Please Sign In!

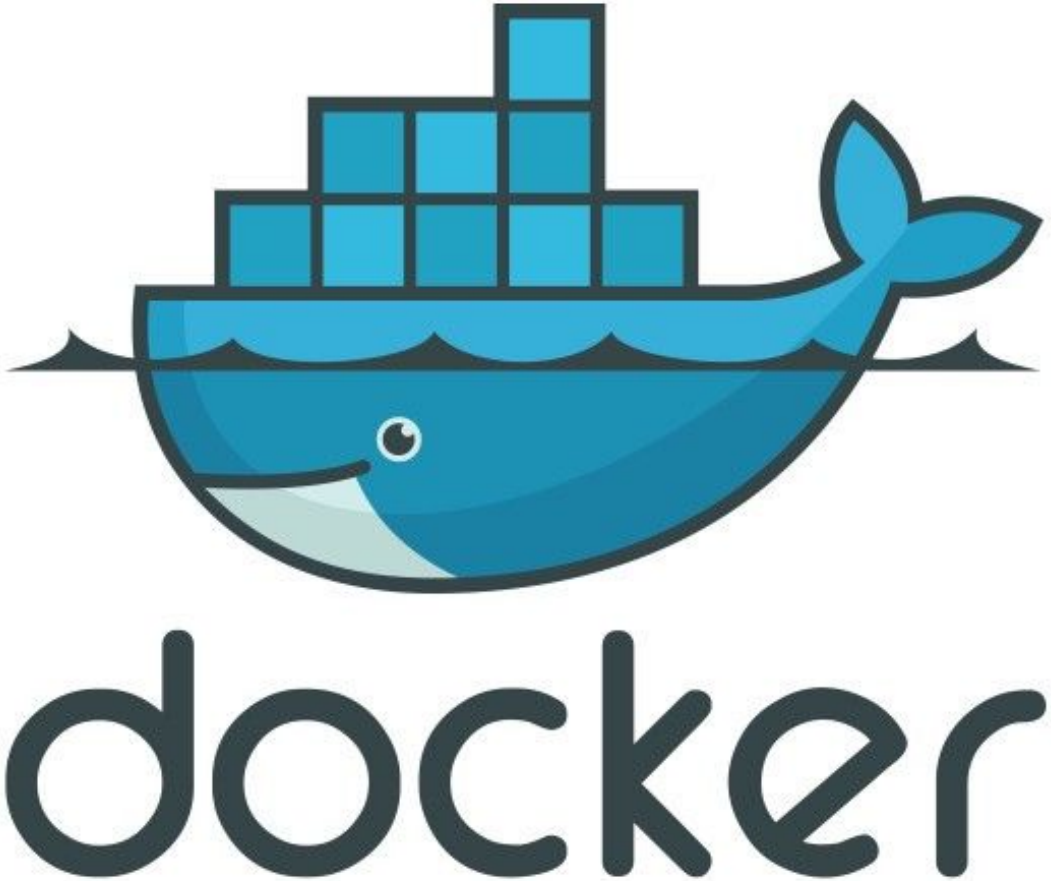
Club Login - ufsit.org/signin

ACM LOGIN - uf-acm.com/e/A4560A

Securing Docker Containers

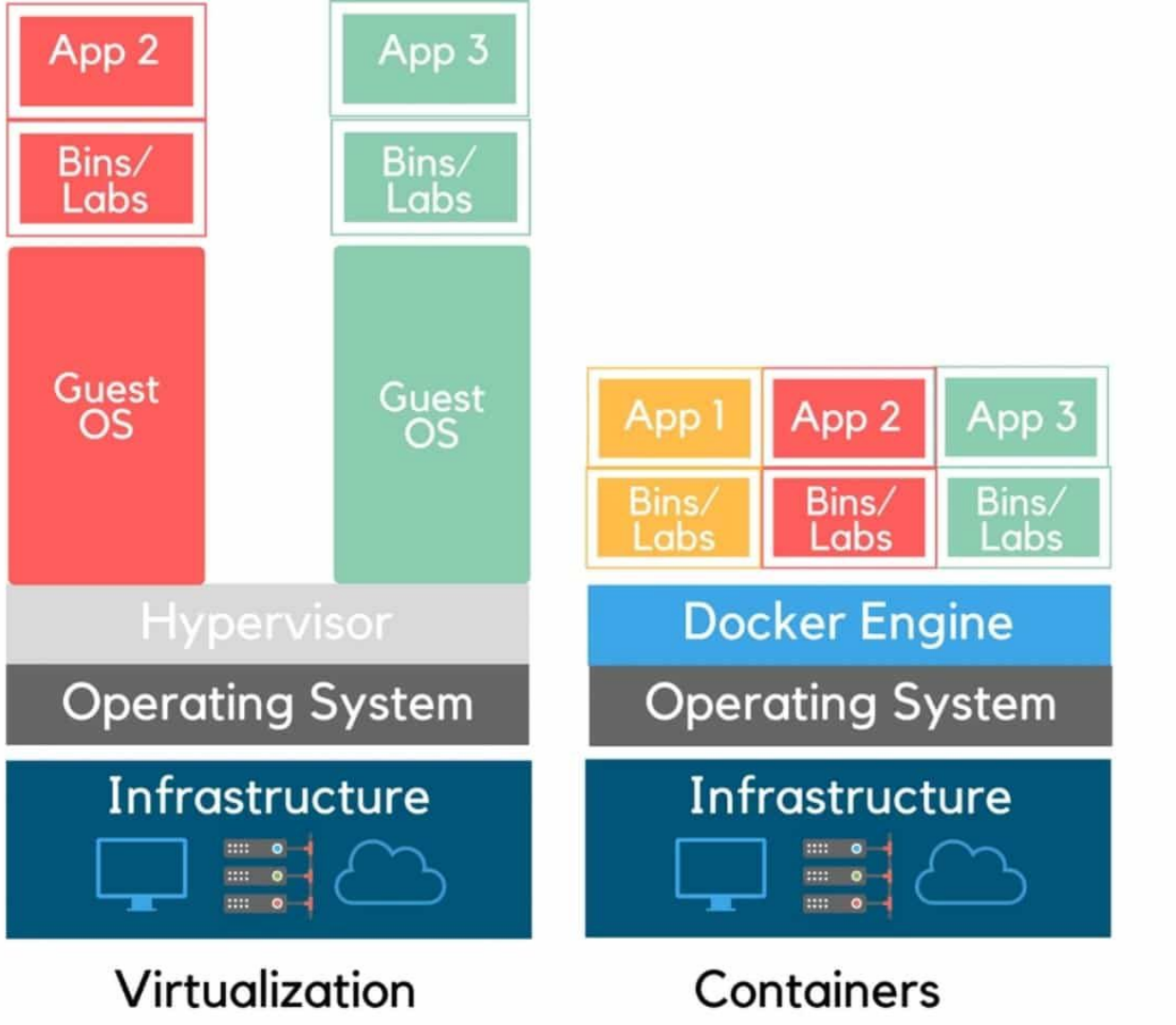
University of Florida InfoSec Team
Jack Polk
2020.02.06

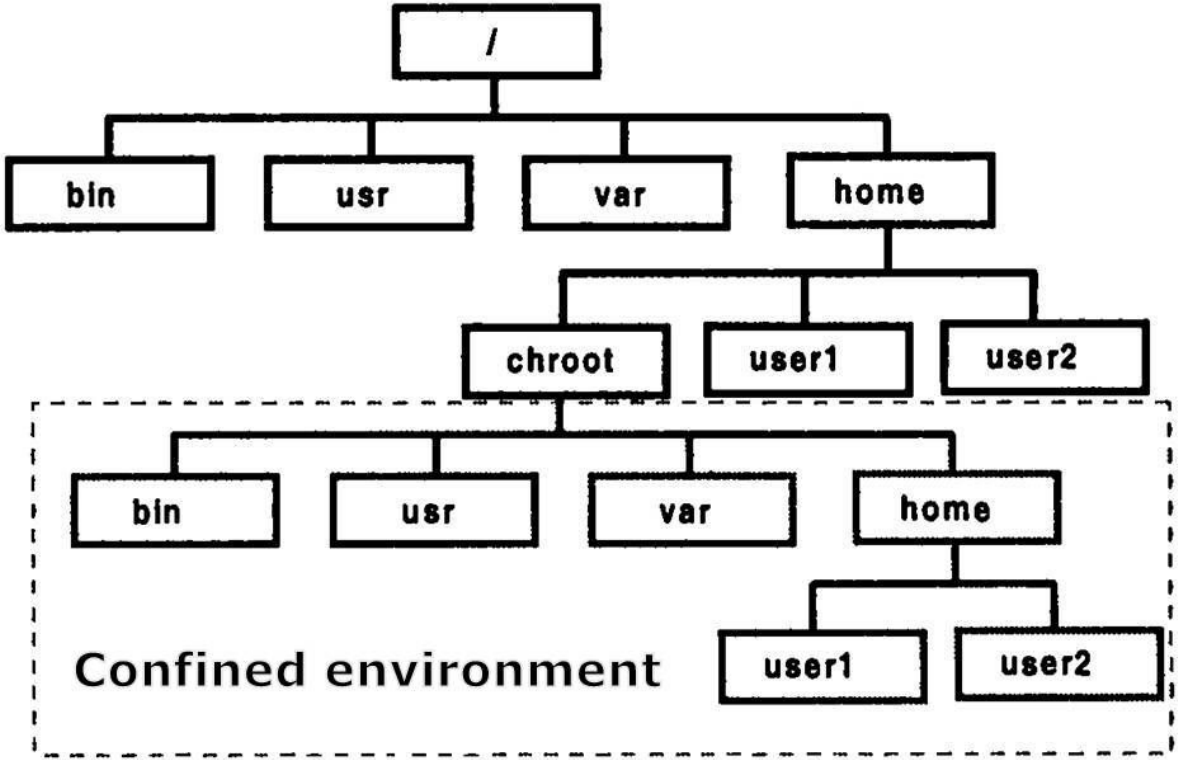




What is Docker?

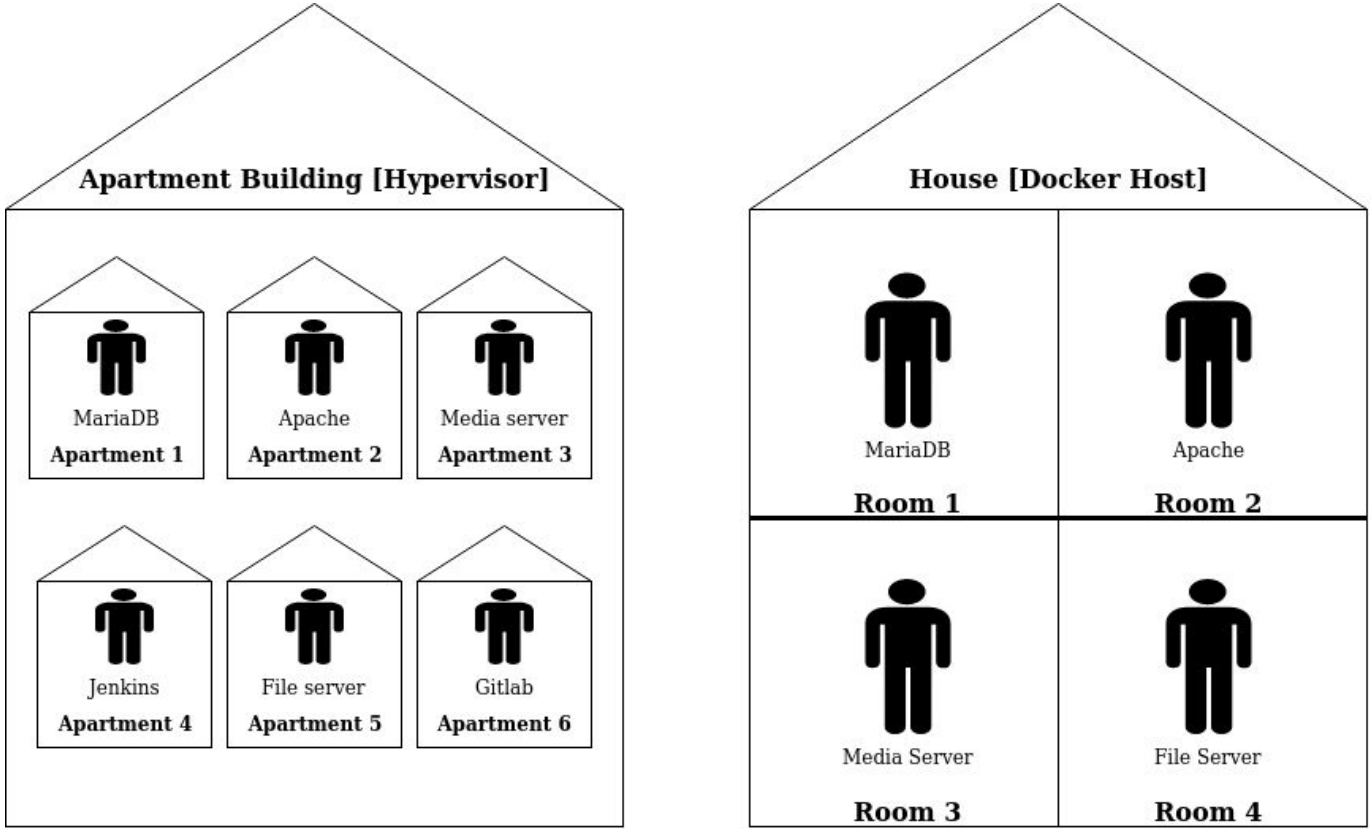
Docker is a platform for Windows, Mac, and Linux that provides a easy to use CLI and daemon for creating, packaging, and distributing Linux containers.


App 2
App 3
Bins/ Labs
Bins/ Labs
Guest OS
Guest OS
Hypervisor
Operating System
Infrastructure
App 1
App 2
App 3
Bins/ Labs
Bins/ Labs
Bins/ Labs
Docker Engine
Operating System
Infrastructure
Virtualization
Containers

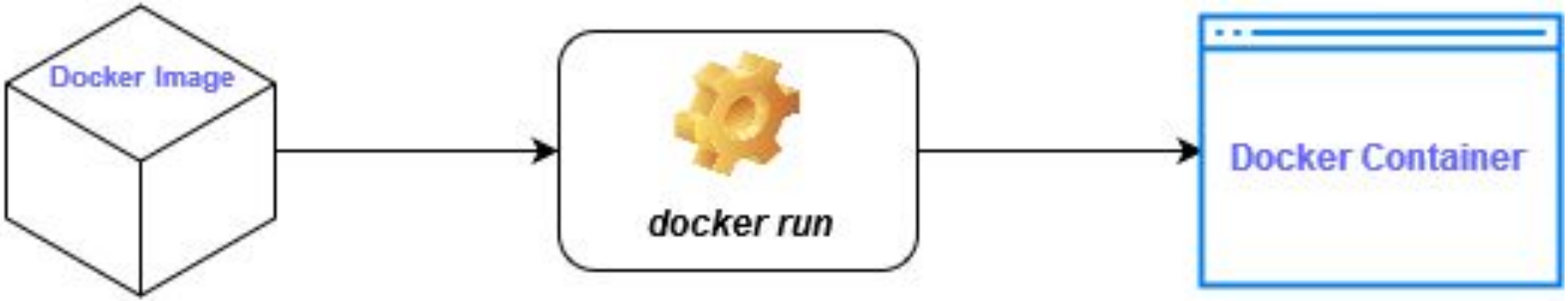


http://albertomatus.com/changing-working-root-directory/

chroot



Non-Technical Analogy



“Class” “Instantiation” “Object”

Object-Oriented Analogy

Installing Docker

Installation

Ubuntu / Debian:

```
apt install docker.io docker-compose
```

Enable Service:

```
systemctl enable docker && systemctl start docker
```

NOTE: Package repository must be updated. (`apt update` or `yum update`)

Docker Images

Image Nomenclature

{User}/{Image Name}:{Version}

Examples

- ubuntu:18.04
- python:3
- node
- alpine
- decaby7e/bind:latest
- 0xcaff/koel
- decaby7e/dhcpd:alpine-latest
- networkboot/dhcpd:1.0.2

Getting Images

Pull Existing Image

```
docker pull {image}:{version}
```

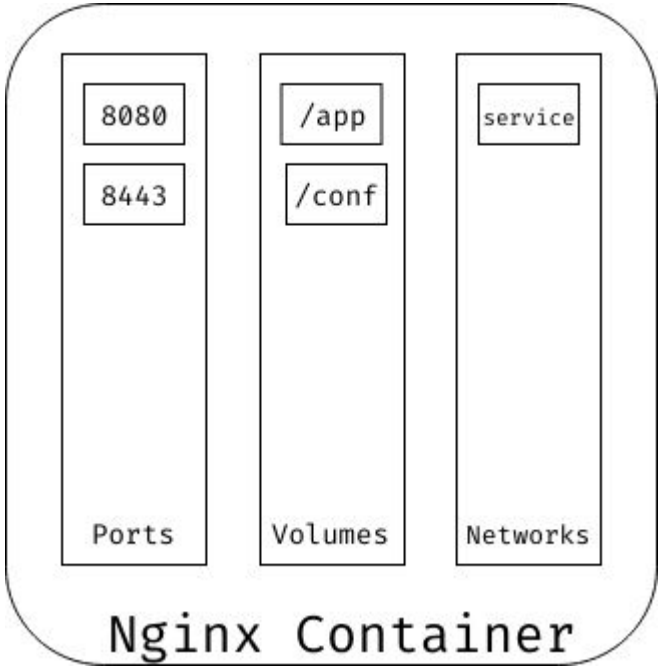
Exercise

Pull Ubuntu 18.04 to your Docker host

```
docker pull ubuntu:18.04
```

Docker Containers

Container Features



Ports

Ports bound from the host to the container

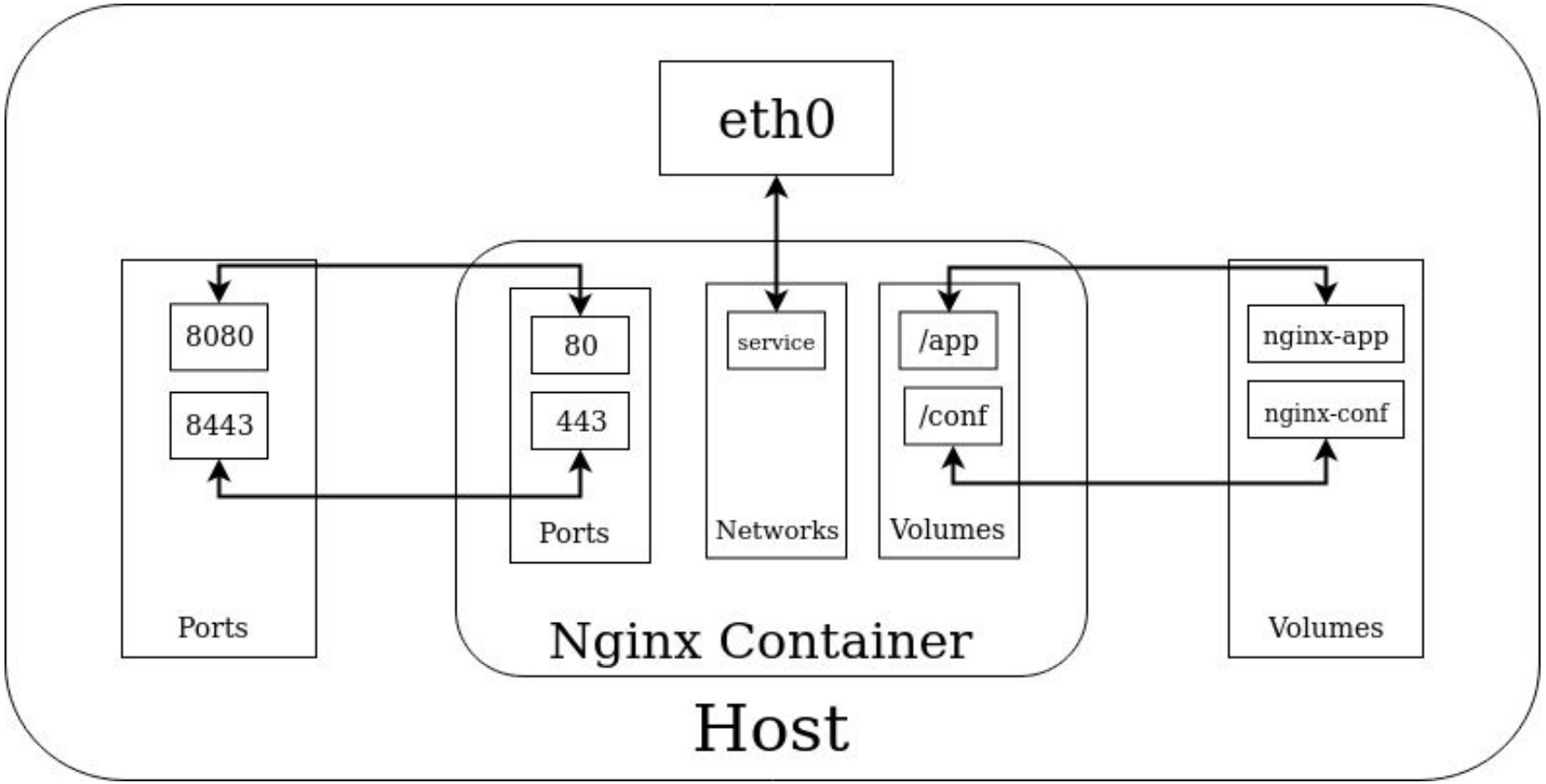
Volumes

Directories bound from the host to the containers

Networks

Interfaces connecting the host and container

Container-Host Interaction



Creating a Container

Create and Start Container

```
docker run -it --rm --name {container_name} {image} {command}
```

Create and Start Container (in background):

```
docker run -itd --rm --name {container_name} {image} {command}
```

Exercise

Create an Ubuntu container

```
docker run -it --rm --name demo ubuntu:18.04 bash
```

Interacting with Containers

Run Command in Container

```
docker exec -it {container_name} {command}
```

Exercise

Grab a file from the container and display it on the screen

```
docker exec -it demo cat /etc/hostname
```

Useful CLI Commands

View Running Containers

```
docker ps
```

Stop a container

```
docker stop {container_name}
```

Remove a container

```
docker rm {container_name}
```

View All Containers

```
docker ps -a
```

Kill a container

```
docker kill {container_name}
```

Resource Monitor

```
docker stats
```

Security Principles with Docker

Limiting Access to Host Filesystem

Limit Memory

```
docker run --memory=2048m
```

Limit CPU

```
docker run --cpus=1.5
```

Use Trusted Images and Always Inspect Dockerfiles

```
FROM ubuntu:18.04

ARG DEBIAN_FRONTEND=noninteractive

RUN apt-get -y update &&\
    apt -y install python python-pip

# Not copying app as we will mount
this as a volume later ...

ENTRYPOINT /app/manage.py dev-server
--password $PASSWORD
```

Good

```
COPY ubuntu:18.04

RUN apt install -y python python-pip
RUN pip install django

COPY app/        /app
COPY passwords/  /passwords
COPY static/     /static

ENTRYPOINT /app/manage.py dev-server
--password iLiKeCoOkIeS123
```

Not so Good

All Containers Run as Root by Default

The default user inside any container is **root** (UID = 0; Power = ∞ !!!)

Exercise

Run a container as a user other than root

Default Behavior

```
$ docker run -it --rm ubuntu:18.04
$ echo $UID
```

Custom Runtime User

```
$ docker run -it --rm --user 1000:1000 ubuntu:18.04
$ echo $UID
```

Limiting Resources to Containers

Limit Memory

```
docker run --memory=2048m
```

Limit CPU

```
docker run --cpus=1.5
```

Segregating Networking

Docker Group = Root = 🗿

Any user added to the group ‘docker’ immediately has root privileges. This is not to be taken lightly!

Users in this group do not require sudo to perform Docker CLI commands.

Attack possibility:

1. Mount host root as root in a container (docker run -v /:/host ...).
2. You now have root access on the host.

Summary

Alternative to Docker

Podman

podman

podman.io

Thank You!

Useful References:

docs.docker.com

My Blog:

blog.ranvier.net

Extra Slides

Build a Custom Image

```
docker build
{path_to_dockerfil
e} -t
{image}:{version}
```

Exercise Pull Ubuntu 18.04 to your Docker host

```
docker pull ubuntu:18.04
```