Byzatine fault tolerance

Xiaojie Zhu

March 20, 2018

Byzatine fault tolerance (BFT) is the dependability of a fault-tolerant computer system, particularly distributed systems, where components may fail and no enough information is available to infer whether a node is failed. In a "Byzantine failure", each component behaves arbitrarily such that different observers may have different symptoms.

Practical byzantine fault tolerance (PBFT) is a form of state machine replication proposed to address the Byzantine failure in practicality. PBFT offers both liveness and safety provided at most $\lfloor (n-1)/3 \rfloor$ out of total of n replicas are simultaneously faulty. It consists of three phases, pre-prepare, prepare and commit. The pre-prepare and prepare phases are used to totally order requests sent in the same view. The prepare and commit phases are used to ensure that committed request are in order across views. Dealing with garbage collection, checkpoint is set and the state proof is added, which is called stable checkpoint. In the PBFT, there are two aspects that can be optimised. One is to reduce communication and another one is to use light cryptography tool, e.g., replace digital signature with message authentication code.

In the Bitcoin, the proof-of-work (PoW) is adopted as consensus protocol. Comparing with BFT replication, PoW has advantage in the degree of decentralisation, scalability of nodes and scalability of clients. However, POW can not achieve consensus finality and has high latency. Moreover, PoW is only able to tolerate less than 25 percent computing power owned by adversary. Contrast with the PoW, BFT is a permissioned protocol and all nodes need to know others. In addition, it has low latency and is able to reach final consensus. The most important property BFT holds is correctness proofs, which distinguishes from PoW.

To overcome the scalability weakness of Blockchain, many new protocols are proposed. The GHOST is proposed to resolve the conflicts in a POW blockchain by weighing the subtrees rooted in blocks rather than the longest chain rooted in given blocks. A variant of the GHOST, GHOST-PoW, is implemented in Ethereum to improvement in throughput. The Bitcoin-NG uses the standard POW for leder election, declaring a node to mine a block with standard difficulty. In addition, directed acyclic graph is applied to maintain blocks

instead of linear chain of blocks. More mechanisms, e.g., eliminating communication and resource overhead, randomized BFT, and mixing PoW and BFT, are proposed to enhance the performance of Pow in Bitcoin.