



≡ First steps > Onboarding Wizard (CLI)

First steps

Onboarding Wizard (CLI)

The onboarding wizard is the **recommended** way to set up OpenClaw on macOS, Linux, or Windows (via WSL2; strongly recommended). It configures a local Gateway or a remote Gateway connection, plus channels, skills, and workspace defaults in one guided flow.

```
openclaw onboard
```

❗ Fastest first chat: open the Control UI (no channel setup needed). Run `openclaw dashboard` and chat in the browser. Docs: [https://docs.openclaw.ai/start/wizard](#).

To reconfigure later:

```
openclaw configure
openclaw agents add <name>
```

❗ `--json` does not imply non-interactive mode. For scripts, use `--non-interactive`.

💡 Recommended: set up a Brave Search API key so the agent can use `web_search` (`web_fetch` works without a key). Easiest path: `openclaw configure --section web` which stores `tools.web.search.apiKey`. Docs: [https://docs.openclaw.ai/start/wizard](#).

QuickStart vs Advanced



The wizard starts with **QuickStart** (defaults) vs **Advanced** (full control).

QuickStart (defaults) **Advanced (full control)**

Local gateway (loopback)

Workspace default (or existing workspace)

Gateway port **18789**

Gateway auth **Token** (auto-generated, even on loopback)

Tailscale exposure **Off**

Telegram + WhatsApp DMs default to **allowlist** (you'll be prompted for your phone number)

What the wizard configures

Local mode (default) walks you through these steps:

1. **Model/Auth** — Anthropic API key (recommended), OpenAI, or Custom Provider (OpenAI-compatible, Anthropic-compatible, or Unknown auto-detect). Pick a default model.
2. **Workspace** — Location for agent files (default `~/.openclaw/workspace`). Seeds bootstrap files.
3. **Gateway** — Port, bind address, auth mode, Tailscale exposure.
4. **Channels** — WhatsApp, Telegram, Discord, Google Chat, Mattermost, Signal, BlueBubbles, or iMessage.
5. **Daemon** — Installs a LaunchAgent (macOS) or systemd user unit (Linux/WSL2).
6. **Health check** — Starts the Gateway and verifies it's running.
7. **Skills** — Installs recommended skills and optional dependencies.



Re-running the wizard does **not** wipe anything unless you explicitly choose **Reset** (or pass `--reset`). If the config is invalid or contains legacy keys, the wizard asks you to run `openclaw doctor` first.

Remote mode only configures the local client to connect to a Gateway elsewhere. It does **not** install or change anything on the remote host.

Add another agent

Use `openclaw agents add <name>` to create a separate agent with its own workspace, sessions, and auth profiles. Running without `--workspace` launches the wizard.

What it sets:

```
agents.list[].name
```

```
agents.list[].workspace
```

```
agents.list[].agentDir
```

Notes:

Default workspaces follow `~/.openclaw/workspace-<agentId>` .

Add `bindings` to route inbound messages (the wizard can do this).

Non-interactive flags: `--model` , `--agent-dir` , `--bind` , `--non-interactive` .

Full reference

For detailed step-by-step breakdowns, non-interactive scripting, Signal setup, RPC API, and a full list of config fields the wizard writes, see the [Wizard Reference](#).

Related docs



CLI command reference: [openclaw onboard](#)

Onboarding overview: [Onboarding Overview](#)

macOS app onboarding: [Onboarding](#)

Agent first-run ritual: [Agent Bootstrapping](#)

[< Onboarding Overview](#)

[Onboarding: macOS App >](#)

Powered by [mintlify](#)