

This proposes an alteration to the validator rewards as of 0.9. Rationale behind the change is based on consideration of the profit made by validators rather than their income, and accounting for the decrease in minimum validators and Vitalik's suggested maximum useful staked Ether number.

## Assumptions and calculations

- Minimum number of validators is  $V_{\min}$

= 16,384

.

- Maximum useful staked Ether is ~40MM; at 32 Ether per validator and picking the nearest multiple of 16,384 (the minimum number of validators) gives the maximum useful number of validators  $V_{\max}$

= 1,245,184

.

- Minimal cost to run a validator is \$25/month

. This may be on the low side, as it should include hardware, connectivity, power, maintenance hours etc.

, but is as good a number as any for now

- Validator return at  $V_{\max}$

should be ~0

- Overall and individual validator uptimes are 100%

(reality will, of course, be lower)

- value of 1 Ether in USD is \$185
- % profit required for a validator to break even is  $\frac{\text{costs} \times 12}{32 \times 185} = 5\%$

## Proposed alterations to the rewards equation

There are two proposed alterations to the rewards and penalties calculation as per [https://github.com/ethereum/eth2.0-specs/blob/03fb0979485a204890053ae0b5dcbb06c7f1f5c/specs/core/0\\_beacon-chain.md](https://github.com/ethereum/eth2.0-specs/blob/03fb0979485a204890053ae0b5dcbb06c7f1f5c/specs/core/0_beacon-chain.md)

### 1) Increase BASE\_REWARD\_FACTOR from 64 to 128

Doing this sets income at  $V_{\max}$

to be approximately equal to costs.

### 2) Have a floor for total validator balance of 65,536 validators at MAX\_EFFECTIVE\_BALANCE

This would change the definition of `total_balance`

in `[get_base_reward`

]([https://github.com/ethereum/eth2.0-specs/blob/03fb0979485a204890053ae0b5dcbb06c7f1f5c/specs/core/0\\_beacon-chain.md#rewards-and-penalties-1](https://github.com/ethereum/eth2.0-specs/blob/03fb0979485a204890053ae0b5dcbb06c7f1f5c/specs/core/0_beacon-chain.md#rewards-and-penalties-1)) from

`total_balance = get_total_active_balance(state)`

to

`total_balance = max(get_total_active_balance(state), MAX_EFFECTIVE_BALANCE*65536)`

Doing this results in a flat rate of return for validation between 16K and 64K validators, which has two benefits. Firstly, it avoids excessive rates of return when the total number of validators is low by keeping it at just under 18% (uncapped return at 16K validators would be over 40%).

Second, it encourages “second movers” (validators who want to join but wish to wait for the chain to start before staking) by providing an equal rate of return whilst the total number of validators remains relatively low.

## Income and returns

Income is defined as the percentage  $\frac{\text{rewards}}{\text{stake}}$

where rewards

are the maximum rewards over a year for any given validator.

Return is defined as the percentage  $\frac{\text{rewards}-\text{costs}}{\text{stake}}$

where costs

are the costs to run the validator for the year.

In general, returns are used in preference to income for the examination of the rewards. This aligns more closely with how validators will consider validating in Ethereum 2 compared to using their funds for another purpose.

Income and returns are shown in the following chart:

The cap can be seen kicking in at the left of the chart, keeping the return flat between 16K and 64K validators. As per requirements, return at  $V_{\text{max}}$

is near-0 (approximately 0.2%).

## Issuance and inflation

Annual issuance in this model is shown in the following chart:

Inflation is based on a total supply of 110 million Ether. It is shown in the following chart:

With  $V_{\text{max}}$

inflation is below 2% (note this will be reduced with phase 2 and burning of gas).

## Doubling threat

The doubling threat looks at the hypothetical situation where someone says to all of the existing Ethereum 2 validators “stop staking instead do our thing; you will earn double the return you are making now”. Assuming that everyone decides to take up that offer, what percentage reduction in validators will it cause? Or to put it more mathematically: in a situation with  $V_c$

validators generating  $y$

% return, what is the doubling threat  $1 - \frac{V_d}{V_c}$

where  $V_d$

validators generate  $2y$

% return?

Doubling threat is shown in the following chart:

This shows that, for example, with approximate 512K validators nearly half would leave if offered double the return. With higher numbers of validators a smaller percentage of validators would leave; a result of the fact that the majority of income is used for costs so return is very small and doubles with a much smaller increase in income.

Note that the 100% threat at low numbers of validators is a consequence of the cap, although there remains a question as to how many “hard core” validators would remain because their participation is not purely financial.

## Tension with ETHUSD

An obvious outstanding issue is that costs are defined in USD and rewards in Ether. If, for example, ETHUSD went to \$1,000 this would result in validators making a higher return as their costs would be a smaller percentage of their income (and, of course, the reverse applies). Thoughts on if, and if so how, this issue could be addressed are welcome.

