PPEC - Week 3
Assignment Report

Siddharth Kumar

September 3, 2025

Contents

Chapter 1

VM Disk Configuration

1.1 Virsh commands

```
1 virsh list --all
2 virsh domblklist stg-hdpsiddharth101
3 virsh domblklist stg-hdpsiddharth102
4 virsh domblklist stg-hdpsiddharth103
5 virsh domblklist stg-hdpsiddharth104
6 virsh domblklist stg-hdpsiddharth105
```

Listing 1.1: Virsh commands

Explanation
- Line 1: Runs `virsh list --all` to display all VMs managed by libvirt
(both running and shut off).
- Line 2: Runs `virsh domblklist stg-hdpsiddharth101` to list block de-
vices (disks) attached to VM `stg-hdpsiddharth101`.
- Line 3: Runs `virsh domblklist stg-hdpsiddharth102` to list block de-
vices for VM `stg-hdpsiddharth102`.
- Line 4: Runs `virsh domblklist stg-hdpsiddharth103` to list block de-
vices for VM `stg-hdpsiddharth103`.
- Line 5: Runs `virsh domblklist stg-hdpsiddharth104` to list block de-
vices for VM `stg-hdpsiddharth104`.
- Line 6: Runs `virsh domblklist stg-hdpsiddharth105` to list block de-
vices for VM `stg-hdpsiddharth105`.

1.2 Attaching disk of size 100G

```bash
#!/bin/bash

# List all available VMs
virsh list --all

echo "Enter the ID of the VM to attach the disk:"
read VM_ID

# Get the VM name from the ID
VM_NAME=$(virsh list --all | awk -v id="$VM_ID" '$1 == id {
    print $2}')

if [ -z "$VM_NAME" ]; then
    echo "Invalid VM ID. Exiting."
    exit 1
fi

echo "Selected VM: $VM_NAME"

echo "Enter the service name:"
read SERVICE

echo "Enter the disk size (e.g., 100G):"
read SIZE

echo "Enter the target device (vdc/vdd/vde/etc.):"
read DEVICE

# Construct disk image path
DISK_PATH="/opt/var/lib/libvirt/disks/1/${VM_NAME}-${SERVICE
    }-${SIZE}.img"

# Create and set permissions for the disk image
qemu-img create -f qcow2 "$DISK_PATH" "$SIZE"
chown libvirt-qemu:kvm "$DISK_PATH"
chmod 755 "$DISK_PATH"
ls -ltrh "$DISK_PATH"

# Attach the disk to the VM
virsh attach-disk --domain "$VM_NAME" "$DISK_PATH" --driver
    qemu --subdriver qcow2 --targetbus virtio --target "
    $DEVICE" --persistent --config --live
```

Listing 1.2: Script for attaching disk

Explanation

- Line 1: Declares the script interpreter as Bash (`!/bin/bash`).
- Line 3: Adds a comment describing that the script lists available VMs.
- Line 4: Runs `virsh list --all` to display all defined VMs (running and shut off).
- Line 6: Prompts the user to enter the ID of the VM to which the disk will be attached.
- Line 7: Reads the VM ID input from the user.
- Line 9: Adds a comment indicating that the script retrieves the VM name from the given ID.
- Line 10: Uses `virsh list --all` and `awk` to extract the VM name corresponding to the entered ID.
- Lines 12–15: Checks if `VM_NAME` is empty. If invalid, prints an error and exits.
- Line 17: Displays the selected VM name.
- Line 19: Prompts the user to enter a service name (identifier for the disk).
- Line 20: Reads the service name input.
- Line 22: Prompts for the new disk size (e.g., `100G`).
- Line 23: Reads the disk size input.
- Line 25: Prompts for the target device name inside the VM (e.g., `vdc`, `vdd`).
- Line 26: Reads the device name input.
- Line 28: Adds a comment indicating construction of the disk image path.
- Line 29: Builds the disk image path using VM name, service name, and size, stored under `/opt/var/lib/libvirt/disks/1/`.
- Line 31: Comment about creating the disk image file and setting proper permissions.
- Line 32: Creates a new `qcow2` disk image with `qemu-img`.
- Line 33: Sets ownership of the disk image to `libvirt-qemu:kvm`.
- Line 34: Sets permissions to `755` for the new disk file.
- Line 35: Lists details of the created disk image for verification.
- Line 37: Comment indicating attachment of the disk to the VM.
- Line 38: Uses `virsh attach-disk` to attach the created disk to the VM, with parameters:
– `domain`: Target VM name.
– `driver qemu --subdriver qcow2`: Specifies the disk type.
– `targetbus virtio`: Ensures the disk uses virtio for performance.
– `target $DEVICE`: Specifies the VM device name (e.g., vdc).
– `persistent --config --live`: Ensures the disk is attached both immediately and persistently across reboots.

1.3   Screenshots

```
[root@stg-hdpinternclouddb102.phonepe.nb6 ~]$ virsh list --all
 Id    Name                  State
----------------------------------
 6     stg-hdpsiddharth101   running
 7     stg-hdpsiddharth102   running
 8     stg-hdpsiddharth103   running
 9     stg-hdpsiddharth104   running
 10    stg-hdpsiddharth105   running

[root@stg-hdpinternclouddb102.phonepe.nb6 ~]$ virsh domblklist 6
 Target    Source
--------------------------------------------------------------------------
 vda       /opt/var/lib/libvirt/images/stg-hdpsiddharth101.img
 vdb       /opt/ppec-agent/cloudinits/stg-hdpsiddharth101.img
 vdc       /opt/var/lib/libvirt/disks/1/stg-hdpsiddharth101-log-100G.img

[root@stg-hdpinternclouddb102.phonepe.nb6 ~]$ virsh domblklist 7
 Target    Source
--------------------------------------------------------------------------
 vda       /opt/var/lib/libvirt/images/stg-hdpsiddharth102.img
 vdb       /opt/ppec-agent/cloudinits/stg-hdpsiddharth102.img
 vdc       /opt/var/lib/libvirt/disks/1/stg-hdpsiddharth102-log-100G.img

[root@stg-hdpinternclouddb102.phonepe.nb6 ~]$ virsh domblklist 8
 Target    Source
--------------------------------------------------------------------------
 vda       /opt/var/lib/libvirt/images/stg-hdpsiddharth103.img
 vdb       /opt/ppec-agent/cloudinits/stg-hdpsiddharth103.img
 vdc       /opt/var/lib/libvirt/disks/1/stg-hdpsiddharth103-log-100G.img

[root@stg-hdpinternclouddb102.phonepe.nb6 ~]$ virsh domblklist 9
 Target    Source
--------------------------------------------------------------------------
 vda       /opt/var/lib/libvirt/images/stg-hdpsiddharth104.img
 vdb       /opt/ppec-agent/cloudinits/stg-hdpsiddharth104.img
 vdc       /opt/var/lib/libvirt/disks/1/stg-hdpsiddharth104-log-100G.img

[root@stg-hdpinternclouddb102.phonepe.nb6 ~]$ virsh domblklist 10
 Target    Source
--------------------------------------------------------------------------
 vda       /opt/var/lib/libvirt/images/stg-hdpsiddharth105.img
 vdb       /opt/ppec-agent/cloudinits/stg-hdpsiddharth105.img
 vdc       /opt/var/lib/libvirt/disks/1/stg-hdpsiddharth105-log-100G.img
```

Figure 1.1: VM's Stautus and block devices attached to them

Chapter 2

VM Resource Modification

2.1 Editing XML file

```
1 #Take backup
2 virsh dumpxml stg-hdpsiddharth105 > /home/sre/stg-
    hdpsiddharth105-bkp
3
4 #shutdown vm
5 virsh shutdown <VM>
6
7 #edit this part
8 virsh edit stg-hdpsiddharth105
9 <memory unit='KiB'>33554432</memory>
10 <currentMemory unit='KiB'>33554432</currentMemory>
11 <vcpu placement='static'>8</vcpu>
12
13
14 #Start vm
15 virsh start <VM>
```

Listing 2.1: Steps for resource modification

Explanation
- Step 1: First, shut down the VM gracefully using `virsh shutdown <vm-name>`
to ensure no data corruption.
- Step 2: Take a backup of the current VM XML definition with `virsh
dumpxml <vm-name> > vm-backup` for recovery if needed.
- Step 3: Edit the VM XML definition with `virsh edit <vm-name>` and
modify the following lines:
`<memory unit='KiB'>33554432</memory>` → Sets total memory to 32 GiB
(in KiB units).
`<currentMemory unit='KiB'>33554432</currentMemory>` → Defines the

memory allocated at runtime (also 32 GiB).

`<vcpu placement='static'>8</vcpu>` → Assigns 8 virtual CPUs to the VM with static placement.

- Step 4: Save and exit the editor to apply the changes.

- Step 5: Restart the VM with `virsh start <vm-name>` so the new CPU and memory settings take effect.

2.2 Screenshots

```
[sre@stg-hdpsiddharth101:~$ free -g
              total        used        free      shared  buff/cache   available
Mem:             31           0          29           0           1          30
Swap:             0           0           0
[sre@stg-hdpsiddharth101:~$ nproc
8
[sre@stg-hdpsiddharth101:~$ lsblk
NAME     MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
vda      252:0    0    50G  0 disk
├─vda1   252:1    0  49.9G  0 part /
├─vda14  252:14   0     4M  0 part
└─vda15  252:15   0   106M  0 part /boot/efi
vdb      252:16   0   368K  0 disk
vdc      252:32   0   100G  0 disk
sre@stg-hdpsiddharth101:~$
```

Figure 2.1: stg-hdpsiddharth101 Status

```
sre@stg-hdpsiddharth102:~$ free -g
              total        used        free      shared  buff/cache   available
Mem:             31           0          30           0           0          30
Swap:             0           0           0
sre@stg-hdpsiddharth102:~$ nproc
8
sre@stg-hdpsiddharth102:~$ lsblk
NAME     MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
vda      252:0    0    50G  0 disk
├─vda1   252:1    0  49.9G  0 part /
├─vda14  252:14   0     4M  0 part
└─vda15  252:15   0   106M  0 part /boot/efi
vdb      252:16   0   368K  0 disk
vdc      252:32   0   100G  0 disk
sre@stg-hdpsiddharth102:~$
```

Figure 2.2: stg-hdpsiddharth102 Status

```
[sre@stg-hdpsiddharth103:~$ nproc
8
[sre@stg-hdpsiddharth103:~$ free -g
              total        used        free      shared  buff/cache   available
Mem:             31           0          30           0           0          30
Swap:             0           0           0
[sre@stg-hdpsiddharth103:~$ nproc
8
[sre@stg-hdpsiddharth103:~$ lsblk
NAME    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
vda       252:0    0    50G  0 disk
├─vda1    252:1    0  49.9G  0 part /
├─vda14   252:14   0     4M  0 part
└─vda15   252:15   0   106M  0 part /boot/efi
vdb       252:16   0   368K  0 disk
vdc       252:32   0   100G  0 disk
sre@stg-hdpsiddharth103:~$ █
```

Figure 2.3: stg-hdpsiddharth103 Status

```
[sre@stg-hdpsiddharth104:~$ free -g
              total        used        free      shared  buff/cache   available
Mem:             31           0          30           0           0          30
Swap:             0           0           0
[sre@stg-hdpsiddharth104:~$ nproc
8
[sre@stg-hdpsiddharth104:~$ lsblk
NAME    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
vda       252:0    0    50G  0 disk
├─vda1    252:1    0  49.9G  0 part /
├─vda14   252:14   0     4M  0 part
└─vda15   252:15   0   106M  0 part /boot/efi
vdb       252:16   0   368K  0 disk
vdc       252:32   0   100G  0 disk
sre@stg-hdpsiddharth104:~$ █
```

Figure 2.4: stg-hdpsiddharth104 Status

```
[sre@stg-hdpsiddharth105:~$ free -g
              total        used        free      shared  buff/cache   available
Mem:             31           0          30           0           0          30
Swap:             0           0           0
[sre@stg-hdpsiddharth105:~$ nproc
8
[sre@stg-hdpsiddharth105:~$ lsblk
NAME    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
vda       252:0    0    50G  0 disk
├─vda1    252:1    0  49.9G  0 part /
├─vda14   252:14   0     4M  0 part
└─vda15   252:15   0   106M  0 part /boot/efi
vdb       252:16   0   368K  0 disk
vdc       252:32   0   100G  0 disk
sre@stg-hdpsiddharth105:~$ █
```

Figure 2.5: stg-hdpsiddharth105 Status

Chapter 3

Remote File Management via NBD

3.1   NBD set-up

```
1 modprobe nbd max_part=8
2 qemu-nbd --connect=/dev/nbd0 /opt/var/lib/libvirt/images/stg-
    hdpsiddharth105.img
3 fdisk -l /dev/nbd0
4 mkdir -p /mnt/nbd_vm
5 mount /dev/nbd0p1 /mnt/nbd_vm
6 #REMOVE THE TEMPORARY FILE BY ACCESING IT THROUGH /mnt/nbd_vm
7 umount /mnt/nbd_vm
8 qemu-nbd --disconnect /dev/nbd0
9 rmdir /mnt/nbd_vm
```

Listing 3.1: steps for nbd setup

Explanation
- Line 1: Loads the `nbd` (Network Block Device) kernel module with support for up to 8 partitions per device.
- Line 2: Connects the VM disk image (`stg-hdpsiddharth105.img`) to the local block device `/dev/nbd0` using `qemu-nbd`.
- Line 3: Lists the partition table of `/dev/nbd0` to identify available partitions inside the VM's disk.
- Line 4: Creates a local mount directory `/mnt/nbd_vm` to serve as the mount point.
- Line 5: Mounts the first partition of the NBD device (`/dev/nbd0p1`) onto `/mnt/nbd_vm`, making the VM filesystem accessible.
- Line 6: Removes the temporary file from inside the mounted filesystem by accessing it through `/mnt/nbd_vm`.

- Line 7: Unmounts the filesystem from **/mnt/nbd_vm** to safely detach it.
- Line 8: Disconnects the NBD device (**/dev/nbd0**) from the VM disk image, freeing the device.
- Line 9: Removes the temporary mount directory **/mnt/nbd_vm** to clean up.

3.2 Screenshots

```
[root@stg-hdpinterncloudb102.phonepe.nb6 ~]$ virsh list --all
 Id   Name                State
----------------------------------------
 6    stg-hdpsiddharth101   running
 7    stg-hdpsiddharth102   running
 8    stg-hdpsiddharth103   running
 9    stg-hdpsiddharth104   running
 -    stg-hdpsiddharth105   shut off

[root@stg-hdpinterncloudb102.phonepe.nb6 ~]$ qemu-nbd --connect=/dev/nbd0 /opt/var/lib/libvirt/images/stg-hdpsiddharth105.img
[root@stg-hdpinterncloudb102.phonepe.nb6 ~]$ fdisk -l /dev/nbd0
Disk /dev/nbd0: 50 GiB, 53687091200 bytes, 104857600 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: A29E570E-1767-4110-86C7-7755F7F80781

Device       Start      End    Sectors  Size Type
/dev/nbd0p1   227328 104857566 104630239 49.9G Linux filesystem
/dev/nbd0p14    2048    10239      8192    4M BIOS boot
/dev/nbd0p15   10240   227327    217088  106M EFI System

Partition table entries are not in disk order.
[root@stg-hdpinterncloudb102.phonepe.nb6 ~]$ mkdir -p /mnt/nbd_vm
[root@stg-hdpinterncloudb102.phonepe.nb6 ~]$ ls
snap
[root@stg-hdpinterncloudb102.phonepe.nb6 ~]$ cd /
[root@stg-hdpinterncloudb102.phonepe.nb6 /]$ ls
backup  bin  boot  dev  disable_if_netplan.sh  etc  home  install_grub.sh  lib  lib32  lib64  libx32  lost+found  media  mnt  opt  proc  root  run  sbin  snap  SPACE_SAVER_FILE  srv  storage  sys  test  tmp  usr  var
[root@stg-hdpinterncloudb102.phonepe.nb6 /]$ cd mnt/
[root@stg-hdpinterncloudb102.phonepe.nb6 mnt]$ ls
nbd_vm
[root@stg-hdpinterncloudb102.phonepe.nb6 mnt]$ mount /dev/nbd0p1 /mnt/nbd_vm
[root@stg-hdpinterncloudb102.phonepe.nb6 mnt]$ ls
nbd_vm
[root@stg-hdpinterncloudb102.phonepe.nb6 mnt]$ cd nbd_vm/
[root@stg-hdpinterncloudb102.phonepe.nb6 nbd_vm]$ ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  lost+found  media  mnt  opt  proc  root  run  sbin  snap  SPACE_SAVER_FILE  srv  sys  tmp  usr  var
[root@stg-hdpinterncloudb102.phonepe.nb6 nbd_vm]$ cd home/
[root@stg-hdpinterncloudb102.phonepe.nb6 home]$ cd sre/
[root@stg-hdpinterncloudb102.phonepe.nb6 sre]$ ls
hello.txt
[root@stg-hdpinterncloudb102.phonepe.nb6 sre]$ rm -f hello.txt
[root@stg-hdpinterncloudb102.phonepe.nb6 sre]$ ls
[root@stg-hdpinterncloudb102.phonepe.nb6 sre]$ umount /mnt/nbd_vm
umount: /mnt/nbd_vm: target is busy.
[root@stg-hdpinterncloudb102.phonepe.nb6 sre]$ cd /
[root@stg-hdpinterncloudb102.phonepe.nb6 /]$ umount /mnt/nbd_vm
[root@stg-hdpinterncloudb102.phonepe.nb6 /]$ qemu-nbd --disconnect /dev/nbd0
/dev/nbd0 disconnected
[root@stg-hdpinterncloudb102.phonepe.nb6 /]$ rmdir /mnt/nbd_vm
[root@stg-hdpinterncloudb102.phonepe.nb6 /]$ virsh start stg-hdpsiddharth105
Domain stg-hdpsiddharth105 started

[root@stg-hdpinterncloudb102.phonepe.nb6 /]$
```

Figure 3.1: using nbd to remove hello.txt on stg-hdpsiddharth105