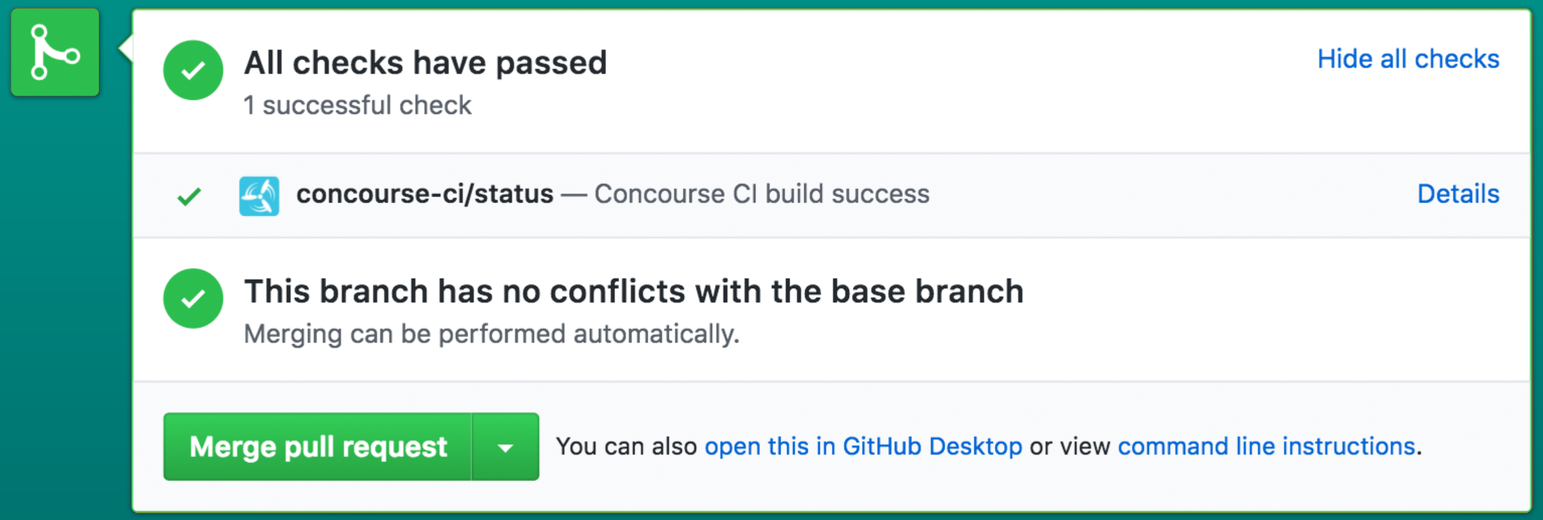
Testing Pull Requests With Concourse

Friday, April 20


All checks have passed
1 successful check
Hide all checks
concourse-ci/status — Concourse CI build success
Details
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.

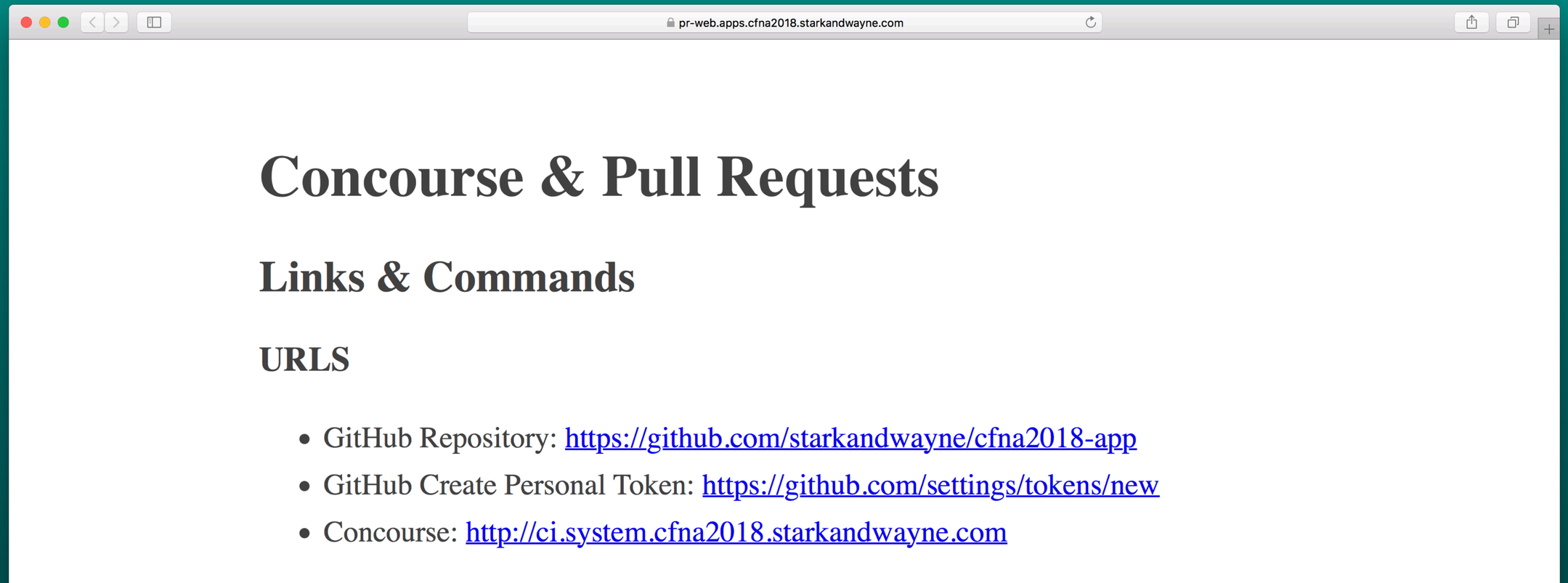
All checks have passed

1 successful check

- It’s already likely that you’re using Concourse for testing or deploying to various lifecycles
- Either add the ability to test pull requests into existing pipelines, or create your own
- We’ll talk about adding PR testing functionality to an existing pipeline

Let's *Git* Setup, *Shell* We?

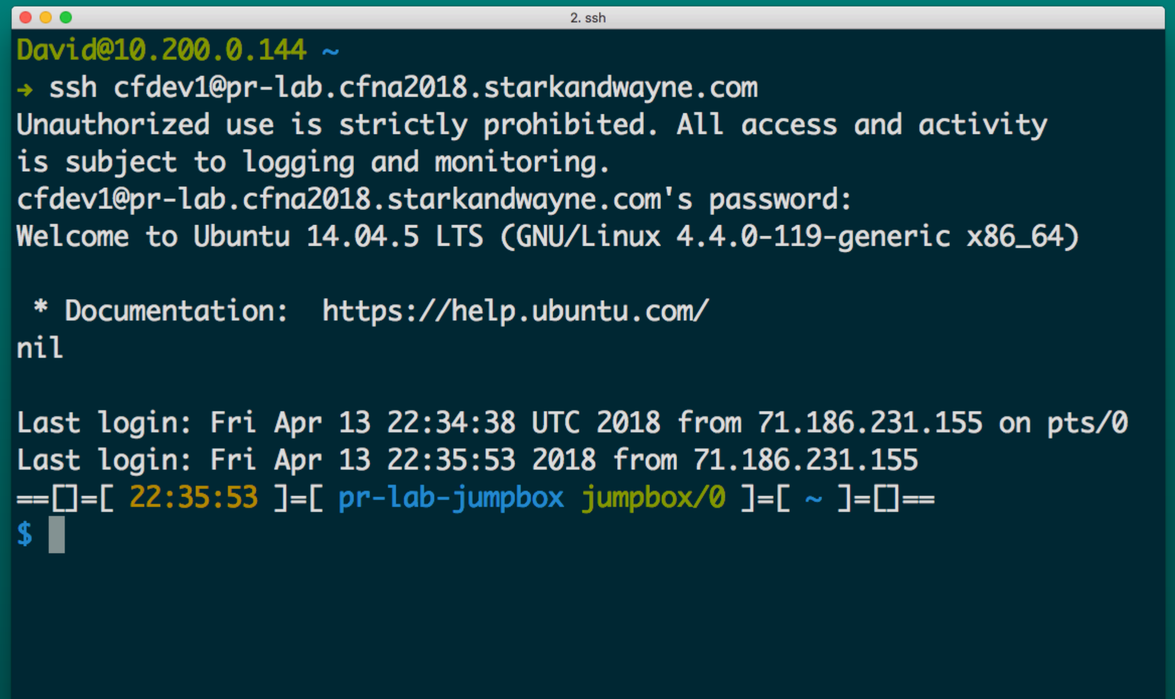
`https://bit.ly/2HcgnnJ`



Let's *Git* Setup, *Shell* We?

`pr-lab1.cfna2018.starkandwayne.com`

`cfna2018`

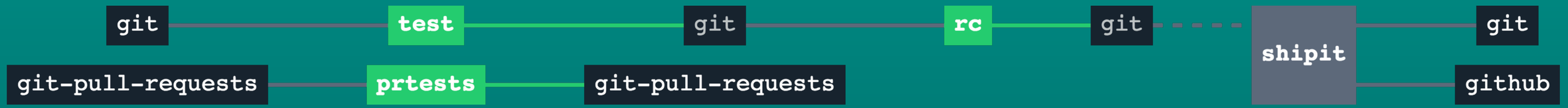


Let's *Git* Setup, *Shell* We?

`ci`

`cfna2018`

git
test
git
rc
git
shipit
git
github


git
test
git
rc
git
shipit
git
git-pull-requests
prtests
git-pull-requests
github

```
resource_types:
- name: pull-request
  type: docker-image
  source:
    repository: jtarchie/pr
```

```
resources:
- name: git-pull-requests
  type: pull-request
  source:
    repo: (( concat meta.github.owner "/" meta.github.repo ))
    access_token: (( grab meta.github.access_token ))
```

```yaml
- name: prtests
  plan:
    - get: git-pull-requests
      trigger: true
      version: every
      params:
        path: git-pull-requests
        status: pending
```

```yaml
      on_success:
        put: git-pull-requests
        params:
          path: git-pull-requests
          status: success
      on_failure:
        put: git-pull-requests
        params:
          path: git-pull-requests
          status: failure
```

```yaml
- name: prtests
  plan:
    - get: git-pull-requests
      trigger: true
      version: every
      params:
        path: git-pull-requests
        status: pending
    - task: test
      config:
        platform: linux
        image_resource:
          type: docker-image
          source:
            repository: (( grab meta.image.name ))
            tag:        (( grab meta.image.tag ))
        inputs:
          - name: git-pull-requests
            path: code
        run:
          path: ./code/ci/scripts/test
          args: []
        params:
          MODULE: (( grab meta.go.module ))
      on_success:
        put: git-pull-requests
        params:
          path: git-pull-requests
          status: success
      on_failure:
        put: git-pull-requests
        params:
          path: git-pull-requests
          status: failure
```

STARK & WAYNE