Context

For the last months Lido has purchased slashing insurance from Unslashed Finance, which has now expired. The premium for this insurance has been <u>estimated</u> to be \sim 25% of the DAO's yearly revenues. This thread's purpose is to discuss the path forward about Lido's insurance plans.

Available Options

· Purchase insurance:

Historically, we have been unable to purchase cover for enough funds at a reasonable premium (e.g. Unslashed provided cover for up to 5% of the funds staked with 1 validator).

· Do not purchase insurance:

In theory, by not purchasing insurance we would be exposing our users to more slashing risk than before. Cover is also important for institutional users. In practice, Lido's governance-gated validator registry has allowed onboarding high quality aligned validators that haven't been slashed to date. All beacon chain slashing events to date are quite <u>minimal</u>: <100 ETH slashed from 145 validators out of >185k validators.

• Lido would provide insurance to its users:

Another approach is to have an in-protocol insurance fund, which covers certain scope of risks (here, slashing risks). For example, that mechanism could either be an <u>AAVE-style safety module</u>, or it could be more ad-hoc via governance proposals.

The path forward

I <u>polled Twitter</u> recently on this subject, and opinions were mixed, with more erring towards not purchasing more insurance from Unslashed. We probably cannot expect that anybody would offer us insurance on the entire principal in Lido (even if they did, it's unlikely they'd be able to pay it out).

Not having any insurance at all could be reckless and hurt customer trust.

As a result, it seems like a reasonable path forward would be for Lido to "own the stack", and informally insure users by reimbursing them if a slashing event happens according to the chosen mechanism(s).

In all cases, we should be explicit about the edge cases involved in order to not create false expectations around what kinds of events users are insured against.