Messaging platforms › **iMessage**

Messaging platforms

iMessage

> ⚠ For new iMessage deployments, use **BlueBubbles**.
>
> The `imsg` integration is legacy and may be removed in a future release.

Status: legacy external CLI integration. Gateway spawns `imsg rpc` and communicates over JSON-RPC on stdio (no separate daemon/port).

BlueBubbles (recommended)
Preferred iMessage path for new setups.

Pairing
iMessage DMs default to pairing mode.

Configuration reference
Full iMessage field reference.

Quick setup

Local Mac (fast path) Remote Mac over SSH

1 **Install and verify imsg**

```
brew install steipete/tap/imsg
imsg rpc --help
```

2 Configure OpenClaw

```
{
  channels: {
    imessage: {
      enabled: true,
      cliPath: "/usr/local/bin/imsg",
      dbPath: "/Users/<you>/Library/Messages/chat.db",
    },
  },
}
```

3 Start gateway

```
openclaw gateway
```

4 Approve first DM pairing (default dmPolicy)

```
openclaw pairing list imessage
openclaw pairing approve imessage <CODE>
```

Pairing requests expire after 1 hour.

Requirements and permissions (macOS)

Messages must be signed in on the Mac running `imsg` .

Full Disk Access is required for the process context running OpenClaw/ `imsg` (Messages DB access).

Automation permission is required to send messages through Messages.app.

Permissions are granted per process context. If gateway runs headless (LaunchAgent/SSH), run a one-time interactive command in that same context to trigger prompts:

```
imsg chats --limit 1
# or
imsg send <handle> "test"
```

Access control and routing

DM policy Group policy + mentions Sessions and deterministic replies

`channels.imessage.dmPolicy` controls direct messages:

`pairing` (default)

`allowlist`

`open` (requires `allowFrom` to include `"*"`)

`disabled`

Allowlist field: `channels.imessage.allowFrom` .

Allowlist entries can be handles or chat targets (`chat_id:*` , `chat_guid:*` , `chat_identifier:*`).

Deployment patterns

Dedicated bot macOS user (separate iMessage identity)

Remote Mac over Tailscale (example)

Multi-account pattern

Media, chunking, and delivery targets

Attachments and media

Outbound chunking

Addressing formats

Config writes

iMessage allows channel-initiated config writes by default (for `/config set|unset` when `commands.config: true`).

Disable:

```
{
  channels: {
    imessage: {
      configWrites: false,
    },
  },
}
```

Troubleshooting

imsg not found or RPC unsupported

DMs are ignored

Group messages are ignored

Remote attachments fail

macOS permission prompts were missed

Configuration reference pointers

- Configuration reference - iMessage
- Gateway configuration
- Pairing
- BlueBubbles

‹ Signal Microsoft Teams ›