

Deploy two separate heroku apps from same git repo

[Ask Question](#)

Within one git repo, I have two separate applications (web server and API server).

How can I deploy each application to its own Heroku app?

(So there are 2 heroku apps, one for the web server and one for the api server)

Note (before marking as duplicate): There are several questions similar to this. Most deal with deploying **one** app to **two** heroku apps - typically for the purpose of staging vs. production. I am looking to deploy **two** apps to **two** heroku apps. ([Question about staging vs prod](#))

[git](#) [heroku](#)

edited May 23 '17 at 12:25



Community ♦

1 1

asked Jan 4 '17 at 10:25



[Don P](#)

15.6k 63 173 291

2 Answers

My understanding of your question is that you have one Git repository, that contains two entirely separate programs: one API server, and one web server.

With this assumption in mind, here's what you'll want to do, step-by-step:

1. Go into your project folder.
2. Define a `Procfile` at the root of your project. This will tell Heroku how to run your web server and your API server.

Here's how you might want your `Procfile` to look (an example):

```
web: node web/index.js
api: node api/index.js
```

In my example above: I'm defining two types of Heroku dynos -- one called `web` and one called `api`. For each one, you'll need to tell Heroku what command to run to start the appropriate server. In this

example, I would run `node web/index.js` to start up my website, and `node api/index.js` to start up my API service.

3. Create two new Heroku applications. You can do this by running `heroku create <desired-app-name> --remote <desired-app-name>` multiple times. **NOTE:** The `--remote` flag will tell Heroku to create a Git remote for each of your applications in the same repo.

4. Next, you'll need to tell Heroku to run your actual web application on one Heroku app, and your API service on another Heroku app. You can do this by using the Heroku CLI:

```
$ heroku ps:scale web=1 --remote webserver-app-name
$ heroku ps:scale api=1 --remote apiserver-app-name
```

These commands will:

- Run a single web dyno for your webserver Heroku app.
- Run a single API dyno for your apiserver Heroku app.

As you can see above, using the `ps:scale` command you can control what types of commands Heroku will run from your `Procfile`, and how many instances of each you'd like to have.

Hopefully this helps!

answered Jan 4 '17 at 14:25



[rdegges](#)

15.8k 10 45 77

Thanks rdegges, I'll give this a try today. Question though - will I just do `git push heroku-api master` and `git push heroku-web master` to push my master branch to each separate heroku app? – [Don P](#) Jan 4 '17 at 18:31

1 Ah, when you want to push your mast branch you'll do: `git push heroku-api master`; `git push heroku-web master`; to deploy it to both projects =) Instead of the usual `git push heroku master` you'll do this, because instead of the `heroku remote` you now have two remotes: `heroku-web` and `heroku-api`. Hope that makes sense! – [rdegges](#) Jan 4 '17 at 18:58

This is great, but you need to somehow install dependencies too. I did that by putting `npm install` in the script to run my apps from subfolders. Isn't perfect, but I don't know how to do it on push. – [Piotr Buda](#) May 23 '17 at 15:09

You don't have to do that. It will install them automatically upon push. – [rdegges](#) May 23 '17 at 15:32

3 @rdegges This doesn't work for me because Heroku says there is no `web process type`. How can I tell Heroku that for example, `api` is my web process type for an app? – [Sergio Tapia](#) Sep 5 '17 at 19:37

The [solution suggested by rdegges](#) unfortunately does not work anymore. See:

The web process type is special as it's the only process type that will receive HTTP traffic from Heroku's routers. Other process types can be named arbitrarily.

from the [Heroku documentation](#). So you won't be able to have `api` and `web` in a Procfile both exposing web apps.

I believe the correct way to tackle this is to use this buildpack provided by the Heroku team: [Heroku Multi Procfile buildpack](#):

Imagine you have a single code base, which has a few different applications within it... or at least the ability to run a few different applications. Or, maybe you're Google with your mono repo?

In any case, how do you manage this on Heroku? You don't. Heroku applications assume one repo to one application.

Enter the Multi Procfile buildpack, where every app gets a Procfile!

Edit: I've been using this buildpack for some months now on a repository using yarn workspaces (multiple Node and React apps in one repo) and everything works fine.

[edited Apr 10 at 10:12](#)

answered Jan 31 at 17:19



[laugri](#)

193 1 11