**Working with container**

[**https://docs.docker.com/engine/reference/commandline**](https://docs.docker.com/engine/reference/commandline)

1. Create a container and run it

Since we use Ubuntu 14.04 on the server, so we will pull that version of Ubuntu from the official Docker hub.

* docker pull ubuntu:14.04

To see the list of current images on your host:

* docker images

Build a container based on the base image

* docker run -it ubuntu:14.04 /bin/bash

Docker will create a new container and take you to it. As you see in the above image, "e0de069a28e9" is the id of this container.

Press **CTRL + P** and then **CTRL + Q** immediatelyto return to the host and then run the command line as follows to see the list of ***current running*** containers:



Jump into a container by usgin attach key word

* docker attach <CONTAINER\_ID> ("e0de069a28e9", for example, but you can use the name of container "sad\_poitras" instead to achieve the same result)

1. **Start and stop**

For example, I will stop the “Postgres” container

* docker stop explosion-postgres

And start it immediately

* docker start explosion-postgres

1. **List of container**

* List of running containers

docker ps

* List all container with all status (using this command if you want to remove a container)

docker ps -a

1. **Remove a container**

Check to view all container

docker ps -a

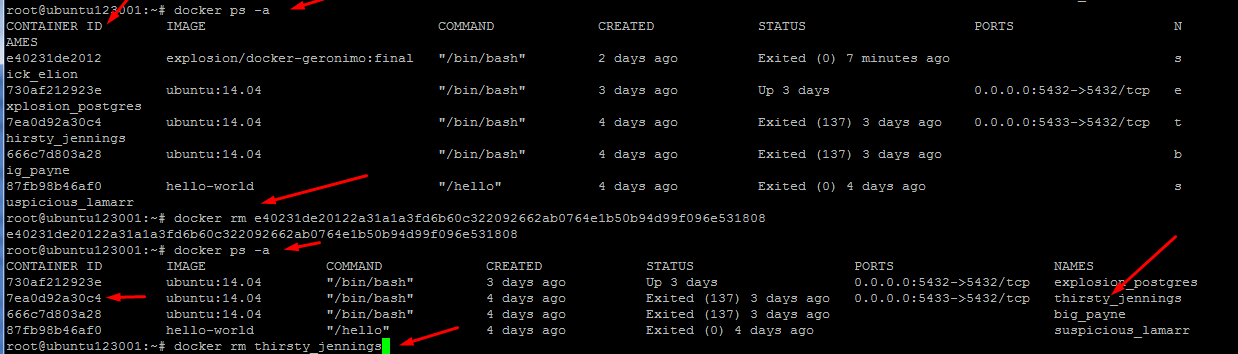
Remove by id

docker rm <id of container>

Remove by name

docker rm <name of container>

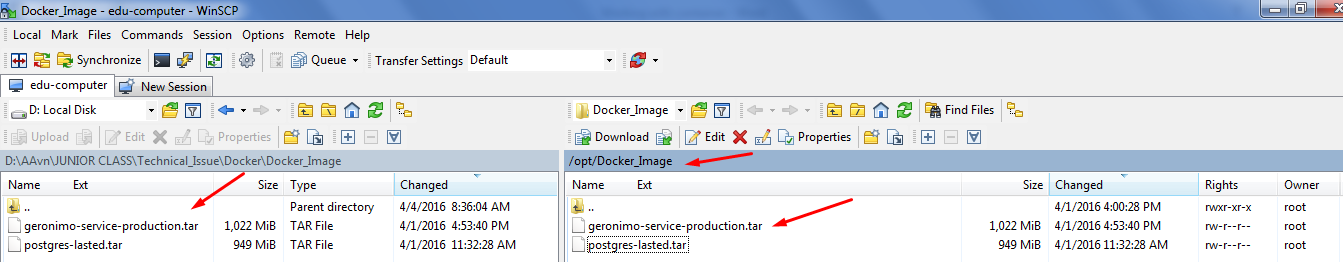
Example:



1. **Reuse one image from a file on a host (other computer)**

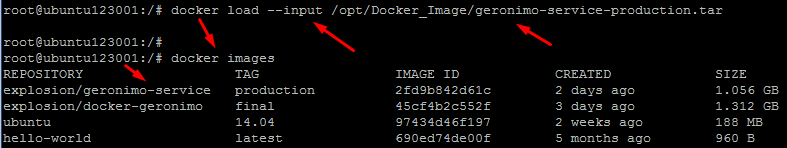
If you use command: **docker save -o {path} {image name}**

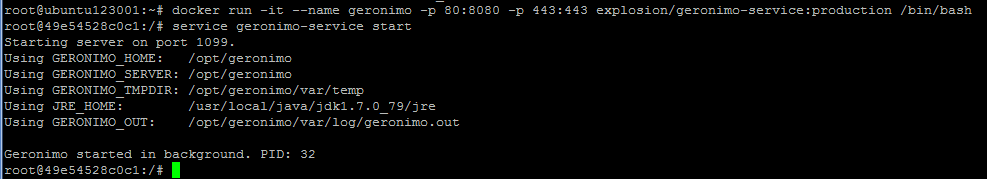
Copy stored image to the host server; I rather than using **Winscp** to copy file



Load the image from a tar file:

docker load -i <path to image tar file>





**Build a container using an image**

Docker run –it –name geronimo -p 80:8080 –p 443:443 explostion/Geronimo-service:production /bin/bash

docker run –it –name geronimo -p 80:8080 –p 443:443 explostion/Geronimo-service:production /bin/bash

Container name

Host port: internal container port

Image name:tag

1. **Delete an image**

docker rmi <image\_id>:<tag>

