Containerization

JLab Software & Computing
Workshop 2023

docker

APPTAINER

Wesley Moore

Scientific Computing Operations Team

Thursday, May 18, 2023

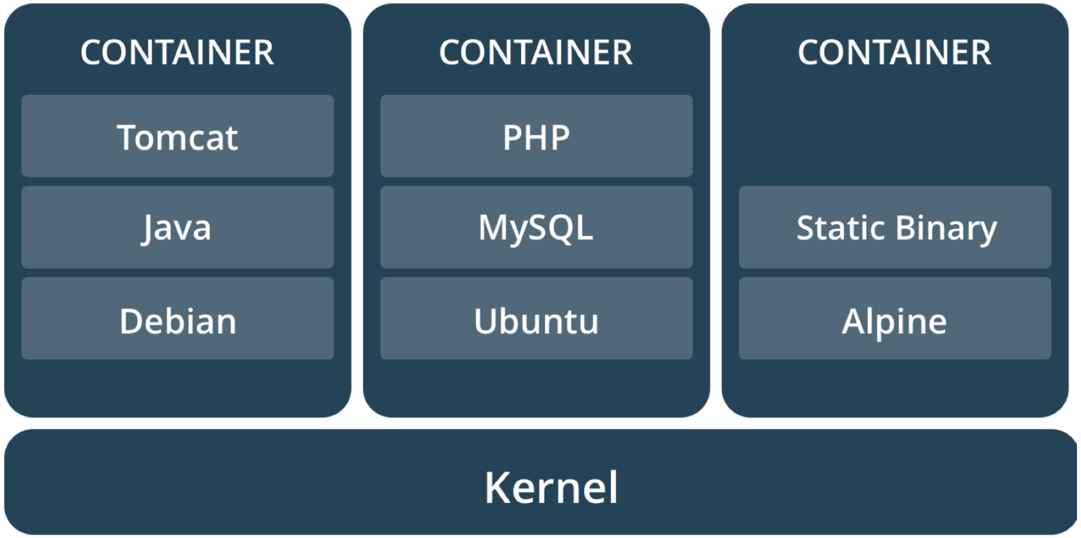
Jefferson Lab

U.S. DEPARTMENT OF ENERGY | Office of Science

JSA

What is a container?

> A container image is a lightweight, stand-alone, executable package of a piece of software that ***includes everything needed to run it:*** code, runtime, system tools, system libraries, settings.



*An **Image** is a file, essentially a container snapshot that produces a **container** when started. Same as VMDK compared to VM.*

https://docker.com/what-container/

What makes them so useful?

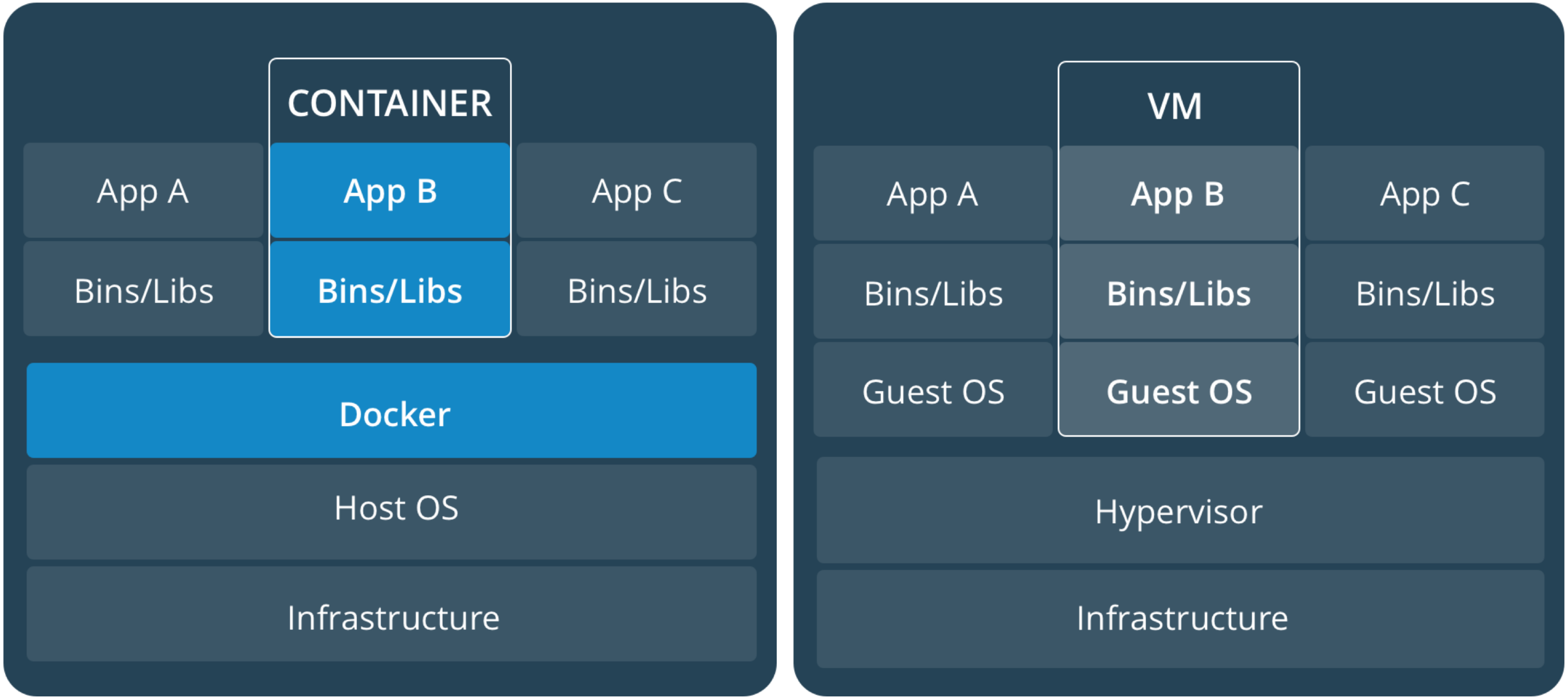
Containerization is increasingly popular because containers are:

- Flexible: Even the most complex applications can be containerized.
- **Lightweight: Containers leverage and share the host kernel.**
- **Interchangeable: You can deploy updates and upgrades on-the-fly.**
- **Portable: You can build locally, deploy to the cloud, and run anywhere.**
- Scalable: You can increase and automatically distribute container replicas.
- Stackable: You can stack services vertically and on-the-fly.

Lightweight

> Containers leverage and share the host kernel.

Containers vs VMs



CONTAINER

App A | App B | App C

Bins/Libs | Bins/Libs | Bins/Libs

Docker

Host OS

Infrastructure

VM

App A | App B | App C

Bins/Libs | Bins/Libs | Bins/Libs

Guest OS | Guest OS | Guest OS

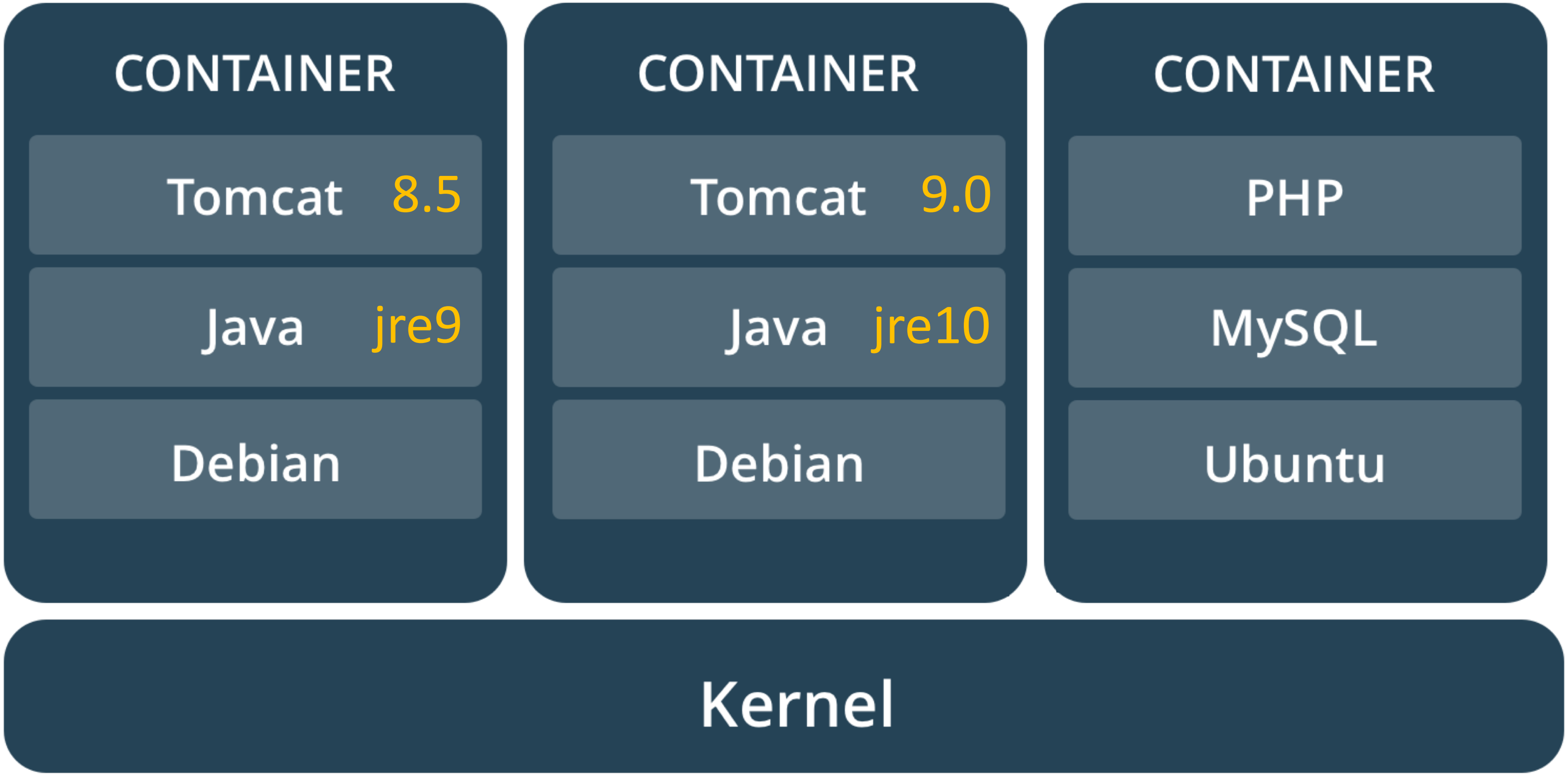
Hypervisor

Infrastructure

https://docs.docker.com/get-started/

Interchangeable

> You can deploy updates and upgrades on-the-fly.



CONTAINER
Tomcat 8.5
Java jre9
Debian

CONTAINER
Tomcat 9.0
Java jre10
Debian

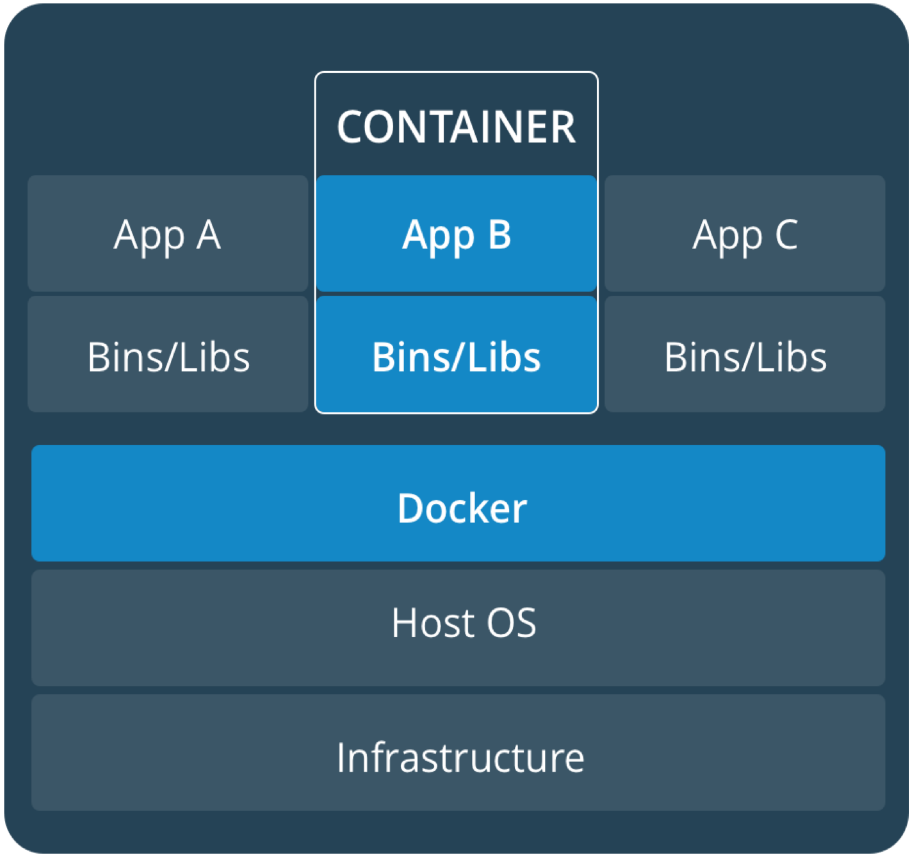
CONTAINER
PHP
MySQL
Ubuntu

Kernel

https://docker.com/what-container/

Portable

> You can build locally, deploy to the cloud, and run anywhere.



Host OS dependency is Docker Engine

- Server - dockerd
- REST API – specifies interfaces that programs use to talk to daemon
- Command line interface (CLI) client - docker

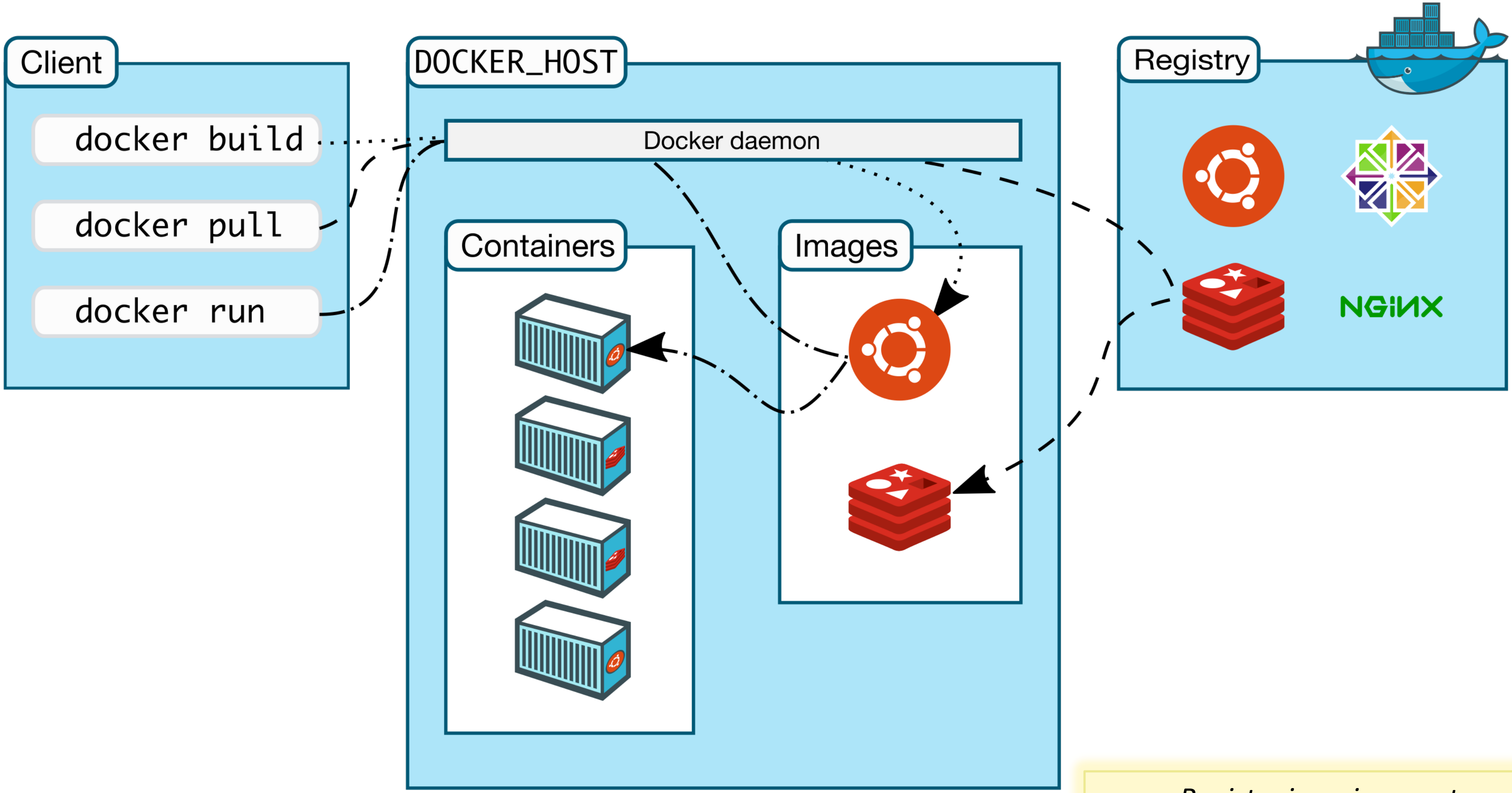
Security, less-is-more

Host OS can be stripped down

- Minimal attack vector
- Less patching involved

https://docs.docker.com/get-started/

Architecture



Registry is an image store

- *Docker Hub*
- *RedHat Registry*
- *On-prem solution*

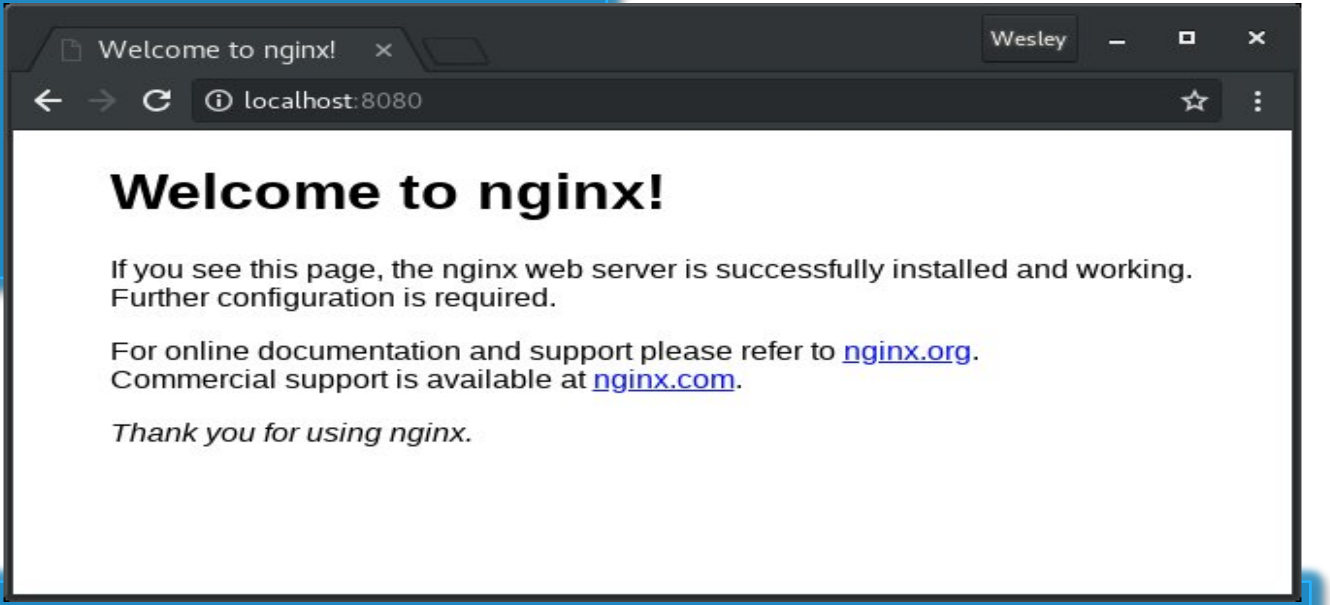
https://docs.docker.com/engine/docker-overview

Industry Example

```
$> sudo docker run --detach --publish 8080:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
…
Status: Downloaded newer image for nginx:latest
2296068eda542ec661b8f254756a8f8213f4a542e67e3a871bcd2af98229
```

NGINX is a web server, can be used as:

- *reverse proxy*
- *load balancer*
- *HTTP cache*



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

```
$> sudo docker ps
CONTAINER ID   IMAGE    COMMAND                  CREATED          STATUS         PORTS                    NAMES
2296068eda54   nginx    "nginx -g 'daemon of…"   3 seconds ago    Up 3 seconds   0.0.0.0:8080->80/tcp     practical_curie
```

Compiling code Example

```
# Dockerfile
FROM ubuntu:14.04
MAINTAINER Shane Canon scanon@lbl.gov

RUN  apt-update –y &&\
     apt-get install -y build-essential

ADD  . /myapp

RUN  cd /myapp && \
     make && make install
```

Parent image

Update packages and install dependencies

Copy in the application

Compile it!

```
> docker build –t scanon/myapp:1.1 .
> docker push scanon/myapp:1.1
```

Image tags

https://github.com/NERSC/ecp-container-tutorial

XRootD Example

```
# Dockerfile
FROM centos:centos7

RUN  yum update -y &&\
     yum install -y epel-release &&\
     yum install -y xrootd-client &&\
     yum clean all

ENV LD_PRELOAD=/usr/lib64/libXrdPosixPreload.so

CMD /bin/bash
```

Parent image *(e.g., latest centos7 image)*

Patch and install xrootd-client

Remove cached files, helps reduce image size

Set environment variable

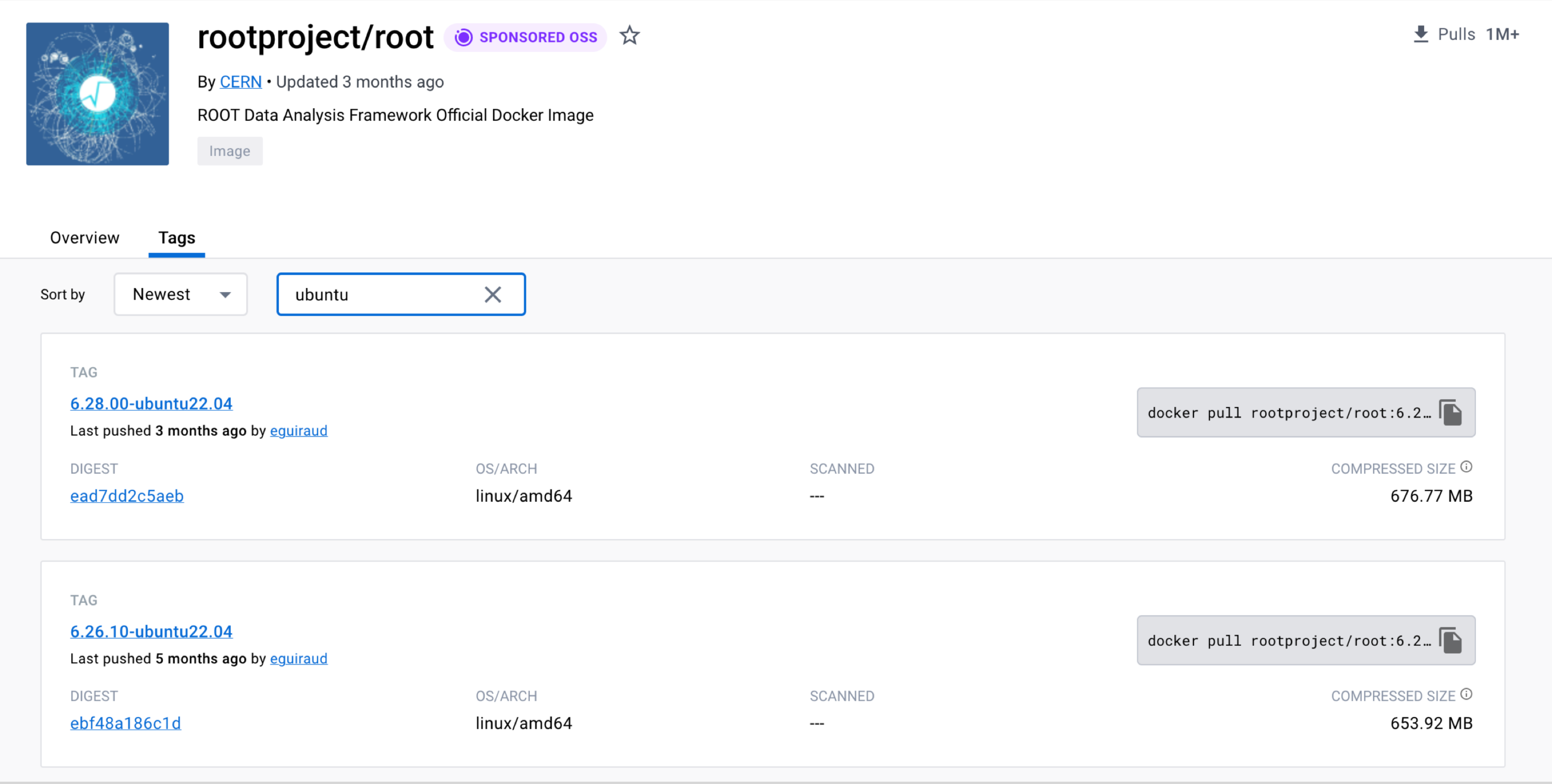
Default to bash shell at runtime

```
[wmoore@photon]$ docker build -t wmoore28/xrootd:latest .
[wmoore@photon]$ docker run -it wmoore28/xrootd:latest
[root@30fc989195d0 /]# ls xroot://sci-xrootd.jlab.org//osgpool/halld/random_triggers/
offmon-2018_01-ver11   recon-2018_01-ver02.2  recon-2019_11-ver01
offmon-2018_08-ver07   recon-2018_08-ver00    recon-2021_08-ver00
recon-2017_01-ver02    recon-2018_08-ver01    recon-2021_08-ver01
```

Image Tags

Tags are aliases for image IDs.

Example usage from DockerHub:



rootproject/root

SPONSORED OSS ☆

Pulls 1M+

By CERN • Updated 3 months ago

ROOT Data Analysis Framework Official Docker Image

Image

Overview | Tags

Sort by: Newest | ubuntu

TAG

6.28.00-ubuntu22.04

Last pushed **3 months ago** by eguiraud

```
docker pull rootproject/root:6.2…
```

DIGEST	OS/ARCH	SCANNED	COMPRESSED SIZE
ead7dd2c5aeb	linux/amd64	---	676.77 MB

TAG

6.26.10-ubuntu22.04

Last pushed **5 months ago** by eguiraud

```
docker pull rootproject/root:6.2…
```

DIGEST	OS/ARCH	SCANNED	COMPRESSED SIZE
ebf48a186c1d	linux/amd64	---	653.92 MB

Docker is great…right?

- Easy to install (linux, mac, windows)
- Well supported/documented
- Share your work with DockerHub

BUT…

- Requires elevated privileges
- Not allowed on HPC clusters

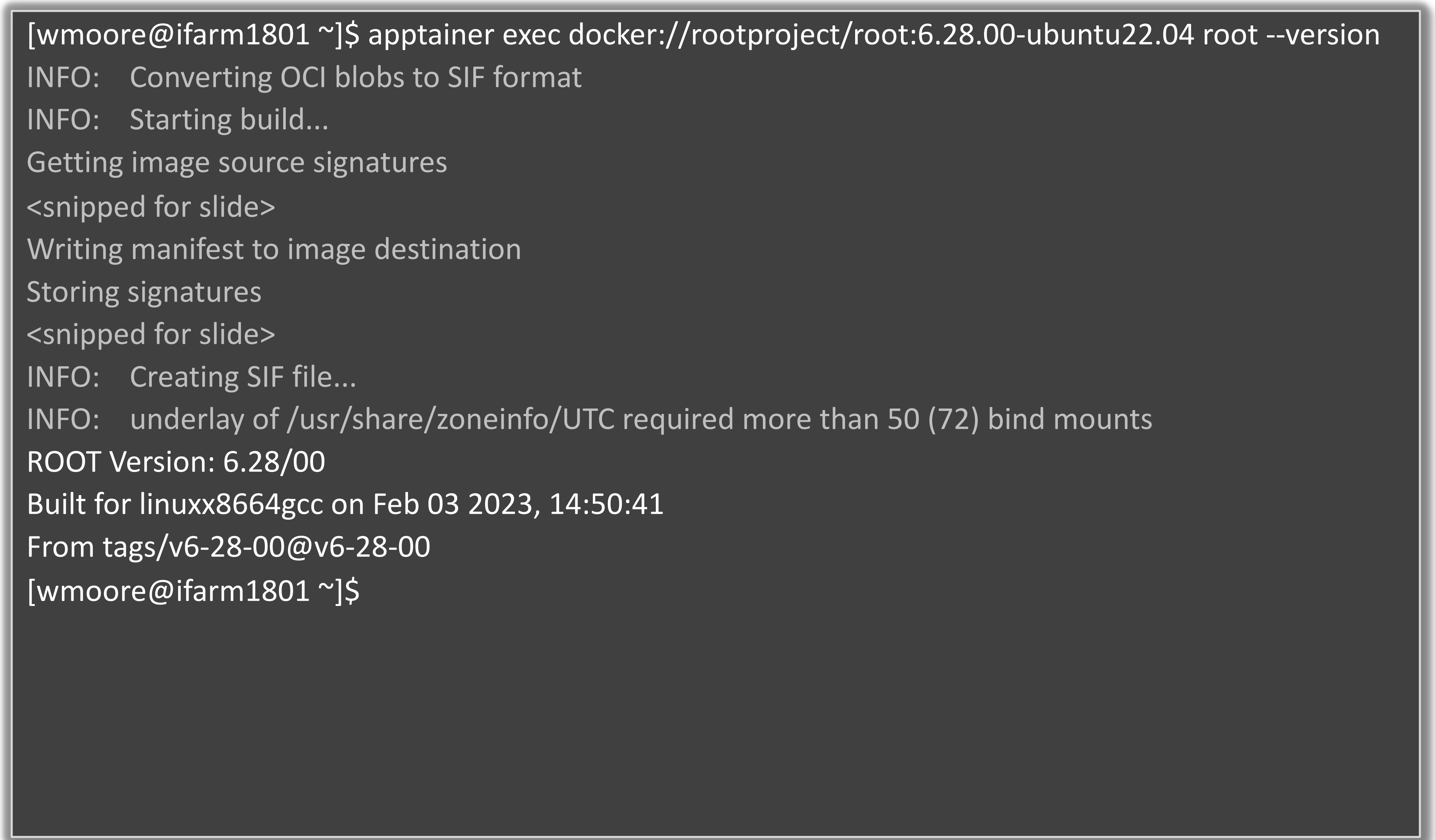
Apptainer *(formerly Singularity)*

- First released 2016 (then Singularity)
- Single file based container images (.sif)
  - Write to disk and reuse
  - Doesn’t require downloads across multiple jobs/nodes
- Restricted user permissions (inside user == outside user)
- No daemon on localhost
- Only needs apptainer installed

AND…

- Accepted by HPC clusters

Apptainer Example – Using DockerHub



```
[wmoore@ifarm1801 ~]$ apptainer exec docker://rootproject/root:6.28.00-ubuntu22.04 root --version
INFO:    Converting OCI blobs to SIF format
INFO:    Starting build...
Getting image source signatures
<snipped for slide>
Writing manifest to image destination
Storing signatures
<snipped for slide>
INFO:    Creating SIF file...
INFO:    underlay of /usr/share/zoneinfo/UTC required more than 50 (72) bind mounts
ROOT Version: 6.28/00
Built for linuxx8664gcc on Feb 03 2023, 14:50:41
From tags/v6-28-00@v6-28-00
[wmoore@ifarm1801 ~]$
```

Apptainer Example – Using local image file

```
[wmoore@ifarm1801 ~]$ apptainer exec /scigroup/scicomp/jupyterhub/ai/ai-notebook.sif python3
Python 3.8.6 | packaged by conda-forge | (default, Dec 26 2020, 05:05:16)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf
>>> print(tf.__version__)
2.9.3
>>> exit()
[wmoore@ifarm1801 ~]$
```

Apptainer Key Options

Use --*bind/-B* for including host filesystems:

```
[wmoore@ifarm1801 ~]$ apptainer exec --bind /work docker://wmoore28/xrootd:latest bash
INFO:    Using cached SIF image
Apptainer> ls /work/
accel          clas12         eic3      hallc     hu              osgpool      xrootd
casa           clas-old       epsci     halld     indra-astra     positron
cebaf24gev     data_science   fel       halld2    JAM             proposals
cfdfac         eic            halla     halld3    jpac            radcon
clas           eic2           hallb     hpqm      muscn           test-xrootd
```

Use --*nv* for including NVIDIA support *(gpus only available on sciml nodes)*:

```
[wmoore@ifarm1801 ~]$ apptainer exec -h | grep -i nvidia
    --nv                        enable Nvidia support
    --nvccli                     use nvidia-container-cli for GPU
```

Apptainer build

You may have issues with your home quota…

To workaround that, set your CACHEDIR and TMPDIR

```
[wmoore@ifarm1801 ~]$ setenv SINGULARITY_CACHEDIR /scratch/$USER/singularity
[wmoore@ifarm1801 ~]$ setenv SINGULARITY_TMPDIR /scratch/$USER/singularity

[wmoore@ifarm1801 ~]$ apptainer build myapp.sif docker://wmoore28/myapp:latest
```

Questions?

Wesley Moore

wmoore@jlab.org

Take aways…

- Build once, run everywhere
  - Bring Your Own Environment
  - Cluster independence (JLab, OSG, etc)

- Docker for an easy development environment
- Apptainer for use on compute clusters