



≡ Configuration and operations > **Troubleshooting**

Configuration and operations

Troubleshooting

This page is the deep runbook. Start at </help/troubleshooting> if you want the fast triage flow first.

Command ladder

Run these first, in this order:

```
openclaw status
openclaw gateway status
openclaw logs --follow
openclaw doctor
openclaw channels status --probe
```

Expected healthy signals:

`openclaw gateway status` shows `Runtime: running` and `RPC probe: ok` .

`openclaw doctor` reports no blocking config/service issues.

`openclaw channels status --probe` shows connected/ready channels.

No replies

If channels are up but nothing answers, check routing and policy before reconnecting anything.



```
openclaw status
openclaw channels status --probe
openclaw pairing list <channel>
openclaw config get channels >
openclaw logs --follow
```

Look for:

Pairing pending for DM senders.

Group mention gating (`requireMention` , `mentionPatterns`).

Channel/group allowlist mismatches.

Common signatures:

`drop guild message (mention required` → group message ignored until mention.

`pairing request` → sender needs approval.

`blocked` / `allowlist` → sender/channel was filtered by policy.

Related:

Dashboard control ui connectivity

When dashboard/control UI will not connect, validate URL, auth mode, and secure context assumptions.

```
openclaw gateway status
openclaw status
openclaw logs --follow
openclaw doctor
openclaw gateway status --json
```

Look for:

Correct probe URL and dashboard URL.

Auth mode/token mismatch between client and gateway.

HTTP usage where device identity is required.

Common signatures:

device identity required → non-secure context or missing device auth.

unauthorized / reconnect loop → token/password mismatch.

gateway connect failed: → wrong host/port/url target.

Related:

Gateway service not running

Use this when service is installed but process does not stay up.

```
openclaw gateway status
openclaw status
openclaw logs --follow
openclaw doctor
openclaw gateway status --deep
```

Look for:



Runtime: stopped with exit hints.

Service config mismatch (`Config (cli) vs Config (service)).

Port/listener conflicts.

Common signatures:

Gateway start blocked: `set gateway.mode=local` → local gateway mode is not enabled. Fix: `set gateway.mode="local"` in your config (or run `openclaw configure`). If you are running OpenClaw via Podman using the dedicated `openclaw` user, the config lives at `~openclaw/.openclaw/openclaw.json` .

refusing to bind gateway ... without auth → non-loopback bind without token/password.

another gateway instance is already listening / `EADDRINUSE` → port conflict.

Related:

[/gateway/background-process](#)

[/gateway/configuration](#)

[/gateway/doctor](#)

Channel connected messages not flowing

If channel state is connected but message flow is dead, focus on policy, permissions, and channel specific delivery rules.

```
openclaw channels status --probe
openclaw pairing list <channel>
openclaw status --deep
openclaw logs --follow
openclaw config get channels
```

Look for:



DM policy (pairing , allowlist , open , disabled).

Group allowlist and mention requirements.

Missing channel API permissions/scopes.

Common signatures:

mention required → message ignored by group mention policy.

pairing / pending approval traces → sender is not approved.

missing_scope , not_in_channel , Forbidden , 401/403 → channel auth/permissions issue.

Related:

[/channels/troubleshooting](#)

[/channels/whatsapp](#)

[/channels/telegram](#)

[/channels/discord](#)

Cron and heartbeat delivery

If cron or heartbeat did not run or did not deliver, verify scheduler state first, then delivery target.

```
openclaw cron status
openclaw cron list
openclaw cron runs --id <jobId> --limit 20
openclaw system heartbeat last
openclaw logs --follow
```

Look for:

Cron enabled and next wake present.



Job run history status (ok , skipped , error).

Heartbeat skip reasons (quiet-hours , requests-in-flight , alerts-disabled).

>

Common signatures:

cron: scheduler disabled; jobs will not run automatically → cron disabled.

cron: timer tick failed → scheduler tick failed; check file/log/runtime errors.

heartbeat skipped with reason=quiet-hours → outside active hours window.

heartbeat: unknown accountId → invalid account id for heartbeat delivery target.

Related:

[/automation/troubleshooting](#)

[/automation/cron-jobs](#)

[/gateway/heartbeat](#)

Node paired tool fails

If a node is paired but tools fail, isolate foreground, permission, and approval state.

```
openclaw nodes status
openclaw nodes describe --node <idOrNameOrIp>
openclaw approvals get --node <idOrNameOrIp>
openclaw logs --follow
openclaw status
```

Look for:

Node online with expected capabilities.



OS permission grants for camera/mic/location/screen.

Exec approvals and allowlist state.

Common signatures:

>

`NODE_BACKGROUND_UNAVAILABLE` → node app must be in foreground.

`*_PERMISSION_REQUIRED` / `LOCATION_PERMISSION_REQUIRED` → missing OS permission.

`SYSTEM_RUN_DENIED`: approval required → exec approval pending.

`SYSTEM_RUN_DENIED`: allowlist miss → command blocked by allowlist.

Related:

[/nodes/troubleshooting](#)

[/nodes/index](#)

[/tools/exec-approvals](#)

Browser tool fails

Use this when browser tool actions fail even though the gateway itself is healthy.

```
openclaw browser status
openclaw browser start --browser-profile openclaw
openclaw browser profiles
openclaw logs --follow
openclaw doctor
```

Look for:

Valid browser executable path.

CDP profile reachability.

Extension relay tab attachment for `profile="chrome"` .

Common signatures:



Failed to start Chrome CDP on port → browser process failed to launch.

`browser.executablePath` not found → configured path is invalid.

Chrome extension relay is running, but no tab is connected → extension relay not attached.

Browser `attachOnly` is enabled ... not reachable → attach-only profile has no reachable target.

Related:

[/tools/browser-linux-troubleshooting](#)

[/tools/chrome-extension](#)

[/tools/browser](#)

If you upgraded and something suddenly broke

Most post-upgrade breakage is config drift or stricter defaults now being enforced.

1) Auth and URL override behavior changed

```
openclaw gateway status
openclaw config get gateway.mode
openclaw config get gateway.remote.url
openclaw config get gateway.auth.mode
```

What to check:

If `gateway.mode=remote` , CLI calls may be targeting remote while your local service is fine.

Explicit `--url` calls do not fall back to stored credentials.

Common signatures:



gateway connect failed: → wrong URL target.

unauthorized → endpoint reachable but wrong auth.

2) Bind and auth guardrails are stricter

```
openclaw config get gateway.bind
openclaw config get gateway.auth.token
openclaw gateway status
openclaw logs --follow
```

What to check:

Non-loopback binds (lan , tailnet , custom) need auth configured.

Old keys like gateway.token do not replace gateway.auth.token .

Common signatures:

refusing to bind gateway ... without auth → bind+auth mismatch.

RPC probe: failed while runtime is running → gateway alive but inaccessible with current auth/url.

3) Pairing and device identity state changed

```
openclaw devices list
openclaw pairing list <channel>
openclaw logs --follow
openclaw doctor
```

What to check:

Pending device approvals for dashboard/nodes.



Pending DM pairing approvals after policy or identity changes.

Common signatures:

device identity required → device auth not satisfied.

pairing required → sender/device must be approved.

If the service config and runtime still disagree after checks, reinstall service metadata from the same profile/state directory:

```
openclaw gateway install --force
openclaw gateway restart
```

Related:

[Multiple Gateways](#)

[Security](#)

Powered by [mintlify](#)