Integrations

Keavy McMinn // Engineer
“Who opens PRs?”
GET /repos/:owner/:repo/pulls
Grants read/write access to code, commit statuses, repository invitations, collaborators, and deployment statuses.
Grants read/write access to code, commit statuses, repository invitations, collaborators, and deployment statuses for public and private repositories.
Grants **read/write** access to code, commit statuses, repository invitations, collaborators, and deployment statuses for **public and private** repositories and organizations.
User visits Integrator site

Authorize Integrator with OAuth

Integrator receives and stores OAuth access token for user

List accessible orgs for user

List accessible repos for user

Display list of repos to user

User selects repos to build

Set up hooks, create keys for selected repos
All checks have passed
4 successful checks

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request
You can also open this in GitHub Desktop or view command line instructions.
Uh. Where's CI?

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request  You can also [open this in GitHub Desktop](https://desktop.github.com) or view [command line instructions](https://help.github.com).
“We need this level of access because GitHub…”
Hey Super ⭐,

Any chance for more granular permission scopes for OAuth / APIs?

I'd like to give a bot the ability to create repos ... and that's it. Right now it seems like I've got to give the bot access to the `repo` scope (scary!) to do this.

<3, you pal §OH☺
=
I’ve got 99 problems, and they’re all OAuth
Scopes
Outside-in flow
Install flow efficiency
No Connection between an org and a service
Multiple accounts
“If we were building integrations from scratch, knowing what we know now… what would we do differently?”
Integrations

🚧 Early Access 🚧
Users
CLA App 2

CLA App 2 is provided by a third-party and is governed by separate terms, privacy, and support documentation. The GitHub logo, Inertial logo, the images and other content are trademarks or copyrights of GitHub, Inc. All other trademarks are the property of their respective owners.

Install

Developer
@api-playground

Homepage
https://github.com/github/platform/issues/320
Where would you like to install CLA App 2?

- keavy
- api-playground
- github
- github-beta
- project-moonstar

Install >
Install CLA App 2

Install on these repositories

- All repositories
  This applies to all current and future repositories.

- Only select repositories

With these permissions:

- Read access to code
- Read access to issues and pull requests
- Read and write access to commit statuses

Events
CLA App 2 will receive events for pull requests.

Install Cancel
Install CLA App 2
on your organization api-playground

Install on these repositories
- All repositories
  This applies to all current and future repositories.
- Only select repositories
  sand

With these permissions:
- Read access to code
- Read access to issues and pull requests
- Read and write access to commit statuses

Events
CLA App 2 will receive events for pull requests.

Install  Cancel
Alright, CLA App 2 was installed on the @api-playground account.

Repository access

You can edit repositories that CLA App 2 is installed on.

- All repositories
  - This applies to all current and future repositories.

- Only select repositories
  - [ ] Select repositories

Selected 1 repository

api-playground/api-sandbox

Save  Cancel
Integrator
Register new integration

Integrations can be installed directly on organization or user accounts. Users select which repositories can be accessed and at what permission level. Integrations can perform actions directly and can replace bot or service accounts.

Integration name

`secret-sauce`

The name of your integration.

Description (optional)

This is displayed to all potential users of your integration.

Homepage URL

`http://secret-sauce.com`

The full URL to your integration’s website.

Callback URL (optional)

`http://secret-sauce.com/callback`

The callback URL when fetching identity tokens

Webhook URL

`http://secret-sauce.com/webhook`

Events will POST to this URL. Read our webhook documentation for more information.

Webhook secret (optional)

`**********`

Read our webhook secret documentation for more information.
<table>
<thead>
<tr>
<th>Integration</th>
<th>Description</th>
<th>Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commit statuses</td>
<td>Commit statuses.</td>
<td>Read &amp; write</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Status</td>
<td>Commit status updated from the API</td>
<td></td>
</tr>
<tr>
<td>Deployments</td>
<td>Deployments and deployment statuses.</td>
<td>No access</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deployment</td>
<td>Repository deployed.</td>
<td></td>
</tr>
<tr>
<td>Deployment status</td>
<td>Deployment status updated from the API</td>
<td></td>
</tr>
<tr>
<td>Issues</td>
<td>Issues and related comments, assignees, labels, and milestones.</td>
<td>Read &amp; write</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Issue comment</td>
<td>Issue comment created, edited, or deleted.</td>
<td></td>
</tr>
<tr>
<td>Issues</td>
<td>Issue opened, assigned, unassigned, labeled, or unlabeled</td>
<td></td>
</tr>
<tr>
<td>Organization members</td>
<td>Organization members and teams.</td>
<td>Read-only</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Membership</td>
<td>Team membership added or removed</td>
<td></td>
</tr>
<tr>
<td>Team add</td>
<td>Team added or modified on a repository</td>
<td></td>
</tr>
<tr>
<td>Pages</td>
<td>Retrieve Pages statuses, configuration, and builds, as well as create new builds.</td>
<td>No access</td>
</tr>
<tr>
<td>Pull requests</td>
<td>Pull requests and related comments, assignees, labels, milestones, and merges.</td>
<td>Read-only</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pull request</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Pages
- Retrieve Pages statuses, configuration, and builds, as well as create new builds.

### Pull requests
- Pull requests and related comments, assignees, labels, milestones, and margins.
  - **Available events**
    - Pull request
      - Pull request opened, closed, reopened, edited, assigned, unassigned, labeled, unlabeled, or synchronized.
  - Pull request review comment
    - Pull request diff comment created, edited, or deleted.

### Repository contents
- Repository contents, commits, branches, downloads, releases, and merges.
  - **Available events**
    - Commit comment
      - Commit or diff commented on.
    - Delete
      - Branch or tag deleted.
    - Push
      - Git push to a repository.
    - Create
      - Branch or tag created.
    - Fork
      - Repository forked.
    - Release
      - Release published in a repository.

---

**Where can this integration be installed?**

- **Only on this account**
  - Only allow this integration to be installed on the api-playground account.
- **Any account**
  - Allow this integration to be installed by any user or organization.

![Create Integration Button](create-integration-button.png)
tarebyte deployed to production 4 days ago

Hubot Deployer triggered deploy on behalf of @tarebyte
Thorny areas
Checking permissions
Checking permissions

- REPO
- ISSUES
- READ?
- WRITE?
Checking permissions
Checking permissions
Checking permissions

# A user with push access to commit status

role :status_writer do |context|
  ...
  user && repo.pushable_by?(user)
end
Checking permissions

# A user with push access to commit status

role :status_writer do |context|
...
  user &&
  (repo.pushable_by?(user) || repo.statuses.writeable_by?(user))
end
Checking permissions

# A user with push access to commit status

role :status_writer do |context|
  ...
  user && repo.resourcesstatuses.writeable_by?(user)
end
Identity flow
Identity flow
Identity flow
Possible solutions:

OAuth
Possible solutions:

Single Sign-On
OAuth-like flow

1. To Callback
2. Authorization Request
3. Authorization Response
4. Request Identity of User
5. User Response
Led by UX
## Objective

+ Make it easier for users to configure and use third-party services with GitHub.
Trust
**Spike: application permission for all repos #43852**

Closed keavy wants to merge 66 commits into application-actor-spke from installation-all-repos-spke

keavy commented on Jul 21, 2015

Continued research for Integrations 2.0. This PR looks at a research TODO of permission to “all repositories” (i.e., every repo that the org owns today, and will create in the future).

---

**Spike: Perform action as a "application actor" #42476**

Closed jasonrudolph wants to merge 19 commits into master from application-actor-spke

jasonrudolph commented on Jun 5, 2015

As part of Integrations 2.0 (github/platform#189), this PR attempts to perform a single action as an "application actor" (AKA "system actor" or "application bot account"). Since creating a commit is a common thing we expect application actors to do, we'll use that action for this spike.

The intent of this PR is to learn about the problem space and identify technical challenges.

---

**[Spike] Installing an Integrations 2 app on a repo #46688**

Closed keavy wants to merge 8 commits into master from installing-app-on-repo

keavy commented on Oct 7, 2015

This PR is a starting point for a conversation. The code is not intended for production.
Alternatives now
Grants read/write access to code, commit statuses, repository invitations, collaborators, and deployment statuses for public and private repositories and organizations.
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commit statuses</td>
<td>Commit statuses.</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
</tr>
<tr>
<td>Status</td>
<td>Commit status updated from the API.</td>
</tr>
<tr>
<td>Deployments</td>
<td>Deployments and deployment statuses.</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
</tr>
<tr>
<td>Deployment</td>
<td>Repository deployed.</td>
</tr>
<tr>
<td>Deployment status</td>
<td>Deployment status updated from the API.</td>
</tr>
<tr>
<td>Issues</td>
<td>Issues and related comments, assignees, labels, and milestones.</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
</tr>
<tr>
<td>Issue comment</td>
<td>Issue comment created, edited, or deleted.</td>
</tr>
<tr>
<td>Issues</td>
<td>Issue opened, assigned, unassigned, labeled, or unlabeled.</td>
</tr>
<tr>
<td>Organization members</td>
<td>Organization members and teams.</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
</tr>
<tr>
<td>Membership</td>
<td>Team membership added or removed.</td>
</tr>
<tr>
<td>Team add</td>
<td>Team added or modified on a repository.</td>
</tr>
<tr>
<td>Pages</td>
<td>Retrieve Pages statuses, configuration, and builds, as well as create new builds.</td>
</tr>
<tr>
<td>Pull requests</td>
<td>Pull requests and related comments, assignees, labels, milestones, and merges.</td>
</tr>
<tr>
<td>Available events</td>
<td></td>
</tr>
<tr>
<td>Pull request</td>
<td>Pull request review comment</td>
</tr>
</tbody>
</table>
User visits Integrator site

Authorize Integrator with OAuth

Integrator receives and stores OAuth access token for user

List accessible orgs for user

List accessible repos for user

Display list of repos to user

User selects repos to build

Set up hooks, create keys for selected repos
User visits integration page

User installs integration on selected repos

Integrator receives webhook event of new installation
Octokit::Client.new(  :access_token => installation_access_token )
When to use an Integration?

- Take action independently of a specific user
- Allow users to install on an organization
- Allow users to install on a per repository basis
- Want granular permissions
What might you use an integration for?

- Code Scanners
- Continuous Integration
- Continuous Deployment
- Issue Management

Anything else you can dream up!
When to use an OAuth application?

Take action only as a specific user

Full access is desired

Using GitHub as an identity provider
Learn more

developer.github.com

platform.github.community
Thank you