Google Cloud Sprint Bootcamp

Team Members

1. Alok Bhatt
2. Vibhuti Talekar
3. M M Kishore.
4. Pragya Kimothi
5. Sapna Bisht

Assignment (Week -01)

1. Create an empty file named "gcp_boot_camp" without using touch command or any editor tool(vim, nano etc.)

Solution:

We can create file using 3 different methods without using touch command or any editor tool are as follows : -

1. **cat > gcp_boot_camp**
   Press ctrl+d

```
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ cat > gcp_boot_camp
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ ls
gcp_boot_camp
```

2. **echo “” >gcp_boot_camp**

```
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ echo "" > gcp_boot_camp
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ ls
gcp_boot_camp
```

3. **$null >> gcp_boot_camp**

```
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ $null >> gcp_boot_camp
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ ls
gcp_boot_camp
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$
```

2. Create a user who should be able to ssh to instance and should be able sudo to root.

1. **#adduser user1**
2. **#usermod -aG sudo user1**
3. **#sudo su - user1**

```
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ adduser newuser
adduser: Only root may add a user or group to the system.
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ sudo adduser newuser
[sudo] password for wolverine:
Adding user `newuser' ...
Adding new group `newuser' (1001) ...
Adding new user `newuser' (1002) with group `newuser' ...
Creating home directory `/home/newuser' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
No password supplied
New password:
Retype new password:
No password supplied
New password:
Retype new password:
No password supplied
passwd: Authentication token manipulation error
passwd: password unchanged
Try again? [y/N] y
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for newuser
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ usermod -aG sudo newuser
usermod: Permission denied.
usermod: cannot lock /etc/passwd; try again later.
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$ sudo usermod -aG sudo newu
ser
wolverine@wolverine-HP-Laptop-15-da0077tx:~/Documents/Google/cloud_sprint-22$
```

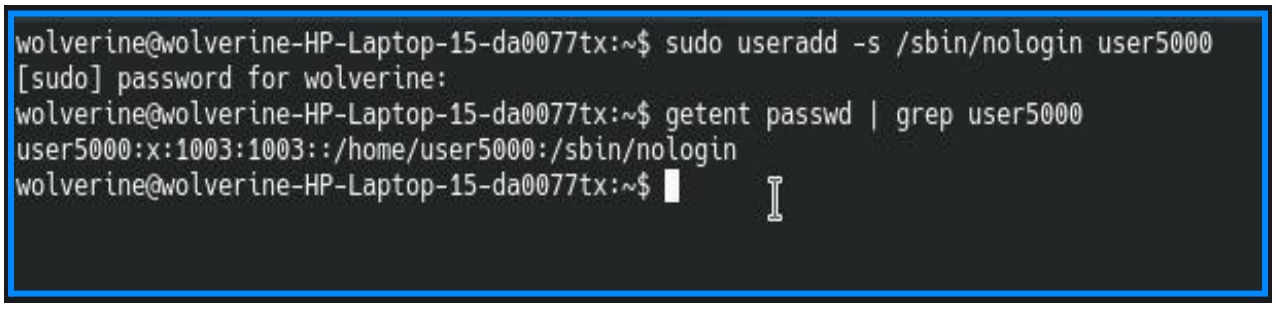
3. What is /sbin/nologin and what it is used for ?

By default, when we add a new Linux user to the system. The system grants shell access to the user. As soon as we add a user, they can log in via ssh or console. The best way to put Linux shell access restrictions is to use a unique shell called nologin. The nologin politely refuses a login attempt . It displays a message that an account is not available and exits non-zero. It is intended as a replacement shell field for system accounts to deny access.

If we set shell to **/sbin/nolgin** user cannot ssh into the server.It is used by system services that need an account but do not want to create security related issues by granting them login access.

nologin displays a message that an account is not available and exits non -zero.
If the **file /etc/nologin.txt** exits , nologin displays it's contents to the user instead of the default message.

You can use it as follows:
 # usermod -s/sbin/nologin userName



4. Write a shell script to extract the ip address from a file and the count of similar ip-address.

Steps:
1.Run the script using ./filename.sh
2.If you do not have the permission to execute the file use chmod +x filename.sh to change the permission to execute
3.You will be prompted to enter the input file name from where you wish to count the Ip addresses.

Code

```
#!/bin/bash

#Input for the file you wish to count Ip Addresses from

echo "Enter the filename where Ip addresses are stored"
read file

#Command used to find and count the Ip Addresses
value="$(grep  -E -o '[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}' $file|sort|uniq -c)"

if [[ -z "$value" ]]
then
  echo "No IP addresses found in the File."
else
  echo '  Count|IpAddresses'
  echo "$value"
```

```
vibhutitalekar@vibhutitalekar-Lenovo-E41-15:~$ ./q4Ip.sh
bash: ./q4Ip.sh: Permission denied
vibhutitalekar@vibhutitalekar-Lenovo-E41-15:~$ chmod +x q4Ip.sh
vibhutitalekar@vibhutitalekar-Lenovo-E41-15:~$ ./q4Ip.sh
Enter the filename where Ip addresses are stored
logfile
  Count|IpAddresses
      2 123.125.114.144
      1 173.194.126.213
      1 173.194.126.214
      1 173.252.110.27
      1 199.59.148.82
      1 199.59.150.39
      1 199.59.150.7
      2 220.181.111.85
      2 220.181.111.86
vibhutitalekar@vibhutitalekar-Lenovo-E41-15:~$ ./q4Ip.sh
Enter the filename where Ip addresses are stored
emptyfile.txt
No IP addresses found in the File.
```

Logfile Content:

logfile

home > vibhutitalekar > logfile

```
1  #gmail.com
2  173.194.126.214
3  173.194.126.213
4  #twitter.com
5  199.59.150.39
6  199.59.148.82
7  199.59.150.7
8  #facebook.com
9  173.252.110.27
10 #baidu.com
11 220.181.111.86
12 123.125.114.144
13 220.181.111.85
14 #baidu.com
15 220.181.111.86
16 123.125.114.144
17 220.181.111.85
18
```

emptyfile.txt

home > vibhutitalekar > emptyfile.txt

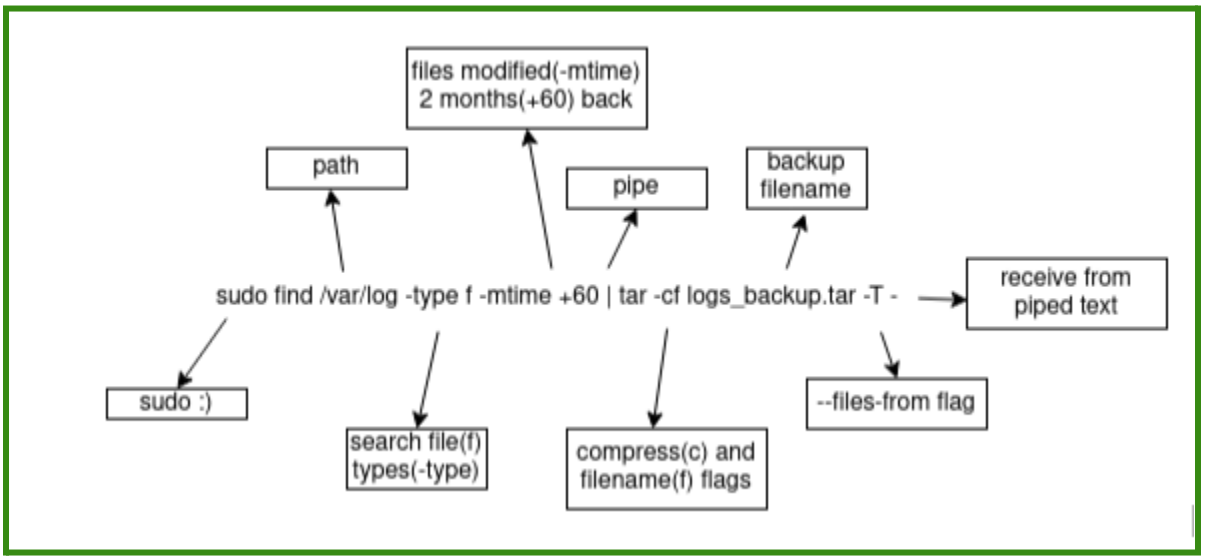
```
1  nkdpdofdj
2  0000000
3  122222
4  3333333
5  22018111185
6
```

5. Write a script to find files older than 2 months in /var/log/ directory and archive them.

Steps :-

- Login to root.
- Run the script using **"sh logBackup.sh"** command.
- **"logs_backup"** file is created in the current working directory.

Explanation



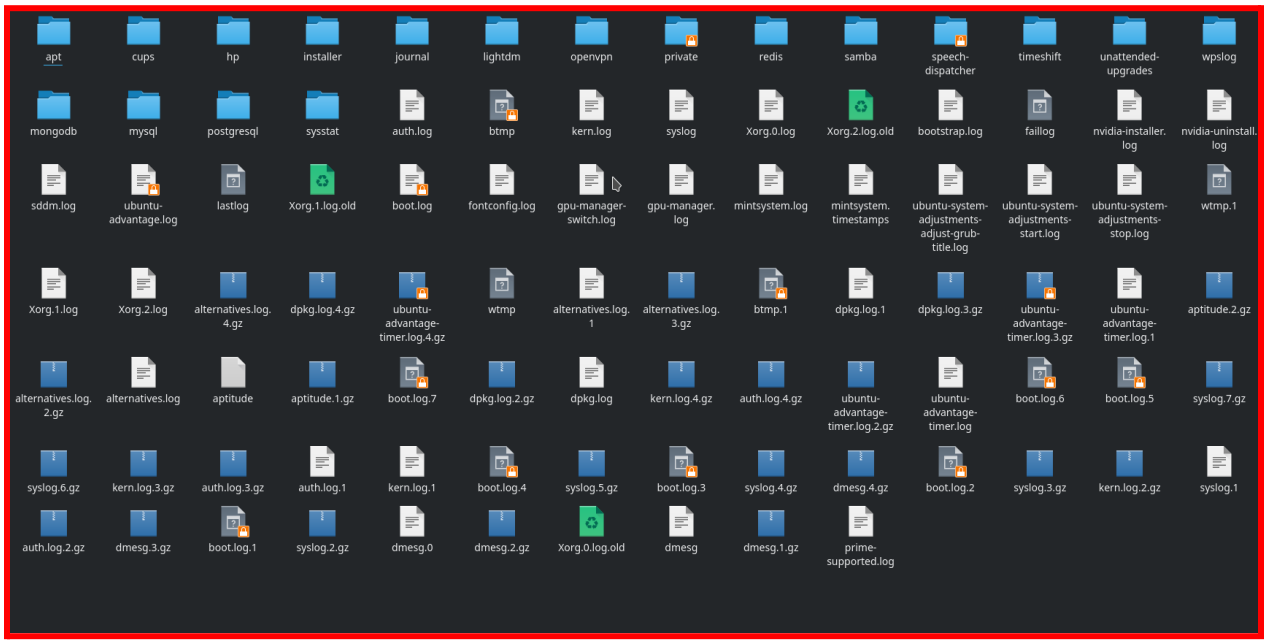
Code :-

```
#!/bin/bash
#file name - logBackup.sh

echo '   Archiving Logs'

value="$(sudo find /var/log -type f -mtime +60 | tar -cvf logs_backup.tar -T -)"

echo "backup saved"
```

Output :-

```
root@wolverine-HP-Laptop-15-da0077tx:/home/wolverine/Documents
/Google/cloud_sprint-22# sh logBackup.sh
     Archiving Logs
tar: Removing leading `/' from member names
backup created
root@wolverine-HP-Laptop-15-da0077tx:/home/wolverine/Documents/Google/cloud_sprint-22# ls
'Coursera MQNDBHU3ATXJ.pdf'   gcp_boot_camp   logfile          'New Folder'
'Coursera Z95C4YGB86ZE.pdf'   logBackup.sh    logs_backup.tar   test.sh
```

All Log Files in the machine



Files present in the backup file.