

Bitcoin basics



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



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

Centre for Research on Cryptography and Security, Masaryk University

CRCS

Centre for Research on
Cryptography and Security

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

Bitcoin is built on decades of research ideas.
The main innovation is double-spending prevention *without* trusted central party



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

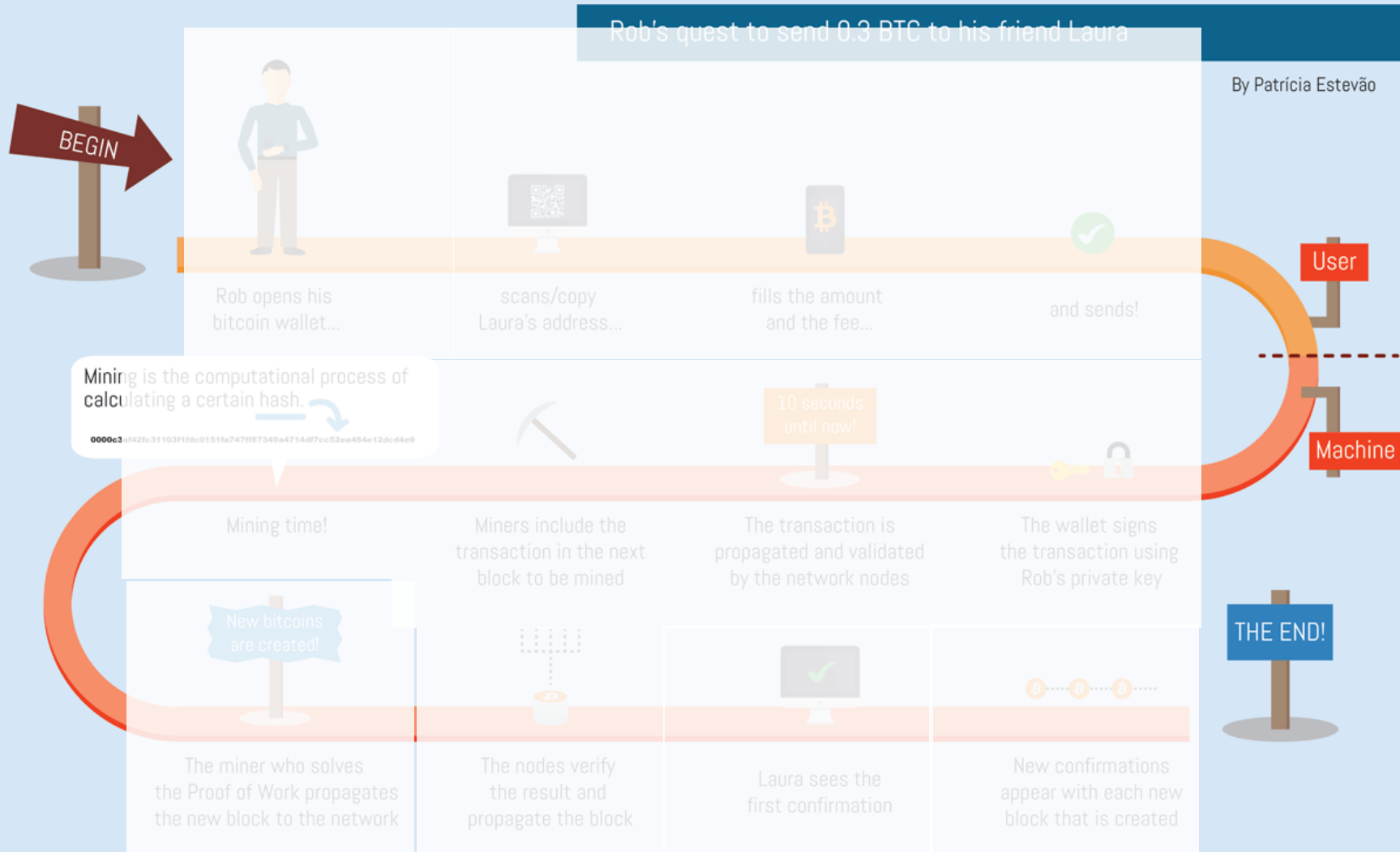


Overview

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



- 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

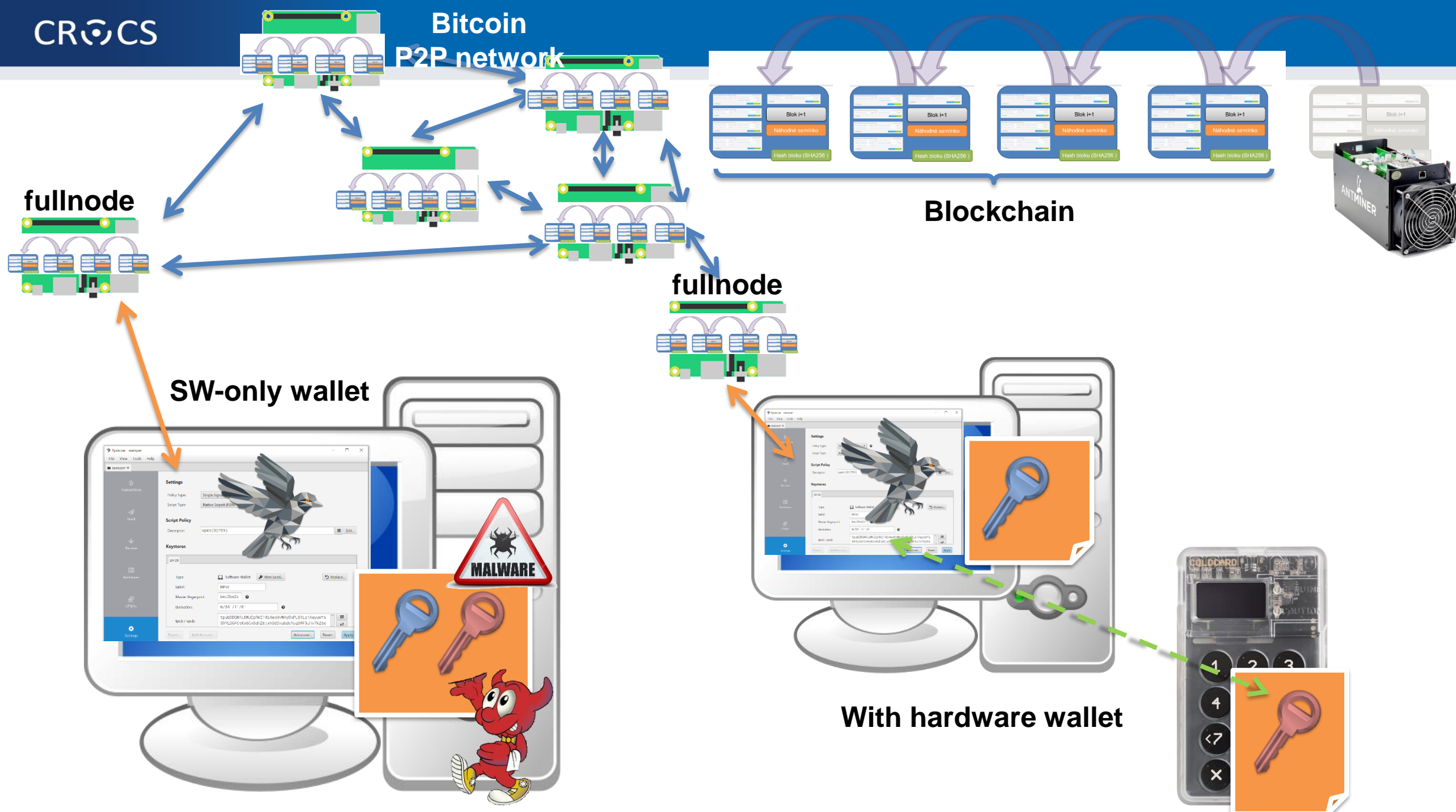
- All transactions recorded on public ledger; validity of every “bitcoin” easy to verify
- Total number of bitcoins in circulation easy to assess (monetary policy, fixed supply)

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)

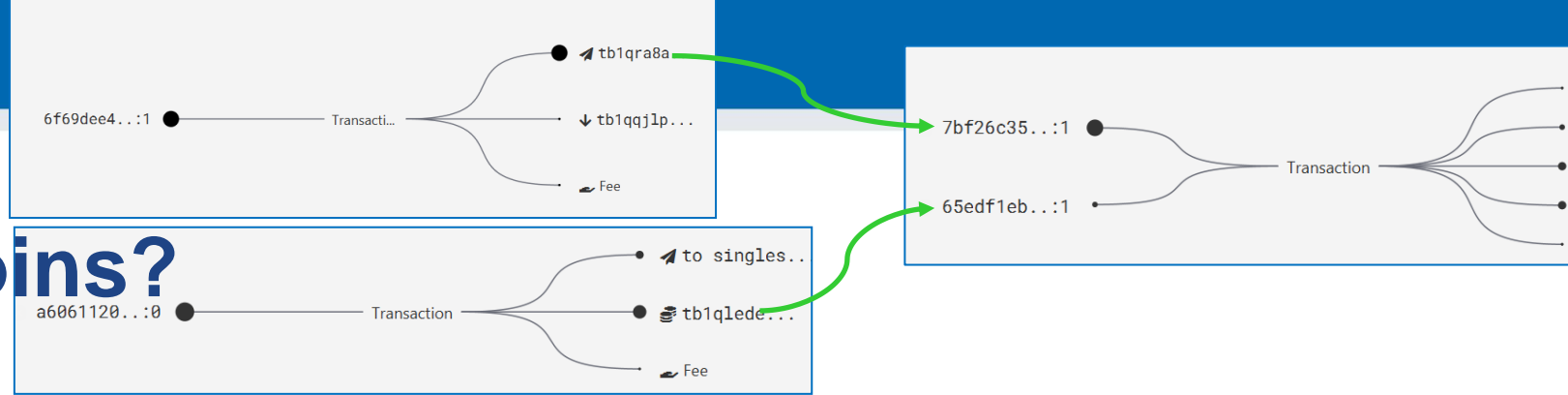
4. Pseudonymity of participants

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



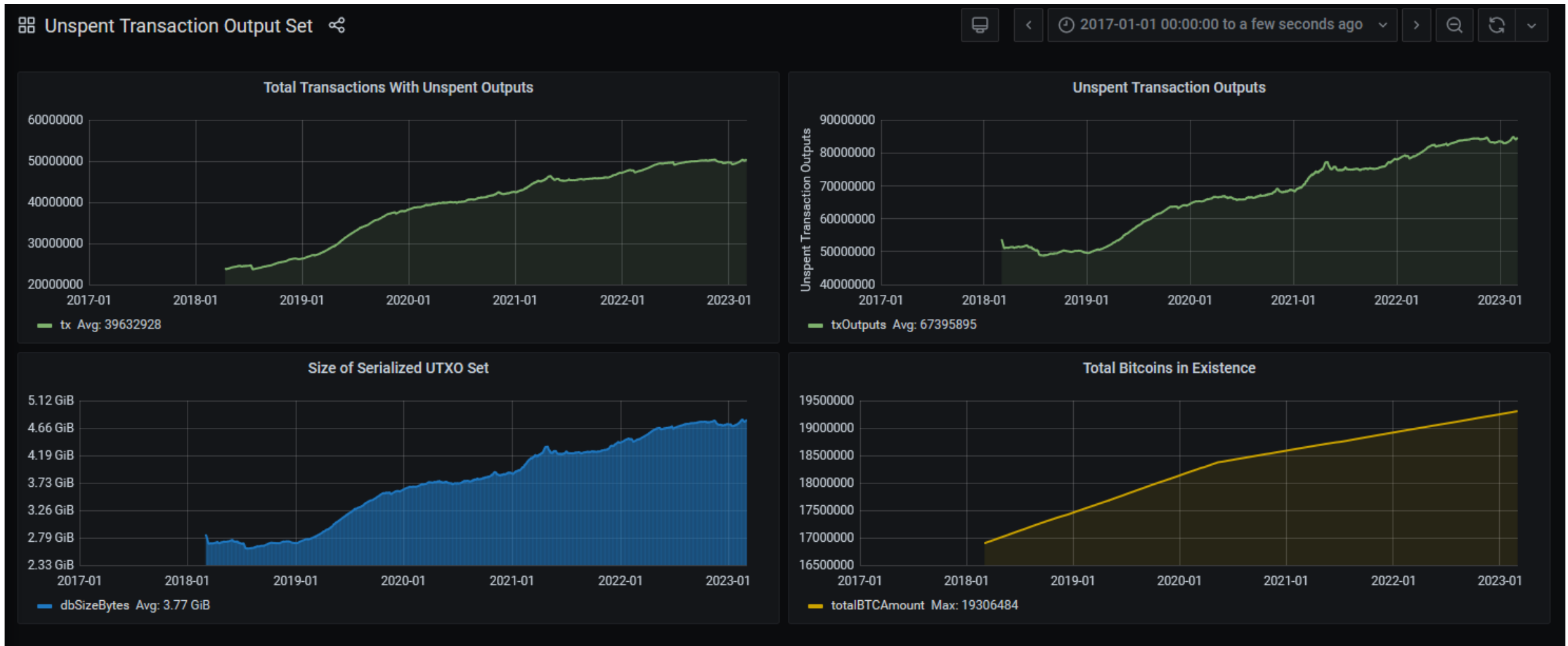


Where are my bitcoins?



- 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 int 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

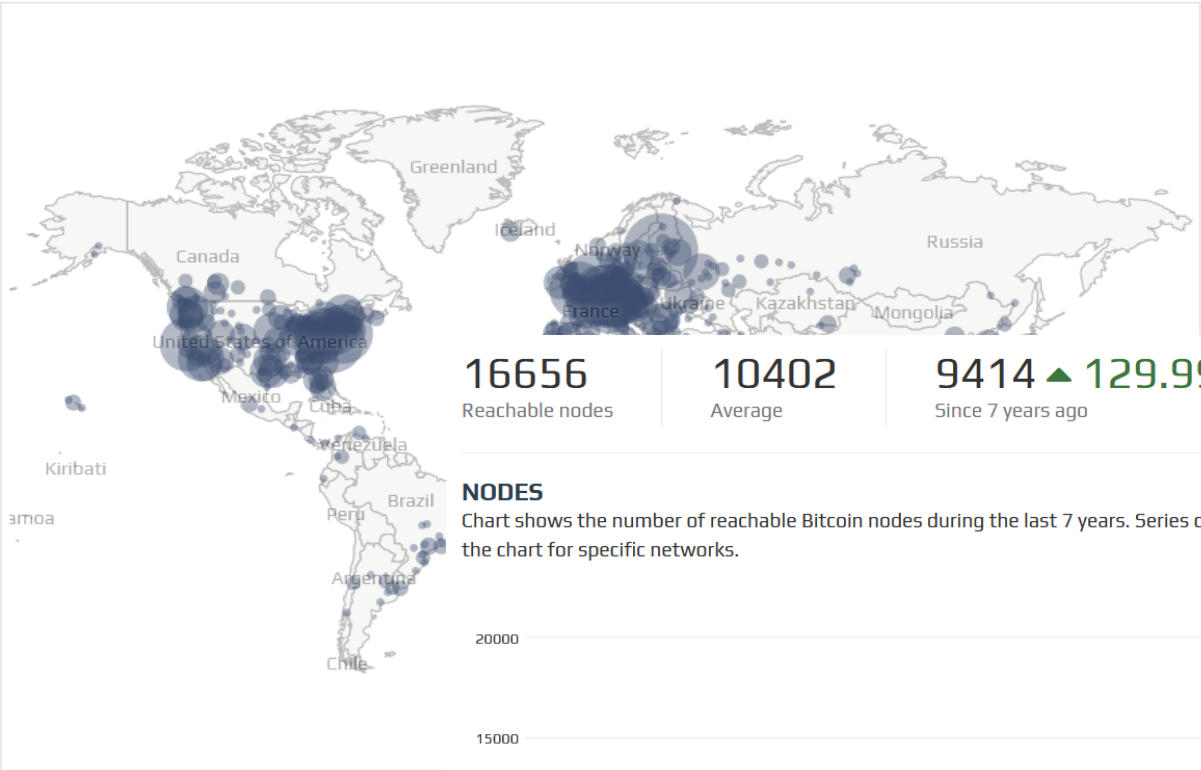
16537 NODES

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%)

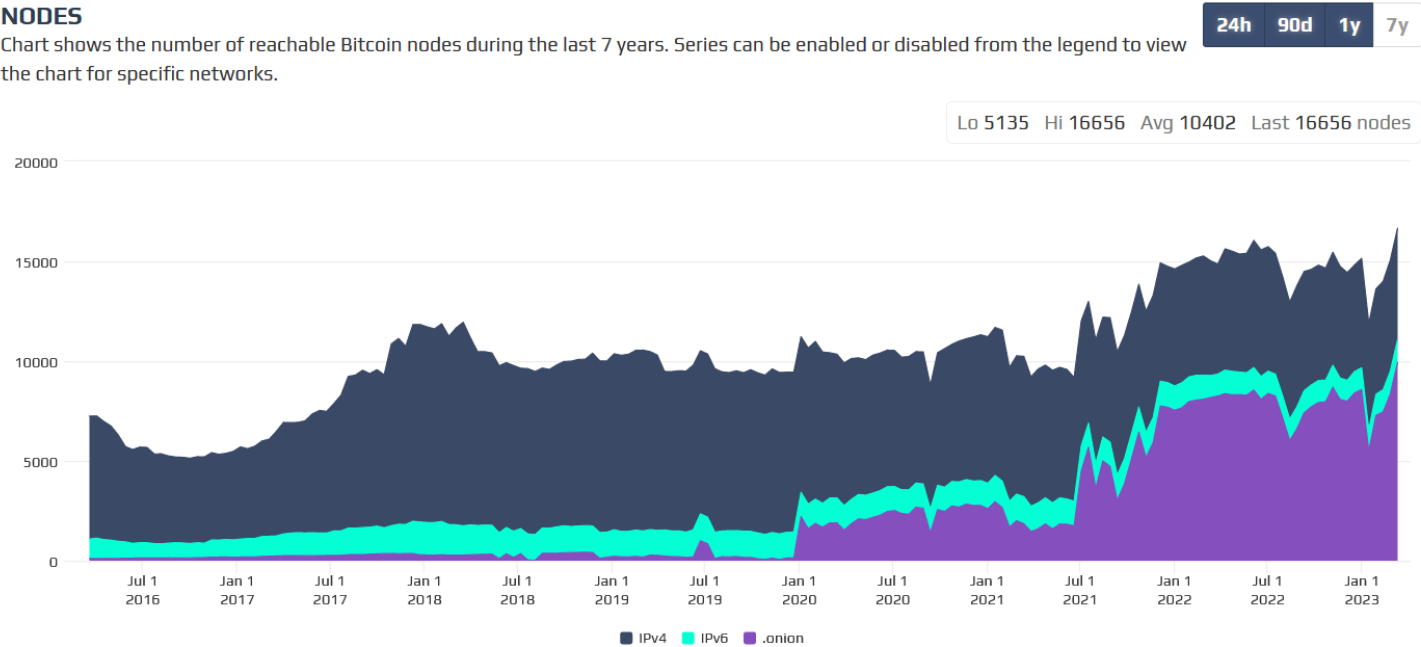


Map shows concentration of reachable

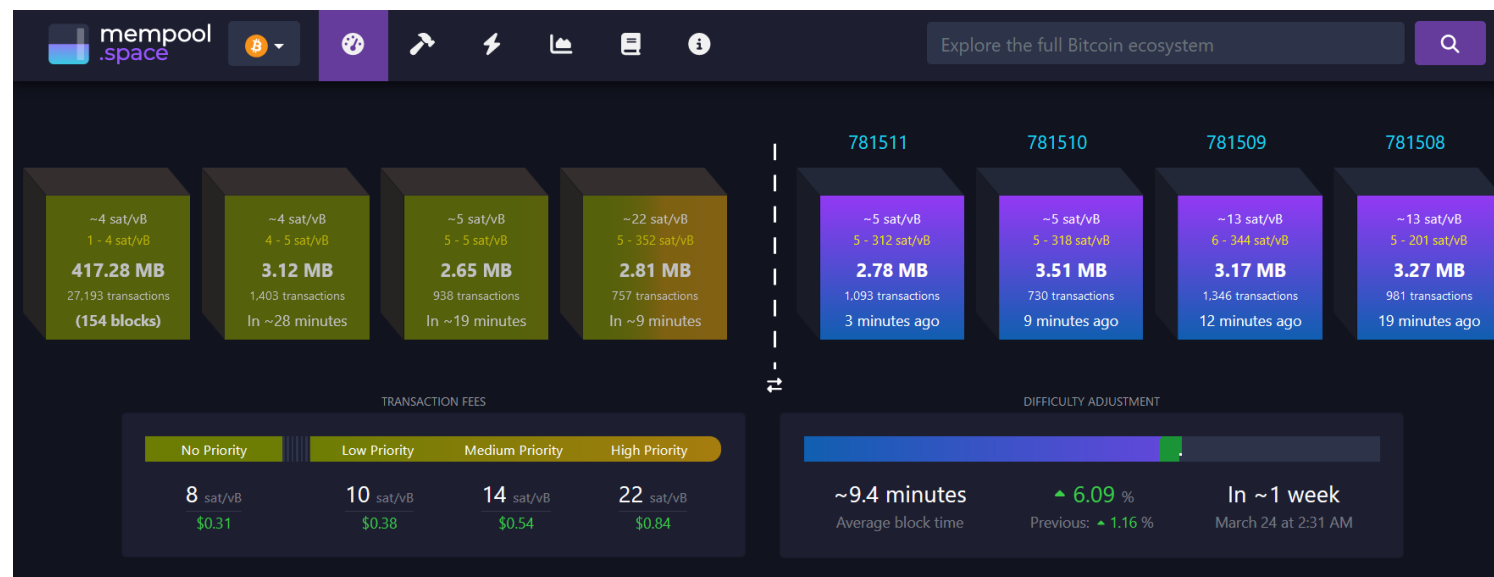
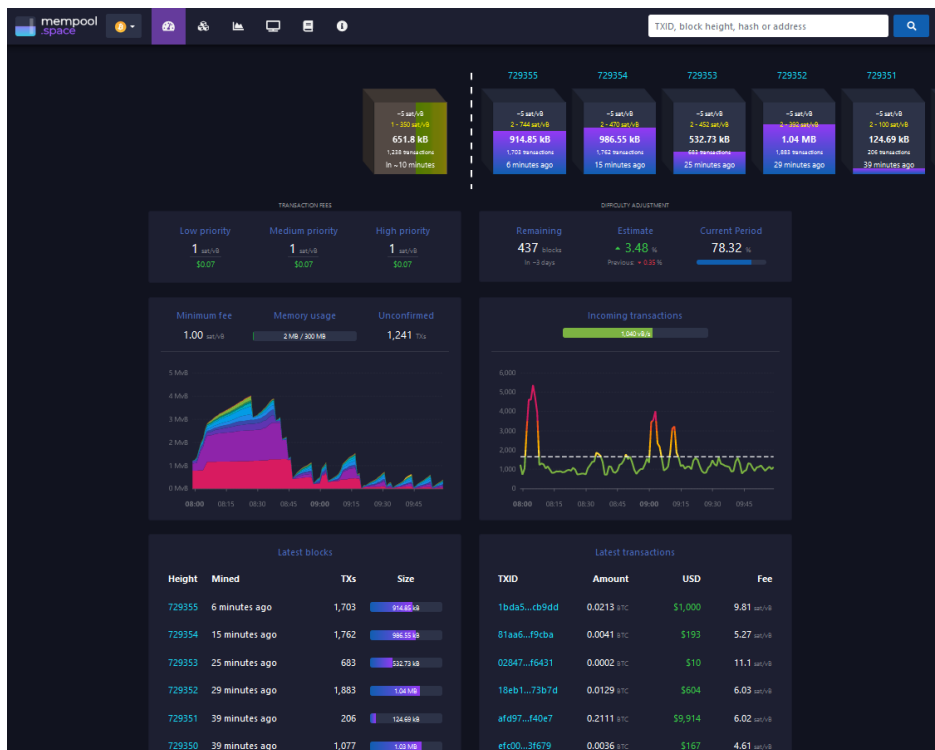
16656 Reachable nodes
10402 Average
9414 ▲ 129.99% Since 7 years ago

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.



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



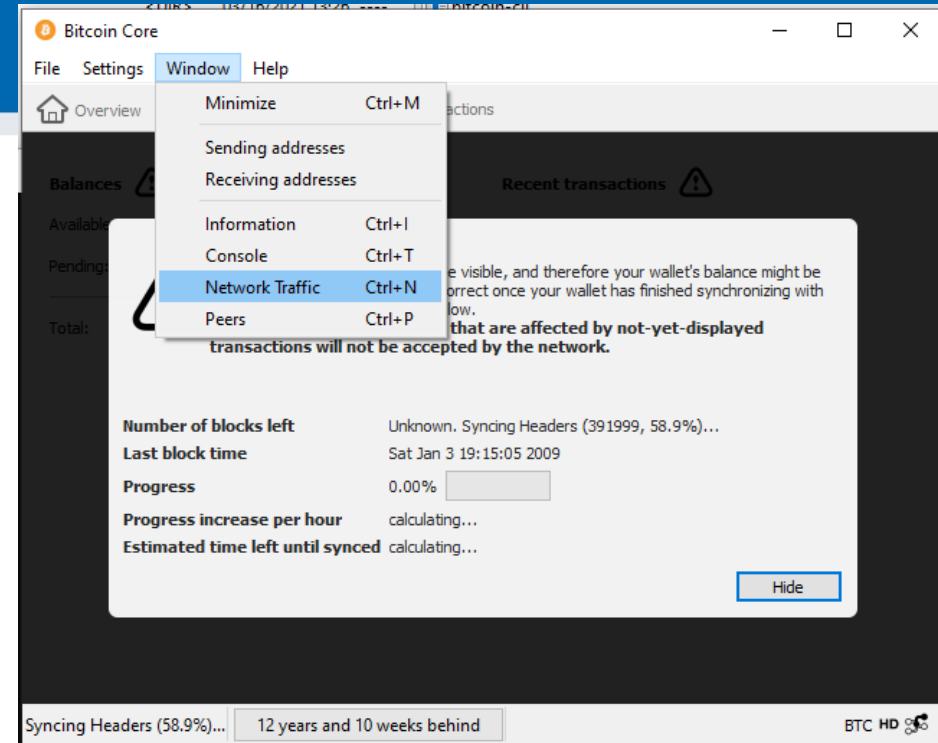
- Can be run on your own fullnode (privacy improvement)
- Testnet version <https://mempool.space/testnet>

INTRO

TASK: USING BITCOIN CORE

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/blk000000.dat (e.g., C:/Bitcoin/blocks/blk000000.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 own full node?
- How can you compute the current supply of bitcoins?

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

File Edit Options Encoding Help

.....;ú ϕ^2 z{.z},>gvĤa■L. |êèQ2:■-K.^
J)½ I¼+ | M.EThe Times &

3/Jan/2009 Chancellor on brink of second bailout for banks

è²■UH'.g±²qθ₁.\r¿(α⁹.²ybαΩ.a■I÷⁴?Lⁿ8-≤U.σ.⊥.■\8M².ìWèLp
.....oΓî.⊥|r⊥²óF«C²0ô.âeßZ.£h.....ÿ Q².K⁰D₁⁴h...e.g{íú|T.²■

.....b0.....

⓪SQ£rj,æµ.⊥...«.ÉÜ:b|f√ïτö{µ<R┘uë7ò.↳α^a.°...üµ"ör.f┘b.s¿,┘#B┘┘

...H`δ... □~öÉⁿèBu.Ao+QY½âhÄÜâ....²|T.% █.zZ█φ≥HX- f\█6○tNΣ

.....a.....

[PR(Σ , L, \sim , $\tilde{U}_{\frac{1}{2}}$, \neg , \exists) $\models \mu \dot{\text{E}} d i j 0 . 8 R 7 ! q \rightarrow \# d F . \frac{1}{2} y \acute{a} \ll A * \square 1 k w \frac{1}{4}$.

..Ö²ú¥í.+.]p.î.ô{¼ïkc. b]...D÷r"É+].r≥√■.û.ç»{σ]√]íu

© M 2

)∞.■ òñ|_`Ö:óÇf-..#-ëqöái.'&t■.Ptsî.u_>5P«çθ.o<_¼..

DFòb«., t^U Ñ5α.o>@ | L≥² úëU. - é...z.Ωÿ-@ | .2ê&+ (cî∞ S7+E j>>^ φ^LΘσó

.....

.fhc\$. .æσ₁. .ñ. .ï. .a0. i. L. . . +07|ñ. . . 1. . . k. |²C7>7«1á¹nLñ. q¼.

äHÄ;ì".ï"Ÿ...ê°ó.ôÉ€U■N....ò.H■—τ%.!ä=#7.Ü8.æ¼\«ê.ç€ö-E(Rc


























TASK: USING BITCOIN-CLI (REGTEST)

Note: Assumed version 24.0.1

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

Note: on Windows, if you use PowerShell, you must prepend .\ before executable (.bitcoin-cli instead of bitcoin-cli)

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`

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)



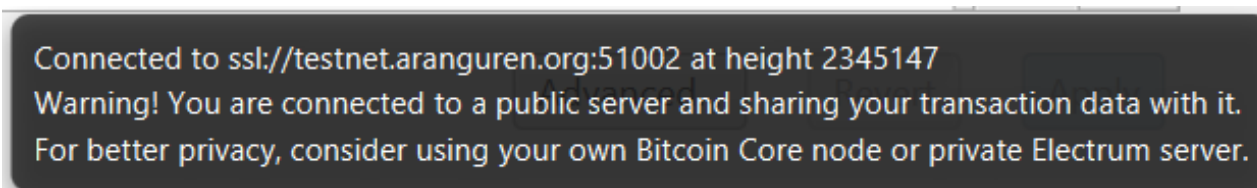
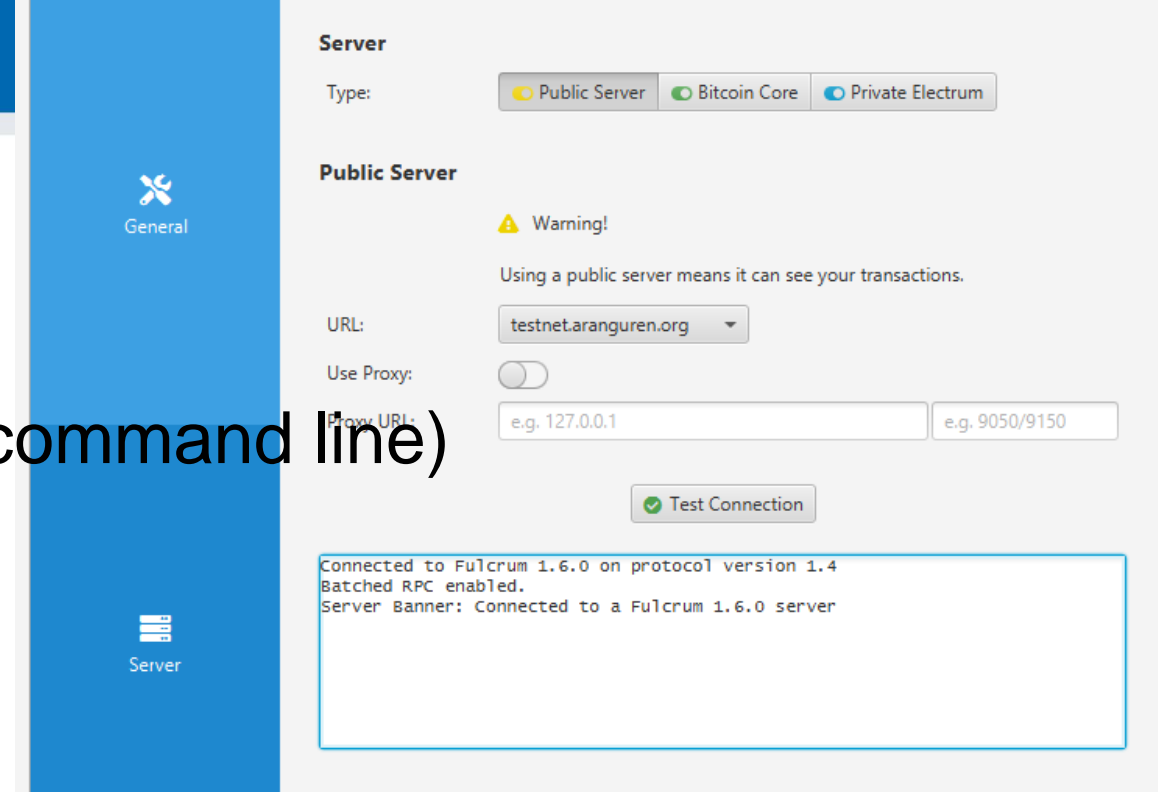
Sparrow wallet (v1.6.6)

- <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)

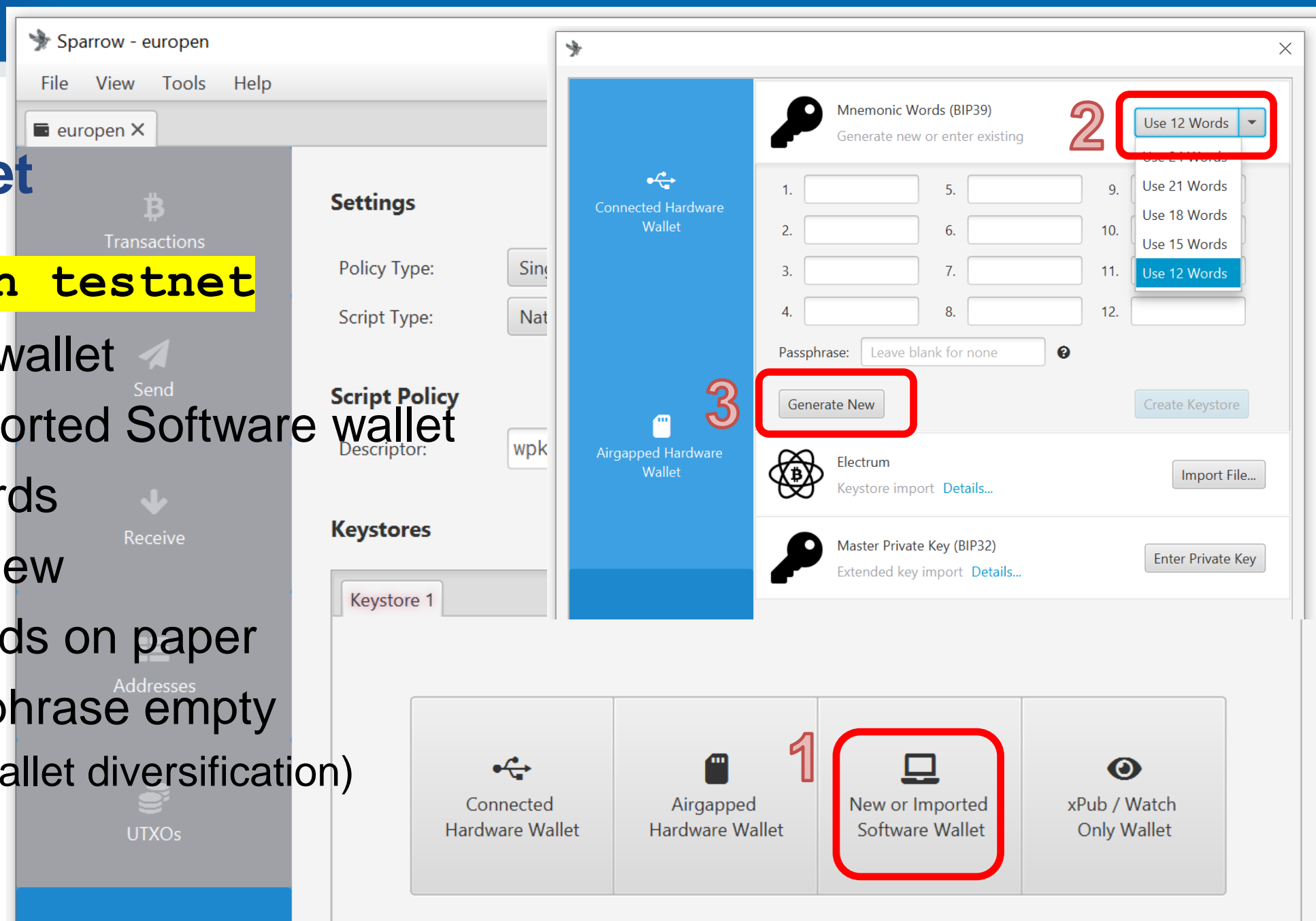
Starting Sparrow wallet

- 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)



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

The screenshot shows the 'Create Keystore' step in the Bitcoin Core wallet creation process. On the left, there are two options: 'Connected Hardware Wallet' (selected) and 'Airgapped Hardware Wallet'. The main area displays 'Mnemonic Words (BIP39)' with a grid of 12 words: 1. ginger, 2. endless, 3. clarify, 4. abuse, 5. explain, 6. trim, 7. draw, 8. usage, 9. gasp, 10. uncle, 11. solid, 12. top. A 'Passphrase' field is set to 'Leave blank for none'. A green checkmark indicates 'Valid checksum'. A red box highlights the 'Create Keystore' button, with a red number '4' next to it. Below the mnemonic words, there is an 'Electrum Keystore import' section with a 'Details...' link and an 'Import File...' button.

The screenshot shows the 'Import Keystore' step in the Bitcoin Core wallet creation process. On the left, there are two options: 'Connected Hardware Wallet' (selected) and 'Airgapped Hardware Wallet'. The main area displays 'Mnemonic Words (BIP39)' with the text 'Ready to import'. A red box highlights the 'Import Keystore' button, with a red number '5' next to it. Below this, there is a text field containing 'm/84'/1'/0' and an 'Import Custom Derivation Keystore' button. At the bottom, there is an 'Electrum' section with a button. To the right of the main window, a partial view of the 'Enter Private Key' window is visible.

Create wallet

6. Apply

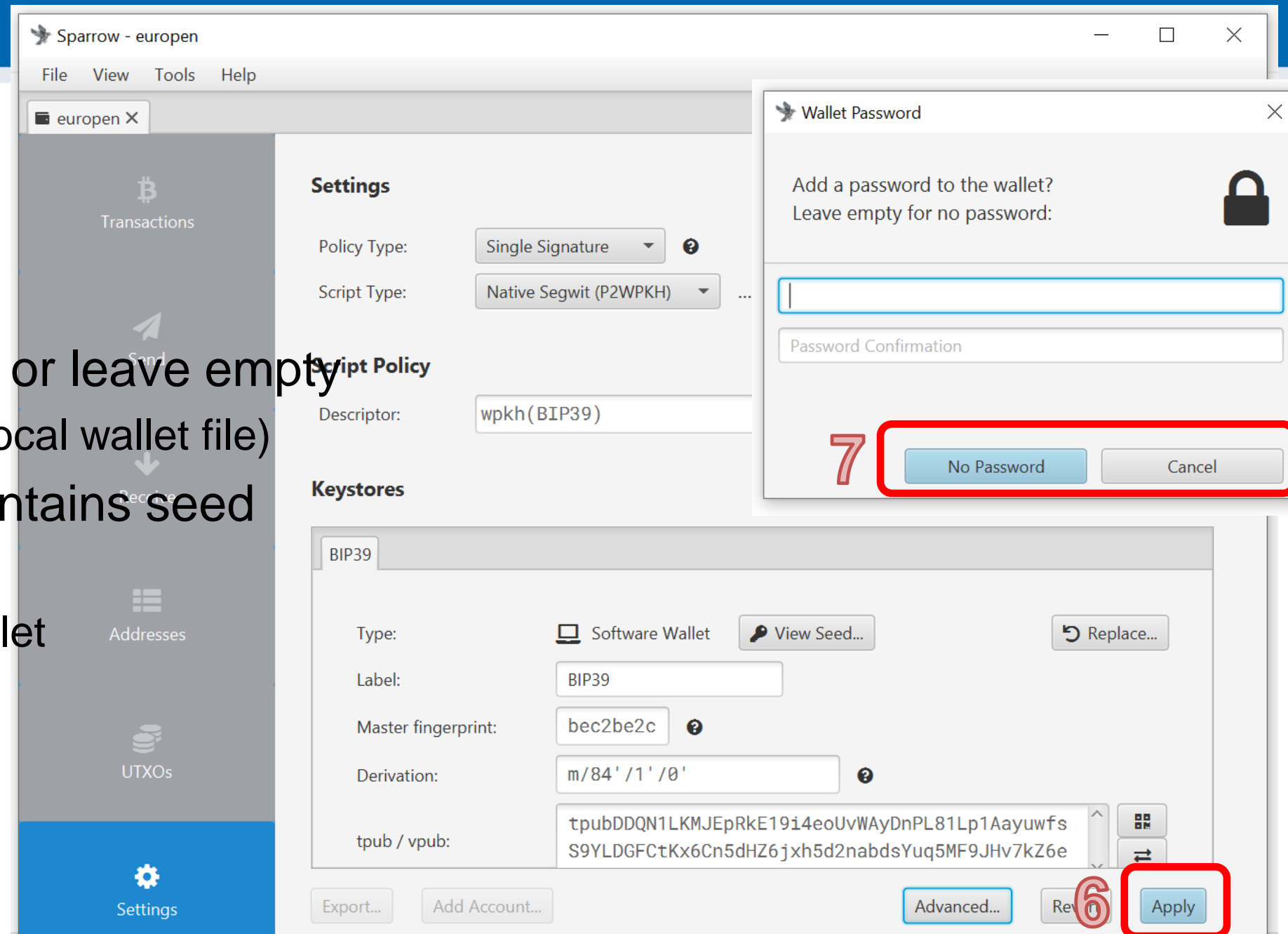
7. Set password or leave empty

– (encryption of local wallet file)

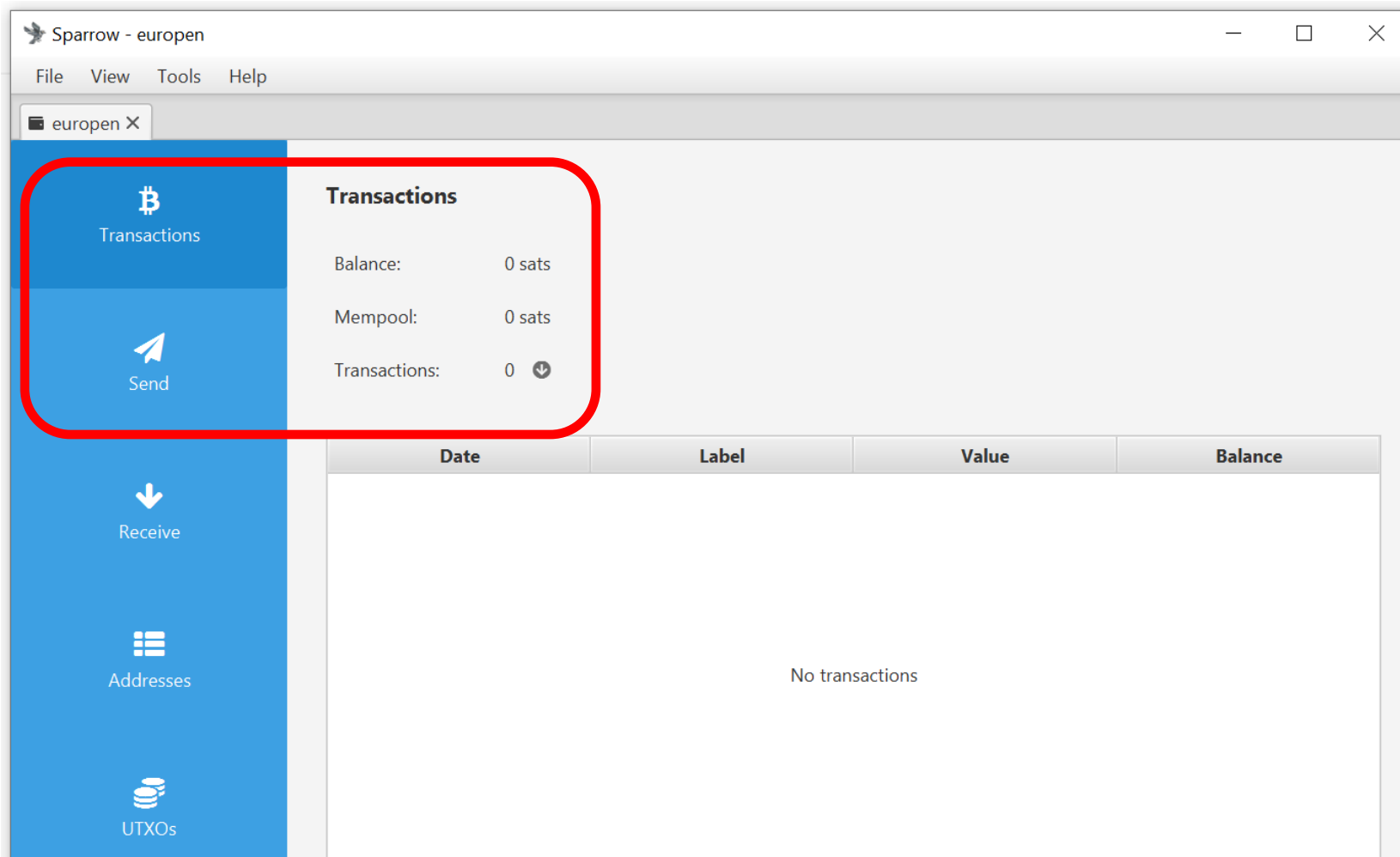
• Local wallet contains seed

– *.mv.db file

– File→Open wallet



Wallet created (but empty 😊)



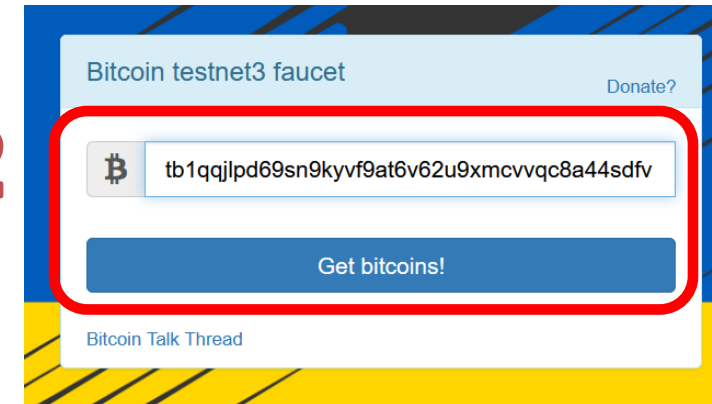
Getting test bitcoins (tBTC)

- 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

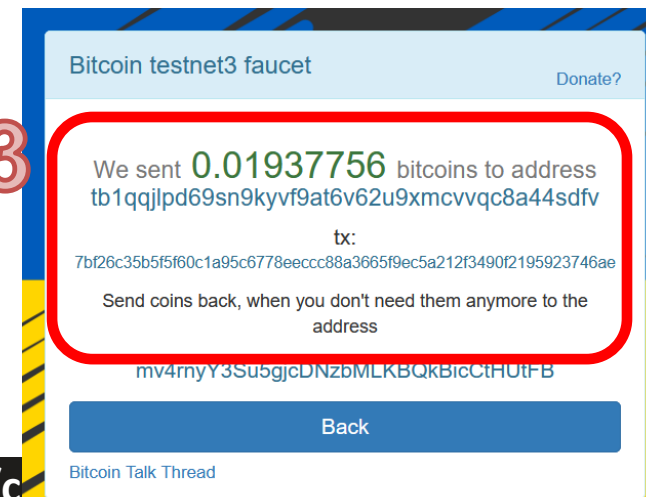
1



2



3





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

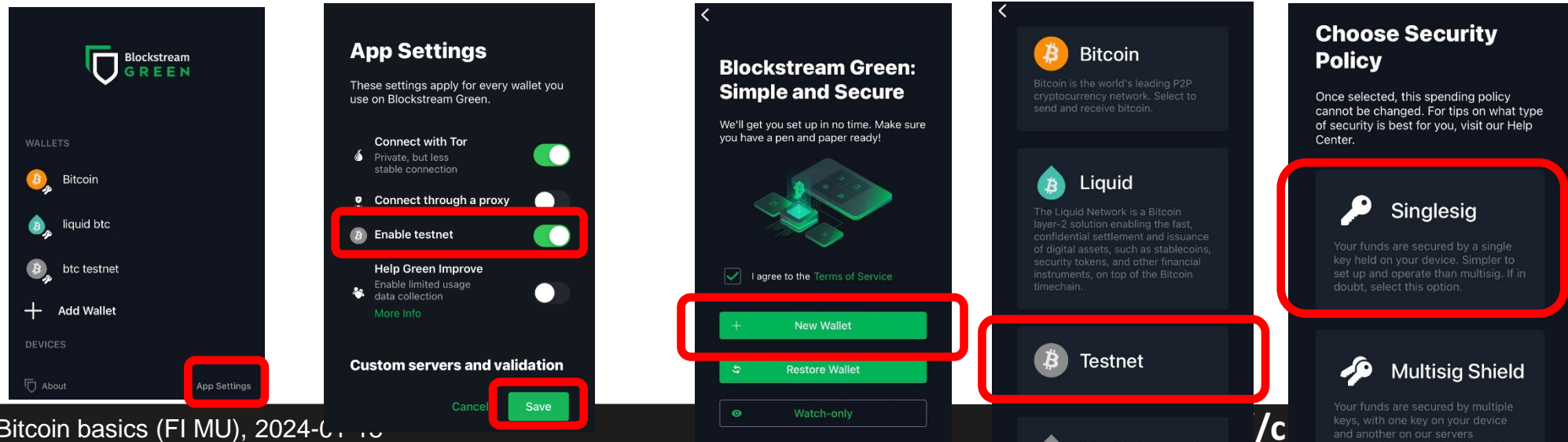
The screenshot shows the Bitcoin Core GUI on PC1. The 'Send' tab is selected and highlighted with a red rectangle. The 'Pay to' field contains the address `tb1qz2qgh3x0kf5vlg8vekcaawuavr1e2z2qd0ru9s`. The 'Amount' is set to 123,000 sats, which is equivalent to \$42.62. The 'Fee' is set to 141 sats, or \$0.05. The 'Rate' is 1.01 sats/vB, and the 'Priority' is set to 'High Priority'. The 'Range' slider is set to 1. The 'Optimize' section shows 'Efficiency' and 'Privacy' options, with 'Analysis...' also available. The 'Create Transaction' button is visible at the bottom right.

PC2

The screenshot shows the Bitcoin Core GUI on PC2. The 'Receive' tab is selected and highlighted with a red rectangle. The 'Address' field contains the address `tb1qz2qgh3x0kf5vlg8vekcaawuavr1e2z2qd0ru9s`. A QR code is displayed next to the address field. The 'Label' field is empty. The 'Last Used' status is 'Unknown'. The 'Required ScriptPubKey' section shows the script `OP_0 <wpkh>`.

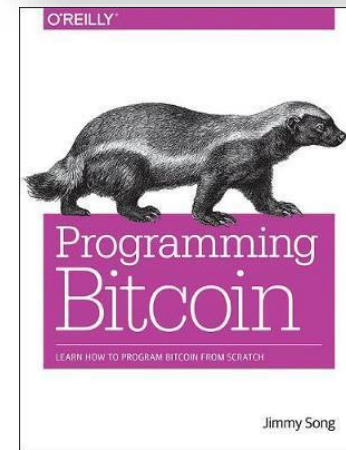
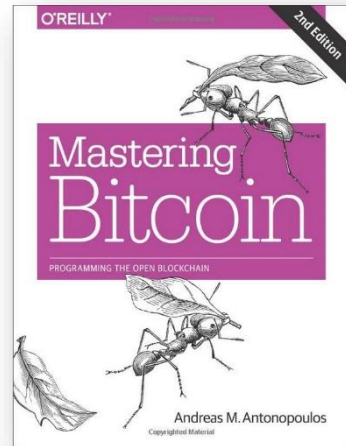
Get mobile wallet

- 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 Green and Sparrow



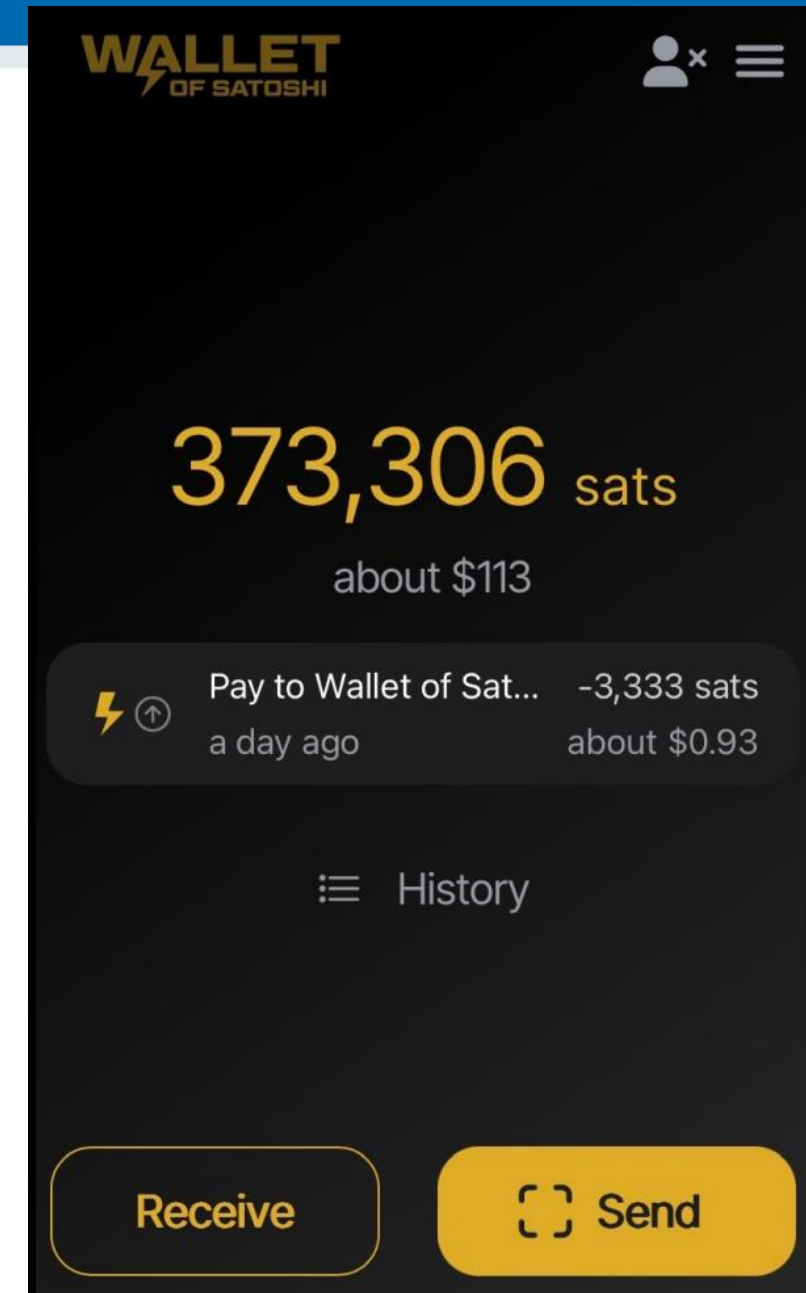
Further reading

- Mastering Bitcoin (Andreas M. Antonopoulos and others)
 - <https://github.com/bitcoinbook/bitcoinbook>
- Programming Bitcoin (Jimmy Song)
 - <https://github.com/jimmy song/programmingbitcoin>
- 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 ₿)

- You can get/buy fraction of bitcoin (sats)
 - Transaction on mainnet
 - potentially costly, ~10mins to execute
 - Mainnet is not for buying coffee!
 - sats 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





Task: send some lighting sats to your peer

- (Assumption: you already some sats on Lighting wallet)
- Try to send between friends
 - Receiver click on 'Receive' button
 - Sender click on 'Send' button, scan QRCode, edit amount, confirm
- Enjoy instant payment
- Inspect Payment Detail (Time, Amount, Total Fees)

