Topic of project: Security key-based network access control in cloud instances

Team members: Ketan Sabne(CS20BTECH11043) TA mentor - Maruthi Inukonda

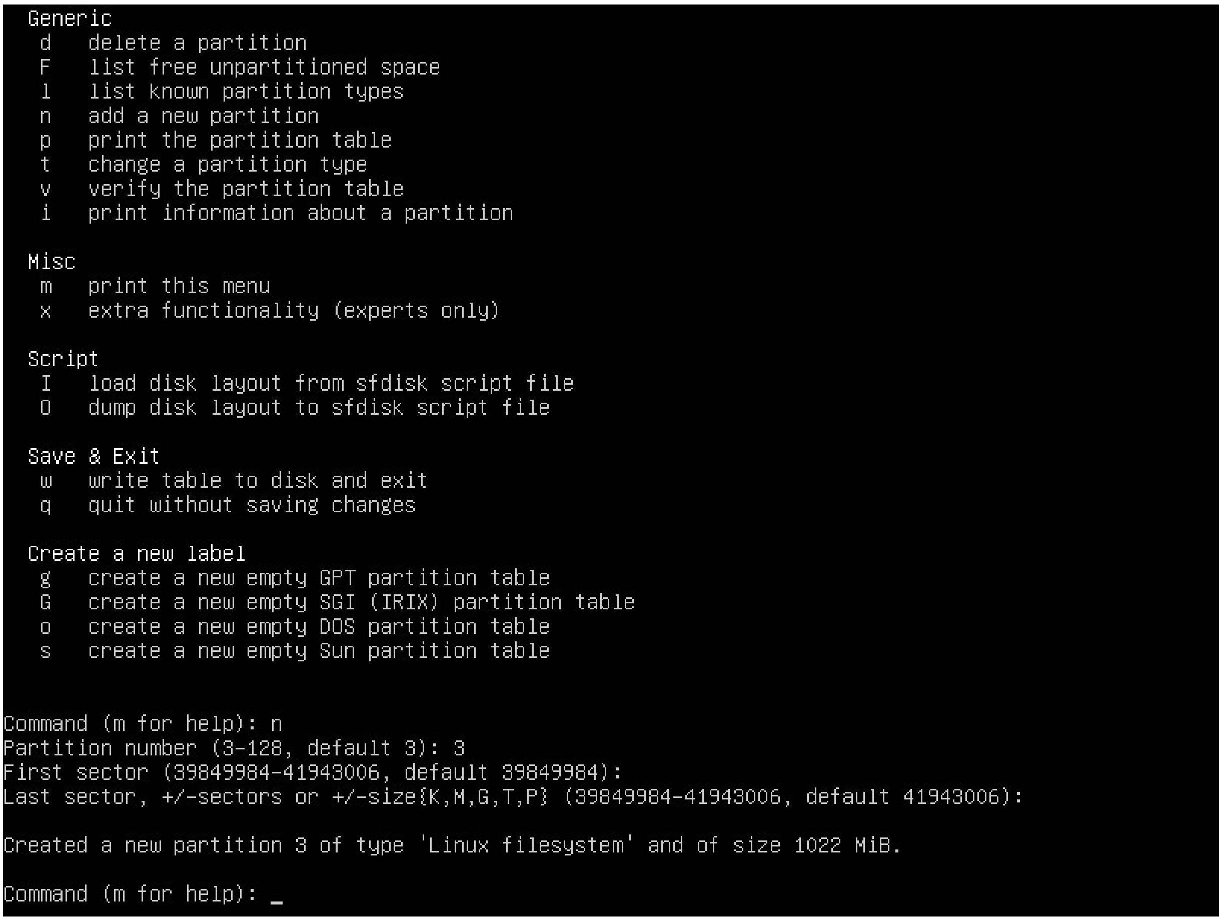
Praneeth Nayak (CS20BTECH11025)

Problem Statement

To provide a secure key-based network authentication for headless Linux systems in cloud instances that use network access control (NAC) and prevent credential leakage via virtual machine (VM) disk images. This involves implementing certificate-based authentication in netplan and utilizing encrypted file systems or disk volumes to ensure zero-leakage of authentication credentials. The goal is to obtain fine-grained identity information (user, instance/device) from network logs to improve security in cloud instances, especially as they are increasingly being used for cybercrime hosting services

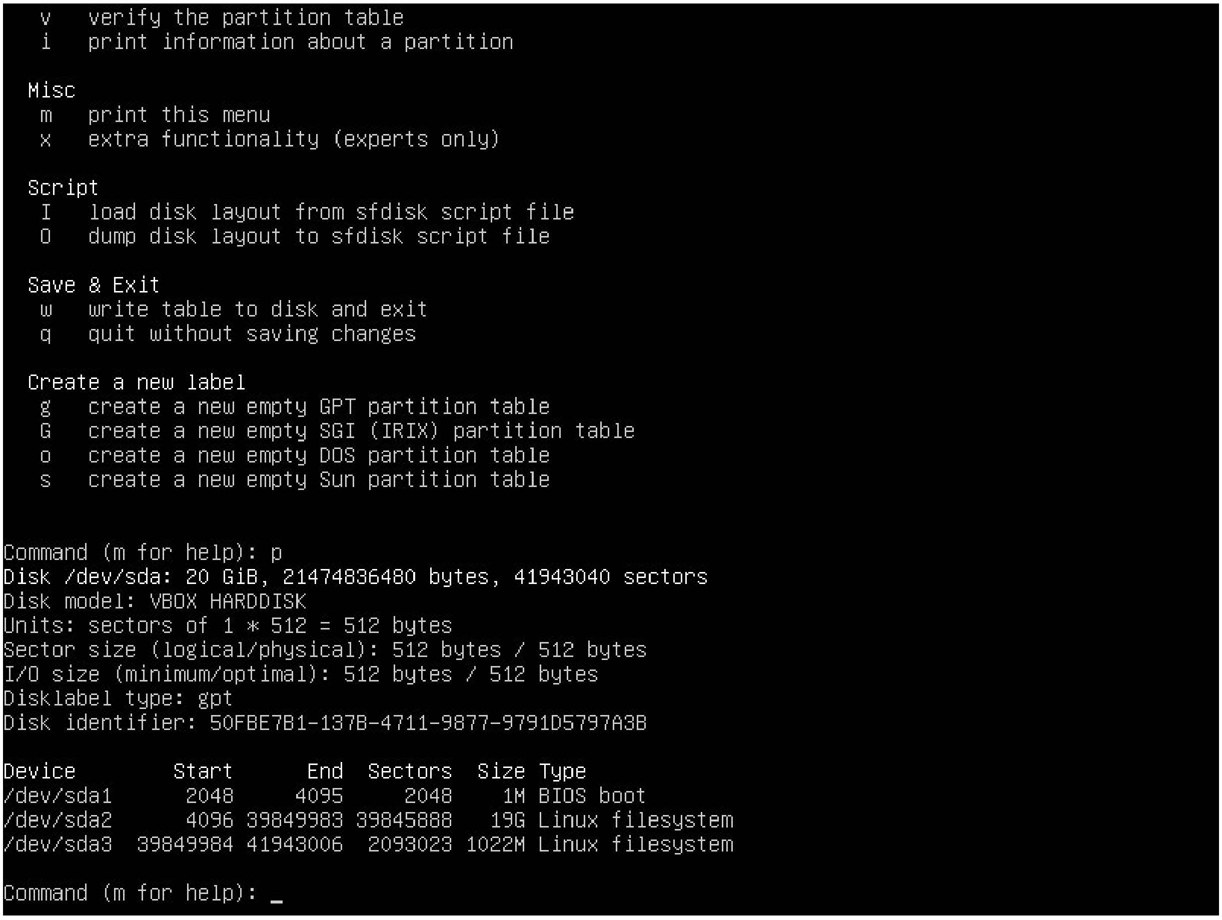
Encrypting the keys stored on the file-system (/secure)

First we have created a partition using sudo fdisk /dev/sda command , below Screenshot is step for creating new partition, for us new partition name created is /dev/sda3.



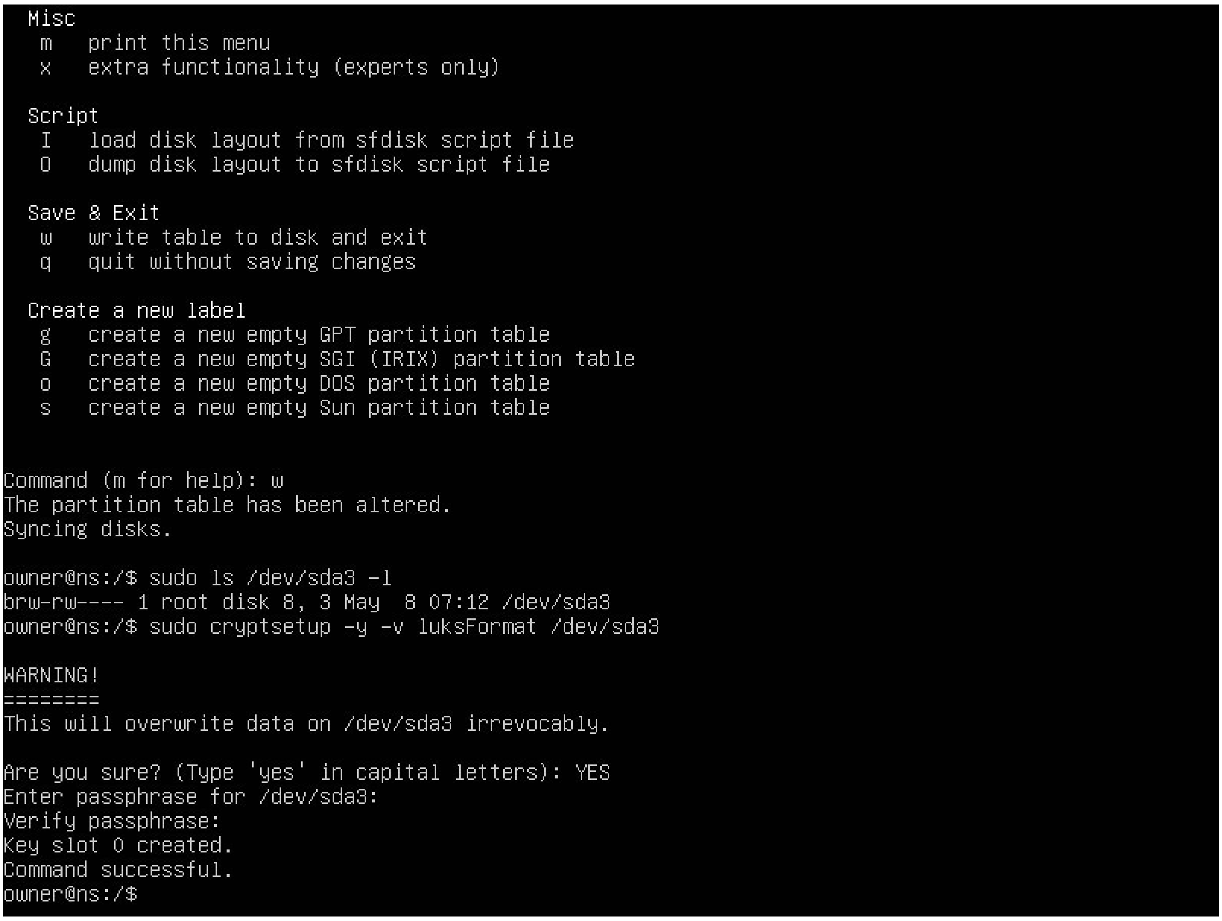
```
  Generic
   d   delete a partition
   F   list free unpartitioned space
   l   list known partition types
   n   add a new partition
   p   print the partition table
   t   change a partition type
   v   verify the partition table
   i   print information about a partition

  Misc
   m   print this menu
   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


Command (m for help): n
Partition number (3-128, default 3): 3
First sector (39849984-41943006, default 39849984):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (39849984-41943006, default 41943006):

Created a new partition 3 of type 'Linux filesystem' and of size 1022 MiB.

Command (m for help): _
```

Then we have checked whether it’s created or not , checking.



```
   v   verify the partition table
   i   print information about a partition

  Misc
   m   print this menu
   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


Command (m for help): p
Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VBOX HARDDISK
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: 50FBE7B1-137B-4711-9877-9791D5797A3B

Device        Start      End  Sectors  Size Type
/dev/sda1      2048     4095     2048    1M BIOS boot
/dev/sda2      4096 39849983 39845888   19G Linux filesystem
/dev/sda3  39849984 41943006  2093023 1022M Linux filesystem

Command (m for help): _
```

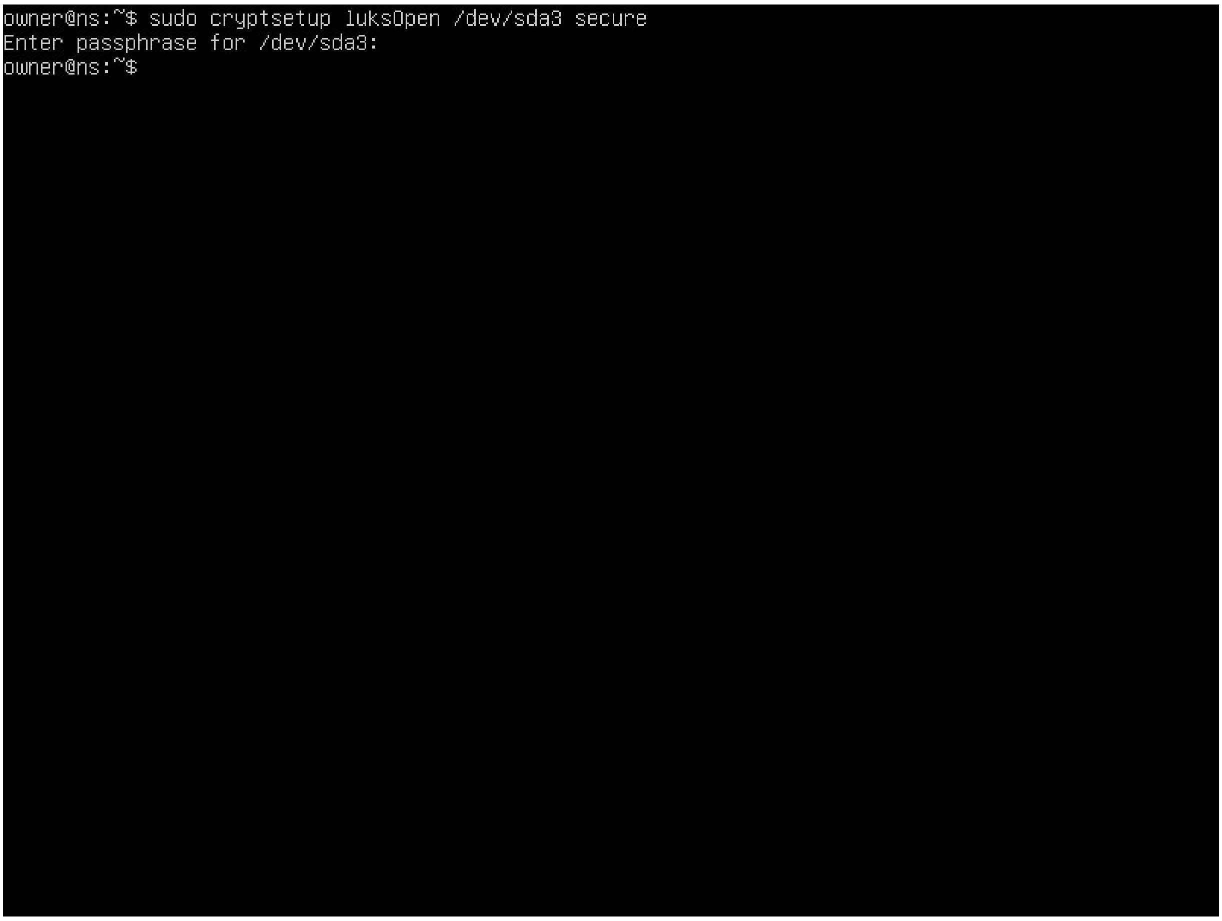
Now we have set up my LUKS partition with cryptsetup command, image is attached below with command. It has asked me to create passphrase.



```
  Misc
   m   print this menu
   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


Command (m for help): w
The partition table has been altered.
Syncing disks.

owner@ns:/$ sudo ls /dev/sda3 -l
brw-rw---- 1 root disk 8, 3 May  8 07:12 /dev/sda3
owner@ns:/$ sudo cryptsetup -y -v luksFormat /dev/sda3

WARNING!
========
This will overwrite data on /dev/sda3 irrevocably.

Are you sure? (Type 'yes' in capital letters): YES
Enter passphrase for /dev/sda3:
Verify passphrase:
Key slot 0 created.
Command successful.
owner@ns:/$
```

Now

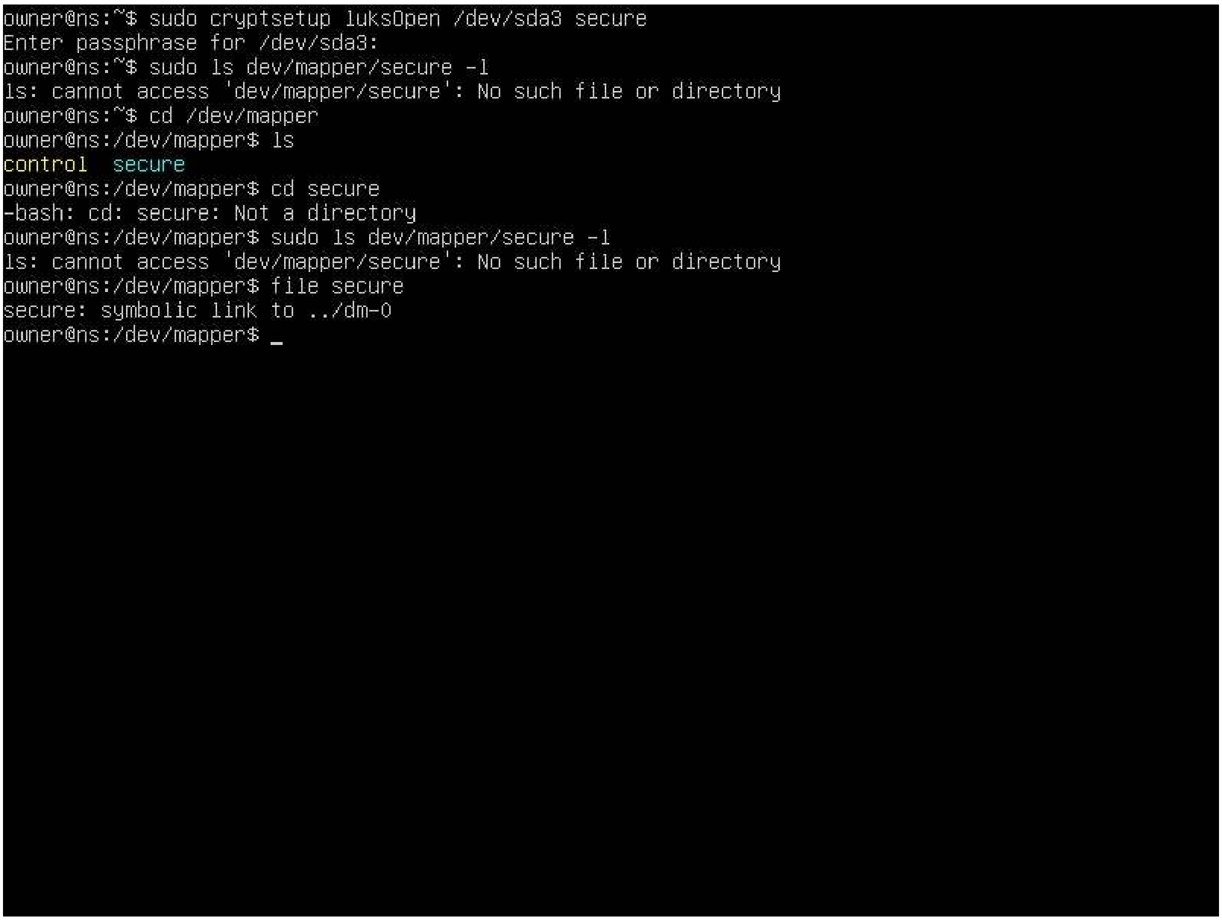
we have open the partition and created a mapping in /dev/mapper using below command. It has asked me to enter correct passphrase or we can't open partition (if u forgot passphrase).



```
owner@ns:~$ sudo cryptsetup luksOpen /dev/sda3 secure
Enter passphrase for /dev/sda3:
owner@ns:~$
```

we have checked that our setup name (secure) is mapped correctly or not.

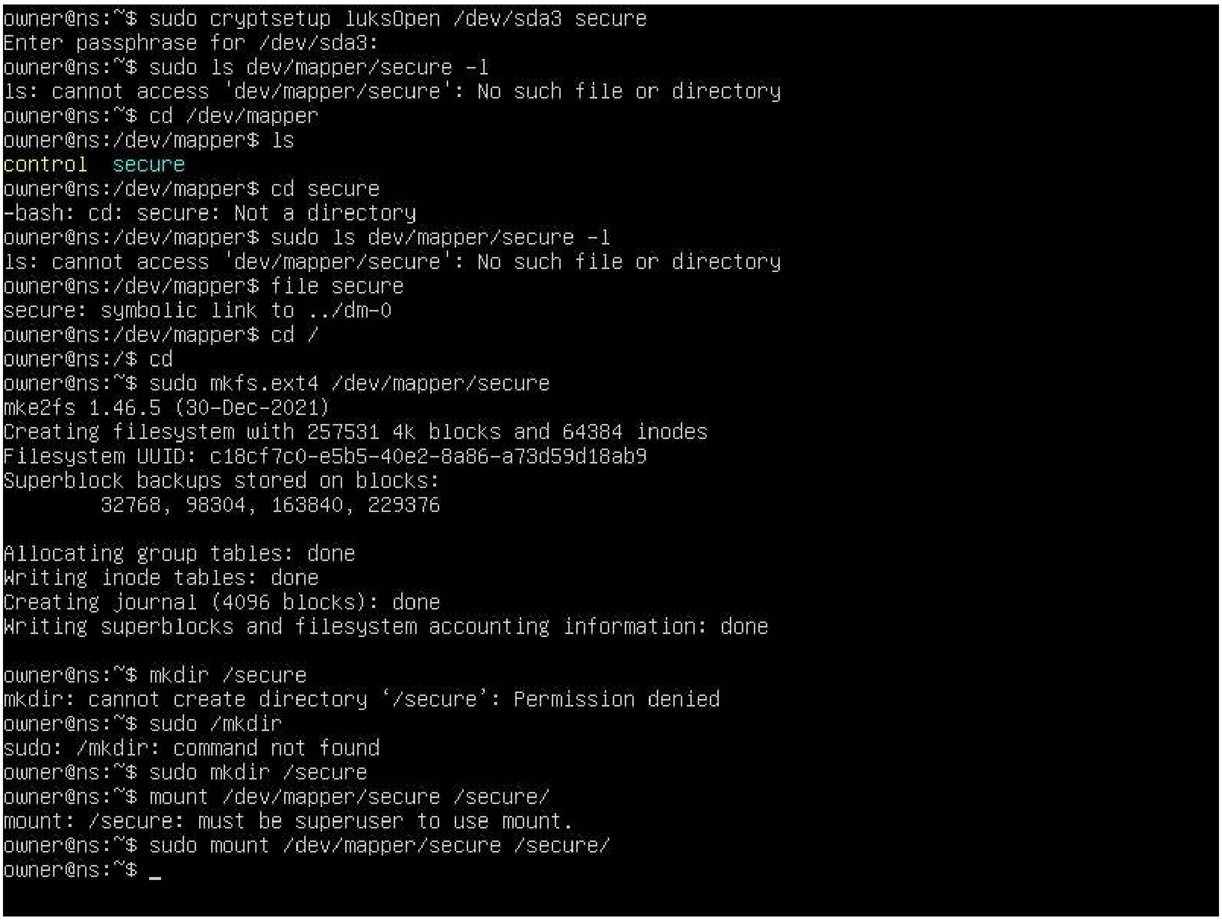
Now



```
owner@ns:~$ sudo cryptsetup luksOpen /dev/sda3 secure
Enter passphrase for /dev/sda3:
owner@ns:~$ sudo ls dev/mapper/secure -l
ls: cannot access 'dev/mapper/secure': No such file or directory
owner@ns:~$ cd /dev/mapper
owner@ns:/dev/mapper$ ls
control  secure
owner@ns:/dev/mapper$ cd secure
-bash: cd: secure: Not a directory
owner@ns:/dev/mapper$ sudo ls dev/mapper/secure -l
ls: cannot access 'dev/mapper/secure': No such file or directory
owner@ns:/dev/mapper$ file secure
secure: symbolic link to ../dm-0
owner@ns:/dev/mapper$ _
```

We are creating a file system with mkfs.ext4 command and mounting a new file system which will be used in storing certificates, Screenshot is attached below.

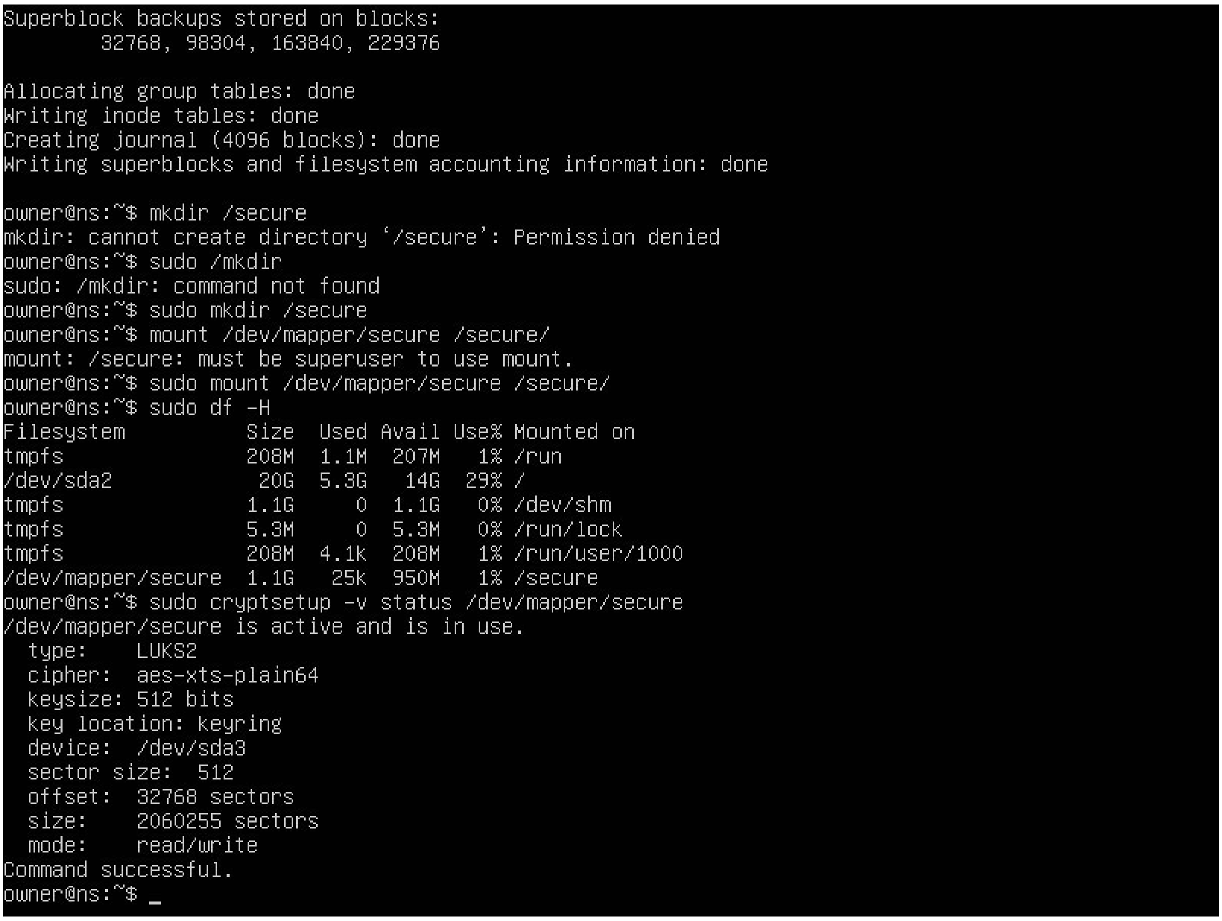
Now



```
owner@ns:~$ sudo cryptsetup luksOpen /dev/sda3 secure
Enter passphrase for /dev/sda3:
owner@ns:~$ sudo ls dev/mapper/secure -l
ls: cannot access 'dev/mapper/secure': No such file or directory
owner@ns:~$ cd /dev/mapper
owner@ns:/dev/mapper$ ls
control  secure
owner@ns:/dev/mapper$ cd secure
-bash: cd: secure: Not a directory
owner@ns:/dev/mapper$ sudo ls dev/mapper/secure -l
ls: cannot access 'dev/mapper/secure': No such file or directory
owner@ns:/dev/mapper$ file secure
secure: symbolic link to ../dm-0
owner@ns:/dev/mapper$ cd /
owner@ns:/$ cd
owner@ns:~$ sudo mkfs.ext4 /dev/mapper/secure
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 257531 4k blocks and 64384 inodes
Filesystem UUID: c18cf7c0-e5b5-40e2-8a86-a73d59d18ab9
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

owner@ns:~$ mkdir /secure
mkdir: cannot create directory '/secure': Permission denied
owner@ns:~$ sudo /mkdir
sudo: /mkdir: command not found
owner@ns:~$ sudo mkdir /secure
owner@ns:~$ mount /dev/mapper/secure /secure/
mount: /secure: must be superuser to use mount.
owner@ns:~$ sudo mount /dev/mapper/secure /secure/
owner@ns:~$ _
```

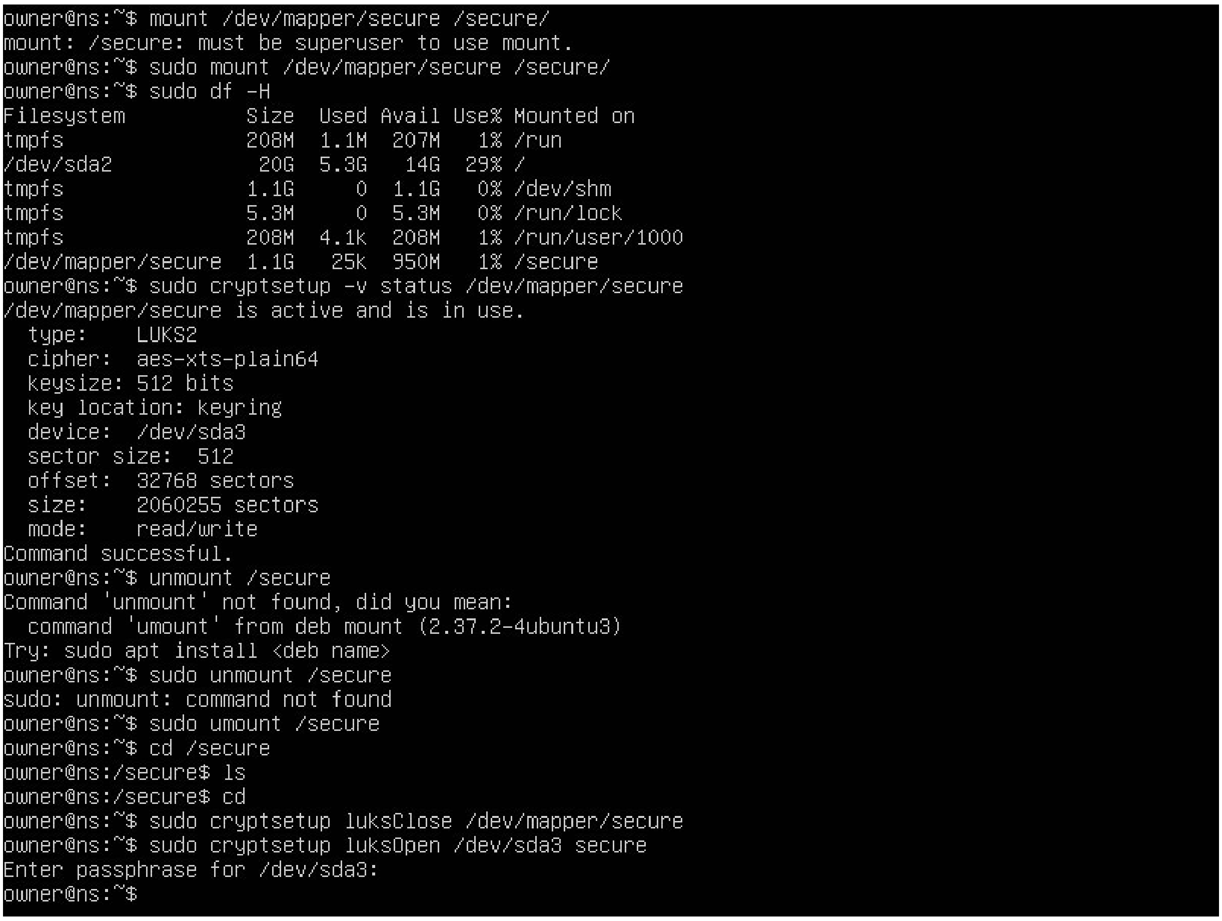
checking status of our /dev/mapper/sda3 , then it shows all the info such that for encryption we have used LUKS 2 also ciphers and key size,etc.

Now



```
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

owner@ns:~$ mkdir /secure
mkdir: cannot create directory ‘/secure’: Permission denied
owner@ns:~$ sudo /mkdir
sudo: /mkdir: command not found
owner@ns:~$ sudo mkdir /secure
owner@ns:~$ mount /dev/mapper/secure /secure/
mount: /secure: must be superuser to use mount.
owner@ns:~$ sudo mount /dev/mapper/secure /secure/
owner@ns:~$ sudo df -H
Filesystem          Size  Used Avail Use% Mounted on
tmpfs               208M  1.1M  207M   1% /run
/dev/sda2            20G  5.3G   14G  29% /
tmpfs               1.1G     0  1.1G   0% /dev/shm
tmpfs               5.3M     0  5.3M   0% /run/lock
tmpfs               208M  4.1k  208M   1% /run/user/1000
/dev/mapper/secure  1.1G   25k  950M   1% /secure
owner@ns:~$ sudo cryptsetup -v status /dev/mapper/secure
/dev/mapper/secure is active and is in use.
  type:    LUKS2
  cipher:  aes-xts-plain64
  keysize: 512 bits
  key location: keyring
  device:  /dev/sda3
  sector size:  512
  offset:  32768 sectors
  size:    2060255 sectors
  mode:    read/write
Command successful.
owner@ns:~$ _
```

Now we have unmounted ‘secure’ and now whenever we have to open our partition created for certificates, we need to enter passphrase . You can see in below screenshot.

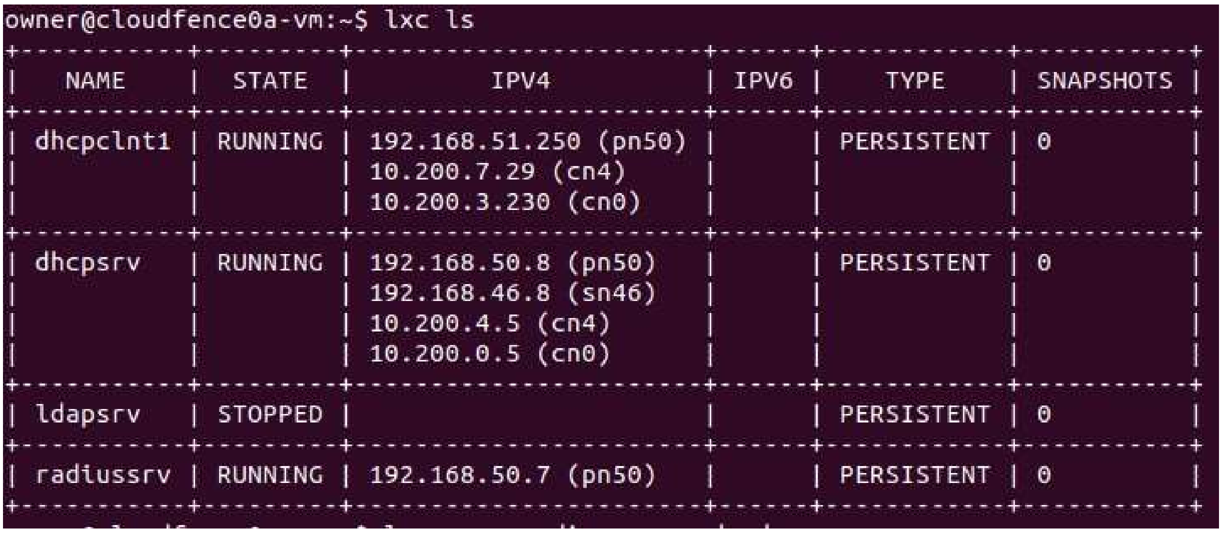


```
owner@ns:~$ mount /dev/mapper/secure /secure/
mount: /secure: must be superuser to use mount.
owner@ns:~$ sudo mount /dev/mapper/secure /secure/
owner@ns:~$ sudo df -H
Filesystem          Size  Used Avail Use% Mounted on
tmpfs               208M  1.1M  207M   1% /run
/dev/sda2            20G  5.3G   14G  29% /
tmpfs               1.1G     0  1.1G   0% /dev/shm
tmpfs               5.3M     0  5.3M   0% /run/lock
tmpfs               208M  4.1k  208M   1% /run/user/1000
/dev/mapper/secure  1.1G   25k  950M   1% /secure
owner@ns:~$ sudo cryptsetup -v status /dev/mapper/secure
/dev/mapper/secure is active and is in use.
  type:    LUKS2
  cipher:  aes-xts-plain64
  keysize: 512 bits
  key location: keyring
  device:  /dev/sda3
  sector size:  512
  offset:  32768 sectors
  size:    2060255 sectors
  mode:    read/write
Command successful.
owner@ns:~$ unmount /secure
Command 'unmount' not found, did you mean:
  command 'umount' from deb mount (2.37.2-4ubuntu3)
Try: sudo apt install <deb name>
owner@ns:~$ sudo unmount /secure
sudo: unmount: command not found
owner@ns:~$ sudo umount /secure
owner@ns:~$ cd /secure
owner@ns:/secure$ ls
owner@ns:/secure$ cd
owner@ns:~$ sudo cryptsetup luksClose /dev/mapper/secure
owner@ns:~$ sudo cryptsetup luksOpen /dev/sda3 secure
Enter passphrase for /dev/sda3:
owner@ns:~$
```

```
network:
  version:
  2 wifis:
  wl0:
      access-points:
        university:
          auth: key-management: eap method: tls
      anonymous-identity: "@cust.example.com"
      identity: "testuser" ca-certificate:
      /secure/cust-cacrt.pem client-certificate:
      /secure/cust-crt.pem client-key: /secure/cust-
      key.pem client-key-password: "xyz" dhcp4: yes
```

802.1x using RADIUS and certificate and keys

Setup



```
owner@cloudfence0a-vm:~$ lxc ls
+-----------+---------+-----------------------+------+------------+-----------+
|   NAME    |  STATE  |         IPV4          | IPV6 |    TYPE    | SNAPSHOTS |
+-----------+---------+-----------------------+------+------------+-----------+
| dhcpclnt1 | RUNNING | 192.168.51.250 (pn50) |      | PERSISTENT | 0         |
|           |         | 10.200.7.29 (cn4)     |      |            |           |
|           |         | 10.200.3.230 (cn0)    |      |            |           |
+-----------+---------+-----------------------+------+------------+-----------+
| dhcpsrv   | RUNNING | 192.168.50.8 (pn50)   |      | PERSISTENT | 0         |
|           |         | 192.168.46.8 (sn46)   |      |            |           |
|           |         | 10.200.4.5 (cn4)      |      |            |           |
|           |         | 10.200.0.5 (cn0)      |      |            |           |
+-----------+---------+-----------------------+------+------------+-----------+
| ldapsrv   | STOPPED |                       |      | PERSISTENT | 0         |
+-----------+---------+-----------------------+------+------------+-----------+
| radiussrv | RUNNING | 192.168.50.7 (pn50)   |      | PERSISTENT | 0         |
+-----------+---------+-----------------------+------+------------+-----------+
```

RADIUS Server

```
root@radiussrv:~# ip r default via
192.168.50.1 dev pn50 proto static
192.168.50.0/23 dev pn50 proto kernel scope link src 192.168.50.7
```

DHCP Server

```
root@dhcpsrv:~# ip r
default via 10.200.0.1 dev cn0 proto static
10.200.0.0/22 dev cn0 proto kernel scope link src 10.200.0.5
10.200.4.0/22 dev cn4 proto kernel scope link src 10.200.4.5
192.168.46.0/23 dev sn46 proto kernel scope link src 192.168.46.8
192.168.50.0/23 dev pn50 proto kernel scope link src 192.168.50.8
```

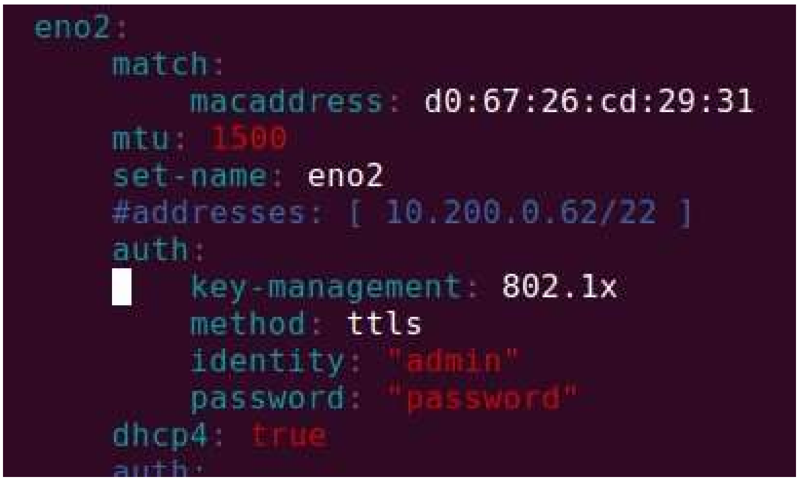
DHCP client1

```
root@dhcpclnt1:~# cat /etc/netplan/50-
cloud-init.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
        version: 2
        ethernets:
```

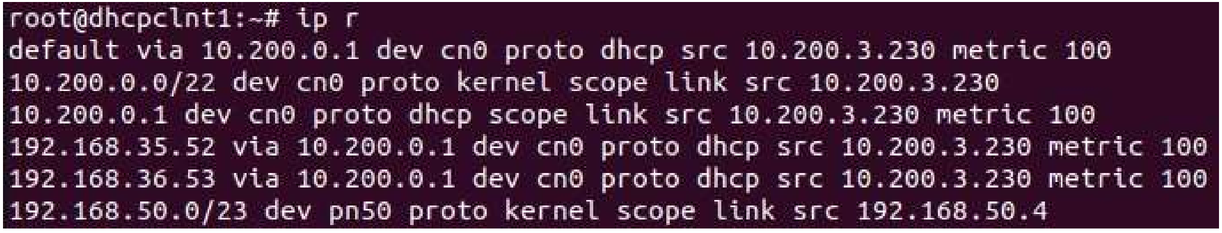
```
#       pn50: # dhcp4:
true #  addresses:
#       - 192.168.50.4/23 #
gateway4: 192.168.50.1 #
nameservers: # addresses:
#               - 192.168.35.52
#               - 192.168.36.53
#               search:
#               - cse.iith.ac.in
#               - iith.ac.in
        cn0:
                dhcp4: true
#       cn4:
#               dhcp4: true
```

https://netplan.io/examples

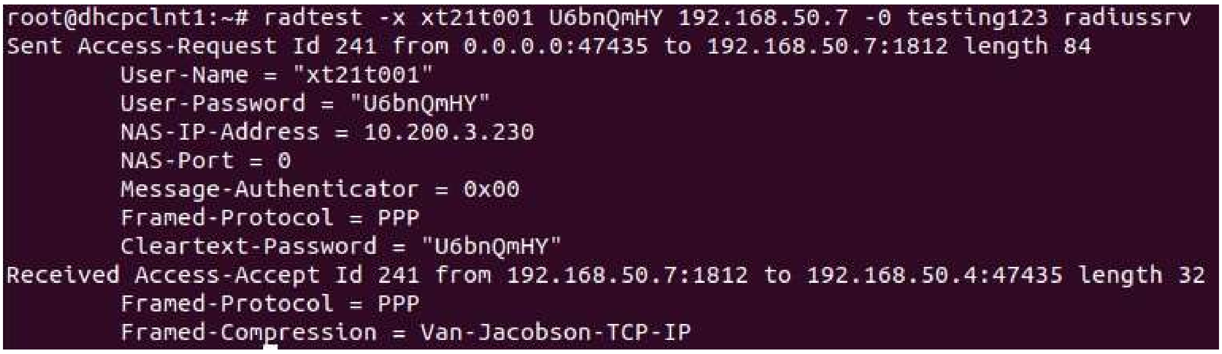
Authblock example using userid and password



```
  eno2:
      match:
          macaddress: d0:67:26:cd:29:31
      mtu: 1500
      set-name: eno2
      #addresses: [ 10.200.0.62/22 ]
      auth:
          key-management: 802.1x
          method: ttls
          identity: "admin"
          password: "password"
      dhcp4: true
      auth:
```



```
root@dhcpclnt1:~# ip r
default via 10.200.0.1 dev cn0 proto dhcp src 10.200.3.230 metric 100
10.200.0.0/22 dev cn0 proto kernel scope link src 10.200.3.230
10.200.0.1 dev cn0 proto dhcp scope link src 10.200.3.230 metric 100
192.168.35.52 via 10.200.0.1 dev cn0 proto dhcp src 10.200.3.230 metric 100
192.168.36.53 via 10.200.0.1 dev cn0 proto dhcp src 10.200.3.230 metric 100
192.168.50.0/23 dev pn50 proto kernel scope link src 192.168.50.4
```



```
root@dhcpclnt1:~# radtest -x xt21t001 U6bnQmHY 192.168.50.7 -0 testing123 radiussrv
Sent Access-Request Id 241 from 0.0.0.0:47435 to 192.168.50.7:1812 length 84
        User-Name = "xt21t001"
        User-Password = "U6bnQmHY"
        NAS-IP-Address = 10.200.3.230
        NAS-Port = 0
        Message-Authenticator = 0x00
        Framed-Protocol = PPP
        Cleartext-Password = "U6bnQmHY"
Received Access-Accept Id 241 from 192.168.50.7:1812 to 192.168.50.4:47435 length 32
        Framed-Protocol = PPP
        Framed-Compression = Van-Jacobson-TCP-IP
```

```
root@dhcpclnt1:~# radtest -x xt21t001 U6bnQmHY 192.168.50.7 -0 testing123 radiussrv
Sent Access-Request Id 152 from 0.0.0.0:55436 to 192.168.50.7:1812 length 84
        User-Name = "xt21t001"
        User-Password = "U6bnQmHY"
        NAS-IP-Address = 192.168.50.4
        NAS-Port = 0
        Message-Authenticator = 0x00
        Framed-Protocol = PPP
        Cleartext-Password = "U6bnQmHY"
Received Access-Accept Id 152 from 192.168.50.7:1812 to 192.168.50.4:55436 length 32
        Framed-Protocol = PPP
        Framed-Compression = Van-Jacobson-TCP-IP
root@dhcpclnt1:~#
```

In one terminal run tcpdump

```
$ sudo tcpdump -i eth0 -w ./radtest-success.pcap “not port ssh”
```

In another terminal run the radtest

```
ubuntu@cloudinst0a-broken:~$ cat wpa_supplicant.conf
ctrl_interface=/var/run/wpa_supplicant
eapol_version=2 ap_scan=0

network={
        # Uncomment this ssid if you want to join to a specific network
        # ssid="YOUR_SSID"
        key_mgmt=IEEE8021X
        eap=TTLS
        identity="xt21t001"
        password="U6bnQmHY"
        phase1="peapver=0"
        phase2="auth=MSCHAPV2"
        eapol_flags=0
        # Uncomment and fill in the RADIUS server details if necessary
        # radius_server=192.168.50.7
        # radius_server_port=1812
        # radius_secret="testing123"
        #anonymous_identity=""
        priority=1
}
```

Test bed

we have added second interface to both the VMs. You can first do "ssh ubuntu@192.168.51.116" (cloudinst0b) from your laptop Then from there do "ssh ubuntu@192.168.47.112" (cloudinst0a) with passwd: u!23 from the first VM. This way you can configure 802.1x with hashed password on eth0 in cloudinst0a's netplan file.

```
ubuntu@cloudinst0a:~$ ip r default via
192.168.50.1 dev eth0 proto static
192.168.46.0/23 dev ens7 proto kernel scope link src 192.168.47.112
192.168.50.0/23 dev eth0 proto kernel scope link src 192.168.51.112
```

```
ubuntu@cloudinst0b:~$ ip r default via 192.168.50.1 dev pn50 proto
static metric 100 onlink default via 10.200.0.1 dev cn0 proto dhcp src
10.200.3.53 metric 100 10.200.0.0/22 dev cn0 proto kernel scope link src
10.200.3.53 metric 100
10.200.0.1 dev cn0 proto dhcp scope link src 10.200.3.53 metric 100
192.168.35.52 via 10.200.0.1 dev cn0 proto dhcp src 10.200.3.53 metric 100
192.168.36.53 via 10.200.0.1 dev cn0 proto dhcp src 10.200.3.53 metric 100
192.168.46.0/23 dev sn46 proto kernel scope link src 192.168.47.116
192.168.50.0/23 dev pn50 proto kernel scope link src 192.168.51.116
```

Authentication using netplan on Ubuntu/Debian

Netplan file is mentioned in folder. we have modified netplan configuration file for that we have created new interface for wired as whenever we run ‘netplan apply’ so we don’t disconnect to vm . for that we are using ens7 interface. Other details u can see in netplan file.

Antiplagiarism statement

We certify that this assignment/report is our own work, based on our personal study and/or research and that we have acknowledged all material and sources used in its preparation, whether they be books, articles, packages, datasets, reports, lecture notes, and any other kind of document, electronic or personal communication. We also certify that this assignment/report has not previously been submitted for assessment/project in any other course lab, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that we have not copied in part or whole or otherwise plagiarized the work of other students and/or persons. We pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, We understand my responsibility to report honor violations by other students if we become aware of it.

Names: Ketan Sabne
Praneeth Nayak
Date: 08/05/2023
Signature: ketan
praneeth