1. Start Here

Use the SIR Giving Partner API to add SIR rewards to your product. You can create donation links, reward users for actions in your own system, track balances, and receive webhook events when rewards are issued or reversed.

Most teams use one of two paths:

If you want to...	Start with
Add a donation button or embedded donation flow	A publishable key (`pk_...`) and the donation endpoints
Reward users for purchases, volunteering, referrals, or other actions	A secret key (`sk_...`), HMAC signing, and the actions endpoints
Listen for reward status changes	Webhooks
Test before launch	Sandbox credentials and a sandbox token pool

What you need before you call the API

You need three things:

1. A partner account.
2. An API key pair for the environment you are using.
3. A funded token pool if you want to distribute SIR tokens.

Creating credentials lets you authenticate. It does not automatically mean you can issue real rewards. Rewards draw from a token pool, and production token pools require SIR Giving approval.

Base URLs

Environment	Base URL	Key prefix
Sandbox	`https://devapi.sirgiving.org`	`pk_test_...`, `sk_test_...`
Production	`https://api.sirgiving.org`	`pk_live_...`, `sk_live_...`

All partner integration endpoints are under `/v1/partner/`.

Interactive API docs are available at `/partner-api` on each host.

Choose your first tutorial

Goal	Use this
I want to make my first signed backend request	Authentication & HMAC Signing
I want to add a donation button	End-to-End Scenarios: Donation widget
I want to reward a user for an action	End-to-End Scenarios: Reward a user
I want to receive events	Webhooks

Key concepts

Partner

Your organization or developer account in SIR Giving. API keys, token pools, campaigns, users, and webhooks all belong to a partner.

API key pair

Each key issuance returns:

- A publishable key, such as `pk_test_...`, safe for browser use.
- A secret key, such as `sk_test_...`, for backend use only.
- An HMAC signing secret, used to sign server-to-server requests.

The secret key and HMAC secret are shown once. Store them in a secret manager.

Token pool

A funded bucket of SIR tokens allocated to your partner. Reward actions debit this pool. If there is no active pool in the same environment as your API key, action submission can authenticate but still fail during processing.

Action

The unit of reward work you send to SIR Giving. For example: a purchase, volunteer shift, donation, referral, or advocacy event. Actions include an `idempotencyKey` so retries do not duplicate rewards.

Partner user

A user from your system mirrored into SIR Giving using your stable `externalUserId`. Partner users receive balances when actions reward them.

Webhook

An outbound event sent from SIR Giving to your server when important things happen, such as `action.completed` or `token_pool.low_balance`.

The shortest path to a working integration

1. Create or receive sandbox credentials.
2. Store the `sk_...` key and HMAC secret in your backend environment.
3. Make a signed `GET /v1/partner/users` request.
4. If you are building a widget, create a donation link with the `pk_...` key.
5. If you are issuing rewards, confirm you have an active sandbox token pool.
6. Submit a test action with a unique `idempotencyKey`.
7. Register a webhook and send a test event.
8. Repeat the same flow in production after your production token pool is approved.

Common confusion

My key works, but actions fail. Why?

Authentication only proves your key is valid. Reward actions also need an active token pool in the same environment.

Can I use a publishable key for server actions?

No. Mutation endpoints such as action submission and webhook registration require a secret key.

Can I use sandbox keys against production?

Do not do this. Use `test` keys with `devapi.sirgiving.org` and `live` keys with `api.sirgiving.org`.

Revision #4
Created 2026-05-10 09:11:09 UTC by krtin shet
Updated 2026-05-30 13:31:54 UTC by krtin shet