Webhooks done right
Nicolas Grenié
Developer Advocate @ Typeform
26 sept 2018 - API Strategy Nashville
Nicolas Grenié
@picsoung
Let’s go fishing
Spiders do it too
Let’s agree on something.

Webhook

Something happens on a platform and it will let you know about it.
API Strategy:
Start with webhook
Thanks to webhook you can integrate with 1000+ services.

zapier

stamplay

autopilot™
What should happen on provider’s side?
Webhooks allow external services to be notified when certain events happen within your form.

Learn more here

Destination URL

https://tf-mixpanel-webhook.glitch.me/hook?mix

Test Webhook
#2 Let me inspect payload
Webhooks

Webhooks allow external services to be notified when certain events happen within your form. Learn more here.

Destination URL

https://tf-mixpanel-webhook.glitch.me/hook?mix

Test Webhook

#3 Events history

<table>
<thead>
<tr>
<th>Recent Requests</th>
<th>Status</th>
<th>Payload ID</th>
<th>Time</th>
<th>Date</th>
<th>Request Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>✓</td>
<td>01CQN0GC51D0XYN8J90BMPY6K</td>
<td>19:15:29</td>
<td>2018-09-17</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>01CQN0DY9QE96Y8J5FHQSFWPR</td>
<td>19:14:10</td>
<td>2018-09-17</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>01CQNZXN68Q7AK3J8W2ZWYHE</td>
<td>13:37:49</td>
<td>2018-09-14</td>
<td></td>
</tr>
</tbody>
</table>
#4 Replay event

**Recent Deliveries**

- Request: 200
- Response: 200
- Completed in 0.07 seconds.

**Headers**

- Request method: POST
- Content-Type: application/json
- Expect:
  - User-Agent: GitHub-Hookshot/2ccd623
  - X-GitHub-Delivery: a7f1f190-92e4-11e8-957a-a9717af5822
  - X-GitHub-Event: installation
  - X-Hub-Signature: sha1=65b0fc0ff13c76de2c57cb1047493a5752e33176

**Payload**

```json
{
  "action": "created",
  "installation": {
    "id": 264702,
    "account": {
      "login": "picsoung",
      "id": 172072,
      "node_id": "MDQ6VXNlcjE3MjA3Mg==",
      "gravatar_id": "",
      "url": "https://api.github.com/users/picsoung",
      "html_url": "https://github.com/picsoung",
      "followers_url": "https://api.github.com/users/picsoung/followers",
      "following_url": "https://api.github.com/users/picsoung/following{/other_user}",
      "repos_url": "https://api.github.com/users/picsoung/repos",
      "events_url": "https://api.github.com/users/picsoung/events{/privacy}",
      "received_events_url": "https://api.github.com/users/picsoung/received_events",
      "type": "User",
      "site_admin": "false"
    }
  
  "created_at": "2018-07-28T23:36:34Z",
  "commit": {
    "id": "9d8615f33fe18b10f44f913b65e402f56796549a",
    "OID": "51f16d3aa7a85367a2a0f69f2ce2f83430f95560",
    "message": "install gh-hookshot"
  }

  "actor": {
    "login": "picsoung",
    "id": 172072,
    "node_id": "MDQ6VXNlcjE3MjA3Mg==",
    "gravatar_id": "",
    "url": "https://api.github.com/users/picsoung",
    "html_url": "https://github.com/picsoung",
    "followers_url": "https://api.github.com/users/picsoung/followers",
    "following_url": "https://api.github.com/users/picsoung/following{/other_user}",
    "repos_url": "https://api.github.com/users/picsoung/repos",
    "events_url": "https://api.github.com/users/picsoung/events{/privacy}",
    "received_events_url": "https://api.github.com/users/picsoung/received_events",
    "type": "User",
    "site_admin": "false"
  }

  "client_payload": null
}
```
Shopify has implemented a 5-second timeout period and a retry period for subscriptions. We wait 5 seconds for a response to each request, and if there isn’t one or we get an error, we retry the connection to a total of 19 times over the next 48 hours. A webhook will be deleted if there are 19 consecutive failures for the exact same webhook. You should monitor the admin of your webhook tool for failing webhooks.

In live mode, we will attempt to deliver your webhooks for up to three days with an exponential back off. In test mode, we retry three times over a few hours. Webhooks cannot be manually retried after this time, though you can [query for the event API](#) to reconcile your data with any missed events.
#6 Notifications

Your Nylas webhook needs attention

Nylas hello@nylas.com via mailbot.inboxapp.com to me

Sun, Aug 26, 10:14 PM

Your webhook needs attention!

It looks like your webhook endpoint at has issues processing the webhooks we've been sending. Unfortunately, if this keeps happening over the next 6 hours, we'll have to disable your webhook. If your webhook does get disabled, don't worry! We keep missed webhooks for up to 15 days – that means that as long as you re-activate your webhook soon, you will not lose any data. Otherwise, you may have to resync data manually through our API.

Webhook Id: 9qpp60ngdtoic8f8q1pf7d0me
Callback Url: https://tranquil-spot.glitch.me/nylas/webhook

Please make sure that your callback url is up and running. For us to consider it an active callback, it should be returning an http status of 200 when we send POST requests to it.

If your webhook does get disabled, you can re-activate it through the "Webhooks" section of the dashboard by clicking on the toggle switch that says "Inactive" or "Failed". The webhook will be re-activated once the callback confirms our verification request.

If you have any questions, please contact us at support@nylas.com.
#7 Sign webhook payload

App Credentials

These credentials allow your app to access the Slack API. They are secret. Please don't share your app credentials with anyone, include them in public code repositories, or store them in insecure ways.

<table>
<thead>
<tr>
<th>App ID</th>
<th>Date of App Creation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>March 21, 2018</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Client ID</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Client Secret</th>
</tr>
</thead>
</table>

You'll need to send this secret along with your client ID when making your `oauth.access` request.

Signing Secret

<table>
<thead>
<tr>
<th>Signing Secret</th>
</tr>
</thead>
</table>

Slack signs the requests we send you using this secret. Confirm that each request comes from Slack by verifying its unique signature.

Verification Token

| Verification Token |

This deprecated Verification Token can still be used to verify that requests come from Slack, but we strongly recommend using the above, more secure, signing secret instead.

https://api.slack.com/docs/verifying-requests-from-slack
#8 Webhook API

- Let me create/change Webhook via API
- Add corresponding oAuth Scopes
- Manage deliveries
#9 More than one webhook

- Specialize each webhook
- Build complex workflow
- Enable more integration
## #10 Many event types

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Any time any event is triggered (Wildcard Event).</td>
</tr>
<tr>
<td>check_run</td>
<td>Any time a check run is created, requested, or rerequested.</td>
</tr>
<tr>
<td>check_suite</td>
<td>Any time a check suite is completed, requested, or rerequested.</td>
</tr>
<tr>
<td>commit_comment</td>
<td>Any time a Commit is commented on.</td>
</tr>
<tr>
<td>create</td>
<td>Any time a Branch or Tag is created.</td>
</tr>
<tr>
<td>delete</td>
<td>Any time a Branch or Tag is deleted.</td>
</tr>
<tr>
<td>deployment</td>
<td>Any time a Repository has a new deployment created from the API.</td>
</tr>
<tr>
<td>deployment_status</td>
<td>Any time a deployment for a Repository has a status update from the API.</td>
</tr>
<tr>
<td>fork</td>
<td>Any time a Repository is forked.</td>
</tr>
<tr>
<td>github_app_authorization</td>
<td>Any time someone revokes their authorization of a GitHub App. GitHub Apps receive this webhook by default and cannot unsubscribe from this event.</td>
</tr>
<tr>
<td>gollum</td>
<td>Any time a Wiki page is updated.</td>
</tr>
</tbody>
</table>
Will serverless functions kill Webhooks?
Serverless is the future of webhooks
Takeaways
Takeaways

Build Webhook first

Dedicated DX effort

Extend with native serverless