

TL;DR

We have released aave-tenderly-cli

, a cli (command-line-interface) utility that you can use to create forks and simulate proposal execution tenderly and on the decentralized Aave user interface.

<https://github.com/bgd-labs/aave-tenderly-cli>

Current State

Over the recent months, we released various tools, [libraries](#), and [templates](#) to reduce the entry barrier for making governance proposals and help people to deliver quality to the community. Still, an area quite untapped until now was UI verification.

At the moment teams who create governance proposals usually have a manual process to test if their proposals work and do what they are supposed to do. We recently added a feature to [Aave Seatbelt](#) which allows running against a fork, so teams can preview their changes before execution.

Still, this process usually involves creating a fork, executing a sequence of transactions to manipulate block-state, and eventually executing the proposal on a fork. This is a quite cumbersome and repetitive task so we set out to automate things - which is exactly what aave-tenderly-cli

does

.

Aave Tenderly cli

Aave tenderly cli, like the name suggests is an opinionated cli tool built on top of the Tenderly API and RPC features. We chose to use Tenderly because of their very good state_diff

feature, having easily shareable simulation reports and forks being publicly available for anyone to use. The library allows you to create simple forks of various networks. On top of that, you can execute proposals in various states of the creation process. Right now the cli assumes that all payloads have a simple execute()

signature.

Getting started

First, you need to install the package: you can add it to a specific project `npm i @bgd-labs/aave-tenderly-cli`

**** or install it globally `npm i -g @bgd-labs/aave-tenderly-cli`

****.

The cli will automatically source .env

from your current folder for fetching tenderly setting. Alternatively, you can set the env manually via export

or you add them to .bashrc

to have them available permanently. Once the tool is installed we can start forking

Simple fork in action!

While the tool will automatically start a dialog to create the fork you desire, there's also a cli option built in for power users.

`aave-tenderly-cli --help` Usage: `npx aave-tenderly-cli`

Options: `--help` Show help [boolean] `--version` Show version number [boolean] `--forkId` Reuse existing fork id [string] `--networkId` Network to be forked [required] [choices: 1, 10, 137, 250, 42161, 43114, 1666600000] `--blockNumber` Blocknumber to fork off [string] [default: "latest"] `--forkNetworkId` NetworkId used by the fork [string] [default: "3030"] `--proposalId` The proposal id to execute [number] `--payloadAddress` The payload address to execute [string] `--artifactPath` The path to the artifact to execute [string] `--aclManagerAddress` ACL manager address [string] `--keepAlive` Keep the fork alive after this session [boolean] [default: false]

Pending Proposal

You can also simulate already proposed proposals that are still within the governance process simply by running `aave-tenderly-cli --proposalId 96`

Note: this command will only work on mainnet

Existing Payload

To create a fork with an already deployed proposal payload executed simply type `aave-tenderly-cli --payloadAddress 0xb8fe2a2104afb975240d3d32a7823a01cb74639f`

Note

: For payload targeting layer 2 you need to specify the [ACLManager](#).

Local Payload

BGD created a [payload](#) to freeze Aave v1 pools. While it's not deployed anywhere yet, you can just checkout the code, build and deploy the local artifact via `aave-tenderly-cli --artifactPath ./out/ProposalPayloadAaveFreezeV1.sol/ProposalPayloadAaveFreezeV1.json`

Note

: For payload targeting layer 2 you need to specify the [ACLManager](#).

Here is how you would execute the btcB proposal shown in the gif of the target market via `aave-tenderly-cli --networkId 43114 --artifactPath ./out/AaveV3AvaBTCBListingSteward.sol/AaveV3AvaBTCBListingSteward.json --aclManagerAddress 0xa72636CbcAa8F5FF95B2cc47F3CDEe83F3294a0B`

Simple fork

If you only want to create a fork to play around on the UI without executing a proposal or similar, you can do so by running `aave-tenderly-cli --keepAlive --networkId 1`

There are various other commands and special settings to accommodate more complicated use-cases like forking of a specific block, reusing an existing fork, or altering the networkId

used by the fork. You can use `--help`

to get a full picture of available settings.

We plan to keep updating the tool with new use cases as they appear, so we encourage the community to create issues on <https://github.com/bgd-labs/aave-tenderly-cli/issues> or create PRs on <https://github.com/bgd-labs/aave-tenderly-cli/pulls> !