

# Tooling Setup & Configuration

## Keys API (KAPI)

[Dedicated Setup Guide](#)

[GitHub Repo](#)

## Validator Ejector (Ejector)

[Dedicated Setup Guide](#)

[GitHub Repo](#)

## Required Infra for New Tooling

In order for the new tooling to read Lido contracts and validator information, tooling needs access to an Execution Node (full node to be exact) and a Consensus Node.

A dedicated CL+EL setup is recommended.

info Although the Ejector has [security protections](#), using hosted RPC providers (Infura, Alchemy, etc) is discouraged. info It's also advised to have secure Ejector->Nodes and KAPI->Nodes communication, for example via a private network.

## Common Configuration Options

### Operator ID

You can find it on the Operators Dashboard (#123 on the operator card) [Goerli](#) , [Mainnet](#)

### Staking Router Module ID:

ID of the [StakingRouter](#) contract module.

Currently, it has only one module ([NodeOperatorsRegistry](#)), it's id is 1 .

### Oracle Allowlist

The oracle members are retrievable from the HashConsensus (for the Validator Exit Bus Oracle ) contract on-chain, using the Aragon App or directly from the contract using etherscan.

network Aragon App Contract Call Mainnet [Aragon getMembers\(\)](#) Goerli [Aragon getMembers\(\)](#) Holesky n/a [getMembers\(\)](#)

## Example Infra Setup

Lido DevOps team prepared an easy way to get the recommended tooling and its dependencies up and running using [Ansible](#) . This is a great way to get familiar with the new tooling. This is an example implementation, and still requires security and hardening by the NO; it can be found on [GitHub](#) .

It sets up 3 hosts:

- Execution Layer + Consensus Layer nodes ([Geth](#)
- + [Lighthouse](#)
- )
- KAPI & Ejector
- Monitoring

Monitoring consists of:

- [Prometheus](#)
- for metrics
- [Alertmanager](#)
- for alerts
- [Loki](#)
- for logs
- [Grafana](#)

- for dashboards [Edit this page](#) [Previous Flow Examples](#) [Next Oracle Operator Manual](#)