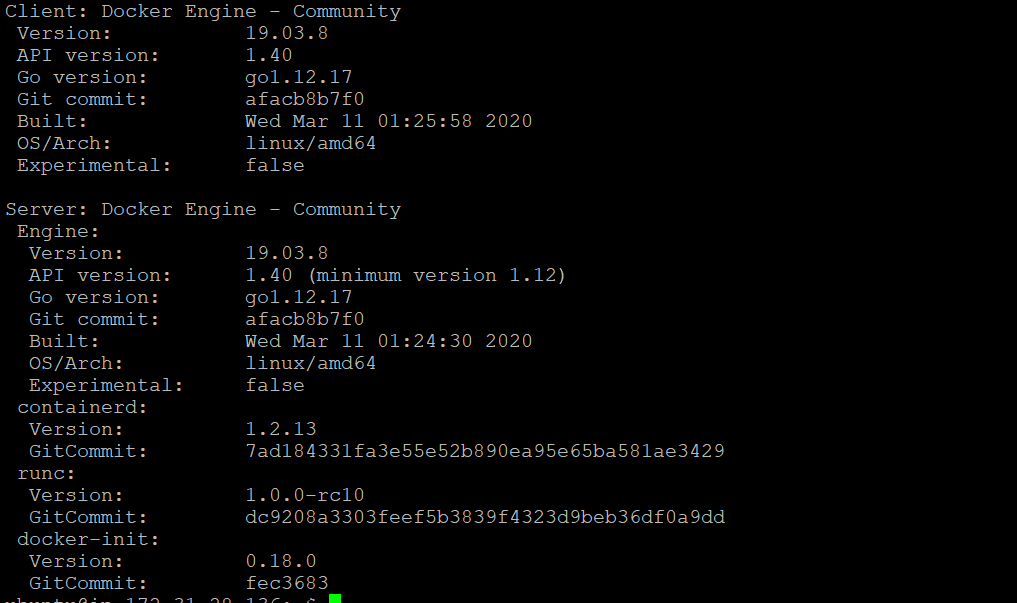
1. Installed docker on Ubuntu using wget command

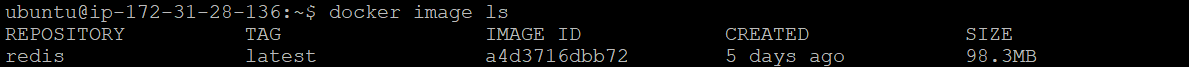
wget –qO- <https://get.docker.com/> | sh



1. Image download image

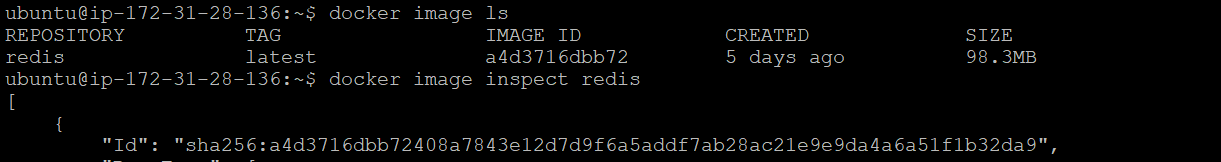
Command :: docker image pull redis

1. Image



1. Find full Image ID

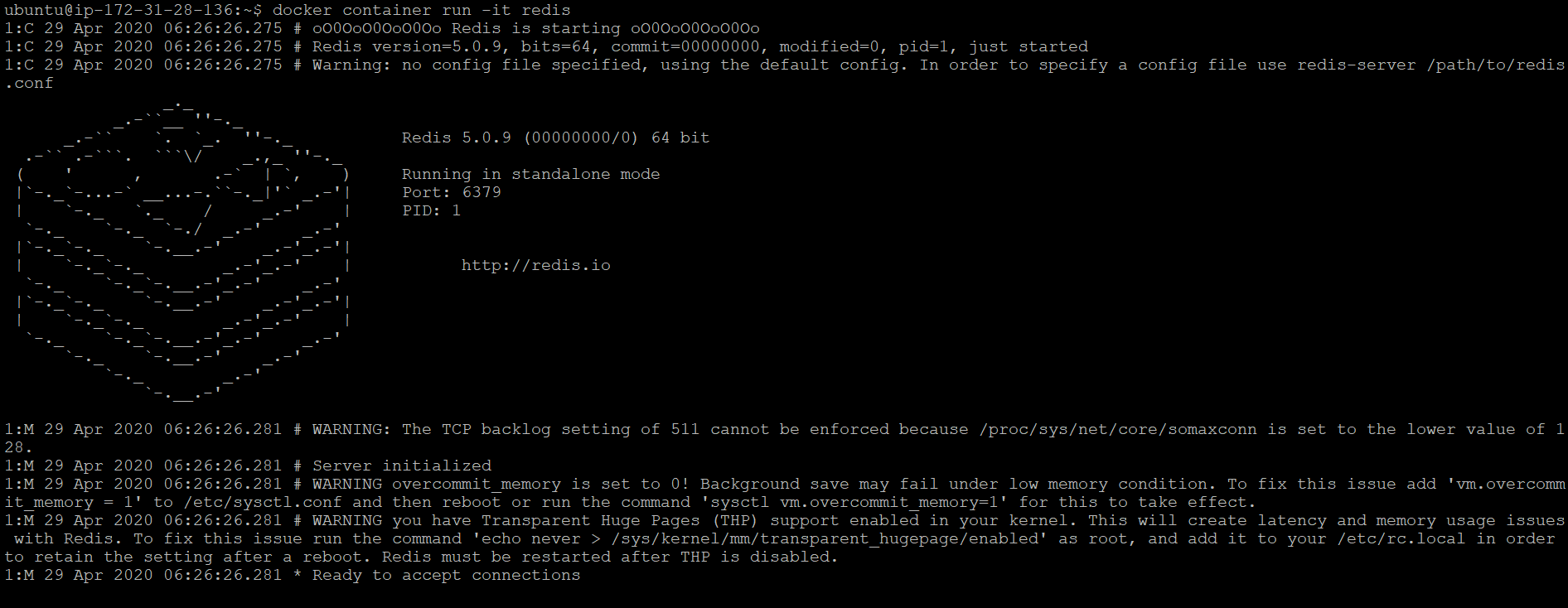
Command :: docker image inspect redis<image name>



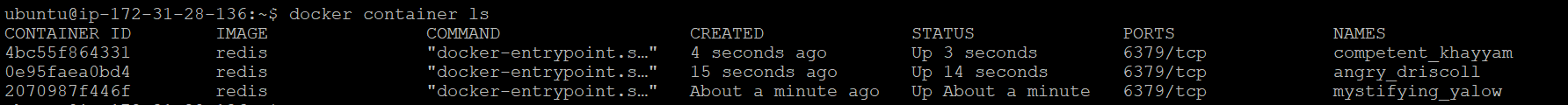
1. Create a container of your image

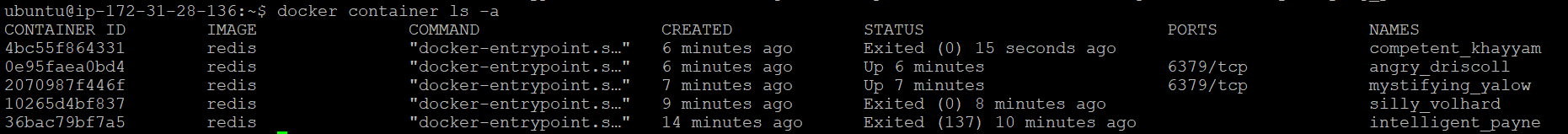
Command :: docker container run –it redis

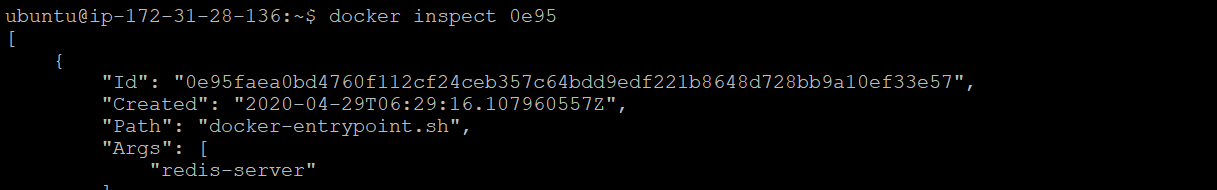
docker container run –it –d redis /bin/bash

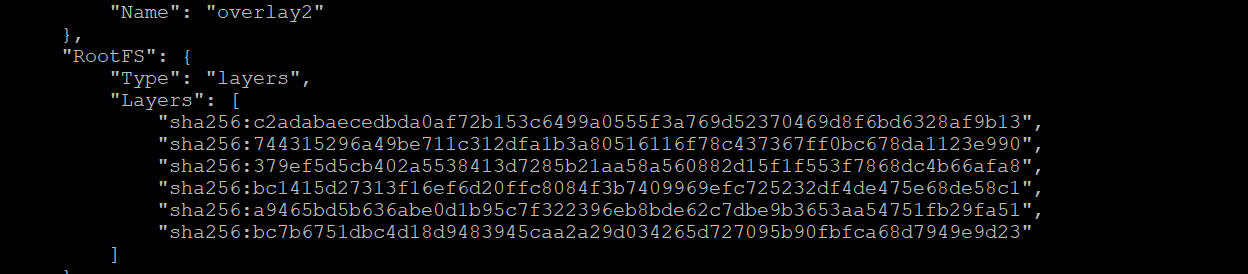


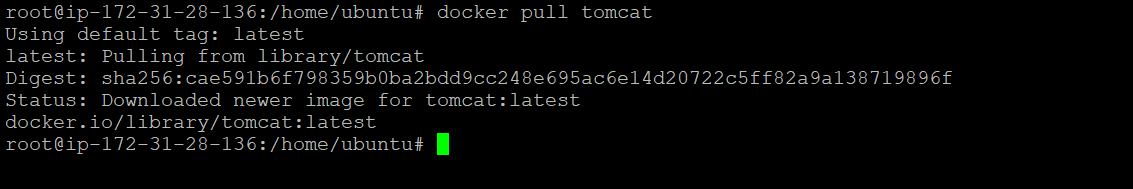


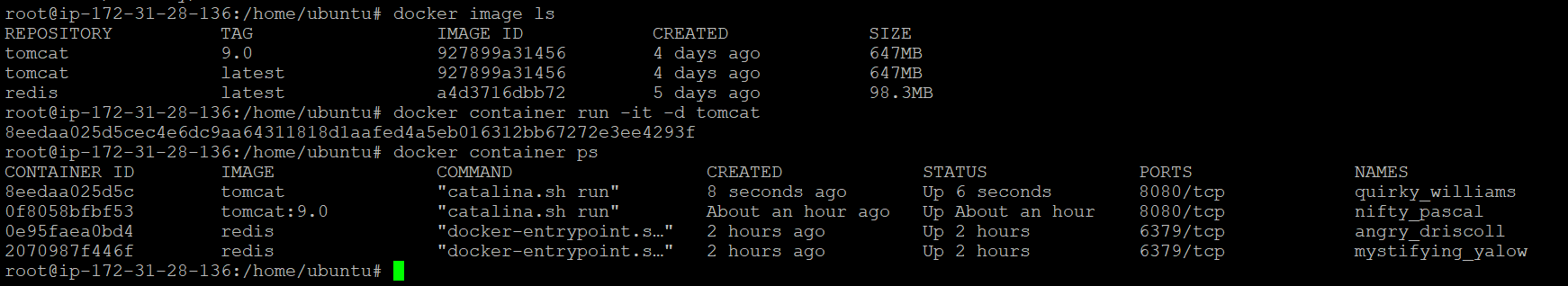












Use docker inspect container ID to check on which ip address tomcat is running on



Port details ::

},

"NetworkSettings": {

"Bridge": "",

"SandboxID": "559a3be0a0ccc5d711be20b971cc3bab1de282b47454467bb564fa2aa0b0d5a2",

"HairpinMode": false,

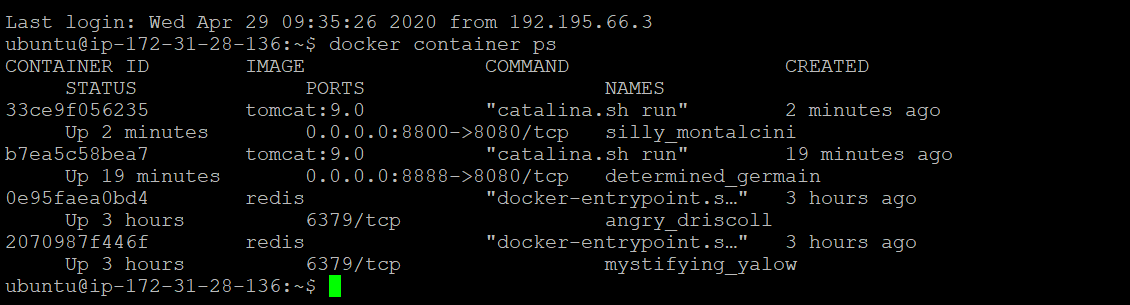
"LinkLocalIPv6Address": "",

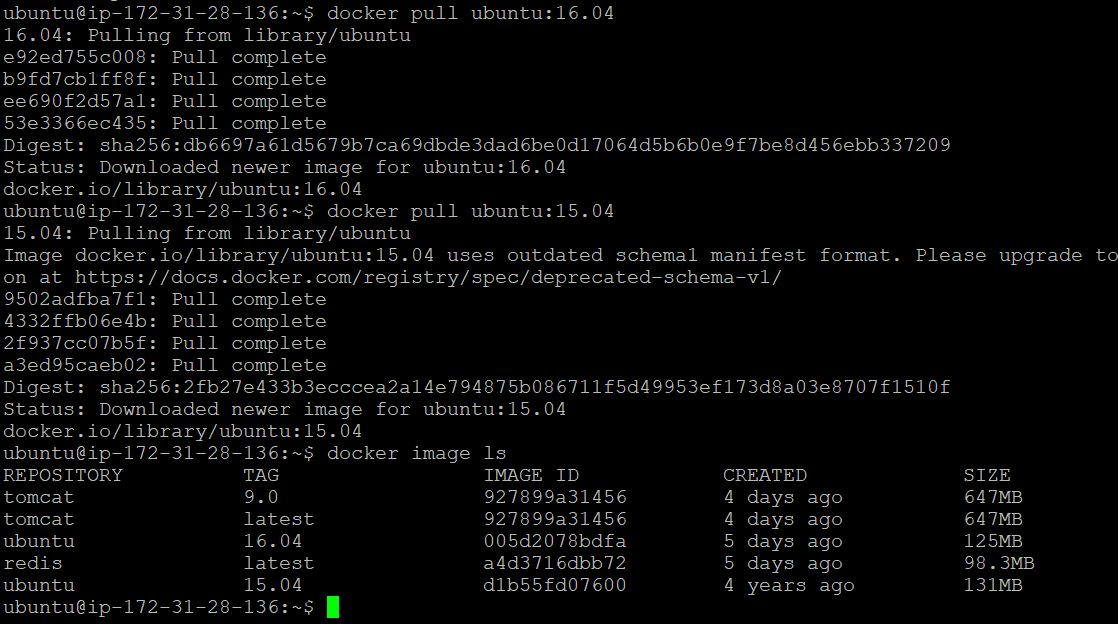
"LinkLocalIPv6PrefixLen": 0,

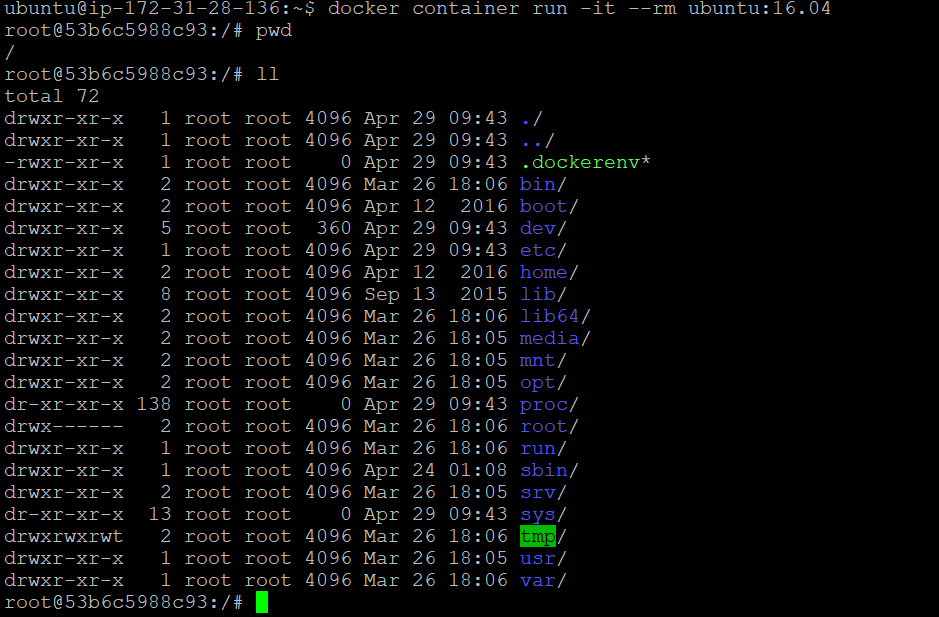
"Ports": {

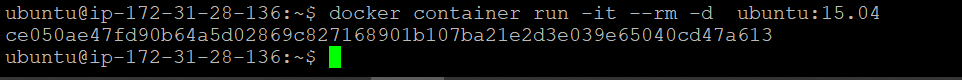
"8080/tcp": null

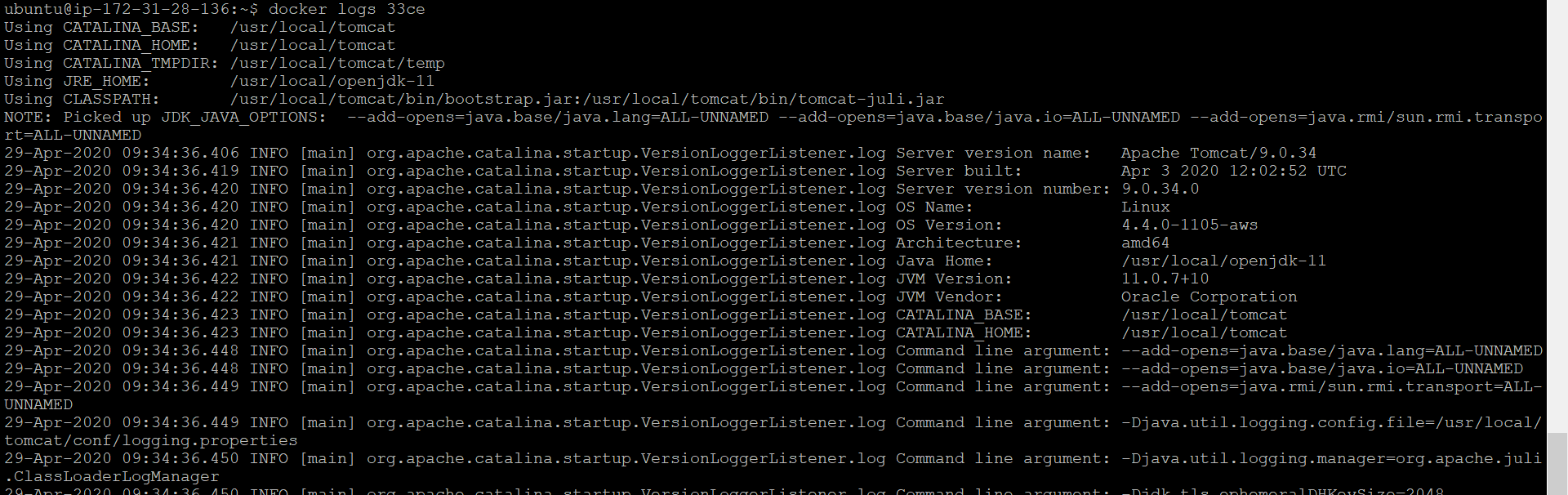


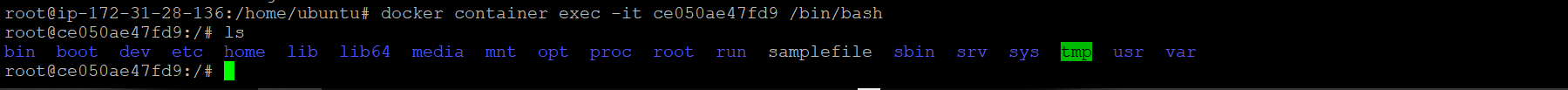


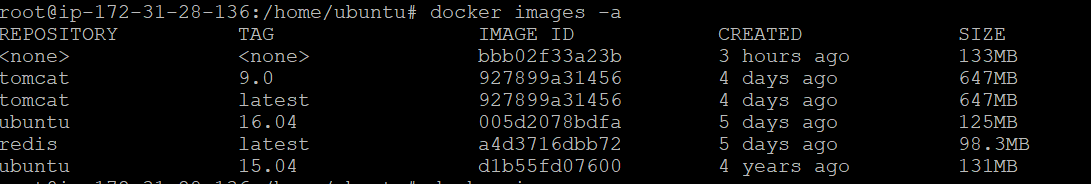


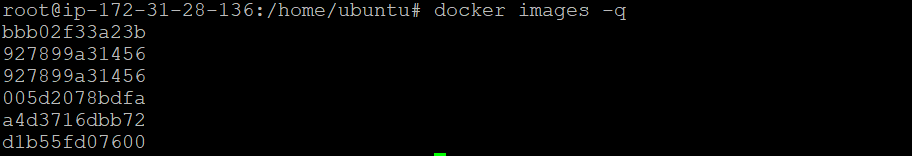


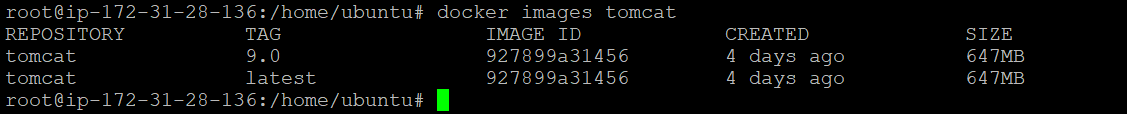




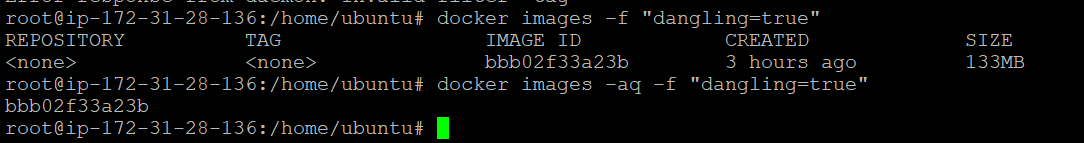


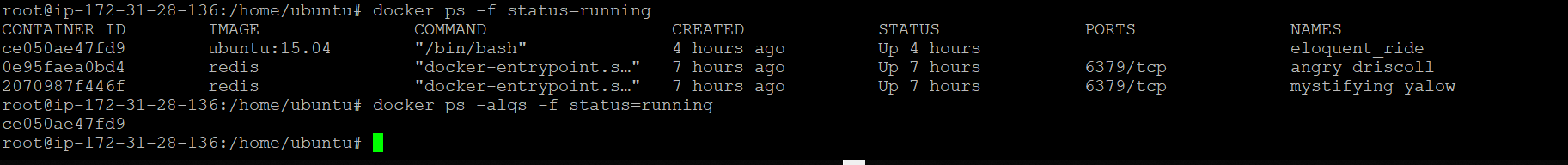


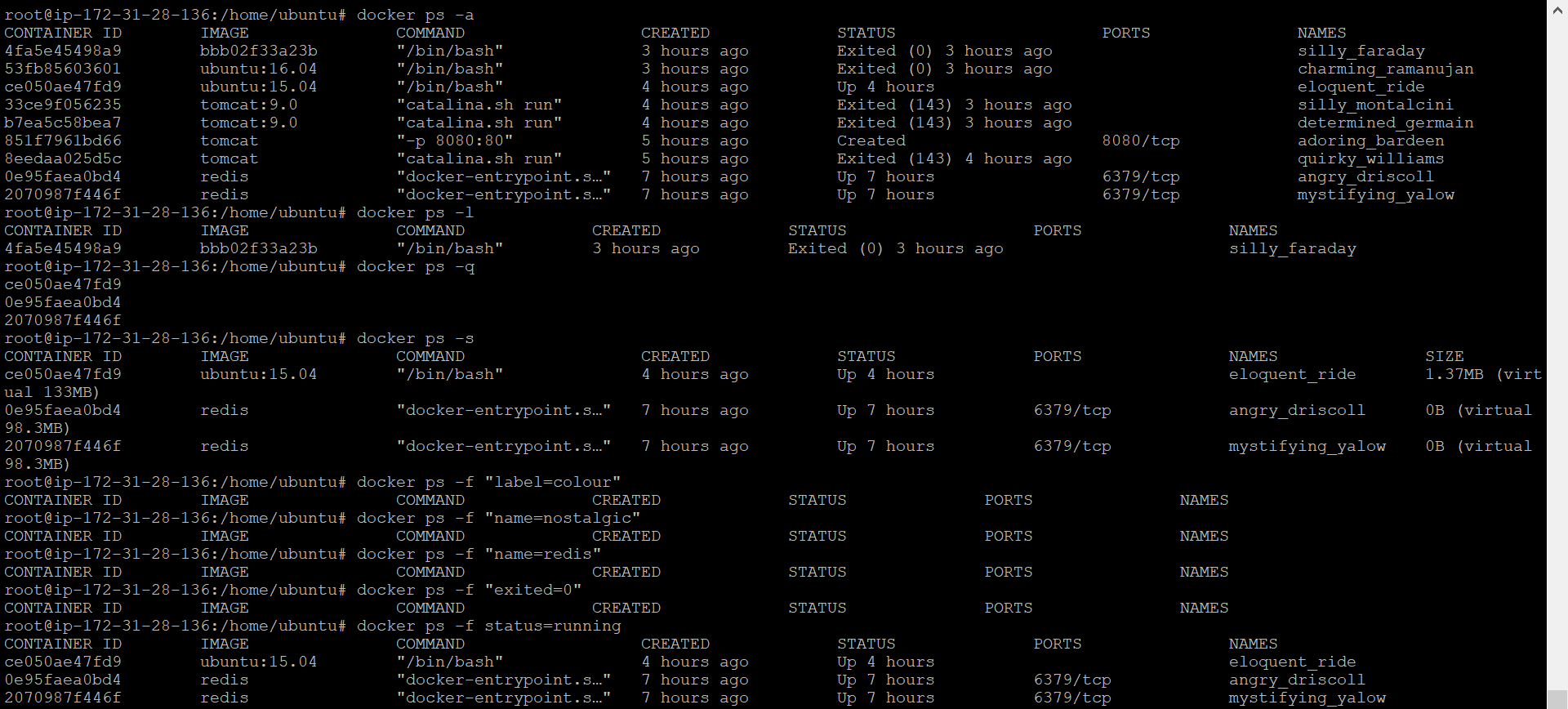




* dangling (boolean - true or false)
* label (label=<key> or label=<key>=<value>)
* before (<image-name>[:<tag>], <image id> or <image@digest>) - filter images created before given id or references
* since (<image-name>[:<tag>], <image id> or <image@digest>) - filter images created since given id or references
* reference (pattern of an image reference) - filter images whose reference matches the specified pattern







Management Commands:

builder Manage builds ---- it helps to build and image from docker file

config Manage Docker configs -- help to create config from a file

container Manage containers -- it help to play with containers.

conext Manage contexts -- It help to create, import, export context This shows a single context called “default”. It’s configured to talk to a Swarm cluster through the local /var/run/docker.sock Unix socket. It has no Kubernetes endpoint configured.

engine Manage the docker engine -- show details about current docker enginer we can update and activate EE

image Manage images -- we can work on images in depth creating deleting etc.

network Manage networks -- we can play with network between bridge and overlay we can create new networks

node Manage Swarm nodes -- managing bulk nodes and containers, act as a cluster

plugin Manage plugins

secret Manage Docker secrets

service Manage services

stack Manage Docker stacks

swarm Manage Swarm

system Manage Docker

trust Manage trust on Docker images

volume Manage volumes

Use each command mentioned below and prove its concepts as described in the --help desription. write what you have understood from the output of the command after its successful execution.

1. cp -- Copy files/folders between a container and the local filesystem

2. create -- to create anything new on docker server

3. export -- Export a container's filesystem as a tar archive

4. history – to fetch history about image

5. info -- information about anything widely

6. login -- login to docker registry

7. logout – logout from docker registry

8. rename -- use to rename a container

9. save -- Save one or more images to a tar archive

10. stats -- Display a live stream of container(s) resource usage statistics

11. top -- running process of container

To Kill all running containers ::

