api0

GUIDE 10

Inbound Authentication: Who Is Calling api0?

How api0 knows who is calling, before every tool: OAuth 2.1 for Claude and MCP clients, Google or the company's identity provider for people, API keys for servers, and account linking for Teams, WhatsApp and Telegram.

Updated 11 November 2026 · Read online: api0.ai/guides/inbound-authentication

Before calling a tool, the gateway always knows **who is calling, and for which tenant**. That's what lets it show only that tenant's tools, meter credits, log the call, then pick the credential to send to your application.

This guide describes every way to identify yourself to api0. What api0 then sends to your application is covered in outbound authentication (guide 11). The full path of a request is in the detailed architecture (guide 04).

The models

Model	Used by	Detail
OAuth 2.1 + PKCE	Claude and MCP clients	The client registers with the tenant's Client ID; the user signs in; the gateway issues a token bound to that user and that tenant
Google sign-in	Default	For the dashboard and user sign-in
Your identity provider (OIDC)	Enterprises	Entra ID or any OIDC IdP configured per tenant: your employees sign in with their account, multi-factor and conditional access included
API key	Server-to-server integrations	One key per tenant or per consumer
Messaging account linking	Teams, WhatsApp, Telegram	Each person links their messaging account to their api0 account once

Claude and MCP clients: OAuth 2.1

`/mcp` follows the MCP protocol's standard authorization: OAuth 2.1 with PKCE. In Claude, you add a custom connector with the tenant's MCP server URL and its **Client ID**, no secret; the person signs in, and the gateway issues a token bound to them and to that tenant. The step-by-step is in the main guide.

Cursor, Claude Desktop and other MCP clients connect the same way. The token is only valid for that tenant: it opens no other.

People: Google or your identity provider

By default, people sign in to api0 with Google, for the dashboard and for Claude's OAuth sign-in alike.

A company can plug in its own identity provider, per tenant: **Entra ID** or any **OIDC** provider. Its employees then sign in with their usual account, and its rules apply: multi-factor, conditional access, leavers. The provider's client secret is encrypted in api0. After sign-in, the provider sends people back to `https://gateway.api0.ai/oauth/idp/callback`.

For Entra ID, api0's setup wizard offers it on its first screen (*Microsoft 365 (Entra ID)*). The same Entra app can handle signing in to api0 and calling a Microsoft API: see guide 05, "Variants".

Servers: API keys

A server-to-server integration authenticates with an **API key**: a tenant key, or one key per consumer.

Acting for an end user

An application can call the gateway with its key **on behalf of one of its users**, by naming them. The gateway checks this is allowed before any call: the key's owner, always; a consumer key, never for anyone else; a tenant key, only for someone who joined that tenant themselves (through OAuth consent or their own key). Being added as a member is not enough. Otherwise: `403`.

Messaging apps: account linking

Teams, WhatsApp and Telegram only know a messaging identifier. So each person links it **once** to their api0 account, with a single-use code sent to the bot, or the channel's **Link my…** button on app.api0.ai. After that, every message is handled as coming from their api0 account, with their access.

Until a person is linked, the bot explains how, without answering their question and without calling any tool. The details are in each channel's guide: the Teams guide, the WhatsApp guide, the Telegram guide.

Which one for whom

Who is calling	Model
A person in Claude or another MCP client	OAuth 2.1, with Google or the company's identity provider
A company that already manages its accounts in Entra ID or an OIDC provider	Its identity provider, so its rules apply
A server, an application, a script	An API key; on behalf of a user, by naming them
A person in Teams, WhatsApp or Telegram	Account linking, once per channel