



Laboration – Användarhantering

OBS det kan förekomma små skillnader vad gäller sökvägar samt befintliga kommandon mellan olika distributioner. T.ex. mellan CentOS och Ubuntu,

Material: För att genomföra laborationen behöver man ha tillgång till en dator (vm) med Linux installerat. Windows Subsystem for Linux går att använda också. För vissa övningar kan lokal administratörsbehörighet behövas. I labben utgår det från att man är inloggad med en användare som heter **sysadmin**

Mål: I denna laboration kommer du att utföra följande uppgifter:

- Skapa en ny grupp med kommandot **groupadd**.
- Gör ändringar i grupper med **groupmod**-kommandot
- Skapa en ny användare med kommandot **useradd**.
- Sätt och återställ en användares lösenord med **passwd**-kommandot
- Gör ändringar i användarkontot med **usermod**-kommandot

Skapa grupper

I denna uppgift skapar du grupp- och användarkonton.

Gruppkonton kan vara hjälpsamma att använda för att kunna tilldela behörigheter på filer som delas av en grupp användare.

Användarkonton i Linux-distributioner baserade på RedHat, som CentOS-distributionen, börjar med första användar-ID (UID) på 500, nästa UID på 501, och så vidare. Den nuvarande trenden som följs av många andra distributioner är att den första UID är 1000, den andra 1001, och så vidare. Från och med RedHat 7 börjar standardanvändarkonton på 1000, en faktor som bör övervägas vid migrering av äldre system med befintliga användarkonton.

Om man hanterar konton för flera system är det önskvärt att ha en nätverksbaserad autentiseringsserver, där konton kan skapas en gång men användas på många maskiner. Annars kan det vara utmanande att hantera flera konton på flera datorer eftersom det kan vara svårt att säkerställa att användaren och alla grupper de tillhör har samma UID och GID på alla datorer.

Ett annat problem med konton på flera maskiner är att det kan vara svårt att hålla lösenorden till varje konto synkroniserade över alla maskiner.

Att hantera konton för lokala användare är fortfarande användbart för enskilda maskiner, även om de har tillgång till en nätverksbaserad autentiseringsserver. I detta labb kommer du att hantera lokala grupp- och användarkonton.

För att administrera användar- och gruppkonton vill du byta användare till root-kontot med följande kommando. Ange root-lösenordet när du blir tillfrågad.

```
su -
```

```
sysadmin@localhost:~$ su -  
Password:  
root@localhost:~#
```



Använd kommandot `groupadd` för att skapa grupper som kallas `research` och `sales`:

```
groupadd -r research
```

```
groupadd -r sales
```

```
root@localhost:~# groupadd -r research
```

```
root@localhost:~# groupadd -r sales
```

`research` och `sales` grupperna som just lades till lades till i *det reserverade* intervallet (mellan 1-999) eftersom `-r`-alternativet användes. Med detta alternativ tilldelas gruppidentifierare (GID) automatiskt ett värde som är mindre än den lägsta normala användar-UID. `groupadd`-kommandot ändrar filen `/etc/group` där gruppkontoinformation lagras.

`groupmod`-kommandot kan användas med ett `-n`-alternativ för att ändra namnet på någon av dessa grupper eller med `-g`-alternativet för att ändra GID för någon av grupperna. `groupdel`-kommandot kan användas för att ta bort någon av grupperna, så länge ingen av dem har gjorts till primär grupp för en användare.

Använd kommandot `getent` för att hämta information om den nya gruppen `research`:

```
getent group research
```

```
root@localhost:~# getent group research
```

```
research:x:999:
```

Ditt resultat bör se likt ut som exemplet ovan, även om den GID som tilldelats kan vara annorlunda. Nu när gruppen `research` har skapats kan befintliga eller nya användare bli medlemmar i gruppen.

Använd `grep`-kommandot för att hämta information om den nya fgruppen `sales`:

```
grep sales /etc/group
```

```
root@localhost:~# grep sales /etc/group
```

```
sales:x:998:
```

Ditt resultat bör se likt ut som exemplet ovan, även om den GID som tilldelats kan vara annorlunda. Nu när gruppen `sales` har skapats kan befintliga eller nya användare bli medlemmar i denna grupp.

Använd `groupmod`-kommandot med `-n`-alternativet för att ändra namnet på gruppen `sales`.

```
groupmod -n clerks sales
```

```
root@localhost:~# groupmod -n clerks sales
```

Använd nu `groupmod`-kommandot med `-g`-alternativet för att ändra GID för gruppen.

```
groupmod -g 10003 clerks
```

```
root@localhost:~# groupmod -g 10003 clerks
```

Använd `grep`-kommandot för att verifiera ändringarna ovan.

```
grep clerks /etc/group
```



```
root@localhost:~# grep clerks /etc/group
clerks:x:10003:
```

Viktigt

Observera att alla filer som tidigare tillhört gruppen `sales` nu inte har något gruppnamn och nu blir *föräldralösa filer*.

Ta bort clerks-gruppen med `groupdel`-kommandot tillsammans med gruppens namn:

```
groupdel clerks
```

```
root@localhost:~# groupdel clerks
```

Använd `grep`-kommandot för att verifiera att kontorsgruppen har tagits bort:

```
root@localhost:~# grep clerks /etc/group
root@localhost:~#
```

Viktigt

Om du bestämmer dig för att radera en grupp med `groupdel`-kommandot, var medveten om att alla filer som ägs av den gruppen också blir föräldralösa.

Användarkonfiguration

Användarkonfiguration börjar med att korrekt konfigurera de grupper som användarna ska placeras i.

Visa standardvärdena som används av `useradd`-kommandot med alternativet `-D`:

```
useradd -D
```

```
root@localhost:~# useradd -D
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
CREATE_MAIL_SPOOL=no
```

`SKEL`-värdet ger administratörer ett enkelt sätt att fylla ett nytt användarkonto med nyckelkonfigurationsfiler. Den avgör vilken skelettkatalog som ska få sitt innehåll kopierat till den nya användarens hemkatalog. `-k`-alternativet i `useradd`-kommandot tillåter att en annan `SKEL`-katalog än standarden används vid skapande av ett nytt användarkonto. Detta är användbart eftersom de flesta system har användare som behöver tillgång till olika resurser beroende på deras arbetsuppgifter.

Ställ in parametern `INACTIVE` så att användare med utgångna lösenord kan logga in i upp till trettio dagar innan deras konton inaktiveras, och se sedan de nya standardvärdena. Alternativet `-D` i kommandot `useradd` låter dig se eller ändra några av standardvärdena som används av `useradd`-kommandot.

I exemplet nedan specificerar `-D`-alternativet ändringar i standardvärdena som används vid skapandet av en ny användare. Alternativet `-f 30` anger att användare med utgångna lösenord fortfarande kan logga in i upp till trettio dagar innan deras konton inaktiveras. Att använda `-D`-



alternativet visar ensam de aktuella standardinställningarna, som har ändrats av föregående kommando.

```
useradd -D -f 30
```

```
useradd -D
```

```
root@localhost:~# useradd -D -f 30
root@localhost:~# useradd -D
GROUP=100
HOME=/home
INACTIVE=30
EXPIRE=
SHELL=/bin/sh
SKEL=/etc/skel
CREATE_MAIL_SPOOL=no
```

Ändra värdet `CREATE_MAIL_SPOOL` i filen `/etc/default/useradd` med `nano` texteditorn:

```
nano /etc/default/useradd
```

```
root@localhost:~# nano /etc/default/useradd
```

```
GNU nano 2.9.3 /etc/default/useradd

# Default values for useradd(8)
#
# The SHELL variable specifies the default login shell on your
# system.
# Similar to DSHELL in adduser. However, we use "sh" here because
# useradd is a low level utility and should be as general
# as possible
SHELL=/bin/sh
#
# The default group for users
# 100=users on Debian systems
# Same as USERS_GID in adduser
# This argument is used when the -n flag is specified.
# The default behavior (when -n and -g are not specified) is to create a
# primary user group with the same name as the user being added to the
```

Tryck på **nedpilen** ↓ för att scrolla till botten av filen:



```
GNU nano 2.9.3 /etc/default/useradd

# copied to the new user's home directory when it is created.
# SKEL=/etc/skel
#
# Defines whether the mail spool should be created while
# creating the account
CREATE_MAIL_SPOOL=no
GROUP=100
HOME=/home
INACTIVE=30
EXPIRE=
SKEL=/etc/skel
█
```

På raden `CREATE_MAIL_SPOOL=no`, ändra till `yes`:

```
GNU nano 2.9.3 /etc/default/useradd

# copied to the new user's home directory when it is created.
# SKEL=/etc/skel
#
# Defines whether the mail spool should be created while
# creating the account
CREATE_MAIL_SPOOL=yes█
GROUP=100
HOME=/home
INACTIVE=30
EXPIRE=
SKEL=/etc/skel
```

Tryck på **Ctrl + X** för att avsluta och skriv **Y**. Tryck **Enter** för att spara dina ändringar och skriv sedan `useradd -D` i prompten för att bekräfta den nya inställningen:

```
useradd -D

root@localhost:~# useradd -D
GROUP=100
HOME=/home
INACTIVE=30
EXPIRE=
SHELL=/bin/sh
SKEL=/etc/skel
CREATE_MAIL_SPOOL=yes
```

Kommandot `useradd` skapar nu en e-postspoolfil.

Skapa en ny användare med namnet `student` som är sekundär medlem i gruppen `research` och primär medlem i sin egen privata grupp. Använd en kommentar `Linux Student` som visas som



användarens fullständiga namn när de gör en grafisk inloggning. Se till att deras hemkatalog skapas genom att ange `-m`-alternativet. Använd sedan `grep` för att verifiera den nya användaren och deras gruppmedlemskap:

```
useradd -G research -c 'Linux Student' -m student
grep student /etc/passwd
grep student /etc/group
```

```
root@localhost:~# useradd -G research -c 'Linux Student' -m student
root@localhost:~# grep student /etc/passwd
student:x:1002:1002:Linux Student:/home/student:/bin/sh
root@localhost:~# grep student /etc/group
research:x:999:student
student:x:1002:
```

Användarens kontoinformation lagras i filerna `/etc/passwd` och `/etc/shadow`. Användarens gruppinformation finns i filerna `/etc/passwd` och `/etc/group`.

Använd `usermod`-kommandot för att lägga till gruppen `research` som en sekundär grupp för `sysadmin`-användaren:

```
usermod -aG research sysadmin
```

```
root@localhost:~# usermod -aG research sysadmin
```

Användare som är aktivt inloggade i systemet kommer inte att kunna använda några nya gruppmedlemskap förrän nästa gång de loggar in i systemet.

Med kommandot `getent` kan du titta på `research`-gruppens medlemmar igen:

```
getent group research
```

```
root@localhost:~# getent group research
research:x:999:student,sysadmin
```

Använd `getent` för att visa gruppen `student`:

```
getent group student
```

```
root@localhost:~# getent group student
student:x:1002:
```

Använd sedan `getent` för att visa `passwd`- och `shadow`-databaserna för användaren `student`:

```
getent passwd student
```

```
getent shadow student
```

```
root@localhost:~# getent passwd student
student:x:1002:1002:Linux Student:/home/student:/bin/sh
root@localhost:~# getent shadow student
student:!:16902:0:99999:7:30::
```



Resultatet bör nu visa att både `sysadmin` och `student` är sekundära medlemmar i gruppen `research`.

Gruppen `student` GID matchar det fjärde fältet i `passwd`-informationen för användaren `student`. Detta är vad som gör eleven till en primär medlem i gruppen `student`.

Slutligen, `!` som visas i det andra lösenordsfältet i `shadow`-filen visar att lösenordet för användaren `student` inte har satts.

Använd `passwd`-kommandot för att ställa in lösenord, för användaren `student`. Ange lösenordet två gånger och titta sedan på `shadow`-filsposten för användaren `student` igen:

```
passwd student
```

```
root@localhost:~# passwd student
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

Notera

Inga tecken kommer att visas när man skriver lösenordet.

Utdata från `/etc/shadow`-filen visar nu ett krypterat lösenord i det andra fältet:

```
getent shadow student
```

```
root@localhost:~# getent shadow student
student:$6$pIEEdvAX$GBo0beYhojL3/vDrOP2UAQR6uVCWMZXxMPqImREJWw/5oR
```

Bara för att en användare har ett lösenord betyder det inte att de någonsin har loggat in i systemet. Använd `last` kommandot för att se om användaren `student` någonsin har loggat in:

```
last
```

```
last student
```

```
root@localhost:~# last
sysadmin pts/0          Sat Feb 24 16:30      still logged in

wtmp begins Sat Feb 24 16:30:01 2024
root@localhost:~# last student

wtmp begins Sat Feb 24 16:30:01 2024
```

Resultatet av det `last` kommandot ska visa att `sysadmin`-användaren har loggat in tidigare, men inte användaren `student`. Det finns också ett `lastb`-kommando, som fungerar liknande det förra kommandot förutom att det visar "dåliga" eller misslyckade inloggningsförsök.

Om du inte längre vill att användaren `student` skulle ha tillgång till systemet, kan `usermod -L student` kommandot användas för att "låsa" kontot. Kontot kunde låsas upp med `usermod -U student` kommandot.



En mer permanent lösning för att förhindra åtkomst till kontot `student` vore att radera kontot med antingen `userdel student` eller `userdel -r student`-kommandot. Genom att använda `-r`-alternativet med kommandot `userdel` tas användarens hemkatalog och e-post bort, utöver att användarens konto tas bort.

Ta bort kontot `student` och ta bort användarens hemkatalog:

```
userdel -r student
```

```
root@localhost:~# userdel -r student
```

Använd `grep`-kommandot för att verifiera att studentanvändaren har tagits bort.

```
root@localhost:~# grep student /etc/group
root@localhost:~#
```