Proof-of-Stake Consensus in the Doge Protocol Blockchain

Authors: The Doge Protocol Community

Publication Date: November 2021

Revision 2 : August 2024

Table of Contents

Introduction

Doge Protocol will be a Quantum Resistance blockchain that initially will use Proof-of-Stake (PoS) consensus and later move on to a PoW+PoS hybrid consensus scheme. This whitepaper is a summary of the Proof-of-Stake consensus system. A follow-up whitepaper will cover the various parts of the consensus scheme in detail.

Doge Protocol mainnet blockchain uses a variant of 3 phase Practical Byzantine Fault consensus (pBFT).

The uber goal is to build a robust blockchain that's resistant to byzantine failures while at the same time is decentralized and scalable. We shall see the tradeoffs involved as part of this design.

Design Principles

The Doge Protocol consensus system will follow the below design principles.

Abstract

The consensus scheme itself will be abstract, leaving out details like cryptography, hardware, programming language etc. to the implementation.

Security

Secure the network in adversarial conditions while at the same time maintaining a fine balance with liveness.

Decentralization

The consensus platform will favor decentralization moderately over scalability and

liveness. It's a fine balance between decentralization and scalability that Doge Protocol blockchain aims to achieve.

Finality

Once a block is created, it should be considered finalized. There shouldn't be scope for rollback of the block or transactions in the block.

Scalability

The consensus model should be adaptable to improving hardware and network speed over a period of time, as a way to increase transactions per second (TPS) (which also improves the usability).

Terminology

Block

A list of transactions combined with other metadata forms a block.

Clients

A client refers to an account that wants to send a transaction for inclusion in the blockchain, such as sending funds to another account.

Proposer

A proposer will propose a block for inclusion in the blockchain.

Validator

A validator notarizes the blocks proposed by the proposer, for inclusion in the blockchain. Validators also become block proposers based on the validator selection algorithm.

Staking

A validator can stake coins to run a validator node which becomes part of the underlying blockchain network.

Rewards

Rewards are paid to validators, proposers, and inspectors for running and securing the blockchain network.

Slashing

Slashing is used to penalize bad actors in the blockchain network (including validators, proposers, and inspectors).

Block Proposer and Validator Selection

Randomness

A robust randomness algorithm is essential to ensure there is less opportunity for bad validators to game the system to make themselves (either directly or using sybil validators) eligible as block proposers. Traditional approaches to randomness selection in blockchains rely on an external source of randomness or use the block hash of the parent block for source of randomness.

Doge Protocol blockchain uses a simple yet effective approach as source of randomness. Let's say N is the current block, X is a constant and V is the number of registered validators.

Randomness Source = Hash(append(BlockHash(N - X), V)

Example:

Current Block Number N = 750000
X constant value = 512000
V = 92
Randomness Source = Hash(append(BlockHash(750000 - 512000), 92)

The advantage of this approach is that it becomes harder for validators to game the system because there are two discrete variables, one is the block hash of a block that is 512000 blocks away while the second variable is the number of validators that have registered validators as of current block.

The number of registered validators is not something that can be easily gameable, since a bad actor will need enough coins to register as a new validator, while the first variable (N - X) reduces chances of affecting a change in the randomness in the next few blocks.

A disadvantage with this approach is that it's easier to predict the block proposer much early in advance, unless new validators register in the meantime. But the advantages of simplicity that doesn't require an external source of randomness makes this an effective approach.

Validator Selection

Every block, 128 validators are selected at random from the list of validators, to vote on the block. The magic number 128 is based on this math. If there are less than 128 validators, all validators are selected. Validators that have been paused or out of balance of staked coins are not selected from the list. If there are more than 128 validators, the above randomness source is used to select a subset of 128 validators, with weightage given to validators based on the number of coins

they have staked. The combined staked coins of all validators should exceed a given threshold, as part of security measures of the blockchain. This value is bound to change in the future and hence is not explicitly called out in this whitepaper.

Block Proposer Selection

Using the above approach to randomness, a block proposer is selected from the list of the selected validators for the block. Once a subset of validators are selected from the list of validators for a given block, there is no further weightage given to the number of staked coins for selecting the block proposer, for fairness.

Consensus Mechanism

Doge Protocol blockchain uses a variant of pBFT consensus. Every block goes through 3 phases of voting.

At a high level, the sequence of steps involved for consensus voting are:

Propose->Ack Proposal->Pre-commit->Commit

Step 1 (proposal) : Block Proposer sends a block proposal consisting of the parent block hash and a list of transaction hashes to include in the block (the list can also contain zero transaction hashes).

Step 2 (vote phase 1 : ack proposal): Once validators get the block proposal and either vote OK or NIL for this proposal block. If validators don't get this proposal after a threshold of wait time (T1), they will vote the block as NIL.

During this stage, it is possible that some validators received the block proposal on time, while some didn't. Details on how this case is handled will be shown in subsequent stages.

Step 3 (vote phase 2 : pre-commit): Validators will get OK votes cast by other validators on the block proposal or NIL votes (and in some cases might not receive all validator votes). Once they get 70% of a specific vote type (OK or NIL), corresponding to number of staked coins in the validator set, the validators will send a pre-commit vote to other validators.

If 70% of votes of a specific vote type are not available within a time threshold T2, validators will re-vote for Round 2, selecting a new block proposer. Some of the lower level heuristics for advancing to round 2 are not detailed in this whitepaper and are left to implementation.

To reduce impact on liveness due to poor networks (FLP theorm), the number of rounds is limited to 2 and for this Round 2, validators only cast NIL votes.

Stage 4 (vote phase 3 : commit): Validators will get pre-commit votes cast by other validators on the block. Once they get 70% of pre-commit votes, corresponding to the number of staked coins in the validator set, the validators will send a commit vote to other validators. Once a validator casts a commit vote, as per the consensus mechanism, the validator can never cast a Round 2 vote or rollback to a different vote type (excluding software bugs and bad actors).

Stage 5 (Finalize Block) : Once validators get 70% of commit votes, corresponding to the number of staked coins in the validator set, the validators will mark the block as finalized and move on to the next block.

If the block was votes as an OK block, the block proposer gets block rewards accumulated. If the block was votes as NIL block, the block proposer gets a slashing (penalty) on the staked coins.

Liveness

As explained in the Fischer-Lynch-Paterson (FLP) theorem, in an asynchronous network, no consensus algorithm can guarantee termination in every execution, if at least one node may fail. Since Doge Protocol blockchain achieves finality immediately after a block is created, there is an effect on Liveness property. To reduce impact on liveness due to poor asynchronous networks, the wait durations for T1 and T2 detailed above are kept at a high threshold. As of August 2024, T1 is 60 seconds and T2 is 300 seconds.

Summary

The Doge Protocol consensus whitepaper outlines typical problems that can happen on a Proof-of-Stake consensus blockchain and how the system works around it. A lot of the items are kept abstract and high level, leaving specifics to implementation details, on purpose. Some essential properties of the blockchain like “Data Availability” will be detailed in follow-up whitepapers.

Appendix

1. Practical Byzantine Fault Tolerance
   https://pmg.csail.mit.edu/papers/osdi99.pdf

2. RandDAO https://github.com/randao/randao

3. Validator Count: https://medium.com/@chihchengliang/minimum-committee-size-explained-67047111fa20

4. Impossibility of Distributed Consensus with One Faulty Process (FLP) https://groups.csail.mit.edu/tds/papers/Lynch/jacm85.pdf