

# op-txproxy

[A passthrough proxy \(opens in a new tab\)](#) for the execution engine endpoint. This proxy does not forward all RPC traffic and only exposes a specific set of methods. Operationally, the ingress router should only re-route requests for these specific methods.

[proxyd](#) as an ingress router supports the mapping of specific methods to unique backends.

## Methods

### eth\_sendRawTransactionConditional

To safely expose this endpoint publicly, additional stateless constraints are applied. These constraints help scale validation rules horizontally and preemptively reject conditional transactions before they reach the sequencer.

Various metrics are emitted to guide necessary adjustments.

#### Authentication

The caller authenticates using any valid ECDSA-secp256k1 key, such as an Ethereum key. The signature is computed over the [EIP-191 \(opens in a new tab\)](#) hash of the request body. The calling address does not need to hold an Ethereum balance ; it is simply used for identification.

With the signature and signing address, the request is authenticated under the X-Optimism-Signature header with the value.

- Requests with a missing authentication header fail with the -32003
- (transaction rejected) json rpc error code.
- Requests with a mismatch in recovered signer and supplied public key will have the http request failed with status code 400 - Bad Request
- .

Currently, no authorization policies are implemented on this endpoint. However, the authentication mechanism is in place to allow for future implementation of policies such as allowlists, localized rate limits, and other potential restrictions.

#### Runtime shutoff

This service can be configured with a flag or environment variable to reject conditional transactions without needing to interrupt the execution engine. This feature is useful for diagnosing issues.

`--sendRawTxConditional.enabled` (default: true) (`OP_TXPROXY_SENDRAWTXCONDITIONAL_ENABLED`)

When disabled, requests will fail with the -32003 (transaction rejected) json rpc error code with a message stating that the method is disabled.

#### Rate limits

Even though the op-geth implementation of this endpoint includes rate limits, it is instead applied here to terminate these requests early.

`--sendRawTxConditional.ratelimit` (default: 5000) (`OP_TXPROXY_SENDRAWTXCONDITIONAL_RATELIMIT`)

#### Stateless validation

- Conditional cost is below the max
- Conditional values are valid (i.e min < max)
- Transaction target are only 4337 Entrypoint contracts

The motivating factor for this endpoint is to enable permissionless 4337 mempools, hence the restricted usage of this methods to just [Entrypoint \(opens in a new tab\)](#) transactions.

Please open up an issue if you'd like this restriction to be optional via configuration to broaden usage of this endpoint. When the request passes validation, it is passed through to the configured backend URL

`--sendRawTxConditional.backend` (`OP_TXPROXY_SENDRAWTXCONDITIONAL_BACKENDS`)

⚠ Per the [specification](#), conditional transactions are not gossiped between peers. Thus, if you use replicas in an active/passive sequencer setup, this request must be broadcasted to all replicas.

[proxyd](#) as an egress router for this method supports this broadcasting functionality.

## How it works

To start using op-txproxy , follow these steps:

### Build the binary or pull the Docker image

1. Run the following command to build the binary

make

build 1. This will build and output the binary under bin/op-txproxy 2. .

The image for this binary is also available as [a docker artifact \(opens in a new tab\)](#) .

### Configure

The binary accepts configuration through CLI flags, which also settable via environment variables. Either set the flags explicitly when starting the binary or set the environment variables of the host starting the proxy.

See [methods](#) on the configuration options available for each method.

### Start

start the service with the following command

op-txproxy

//

...

with

flags

if

env

variables

are

not

set

[op-txproxy OP Stack hacks](#)