Monero

Saravanan Vijayakumaran

Department of Electrical Engineering
Indian Institute of Technology Bombay

April 16, 2024

Monero

- Privacy-oriented cryptocurrency launched in April 2014
- Transaction amounts are hidden
- Transaction inputs and outputs have one-time addresses
- Ring signatures are used to weaken blockchain analysis
- Based on CryptoNote protocol by Nicolas van Saberhagen
    - Initial proposal had amounts in the clear
- Popular for cryptojacking, ransomware, compute-based donations

One-Time Addresses

- Also called stealth addresses
- Each user has two private-public key pairs from an elliptic curve group with base point G and cardinality p
- Let Bob's private keys be (x_1, x_2) with public keys (P_1, P_2) given by $(x_1 G, x_2 G)$
- Let H_s be a scalar-valued cryptographic hash function
- Suppose Alice wants to send a payment to Bob
    1. Alice generates a random $r \in \mathbb{Z}_p^*$ and computes a one-time public key $P = H_s(rP_1)G + P_2$
    2. Alice specifies P as destination address and $R = rG$ in transaction output
    3. Bob reads every transaction and computes $P' = H_s(x_1 R)G + P_2$
    4. If $P' = P$, the Bob knows the private key $x = H_s(x_1 R) + x_2$ such that $P = xG$
    5. Bob can spend the coins in the one-time address P using x
- The pair (x_1, P_2) is called the tracking key
- Tracking key can be safely shared with third parties

Ring Signatures

- Traditional digital signatures prove knowledge of a private key
- Ring signatures prove signer knows 1 out of N private keys
- Example application of ring signatures
  - Suppose a whistleblower in a corporation wants to leak info to a journalist
  - Whistleblower wants to keep his/her identity secret
  - An anonymous message will not convince journalist
  - Suppose a public key corresponding to each employee is made public
  - The whistleblower can sign his/her message using a ring signature
- Linkable ring signatures are ring signatures which reveal a **key image** of the private key
  - Example: For public key $P = xG$, the key image could be $xH_p(P)$ where H_p is a point-valued hash function
  - Key image does not reveal identity of signer but links signatures from same signer
- Example application of linkable ring signatures
  - Suppose the board of directors of a corporation wants to vote on an issue
  - The directors do not want to reveal their votes (yes/no/abstain)
  - Suppose each director has a public key which is known to the others
  - Each director can sign his/her message using a linkable ring signature
  - Multiple votes by same director will be detected

Ring Signatures

- Consider an elliptic curve group E with cardinality p and base point G
- Let $x_i \in \mathbb{Z}_p^*, i = 0, 1, \ldots, n-1$ be private keys with public keys $P_i = x_i G$
- Suppose a signer knows only x_j and not any of x_i for $i \neq j$
- For a given message m, the signer generates the ring signature as follows:
  1. Signer picks $\alpha, s_i, i \neq j$ randomly from $\mathbb{Z}_p$
  2. Signer computes $L_j = \alpha G$ and $c_{j+1} = H_s(m, L_j)$
  3. Increasing j modulo n, signer computes

$$\begin{aligned}
L_{j+1} &= s_{j+1} G + c_{j+1} P_{j+1} \\
c_{j+2} &= H_s(m, L_{j+1}) \\
&\vdots \\
L_{j-1} &= s_{j-1} G + c_{j-1} P_{j-1} \\
c_j &= H_s(m, L_{j-1})
\end{aligned}$$

  4. Signer computes $s_j = \alpha - c_j x_j$ which implies $L_j = s_j G + c_j P_j$
  5. The ring signature is $\sigma = (c_0, s_0, s_1, \ldots, s_{n-1})$
- Verifier computes L_j's, remaining c_j's, and checks that $H_s(m, L_{n-1}) = c_0$

Linkable Ring Signatures

- Consider an elliptic curve group E with cardinality p and base point G
- Let $x_i \in \mathbb{Z}_p^*, i = 0, 1, \ldots, n-1$ be private keys with public keys $P_i = x_i G$
- Suppose a signer knows only x_j and not any of x_i for $i \neq j$
- The **key image** corresponding to P_j is $I = x_j H_p(P_j)$
- For a given message m, the signer generates the LSAG signature as follows:
  1. Picks $\alpha, s_i, i \neq j$ randomly from $\mathbb{Z}_p$
  2. Computes $L_j = \alpha G$, $R_j = \alpha H_p(P_j)$, and $c_{j+1} = H_s(m, L_j, R_j)$
  3. Increasing j modulo n, computes

$$
\begin{aligned}
L_{j+1} &= s_{j+1} G + c_{j+1} P_{j+1} \\
R_{j+1} &= s_{j+1} H_p(P_{j+1}) + c_{j+1} I \\
c_{j+2} &= H_s(m, L_{j+1}, R_{j+1}) \\
&\vdots \\
L_{j-1} &= s_{j-1} G + c_{j-1} P_{j-1} \\
R_{j-1} &= s_{j-1} H_p(P_{j-1}) + c_{j-1} I \\
c_j &= H_s(m, L_{j-1}, R_{j-1})
\end{aligned}
$$

  4. Computes $s_j = \alpha - c_j x_j \implies L_j = s_j G + c_j P_j, R_j = s_j H_p(P_j) + c_j I$
  5. The ring signature is $\sigma = (I, c_0, s_0, s_1, \ldots, s_{n-1})$
- Verifier computes L_j, R_j, remaining c_j's, and checks that $H_s(m, L_{n-1}, R_{n-1}) = c_0$
- Signatures with duplicate key images I will be rejected

Source Address Obfuscation

- Suppose Alice wants to spend coins from an address P she owns
- Alice assembles a list $\{P_1, P_2, \ldots, P_N\}$ where $P_j = P$ for exactly one j
- Alice knows x_j such that $P_j = x_j G$
- Key image of P_j is $I = x_j H_p(P_j)$ where H_p is a point-valued hash function
  - Distinct public keys will have distinct key images
- A linkable ring signature over $\{P_1, P_2, \ldots, P_N\}$ will have the key image I of P_j
  - Signature proves Alice one of the private keys
  - Double spending is detected via duplicate key images
- One cannot say if a Monero address belongs to the UTXO set or not

Confidential Transactions

Balance Condition

- Each one-time address has some amount of coins associated with it
- Suppose a transaction has input amounts a_1, a_2, a_3 and output amounts b_1, b_2
- For transaction validity, we require

$$a_1 + a_2 + a_3 \geq b_1 + b_2$$

- In the first version of Monero, the amounts were not hidden
- To spend from an address using a linkable ring signature, user had to choose ring members from other addresses which had the same amount
- Unencrypted amounts are bad for privacy
- Encryption method should allow third-party verification of the balance condition using only the ciphertexts

Pedersen Commitments

- Let a denote an amount we want to hide
- Let G be the base point of an elliptic curve E of prime order p
- Let H be another curve point in E with an unknown discrete logarithm with respect to G
  - No one knows $k \in \{1, 2, \ldots, p-1\}$ such that $H = kG$
- The Pedersen commitment to amount $a \in \mathbb{Z}_p$ with blinding factor $x \in \mathbb{Z}_p$ is

$$C(a, x) = xG + aH$$

- **Hiding:** If x is chosen uniformly from $\mathbb{Z}_p$, then $C(a, x)$ reveals nothing about a
- **Binding:** If $\log_G H$ is unknown, $C(a, x)$ cannot be revealed to be a commitment to some $a' \neq a$
  - If an adversary finds x', a' such that $C(a, x) = x'G + a'H$ with $a' \neq a$, then

$$xG + aH = x'G + a'H \implies H = \left(a - a'\right)^{-1} \left(x' - x\right) G$$

- **Homomorphic:** $C(a_1, x_1) + C(a_2, x_2) = C(a_1 + a_2, x_1 + x_2)$

$$x_1 G + a_1 H + x_2 G + a_2 H = (x_1 + x_2) G + (a_1 + a_2) H$$

Proving Statements About Commitments

- How to prove that C is a commitment to the zero amount without revealing blinding factor?

  Ans: If $C = C(0, x) = xG$, then give a digital signature verifiable by C as the public key

  If C is a commitment to a non-zero amount a, signature with C as public key will mean discrete log of H is known

  $$C = xG + aH = yG \implies H = a^{-1}(y - x)G$$

- How to prove that C is a commitment to the an amount a without revealing blinding factor?

  Ans: If $C = C(a, x) = xG + aH$, then give a digital signature verifiable by $C - aH$ as the public key

- How to prove that two commitments C_1 and C_2 are commitments to the same amount a without revealing blinding factors?

  Ans:

  $$C_1 = C(a, x_1) = x_1 G + aH$$
  $$C_2 = C(a, x_2) = x_2 G + aH$$

  Give a digital signature verifiable by $C_1 - C_2$ as the public key

Communicating the Commitment Opening

- Suppose Alice want to send coins to Bob
- To send coins with amount hidden in a Pedersen commitment, the opening has to be communicated to him
- Let Bob's public keys be (P_1, P_2)
- Suppose $C(a, y)$ is the commitment Alice creates for Bob
- To communicate a and y to Bob, Alice includes

$$
\begin{aligned}
a' &= a \oplus H_K(H_K(rP_1)) \\
y' &= y \oplus H_K(rP_1)
\end{aligned}
$$

  in the transaction, where $\oplus$ is bitwise XOR and H_K is the Keccak hash function.
- As the point R is contained in the transaction, Bob can use his private key x_1 to recover a and y from a' and y' as

$$
\begin{aligned}
a &= a' \oplus H_K(H_K(x_1 R)), \\
y &= y' \oplus H_K(x_1 R).
\end{aligned}
$$

Proving the Balance Condition

- Suppose $C_1^{\text{in}}, C_2^{\text{in}}, C_3^{\text{in}}$ are commitments to input amounts a_1, a_2, a_3
- Suppose $C_1^{\text{out}}, C_2^{\text{out}}$ are commitments to output amounts b_1, b_2
- To prove $a_1 + a_2 + a_3 \geq b_1 + b_2$, we will prove

$$a_1 + a_2 + a_3 = b_1 + b_2 + f$$

  for some $f \geq 0$
- A digital signature with

$$C_1^{\text{in}} + C_2^{\text{in}} + C_3^{\text{in}} - C_1^{\text{out}} - C_2^{\text{out}} - fH$$

  as public key is enough
- **Almost enough!** It only shows that

$$\begin{aligned} a_1 H + a_2 H + a_3 H &= b_1 H + b_2 H + fH \\ \implies a_1 + a_2 + a_3 &= b_1 + b_2 + f \bmod p, \end{aligned}$$

  since $pH = \mathcal{O}$ (the identity of the elliptic curve group)

Exploiting the Modular Balance Condition

- Using only the modular balance check is risky

$$a_1 + a_2 + a_3 = b_1 + b_2 + f \bmod p$$

- **Example:** $a_1 = 1, a_2 = 1, a_3 = 1$ and $b_1 = p - 4, b_2 = 6, f = 1$
- Attacker can create C_1^{out} as a commitment to the amount $p - 4$
- Typically $p \approx 2^{256} \implies p - 4$ is much larger than the sum of the input amounts
- Attacker can now spend large amounts from C_1^{out}

Solution using Range Proofs

- **Example:** $a_1 = 1, a_2 = 1, a_3 = 1$ and $b_1 = p - 4, b_2 = 6, f = 1$
- Typically $p \approx 2^{256}$ and amounts are in a smaller range like $\{0, 1, 2, \ldots, 2^{64} - 1\}$
- Proving that C_1^{out} and C_2^{out} commit to amounts in the range $\{0, 1, 2, \ldots, 2^{64} - 1\}$ solves the problem
- How to prove that C is a commitment to the an amount a in the range $\{0, 1, 2, 3, 4, 5\}$?

  Ans: Give a **ring signature** verifiable by the public keys

  $$\{C, C - H, C - 2H, C - 3H, C - 4H, C - 5H\}$$

- A naïve ring signature over the keys $\{C - iH \mid i = 0, 1, \ldots, 2^{64} - 1\}$ would be very inefficient

A Better Range Proof

- Let $a = \sum_{i=0}^{63} a_i 2^i$ where each a_i is either 0 or 1
- Create commitments $C_i = C(a_i 2^i, x_i) = x_i G + a_i 2^i H$
- If we consider $\{C_i, C_i - 2^i H\}$ as a pair of public keys, we know exactly one of the corresponding private keys
- A ring signature for each i proves that either C_i or $C_i - 2^i H$ is a commitment to 0
- By picking blinding factors such that $x = \sum_{i=0}^{63} x_i$, we have

$$C(a, x) = \sum_{i=0}^{63} C_i = \sum_{i=0}^{63} x_i G + \sum_{i=0}^{63} a_i 2^i H$$

- This proves $C(a, x)$ is a commitment to an amount in $\{0, 1, 2, \ldots, 2^{64} - 1\}$
- **Bulletproofs** improve this even further and reduce proof sizes to $\mathcal{O}(\log_2 n)$ for an n-bit range proof

Monero RingCT

- Each output in Monero has a one-time address P and a Pedersen commitment C
- Consider a transaction which unlocks funds in m one-time addresses
- MLSAG signatures are linkable ring signatures over a set of n key-vectors
- Spender assembles an $m \times n$ matrix of one-time addresses

$$\begin{bmatrix} P_{1,1} & P_{1,2} & \cdots & P_{1,\pi} & \cdots & P_{1,n} \\ P_{2,1} & P_{2,2} & \cdots & P_{2,\pi} & \cdots & P_{2,n} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ P_{m,1} & P_{m,2} & \cdots & P_{m,\pi} & \cdots & P_{m,n} \end{bmatrix}$$

  where the signer knows $x_{i,\pi}$ such that $P_{i,\pi} = x_{i,\pi} G$ for $i = 1, 2, \ldots, m$
- Each one-time address has a Pedersen commitment $C_{i,j}$ associated with it
- Spender creates commitments $C_1', C_2', \ldots, C_m'$ such that C_i' and $C_{i,\pi}$ commit to the same amount
- The jth column in above matrix is appended with the column vector $\begin{bmatrix} C_1' - C_{1,j} & C_2' - C_{2,j} & \cdots & C_m' - C_{m,j} \end{bmatrix}^T$
- Prover proves knowledge of private keys of all public keys in one of the columns
- For fees f and output commitment C^{out}, the following condition is checked along with range proofs

$$\left(\sum_{i=1}^{m} C_i' \right) - C^{\text{out}} - fH = 0$$

References

- Monero Wikipedia page
  https://en.wikipedia.org/wiki/Monero_(cryptocurrency)
- UNICEF cryptomining donation https://www.thehopepage.org/
- Monero website https://getmonero.org/
- CryptoNote Protocol https://bytecoin.org/old/whitepaper.pdf
- Zero to Monero
  https://www.getmonero.org/library/Zero-to-Monero-2-0-0.pdf
- Monero's Building Blocks by Bassam El Khoury Seguias (10 articles)
  https://delfr.com/category/monero/
- A first look at browser-based cryptojacking
  https://arxiv.org/abs/1803.02887
- Monero block explorers https://xmrchain.net/,
  https://moneroblocks.info/
- Confidential transactions writeup, Greg Maxwell
  https://people.xiph.org/~greg/confidential_values.txt
- An investigation into confidential transactions, Adam Gibson
  https://github.com/AdamISZ/ConfidentialTransactionsDoc
- Bulletproofs https://crypto.stanford.edu/bulletproofs/