

Gymnázium Zlín



Bitcoin basics

<https://crocs.fi.muni.cz/papers/btc>



Petr Švenda  svenda@fi.muni.cz  [@rngsec](https://twitter.com/rngsec)

with help from Antonín Dufka, Lukasz Chmielewski

Centre for Research on Cryptography and Security, Masaryk University



WHY BITCOIN?

Especially if you are not interested in Bitcoin.

“Bitcoin fixes everything!”



fixes this

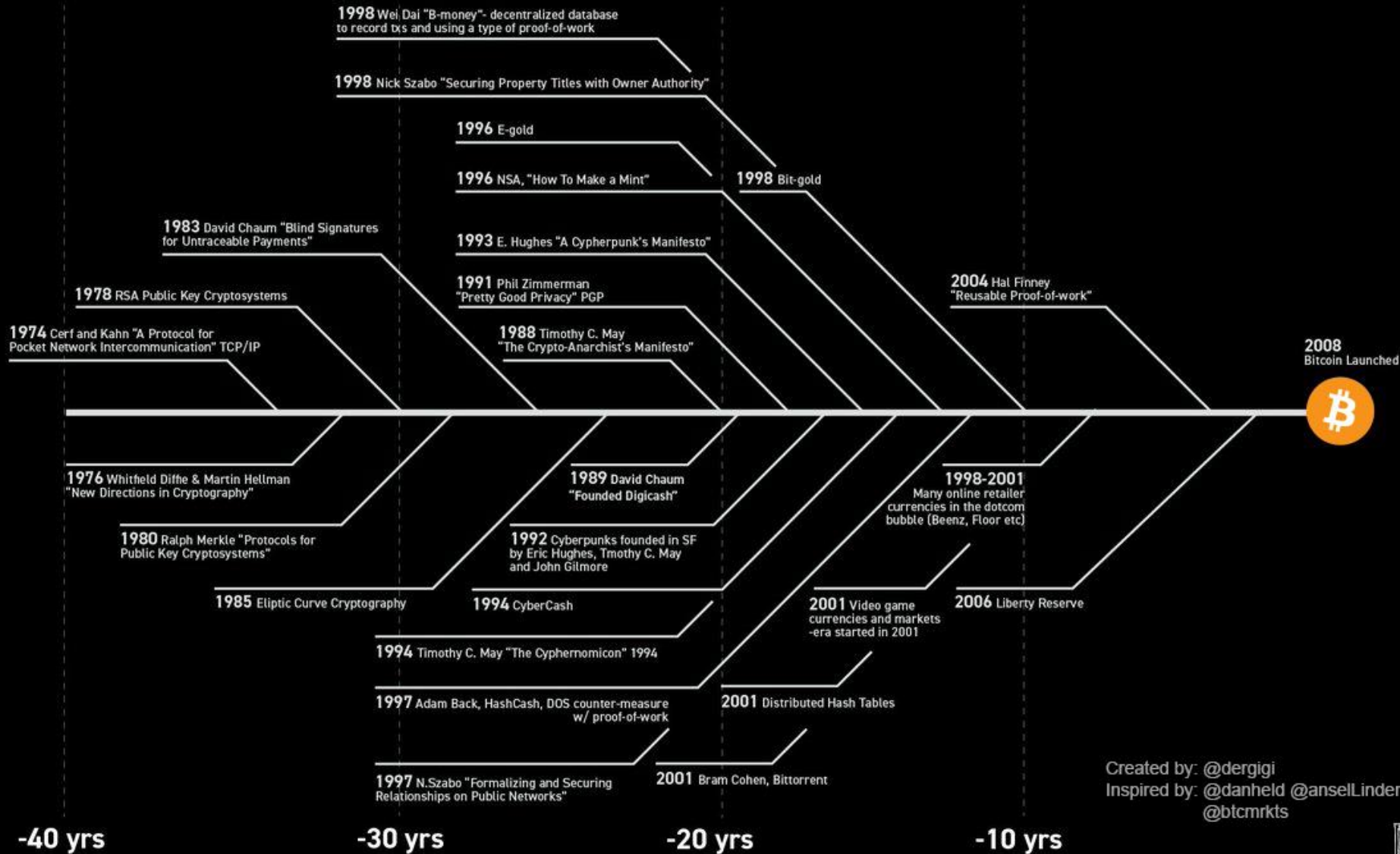
Important questions we will NOT cover:
 Lighting network, mining enviro impact,
 OP_RETURN, price volatility, altcoins tech...
 – great topics for chat afterwards! 🍺

Goals for this tutorial

- Bitcoin does not fix everything, but is on a frontline
 - No safety net, no chargeback, attacker anonymous => security technique must really work, great for battle-testing security ideas, natural “bug bounty program”
- 6 main tech pieces we will cover (also usable outside Bitcoin world)
 1. How to backup key(s) (single seed, BIP39, Shamir)
 2. ~~How to make always fresh keys (derivation via BIP32, also address privacy)~~
 3. ~~How to protect signing key against malware~~
 - (multisig, hardware wallet, airgap pc + tx b
 4. ~~How to introduce restricted signing policy (ti~~
 5. ~~How to protect your financial privacy (CoinJo~~
 6. ~~How to use hardware wallet with secure element~~

If interested in more details about
 Bitcoin usage tutorial, visit
<https://crocs.fi.muni.cz/papers/btc>

Bitcoin prehistory - It's the result of 40 years of research, development and demand



Created by: @dergigi
 Inspired by: @danheld @anselLinder @btcmrkt



Overview

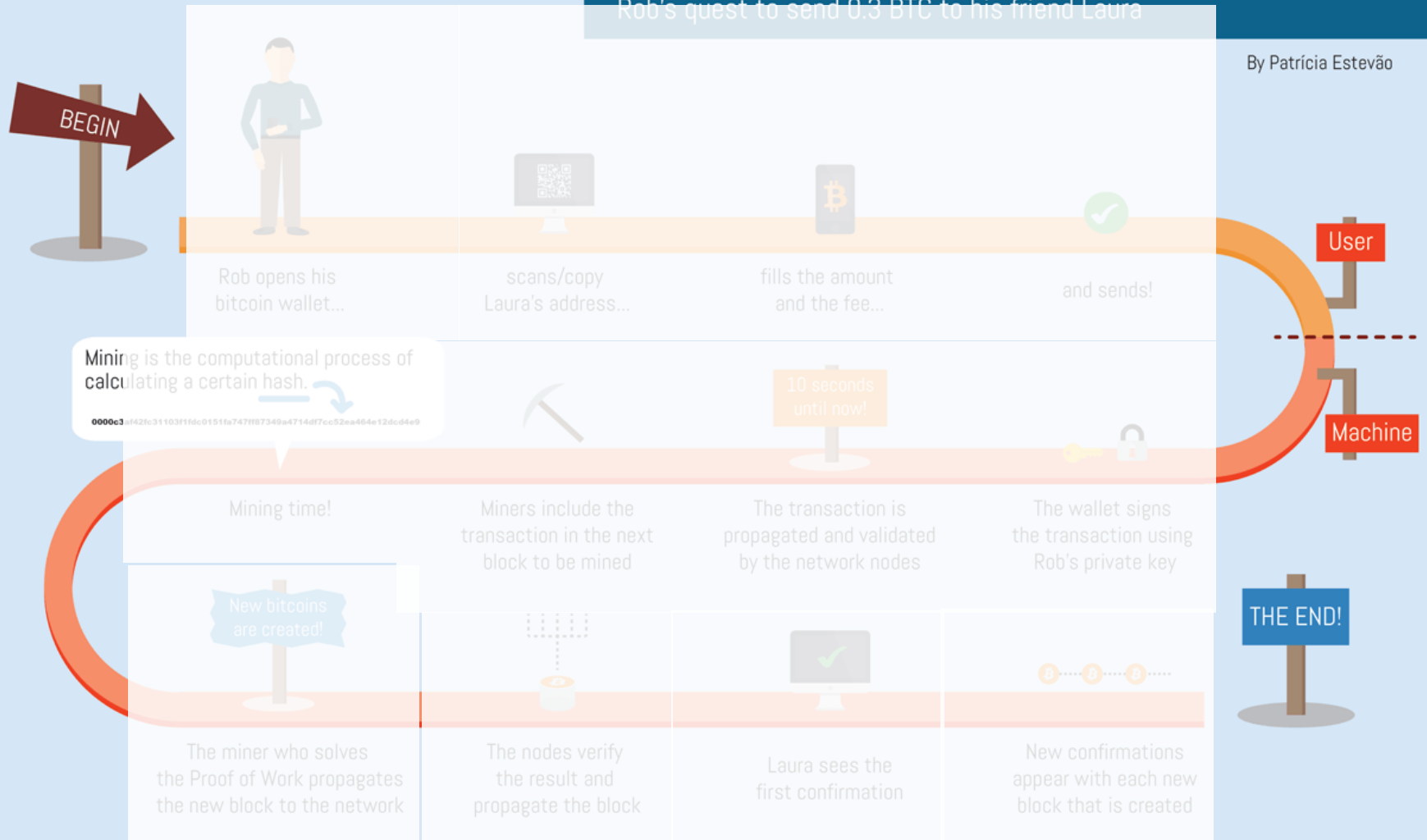
1. Using Bitcoin Core full node (mainnet)
 - Start downloading blocks, investigate connected peers, network
2. Using Bitcoin Core full node locally (regtest)
 - cli, mining, sending, transactions
3. Group discussions – basic Bitcoin questions
4. Getting and sending some (testnet) bitcoins using SparrowWallet
5. Getting some real sats

BASICS

THE BITCOIN TRANSACTION LIFE CYCLE

Rob's quest to send 0.3 BTC to his friend Laura

By Patrícia Estevão



- Wallet
- Address
- Fee
- Transaction
- Signing
- Network nodes
- Block
- Mining
- Proof of Work
- Verification
- Block reward
- Tx confirmation
- And many more...

Main design goals of the Bitcoin

1. Decentralization

- No central authority or intermediary (=> no single point of failure), possibility of self-custody
- No limitation on network participants (no permission to join is required)
- Applies to executing a transaction, but also development, infrastructure, mining...

2. Transparency

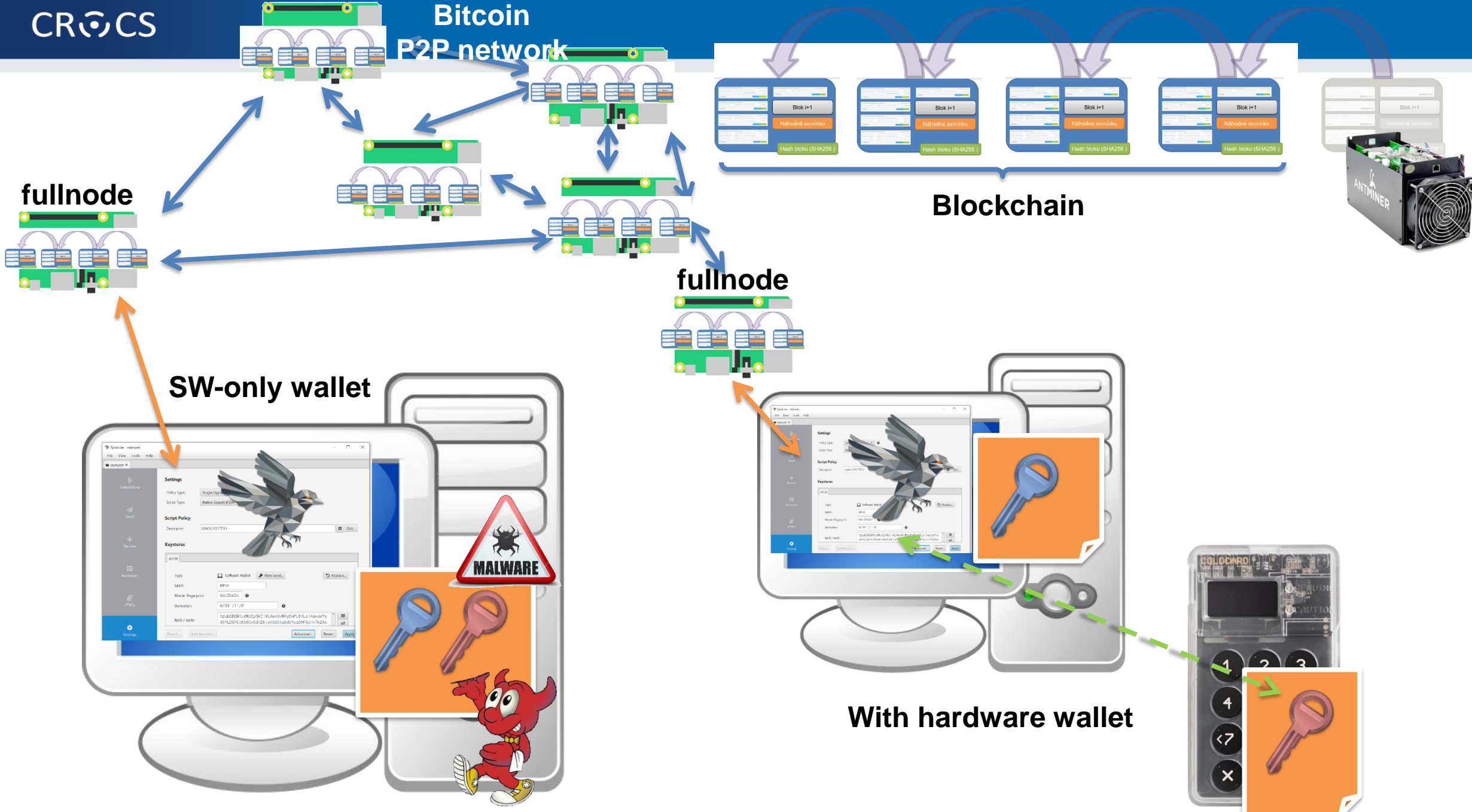
- All transactions recorded on public ledger; validity of every “bitcoin” easy to verify
- Total number of bitcoins in circulation easy to assess (monetary policy, fixed supply)

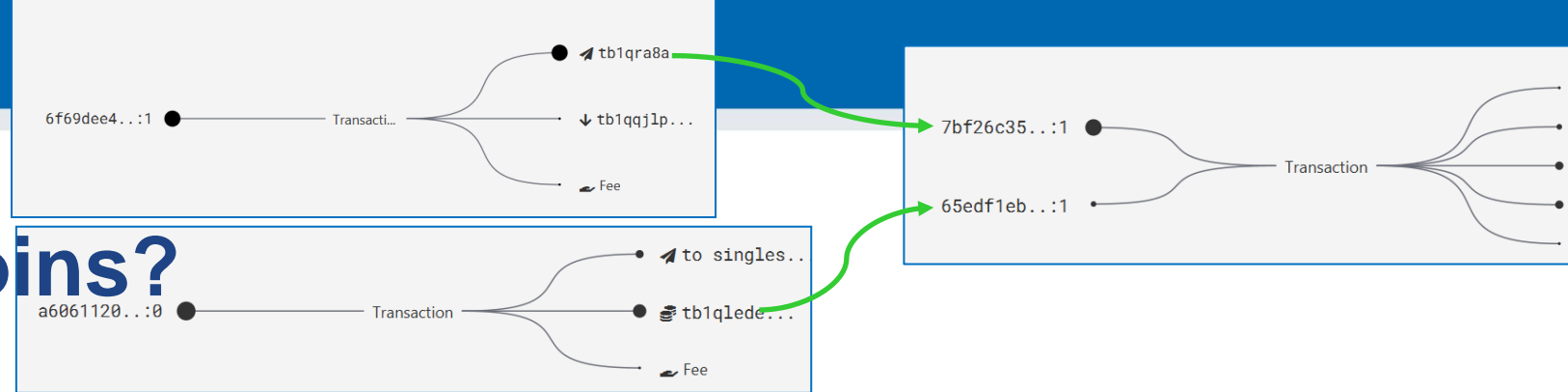
3. Security based on cryptography (mainly signature, hash functions)

- Ownership of bitcoins proved only cryptographically (no “chargeback” based on human decision)
- Protection of bitcoins reduced to protection of private key(s)

4. Pseudonymity of participants

- bitcoins connected to public keys, not usernames (does not automatically mean anonymity!)





Where are my bitcoins?

- Public ledger of all transactions (blockchain)
 - Propagated between Bitcoin fullnodes (P2P network)
- “Bitcoin holdings” - sum of values of not-yet-spent transactions control
 - Unspent Transaction Output (UTXO)
- “Bitcoin send” – take “your” UTXO and use it as input to new one
 - Specify recipient by script specifying what must be done in future send (lockscript)
 - Typical lockscript is “prove that you can sign with private key corresponding to THIS public key”
- “Bitcoin receive” – generate variable part of lockscript (public) and share with sender + monitor blockchain for my transaction
- Protection and handling of private keys is paramount
 - “Not your keys, not your bitcoin! “

UTXO set = all currently valid “bitcoins”



<https://statoshi.info/d/000000009/unspent-transaction-output-set?orgId=1&refresh=10m&from=1483225200000&to=now>

Networks in Bitcoin (Mainnet, Testnet, Regtest)

- **Mainnet** – main. global production network
- **Testnet** – testing network (global, some mining happens...)
 - Restarted from time to time, contains many different types and versions of TXs
- **Regtest** – local instance of Bitcoin network
 - Used for local testing (integration, regression, debugging)
 - Blockchain started from block 0, you are the only miner
 - (mined bitcoins unusable on Mainnet)
 - You can insert own transactions, decide on mining new blocks, debug...
- **Lightning** – second layer network of payment channels atop of mainnet
 - Practically instant and very low fees independently from mainnet

P2P Bitcoin network map <https://bitnodes.io/>

REACHABLE BITCOIN NODES

Updated: Sat Mar 18 10:21:17 2023 CET

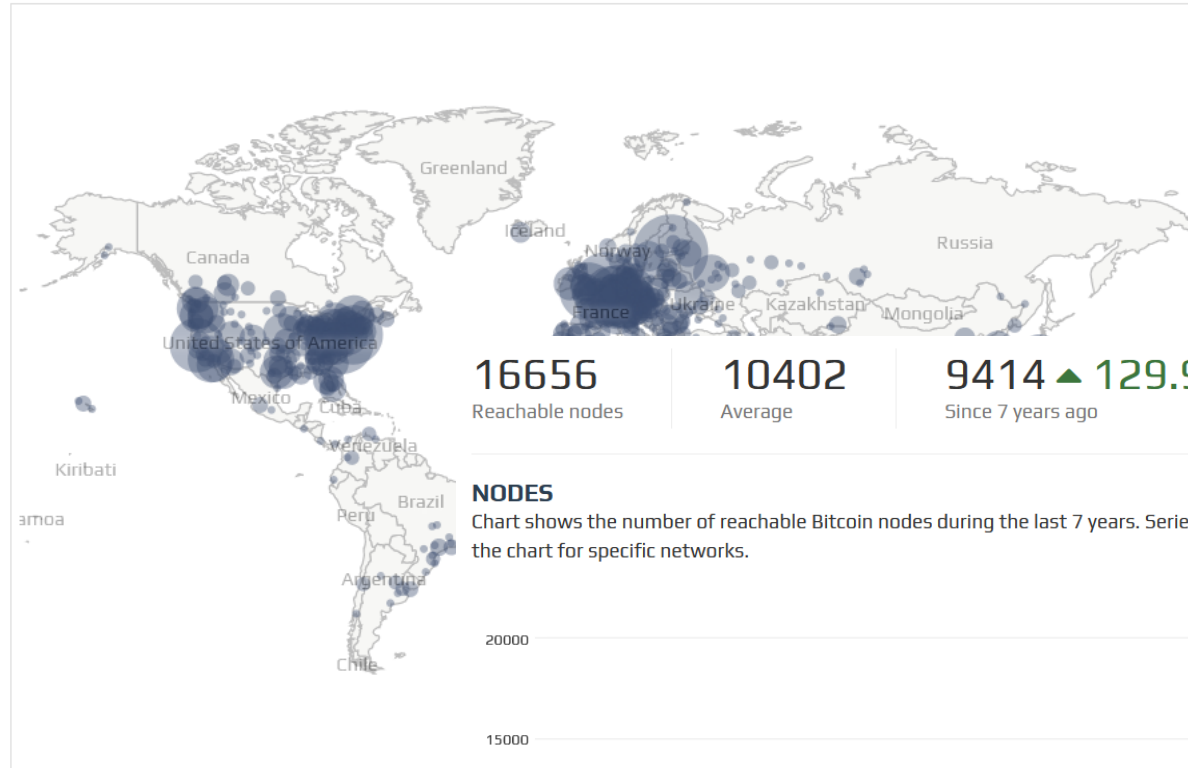
16537 NODES

CHARTS

IPv4: +0.1% / IPv6: +0.6% / .onion: +21.8%

Top 10 countries with their respective number of reachable nodes are as follows.

RANK	COUNTRY	NODES
1	n/a	9992 (60.42%)
2	United States	1752 (10.59%)
3	Germany	1403 (8.48%)
4	France	448 (2.71%)
5	Netherlands	398 (2.41%)
6	Canada	273 (1.65%)
7	Finland	240 (1.45%)
8	United Kingdom	211 (1.28%)
9	Russian Federation	169 (1.02%)



Map shows concentration of reachable

16656

Reachable nodes

10402

Average

9414 ▲ 129.99%

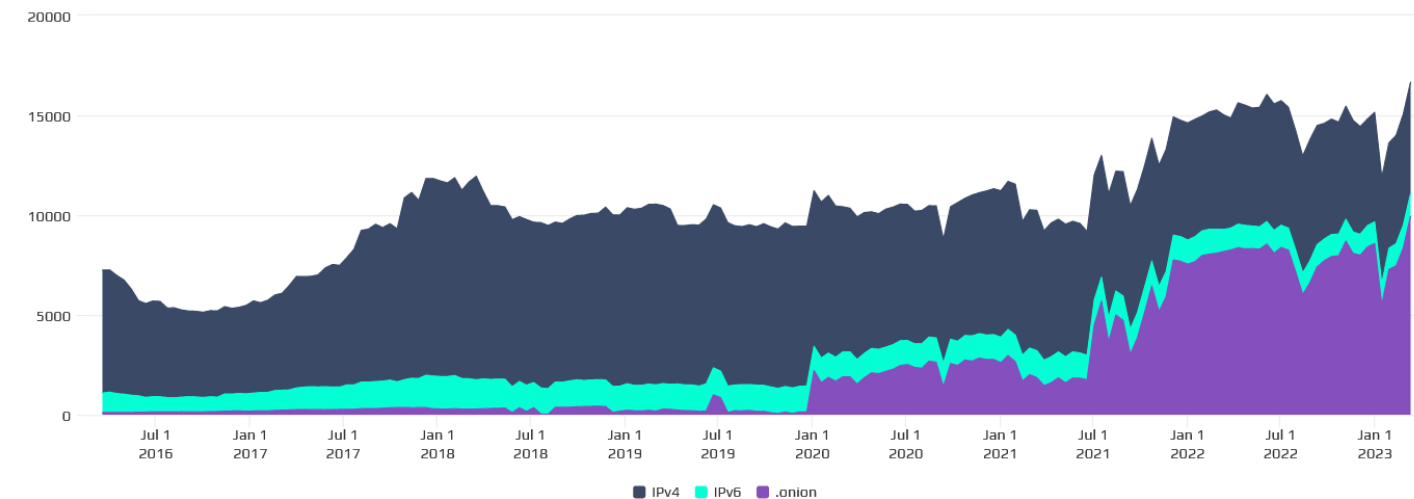
Since 7 years ago

NODES

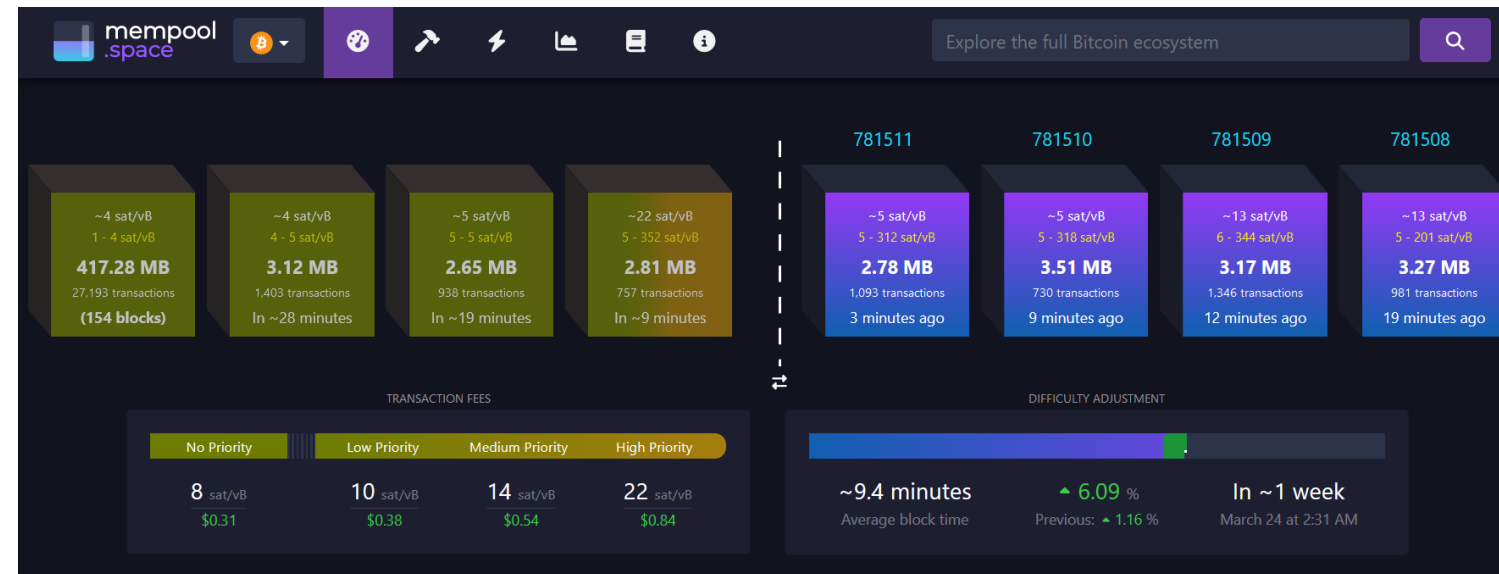
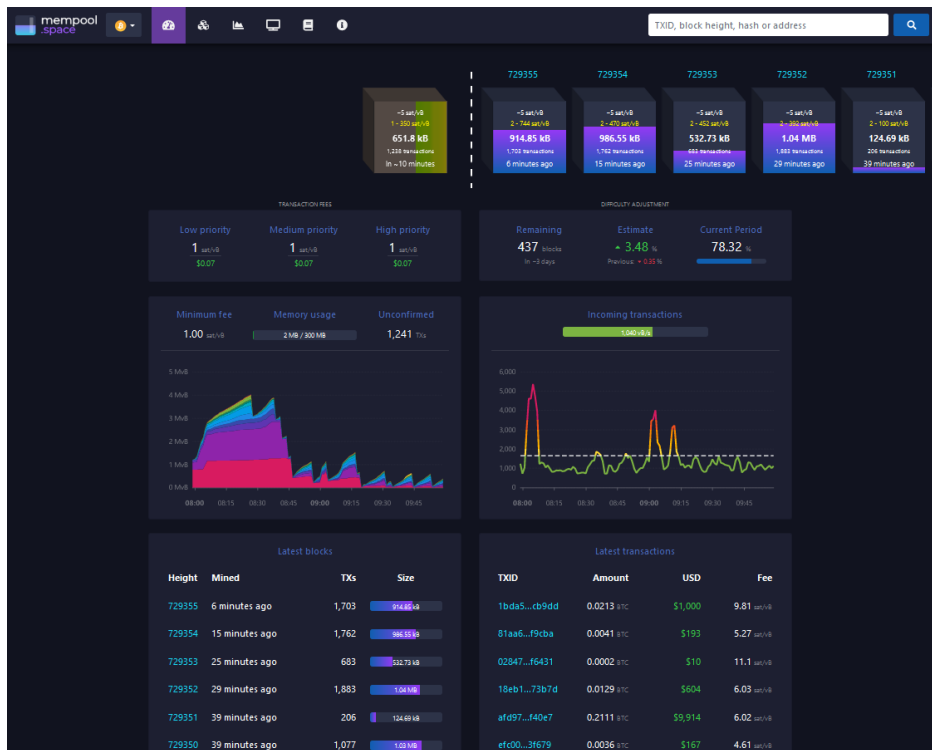
Chart shows the number of reachable Bitcoin nodes during the last 7 years. Series can be enabled or disabled from the legend to view the chart for specific networks.

24h 90d 1y 7y

Lo 5135 Hi 16656 Avg 10402 Last 16656 nodes

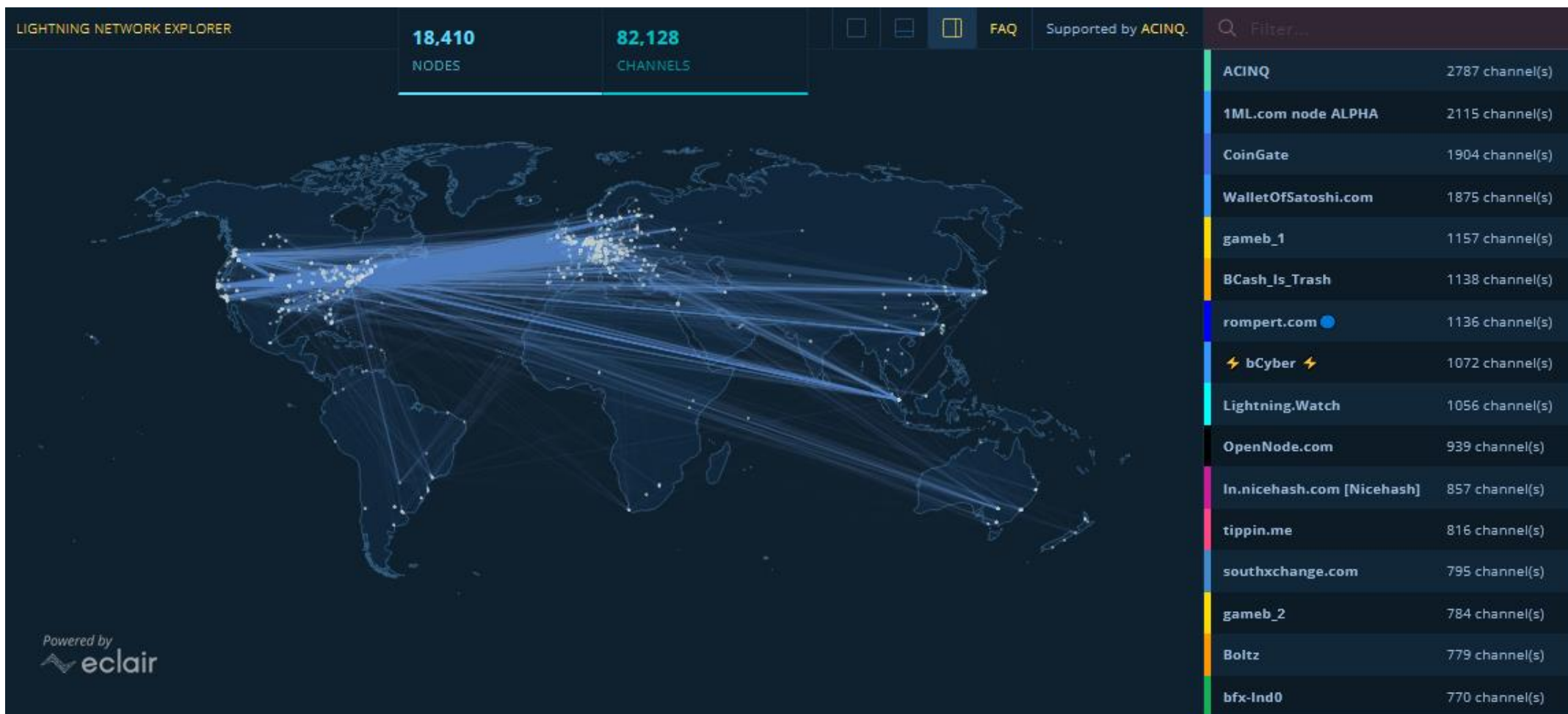


Popular mempool explorer – <https://mempool.space>



- Can be run on your own fullnode (privacy improvement)
- Testnet version <https://mempool.space/testnet>

Lightning network <https://explorer.acinq.co/>



- Frequent issue – Sparrow not started with `–testnet` switch
- Make breakout rooms (3 people per room)
- Ask them to perform some mutual transactions
- Connect to groups and discuss:
 - Why is tx as unconfirmed? Where it is?
 - Why are we waiting some minutes?
 - How to cheat on someone who will sell us car right after tx is in mempool?
 - double spent with higher fee => Invalidation of original tx
 - Doublespent tx can be prepared into file as binary blob
 - If already mined, it is more difficult, but still somewhat possible (chain reorg)

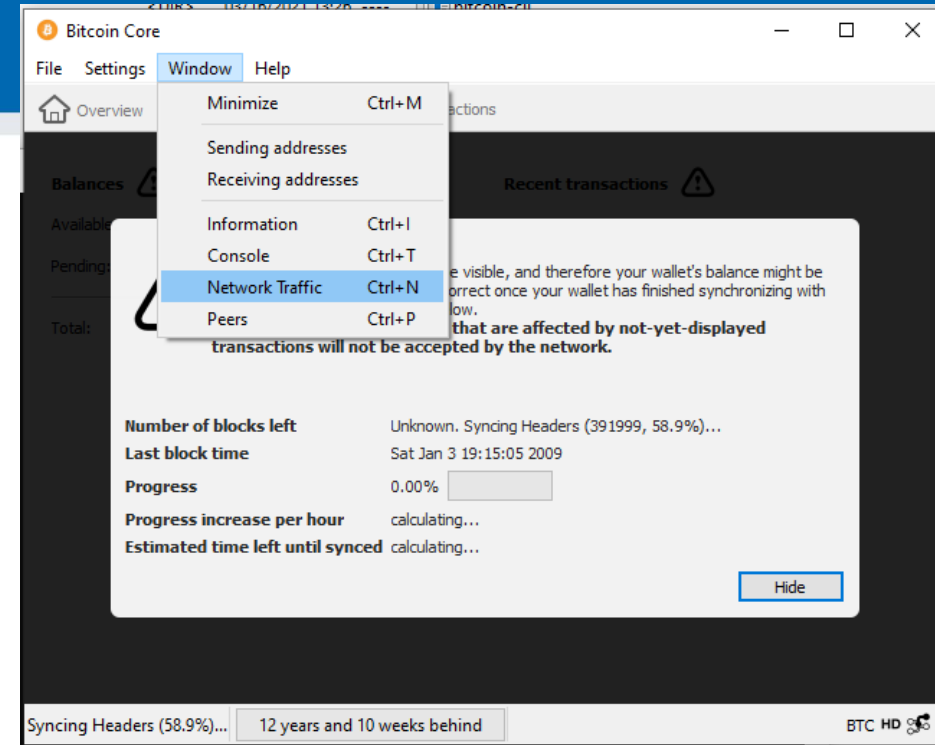
INTRO

TASK: USING BITCOIN CORE

- Ask people to run GUI, see the connected nodes, locate block files
- Explain how “Chancellor ... “ got there (coinbase data, used to randomize block during mining, pool info...)
- 100 blocks / file
- Importance of small blockchain => more fullnodes possible to be run
=> better decentralization

Own work: Using API of full node

- Get Bitcoin full node **24.0.1** (pick .zip or .gz)
 - <https://github.com/bitcoin/bitcoin/releases>
 - <https://bitcoincore.org/bin/bitcoin-core-24.0.1/>
 - Download and unpack .zip or .gz
- Download few blocks from real Bitcoin P2P network
 - Run bitcoin-qt, Window → Network Traffic (Ctrl+N), Peers (Ctrl+P)
 - Observe and document peers to which you connected (number, version, IP)
- Analyze first few blocks from blockchain
 - Look into Bitcoin/blocks/blk00000.dat (e.g., C:/Bitcoin/blocks/blk00000.dat)
 - If on Windows, Look for bitcoin folder also in your profile
 - c:\Users\your_name\AppData\Roaming\Bitcoin\blocks\



Questions

- Why is your full node connecting to other nodes?
- For how long is the Bitcoin network running now?
- What is the content of first block?
- What is the privacy advantage of sending/querying TXs using your full node?
- How can you compute the current supply of bitcoins?

Lister - [c:\Bitcoin\blocks\blk00000.dat]

File Edit Options Encoding Help

.....;úφ²z{.■z|,>gvÅa■L.¦êèQ2:■₁-K.^\n
J)½ I ...¼+|..... M.EThe Times 6
3/Jan/2009 Chancellor on brink of second bailout for banks ...≥.*....CA.g
è²■UH'.g±ªqθ₁.\r{α9.ªybαΩ.a ■I÷⁄?Lⁿ8-≤U.σ.±. ■\8M=|.ìWèLp+k±.¼....⁄⁄⁄..
.....oΓî.¦±|r±ªóF«c=0ô.âeßZ.£h₁.....ÿ Q².KºD₁⁄h...e.g{íú|T.≈■-|.ΦW#>.a⁄fI
.....nbö..... ≥.*....CA.û|8
ΦSQErj,æµ.±...«.Éü:b|f√iτö{µ<R₁uë7ò.¼αª.º..üµ"ör.f₁b.s¿,₁#B⁴Xe¼....⁄⁄⁄....
...H`δ.₁... π~óÉ"èBu.Ao⁄QY½ãhÄÜâ.....r²|T.% ■.ZZ|φ≥HX₁₁F\■óñtNΣ,1`"₁.ç■⁄fI ..
.T⁄a..... ≥.*....CA.r.¿\$]
[PR(Σ|₁.L.±.ñU½■7.⁄z@>ts ■µÉdÿ0.8R7⁄!g⁄>#dF|.½yá"«A*ñ1kw¼....⁄⁄⁄....
.⁄|ö|²ú#í■.⁄.p.ì.û{¼|ikc._bj....D÷r"´É+|]r≥v■_ .û. |ç>>{σ₁√₁íu|â.Ptsö}⁄fI ...α
φm..... ≥.*....CA.ö|Lτl[-
)∞.■ ò|ñ₁r⁴|τ`ö;óçf| .⁄. #-èqöáì.' &+t■.⁄Ptsî. u■>5P«çθ.o<r-¼....⁄⁄⁄.... I
DFòb«.,t⁄Ñ5α.o>@ ||²²úëU.⁄é....z.Qÿ-@|.2ê&+(cî∞S7±Ej»^φ⁴θσó..+■î⁄fI ..+■αr
..... ≥.*....CA..02■.ü\NR
.fhc\$. ≈σ₁.ñ■.ï.a0₁i....L..⁄07|ñ.■1₁.k|²C7>7«1á⁄ⁿñ⁴n₁g¼....⁄⁄⁄.... à.J
ãHÄ;ì".ï₁⁴Y₁...êºó_rûÉεU■|N....D.H■-τ%.±ä=#7.Ü8₁æ¼\«ê.çεö_T E(RcD|fIΣw..
 \ * ca uññæ-P./Åö-h

Run strings on already downloaded blocks

- **strings** command on Linux
- **strings** on Windows: <https://docs.microsoft.com/en-us/sysinternals/downloads/strings>
- **c:\Bitcoin\blocks**>strings -n 20 *.dat

TASK: USING BITCOIN-CLI (REGTEST)

- Check before seminar the commands (frequent change of API)
 - 0.21.0 requires min fee etc (`settxfee 0.00002`)
- Explain why `generatetoaddress 101`
 - Consensus rule: coinbase can be spent only after 100 blocks => less motivation for miners to fiddle with chain reorg
- Explain halving (210000 blocks on mainnet, 150 on regtest)
- Explain why after sending 10 to other, resulting balance is 49.999...
 - Mining fee subtracted (but we are miners! But we will get fee only in 100 blocks)

Note: Assumed version 24.0.1

```
>bitcoin-cli -regtest getbalance  
50.00000000
```

Note: on Windows, do not use PowerShell

Using API: Bitcoin -regtest

- Optional: regtest network blocks are stored in \Bitcoin\regtest\ (Windows) or ~/.bitcoin/regtest (Linux)
 - Run "del /S /Q "%APPDATA%\Bitcoin\regtest\" to erase previous one (on LINUX, remove ~/.bitcoin/regtest)
- Run local network (bitcoin daemon)
 - `bitcoind -regtest`
- Open one additional terminal (Win->cmd; Linux->terminal)
- Create new wallet
 - `bitcoin-cli -regtest createwallet "testwallet"`
- Obtain new address for future mined bitcoins (=> `miner_address`)
 - `bitcoin-cli -regtest getnewaddress`
- Mine 101 blocks: `bitcoin-cli -regtest generatetoaddress 101 miner_address`
- Check your balance: `bitcoin-cli -regtest getbalance`

This is necessary from
0.20.0 and higher

Using API: Bitcoin -regtest

- Set desired transaction fee BTC/kvB (wallets typically auto computing for you)
 - `bitcoin-cli -regtest settxfee 0.00002`
- Send previously mined bitcoins to new address (`getnewaddress`→`new_address`)
 - `bitcoin-cli -regtest sendtoaddress new_address 10.00`
- Display info about transaction:
 - `bitcoin-cli -regtest gettransaction txid`
- Mine additional to block to include new TX into blockchain...
 - <https://bitcoin.org/en/developer-examples>, <https://bitcoin.org/en/developer-reference#bitcoin-core-apis>
- Verify total supply: `bitcoin-cli -regtest gettxoutsetinfo`

Questions A.

- What type of address you get via **getnewaddress** command?
- How you can distinguish between addresses for mainnet, testnet and regtest?
- Can you send mined regtest bitcoins to mainnet address (e.g., bc1xxxx...)?
- How many bitcoins you are supposed to have after mining 150 blocks? Why **getbalance** is showing only 2500 btc?
- How the block reward changes on mainnet? How it changes on regtest net?

TASK: BITCOIN QUESTIONS

Questions B (you and ChatGPT)

- Answer the question below with your peers
 - How can I pay you 1btc if I have only one UTXO worth of 5btc?
 - What will happen if I will try send double-spending transaction to Bitcoin network?
 - Why should you use fresh new address for every receive transaction?
 - What will happen if you create pull request to increasing total number of bitcoins from 21M to 100M at <https://github.com/bitcoin/bitcoin>?
- Ask ChatGPT the question below, then discuss the answer provided critically
 - What attacks are possible if I'm using Bitcoin wallet which is not connected to my trusted full node?

TASK: USING SIGNATURE COORDINATOR



SINGLE-SIGNATURE WALLET (SW-ONLY)



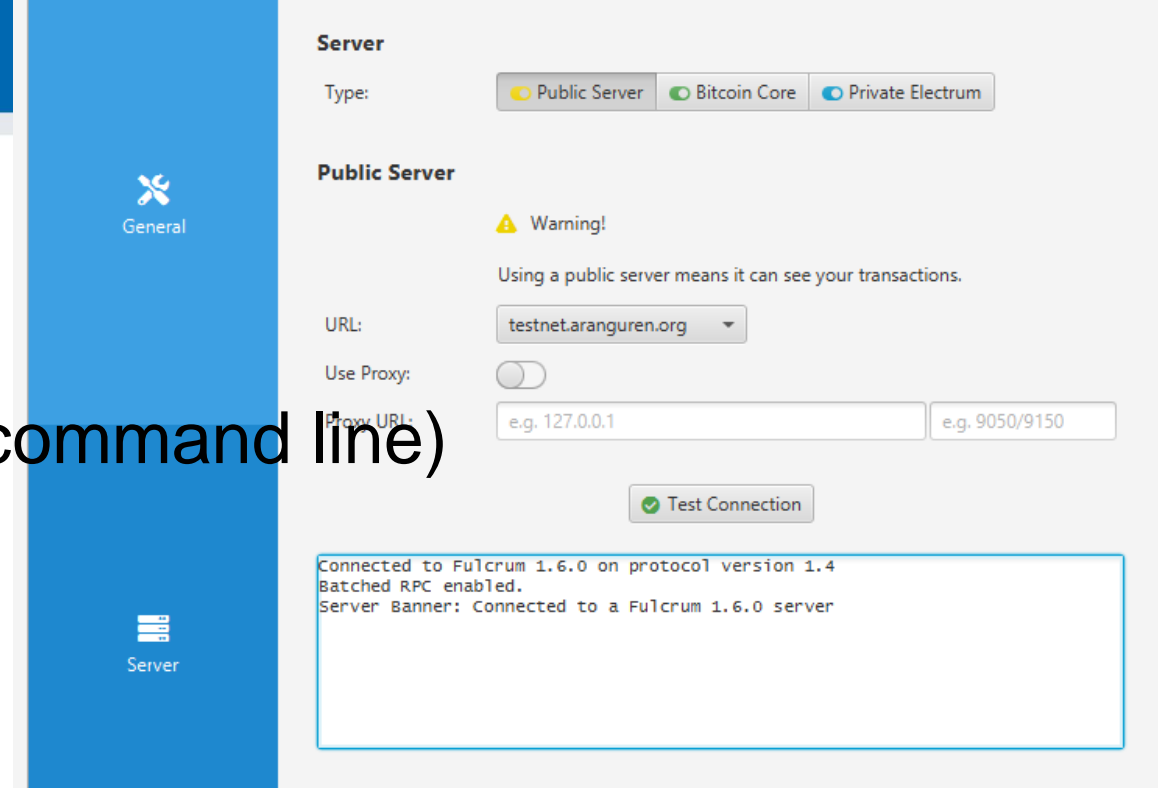
Sparrow wallet (v1.6.6)

- <https://www.sparrowwallet.com/download/>
- For serious work, always verify binary releases (`gpg --verify`)
- Well-known and maintained, Java-based, minimum other dependencies, focus on medium and advanced users
- Sparrow is “signing coordinator” – private keys can inside or elsewhere
- Basic functionality
 - Open-source wallet, non-custodial wallet
 - Support for software and hardware wallets, multisignature coordinator
 - Whirlpool CoinJoin client
 - Supports also advanced features (PayJoin, Taproot addresses...)

(Examples created for Sparrow 1.6.6)

Starting Sparrow wallet

- Run your wallet with testnet switch (command line)
 - `./sparrow -n testnet`
 - `Sparrow.exe -n testnet`
- Use Public Server option if asked
 - Test Connection to verify connectivity
 - Can be changed later File → Settings
- (Bitcoin Core and Private Electrum are more private options)
 - You would be connecting to your own fullnode (but you must have one 😊)
- Check that you are online
 - (right bottom)



Connected to ssl://testnet.aranguren.org:51002 at height 2345147
 Warning! You are connected to a public server and sharing your transaction data with it.
 For better privacy, consider using your own Bitcoin Core node or private Electrum server.

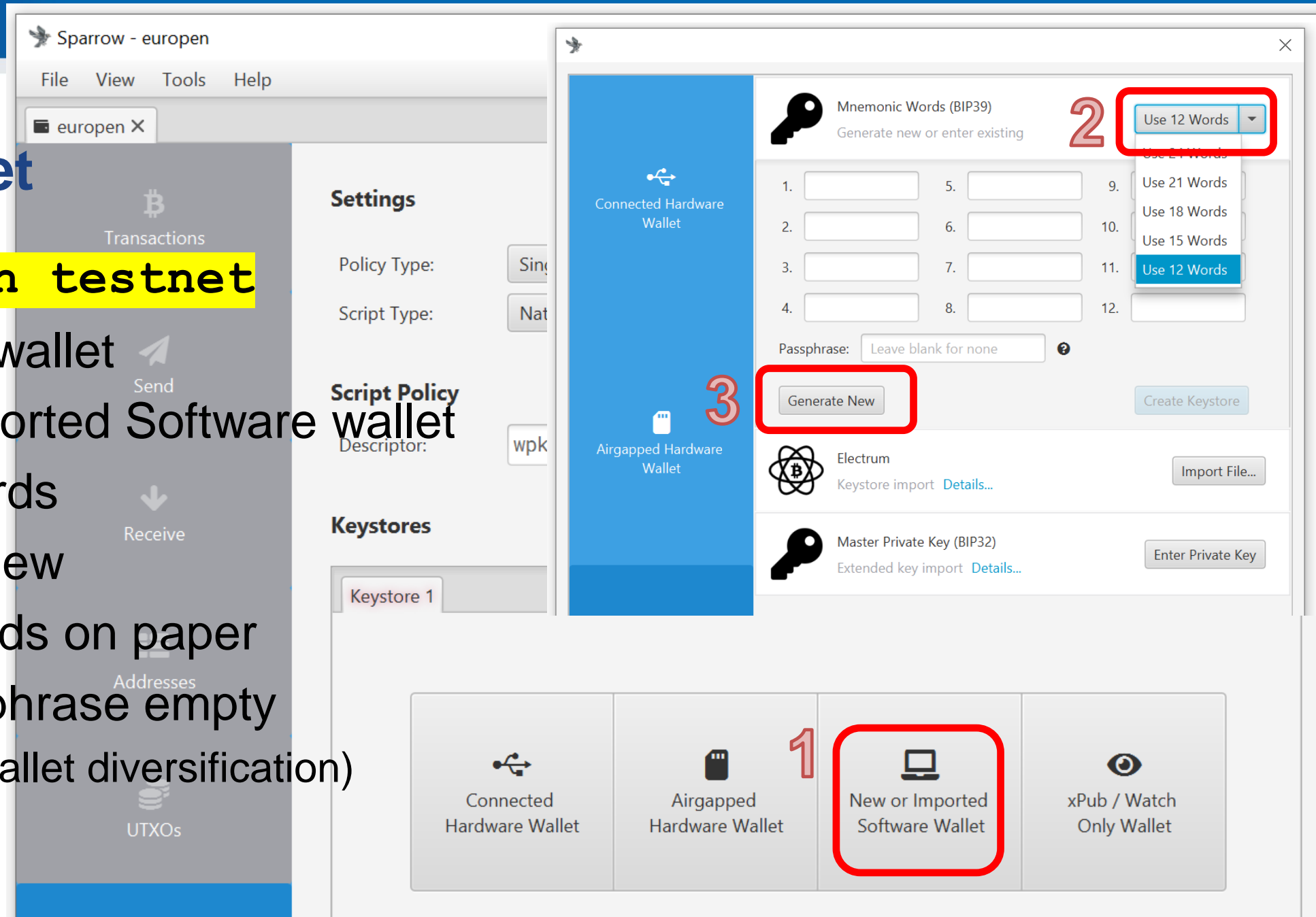


Generating new “wallet”

- A “wallet” is key management software controlling your private and public keys (ECDSA, Schnorr)
- The most important part of wallet is random number called root seed (128 or 256 bits)
- Root seed is used to deterministically generate practically unlimited number of keypairs
 - Specified in BIP32, “root seed” and “derivation path” used to derive next private key => next public key => next address
- Clever construction allowing to compute future public keys (and only public keys) for specified derivation path without the need for root seed (aka xpub or extended public key)
 - Knowledge of xpub allows to compute all future public keys, but not private keys
 - Owner of root seed can compute all future private keys and their corresponding public keys
 - xpub allows to pay someone to fresh addresses noninteractively (no interaction with owner of root seed required), receiver will only later compute candidate private keys and their public keys to check for total balance (== set of UTXOs)
- Wallet software is monitoring blockchain for addresses corresponding to stored root seed (or xpub)
- Root seed can be stored:
 1. Directly in software wallet (file on harddisk, optionally encrypted) == aka hot wallet, least secure against malware
 2. Loaded every time before use (e.g., from QR code), still vulnerable to malware during use
 3. On external hardware signing device called hardware wallet (the most secure option)

Create wallet

- `sparrow -n testnet`
- File → New wallet
- 1. New or Imported Software wallet
- 2. Use 12 Words
- 3. Generate New
- Write 12 words on paper
- Leave Passphrase empty
– (additional wallet diversification)

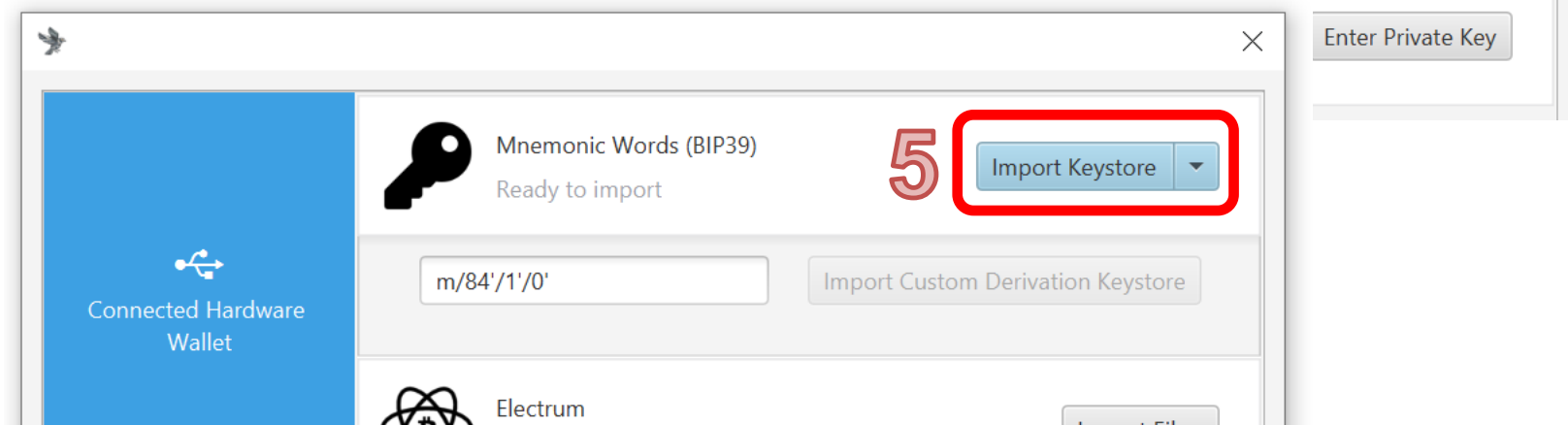
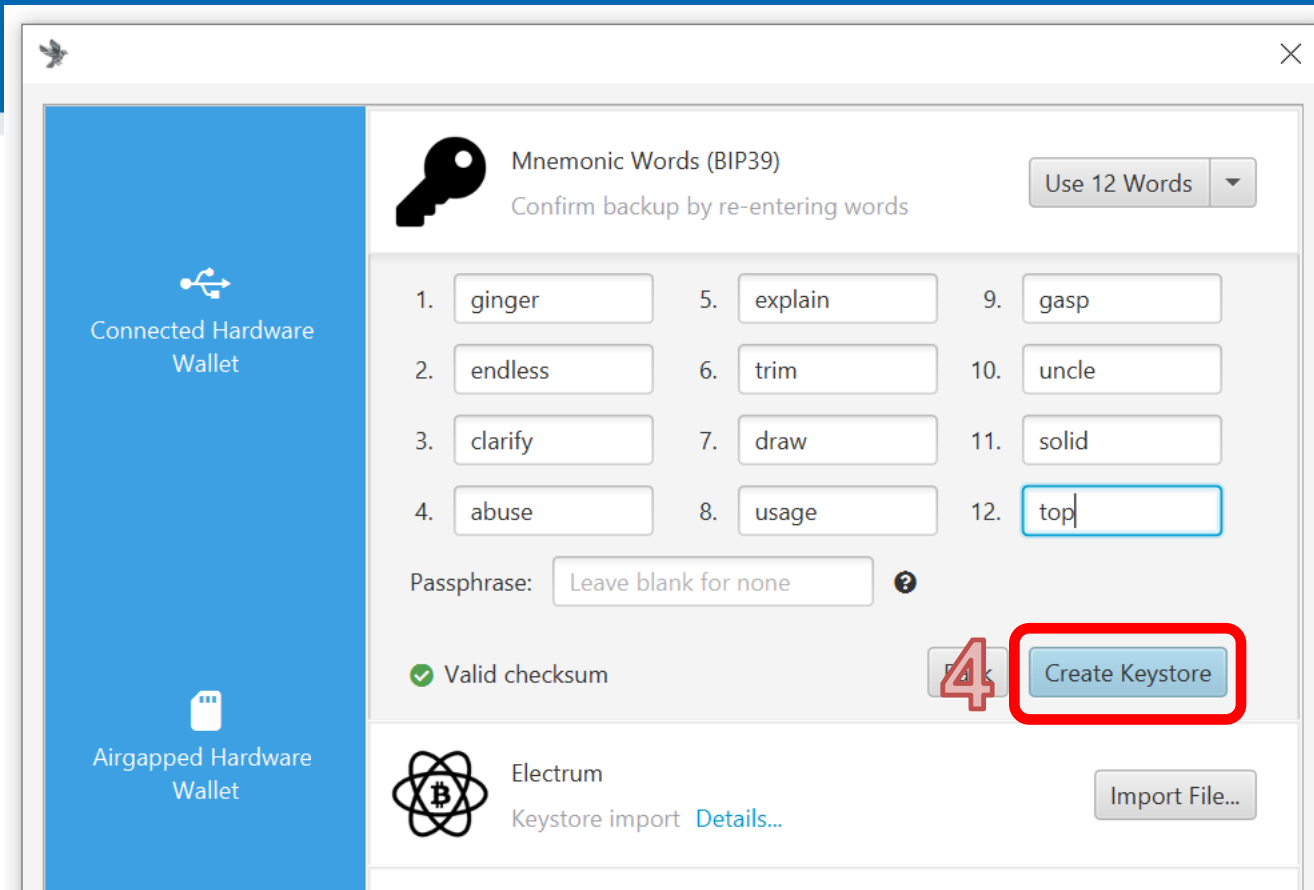


Create wallet

4. Create Keystore

- Confirm backup
- Reenter words

5. Import Keystore



Create wallet

6. Apply

7. Set password or leave empty

- (encryption of local wallet file)

- Local wallet contains seed

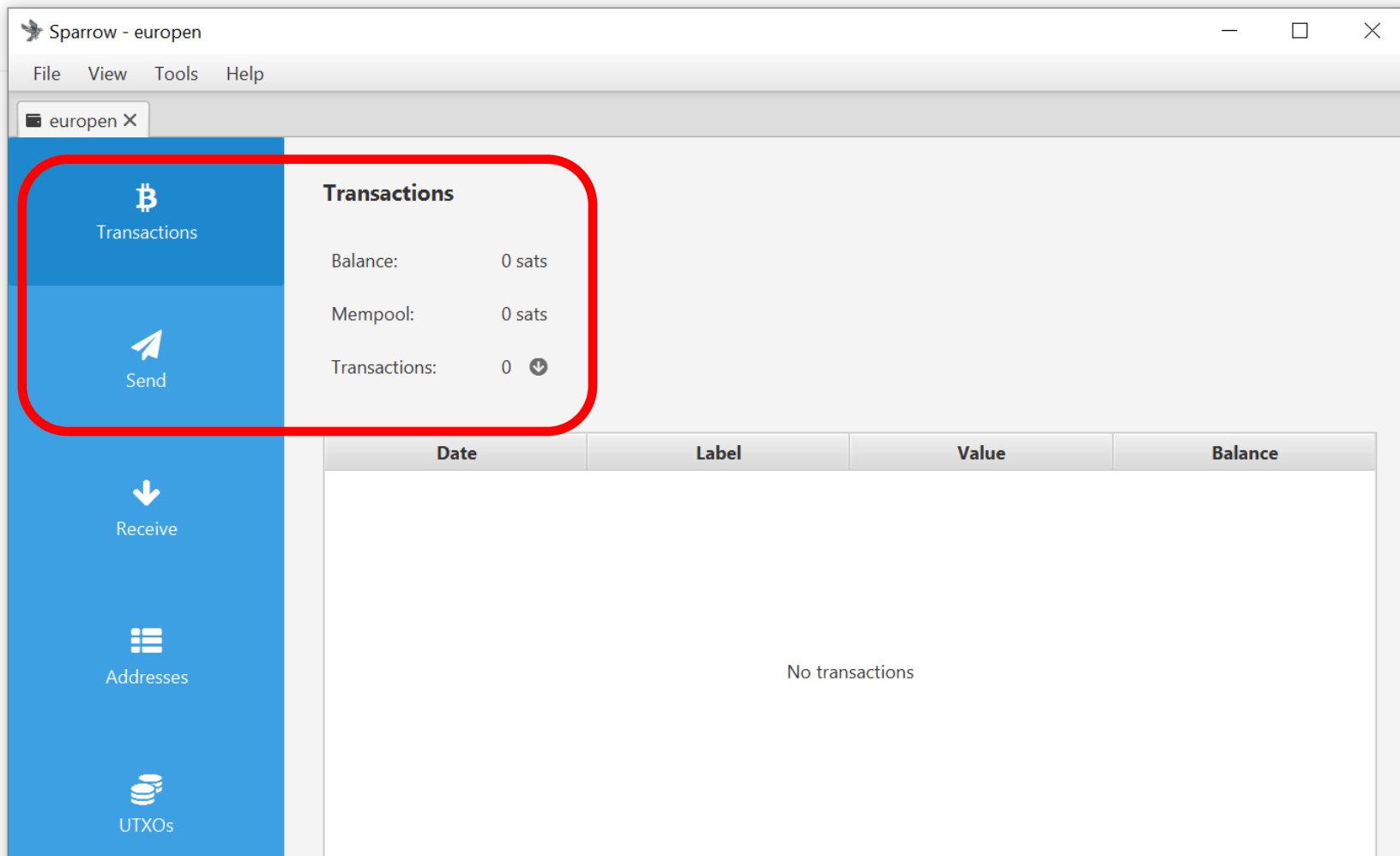
- *.mv.db file

- File → Open wallet



The screenshot shows the Sparrow wallet interface. The 'Settings' panel is active, showing 'Policy Type' set to 'Single Signature' and 'Script Type' set to 'Native Segwit (P2WPKH)'. The 'Keystores' panel shows a 'BIP39' wallet with a 'Software Wallet' type, label 'BIP39', and master fingerprint 'bec2be2c'. The 'Derivation' path is 'm/84' / 1' / 0''. The 'tpub / vpub' address is visible. A 'Wallet Password' dialog box is open, asking to add a password. The 'No Password' button is highlighted with a red box and the number 7. The 'Apply' button in the 'Keystores' panel is highlighted with a red box and the number 6.

Wallet created (but empty 😊)

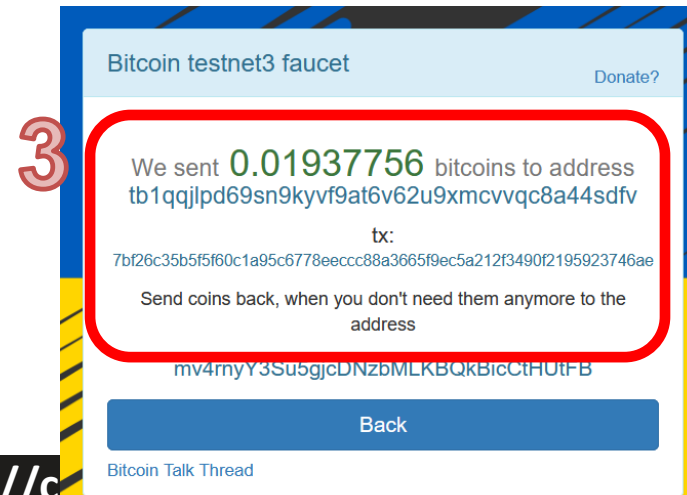
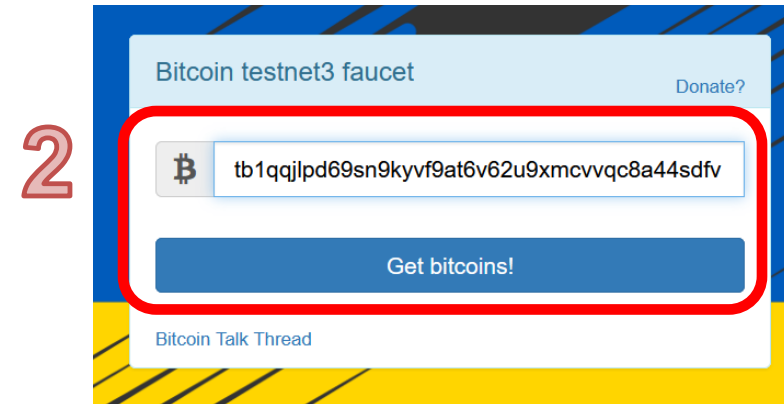
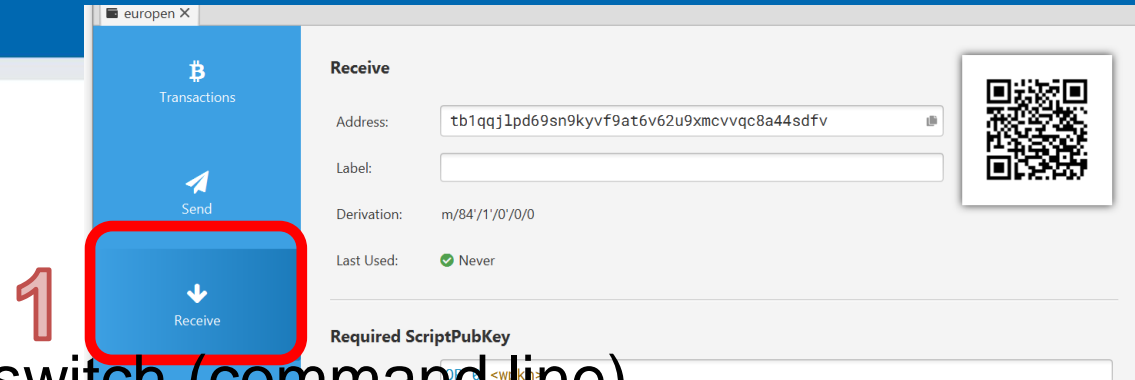


Receiving (testnet) bitcoins

- You generate new “address”
 - deterministically derived from your root seed and fresh derivation path (path + counter) => new ECDSA keypair [BIP32]
 - public key X is pasted into locking script (“who can sign with private key verifiable with X can move bitcoin further”) and hashed => “address” [P2SH/P2WSH] (Pay to witness script hash)
- Service coinfaucet.eu owns multiple tBTC
 - Service is providing limited number of test bitcoins (tBTC) for free
 - Service owns UTXOs => someone previously locked some tBTC to their keypair(s)
 - Service creates new transaction with some tBTC locked to your “address”
 - New transaction is broadcasted to Bitcoin P2P network and stored in mempools (set of unconfirmed transactions)
- Miners will eventually include this transaction into new block (head of blockchain)
 - Confirmed and removed from mempools
 - Your Sparrow wallet is monitoring both mempool and blockchain (instant notification about pending transaction)

Getting test bitcoins (tBTC)

- If not running, run your wallet with testnet switch (command line)
 - E.g., `./sparrow -n testnet`
 - Generate new (testnet) receive address
- Go to <https://coinfaucet.eu/en/btc-testnet/>
 - If doesn't work use <https://testnet-faucet.com/btc-testnet/>
 - Insert your testnet receive address
 - You may get more every 12 hours (per single IP)
 - (but please don't abuse)
- Check your tx: <https://mempool.space/testnet>
- Testnet TX explorer: <https://blockstream.info/testnet/>
 - Software visualizing blockchain



Sparrow - europen

File View Tools Help

europen X

Transactions

Send

Receive

Addresses

UTXOs

Settings

Receive

Address:

Label:

Derivation: m/84'/1'/0'/0/0

Last Used: Never

Required ScriptPubKey

Script:

Output

Sparrow - europen

File View Tools Help

europen X

Bitcoin testnet3 faucet [Donate?](#)

Get bitcoin

Bitcoin Talk Thread

Bitcoin testnet3 faucet [Donate?](#)

We sent **0.01937756** bitcoins to address
tb1qqj1pd69sn9kyvf9at6v62u9xmcvvc8a44sdfv

tx:
7bf26c35b5f5f60c1a95c6778ecccc88a3665f9ec5a212f3490f2195923746ae

Send coins back, when you don't need them anymore to the
address

mv4rnyY3Su5gjcDNzbMLKBQkBicCtHUtFB

Back

Bitcoin Talk Thread

Sparrow - europen

File View Tools Help

europen X

Transactions

Send

Receive

Transactions

Balance: 1,937,756 sats \$ 696.32

Mempool: 1,937,756 sats \$ 696.32

Transactions: 1

0 10 20 30 40 50 60 70 80 90 100 110

Date	Label	Value	Balance
▼ Unconfirmed		○ 1,937,756	○ 1,937,756
	Received to 7b... 🔍 ↗	1,937,756	

Blockchain explorers

- Everybody with access to Bitcoin P2P network can analyze blockchain
 - Everybody running Bitcoin fullnode
 - All past transactions, human-readable visualizations, search for address...
 - Convenient quick check of funds send
- Third parties are operating public explorers (convenient, but privacy)
 - It is very important to use Tor Browser when accessing public block explorers
 - Explorer operator may log your IP address and transactions you are searching for and later sell it (chain surveillance companies)
 - Heuristic assumption that you are the owner of funds for searched transaction
- Ideally use your own full node with your own blockchain explorer
- Sparrow wallet allows you to visualize your transactions
 - Inputs, outputs, feed paid



Task: send some tBTC to your peer

- Select one of your neighbors as peer (PC1 and PC2)
- Obtain his/her receive address
 - Via messenger: PC2 → Receive tab → Copy address → send via Signal → PC1
 - Via QR: PC2 → Receive tab ; PC1 → Send → camera icon → scan address QR
- Enter some sats into Amount box
 - Observe visualized transaction below (more inputs may be added)
- Try again, but now with manual coin selection
 - UTXO tab → select one or more → Send Selected

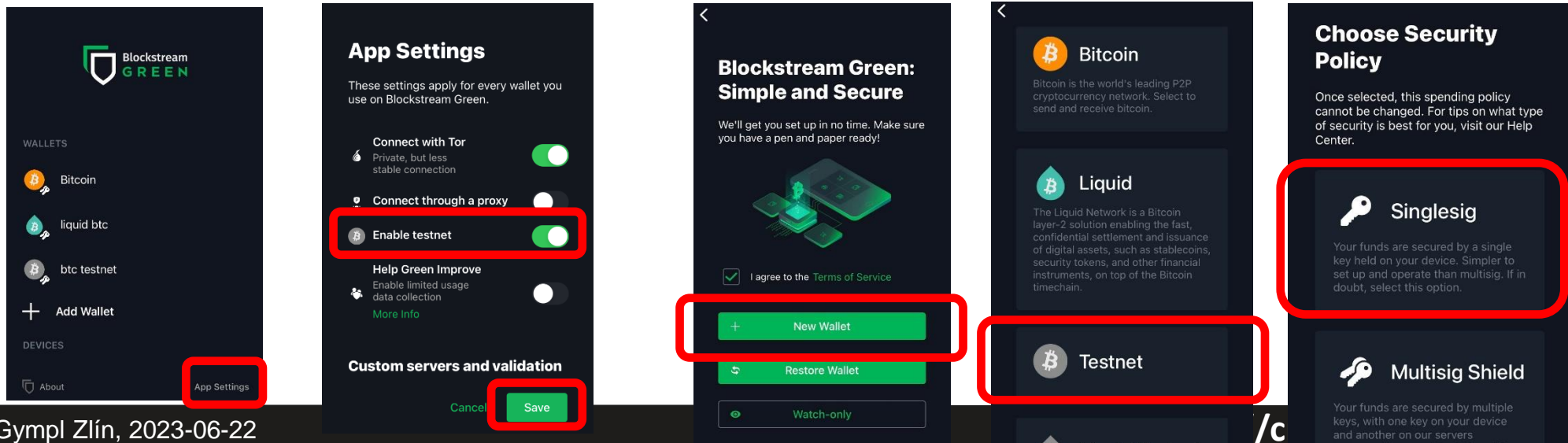
PC1

PC2

The image shows two computer screens, PC1 and PC2, displaying a Bitcoin wallet interface. On PC1, the 'Send' screen is active, with a red box highlighting the 'Pay to' field containing the address `tb1qz2qgh3x0kf5v1g8vekcaawuavr1e2z2qd0ru9s`. The amount is set to 123,000 sats (\$42.62) and the fee is 141 sats (\$0.05). A transaction diagram at the bottom shows an input UTXO being split into an output for 'to eur2', a change output 'tb1qwadf...', and a fee. On PC2, the 'Receive' screen is shown, also with a red box highlighting the 'Address' field with the same address and a QR code to the right. The 'Required ScriptPubKey' field contains the script `OP_0 <wpkh>`.

Get mobile wallet

- Get Green wallet by Blockstream on your mobile phone
 - <https://apps.apple.com/us/app/green-bitcoin-wallet/id1402243590>
 - https://play.google.com/store/apps/details?id=com.greenaddress.greenbits_android_wallet&hl=en&gl=us
 - Pick testnet option
- Try send between to Green and Sparrow



Questions

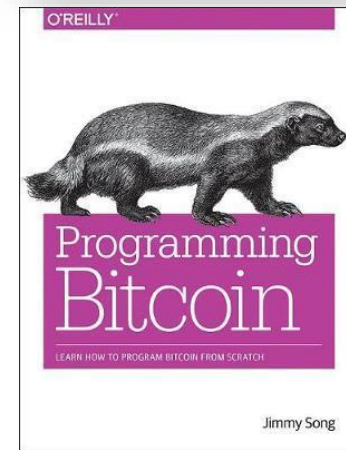
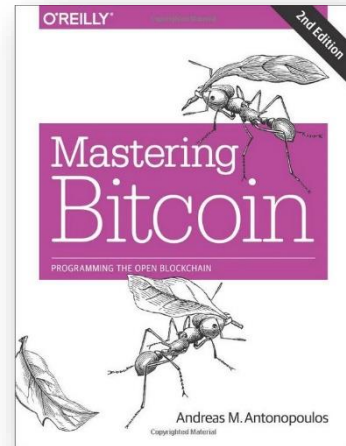
- Can you get less than 1 bitcoin?
- How can you get some real bitcoin(s)? (three different options)
- How can I pay you 1btc if I have only one UTXO worth of 5btc?
- Can you reverse bitcoin payment if send to wrong address?
- Why “Not your keys, not your bitcoin”? What is non-custodial wallet?
- How can someone steal your bitcoins? (At least three different options)
- For what reason are miners consuming a lot of energy?
- How frequently is new block with transactions included to blockchain?
- If I will send you bitcoin on-chain, can you tell from whom I got it?
- Why should you use fresh new address for every receive transaction?
- Why is theoretical maximal limit of on-chain transactions ~6-7tx/sec?
- Can I operate full Bitcoin node without owning any bitcoin?
- Can you receive bitcoins without operating full node?
- What attacks are possible if I’m using Bitcoin wallet which is not connected to my trusted full node?

(Look for your testnet txs from bitcoin core client)

- We send testnet tBTC => there must be corresponding transaction
- Can we look it on our own fullnode (bitcoin-qt we used previously)?
- Possible, but you need to download whole testnet3 blockchain
 - Files are located in `\Bitcoin\testnet3\`
- When searching for transaction (locally), use `--testnet` switch
 - `bitcoin-cli -testnet`

Further reading

- Mastering Bitcoin (Andreas M. Antonopoulos and others)
 - <https://github.com/bitcoinbook/bitcoinbook>
- Programming Bitcoin (Jimmy Song)
 - <https://github.com/jimmy-song/programming-bitcoin>
- List of interesting resources
 - <https://blockonomi.com/bitcoin-educational-resources/>
 - <https://learnmeabitcoin.com/>, <https://learnmeabitcoin.com/technical/>
- Bitcoin Twitter, Nostr (<https://nostr.com/clients>)
 - [@adam3us](#) [@gladstein](#) [@ODELL](#) [@saylor](#) ...
- Podcasts
 - <https://www.whatbitcoindid.com/> <https://stephanlivera.com/>



Getting some real sats (1/100000000 B)

- You can get/buy fraction of bitcoin
- Transaction on mainnet
 - potentially costly, ~10mins to execute
 - Mainnet is not for buying coffee
- Satoshi's on Lightning – instant and near free
 1. Download Wallet of Satoshi
 2. Click Receive → QRCode displayed
 3. Come to get some
 4. Try and learn

