Bitcoin Mechanics

Transactions, Mining, and PUSH-DOWN AUTOMATA?!

Overview

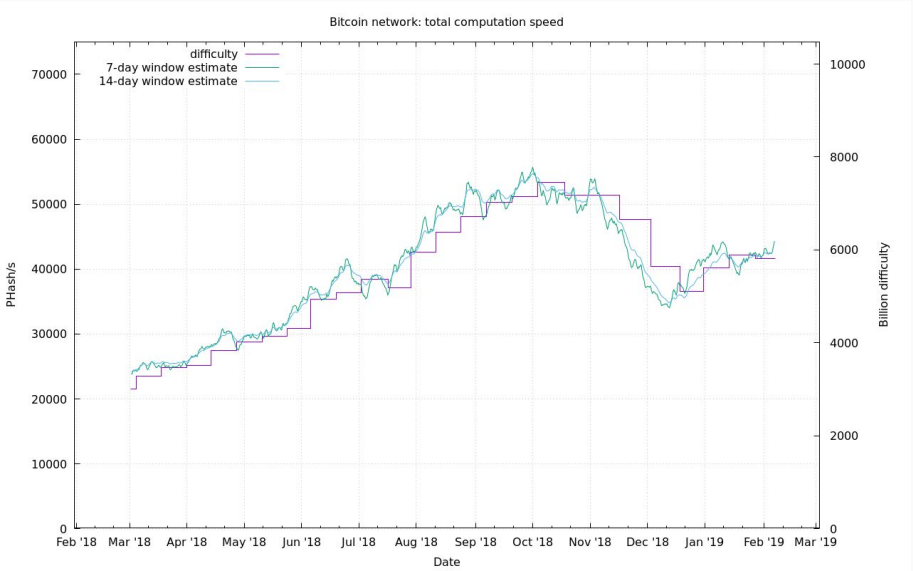
- Blocks
  - Mining & Incentives
- Transactions
- Scripts
- Wallets

News?

Mining

Mining & Incentives

- Difficulty
  - Adaptive threshold
- Block rewards
  - “Coinbase” address
- Transaction Fees



http://bitcoin.sipa.be/

Mining & Incentives

256-bit hash function H, x from domain, i from range

- Outputs?
- P[H(x) == i]?
- P[H(x) < i]?

Mining & Incentives

Block Rewards

- “Coinbase” transaction
- 50 BTC, halved periodically
- Currently 12.5, 6.25 in ~May 2020

Mining & Incentives

Transaction Fees

- Unspent inputs of transactions
- Can miners modify transactions?

Blocks

Block Structure

Header

Transactions

Block Structure

Header

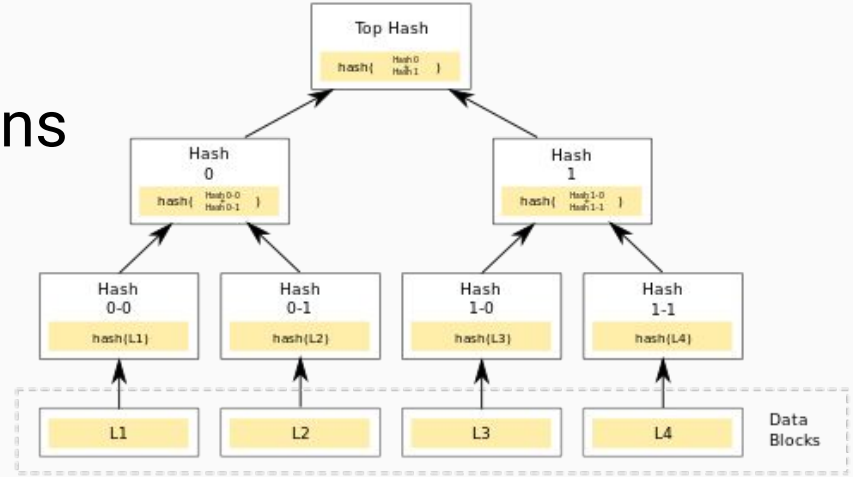
- Proof of Work hash
- Hash of previous block
- Time
- Nonce
- Merkle root
- ...

Transactions

Block Structure

Merkle Root

- Bind block to all transactions
- Easily verify single transactions



Transactions

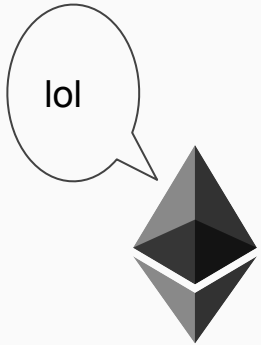
Transactions

- Header
- Inputs
  - Hash
  - scriptSig
- Outputs
  - Value
  - scriptPubKey
- Witnesses
- Lock time

Scripts

Scripts

- scriptSig and scriptPubKey
- Stack Machine (PDA)
  - Data, Opcodes
  - No loops
- Not Turing-complete



Scripts

scriptSig

- In Input
- Sign(Hash(tx)), PubKey
- Combined w/prev output script

Scripts

scriptPubKey

- In Output
- Verify signature of spender
- Combined with some future scriptSig

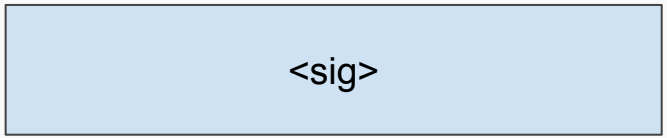
Scripts

Pay to PubKey Hash

- scriptPubKey: OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG
- scriptSig: <sig> <pubKey>

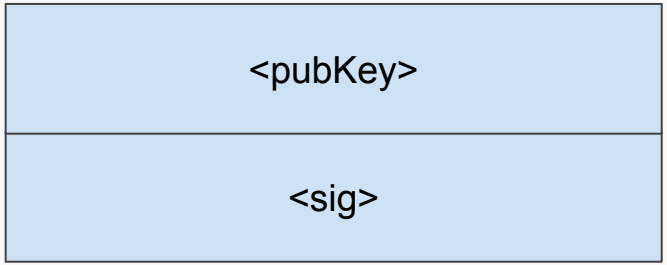
<sig>

Signed hash of a subset of the transaction



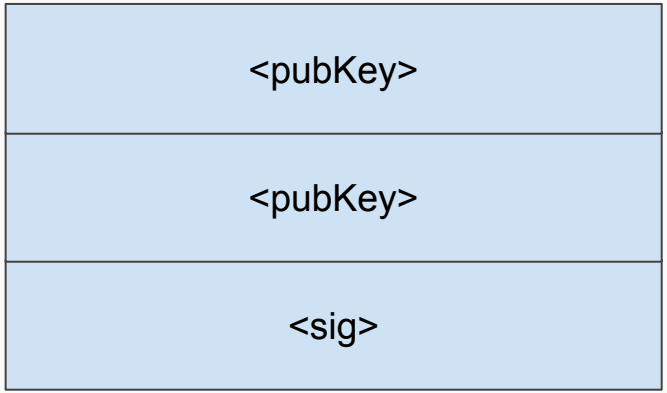
<pubKey>

Public key corresponding to the signature provided



OP_DUP

Duplicate top of the stack



OP_HASH160

Pop, and push hash of popped data

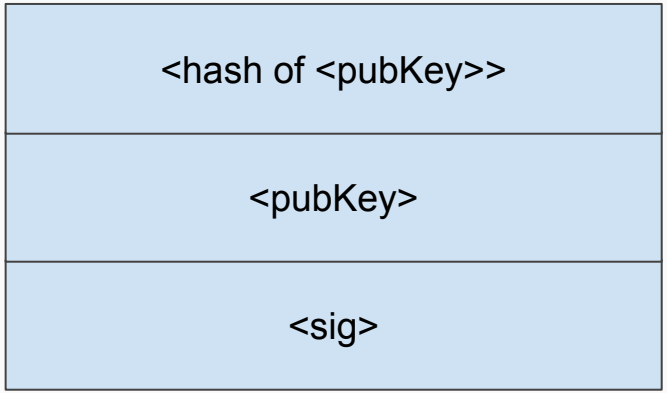
RIPEMD-160(

SHA256(

Data

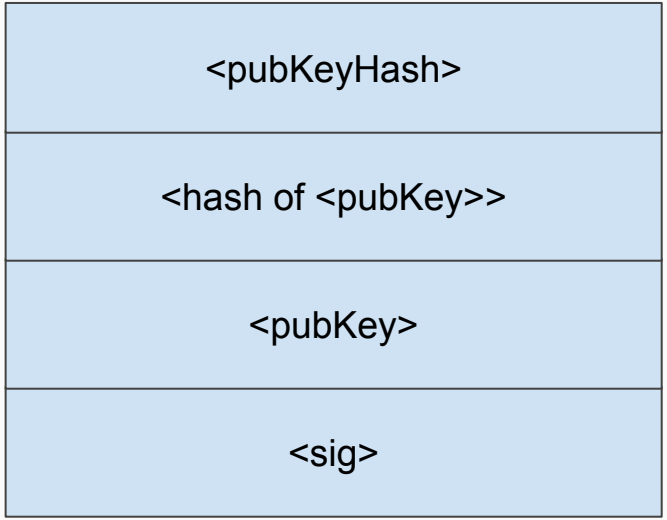
)

)



<pubKeyHash>

Public key hash provided by previous output

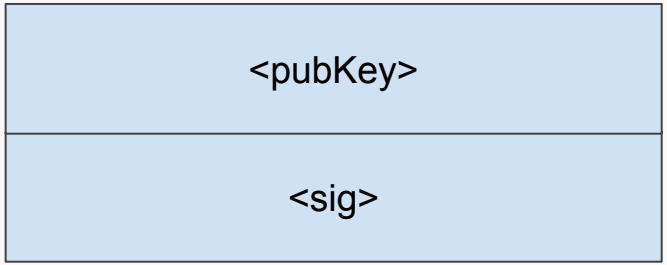


OP_EQUALVERIFY

OP_EQUAL and OP_VERIFY in one

Pop two items

If not equal, mark invalid



OP_CHECKSIG

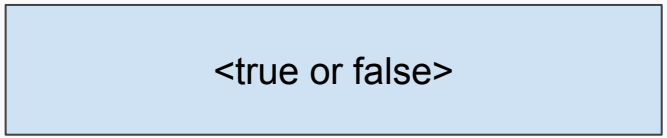
Hash transaction

Pop public key

Pop sig

Verify sig

Compare to hash



Scripts

Pay to Script Hash

- scriptPubKey: OP_HASH160 <hash> OP_EQUAL
- scriptSig: <arbitrary script>
- E.g. Time-lock, Multi-sig, SHA1 collision, OP_RETURN

Wallets

Wallets

- Store private keys
- Fetch public keys
- Generate transactions
- Communicate w/network

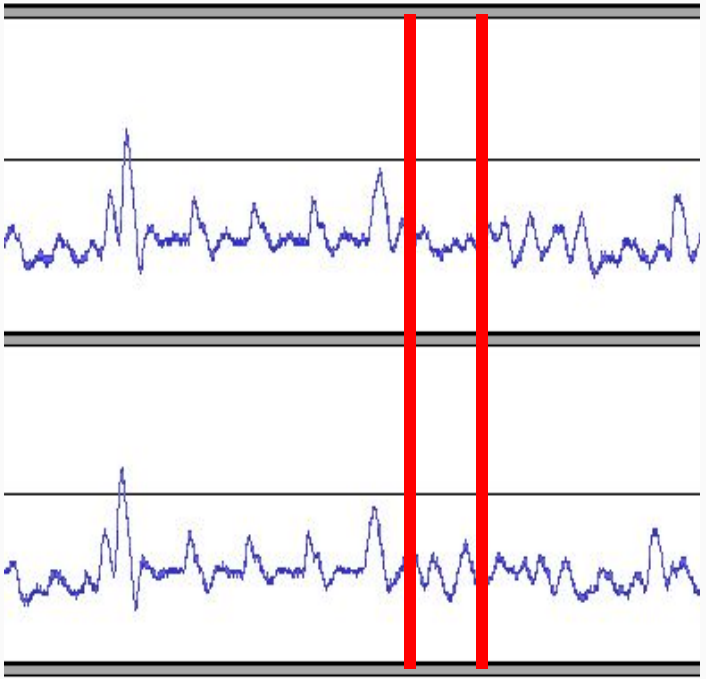
Wallets

- “Hot” wallet
- “Warm”
- “Cold”/offline

Wallets

HSM/Hardware Wallet

- Trezor side channel attack (2015)



Overview

- Hash of previous links the chain
- PoW+Incentives drive consensus
- Merkle root binds all transactions to a block
- scriptSig and scriptPubKey are executed
- SegWit prevents malleability