How To Preserve Transaction Privacy

Transaction privacy preserving technology and solutions

Radar Lab

Abstract

This article introduces a series of technologies including Ring Signature, Zero-knowledge Proofs, and Stealth Address to preserve transaction privacy in a decentralized cryptocurrency system, which enables transactions on-chain to achieve unconditional anonymity with certain strength. It would continue to preserve transaction privacy even if the algorithm basis of cryptocurrency such as elliptic curve related algorithms has been breached.

Contents

Transaction privacy preserving technology introduction

Transaction privacy was initially achieved in cryptocurrency transactions through anonymous addresses. Users randomly generate key pairs and then use a public key to generate an address to preserve their identity. However, more and more studies showed that this identity protection cannot achieve the expected effect[1][2][3][4] in many cases. Consequently, many technologies have been developed to preserve transaction privacy.

The disposable key pairs & address is one of the most widely used privacy protection technology, which can well protect identity privacy. Let's take BIP32[5] and Stealth Address[6] for example, they are based on a certain key, superimposing a serial number or random number hash to generate a new key pair and using it as a disposable address.

Ring signature[7] is a special multi-signature technology to protect identity privacy. It doesn't require a center and can achieve the purpose of hiding the signer identity. Subsequently, many ring signature-based technologies with different goals have been developed and they work by forming a ring with multiple key pairs, calculating signature with all public keys and one of the private keys, and filling the other private keys with random numbers. The calculation is a loop nesting process, you can verify it from the outside world but you cannot get the exact location of the real private key in the ring,

Homomorphic encryption is another kind of technology that preserves transaction privacy and protect transaction contents. It's about keeping the result of algebraic manipulation on the ciphertext consistent with the result of algebraic manipulation and encryption on the original text, performing algebraic manipulation without decrypting the content[8].

Confidential transaction[9] is an implementation of homomorphic encryption. Generally, its principle is to sign the content such as the amount as `v` and hide it in `r*G + v*H`. When `v` changes, we have to know the initial `r1` value and select a new `r2` to get the result of `(r1*G + v*H) - (r2*G + v*H) = (r3*G + 0*H)` while the outside world cannot calculate the exact value of `v`.

[1] Reid, Fergal, and Martin Harrigan. "An analysis of anonymity in the bitcoin system." Security and privacy in social networks. Springer, New York, NY, 2013. 197-223.

[2] Meiklejohn, Sarah, et al. "A fistful of bitcoins: characterizing payments among men with no names." Proceedings of the 2013 conference on Internet measurement conference. 2013.

[3] Ron, Dorit, and Adi Shamir. "Quantitative analysis of the full bitcoin transaction graph." International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2013.

[4] Conti, Mauro, et al. "A survey on security and privacy issues of bitcoin." IEEE Communications Surveys & Tutorials 20.4 (2018): 3416-3452.

[5] Pieter, W. "Bip32: Hierarchical Deterministic Wallets." HYPERLINK https://github.com/bitcoin/bips/blob/master/bip0032.mediawiki (2013).

[6] Courtois, Nicolas T., and Rebekah Mercer. "Stealth Address and Key Management Techniques in Blockchain Systems." ICISSP 2017 (2017): 559-566.

[7] Rivest, Ronald L., Adi Shamir, and Yael Tauman. "How to leak a secret." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2001.

[8] Gentry, Craig. "Fully homomorphic encryption using ideal lattices." Proceedings of the forty-first annual ACM symposium on Theory of computing. 2009.

[9] Maxwell, Greg. "Confidential transactions." URL: https://people.xiph.org/greg/confidential_values.txt (Accessed 09/05/2016) (2015).

Zero-knowledge Proof is another technology to protect transaction content, supporting a certain assertion to the outside world without exposing the content.Zerocoin[10] is an early implementation that applies zero-knowledge proofs to cryptocurrency. At present, the proven technology is zk-SNARKs[11] introduced by Zerocash.

Implementation

Through the study of these privacy-protection technologies, we believe we can eliminate the correlation between address and identity via Stealth Address, protect the transaction amount by Confidential amount and remove counterparty correlations via Ring Signature, as shown in Figure 1. In this way, we achieve a higher intensity of privacy protection. Furthermore, these technologies are proven to be high security.

Technical Details

Ring Signature

We eliminate the correlation between the transaction sender and receiver via a two-step operation of `Deposit` and `Withdraw`. Since the correlation between the two steps was eliminated by ring signature, the outside world has no way to learn who is the sender of the transaction.

The signature here adopts ring signature technology based on elliptic curve, mainly referring to Linkable spontaneous anonymous group for ad hoc groups[12] Traceable ring signature[13] and Borromean ring signatures[14].

Detailed descriptions are as follows:

- Select a elliptic curve $E(a, b)$ with the base point G.
- We take $Sk \in_R \mathbb{Z}_q$ as the private key and $Pk = Sk \times G$ as the public key.
- We can get hash algorithm $Hg : \{0, 1\}^* \to E(a, b)$, $Hq : \{0, 1\}^* \to \mathbb{Z}_q$.

Create a ring with Deposit

1. Generate a random key pair (Pk, Sk).

[10] Miers, Ian, et al. "Zerocoin: Anonymous distributed e-cash from bitcoin." 2013 IEEE Symposium on Security and Privacy. IEEE, 2013.

[11] Sasson, Eli Ben, et al. "Zerocash: Decentralized anonymous payments from bitcoin." 2014 IEEE Symposium on Security and Privacy. IEEE, 2014.

[12] Liu, Joseph K., Victor K. Wei, and Duncan S. Wong. "Linkable spontaneous anonymous group for ad hoc groups." Australasian Conference on Information Security and Privacy. Springer, Berlin, Heidelberg, 2004.

[13] Fujisaki, Eiichiro, and Koutarou Suzuki. "Traceable ring signature." International Workshop on Public Key Cryptography. Springer, Berlin, Heidelberg, 2007.

[14] Maxwell, Gregory, and Andrew Poelstra. "Borromean ring signatures." http://diyhpl.us/~bryan/papers2/bitcoin/Borromean%20ring%20signatures.pdf, 2015.

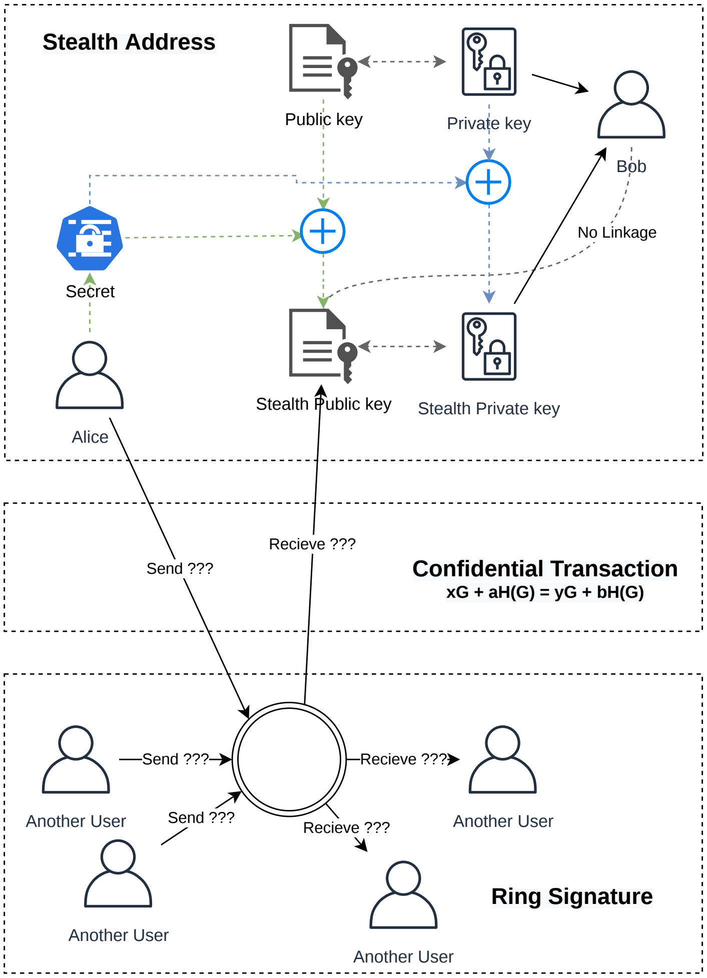

Figure 1: Transaction privacy protection scheme

2. Send the amount and the public key Pk to a ring.
3. Go to the next step when a certain amount of Pk is accumulated.

Complete transaction via Withdraw

We let $L = \{Pk_1, \ldots, Pk_n\}$ be an ordered public key set for the ring.

1. Generate ring signature σ and key fingerprint I with the private key Sk and L.
2. Send ring signature σ and fingerprint I to received amount from the ring.

Signing algorithm

Let j be the sender's index in the ring. We define its key pair as (Pk_j, Sk_j) and contents to be signed as $m \in \{0,1\}^*$

1. Compute $H = Hg(L)$, and the key fingerprint $I = Sk_j \times H$.
2. Pick up random $r \in_R \mathbb{Z}_q$, compute
    1. $L_j = r \times G$
    2. $R_j = r \times H$
    3. $c_{j+1} = Hq(L, I, m, L_j, R_j)$
3. Pick up random $s_{j+1} \in_R \mathbb{Z}_q$, repeat the followings
    1. $L_{j+1} = s_{j+1} \times G + c_{j+1} \times Pk_{j+1}$
    2. $R_{j+1} = s_{j+1} \times H + c_{j+1} \times I$
    3. $c_{j+2} = Hq(L, I, m, L_{j+1}, R_{j+1})$
4. Until we have L_{j-1}, R_{j-1}, c_j, let $s_j = (r - c_j \times Sk_j) \mod q$.
5. Output $\sigma = (I, c_1, s_1, ..., s_n)$ as the signature.

Verification algorithm

Loop the following computations

1. $L'_i = s_i \times G + c_i \times Pk_i$
2. $R'_i = s_i \times H + c_i \times I$
3. $c_{i+1} = Hq(L, I, m, L_i, R_i)$

for all i until we have c_{n+1}. Then we can get the signature validity by verifying $c_1 \overset{?}{=} c_{n+1}$.

Security analysis

Forgery prevention

1. Since SK_j can generate a unique I, so people in the ring cannot forge two different signatures.
2. If we calculate $I = Sk'_j \times H$ with a wrong Sk'_j, then $L'_j = s_j \times G + c_j \times Sk'_j \times G$ is not equal to $s_j \times G + c_j \times Pk_j$, and the validation cannot be passed. Therefore, a verified signature means the signer must have known a certain SK_j, and no one outside the ring can forge the signature.

Correlation Elimination

Since $s_j = (r - c_j \times Sk_j) \mod q$, we can get

1. $r = (s_j + c_j \times Sk_j) \mod q$
2. $L_j = r \times G = s_j \times G + c_j \times Sk_j \times G = s_j \times G + c_j \times Pk_j$
3. $R_j = r \times H = s_j \times H + c_j \times SK_j \times H = s_j \times H + c_j \times I$

Without the value of Sk_j, we cannot get s_j from L_j or R_j. Since s_j has no characteristics and it has nothing different with other s_i, we cannot get j to verify who is the signer.

Confidential amount

The elliptic curve can also be used to protect transaction amount.

- We assume that $H \in_R E(a, b)$, and it makes $\log_G H$ uncomputable.
- Pick up $r \in_R \mathbb{Z}_q$, Sign the cleartext amount v as confidential amount $C(v, r) = r \times G + v \times H$.

Peer-to-peer transaction

Assuming the balance is v, the transaction amount is t, the cleartext amount of a peer-to-peer transaction can be calculated as $v' = v - t$. The confidential amount can be calculated as $(r' \times G + v' \times H) = (r \times G + v \times H) - (r_t \times G + t \times H)$. In this case, we can achieve the goal of protecting transaction amount.

Ring transaction

Let d_i be the deposit amounts for a ring and w_i be the withdraw amounts for the same ring, thus we can compute the cleartext amount as $\sum_{i=1}^{n} d_i = \sum_{i=1}^{n} w_i$.

The computation of confidential amounts becomes $\sum_{i=1}^{n}(r_i \times G + d_i \times H) = \sum_{i=1}^{n}(r'_i \times G + w_i \times H)$.

Security analysis

To an outsider, the transaction amount is $(r_2 \times G + t \times H)$ and it's impossible to calculate the exact t without r_2, protecting the privacy of transaction amount. Meanwhile, the sender amount is changed from $(r \times G + v \times H)$ to $(r' \times G + v' \times H)$. The sender does not need to release r or r' to anyone and no one can calculate the value of v or v' without the value of r or r', protecting the balance information of the sender.

Stealth Address

Let's assume the key pair of receiver is (Sk, Pk), Sk is the private key and Pk is the public key. The stealth address has two points. One is to generate it without the receiver's Sk and the other is that the receiver's Pk cannot be speculate.

The stealth address needs to be implemented via DH key exchange on the elliptic curve and the specific algorithm is as follows:

1. The receiver discloses two public keys (A, B) where the corresponding private keys are (a, b).
2. The sender picks a randome nonce r to calculate the stealth public key $Pk_s = Hq(r \times A) \times G + B$.
3. The sender calculates $R = r \times G$ and discloses the value of R.
4. The receiver calculates $Pk'_s = Hq(a \times R) \times G + B$ and checks $Pk_s \overset{?}{=} Pk'_s$ to determine if it is related to the receiver.
5. The receiver calculates $Sk_s = Hq(a \times R) + b$, and use $Pk_s = Sk_s \times G$ to verify Pk_s.

Security analysis

1. Since the receiver's private key is not used in the calculation of Pk_s, we can calculate it under any circumstance.
2. The stealth private key $Sk_s = Hq(r \times A) + b = Hq(a \times R) + b$ is impossible to calculate without knowing b. So it can be guaranteed that only the receiver can calculate.
3. The stealth public key $Pk_s = Hq(r \times A) \times G + B = Hq(a \times R) \times G + B$ is impossible to calculate B without knowing $(r \times A)$ or $(a \times R)$. Therefore, the specific receiver can not be speculated.