

Special thanks to [cyberhorsey](#), [d1onys1us](#), [Brecht Devos](#), [Lisa](#), [Dani](#), [Daniel Wang](#), and other Taiko teammates for their feedback.

TL;DR:

Provers on Eldfell L3 have seen a significant portion of their TTKOe slashed. That's because of three reasons:

1. A substantial increase in transaction volumes and latency on Grímsvötn L2;
2. MIN\_CAPACITY

set to too large of a value (128

);

1. Unfavorable slashing formula.

We've addressed these issues by clearing the mempools on both Eldfell L3 and Grímsvötn L2, reducing MIN\_CAPACITY to 32

, and adjusting the slashing formula.

Eldfell, Taiko's L3 testnet, launched two weeks ago with two explicit goals: Experimenting with Inception Layers (L3s) and testing out a new staking-based proving design.

Provers on Eldfell L3, and those that have been paying attention in general, have perhaps noticed a significant increase in [slashed TTKOe](#). There are interesting reasons behind the slashing events that we thought would be worth sharing and discussing with the community.

So let's dive in!

## Comparing proving schemes on Grímsvötn and Eldfell

Before discussing slashing, let's first take a brief look at how the staking-based proving design works on Eldfell L3 and how it differs from Grímsvötn L2.

Note

: If you know how proving works on Grímsvötn L2 and Eldfell L3, you can skip this section and go to [Why provers were getting slashed](#)

.

You can also learn more about the similarities and differences between the two proving schemes [here](#).

### Most efficient prover wins

In Grímsvötn L2, the fastest prover always got to prove blocks. This meant that multiple provers could work simultaneously on a single block but only one would win, essentially making others waste their resources.

While this had virtually no real-life economic impact on provers in a testnet environment because of low circuit coverage, provers with weaker hardware on mainnet would presumably want to avoid wasting their work and eventually give up. This would lead to centralization as only the most efficient provers would be incentivized to stay in the network.

[

If you know what I mean.

1920×1057 137 KB

](<https://global.discourse-cdn.com/standard17/uploads/taiko/original/1X/1a8440c08d0c6ad62a53e004d10257f63b6ceefb.jpeg>)

### Staking-based mechanism

In Eldfell L3, we decided to test a new staking-based proving design to see whether it's possible to reduce resource wasting, decrease the risk of centralization, and incentivize low-cost proving.

Here's how it works:

1. The protocol forms a prover pool of the top 32 provers for each slot.
2. It then selects a prover from the prover pool and assigns it a block to prove within a specific time window (proofTimeWindow).
1. If the selected prover proves the block within the specified time window, it gets a reward. If the prover fails to submit a proof on time, it gets slashed and the block becomes open to everyone. The most efficient prover gets to prove the open block.

Note:

If all the provers from the top 32 prover pool are at MAX\_CAPACITY

when a block is proposed, the block becomes open to everyone to prove.

The top 32 prover pool

is determined by the number of tokens a prover stakes per capacity (stakedAmount

). The higher the stake, the more likely the prover will be selected.

As for block assignment

, this is determined by:

1. The expected reward earned per block (rewardPerGas
- ). Higher rewards expected (or demanded) make it less likely for the prover to be selected.

1. The current fee per gas (feePerGas

) set by the protocol.

1. The capacity specified by the prover. If a prover is selected, the capacity reduces by one, and when the capacity hits zero, the prover will no longer be selected. When a block assigned to this prover is proven (by them or any other prover), the capacity increases by one, up to the max capacity specified during staking. MAX\_CAPACITY

is set by the prover themselves (no lower than 128

) and MIN\_CAPACITY

, originally, was set to 128

.

Now let's see why a significant number of provers on Eldfell L3 were getting slashed.

## Why provers were getting slashed

No, it's not the hash-slinging slasher that's been slashing Eldfell L3 provers (though the dude does look suspicious). There are two main reasons why multiple provers on Eldfell L3 were getting their TTKOe slashed:

1. The latency of a based L3 like Eldfell can be high. An L3 proposer needs to pick up transactions from the mempool, create a block, and submit a proposeBlock

transaction to Grímsvötn L2. Then that proposeBlock

transaction has to be included in an L2 block by an L2 proposer. When it's picked up and included in an L2 block, then the L2 proposer also has to submit a proposeBlock

transaction with

the L3 proposeBlock

transaction in it to the Taiko L1 contract.

When the L2 proposeBlock

transaction with the L3 proposeBlock

transaction in it hits the base chain (Ethereum L1), then the base chain emits a blockProposed

event. After that, the L2 block with L3 proposeBlock

transaction in it is added to Grímsvötn L2, and when it's added to Grímsvötn L2, the L3 block is added to Eldfell L3. Only when the L3 block is added to Eldfell L3 can it be assigned to an L3 prover.

[  
Lifecycle of an L3 block on Eldfell.  
1920×1433 133 KB  
(<https://global.discourse-cdn.com/standard17/uploads/taiko/original/1X/78015b1a8044889a644475ee25f38e3b4c885b98.jpeg>)  
Once an L3 prover is assigned the L3 block, they need to submit a proveBlock

transaction within a specific time window (proofTimeWindow  
) to Grímsvötn L2. Then the process is the same as described above: An L2 proposer has to pick up the L3 prover's proveBlock  
transaction, construct an L2 block, and submit a proposeBlock  
transaction to the Taiko contract on Ethereum L1.

[  
Lifecycle of an L3 prover.  
1549×2140 138 KB  
(<https://global.discourse-cdn.com/standard17/uploads/taiko/original/1X/9b9a000253f061303012ac6f48410280746e3d82.jpeg>)  
If the L2 proposeBlock

transaction hits the base chain (Ethereum L1) within the specific time window (proofTimeWindow  
) that the L3 prover was assigned, then the prover is not slashed. However, the L3 prover still needs to wait for the block to be verified later: If the proof turns out to be valid after the block is verified, the prover gets rewarded; if proof turns out to be invalid after the block is verified, the prover gets slashed for generating a proof with incorrect values (even if they submitted proveBlock  
in time).

[  
Lifecycle of an L3 prover part 2.  
1774×1583 198 KB  
(<https://global.discourse-cdn.com/standard17/uploads/taiko/original/1X/bfec7704993c84e738e96e216e7b08214b62aab4.png>)

In general, everything that happens on Eldfell L3 has to go through Grímsvötn L2 to the base chain. Hence an L3 like Eldfell is subject to some two-mempool latency.

1. As we said earlier, MIN\_CAPACITY  
was originally set to 128

. This meant that to be successful and not get slashed, provers had to be able to prove 128 blocks in parallel. The majority of provers on Eldfell L3 are incapable of proving that many blocks simultaneously but still had to choose their minimal capacity as 128 and later got slashed. We made this design choice consciously for testing protocol and prover economics.

Additionally, some provers staked a lot of TTKOe and specified their expected reward and capacity in a way that included them in the proving pool. However, what they lacked was powerful enough hardware to actually prove the blocks they got assigned, and thus they got slashed.

Note

: TTKOe is not to be confused with TTKO, which is a separate test token on Grímsvötn L2.

## Adjustments we've made

To make proving on Eldfell L3 more accessible and fair, we've made the following adjustments:

1. Both mempools on Grímsvötn L2 and Eldfell L3 are now at stable levels. This doesn't mean that we won't see another substantial increase in transaction volume in the future or that no gas wars start. However, we've put extra resources towards ensuring the mempools stay manageable.

2. MIN\_CAPACITY

has been lowered from 128

to 32

. This means that provers no longer need to be capable of proving 128 blocks in parallel and can choose to take up fewer blocks to prove at a time.

However, we want to stress that even if MIN\_CAPACITY

has been lowered to 32

, there are still going to be provers who'll get slashed. Again, we want to see slashing events in part because we need to test the slashing logic.

1. The formula for slashing has been updated. Previously, the protocol slashed 0.25% of the total staked amount. Now, if, say, a prover has set MAX\_CAPACITY

at 100, the protocol slashes  $0.25\% * 1/100$  of the total staked amount. In other words, slashed provers are going to lose less of their staked TTKOe tokens.

## Your thoughts?

We'd love to hear the community's thoughts on the slashing events and the adjustments we've made. We also want to thank the community for their active involvement in Eldfell L3 and other testnets. Your feedback on Discord and elsewhere is invaluable.

On a separate note, we want to remind you that we launched the first [Taiko Community Grant Program](#) a couple of weeks ago. If you've got ideas on how to improve the staking-based proving design, proof markets, or any other Taiko-related topic, check out the program and consider applying.