



☰ Hosting and deployment > macOS VMs

Hosting and deployment

macOS VMs

Recommended default (most users)

Small Linux VPS for an always-on Gateway and low cost. See [VPS hosting](#).

Dedicated hardware (Mac mini or Linux box) if you want full control and a **residential IP** for browser automation. Many sites block data center IPs, so local browsing often works better.

Hybrid: keep the Gateway on a cheap VPS, and connect your Mac as a **node** when you need browser/UI automation. See [Nodes](#) and [Gateway remote](#).

Use a macOS VM when you specifically need macOS-only capabilities (iMessage/BlueBubbles) or want strict isolation from your daily Mac.

macOS VM options

Local VM on your Apple Silicon Mac (Lume)

Run OpenClaw in a sandboxed macOS VM on your existing Apple Silicon Mac using [Lume](#).

This gives you:

- Full macOS environment in isolation (your host stays clean)

- iMessage support via BlueBubbles (impossible on Linux/Windows)



Instant reset by cloning VMs

No extra hardware or cloud costs

>

Hosted Mac providers (cloud)

If you want macOS in the cloud, hosted Mac providers work too:

MacStadium (hosted Macs)

Other hosted Mac vendors also work; follow their VM + SSH docs

Once you have SSH access to a macOS VM, continue at step 6 below.

Quick path (Lume, experienced users)

1. Install Lume
 2. `lume create openclaw --os macos --ipsw latest`
 3. Complete Setup Assistant, enable Remote Login (SSH)
 4. `lume run openclaw --no-display`
 5. SSH in, install OpenClaw, configure channels
 6. Done
-

What you need (Lume)

Apple Silicon Mac (M1/M2/M3/M4)

macOS Sequoia or later on the host

~60 GB free disk space per VM

~20 minutes



1) Install Lume

>

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/trycua/ci
```

If `~/local/bin` isn't in your PATH:

```
echo 'export PATH="$PATH:$HOME/local/bin"' >> ~/.zshrc && source ~/.z
```

Verify:

```
lume --version
```

Docs:

2) Create the macOS VM

```
lume create openclaw --os macos --ipsw latest
```

This downloads macOS and creates the VM. A VNC window opens automatically.

Note: The download can take a while depending on your connection.

3) Complete Setup Assistant

In the VNC window:



1. Select language and region
2. Skip Apple ID (or sign in if you want iMessage later)
3. Create a user account (remember the username and password)
4. Skip all optional features

After setup completes, enable SSH:

1. Open System Settings → General → Sharing
2. Enable “Remote Login”

4) Get the VM's IP address

```
lume get openclaw
```

Look for the IP address (usually `192.168.64.x`).

5) SSH into the VM

```
ssh youruser@192.168.64.X
```

Replace `youruser` with the account you created, and the IP with your VM's IP.

6) Install OpenClaw



Inside the VM:

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

Follow the onboarding prompts to set up your model provider (Anthropic, OpenAI, etc.).

7) Configure channels

Edit the config file:

```
nano ~/.openclaw/openclaw.json
```

Add your channels:

```
{  
  "channels": {  
    "whatsapp": {  
      "dmPolicy": "allowlist",  
      "allowFrom": ["+15551234567"]  
    },  
    "telegram": {  
      "botToken": "YOUR_BOT_TOKEN"  
    }  
  }  
}
```

Then login to WhatsApp (scan QR):



openclaw channels login

>

8) Run the VM headlessly

Stop the VM and restart without display:

```
lume stop openclaw  
lume run openclaw --no-display
```

The VM runs in the background. OpenClaw's daemon keeps the gateway running.

To check status:

```
ssh youruser@192.168.64.X "openclaw status"
```

Bonus: iMessage integration

This is the killer feature of running on macOS. Use [BlueBubbles](#) to add iMessage to OpenClaw.

Inside the VM:

1. Download BlueBubbles from [bluebubbles.app](#)
2. Sign in with your Apple ID
3. Enable the Web API and set a password
4. Point BlueBubbles webhooks at your gateway (example: `https://your-gateway-host:3000/bluebubbles-webhook?password=<password>`)

Add to your OpenClaw config:



```
{  
  "channels": {  
    "bluebubbles": {  
      "serverUrl": "http://localhost:1234",  
      "password": "your-api-password",  
      "webhookPath": "/bluebubbles-webhook"  
    }  
  }  
}
```

Restart the gateway. Now your agent can send and receive iMessages.

Full setup details:

Save a golden image

Before customizing further, snapshot your clean state:

```
lume stop openclaw  
lume clone openclaw openclaw-golden
```

Reset anytime:

```
lume stop openclaw && lume delete openclaw  
lume clone openclaw-golden openclaw  
lume run openclaw --no-display
```

Running 24/7



Keep the VM running by:

Keeping your Mac plugged in

Disabling sleep in System Settings → Energy Saver

Using `caffeinate` if needed

For true always-on, consider a dedicated Mac mini or a small VPS. See [VPS hosting](#).

Troubleshooting

Problem	Solution
Can't SSH into VM	Check "Remote Login" is enabled in VM's System Settings
VM IP not showing	Wait for VM to fully boot, run <code>lume get openclaw</code> again
Lume command not found	Add <code>~/local/bin</code> to your PATH
WhatsApp QR not scanning	Ensure you're logged into the VM (not host) when running <code>openclaw channels login</code>

Related docs

[VPS hosting](#)

[Nodes](#)

[Gateway remote](#)

[BlueBubbles channel](#)



[Lume Quickstart](#)

[Lume CLI Reference](#)

[Unattended VM Setup](#) (advanced)

[Docker Sandboxing](#) (alternative isolation approach)

[< GCP](#)

[exe.dev >](#)

Powered by [mintlify](#)