

## DVA249/DVA267 Linux, HT2023

### - Laboration 5 -

## Förberedelser

Läs instruktionerna på Canvas och läs veckans kurslitteratur innan du börjar med laborationen. Vi rekommenderar även att du tittar på videomaterialet som tillhör laborationen. Svaren till flera av uppgifterna finns i kursmaterialet.

## 1 Filrättigheter, användare, grupper och länkar

I Linux styrs åtkomsten till filer, kataloger och kringutrustning med användare och grupper. Vem som äger filerna är avgörande för säkerheten. Vanligtvis ägs filen av den användaren som skapade filen. För att ändra filägare krävs administrativa rättigheter.

Kommandot **sudo** kör ett kommando med **root**-rättigheter. När du använder **sudo** innan ett kommando frågar systemet om ditt lösenord innan den kör kommandot som användaren *root*.

Användaren måste vara medlem i gruppen *sudo* för att använda **sudo**. En fördel med att använda kommandot **sudo** istället för att byta användare till *root* är att endast ett kommando exekveras och återgår sedan till din användare. Detta förhindrar bland annat oavsiktlig borttagning och ändring av viktiga systemfiler.

### 1.1 Hantera länkar

En länk är ett sätt att göra en fil mer tillgänglig, exempelvis genom att ge den fler identiteter eller ge den flera olika namn. Det finns två typer av länkar, symboliska länkar (*'soft link/symbolic link'*) och hårda länkar (*'hard link'*).

1. Börja med att skapa två kataloger i din hemkatalog, **lab5test** och **subdir1**. Skapa sedan en ny fil med namnet **file1.bash** i katalogen **subdir1**.

Skapa sedan en symbolisk länk i katalogen **lab5test** till filen **file1.bash** i **subdir1**-katalogen. Använd filnamnet (**file1.bash**) som länknamn. Använd **ls** för att säkerställa att länken skapades.

2. Vad händer om du tar bort **file1.bash** från **subdir1**? Kan filen öppnas via den symboliska länken i katalogen **lab5test**? Förklara varför eller varför inte.
3. Skapa en ny fil med namnet **file2.txt** i katalogen **subdir1**. Skapa nu en hård länk i katalogen **lab5test** till filen **file2.txt** i **subdir1**. Använd filnamnet som länknamn.
4. Vad händer om du tar bort **file2.txt** från **subdir1**? Kan filen öppnas via den hårda länken i katalogen **lab5test**? Förklara varför eller varför inte.
5. Det går även att skapa symboliska länkar till kataloger. Skapa en symbolisk länk till **/var/log** med namnet **logs** i din hemkatalog. Vilket kommando använde du?
6. Testa nu att skapa en hård länk till en katalog. Vad händer?

## 1.2 Användare och grupper

1. Testa kommandot `id`. Läs manualen för `id`.
  - (a) Med vilken flagga för `id`-kommandot kan du få fram ditt UID-nummer (*'användar-id/user identification'*)
  - (b) Med vilken flagga för `id`-kommandot kan du få fram en lista med namnet på alla grupper som du är medlem i?
2. Titta nu på filen `group` i ditt filsystem, använd `less /etc/group`
  - (a) Kan du hitta några likheter med `id`-kommandot?
  - (b) Vilket kommando (med pipe) kan du använda för att endast visa rader som inkluderar ditt användarnamn?
3. Titta nu på filen `passwd` i ditt filsystem, använd `less /etc/passwd`
  - (a) Vad innehåller filen?
  - (b) Titta på raden som innehåller ditt användarnamn. Beskriv vad de olika fälten (kolon-separerade) innehåller.
4. Skapa ett användarkonto för *Alice Stone* genom att använda kommandot `useradd` med `sudo`.
  - Användarnamn (LOGIN): `alisto`
  - Kommentar (COMMENT): `Alice Stone`
  - Hemkatalog (HOMEDIR): `/home/alisto`
  - Skal (SHELL): `/bin/bash`
  - Sätt lösenord på kontot.

Vilket/Vilka kommandon använde du?

**Tips:** Kommandot `su --login användarnam` byter användare till den användare som anges. Läs manualsidan för `su` för mer information.

5. Skapa ett användarkonto för *Bob Anderson* genom att använda kommandot `adduser` (*inte useradd*) med `sudo`.
  - Användarnamn (LOGIN): `boband`
  - Kommentar (COMMENT): `Bob Anderson`
  - Hemkatalog (HOMEDIR): `/home/boband`
  - Skal (SHELL): `/bin/bash`
  - Sätt lösenord på kontot.

Vilket/Vilka kommandon använde du?

6. Skapa en ny grupp med namnet *solarproj* och lägg till Alice och Bob som medlemmar i gruppen. Vilka kommandon använde du?

Användarna *Alice* och *Bob*, och gruppen *solarproj* kommer användas i nästa del av laborationen.

## 1.3 Ägare och rättigheter

1. Skapa ett skript med namnet `1.3-listing.bash` enligt Kodruta 1.

```
#!/bin/bash
echo
echo Showing content of directory $(pwd)
echo -----
ls
```

Kodruta 1: `1.3-listing.bash`

Ändra rättigheterna på filen `1.3-listing.bash`

- (a) till endast läs-, skriv- och exekveringsrättigheter för användare (`-rwx-----`). Vilket kommando använde du?

**Tips:** Genom att sätta exekveringsrättigheter på filen som innehåller ett bash-skript behöver vi inte längre ange skriptet som argument till `bash`. Kör skriptet genom att skriva `./1.3-listing.bash`

- (b) till läs- och skrivrättigheter för användaren, endast läsrättigheter för gruppen och inga rättigheter för andra (`-rw-r----`). Vilket kommando använde du?
2. Ändra rättigheterna på filen `1.3-listing.bash` till endast läsrättigheter för användaren och ta bort rättigheter för grupp och andra (`-r-----`).
    - (a) Går det att skriva ut filen `1.3-listing.bash` på skärmen med `cat`? Varför/varför inte?
    - (b) Går det att editera filen `1.3-listing.bash` och spara ändringarna? Varför/varför inte?
  3. Beskriv hur kommandot `umask` kan användas för att ge nya filer läs- och skrivrättigheter för användaren, endast läsrättigheter för gruppen och inga rättigheter för andra (`-rw-r---`).

**Tips:** Kommandot `umask` påverkar endast filer och kataloger som skapas, inte filer och kataloger som redan existerar. Värdet subtraheras från *mode value* (*default permissions*) som specificeras av program, exempelvis `touch`. Standard *mode* för filer är `666` och `777` för kataloger.

4. Skapa en katalog med namnet `projects` i root-katalogen. Skapa sedan katalogen `solarproj` i `projects`-katalogen. Resultatet ska se precis ut som i Figur 1. Vilka kommandon använde du?

```
awk03@xubuntu:/projects$ ls -al
total 12
drwxr-xr-x  3 root root    4096 Nov  28 15:29 .
drwxr-xr-x 21 root root    4096 Nov  28 15:26 ..
drwxrwsr-t  2 root solarproj 4096 Nov  28 16:09 solarproj
```

Figur 1: Listar innehållet i `projects`-katalogen.

## 2 Nätverk och fjärråtkomst

De flesta Linux-system är anslutna till ett nätverk, antingen som en klient eller en server. Det är därför viktigt att ha grundläggande kunskap om nätverk och nätverksinställningar i Linux. Det är även viktigt att veta hur man fjärransluter till ett Linux-system, eftersom Linux ofta används som operativsystem på servrar.

### 2.1 Nätverkskonfiguration

I Linux kan nätverksinställningar göras med verktyg i GUI eller genom konfigurationsfiler i CLI. Verktygen som finns tillgängliga för dig varierar beroende på Linuxdistribution. Ubuntu/Xubuntu använder verktygen Netplan, NetworkManager och Networkd. Netplan konfigureras och skickar sedan vidare konfigurationen till antingen verktyget NetworkManager (nätverksinställningar i GUI) eller verktyget systemd-networkd (nätverksinställningar i CLI). Konfigurationen för Netplan görs via en YAML-fil i katalogen `/etc/netplan/`. Läs mer om Netplan och se konfigurationsexempel på <https://netplan.io/>.

#### 2.1.1 Nätverkskort

I terminalen kan du använda följande kommandon för att lista alla installerade nätverkskort. Undersök kommandona och beskriv dessa kortfattat.

1. `lspci | grep -i -E 'network|ethernet'`
2. `sudo lshw -class network`

#### 2.1.2 IP-kommandot

Kommandot `ifconfig` (Net-tools-paketet) var tidigare standard för att lista, stänga av och sätta på nätverksinterface. Kommandot `ifconfig` har ersatts av kommandot `ip` och finns inte längre installerat på nyare versioner av Ubuntu/Xubuntu.

Använd kommandot `ip` och besvara följande frågor:

1. Vilket kommando används för att lista (*'show'*) information om alla nätverksinterface?
2. Vilket kommando används för att stänga av (*'interface down'*) ett nätverksinterface?
3. Vilket kommando används för att sätta på (*'interface up'*) ett nätverksinterface?

#### 2.1.3 NetworkManager

För Ubuntu/Xubuntu med grafiskt gränssnitt är det NetworkManager som hanterar nätverksinterface och inställningar. Netplan läser in konfigurationen i filen `/etc/netplan/01-network-manager-all.yaml` som talar om att *renderer* ska vara NetworkManager, se Kodruta 2.

```
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
```

Kodruta 2: `/etc/netplan/01-network-manager-all.yaml`

Med endast *version* och *renderer* konfigurerad i YAML-filen, överläter Netplan nätverkskonfigurationen till NetworkManager. Standardinställningarna i NetworkManager är att maskinen får en IP-adress tilldelad via nätverket med hjälp av DHCP.


### 2.1.4 Networkd

Flytta YAML-filen `/etc/netplan/01-netork-manager-all.yaml` till din hemkatalog. Då har du en backup på konfigurationsfilen ifall något skulle gå fel. Skapa en ny YAML-fil med namnet `01-networkd.yaml` i katalogen `/etc/netplan/`. Byt *renderer* till `systemd-networkd` genom att skriva konfiguration i filen enligt Kodruta 3. Maskinen får fortfarande en IP-adress tilldelad via nätverket med hjälp av DHCP.

```
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: true
```

Kodruta 3: `/etc/netplan/01-networkd.yaml`

Applicera inställningarna genom att köra kommandot `sudo netplan apply`.

**OBS!** Använd inte  (TAB-tangenten) i YAML-filen. Använd endast mellanslag. Du måste skriva in namnet på ditt nätverksinterface till exempel `enp0s3` eller `ens160`. Ta reda på namnet med hjälp av `ip`-kommandot.

Titta nu på nätverksinställningarna i GUI. Ser du någon skillnad efter byte av *renderer*?

**Tips:** Om ingenting händer kan du behöva stänga av och sätta på nätverksinterfacet.

### 2.1.5 Statisk IP-adress

Serverar har oftast manuellt konfigurerade IP-adresser, så kallade statiska adresser. Detta för att servern alltid ska ha samma IP-adress. Konfigurationen görs i YAML-filen.

Ändra nu konfigurationen i YAML-filen till konfigurationen i Kodruta 4. Kör sedan kommandot `sudo netplan apply`.

```
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      addresses:
        - 10.0.2.16/24
      nameservers:
        addresses: [1.1.1.1, 8.8.8.8]
      routes:
        - to: default
          via: 10.0.2.2
```

Kodruta 4: `/etc/netplan/01-networkd.yaml`

Säkerställ att du fick den önskade IP-adressen genom kommandot `ip address`.

**OBS!** Interface-namn och IP-adresser kan skilja sig beroende på system och nätverk. Konfigurationen i Kodruta 4 fungerar på Xubuntu i VirtualBox.

## 2.2 Fjärråtkomst

Secure Shell (SSH) är ett verktyg för att ansluta och logga in på andra datorer. SSH kan användas mellan olika operativsystem, exempelvis från Windows, Mac eller Linux till en Linux-server i molnet. Med SSH går det även att kopiera filer mellan system med kommandot `scp`. Kommandot `scp` fungerar ungefär som kommandot `cp` men från en dator till en annan dator.

1. Börja med att installera openssh-server på din maskin.
2. Hur kan du ansluta till en annan dator med `ssh`?

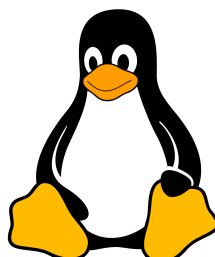
**Tips:** Använd loopback-adressen (*127.0.0.1* eller *localhost*) om du inte har tillgång till en annan Linux-dator.

3. Hur kan du använda `ssh` för att köra ett kommando på en annan dator utan att först logga in?
4. Hur kan du kopiera en fil från en annan dator till din dator med hjälp av kommandot `scp`?

**Tips:** Om du senare kommer arbeta mycket med nätverk, Linux-instanser i molnet och fjärranslutningar kan följande vara användbart.

Det går att skapa RSA-autentiseringsnycklar som kan användas för lösenordslös inloggning med SSH. Detta ingår inte i kursen, men du kan hitta information online eller i manualsidorna för `ssh-keygen` och `ssh-copy-id`.

**Tips:** SSH File Transfer Protocol (SFTP) är ett nätverksprotokoll för filhantering över SSH. Jämfört med `scp` har SFTP utökad funktionalitet.



## DVA249/DVA267 Linux, HT2023

### - Quiz 5 -

Denna uppgift genomförs **individuellt och lämnas in genom att göra quiz "Quiz 5" i Canvas** när du är klar med uppgiften. Uppgiften ska vara inlämnad innan deadline, annars kan vi inte garantera att vi hinner titta igenom era inlämningar innan examinationstillfället.

### 3 Filrättigheter

Bob (*boband*) och Alice (*alisto*) är medlemmar i gruppen *solarproj* och kommer använda katalogen *solarproj* för att tillsammans arbeta med ett mjukvaruprojekt. Figur 2 visar projektets filer (innehållet i katalogen *solarproj*).

```
awk03@xubuntu:/projects/solarproj$ ls -al
totalt 32
drwxrwsr-t 3 root  solarproj 4096 Nov 28 16:02 .
drwxr-xr-x 3 root  root      4096 Nov 28 15:29 ..
-rw-rw---- 1 boband boband   802 Nov 28 15:44 bearings.jpg
-rw-rw---- 1 boband solarproj 101 nov 28 15:43 controller.c
drwxrwsr-x 2 alisto solarproj 4096 Nov 28 15:40 old
-----rwx 1 alisto solarproj  19 Nov 28 14:08 power
-rw-r--r-- 1 alisto solarproj  81 Nov 28 15:44 schedule.txt
-rw----r-- 1 boband solarproj 576 Nov 28 15:58 temp
lrwxrwxrwx 1 alisto solarproj  24 Nov 28 16:02 temp.sym -> ../solarproj/temp
```

Figur 2: Listar innehållet i *solarproj*-katalogen.

Studera filrättigheterna i katalogen *solarproj* och svara på följande frågor.

**Tips:** Skapa filerna, länkarna och katalogerna på din Linux-maskin. Sätt rättigheter enligt Figur 2. Använd kommandot `su --login användarnam` för att logga in som **alisto** eller **boband**.

1. Kan *Bob* ändra innehållet i filen *schedule.txt*? Varför/varför inte?
  - ☐ Ja, gruppen *solarproj* har skrivrättigheter.
  - ☐ Ja, gruppen *solarproj* har läsrättigheter.
  - ☐ Nej, gruppen *solarproj* har inga skrivrättigheter.
  - ☐ Nej, exekveringsrättigheter saknas för alla.
2. Kan *Alice* ta bort filen *controller.c*? Varför/varför inte?
  - ☐ Ja, setgid är satt på katalogen *solarproj*.
  - ☐ Ja, gruppen *solarproj* har skrivrättigheter på filen.
  - ☐ Nej, gruppen *solarproj* har inte skriv- och exekveringsrättigheter på katalogen *solarproj*.
  - ☐ Nej, Sticky bit är satt på katalogen *solarproj*.

3. Vem kan läsa filen `power`? Varför?

- ☐ Bara medlemmar i gruppen *solarproj* kan läsa filen `power`.
- ☐ Alla andra förutom ägare och medlemmar i gruppen *solarproj*.
- ☐ Ingen kan läsa filen `power`.
- ☐ *Alice* kan läsa filen eftersom hon äger den.

4. Varför kan *Alice* men inte *Bob* byta arbetskatalog till `old`?

- ☐ *Alice* har exekveringsrättigheter på katalogen `old` men gruppen *solarproj* har inte exekveringsrättigheter.
- ☐ Setgid är satt på katalogen `old`.
- ☐ Katalogen `old` tillhör gruppen *alisto*.
- ☐ Gruppen *boband* har inte exekveringsrättigheter.

5. Kan *Alice* läsa filen `temp.sym`? Varför/varför inte?

- ☐ Ja, *Alice* har läsrättigheter på `temp.sym` och på filen som länken pekar på.
- ☐ Ja, *Alice* har läsrättigheter på `temp.sym` och kan då även läsa den länkade filen.
- ☐ Nej, *Alice* kan läsa länken men har inte läsrättigheter på filen `temp`.
- ☐ Nej, *Alice* har inte läsrättigheter på `temp.sym`.

## 4 SSH

*Alice* har installerat openssh-server på en dator för att låta *Bob* fjärransluta till datorn (Xubuntu/Ubuntu).

1. Vilket kommando ska hon använda för att se status på tjänsten ('*service*') `ssh`?
2. Hur kan tjänsten ('*service*') `ssh` stoppas?
3. Hur kan tjänsten ('*service*') `ssh` startas?

Quizet anses vara klar när du har 10 poäng på uppgiften i Canvas. Deadline för uppgiften är 8/12 kl 23:59.



## DVA249/DVA267 Linux, HT2023

### - Inlämningsuppgift 5 -

Denna uppgift genomförs **i grupp och lämnas in genom att ladda upp filen i Canvas** när du är klar med uppgiften. Uppgiften ska vara inlämnad innan deadline, annars kan vi inte garantera att vi hinner titta igenom era inlämningar innan examinationstillfället.

Skapa ett skalskript med namnet `users.bash`. Skriptet ska ge information om användare beroende på vilken flagga användaren anger, se Tabell 1.

**Tabell 1:** Option och funktion för skriptet `users.bash`

Option	Funktion
<code>-u, --users</code>	Listar alla användarkonton
<code>-o, --online</code>	Skriver ut alla användarkonton som är inloggade just nu
<code>--help</code>	Skriv ut information om skriptet och tillgängliga flaggor

Output till terminalen ska likna exemplet i Figur 3. I skriptet måste du förändra output från olika kommandon för att få en stilren och snygg output till terminalen med enbart den efterfrågade informationen. Skriptet behöver inte hantera flaggor som skrivs ihop, exempelvis `user.bash -uo`.

**Tips:** Titta på kommandot `who` och filen `/etc/passwd`. För att skriva en snyggare lista på användare kan kommandot `column` användas. Andra användbara kommandon är `grep`, `awk` och `cut`.

**OBS!** Output i Figur 3 är endast ett exempel. Ert skript kommer förmodligen lista fler användare. Output från kommandot `who` varierar beroende på hur många som är inloggade och om användarna är inloggade lokalt på datorn eller fjärranslutna via SSH. IP-adressen visar från vilken adress användaren har anslutit sig. Vid inloggning lokalt visas inte IP-adressen.

```
awk03@xubuntu:~$ ./users.bash --help
Usage: users.bash OPTION...
Print users

OPTIONS
-u, --users      list all user accounts
-o, --online     List all online users
    --help      display this help and exit
awk03@xubuntu:~$ ./users.bash -u
root          systemd-timesync      sshd
daemon        proxy                anka
bin           www-data            syslog
sync          tss                fwupd-refresh
games         irc                uidd
man           gnats              tcpdump
dnsmasq       nobody            kalle
mail          systemd-network    awk03
awk03@xubuntu:~$ ./users.bash -o
awk03 pts/0      2023-12-08 10:53 (10.0.0.10)
kalle pts/1      2023-12-08 11:13 (192.168.105.8)
anka pts/2      2023-12-08 13:01 (192.168.105.65)
```

**Figur 3:** Exempel på output från `users.bash`