title: Proof-of-stake vs proof-of-work description: A comparison between Ethereum's proof-of-stake and proof-of-work based consensus mechanism lang: en

When Ethereum launched, proof-of-stake still needed a lot of research and development before it could be trusted to secure Ethereum. Proof-of-work was a simpler mechanism that had already been proven by Bitcoin, meaning core developers could implement it right away to get Ethereum launched. It took a further eight years to develop proof-of-stake to the point where it could be implemented.

This page explains the rationale behind Ethereum's switch to proof-of-stake from proof-of-work and the trade-offs involved.

Security {#security}

Ethereum researchers consider proof-of-stake more secure than proof-of-work. However, it has only recently been implemented for the real Ethereum Mainnet and is less time-proven than proof-of-work. The following sections discuss the pros and cons of proof-of-stake's security model compared to proof-of-work.

Cost to attack {#cost-to-attack}

In proof-of-stake, validators are required to escrow ("stake") at least 32 ETH in a smart contract. Ethereum can destroy staked ether to punish validators that misbehave. To come to consensus, at least 66% of the total staked ether has to vote in favour of a particular set of blocks. Blocks voted for by >=66% of the stake become "finalized", meaning they can't be removed or reorganized.

Attacking the network can mean preventing the chain from finalizing or ensuring a certain organization of blocks in the canonical chain that somehow benefits an attacker. This requires the attacker to divert the path of honest consensus either by accumulating a large amount of ether and voting with it directly or tricking honest validators into voting in a particular way. Sophisticated, low-probability attacks that trick honest validators aside, the cost to attack Ethereum is the cost of the stake that an attacker has to accumulate to influence consensus in their favour.

The lowest cost of attack is >33% of the total stake. An attacker holding >33% of the total stake can cause a finality delay simply by going offline. This is a relatively minor problem for the network as there is a mechanism known as the "inactivity leak" that leaks stake away from offline validators until the online majority represents 66% of the stake and can finalize the chain again. It is also theoretically possible for an attacker to cause double finality with a little over 33% of the total stake by creating two blocks instead of one when they are asked to be a block producer and then double-vote with all of their validators. Each fork only requires 50% of the remaining honest validators to see each block first, so if they manage to time their messages just right, they may be able to finalize both forks. This has a low likelihood of success, but if an attacker was able to cause double-finality, the Ethereum community would have to decide to follow one fork, in which case the attacker's validators would necessarily be slashed on the other.

With >33% of the total stake, an attacker has a chance to have a minor (finality delay) or more severe (double finality) effect on the Ethereum network. With more than 14,000,000 ETH staked on the network and a representative price of $1000/ETH, the minimum cost to mount these attacks is `1000 x 14,000,000 x 0.33 = $4,620,000,000`. The attacker would lose this money through slashing and get ejected from the network. To attack again, they would have to accumulate >33% of the stake (again) and burn it (again). Each attempt to attack the network would cost >$4.6 billion (at $1000/ETH and 14M ETH staked). The attacker is also ejected from the network when they are slashed, and they have to join an activation queue to rejoin. This means the rate of a repeat attack is limited not only to the rate the attacker can accumulate >33% of the total stake but also the time it takes to onboard all their validators onto the network. Each time the attacker attacks, they get much poorer, and the rest of the community gets richer, thanks to the resulting supply shock.

Other attacks, such as 51% attacks or finality reversion with 66% of the total stake, require substantially more ETH and are much more costly to the attacker.

Compare this to proof-of-work. The cost of launching an attack on proof-of-work Ethereum was the cost of consistently owning >50% of the total network hash rate. This amounted to the hardware and running costs of sufficient computing power to outcompete other miners to compute proof-of-work solutions consistently. Ethereum was mostly mined using

GPUs rather than ASICs, which kept the cost down (although had Ethereum stayed on proof-of-work, ASIC mining may have become more popular). An adversary would have to purchase a lot of hardware and pay for the electricity to run it to attack a proof-of-work Ethereum network, but the total cost would be less than the cost required to accumulate enough ETH to launch an attack. A 51% attack is ~20x less expensive on proof-of-work than proof-of-stake. If the attack was detected and the chain hard-forked to remove their changes, the attacker could repeatedly use the same hardware to attack the new fork.

Complexity {#complexity}

Proof-of-stake is much more complex than proof-of-work. This could be a point in favour of proof-of-work as it is harder to introduce bugs or unintended effects into simpler protocols accidentally. However, the complexity has been tamed by years of research and development, simulations, and testnet implementations. The proof-of-stake protocol has been independently implemented by five separate teams (on each of the execution and consensus layers) in five programming languages, providing resilience against client bugs.

To safely develop and test the proof-of-stake consensus logic, the Beacon Chain was launched two years before proof-of-stake was implemented on Ethereum Mainnet. The Beacon Chain acted as a sandbox for proof-of-stake testing, as it was a live blockchain implementing the proof-of-stake consensus logic but without touching real Ethereum transactions - effectively just coming to consensus on itself. Once this had been stable and bug-free for a sufficient time, the Beacon Chain was "merged" with Ethereum Mainnet. This all contributed to taming the complexity of proof-of-stake to the point that the risk of unintended consequences or client bugs was very low.

Attack surface {#attack-surface}

Proof-of-stake is more complex than proof-of-work, which means there are more potential attack vectors to handle. Instead of one peer-to-peer network connecting clients, there are two, each implementing a separate protocol. Having one specific validator pre-selected to propose a block in each slot creates the potential for denial-of-service where large amounts of network traffic knock that specific validator offline.

There are also ways that attackers can carefully time the release of their blocks or attestations so that they are received by a certain proportion of the honest network, influencing them to vote in certain ways. Finally, an attacker can simply accumulate sufficient ETH to stake and dominate the consensus mechanism. Each of these attack vectors has associated defenses, but they do not exist to be defended under proof-of-work.

Decentralization {#decentralization}

Proof-of-stake is more decentralized than proof-of-work because mining hardware arms races tend to price out individuals and small organizations. While anyone can technically start mining with modest hardware, their likelihood of receiving any reward is vanishingly small compared to institutional mining operations. With proof-of-stake, the cost of staking and the percentage return on that stake are the same for everyone. It currently costs 32 ETH to run a validator.

On the other hand, the invention of liquid staking derivatives has led to centralization concerns because a few large providers manage large amounts of staked ETH. This is problematic and needs to be corrected as soon as possible, but it is also more nuanced than it seems. Centralized staking providers do not necessarily have centralized control of validators - often it is just a way to create a central pool of ETH that many independent node operators can stake without every participant requiring 32 ETH of their own.

The best option for Ethereum is for validators to be run locally on home computers, maximizing decentralization. This is why Ethereum resists changes that increase the hardware requirements for running a node/validator.

Sustainability {#sustainability}

Proof-of-stake is a carbon-cheap way to secure the blockchain. Under proof-of-work miners compete for the right to mine a block. Miners are more successful when they can perform calculations faster, incentivizing investment in hardware and energy consumption. This was observed for Ethereum before it switched to proof-of-stake. Shortly before the transition to proof-of-stake, Ethereum was consuming approximately 78 TWh/yr - as much as a small country. However, switching to proof-of-stake reduced this energy expenditure by ~99.98%. Proof-of-stake made Ethereum an energy-efficient, low carbon platform.

More on Ethereum's energy consumption

Issuance {#issuance}

Proof-of-stake Ethereum can pay for its security by issuing far fewer coins than proof-of-work Ethereum because validators do not have to pay high electricity costs. As a result, ETH can reduce its inflation or even become deflationary when large amounts of ETH are burned. Lower inflation levels mean Ethereum's security is cheaper than it was under proof-of-work.

More of a visual learner? {#visual-learner}

Watch Justin Drake explain the benefits of proof-of-stake over proof-of-work:

Further reading {#further-reading}

- Vitalik's proof-of-stake design philosophy
- Vitalik's proof-of-stake FAQs
- "Simply Explained" video on pos vs pow