Bitcoin beyond money

p2p cash is one use case.

Justin Moore

March 22, 2019

Insight Decentralized Consensus

Motivation: assume all hardware is insecure

- Even if you follow computer security best practices, you can't be certain your hardware is clean. It's likely that it's fine - but that is an assumption.
- Supply chain is a huge threat vector that most everyday user has never pondered.
  - "The Prototype iPhones That Hackers Use to Research Apple's Most Sensitive Code"
    - E.g: https://motherboard.vice.com/en_us/article/gyakgw/the-prototype-dev-fused-iphones-that-hackers-use-to-research-apple-zero-days
- How do I know the chip in my machine doesn't have a backdoor? Trust the manufacturer?
- How do I know my ASIC miner's streamed-via-the-internet firmware update wasn't mitm'd to do something strange 0.01% of the time it's running (e.g. throw hash power at a particular pool)?
- So the website/webserver is hosting a file with a checksum, but how do I know the site isn't compromised with a compromised checksum?

Bitcoin is secure as far as we know

What's in the chain is in the chain and is immutable. if you wait a little while we can be pretty certain about the integrity / immutability.

I want to write code to the chain using the chain, and my first idea was to use the fact that a satoshi maps to a bit

E.g. 0.11111111 bitcoin can be seen as 0xFF bitcoin in hex. But this would get prohibitively expensive quickly.

Next I thought about encoding/compression

First idea was something like 0xFF could go to .00002222 by stacking. Playing around in this space, but still not quite where it should be to have any meaningful program.

I also thought about creating a set of addresses that contained a bunch of options and then using an ordering of the addresses / transactions outside of the chain to reconstruct the code.

I actually think in the future a special kind of service will be provided (by miners or specialized miners) for searching the chain and remembering where things live. A sort of on-chain PageRank for a small fee instead of a digital replication of you being compiled, packaged, and sold to advertisers.

Breakdown

Impact: global impact and global audience

Pieces, components: bitcoin libraries

How to build ? Start simple, then iterate to include complexity.

Programming Language: Probably C++ or Python, but I'd like to try LISP

Network: Bitcoin (BTC/BCH (try to optimize for small) / BSV [more about searching])

Algorithm: tbd