

Run an OP Stack rollup with Celestia underneath

This guide will show you how to run your own OP Stack devnet and testnet that posts data to Celestia's Mocha testnet using [roll-op](#) and [op-plasma-celestia](#) .

The roll-op tool is used to deploy and manage the OP Stack rollup environment, including the rollup, batcher, and other components. While the op-plasma-celestia integration allows the OP Stack to use Celestia as the data availability (DA) layer.

This guide is in two parts:

- First, you'll spin up a mock L1 environment and deploy a devnet that posts data to the Mocha testnet.
- In the second part, you'll deploy a testnet that posts data to the Mocha testnet, but this time on a real L1 environment; the Ethereum Sepolia testnet. This will involve setting up a configuration file with the necessary details like Sepolia chain ID, RPC URL, and your deployment keys.

After successful deployments, you'll be able to observe data blobs being successfully submitted to the Mocha testnet in the logs, as well as some activity on your rollup account on [Celenium](#) .

If you don't have devops experience and would like to use a Rollups as a Service (RaaS) provider, see the RaaS category in the menu.

This guide is also available on [YouTube](#) if you'd like to follow along with a video.

Dependency setup

- [celestia-node](#)

Setting up your light node

Sync and fund a Celestia light node. The light node must be fully synced and funded for you to be able to submit and retrieve PayForBlobs to Mocha Testnet. This allows your rollup to post and retrieve data without any errors.

In order to mount existing data, you must have a node store that is in the default directory:

Mocha

Mainnet Beta

Arabica bash HOME/.celestia-light-mocha-4 HOME/.celestia-light-mocha-4 bash HOME/.celestia-light HOME/.celestia-light bash HOME/.celestia-light-arabica-11 HOME/.celestia-light-arabica-11 By default, the node will run with the account named my_celes_key on Mocha. This is the account that needs to be funded.

TIP

Unless you changed your configuration, you won't have to change anything. 😊

Deploying a devnet to Mocha

See [the Alt-DA x Celestia README](#) for instructions on [how to deploy a Devnet](#) .

TIP for macOS users

If you are on macOS, you will need to run `avenv` before starting `roll-op` .

```
sh cd HOME /roll-op python3
```

```
-m
```

```
venv
```

```
./venv source
```

```
./venv/bin/activate cd HOME /roll-op python3
```

```
-m
```

```
venv
```

```
./venv source
```

./venv/bin/activate Congrats! Your devnet is running on a mock EVM chain and Celestia Mocha.

Deploying a testnet to an L1 (or L2) and Mocha

See [the Alt-DA x Celestia README](#) for instructions on [how to deploy a Testnet](#).

TIP

If you are using a public RPC for your EVM chain, you should enable `deploy_slowly = true` in your `config.toml`. If you still have issues, we recommend running the integration with a high-availability, paid endpoint. When you are deploying to a live EVM network, pay attention and modify the configuration to post to non-Sepolia EVM chains.

Here is an example:

toml

Chain ID of your rollup

`l2_chain_id = 1117733`

Sepolia Ethereum

`l1_chain_id = 11155111 l1_rpc_url = "https://ethereum-sepolia-rpc.publicnode.com"`

Avoid issues with public RPC

`deploy_slowly = true`

Keys

`contract_deployer_account = "0xaddress" contract_deployer_key = "privatekey" batcher_account = "0xaddress"`
`batcher_key = "privatekey" proposer_account = "0xaddress" proposer_key = "privatekey" admin_account = "0xaddress"`
`admin_key = "privatekey" p2p_sequencer_account = "0xaddress" p2p_sequencer_key = "privatekey"`

Chain ID of your rollup

`l2_chain_id = 1117733`

Sepolia Ethereum

`l1_chain_id = 11155111 l1_rpc_url = "https://ethereum-sepolia-rpc.publicnode.com"`

Avoid issues with public RPC

`deploy_slowly = true`

Keys

`contract_deployer_account = "0xaddress" contract_deployer_key = "privatekey" batcher_account = "0xaddress"`
`batcher_key = "privatekey" proposer_account = "0xaddress" proposer_key = "privatekey" admin_account = "0xaddress"`
`admin_key = "privatekey" p2p_sequencer_account = "0xaddress" p2p_sequencer_key = "privatekey"` Your `0xaddress` key must also be funded with testnet ETH. We recommend at least 10 SepoliaETH to get your chain started, but you will need more to keep it running longer.

Congratulations

Congrats! You now have an OP Stack rollup running with Celestia underneath.

You can [learn more about Alt-DA in Optimism docs](#). [[Edit this page on GitHub](#)] Last updated: [Previous page Intro to OP Stack integration](#) [Next page Bubs testnet](#) []