HOW TO SET UP YOUR OWN AI AGENT SERVER

A Complete Step-by-Step Guide
by Shea Johnson | www.mssheajohnson.com
==========

This guide walks you through setting up your own
always-on AI server using a Mac Mini.

No coding experience required.
If you can copy and paste, you can do this.

Throughout this guide, you'll see these labels:

READ — Information to understand before moving on
DO — An action to take (click something, open something)
COPY — A command to copy and paste into Terminal
CHECK — A command to verify the install worked
EXPECT — What you should see after running a check

If a CHECK doesn't show what you EXPECT,
re-run the COPY command for that step.

==========
BEFORE YOU START — WHAT YOU'LL NEED
==========

HARDWARE:

Mac Mini M4 (16GB RAM, 256GB storage) — ~$530
Bluetooth keyboard and mouse — for initial setup only
HDMI cable — to connect to a TV or monitor
HDMI dummy plug — ~$10 on Amazon or Micro Center
External hard drive — optional, for training documents

ACCOUNTS TO CREATE BEFORE YOU BEGIN:

Anthropic API account — console.anthropic.com
(This is pay-as-you-go. You'll create API keys here.)

Gmail account for your bot — gmail.com

(Separate from your personal email.)

Tailscale account — tailscale.com
(Free. This lets you access your server from anywhere.)

Telegram account — telegram.org
(Free. This is how you'll chat with your bot.)

Google Cloud account — console.cloud.google.com
(For connecting Google Workspace apps.)

Apple ID for your server — appleid.apple.com
(Optional. Only if you want iCloud app integration.
DO NOT use your personal Apple ID.)

API KEYS TO CREATE ON console.anthropic.com:

READ: Create a separate API key for each platform.
Name each key after what it connects to.
Example: "Claude-Code-Server" and "OpenClaw"

You will also need:
Gemini API key — aistudio.google.com (free tier)
ElevenLabs API key — elevenlabs.io (optional, for text-to-speech)

==========
PHASE 1 — macOS SETUP
==========

READ: Plug the Mac Mini into your TV via HDMI.
Connect your keyboard and mouse. Power it on.
You'll click through several setup screens.

Screen 1: Language

DO: Select English
DO: Select United States
DO: Click Continue

Screen 2: Accessibility

DO: Click "Not Now" or Continue (skip this)

Screen 3: Wi-Fi

DO: Select your Wi-Fi network
DO: Enter your Wi-Fi password
DO: Click Continue

Screen 4: Data & Privacy

DO: Click Continue

Screen 5: Migration Assistant

DO: Select "Not now"
DO: Click Continue

Screen 6: Apple ID / iCloud — IMPORTANT

READ: Do NOT sign in with your personal Apple ID.
Do NOT create a new Apple ID right now.
You can add one later for iCloud integration.

DO: Look for "Set Up Later" or "Skip" at the bottom
DO: If it asks "Are you sure?" — click Skip again

Screen 7: Terms and Conditions

DO: Click Agree

Screen 8: Create Your Computer Account — IMPORTANT

READ: This is your admin login for the server.
Choose a name and password you'll remember.

DO: Enter a Full Name (example: Agent Server)
DO: Enter an Account Name (lowercase, no spaces)
DO: Enter a strong password

*** WRITE THIS PASSWORD DOWN. KEEP IT SAFE. ***

Screen 9: Location Services

DO: Uncheck / Disable location services
DO: Click Continue

Screen 10: Analytics

DO: Uncheck "Share Mac Analytics with Apple"
DO: Uncheck "Share crash data with app developers"
DO: Click Continue

Screen 11: Screen Time

DO: Click "Set Up Later"

Screen 12: Siri

DO: Click "Set Up Later" or Disable

Screen 13: Appearance

DO: Pick Light or Dark mode (your preference)
DO: Click Continue

Screen 14: Desktop

READ: You should see the macOS desktop.
Congratulations — the operating system is set up.

==========
PHASE 1 (continued) — ESSENTIAL SETTINGS
==========

READ: These settings make your Mac Mini work as a
server. Do all of them before moving on.

Prevent Sleeping

DO: Click the Apple menu (top left corner)
DO: Click System Settings
DO: Click Energy
DO: Set "Turn display off after" to NEVER
DO: Turn ON "Prevent automatic sleeping
when display is off"
DO: Turn ON "Wake for network access"

Enable Screen Sharing

DO: Go to System Settings
DO: Click General
DO: Click Sharing
DO: Turn ON "Screen Sharing"
DO: Write down the address it shows you

Enable Remote Login (SSH)

DO: Still in System Settings > General > Sharing
DO: Turn ON "Remote Login"

Set Computer Name

DO: Go to System Settings > General > About
DO: Click the name at the top
DO: Change it to something recognizable
(example: Agent-Server)

Enable Auto-Login

DO: Go to System Settings > Users & Groups
DO: Set "Automatic Login" to your admin account

Auto-Start After Power Failure

DO: Go to System Settings > Energy
DO: Turn ON "Start up automatically after
a power failure"

==========
PHASE 2 — SOFTWARE INSTALLATION
==========

READ: This is where you install all the tools your
server needs. You'll be working in the Terminal app.

TIP: How to Copy and Paste Commands Easily

READ: If you're reading this on your phone and the
Mac Mini only has a keyboard, typing long commands
is painful. Here's the trick:

DO: Open Safari on the Mac Mini
DO: Sign into your bot's Gmail
DO: On your phone, email yourself the commands
DO: Open the email on the Mac Mini
DO: Copy and paste from the email into Terminal

Open Terminal

DO: Press Command + Space (opens Spotlight)
DO: Type: Terminal
DO: Press Enter

READ: A black/white window opens. This is Terminal.
This is where you paste all the commands below.

==========
STEP 1: Xcode Command Line Tools
==========

READ: This is Apple's developer foundation.
Everything else needs this installed first.

COPY AND PASTE INTO TERMINAL:

```
xcode-select --install
```

DO: A popup appears — click "Install"
DO: Wait 5-10 minutes for it to finish

CHECK — PASTE THIS TO VERIFY:

```
xcode-select --version
```

EXPECT: A version number (like "xcode-select version 2416")

==========
STEP 2: Homebrew
==========

READ: Homebrew is like an app store for Terminal.
Instead of downloading from websites, you just type
"brew install [thing]" and it handles everything.

COPY AND PASTE INTO TERMINAL (this is one long line):

```
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

DO: When it asks for your password, type it
(characters won't show — that's normal)
DO: Wait for it to finish

READ: When it's done, it shows TWO COMMANDS you
need to run. They look something like this:

```
(echo; echo 'eval "$(/opt/homebrew/bin/brew shellenv)"') >> ~/.zprofile
eval "$(/opt/homebrew/bin/brew shellenv)"
```

DO: Copy and paste BOTH of those commands
(the ones YOUR Terminal shows you, not these)

CHECK — PASTE THIS TO VERIFY:

```
brew --version
```

EXPECT: Something like "Homebrew 5.0.16"

==========
STEP 3: Node.js
==========

READ: Node.js is the programming language that
Claude Code and OpenClaw need to run.

COPY AND PASTE INTO TERMINAL:

```
brew install node
```

DO: Wait for it to finish

CHECK — PASTE THIS TO VERIFY:

```
node --version
```

EXPECT: v20 or higher (like v25.8.0)

ALSO CHECK:

```
npm --version
```

EXPECT: A number (like 11.11.0)

==========
STEP 4: Python
==========

READ: Python is the programming language your bots
use. Your Mac may have an old version — this
installs the latest.

COPY AND PASTE INTO TERMINAL:

brew install python

DO: Wait for it to finish

CHECK — PASTE THIS TO VERIFY:

python3 --version

EXPECT: Python 3.12 or higher

ALSO CHECK:

pip3 --version

EXPECT: pip with the matching Python version

READ: If pip3 shows an older Python (like 3.9),
close Terminal completely (Command+Q), reopen it,
and check again. The restart fixes the path.

==========
STEP 5: Git
==========

READ: Git downloads code from the internet.
OpenClaw needs it to install.

COPY AND PASTE INTO TERMINAL:

brew install git

CHECK — PASTE THIS TO VERIFY:

git --version

EXPECT: git version 2.x.x

==========
STEP 6: Claude Code — THE BIG ONE
==========

READ: Claude Code is an AI developer that lives
inside your Terminal. You type in English,
it writes and runs code for you.

Once this is installed, you stop typing commands.
You start TALKING.

COPY AND PASTE INTO TERMINAL:

```
npm install -g @anthropic-ai/claude-code
```

CHECK — PASTE THIS TO VERIFY:

```
claude --version
```

EXPECT: A version number (like 2.1.72)

==========
STEP 7: Set Your API Key
==========

READ: This connects Claude Code to Anthropic's
servers. Without it, Claude Code can't think.

Use the API key you created on console.anthropic.com
(the one named for your server or for Claude Code).

COPY AND PASTE INTO TERMINAL
(replace YOUR-API-KEY-HERE with your actual key):

```
export ANTHROPIC_API_KEY='YOUR-API-KEY-HERE'
```

THEN COPY AND PASTE THIS
(to save it permanently — replace the key again):

```
echo 'export ANTHROPIC_API_KEY="YOUR-API-KEY-HERE"' >> ~/.zprofile
```

THEN COPY AND PASTE THIS
(to activate it now):

source ~/.zprofile

CHECK — PASTE THIS TO VERIFY:

echo $ANTHROPIC_API_KEY

EXPECT: Your API key prints out (starts with sk-ant-)

READ: If it's blank, you made a typo in the key.
Re-run the export command carefully.

==========
STEP 8: Launch Claude Code
==========

COPY AND PASTE INTO TERMINAL:

claude

EXPECT: Claude Code starts up and greets you.
You can now type in plain English.

DO: Test it by typing: Hello, what can you help me with?
EXPECT: Claude responds intelligently.

DO: Type /exit to leave Claude Code

==========
*** THE HARD PART IS DONE ***
==========

READ: Claude Code is running. From here on out,
you tell Claude Code what to install by TALKING.

Open Claude Code (type "claude" in Terminal)
and say each of the following one at a time.
Wait for each one to finish before the next.

==========
STEPS 9-15: Remaining Tools (via Claude Code)
==========

READ: Open Claude Code first.

COPY AND PASTE INTO TERMINAL:

claude

READ: Now type (or paste) each of these into Claude Code, one at a time:

SAY THIS TO CLAUDE CODE:

Install tmux and verify it's working

READ: tmux keeps your bots running even when you close Terminal.

SAY THIS TO CLAUDE CODE:

Install pm2 globally via npm and verify it's working

READ: pm2 auto-restarts bots if they crash and auto-starts them after power outages.

SAY THIS TO CLAUDE CODE:

Install ffmpeg and verify it's working

READ: ffmpeg processes audio and video files.

SAY THIS TO CLAUDE CODE:

Install imagemagick and verify it's working

READ: imagemagick processes and resizes images.

SAY THIS TO CLAUDE CODE:

Install jq and verify it's working

READ: jq reads data from API responses.

SAY THIS TO CLAUDE CODE:

Install wget and verify it's working

READ: wget downloads files from the internet.

SAY THIS TO CLAUDE CODE:

Install htop and verify it's working

READ: htop shows you what's happening on your
server in real time.

==========
STEP 16: Tailscale
==========

READ: Tailscale lets you access your server from
anywhere in the world — your phone, your laptop,
a coffee shop, anywhere.

DO: Open Safari on the Mac Mini
DO: Go to tailscale.com/download
DO: Download and install the Mac version

OR SAY THIS TO CLAUDE CODE:

Install Tailscale via Homebrew cask and verify it's working

DO: Open Tailscale from Applications
DO: Sign in (create an account or use Google)
DO: Write down the Tailscale IP it gives you

(starts with 100.)

READ: You also need to install Tailscale on your
laptop and your phone — using the SAME account.
Then all three devices can reach each other
from anywhere.

==========
STEP 17: Rclone
==========

READ: rclone uploads files from your server to
Google Drive without opening a browser.

SAY THIS TO CLAUDE CODE:

Install rclone and verify it's working

==========
STEP 18: Python Packages
==========

READ: These are libraries your bots need.
Say each one to Claude Code, one at a time.

SAY THIS TO CLAUDE CODE:

Install python-telegram-bot via pip3 and verify it's working

READ: This is how your bot hears and speaks
on Telegram. It's your bot's phone.

SAY THIS TO CLAUDE CODE:

Install the anthropic Python package via pip3 and verify it's working

READ: This is how your bot thinks. It connects
to Claude's brain.

SAY THIS TO CLAUDE CODE:

Install the requests Python package via pip3 and verify it's working

READ: This lets your bots talk to websites and APIs.

SAY THIS TO CLAUDE CODE:

Install the schedule Python package via pip3 and verify it's working

READ: This lets your bots do things at specific times.

==========
STEP 19: Nice-to-Haves (Optional)
==========

SAY THIS TO CLAUDE CODE:

Install neofetch and verify it's working

SAY THIS TO CLAUDE CODE:

Install speedtest-cli and verify it's working

==========
FINAL VERIFICATION — THE BIG CHECK
==========

READ: This checks everything at once.
Exit Claude Code first (type /exit).
Then paste this entire block into Terminal:

COPY AND PASTE ALL OF THIS INTO TERMINAL:

```
echo "=== VERIFICATION ==="
echo "Xcode:"; xcode-select --version
echo "---"
echo "Homebrew:"; brew --version
echo "---"
echo "Node:"; node --version
echo "---"
```

```
echo "npm:"; npm --version
echo "---"
echo "Python:"; python3 --version
echo "---"
echo "pip:"; pip3 --version
echo "---"
echo "Git:"; git --version
echo "---"
echo "Claude Code:"; claude --version
echo "---"
echo "tmux:"; tmux -V
echo "---"
echo "pm2:"; pm2 --version
echo "---"
echo "jq:"; jq --version
echo "---"
echo "wget:"; wget --version 2>&1 | head -1
echo "---"
echo "htop:"; htop --version
echo "---"
echo "rclone:"; rclone --version 2>&1 | head -1
echo "---"
echo "API Key set:"; if [ -n "$ANTHROPIC_API_KEY" ]; then echo "YES"; else echo "NO - FIX
THIS"; fi
echo "=== DONE ==="
```

EXPECT: Version numbers for everything.
If anything says "command not found," reinstall it.

==========

PHASE 3 — REMOTE ACCESS

==========

READ: If you installed Tailscale in Step 16 on all three devices (Mac Mini, laptop, phone), you can now access your server from anywhere.

How to Connect from Your Laptop

READ: Open Terminal on your laptop.

COPY AND PASTE (replace with your info):

ssh [your-username]@[your-tailscale-ip]

DO: Enter your server password when prompted
EXPECT: You see the Mac Mini's terminal prompt

How to Screen Share from Your Laptop

DO: Open Finder on your laptop
DO: Click Go (in the menu bar) > Connect to Server

TYPE THIS (replace with your info):

vnc://[your-tailscale-ip]

DO: Enter your server username and password
EXPECT: You see the Mac Mini's desktop in a window

Important Info to Save

READ: Fill in these blanks and save somewhere safe.

Server Admin Username: _____
Server Admin Password: _____
Server Local IP: _____
Server Tailscale IP: 100._____
Screen Sharing: vnc://_____
SSH: ssh _____@_____
Anthropic API Key: _____
Bot Gmail: _____
Telegram Bot Token: _____

What NOT to Install on Your Server

READ: Keep your server clean and lean.

No personal iCloud (use a separate Apple ID)
No Microsoft Office (Claude Code makes documents)

No social media apps (APIs handle posting)
No Telegram desktop app (bot runs through Python)
No personal files (this is a server, not a PC)

==========
PHASE 4 — INSTALLING OPENCLAW
==========

READ: OpenClaw is the platform that manages your AI agent — her skills, her memory, her connections, and her conversations.

Step 1: Install OpenClaw

COPY AND PASTE INTO TERMINAL:

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

CHECK — PASTE THIS TO VERIFY:

```
openclaw --version
```

EXPECT: A version number (like 2026.3.8)

Step 2: Run the Setup Wizard

COPY AND PASTE INTO TERMINAL:

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

READ: The wizard walks you through everything. Here's what to expect at each screen:

SECURITY WARNING:
DO: Read it and accept it.
READ: It's saying this is a personal agent for one person (you). Don't give others access.

MODEL SELECTION:
DO: Select these two models:

Primary: claude-sonnet-4-6
Fallback: claude-haiku-4-5
READ: Do NOT select claude-opus as your default.
It's the most expensive model and your agent pings
every 30 minutes. That will add up fast.

API KEY:
DO: Paste your Anthropic API key when asked
READ: Use the one you created specifically
for OpenClaw.

WEB SEARCH:
DO: Select "Skip for now"

SKILL DEPENDENCIES:
READ: Use arrow keys to move between options.
Press spacebar to check or uncheck each one.

DO: Check these skills:
clawhub — skill store
google-workspace — Gmail, Drive, Calendar, Docs
model-usage — tracks your API spending
nano-banana-pro — AI image generation
nano-pdf — PDF processing
openai-whisper — speech-to-text
summarize — summarize URLs, podcasts, files

READ: You can add more later.
Only install what you need.

NODE MANAGER:
DO: Select npm

HOOKS:
DO: Check session-memory
DO: Check command-logger

HATCH YOUR BOT:
DO: Select "Hatch in TUI"

Step 3: Verify OpenClaw

COPY AND PASTE INTO TERMINAL:

openclaw status

EXPECT:
Gateway: running
Telegram: connected
Session: active

Step 4: Check Your Skills

COPY AND PASTE INTO TERMINAL:

openclaw skills

EXPECT: A table showing your skills as "ready"

Step 5: Open the Dashboard

DO: Open Safari on the Mac Mini

TYPE THIS IN THE ADDRESS BAR:

http://127.0.0.1:18789/

READ: This is your control panel. Bookmark it.

Troubleshooting OpenClaw

IF YOU GET A 401 AUTHENTICATION ERROR:

READ: Your API key didn't save correctly.

DO: Open Claude Code (type "claude" in Terminal)

SAY THIS TO CLAUDE CODE:

Open ~/.openclaw/openclaw.json and verify the

API key and model configuration are correct.
The primary model should be claude-sonnet-4-6
with a fallback of claude-haiku-4-5-20251001.

IF THE MODEL SHOWS AS OPUS INSTEAD OF SONNET:

DO: Press Ctrl+C to exit
DO: Start a new session

IF SKILLS SHOW AS "MISSING":

COPY AND PASTE INTO TERMINAL:

clawhub install [skill-name]

READ: Replace [skill-name] with the actual name.
If rate limited, wait a few minutes and try again.

IF ANYTHING FEELS OFF:

COPY AND PASTE INTO TERMINAL:

openclaw doctor

READ: This checks everything and tells you
what needs fixing.

==========
PHASE 5 — CONNECTING EVERYTHING
==========

Step 1: Set Up Your Telegram Bot

DO: Open Telegram on your phone
DO: Search for @BotFather
DO: Send the message: /newbot
DO: Give it a name when asked
DO: Give it a username (must end in "bot")

READ: BotFather will give you a BOT TOKEN.
Save this — you'll need it.

READ: During the OpenClaw setup wizard, you entered
this token when it asked about Telegram.
If you haven't done this yet, run the wizard again.

DO: Message your bot on Telegram from your phone

READ: It may ask you to pair. Follow the pairing
instructions on screen. Your user ID needs to be
in the allowed senders list.

IF PAIRING DOESN'T WORK:

DO: Open Claude Code on the Mac Mini

SAY THIS TO CLAUDE CODE
(replace the number with YOUR user ID):

Add my Telegram user ID 123456789 to the
OpenClaw Telegram allowFrom list in the
config file

DO: Test the bot by sending: Hello, are you there?
EXPECT: The bot responds through OpenClaw

Step 2: Connect Google Workspace

READ: This lets your agent use Gmail, Google Drive,
Google Calendar, and Google Docs.

DO: Open a browser and go to console.cloud.google.com
DO: Sign in with your bot's Google account
DO: Create a new project (or select existing)
DO: Enable these APIs:
— Gmail API
— Google Drive API
— Google Calendar API
— Google Docs API
DO: Go to APIs & Services > Credentials
DO: Create OAuth 2.0 Client ID
DO: Download the credentials JSON file

READ: The OpenClaw Google Workspace skill will walk

you through the OAuth authorization flow. You'll
need to authorize access in a browser window.

DO: Test by asking your agent:
"What's in my Google Drive?"
"What's on my calendar today?"

Step 3: Connect iCloud (Optional)

READ: Only do this if you want your agent to access
Apple Notes, Reminders, and Calendar on the Mac Mini.

DO: Create a NEW Apple ID at appleid.apple.com
(Do NOT use your personal Apple ID)
DO: On the Mac Mini, go to System Settings > Apple ID
DO: Sign in with the new Apple ID
DO: Open each app and allow iCloud sync:
— Open Notes > allow iCloud sync
— Open Reminders > allow iCloud sync
— Open Calendar > allow iCloud sync

READ: You'll go through permission screens for each
app, granting access one by one. This is normal.

Step 4: Verify All Skills

COPY AND PASTE INTO TERMINAL:

openclaw skills

EXPECT: Every skill you installed shows "ready"

DO: Test each skill by asking your agent:
"Summarize this URL: [paste any article URL]"
"What's the weather in [your city]?"
"Generate an image of a sunset over the ocean"
"What's my API usage looking like?"

Step 5: Verify 24/7 Operation

READ: Your server should stay on and your agent
should respond even when the TV is off and
you've walked away for hours.

DO: Turn off your TV or monitor
DO: Wait 30 minutes
DO: Message your bot on Telegram from your phone
EXPECT: The bot responds normally

WHEN YOUR HDMI DUMMY PLUG ARRIVES:
DO: Plug it into the Mac Mini's HDMI port
DO: Disconnect the TV or monitor
READ: The Mac Mini thinks a display is connected.
Screen Sharing will work at full resolution.

Step 6: Verify API Keys

DO: Chat with the bot on Telegram
(tests your Anthropic key)
DO: Ask for an image
(tests your Gemini key)
DO: Ask about Google Drive
(tests Google Workspace)

IF ANY KEY ISN'T WORKING:

DO: Open Claude Code

SAY THIS TO CLAUDE CODE:

Check all API keys in the OpenClaw configuration
and tell me which ones are set and which are missing

Monthly Security Check

COPY AND PASTE INTO TERMINAL:

openclaw security audit --deep

THEN:

openclaw security audit --fix

READ: Run these once a month. Keep your API keys secure. Don't share your bot token. Don't give others access. Keep your server behind Tailscale.

==========
REFERENCE — WHAT EVERY TOOL DOES
==========

READ: Here's a plain-English explanation of every tool you just installed and why you need it.

Xcode Command Line Tools
Apple's developer foundation. Nothing installs without it. Like a building's concrete foundation.

Homebrew
An app store for Terminal. You say what you want, it gets it. Like a personal shopper.

Node.js
The language Claude Code and OpenClaw speak. Like installing a language on your computer.

Python
The language your bots speak. Now your computer is bilingual.

Git
Downloads code from the internet. Like a librarian who brings back any book you need.

Claude Code
An AI developer in your Terminal. You type English, it codes. Like hiring a developer who speaks your language.

tmux
Keeps bots running when you close Terminal. Like giving each bot their own office.

pm2
Auto-restarts crashed bots. Auto-starts after power outages. Like a building manager.

ffmpeg
Processes audio and video files. Like a production studio you control by typing.

ImageMagick
Processes and resizes images. Like a graphic designer who works instantly.

jq
Reads data from API responses. Like a translator who finds the key info in a giant document.

wget
Downloads files from URLs. Like a delivery driver.

htop
Shows what your server is doing in real time. Like a vital signs monitor for your computer.

python-telegram-bot
How your bot hears and speaks on Telegram.

anthropic (Python library)
How your bot thinks. Connects to Claude's brain.

requests
Lets your bots talk to websites and APIs.

schedule
Lets your bots do things at specific times.

Tailscale
Access your server from anywhere in the world. A private tunnel between all your devices.

rclone
Uploads files to Google Drive from Terminal.

OpenClaw

The platform that manages your AI agent.
Skills, memory, conversations, connections.

==========

WHAT'S NEXT

==========

READ: Once your server is running and your agent is chatting on Telegram, explore these next:

Train your agent's personality and identity
Load training documents onto an external drive
Build specialized agents for different roles

==========

This guide was created by Shea Johnson.
AI Business Coach | www.mssheajohnson.com

2026 Shea Johnson LLC. All rights reserved.

==========