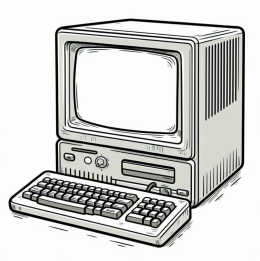
Minicurso Bitcoin

Luiz Felipe (@felipalds)
Bit Empresa Júnior

Conteúdos



O que é Bitcoin?
A história - cypherpunks
Hands-on : código-fonte (C++)

Blocos e transações
Estruturas
Hands-on : navegando pela mempool

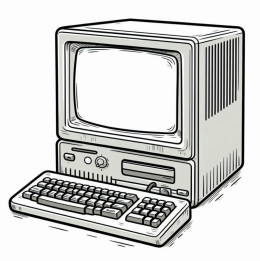
Mineração
Proof-of-Work
Hands-on : rodando um simulador

Curva elíptica ECDSA
Chaves públicas e privadas
Hands-on : gerando mnemônicos

Hands-on : gerando uma wallet
Importando na BlueWallet
Hard Wallet

Rede Lightning
Hands-on : Usando no dia a dia
Strike

Repositórios



https://github.com/Felipalds/bitcoin-minicourse

https://github.com/bitcoin/bitcoin

O que é Bitcoin?

O que é moeda?

Ouro?

Moeda fiduciária

O que é Bitcoin?

Moeda FIAT:

O que é Bitcoin?

Para que serve?

Transacionar

Reserva de valor

Proteção contra o Estado

Propriedades

Inconfiscável

Inflação controlada (halvings)

Escassez (21M)

Portabilidade

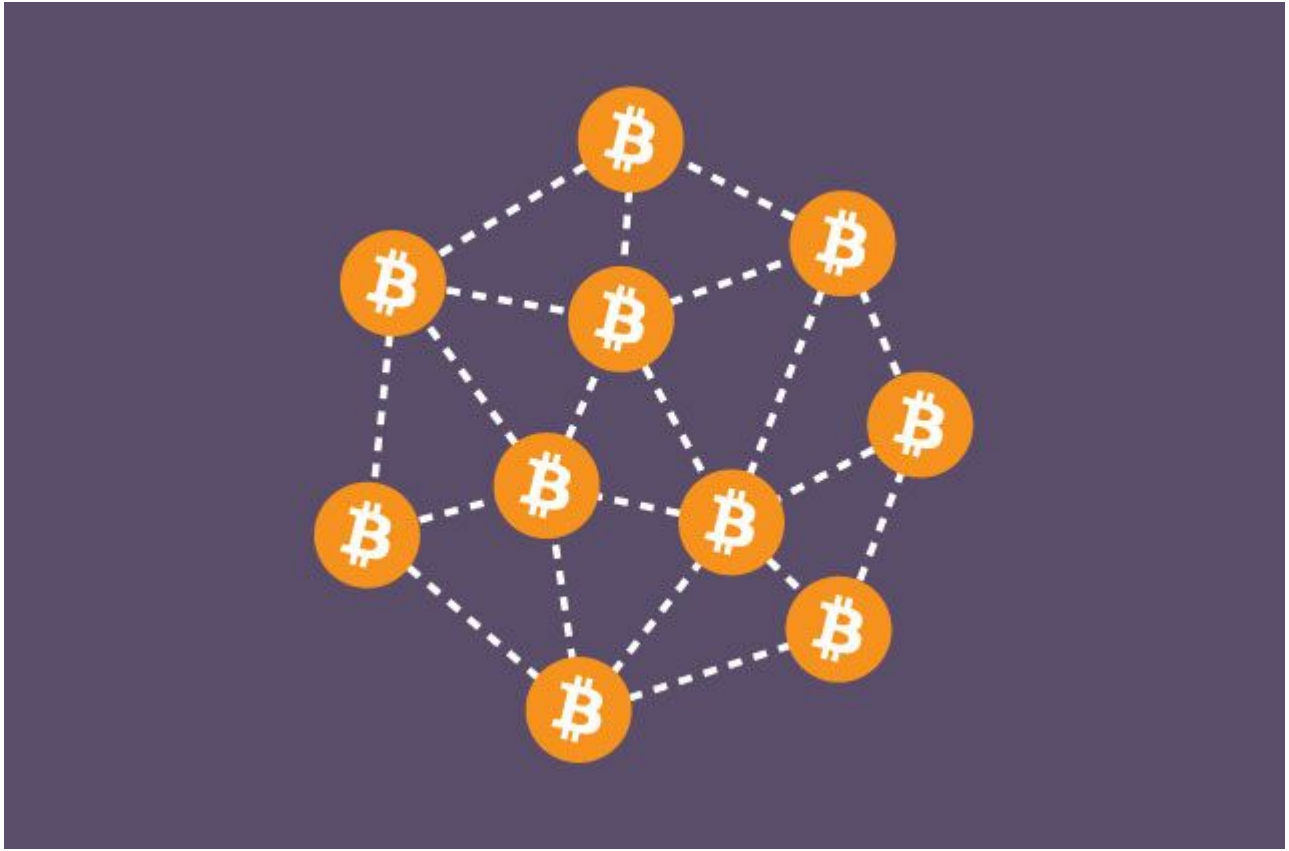
Divisibilidade (satoshis)

Verificabilidade

Resistência à censura

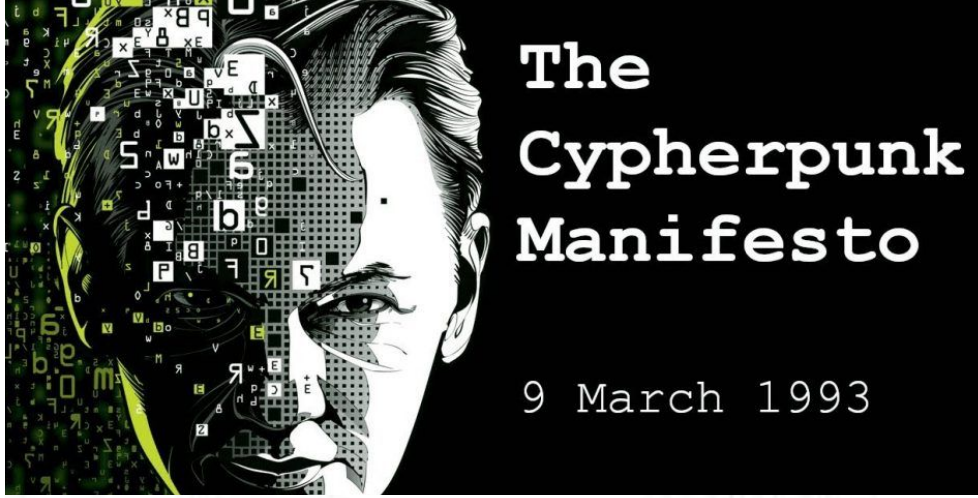
Descentralização

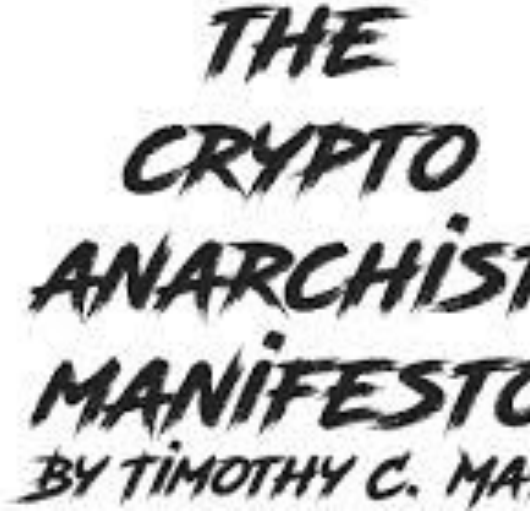
Transparência




make a gif.com


CYPHERPUNK
FUTURE IS
NOW
THE CRYPTO ANARCHIST MANIFESTO
A SPECTER IS HAUNTING THE MODERN WORLD, THE SPECTER OF CRYPTOANARCHY.
ARISE, YOU HAVE NOTHING TO LOSE BUT YOUR BARBED WIRE FENCES!


The
Cypherpunk
Manifesto
9 March 1993


THE
CRYPTO
ANARCHIST
MANIFESTO
BY TIMOTHY C. MA


@areabitcoin

https://bitcoin.org/bitcoin.pdf

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

Satoshi Nakamoto Sat, 01 Nov 2008 16:16:33 -0700
I've been working on a new electronic cash system that's fully
peer-to-peer, with no trusted third party.
The paper is available at:
http://www.bitcoin.org/bitcoin.pdf

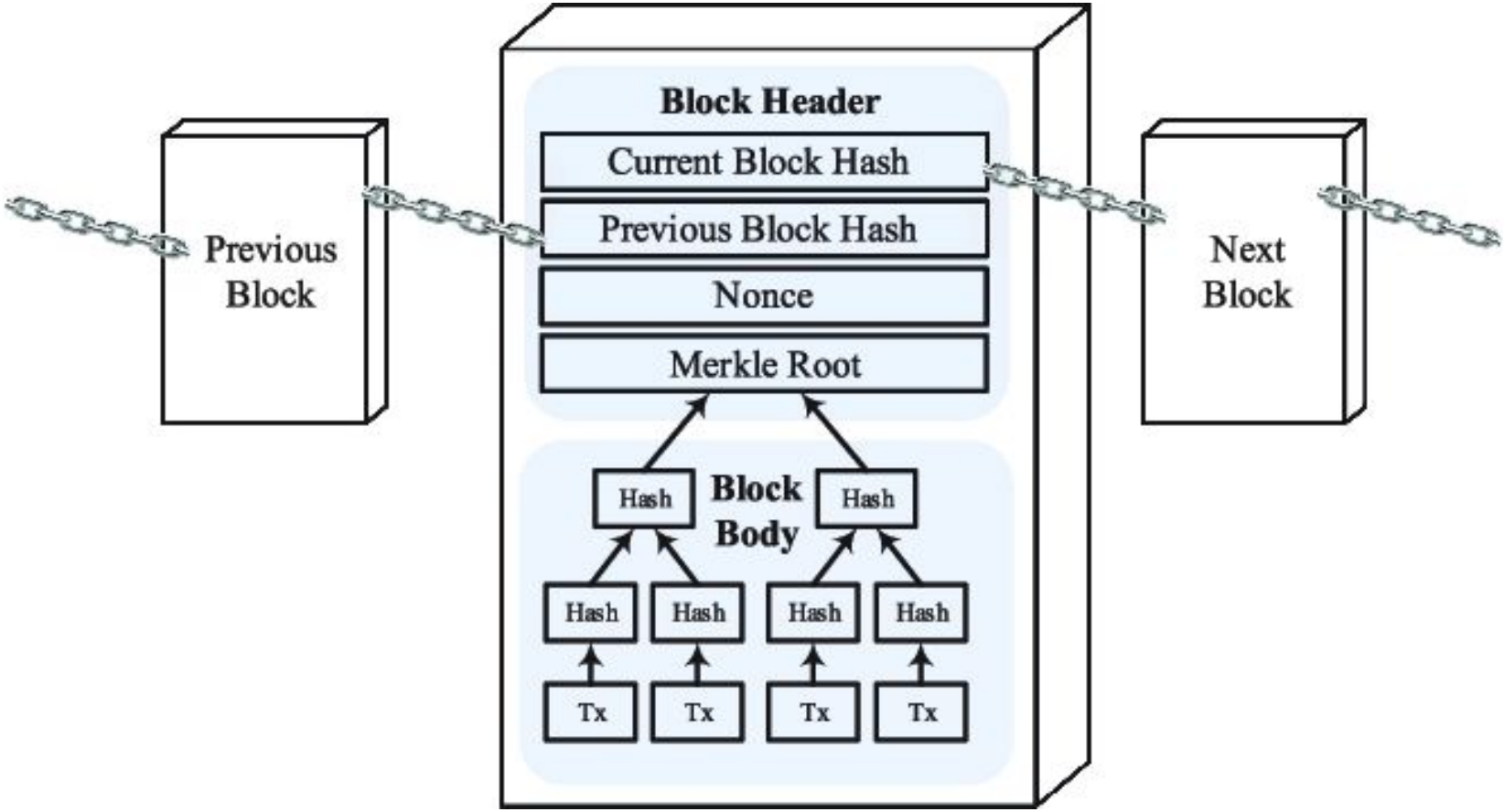
The main properties:
Double-spending is prevented with a peer-to-peer network.
No mint or other trusted parties.
Participants can be anonymous.
New coins are made from Hashcash style proof-of-work.
The proof-of-work for new coin generation also powers the
network to prevent double-spending.

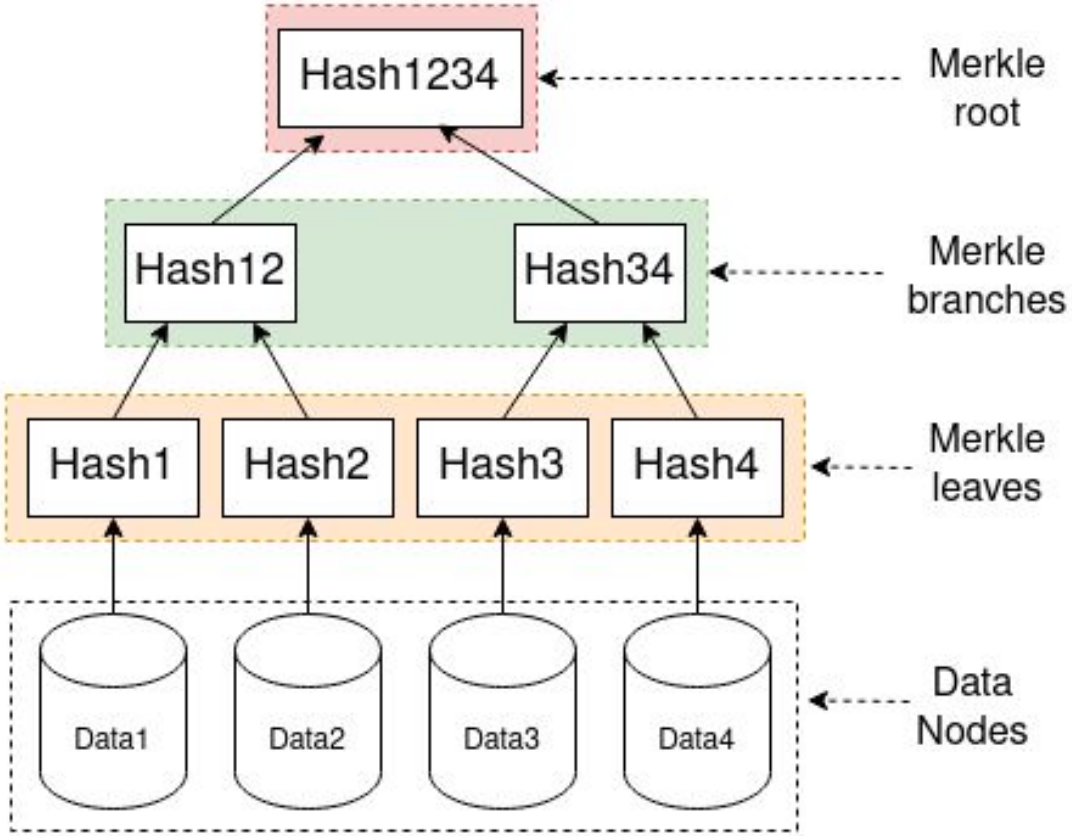
Bitcoin: A Peer-to-Peer Electronic Cash System
Abstract.
A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without the burdens of going through a financial institution. Digital signatures provide part of the solution,
but the main benefits are lost if a trusted 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 honest nodes control the most CPU power on the network, they can generate
the longest chain and outpace any attackers. The network itself requires minimal structure. Messages are broadcasted 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.
Full paper at: http://www.bitcoin.org/bitcoin.pdf
Satoshi Nakamoto
The Cryptography Mailing List

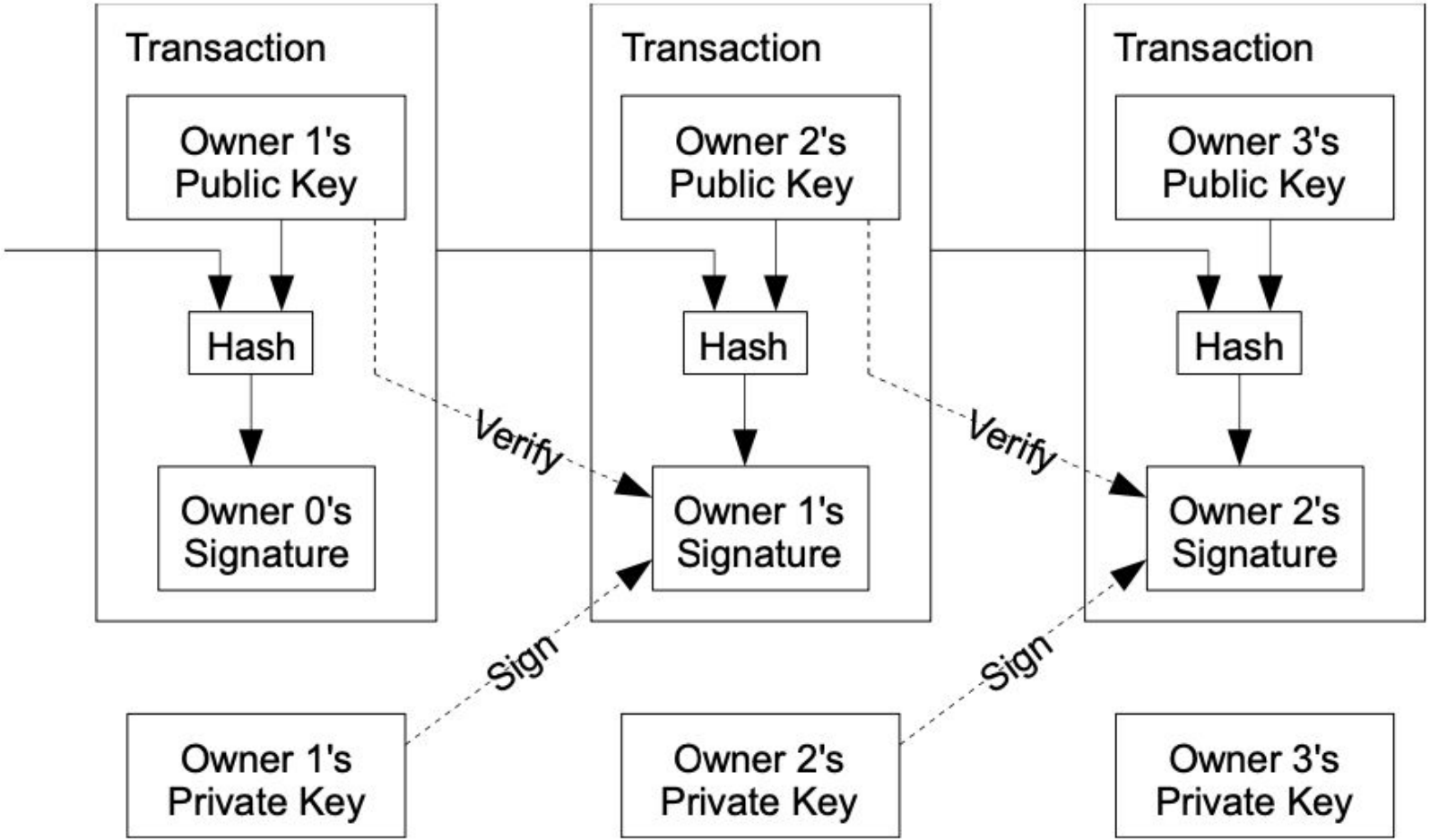
Estruturas

Bloco (blockchain)

Transações


Block Header
Current Block Hash
Previous Block Hash
Nonce
Merkle Root
Previous Block
Next Block
Hash
Block Body
Hash
Hash
Hash
Hash
Hash
Tx
Tx
Tx
Tx


Hash1234
Merkle root
Hash12
Hash34
Merkle branches
Hash1
Hash2
Hash3
Hash4
Merkle leaves
Data1
Data2
Data3
Data4
Data Nodes


Transaction
Owner 1's Public Key
Hash
Owner 0's Signature
Transaction
Owner 2's Public Key
Hash
Owner 1's Signature
Transaction
Owner 3's Public Key
Hash
Owner 2's Signature
Verify
Verify
Sign
Sign
Owner 1's Private Key
Owner 2's Private Key
Owner 3's Private Key

Alice's Wallet

Made up of 1 UTXO:

SPENT

UTXO 1: 0.4 BTC

Alice's Bitcoin Transaction

Transaction Inputs

1 INPUTS = 0.4 BTC

Transaction Outputs

3 OUTPUTS = 0.4 BTC

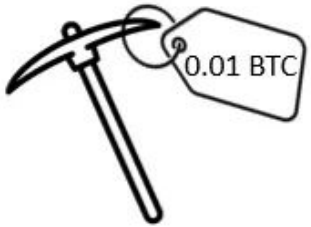
TXO 1: 0.1 BTC

TXO 2: 0.009 BTC

TXO 3: 0.291 BTC

Conference Provider

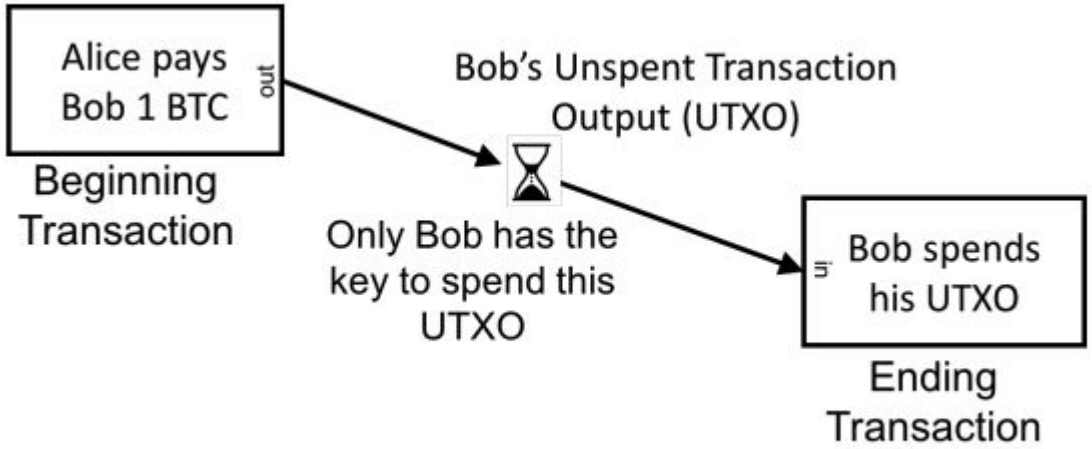
Bitcoin Mining Fee

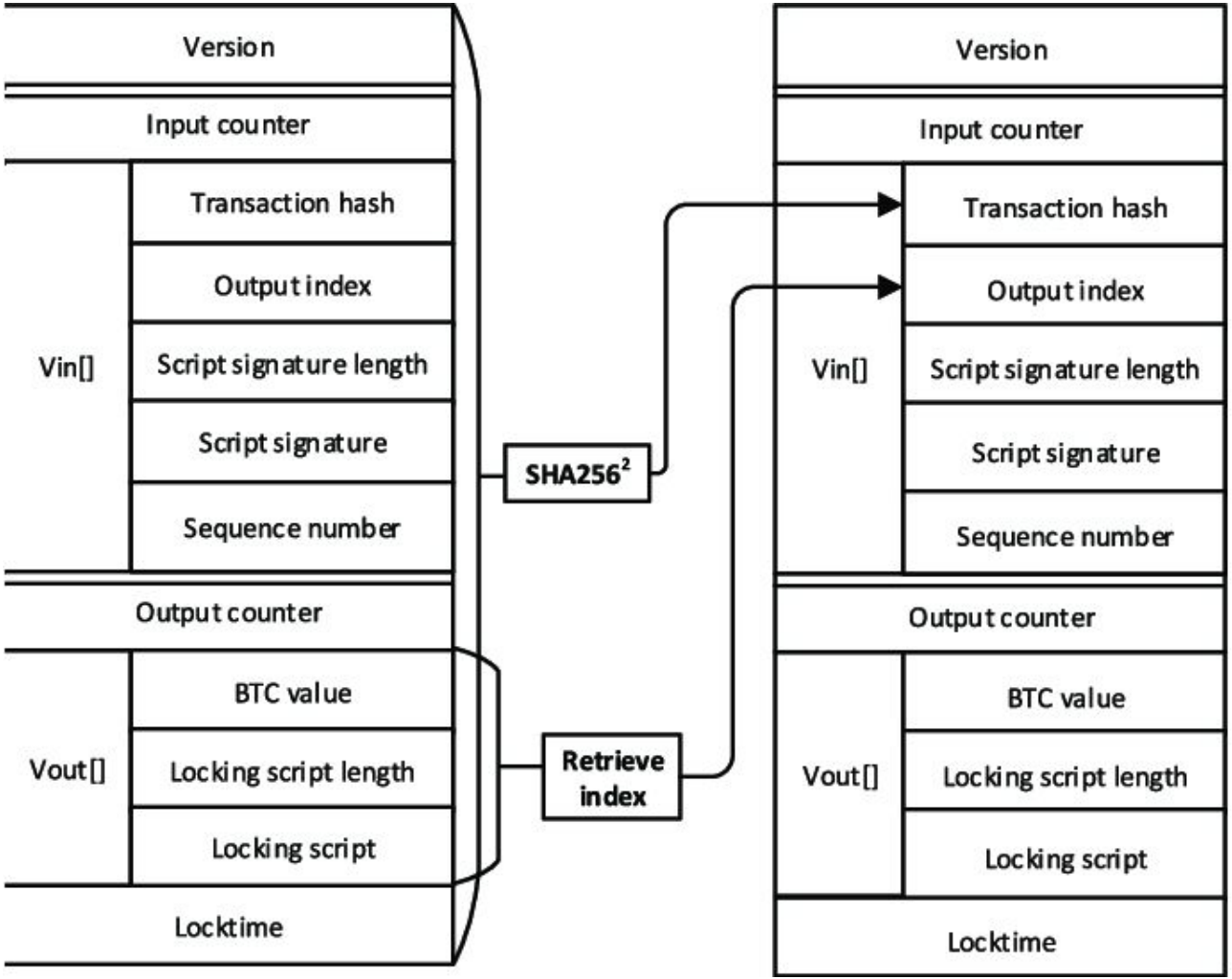


Alice's 0.291 BTC Change

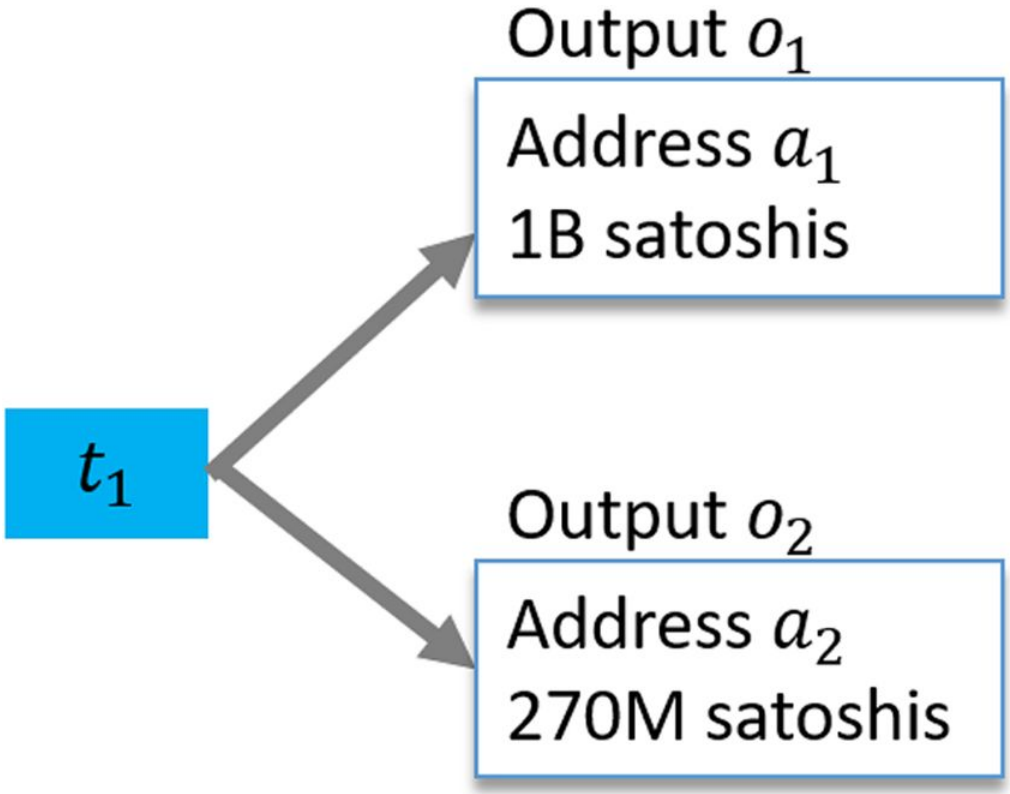
UTXO 3: 0.291 BTC

UTXO – unspent transaction output




Version
Input counter
Vin[]
Transaction hash
Output index
Script signature length
Script signature
Sequence number
Output counter
Vout[]
BTC value
Locking script length
Locking script
Locktime
SHA256²
Retrieve index
Version
Input counter
Vin[]
Transaction hash
Output index
Script signature length
Script signature
Sequence number
Output counter
Vout[]
BTC value
Locking script length
Locking script
Locktime

Coinbase transaction



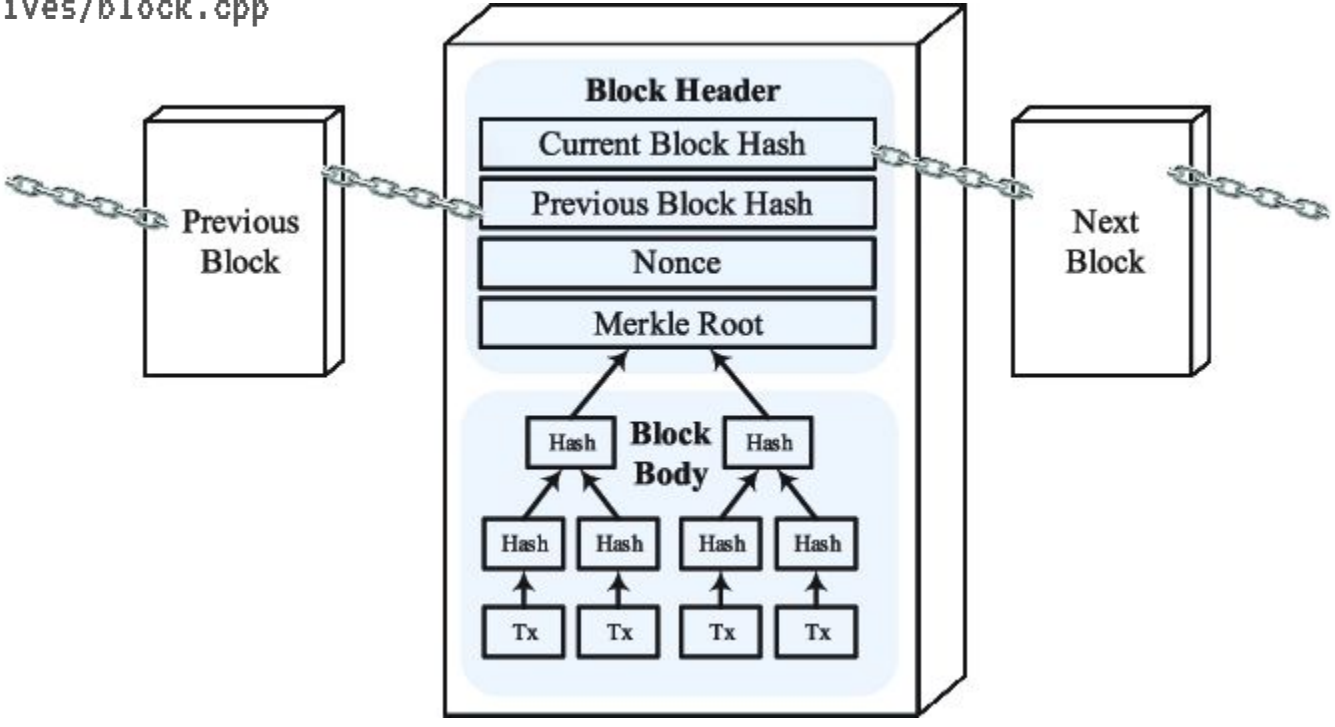
Prática 1: Código-fonte

https://github.com/bitcoin/bitcoin

Estrutura do Bloco

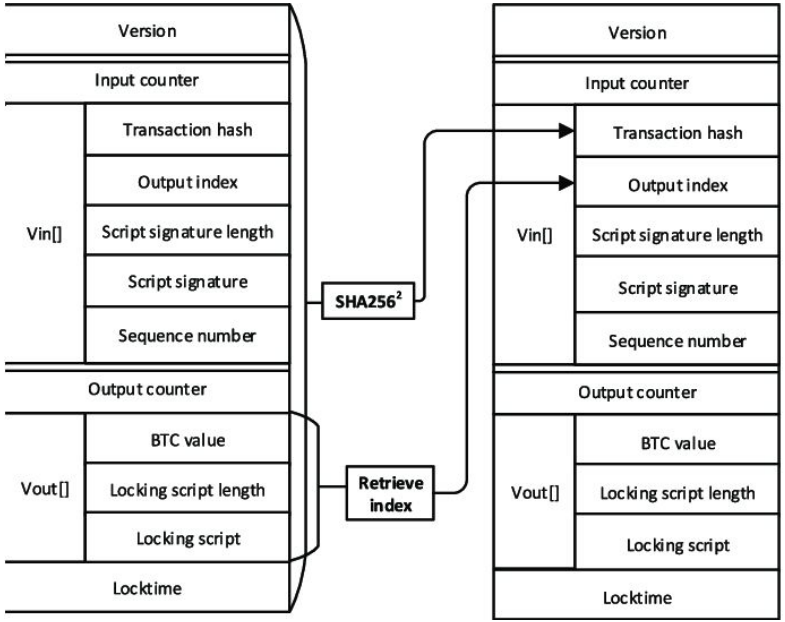
/src/primitives/block.h

/src/primitives/block.cpp



Estrutura das Transações

/src/primitives/transaction.h



Merkle Tree

/src/consensus/merkle.cpp

Propriedades interessantes

/src/consensus/amount.h

/src/validation.cpp - GetBlockSubsidy

/src/pow.cpp - CheckProofOfWork

/src/random.cpp

/wallet/wallet.cpp

Prática 2: mempool

https://mempool.space

https://mempool.space/block/000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f

scriptSig?

Mineração

Dois principais consensos: Prova de Trabalho (PoW) e Prova de Posse (PoS)

Proof of Work

Processo com altíssimo gasto computacional

Encontrar ***nonce***

Número de zeros -> HASH (SHA(256))

$$H(\mathbf{nonce} \;//\; prev \;//\; mrkl\_root \;//\; timestamp \;//\; target) < target)$$

Prova de Posse

Centralização

Poucos nós controlam a rede

Menor gasto energético

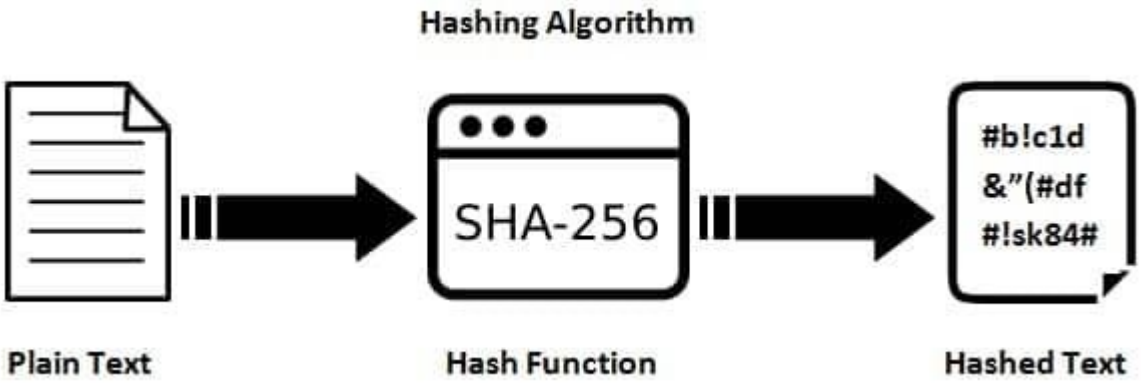
Ethereum a partir de 2022 (The Merge)

Mais usada (em mais *blockchains*)

Prática 3: minerando

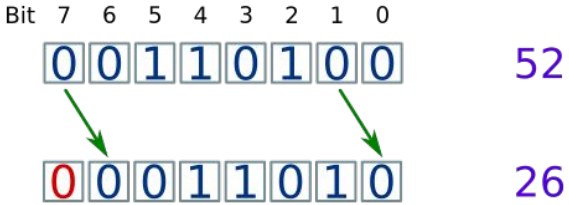
/practice/proof-of-work.py

Quanto tempo leva para minerar com simples 6 zeros?

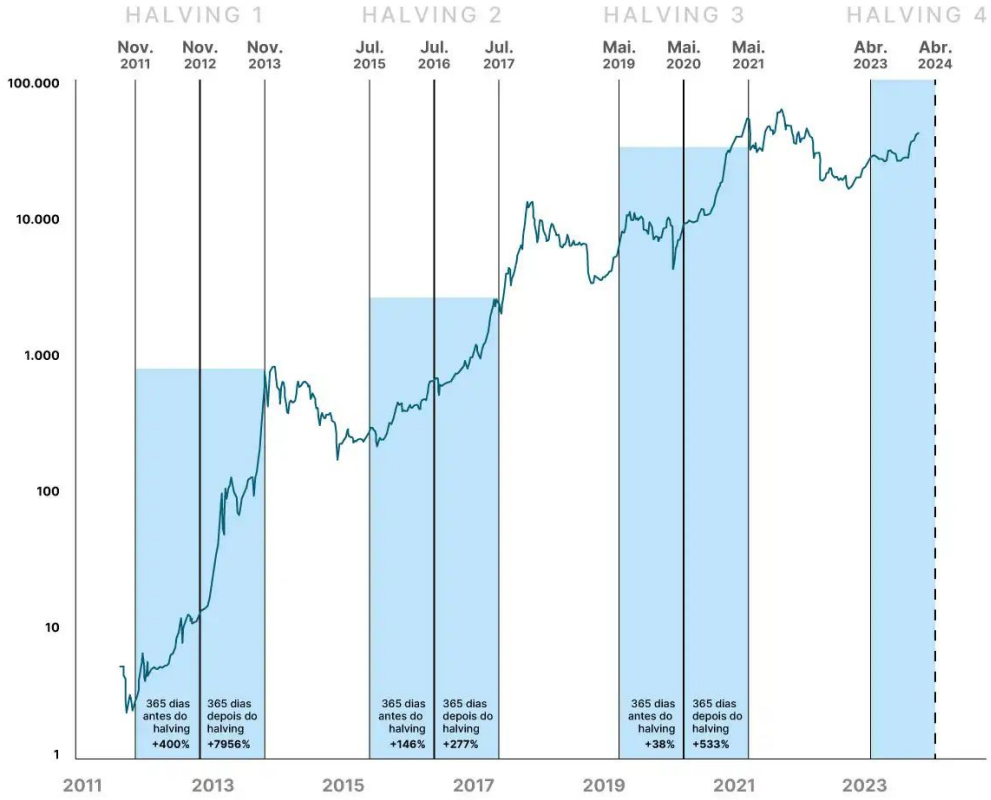


Halving

/src/validation.cpp - GetBlockSubsidy

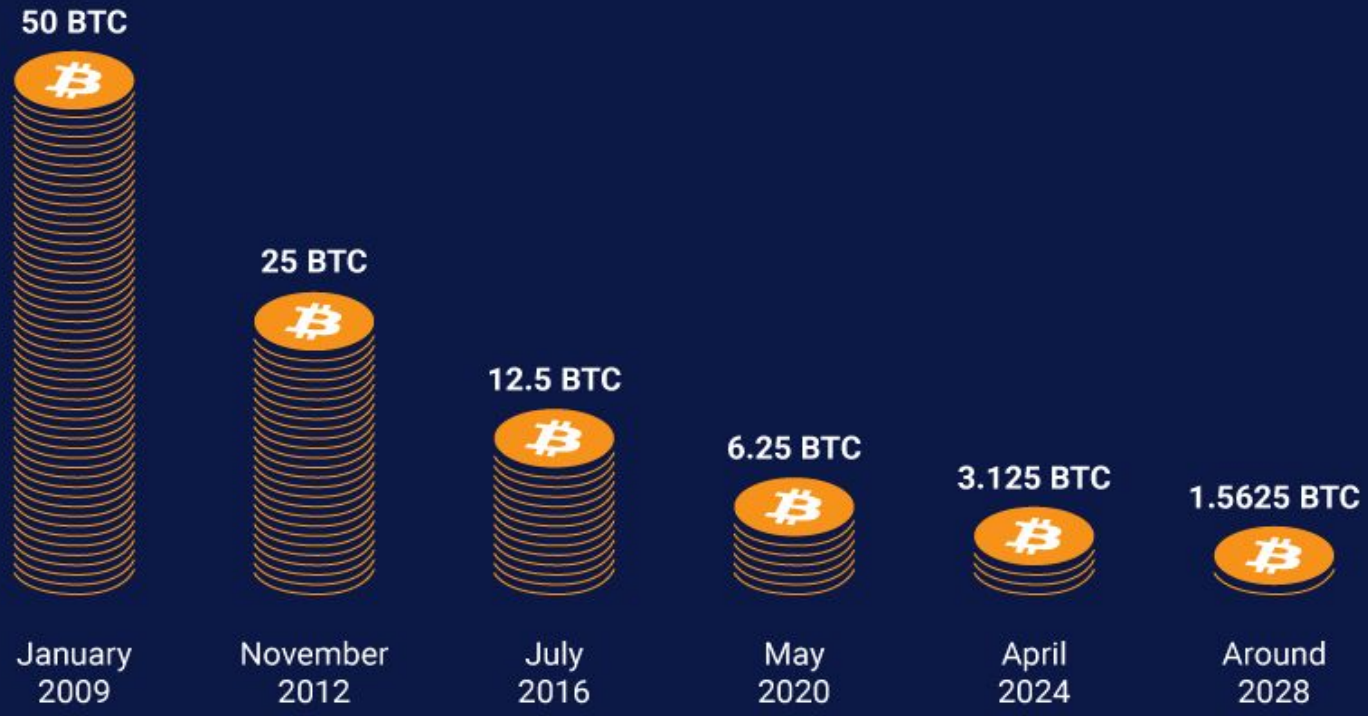


Histórico dos halvings do Bitcoin



Fonte: Ernst & Young e techopedia.com

Bitcoin halving rewards



NEXT BITCOIN *HALVINGS*

(years are estimated)

Genesis	-	Block reward 50
2012 Halving	-	Block reward 25
2016 Halving	-	Block reward 12.5
2020 Halving	-	Block reward 6.25
2024 Halving	**-**	**Block reward 3.125**
2028 Halving	-	Block reward 1.5625

WE ARE HERE

NEXT BITCOIN *HALVINGS*

(years are estimated)

Genesis	-	Block reward 50
2012 Halving	-	Block reward 25
2016 Halving	-	Block reward 12.5
2020 Halving	-	Block reward 6.25
2024 Halving	**-**	**Block reward 3.125**
2028 Halving	-	Block reward 1.5625
2032 Halving	-	Block reward 0.78125
2036 Halving	-	Block reward 0.390625
2040 Halving	-	Block reward 0.1953125
2044 Halving	-	Block reward 0.09765625
2048 Halving	-	Block reward 0.04882812
2052 Halving	-	Block reward 0.02441406
2056 Halving	-	Block reward 0.01220703
2060 Halving	-	Block reward 0.00610351
2064 Halving	-	Block reward 0.00305175
2068 Halving	-	Block reward 0.00152587
2072 Halving	-	Block reward 0.00076293
2076 Halving	-	Block reward 0.00038146
2080 Halving	-	Block reward 0.00019073
2084 Halving	-	Block reward 0.00009536
2088 Halving	-	Block reward 0.00004768
2092 Halving	-	Block reward 0.00002384
2096 Halving	-	Block reward 0.00001192
2100 Halving	-	Block reward 0.00000596
2104 Halving	-	Block reward 0.00000298
2108 Halving	-	Block reward 0.00000149
2112 Halving	-	Block reward 0.00000074
2116 Halving	-	Block reward 0.00000037
2120 Halving	-	Block reward 0.00000018
2124 Halving	-	Block reward 0.00000009
2128 Halving	-	Block reward 0.00000004
2132 Halving	-	Block reward 0.00000002
2136 Halving	-	Block reward 0.00000001

NEXT BITCOIN HALVINGS

(years are estimated)

Genesis	-	Block reward 50
2012 Halving	-	Block reward 25
2016 Halving	-	Block reward 12.5
2020 Halving	-	Block reward 6.25
2024 Halving	-	**Block reward 3.125**
2028 Halving	-	Block reward 1.5625
2032 Halving	-	Block reward 0.78125
2036 Halving	-	Block reward 0.390625
2040 Halving	-	Block reward 0.1953125
2044 Halving	-	Block reward 0.09765625
2048 Halving	-	Block reward 0.04882812
2052 Halving	-	Block reward 0.02441406
2056 Halving	-	Block reward 0.01220703
2060 Halving	-	Block reward 0.00610351
2064 Halving	-	Block reward 0.00305175
2068 Halving	-	Block reward 0.00152587
2072 Halving	-	Block reward 0.00076293
2076 Halving	-	Block reward 0.00038146
2080 Halving	-	Block reward 0.00019073
2084 Halving	-	Block reward 0.00009536
2088 Halving	-	Block reward 0.00004768
2092 Halving	-	Block reward 0.00002384
2096 Halving	-	Block reward 0.00001192
2100 Halving	-	Block reward 0.00000596
2104 Halving	-	Block reward 0.00000298
2108 Halving	-	Block reward 0.00000149
2112 Halving	-	Block reward 0.00000074
2116 Halving	-	Block reward 0.00000037
2120 Halving	-	Block reward 0.00000018
2124 Halving	-	Block reward 0.00000009
2128 Halving	-	Block reward 0.00000004
2132 Halving	-	Block reward 0.00000002
2136 Halving	-	Block reward 0.00000001

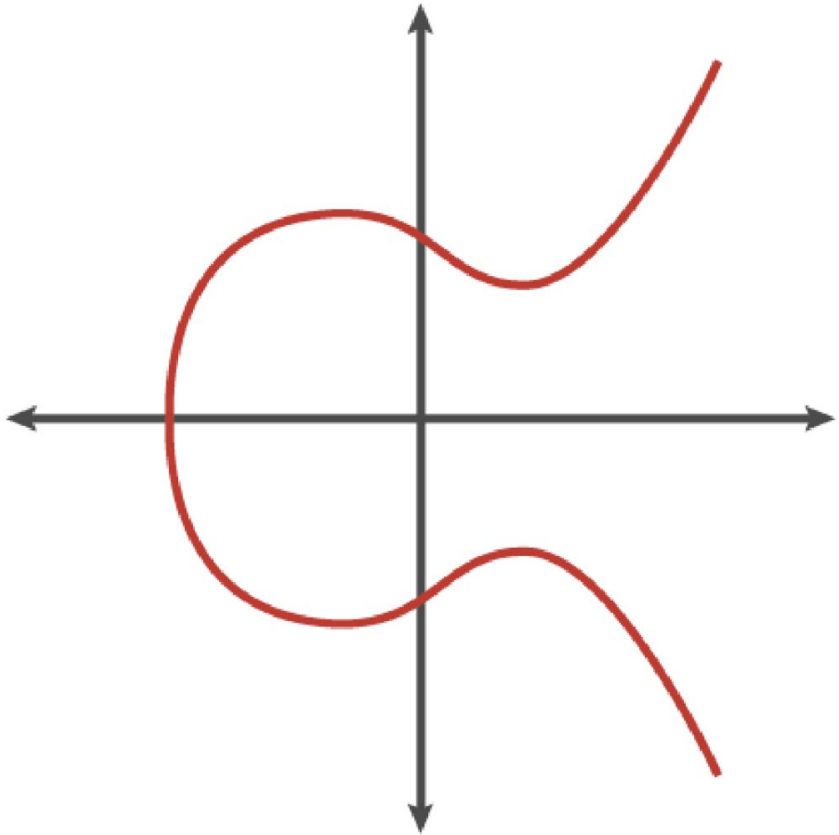
WE ARE HERE

Dúvidas?

Próxima parte:

- ECDSA
- Chaves privadas
- Gerando chaves
- Cuidando das chaves
- Hard wallet
- Importando chaves
- Lightning

Curvas elípticas ECDSA



ECDSA

secp256k1

$y^2 = x^3 + 7$ over Fp

$y^2 \bmod p = x^3 + 7 \bmod p$

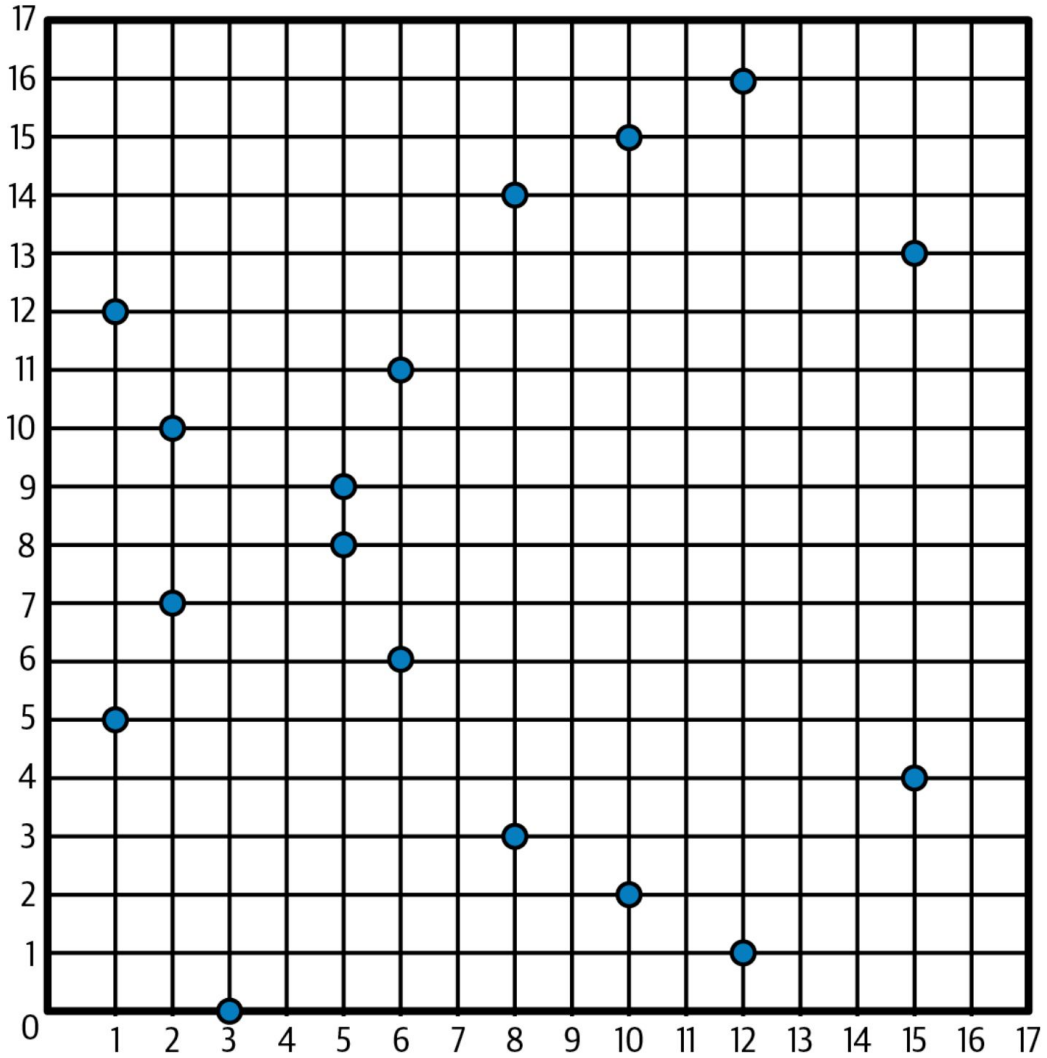
$p = Fp$, where $p = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$

ECDSA

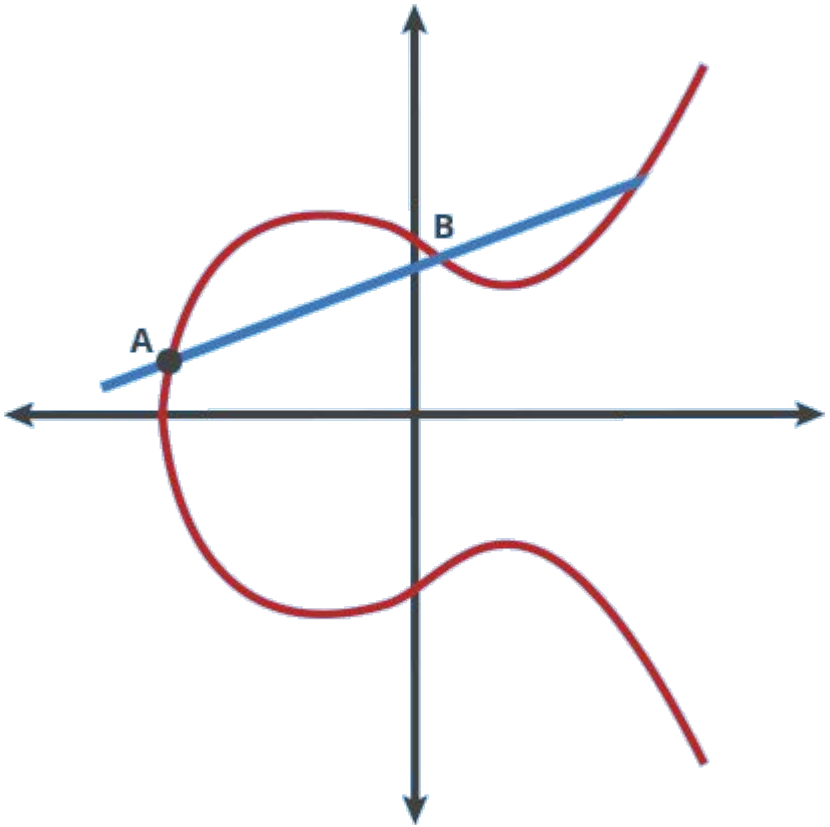
Python3

```
>>> p = 115792089237316195423570985008687907853269984665640564039457584007908834671663

>>> x = 55066263022277343669578718895168534326250603453777594175500187360389116729240

>>> y = 32670510020758816978083085130507043184471273380659243275938904335757337482424

>>> (x ** 3 + 7 - y ** 2) % p
```

ECDSA


17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

ECDSA



Chaves privadas?

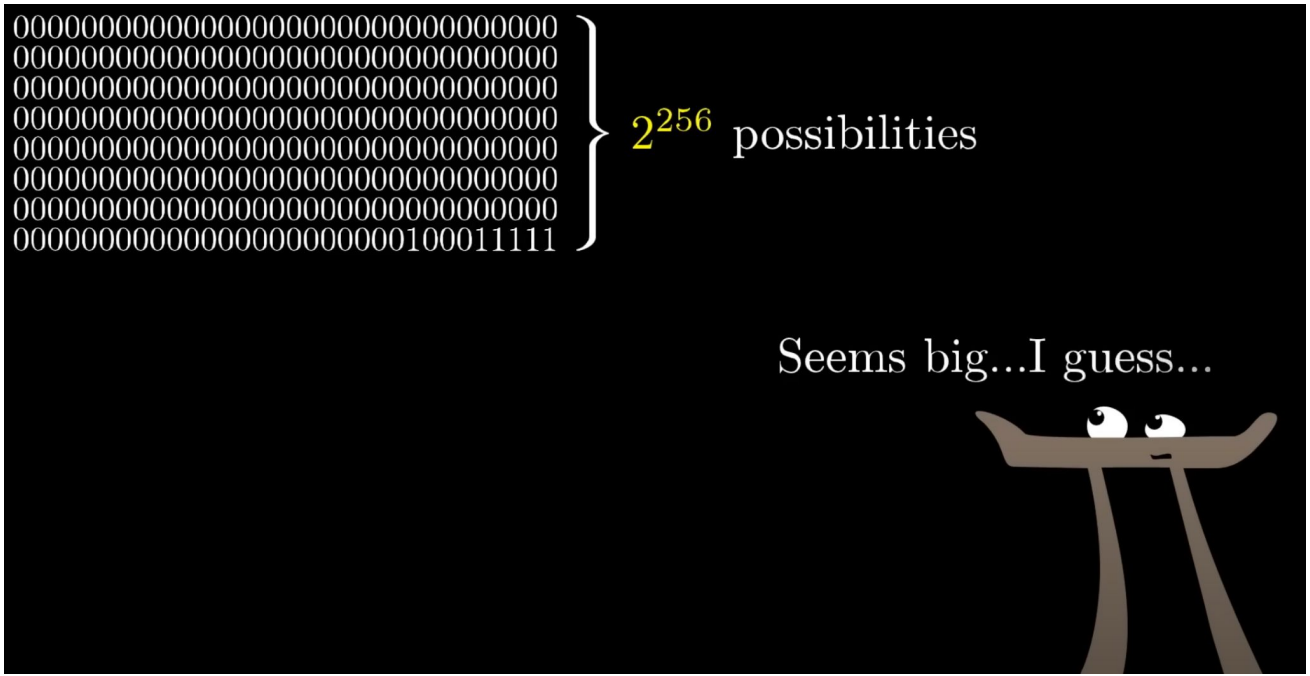
É o que você precisa armazenar de forma segura!!!!!!!

Número de 256 bits **aleatório**

https://www.youtube.com/watch?v=S9JGmA5_unY&t=231s

Chaves privadas?

O quão seguro é?



00000000000000000000000000000000 2^{32} possibilities

00000000000000000000000000000000 2^{32} possibilities

00000000000000000000000000000000 2^{32} possibilities

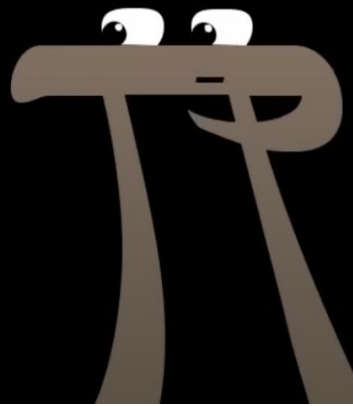
00000000000000000000000000000000 2^{32} possibilities

00000000000000000000000000000000 2^{32} possibilities

00000000000000000000000000000000 2^{32} possibilities

00000000000000000000000000000000 2^{32} possibilities

00000000000000000000001010100100 2^{32} possibilities



$$2^{32} = 4{,}294{,}967{,}296$$

(4 Billion)(4 Billion)(4 Billion)(4 Billion)(4 Billion)(4 Billion)(4 Billion)(4 Billion)

000000000000000000000000000000000

000000000000000000000000000000000

000000000000000000000000000000000

000000000000000000000000000000000

000000000000000000000000000000000

000000000000000000000000000000000

000000000000000000000000000000000

1/8

GPU boa: 1 bilhão de hashes/segundo

Bom computador: 4 bilhões de hashes/segundo

2/8

Um grande servidor: alguns milhares de computadores

Google: alguns milhões

Vamos supor então 4000 googles

Todas as pessoas são donas de 2000 servidores do Google

4 bilhões de planetas Terra

4 bilhões de galáxias

6/8

4 bilhões de segundos = 127 anos

7/8

4 bi x 127 anos = 507 bilhões de anos

37x a idade do universo

Mesmo se todas 8 bilhões de pessoas

Tivessem 500 servidores do Google

Com ultra GPUs

Em 4 bilhões de planetas diferentes

Em 4 bilhões de galáxias diferentes

Processando durante toda a idade do universo (37 vezes, hein!)

Ainda teriam 1/4000000000 de chances

De sucesso em uma tentativa

$$\frac{(4\text{ Billion})\text{H/s}}{(4\text{ Billion})\text{KG++}}\,\frac{(4\text{ Billion})\text{KG++}}{(4\text{ Billion})}\,\frac{(4\text{ Billion})}{(4\text{ Billion})\text{GGSC}}\,\frac{(\quad)(\quad)}{(\quad)}$$






1 in 4 Billion chance of success

4 Billion seconds $\approx$ 126.8 years

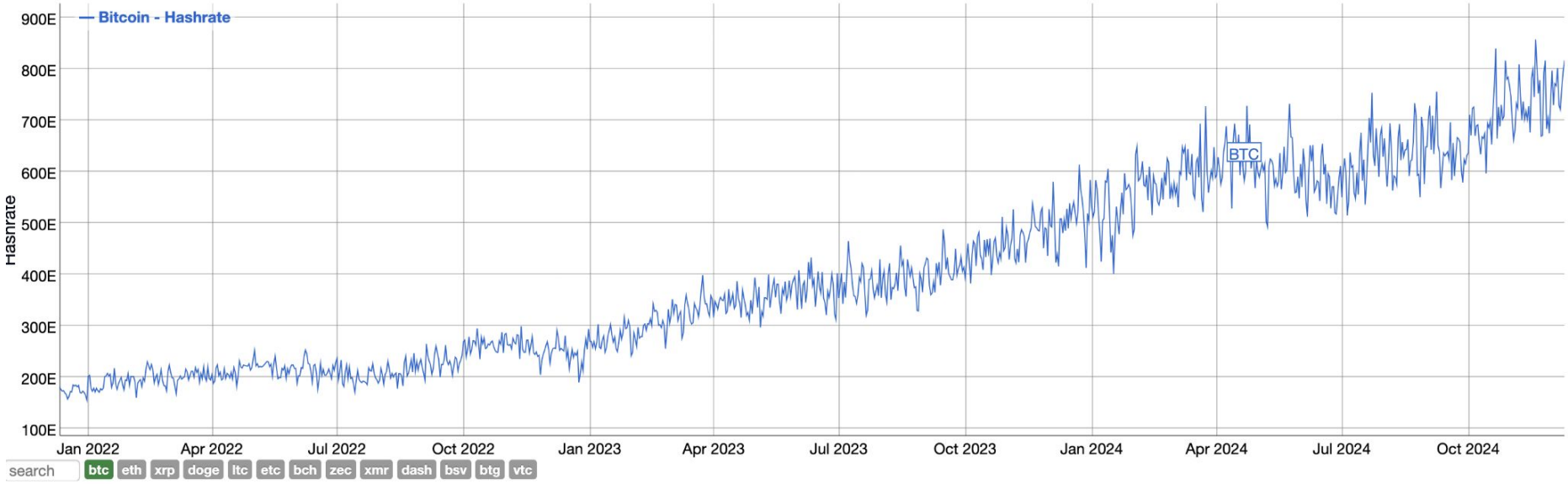
4 Billion $\times$ 126.8 years $\approx$ 507 Billion years

$\approx 37\times$ Age of universe

Bitcoin Hashrate historical chart

Average hashrate (hash/s) per day | 794.39 Ehash/s -3.17% in 24 hours

Share:



search btc eth xrp doge ltc etc bch zec xmr dash bsv btg vtc

Scale: linear log

Latest Prices: BTC/USDT: 97320.71 (binance) | BTC/USD: 97271.65 (coinbase) | BTC/USD: 97415.79 (p2pb2b) | BTC/USD: 97439 (bitstamp)

Zoom: 3 months 6 months 1 year 3 years all time

900 quintilhões de hashes por segundo (9^{18})

9 bilhões de bilhões

9 pessoas possuem um superservidor do google (das 8 bilhões)

Prática 4: mnemônicos em Python ;)

/src/mnemonics.py

BIP 39 - 2013

https://github.com/bitcoin/bips/blob/master/bip-0039/english.txt

Palavrinhas mágicas

Reservatório de **entropia**

Prática 4: mnemônicos em Python ;)

Cada palavra: 11 bits

2^{11} = 2048 palavras

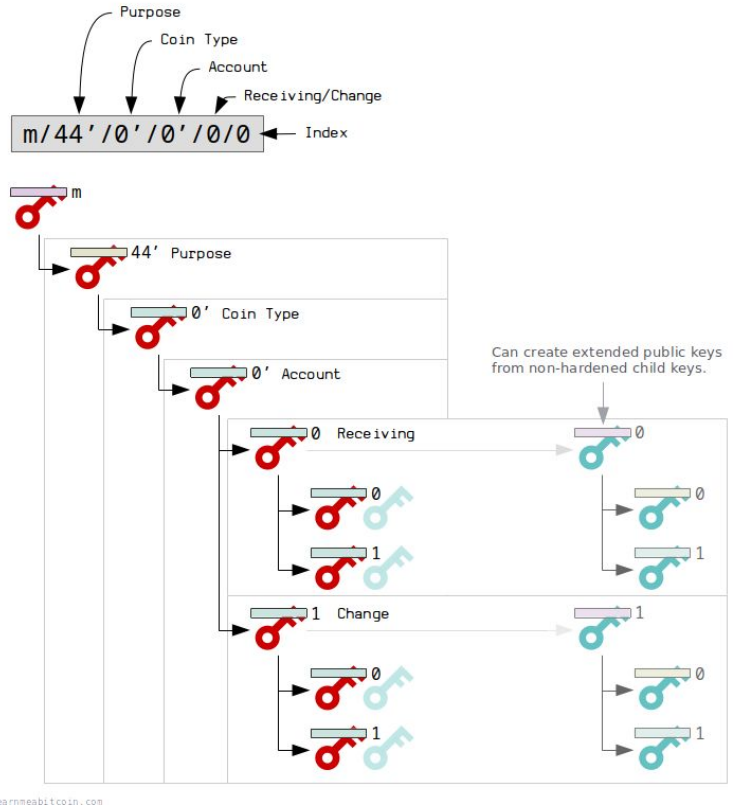
12 palavrinhas: 2^{132} possibilidades (em bits)

24 palavrinhas: 2^{256} possibilidades (em bits) ->>> TAMANHO DA CHAVE!

Prática 4: mnemônicos em Python ;)

As palavras **NÃO** são sua chave privada

Elas são a **seed** de **entropia**



Prática 5: gerando uma wallet de verdade - do jeito certo!

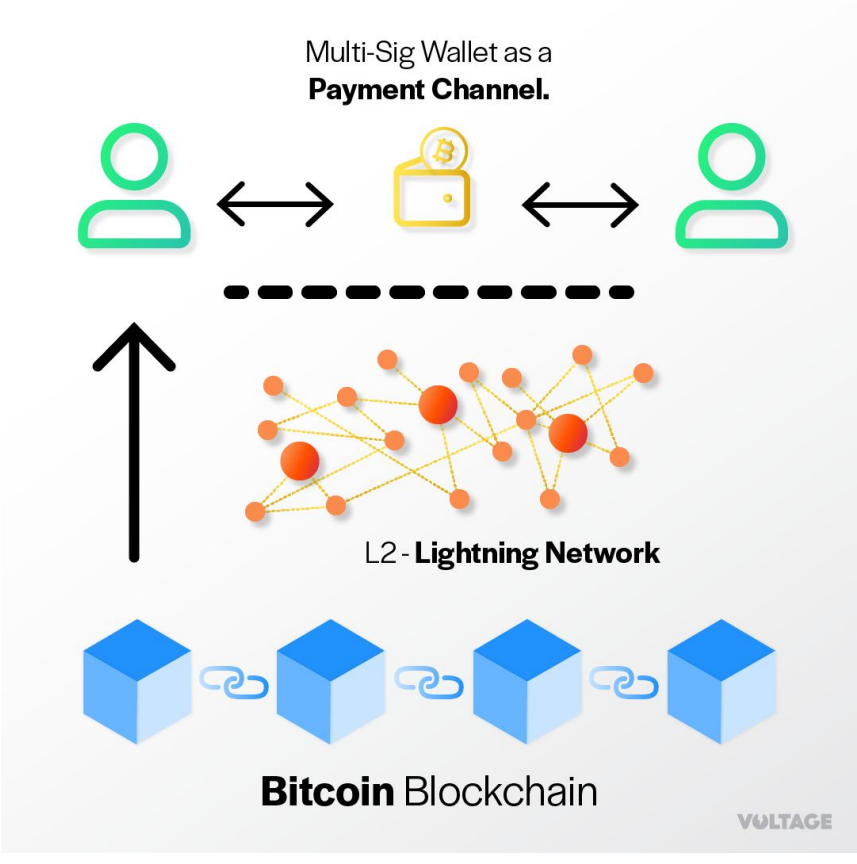
Ian Colleman

https://www.iancoleman.net/bip39/

Electrum - https://electrum.org/#download

BlueWallet - smartphone

Prática 6: Lightning



Prática 6: Lightning

Strike

Wallet of Satoshi

Agradecimentos

- Unioeste - Bit Empresa Júnior
- RNP - Rede Nacional de Pesquisa
- Scalar School - Human Rights Foundation
- Vinteum
- **Muito obrigado, vocês fizeram parte disso!**

:~$ Scalar School


@cryptomkg

Dúvidas?

Criptografia quântica

Shitcoins

Impostos

?