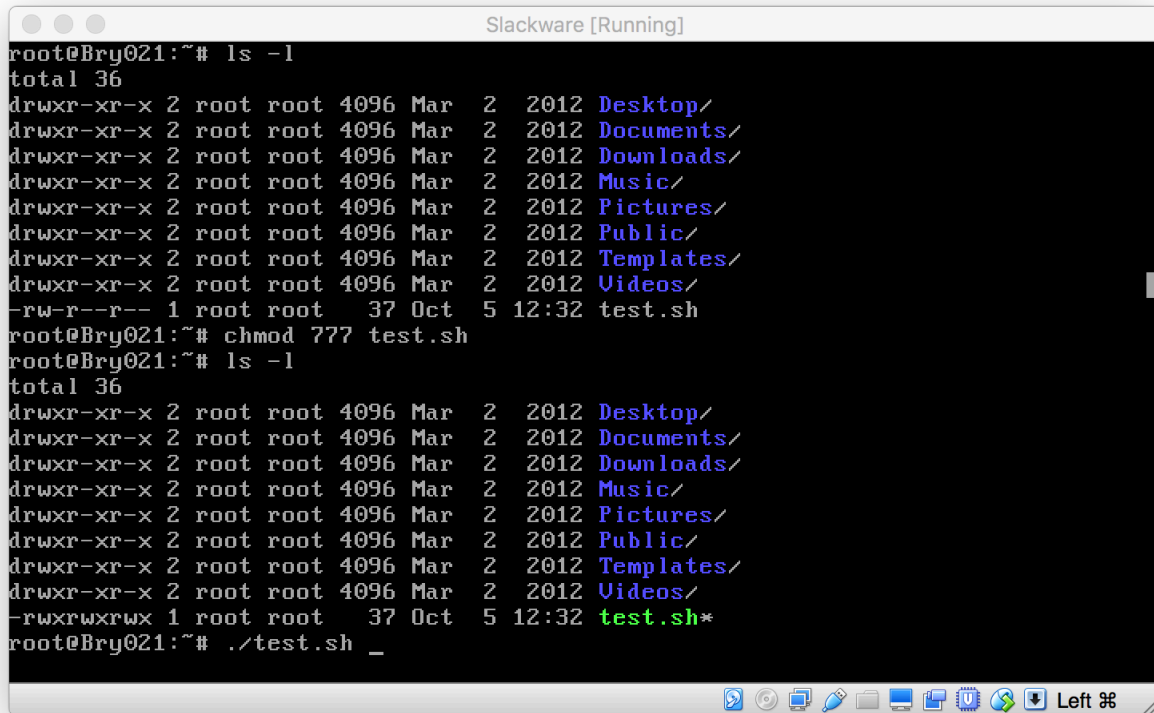


Lab 3 : Linux systems and shell scripting

Exercise 1. Creating an executable bash script.

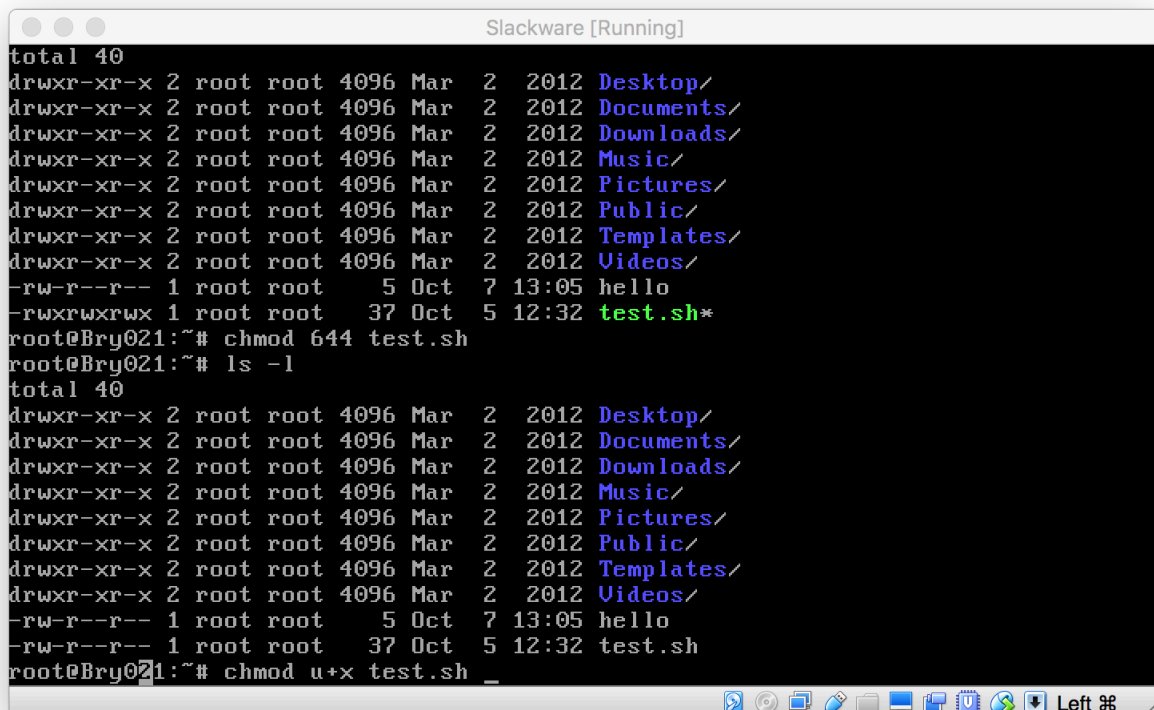
1.1. Create test.sh in your home directory. Make sure that the script is executable.

When a file is first created, group and other members can only read it (rw-r--r--).



```
Slackware [Running]
root@Bry021:~# ls -l
total 36
drwxr-xr-x 2 root root 4096 Mar  2  2012 Desktop/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Documents/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Downloads/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Music/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Pictures/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Public/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Templates/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Videos/
-rw-r--r-- 1 root root  37 Oct  5 12:32 test.sh
root@Bry021:~# chmod 777 test.sh
root@Bry021:~# ls -l
total 36
drwxr-xr-x 2 root root 4096 Mar  2  2012 Desktop/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Documents/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Downloads/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Music/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Pictures/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Public/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Templates/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Videos/
-rwxrwxrwx 1 root root  37 Oct  5 12:32 test.sh*
root@Bry021:~# ./test.sh _
```

1.2. An student execute “chmod 777”. Explain why giving those permissions is a bad idea.

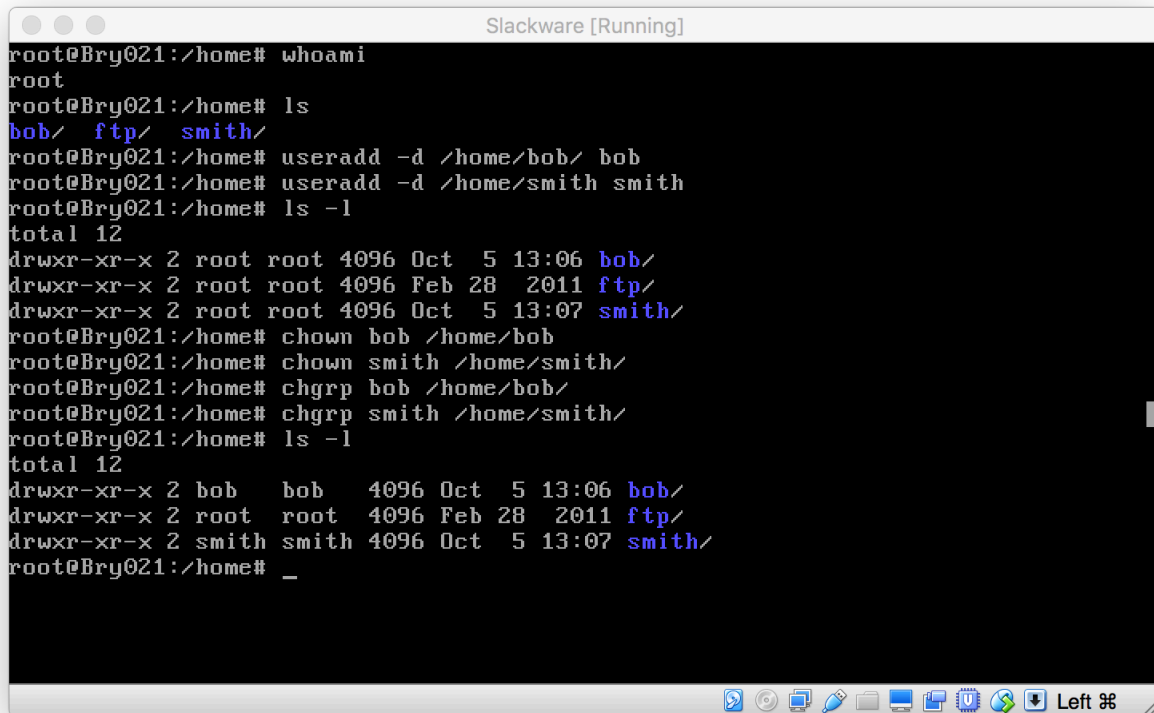


```
Slackware [Running]
total 40
drwxr-xr-x 2 root root 4096 Mar  2  2012 Desktop/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Documents/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Downloads/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Music/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Pictures/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Public/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Templates/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Videos/
-rw-r--r-- 1 root root  5 Oct  7 13:05 hello
-rwxrwxrwx 1 root root  37 Oct  5 12:32 test.sh*
root@Bry021:~# chmod 644 test.sh
root@Bry021:~# ls -l
total 40
drwxr-xr-x 2 root root 4096 Mar  2  2012 Desktop/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Documents/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Downloads/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Music/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Pictures/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Public/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Templates/
drwxr-xr-x 2 root root 4096 Mar  2  2012 Videos/
-rw-r--r-- 1 root root  5 Oct  7 13:05 hello
-rw-r--r-- 1 root root  37 Oct  5 12:32 test.sh
root@Bry021:~# chmod u+x test.sh _
```

The command “**chmod 777**” allows anybody not only execute the file but also write or delete it (rwx.rwx.rwx). This is dangerous as the script should be only modified or executed by its owner.

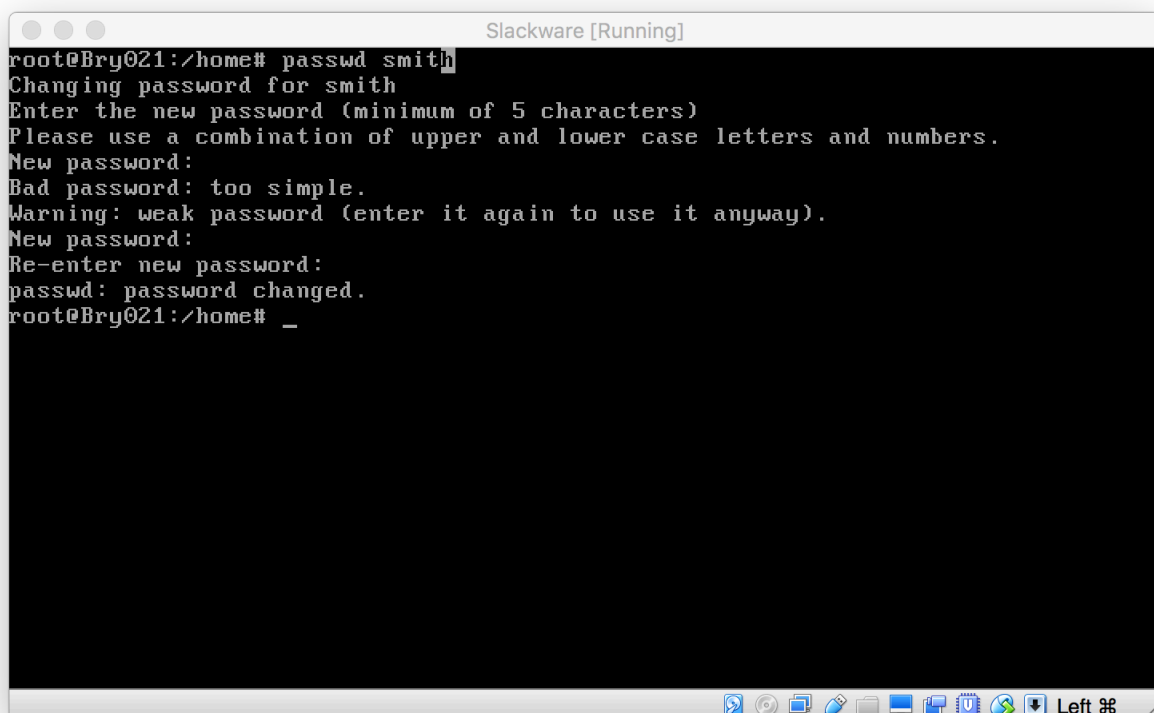
In conclusion, the right option would be “**chmod u+x**”.

Exercise 2. Create new users



```
Slackware [Running]
root@Bry021:/home# whoami
root
root@Bry021:/home# ls
bob/  ftp/  smith/
root@Bry021:/home# useradd -d /home/bob/ bob
root@Bry021:/home# useradd -d /home/smith smith
root@Bry021:/home# ls -l
total 12
drwxr-xr-x 2 root root 4096 Oct  5 13:06 bob/
drwxr-xr-x 2 root root 4096 Feb 28 2011 ftp/
drwxr-xr-x 2 root root 4096 Oct  5 13:07 smith/
root@Bry021:/home# chown bob /home/bob
root@Bry021:/home# chown smith /home/smith/
root@Bry021:/home# chgrp bob /home/bob/
root@Bry021:/home# chgrp smith /home/smith/
root@Bry021:/home# ls -l
total 12
drwxr-xr-x 2 bob  bob  4096 Oct  5 13:06 bob/
drwxr-xr-x 2 root root 4096 Feb 28 2011 ftp/
drwxr-xr-x 2 smith smith 4096 Oct  5 13:07 smith/
root@Bry021:/home# _
```

The commands **useradd** and **passwd** are required to manage new user accounts. Let's create the password for smith (smith):

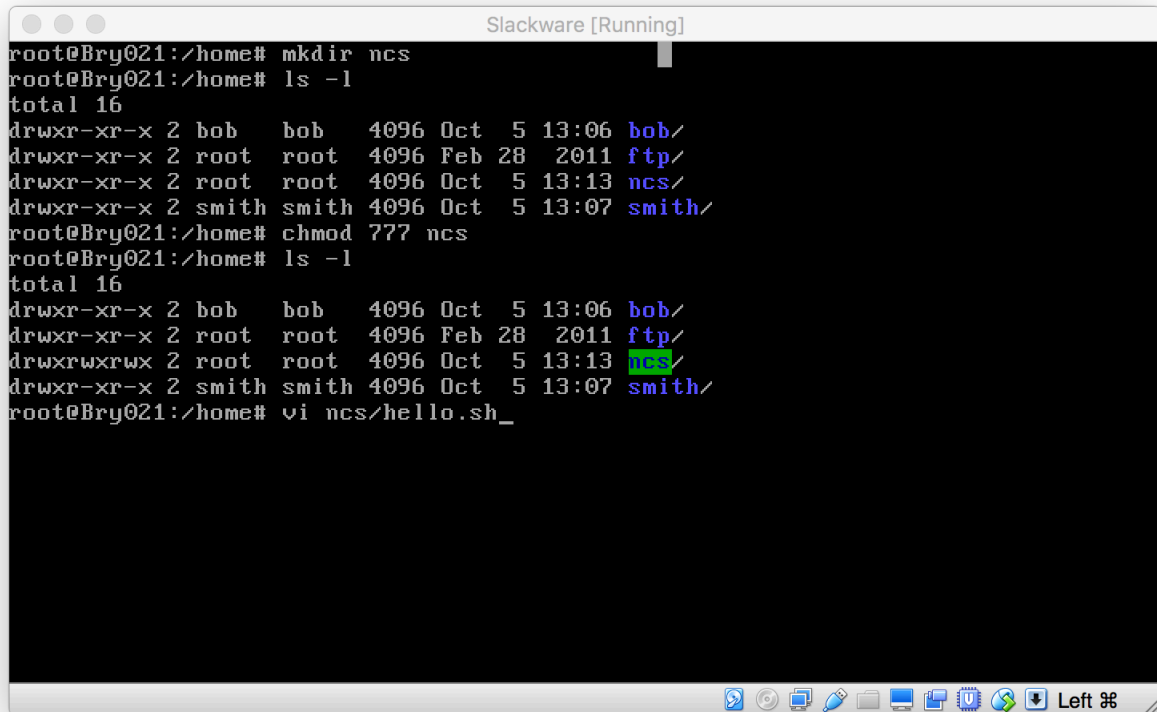


```
Slackware [Running]
root@Bry021:/home# passwd smith
Changing password for smith
Enter the new password (minimum of 5 characters)
Please use a combination of upper and lower case letters and numbers.
New password:
Bad password: too simple.
Warning: weak password (enter it again to use it anyway).
New password:
Re-enter new password:
passwd: password changed.
root@Bry021:/home# _
```

Exercise 3. Create a shared executable script

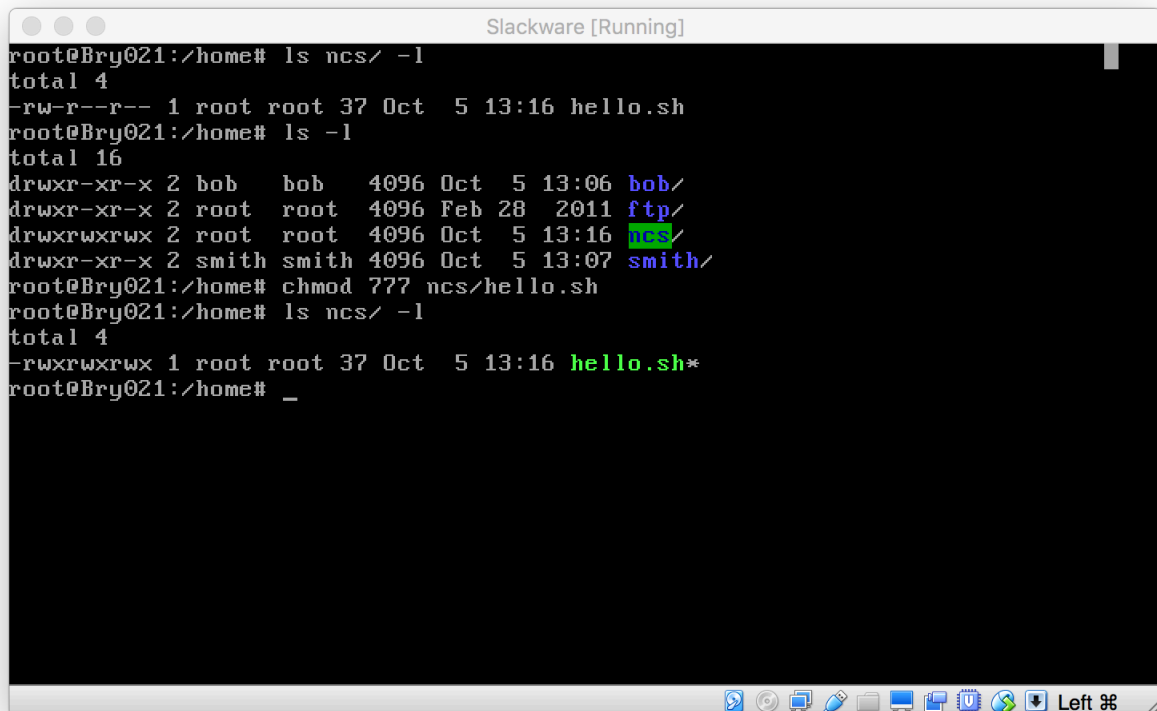
3.1. Create a public directory /home/ncs and a script called "hello.sh" to print a message.

A file has been created in a public folder /home/ncs/ but apparently it doesn't inherit the permissions given to the folder (644 by default).



```
Slackware [Running]
root@Bry021:/home# mkdir ncs
root@Bry021:/home# ls -l
total 16
drwxr-xr-x 2 bob   bob   4096 Oct  5 13:06 bob/
drwxr-xr-x 2 root  root  4096 Feb 28 2011 ftp/
drwxr-xr-x 2 root  root  4096 Oct  5 13:13 ncs/
drwxr-xr-x 2 smith smith 4096 Oct  5 13:07 smith/
root@Bry021:/home# chmod 777 ncs
root@Bry021:/home# ls -l
total 16
drwxr-xr-x 2 bob   bob   4096 Oct  5 13:06 bob/
drwxr-xr-x 2 root  root  4096 Feb 28 2011 ftp/
drwxrwxrwx 2 root  root  4096 Oct  5 13:13 ncs/
drwxr-xr-x 2 smith smith 4096 Oct  5 13:07 smith/
root@Bry021:/home# vi ncs/hello.sh_
```

In this case we should change the permission to 777 as the folder /home/ncs/ is a public folder.



```
Slackware [Running]
root@Bry021:/home# ls ncs/ -l
total 4
-rw-r--r-- 1 root root 37 Oct  5 13:16 hello.sh
root@Bry021:/home# ls -l
total 16
drwxr-xr-x 2 bob   bob   4096 Oct  5 13:06 bob/
drwxr-xr-x 2 root  root  4096 Feb 28 2011 ftp/
drwxrwxrwx 2 root  root  4096 Oct  5 13:16 ncs/
drwxr-xr-x 2 smith smith 4096 Oct  5 13:07 smith/
root@Bry021:/home# chmod 777 ncs/hello.sh
root@Bry021:/home# ls ncs/ -l
total 4
-rwxrwxrwx 1 root root 37 Oct  5 13:16 hello.sh*
root@Bry021:/home# _
```

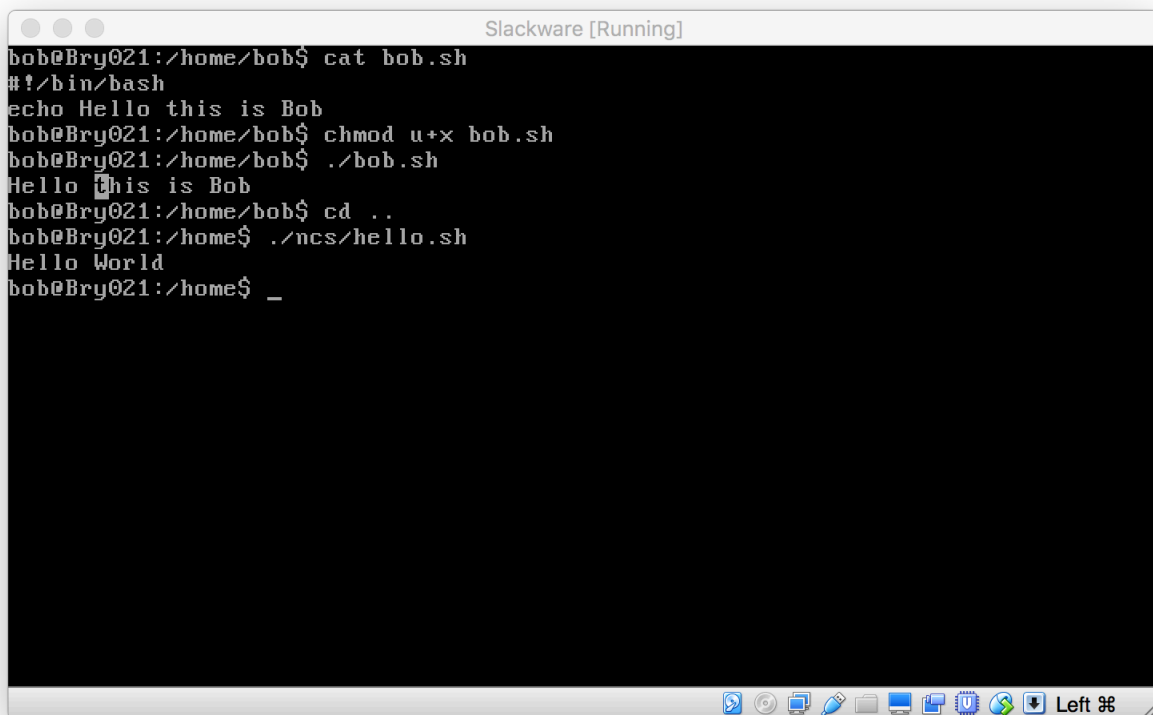
3.2. Execute the script and note down the owner/group ownership and the file permissions of his script

The permissions of this script have changed from 644 (rw-r--r--) to 777 (rwx.rwx.rwx). Now not only the owner could modify it but also everybody else in the group and outside the group.

Exercise 4. Accessing files form different user accounts

a) Log in as bob in terminal TTY2

4.1. Execute ./bob/bob.sh and ./ncs/hello.sh.



```
bob@Bry021:/home/bob$ cat bob.sh
#!/bin/bash
echo Hello this is Bob
bob@Bry021:/home/bob$ chmod u+x bob.sh
bob@Bry021:/home/bob$ ./bob.sh
Hello this is Bob
bob@Bry021:/home/bob$ cd ..
bob@Bry021:/home$ ./ncs/hello.sh
Hello World
bob@Bry021:/home$ _
```

Bob can run his script as the permission has been modified to “u+x”. The script in the public folder /home/ncs/ is also accessible for him.

b) Log in as smith in terminal TTY3

4.2. Execute ./bob/bob.sh and ./ncs/hello.sh. Explain the results you get.

The script hello.sh does work (public folder) but not bob.sh (permission denied) as smith is not bob, obviously.

```
Slackware [Running]
Hello World
smith@Bry021:/home$ ls bob/ -l
total 4
-rwxr--r-- 1 bob bob 37 Oct  7 12:53 bob.sh*
smith@Bry021:/home$ ./bob/bob.sh
-bash: ./bob/bob.sh: Permission denied
smith@Bry021:/home$ _
```

A possible solution would be to make the script accessible to the group members and include smith in the same group as bob. Actually, a similar solution is presented below (exercise 5).

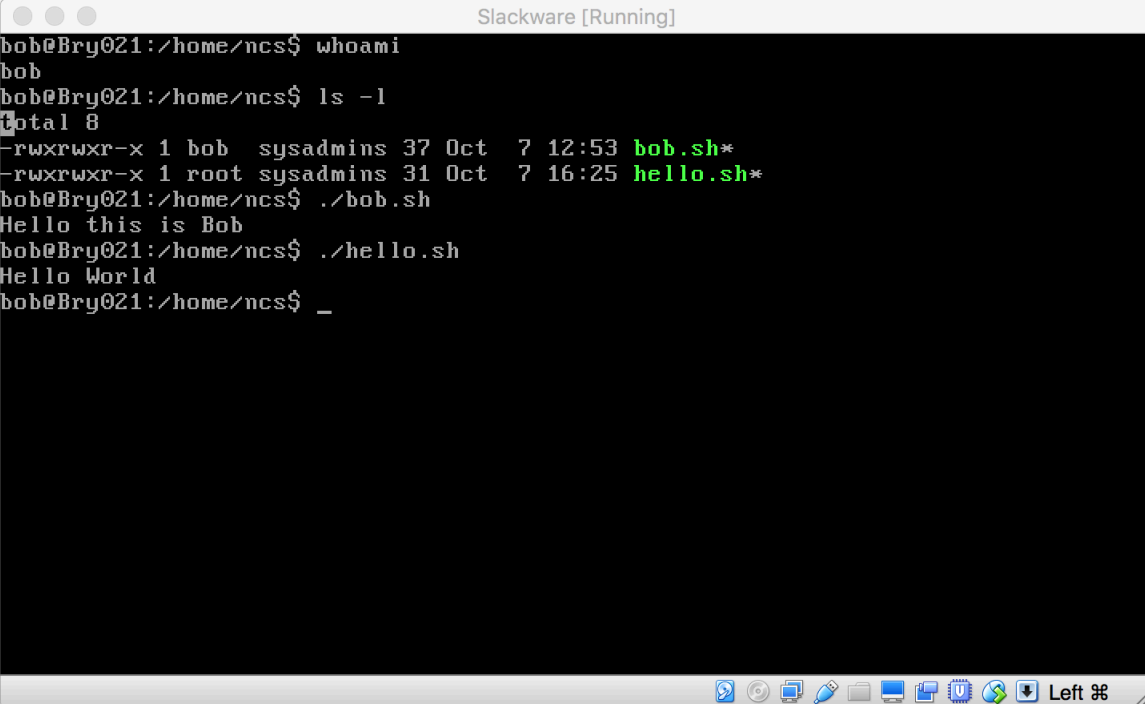
Exercise 5 (optional)

5.1. Create a group (sysadmins) and add bob and smith as members.

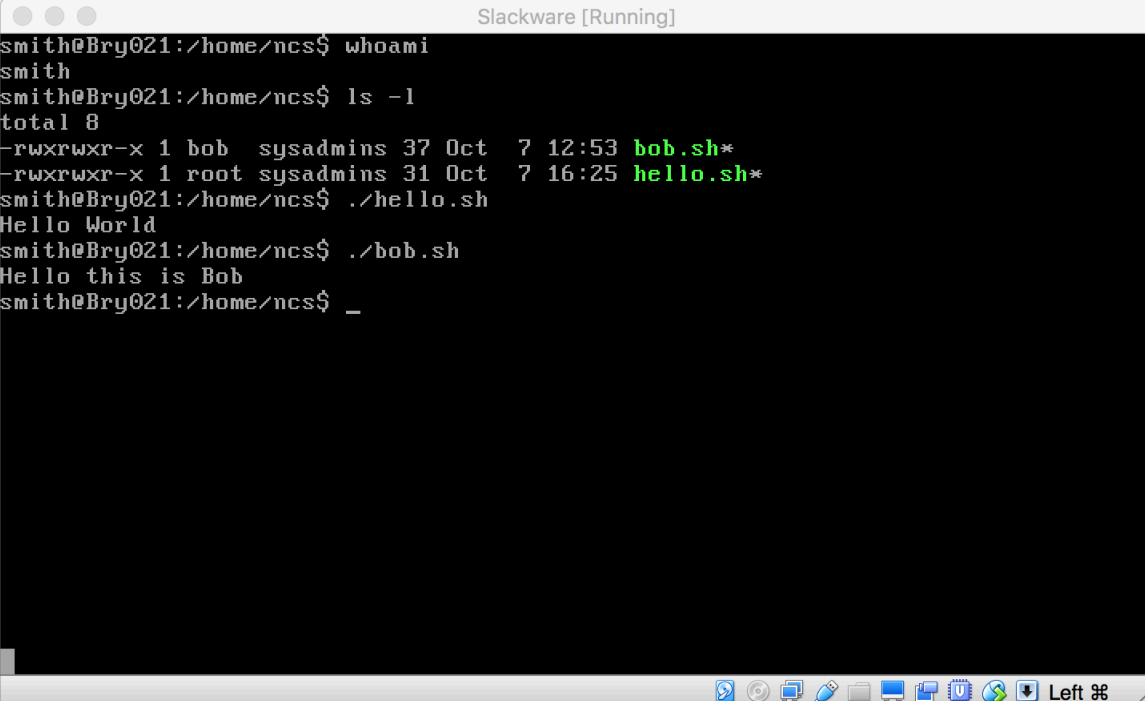
The command **groupadd** adds a new group in the system. To change the owner of a file, the command **chown** is used but to change the group owner, the command **chgrp** is required.

```
Slackware [Running]
root@Bry021:~# groupadd sysadmins
root@Bry021:~# chown bob:sysadmin /home/ncs/
chown: invalid group: 'bob:sysadmin'
root@Bry021:~# chown bob:sysadmins /home/ncs/
root@Bry021:~# chown smith:sysadmins /home/ncs/
root@Bry021:~#
root@Bry021:~# cd /home/
root@Bry021:/home# ls
bob/  ftp/  ncs/  smith/
root@Bry021:/home# chown sysadmins ncs/
chown: invalid user: 'sysadmins'
root@Bry021:/home# chgrp sysadmins ncs/
root@Bry021:/home# cd ncd
-bash: cd: ncd: No such file or directory
root@Bry021:/home# cd ncs
root@Bry021:/home/ncs# ls
hello.sh*
root@Bry021:/home/ncs# chgrp sysadmins hello.sh
root@Bry021:/home/ncs# ls -l
total 4
-rwxrwxrwx 1 root sysadmins 37 Oct  5 13:16 hello.sh*
root@Bry021:/home/ncs# chmod 775 hello.sh
root@Bry021:/home/ncs# _
```

For bob and smith it is possible to run both scripts as expected. Notice that the permission have been updated to 775 (rwx.r-x.r-x) as it didn't work for 754 (rwx.r-x.r—)

A terminal window titled "Slackware [Running]" showing a user named bob at the prompt bob@Bry021:/home/ncs\$. The user runs 'whoami' and gets 'bob'. Then they run 'ls -l' and see a directory listing with permissions -rwxrwxr-x for files bob.sh* and hello.sh*. They then run './bob.sh' which outputs 'Hello this is Bob', and './hello.sh' which outputs 'Hello World'. The prompt returns to bob@Bry021:/home/ncs\$.

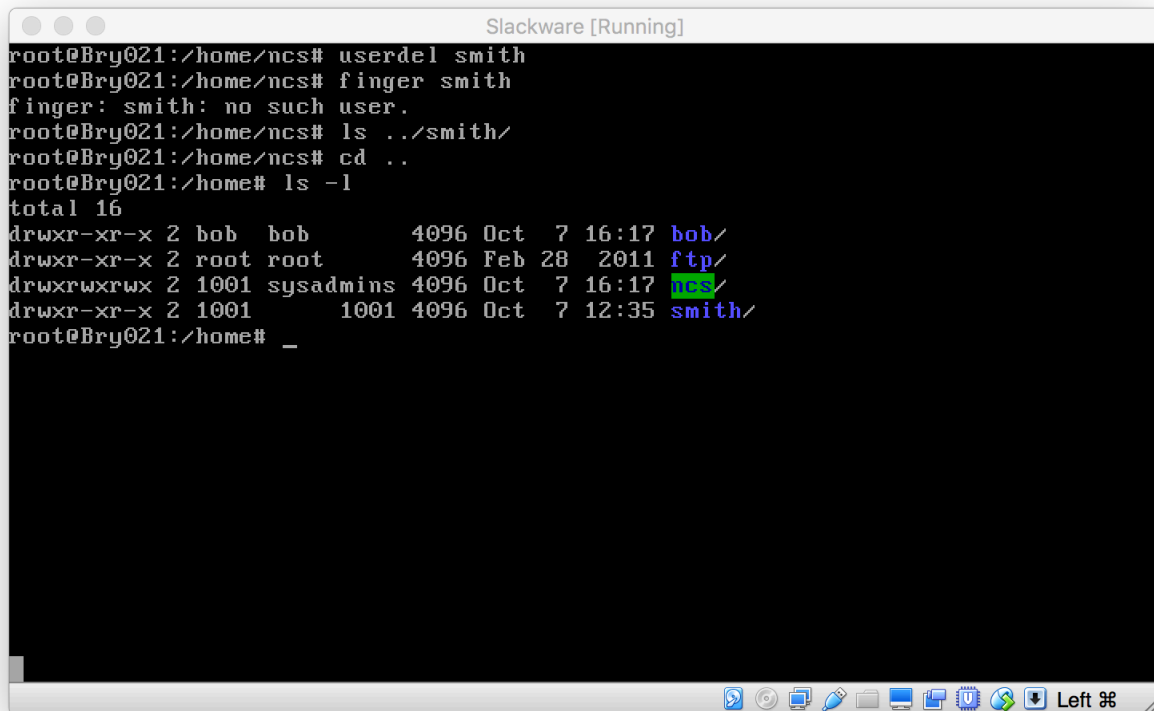
```
Slackware [Running]
bob@Bry021:/home/ncs$ whoami
bob
bob@Bry021:/home/ncs$ ls -l
total 8
-rwxrwxr-x 1 bob sysadmins 37 Oct  7 12:53 bob.sh*
-rwxrwxr-x 1 root sysadmins 31 Oct  7 16:25 hello.sh*
bob@Bry021:/home/ncs$ ./bob.sh
Hello this is Bob
bob@Bry021:/home/ncs$ ./hello.sh
Hello World
bob@Bry021:/home/ncs$ _
```

A terminal window titled "Slackware [Running]" showing a user named smith at the prompt smith@Bry021:/home/ncs\$. The user runs 'whoami' and gets 'smith'. Then they run 'ls -l' and see the same directory listing as bob. They then run './hello.sh' which outputs 'Hello World', and './bob.sh' which outputs 'Hello this is Bob'. The prompt returns to smith@Bry021:/home/ncs\$.

```
Slackware [Running]
smith@Bry021:/home/ncs$ whoami
smith
smith@Bry021:/home/ncs$ ls -l
total 8
-rwxrwxr-x 1 bob sysadmins 37 Oct  7 12:53 bob.sh*
-rwxrwxr-x 1 root sysadmins 31 Oct  7 16:25 hello.sh*
smith@Bry021:/home/ncs$ ./hello.sh
Hello World
smith@Bry021:/home/ncs$ ./bob.sh
Hello this is Bob
smith@Bry021:/home/ncs$ _
```

5.2. Disable smith's user account

The command **userdel** only deletes users login details but not his home directory (1001:1001)



```
Slackware [Running]
root@Bry021:/home/ncs# userdel smith
root@Bry021:/home/ncs# finger smith
finger: smith: no such user.
root@Bry021:/home/ncs# ls ../smith/
root@Bry021:/home/ncs# cd ..
root@Bry021:/home# ls -l
total 16
drwxr-xr-x 2 bob bob 4096 Oct 7 16:17 bob/
drwxr-xr-x 2 root root 4096 Feb 28 2011 ftp/
drwxrwxrwx 2 1001 sysadmins 4096 Oct 7 16:17 ncs/
drwxr-xr-x 2 1001 1001 4096 Oct 7 12:35 smith/
root@Bry021:/home# _
```

The terminal window shows the execution of the `userdel smith` command. Subsequent `finger smith` and `ls ../smith/` commands confirm the user's removal. The `ls -l` command shows the remaining directories in the home folder, with `smith/` still present, indicating that the home directory was not removed. The window title is "Slackware [Running]" and the desktop environment includes a taskbar at the bottom with various icons and a "Left" button.