Docker setup

This page has instructions to run celestia-node using Docker. If you are looking for instructions to run celestia-node using a binary, please refer to thecelestia-node page .

Using Docker is the easiest way to run celestia-node for most users. Docker is a containerization platform that allows you to run celestia-node in an isolated environment.

This means that you can run celestia-node on your machine without having to worry about installing and configuring all of the dependencies required to run the node.

If you would like to learn more about key management in Docker, visit theDocker andcel-key section .

The easiest way to install Docker is to use the Docker Desktop installer or Ubuntu. You canfollow the instructions for your operating system .

Prerequisites

- Docker Desktop for Mac or Windows
- and a basic understanding of Docker
- Docker Engine for Linux
- and a basic understanding of Docker

Quick start

1. Setthe network
2. you would like to run your node on:
3. Mainnet Beta
4. Mocha
5. Arabica
6. bash
7. export
8. NETWORK
9. =
10. celestia
11. export
12. NETWORK
13. =
14. celestia
15. bash
16. export
17. NETWORK
18. =
19. mocha
20. export
21. NETWORK
22. =
23. mocha
24. bash
25. export
26. NETWORK
27. =
28. arabica
29. export
30. NETWORK
31. =
32. arabica
33. Set the node type
34. Light
35. Bridge
36. Full
37. bash
38. export
39. NODE_TYPE
40. =
41. light
42. export

NODE_TYPE
=
light
bash
export
NODE_TYPE
=
bridge
export
NODE_TYPE
=
bridge
bash
export
NODE_TYPE
=
full
export
NODE_TYPE
=
full
Set an RPC endpoint for either Mainnet Beta
, Mocha
, or Arabica
using the bare URL (without http or https):
bash
export
RPC_URL
=
this-is-an-rpc-url.com
export
RPC_URL
=
this-is-an-rpc-url.com
Run the image from the command line:
Mainnet Beta
Mocha
Arabica
bash
docker
run
-e
NODE_TYPE=
NODE_TYPE
-e
P2P_NETWORK=
NETWORK
\
ghcr.io/celestiaorg/celestia-node:v0.12.4
\
celestia
NODE_TYPE
start
--core.ip
RPC_URL
--p2p.network
NETWORK
docker
run
-e
NODE_TYPE=
NODE_TYPE
-e
P2P_NETWORK=
NETWORK
\
ghcr.io/celestiaorg/celestia-node:v0.12.4
\

```
111. celestia
112. NODE_TYPE
113. start
114. --core.ip
115. RPC_URL
116. --p2p.network
117. NETWORK
118. bash
119. docker
120. run
121. -e
122. NODE_TYPE=
123. NODE_TYPE
124. -e
125. P2P_NETWORK=
126. NETWORK
127. \
128. ghcr.io/celestiaorg/celestia-node:v0.13.1
129. \
130. celestia
131. NODE_TYPE
132. start
133. --core.ip
134. RPC_URL
135. --p2p.network
136. NETWORK
137. docker
138. run
139. -e
140. NODE_TYPE=
141. NODE_TYPE
142. -e
143. P2P_NETWORK=
144. NETWORK
145. \
146. ghcr.io/celestiaorg/celestia-node:v0.13.1
147. \
148. celestia
149. NODE_TYPE
150. start
151. --core.ip
152. RPC_URL
153. --p2p.network
154. NETWORK
155. bash
156. docker
157. run
158. -e
159. NODE_TYPE=
160. NODE_TYPE
161. -e
162. P2P_NETWORK=
163. NETWORK
164. \
165. ghcr.io/celestiaorg/celestia-node:v0.13.1
166. \
167. celestia
168. NODE_TYPE
169. start
170. --core.ip
171. RPC_URL
172. --p2p.network
173. NETWORK
174. docker
175. run
176. -e
177. NODE_TYPE=
178. NODE_TYPE
```

179. -e
180. P2P_NETWORK=
181. NETWORK
182. \
183. ghcr.io/celestiaorg/celestia-node:v0.13.1
184. \
185. celestia
186. NODE_TYPE
187. start
188. --core.ip
189. RPC_URL
190. --p2p.network
191. NETWORK

Congratulations! You now have a celestia-node running!

If you would like to run the node with custom flags, you can refer to thecelestia-node tutorial page. Refer tothe ports section of the celestia-node troubleshooting page for information on which ports are required to be open on your machine.

Light node setup with persistent storage

If you delete a container that you started above, all data will be lost. To avoid this, you can mount a volume to the container. This will allow you to persist data even after the container is deleted.

First, you will need to create a directory on your host machine. This directory will be used to store the data for the container. Create a directory on your host machine and give it a name. For example, you can name itmy-node-store :

bash cd HOME mkdir

my-node-store cd HOME mkdir

my-node-store Now, you can mount this directory to the container. Before mounting a volume, youmay need to set permissions for the user on the host machine by running:

Docker Engine on Linux

Docker Desktop on Mac bash sudo

chown

10001 :10001 HOME /my-node-store sudo

chown

10001 :10001 HOME /my-node-store bash

you're good to go 😎

you're good to go 😎

Initialize the node store and key

In order to mount a volume to the container, you need to specify the path to the volume. When you run your container, you can specify the path to the volume using the--volume (or-v for short) flag. In this command, we'll create our key and initialize the node store, using the variables we set in thequick start section:

bash

--volume == -v [local path]:[container path]

docker

run [args...] -v HOME/my-node-store:/home/celestia \ celestia NODE_TYPE init [args...]

--volume == -v [local path]:[container path]

docker

run [args...] -v HOME/my-node-store:/home/celestia \ celestia NODE_TYPE init [args...] An example init command will look similar to below:

Mainnet Beta

Mocha

Arabica bash docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.12.4

\ celestia

light

init

--p2p.network NETWORK docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.12.4

\ celestia

light

init

--p2p.network NETWORK bash docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.13.1

\ celestia

light

init

--p2p.network NETWORK docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.13.1

\ celestia

light

init

--p2p.network NETWORK bash docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.13.1

\ celestia

light

init

--p2p.network NETWORK docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.13.1

\ celestia

light

init

--p2p.network NETWORK

Start the node

Run the following command to start the node:

bash

--volume == -v [local path]:[container path]

docker

run [...args] -v HOME/my-node-store:/home/celestia \ celestia

< node-typ e

start [...args]

--volume == -v [local path]:[container path]

docker

run [...args] -v HOME/my-node-store:/home/celestia \ celestia

< node-typ e

start [...args] A full start command will look similar to below.

Mainnet Beta

Mocha

Arabica bash docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.12.4

\ celestia

light

start

--core.ip RPC_URL docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.12.4

\ celestia

light

start

--core.ip RPC_URL bash docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.13.1

\ celestia

light

start

--core.ip RPC_URL docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.13.1

\ celestia

light

start

--core.ip RPC_URL bash docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.13.1

\ celestia

light

start

--core.ip RPC_URL docker

run

-e

NODE_TYPE= NODE_TYPE -e

P2P_NETWORK= NETWORK \ -v HOME /my-node-store:/home/celestia

\ ghcr.io/celestiaorg/celestia-node:v0.13.1

\ celestia

light

start

--core.ip RPC_URL Congratulations! You now have a node running with persistent storage.

Video walkthrough

2.5 minute version

Troubleshooting

For security purposes Celestia expects to interact with the your node's keys in a read-only manner. This is enforced using linux style permissions on the filesystem. Windows NTFS does not support these types of permissions. As a result the recommended path for Windows users to mount a persisted volume is to do so within WSL. You can findinstructions for installing WSL . [][Edit this page on GitHub] Last updated: Previous page Install celestia-app Next page Networks overview []