

AEQUITAS NODE OPERATOR GUIDE

v1.0 · June 2026 · aequitas.digital

Complete step-by-step guide · No prior blockchain experience required · ~20–30 min

Before You Start — What You Need

1. An Aequitas account: You must first be registered as a human on Aequitas. Install the Android app, complete biometric registration, and note your wallet address. Without this, you cannot receive validator rewards.
2. A GitHub account (free): Go to github.com and create a free account. You need this to copy (fork) the Aequitas code so Railway can deploy it.
3. A Railway account (free): Go to railway.app and sign in with GitHub. Railway is a hosting platform that runs your node in the cloud — no server or command line required.
4. Node signing key (RELAYER_PRIVATE_KEY): Your node needs a dedicated Ethereum wallet to sign on-chain registrations. This can be any MetaMask wallet. Export its private key: MetaMask → Account Details → Show Private Key → enter password → copy. Keep strictly private. IMPORTANT: To receive validator rewards you also need NODE_OPERATOR_WALLET set to your registered Aequitas human wallet (the one verified with AequitasBio). Only verified humans can earn validator rewards.
5. 10–30 minutes of your time. Railway does most of the work automatically.

Step 1 — Environment Variables

Security Warning: Your RELAYER_PRIVATE_KEY is like a master password. Anyone who has it controls your node wallet. Never share it publicly, never paste it in chat or email. Use a separate MetaMask wallet for RELAYER_PRIVATE_KEY (signing). NODE_OPERATOR_WALLET (for rewards) must be your registered Aequitas human wallet.

Variable	Required?	What to set
DATABASE_URL	YES	Your PostgreSQL connection string. On Railway: auto-injected when PostgreSQL is in the same project. Format: postgres://user:pass@host:5432/dbname
RELAYER_PRIVATE_KEY	YES	The private key (0x..., 66 chars) of your dedicated node wallet. MetaMask: Account Details → Show Private Key → enter password → copy.
RELAYER_ADDRESS	Recommended	The wallet address (0x..., 42 chars) matching RELAYER_PRIVATE_KEY. Copy from MetaMask. A fallback exists but setting this explicitly prevents startup errors.
NODE_OPERATOR_WALLET	For rewards	Your Aequitas human wallet address — registered via the Android app. Receives your daily validator rewards (40% of all protocol fees). Must be a registered human.
NODE_OPERATOR_BINDING_SIGNATURE	For rewards	Proves you own NODE_OPERATOR_WALLET. Generate it at aequitas.digital/node-binding : sign the shown message with your human wallet in MetaMask, paste the resulting signature here. Without it your node still runs, but cannot auto-register for validator rewards.

Variable	Required?	What to set
PEER_SECRET	Optional/Legacy	Legacy shared-secret fallback. No longer required — nodes authenticate automatically via cryptographic challenge-response (RELAYER_PRIVATE_KEY). Only needed for backward compatibility with older deployments.
SELF_URL	Multi-node	Your node's own public HTTPS URL (e.g. https://my-node.up.railway.app). Required for peer discovery self-exclusion. Find in Railway: Settings → Networking → Public Networking.
PRIMARY_NODE_URL	Multi-node	Set to: https://aequitas.digital — the primary node your node registers with for automatic peer discovery. On startup your node posts its URL + signing address to the primary, gets the full peer list back, and joins the network automatically.
BOOTSTRAP_SNAPSHOT_URL	Recommended	Set to: https://aequitas.digital/api/snapshot — lets a brand-new node start from the network's current state instead of replaying the entire history from genesis. Dramatically faster first sync.
BOOTSTRAP_SIGNER	With snapshot	The primary node's signing address, used to verify the snapshot is genuine before importing it. Get the current value from https://aequitas.digital/api/status → "signing_address". Required whenever BOOTSTRAP_SNAPSHOT_URL is set.
SNAPSHOT_TOKEN	Optional	Not required to bootstrap a new node — without it you still get everything needed to run correctly (accounts, balances, pool, config). Only unlocks the full export (nullifier/wallet linkage + bio_registrations), used for authoritative resync of an already-diverged node. Ask the network operator only if you actually need that.
RESYNC_FROM_SNAPSHOT	Recovery only	DANGEROUS, temporary: set to true together with BOOTSTRAP_SNAPSHOT_URL and BOOTSTRAP_SIGNER only to recover a node whose state has diverged from the network. Replaces local state outright. Restart once, then remove this variable again — leaving it set forces a full resync on every restart.
AUTO_HEAL_ON_DIVERGENCE	Strongly recommended	Important for network security and speed: if your node's chain ever diverges from the network (e.g. after downtime or a bad restart), it detects this itself by comparing against PRIMARY_NODE_URL every few minutes and resyncs automatically — no manual RESYNC_FROM_SNAPSHOT needed. A node that stays diverged and keeps broadcasting its own blocks can slow down or destabilize the network for every other operator. Set to true together with PRIMARY_NODE_URL, BOOTSTRAP_SNAPSHOT_URL and BOOTSTRAP_SIGNER.
PORT	No	Leave unset on Railway — Railway sets this automatically. Default is 8080.
NODE_KEY	No	Base64 libp2p key for stable P2P identity. Auto-generated if omitted, but changes on every restart. If not set, the node prints it to stderr: "SAVE THIS AS NODE_KEY ENVIRONMENT VAR: ". Copy and paste it here.
IS_PRIMARY_NODE	No	Leave unset or false. Distribution now uses a DB-level lock — any node can run it without this variable. Setting true on a secondary node is no longer necessary.
RESET_STATE	No	DANGEROUS: Setting this to true wipes your entire database on every restart. Development use only. Never in production.

Step 2 — Deploy on Railway (Recommended)

Railway is the easiest way to run your node — no server setup, no command line required. The free tier covers all requirements. Total time: about 10–15 minutes.

- 1 Fork github.com/hanoi96international-gif/Aequitas to your own GitHub account (click Fork → Create fork)
- 2 On railway.app, sign in with GitHub, then New Project → + New → Database → Add PostgreSQL
- 3 In that same Railway project, click + New → GitHub Repo and select your Aequitas fork — Railway detects the Dockerfile automatically
- 4 Click Deploy Now — a first build starts (may fail without env vars, that is normal)
- 5 Click your Aequitas service → Variables → add each variable (see table above). Minimum required: RELAYER_PRIVATE_KEY, RELAYER_ADDRESS, NODE_OPERATOR_WALLET, SELF_URL, PRIMARY_NODE_URL=<https://aequitas.digital> (PEER_SECRET is no longer required)
- 6 Click Deploy (or save variables to trigger auto-redeploy). Build takes ~3 minutes while Go compiles the node binary.
- 7 Watch Deploy Logs. Success looks like: [Aequitas Node Running](#) and [\[NODE\] Registered node operator wallet: 0x...](#)
- 8 Go to Settings → Networking → Generate Domain to get your public URL
- 9 Open <https://YOUR-URL/api/status> — you should see JSON with height climbing every ~6 seconds

```
# Railway auto-sets DATABASE_URL if PostgreSQL is in the same project
RELAYER_PRIVATE_KEY = 0xYOUR_PRIVATE_KEY
RELAYER_ADDRESS     = 0xYOUR_NODE_WALLET_ADDRESS
NODE_OPERATOR_WALLET = 0xYOUR_HUMAN_WALLET
# PEER_SECRET is no longer required — authentication is automatic
SELF_URL            = https://YOUR-RAILWAY-DOMAIN.up.railway.app
PRIMARY_NODE_URL    = https://aequitas.digital
AUTO_HEAL_ON_DIVERGENCE = true
```

Step 2b — Alternative: Deploy with Docker (Advanced)

Use this if you have your own server (VPS, home server, cloud VM). Requires Docker and a PostgreSQL database.

```
# 1. Download the code
git clone https://github.com/hanoi96international-gif/Aequitas && cd Aequitas

# 2. Build the node image (~3 min for Go compilation)
docker build -t aequitas-node .

# 3. Start the node
docker run -d --name aequitas-node --restart unless-stopped \
-e DATABASE_URL="postgres://user:pass@host:5432/aequitas" \
-e RELAYER_PRIVATE_KEY="0xYOUR_PRIVATE_KEY" \
-e RELAYER_ADDRESS="0xYOUR_NODE_WALLET_ADDRESS" \
-e NODE_OPERATOR_WALLET="0xYOUR_HUMAN_WALLET" \
# -e PEER_SECRET="..." (optional/legacy, not required) \
-e SELF_URL="https://YOUR-PUBLIC-URL" \
-e PRIMARY_NODE_URL="https://aequitas.digital" \
-e AUTO_HEAL_ON_DIVERGENCE="true" \
-p 8080:8080 aequitas-node

# 4. Watch the live logs
docker logs -f aequitas-node
```

Step 3 — Verify Your Node is Running

Open these URLs in your browser. Replace YOUR-NODE-URL with your actual Railway domain or server address.

```
https://YOUR-NODE-URL/api/status
→ Expected: {"height": 1234, "total_humans": N, "aequitas_index": N}

https://YOUR-NODE-URL/rpc
→ Expected: {"jsonrpc": "2.0", "error": "method not specified"} — RPC is alive
```

The block height should match the primary node within 1–2 blocks within seconds of startup. If it stays at 0, check that PRIMARY_NODE_URL=https://aequitas.digital is set and reachable.

Step 3b — Register Your Validator Key (Decentralized Auth)

Instead of a shared PEER_SECRET, register your node signing key with your human wallet. This cryptographically proves you control both keys. Get the signing key signature by running this on your server (SSH/Railway shell):

```
curl "http://localhost:8080/api/sign-validator-challenge?wallet=0xYOUR_HUMAN_WALLET"
```

Then use the website Network → Run a Node tab and click "Sign with MetaMask & Register" to complete the registration.

Step 4 — Connect MetaMask to Your Node (Optional)

In MetaMask: click the network dropdown → Add network → Add a network manually, then enter:

Network Name	Aequitas Chain
RPC URL	https://YOUR-NODE-URL/rpc
Chain ID	1926
Currency Symbol	AEQ
Decimals	18
Block Explorer	https://aequitas.digital

Step 5 — Earning Validator Rewards

The Validators Pool collects 40% of all protocol fees (swap fees, demurrage, wealth cap overflow). Every day at 20:00 Berlin time (CEST/CET, handles DST automatically) the node distributes the pool balance to all registered node operators proportionally by blocks produced. The more consistently your node runs, the larger your share.

- 1 Make sure you are registered as a human on Aequitas. If not: install the Android app and complete biometric registration first. You will receive a wallet address and 1,000 AEQ.
- 2 Set NODE_OPERATOR_WALLET = your Aequitas human wallet address in your Railway Variables
- 3 Save — Railway redeploys automatically. On Docker: docker restart aequitas-node
- 4 In your node logs, confirm: [NODE] Registered node operator wallet: 0x...
- 5 Rewards are distributed automatically every day at 20:00 Berlin time (CEST/CET). Just keep your node running — no further action needed.

Troubleshooting

Symptom	Likely Cause	Solution
Block height stays at 0	PRIMARY_NODE_URL not set or wrong	Set PRIMARY_NODE_URL=https://aequitas.digital and redeploy. Also set SELF_URL to your node's public URL.
DATABASE_URL error on startup	Wrong connection string	Check format: postgres://user:pass@host:5432/dbname — make sure PostgreSQL is running and accessible.
"no code at address" in logs	V7 contract not yet deployed	Normal on first start — node auto-deploys V7. Wait a few seconds and check again.
"NODE_OPERATOR_WALLET not set"	Missing environment variable	Add NODE_OPERATOR_WALLET=0xYOUR_HUMAN_WALLET. Node runs fine without it but you won't receive rewards.
Railway "Application error"	Build or startup failure	Check Deploy Logs. Most common: DATABASE_URL missing or RELAYER_PRIVATE_KEY in wrong format (must start with 0x).
Port 8080 not reachable (Docker)	Firewall or cloud provider config	Open TCP port 8080 inbound in your firewall or cloud security group settings.
Docker build fails (module error)	No internet during build	Docker build needs outbound internet to download Go modules. Railway handles this automatically.