

MP510-80 MapleClaw 個人 AI Agent 快速安裝指南



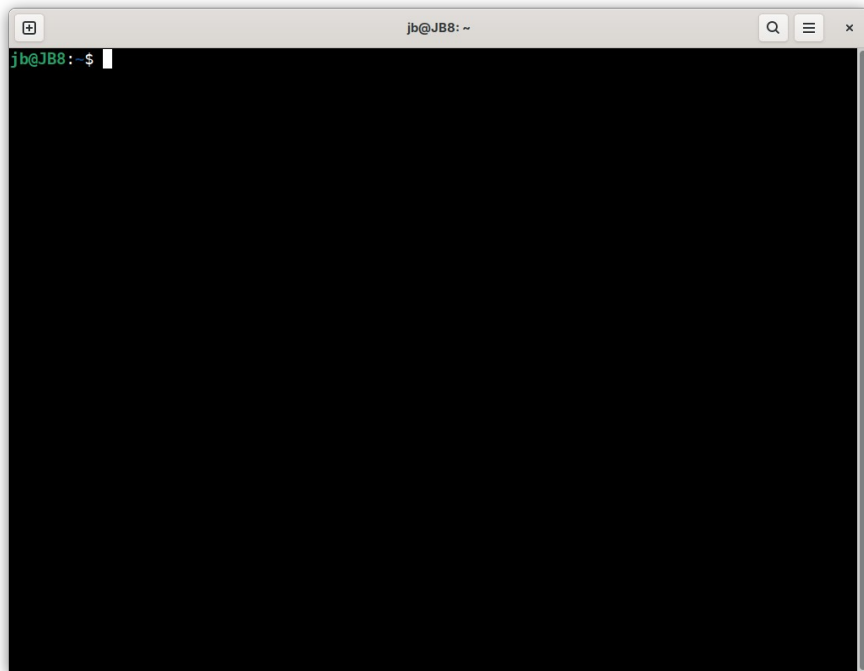
(./images/a0-hero.png)

目標是在你的手機、筆電或桌機安裝即使通訊軟體 Telegram，在 Telegram 中建立一個 bot，取名 jkmolty_bot（或者任何符合一般電腦使用者命名規則的名稱均可，但是結尾一定要是 bot 三個字母）。然後將 jkmolty_bot 綁 (bind) 到 MP510-80 MapleClaw 裡的 OpenClaw Channel 上。

你就可以在 Telegram 中，對 jkmolty_bot 發號施令，要它幫你做事。

MP510-80 MapleClaw 已經預裝好 OpenClaw，以及所需的相依套件，使用者只須要設定 LLM 模型和連接 Telegram，立即可用。

在 Linux、或 Windows（推薦使用 Powershell 終端機）、或 Mac 中打開終端機：



(./images/a1.png)

將 MP510-80 MapleClaw 接上網路線和電源線後，稍待幾分鐘，MapleClaw 前面板 LED 顯示器開始輪流顯示網路 IP 位址(例如：192.168.xx.yy)、CPU 溫度、現在時刻。

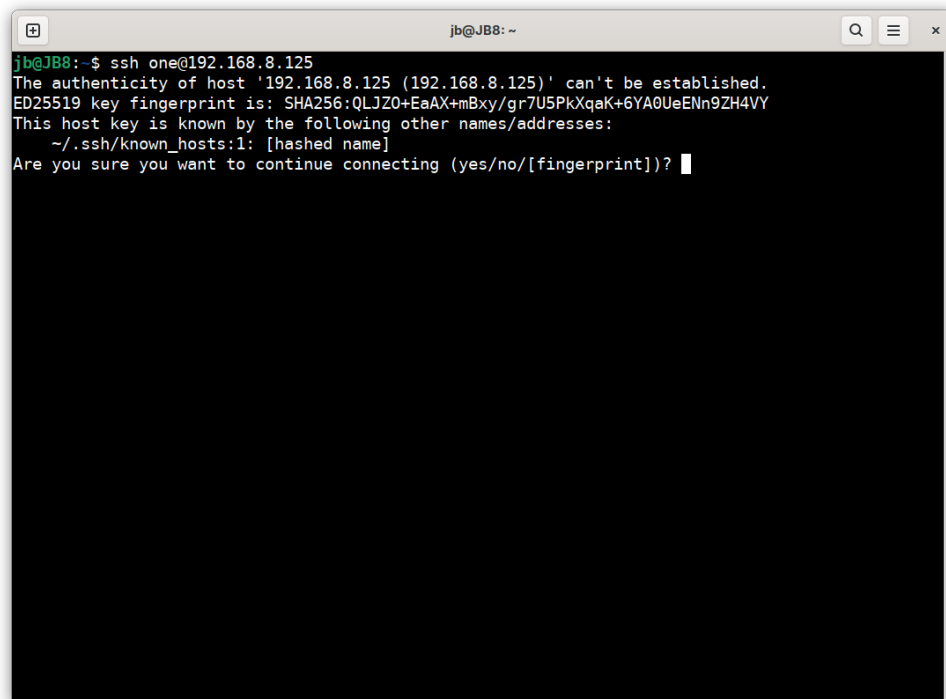


(./images/a1-1.png)

下面的例子假設 MapleClaw 的 ip 位址為：192.168.8.125；主機名稱為：DS104
預設使用者名稱：one，密碼：1234

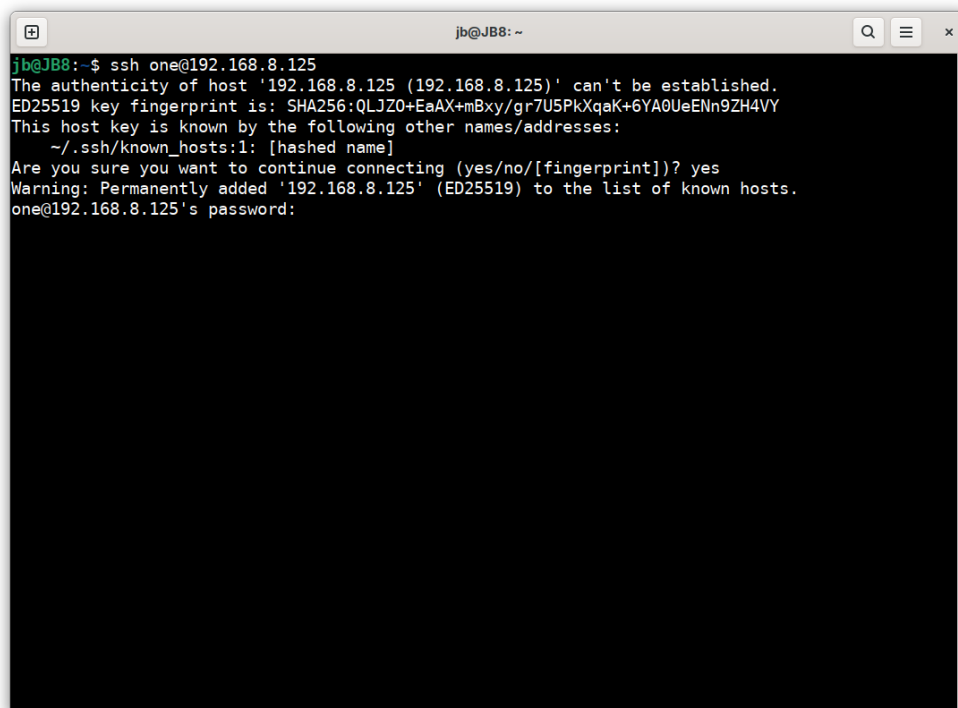
在終端機中，用 ssh 命令登入 MapleClaw：

\$ ssh one@192.168.8.125



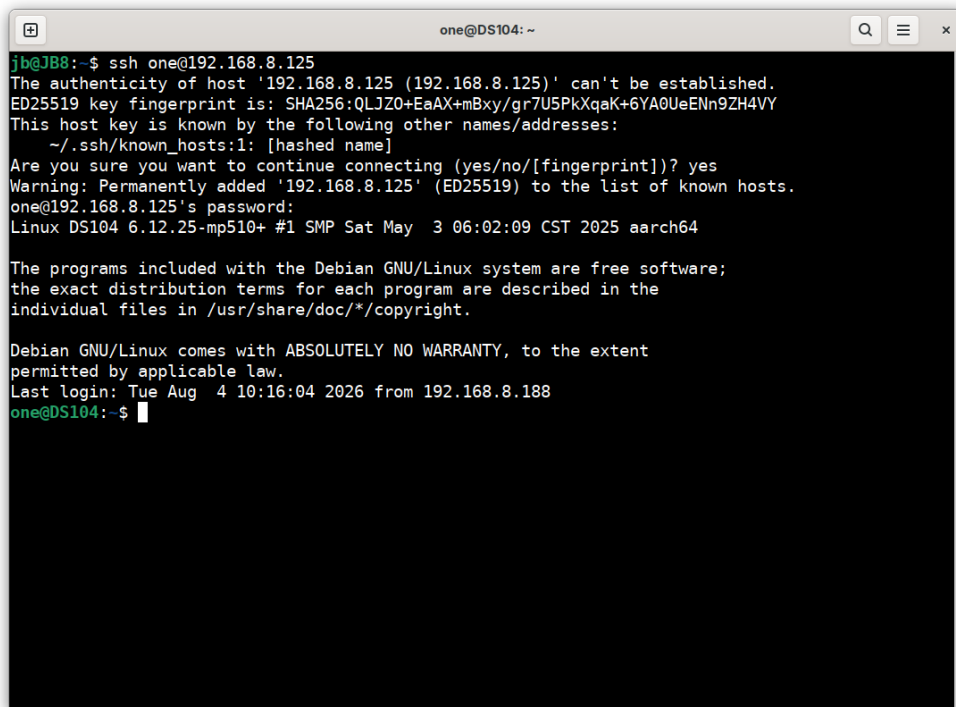
(./images/a3.png)

輸入 yes 確認要登入 MapleClaw。如果先前已經登入過，就不會再詢問。



(./images/a4.png)

輸入密碼 1234,



```
one@DS104: ~  
jb@JB8:~$ ssh one@192.168.8.125  
The authenticity of host '192.168.8.125 (192.168.8.125)' can't be established.  
ED25519 key fingerprint is: SHA256:QLJZ0+EeAX+mBxy/gr7U5PkXqaK+6YA0UeENn9ZH4VY  
This host key is known by the following other names/addresses:  
  ~/.ssh/known_hosts:1: [hashed name]  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.8.125' (ED25519) to the list of known hosts.  
one@192.168.8.125's password:  
Linux DS104 6.12.25-mp510+ #1 SMP Sat May 3 06:02:09 CST 2025 aarch64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Tue Aug 4 10:16:04 2026 from 192.168.8.188  
one@DS104:~$
```

(./images/a5.png)

登入 MapleClaw 成功,

one@DS104: ~\$ 各欄位意義說明如下:

- \$: 命令輸入提示
- ~: 當前的工作目錄是在家目錄, ~ 代表家目錄
- DS104: 主機名稱
- @: 在
- one: 使用者名稱

雖然 OpenClaw 支援的模型非常多, 只要市面上有新的模型都會被列入支援。但是在這裏 MapleClaw 重點推薦 OpenAI 的 Codex 和 GPT-5.6 sol 組合。因為:

一、使用強模型非常重要, 不但可以順暢完成任務, 而且可以避免被駭, 例如提示注入攻擊。使用弱模型不但錯誤百出, 而且容易被駭。雖然開源本地模型很具前瞻性, 但是現在生態環境還不成熟, 至少還需要兩、三年以上的時間, 才會成熟到一般人可以順暢使用。現在使用, 保證卡卡的, 花下一大堆時間精力, 換來巨大的挫折感, 甚至對代理人式 AI 產生嚴重懷疑, 得不償失。Codex + GPT-5.6 sol 是目前最前緣最適合的強模型組合之一。

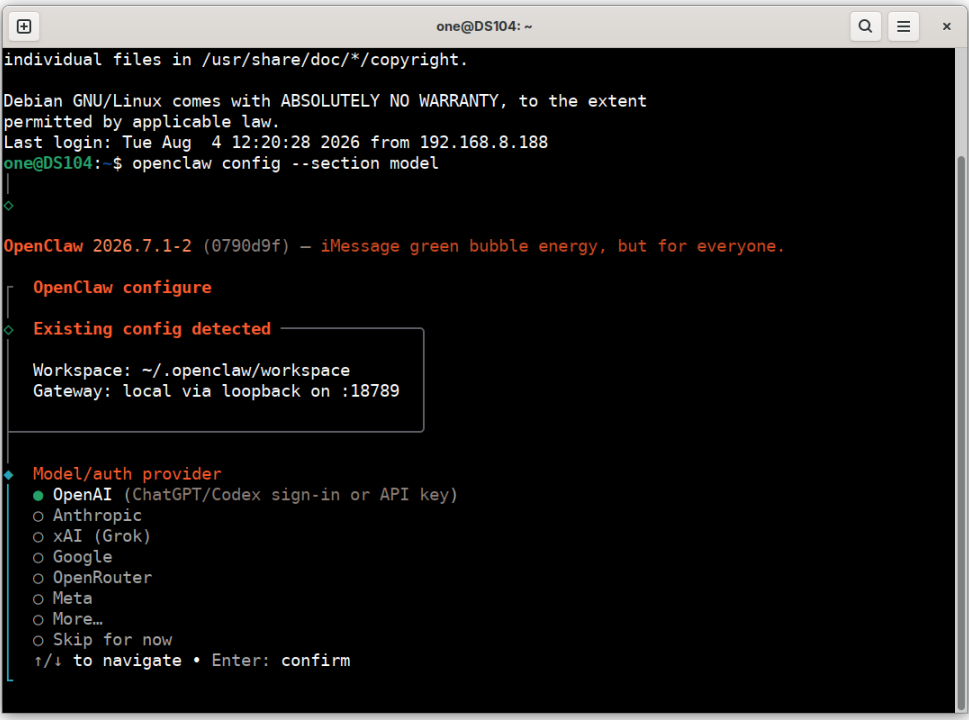
二、使用 ChatGPT Plus 訂閱版，每月 20 美元，最合適。現有的訂閱用戶立即可使用。其他強模型，例如 Anthropic、Google，無法使用訂閱授權，必須使用 API key 授權，以使用量計價。剛開始，還不知道如何控制預算，萬一誤操作，有可能導致巨額的使用費被刷信用卡。

三、Codex 是在 MapleClaw 本機中執行的 agent，和 GPT-5.6 模型都是 OpenAI 開發的，兩者最 match。採用其他模型時，OpenClaw 使用 Pi agent，最佳化不如 Codex + GPT-5.6。

在終端機中輸入命令

```
$ openclaw config --section model
```

選擇安裝模型



```
one@DS104: ~  
individual files in /usr/share/doc/*/copyright.  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Tue Aug 4 12:20:28 2026 from 192.168.8.188  
one@DS104:~$ openclaw config --section model  
◇  
OpenClaw 2026.7.1-2 (0790d9f) — iMessage green bubble energy, but for everyone.  
  
OpenClaw configure  
◇ Existing config detected  
Workspace: ~/.openclaw/workspace  
Gateway: local via loopback on :18789  
  
◆ Model/auth provider  
● OpenAI (ChatGPT/Codex sign-in or API key)  
○ Anthropic  
○ xAI (Grok)  
○ Google  
○ OpenRouter  
○ Meta  
○ More...  
○ Skip for now  
↑/↓ to navigate • Enter: confirm
```

(./images/a8.png)

確定點選 OpenAI，然後按 enter 鍵

```
one@DS104: ~  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Tue Aug  4 12:20:28 2026 from 192.168.8.188  
one@DS104:~$ openclaw config --section model  
|  
◇  
  
OpenClaw 2026.7.1-2 (0790d9f) – iMessage green bubble energy, but for everyone.  
  
  OpenClaw configure  
◇ Existing config detected  
  Workspace: ~/.openclaw/workspace  
  Gateway: local via loopback on :18789  
  
◇ Model/auth provider  
  OpenAI  
  
◆ OpenAI auth method  
  ● ChatGPT Login (Sign in with your ChatGPT or Codex subscription)  
  ○ OpenAI API Key  
  ○ Back  
  ↑/↓ to navigate • Enter: confirm
```

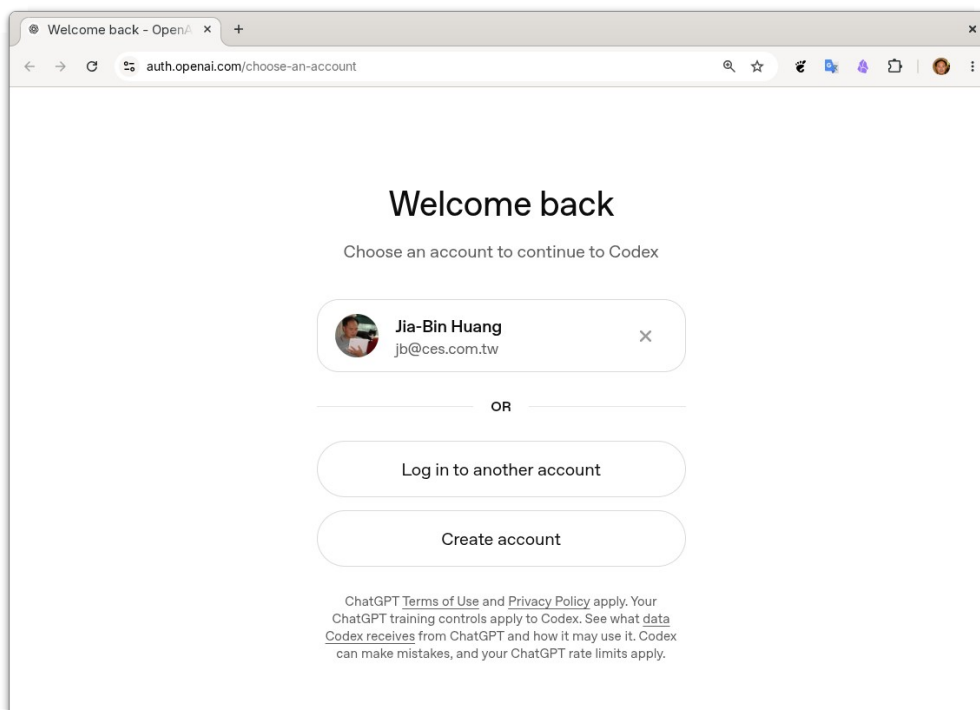
(./images/a9.png)

選擇 ChatGPT Login，然後按 enter

```
one@DS104: ~  
◇ Model/auth provider  
  OpenAI  
  
◇ OpenAI auth method  
  ChatGPT Login  
  
◇ OpenAI Codex OAuth  
  You are running in a remote/VPS environment.  
  A URL will be shown for you to open in your LOCAL browser.  
  Open it, sign in, then paste the redirect URL here.  
  If this OpenClaw process can receive the browser callback, sign-in may finish  
  automatically before you paste.  
  
◇ OAuth URL ready  
  
Open this URL in your LOCAL browser:  
  
https://auth.openai.com/oauth/authorize?response_type=code&client_id=app_  
_____  
true&originator=openclaw  
  
◆ Paste the authorization code (or full redirect URL):  
|
```

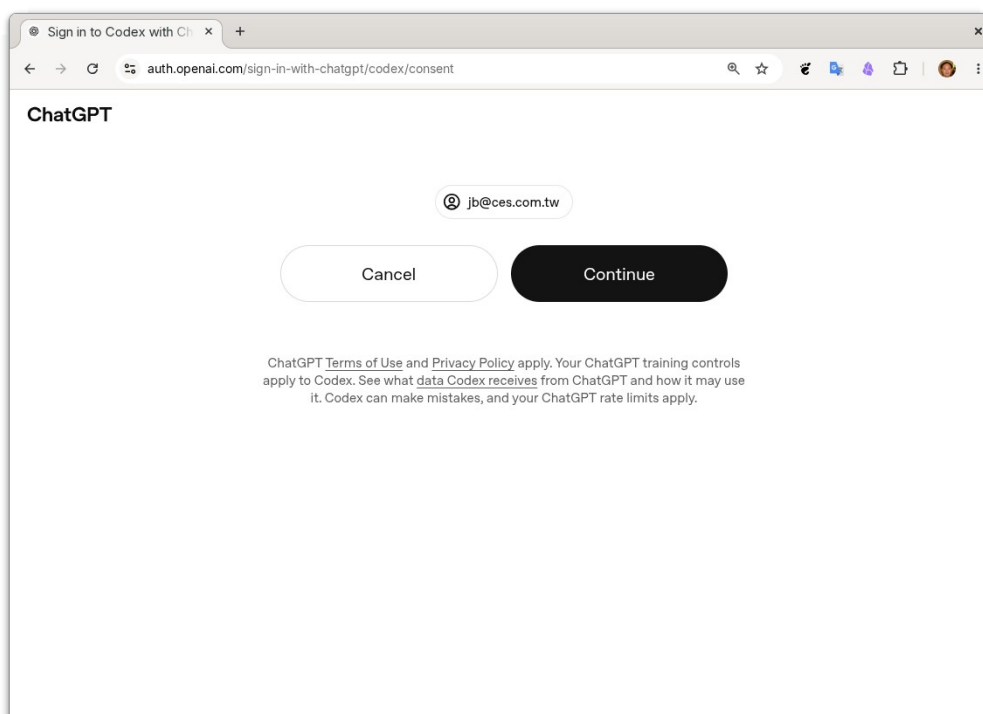
(./images/a10.png)

螢幕中會顯示一包含 token key 的 URL (已部份遮蓋住)。用滑鼠同時按 ctrl 鍵點選此 URL 打開瀏覽器，或將 URL 複製貼到瀏覽器對話框中，



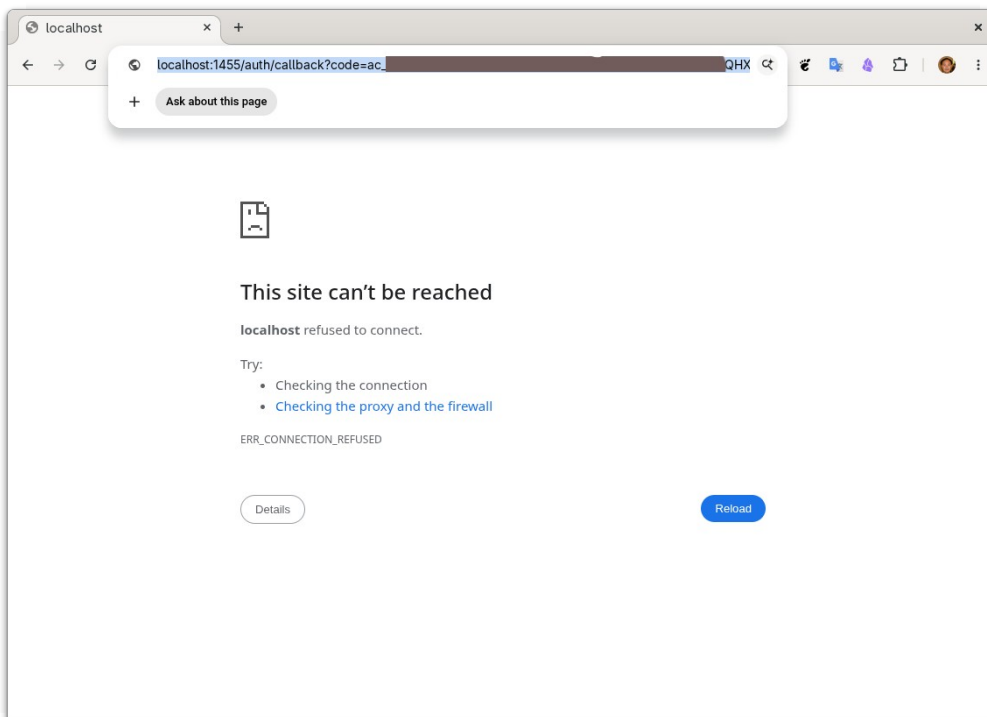
(./images/a11.png)

進入 OpenAI 的身份認證網頁，



(./images/a12.png)

按下 Continue，完成身份認證，



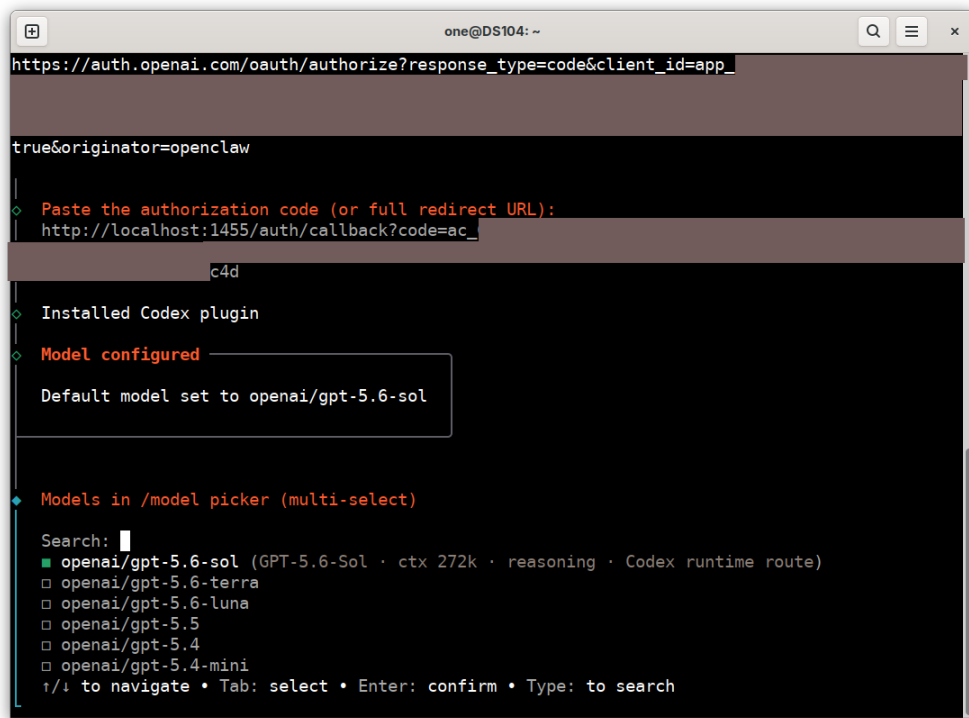
(./images/a13.png)

接下來看似吊詭的，網頁回應 This site can't be reached，但是這是正確的。因為 OpenAI 回應一包包含授權碼的本機 (localhost) URL。本機是指 MapleClaw。把這個 URL 從瀏覽器中的對話框複製下來，貼到終端機裏，

```
one@DS104: ~  
OpenAI  
◆ OpenAI auth method  
ChatGPT Login  
◆ OpenAI Codex OAuth  
You are running in a remote/VPS environment.  
A URL will be shown for you to open in your LOCAL browser.  
Open it, sign in, then paste the redirect URL here.  
If this OpenClaw process can receive the browser callback, sign-in may finish  
automatically before you paste.  
◆ OAuth URL ready  
Open this URL in your LOCAL browser:  
https://auth.openai.com/oauth/authorize?response_type=code&client_id=app_  
true&originator=openclaw  
◆ Paste the authorization code (or full redirect URL):  
http://localhost:1455/auth/callback?code=ac_  
c4d
```

(./images/a14.png)

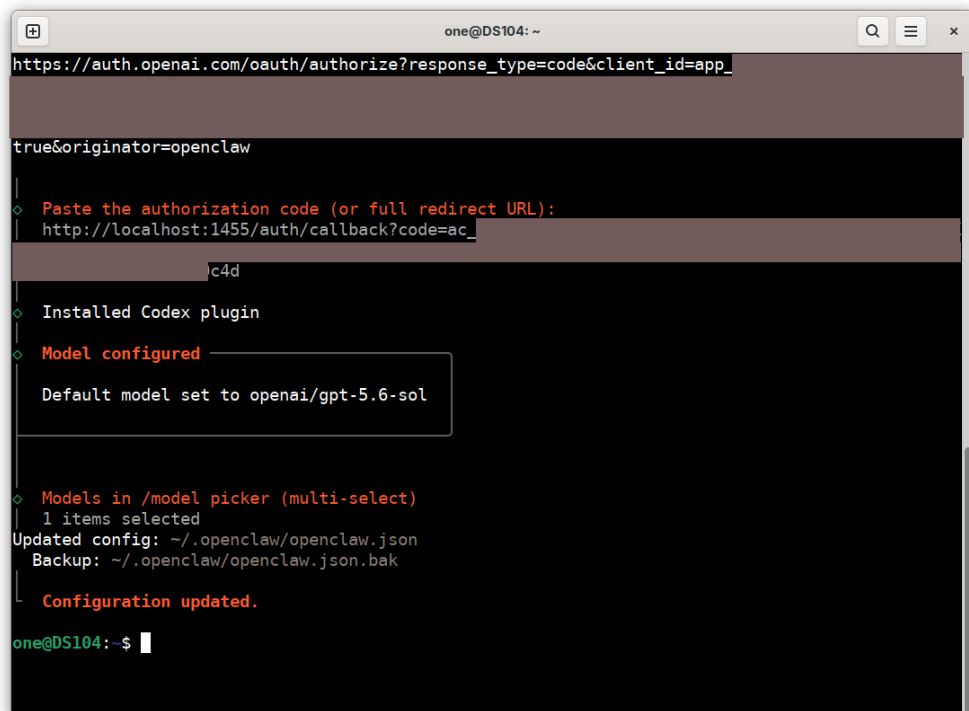
按 enter,



```
one@DS104: ~  
https://auth.openai.com/oauth/authorize?response_type=code&client_id=app_  
  
true&originator=openclaw  
  
Paste the authorization code (or full redirect URL):  
http://localhost:1455/auth/callback?code=ac_  
c4d  
  
Installed Codex plugin  
Model configured  
Default model set to openai/gpt-5.6-sol  
  
Models in /model picker (multi-select)  
Search:   
■ openai/gpt-5.6-sol (GPT-5.6-Sol · ctx 272k · reasoning · Codex runtime route)  
□ openai/gpt-5.6-terra  
□ openai/gpt-5.6-luna  
□ openai/gpt-5.5  
□ openai/gpt-5.4  
□ openai/gpt-5.4-mini  
↑/↓ to navigate • Tab: select • Enter: confirm • Type: to search
```

(./images/a16.png)

再按 enter 確認是 openai/gpt-5.6-sol，完成模型安裝設定：



```
one@DS104: ~  
https://auth.openai.com/oauth/authorize?response_type=code&client_id=app_  
  
true&originator=openclaw  
  
Paste the authorization code (or full redirect URL):  
http://localhost:1455/auth/callback?code=ac_  
c4d  
  
Installed Codex plugin  
Model configured  
Default model set to openai/gpt-5.6-sol  
  
Models in /model picker (multi-select)  
1 items selected  
Updated config: ~/.openclaw/openclaw.json  
Backup: ~/.openclaw/openclaw.json.bak  
Configuration updated.  
one@DS104: ~$
```

(./images/a17.png)

下一步安裝 Channel，亦即即時通訊軟體。OpenClaw 支援了幾乎市面上所有的即時通訊軟體，但是 MapleClaw 推薦 Telegram，因為：

全球即時通訊軟體用戶數排名（按每月活躍用戶 MAU 計）

排名	軟體名稱	每月活躍用戶數 (MAU)	主要核心市場與特點
1	WhatsApp	約 30 億 ~ 33 億	全球第一，主導歐美、南亞、拉美及非洲市場。
2	WeChat（微信）	約 14.1 億	中國市場霸主，集通訊、支付、小程序於一身的超級 App。
3	Facebook Messenger	約 10 億 ~ 13 億	歐美主流，與 Facebook 社交生態深度綁定。
4	Telegram（電報）	約 10 億	全球成長最快，主打隱私、大型群組與區塊鏈 Web3 生態。
5	Snapchat	約 9.3 億	歐美年輕人首選，主打閱後即焚與 AR 濾鏡特效。
6	Tencent QQ	約 5.3 億	中國市場，騰訊早期的老牌 IM，現以年輕族群與辦公為主。
7	Rakuten Viber	約 2.6 億 ~ 3 億	中東歐、東南亞及部分歐洲地區愛用。隸屬日本樂天集團。
8	Discord	約 2 億	全球遊戲與社群群組，以高畫質語音和社群伺服器為核心。
9	LINE	約 1.7 億 ~ 1.9 億	東亞特化型，台灣、日本、泰國的「國民級」基礎建設軟體。
10	KakaoTalk	約 5,000 萬	南韓獨佔，在南韓本地市佔率高達 99% 的絕對主流。

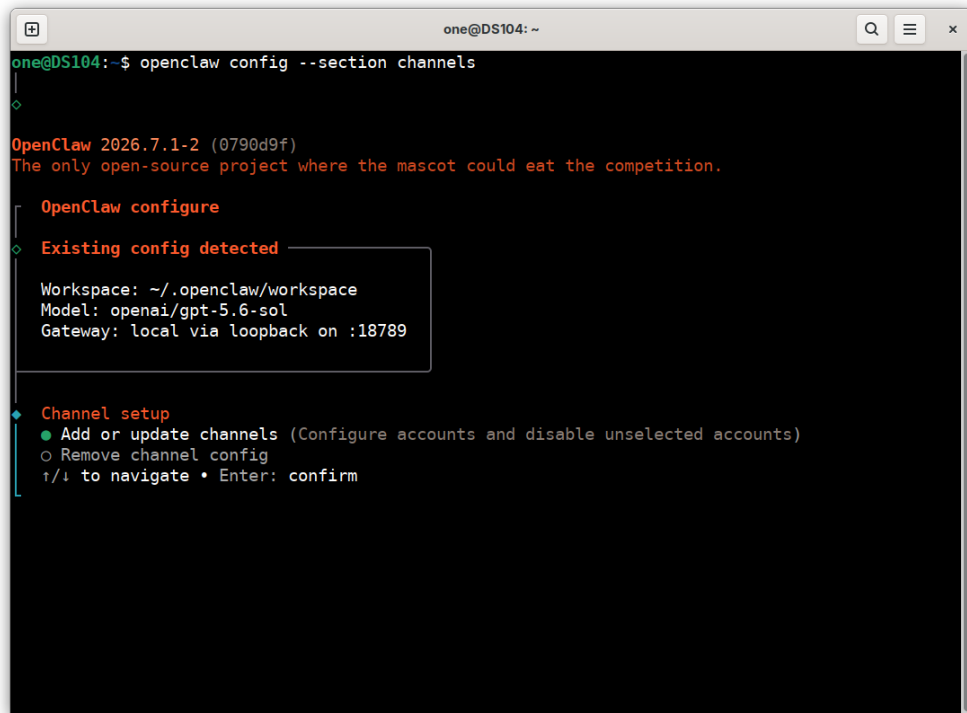
一、Telegram 有 10 億活躍用戶，是世界上主要即時通訊軟體之一，規模度夠大，做的相當不錯。反觀 Line 只有 1.8 億活躍用戶，是小眾軟體。

二、Telegram 在 Web3 和 AI 上的技術支援非常好，它的 bot API 深獲 OpenClaw 開發者和使用者的喜好。

三、只在 Telegram 上透過 Bot 和 OpenClaw 對話，隔開我們日常使用的 Line，也是一種穩健的安全措施。

在終端機上輸入 openclaw 命令設定 channel：

\$ openclaw config --section channels

A terminal window titled 'one@DS104: ~' showing the output of the command 'openclaw config --section channels'. The output includes the OpenClaw version (2026.7.1-2), a mascot message, and configuration details for the workspace, model, and gateway. It also shows a 'Channel setup' menu with options to add or update channels or remove channel config. A box highlights the 'Existing config detected' section.

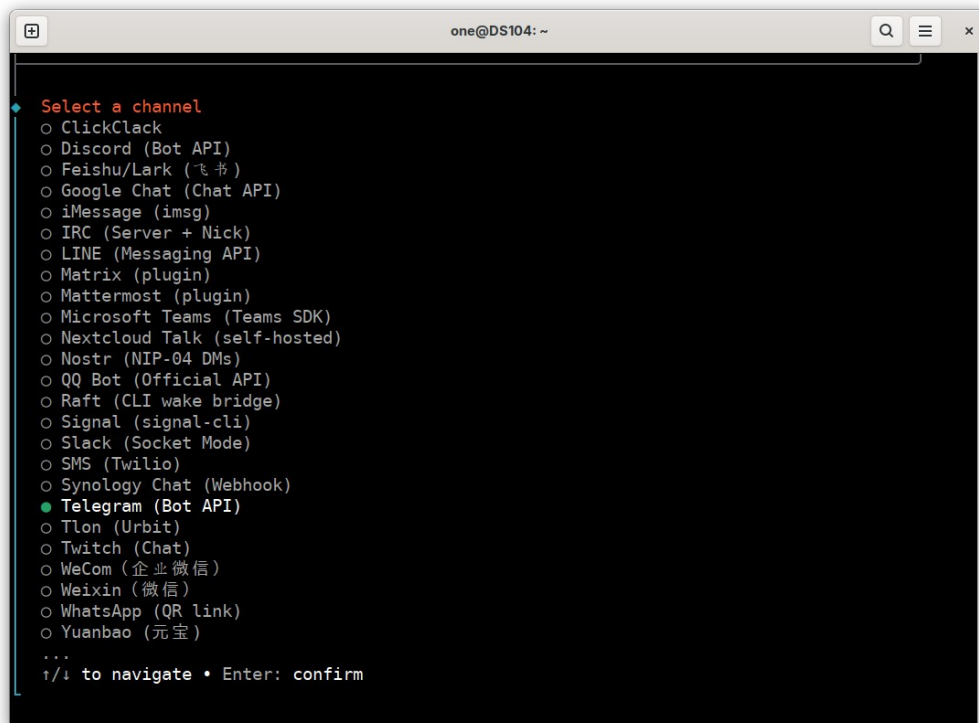
```
one@DS104:~$ openclaw config --section channels
|
◇
OpenClaw 2026.7.1-2 (0790d9f)
The only open-source project where the mascot could eat the competition.

┌ OpenClaw configure
◇ Existing config detected
  Workspace: ~/.openclaw/workspace
  Model: openai/gpt-5.6-sol
  Gateway: local via loopback on :18789
└

◆ Channel setup
  ● Add or update channels (Configure accounts and disable unselected accounts)
  ○ Remove channel config
  ↑/↓ to navigate • Enter: confirm
```

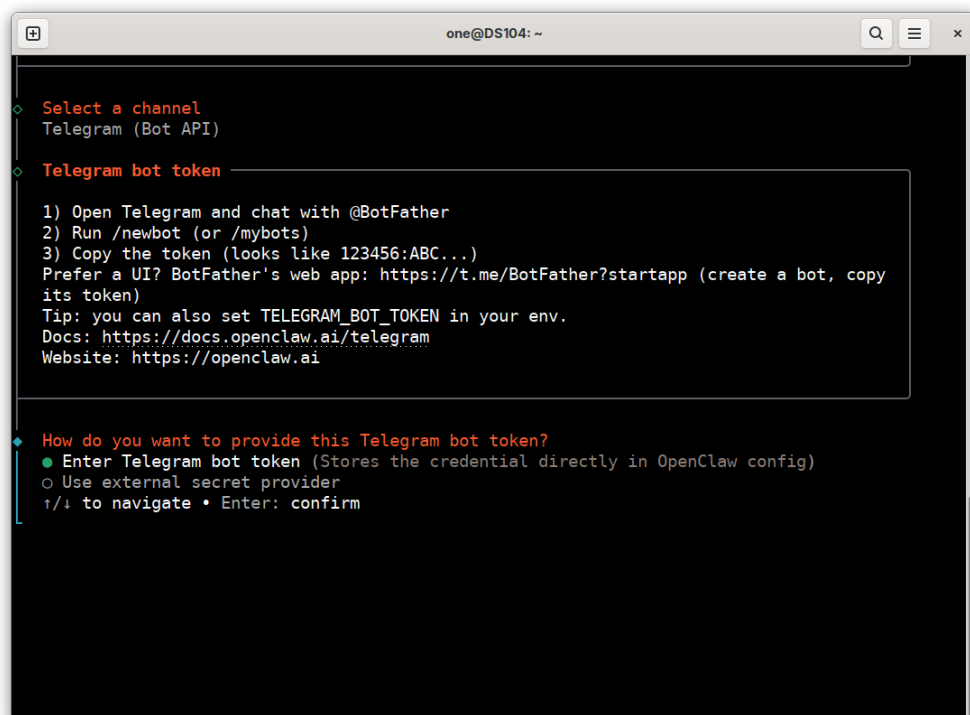
(./images/a19.png)

選取 Add or update channels, 按 enter



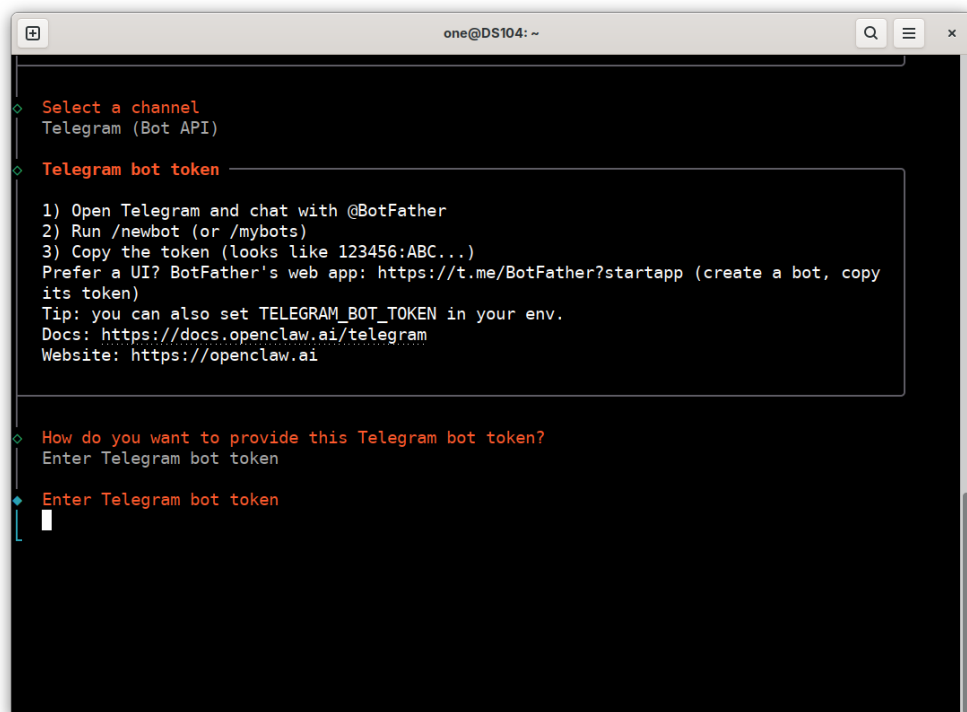
(./images/a20.png)

選取 Telegram，按 enter



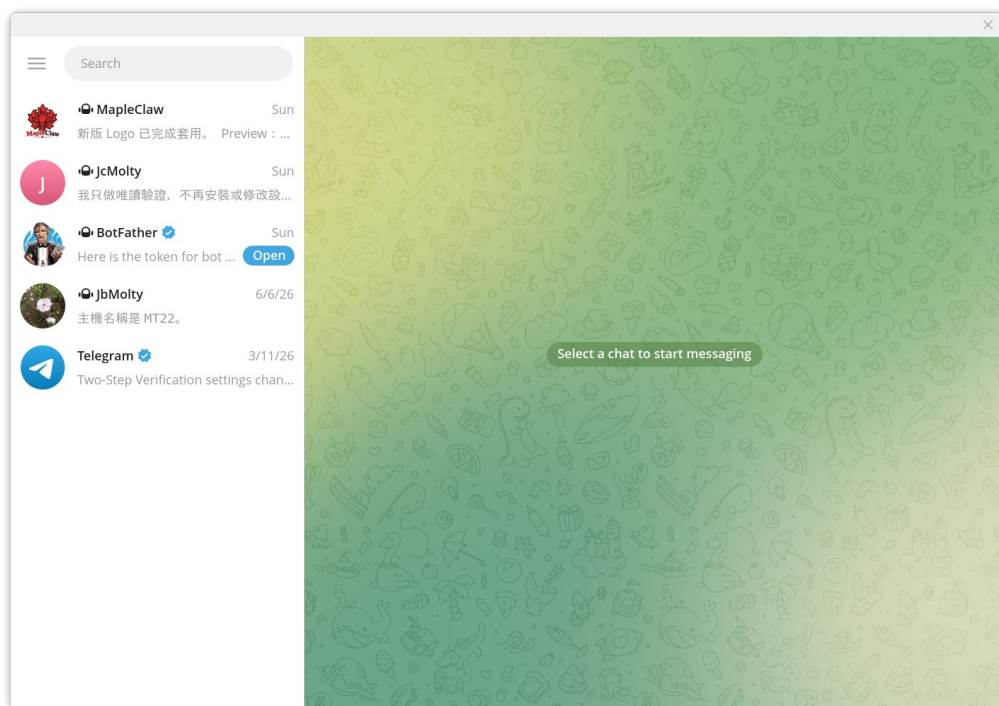
(./images/a21.png)

選取 Enter Telegram bot token,



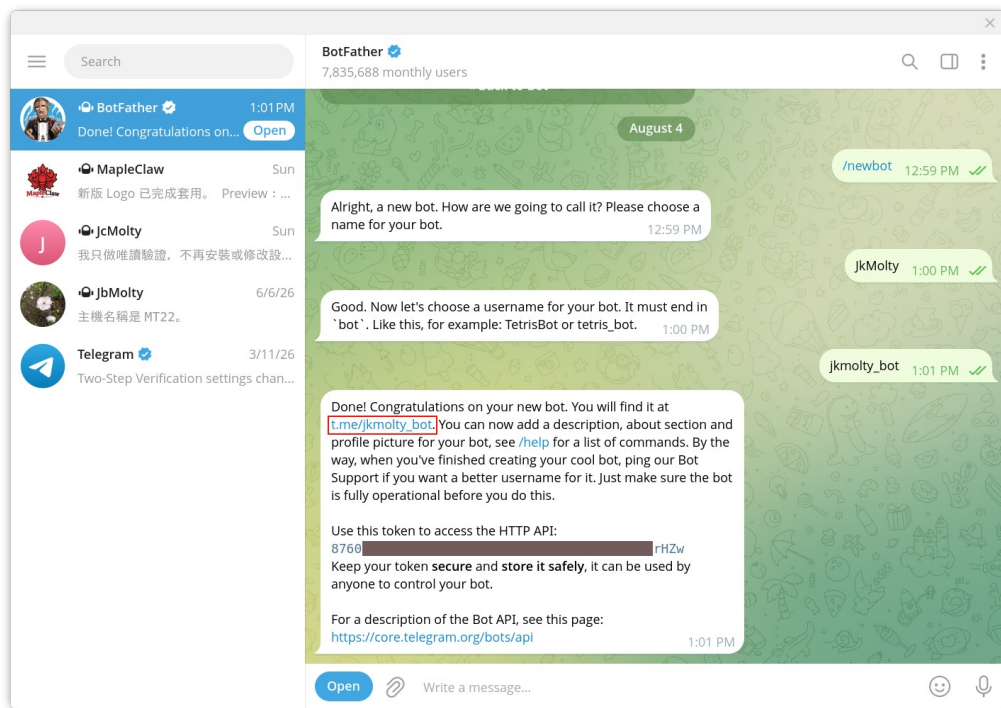
(./images/a22.png)

依照 Telegram 官網指示步驟，在手機和筆電/桌機上安裝好 Telegram，



(./images/a23.png)

點選 BotFather 來創建 OpenClaw Agent bot 分身，



(./images/a26.png)

在 Telegram 對話框輸入 /newbot,

BotFather 回應：新 bot 怎麼稱呼？

回應：JkMolty (任何符合命名法則的名字均可)

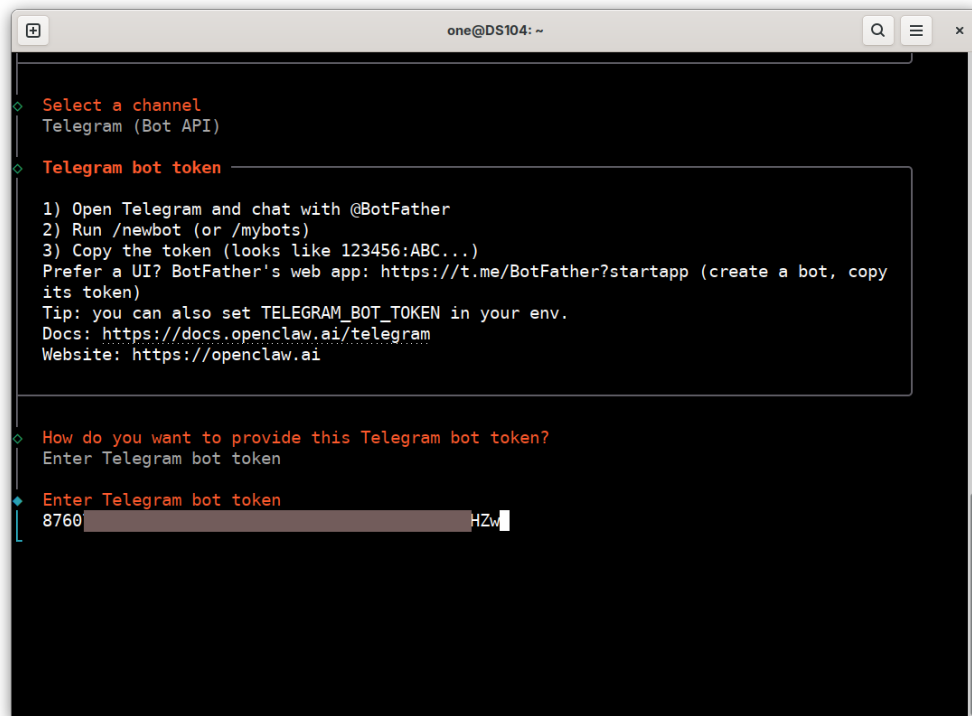
BotFather：新 bot 的使用者名稱 (必須以 bot 三個字母做結尾)？

回應：jkmolty_bot

BotFather 回應的落落長訊息中，兩個重點：

一、新 bot 呼叫位址：t.me/jkmolty_bot。很重要，但是只需要用到一次，第一次呼叫它的時候。

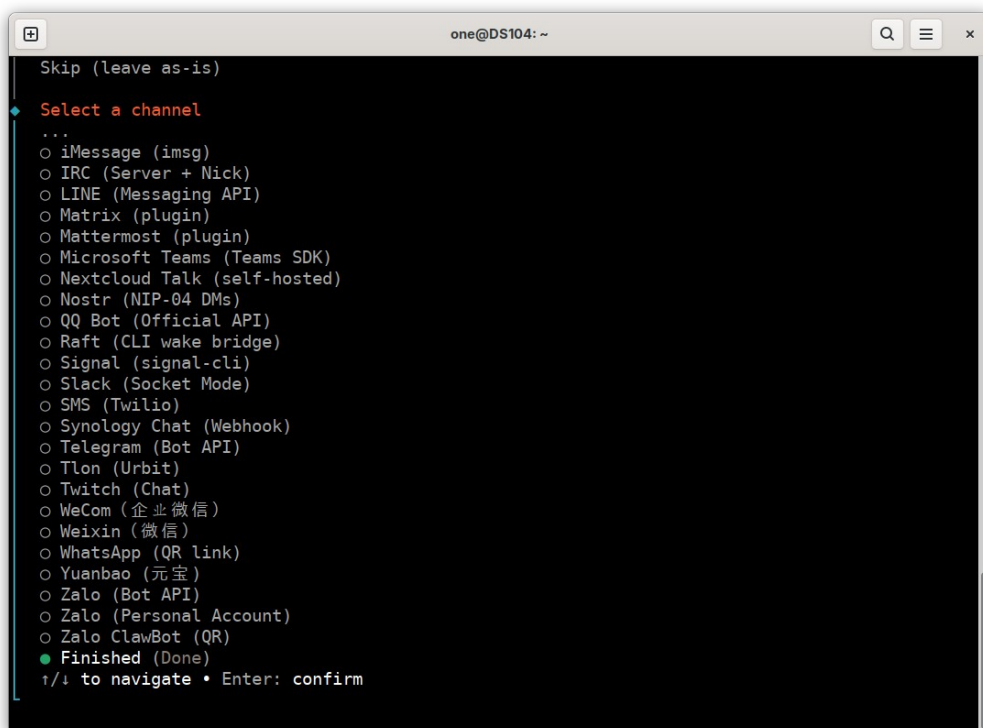
二、Telegram token：8760 ，請立即點擊此 token 複製到剪貼簿裏。



```
one@DS104: ~  
Select a channel  
Telegram (Bot API)  
Telegram bot token  
1) Open Telegram and chat with @BotFather  
2) Run /newbot (or /mybots)  
3) Copy the token (looks like 123456:ABC...)  
Prefer a UI? BotFather's web app: https://t.me/BotFather?startapp (create a bot, copy its token)  
Tip: you can also set TELEGRAM_BOT_TOKEN in your env.  
Docs: https://docs.openclaw.ai/telegram  
Website: https://openclaw.ai  
How do you want to provide this Telegram bot token?  
Enter Telegram bot token  
Enter Telegram bot token  
8760 HZw
```

(./images/a28.png)

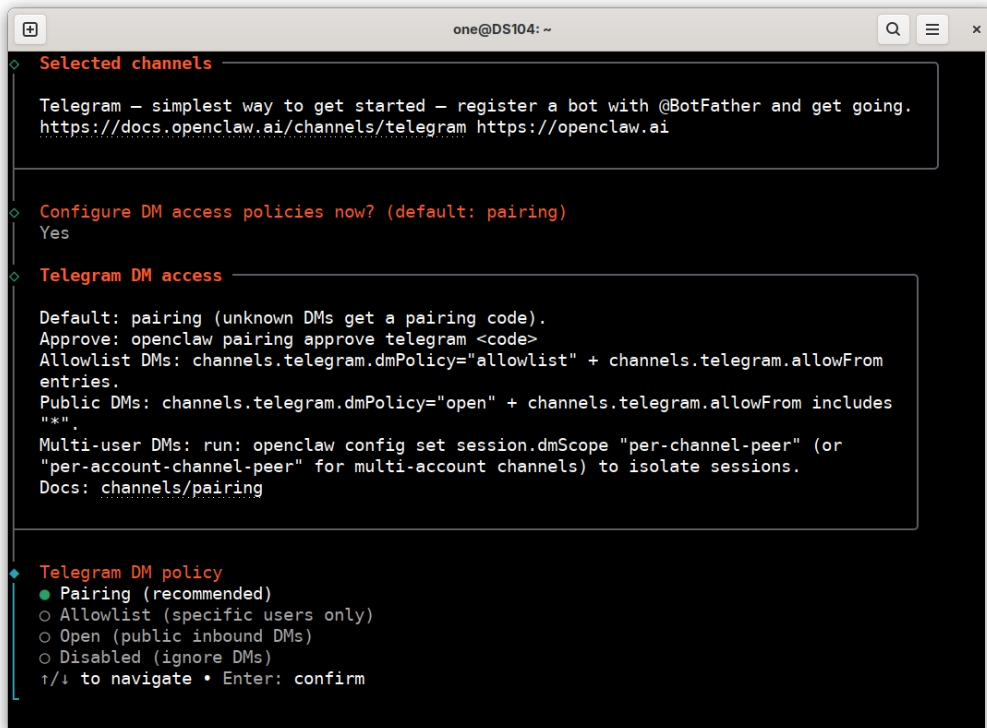
將 Telegram token 貼入終端機裏，按 enter，



```
one@DS104: ~  
Skip (leave as-is)  
Select a channel  
...  
o iMessage (img)  
o IRC (Server + Nick)  
o LINE (Messaging API)  
o Matrix (plugin)  
o Mattermost (plugin)  
o Microsoft Teams (Teams SDK)  
o Nextcloud Talk (self-hosted)  
o Nostr (NIP-04 DMs)  
o QQ Bot (Official API)  
o Raft (CLI wake bridge)  
o Signal (signal-cli)  
o Slack (Socket Mode)  
o SMS (Twilio)  
o Synology Chat (Webhook)  
o Telegram (Bot API)  
o Tlon (Urbit)  
o Twitch (Chat)  
o WeCom (企业微信)  
o Weixin (微信)  
o WhatsApp (QR link)  
o Yuanbao (元宝)  
o Zalo (Bot API)  
o Zalo (Personal Account)  
o Zalo ClawBot (QR)  
● Finished (Done)  
↑/↓ to navigate • Enter: confirm
```

(./images/a29.png)

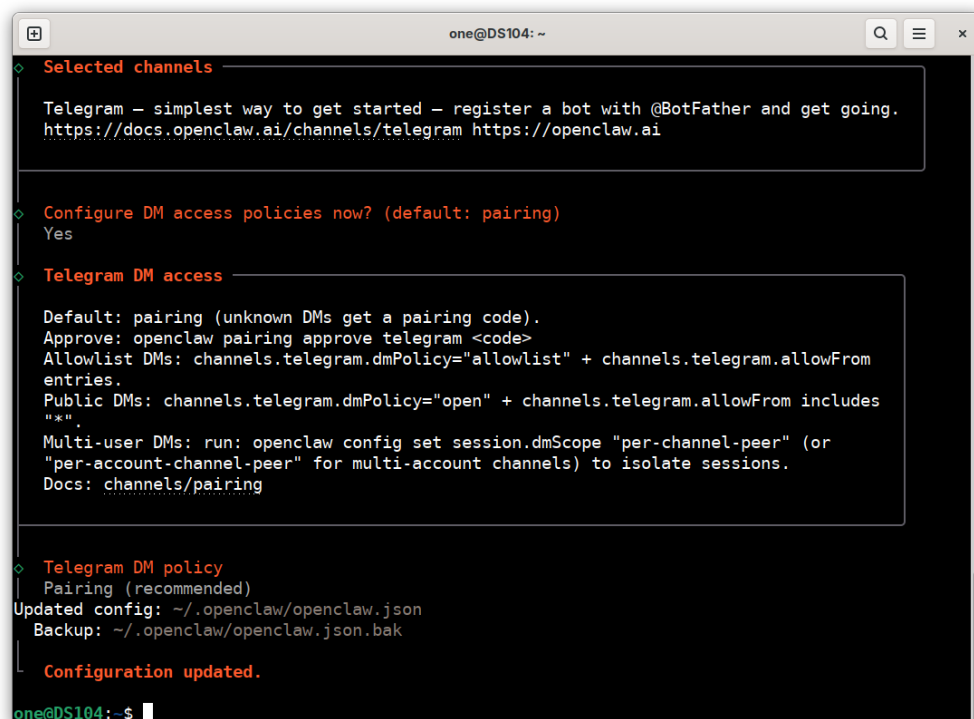
移到選取 Channel 的最後：Finished，按 enter，



```
one@DS104: ~  
Selected channels  
Telegram – simplest way to get started – register a bot with @BotFather and get going.  
https://docs.openclaw.ai/channels/telegram https://openclaw.ai  
  
Configure DM access policies now? (default: pairing)  
Yes  
Telegram DM access  
Default: pairing (unknown DMs get a pairing code).  
Approve: openclaw pairing approve telegram <code>  
Allowlist DMs: channels.telegram.dmPolicy="allowlist" + channels.telegram.allowFrom  
entries.  
Public DMs: channels.telegram.dmPolicy="open" + channels.telegram.allowFrom includes  
"*".  
Multi-user DMs: run: openclaw config set session.dmScope "per-channel-peer" (or  
"per-account-channel-peer" for multi-account channels) to isolate sessions.  
Docs: channels/pairing  
  
Telegram DM policy  
● Pairing (recommended)  
○ Allowlist (specific users only)  
○ Open (public inbound DMs)  
○ Disabled (ignore DMs)  
↑/↓ to navigate • Enter: confirm
```

(./images/a30.png)

選取 Pairing，按 enter，

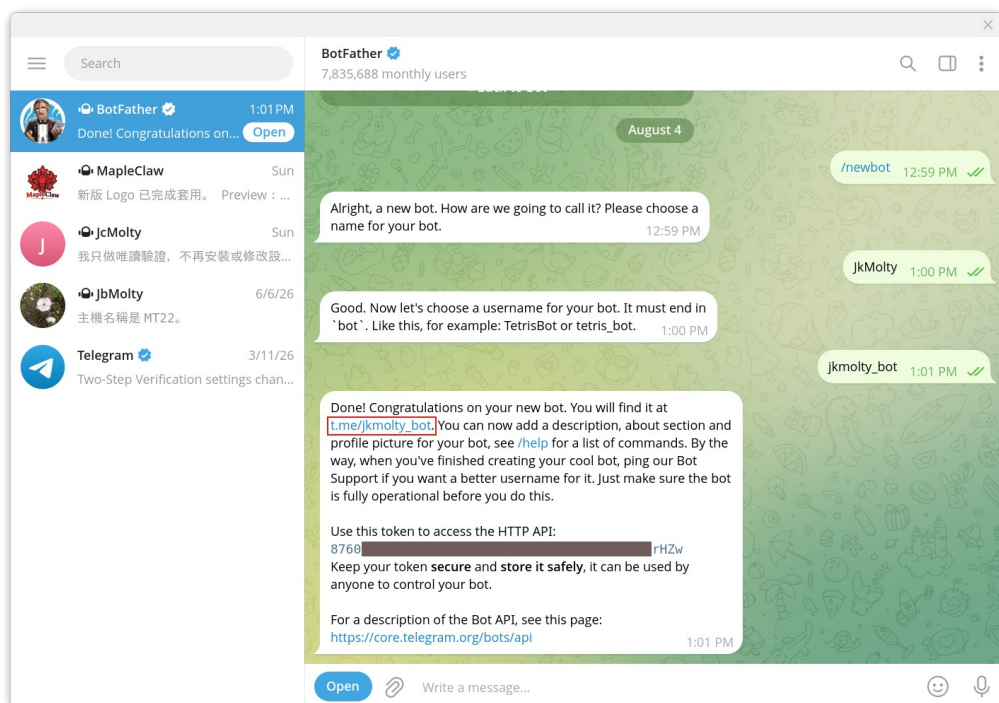


```
one@DS104: ~  
Selected channels  
Telegram – simplest way to get started – register a bot with @BotFather and get going.  
https://docs.openclaw.ai/channels/telegram https://openclaw.ai  
  
Configure DM access policies now? (default: pairing)  
Yes  
Telegram DM access  
Default: pairing (unknown DMs get a pairing code).  
Approve: openclaw pairing approve telegram <code>  
Allowlist DMs: channels.telegram.dmPolicy="allowlist" + channels.telegram.allowFrom  
entries.  
Public DMs: channels.telegram.dmPolicy="open" + channels.telegram.allowFrom includes  
"*".  
Multi-user DMs: run: openclaw config set session.dmScope "per-channel-peer" (or  
"per-account-channel-peer" for multi-account channels) to isolate sessions.  
Docs: channels/pairing  
  
Telegram DM policy  
● Pairing (recommended)  
Updated config: ~/.openclaw/openclaw.json  
Backup: ~/.openclaw/openclaw.json.bak  
Configuration updated.  
one@DS104: ~$
```

(./images/a31.png)

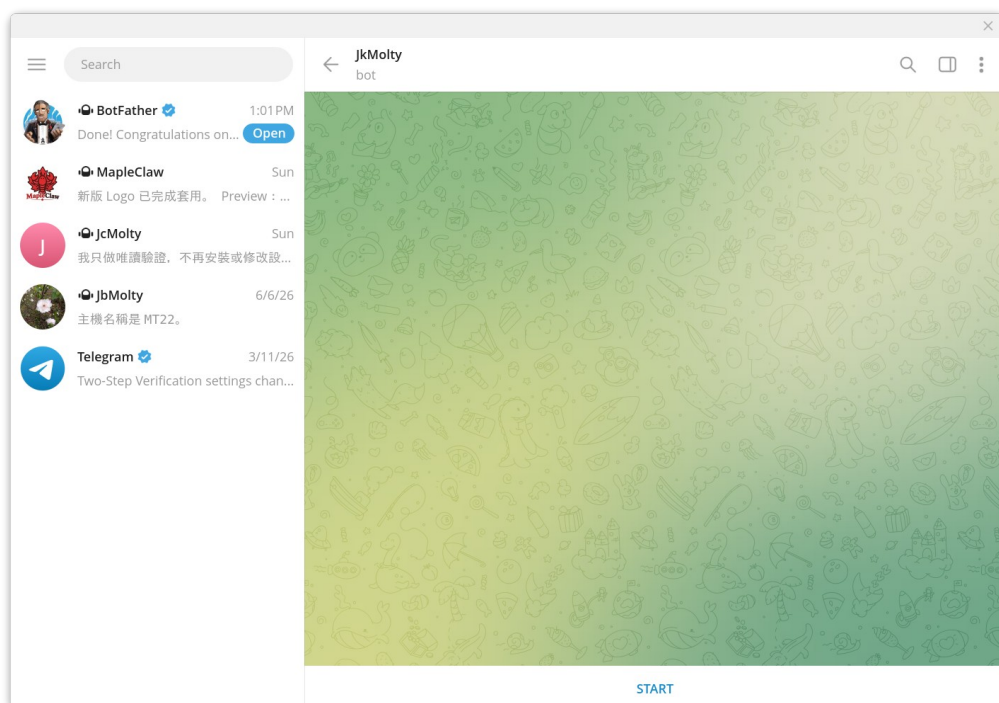
結束終端機上 Channels 選取設定。

回到 Telegram 應用程式視窗，



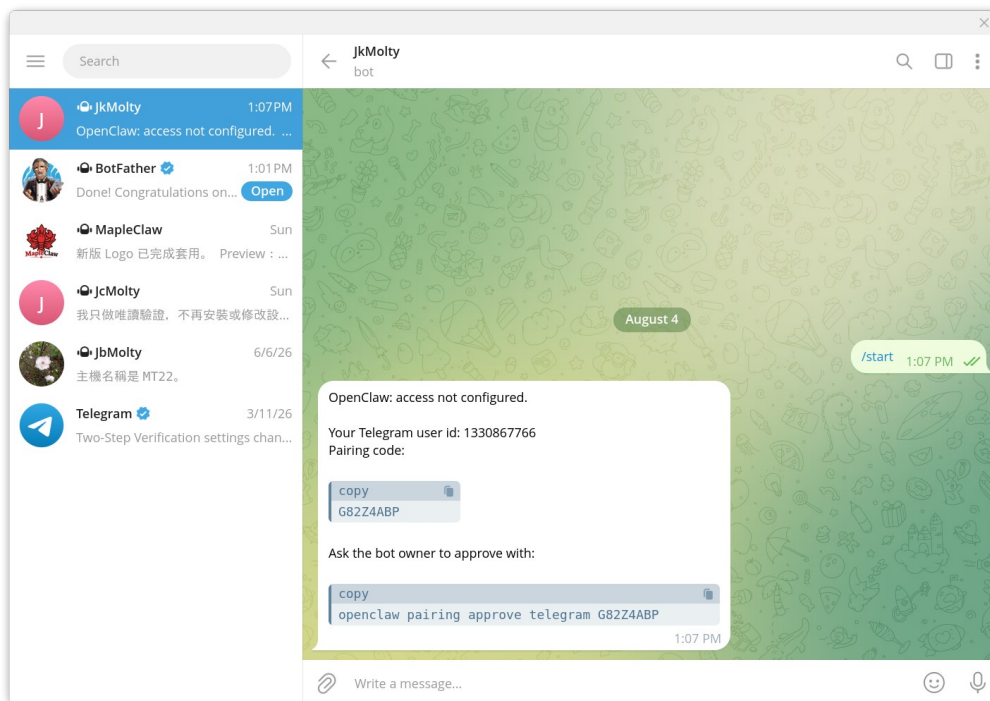
(./images/a26.png)

點擊 t.me/jkmolty_bot, 進入 JkMolty 啟動視窗：



(./images/a32.png)

點擊最下面的 start，



(./images/a33.png)

複製

openclaw pairing approve telegram XXXXX

```

one@DS104: ~
Telegram — simplest way to get started — register a bot with @BotFather and get going.
https://docs.openclaw.ai/channels/telegram https://openclaw.ai

Configure DM access policies now? (default: pairing)
Yes

Telegram DM access
Default: pairing (unknown DMs get a pairing code).
Approve: openclaw pairing approve telegram <code>
Allowlist DMs: channels.telegram.dmPolicy="allowlist" + channels.telegram.allowFrom
entries.
Public DMs: channels.telegram.dmPolicy="open" + channels.telegram.allowFrom includes
"*"
Multi-user DMs: run: openclaw config set session.dmScope "per-channel-peer" (or
"per-account-channel-peer" for multi-account channels) to isolate sessions.
Docs: channels/pairing

Telegram DM policy
Pairing (recommended)
Updated config: ~/.openclaw/openclaw.json
Backup: ~/.openclaw/openclaw.json.bak
Configuration updated.

one@DS104:~$ openclaw pairing approve telegram G82Z4ABP

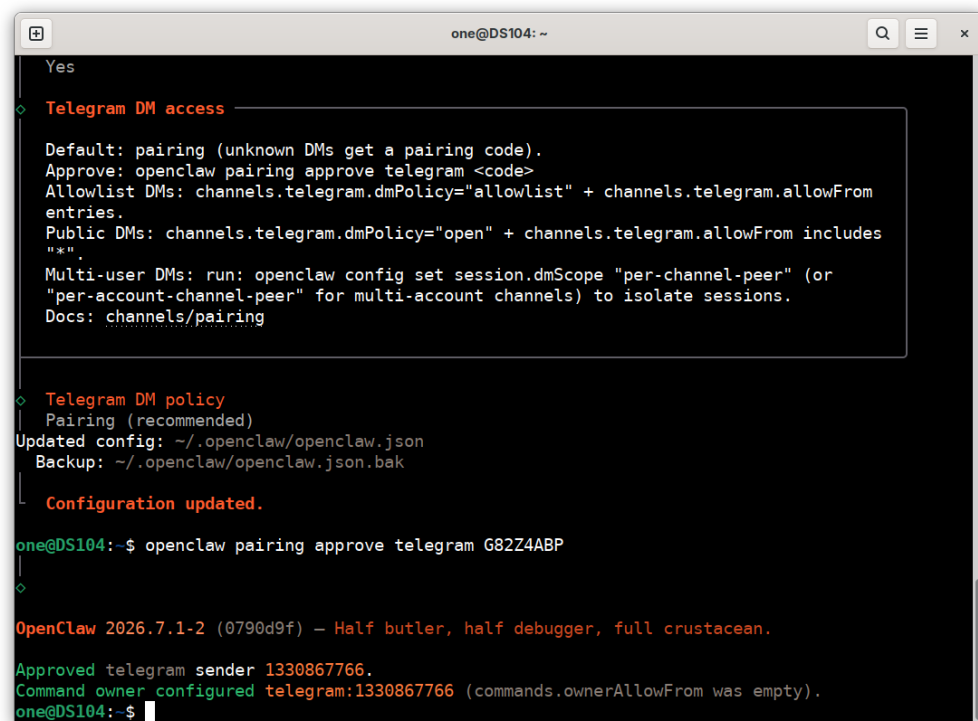
```

(./images/a34.png)

貼入終端機裏

\$ openclaw pairing approve telegram XXXXX

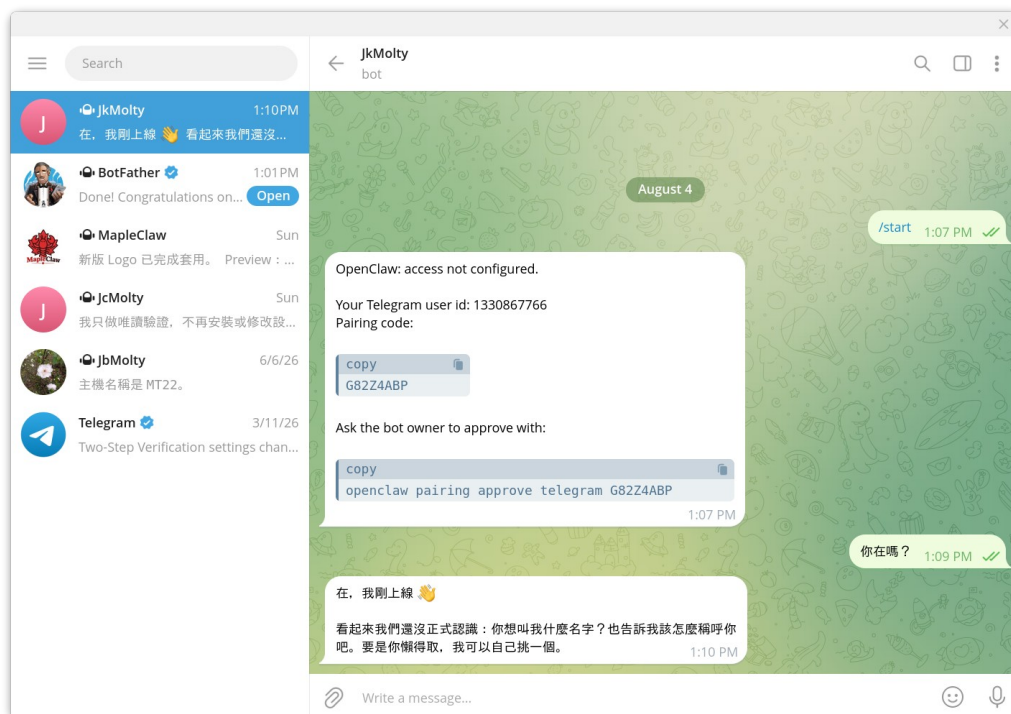
完成安全配對，



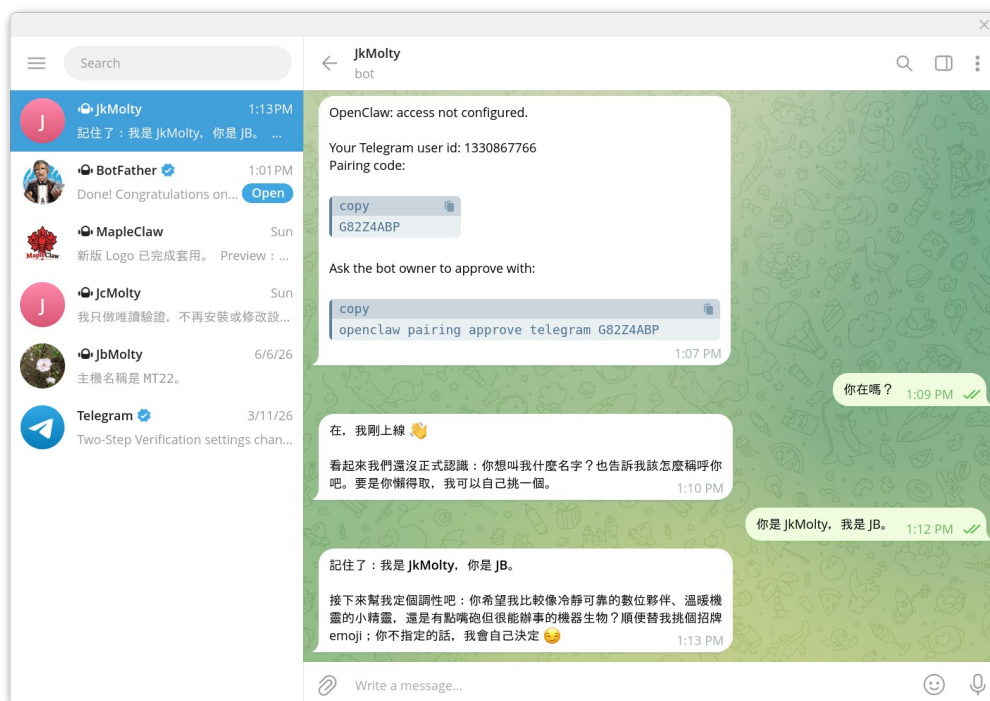
```
one@DS104: ~  
Yes  
Telegram DM access  
Default: pairing (unknown DMs get a pairing code).  
Approve: openclaw pairing approve telegram <code>  
Allowlist DMs: channels.telegram.dmPolicy="allowlist" + channels.telegram.allowFrom  
entries.  
Public DMs: channels.telegram.dmPolicy="open" + channels.telegram.allowFrom includes  
"*"  
Multi-user DMs: run: openclaw config set session.dmScope "per-channel-peer" (or  
"per-account-channel-peer" for multi-account channels) to isolate sessions.  
Docs: channels/pairing  
Telegram DM policy  
Pairing (recommended)  
Updated config: ~/.openclaw/openclaw.json  
Backup: ~/.openclaw/openclaw.json.bak  
Configuration updated.  
one@DS104:~$ openclaw pairing approve telegram G82Z4ABP  
OpenClaw 2026.7.1-2 (0790d9f) - Half butler, half debugger, full crustacean.  
Approved telegram sender 1330867766.  
Command owner configured telegram:1330867766 (commands.ownerAllowFrom was empty).  
one@DS104:~$
```

(./images/a35.png)

回到 Telegram 視窗，開始和小龍蝦對話，

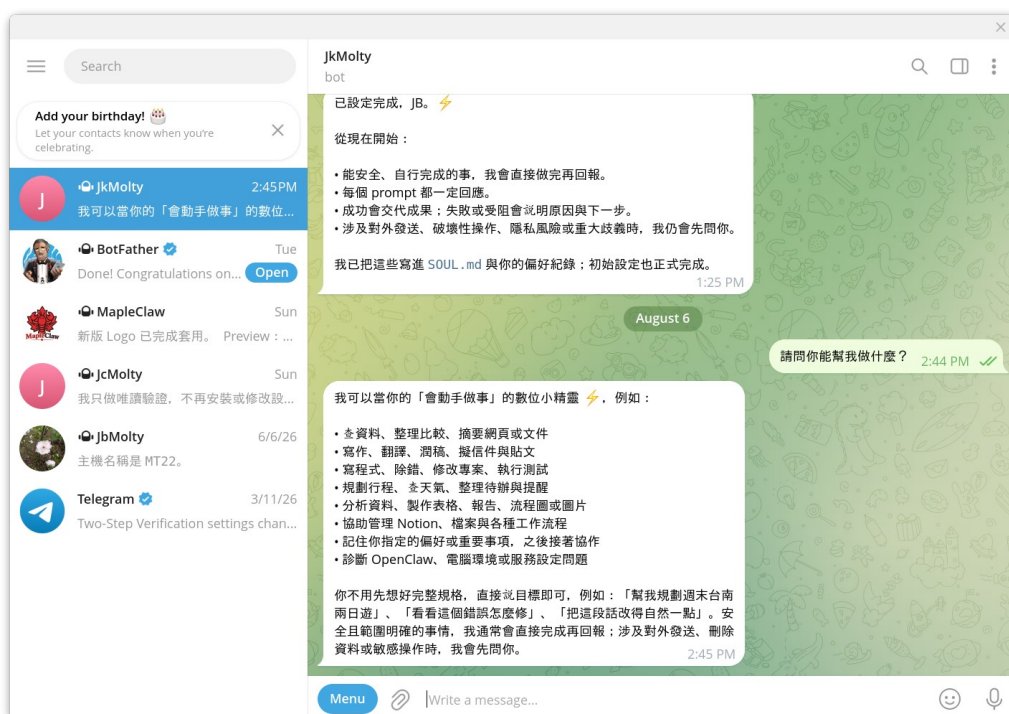


(./images/a36.png)



(./images/a39.png)

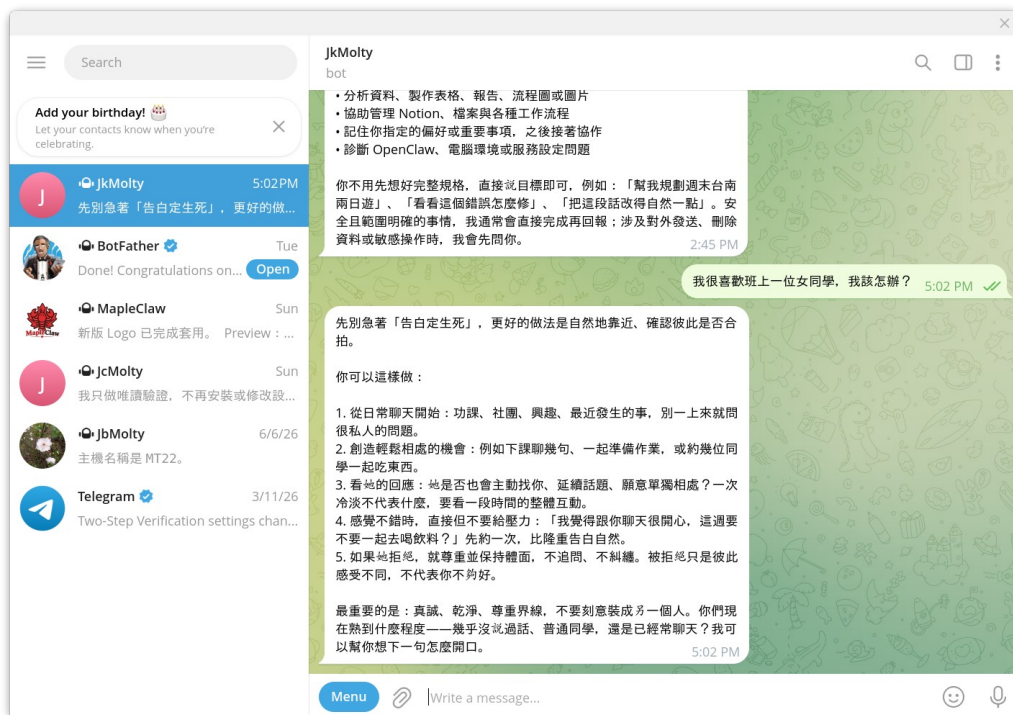
小龍蝦可以做什麼？問它啊。。。



(./images/a40.png)

或者把你想做的事情問它：

我很喜歡班上一位女同學，我該怎辦？



(./images/a41.png)

加油囉，小龍蝦可以幫你心想事成。

The End。