Running a Builder

Information about running a builder for the Espresso Network

Overview

Participants creating blocks for the Espresso Network must run a builder, a piece of software which tracks the state of HotShot consensus so that it is able to and propose blocks at the correct time. The builder functions to create a block filled with transactions, drawing from transactions accessible in Espresso's public mempool as well as its own private mempool.

Espresso provides a simple builder implementation, which participants can run out-of-the-box or build their own software on top of. This document describes how to run the basic builder software.

For comprehensive guidance on the design of an Espresso builder, it is recommended to refer to the hotshot-builder-core repository. Furthermore, to gain insight into the process of enabling builder services for a sequencer, pertinent information can be found at espresso-sequencer-builder. For access to the most recent builder Docker images, please visit here.

Usage

Copy

Clone the espresso-sequencer repository

git clone https://github.com/EspressoSystems/espresso-sequencer

Build the executable in release mode

cargo build --release

Run a builder natively

target/release/permissionless-builder [options]

To understand more about the available builder options

target/release/permissionless-builder -h

Run a node with the builder docker image

docker run -it \

 [-e ENV=VALUE...] \

 ghcr.io/espressosystems/espresso-sequencer/builder:main

For a quick start, we recommend referring to the espresso-sequencer docker-compose file, and looking particularly at permissionless-builder.

Parameters & options

Environment variable

CLI flag

Description

ESPRESSO_SEQUENCER_HOTSHOT_EVENT_STREAMING_API_URL

--hotshot-event-streaming-url

URL of hotshot events API running on Espresso Sequencer DA committee node. A builder will subscribe to this server to receive hotshot events. (e.g. http://localhost:8081)

ESPRESSO_BUILDER_ETH_MNEMONIC

--eth-mnemonic

Mnemonic phrase for builder account (e.g. "test test test test test test test test test test test junk")

ESPRESSO_BUILDER_ETH_ACCOUNT_INDEX

--eth-account-index

Index of a funded account. **Note**: This account must be funded in *Espresso*, meaning ETH must be bridged from the L1

ESPRESSO_BUILDER_L1_PROVIDER

--l1-provider-url

A Url builder will use for RPC communication with L1 (e.g. http://demo-l1-network:8545)

ESPRESSO_SEQUENCER_STATE_PEERS

--state-peers

Peer nodes use to fetch missing state

ESPRESSO_SEQUENCER_CHAIN_ID

--chain-id

Unique identifier for an instance of the sequencer network

ESPRESSO_SEQUENCER_MAX_BLOCK_SIZE

--max-block-size

Maximum allowed size (in bytes) for a block

ESPRESSO_SEQUENCER_BASE_FEE

--base-fee

Minimum fee in WEI per byte of payload

ESPRESSO_BUILDER_SERVER_PORT

--port

Port to run builder server on through which sequencer node can query builder provided APIs (e.g 41003)

ESPRESSO_BUILDER_BOOTSTRAPPED_VIEW

--view-number

Bootstrapping View number (e.g. 0)

ESPRESSO_BUILDER_CHANNEL_CAPACITY

--channel-capacity

The most outstanding hotshot events a builder wants at a point in time (e.g. 1024)

ESPRESSO_BUILDER_WEBSERVER_RESPONSE_TIMEOUT_DURATION

--max-api-timeout-duratio

The amount of time a builder can wait before timing out a request to the API (e.g 1s)

ESPRESSO_BUILDER_BUFFER_VIEW_NUM_COUNT

--buffer-view-num-count

The number of views to buffer before a builder garbage collects its state (e.g. 15)

Hardware requirements

Hardware requirements are subject to change as new features are added, but for now we recommend the following:

- **Memory**: 4-8 GB
- **CPU:** 2-4 Cores