Current Topics in Bitcoin

2018-01-18 Jonas Nick jonasd.nick@gmail.com https://nickler.ninja @n1ckler

Peer-to-Peer Cash

- Ideal: Internet money without central control and anonymous

```
I've been working on a new electronic cash system that's fully
peer-to-peer, with no trusted third party.

[...]

Satoshi Nakamoto

---------------------------------------------------------------------
The Cryptography Mailing List
```

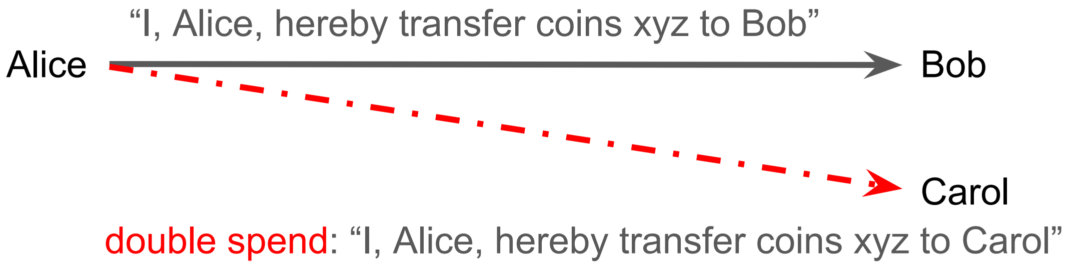
Why? Resist state control

- In practice: failed previous attempts
- It’s digital, global, open to anyone, no registration, no KYC
- No trusted third party
    - programmable money
    - censorship resistance
    - permissionless innovation
    - maximum robustness
    - uncorruptable
- The software is free, anyone can understand, modify and improve it

	Bitcoin	Tulip bulbs
Can be stored in your mind with a mnemonic seed - making sure no authority can seize them?	✓	✗
Capped supply. Once reached - no one can ever generate more?	✓	✗
Costs roughly 1000 USD in electricity to generate one unit - giving the creators an incentive to not sell for less?	✓	✗
Traded on a worldwide market, and can digitally be transfered to anyone on earth?	✓	✗
Was created as a solution to a problem that scientists were unable to solve for decades?	✓	✗
Might morph into a flower?	✗	✓

A toy currency

- Start with arbitrary bits that you call coins from now on
- Use cryptographic signatures to make forging messages impossible



“I, Alice, hereby transfer coins xyz to Bob”

Alice

Bob

Carol

double spend: “I, Alice, hereby transfer coins xyz to Carol”

- A central bank could tell which transaction came first.

A toy currency

- Decentralize control: Shared ledger
  - Every participant keeps a record of the transaction history
  - This works as long you know all the participants and trust a majority.
- But in open peer-to-peer systems
  - It is impossible to know all the participants.
  - It is impossible to meaningfully count votes.
- Want: dynamic membership of the participant set

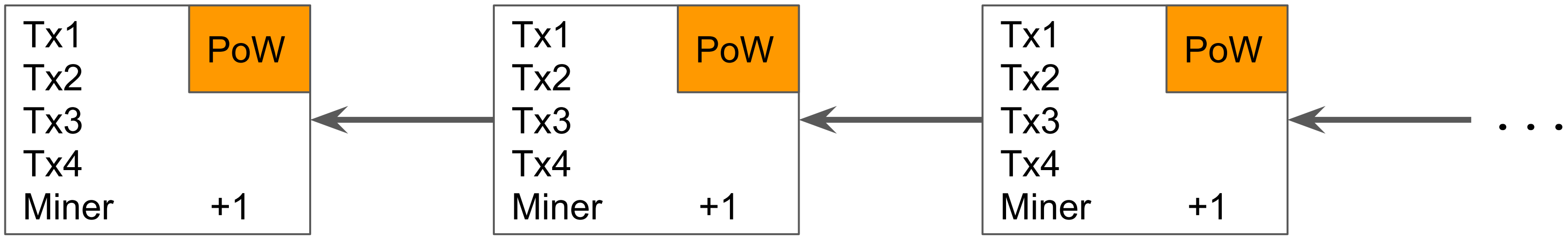
Bitcoin

- *Proof of Work*: small proof that some computation was done

1. A *transaction history* is a list of valid transactions
2. A Bitcoin node uses the *history* with the most proof of work
3. Providing PoW (mining) to a history is rewarded with coins in that history

Mining

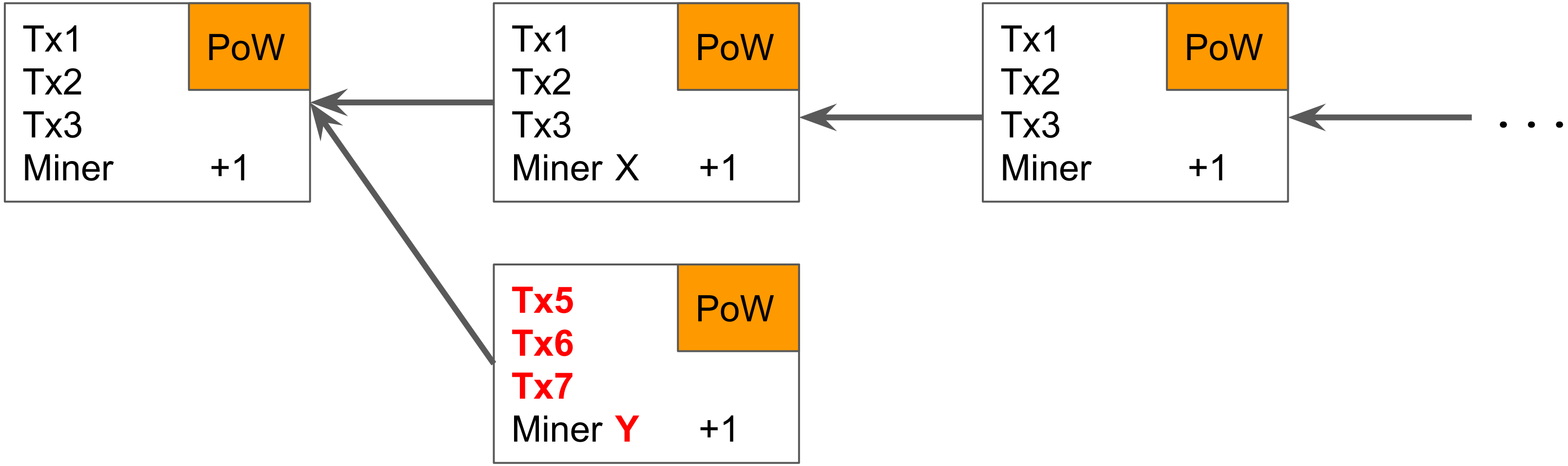
- History is represented as a chain of blocks.
  - Blocks contain transactions.
- Miners create blocks by collecting transactions.
- And attempt to solve the PoW function.
- Blocks are mined on expectancy every 10 minutes.
- The miner gets a mining reward.



Bitcoin

- *Proof of Work*: small proof that some compution was done

1. A *transaction history* is a list of valid transactions
2. A Bitcoin node uses the *history* with the most proof of work
3. Providing PoW (mining) to a history is rewarded with coins in that history



Bitcoin

- *Proof of Work*: small proof that some compution was done

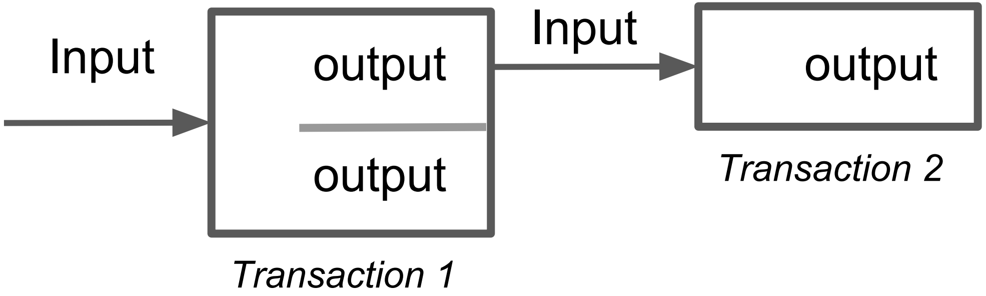
1. A *transaction history* is a list of valid transactions
2. A Bitcoin node uses the *history* with the most proof of work
3. Providing PoW (mining) to a history is rewarded with coins in that history

Effect:

- Consensus on a history.
- Incentivizes mining on a history.
- Incentivizes mining on the history with the most proof of work.

Basic Concepts

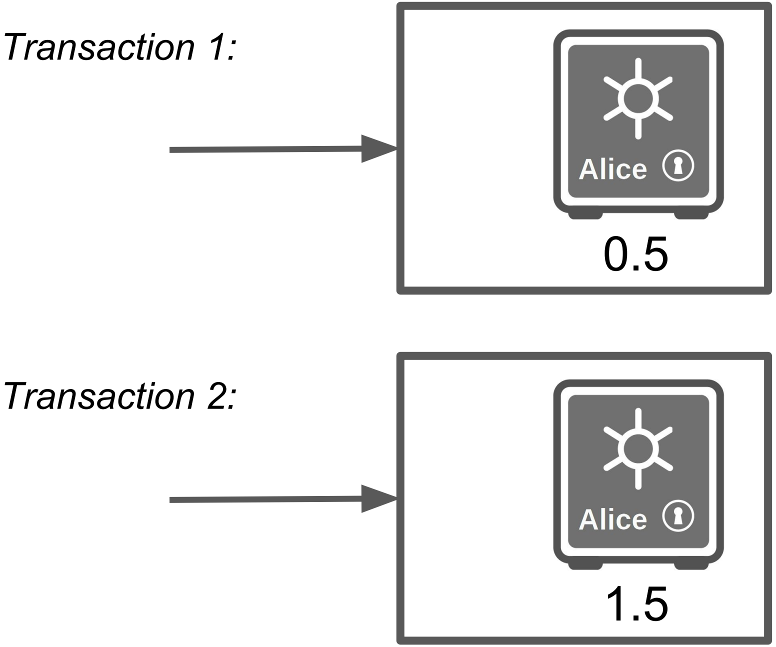
Transactions Inputs & Outputs



Transaction   output: tuple of recipient and value
              input: tuple of txid, vout and signature

Unspent Transaction Outputs (UTXOs)

- Alice owns 2 coins = Alice can spend transaction outputs whose values sum to 2



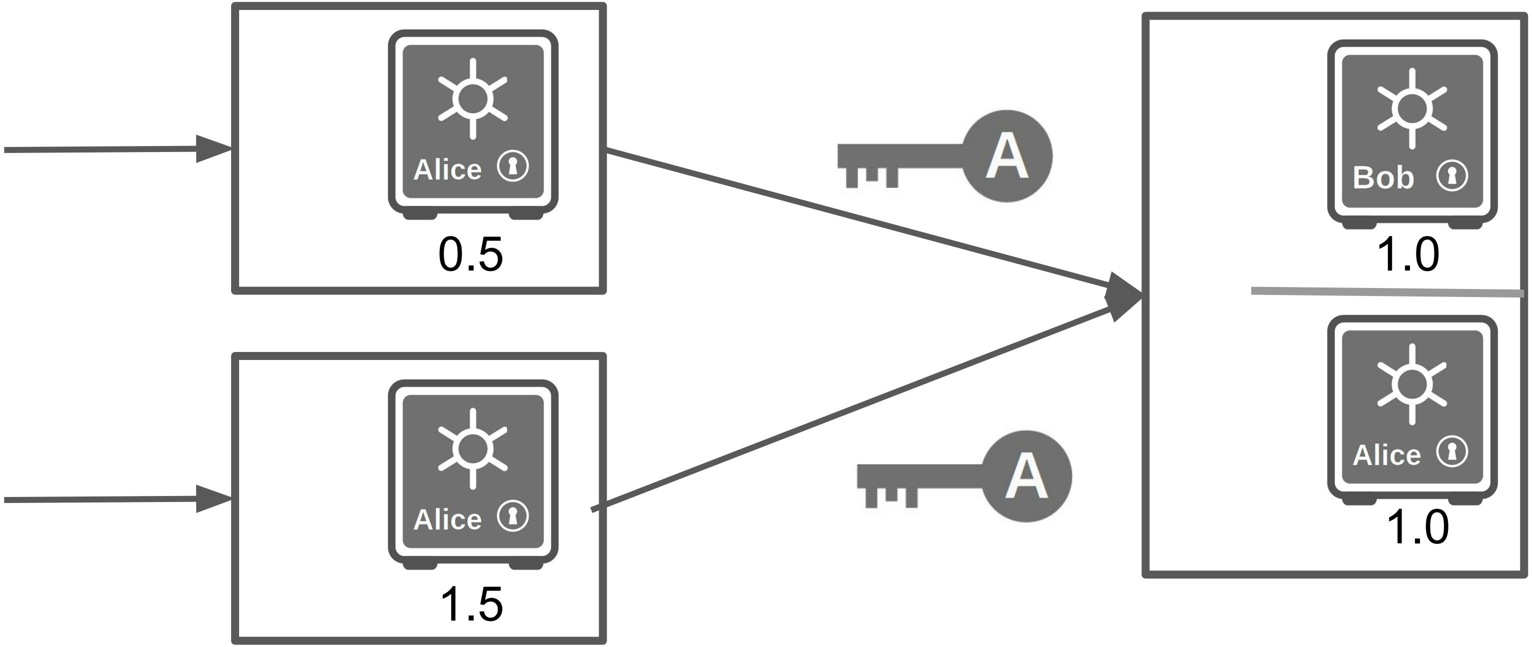
Transaction 1:

0.5

Transaction 2:

1.5

Spending Outputs



Script Evaluation: Pay-to-pubkey (P2PK)

`<pubKey> OP_CHECKSIG`

`<sig>`

+

= `<sig> <pubKey> OP_CHECKSIG`

= `true`

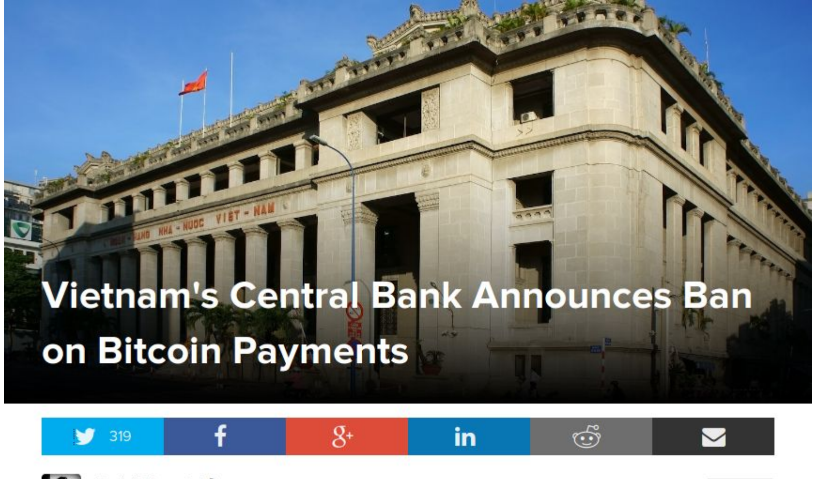
Multisig

2 of 3 Multisig Output

Use cases: Wallet security, Escrow, Micropayment Channels

scriptPubKey: `<m> <pubKey_1> … <pubKey_n> <n> OP_CHECKMULTISIG`
scriptSig: `<sig_1> … <sig_m>`

Current Topics

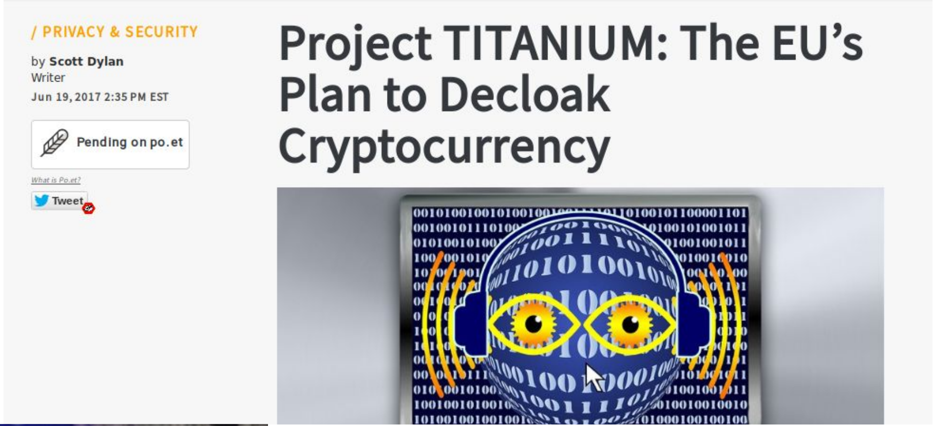


Vietnam's Central Bank Announces Ban on Bitcoin Payments

Daniel Palmer
Oct 31, 2017 at 15:10 UTC

Vietnam's central bank is prohibiting the use

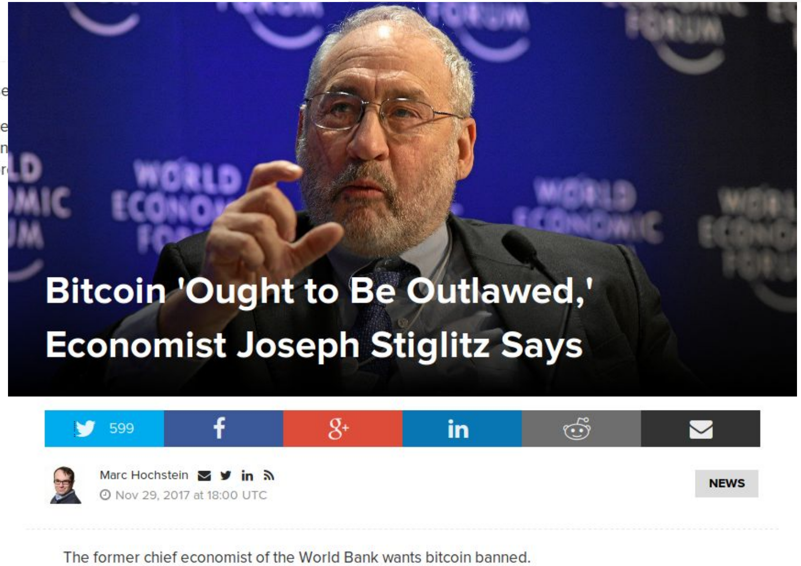
According to an Oct. 30 statement, the State
"lawful means of payment" in the country, an
virtual currency as a means of payment is pr



/ PRIVACY & SECURITY

by **Scott Dylan**
Writer
Jun 19, 2017 2:35 PM EST

Project TITANIUM: The EU's Plan to Decloak Cryptocurrency

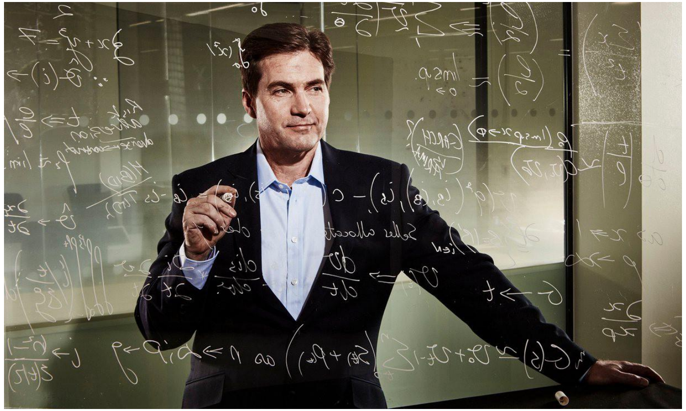


Bitcoin 'Ought to Be Outlawed,' Economist Joseph Stiglitz Says

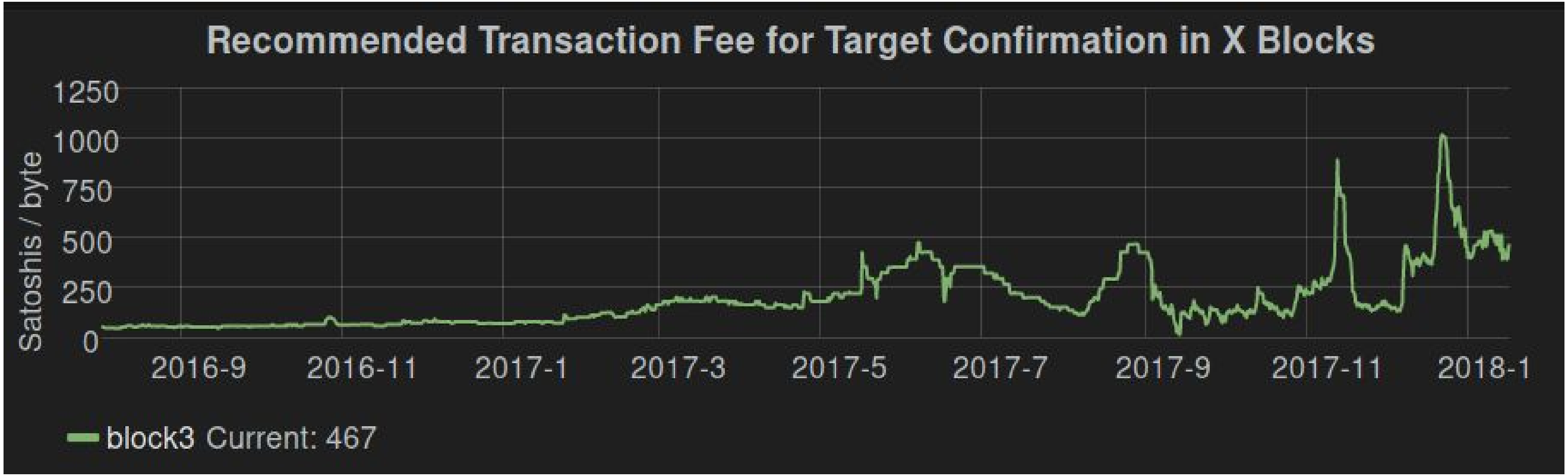
Marc Hochstein
Nov 29, 2017 at 18:00 UTC

NEWS

The former chief economist of the World Bank wants bitcoin banned.

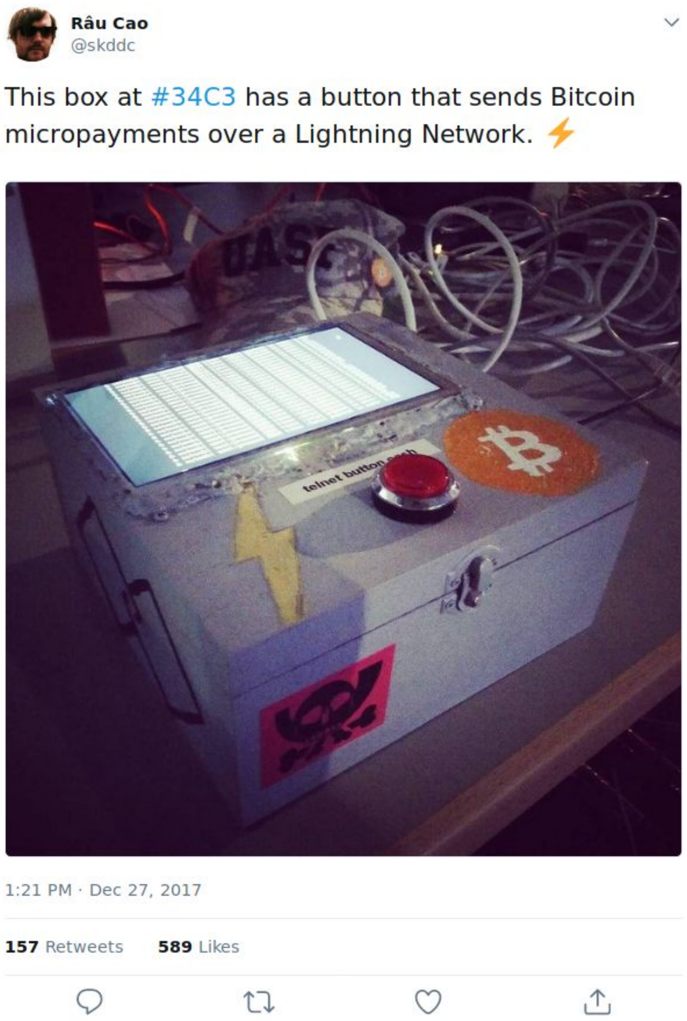


Fake Satoshi



statoshi.info

Average (segwit) transaction: 6.3 EUR (at 10,000 EUR/BTC)



Râu Cao
@skddc

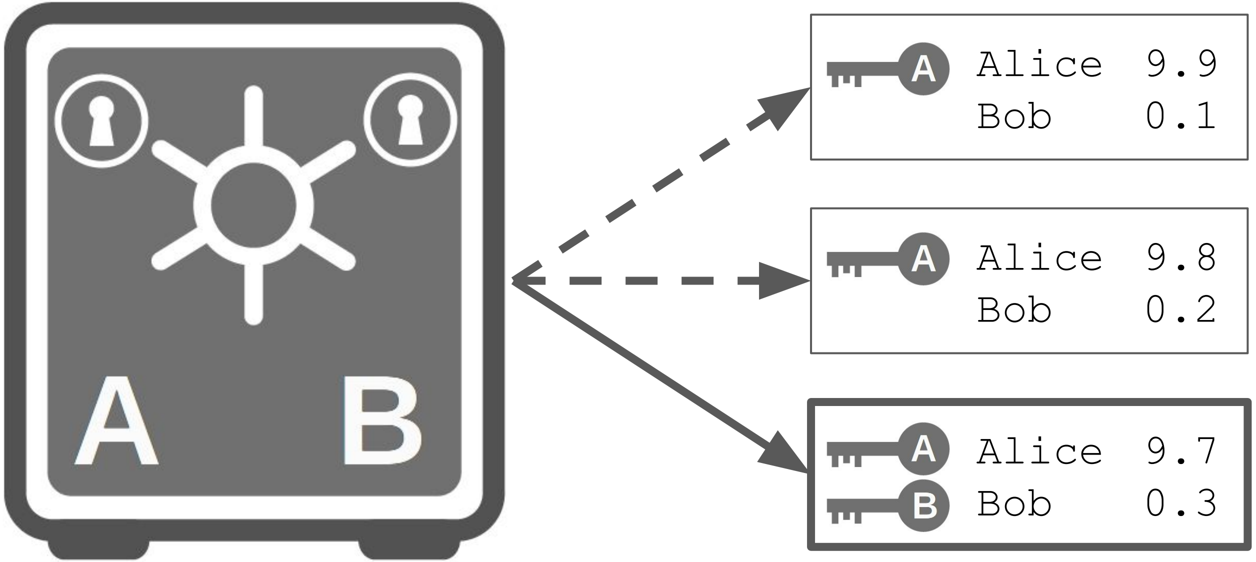
This box at #34C3 has a button that sends Bitcoin micropayments over a Lightning Network. ⚡

1:21 PM · Dec 27, 2017

157 Retweets **589** Likes

Payment Channels

Setup: Alice creates transaction with 10 bitcoin to a 2-of-2 multisig with Bob



Alice 9.9
Bob 0.1

Alice 9.8
Bob 0.2

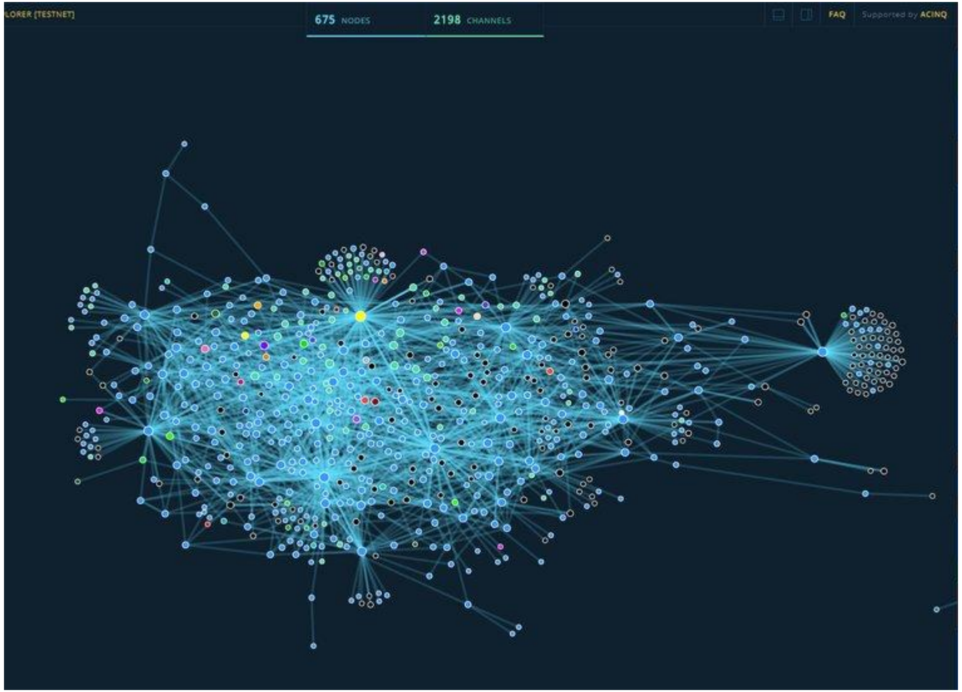
Alice 9.7
Bob 0.3

Alice pays by signing tx and sending it directly to Bob

Closing the Channel:
Bob signs tx and broadcast to miners

Lightning Protocl

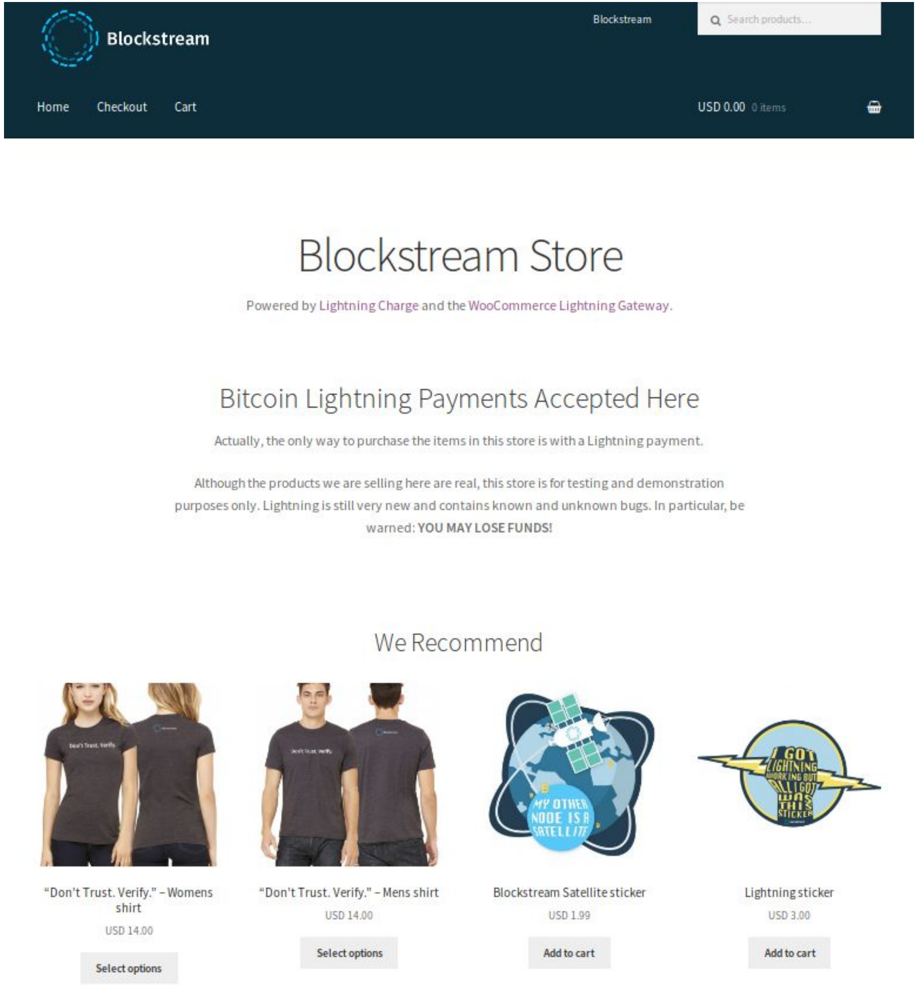
- Lightning = payment channels + routing



https://explorer.acinq.co

Lightning

- Lightning = payment channels + routing
- Payment flow:
  - 1st payment: open a direct channel with the merchant: 1 Bitcoin transaction
  - N-th payment: use the lightning network to route the payment: No transaction
  - When capacity exceeded: close the channel
- c-lightning operations
  - Create channel: `fundchannel <peer_id> <amount>`
  - Receiver: `invoice`
  - Sender: `pay <invoice>`
  - Close channel close `<peer_id>`
- Low fees, micro payments, instant confirmations
- Status: Spec finalized, running on testnet, UX iterations, lots of PoCs are created



Blockstream Store

Powered by Lightning Charge and the WooCommerce Lightning Gateway.

Bitcoin Lightning Payments Accepted Here

Actually, the only way to purchase the items in this store is with a Lightning payment.

Although the products we are selling here are real, this store is for testing and demonstration purposes only. Lightning is still very new and contains known and unknown bugs. In particular, be warned: **YOU MAY LOSE FUNDS!**

We Recommend

"Don't Trust. Verify." – Womens shirt
USD 14.00
Select options

"Don't Trust. Verify." – Mens shirt
USD 14.00
Select options

Blockstream Satellite sticker
USD 1.99
Add to cart

Lightning sticker
USD 3.00
Add to cart

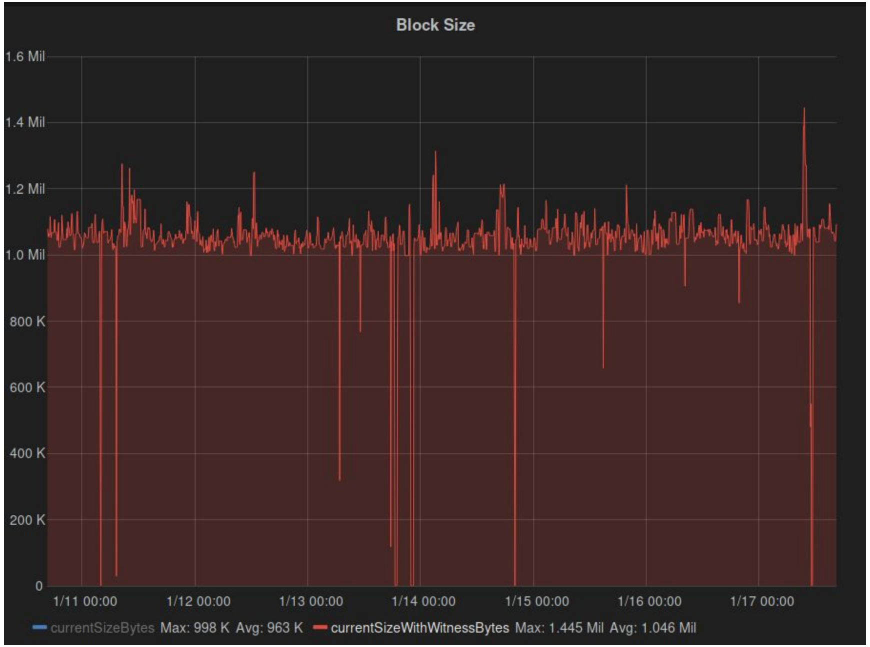
Don't Trust. Verify.
MY OTHER NODE IS A SATELLITE
I GOT LIGHTNING WORKING BUT ALL I GOT WAS THIS STICKER

store.blockstream.com

github.com/ElementsProject/lightning-charge

Segregated witness (Segwit)

- malleability fix and block size increase activated last year



statoshi.info

Native segwit transactions

- “Native”: Change transaction format to reduce size
- Goes along with address format change
  - old: `1FJJdX5g1DX7FRxJBhJNTDrRjTeihhsJLs`
  - bech32: `bc1qnntcclssmtuvfw2te7q49lzvw67cfvpzxger4j`
  - Why? Easier to type and pronounce, better error detection
- Status: is being rolled out

Schnorr Signatures

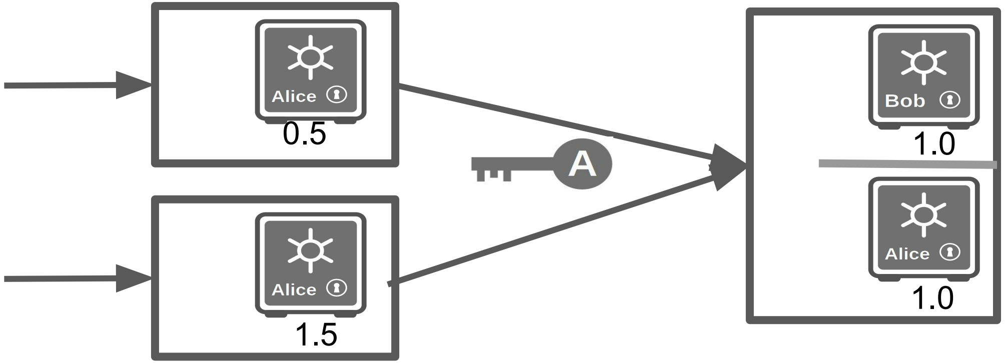
- Different signature scheme, right now it’s ECDSA
- Simpler algorithm and stronger security proof, but was patented
- Allows batch verification, scriptless scripts (key aggregation) and signature aggregation

Schnorr Signatures: Key Aggregation

- n-of-n OP_CHECKMULTISIG
  - scriptPubKey: <n> <pubKey_1> … <pubkey_n> <n> OP_CHECKMULTISIG
  - scriptSig: <sig_1> … <sig_m>
- OP_SCHNORR
  - Idea (simplified)
    - Pubkey = pubkey_1 + pubkey_2 + … pubkey_n
    - Sig = sig_1 + sig_2 … + sig_n
  - scriptPubKey: <pubKey> OP_SCHNORR
  - scriptSig: <sig>
- Result: saves space, looks like any other payment
- Generalization: scriptless scripts
  - allows more smart contracts in crypto-currencies that don’t have any native smart contract support (lightning, atomic swaps)

Schnorr Signatures: Signature Aggregation

- Rolled out with Schnorr signatures
- Allows adding up unrelated signatures
- Result is creating one signature per transaction instead of one per input
- Reduces transaction size, Incentivizes coinjoin



Merkelized abstract syntax tree (MAST)

- Given a script with branches (OP_IF …. OP_ELSE … OP_ENDIF)
  - For example cooperative vs. uncooperative case in Lightning
- Only state the branches that are executed

MAST + Key aggregation

- Lightning script before
  - scriptPubKey: `OP_IF 2 <pubkey_1> <pubkey_2> 2 OP_CHECKMULTISIG OP_ELSE` … `OP_ENDIF`
  - scriptSig: 1 <sig_1> <sig_2>
- Lightning script now
  - scriptPubKey: <merkleroot> OP_MERKLEBRANCHVERIFY
  - scriptSig: <sig> <`<pubkey> OP_SCHNORR> <merkleproof>`
- Result: smaller and looks like any other payment

Confidential Transactions

- Hides amounts in transactions
  - Verifier: input_amounts = output_amounts
  - Verifier: Enc(input_amounts) = Enc(output_amounts)
- Used in elementsproject.org sidechain, Monero, Mimblewimble
- Allows for Confidential Assets
- Feasibility of Bitcoin softfork?
- Bulletproofs: reduce size massively

Conclusion

- Bitcoin is a peer to peer currency
- Run your own full node
- Proof of Work isn’t going away any time soon
- Lots novel of research, engineering and experimentation is happening
- **Do something!**

- Slides: https://nickler.ninja/slides/2018-Frankfurt.pdf

2018-01-18 Jonas Nick jonasd.nick@gmail.com https://nickler.ninja @n1ckler