Deposit Security Committee manual

This instruction has been prepared for the participants of the Deposit Security Committee and describes the general points, the preparation steps to act as a guardian, and the details of the protection mechanism. The Deposit Security Committee is necessary to prevent the substitution of withdrawal credentials with frontrunning by node operators. Each member of the committee must perform several actions to ensure the security of deposits made by Lido. To participate in the validation, you will need to deploy alido-council-daemon and prepare a private key for signing messages about the correctness of data or the need to stop deposits in case of attack.

TL;DR

Before running in the mainnet all steps should be done in the Goerli testnet.

1. Prepare an EOA account for signing data with a private key on hand (not in hardware wallet). It will be a moderately sensitive hot private key. Use different accounts for testnet and mainnet.
2. Send the account address to Lido for submitting it to the smart contract.
3. Deploy and runlido-council-daemon
4. with the private key from the EOA account. It would work in a dry-run mode until your address would be included in the smart contract.

Detailed description

The vulnerability

There is the vulnerability allowing the malicious Node Operator to intercept the user funds on deposits to the Beacon chain in the Lido protocol. The vulnerability could only be exploited by the Node Operator front-running theLido.depositBufferedEther transaction with direct deposit to the DepositContract of no less than 1 ETH with the same validator public key & withdrawal credentials different from the Lido's ones, effectively getting control over 32 ETH from Lido. To mitigate this, Lido contracts should be able to check that Node Operators' keys hadn't been used for malicious pre-deposits.

The Deposit Security Committee

We propose to establish the Deposit Security Committee dedicated to ensuring the safety of deposits on the Beacon chain:

- monitoring the history of deposits and the set of Lido keys available for the deposit, signing and disseminating messages allowing deposits;
- signing the special message allowing anyone to pause deposits once the malicious Node Operator pre-deposits are detected.

To make a deposit, we propose to collect a quorum of 2/3 of the signatures of the committee members. Members of the committee can collude with node operators and steal money by signing bad data that contains malicious pre-deposits. To mitigate this we propose to allow single committee member to stop deposits and also enforce space deposits in time (e.g. no more than 150 deposits with 150 blocks in between them), to provide single honest participant an ability to stop further deposits even if the supermajority colludes. The idea was outlined on research forum post as the optiond .

Committee membership

The first set of guardians is six node operators (Stakefish, Skillz, Chorus one, Blockscape, Staking facilities, P2P) and Lido dev team. In the future, we want to bring as many node operators as possible into the mix, so the expectation will be that while the 7 guardians start the rest of the node operators can also participate via testnet and gradually get pulled into mainnet.

Members responsibilities

Each member must prepare an EOA account to sign the pair(depositRoot, keysOpIndex) with its private key. The addresses of the committee members will be added to the smart contract. Also, member has to runDSC Daemon that monitors the validators' public keys in theDepositContract andNodeOperatorRegistry . The daemon must have access to the committee member's private key to be able to perform ECDSA signing.

The daemon constantly watches all updates inDepositContract andNodeOperatorRegistry :

- If the state is correct, it signes the current to_sign struct and emits it to an off-chain message queue.
- If the state has malicious pre-deposits, it signs the "something's wrong" message at the current block, emits it to a message queue, and attempts to send pauseDeposits() tx.

Preparation steps

Before running in the mainnet, all steps should be completed in the Goerli testnet.

EOA account

It might be any EOA account under the member's control. Send the address of its account to Lido for submitting it to the smart contract. Lido will provide some ETH to make stopping transactions if needed (shouldn't ever be the case). Note, all actions, except sending the stop message, would be done off-chain.

Run lido-council-daemon

To start the application, see the technical documentation in the projectepository . Edit this page Previous Validator exits and penalties Next Tooling Overview