

Unix/Linux — en annen måte å gjøre ting på

Dag Langmyhr

Institutt for informatikk

Operativsystemer

Alle datamaskiner (unntatt de helt knøttsmå) har et *operativsystem* som

- administrerer datamaskinen og
- tar seg av kommunikasjonen mellom bruker og maskin.

For «vanlige datamaskiner» dominerer tre OS:

- Windows (XP, Vista, Windows7, Windows8, ...)
- Linux og andre Unix-er
- MacOS X (som er en Unix i bunnen)

Unix kontra Windows

Alder

Unix er opprinnelig fra 1970-årene og følgerlig kommandobasert.

Windows er fra 1990-årene og orientert mot «pek-og-klikk».

Dette forklarer noen av forskjellene:

- Windows er enklere å *bruke* for nybegynnere og folk med lite datakunnskap.
- Det er enklere å *utvikle* nye programmer under Unix.



Store og små programmer

Under Windows kjører man gjerne noen få store programmer; under Unix kjører man heller flere små.

For eksempel: Jeg ønsker en sortert oversikt over hvem som kjører på samme datamaskin som jeg:

```
dag@diamant ~$ who | awk '{print $1}' | sort | uniq | column -x
aasmunds    andersmo    bjarnej     cristi       dag
eivinah      elien       endreak     ernstgr      eskil
fredrihl     fredrw      haakonks    hilia        jae
jimrj        juanle     larsg       larsha       mariusno
martbo       martinwa    martiso     mshaugla    nikolahl
oleiul       oysteint   per         siri         sivertkh
svenar       sveniu     vegardno    viet
```



Hvert lille program har sin spesielle oppgave:

who lager en oversikt over alle som er logget inn på datamaskinen:

```
dag@diamant ~$ who
larsha pts/9 2011-08-01 19:16 (c4baf5bc1.dhcp)
martiso pts/25 2011-08-16 07:23 (193.71.175.103)
:
```

awk plukker ut kolonne 1.

sort sorterer navnene.

uniq fjerner duplikater.

column setter opp svaret i pene kolonner.

Unix-måten er tyngre å lære men gir flere muligheter.

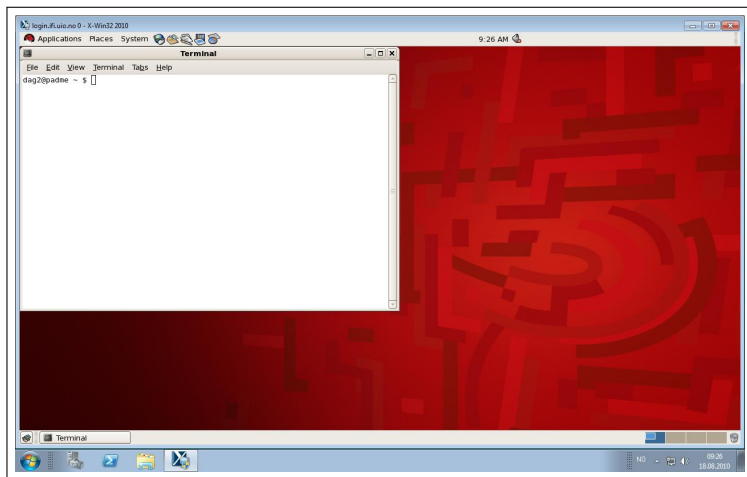


Hvorfor Unix?

Det er flere grunner til at *dere* skal lære Unix:

- Dere kan bruke Windows fra tidligere.
- Unix utgjør en stor minoritet – som fremtidige dataeksperter må dere kjenne til det.
- Det er alltid nyttig å lære to varianter av noe: naturlige språk, operativsystemer, programmeringsspråk, dokumentverktøy, etc.
- Enkelte hjelpemidler dere skal bruke etter hvert i studiet, finnes bare for Unix.

Standard oppsett ved Ifi



Brukernavn og passord

For å få tilgang til datamaskinene, trenger man **brukernavn** og **passord**. Det skal dere ha fått tilsendt.

Å endre passordet

På <http://brukerinfo.uio.no> kan man blant annet endre passordet sitt. Velg et på minst 8 tegn med minst ett tegn fra tre av disse gruppene:

- liten bokstav
- stor bokstav
- siffer 0-9
- annet tegn

Hint

Gjør det alltid på slutten av dagen.

Eksempel på passord

Ne Ne
& and (på engelsk)
= er (som i $1 + 1 = 2$)
1234 tal(l)

nE&=1234 blir «Neandertal».

Hint

Om du bruker Mac, unngå tegn plassert ulikt på tastaturet i forhold til Windows og Linux.

Egentlig har vi 4 skjermer

Virtuelle skjermer

Linux-en vår har egentlig 4 skjermer vi kan veksle mellom.
Se «kartet» i nedre høyre hjørne.

Hva er en fil?

Windows og Mac er *programorienterte*: alle filer «hører til» et program. Dokumenter tilhører Word, bilder er koblet til The Gimp (eller tilsvarende), osv.

Unix er *filorientert*: filer eksisterer for seg selv, og brukeren velger fritt hvilke verktøy han eller hun vil bruke på dem.

Tekstfiler

Slike filer, som inneholder «ren» tekst, finnes knapt nok i Windows der brukerne ser dem, men er helt grunnleggende i Unix.

La oss lage en tekstfil med arbeidshesten **Emacs**.


(Om du ikke liker Emacs, finnes det alternativer; for eksempel **TextPad** for Windows.)

Emacs

Vi kan starte redigeringsprogrammet Emacs ved å velge i menyene:



Applications → Ifi Programs → GNU Emacs

eller med kommandoen





emacs & 

Hva kan jeg gjøre i Emacs?

Det mest nødvendige er

- å hente inn en tekstfil (med mappeikonet eller Visit New File i File-menyen eller  + )

NB! Legg merke til at spørsmål om filnavnet havner *nederst* i vinduet!

- å endre på filen, og
- å skrive filen tilbake til disken igjen (med diskettikonet eller Save i File-menyen eller  + 
 + ).

Ondsinnet rykte:

Navnet Emacs står for *Escape-Meta-Alt-Control-Shift*.



Et lite hint

Alle Emacs-kommandoene kan utføres med tastaturet (i stedet for med musen). Prøv å lære å bruke disse.

- Arbeidet går raskere.
- Det reduserer faren for *musesyke* i fremtiden.
- Du finner tastekombinasjonen i menyene.

Hva mer kan Emacs gjøre?

- Emacs kan spille spillet *fem på rad*:

Tools → Games → Gomoku

- Du kan legge dine problemer frem for Emacs-psykologen:

Help → Emacs Psychotherapist

Filer

Hvilke filer har jeg?

Kommandoen `ls` (for «list files») gir en oversikt over de filene jeg har. Den benyttes i kommandovinduet:

```
$ ls  
brev  huskeliste  inf1000.info
```

Man kan bruke *joker-notasjon* med `*` for å angi bare noen filer. Hvis jeg vil se hvilke filer som begynner med `b`, skriver jeg

```
$ ls b*  
brev
```

Å skrive på laserskriver

Kommandoen `print` brukes til å skrive ut filer på laserskriver:

```
$ print -multi8 ~dag/www_docs/unix-1+2.pdf
Queued 4 sheets on printer nimbus:
  PDF-BEAMER file /hom/dag/www_docs/unix-1+2.pdf.
You are first in the queue to nimbus.
```

Så er det bare å vente på utskriften som kommer på den aktuelle skriveren.

De viktigste skriverne

På <http://www.mn.uio.no/ifi/tjenester/it/hjelp/utskrift/>
finner vi:

bodoni OJD 3267

bookman OJD biblioteket

caslon OJD 3240

futura OJD 3204

garamond OJD 2467

lucida OJD 2242

Hva kan jeg gjøre med filer?

Diverse fil-kommandoer

Lage kopi av filen

```
$ ls  
brev  huskeliste  inf1000.info  
$ copy brev brev-anne  
$ ls  
brev  brev-anne  huskeliste  inf1000.info
```

Lage en tom fil

```
$ touch nyfil
```

Fjerne filen

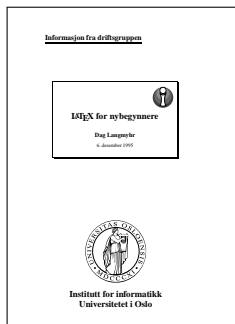
```
$ del huskeliste  
$ ls  
brev  brev-anne  inf1000.info  nyfil
```



Filtyper

Hittil har vi kun sett på tekstfiler, men det finnes ganske mange andre typer, for eksempel

- programmer som datamaskinen kan utføre
- bilder
- dokumenter



Filtyper og filnavn

Det er vanlig å la slutten av filnavnet angi hva slags fil det er snakk om, men det er intet krav i Unix. De vanligste endelsene er:

Endelse	Filen inneholder
.java	Et Java-program
.class	Et kompilert Java-program
.py	Et Python-program
.pdf	En PDF-fil (dvs et dokument)
.gz	En fil som er komprimert med gzip

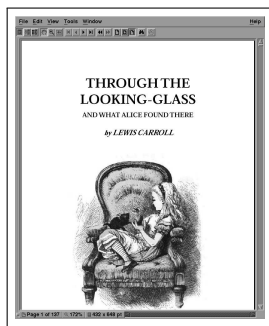
Programmer har vanligvis ingen endelse (f.eks. ls).

Ser filer på skjermen

Programmet preview

preview er laget for å vise alle type filer på skjermen; det vil automatisk velge riktig verktøy for jobben:

```
$ preview ~dag/www_docs/books/alice-tlg.pdf
```



Beskyttelse av filer

Kommandoen `ls` forteller litt mer hvis vi gir den *opsjonen* `-l`:

```
$ ls -l
total 0
-rw----- 1 dag 52 Jan 12 11:34 brev
-rw----- 1 dag 52 Jan 12 11:46 brev-anne
-rw----- 1 dag 0 Jan 12 11:13 brev~
-rw-r--r-- 1 dag 0 Jan 12 11:13 inf1000.info
```

Her ser vi at bare jeg kan lese brevene, men alle kan lese `inf1000.info`.

For å forstå beskyttelsen av filer i Unix, må man vite at Unix opererer med tre kategorier brukere:

- user** (forkortet «**u**») er filens eier, dvs den som opprettet den.
- group** (forkortet «**g**») er gruppen tilordnet filen. (Mer om dette siden.)
- other** (forkortet «**o**») er alle andre brukere.

Det finnes tre privilegier for filer:

read (forkortet «**r**») er retten til å lese filen.

write (forkortet «**w**») er retten til å skrive på filen, dvs endre den.

execute (forkortet «**x**») er retten til å utføre den, dvs bruke den som et program.

Privilegiene angis i bolker på 3×3 bokstaver:

u g o
rwx r-x r--

En bokstav angir at man har privilegiet, en «-» at man ikke har det.

Hvis jeg ønsker å la alle få lese brev men ingen andre få lese inf1000.info, skriver jeg

```
$ chmod go+r brev
$ chmod go-r inf1000.info
```

Nå ser beskyttelsen slik ut:

```
$ ls -l
total 0
-rw-r--r-- 1 dag 52 Jan 12 11:34 brev
-rw----- 1 dag 52 Jan 12 11:46 brev-anne
-rw----- 1 dag 0 Jan 12 11:13 brev~
-rw----- 1 dag 0 Jan 12 11:13 inf1000.info
```

Filer og sikkerhet

Filer fjernet med del er ikke fortapt.

lsdel gir en oversikt over «fjernede» filer.

undel henter filene tilbake.

Filene forsvinner først når man logger ut.

Hva om jeg mister filene mine?

«Snapshots»

Brukerfilene ligger på egne filmaskiner med spesielle egenskaper, for eksempel lagres «snapshots» hvor fjernete filer ligger i omtrent et døgn. Se på mappen `~/snapshot`.

Spesielle programmer

Noen programmer — som Emacs — lager automatisk sikkerhetskopier:

`#filnavn#` er en kopi av filen som endres for øyeblikket; kopien tas hvert 5. minutt. Den fjernes når filen skrives til disk.

`filnavn~` er forrige lagrete versjon. Den fjernes etter tre dager.



Siste sjanse

Hver natt tas kopi av alle brukerfiler. Om du trenger det, kan du sende en e-post til restore@usit.uio.no og be dem hente filen tilbake.