Devin Young

Bowie State University

CTEC 435

Professor Anthony

Assignment: Exploring and Managing Users and Groups in Linux

Part 1: User Account Management

Task 1: Creating User Accounts

1. Create three new user accounts with the following usernames: `user1`, `user2`, and `user3`.

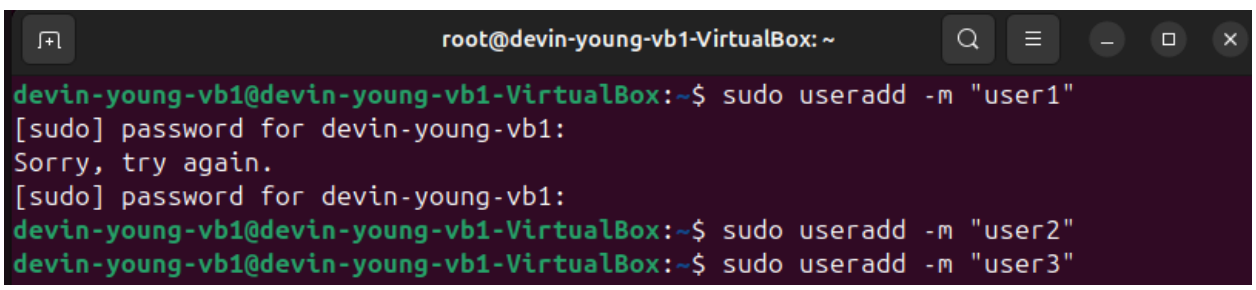
- Use the `useradd` command to create the accounts.

- Set a password for each user using the `passwd` command.

2. Verify the creation of the accounts by listing all users in the `/etc/passwd` file.

3. Document the commands used and the output, including the entries in the `/etc/passwd` file.

Step 1 - Using the user-add command to create the user accounts



Step 2 - Using the passwd command to change the password of my root user and creating passwords for the three users I created.

```
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo -i
root@devin-young-vb1-VirtualBox:~# passwd root
New password:
Retype new password:
passwd: password updated successfully
root@devin-young-vb1-VirtualBox:~# passwd user1
New password:
Retype new password:
passwd: password updated successfully
root@devin-young-vb1-VirtualBox:~# passwd user2
New password:
Retype new password:
passwd: password updated successfully
root@devin-young-vb1-VirtualBox:~# passwd user3
New password:
Retype new password:
passwd: password updated successfully
root@devin-young-vb1-VirtualBox:~#
```

Step 3 - Using the cat /etc/passwd command to verify that the user accounts have been created

```
root@devin-young-vb1-VirtualBox:~# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
```

```
devin-young-vb1:x:1000:1000:devin-young-vb1:/home/devin-young-vb1:/bin/bash
jacob:x:1001:1001::/home/jacob:/bin/sh
user1:x:1002:1002::/home/user1:/bin/sh
user2:x:1003:1003::/home/user2:/bin/sh
user3:x:1004:1004::/home/user3:/bin/sh
root@devin-young-vb1-VirtualBox:~#
```

Task 2: Modifying User Accounts

1. Change the default shell for `user1` to `/bin/bash`.

- Use the `usermod` command to modify the shell.

2. Set an expiration date for `user2`'s account to one week from today.

- Use the `chage` command to set the expiration date.

3. Lock `user3`'s account to prevent them from logging in.

- Use the `passwd -l` or `usermod -L` command to lock the account.

4. Verify the changes made to the accounts by inspecting the `/etc/passwd` and `/etc/shadow` files.

5. Document the commands used and the output, including the changes in the configuration files.

Step 1 - Switching to the root account, using the usermod command to change the default shell for user 1 to bin/bash, setting an expiration date on user 2's account to one week from today using the chage command, using the chage command with an -l flag to verify that user 2's expiration date, and locking user 3's account with the passwd -l command to prevent them from logging in.

```
devin-young-vb1@devin-young-vb1-VirtualBox:~$ su -
Password:
root@devin-young-vb1-VirtualBox:~# usermod -s /bin/ksh user1
usermod: Warning: missing or non-executable shell '/bin/ksh'
root@devin-young-vb1-VirtualBox:~# usermod -s /bin/bash user1
root@devin-young-vb1-VirtualBox:~# chage -E 2024-09-23 user2
root@devin-young-vb1-VirtualBox:~# chage -l user2
Last password change                                    : Sep 16, 2024
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : Sep 23, 2024
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
root@devin-young-vb1-VirtualBox:~# sudo passwd -l user3
passwd: password changed.
root@devin-young-vb1-VirtualBox:~#
```

Step 2 - Verifying changes using the cat /etc/passwd and cat /etc/shadow command. The cat /etc/passwd command showed me that user 1 is now in the bash shell, and the cat /etc/shadow command didn't really show me what I was looking for so I used the passwd -S command to verify that user 3's account is locked.

```
root@devin-young-vb1-VirtualBox:~# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
```

```
devin-young-vb1:x:1000:1000:devin-young-vb1:/home/devin-young-vb1:/bin/bash
jacob:x:1001:1001::/home/jacob:/bin/sh
user1:x:1002:1002::/home/user1:/bin/bash
user2:x:1003:1003::/home/user2:/bin/sh
user3:x:1004:1004::/home/user3:/bin/sh
root@devin-young-vb1-VirtualBox:~#
```

```
root@devin-young-vb1-VirtualBox: ~
root@devin-young-vb1-VirtualBox:~# cat /etc/shadow
root:$y$j9T$sN/7lkcpE6HyNtakm5Hnj.$4DMei21TQC1ykg2Kf6Cud8xjpbjsdTBgyzI43mgDxq7:19982:0:99999:7:::
daemon:*:19962:0:99999:7:::
bin:*:19962:0:99999:7:::
sys:*:19962:0:99999:7:::
sync:*:19962:0:99999:7:::
games:*:19962:0:99999:7:::
```

```
devin-young-vb1:$6$U9bBRxMZgeAJuei6$luspueOebFUxO0zyHBCgKKNS9AMFrpKkydTbo4X2hxnyT.jdGB9uGMgVRKHJPZKxARb5N9tRyssntRa0GccX
P1:19967:0:99999:7:::
jacob:$y$j9T$fdkorQXd4308CwSxJmGgi1$UKYbvJyJkzHsYQT8Wq8jiYbM8iIsCyafVMqohQwqoZB:19977:0:99999:7:::
user1:$y$j9T$7Q6bk0smqJb3z6GoB7W.p0$bSoJoK09CIKUjOnnOA8p0pKSX8SCrpjh.YOxx0ifAMD:19982:0:99999:7:::
user2:$y$j9T$r/ACTpbdM8Yi9KdIUY0H3.$hCpu4pArsjFMomKulw09Uhl4LWASxrsnDyuVtMXS4M0:19982:0:99999:7::19989:
user3:!$y$j9T$eQoUxrEQduBfybHeNSNBH/$EoT6QWsPOjiyBKzGpWJ9eu1HvztlRbOmll77L.Tzq.A:19982:0:99999:7:::
root@devin-young-vb1-VirtualBox:~#
```

```
root@devin-young-vb1-VirtualBox:~# passwd -S user3
user3 L 2024-09-16 0 99999 7 -1
root@devin-young-vb1-VirtualBox:~#
```

Part 2: Group Management

Task 3: Creating and Managing Groups

1. Create two new groups called `group1` and `group2`.

- Use the `groupadd` command to create the groups.

2. Add `user1` to `group1` and `user2` to `group2`.

- Use the `usermod` or `gpasswd` command to add users to groups.

3. Add `user3` to both `group1` and `group2` as a secondary group membership.

- Ensure that `user3` retains their primary group membership as well.

4. Verify group memberships by listing the groups associated with each user using the `groups` command.

5. Document the commands used and the output, including the entries in the `/etc/group` file.

Step 1 - Using the groupadd command to create group 1 and group 2, then using the usermod command to add user 1 to group 1, user 2 to group 2, and user 3 to both group 1 and group 2

```
root@devin-young-vb1-VirtualBox: ~
devin-young-vb1@devin-young-vb1-VirtualBox:~$ su -
Password:
root@devin-young-vb1-VirtualBox:~# groupadd group1
groupadd: group 'group1' already exists
root@devin-young-vb1-VirtualBox:~# groupadd group2
groupadd: group 'group2' already exists
root@devin-young-vb1-VirtualBox:~# usermod -aG group1 user1
root@devin-young-vb1-VirtualBox:~# usermod -aG group2 user2
root@devin-young-vb1-VirtualBox:~# usermod -aG group1,group2 user3
root@devin-young-vb1-VirtualBox:~#
```

Step 2 - Verifying the groups and users within the groups using the groups and cat /etc/group command.

```
root@devin-young-vb1-VirtualBox:~# groups user1
user1 : user1 group1
root@devin-young-vb1-VirtualBox:~# groups user2
user2 : user2 group2
root@devin-young-vb1-VirtualBox:~# groups user3
user3 : user3 group1 group2
root@devin-young-vb1-VirtualBox:~# cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,devin-young-vb1
tty:x:5:
```

```
devin-young-vb1:x:1000:
students:x:1001:
user1:x:1002:
user2:x:1003:
user3:x:1004:
group1:x:1005:user1,user3
group2:x:1006:user2,user3
root@devin-young-vb1-VirtualBox:~#
```

Task 4: Managing Group Permissions

1. Create a shared directory `/shared` and set `group1` as the group owner of the directory.

- Use the `chgrp` command to change the group ownership of the directory.

2. Configure the directory permissions to allow members of `group1` to read, write, and execute files in the directory, while denying access to others.

- Use the `chmod` command with the appropriate permission settings.

3. Test the permissions by logging in as `user1` (who is a member of `group1`) and attempting to create a file in the `/shared` directory.

- Verify that `user1` can create and edit files in the directory.

4. Attempt to access the directory as `user2` (who is not a member of `group1`) and confirm that access is denied.

5. Document the commands used, the permissions set, and the results of the access tests.

Step 1 - Creating a shared directory using the mkdir /shared command and setting group 1 as the group owner of the directory with the chgrp group 1 /shared command.

```
root@devin-young-vb1-VirtualBox:~# mkdir /shared
root@devin-young-vb1-VirtualBox:~# chgrp group1 /shared
root@devin-young-vb1-VirtualBox:~#
```

Step 2 - Configuring the directory permissions to allow members of `group1` to read, write, and execute files in the directory, while denying access to others using the chmod command. The 2 makes sure that files created in the directory inherit the group ownership, the first two 7s allow the owner of group 1 and the group itself to read, write, and execute files, and the 0 prevents others from doing any of those.

```
root@devin-young-vb1-VirtualBox:~# chmod 2770 /shared
root@devin-young-vb1-VirtualBox:~#
```

Step 3 - Successfully logging in as user 1 and creating a file within the shared directory after setting the previous permissions.

```
root@devin-young-vb1-VirtualBox:~# su - user1
user1@devin-young-vb1-VirtualBox:~$ sudo touch /shared/testfile.txt
[sudo] password for user1:
user1 is not in the sudoers file.
user1@devin-young-vb1-VirtualBox:~$ touch /shared/testfile.txt
user1@devin-young-vb1-VirtualBox:~$ ls -l /shared/testfile.txt
-rw-rw-r-- 1 user1 group1 0 Sep 16 14:02 /shared/testfile.txt
user1@devin-young-vb1-VirtualBox:~$
```

Step 4 - Logging in as user 2 and attempting to access the shared directory (being denied because of the permissions I set).

```
user1@devin-young-vb1-VirtualBox:~$ su - user2
Password:
$ ls -l /shared
ls: cannot open directory '/shared': Permission denied
$
```

Part 3: User Account Security

Task 5: Configuring Password Policies

1. Set up password aging policies for all users to enforce password changes every 30 days.

- Use the `chage` command to configure password aging.

2. Configure the system to lock user accounts after three failed login attempts.

- Use the `pam_tally2` module or other relevant PAM modules to enforce account locking.

3. Test the password policy by attempting to log in with incorrect passwords and verifying that the account is locked after three attempts.

4. Document the commands used and the results of the tests.

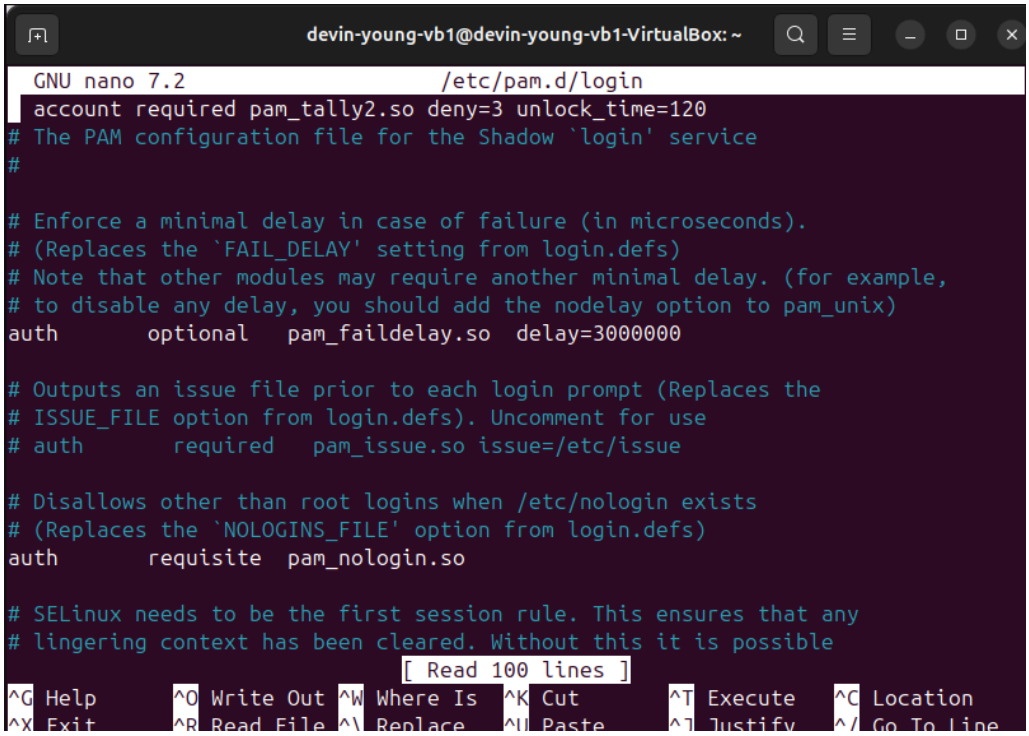
Step 1 - Setting up password aging policies for all users to enforce password changes every 30 days using the chage command and etc.

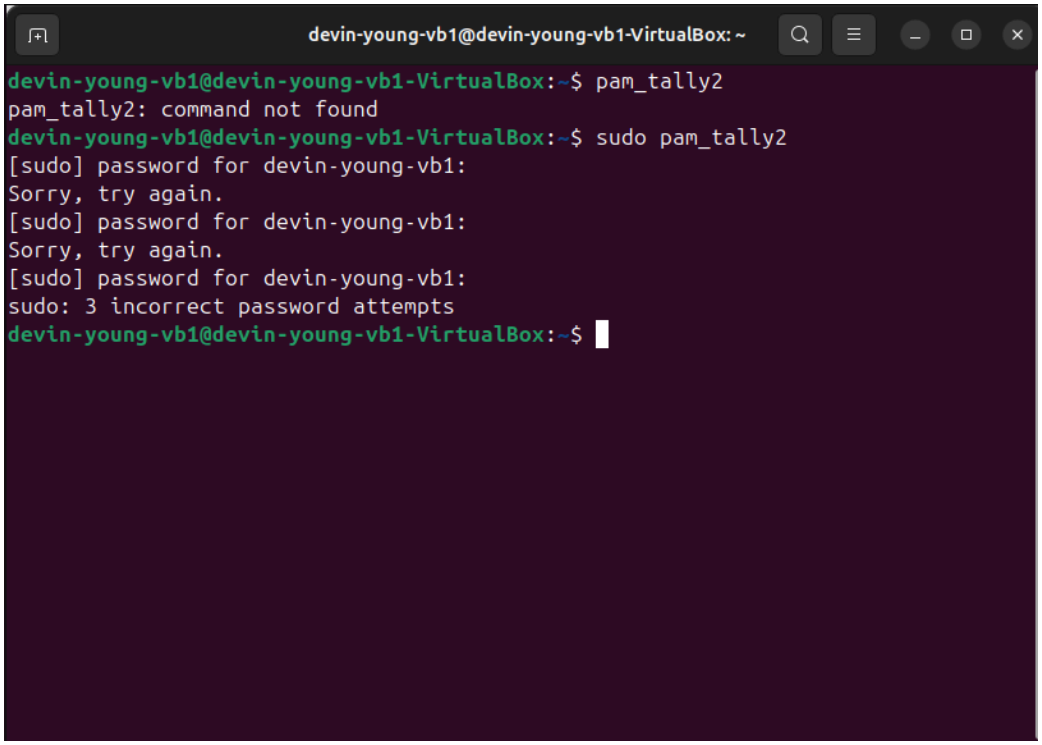
```
root@devin-young-vb1-VirtualBox:~# # Define the maximum number of days a password can be used
MAX_DAYS=30

# Configure system-wide default password aging settings
echo "Configuring system-wide password aging settings..."
sudo sed -i "s/^PASS_MAX_DAYS.*/PASS_MAX_DAYS   $MAX_DAYS/" /etc/login.defs

# Apply password aging settings to all users
echo "Applying password aging settings to all users..."
for USER in $(cut -f1 -d: /etc/passwd); do
    sudo chage -M $MAX_DAYS $USER
    echo "Updated password aging for $USER"
done

echo "Password aging policies have been set to require password changes every $MAX_DAYS days."
Configuring system-wide password aging settings...
Applying password aging settings to all users...
Updated password aging for root
Updated password aging for daemon
Updated password aging for bin
Updated password aging for sys
Updated password aging for sync
Updated password aging for games
```

```
Updated password aging for cups-pk-helper
Updated password aging for fwupd-refresh
Updated password aging for saned
Updated password aging for geoclue
Updated password aging for cups-browsed
Updated password aging for hplip
Updated password aging for gnome-remote-desktop
Updated password aging for polkitd
Updated password aging for rtkit
Updated password aging for colord
Updated password aging for gnome-initial-setup
Updated password aging for gdm
Updated password aging for nm-openvpn
Updated password aging for devin-young-vb1
Updated password aging for jacob
Updated password aging for user1
Updated password aging for user2
Updated password aging for user3
Password aging policies have been set to require password changes every 30 days.
```

Step 2 - Configuring the system to lock user accounts after three failed login attempts using the `pam_tally2` module or other relevant PAM modules to enforce account locking; using the sudo nano /etc/pam.d/login command and testing the password policy by attempting to

make changes on my user account directly after a reboot with incorrect passwords (I wasn't officially recognized as logged in so it would stop me after 3 failed login attempts).





Task 6: Configuring Sudo Access

1. Grant `user1` sudo privileges, allowing them to execute administrative commands.

- Use the `visudo` command to edit the sudoers file and add `user1` to the sudoers list.

2. Test `user1`'s sudo access by executing a command that requires elevated privileges (e.g., updating the system).

3. Document the commands used and the output of the sudo test.

Step 1 - Using the sudo visudo command to allow user 1 to execute administrative commands. After I made the changes, I used the sudo whoami command to confirm that user1 had root level privileges, which is why root was returned.

```
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo visudo
devin-young-vb1@devin-young-vb1-VirtualBox:~$ su - user1
Password:
user1@devin-young-vb1-VirtualBox:~$ sudo whoami
[sudo] password for user1:
root
user1@devin-young-vb1-VirtualBox:~$
```

```
  GNU nano 7.2                    /etc/sudoers.tmp
user1 ALL=(ALL) ALL
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:>

# This fixes CVE-2005-4890 and possibly breaks some versions of kdesu
# (#1011624, https://bugs.kde.org/show_bug.cgi?id=452532)
Defaults        use_pty

# This preserves proxy settings from user environments of root
# equivalent users (group sudo)
#Defaults:%sudo env_keep += "http_proxy https_proxy ftp_proxy all_proxy no_prox>

                              [ Read 57 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut       ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste     ^J Justify    ^/ Go To Line
```

Step 2 - Testing user1's sudo access by using the sudo cat /etc/fstab command (which can only be accessed by the root user) and the sudo ps aux command, which can only be accessed by the root user and is for processes owned by other users.

```
user1@devin-young-vb1-VirtualBox:~$ sudo cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
/dev/disk/by-uuid/97bc6196-4b01-417f-b912-dc63d6a03362 none swap sw 0 0
# / was on /dev/sda3 during curtin installation
/dev/disk/by-uuid/630e187f-0970-4c19-9fcf-03dd81ef4c1f / ext4 defaults,usrquota,
grpquota 0 1
user1@devin-young-vb1-VirtualBox:~$ sudo ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.2  23340 14164 ?        Ss   15:05   0:01 /sbin/init sp
root           2  0.0  0.0      0     0 ?        S    15:05   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        S    15:05   0:00 [pool_workque
root           4  0.0  0.0      0     0 ?        I<   15:05   0:00 [kworker/R-rc
root           5  0.0  0.0      0     0 ?        I<   15:05   0:00 [kworker/R-rc
root           6  0.0  0.0      0     0 ?        I<   15:05   0:00 [kworker/R-sl
root           7  0.0  0.0      0     0 ?        I<   15:05   0:00 [kworker/R-ne
root          11  0.0  0.0      0     0 ?        I    15:05   0:00 [kworker/u6:0
root          12  0.0  0.0      0     0 ?        I<   15:05   0:00 [kworker/R-mm
```

```
user1@devin-young-vb1-VirtualBox: ~
root         922  0.0  0.1 318388  6656 ?        Ssl  15:05   0:00 /usr/libexec/
root         949  0.0  0.1  18132  8576 ?        Ss   15:05   0:00 /usr/lib/syst
root         950  0.0  0.2 469668 13864 ?        Ssl  15:05   0:00 /usr/libexec/
syslog       952  0.0  0.1 222564  5888 ?        Ssl  15:05   0:00 /usr/sbin/rsy
avahi        957  0.0  0.0   8476  1296 ?        S    15:05   0:00 avahi-daemon:
root         958  0.0  0.3 345028 19128 ?        Ssl  15:05   0:00 /usr/sbin/Net
root         959  0.0  0.1  17376  6144 ?        Ss   15:05   0:00 /usr/sbin/wpa
root         996  0.0  0.2 318376 12588 ?        Ssl  15:05   0:00 /usr/sbin/Mod
root        1138  0.0  0.2  46916 11776 ?        Ss   15:05   0:00 /usr/sbin/cup
root        1141  0.0  0.4 120904 22656 ?        Ssl  15:05   0:00 /usr/bin/pyth
kernoops    1171  0.0  0.0  12744  2324 ?        Ss   15:05   0:00 /usr/sbin/ker
kernoops    1174  0.0  0.0  12744  2324 ?        Ss   15:05   0:00 /usr/sbin/ker
root        1181  0.0  0.1 323488  9216 ?        Ssl  15:05   0:00 /usr/sbin/gdm
cups-br+    1187  0.0  0.3 268400 19584 ?        Ssl  15:05   0:00 /usr/sbin/cup
root        1190  0.0  0.1 398396 10496 ?        Sl   15:05   0:00 gdm-session-w
root        1199  0.0  0.0      0     0 ?        S    15:05   0:00 [psimon]
devin-y+    1201  0.0  0.2  21192 12692 ?        Ss   15:05   0:00 /usr/lib/syst
devin-y+    1202  0.0  0.0  21460  3612 ?        S    15:05   0:00 (sd-pam)
devin-y+    1210  0.0  0.2 123860 14080 ?        S<sl 15:05   0:00 /usr/bin/pipe
devin-y+    1211  0.0  0.1 106404  5760 ?        Ssl  15:05   0:00 /usr/bin/pipe
devin-y+    1214  0.0  0.3 415496 18432 ?        S<sl 15:05   0:00 /usr/bin/wire
devin-y+    1215  0.0  0.3 124996 17180 ?        S<sl 15:05   0:00 /usr/bin/pipe
devin-y+    1216  0.0  0.1 325176  9856 ?        SLsl 15:05   0:00 /usr/bin/gnom
devin-y+    1228  0.0  0.1  11120  6528 ?        Ss   15:05   0:00 /usr/bin/dbus
```

Part 4: Cleanup

Task 7: Cleaning Up

1. After completing all tasks, remove the user accounts (`user1`, `user2`, `user3`) and groups (`group1`, `group2`) that were created.

- Use the `userdel` and `groupdel` commands for cleanup.

2. Verify that the users and groups have been successfully removed by checking the `/etc/passwd`, `/etc/shadow`, and `/etc/group` files.

3. Document the cleanup process and the final state of the system.

Step 1 - Removing user accounts with the userdel command and removing group accounts with the group del command, and verifying that those groups have been deleted. I'm only left with the students group (I created prior to this assignment) and the user jacob (which I also created prior to this assignment).

```
devin-young-vb1@devin-young-vb1-VirtualBox: ~
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo kill -9 3501
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo userdel user1
userdel: user user1 is currently used by process 4301
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo userdel --force user1
userdel: user user1 is currently used by process 4301
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo kill -9 4301
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo userdel user1
userdel: user 'user1' does not exist
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo userdel user2
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo userdel user3
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo groupdel group1
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo groupdel group2
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
```

```
devin-young-vb1@devin-young-vb1-VirtualBox: ~
devin-young-vb1:x:1000:1000:devin-young-vb1:/home/devin-young-vb1:/bin/bash
jacob:x:1001:1001::/home/jacob:/bin/sh
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo cat /etc/shadow
root:$y$j9T$sN/7lkcpE6HyNtakm5Hnj.$4DMei21TQC1ykg2Kf6Cud8xjpbjsdTBgyzI43mgDxq7:19982:0:30:7:::
daemon:*:19962:0:30:7:::
bin:*:19962:0:30:7:::
sys:*:19962:0:30:7:::
sync:*:19962:0:30:7:::
games:*:19962:0:30:7:::
man:*:19962:0:30:7:::
lp:*:19962:0:30:7:::
mail:*:19962:0:30:7:::
news:*:19962:0:30:7:::
uucp:*:19962:0:30:7:::
proxy:*:19962:0:30:7:::
www-data:*:19962:0:30:7:::
backup:*:19962:0:30:7:::
list:*:19962:0:30:7:::
irc:*:19962:0:30:7:::
_apt:*:19962:0:30:7:::
nobody:*:19962:0:30:7:::
systemd-network:!*:19962::30::::
systemd-timesync:!*:19962::30::::
dhcpcd:!:19962::30::::
messagebus:!:19962::30::::
syslog:!:19962::30::::
systemd-resolve:!*:19962::30::::
uuidd:!:19962::30::::
usbmux:!:19962::30::::
tss:!:19962::30::::
systemd-oom:!*:19962::30::::
kernoops:!:19962::30::::
```

devin-young-vb1@devin-young-vb1-VirtualBox: ~

```
nm-openvpn:!:19962::30::::
devin-young-vb1:$6$U9bBRxMZgeAJuei6$luspueOebFUxO0zyHBCgKKNS9AMFrpKkydTbo4X2hxnyT.jdGB9uGMgVRKHJPZKxARb5N9tRyssntRa0GccX
P1:19967:0:30:7:::
jacob:$y$j9T$fdkorQXd4308CwSxJmGgi1$UKYbvJyJkzHsYQT8Wq8jiYbM8iIsCyafVMqohQwqoZB:19977:0:30:7:::
devin-young-vb1@devin-young-vb1-VirtualBox:~$ sudo cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,devin-young-vb1
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:devin-young-vb1
floppy:x:25:
tape:x:26:
sudo:x:27:devin-young-vb1
audio:x:29:
dip:x:30:devin-young-vb1
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
```

devin-young-vb1@devin-young-vb1-VirtualBox: ~

```
syslog:x:102:
systemd-resolve:x:991:
uuidd:x:103:
_ssh:x:104:
tss:x:105:
ssl-cert:x:106:
systemd-oom:x:990:
bluetooth:x:107:
rdma:x:108:
whoopsie:x:109:
netdev:x:110:
avahi:x:111:
tcpdump:x:112:
sssd:x:113:
lpadmin:x:114:devin-young-vb1
fwupd-refresh:x:989:
scanner:x:115:saned
saned:x:116:
geoclue:x:117:
pipewire:x:118:
gnome-remote-desktop:x:988:
polkitd:x:987:
rtkit:x:119:
colord:x:120:
gdm:x:121:
nm-openvpn:x:122:
lxd:x:123:
gamemode:x:986:
gnome-initial-setup:x:985:
devin-young-vb1:x:1000:
students:x:1001:
devin-young-vb1@devin-young-vb1-VirtualBox:~$
```

Reflection

As it pertains to linux groups and user management, I learned that it's very easy to create groups and set passwords, and that there are ways to manage permissions for all users at the same time. In addition, I learned that certain commands for managing file systems are involved in managing groups and users as well. Next, I also learned that at times when trying to delete a user's account, it may delete because of a process that's still running, so then you have to manually kill that process and attempt to delete the user's account again. Fourthly, I learned how to escalate the privileges of specific user accounts and give them privileges only root users have (using the sudo visudo command), which was pretty fun. Lastly, a new command I learned about is the chage command, which changes the number of days between password changes and the date of the last password change.