



☰ CLI commands > **onboard**

CLI commands

onboard

Interactive onboarding wizard (local or remote Gateway setup).

Related guides

CLI onboarding hub: [Onboarding Wizard \(CLI\)](#)

Onboarding overview: [Onboarding Overview](#)

CLI onboarding reference: [CLI Onboarding Reference](#)


CLI automation: [CLI Automation](#)

macOS onboarding: [Onboarding \(macOS App\)](#)

Examples

```
openclaw onboard
openclaw onboard --flow quickstart
openclaw onboard --flow manual
openclaw onboard --mode remote --remote-url ws://gateway-host:18789
```

Non-interactive custom provider:



```

openclaw onboard --non-interactive \
  --auth-choice custom-api-key \
  --custom-base-url "https://llm.example.com/v1" \
  --custom-model-id "foo-large" \
  --custom-api-key "$CUSTOM_API_KEY" \
  --custom-compatibility openai

```

`--custom-api-key` is optional in non-interactive mode. If omitted, onboarding checks `CUSTOM_API_KEY`.

Non-interactive Z.AI endpoint choices:

Note: `--auth-choice zai-api-key` now auto-detects the best Z.AI endpoint for your key (prefers the general API with `zai/glm-5`). If you specifically want the GLM Coding Plan endpoints, pick `zai-coding-global` or `zai-coding-cn`.

```

# Promptless endpoint selection
openclaw onboard --non-interactive \
  --auth-choice zai-coding-global \
  --zai-api-key "$ZAI_API_KEY"

# Other Z.AI endpoint choices:
# --auth-choice zai-coding-cn
# --auth-choice zai-global
# --auth-choice zai-cn

```

Flow notes:

`quickstart` : minimal prompts, auto-generates a gateway token.

`manual` : full prompts for port/bind/auth (alias of `advanced`).

Fastest first chat: `openclaw dashboard` (Control UI, no channel setup).

Custom Provider: connect any OpenAI or Anthropic compatible endpoint, including hosted providers not listed. Use Unknown to



auto-detect.

Common follow-up commands

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

❗ `--json` does not imply non-interactive mode. Use `--non-interactive` for scripts.

< nodes

pairing >

Powered by [mintlify](#)