

automated deployments from github

with circleci & ansible

<https://github.com/shortstack/circleci-demo-ermahgerd>

the plan

1. push to github
2. build in circleci
3. deploy via awscli *OR* circleci + ansible

srsly -->



things you'll need

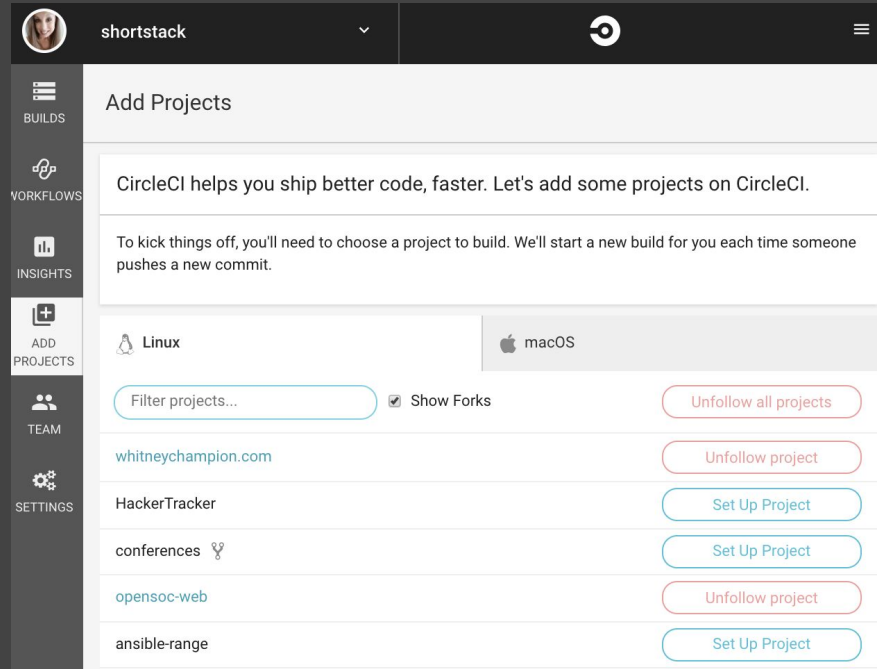
- github account
- static content in a repo in said github account
- circleci account (can auth with github)
- AWS access keys
 - if deploying to AWS*
- SSH key pair
 - if deploying to a server*

pick one



set up circleci

- log into circleci → add projects → set up project
- pick OS and language
- make a .circleci directory in your repo
- touch .circleci/config.yml
- git commit -a
- git push origin master
- click start building
- FYI, this will probably fail
- BUT THAT'S OK! \o/



The screenshot shows the CircleCI web interface for a user named 'shortstack'. The main heading is 'Add Projects'. Below this, there is a message: 'CircleCI helps you ship better code, faster. Let's add some projects on CircleCI. To kick things off, you'll need to choose a project to build. We'll start a new build for you each time someone pushes a new commit.' There are two tabs for operating systems: 'Linux' (selected) and 'macOS'. Below the tabs, there is a search bar 'Filter projects...' and a checkbox 'Show Forks'. A list of projects is displayed with buttons to 'Unfollow all projects', 'Unfollow project', or 'Set Up Project'.

Project Name	Action
whitneychampion.com	Unfollow project
HackerTracker	Set Up Project
conferences	Set Up Project
opensoc-web	Unfollow project
ansible-range	Set Up Project

if using S3...

configure aws

- if you already have access keys with at least S3 permissions, skip this
- if you don't...
 - log into AWS IAM → users
 - add user
 - name + programmatic access
 - permissions → “attach existing policies”
 - S3 full access
 - create user
 - download keys, keep somewhere safe

configure aws in circleci

- project settings
- AWS permissions
- add access keys



deploying to S3

- make an S3 bucket
- properties → static website → use this to host a website



deploying to S3

config.yml

```
1  version: 2
2  jobs:
3    deploy:
4      docker:
5        - image: circleci/python:2.7
6      working_directory: ~/circleci-demo-ermahgerd
7      steps:
8        - checkout
9        - run:
10           name: install awscli
11           command: sudo pip install awscli
12        - run:
13           name: remove .git and .circleci
14           command: rm -rf .git && rm -rf .gitignore && rm -rf .circleci
15        - run:
16           name: deploy to S3
17           command: aws s3 sync . s3://ermahgerd --region us-east-1 --acl public-read
18 workflows:
19   version: 2
20   build-deploy:
21     jobs:
22       - deploy:
23         filters:
24           branches:
25             only: master
26
```

if using your own server
instead of S3...

add user & ssh key

- ssh you@server

```
sudo useradd circleci
sudo chown circleci:circleci /var/www/html
su - circleci
ssh-keygen -t rsa -b 4096
cat ~/.ssh/id_rsa.pub > ~/.ssh/authorized_keys
chmod 644 ~/.ssh/authorized_keys
```



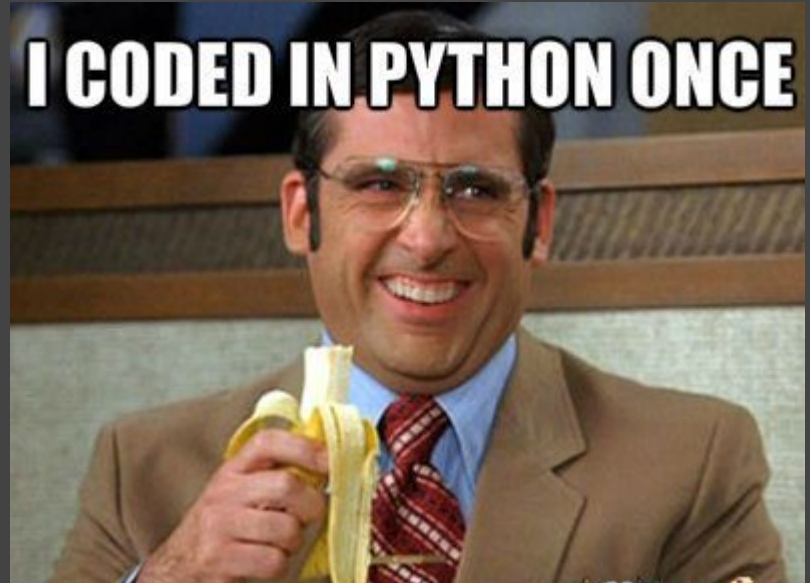
configure ssh

- circleci → project settings
- SSH permissions
- add SSH key
 - set hostname
 - paste private key (.ssh/id_rsa contents)



install python

- `sudo apt or yum install python -y`



environment variables

- circleci → project settings
- environment variables
- add variable
 - ANSIBLE_HOSTS = ~/repo-name/.circleci/hosts
 - ANSIBLE_HOST_KEY_CHECKING = False



configure ansible

- this is your ansible inventory
- `.circleci/hosts`

```
1 [hostname]
2 IP address
3
4 |
```

deploying to a server

- this assumes you already have a web server running
 - install web server package(s)
 - start and enable web server
 - locate default directory to put content in
 - `chown circleci:circleci /directory`

deploying to a server

config.yml

```
1  version: 2
2  jobs:
3    deploy:
4      docker:
5        - image: circleci/python:2.7
6      working_directory: ~/circleci-demo-ermahgerd
7      steps:
8        - checkout
9        - run:
10           name: install ansible
11           command: sudo pip install ansible
12        - run:
13           name: install rsync
14           command: sudo apt install -y rsync
15        - run:
16           name: deploy via ansible
17           command: ansible-playbook .circleci/deploy.yml
18
19 workflows:
20   version: 2
21   build-deploy:
22     jobs:
23       - deploy:
24         filters:
25           branches:
26             only: master
27
```

ansible playbook - deploy.yml

```
1  ---
2  - hosts: 167.99.148.100
3    become: yes
4    tasks:
5
6      - name: rsync files
7        synchronize:
8          src: ~/circleci-demo-ermahgerd/
9          dest: /var/www/html/
10         rsync_opts:
11           - "--exclude=.git --exclude=.circleci"
12
```

now what?!

let's test it out

- git commit & git push
- go to circleci dashboard
- SEE WHAT HAPPENS



the end <3