Distributed Consensus

22 August 2017 03:35 AM

Key Challenge of decentralized e-cash: Distributed Consensus

How to solve the problem of decentralizing scrooge coins

Motivation: Reliability in Distributed Systems
If error in saving data in all Data Bases then we can ask user to re-enter the data
If error in saving data in some if the Dbs then it would lead to in-consistency in data

Applications: Distributed KV store for
DNS, Public Key Directory (Email - Public Key), Stock Trades

Definition :

The protocol terminates and all correct nodes decide on the same value

This value must have been proposed by some correct node

Bitcoin how to achieve consensus:

Eg: Alice broadcasts its pay to bob over the network. The Data Structure has Alice Signature, Public Key of Bob and a Hash. Bob may or may not be in the network.

For Node to reach consensus:

All nodes store block of Txs they reached consensus on
Each Node has a set of outstanding transaction its heard about (Some node may have heard some may not)

Technical Problems:

Nodes may crash
Nodes may be malicious

Network is imperfect
- Not all pair of nodes connected
- Fault in Network
- Latency

Proves impossibility
Byzantine General Problem
Fischer-Lynch-Paterson (deterministic node)

Well know protocol:
Paxos

Never inconsistent but it may get stuck

The results say more about the model than problems

More models were developed to study systems like distributed databases

Bitcoin work better in practice than Theory -- unforeseen attacks

Bitcoins does it differently:

Introduces incentives -- Possible because its currency
Embraces randomness --
Does away with notion of a specific end-point
Consensus happens over long time scale -- 1 hr