# Hands on portion

- Set port forwarding on “default” in VirtualBox so that 8080 on host is forwarded to 8080 on VM.

# docker pull ubuntu

- Execute “docker pull ubuntu”
- This loads an image from the docker library
- The image contains bare copy of ubuntu

# docker images

- Execute “docker images”
- This generates a list of images known to Docker on your machine
- You should see ubuntu

# docker run –i –t ubuntu

- Execute docker run –i –t Ubuntu
- This executes an image. An executing image is called a “container”.
- You are now inside the container.
- Execute “ls”.
  - A directory structure is set up but only a bare bones OS has been loaded

# Install software on container

Execute

apt-get update

apt-get install wget

apt-get install nodejs

apt-get install npm

<cntl d>

This installs the software you will use during this session and exits the container

# docker ps –a

Execute “docker ps –a”

This generates a list of all of the containers that have been run

# Output from docker ps -a

| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | PORTS | NAMES |
|---|---|---|---|---|---|---|
| 174268c64fbd | ubuntu | "/bin/bash" | 7 minutes ago | Exited (0) About a minute ago | | sharp_mcnulty |

# docker commit sharp_mcnulty workshop

Note that the ubuntu container has a name of “sharp_mcnulty” (on my machine). It will be different on yours.

“docker commit sharp_mcnulty workshop” creates an image with the name workshop

# Execute “docker images”

| REPOSITORY | TAG | IMAGE ID | CREATED | SIZE |
|---|---|---|---|---|
| workshop | latest | a70567971230 | 13 seconds ago | 456 MB |
| ubuntu | latest | 0ef2e08ed3fa | 8 days ago | 130 MB |

# Execute “docker run –i –t workshop”

You are back inside a container. Load application:

wget https://raw.githubusercontent.com/cmudevops/ipshow.js/master/initialization_script

wget https://raw.githubusercontent.com/cmudevops/ipshow.js/master/ipshow.js

# Exit the container - <cntl d>

# List containers

```
$docker ps -a
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS                      PORTS               NAMES
9c4b32145fa3        workshop            "/bin/bash"              2 minutes ago       Exited (0) 8 seconds ago                        reverent_lewin
174268c64fbd        ubuntu              "/bin/bash"              30 minutes ago      Exited (0) 24 minutes ago                       sharp_mcnulty
```

# Make an image called ipshow

docker commit reverent_lewin ipshow

$ docker images

| REPOSITORY | TAG | IMAGE ID | CREATED | SIZE |
|---|---|---|---|---|
| ipshow | latest | 8f7afedea65d | 6 seconds ago | 456 MB |
| workshop | latest | a70567971230 | 11 minutes ago | 456 MB |
| <none> | <none> | b348af319cbc | 21 minutes ago | 456 MB |
| ubuntu | latest | 0ef2e08ed3fa | 8 days ago | 130 MB |





 

# Execute app

docker run –i –t –p 0.0.0.0:8080:8080 ipshow /bin/bash /initialization_script

(pay attention to blanks – between ipshow and /bin and between bash and /inititalization_script)

In browser: localhost:8080

You should see three ip addresses in the browser:

- Ip address of local host
- 127.0.0.1 (conventionally this is local host)
- Ip address of container

# What have we seen

- Distinction between docker images and containers
- Creating a docker image in layers
- Provisioning the docker image from the internet