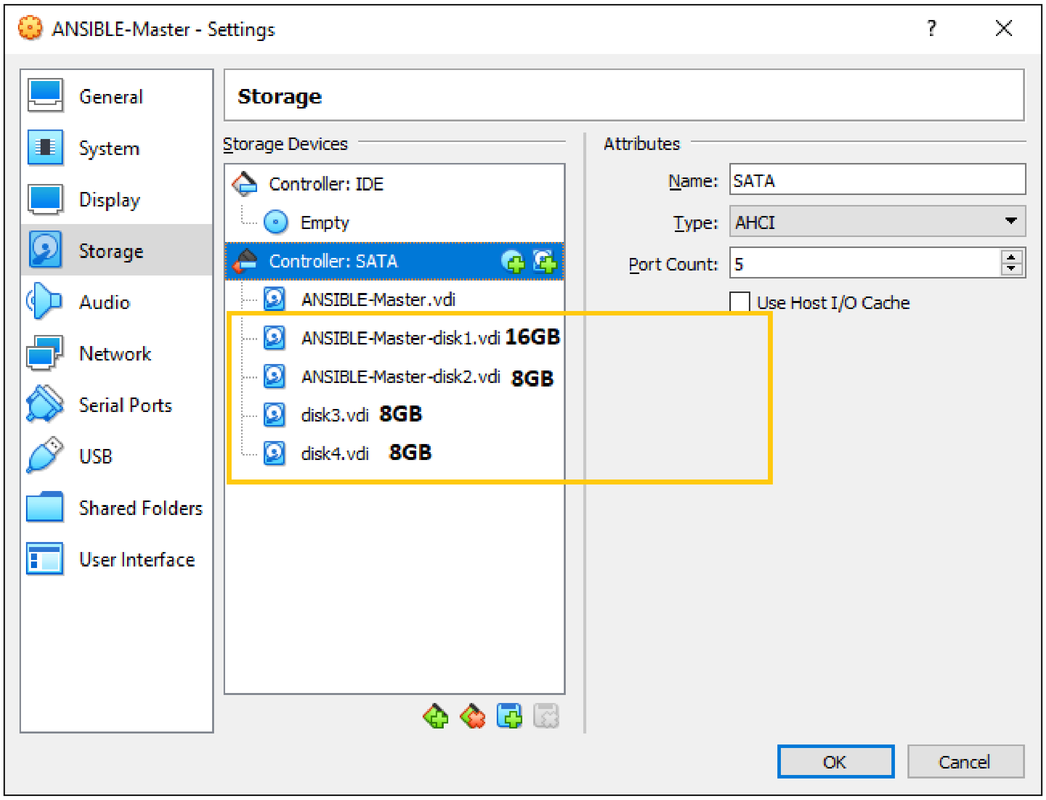
01-03-2020
https://youtu.be/CMIUv73DQZs

ANSIBLE – HOST SETUP

Create a master copy as shown



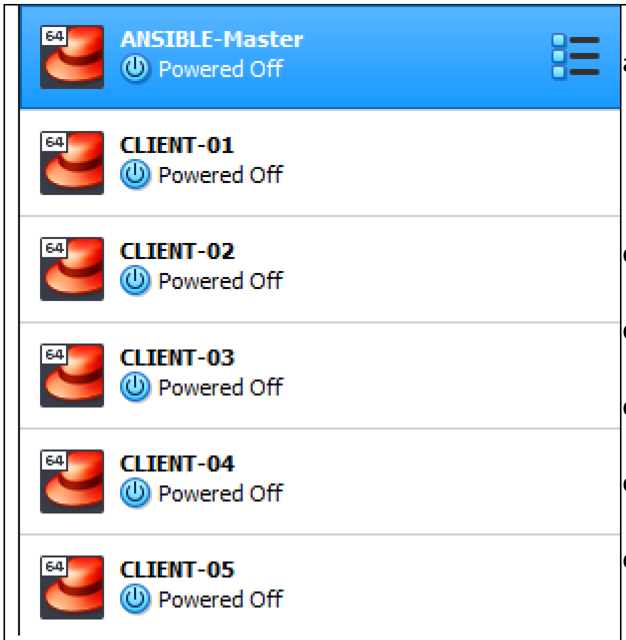
Storage

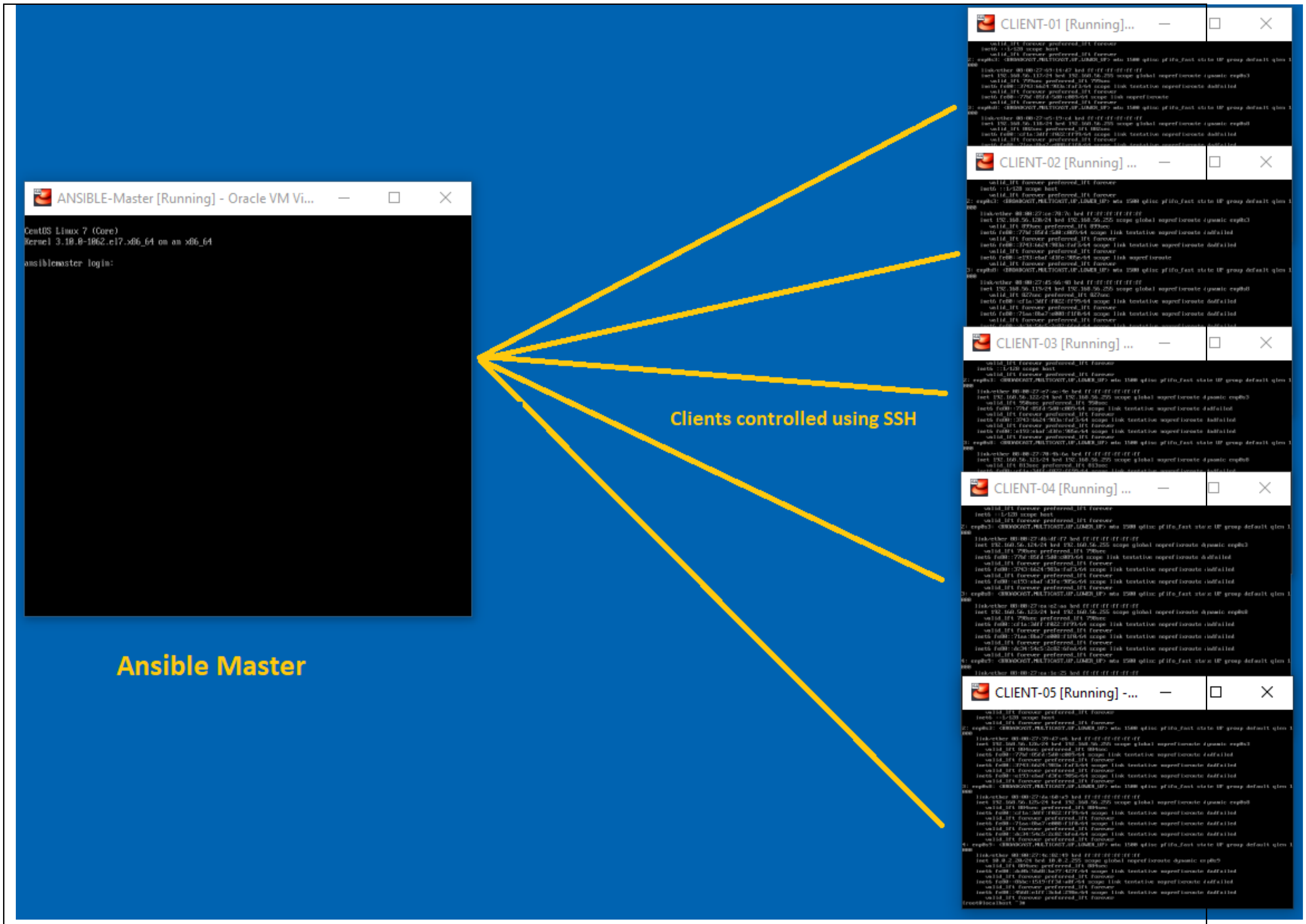
Storage Devices
- Controller: IDE
  - Empty
- Controller: SATA
  - ANSIBLE-Master.vdi
  - ANSIBLE-Master-disk1.vdi **16GB**
  - ANSIBLE-Master-disk2.vdi **8GB**
  - disk3.vdi **8GB**
  - disk4.vdi **8GB**

Attributes

Name: SATA
Type: AHCI
Port Count: 5
□ Use Host I/O Cache

Clone five times CLIENT-01 TO CLIENT-05

 ANSIBLE-Master Powered Off CLIENT-01 Powered Off CLIENT-02 Powered Off CLIENT-03 Powered Off CLIENT-04 Powered Off CLIENT-05 Powered Off	ansible01.zmpt.com client01.zmpt.com client02.zmpt.com client03.zmpt.com client04.zmpt.com client05.zmpt.com
01-10-2021 https://youtu.be/gzTJ_T3Ttus	



```
CentOS Linux 7 (Core)
Kernel 3.10.0-1062.el7.x86_64 on an x86_64

ansiblemaster login:
```

Ansible is an open-source software provisioning, configuration management, and application-deployment tool enabling infrastructure as code. It runs on many Unix-like systems, and can configure both Unix-like systems as well. It includes its own declarative language to describe system configuration. Ansible was written by Michael DeHaan and acquired by Red Hat in 2015. Ansible is agentless, temporarily connecting remotely via SSH.

Ansible is written Python

RHEL
SSH – Secure Shell
Agentless – no need to download the utility or agent on clients

Ansible Documentation

https://docs.ansible.com/

Ansible Master	Clients
192.168.56.116	192.168.56.117 192.168.56.120 192.168.56.122 192.168.56.124 192.168.56.126

Ansible installation on Master

```
[root@ansiblemaster ~]# yum install epel-release –y

[root@ansiblemaster ~]# yum install ansible –y
```

```
[root@ansiblemaster ~]# ansible --version

ansible 2.9.16                                          #< --- Ansible Version
  config file = /etc/ansible/ansible.cfg                    #< --- Configuration File (control behavior)
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Aug  7 2019, 00:51:29) [GCC 4.8.5 20150623 (Red Hat 4.8.5-39)]
```

```
[root@ansiblemaster ~]# vi /etc/ansible/ansible.cfg
```

```
[defaults]

# some basic default values...

#inventory      = /etc/ansible/hosts                    #< ---by default this is read for list of servers
#library        = /usr/share/my_modules/
#module_utils   = /usr/share/my_module_utils/
#remote_tmp     = ~/.ansible/tmp
#local_tmp      = ~/.ansible/tmp
#plugin_filters_cfg = /etc/ansible/plugin_filters.yml
```

```
#forks          = 5                              #< ---by default it manages 5 servers at a time
#poll_interval  = 15
#sudo_user      = root
#ask_sudo_pass = True
#ask_pass      = True
#transport      = smart
#remote_port    = 22
#module_lang    = C
#module_set_locale = False
```

Master node
Ansible Master – 192.168.56.116

Clients

192.168.56.117
192.168.56.120
192.168.56.122
192.168.56.124
192.168.56.126

Establish passwordless SSH

```
[root@ansiblemaster ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):          #< --- Hit Enter
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):                       #< --- Hit Enter
Enter same passphrase again:                                      #< --- Hit Enter
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:amnV+YWCR3RROkoY7JjZJFt3wCrnKZNXm1YI1pIvgJc root@ansiblemaster.zmpt.com
The key's randomart image is:
+---[RSA 2048]----+
|    . o.+o.oo.  |
|   . E O++...   |
|    . &.*ooo    |
|     * *=+o.o   |
|      =S+*+. .  |
|     +++.+o .   |
|     =+ .  .    |
|    o           |
|                |
+----[SHA256]-----+
```

```
[root@ansiblemaster ~]# cd /root/.ssh/
[root@ansiblemaster .ssh]# ls -la
total 8
drwx------. 2 root root   38 Jan 10 15:33 .
dr-xr-x---. 5 root root  175 Jan 10 15:32 ..
-rw-------. 1 root root 1675 Jan 10 15:33 id_rsa          #< --- Private Key
-rw-r--r--. 1 root root  409 Jan 10 15:33 id_rsa.pub     #< --- Public key
```

```
[root@ansiblemaster .ssh]# cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEApC6EKo2ruvwjoFvQsR5ZfqCbiZTBKeo0UB89olHoIWihcke8
Q/V4kVPvxt9rb6QMqINovoPUt9b8TdoDEpYVxcDQiBNN7VJw1Y25Y11aACKgSn/8
djOeowsQavJE74QQH5INKciVVWC5CxN8PSrfWFgb8u9k9N+Q7Bj1kLFwzbM6I0tT
v0KYIrHUcZV3QGEEiKMoacRvOK94vnNF87GmOng2pwy2nB4oSNtCanCkzz5sx8Vq
FfWRVOl4sWWbhrfXJTx7Ft3O722qocRbDaJcS5qr+Hh6//Tkqhkd06ZRR1dHK6Zd
hRMrCe68Dvxba0kf2YbAbdZ61M9w5qL3oTvNVQIDAQABAoIBAEkzhDIE4GCKMsg4
p5+QzI4sxaxd6pHpnlFq3GwW4MykffcaDNKwC4WgSvP8gUZRemNX08e9t4YQEIYV
gfEzgZmeElvZX5FEhJJykgm/gXU1+wm8bq+07DH9qOBGDH5N/3JUCBJtIEfkzBwo
H+iVyOb2wd9a5URYuaDKsBJHSugrHcP3DbpPUY5bbEI0gN7ie7UiI2ipUZfCC0y0
w/5AE4IiHNwlkT/2au+yEoYAYiWAHmSxksQxjoxGoxWacxacBrnVsbtXdTaCkR/4
aq83iWVSCot+mgTpX3p1yn6weVNWLSlk17JX4h3dNz8yvWb3Ees9qolXkH1p+ESV
PSjAxjECgYEA2fvC0HjZUeT7MrLsPR1TRYKiUASqKVq4qD71jjScdAuczprHKPcz
yubO99EY1N40yAxwE+/LYydZLI8NRW9tWs99ulhE1T5AIWZZtKvd+DkwwsfWRvzn
YhZHJBUoe9zTQVjZ1RJUvN8xNtRt+9eSGG14AQzwcZal0FRf4YnCb18CgYEAwNCv
kpPZssJZJS6xDNRrZbvDTWtDmwz/O76+7e8O32N6zC9vYlnRbDktY5XHLq9zF+F/
QSRW8XcBrAOHC2bifzoyTGNC3b6RGcS1GKFRgnuoQZgBOwZsIHRN2XbDtlVRcv2k
DAwjJOieAXU0Bf0E/sfpsKIrBczFqmV/Pg9co8sCgYA5AL83KQMbiVevMF7atHp8
KOctKasd2V2EkcJBB70KE6dT1+HQ8qQWoTjqUUG/GMSQs7/zVrfHHBkTr+z2hWcB
YlUsj7ygTwESOuOmfXM0YDBCH2QHeEdBvTWpV+cYTOzpn8SHFZ4XLf5+qhqAyySn
QqkpDSsraQu/aPER16aC3QKBgA1Q3C81Qr2TqnBk7xlBukEGXJCGLd8RYLcRj0ID
VnWa+yoKxJh4N3xP3pCYEW9p8nrdudoX1mFKpLTqcXMaId5DeKHJ4R5LQN4UtaUD
zwtB2vbctokLAoCjR5UTHKUE82ELnz6tv+eWoQLcw0D+nk+yIrNuH6aA483GwKja
ir1NAoGBANCFfE0Y4UZwyTfcTPDEOe7+/euPAwV41FM59UcRcYgHwS5MKdbpQKXg
6JmZSpTX4JvUTBJ/RHw0P9c98lIxBrMPIQ4ER/cOyAykuNctLFoN2PZgfxeo3FZw
1kWbgN7t5Wey0IanfGYRnyaw2Z2t2+L+Cms58eX/q8P8V5Es5onG
-----END RSA PRIVATE KEY-----
```

```
[root@ansiblemaster .ssh]# cat id_rsa.pub
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQCkLoQqjau6/COgW9CxHll+oJuJlMEp6jRQHz2iUeghaKFyR7xD9XiRU+/G32tvpAyog2i+g9S31vxN2gMSlhXFwNCIE03tUnDVjbljXVoAIqBKf/x2M56jCxBq8kTvhBAfkg0pyJV
VYLkLE3w9Kt9YWBvy72T035DsGPWQsXDNszojS1O/QpgisdRxlXdAYQSIoyhpxG84r3i+c0XzsaY6eDanDLacHihI20JqcKTPPmzHxWoV9ZFU6XixZZuGt9clPHsW3c7vbaqhxFsNolxLmqv4eHr/9OSqGR3TplFHV0crpl2F
EysJ7rwO/FtrSR/ZhsBt1nrUz3DmovehO81V root@ansiblemaster.zmpt.com
```

Copy the public keys to the Ansible clients

```
[root@ansiblemaster .ssh]# ssh-copy-id root@192.168.56.120
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '192.168.56.120 (192.168.56.120)' can't be established.
ECDSA key fingerprint is SHA256:e3LN1URGQEPwXaMbDeo+aTYev2cOOWnP3WKmaRG9gRU.
ECDSA key fingerprint is MD5:de:11:30:dd:ef:9e:ae:0a:ab:49:16:29:c9:08:36:8f. Are you sure you want to
continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
```

```
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@192.168.56.120's password:
```

Now establish connection for rest of the Ansible Clients

192.168.56.117
192.168.56.120
192.168.56.122
192.168.56.124
192.168.56.126

Now make the entry into /etc/ansible/hosts

```
[root@ansiblemaster ~]# vi /etc/ansible/hosts          #< --- Delete all content, add ip addressess

192.168.56.117
192.168.56.120
192.168.56.122
192.168.56.124
192.168.56.126
```

Ansible ad-hoc commands

```
[root@ansiblemaster ~]# ansible all -m ping
```

Command	All servers in /etc/ansible/hosts	Ansible Module	Ping module
ansible	all	-m	ping

```
192.168.56.122 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
```

Shell module – This allow you to run you familiar Linux commands

```
[root@ansiblemaster ~]# ansible all -m shell -a "ls -l"

[root@ansiblemaster ~]# ansible all -m shell -a "uptime"
```

```
[root@ansiblemaster ~]# ansible all -m shell -a "lsblk"

[root@ansiblemaster ~]# ansible all -m shell -a "df -h"

[root@ansiblemaster ~]# ansible all -m shell -a "free -h"

[root@ansiblemaster ~]# ansible all -m shell -a "free -h" > output.txt – redirect
```

Groups in hosts file

```
[chicago]
192.168.56.117
192.168.56.120
[ny]
192.168.56.122
192.168.56.124
[dc]
192.168.56.126
```

Specify group name instead of all

```
[root@ansiblemaster ~]# ansible ny -m shell -a "uptime"
192.168.56.124 | CHANGED | rc=0 >>
 16:13:00 up 12 min,  1 user,  load average: 0.04, 0.10, 0.11
192.168.56.122 | CHANGED | rc=0 >>
 16:13:00 up 12 min,  1 user,  load average: 0.00, 0.06, 0.09
```

Use a specific file instad of default /etc/ansible/hosts

-i to specify the file and its location

```
[root@ansiblemaster ~]# ansible all -i myservers -m shell -a "hostname"

[root@ansiblemaster ~]# ansible all -i /root/myservers -m shell -a "hostname"

[root@ansiblemaster ~]# ansible ny -i /root/myservers -m shell -a "hostname"
```

client01.zmpt.com - 192.168.56.117

client02.zmpt.com - 192.168.56.120

client03.zmpt.com - 192.168.56.122

client04.zmpt.com - 192.168.56.124

client05.zmpt.com - 192.168.56.126

Changing host name of the client

```
[root@ansiblemaster ~]# ansible "192.168.56.117" -m shell -a "echo "client01.zmpt.com" > /etc/hostname"
192.168.56.117 | CHANGED | rc=0 >>

[root@ansiblemaster ~]# ansible "192.168.56.120" -m shell -a "echo "client02.zmpt.com" > /etc/hostname"
192.168.56.120 | CHANGED | rc=0 >>

[root@ansiblemaster ~]# ansible "192.168.56.122" -m shell -a "echo "client03.zmpt.com" > /etc/hostname"
192.168.56.122 | CHANGED | rc=0 >>

[root@ansiblemaster ~]# ansible "192.168.56.124" -m shell -a "echo "client04.zmpt.com" > /etc/hostname"
192.168.56.124 | CHANGED | rc=0 >>

[root@ansiblemaster ~]# ansible "192.168.56.126" -m shell -a "echo "client05.zmpt.com" > /etc/hostname"
192.168.56.126 | CHANGED | rc=0 >>
```

01-16-2021
https://youtu.be/4xilWXES28c

01-17-2021

https://youtu.be/goqRTVYP-pw

Make hostname entries into DNS server

Make entries into DNS Primary

/var/named/forward.zmpt

/var/named/reverse.zmpt

Make DNS RELATED ENTRIES TO – **ANSIBLE MASTER**

Edit the following files

```
[root@localhost ~]# vi /etc/sysconfig/network-scripts/ifcfg-enp0s3
[root@localhost ~]# vi /etc/hostname
```

```
[root@localhost ~]# vi /etc/networks
[root@localhost ~]# vi /etc/hosts
[root@localhost ~]# vi /etc/resolv.conf
```

Create input file with DNS names

FQDN -

```
[root@ansiblemaster ~]# vi dnsnameservers

client01.zmpt.com
client02.zmpt.com
client03.zmpt.com
client04.zmpt.com
client05.zmpt.com
```

Create LVM using script

http://oct20.zmprotech.com/DATA/script.txt

```
#!/bin/bash

#This is a lvm script

echo "pv create is running"

pvcreate /dev/sdc /dev/sdd /dev/sde

echo "Script is creating VG ZMPT1"

vgcreate zmpt1 /dev/sdc /dev/sdd /dev/sde

echo "Creating Accounting LV"

lvcreate -n Accounting -L 4G zmpt1

echo "Creating Finance LV"

lvcreate -n Finance -L 6G zmpt1

echo "Creating HR LV"

lvcreate -n HR -L 2G zmpt1
```

```
echo "Creating Recruiting LV"

lvcreate -n Recruiting -L 4G zmpt1

echo "Creating file system"

mkfs.xfs /dev/zmpt1/Accounting

mkfs.xfs /dev/zmpt1/Finance

mkfs.xfs /dev/zmpt1/HR

mkfs.xfs /dev/zmpt1/Recruiting

echo "Performing partprobe"

partprobe

echo "Script is creating associated directories"

mkdir /accounting
mkdir /finance
mkdir /hr
mkdir /recruiting

echo "Making fstab entries"

echo "/dev/mapper/zmpt1-Accounting  /accounting      xfs     defaults       0 0" >>
/etc/fstab

echo "/dev/mapper/zmpt1-Finance     /finance      xfs     defaults       0 0" >>
/etc/fstab

echo "/dev/mapper/zmpt1-HR         /hr      xfs     defaults       0 0" >> /etc/fstab

echo "/dev/mapper/zmpt1-Recruiting  /recruiting      xfs     defaults       0 0" >>
/etc/fstab

echo "mounting from fstab"

mount -a

echo "Check the mount now"

df -h
```

Copy script to hosts

[root@ansiblemaster ~]# ansible all -i dnsnameservers -m copy -a "src=/root/lvmscript.scr mode=preserve dest=/root"

-m copy -a "src=/root/lvmscript.scr mode=preserve dest=/root"

-m	copy	src=/root/lvmscript.scr	Mode=preserve	Dest=/root
Module	Copy module	Source location	Preserve permissions	Location on remote host

Execute script on all the servers

[root@ansiblemaster ~]# ansible all -i dnsnameservers -m shell -a "./lvmscript.scr"

```
client01.zmpt.com | CHANGED | rc=0 >>
Filesystem                  Size  Used Avail Use% Mounted on
devtmpfs                    484M     0  484M   0% /dev
tmpfs                     496M     0  496M   0% /dev/shm
tmpfs                     496M   13M  483M   3% /run
tmpfs                     496M     0  496M   0% /sys/fs/cgroup
/dev/mapper/centos-root       14G  1.2G   13G   9% /
/dev/sda1                  1014M  136M  879M  14% /boot
tmpfs                     100M     0  100M   0% /run/user/0
/dev/mapper/zmpt1-Accounting  4.0G   33M  4.0G   1% /accounting
/dev/mapper/zmpt1-Finance     6.0G   33M  6.0G   1% /finance
/dev/mapper/zmpt1-HR          2.0G   33M  2.0G   2% /hr
/dev/mapper/zmpt1-Recruiting  4.0G   33M  4.0G   1% /recruiting
```

So far we were using ad-hoc commands

But ansible has comething called play-book – this uses built-in-commands created by ansible

Ansible play-books are written YAML language

YAML is a human-readable data-serialization language. It is commonly used for configuration files and in applications where data is being stored or transmitted.

Example of Playbook in YAML

```
---    #< --- you must have three hipens - for it to become yaml script

- hosts: all   #< ---hosts is built in module - reads the ansible defult file /etc/ansbile/hosts
  gather_facts: false  #< --- gather_facts is built in module
  tasks:              #< --- task" built in module
    - ping:           #< --- ping" built in module
```

01-23-2021
https://youtu.be/OY6ODRsp0Sc

```
---

- hosts: client01.zmpt.com
  gather_facts: true
  tasks:
    - ping:
```

```
[root@ansiblemaster ~]# vi lvmplaybook.yaml
```

```
---
 - hosts: all
   user: root
   tasks:
       - name: Create volume group on /dev/sdc /dev/sdd /dev/sde
         lvg:
             vg: zmpt1
             pvs: /dev/sdb,/dev/sdd,/dev/sde

#creating lvms

       - name: Accounting lvm
         lvol:
             vg: zmpt1
             lv: Accounting
             size: 10G

       - name: Finance lvm
         lvol:
             vg: zmpt1
```

```
            lv: Finance
            size: 6G

      - name: HR lvm
        lvol:
            vg: zmpt1
            lv: HR
            size: 6G

      - name: Recruiting lvm
        lvol:
            vg: zmpt1
-            lv: Recruiting
            size: 6G

#Creating file system

      - name: create file system for Accounting
        filesystem:
             fstype: xfs
             dev: /dev/zmpt1/Accounting

      - name: create file system for Finance
        filesystem:
             fstype: xfs
             dev: /dev/zmpt1/Finance
      - name: create file system for HR
        filesystem:
             fstype: xfs
             dev: /dev/zmpt1/HR

      - name: create file system Recruiting
        filesystem:
             fstype: xfs
             dev: /dev/zmpt1/Recruiting

#Create mount point, mount and fstab entry

      - name: mount Accounting logical volumes
        mount:
             name: /Accounting
             src: /dev/zmpt1/Accounting
             fstype: xfs
             state: mounted
```

```
   - name: mount Finance logical volumes
     mount:
         name: /Finance
         src: /dev/zmpt1/Finance
         fstype: xfs
         state: mounted
  - name: mount HR logical volumes
     mount:
         name: /HR
         src: /dev/zmpt1/HR
         fstype: xfs
         state: mounted

   - name: mount Recruiting logical volumes
     mount:
         name: /Recruiting
         src: /dev/zmpt1/Recruiting
         fstype: xfs
         state: mounted
```

```
[root@ansiblemaster ~]# ansible-playbook lvmplaybook.yaml
```

```
PLAY RECAP ****************************************************************************
client01.zmpt.com        : ok=14   changed=13   unreachable=0   failed=0   skipped=
client02.zmpt.com        : ok=14   changed=13   unreachable=0   failed=0   skipped=
client03.zmpt.com        : ok=14   changed=13   unreachable=0   failed=0   skipped=
client04.zmpt.com        : ok=14   changed=13   unreachable=0   failed=0   skipped=
client05.zmpt.com        : ok=14   changed=13   unreachable=0   failed=0   skipped=
```