



≡ Gateway > Gateway Runbook

Gateway

Gateway Runbook

Use this page for day-1 startup and day-2 operations of the Gateway service.

Deep troubleshooting

Symptom-first diagnostics with exact command ladders and log signatures.

Configuration

Task-oriented setup guide + full configuration reference.

5-minute local startup

1 Start the Gateway

```
openclaw gateway --port 18789
# debug/trace mirrored to stdio
openclaw gateway --port 18789 --verbose
# force-kill listener on selected port, then start
openclaw gateway --force
```

2 Verify service health



```
openclaw gateway status
openclaw status
openclaw logs --follow
>
```

Healthy baseline: Runtime: running and RPC probe: ok .

3 Validate channel readiness

```
openclaw channels status --probe
```

❗ Gateway config reload watches the active config file path (resolved from profile/state defaults, or `OPENCLAW_CONFIG_PATH` when set). Default mode is `gateway.reload.mode="hybrid"` .

Runtime model

One always-on process for routing, control plane, and channel connections.

Single multiplexed port for:

WebSocket control/RPC

HTTP APIs (OpenAI-compatible, Responses, tools invoke)

Control UI and hooks

Default bind mode: `loopback` .

Auth is required by default (`gateway.auth.token` / `gateway.auth.password` , or `OPENCLAW_GATEWAY_TOKEN` / `OPENCLAW_GATEWAY_PASSWORD`) .

Port and bind precedence



Resolution order

Gateway port	--port → OPENCLAW_GATEWAY_PORT → gateway.port → 18789
Bind mode	> CLI/override → gateway.bind → loopback

Hot reload modes

gateway.reload.mode	Behavior
off	No config reload
hot	Apply only hot-safe changes
restart	Restart on reload-required changes
hybrid (default)	Hot-apply when safe, restart when required

Operator command set

```

openclaw gateway status
openclaw gateway status --deep
openclaw gateway status --json
openclaw gateway install
openclaw gateway restart
openclaw gateway stop
openclaw logs --follow
openclaw doctor

```

Remote access

Preferred: Tailscale/VPN. Fallback: SSH tunnel.

```
ssh -N -L 18789:127.0.0.1:18789 user@host
```

Then connect clients to `ws://127.0.0.1:18789` locally.



⚠ If gateway auth is configured, clients still must send auth (token / password) even over SSH tunnels.

See: [Remote Gateway](#), [Authentication](#), [Tailscale](#).

Supervision and service lifecycle

Use supervised runs for production-like reliability.

macOS (launchd) Linux (systemd user) Linux (system service)

```
openclaw gateway install
openclaw gateway status
openclaw gateway restart
openclaw gateway stop
```

LaunchAgent labels are `ai.openclaw.gateway` (default) or `ai.openclaw.<profile>` (named profile). `openclaw doctor` audits and repairs service config drift.

Multiple gateways on one host

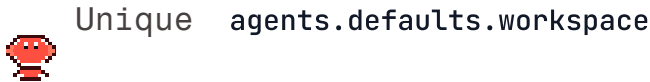
Most setups should run **one** Gateway. Use multiple only for strict isolation/redundancy (for example a rescue profile).

Checklist per instance:

Unique `gateway.port`

Unique `OPENCLAW_CONFIG_PATH`

Unique `OPENCLAW_STATE_DIR`



Unique `agents.defaults.workspace`

Example:

```
>
OPENCLAW_CONFIG_PATH=~/.openclaw/a.json OPENCLAW_STATE_DIR=~/.openclaw
OPENCLAW_CONFIG_PATH=~/.openclaw/b.json OPENCLAW_STATE_DIR=~/.openclaw-b openclaw
```

See:

.

Dev profile quick path

```
openclaw --dev setup
openclaw --dev gateway --allow-unconfigured
openclaw --dev status
```

Defaults include isolated state/config and base gateway port `19001` .

Protocol quick reference (operator view)

First client frame must be `connect` .

Gateway returns `hello-ok` snapshot (`presence` , `health` , `stateVersion` , `uptimeMs` , `limits/policy`).

Requests: `req(method, params)` → `res(ok/payload/error)` .

Common events: `connect.challenge` , `agent` , `chat` , `presence` , `tick` , `health` , `heartbeat` , `shutdown` .

Agent runs are two-stage:

1. Immediate accepted ack (`status:"accepted"`)
2. Final completion response (`status:"ok"|"error"`), with streamed `agent` events in between.

See full protocol docs: [Gateway Protocol](#).



Operational checks

Liveness

Open WS and send `connect` .

Expect `hello-ok` response with snapshot.

Readiness

```
openclaw gateway status
openclaw channels status --probe
openclaw health
```

Gap recovery

Events are not replayed. On sequence gaps, refresh state (`health` , `system-presence`) before continuing.

Common failure signatures

Signature	Likely issue
<code>refusing to bind gateway ... without auth</code>	Non-loopback bind without token/password
<code>another gateway instance is already listening</code> / <code>EADDRINUSE</code>	Port conflict
<code>Gateway start blocked: set gateway.mode=local</code>	Config set to remote mode
<code>unauthorized</code> during connect	Auth mismatch between client and gateway

For full diagnosis ladders, use [Gateway Troubleshooting](#).



Safety guarantees

Gateway protocol clients fail fast when Gateway is unavailable (no implicit direct-channel fallback).

Invalid/non-connect first frames are rejected and closed.

Graceful shutdown emits `shutdown` event before socket close.

Related:

[Troubleshooting](#)

[Background Process](#)

[Configuration](#)

[Health](#)

[Doctor](#)

[Authentication](#)

[Configuration](#) >

Powered by [mintlify](#)