

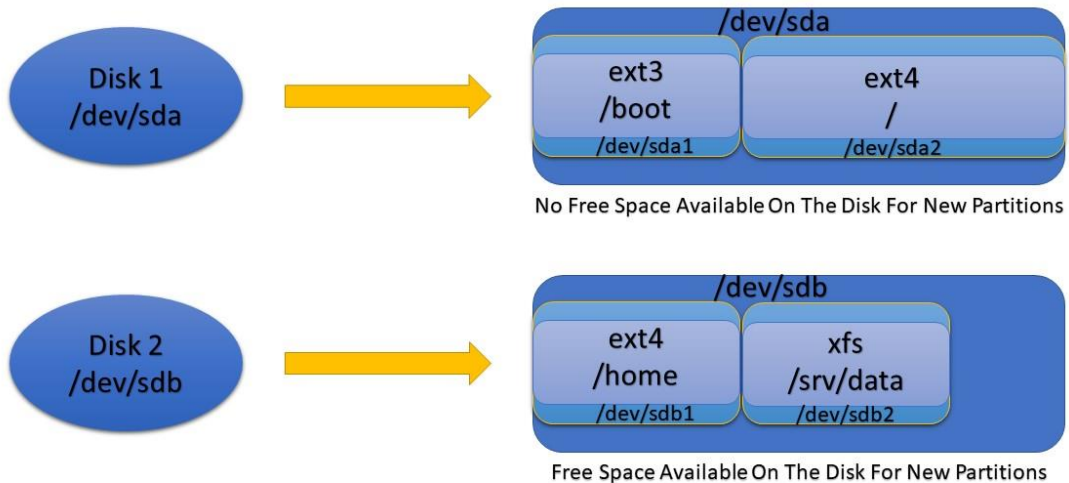
Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

Session - 32 Agenda:

1. Using LVM With Ansible

Problems With Standard Partitions



Suppose you want to enlarge the space available for /srv/data on disk 2, What can you do? The solution will always force you to unmount the file system, take a backup of the data, remove and recreate partitions, and then restore the data and remount the file system.

Solution With Logical Volume (LVM)



Managing Storage With LVM:

To create the logical volumes, we create the partitions on the raw disk(s) first and then we create the physical volumes with the help of these partitions. After that we create the volume groups by using the physical volumes created earlier and then we create the logical volumes on these volume groups. After that filesystems can be created on these logical volumes. These filesystems which are created on the logical volumes can be easily extended on the go without doing any extra efforts.

Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
# fdisk -l
# lsblk
# fdisk /dev/sda
# partprobe /dev/sda
# fdisk -l
# lsblk
# pvcreate /dev/sda1 /dev/sda2
# vgcreate nehraclasses /dev/sda1 /dev/sda2
# lvcreate -l 200 -n linux nehraclasses
# lvcreate -L 1G -n unix nehraclasses
# mkfs.ext4 /dev/nehraclasses/linux
# mkfs.xfs /dev/nehraclasses/unix
# mkdir /linux ; mkdir /unix
# mount /dev/nehraclasses/linux /linux
# mount /dev/nehraclasses/unix /unix
# df -hT
# cd /linux
# touch file{1..20}.txt
# ls -lh
# cd
# umount /linux
# vim /etc/fstab
/dev/nehraclasses/linux /linux ext4 defaults 0 0
/dev/nehraclasses/unix /unix xfs defaults 0 0
# mount -a
# df -hT
# ls -lh /linux
# lsblk -f
# lvextend -l +200 /dev/nehraclasses/linux
# lsblk -f
# df -hT
# resize2fs /dev/nehraclasses/linux
# lsblk -f
# df -hT
# lvextend -L 2G /dev/nehraclasses/unix
# lsblk -f
# df -hT
# xfs_growfs /dev/nehraclasses/unix
# lsblk -f
# df -hT
```

Using LVM With Ansible:

There are two ways of managing LVM in Ansible using:

1. Ansible Ad-hoc Commands
2. Ansible Playbooks

1. Ansible Ad-hoc Commands:

We can use ansible ad-hoc commands to create/manage the logical volumes with the help of lvg, lvof, filesystem & mount modules in ansible to get our job done on the managed nodes. In order



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

to use these modules, we need to install their collections from ansible galaxy because ansible-core doesn't provide these modules in RHEL9 by default.

\$ ansible-galaxy collection install ansible.posix

\$ ansible-galaxy collection install community.general

Let's understand the working of these modules with the help of ansible ad-hoc commands:

\$ ansible node1 -m raw -a 'echo -e "\n\n\n1\n\n\nw" | sudo fdisk /dev/sda1 -b

\$ ansible node1 -m command -a 'lsblk -f'

\$ ansible node1 -m lvg -a 'vg=nehraclasses pvs=/dev/sda1 pesize=32' -b

\$ ansible node1 -m lvol -a 'vg=nehraclasses lv=test size=100m' -b

\$ ansible node1 -m command -a 'pvs' -b

\$ ansible node1 -m command -a 'vgs' -b

\$ ansible node1 -m command -a 'lvs' -b

\$ ansible node1 -m filesystem -a 'dev=/dev/nehraclasses/test fstype=ext4 state=present' -b

\$ ansible node1 -m command -a 'lsblk -f'

\$ ansible node1 -m mount -a 'path=/opt/ src=/dev/nehraclasses/test fstype=ext4 opts=defaults state=mounted' -b

\$ ansible node1 -m command -a 'tail -1 /etc/fstab'

\$ ansible node1 -m command -a 'df -hT'

\$ ansible node1 -m command -a 'lvs' -b

\$ ansible node1 -m command -a 'lsblk -f'

\$ ansible node1 -m lvol -a 'vg=nehraclasses lv=test size=+1G resizefs=true' -b

\$ ansible node1 -m command -a 'df -hT'

\$ ansible node1 -m command -a 'lvs' -b

\$ ansible node1 -m command -a 'lsblk -f'

\$ ansible node1 -m filesystem -a 'dev=/dev/nehraclasses/test2 fstype=xfs state=present' -b

\$ ansible node1 -m command -a 'lvs' -b

\$ ansible node1 -m command -a 'lsblk -f'

\$ ansible node1 -m lvol -a 'vg=nehraclasses lv=test2 size=+1G resizefs=true' -b

\$ ansible node1 -m command -a 'lvs' -b

\$ ansible node1 -m filesystem -a 'dev=/dev/nehraclasses/test2 state=absent' -b

\$ ansible node1 -m command -a 'lsblk -f'

\$ ansible node1 -m lvol -a 'vg=nehraclasses lv=test2 state=absent force=true' -b

\$ ansible node1 -m command -a 'lvs' -b

\$ ansible node1 -m mount -a 'path=/opt/ src=/dev/nehraclasses/test fstype=ext4 opts=defaults state=absent' -b

\$ ansible node1 -m command -a 'df -hT' -b

\$ ansible node1 -m command -a 'tail -1 /etc/fstab' -b

\$ ansible node1 -m command -a 'lsblk -f'

\$ ansible node1 -m filesystem -a 'dev=/dev/nehraclasses/test state=absent' -b

\$ ansible node1 -m lvol -a 'vg=nehraclasses lv=test state=absent force=true' -b

\$ ansible node1 -m command -a 'lvs' -b

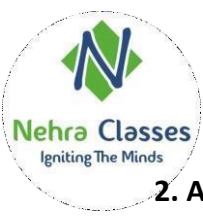
\$ ansible node1 -m command -a 'lsblk -f'

\$ ansible node1 -m lvg -a 'vg=nehraclasses pvs=/dev/sda1 state=absent' -b

\$ ansible node1 -m command -a 'vgs' -b

(Revert node1 machine from snapshot and add 3 raw disks to use Ansible Playbook examples.)

\$ ansible node1 -m command -a 'lsblk -f' -b



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

2. Ansible Playbooks:

Let's create some ansible playbooks using lvg, lvof, filesystem & mount modules to manage the logical volumes at the node1 machine.

\$ vim partition.yml

- name: Managing partitions on the disk(s)

hosts: node1

become: true

tasks:

- name: Create a new primary partition with a size of 1GiB on /dev/sda disk

community.general.partitioned:

device: /dev/sda

number: 1

state: present

part_end: 1GiB

...

\$ ansible-playbook partition.yml

\$ ansible node1 -m command -a 'lsblk -f' -b

\$ vim volgrp.yml

- name: Managing Volume Groups

hosts: node1

become: true

tasks:

- name: Create a volume group on top of /dev/sda1 with physical extent size = 32MB

community.general.lvg:

vg: nehraclasses

pvs: /dev/sda1

pesize: 32

...

\$ ansible-playbook volgrp.yml

\$ ansible node1 -m command -a 'pvs' -b

\$ ansible node1 -m command -a 'vgs' -b

\$ vim volgrp2.yml

- name: Managing Volume Groups

hosts: node1

become: true

tasks:

- name: Create a volume group on top of /dev/sdb with physical extent size = 128KiB

community.general.lvg:

vg: nehraclasses

pvs: /dev/sdb

pesize: 128K

...



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
$ ansible-playbook volgrp2.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ vim volgrp_res.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Create or resize a volume group on top of /dev/sdc.
```

```
community.general.lvg:
```

```
vg: nehraclasses
```

```
pvs: /dev/sdc
```

```
...
```

```
$ ansible-playbook volgrp_res.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ vim volgrp_rem.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Remove a volume group with name nehraclasses
```

```
community.general.lvg:
```

```
vg: nehraclasses
```

```
state: absent
```

```
...
```

```
$ ansible-playbook volgrp_rem.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ vim volgrp3.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Create a volume group on top of /dev/sda1 and resize the volume group /dev/sda1  
to the maximum possible
```

```
community.general.lvg:
```

```
vg: resizableVG
```

```
pvs: /dev/sda1
```

```
pvresize: true
```

```
...
```




Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
$ ansible-playbook volgrp3.yml
$ ansible node1 -m command -a 'pvs' -b
$ ansible node1 -m command -a 'vgs' -b
```

```
$ vim volgrp_deact.yml
```

```
- name: Managing Volume Groups
  hosts: node1
  become: true
  tasks:
    - name: Deactivate a volume group
      community.general.lvg:
        state: inactive
        vg: resizableVG
```

...

```
$ ansible-playbook volgrp_deact.yml
$ ansible node1 -m command -a 'pvs' -b
$ ansible node1 -m command -a 'vgs' -b
```

```
$ vim volgrp_act.yml
```

```
- name: Managing Volume Groups
  hosts: node1
  become: true
  tasks:
    - name: Deactivate a volume group
      community.general.lvg:
        state: active
        vg: resizableVG
```

...

```
$ ansible-playbook volgrp_act.yml
$ ansible node1 -m command -a 'pvs' -b
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'vgdisplay' -b
```

```
$ vim volgrp_uuid.yml
```

```
- name: Managing Volume Groups
  hosts: node1
  become: true
  tasks:
    - name: Reset a volume group UUID
      community.general.lvg:
        state: inactive
        vg: resizableVG
        reset_vg_uuid: true
```

...



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
$ ansible-playbook volgrp_uuid.yml
```

```
$ ansible node1 -m command -a 'vgdisplay' -b
```

```
$ ansible node1 -m command -a 'pvdisplay' -b
```

```
$ vim volgrp_uuid_pv.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Reset both volume group and pv UUID
```

```
community.general.lvg:
```

```
state: inactive
```

```
vg: resizableVG
```

```
pvs: /dev/sda1
```

```
reset_vg_uuid: true
```

```
reset_pv_uuid: true
```

```
...
```

```
$ ansible-playbook volgrp_uuid_pv.yml
```

```
$ ansible node1 -m command -a 'pvdisplay' -b
```

```
$ ansible node1 -m command -a 'vgdisplay' -b
```

```
$ ansible node1 -m lvg -a 'vg=resizableVG pvs=/dev/sda1 state=absent' -b
```

```
$ vim lvm.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Create a logical volume of 512m
```

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test
```

```
size: 512
```

```
...
```

```
$ ansible-playbook lvm.yml
```

```
$ ansible-playbook volgrp.yml
```

```
$ ansible-playbook lvm.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ ansible node1 -m command -a 'lsblk -f' -b
```



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
$ vim lvm2.yml
```

```
---
```

```
- name: Managing Volume Groups
hosts: node1
become: true
tasks:
  - name: Create cache pool logical volume
    community.general.lvol:
      vg: nehraclasses
      lv: lvcache
      size: 512m
      opts: --type cache-pool
```

```
...
```

```
$ ansible-playbook lvm2.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
$ ansible node1 -m command -a 'vgs' -b
$ ansible node1 -m command -a 'lvs' -b
$ ansible node1 -m command -a 'lsblk -f' -b
```

```
$ vim lvm3.yml
```

```
---
```

```
- name: Managing Volume Groups
hosts: node1
become: true
tasks:
  - name: Create a logical volume the size of all remaining space in the volume group
    community.general.lvol:
      vg: nehraclasses
      lv: test2
      size: 100%FREE
```

```
...
```

```
$ ansible-playbook lvm3.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
$ ansible node1 -m command -a 'vgs' -b
$ ansible node1 -m command -a 'lvs' -b
$ ansible node1 -m command -a 'lsblk -f' -b
```

```
$ vim lvm4.yml
```

```
---
```

```
- name: Managing Volume Groups
hosts: node1
become: true
tasks:
  - name: Remove the logical volume test2
    community.general.lvol:
      vg: nehraclasses
      lv: test2
```




Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
state: absent
force: true
```

...

```
$ ansible-playbook lvm4.yml
$ ansible node1 -m command -a 'pvs' -b
$ ansible node1 -m command -a 'vgs' -b
$ ansible node1 -m command -a 'lvs' -b
$ ansible node1 -m command -a 'lsblk -f' -b
```

```
$ vim lvm5.yml
```

```
- name: Managing Volume Groups
hosts: node1
become: true
tasks:
  - name: Create a logical volume with special options (Read ahead sectors)
# Readahead is a system call of the Linux kernel that loads a file's contents into the page cache.
  community.general.lvol:
    vg: nehraclasses
    lv: test3
    size: 5g
    opts: -r 16
```

...

```
$ ansible-playbook lvm5.yml
$ ansible node1 -m command -a 'pvs' -b
$ ansible node1 -m command -a 'vgs' -b
$ ansible node1 -m command -a 'lvdisplay' -b
$ ansible node1 -m command -a 'lsblk -f' -b
```

```
$ vim lvm6.yml
```

```
- name: Managing Volume Groups
hosts: node1
become: true
tasks:
  - name: Extend the logical volume to 1024m.
    community.general.lvol:
      vg: nehraclasses
      lv: test
      size: 1024
```

...

```
$ ansible-playbook lvm6.yml
$ ansible node1 -m command -a 'pvs' -b
$ ansible node1 -m command -a 'vgs' -b
$ ansible node1 -m command -a 'lvs' -b
$ ansible node1 -m command -a 'lsblk -f' -b
```



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
$ vim lvm7.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Extend the logical volume to consume all remaining space in the volume group
```

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test
```

```
size: +100%FREE
```

```
...
```

```
$ ansible-playbook lvm7.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ vim lvm8.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Reduce the logical volume to 512m
```

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test
```

```
size: 512
```

```
force: true
```

```
...
```

```
$ ansible-playbook lvm8.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ vim lvm9.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Extend the logical volume by given space
```

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test
```

```
size: +512M
```

```
...
```



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
$ ansible-playbook lvm9.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ vim lvm10.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Resize the logical volume to % of VG
```

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test
```

```
size: 20%VG
```

```
force: true
```

```
...
```

```
$ ansible-playbook lvm10.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ vim lvm11.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Reduce the logical volume by given space
```

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test
```

```
size: -512M
```

```
force: true
```

```
...
```

```
$ ansible-playbook lvm11.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ vim lvm12.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Create a snapshot volume of the test logical volume.
```



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test
```

```
snapshot: snap1
```

```
size: 100m
```

```
...
```

```
$ ansible-playbook lvm12.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ vim lvm13.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Deactivate a logical volume
```

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test
```

```
active: false
```

```
...
```

```
$ ansible-playbook lvm13.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ ansible node1 -m command -a 'lvscan' -b
```

```
$ vim lvm14.yml
```

```
---
```

```
- name: Managing Volume Groups
```

```
hosts: node1
```

```
become: true
```

```
tasks:
```

```
- name: Create a deactivated logical volume
```

```
community.general.lvol:
```

```
vg: nehraclasses
```

```
lv: test4
```

```
size: 1g
```

```
active: false
```

```
force: true
```

```
...
```

```
$ ansible-playbook lvm14.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```



Using LVM With Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
$ ansible node1 -m command -a 'lvscan' -b
```

```
$ vim lvm15.yml
```

```
---
```

```
- name: Managing Volume Groups
  hosts: node1
  become: true
  tasks:
    - name: Create a thin pool of 50m
      community.general.lvol:
        vg: nehraclasses
        thinpool: testpool
        size: 50m
```

```
...
```

```
$ ansible-playbook lvm15.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

```
$ vim lvm16.yml
```

```
---
```

```
- name: Managing Volume Groups
  hosts: node1
  become: true
  tasks:
    - name: Create a thin volume of 300m
      community.general.lvol:
        vg: nehraclasses
        lv: test6
        thinpool: testpool
        size: 300m
```

```
...
```

```
$ ansible-playbook lvm16.yml
```

```
$ ansible node1 -m command -a 'pvs' -b
```

```
$ ansible node1 -m command -a 'vgs' -b
```

```
$ ansible node1 -m command -a 'lvs' -b
```

Thank You