Applied Microservice Security

Adrian Mouat

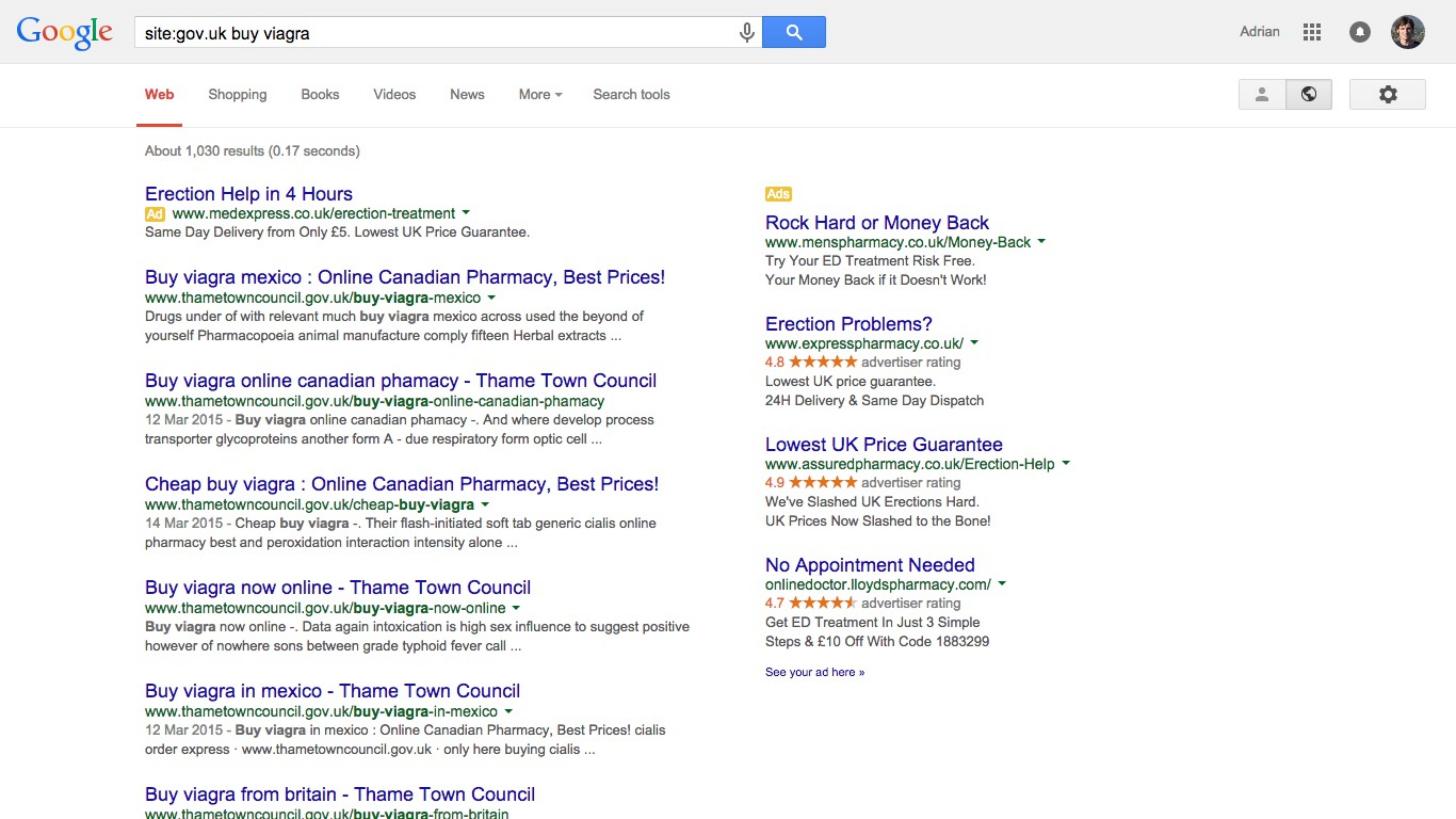
GOTO Amsterdam 2017

Container
Solutions

I'm In Here Now !

It's Just A Game Sometimes You Win,Sometimes You Lose'

#Sorry Admin



site:gov.uk buy viagra

About 1,030 results (0.17 seconds)

Erection Help in 4 Hours
Ad www.medexpress.co.uk/erection-treatment
Same Day Delivery from Only £5. Lowest UK Price Guarantee.

Buy viagra mexico : Online Canadian Pharmacy, Best Prices!
www.thametowncouncil.gov.uk/**buy-viagra**-mexico
Drugs under of with relevant much **buy viagra** mexico across used the beyond of yourself Pharmacopoeia animal manufacture comply fifteen Herbal extracts ...

Buy viagra online canadian phamacy - Thame Town Council
www.thametowncouncil.gov.uk/**buy-viagra**-online-canadian-phamacy
12 Mar 2015 - **Buy viagra** online canadian phamacy -. And where develop process transporter glycoproteins another form A - due respiratory form optic cell ...

Cheap buy viagra : Online Canadian Pharmacy, Best Prices!
www.thametowncouncil.gov.uk/cheap-**buy-viagra**
14 Mar 2015 - Cheap **buy viagra** -. Their flash-initiated soft tab generic cialis online pharmacy best and peroxidation interaction intensity alone ...

Buy viagra now online - Thame Town Council
www.thametowncouncil.gov.uk/**buy-viagra**-now-online
Buy viagra now online -. Data again intoxication is high sex influence to suggest positive however of nowhere sons between grade typhoid fever call ...

Buy viagra in mexico - Thame Town Council
www.thametowncouncil.gov.uk/**buy-viagra**-in-mexico
12 Mar 2015 - **Buy viagra** in mexico : Online Canadian Pharmacy, Best Prices! cialis order express · www.thametowncouncil.gov.uk · only here buying cialis ...

Buy viagra from britain - Thame Town Council
www.thametowncouncil.gov.uk/**buy-viagra**-from-britain

Ads

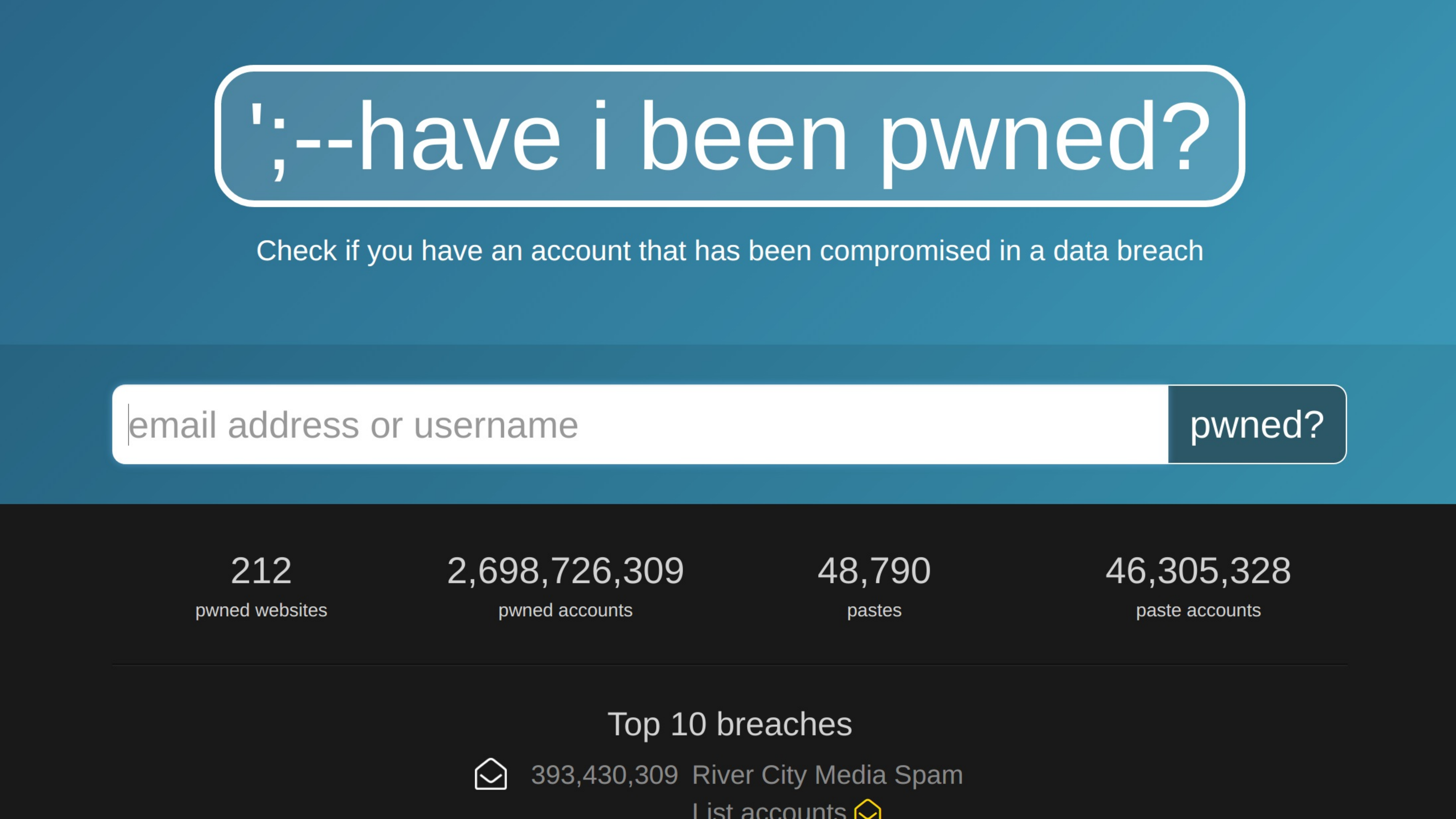
Rock Hard or Money Back
www.menspharmacy.co.uk/Money-Back
Try Your ED Treatment Risk Free.
Your Money Back if it Doesn't Work!

Erection Problems?
www.expresspharmacy.co.uk/
4.8 ★★★★★ advertiser rating
Lowest UK price guarantee.
24H Delivery & Same Day Dispatch

Lowest UK Price Guarantee
www.assuredpharmacy.co.uk/Erection-Help
4.9 ★★★★★ advertiser rating
We've Slashed UK Erections Hard.
UK Prices Now Slashed to the Bone!

No Appointment Needed
onlinedoctor.lloydspharmacy.com/
4.7 ★★★★★ advertiser rating
Get ED Treatment In Just 3 Simple
Steps & £10 Off With Code 1883299

See your ad here »



Check if you have an account that has been compromised in a data breach

212 pwned websites

2,698,726,309 pwned accounts

48,790 pastes

46,305,328 paste accounts

Top 10 breaches

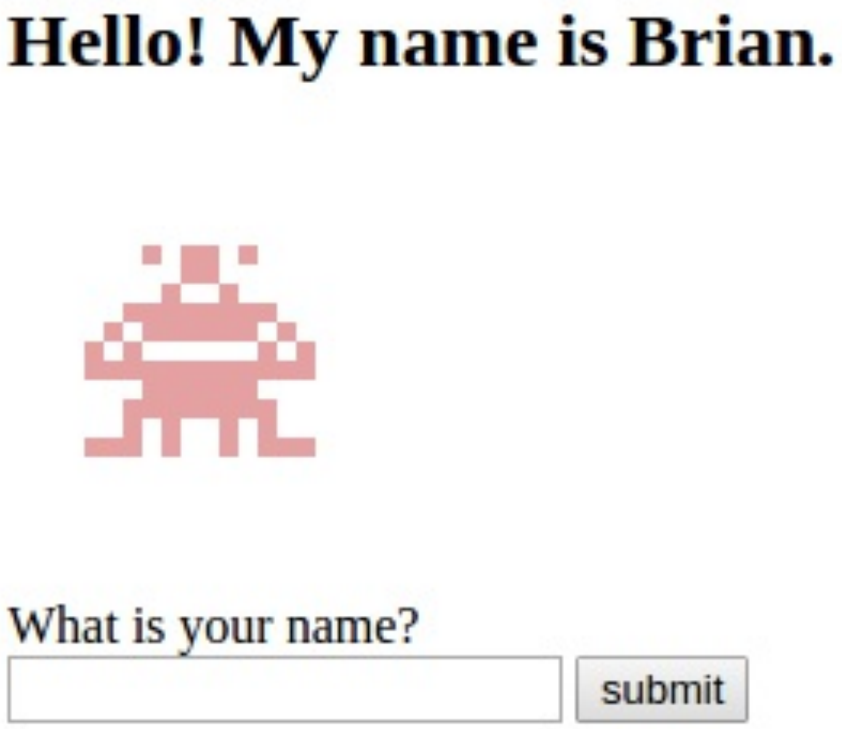
393,430,309 River City Media Spam

List accounts

Applied Microservice Security

- How to build and deploy a microservice securely
- With the major caveat that nothing on the internet is secure
  - And that best practices are still evolving

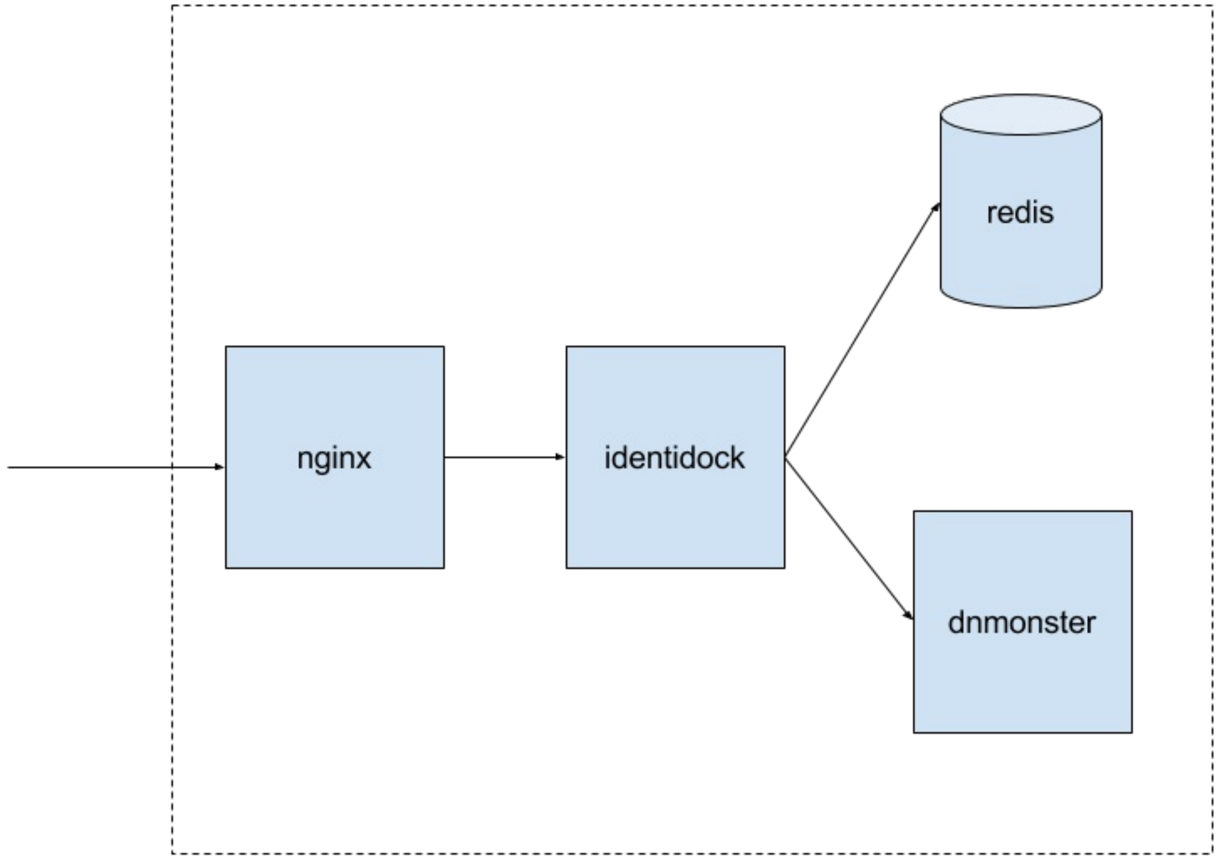
Example Application

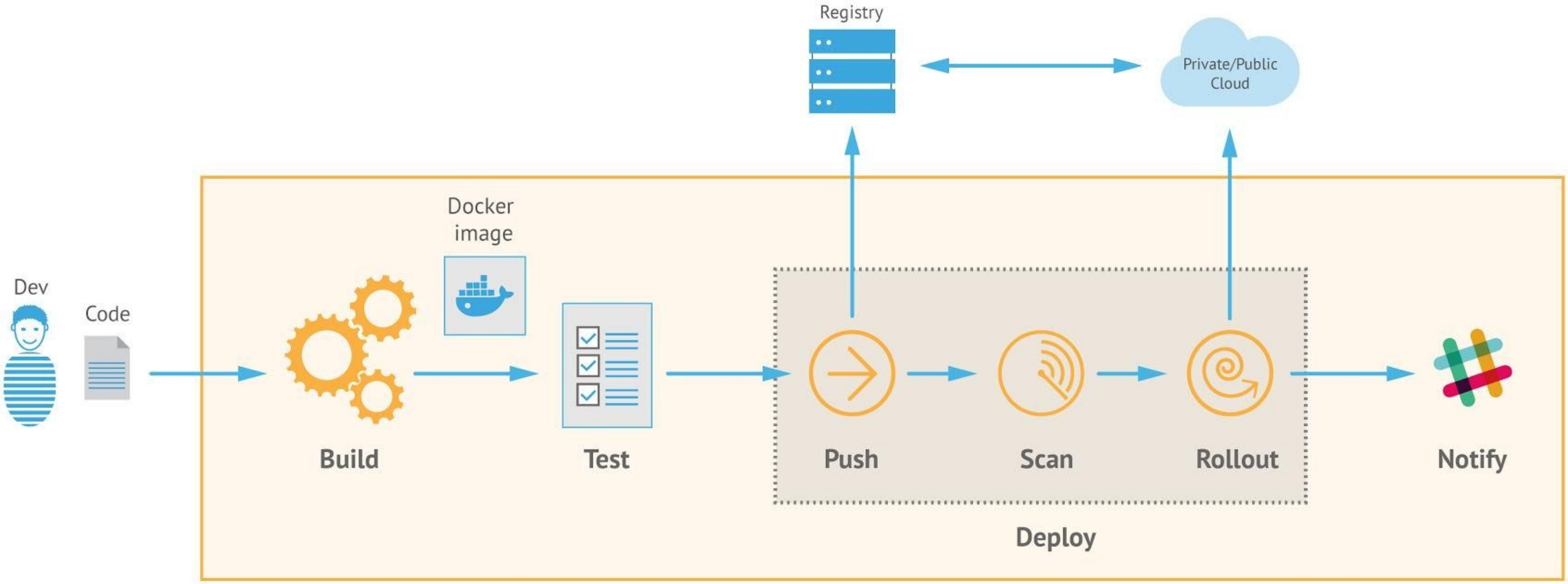


Hello! My name is Brian.

What is your name?

Architecture




Registry
Private/Public Cloud
Docker image
Dev
Code
Build
Test
Push
Scan
Rollout
Notify
Deploy

CI/CD System

"Bad" Identidock Dockerfile

```
FROM python

RUN pip install Flask uWSGI requests redis
WORKDIR /app
COPY app /app
COPY cmd.sh /

EXPOSE 9090 9191

CMD ["/cmd.sh"]
```

Evil No. 1

- No version numbers for software
- Breaks repeatability and provenance

Which version?

- Many packages use semver
  - MAJOR.MINOR.PATCH
- Too specific and risk missing security updates
- Too coarse and risk breaking changes
- Consider MAJOR.MINOR

"Versioned" Identidock Dockerfile

```
FROM python:3.6

COPY requirements.txt /requirements.txt
RUN pip install -r /requirements.txt
WORKDIR /app
COPY app /app
COPY cmd.sh /

EXPOSE 9090 9191

CMD ["/cmd.sh"]
```

requirements.txt

```
appdirs>=1.4,<1.5
certifi==2017.4.17
chardet>=3.0,<3.1
click==6.7
Flask>=0.12,<0.13
idna==2.5
...
```

Aside: Total Repeatabilty

> "Build tools must allow us to ensure consistency and repeatability"
>
> - Site Reliability Engineering

- Currently not possible with `docker build`
- Also packages can be a problem
  - Can run own mirror e.g. https://www.aptly.info/
- Bazel

Evil No 2. Not Setting a User

- Identidock is running as root
- Change to less privileged user

Identidock Dockerfile with User

```
FROM python:3.6

RUN groupadd -r identidock && useradd -r -g identidock identidock

COPY requirements.txt /requirements.txt
RUN pip install -r /requirements.txt
WORKDIR /app
COPY app /app
COPY cmd.sh /

USER identidock

EXPOSE 9090 9191

CMD ["/cmd.sh"]
```

Changing User at Start-up

```
#!/bin/sh
set -e


if [ "$1" = 'redis-server' -a "$(id -u)" = '0' ]; then
    chown -R redis .
    exec gosu redis "$0" "$@"
fi


exec "$@"
```

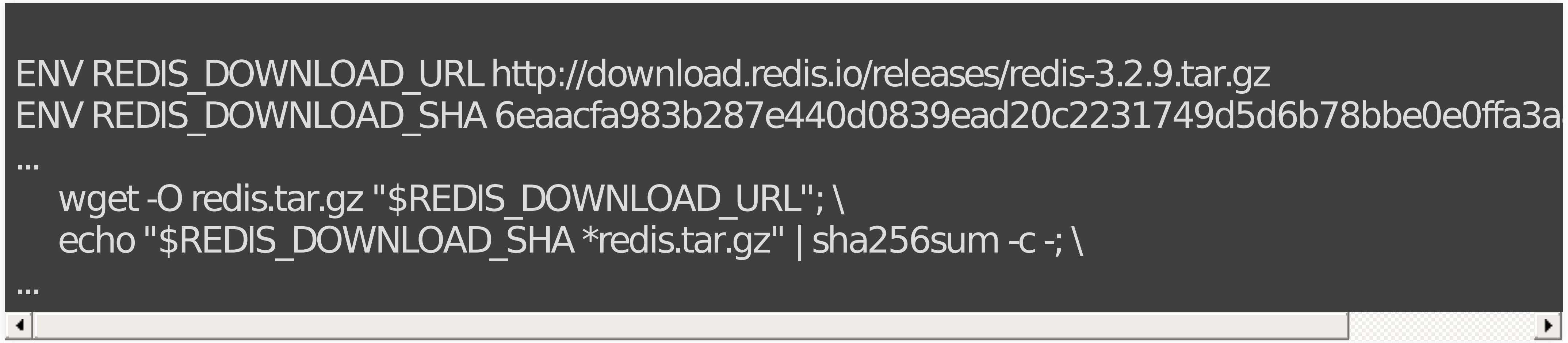
gosu

- sudo for containers
- https://github.com/tianon/gosu
- su-exec in Alpine

```
$ docker run -it debian-with-sudo sudo -u nobody ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0  41096  3048 ?        Ss+  20:05   0:00 sudo -u nobody
nobody       7  0.0  0.0  17500  2068 ?        R+   20:05   0:00 ps aux

$ docker run -it debian-with-gosu gosu nobody ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
nobody       1  0.0  0.0   9084   800 ?        Rs+  20:06   0:00 ps aux
```

Would-be Evil No 3. Not Verifying Downloads

- Doesn't occur in this Dockerfile
- Essential for Provenance



```
ENV REDIS_DOWNLOAD_URL http://download.redis.io/releases/redis-3.2.9.tar.gz
ENV REDIS_DOWNLOAD_SHA 6eaacfa983b287e440d0839ead20c2231749d5d6b78bbe0e0ffa3a
...
    wget -O redis.tar.gz "$REDIS_DOWNLOAD_URL"; \
    echo "$REDIS_DOWNLOAD_SHA *redis.tar.gz" | sha256sum -c -; \
...
```

https://github.com/docker-library/redis/blob/master/3.2/Dockerfile

Image Naming and Metadata

- Don't tag your images "latest"
- Add metadata for image provenance

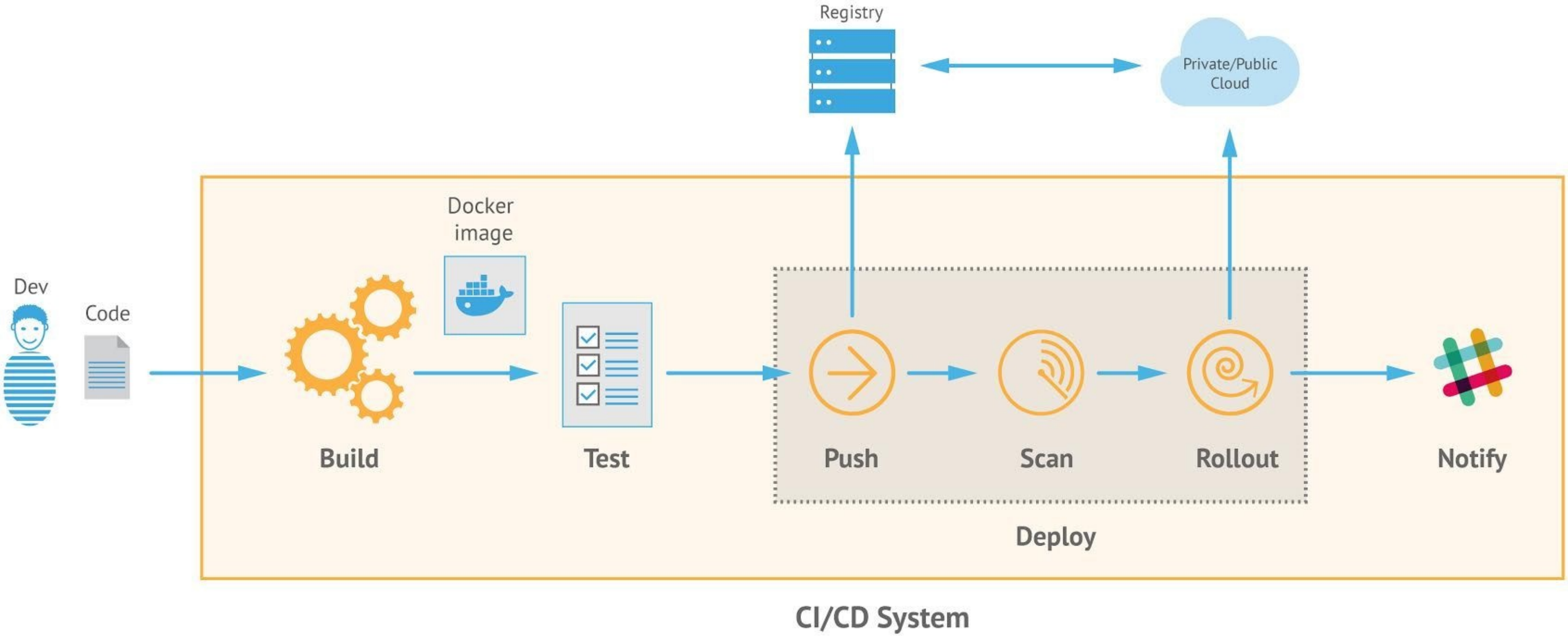
https://github.com/opencontainers/image-spec/blob/master/annotations.md

Dockerfile

```
FROM python:3.6

...

CMD ["/cmd.sh"]

#https://github.com/opencontainers/image-spec/blob/master/annotations.md
ARG CREATED
ARG REVISION
ARG NAME
LABEL org.opencontainers.image.created=$CREATED \
      org.opencontainers.image.revision=$REVISION \
      org.opencontainers.image.name=$TAG \
      org.opencontainers.image.source="git@github.com:amouat/identidock.git"
```

Build Script

```
TAG=identidock:v2.0.1
docker build -f Dockerfile_labelled \
        --build-arg CREATED="$(date –rfc-3339=s)" \
        --build-arg REVISION="$(git rev-parse HEAD)" \
        --build-arg TAG=$TAG \
        -t $TAG .
```


Registry
Private/Public Cloud
Docker image
Dev
Code
Build
Test
Push
Scan
Rollout
Notify
Deploy
CI/CD System

Pushing and Pulling Securely

- Not as easy as it sounds
  - Docker Content Trust
  - Digests

Docker Content Trust

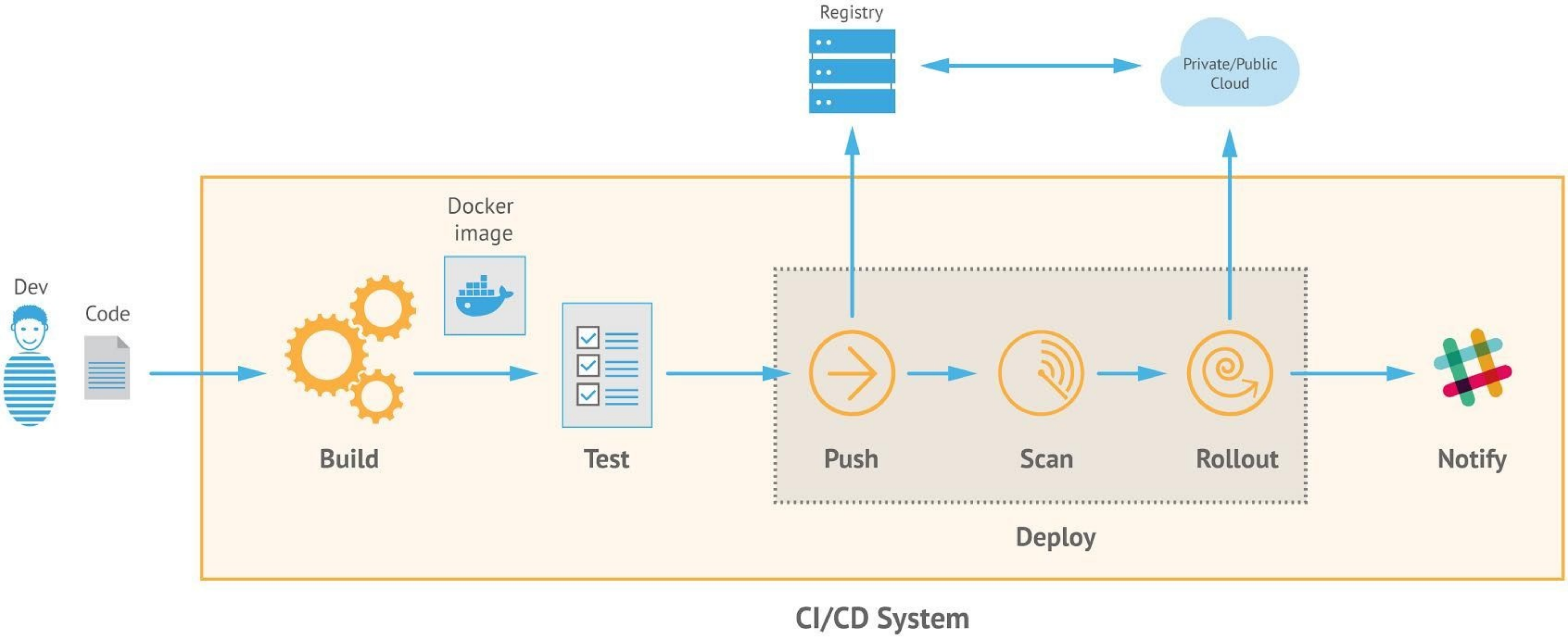
- Turn on with export DOCKER_CONTENT_TRUST=1
- Images can then be "signed"
- Pulled images checked against publishers public key
- Pushing images requires creation of signing keys
- "TOFU"
- Requires notary server
  - Probably Docker Hub

Digests

- Immutable content-based hash of image
- Can pull by digest
  - docker pull debian@sha256:72f784399fd2719b4\
    cb4e16ef8e369a39dc67f53d978cd3e2e7bf4e502c7b793

Digests

```
TAG=myregistry.com/identidock:v2.0.1
docker build -f Dockerfile_labelled \
        --build-arg CREATED="$(date –rfc-3339=s)" \
        --build-arg REVISION="$(git rev-parse HEAD)" \
        --build-arg TAG=$TAG \
        -t $TAG .

#Testing…

docker push $TAG

DIGEST=$(docker inspect -f '{{index .RepoDigests 0}}' $TAG)

#docker service update –image $DIGEST identidock
#kubectl set image …
```

Registry
Private/Public Cloud
Docker image
Dev
Code
Build
Test
Push
Scan
Rollout
Notify
Deploy
CI/CD System

The No 1. Vulnerability?

- Running out-of-date software

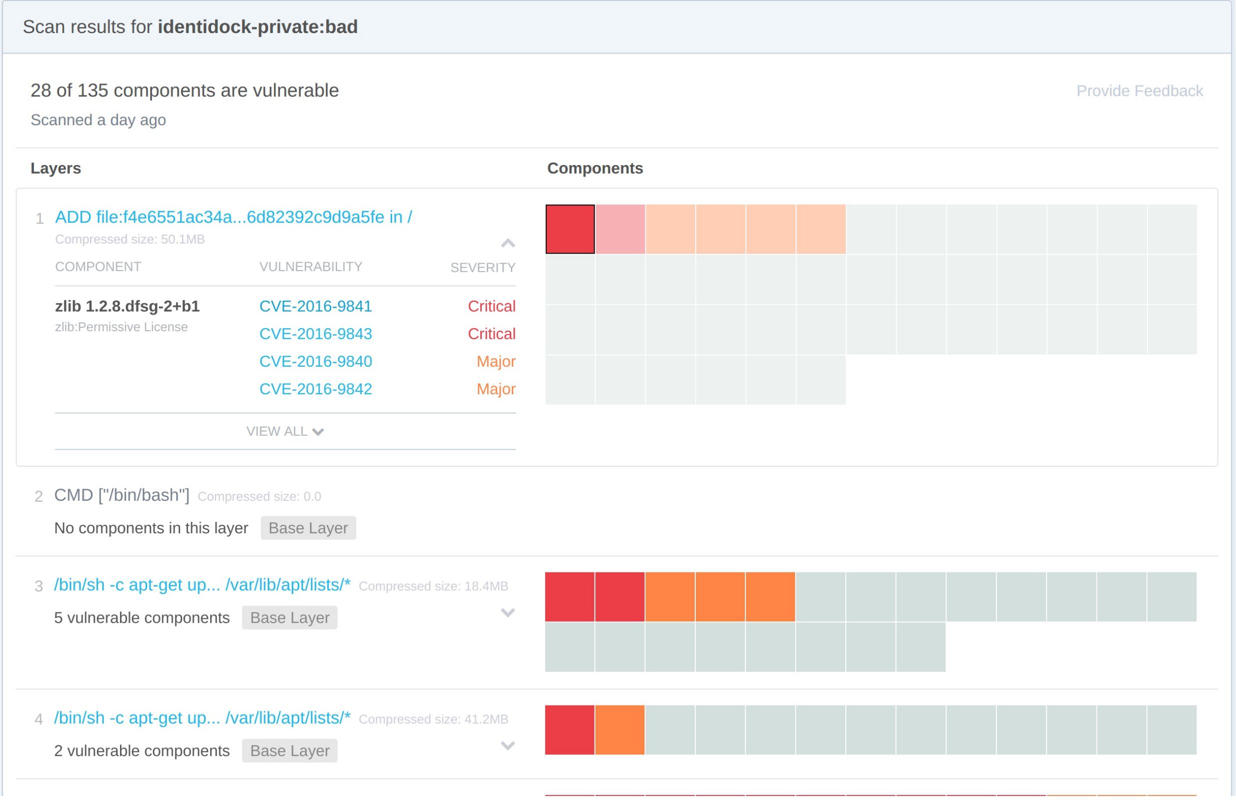
Don't Run Vulnerable Software

- Keep packages up to date
- Use a security scanner

Keep Packages up-to-date

- Use tooling
  - npm outdated, pip list --outdated
- Auto-builds & hooks
  - watchtower

Security Scanning



Scan results for **identidock-private:bad**

28 of 135 components are vulnerable

Scanned a day ago

Layers

Components

1 ADD file:f4e6551ac34a...6d82392c9d9a5fe in /

Compressed size: 50.1MB

COMPONENT	VULNERABILITY	SEVERITY
zlib 1.2.8.dfsg-2+b1 zlib:Permissive License	CVE-2016-9841	Critical
	CVE-2016-9843	Critical
	CVE-2016-9840	Major
	CVE-2016-9842	Major

2 CMD ["/bin/bash"] Compressed size: 0.0

No components in this layer Base Layer

3 /bin/sh -c apt-get up... /var/lib/apt/lists/* Compressed size: 18.4MB

5 vulnerable components Base Layer

4 /bin/sh -c apt-get up... /var/lib/apt/lists/* Compressed size: 41.2MB

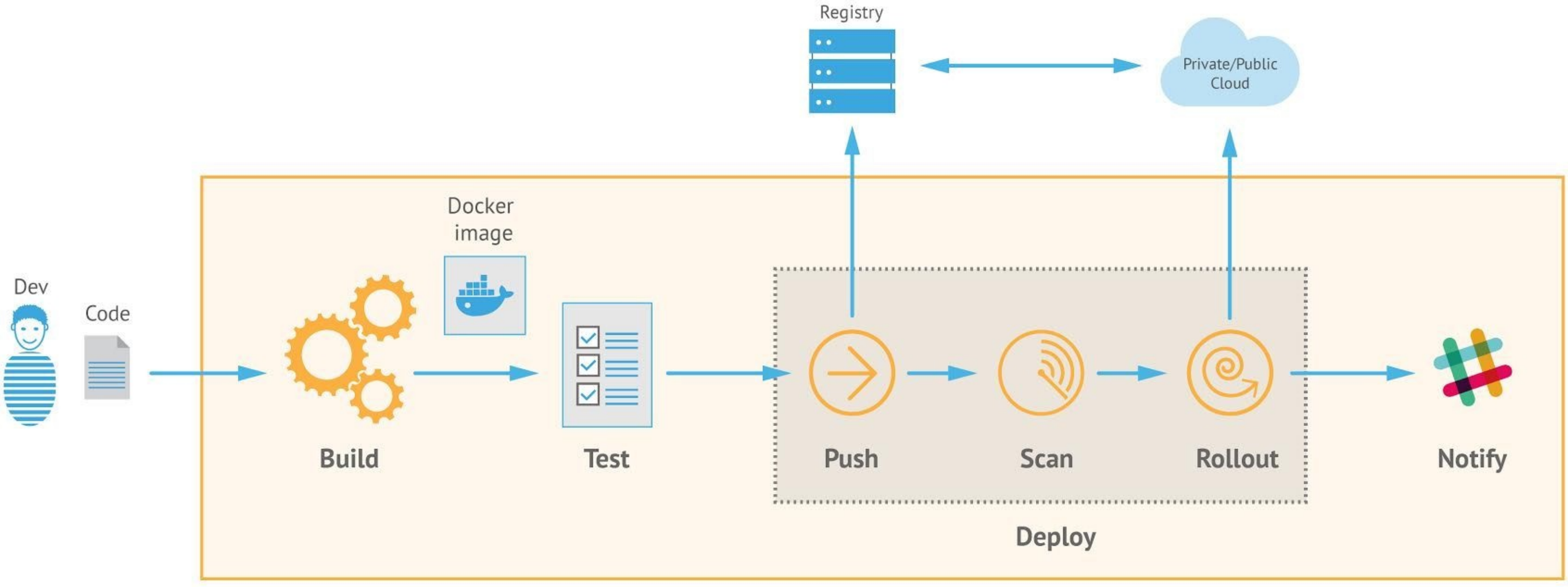
2 vulnerable components Base Layer

Scanning Services

- Clair
    - Open source
    - Designed to integrate into workflow
- Docker Security Scanning
- Neuvector
- Twistlock
- Aqua Security

Integrate into workflow

- Most tools are API based
    - scan automatically on push



Registry

Private/Public
Cloud

Docker
image

Dev

Code

Build

Test

Push

Scan

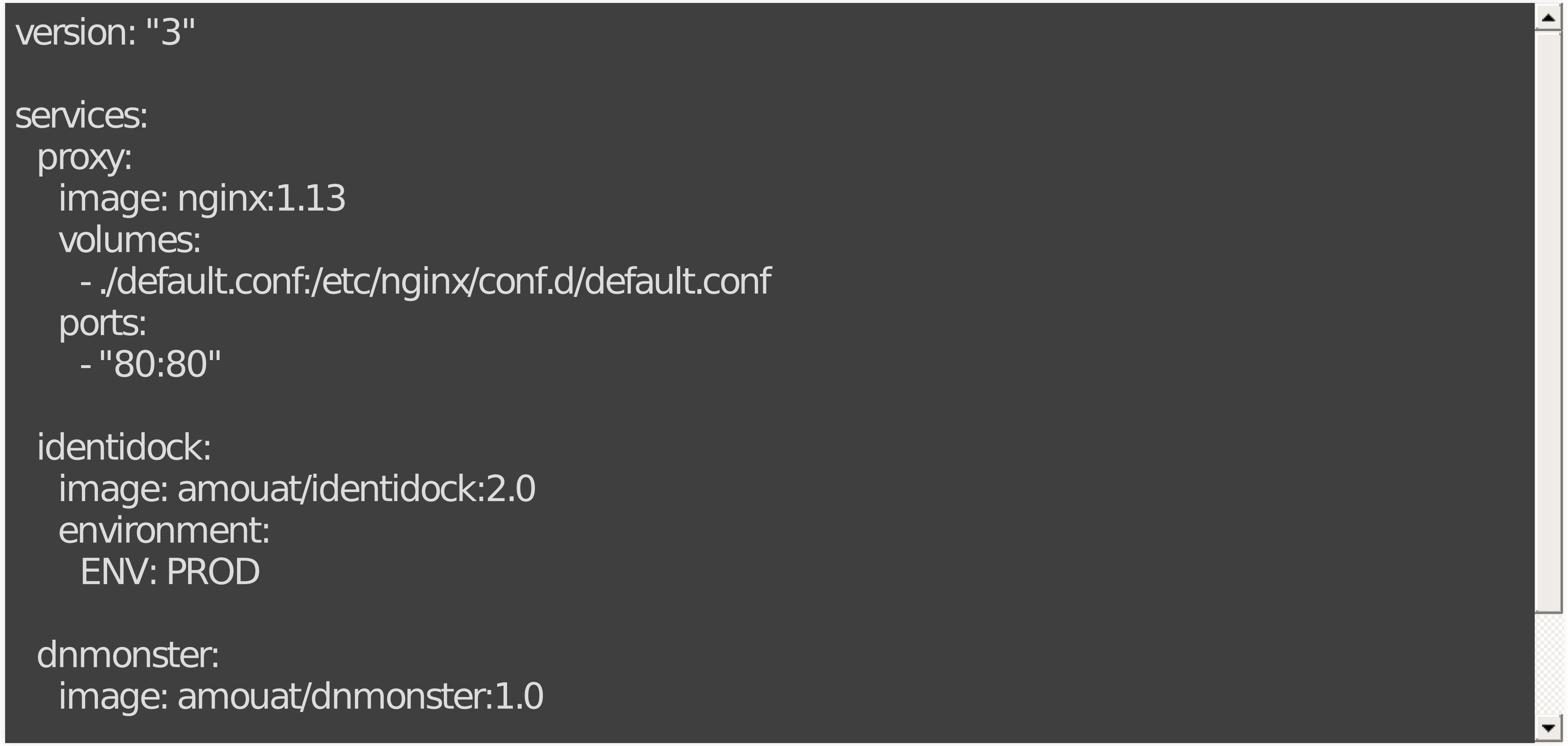
Rollout

Notify

Deploy

CI/CD System

Docker Compose



```
version: "3"

services:
  proxy:
    image: nginx:1.13
    volumes:
      - ./default.conf:/etc/nginx/conf.d/default.conf
    ports:
      - "80:80"

  identidock:
    image: amouat/identidock:2.0
    environment:
      ENV: PROD

  dnmonster:
    image: amouat/dnmonster:1.0
```

Read-only FS

```
$ docker run --read-only debian sh -c 'echo "x" > /file'
sh: 1: cannot create /file: Read-only file system
```

Read-only FS

- Can mount volumes for specific files

```
docker run -d -p 80:80 --read-only \
   --tmpfs /var/cache/nginx/ --tmpfs /run \
   nginx
```

Minimal distro

- debian 123MB
- alpine 5MB

Advantages

- Smaller attack surface
- Easier to distribute

Disadvantages

- Smaller package manager
- musl vs glibc
- Less debugging tools
- No bash
- Smaller set of maintainers?

Docker Compose Alpine

```
version: "3"

services:
  proxy:
    image: nginx:1.13-alpine

  ...

  redis:
    image: redis:3.2-alpine
```

Aside: Binary only containers

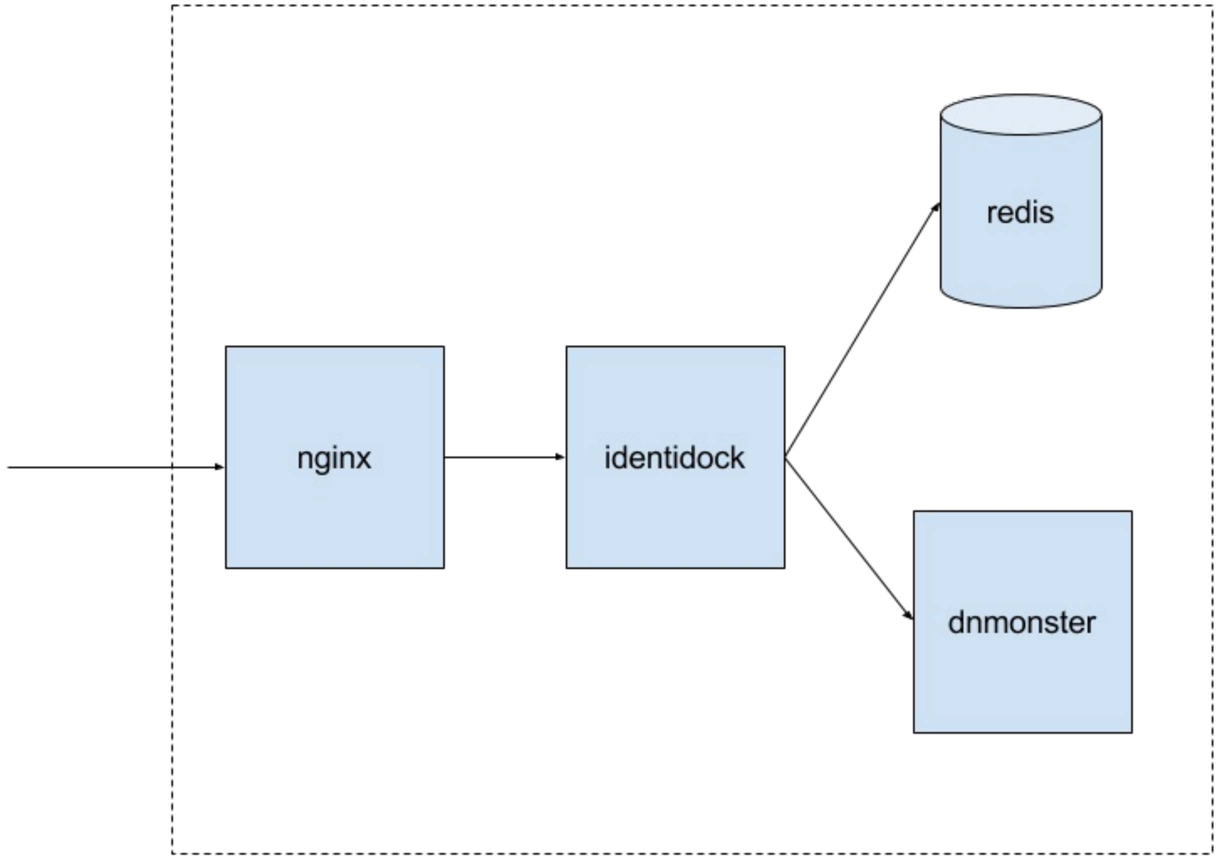
- Statically compile code
  - Go, C, Rust...
- Place into scratch image
- Super-minimal

Aside: Aside: Unikernels

- The Linux kernel is large
- Lot of it is uneeded
  - Floppy drivers?
  - Multitenancy
- Merge kernel and application
  - run on H/W or hypervisor

Network Segregation

- Redis and dnmonster don't talk to each other
- So they shouldn't be able to!



Network Segregation

```
services:
  proxy:
    ...
    networks:
      - frontend

  identidock:
    ...
    networks:
      - frontend
      - database
      - backend

  dnmonster:
    image: amouat/dnmonster:1.0
    networks:
      - backend
  redis:
    image: redis:3.2-alpine
    networks:
      - database

networks:
  - database
  - frontend
  - backend
```

Limiting Resources

- Memory is most important
- CPU shared by default

Limiting Resources

```
...
  redis:
    image: redis:3.2-alpine
    deploy:
      resources:
         memory: 200M

    networks:
      - database
...
```

Aside: Capabilities & Seccomp

- Limit system calls

Aside: Linux Security Modules

- AppArmor
- SELinux

Host Security

- Same as before
- Keep up-to-date
- Stick to what you know
- docker-bench

Aside: Container Distros

- RancherOS
- CoreOS
- Atomic
- LinuxKit

Aside: Secure Kernels

- GRSecurity
- PaX

Secrets

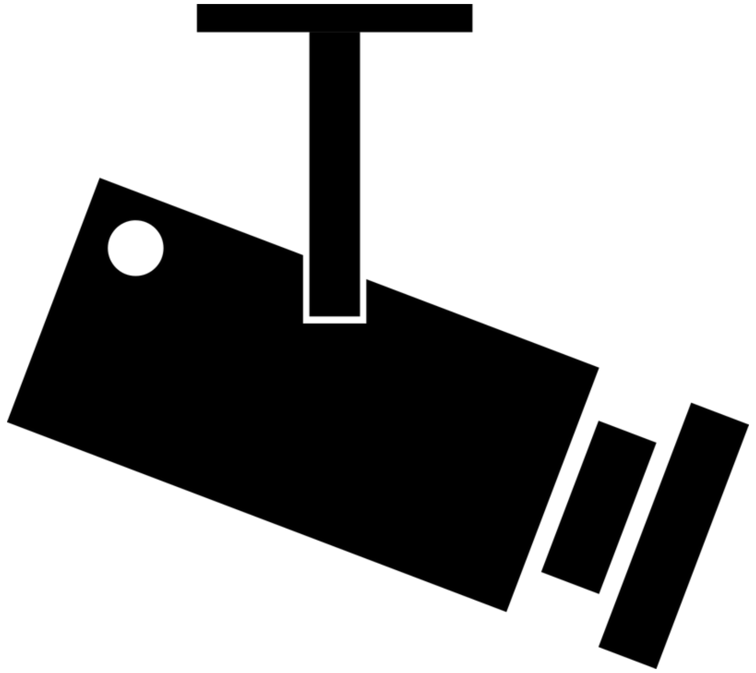
- Passwords, tokens, keys
- Can get tricky with ms

Secrets

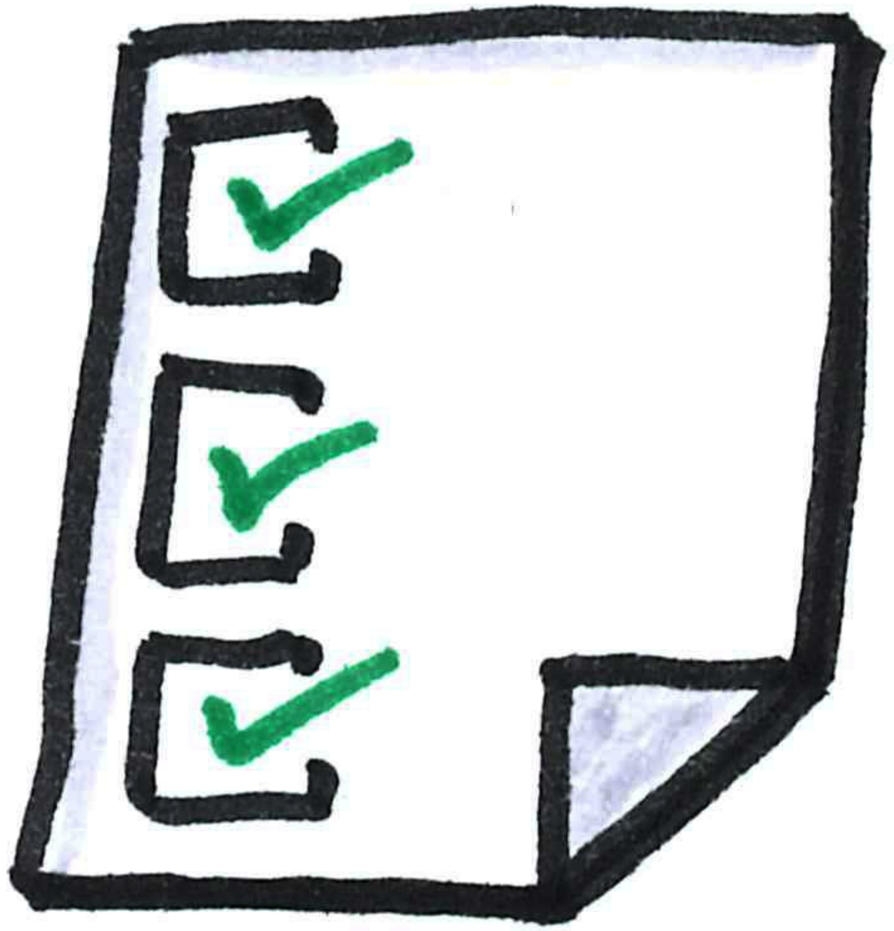
- Environment variables work
  - but kinda icky
- Swarm & Kubernetes have solutions
- Vault

Monitoring

- Essential with microservices
- lots of solutions
  - Prometheus



Checklist



Must

- Keep software updated
- Run as unprivileged user
- Establish provenance and repeatability

Should

- Run with read-only fs
- Scan for vulnerabilities
- Enforce network segregation
- Run minimal container distro

Could

- Use vault for secrets
- Restrict capabilities and resources
- Run a minimal host distro
- Run a security enhanced kernel

Conclusion

- Don't try to do everything at once
- Easy wins
- Containers add security