

ASSIGNMENT IV

Microsoft Azure Administrator

Neme Aduh

How to Add a Data Disk to a Rocky LinuxVM using after the VM has been deployed

- Create a Rocky LVM and deploy
- Goto the VM deployed> click on settings>Disk
- Under Data disks - Click ‘Create and attach a new disk’

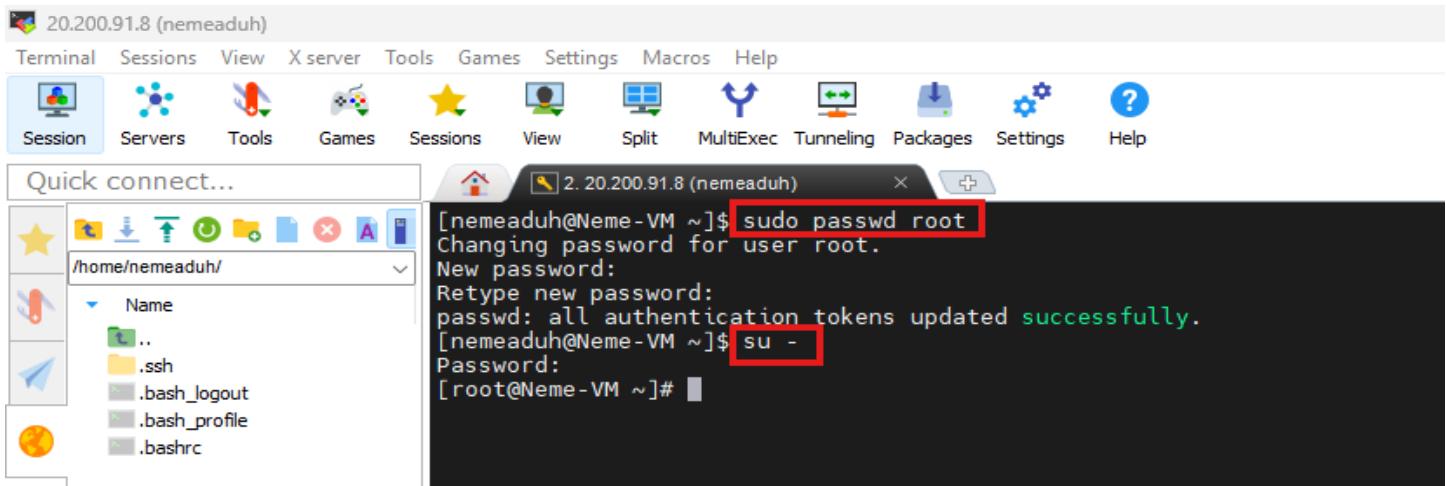
The screenshot shows the Microsoft Azure portal interface for managing a virtual machine named 'Neme-VM'. The left sidebar shows various navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, and Settings. The 'Disks' option under Settings is highlighted with a red box. The main content area shows the 'OS disk' (Neme-VM_OsDisk_1) and the 'Data disks' section. The 'Data disks' section has a 'Filter by name' input field and a link to 'Attach existing disks'. Below these, there is a button labeled '+ Create and attach a new disk' which is also highlighted with a red box.

- Define the Disk name, storage type, size etc then hit ‘Apply’ to deploy the newly created disk

This screenshot shows the 'Data disks' configuration page for the 'Neme-VM'. It displays a table with columns for LUN, Disk name, Storage type, Size (GiB), Max IOPS, Max throughput, Encryption, and Host caching. A new disk entry for 'Neme-Datadisk' is listed, with its details filled in: Storage type is Premium SSD LRS, Size is 64 GiB, Max IOPS is 240, Max throughput is 50, Encryption is SSE with PMK, and Host caching is set to 'Read-only'. The 'Host caching' dropdown is highlighted with a red box.

Next Step is to connect to the VM using MobaXterm and Initialize the Disk

- Connect to the LVM using MobaXterm
- For me to become the root user that is the super user to be able to run admin commands without using sudo, I used the command ‘**sudo passwd root**’ to change the password then I switch user by running command ‘**su -**’



- To show the Disk usage, run a command ‘df -h’
- Sda2 is the OS disk
- Sdb1 is the Temporary disk
- So we currently do not have the Data disk showing yet

```
[root@Neme-VM ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        3.8G   0    3.8G  0% /dev
tmpfs          3.8G   0    3.8G  0% /dev/shm
tmpfs          3.8G  17M  3.8G  1% /run
tmpfs          3.8G   0    3.8G  0% /sys/fs/cgroup
/dev/mapper/rocky-root  9.0G  1.4G  7.6G  16% /
/dev/sda2       994M 235M  760M 24% /boot
/dev/sda1        99M  5.8M  94M  6% /boot/efi
/dev/sdb1        16G  28K   15G  1% /mnt
tmpfs          769M   0   769M  0% /run/user/1000
```

- But, typing the command ‘fdisk -l’ shows the Data Disk which is sdc but it has not been partitioned

```
[root@Neme-VM ~]# fdisk -l
Disk /dev/sdb: 16 GiB, 17179869184 bytes, 33554432 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: dos
Disk identifier: 0x01fed12e

Device      Start     End   Sectors  Size Type
/dev/sdb1    2048 33552383 33550336  16G 7 HPFS/NTFS/exFAT

Disk /dev/sda: 10 GiB, 10737418240 bytes, 20971520 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: gpt
Disk identifier: 1853C561-0084-4B2A-B8F9-9C97DE4DC8E4

Device      Start     End   Sectors  Size Type
/dev/sda1    2048  204799  202752   99M EFI System
/dev/sda2   204800 2252799 2048000 1000M Linux filesystem
/dev/sda3   2252800 2260991   8192    4M PowerPC PreP boot
/dev/sda4   2260992 2263039    2048    1M BIOS boot
/dev/sda5   2265088 20969471 18704384  8.9G Linux LVM

Disk /dev/mapper/rocky-root: 8.9 GiB, 9575596032 bytes, 18702336 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/sdc: 64 GiB, 68719476736 bytes, 134217728 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
[root@Neme-VM ~]#
```

- Since the name of the DataDisk as seen is ‘dev/sdc/ , we will now go into the disk using command ‘fdisk /dev/sdc/’
- Type m for help
- So we want to type ‘n’ to add a new partition

```
[root@Neme-VM ~]# fdisk /dev/sdc
Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xb91d158a.

Command (m for help): m
Help:

DOS (MBR)
a    toggle a bootable flag
b    edit nested BSD disklabel
c    toggle the dos compatibility flag

Generic
d    delete a partition
F    list free unpartitioned space
l    list known partition types
n    add a new partition
p    print the partition table
t    change a partition type
v    verify the partition table
i    print information about a partition

Misc
m    print this menu
u    change display/entry units
x    extra functionality (experts only)

Script
I    load disk layout from sfdisk script file
O    dump disk layout to sfdisk script file

Save & Exit
w    write table to disk and exit
q    quit without saving changes

Create a new label
g    create a new empty GPT partition table
G    create a new empty SGI (IRIX) partition table
o    create a new empty DOS partition table
s    create a new empty Sun partition table
```

- In this first partition, we have created 20GB out of the 64GB Data

```
Command (m for help): n
Partition type
 p primary (0 primary, 0 extended, 4 free)
 e extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (1-4, default 1):
First sector (2048-134217727, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-134217727, default 134217727): +20G
Created a new partition 1 of type 'Linux' and of size 20 GiB.
```

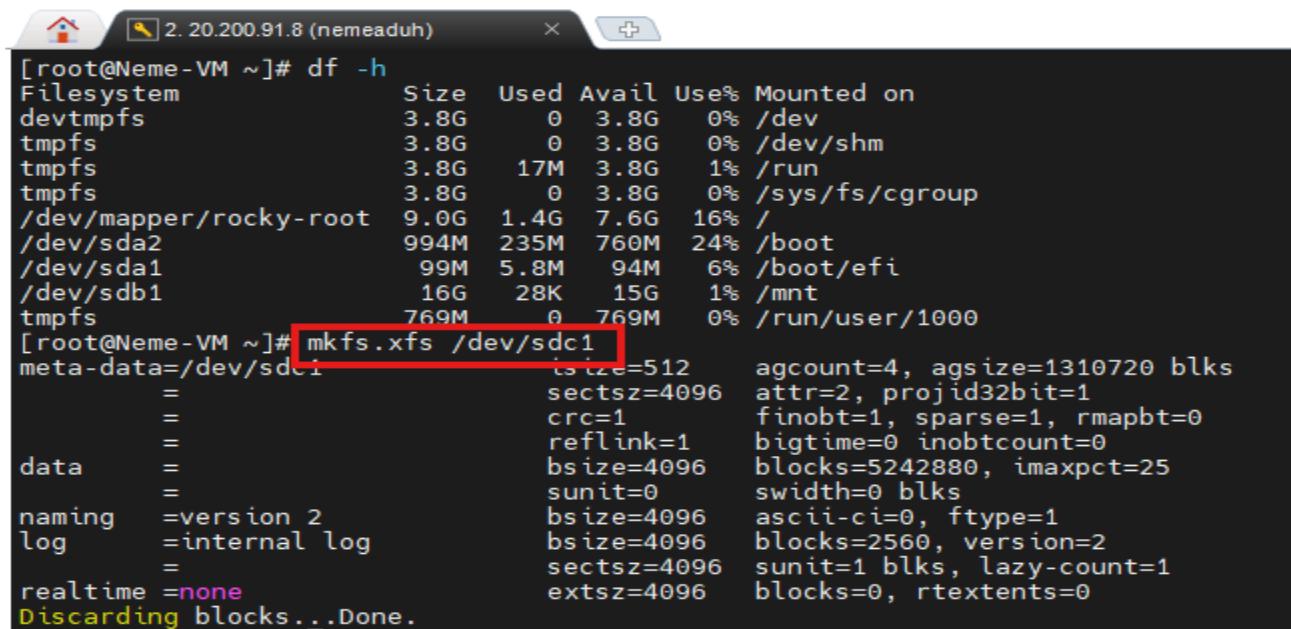
- To save this partition, run '**w**' and enter, The partition table been altered means is now saved

```
Command (m for help): p
Disk /dev/sdc: 64 GiB, 68719476736 bytes, 134217728 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: dos
Disk identifier: 0x3755b223

Device      Boot Start      End  Sectors Size Id Type
/dev/sdc1        2048 41945087 41943040  20G  0 Empty

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

- Next Step, In as much this sdc has been partitioned, if you run a command '**df -h**' you won't still see it which means it is yet to be 'cooked'
- So to cook this we run a command '**mkfs.xfs /dev/sdc1**



```
[root@Neme-VM ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        3.8G   0    3.8G  0% /dev
tmpfs          3.8G   0    3.8G  0% /dev/shm
tmpfs          3.8G  17M  3.8G  1% /run
tmpfs          3.8G   0    3.8G  0% /sys/fs/cgroup
/dev/mapper/rocky-root  9.0G  1.4G  7.6G  16% /
/dev/sda2       994M 235M  760M 24% /boot
/dev/sda1       99M  5.8M  94M  6% /boot/efi
/dev/sdb1       16G  28K  15G  1% /mnt
tmpfs          769M   0  769M  0% /run/user/1000
[root@Neme-VM ~]# mkfs.xfs /dev/sdc1
meta-data=/dev/sdc1              isize=512    agcount=4, agsize=1310720 blks
                                =                      sectsz=4096  attr=2, projid32bit=1
                                =                      crc=1     finobt=1, sparse=1, rmapbt=0
                                =                      reflink=1  bigtime=0 inobtcount=0
data             =              bszie=4096   blocks=5242880, imaxpct=25
                                =                      sunit=0    swidth=0 blks
naming           =version 2    bsize=4096   ascii-ci=0, ftype=1
log              =internal log  bsize=4096   blocks=2560, version=2
                                =                      sectsz=4096  sunit=1 blks, lazy-count=1
realtime         =none          bsize=4096   extsz=4096  blocks=0, rtextents=0
Discarding blocks...Done.
```

- In as much as this has been cooked, we still need to mount the partition data disk for it to display
- So we are going to make a directory where we want to mount the data disk by running the command ‘**mkdir /nemeData**’
- Once a directory has been made, we can then run the command ‘**mount /dev/sdc1 /nemeData**/’

[root@Neme-VM /]# mount /dev/sdc1 /nemeData/

- To confirm the mount has been done, run ‘**df -h**’

```
[root@Neme-VM /]# df -h
Filesystem           Size  Used Avail Use% Mounted on
devtmpfs              3.8G   0    3.8G  0% /dev
tmpfs                 3.8G   0    3.8G  0% /dev/shm
tmpfs                 3.8G  17M  3.8G  1% /run
tmpfs                 3.8G   0    3.8G  0% /sys/fs/cgroup
/dev/mapper/rocky-root 9.0G  1.4G  7.6G  16% /
/dev/sda2             994M 235M  760M  24% /boot
/dev/sda1             99M  5.8M  94M  6% /boot/efi
/dev/sdb1             16G  28K   15G  1% /mnt
tmpfs                760M   0    760M  0% /run/user/1000
/dev/sdc1              20G  176M  20G  1% /nemeData
[root@Neme-VM /]#
```

- But this mount has not been saved permanently, so to do this, we run the command ‘**echo “/dev/sdc1 /nemeData xfs defaults 0 0” >> /etc/fstab**’
- To confirm this is now mounted by default/permanently, run the command ‘**cat /etc/fstab**’

Validation check: **/dev/sdc1 /nemeData xfs defaults 0 0** displays which shows this is now mounted by defaulted into fstab

```
[root@Neme-VM /]# echo "/dev/sdc1 /nemeData xfs defaults 0 0" >> /etc/fstab
[root@Neme-VM /]# df -h
Filesystem           Size  Used Avail Use% Mounted on
devtmpfs              3.8G   0    3.8G  0% /dev
tmpfs                 3.8G   0    3.8G  0% /dev/shm
tmpfs                 3.8G  17M  3.8G  1% /run
tmpfs                 3.8G   0    3.8G  0% /sys/fs/cgroup
/dev/mapper/rocky-root 9.0G  1.4G  7.6G  16% /
/dev/sda2             994M 235M  760M  24% /boot
/dev/sda1             99M  5.8M  94M  6% /boot/efi
/dev/sdb1             16G  28K   15G  1% /mnt
tmpfs                769M   0    769M  0% /run/user/1000
/dev/sdc1              20G  176M  20G  1% /nemeData
[root@Neme-VM /]# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Sun Nov 19 14:47:37 2023
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/rocky-root /          xfs      defaults        0 0
UUID=13ced322-b2af-49d5-a9de-49ade19ac569 /boot      xfs      defaults        0 0
UUID=006E-25E5      /boot/efi     vfat    defaults,uid=0,gid=0,umask=077,shortname=winnt 0 2
/dev/disk/by-label/azure_resource-part1  /mnt      auto    defaults,nofail,x-systemd.requires=cloud-init.service,_netdev,comment=cloudconfig      0      2
/dev/sdc1 /nemeData xfs defaults 0 0
```

We can unmount using command ‘**umount /nemeData**’

We can mount it back using the command ‘**mount -a**’