


```
git clone https://github.com/openclaw/openclaw
cd openclaw
export OPENCLAW_IMAGE="ghcr.io/openclaw/openclaw:latest"
./scripts/docker/setup.sh
```

Debian 13

Run as root

```
# Install Node.js
# Run as root
curl -fsSL https://deb.nodesource.com/setup_24.x | bash -
apt install nodejs
```

Run as non-root

```
# Switch to non-root account
# NOTE: Node.js 22.14 or newer is required for openclaw
su - <your-username>
node -v
npm --version

mkdir -p "$HOME/.npm-global"
npm config set prefix "$HOME/.npm-global"
npm prefix -g
export PATH="$HOME/.npm-global/bin:$PATH"
```

Edit: ~/.bashrc

```
export PATH="$HOME/.npm-global/bin:$PATH"
```

Install openclaw

```
npm install -g openclaw@latest
openclaw onboard --install-daemon
```

Post-Install

?? AI ??

?? 3 ??????????

- SOUL.md — ??????????????????
- AGENTS.md — ??????????????????
- IDENTITY.md — ??????

Timezone

```
openclaw config set agents.defaults.userTimezone "Asia/Taipei"
```

CLI Commands

Check Service

```
docker compose run --rm openclaw-cli status
docker compose run --rm openclaw-cli gateway status
```

Pairing required

Approve the device

```
docker compose run --rm openclaw-cli devices list
docker compose run --rm openclaw-cli devices approve <request-id>
```

Model Configuration

```
# 模型管理
openclaw models list
openclaw models status
openclaw models set <provider/model>
openclaw models set-image <provider/model>

# 模型回退
openclaw models fallbacks list
openclaw models fallbacks add <provider/model>
openclaw models fallbacks remove <provider/model>
openclaw models fallbacks clear
```

Telegram

Configuration

Create a Bot from Telegram

1. Open Telegram
2. Chat with @BotFather
3. Run /newbot , follow prompts to create your bot, and save the token.
 - Bot Name: <whatever>
 - Bot Username: <unique-name and must end in 'bot'>

Go to OpenClaw CLI

```
docker compose run --rm openclaw-cli channels add --channel telegram --token "<bot-token>"
```

Go to Telegram

1. Open you-bot
2. Send "Hi"
3. Getting a pairing code "XXXXX"

Go to OpenClaw CLI

```
docker compose run --rm openclaw-cli pairing list telegram  
docker compose run --rm openclaw-cli pairing approve telegram XXXXX
```

Done

Bot Commands

- /btw : ??
- /subagent : ??
- /new : ?? Token ???

Gateway

Local only (default)

- ?????????? SSH Port Forwarding

```
~/openclaw/openclaw.json :
```

```
"gateway": {
  "port": 18789,
  "mode": "local",
  "bind": "loopback",
  "controlUi": {
    "allowedOrigins": [
      "http://localhost:18789",
      "http://127.0.0.1:18789"
    ]
  },
  "auth": {
    "mode": "token",
    "token": "YOUR-OPENCLAW-TOKEN"
  },
}
```

Tailscale

openclaw.json ? tailscale ?????? Docker ???

1. Tailscale Console: Enable **HTTPS Certificates**
2. OpenClaw Host: Run
 1. `sudo tailscale serve --bg --https=443 127.0.0.1:18789`
 2. `tailscale serve status`
3. Change `~/.openclaw/openclaw.json` as follows
4. Connect to tailnet, and then visit `https://your-openclaw-device.tailnet-domain/?token=YOUR-TOKEN`, such as `https://dockers-vm.tailcb58c9.ts.net/?token=YOUR-TOKEN`

`~/.openclaw/openclaw.json` :

- mode: local
- bind: loopback
- trustedProxies: ["127.0.0.1", "::1"]
- allowedOrigins: `https://yourdevice.tailnet-domain`

```
"gateway": {
  "port": 18789,
  "mode": "local",
  "bind": "loopback",
  "trustedProxies": ["127.0.0.1", "::1"],
  "controlUi": {
```

```

    "allowedOrigins": [
      "http://localhost:18789",
      "http://127.0.0.1:18789",
      "https://yourdevice.tailnet-domain"
    ]
  },
  "auth": {
    "mode": "token",
    "token": "YOUR-OPENCLAW-TOKEN"
  },
},

```

openclaw.json ? tailscale ?????????? Docker ???

??	????	HTTPS	??	????
serve	? tailnet	??	Tailscale identity headers/Token/Password	????
funnel	????	??	Password	Webhook
bind: "tailnet"	? tailnet	?	Token/Password	???
off	? localhost	?	Token	SSH Port Forward

Skill

3rd Party

- [Architecture Diagram Generator](#): ?????????? HTML ???
- [Khazix Skills](#): ????????????

FAQ

Context limit exceeded

“ Context limit exceeded. I've reset our conversation to start fresh - please try again.

To prevent this, increase your compaction buffer by setting `agents.defaults.compaction.reserveTokensFloor` to 20000 or higher in your config.

Cause: ??????? Context Window ? ???????????

Solution: ? Gemma4-31B-it ??

Edit: ~/.openclaw/openclaw.json

- `contextWindow` : 128000
- `reserveTokensFloor` : 40000

```
"models": {
  "mode": "merge",
  "providers": {
    "custom-cpamc": {
      "baseUrl": "http://192.168.31.89:8317/v1",
      "api": "openai-completions",
      "apiKey": "sk-cvgaT1Z3EhJRBB5pu",
      "models": [
        {
          "id": "gemma-4-31b-it",
          "name": "gemma-4-31b-it (Custom Provider)",
          "contextWindow": 128000,
          "maxTokens": 4096,
          "input": [
            "text"
          ],
        }
      ],
    }
  }
}
```

```
"agents": {
  "defaults": {
    "workspace": "/home/along/.openclaw/workspace",
    "model": {
      "primary": "custom-cpamc/gemma-4-31b-it"
    },
    "models": {
      "custom-cpamc/gemma-4-31b-it": {}
    },
    "compaction": {
      "reserveTokensFloor": 40000
    }
  }
},
```

???? CLI

```
openclaw config set agents.defaults.compaction.reserveTokensFloor 40000
```

LLM idle timeout

“ The model did not produce a response before the LLM idle timeout. Please try again, or increase `agents.defaults.llm.idleTimeoutSeconds` in your config (set to 0 to disable).

Solution:

```
openclaw config set agents.defaults.timeoutSeconds 600  
openclaw config set agents.defaults.llm.idleTimeoutSeconds 600
```

gateway connect failed

“ gateway connect failed: GatewayClientRequestError: pairing required

???? pending devices

```
openclaw devices list  
openclaw devices list --json
```

Revision #49

Created 2026-03-24 12:00:24 CST by A-Lang (Admin)

Updated 2026-04-23 09:58:47 CST by A-Lang (Admin)