

title: Usage

When connecting to different networks using mev-boost, specify the desired network using its dedicated flag. Here are the available Network Flags:

| Network | Flag | | ----- | ----- | | Mainnet | -mainnet | | Goerli Testnet | -goerli | | Sepolia Testnet | -sepolia | | Kiln | -kiln | | Ropsten | -ropsten |

You can add multiple relays comma-separated to the -relays flag, like this: -relays RELAY_URL_1,RELAY_URL_2. Refer to ETH staker [Relay List](#) for a list of available relays.

Mainnet Relay

Run mev-boost pointed at our [Mainnet Relay](#):

```
bash ./mev-boost -mainnet -relay-check -relays https://0xac6e77dfe25ecd6110b8e780608cce0dab71fdd5ebea22a16c0205200f2f8e2e3ad3b71d3499c54ad14d6c21b41a37ae@boost-relay.flashbots.net
```

Goerli testnet

Run mev-boost pointed at our [Goerli Relay](#):

```
bash ./mev-boost -goerli -relay-check -relays https://0xafa4c6985aa049fb79dd37010438cfebeb0f2bd42b115b89dd678dab0670c1de38da0c4e9138c9290a398ecd9a0b3110@builder-relay-goerli.flashbots.net
```

Sepolia testnet

Run mev-boost pointed at our [Sepolia Relay](#):

```
bash ./mev-boost -sepolia -relay-check -relays https://0x845bd072b7cd566f02faeb0a4033ce9399e42839ced64e8b2adcf859ed1e8e1a5a293336a49feac6d9a5edb779be53a@builder-relay-sepolia.flashbots.net
```

mev-boost CLI arguments

bash \$./mev-boost -help Usage of ./mev-boost: -addr string listen-address for mev-boost server (default "localhost:18550") -debug shorthand for "-loglevel debug" -genesis-fork-version string use a custom genesis fork version -goerli use Goerli -json log in JSON format instead of text -loglevel string minimum loglevel: trace, debug, info, warn/warning, error, fatal, panic (default "info") -mainnet use Mainnet -relay-check check relay status on startup and on the status API call -relay-monitors string relay monitor urls - single entry or comma-separated list (scheme://host) -relays string relay urls - single entry or comma-separated list (scheme://pubkey@host) -request-timeout-getheader int timeout for getHeader requests to the relay [ms] (default 950) -request-timeout-getpayload int timeout for getPayload requests to the relay [ms] (default 4000) -request-timeout-regval int timeout for registerValidator requests [ms] (default 3000) -sepolia use Sepolia -version only print version

Verifying Your Setup

You can check if you setup works by looking up the validator registration of your proposer using the [Relay Data API](#).

For example, let's suppose that your validator's public key is 0xb606e206c2bf3b78f53ebff8be8e8d4af2f0da68646b5642c4d511b15ab5ddb122ae57b48eab614f8ca5bafbe75a5999. You can check if your validator is registered by querying the following endpoint:

url https://boost-relay.flashbots.net/relay/v1/data/validator_registration?pubkey=0xb606e206c2bf3b78f53ebff8be8e8d4af2f0da68646b5642c4d511b15ab5ddb122ae57b48eab614f8ca5bafbe75a5999

Below is the response:

```
json { "message": { "fee_recipient": "0x6db5c947ba388f0e708c03339d534af9fae5679c", "gas_limit": "30000000", "timestamp": "1663063595", "pubkey": "0xb606e206c2bf3b78f53ebff8be8e8d4af2f0da68646b5642c4d511b15ab5ddb122ae57b48eab614f8ca5bafbe75a5999" }, "signature": "0x94f1f635336f8c38909842106550026bbcb198f892c6a79c22a3365e547e87665e31fecf8da3533ef3cf8e2c965e03d1594cf8e981dfc75011b4d237552386377bcea714bcfb78d303fe5624625e1bbb8b23f3a6d1b7c9efb3d76cad4ca01a5" } }
```

The fee_recipient field should match the address you provided when registering your validator. If it does, then your setup is working correctly.