A Drivechain BIP enabling the OP_COUNT_ACKS opcode to add Bitcoin drivechain capabilities as a soft-fork.

BUILDING ON BITCOIN Conference
Lisboa, Portugal, 3-4 July 2018

Sergio Demian Lerner
Chief Scientist, RSK Labs

www.rsk.co
About me  🍦 @SDLerner

Lic. Sergio Demian Lerner
Computer Security Researcher,
Chief Scientist, RSK Labs Inc.
<table>
<thead>
<tr>
<th>Code</th>
<th>Product</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>Bitcoin Core</td>
<td>Lack of orphan tx limit prior v0.5.3</td>
</tr>
<tr>
<td>CVE-2012-3789</td>
<td>Bitcoin Core</td>
<td>Multiple DoS Vulnerabilities in Satoshi client</td>
</tr>
<tr>
<td>CVE-2012-4683</td>
<td>Bitcoin Core</td>
<td>Targeted DoS by CPU exhaustion using alerts</td>
</tr>
<tr>
<td>CVE-2012-4684</td>
<td>Bitcoin Core</td>
<td>Network-wide DoS using malleable signatures in alerts</td>
</tr>
<tr>
<td>CVE-2013-2272</td>
<td>Bitcoin Core</td>
<td>Remote discovery of node's wallet addresses</td>
</tr>
<tr>
<td>CVE-2013-2292</td>
<td>Bitcoin Protocol</td>
<td>A transaction that takes 3 minutes to verify using $O(n^2)$ hashing</td>
</tr>
<tr>
<td>CVE-2013-2293</td>
<td>Bitcoin Core</td>
<td>Continuous hard disk seek</td>
</tr>
<tr>
<td>2014</td>
<td>BitcoinJ</td>
<td>Security vulnerability in BouncyCastle ECDSA (BJB-22)</td>
</tr>
<tr>
<td>2014</td>
<td>Ethereum/Bitcoin</td>
<td>Unhandled point-at-infinity in public key recovery</td>
</tr>
<tr>
<td>2016</td>
<td>Bitcoin protocol</td>
<td>A Bitcoin transaction that takes at least 5 hours to verify</td>
</tr>
<tr>
<td>2016</td>
<td>Ethereum</td>
<td>Uncle Mining, an Ethereum Consensus Protocol Flaw</td>
</tr>
<tr>
<td>CVE-2017-12842</td>
<td>Bitcoin protocol</td>
<td>Leaf-Node weakness in Bitcoin Merkle Tree Design</td>
</tr>
</tbody>
</table>
Bitcoin & Radical Innovation

Confidential Transactions  Faster confirmation

Onchain space  Stateful smart-contracts
Where the new transactions go?

• Overlay protocols
• Extension blocks
• Parallel blockchains
  (now generically called sidechains)

} Preserve the 10 minute block interval
Increases block size in the same network
Sidechains: who controls the locked funds?

• Consensus-enforced (original SPV sidechains)
• Federation
• Miners (drivechains/Hashrate Escrow)
• Hybrids
RSK blockchain

• Uses Smart Bitcoin as its native currency
• Provides stateful smart-contracts and 15-secs block times.
• 21% of Bitcoin’s hashing rate (in merge-mining)
• 2-way (1:1) peg with global federation
• Soon to deploy custom and auditable HSMs for federators
• 2-way peg controlled by smart-contract
• Next release: intelligent HSMs that validate PoW, real-world delays and generate time-locked transactions with covenants.
SPV Sidechain

- Bitcoin
  - lock btc
  - tx mined
  - confirmation period
  - unlock btc
  - tx mined
  - Reorg period
  - Verification of SPV proofs (sidechain)

- Secondary blockchain
  - Verification of SPV proofs (sidechain)
  - reorg period
  - unlock secoins
  - tx mined
  - lock secoins
  - tx mined
  - confirmation period

Legend:
- Green: Affected blocks in secoins -> BTC transfer
- Pink: Affected blocks in BTC-> secoins transfer
The RSK Case

CountAcks

Drivechain in the future?
Drivechain / Hashrate Escrow

Bitcoin
Blockchain

Sidechain
I must signal the sidechain wants to execute transaction T.
Drivechain / Hashrate Escrow

I saw the command too. Signal it!

H(T)

Command:
Immediately Release of Funds using transaction T
Now because of our signals, T has become valid. Let’s include it!
ACKs and NACKs

- "ACK:" following FULL_ACK_LIST
- FULL_ACK_LIST: { CHAIN_ACK_LIST... }
- CHAIN_ACK_LIST: { sidechain_id ACK_LIST }
- ACK_LIST: { ACK... }
- ACK: { tx_hash_prefix [ tx_hash_preimage ] }
- {} = empty list
ACKs and NACKs

• ACK: { { XNET { {} 0x101010....10 } } } Proposal and ack in XNET
  h(0x10....10)=0xbaa501b37267c06d8d20f316622f90a3e343e9e730771f2ce2e314b794e31853)
• ACK: { { XNET { {0xba} } } } 2nd positive ack for the proposal
• ACK: { { XNET {} } } negative ack for all proposals in XNET
• ACK: { {XNET {} } { YNET { {0x3e9e7307} } } } Mix for 2 sidechains

• Note: serialization is binary, not ASCII.
The opcode has the following arguments:

- Poll_start_blocknum
- sidechain_id
- ack_period (in blocks)
- delay_period (in blocks)
- liveness_period (in blocks)

**OP_COUNT_ACKS**

- Ack_period: Currently max 288
- Delay Period: Currently max 12960, Min 100 blocks
- liveness_period: Currently max 288
OP_COUNT_ACKS

- The opcode results:
  - ACK count
  - NACK count

Currently max 288
Sample ScriptPub / ScriptSig (no P2SH / P2WSH)

ScriptSig: 521000
ScriptPub:

58 4e 45 54  // ("XNET")
144
1440
144

OP_COUNT_ACKS  // Push Results
OP_2DUP  // duplicate ack counts
OP_GREATER_THAN // more positive than negative acks?
OP_VERIFY  // abort if not
OP_SUB  // compute positive minus negative, push result into stack
72  // difference (positive-negative) acks required
OP_GREATER_THAN // More than 50% positive difference, put 1 on stack, else put 0
Sample script: Drivechain + 2 notaries

• ScriptPub:

0 OP_TOALTSTACK
OP_IF <pubkey1>
OP_FROMALTSTACK OP_ADD
OP_TOALTSTACK OP_ENDIF
OP_IF <pubkey2>
OP_FROMALTSTACK OP_ADD
OP_TOALTSTACK OP_ENDIF

58434f494e 144 144
OP_COUNT_ACKS OP_SWAP
OP_FROMALTSTACK OP_ADD
OP_DUP OP_ADD OP_DUP OP_ADD
OP_DUP OP_ADD OP_ADD
OP_SWAP OP_2DUP
OP_GREATERTHAN OP_VERIFY
OP_SUB 72 OP_GREATERTHAN

• ScriptSig: 1 <Signature1> 1 <Signature2> 500000

• Condition: x=(4 * sig + acks), then (x > naks) and (x-naks > 72)
Mandatory Delays & Chances to Revert in RSK

1. **RSK User commands fund release**
2. **RSK Smart-contract receives command and forces delay**
3. **RSK Smart-contract buids tx commands sign tx**
4. **Federation sign transaction**

**Drivechain**
- Miners acknowledge during a poll period and forces delay
- Acks threshold reached. COUNT_ACKS forces delay

- **Transaction is included in block**
- **CheckSequenceVerify output forced delay with covenants**
- **Funds transfer enabled**
- **User use funds**
CountAcks Design Rationale

- Lightweight soft-fork
- Interoperability with scripting system
- Zero risk of invalidating a block
- No additional computation during blockchain management and re-org.
- Incentive compatible: sidechain pays for withdrawal cost
- No inherent change in Bitcoin security model
- Bounded computation of poll results (2 sigops cost)
- Strong protection from DoS attacks
- Minimum block space consumption (800 bytes per withdrawal typical)
- Zero risk of cross-secondary chain invalidation
- Time for proactive and reactive measures (up to 90 days)
Comparison between CountAcks BIP and Hashrate Escrows BIP memory use

<table>
<thead>
<tr>
<th>Property</th>
<th>CountAcks</th>
<th>Hashrate Escrows</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lines of code</td>
<td>~600</td>
<td>~4000</td>
</tr>
<tr>
<td>Initial sidechain registration (in DB)</td>
<td>0</td>
<td>125 Kbytes</td>
</tr>
<tr>
<td>Withdrawal (max blockchain space)</td>
<td>3 Kbytes</td>
<td>157 Kbytes</td>
</tr>
<tr>
<td>Withdrawal (avg blockchain space)</td>
<td>864 bytes</td>
<td>157 Kbytes</td>
</tr>
</tbody>
</table>

Sources:
https://github.com/drivechain-project/docs/blob/master/bip1-hashrate-escrow.md
https://github.com/rsksmart/bips/blob/master/BIP-R11.md
New BIP and reference implementation

https://github.com/rsksmart/bips/blob/master/BIP-R11.md

https://github.com/rootstock/bitcoin/tree/op-countacks_devel
Summary

• Bitcoin federated sidechains have risks of federators stealing the locked funds

• Adding a CountAcks drivechain layer miners prevent federators malicious activity

• You can use also use a pure CountAcks sidechain.

www.RSK.co

@SDLerner | @RSKSmart
Interoperability

- COUNT_ACKS opcode allow the combination of a drivechain with any other feature of the scripting system.

- Allows to bootstrap a merged-mining two-way pegged cryptocurrency from an initial state when is has no merge-mining engagement to a state where it has a high merge-mining engagement, using notary signatures during the initial period.

- scriptPub can be parametrized for any combination
Zero risk of block invalidation

- The opcode and miner's ack-ing algorithm was designed such that acks in the coinbase field can never invalidate a block.
- This prevents attacks against pools from malicious or faulty proxy consensus observer plug-ins.
- Reduces the risk for miners not implementing the soft-fork of extending a soft-forked block that is invalid because of the coinbase tag.
Minimum Computation and Incentive compatibility

• No blockchain computation overhead if there is no sidechain activity
• Sidechain pays for every cycle of computation
Bitcoin security model

- poll liveness period to be equal or higher than 100 blocks, to respect the same maturity rule as coinbases (enables urgent community hard-forks)
- Any blockchain that uses the bitcoin unit of account and holds a high amount of bitcoins could affect the security of Bitcoin.
- Also merge-mining can modify the incentives of Bitcoin miners, and those incentives should be analyzed.
Time for proactive and reactive measures

• 2 days max for polls allow humans to detect corrupted or hacked miners and warn to stop acknowledge process.
• 30 days before transaction becomes valid prevents from massive dishonest miners behavior.
• 2 days of liveness enables publication even if miners interest decrease significantly.
Bounded computation of poll results

• The liveness period and ack period have maximum values (currently 4320 blocks, or one month).

• Benefits:
  • sets a bound to opcode running time
  • is compatible with blockchain pruning

• Still to cache one months of tags requires 1.3 Mbytes top
Strong protection from DoS attacks

- Polls created for unknown sidechains can be safely ignored by miners.
- Unknown or fake transaction candidates do interfere with honest candidates and are automatically negatively acknowledged.
Minimum block space consumption

• Transaction id prefixes for candidates could reduce space in average to 2 bytes per ACK.

• Pre-image publication prevents prefix collusion to force miners to use full ids.

• For example, if 100% of the miners acknowledge a proposal for 100 blocks then the space consumption would be ~ 234 bytes/proposal.

• Cloinbase space allows 12 sidechains making 4 withdrawals per day each (or one sidechain making 50).
Zero risk of cross-sidechain invalidation

• Sizes in bytes.
• Easy skip if inner tag is malformed.
• Miner may collect sidechain acks in serialized format without risk of interaction.
Security

• The security parameters of a specific sidechain are defined by the sidechain designers.

• Exodus addresses should be pay-to-witness-script-hash (P2WSH) address containing all arguments.

• There COUNT_ACKS opcode cannot be used as a vector to perform a denial-of-service attacks (CPU, memory, disk access)

• Sidechain designers should be able to choose between long pre-inclusion delays or long post-inclusion covenants.
Computational Cost

• The cost of the COUNT_ACKS opcode in terms of sigops is set to 2 (a maximum of 288 blocks are scanned).

• The maximum amount of information that has to be fetched is 12 Kbytes.

• Assumes in-memory cache (maximum 500 Kbytes, typically 3 Kbytes).

• Max cost in hashing of tx_hash_preimage to obtain tx_hash is 1440 hash digests. This is comparable to the cost of 2 signature verifications.
Changes from previous proposal (2016)

• Liveness and poll times incremented from 1 day to 2 days
• Variable delay time added of to 3 months of blocks (before it was 100 blocks)
Blind Merge-mining

• Need High sidechain Tx fees
Protections against 51% dishonest miners using Intelligent HSMs

- On-chain release pre-signals
  - with minimum accumulated difficulty
  - Combined other soft-forks
    - Transactions ids that also derive from block hash using a bit in nVersion (finalID = H(blockHash | originalID )
    - Using conditional to block difficulty (OP_DIFFICULTY opcode)
    - Or transactions that can only be anchored only after certain block (OP_BLOCK_HASH_AT opcode)
    - No need to standardize txs using new opcodes
- Covenants through txs with time-locked txs, and return outputs paths