

# **Luganodes Project Development**

**Name: Kiran Manikandhan**

**Reg No: 20BCE1786**

**VIT Chennai**

## **Introduction:**

The Cosmos Hub, a proof-of-stake blockchain, is powered by its native ATOM cryptocurrency. Cosmos aims to create a network of crypto networks united by open-source tools for streamlining transactions between them. Customizability and Interoperability sets Cosmos apart from other projects.

Cosmos aims to foster an ecosystem of networks that can share data and tokens programmatically, with no central party facilitating the activity rather than prioritizing its own network.

## **Setting up a node in Cosmos:**

1) Installing tools and Go language in a Linux-based virtual machine.

- Installed Go version: 1.20.3.
- 'Make' command for compiling different sub-programs and building a final executable.

```

(kiran@kali)-[~]
$ sudo apt-get install -y make gcc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gcc is already the newest version (4:12.2.0-3).
make is already the newest version (4.3-4.1).
The following package was automatically installed and is no longer required:
  ruby3.0
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 1741 not upgraded.

(kiran@kali)-[~]
$ wget https://go.dev/dl/go1.20.3.linux-amd64.tar.gz
--2023-08-01 13:43:13-- https://go.dev/dl/go1.20.3.linux-amd64.tar.gz
Resolving go.dev (go.dev)... 216.239.36.21, 216.239.38.21, 216.239.32.21, ...
Connecting to go.dev (go.dev)|216.239.36.21|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://dl.google.com/go/go1.20.3.linux-amd64.tar.gz [following]
--2023-08-01 13:43:14-- https://dl.google.com/go/go1.20.3.linux-amd64.tar.gz
Resolving dl.google.com (dl.google.com)... 142.250.182.46, 2404:6800:4007:81a::200e
Connecting to dl.google.com (dl.google.com)|142.250.182.46|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 100142274 (96M) [application/x-gzip]
Saving to: 'go1.20.3.linux-amd64.tar.gz'

go1.20.3.linux-amd64.tar.gz 100%[=====] 95.50M 425KB/s

2023-08-01 13:46:54 (446 KB/s) - 'go1.20.3.linux-amd64.tar.gz' saved [100142274/100142274]

(kiran@kali)-[~]
$ sudo rm -rf /usr/local/go 66 sudo tar -C /usr/local -xzf go1.20.3.linux-amd64.tar.gz

(kiran@kali)-[~]
$ export PATH=$PATH:/usr/local/go/bin

```

## 2) Installation and Configuration:

### a. Build gaiad binary and initialize chain home folder.

```

git clone https://github.com/cosmos/gaia
cd gaia
# To sync from genesis, comment out the next line.
git checkout v10.0.2
# To sync from genesis, uncomment the next line and skip the State S
# git checkout v6.0.4
make install
export PATH=$PATH:$HOME/go/bin
gaiad init <custom_moniker>

```

```

(kiran@kali)-[~/go]
$ git clone https://github.com/cosmos/gaia
Cloning into 'gaia' ...
remote: Enumerating objects: 156150, done.
remote: Counting objects: 100% (8202/8202), done.
remote: Compressing objects: 100% (2422/2422), done.

```

```

(kiran@kali)~/gaia
$ export PATH=$PATH:$HOME/go/bin

(kiran@kali)~/gaia
$ gaiad init <custom_moniker>
sh: parse error near '\n'

(kiran@kali)~/gaia
$ gaiad init
error: accepts 1 arg(s), received 0
usage:
  gaiad init [moniker] [flags]

Flags:
  -chain-id string    genesis file chain-id, if left blank will be randomly created
  -h, --help          help for init
  --home string       node's home directory (default "/home/kiran/.gaia")
  -o, --overwrite     overwrite the genesis.json file
  --recover           provide seed phrase to recover existing key instead of creating

Global Flags:
  --log_format string  The logging format (json|plain) (default "plain")
  --log_level string   The logging level (trace|debug|info|warn|error|fatal|panic) (default "info")
  --trace             print out full stack trace on errors

(kiran@kali)~/gaia
$ gaiad init kiran
{"app_message":{"auth":{"accounts":[],"params":{"max_memo_characters":"256","sig_verify_cost_ed25519":"590","sig_verify_cost_secp256k1":"1000","tx_sig_limit":"7","tx_size_cost_per_byte":"10"}}, "authz":{"authorization":[]}},
{"genesis_time":"2023-08-01T14:13:29.123456789Z","initial_height":1,"moniker":"kiran","network":"testnet","version":"0.46.0"}

```

## b. Prepare the genesis file.

```

(kiran@kali)~/gaia
$ cd $HOME

(kiran@kali)~
$ wget https://github.com/cosmos/testnets/raw/master/public/genesis.json.gz
--2023-08-01 14:13:29-- https://github.com/cosmos/testnets/raw/master/public/genesis.json.gz
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://raw.githubusercontent.com/cosmos/testnets/master/public/genesis.json.gz [following]
--2023-08-01 14:13:41-- https://raw.githubusercontent.com/cosmos/testnets/master/public/genesis.json.gz
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.109.133, 185.199.110.133, 185.199.111.133, 185.199.112.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 97344371 (93M) [application/octet-stream]
Saving to: 'genesis.json.gz'

genesis.json.gz      100%[=====>] 92.83M  475KB/s  in 4m 6s

2023-08-01 14:17:55 (387 KB/s) - 'genesis.json.gz' saved [97344371/97344371]

(kiran@kali)~
$ gzip -d genesis.json.gz

(kiran@kali)~
$ mv genesis.json $HOME/.gaia/config/genesis.json

(kiran@kali)~
$ cd $HOME/.gaia/config

(kiran@kali)~/gaia/config
$ sed -i 's/minimum-gas-prices = ""/minimum-gas-prices = "0.0025uatom"/' app.toml

(kiran@kali)~/gaia/config
$ sed -i 's/seeds = ""/seeds = "639d50339d7045436c756a042906b9a69970913f@seed-01.theta-testnet.polypore.xyz:26656,3e506472683ceb7ed75c1578d092c79785c27857@seed-02.theta-testnet.polypore.xyz:26656"/' config.toml~
sed: can't read config.toml~: No such file or directory

(kiran@kali)~/gaia/config
$ sed -i 's/seeds = ""/seeds = "639d50339d7045436c756a042906b9a69970913f@seed-01.theta-testnet.polypore.xyz:26656,3e506472683ceb7ed75c1578d092c79785c27857@seed-02.theta-testnet.polypore.xyz:26656"/' config.toml

(kiran@kali)~/gaia/config
$

```

### 3) State Sync Setup:

```
(kiran@kali)-[~/gaia/config]
$ cd $HOME/gaia/config

(kiran@kali)-[~/gaia/config]
$ sed -i 's/enable = false/enable = true/' config.toml

(kiran@kali)-[~/gaia/config]
$ sed -i 's/trust_height = 0/trust_height = <BLOCK_HEIGHT>/' config.toml

(kiran@kali)-[~/gaia/config]
$ sed -i 's/trust_hash = "<BLOCK_HASH>/' config.toml
sed: -e expression #1, char 30: unterminated `s' command

(kiran@kali)-[~/gaia/config]
$ sed -i 's/trust_hash = ""/trust_hash = "<BLOCK_HASH>/' config.toml

(kiran@kali)-[~/gaia/config]
$ sed -i 's/rpc_servers = ""/rpc_servers = "http://state-sync-01.theta-testnet.polypore.xyz:26
-sync-02.theta-testnet.polypore.xyz:26657/' config.toml

(kiran@kali)-[~/gaia/config]
$
```

### 4) Cosmovisor setup:

```
(kiran@kali)-[~]
$ go install github.com/cosmos/cosmos-sdk/cosmovisor/cmd/cosmovisor@v1.3.0
go: downloading github.com/cosmos/cosmos-sdk/cosmovisor v1.3.0
go: downloading github.com/rs/zerolog v1.28.0
go: downloading github.com/spf13/cobra v1.5.0
go: downloading github.com/cosmos/cosmos-sdk v0.46.0-beta2.0.20220909113810-4882f933
go: downloading github.com/hashicorp/go-getter v1.6.2
go: downloading github.com/otiai10/copy v1.7.0
go: downloading github.com/tendermint/tendermint v0.37.0-alpha.2
go: downloading github.com/cosmos/cosmos-proto v1.0.0-alpha7
go: downloading github.com/cosmos/gogoproto v1.4.1
go: downloading google.golang.org/genproto v0.0.0-20220725144611-272f38e5d71b
go: downloading google.golang.org/grpc v1.49.0

go: downloading github.com/prometheus/client_model v0.2.0
go: downloading github.com/golang/groupcache v0.0.0-20210331224755-41bb18bfe9da
go: downloading golang.org/x/term v0.0.0-20220722155259-a9ba230a4035
go: downloading github.com/bgentry/speakeasy v0.1.0
go: downloading github.com/tendermint/btcd v0.1.1
go: downloading golang.org/x/text v0.3.7

(kiran@kali)-[~]
$ mkdir -p ~/.gaia/cosmovisor/genesis/bin

(kiran@kali)-[~]
$ cp ~/go/bin/gaiad ~/.gaia/cosmovisor/genesis/bin

(kiran@kali)-[~]
$
```

## 5) Create Service File:

```
[Unit]
Description=Cosmovisor service
After=network-online.target

[Service]
User=kiran
ExecStart=/home/kiran/go/bin/cosmovisor run start --x-crisis-skip-assert-invariants --home /home/kiran/.gaia
Restart=no
LimitNOFILE=4096
Environment='DAEMON_NAME=gaiad'
Environment='DAEMON_HOME=/home/kiran/.gaia'
Environment='DAEMON_ALLOW_DOWNLOAD_BINARIES=true'
Environment='DAEMON_RESTART_AFTER_UPGRADE=true'
Environment='DAEMON_LOG_BUFFER_SIZE=512'
Environment='UNSAFE_SKIP_BACKUP=true'

[Install]
WantedBy=multi-user.target
```

## 6) Setting up the keyring:

The keyring holds the private/public keypairs used to interact with a node.

Here, we have used the libsecret library as the backend.

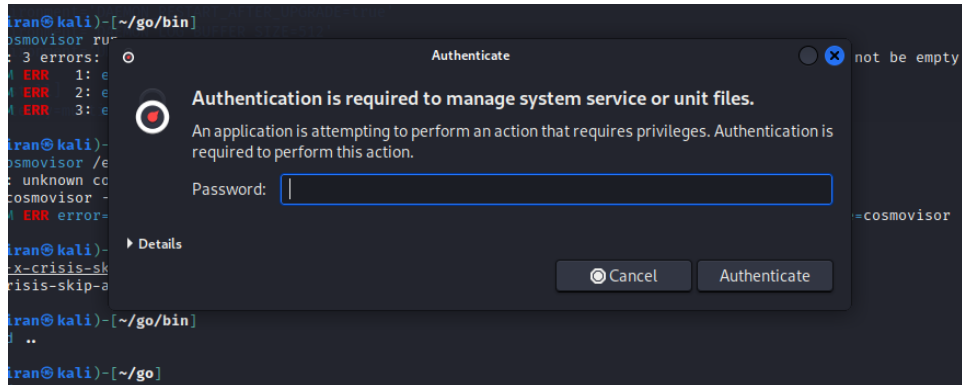
```
(kiran@kali)-[~]
$ git clone https://gitlab.gnome.org/GNOME/libsecret
Cloning into 'libsecret' ...
warning: redirecting to https://gitlab.gnome.org/GNOME/libsecret.git/
remote: Enumerating objects: 4539, done.
remote: Counting objects: 100% (523/523), done.
remote: Compressing objects: 100% (226/226), done.
remote: Total 4539 (delta 323), reused 484 (delta 296), pack-reused 4016
Receiving objects: 100% (4539/4539), 1.38 MiB | 436.00 KiB/s, done.
Resolving deltas: 100% (3308/3308), done.
```

## 7) Start the service:

```
(kiran@kali)-[~]
$ systemctl daemon-reload

(kiran@kali)-[~]
$ systemctl restart systemd-journald

(kiran@kali)-[~]
$
```



```
(kiran@kali)-[~]
$ systemctl enable cosmovisor.service
Created symlink /etc/systemd/system/multi-user.target.wants/cosmovisor.service → /etc/systemd/system/cosmovisor.service.

(kiran@kali)-[~]
$ systemctl start cosmovisor.service

(kiran@kali)-[~]
```

```
(kiran@kali)-[~]
$ journalctl -fu cosmovisor
Aug 01 15:22:20 kali systemd[1]: Started Cosmovisor service.
Aug 01 15:22:20 kali systemd[46266]: cosmovisor.service: Failed to locate executable /root/go/bin/cosmovisor: No such file or directory
Aug 01 15:22:20 kali systemd[46266]: cosmovisor.service: Failed at step EXEC spawning /root/go/bin/cosmovisor: No such file or directory
Aug 01 15:22:20 kali systemd[1]: cosmovisor.service: Main process exited, code=exited, status=203/EXEC
Aug 01 15:22:20 kali systemd[1]: cosmovisor.service: Failed with result 'exit-code'.
```

```
(kiran@kali)-[~/gaia/config]
$ journalctl -fu cosmovisor
Aug 01 19:19:46 kali cosmovisor[80485]: --transport string      Transport protocol:
socket, grpc (default "socket")
Aug 01 19:19:46 kali cosmovisor[80485]: --unsafe-skip-upgrades ints  Skip a set of upgrades
de heights to continue the old binary
Aug 01 19:19:46 kali cosmovisor[80485]: --with-tendermint          Run abci app embedded
ed in-process with tendermint (default true)
Aug 01 19:19:46 kali systemd[1]: cosmovisor.service: Failed with result 'exit-code'.
Aug 01 19:19:46 kali cosmovisor[80485]: --x-crisis-skip-assert-invariants  Skip x/crisis invariants
check on startup
Aug 01 19:19:46 kali cosmovisor[80485]: Global Flags:
Aug 01 19:19:46 kali cosmovisor[80485]: --log_format string    The logging format (json|plain) (default "plain")
Aug 01 19:19:46 kali cosmovisor[80485]: --log_level string    The logging level (trace|debug|info|warn|error|fatal|panic) (default "info")
Aug 01 19:19:46 kali cosmovisor[80479]: Error: exit status 1
Aug 01 19:19:46 kali cosmovisor[80479]: 7:19PM ERR error="exit status 1" module=cosmovisor
```