



☰ Messaging platforms > **Zalo**

Messaging platforms

Zalo

Status: experimental. Direct messages only; groups coming soon per Zalo docs.

Plugin required

Zalo ships as a plugin and is not bundled with the core install.

Install via CLI: `openclaw plugins install @openclaw/zalo`

Or select **Zalo** during onboarding and confirm the install prompt

Details: [Plugins](#)

Quick setup (beginner)

1. Install the Zalo plugin:

From a source checkout: `openclaw plugins install ./extensions/zalo`

From npm (if published): `openclaw plugins install @openclaw/zalo`

Or pick **Zalo** in onboarding and confirm the install prompt

2. Set the token:

Env: `ZALO_BOT_TOKEN=...`

Or config: `channels.zalo.botToken: "..." .`

3. Restart the gateway (or finish onboarding).

- 
4. DM access is pairing by default; approve the pairing code on first contact.

Minimal config: >

```
{  
  channels: {  
    zalo: {  
      enabled: true,  
      botToken: "12345689:abc-xyz",  
      dmPolicy: "pairing",  
    },  
  },  
}
```

What it is

Zalo is a Vietnam-focused messaging app; its Bot API lets the Gateway run a bot for 1:1 conversations. It is a good fit for support or notifications where you want deterministic routing back to Zalo.

A Zalo Bot API channel owned by the Gateway.

Deterministic routing: replies go back to Zalo; the model never chooses channels.

DMs share the agent's main session.

Groups are not yet supported (Zalo docs state “coming soon”).

Setup (fast path)

1) Create a bot token (Zalo Bot Platform)

1. Go to [Zalo Bot Platform](#) and sign in.
2. Create a new bot and configure its settings.
3. Copy the bot token (format: 12345689:abc-xyz).

2) Configure the token (env or config)



Example:

```
{  
  channels: {  
    zalo: {  
      enabled: true,  
      botToken: "12345689:abc-xyz",  
      dmPolicy: "pairing",  
    },  
  },  
}
```

Env option: ZALO_BOT_TOKEN=... (works for the default account only).

Multi-account support: use channels.zalo.accounts with per-account tokens and optional name .

3. Restart the gateway. Zalo starts when a token is resolved (env or config).
4. DM access defaults to pairing. Approve the code when the bot is first contacted.

How it works (behavior)

Inbound messages are normalized into the shared channel envelope with media placeholders.

Replies always route back to the same Zalo chat.

Long-polling by default; webhook mode available with channels.zalo.webhookUrl .

Limits



Outbound text is chunked to 2000 characters (Zalo API limit).

Media downloads/uploads are capped by `channels.zalo.mediaMaxMb` (default 5).

>

Streaming is blocked by default due to the 2000 char limit making streaming less useful.

Access control (DMs)

DM access

Default: `channels.zalo.dmPolicy = "pairing"`. Unknown senders receive a pairing code; messages are ignored until approved (codes expire after 1 hour).

Approve via:

```
openclaw pairing list zalo
```

```
openclaw pairing approve zalo <CODE>
```

Pairing is the default token exchange. Details: [Pairing](#)

`channels.zalo.allowFrom` accepts numeric user IDs (no username lookup available).

Long-polling vs webhook

Default: long-polling (no public URL required).

Webhook mode: set `channels.zalo.webhookUrl` and `channels.zalo.webhookSecret`.

The webhook secret must be 8-256 characters.

Webhook URL must use HTTPS.

Zalo sends events with `X-Bot-Api-Secret-Token` header for verification.



Gateway HTTP handles webhook requests at `channels.zalo.webhookPath` (defaults to the webhook URL path).

Note: `getUpdates` (polling) and webhook are mutually exclusive per Zalo API docs.

Supported message types

Text messages: Full support with 2000 character chunking.

Image messages: Download and process inbound images; send images via `sendPhoto`.

Stickers: Logged but not fully processed (no agent response).

Unsupported types: Logged (e.g., messages from protected users).

Capabilities

Feature	Status
Direct messages	✓ Supported
Groups	✗ Coming soon (per Zalo docs)
Media (images)	✓ Supported
Reactions	✗ Not supported
Threads	✗ Not supported
Polls	✗ Not supported
Native commands	✗ Not supported
Streaming	⚠ Blocked (2000 char limit)

Delivery targets (CLI/cron)

Use a chat id as the target.

 Example: `openclaw message send --channel zalo --target 123456789 --message "hi" .`

>

Troubleshooting

Bot doesn't respond:

Check that the token is valid: `openclaw channels status --probe`

Verify the sender is approved (pairing or allowFrom)

Check gateway logs: `openclaw logs --follow`

Webhook not receiving events:

Ensure webhook URL uses HTTPS

Verify secret token is 8-256 characters

Confirm the gateway HTTP endpoint is reachable on the configured path

Check that getUpdates polling is not running (they're mutually exclusive)

Configuration reference (Zalo)

Full configuration: [Configuration](#)

Provider options:

`channels.zalo.enabled` : enable/disable channel startup.

`channels.zalo.botToken` : bot token from Zalo Bot Platform.

`channels.zalo.tokenFile` : read token from file path.

`channels.zalo.dmPolicy` : `pairing` | `allowlist` | `open` | `disabled` (default: `pairing`).

`channels.zalo.allowFrom` : DM allowlist (user IDs). `open` requires `"*"`. The wizard will ask for numeric IDs.



channels.zalo.mediaMaxMb : inbound/outbound media cap (MB, default 5).
channels.zalo.webhookUrl : enable webhook mode (HTTPS required).
channels.zalo.webhookSecret : webhook secret (8-256 chars).
channels.zalo.webhookPath : webhook path on the gateway HTTP server.
channels.zalo.proxy : proxy URL for API requests.

Multi-account options:

channels.zalo.accounts.<id>.botToken : per-account token.
channels.zalo.accounts.<id>.tokenFile : per-account token file.
channels.zalo.accounts.<id>.name : display name.
channels.zalo.accounts.<id>.enabled : enable/disable account.
channels.zalo.accounts.<id>.dmPolicy : per-account DM policy.
channels.zalo.accounts.<id>.allowFrom : per-account allowlist.
channels.zalo.accounts.<id>.webhookUrl : per-account webhook URL.
channels.zalo.accounts.<id>.webhookSecret : per-account webhook secret.
channels.zalo.accounts.<id>.webhookPath : per-account webhook path.
channels.zalo.accounts.<id>.proxy : per-account proxy URL.

< Matrix

Zalo Personal >

Powered by [mintlify](#)