About cryptocurrencies and blockchains – part 1

Jyväskylä 17th of April 2018

Henri Heinonen (henri.t.heinonen@jyu.fi)

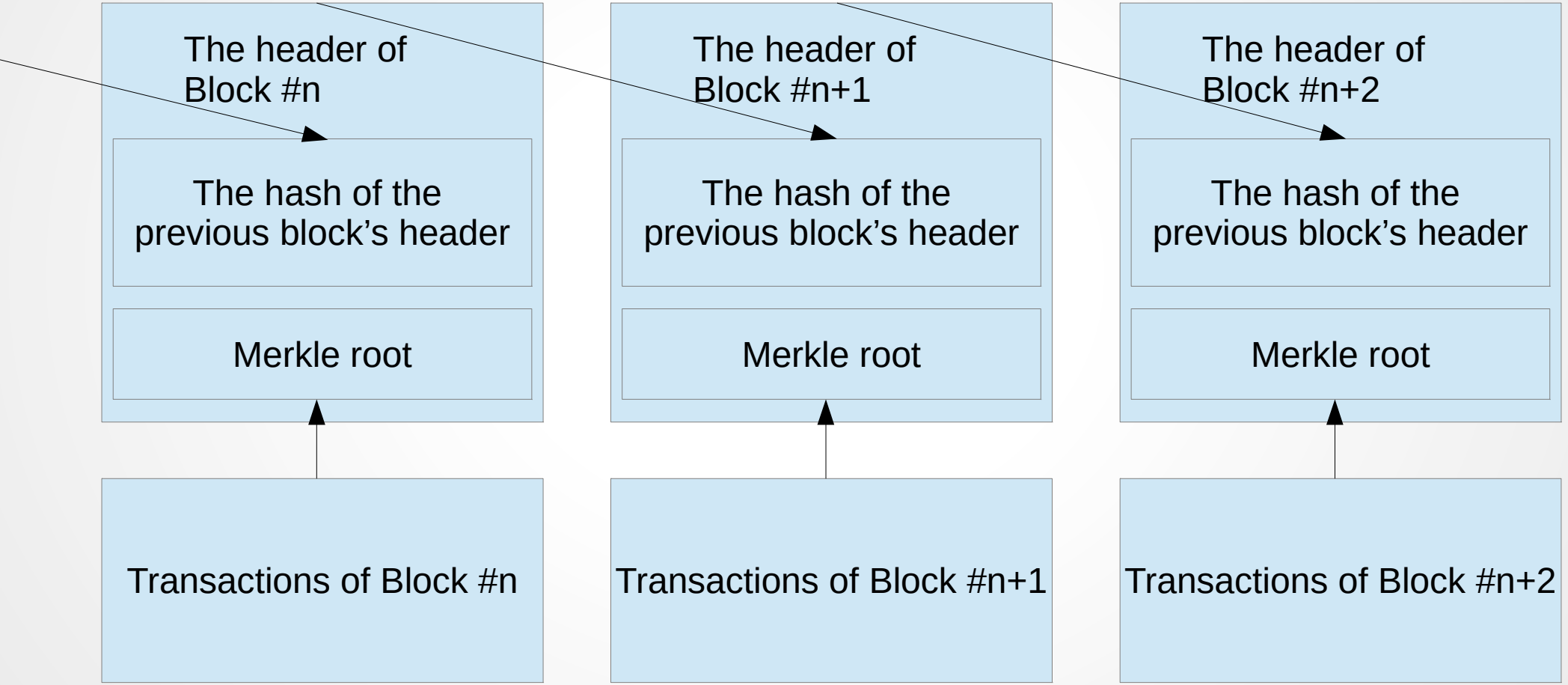
What is a blockchain?

- BitTorrent is a famous example of a peer-to-peer network (P2P) implementation.
- BitTorrent makes it possible to share files without a central server, so that every user is kind of both a client and a server.
- Peer-to-peer networks make sharing files cheap and fast, because no central server needs to be set up behind a fast Internet connection.

What is a blockchain?

- A blockchain is a database/ledger distributed on a peer-to-peer network.
- Blocks typically contain transactions and header information.
- New blocks will be generated at a constant rate.
- Chaining comes from the fact that every block refers to the previous block by using a hash pointer. An exception is the genesis block, which had nothing before it.

What is a blockchain?



How to add blocks into the blockchain?

- In peer-to-peer network a device sends data (transaction) into the network for addition into a new block: for example, a user of a bitcoin wallet sends bitcoin to his/her friend.
- Mining nodes will get this transaction and several other transactions, and will arrange a lottery to decide whose block will be added into the blockchain.
- As a reward, the miner who added the block will get the block's block reward (Bitcoin in 2018: 12,5 BTC) + transaction fees, which are coming from the senders of the transactions.

  Source: Chainfrog Blockchain Lecture part 1: https://www.youtube.com/watch?v=TyZBV_w4MHg

How to add blocks into the blockchain?

Several consensus algorithms in use:

- Proof-of-work: Bitcoin, Litecoin, Ethereum...
- Proof-of-stake: Peercoin, Ethereum (in the future)…
- Practical Byzantine fault tolerance: Hyperledger Fabric.
- Proof of elapsed time: Hyperledger Sawtooth.

Source: Chainfrog Blockchain Lecture part 1: https://www.youtube.com/watch?v=TyZBV_w4MHg

Proof-of-Work

- The roots of Bitcoin’s Proof-of-Work are in Adam Back’s Hashcash system created in 1997, which was meant to make sending e-mail computationally difficult, so that the activities of spammers would become more difficult.
- Take data and add nonce to it (concatenate data and nonce): data + nonce.
- Hash it twice: SHA256d(data + nonce) or SHA256(SHA256(data + nonce)).
- Check, if the hash is lower than the target defined by the network’s difficulty. (Check, if the hash has enough leading zeros.)
- If not, repeat it with the new value: nonce = nonce + 1.

  Sources: ”Chainfrog Blockchain Lecture part 3” https://www.youtube.com/watch?v=peckb7295fk, https://en.bitcoin.it/wiki/Difficulty, https://en.bitcoin.it/wiki/Target

What is Bitcoin?

- Bitcoin is a cryptocurrency and the first application of blockchain technologies.
- Bitcoin was originally meant to be a payment systems, because not everybody is able to get a bank account and/or a credit card.
- Bitcoin is designed so that new blocks will be generated in every ten minutes in average.
- Bitcoin was constructed to have its own token: bitcoin. Its currency unit is BTC or more formally XBT. The accepted symbol in Unicode's code point U+20BF is: ₿
- Validating bitcoin transactions needs computing work, which needs lots of electricity. This mining is incentivized by the block reward and transaction fees by every new mined block.
- There were 50 new bitcoin introduced into the system in every new block in the early days. In 2018 the block reward is 12.5 BTC/block. The block reward is halved every four years.
- The market value of bitcoin has raisen from zero to around 117 billion euros in less than a decade.

Some history of e-currencies

- David Chaum’s paper from 1983 ”Blind signatures for untraceable payments” introduced an idea of anonymous electrical money.
- DigiCash
- Ecash (In Finland a trial by Merita bank)
- b-money
- Smart contracts in the beginning of 1990’s.

Some history of e-currencies

- In October 2009 an Internet exchange sold five thousand and fifty bitcoin at 5.02 USD. The exchange rate would be 0.000994 USD/BTC. The price was calculated from the value of electricity used to produce those bitcoin. This might be the first time when bitcoin had a defined price. Source: The Bitcoin Standard: The Decentralized Alternative to Central Banking.
- Bitcoin was used as a medium of exchange for the first time in May 22nd 2010; the price for the two pizzas was 25 USD or ten thousand bitcoin. Source: The Bitcoin Standard: The Decentralized Alternative to Central Banking.

Some history of e-currencies

- Pseudonymous Satoshi Nakamoto wrote to the e-mail list of cypherpunks in 1st of November 2008 to announce that he/she has developed a new e-cash system, which is entirely peer-to-peer. Source: The Bitcoin Standard: The Decentralized Alternative to Central Banking.
- The Genesis block of Bitcoin was mined in 3rd of January 2009 or slightly later. There is a reference to a fresh The Times newspaper headline: ”The Times 03/Jan/2009 Chancellor on brink of second bailout for banks.”

The scarcity of bitcoin

- Also purely digital resources can be scarce.
- Bitcoin has been designed so that there will ever be about 21 million pieces of them.
- This limit will be reached around the year 2140.
- The vast majority of bitcoin has already been mined during the first ten years. This makes sense, because the value of bitcoin was nearly zero in the beginning and it was handy to get 50 BTC a block. Because a new block will be created every 10 minutes or so, there were 300 new bitcoin an hour.

What does bitcoin mean?

Bitcoin means various of different things.

- **Protocol**: Specification that tells how to build a distributed database (blockchain), how to parse it, how to construct transactions, what is a valid transaction, etc.
- **Network**: This is the peer-to-peer network (P2P) to be connected by nodes.
- **Currency**: bitcoin (written with a small letter or lowercase or minuscule). This is the native currency of the Bitcoin network. There will be about 21 million bitcoin. At this moment, the smallest amount of bitcoin is the one-hundred-millionth of a bitcoin, or one satoshi.
- **Open source implementation**: This is the original open source code project implemented in C++ language. Nowadays the project is known as Bitcoin Core.

Source: ”Understanding Bitcoin: Cryptography, Engineering and Economics”

Hash functions

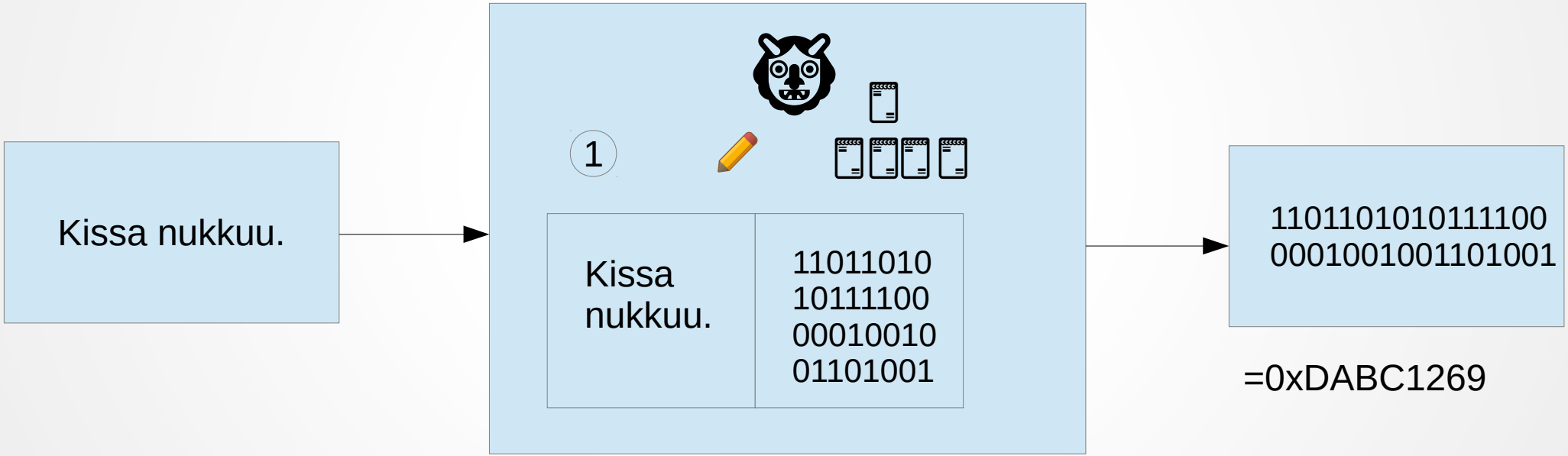
- General hash functions: Any function that can be used to map data of arbitrary size into data of exact size. These outputs of hash functions are called hash codes, hash values or hashes.
- Cryptographic hash functions: A cryptographic hash functions is a special case of a hash function with some properties that make it usable in cryptography. It is a mathematical algorithm that maps data of arbitrary size into a bit string of exact size, and it is also designed to be a one-way function meaning that it is not practical to find its inverse function.

Hash functions

- Let’s imagine a box that contains a creature, a book, a pen, Post-it Notes, and a coin.
- The heads side of the coin shows ”1” and the tails side ”0”.
- There is a hole in the left-hand side of the box that is used to put a message into the box.
- The creature will check up the book whether the same message has been encountered before. If not, the creature wil toss the coin 32 times and writes the results on the book and on the Post-it Note. If has, the creature will copy the old results from the book on the Post-it Note.
- There is a hole in the right-hand side of the box that is used by the creature to give the Post-it Note out of the box.

  Source: ”Owning Bitcoin: The Illustrated Guide to Security, Privacy, and Possibility”

Hash functions



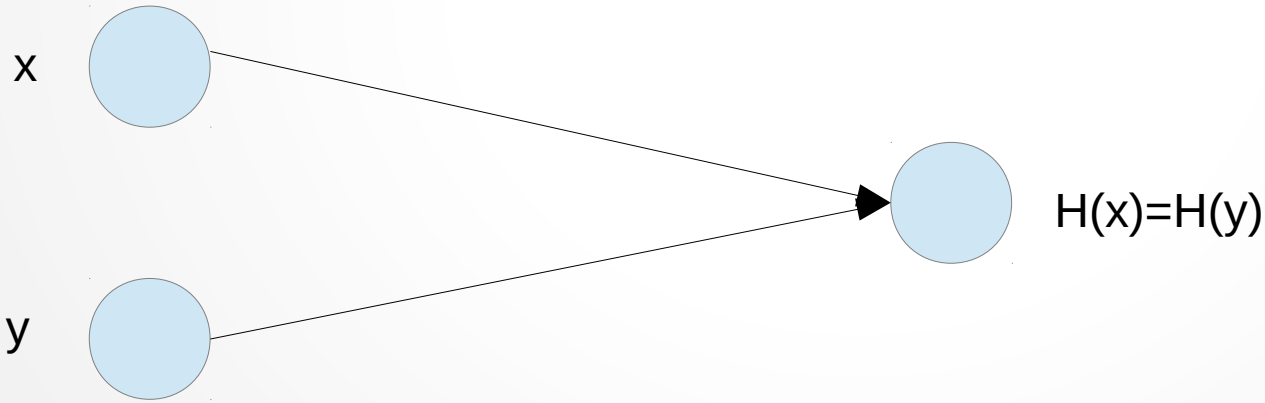
The security properties of a cryptographic hash function

- No (findable) collisions
- (other properties, too)

  Sources: ”Introduction to Crypto and Cryptocurrencies” https://www.coursera.org/learn/cryptocurrency, ”Understanding Bitcoin: Cryptography, Engineering and Economics”

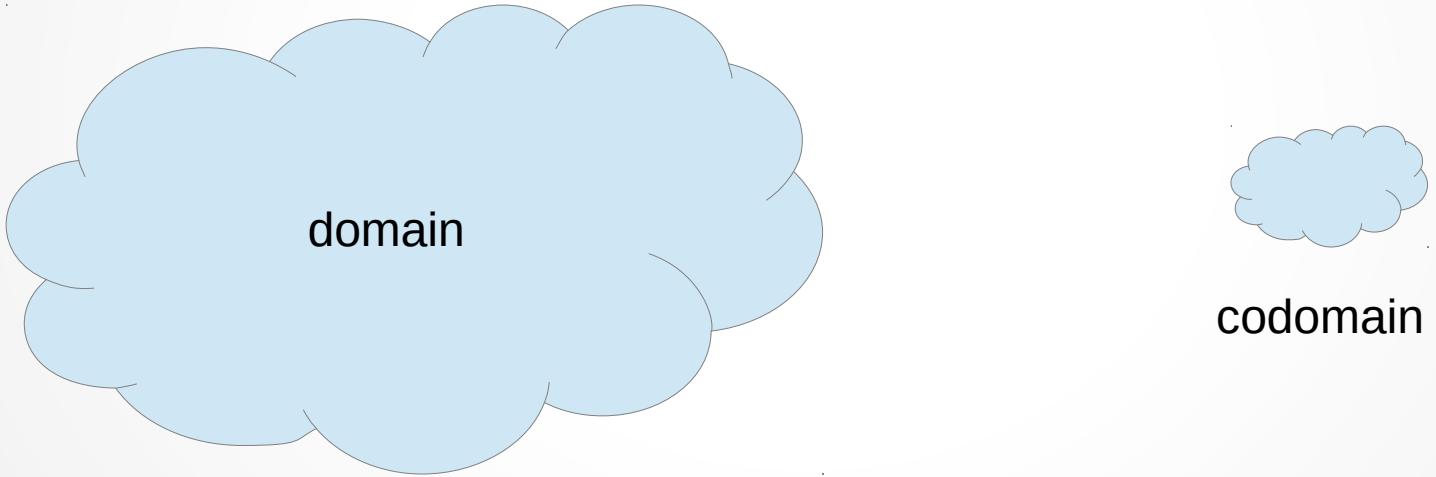
No (findable) collisions

No one can find x and y so that $x \neq y$ and $H(x) = H(y)$.



No (findable) collisions

- Surely there are collisions, because the domain of a hash function is much larger than the codomain.



An example of a collision

- H(”Kissa nukkuu.”) = 0xDABC1269
- H(”Kissa nukkuu!”) = 0x572AFFF0
- H(”Koira haukkuu!”) = 0xDABC1269

The fingerprint of files

- A file can be identified by the fingerprint given by a hash function.
- MD5(setup.exe) = 0x9b51f5c0f8132886839d806f49f9ff20
- SHA-256(setup.exe) = 0xa490014abc4c937916cd52e58a39a77131e64caa8b8bd7870b8300eb8429085c
- If a web page claims that the SHA-256 hash of setup.exe file must be the one mentioned above, but your SHA-256 check in your own computer for the file setup.exe gives a different hash, it is possible that the file is tampered! Yet another unpleasant possiblity is that the file got broken during the download (for example, the downloading process was left unfinished).

Key management

- The owning of bitcoin is implemented by digital keys, bitcoin addresses and digital signatures.
- Digital keys will not be stored on the network (usually) but in a file or database, also called the wallet, on the user's computer.
- Key management is completely detached from the Bitcoin protocol itself and keys can be created and managed without Internet connection.
- In many cases it is even recommended to create keys using a computer that is never connected to the Internet.

  Source: ”Mastering Bitcoin: Programming the Open Blockchain”

Private key

- A private key is usually a 256-bit number meaning that it contains 256 binary digits. In the Bitcoin world a private key is a random integer number between [1, n-1], where n = FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFE BAAE DCE6 AF48 A03B BFD2 5E8C D036 4140.
- Generating a private key is essentially choosing a number between [1, 2^{256}].
- The range of valid private keys is coming from the Bitcoin's standard ECDSA secp256k1.

  Source: https://en.bitcoin.it/wiki/Private_key

Private key

- With a private key one can spend bitcoin or send them forwards.
- A private key is bit like the secret PIN code of a bank account. One should never reveal it to anybody, because it is enough for stealing the money.
- Private keys are not usually stored in the computer in plaintext.
- Usually the bitcoin wallet automatically handles the private keys.
- You should become suspicious if someone requests you to give the command ”bitcoin-cli dumpprivkey 1J7mdg5rbQyUHENYdx39WVWK7fsLpEoXZy” and then send the result via e-mail ”KxFC1jmwwCoACiCAWZ3eXa96mBM6tb3TYzGmf6YwgdGWZgawvrtJ”.

How to create a Bitcoin address? (1/)

Let’s create a private key by tossing a coin 256 times. We will get a binary number:

0001100011100001010010100111101101101010001100000111111101000010011010101001010011111000000100010100011100000001111001111100100011100111011101001110011111111001101001000111111000010110000100000001101011101101100101001101000100000011000110010000101110010100.

This is in hexadecimal:
0x18E14A7B6A307F426A94F8114701E7C8E774E7F9A47E2C2035DB29A206321725.

How to create a Bitcoin address? (2/)

A public key (K) is calculated from the private key (k) by elliptic curve multiplication: K = k * G, where G is the generator point, which is defined in the secp256k1 standard, and which is the same for all the Bitcoin keys. The public key is K = (x, y), where

x = 0x50863AD64A87AE8A2FE83C1AF1A8403CB53F53E486D8511DAD8A04887E5B2352,
and y = 0x2CD470243453A299FA9E77237716103ABC11A1DF38855ED6F2EE187E9C582BA6.

Let’s concatenate these coordinates and let’s add a hex prefix of 0x04, which leads to

K =

0x0450863AD64A87AE8A2FE83C1AF1A8403CB53F53E486D8511DAD8A04887E5B23522CD470243453A299FA9E77237716103ABC11A1DF38855ED6F2EE187E9C582BA6.

How to create a Bitcoin address? (3/)

Let’s take the SHA-256 hash from the public key:
SHA256(K) = SHA256(0x0450...A6) =
0x600FFE422B4E00731A59557A5CCA46CC183944191006324A447BDB2D98D4B408.

How to create a Bitcoin address? (4/)

Let’s take the RIPEMD-160 hash from the previous SHA-256 hash: RIPEMD160(0x60...08) = 0x0109667760069 53D5567439E5E39F86A0D273BEE.

How to create a Bitcoin address? (5/)

Let’s add the version prefix (0x00, when using the Bitcoin mainnet) into the beginning of the previous RIPEMD-160 hash: 0x00010966776006953D5567439E5E39F86A0D273BEE.

How to create a Bitcoin address? (6/)

Let’s take the SHA-256 hash from the previous extended RIPEMD-160 result:
SHA256(0x00010966776006953D5567439E5E39F86A0D273BEE) =
0x445C7A8007A93D8733188288BB320A8FE2DEBD2AE1B47F0F50BC10BAE845C094.

How to create a Bitcoin address? (7/)

Let’s take the SHA-256 hash from the previous SHA-256 hash:
SHA256(0x445C7A8007A93D8733188288BB320A8FE2DEBD2AE1B47F0F50BC10BAE845C094) = 0xD61967F63C7DD183914A4AE452C9F6AD5D462CE3D277798075B107615C1A8A30.

How to create a Bitcoin address? (8/)

Let’s take the 4 first bytes or 8 first hexadecimals from the previous SHA-256 hash: 0xD61967F6.

(1 byte = 8 bits: for example 1111 1111 = FF = 2 hexas.
4 bytes = 32 bits: for example 11111111 11111111 11111111 11111111 = FF FF FF FF = 8 hexas.)

How to create a Bitcoin address? (9/)

Let’s add the four checksum bytes (0xD61967F6) from the previous step into the end of extended RIPEMD-160 hash from step 5 (0x00010966776006953D5567439E5E39F86A0D273BEE): 0x00010966776006953D5567439E5E39F86A0D273BEED61967F6.

How to create a Bitcoin address? (10/10)

Let’s encode the previous result into Base58 form:
Base58(0x00010966776006953D5567439E5E39F86A0D273B
EED61967F6) = 16UwLL9Risc3QfPqBUvKofHmBQ7wMtjvM.