Lido on Polygon V2 upgrade proposal

Reasoning:

Following the successful launch of Lido on Polygon V1, we've been working hard incorporating stakeholder feedback to make marked improvements in reducing both staker as well as node operator friction. As a result, we've come up with a V2 of Lido on Polygon that makes fewer compromises, eases operator entry and exit, and provides a smoother flow for users and network participants. This version not only brings nice-to-have features for node operators participating in the Lido on Polygon protocol, but also reduces the gas needed for depositing and withdrawing tokens.

As we are committed to the values put forward by the community and the DAO, this upgrade will enable us to onboard a new set of node operators and further decentralise the Polygon network while providing options for more top tier node operators to join the Lido family.

Security:

As Lido takes pride in its dedication to security and this upgrade is no exception. A full smart contract code audit was done by Oxorio and is available for you to read here.

TL;DR change log:

- Rebalancing stake among validators on deposits and withdrawals
- Simplified logic for node operator registry
- No need to transfer validator ownership to Lido's smart contracts
- Finer role based access control
- Gas optimizations for delegations/withdrawals and node operator controls
- Code refactoring for readability and maintenance
- Small fixes and code improvements

Full Technical Changelog:

NodeOperatorRegistry

- New logic for NodeOperatorRegistry contract.
- The validatorFactory Contract is no longer used by the NodeOperatorRegistry logic.
- Removed validator proxy logic, the validator can join the protocol without transferring the validator slot to the protocol.
- Added logic to rebalance delegations.
- Added logic to delegate tokens between validators.
- Improved the logic to request a withdrawal from validators.
- Added ADD_NODE_OPERATOR_ROLE and REMOVE_NODE_OPERATOR_ROLE roles to manage operator life cycle
- Split the pause role into 2 sub roles PAUSE_ROLE and UNPAUSE_ROLE
- Added possibility to remove a validator by any wallet if its status is in UNSTAKE or EJECTED
- Added getProtocolStats function which allows to get info about if the system is balanced or not.

stMATIC

- The requestWithdraw algorithm was changed to a new one that keeps the validators balanced.
- Each requestWithdraw will generate 1 NFT token even if the request will be distributed between multiple validators.
- Added token2WithdrawRequests to map 1 NFT token with multiple withdrawal requests.

- Added PAUSE_ROLE and UNPAUSE_ROLE
- The delegate function algorithm was improved to automatically do rebalance and delegate to the validators who have less delegation.
- The claimTokens function is compatible with NFTs generated by the requestWithdraw in V1 and V2.
- Added protocolFee setter.
- Added rebalanceDelegatedTokens; it rebalances the system.

LidoNFT

- Delete the NFT from token2Index after request withdrawal, in V1 it was set to ZERO.
- Delete the NFT from tokenId2ApprovedIndex after approving the NFT, in V1 it was set to ZERO.

[1.0.0

](https://github.com/shard-Labs/PoLido/)

Action plan

Snapshot voting

A snapshot vote will be started next week to measure the DAO sentiment towards the proposed changes.

Deployment:

Deployment will go in 2 phases.

Phase one: Testnet release

Phase two. Mainnet release (approx 2 weeks after the testnet launch)

Marketing

Content will be released to inform the community about the upgrade and downtime of staking services and to address concerns with timely release.

Node Operator Operations

- Validator slots will be retrieved by existing Lido on Polygon Node Operators from the contract
- Existing Node operators will be registered in the V2 contract

Validator onboarding

Additions to the Lido on Polygon validator will be onboarded in Q4 or early Q1/2023. Depending on the TVL at a time, the Lido Node Operator Subgovernance Group (LNOSG) will decide on the number of node operators to be onboarded in the next round.