

ClawSheet

The OpenClaw Quick Start Cheat Sheet

From zero to your first AI conversation in 30 minutes.
No experience needed.

v2026 Edition

What's Inside

- Prerequisites checklist & 3 install methods
- &j Step-by-step setup wizard walkthrough
- Ø=Ü Connect Telegram, WhatsApp, or Discord
- Ø=Ý Essential CLI commands (the 20 you actually need)
- Ø=Þ Security hardening (don't skip this)
- Ø=Ý Troubleshooting the 10 most common issues

Not affiliated with the OpenClaw project. Created for beginners who just want it to work.

Contents

- 01 What is OpenClaw?**
The 30-second version
- 02 Before You Start**
Prerequisites checklist
- 03 Installation**
Three ways to install (pick one)
- 04 The Setup Wizard**
Guided configuration walkthrough
- 05 Connect a Chat Channel**
Telegram, WhatsApp, or Discord
- 06 Your First Conversation**
Testing that everything works
- 07 Essential Commands**
The 20 CLI commands you'll actually use
- 08 Customize Your Agent**
Personality, memory, & workspace
- 09 Models & Cost Control**
Choose your AI and manage spend
- 10 Security Hardening**
Must-do steps before going further
- 11 Troubleshooting**
Fixes for the 10 most common issues
- 12 What's Next?**
Skills, automation, and advanced setup

How to use this guide:

Follow sections 1-6 in order for your first setup (~30 min). Then keep sections 7-12 as a reference.

1 🦞 What is OpenClaw?

OpenClaw is an open-source personal AI assistant that runs on your own machine. Unlike ChatGPT or other browser-based AI tools, OpenClaw becomes an always-on agent that can actually DO things for you — search the web, manage your calendar, send messages, control your browser, run shell commands, and automate workflows.

You chat with it through the messaging apps you already use: Telegram, WhatsApp, Discord, Slack, Signal, or a built-in web dashboard. It feels like texting a very capable friend.

Why people love it

- It runs locally — your data never leaves your machine unless you choose to share it
- It connects to your existing messaging apps (text your AI from anywhere)
- It can take real actions: send emails, book meetings, search the web, write code
- It's open source — free to use, huge community, constantly improving
- It remembers context across conversations (long-term memory)

Formerly known as...

OpenClaw was previously called Clawdbot (or MoltBot). If you see those names online, it's the same project. The "lobster" branding is intentional — 🦞 is the unofficial mascot.

🦞 Tip Run on the computer you already use. You do NOT need a Mac Mini or a VPS. Move to a dedicated device later if you want it running 24/7.

2 Before You Start

Gather these three things before you begin. The whole install takes about 10 minutes if everything's ready.

Checklist

& Node.js v22+

Required. OpenClaw is a Node.js application. Check with: `node --version`

& An AI API Key

You need at least one: Anthropic (recommended), OpenAI, or a free option like OpenRouter/Qwen

& A Chat Platform (optional)

Telegram is easiest for beginners. WhatsApp and Discord also work. Or just use the web dashboard.

Installing Node.js (if you don't have it)

```
# Check your current version
$ node --version

# If missing or below v22, install via nvm (recommended):
$ curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.0/install.sh | bash
$ nvm install 22
$ nvm use 22
```

bash

🚫 Pro Tip: Windows users: Install WSL2 first (Windows Subsystem for Linux). OpenClaw runs much better on WSL2 than native Windows. Search "Install WSL" in your Start menu.

Getting an AI API Key

OpenClaw needs to connect to an AI model to "think." Here are your options:

Anthropic (Claude)

Best quality. Use OAuth (free to sign in) or buy API credits at console.anthropic.com

+P Recommended

OpenAI (GPT)

Also great. Get key at platform.openai.com/api-keys

Good option

OpenRouter

Access to 100+ models, some free tiers. openrouter.ai

Budget-friendly

Ollama (local)

Run AI completely offline on your machine. No API costs.

Free / private

& ⚠ Warning! API costs can add up! Most casual users spend \$5-\$20/month. Use the cost calculator in section 9, or start with free models on OpenRouter to test first.

3 Ø=Üæ Installation

Pick ONE method below. The one-liner is recommended for most people.

Method 1: One-Liner (Recommended)

Automatically detects your system, installs Node.js if needed, and sets everything up:

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

bash

Method 2: NPM Install

If you already have Node.js 22+ and prefer package managers:

```
$ npm install -g openclaw@latest  
# or  
$ pnpm add -g openclaw@latest
```

bash

Method 3: Ollama (Easiest if you use Ollama)

If you already have Ollama installed, this is the fastest path:

```
$ ollama launch openclaw
```

bash

Ollama will detect if OpenClaw is missing and install it automatically. Models are included.

Verify Installation

```
$ openclaw --version  
# Should print something like: openclaw v2026.x.x
```

bash

Ø=Ü Not working? Close and reopen your terminal after installation, then try again. The PATH may not have updated yet.

Ø=Ü; Pro Tip: Make sure git is installed before running the installer. Some users report issues without it. Check with: `git --version`

4 & The Setup Wizard

The onboarding wizard walks you through everything. This is the most important step.

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

bash

The `--install-daemon` flag makes OpenClaw run as a background service that survives reboots. Here's what the wizard asks:

- 1 Gateway Setup — Choose "Local" (default). This runs the AI gateway on your machine.
- 2 AI Authentication — Enter your API key or choose OAuth sign-in. Anthropic OAuth is recommended — it's the simplest. The wizard will test the connection.
- 3 Chat Channels — Choose which messaging apps to connect. You can skip this and add them later. Telegram is easiest for beginners.
- 4 Skills — Optional add-ons (web search, file management, etc.). Choose "Skip for now" if unsure — you can install skills later.
- 5 Node Manager — Choose "npm" when asked (it's the default).
- 6 Hatch Your Bot — The wizard generates your bot's identity. Give it a name if you want! This creates the personality file.

ProTip: wizard recommends the TUI (terminal interface) for first-time setup. Use it! The web-based Control UI is better once you're familiar with the basics.

Warning: If the wizard hangs on "testing connection," your API key may be wrong or your network is blocking outbound requests. Double-check the key and try again.

After the Wizard

If everything worked, you should see a success message with a URL for the dashboard. Verify:

```
# Check gateway is running
$ openclaw gateway status

# Open the web dashboard
$ openclaw dashboard

# Opens http://127.0.0.1:18789/ in your browser
```

bash

ProTip: If the gateway isn't running, start it with: `openclaw gateway start`

5 🚪 Connect a Chat Channel

This is where OpenClaw gets fun — you'll be able to text your AI from your phone. Pick one platform to start.

Option A: Telegram (Recommended for Beginners)

- 1 Open Telegram and search for @BotFather
- 2 Send /newbot and follow the prompts to create your bot
- 3 Copy the bot token BotFather gives you
- 4 In your terminal: `openclaw channels add telegram`
- 5 Paste the token when prompted
- 6 Open a chat with your bot in Telegram and send "Hello"
- 7 You'll get a pairing code — approve it (see below)

```
# Approve the pairing request
$ openclaw pairing list telegram
$ openclaw pairing approve telegram <code>
```

bash

Option B: WhatsApp

- 1 Run: `openclaw channels login` (in terminal)
- 2 A QR code appears in your terminal
- 3 On your phone: WhatsApp > Settings > Linked Devices > Link a Device
- 4 Scan the QR code
- 5 Send a message from another number (or use WhatsApp Web) to test

🚪 Note: WhatsApp requires the phone to stay connected. VoIP numbers are not recommended. Use your real phone number.

Option C: Discord

- 1 Go to discord.com/developers/applications and create a new application
- 2 Under "Bot," create a bot and copy the token
- 3 Run: `openclaw channels add discord`
- 4 Paste the bot token when prompted
- 5 Invite the bot to your server using the OAuth2 URL generator

Option D: Web Dashboard Only

Don't want to connect any chat app? No problem. The built-in dashboard works great:


```
$ openclaw dashboard
# Chat directly at http://127.0.0.1:18789/
```

bash

& ⚠ Warning Never expose the dashboard to the public internet! It has full access to your machine. Keep it on localhost (127.0.0.1) only.

6 🎉 Your First Conversation

If you've followed sections 1-5, your AI assistant is running. Let's make sure it works.

Test via Dashboard

```
$ openclaw dashboard
```

bash

Open <http://127.0.0.1:18789/> in your browser and type: "Hello! What can you do?"

Test via Messaging App

Open your connected chat app (Telegram, WhatsApp, etc.) and send your bot a message. Try these:

"What's the weather in London?"	! Tests web search (if enabled)
"Summarize this article: [paste URL]"	! Tests reading comprehension
"What time is it?"	! Tests basic tool use
"Remember that my favorite color is blue"	! Tests memory
"What's my favorite color?"	! Tests memory recall

Not getting responses?

- 1 Check the gateway: `openclaw gateway status` (must be "running")
- 2 Check logs: `openclaw logs --follow` (look for errors)
- 3 Run diagnostics: `openclaw doctor --deep --yes` (auto-fixes common issues)
- 4 Verify your API key: `openclaw models status --probe`
- 5 If using Telegram/WhatsApp: did you approve the pairing code?

🎉 **Congratulations!** If you got a response, you're done with the basic setup! The rest of this guide is reference material you'll come back to.

7 Ø=Ý' Essential Commands

You don't need to memorize 150+ commands. Here are the ~20 you'll use 90% of the time:

Setup & Health

Command	What It Does
<code>openclaw onboard</code>	Run the setup wizard again (re-configure)
<code>openclaw doctor</code>	Health check — finds and reports issues
<code>openclaw doctor --deep --yes</code>	Deep scan + auto-fix. Run this when things break.
<code>openclaw status --all</code>	Complete debug report (share this for support)
<code>openclaw --version</code>	Check installed version
<code>openclaw update</code>	Update to the latest version

Gateway (the engine)

Command	What It Does
<code>openclaw gateway status</code>	Is the gateway running?
<code>openclaw gateway start</code>	Start the background service
<code>openclaw gateway restart</code>	Restart (fixes most issues)
<code>openclaw gateway stop</code>	Stop the service
<code>openclaw dashboard</code>	Open the web dashboard UI

Channels & Chat

Command	What It Does
<code>openclaw channels list</code>	Show all connected platforms
<code>openclaw channels login</code>	Connect a new platform (QR code flow)
<code>openclaw channels add <type></code>	Add Telegram/Discord/etc.
<code>openclaw pairing list <channel></code>	Show pending pairing requests
<code>openclaw pairing approve <ch> <code></code>	Approve a chat contact

Models & Config

Command	What It Does
<code>openclaw models list --all</code>	Show available AI models
<code>openclaw models set <model></code>	Change primary AI model
<code>openclaw models status --probe</code>	Test API key / connection
<code>openclaw config get <key></code>	Read a config value
<code>openclaw config set <key> <val></code>	Write a config value

Pro Tip In doubt, run: `openclaw doctor --deep --yes` — it fixes most common issues automatically.

8 Customize Your Agent

Your agent has a personality, long-term memory, and a workspace. All of it is configurable.

Key Files & Directories

Command	What It Does
<code>~/ .openclaw/</code>	Home directory — all config, credentials, logs
<code>~/ .openclaw/workspace/</code>	Agent workspace — memory & personality files
<code>~/ .openclaw/workspace/SYSTEM.md</code>	Agent personality — tone, values, rules
<code>~/ .openclaw/credentials/</code>	API keys and OAuth tokens (keep private!)
<code>~/ .openclaw/agents/<id>/</code>	Per-agent configuration

Editing Your Agent's Personality

The SYSTEM.md file is your agent's "character sheet." It defines how the agent behaves, responds, and what it knows about you. Open it and make it yours:

```
# Open the personality file
$ nano ~/ .openclaw/workspace/SYSTEM.md

# Example content:
# You are Alex, a helpful AI assistant.
# You speak casually and use humor.
# You know I work in marketing and prefer
# concise answers with bullet points.
```

bash

Memory System

OpenClaw remembers things across conversations. You can manage this memory:

```
# Reindex memory (after manual file edits)
$ openclaw memory reindex

# Search the agent's memory
$ openclaw memory search "project deadline"
```

bash

You can also tell your agent to remember things naturally in chat: "Remember that my project deadline is March 15th."

Pro Tip Keep your SYSTEM.md concise. The longer it is, the more tokens it uses every message. 20-30 lines is a sweet spot.

9 Ø=Ü° Models & Cost Control

OpenClaw supports multiple AI providers. Your choice affects quality, speed, and cost.

Popular Model Choices

Model	Provider	Notes	Cost
Claude Opus 4.5	Anthropic	Best quality, highest cost	\$\$\$
Claude Sonnet 4	Anthropic	Great balance of quality & cost	\$\$
GPT-4.1	OpenAI	Strong general-purpose model	\$\$
Claude Haiku 4	Anthropic	Fast & cheap, good for simple tasks	\$
Qwen (free tier)	OpenRouter	Free! Lower quality but zero cost	Free
Ollama (local)	Your machine	No API cost, runs offline	Free

Setting Your Model

```
# Set your primary model
$ openclaw models set claude-sonnet-4

# Add a cheaper fallback model
$ openclaw models fallbacks add claude-haiku-4

# Check what you're currently using
$ openclaw models status --probe
```

bash

Controlling Costs

- Start with cheaper models (Haiku or free OpenRouter models) to test your setup
- Use fallback chains: primary expensive model !' cheaper fallback when rate-limited
- Casual users spend \$5-20/month. Heavy users can spend \$50+/month.
- Check usage regularly at your provider's dashboard (console.anthropic.com, platform.openai.com)
- Ollama is completely free but requires a decent GPU (8GB+ VRAM recommended)

& p Warning Error stories exist of users getting \$100+ bills from leaving OpenClaw running with an expensive model. Set spending limits on your AI provider dashboard!

10 🛡️ Security Hardening

OpenClaw has deep access to your machine. Do NOT skip these steps.

🚨 CRITICAL: Security is not optional

OpenClaw can run shell commands, access files, and browse the web. Unsecured gateways have been found on Shodan. Credentials are stored unencrypted. Take this seriously.

Must-Do Security Steps

- 1 Run the security audit: `openclaw security audit --deep`
- 2 Bind to localhost only — in your config, set the gateway host to `127.0.0.1` (not `0.0.0.0`)
- 3 Enable `exec_approval` — requires your OK before the agent runs terminal commands, deletes files, or pushes to git
- 4 Vet community skills — install only from trusted sources. Use Cisco Skill Scanner if available.
- 5 Set up allowlisting — only approved contacts can talk to your bot
- 6 Give the agent its own credentials — don't use your personal API keys for accounts you care about

Security Commands

```
# Run full security audit
$ openclaw security audit --deep

# Bind gateway to localhost only
$ openclaw config set gateway.host "127.0.0.1"

# Enable tool approval (recommended!)
$ openclaw config set agents.defaults.exec_approval true

# Check secrets and credentials
$ openclaw secrets
```

bash

👉 **ProTip** golden rule: run OpenClaw on a dedicated machine or VM if possible. If it has to share your daily-driver, enable `exec_approval` so it asks before doing anything destructive.

11 🚫 Troubleshooting

The 10 most common issues and how to fix them. Run `openclaw doctor --deep --yes` first — it auto-fixes many of these.

- 1 "command not found: openclaw"**
Fix: Node.js not in PATH. Close and reopen terminal. If still broken: `nvm install 22 && npm install -g openclaw@latest`
- 2 "RPC probe: failed"**
Fix: Port 18789 is blocked. Find the conflict: `sudo lsof -i :18789` and kill the process.
- 3 Gateway not responding**
Fix: The daemon stopped. Fix: `openclaw gateway restart`. Check logs: `openclaw logs --follow`
- 4 "0 tokens used" or no response**
Fix: API auth failed. Verify: `openclaw models status --probe`. Re-enter key if needed.
- 5 Token/plan limit errors**
Fix: Premium models (Opus, GPT-5) need prepaid API credit, not just a subscription. Add credits at your provider's billing page.
- 6 WhatsApp QR code won't scan**
Fix: Try a fresh session: `openclaw channels remove whatsapp` then `openclaw channels login` again.
- 7 Telegram bot ignores messages**
Fix: Did you approve the pairing? Check: `openclaw pairing list telegram` then approve the code.
- 8 High memory usage**
Fix: Common on low-RAM systems. Add a 4GB swap file. Or use a lighter model.
- 9 Skills won't install**
Fix: Check npm permissions: `npm config set prefix ~/.npm-global` and add it to PATH.
- 10 "Access not configured"**
Fix: Your user ID isn't allowlisted. Run: `openclaw security audit` to see the config.

🚫 **Still stuck?** `openclaw status --all` and share the output in the OpenClaw Discord or GitHub Issues. The community is very active and helpful.

12 What's Next?

You've got a working OpenClaw setup. Here's how to level up.

Install Skills

Skills are plugins that give your agent new abilities: web search, GitHub integration, file management, coding assistance, and more.

```
# List available skills
$ openclaw skills list

# Install a skill
$ openclaw skills install @openclaw/skill-web-search
$ openclaw skills install @openclaw/skill-git

# Check installed skills
$ openclaw skills info
```

bash

Set Up Automation (Cron Jobs)

Have your agent do things on a schedule — daily briefings, news summaries, reminders:

```
# Example: daily morning briefing at 8am
$ openclaw cron add "0 8 * * *" "Send me a morning briefing"

# List scheduled tasks
$ openclaw cron list
```

bash

Browser Automation

OpenClaw can control a browser to fill forms, scrape pages, and interact with websites:

```
# Enable browser skill
$ openclaw skills install @openclaw/skill-browser

# Then in chat: "Go to example.com and take a screenshot"
```

bash

Multi-Agent Setup

Advanced users can run multiple agents with different specializations (a coder, a researcher, a writer):

```
$ openclaw agents create coding-assistant --temperature 0.2
$ openclaw agents create researcher --temperature 0.7
$ openclaw agents list
```

bash

Dedicated Device (24/7 Operation)

- Any old MacBook, Mac Mini, or Linux box works great

- On macOS: install Amphetamine (free) to prevent sleep
- Give the bot its own user account and credentials
- Consider a VPS (DigitalOcean, Hetzner) for cloud hosting
- Use the --install-daemon flag to survive reboots

🎉 You did it!

You now have your own personal AI assistant. Experiment, break things, and have fun.
Questions? Visit github.com/openclaw/openclaw or the OpenClaw Discord community.

Quick Reference Card

Print this page and keep it next to your terminal

🩺 Health & Diagnostics

```
openclaw doctor --deep --yes
Fix everything

openclaw status --all
Full debug report

openclaw logs --follow
Live log stream

openclaw --version
Check version

openclaw update
Update OpenClaw
```

🚪 Gateway

```
openclaw gateway status
Is it running?

openclaw gateway restart
Restart service

openclaw dashboard
Open web UI
```

📶 Channels

```
openclaw channels list
Connected apps

openclaw channels login
Connect WhatsApp

openclaw pairing approve <ch> <code>
Allow contact
```

🧠 Models

```
openclaw models list --all
Available models

openclaw models set <model>
Switch model

openclaw models status --probe
Test connection
```

🧑 Skills

```
openclaw skills list
Available skills

openclaw skills install <name>
Add a skill
```

🧠 Memory & Config

```
openclaw memory reindex
Rebuild memory

openclaw memory search "query"
Search memory

openclaw config get <key>
Read config

openclaw config set <key> <val>
Write config
```

🔒 Security

```
openclaw security audit --deep
Security scan

openclaw secrets
Check credentials
```

📁 Key Paths

~/.openclaw/	Home directory (all config & data)
~/.openclaw/workspace/	Agent workspace (memory & personality)
~/.openclaw/workspace/SYSTEM.md	Agent personality file
~/.openclaw/credentials/	API keys & tokens (keep private!)

🛑 Emergency Stop

```
openclaw gateway stop # Stop the AI agent immediately
```