ELT-53206 Peer-to-Peer Networks

P2P BitCoin: Technical details

Mathieu Devos
Tampere University of Technology
Department of Electronics & Communications Engineering
mathieu.devos@tut.fi – TG406

Outline

1. What is Bitcoin
    1. Basic explanation
    2. Inventor
    3. BTC vs Traditional Monetary systems
2. Technical basics
    1. Cryptographic hash functions
    2. Digital signature
    3. Transaction Records
    4. Proof of Work protocols
3. BitCoin technical explanation
    1. Sending BTC from A to B
    2. Block chain & mining
    3. Vulnerabilities?
4. Outcome

Outline

1. What is Bitcoin
    1. Basic explanation
    2. Inventor
    3. BTC vs Traditional Monetary systems
2. Technical basics
    1. Cryptographic hash functions
    2. Digital signature
    3. Transaction Records
    4. Proof of Work protocols
3. BitCoin technical explanation
    1. Sending BTC from A to B
    2. Block chain & mining
    3. Vulnerabilities?
4. Outcome

BitCoin – Basic explanation

- Open Source P2P currency
- Currency = ”a type of unregulated, digital money, which is issued and usually controlled by its developers, and used and accepted among the members of a specific virtual community” – 2012 European Central Bank
- Value depends on how much users are willing to pay for it
- Similar with normal money or other valuable items
- Value based on scarcity, similar to gold
- Maximum of 21 million BTC, can’t be added like printed money
- Mining “creates” bitcoins, rewards for mining lowered by time, 21 million reached on 2140.
- Purely based on math and distributed systems, everybody knows everything!

BitCoin – Inventor and origin

1 SATOSHI = 0.000 000 01 BTC

- Inventor = Satoshi Nakamoto
- Pseudonym
  - Single person (genius) or group
  - Timestamps of posts and perfect English > US timezone
  - Best guess: Nick Szabo => base on stylometric analysis, genius interested in virtual money, produced paper ”bit gold” before bitcoin paper was released
  - Dorian Nakamoto? => After interview was conducted where he denied everything, original forum account came to life after 5 years and posted: ”I am not Dorian Nakamoto”
- Currency with over 5 billion USD value (max 13 billion USD) based on a paper from unknown writer(s)...
- BUT! Fully based on math, code written by early inventors/adoptors (rich people now, satoshi owns >11mil BTC)

BitCoin – BTC vs Traditional Money Systems

- BTC = Online only (physical coins excist but they are representations of virtual address and wallets)
- No backup plans!
  - Lose the key to your private wallet? -> goodbye BTC (ever increasing scarcity)
  - Wallet got stolen? No way to get it back from anybody
- No government interaction possible
  - How to report your taxes?
  - Full anonimity possible through use of tor system
  - Illegal usage?

Outline

1. What is Bitcoin
    1. Basic explanation
    2. Inventor
    3. BTC vs Traditional monetary systems
2. Technical basics
    1. Cryptographic hash functions
    2. Digital signature
    3. Transaction Records
    4. Proof of Work protocols
3. BitCoin technical explanation
    1. Sending BTC from A to B
    2. Block chain & mining
    3. Vulnerabilities?
4. Outcome

Technical basics

- Before we dive into bitcoin and similar e-currency, need basic technical understanding of some well-known mechanics
- First learn the tools, then see how BTC uses them to together
- Cryptographic hash functions, data verification
- Digital signature and authenticity of file ownership
- Transaction record, logbooks

Home Welcome to Blockchain

More...

Height	Age	Transactions	Total Sent	Relayed By	Size (kB)
331697	< 1 minute	128	50.0650928 BTC	62.210.125.106	44
331696	20 minutes	1469	11,980.59 BTC	KnCMiner	731.6
331695	18 minutes	1024	72,841.57 BTC	Polmine	292.29
331694	40 minutes	1024	11,323.31 BTC	Polmine	275.47
331693	1 hour 35 minutes	127	1,260.81 BTC	KnCMiner	61.26
331692	1 hour 36 minutes	144	2,191.21 BTC	Discus Fish	98.12

Cryptographic hash functions

- A function that takes any size of digital data and transforms it to a digital data with fixed length
- Input => (hash-function) => Digest
- Has to be pseudorandom!
  - Difficult to find 2 different inputs that lead to the same
  - No correlation between input and digest
- Will this have enough possibilities?
- Example: SHA-256, 32bytes = 256bit = $1.16*10^{77}$
- Going from a smaller input to larger digest = upward hashing
  - Used mostly to store passwords
  - Hash of (plaintext password + random salt) = unique digest
  - Store digest and salt, never store passwords!
- Common usage of hashes: check file integrity
  If 1 bit is off, totally different digest

Digital signature

- How to prove you own a file?
- Alice sends Message ”M” to Bob, how does he know he got it from Alice?
- Alice has:
  - Signature key sk private
  - Verification key pk public
- Signature function
  - Input: Message and Signature key
  - Output: Signature of Message (unique to Alice)
  - M + sk => sM
- Verification function
  - Input: Message, Signature of Message, Verification Key
  - Output: Yes or No
  - M + sM + pk => Yes/No

Transaction Records

- BitCoin = collected entries in ledger, no real coins, just history of transactions
- Alice wants to transfer to Bob
- Alice has:
  - Verification key vkA public
  - Signature key skA private
- Bob has:
  - Verification key vkB public
  - Signature key skA private
- Alice has received previous transactions (to proof she has the **capability** of doing a transfer herself)
- Transaction log regarding Alice:
  - Alice received 10 BTC from Charlie (vkC, 10 → vkA)
  - Alice received 60 BTC from Daniel (vkD, 60 → vkA)
  - Alice received 35 BTC from Ella (vkE, 35 → Alice)
  - Total received = 105 BTC

Transaction Records - continued

- Now Alice has the proof to send BTC to BoB, example 100 BTC
- Digest each transaction entry
  - Charlie → Alice ➔ dC
  - David → Alice ➔ dD
  - Ella → Alice ➔ dE
  - People want to check ownership? Take transaction entry from log, apply cryptographic function and compare digest
- Alice's her transaction of Bob takes these digests as input, her actual new transaction becomes the output

dC	vkB, 100
dD	vkA, 3
dE	signature of this table(proof of ownership from Alice)

- But! 100+3 != 105 ➔ Transaction fee
- Broadcast entire entry!

Proof of Work protocols

- Solve the puzzle!
- Difficult to find, easy to verify
- Examples:
  - Counter against DOS ➔ Solve puzzle first before accept service
  - Counter against spam
  - CAPTCHA
- Challenge (C)(Input) and Proof (P)(Nonce)
- Hash the challenge and the proof until value
  - Output is specific to difficulty of puzzle
  - Requirement: amount of needed leading 0's in output bit-string
  - 30 leading zero's? Average need of $2^{30} = 1.07 * 10^9$ attempts
  - Add required leading zero: twice as difficult!
- Very easy to verify: take challenge and proposal (proof), hash it and count leading zeros.

Outline

1. What is Bitcoin
    1. Basic explanation
    2. Inventor
    3. BTC vs Traditional monetary systems
2. Technical basics
    1. Cryptographic hash functions
    2. Digital signature
    3. Transaction Records
    4. Proof of Work protocols
3. BitCoin technical explanation
    1. Sending BTC from A to B
    2. Block chain & mining
    3. Vulnerabilities?
4. Outcome

BitCoin technical explanation

- Uses all of the earlier mentioned functions
- No timestamps, biggest proof of work (most math) leads the way
- Public records of **every** transaction
- After transaction is made, broadcast information to everybody
- Verification done through mining
- Mining is (was) big business!
  - Current network does ~270 PHash/sec
  - 1 Hash ~~ 7500 FLOPS
  - 2 000 000 PFlops/sec
- What about super computers?
  - Top500.org -> #1 has 55 Pflops/sec peak (33PFlops/sec avg)
  - Top 10 together ~150 PFlops
  - <0.01% of the BTC mining network...

Sending BTC from A to B

- Based on transaction records
- Proof that you have received enough BTCs in the past
- Make new transaction to and add it to the transaction log
- This is broadcasted to **all** nodes, pure P2P
- Network is only used for this, no big data is being transferred over this, so flooding works
- You do not own any bitcoins, the networks owns the proof that you have received this many bitcoins
- You have to use previous transactions until amount of BTC >= the amount you want to send
- If >, send rest back to you, use up all transactions!

Block chain & mining

- Transaction block = page in ledger about transactions, public ledger
- BTC Miners
  - Collect all transactions (currently 60GB for **ALL** financial information)
  - Hash these transactions blocks pairwise like a folding tree until single Digest (mostly done by BTC mining pools, you don't need 60GB on your pc for this)
  - Add hash of previously accepted transaction block (every block links to previous one
  - These are not individual pages anymore, but linked through previous blocks, thus **block chain**
  - Transform this into a Challenge, generate your proof, hash all and check versus difficulty
- Difficulty changes over time, depends on time spent on block, average it should be 10 minutes / block. Recalibration every 2016 blocks, equals 2 weeks

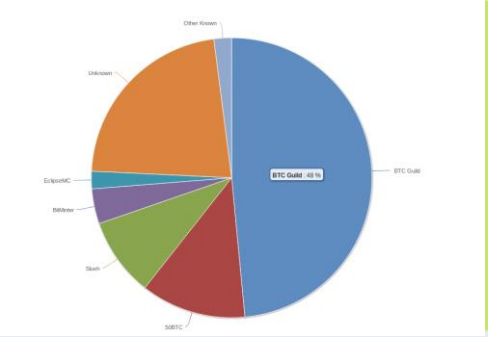
Block chain & mining - continued

- Proof found?
  Drop work and start working on a new block
- On your own this is not doable anymore.
- Part of mining pool where you earn your shares based on the amount of hashingpower you provide. Small people can join!
- Currently this is not worth it, difficulty too high, it will never show any return value and/or price of operation is too high compared to output
- Miner evolution:
  - CPU
  - GPU
  - FPGA
  - ASIC
  - ... (Neural chips? TrueNorth, ...)

Block chain & mining - continued

- Why would you mine?
  - Miner who finds the proof gets **all** the transaction fees
  - Transaction fees are currently around ~10BTC per day = 10 / (24*6) = ~0.07 BTC/block
  - Miner who mines can reward himself by setting the reward to his address, current reward = 25 BTC, halved every 210 000 blocks (+- every 4 years). Many old accounts have static amount of 50 (original BTC miner reward) on them, but the keys to the account might be lost! Forever lost bitcoins → comodity becomes rare
- This keeps on getting halved until all 21 million BTC are in use (target year = 2140)
- After (closer) to this year, transaction fees will become needed
- Miners build the blocks themselves! No fee → put your transaction on hold for a ”later” block, you might never get served!

Vulnerabilities



- Assume Alice send 50 BTC to Bob, with Digests from Charlie and David. After she has sent it, she sends them to Ella, an account owned by Alice
  = double spending
- Bob should wait until the block gets **verified** (blockchain.info)
- But Alice can still cheat! Alice sends them to Bob, wait until verification, then sends them to Ella. After this, Alice has to provide a blockchain longer than the current one.
- Is this an issue? Alice is racing on her own against the network, she has to beat 50% of the network (currently 500 PHash/sec for 50%)
- However with mining pools this forking can happen, mining pools mostly self regulate so they do not become too big to be able to do a network takeover (>50%)
- For proper verification, wait ~6 blocks for transaction

Outcome

- Basic explanation about bitcoin
- Difference between traditional systems
- Technical functions
  - Cryptographic Hash
  - Electronic signature / ownership
  - Transaction records
  - Proof of work
- Bitcoin specific
  - Mining & block chain
  - How to do a basic transaction
  - Vulnerabilities

Any questions?

mathieudevos@tut.fi

Next week: Exam example, Beyond P2P