Challenge Lab A: User Management **in the Cisco NetAcad & LPI Linux Essentials learning path**

Case Scenario

- As the Linux Administrator for fast-growing company, you have been tasked with creating, modifying, and removing user accounts from the Linux server. The company has just hired 9 new employees to fill 3 newly designed departments. The departments that have been created are Engineering, Sales and IS. The server must be setup with the appropriate files, folders, users, groups and permissions to ensure a successful launch of the newly designed departments.

Objectives

- Create a directory at the root (/) of the file system for each department. This name should reflect the department name that will use the directory.
- Create a group for each department. This name should reflect the department name that the group will be assigned.

```
This lab has two user accounts (username :: password )

   root     :: netlab123
   sysadmin :: netlab123

Press the [Enter] key to begin...


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

sysadmin@localhost:~$ sudo mkdir /Engineering
[sudo] password for sysadmin:
sysadmin@localhost:~$ sudo mkdir /Sales
sysadmin@localhost:~$ sudo mkdir /IS
sysadmin@localhost:~$ sudo groupadd engineering
sysadmin@localhost:~$ sudo groupadd sales
sysadmin@localhost:~$ sudo groupadd is
sysadmin@localhost:~$
```

- Create an administrative user for each of the departments.
    - The user will have a Bash login shell.
    - The user will belong to the respective group for each department. This will need to be the user's primary group.

```
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g engineering eng_admin
sysadmin@localhost:~$ sudo passwd eng_admin
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g sales sales_admin
sysadmin@localhost:~$ sudo passwd sales_admin
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g is is_admin
sysadmin@localhost:~$ sudo passwd is_admin
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$
```

- Create two additional users for each department.
    - The users will have a Bash login shell.
    - The users will belong to their respective group for each department. This will need to be the user's primary group.

```
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g engineering eng_user1
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g engineering eng_user2
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g sales sales_user1
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g sales sales_user2
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g is is_user1
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo useradd -m -s /bin/bash -g is is_user2
sysadmin@localhost:~$
```

    - Set passwords for each new user

```
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$ sudo passwd eng_user2
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$ sudo passwd sales_user1
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$ sudo passwd sales_user2
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$ sudo passwd is_user1
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$ sudo passwd is_user2
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sysadmin@localhost:~$
```

- For security reasons, the following modifications will need to be made to each of the departments' respective directories:
    - Ensure that the owner of each of the directories is the department administrator and the group ownership is the group for each department.
    - The department administrator will have full access to their respective department directories.
    - Ensure that only the owner of a file in the department's directory can delete the file. The user will also have ownership of their respective department folders.
    - Normal users in each department will have full access (Read, Write and Execute) to their respective department folders.
    - The department folders will ONLY be accessible by users/administrators in each of the respective departments. Ensure that no one else will have permissions to the folders.
- Create a document in each of the department directories.
    - The ownerships on this file will be the same as the directory it is located in.
    - The document should contain only one line of text that states, "This file contains confidential information for the department."
    - This file can be read by any user in the department, but can only be modified by the department administrator. No one else has permissions to this file.

Here I set ownership and permissions for each department directory.

```
sysadmin@localhost:~$ sudo chown eng_admin:engineering /Engineering
sysadmin@localhost:~$
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo chmod 770 /Engineering
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo chmod +t /Engineering
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo chown sales_admin:sales /sales
chown: cannot access `/sales': No such file or directory
sysadmin@localhost:~$ sudo chown sales_admin:sales /Sales
sysadmin@localhost:~$ sudo chmod +t /Sales
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo chown is_admin:is /IS
sysadmin@localhost:~$ sudo chmod 770 /IS
sysadmin@localhost:~$ sudo chmod +t /IS
sysadmin@localhost:~$
```

I created a confidential file in each department directory.

```
sysadmin@localhost:~$ sudo tee /Engineering/confidential.txt > /dev/null <<
-bash: syntax error near unexpected token `newline'
sysadmin@localhost:~$ sudo tee /Engineering/confidential.txt > /dev/null << EOF
>
> This file contains confidential information for the department.
>
> EOF
[sudo] password for sysadmin:
sysadmin@localhost:~$ sudo tee /Sales/confidential.txt > /dev/null << EOF
>
> This file contains confidential information for the department.
>
> EOF
sysadmin@localhost:~$ sudo tee /IS/confidential.txt > /dev/null << EOF
>
> This file contains confidential information for the department.
>
> EOF
sysadmin@localhost:~$
```

Set appropriate permissions for the confidential files.

```
sysadmin@localhost:~$ sudo chown eng_admin:engineering /Engineering/confidential
.txt
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo chmod 764 /Engineering/confidential.txt
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo chown sales_admin:sales /Sales/confidential.txt
sysadmin@localhost:~$ sudo chmod 764 /Sales/confidential.txt
sysadmin@localhost:~$
sysadmin@localhost:~$ sudo chown is_admin:is /IS/confidential.txt
sysadmin@localhost:~$ sudo chmod 764 /IS/confidential.txt
sysadmin@localhost:~$
```

Verify Users and Groups Creation

```
sysadmin@localhost:~$ cat /etc/group | grep -E "(engineering|sales|is)"
disk:x:6:
list:x:38:
engineering:x:1002:
sales:x:1003:
is:x:1004:
sysadmin@localhost:~$ cat /etc/passwd | grep -E "eng_|sales_|is_)"
grep: Unmatched ) or \)
sysadmin@localhost:~$ cat /etc/passwd | grep -E "(eng_|sales_|is_)"
eng_admin:x:1002:1002::/home/eng_admin:/bin/bash
sales_admin:x:1003:1003::/home/sales_admin:/bin/bash
is_admin:x:1004:1004::/home/is_admin:/bin/bash
eng_user1:x:1005:1002::/home/eng_user1:/bin/bash
eng_user2:x:1006:1002::/home/eng_user2:/bin/bash
sales_user1:x:1007:1003::/home/sales_user1:/bin/bash
sales_user2:x:1008:1003::/home/sales_user2:/bin/bash
is_user1:x:1009:1004::/home/is_user1:/bin/bash
is_user2:x:1010:1004::/home/is_user2:/bin/bash
sysadmin@localhost:~$
```

Verify user group assignments

```
sysadmin@localhost:~$ groups eng_admin
eng_admin : engineering
sysadmin@localhost:~$ groups eng_user1
eng_user1 : engineering
sysadmin@localhost:~$ group sales_admin
-bash: group: command not found
sysadmin@localhost:~$ group sales_user1
-bash: group: command not found
sysadmin@localhost:~$ groups sales_admin
sales_admin : sales
sysadmin@localhost:~$ groups sales_user1
sales_user1 : sales
sysadmin@localhost:~$ groups is_admin
is_admin : is
sysadmin@localhost:~$ groups is_user1
is_user1 : is
sysadmin@localhost:~$
```

Verify Directory Creation and Permissions:

Check directory structure and permissions

```
sysadmin@localhost:~$ ls -id /Engineering
232031 /Engineering
sysadmin@localhost:~$ ls -id /Sales
239341 /Sales
sysadmin@localhost:~$ ls -id /IS
2147902681 /IS
sysadmin@localhost:~$
```

Check directory contents-

(I hit a bit of a snag here, not sure why permission was denied when trying to verify. I took a look at my script but I can't seem to spot my error with ENGINEERING and IS, I'll do more research and try find the reasoning)

```
sysadmin@localhost:~$ ls -la /Engineering/
ls: cannot open directory /Engineering/: Permission denied
sysadmin@localhost:~$ ls -la /Sales/
total 4
drwxr-xr-t 2 sales_admin sales  30 Oct 21 10:02 .
drwxr-xr-x 1 root        root  140 Oct 21 09:05 ..
-rwxrw-r-- 1 sales_admin sales  66 Oct 21 10:02 confidential.txt
sysadmin@localhost:~$ ls -la /IS/
ls: cannot open directory /IS/: Permission denied
sysadmin@localhost:~$
```

```
sysadmin@localhost:~$ ls -la /Engineering/confidential.txt
ls: cannot access /Engineering/confidential.txt: Permission denied
sysadmin@localhost:~$ cat /Engineering/confidential.txt
cat: /Engineering/confidential.txt: Permission denied
sysadmin@localhost:~$ ls -la /Sales/confidential.txt
-rwxrw-r-- 1 sales_admin sales 66 Oct 21 10:02 /Sales/confidential.txt
sysadmin@localhost:~$ ls -la /IS/confidential.txt
ls: cannot access /IS/confidential.txt: Permission denied
sysadmin@localhost:~$ cat /IS/confidential.txt
cat: /IS/confidential.txt: Permission denied
sysadmin@localhost:~$
```

My Security Explanations

- **Directory permissions (770): Only owner and group members can access (read/write/execute)**
- **Sticky bit (+t): Prevents users from deleting files they don't own**
- **File permissions (764):**
    - **Owner (admin): read, write, execute (7)**
    - **Group members: read, write (6)**
    - **Others: read only (4) - though directory permissions prevent others from accessing**

Curriculum Resources

- Module 8 – Managing Files and Directories
- Module 15 – System and User Security
- Module 16 – Creating Users and Groups
- Module 17 – Ownership and Permissions
- Module 18 – Special Directories and Files

Deliverables

- Use the appropriate command to verify each user and group has been created.
- Use the appropriate command to verify each user's group assignment.
- Use the appropriate command to verify the directory creation and the permission settings.
- Use the appropriate command to verify the files are created in their respective directories.

The END