

# JSON-RPC server

Interacting with Nethermind requires using the JSON-RPC 2.0 protocol. Nethermind provides JSON-RPC over [HTTP](#), [WebSocket](#), and [IPC socket](#) transports. Each transport must be enabled with the respective configuration option, as shown below. For more details, see the [JSON-RPC configuration options](#).

The JSON-RPC API methods are grouped into several categories (namespaces) depending on their purpose. All API method names are composed of the namespace and the actual method name in that namespace. For instance, the `theeth_call` method belongs to the `theeth` namespace. See the sidebar for all supported namespaces and methods.

note Not all of the JSON-RPC namespaces are enabled by default. Instead, they must be enabled explicitly with the `--JsonRpc.EnabledModules` command line option. Otherwise, error code-32600 is returned. The enabled namespaces can be found in the configuration file specified with the `--config` command line option.

## Transports

tip The right choice of transport depends on the specific use case.

- HTTP is a familiar and idempotent transport that closes connections between requests and can, therefore, have lower overall overhead for a relatively low number of requests.
- WebSocket provides a continuous open channel that enables event subscriptions and streaming and handles large volumes of requests with more negligible per-message overhead.
- IPC is generally the most secure as it is limited to local interactions and cannot be exposed to external traffic. It can also be used for event subscriptions.

## HTTP

HTTP is the most widely used transport for Nethermind. To enable the HTTP server, use the `--JsonRpc.Enabled true` command line option. By default, Nethermind uses local loopback (`127.0.0.1` or `localhost`) and `8545` port. To use a different host or port, set the `--JsonRpc.Host` and `--JsonRpc.Port` command line options, respectively.

## WebSocket

The configuration of the WebSocket server follows the same pattern as the HTTP server. The WebSocket server is enabled automatically when the HTTP server is enabled and uses the same host and port. To disable the WebSocket server, use the `--Init.WebSocketsEnabled false` command line option. To use a different port, set the `--JsonRpc.WebSocketsPort` command line option.

## IPC socket

Nethermind uses IPC based on Unix domain socket. To enable the IPC server, use the `--JsonRpc.IpcUnixDomainSocketPath path/to/ipc` command line option. If the `path/to/ipc` doesn't exist, Nethermind creates one.

## Engine API

The Engine API is a set of RPC methods that enable communication between an execution and consensus client. The clients call these methods automatically when they need to exchange information. Engine API is enabled automatically by default and is not designed to be exposed to the user.

By default, the Engine API uses local loopback (`127.0.0.1` or `localhost`) and `8551` port. To use a different host or port, set the `--JsonRpc.EngineHost` and `--JsonRpc.EnginePort` command line options, respectively. For example, this can be useful when execution and consensus clients are on different machines.

Important When the `--JsonRpc.EngineHost` option is specified, the `--JsonRpc.EnginePort` option must be specified as well. The Engine API uses JWT authentication and requires a JWT secret. By default, Nethermind creates one at `atkeystore/jwt-secret` path in its root directory. To use a different path, specify the `--JsonRpc.JwtSecretFile path/to/jwt.hex` command line option. [Edit this page](#) Last updated on Feb 17, 2024 [Previous](#) [Private networks](#) [Next](#) [admin](#)