The Blockchain Privacy Problem

Fadi Barbàra

Università degli Studi di Torino

April 2019

Goal of the Talk

Difficulty of the treatment of the privacy concept applied to blockchains, given the interaction of two major privacy threats:

- Internal privacy threats: the set of potential privacy leaks that emerge looking *only* at one particular blockchain
- External privacy threats: the complementary set of privacy leaks

Useful to analyze blockchain projects, whitepapers, technical papers or press releases.

Overview of the talk

- First part
  - Motivation and terminology
  - Assumptions
  - Overview of External blockchain privacy threats
- Second part
  - Internal blockchain privacy threats
  - Proposed method to solve the problems

Blockchain characteristics

Definition

- Append only distributed database
- Data integrity via backlinking

Blockchain problems

- Scalability - transaction throughput
- Interoperability - data in silos
- Privacy and/or Anonymity
- Law and regulations
- User experience

Privacy

> *Bitcoin comes with high-integrity at the cost of a public ledger with little privacy*

- Carmela Troncoso, Marios Isaakidis, George Danezis, and Harry Halpin *Systematizing Decentralization and Privacy: Lessons from 15 Years of Research and Deployments*

Privacy

- Privacy != Anonymity
- Privacy = "only people directly interested in the communication are able to read and understand it"
  - Who are the "directly interested" people is debatable, but this talk is from a purely technical point of view
- Anonimity = "don't know whom I am communication with"

What is a Transaction

- Bitcoin is finance related, but *transactions* are database operations
- Transactions are database appends of information
- Can be any kind of information
  - Medical helth records
  - Marital status + adult age + eye color + ...
  - Percentage free cpu power to be allocated in the next five minutes
  - ...

Privacy in the Talk

- I have to use the common terminology
- Blockchain Transaction = Append Operation on a Distributed Database
  - E.g. medical health record update
- There is no "I've got nothing to hide"
  - Even in your house, you sometimes close a door (e.g. bathroom)
  - You whisper sometimes

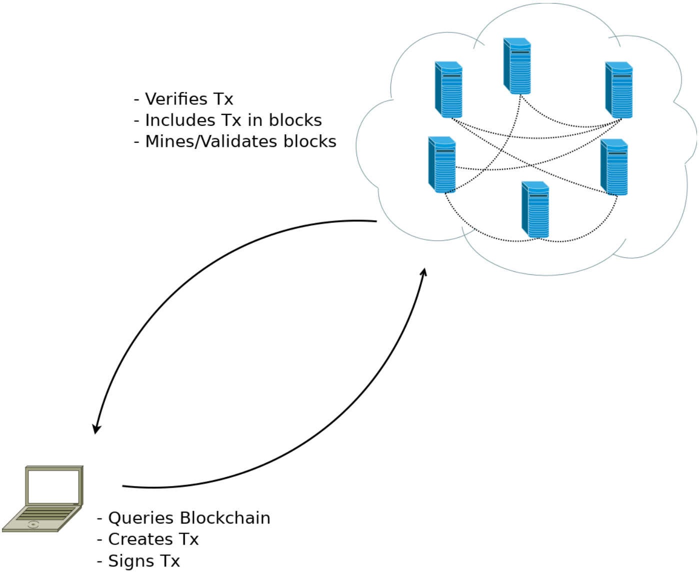
Distopic example

- Can't buy your favourite food (e.g. a sugar based cake) on your birthday because the store does know your health records; you risk diabetes (but you don't have yet), therefore you could be a liability for them

More Distopic Example

- You *can* buy your favourite food on your birthday, because even if you risk diabetes, the store *knows your pharmaceutical purchases* and therefore you are less a potential liability.
  - Still, you pay a premium on that cake becuse they think they risk a lawsuit

Journey of a Blockchain Transaction



Assumpition on threats

- Necessary security assumptions to talk about privacy
  - Idelized, not entirely true
  - Assume security concious person
- Assumptions on:
  - Operating System and software
  - Internet infrastructure

Assumpition on threats - OS

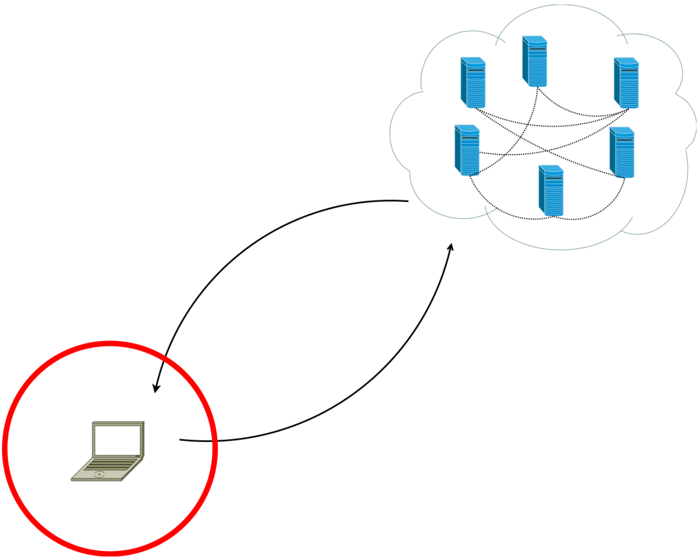
- Ideally Secure operating system and software
  - Nobody tampered with it nor your download
  - Nobody has access remotely
    - Firewall
    - No data usage analytics
    - No location access
    - No wifi probing
    - Encrypted HD

Assumpition on threats - Internet

- Ideally secure Internet Infrastructure
  - All transnational cables
    - Everything encrypted
    - Nobody can read
  - DNS service
    - Perfectly secure
    - No leaks
  - ISP
    - Does not see the content your traffic

Three potential threats

Generated by your use of the computer



The computer

NOTE: a smartphone is an ultra portable computer

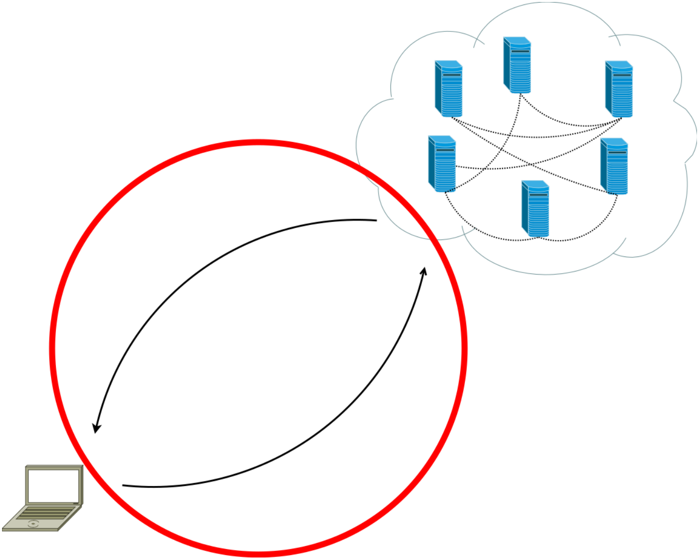
- Browser history
- Cookies
  - Of the store
  - Of your wallet (e.g. Trezor or Ledger)
- Blockchain client
  - (Your) transaction indexing
  - Light clients and Blum Filters
  - Shell history
  - Stored keys (also security problem)

The computer - defenses

- Browser: buy in Private/Incognito windows
- Cookies: cancel them every session
- Blockchain client
  - Enable general indexing
  - Link your light client to a trusted full node (ideally your own)
  - Cancel sensible commands from the history
  - Store keys in cold wallets

Three potential threats

On the network



The Network

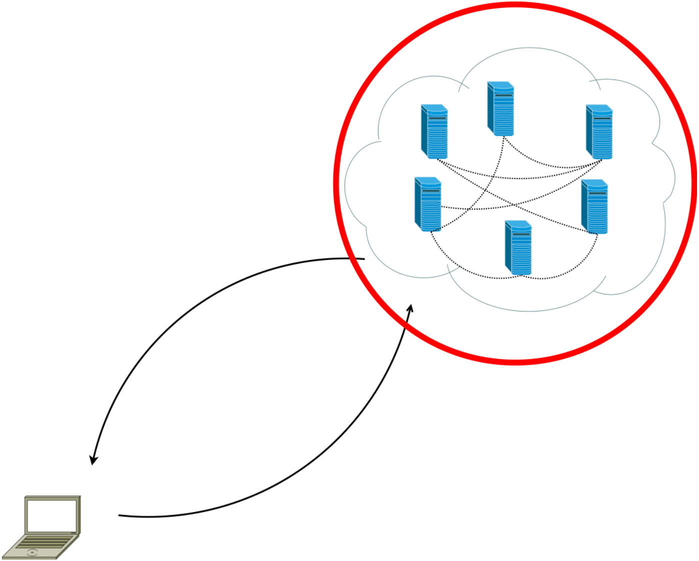
- Your IP known to everybody you communicate with
- Your ISP knows what you do
  - SSL/TLS encrypts the *content* of page, not the address
  - Open standard ports are a risk? We do not know
- Man-in-the-middle
- You post your address

The Network - defenses

- Obfuscation (proxy, VPN, TOR, I2P,...)
  - Check what's legal in your jurisdiction (e.g. TOR relays or exit nodes)
- No public Wi-Fi
- Be a miner/validator
- Do not post your address
  - Or use code that changes it in the web page

Three potential threats

On the blockchain



Blockchain inner privacy threats

All the transactions are in plain text, publicly available to everybody.

- Transaction clustering: attempt to de-anonymize blockchain users via discovering all addresses generated by a single user
    - Heuristic: in a transaction, all inputs in a transaction belong to the same person
    - Address reuse
- Following transaction: from one address (the one you gave to receive a payment)

Blockchain inner privacy threats

> *The combination of internal and external threats can lead to the discovery of the real identities behind both transactions and addresses*

Examples:

- Correlation between cookie metadata (e.g. time) and transaction
- Transaction metadata (e.g. IP address) and connection metadata
- Transaction metadata (e.g. time) leads to geographic area: cultural choices, political preference
- Use of centralized services (e.g. stores, exchanges) + Following transactions

Privacy defenses

Goal: Hide or Obfuscate data on blockchain
Six methods proposed to solve the problems, But each one has its own relative problems. The methods are:

- Layer 1 solutions
  - Zero-knowledge proofs (ZKP)
  - Digital signatures
  - Confidential transactions (CT)
- Layer 2 solutions (solutions that store only *some* operations on blockchain)
  - Statechannels (offchain transactions)
  - MultiParty Computation (MPC)
  - External services

ZKP

In standard (interactive) process

- Two parties, Prover and Verifier
- Prover presents (correct) answers to challenges
- Verifier issues challenges and verifies if Prover answers are correct

ZKP

In non-interactive processes (as in blockchain)

- Prover randomize challenges and proves them
- Prover present a single proof
- Verifiers check that the proof is correct

Uses in Blockchain:

- "I know the key and the value of a specific coin"
- "I can prove that this value is greater than 0"

Problems

- Elevated computational costs
- High blockchain overload
- Trusted setup (but it is highly researched)

Digital signatures

Many differnet digital signatures schemes in blokchain:

- ECDSA: used in Bitcon and Ethereum
- EdDSA: used in Tezos
- Ring signatures: used in Cryptonote and drivatives (Monero, Dash,...)

Proposals:

- Schnorr: proposed in Bitcoin
- BLS: proposed in Ethereum

It is not the signature, but how you use it

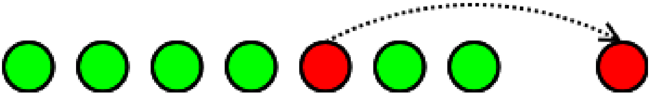
Ring Signatures

Particular instance of Group signature: many people with many keys produce a single signature of a document/message. How ring signature works:

- *One* person collects many public keys
- Multiple public keys and one private (signing) key create a signature of a message
- The verifier see multiple key but can not discern the private one from the public ones

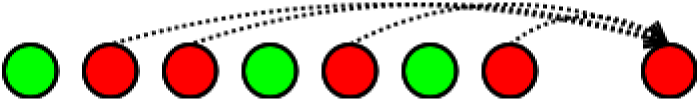
Cryptonote

- Bitcoin derivation: every coin has a key pair, when you spend a coin you expose your public key



Cryptonote

- Bitcoin derivation: every coin has a key pair, when you spend a coin you expose your public key
- You apply ring signatures to Monero: when you spend your coins you sign the transaction with your private key plus many public keys of other "chaff" coins
- Verifiers know you spent a valid coin that you own, but they do not know which one
- The procedure and results are called *mixins*



Problems

- Mixins are automatic: a user does not need to buld them manually
  - If the algorithm is not good there are leaks (see RNGs)
- Old implementation: choose coins from equal value (not true anymore: Monero uses Ring Confidential transactions)
  - If there are few coins with that denomination, there is a small anonymity set

Aggregate signatures

- Let more people sign a message
  - Different from multisignatures
- Privacy: you can hide how many parties are involved in a transaction

Problems

- Not possible today: need new signature schemes

Confidential Transactions (CT)

- Transactions that do not reveal the amount (amount is encrypted)
  - Ring Signatures: obfuscate the sender; CTs: hide the amount
- Prove that the input amount is equal to the output amount
  - No new money creation
  - The user has enough funds to do the transaction
- Methods:
  - Proof via homomorphic encryption

$$Enc(val_1 - val_2) = Enc(val_1) - Enc(val_2)$$

  - Proof via zero-knowledge proofs (range proofs)

Problems

- Bigger size of transactions (exacerbate the scalability problem)
- Difficult to implement the encryption method on current signature schemes

Statechannels

- Most studied and developend
  - Examples: Lightning Network, Raiden Network
- Exchange of funds off-chain

Problems

- Security problems
- Small amount exchanges
- Privacy problems

Multiparty Computation

Transactions to functions of a smart contract trigger executions. These executions could be performed off chain

- Distribute computation between multiple parties
  - E.g.: The Millionare Problem: find the richest person in a set without revealing the net worth
- Using external/off-chain machines (Trusted Execution Environments)

Problems

- Need to trust the external machines
- Difficult to securely implement the computation distribution

External services

- Mixers: mix coins
- Coinjoins: aggregate many inputs and outputs in one transaction

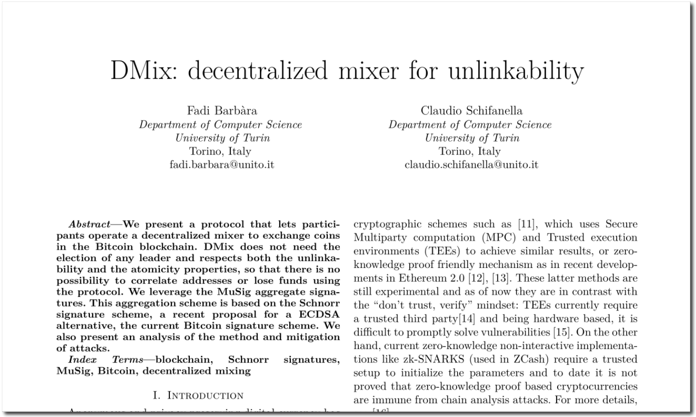
Problems

- Centralized mixers: what if the service steals funds?
- Coinjoins: tainted coins

Our Work

DMix: decentralized mixer for unlinkability

- Decentralized mixer via signature aggregation
- Problem: needs the new signatures schemes (Schnorr)



DMix: decentralized mixer for unlinkability

Fadi Barbàra
Department of Computer Science
University of Turin
Torino, Italy
fadi.barbara@unito.it

Claudio Schifanella
Department of Computer Science
University of Turin
Torino, Italy
claudio.schifanella@unito.it

***Abstract*—We present a protocol that lets participants operate a decentralized mixer to exchange coins in the Bitcoin blockchain. DMix does not need the election of any leader and respects both the unlinkability and the atomicity properties, so that there is no possibility to correlate addresses or lose funds using the protocol. We leverage the MuSig aggregate signatures. This aggregation scheme is based on the Schnorr signature scheme, a recent proposal for a ECDSA alternative, the current Bitcoin signature scheme. We also present an analysis of the method and mitigation of attacks.**

***Index Terms*—blockchain, Schnorr signatures, MuSig, Bitcoin, decentralized mixing**

I. Introduction

cryptographic schemes such as [11], which uses Secure Multiparty computation (MPC) and Trusted execution environments (TEEs) to achieve similar results, or zero-knowledge proof friendly mechanism as in recent developments in Ethereum 2.0 [12], [13]. These latter methods are still experimental and as of now they are in contrast with the “don’t trust, verify” mindset: TEEs currently require a trusted third party[14] and being hardware based, it is difficult to promptly solve vulnerabilities [15]. On the other hand, current zero-knowledge non-interactive implementations like zk-SNARKS (used in ZCash) require a trusted setup to initialize the parameters and to date it is not proved that zero-knowledge proof based cryptocurrencies are immune from chain analysis attacks. For more details,

Overview of the talk

- First part
  - Motivation and terminology
  - Assumptions
  - Overview of External blockchain privacy threats
- Second part
  - Internal blockchain privacy threats
  - Proposed method to solve the problems

The End

Thank you for your attention.
Contacts:

- me@fadibarbara.it
- fadi@di.unito.it