Decentralized mixers in Bitcoin

How to dispense with the trusted third party

Olivier Coutu

Master's student under the supervision of Alain Tapp
Computer science department (DIRO)
University of Montreal

Bitcoin 2013 conference, San Jose, CA

Université de Montréal

Anonymity in Bitcoin

Bitcoin is **not** anonymous!

- Reid and Harrigan, 2012
- Ron and Shamir, 2012
- Narayanan and Shmatikov, 2009

Personal identification leaks

- Entering and exiting the Bitcoin network
- Addresses for donations

How to become anonymous?

- People know address A_1 is associated with Alice
- $A_1 \rightarrow A_2$
- A_2 is still associated with Alice
- No anonymity is gained

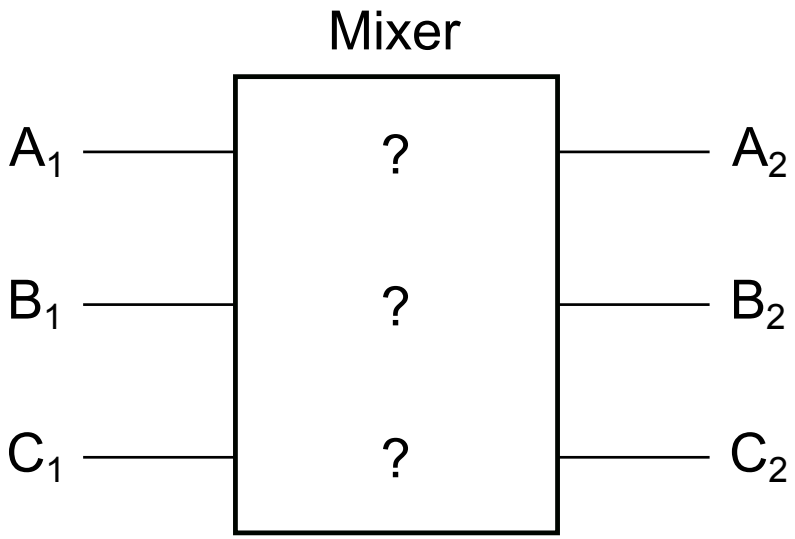
What are the consequences of this lack of anonymity?

Université de Montréal

Centralized mixers

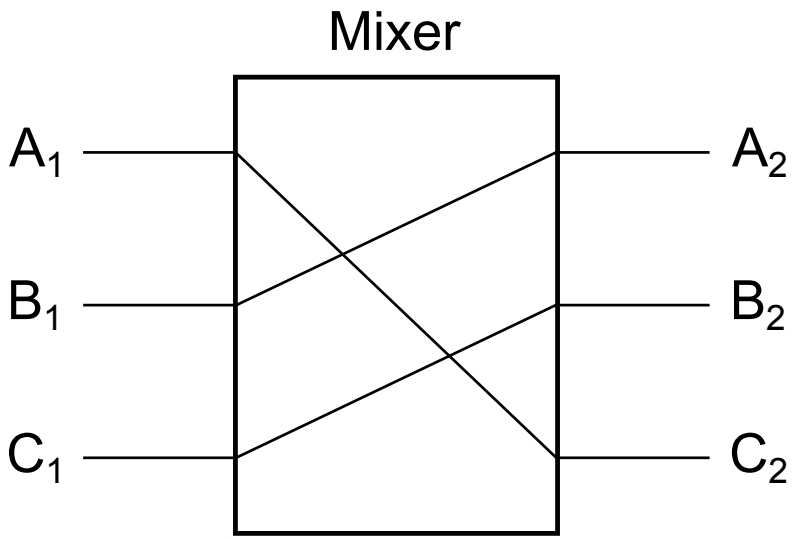
These entities act as trusted third parties (TTP)

1. Receive public input and private output addresses
2. Receive coins from public input addresses
3. Mix coins
4. Send back the mixed coins to output addresses



Université de Montréal

Result after mixing



- Hard to guess which address is A_2, B_2, C_2
- Only the mixer knows that correspondence
- Number of input and output addresses must be the same
- Number of BTC mixed must be the same

Université de Montréal

Can you trust the mixer?

- Mixer can disappear with your coins
- The mixer knows the correspondence between addresses

Can things be done differently?

How to take out the centralized mixer

Replace the TTP using secure Multi-Party Computation (MPC)

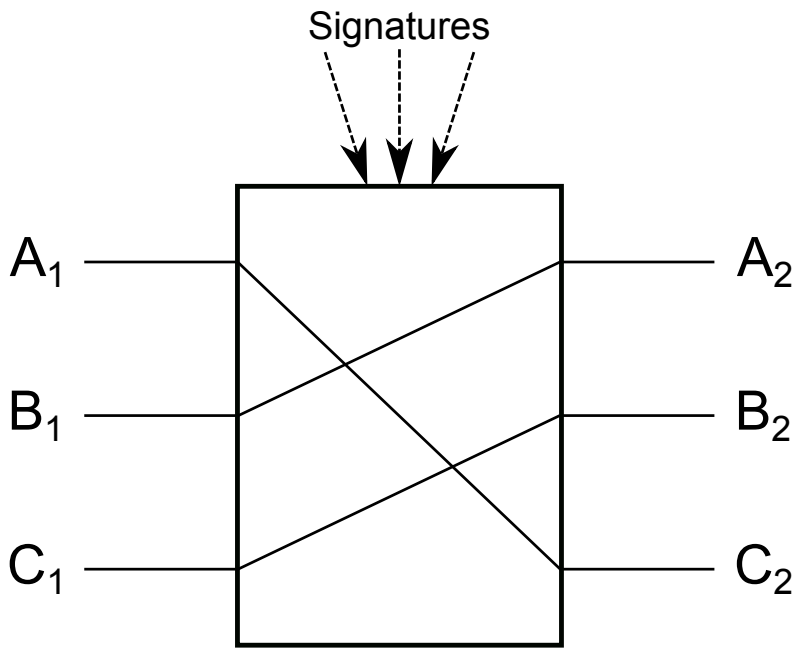
- Enables to remove TTP, at a cost
- Users work together to mix coins
- Output addresses unlinked to input addresses

Decentralized mixer mode of operation

Centralized mixer	**Decentralized mixer**
Users give coins to the mixer	Users keep their coins
Mixer chooses permutation	Users choose permutation
Not anonymous w.r.t the mixer	Anonymous w.r.t everyone

Université de Montréal

Transaction blueprint



1. Securely choose permutation
2. Propose transaction
3. Parties sign transaction
4. Transaction sent to the network

Université de Montréal

Creating the blueprint

Commutative encryption

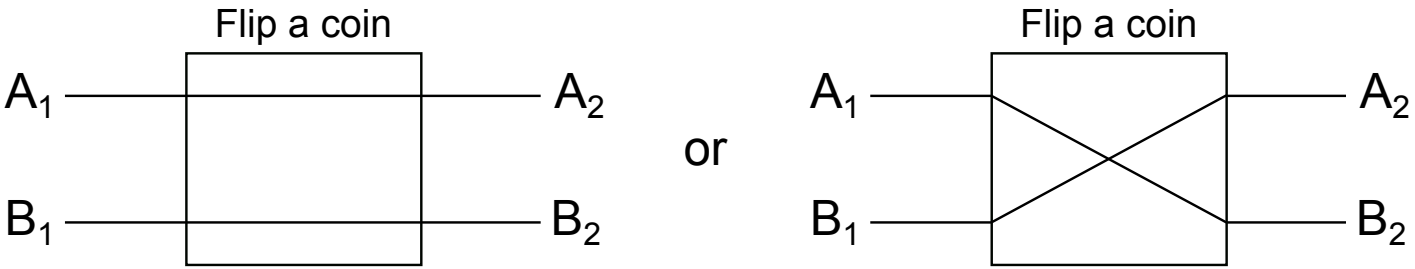
- Proposed by Meni Rosenfeld in 2011
- Deliver addresses in a secure and anonymous manner
- $O(N^2)$ encryptions/decryptions

Secure multi-party sorting

- Proposed by Edward Z. Yang in 2012
- Secure alphanumeric sorting of the output addresses
- $O(N \log^2 N)$ comparisons in $O(\log^2 N)$ rounds

Université de Montréal

Circuit of transactions



Uses 2-party anonymization gate

1. Two parties decide they want to mix their coins
2. Each party has an input and output address ready
3. Flip a coin to decide if they switch their outputs or not
4. They have gained anonymity with respect to other parties

Université de Montréal

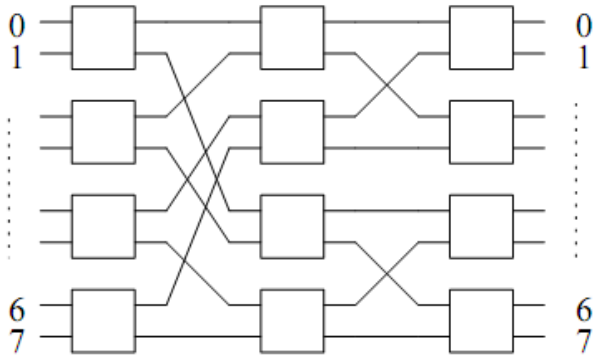
Analysing anonymity

- 2-party anonymization gates combined into circuits
- Maximize resulting min-entropy
- Depends on
  - number of adversaries
  - positions of adversaries
  - coin flips

Random pairing

- N users get together
- Rounds are at predetermined times
- Each round, each player finds a random person to mix with
- After L rounds the protocol stops
- What anonymity is gained?

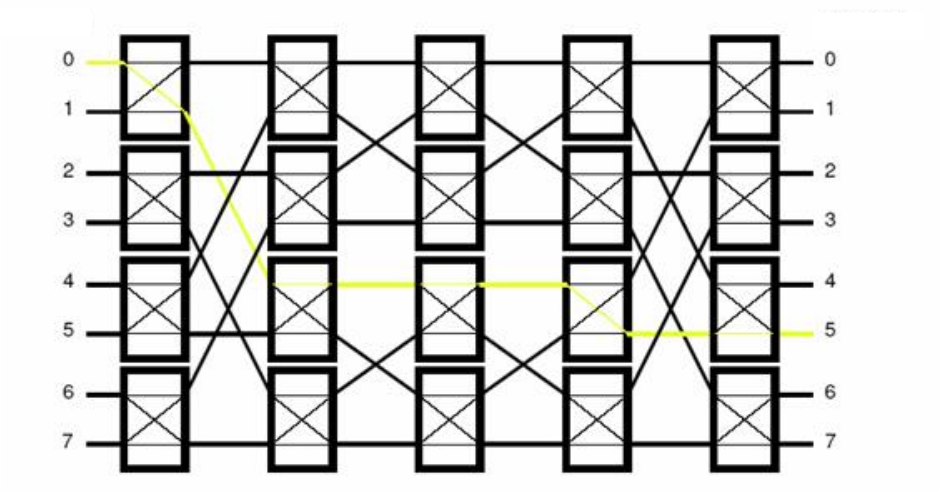
Butterfly network



- Alice could be anywhere
- Optimal depth of lg N
- All permutations are not possible

Université de Montréal

Benes network



- Enables all possible permutations
- Depth $2 \lg N - 1$

Université de Montréal

Comparing the two approaches

Transaction blueprint	**Circuit of transactions**
Computationally hard	Computationally easy
DOS prone	Kick out troublemakers
Quick	Takes a lot of time
Easy on the network	Burdens the network
Low transaction cost	High transaction cost

Hybrid approach

What about using a trusted third party to create a blueprint?

- TTP can't run with the money
- No anonymity gained with respect to the TTP
- Easy to implement

Can be chained together for increased anonymity

What's next?

1. New ways to create blueprints
2. Analysis of the anonymizing circuits
3. Real world implementation

We must discuss how we deal with anonymity