
nethermind Documentation

Release 0.9.2

Tomasz K. Stanczak

Nov 18, 2018

Contents

1	ToC	1
1.1	Download	1
1.2	Releases	1
1.3	Build	1
1.4	Networks	2
1.5	Configuration	3
1.6	Docker	5

1.1 Download

You can download the latest packages for all supported platforms from [here](#)

1.2 Releases

You can find the latest Nethermind releases [here](#).

1.3 Build

1.3.1 IDE

- JetBrains Rider <https://www.jetbrains.com/rider/>
- VS Code <https://code.visualstudio.com/docs/other/dotnet>

1.3.2 SDKs

- Windows https://www.microsoft.com/net/download?initial-os=window* Linux <https://www.microsoft.com/net/download?initial-os=linux> (make sure to select the right distribution)
- Mac <https://www.microsoft.com/net/download?initial-os=macos>

1.3.3 Linux

```
sudo apt-get update && sudo apt-get install libsappy-dev libc6-dev libc6
git clone https://github.com/tkstanczak/nethermind --recursive
cd nethermind/src/Nethermind
dotnet build -c Release
cd Nethermind.Runner
dotnet run
```

1.3.4 MacOS

```
brew install gmp
git clone https://github.com/tkstanczak/nethermind --recursive
cd nethermind/src/Nethermind
dotnet build -c Release
cd Nethermind.Runner
dotnet run
```

1.3.5 Windows

you may need to install <https://support.microsoft.com/en-us/help/2977003/the-latest-supported-visual-c-downloads>

```
git clone https://github.com/tkstanczak/nethermind --recursive
cd nethermind/src/Nethermind
dotnet build -c Release
cd Nethermind.Runner
dotnet run
```

1.4 Networks

By default Nethermind launches with mainnet network configuration but you can sync any of the below networks by adding a command line switch:

```
--config name_of_the_network
```

Network name can be any of the following

- mainnet
- rinkeby
- goerli
- ropsten

1.4.1 Mainnet

This is the main public Ethereum network secured by the ethash PoW algorithm.

```
Nethermind.Runner --config mainnet
```

1.4.2 Görli (goerli)

This is a Clique-PoA based testnet supported by all major clients. It has a 15 seconds delay between blocks that are sealed by Goerli validators.

```
Nethermind.Runner --config goerli
```

1.4.3 Ropsten

This is a test network secured by ethash PoW algorithm which resembles mainnet most closely but is prone to hashrate-based attacks.

```
Nethermind.Runner --config ropsten
```

1.4.4 Rinkeby

This is a Clique-PoA based testnet supported by Geth and Nethermind. It has a 15 seconds delay between blocks.

```
Nethermind.Runner --config rinkeby
```

1.4.5 Kovan

This is an Aura-PoA based testnet supported by Parity only.

1.5 Configuration

1.5.1 InitConfig

- `JsonRpcEnabled` - enables RPC endpoints (configured also through `HttpHost` and `HttpPort`)
- `NetworkEnabled` - if disabled then all networking (discovery, peer manager, sync) is disabled
- `DiscoveryEnabled` - enables / disables discovery protocol for finding new peers on the network
- `SynchronizationEnabled` - enables / disabled eth62/ eth63 synchronizations (if disabled then new blocks will not be downloaded and downloaded)
- `ProcessingEnabled` - enables / disables processing of downloaded blocks
- `IsMining` - enables blocks production (by validators in PoA networks and miners in PoW networks)
- `P2PPort` - port to listen for incoming connections in devp2p protocols (most typically 30303)
- `DiscoveryPort` - port to listen for incoming connections in discovery protocol
- `ChainSpecPath` - path to a chainspec file that defines network properties like genesis block, initial allocations, network id, sealing engine
- `GenesisHash` - if left empty no genesis block hash check is done, if set then chainspec genesis processing result is validated against the hash provided
- `BaseDbPath` - base path for all database directories (it is combined with `baseDbPath` from command line)
- `LogFileName` - name of the log file (e.g. mainnet.log)

- `ObsoletePendingTransactionInterval` - if transaction is known to be older than this many seconds then we do not broadcast it after hearing about it from other nodes
- `RemovePendingTransactionInterval` - time (in seconds) after which pending transactions are removed from memory
- `PeerNotificationThreshold` - percentage of peers that will be notified about new pending transactions

1.5.2 DbConfig

- `WriteBufferSize` - size of a single memory buffer for RocksDB data
- `WriteBufferNumber` - number of RocksDB write buffers
- `BlockCacheSize` - size of the data block cache for RocksDB
- `CacheIndexAndFilterBlocks` - set to true to limit memory taken by RocksDB to the size of `WriteBufferSize * WriteBufferNumber + BlockCacheSize`, otherwise index and filter blocks may take significant amount of memory (many gigabytes)

1.5.3 Sample configuration (mainnet)

```
[
  {
    "ConfigModule": "InitConfig",
    "ConfigItems": {
      "JsonRpcEnabled": false,
      "NetworkEnabled": true,
      "DiscoveryEnabled": true,
      "SynchronizationEnabled": true,
      "PeerManagerEnabled": true,
      "ProcessingEnabled": true,
      "IsMining": false,
      "DiscoveryPort": 30312,
      "P2PPort": 30312,
      "HttpHost": "127.0.0.1",
      "HttpPort": 8345,
      "ChainSpecPath": "chainspec/foundation.json",
      "GenesisHash":
      ↪ "0xd4e56740f876aef8c010b86a40d5f56745a118d0906a34e69aec8c0db1cb8fa3",
      "BaseDbPath": "nethermind_db/mainnet",
      "LogFileName": "mainnet.logs.txt",
      "ObsoletePendingTransactionInterval": 15,
      "RemovePendingTransactionInterval": 600,
      "PeerNotificationThreshold": 20
    }
  },
  {
    "ConfigModule": "DbConfig",
    "ConfigItems": {
      "WriteBufferSize": 67108864,
      "WriteBufferNumber": 6,
      "BlockCacheSize": 67108864,
      "CacheIndexAndFilterBlocks": true
    }
  }
]
```


1.6 Docker

<https://hub.docker.com/r/nethermind/nethermind.runner/>

Docker pull command:

```
docker pull nethermind/nethermind.runner
```

In order to start Nethermind.Runner, simply run:

```
docker run -it nethermind/nethermind.runner
```

To enable JSON RPC, publish port 8345 and set `NETHERMIND_INITCONFIG_JSONRPCENABLED=true`:

```
docker run -it -p 8345:8345 -e NETHERMIND_INITCONFIG_JSONRPCENABLED=true nethermind/  
↪nethermind.runner
```

To enable P2P communication, additionally, publish port 30312.

To switch the network, set `NETHERMIND_CONFIG` variable (default value is mainnet).

Available configurations

- mainnet
- goerli
- rinkeby
- ropsten
- mainnet

For example:

```
docker run -it -e NETHERMIND_CONFIG=goerli nethermind/nethermind.runner
```

It's possible to modify each configuration property via environment variable, using a simple convention:

```
NETHERMIND_{MODULE}_{PROPERTY}
```

For example:

```
NETHERMIND_INITCONFIG_NETWORKENABLED=true
```

You can find all of the possible settings [here](#).

To fully override a configuration file, you need to use a volume:

```
-v $/home/mainnet.cfg:/app/configs/mainnet.cfg
```

In order to map an internal database to a local volume, you need to map `/app/nethermind_db/`:

```
-v /tmp/nethermind_db:/app/nethermind_db
```