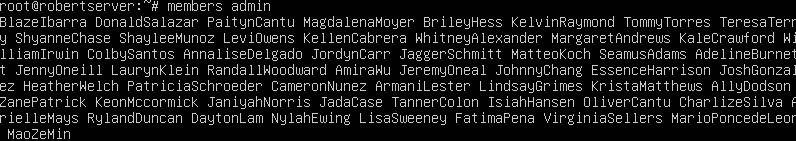
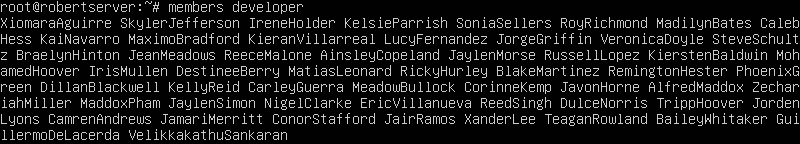
Coming up with a method to generate usernames was actually pretty simple. Just first name followed by last name. I also operated under the assumption that all files and the skeleton directories had already been set up. Setting up groups was easy, but creating users was a bit difficult. The only way I could find to create users required more information than just a username. Then I realized I could use sed to put the details into the file. This would allow me to use the command 'newusers' to create the users and place them into groups. All I needed to do was edit to add a string of characters identifying which group they would be placed in. In the second group of users, I noticed two especially long names. I decided it would be best to cut them down to their first and second names. After being unable to find a way to do that, I decided instead to change my rules so it's just the first eighteen characters of their names. Finally, I figured out a way to work everything into just one script.

So now I have the four groups with the following members

Admins



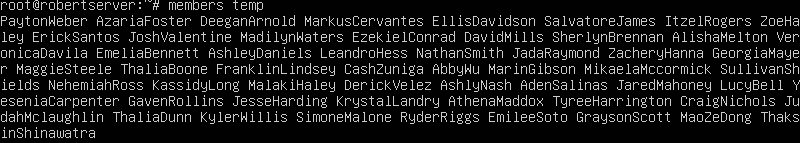
Developers



Staff



Temps



The script follows

#This is essentially all of my previous work on this assignment condensed into a single script

#There is definitely a better way to do this, but this is the one I pieced together

#These first two lines remove spaces and shorten all names to 18 characters in length

#Also forgot how to make permanent changes with sed, found a work around for that

sed -e 's/[[:space:]]//g' UserNamesLvl1.txt > newusers1a.txt

cut -c -18 newusers1a.txt > newusers1b.txt

#I mentioned I forgot how to make changes with sed, well this my work around, create an output file, then remove the file you just used the command on

rm newusers1a.txt

#These next lines fill out all the information needed for the newusers command

#Every first user is an admin, every second is a developer, every third is general staff, and every fourth is a temp

#I know this does not reflect real world situations, but it was the first thing I figured out

sed -e '1~4 s,$,:password:2000:2000:Admin:/etc/admin\_skeleton:/bin/bash,' newusers1b.txt > newusers1c.txt

rm newusers1b.txt

sed -e '2~4 s,$,:password:3000:3000:Developer:/etc/developer\_skeleton:/bin/tcsh,' newusers1c.txt > newusers1d.txt

rm newusers1c.txt

sed -e '3~4 s,$,:password:4000:4000:Staff:/etc/staff\_skeleton:/bin/bash,' newusers1d.txt > newusers1e.txt

rm newusers1d.txt

sed -e '4~4 s,$,:password:5000:5000:Temp:/etc/temp\_skeleton:/bin/bash,' newusers1e.txt > newusers1f.txt

rm newusers1e.txt

#Now these lines set up the second set of new users for the newusers command

sed -e 's/[[:space:]]//g' UserNamesLvl2.txt > newusers2a.txt

cut -c -18 newusers2a.txt > newusers2b.txt

rm newusers2a.txt

sed -e '1~4 s,$,:password:2000:2000:Admin:/etc/admin\_skeleton:/bin/bash,' newusers2b.txt > newusers2c.txt

rm newusers2b.txt

sed -e '2~4 s,$,:password:3000:3000:Developer:/etc/developer\_skeleton:/bin/tcsh,' newusers2c.txt > newusers2d.txt

rm newusers2c.txt

sed -e '3~4 s,$,:password:4000:4000:Staff:/etc/staff\_skeleton:/bin/bash,' newusers2d.txt > newusers2e.txt

rm newusers2d.txt

sed -e '4~4 s,$,:password:5000:5000:Temp:/etc/temp\_skeleton:/bin/bash,' newusers2e.txt > newusers2f.txt

rm newusers2e.txt

#Now these next few lines create the new users, give them passwords, assign them to groups, and give them their skeleton directory

#I assume that all groups and directories have been created

#It should be noted that using newusers will not allow two of the same user to be created

newusers newusers1f.txt

rm newusers1f.txt

newusers newusers2f.txt

rm newusers2f.txt

<https://github.com/RobertgSchultz8/Networking>