Going Minimal

A guide to improving container image security

Who am I?

Software engineer turned Cloud Enthusiast ☁️

Kubernetes wizard 🪄

Linux Nerd 🐧

Vulnerable AWK Playground

```
> vulnerable-awk-playground
   ____    __
  / __/___/ /  ___
 / _// __/ _ \/ _ \
/___/\__/_//_/\___/ v4.11.2
High performance, minimalist Go web framework
https://echo.labstack.com
____________________________________O/_______
                                    O\
```

```
> curl -X POST \
>   -F 'text=hello world' \
>   -F 'awkScript={print $1}' localhost:8080/test

{"stderr":"","stdout":"hello\n"}
```

dockerfiles/Dockerfile.all-in-one

```
# Our dockerfile
```

dockerfiles/Dockerfile.all-in-one

```dockerfile
# Our dockerfile
FROM golang:1.21.3-bookworm
```

dockerfiles/Dockerfile.all-in-one

```dockerfile
# Our dockerfile
FROM golang:1.21.3-bookworm

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go
RUN cp vulnerable-awk-playground /usr/bin/
ENTRYPOINT ["vulnerable-awk-playground"]
```

```
> docker build . \
>       -t vulnerable-awk-playground:all-in-one \
>       -f dockerfiles/Dockerfile.all-in-one
> docker build . \
>       -t vulnerable-awk-playground:all-in-one \
>       -f dockerfiles/Dockerfile.all-in-one
...
=> exporting layers                                                 1.0s
=> writing image ....                                               0.0s
=> naming to docker.io/library/vulnerable-awk-playground:all-in-one 0.0s
```

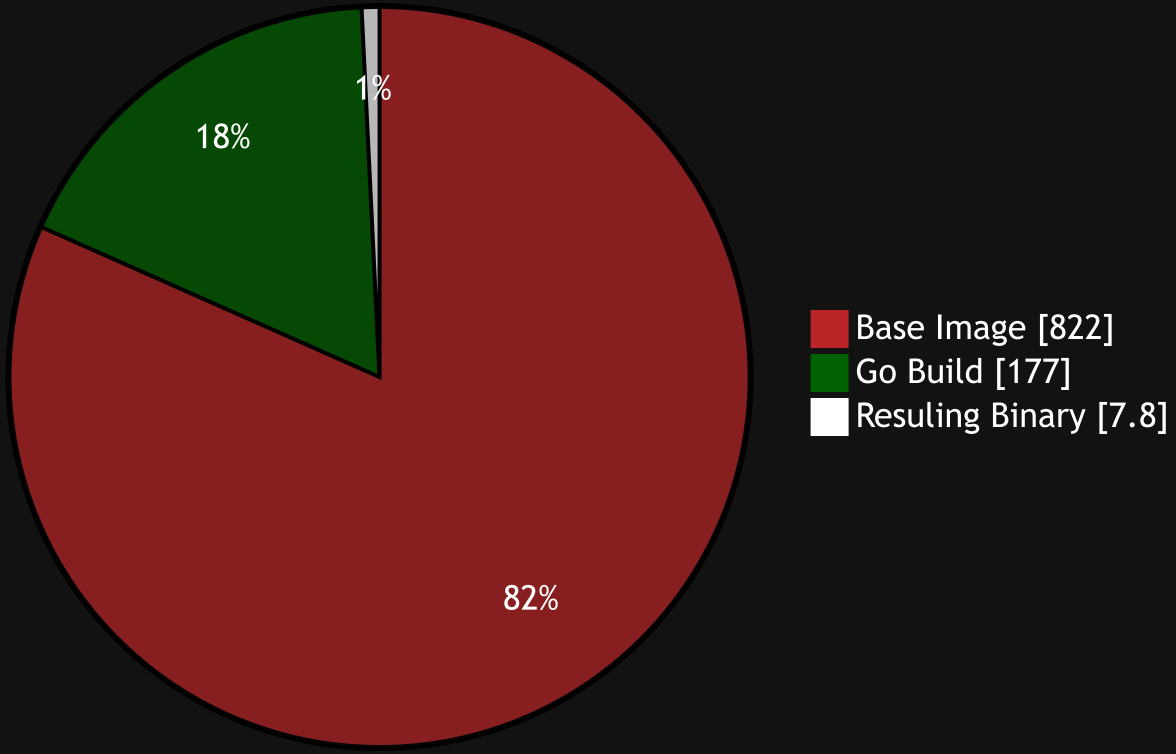
```
> docker build . \
>       -t vulnerable-awk-playground:all-in-one \
>       -f dockerfiles/Dockerfile.all-in-one
...
=> exporting layers                                                 1.0s
=> writing image ....                                               0.0s
=> naming to docker.io/library/vulnerable-awk-playground:all-in-one 0.0s

> docker image ls vulnerable-awk-playground:all-in-one \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'
```

KUSAMA
KUSAMA
20kg
0
1kg
2
19
18
17
16
15
14
13
12
11
10
CAPACITY
GRADUATION

```
> docker build . \
>       -t vulnerable-awk-playground:all-in-one \
>       -f dockerfiles/Dockerfile.all-in-one
...
=> exporting layers                                                 1.0s
=> writing image ....                                               0.0s
=> naming to docker.io/library/vulnerable-awk-playground:all-in-one 0.0s

> docker image ls vulnerable-awk-playground:all-in-one \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'

vulnerable-awk-playground:all-in-one 1.01GB
```

1.01GB

vulnerable-awk-playground:all-in-one



Multi-Stage Builds

dockerfiles/Dockerfile.multi-stage-ubuntu

```dockerfile
# Our dockerfile
FROM golang:1.21.3-bookworm

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go
RUN cp vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

dockerfiles/Dockerfile.multi-stage-ubuntu

```dockerfile
# Our dockerfile
FROM golang:1.21.3-bookworm as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go
RUN cp vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

dockerfiles/Dockerfile.multi-stage-ubuntu

```dockerfile
# Our dockerfile
FROM golang:1.21.3-bookworm as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM ubuntu:23.10

RUN cp vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

dockerfiles/Dockerfile.multi-stage-ubuntu

```dockerfile
# Our dockerfile
FROM golang:1.21.3-bookworm as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM ubuntu:23.10

COPY --from=build /app/vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

dockerfiles/Dockerfile.multi-stage-ubuntu

```dockerfile
# Our dockerfile
FROM golang:1.21.3-bookworm as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM ubuntu:23.10

COPY --from=build /app/vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

```diff
--- dockerfiles/Dockerfile.all-in-one
+++ dockerfiles/Dockerfile.multi-stage-ubuntu
@@ -1,11 +1,13 @@
-FROM golang:1.21.3-bookworm
+FROM golang:1.21.3-bookworm as build

 WORKDIR /app

 COPY go.mod go.sum main.go /app/

 RUN go build -o vulnerable-awk-playground ./main.go

-RUN cp vulnerable-awk-playground /usr/bin/
+FROM ubuntu:23.10

+COPY --from=build /app/vulnerable-awk-playground /usr/bin/
+
 ENTRYPOINT ["vulnerable-awk-playground"]
```

```
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-ubuntu \
>       -f dockerfiles/Dockerfile.multi-stage-ubuntu
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-ubuntu \
>       -f dockerfiles/Dockerfile.multi-stage-ubuntu
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-ubuntu 0.0s
```

```
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-ubuntu \
>       -f dockerfiles/dockerfile.multi-stage-ubuntu
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-ubuntu 0.0s

> docker image ls vulnerable-awk-playground:multi-stage-ubuntu \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-ubuntu \
>       -f dockerfiles/dockerfile.multi-stage-ubuntu
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-ubuntu 0.0s

> docker image ls vulnerable-awk-playground:multi-stage-ubuntu \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'

vulnerable-awk-playground:multi-stage-ubuntu 101MB
```

Alpine

Minimal Linux Distribution

Minimal Linux Distribution

Often used for containers

Minimal Linux Distribution

Often used for containers

Using APK package manager

dockerfiles/Dockerfile.multi-stage-alpine

```
# Our dockerfile
FROM golang:1.21.3-bookworm as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM ubuntu:23.10

COPY --from=build /app/vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

dockerfiles/Dockerfile.multi-stage-alpine

```dockerfile
# Our dockerfile
FROM golang:1.21.3-alpine3.18 as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM alpine:3.18

COPY --from=build /app/vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

dockerfiles/Dockerfile.multi-stage-alpine

```
# Our dockerfile
FROM golang:1.21.3-alpine3.18 as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM alpine:3.18

COPY --from=build /app/vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

```
--- dockerfiles/Dockerfile.multi-stage-ubuntu   2023-11-10 21:13:49
+++ dockerfiles/Dockerfile.multi-stage-alpine   2023-11-10 21:13:49
@@ -1,13 +1,13 @@
-FROM golang:1.21.3-bookworm as build
+FROM golang:1.21.3-alpine3.18 as build

 WORKDIR /app

 COPY go.mod go.sum main.go /app/

 RUN go build -o vulnerable-awk-playground ./main.go

-FROM ubuntu:23.10
+FROM alpine:3.18

 COPY --from=build /app/vulnerable-awk-playground /usr/bin/

 ENTRYPOINT ["vulnerable-awk-playground"]
```

```
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-alpine \
>       -f dockerfiles/Dockerfile.multi-stage-alpine
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-alpine \
>       -f dockerfiles/Dockerfile.multi-stage-alpine
...
=> exporting layers                                          1.0s
=> writing image ....                                        0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-alpine 0.0s
```

```
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-alpine \
>       -f dockerfiles/dockerfile.multi-stage-alpine
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-alpine 0.0s

> docker image ls vulnerable-awk-playground:multi-stage-alpine \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-alpine \
>       -f dockerfiles/dockerfile.multi-stage-alpine
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-alpine 0.0s

> docker image ls vulnerable-awk-playground:multi-stage-alpine \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'

vulnerable-awk-playground:multi-stage-alpine 15.3MB
```

Distroless

Google Distroless

Chainguard-Images

Google Distroless

Chainguard-Images

Google Distroless

Maintained by Google

Chainguard-Images

Google Distroless

Chainguard-Images

Maintained by Google

Based on Debian

Google Distroless

Chainguard-Images

Maintained by Google

Based on Debian

Various images available:

- Static, Java, Python, NodeJS

Google Distroless

Maintained by Google

Based on Debian

Various images available:

- Static, Java, Python, NodeJS

Chainguard-Images

Maintained by Chainguard

Google Distroless

Maintained by Google

Based on Debian

Various images available:

- Static, Java, Python, NodeJS

Chainguard-Images

Maintained by Chainguard

Various images available:

- ArgoCD, Nginx, Golang, NodeJS

Google Distroless

Maintained by Google

Based on Debian

Various images available:

- Static, Java, Python, NodeJS

Chainguard-Images

Maintained by Chainguard

Various images available:

- ArgoCD, Nginx, Golang, NodeJS

Based on Wolfi

Wolfi?

Image by diana.grytsku on Freepik

Linux Distribution

Linux Distribution

Maintained by Chainguard

Linux Distribution

Maintained by Chainguard

Using APK package manager

Linux Distribution

Maintained by Chainguard

Using APK package manager

Custom package repository

Vulnerable AWK Playground = depends on => AWK

dockerfiles/Dockerfile.multi-stage-chainguard

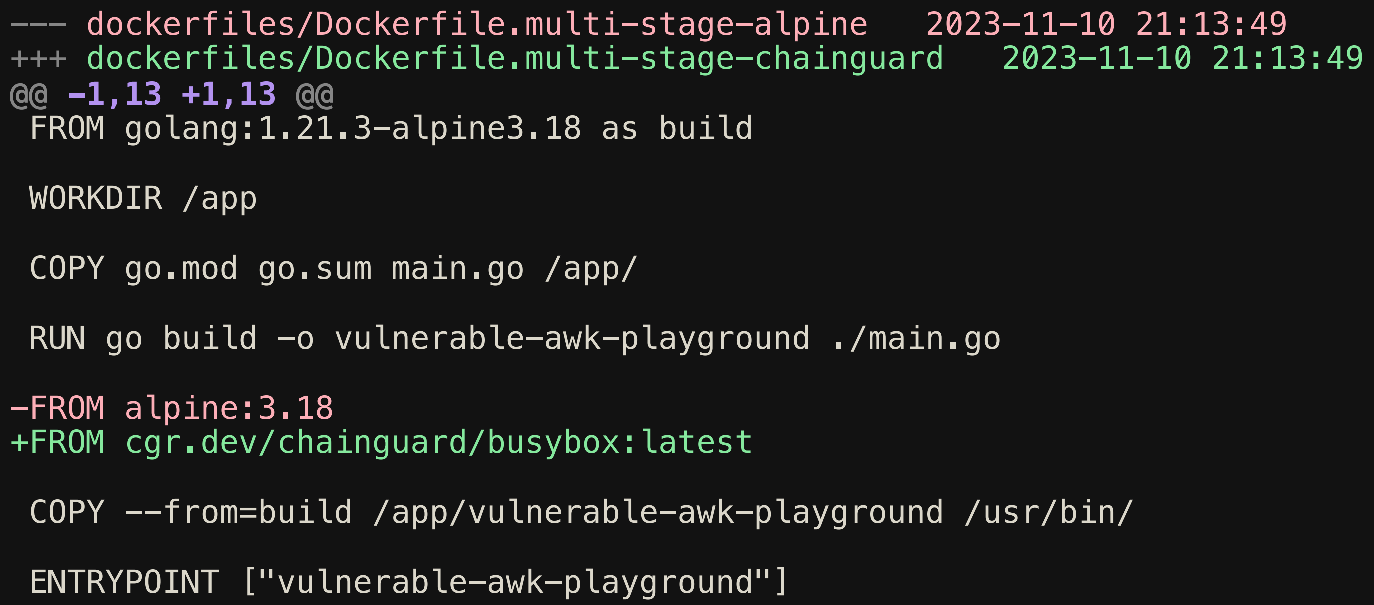
```dockerfile
# Our dockerfile
FROM golang:1.21.3-alpine3.18 as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM alpine:3.18

COPY --from=build /app/vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

dockerfiles/Dockerfile.multi-stage-chainguard

```
# Our dockerfile
FROM golang:1.21.3-alpine3.18 as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM cgr.dev/chainguard/busybox:latest

COPY --from=build /app/vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```

dockerfiles/Dockerfile.multi-stage-chainguard

```dockerfile
# Our dockerfile
FROM golang:1.21.3-alpine3.18 as build

WORKDIR /app
COPY go.mod go.sum main.go /app/
RUN go build -o vulnerable-awk-playground ./main.go

FROM cgr.dev/chainguard/busybox:latest

COPY --from=build /app/vulnerable-awk-playground /usr/bin/

ENTRYPOINT ["vulnerable-awk-playground"]
```



```diff
--- dockerfiles/Dockerfile.multi-stage-alpine   2023-11-10 21:13:49
+++ dockerfiles/Dockerfile.multi-stage-chainguard   2023-11-10 21:13:49
@@ -1,13 +1,13 @@
 FROM golang:1.21.3-alpine3.18 as build

 WORKDIR /app

 COPY go.mod go.sum main.go /app/

 RUN go build -o vulnerable-awk-playground ./main.go

-FROM alpine:3.18
+FROM cgr.dev/chainguard/busybox:latest

 COPY --from=build /app/vulnerable-awk-playground /usr/bin/

 ENTRYPOINT ["vulnerable-awk-playground"]
```

```
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-chainguard \
>       -f dockerfiles/Dockerfile.multi-stage-chainguard
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-chainguard \
>       -f dockerfiles/Dockerfile.multi-stage-chainguard
...
=> exporting layers                                               1.0s
=> writing image ....                                             0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-chainguard 0.0s
```

```
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-chainguard \
>       -f dockerfiles/Dockerfile.multi-stage-chainguard
...
=> exporting layers                                               1.0s
=> writing image ....                                             0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-chainguard 0.0s

> docker image ls vulnerable-awk-playground:multi-stage-chainguard \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'
> docker build . \
>       -t vulnerable-awk-playground:multi-stage-chainguard \
>       -f dockerfiles/Dockerfile.multi-stage-chainguard
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../vulnerable-awk-playground:multi-stage-chainguard 0.0s

> docker image ls vulnerable-awk-playground:multi-stage-chainguard \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'

vulnerable-awk-playground:multi-stage-chainguard 14.5MB
```

Image	Size
vulnerable-awk-playground:all-in-one	1.01GB
vulnerable-awk-playground:multi-stage-ubuntu	101MB
vulnerable-awk-playground:multi-stage-alpine	15.3MB
vulnerable-awk-playground:multi-stage-chainguard	14.5MB

Size reduction of 69x

Dockerfile footguns

Image by 8photo on Freepik

```
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```dockerfile
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```dockerfile
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```dockerfile
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

I AM WRITING A SECRET IN A DOCKERFILE
AND DELETE IT IN AN LATER COMMAND
THE SECRET IS GONE FOR GOOD, RIGHT?
THE SECRET IS GONE FOR GOOD, RIGHT?

What even is an container image?

Images = tar balls

Images = tar balls

Metadata

Images = tar balls

Metadata

Layers

CONTAINERS ARE LIKE ONIONS
THEY HAVE LAYERS

Docker

OCI

(Open Container Initiative)

Docker

OCI

Docker

Superset of OCI spec

OCI

Docker

Superset of OCI spec

Specific extensions

OCI

Docker

Superset of OCI spec

Specific extensions

Governed by Docker

OCI

Docker

Governed by Docker

Superset of OCI spec

Specific extensions

OCI

Docker

Governed by Docker

Superset of OCI spec

Specific extensions

OCI

Industry standard

Docker

Governed by Docker

Superset of OCI spec

Specific extensions

OCI

Industry standard

Governed by Open Container

Initiative

Docker

Governed by Docker

Superset of OCI spec

Specific extensions

OCI

Industry standard

Governed by Open Container Initiative

Part of Linux Foundation

Docker

Governed by Docker

Superset of OCI spec

Specific extensions

OCI

Industry standard

Governed by Open Container Initiative

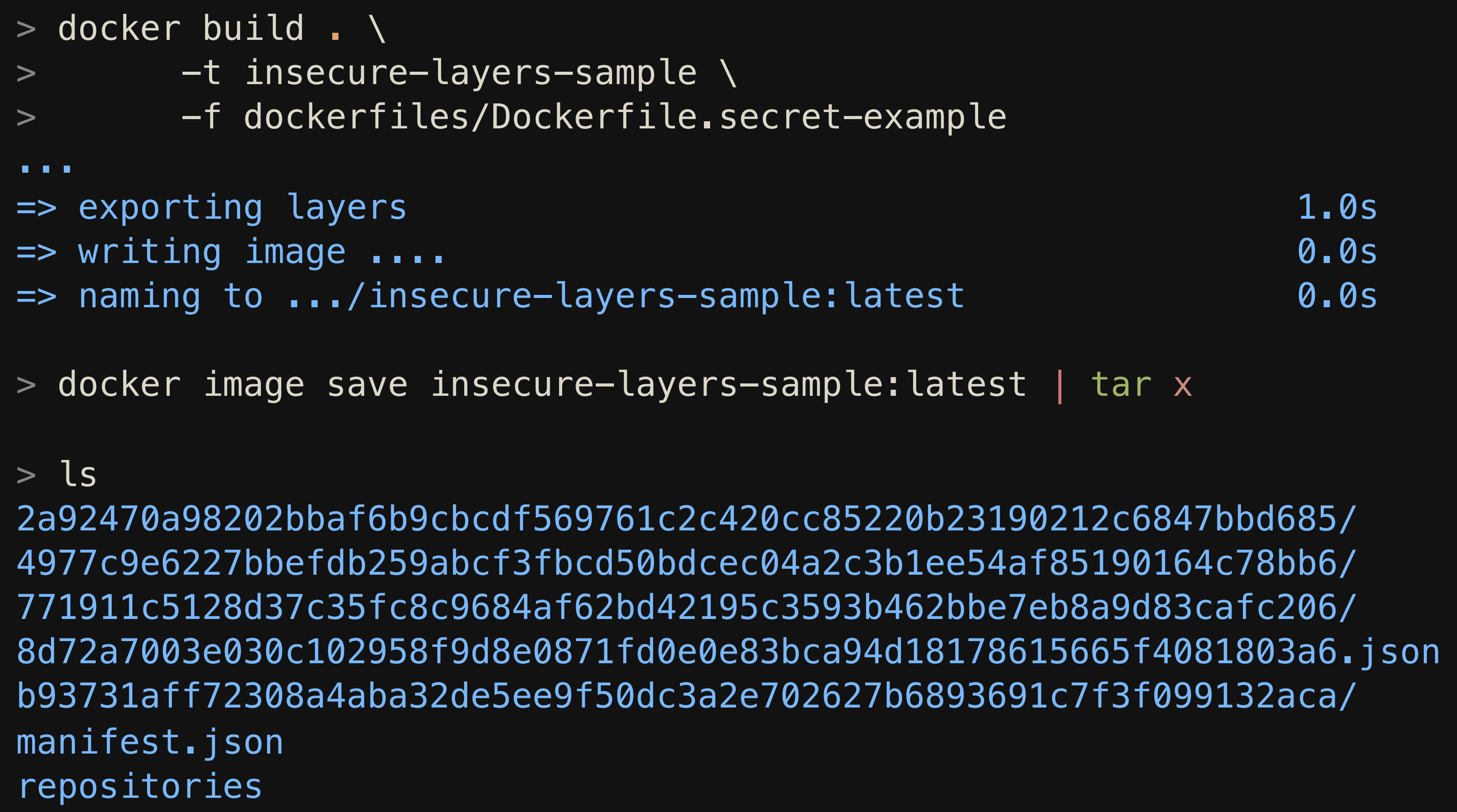
Part of Linux Foundation

Very flexible (e.g. storying binaries)

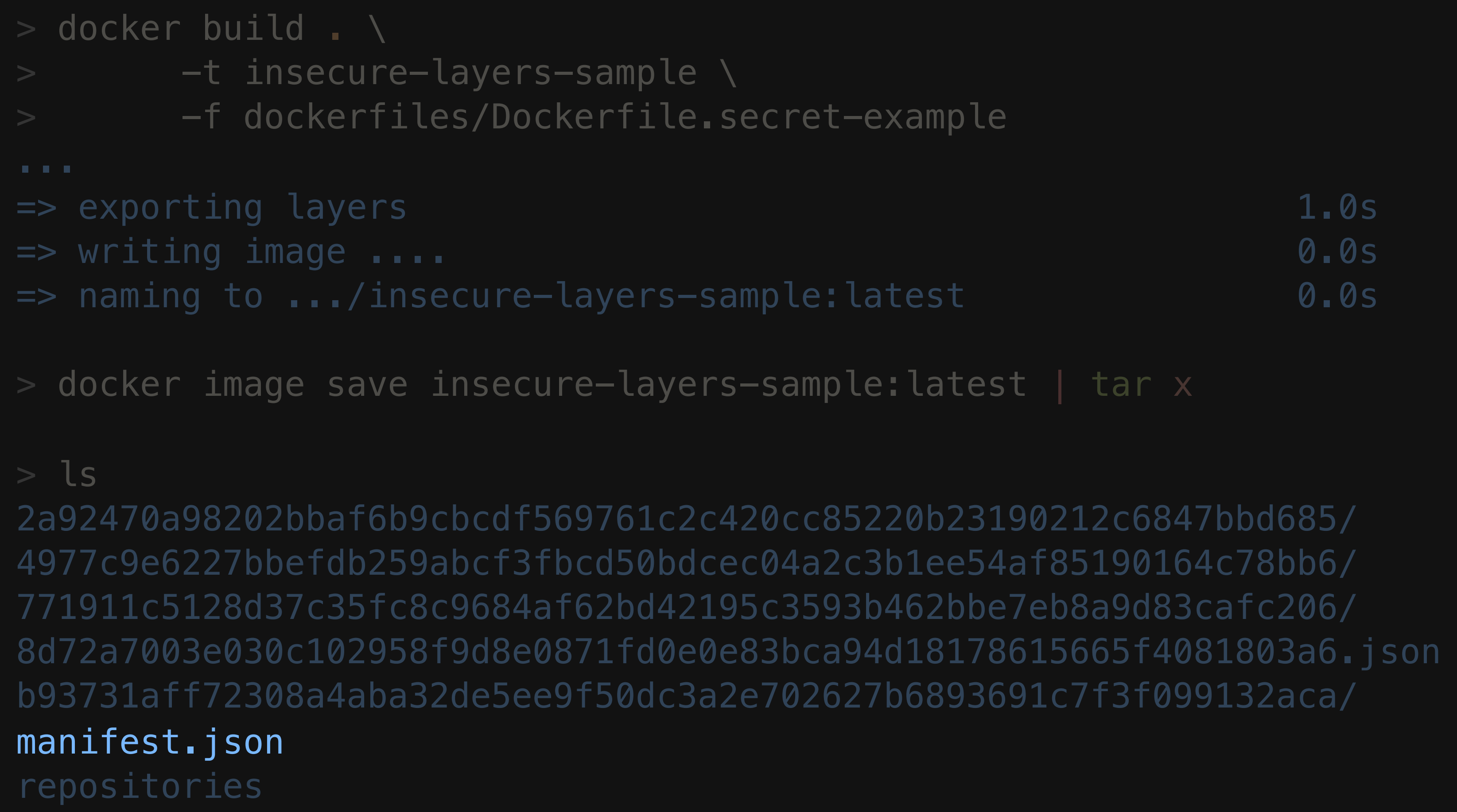
```
> docker build . \
>       -t insecure-layers-sample \
>       -f dockerfiles/Dockerfile.secret-example
...
```

```
> docker build . \
>       -t insecure-layers-sample \
>       -f dockerfiles/Dockerfile.secret-example
...
=> exporting layers                                          1.0s
=> writing image ....                                        0.0s
=> naming to .../insecure-layers-sample:latest               0.0s
```

```
> docker build . \
>       -t insecure-layers-sample \
>       -f dockerfiles/Dockerfile.secret-example
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../insecure-layers-sample:latest                0.0s

> docker image save insecure-layers-sample:latest | tar x
> docker build . \
>       -t insecure-layers-sample \
>       -f dockerfiles/Dockerfile.secret-example
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../insecure-layers-sample:latest                0.0s

> docker image save insecure-layers-sample:latest | tar x

> ls
```



```
> docker build . \
>       -t insecure-layers-sample \
>       -f dockerfiles/Dockerfile.secret-example
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../insecure-layers-sample:latest                0.0s

> docker image save insecure-layers-sample:latest | tar x

> ls
2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/
4977c9e6227bbefdb259abcf3fbcd50bdcec04a2c3b1ee54af85190164c78bb6/
771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/
8d72a7003e030c102958f9d8e0871fd0e0e83bca94d18178615665f4081803a6.json
b93731aff72308a4aba32de5ee9f50dc3a2e702627b6893691c7f3f099132aca/
manifest.json
repositories
```



```
> docker build . \
>       -t insecure-layers-sample \
>       -f dockerfiles/Dockerfile.secret-example
...
=> exporting layers                                           1.0s
=> writing image ....                                         0.0s
=> naming to .../insecure-layers-sample:latest                0.0s

> docker image save insecure-layers-sample:latest | tar x

> ls
2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/
4977c9e6227bbefdb259abcf3fbcd50bdcec04a2c3b1ee54af85190164c78bb6/
771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/
8d72a7003e030c102958f9d8e0871fd0e0e83bca94d18178615665f4081803a6.json
b93731aff72308a4aba32de5ee9f50dc3a2e702627b6893691c7f3f099132aca/
manifest.json
repositories
```

manifest.json

```
[
  {
    "Config": "8d72a7<...>3a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff<...>132aca/layer.tar",
      "2a92470a9<...>bbd685/layer.tar",
      "4977c9e62<...>c78bb6/layer.tar",
      "771911c51<...>afc206/layer.tar"
    ]
  }
]
```

manifest.json

```
[
  {
    "Config": "8d72a7<...>3a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff<...>132aca/layer.tar",
      "2a92470a9<...>bbd685/layer.tar",
      "4977c9e62<...>c78bb6/layer.tar",
      "771911c51<...>afc206/layer.tar"
    ]
  }
]
```

manifest.json

```
[
  {
    "Config": "8d72a7<...>3a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff<...>132aca/layer.tar",
      "2a92470a9<...>bbd685/layer.tar",
      "4977c9e62<...>c78bb6/layer.tar",
      "771911c51<...>afc206/layer.tar"
    ]
  }
]
```

manifest.json

```
[
  {
    "Config": "8d72a7<...>3a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff<...>132aca/layer.tar",
      "2a92470a9<...>bbd685/layer.tar",
      "4977c9e62<...>c78bb6/layer.tar",
      "771911c51<...>afc206/layer.tar"
    ]
  }
]
```

manifest.json

```
[
  {
    "Config": "8d72a7<...>3a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff<...>132aca/layer.tar",
      "2a92470a9<...>bbd685/layer.tar",
      "4977c9e62<...>c78bb6/layer.tar",
      "771911c51<...>afc206/layer.tar"
    ]
  }
]
```

8d72a7…3a6.json (config)

```
{
  "architecture": "arm64",
  "os": "linux",
  "variant": "v8",
  "config": {
    "Env": [ "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin" ],
    "Entrypoint": [ "sh" ],
  },
  "history": [
    {
      "created": "2023-09-28T20:39:33.966748405Z",
      "created_by": "/bin/sh -c #(nop) ADD file:ff311...b41717 in / "
    },
    {
      "created": "2023-09-28T20:39:34.079909813Z",
      "created_by": "/bin/sh -c #(nop)  CMD [\"/bin/sh\"]",
      "empty_layer": true
    },
  ]
}
```

8d72a7…3a6.json (config)

```
{
  "architecture": "arm64",
  "os": "linux",
  "variant": "v8",
  "config": {
    "Env": [ "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin" ],
    "Entrypoint": [ "sh" ],
  },
  "history": [
    {
      "created": "2023-09-28T20:39:33.966748405Z",
      "created_by": "/bin/sh -c #(nop) ADD file:ff311...b41717 in / "
    },
    {
      "created": "2023-09-28T20:39:34.079909813Z",
      "created_by": "/bin/sh -c #(nop)  CMD [\"/bin/sh\"]",
      "empty_layer": true
    },
  ]
}
```

8d72a7…3a6.json (config)

```
{
  "architecture": "arm64",
  "os": "linux",
  "variant": "v8",
  "config": {
    "Env": [ "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin" ],
    "Entrypoint": [ "sh" ],
  },
  "history": [
    {
      "created": "2023-09-28T20:39:33.966748405Z",
      "created_by": "/bin/sh -c #(nop) ADD file:ff311...b41717 in / "
    },
    {
      "created": "2023-09-28T20:39:34.079909813Z",
      "created_by": "/bin/sh -c #(nop)  CMD [\"/bin/sh\"]",
      "empty_layer": true
    },
  ]
}
```

8d72a7…3a6.json (config)

```
{
  "architecture": "arm64",
  "os": "linux",
  "variant": "v8",
  "config": {
    "Env": [ "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin" ],
    "Entrypoint": [ "sh" ],
  },
  "history": [
    {
      "created": "2023-09-28T20:39:33.966748405Z",
      "created_by": "/bin/sh -c #(nop) ADD file:ff311...b41717 in / "
    },
    {
      "created": "2023-09-28T20:39:34.079909813Z",
      "created_by": "/bin/sh -c #(nop)  CMD [\"/bin/sh\"]",
      "empty_layer": true
    },
  ]
}
```

A word about layers

```dockerfile
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```dockerfile
FROM alpine:3.18

COPY super-secure.txt /etc/secret.txt

RUN echo "do something with secret"

RUN rm /etc/secret.txt

ENTRYPOINT [ "sh" ]
```

```
> tar xf layer1.tar --directory image/

> tar xf layer2.tar --directory image/

> tar xf layer3.tar --directory image/
```

What about deleting a file?

Photo by Samu Lopez on Unsplash

Special whiteout files

Special whiteout files

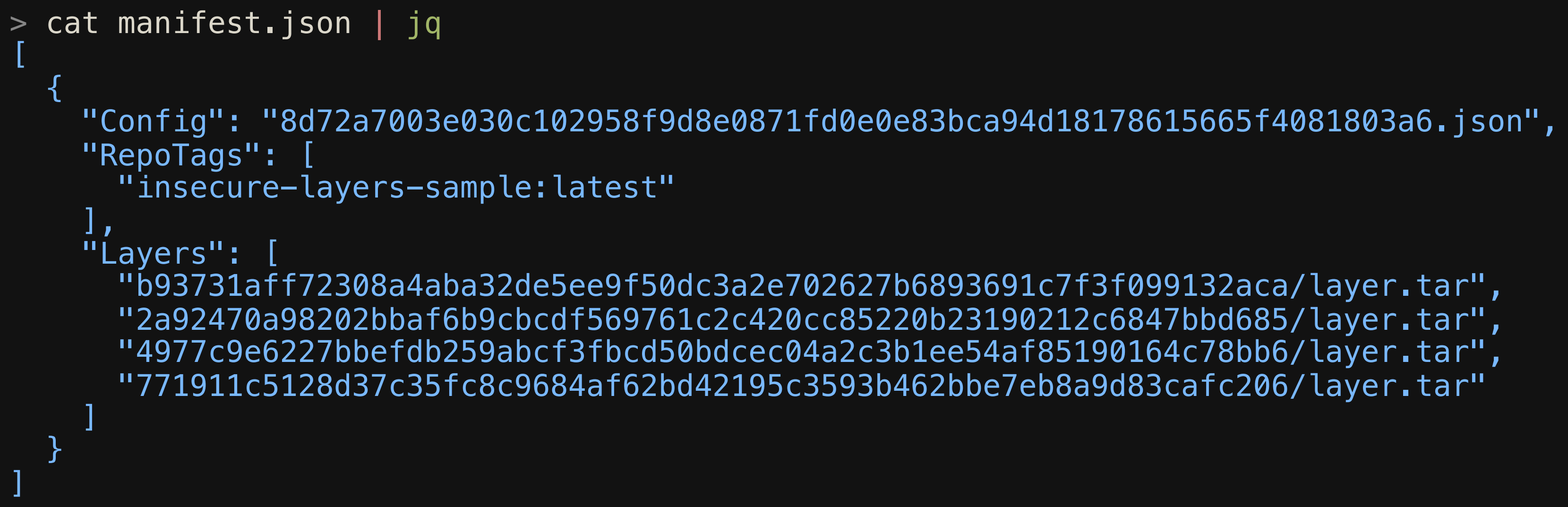
Empty file

Special whiteout files

Empty file

Prefix with `` `.wh.` ``

```
> cat manifest.json | jq
```



```
> cat manifest.json | jq
[
  {
    "Config": "8d72a7003e030c102958f9d8e0871fd0e0e83bca94d18178615665f4081803a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff72308a4aba32de5ee9f50dc3a2e702627b6893691c7f3f099132aca/layer.tar",
      "2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar",
      "4977c9e6227bbefdb259abcf3fbcd50bdcec04a2c3b1ee54af85190164c78bb6/layer.tar",
      "771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/layer.tar"
    ]
  }
]
```

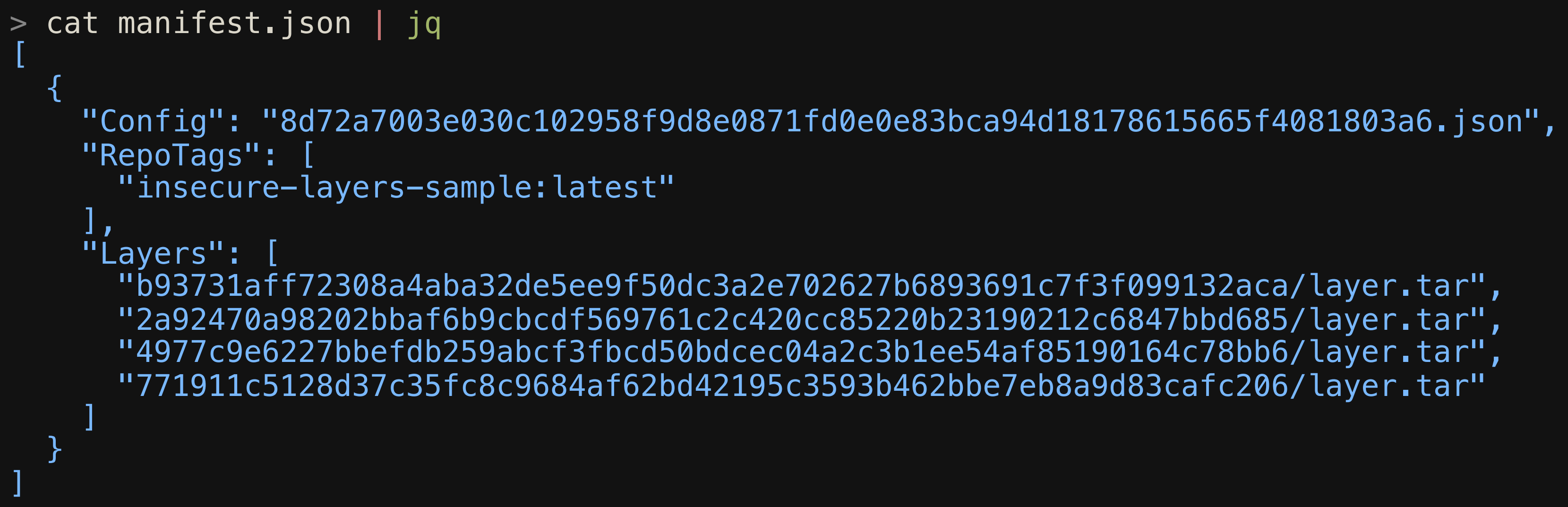
```
> cat manifest.json | jq
[
  {
    "Config": "8d72a7003e030c102958f9d8e0871fd0e0e83bca94d18178615665f4081803a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff72308a4aba32de5ee9f50dc3a2e702627b6893691c7f3f099132aca/layer.tar",
      "2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar",
      "4977c9e6227bbefdb259abcf3fbcd50bdcec04a2c3b1ee54af85190164c78bb6/layer.tar",
      "771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/layer.tar"
    ]
  }
]
```

```
> tar tvf 771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/layer.tar
> tar tvf 771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/layer.tar

drwxr-xr-x  0 0      0           0 Nov 13 20:07 etc/
----------  0 0      0           0 Nov 13 20:07 etc/.wh.secret.txt
```

I AM WRITING A SECRET IN A DOCKERFILE
AND DELETE IT IN AN LATER COMMAND
THE SECRET IS GONE FOR GOOD, RIGHT?
THE SECRET IS GONE FOR GOOD, RIGHT?

Well yes, but actually no

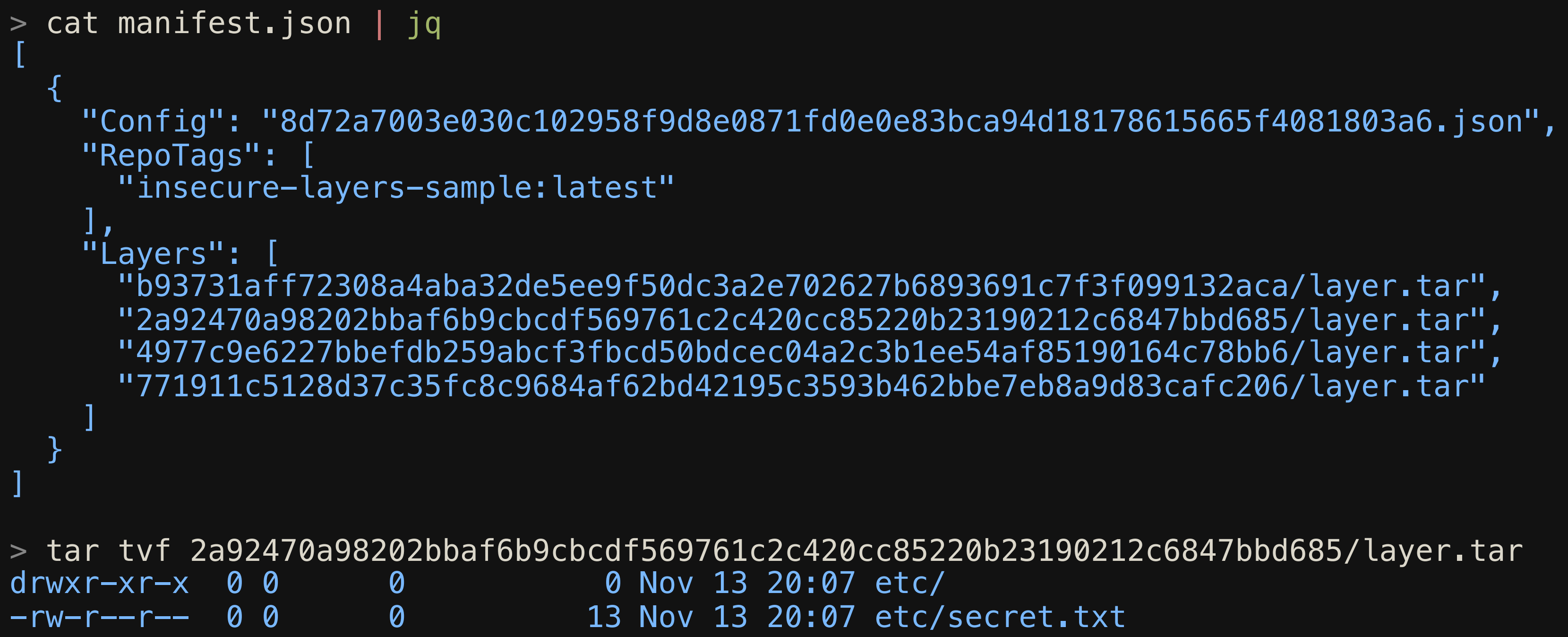
```
> docker run --rm -it insecure-layers-sample:latest \
>           -c "cat /etc/secret.txt"
> docker run --rm -it insecure-layers-sample:latest \
>           -c "cat /etc/secret.txt"

cat: can't open '/etc/secret.txt': No such file or directory
```



```
> cat manifest.json | jq
[
  {
    "Config": "8d72a7003e030c102958f9d8e0871fd0e0e83bca94d18178615665f4081803a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff72308a4aba32de5ee9f50dc3a2e702627b6893691c7f3f099132aca/layer.tar",
      "2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar",
      "4977c9e6227bbefdb259abcf3fbcd50bdcec04a2c3b1ee54af85190164c78bb6/layer.tar",
      "771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/layer.tar"
    ]
  }
]
```

```
> cat manifest.json | jq
[
  {
    "Config": "8d72a7003e030c102958f9d8e0871fd0e0e83bca94d18178615665f4081803a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff72308a4aba32de5ee9f50dc3a2e702627b6893691c7f3f099132aca/layer.tar",
      "2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar",
      "4977c9e6227bbefdb259abcf3fbcd50bdcec04a2c3b1ee54af85190164c78bb6/layer.tar",
      "771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/layer.tar"
    ]
  }
]
```

```
> cat manifest.json | jq
[
  {
    "Config": "8d72a7003e030c102958f9d8e0871fd0e0e83bca94d18178615665f4081803a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff72308a4aba32de5ee9f50dc3a2e702627b6893691c7f3f099132aca/layer.tar",
      "2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar",
      "4977c9e6227bbefdb259abcf3fbcd50bdcec04a2c3b1ee54af85190164c78bb6/layer.tar",
      "771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/layer.tar"
    ]
  }
]

> tar tvf 2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar
```



```
> cat manifest.json | jq
[
  {
    "Config": "8d72a7003e030c102958f9d8e0871fd0e0e83bca94d18178615665f4081803a6.json",
    "RepoTags": [
      "insecure-layers-sample:latest"
    ],
    "Layers": [
      "b93731aff72308a4aba32de5ee9f50dc3a2e702627b6893691c7f3f099132aca/layer.tar",
      "2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar",
      "4977c9e6227bbefdb259abcf3fbcd50bdcec04a2c3b1ee54af85190164c78bb6/layer.tar",
      "771911c5128d37c35fc8c9684af62bd42195c3593b462bbe7eb8a9d83cafc206/layer.tar"
    ]
  }
]

> tar tvf 2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar
drwxr-xr-x  0 0      0           0 Nov 13 20:07 etc/
-rw-r--r--  0 0      0          13 Nov 13 20:07 etc/secret.txt
```

```
> tar xf  2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar \
>       --to-stdout etc/secret.txt
> tar xf  2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar \
>       --to-stdout etc/secret.txt

super-secure
```

```
> tar xf  2a92470a98202bbaf6b9cbcdf569761c2c420cc85220b23190212c6847bbd685/layer.tar \
>       --to-stdout etc/secret.txt

super-secure
```

Image by benzoix on Freepik

Docker secrets mounts

```
FROM alpine:3.18

# Secret available under /run/secrets/mysecret
RUN --mount=type=secret,id=mysecret echo "do something with secret"

ENTRYPOINT [ "sh" ]
```

```
> docker build . \
>       -t insecure-layers-sample:secret-mount \
>       -f dockerfiles/Dockerfile.secret-example \
>       --secret id=mysecret,src=secret.txt
> docker build . \
>       -t insecure-layers-sample:secret-mount \
>       -f dockerfiles/Dockerfile.secret-example \
>       --secret id=mysecret,src=secret.txt
```

Other footguns

Other footguns

Leaking secrets through ENV

Other footguns

Leaking secrets through ENV

ENV vs ARG

Other footguns

Leaking secrets through ENV

ENV vs ARG

Not cleaning up package manager cache

Other footguns

Leaking secrets through ENV

ENV vs ARG

Not cleaning up package manager cache

Running image as root user

jib/ko

Image by wayhomestudio on Freepik

jib

Developed by Google

Developed by Google

Mainly for Java

Developed by Google

Mainly for Java

Plugin for Gradle/Maven

Developed by Google

Mainly for Java

Plugin for Gradle/Maven

Optimized images

ko

Like jib but for Golang

Like jib but for Golang

Run `` `ko build` ``

Nice

No need for docker daemon

Not so Nice

Hard to install additional dependencies

Image by wayhomestudio on Freepik

apko

Developed by Chainguard

Developed by Chainguard

Define images in declarative way

Developed by Chainguard

Define images in declarative way

Specify APK packages in resulting image

Developed by Chainguard

Define images in declarative way

Specify APK packages in resulting image

Melange is sister tool for building APKs

melange.yaml

```yaml
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

melange.yaml

```yaml
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

melange.yaml

```yaml
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

melange.yaml

```
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

melange.yaml

```yaml
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

melange.yaml

```yaml
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

melange.yaml

```yaml
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

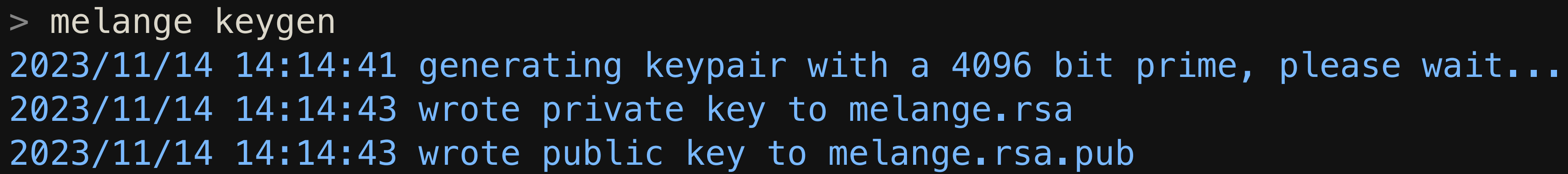
melange.yaml

```yaml
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

melange.yaml

```yaml
package:
  name: "vulnerable-awk-playground"
  version: v0.0.1
  epoch: 0
environment:
  contents:
    keyring:
      - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    repositories:
      - https://packages.wolfi.dev/os
pipeline:
  - uses: go/build
    with:
      packages: ./main.go
      output: vulnerable-awk-playground
```

```
> melange keygen
```



```
> melange keygen
2023/11/14 14:14:41 generating keypair with a 4096 bit prime, please wait...
2023/11/14 14:14:43 wrote private key to melange.rsa
2023/11/14 14:14:43 wrote public key to melange.rsa.pub
```

```
> melange keygen
2023/11/14 14:14:41 generating keypair with a 4096 bit prime, please wait...
2023/11/14 14:14:43 wrote private key to melange.rsa
2023/11/14 14:14:43 wrote public key to melange.rsa.pub

> melange build --signing-key melange.rsa
> melange keygen
2023/11/14 14:14:41 generating keypair with a 4096 bit prime, please wait...
2023/11/14 14:14:43 wrote private key to melange.rsa
2023/11/14 14:14:43 wrote public key to melange.rsa.pub

> melange build --signing-key melange.rsa
... Lots and lots of output ...
```

```
> melange keygen
2023/11/14 14:14:41 generating keypair with a 4096 bit prime, please wait...
2023/11/14 14:14:43 wrote private key to melange.rsa
2023/11/14 14:14:43 wrote public key to melange.rsa.pub

> melange build --signing-key melange.rsa
... Lots and lots of output ...

> ls packages/aarch64/
> melange keygen
2023/11/14 14:14:41 generating keypair with a 4096 bit prime, please wait...
2023/11/14 14:14:43 wrote private key to melange.rsa
2023/11/14 14:14:43 wrote public key to melange.rsa.pub

> melange build --signing-key melange.rsa
... Lots and lots of output ...

> ls packages/aarch64/
APKINDEX.json
APKINDEX.tar.gz
vulnerable-awk-playground-v0.0.1-r0.apk
```

apko.yaml

```yaml
contents:
  keyring:
    - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    - ./melange.rsa.pub
  repositories:
    - https://packages.wolfi.dev/os
    - ./packages
  packages:
    - gawk
    - vulnerable-awk-playground

accounts:
  groups:
    - groupname: nobody
      gid: 65534
  users:
    - username: nobody
      uid: 65534
  run-as: nobody

entrypoint:
  command: vulnerable-awk-playground
```

apko.yaml

```yaml
contents:
  keyring:
    - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    - ./melange.rsa.pub
  repositories:
    - https://packages.wolfi.dev/os
    - ./packages
  packages:
    - gawk
    - vulnerable-awk-playground

accounts:
  groups:
    - groupname: nobody
      gid: 65534
  users:
    - username: nobody
      uid: 65534
  run-as: nobody

entrypoint:
  command: vulnerable-awk-playground
```

apko.yaml

```yaml
contents:
  keyring:
    - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    - ./melange.rsa.pub
  repositories:
    - https://packages.wolfi.dev/os
    - ./packages
  packages:
    - gawk
    - vulnerable-awk-playground

accounts:
  groups:
    - groupname: nobody
      gid: 65534
  users:
    - username: nobody
      uid: 65534
  run-as: nobody

entrypoint:
  command: vulnerable-awk-playground
```

apko.yaml

```
contents:
  keyring:
    - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    - ./melange.rsa.pub
  repositories:
    - https://packages.wolfi.dev/os
    - ./packages
  packages:
    - gawk
    - vulnerable-awk-playground

accounts:
  groups:
    - groupname: nobody
      gid: 65534
  users:
    - username: nobody
      uid: 65534
  run-as: nobody

entrypoint:
  command: vulnerable-awk-playground
```

apko.yaml

```yaml
contents:
  keyring:
    - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    - ./melange.rsa.pub
  repositories:
    - https://packages.wolfi.dev/os
    - ./packages
  packages:
    - gawk
    - vulnerable-awk-playground

accounts:
  groups:
    - groupname: nobody
      gid: 65534
  users:
    - username: nobody
      uid: 65534
  run-as: nobody

entrypoint:
  command: vulnerable-awk-playground
```

apko.yaml

```yaml
contents:
  keyring:
    - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    - ./melange.rsa.pub
  repositories:
    - https://packages.wolfi.dev/os
    - ./packages
  packages:
    - gawk
    - vulnerable-awk-playground

accounts:
  groups:
    - groupname: nobody
      gid: 65534
  users:
    - username: nobody
      uid: 65534
  run-as: nobody

entrypoint:
  command: vulnerable-awk-playground
```

apko.yaml

```yaml
contents:
  keyring:
    - https://packages.wolfi.dev/os/wolfi-signing.rsa.pub
    - ./melange.rsa.pub
  repositories:
    - https://packages.wolfi.dev/os
    - ./packages
  packages:
    - gawk
    - vulnerable-awk-playground

accounts:
  groups:
    - groupname: nobody
      gid: 65534
  users:
    - username: nobody
      uid: 65534
  run-as: nobody

entrypoint:
  command: vulnerable-awk-playground
```

```
> apko build ./apko.yaml \
>       vulnerable-awk-playground:apko \
>       vulnerable-awk-playground.tar
> apko build ./apko.yaml \
>       vulnerable-awk-playground:apko \
>       vulnerable-awk-playground.tar
... lots of build output ...
```

```
> apko build ./apko.yaml \
>       vulnerable-awk-playground:apko \
>       vulnerable-awk-playground.tar
... lots of build output ...

> docker load < vulnerable-awk-playground.tar
> apko build ./apko.yaml \
>       vulnerable-awk-playground:apko \
>       vulnerable-awk-playground.tar
... lots of build output ...

> docker load < vulnerable-awk-playground.tar
856409ea41b2: Loading layer [============================>]  8.732MB/8.732MB
Loaded image: vulnerable-awk-playground:apko-arm64
```

```
> apko build ./apko.yaml \
>       vulnerable-awk-playground:apko \
>       vulnerable-awk-playground.tar
... lots of build output ...

> docker load < vulnerable-awk-playground.tar
856409ea41b2: Loading layer [============================>]  8.732MB/8.732MB
Loaded image: vulnerable-awk-playground:apko-arm64

> docker image ls vulnerable-awk-playground:apko-arm64 \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'
> apko build ./apko.yaml \
>       vulnerable-awk-playground:apko \
>       vulnerable-awk-playground.tar
... lots of build output ...

> docker load < vulnerable-awk-playground.tar
856409ea41b2: Loading layer [============================>]  8.732MB/8.732MB
Loaded image: vulnerable-awk-playground:apko-arm64

> docker image ls vulnerable-awk-playground:apko-arm64 \
>           --format '{{.Repository}}:{{.Tag}} {{.Size}}'
vulnerable-awk-playground:apko-arm64 20.8MB
```

Should I use apko/melange for everything now?

Probably not

Why should I even care about minimal images?

Lower attack surface

Lower attack surface

Lower storage cost

Lower attack surface

Lower storage cost

Faster download times

Lower attack surface

Lower storage cost

Faster download times

Reduced number of CVEs

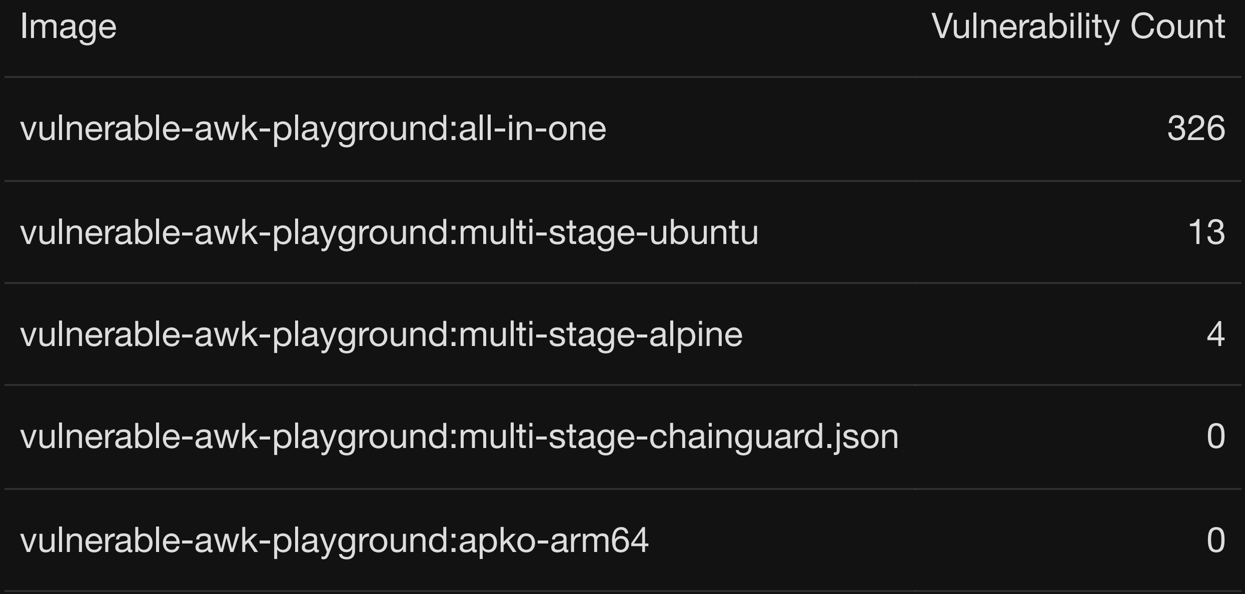
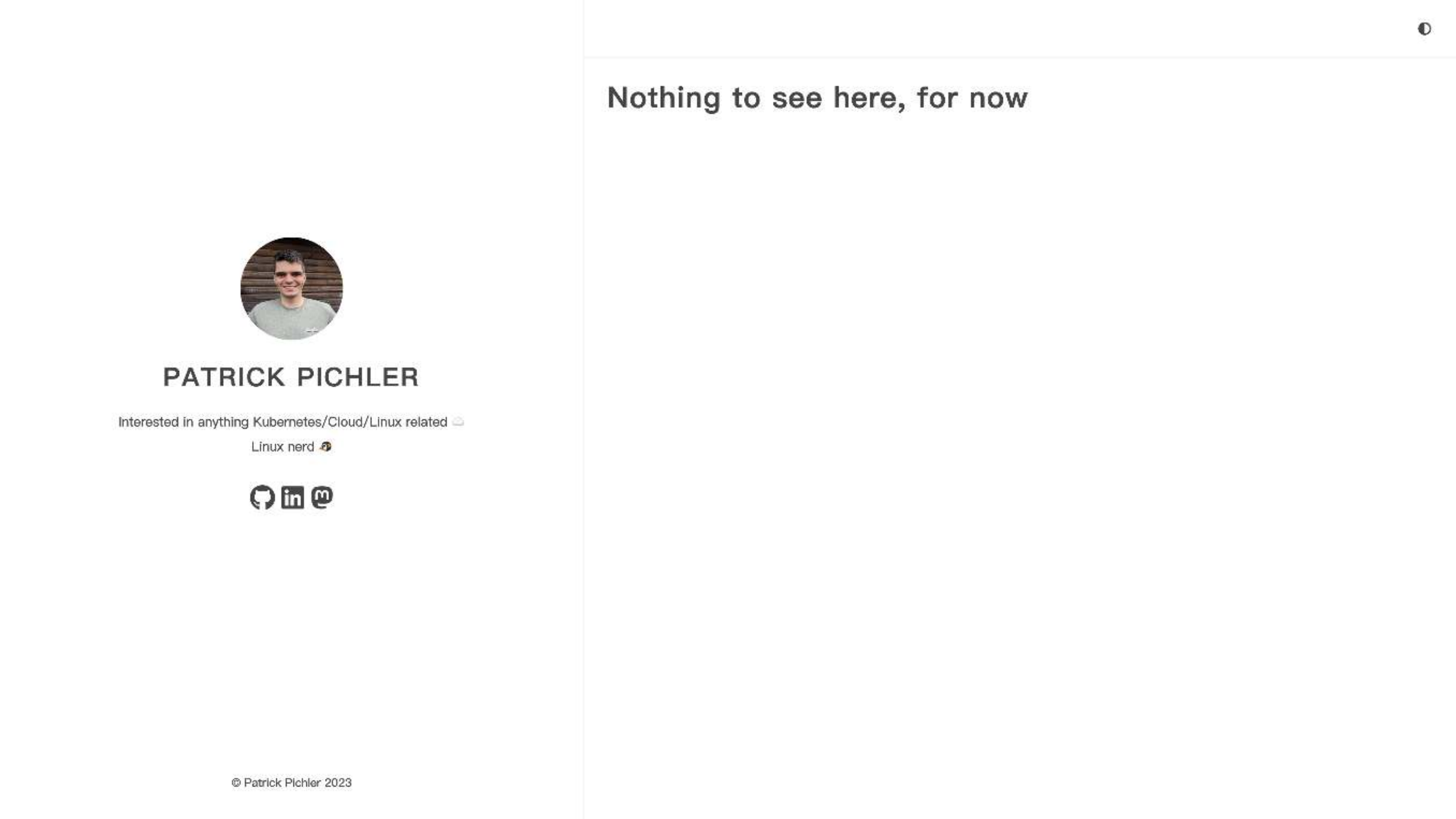


Image	Vulnerability Count
vulnerable-awk-playground:all-in-one	326
vulnerable-awk-playground:multi-stage-ubuntu	13
vulnerable-awk-playground:multi-stage-alpine	4
vulnerable-awk-playground:multi-stage-chainguard.json	0
vulnerable-awk-playground:apko-arm64	0

To find out more head over to patrickpichler.dev



PATRICK PICHLER

Interested in anything Kubernetes/Cloud/Linux related ☁️

Linux nerd 🐧

Nothing to see here, for now

© Patrick Pichler 2023

Be mindful about base image

Be mindful about base image

Dockerfiles seem simple, can cause headaches

Be mindful about base image

Dockerfiles seem simple, can cause headaches

Good alternatives to Dockerfiles exist

Also checkout

Building Container Images the Modern Way - Adrian Mouat, Chainguard

HTTPS://WWW.YOUTUBE.COM/WATCH?V=NZLZ0O4DURS

Feel free to approach me and have a chat