Advanced Network Security

-. Bitcoin

Jaap-Henk Hoepman

Digital Security (DS)
Radboud University Nijmegen, the Netherlands
@xotoxot // 🖂 *jhh@cs.ru.nl* // 🖰 *www.cs.ru.nl/~jhh*

Bitcoin

Who am I?

- Tommy Koens
- PhD student on Privacy & Security in Cryptocurrenc
- Promotor: Bart Jacobs; Supervisor: Jaap-Henk Hoepman
- Also working at ING's Cyber Security team
- Contact: tkoens@cs.ru.nl

Today’s topics

- On Bitcoin
- Bitcoin transactions
- The Bitcoin network and actors
- Mining and incentives
- Attacks and possible solutions
- Other uses of a blockchain

Payment systems – Some properties

- Cash – transactions anonymous, slow on a global sc

- Online banking – central system, not anonymous
- E-cash (Chaum's) – anonymous, centralized

- Bitcoin – decentralized, not anynomous
  - Over 600 other cryptocurrencies
  - See: https://coinmarketcap.com/
- ZCash – decentralized, anonymous

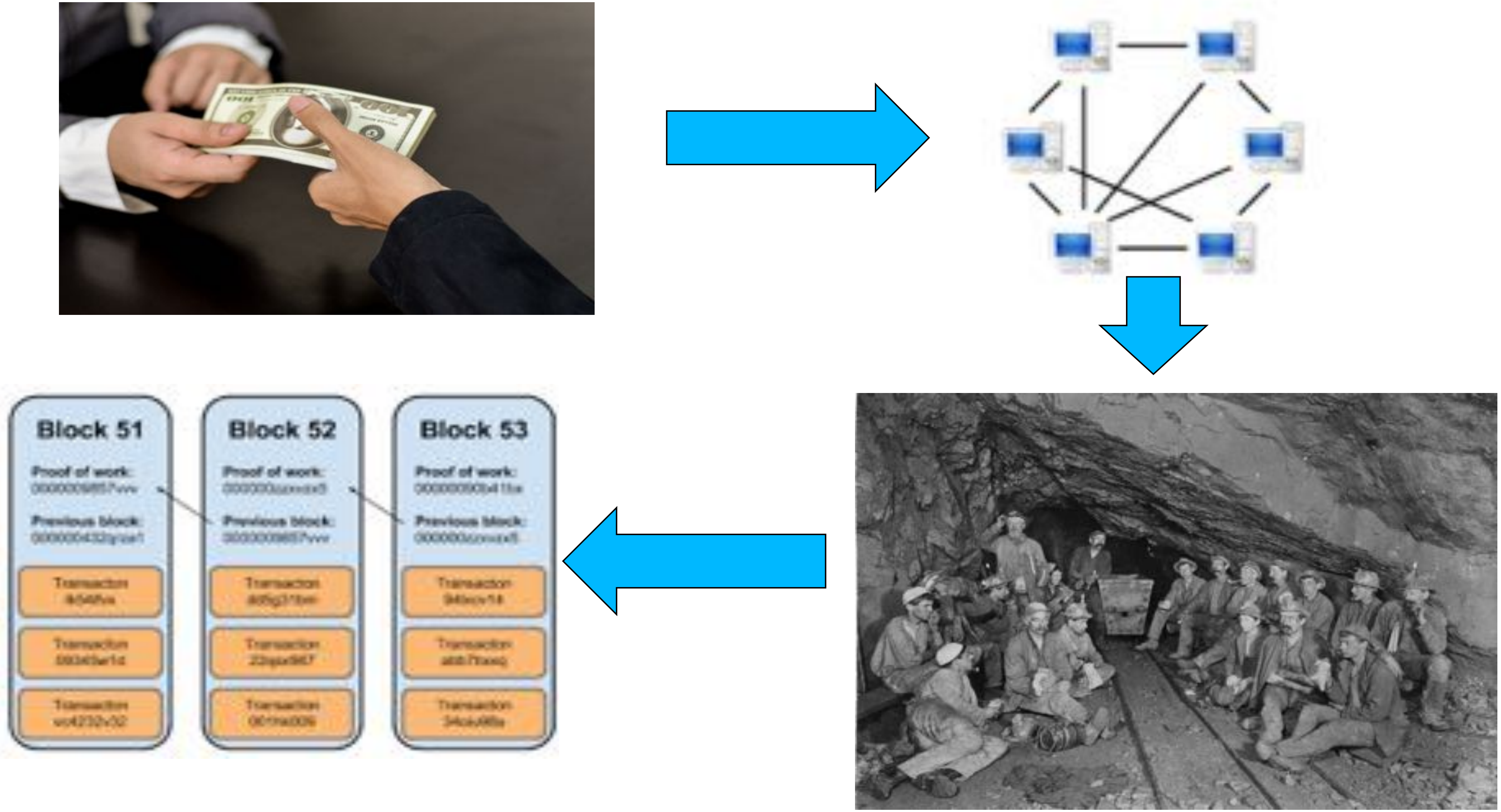
On Bitcoin

- Bitcoin: the paper
  - Satoshi Nakamoto, 2008
  - Bitcoin: A Peer-to-Peer Electronic Cash System
- Bitcoin: the system
  - A trustless payment system, backed by cryptography
- bitcoin: the coin
  - One bitcoin (BTC; 1200 €) consists of one hundred million Satoshis.

Why is Bitcoin so interesting?

- Before 2009, several proposals were made for electronic cash, like E-cash (Chaum, 1983); BitGold (Szabo, 1998); b-money (Dai, 1998)
- However, Bitcoin combines the best aspects of these technologies to achieve distributed consensus
- To achieve distributed consensus Bitcoin uses a technology called blockchain

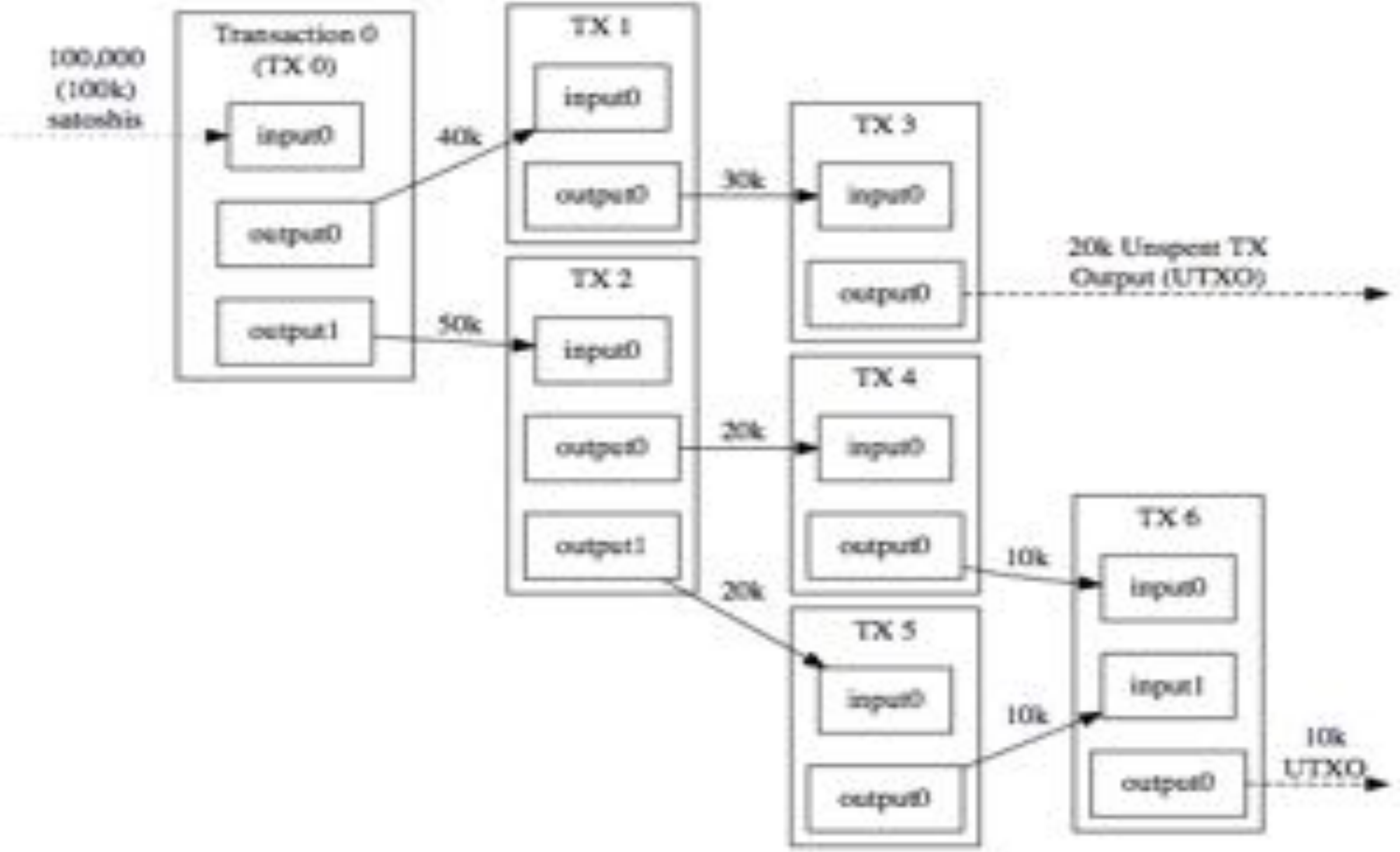
How does Bitcoin work? High level overview



Agenda

- On Bitcoin
- Transactions
- Mining / incentives
- Blockchain(s) and consensus
- Attacks
- Other uses of a blockchain

Regular transactions and fees

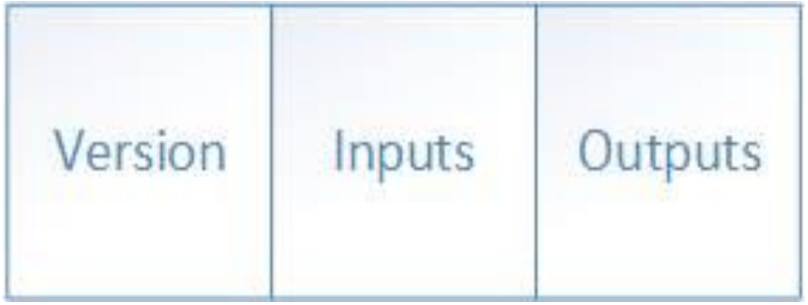


Triple-Entry Bookkeeping (Transaction-To-Transaction Payments) As Used By Bitcoin

Source: https://bitcoin.org/en/developer-guide#block-chain-overview

Transactions

- A transaction frame

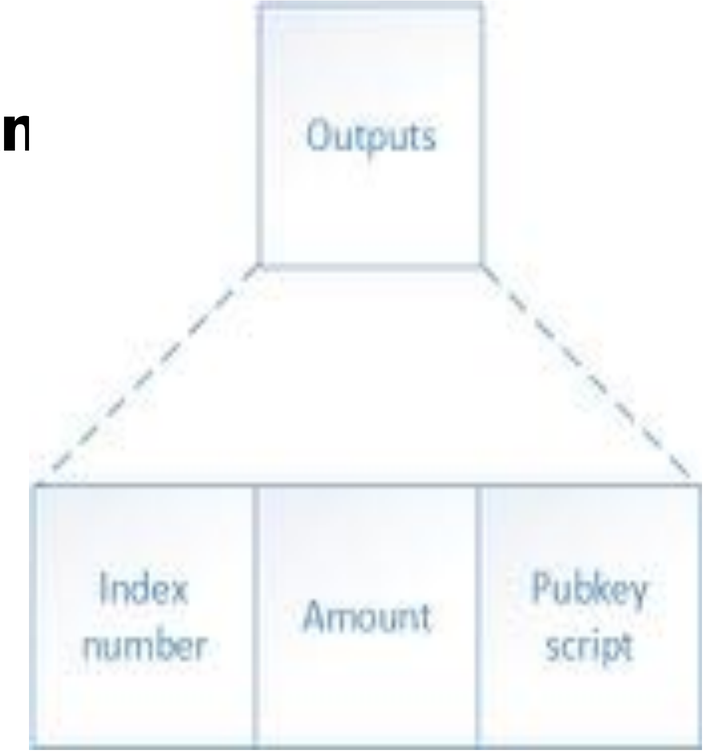


Version	Inputs	Outputs

- Version – Which protocol version is used
- Inputs – Proof ownership of coins
- Outputs – Set requirements to proof ownership
- An Input always references to an (previous) Output

Transaction: Outputs frame

- Index number – Location in the transaction (sequential. 0, 1, etc)
- Amount – Number of coins sent
- Pubkey script – Conditions set to spend the An



Transactions: Inputs frame

- **Transaction identifier – Uniqueliy identifies a transaction (SHA256d)**

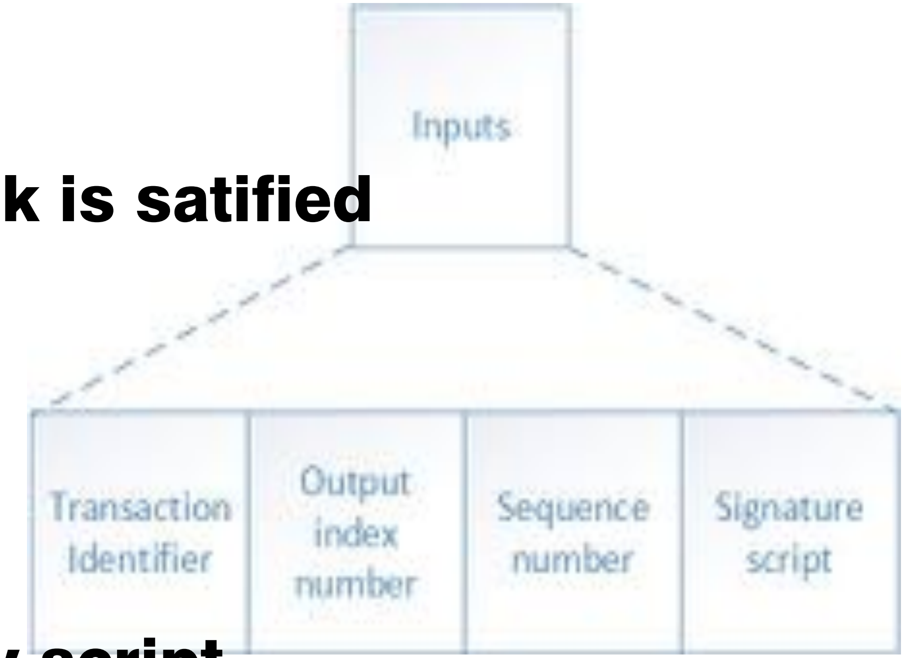
- **Outut index number – References to a particular output from which coins are spent**

- **Sequence number – mine tx when timelock is satified**

- **Signature script – Provides parameters**

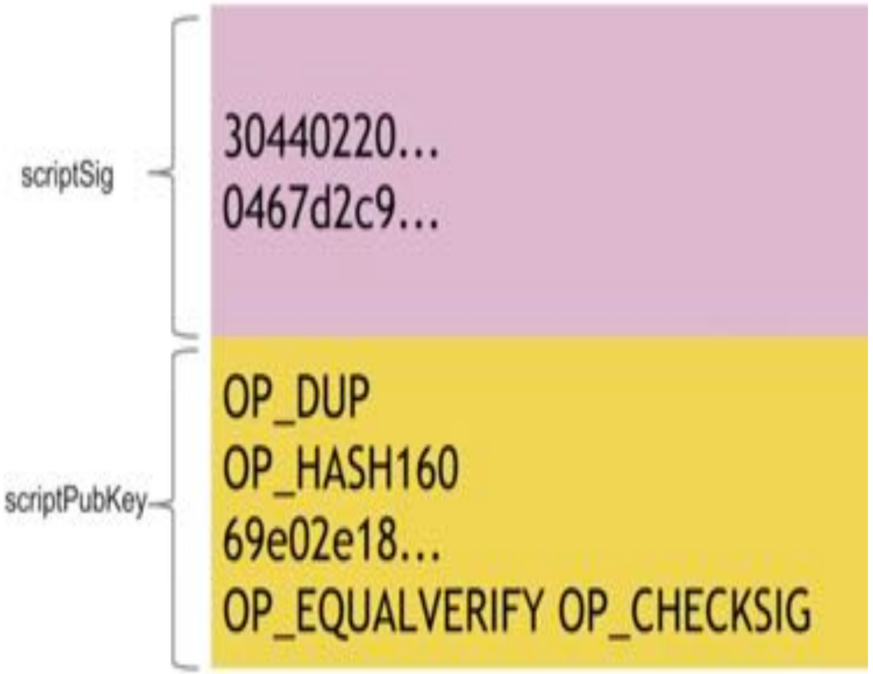
  to satisfy the Pubkey script

- **Combining Signature script with a Pubkey script**

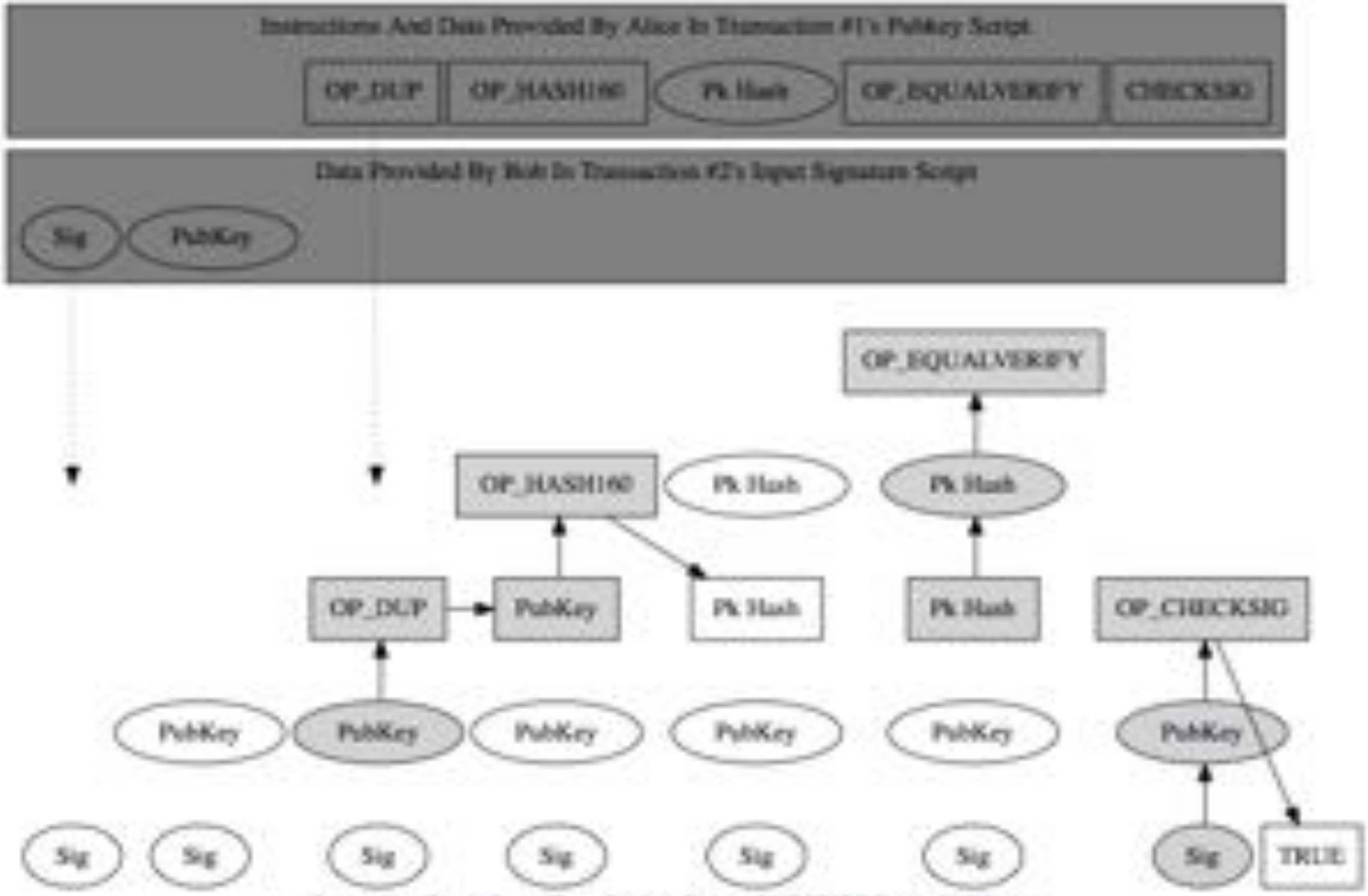


Transactions: Script validation

Scripts: Stack based language

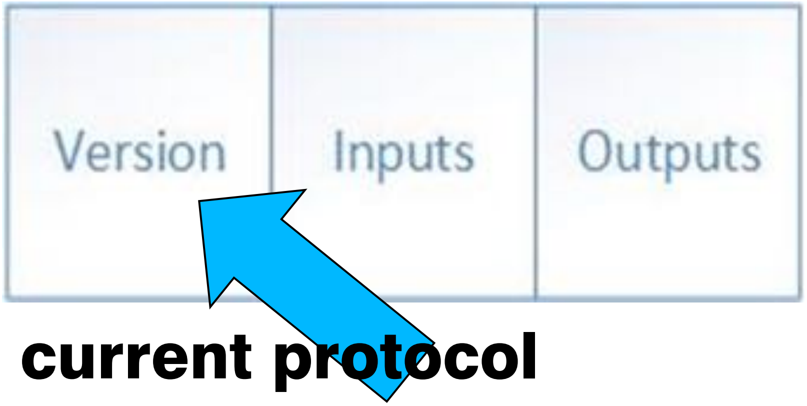


```
30440220...
0467d2c9...
OP_DUP
OP_HASH160
69e02e18...
OP_EQUALVERIFY OP_CHECKSIG
```



Evaluation Stack Over Time During Succesful P2PKH Script Validation

Transactions: Validity rules



Version | Inputs | Outputs

When is a transaction valid? E.g.:

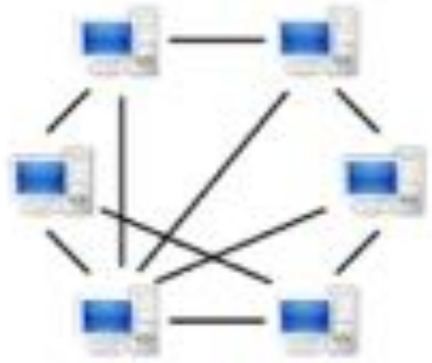
- **It should confirm to the rules according to the current protocol version format**
- **The amount of the transaction cannot be larger then the sum of the total inputs**
- **Proof of ownership must be present – script validation**
- **See: https://en.bitcoin.it/wiki/Protocol_rules#Transactions**

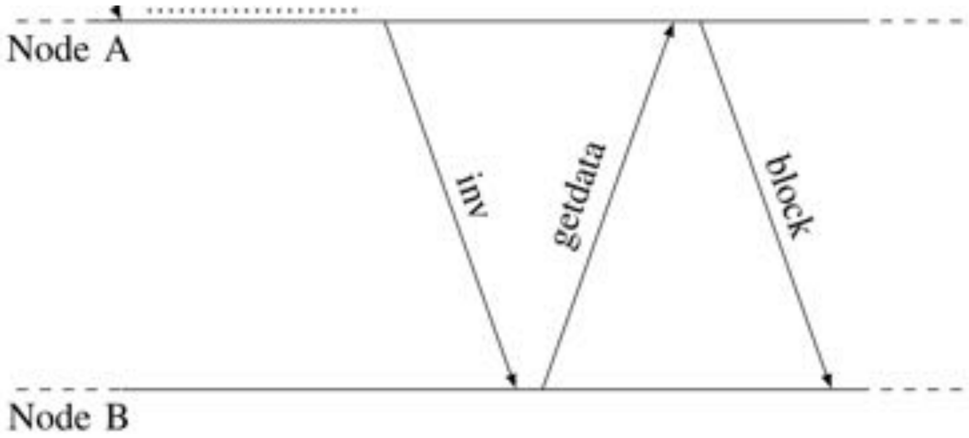
Agenda

- On Bitcoin
- Transactions
- The Bitcoin network / actors
- Mining / incentives
- Attacks
- Other uses of a blockchain

The Bitcoin network / actors

- P2P network
- Propagation method: <inv> & <getdata>
- No broadcasts. Why not?





The Bitcoin network / actors

- Buyers – create transactions
- Sellers – offer goods
- Miners / Mining pools – provide network security
- Core developers – Maintain Bitcoin code
- Community – Discussion and direction / run DNS servers
- Government / Law enforcement / Financial institutions
- Other parties (servcies): Exchanges / Wallet providers / Mixers

Bitcoin types

- Bitcoin Core
  - ➢ Vanilla Bitcoin
- Bitcoin XT (fork)
  - ➢ Blocksize debate (8 MB blocks)
- Bitcoin classic (fork)
  - ➢ Blocksize debate (2 MB blocks)
- Bitcoin unlimited (fork)
  - ➢ Blocksize debate (block size by consensus)

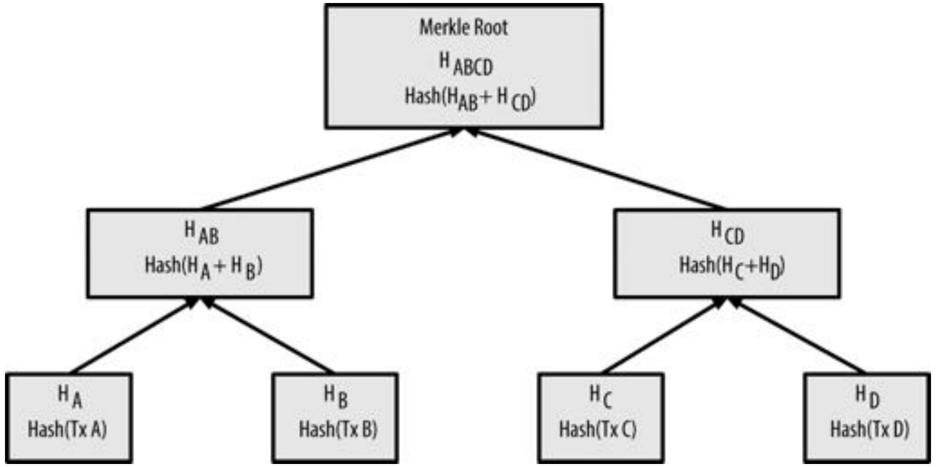
Agenda

- On Bitcoin
- Transactions
- The Bitcoin network / actors
- Mining / incentives
- Attacks
- Other uses of a blockchain

Mining blocks

- How to prevent a double spend?
- “The payee needs proof that at the time of each transaction, the majority of nodes agreed it was the first received.” (Nakamoto, 2008)
- Miners secure the network, by timestamping sets of transactions
- Set of transactions = block

Mining blocks – Preparation



- **Collect and validate transactions**
  - **If not valid, ignore transaction**
- **Store transactions in mempool (volatile memory)**
- **Select transactions and create a Merkle Root**
- **Selected transactions are store in the 'block body'**
- **The Merkle root goes into the 'block header'**
- **A block has a fixed size (in Bitcoin, currently) of 1 MegaByte**

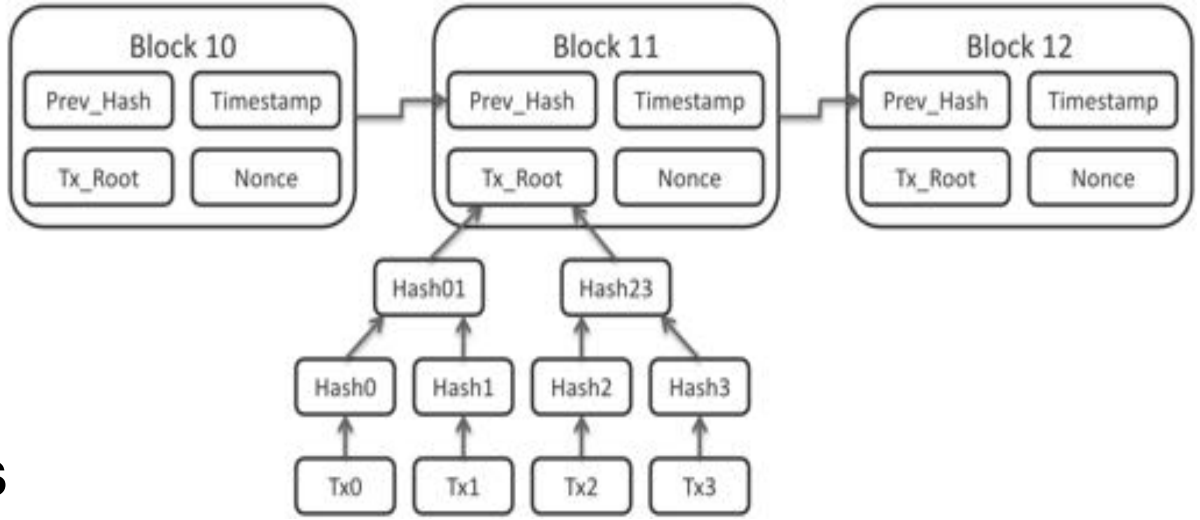
Mining blocks – Block body

The block body contains:

- Coinbase tx
- tx1
- tx2
- …
- tx-n

- **Transactions**
- **Coinbase transaction**
- **If succesfully mined (block header), the miner sends 12.5 BTC (block reward) to himself**
- **Thus, Bitcoins are generated out of thin air, each time a block is mined**
- **Block reward halfs every 210.000 blocks**
- **Maximum no. of BTC to be ever produced: 21.000.000**

Mining blocks



The block <u>header</u> contains:

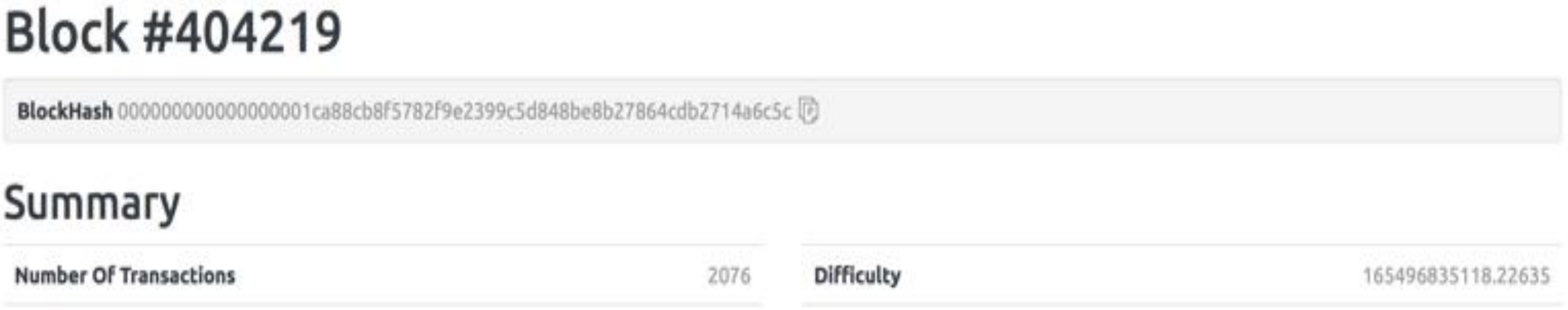
- **Version – current protocol version**
- **Hash previous block – links blocks**
- **Merkle root – from transactions in block body**
- **Timestamp – current time (Unix time)**
- **Bits – represents current difficulty**
- **Nonce – 32-bit number, starts at 0**

Source: https://21.co/learn/bitcoin-mining/#the-merkle-root

Mining blocks

- Mining is finding a hash that matches the target
- Target – a hash with a specific number of leading zeros
- Hash the block header, if no match, nonce++, repeat.
- Difficulty – How difficult it is to find the next block hash (i.e. # of zeros)
- Current

Meaning



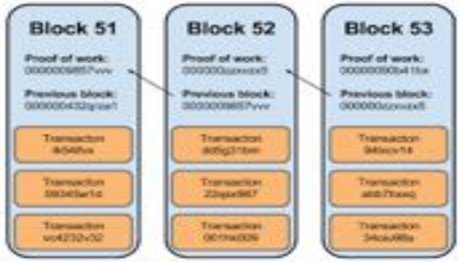
Block #404219

BlockHash 000000000000000001ca88cb8f5782f9e2399c5d848be8b27864cdb2714a6c5c

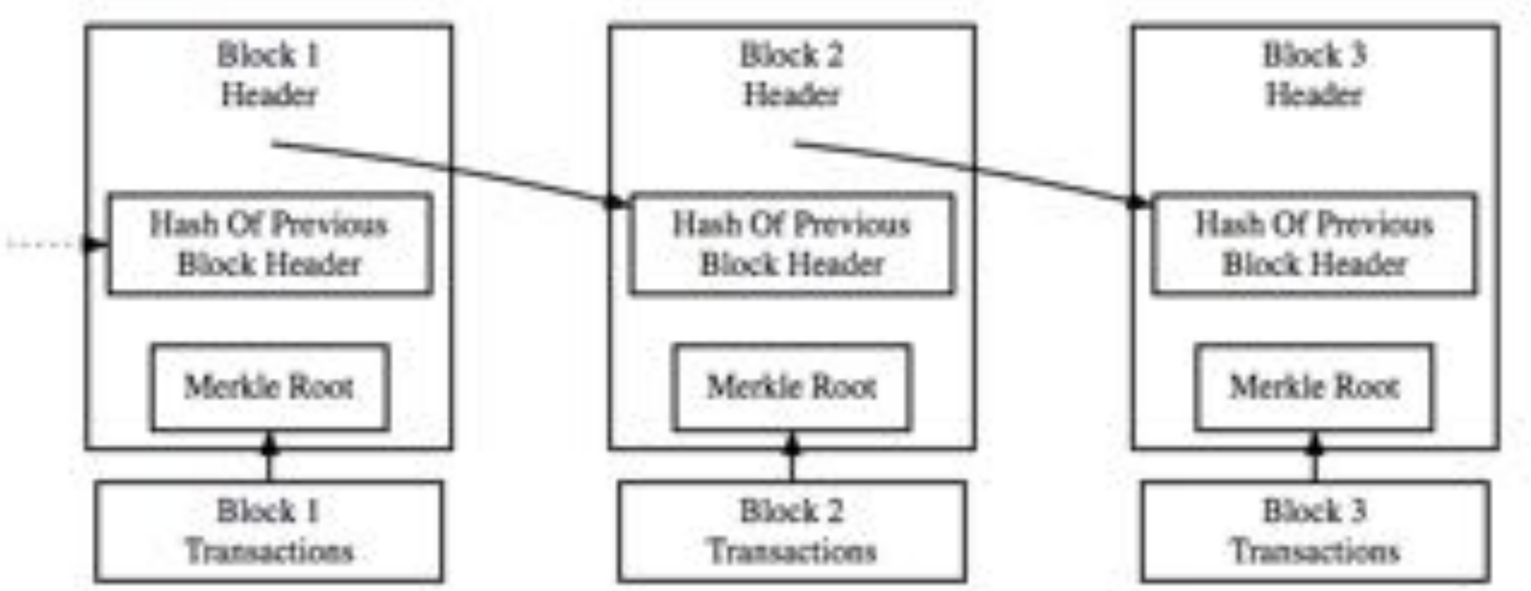
Summary

Number Of Transactions	2076
Difficulty	165496835118.22635

Mining result

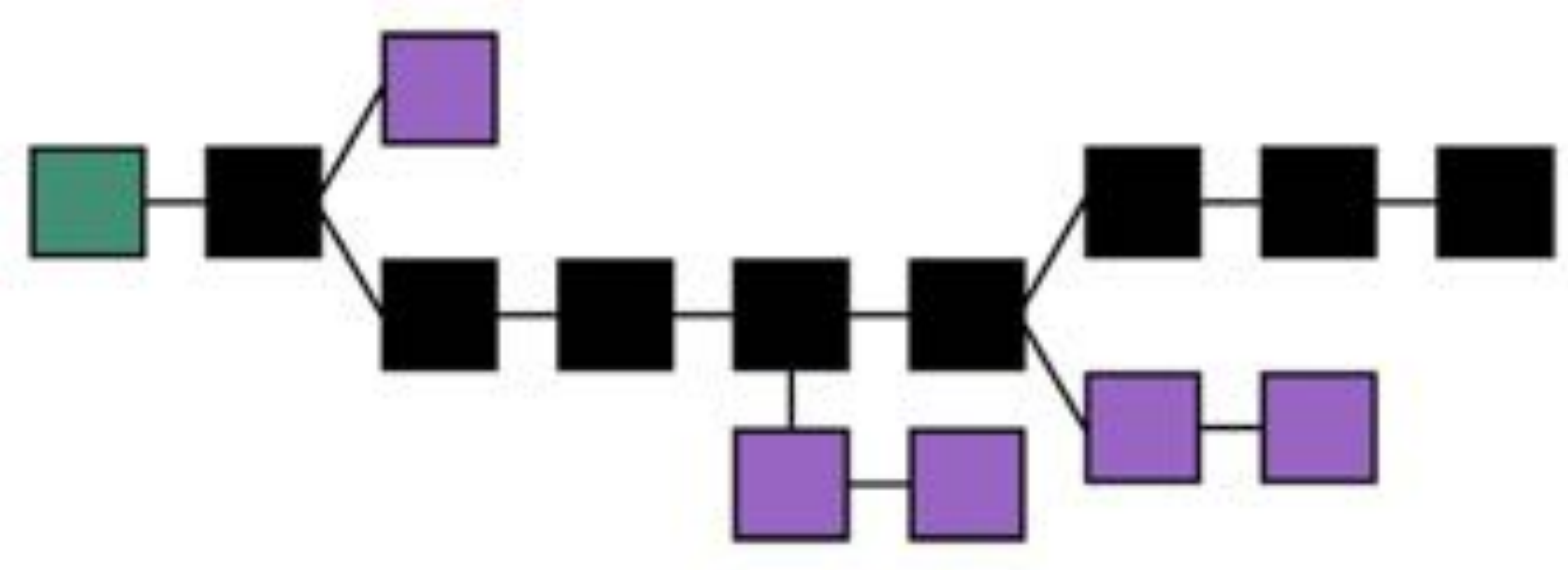


- Block is 'broadcast'
- If a node accept the block, the block is added to the blockchain
- Thus, consensus is reached; transaction and mining process starts again



Simplified Bitcoin Block Chain

Blockchain forks



How does Bitcoin prevent (or mitigate) this issue?

Mining – proof-of-methods

- Proof-of-Work – find a SHA256 hash, based on processor resource (external)

Proof-of-'useful'-Work

- PrimeCoin – find prime numbers
- Proof-of-Research – protein folding
- SolarCoin – Gain reward based on solar energy

Mining – other proof-of-methods

- **Proof-of-Work variations (e.g.):**
  - **Hash variants (e.g. BlakeCoin, Blake-256)**
  - **Cuckoo hashing, ASIC resistant (Tromp, 2015)**
- **Proof-of-Stake – Coins as internal resource (e.g. Kind and Nadal, 2012)**
- **Proof-of-Stake-time – Time as a resource (Milutinovic, 2016)**
- **Proof-of-Space – Disk space as a resource (Dziembowski et al., 2013)**

Agenda

- On Bitcoin
- Transactions
- The Bitcoin network / actors
- Mining / incentives
- Attacks
- Other uses of a blockchain

Attacks and Concerns

Just to mention a few:

- Finney attack
- 51% attack
- Power concerns
- Scalability (blockchain / transaction)
- Privacy
- Decentralization

Finney attack

How can we prevent (or mitigate) this attack?

- Mine a block *b* which includes a transaction *t1* with coins *xyz* sending to self
- Buy goods with coins *xyz* in transaction *t2* from vendor
- Once goods obtained, send block *b*
- *t2* likely will be in included in block *b*'
- *b* is likely the longest chain (sent first), so *t1* prevails, *t2* is discarded
- Goods are obtained – for free.

51% attack

<u>How can we prevent (or mitigate) this attack?</u>

- **Suppose a miner obtains more than 50% of the total network's hashing power**
- **The attacker can create blocks faster than the rest of the network**
- **Which enables double spends (see Finney attack)**
- **>50% hashing power = 100% probability of double spend**
- **<50% hashing power = lower probability (but not 0!)**

Power concerns

- Bitcoin's PoW currently is 1.27 exahash

Kilo, Mega, Giga, Tera, Peta, Exa (10 ^ 18), Zetta, Yotta.

- That's almost the amount of Ireland's yearly energy consumption (O'Dwyert & Malone, 2013)
- Is Bitcoin really cheaper than a central financial institution?
- Possible solution: Other proof-of-methods aim to solve this issue, like proof-of-stake

Propagation / verification time

- Transaction propagation – couple of seconds on average for 95% of the network – approx. 3 seconds on average.
- Block propagation (max 1 MB) – about 40 seconds (for 95% of the network) – 12,6 on average
- What happens if we increase the block size, as with Bitcoin Classic (2 MB blocks), or Bitcoin XT (8 MB blocks)?
- Block generation frequency: 10 minutes, on average.
- Want to be pretty sure? 6 blocks = 60 minutes

Scalability

- Blockchain is over 100 GB in size – and growing
  - Not an ideal scenario for the Internet-of-Things
  - Cryptonite: fixed blockchain size by separating blockchain functionalities (Bruce, 2014)

- Bitcoin can handle at most 7 transactions per second
  - (1.000.000 bytes block size / 240 byte transaction (lower bound)) / 600 seconds = 7
  - Segregated Witness (Wuille, 2015) – approx 45% increase for

Privacy (1/3)

Is Bitcoin privacy friendly? No.

- **Public blockchain links transactions (unlinkability)**

Examples:

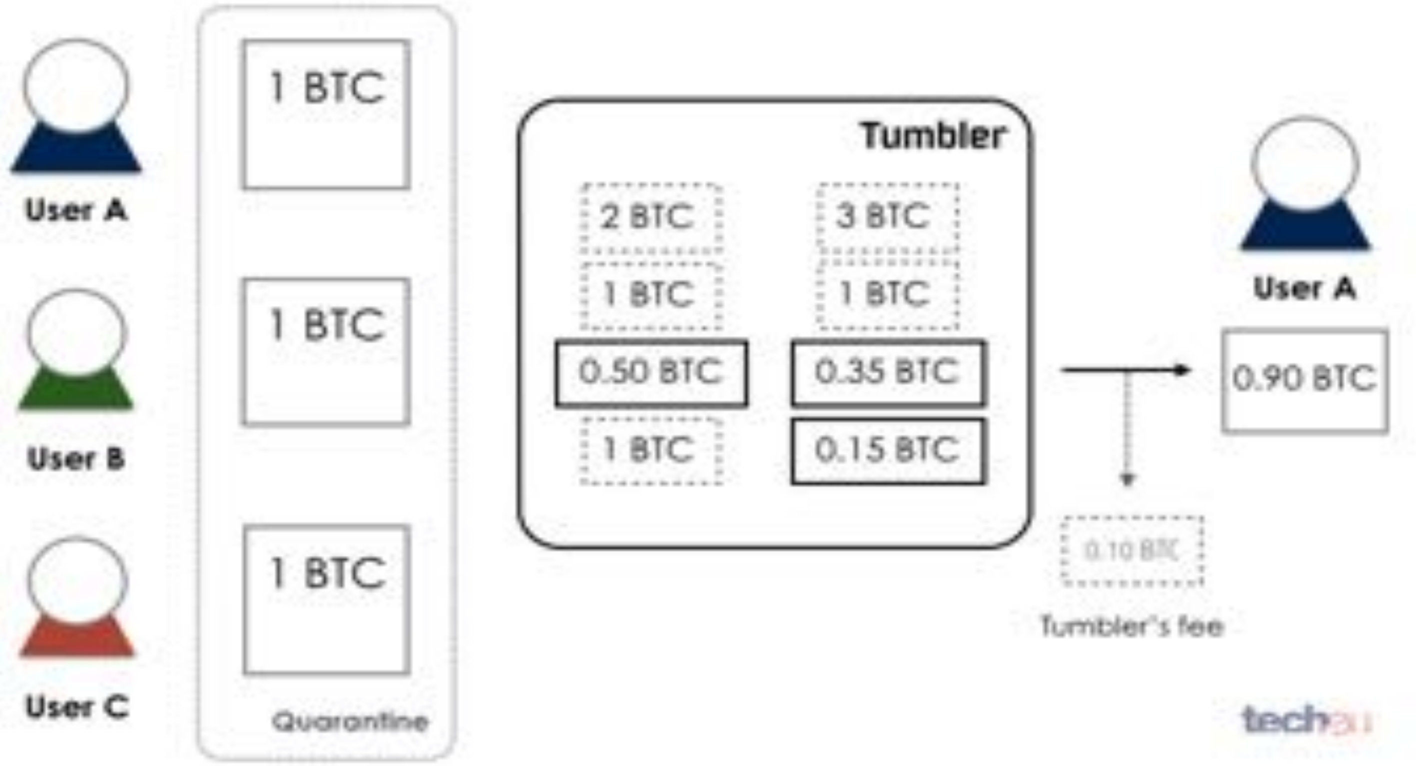
- **MtGox**
- **Silk Road**
- **DD4BC**

See: A fistful of bitcoins: characterizing payments among men with no names (Meiklejohn et al., 2013)

Privacy (2/3)

<u>What is the main issue here, from Bitcoin's perspective?</u>

- **Mixers – break the link between payer and payee**



Privacy (3/3)

- ZeroCash provides privacy – the protocol
- Improved version of ZeroCoin
- Zcash – the currency (referenced as ZEC), implementation of ZeroCash
- Key cryptographic component: zk-SNARKS
- Zero-knowledge succinct non-interactive arguments of knowledge
- Main property over zk: require no interaction bewteen prover and verifier
- See: Zerocash, Decentralized Anonymous Payments from Bitcoin

Decentralization. Who is in charge?

- Core Developers do the coding
  - Community has its say through forums
  - Users are free (not) to use the software
- Payers/Payees perform transactions
- Miners ensure security / generation of new coins
- Merchants offer goods for BTC

Agenda

- On Bitcoin
- Transactions (regular / pay-to-script-hash)
- The Bitcoin network / actors
- Mining / incentives
- Attacks
- Other uses of a blockchain

Blockchain, beyond transactions

- Storage of data – pictures, texts, patents
  - Genesis block: 'The Times 03/Jan/2009 Chancellor on brink of second bailout for banks'

- National money – Ecuador
- Carbon dioxide recording
- DNS registration – NameCoin
- Identity management – onename.com
- Transfer of assets – mortgages, car keys(!?)

Real world implementations of blockchain tech

Beyond the blockchain hype, some examples:

- **Microsoft – Blockchain as a Service**
  - **Run a blockchain node at the service provic**

- **IBM – Oil trading platform (based on Hyperlec**
- **MAERSK – Freight tracking**
- **Switserland's post-trade market – bonds (debt investment) life cycle**
- **Sweden's land registry authority – land registration on blockchain**

Summary

- Many types of payment systems – most are centralized
- Bitcoin achieves decentralized consensus
- Bitcoin essentials: Transactions, P2P network, Mining, and Stakeholders
- Many (open) issues – Privacy, Scalability, Power concerns, Decentralisation
- Many applications - Payment system, Contracts, Data storage, Car keys

Questions