

Rapport

**Introduktion till Linux och små nätverk, 7,5 hp
VT 2025**

Projektuppgift

Laboration 3: Administration av program

av

Sixten Peterson (050402-XXXX)

**Akademin för teknik och miljö
Avdelningen för industriell utveckling, IT och samhällsbyggnad**

Högskolan i Gävle
S-801 76 Gävle, Sweden

Datorpost:

education@snicon.rip

Innehållsförteckning

1. Inledning.....	1
1.1. Bakgrund.....	1
1.2. Syfte.....	1
2. Planering och genomförande.....	1
2.1. Planering.....	1
2.2. Genomförande.....	1
2.2.1. Del ett – Programmet ntp.....	2
2.2.2. Del 2 – Ny användare.....	4
2.2.3. Del tre - Rättigheter.....	5
3. Beskrivning av slutresultat.....	8
4. Diskussion.....	8
5. Slutsatser.....	8
6. Referenser.....	9
I. Bilaga 1: Skript.....	10

1. Inledning

Laborationen är det tredje momentet i kursen med tre delar som omfattar grundläggande systemadministration i form av paketinstallation och konfiguration, användar- och grupphantering samt rättigheter. Syftet är att öva på enkel systemadministration samt bilda en uppfattning om processen av installation och konfiguration av ett program/paket, i detta fall *ntp/ntpd*.

1.1. Bakgrund

Denna laboration utgör det tredje momentet i kursen ”Introduktion till Linux och små nätverk” (DVG001) och består av tre delar. Den första delen behandlar processen av att installera och konfigurera ett program i form av ett tidsynkroniseringsprogram vid namn *ntp* genom *sudo*. Del två behandlar processen av att skapa en ny användare samt att tilldela användaren gruppen ”*sudo*”. Avslutningsvis består del tre av att skapa filer och kataloger samt att justera rättigheterna för att därpå praktiskt testa vad de olika rättigheterna resulterar i vid exekvering av de kommandon som specificerats i uppgiftsbeskrivningen.

Under utförandet används den Linuxmiljö som installerats under den första laborationen i kursen.

1.2. Syfte

Laborationen syftar till att öva på systemadministration i form av rättighetshantering för såväl användare som grupper, filer och kataloger. Yttermera syftar laborationen till att bilda en uppfattning om hur en installation och konfiguration av program kan gå till i Debian.

2. Planering och genomförande

Genom kontinuerlig dokumentation samt en SSH-anslutning till Linuxmiljön genomförs de tre stegen av laborationen ett efter ett med olika figurer som komplement.

2.1. Planering

Arbetet dokumenteras kontinuerligt i realtid i samband med laborationens utförande för att säkerställa så akkurat information i rapporten som möjligt. Vid eventuella problem eller funderingar nyttjas i första hand kursboken och i andra hand en sökmotor såsom DuckDuckGo eller Google. Där officiell dokumentation för distributionen finns tillgänglig på internet prioriteras denna högst.

Tillvägagångssättet bygger i övrigt på en SSH-anslutning från min Macbook Pro med macOS Sequoia 15.3.1 till laborationsmiljön som kör Debian 12 Bookworm på en gammal Dell Inspiron 570.

2.2. Genomförande

Genomförandet är uppdelat i tre olika rubriker som speglar de olika delarna av laborationen som utförs. Nedan följer även en redogörelse av det som utförst innan samtliga delar.

Först upprättades en anslutning till Linuxmiljön genom *ssh* *hig-25sipe01@192.168.1.250* därefter skrevs lösenordet för kontot in. Väl ansluten användes *sudo apt update* för att updatera listan med tillgängliga paket, detta då det rekommenderas utföras innan användning av APT enligt kursboken [1, pp 116]. Då systemet inte heller har uppdaterats sedan installationen som utfördes i första laborationen kördes även *sudo apt upgrade* vilket installerar de senaste säkerhetsuppdateringarna [1, pp 120].

2.2.1. Del ett – Programmet ntp

Då sudo redan var installerat så skippades detta steg, jag fattar det som att sudo redan är installerat eftersom jag inte valde att sätta ett lösenord på root under installationen. Däremot kunde jag inte använda `aptitude(8)` då kommandot inte kunde hittas, se Figur 1. Således användes `sudo apt search aptitude` vilket bland annat visade paketet *aptitude* som beskrevs som en terminal-baserad pakethanterare. Övertygad om att *aptitude* var rätt paket att installera utfördes en installation genom `sudo apt install aptitude`.

```
[hig-25sipe01@dvg001:~$ sudo aptitude
sudo: aptitude: command not found
```

Figur 1: Kommandot *aptitude* kunde inte hittas

Efter *aptitude* var installerat användes `sudo aptitude search ntp` för att hitta paketet som skulle installeras. Resultaten listade bland annat *ntp* och *ntpsec*, det stod även att *ntp* var ett "transitional package". Min misstanke är att *ntpsec* ersätter *ntp* utifrån beskrivningarna men eftersom uppgiften specifikt nämner paketet *ntp* så valde jag att installera det med hjälp av `sudo aptitude install ntp`. Strax därefter uppstod ett problem, se Figur 2, efter att tagit del av informationen valde jag att acceptera lösningen genom att skriva in "y" och klicka på enter.

```
The following packages have unmet dependencies:
 ntp : Depends: ntpsec but it is not going to be installed
The following actions will resolve these dependencies:

  Remove the following packages:
1)      systemd-timesyncd [252.33-1~deb12u1 (now, stable)]

  Install the following packages:
2)      ntpsec [1.2.2+dfsg1-1+deb12u1 (stable, stable-security)]
3)      python3-ntp [1.2.2+dfsg1-1+deb12u1 (stable, stable-security)]

Accept this solution? [Y/n/q/?]
```

Figur 2: Lösningsförslag för att installera paketen som *ntp* är beroende av.

Därefter lyckades installationen och *zmore* användes för att läsa på lite om paketet. Genom att skriva `zmore /usr/share/doc/ntp` och klicka på tab så visades två förslag, se Figur 3, då jag inte kunde hitta någon *README.Debian* så valde jag att läsa innehållet i *NEWS.Debian.gz*. Väl där står det klart och tydligt att *ntpsec* ersätter *ntp* i Debian, vilket bekräftade min misstanke. Eftersom *ntpsec* installerades genom lösningen som syns i Figur 2 så valde jag att fortsätta med antagandet att *ntp*-paketet fortfarande kommer att fungera då det är ett övergångspaket. Tyvärr gjorde varken *changelog.Debian.gz* eller *NEWS.Debian.gz* något visare om hur användning och konfiguration av paketet går till. Således valde jag att ta hjälp av Debians wiki.

```
[hig-25sipe01@dvg001:~$ zmore /usr/share/doc/ntp/
changelog.Debian.gz  copyright          NEWS.Debian.gz
hig-25sipe01@dvg001:~$
```

Figur 3: Filer i katalogen */usr/share/doc/ntp*

Wikin går bland annat igenom installationen av paketet med hjälp av aptitude, men nämner också kommandot `ntpq -p` [2]. Då kommandot visar vilka serverar jag synkar med [2] så ville jag såklart testa och se hur det ser ut innan konfiguration. Således körde jag `ntpq -p`, för att se resultatet se Figur 4. Då jag inte kunde hitta någon beskrivning för konfiguration av ntp valde jag att istället leta efter en konfigfil i `/etc/`, genom att skriva `nano /etc/ntp` och därefter klicka på tabb fick jag förslaget `ntpsec`, efter att ännu en gång klicka på tabb blev resultatet `nano /etc/ntpsec/ntp.conf`, vilket jag körde. Väl inne i konfigurationsfilen hittade jag några rader som såg intressanta ut, se Figur 5. Då raderna inkluderade flera adresser till vad jag antar är olika poler av ntp-serverar så teestade jag att byta ut dessa mot poolerna som fanns tillgängliga på `se.pool.ntp.org` [3], det vill säga från `0.se.pool.ntp.org` till `3.se.pool.ntp.org`. Tillägger även att det går att läsa om konfigurationsfilen i man `ntp.conf(5)`.

remote	refid	st	t	when	poll	reach	delay	offset	jitter
0.debian.pool.ntp.org	.POOL.	16	p	-	256	0	0.0000	0.0000	0.0001
1.debian.pool.ntp.org	.POOL.	16	p	-	256	0	0.0000	0.0000	0.0001
2.debian.pool.ntp.org	.POOL.	16	p	-	256	0	0.0000	0.0000	0.0001
3.debian.pool.ntp.org	.POOL.	16	p	-	256	0	0.0000	0.0000	0.0001
-time100.stupi.se	.PPS.	1	u	1008	1024	377	3.1989	-0.1624	0.0644
+time.cloudflare.com	10.128.8.73	3	u	923	1024	377	3.3563	-0.1038	0.0726
-sth4.ntp.netnod.se	.PPS.	1	u	952	1024	377	3.0713	-0.0101	0.0656
+ntp3.flashdance.cx	193.11.166.52	2	u	914	1024	377	3.2233	-0.0779	0.0700
*ntp.netnod.se	.PPS.	1	u	40	1024	377	3.2160	-0.0864	0.0781
+sth1.ntp.netnod.se	.PPS.	1	u	584	1024	377	3.2739	-0.0761	0.0639
-192.36.143.134	.PPS.	1	u	914	1024	377	3.2847	-0.1694	0.0514

Figur 4: Utdata från kommandot `ntpq -p`.

```
# pool.ntp.org maps to about 1000 low-stratum NTP servers. Your server will
# pick a different set every time it starts up. Please consider joining the
# pool: <https://www.pool.ntp.org/join.html>
pool 0.se.pool.ntp.org iburst
pool 1.se.pool.ntp.org iburst
pool 2.se.pool.ntp.org iburst
pool 3.se.pool.ntp.org iburst
```

Figur 5: Intressanta rader i konfigurationsfilen `ntp.conf`.

Därpå kördes `sudo service ntpsec restart`, och senare `sudo service ntpsec status` för att säkerställa att programmet startats om som det skulle och fungerar. Mycket riktigt startades programmet som det skulle, se Figur 6. Anledningen till att jag använder mig av `ntpsec` grundar sig i att det var i `/etc/ntpsec` som jag hittade konfigurationsfilen för ntp. Som Figur 6 visar så verkar det som att bland annat `0.se.pool.ntp.org` numera används för tidssynkronisering. Men för säkerhets skull kördes även `ntpq -p` för att dubbelkolla att allt stämmer, för resultat se Figur 7.

```
hig-25sipe01@dvg001:/etc/ntpsec$ sudo service ntpsec restart
hig-25sipe01@dvg001:/etc/ntpsec$ sudo service ntpsec status
● ntpsec.service - Network Time Service
   Loaded: loaded (/lib/systemd/system/ntpsec.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-03-10 14:33:49 CET; 2s ago
     Docs: man:ntpd(8)
  Process: 2250 ExecStart=/usr/libexec/ntpsec/ntp-systemd-wrapper (code=exited, status=0/SUCCESS)
 Main PID: 2253 (ntpd)
    Tasks: 1 (limit: 11809)
   Memory: 10.5M
      CPU: 76ms
   CGroup: /system.slice/ntpsec.service
           └─2253 /usr/sbin/ntpd -p /run/ntpd.pid -c /etc/ntpsec/ntp.conf -g -N -u ntpsec:ntpsec

Mar 10 14:33:50 dvg001 ntpd[2253]: DNS: Pool taking: 193.182.111.13
Mar 10 14:33:50 dvg001 ntpd[2253]: DNS: Pool taking: 194.58.207.20
Mar 10 14:33:50 dvg001 ntpd[2253]: DNS: dns_take_status: 0.se.pool.ntp.org=>good, 8
Mar 10 14:33:51 dvg001 ntpd[2253]: DNS: dns_probe: 1.se.pool.ntp.org, cast_flags:8, flags:101
Mar 10 14:33:51 dvg001 ntpd[2253]: DNS: dns_check: processing 1.se.pool.ntp.org, 8, 101
Mar 10 14:33:51 dvg001 ntpd[2253]: DNS: Pool taking: 194.58.204.20
Mar 10 14:33:51 dvg001 ntpd[2253]: DNS: Pool taking: 98.128.175.45
Mar 10 14:33:51 dvg001 ntpd[2253]: DNS: Pool taking: 155.4.55.243
Mar 10 14:33:51 dvg001 ntpd[2253]: DNS: Pool taking: 77.239.113.15
Mar 10 14:33:51 dvg001 ntpd[2253]: DNS: dns_take_status: 1.se.pool.ntp.org=>good, 8
hig-25sipe01@dvg001:/etc/ntpsec$
```

Figur 6: Utdata från statusen för programmet `ntpsec`.

```
hig-25sipe01@dvg001:/etc/ntpsec$ ntpq -p
```

remote	refid	st	t	when	poll	reach	delay	offset	jitter
0.se.pool.ntp.org	.POOL.	16	p	-	256	0	0.0000	0.0000	0.0001
1.se.pool.ntp.org	.POOL.	16	p	-	256	0	0.0000	0.0000	0.0001
2.se.pool.ntp.org	.POOL.	16	p	-	256	0	0.0000	0.0000	0.0001
3.se.pool.ntp.org	.POOL.	16	p	-	256	0	0.0000	0.0000	0.0001
-192.36.143.134	.PPS.	1	u	559	1024	377	3.3391	-0.0852	0.0491
*time100.stupi.se	.PPS.	1	u	607	1024	377	3.2769	-0.0211	0.0180
-lul1.ntp.netnod.se	.PPS.	1	u	347	1024	377	14.2384	-0.0596	0.0196
+ntp3.flashdance.cx	193.11.166.52	2	u	713	1024	377	3.3399	0.0722	0.0629
+time.cloudflare.com	10.128.8.73	3	u	445	1024	377	3.4186	0.0761	0.0399
+ntp4.flashdance.cx	193.11.166.52	2	u	709	1024	377	3.2639	0.0802	0.0787
-h-155-4-55-243.a444.priv.bahnhof.se	..	1	u	526	1024	377	6.0466	-0.1190	0.8876

Figur 7: Resultat av kommandot `ntpq -p`.

2.2.2. Del 2 – Ny användare

För detta moment redovisas alla de steg som utförts och dess resultat i form av två figurer nedan, först skapandet av användaren och därefter tilldelningen av gruppen `sudo`. Se först Figur 8 och därefter Figur 9.

```
kommandon-användare.txt

Script started on 2025-03-10 14:38:30+01:00 [TERM="xterm-256color" TTY="/dev/pts/0"
COLUMNS="362" LINES="98"]
hig-25sipe01@dvg001:~/lab3$ sudo adduser deltva
[sudo] password for hig-25sipe01:
Adding user `deltva' ...
Adding new group `deltva' (1001) ...
Adding new user `deltva' (1001) with group `deltva (1001)' ...
Creating home directory `/home/deltva' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for deltva
Enter the new value, or press ENTER for the default
  Full Name []: Del tva
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
Adding new user `deltva' to supplemental / extra groups `users' ...
Adding user `deltva' to group `users' ...
hig-25sipe01@dvg001:~/lab3$
exit

Script done on 2025-03-10 14:39:46+01:00 [COMMAND_EXIT_CODE="0"]
```

Figur 8: In- och utdata från skapandet av användaren "deltva".

```
kommandon-grupp.txt

Script started on 2025-03-10 14:58:22+01:00 [TERM="xterm-256color" TTY="/dev/pts/0"
COLUMNS="362" LINES="98"]
hig-25sipe01@dvg001:~/lab3$ sudo adduser deltva sudo
[sudo] password for hig-25sipe01:
Adding user `deltva' to group `sudo' ...
Done.
hig-25sipe01@dvg001:~/lab3$ groups deltva
deltva : deltva sudo users
hig-25sipe01@dvg001:~/lab3$
exit

Script done on 2025-03-10 14:59:11+01:00 [COMMAND_EXIT_CODE="0"]
```

Figur 9: In- och utdata från tilldelningen av gruppen "sudo" för användaren "deltva".

2.2.3. Del tre - Rättigheter

För att repetera skalprogrammeringen från förra laborationen valde jag att skapa ett skript som skapar filstrukturen enligt uppgiftsbeskrivningen. Skriptet är knappast ett mästerverk och kan enkelt förbättras med hjälp av t.ex. variabler. Fokuset i skriptet varit på att kunna återskapa samma filstruktur som den jag har tagit fram snarare än att skriva den bästa koden. Dessutom är det alltid övning att skriva ut hela kommandon och/eller sökvägar. Skriptet återfinns i Bilaga 1: Skript och bygger på följande kommandon:

- `mkdir(1)`
- `chmod(1)`
- `chown(1)`
- `touch(1)`

Så snart skriptet var klart kördes skriptet genom `sudo ./del3.sh`, där `sudo` användes för att skriptet ska kunna justera rättigheter. Därpå kontrollerades rättigheterna mot uppgiftskraven med hjälp av `ls -l`, som visar de filer/kataloger och dess rättigheter i den katalog jag står i, se Figur 8 och Figur 9.

```
[deltva@dvg001:~$ sudo ./del3.sh
[deltva@dvg001:~$ cd /tmp/del3/
[deltva@dvg001:/tmp/del3$ ls -l
total 8
drwx----- 2 root    deltva 4096 Mar 10 16:21 a1
-rwxr--r--  1 deltva  root      0 Mar 10 16:21 a2
drwxr--r--  2 deltva  deltva 4096 Mar 10 16:21 a3
-rwxrwx---  1 root    root      0 Mar 10 16:21 a4
```

```
[deltva@dvg001:/tmp$ ls -l
total 12
drwxr-xr-x  4 deltva  deltva 4096 Mar 10 16:21 del3
```

Med alla nya kataloger, filer och rättigheter var det dags att testa de olika kommandon som finns med i uppgiftsbeskrivningen, nedan följer kommando efter kommando samt beskrivningar om vem som utför kommandot samt av vad som sker och varför:

- `touch /tmp/del3/a1/f1`
 - Användare: deltva
 - Vad: Skapar en tom fil `/tmp/del3/a1/f1` om den inte existerar
 - Resultat: Misslyckas, filen skapas inte.
 - Varför: Användaren saknar skrivrättigheter i katalogen `a1`

- `sudo touch /tmp/del3/a3/f1`
 - Användare: root
 - Vad: Skapar en tom fil /tmp/del3/a1/f1 om den inte existerar.
 - Resultat: Lyckas, filen skapas utan problem.
 - Varför: Användaren root har full tillgång till systemet, varför det är så viktigt att vara försiktig med såväl *sudo* som eventuellt *root-lösenord* i en Linuxmiljö.
- `sudo echo "Hello World" | tee /tmp/del3/a3/f1`
 - Användare: root för echo, deluva för tee
 - Vad: Skriver ut "Hello World" till standardutgången, med hjälp av "|" -symbolen används standardutgången från echo som indata till tee-kommandot som skriver ut indatan i såväl standardutgången som den specificerade filen (/tmp/del3/a3/f1).
 - Resultat: Lyckas med echo, misslyckas med att skriva till filen f1 genom tee men lyckas med att skriva ut till standardutgången (terminalen).
 - Varför: Användaren root har såklart full tillgång till systemet, däremot har deluva-användaren inte skrivrättigheter till katalogen a3.
- `sudo echo "Hello World" | tee /tmp/del3/a2`
 - Användare: root för echo, deluva för tee.
 - Vad: Samma som i det tidigare kommandot fast för /tmp/del3/a2 som fil istället.
 - Resultat: Lyckas med såväl echo som tee och "Hello World" skrivs till filen a2.
 - Varför: Användaren root har såklart rättigheter för echo, och deluva-användaren har skrivrättigheter för a2.
- `sudo echo "Hello World" | tee /tmp/del3/a4`
 - Användare: root för echo, deluva för tee.
 - Vad: Samma som i det tidigare kommandot fast för /tmp/del3/a4 som fil istället.
 - Resultat: echo lyckas, men tee misslyckas.
 - Varför: Användaren deluva saknar skrivrättigheter till filen.
- `cat /tmp/del3/a2`
 - Användare: deluva
 - Vad: Skriver ut innehållet i filen /tmp/del3/a2 till standardutgången.
 - Resultat: Lyckas.
 - Varför: Användaren deluva har läsrättigheter för filen.

- `sudo cat /tmp/del3/a2`
 - Användare: root
 - Vad: Samma som tidigare kommando men som root-användare genom sudo.
 - Resultat: Lyckas.
 - Varför: Användaren root har full tillgång till hela systemet.
- `cat /tmp/del3/a4`
 - Användare: deltva
 - Vad: Samma som tidigare kommando men utan sudo och för filen /tmp/del/a4 istället.
 - Resultat: Misslyckas.
 - Varför: Användaren deltva saknar läsrättigheter för filen.
- `sudo cat /tmp/del3/a4`
 - Användare: root
 - Vad: Samma som tidigare kommando fast med sudo.
 - Resultat: Lyckas.
 - Varför: Användaren root har full tillgång till systemet.
- `echo "Goodbye World" | sudo tee /tmp/del3/a2`
 - Användare: deltva för echo, root för sudo.
 - Vad: Skriver ut "Goodbye World" till filen /tmp/del3/a2 och terminalen.
 - Resultat: Lyckas.
 - Varför: echo kräver inga särskilda rättigheter, tee körs av root som har full tillgång till systemet.
- `cat /tmp/del3/a2`
 - Användare: deltva
 - Vad: Skriver ut innehållet i /tmp/del3/a2 till terminalen.
 - Resultat: Lyckas.
 - Varför: Användaren har läsrättigheter till filen a2.

- `sudo cat /tmp/del3/a2`
 - Användare: root
 - Vad: Samma som det tidigare kommandot men genom sudo, vilket gör att root kör kommandot.
 - Resultat: Lyckas.
 - Varför: Användaren root har full tillgång till systemet.
- `sudo rm -r /tmp/del3/`
 - Användare: root
 - Vad: Raderar katalogen /tmp/del3/ rekursivt, vilket innebär att kommandot även raderar innehållet i katalogen. Utan -r går det inte att ta bort katalogen eftersom den innehåller andra kataloger/filer.
 - Resultat: Lyckas
 - Varför: Användaren root har full tillgång till systemet samt -r används.

3. Beskrivning av slutresultat

Resultatet är först och främst att ntp-paketet (samt ntpsec) är installerat och konfigurerat på systemet. Ytterligare har det tillkommit en ny användare (deltva) som tillhör såväl gruppen deltva som gruppen sudo. Slutligen har en ny katalogstruktur med diverse rättigheter skapats och raderats i /tmp. Med andra ord har väldigt enkel systemadministration redovisats genom rapporten där såväl rättigheter som programhantering har varit centralt.

4. Diskussion

Min största lärdom från uppgiften är skrapandet på ytan av ntp-paketet, något jag aldrig tidigare använt mig av. Det har varit mycket lärorikt att leta runt i /usr/share/doc/PAKET, då detta är något jag inte stött på tidigare i min användning av någon Linuxdistribution. Jag gillar idén även om just ntp-paketet hade den README.Debian.gz filen jag förväntade mig. Det var inte riktigt helt lätt att veta hur jag förväntas konfigurera och hantera ntp-paketet i samband med det nya paketet *ntpsec* men tack vare mina antaganden så upplever jag att det gått bra iallafall. Något annat som var nytt för mig var *aptitude*, jag har så länge jag kan minnas utnyttjat mig av *apt-get* eller *apt* i Ubuntu så *aptitude* var helt nytt. Precis som kursboken nämner så var aptitude väldigt likt apt i sin syntax – vilket kändes bekvämt.

5. Slutsatser

Sammanfattningsvis beskriver rapporten mycket väl hur grundläggande systemadministration i form av såväl användar-, grupp- och rättighetshantering. Yttermera beskriver rapporten hur konfiguration och installation av paketet ntp/ntpsec kan gå till – något som skulle kunna appliceras på andra paket.

6. Referenser

Litteraturförteckning

[1]: R. Hertzog, R. Mas, The Debian Administrator's Handbook, Debian Buster from Discovery to Mastery, 2020.

[2]: Debian, "NTP," Debian Wiki. [Online]. Tillgänglig: <https://wiki.debian.org/NTP>. Besökt: 2025-03-08.

[3]: NTP Pool Project, "Sweden — se.pool.ntp.org" NTP Pool. [Online]. Tillgänglig: <https://www.ntppool.org/zone/se>. Besökt: 2025-03-10.

I. Bilaga 1: Skript

```
#!/usr/bin/bash

# baserat på rättigheterna ser vi att del3 är en katalog, således skapar vi den
mkdir /tmp/del3
# Nedan justeras rättigheterna baserat på F3.pdf
chmod ugo+x,ug+r,u+r /tmp/del3
# Justerar ägarskap med chown baserat på man
chown deltva:deltva /tmp/del3

# Även a1 är en katalog
mkdir /tmp/del3/a1
# justerar rättigheter
chmod go=-rwx,u+rwx /tmp/del3/a1
# justerar ägarskap
chown root:deltva /tmp/del3/a1

# baserat på rättigheterna så kan vi utläsa att detta ska vara en fil
touch /tmp/del3/a2
# rättigheter
chmod u+rwx,go=r /tmp/del3/a2
# ägarskap
chown deltva:root /tmp/del3/a2

# denna gång ytterligare en katalog
mkdir /tmp/del3/a3
# rättigheter
chmod u+rwx,go=r /tmp/del3/a3
# ägarskap
chown deltva:deltva /tmp/del3/a3

# dags för en fil igen
touch /tmp/del3/a4
# rättigheter
chmod ug=rwx,o=-rwx /tmp/del3/a4
# ägarskap
chown root /tmp/del3/a4
```