Hackathon Soal 2 - Linux Server Hardening

Muhammad Rafif Tri Risqullah

ID: 06

Degree: S2

Soal 2

Creating users with sudo access (sudo group)

Implement a bash script for looping purposes creating new users from(range 1 - 100) with password and given pubkey in the /home/semesta/adm.pub

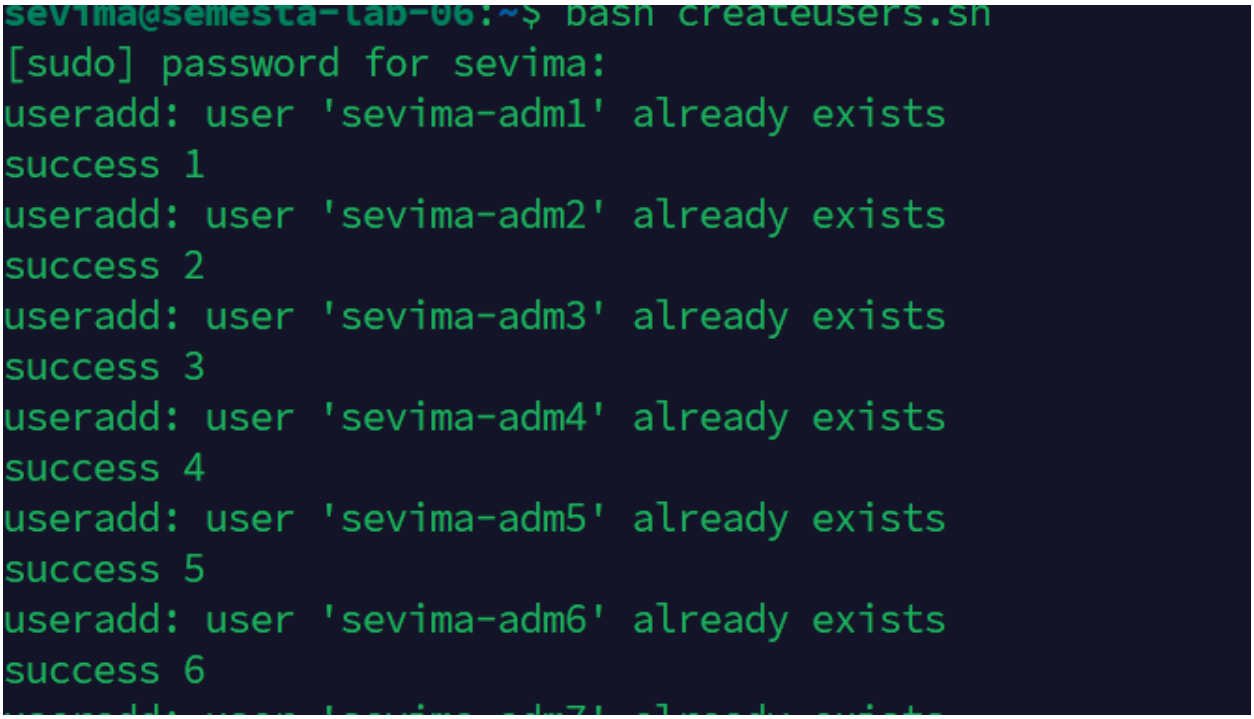
```
sevima@semesta-lab-06:~$ cat createusers.sh
#!/bin/bash

counter=1
placeholderName=sevima-adm
password=S3m3st4#2025
pubkey="/home/semesta/adm.pub"
while [ $counter -le 100 ]
do
        username="${placeholderName}${counter}"
        userhome="/home/$username"
        sudo useradd -m "$username"
        sudo usermod -aG sudo "$username"
        sudo mkdir -p "$userhome/.ssh"
        sudo chmod 700 "$userhome/.ssh"
        sudo cp /home/semesta/adm.pub $userhome/.ssh/authorized_keys
        echo "$username:$password" | sudo chpasswd
        echo "success $counter"
```

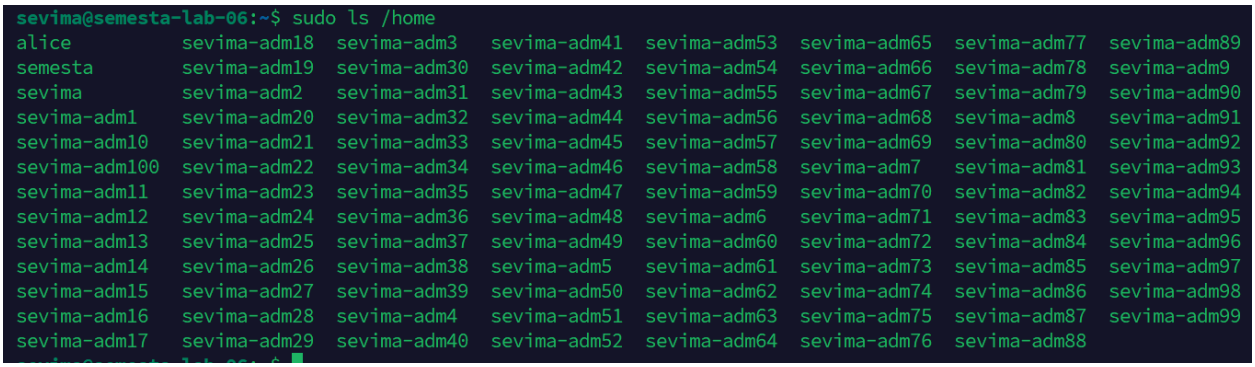
```
    ((counter++))
done

echo all done
```

Steps done consists of below

```
nano createusers.sh
--coded the file + save--
chmod +x createusers.sh
bash createusers.sh
```

Output results





Disabling password access for Root Login

Implementing PermitRootLogin prohibit-password in the /etc/ssh/sshd_config file. In this case the sshd_config is inheriting some values within sshd_config.d/50-cloud-init.conf, so we need to implement some match case for root user

```
#updated sections

# Authentication:

#LoginGraceTime 2m
PermitRootLogin prohibit-password    <-----
#StrictModes yes
MaxAuthTries 6
#MaxSessions 10

Match User root
      PasswordAuthentication no
```

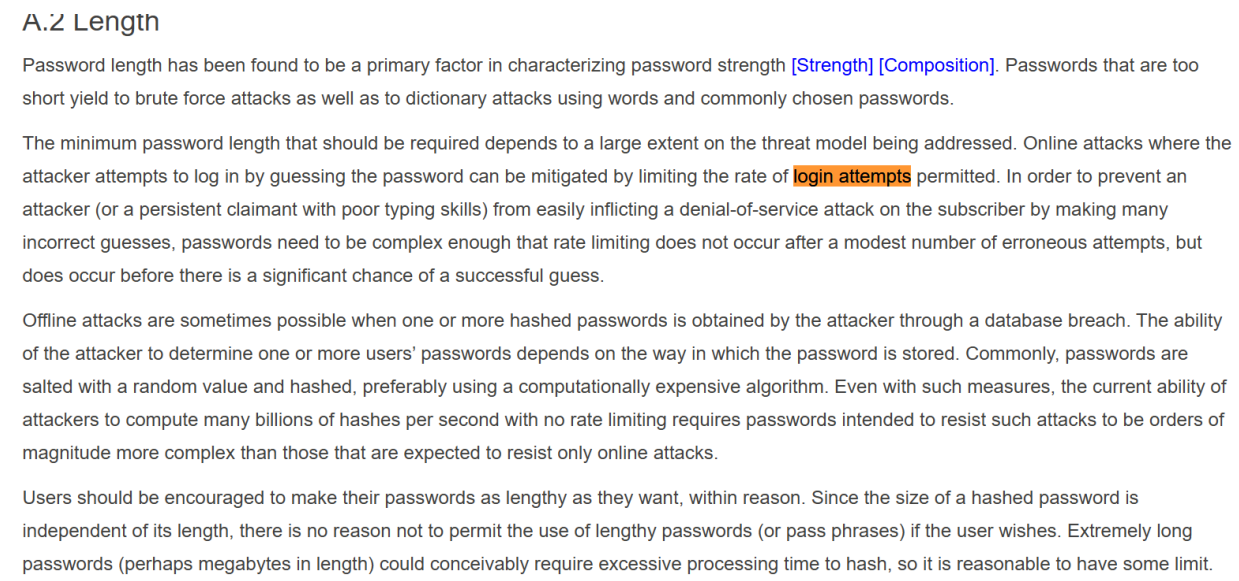
Steps

```
cd /etc/ssh
nano sshd_config
-- coded some stuff and save --
sudo systemctl restart ssh.service && sudo systemctl restart ssh.socket
```

Output

```
C:\Users\zeonk>ssh root@192.168.99.16
root@192.168.99.16: Permission denied (publickey).
```

Limiting Login Attempts (max-retry)



A.2 Length

Password length has been found to be a primary factor in characterizing password strength [Strength] [Composition]. Passwords that are too short yield to brute force attacks as well as to dictionary attacks using words and commonly chosen passwords.

The minimum password length that should be required depends to a large extent on the threat model being addressed. Online attacks where the attacker attempts to log in by guessing the password can be mitigated by limiting the rate of login attempts permitted. In order to prevent an attacker (or a persistent claimant with poor typing skills) from easily inflicting a denial-of-service attack on the subscriber by making many incorrect guesses, passwords need to be complex enough that rate limiting does not occur after a modest number of erroneous attempts, but does occur before there is a significant chance of a successful guess.

Offline attacks are sometimes possible when one or more hashed passwords is obtained by the attacker through a database breach. The ability of the attacker to determine one or more users' passwords depends on the way in which the password is stored. Commonly, passwords are salted with a random value and hashed, preferably using a computationally expensive algorithm. Even with such measures, the current ability of attackers to compute many billions of hashes per second with no rate limiting requires passwords intended to resist such attacks to be orders of magnitude more complex than those that are expected to resist only online attacks.

Users should be encouraged to make their passwords as lengthy as they want, within reason. Since the size of a hashed password is independent of its length, there is no reason not to permit the use of lengthy passwords (or pass phrases) if the user wishes. Extremely long passwords (perhaps megabytes in length) could conceivably require excessive processing time to hash, so it is reasonable to have some limit.

cited from https://pages.nist.gov/800-63-3/sp800-63b.html

In this case I am implementing 6 attempts max in the sshd_config file (default from the config file)

```
#updated sections

# Authentication:

#LoginGraceTime 2m
PermitRootLogin prohibit-password
#StrictModes yes
MaxAuthTries 6      <----
#MaxSessions 10
```

```
Match User root
        PasswordAuthentication no
```

Steps

```
cd /etc/ssh
nano sshd_config
-- coded some stuff and save --
sudo systemctl restart ssh.service && sudo systemctl restart ssh.socket
```

Output

```
C:\Users\zeonk>ssh sevima-adm1@192.168.99.16
sevima-adm1@192.168.99.16's password:
Permission denied, please try again.
sevima-adm1@192.168.99.16's password:
Permission denied, please try again.
sevima-adm1@192.168.99.16's password:
sevima-adm1@192.168.99.16: Permission denied (publickey,password).
```

```
2025-07-19T06:36:11.228582+00:00 hachathon-lab-main sudo: pam_unix(sudo:
2025-07-19T06:36:18.544013+00:00 hachathon-lab-main sshd[18703]: Failed pa
2025-07-19T06:36:20.347378+00:00 hachathon-lab-main sshd[18703]: messag
2025-07-19T06:36:20.372880+00:00 hachathon-lab-main sshd[18703]: Connec
```

Enabling firewall

Step

```
sudo ufw status
sudo ufw enable
sudo ufw allow 22/tcp ⟵ current server purpose only for SSH. Not needed fo
```

Output

```
sevima@semesta-lab-06:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
22/tcp (v6)                ALLOW       Anywhere (v6)
```

LVM setup from block storage sdb with encryption enabled

using LUKS - Linux Unified Key Setup for disk encryption.

Step

```
lsblk
sudo cryptsetup luksFormat /dev/sdb
sudo cryptsetup luksDump /dev/sdb
sudo cryptsetup luksOpen /dev/sdb encrypted
sudo dmsetup info encrypted # info for the encrypted LUKS part
sudo pvcreate /dev/mapper/encrypted # physical volume creation
sudo vgcreate hackathon-syadm7 /dev/mapper/encrypted # volume group creat
sudo lvcreate -L 1.5G -n nfs-hackathon-syadm7 hackathon-syadm7 # logical vol
sudo mkfs.ext4 /dev/hackathon-syadm7/nfs-hackathon-syadm7 # format to ext4
sudo lvdisplay /dev/hackathon-syadm7/nfs-hackathon-syadm7 # lv info
cd /mnt && sudo mkdir nfs-hackathon
sudo apt update && sudo apt install nfs-common -y
sudo mount -t nfs 192.168.99.3:nfs-semesta7 /mnt/nfs-hackathon # mount the nf
df -h /mnt/nfs-hackathon
```

Output

```
sevima@semesta-lab-06:/mnt$ sudo mount -t nfs 192.168.99.3:/nfs-semesta7 /mnt/nfs-hackathon
sevima@semesta-lab-06:/mnt$ cd nfs-hackathon/
sevima@semesta-lab-06:/mnt/nfs-hackathon$ df -h
Filesystem                         Size  Used Avail Use% Mounted on
tmpfs                              392M  1.1M  391M   1% /run
/dev/mapper/ubuntu--vg-ubuntu--lv   24G  7.0G   16G  32% /
tmpfs                              2.0G     0  2.0G   0% /dev/shm
tmpfs                              5.0M     0  5.0M   0% /run/lock
/dev/sda2                          2.0G  100M  1.7G   6% /boot
tmpfs                              392M   12K  392M   1% /run/user/1000
192.168.99.3:/nfs-semesta7          24G  6.4G   16G  29% /mnt/nfs-hackathon
sevima@semesta-lab-06:/mnt/nfs-hackathon$ df -h /mnt/nfs-hackathon
Filesystem                 Size  Used Avail Use% Mounted on
192.168.99.3:/nfs-semesta7   24G  6.4G   16G  29% /mnt/nfs-hackathon
sevima@semesta-lab-06:/mnt/nfs-hackathon$ ls
achmadalifnasrulloh-192.168.99.11  test1
sevima@semesta-lab-06:/mnt/nfs-hackathon$ cd achmadalifnasrulloh-192.168.99.11/
sevima@semesta-lab-06:/mnt/nfs-hackathon/achmadalifnasrulloh-192.168.99.11$ ls
sevima@semesta-lab-06:/mnt/nfs-hackathon/achmadalifnasrulloh-192.168.99.11$ cd ..
sevima@semesta-lab-06:/mnt/nfs-hackathon$ cat test1
sevima@semesta-lab-06:/mnt/nfs-hackathon$
```

LV info

```
sevima@semesta-lab-06:~$ sudo lvdisplay /dev/hackathon-syadm7/nfs-hackathon-syadm7
  --- Logical volume ---
  LV Path                /dev/hackathon-syadm7/nfs-hackathon-syadm7
  LV Name                nfs-hackathon-syadm7
  VG Name                hackathon-syadm7
  LV UUID                S7lO70-VnEG-5s2e-mNSF-uLrN-6Sgq-11rseN
  LV Write Access        read/write
  LV Creation host, time semesta-lab-06, 2025-07-19 07:23:35 +0000
  LV Status              available
  # open                 0
  LV Size                1.50 GiB
  Current LE             384
  Segments               1
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           252:2
```

lsblk result

```
NAME                                    MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINTS
sda                                       8:0    0   50G  0 disk
├─sda1                                    8:1    0    1M  0 part
├─sda2                                    8:2    0    2G  0 part  /boot
└─sda3                                    8:3    0   48G  0 part
  └─ubuntu--vg-ubuntu--lv               252:0    0   24G  0 lvm   /
sdb                                       8:16   0    2G  0 disk
└─encrypted                             252:1    0    2G  0 crypt
  └─hackathon--syadm7-nfs--hackathon--syadm7 252:2    0  1.5G  0 lvm
sdc                                       8:32   0    2G  0 disk
sr0                                      11:0    1    3G  0 rom
```