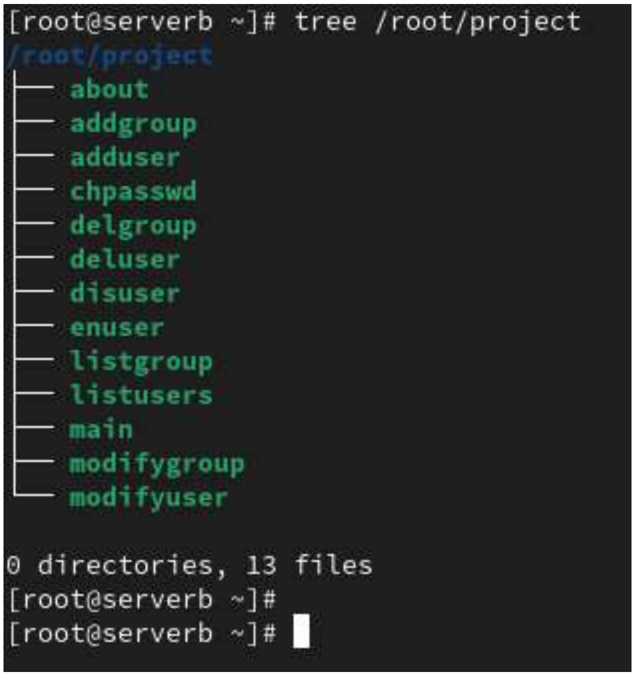
Linux User & Group Management Menu (Bash + Whiptail)

By: Omar Khaled Mohamed

Supervised by: Eng/ Romany Nageh

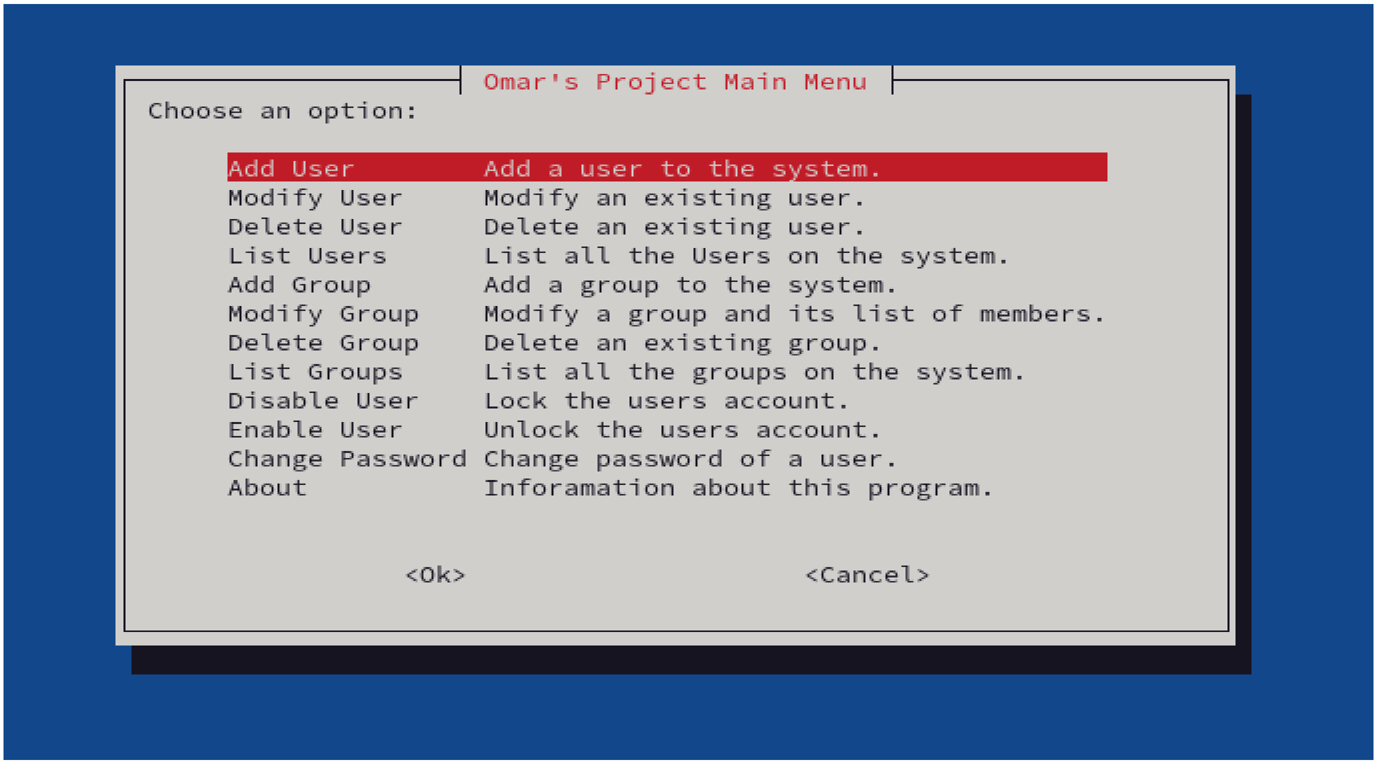
- All scripts are stored under: ***/root/project***



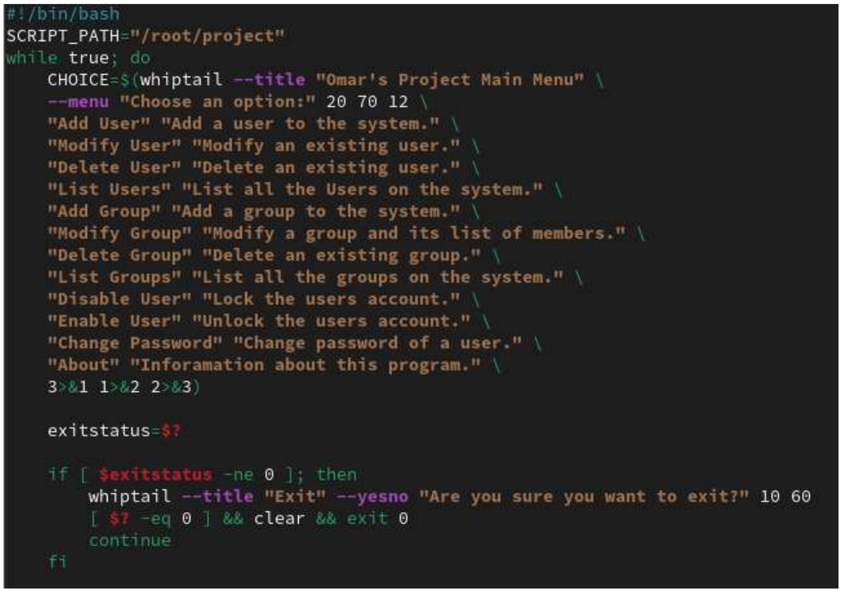
```
[root@serverb ~]# tree /root/project
/root/project
├── about
├── addgroup
├── adduser
├── chpasswd
├── delgroup
├── deluser
├── disuser
├── enuser
├── listgroup
├── listusers
├── main
├── modifygroup
└── modifyuser

0 directories, 13 files
[root@serverb ~]#
[root@serverb ~]#
```

Script Name	**Description**
main.sh	The main Whiptail menu interface that connects all scripts.
adduser	Adds a new user account.
modifyuser	Opens a submenu to modify username, UID, home directory, or group.
deluser	Deletes a user account.
listuser	Displays a list of all system users.
addgroup	Creates a new group.
modifygroup	Opens a submenu to rename group or edit GID.
delgroup	Deletes a group.
listgroup	Displays a list of system groups.
disuser	Disables a user account (locks it).
enuser	Enables a user account (unlocks it).
chpasswd	Changes a user's password.
about	Displays project information.

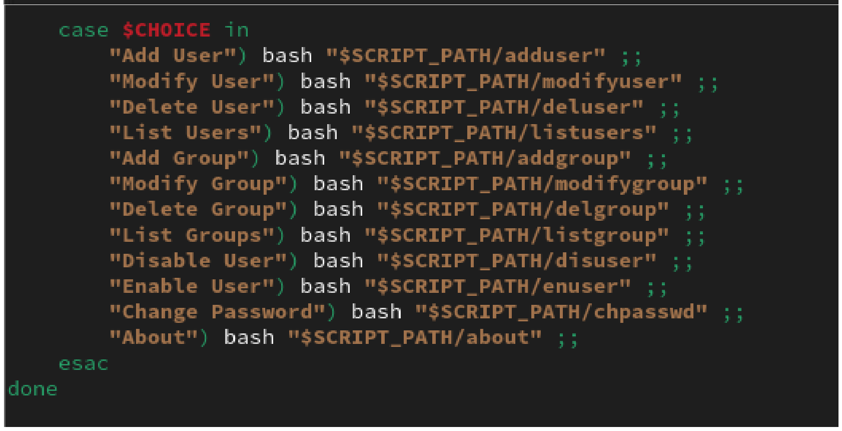
Main Menu (*main.sh*):


Omar's Project Main Menu
Choose an option:
Add User	Add a user to the system.
Modify User	Modify an existing user.
Delete User	Delete an existing user.
List Users	List all the Users on the system.
Add Group	Add a group to the system.
Modify Group	Modify a group and its list of members.
Delete Group	Delete an existing group.
List Groups	List all the groups on the system.
Disable User	Lock the users account.
Enable User	Unlock the users account.
Change Password	Change password of a user.
About	Inforamation about this program.
<Ok>
<Cancel>

main.sh



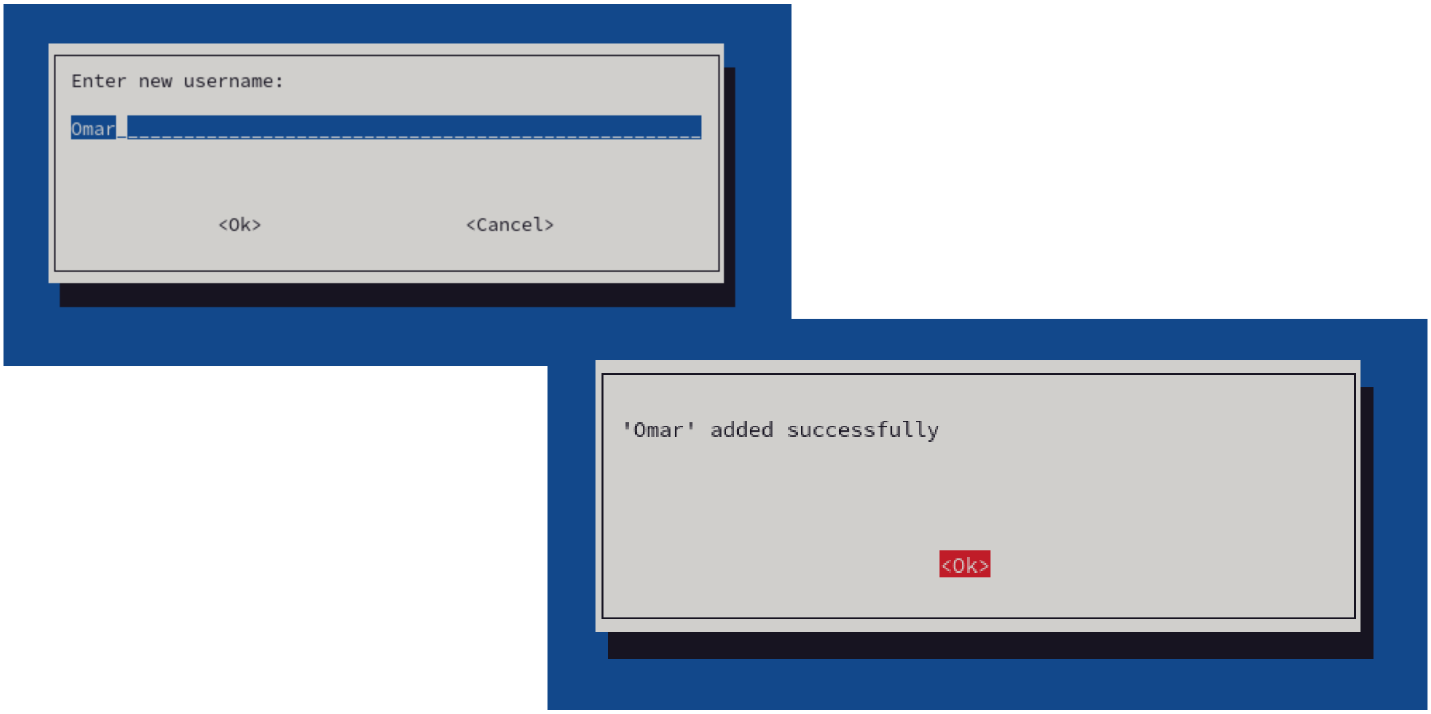
```
#!/bin/bash
SCRIPT_PATH="/root/project"
while true; do
    CHOICE=$(whiptail --title "Omar's Project Main Menu" \
    --menu "Choose an option:" 20 70 12 \
    "Add User" "Add a user to the system." \
    "Modify User" "Modify an existing user." \
    "Delete User" "Delete an existing user." \
    "List Users" "List all the Users on the system." \
    "Add Group" "Add a group to the system." \
    "Modify Group" "Modify a group and its list of members." \
    "Delete Group" "Delete an existing group." \
    "List Groups" "List all the groups on the system." \
    "Disable User" "Lock the users account." \
    "Enable User" "Unlock the users account." \
    "Change Password" "Change password of a user." \
    "About" "Inforamation about this program." \
    3>&1 1>&2 2>&3)

    exitstatus=$?

    if [ $exitstatus -ne 0 ]; then
        whiptail --title "Exit" --yesno "Are you sure you want to exit?" 10 60
        [ $? -eq 0 ] && clear && exit 0
        continue
    fi
```



```
    case $CHOICE in
        "Add User") bash "$SCRIPT_PATH/adduser" ;;
        "Modify User") bash "$SCRIPT_PATH/modifyuser" ;;
        "Delete User") bash "$SCRIPT_PATH/deluser" ;;
        "List Users") bash "$SCRIPT_PATH/listusers" ;;
        "Add Group") bash "$SCRIPT_PATH/addgroup" ;;
        "Modify Group") bash "$SCRIPT_PATH/modifygroup" ;;
        "Delete Group") bash "$SCRIPT_PATH/delgroup" ;;
        "List Groups") bash "$SCRIPT_PATH/listgroup" ;;
        "Disable User") bash "$SCRIPT_PATH/disuser" ;;
        "Enable User") bash "$SCRIPT_PATH/enuser" ;;
        "Change Password") bash "$SCRIPT_PATH/chpasswd" ;;
        "About") bash "$SCRIPT_PATH/about" ;;
    esac
done
```

Add user (*adduser.sh*):

Adds a new user account



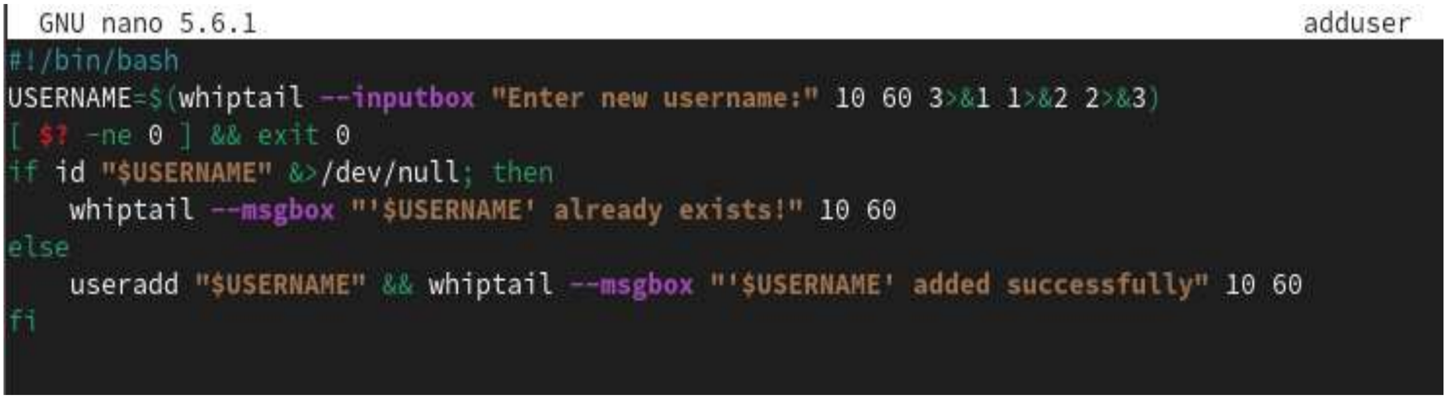
Enter new username:

Omar

'Omar' added successfully

adduser.sh

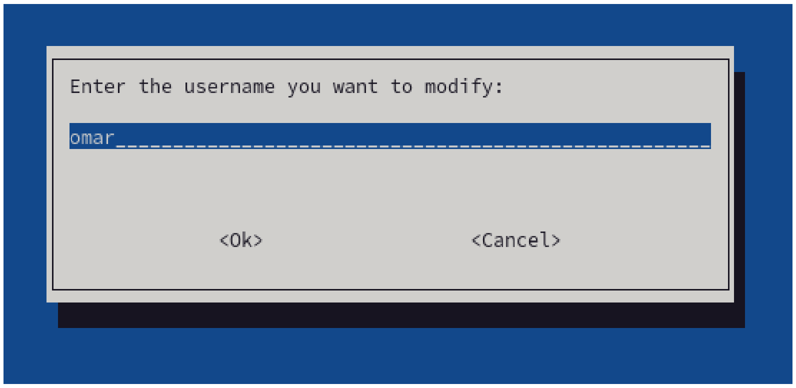
Allows the administrator to create a new system user through a simple prompt by asking for a username using an input box. The script uses *useradd* to create the user, then Displays a message confirming success or failure.



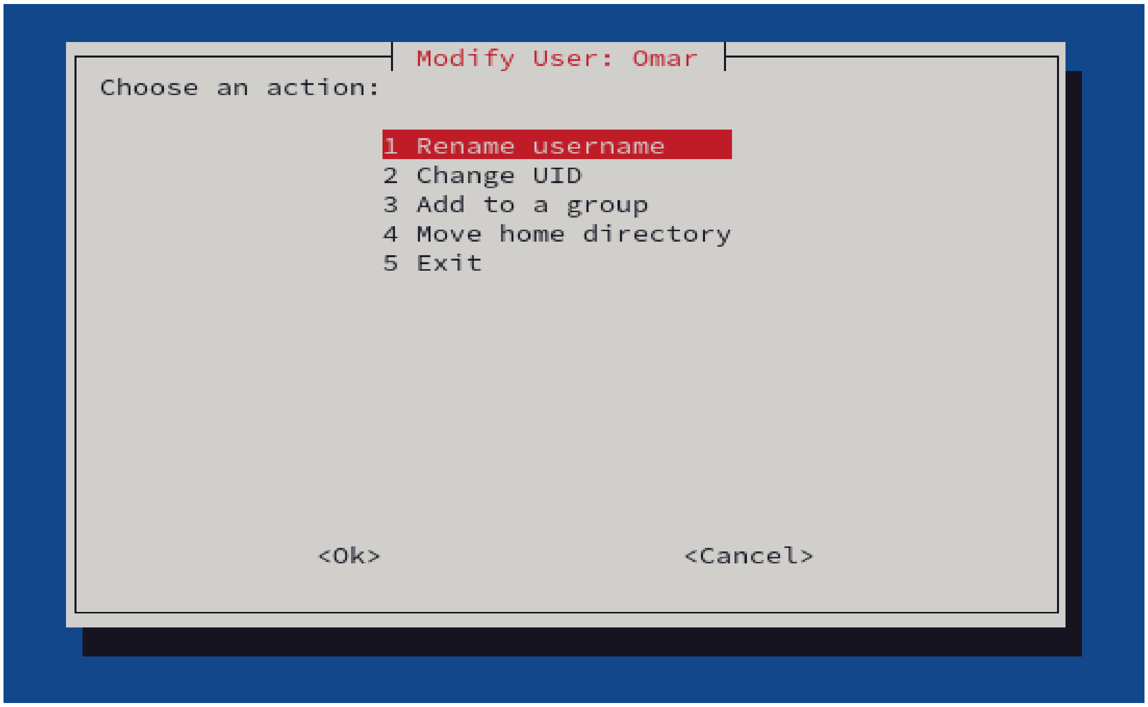
```
GNU nano 5.6.1                                                        adduser
#!/bin/bash
USERNAME=$(whiptail --inputbox "Enter new username:" 10 60 3>&1 1>&2 2>&3)
[ $? -ne 0 ] && exit 0
if id "$USERNAME" &>/dev/null; then
    whiptail --msgbox "'$USERNAME' already exists!" 10 60
else
    useradd "$USERNAME" && whiptail --msgbox "'$USERNAME' added successfully" 10 60
fi
```

Modify user (*modifyuser.sh*):

Displays a submenu to choose what to modify

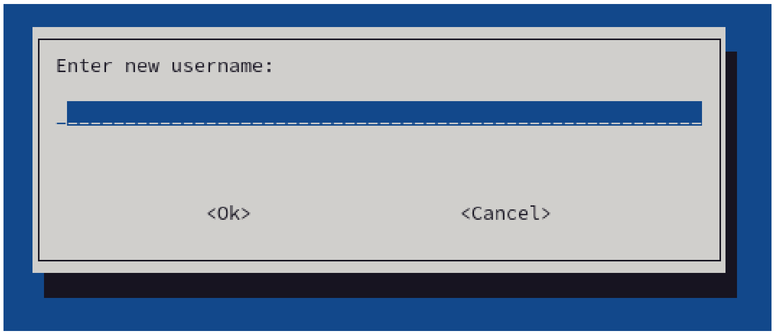


```
Enter the username you want to modify:

omar
```



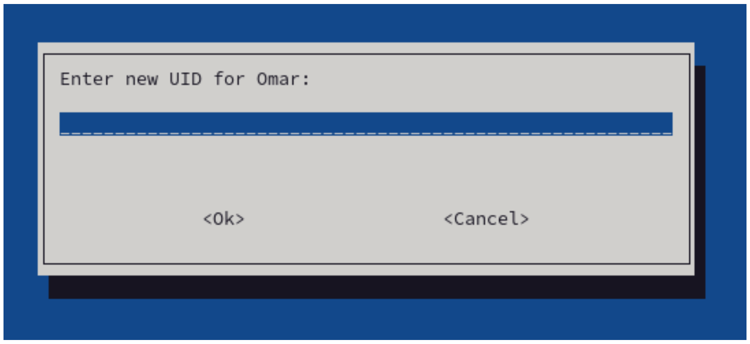
```
Modify User: Omar
Choose an action:
```

- Rename user *(usermod -l)*



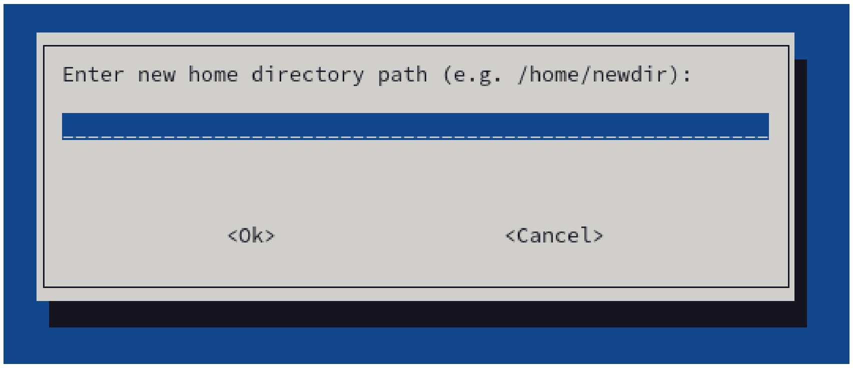
```
Enter new username:
```

- Change UID *(usermod -u)*



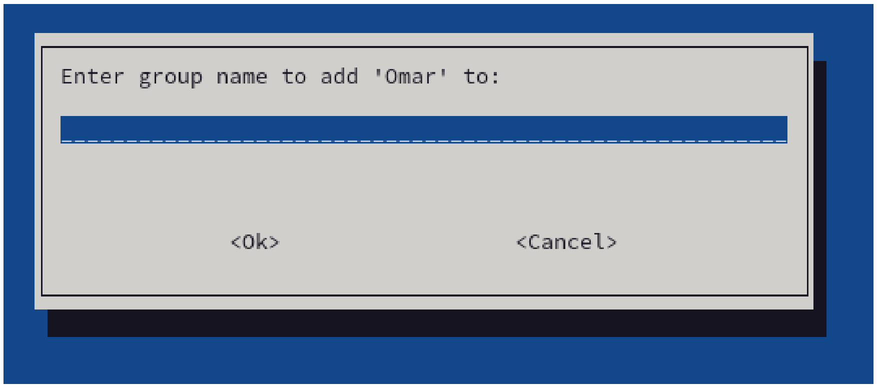
```
Enter new UID for Omar:
```

- Move home directory *(usermod -d)*



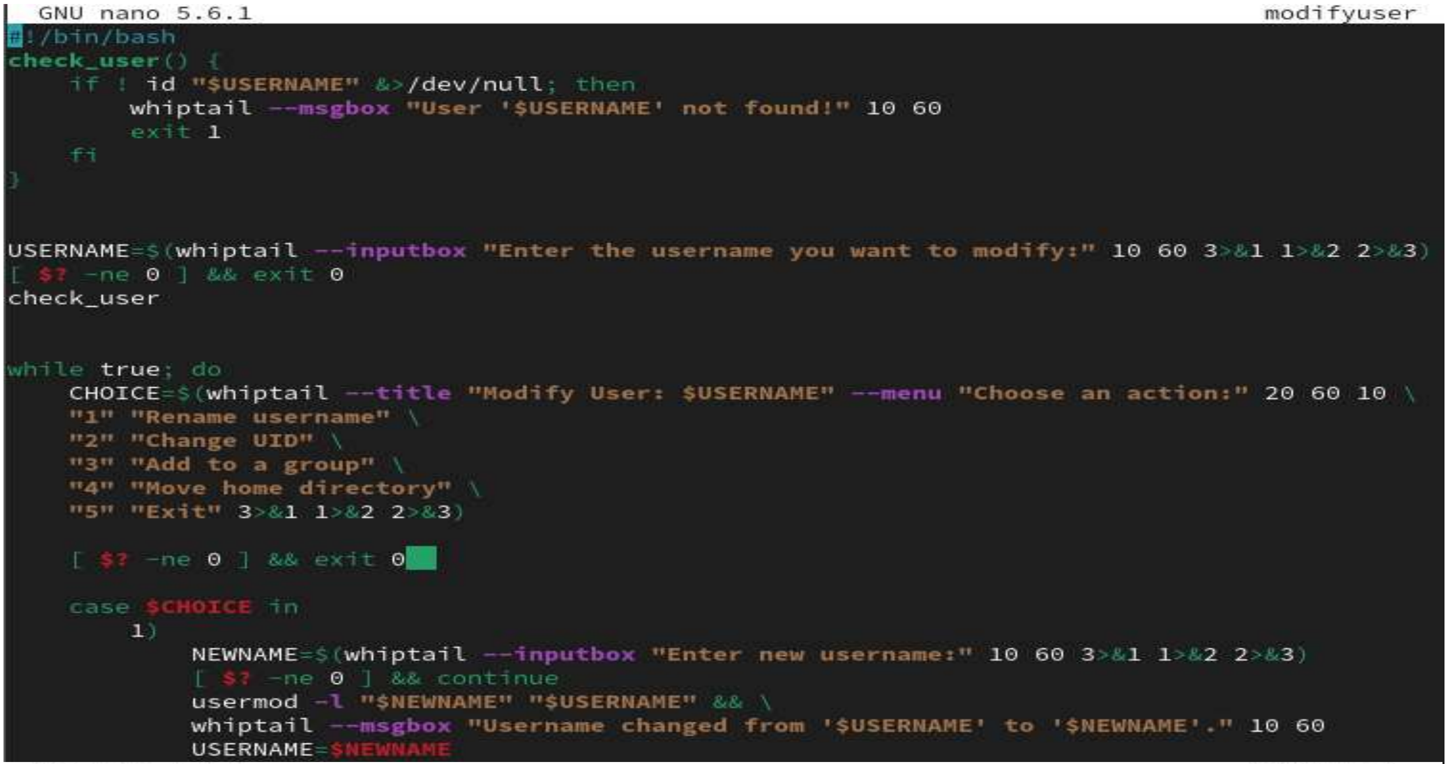
```
Enter new home directory path (e.g. /home/newdir):
```

- Add to another group *(usermod –aG)*

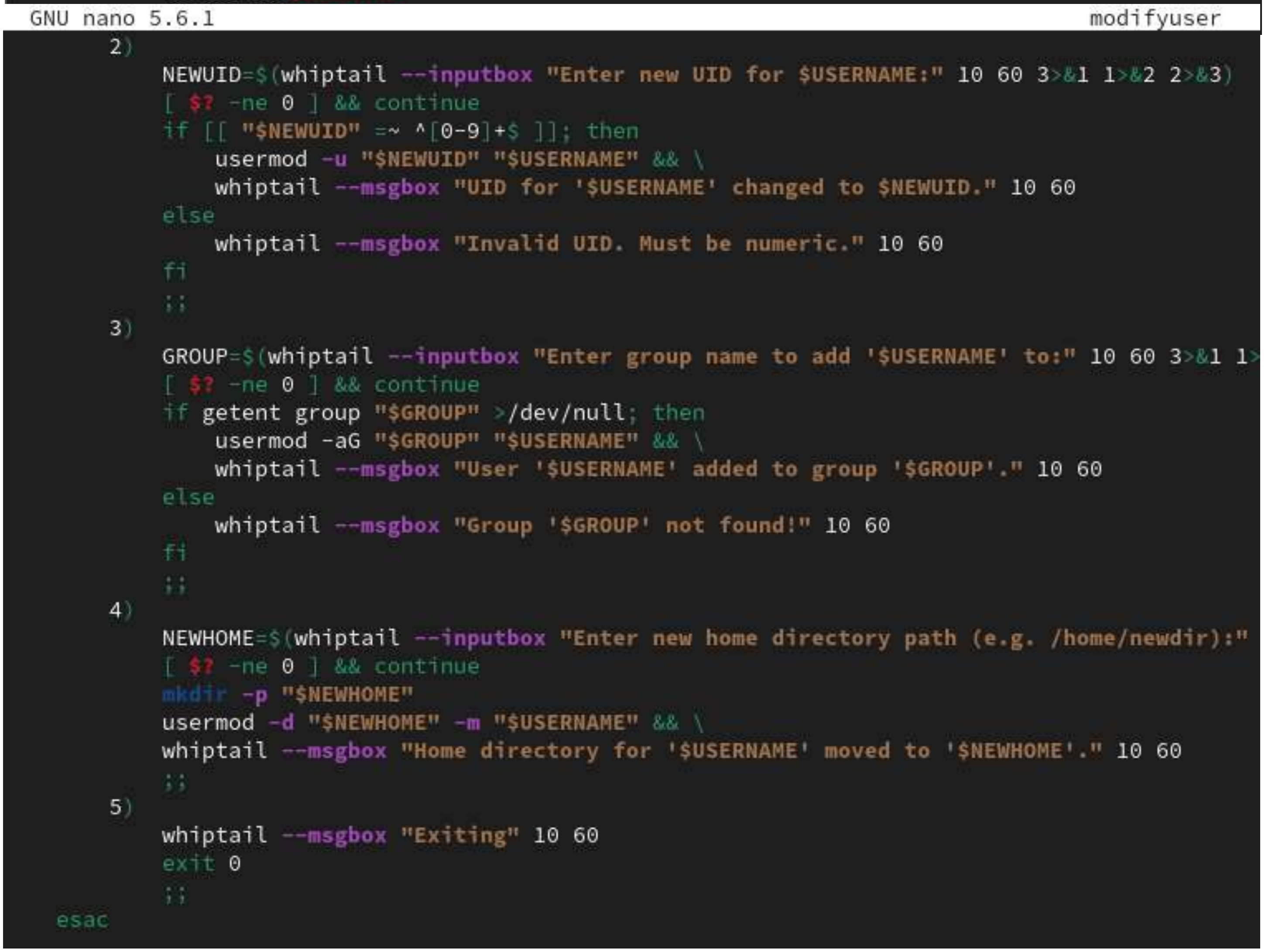


```
Enter group name to add 'Omar' to:
```

modifyuser.sh



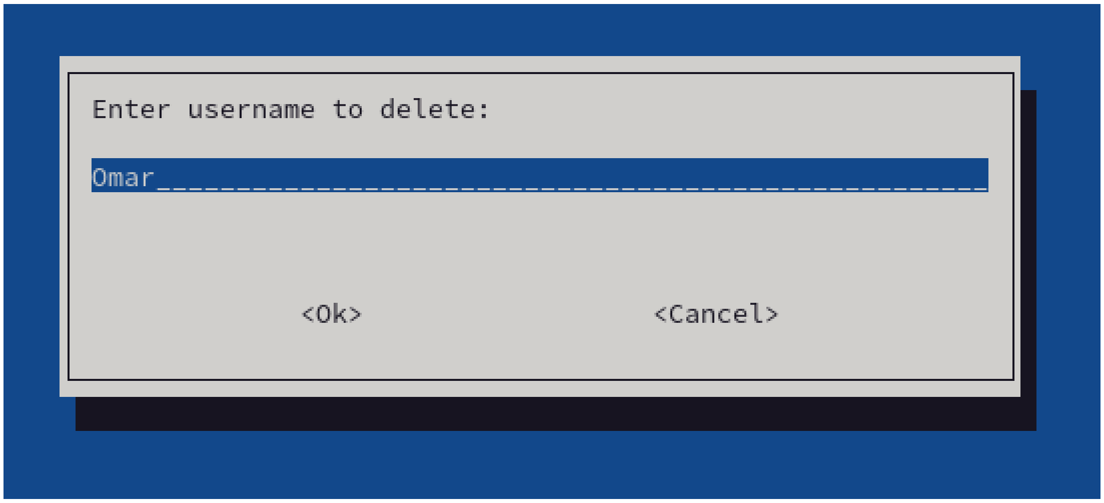
```
  GNU nano 5.6.1                                                                  modifyuser
#!/bin/bash
check_user() {
    if ! id "$USERNAME" &>/dev/null; then
        whiptail --msgbox "User '$USERNAME' not found!" 10 60
        exit 1
    fi
}


USERNAME=$(whiptail --inputbox "Enter the username you want to modify:" 10 60 3>&1 1>&2 2>&3)
[ $? -ne 0 ] && exit 0
check_user


while true; do
    CHOICE=$(whiptail --title "Modify User: $USERNAME" --menu "Choose an action:" 20 60 10 \
    "1" "Rename username" \
    "2" "Change UID" \
    "3" "Add to a group" \
    "4" "Move home directory" \
    "5" "Exit" 3>&1 1>&2 2>&3)

    [ $? -ne 0 ] && exit 0

    case $CHOICE in
        1)
            NEWNAME=$(whiptail --inputbox "Enter new username:" 10 60 3>&1 1>&2 2>&3)
            [ $? -ne 0 ] && continue
            usermod -l "$NEWNAME" "$USERNAME" && \
            whiptail --msgbox "Username changed from '$USERNAME' to '$NEWNAME'." 10 60
            USERNAME=$NEWNAME
```



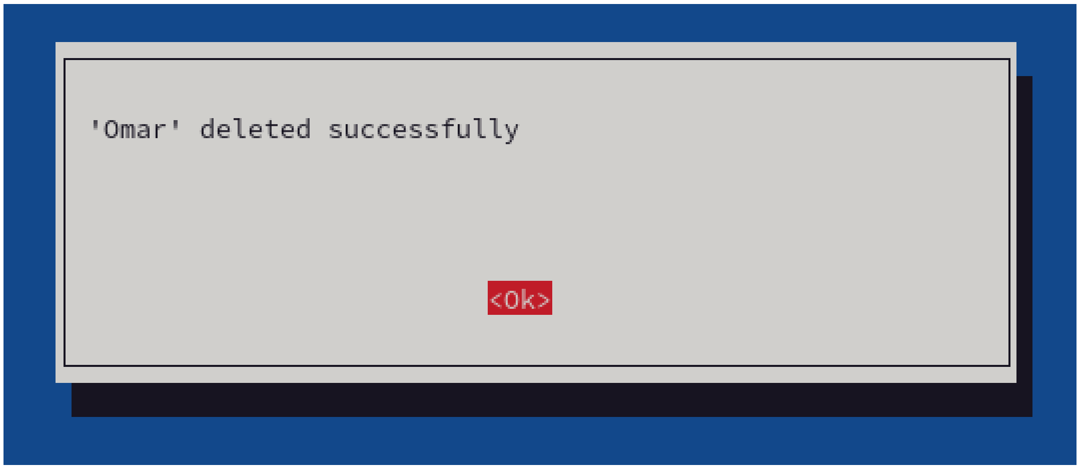
```
  GNU nano 5.6.1                                                                  modifyuser
        2)
            NEWUID=$(whiptail --inputbox "Enter new UID for $USERNAME:" 10 60 3>&1 1>&2 2>&3)
            [ $? -ne 0 ] && continue
            if [[ "$NEWUID" =~ ^[0-9]+$ ]]; then
                usermod -u "$NEWUID" "$USERNAME" && \
                whiptail --msgbox "UID for '$USERNAME' changed to $NEWUID." 10 60
            else
                whiptail --msgbox "Invalid UID. Must be numeric." 10 60
            fi
            ;;
        3)
            GROUP=$(whiptail --inputbox "Enter group name to add '$USERNAME' to:" 10 60 3>&1 1>
            [ $? -ne 0 ] && continue
            if getent group "$GROUP" >/dev/null; then
                usermod -aG "$GROUP" "$USERNAME" && \
                whiptail --msgbox "User '$USERNAME' added to group '$GROUP'." 10 60
            else
                whiptail --msgbox "Group '$GROUP' not found!" 10 60
            fi
            ;;
        4)
            NEWHOME=$(whiptail --inputbox "Enter new home directory path (e.g. /home/newdir):"
            [ $? -ne 0 ] && continue
            mkdir -p "$NEWHOME"
            usermod -d "$NEWHOME" -m "$USERNAME" && \
            whiptail --msgbox "Home directory for '$USERNAME' moved to '$NEWHOME'." 10 60
            ;;
        5)
            whiptail --msgbox "Exiting" 10 60
            exit 0
            ;;
    esac
```

Delete user (*deluser.sh*):

Deletes an existing user

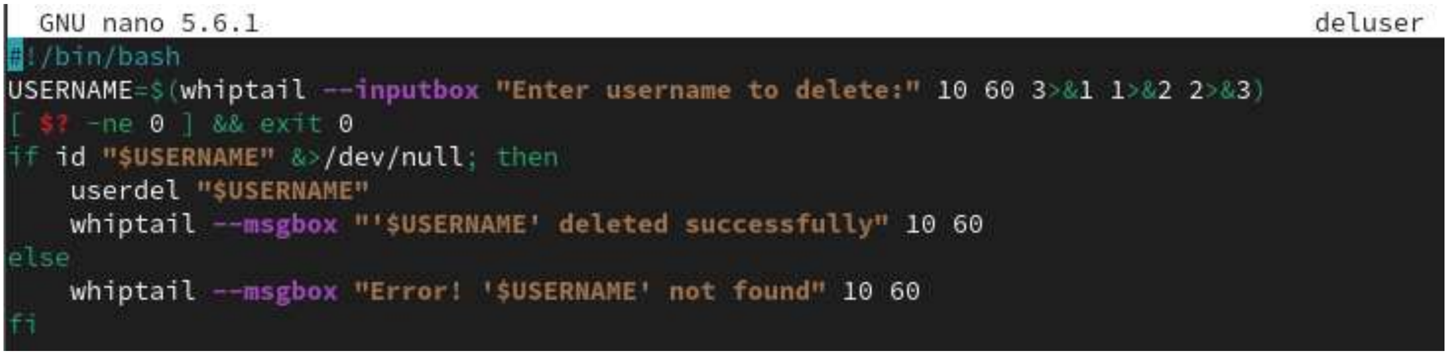

Enter username to delete:

Omar


'Omar' deleted successfully

deluser.sh

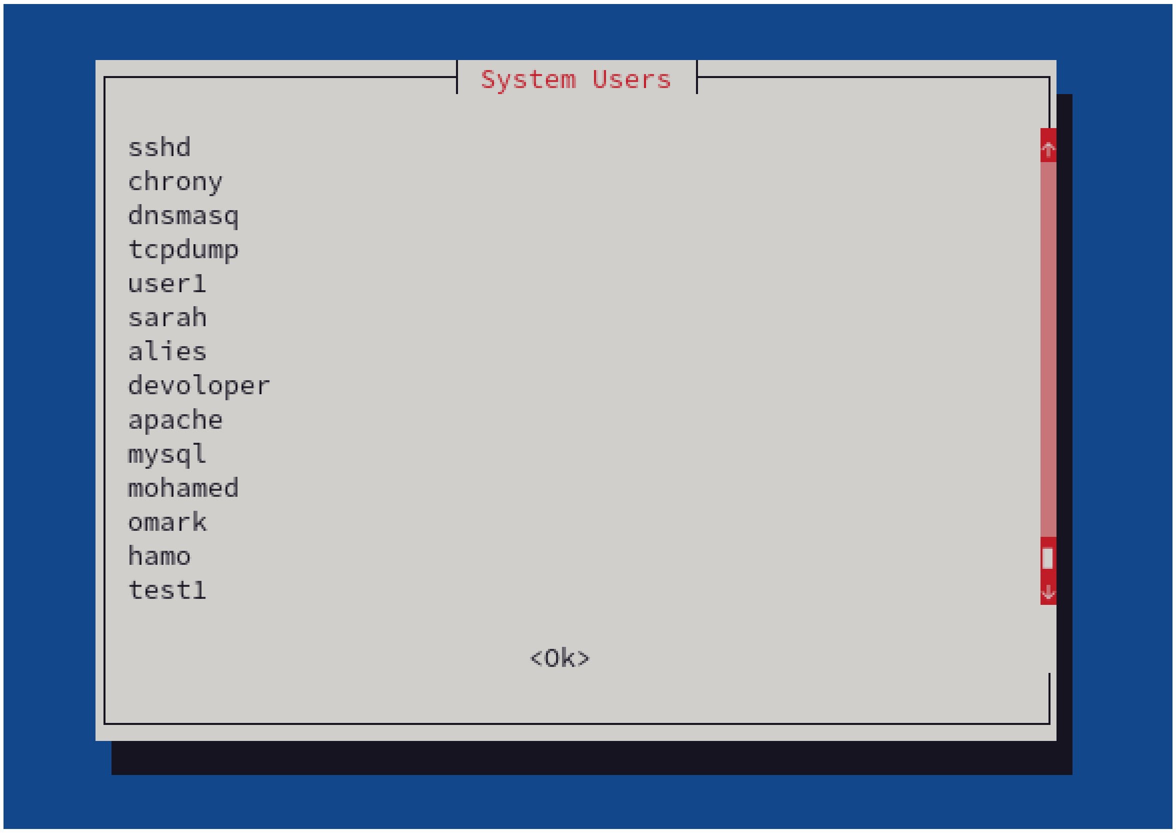
Checks if the user you entered exists then uses *userdel* to remove it and displays a confirmation message



```
  GNU nano 5.6.1                                                                    deluser
#!/bin/bash
USERNAME=$(whiptail --inputbox "Enter username to delete:" 10 60 3>&1 1>&2 2>&3)
[ $? -ne 0 ] && exit 0
if id "$USERNAME" &>/dev/null; then
    userdel "$USERNAME"
    whiptail --msgbox "'$USERNAME' deleted successfully" 10 60
else
    whiptail --msgbox "Error! '$USERNAME' not found" 10 60
fi
```

List users (*listusers.sh*):

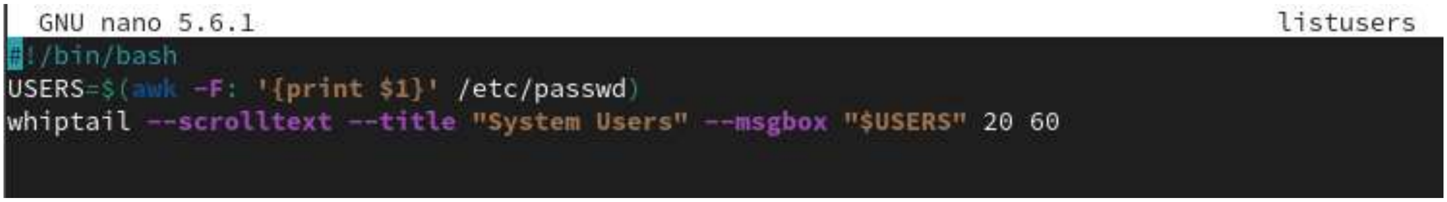
Shows a list of all system users



```
System Users

sshd
chrony
dnsmasq
tcpdump
user1
sarah
alies
devoloper
apache
mysql
mohamed
omark
hamo
test1

<Ok>
```

listusers.sh

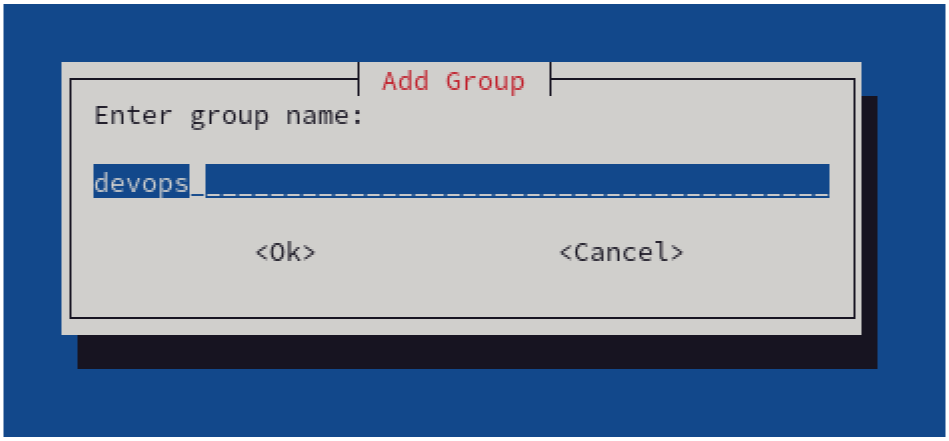
Extracts usernames from */etc/passwd* by using *awk* and displays it in a Whiptail textbox (scroll text).



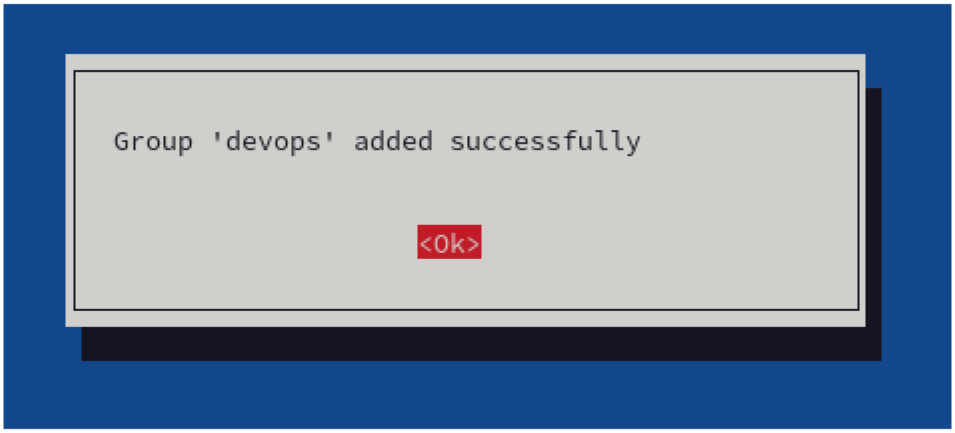
```
  GNU nano 5.6.1                                                    listusers
#!/bin/bash
USERS=$(awk -F: '{print $1}' /etc/passwd)
whiptail --scrolltext --title "System Users" --msgbox "$USERS" 20 60
```

Add Group (*addgroup.sh*):

Creates a new group in the system



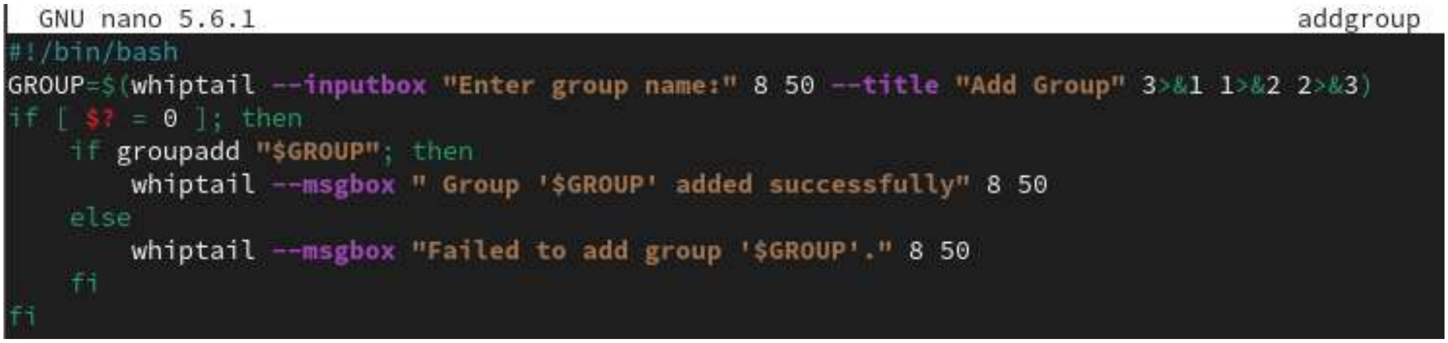
```
Add Group
Enter group name:

devops
```



```
Group 'devops' added successfully
```

addgroup.sh

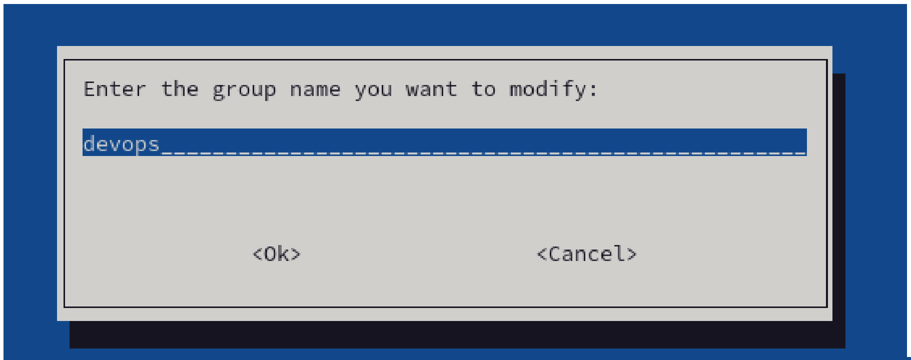
Allows the user to enter a group then displays a confirmation message, unless the group already exists then it will display a message that the process failed.



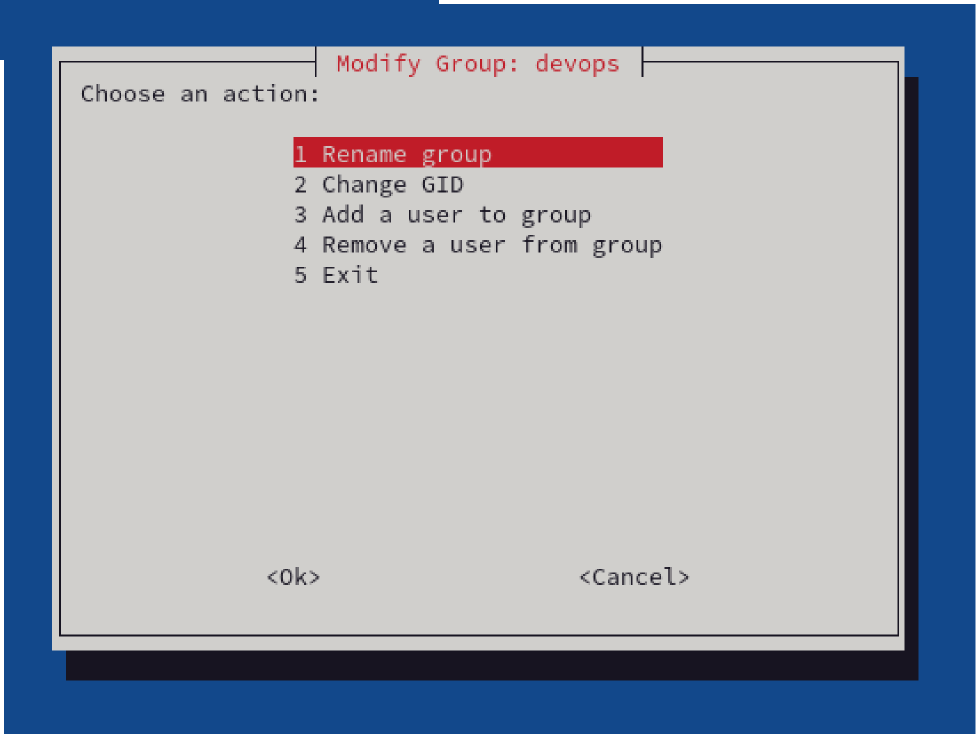
```
#!/bin/bash
GROUP=$(whiptail --inputbox "Enter group name:" 8 50 --title "Add Group" 3>&1 1>&2 2>&3)
if [ $? = 0 ]; then
    if groupadd "$GROUP"; then
        whiptail --msgbox " Group '$GROUP' added successfully" 8 50
    else
        whiptail --msgbox "Failed to add group '$GROUP'." 8 50
    fi
fi
```

Modify Group (*modifygroup.sh*):

Lets administrators modify group properties through a submenu

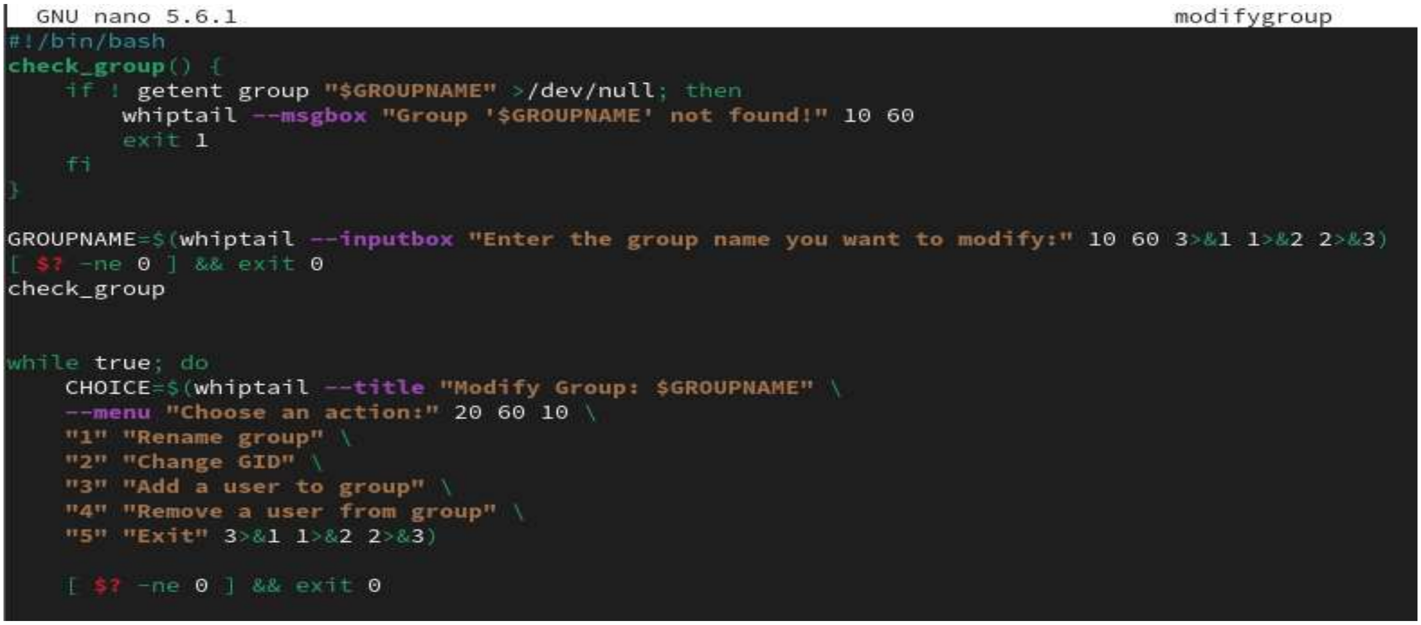


```
Enter the group name you want to modify:

devops

        <Ok>                    <Cancel>
```



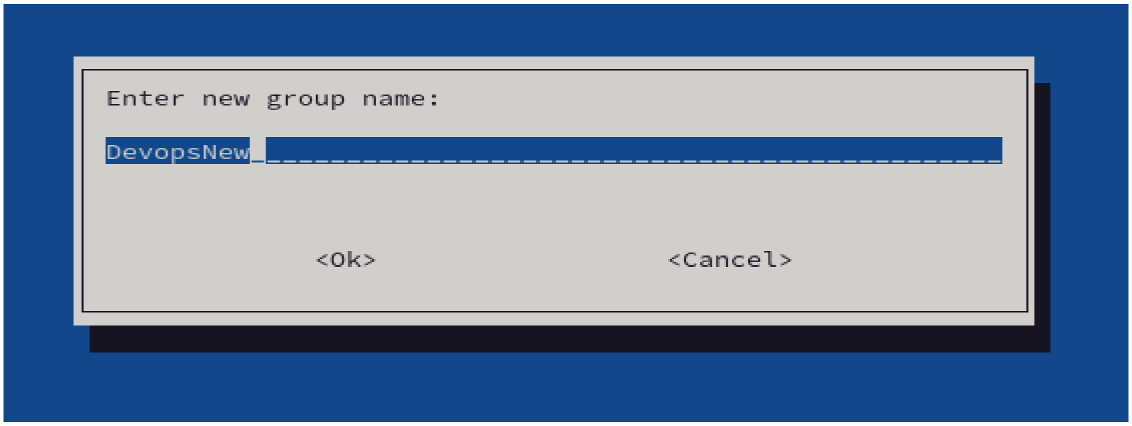
```
                 Modify Group: devops
Choose an action:

               1 Rename group
               2 Change GID
               3 Add a user to group
               4 Remove a user from group
               5 Exit

        <Ok>                    <Cancel>
```

Modifygroup.sh



```
GNU nano 5.6.1                                                           modifygroup
#!/bin/bash
check_group() {
    if ! getent group "$GROUPNAME" >/dev/null; then
        whiptail --msgbox "Group '$GROUPNAME' not found!" 10 60
        exit 1
    fi
}

GROUPNAME=$(whiptail --inputbox "Enter the group name you want to modify:" 10 60 3>&1 1>&2 2>&3)
[ $? -ne 0 ] && exit 0
check_group


while true; do
    CHOICE=$(whiptail --title "Modify Group: $GROUPNAME" \
    --menu "Choose an action:" 20 60 10 \
    "1" "Rename group" \
    "2" "Change GID" \
    "3" "Add a user to group" \
    "4" "Remove a user from group" \
    "5" "Exit" 3>&1 1>&2 2>&3)

    [ $? -ne 0 ] && exit 0
```

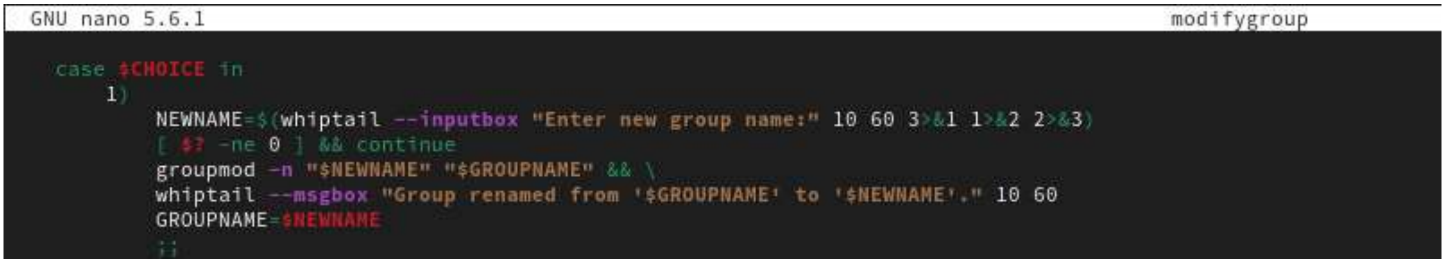
- Changing the group name.



Enter new group name:

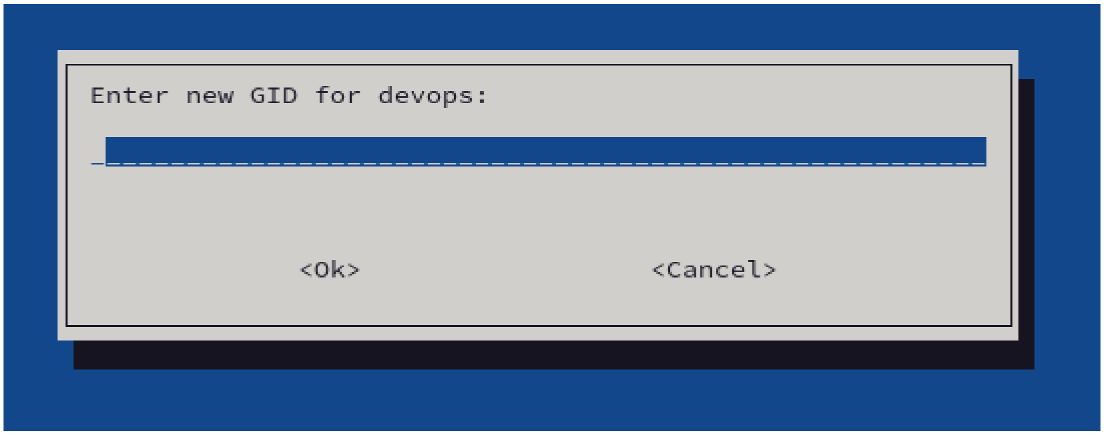
DevopsNew

When choosing the rename group option, it allows you to change the group name then displaying a confirmation message afterwards.



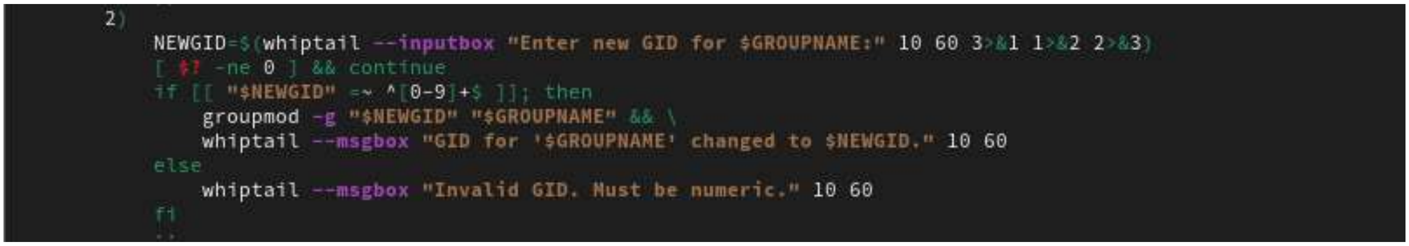
```
 GNU nano 5.6.1                                                                                    modifygroup

    case $CHOICE in
        1)
            NEWNAME=$(whiptail --inputbox "Enter new group name:" 10 60 3>&1 1>&2 2>&3)
            [ $? -ne 0 ] && continue
            groupmod -n "$NEWNAME" "$GROUPNAME" && \
            whiptail --msgbox "Group renamed from '$GROUPNAME' to '$NEWNAME'." 10 60
            GROUPNAME=$NEWNAME
            ;;
```

- Editing the GID.



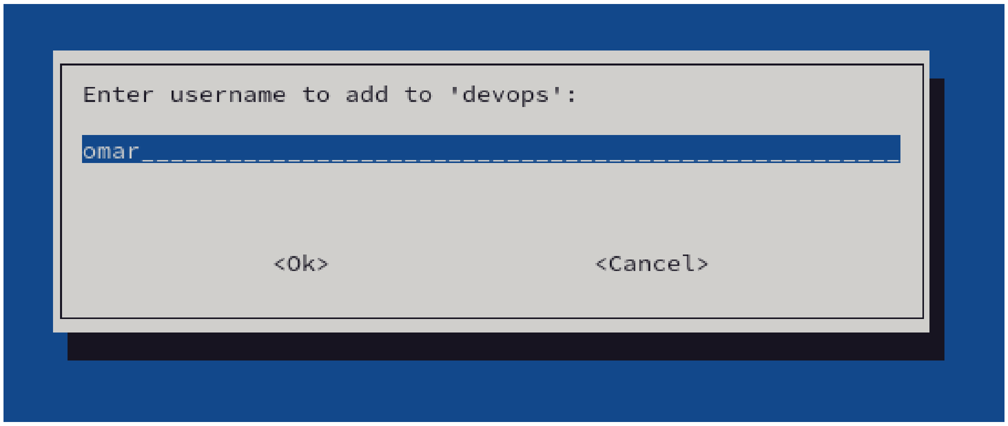
Enter new GID for devops:

When choosing the Change GID option, it allows you to change the group id to a new id but it has to be a numeric value, if so it displays a confirmation message about changing your GID



```
        2)
            NEWGID=$(whiptail --inputbox "Enter new GID for $GROUPNAME:" 10 60 3>&1 1>&2 2>&3)
            [ $? -ne 0 ] && continue
            if [[ "$NEWGID" =~ ^[0-9]+$ ]]; then
                groupmod -g "$NEWGID" "$GROUPNAME" && \
                whiptail --msgbox "GID for '$GROUPNAME' changed to $NEWGID." 10 60
            else
                whiptail --msgbox "Invalid GID. Must be numeric." 10 60
            fi
            ;;
```

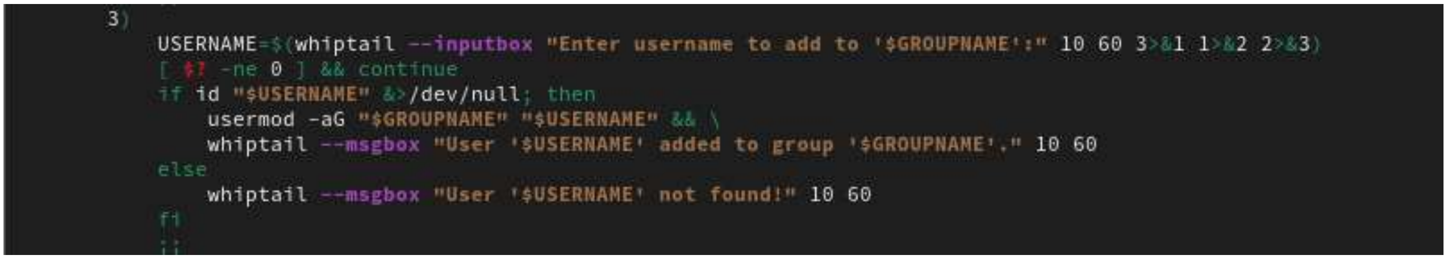
- Adding a user to a group.



Enter username to add to 'devops':

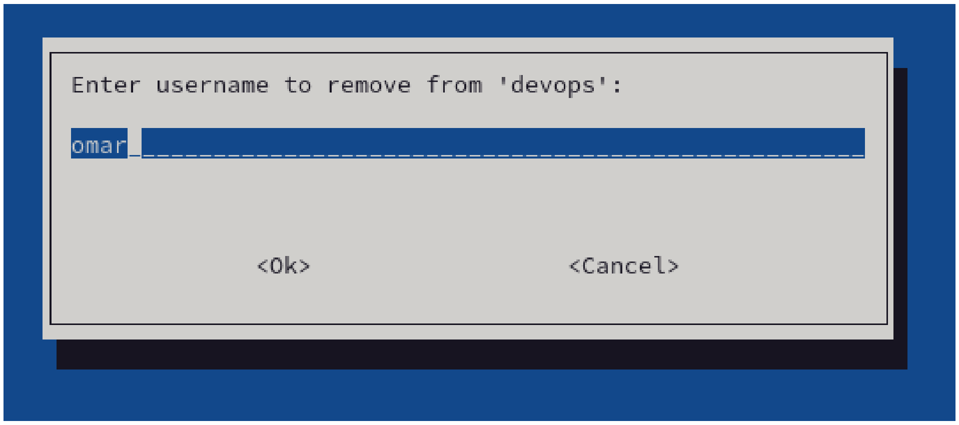
omar

When choosing the add user to group option, it tells you to enter the username you want to add to the group you're currently modifying and checks if it exists, if so it will display a confirmation message that the user was added successfully.



```
3)
    USERNAME=$(whiptail --inputbox "Enter username to add to '$GROUPNAME':" 10 60 3>&1 1>&2 2>&3)
    [ $? -ne 0 ] && continue
    if id "$USERNAME" &>/dev/null; then
        usermod -aG "$GROUPNAME" "$USERNAME" && \
        whiptail --msgbox "User '$USERNAME' added to group '$GROUPNAME'." 10 60
    else
        whiptail --msgbox "User '$USERNAME' not found!" 10 60
    fi
    ;;
```

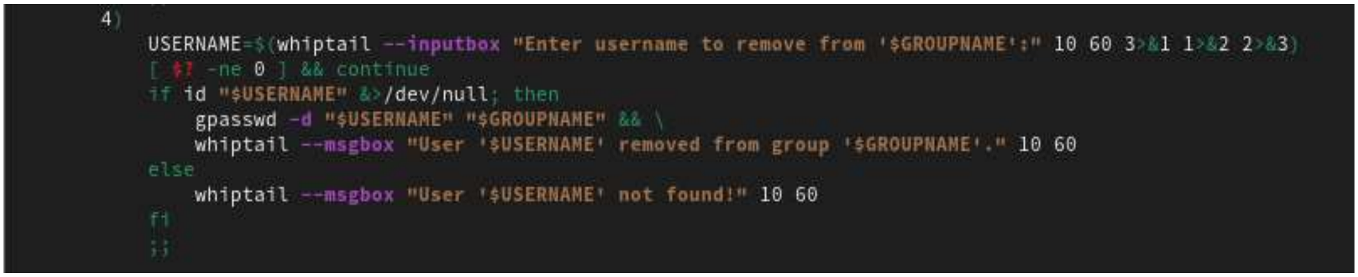
- Removing a user from the group



Enter username to remove from 'devops':

omar

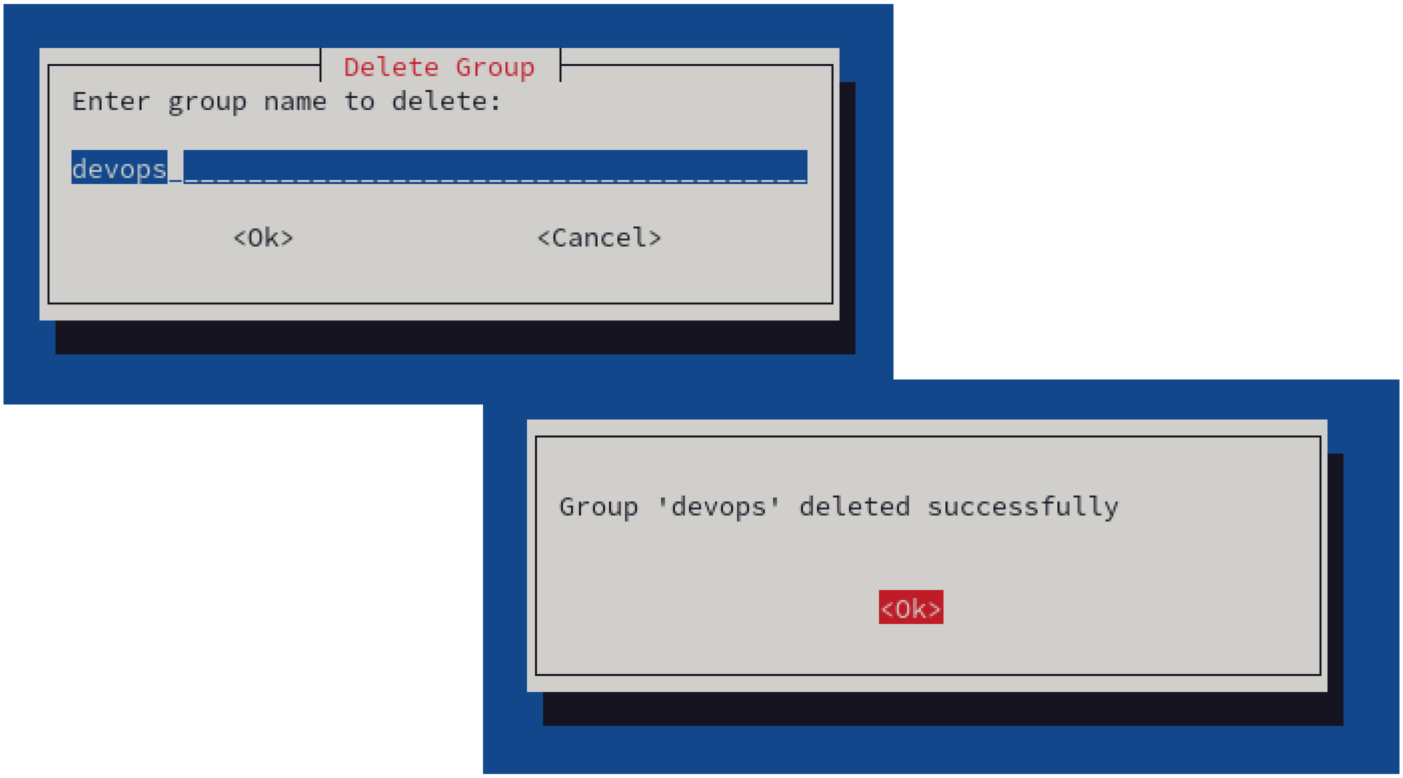
When choosing the remove user from group option, it tells you to enter the username you want to remove from the group you're currently modifying and checks if it exists, if so it will display a confirmation message that the user was removed successfully.



```
4)
    USERNAME=$(whiptail --inputbox "Enter username to remove from '$GROUPNAME':" 10 60 3>&1 1>&2 2>&3)
    [ $? -ne 0 ] && continue
    if id "$USERNAME" &>/dev/null; then
        gpasswd -d "$USERNAME" "$GROUPNAME" && \
        whiptail --msgbox "User '$USERNAME' removed from group '$GROUPNAME'." 10 60
    else
        whiptail --msgbox "User '$USERNAME' not found!" 10 60
    fi
    ;;
```

Delete Group (*deletegroup.sh*):

Removes a group from the system

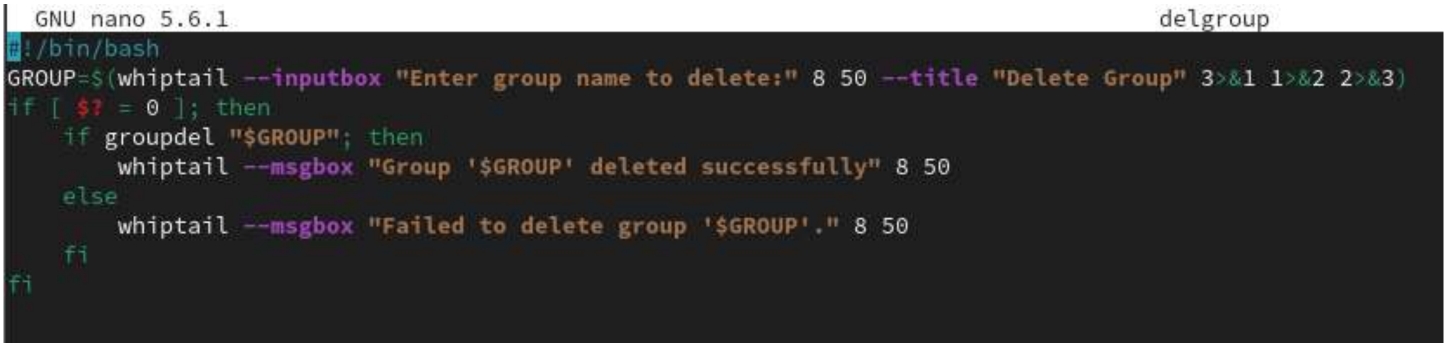


```
Delete Group
Enter group name to delete:

devops_

<Ok>                <Cancel>
```

```
Group 'devops' deleted successfully

<Ok>
```

deletegroup.sh

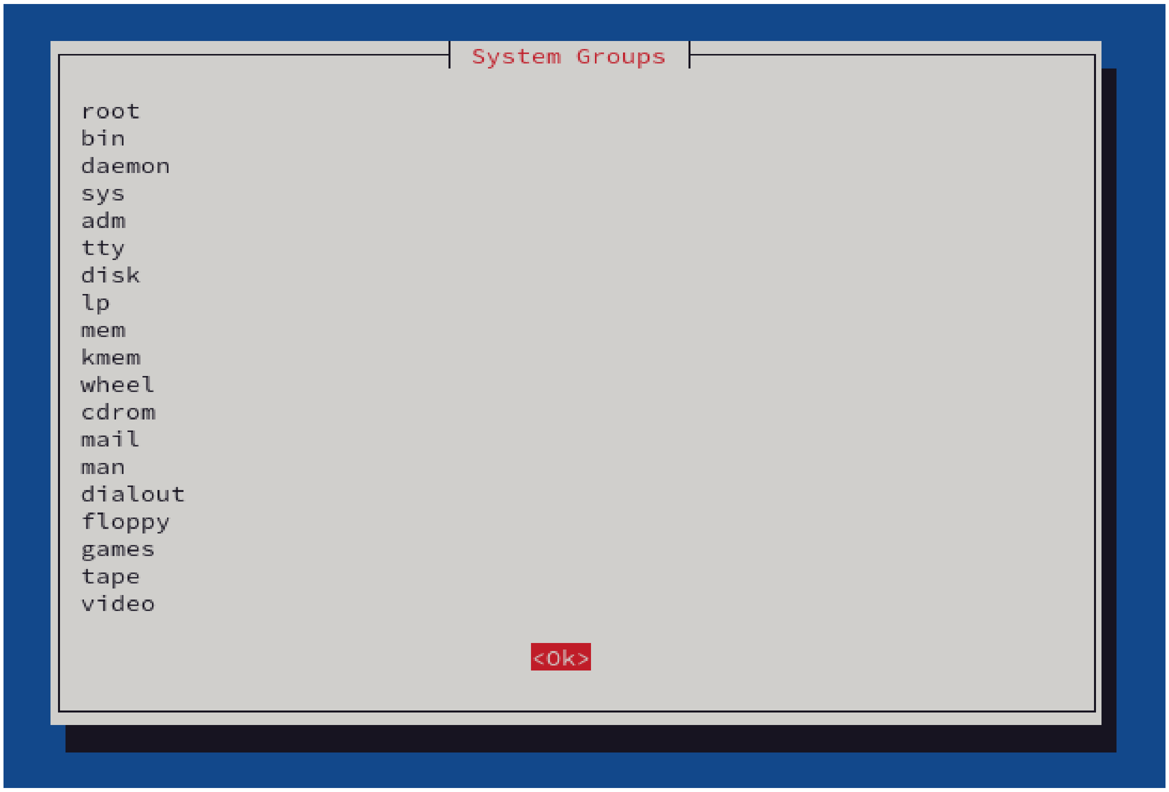
Allows the user to enter a group they want to delete and Executes *groupdel* to remove it, then displays a success or error message.



```
  GNU nano 5.6.1                                                              delgroup
#!/bin/bash
GROUP=$(whiptail --inputbox "Enter group name to delete:" 8 50 --title "Delete Group" 3>&1 1>&2 2>&3)
if [ $? = 0 ]; then
    if groupdel "$GROUP"; then
        whiptail --msgbox "Group '$GROUP' deleted successfully" 8 50
    else
        whiptail --msgbox "Failed to delete group '$GROUP'." 8 50
    fi
fi
```

List groups (*listgroup.sh*):

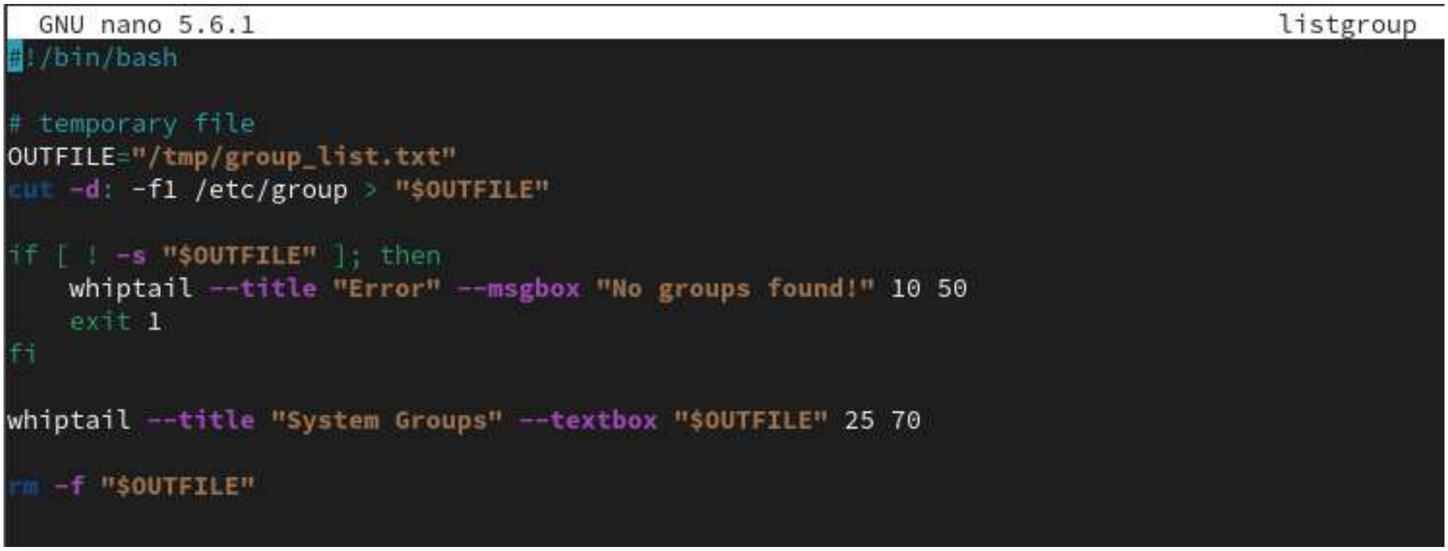
Shows a list of all system groups



```
root
bin
daemon
sys
adm
tty
disk
lp
mem
kmem
wheel
cdrom
mail
man
dialout
floppy
games
tape
video
```

listgroup.sh

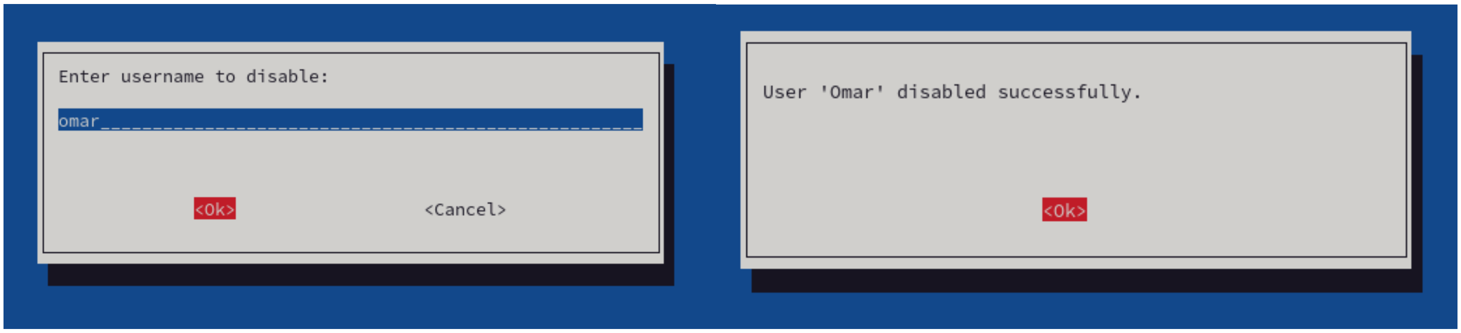
Extracts group names from */etc/group* using *cut*, *awk*, or *sed* and redirects them to a temporary file then displays the file content using *whiptail --textbox*



```
#!/bin/bash

# temporary file
OUTFILE="/tmp/group_list.txt"
cut -d: -f1 /etc/group > "$OUTFILE"

if [ ! -s "$OUTFILE" ]; then
    whiptail --title "Error" --msgbox "No groups found!" 10 50
    exit 1
fi

whiptail --title "System Groups" --textbox "$OUTFILE" 25 70

rm -f "$OUTFILE"
```

Disable User (*disuser.sh*):

Locks a user account so it cannot log in



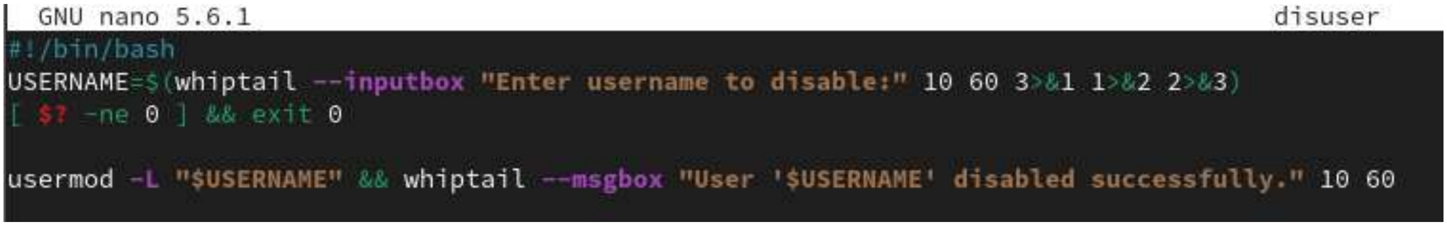
```
Enter username to disable:
omar
```

```
User 'Omar' disabled successfully.
```

disuser.sh

It asks for username and Displays a success message once locked.

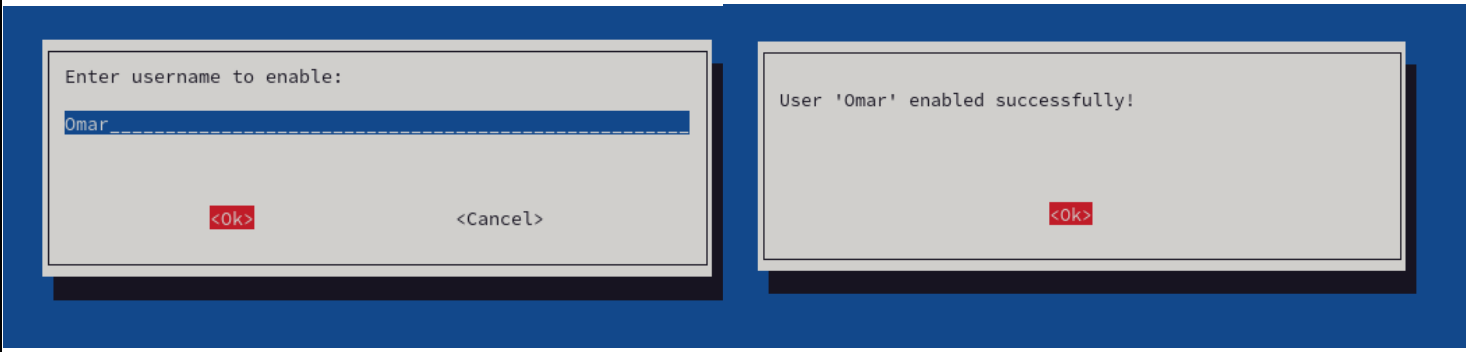
The *-L* in the *usermod –L $USERNAME* locks the password by adding a **!** in */etc/shadow*



```
  GNU nano 5.6.1                                                   disuser
#!/bin/bash
USERNAME=$(whiptail --inputbox "Enter username to disable:" 10 60 3>&1 1>&2 2>&3)
[ $? -ne 0 ] && exit 0

usermod -L "$USERNAME" && whiptail --msgbox "User '$USERNAME' disabled successfully." 10 60
```

Enable User (*user.sh*):

Unlocks a previously locked user account.



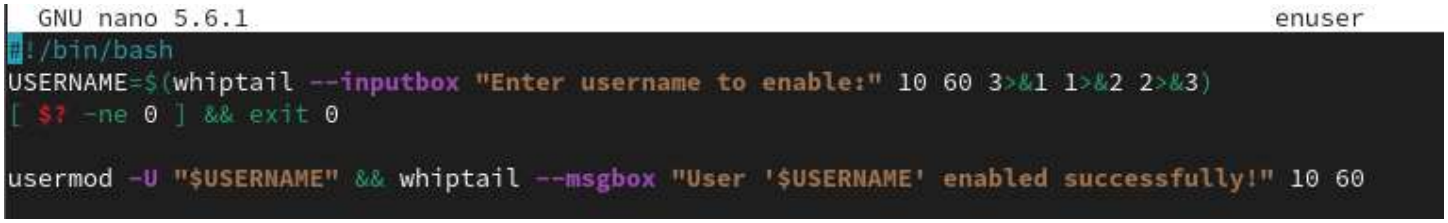
```
Enter username to enable:
Omar
```

```
User 'Omar' enabled successfully!
```

enuser.sh

It asks for username and Displays a success message once unlocked or enabled.

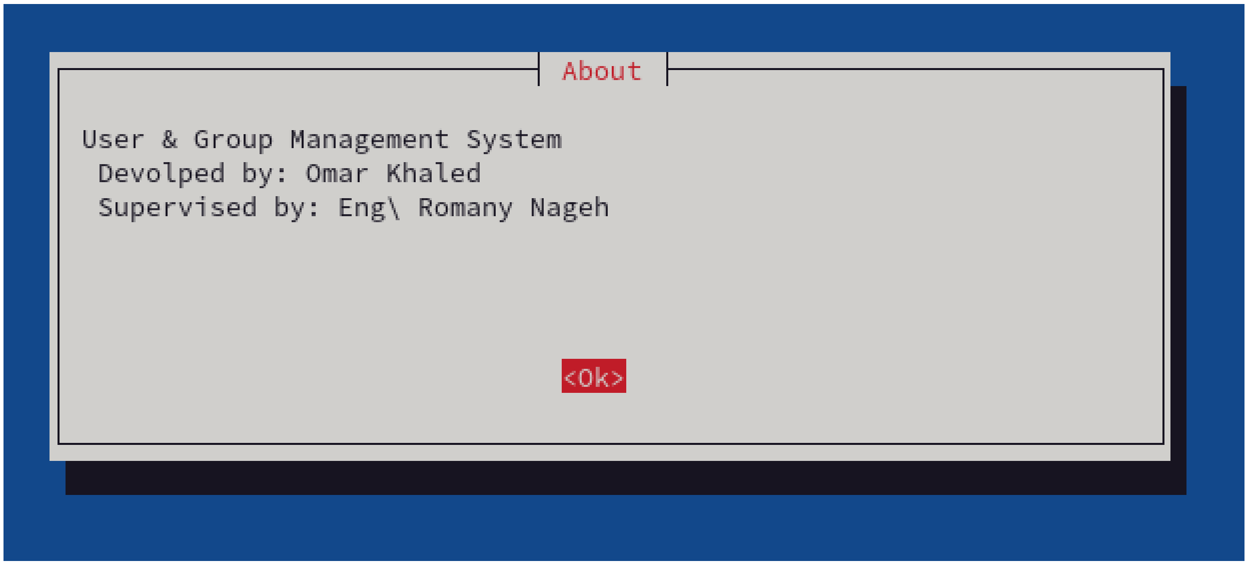
The *-U* in the *usermod –U $USERNAME* unlocks the password by removing the **!** in */etc/shadow*



```
  GNU nano 5.6.1                                                   enuser
#!/bin/bash
USERNAME=$(whiptail --inputbox "Enter username to enable:" 10 60 3>&1 1>&2 2>&3)
[ $? -ne 0 ] && exit 0

usermod -U "$USERNAME" && whiptail --msgbox "User '$USERNAME' enabled successfully!" 10 60
```

About Menu (*about.sh*):

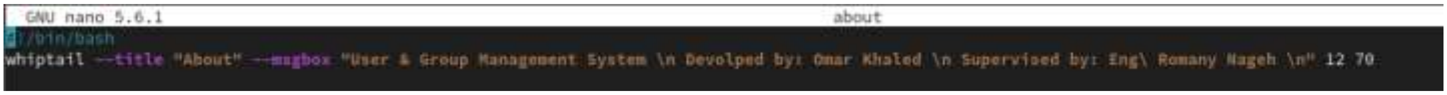
Displays details about the project and author



```
User & Group Management System
 Devolped by: Omar Khaled
 Supervised by: Eng\ Romany Nageh
```

about.sh

Displaying the content you need using *whiptail --msgbox*



```
#!/bin/bash
whiptail --title "About" --msgbox "User & Group Management System \n Devolped by: Omar Khaled \n Supervised by: Eng\ Romany Nageh \n" 12 70
```