[#](#)

Introduction

The iris executable program is the entry point for running an IRIShub node. All the validator nodes and full nodes need to install the iris and launching the daemon to join the IRIShub network. You can also use iris to start your own test network locally.

[#](#)

Hardware Requeirment

It's recommended that you run irishub nodes on Linux Server.

[#](#)

Minimum Requeirment

- 2 CPU
- Memory: 6GB
- Disk: 256GB SSD
- OS: Ubuntu 16.04 LTS +
- Bandwidth: 20Mbps
- Allow all incoming connections on TCP port 26656 and 26657

[#](#)

Home Directory

The home directory is the working directory of the iris node. The home directory contains all the configuration information and all the data that the node runs.

In the iris command, you can specify the home directory of the node by using flag --home . If you run multiple nodes on the same machine, you need to specify different home directories for them. If the --home flag is not specified in the iris command, the default value HOME/.iris is used as the home directory.

The iris init command is responsible for initializing the specified --home directory and creating the default configuration files. Except the iris init command, the home directory used by any other iris sub commands must be initialized, otherwise an error will be reported.

The data of the IRIShub node is stored in the data directory of the home, including blockchain data, application layer data, and index data.

All configuration files are stored in the config directory:

[#](#)

genesis.json

genesis.json defines the genesis block data, which specifies the system parameters such as chain_id, consensus parameters, initial account token allocation, creation of validators, and parameters for modules. See [genesis-file](#) for details.

[#](#)

node_key.json

node_key.json is used to store the node's key. The node-id queried by iris tendermint show-node-id is derived by the key, which is used to indicate the unique identity of the node. It is used in p2p connection.

[#](#)

priv_validator.json

pri_validator.json is the [Tendermint Key](#) file that the validator will use to sign Pre-vote/Pre-commit in each round of consensus voting. As the consensus progresses, the tendermint consensus engine will continuously update last_height /last_round /last_step values.

[#](#)

config.toml

config.toml is the non-consensus configuration of the node. Different nodes can configure themselves according to their own situation. Common modifications arepersistent_peers ,moniker ,laddr .

[#](#)

app.toml

app.toml provides base configuration, telemetry configuration, API configuration, gRPC configuration and state sync configuration for IRIShub.