

PostingPilot Channel Credentials

Setup steps, required values and a printable checklist for every network

Version 1.5 · postingpilot.app/docs

Create the app, token and scopes each social network requires, paste them into PostingPilot, and tick off the checklist at the back before you schedule. The troubleshooting section maps common errors back to these steps.

Contents

How channel credentials work

LinkedIn

Facebook Page

Instagram Business

Threads

YouTube

X (Twitter)

TikTok

Storage, rotation and troubleshooting

Setup checklist — how to use it

Checklist — every network

How to use this page

Authentication and token errors

Permission and scope errors

Wrong or missing account IDs

Media and content rejections

Rate limits and delivery delays

Still stuck

How channel credentials work

PostingPilot publishes directly to each platform using credentials you create in that platform's own developer console. Nothing is routed through a third-party scheduler, so the permissions stay under your control and you can revoke them at any time.

Every credential is stored per workspace, encrypted at rest with AES-256-GCM, and is only decrypted server-side at publish time. Keys are never returned to the browser.

- 1 Create the app/token in the platform's developer console using the steps below.
- 2 In PostingPilot open Settings, choose the workspace, and paste the values into the fields for that platform.
- 3 Go to Channels, add the channel, and press Test connection - PostingPilot verifies the token and pulls the connected account name.
- 4 The channel shows Ready when it can publish. Anything missing (token, account ID, expired key) is listed on the connection health panel.

Tip: Use the redirect/callback URL <https://postingpilot.app/app/channels> for any OAuth app you create, unless the platform-specific steps say otherwise.

LinkedIn

LinkedIn posts are published to your member profile (or an organisation page if your app is approved for it).

- 1 Go to the LinkedIn Developer portal and create an app, linking it to a LinkedIn Page you administer.
 - 2 On the Products tab, request 'Share on LinkedIn' and 'Sign In with LinkedIn using OpenID Connect'. For company pages also request 'Community Management API'.
 - 3 On the Auth tab add the redirect URL <https://postingpilot.app/app/channels> and copy the Client ID and Client Secret.
 - 4 Authorise with the scopes openid, profile, w_member_social (add w_organization_social for pages) and copy the resulting access token.
 - 5 In PostingPilot Settings paste the access token as the LinkedIn credential.
 - 6 Add the LinkedIn channel and press Test connection - PostingPilot resolves your member URN automatically.
- Required values: access token (member or organisation).
 - Token lifetime: 60 days for member tokens - reconnect when the token health panel flags it.
 - Limits: text and link posts; images supported, video via upload.

Facebook Page

- 1 At developers.facebook.com create an app of type Business and add the Facebook Login and Pages products.
 - 2 In the Graph API Explorer select your app, then select the Page you want to publish to.
 - 3 Request the permissions `pages_manage_posts`, `pages_read_engagement` and `pages_show_list`, then generate a user access token.
 - 4 Exchange it for a long-lived Page access token (Access Token Debugger shows the expiry; a Page token derived from a long-lived user token does not expire).
 - 5 Copy the Page ID from the Page's About tab or from `/me/accounts`.
 - 6 In PostingPilot Settings paste the Meta access token, then add the Facebook channel and enter the Page ID.
- Required values: long-lived Page access token, Page ID.
 - App review: publishing to Pages you do not administer requires Meta App Review; your own Pages work in development mode.
 - Status reconciliation: add the PostingPilot webhook in the Meta app's Webhooks product if you want delivery confirmations pushed back.

Instagram Business

Instagram publishing uses the same Meta app and token as Facebook. The account must be an Instagram Business or Creator account linked to a Facebook Page.

- 1 Convert the Instagram account to Business/Creator and link it to your Facebook Page (Page settings, Linked accounts).
 - 2 In the Meta app add the Instagram Graph API product.
 - 3 Add the permissions `instagram_basic`, `instagram_content_publish` and `pages_read_engagement` to the token you generated for Facebook.
 - 4 Find the Instagram Business account ID via the Graph API:
`/[page-id]?fields=instagram_business_account`.
 - 5 In PostingPilot add the Instagram channel and enter the Instagram Business account ID; the Meta token from Settings is reused.
- Required values: Meta access token, Instagram Business account ID.
 - Every Instagram post must include an image or video - PostingPilot blocks publishing otherwise.
 - Limit: 25 API-published posts per account per 24 hours (Meta's cap).

Threads

- 1 In the Meta developer console add the Threads API product to your app (it uses a separate Threads app ID and secret).
 - 2 Add the redirect URL <https://postingpilot.app/app/channels>.
 - 3 Authorise with the scopes `threads_basic` and `threads_content_publish`.
 - 4 Exchange the short-lived token for a long-lived Threads token (60 days).
 - 5 Paste the Threads access token in PostingPilot Settings, then add the Threads channel and Test connection.
- Required values: Threads long-lived access token.
 - Limits: 500 characters, one image or video per post, 250 posts per 24 hours.

YouTube

- 1 In Google Cloud Console create a project and enable the YouTube Data API v3.
 - 2 Configure the OAuth consent screen (External), add your Google account as a test user, and add the scope <https://www.googleapis.com/auth/youtube.upload>.
 - 3 Create an OAuth client ID of type Web application with the redirect URI <https://postingpilot.app/app/channels>.
 - 4 Authorise once with `access_type=offline` and `prompt=consent` to obtain a refresh token.
 - 5 In PostingPilot Settings paste the client ID, client secret and refresh token, then add the YouTube channel.
- Required values: OAuth client ID, client secret, refresh token.
 - Default quota is 10,000 units/day - each upload costs about 1,600 units (roughly six uploads a day).
 - Unverified apps upload videos as private until Google verifies the project.

X (Twitter)

- 1 At developer.x.com create a project and an app on a plan that allows write access (Free tier permits limited posting).
- 2 In User authentication settings enable OAuth 2.0, set App permissions to Read and write, app type Web App, and the callback URL <https://postingpilot.app/app/channels>.
- 3 Copy the OAuth 2.0 Client ID and Client Secret.
- 4 Authorise with the scopes `tweet.read`, `tweet.write`, `users.read` and `offline.access`, and copy the access and refresh tokens.

- 5 Paste the client ID, client secret and refresh token in PostingPilot Settings, then add the X channel.
- Required values: OAuth 2.0 client ID, client secret, refresh token.
 - Access tokens last two hours - PostingPilot refreshes them automatically using the refresh token.
 - Limits: 280 characters, up to four images or one video.

TikTok

- 1 At developers.tiktok.com create an app and add the Content Posting API product.
 - 2 Add the scopes video.publish (or video.upload) and user.info.basic.
 - 3 Set the redirect URI to <https://postingpilot.app/api/public/tiktok/callback>.
 - 4 Copy the Client key and Client secret into PostingPilot Settings.
 - 5 On the Channels page press Connect TikTok and complete the TikTok login - PostingPilot stores the tokens for you.
- Required values: client key, client secret (the OAuth flow supplies the tokens).
 - Until your app passes TikTok audit, posts publish to a private/draft state for the creator to confirm.
 - Refresh tokens last 365 days; re-authorise from Channels when flagged.

Storage, rotation and troubleshooting

- Encryption: keys are encrypted with AES-256-GCM before they touch the database and are decrypted only inside the publisher.
- Rotation: replacing a key in Settings re-encrypts it immediately; the key rotation banner tells you if any legacy value still needs re-saving.
- Access: only workspace owners and editors can view or change credentials; viewers and client portal users never see them.
- Revoking: delete the value in Settings, then revoke the app's access in the platform console to be certain.
- 'Test connection' fails with 401/invalid token: the token expired or the app permissions changed - regenerate and paste it again.
- Publishing fails with a permissions error: a required scope is missing; re-authorise with the scopes listed for that platform.
- Instagram publish blocked: the post has no image, or the account is not a Business/Creator account linked to a Page.

- Rate limited: PostingPilot queues and retries automatically with backoff - watch the retry queue and delivery history.

Tip: Still stuck? Ask Pilot inside the app, or send the workspace, channel and timestamp from Delivery history to support.

Setup checklist — how to use it

Print this checklist or share the PDF with whoever administers each social account. Tick every line for a network before you schedule to it; the in-app credentials wizard (Setup, Credentials wizard) checks the same items automatically and finishes with a live connection test.

- Values marked required must be present or publishing will fail.
- Paste secrets into PostingPilot only — they are encrypted at rest and never emailed.
- After saving, press Test connection: a green result means the token, scopes and account ID all check out.

Checklist — every network

LinkedIn

- ☐ App created in the LinkedIn developer portal and linked to a Page
- ☐ Products approved: Share on LinkedIn, Sign In with OpenID Connect (plus Community Management for pages)
- ☐ Scopes on the token: openid, profile, w_member_social (w_organization_social for pages)
- ☐ Required value saved: access token
- ☐ Optional: author URN for the profile or organisation
- ☐ Test connection passes

Facebook Page

- ☐ Business app created with Facebook Login and Pages products
- ☐ Permissions granted: pages_manage_posts, pages_read_engagement, pages_show_list
- ☐ Short-lived token exchanged for a long-lived Page token
- ☐ Required values saved: Page access token, Page ID
- ☐ Test connection passes

Instagram Business

- ☐ Account converted to Business/Creator and linked to the Facebook Page
- ☐ Instagram Graph API product added to the Meta app

- ☐ Permissions granted: instagram_basic, instagram_content_publish, pages_read_engagement
- ☐ Required values saved: Meta access token, Instagram Business account ID
- ☐ Every post has an image or video attached
- ☐ Test connection passes

Threads

- ☐ Threads API product added, redirect URL registered
- ☐ Scopes granted: threads_basic, threads_content_publish
- ☐ Short-lived token exchanged for the 60-day long-lived token
- ☐ Required values saved: Threads access token, Threads user ID
- ☐ Test connection passes

YouTube

- ☐ Google Cloud project created with YouTube Data API v3 enabled
- ☐ OAuth consent screen configured with the youtube.upload scope
- ☐ Web application OAuth client created with the PostingPilot redirect URI
- ☐ Authorised with access_type=offline and prompt=consent
- ☐ Required values saved: client ID, client secret, refresh token
- ☐ Test connection passes

X (Twitter)

- ☐ Project and app created on a plan that allows write access
- ☐ OAuth 2.0 enabled, App permissions set to Read and write, callback URL registered
- ☐ Scopes granted: tweet.read, tweet.write, users.read, offline.access
- ☐ Required values saved: client ID, client secret, refresh token
- ☐ Test connection passes

TikTok

- ☐ App created with the Content Posting API product
- ☐ Scopes granted: video.publish (or video.upload), user.info.basic
- ☐ Redirect URI set to the PostingPilot TikTok callback
- ☐ Required values saved: client key, client secret
- ☐ Connect TikTok completed on the Channels page
- ☐ Test connection passes

Tip: Keep a note of who owns each developer account — token renewals need the same login.

How to use this page

Find the message you saw — in a Test connection result, on Delivery history, or in a failed-post email — and follow the fix. Each fix links back to the exact setup step in the Channel credentials guide.

- 1 Open Channels (or the credentials wizard) and press Test connection to get the precise failure.
- 2 Match the message below and apply the fix.
- 3 Re-run Test connection: every check must show green before the channel will publish.
- 4 If the post already failed, requeue it from the Retry queue.

Authentication and token errors

- 401 Unauthorized / 'invalid token' / 'Could not authenticate you' — the stored token expired or was revoked. Generate a new token in the platform console and re-save it in the credentials wizard. See [/docs/credentials](#) for that network's steps.
- 'Error validating access token: Session has expired' (Meta, code 190) — the Page token was short-lived. Exchange it for a long-lived token in the Access Token Debugger and save that value. See [credentials-facebook](#).
- 'invalid_grant' (YouTube/Google) — the refresh token was revoked or was issued without offline access. Re-authorise with `access_type=offline` and `prompt=consent`, then save the new refresh token. See [credentials-youtube](#).
- 'unauthorized_client' or refresh fails immediately (X) — the client ID/secret saved don't match the app that issued the refresh token. Re-copy both from the X developer portal. See [credentials-x](#).
- 'No ... token saved for this workspace' — nothing is stored under that provider in the current workspace. Check the workspace switcher, then save the credential again.
- 'This stored key predates encryption — re-save it' — open the credentials wizard and paste the value again to re-encrypt it.

Permission and scope errors

- 403 with 'insufficient permissions' / 'scope' in the message — the token was issued before you added the scope. Add the scope, re-authorise, and save the new token; adding a product in the console does not upgrade an existing token.

- Meta '(#200) Requires pages_manage_posts' — regenerate the Page token with pages_manage_posts, pages_read_engagement and pages_show_list. See credentials-facebook.
- Instagram '(#10) Application does not have permission for this action' — the account is not Business/Creator, or is not linked to the Page the token covers. See credentials-instagram.
- LinkedIn 'ACCESS_DENIED' when posting to a company page — the app needs Community Management API approval plus w_organization_social. See credentials-linkedin.
- X 403 'oauth1 app permissions' / 'Read-only application cannot POST' — set App permissions to Read and write, then re-authorise. See credentials-x.
- TikTok 'scope_not_authorized' — add video.publish and user.info.basic, then press Connect TikTok again. See credentials-tiktok.

Wrong or missing account IDs

- 'Add the Facebook Page ID to this channel' — the channel has no Page ID. Copy it from the Page's About tab or /me/accounts and save it on the channel. See credentials-facebook.
- 'Add the Instagram Business account ID' — read it from /{page-id}?fields=instagram_business_account. See credentials-instagram.
- Meta '(#803) Some of the aliases you requested do not exist' — the ID belongs to a different asset, or the token doesn't cover it. Re-check the ID and that the token was generated for that Page.
- 'Threads user ID' missing — Threads publishing needs the numeric user ID the token belongs to. See credentials-threads.
- YouTube uploads land on the wrong channel — the Google account manages several channels; set the channel ID on the PostingPilot channel. See credentials-youtube.

Media and content rejections

- 'Instagram posts require an image or video' — attach media in the editor; Instagram has no text-only posts.
- 'X posts are limited to 280 characters' — shorten the draft or let the pre-publish score suggest a trim.
- Threads 'post too long' — Threads caps posts at 500 characters, one image or video.
- Meta 'media could not be fetched' — the image URL must be publicly reachable; re-upload it to the media library and reschedule.

- YouTube video stays private after upload — Google has not verified the project yet; unverified apps upload as private.
- TikTok post appears as a draft — until your TikTok app passes audit, posts land in the creator's inbox to confirm manually.

Rate limits and delivery delays

- 429 / 'rate limit' — PostingPilot retries automatically with backoff; watch the Retry queue rather than rescheduling by hand.
- Instagram 'reached the maximum number of posts' — Meta allows 25 API posts per account per 24 hours.
- Threads 250 posts per 24 hours; X limits depend on your developer plan.
- YouTube 'quotaExceeded' — the default 10,000 units/day allows roughly six uploads; request more quota or spread uploads out.
- A post sits on 'scheduled' past its time — check Scheduler runs for the last run and Publishing health for a stalled queue.

Still stuck

- Re-run Test connection and note which check turns red — that names the exact missing piece.
- Confirm you are in the right workspace: credentials are stored per workspace.
- Check Delivery history for the raw platform error on the failed attempt.
- Ask Pilot in the app — it has this page and the credentials guide in its knowledge.

Tip: When contacting support, include the workspace, channel, timestamp and the exact error text from Delivery history.

Version history

v1.5 · 2026-08-25

- Added the in-app credentials wizard: guided per-network setup with required-field checks and a live connection test.
- Added a unified Test connection action covering LinkedIn, Facebook, Instagram, Threads, YouTube, X and TikTok.
- Added a printable channel credentials PDF with a per-network setup checklist.
- Added the troubleshooting guide mapping common connection and publishing errors to exact setup fixes.

v1.4 · 2026-08-25

- Added the Channel credentials guide: step-by-step app, token and scope setup for LinkedIn, Facebook, Instagram, Threads, YouTube, X and TikTok.
- Documented credential storage, encryption, rotation and connection troubleshooting.

v1.3 · 2026-08-19

- Added TikTok and Threads publishing, plus the public status page for all seven channels.
- Documented the engagement dashboard, metrics freshness, historical snapshots and account-level metrics.
- Added campaigns and content mix, A/B experiments, UTM links and conversion attribution.
- Added bulk CSV import/export, the demo workspace and email invites for clients.
- Documented agency operations: white-label, client billing rollup, audit retention and export attestation.
- Added seat add-ons, trial limit nudges, account deletion and data export, and the Ask Pilot help assistant.

v1.2 · 2026-08-17

- Documented the guided setup wizard and its channel connection health panel.
- Added Facebook, Instagram, YouTube and X (Twitter) direct publishing, including media and token requirements.
- Added sections for the notifications centre, team members and roles, and bulk scheduling.
- Documented the pattern library, trend radar and pre-publish scoring.

v1.1 · 2026-07-30

- Added in-page search, printable PDF editions and a feedback widget.

- Expanded the troubleshooting section with retry-queue and usage-limit guidance.

v1.0 · 2026-07-29

- First published edition of the User Manual and Quick Start Guide.