. In 2020, Hussein and Kashmar modified ECDSA by using two private keys d_1 and d_2 [6]. Later on, in 2021, Liu, Chen, and Liu modified ECDSA by using two random number k and k_1 [7].

Those modifications are constructed by assuming Elliptic Curve Discrete Logarithm Problem (ECDLP) is difficult to be solved. ECDLP is the problem of finding the integer d which satisfies $Q = dP$ [8]. In case ECDLP is solved, the users' private keys will be revealed. There are some methods to solve ECDLP, one of which is the Rho method attack [9]. The Rho method attack can solve ECDLP when the used elliptic curve has the base point P with order n, where $0.886\sqrt{n} < 2^{100}$ [10].

To make the digital signature scheme more secure, it is needed to create a modification of ECDSA which can avoid both ECDSA weak randomness exploitations and also the Rho method attack. In order to construct a modified ECDSA which is resistant to the Rho method attack, it is important to construct an ECDSA which cannot be attacked by solving the ECDLP. In this paper, we modify ECDSA by using three private keys to avoid ECDSA weak randomness exploitations and also the Rho method Attack.

Elliptic curve is a curve which is commonly used in practical cryptography. Elliptic curve E is a set of points which satisfy $y^2 = x^3 + ax + b$ where $4a^3 + 27b^2 \neq 0$ together with O, the point which is not in XY plane but assumed exists in every vertical line in XY plane [11]. Elliptic curve has its own addition rule which can be described as follows.

Let $P, Q \in E$ where E is an elliptic curve. To obtain $P + Q$, we create a linear line through P and Q. Let the intersection point between PQ and E is R. The reflection of R with respect to x axis R' is the result of $P + Q$. By applying the addition rule, an elliptic curve E is an addition abelian group with O as the element of identity and every point $P \in E$ has its invers $-P$ which is the reflection of P with respect to x axis [8]. The addition rule is illustrated in Figure 1.

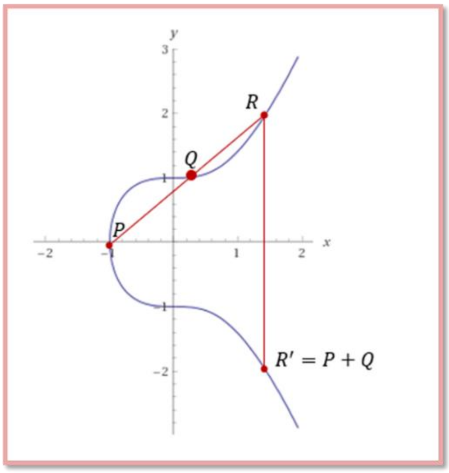

Figure 1 Addition rule on elliptic curve.

Let $n \in \mathbb{Z}$ and $P \in E$ where E is an elliptic curve. The scalar multiplication nP is defined as these following equations.

$$nP = \underbrace{P + P + \cdots + P}_{n \text{ times}}, \text{ for } n \geq 0 \qquad (1)$$

$$nP = -\underbrace{P + (-P) + \cdots + (-P)}_{-n \text{ times}}, \text{ for } n < 0 \qquad (2)$$

Elliptic curve is defined over finite field $\mathbb{F}_{2^m}$ as follow.

Elliptic curve over $\mathbb{F}_{2^m}$ (denoted by $E(\mathbb{F}_{2^m})$) is a set of points which satisfy $y^2 + xy = x^3 + ax^2 + b$ with $a, b \in \mathbb{F}_{2^m}$ [12]. Let $P, Q \in E(\mathbb{F}_{2^m})$, $P = (x_1, y_1)$ and $Q = (x_2, y_2)$. The addition law of elliptic curve $E(\mathbb{F}_{2^m})$ has the following properties [13].

1. $P + O = O + P = P$.
2. $(x_1, y_1) + (x_1, x_1 + y_1) = O$. In other word, if $P = (x_1, y_1)$, $-P = (x_1, x_1 + y_1)$.
3. If $P \neq \pm Q$, $P + Q = (x_3, y_3)$, where $x_3 = \left(\frac{y_1+y_2}{x_1+x_2}\right)^2 + \frac{y_1+y_2}{x_1+x_2} + x_1 + x_2 + a$ and $y_3 = \left(\frac{y_1+y_2}{x_1+x_2}\right)(x_1 + x_3) + x_3 + y_1$.
4. If $P = Q$, $P + Q = (x_3, y_3)$ where $x_3 = x_1^2 + \frac{b}{x_1^2}$ and $y_3 = x_1^2 + \left(x_1 + \frac{y_1}{x_1}\right)x_3 + x_3$.

Elliptic curve over finite field $\mathbb{F}_{2^m}$ is used in practical cryptography, one of which is elliptic curve digital signature algorithm (ECDSA). ECDSA consists three steps: key generation, signature generation, and verification algorithm [14].

Key Generation

1. Select an elliptic curve $E(\mathbb{F}_{2^m})$ which contains a point with order n where n is a large prime.
2. Select a point $P \in E(\mathbb{F}_{2^m})$ of order n.
3. Select an integer $d \in [1, n - 1]$.
4. Calculate $Q = dP$.
5. Output: A public key (E, P, n, Q) and a private key d.

Signature Generation

1. Input: A private key d, a message m, and a hash function h where h is SHA-1
2. Select an integer $k \in [1, n - 1]$.
3. Calculate $kP = (x, y)$ and $r = x \bmod n$.
4. Calculate $s = k^{-1}(h(m) + dr) \bmod n$.
5. Output: A signature (r, s).

The signature generation can be illustrated as the scheme in Figure 2.

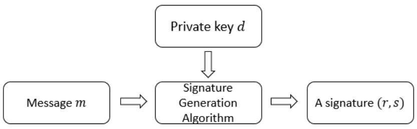

Figure 2 Illustration of signature algorithm.

Verification Algorithm

1. Input: A public key (E, P, n, Q), a message m, and a signature (r, s).
2. Compute $h(m)$ where h is SHA-1.
3. Verify $r, s \in [1, n-1]$. If $r \notin [1, n-1]$, the signature is not accepted.
4. Compute $u_1 = s^{-1}h(m) \bmod n$ and $u_2 = s^{-1}r \bmod n$.
5. Compute $u_1P + u_2Q = (x_0, y_0)$ and $v = x_0 \bmod n$.
6. The signature is accepted if and only if $v = r$.

The verification algorithm can be illustrated as the scheme in Figure 3.

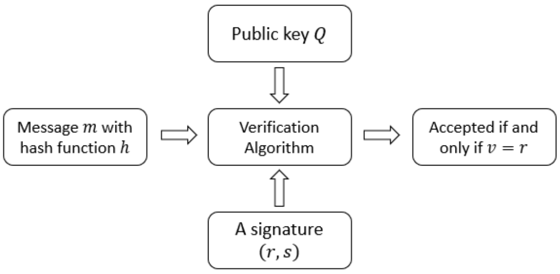
Public key Q
Message m with hash function h
Verification Algorithm
Accepted if and only if $v = r$
A signature (r, s)

Figure 3 Illustration of verification algorithm.

The paper is structured as follow: first, we present some modified ECDSAs which have been constructed to avoid ECDSA weak randomness exploitations. Then, we construct a modification of ECDSA to avoid ECDSA weak randomness exploitations and the Rho method attack.

2. RESEARCH METHODOLOGY

We use literature study in order to understand ECDSA weak randomness and the Rho method attack. Then, we analyse some modifications of ECDSA from previous research and find that the modified ECDSAs cannot prevent the Rho method attack. Finally, we propose a modification of ECDSA which is resistant to ECDSA weak randomness exploitations and also the Rho method attack.

3. THE MODIFICATION OF ECDSA

ECDSA as described on the previous section is not resistant to ECDSA weak randomness exploitations. ECDSA weak randomness is generating a random number which is not cryptographically secure. Reusing a random number is one of the cases of ECDSA weak randomness. There are two cases of ECDSA weak randomness which reveal the user's private keys. The first case is reusing the same random number k for different public keys. The second case is reusing the same random number k for the same public key.

There are some modifications of ECDSA which prevent ECDSA weak randomness exploitation. Hussein and Kashmar (2020) has modified ECDSA with two private keys d_1 and d_2. On 2021, Liu, Chen, Liu modified ECDSA by using two random number k and k_1. The differences between the original ECDSA, Hussein and Kashmar Modified ECDSA, and Liu Chen Liu Modified ECDSA is described in Table 1.

Table 1. Modifications of ECDSA

	Original ECDSA	**Hussein and Kashmar (2020)**	**Liu, Chen, Liu (2021)**
Key Generation	Private key: d Public key: (E, P, n, Q)	Private key: d_1, d_2 Public key: (E, P, n, Q_1, Q_2)	Private key: d Public key: (E, P, n, Q)
Signature Generation	Random number: k Signature: (r, s)	Random number: k Signature: (r, s)	Random number: k, k_1 Signature: (r, s, k_2)
Verification Algorithm	Signature is accepted iff $v = r$	Signature is accepted iff $v = r$	Signature is accepted iff $v = r$

On the original ECDSA and the modified ECDSA by Liu, Chen, Liu (2021), the point Q is obtained by multiplying the integer d and the point of elliptic curve P (in other word, $Q = dP$) [7]. On the modified ECDSA by Hussein and Kashmar (2020), the points Q_1 and Q_2 are obtained by computing $Q_1 = d_1P$ and $Q_2 = d_2P$ [6]. The private keys d, d_1, d_2 can be revealed by solving the ECDLP problem. There are some algorithms which are used to solve ECDLP, one of which is the Rho method attack. The Rho method attack can reveal the private key when the order of P which is denoted by n satisfies $0.886\sqrt{n} < 2^{100}$ [15]. It can be concluded that by using the original ECDSA, the modified algorithm by Hussein and Kashmar and the modified algorithm by Liu, Chen, Liu, the private key can be revealed whenever P with order n where $0.886\sqrt{n} < 2^{100}$ is chosen.

As a result, it is needed to construct a modified ECDSA which is resistant not only to ECDSA weak randomness exploitations but also to the Rho method attack. To avoid the Rho method attack, we need to construct a key generation algorithm such that the user's private key cannot be revealed by solving the ECDLP. To prevent ECDSA weak randomness exploitations, we use two points of elliptic curve. Finally, we present a modification of ECDSA.

Key Generation

1. Select an elliptic curve $E(\mathbb{F}_{2^m})$ which contain a point with order n where n is a large prime.
2. Select an integer $d_1, d_2, d_3 \in [1, n-1]$.
3. Compute $Q_1 = d_1 d_3 P$ and $Q_2 = (d_1 + d_2)P$
4. Output: private keys d_1, d_2, and d_3 and a public key (E, P, n, Q_1, Q_2).

Signature Generation

1. Select an integer $k \in [1, n-1]$
2. Compute $kP = (x, y)$ and $r = x \bmod n$.
3. Compute $s = k^{-1}(d_1 d_3 + (d_1 + d_2)r)h(m)modn$.
4. If $s = 0$, go to step 1.
5. Output: Signature (r, s)

Verification Algorithm

1. Input: A public key (E, P, n, Q_1, Q_2), a message m, and a signature (r, s).
2. Verify $r, s \in [1, n-1]$.
3. Compute $h(m)$ where h is SHA-1
4. Compute $u_1 = s^{-1}h(m) \bmod n$ and $u_2 = s^{-1}rh(m) \bmod n$.
5. Compute $u_1 Q_1 + u_2 Q_2 = (x_0, y_0)$ and $v = x_0 \bmod n$.
6. A signature is accepted if and only if $v = r$.

3.1. Validation Analysis on The Modified ECDSA

If a signature (r, s) is generated by a user's with public key Q, the following equation is obtained.

$$s = k^{-1}(d_1 d_3 + (d_1 + d_2)r)h(m) \bmod n \tag{3}$$

$$k = s^{-1}(d_1 d_3 + (d_1 + d_2)r)h(m) \bmod n \tag{4}$$

Since Equation 2 holds, $u_1 = s^{-1}h(m) \bmod n$, and $u_2 = s^{-1}rh(m) \bmod n$, we obtain

$$kP = u_1 Q_1 + u_2 Q_2 = (x_0, y_0) \tag{5}$$

Since $kP = (x, y)$ and $r = x \bmod n$, $v = r$ is required.

3.2. ECDSA Weak Randomness Analysis on The Modified ECDSA

On this section, we will show that the modified ECDSA is resistant to ECDSA weak randomness exploitations. Using the modified algorithm, if the same random number k, private keys d_1, d_2, d_3, and public key (E, P, n, Q_1, Q_2) on two different messages m_1 and m_2, the signatures (r, s_1) and (r, s_2) are obtained. Therefore, the following equations are hold.

$$s_1 = k^{-1}(d_1 d_3 + (d_1 + d_2)r)h(m_1) \bmod n \tag{6}$$

$$s_2 = k^{-1}(d_1 d_3 + (d_1 + d_2)r)h(m_2) \bmod n \tag{7}$$

By using Equation 6 and Equation 7,

$$k = (s_1 - s_2)^{-1}(d_1 d_3 + (d_1 + d_2)r)\big(h(m_1) - h(m_2)\big) \bmod n \tag{8}$$

$$d_1 = (d_3 + r)^{-1}\Big(k(s_1 - s_2) - d_2 r\big(h(m_1) - h(m_2)\big)\Big) \bmod n \tag{9}$$

$$d_2 = r^{-1}\Big((s_1 - s_2)k\big(h(m_1) - h(m_2)\big)^{-1} - d_1 d_3\Big) - d_1 \bmod n \tag{10}$$

$$d_3 = d_1^{-1}\Big(k(s_1 - s_2) - d_2 r\big(h(m_1) - h(m_2)\big)\Big) - r \bmod n \tag{11}$$

We can conclude that in order to reveal d_1, it requires to have an information about the value of k, d_2, d_3. To obtain k, d_2, d_3, it requires the value of d_1. Therefore, by using this algorithm, ECDSA weak randomness cannot be manipulated in order to reveal the user's private keys.

3.3. The Resistance of The Modified ECDSA Against The Rho Method Attack

If the Rho method attack is implemented on the modified ECDSA, the values $d_1 d_3 \equiv a \bmod n$ and $d_1 + d_2 \equiv b \bmod n$ are revealed. However, an integer which is equivalent to $a \bmod n$ or $b \bmod n$ is not unique. As a result, even though the values of a and b are known and Equation 9 is obtained, the private keys d_1, d_2, d_3 cannot be found.

4. CONCLUSION

The modification of ECDSA as presented on the previous section is resistant to ECDSA weak randomness exploitations and also the Rho method attack.

AUTHORS' CONTRIBUTIONS

Conceptualization, A.Z., K.A.S.; methodology, A.Z.; writing—original draft preparation, A.Z.; writing—review and editing, A.Z., K.A.S.; funding acquisition, K.A.S. All authors have read and agreed to the published version of the manuscript.

ACKNOWLEDGMENTS

This research has supported by PUTI-Universitas Indonesia 2020 Research Grant No. NKB-779/UN2.RST/HKP.05.00/2020.

REFERENCES

[1] L. Yehuda, Fast Secure Two-Party ECDSA Signing, in: Katz J., Shacham H. (Eds.) Advances in Cryptology – CRYPTO 2017. CRYPTO 2017. Lecture Notes in Computer Science, 2017, vol. 10402 Springer, Cham. DOI: https://doi.org/10.1007/978-3-319-63715-0_21

[2] X. Yi and K.-Y. Lam, New Blind ECDSA Scheme for Bitcoin Transaction Anonymity. In Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security (Asia CCS '19). Association for Computing Machinery, New York, NY, USA, 2019, pp. 613–620. DOI: https://doi.org/10.1145/3321705.3329816

[3] N. Schneider, Recovering Bitcoin private keys using weak signatures from the blockchain, 2013, http://www.nilsschneider.net/2013/01/28/ recovering-bitcoin-private-keys.html

[4] Z. Wang, H. Yu, Z. Zhang, J. Piao, J. Liu, ECDSA weak randomness in Bitcoin, Future Generation Computer Systems, vol. 102, 2020, pp. 507-513. DOI: https://doi.org/10.1016/j.future.2019.08.034

[5] S. Bistarelli, I. Mercanti, F. Faloci, and F. Santini, Highlighting poor anonymity and security practice in the blockchain of Bitcoin, in: Proceedings of the 36th Annual ACM Symposium on Applied Computing, 2021, pp. 265-272. DOI: https://doi.org/10.1145/3412841.3441909

[6] N.T. Hussein and A.H. Kashmar, An Improvement of ECDSA Weak Randomness in Blockchain, in: IOP Conference Series: Materials Science and Engineering, IOP Publishing, 2020, vol. 928, no. 3. DOI: https://doi.org/10.1088/1757-899X/928/3/032022

[7] S.G. Liu, W.Q. Chen and J.L. Liu, An Efficient Double Parameter Elliptic Curve Digital Signature Algorithm for Blockchain, in: IEEE Access, vol. 9, pp. 77058-77066, 2021, DOI: 10.1109/ACCESS.2021.3082704.

[8] Y. Luo, X. Ouyang, J. Liu and L. Cao, An Image Encryption Method Based on Elliptic Curve Elgamal Encryption and Chaotic Systems, in IEEE Access, vol. 7, 2019, pp. 38507-38522. DOI: 10.1109/ACCESS.2019.2906052.

[9] S.B. Sadkhan, A Proposed Developments of Pollards Rho Method for Attacking the ECDLP, 2021 7th International Engineering Conference “Research & Innovation amid Global Pandemic" (IEC), 2021, pp. 151-155, DOI: 10.1109/IEC52205.2021.9476119.

[10] D.J. Bernstein and T. Lange, SafeCurves: choosing safe curves for elliptic-curve cryptography, 2013, https://safecurves.cr.yp.to

[11] J. Wu, X. Liao, B. Yang, Color image encryption based on chaotic systems and elliptic curve ElGamal scheme, Signal Processing, Volume 141, 2017, Pages 109-124.

[12] C.A. Lara-Nino, A. Diaz-Perez and M. Morales-Sandoval, Elliptic Curve Lightweight Cryptography: A Survey, in IEEE Access, vol. 6, pp. 72514-72550, 2018, DOI: 10.1109/ACCESS.2018.2881444.

[13] D. Johnson, A. Menezes, and S. Vanstone, The Elliptic Curve Digital Signature Algorithm (ECDSA), in: IJIS 1, 2001, pp. 36–63. DOI: https://doi.org/10.1007/s102070100002

[14] J. Breitner and N. Heninger, Biased Nonce Sense: Lattice Attacks Against Weak ECDSA Signatures in Cryptocurrencies. in: Goldberg I., Moore T. (Eds) Financial Cryptography and Data Security, FC 2019, Lecture Notes in Computer Science, vol 11598. Springer, Cham, 2019. DOI: https://doi.org/10.1007/978-3-030-32101-7_1

[15] V.G. Martinez, L. González-Manzano, A.M. Muñoz, Secure Elliptic Curves in Cryptography, in: Daimi K. (eds) Computer and Network Security Essentials. Springer, Cham, 2018. DOI: https://doi.org/10.1007/978-3-319-58424-9_16