



دانشگاه اصفهان

دانشکده کامپیوتر

گروه مهندسی نرم افزار

گزارش آزمایشگاه سیستم عامل

موضوع پروژه: اسکریپت مدیریت کاربر در شل لینوکس

نویسنده

سیدعلی میرامیرخانی

استاد درس

مهندس احمد بابایی مقدم

خردادماه ۱۳۹۷

۱- توضیح کلی اسکریپت

۱-۱- قسمت افزودن کاربر

در این قسمت امکان افزودن کاربر فراهم شده است. این کار به صورت manual بدون استفاده از دستورات آماده و در ۶ مرحله اصلی و چند فرآیند جانبی میسر شده است. ورودی های کاربر اعم از سوئیچ ها و کلمه های ورودی دقیقی بررسی شده و در صورت عدم وجود کاربر با این نام و نام خانودگی سعی در ایجاد کاربر می کند. نام کامل (نام + نام خانوادگی) کاربر در قسمت توضیحات ذخیره شده (GECOS) تا در مراجعات بعدی تکراری بودن کاربر به سرعت تشخیص داده شود. از آنجایی که پسورد برای کاربر تازه ایجاد شده برابر * یا عدم امکان ورود به آن حساب کاربری است از کاربر خواسته خواهد شد پسوردی برای user جدید ایجاد شده انتخاب نماید. در نهایت با نمایش پیغام موفقیت آمیز بودن عملیات و ثبت در logFile ایجاد خاتمه می یابد.

۱-۲- قسمت حذف کاربر

در این قسمت نیز مانند قسمت قبل ورودی های کاربر به دقت کنترل شده و در صورت وجود نام کاربری درخواست شده در دیتابیس عملیات حذف کاربر آغاز می شود. این کار نیز به صورت manual و در چهار مرحله اصلی انجام می پذیرد و در نهایت با نمایش پیغام موفقیت آمیز بودن عملیات و ثبت رکورد مذکور در logFile کار خاتمه می یابد.

۱-۳- قسمت بررسی کاربر موجود

در این قسمت روند مشابه قبل است و ورودی ابتدا بررسی می شود. پس از تأیید وجود کاربر در دیتابیس جزئیات کاربر خواسته شده نمایش داده شده و در نهایت رکورد این فعالیت در logFile ذخیره می شود.

۲- نحوه اجرا اسکریپت

تمام ورودی های مورد نیاز برای خلق/حذف/بررسی کاربر باید هنگام فراخوانی اسکریپت وارد شود. پس لازم است کاربر در هنگام کار با اسکریپت از ورودی ها و فرمت آنها اطلاع کامل داشته باشد. استثناء مورد فوق هنگام ایجاد کاربر است که در ادامه از کاربر درخواست ورود پسورد را خواهد کرد.

نکته مهم اینکه در هنگام فراخوانی اسکریپت لازم است دسترسی های روت برای کاربر اجرا کننده در دسترس باشد چرا که سطح دسترسی برای ایجاد کاربر و کار با فایل های مورد استفاده در اسکریپت دسترسی روت لازم دارد.

به طور کلی از دستورات ذیل برای اجرای اسکریپت می توان استفاده کرد:

می توان فایل اسکریپت را با استفاده از دستوراتی فراخوانی و اجرا کرد

```
[prompt]# bash usermng.sh
```

```
[prompt]# sh usermng.sh
```

هم چنین می توان اسکریپت را تبدیل به فایل اجرایی کرده و مستقیم آن را اجرا کرد

```
[prompt]# chmod +x usermng.sh
```

```
[prompt]# ./usermng.sh
```

دستورات فوق اسکریپت را در subshell اجرا می کنند و اگر بخواهیم در shell جاری آن را اجرا کنیم از دستورات زیر بهره می بریم:

```
[prompt]# source usermng.sh
```

```
[prompt]# . usermng.sh
```

۳- دستورات استفاده شده

کلیت این اسکریپت در ۱۲۰ خط با کامنت گذاری در مورد قسمت های مختلف اجرا تهیه شده است. در نوشتن آن از استفاده دستوراتی همچون userdel , useradd پرهیز شده و عملیات ایجاد و حذف کاربر به صورت manual است. به طور کلی از دستورات :

```
echo, awk, grep, cp, mkdir, rm, sed, cut, head, chown, passwd, shuf, id, find
```

استفاده شده است.

۴- توضیح کد اسکریپت

۱-۴- ایجاد کاربر

۱-۱-۴. بررسی ورودی

```
-->recognize create command
if [ "$1" = "--create" ] || [ "$1" = "-c" ] ; then
    if [ "$2" = "-f" ] ; then ##check for the first name
        if [ -z "$3" ] ; then ##check if First name is empty
```

پس از تایید این که دستور مربوط به ایجاد کاربر می باشد، بررسی می شود که سوئیچ نام صحیح بوده و وجود یک نام (null نبودن firstName) برای نام را بررسی می کند.

```
if [ -z "$5" ] ; then ##check if Last name is empty
    if [ "$4" = "-l" ] ; then ##check for the last name
```

همین عملیات برای نام خانودگی نیز بررسی می شود تا اطمینان از وارد شدن سوئیچ و رشته مناسب حاصل شود. واضح است که در صورت اشتباه وارد کردن هریک از فیلدهای ورودی پیام مربوط به او نمایش داده خواهد شد.

۲-۱-۴. بررسی موجود بودن در دیتابیس

```
fname=$3
lname=$5
grep "$3$5" /etc/passwd > /dev/null ##check if the user is duplicate
if [ $? -ne 0 ] ; then
```

پس از گذشتن موفقیت آمیز از قسمت های کنترلی ورودی نام و نام خانوادگی کاربر جدید برای پردازش های آینده ذخیره می شود و از آنجا که در هر ایجاد کاربر نام کامل کاربر جدید در فیلد GECOS ذخیره می شود ، در دیتابیس نام کامل کاربر برای جلوگیری از ایجاد دوباره کاربر جست و جو شده و پس از تائید آن روند ایجاد کاربر ادامه می یابد در غیراینصورت با نمایش پیام مربوط اجرای اسکریپت خاتمه می یابد.

۴-۱-۳. بررسی تکرار در نام کاربری

پس از تأیید در تکراری بودن نام کاربری نام نهایی ذخیره شده (texture) به صورت ترکیب حرف اول از نام ، نقطه و نام کامل خانوادگی خواهد بود پس بررسی می‌شود که اینگونه نامی از قبل نیز در دیتابیس وجود نداشته باشد.

کد این انعطاف را دارد که تا حرف دوم از نام را نیز در صورت تکراری بودن حرف اول نام بررسی کند تا در صورت تکراری نبودن ایجاد کند. برای جلوگیری از نمایش داده‌های نامربوط به کاربر اجرا کننده اسکریپت از انتقال خروجی به null در بعضی موارد استفاده شده است(> /dev/null)

```
compName=$(echo $fname | head -c 1)\.$lname ##create texture username
grep "^$compName" /etc/passwd > /dev/null
##check if texture user name is duplicate
if [ $? -eq 0 ] ; then
compName=$(echo $fname | cut -c 2)\.$lname
##ckeck if the new testure user name is duplicate
grep "^$compName" /etc/passwd > /dev/null
##abort the command becuaseof exture username failure
if [ $? -eq 0 ] ; then
```

۴-۱-۴. تابع ایجاد کاربر

```
#--> Function to create the user
createUser ()
```

ار آنجایی در قسمت‌های شرطی بیش از یک بار نوشتن روند اجرایی ایجاد کاربر مورد نیاز است گس بدین منظور تابعی تعریف می‌شود تا این کار را انجام دهد

```
#--> This loope will create an random ID and check if it is duplicate
while true ; do
    theRandomID=$(shuf -i 1000-9999 -n 1)    ##generate random ID
    awk -F ":" '{print $3}' /etc/passwd | grep "$theRandomID" > /dev/null
##check for duplicate
    if [ $? -ne 0 ] ; then
        break ##ID is valid
    fi
done
```

در ابتدای تابع با کمک یک حلقه و دستور shuf یک UID-GID برای کاربر جدید در رنج ۹۹۹۹-۱۰۰۰ (بر اساس رنج چهار رقمی اوپونتو) برای کاربر خلق شده و بررسی می‌شود که آیدی ایجاد شده از قبل در دیتابیس وجود نداشته باشد چراکه منحصر بفرد بودن این آیدی برای سیستم عامل اهمیت دارد.

```
#-->procedure to create an user
mkdir /home/"$1" ##make home directory for the user
```

```

echo "$1:x:$theRandomID:$theRandomID:$2$3:/home/$1:/bin/bash" >>
/etc/passwd ##put an entry in passwd file
echo "$1:x:$theRandomID:" >> /etc/group ##put an entry in group file
cp -r /etc/skel/* /home/"$1"/ ##fill the home directory with basic files
chown -R "$1":"$1" /home/"$1" ##give the user own access to his/her home
directory
passwd "$1" ##put an entry in shadow file and set password for the user
echo "UserID $1 with UID $theRandomID and GID $theRandomID created for $2
$3 @ $(date)"
echo "UserID $1 with UID $theRandomID and GID $theRandomID created for $2
$3 @ $(date)" >> /var/log/usermanagement.log

```

در نهایت طی چند مرحله اصلی شامل:

- ایجاد دایرکتوری home برای کاربر
- ثبت اطلاعات کاربر در passwd
- ایجاد گروه نظیر برای کاربر و ثبت اطلاعات مربوط در فایل group
- تکمیل home دایرکتوری کاربر با کپی کردن اطلاعات skel در آن
- انتصاب دایرکتوری ایجاد شده به کاربر جدید در مالکیت
- درخواست پسورد برای کاربر جدید و ثبت اطلاعات مربوط در فایل shadow

سپس پیام موفقیت آمیز بودن ایجاد کاربر نمایش داده شده و رکورد مربوط در usermanagement.log ذخیره می‌شود.

۴-۲- حذف کاربر

۴-۲-۱. بررسی ورودی

این روند بسیار مشابه روند بررسی صحت آرگومان‌های ورودی در ایجاد کاربر است

```

#--->recognize delete command
elif [ "$1" = "--delete" ] || [ "$1" = "-d" ] ; then
    if [ "$2" = "-u" ] ; then ##check for the correct switch
        if [ -z "$3" ] ; then ##check if the userName field is empty

```

و پس از تأیید انتخاب درست سوئیچ‌های delete و -u و مطمئن شدن از ورود یک رشته به عنوان نام کاربر کار ادامه روند حذف کاربر دنبال می‌شود.

۴-۲-۲. بررسی وجود کاربر

```

gsrep "^$3" /etc/passwd > /dev/null
##check if the entered user exist in the database
if [ $? -eq 0 ] ; then
theID=$(id -u $3)
##save some attribute for futhur useage
textID=$3

```

با استفاده از نام کاربری داده شده به اسکریپت ابتدا وجود آن کاربر در سیستم چک می‌شود و آیدی و texture id کاربر برای استفاده در آینده (ثبت در logFile) ذخیره می‌شود چرا که پس از حذف کاربر اطلاعات مورد نیاز از این کاربر دیگر در دسترس نیست.

۴-۲-۳. روند حذف کاربر

```
#-----IMPORTANT-----
#these two commands will take a lot of time to execute if you sure about it
you can uncommnet them
# find / -user $(id -u $3) -exec rm -rf {} \; 2>/dev/null
##delete whatever blongs to the user
# find / -group $(id -u $3) -exec rm -fr {} \; 2>/dev/null
##delte whatever blongs to the user group
```

دستور فوق برای حذف کامل تمام فایل‌ها و دایرکتوری‌های که مالکیت آن بر کاربر و یا گروه کاربر است ، که البته به دلیل زمان‌بر بودن و مختل کردن(به دلیل شروع عملیات از ریشه) روند اجرای کامنت شده است که در صورت تمایل قابل فعال سازی است. لازم به ذکر است در صورت فعال کردن این دو دستور دیگر اجرای دستور حذف دایرکتوری home کاربر معنایی ندارد و بهتر است کامنت شود.

```
rm -rf /home/"$3"/
##delete all the data of the user in his\her home directory
rm -rf /var/mail/"$3"/ ##delete user email box
sed -i "/^$3/d" /etc/passwd ##delete user entry from passwd file
sed -i "/^$3/d" /etc/shadow ##delete user entry from shadow file
sed -i "/^$3/d" /etc/group ##delte user entry from group file
echo "UserID $textID with UID $theID removed @ $(date)"
echo "UserID $textID with UID $theID removed @ $(date)" >>
/var/log/usermanagement.log
```

در نهایت حذف کاربر درخواست شده طی چندین مرحله اصلی شامل:

- حذف دایرکتوری home کاربر
- حذف ایمیل‌های کاربر
- پاک کردن entry کاربر از فایل passwd
- پاک کردن entry کاربر از فایل shadow
- پاک کردن گروه کاربر از فایل group

انجام شده و کار حذف با نمایش موفقیت‌آمیز بودن عملیات به کاربر و ثبت رکورد مربوط در log پایان می‌پذیرد.

۳-۴ - بررسی کاربر موجود

۱-۳-۴. بررسی ورودی

```
#-->recognize check command
elif [ "$1" = "--check" ] || [ "$1" = "-i" ] ; then
    if [ "$2" = "-u" ] ; then ##check if swtich is entered correctly
        if [ -z "$3" ] ; then ##check if the userName is empty
```

در ابتدا سوئیچها و خالی نبودن قسمت نام کاربر چک شده و در صورت تائید عملیات ادامه می‌یابد.

۲-۳-۴. بررسی وجود در دیتابیس

```
grep "^$3" /etc/passwd > /dev/null
##check if the user exist in the database
if [ $? -eq 0 ] ; then
```

سپس با دستور grep وجود چنین کاربری در فایل passwd بررسی می‌شود

۳-۳-۴. نمایش جزئیات

```
echo "the user information is: \n $(id $3)"
##return the required information of the user
```

با کمک گرفتن از دستور id اطلاعات جامعی از کاربر مدنظر شامل UID, GID, username نمایش داده خواهد شد.

۴-۳-۴. ثبت در logFile

```
echo "UserID $3 with UID $(id -u $3) has been checked @ $(date)" >>
/var/log/usermanagement.log
```

در پایان نیز اطلاعات کاربر در usermanagement.log ذخیره می‌شود.