Sei Node Setup Guide

System Requirements

CPU Cores RAM Disk 16 cores 64GB 1TB NVMe

Build Version and Genesis Table

Network Version Chain ID Genesis URL Mainnet v5.5.2 pacific-1 Genesis Testnet v5.5.1 atlantic-2 Genesis Devnet v5.5.1 arctic-1 Genesis

Getting Started

The following is intended for Debian-based systems. Others like MacOS or Archlinux will differ slightly

Update and Upgrade System Packages

apt

update

&&

apt

upgrade

-y

Install Dependencies [use a package manager likeapt

]

apt

install

nano

make

build-essential

gcc

git

jq

chrony

tar

curl

lz4

wget

-y

Install Golang [do not use a package manager for this step]

1. Remove old Go version if necessary:
2. rm
3. -rvf
4. /usr/local/go/
5. Install Golang:
6. wget
7. https://golang.org/dl/go1.21.1.linux-amd64.tar.gz
8. tar
9. -C
10. /usr/local
11. -xzf
12. go1.21.1.linux-amd64.tar.gz
13. rm
14. go1.21.1.linux-amd64.tar.gz
15. Configure PATH and GOPATH (add to~/.profile
16. or~/.bashrc
17. to make persistent):
18. export
19. GOROOT
20. =
21. /usr/local/go

22. export
23. GOPATH
24. =
25. HOME
26. /go
27. export
28. GO111MODULE
29. =
30. on
31. export
32. PATH
33. =
34. PATH
35. :/usr/local/go/bin:
36. HOME
37. /go/bin

Installseid

1. Define the variables for your network and (optional) a moniker or name to assign your node:Replace these with real values, found in the reference table
2. above
3. VERSION
4. =
5. v1.2.3
6. CHAIN_ID
7. =
8. mainnet-1
9. GENESIS_URL
10. =
11. https://example.com/genesis.json
12. MONIKER
13. =
14. "your node name"
15. Clone the repository and install:
16. cd
17. ~/
18. &&
19. git
20. clone
21. https://github.com/sei-protocol/sei-chain.git
22. cd
23. sei-chain
24. git
25. checkout
26. VERSION
27. make
28. install

Initialize Node

1. Initialize the node:
2. seid
3. init
4. MONIKER
5. --chain-id
6. CHAIN_ID
7. Download and place genesis file:
8. wget
9. -O
10. genesis.json
11. GENESIS_URL
12. mv
13. genesis.json
14. ~/.sei/config

For light-client setup stop here, and add an RPC connection toclient.toml as a final step.

Configure Node

1. Set persistent peers:
2. sed
3. -i
4. '/^# Comma separated list of nodes to keep persistent connections to/,/^/ s/^persistent-peers = ""/persistent-peers = "de8b1df70c7a8817ed121908e7c6e6059f4238f9@3.142.50.176:26656,7a962f3a928ca4e0e58355e6e798aba1ea253272@34.242.85.117:26656"/'
5. ~/.sei/config/config.toml
6. Enablesei-db
7. (dependent on snapshot provider):
8. sed
9. -i
10. 's/^sc-enable = false/sc-enable = true/'
11. ~/.sei/config/app.toml

Create Systemd Service

1. Create the service file:
2. nano
3. /etc/systemd/system/seid.service
4. Add the following content:
5. [Unit]
6. Description=
7. "Sei Daemon"
8. After=
9. network-online.target
10. [Service]
11. User=
12.
13. ExecStart=
14. /home//go/bin/seid start
15. Restart=
16. always
17. RestartSec=
18. 3
19. LimitNOFILE=
20. 8192
21. [Install]
22. WantedBy=
23. multi-user.target

Download & Apply Snapshot

Find a snapshot from a provider likePolkachu(opens in a new tab) , and either download, or define SNAP_URL with it.

1. Download snapshot:
2. wget
3. -O
4. SNAP_URL SNAP
5. Stop the node (if running as systemd service):
6. systemctl
7. stop
8. seid
9. Unpack snapshot to location:
10. lz4
11. -c
12. -d
13. SNAP
14. |
15. tar
16. -x
17. -C
18. HOME
19. /.sei
20. Start service and confirm operation:
21. systemctl
22. start
23. seid
24. systemctl
25. status
26. seid
27. Remove snapshot archive:
28. rm
29. -v
30. SNAP

Appendix

Node Types

- RPC / Full Nodes:
- Used for querying data or interacting with the chain. Default settings run RPC / full nodes.
- Archive Nodes:
- Maintain full state from genesis. Setmin-retain-blocks=0
- andpruning="nothing"
- inapp.toml
- .
- State Sync Nodes:
- Provide snapshot data for other nodes to bootstrap. Set a value greater than zero forsnapshot-interval
- under[statesync]
- inapp.toml
- .
- Validator Nodes:
- Secure the chain by proposing and signing blocks. Setmode=validator
- inconfig.toml
- .

Default Service Ports

The standard service ports can be manually configured inHOME/.sei/config/config.toml andHOME/.sei/config/app.toml :

- 26656
- : P2P
- 26657
- : RPC
- 1317
- : REST
- 9090
- : GRPC
- 8545
- : EVM RPC
- 8546
- : EVM Websocket
- 26660
- : Tendermint Prometheus Metrics Exporter

The standard websocket rides on the same connection as the RPC server. Example: [non-TLS]wss://localhost:26657/websocket .