Logs

warning This article requires a revision.

Log config file location

Logging in Nethermind is done via NLog library that can be configured by editing the NLog.config file.

Environment Type NLog.config location built from src - Debug mode src\Nethermind\Nethermind.Runner\bin\Debug\netcoreapp3.1\NLog.config built from src - Release mode src\Nethermind\Nethermind.Runner\bin\Release\netcoreapp3.1\NLog.config PPA /usr/share/nethermind/NLog.config Docker /nethermind/NLog.config fromdownloads page top level directory after unzipping the package fromGitHub releases page top level directory after unzipping the package dAppNode ? [to be documented]

Log config file syntax

Detailed NLog configuration options can be found here:https://nlog-project.org/config/

Config or CLI log rules

Simple logging rules can be added through configuration file or command line argument.

For example this would addTrace level logs to any logger underSynchronization module andDebug level logs forBlockTree fromBlockchain module: --Init.LogRules Synchronization.*:Trace;Blockchain.BlockTree:Debug

Global logging override

Additionally there are global logging override that you can use temporarily:

Command line override Log level ./Nethermind.Runner --config mainnet --log TRACE TRACE ./Nethermind.Runner --config mainnet --log DEBUG DEBUG ./Nethermind.Runner --config mainnet --log INFO INFO ./Nethermind.Runner --config mainnet --log WARN WARN ./Nethermind.Runner --config mainnet --log ERROR ERROR

JSON RPC logging level

This can be done by including these lines in the logging configuration file:

Enterprise Logging

See how to configure Sehere

Explaining Nethermind logs

You can check the supported operating systems, architectures and hardware requirements heresystem-requirements.md

After the node starts, you will see some initial info about the node and then the sync will start. Görli fast sync uses afast blocks sync mode initially. Thefast blocks sync picks some knownpivot block from the past and downloadsheaders ,bodies , andreceipts downwards all the way togenesis block . All blocks from 0 to thepivot block are showed asOld Headers in thefast blocks sync logs. The console display shows the number growing from 0 to pivot, but this is just to make the display more user-friendly.

You will see some information about the sync progress, like below:

1. Shows the number of already downloadedheaders
2. ,bodies
3. andreceipts
4. under the nameDownloaded
5. out of all to
6. be downloaded in the fast blocks stage.
7. Shows the current queue of already downloadedblocks
8. ,headers
9. andreceipts
10. waiting for being saved to the
11. database.
12. Shows the current download speed (blocks per second - bps).

13. Shows the average download speed (blocks per second - bps).

When the fast blocks stage finishes, there will be some period of downloading blocks between thepivot and thelatest blocks which will have some additional info:

1. Shows the last entry from the fast blocks stage.
2. Shows the mode transition moment.
3. Displays the speed (in blocks per second) of allheaders
4. ,bodies
5. andreceipts
6. at the same time.
7. Additional info will appear every 30000 blocks with information about the Görli epoch being stored.

After thefast sync part finished, the node will transition to thestate sync stage when thestate trie is being downloaded. Much information is displayed about the progress, as this process may take a long time on mainnet (a few hours).

1. Total elapsed time instate sync
2. is displayed.
3. The total percentage of downloaded DB size is displayed (on mainnet the sync finishes around 34GB in March 2020, on
4. Görli around 800MB).
5. branches
6. stands for the percentage of downloaded branches.
7. Download speed in kilobytes per second is displayed.
8. accounts
9. stands for the number of accounts data downloaded already.
10. nodes
11. stands for the number ofPatricia trie nodes
12. downloaded by the sync process.
13. diagnostic
14. shows the time spent in the DB write / read access. The higher the value, the worse. It may get much
15. worse if you restart the node during the sync process, as we need to recreate some caches then by reading data from
16. the DB.

When the state sync is nearing completion, you may see a series ofbranch sync information reloading many times from 0% to nearly 100%. This is the node trying to retrieve the few remaining state nodes and progressing with the head block rapidly:

At some point, the entire state is downloaded and the node enters thefull sync mode and will allow you to issue CLI / Web3 queries and send / receive transactions

1. Theroot
2. is saved at the moment when the entirePatricia trie
3. is downloaded.
4. We also clearly state that the node transitions to thefull sync
5. .
6. When you see the block being processed, then you are in thefull sync
7. and the newly arrived block is being
8. calculated.
9. Every two minutes you will see a summary of connected peers with their client version, IP address, highest synced
10. block, and data download speeds.

Also, every now and then, a peer report will appear like below:

1. First bracket is for Allocated contexts. It has possible values ofH
2. for Headers,B
3. for Bodies,R
4. for
5. Receipts,N
6. for State,S
7. for Snap, andW
8. for Witness.
9. Second bracket is for Sleeping contexts. It has possible values ofH
10. for Headers,B
11. for Bodies,R
12. for
13. Receipts,N
14. for State,S
15. for Snap, andW
16. for Witness.

17. Third bracket contains Peer Info.
18. Fourth bracket is for Speeds as Follows:* Latency
19.
    - Headers Transfer
20.
    - Bodies Transfer
21.
    - Receipts Transfer
22.
    - Node Data Transfer
23.
    - Snap Ranges Transfer
24. Fifth bracket is for Client Info like Client Name, Client Version, Operating System and Language Version.Edit this page Last updatedonFeb 17, 2024 Previous Security Next Database