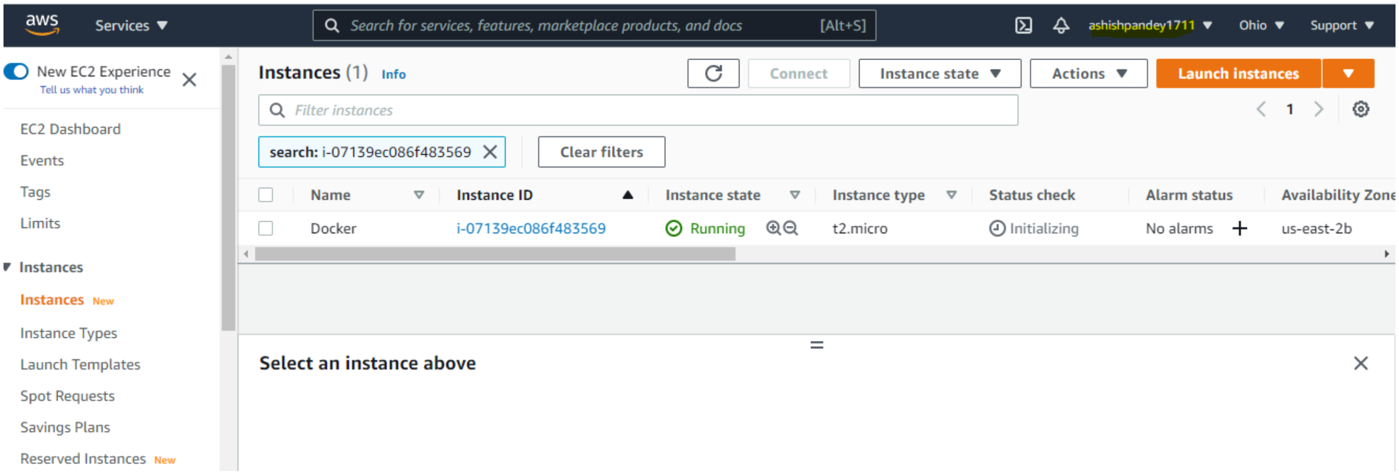
Pandeyashish1234@gmail.com , mob no : 9971490122 , Day2 :Assignment : 15th Aug 2021

1) Create an EC2 AWS instance



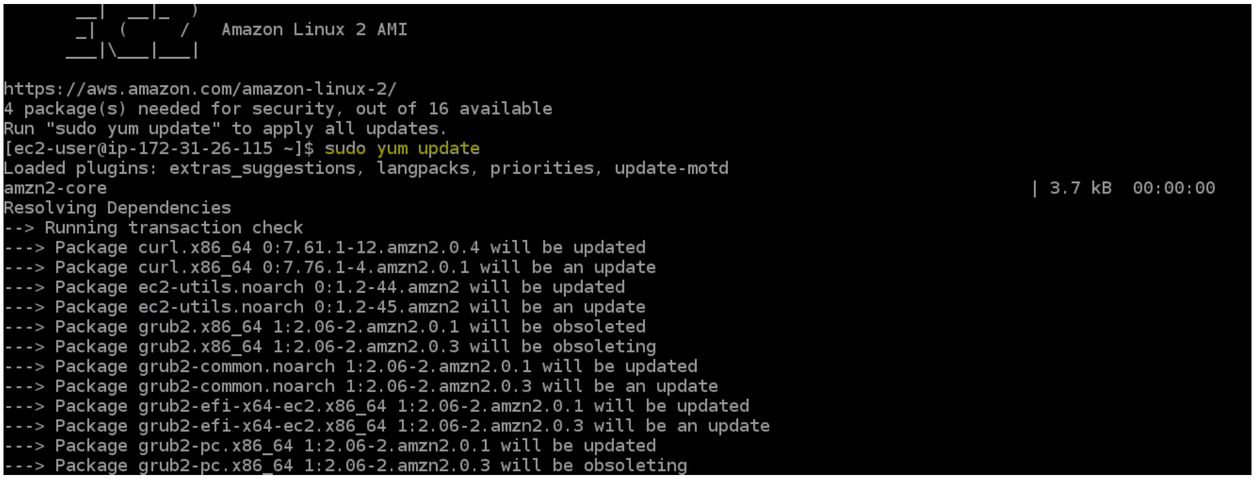
Instances (1) Info

search: i-07139ec086f483569

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
□	Docker	i-07139ec086f483569	✓ Running	t2.micro	Initializing	No alarms	us-east-2b

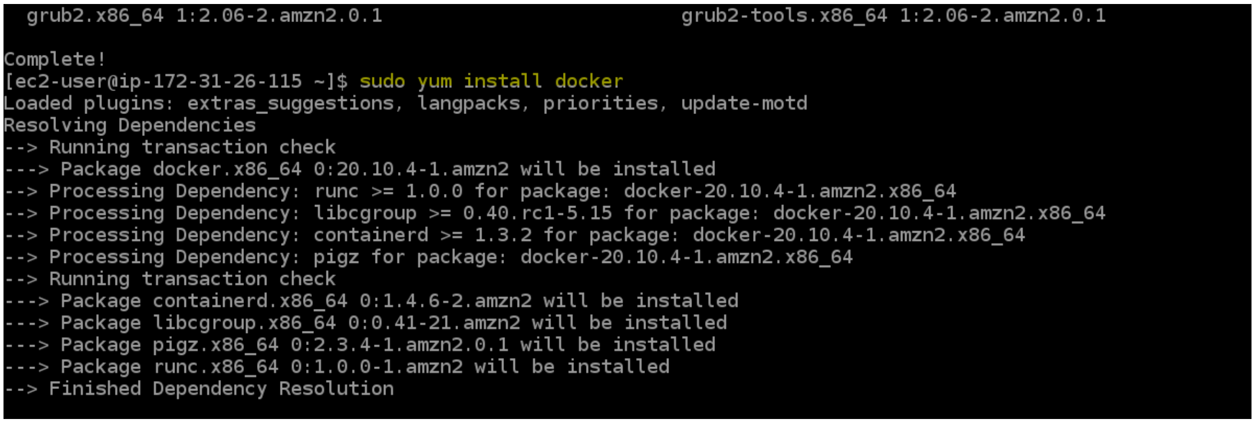
Select an instance above

2) Sudo Yum update



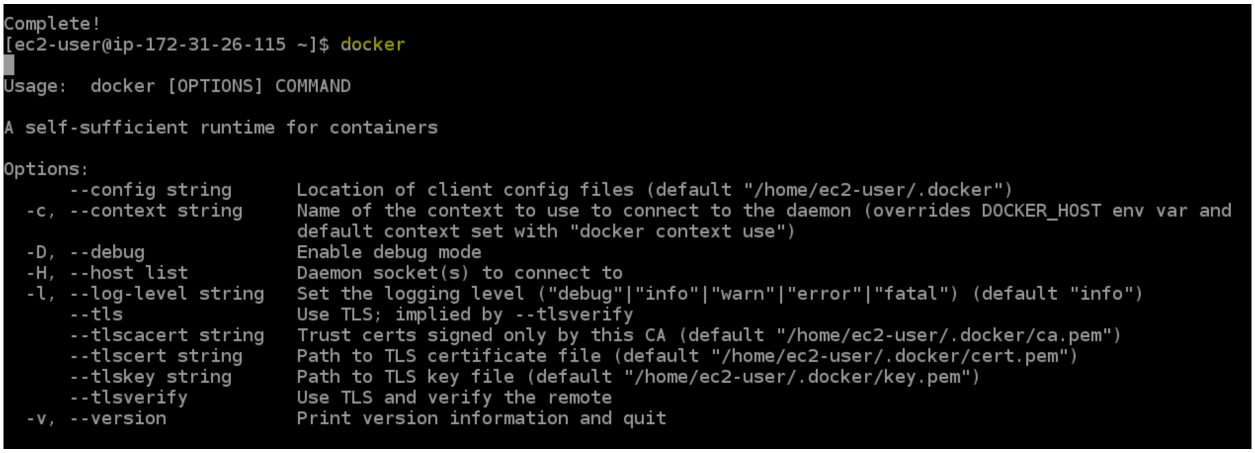
```
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
4 package(s) needed for security, out of 16 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-26-115 ~]$ sudo yum update
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                                   | 3.7 kB  00:00:00
Resolving Dependencies
--> Running transaction check
---> Package curl.x86_64 0:7.61.1-12.amzn2.0.4 will be updated
---> Package curl.x86_64 0:7.76.1-4.amzn2.0.1 will be an update
---> Package ec2-utils.noarch 0:1.2-44.amzn2 will be updated
---> Package ec2-utils.noarch 0:1.2-45.amzn2 will be an update
---> Package grub2.x86_64 1:2.06-2.amzn2.0.1 will be obsoleted
---> Package grub2.x86_64 1:2.06-2.amzn2.0.3 will be obsoleting
---> Package grub2-common.noarch 1:2.06-2.amzn2.0.1 will be updated
---> Package grub2-common.noarch 1:2.06-2.amzn2.0.3 will be an update
---> Package grub2-efi-x64-ec2.x86_64 1:2.06-2.amzn2.0.1 will be updated
---> Package grub2-efi-x64-ec2.x86_64 1:2.06-2.amzn2.0.3 will be an update
---> Package grub2-pc.x86_64 1:2.06-2.amzn2.0.1 will be updated
---> Package grub2-pc.x86_64 1:2.06-2.amzn2.0.3 will be obsoleting
```

3)sudo yum install docker



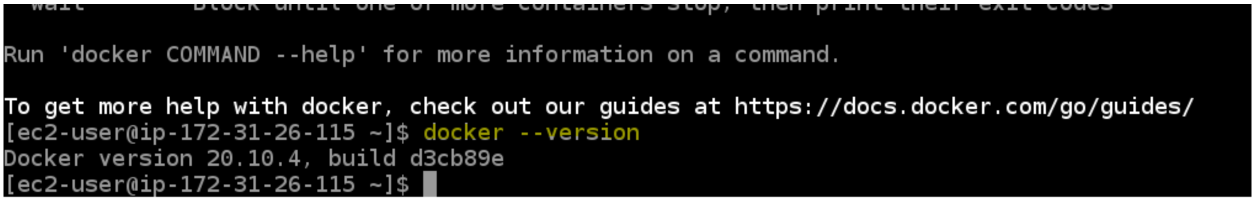
```
  grub2.x86_64 1:2.06-2.amzn2.0.1                          grub2-tools.x86_64 1:2.06-2.amzn2.0.1

Complete!
[ec2-user@ip-172-31-26-115 ~]$ sudo yum install docker
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package docker.x86_64 0:20.10.4-1.amzn2 will be installed
--> Processing Dependency: runc >= 1.0.0 for package: docker-20.10.4-1.amzn2.x86_64
--> Processing Dependency: libcgroup >= 0.40.rc1-5.15 for package: docker-20.10.4-1.amzn2.x86_64
--> Processing Dependency: containerd >= 1.3.2 for package: docker-20.10.4-1.amzn2.x86_64
--> Processing Dependency: pigz for package: docker-20.10.4-1.amzn2.x86_64
--> Running transaction check
---> Package containerd.x86_64 0:1.4.6-2.amzn2 will be installed
---> Package libcgroup.x86_64 0:0.41-21.amzn2 will be installed
---> Package pigz.x86_64 0:2.3.4-1.amzn2.0.1 will be installed
---> Package runc.x86_64 0:1.0.0-1.amzn2 will be installed
--> Finished Dependency Resolution
```

4)Docker



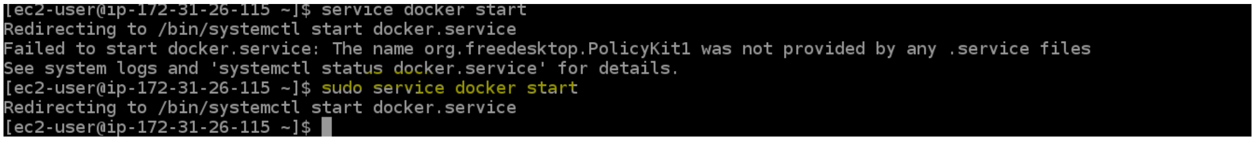
```
Complete!
[ec2-user@ip-172-31-26-115 ~]$ docker

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:
      --config string      Location of client config files (default "/home/ec2-user/.docker")
  -c, --context string     Name of the context to use to connect to the daemon (overrides DOCKER_HOST env var and
                           default context set with "docker context use")
  -D, --debug              Enable debug mode
  -H, --host list          Daemon socket(s) to connect to
  -l, --log-level string   Set the logging level ("debug"|"info"|"warn"|"error"|"fatal") (default "info")
      --tls                Use TLS; implied by --tlsverify
      --tlscacert string   Trust certs signed only by this CA (default "/home/ec2-user/.docker/ca.pem")
      --tlscert string     Path to TLS certificate file (default "/home/ec2-user/.docker/cert.pem")
      --tlskey string      Path to TLS key file (default "/home/ec2-user/.docker/key.pem")
      --tlsverify          Use TLS and verify the remote
  -v, --version            Print version information and quit
```

5) docker –version

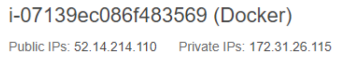


```
  wait        Block until one or more containers stop, then print their exit codes

Run 'docker COMMAND --help' for more information on a command.

To get more help with docker, check out our guides at https://docs.docker.com/go/guides/
[ec2-user@ip-172-31-26-115 ~]$ docker --version
Docker version 20.10.4, build d3cb89e
[ec2-user@ip-172-31-26-115 ~]$
```

6)service docker start



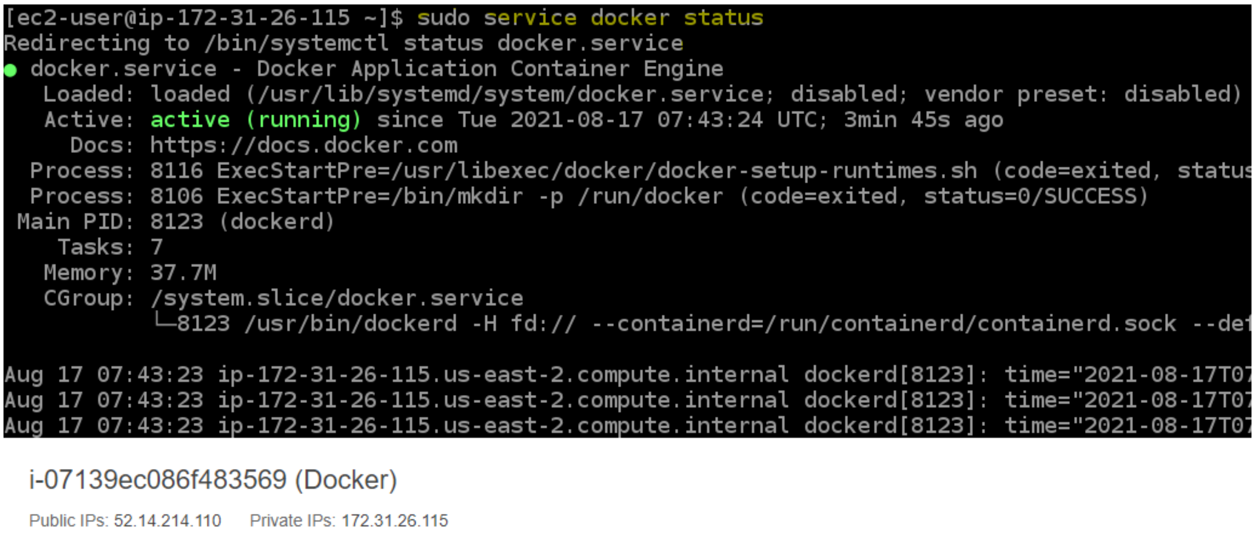
```
[ec2-user@ip-172-31-26-115 ~]$ service docker start
Redirecting to /bin/systemctl start docker.service
Failed to start docker.service: The name org.freedesktop.PolicyKit1 was not provided by any .service files
See system logs and 'systemctl status docker.service' for details.
[ec2-user@ip-172-31-26-115 ~]$ sudo service docker start
Redirecting to /bin/systemctl start docker.service
[ec2-user@ip-172-31-26-115 ~]$
```



i-07139ec086f483569 (Docker)

Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

7) service docker status

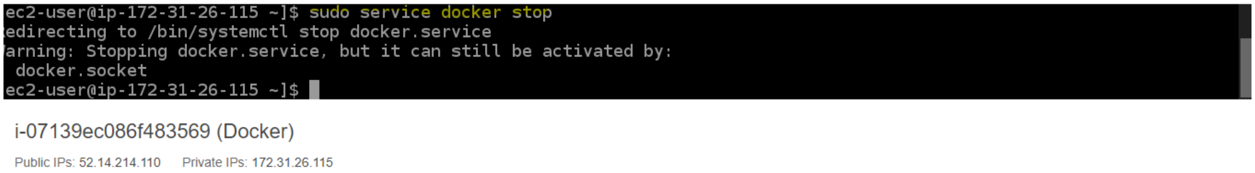


```
[ec2-user@ip-172-31-26-115 ~]$ sudo service docker status
Redirecting to /bin/systemctl status docker.service
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor preset: disabled)
   Active: active (running) since Tue 2021-08-17 07:43:24 UTC; 3min 45s ago
     Docs: https://docs.docker.com
  Process: 8116 ExecStartPre=/usr/libexec/docker/docker-setup-runtimes.sh (code=exited, status
  Process: 8106 ExecStartPre=/bin/mkdir -p /run/docker (code=exited, status=0/SUCCESS)
 Main PID: 8123 (dockerd)
    Tasks: 7
   Memory: 37.7M
   CGroup: /system.slice/docker.service
           └─8123 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock --def

Aug 17 07:43:23 ip-172-31-26-115.us-east-2.compute.internal dockerd[8123]: time="2021-08-17T07
Aug 17 07:43:23 ip-172-31-26-115.us-east-2.compute.internal dockerd[8123]: time="2021-08-17T07
Aug 17 07:43:23 ip-172-31-26-115.us-east-2.compute.internal dockerd[8123]: time="2021-08-17T07
```

i-07139ec086f483569 (Docker)

Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

8) service docker stop



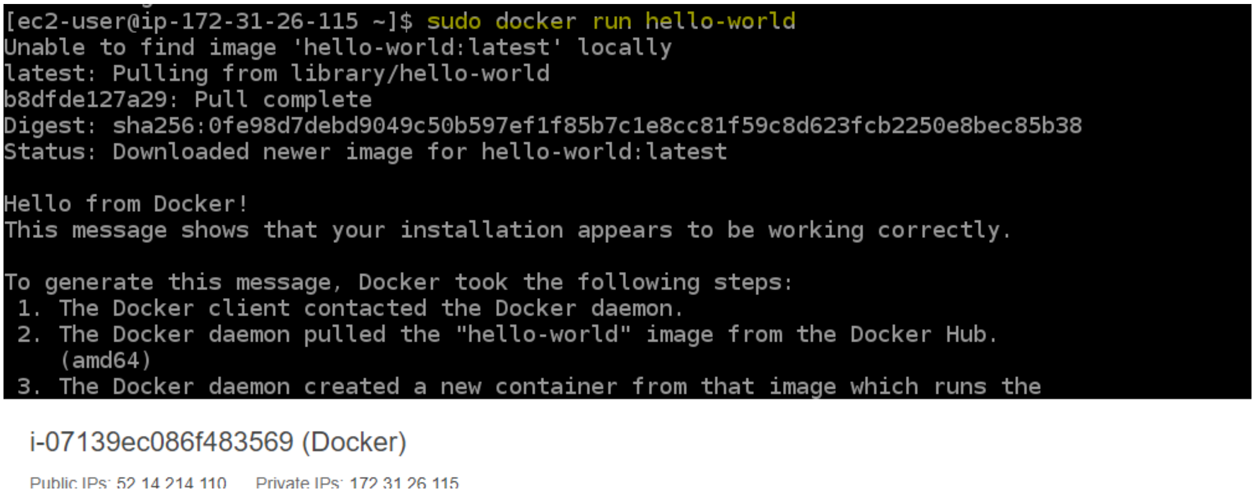
```
ec2-user@ip-172-31-26-115 ~]$ sudo service docker stop
edirecting to /bin/systemctl stop docker.service
arning: Stopping docker.service, but it can still be activated by:
 docker.socket
ec2-user@ip-172-31-26-115 ~]$ 
```

i-07139ec086f483569 (Docker)

Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

After this , again started the docker service.

9)docker run hello-world

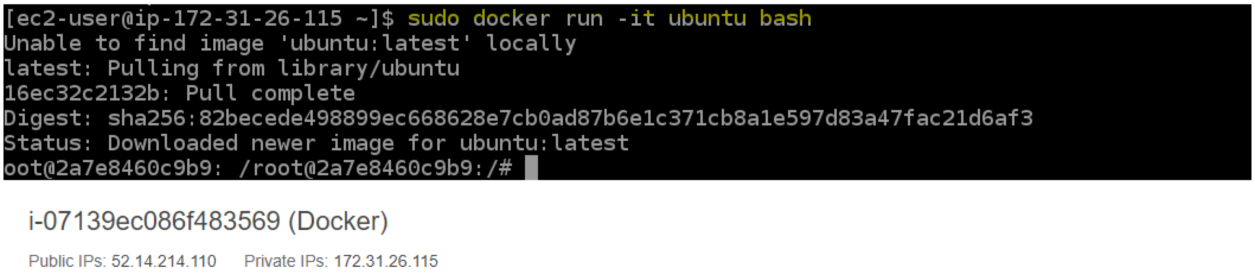


```
[ec2-user@ip-172-31-26-115 ~]$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
b8dfde127a29: Pull complete
Digest: sha256:0fe98d7debd9049c50b597ef1f85b7c1e8cc81f59c8d623fcb2250e8bec85b38
Status: Downloaded newer image for hello-world:latest

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
```

i-07139ec086f483569 (Docker)

Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

10) docker run -it ubuntu bash



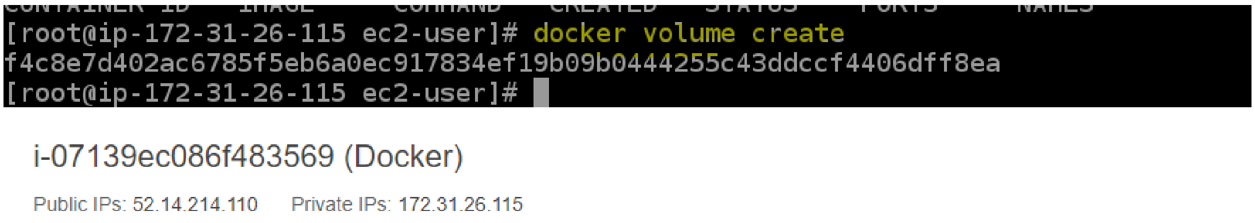
```
[ec2-user@ip-172-31-26-115 ~]$ sudo docker run -it ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
16ec32c2132b: Pull complete
Digest: sha256:82becede498899ec668628e7cb0ad87b6e1c371cb8a1e597d83a47fac21d6af3
Status: Downloaded newer image for ubuntu:latest
oot@2a7e8460c9b9: /root@2a7e8460c9b9:/#
```

i-07139ec086f483569 (Docker)

Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

Working with volume

Docker volume create:

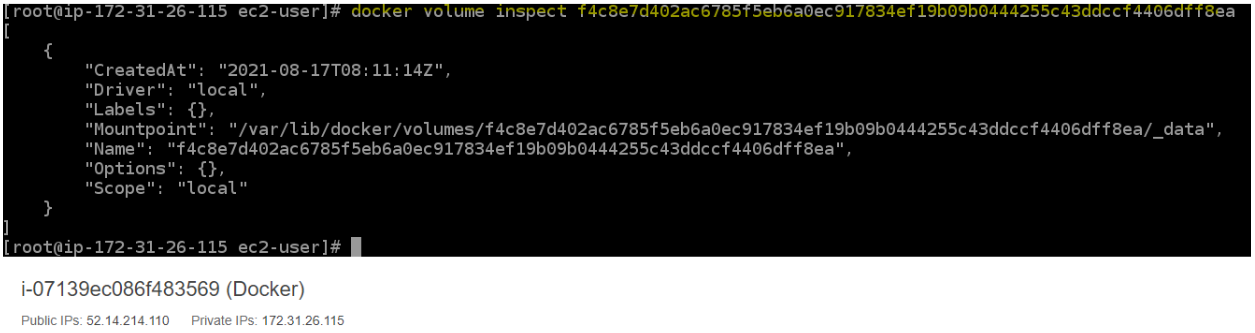


```
CONTAINER ID   IMAGE   COMMAND   CREATED   STATUS   PORTS   NAMES
[root@ip-172-31-26-115 ec2-user]# docker volume create
f4c8e7d402ac6785f5eb6a0ec917834ef19b09b0444255c43ddccf4406dff8ea
[root@ip-172-31-26-115 ec2-user]#
```

i-07139ec086f483569 (Docker)

Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

Docker volume inspect:

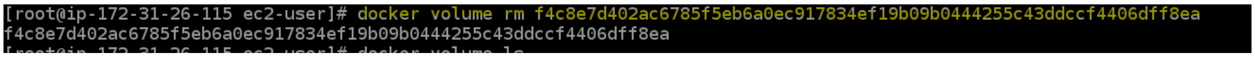


```
[root@ip-172-31-26-115 ec2-user]# docker volume inspect f4c8e7d402ac6785f5eb6a0ec917834ef19b09b0444255c43ddccf4406dff8ea
[
    {
        "CreatedAt": "2021-08-17T08:11:14Z",
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "/var/lib/docker/volumes/f4c8e7d402ac6785f5eb6a0ec917834ef19b09b0444255c43ddccf4406dff8ea/_data",
        "Name": "f4c8e7d402ac6785f5eb6a0ec917834ef19b09b0444255c43ddccf4406dff8ea",
        "Options": {},
        "Scope": "local"
    }
]
[root@ip-172-31-26-115 ec2-user]#
```

i-07139ec086f483569 (Docker)

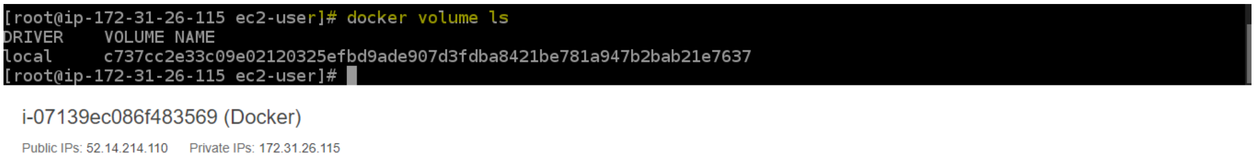
Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

Docker volume rm volid



```
[root@ip-172-31-26-115 ec2-user]# docker volume rm f4c8e7d402ac6785f5eb6a0ec917834ef19b09b0444255c43ddccf4406dff8ea
f4c8e7d402ac6785f5eb6a0ec917834ef19b09b0444255c43ddccf4406dff8ea
[root@ip-172-31-26-115 ec2-user]# docker volume ls
```

Docker volume ls

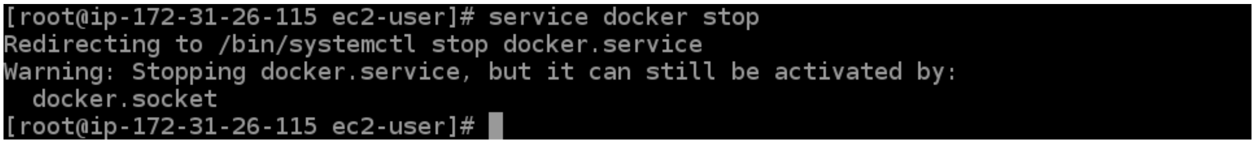


```
[root@ip-172-31-26-115 ec2-user]# docker volume ls
DRIVER    VOLUME NAME
local     c737cc2e33c09e02120325efbd9ade907d3fdba8421be781a947b2bab21e7637
[root@ip-172-31-26-115 ec2-user]#
```

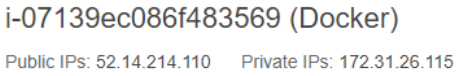
i-07139ec086f483569 (Docker)

Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

Service docker stop:



```
[root@ip-172-31-26-115 ec2-user]# service docker stop
Redirecting to /bin/systemctl stop docker.service
Warning: Stopping docker.service, but it can still be activated by:
  docker.socket
[root@ip-172-31-26-115 ec2-user]# 
```



i-07139ec086f483569 (Docker)

Public IPs: 52.14.214.110   Private IPs: 172.31.26.115

------- Assignment end ----------------------