CACHE Project

proof of WORK
proof of STAKE

Separated difficulty recalculation functions for two types of work evidence

Block time proof of WORK - 900 sec

Block time proof of STAKE - 600 sec

Specifications

Scrypt Jane mining algorithm - Nf 19
Two consensuses mechanism - Proof of Work & Proof of Stake
Preferred consensus device - CPU
Total supply - 2000000000
Difficulty adjustment algorithm for two consensuses divided
Algorithm - VALM-Cache (logical analysis, mathematically variable)
Block time Proof of Work - 15 minutes
Block time Proof of Stake - 10 minutes
Difficulty adjustment - each block according to the results of three intervals
Spam-Hash control function added
Subsidy algorithm modified - proportional to the difficulty (min 40 CACHE, max 80 CACHE)
Block maturity - 520 confirms
Transaction maturity - 6 confirmations
Coin age to Stake - 7 days
Coins per annum - 10%
RPCPort - 2224
Network Port - 2225

VALM-Cache(logical analysis, mathematically variable) difficulty recalculation algorithm

The difficulty-readjustment algorithm VALM-CACHE is quite complex and represents the first solution of its type. VALM includes several analytical functions and several more that perform value recalculation. One could compare VALM to the Russian writer Krylov's famous fable The Swan, the Crayfish and the Pike, in which the three animals try to pull a loaded cart, but each of them pulls in a different direction. Each function in VALM uses its own algorithm to analyze and readjust value, essentially "pulling" the final value in its own direction. Resulting value changes are completely non-linear and can even seem wrong to users who are familiar with the subject. When using VALM, compared to different algorithms, in cases where difficulty would normally increase we will see it remain virtually unchanged, decrease, increase to the same extent, or increase several times faster.

As an example, let us examine a network of a certain processing power. VALM uses three intervals to calculate an average value, then preselects the necessary value based on the analysis of non-pooled data, corrects the TargetSpacing, and finally all these values – after many more transformations – are inserted into the final function, with actually readjusts difficulty. Let's explore two sample cases.

1. A network of a stable processing power, with the average value of the three intervals close to the TargetSpacing value. VALM recalculates difficulty gradually, without significant increases or decreases, even with fast blocks or long intervals. VALM partially ignores single outlying value that differ significantly from the target spacing value and readjusts difficulty taking into account the predetermined number of blocks per 24 hours.

2. A network with a hashrate over double that of the stable network from example 1) connects to it to carry out an attack. Based on the results of the first and even second short interval, VALM slightly increases the difficulty, checking for the possibility to generate blocks with a high luck value. After the third fast block, the algorithm's aggressiveness level changes, and if short intervals continue, the "Swan, Crayfish and Pike" start pulling in the same direction. As a result, the processing power of the attacking network does not matter, since by the fifth or sixth block the network will have a difficulty that corresponds to its hashrate, and by the 7th-10th block - rounds will be sufficiently long.

Response to the command «getdifficulty»

OBJ

```
{
"proof-of-work" : 0.00024414,
"search-interval-powblock" : 396,
"search-twointerval-powblock" : 2042,
"search-full-result-powblock" : 2042,
"pow-target-spacing-variable" : 1067,
"UpperLower-pow" : 128,
"XUpper-pow" : 1,
"XLower-pow" : 108,
"proof-of-stake" : 1.19072277,
"search-interval-posblock" : 1174,
"search-twointerval-posblock" : 1153,
"search-full-result-posblock" : 1153,
"pos-target-spacing-variable" : 409,
"UpperLower-pos" : 101,
"XUpper-pos" : 1,
"XLower-pos" : 72,
"search-interval-without pow block" : 5708,
"search-interval-without pos block" : 925,
"UnixCachChainTime" : 1541063732,
"study" : 0.00000000,
"studys" : 0.00000000
}
```

OBJ

```
{
"proof-of-work" : 0.00024414,
"search-interval-powblock" : 5078,
"search-twointerval-powblock" : 10258,
"search-full-result-powblock" : 5078,
"pow-target-spacing-variable" : 601,
"UpperLower-pow" : 51,
"XUpper-pow" : 1,
"XLower-pow" : 78,
"proof-of-stake" : 0.62032050,
"search-interval-posblock" : 211,
"search-twointerval-posblock" : 527,
"search-full-result-posblock" : 527,
"pos-target-spacing-variable" : 728,
"UpperLower-pos" : 300,
"XUpper-pos" : 60,
"XLower-pos" : 1,
"search-interval-without pow block" : 1448,
"search-interval-without pos block" : 916,
"UnixCachChainTime" : 1541143357,
"study" : 0.00000000,
"studys" : 0.00000000
}
```

Spam-Hash Control function

A properly determined interval between blocks by circa 30% guarantees a stable and reliable network. There are projects with both very small TargetSpacing values of 25 seconds and very long ones (25 minutes). The optimal value is between 8 and 15 minutes, and Bitcoin's 10 minutes fall into this interval. However, due to the large number of transactions a TargetSpacing value of 10 minutes is too large for Bitcoin: transactions wait for a long time to be included in a block and become too large. The issue can be solved – without decreasing the TargetSpacing value and reward – by introducing a Proof-of-Stake consensus mechanism, that is, increasing the number of blocks solved every day. A wrongly determined blocks interval is one of the main reason for branching (fork-blockchain), though it is not the only one. In any networks, assets (transactions) move around constantly, and they all need to be confirmed in a new block. Nodes or node groups don't all are using a stable connection. When a new block is announced, some nodes have to prepare and record transactions, sending corresponding information to the network: these data are transmitted with a delay and received with a delay by other nodes that have a slower channel or to weak platform.

In some cases it takes several minute for the information about transactions and new blocks to be delivered to all network participants. If one of the nodes generates a new fast block in the meantime, information about it will also spread with a delay, and the first nodes to receive informations will be those that ere not puzzleds by work during that time. In such situations a fork-blockchain can emerge.

Conclusion: a network needs some idle time for information about new transactions and blocks to reach all nodes. Experiments show that the inactivity period should equal circa 30% of the TargetSpacing value. Its length is controlled by the Spam-Hash Control function.

When attacking network, this function carries a certain psychological element of disappointment. With the TargetSpacing value of 10 minutes, regardless of the attacker's hashrate, the next block will be solved no sooner than 3 minutes after the previous one.

Proof of Stake consensus mode

The coin generation process using Proof-of-Stake starts on the 8th day. For reward calculation the system uses the total of transactions with timestamp values older than the past seven days. On the 8th day Proof-of-Stake generation begins at the interest rate of 3% per annum, the interest rate increases every day to reach 10% on the 31st day.

There are several ways to activate Proof-of-Stake generation:

1. *In the configuration file using the line - posgen=1*
2. *Using the console command - setposgensingle (find a single block and switch off the Proof-of-Stake generation mode)*
3. *With the console command - setposgenfull (search for all blocks using Proof-of-Stake generation mode)*

To limit the number of coins participating in the Proof-of-Stake generation (place them in the reserve), use the command - reservebalance=xxxxxxxxx

Hard Fork - Block 364000

Hard - Fork, about the new in the wallet

The new difficulty recalculation algorithm was tested and showed stable operation. The "VALM-Cache" algorithm quickly reconfigures with an increase or decrease in the hash rate, from micro to maxi. The functions “Spam-Hash control” and “VALM-Cache” complicate the possibility of overtaking the primary blockchain to a critical difference.

The introduction of new functions allowed to reduce the minimum complexity value for the mining algorithm Scrypt Jane - Nf 19. This value is - 0.00000092. Mining is now possible when using a single processor in solo mode.

The checks system with a call from the "ProcessMessage" function significantly reduced the conflicts of the mempool and almost completely eliminated spam of various kinds. Having a sheet storing a bad hash allows the client to ignore duplicate packets at the "inv" level of the command, excluding re-checking, and without using additional resources.

Changes to the protocol.
To achieve consensus, the rule of the longest chain and the calculation of the Trust are not used, he is left for compatibility, and manipulations with it allow you to use it as a trigger. Priority has a block with an earlier timestamp. The protocol assumes that a group of nodes with a large hashrate for some reason delayed information about the new block. Such a chain of blocks will be approved by the protocol if this chain is ahead of the competing chain by one block and the base of this chain is two blocks deeper than the best block. The protocol also allows for the possibility that the blockchain, which he loves, controls and considers the main, is not such. This chain has a later timestamp. For a similar situation, there are two ways out.
1. To reorganize, need to be one block behind the competitor chain.
2. To win, it is necessary to overtake a competing block chain by two blocks, provided that the height of the best block of this chain is higher than the height of the initial block of the fork by two blocks.
All reorganizations of any of the chains are possible only within the specified boundaries of the dispute zone. Outside this zone, the reorganization is not feasible, even when using 100% of the total hashrate.

A function has been implemented to protect transactions in the presence of a fork in the dispute zone.

The above has become possible due to the implementation of continuous scanning of the blockchain for the presence of a fork and the use of a decentralized checkpoint.

Also, over the past year, the correctness of the decision to separate the functions for recalculating the difficulties of ProofOfStake and ProofOfWork was confirmed. The intervals for ProofOfStake and ProofOfWork are different and not interrelated. The data for recalculating the difficulty ProofOfStake is taken from the previous ProofOfStake intervals, and ProofOfWork from ProofOfWork, respectively.

end...

Block diagram of the new blockchain protocol.

Purpose: Creating an algorithm that excludes the possibility of reorganizing a chain of blocks in order to realize double spending, using even 100% of the total hashrate.
Bottom line: Testing required.

Condition "B"
The protocol unambiguously accepts new blocks with an earlier timestamp. The protocol accepts blocks claiming to be the best block without reorganizing the block chain, and blocks with a delay in distribution to a depth of no more than two blocks (inclusive) - with reorganization.

Condition "C"
Blocks with conditions as in "Condition "B"" but with a later timestamp are ignored.

Condition "A"
Blocks with an earlier time stamp and a delay of more than that provided for in "Condition "B"", but not more than the depth of the boundary of the protocol work area, are accepted by the protocol, provided that the competing chain must exceed the main chain by one block or more.

Condition "D"
Blocks with conditions as in "Condition "A"", but without the ability to get ahead of the main chain will be reorganized in its favor.

Condition "E"
Blocks with conditions as in "Condition "A"", but if the difference between the height of the best block and the height of the fork exceeds 30 blocks, the block chain is ignored. Reorganization of such a chain in favor of the main chain will occur when its height lags behind the competitor's block chain height.
However, if such a chain also has more than 30 blocks from the parent block of the fork, the reorganization does not occur with the formation of stable branching. This situation requires more analysis.

The stability zone without the possibility of double spending starts after 31 confirmations.