Acme Widget API

Version 2.1.0

Acme Engineering Team

January 15, 2025

Table of Contents

Introduction

The Acme Widget API provides a RESTful interface for creating, managing, and monitoring widgets. This document serves as the complete reference for integrating with the platform.

Who Should Read This

This guide is intended for:

- Backend engineers integrating widgets into their services
- Frontend developers building widget-powered UIs
- DevOps teams configuring webhook delivery
- Technical writers maintaining API documentation

Authentication

API Keys

All requests must include an API key in the `Authorization` header:

```
Authorization: Bearer sk-xxxx-your-key-here
```

Keys are scoped to a single project. Generate and manage keys from the developer dashboard.

Key Rotation

API keys should be rotated every 90 days. The rotation process is zero-downtime: both the old and new key remain valid for a 24-hour grace period.

Endpoints

List Widgets

```
GET /v1/widgets?status=active&page=1&per_page=20
```

Returns a paginated list of widgets. Supports filtering by `status` (`active`, `paused`, `archived`) and sorting by `created_at` or `name`.

Create Widget

```
POST /v1/widgets
Content-Type: application/json
```

Request body:

```
{
  "name": "Weather Widget",
  "config": {
    "refresh_interval": 300,
    "theme": "dark"
  }
}
```

Get Widget

```
GET /v1/widgets/{widget_id}
```

Returns the full widget object including configuration, version history, and current status.

Update Widget

```
PATCH /v1/widgets/{widget_id}
```

Accepts a partial update. Only the fields included in the request body are modified; all other fields remain unchanged.

Delete Widget

```
DELETE /v1/widgets/{widget_id}
```

Permanently deletes a widget and all associated data. This action cannot be undone.

Error Handling

The API returns errors in a consistent JSON format:

```
{
  "error": {
    "code": "WIDGET_NOT_FOUND",
    "message": "No widget found with ID wgt_abc123.",
    "status": 404
  }
}
```

Common error codes:

- `INVALID_API_KEY` — the key is missing or malformed.
- `RATE_LIMIT_EXCEEDED` — you've exceeded your plan's limit.
- `WIDGET_NOT_FOUND` — the specified widget does not exist.
- `VALIDATION_ERROR` — the request body is invalid.

Best Practices

Pagination

Always paginate list requests. The default page size is 20, and the maximum is 100. Use the `next_cursor` field in the response to fetch subsequent pages.

Idempotency

For `POST` requests, include an `Idempotency-Key` header to safely retry failed requests without creating duplicates.

Webhooks

Configure webhooks in the dashboard to receive real-time notifications when widget status changes. Webhook payloads are signed with HMAC-SHA256 for verification.