Blockchains and Bitcoins – how and why it works

Blockchain Workshop, Dallund Slot, May 2018

Martin Elsman
Department of Computer Science
University of Copenhagen (DIKU)

The Speaker

Martin Elsman, Associate Professor at DIKU

Research activities:

- Certified management of financial contracts.
- Programming language design and implementation (functional languages)
- Parallel programming languages - getting programs, such as simulations, to run efficiently on GPUs.

Other activities:

- Manager, HIPERFIT Research Center, DIKU (2012-2018)
- CTO and partner in iAlpha - a London-, DK-, and Swiss-based startup specialising in financial analytics.

Here is a House

How do we know who owns this house?

Because someone has a key?

No! *It's easy to fake a key...*

Because someone has a title deed (skøde) that demonstrates transfer?

No! *It's even easier to fake a letter...*

Here is a House

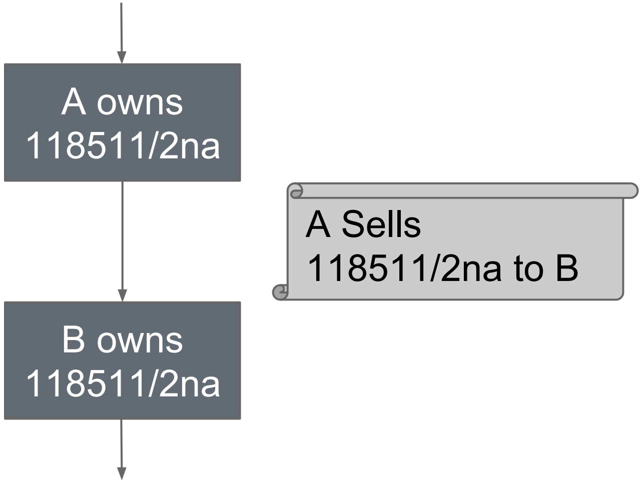
<u>How do we know who owns this house?</u>

Because the title deed is published in the Land Registry (tingbog), which

1. Records who owns what, publicly!
2. Certifies transfer of ownership (A -> B), but only if A was owner according to the Land Registry itself.

Here is a House

Chain of ownership:



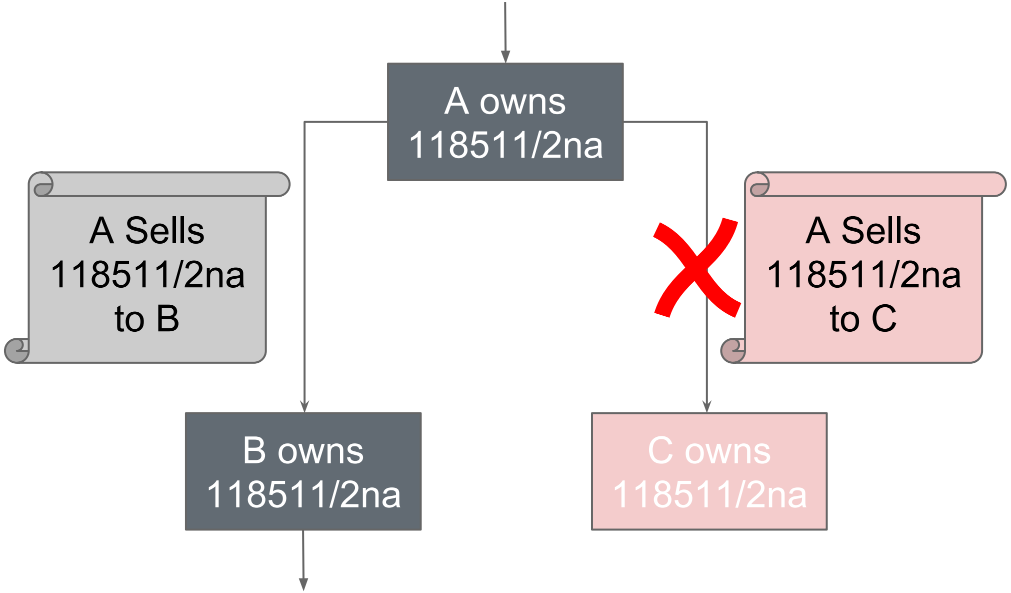
A owns 118511/2na

A Sells 118511/2na to B

B owns 118511/2na

Here is a House

The Land Registry is a *linear chain*, which ensures **NO** double spending.



A owns 118511/2na

A Sells 118511/2na to B

A Sells 118511/2na to C

B owns 118511/2na

C owns 118511/2na

The Land Registry *linear public timeline* is:

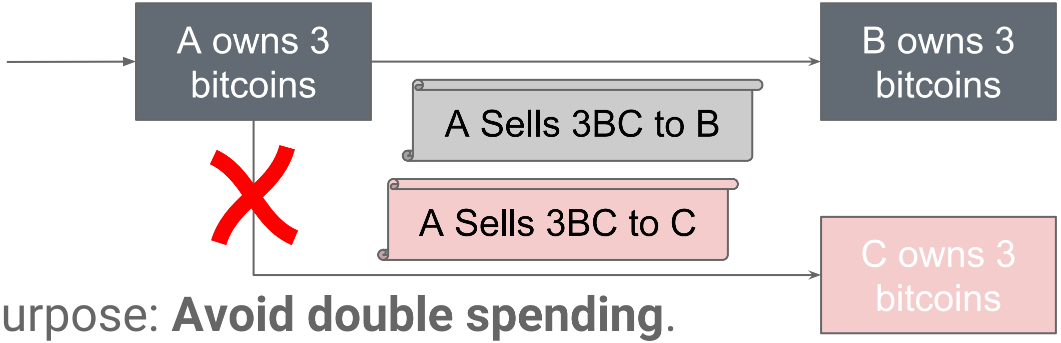
1. A ***centralized*** consensus mechanism
2. Run by a ***single trusted*** entity (the government)
3. Mandated by ***law*** (1551 in DK)

Bitcoin’s Blockchain

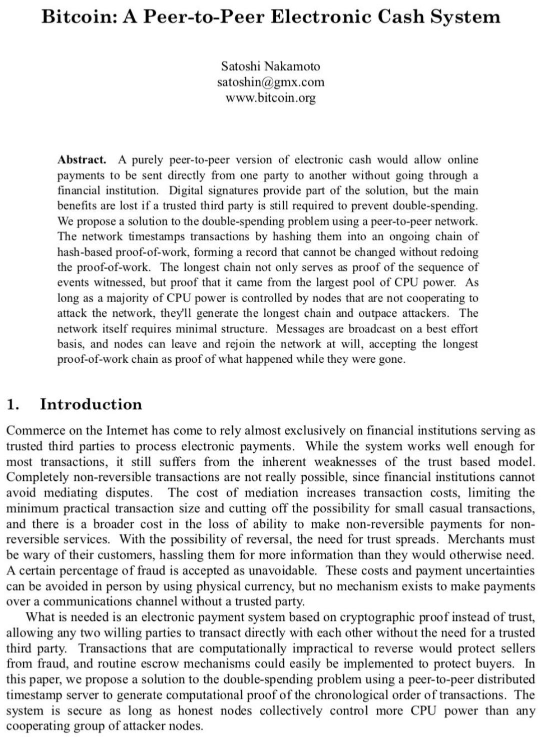
Bitcoin white paper (October 31th, 2008) by ***Satoshi Nakamoto*** (unknown author).

A ***decentralised*** *linear public timeline* of Bitcoin transfers.

Run by ***multiple untrusted entities.***



Purpose: **Avoid double spending**.



Bitcoin: A Peer-to-Peer Electronic Cash System

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

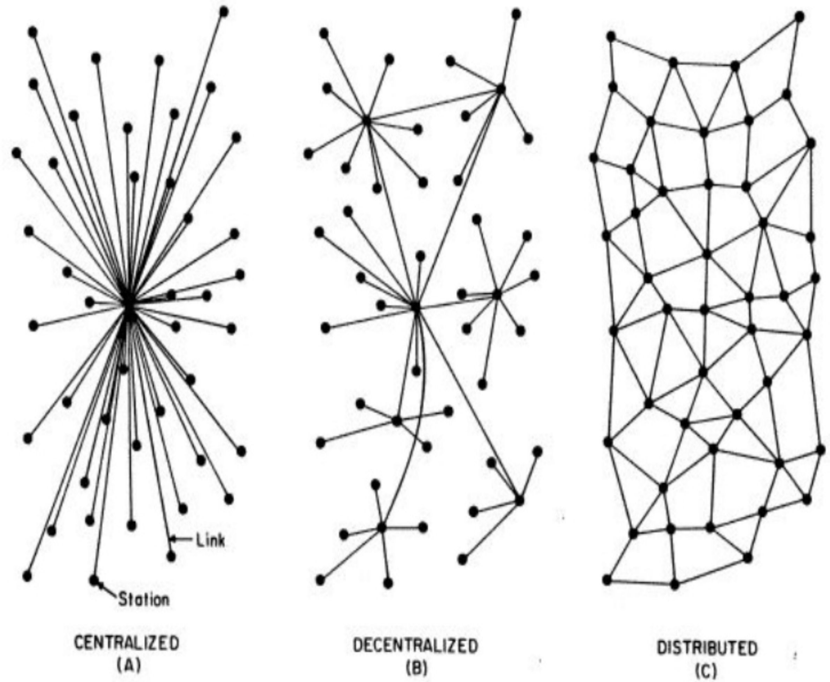
Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required 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.

Distributed Systems



CENTRALIZED (A)

DECENTRALIZED (B)

DISTRIBUTED (C)

Shared: Every node in the P2P network is client as well as server.

Trusted: Game theory is used to model Economic incentives for nodes (open protocol).

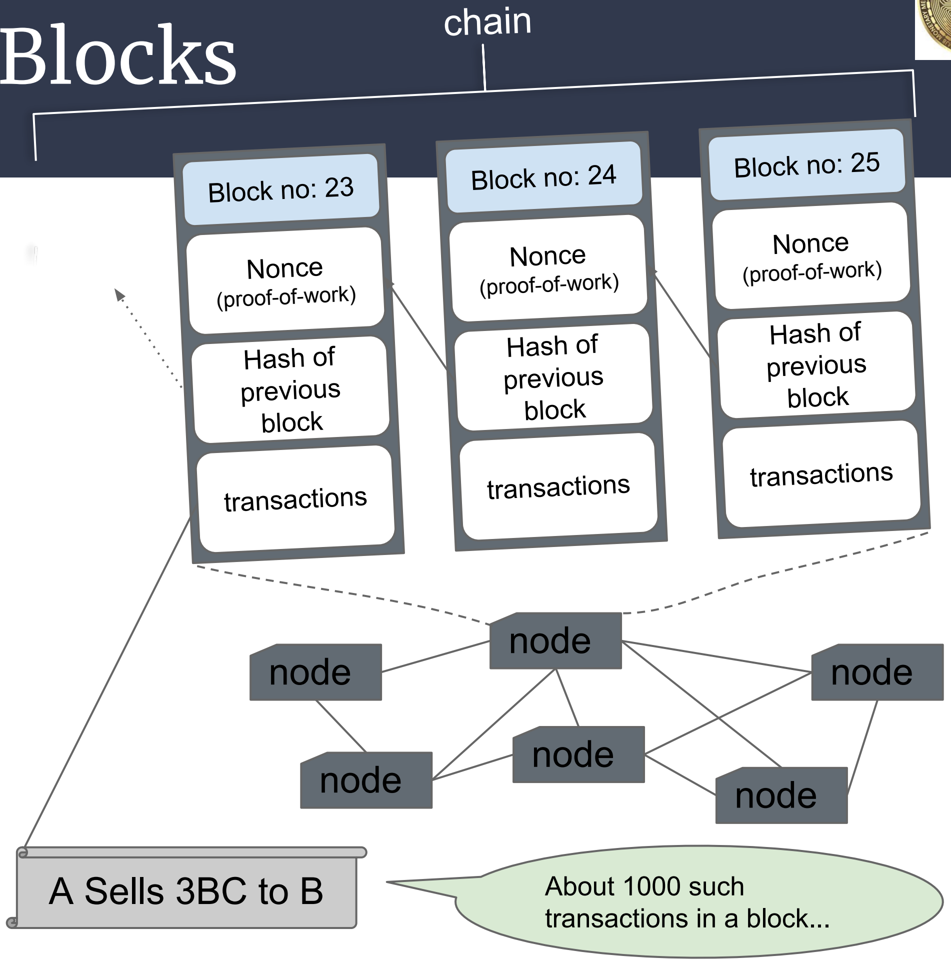
Public ledger transactions: Cryptography in the form of a distributed hash chain guarantees security and privacy.

The Bitcoin Chain of Blocks

Records ***mutual agreed-upon*** transactions.

A ***mutual consensus*** mechanism (proof-of-work) ensures that nodes agree on transactions.

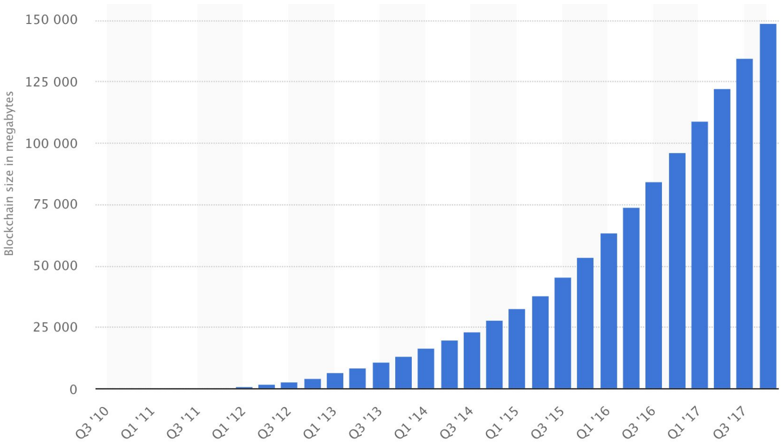
The mechanism used is ***cryptographic hash functions.***

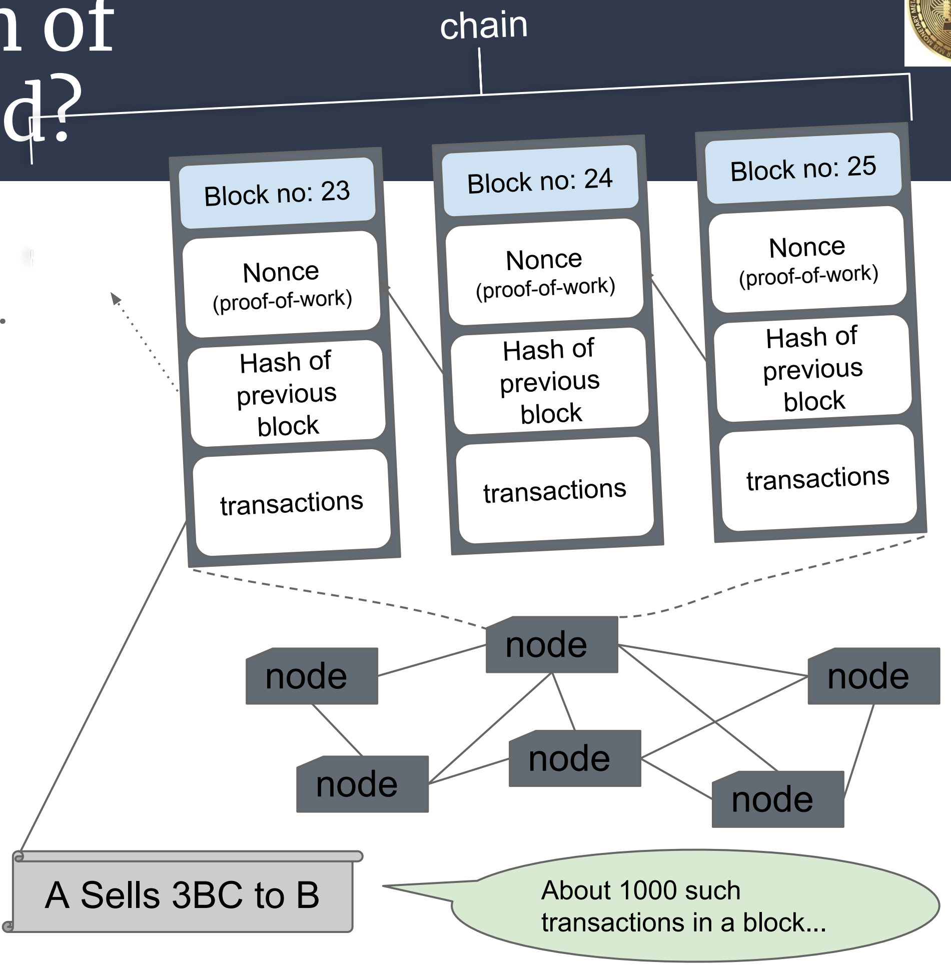


So Where is the Chain of Blocks Actually Stored?

Each (full) ***node*** in the distributed network has a ***copy*** of the blockchain.

New (full) nodes get the chain from their peers (150GB in 2017)

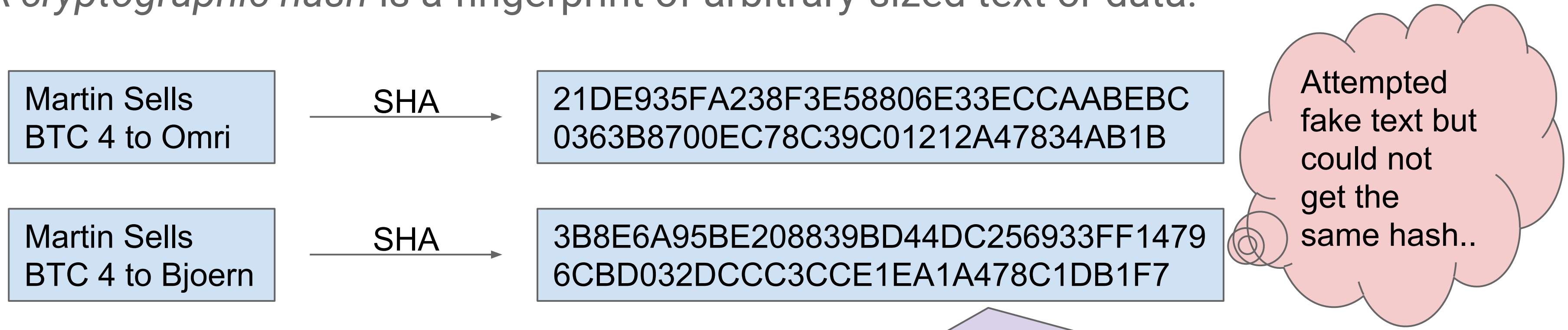




Cryptographic Hash Functions

(small side-step)

A *cryptographic hash* is a fingerprint of arbitrary-sized text or data:



Martin Sells BTC 4 to Omri —SHA→ 21DE935FA238F3E58806E33ECCAABEBC0363B8700EC78C39C01212A47834AB1B

Martin Sells BTC 4 to Bjoern —SHA→ 3B8E6A95BE208839BD44DC256933FF14796CBD032DCCC3CCE1EA1A478C1DB1F7

Attempted fake text but could not get the same hash..

64 characters; try on https://passwordsgenerator.net/sha256-hash-generator/

A ***hash*** is a 1-way function:

- It is easy to compute SHA(*x*) for any *x*.
- Given *h*=SHA(*x*), it is very difficult to find *x'*≠*x* such that *h*=SHA(*x'*).

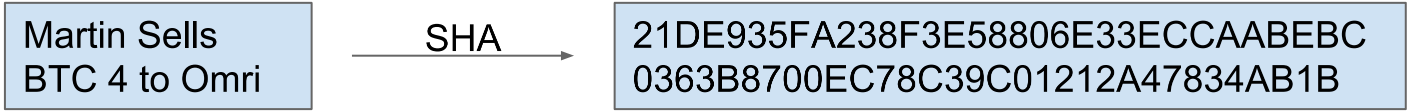
Blockchain purpose: make it ***very hard*** to rewrite history.

Hashes for Integrity Proofs

(small side-step)

Hashes can be used for certifying that two parties have **agreed on a contract**.

The two parties ***hash the contract*** and they **both tweet** the hash:


Martin Sells
BTC 4 to Omri

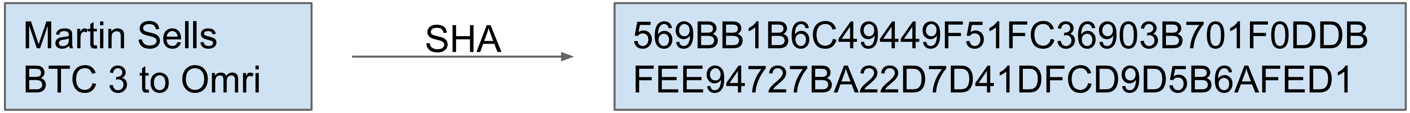
SHA

21DE935FA238F3E58806E33ECCAABEBC
0363B8700EC78C39C01212A47834AB1B

Better than written signatures - it is now easy to prove in court which is the agreed contract...

Privacy: No one else can infer the contract from the hash!

None of the parties can fake the contract - computationally, it is **very difficult** to get the same hash:


Martin Sells
BTC 3 to Omri

SHA

569BB1B6C49449F51FC36903B701F0DDB
FEE94727BA22D7D41DFCD9D5B6AFED1

Martin attempted to fake it but could not get the same hash..

The NIST SHA Function

(small side-step)

NIST: National Institute of Standards and Technology, USA.

SHA-2 is Standard.

SHA-3 chosen in 2012 for future use.

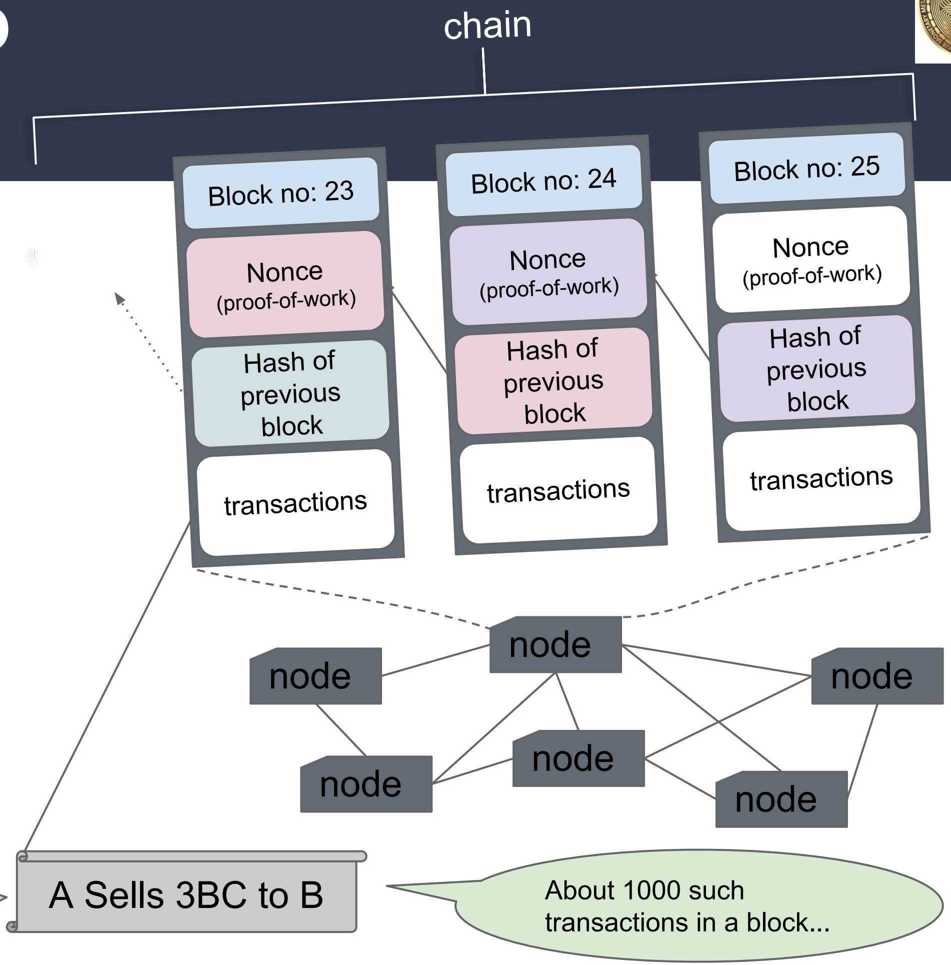
It is **secure because it is *open:***

- Open standard
- It is chosen in competition
- Analysed by industry, academia, and defense
- Many open source implementations

Hashes are Central to Blockchains

The ***nonce*** value stored in a block *N* is a value (text string) such that the hash of the following text items include *K* (e.g., 10) leading zeros:

1. The block no *N*
2. The ***nonce*** value
3. The previous hash
4. The transactions (transfers of bitcoins)

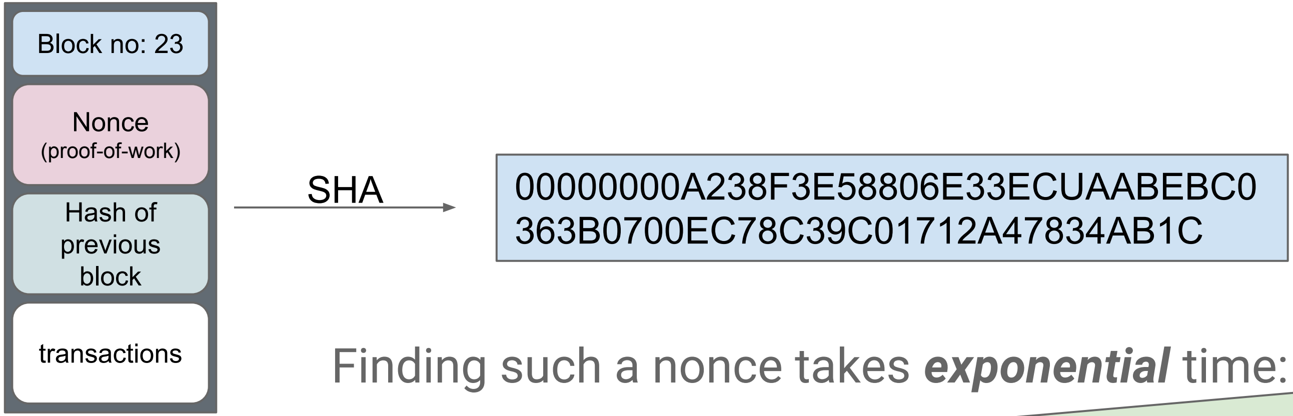


Hashes are also used to identify parties

About 1000 such transactions in a block...

Basic Crypto Puzzle

Puzzle: Given a candidate block (block without a nonce), find a nonce that make the SHA of the block contain *K* (e.g., 8) leading zeros:



Finding such a nonce takes ***exponential*** time:

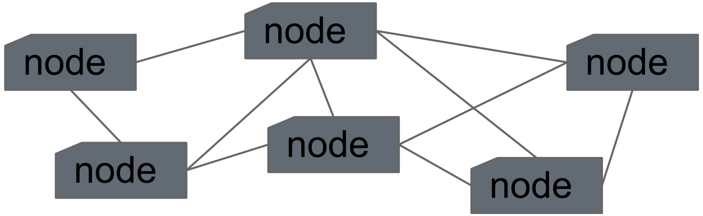
Why 35?

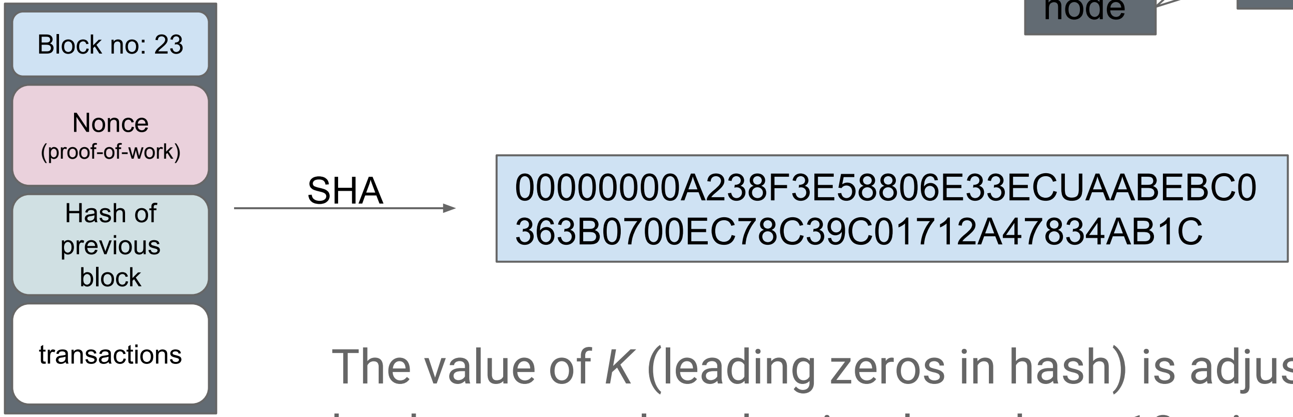
- 35^K attempts (35^8 = 2.251.875.390.625)
- There exists no other way than to try with new nonces!

Mining

Mining

The incentive: the ***fastest*** compute node wins the mining race and is awarded with bitcoins!




Block no: 23

Nonce (proof-of-work)

Hash of previous block

transactions

SHA

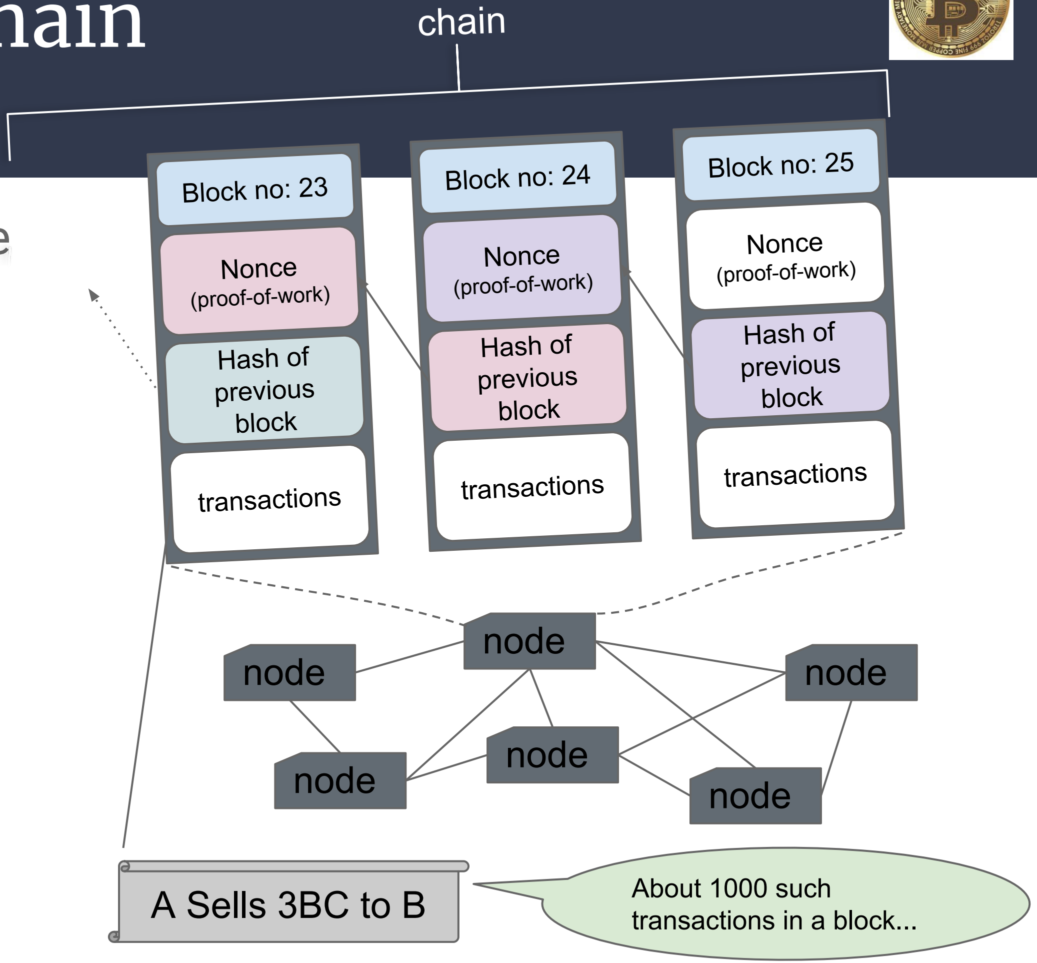
00000000A238F3E58806E33ECUAABEBC0
363B0700EC78C39C01712A47834AB1C

The value of *K* (leading zeros in hash) is adjusted regularly by the protocol so that it takes about 10 minutes to mine a block (find the nonce).

The Strength of the Chain

The consequence of the nonce and the zero-leading hashing:

1. It becomes **impossible to alter** a transaction without redoing the work to reestablish a valid block.
2. **In the meantime** a new additional block may have been mined…
3. “If a majority of CPU power is controlled by honest nodes, the honest chain will grow the fastest…” ← **Nakamoto’08**



Purpose of the Hash and Proof-of-Work

Hash of the previous block

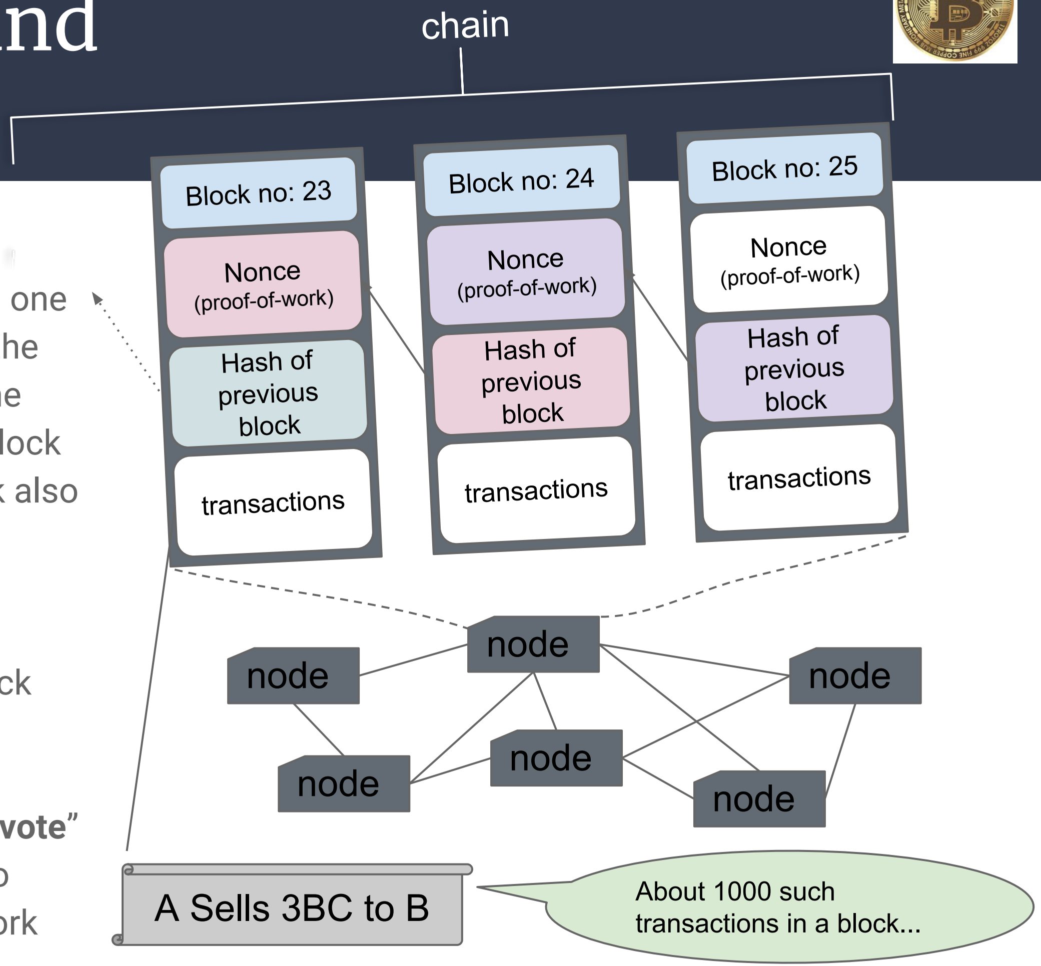
- Proof that a block was ***made after*** the previous one
- A node cannot make a block without knowing the previous one; **cannot work ahead** of the timeline
- If one wants to change/falsify (any) previous block one must change/falsify **any subsequent** block also

Proof of work

- Makes it **very expensive** (in compute power, electricity) to change/falsify also the latest block

In combination

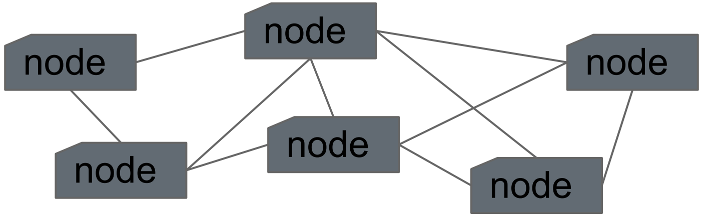
- A form of voting, consensus by "**one CPU, one vote**"
- To subvert the blockchain, an attacker needs to control at least ¼ of all CPU power in the network



Why Run a Node (Mining)

If the new block is accepted, the creator gets credited with (currently 12.5) Bitcoin:

- Each new block contains a transaction that gives the node's owner some fresh Bitcoin
- So an accepted block enriches the node owner
- A rejected block has no effect



This is ***Bitcoin mining***, the node's reward for helping to maintain the blockchain

Bitcoin Observations and Challenges

By design, only one new block per 10 minutes

- Hence settlement time >= 10 mins: high latency
- At most 3 transactions/sec: low scalability
- Solutions: Sidechains, Bitcoin-NG, Ethereum, ...

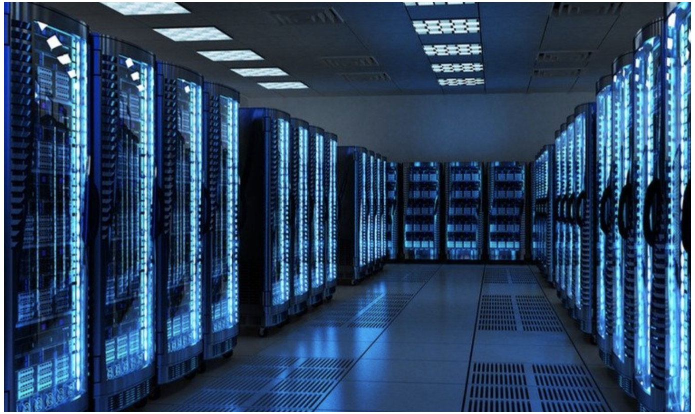
Sensitive to compute node concentration

- Mining pools
- Many Bitcoin nodes are in China

Proof of work is wasteful

- Extremely costly in CPU-time/electricity/CO_2
- Cost is the same for $1 and $1,000,000 transaction
- ***Proof-of-space***: Use disk space for “voting” instead?
- ***Proof-of-stake***: Use Bitcoin ownership for “voting” instead?

Lots of data-replication



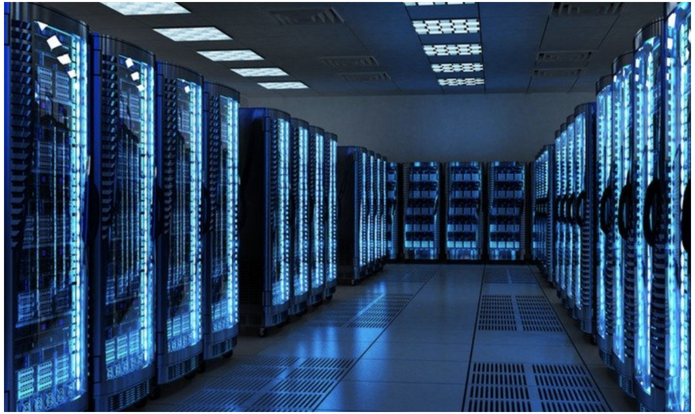
Estimated annualised global mining cost: $3,246,936,129

Close to electricity consumption for the Czech Republic!

References

[1] Satoshi Nakamoto: Bitcoin: A peer-to-peer electronic cash system, 2008. https://bitcoin.org/bitcoin.pdf

[2] Lauri Hartikka. A blockchain in 200 lines of code. March 2017. https://medium.com/@lhartikk/a-blockchain-in-200-lines-of-code-963cc1cc0e54



Thanks!