api0

GUIDE 09

# Ton API dans Telegram, avec un bot créé en deux minutes

Le pas-à-pas pour mettre les outils d'un tenant api0 dans Telegram : le bot chez @BotFather, son token, le branchement dans api0, la liaison du compte de chacun et le partage avec l'équipe, avec ce que le test d'api0 vérifie.

Mis à jour le 11 novembre 2026 · Lire en ligne : api0.ai/fr/guides/ton-api-dans-telegram

---

À la fin de ce guide, ton équipe écrit à un bot Telegram, et l'assistant d'api0 répond en appelant les outils du tenant **au nom de chaque personne**, avec ses droits. Les outils et les accès sont ceux que tu as déjà configurés dans api0, les mêmes que dans Claude.

Compte dix minutes. Contrairement à WhatsApp ou Teams, il n'y a rien à recoller ailleurs : api0 branche le webhook lui-même.

Avant de commencer :

- **Une API qui marche déjà dans api0** : voir le guide principal.
- **Telegram**, sur ton téléphone ou ton ordinateur.

## Comment un message circule

```
Personne sur Telegram ⟶ Telegram ⟶ webhook api0 ⟶ assistant ⟶ ton API
```

Au branchement, api0 vérifie le token auprès de Telegram et lui indique où livrer les messages du bot. L'assistant appelle ensuite les outils pour la personne liée à ce compte Telegram.

## 1. Créer le bot chez @BotFather

Dans Telegram, ouvre @BotFather et envoie `/newbot` :

1. un **nom affiché**, par exemple *Acme Assistant* ;
2. un **nom d'utilisateur**, qui doit finir par `bot`, par exemple `acme_assistant_bot`.

Le nom d'utilisateur est unique sur tout Telegram : si BotFather le refuse, essaie-en un autre. Il devient le lien du bot (`t.me/acme_assistant_bot`) et ne peut plus être changé. La page Telegram d'api0 propose des noms tirés du nom de ton tenant.

**Prends un bot dédié.** Le brancher sur api0 le détourne : un bot déjà relié à un autre service n'y reçoit plus les messages.

## 2. Copier le token

BotFather répond avec un token du type `123456789:AAH…`. Traite-le comme un mot de passe : qui le détient contrôle le bot.

Perdu ou divulgué ? Dans BotFather : `/mybots` → ton bot → **API Token** → **Revoke current token**, puis rebranche le bot avec le nouveau.

Facultatif, toujours dans BotFather : `/setuserpic` pour une photo, `/setdescription` pour ce que les gens voient avant de démarrer la conversation, `/setabouttext` pour son profil.

## 3. Brancher le bot dans api0

Sur app.api0.ai → **Channels** → **Telegram**, colle le token, ajoute si tu veux des instructions pour le bot (ton, longueur des réponses, confirmation avant une action), et clique **Connect**.

api0 vérifie le token avec Telegram et enregistre le webhook avant de sauvegarder quoi que ce soit : un bloc qui dit « connecté » l'est vraiment. Le token est gardé chiffré.

## 4. Lier son compte et essayer

Chaque personne le fait une fois, de préférence sur le téléphone où elle utilise Telegram :

1. Dans api0, le bloc Telegram affiche **Link my Telegram**. Touche-le, puis **Open** : le bot s'ouvre.
2. Appuie sur **Start**. L'encadré passe au vert quand le compte est lié.
3. Écris au bot une vraie demande. Il répond avec les outils d'api0, en ton nom.

Une personne pas encore liée reçoit une réponse qui explique comment se lier, pas une réponse à sa question.

## 5. Partager avec l'équipe

Le bloc Telegram affiche le **lien du bot**. Envoie-le à ton équipe : chacun l'ouvre et lie son propre compte comme à l'étape 4.

Ce que chacun peut faire suit le réglage d'accès du tenant. Si l'accès passe par l'identifiant de chaque personne, elle doit aussi l'avoir ajouté dans api0 (**Settings** → **My access tokens**) avant que les outils marchent pour elle.

## 6. Vérifier avec le test

Clique **Test connection** dans le bloc Telegram. Il vérifie que Telegram accepte le token, qu'il livre les messages du bot à api0, que ton propre compte est lié, et que le dernier message est bien arrivé.

## Récapitulatif des erreurs

| Message | Cause | Correctif |
|---|---|---|
| BotFather refuse le nom d'utilisateur | Déjà pris, ou ne finit pas par `bot` | En essayer un autre qui finit par `bot` |
| `Telegram rejected the stored bot token.` | Token révoqué ou mal copié | `/mybots` → **API Token**, puis rebrancher avec le nouveau token |
| `Telegram has no webhook registered for this bot.` | Le webhook a été retiré | Rebrancher le bot (**Disconnect**, puis **Connect** avec le même token) |
| `Telegram delivers this bot's messages to …, not to api0.` | Un autre service a repris le bot | Prendre un bot dédié, ou rebrancher celui-ci sur api0 |
| `Telegram cannot deliver — … update(s) waiting.` | Telegram n'arrive pas à livrer les messages à api0 | Le test affiche la dernière erreur de Telegram : la transmettre à api0 si elle dure |
| `Your own Telegram account is not linked yet.` | Pas encore de liaison pour ton compte | **Link my Telegram**, puis **Start** (étape 4) |
| Le bot répond seulement comment se lier | La personne n'a pas lié son compte | Elle fait l'étape 4 |