

# Exercise 4

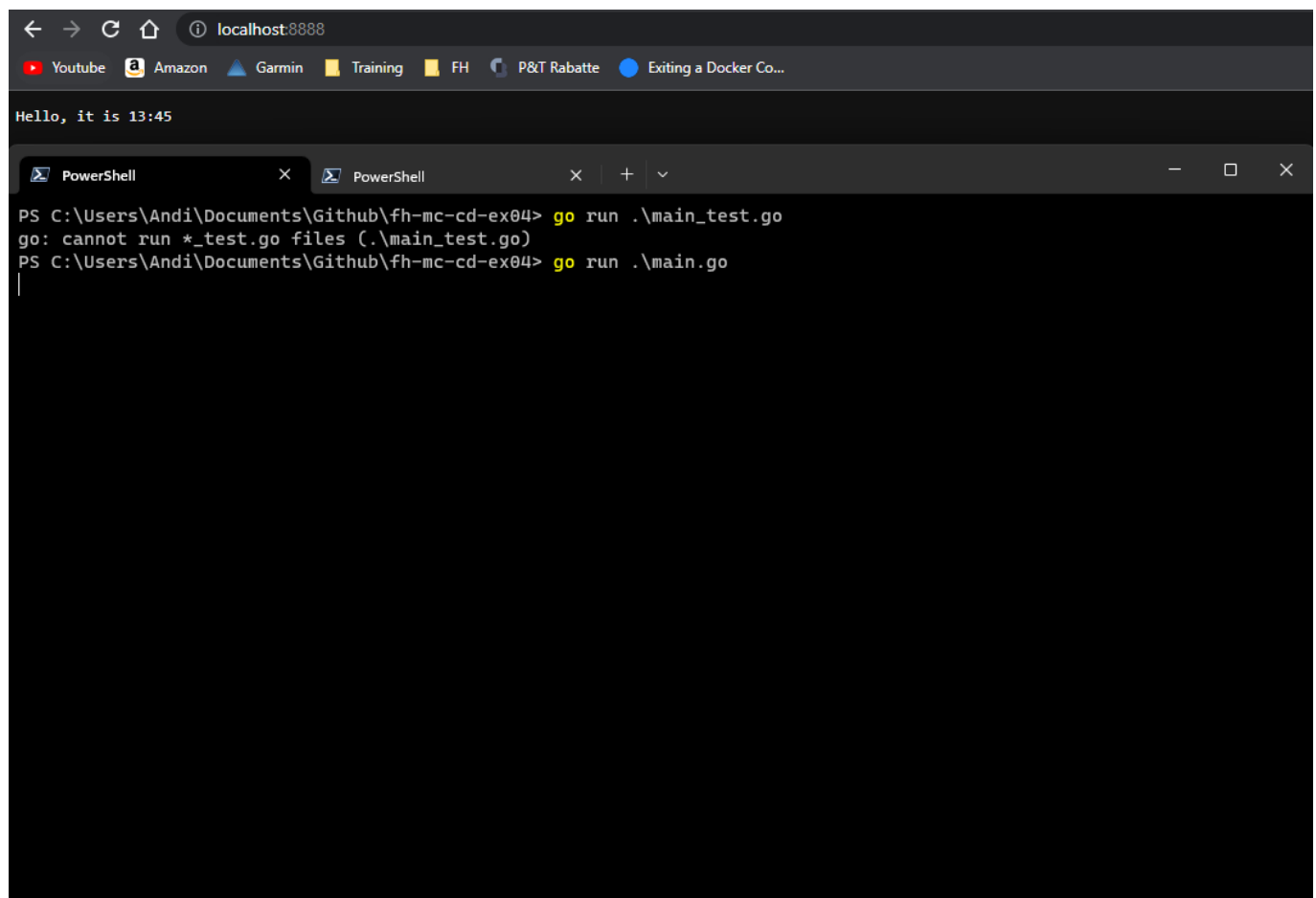
---

## Part 1

### 1. Check if the go program runs locally

First create a go.mod file (go init [NAME]).

```
go run main.go
```



### 2. Modify the dockerfile in the repo

```
FROM golang:1.20-alpine

LABEL maintainer="andreas.wenzelhuemer@gmail.com"

WORKDIR /src

COPY . .

RUN ls -al
RUN go build -o myapp
```

```
RUN mv myapp /usr/

EXPOSE 8888

CMD ["/usr/myapp"]
```

### 3. Build a docker image based on your dockerfile

```
docker image build -f -t awenzelhuemer/my-first-image:0.0.1 ./
```

The screenshot shows the Docker Hub interface for a repository named 'awenzelhuemer / my-first-image'. The page includes a header with the Docker Hub logo, a search bar, and navigation links. The repository page itself has tabs for 'General', 'Tags', 'Builds', 'Collaborators', 'Webhooks', and 'Settings'. A blue banner at the top prompts the user to 'Add a short description for this repository'. Below this, the repository name is displayed, followed by a description field (currently empty) and a 'Last pushed' timestamp of '2 hours ago'. To the right, there are 'Docker commands' for pushing a new tag, with a 'Public View' button. The 'Tags' section shows a table with one tag, '0.0.1', which is an 'Image' type, pushed '2 hours ago'. The 'Automated Builds' section provides information on how to connect to GitHub or Bitbucket for automated builds, with an 'Upgrade' button at the bottom.

awenzelhuemer / my-first-image

**Description**

This repository does not have a description

Last pushed: 2 hours ago

**Tags**

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
0.0.1		Image	2 hours ago	2 hours ago

[See all](#) [Go to Advanced Image Management](#)

**Docker commands**

To push a new tag to this repository,

```
docker push awenzelhuemer/my-first-image:tagname
```

[Public View](#)

**Automated Builds**

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions. [Read more about automated builds](#)

[Upgrade](#)

### 4. List all images that are stored in your local registry

```
docker images
```

```
PS C:\Users\Andi> docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
awenzelhuemer/go-service	latest	e454fdd25f76	5 minutes ago	339MB
awenzelhuemer/go-service	<none>	eebfefdb6b44	11 minutes ago	339MB
awenzelhuemer/go-service	<none>	fca2c6fd9398	25 minutes ago	318MB
<none>	<none>	ae73f07dae0f	46 minutes ago	318MB
quay.io/quarkus/ubi-quarkus-mandrel-builder-image	22.3-java17	413fc1ddd944	2 days ago	1.07GB
neo4j	latest	bbff370c3917	6 days ago	543MB
fhoee-mongo-dock-php-apache	latest	75e277b581fa	12 days ago	681MB
postgres	latest	2bb008a38e7c	4 weeks ago	379MB
mongo	latest	9a5e0d0cf6de	6 weeks ago	646MB
ghcr.io/muchobien/pocketbase	latest	d008fe52faf6	6 weeks ago	55.5MB
postgres	14	176399451347	8 weeks ago	377MB
elk-stack-dock-setup	latest	67830e84b31c	8 weeks ago	1.29GB
elk-stack-dock-elasticsearch	latest	9227ce144fcd	3 months ago	1.29GB
elk-stack-dock-kibana	latest	f46724c4e069	3 months ago	717MB
elk-stack-dock-metricbeat	latest	35f9658d367f	3 months ago	320MB
elk-stack-dock-logstash	latest	84efc0f6fcfb	3 months ago	748MB
postgres	14.1-alpine	1149d285a5f5	15 months ago	209MB
mongo-express	latest	2d2fb2cabc8f	18 months ago	136MB

## 5. Authenticate to the container registry

```
PS C:\Users\Andi> docker login
```

Login with your Docker ID to push and pull images from Docker Hub. If you do n't have a Docker ID, head over to <https://hub.docker.com> to create one.

Username: awenzelhuemer

Password:

Login Succeeded

Logging in with your password grants your terminal complete access to your account.

For better security, log in with a limited-privilege personal access token. Learn more at <https://docs.docker.com/go/access-tokens/>

## 6. Push the created image to your DockerHub account

```
docker image push awenzelhuemer/my-first-image:0.0.1
```

```
PS C:\Users\Andi\Documents\Github\fh-mc-cd-ex04> docker image push awenzelhuemer/my-first-image:0.0.1
```

The push refers to repository [docker.io/awenzelhuemer/my-first-image]

09636513e2b4: Pushed

403c1463e48b: Pushed

8130600bac74: Pushed

9d1f8a91962b: Pushed

b321ce0ae8c5: Pushed

0ac507e92643: Mounted from awenzelhuemer/go-service

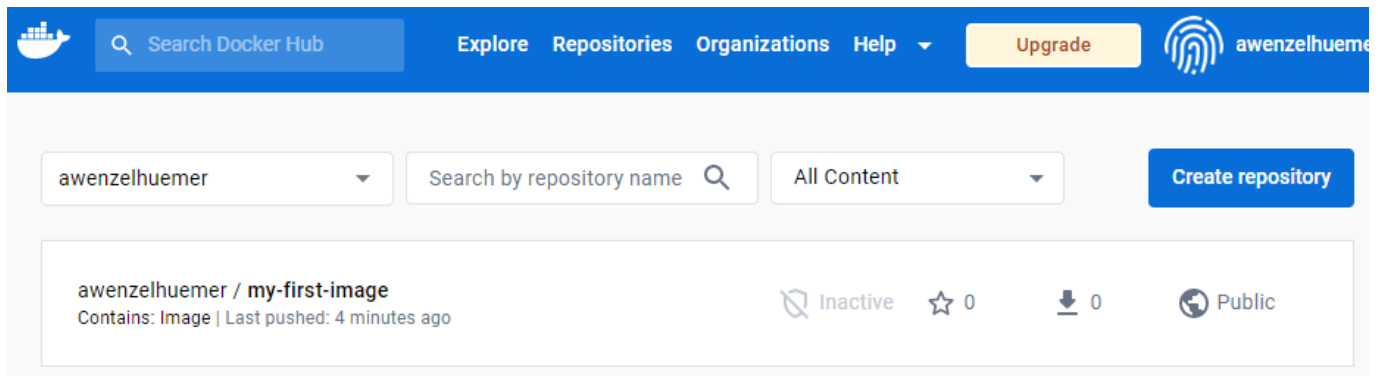
84c52813c38c: Mounted from awenzelhuemer/go-service

0403d7f628d3: Mounted from awenzelhuemer/go-service

f1417ff83b31: Mounted from awenzelhuemer/go-service

0.0.1: digest: sha256:e47c5e6824fe26ccf0632d55da0c6e894e815f2e6e4379a643b615f7573161c9 size: 2205

## 7. Verify deployed image



## Part 2

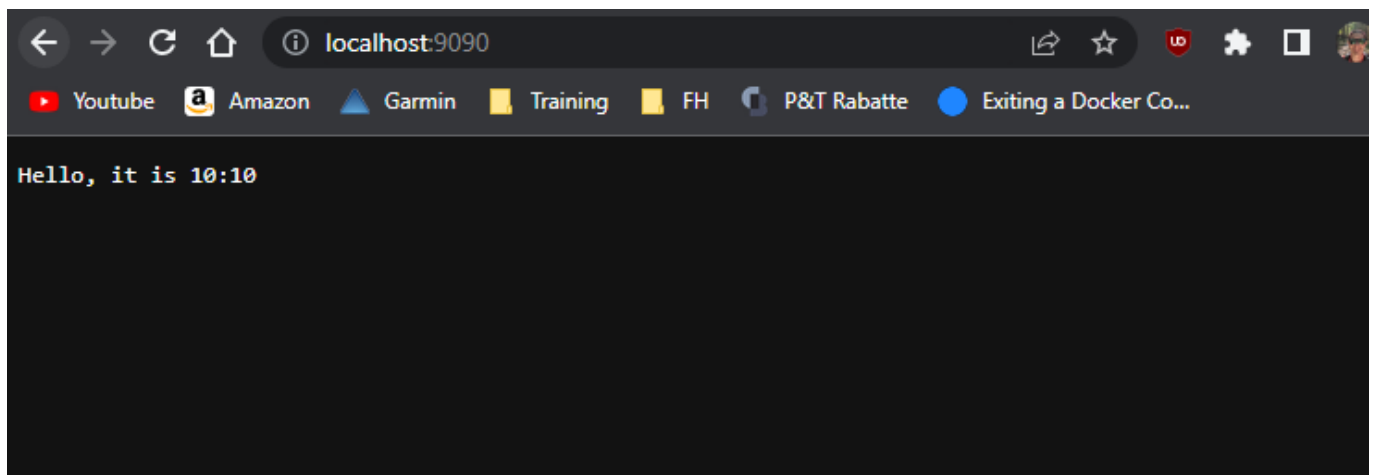
### 1. Create image from the provided Dockerfile

```
docker image build -t awenzelhuemer/myhello:0.0.1 ./
```

### 2. Run image

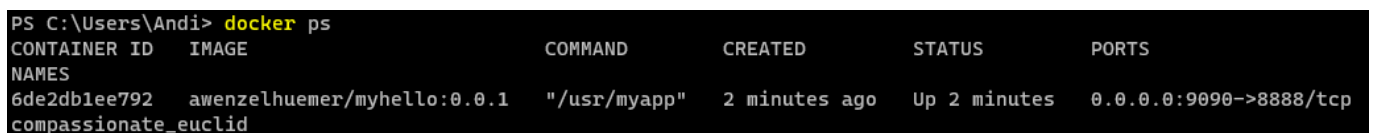
```
docker run --rm -p 9090:8888 awenzelhuemer/myhello:0.0.1
```

### 3. Check if application is running on localhost:9090



### 4. See your container running on your local Docker daemon

```
docker ps
```



### 5. Stop your container

```
docker stop 6de2db1ee792
```

## Part 3

Created with github actions in separate steps:

- build: Run go tests
- docker: Build image and deploy to docker hub

```
name: Docker Image CI

on:
  push:
    branches: [ "master" ]
  pull_request:
    branches: [ "master" ]

jobs:
  build:
    name: Build and test application
    runs-on: ubuntu-latest
    steps:
      - uses: actions/setup-go@v4
        with:
          go-version: 'stable'
      - name: Checkout
        uses: actions/checkout@v3
      - name: Run tests
        run: go test -v

  docker:
    name: Build and push Docker image
    needs: build
    runs-on: ubuntu-latest
    steps:
      - name: Checkout
        uses: actions/checkout@v3
      - name: Build and tag Docker image
        run: docker image build -t awenzelhuemer/myhello:${{ github.sha }} -t
awenzelhuemer/myhello:latest ./
      - name: Push Docker image
        uses: docker/login-action@v2
        with:
          username: ${{ secrets.DOCKER_USERNAME }}
          password: ${{ secrets.DOCKER_TOKEN }}
      - run: docker image push -a awenzelhuemer/myhello
```

awenzelhuemer / fh-mc-cd-ex04 Public  
forked from mrckurz/cd2020-ex04

Pin Watch 0 Fork 25 Star 0

Code Pull requests Actions Projects Wiki Security Insights Settings

← Docker Image CI

✓ Update docker-image.yml #10 Re-run all jobs ...

Summary

Jobs

- ✓ build

Run details

Usage

Workflow file

**build**  
succeeded 1 minute ago in 1m 15s

Search logs

- > ✓ Set up job 3s
- > ✓ Run actions/setup-go@v4 2s
- > ✓ Checkout 1s
- > ✓ Run tests 25s
- > ✓ Build and tag Docker image 31s
- > ✓ Push Docker image 1s
- > ✓ Run docker image push \*\*\*/myhello:d910a575355bb107c69c2714e15c2767b86c5be1 9s
- > ✓ Post Push Docker image 0s
- > ✓ Post Checkout 0s
- > ✓ Post Run actions/setup-go@v4 0s
- > ✓ Complete job 0s

```
docker image pull awenzelhuemer/myhello
```

```
PS C:\Users\Andi\Documents\Github\fh-mc-cd-ex04> docker pull awenzelhuemer/myhello
Using default tag: latest
latest: Pulling from awenzelhuemer/myhello
f56be85fc22e: Already exists
85791d961cd3: Already exists
9a18547b54a1: Already exists
55e4c593b212: Already exists
823b3e814265: Pull complete
921601d28df2: Pull complete
3ad45adb2eda: Pull complete
7275103ff430: Pull complete
Digest: sha256:b679316d7a939f4a1f3f8597e15a6424169da68798c08f5403f33d261b22d0e6
Status: Downloaded newer image for awenzelhuemer/myhello:latest
docker.io/awenzelhuemer/myhello:latest
```

## Part 4

Scan docker and code for vulnerabilities.

```
trivy-docker:
  name: "Scan for docker vulnerabilities"
  needs: docker
  runs-on: ubuntu-latest
  steps:
    - name: Run scanner
      uses: aquasecurity/trivy-action@master
      with:
        image-ref: "awenzelhuemer/myhello:latest"
        format: "table"
        exit-code: "1"
        ignore-unfixed: true
        skip-files: "*.go"
        vuln-type: "os,library"
        severity: "CRITICAL"

trivy-repo:
  name: "Scan for code vulnerabilities"
  runs-on: ubuntu-latest
  steps:
    - name: Checkout
      uses: actions/checkout@v3
    - name: Run scanner
      uses: aquasecurity/trivy-action@master
      with:
        scan-type: "fs"
        scan-ref: "."
        format: "sarif"
        output: "trivy-results.sarif"
        severity: "CRITICAL,HIGH"
```

The screenshot shows a GitHub Actions workflow run for the file `Update docker-image.yml #21`. The workflow is triggered by a push to the `master` branch. The status is **Success**, and the total duration is **1m 37s**. The workflow consists of four jobs:

- Build and test application** (23s)
- Build and push Docker image** (35s)
- Scan for docker vulnerabilities** (18s)
- Scan for code vulnerabilities** (18s)

The workflow is triggered via push 1 minute ago by `awenzelhuemer` pushed to `2660537 master`. The workflow file is `docker-image.yml` on push.

## Log for docker vulnerabilities:

```
Scan for docker vulnerabilities
succeeded 1 minute ago in 18s

> Set up job 1s
> Build aquasecurity/trivy-action@master 10s
v Run scanner 7s

1 ▶ Run aquasecurity/trivy-action@master
13 /usr/bin/docker run --name c0442650c49b410014a378262f756a5ef0f2b_1643a5 --label 6c0442 --workdir /github/workspace --rm -e "INPUT_IMAGE-REF" -e "INPUT_FORMAT" -e "INPUT_EXIT-CODE" -e
"INPUT_IGNORE-UNFIXED" -e "INPUT_SKIP-FILES" -e "INPUT_VULN-TYPE" -e "INPUT_SEVERITY" -e "INPUT_SCAN-TYPE" -e "INPUT_INPUT" -e "INPUT_SCAN-REF" -e "INPUT_TEMPLATE" -e "INPUT_OUTPUT" -e
"INPUT_SKIP-DIRS" -e "INPUT_CACHE-DIR" -e "INPUT_TIMEOUT" -e "INPUT_IGNORE-POLICY" -e "INPUT_HIDE-PROGRESS" -e "INPUT_LIST-ALL-PKGS" -e "INPUT_SCANNERS" -e "INPUT_TRIVYIGNORES" -e
"INPUT_ARTIFACT-TYPE" -e "INPUT_GITHUB-PAT" -e "INPUT_TRIVY-CONFIG" -e "INPUT_LIMIT-SEVERITIES-FOR-SARIF" -e "HOME" -e "GITHUB_JOB" -e "GITHUB_REF" -e "GITHUB_SHA" -e
"GITHUB_REPOSITORY" -e "GITHUB_REPOSITORY_OWNER" -e "GITHUB_REPOSITORY_OWNER_ID" -e "GITHUB_RUN_ID" -e "GITHUB_RUN_NUMBER" -e "GITHUB_RETENTION_DAYS" -e "GITHUB_RUN_ATTEMPT" -e
"GITHUB_REPOSITORY_ID" -e "GITHUB_ACTOR_ID" -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_WORKFLOW_REF" -e "GITHUB_WORKFLOW_SHA" -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
"GITHUB_STATE" -e "GITHUB_OUTPUT" -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/fh-mc-cd-ex04/fh-mc-cd-ex04":"/github/workspace" 6c0442:650c49b410014a378262f756a5ef0f2b
"-a image" "-b table" "-c " "-d 1" "-e true" "-f os,library" "-g CRITICAL" "-h " "-i awenzelhuemer/myhello:latest" "-j ." "-k " "-l " "-m " "-n " "-o " "-p " "-q *.go" "-r false" "-s "
"-t " "-u " "-v " "-z "
14 Running trivy with options: trivy image --format table --exit-code 1 --ignore-unfixed --vuln-type os,library --severity CRITICAL --skip-files *.go awenzelhuemer/myhello:latest
15 Global options:
16 2023-04-27T11:41:21.214Z INFO Need to update DB
17 2023-04-27T11:41:21.214Z INFO DB Repository: ghcr.io/aquasecurity/trivy-db
18 2023-04-27T11:41:21.214Z INFO Downloading DB...
19 35.37 MiB / 36.50 MiB [----->] 96.92% ? p/s ?36.50 MiB / 36.50 MiB [----->]
100.00% ? p/s ?36.50 MiB / 36.50 MiB [----->] 100.00% ? p/s ?36.50 MiB / 36.50 MiB [----->]
->] 100.00% 1.95 MiB p/s ETA 0s36.50 MiB / 36.50 MiB [----->] 100.00% 1.95 MiB p/s ETA 0s36.50 MiB / 36.50 MiB [----->]
"----->] 100.00% 1.95 MiB p/s ETA 0s36.50 MiB / 36.50 MiB [----->] 100.00% 1.83 MiB p/s ETA 0s36.50 MiB / 36.50 MiB [----->]
----->] 100.00% 1.83 MiB p/s ETA 0s36.50 MiB / 36.50 MiB [----->] 100.00% 23.98 MiB p/s 1.7s2023-04-
27T11:41:23.433Z INFO Vulnerability scanning is enabled
20 2023-04-27T11:41:23.433Z INFO Secret scanning is enabled
21 2023-04-27T11:41:23.433Z INFO If your scanning is slow, please try '--scanners vuln' to disable secret scanning
22 2023-04-27T11:41:23.433Z INFO Please see also https://aquasecurity.github.io/trivy/v0.40/docs/secret/scanning/#recommendation for faster secret detection
23 2023-04-27T11:41:27.196Z INFO Detected OS: alpine
24 2023-04-27T11:41:27.196Z INFO Detecting Alpine vulnerabilities...
25 2023-04-27T11:41:27.198Z INFO Number of language-specific files: 5
26 2023-04-27T11:41:27.199Z INFO Detecting gomod vulnerabilities...
27 2023-04-27T11:41:27.200Z INFO Detecting node-pkg vulnerabilities...
28
29 awenzelhuemer/myhello:latest (alpine 3.17.3)
30 =====
31 Total: 0 / (CRITICAL: 0)
```