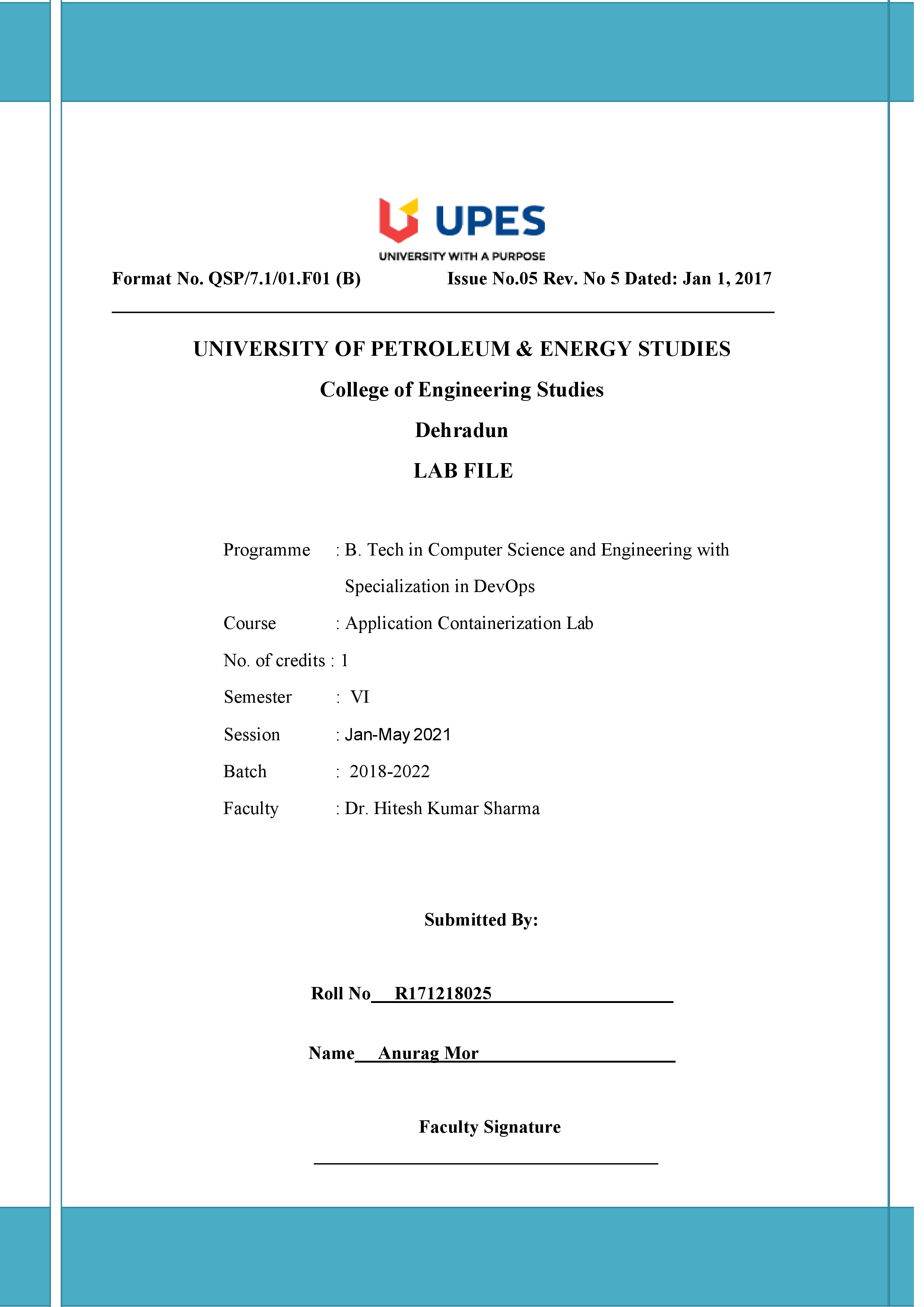
****

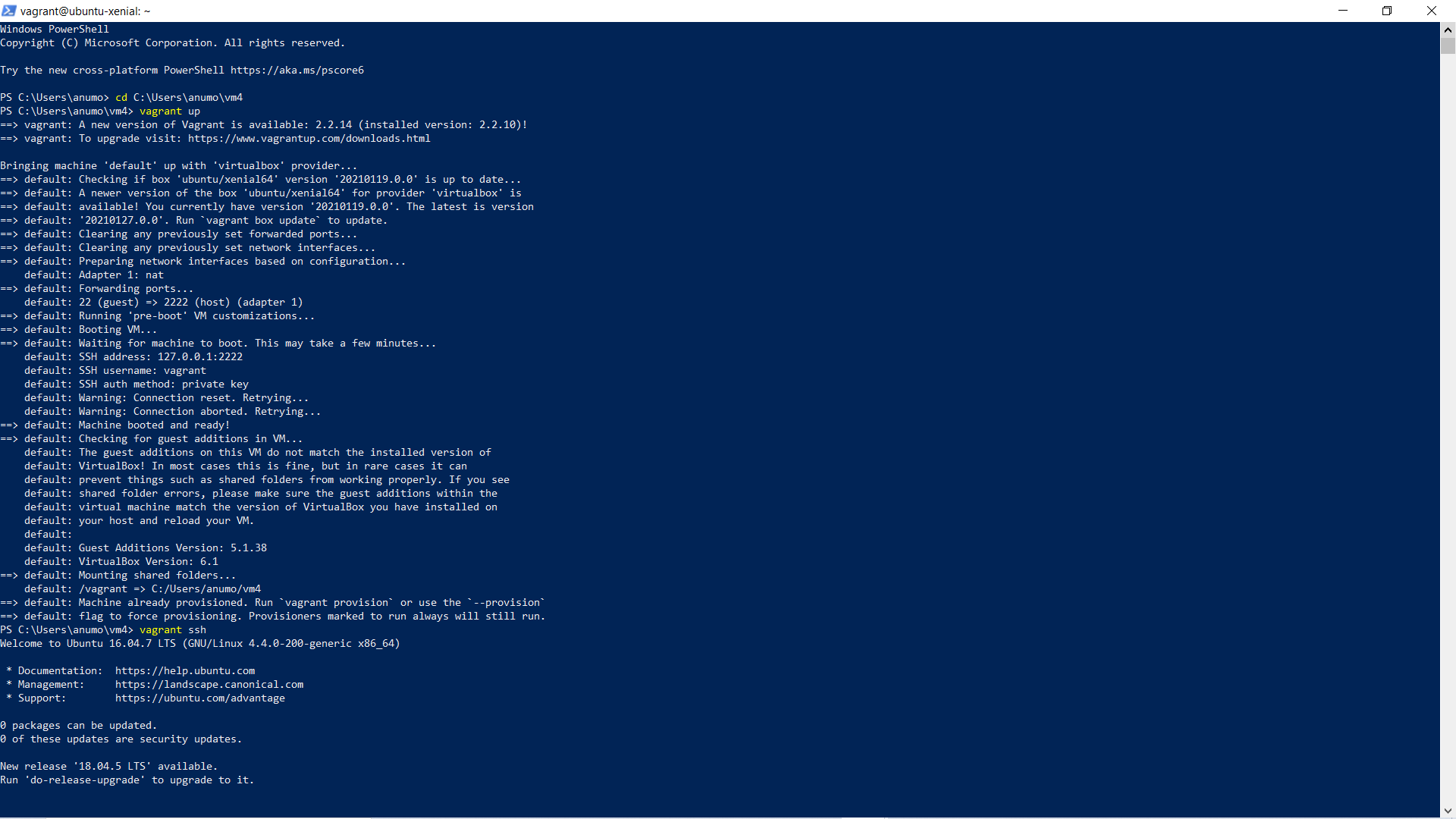
**Experiment 2 Date: - 29-01-2021**

**Aim: - To create volume in Docker container and perform various tasks.**

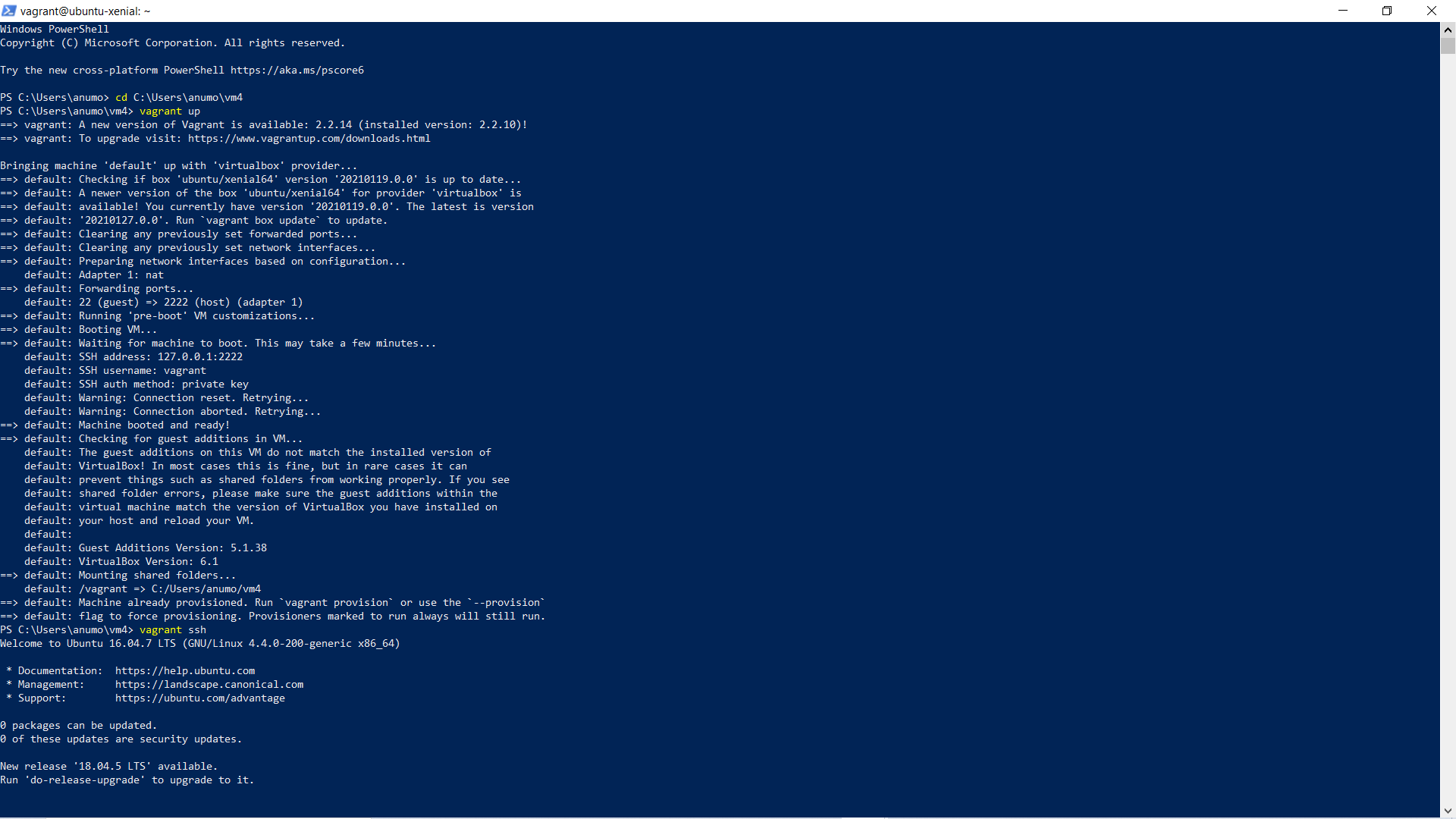
**Solution: -**

1. Now we can run the vagrant on command prompt or PowerShell using command: -

$ Vagrant up

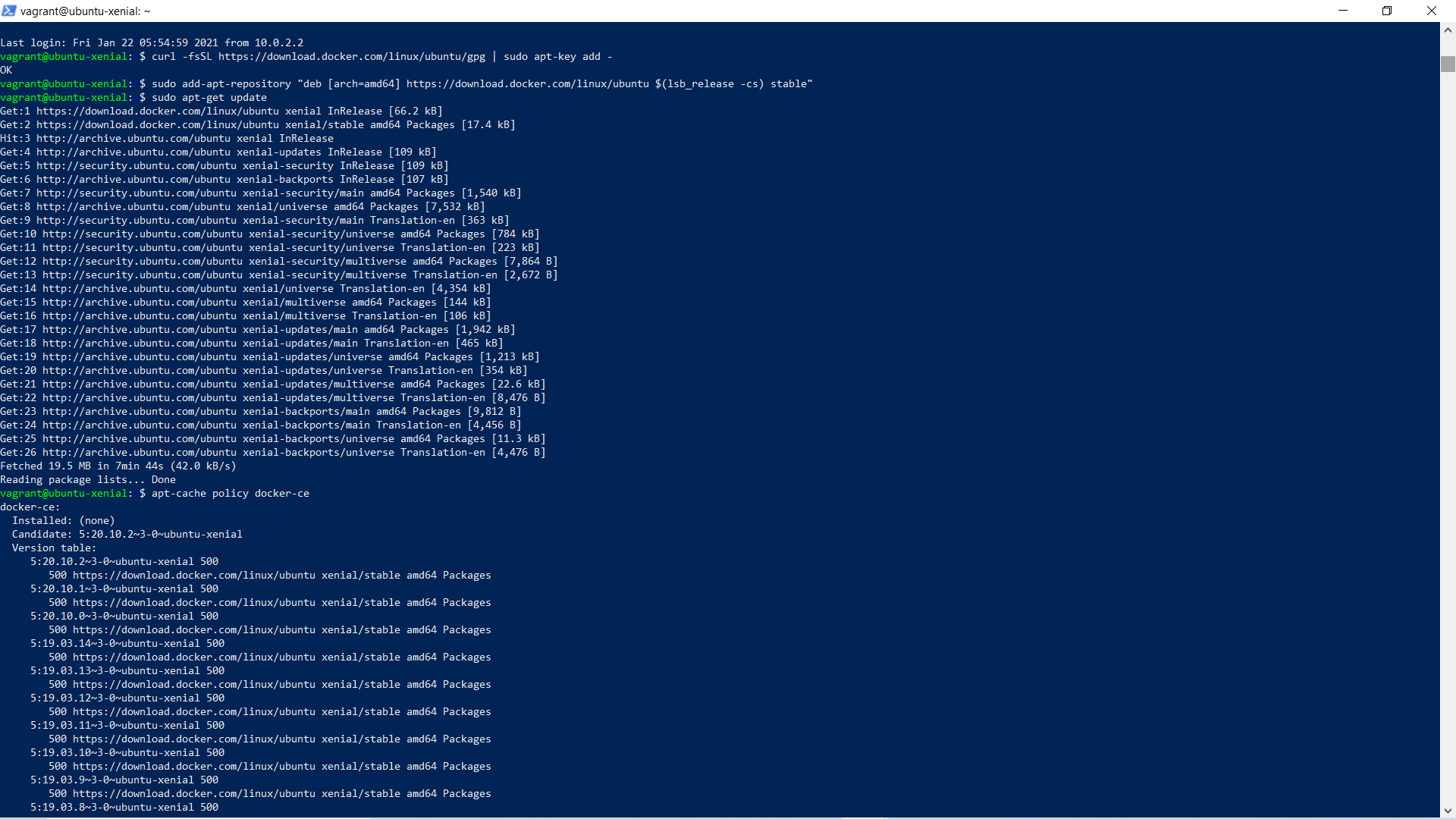


1. To ssh into the virtual machine simply run the below command: -   
   $ Vagrant ssh



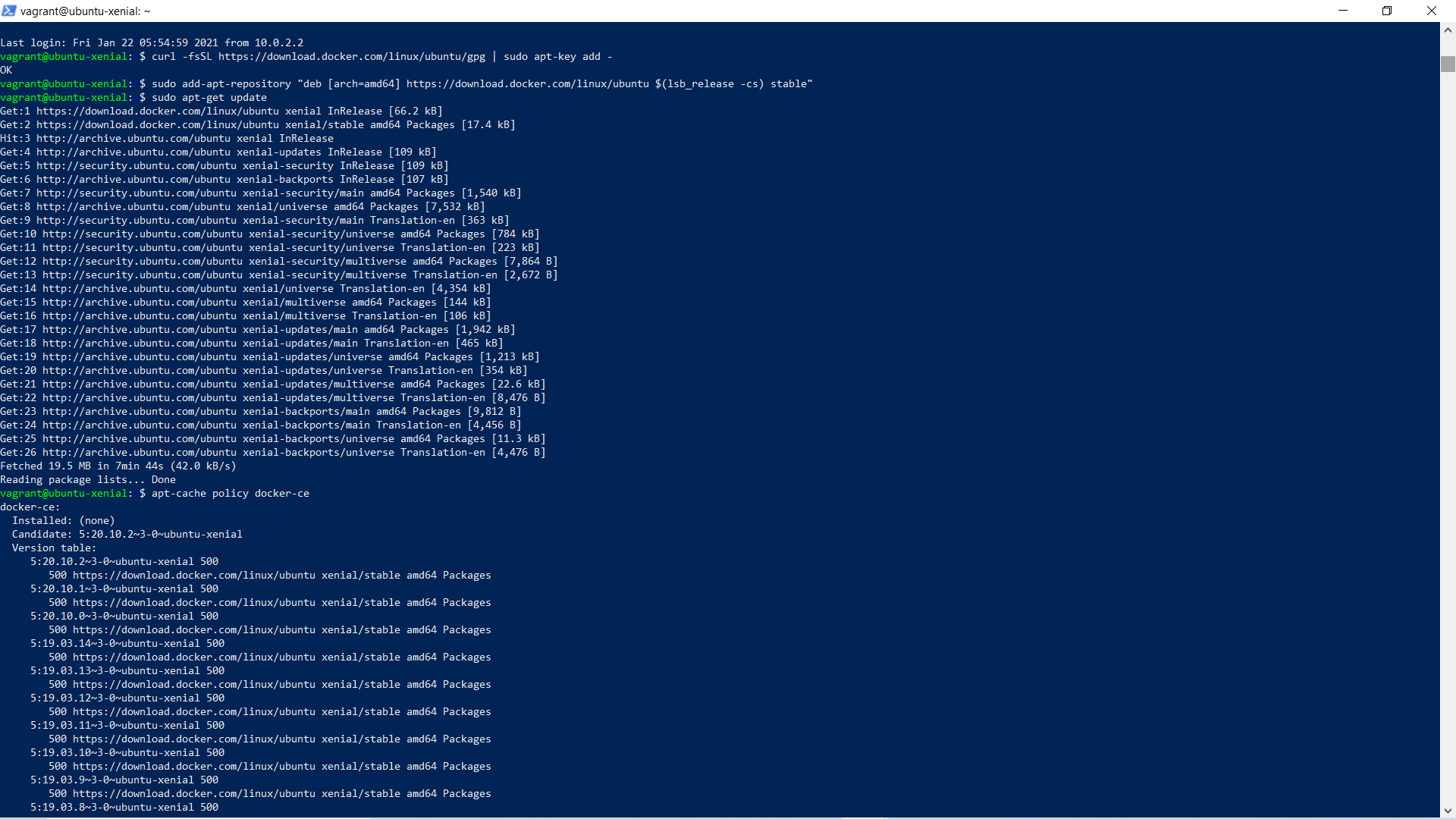
1. First, in order to ensure the downloads are valid, add the GPG key for the official Docker repository to your system:

$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add –



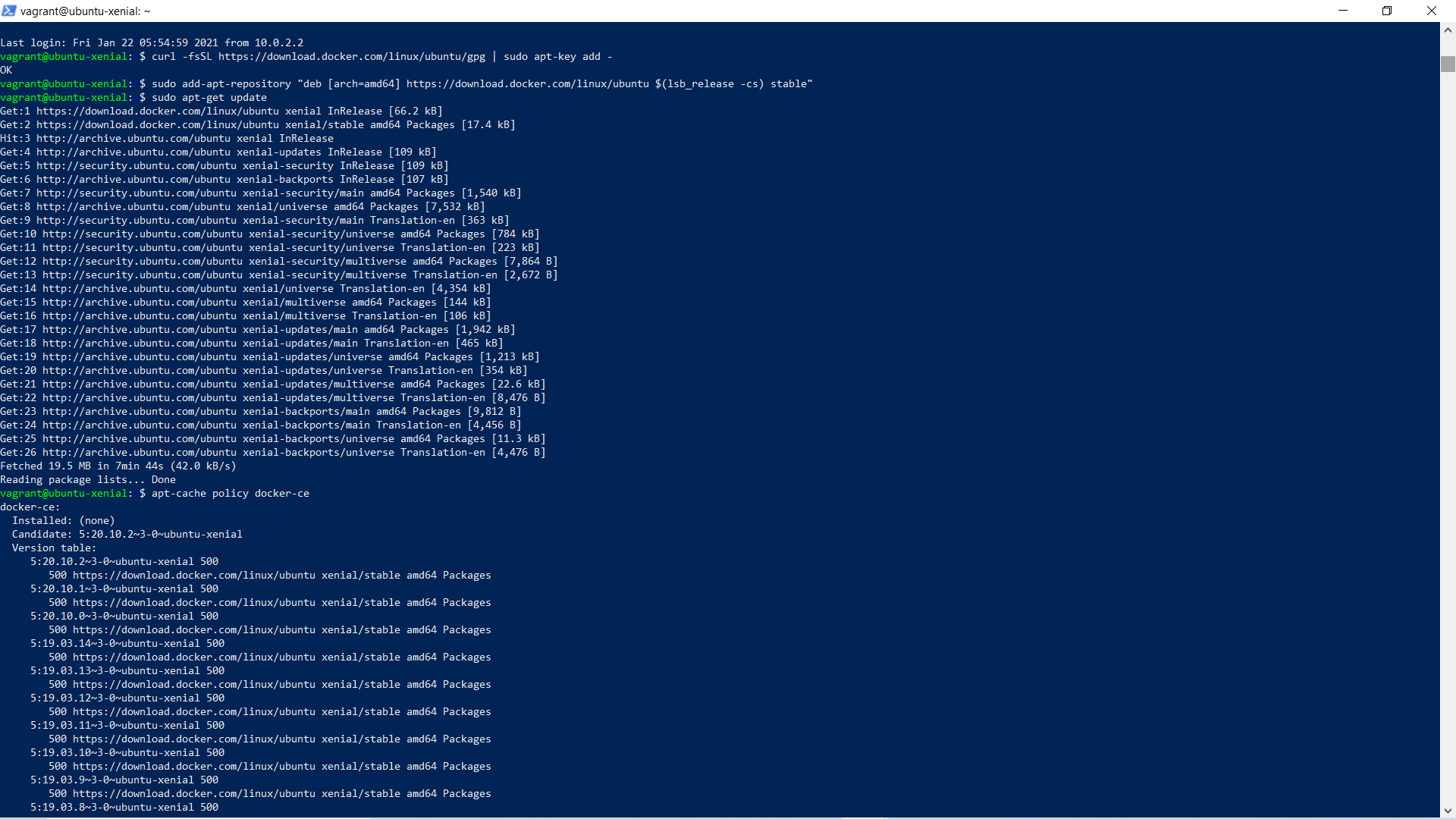
1. Add the Docker repository to APT sources:

$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb\_release -cs) stable"



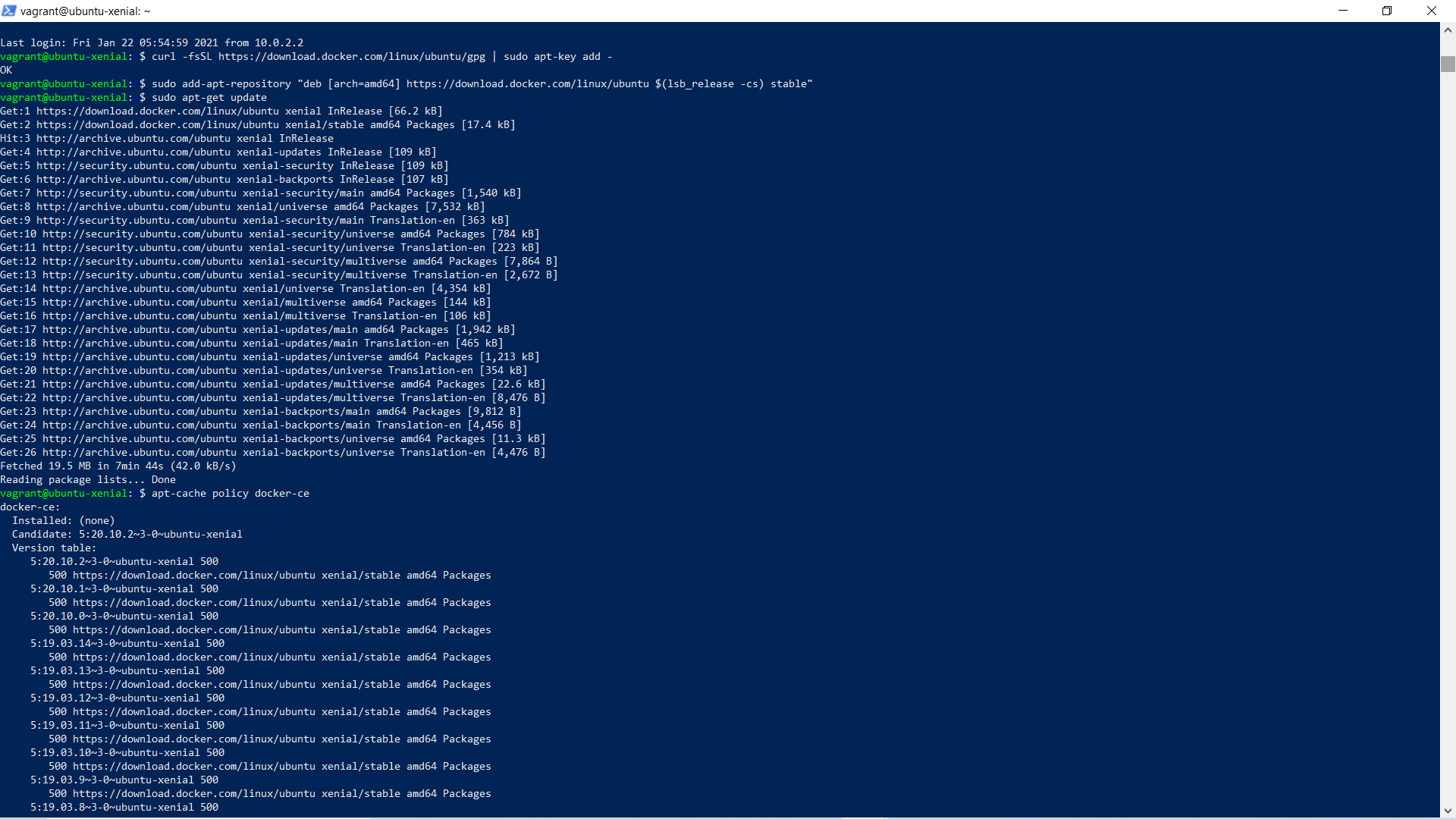
1. Next, update the package database with the Docker packages from the newly added repo:

$ sudo apt-get update



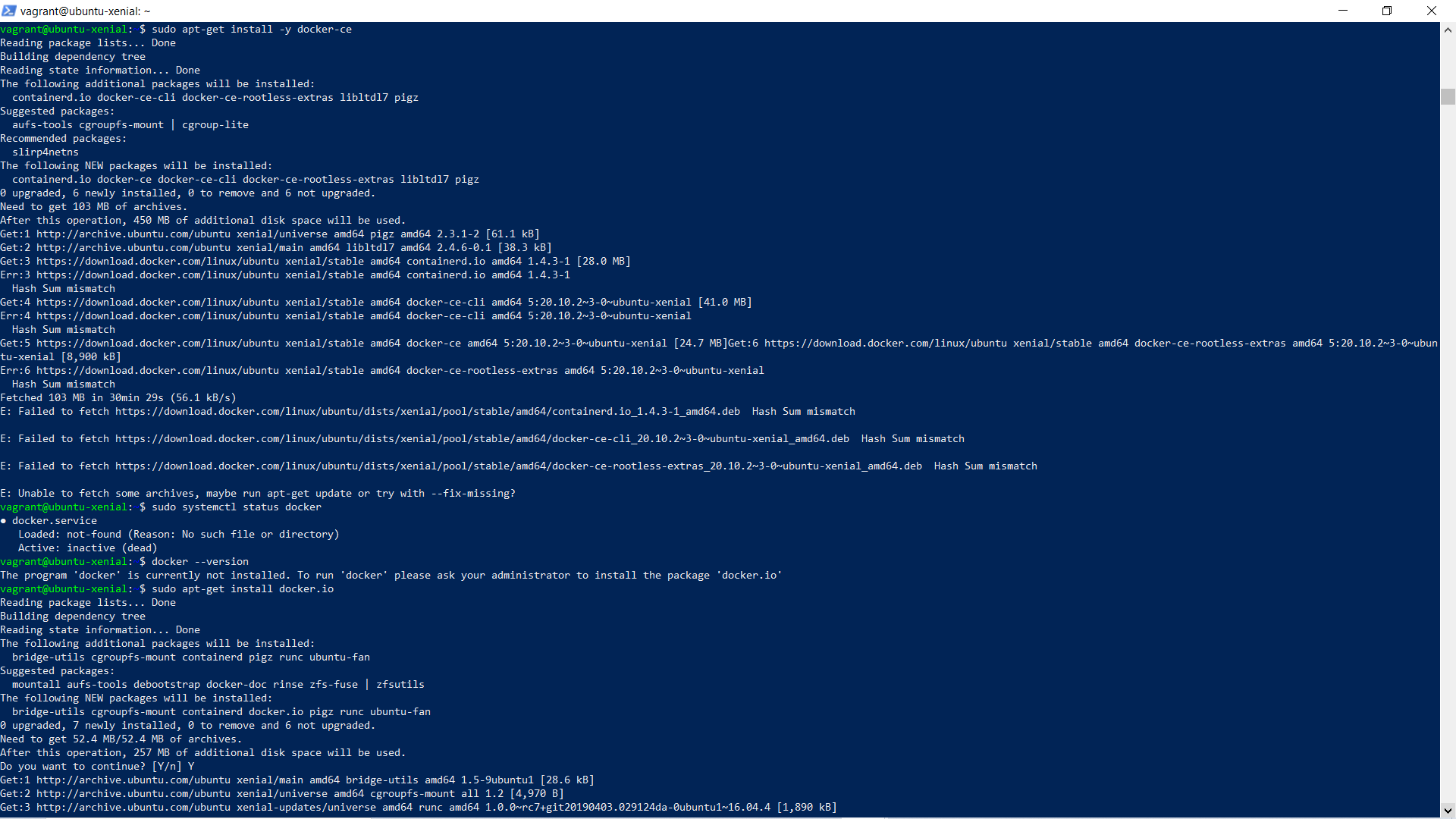
1. Make sure you are about to install from the Docker repo instead of the default Ubuntu 16.04 repo:

$ apt-cache policy docker-ce



1. Finally, install Docker:

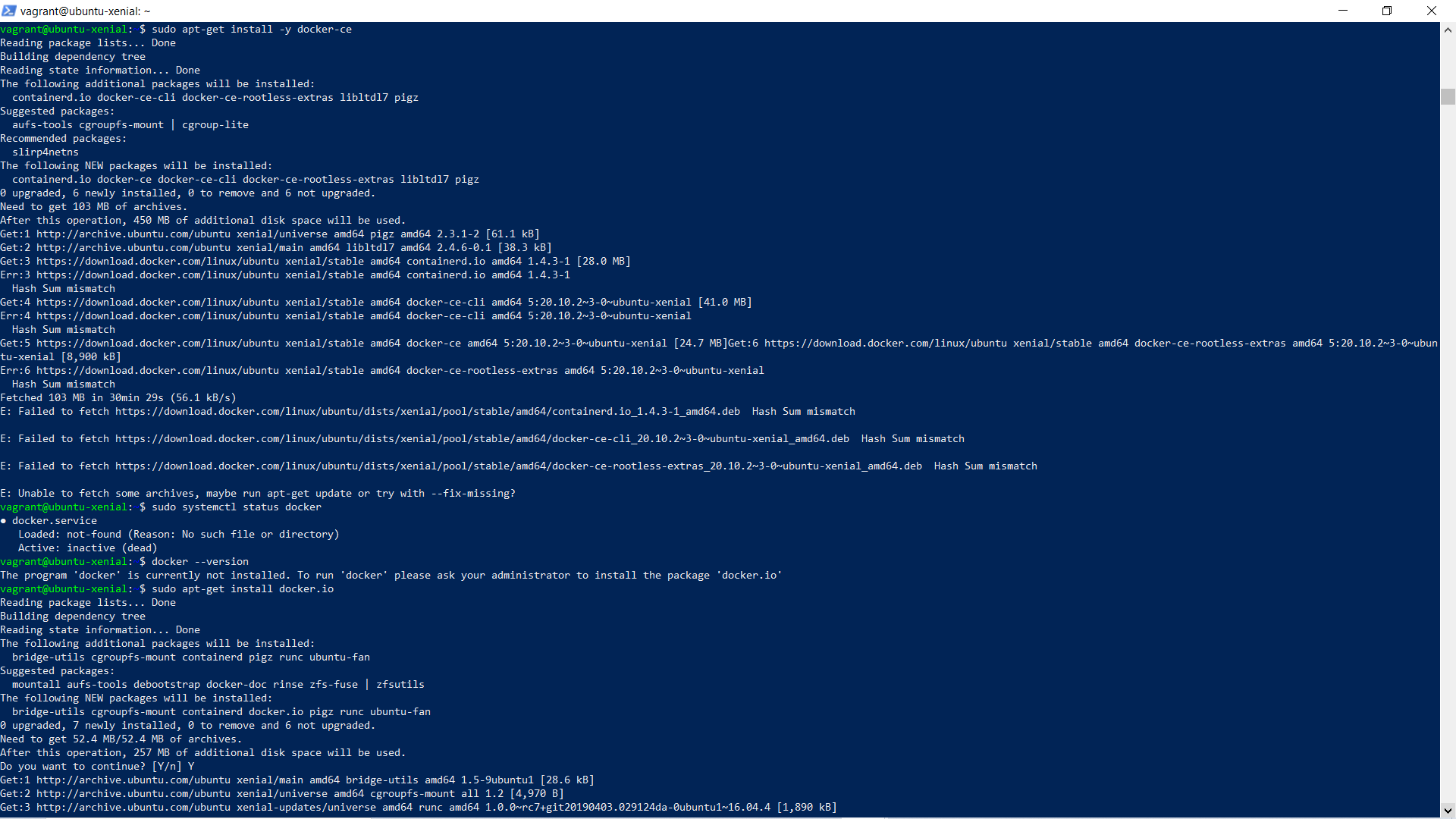
$ sudo apt-get install -y docker-ce



Or

Install docker using: -

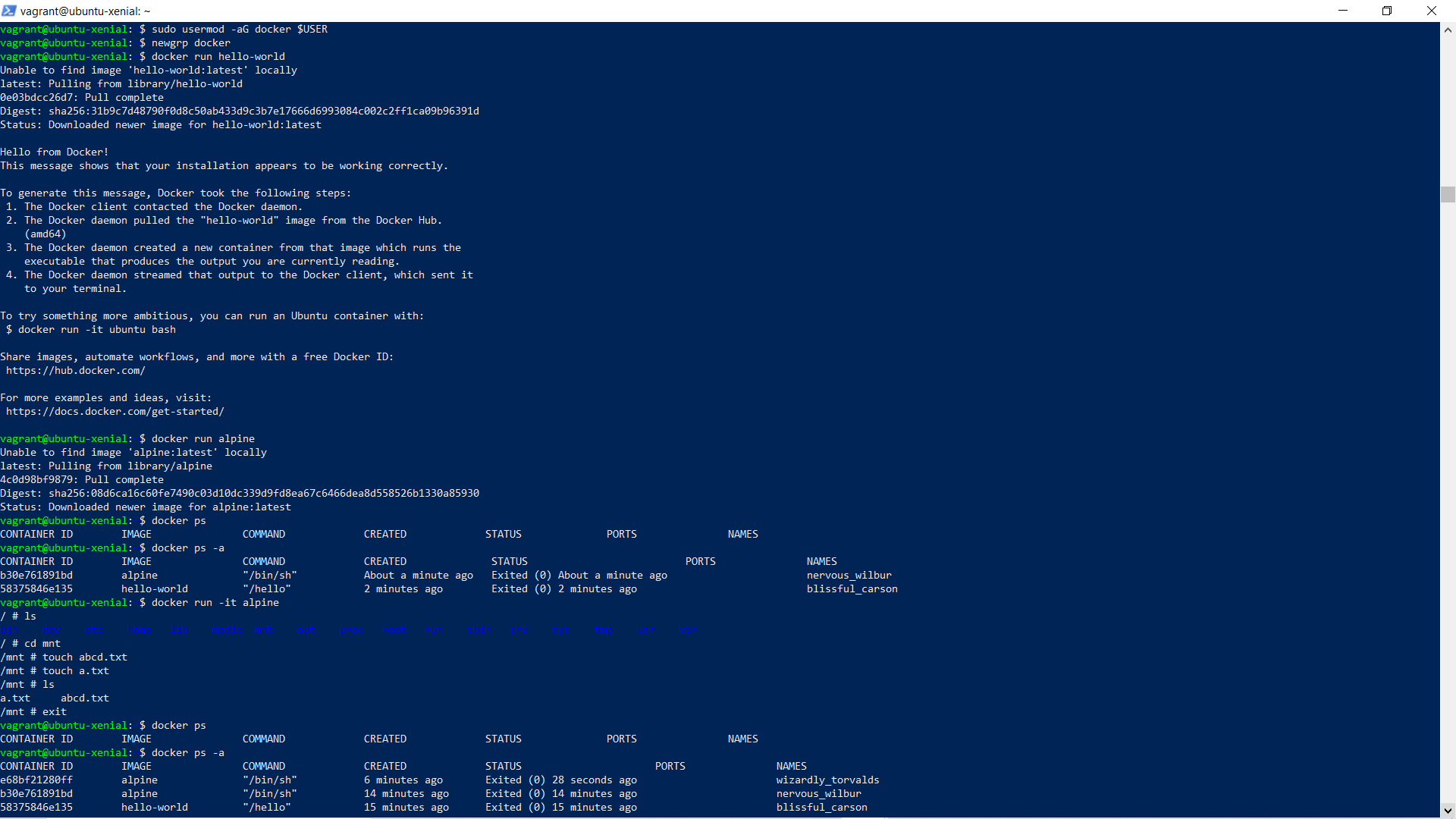
sudo apt-get install -y docker.io



1. Executing the Docker Command Without Sudo

If you want to avoid typing sudo whenever you run the docker command, add your username to the docker group:

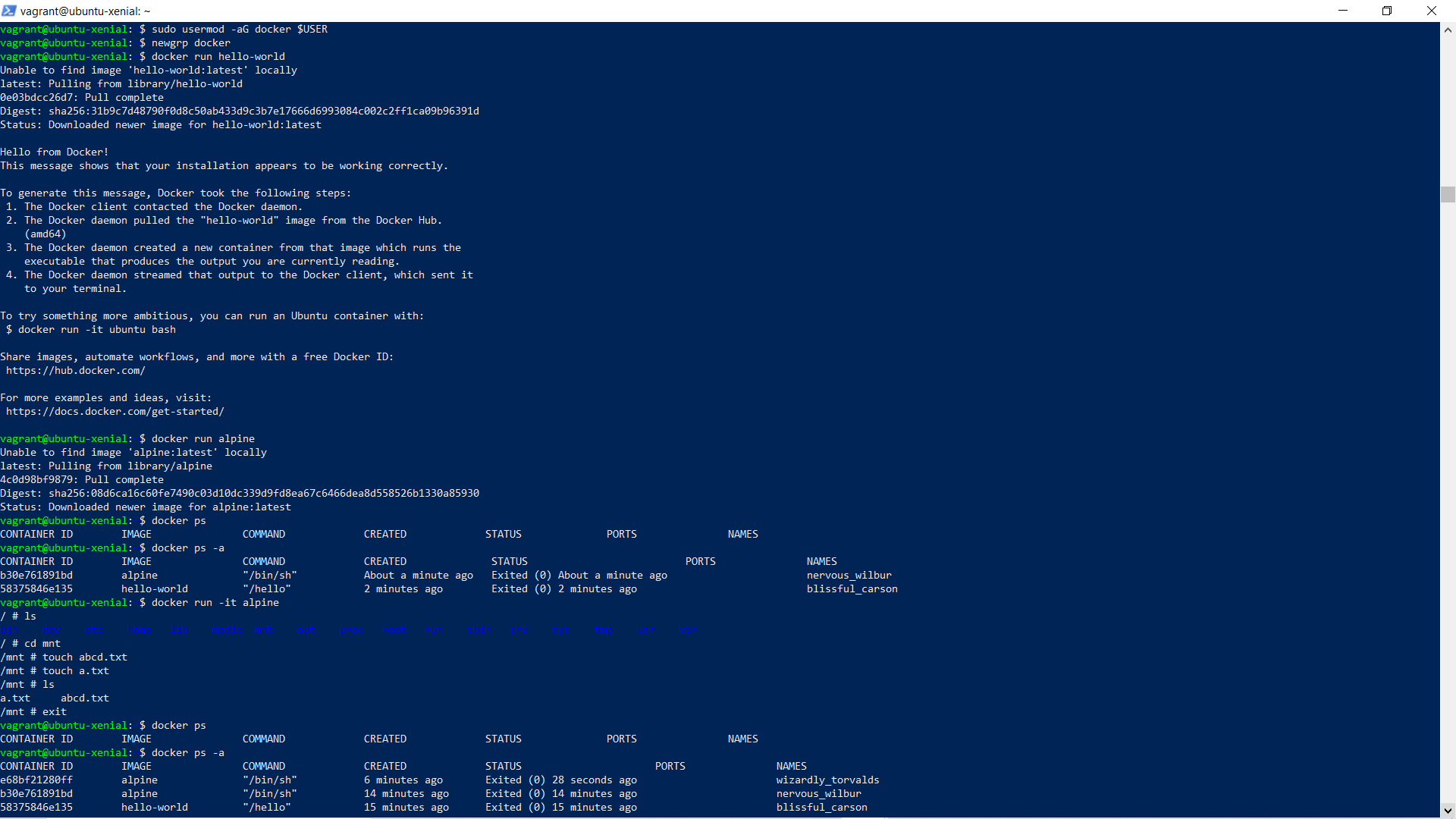
$ sudo usermod -aG docker ${USER}



1. To apply the new group membership, you can log out of the server and back in, or you can type the following:

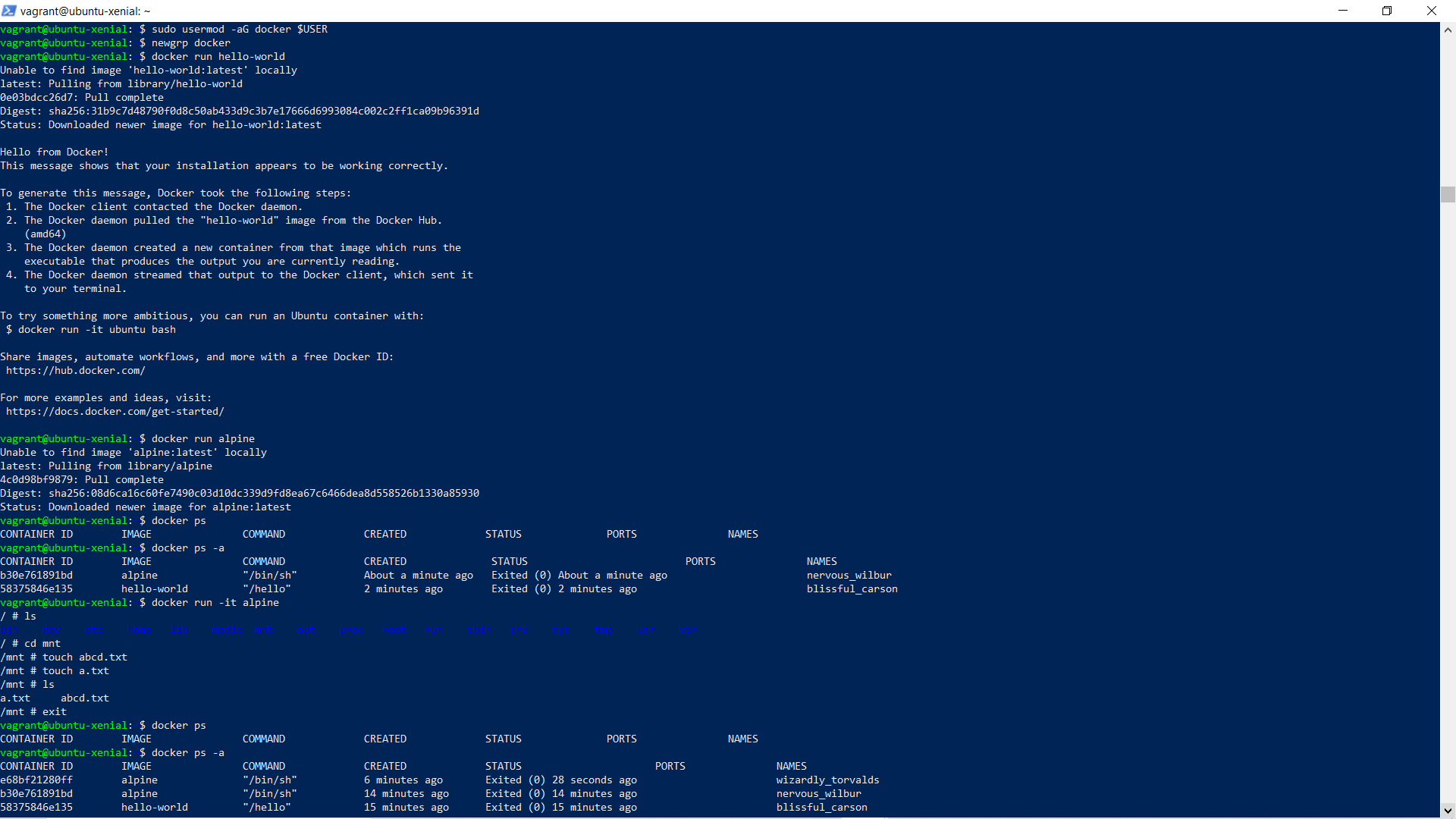
$ su - ${USER}

You will be prompted to enter your user’s password to continue. Afterwards, you can confirm that your user is now added to the docker group by typing:



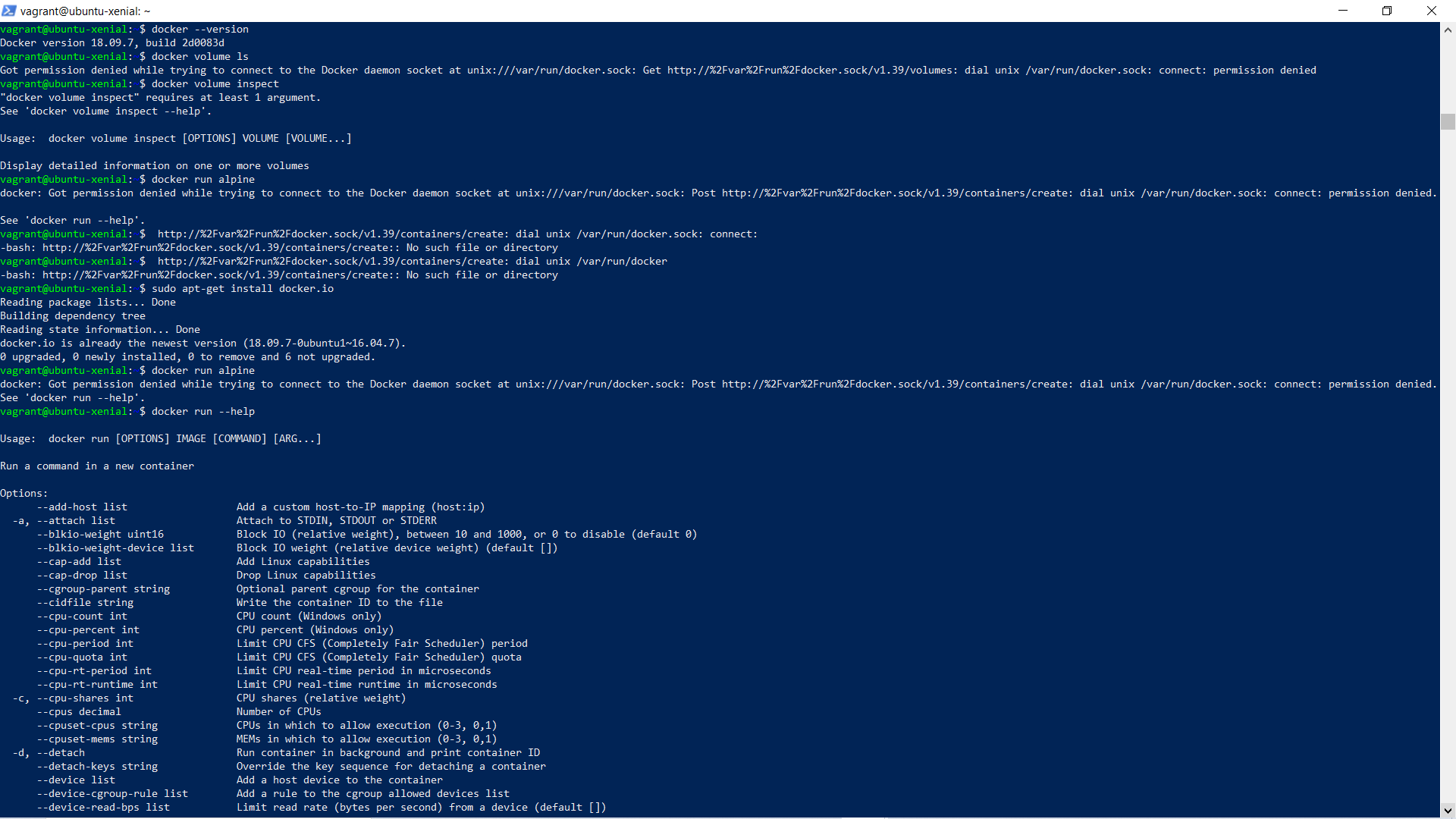
1. To check whether you can access and download images from Docker Hub, type:

$ docker run hello-world



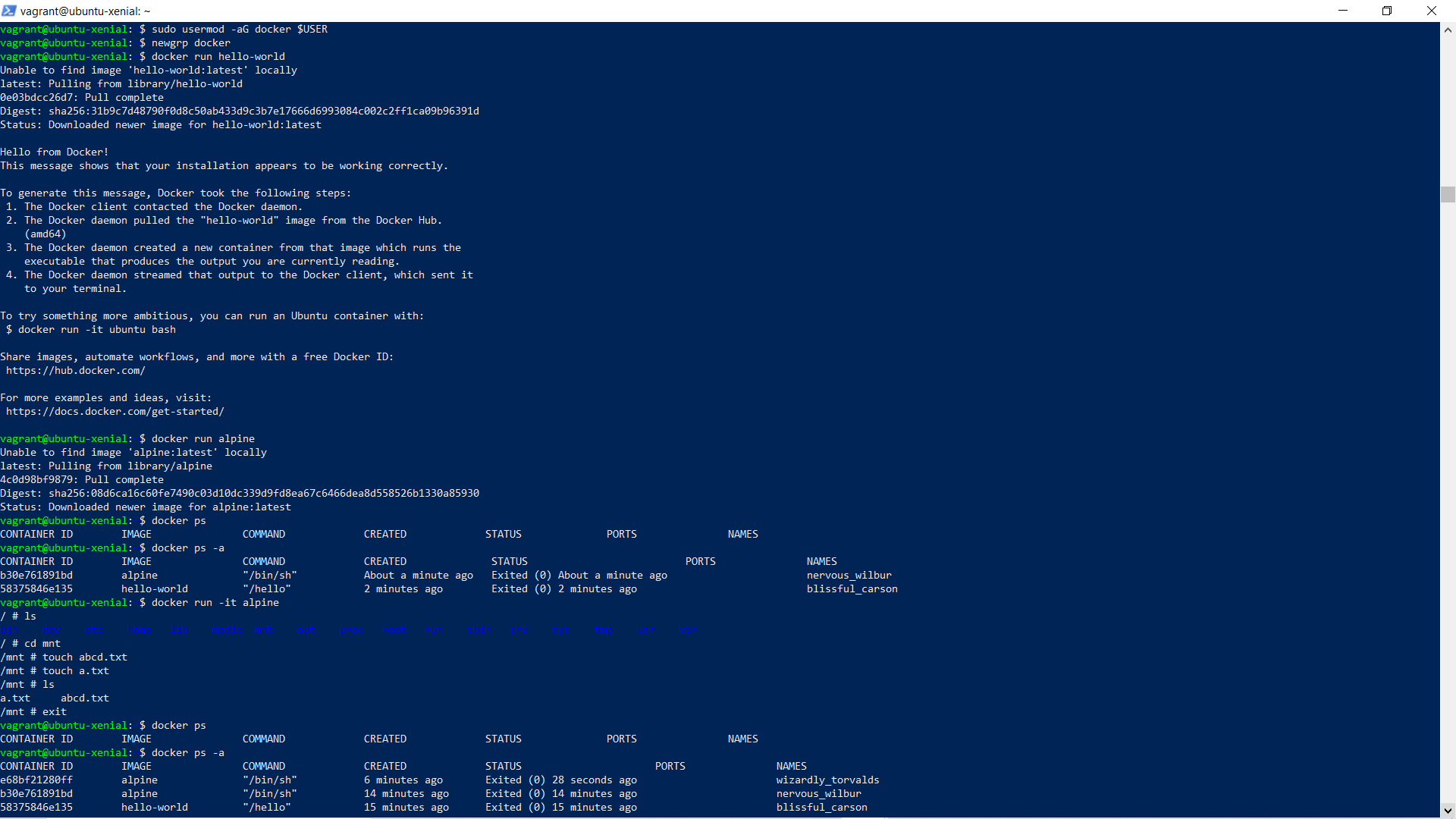
1. Now to check the version of docker type the command :-

$ docker –version



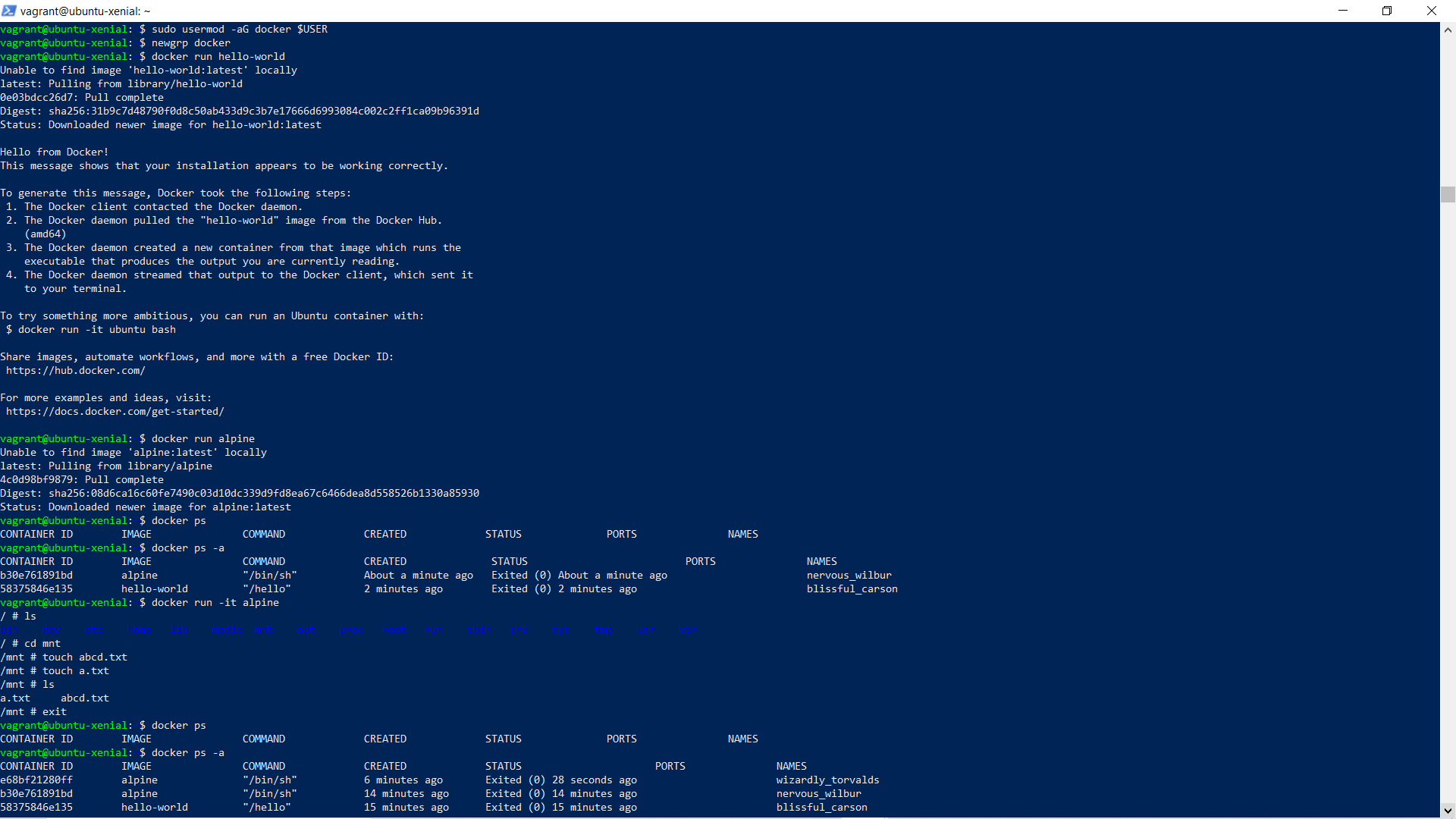
1. To make volume and files in docker we need an ubuntu ,so we will install and run alpine ubuntu primary using command:-

$ docker run alpine



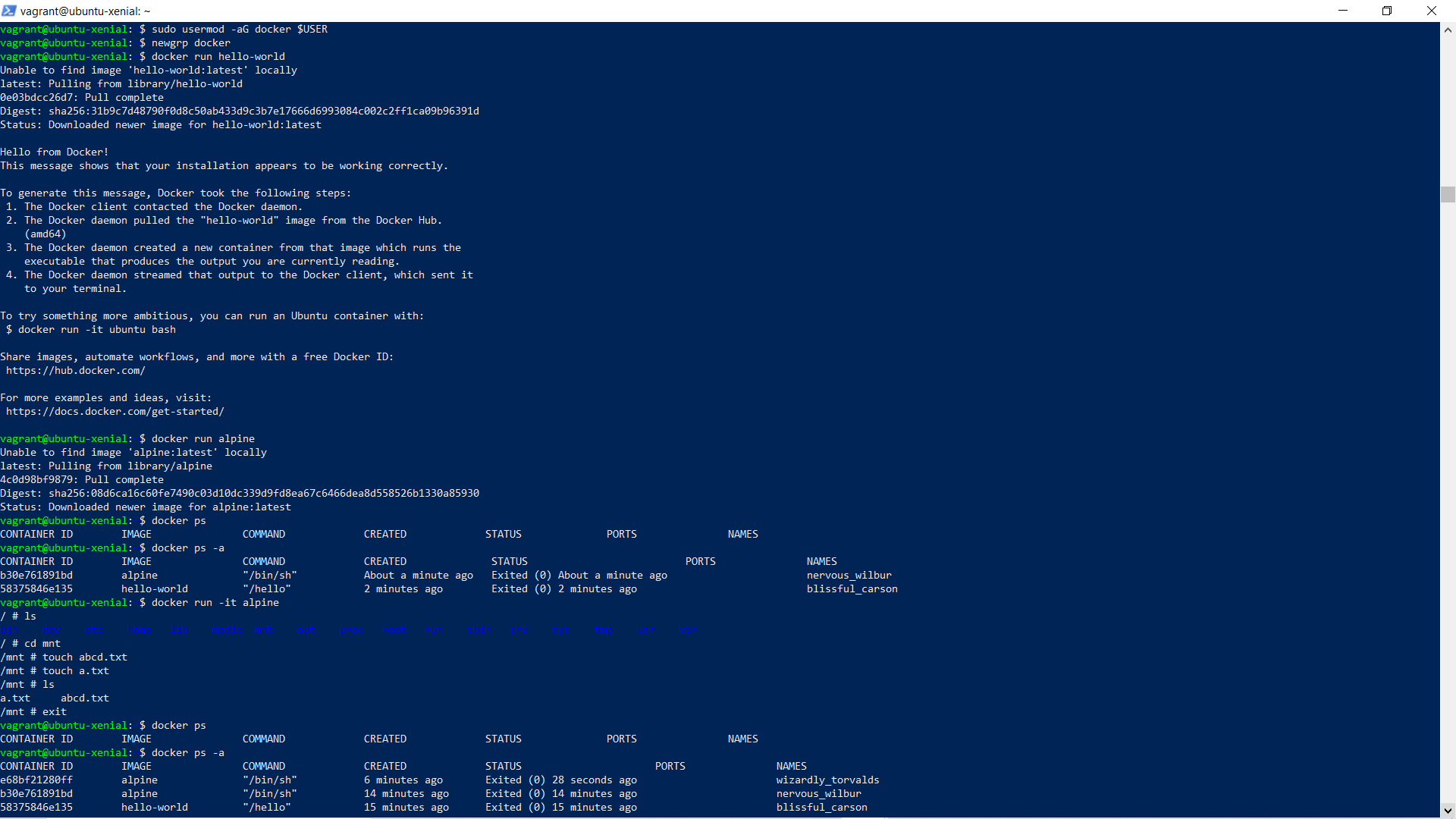
1. To check which images are running in docker image we use the command given below which tell all the details of image:-

$ docker ps -a



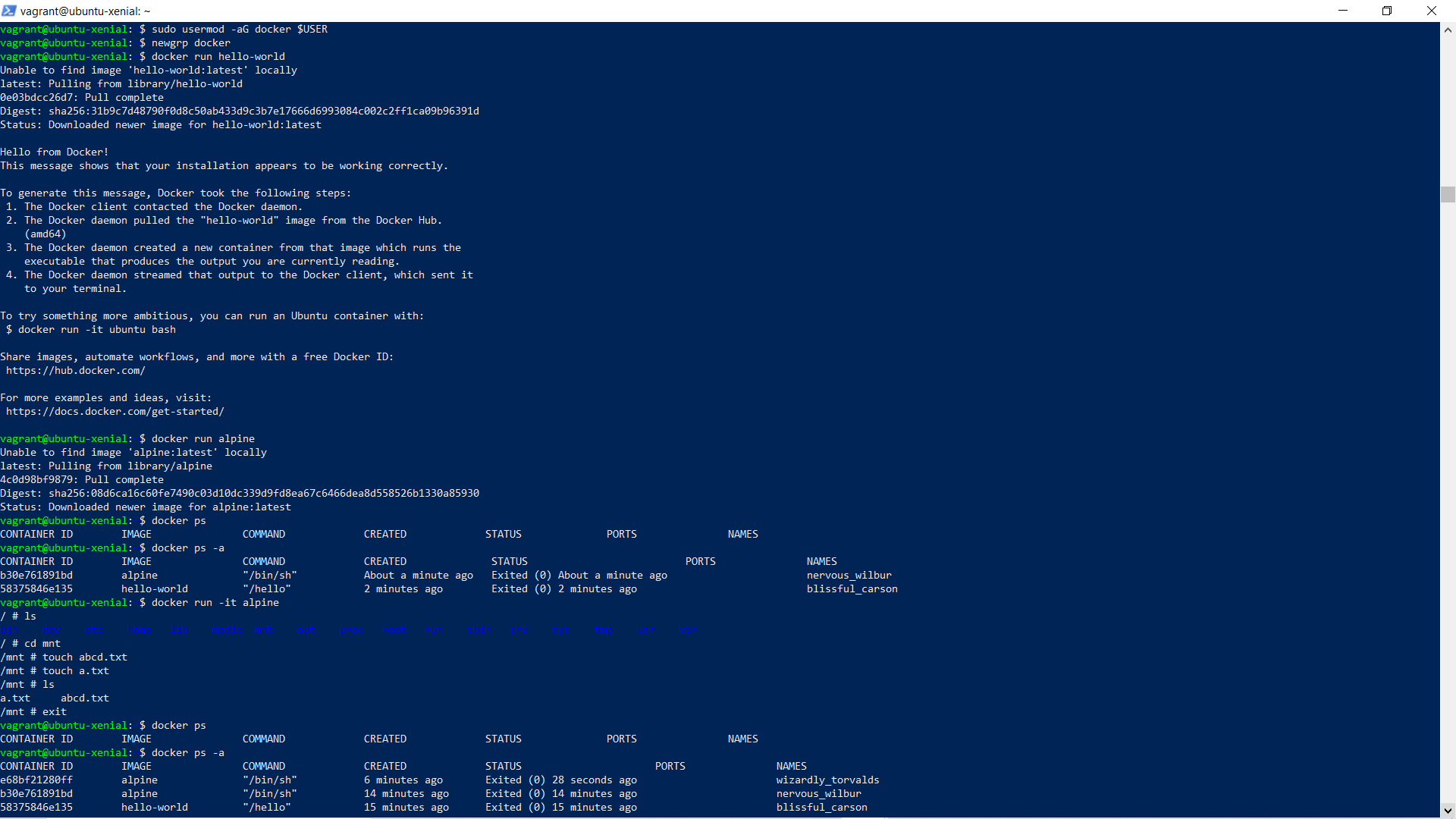
1. To go inside alpine image and make a file we use the command :-

$ docker run -it alpine



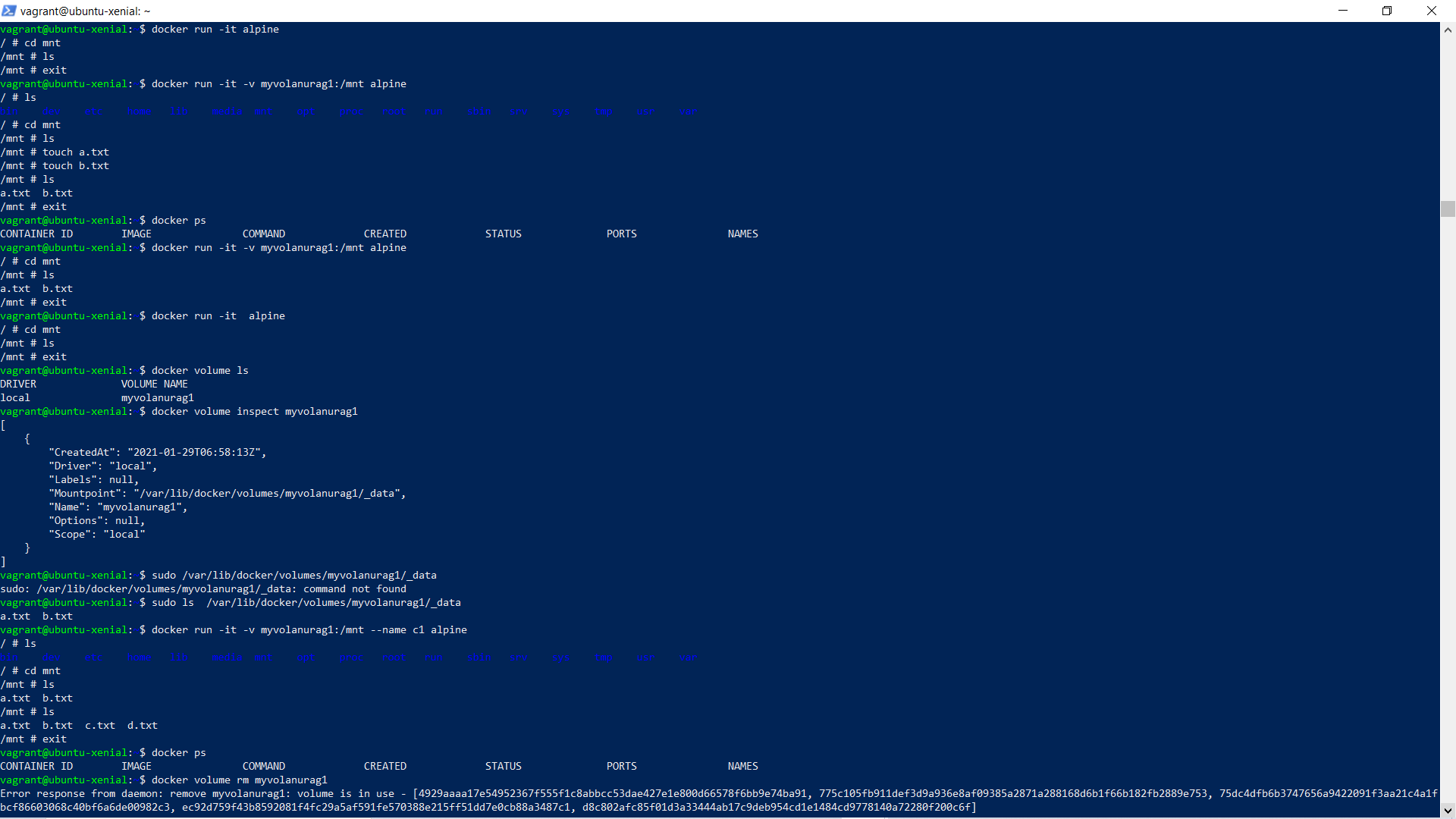
1. Now check again whether the alpine image is running or had been exited using the command:-

$ docker ps -a



1. Now we seen that the alpine image had been exited , now go again in alpine image and check that the file you made is still there or not using command :-

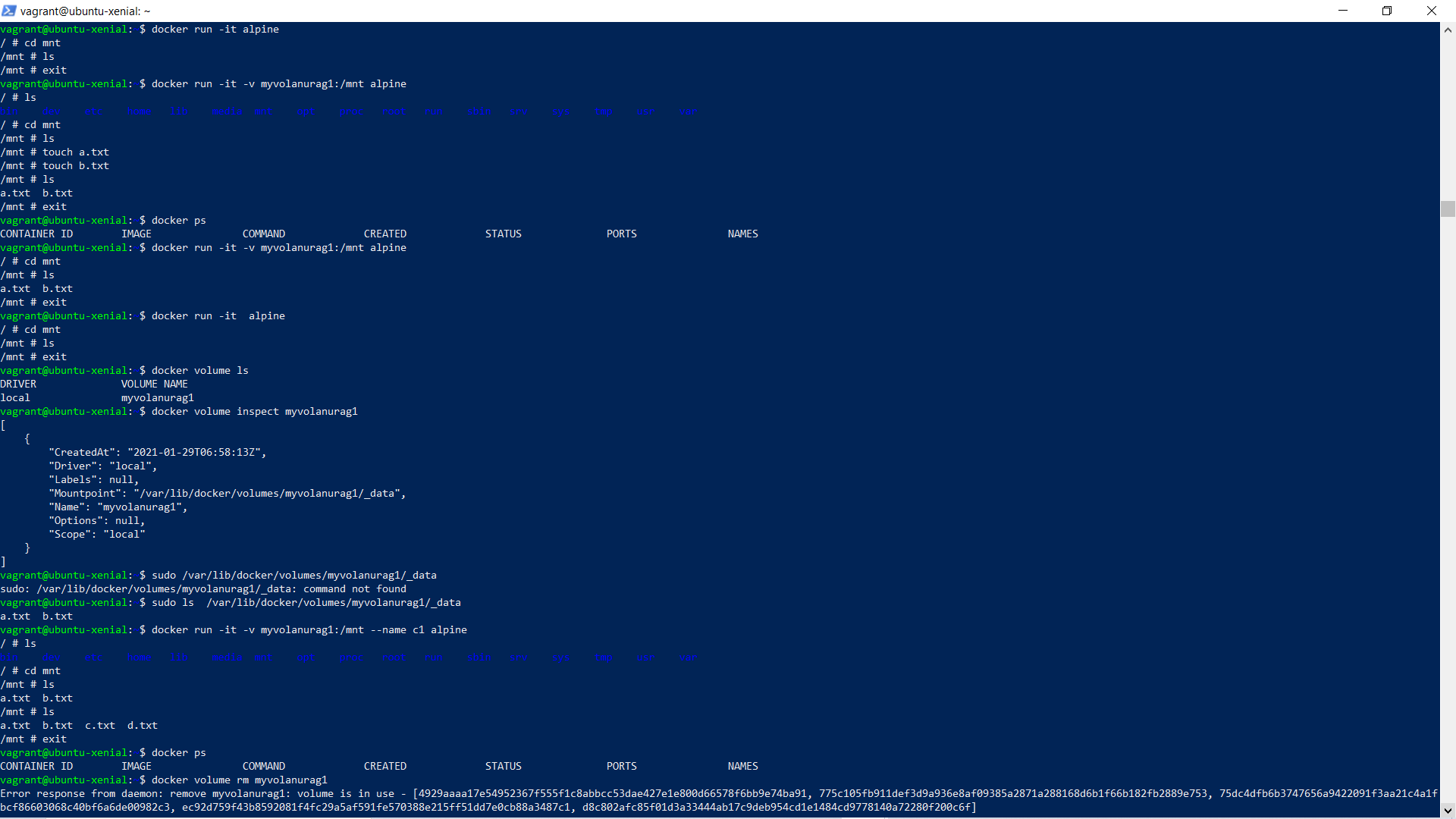
$ docker run -it alpine



We did not find the made file.

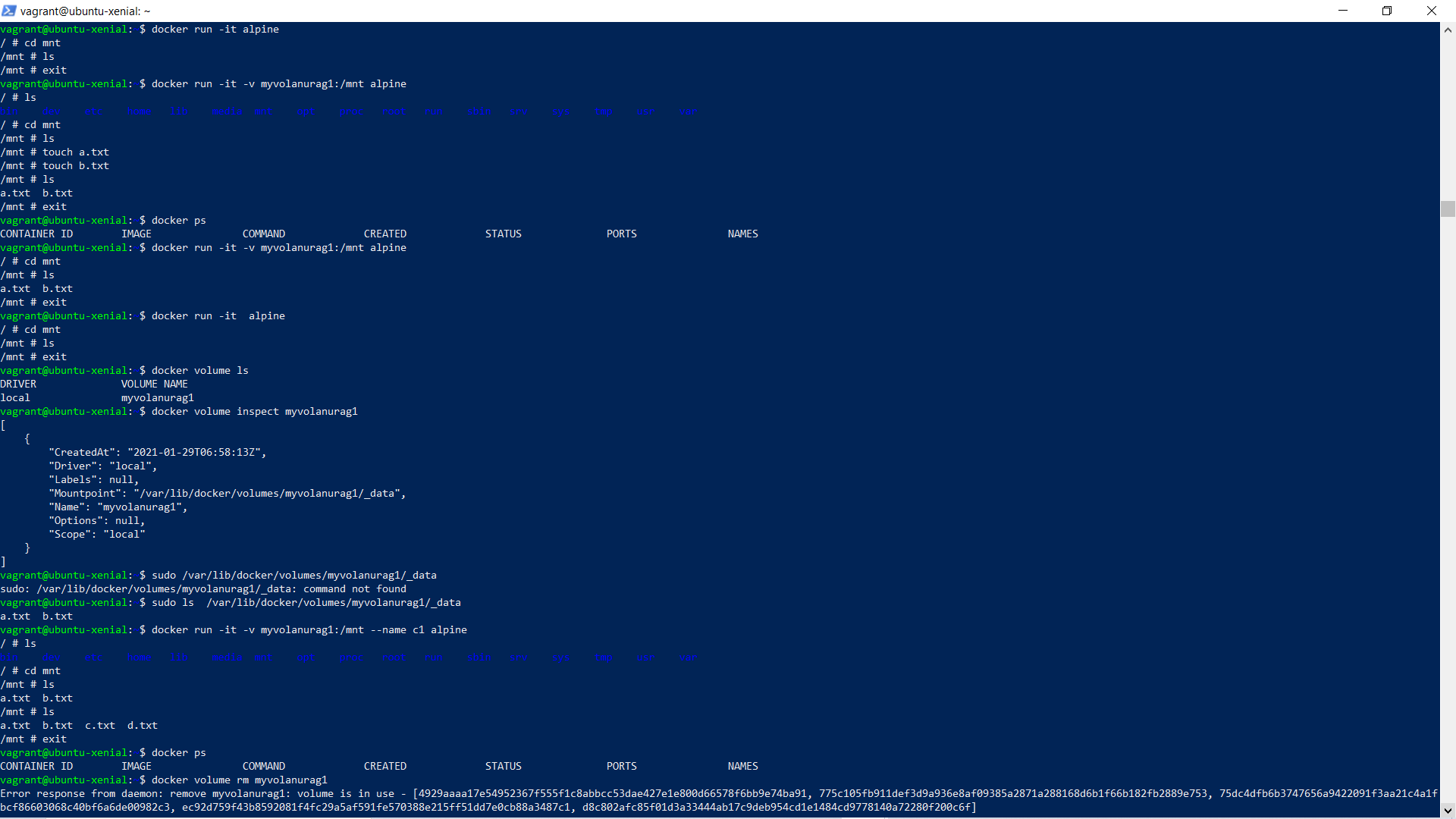
1. To make file visible after exit, we need to create a volume and then enter in image , and create the file using the command :-

$ docker run -it -v myvolanurag1:/mnt alpine



1. Now again login to alpine image to check whether the made file are there or not using command:-

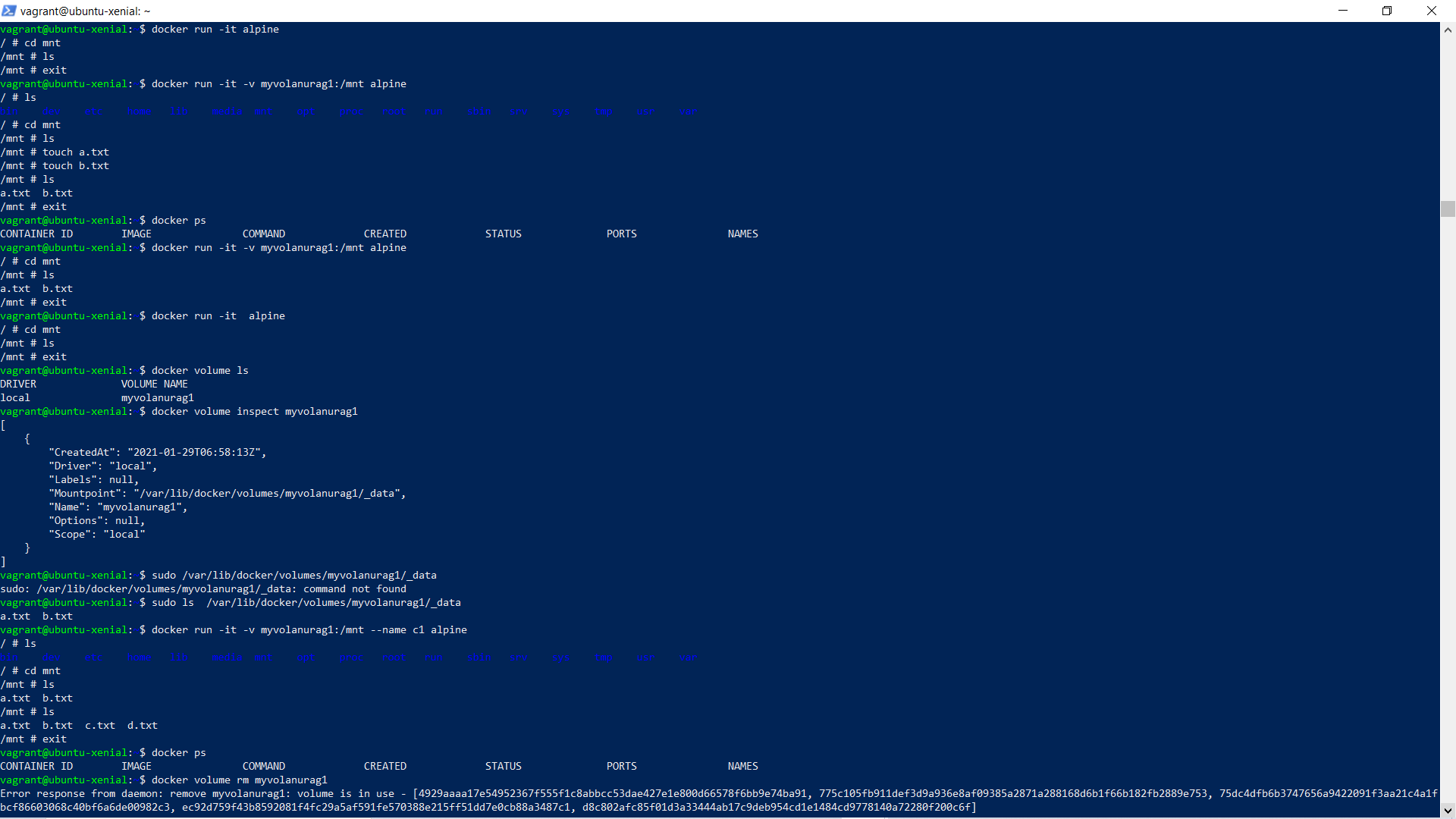
$ docker run -it -v myvolanurag1:/mnt alpine



The files are visible even after exited.

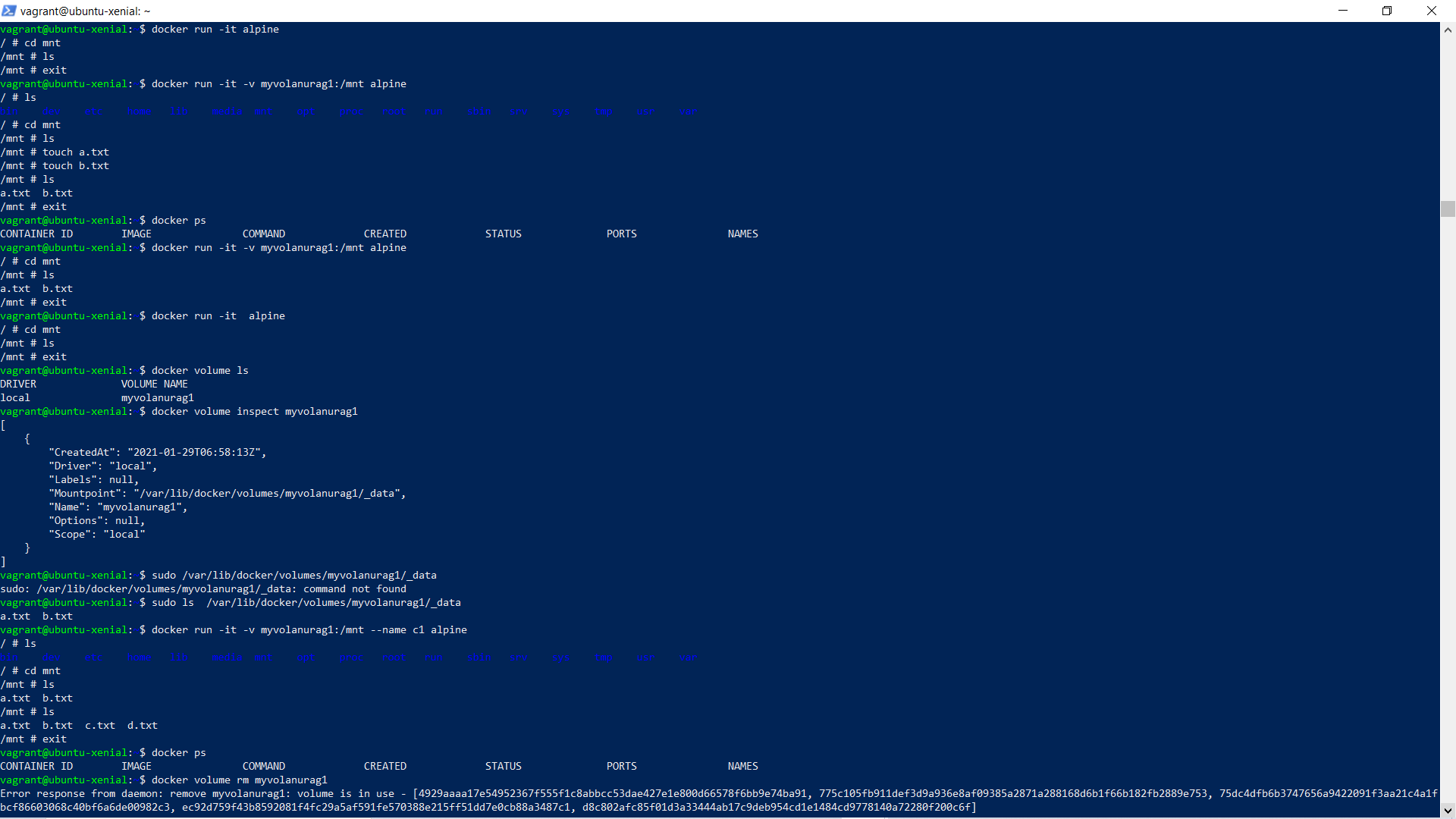
1. To check the volume present in the docker we use the command :

$ docker volume ls



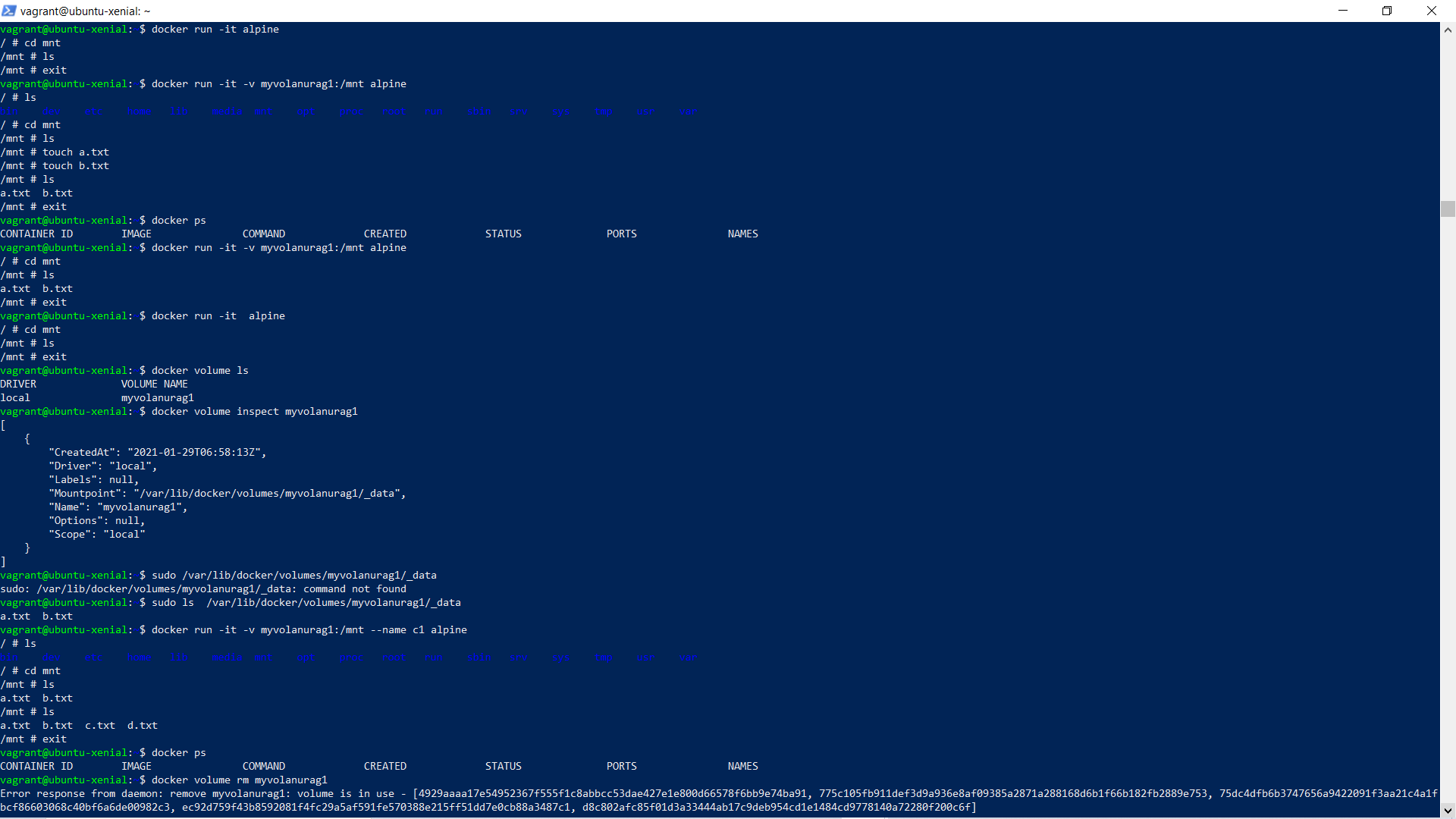
1. To check the local of the volume we use the command :-

$ docker volume inspect myvolanurag1



1. We can also check the file made by going to the location of volume also using command :-

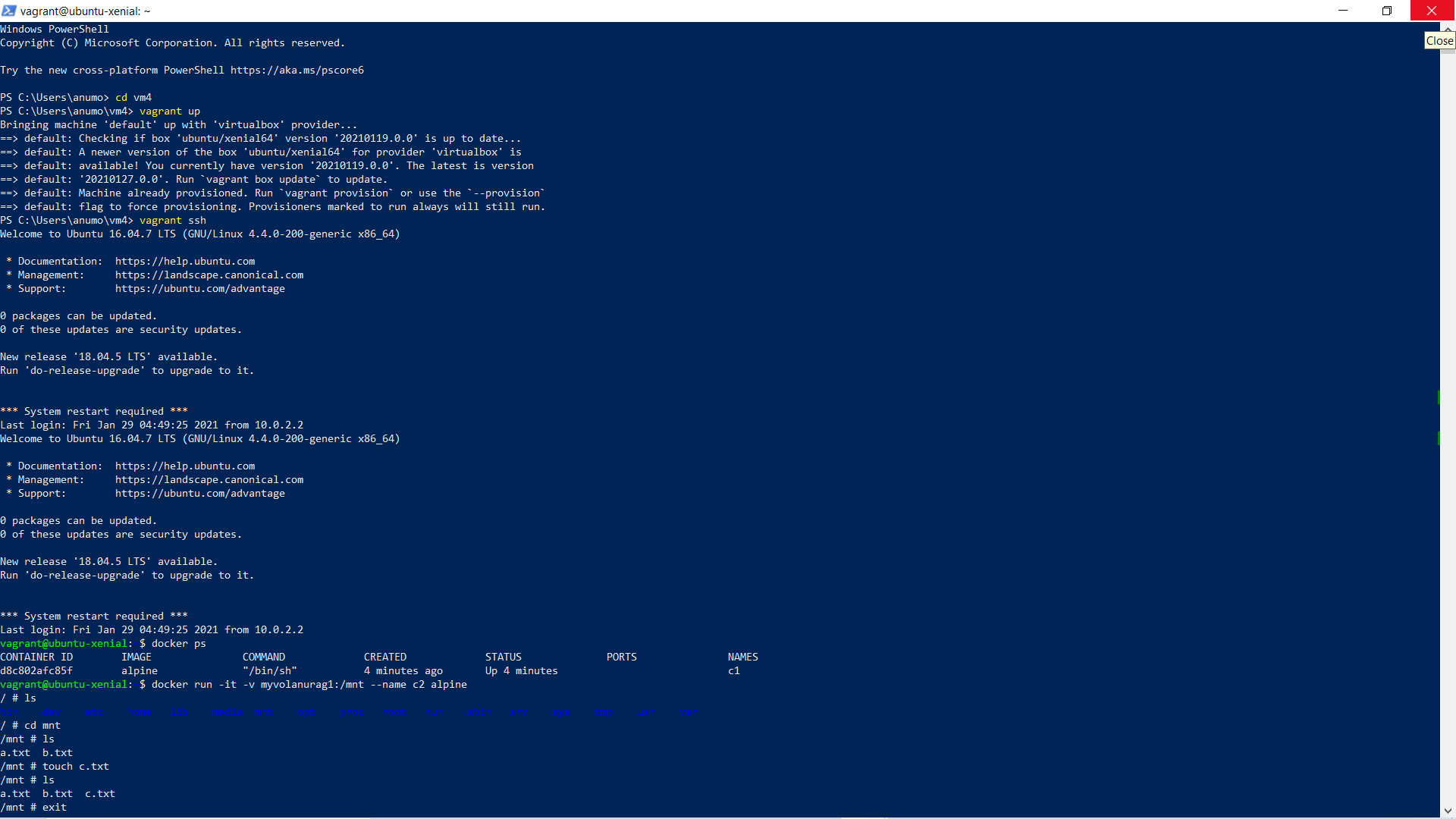
$ sudo “path location”



1. Now open a new command prompt or PowerShell and run a parallel vagrant using command: -

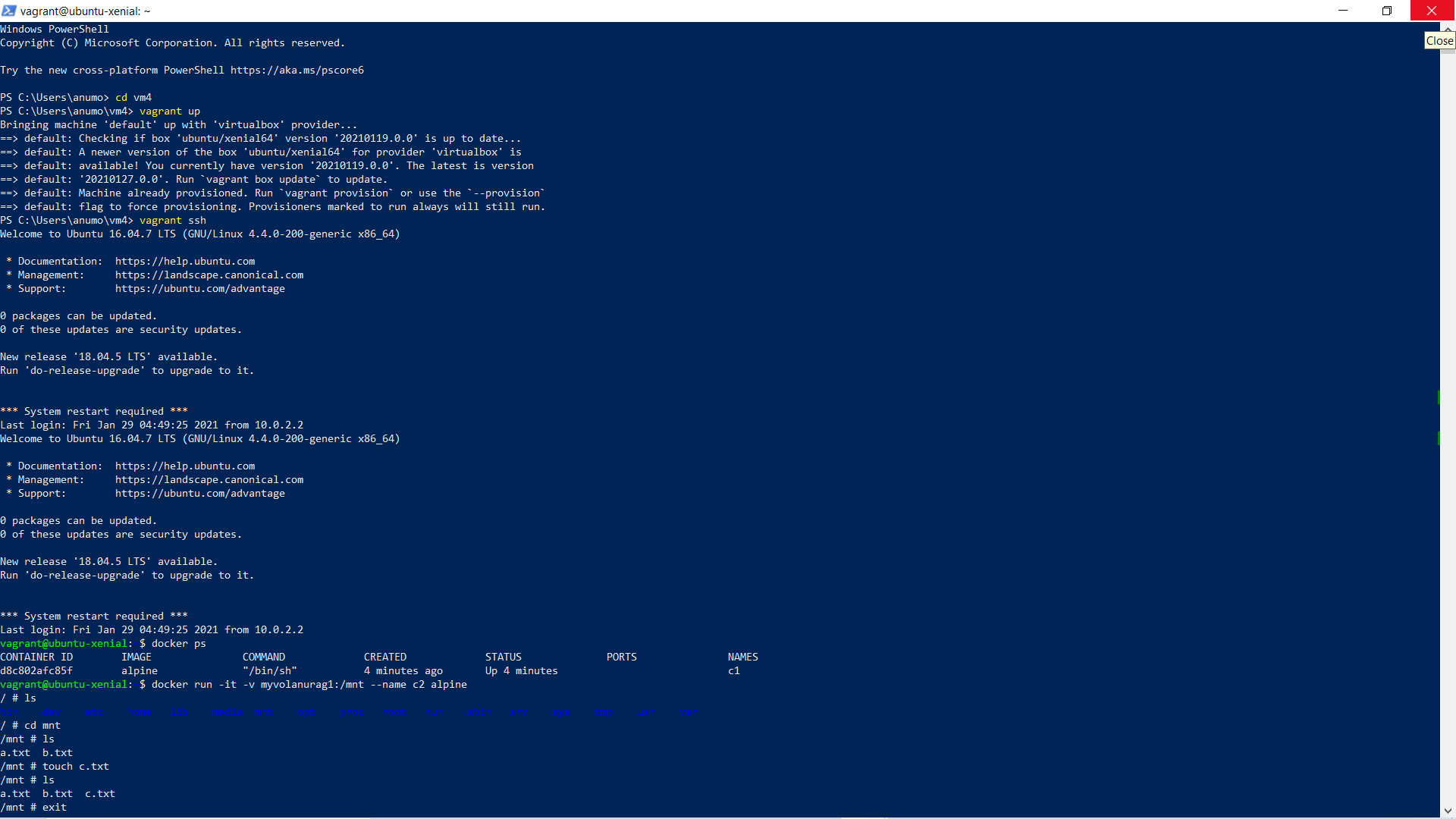
$ vagrant up

$ vagrant ssh

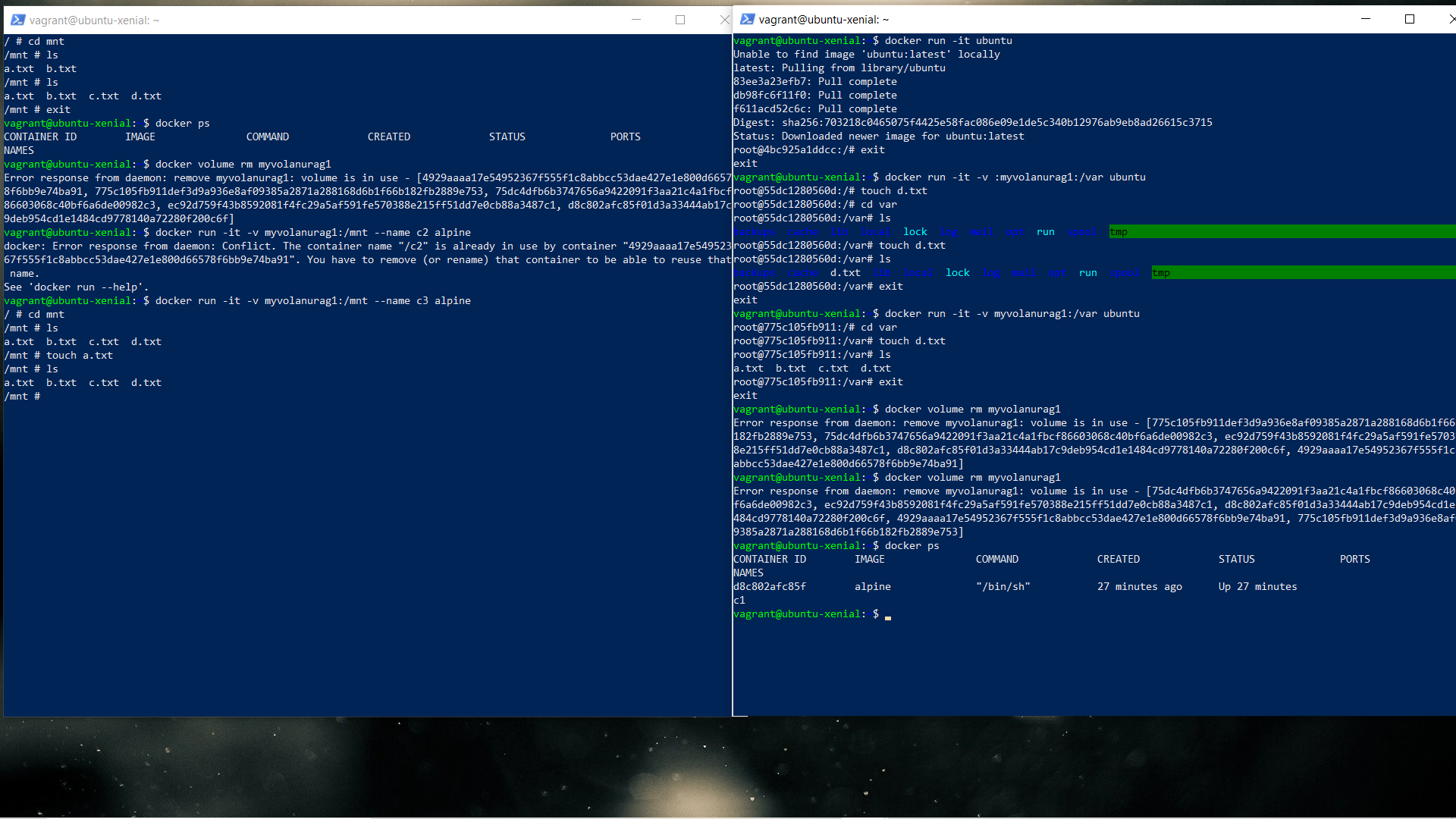


1. Check the running image in docker of the other prompt using command: -

$docker ps -a



1. Go to the alpine image at the new prompt and create the new file in the same volume that was created and now check the made file in other prompt.Both are visible.



1. Now remove the file and try to overwrite it.

