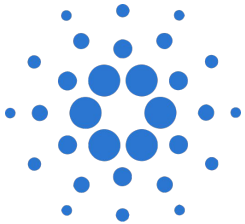
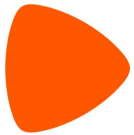


Blockchain 101

Maksim Ekimovskii

About me

Software Engineer at



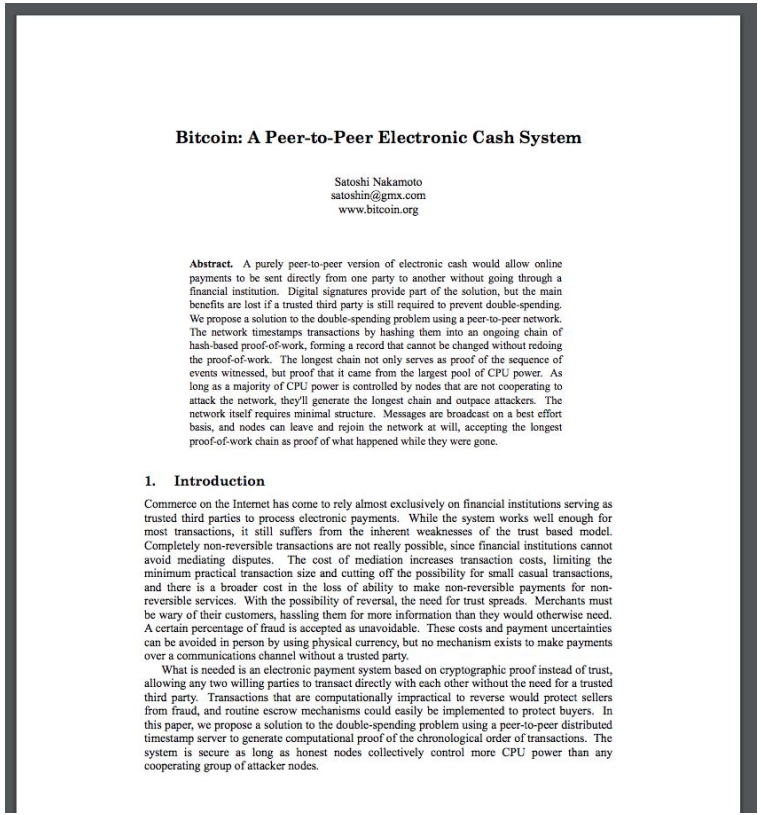
@prawn_cake

prawn-cake

Agenda

- Why?
- What?
- How?

Why?



Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

1. Introduction

Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model. Completely non-reversible transactions are not really possible, since financial institutions cannot avoid mediating disputes. The cost of mediation increases transaction costs, limiting the minimum practical transaction size and cutting off the possibility for small casual transactions, and there is a broader cost in the loss of ability to make non-reversible payments for non-reversible services. With the possibility of reversal, the need for trust spreads. Merchants must be wary of their customers, hassling them for more information than they would otherwise need. A certain percentage of fraud is accepted as unavoidable. These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

What is needed is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party. Transactions that are computationally impractical to reverse would protect sellers from fraud, and routine escrow mechanisms could easily be implemented to protect buyers. In this paper, we propose a solution to the double-spending problem using a peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions. The system is secure as long as honest nodes collectively control more CPU power than any cooperating group of attacker nodes.

https://bitcoin.org/bitcoin.pdf

Why?

To have a digital currency with **no central** “trusted” 3rd party (i.e bank)

(bitcoin)

Why?

To build a **trustworthy** decentralised network with the **only one rule**

Why?

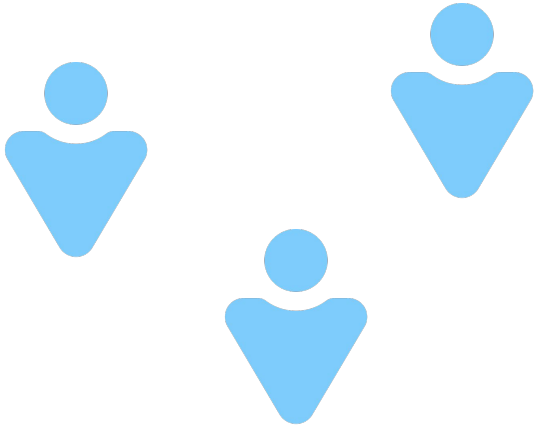
Trustworthy network where

Everyone trusts nobody

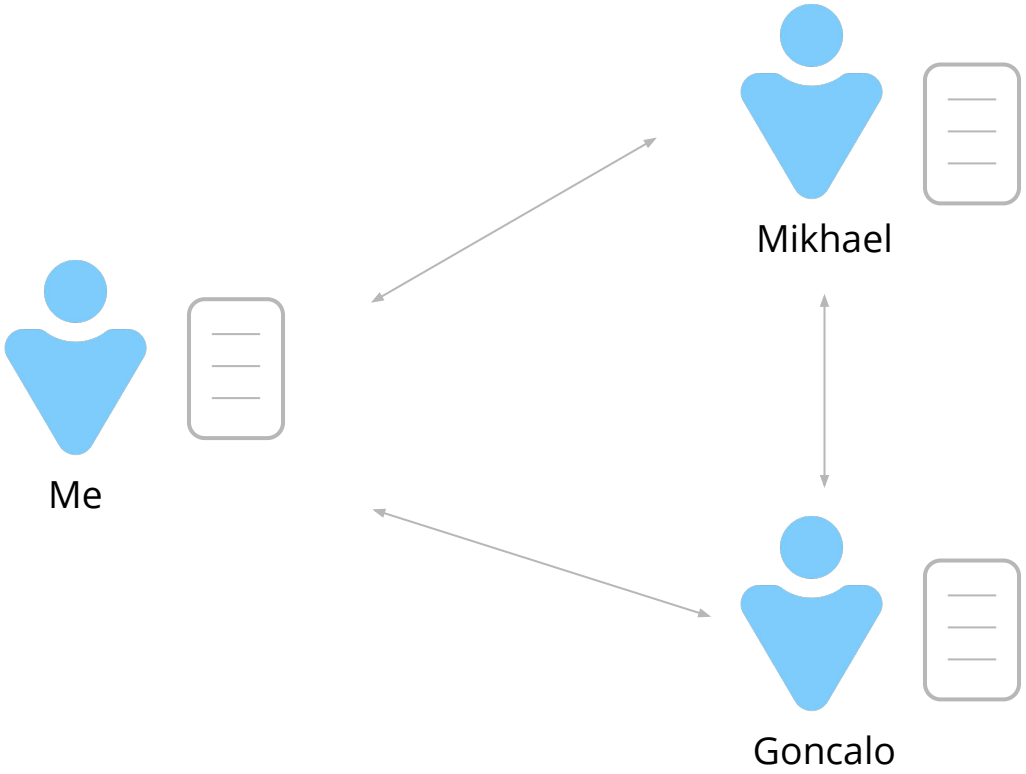
What?

Linked list of facts, e.g

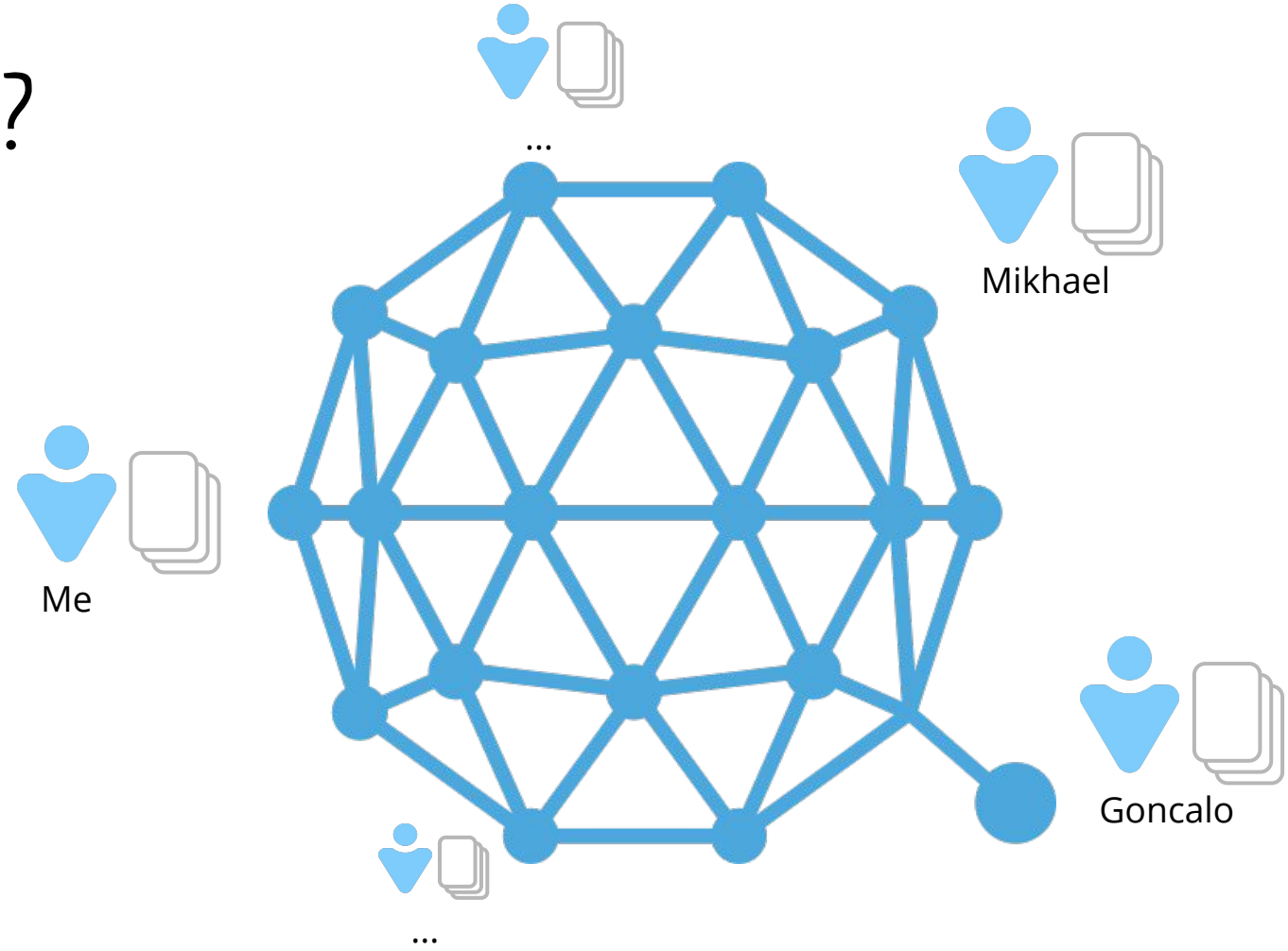
- ➔ Paid one beer (7€) for Goncalo
- ➔ Borrowed 10€ to Mikhael
- ➔ Got 7€ from Goncalo
- ➔ Mikhael sent 10€ to me



What?



What?



What if someone...

- Tampers the data
- Fakes the identity
- Sends transaction twice

How to...

- Order transaction
- Order simultaneously created blocks

How?

Demo

How does a network operate?

New transaction's ***broadcast***

Nodes **add** it to the ***pool*** *of unconfirmed transactions*

Nodes **pick** transactions from the pool and **mine** the hash to complete the block

Node which found a hash, ***broadcasts*** the block

All nodes ***verify*** validity of the transactions in the block

When accepted nodes create a **new block**, it **uses** the **hash** of the *accepted block* as the *previous hash*

What is a transaction?

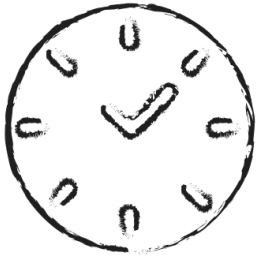
A serializable object with a set of fields

e.g bitcoin transaction looks like

General format of a Bitcoin transaction (inside a block)

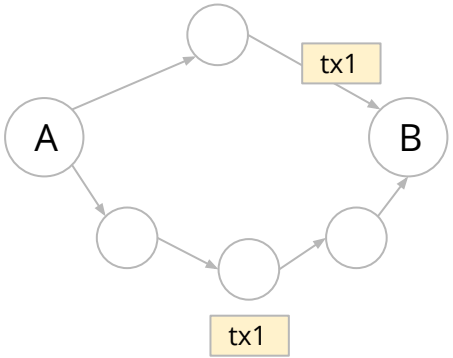
Field	Description	Size
Version no	currently 1	4 bytes
Flag	If present, always 0001, and indicates the presence of witness data	optional 2 byte array
In-counter	positive integer VI = VarInt	1 - 9 bytes
list of inputs	the first input of the first transaction is also called "coinbase" (its content was ignored in earlier versions)	<in-counter>-many inputs
Out-counter	positive integer VI = VarInt	1 - 9 bytes
list of outputs	the outputs of the first transaction spend the mined bitcoins for the block	<out-counter>-many outputs
Witnesses	A list of witnesses, 1 for each input, omitted if flag above is missing	variable, see Segregated_Witness
lock_time	if non-zero and sequence numbers are < 0xFFFFFFFF: block height or timestamp when transaction is final	4 bytes

Transactions order and double-spending. How?



10 min...10 sec...

There is no order



The answer is “***block***chain”

Transaction identity

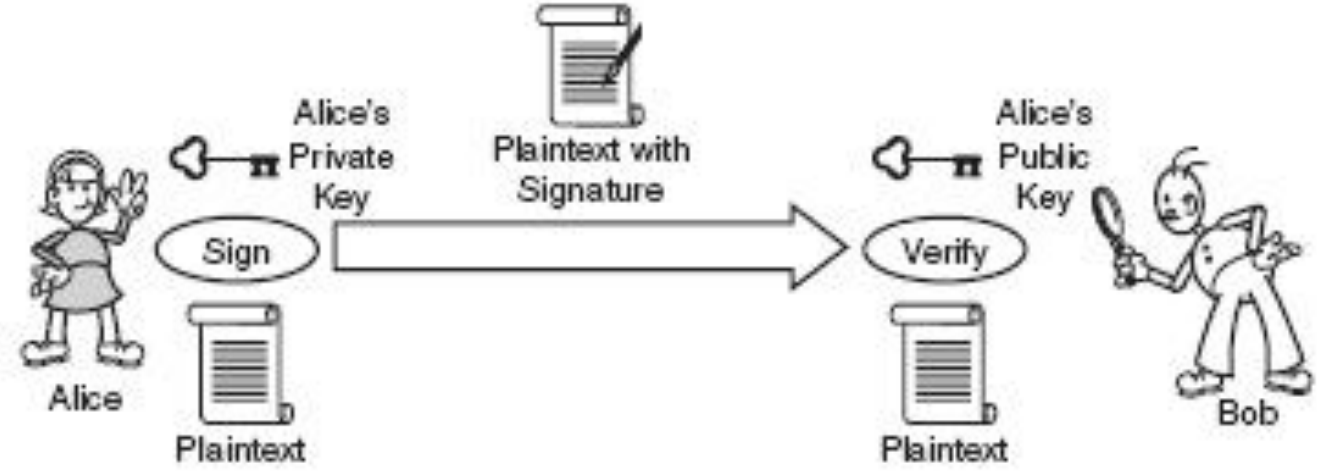
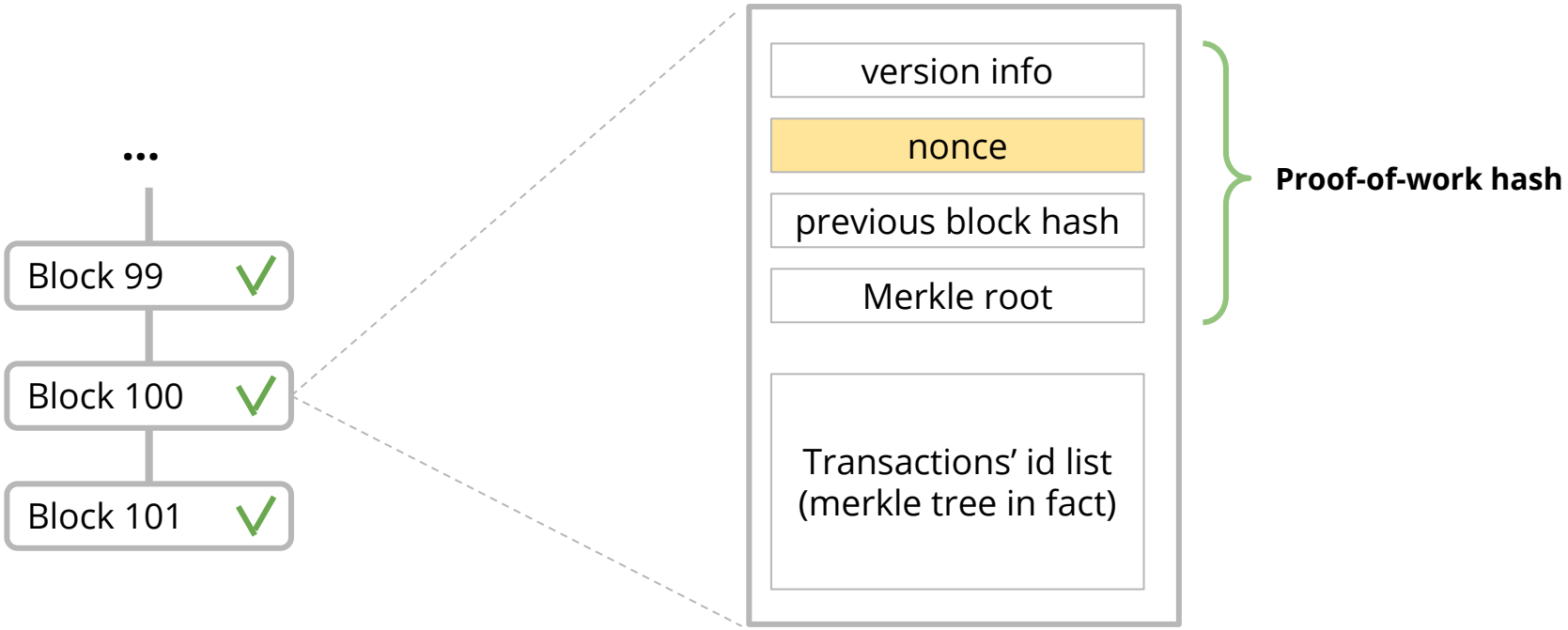


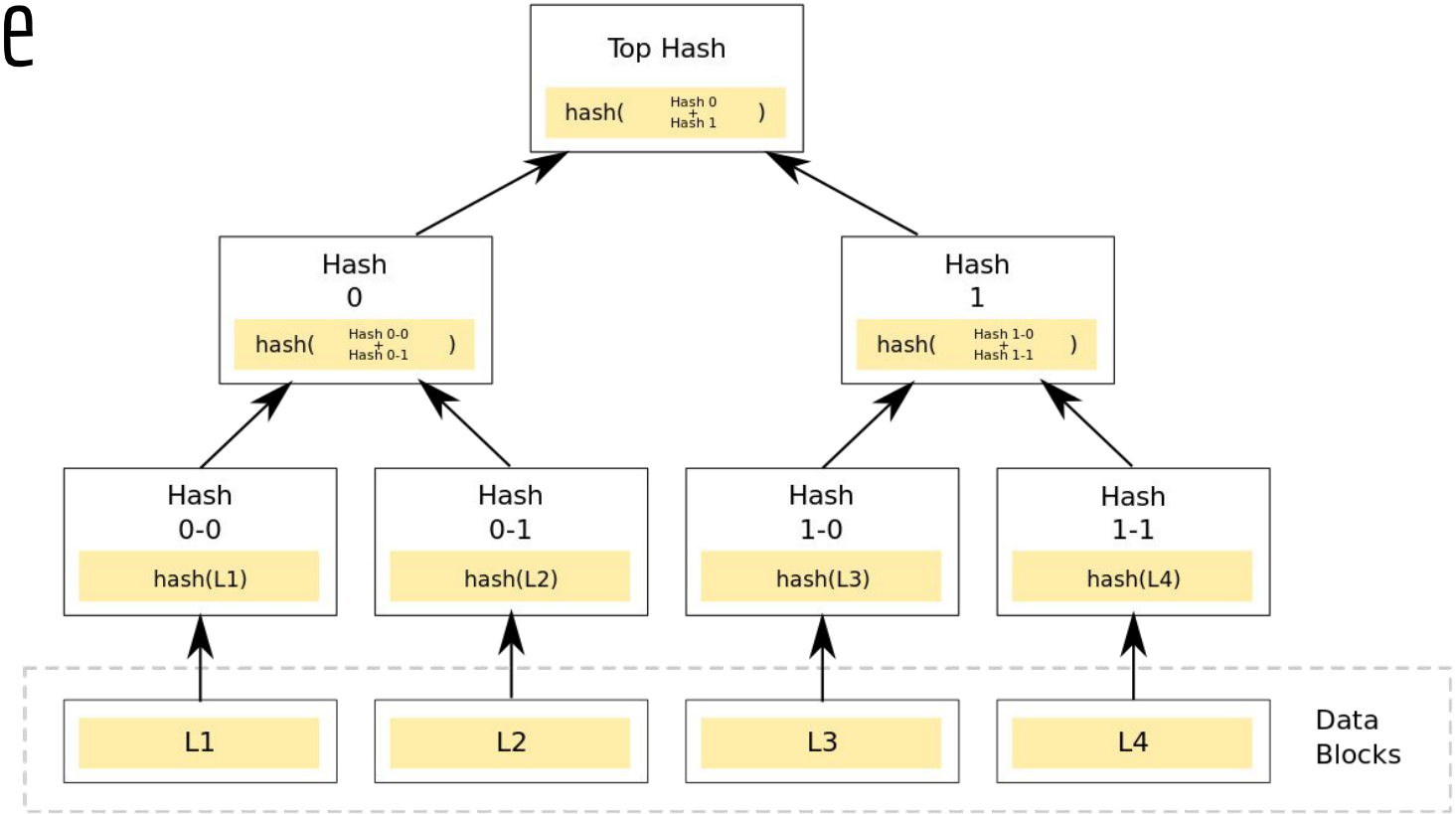
Figure 9.4 Digital signature

https://searchsap.techtarget.com/feature/Chapter-9-Securing-Web-Services

What is block?



Merkle tree


Top Hash
hash(Hash 0 + Hash 1)
Hash 0
hash(Hash 0-0 + Hash 0-1)
Hash 1
hash(Hash 1-0 + Hash 1-1)
Hash 0-0
hash(L1)
Hash 0-1
hash(L2)
Hash 1-0
hash(L3)
Hash 1-1
hash(L4)
L1
L2
L3
L4
Data Blocks

What is mining?

When thousands of CPUs and GPUs try to find a hash starting with X zeros

Or it's *a process of finding proof-of-work*

Proof-of-work

- Idea was borrowed from the Hashcash
- Hard to find, easy to check

0000000000000756af69e2ffbdb930261873cd71

Also called block **difficulty**

What is mining?

A computer needs **tens of years** to to find a hash starting with 18 zeros

Mining pool of computers does it within 10 minutes

Check the live block

Who is a miner?

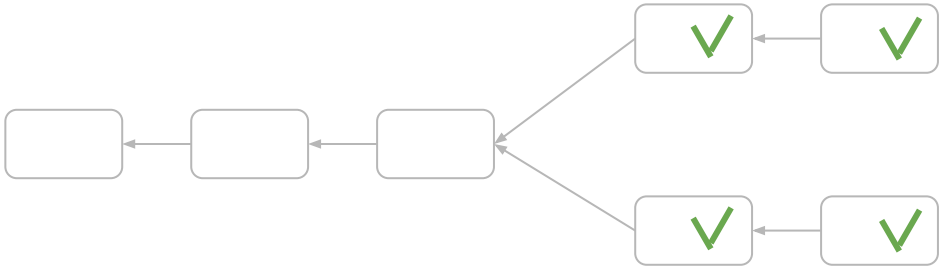
Miners are nodes which **is a crucial** part of the network

No miners - **who** is going to confirm all that crap?

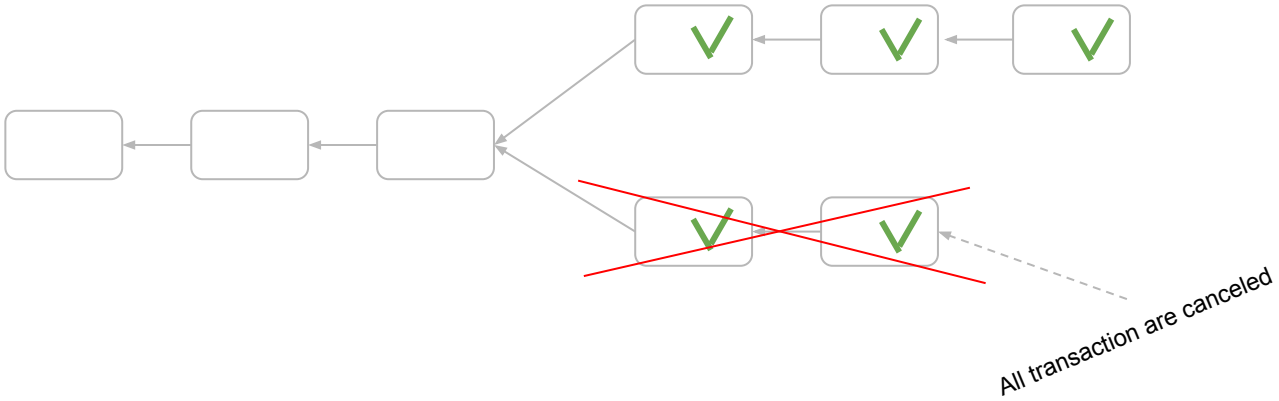
Why would anybody start mining?

- Incentive per block
- Incentive per transaction

Race conditions. Huh?

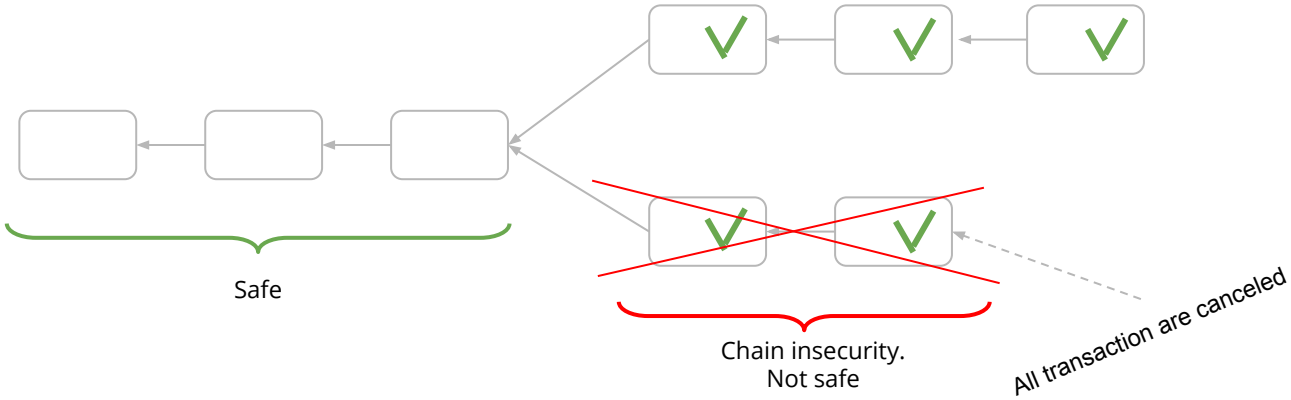


Race conditions. Huh?



Longest chain wins

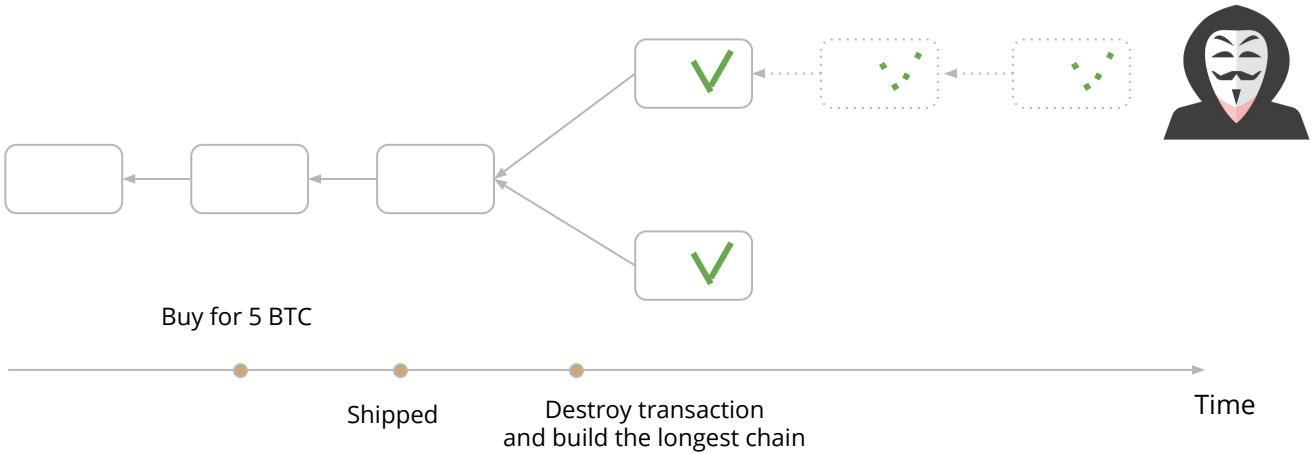
Race conditions. Huh?



Tail optimizations

- Incentive can be used only after 20 blocks passed **[client option]**
- If you received currency (bitcoin), you can use at only after 5 blocks passed **[client option]**

Let’s hack blockchain



Let’s hack blockchain

50% of compute power give 50% of probability to win the race

Limitations

- Scalability and throughput (bitcoin: 7tx / sec, ethereum: 2500 tx / sec per shard, visa 24 000 tx/sec)
- 51% attack
- Storage constraints (Ethereum blockchain grows 55 GB/year (full mode))

My initiatives

https://github.com/prawn-cake/lecs/issues

Questions?