Operate the node

Monitoring

Endpoint (internal interface)	Use it for
`/health`	Health checks by Docker or Kubernetes. Returns `200 OK` when the node is healthy.
`/status/diagnostics`	Details about the node's state, for administrators
`/metrics`	Metrics in Prometheus format

See monitoring in the Nuts node documentation.

Backups

Back up both, and test that you can restore them:

- the SQL database (`storage.sql.connection`);
- the private keys (`crypto.storage`, e.g. HashiCorp Vault or Azure Key Vault).

Without the private keys, the node can't use its DIDs anymore, and every care organization needs a new subject, new credentials and a new registration.

Upgrades

- Read the release notes before you upgrade.
- Upgrade test and acceptance first, then production.
- Keep a fixed version in your deployment (see "Deploy the Nuts node").

Troubleshooting

Problem	Likely cause	What to do
The node doesn't start	A Discovery Service definition can't be read	The log says which file. Use the files from the use case's IG, unchanged.
A DID doesn't resolve from the internet	Reverse proxy, TLS or DNS	See "Endpoints and TLS certificates" and check 2 in "Test your setup step by step"

Problem	Likely cause	What to do
Discovery registration returns `412`	Missing or expired credentials, or the Discovery Server can't be reached	The response says which. Fix it and call the registration `POST` again: the node doesn't retry by itself.
Registration status is `error`	A refresh failed	`error` in the status says why. Fix it; the node retries at the next refresh.
You can't find a care organization that just registered	Your node updates its copy of the Discovery Service every 10 minutes by default	Wait, or check `discovery.client.refreshinterval`
Token request returns `412`	Your wallet doesn't have the credentials the other party's policy requires	Check the wallet and the use case's policy definition
Token request returns `503`	The other party's authorization server can't be reached	Check the `authServerURL` and your outbound network
The other party returns `401` for a token that hasn't expired	Their authorization server may have lost the token, e.g. after a restart	Request a new token with the header `Cache-Control: no-cache`, which skips your node's token cache. Don't do this by default: it adds load on both sides.
DPoP validation fails for incoming requests	The URL your API sees differs from the URL the requester used, e.g. after a reverse proxy rewrote it	Validate with the URL as the requester called it (see "Validate the access token")

Still stuck? Set `verbosity` to `debug` for more detail in the log, and ask on the Nuts Slack.

Revision #2
Created 2026-10-27 15:13:58 UTC by Rein Krul
Updated 2026-10-27 15:28:20 UTC by Rein Krul