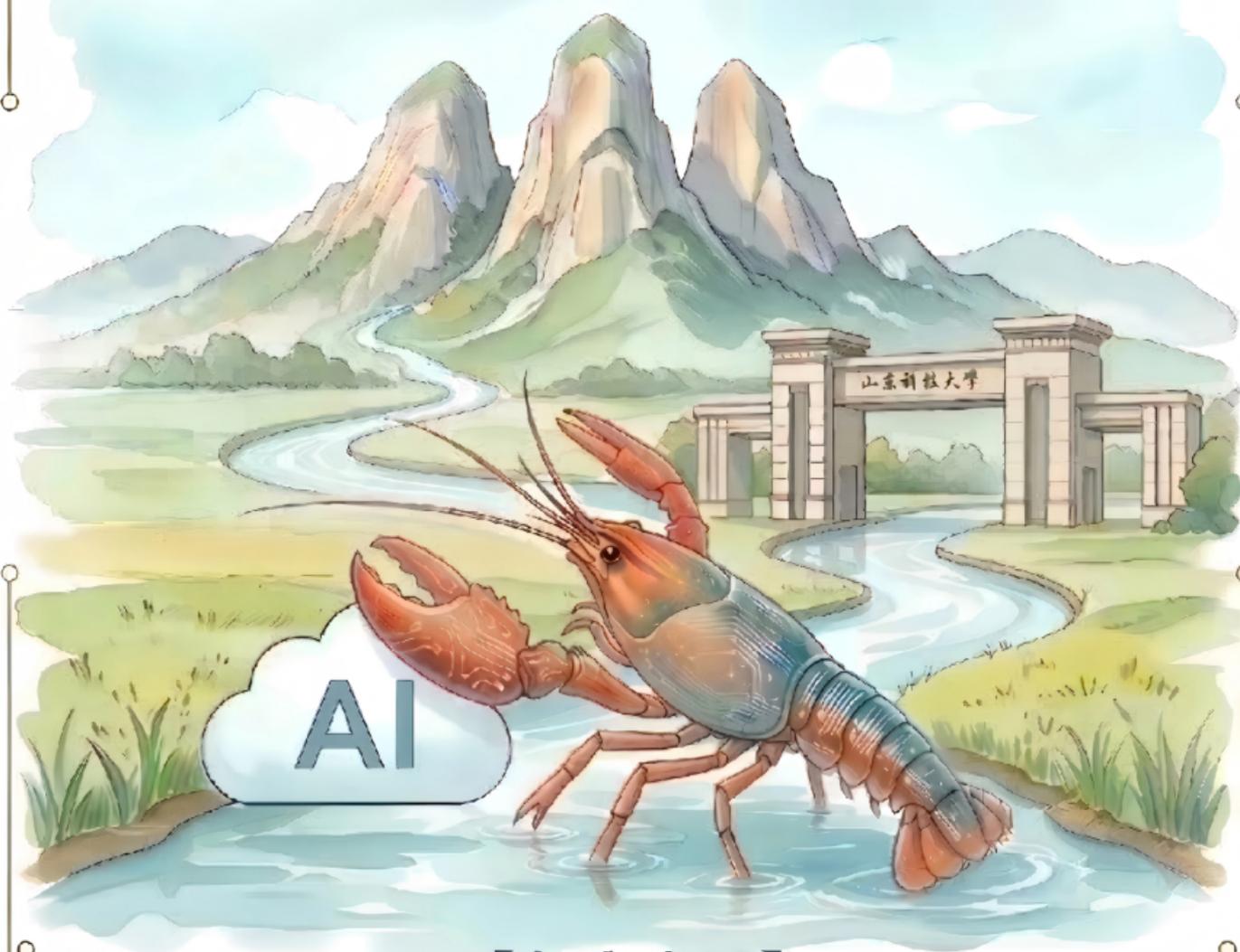


# 大学生就业能力提升 工作室发布

## OpenClaw 本地部署教程 离线、安全、高效



【免责声明】

本教程内容**仅供参考**，  
具体操作风险由用户自行承担。

# 目 录

一、OpenClaw 到底是什么？ .....	1
二、安装基础 .....	1
(一) 最低硬件与系统要求 .....	1
(二) 核心依赖 .....	1
三、详细部署步骤 .....	4
(一) Windows 原生一键脚本部署（新手首选） .....	4
(二) WSL2 环境部署（官方推荐，稳定首选） .....	4
四、OpenClaw 基础设置 .....	6
(一) 核心初始化配置（必做） .....	6
(二) Web 控制台基础设置 .....	10
(三) 核心配置文件说明 .....	11
五、启动验证与常见问题排查 .....	12
(一) 启动与功能验证 .....	12
(二) 常见问题排查 .....	12
(三) 重要注意事项 .....	12
(四) 其他可参考教程 .....	13

# 一、OpenClaw 到底是什么？

很多新手刚接触，只知道它叫“龙虾”，却不知道它到底能干嘛。简单说，OpenClaw 是一个开源的 AI 智能体框架，你可以把它理解成一个能 7×24 小时帮你干活的 AI 员工。它能做到的事远不止聊天：帮你处理邮件、管理日历、自动写代码、爬取网页数据、对接飞书/钉钉/微信群聊自动回复、执行定时任务，甚至能通过自定义技能实现你能想到的几乎所有自动化操作。而这一切的前提，就是先搞定 openclaw 安装与部署，这也是 90%新手卡壳的第一步。

## 二、安装基础

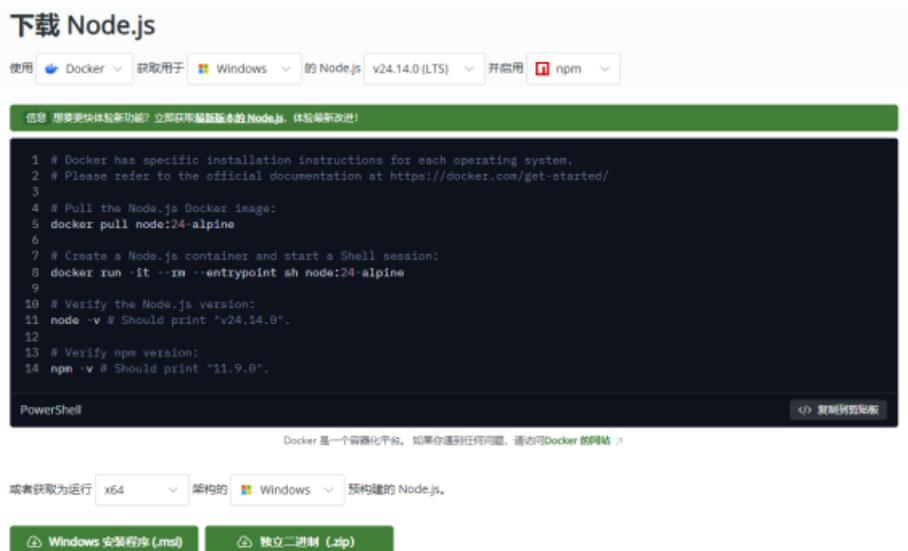
### （一）最低硬件与系统要求

- 系统：Windows10/11（64 位）、macOS12+、主流 Linux 发行版（Ubuntu/CentOS 等）
- CPU：最低 2 核，推荐 4 核及以上
- 内存：最低 2GB，推荐 4GB 及以上
- 磁盘：至少 10GB 可用空间
- 网络：能正常访问国内镜像源，无需特殊网络环境（本文有国内专属适配方案）

### （二）核心依赖

#### 1、Windows 系统前置依赖安装

①安装 Node.js 直接去 Node.js 官网下载 22+版本的 LTS 安装包，默认路径下一步安装即可，全程不用改任何配置。官网链接：<https://nodejs.cn/en/download>



②下载完成后双击安装包，全程默认下一步即可，务必勾选“Automatically install the necessary tools”，自动安装配套工具，避免后续编译报错。

③安装完成后，验证是否安装成功：按下 Win+R，输入 cmd 打开命令提示符，输入以下命令，输出版本号即代表安装成功。若提示“不是内部或外部命令”，重启电脑即可自动加载环境变量。

示例：

```
# 验证 Node.js 版本：  
node -v  
# 输出"v24.14.0"表示安装成功。  
#验证 npm 版本：  
npm -v  
# 输出"11.9.0"表示安装成功。
```

## 2、Git 依赖安装

用于源码拉取、自定义技能开发，官网链接（<https://github.com/git-for-windows/git/releases/download/v2.53.0.windows.2/Git-2.53.0.2-64-bit.exe>）下载安装，全程默认下一步即可，安装后在 PowerShell/CMD 终端输入 git --version 验证。

安装完成后配置 Git，不然可能会报错，按下 Win+S，搜索 PowerShell，右键选择以管理员身份运行下面代码：

```
git config --global url."https://github.com/"insteadOf ssh://git@github.com/  
git config --global url."https://".insteadOf git://
```



## 6、国内网络优化

国内用户需提前配置 npm 国内镜像源，避免依赖包下载超时，操作如下：

①打开普通 PowerShell/CMD 终端，输入以下命令切换淘宝镜像：

```
npm config set registry https://registry.npmmirror.com/
```

②验证是否生效，输入以下命令，返回镜像地址即代表配置成功：

```
npm config get registry
```

# 三、详细部署步骤

## （一）Windows 原生一键脚本部署（新手首选）

全程无需复杂操作，复制粘贴命令即可完成安装，10 分钟搞定。

①重新打开普通权限 PowerShell（无需管理员，避免后续配置文件权限异常）。

②复制以下官方一键安装命令，粘贴到终端中，按下回车执行（官方版：推荐）：

```
iwr -useb https://openclaw.ai/install.ps1 | iex
```

国内用户若出现网络超时，可使用中国社区镜像命令（国内修改版）：

```
iwr -useb https://open-claw.org.cn/install-cn.ps1 | iex
```

③等待安装完成：脚本会自动检测环境、补全缺失依赖、安装 OpenClaw 主程序，全程无需手动操作，耗时 3-10 分钟（取决于网络速度），过程中不要关闭终端。

④安装成功验证：终端显示 OpenClaw installed successfully，即代表主程序安装完成。

⑤验证安装：终端输入以下命令，输出版本号即代表部署成功：

```
openclaw --version
```

## （二）WSL2 环境部署（官方推荐，稳定首选）

OpenClaw 官方明确说明：Windows 用户通过 WSL2 运行可获得最佳兼容性，原生 Windows 未经完整测试，工具兼容性较差，长期使用推荐此方案。

### 1、安装 WSL2 环境

以管理员身份打开 PowerShell，输入以下命令，自动安装 WSL2 和 Ubuntu 系统：

```
wsl --install
```

执行完成后，重启电脑，系统会自动完成 Ubuntu 的后续配置，按提示设置用户名和密码即可。

## 2、进入 WSL2 Ubuntu 系统，更新系统依赖

打开 Ubuntu 终端，输入以下命令更新系统：

```
sudo apt update && sudo apt upgrade -y
```

## 3、配置环境

```
# 下载并安装 nvm:
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.3/install.sh | bas
h
# 代替重启 shell
\ "$HOME/.nvm/nvm.sh"
# 下载并安装 Node.js:
nvm install 24
# 验证 Node.js 版本:
node -v # 输出 "v24.14.0"表示安装成功.
# 验证 npm 版本:
npm -v # 输出 "11.9.0"表示安装成功.
```

验证安装：node -v 输出版本号 $\geq 22$  即可。

## 4、配置国内镜像源（同前文步骤），避免下载超时。

执行官方一键安装命令：

```
curl -fsSL https://openclaw.ai/install.sh | bash
```

国内用户使用镜像命令：

```
curl -fsSL https://open-claw.org.cn/install-cn.sh | bash
```

等待安装完成，终端显示成功提示后，输入 `openclaw --version` 验证部署结果。

## 5、补充方案：手动 npm 安装（进阶用户自定义）

适合需要自定义安装路径、管理版本的进阶用户，步骤如下：

完成前文所有环境依赖配置，确保 Node.js 版本符合要求。

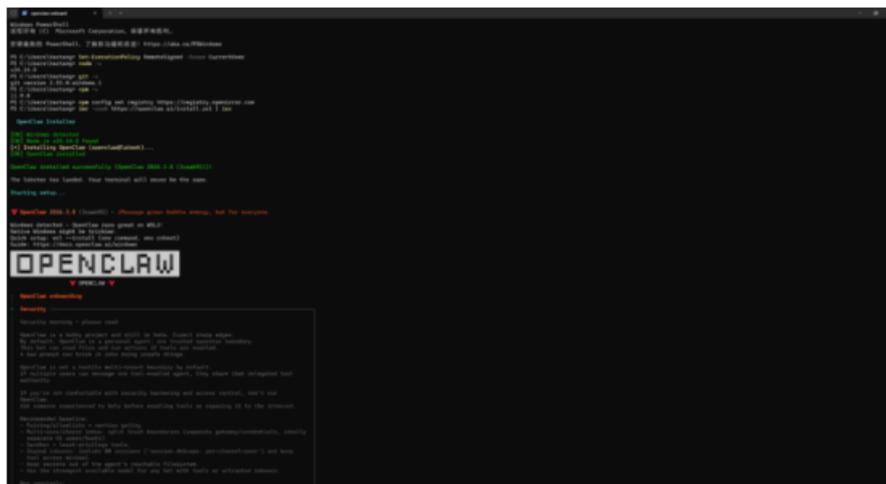
打开 PowerShell/终端，输入以下命令，全局安装 OpenClaw 最新版：

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

等待安装完成，输入 `openclaw --version` 验证安装结果即可。

## 四、OpenClaw 基础设置

跳出这个界面说明安装完成了，正常情况下会自动进入配置环节：



安装完成后，必须完成初始化配置，才能正常使用 OpenClaw 的核心功能。

### (一) 核心初始化配置（必做）

1、（可选：直接在安装过程中进入配置）打开 PowerShell/终端，输入以下命令，启动初始化配置向导：

```
openclaw onboard
```

2、若想配置开机自启、后台守护进程，输入以下命令：

```
openclaw onboard --install-daemon
```

3、按照终端向导分步完成配置：

①风险告知确认：终端会提示 OpenClaw 具备系统执行权限，输入 `yes` 或者左右方向键选择 `YES`，回车进入下一步。

```
openclaw-onboard x + v

Security

Security warning - please read.

OpenClaw is a hobby project and still in beta. Expect sharp edges.
By default, OpenClaw is a personal agent: one trusted operator boundary.
This bot can read files and run actions if tools are enabled.
A bad prompt can trick it into doing unsafe things.

OpenClaw is not a hostile multi-tenant boundary by default.
If multiple users can message one tool-enabled agent, they share that delegated tool
authority.

If you're not comfortable with security hardening and access control, don't run
OpenClaw.
Ask someone experienced to help before enabling tools or exposing it to the internet.

Recommended baseline:
- Pairing/allowlists + mention gating.
- Multi-user/shared inbox: split trust boundaries (separate gateway/credentials, ideally
  separate OS users/hosts).
- Sandbox + least-privilege tools.
- Shared inboxes: isolate DM sessions ('session.dmScope: per-channel-peer') and keep
  tool access minimal.
- Keep secrets out of the agent's reachable filesystem.
- Use the strongest available model for any bot with tools or untrusted inboxes.

Run regularly:
openclaw security audit --deep
openclaw security audit --fix

Must read: https://docs.openclaw.ai/gateway/security

I understand this is personal-by-default and shared/multi-user use requires lock-down. Continue?
  Yes / No
```

②运行模式选择：新手直接选择 quickstart（快速启动，默认配置），进阶用户选择 custom（自定义配置，可修改端口、工作目录）。

```
openclaw-onboard x + v

OpenClaw is not a hostile multi-tenant boundary by default.
If multiple users can message one tool-enabled agent, they share that delegated tool
authority.

If you're not comfortable with security hardening and access control, don't run
OpenClaw.
Ask someone experienced to help before enabling tools or exposing it to the internet.

Recommended baseline:
- Pairing/allowlists + mention gating.
- Multi-user/shared inbox: split trust boundaries (separate gateway/credentials, ideally
  separate OS users/hosts).
- Sandbox + least-privilege tools.
- Shared inboxes: isolate DM sessions ('session.dmScope: per-channel-peer') and keep
  tool access minimal.
- Keep secrets out of the agent's reachable filesystem.
- Use the strongest available model for any bot with tools or untrusted inboxes.

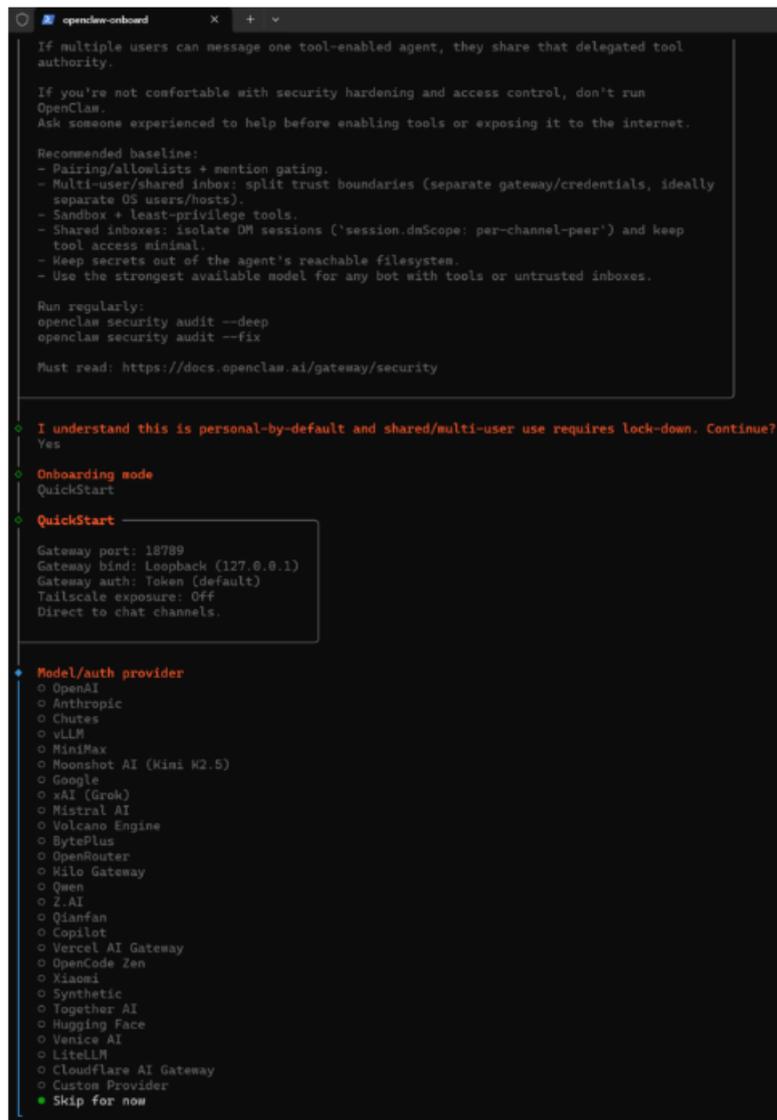
Run regularly:
openclaw security audit --deep
openclaw security audit --fix

Must read: https://docs.openclaw.ai/gateway/security

I understand this is personal-by-default and shared/multi-user use requires lock-down. Continue?
  Yes

Onboarding mode
  QuickStart (Configure details later via openclaw configure.)
  Manual
```

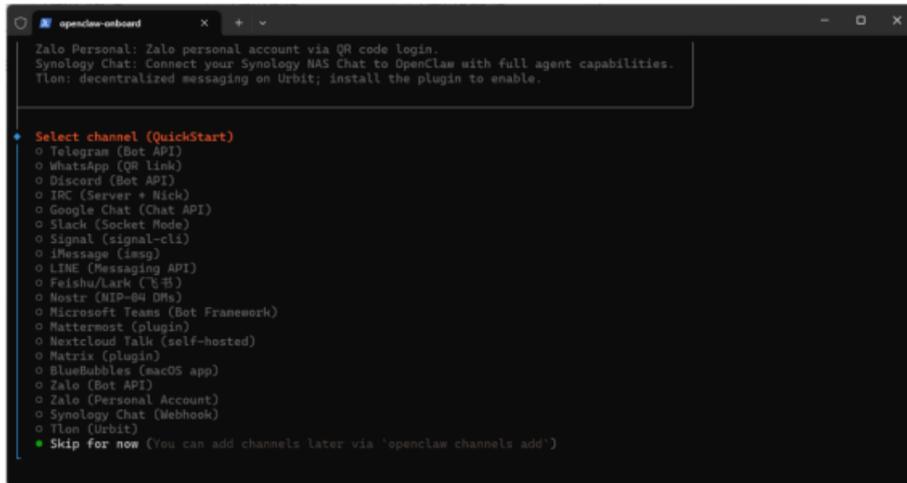
③大模型供应商选择（核心步骤）：



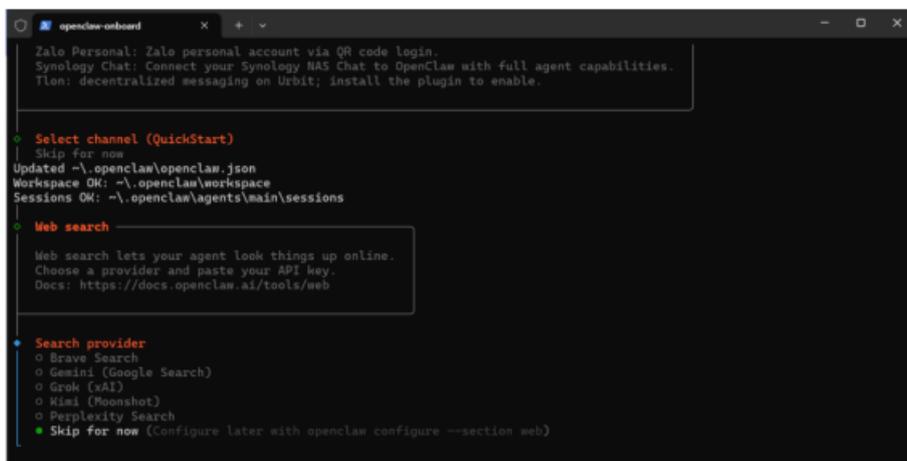
● 云端大模型：选择你使用的模型平台，可选择火山引擎、MiniMax、OpenAI、Claude、Kimi、通义千问、硅基流动等主流服务商，后续需粘贴对应 API Key。

● 云端模型：粘贴提前准备好的对应服务商 API Key，回车自动验证，验证通过即可进入下一步。

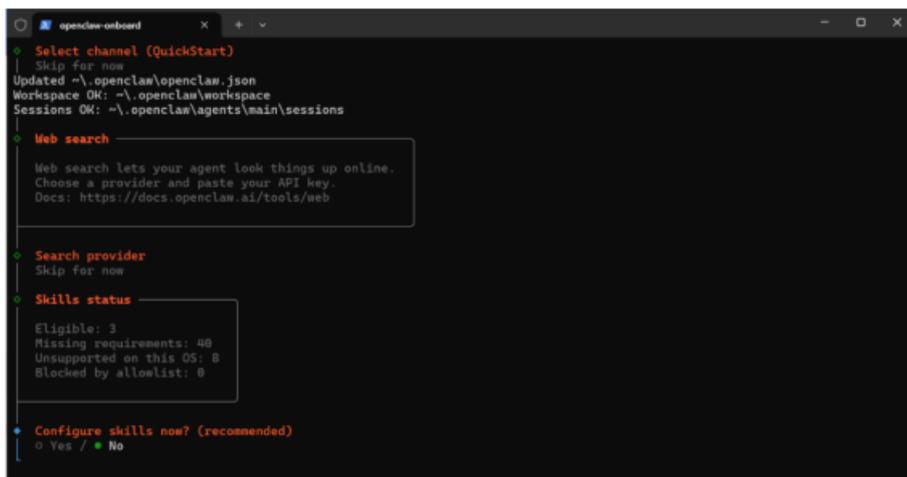
④通信方式配置（可先跳过，后面详细设置）



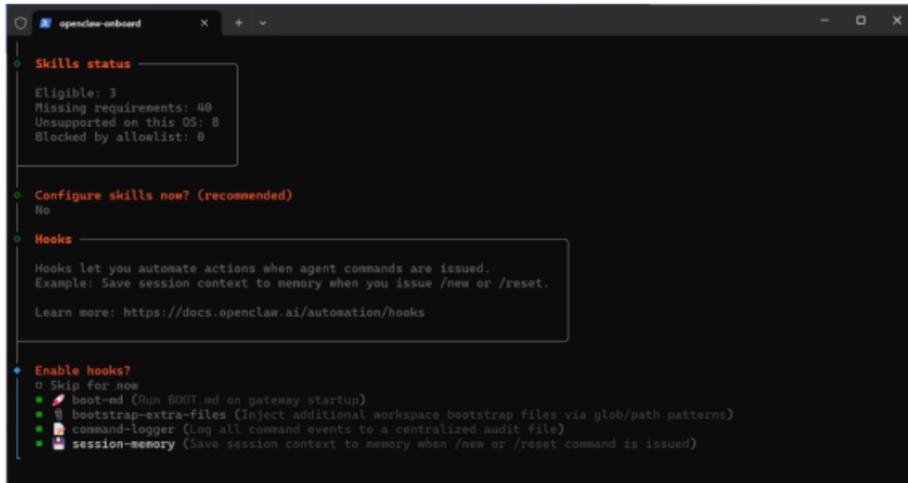
⑤搜索方式设置（可先跳过，后面详细设置）



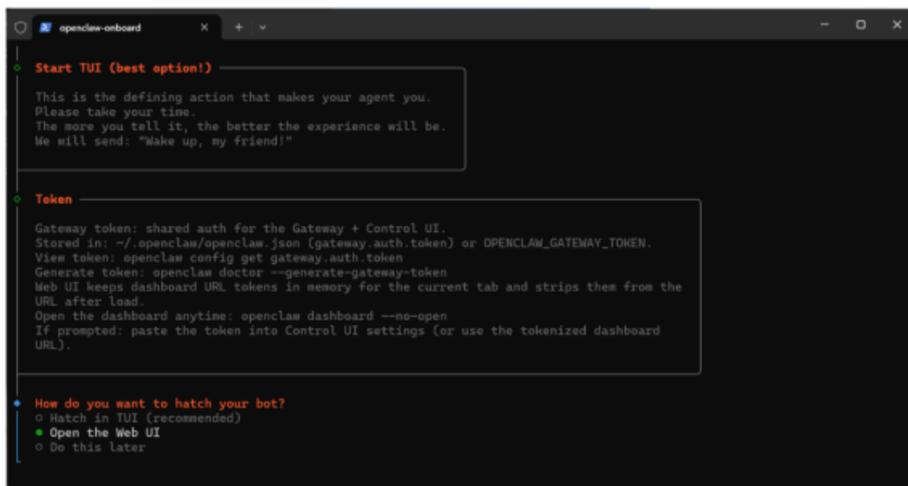
⑥技能配置（可先跳过，后面详细设置）



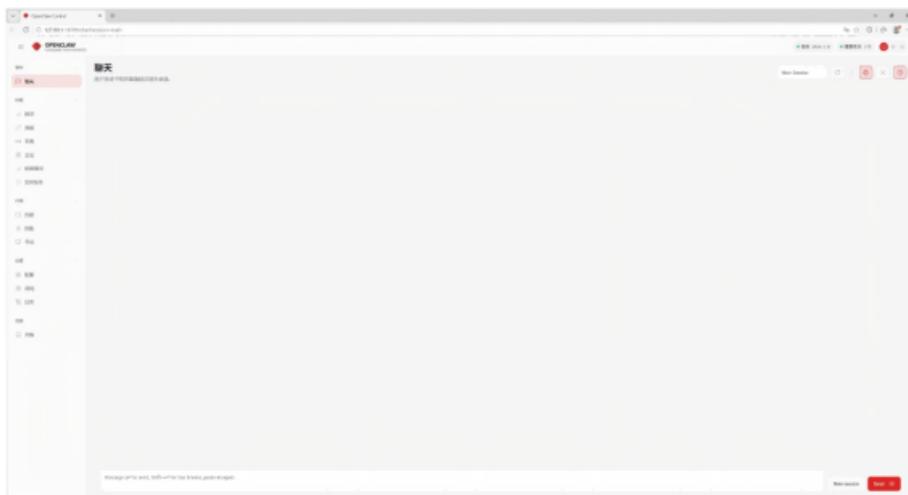
⑦日志配置（按需选择，可全选，选择的方法是通过方向键上下选择后按空格打开或关闭，然后按回车确认设定）



⑧选择机器人交互界面（按需选择后回车，可以选择 Web UI）



此时会自动打开浏览器界面，安装到此结束，后面是配置阶段



## （二）Web 控制台基础设置

OpenClaw 提供可视化 Web 控制台，所有基础设置均可在页面内完成，无需修改配置文件。

### 1、如果没有启动则启动 OpenClaw 服务

终端输入以下命令，启动核心服务：

```
openclaw start
```

终端显示 OpenClaw is running，即代表服务启动成功。

### 2、访问 Web 控制台

打开任意浏览器，输入默认地址：<http://127.0.0.1:18789>，即可进入控制台。

若浏览器未自动打开，手动复制地址访问即可；局域网内其他设备访问，将 127.0.0.1 替换为部署设备的局域网 IP。

### 3、核心基础设置

**访问安全设置：**进入控制台【设置】-【安全设置】，设置控制台登录密码、开启访问令牌验证，防止未授权访问，保障本地服务安全。

**模型参数设置：**进入【模型管理】，可切换默认模型、调整模型调用温度（temperature）、最大 Token 数、上下文窗口大小，新手建议保持默认参数。

**服务基础设置：**进入【系统设置】，可修改服务端口、工作目录、日志存储路径，配置服务异常自动重启、开机自启。

**基础技能配置：**进入【技能市场】，可启用/禁用基础自动化技能，设置技能的执行权限、调用限制，新手建议先启用系统基础技能。

## （三）核心配置文件说明

OpenClaw 所有配置均保存在配置文件中，进阶用户可直接修改配置文件完成自定义设置，配置文件路径：

Windows 原生环境：C:\Users\你的用户名\openclaw\openclaw.json

WSL2/Linux 环境：~/openclaw/openclaw.json

核心配置项说明：

配置项	作用
env	存储 API Key、环境变量等敏感信息
agents.defaults.model	配置默认使用的大模型、主模型/备用模型
server	配置服务端口、绑定地址、超时时间
auth	配置访问权限、登录凭证、令牌有效期
log	配置日志级别、存储路径、日志文件大小限制

## 五、启动验证与常见问题排查

### （一）启动与功能验证

每次使用前，在终端输入 `openclaw start` 启动服务，服务启动成功后，访问 `http://127.0.0.1:18789` 进入控制台。

在对话框输入简单指令，如“打开我的电脑”“查看当前系统磁盘占用”，若 OpenClaw 正常执行，即代表部署完全成功。

停止服务：终端按下 `Ctrl+C`，或输入 `openclaw stop` 即可停止服务。

### （二）常见问题排查

1、脚本执行报错，提示“无法加载文件 因为在此系统上禁止运行脚本”

解决：重新以管理员身份打开 PowerShell，执行前文的权限解锁命令，输入 Y 确认后，重新执行安装命令。

2、安装过程中提示 `network timeout/下载超时`

解决：检查网络连接，配置 `npm` 国内淘宝镜像，切换手机热点重试，关闭 VPN/代理工具。

3、安装完成后，输入 `openclaw` 提示“不是内部或外部命令”

解决：重启电脑重新加载环境变量；若仍报错，重新安装 Node.js，确保勾选了“Add to PATH”选项。

4、服务启动失败，提示端口 18789 被占用

解决：关闭占用端口的程序，或启动时指定其他端口，命令如下：

```
openclaw start --port 18790
```

5、模型连接失败，提示 API Key 无效

解决：检查 API Key 是否正确、是否有可用余额、接口地址是否配置正确，部分国内服务商需要配置自定义 `baseUrl`。

### （三）重要注意事项

1、微信、企业微信等平台对非官方自动化操作有严格风控机制，请勿使用 OpenClaw 高频执行自动化操作，避免账号功能受限或封禁。

2、请勿将 OpenClaw 用于执行支付、实名、敏感信息读取等高危操作，避免造成财产损失和信息泄露。

3、所有安装路径、工作目录禁止包含中文、空格和特殊字符，避免出现服务异常、文件读取失败等问题。

4、第三方杀毒软件可能会误判 OpenClaw 的自动化操作，若出现拦截，可将 OpenClaw 加入白名单。

#### （四）其他可参考教程

1、<https://zhuanlan.zhihu.com/p/2015451351220528886>

2、[https://blog.csdn.net/weixin\\_72188539/article/details/158882412?fromshare=blogdetail&sharetype=blogdetail&sharerId=158882412&sharerefer=PC&sharesource=qq\\_52422697&sharefrom=from\\_link](https://blog.csdn.net/weixin_72188539/article/details/158882412?fromshare=blogdetail&sharetype=blogdetail&sharerId=158882412&sharerefer=PC&sharesource=qq_52422697&sharefrom=from_link)

3、<https://cloud.tencent.com/developer/article/2625313>

4、[https://blog.csdn.net/weixin\\_50764857/article/details/159006711?fromshare=blogdetail&sharetype=blogdetail&sharerId=159006711&sharerefer=PC&sharesource=qq\\_52422697&sharefrom=from\\_link](https://blog.csdn.net/weixin_50764857/article/details/159006711?fromshare=blogdetail&sharetype=blogdetail&sharerId=159006711&sharerefer=PC&sharesource=qq_52422697&sharefrom=from_link)