The Advance of Ethereum Digital Signature

Mengyao Li *

Department of Mathematics, Muhlenberg College, USA

* Corresponding author email: eli@muhlenberg.edu

Abstract. Ethereum digital signatures have been found to make progress, but there has yet to be a consensus regarding their significance and process. Using digital signatures in Ethereum ensures the security of users' money and the Safety of their transportation. Elliptic Curve Digital Signature Algorithm (ECDSA), private key, and public key play an important role here. This paper analyzes the details of Ethereum's Digital Signature based on existing information. First, it explained the general background information about Ethereum. Send explain the ERC-20. ERC-20 are digital assets that represent value. They could become more compatible with the new standard and allow other features. Then digital signatures are explained in detail. This part included ECDSA, private key, and public key. ECDSA is the most critical algorithm during the production process. Private and public keys are the key features to ensure Safety. The last part is the Keystore file system. Using the Keystore file system, users can secure their private keys for their cryptocurrency wallets. To access their account, the user must have a password generated by them. The unencrypted files, on the other hand, create a vulnerability in the key store, which is exposed by these unencrypted files. Once an attacker obtains the user's unencrypted key, he cannot only take control of the account but also sign transactions on behalf of the user.

Keywords: ECDSA; Private Key; Public Key; ECDSA; RSA; Keystore File System.

1. Introduction

Ethereum is a technology that allows the creation of apps and organizations, the hold of assets, the exchange of information, and the communication of assets without being controlled by any central authority or ruled by any central authority. Using Ethereum does not require the user to reveal any personal information; the user remains in control of their own data and how it is shared and used [1]. Certain activities on the Ethereum network require payment in the form of Ether, the Ethereum network's cryptocurrency [2,3]. Miners create Ethereum by building blocks and solving puzzles. To issue and implement tokens, the ERC-20 standard is used. A token represents a digital asset and has value, but not all tokens are standardized. As a result of standardization, these assets can be used on various platforms and projects. It is possible to validate the authenticity of data (such as a file) using digital signature algorithms (such as ECDSA) without compromising the security of the data in question. Signatures generated by ECDSA cannot be forged in any way. The process of ECDSA is to select an elliptic curve and base point, generate a random integer, calculate a point, and calculate a hash. Ethereum private keys are stored in the Keystore file, a JSON file containing various data [4]. The blockchain for Ethereum works in a similar manner to the blockchain for Bitcoin. Ethereum blocks differ from other cryptocurrency blocks in several key ways, including the fact that, in addition to including the number of blocks, the difficulty level, and a list of recent transactions as of the time the block was published, they also include information on the state of the transactions at the time thek was published. Among the ways in which Ethereum blocks differ from other cryptocurrencies is in this regard. Each time a brand-new state is created in a transaction list, it is reverted automatically to the previous state whenever the transaction list is reverted to the previous state [5].

The second section of the paper explained the general information and background of the Ethereum and ERC-20. They both important basics of the digital signature. The third section explains how ECDSA produces a signature, and compares the differences and similarities between ECDSA and RSA. The fourth section explains that private key and public associate with the ECDSA. And how they were used in the digital signature production. Lastly, the keystone file system was explained. The keystone file system is actually where private key and public key run and generate the results.

2. Preliminaries

2.1 Ethereum

A decentralized cryptocurrency based on open-source blockchain technology, Ethereum is a decentralized cryptocurrency built using open-source blockchain technology. To facilitate the security of a user's wallet by identifying the Ethereum market price when the user is trying to conduct a business transaction or purchase a product, Currency Security Standard (CCSS) must be utilized [6]. For the money to be proven to be the rightful owner and for the process to be secure, it is crucial to use digital signatures at the time of the transition to prove that the money is theirs. One of the primary contributions of Ethereum is its Turing completeness, which is a property that implies it is capable of performing loops as well as other types of computations. Additionally, Ethereum offers many improvements over traditional blockchain technology, such as the ability to support the state of the transaction. Blockchains that Ethereum powers are engineered using Turing-complete programming languages. Anyone can define a state transition function and ownership rules through an abstraction layer that allows them to create their own rules through the abstraction layer. By utilizing cryptographic smart contracts, this can be accomplished securely and efficiently.

2.2 ERC-20

The Ethereum smart contract will enable users to run smart contracts on a decentralized platform through the use of a new generation of Web apps, which will be made possible by the introduction of Ethereum smart contracts on a decentralized platform. Consequently, it was the first time that users were able to deploy machine-readable code (i.e., smart contracts) on a blockchain network in an attempt to control the process of making payments on the network using machine-readable code (i.e., smart contracts). The community has been working on improving or adjusting consensus algorithms commonly used in distributed systems for use in blockchain networks in order to improve consensus, especially in the last few years [6]. ERC-20 is an open standard that is used to issue tokens on the Ethereum blockchain. In November 2015, an Ethereum developer named Fabian Vogelsteller proposed a new protocol that he developed. It is essential to follow a set of rules for a token to operate properly within the Ethereum ecosystem for it to be successful. Due to this fact, ERC-20 is not a piece of software or code but rather a protocol. Using technical guidelines or specifications would be more appropriate than technical guidelines [7].

A token represents a digital asset and has value, but not all tokens are standardized. The ERC-based currencies are more interchangeable and are compatible with DApps. Tokens could become more compatible with the new standard and allow other features, such as the tokenization of voting. It is more like a polling operation [8]. Tokens complying with ERC20 allow anyone to track how many tokens they own at any given time, and the token holder retains complete control over the assets. It is a sub-currency based on an Ethereum contract, so it is straightforward to implement. It would help if the user transferred it [9]. How to recognize “user,” not someone else. Users need to use a digital signature. As a result of standardization, these assets can be used on various platforms and projects that would otherwise only apply to specific circumstances.

3. Digital Signature

ETH is the cryptocurrency that Ethereum uses to build and maintain the operation of its network. It works similarly to Bitcoin, with miners creating ETH by building blocks and solving puzzles, the action known as "mining." The Digital signature during this process is used for ownership certification without displaying a private key. The concept of a digital signature can be described as a form of electronic signature equivalent to a written signature. Digital signatures enable you to prove to a third party that the creator signed the information user gave them. In addition to verifying the integrity of information, digital signatures function similarly to their written counterparts [6]. Data and programs stored in the computer could also be verified at any time in the future by using digital

signatures that can be generated to ensure that the programs and data have been signed digitally. A verification of the data and programs' integrity would be possible at any time, thus ensuring their integrity.

3.1 How Signatures for ETH Were Produced?

ECDSA (Elliptic Curve Digital Signature Algorithm) uses elliptic curves to create a digital signature of data so that users can verify its authenticity without compromising security [10]. Imagine it as a real signature. Signatures can be recognized by users but they cannot be forged without the other person's knowledge. ECDSA signatures, on the other hand, cannot be forged, unlike handwritten signatures. A value must be selected at this point in order for the process to begin. The Ethereum platform introduced a solution in the form of EIP-2 to deal with this issue. As a result, signature malleability can be prevented using EIP-2. For the same number [2], there is only one valid x value since half of the number was deleted successfully. In order to calculate ECDSA, we use the following equation:

$$y^2 = (x^3 + a * x + b) \quad (1)$$

ECDSA consists of the following steps:

- Users are required to choose the elliptic curve $Ep\ (a, b)$ must wish a base point G.
- Pick the private key k (k<n, n means the order of G), and use the base point G to calculate the public key $K = k\ G$.
- Make a random integer $r\ (r < n)$ and calculate the point $R = r\ G$.
- Calculate SHA256 as Hash using the original data and (x,y) values x and y of point R as parameters, namely, Hash = SHA1(the data like beginning, x, y).
- Use the following formula to determine s≡ r - Hash * k (in mod n).
- r and s, determine that signature value. Operating Step 3 again if either r or s is zero [11].

Verification involves the following steps:

- The receiver performs the following operations after receiving the message (m) and signature value (r, s).
- The formula is $sG + H(m)\ P = (x1, y1)$, r1 x1 mod p.
- R1 mod p should be verified by solving the equation.
- It is acceptable to accept the signature if the equation is correct. In any other case, the signature is invalid 3.2and RSA

To encrypt/decrypt in RSA Algorithm, there are a few steps.

- Unpack the key into its components.
- Convert and generate.
- Return the output value.

The RSA algorithm was first standardized in 1994 and has been one of the most widely used algorithms of all time. There are many applications for email encryption, message encryption, cryptocurrencies, and many others. RSA's strength lies in its use of long-length keys, making it difficult to break for quite some time. As a result, it will be safe for a long time to come. Despite this, a significant number of people still enjoy using it for this reason.

Prime factorization is an integral part of RSA's working process. RSA uses two extremely large numbers multiplied together to achieve the highest level of security. There is a considerable difference in length between the RSA key and the ECDSA key. Implementing ECDSA is more complicated than RSA because of the elliptic curves. Since ECDSA saves a lot of key length space, it became a major trend in encryption methods. Shorter key lengths make for better performance; therefore, this method has become a popular way to encrypt or decrypt.

The following methods can be used to attack ECDSA:

- A discrete logarithm problem for elliptic curves with a new approach.
- Attacks on hash functions.
- The other attacks [12].

4. Private Key and Public Key

Private and public keys are used for encrypting and authenticating data in Ethereum. After the message was created. It was attached with a signature from the sender [13]. No one can generate a fake signature without getting the private key from the sender. Comparison of the signature with the sender's public key, which is derived from the sender's private key, is one way to verify the signature's authenticity [8].

- Make a private key. Any string type input could be a private key.
- Use ECDSA to make a public key and Hash. As mentioned before.
- Address.

The remaining 40 hexadecimal characters/20 bytes/160 bits of data were used to create the address. Whenever possible, it is important to protect the key generation process against unauthorized access so that unauthorized access cannot be gained to the key generation process. In the case of a hardware or software failure, partial information can be retained if no failures occur on the equipment that generates keys, thereby preventing the loss of partial information in the case of a hardware or software failure. Keying material data may be captured as a result of a system crash that results in the creation of a core dump. Compromised keying material can be produced using the same SEEDs.

5. Keystore File System

By using the Keystore file system that is part of the Ethereum platform, users can secure their private keys for their cryptocurrency wallets. They can easily manage their keys by memorizing their choice's password to manage them more easily. ETH private keys stored in the Keystore file (.ethereum/Keystore) are JSON files, as illustrated in the example, containing various data, such as encrypted user addresses. To access their account, the user must have a password generated by them to authenticate the Keystore file. When a user loses their Keystore file, they lose access to their private keys and cannot access their accounts [14].

ETH private keys can be stored in any file chosen by the user. On the other hand, the unencrypted files create a vulnerability in the key store, which is exposed by these unencrypted files [15]. Once an attacker obtains the user's unencrypted key, he cannot only take control of the account but also sign transactions on behalf of the user. In this sense, the Keystore file compromises security properties and availability characteristics.

A hashcat application can be used to crack the password of a Keystore file found by an attacker if the attacker recovers the Keystore file. The Keystore file stores the private key of the Ethereum wallet, which can be hacked both by dictionary and brute force attacks. The information on a wallet's password can be used by hackers to gain full control of the wallet [15].

6. Conclusion

The Digital signature during this process is used for ownership certification without displaying a private key. Using EIP-2, the system can limit the value of s, preventing signature malleability. Then ECDSA generates the unique public key and Hash. Finally, the address was made after getting rid of some digits from the key. This is a complete digital signature process. Because RSA uses very long keys, making it a complicated algorithm to break for quite some time, its strength lies in implementing long-term keys. RSA keys are much longer than ECDSA keys, which is why they are much more secure. Implementing ECDSA is more complicated than RSA because of the elliptic curves. Since ECDSA saves a lot of key length space, it became a major trend in encryption methods. Shorter key lengths improve performance; therefore, this method has become a popular way to encrypt or decrypt. For future works, the coding of ECDSA and Keys is needed. It could display better results. Then buy some ETH from exchanges or Wallets. And using the newest trends in pricing to predict the income for the day. Because real practice is an important part of the research, in Ethereum smart contracts, the ECDSA signature verification algorithm is built-in. This can be accessed through the system

ecrecover, which provides access to the algorithm. ERC20-Permit's ecrecover function can be used to obtain the signer's public key of the signed and hashed data and, as a result, verify the integrity of the signature and Hash. The message is hashed using V, R, and S from ECDSA. Any arbitrary data can be signed using a private key. As a result, there are many new opportunities and pitfalls. Adding these to future works is impossible. This study's general objective is to synthesize the mainstream academic views on Ethereum digital signatures based on secondary research and provide a brief analysis of their rationale. It has considerably better compatibility and functionality than the Bitcoin chain ecology. However, its flaws will also provide a direction for its future improvement.

References

[1] Antonopoulos, A. M., & Wood, G. (2018). Mastering ethereum: building smart contracts and dapps. O'reilly Media.

[2] Saranya, S., Muvvala, S. P., Chauhan, V., et al. (2022, July). Crowdfunding Charity Platform Using Blockchain. In 2022 International Conference on Inventive Computation Technologies (ICICT) (pp. 1-7). IEEE.

[3] Vujičić, D., Jagodić, D., & Ranđić, S. (2018, March). Blockchain technology, bitcoin, and Ethereum: A brief overview. In 2018 17th international symposium infoteh-jahorina (infoteh) (pp. 1-6). IEEE.

[4] Johnson, D., Menezes, A., & Vanstone, S. (2001). The elliptic curve digital signature algorithm (ECDSA). International journal of information security, 1(1), 36-63.

[5] Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. Ethereum project yellow paper, 151(2014), 1-32.

[6] Dannen, C. (2017). Introducing Ethereum and solidity (Vol. 1, pp. 159-160). Berkeley: Apress.

[7] ANSIX, A. (1998). 31: 1998: Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA). American National Standards Institute.

[8] Luo, Z., Murukutla, R., & Kate, A. (2022). Last Mile of Blockchains: RPC and Node-as-a-service. arXiv preprint arXiv:2212.03383.

[9] Rahimian, R., & Clark, J. (2021). TokenHook: Secure ERC-20 smart contract. arXiv preprint arXiv: 2107. 02997.

[10] Johnson, D., Menezes, A., & Vanstone, S. (2001). The elliptic curve digital signature algorithm (ECDSA). International journal of information security, 1(1), 36-63.

[11] Burgoa, F. J. C. (2021). Addressing Security and Privacy Issues by Analyzing Vulnerabilities in IoT Applications (Doctoral dissertation, The University of Texas at El Paso).

[12] Toradmalle, D., Singh, R., Shastri, et al. (2018, August). Prominence of ECDSA over RSA digital signature algorithm. In 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC) I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2018 2nd International Conference on (pp. 253-257). IEEE.

[13] ANSIX, A. (1998). 31: 1998: Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA). American National Standards Institute.

[14] Siriwardena, P. (2019). Securing apis with Transport Layer Security (TLS). Advanced API Security, 69–79. https://doi.org/10.1007/978-1-4842-2050-4_3.

[15] Praitheeshan, P., Xin, Y.W., et al. (2019). Attainable Hacks on Keystore Files in Ethereum Wallets--A Systematic Analysis. In: Doss, R., Piramuthu, S., Zhou, W. (eds) Future Network Systems and Security. FNSS 2019. Communications in Computer and Information Science, vol 1113. Springer, Cham. https:// doi.org/ 10.1007/978-3-030-34353-8_7.