



Laboration – Navigera filsystemet

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:

- Navigera till hem- och systemkataloger
- Lista filer och mappar

Filer och mappar

I den här uppgiften kommer du att utforska begreppen filer och kataloger.

På ett Linux-operativsystem lagras data i *filer* och filer lagras i *kataloger*. Du kanske är van vid att använda termen *mappar* för att beskriva kataloger.

Kataloger är faktiskt också filer; den data de innehåller är namnen på de filer som har lagts in i dem, tillsammans med inode-numret (ett unikt identifieringsnummer som tilldelas varje fil) för var filens data finns på disken.

Som Linux-användare vill du kunna hantera dessa filer och kataloger, inklusive hur du listar filer i en katalog, kopierar, tar bort och flyttar filer.

Varning

Fil- och katalognamn i Linux är skiftlägeskänsliga *case-sensitive*. Det innebär att en fil som heter ABC inte är samma som en fil som heter abc.

Skriv följande kommando för att skriva ut *arbetskatalogen* :

```
pwd
```

```
sysadmin@localhost:~$ pwd  
/home/sysadmin
```

Arbetskatalogen är den katalog där din terminal just nu "befinner sig". Detta kallas också för *nuvarande katalog*. Det här kommer vara viktigt för när du kör kommandon, eftersom de betar sig olika beroende på vilken katalog du är i.

Utdata från `pwd`-kommandot (`/home/sysadmin` i exemplet ovan) kallas för *sökväg*. Det första snedstreckat representerar rotkatalogen, den översta nivån i katalogstrukturen.

I utdata ovan är `home` en katalog under rotkatalogen och `sysadmin` är en katalog under `home`-katalogen.

När du öppnar ett terminalfönster hamnar du först i din hemkatalog. Det är en katalog där du har full åtkomst och där andra användare normalt inte har tillgång som standard. För att se sökvägen till din hemkatalog kan du köra följande kommando för att visa värdet på `HOME`-variabeln:



```
echo $HOME
```

```
sysadmin@localhost:~$ echo $HOME  
/home/sysadmin
```

Du kan använda kommandot `cd` tillsammans med en sökväg till en katalog för att byta din nuvarande katalog. Skriv följande kommando för att göra rotkatalogen till din aktuella arbetskatalog och kontrollera det med kommandot `pwd`:

```
cd /  
pwd
```

```
sysadmin@localhost:~$ cd /  
sysadmin@localhost:/$ pwd  
/  
sysadmin@localhost:/$
```

För att gå tillbaka till din hemkatalog kan du köra kommandot `cd` utan någon sökväg. Byt tillbaka till din hemkatalog och kontrollera detta genom att skriva följande kommandon:

```
cd  
pwd
```

```
sysadmin@localhost:/$ cd  
sysadmin@localhost:~$ pwd  
/home/sysadmin  
sysadmin@localhost:~$
```

Lägg märke till att prompten ändras. Tilde-tecknet (`~`) representerar din hemkatalog. Den här delen av prompten visar vilken katalog du befinner dig i just nu.

Kommandot `cd` kan användas tillsammans med en sökväg till en katalog angiven som ett *argument*. Kör kommandot `cd` med katalogen `/home` som argument genom att skriva följande:

```
cd /home  
pwd
```

```
sysadmin@localhost:~$ cd /home  
sysadmin@localhost:/home$ pwd  
/home  
sysadmin@localhost:/home$
```

När sökvägen som anges som argument till kommandot `cd` börjar med snedstreck (`/`), kallas den för en "absolut sökväg". Absoluta sökvägar är alltid fullständiga vägar från rotkatalogen till en underkatalog eller fil.

Byt tillbaka till din hemkatalog genom att använda kommandot `cd` med tilde-tecknet (`~`) som argument:

```
cd ~  
pwd
```



```
sysadmin@localhost:~/home$ cd ~
sysadmin@localhost:~$ pwd
/home/sysadmin
sysadmin@localhost:~$
```

När sökvägen som anges som argument till kommandot `cd` börjar med ett tilde-tecken (`~`), kommer terminalen att expandera tecknet till hemkatalogen för den användare som har ett konto på systemet.

Om inget annat tecken eller ett snedstreck följer efter tilde-tecknet så expanderas det till hemkatalogen för **den användare som för tillfället är aktiv** i skalet.

Om ett användarnamn direkt följer efter tilde-tecknet kommer skalet att expandera tilde och användarnamn till hemkatalogen för det användarnamnet. Exempelvis kommer `~bob` att expanderas till `/home/bob`.

Sökvägar som börjar med tilde betraktas som absoluta sökvägar eftersom det bildas en absolut sökväg efter att skalet har expanderat tilde-vägen.

Använd kommandot `echo` nedan för att visa några andra exempel på hur tilde används som en del av sökvägen:

```
echo ~ ~sysadmin ~root ~mail ~nobody
```

```
sysadmin@localhost:~$ echo ~ ~sysadmin ~root ~mail ~nobody
/home/sysadmin /home/sysadmin /root /var/mail /nonexistent
sysadmin@localhost:~$
```

Försök att byta till root-användarens hemkatalog genom att skriva följande kommando:

```
cd ~root
```

```
sysadmin@localhost:~$ cd ~root
-bash: cd: /root: Permission denied
sysadmin@localhost:~$
```

Observera felmeddelandet; det visar att skalet försökte köra `cd` med `/root` som argument men misslyckades eftersom behörighet saknas. Du kommer att lära dig mer om fil- och katalogbehörigheter i en senare labb.

Använd en absolut sökväg för att byta till katalogen `/usr/bin` och visa den aktuella katalogen med följande kommandon:

```
cd /usr/bin
```

```
pwd
```

```
sysadmin@localhost:~$ cd /usr/bin
sysadmin@localhost:/usr/bin$ pwd
/usr/bin
sysadmin@localhost:/usr/bin$
```

Använd en absolut sökväg för att byta till katalogen `/usr` och visa den aktuella katalogen med följande kommandon:



```
cd /usr
```

```
pwd
```

```
sysadmin@localhost:/usr/bin$ cd /usr
sysadmin@localhost:/usr$ pwd
/usr
sysadmin@localhost:/usr$
```

Använd en absolut sökväg för att byta till katalogen `/usr/share/doc` och visa den aktuella katalogen med följande kommandon:

```
cd /usr/share/doc
```

```
pwd
```

```
sysadmin@localhost:/usr$ cd /usr/share/doc
sysadmin@localhost:/usr/share/doc$ pwd
/usr/share/doc
sysadmin@localhost:/usr/share/doc$
```

Absoluta vs. relativa sökvägar

Anta att du befinner dig i katalogen `/usr/share/doc` och vill gå till katalogen `/usr/share/doc/bash`. Att skriva kommandot `cd /usr/share/doc/bash` innebär en hel del skrivande. I sådana fall vill du använda relativa sökvägar.

Med relativa sökvägar anger du "vägbeskrivning" från den aktuella katalogen till dit du vill gå. Följande exempel visar hur du kan använda relativa sökvägar.

Använd en relativ sökväg för att byta till katalogen `/usr/share/doc/bash` och visa den aktuella katalogen med följande kommandon:

```
cd bash
```

```
pwd
```

```
sysadmin@localhost:/usr/share/doc$ cd bash
sysadmin@localhost:/usr/share/doc/bash$ pwd
/usr/share/doc/bash
sysadmin@localhost:/usr/share/doc/bash$
```

Observera

Om det inte fanns någon bash-katalog under den aktuella katalogen skulle föregående kommando misslyckas.

Använd en relativ sökväg för att byta till katalogen ovanför den nuvarande katalogen:

```
cd ..
```

```
pwd
```



```
sysadmin@localhost:/usr/share/doc$ cd ..
sysadmin@localhost:/usr/share/doc$ pwd
/usr/share/doc
sysadmin@localhost:/usr/share/doc$
```

`..` betyder att du går upp en nivå från den katalog du befinner dig i.

Använd en relativ sökväg för att gå upp en nivå från den aktuella katalogen och sedan ner i `dict`-katalogen:

```
cd ../dict
```

```
pwd
```

```
sysadmin@localhost:/usr/share/doc$ cd ../dict
sysadmin@localhost:/usr/share/dict$ pwd
/usr/share/dict
sysadmin@localhost:/usr/share/dict$
```

Lista filer och mappar

För att lista innehållet i den aktuella katalogen använder du kommandot `ls`:

```
cd
```

```
ls
```

Din utdata bör se ut ungefär så här:

```
sysadmin@localhost:/usr/share/dict$ cd
sysadmin@localhost:~$ ls
Desktop Documents Downloads Music Pictures Public Templates
Videos
sysadmin@localhost:~$
```

I utdata från det tidigare `ls`-kommandot visades filnamnen i ljusblå färg. Detta är en funktion som många Linux-distributioner automatiskt erbjuder genom något som kallas alias (mer om detta i en senare övning).

Färgen visar vilken typ av objekt det är. Följande tabell beskriver några av de vanligaste färgerna:

Färg	Typ av Fil
Svart eller Vit	Vanlig fil
Blå	Mapp (Directory file)
Cyan	Symbolisk länk (en fil som pekar till en annan fil)
Green	Exekverbar fil (ett program)

Alla filer visas inte som standard. Det finns filer, så kallade dolda filer, som inte visas automatiskt. För att visa alla filer, även dolda, används `-a`-alternativet till `ls`-kommandot:

```
ls -a
```



```
sysadmin@localhost:~$ ls -a
.          .bashrc   .selected_editor  Downloads  Public
..         .cache    Desktop           Music      Templates
.bash_logout .profile  Documents         Pictures   Videos
sysadmin@localhost:~$
```

Namn på dolda filer börjar med **en punkt** (ett punkttecken). Ofta är dessa filer och ibland mappar dolda eftersom det inte är sådant du normalt vill se.

Exempelvis innehåller filen `.bashrc`, som visas i exemplet ovan, konfigurationsinformation för bash-skalet. Det är en fil du vanligtvis inte behöver titta på regelbundet.

Två viktiga "punktfiler" finns i varje katalog: `.` (som representerar den aktuella katalogen) och `..` (som representerar katalogen ovanför den aktuella).

Om du bara skriver kommandot `ls` får du endast namnen på filer och kataloger i den angivna (eller aktuella) katalogen. Kör följande kommando för att se hur `-l`-flaggan ger mer information om en fil:

```
ls -l /etc/hosts
```

Din utskrift bör se ut ungefär så här:

```
sysadmin@localhost:~$ ls -l /etc/hosts
-rw-r--r-- 1 root root 150 Jan 22 15:18 /etc/hosts
sysadmin@localhost:~$
```

Men vad betyder all den här extra informationen egentligen? Tabellen nedan ger en kort genomgång av vad varje del i utskriften från `ls -l` betyder:

-	Det första tecknet, ett - i exemplet ovan visar vilken typ av "fil" det är. Ett - tecken betyder att det är en vanlig fil, medan ett d tecken betyder att det är en katalog.
rw-r--r--	Detta visar filens behörigheter. Behörigheter behandlas i en senare övning.
1	Detta visar något som kallas hårdlänkantal (förklaras senare).
root	Filens ägare.
root	Filen ägargrupp.
150	Storleken för filen angiven i bytes
Jan 22 15:18	Datum/tid då filen senast ändrades.

Ibland vill du inte bara se innehållet i en katalog, utan även innehållet i dess underkataloger. Du kan använda `-R`-flaggan för att göra detta:

```
ls -R /etc/udev
```

```
sysadmin@localhost:~$ ls -R /etc/udev
/etc/udev:
hwdb.d  rules.d  udev.conf

/etc/udev/hwdb.d:

/etc/udev/rules.d:
70-persistent-cd.rules  README
sysadmin@localhost:~$
```



`-R`-alternativet står för "rekursiv". Alla filer i katalogen `/etc/udev` kommer att visas, liksom alla filer i varje underkatalog, till exempel underkatalogen `rules.d`.

Var försiktig med `-R`-alternativet. Vissa kataloger är riktigt, riktigt stora!

Du kan använda fil *globbing* (wildcards) för att begränsa vilka filer eller kataloger du vill se. Till exempel matchar tecknet `*` "noll eller fler av valfria tecken" i ett filnamn. Kör följande kommando för att bara visa filer som börjar med bokstaven `s` i katalogen `/etc`:

```
ls -d /etc/s*
```

Din utdata bör likna följande:

```
sysadmin@localhost:~$ ls -d /etc/s*
/etc/securetty      /etc/sgml          /etc/shells        /etc/ssl
/etc/sysctl.conf    /etc/security      /etc/shadow         /etc/skel
/etc/sudoers        /etc/sysctl.d      /etc/services      /etc/shadow-
/etc/ssh            /etc/sudoers.d     /etc/systemd
```

Observera att `-d`-alternativet hindrar filer från underkataloger från att visas. Det bör alltid användas med `ls`-kommandot när du använder filglobbing.

Tecknet `?` kan användas för att matcha exakt ett tecken i ett filnamn. Kör följande kommando för att visa alla filer i katalogen `/etc` som är exakt fyra tecken långa:

```
ls -d /etc/????
```

Din utdata bör likna följande:

```
sysadmin@localhost:~$ ls -d /etc/????
/etc/bind           /etc/init          /etc/motd          /etc/perl          /etc/skel
/etc/dpkg           /etc/ldap          /etc/mtab          /etc/sgml          /etc/udev
sysadmin@localhost:~$
```

Genom att använda hakparenteser `[]` kan du ange ett tecken som ska matchas från en uppsättning tecken. Kör följande kommando för att visa alla filer i katalogen `/etc` som börjar med bokstäverna `a`, `b`, `c` eller `d`:

```
ls -d /etc/[abcd]*
```

Din utdata bör likna följande:

```
sysadmin@localhost:~$ ls -d /etc/[abcd]*
/etc/adduser.conf      /etc/blkid.conf    /etc/cron.weekly
/etc/adjtime           /etc/blkid.tab     /etc/crontab
/etc/alternatives      /etc/ca-certificates /etc/dbus-1
/etc/apparmor.d        /etc/ca-certificates.conf /etc/debconf.conf
/etc/apt               /etc/calendar      /etc/debian_version
/etc/bash.bashrc       /etc/cron.d        /etc/default
/etc/bash_completion.d /etc/cron.daily    /etc/deluser.conf
/etc/bind              /etc/cron.hourly   /etc/depmod.d
/etc/bindresvport.blacklist /etc/cron.monthly /etc/dpkg
sysadmin@localhost:~$
```