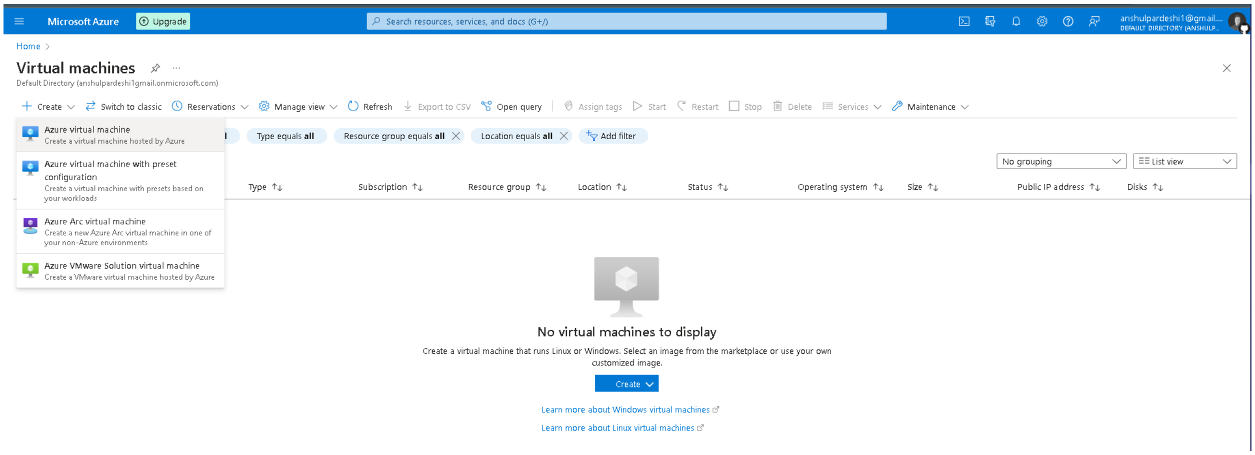
Azure-App-And-Container-Services-1

Do the following:

1.Install Docker using VM.

2. Pull HSHAR/WEBAPP (https://hub.docker.com/r/hshar/webapp) repository . Create a new file in this repository.

Let us create a VM first.
Goto Virtual MAchines from search bar and click on plus icon to create.



Virtual machines

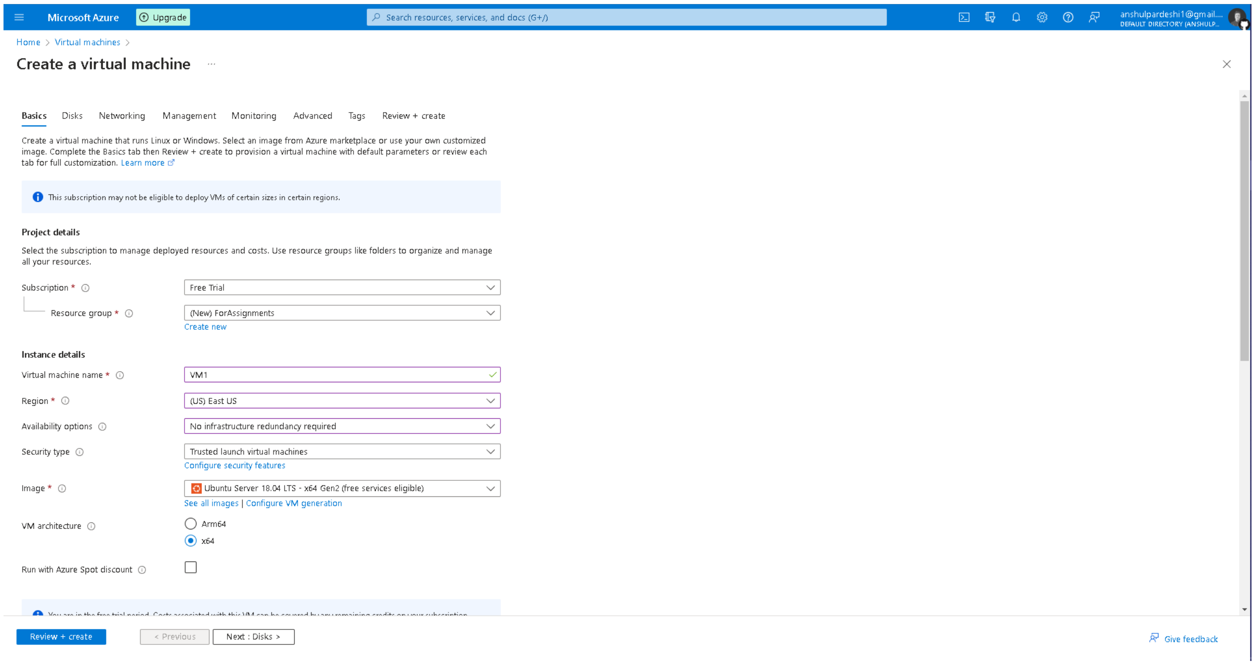
Default Directory (anshulpardeshi1gmail.onmicrosoft.com)

- Azure virtual machine — Create a virtual machine hosted by Azure
- Azure virtual machine with preset configuration — Create a virtual machine with presets based on your workloads
- Azure Arc virtual machine — Create a new Azure Arc virtual machine in one of your non-Azure environments
- Azure VMware Solution virtual machine — Create a VMware virtual machine hosted by Azure

No virtual machines to display

Create a virtual machine that runs Linux or Windows. Select an image from the marketplace or use your own customized image.

Give your VM a name. Select proper RG and we will make a VM using ubuntu OS in east us region.


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
anshulpardeshi1@gmail....
DEFAULT DIRECTORY (ANSHULP...
Home > Virtual machines >
Basics
Disks
Networking
Management
Monitoring
Advanced
Tags
Review + create
Create new
Configure security features
See all images | Configure VM generation
Review + create
< Previous
Next : Disks >
Give feedback

Create a virtual machine

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more

This subscription may not be eligible to deploy VMs of certain sizes in certain regions.

Project details

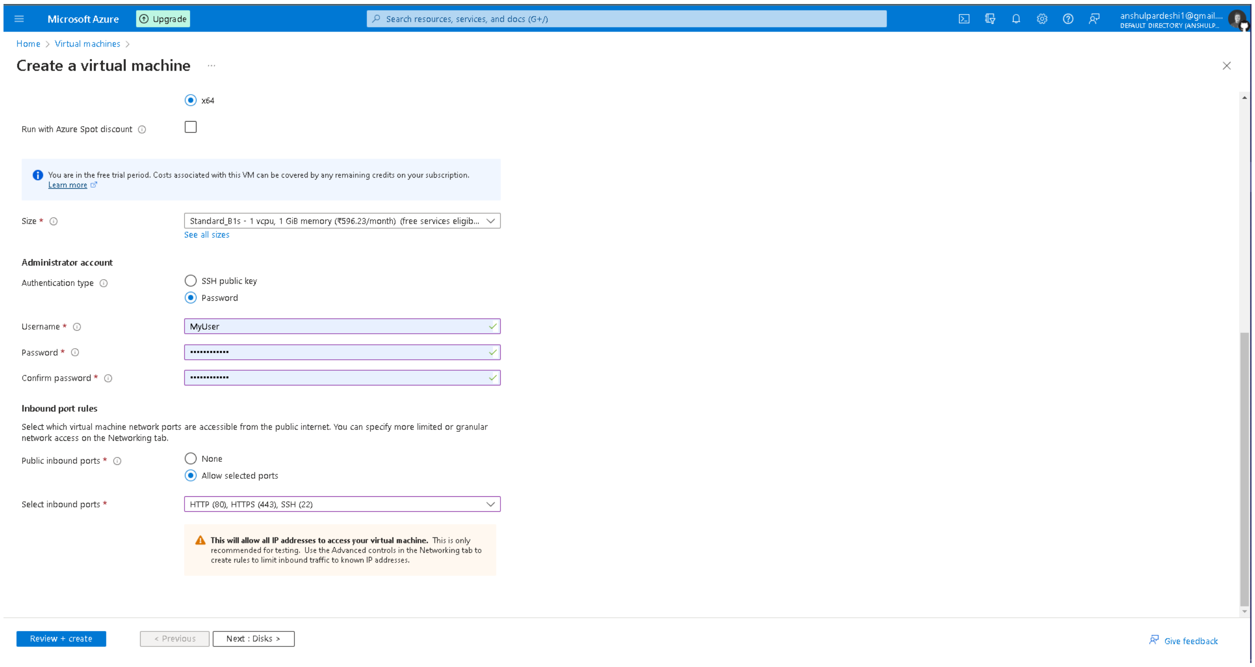
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Free Trial
Resource group *	(New) ForAssignments

Instance details

Virtual machine name *	VM1
Region *	(US) East US
Availability options	No infrastructure redundancy required
Security type	Trusted launch virtual machines
Image *	Ubuntu Server 18.04 LTS - x64 Gen2 (free services eligible)
VM architecture	○ Arm64 ● x64
Run with Azure Spot discount	□

Choose your authentication type. Open ssh port. We can open other ports later if needed.


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
anshulpardeshi1@gmail...
DEFAULT DIRECTORY (ANSHULP...
Home > Virtual machines >

Create a virtual machine

● x64

Run with Azure Spot discount ⓘ □

ⓘ You are in the free trial period. Costs associated with this VM can be covered by any remaining credits on your subscription. Learn more

Size * ⓘ Standard_B1s - 1 vcpu, 1 GiB memory (₹596.23/month) (free services eligib...
See all sizes

Administrator account

Authentication type ⓘ ○ SSH public key ● Password

Username * ⓘ MyUser ✓

Password * ⓘ •••••••••••• ✓

Confirm password * ⓘ •••••••••••• ✓

Inbound port rules

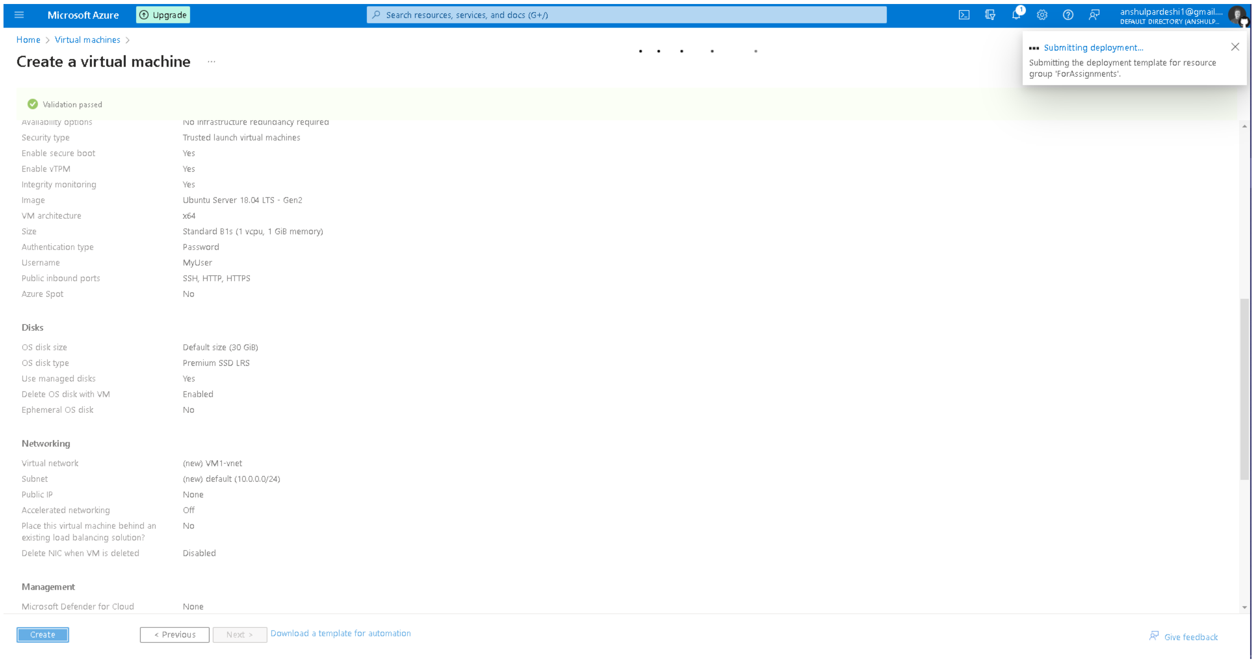
Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ ○ None ● Allow selected ports

Select inbound ports * HTTP (80), HTTPS (443), SSH (22)

⚠ **This will allow all IP addresses to access your virtual machine.** This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

Click on review and create.


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
anshulpardeshi1@gmail...
DEFAULT DIRECTORY (ANSHULP...
Home > Virtual machines >

Create a virtual machine

Submitting deployment...
Submitting the deployment template for resource group 'ForAssignments'.

✓ Validation passed

Availability options	No infrastructure redundancy required
Security type	Trusted launch virtual machines
Enable secure boot	Yes
Enable vTPM	Yes
Integrity monitoring	Yes
Image	Ubuntu Server 18.04 LTS - Gen2
VM architecture	x64
Size	Standard B1s (1 vcpu, 1 GiB memory)
Authentication type	Password
Username	MyUser
Public inbound ports	SSH, HTTP, HTTPS
Azure Spot	No

Disks

OS disk size	Default size (30 GiB)
OS disk type	Premium SSD LRS
Use managed disks	Yes
Delete OS disk with VM	Enabled
Ephemeral OS disk	No

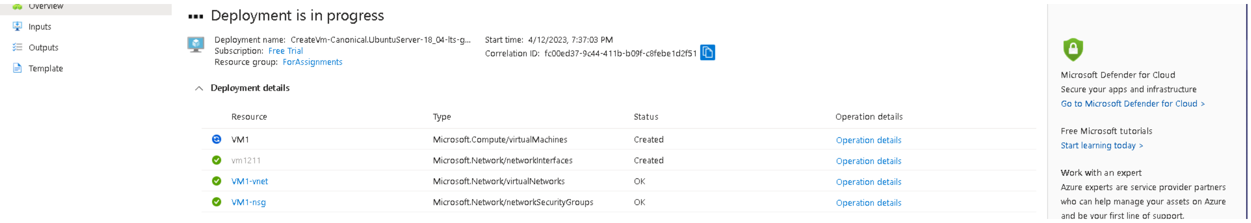
Networking

Virtual network	(new) VM1-vnet
Subnet	(new) default (10.0.0.0/24)
Public IP	None
Accelerated networking	Off
Place this virtual machine behind an existing load balancing solution?	No
Delete NIC when VM is deleted	Disabled

Management

Microsoft Defender for Cloud	None

Go to resource once deployment is done.



Deployment is in progress

Deployment name: CreateVm-Canonical.UbuntuServer-18_04-lts-g...
Subscription: Free Trial
Resource group: ForAssignments

Start time: 4/12/2023, 7:37:03 PM
Correlation ID: fc00ed37-9c44-411b-b09f-c8febe1d2f51

Deployment details

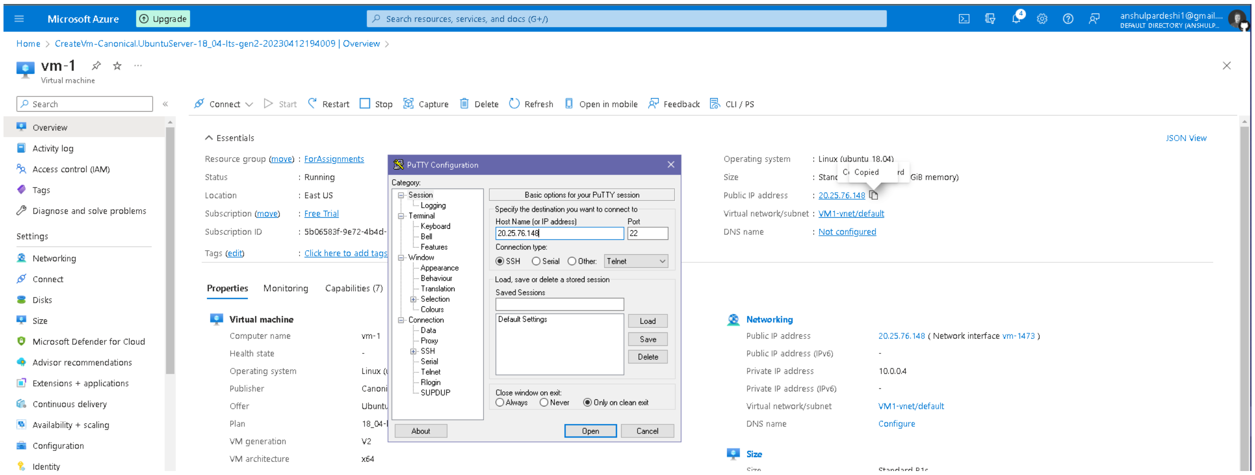
Resource	Type	Status	Operation details
VM1	Microsoft.Compute/virtualMachines	Created	Operation details
vm1211	Microsoft.Network/networkInterfaces	Created	Operation details
VM1-vnet	Microsoft.Network/virtualNetworks	OK	Operation details
VM1-nsg	Microsoft.Network/networkSecurityGroups	OK	Operation details

Microsoft Defender for Cloud
Secure your apps and infrastructure
Go to Microsoft Defender for Cloud >

Free Microsoft tutorials
Start learning today >

Work with an expert
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.

Let us ssh into VM. Copy public ip and paste it in PUTTY. YOu can use any other method to ssh if you want.



Resource group (move) : ForAssignments
Status : Running
Location : East US
Subscription (move) : Free Trial
Subscription ID : 5b06583f-9e72-4b4d-
Tags (edit) : Click here to add tags

Operating system : Linux (ubuntu 18.04)
Size : Stanc … GiB memory)
Public IP address : 20.25.76.148
Virtual network/subnet : VM1-vnet/default
DNS name : Not configured

Virtual machine
Computer name vm-1
Health state -
Operating system Linux (
Publisher Canoni
Offer Ubuntu
Plan 18_04-l
VM generation V2
VM architecture x64

Specify the destination you want to connect to
Host Name (or IP address) 20.25.76.148
Port 22
Connection type: ● SSH ○ Serial ○ Other: Telnet
Load, save or delete a stored session
Saved Sessions
Default Settings
Close window on exit: ○ Always ○ Never ● Only on clean exit

Networking
Public IP address 20.25.76.148 (Network interface vm-1473)
Public IP address (IPv6) -
Private IP address 10.0.0.4
Private IP address (IPv6) -
Virtual network/subnet VM1-vnet/default
DNS name Configure

Size
Size Standard B1s

Update the VM first using: sudo apt-get update -y
Once updated let us install docker.

Steps to install Docker:

1.sudo apt-get update

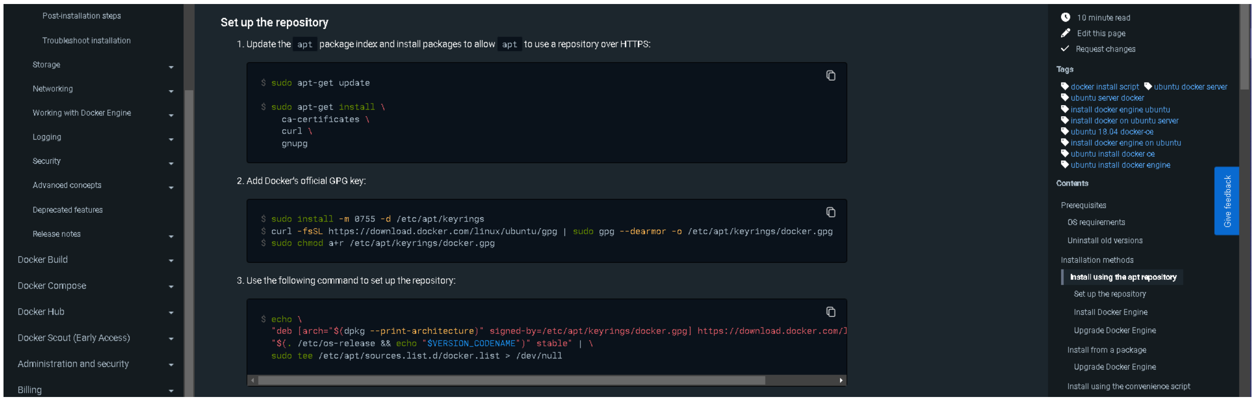
```
   sudo apt-get install \
     ca-certificates \
     curl \
     gnupg
```

2. sudo install -m 0755 -d /etc/apt/keyrings

```
    curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
     sudo chmod a+r /etc/apt/keyrings/docker.gpg
```

3. echo \

```
  "deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/ubuntu \
  "$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```



Set up the repository

1. Update the `apt` package index and install packages to allow `apt` to use a repository over HTTPS:

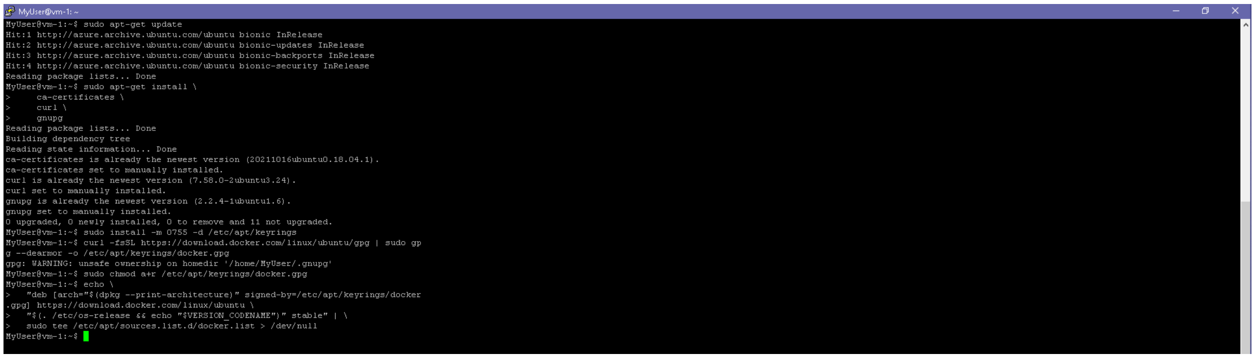
```
$ sudo apt-get update

$ sudo apt-get install \
    ca-certificates \
    curl \
    gnupg
```

2. Add Docker's official GPG key:

```
$ sudo install -m 0755 -d /etc/apt/keyrings
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
$ sudo chmod a+r /etc/apt/keyrings/docker.gpg
```

3. Use the following command to set up the repository:

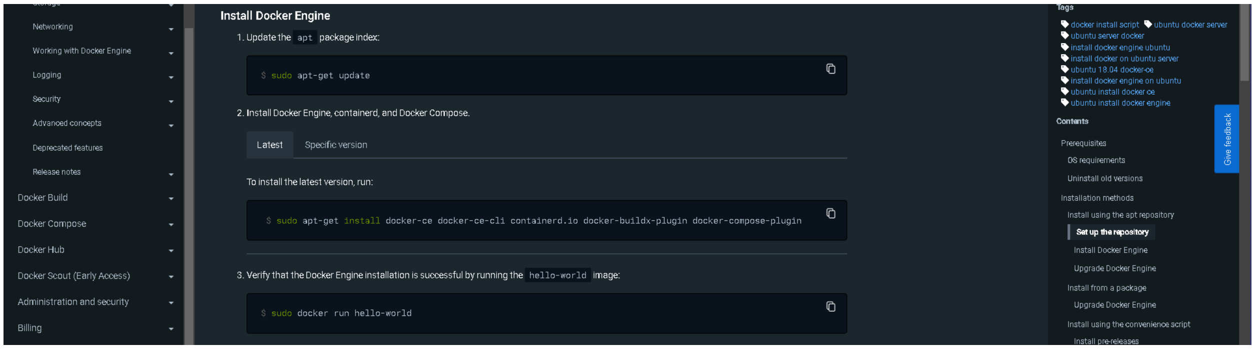
```
$ echo \
  "deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/l
  "$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```



```
MyUser@vm-1:~$ sudo apt-get update
Hit:1 http://azure.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
MyUser@vm-1:~$ sudo apt-get install \
>     ca-certificates \
>     curl \
>     gnupg
Reading package lists... Done
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20211016ubuntu0.18.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.58.0-2ubuntu3.24).
curl set to manually installed.
gnupg is already the newest version (2.2.4-1ubuntu1.6).
gnupg set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.
MyUser@vm-1:~$ sudo install -m 0755 -d /etc/apt/keyrings
MyUser@vm-1:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gp
g --dearmor -o /etc/apt/keyrings/docker.gpg
gpg: WARNING: unsafe ownership on homedir '/home/MyUser/.gnupg'
MyUser@vm-1:~$ sudo chmod a+r /etc/apt/keyrings/docker.gpg
MyUser@vm-1:~$ echo \
>   "deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker
.gpg] https://download.docker.com/linux/ubuntu \
>   "$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | \
>   sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
MyUser@vm-1:~$
```

Next:

1. **sudo apt-get update**
2. **sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin**
3. **sudo docker run hello-world**



Install Docker Engine

1. Update the `apt` package index:

```
$ sudo apt-get update
```

2. Install Docker Engine, containerd, and Docker Compose.

Latest | Specific version

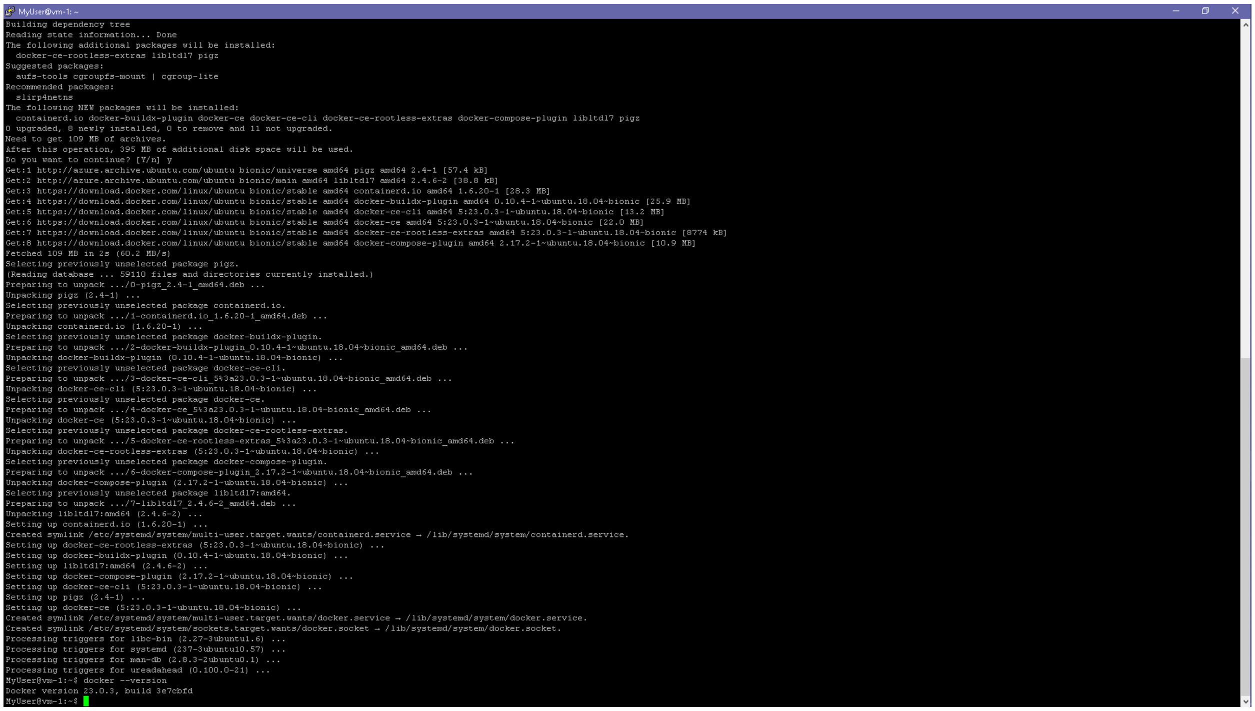
To install the latest version, run:

```
$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

3. Verify that the Docker Engine installation is successful by running the `hello-world` image:

```
$ sudo docker run hello-world
```

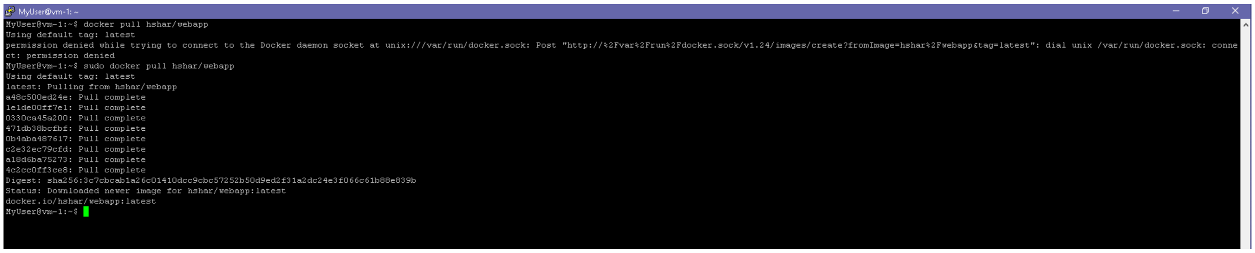
Docker is installed



```
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  docker-ce-rootless-extras libltdl7 pigz
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
Recommended packages:
  slirp4netns
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin libltdl7 pigz
0 upgraded, 8 newly installed, 0 to remove and 11 not upgraded.
Need to get 109 MB of archives.
After this operation, 395 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://azure.archive.ubuntu.com/ubuntu bionic/universe amd64 pigz amd64 2.4-1 [57.4 kB]
Get:2 http://azure.archive.ubuntu.com/ubuntu bionic/main amd64 libltdl7 amd64 2.4.6-2 [38.8 kB]
Get:3 https://download.docker.com/linux/ubuntu bionic/stable amd64 containerd.io amd64 1.6.20-1 [28.3 MB]
Get:4 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-buildx-plugin amd64 0.10.4-1~ubuntu.18.04~bionic [25.9 MB]
Get:5 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce-cli amd64 5:23.0.3-1~ubuntu.18.04~bionic [13.2 MB]
Get:6 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce amd64 5:23.0.3-1~ubuntu.18.04~bionic [22.0 MB]
Get:7 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce-rootless-extras amd64 5:23.0.3-1~ubuntu.18.04~bionic [8774 kB]
Get:8 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-compose-plugin amd64 2.17.2-1~ubuntu.18.04~bionic [10.9 MB]
Fetched 109 MB in 2s (60.2 MB/s)
Selecting previously unselected package pigz.
(Reading database ... 59110 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.4-1_amd64.deb ...
Unpacking pigz (2.4-1) ...
Selecting previously unselected package containerd.io.
Preparing to unpack .../1-containerd.io_1.6.20-1_amd64.deb ...
Unpacking containerd.io (1.6.20-1) ...
Selecting previously unselected package docker-buildx-plugin.
Preparing to unpack .../2-docker-buildx-plugin_0.10.4-1~ubuntu.18.04~bionic_amd64.deb ...
Unpacking docker-buildx-plugin (0.10.4-1~ubuntu.18.04~bionic) ...
Selecting previously unselected package docker-ce-cli.
Preparing to unpack .../3-docker-ce-cli_5%3a23.0.3-1~ubuntu.18.04~bionic_amd64.deb ...
Unpacking docker-ce-cli (5:23.0.3-1~ubuntu.18.04~bionic) ...
Selecting previously unselected package docker-ce.
Preparing to unpack .../4-docker-ce_5%3a23.0.3-1~ubuntu.18.04~bionic_amd64.deb ...
Unpacking docker-ce (5:23.0.3-1~ubuntu.18.04~bionic) ...
Selecting previously unselected package docker-ce-rootless-extras.
Preparing to unpack .../5-docker-ce-rootless-extras_5%3a23.0.3-1~ubuntu.18.04~bionic_amd64.deb ...
Unpacking docker-ce-rootless-extras (5:23.0.3-1~ubuntu.18.04~bionic) ...
Selecting previously unselected package docker-compose-plugin.
Preparing to unpack .../6-docker-compose-plugin_2.17.2-1~ubuntu.18.04~bionic_amd64.deb ...
Unpacking docker-compose-plugin (2.17.2-1~ubuntu.18.04~bionic) ...
Selecting previously unselected package libltdl7:amd64.
Preparing to unpack .../7-libltdl7_2.4.6-2_amd64.deb ...
Unpacking libltdl7:amd64 (2.4.6-2) ...
Setting up containerd.io (1.6.20-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up docker-ce-rootless-extras (5:23.0.3-1~ubuntu.18.04~bionic) ...
Setting up docker-buildx-plugin (0.10.4-1~ubuntu.18.04~bionic) ...
Setting up libltdl7:amd64 (2.4.6-2) ...
Setting up docker-compose-plugin (2.17.2-1~ubuntu.18.04~bionic) ...
Setting up docker-ce-cli (5:23.0.3-1~ubuntu.18.04~bionic) ...
Setting up pigz (2.4-1) ...
Setting up docker-ce (5:23.0.3-1~ubuntu.18.04~bionic) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for libc-bin (2.27-3ubuntu1.6) ...
Processing triggers for systemd (237-3ubuntu10.57) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
MyUser@vm-1:~$ docker --version
Docker version 23.0.3, build 3e7cbfd
MyUser@vm-1:~$
```

Let us pull hshar/webapp repo now.

For that simply use command: docker pull hshar/webapp



```
MyUser@vm-1:~$ docker pull hshar/webapp
Using default tag: latest
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Post "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/images/create?fromImage=hshar%2Fwebapp&tag=latest": dial unix /var/run/docker.sock: conne
ct: permission denied
MyUser@vm-1:~$ sudo docker pull hshar/webapp
Using default tag: latest
latest: Pulling from hshar/webapp
a48c500ed24e: Pull complete
1e1de00ff7e1: Pull complete
0330ca45a200: Pull complete
471db38bcfbf: Pull complete
0b4aba487617: Pull complete
c2e32ec79cfd: Pull complete
a18d6ba75273: Pull complete
4c2cc0ff3ce8: Pull complete
Digest: sha256:3c7cbcab1a26c01410dcc9cbc57252b50d9ed2f31a2dc24e3f066c61b88e839b
Status: Downloaded newer image for hshar/webapp:latest
docker.io/hshar/webapp:latest
MyUser@vm-1:~$
```

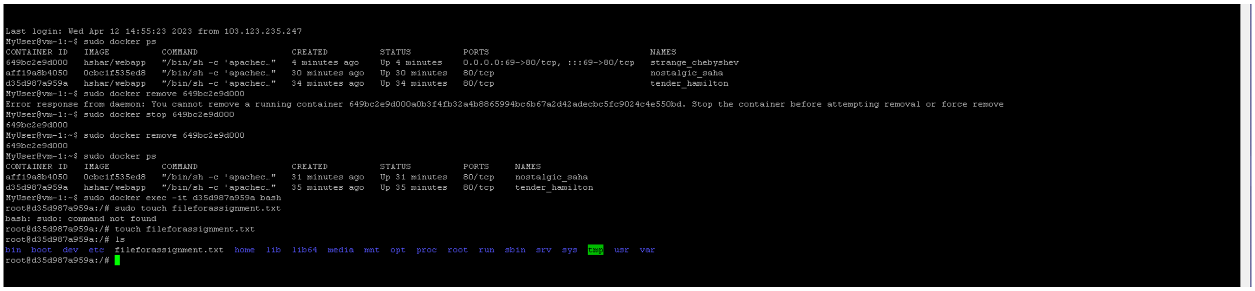
Now let us create a new file in this repo.

For that first: sudo docker ps

Copy the container id of hshar/webapp image.

Then: sudo docker exec -it <container id> bash

Then create a file: touch <file name>



```
Last login: Wed Apr 12 14:55:23 2023 from 103.123.235.247
MyUser@vm-1:~$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS                               NAMES
649bc2e9d000   hshar/webapp   "/bin/sh -c 'apachec…"   4 minutes ago    Up 4 minutes    0.0.0.0:69->80/tcp, :::69->80/tcp   strange_chebyshev
aff19a8b4050   0cbc1f535ed8   "/bin/sh -c 'apachec…"   30 minutes ago   Up 30 minutes   80/tcp                              nostalgic_saha
d35d987a959a   hshar/webapp   "/bin/sh -c 'apachec…"   34 minutes ago   Up 34 minutes   80/tcp                              tender_hamilton
MyUser@vm-1:~$ sudo docker remove 649bc2e9d000
Error response from daemon: You cannot remove a running container 649bc2e9d000a0b3f4fb32a4b8865994bc6b67a2d42adecbc5fc9024c4e550bd. Stop the container before attempting removal or force remove
MyUser@vm-1:~$ sudo docker stop 649bc2e9d000
649bc2e9d000
MyUser@vm-1:~$ sudo docker remove 649bc2e9d000
649bc2e9d000
MyUser@vm-1:~$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS     NAMES
aff19a8b4050   0cbc1f535ed8   "/bin/sh -c 'apachec…"   31 minutes ago   Up 31 minutes   80/tcp    nostalgic_saha
d35d987a959a   hshar/webapp   "/bin/sh -c 'apachec…"   35 minutes ago   Up 35 minutes   80/tcp    tender_hamilton
MyUser@vm-1:~$ sudo docker exec -it d35d987a959a bash
root@d35d987a959a:/# sudo touch fileforassignment.txt
bash: sudo: command not found
root@d35d987a959a:/# touch fileforassignment.txt
root@d35d987a959a:/# ls
bin  boot  dev  etc  fileforassignment.txt  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@d35d987a959a:/#
```