There is now a lot of interest in “Hashgraph-the-Blockchain-Killer” in the Silicon Valley investment community, so I have been asked recently to do a review of the Hashgraph whitepaper. I am publishing it here because it may be interesting for some people

My humble opinion after reading Hashgraph whitepaper is that every page potentially contains vulnerabilities, since they have proofs that are not really proofs, theorems that are not theorems and lemmas which are not lemmas…Every object is vaguely defined, so it becomes hard to analyze.

Hashgraph is a directed acyclic graph (similar conceptually to IOTA I guess). Then they need to somehow transform a DAG into a linearly ordered chain, at this moment they are using a terribly flawed time-stamp-based “consensus” as explained, below.

If you go to page 11 of the Hashgraph whitepaper, the paragraph that starts with

Then, the received time is calculated. … The received time for x

is the median of all

such timestamps …

you see that the way they go from a graph to a chain is by taking the median received time T_{median}

for each transaction and then ordering transactions according to T_{median}

.

T_{median}

is a median of the reported received time for each transaction. Since the network is asynchronous and transactions are gossiped different nodes end up receiving transactions at different times.

This is imho the weakest point of their whitepaper for the following reason: if I am a bad guy, I can withhold reporting for a while, wait until the chain is settled and smart contract is executed, and a smartcontract and then report a transaction screwing the entire system.

Lets consider the following example where there are two transactions A and B, there are four good nodes and one malicious node.

The way an attack goes is as follows:

1. The good guys report received time 1.3, 1.4, 1.5, 1.6 for the transaction A and 1.3, 1.42, 1.42, 1.7 for transaction B. The bad guy waits.

2. T_{median}

for A is 1.45 for A and 1.42 for B. So B gets included in the chain before A.

1. A smart contract is executed on the chain.

2. A bad guy comes a minute later, and reports 1.2 for A and 1.7 for B. Now

the median time for A is 1.4 and median time for B is 1.42. So now A goes before B.

1. Since a smart contract has already been executed under assumption that B goes before A , there is a contradiction. This logical contradiction causes an security harm to the chain beyond repair.

The example above is a simple scenario, one can cook up lots of more complex scenarios - basically using time stamps opens up a Pandora box. The only condition under which Hashgraph may be secure is if 100% of the nodes are honest.

Ironically, in 80s and 90s before the seminal paper from MIT (Barbara Liskov) came out, people tried all kinds of variations for time-stamp-based consensus systems and all of them failed.