DOCKER CONTAINIZATION INSTALLATION

EMMANUEL APIETU : DOCKER DOWNLOAD WALK THRU

DOCKER DOWNLOAD WALK THRU

1. Download docker for preferred system macOS or windows
2. Open terminal sudo apt update, sudo apt upgrade
3. Docker -v
4. Test command docker run hello-world this confirms successful download
5. Then we pull an image docker pull centos or nginx
6. Container is a running instance from an image
7. Command to create a container docker run -d –name <name> centos
8. Command with localhost:port docker run -d -p 8080:80 centos
9. Go to url type in localhost8080:80 should send you to web site
10. Docker ps shows list of containers running
11. To remove containers, it has to be stopped
12. Now you could open two localhost to a container: docker run -d -p 8081:80 -p 3000:80 centos. This allows both localhost into the container port 80
13. A file could also be created and shared in container.

DOCKER CONTAINERIZATION INSTALLATION

EMMANUEL APIETU P2 TKH DOCKER INSTALLATION

There are three linux permissions: read, write and execute, which is assigned to owner, group and others. The attributes -r (read) let's you read -w (write) Allows modification of file - - (no permission)

The principles of least privilege is given minimum access or permission needed to perform a particular task or access to restricted resources

IAM policies that are put in place to give the right people access to certain resources in an organization. This prevents risk access, access management and secure data

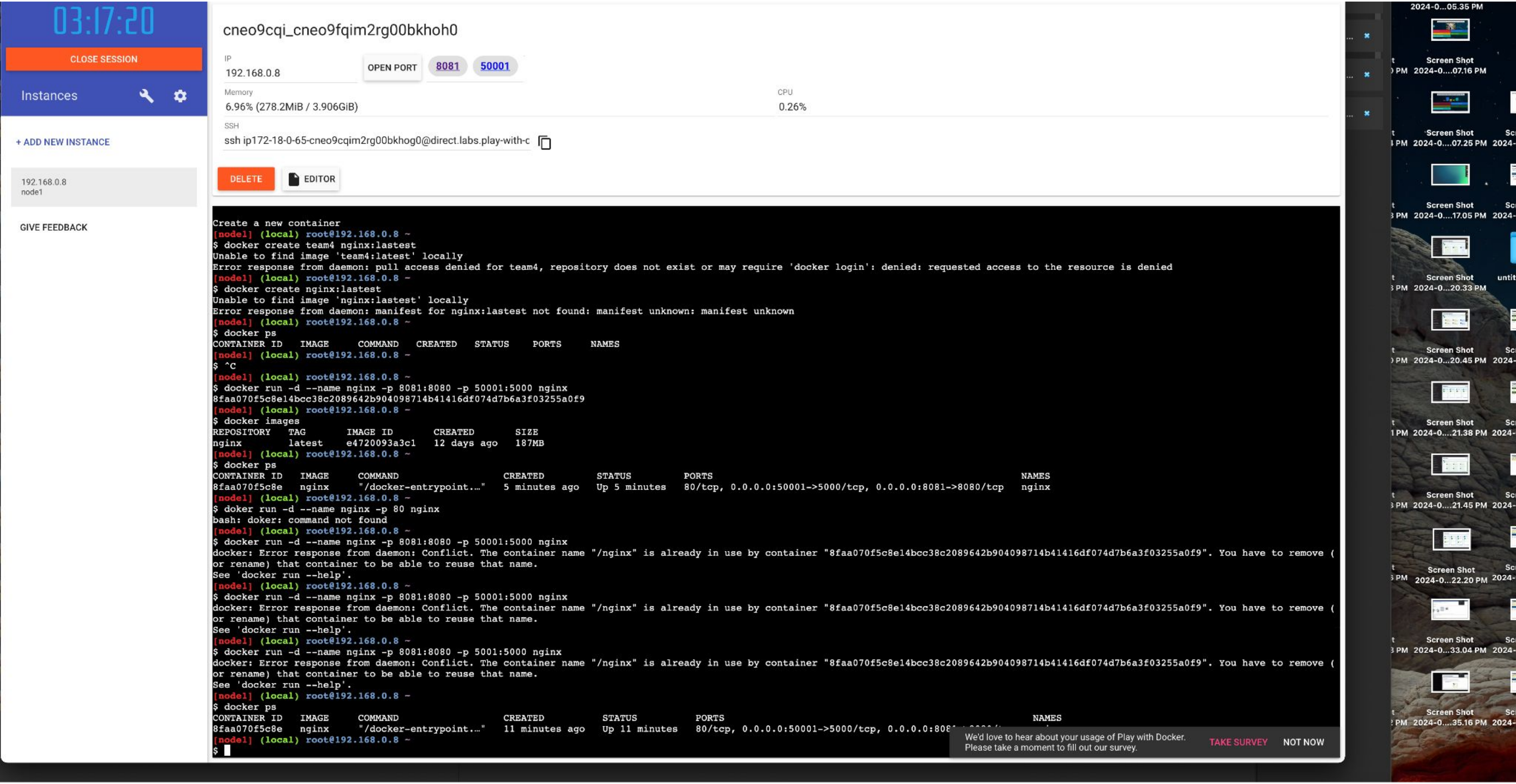
Authentification- The verification of user by password, user name and biometrics to gain access to an organization's resources, while Authorization is granting a user access to specific roles and group in the organization's resources

The responsibility of customer and cloud service provider. Provider protects certain aspect of the cloud service while customer has shared responsibility
.
The list of provider responsibilities are: Security and data center, data encryption at rest and in transit., availability of services and network and platform security.Customer responsibilities are : compliance of legal requirements, data encryption and protection, identity and access management and secure cloud resources

AWS security service: IAM manages access to organization's resources and AWS services Web application firewall: prevents web exploits that tamper availability of services
Key management services (KMS): Create encryption keys used to encrypt data, encryption and decryption of AWS services like S3 and EBS

The best practices in creating passwords in AWS:

1. Regularly change password
2. Use uppercase, lowercase, letters, numbers and special characters
3. Enable MFA (Multi Factor Authentification
4. Use long passwords


03:17:20
CLOSE SESSION
Instances
+ ADD NEW INSTANCE
192.168.0.8
node1
GIVE FEEDBACK
OPEN PORT
8081
50001
DELETE
EDITOR
TAKE SURVEY
NOT NOW
Screen Shot 2024-0....07.16 PM
Screen Shot 2024-0....07.25 PM
Screen Shot 2024-0....17.05 PM
Screen Shot 2024-0....20.33 PM
Screen Shot 2024-0....20.45 PM
Screen Shot 2024-0....21.38 PM
Screen Shot 2024-0....21.45 PM
Screen Shot 2024-0....22.20 PM
Screen Shot 2024-0....33.04 PM
Screen Shot 2024-0....35.16 PM

cneo9cqi_cneo9fqim2rg00bkhoh0

IP
192.168.0.8

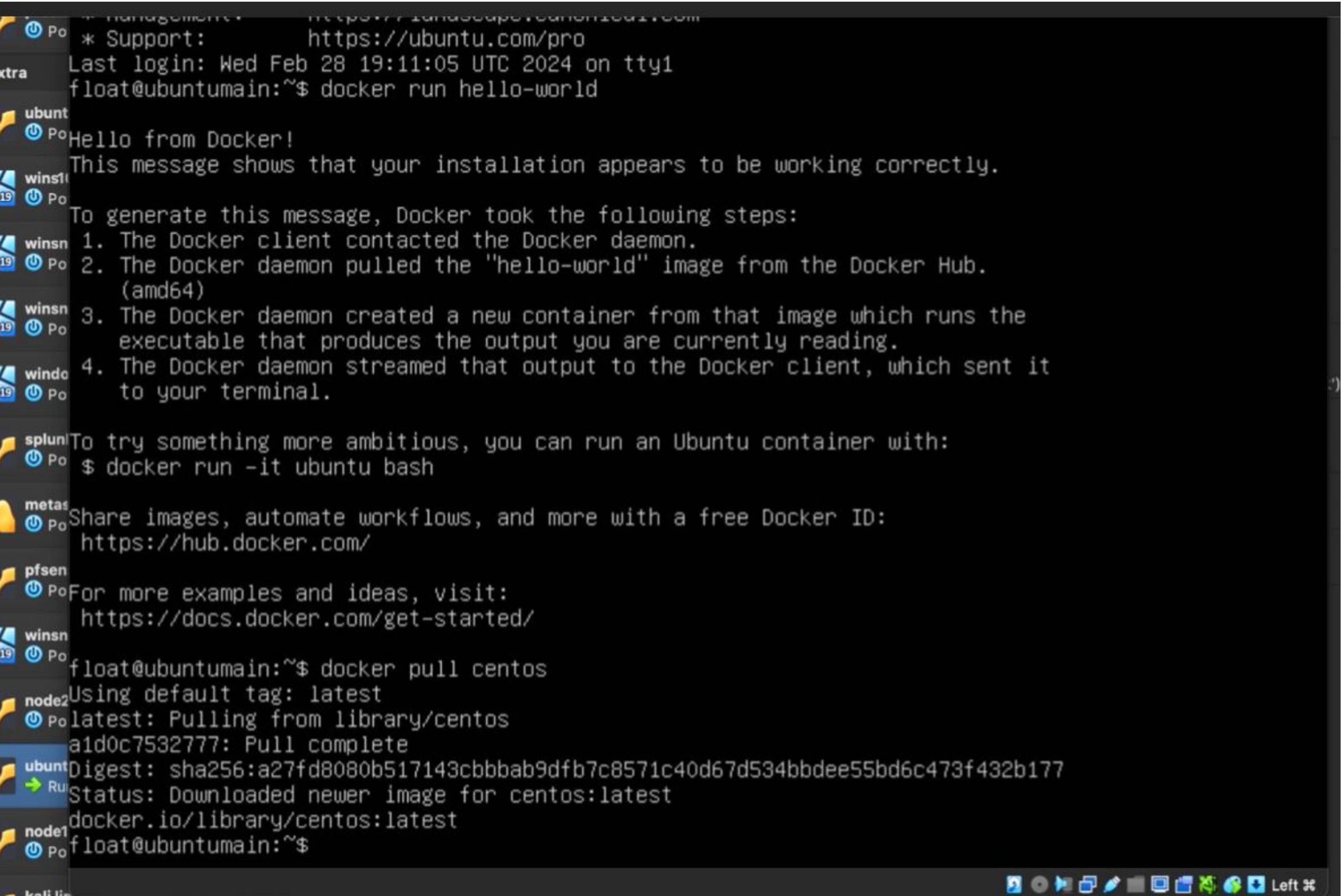
Memory
6.96% (278.2MiB / 3.906GiB)

CPU
0.26%

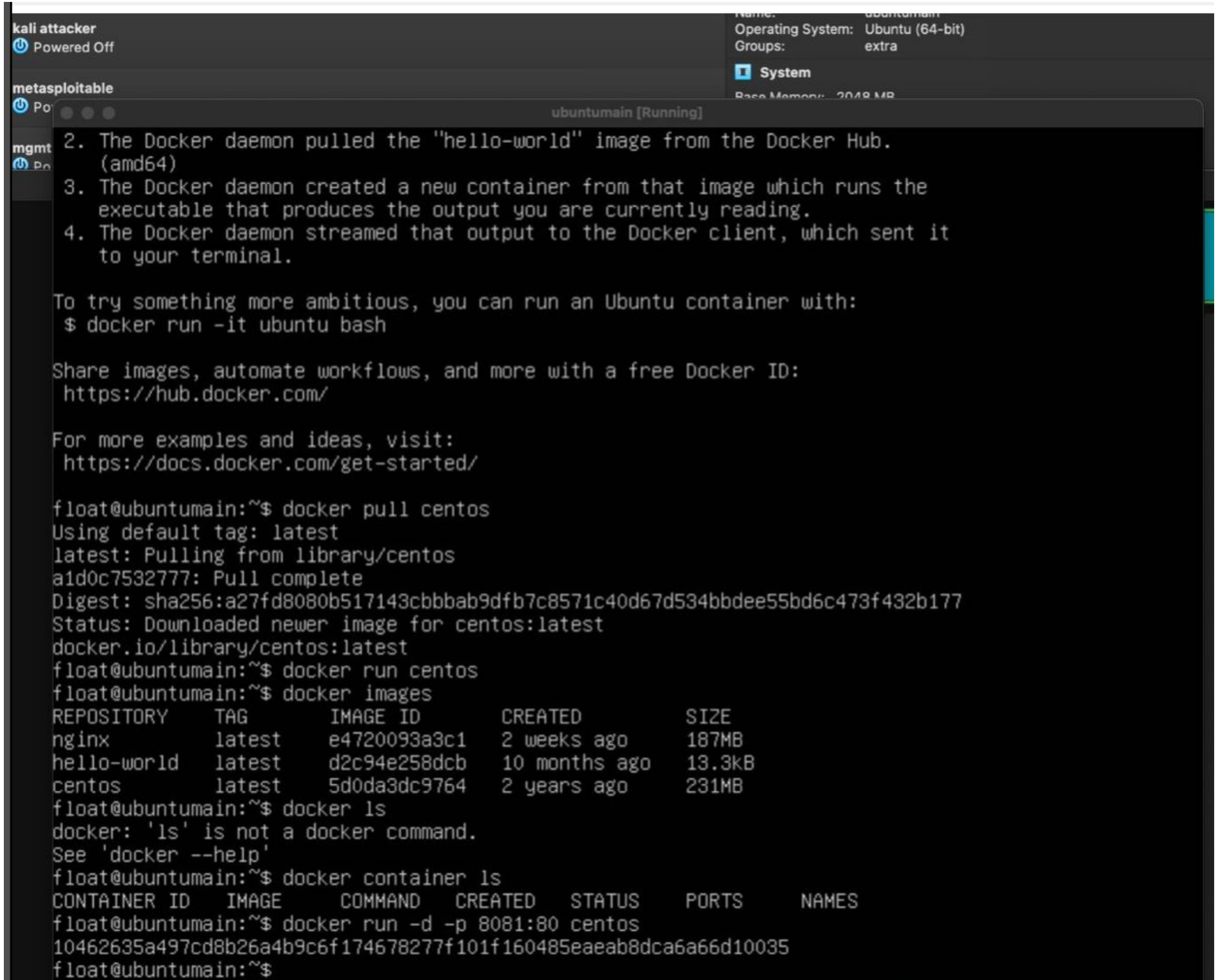
SSH
ssh ip172-18-0-65-cneo9cqim2rg00bkhog0@direct.labs.play-with-c

```
Create a new container
[node1] (local) root@192.168.0.8 ~
$ docker create team4 nginx:lastest
Unable to find image 'team4:latest' locally
Error response from daemon: pull access denied for team4, repository does not exist or may require 'docker login': denied: requested access to the resource is denied
[node1] (local) root@192.168.0.8 ~
$ docker create nginx:lastest
Unable to find image 'nginx:lastest' locally
Error response from daemon: manifest for nginx:lastest not found: manifest unknown: manifest unknown
[node1] (local) root@192.168.0.8 ~
$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
[node1] (local) root@192.168.0.8 ~
$ ^C
[node1] (local) root@192.168.0.8 ~
$ docker run -d --name nginx -p 8081:8080 -p 50001:5000 nginx
8faa070f5c8e14bcc38c2089642b904098714b41416df074d7b6a3f03255a0f9
[node1] (local) root@192.168.0.8 ~
$ docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
nginx        latest    e4720093a3c1   12 days ago   187MB
[node1] (local) root@192.168.0.8 ~
$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                                            NAMES
8faa070f5c8e   nginx     "/docker-entrypoint.…"   5 minutes ago   Up 5 minutes   80/tcp, 0.0.0.0:50001->5000/tcp, 0.0.0.0:8081->8080/tcp   nginx
[node1] (local) root@192.168.0.8 ~
$ doker run -d --name nginx -p 80 nginx
bash: doker: command not found
[node1] (local) root@192.168.0.8 ~
$ docker run -d --name nginx -p 8081:8080 -p 50001:5000 nginx
docker: Error response from daemon: Conflict. The container name "/nginx" is already in use by container "8faa070f5c8e14bcc38c2089642b904098714b41416df074d7b6a3f03255a0f9". You have to remove (
or rename) that container to be able to reuse that name.
See 'docker run --help'.
[node1] (local) root@192.168.0.8 ~
$ docker run -d --name nginx -p 8081:8080 -p 50001:5000 nginx
docker: Error response from daemon: Conflict. The container name "/nginx" is already in use by container "8faa070f5c8e14bcc38c2089642b904098714b41416df074d7b6a3f03255a0f9". You have to remove (
or rename) that container to be able to reuse that name.
See 'docker run --help'.
[node1] (local) root@192.168.0.8 ~
$ docker run -d --name nginx -p 8081:8080 -p 5001:5000 nginx
docker: Error response from daemon: Conflict. The container name "/nginx" is already in use by container "8faa070f5c8e14bcc38c2089642b904098714b41416df074d7b6a3f03255a0f9". You have to remove (
or rename) that container to be able to reuse that name.
See 'docker run --help'.
[node1] (local) root@192.168.0.8 ~
$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                                            NAMES
8faa070f5c8e   nginx     "/docker-entrypoint.…"   11 minutes ago   Up 11 minutes   80/tcp, 0.0.0.0:50001->5000/tcp, 0.0.0.0:808
[node1] (local) root@192.168.0.8 ~
$ 
```

We'd love to hear about your usage of Play with Docker.
Please take a moment to fill out our survey.

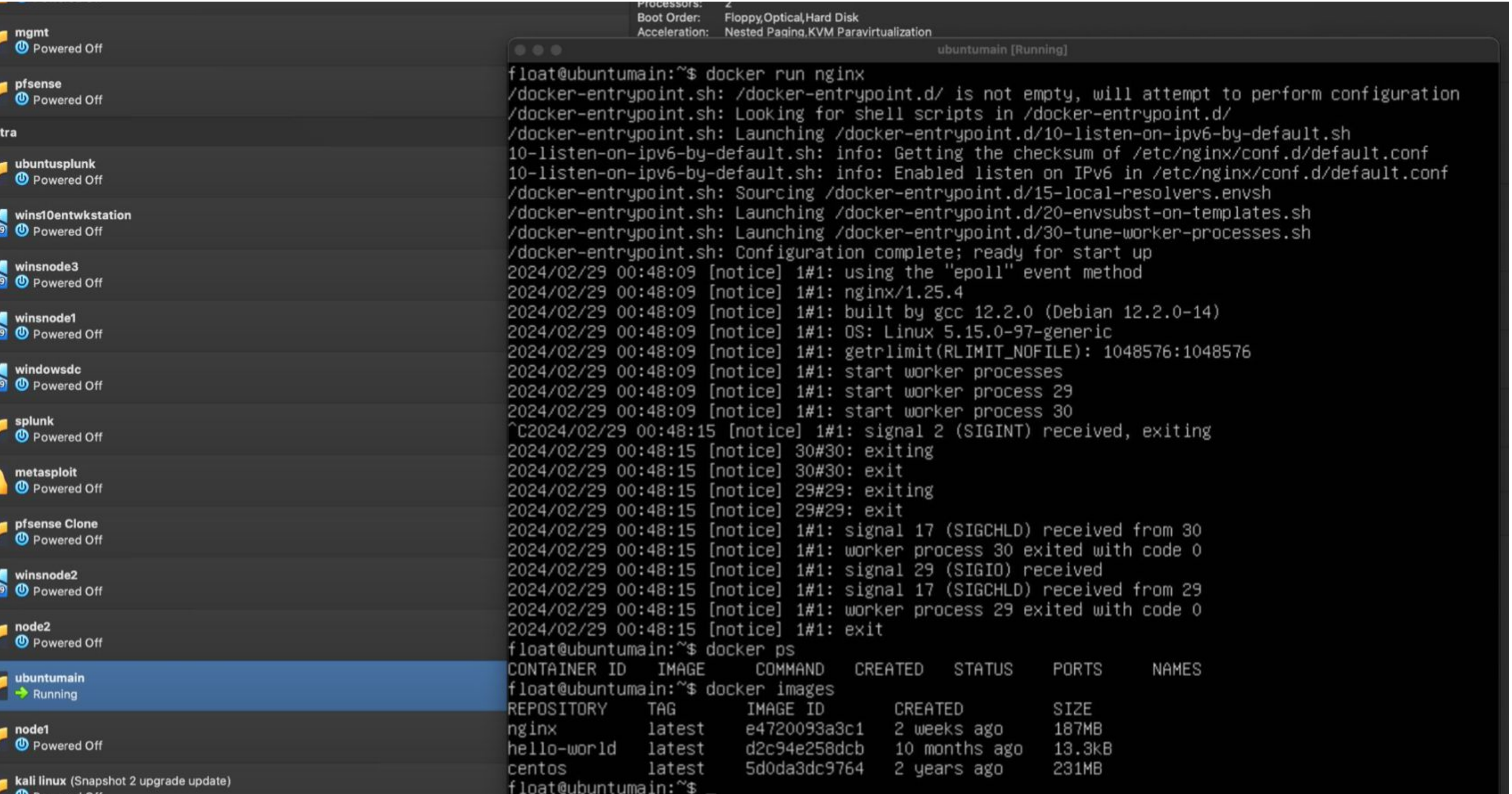


```
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro
Last login: Wed Feb 28 19:11:05 UTC 2024 on tty1
float@ubuntumain:~$ docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

float@ubuntumain:~$ docker pull centos
Using default tag: latest
latest: Pulling from library/centos
a1d0c7532777: Pull complete
Digest: sha256:a27fd8080b517143cbbbab9dfb7c8571c40d67d534bbdee55bd6c473f432b177
Status: Downloaded newer image for centos:latest
docker.io/library/centos:latest
float@ubuntumain:~$
```



Operating System: Ubuntu (64-bit)
Groups: extra
Base Memory: 2048 MB

```
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

float@ubuntumain:~$ docker pull centos
Using default tag: latest
latest: Pulling from library/centos
a1d0c7532777: Pull complete
Digest: sha256:a27fd8080b517143cbbbab9dfb7c8571c40d67d534bbdee55bd6c473f432b177
Status: Downloaded newer image for centos:latest
docker.io/library/centos:latest
float@ubuntumain:~$ docker run centos
float@ubuntumain:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
nginx         latest    e4720093a3c1   2 weeks ago     187MB
hello-world   latest    d2c94e258dcb   10 months ago   13.3kB
centos        latest    5d0da3dc9764   2 years ago     231MB
float@ubuntumain:~$ docker ls
docker: 'ls' is not a docker command.
See 'docker --help'
float@ubuntumain:~$ docker container ls
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
float@ubuntumain:~$ docker run -d -p 8081:80 centos
10462635a497cd8b26a4b9c6f174678277f101f160485eaeab8dca6a66d10035
float@ubuntumain:~$
```



Processors: 2
Boot Order: Floppy,Optical,Hard Disk
Acceleration: Nested Paging,KVM Paravirtualization

```
float@ubuntumain:~$ docker run nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/02/29 00:48:09 [notice] 1#1: using the "epoll" event method
2024/02/29 00:48:09 [notice] 1#1: nginx/1.25.4
2024/02/29 00:48:09 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2024/02/29 00:48:09 [notice] 1#1: OS: Linux 5.15.0-97-generic
2024/02/29 00:48:09 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2024/02/29 00:48:09 [notice] 1#1: start worker processes
2024/02/29 00:48:09 [notice] 1#1: start worker process 29
2024/02/29 00:48:09 [notice] 1#1: start worker process 30
^C2024/02/29 00:48:15 [notice] 1#1: signal 2 (SIGINT) received, exiting
2024/02/29 00:48:15 [notice] 30#30: exiting
2024/02/29 00:48:15 [notice] 30#30: exit
2024/02/29 00:48:15 [notice] 29#29: exiting
2024/02/29 00:48:15 [notice] 29#29: exit
2024/02/29 00:48:15 [notice] 1#1: signal 17 (SIGCHLD) received from 30
2024/02/29 00:48:15 [notice] 1#1: worker process 30 exited with code 0
2024/02/29 00:48:15 [notice] 1#1: signal 29 (SIGIO) received
2024/02/29 00:48:15 [notice] 1#1: signal 17 (SIGCHLD) received from 29
2024/02/29 00:48:15 [notice] 1#1: worker process 29 exited with code 0
2024/02/29 00:48:15 [notice] 1#1: exit
float@ubuntumain:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
float@ubuntumain:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
nginx         latest    e4720093a3c1   2 weeks ago     187MB
hello-world   latest    d2c94e258dcb   10 months ago   13.3kB
centos        latest    5d0da3dc9764   2 years ago     231MB
float@ubuntumain:~$ _
```