PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
100 Ft. Road, BSK III Stage, Bengaluru – 560 085

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Student Name: PARIMALA S

Roll No.: 56

Section : 2-M

SRN : PES1UG19CS323

ASSIGNMENT NO: 9

QUESTION:

Write a program in C to mimic the “delgroup” command on Linux. This command will delete a group. It has to handle 2 files “group” and “gshadow”. Both these files will be in some folder specified by an environment variable PFILE. The program has to take all arguments as command line arguments (Refer man pages for the command line arguments)

The program has to delete all references to this group in the files group and shadow. Copy these two files /etc/group and /etc/gshadow from any Linux system into another folder and then execute your program.

delgroup GName (The only argument is the name of the group being deleted)

DESCRIPTION:

This program asks the user to input a groupname and the group's information from the two files named group,gshadow will be removed.If the group is not present it will not edit the file.If present , it deletes the references to the group if its not the users primary group. All the values are validated and errors handled in letter and spirit as much possible.

STEPS TO BE FOLLOWED BEFORE AND DURING EXECUTION (Assumptions/Conditions):

1. Place group,gshadow and password files in the path provided by environment variable PFILE. Put the files in that specific folder.

2. In case of Windows, it is by default set as PFILE = C:\PESU. It can be changed if user wishes.

3. For Ubuntu user, the provision has been done to change the value of the environment variable as per their file path in ubuntu. The corresponding code line- setenv() is commented. Just uncomment it to run on Ubuntu

4. Command for executing make file on windows:

mingw32-make -f makefile_windows.mak

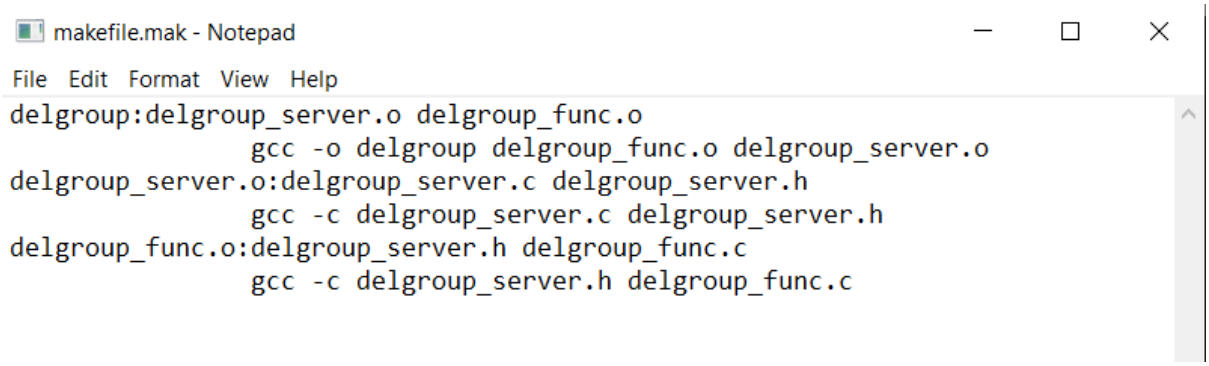
on Ubuntu: make -f makefile_ubuntu.mk

5.Two makefiles have been made : one for ubuntu with file extension .mk and one for windows with file extension .mak.

6. After executing this line, enter delgroup <groupname> to view the required results for windows. Enter _delgroup_ <groupname> for Ubuntu. Here, the first word is changed because if we enter “delgroup” in ubuntu , Linux will consider the system command delgroup and not our executable file.

CODES:

makefile.mk :



delgroup_func.c:

delgroup_func - Notepad

File Edit Format View Help

```
//PROGRAMME TO MIMIC THE "delgroup" COMMAND IN LINUX USING C

/*This program asks the user to input a groupname and the group's information from the two files named
group,gshadow will be removed.If the group is not present it will not edit the file.If present , it deletes
the references to the group if its not the users primary group.*/

/*STEPS TO BE FOLLOWED BEFORE AND DURING EXECUTION
1. Place group,gshadow and password files in the path provided by environment variable PFILE. Put the files in that specific folder.
2. In case of Windows, it is by default set as PFILE = C:\PESU. It can be changed if user wishes.
3. For Ubuntu user, the provision has been done to change the value of the environment variable as per their file path in ubuntu.
The corresponding code line is commented.Just uncomment it to run on Ubuntu
4. Command for executing make file on windows: mingw32-make -f delgroup_makefile.mak
on Ubuntu : make -f delgroup_makefile.mk
5. After executing this line , enter delgroup <groupname> to view the required results
*/

#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include"delgroup_server.h"

int primary_group(char groupname[], char * pfile)                    //Function definition to check if the group entered is primary
{
        int flag = 0;                                                //flag variable
        int passwd_line = 1, gname_len, iter_len;                    //variables to store length of groupname and input buffer from file
        char colon = ':';
        gname_len = strlen(groupname);                               //stores length of groupname
        char iter_str[100], gname[20];                               //variable of type string to store input read from file

        strcpy(gname,groupname);                                     //copying groupname to another string for further modification
        strncat(gname,&colon,1);

        FILE *password = fopen(pfile,"r");                           //Opening password file to verify if the groupname is present
         if (password == NULL)
    {
        printf ("Error opening %s file for reading\n", password);    //Handling error in opening files
        exit (1);
    }
```

```
  while(fgets(iter_str, gname_len+2, password)!=NULL)
                                                          /*The while loop looks for the line where the group comes and
                                                          stores it in passwd_line. If the group is found the flag
                                                          variable is set to 1. The final value of flag is returned
                                                          to the delgroup function - from where it is called */

 {
   iter_len = strlen(iter_str);

   if(iter_str[iter_len-1] =='\n')                        //variable storing count of lines is incremented after every \n
   passwd_line++;
   if(strcmp(iter_str, gname)==0)                         //groupname if found sets flag to 1 and breaks out of the loop
       {
       flag = 1;
       break;
       }
 }
   return flag;                                           //returned value is flag and of type - int
}


void update_file(char * gfile, char * gsfile, int gshadline, int group_line, char * replica_grp, char * replica_gshad )
{                                                         /*update_file function is called only when group is found in the
                                                          group,gshadow files.It removes the group details in the files*/

 FILE *newgshadow;                                        //File pointers to group and gshadow files
 FILE *newgroup;
 FILE *gshadow = fopen(gsfile,"r");
 if (gshadow == NULL)                                     //Error handling in opening files gshadow,group
   {
       printf ("Error opening %s file for reading\n", gshadow);
       exit (1);
   }

FILE *group = fopen(gfile,"r");
 if (group == NULL)
   {
       printf ("Error opening %s file for reading\n", group);
       exit (1);
   }
```

```
newgshadow = fopen(replica_grp,"w");                    //Opening 2 new filepointers in write mode to copy group &gshadow files
if (newgshadow == NULL)
  {
     printf ("Error opening %s file for writing\n", newgshadow);    //Error handling in opening the copy files of group,gshadow
     exit (1);
  }

newgroup = fopen(replica_gshad,"w");
if (newgroup == NULL)
  {
     printf ("Error opening %s file for writing\n", newgroup);
     exit (1);
  }


char iter_ch1, iter_ch2;                                //These characters are for iterating through the 2 files
int gsn=1, gn=1;                                        //Variables to keep count of iterating line nos in both files

while ((iter_ch1 = fgetc(gshadow)) != EOF)              /*This while loop copies the content of existing gshadow file
                                                        to the copy-gshadow file except the line where groupname is found*/
{
  if (iter_ch1 == '\n')
    gsn++;
  if (gsn != gshadline)                                 /*By applying this logic, the line with groupdetails is
                                                        omitted from writing in the copied file*/
  {
    fputc(iter_ch1,newgshadow);
  }

}

while ((iter_ch2 = fgetc(group)) != EOF)                /*This while loop copies the content of existing gshadow file
                                                        to the copy-gshadow file except the line where groupname is found*/
{
  if (iter_ch2 == '\n')
    gn++;
```

```
    if (gn != group_line)                          /*By applying this logic, the line with groupdetails is
                                                   omitted from writing in the copied file*/

    {
        fputc(iter_ch2,newgroup);
    }
  }
  fclose(gshadow);                                 //closing all the opened files - both original and copied
  fclose(newgshadow);
  fclose(group);
  fclose(newgroup);
  }


void delgroup(char groupname[],char *gfile,char *gsfile)   /*Defining the fucntion to mimic the delgroup command
                                                            which takes groupname and group,gshadow file paths
                                                            as parameters,called from main() */
{
  int gshadline = 1, group_line = 1, gname_len, iter_len, value = 0;   /*variables gshadline,group_line to store count of no of
                                                                         iterated lines in the files and gname_len,iter_len variables
                                                                        to store lengths of groupname and input buffer read
                                                                        from files*/

  char colon = ':';
  gname_len = strlen(groupname);                   //stores length of groupname
  char iter_str1[100], iter_str2[100], gname[20], t_password[80];   //char arrays iter_str1 and iter_str2 to store input read from file

  strcpy(gname,groupname);                         //copying groupname to another string for further modification
  strncat(gname,&colon,1);
  printf("\n %s",gname);

  char replica_grp[80] , replica_gshad[80] ;       //char array variables to store duplicate filenames of group,gshadow
  sprintf (replica_grp, "%s\\%s", getenv("PFILE"), "replica1");   //creating filenames with path for the copy files
  sprintf (replica_gshad, "%s\\%s", getenv("PFILE"), "replica2");

  FILE *gshadow = fopen(gsfile,"r");
  if (gshadow == NULL)
    {
        printf ("Error opening %s file for reading\n", gshadow);   //Opening and handling error in opening files for group,gshadow in read mode
        exit (1);
    }
```

```
FILE *group = fopen(gfile,"r");
if (group == NULL)
  {
      printf ("Error opening %s file for reading\n", group);
      exit (1);
  }

sprintf (t_password, "%s\\%s", getenv ("PFILE"), "passwd");          //setting filename with env variable PFILE for the password file for access

 while(fgets(iter_str1, gname_len+2, gshadow) !=NULL)

                                                                     /*This while loop reads the content of existing gshadow file
                                                                     and find the line where the groupname is found. It sets the (flag)value
                                                                     to 1 ifits a primary group, -1 if group is going  to be removed,
                                                                     0 otherwise.*/
{
  iter_len = strlen(iter_str1);                                      //iter_len stores length of input read(string) ,used to compare & find grpname
  if(iter_str1[iter_len -1] == '\n')
  gshadline++;                                                       //line count variable is incremented,stores the final value once it
                                                                     //finds the groupname
  if(strcmp(iter_str1,gname) == 0)
  {

  printf("GROUP FOUND!");                                            //groupname is found by strcmp(),if found, checks if group is primary
  if(primary_group(groupname,t_password))                            //function call to primary_group function
      {
      printf("\n It is a Primary Group! Can not be removed...");     //display statement
      value = 1;
      return;
      }
  else
      {
      value = -1;
      printf("\n %s Being Removed...\n Done.",groupname);            //display statement
      }
  break;
  }

}
```

```
  if(value == 0)
  {
    printf("GROUP NAME NOT FOUND!");                    //If group not found,it exits the function
    return;
  }

  while(fgets(iter_str2, gname_len+2, group) != NULL)   /*This while loop is to check if the group details are there
                                                        in group file or not. The line in which it is found is stored
                                                        in group_line*/
  {
    iter_len = strlen(iter_str2);
    if(iter_str2[iter_len -1] == '\n')
    group_line++;
    if(strcmp(iter_str2,gname) == 0)
    break;
  }
  fclose(gshadow);                                      //closing the opened files
  fclose(group);


  update_file(gfile,gsfile,gshadline,group_line,replica_grp,replica_gshad);
                                                        //update_file function call to remove group details from files
  printf("\n Removing the Replicated Files...");

                                                        //Removing the files which are unedited

  if(remove(gsfile) == -1)      printf("Error removing replica_gshadow file\n");
  if(remove(gfile) == -1)       printf("Error removing replica_group file\n");

  rename(replica_grp,gsfile);                           //Now renaming the 2 copied files to the original name
                                                        //They contain all the group details except the one to be deleted
  rename(replica_gshad,gfile);
}
```

delgroup_server.c:

delgroup_server - Notepad

File Edit Format View Help

```
//PROGRAMME TO MIMIC THE "delgroup" COMMAND IN LINUX USING C

/*This program asks the user to input a groupname and the group's information from the two files named
group,gshadow will be removed.If the group is not present it will not edit the file.If present , it deletes
the references to the group if its not the users primary group.*/

/*STEPS TO BE FOLLOWED BEFORE AND DURING EXECUTION
1. Place group,gshadow and password files in the path provided by environment variable PFILE. Put the files in that specific folder.
2. In case of Windows, it is by default set as PFILE = C:\PESU. It can be changed if user wishes.
3. For Ubuntu user, the provision has been done to change the value of the environment variable as per their file path in ubuntu.
The corresponding code line is commented.Just uncomment it to run on Ubuntu
4. Command for executing make file on windows: mingw32-make -f delgroup_makefile.mak
on Ubuntu : make -f delgroup_makefile.mk
5. After executing this line , enter delgroup <groupname> to view the required results
*/



#include"delgroup_server.h"
#include<stdio.h>
#include<stdlib.h>
#include<string.h>

int main(int argc,char *argv[])                         //main function takes the groupname as the main command line argument
{
  char *temp,*groupname;                                //pointer variable declared to store groupname and environment variable value
  char p_group [80], p_gshadow [80];                    //variables to store path of files

  if(argc==2)                                           //Validation of Input entered
       groupname = argv[1];
  else if(argc<2)                                       //Handling error when there are less no of arguments
  {
    printf("INVALID INPUT : Less Arguments given!\n Key in the command in the format\n delgroup groupname");
    exit(1);
  }
  else                                                  //Handling error when there are more no of arguments
  {
    printf("INVALID INPUT : Too Many Arguments given!\n Key in the command in the format\n delgroup groupname");
    exit(1);
  }
  putenv("PFILE=C:\\PESU");                             //setting the environment variable by default to C:\PESU in case of windows
```

```
  //setenv("PFILE","\\home\\desktop",1);
                                                    /*uncomment the above statement to make it work absolutely error free
                                                     on Ubuntu - setting a diffrent suitable PFILE value for Ubuntu*/
  if ((temp = getenv ("PFILE")) == NULL)
    {
        printf ("The environment variable PFILE has not been set\n");   //error handling if PFILE is not set
        exit (1);
    }
  sprintf (p_group, "%s\\%s", temp, "group");                           //Setting the path-filenames of group and gshadow to PFILE
  sprintf (p_gshadow, "%s\\%s", temp, "gshadow");

  delgroup(groupname,p_group,p_gshadow);                                //Function Call to delgroup
  return 0;
}
```

SCREENSHOTS OF OUTPUT:

```
C:\TDM-GCC-64>mingw32-make -f makefile.mak
gcc -c delserver.c delserver.h
gcc -c delserver.h delmain.c
gcc -o delgroup delmain.o delserver.o
```

If PFILE IS NOT SET:

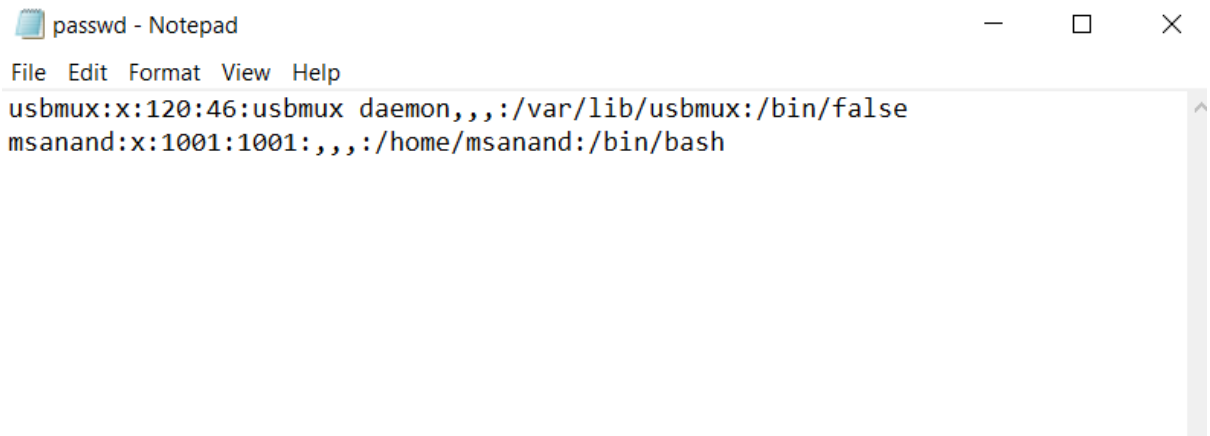
```
C:\TDM-GCC-64>delgroup bin
The environment variable PFILE has not been set
```

IF FILE IS NOT OPENING:

```
C:\TDM-GCC-64>delgroup logs

 logs:Error opening (null) file for reading
```

```
C:\TDM-GCC-64>delgroup
INVALID INPUT : Less Arguments given!
 Key in the command in the format
 delgroup groupname
C:\TDM-GCC-64>delgroup sys bro
INVALID INPUT : Too Many Arguments given!
 Key in the command in the format
 delgroup groupname
C:\TDM-GCC-64>delgroup 62486

 62486:GROUP NAME NOT FOUND!
C:\TDM-GCC-64>delgroup @#$%^&*(

 @#$%&*(:GROUP NAME NOT FOUND!
C:\TDM-GCC-64>delgroup adm

 adm:GROUP FOUND!
 adm Being Removed...
 Done.
 Removing the Replicated Files...
C:\TDM-GCC-64>delgroup msanand

 msanand:GROUP FOUND!
 It is a Primary Group! Can not be removed...
```

Password file has been used here to check for primary groups:



Gshadow and group before deletion:

gshadow - Notepad

File Edit Format View Help

```
root:*::
daemon:*::
bin:*::
sys:*::
adm:*::syslog,anandms
tty:*::
disk:*::
lp:*::
mail:*::
news:*::
uucp:*::
man:*::
proxy:*::
kmem:*::
dialout:*::
fax:*::
voice:*::
cdrom:*::anandms
floppy:*::
tape:*::
sudo:*::anandms
audio:*::pulse
dip:*::anandms
www-data:*::
backup:*::
operator:*::
list:*::
irc:*::
src:*::
gnats:*::
shadow:*::
utmp:*::
video:*::
sasl:*::
plugdev:*::anandms
staff:*::
games:*::
users:*::
nogroup:*::
systemd-journal:!::
systemd-timesync:!::
systemd-network:!::
systemd-resolve:!::
systemd-bus-proxy:!::
input:!::
crontab:!::
syslog:!::
netdev:!::
messagebus:!::
uuidd:!::
ssl-cert:!::
lpadmin:!::anandms
lightdm:!::
nopasswdlogin:!::
ssh:!::
whoopsie:!::
mlocate:!::
avahi-autoipd:!::
avahi:!::
scanner:!::saned
colord:!::
pulse:!::
pulse-access:!::
rtkit:!::
saned:!::
sambashare:!::anandms
msanand:!::
```

group - Notepad

File Edit Format View Help

```
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,anandms
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:anandms
floppy:x:25:
tape:x:26:
sudo:x:27:anandms
audio:x:29:pulse
dip:x:30:anandms
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
irc:x:39:
src:x:40:
gnats:x:41:
shadow:x:42:
utmp:x:43:
video:x:44:
sasl:x:45:
plugdev:x:46:anandms
staff:x:50:
games:x:60:
users:x:100:
nogroup:x:65534:
systemd-journal:x:101:
systemd-timesync:x:102:
systemd-network:x:103:
systemd-resolve:x:104:
systemd-bus-proxy:x:105:
input:x:106:
crontab:x:107:
syslog:x:108:
netdev:x:109:
messagebus:x:110:
uuidd:x:111:
ssl-cert:x:112:
lpadmin:x:113:anandms
lightdm:x:114:
nopasswdlogin:x:115:
ssh:x:116:
whoopsie:x:117:
mlocate:x:118:
avahi-autoipd:x:119:
avahi:x:120:
scanner:x:122:saned
colord:x:123:
pulse:x:124:
pulse-access:x:125:
rtkit:x:126:
saned:x:127:
sambashare:x:128:anandms
msanand:x:1001:
```

Group, gshadow files after deletion of sys group and after trying to delete msanand (primary)grp:

group - Notepad

File Edit Format View Help

```
root:x:0:
daemon:x:1:
bin:x:2:
adm:x:4:syslog,anandms
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:anandms
floppy:x:25:
tape:x:26:
sudo:x:27:anandms
audio:x:29:pulse
dip:x:30:anandms
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
irc:x:39:
src:x:40:
gnats:x:41:
shadow:x:42:
utmp:x:43:
video:x:44:
sasl:x:45:
plugdev:x:46:anandms
staff:x:50:
games:x:60:
users:x:100:
nogroup:x:65534:
systemd-journal:x:101:
systemd-timesync:x:102:
systemd-network:x:103:
systemd-resolve:x:104:
systemd-bus-proxy:x:105:
input:x:106:
crontab:x:107:
syslog:x:108:
netdev:x:109:
messagebus:x:110:
uuidd:x:111:
ssl-cert:x:112:
lpadmin:x:113:anandms
lightdm:x:114:
nopasswdlogin:x:115:
ssh:x:116:
whoopsie:x:117:
mlocate:x:118:
avahi-autoipd:x:119:
avahi:x:120:
scanner:x:122:saned
colord:x:123:
pulse:x:124:
pulse-access:x:125:
rtkit:x:126:
saned:x:127:
sambashare:x:128:anandms
msanand:x:1001:
```

gshadow - Notepad

File Edit Format View Help

```
root:*::
daemon:*::
bin:*::
adm:*::syslog,anandms
tty:*::
disk:*::
lp:*::
mail:*::
news:*::
uucp:*::
man:*::
proxy:*::
kmem:*::
dialout:*::
fax:*::
voice:*::
cdrom:*::anandms
floppy:*::
tape:*::
sudo:*::anandms
audio:*::pulse
dip:*::anandms
www-data:*::
backup:*::
operator:*::
list:*::
irc:*::
src:*::
gnats:*::
shadow:*::
utmp:*::
video:*::
sasl:*::
plugdev:*::anandms
staff:*::
games:*::
users:*::
nogroup:*::
systemd-journal:!::
systemd-timesync:!::
systemd-network:!::
systemd-resolve:!::
systemd-bus-proxy:!::
input:!::
crontab:!::
syslog:!::
netdev:!::
messagebus:!::
uuidd:!::
ssl-cert:!::
lpadmin:!::anandms
lightdm:!::
nopasswdlogin:!::
ssh:!::
whoopsie:!::
mlocate:!::
avahi-autoipd:!::
avahi:!::
scanner:!::saned
colord:!::
pulse:!::
pulse-access:!::
rtkit:!::
saned:!::
sambashare:!::anandms
msanand:!::
```