Troubleshooting

File descriptor limits

In some cases, file descriptor limits may cause errors like "Too many open files". To solve that, see the instructions for your platform below.

- Linux
- macOS

To increase the limits for the user running Nethermind (given the process name of nethermind), run:

sudo

echo

"nethermind soft nofile 100000"

/etc/security/limits.d/nethermind.conf sudo

echo

"nethermind hard nofile 100000"

/etc/security/limits.d/nethermind.conf To increase the limits, run:

ulimit

-n

10000 If you run into issues with the above command, see the workaround .

Note that the changes above are temporary and will be reset after the system reboot. To make them permanent, you can add them to your ~/.bashrc or ~/.bash_profile shell configuration file.

Database corruption issues

Database corruption is one of the issues that happen now and then; it has many possible causes among them:

- Hardware failures: disk failures, memory errors, hardware overheating, etc.
- Power cuts and abrupt shutdowns

There's no shortcut in such situations, and resyncing Nethermind from scratch is the recommended remedy.

Issues with lock files

If Nethermind complains about the lock files, it perhaps because of one of the following:

- Another Nethermind process is running using the same database
- The database has not been appropriately closed on the last run.
- In this case, run the following command from the Nethermind database directory:
- find
- .
- -type
- f
- -name
- 'LOCK'
- -delete

Block checksum mismatch

Sometimes Nethermind may fail with an error similar the following:

Corruption: block checksum mismatch: expected 2087346143, got 2983326672 in... This tends to happen on XFS file systems under very high memory pressure. The issue can be mitigated by using the -- Db.UseDirectIoForFlushAndCompactions true option although at the cost of performance.

However, quite often, this is because of memory module issues. Edit this page Last updated on Mar 26, 2024 Previous FAQ