Bitcoin: decentralized electronic cash system
Part 2

Giacomo Scornavacca

December 20, 2016

Recap

In the first lesson:

- We have given an idea about what money is today and why electronic currencies have nearly the same properties of physical currencies.
- We have "implemented" an electronic currency using a Central Authority
- Without a Central Authority we have introduced the Double-Spending problem and formulated it as a Consensus problem.
- We have analysed the Satoshi Nakamoto's paper and we have understood how Bitcoin **should** solve the Consensus problem using the Proof-of-Work idea.

Analysis of the Bitcoin protocol

The actual Bitcoin protocol is different from the simplification made by Satoshi Nakamoto and is not well documented. The code is the only official documentation.

In these slides we will see where the bitcoin protocol is different from the "theory" and which vulnerabilities are been already discovered.

The Bitcoin Protocol:

1. **New transactions are broadcast to all nodes.**
2. Each node collects new transactions into a block.
3. Each node works on finding a difficult proof-of-work for its block.
4. When a node finds a proof-of-work, it broadcasts the block to all nodes.
5. Nodes accept the block only if all transactions in it are valid and not already spent.
6. Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.

Privacy in the Bitcoin System

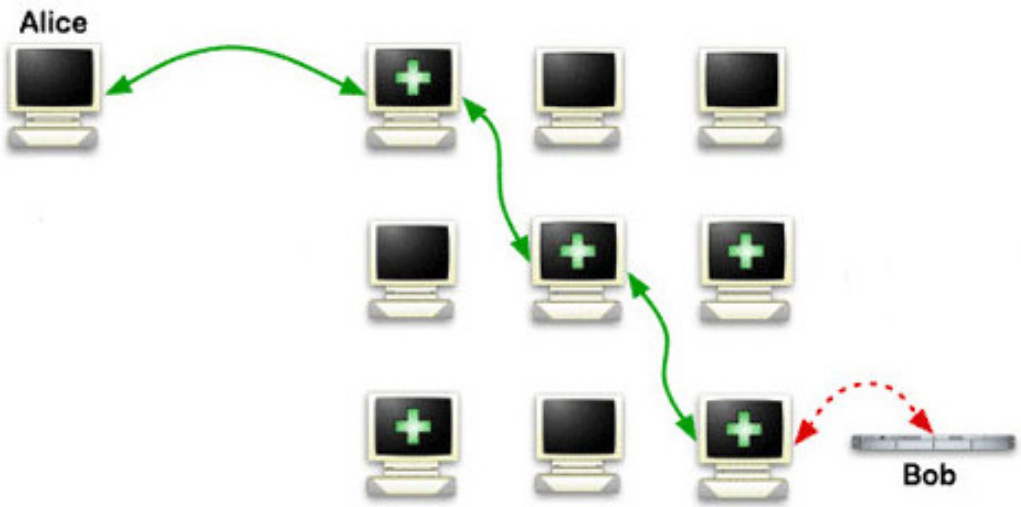
When a new transaction is created by a node, a broadcast operation starts. In a asynchronous network the transaction can arrive to the different users in different ways. But **probably** the neighbours of the node will receive always the transaction from the sender. In general in a peer-to-peer network is possible to link the source of a message with the message itself (a public key to an ip address).

THE SYSTEM IS NOT PSEUDO-ANONYMOUS

A possible solution is the use of a Proxy, but is a centralized solution in the sense we are only moving the anonymity problem on the Proxy.

Privacy in the Bitcoin System: The Onion Router

Tor is free software for enabling anonymous communication. Tor encrypts the data, including the destination IP address, multiple times and sends it through a virtual circuit comprising successive, **randomly selected** Tor relays. Each relay decrypts a layer of encryption to reveal only the next relay in the circuit in order to pass the remaining encrypted data on to it. The final relay decrypts the innermost layer of encryption and sends the original data to its destination without revealing, or even knowing, the source IP address.



Privacy in the Bitcoin System: **T**he **O**nion **R**outer

Both Tor and Bitcoin networks are vulnerable to Sibyl attacks!

What if I'm a malicious user that controls a big fraction of the Tor relays in the Tor network AND a lot of nodes in the Bitcoin Network? With good probability i can know the source of a transaction.

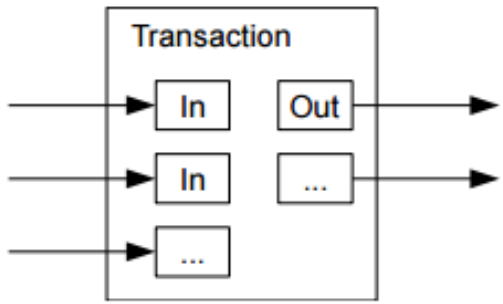
Privacy in the Bitcoin System: Transactions Tree

The block chain is public and it is possible trace the origin of every bitcoin: `https://blockchain.info/tree/114688189`

Let's suppose NSA knows that a certain Public Key (owned by Alice) is used by a drug dealer and this information becomes public. Probably people will not accept payments originated from this Public Key.
What Alice can do?

Privacy in the Bitcoin System: Laundry Services

Recall a Transaction can have multiple inputs and multiple outputs:



Laundry Service: An user is used as intermediate layer to make impossible trace the origin of the Bitcoins in output.

Laundry Services are a Centralized Solution

Privacy in the Bitcoin System: Zerocoin

Zerocoin is an extension to the Bitcoin protocol that would add **true** cryptographic anonymity to bitcoin transactions.

VERY informally:

- Users can use this extension to mint Zerocoin from Bitcoin (e.g. I use 10 Bitcoin to "buy" 10 Zerocoin and this information is anonymously published on the Block Chain).
- Users that have Zerocoins can spend them using a **zero-knowledge proof**. (e.g. I can send Bitcoins to some user using the Zerocoin proving I was the owner of 10 Bitcoins but not saying which Bitcoins).

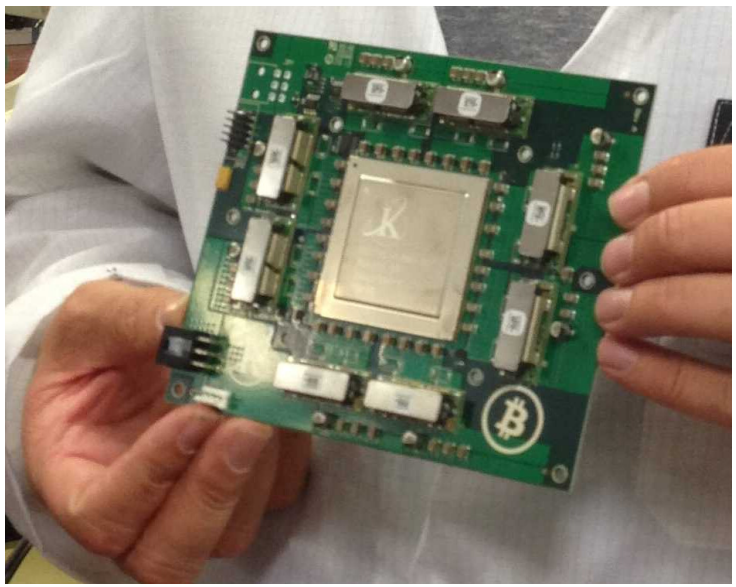
Zerocash is another cryptocurrecy based on the proof-of-work idea and on this zero-knowledge protocol.

The Bitcoin Protocol:

1. New transactions are broadcast to all nodes.
2. **Each node collects new transactions into a block.**
3. **Each node works on finding a difficult proof-of-work for its block.**
4. When a node finds a proof-of-work, it broadcasts the block to all nodes.
5. Nodes accept the block only if all transactions in it are valid and not already spent.
6. Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.

One-CPU-One-Vote is no longer true

Proof-of-work means repeatedly perform hash (SHA 256) operations. **Dedicated hardware** is so quick (and energy efficient) that mining with a CPU or a GPU is not worth.

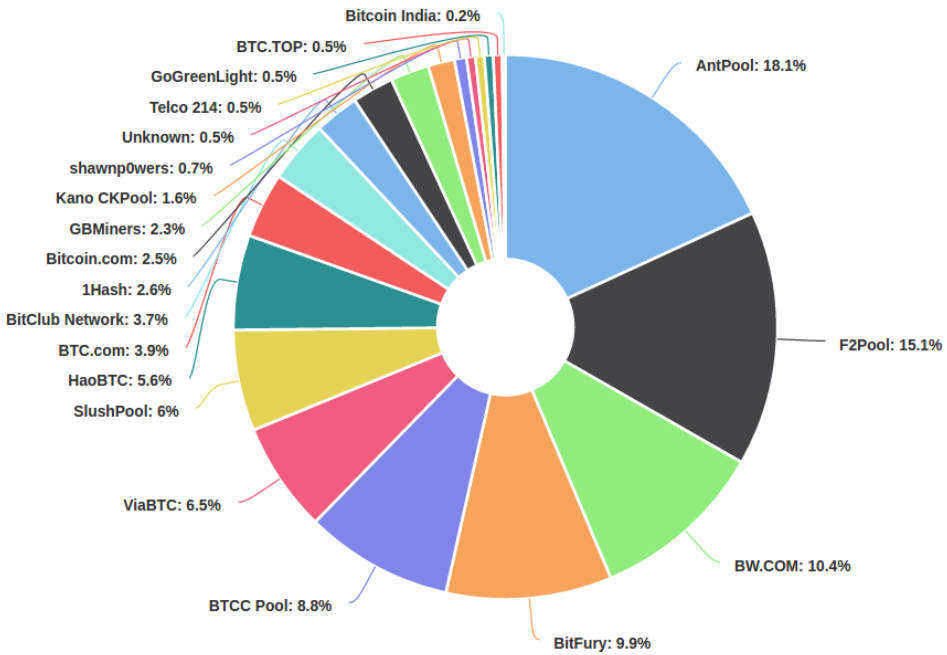


The nodes provided with this special hardware are called **miners**.

The network is not any more homogeneous.

Incentive & Bitcoin Hashrate Distribution 5, jun 2015

Multiple miners that are in competition, working on a proof-of-work on the same block can try the same NONCE multiple times. **If they "collude" they can partition the search space** in order to **increase the expected** rewards (by only some $\epsilon > 0$). Moreover, working together and dividing the rewards, the miners **decrease the variance** of the rewards.



Four mining pools own 50% of the hash power

The Bitcoin Protocol:

1. New transactions are broadcast to all nodes.
2. Each node collects new transactions into a block.
3. Each node works on finding a difficult proof-of-work for its block.
4. **When a node finds a proof-of-work, it broadcasts the block to all nodes.**
5. Nodes accept the block only if all transactions in it are valid and not already spent.
6. Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.

Majority is not Enough: Bitcoin Mining is Vulnerable
Ittay Eyal and Emin Gun Sirer, 2013

The authors show a strategy, called **Selfish-Mine**, that allows a pool of sufficient size to obtain a revenue larger than its ratio of mining power [...] When the selfish miner pool finds a block,[...] instead of naively publishing this private block and notifying the rest of the miners of the newly discovered block, **selfish miners keep this block private to the pool.** The selfish miners start working on a new block while the rest of the honest miners work on the old block.

Now two scenarios can arise.

Majority is not Enough: Bitcoin Mining is Vulnerable

First scenario: the honest nodes succeed in finding a block on the public branch, nullifying the selfish pools lead, the pool immediately publishes its private branch (of length 1). This yields a toss-up where either branch may win. The selfish miners unanimously adopt and extend the previously private branch, while the honest miners will choose to mine on either branch, **depending on the propagation of the notifications**.

The scenario can end with two, one or zero blocks mined by the selfish miners.

Majority is not Enough: Bitcoin Mining is Vulnerable

Second scenario: the selfish pool succeeds in finding a second block. Once the pool reaches this point, it continues to mine at the head of its private branch. It publishes one block from its private branch for every block the others find. Since the selfish pool is a minority, its lead will eventually reduce to a single block with high probability. At this point, the honest miners are too close, so the pool publishes its private branch. Since the private branch is longer than the public branch by one block, it is adopted by all miners as the main branch, and the pool enjoys the revenue of all its blocks.

Majority is not Enough: Bitcoin Mining is Vulnerable

The authors show that selfish miners can obtain a revenue larger than its ratio of mining power, and moreover, using this strategy the 33% of the hash power can be sufficient to control almost all the blocks in the block chain (the exact value depends by a information spreading parameter).

The Bitcoin Backbone Protocol Analysis and Applications Garay et al. 2016

Under the following (strong) assumptions:

- The number of nodes are fixed
- A synchronous communication network
- The presence of a $\frac{1}{3}$ computational bounded adversary

it is possible prove the **Common Prefix** and **Chain Quality** properties.

Common Prefix means that the block chain maintained by the honest player will posses a large common prefix. It is used to prove **Persistence** of transactions. **Chain Quality** means that a good ratio of blocks in the block chain is mined by honest miners. It is used to prove **Liveness**.

How much it cost to have the 50% of the hash power?

Total hash rate: 2.500.000 TH/s
`https://blockchain.info/charts/hash-rate`

Dedicaded hardware: 1 TH/s $\approx$ 100$
`https://www.hobbymining.com/mining-hardware/`

Total price $\leq$ 250 millions of dollar
If you buy your own factory of dedicated hardware this price will drop.

Market Capitalization: 12 BILLIONS of dollar
`https://blockchain.info/charts/market-cap`

Conclusions

- The anonymity of the system relies on the anonymity of Tor. Moreover, without the use of some extension (Zerocoin), sometimes it is possible trace the origin of Bitcoins.
- The network is not homogeneous and today the three most important mining pools can collude in order to perform Double Spending Attacks.
- Some vulnerabilities that use the delays of informations are been discovered. What we will find out more?
- The only formal analysis of the Bitcoin protocol uses strong hypothesis.
- The market value of Bitcoins is 50-500 times the dollar needed to have the half of the hashing power.