

Avancerad användarhandbok för Solaris

Sun Microsystems AB
Box 51
164 94 Kista

Besöksadress: Esbogatan 18

Artikelnr: 801-7485-10
Utgåva A, augusti 1994



© 1994 Sun Microsystems, Inc.
2550 Garcia Avenue, Mountain View, California 94043-1100 USA

Alla rättigheter förbehålles. Denna produkt och annan dokumentation som hör ihop med den är copyright-skyddad. Den distribueras med en licens som reglerar hur den får användas, kopieras, distribueras och omarbetas. Denna produkt eller delar av den får inte utan upphovsmannens skriftliga tillåtelse kopieras på något sätt.

Delar av denna produkt härrör från UNIX® och Berkeley 4.3 BSD, för vilka Sun har licenser från UNIX Systems Laboratories, Inc., ett av Novell, Inc. helägt bolag, respektive University of California. Teckensnitt från tredjepartsleverantörer som används är copyright-skyddade och används med licens från leverantörerna.

Den produkt som beskrivs i den här handboken kan vara skyddad av ett eller flera patent i USA, andra länder och/eller av patentansökningar.

VARUMÄRKEN

Sun, Sun Microsystems, Suns logotyp, SMCC, SunSoft, SunSofts logotyp, Solaris, SunOS, OpenWindows, DeskSet, ONC, ONC+ och NFS är varumärken eller registrerade varumärken för Sun Microsystems, Inc i USA och vissa andra länder . UNIX och OPEN LOOK är registrerade varumärken för UNIX System Laboratories, Inc, ett av Novell, Inc. helägt bolag, PostScript och Display PostScript är varumärken för Adobe Systems, Inc. Alla andra produktnamn som nämns är varumärken för sina respektive ägare.

Alla SPARC-varumärken, till exempel logotypen för SCD Compliant, är varumärken eller registrerade varumärken för SPARC International, Inc. Sun Microsystems, Inc. har en exklusiv licens för SPARCstation, SPARCserver, SPARCengine, SPARCstorage, SPARCware, SPARCcenter, SPARCclassic, SPARCcluster, SPARCdesign, SPARC811, SPARCprinter, UltraSPARC, microSPARC, SPARCworks och SPARCcompiler. Produkter med varumärket SPARC är baserade på en arkitektur som är utvecklad av Sun Microsystems, Inc.

OPEN LOOK® och Suns™ grafiska användargränssnitt har utvecklats av Sun Microsystems, Inc. för dess användare och licenstagare. Sun erkänner Xerox banbrytande insatser inom forskningen om och utvecklingen av begreppet visuellt eller grafiskt användargränssnitt för datorindustrin. Sun har en icke-exklusiv licens från Xerox avseende Xerox grafiska användargränssnitt, som också omfattar Suns licenstagare som utvecklar grafiska användargränssnitt enligt OPEN LOOK och för övrigt uppfyller Suns skrivna licensavtal.

X Window System är ett varumärke och en produkt från Massachusetts Institute of Technology.

DENNA HANDBOK LEVERERAS I BEFINTLIGT SKICK UTAN NÅGRA SOM HELST GARANTIER. SUN MICROSYSTEMS, INC. GARANTERAR TILL EXEMPEL INTE ATT DE BESKRIVNA PRODUKTERNA ÄR I SÄLJBART SKICK, ATT DE ÄR LÄMPLIGA FÖR ETT VISST ÄNDAMÅL ELLER ATT DE INTE INKRÄKTAR PÅ ANDRA FÖRETAGS RÄTTIGHETER.

I DENNA HANDBOK KAN DET FÖREKOMMA TEKNISKA FELAKTIGHETER OCH TRYCKFEL. MED JÄMNA MELLANRUM GÖRS ÄNDRINGAR I DEN PRODUKT OCH/ELLER DE PROGRAM SOM BESKRIVS HÄR. KOMMANDE UPPLAGOR AV HANDBOKEN KOMMER ATT INNEHÅLLA INFORMATION OM DESSA ÄNDRINGAR. SUN MICROSYSTEMS, INC. KAN NÄR SOM HELST ÄNDRA PRODUKTEN OCH/ELLER PROGRAMMEN SOM BESKRIVS I DENNA HANDBOK.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the United States Government is subject to the restrictions set forth in DFARS 252.227-7013 (c)(1)(ii) and FAR 52.227-19.

Lokalanpassad i Irland.



Papperet
kan
återvinnas



Adobe PostScript

Innehåll

Om den här boken.	xiii
Konventioner gällande plattformar.	xvi
1. Logga in i SunOS och starta OpenWindows	1
Logga in	1
Ditt inloggningskal	2
Logga ut	3
Snabbkommandon.	4
Att tänka på innan du startar OpenWindows	5
Omgivningsvariabeln OPENWINHOME	6
Använda rätt startfil	7
Starta OpenWindows	9
Starta OpenWindows	9
Om OpenWindows inte startar	11
Avsluta OpenWindows	11
Starta OpenWindows med tillägg	12

Starta med reducerad nätverkssäkerhet	12
Starta OpenWindows med olika typer av bildskärmar . . .	12
<code>SPARC</code> Starta Open Windows på flera bildskärmar	13
2. Grundläggande SunOS-kommandon	17
Kommandotexten	17
Ge kommandon	18
Korrigera skrivfel	18
Skriva långa kommandon och flera kommandon på samma rad	19
Upprepa föregående kommandon	19
Använda tillägg till kommandon	21
Omdirigera dataflöde	21
Utföra kommandon i bakgrunden	22
Få hjälp om SunOS-kommandon	23
Ta fram direkthjälp om ett kommando med kommandot <code>man</code>	23
Ta fram en kort sammanfattning av ett kommando med kommandot <code>what is</code>	24
Sökning efter nyckelord med kommandot <code>apropos</code>	24
3. Arbeta med filer och kataloger	25
Viktiga begrepp	25
Använda kommandon för filhantering	26
Innan du börjar	26
Skapa en testfil	27
Kontrollera innehållet i en katalog (<code>ls</code>)	27
Kopiera filer (<code>cp</code>)	27

Flytta och byta namn på filer (<code>mv</code>)	28
Ta bort filer (<code>rm</code>)	28
Titta på innehållet i filer (<code>more</code> , <code>cat</code>)	29
Kontrollera filtyp (<code>file</code>)	30
Kataloger och katalogstruktur	30
Katalogstruktur	30
Kontrollera aktuell katalog (<code>pwd</code>)	31
Hemkatalogen	32
Byta arbetskatalog (<code>cd</code>)	32
Skapa kataloger (<code>mkdir</code>)	34
Relativa sökvägar	34
Flytta och byta namn på kataloger	35
Kopiera kataloger	35
Ta bort kataloger (<code>rmdir</code>)	36
Jämföra filer (<code>diff</code>)	36
Jämföra tre filer (<code>diff3</code>)	39
Använda <code>bdiff</code> för att jämföra stora filer	39
Söka efter filer (<code>find</code>)	39
Behörighet till filer och kataloger	42
Ta fram information om behörighet och status (<code>ls -l</code>)	43
Ta fram information om dolda filer (<code>ls -a</code>)	44
Ändra behörighet (<code>chmod</code>)	45
Ställa in absolut behörighet	48

4. Söka i filer	51
Söka efter mönster med grep	51
grep som ett filter	52
grep med strängar som består av flera ord	53
Söka efter rader utan textsträngar	54
Mer om grep	54
Söka efter metatecken	56
Enkla eller dubbla citationstecken på kommandoraden ..	57
5. Lösenord, processer och skivminnen	59
Använda lösenord	59
Byta lösenord	60
Tidsbegränsade lösenord	61
Processer och process-id	62
Kontrollera vilka kommandon som utförs (ps)	62
Avbryta processer (kill)	63
Hantera skivminnen	64
Kontrollera utrymme på skivminnen (df -k)	64
Kontrollera utrymmet i en katalog (du)	64
6. Redigeringsprogrammet vi	67
Starta vi	68
Skapa en fil	68
Statusraden	69
De två arbetslägena i vi	69

Inmatningsläget	69
Kommandoläget	70
Avsluta redigering	71
Spara ändringar och avsluta vi	71
Skriva ut filer på skrivaren	72
Grundläggande vi-kommandon	73
Förflyttning i filer	73
Infoga text	76
Ändra text	77
Ångra ändringar	78
Ta bort text	79
Kopiera och flytta text	80
Upprepa kommandon	81
Använda ex-kommandon	82
Sätta på och stänga av visning av radnummer	82
Kopiera rader	83
Flytta rader	84
Ta bort rader	85
Söka och ersätta text med vi	85
Söka efter en teckensträng	85
Förbättra sökningen	86
Byta ut en teckensträng	88
Flytta markören till en viss rad	89
Infoga en fil i en annan	89

Redigera flera filer samtidigt	90
Redigera flera filer i en följd	90
Kopiera rader mellan filer	91
Kontrollera variabelinställningar	91
Om systemet hänger sig	92
Sammanställning av grundläggande vi-kommandon	92
7. Elektronisk post	97
Grunderna i mailx	97
Starta mailx	98
Skicka ett provbrev till dig själv	98
Läsa ditt provbrev	100
Avsluta mailx	101
Läsa brev	102
Ta bort och återfå borttagna brev	104
Skriva ut brev	105
Sända brev	105
Brev som inte kan sändas	107
Avbryta ett osänt brev	107
Lägga till kopior och dolda kopior	108
Infoga en kopia av ett brev eller en fil	109
Svara på ett brev	110
Spara och återfå brev	110
Spara brev i och kopiera brev till filer	111
Spara brev i och kopiera brev till mappar	111

Läsa brev i filer och mappar	113
Använda vi tillsammans med mailx	114
Aliasnamn	115
Definiera aliasnamn i .mailrc	115
Definiera aliasnamn i /etc/aliases	117
Kommandon med tecknet ~	121
Om du behöver hjälp: andra mailx-kommandon	122
8. Skriva ut	123
Begära en utskrift.	123
Skriva ut på standardskrivaren	124
Skriva ut genom att ange skrivarnamn.	124
Begära meddelande om att utskriften är färdig.	125
Skriva ut flera exemplar	126
Översiktlig tabell över tillägg till kommandot lp	126
Kontrollera skrivaren	127
Kontrollera dina skrivare	127
Ta reda på vilka skrivare som är tillgängliga.	128
Visa all statusinformation.	128
Visa skrivarnas status	129
Visa skrivares inställningar	130
Översiktlig tabell över tillägg till kommandot lpstat	131
Avbryta en utskrift.	132
Avbryta en utskrift genom att ange id-nummer	132
Avbryta en utskrift genom att ange skrivarnamn	132

9. Arbeta i nätverk	135
Viktiga begrepp	136
Logga in på en annan arbetsstation (rlogin).....	136
Logga in på en annan arbetsstation utan att ha en hemkatalog.....	137
Använda rlogin för att logga in med någon annans användarnamn.....	138
Använda rlogin för att logga in på en arbetsstation vars namn du inte känner till.....	138
Logga ut från en arbetsstation som anslutits med rlogin..	139
Göra en paus i en rlogin-förbindelse.....	140
Kontrollera vilken arbetsstation du arbetar med (whoami)	140
Kopiera filer mellan olika arbetsstationer (rcp)	141
Kopiera från en annan arbetsstation till din egen	141
Kopiera från din egen arbetsstation till en annan	142
Ge kommandon på andra arbetsstationer (rsh).....	142
Kontrollera användarinformation (rusers).....	143
10. Anpassa din arbetsmiljö	145
Startfiler.....	145
Omgivningsvariabler.....	146
Användarens inställningsfiler	147
Ställa in variabeln PATH	149
Aliasnamn (gäller bara C-skalet).....	150
Ändra kommandoledtexten	151
Andra användbara variabler	152

Ställa in standardbehörighet till filer och kataloger	153
Välja teckensnitt för OpenWindows	154
Välja teckensnitt och storlek.	155
Ta fram en lista över tillgängliga teckensnitt.	158
A. Byta till OpenWindows 3.3 eller senare version.	159
<code>SPARC</code> Byta till OpenWindows 3.3 från SunView	159
<code>SPARC</code> Filerna .defaults och .Xdefaults	159
Byta till OpenWindows 3.3 från tidigare versioner.	160
Omgivningsvariabeln OPENWINHOME.	160
Filen .xinitrc	161
Använda rätt startfil	161
Egenskaper för arbetsytan	163
Anpassa menyn Arbetsyta	163
B. Anpassa tangentbordet	165
Aktivera och stänga av tangenten Compose	165
<code>SPARC</code> Anpassa tangentbordet för vänsterhänta	166
<code>SPARC</code> Definiera om tangentbordet med en kommandofil	166
<code>SPARC</code> Byta tillbaka till det gamla tangentbordet	169
C. Använda program över nätverk	173
Använda rlogin för att använda ett program över nätverk	174
Mer om säkerhet	175
Vem ska läsa det här avsnittet?	175
Behörighetskontroll	175
Behörighetsprotokoll	176

Styra behörigheten till hanteraren	178
Använda klienter från någon annans arbetsstation eller som en annan användare på den egna arbetsstationen	182
D. SPARC-system och DNI (DECnet Interworking)	183
Inställningar för att kommunicera med DECnet	184
Visa en fjärransluten klient på en arbetsstation med OpenWindows	185
Visa en fjärransluten klient på en VAX-dator	185
E. Installera modem och skrivare	189
Installera modem i systemet	189
Mer information om modem	191
Installera en lokal skrivare i systemet	191
Mer information om skrivare	192
Sakregister	193

Om den här boken

Vem ska läsa den här boken?

Den här boken är skriven för användare av systemprogrammet Solaris™. Solaris består av SunOS™ och OpenWindows™.

Innan du läser den här boken

Ditt system bör vara installerat och klart att använda innan du läser den här boken. Om det inte är det, kan du läsa *SPARC: Installing Solaris Software, x86: Installing Solaris Software* eller installationshandboken till systemet.

Andra böcker från Sun

Solaris System Software Answerbook ger dig tillgång till flera böcker om programmet Solaris. Böckerna är uppdelade i följande paket:

- Sun Administrator's Set
Här beskrivs i detalj hur du installerar och underhåller olika systemkonfigurationer, bland annat större nätverk med Sun-arbetsstationer.
- Sun Developer's Set
Här kan programutvecklare läsa om hur de ska skriva, felsöka i och underhålla program.

-
- **Sun Reference Manual Set**
Här beskrivs alla SunOS-kommandon. Böckerna kallas för *man pages* och kan enkelt installeras som direkthjälp på skärmen.
 - **Sun User Set**
Här beskrivs olika funktioner i SunOS-systemet på ett mer detaljerat sätt, däribland hur du använder SunOS-kommandon, arbetar med OpenWindows, anpassar din arbetsmiljö, hanterar problem, skriver kommandofiler, hanterar elektronisk post samt använder nätverket.

Om du vill läsa mer om Solaris System Software Answerbook, se *Översikt över dokumentationen för Solaris*.

Typografiska konventioner

I den här boken används följande typografiska konventioner:

- *Kursiv stilvariant* används för variabler, boktitlar och argument till kommandon samt för att framhäva särskilt viktig text. Ett exempel:

Du måste skriva argumentet *filnamn* på det sätt som beskrivs i *man Pages(1): User Commands*.

- Teckensnittet **Courier** i fet snittvariant används i bildskärmsexemplen för text och tecken som du ska skriva på tangentbordet. Ett exempel:

```
$ date
```

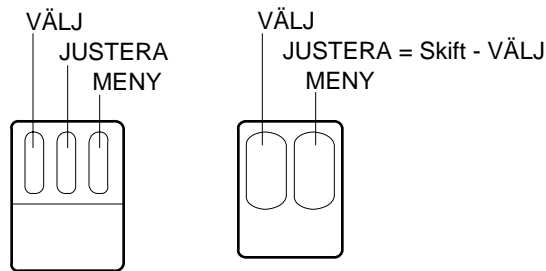
- Teckensnittet **Courier** används för innehållet i programfiler samt namn på kommandon, program och arbetsstationer. Ett exempel:

Du kan bli superanvändare genom att skriva `root` vid ledtexten för inloggning eller genom att skriva kommandot `su` vid kommandoledtexten.

Detta teckensnitt används också för text som visas på bildskärmen. Exempel:

```
$ who  
bozo      console      sep 11 15:36
```

- Namnen på musens knappar syftar på deras funktioner, inte deras placering. Till exempel står det "Klicka med VÄLJ" i stället för "Klicka med den vänstra musknappen".



Figur P-1 Musknapparnas standardfunktioner

På en mus med tre knappar är funktionerna vanligtvis

- VÄLJ = den vänstra musknappen
- JUSTERA = den mittersta musknappen
- MENY = den högra musknappen

På en mus med två knappar är funktionerna vanligtvis

- VÄLJ = den vänstra musknappen
- MENY = den högra musknappen

För att använda funktionen JUSTERA på en mus med två knappar klickar du med knappen VÄLJ samtidigt som du håller ner Skift-tangenten på tangentbordet.

- Ett ord mitt i en text som börjar med en versal visar att det handlar om en tangent som du ska trycka på. Ett exempel:

Tryck på Return.

Tangenten Return har präglingen Enter på vissa typer av tangentbord.

Observera – Solaris går att köra på både SPARC- och x86-baserade system. Eftersom systemen har olika tangentbord är också vissa snabbkommandon beroende av vilket tangentbord de utförs på.

De flesta snabbkommandona består av tangentkombinationer med

Metatangenten som går att åstadkomma på båda typer av tangentbord. I kapitel 1 "Logga in i SunOS och starta OpenWindows" finns en tabell över kommandon och motsvarande snabbkommandon.

- När texten handlar om en viss meny eller ett standardvalt menyalternativ är detta alternativ inringat. När du klickar med VÄLJ på en menyknapp utan att menyn visas väljs det standardvalda menyalternativet.
- Skärmbilderna i den här boken stämmer i allt väsentligt överens med systemet, men skillnader kan finnas. DeskSet-programmens versionsnummer visas till exempel inte i bilderna trots att de på skärmen visas i fönstrens namnlistor.
- Ett bindestreck (-) visar att du trycker ner två tangenter på samma gång, dvs att ett menykommando i OpenWindows utförs med hjälp av ett snabbkommando.

Ctrl-Alt betyder med andra ord att du samtidigt ska trycka ner och hålla nere både tangenten Control och Alt. Ctrl-Alt-c betyder att du ska trycka på och hålla nere tangenterna Control och Alt samtidigt som du trycker på c.

- Snabbkommandon utförs genom att man trycker på Meta-tangenten och på en eller flera andra tangenter. På tangentbord av SPARC-typ är Meta-tangenten märkt \blacklozenge . På tangentbord av typen X86 utför du Metatangentens funktion genom att samtidigt trycka på Ctrl och Alt (Ctrl-Alt). Snabbkommandon och funktionstangenter beskrivs i kapitel 1 "Logga in i SunOS och starta OpenWindows".
- Följande skrivsätt visar att man väljer ett alternativ på en meny: Meny \blacktriangleright Alternativ, till exempel: Välj Arbetsyta \blacktriangleright Program.

Konventioner gällande plattformar

Om följande symboler föregår ett avsnitt eller moment betyder det att texten endast behandlar den aktuella plattformen:

$\overline{\text{SPARC}}$ betyder att avsnittet eller momentet endast gäller SPARC-plattformar.

$\overline{\text{x86}}$ betyder att avsnittet eller momentet endast gäller x86-plattformar.

En anmärkning som berör en specifik plattform utmärks på följande vis:

SPARC – Denna anmärkning gäller endast SPARC-plattformar.

x86 – Denna anmärkning gäller endast x86-plattformar.

Vid punktning indikeras plattformsspecifik information på följande vis:

- **SPARC**: Denna punkt behandlar endast SPARC-plattformar.
- **x86**: Denna punkt behandlar endast x86-plattformar.

Om ett urval av programkod, en tabell eller en bild endast angår en specifik plattform, anges typen av plattform i början av titeln. Det kan t ex se ut så här:

Tabell 1-1 SPARC: Rubrik för SPARC-plattformar

Tabell 1-2 x86: Rubrik för x86-plattformar

Logga in i SunOS och starta OpenWindows



OpenWindows bör redan vara installerat på skivminnet eller på en hanterare (*server*) i filsystemet. Om du inte vet om så är fallet kan du tala med systemadministratören eller läsa *SPARC: Installing Solaris Software* eller *x86: Installing Solaris Software*.

Det här kapitlet beskriver hur du loggar in i systemet, hur du använder ett skalfönster, hur du monterar och startar OpenWindows, hur du avslutar systemet och hur du loggar ut. I kapitlet finns också viss specialinformation, till exempel hur du gör inställningar för dubbla bildskärmar.

Logga in

Ett *arbetspass* kan definieras som intervallet mellan inloggning och utloggning. Fleranvändarmiljön SunOS kräver att du identifierar dig varje gång du vill använda systemet. Ditt *namn för inloggning* (kallas även för *användarnamn* eller *användarkonto*) bekräftar din identitet för systemet och för andra användare. Ditt *lösenord* förhindrar att personer som känner till ditt användarnamn kan använda systemet. Om du inte redan har ett användarnamn och lösenord bör du kontakta *systemadministratören*. När du har den här informationen kan du logga in.

Innan du loggar in i systemet bör din skärm se ut ungefär så här:

Ange ditt användarnamn som du fått av systemadministratören och tryck på Return. Om ditt användarnamn är till exempel `roger`, skriver du

```
login: roger
```

och trycker på Return. Du blir nu ombedd att ange lösenord:

```
login: roger
Password:
```

Skriv ditt lösenord vid ledtexten och tryck på Return. (Om det inte krävs något lösenord för att logga in på systemet, kan du logga in utan att ange lösenord.) Observera att systemet inte visar det lösenord du har skrivit. Detta för att förhindra att någon annan ska kunna se ditt lösenord.

Om det inte krävs att du anger något lösenord men du ändå vill ha ett, eller om du vill ändra ett aktuellt lösenord, kan du läsa avsnittet "Använda lösenord" i kapitel 5 "Lösenord, processer och skivminnen".

Ditt inloggningskal

I de kapitel som följer kommer du att ge SunOS-kommandon. När du ger ett kommando till systemet, sänder du information till en kommandotolk, som kallas ett *skal*. Skalprogrammet läser informationen och utför den aktuella funktionen.

Standardskalet i SunOS-systemet är Bourne-skalet, men det finns även två andra skalprogram tillgängliga: C-skalet och Korn-skalet. Varje skal har sina speciella egenskaper.

Observera – Du kan få utförlig information om ett SunOS-kommando, inklusive varje tillgängligt skal, genom att titta i direkthjälpen (`man`) för det. Om du vill ha mer utförlig information om ett SunOS-kommando, inklusive information om direkthjälpen, kan du läsa "Ta fram direkthjälp om ett kommando med kommandot `man`" i kapitel 2 "Grundläggande SunOS-kommandon".

När du loggar in i systemet (eller öppnar ett nytt kommando- eller skalfönster) och du ser kommandoledtexten, indikerar den att ett skalprogram har startats automatiskt. Detta skal kallas *inloggningsskal*. Om ditt inloggningsskal inte är standardskalet för SunOS (Bourne-skalet), beror det på att ett annat skal (antingen C- eller Korn-skalet) har specificerats av systemadministratören.

Eftersom skalen är unika kanske vissa kommandon eller funktioner inte finns tillgängliga i en del skal. Om kommandon eller funktioner som inte finns tillgängliga i standardskalet för SunOS (Bourne-skalet) beskrivs i den här handboken, anges detta tydligt i avsnittet.

Logga ut

När du har avslutat ditt arbetspass och vill logga ut, skriver du följande:

```
$ exit
```

Efter en stund visas inloggningsledtexten:

```
$ exit  
login:
```

När inloggningsledtexten visas vet du att du har loggat ut. Du eller någon annan användare kan nu logga in i systemet.

Observera – När du använder operativsystemet SunOS räcker det inte med att stänga av arbetsstationen eller terminalen för att logga ut. Om du inte loggar ut kan du fortfarande vara inloggad i systemet trots att du har stängt av datorn.

Snabbkommandon

I stället för att utföra ett kommando genom att trycka på musen, välja från en meny eller använda fördefinierade tangenter kan du utföra kommandot genom att ange ett *snabbkommando*, dvs trycka på en eller flera tangenter. Snabbkommandona gör ofta att arbetet vid datorn går betydligt fortare.

I tabellen nedan finner du flera användbara kommandon samt de snabbkommandon som hör till dem. Snabbkommandona som redovisas i tabellen gäller både för SPARC- och x86-baserade system.

Observera – På tangentbord för SPARC-system är Meta-tangenten märkt <>. Tangentbord för x86-baserade system har inte motsvarande tangent. På dessa tangentbord trycker du i stället på Ctrl-Alt.

När du ska utföra ett snabbkommando börjar du med att trycka på och hålla nere den första tangenten (Meta alternativt de båda tangenterna Ctrl-Alt), sedan trycker du på den andra tangenten. Om du t ex har markerat ett stycke och vill ange snabbkommandot för Klipp ut gör du så här: har du ett tangentbord för SPARC-system trycker du först på och håller nere Meta-tangenten, sedan trycker du på x, har du ett tangentbord för x86-baserade system trycker du först på och håller nere de båda tangenterna Control och Alt, sedan trycker du på x.

Tabell 1-1 Snabbkommandon

Kommando	Snabbkommando	Funktion
Upprepa	Meta - a	Upprepar föregående moment
Kopiera	Meta - c	Kopierar det du har markerat till urklippsminnet
Klipp ut	Meta - x	Klipper ut det du har markerat och lägger det i urklippsminnet
Sök markerad text	Meta - f	Söker framåt i texten efter det du har markerat
Hjälp	Hjälp eller F1	Visar ett hjälpfönster som innehåller kontextrelaterad hjälp om det pekaren pekar på
Nytt	Meta - n	Skapar en ny fil

Tabell 1-1 Snabbkommandon (fortsättning)

Kommando	Snabbkommando	Funktion
Öppna (fil)	Meta - o	Öppnar markerad fil t ex en markerad fil i filhanteraren
Öppna (fönster)	Meta - w	Öppnar markerad ikon eller stänger ett fönster och omvandlar det till en ikon
Klistra in	Meta - v	Kopierar det som finns i urklippsminnet och klistrar in det på den plats markören befinner sig på
Skriv ut	Meta - p	Skickar filen till skrivaren, t ex en fil som du har markerat i filhanteraren
Egenskaper	Meta - i	Visar egenskapsfönstret för det program som pekaren pekar på
Gör om	Skift-Meta - p	Gör om det du har ångrat
Spara	Meta - s	Sparar aktuell fil
Avbryt	Avbryt eller Esc	Avbryter aktuellt moment
Ångra	Meta - u	Återgår till det läge som rädde innan du utförde det närmast föregående momentet

Att tänka på innan du startar OpenWindows

Innan du startar OpenWindows bör du observera punkterna nedan. Om punkterna inte är aktuella för dig kan du gå vidare till avsnittet "Starta OpenWindows" i det här kapitlet.

- Om du har ansvar för att installera en OpenWindows-hanterare i nätverket, bör du läsa *SPARC: Installing Solaris Software* eller *x86: Installing Solaris Software*.
- Om du använder OpenWindows Version 2 bör du läsa bilaga A "Byta till OpenWindows 3.3 eller senare version".
- Om du arbetar i SunView-miljön kan du läsa bilaga A "Byta till OpenWindows 3.3 eller senare version".

- Om du använder en tidigare version av OpenWindows än version 3.3, kan du läsa "Omgivningsvariabeln OPENWINHOME" och "Använda rätt startfil" i det här kapitlet.

x86 – Om du byter pekdon, tangentbord eller bildskärmskort i systemet efter att du installerat Solaris måste du köra programmet `devconfig` för att uppdatera systemkonfigurationen. I direkthjälpen (`man`) för `devconfig(1M)` kan du läsa mer om detta.

Omgivningsvariabeln OPENWINHOME

Om du använder en tidigare version av OpenWindows än version 3.3, kan ditt system vara inställt för att använda omgivningsvariabeln `OPENWINHOME`. Vi rekommenderar inte att du använder omgivningsvariabeln `OPENWINHOME`, varken manuellt eller från en startfil.

När du använder kommandot `openwin` ställs omgivningsvariabeln `OPENWINHOME` automatiskt in på `/usr/openwin`. Du behöver därmed inte göra denna inställning själv.

Om du har ställt in omgivningsvariabeln `OPENWINHOME` i filen `.profile` eller `.cshrc` i hemkatalogen, bör du kommentera bort inställningen eller ta bort den innan du startar OpenWindows version 3.3 eller senare.

Gör så här för att ta bort, eller kommentera bort, omgivningsvariabeln `OPENWINHOME` i filen `.profile` eller `.cshrc`:

1. Öppna filen `.profile` eller `.cshrc` med hjälp av ett redigeringsprogram, till exempel `vi`.

2. Skriv ett nummertecken (#) före variabeln (se nedan) eller radera hela raden.

Om du arbetar i filen `.profile` följer du steg a och om du arbetar i filen `.cshrc` följer du steg b.

a. I filen `.profile`:

```
#OPENWINHOME=/usr/openwin
```


b. I filen .cshrc:

```
#setenv OPENWINHOME /usr/openwin
```

3. Spara filen och avsluta redigeringsprogrammet.**4. Ta bort inställningen för omgivningsvariabeln OPENWINHOME (eller logga ut och logga in på nytt).**

Om du använder ett Bourne- eller Korn-skal följer du steg a och om du använder ett C-skal följer du steg b.

a. I Bourne- eller Korn-skalet skriver du:

```
$ unset OPENWINHOME
```

b. I C-skalet skriver du:

```
exempel% unsetenv OPENWINHOME
```

När du har tagit bort inställningen för omgivningsvariabeln kan du börja använda OpenWindows, som beskrivs i "Starta OpenWindows" i det här kapitlet.

Använda rätt startfil

Om du använder en tidigare version av OpenWindows än version 3.3, är det viktigt att bestämma statusen för filen `.xinitrc`. Filen `.xinitrc` är en startfil för OpenWindows i din hemkatalog och den kan innehålla användardefinierade funktioner.

Skriv följande kommandon om du vill bestämma status för filen `.xinitrc`:

```
$ cd  
$ ls -a .xinitrc
```

Beroende på resultatet av kommandot gör du något av följande:

- Om du inte har filen `.xinitrc` (filens namn visades inte när du gav kommandot `ls -a`) behöver du inte göra någonting. Om du inte har filen `.xinitrc` i din hemkatalog, använder OpenWindows standardstartfilen.
- Om du har filen `.xinitrc` (filens namn visades när du gav kommandot `ls -a`), men aldrig har gjort några ändringar i den eller inte vill behålla de ändringar du har gjort, följer du steg 1 i avsnittet "Så här hanterar du startfilen".
- Om du har filen `.xinitrc` (filens namn visades när du gav kommandot `ls -a`), och du har gjort ändringar som du vill spara, följer du steg 2 i avsnittet "Så här hanterar du startfilen".

Så här hanterar du startfilen

1. Skriv följande om du vill ta bort filen `.xinitrc` från din hemkatalog:

```
$ rm .xinitrc
```

2. Gör så här om du vill återfå de ändringar du har gjort i filen `.xinitrc`:

a. Flytta `.xinitrc` till `.xinitrc.save`:

```
$ mv .xinitrc .xinitrc.save
```

b. Kopiera `$OPENWINHOME/lib/Xinitrc` till `.xinitrc` i din hemkatalog:

```
$ cp $OPENWINHOME/lib/Xinitrc $HOME/.xinitrc
```

c. Lägg till de rader du vill behålla i `.xinitrc.save` till `.xinitrc`.



Varning - När du redigerar filen `.xinitrc` ska du inte lägga till en andra version av `olwm`, lägga till kommandot `svenv` eller ta bort raden som innehåller `$OPENWINHOME/lib/openwin-sys`.

Starta OpenWindows

Gör så här om du vill starta OpenWindows:

1. Montera OpenWindows med hjälp av NFS från den hanterare OpenWindows finns installerat på.

Om du vill veta hur du monterar OpenWindows från en hanterare kan du läsa *OpenWindows Reference Manual* eller fråga systemadministratören.

2. Starta OpenWindows med kommandot `openwin` och eventuella tillägg.

Starta OpenWindows

När du har monterat OpenWindows från en hanterare och använt OpenWindows installationsprogram kan du börja använda OpenWindows.

Om du vill starta OpenWindows skriver du `/usr/openwin/bin/openwin` vid skalledtexten och trycker på Return.

```
$ /usr/openwin/bin/openwin
```

OpenWindows visas nu på skärmen och du kan börja arbeta i OpenWindows-miljön.

Starta OpenWindows snabbt

När du har startat OpenWindows kan du definiera ett kortkommando för att starta programmet. Då behöver du inte ange den kompletta sökvägen varje gång du startar OpenWindows.

Om du använder ett Bourne- eller Korn-skal definierar du ett kortkommando genom att placera en skalfunktion i filen `.profile`. Om du använder C-skalet skapar du ett *alias* i filen `.cshrc`. Filerna `.profile` och `.cshrc` finns i din hemkatalog.

När du har definierat ett kortkommando i den aktuella filen för skalet startar du OpenWindows genom att skriva:

```
$ openwin
```

Hur du lägger in kortkommandot för att starta OpenWindows i startfilen beskrivs i avsnitten "I filen .profile" och "I filen .cshrc".

I filen .profile

Gör så här om du vill lägga in ett kortkommando i filen .profile:

- 1. Öppna filen .profile i ett redigeringsprogram, till exempel vi.**
- 2. Ange följande skalfunktion, exakt som i exemplet nedan, i filen:**

```
openwin () {  
    /usr/openwin/bin/openwin  
}
```

- 3. Spara filen och avsluta redigeringsprogrammet.**

- 4. Logga ut och logga in på nytt för att aktivera kortkommandot, eller skriv:**

```
$ . .profile
```

Nu behöver du bara skriva openwin när du vill starta OpenWindows.

I filen .cshrc

Gör så här om du vill lägga in ett kortkommando i filen .cshrc:

- 1. Öppna filen .cshrc i ett redigeringsprogram, till exempel vi.**
- 2. Ange följande aliaskommando, exakt som i exemplet nedan, i filen:**

```
alias openwin /usr/openwin/bin/openwin
```

- 3. Spara filen och avsluta redigeringsprogrammet.**

- 4. Logga ut och logga in på nytt för att aktivera kortkommandot, eller skriv:**

```
example% source .cshrc
```

Nu behöver du bara skriva openwin när du vill starta OpenWindows.

Om OpenWindows inte startar

Som standard installeras OpenWindows i katalogen `/usr/openwin`. Därifrån startas också OpenWindows. Många program, till exempel kalendern, kan inte installeras om inte OpenWindows är korrekt installerat i `/usr/openwin`.

Om OpenWindows inte startar när du skriver kommandot `/usr/openwin/bin/openwin`, är OpenWindows inte installerat eller installerat i en annan katalog än `/usr/openwin` (se *OpenWindows Reference Manual* eller kontakta systemadministratören).

Avsluta OpenWindows

När du arbetar med OpenWindows kan du inte logga ut som du gör från en vanlig SunOS-kommandorads-session. Du måste först avsluta fönstermiljön innan du loggar ut.

Om du skriver `logout` vid skalledtexten visas följande meddelande:

Detta är inte ett inloggningsskal.

Gör så här om du vill avsluta OpenWindows:

- 1. Placera musen så att pilen (*pekaren*) befinner sig på skärmens bakgrund (*arbetsytan*).**
- 2. Håll ned den högra musknappen, MENY.**
Menyn Arbetsyta visas.
- 3. Dra pekaren nedför menyn tills alternativet Avsluta markeras.**
- 4. Släpp musknappen.**
Ett snabbfönster visas där du måste bekräfta att du vill avsluta OpenWindows.
- 5. Placera pekaren på Avsluta och klicka med VÄLJ.**
Efter ett tag försvinner alla fönster från skärmen och ledtexten visas i det nedre vänstra hörnet av skärmen.

Starta OpenWindows med tillägg

De flesta användare kan starta OpenWindows genom att följa de steg som beskrivs i "Starta OpenWindows". I vissa fall kan du behöva använda kommandoradstillägg när du startar OpenWindows.

Det här avsnittet beskriver hur du använder tillägg när du vill starta OpenWindows

- med reducerad nätverkssäkerhet
- med olika typer av bildskärmar
- med flera bildskärmar

Om du vill starta OpenWindows med något tillägg använder du kommandot `openwin` på följande sätt:

```
$ openwin [ tillägg ]
```

Följande avsnitt beskriver några av de vanligaste tilläggen.

Starta med reducerad nätverkssäkerhet

Om du arbetar i ett öppet nätverk *och du inte bryr dig om säkerheten i nätverket*, kan du använda tillägget `-noauth`. Om du gör det kan andra användare i nätverket starta program på din arbetsstation.

Om du startar OpenWindows på följande sätt kan du således inte längre specificera vilka användare som ska få tillgång till din fönsterhanterare:

```
$ openwin -noauth
```

Starta OpenWindows med olika typer av bildskärmar

Om du har en bildskärm som endast kan visa gräskalor (det vill säga en monokrom bildskärm med ett bildskärmsminne som hanterar 8-bitsinformation) kan du använda tillägget `grayvis` när du startar OpenWindows. Du behöver inte använda det här tillägget, men OpenWindows ser bättre ut på din bildskärm om du gör det.

Om du vill använda det här alternativet skriver du följande vid systemledtexten:

```
$ cd  
$ openwin -dev /dev/fb grayvis
```

SPARC *Starta Open Windows på flera bildskärmar*

SPARC – Följande avsnitt gäller bara för SPARC-system. För att du ska kunna följa instruktionerna nedan krävs att du har viss vana vid systemadministration. Tala med systemadministratören om du behöver hjälp.

Om du vill starta OpenWindows på flera bildskärmar samtidigt måste du ange vilken utrustning du kommer att använda. Det kan du göra genom att antingen använda tillägget `-dev` eller standardinställningarna i den kommandofil (`openwin`) som startar OpenWindows.

Två tillägg måste alltid anges om du vill starta OpenWindows på dubbla bildskärmar:

```
$ openwin [ [ -dev enhet ] [ tillägg för enheten ] ]
```

De yttre hakparenteserna anger att det som står mellan dem, [`-dev enhet`] [`tillägg för enheten`], kan anges flera gånger (en gång per enhet).

[`-dev enhet`]

Med tillägget `-dev` anger du vilket bildskärmsminne som ska användas.

Om du inte anger detta tillägg används standardinställningen `/dev/fb`. Om du har anslutit flera bildskärmar till en hanterare anger du tillvalet `-dev` flera gånger på kommandoraden.

[`tillägg för enheten`]

Med `tillägg för enheten` kan du ändra funktionen hos den enhet som anges med tillägget `-dev`.

Exempel

Det här avsnittet visar exempel på hur du kan starta OpenWindows och använda flera bildskärmar.

Observera – I dessa exempel är ordningen i vilken du anger enheterna viktig. Den första enhet som anges måste vara placerad till vänster om eller ovanför den andra enheten. Den andra enheten måste vara placerad till höger om eller under den första enheten.

[left] [right]

Följande kommando anger att två bildskärmar ska användas. Den vänstra skärmen är standardskärmen och den högra en monokrom skärm. När OpenWindows startas kan du flytta pekaren i sidled mellan de båda bildskärmarna.

```
$ openwin -dev /dev/fb left -dev /dev/fbs/bwtwo0 right
```

Följande kommando ger samma resultat som det förra exemplet. Om du inte anger annat antar systemet att den första enhet du anger står till vänster.

```
$ openwin -dev /dev/fb -dev /dev/fbs/bwtwo0
```

Följande kommando anger att två bildskärmar ska användas. Den högra skärmen är standardskärmen och den vänstra en monokrom skärm. När OpenWindows startas kan du flytta pekaren i sidled mellan de båda bildskärmarna.

```
$ openwin -dev /dev/fb right -dev /dev/fbs/bwtwo0 left
```


[top] [bottom]

Följande kommando anger att två bildskärmar ska användas. Den övre skärmen är av typen CG6 och den nedre en monokrom skärm. När OpenWindows startas kan du flytta pekaren i höjdled mellan de båda bildskärmarna.

```
$ openwin -dev /dev/fbs/cgsix0 top -dev /dev/fbs/bwtwo0 bottom
```

Följande kommando ger *inte* samma resultat som det förra exemplet. Om du inte anger annat antar systemet att den första enhet du anger står till *vänster*.

```
$ openwin -dev /dev/fbs/cgsix0 -dev /dev/fbs/bwtwo0
```

Följande kommando anger att två bildskärmar ska användas. Den nedre skärmen är av typen CG6 och den övre en monokrom skärm. När OpenWindows startas kan du flytta pekaren i höjdled mellan de båda skärmarna.

```
$ openwin -dev /dev/fbs/cgsix0 bottom -dev /dev/fbs/bwtwo0 top
```

Kommentarer

Tänk på följande när du använder flera bildskärmar:

- standardinställningen är att `o1wm` hanterar alla bildskärmar
- du kan inte flytta fönster mellan bildskärmar

Grundläggande SunOS- kommandon



Det här kapitlet ger en introduktion till hur du använder kommandon i operativsystemet SunOS. Kapitlet beskriver bland annat hur du ger kommandon, hur du korrigerar skrivfel, hur du skriver långa kommandon eller flera kommandon efter varandra och hur du använder tillägg i kommandon.

När du ska ge kommandon använder du ett kommandofönster eller ett skalfönster. Du tar fram dessa fönster via undermenyn Program på menyn Arbetsyta.

Kommandoledtexten

När du öppnar ett kommando- eller skalfönster eller startar arbetsstationen utan att starta OpenWindows visas kommandoledtexten för SunOS. Hur denna ledtext ser ut beror på vilket skal du använder och vilka inställningar systemadministratören gjort. Eftersom standardledtexten för SunOS är ett dollartecken (§) används denna ledtext i exemplen i den här handboken.

Om du senare bestämmer dig för att ändra ledtexten kan du läsa om hur du gör det i avsnittet "Ändra kommandoledtexten" i kapitel 10 "Anpassa din arbetsmiljö".

Ge kommandon

När kommandoledtexten visas betyder det att systemet väntar på att du ska ge ett kommando. Prova att skriva kommandot `date` vid ledtexten och tryck sedan på Return.

```
$ date
måndag, 15 mars 1994 kl 10:12:51 MET DST
$
```

Som du ser visas aktuellt datum och klockslag när du ger detta kommando. Prova nu att skriva samma kommando med inledande versal (stor bokstav).

```
$ Date
Date: Det går inte att hitta kommandot.
$
```

Som du ser tolkar systemet inte ett litet `d` på samma sätt som ett stort. Nästan alla kommandon i SunOS skrivs med gemener (små bokstäver).

Korrigera skrivfel

Kommandot du skriver bearbetas inte av systemet förrän du trycker på Return. Om du innan du trycker på Return upptäcker att du skrivit fel, kan du korrigera felet på två olika sätt:

- tryck på Del eller backstegstangenten (Back Space) för att radera de felaktiga tecknen
eller
- tryck på Ctrl-U för att radera alla tecken på raden och börja om från början (håll ned Ctrl och tryck på U)

Prova båda dessa metoder för att se hur de fungerar. (Funktionen hos tangenterna Del och backstegstangenten skiljer sig åt mellan olika tangentbord. Ctrl-U bör fungera på de flesta tangentbord.)

Skriva långa kommandon och flera kommandon på samma rad

Du kan skriva flera kommandon på samma rad genom att skriva ett semikolon (;) mellan kommandona. Följande exempel visar kommandona `date` och `logname` skrivna på samma rad.

```
$ date; logname
måndag, 15 mars 1994 kl 10:12:51 MET DST
roger
$
```

Som du ser visas först aktuellt datum och klockslag (från kommandot `date`) och sedan användarnamnet på den användare som loggat in i systemet (från kommandot `logname`).

Om du skriver ett mycket långt kommando kan du skriva ett omvänt snedstreck (\) och sedan fortsätta på nästa rad. Exempel:

```
$ date; \
logname
måndag, 15 mars 1994 kl 10:12:51 MET DST
roland
$
```

Även om kommandona ovan inte är särskilt långa visar exemplet hur du skriver ett kommando på flera rader. Denna funktion blir användbar längre fram i handboken, när du stöter på kommandon som inte ryms på en rad.

Observera – Om du använder skal- eller kommandofönstret i OpenWindows behöver du inte skriva ett \-tecken om kommandot inte ryms på en rad. Där bryts raderna nämligen automatiskt när du når radens slut och hela kommandot utförs när du trycker på Return.

Upprepa föregående kommandon

Observera – Den funktion för att upprepa kommandon som beskrivs i detta avsnitt kan bara användas om du använder C-skalet.

Ett snabbt sätt att upprepa det senaste kommandot är att skriva `!!` och trycka på Return. Systemet sparar nämligen alla kommandon du gett i en lista. Prova till exempel att ge kommandot `date` och sedan skriva `!!`. Så här ser det ut på skärmen:

```
exempel% !!
date
måndag, 15 mars 1994 kl 10:12:51 MET DST
exempel%
```

Du kan också upprepa ett tidigare kommando genom att skriva `!` följt av kommandots nummer i listan över kommandon. Du tar fram denna lista genom att skriva kommandot `history` och trycka på Return. Så här kan listan se ut:

```
exempel% history
1  pwd
2  clear
3  ls -l
4  cd /usr/home/worker
5  logname
6  date
7  history
```

Ett annat sätt att använda denna lista är att skriva `!` följt av ett negativt tal. Om du till exempel vill upprepa kommando nummer 5, det tredje nedifrån, i listan ovan (`logname`) skriver du så här:

```
exempel% !-3
logname
roland
exempel%
```

Du kan också skriva `!` följt av de första tecknen i ett tidigare kommando. Om du till exempel har gett kommandot `clear` (som tömmer skärmen) kan du upprepa detta kommando genom att skriva `!c1`. Kom ihåg att du måste ange tillräckligt många tecken för att kommandot ska vara unikt. Om du bara skriver en bokstav efter utropstecknet kommer det senaste kommandot som börjar med den bokstaven att upprepas.

Använda tillägg till kommandon

Många kommandon har tillägg som påverkar hur kommandot utförs. Om du till exempel ger kommandot `date` med tillägget `-u` visas Greenwich Mean Time i stället för lokal tid.

```
$ date -u
måndag, 15 mars 1994 kl 10:12:51 GMT
$
```

De flesta tillägg skrivs med ett tecken och inleds med ett bindestreck (-). Vissa kommandon har flera tillägg och andra har inga alls. Om du använder flera tillägg till ett kommando kan du antingen skriva dem separat (`-a -b`) eller tillsammans (`-ab`).

Omdirigera dataflöde

Om du inte speciellt anger något annat visas resultatet av ett kommando på bildskärmen. Med speciella tecken kan du *omdirigera* dataflödet så att resultatet av ett kommando sänds någon annanstans, till exempel till en fil. Följande exempel visar hur du kan använda tecknet för omdirigering (`>`).

```
$ date > exempel.fil
$
```

I exemplet ovan omdirigeras resultatet från kommandot `date` till en ny fil med namnet `exempel.fil`. Nästa exempel visar hur du tar fram innehållet i denna fil med kommandot `more`.

```
$ more exempel.fil
måndag, 15 mars 1994 kl 10:12:51 MET DST
$
```

Som du ser innehåller filen `exempel.fil` resultatet av kommandot `date`. (I kapitel 3 "Arbeta med filer och kataloger" kan du läsa mer om kommandot `more`.)

Ibland vill du styra resultatet av ett kommando direkt till ett annat kommando. När två eller flera kommandon kombineras på detta sätt kallas det *kommandokoppling*. Det tecken som används för denna typ av omdirigering är ett vertikalt streck (|).

I stället för att spara resultatet av ett kommando i en fil kanske du vill styra det till utskriftskommandot `lp` genom att använda ett |-tecken. Om du till exempel vill styra resultatet av kommandot `date` direkt till skrivaren ger du följande kommando:

```
$ date | lp
$
```

(I avsnittet "Skriva ut på standardskrivaren" i kapitel 8 "Skriva ut" kan du läsa om hur du använder kommandot `lp` för att skriva ut filer.)

Ovanstående exempel på omdirigering av kommandon är relativt enkla, men när du lär dig mer om avancerade SunOS-kommandon kommer du att upptäcka att omdirigering är mycket användbart.

Utföra kommandon i bakgrunden

Eftersom vissa kommandon tar lång tid kan det vara lämpligt att utföra dem i *bakgrunden*. Då kan du arbeta vidare och ge andra kommandon medan kommandot utförs. Om ett kommando inte utförs i bakgrunden visas inte kommandoledtexten förrän kommandot utförts.

Om du vill utföra ett kommando i bakgrunden skriver du ett &-tecken efter kommandot. Numret som visas är processens id-nummer.

```
$ stortjobb &
[1] 21414
$
```


Programmet `stortjobb` kommer att köras i bakgrunden och du kan fortsätta ge kommandon under tiden. När bakgrundsjobbet är klart visas ett meddelande som talar om detta nästa gång du ger ett kommando. Så här kan det se ut om du ger kommandot `date` när ett bakgrundsjobb är klart:

```
$ date
måndag, 15 mars 1994 kl 10:12:51 MET DST
[1] + Klart    stortjobb
$
```

Om du räknar med att logga ut innan bakgrundsjobbet är klart ska du använda kommandot `nohup` för att se till att jobbet utförs ändå (se exemplet nedan). Annars avslutas bakgrundsjobbet när du loggar ut.

```
$ nohup stortjobb &
[1] 21414
$
```

Få hjälp om SunOS-kommandon

Det här avsnittet beskriver olika metoder att ta fram hjälpinformation på skärmen. Med dessa metoder kan du ta fram referensinformation vid din arbetsstation eller terminal.

Observera – Förutom dessa metoder kan du använda hjälpfunktionerna i OpenWindows för att få hjälp.

Ta fram direkthjälp om ett kommando med kommandot `man`

Om du känner till namnet på ett kommando men inte är säker på hur det används eller vilken funktion det har, kan du använda kommandot `man`. Ge följande kommando om du vill ta fram direkthjälpen om kommandot `man`:

```
$ man man
```

Detta kommando tar fram den första sidan i direkthjälpen om man. Tryck på mellanslagstangenten om du vill se nästa sida eller på Q om du vill avbryta direkthjälpen och återgå till kommandoledtexten. Med kommandot `man` kan du ta reda på syntaxen för ett kommando och vilka tillägg som kan användas med kommandot. Dessutom finns ofta exempel som visar hur kommandot kan användas.

Ta fram en kort sammanfattning av ett kommando med kommandot `whatis`

Om du bara vill ha en kort beskrivning av hur ett kommando används kan du använda kommandot `whatis`.

```
$ whatis date
date (1)          -display or set the date
$
```

Lägg märke till numret som visas inom parenteser i exemplet ovan. Det numret anger i vilket avsnitt kommandot finns. Kommandona är indelade i olika kategorier beroende på funktion. De flesta kommandon en normal användare utnyttjar finns i avsnitt 1. I direkthjälpen för ett kommando visas kommandona i bokstavsordning inom varje grupp.

Sökning efter nyckelord med kommandot `apropos`

Om du vet vad du vill göra men inte vilket kommando du ska använda kan kommandot `apropos` vara till hjälp. Detta kommando tar fram en lista över alla kommandon vars korta sammanfattningar (se föregående avsnitt) innehåller det eller de nyckelord du anger. I vissa fall kan listan bli mycket lång, eftersom vissa nyckelord förekommer på flera ställen.

Prova följande kommandon för att se vilka resultat kommandot `apropos` kan ge:

- `apropos who`
- `apropos execute`
- `apropos apropos`

Om du anger ett nyckelord som resulterar i en mycket lång lista kan du trycka på Ctrl-C för att avbryta kommandot (håll ned Ctrl och tryck på C).

Arbeta med filer och kataloger



I stället för att använda menyerna och ikonerna i filhanteraren i OpenWindows kan du använda kommandoraden i SunOS för att hantera filer och kataloger. Genom att kombinera SunOS-kommandon med fil- och katalognamn kan du bland annat kopiera, flytta, byta namn på och ta bort filer och kataloger.

Det här kapitlet ger en introduktion till hur du arbetar med filer och kataloger genom att ge SunOS-kommandon. Det spelar ingen roll om du arbetar i ett skal- eller kommandofönster i OpenWindows eller om du loggat in från en terminal. För att du ska kunna dra full nytta av operativsystemet SunOS måste du förstå de begrepp och funktioner som beskrivs i det här kapitlet.

Viktiga begrepp

En *fil* är den grundläggande enheten i operativsystemet SunOS. Nästan allt betraktas som filer, bland annat följande:

- **Dokument** – till denna grupp hör textfiler, till exempel brev eller rapporter, källkod till program eller något annat du skriver och vill spara.
- **Kommandon** – de flesta kommandon är *körbara* filer, det vill säga filer som innehåller ett program som kan köras. Kommandot `date`, som beskrivs i det föregående kapitlet, startar ett program som tar fram aktuellt datum och klockslag.
- **Enheter** – terminaler, skrivare och skivenheter behandlas också som filer.
- **Kataloger** – en katalog är helt enkelt en fil som innehåller andra filer.

Följande avsnitt beskriver de kommandon du kan använda för att skapa, kopiera, flytta och ta bort filer samt ta fram listor över filer. Du får också lära dig att titta på innehållet i en fil och hur du tar reda på vilken typ av fil det är.

Använda kommandon för filhantering

I de olika avsnitten nedan beskrivs ett antal kommandon för filhantering. I varje avsnitt finns en förklaring av kommandot och ett exempel på hur kommandot kan användas. Prova exemplen när du läser om de olika kommandona. Då kommer du lättare ihåg kommandona och förstår hur de används.

Innan du börjar

Innan du börjar arbeta med filerna ska du se till att du är i din *hemkatalog*. Hemkatalogen är en katalog som systemadministratören skapat speciellt för dig när han eller hon skapade ditt konto. Om du arbetar i hemkatalogen i exemplen nedan är det ingen risk att du skapar, flyttar, kopierar eller (i värsta fall) tar bort filer i delar av filsystemet som inte ska förändras.

För att vara säker på att du är i din hemkatalog ska du ge kommandot `cd` utan några argument eller tillägg. Ge sedan kommandot `pwd` för att kontrollera var i filsystemet du befinner dig. Den katalog som visas är din hemkatalog.

```
$ cd
$ pwd
/export/home/användarnamn
```

I det här exemplet är hemkatalogen `/export/home/användarnamn`. *användarnamn* är namnet på den användare som äger hemkatalogen.

Skapa en testfil

Använd kommandot `touch` för att skapa en tom fil. Om det inte finns en fil med det namn du anger skapas en ny, tom fil. (Om det finns en fil med det namnet uppdateras tiden då filen senast användes.)

```
$ touch testfil
$
```

Kontrollera innehållet i en katalog (`ls`)

Kontrollera att filen skapats genom att ge kommandot `ls`:

```
$ ls testfil
testfil
```

Om du ger kommandot `ls` utan argument eller tillägg visas alla filer i den aktuella katalogen. Om du anger ett filnamn visas bara den angivna filen (om det finns en fil med det namnet).

Du kan läsa mer om kommandot `ls` i *man Pages(1): User Commands*.

Kopiera filer (`cp`)

Använd kommandot `cp` för att kopiera filen `testfil` till en fil med namnet `kopiefil`:

```
$ cp testfil kopiefil
$
```

Ta sedan fram en lista över filerna. Lagg märke till att båda filerna slutar med `fil`. Du kan använda *jokertecknet* asterisk (*) som ersättning för ett valfritt tecken eller en följd av tecken. Om du ger kommandot `ls *fil` visas både `testfil` och `kopiefil` (och alla andra filer i den aktuella katalogen vars namn slutar med `fil`).

```
$ ls *fil
kopiefil  testfil
```

Lägg märke till att `kopiefil` visas först i listan. Filerna visas i bokstavsordning. Versaler (stora bokstäver) och siffror sorteras före gemener (små bokstäver).

I *man Pages(1): User Commands* hittar du mer utförlig information om kommandot `cp`.

Flytta och byta namn på filer (mv)

Med kommandot `mv` kan du både flytta och byta namn på filer. I det här exemplet ska du använda `mv` för att byta namn på `testfil` till `tomfil`:

```
$ mv testfil tomfil
$
```

Ta sedan fram en lista över filerna för att kontrollera att allt gått bra:

```
$ ls *fil
kopiefil  tomfil
```

Som du ser har `testfil` ersatts av `tomfil`.

Du kan läsa mer om kommandot `mv` i *man Pages(1): User Commands*.

Ta bort filer (rm)

Använd slutligen kommandot `rm` för att ta bort filen `kopiefil` och kontrollera sedan resultatet med `ls`:

```
$ rm kopiefil
$ ls *fil
tomfil
```



Varning – När du tagit bort en fil med kommandot `rm` kan du inte få tillbaks den, såvida det inte finns en säkerhetskopia. Var därför mycket försiktig när du använder kommandot `rm`, särskilt om du använder jokertecknet `*`.

I *man Pages(1): User Commands* hittar du mer utförlig information om kommandot `rm`.

Titta på innehållet i filer (`more`, `cat`)

Med kommandot `more` kan du titta på innehållet i en fil. Skriv `more` följt av namnet på den fil vars innehåll du vill titta på. När du trycker på Return rullas filens innehåll upp på skärmen. Om filen inte ryms på en sida på skärmen visas följande meddelande:

```
--Mer--(nn%) [Tryck på mellanslag för att fortsätta och q för  
att avsluta.]
```

`nn` anger hur stor del (i procent) av filen som har visats.

Du kan också använda kommandot `cat` för att titta på innehållet i en fil, men då rullas hela filens innehåll förbi på skärmen utan att stanna efter varje sida. Kommandot `cat` används oftare för att foga samman två eller flera filer till en stor fil, som i exemplet nedan:

```
$ cat fil1 fil2 fil3 > storfil  
$ ls *fil  
storfil  
fil1  
fil2  
fil3  
$
```

Du kan läsa mer om kommandona `more` och `cat` i *man Pages(1): User Commands*.

Kontrollera filtyp (file)

Vissa typer av filer, till exempel binära eller körbara filer, kan inte skrivas ut på en skrivare eller visas på skärmen. Om du vill ta reda på av vilken typ en fil är kan du använda kommandot `file`.

```
$ file kopiefil
kopiefil:      ascii text
```

Kataloger och katalogstruktur

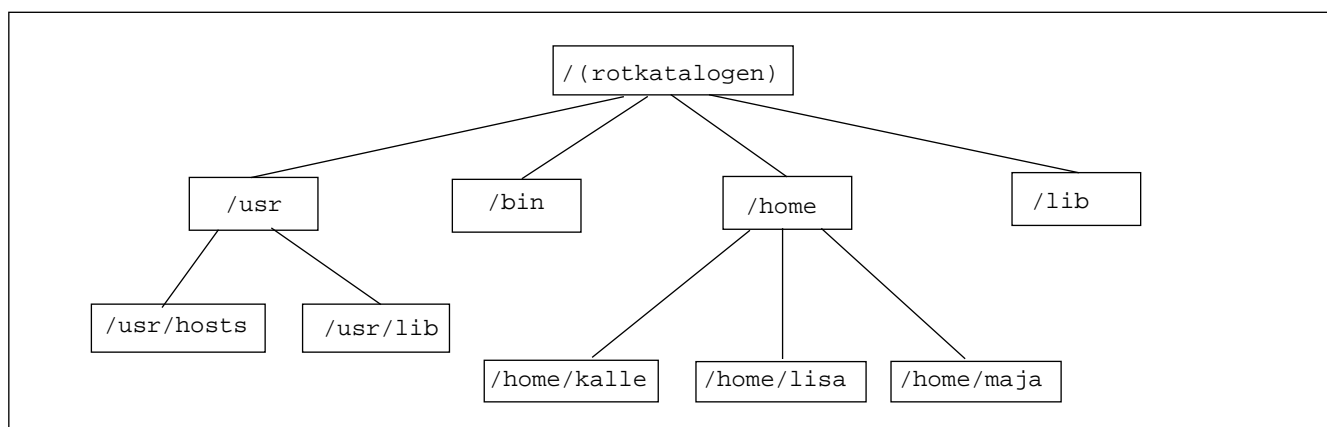
Nu har du lärt dig att ta fram listor över, kopiera, byta namn på och ta bort filer. Nu är det dags att ta upp ytterligare ett begrepp, nämligen *kataloger*. Det här avsnittet beskriver katalogstrukturen och hur du arbetar med kataloger. Läs instruktionerna noggrant och prova sedan kommandona i de olika exemplen.

Katalogstruktur

Filer samlas i olika kataloger som i sin tur är organiserade i ett hierarkiskt system: *filsystemet*. Överst i katalogstrukturen finns rotkatalogen, som symboliseras av `/`-tecknet.

Som du kan se i figur 3-1 kan varje katalog i filsystemet innehålla flera kataloger. Katalogerna skiljs åt med `/`-tecken. Rotkatalogen innehåller alltså bland annat underkatalogerna `/usr`, `/bin`, `/home` och `/lib`. Underkatalogen `/home` innehåller i sin tur katalogerna `kalle`, `lisa` och `maja`.

När du ska beskriva en katalog (och de filer den innehåller) anger du även namnen på de kataloger den ligger i. Detta kallas för en *sökväg*. Sökvägen till katalogen `maja` i bilden nedan är alltså `/home/maja`.



Figur 3-1 Hierarkin i filsystemet

Alla filnamn och namn på underkataloger i en katalog måste vara unika. Däremot kan underkataloger och filer som ligger i olika kataloger ha samma namn. I bilden ovan har katalogen /usr en underkatalog som heter /usr/lib. Dessutom finns det i rotkatalogen en katalog som heter /lib. Detta går bra eftersom katalogerna har olika sökvägar.

Sökvägar till filer fungerar på samma sätt som sökvägar till kataloger. Sökvägen till en fil anger var i katalogstrukturen den finns. Om det till exempel i katalogen /home/lisa finns en fil som heter rapport5, är sökvägen till den filen /home/lisa/rapport5. Detta anger att filen rapport5 ligger i katalogen lisa, som ligger i katalogen home, som i sin tur ligger i rotkatalogen (/).

Kataloger kan innehålla filer, underkataloger eller bådadera.

Kontrollera aktuell katalog (pwd)

Med kommandot `pwd` kan du ta reda på vilken katalog som är den aktuella, det vill säga var i filsystemet du befinner dig:

```

$ pwd
/home/kalle
  
```

Hur resultatet av kommandot ser ut beror på filsystemets struktur i ditt system. Kom ihåg att arbetskatalogen (den aktuella katalogen) är den katalog där du befinner dig.

Hemkatalogen

Alla användare i systemet har en *hemkatalog*. Första gången du öppnar ett kommando- eller skalfönster i OpenWindows är hemkatalogen arbetskatalog. Systemadministratören skapar din hemkatalog när han eller hon skapar ditt konto.

Byta arbetskatalog (cd)

Kommandot `cd` använder du för att gå till en ny arbetskatalog i filsystemet:

```
$ cd /usr/lib
$ pwd
/usr/lib
```

Om du ger kommandot `cd` utan några tillägg går du till hemkatalogen. Så här kan det se ut om katalogen `/home/kalle` är din hemkatalog:

```
$ cd
$ pwd
/home/kalle
```

I C-skalet kan du använda tecknet tilde (`~`) för att ange hemkatalogen. Om du använder detta skal kan du ge följande kommando för att komma till katalogen `musik` i hemkatalogen:

```
exempel% cd ~/musik
```

Du kan också använda detta tecken för att gå till en annan användares hemkatalog:

```
exempel% cd ~användarnamn
```

användarnamn är det namn en annan användare använder vid inloggning.

Observera – Om du använder Bourne-skalet kan du inte använda `~`-tecknet.

Om du använder Bourne-skalet kan systemadministratören ordna så att du kan skriva `$home` för att ange hemkatalogen. I så fall går du till katalogen `musik` i hemkatalogen med följande kommando:

```
$ $home/musik
```

Du kan också gå till en annan användares hemkatalog genom att ge följande kommando:

```
$ $home/användarnamn
```

användarnamn är det namn en annan användare använder vid inloggning.

Den katalog som ligger omedelbart ovanför en underkatalog kallas för en *överordnad katalog*. I föregående exempel är `/home` överordnad katalog till `/home/kalle`. En överordnad katalog representeras av två punkter (`..`) i en filförteckning. Om du vill gå till den överordnade katalogen kan du använda kommandot `cd ..`:

```
$ pwd
/home/kalle
$ cd ..
$ pwd
/home
```

Låt oss säga att den aktuella katalogen (arbetskatalogen) är `/home/kalle` och du vill arbeta med filer i katalogen `/home/lisa`. Då kan du ge följande kommandon:

```
$ pwd
/home/kalle
$ cd ../lisa
$ pwd
/home/lisa
```

`../lisa` anger att systemet ska leta efter `lisa` i den överordnade katalogen. Som du ser är detta enklare än att skriva hela sökvägen: `/home/lisa`.

Skapa kataloger (mkdir)

Om du vill skapa en ny katalog skriver du kommandot `mkdir` följt av namnet på den nya katalogen:

```
$ mkdir vegetarian
$ cd vegetarian
$ mkdir broccoli
$ cd broccoli
$ pwd
/home/lisa/vegetarian/broccoli
```

Relativa sökvägar

Den fullständiga sökvägen till en katalog inleds med ett `/`-tecken och beskriver sedan hela katalogstrukturen mellan filen (eller katalogen) och rotkatalogen. Du kan emellertid ofta använda kortare sökvägar som bara anger var en fil eller katalog ligger i förhållande till den aktuella arbetskatalogen.

Om du befinner dig i en överordnad katalog behöver du inte ange den fullständiga sökvägen när du ska gå till en underkatalog. Det räcker om du anger underkatalogens namn. I kommandot `cd vegetarian` i exemplet ovan används en relativ sökväg för att gå till katalogen `vegetarian`. Om den aktuella arbetskatalogen är `/home/lisa` är den fullständiga sökvägen till katalogen `/home/lisa/vegetarian`.

Prova att skapa flera underkataloger och flytta sedan mellan katalogerna i denna katalogstruktur. Använd både fullständiga och relativa sökvägar. Kontrollera hela tiden vilken katalog som är den aktuella med kommandot `pwd`.

Flytta och byta namn på kataloger

Kommandot `mv` används för att både flytta och byta namn på kataloger. När du byter namn på en katalog flyttas den till en katalog med ett nytt namn:

```
$ pwd
/home/lisa/vegetarian
$ ls
broccoli
$ mv broccoli gurka
$ ls
gurka
```

Du kan också använda `mv` för att flytta en katalog till en annan katalog:

```
$ pwd
/home/lisa/vegetarian
$ ls
gurka
$ mv gurka ../vegetarian2
$ ls ../vegetarian2
gurka
```

I detta exempel flyttades katalogen `gurka` från katalogen `vegetarian` till `vegetarian2` med kommandot `mv`.

Kopiera kataloger

Använd kommandot `cp -r` för att kopiera en katalog och de filer den innehåller:

```
$ cp -r vegetarian vegetarian3
$
```

Detta kommando kopierar alla filer och underkataloger i katalogen `vegetarian` till den nya katalogen `vegetarian3`. Detta kallas för en *rekursiv* kopiering, vilket anges med tillägget `-r`. Om du försöker kopiera en katalog utan detta tillägg visas ett felmeddelande.

Ta bort kataloger (`rmdir`)

Om du vill ta bort en tom katalog använder du kommandot `rmdir`:

```
$ rmdir vegetarian3
$
```

Om katalogen innehåller filer eller underkataloger kommer den inte att tas bort.

Om du vill ta bort en katalog och allt dess innehåll, inklusive eventuella underkataloger och filerna där, använder du kommandot `rm -r`:

```
$ rm -r vegetarian3
$
```



Varning – När du tagit bort en katalog med kommandot `rmdir` kan du inte få den tillbaka. Detsamma gäller om du tagit bort en katalog och dess innehåll med `rm -r`. Var därför försiktig när du tar bort kataloger.

Jämföra filer (`diff`)

Med kommandot `diff` kan du jämföra ASCII-filer för att se vilka skillnader som finns. Detta är användbart till exempel om flera personer arbetat med olika versioner av en fil. Följande kommando söker igenom filerna `vfil` och `hfil` rad för rad och letar efter skillnader:

```
$ diff vfil hfil
```

När kommandot hittar en eller flera rader som skiljer sig åt, kontrollerar det om skillnaden beror på att rader lagts till, tagits bort eller ändrats samt hur många rader som påverkas. Som resultat visas respektive radnummer i de båda filerna och relevant text på raderna.

Om skillnaden beror på att information lagts till visas en rad som ser ut så här:

```
v[,v] a h[,h]
```

v är ett radnummer i *vfil* och *h* är ett radnummer i *hfil*.

Om skillnaden beror på att information tagits bort visas ett *d* i stället för ett *a* och om skillnaden beror på att information ändrats visas ett *c*.

De relevanta textraderna från båda filerna visas efter informationen om radnummer. Text från *vfil* föregås av ett <-tecken och text från *hfil* föregås av ett >-tecken.

I följande exempel visas innehållet i två filer. Därefter visas resultatet av kommandot `diff`:

```

$ cat schema.v.5
Vecka 5

Dag: Klockslag: Bokning: Kommentar:

T 10:00 Möte hårdvarugruppen varannan vecka
O 13:30 Möte programgruppen
T 15:00 Möte dok.gruppen
F 13:00 Anställningsintervju
$ cat schema.v.7
Vecka 7

Dag: Klockslag: Bokning: Kommentar:

M 8:30 Personalmöte hela dagen
T 10:00 Möte hårdvarugruppen varannan vecka
O 13:30 Möte programgruppen
T 15:00 Möte dok.gruppen
$ diff schema.v.5 schema.v.7
1c1
< Vecka 5
---
> Vecka 7
4a5
> M 8:30 Personalmöte hela dagen
8d8
< F 13:00 Anställningsintervju

```

Om filerna som jämförs är identiska ger kommandot `diff` inget resultat.

Kommandot `diff` har ett flertal olika tillägg. Du kan läsa mer om kommandot `diff` i *man Pages(1): User Commands*.

Jämföra tre filer (`diff3`)

Om du har tre olika filer som du vill jämföra kan du använda kommandot `diff3`:

```
$ diff3 fil1 fil2 fil3
```

`diff3` jämför tre filer och visar information om skillnaderna. Följande tecken används för att markera skillnader:

==== alla tre filerna skiljer sig åt

====1 *fil1* är annorlunda

====2 *fil2* är annorlunda

====3 *fil3* är annorlunda

Använda `bdiff` för att jämföra stora filer

Om du ska jämföra mycket stora filer ska du använda `bdiff` i stället för `diff`. Båda kommandona fungerar på ungefär samma sätt:

```
$ bdiff vfil hfil
```

Använd `bdiff` i stället för `diff` om filerna du ska jämföra innehåller mer än 3500 rader.

Söka efter filer (`find`)

Med kommandot `find` kan du söka efter filer som svarar mot de villkor du anger. Du kan till exempel söka efter filer vars namn innehåller vissa tecken eller filer som ändrats under en viss tidsperiod.

Till skillnad från de flesta andra kommandonas tillägg består tilläggen till `find` av flera tecken och du måste alltid ange i vilken katalog sökningen ska börja. Kommandot har följande syntax:

```
$ find katalog tillägg
```

katalog är namnet på den katalog där sökningen ska börja och *tillägg* är de tillägg du kan ange med kommandot `find`.

Tilläggen använder du för att ställa upp sökvillkor. En fil måste uppfylla alla sökvillkor för att den ska hittas. Detta innebär att ju fler tillägg du anger, desto färre filer kommer att hittas. Med tillägget `-print` anger du att resultatet ska skrivas ut på en skrivare. (Längre fram får du lära dig hur du kan använda `find` för att utföra kommandon. I det fallet kanske du inte vill att de filer som hittas ska visas på skärmen.)

Med tillägget `-name filnamn` anger du att `find` ska söka efter filer med ett visst namn. Med *filnamn* avses den högra delen av en fils fullständiga sökväg. I sökvägen `/usr/lib/kalender` räknas `kalender` som *filnamn*.

Om du till exempel vill se vilka filer i den aktuella katalogen, representerad av en punkt (`.`), och dess underkataloger som har namn som slutar med `s` ger du följande kommando:

```
$ find . -name '*s' -print
./programs
./programs/graphics
./programs/graphics/gks
./src/gks
$
```

Du kan använda följande tillägg till `find`:

`-name filnamn`

Söker efter filer vars namn överensstämmer med *filnamn*. Skriv filnamnet inom enkla citationstecken om det innehåller ersättningsmönster, till exempel jokertecken.

`-user användar-id`

Söker efter filer som ägs av en viss användare. *användar-id* kan antingen vara ett användarnamn eller ett användar-id.

`-group grupp`

Söker efter filer som tillhör en viss grupp.

`-mtime n`

Söker efter filer som har ändrats inom de *n* senaste dagarna.

`-newer kontrollfil`

Söker efter filer som har ändrats senare än *kontrollfil*.

Du kan ange i vilken ordning tilläggen ska gälla genom att skriva dem inom bakvända snedstreck och parenteser (till exempel `\(tillägg \)`). Inom parenteserna kan du använda tillägget `-o` för att ange att det räcker att ett av villkoren uppfylls. Om du inte använder detta tillägg måste alla villkor inom parenteserna uppfyllas. Exempel:

```
$ find . \( -name AAA -o -name BBB \) -print
./AAA
./BBB
```

Om du vill söka efter filer som *inte* uppfyller ett villkor anger du ett bakvänt snedstreck och ett utropstecken före tillägget:

```
$ find . \!-name BBB -print
./AAA
```

Du kan också använda `find` för att koppla kommandon till de filer som hittas. Då använder du följande tillägg:

`-exec kommando '{ }' \;`

Du måste avsluta tillägget med ett bakvänt snedstreck och ett semikolon (`\;`). Klammrarna inom enkla citationstecken ersätts med namnen på de filer som hittas.

Du kan till exempel använda `find` för att automatiskt ta bort tillfälliga filer. Om du ger dessa filer enhetliga namn kan du använda `find` för att söka efter dem och sedan ta bort dem var de än ligger. Om du till exempel ger alla tillfälliga filer namnen `tillf` eller `temp` kan du ge följande kommando för att ta bort dem:

```
$ find . \( -name tillf -o -name temp \) -exec rm '{}' \;
```

Du kan läsa mer om kommandot `find` i *man Pages(1): User Commands*.

Behörighet till filer och kataloger

Observera – Läs detta avsnitt noggrant. I det dagliga arbetet har du stor nytta av att förstå hur behörighetssystemet fungerar.

Med inställningarna för behörighet kan du bestämma vem som ska få läsa eller ändra en viss fil eller katalog. Du kanske har textfiler som du vill låta andra personer läsa men inte ändra eller programfiler som du vill att andra personer ska kunna använda.

Det finns tre huvudsakliga behörighetstyper för filer och kataloger:

- **Läsa** (`r`) som innebär att du får se innehållet i eller kopiera en fil eller ta fram en lista över innehållet i en katalog.
- **Ändra** (`w`) som innebär att du får ändra, ta bort eller byta namn på en fil eller lägga till eller ta bort filer i en katalog.
- **Använda** (`x`) som innebär att du får köra en fil som ett program eller komma åt filer eller underkataloger i en katalog.

Du kan ange vilka personer som ska omfattas av behörigheten till att läsa, ändra eller använda en fil eller katalog:

- **Ägaren** – den som skapat filen eller katalogen.
- **Gruppen** – användare som tillhör samma grupp som ägaren (till exempel alla användare på en viss avdelning). Systemadministratören skapar och underhåller grupper.
- **Alla** – alla användare i nätverket.

Ta fram information om behörighet och status (ls -l)

Du har redan lärt dig använda kommandot `ls` för att ta fram en lista över filerna i en katalog. Kommandot `ls` har flera tillägg. Om du använder tillägget `-l` visas listan i långt format. Filerna och katalogerna visas i bokstavsordning. I figur 3-2 kan du se ett exempel på hur detta kommando kan användas.

```

$ pwd
/home/ocean/lisa
$ ls -l
total 8
drwxr-xr-x  2 lisa          1024 feb  9 14:22 katalog1
-rw-r--r--  1 lisa              0 feb 10 10:20 tomfil
-rw-r--r--  1 lisa       104357 feb  5 08:20 storfil
drwxr-xr-x  3 lisa          1024 feb 10 11:13 vegetarian2

```

Behörighet Länkar Ägare Storlek Datum Klockslag Fil- eller katalognamn

Figur 3-2 Ta fram information om behörighet och status

Det första tecknet på raden anger vilken typ av fil det är. Ett bindestreck (-) innebär att det är en vanlig fil och ett `d` att det är en katalog. Dessutom finns det andra tecken som anger speciella filtyper.

De följande nio tecknen anger behörigheten till filen eller katalogen. Dessa nio tecken består av tre grupper med tre tecken i varje. De första tre tecknen anger behörigheten för ägaren, nästföljande tre tecken anger behörigheten för gruppen och de tre sista tecknen anger behörigheten för alla användare. Behörigheten till filen `tomfil` är `rw-r--r--`, vilket innebär att alla kan läsa filen men bara ägaren kan ändra den. Ingen har behörighet att använda (köra) filen. Behörigheten till katalogen `vegetarian2` är `rw-r-xr-x`, vilket innebär att alla har behörighet att titta på innehållet i katalogen och att komma åt filer och underkataloger i den. Det är dock bara ägaren som får skapa eller ändra filer i den.

Förutom informationen om behörighet visas även följande information:

- antalet länkar till filen eller katalogen
- namnet på ägaren (`lisa` i exemplet ovan)
- storleken i antal byte (tecken)
- datum och klockslag då filen eller katalogen senast ändrades
- namnet på filen eller katalogen

Använd kommandot `cd` för att gå till hemkatalogen och ge sedan kommandot `ls -l`. (Resultatet skiljer sig givetvis från exemplet ovan.)

Skriv nu följande kommando:

```
$ ls -l katalognamn
```

katalognamn är namnet på en underkatalog i den aktuella katalogen eller sökvägen till en katalog i filsystemet. När du anger namnet på en katalog visas information om alla filer och underkataloger i den katalogen.

Ta fram information om dolda filer (ls -a)

Vissa filer visas normalt inte när du ger kommandot `ls`. Detta gäller filer vars namn börjar med en punkt, till exempel `.cshrc`, `.login` och `.profile`. Om du vill att dessa filer ska visas i listan ger du kommandot `ls -a`:

```
$ ls -a
.
..
.cshrc
.login
.profile
tomfil
```

Lägg märke till att filer vars namn börjar med en punkt visas före de andra filerna. I listan finns två specialfiler: `.` som är en referens till den aktuella katalogen och `..` som är en referens till den överordnade katalogen.

De flesta filer vars namn inleds med en punkt är systemfiler och ska inte ändras av en användare, men det finns vissa undantag.

Ändra behörighet (chmod)

Med kommandot `chmod` kan du ändra behörigheten till en fil eller katalog. Du måste vara ägare till en fil eller katalog eller logga in som superanvändare för att du ska kunna ändra filens eller katalogens behörighet. Syntaxen för kommandot `chmod` är följande:

```
chmod behörighet namn
```

behörighet är de nya inställningarna för behörighet och *namn* är namnet på filen eller katalogen.

Du kan ange behörigheten på flera olika sätt. Det här är ett av de lättaste:

1. Använd en eller flera bokstäver för att ange vilka användare behörigheten ska gälla:
 - `u` (bara *användaren*)
 - `g` (hela *gruppen*)
 - `o` (*andra* användare)
 - `a` (*alla* ovanstående kategorier)
2. Ange om behörigheten ska läggas till (+) eller tas bort (-).
3. Använd en eller flera bokstäver för att ange vilken behörighet användaren ska ha till filen eller katalogen:
 - `r` (*läsa*)
 - `w` (*ändra*)
 - `x` (*använda*)

I exemplet nedan har behörigheten till filen utökats så att alla användare i gruppen får rätt att ändra filen `gurka` (det vill säga *behörighet* är `g+w` och *namn* är `gurka`):

```
$ ls -l gurka
drwxr-xr-x 3 lisa          1024 feb 10 11:15 gurka
$ chmod g+w gurka
$ ls -l gurka
drwxrwxr-x 3 lisa          1024 feb 10 11:15 gurka
$
```

Som du kan se ändras bindestrecket (-) i den teckengrupp som visar gruppens behörighet till ett w.

Om du vill ändra behörigheten till katalogen `gurka` så att användare som inte ingår i din grupp varken kan titta på innehållet i katalogen eller komma åt dess underkataloger, ger du följande kommandon (det vill säga *behörighet* är `o-rx` och *namn* är `gurka`):

```
$ ls -l gurka
drwxrwxr-x 3 lisa          1024 feb 10 11:15 gurka
$ chmod o-rx gurka
$ ls -l gurka
drwxrwx--- 3 lisa          1024 feb 10 11:15 gurka
$
```

Nu ändras `r` (behörighet att läsa) och `x` (behörighet att använda) i den teckengrupp som visar andra användares behörighet till bindestreck (-).

När du skapar en ny fil ställs behörigheten till den in automatiskt av systemet.

Normalt är standardinställningen för nya filer följande:

```
-rw-r--r--
```

Standardinställningen för nya kataloger är följande:

```
drwxr-xr-x
```

Om du vill att ägaren till den nya filen `rova` (`lisa`) ska ha behörighet att köra filen ger du följande kommandon:

```
$ ls -l rova
-rw-r--r-- 3 lisa          1024 feb 10 12:27 rova
$ chmod u+x rova
$ ls -l rova
-rwxr--r-- 3 lisa          1024 feb 10 12:27 rova
$
```


Om du vill ändra behörigheten för alla användarkategorier på en gång använder du tillägget `a`. Om du vill att alla användare ska ha behörighet att köra den nya filen `sallad` ger du följande kommandon:

```
$ ls -l sallad
-rw-r--r-- 3 lisa          1024 feb 10 11:31 sallad
$ chmod a+x sallad
$ ls -l sallad
-rwxr-xr-x 3 lisa          1024 feb 10 11:31 sallad
$
```

Nu visas ett `x` i alla tre teckengrupperna.

Du kan också ändra behörigheten till flera filer eller kataloger på en gång. Då använder du jokertecknet `*`. Om du till exempel vill ändra behörigheten till alla filer i den aktuella katalogen (`vegetarian`) så att bara du har rätt att ändra dem, ger du följande kommandon:

```
$ pwd
/home/lisa/vegetarian
$ ls -l
-rwxrwxrwx 3 lisa          21032 feb 12 10:31 betor
-rwxrwxrwx 2 lisa           68 feb 10 11:09 majs
-rwxrwxrwx 3 lisa          12675 feb 08 09:31 sallad
-rwxrwxrwx 1 lisa          1024 feb 14 16:38 tomater
$ chmod go-w *
$ ls -l
-rwxr-xr-x 3 lisa          21032 feb 12 10:31 betor
-rwxr-xr-x 2 lisa           68 feb 10 11:09 majs
-rwxr-xr-x 3 lisa          12675 feb 08 09:31 sallad
-rwxr-xr-x 1 lisa          1024 feb 14 16:38 tomater
$
```

Kontrollera alltid först med kommandot `pwd` vilken katalog som är den aktuella. Kommandot `chmod` påverkar bara filerna i den aktuella katalogen.

Ställa in absolut behörighet

Hittills har du använt kommandot `chmod` för att ändra inställningarna för behörighet i förhållande till de aktuella inställningarna. Du kan emellertid också använda kommandot `chmod` för att ställa in *absolut* behörighet med hjälp av sifferkoder.

Om du använder `chmod` på detta sätt ska du använda följande syntax:

```
chmod sifferkod namn
```

sifferkod är en kod som anger behörigheten och *namn* är namnet på den fil eller katalog som du vill ändra behörigheten till.

Den fullständiga sifferkoden består av tre siffror. Den första siffran anger ägarens behörighet, den andra gruppens och den tredje behörigheten för övriga användare. Följande kommando ger ägaren och alla användare i gruppen behörighet att läsa, ändra och använda (köra) filen `sallad`. Övriga användare får bara behörighet att läsa filen.

```
$ chmod 771 sallad
```

Tabell 3-1 visar hur den angivna behörigheten till filen `sallad` representeras av sifferkoden `771`.

Tabell 3-1 Behörigheten till filen `sallad`

Behörighet	Ägare	Grupp	Andra
Läsa	4	4	0
Ändra	2	2	0
Använda	1	1	1
Totalt	7	7	1

Varje kolumn i tabell 3-1 representerar en av användarkategorierna ägare, grupp och andra användare. Om du vill ge någon kategori behörighet att läsa en fil lägger du till 4 i den kolumnen. Vill du ge någon behörighet att ändra en fil lägger du till 2 och vill du ge någon behörighet att använda (köra) en fil lägger du till 1. Summera sedan värdena i kolumnen. Resultatet är den sifferkod du ska ange i kommandot `chmod`.

Följande exempel visar hur du kan använda denna metod för att ändra behörigheten till en fil. Här används också kommandot `ls -l` för att kontrollera resultatet:

```
$ ls -l tomat
-rw-r--r-- 3 lisa          1024 feb 10 11:46 tomat
$ chmod 755 tomat
$ ls -l tomat
-rwxr-xr-x 3 lisa          1024 feb 10 11:48 tomat
$
```

Behörigheten till filen `tomat` ändras så att ägaren kan läsa, ändra och använda filen medan gruppmedlemmar och andra användare bara kan läsa och använda den. Tabell 3-2 visar hur sifferkoden är uppbyggd för att ställa in denna behörighet till filen `tomat`.

Tabell 3-2 Behörigheten till filen `tomat`

Behörighet	Ägare	Grupp	Andra
Läsa	4	4	4
Ändra	2	0	0
Använda	1	1	1
Totalt	7	5	5

Om du vill ändra behörigheten till filen `palsternacka` så att både ägaren, gruppen och alla andra användare har rätt att läsa, ändra och använda den, ger du följande kommandon:

```
$ ls -l palsternacka
-rw-r--r-- 3 lisa          1024 feb 10 11:51 palsternacka
$ chmod 777 palsternacka
$ ls -l palsternacka
-rwxrwxrwx 3 lisa          1024 feb 10 11:53 palsternacka
$
```

I tabell 3-3 kan du se hur denna sifferkod är uppbyggd.

Tabell 3-3 Behörighet till filen palsternacka

Behörighet	Ägare	Grupp	Andra
Läsa	4	4	4
Ändra	2	2	2
Använda	1	1	1
Totalt	7	7	7

Sifferkoden 777 motsvarar den högsta behörighetsnivån.

Du kan använda jokertecknet * för att ändra behörigheten till alla filer i den aktuella katalogen. Om du till exempel vill ändra behörigheten till alla filer i katalogen vegetarian så att du får läsa, ändra och använda filerna, gruppmedlemmarna får läsa och använda dem och övriga användare bara får använda dem, ger du följande kommandon:

```
$ pwd
/home/lisa/vegetarian
$ ls -l
-rwxrwxrwx 3 lisa          21032 feb 12 10:31 betor
-rwxrwxrwx 2 lisa           68 feb 10 11:09 majs
-rwxrwxrwx 3 lisa        12675 feb 08 09:31 sallad
-rwxrwxrwx 1 lisa         1024 feb 14 16:38 tomater
$ chmod 751 *
$ ls -l
-rwxr-x--x 3 lisa          21032 feb 12 10:31 betor
-rwxr-x--x 2 lisa           68 feb 10 11:09 majs
-rwxr-x--x 3 lisa        12675 feb 08 09:31 sallad
-rwxr-x--x 1 lisa         1024 feb 14 16:38 tomater
$
```

Kontrollera alltid först med kommandot `pwd` vilken katalog som är den aktuella. Kommandot `chmod` påverkar bara filerna i den aktuella katalogen. Kontrollera också med kommandot `ls -l` att du ställt in rätt behörighet. Du behöver inte känna till vilken behörighet filerna har innan du ändrar den.

Du kan läsa mer om kommandot `chmod` i *man Pages(1): User Commands*.

Söka i filer



Det här kapitlet beskriver hur du med nyckelord och strängar söker efter kataloger och filer med hjälp av SunOS-kommandot `grep`.

Söka efter mönster med `grep`

Om du vill söka efter en specifik teckensträng i en fil, använder du kommandot `grep`. Grundsyntaxen i kommandot `grep` ser ut så här:

```
$ grep sträng fil
```

sträng är det ord eller den fras du söker och *fil* är filen du letar i.

Observera – En *sträng* består av ett eller flera tecken; en enstaka bokstav kan vara en sträng, liksom ett ord eller en mening kan utgöra en sträng. Strängar kan inkludera mellanslag, skiljetecken och osynliga (kontroll-)tecken.

Om du till exempel söker efter Edgar Allan Poes telefonanknytning skriver du `grep`, hela eller en del av hans namn och den fil som innehåller informationen:

```
$ grep Poe anknytningar
Edgar Allan Poe      x72836
$
```

Observera att du kan få fler än ett svar som motsvarar din sökning:

```
$ grep Allan anknytningar
David Allan          x76438
Edgar Allan Poe     x72836
$ grep Al anknytningar
Louisa May Alcott   x74236
David Allan          x76438
Edgar Allan Poe     x72836
$
```

`grep` skiljer mellan versaler och gemener (stora respektive små bokstäver), vilket innebär att du måste anpassa din sökning efter namnet på det du söker:

```
$ grep allan anknytningar
$ grep Allan anknytningar
David Allan          x76438
Edgar Allan Poe     x72836
$
```

I det första exemplet hittades inga motsvarigheter till sökningen, eftersom det inte finns några svarsalternativ som börjar med ett gement a.

`grep` som ett filter

`grep` används ofta som ett filter tillsammans med andra kommandon. Det gör att du kan filtrera bort onödig information från resultatet av olika kommandon. Om du vill använda `grep` som ett filter måste du koppla resultatet till `grep`. Symbolen för en sådan programkoppling är `|`.

Följande exempel visar filer som slutar på `.ps` och som är skapade i maj:

```
$ ls -l *.ps | grep maj
```

Den första delen av kommandoraden ser ut så här:

```
ls -l *.ps
```

Den skapar följande lista med filer:

```
$ ls -l *.ps
-rw-r--r-- 1 elvis      7228 apr 22 15:07 change.ps
-rw-r--r-- 1 elvis      2356 maj 22 12:56 clock.ps
-rw-r--r-- 1 elvis      1567 jun 22 12:56 cmdtool.ps
-rw-r--r-- 1 elvis     10198 jun 22 15:07 command.ps
-rw-r--r-- 1 elvis      5644 maj 22 15:07 buttons.ps
$
```

Den andra delen av kommandoraden ser ut så här:

```
| grep maj
```

Den kopplar listan till `grep` och söker efter maj:

```
$ ls -l *.ps | grep maj
-rw-r--r-- 1 elvis      2356 maj 22 12:56 clock.ps
-rw-r--r-- 1 elvis      5644 maj 22 15:07 buttons.ps
$
```

grep med strängar som består av flera ord

Om du vill söka efter en sträng som består av flera ord, omger du strängen med enkla eller dubbla citationstecken:

```
$ grep "Louisa May" anknytningar
Louisa May Alcott      x74236
$
```

Med `grep` kan du söka efter en textsträng i grupper av filer. När en eller flera motsvarigheter finns i fler än en fil, skrivs filnamnet, följt av ett kolon, och den aktuella raden ut:

```
$ grep ar *
film:Humphrey Bogart
alaska:Alaska har den största ytan av alla stater i USA.
wilde:Böcker kan vara bra eller dåligt skrivna.
$
```

Söka efter rader utan textsträngar

Om du vill söka efter alla rader i en fil som *inte* innehåller en specifik sträng, använder du `grep` med alternativet `-v`. Följande exempel visar hur du söker efter alla rader som inte innehåller bokstaven `e` i användaren `medicis` hemkatalogfiler:

```
$ ls
actors    alaska    hinterland  tutors    wilde
$ grep -v e *
actors:Mon Mar 14 10:00 PST 1936
wilde:Allt är bra.
$
```

Mer om grep

Du kan också använda kommandot `grep` för sökning med hjälp av *söksträngar*. Söksträngar består av bokstäver och siffror, tillsammans med tecken som har speciell betydelse för `grep`. Dessa specialtecken, som kallas *metatecken*, betyder något specifikt för systemet och måste stå inom citationstecken eller föregås av ett omvänt snedstreck (`\`). Ett metatecken kan vara något av följande tecken: `&`, `!`, `.`, `*`, `$`, `?` eller `\`.

- En cirkumflex (`^`) anger början på raden. Följande kommando söker efter alla rader som börjar med bokstaven `b`:

```
$ grep '^b' list
```


- Ett dollar-tecken (\$) anger slutet på raden. Följande kommando söker efter alla rader som slutar med bokstaven b:

```
$ grep 'b$' list
```

Följande kommando söker efter alla rader i `list` som *bara* innehåller b.

```
$ grep '^b$' list
```

- En punkt (.) anger ett godtyckligt tecken. Följande kommando söker efter alla ord där "an" följs av en bokstav, till exempel "land," "hantera", "antilop" och "semesterplan" (mellanslag räknas också som ett tecken).

```
$ grep 'an.' list
```

- En asterisk (*) tolkas som noll, en eller flera förekomster av ett tecken. Detta innebär att när en asterisk följer efter ett tecken ignorerar `grep` det tecken som ligger direkt före asterisken och söker efter det tecken som ligger i positionen dessförinnan. När asterisken följer efter en söksträng, söker `grep` efter noll eller flera tecken som motsvarar söksträngen.

Låt säga att du vill söka efter alla ord som innehåller bokstäverna qu. Om du skriver följande kommando söker funktionen efter alla ord som innehåller "qu", men också efter ord som bara innehåller bokstaven q.

```
$ grep 'qu*' list
```

Om du vill söka efter alla ord som innehåller bokstaven n, skriver du:

```
$ grep 'nn*' list
```

Om du vill söka efter alla ord som innehåller bokstäverna nn, skriver du:

```
$ grep 'nn*' list
```

Om du vill kan du prova och se vad som händer om du inte gör på det här sättet.

- Om du vill söka efter noll eller fler förekomster av *något* tecken i `list`, skriver du:

```
$ grep .* list
```

Söka efter metatecken

Låt oss anta att du vill söka efter textrader som innehåller ett dollartecken (\$). Om dollartecknet föregås av ett omvänt snedstreck (\) vet `grep` att dollartecknets egentliga innebörd ska ignoreras. Detta gäller även för de andra metatecknen (& ! . * ? och \).

Följande kommando söker efter alla rader som börjar med en punkt:

```
$ grep ^\.
```

Kommandot är användbart när du söker efter formateringskommandona `nroff` eller `troff` (som börjar med en punkt).

Följande tabell, tabell 4-1, visar en lista över vanliga sökelement som du kan använda med `grep`.

Tabell 4-1 Sökelement vid sökning med `grep`

Tecken	Sökning
<code>^</code>	Början av en rad
<code>\$</code>	Slutet av en rad
<code>.</code>	Ett enskilda tecken
<code>[...]</code>	Vart och ett av de enskilda tecknen inom hakparentesen
<code>[^...]</code>	Ett tecken som inte finns inom hakparentesen
<code>*</code>	Ingen eller flera förekomster av det föregående tecknet eller söksträngen
<code>.*</code>	Ingen eller flera förekomster av ett enskilda tecken
<code>\</code>	Ignorerar specialbetydelsen av nästa tecken

Observera att dessa söktecken också kan användas vid sökning i redigeringsprogrammet `vi`.

Enkla eller dubbla citationstecken på kommandoraden

Citationstecken använder du för att markera att text ska tolkas som ett ord. Om du skriver följande kommer `grep` att söka efter alla filer som innehåller frasen "heja, grabbar":

```
$ grep "heja, grabbar" *
```

Du kan även använda enkla citationstecken (`'`) för att visa att en grupp av ord ska betraktas som en enhet. Enkla citationstecken signalerar att vissa tecken, till exempel `$`, ska tolkas bokstavligt. (Kommandot `history` tolkar alltid `!` bokstavligt, även inom citationstecken, om du inte placerar ett omvänt snedstreck framför det.) Det kan dock vara bra att använda ett omvänt snedstreck framför tecken som `&` `!` `$` `?` `.` `;` och `\` när du vill att de ska behandlas som vanliga typografiska tecken.

Om du skriver följande kommer alla rader i `list` att visas:

```
$ grep $ list
```

Om du skriver följande kommer du bara att se de rader som innehåller tecknet `$`:

```
$ grep '\$' list
```

Om du vill veta mer om kommandot `grep`, kan du läsa *man Pages(1): User Commands*.

Lösenord, processer och skivminnen



I SunOS finns ett stort antal kommandon som du kan använda för att utföra olika uppgifter i systemet från kommandoraden. Det här kapitlet beskriver hur du bestämmer lösenord, kontrollerar vilka processer som är igång, avbryter pågående processer och kontrollerar hur mycket utrymme på skivminnet som är ledigt.

Använda lösenord

För att förhindra att obehöriga får tillgång till systemet används i SunOS lösenord vid inloggning. Du bör ändra ditt lösenord flera gånger per år, så att ingen annan kan logga in på ditt konto. Om du misstänker att någon använt ditt konto utan tillåtelse bör du ändra lösenordet omedelbart.

När du väljer ett lösenord ska du tänka på följande:

- Välj ett lösenord som du kan komma ihåg utan att skriva upp det. Om det är så svårt att du inte kan komma ihåg det är det sämre än om det är lätt att gissa.
- Välj ett lösenord som består av minst sex tecken och innehåller minst en siffra.
- Använd inte ditt namn eller dina initialer eller namnet på någon i din familj.
- Använd inte namnet på ditt husdjur eller något som kan hänföras till ett fritidsintresse.
- Använd inte enbart versaler (stora bokstäver).

- Om du har flera konton ska du inte använda samma lösenord för kontona.
- Du kan använda vilka tecken som helst i lösenordet. Undvik emellertid vissa specialtecken, till exempel Ctrl-C, Ctrl-Z, Ctrl-U, Ctrl-S, Esc, Tab, # och @. Dessa tecken kan i vissa fall tolkas som signaler i stället för som text.

Byta lösenord

Om du vill byta lösenordet använder du kommandot `passwd`:

```
$ passwd
Byter lösenord för janneb på arbetare
Gammalt lösenord:
Nytt lösenord:
Skriv det nya lösenordet igen:
$
```

1. När ledtexten `Gammalt lösenord: visas skriver du det aktuella lösenordet.`
(Om det inte finns något lösenord till ditt konto visas inte ledtexten `Gammalt lösenord:`.) Lagg märke till att lösenordet inte visas på skärmen när du skriver det. Skälet till detta är att ingen annan ska kunna se ditt lösenord.
2. När ledtexten `Nytt lösenord: visas skriver du det nya lösenordet.`
Lösenordet visas inte heller denna gång.
3. När ledtexten `Skriv det nya lösenordet igen: visas skriver du det nya lösenordet en gång till.`
Detta är en kontroll av att du skrivit det lösenord du tänkt dig.

Om du inte skriver samma lösenord båda gångerna visas ledtexten `Tyvärr` och lösenordet ändras inte. Om detta inträffar upprepade gånger kan du kontakta systemadministratören och be honom eller henne att hjälpa dig att ändra lösenordet.

Observera – Lösenord med mindre än sex tecken är inte tillåtna. Det nya lösenordet måste ha minst tre tecken som skiljer sig från det gamla.

Tidsbegränsade lösenord

Genom att använda vissa tillägg till kommandot `passwd` kan man begränsa den tid ett lösenord får gälla. Med dessa tillägg kan systemadministratören ange hur länge du får ha ett lösenord innan du måste ändra det. Det går också att ange en kortaste tid som lösenordet ska gälla.

När tidsgränsen för lösenordet är nådd visas ett meddelande om att du måste ändra lösenord. Detta inträffar när du loggar in. Följande meddelande visas:

```
Ditt lösenord har gått ut. Välj ett nytt.
```

Sedan startas programmet `passwd` automatiskt och ledtexten `Nytt lösenord:` visas.

Om systemadministratören angett att du måste ha samma lösenord i minst två veckor och du försöker ändra lösenordet innan dess, visas följande meddelande:

```
Det har gått mindre än 2 veckor sedan senaste bytet.
```

Om du vill ta reda på om lösenordet är tidsbegränsat kan du använda tillägget `-d` till kommandot `passwd`:

```
$ passwd -d  
användarnamn 2-14-92 14 60
```

Först visas det datum då det nuvarande lösenordet skapades. Sedan visas i tur och ordning det minsta respektive största antal dagar lösenordet får gälla. (Denna information visas bara om tidsbegränsade lösenord används.)

Du kan läsa mer om tidsbegränsade lösenord och lösenord i allmänhet i *man Pages(1): User Commands*.

Processer och process-id

När ett kommando tolkats av systemet startas en *process* för kommandot. Varje process tilldelas ett unikt process-id (PID). Systemet använder process-id för att kontrollera de olika processernas status.

Kontrollera vilka kommandon som utförs (ps)

Med kommandot `ps` kan du kontrollera vilka processer du äger. (När du ger ett kommando startas en process som du står som ägare till.) Förutom process-id (PID) visas även från vilken terminal processen startades (TTY), hur mycket *processortid* som har använts (TID) och vilket *kommando* som utförs (KOMMANDO).

Om du använder tillägget `-l` visas ytterligare information om de pågående processerna, till exempel processernas status (S). Följande koder används för att visa status:

- `O` (On processor) Processen utförs i en processor.
- `S` (Sleeping) Processen väntar på att att en annan process ska ha slutförts.
- `R` (Runnable) Processen väntar i en kö på att andra processer ska bli klara.
- `I` (Idle) Processen håller på att skapas.
- `Z` (Zombie) Processen har avslutats och den överordnade processen väntar inte.
- `T` (Traced) Processen har stoppats av en signal eftersom den överordnade processen håller på att spåra den.
- `X` (SXBK) Processen väntar på att mer primärminne ska bli tillgängligt.

Tänk på att saker och ting kan ändras under tiden `ps` körs, eftersom kommandot bara visar processernas status i det ögonblick kommandot ges.

Du kan läsa mer om kommandot `ps` och de tillägg som kan användas i *man Pages(1): User Commands*.

Avbryta processer (`kill`)

Med kommandot `kill` kan du avbryta en pågående process. Detta är användbart till exempel om du av misstag gett ett kommando som tar lång tid att utföra.

Så här avbryter du en process:

1. Ge kommandot `ps` för att ta reda på processens id-nummer (PID).
2. Ge kommandot `kill` följt av id-numret.

Så här kan det se ut på skärmen:

```
$ ps
PID    TTY    TID    KOMMANDO
1291   co     0:12   -bin/csh (csh)
3250   p0     0:00   ps
1286   p1     0:05   -bin/csh (csh)
3248   p1     0:05   vi commands
$ kill 1291
[1]  Avslutad          -bin/csh/ (csh)
$
```

Ett snabbare sätt att ta reda på process-id för en process är att kombinera kommandona `ps` och `grep` enligt följande:

```
$ ps | grep kommando
```

kommando är namnet på den kommandoprocess du vill avbryta. (Du kan läsa mer om kommandot `grep` i kapitel 4 "Söka i filer".)

Om en process inte avbryts när du ger kommandot `kill` utan tillägg kan du prova att använda tillägget `-9` enligt följande:

```
$ kill -9 process-id
```

process-id är identifikationsnumret (PID) för den process du vill avbryta.

Hantera skivminnen

Eftersom utrymmet på skivminnet är begränsat är det bra att då och då kontrollera hur mycket ledigt utrymme som finns.

Kontrollera utrymme på skivminnen (`df -k`)

Med kommandot `df -k` tar du fram information om hur stor del som används av de skivminnen som monterats i systemet. Skriv följande kommando:

```
$ df -k
```

Då visas hur mycket utrymme som finns på de olika skivminnena, hur mycket utrymme som är ledigt och hur stor andel (i procent) av skivminnena som används.

Om över 90 procent av ett skivminne används bör du "städa" det genom att ta bort de filer som inte längre behövs. Detta kan du göra genom att antingen flytta dem till ett annat skivminne eller till ett band eller ta bort dem (använd kommandona `cp` respektive `rm`). Självfallet ska du bara ta bort filer som du själv äger.

Kontrollera utrymmet i en katalog (`du`)

Med kommandot `du` kan du ta fram information om hur många block om 512 byte (tecken) som en katalog och dess underkataloger innehåller.

Med kommandot `du` tar du fram en lista över hur mycket skivutrymme varje underkatalog tar upp. Om du vill ta fram en lista över underkataloger i ett filsystem använder du kommandot `cd` för att gå till den katalog där filsystemet är monterat och ge följande kopplade kommandon:

```
$ du | sort -r -n
```

I exemplet ovan användes kommandot `sort` med tilläggen `-r` (omvänd ordning) och `-n` (numerisk) för att sortera listan. På så sätt kan du lätt hitta stora kataloger. Med kommandot `ls -l` kan du sedan kontrollera storlek och datum för filerna i de olika katalogerna. Flytta gärna mycket stora filer och filer som inte längre behövs till andra lagringsmedia, till exempel magnetband.

Redigeringsprogrammet `vi`



`vi` är det redigeringsprogram som är standard i SunOS. Eftersom `vi` inte är fönsterbaserat kan du använda det på vilken terminal som helst för att redigera ett flertal olika typer av filer.

Du kan skriva och redigera text med `vi`, men det är inget ordbehandlingsprogram i vanlig mening. Det har inte konstruerats för att skapa och bearbeta formaterad text på samma sätt som ett vanligt ordbehandlingsprogram. Om du vill skapa snygga dokument måste du kombinera `vi` med ett formateringsprogram, till exempel `nroff`, `troff` eller `ditroff`. Du infogar då speciella koder i dokumenten, koder som dessa program sedan omvandlar till formateringsinstruktioner till skrivaren eller bildskärmen.

I `vi` finns ett stort antal kommandon som i vissa fall har funktioner som överlappar varandra. Detta kan till en början verka avskräckande, men snart upptäcker du att `vi` är ett mycket kraftfullt redigeringsprogram som du kommer att ha stor nytta av. Det här kapitlet beskriver de mest användbara kommandona i programmet.

Det finns också en speciell version av `vi` som du kan använda för att titta på innehållet i en fil. Detta program heter `view`. Om du öppnar en fil med `view`, kan du använda `vi`-kommandon, men du kan inte spara de ändringar du gör. Med detta program kan du eller andra läsa `vi`-filer utan att ändra dem oavsiktligt.

Starta vi

I de följande avsnitten får du lära dig att starta vi, skriva text i en fil, spara filen och avsluta vi. Du kommer också att skapa en övningsfil som du sedan kommer att använda i det här kapitlet.

Skapa en fil

Starta vi och ange att du ska redigera filen `rita` med följande kommando:

```
$ vi rita
```

Om det redan finns en fil med namnet `rita` öppnas den filen, annars skapas en ny fil (i det här exemplet bör filen vara ny).

Efter ett tag visas redigeringsbilden i vi:



Figur 6-1 Redigeringsbilden i vi

Markören visas längst upp till vänster på skärmen. Tomma rader markeras med ~-tecken.

Du kan också starta vi utan att ange något filnamn och sedan namnge filen när du avslutar programmet.

Statusraden

Den nedersta raden på skärmen kallas för *statusraden*. Där visas filens namn och antalet rader och tecken filen innehåller. När du skapar en ny fil visas även texten [Ny fil] på statusraden.

De två arbetslägena i vi

I vi kan du arbeta i två olika lägen: *inmatningsläget* och *kommandoläget*. I inmatningsläget skriver du text i en fil och i kommandoläget ger du kommandon för att utföra vi-funktioner. Standardläget för vi är kommandoläget.

Eftersom vi inte visar vilket läge du arbetar i kan det ibland vara svårt att veta om du arbetar i inmatningsläget eller kommandoläget. Om du bara lär dig några grundläggande regler kan du dock undvika de vanligaste problemen.

Standardläget när du öppnar en fil i vi är alltid kommandoläget. Innan du kan skriva text i filen måste du ange något av inmatningskommandona i vi, till exempel i för att infoga text på den plats där markören är placerad eller a för att infoga text efter markörens position. (Dessa och andra inmatningskommandon i vi beskrivs mer utförligt längre fram i kapitlet.)

Om du arbetar i inmatningsläget och vill gå tillbaka till kommandoläget trycker du på Esc. Om du inte vet vilket läge du arbetar i kan du alltid trycka på Esc för att gå till kommandoläget och sedan fortsätta därifrån. Om du trycker på Esc när du arbetar i kommandoläget hörs en ljudsignal och skärmen blinkar till.

Inmatningsläget

För att du ska kunna skriva text i exempelfilen *rita* ska du skriva vi-kommandot *i*. Då går du över från kommandoläget till inmatningsläget.

Skriv sedan några korta textrader och avsluta varje rad genom att trycka på Return. De tecken du skriver visas till vänster om markören och befintlig text flyttas åt höger. I det här läget kan du korrigera dina skrivfel genom att radera felaktiga tecken med backstegstangenten och sedan skriva om tecknen. Längre fram i det här kapitlet får du lära dig hur du redigerar den text du skrivit.

När du skrivit klart texten trycker du på Esc för att återgå till kommandoläget. Markören flyttas till det sista tecken du skrev. Då kan du ge nya vi-kommandon.

Om programmet uppträder konstigt kan du kontrollera att tangenten Caps Lock inte är nedtryckt. Är den det blir alla bokstäver du skriver versaler (stora bokstäver). På vissa tangentbord fungerar tangenten F1 på samma sätt som Caps Lock. Eftersom F1 är placerad nära Esc är det lätt hänt att man trycker på fel tangent.

Observera – Ibland kan det vara nödvändigt att rita om skärmbilden för att exempelvis ta bort ovidkommande systemmeddelanden. Du ritar om skärmbilden genom att gå in i kommandoläget och trycka på Ctrl-L. Detta motsvaras i OpenWindows av kommandot Rita om.

Kommandoläget

När du öppnar en fil i vi kommer du till kommandoläget, där du kan ge kommandon för en rad olika funktioner. De flesta vi-kommandon består av en eller två bokstäver och ett nummer. Det är skillnad om du använder stora eller små bokstäver, men funktionerna är ofta likartade. Du kan till exempel infoga ett tecken till höger om markören genom att skriva a. Om du i stället skriver A läggs texten till i slutet av raden.

Du behöver normalt inte trycka på Return för att utföra ett kommando. Detta gäller dock inte kommandon som inleds med kolon (:). Dessa måste avslutas med Return. När du skrivit ett sådant kommando medan du arbetar i kommandoläget visas de tecken du skriver efter kolonet längst ned på skärmen. (Ibland räknas detta som ett tredje arbetsläge i vi. I den här handboken talar vi dock bara om två arbetslägen.)

Kommandon som inleds med ett kolon är egentligen ex-kommandon. vi och ex är två olika gränssnitt till ett och samma redigeringsprogram. vi är ett fullskärmsorienterat gränssnitt och ex är ett radorienterat gränssnitt. Du kan använda alla ex-kommandon i vi. När du skriver ett kolon går du egentligen över till ex. På så sätt kan du ge en rad olika ex-kommandon för redigering utan att lämna vi. I avsnittet "Använda ex-kommandon" kan du läsa mer om ex.

Avsluta redigering

När du redigerar en fil med `vi` arbetar du egentligen med en kopia av filen. Denna kopia finns i ett tillfälligt minnesutrymme, en så kallad *buffert*. För att de ändringar du gjort i en fil inte ska gå förlorade måste du spara filen på skivminnet.

Detta har både fördelar och nackdelar. Å ena sidan innebär det att du kan återfå den ursprungliga filen genom att avsluta redigeringen utan att spara ändringarna. Å andra sidan kan du förlora dina ändringar om systemet hänger sig. (De som arbetar vid terminaler som är anslutna via telefonnätet är extra sårbara för oplanerade avbrott.)

Gör det till en vana att spara dina filer ofta, särskilt om du gör stora ändringar.



Varning – Även om det är möjligt att starta flera `vi`-sessioner och öppna samma fil i dem, är detta inte lämpligt. Ändringarna i den ena `vi`-sessionen kan då komma i konflikt med de i den andra sessionen.

Spara ändringar och avsluta `vi`

Det finns flera likartade kommandon för att spara innehållet i bufferten och avsluta `vi`. Du kan välja om du vill spara en fil, spara filen och avsluta `vi` eller avsluta `vi` utan att spara filen.

Spara

Du sparar en fil med följande kommando (tryck på Return när du skrivit kommandot):

```
:w
```

Spara och avsluta

Du sparar en fil och avslutar vi med följande kommando (tryck på Return när du skrivit kommandot):

```
:wq
```

Du kan också ge kommandot ZZ. Då behöver du inte skriva något kolon före eller trycka på Return efter kommandot.

Avsluta utan att spara

Om du vill avsluta vi utan att spara en fil ger du följande kommando (tryck på Return när du skrivit kommandot):

```
:q
```

Om du har gjort ändringar i en fil kan du inte avsluta vi med kommandot :q. Om du försöker göra det visas meddelandet Inget har skrivits sedan senaste ändringen (:quit! overrides).

Om du gjort ändringar i en fil och vill avsluta vi utan att spara dessa ändringar ger du följande kommando (tryck på Return när du skrivit kommandot):

```
:q!
```

Skriva ut filer på skrivaren

När du har avslutat vi kan du skriva ut en fil på skrivaren med följande kommando:

```
$ lp filnamn
```

filnamn är namnet på den fil du vill skriva ut. Med det här kommandot skriver du ut en fil på standardskrivaren. Filen skrivs ut utan formatering, rad för rad, precis som den ser ut på skärmen. I kapitel 8 "Skriva ut" kan du läsa mer om utskriftskommandon.

Grundläggande vi-kommandon

De följande avsnitten beskriver de viktigaste vi-kommandona. Du kan använda dessa kommandon för att:

- flytta runt i en fil
- infoga text
- ändra och ersätta text
- ångra ändringar
- ta bort text
- kopiera och flytta text
- upprepa kommandon

Förflyttning i filer

I de föregående avsnitten fick du lära dig att skapa, spara och skriva ut en fil samt avsluta vi. Det här avsnittet beskriver hur du flyttar mellan olika delar av en fil. Öppna övningsfilen och prova de kommandon som beskrivs i avsnittet.

Flytta markören

När du startar vi står markören längst upp till vänster på skärmen. I kommandoläget kan du använda ett flertal olika kommandon för att flytta markören. Du använder då vissa bokstavstangenter, piltangenterna, Return, backstegstangenten (eller Del) och mellanslagstangenten.

Observera – Det är viktigt att du skiljer mellan stora och små bokstäver när du skriver kommandon. Samma bokstäver kan användas för olika kommandon.

Flytta markören med piltangenterna

Om det finns piltangenter på ditt tangentbord kan du prova dem nu. Du bör kunna flytta markören uppåt, nedåt, till vänster och till höger med dessa tangenter. Lägg märke till att du bara kan flytta markören över befintlig text.

Om du använder vi på en fjärransluten terminal är det inte säkert att piltangenterna fungerar som de ska. Detta beror på terminalemulatorn. Om piltangenterna inte fungerar kan du använda följande tangenter som ersättning:

- tryck på h för att flytta markören åt vänster
- tryck på l för att flytta markören åt höger
- tryck på j för att flytta markören nedåt
- tryck på k för att flytta markören uppåt

Flytta markören ett ord i taget

Tryck på w för att flytta markören ett ord åt höger.

Tryck på b för att flytta markören ett ord åt vänster.

Tryck på W eller B för att flytta markören förbi nästa eller föregående punkt eller kommatecken till nästa eller föregående mellanslag.

Tryck på e för att flytta markören till det sista tecknet i det aktuella ordet.

Flytta markören till början eller slutet av raden

Tryck på ^ för att flytta markören till början av den aktuella raden.

Tryck på \$ för att flytta markören till slutet av den aktuella raden.

Flytta markören en rad nedåt

Tryck på Return för att flytta markören till början av nästa rad.

Flytta markören åt vänster

Tryck på backstegstangenten för att flytta markören ett tecken åt vänster.

Flytta markören åt höger

Tryck på mellanslagstangenten för att flytta markören ett tecken åt höger.

Flytta markören längst upp på skärmen

Tryck på H för att flytta markören längst upp på skärmen.

Flytta markören till mitten av skärmen

Tryck på M för att flytta markören till mitten av skärmen.

Flytta markören längst ned på skärmen

Tryck på L för att flytta markören längst ned på skärmen.

Bläddra och rulla

Om du flyttar markören nedåt när den är nederst på skärmen eller uppåt när den är överst på skärmen, kommer texten att rullas uppåt eller nedåt. Om filen är liten går det bra att göra på detta sätt för att se hela innehållet, men är filen lång är det opraktiskt och långsamt.

Du kan bläddra eller rulla framåt eller bakåt i en fil, en hel eller en halv skärmbild i taget. (Om du vill prova dessa kommandon måste du kanske lägga till ett antal rader i övningsfilen `rita`.)

Observera att det finns en viktig skillnad mellan begreppen bläddra och rulla. När du rullar texten flyttas markören uppåt eller nedåt *en rad i taget*. När du bläddrar flyttas markören *en skärmbild i taget*. Om du har en snabb arbetsstation är det inte säkert att du märker någon skillnad, men om du arbetar på en fjärransluten terminal eller om systemet av någon anledning arbetar långsammare än normalt blir skillnaden uppenbar.

Bläddra en sida framåt

Tryck på Ctrl-F för att bläddra en skärmsida framåt (nedåt). (Håll ned Ctrl och tryck på F.) Markören flyttas till det övre vänstra hörnet av nästa sida.

Bläddra en halv sida framåt

Tryck på Ctrl-D för att bläddra en halv skärmsida framåt.

Bläddra en sida bakåt

Tryck på Ctrl-B för att bläddra en skärmsida bakåt (uppåt).

Bläddra en halv sida bakåt

Tryck på Ctrl-U för att bläddra en halv skärmsida bakåt.

Infoga text

Det här avsnittet beskriver de mest användbara vi-kommandona för att infoga text. Lägg märke till att vissa av kommandona gör att du går över i inmatningsläget. För att du ska kunna ge dessa kommandon måste du arbeta i kommandoläget. Kom ihåg att trycka på Esc för att försäkra dig om att du arbetar i kommandoläget innan du ger dessa kommandon.

Infoga text med kommandona a och A

Skriv a för att infoga text *till höger om markören*. Prova att ställa markören någonstans på en rad och skriva a. Skriv sedan den text du vill infoga. Tryck på Esc när du är klar.

Skriv A för att infoga text *i slutet av en rad*. Prova att ställa markören någonstans på en rad och skriva A. Markören flyttas då till radens slut. Skriv sedan den text du vill infoga. Tryck på Esc när du är klar.

Infoga text med kommandona i och I

Skriv i i kommandoläget för att infoga text *till vänster om markören*.

Skriv I för att infoga text *i början av en rad*. (Markören flyttas till radens början.) Tryck på Esc när du är klar.

Infoga nya rader

Med nedanstående kommandon kan du infoga tomma rader ovanför eller nedanför markören.

Skriv o för att infoga en tom rad *nedanför* markören. Prova att skriva o följt av någon text. Du kan skriva flera rader med text om du vill. Tryck på Esc när du är klar.

Skriv O för att infoga en tom rad *ovanför* markören.

Ändra text

I `vi` finns det flera olika sätt att byta ut ett textavsnitt mot ett annat. De följande avsnitten beskriver dessa olika metoder.

Byta ut ett ord

Om du vill byta ut ett ord mot ett annat placerar du markören i början av det ord som ska bytas ut. Skriv sedan `cw` följt av det nya ordet. Tryck på Esc när du är klar.

Om du vill ändra en del av ett ord placerar du markören i ordet, till höger om den del som ska vara kvar. Skriv sedan `cw` följt av den nya texten och tryck på Esc.

Byta ut en rad

Om du vill byta ut en hel rad placerar du markören någonstans på raden och skriver `cc`. Raden försvinner och ersätts med en tom rad. Skriv sedan den nya texten och tryck på Esc när du är klar.

Byta ut en del av en rad

Om du vill byta ut en del av en rad placerar du markören till höger om den del som ska vara kvar och skriver `c`. Skriv sedan den nya texten och tryck på Esc när du är klar. Allting till höger om markören ersätts med den nya texten.

Byta ut ett tecken mot ett eller flera andra

Om du vill ersätta det tecken markören står under med ett eller flera andra tecken skriver du `s` följt av den nya texten. Tryck på Esc för att gå tillbaka till kommandoläget.

Byta ut ett tecken mot ett annat

Med kommandot `r` kan du byta ut det tecken som markören står på mot ett annat tecken. Placera markören på det tecken du vill byta ut och skriv `r` följt av ersättningstecknet. När ersättningen är klar går `vi` automatiskt tillbaka till kommandoläget utan att du behöver trycka på Esc.

Byta plats på tecken

Du kan enkelt byta plats på tecken som skrivits i fel ordning. Låt oss säga att du skrivit "coh" i stället för "och". Du rättar detta fel genom att placera markören på det första tecken som ska flyttas (i det här fallet c) och skriver xp. Bokstäverna c och o byter då plats och vi återgår till kommandoläget.

Kontrollera radbrytningar

Om du vill bryta en rad utan att påverka texten flyttar du markören till det ordmellanrum där du vill att raden ska brytas, skriver r och trycker på Return. Om du skriver r och trycker på Return när markören står på ett tecken ersätts detta tecken med radbrytningstecknet (Return).

Om du vill sammanfoga två rader placerar du markören på den övre raden och skriver ett stort J. Du behöver inte trycka på Esc.

Ångra ändringar

Om du gör en felaktig ändring kan du använda något av kommandona i vi för att ångra ändringen. Dessa kommandon beskrivs nedan.

Ångra föregående kommando

Om du ger fel kommando eller ändrar dig när kommandot utförts kan du ångra det senaste kommandot genom att trycka på u omedelbart efter kommandot. Du behöver inte trycka på Esc när du skrivit u. Om du trycker på u ytterligare en gång ångrar du ångra-kommandot.

Ångra alla ändringar på en rad

Skriv U om du vill ångra alla ändringar du gjort på en rad. Detta kommando fungerar bara så länge markören står kvar på raden. Du behöver inte trycka på Esc när du skrivit U.

Ta bort text

Med de kommandon som beskrivs nedan kan du ta bort enstaka tecken, ord eller rader. När kommandona utförts återgår du till kommandoläget, så om du då vill infoga text måste du ge ett kommando för att gå till inmatningsläget.

Ta bort ett tecken

Om du vill ta bort ett tecken placerar du markören på tecknet och skriver `x`.

När du tar bort ett tecken flyttas efterföljande tecken ett steg åt vänster och fyller upp tomrummet. Du kan också ta bort mellanslag med kommandot `x`.

Om du vill ta bort tecknet till vänster om markören skriver du ett stort `X`.

Ta bort ett ord eller en del av ett ord

Om du vill ta bort ett ord placerar du markören i början av ordet och skriver `dw`. Ordet tas bort och efterföljande tecken flyttas åt vänster.

Om du vill ta bort en del av ett ord placerar du markören på ordet, till höger om den del som ska vara kvar. Skriv sedan `dw` för att ta bort resten av ordet.

Ta bort en rad

Om du vill ta bort en rad placerar du markören någonstans på raden och skriver `dd`. Raden tas bort och efterföljande rader flyttas upp.

Ta bort en del av en rad

Du kan också ta bort en del av en rad.

Om du vill ta bort allting till *höger* om markören skriver du `D`.

Om du vill ta bort allting till *vänster* om markören skriver du `d0` (ett `d` och en nolla).

Ta bort allt till filens slut

Om du vill ta bort allt från och med den aktuella raden till filens slut skriver du `dG`.

Ta bort allt från filens början

Om du vill ta bort allt från filens början till och med den aktuella raden skriver du `d1G`.

Kopiera och flytta text

I många ordbehandlingsprogram finns funktioner för att kopiera, klippa ut och klistra in text. Sådana funktioner finns också i `vi`.

När du ska kopiera, klippa ut och klistra in text i `vi` använder du kommandona `yy` eller `Y` (kopiera), `dd` (klippa ut) och `p` eller `P` (klistra in).

Kopiera rader

Om du vill kopiera en rad använder du två kommandon: först `yy` eller `Y` för att kopiera raden och sedan antingen `p` eller `P`. Kommandona `Y` och `yy` fungerar på samma sätt.

Om du vill kopiera en rad placerar du markören på raden och skriver `yy`. Placera sedan markören ovanför den rad där du vill klistra in raden och skriv `p`. Då infogas en ny rad *nedanför* markören och den kopierade raden klistras in.

Om du vill klistra in raden *ovanför* markören skriver du `P`.

Du kan också kopiera flera rader med kommandot `yy`. Om du till exempel vill kopiera elva rader skriver du `11yy`. Då kopieras den rad där markören står och ytterligare tio rader nedanför markören. Längst ned på skärmen visas meddelandet `11 rader kopierade`.

Du kan också använda kommandona `P` och `p` omedelbart efter det att du tagit bort text med något av de kommandon som beskrivits tidigare i det här kapitlet. Då klistras den text du tagit bort in på raden ovanför respektive nedanför markören.



Varning – Använd bara kommandon för markörförflyttning från det att du kopierat eller tagit bort text tills du klistrar in texten igen. Om du tar bort eller kopierar ytterligare text försvinner den text du tagit bort eller kopierat.

Flytta rader

När du ska flytta rader måste du också använda två kommandon: `dd` för att klippa ut text och antingen `p` eller `P` för att klistra in den igen.

Om du vill flytta en rad placerar du markören någonstans på raden och skriver `dd`. Vill du flytta flera rader anger du hur många, till exempel `5dd`.

Flytta sedan markören till raden ovanför den rad där du vill klistra in de urklippta raderna och skriv `p`. Då infogas texten på en ny rad nedanför markören.

Du kan också placera texten ovanför markören genom att skriva `P`.

Använda buffertar

Om du vill kopiera en eller flera rader och klistra in dem på flera ställen i ett dokument kan du kopiera eller klippa ut dem och placera dem i en buffert. Du skapar en buffert genom att före kommandot skriva ett citationstecken (`"`) och ange ett namn på bufferten. Om du till exempel vill kopiera fyra rader och placera dem i en buffert med namnet *a* skriver du `"a4yy`. Du kan använda flera olika buffertar. Du kanske dessutom vill ta bort tolv rader och placera dem i bufferten *b*. Skriv då `"b12dd`.

När du vill infoga texten i en buffert i dokumentet skriver du `"` följt av buffertens namn före kommandot för inklistring (`p` or `P`). Om du till exempel vill infoga de rader som sparats i bufferten *b* skriver du `"bP`.

Du kan ersätta innehållet i en buffert med ny text. Buffertarnas innehåll sparas tills du avslutar `vi`.

Om du använder namngivna buffertar kan du ta bort och kopiera annan text utan att påverka innehållet i buffertarna. Om du vill kan du ersätta innehållet i en buffert med ny text. Buffertarnas innehåll sparas tills du avslutar `vi`.

Upprepa kommandon

Du kan upprepa ett `vi`-kommando flera gånger genom att före kommandot ange en siffra som talar om hur många gånger det ska upprepas. Detta gäller inte alla `vi`-kommandon, men ett stort antal.

De flesta kommandona i föregående avsnitt kan upprepas. Om du till exempel skriver `3dd` tas tre rader bort (kommandot upprepas tre gånger). Om du skriver `2dw` tas två ord bort och skriver du `4x` tas fyra tecken eller mellanslag bort. Du kan också upprepa kommandon för markörflyttning, till exempel `3w` och `2Ctrl-F`. Du lär dig snart vilka kommandon som kan upprepas på detta sätt.

Genom att skriva en punkt (.) upprepas det senaste kommandot för textändring. Om du till exempel just tagit bort en rad med kommandot `dd` kan du flytta markören till en annan rad och ta bort den genom att skriva en punkt.

Använda ex-kommandon

Om du ska kopiera, klippa ut och klistra in större textblock är det lättare att använda *ex*-kommandon än *vi*-kommandona `yy` eller `Y`, `dd` och `p` eller `P`. I stället för att räkna rader och flytta markören till rätt plats på skärmen kan du med hjälp av radnummer ange vilka rader som ska flyttas eller kopieras och sedan ange den rad efter vilken informationen ska klistras in.

Sätta på och stänga av visning av radnummer

Om du vill att radnumren ska visas på skärmen skriver du `:set nu` och trycker på Return.

Radnumren visas i vänstermarginalen. Däremot visas de inte om du skriver ut filen på en skrivare.

```
1 Som en våg, sköljd upp mot stranden
2 vilar du hos mig.
3 När jag smeker dig med handen
4 skälver havet ini dig.
5 Djupa hav
6 som födde dig.
7
8 Kom intill mig, nära till mig,
9 djup som blivit du.
10 Detta som inom dig skälver
11 är ditt hjärta ju.
12 Är ett mänskohjärta ju.
~
~
:set nu
```

Om du vill ta bort radnumren skriver du `:set nonu` och trycker på Return.

Kopiera rader

Du kan kopiera rader med `ex`-kommandot `co`. Detta kommando har följande grundform:

```
:radnummer , radnummer co radnummer
```

De två första radnumren (som skiljs åt med ett kommatecken) anger den första och sista raden i det område som ska kopieras. Det tredje radnumret är numret på raden före den rad där texten ska klistras in.

Om du till exempel vill kopiera raderna ett till och med fem i filen `rita` och placera kopian efter rad tolv skriver du följande kommando:

```
:1,5 co 12
```

Tryck sedan på Return.

När du anger radområden kan du även använda följande tecken:

- punkt (.), vilket betyder "från den aktuella raden"
- dollartecken (\$), vilket betyder "till filens slut"

Om du vill kopiera raderna från och med den aktuella raden till och med rad fem och infoga dessa rader efter rad tolv skriver du följande kommando:

```
:.,5 co 12
```

Om du vill kopiera raderna från och med rad sex till slutet av filen och infoga dessa rader efter rad två skriver du följande kommando:

```
:6,$ co 2
```

Flytta rader

Grundformen för `ex`-kommandot för att flytta rader liknar den för kommandot för kopiering som beskrevs ovan:

```
:radnummer , radnummer m radnummer
```

Du anger vilka rader som ska flyttas och var de ska klistras in på samma sätt som tidigare. Även här kan du använda `.` och `$`. Den enda skillnaden är att kommandot `m` flyttar textblocket från en plats till en annan i stället för att kopiera det.

Om du till exempel vill flytta raderna från och med rad ett till och med rad fem och infoga dessa rader efter rad tolv skriver du följande kommando:

```
:1,5 m 12
```

Tryck sedan på Return.

Ta bort rader

Om du vill ta bort ett antal rader använder du följande kommando:

```
:radnummer , radnummer d
```

Om du till exempel vill ta bort raderna ett till och med fem skriver du följande kommando:

```
:1,5 d
```

Söka och ersätta text med vi

I vi finns flera olika funktioner för att söka efter en viss textmängd i en fil samt en kraftfull funktion för att ersätta text.

Söka efter en teckensträng

En *teckensträng* består av ett eller flera tecken i rad. En teckensträng kan innehålla bokstäver, siffror, skiljetecken, specialtecken, mellanslag, tabbtecken och vagnreturtecken. En sträng kan vara en mening, ett ord, en del av ett ord eller någon annan textmängd.

Om du vill söka efter en teckensträng skriver du / följt av den sträng du vill söka efter och trycker sedan på Return. Markören flyttas till nästa förekomst av strängen. Om du till exempel vill söka efter strängen "bete" skriver du /bete och trycker på Return.

Om du vill gå till *nästa* förekomst av strängen skriver du n, vill du gå till *föregående* förekomst skriver du N.

Om du vill söka baklänges i filen kan du använda ? i stället för /. Då ändras sökriktningen för kommandona n och N så att de fungerar tvärtom mot tidigare.

Normalt har det betydelse om du skriver strängen med små eller stora bokstäver. Om du söker efter "kina" kommer du inte att hitta "Kina". Om du vill att vi ska bortse från om texten är skriven med små eller stora bokstäver skriver du `:set ic`. För att ändra tillbaka till standardinställningen skriver du `:set noic`.

Om vi hittar den sökta strängen stannar markören vid den första förekomsten. Om strängen inte hittas visas meddelandet `Mönstret gick inte att hitta på den sista raden på skärmen`.

Vissa specialtecken (/ & ! . ^ * \$ \ ?) har speciell betydelse i sökprocessen: de kan användas som kommandon som styr sökningen. Om du vill söka efter ett sådant tecken måste du därför skriva ett omvänt snedstreck (\) före tecknet. Om du till exempel vill söka efter strängen "någonting?" skriver du `/någonting\?` och trycker på Return. Om du vill söka efter ett omvänt snedstreck skriver du `/\`.

Förbättra sökningen

Du kan göra sökningen mer exakt genom att i sökkommandot lägga till koder för följande:

- början av rad
- slutet av rad
- början av ord
- slutet av ord
- jokertecken

Om du vill söka efter början av en rad inleder du söksträngen med en cirkumflex (^). Om du till exempel vill söka efter nästa rad som börjar med "Leta" skriver du följande kommando:

```
/^Leta
```

Om du vill söka efter slutet av en rad avslutar du söksträngen med ett dollartecken (\$). Om du till exempel vill söka efter nästa rad som slutar med "leta" skriver du följande kommando:

```
/leta\.$
```


(Lägg märke till att punkten föregås av ett omvänt snedstreck.)

Om du vill söka efter början av ett ord skriver du \<> i början av strängen. Vill du söka efter slutet av ett ord skriver du \> i slutet av strängen. Om du vill söka efter ett ord i stället för efter en sträng kan du kombinera båda dessa koder i sökkommandot. Om du till exempel vill söka efter nästa förekomst av ordet - inte strängen - "leta" skriver du följande kommando:

```
/\
```

En punkt i ett sökkommando står för ett valfritt tecken. Om du till exempel vill söka efter nästa förekomst av "skalle" eller "skulle" skriver du följande kommando:

```
/sk.lle
```

Eftersom du söker efter en sträng och inte ett ord kommer även ord som "träskalle" och "höskulle" att hittas.

Om du vill söka efter ett av två tecken skriver du dem inom hakparenteser. Kommandot `/[md]sträng` söker efter strängar som börjar med antingen m eller d. Om du däremot skriver `/[d-m]sträng` kommer du att hitta strängar som börjar med en bokstav mellan d och m.

En asterisk (*) står för ett valfritt antal valfria tecken. Du kan kombinera asterisken med hakparenteser. Om du till exempel vill söka efter strängar som börjar med en bokstav mellan a och v och som slutar med "lok" och efter alla förekomster av strängen "lok" skriver du följande kommando:

```
/[a-v]*lok
```

Byta ut en teckensträng

Om du vill ersätta en textsträng med en annan använder du sökfunktionerna som beskrivits ovan. Alla specialtecken för sökning kan även användas vid ersättning.

Kommandot har följande grundform:

```
:g/söksträng/s//ersättningssträng/g
```

Avsluta kommandot med att trycka på Return.

Om du till exempel vill ersätta alla förekomster av strängen "bananer" med strängen "apelsiner" skriver du följande kommando:

```
:g/bananer/s//apelsiner/g
```

Tryck sedan på Return.

Om du vill kan du få `vi` att stanna varje gång den sökta strängen hittas och fråga om strängen ska ersättas eller inte.

I följande exempel används kommandot `gc` för att söka efter "apelsiner". Varje gång söksträngen hittas visas en fråga om du vill ersätta strängen eller inte. Svara `y` om du vill ersätta strängen och `n` om du inte vill det.

```
:g/apelsiner/s//bananer/gc
```

Observera – Du kan avbryta en sökning med kontrollfrågor genom att trycka på Ctrl-C.

Flytta markören till en viss rad

Om du vill flytta markören till den sista raden i en fil skriver du G. Vill du återvända till den första raden skriver du 1G.

Om du vill gå till någon annan rad skriver du radens nummer följt av G.

Låt oss säga att du stängde filen `rita` efter att du redigerat rad 51. Du kan då komma tillbaka till den raden genom att öppna filen på nytt och skriva 51G.

Infoga en fil i en annan

Med `vi` är det lätt att infoga en fil i den fil du redigerar. Kommandot för detta har följande grundform:

```
:radnummer r filnamn
```

Om du inte anger ett radnummer infogas filen vid markören.

Om du till exempel vill infoga filen `strindberg` på rad 84 i filen `rita` skriver du följande kommando:

```
:84 r strindberg
```

Du kan också flytta markören till rad 84 och skriva följande kommando:

```
:r strindberg
```

Redigera flera filer samtidigt

Du kan redigera flera filer på samma gång med `vi`. Om du till exempel vill redigera filen `strindberg` samtidigt som du redigerar filen `rita` gör du så här:

1. **Spara först de ändringar du gjort i filen `rita`. Skriv `:w` och tryck på Return.**
2. **Skriv `:n strindberg` och tryck på Return för att öppna filen `strindberg`.**
3. **Gör önskade ändringar i filen `strindberg` och spara filen.**
4. **När du har gjort önskade ändringar och sparat filen `strindberg`, kan du välja mellan följande:**
 - avsluta `vi` genom att skriva `:q` och trycka på Return
 - återgå till `rita` genom att skriva `:n #` och trycka på Return.
 - växla mellan filerna med kommandot `:n #`

Redigera flera filer i en följd

Om du vill redigera flera filer i en följd kan du ange filernas namn efter kommandot `vi` när du startar `vi` från kommandoledtexten:

```
$ vi rita strindberg
```

Filerna visas i den ordning du anger dem. Först visas filen `rita`. När du redigerat den filen skriver du `:n` för att gå vidare till nästa fil (`strindberg`). Om du vill gå vidare till nästa fil utan att spara ändringarna i den aktuella filen skriver du `:n!` i stället för `:n`.

Om du vill redigera filer med liknande namn (till exempel `test1`, `test2` och `test3`) kan du använda jokertecken i `vi`-kommandot:

```
$ vi test*
```

Filerna visas i bokstavsordning efter filnamnen.

Kopiera rader mellan filer

Om du vill kopiera rader från en fil till en annan gör du så här:

1. Redigera den första filen.
2. Spara de rader du vill kopiera i en buffert med kommandot `yy` eller `Y`.
Om du till exempel vill spara tio rader i bufferten `a` skriver du `a10Y`.
3. Använd följande kommando för att gå till nästa fil (`strindberg` i det här exemplet) utan att avsluta `vi`:

```
:n strindberg
```

4. Infoga raderna från den första filen med kommandot `P` eller `p`.

Om du till exempel vill infoga innehållet i bufferten `a` i den nya filen nedanför markören skriver du `ap`.

Kom ihåg att innehållet i buffertarna försvinner om du avslutar `vi`. Använd inte kommandot för att avsluta `vi` (`:q`) förrän du är klar med användandet av buffertarna.

Kontrollera variabelinställningar

I `vi` finns många variabler som påverkar programmets funktion. Du kan ta fram en lista över dessa variabler och deras aktuella inställningar genom att skriva följande kommando när du arbetar med `vi`:

```
:set all
```

Tryck sedan på Return.

Om systemet hänger sig

Om systemet hänger sig är det risk att innehållet i buffertarna går förlorat. Oftast kan du dock återfå den förlorade informationen genom att starta om `vi` med följande kommando:

```
vi -r filnamn
```

filnamn är namnet på den fil du redigerade när systemet slutade fungera. Normalt skickar systemet ett brev till dig med ett meddelande om att det finns en fil där den förlorade informationen sparats.

Sammanställning av grundläggande `vi`-kommandon

I tabellen nedan kan du snabbt och enkelt hitta information om de viktigaste `vi`-kommandona.

Tabell 6-1 Grundläggande `vi`-kommandon

Kommando	Funktion
<i>Starta vi</i>	
<code>vi filnamn</code>	öppnar eller skapar en fil
<code>vi</code>	skapar en ny fil som ska namnges senare
<code>vi -r filnamn</code>	återställer fil som skadats när systemet slutat fungera
<code>view filnamn</code>	öppnar en fil för läsning (det går inte att ändra filen)
<i>Markörflyttning</i>	
<code>h</code>	flyttar markören ett tecken åt vänster
<code>j</code>	flyttar markören en rad nedåt
<code>k</code>	flyttar markören en rad uppåt
<code>l</code>	flyttar markören ett tecken åt höger
<code>w</code>	flyttar markören ett ord åt höger
<code>W</code>	flyttar markören ett ord åt höger (förbi skiljetecken)
<code>b</code>	flyttar markören ett ord åt vänster
<code>B</code>	flyttar markören ett ord åt vänster (förbi skiljetecken)

Tabell 6-1 Grundläggande vi-kommandon (fortsättning)

Kommando	Funktion
e	flyttar markören till slutet av det aktuella ordet
Return	flyttar markören en rad nedåt
backstegstangenten	flyttar markören ett tecken åt vänster
mellanslagstangenten	flyttar markören ett tecken åt höger
H	placerar markören överst på skärmbilden
M	placerar markören i mitten av skärmbilden
L	placerar markören nederst på skärmbilden
Ctrl-F	bläddrar en sida framåt
Ctrl-D	bläddrar en halv sida framåt
Ctrl-B	bläddrar en sida bakåt
Ctrl-U	bläddrar en halv sida framåt
<i>Infoga tecken och rader</i>	
a	infogar tecken till höger om markören
A	infogar tecken i slutet av raden
i	infogar tecken till vänster om markören
I	infogar tecken i början av raden
o	infogar en ny rad nedanför markören
O	infogar en ny rad ovanför markören
<i>Ändra text</i>	
cw	byter ut ett ord (eller en del av ett ord) till höger om markören
c	byter ut en rad
C	byter ut texten från markören till radens slut
s	ersätter tecknet vid markören med ett eller flera tecken
r	ersätter tecknet vid markören med ett tecken
r Return	bryter rader
J	sammanfogar den aktuella raden med raden nedanför

Tabell 6-1 Grundläggande vi-kommandon (fortsättning)

Kommando	Funktion
xp	byter plats på tecknet vid markören och tecknet till höger om markören
~	ändrar från versaler till gemener och vice versa
u	ångrar föregående kommando
U	ångrar alla ändringar på den aktuella raden
:u	ångrar föregående kommando på sista raden
<i>Ta bort text</i>	
x	tar bort tecknet vid markören
X	tar bort tecknet till vänster om markören
dw	tar bort ett ord (eller den del av ordet som står till höger om markören)
dd	tar bort den rad där markören står
D	tar bort den del av raden som står till höger om markören
dG	tar bort all text från markören till filens slut
d1G	tar bort från filens början till markören
:5,10 d	tar bort raderna 5 – 10
<i>Kopiera och flytta text</i>	
yy	kopierar en rad
Y	kopierar en rad
p	klistrar in en kopierad eller borttagen rad nedanför den aktuella raden
P	klistrar in en kopierad eller borttagen rad ovanför den aktuella raden
:1,2 co 3	kopierar raderna 1 – 2 och placerar dem efter rad 3
:4,5 m 6	flyttar raderna 4 – 5 och placerar dem efter rad 6
<i>Radnummer</i>	
:set nu	tar fram radnummer
:set nonu	tar bort radnummer

Tabell 6-1 Grundläggande vi-kommandon (fortsättning)

Kommando	Funktion
<i>Versaler och gemener</i>	
:set ic	anger att ingen skillnad ska göras mellan versaler och gemener
:set noic	anger att skillnad ska göras mellan versaler och gemener
<i>Gå till rader</i>	
G	flyttar markören till den sista raden i filen
1G	flyttar markören till den första raden i filen
21G	flyttar markören till rad 21
<i>Söka och ersätta</i>	
/sträng	söker efter <i>sträng</i>
?sträng	söker bakåt efter <i>sträng</i>
n	söker i den angivna sökriktningen efter nästa förekomst av <i>sträng</i>
N	söker i den angivna sökriktningen efter föregående förekomst av <i>sträng</i>
:g/söka/s//ersätta/g	söker och ersätter text
<i>Rita om skärmen</i>	
Ctrl-L	ritar om skärmen
<i>Infoga en fil i en annan</i>	
:r filnamn	infogar en fil efter markören
:34 r filnamn	infogar en fil efter rad 34
<i>Spara och avsluta</i>	
:w	sparar ändringarna i den aktuella filen
:w filnamn	sparar ändringarna i den angivna filen
:wq	sparar ändringarna och avslutar vi
ZZ	sparar ändringarna och avslutar vi
:q!	avslutar vi utan att spara ändringarna

Elektronisk post



SunOS innehåller programmet `mailx` som du använder för att sända och ta emot elektronisk post (*email*). `mailx` erbjuder dig hjälpmedel för att läsa, skriva, sända, ta emot, spara och ta bort brev. Programmet `mailx` är inte fönsterbaserat och kan därför användas på vilken terminal som helst. Även om du föredrar ett fönsterbaserat postverktyg kommer du att upptäcka att `mailx` är enkelt att hantera när du snabbt vill sända ett brev. Här kan du också läsa om hur du definierar egna aliasnamn.

Observera – Om du arbetar i OpenWindows-miljö och ikonerna för postverktyget visas på skärmen, bör du avsluta postverktyget innan du följer exemplen i det här kapitlet. Du kommer annars att ha två postfunktioner aktiva, vilket kan leda till fel- och varningsmeddelanden. Det är ingen fara att sända brev i kommando- eller skalfönstret, men om du läser din post och sparar eller tar bort brev, kan detta påverka din inkorg och orsaka problem för postverktyget.

Grunderna i mailx

I det här avsnittet kan du läsa om hur du använder `mailx`. I senare avsnitt kommer du att lära dig mer om avancerade funktioner.

En mottagares användarnamn och systemnamn fungerar som unika adresser i programmet `mailx`. Om mottagaren befinner sig i samma system som avsändaren, behöver avsändaren bara ange mottagarens användarnamn. Varje

användare har en *inkorg* där han eller hon tar emot post. Inkorgen finns vanligtvis i katalogen `/var/mail/användarnamn`, där användarnamnet är det namn som används vid inloggning.

Programmet `mailx` meddelar dig när du har fått post och placerar posten i inkorgen. När du har läst posten placerar `mailx` automatiskt breven i en fil med namnet `mb0x`, som finns i din hemkatalog.

Starta mailx

Starta `mailx` genom att skriva följande kommando vid en ledtext och trycka på Return:

```
$ mailx
```

Om du inte har någon oläst post visas följande meddelande på skärmen:

```
Det finns ingen post till användarnamn
$
```

användarnamn är ditt namn för inloggning.

Skicka ett provbrev till dig själv

Om du vill se hur `mailx` fungerar kan du börja med att skicka ett provbrev till dig själv. Ge kommandot `mailx` igen, men ange nu din adress (ditt användarnamn och systemnamn). Om du till exempel loggat in som `vera` och systemnamnet är `dalsland`, blir din adress `vera@dalsland`. Ibland räcker det med att ange användarnamnet. Kontrollera med systemadministratören vad som gäller i ditt nätverk.

```
$ mailx vera@dalsland
```

Då visas svarar med fältet `Subject :` på skärmen:

```
$ mailx vera@dalsland
Subject :
```

Om du vill kan du skriva några ord som beskriver innehållet i brevet du ska skicka till dig själv och trycka på Return. Skriv sedan själva brevet. Använd korta rader och tryck på Return i slutet av varje rad. (Observera att du endast kan göra ändringar genom att trycka på backstegstangenten eller skriva om raderna *innan* du trycker på Return.)

Ditt provbrev kan nu se ut ungefär så här (avståndet mellan raderna får du genom att trycka på Return två gånger):

```
$ mailx vera@dalsland
Subject: till någon som inte längre bryr sig om

Kära Vera!

Vad glömmas kan under åren, glöm!
Lyft mig ut som en död ur ditt sinne!
I dag vår kärlek en yster dröm
i morgon ett ungdomsminne.

Vi ses!

Vera
```

När du vill skicka provbrevet trycker du på Return för att avsluta den sista raden i brevet och trycker sedan på Ctrl-D. När ditt brev har skickats iväg visas kommandoledtexten igen.

Läsa ditt provbrev

Om du vill läsa ditt provbrev anger du kommandot `mailx` igen. På skärmen visas ungefär följande information:

```
$ mailx
Mail version 4.0 Thu Jan 16 12:59:09 PST 1992  Skriv ? om du vill
ha hjälp.
"/var/mail/vera": 2 brev 1 nytt/nya
  U 2 halvar@medelpad  mån mar 15 09:01  14/318 slutgiltig vers.
>N 1 vera@dalsland   mån mar 15 11:12  21/453 till någon som
&
```

Den första raden identifierar programversionen. Den andra raden anger din inkorg, som vanligtvis finns i `/var/mail/ användarnamn`, där din post finns. Den tredje raden i det här exemplet är brevhuvudet i ditt brev. Bokstaven N i början av raden betyder att det är ett nytt brev. Ett U (unread) innebär att brevet var nytt, men oläst innan du avslutade programmet `mailx`. (Informationen på skärmen beskrivs i detalj i avsnittet "Läsa brev" i det här kapitlet.)

Varje brev tilldelas ett nummer när det tas emot: Veras brev till sig själv får nummer 1.

Om du vill läsa ett brev, skriver du dess nummer vid ledtexten för mailx (tecknet &), enligt följande:

```
$ mailx
Mail version 4.0 Thu Jan 16 12:59:09 PST 1992  Skriv ? om du vill
ha hjälp.
"/var/mail/vera": 1 brev 1 nytt/nya
>N 1 vera@dalsland  mån mar 15 11:12 21/453 till någon som
& 1

To: vera@dalsland
From: vera@dalsland
Subject: till någon som inte längre bryr sig om

Kära Vera!

Vad glömmas kan under åren, glöm!
Lyft mig ut som en död ur ditt sinne!
I dag vår kärlek en yster dröm
i morgon ett ungdomsminne.

Vi ses!

Vera

&
```

Avsluta mailx

När du är klar med mailx kan du avsluta programmet med hjälp av två kommandon: q (quit) och x (exit).

Om du använder q vid ledtexten för mailx ser skärmen ut så här:

```
& q
```

Om du trycker på Return visas följande meddelande:

Ett brev har sparats i *hemkatalog*/mbox.

hemkatalog är sökvägen till din hemkatalog.

När du använder `q` för att avsluta `mailx` efter att ha läst ett meddelande, flyttas ditt brev från inkorgen och sparas i filen `mbox` i din hemkatalog. `mailx` sparar också de ändringar du gjort.

Om du skriver `x` vid ledtexten för `mailx` ser skärmen ut så här:

```
& x
```

Om du sedan trycker på Return sparas inte dina ändringar. De brev du har läst flyttas inte till filen `mbox`.

Läsa brev

Om du har fått brev meddelar `mailx` detta varje gång du loggar in i systemet. Följande meddelande visas:

```
Du har fått post
```

eller

```
Du har fått ny post
```

Om du vill läsa dina brev skriver du `mailx` vid kommandoledtexten och trycker på Return. Om du inte har fått någon post visas följande meddelande:

```
Det finns ingen post till användarnamn
```

Om du har fått post kan listan som visas se ut så här:

```
$ mailx
Mail version 4.0 Thu Jan 16 12:59:09 PST 1992  Skriv ? om du vill
ha hjälp.
"/var/mail/vera": 4 brev 1 nytt/nya 2 olästa
  1 vera@dalsland   mån mar 15 11:12 21/453 till någon som
  U 2 henrik@storstad mån mar 15 18:31 19/353 min ensamhet
  U 3 fredrik@landsort tis mar 16 10:22 24/557 vårens projekt
>N 4 ulrik@vildmark  tis mar 16 23:59 14/280 stora problem

&
```


mailx visar information om själva programmet (versionsnummer och datum) och instruktioner om hur du får hjälp (skriv ett frågetecken).

På nästa rad anges var din inkorg finns placerad, det antal brev du har fått och brevens status.

mailx visar en numrerad lista över breven i din inkorg. Följande information visas (från vänster till höger):

- *Status*: anger om brevet är nytt (N), oläst (U) eller läst (ingen symbol). Tecknet > i början av raden står framför det aktuella brevet. Brev som har tagits bort markeras med en asterisk (*).
- *Nummer*: anger den ordning i vilken breven togs emot.
- *Avsändare*: anger namnet på användaren (och vanligtvis värddatorn) som har skickat brevet.
- *Datum och klockslag*: anger datum och tid när brevet sändes.
- *Storlek*: anger antal rader/antal tecken i brevet.
- *Ämne*: anger vad brevet handlar om.

Om du har många brev i din inkorg kanske inte alla brev visas i listan. Om så är fallet skriver du något av följande kommandon:

- z - visar nästa skärmbild med brevhuvuden.
- h - visar föregående skärmbild med brevhuvuden.
- h - visar listan med brevhuvuden en gång till.

Om du vill titta på det aktuella brevet (det brev som är markerat med >), trycker du på Return. Om du trycker på Return en gång till visas nästa brev. Om du vill läsa ett specifikt brev i listan skriver du brevets nummer och trycker på Return.

Ta bort och återfå borttagna brev

När du har läst klart ett brev kanske du vill ta bort det i stället för att spara det i filen `mbox` (vilket är standardinställningen när du avslutar `mailx`).

Om du vill ta bort det senaste brevet du läst, skriver du `d` vid ledtexten för `mailx`. Om du vill ta bort ett specifikt brev från inkorgen, använder du kommandot

`d nummer`

Ge följande kommando om du vill ta bort brev nummer 2:

```
& d 2
```

Du kan också ta bort flera brev i taget. Om du vill ta bort brev nummer 1 och 3, skriver du följande:

```
& d 1 3
```

Om du vill ta bort en hel serie med brev, till exempel brev 1 till 3, skriver du följande:

```
& d 1-3
```

Innan du avslutar programmet `mailx`, kan du återfå borttagna brev. Använd följande kommando:

`u nummer`

Tryck sedan på Return. Om du till exempel vill återfå brev nummer 2, anger du följande kommando:

```
& u 2
```

Om du vill återfå det brev du senast tagit bort, skriver du bara `u` vid ledtexten för `mailx` direkt efter borttaget. Om ditt senaste kommando till exempel var `d 2-5`, återfår du brev nummer 2, 3, 4 och 5.

När du avslutar programmet `mailx` med kommandot `q` kan du inte längre återfå brev du har tagit bort. Om du däremot avslutar `mailx` med kommandot `x` sparas inte ändringarna i inkorgen. Det innebär att brev som du läst behåller markeringen `U`, borttagna brev återfås och så vidare.

Skriva ut brev

Du kan skriva ut ett brev på skrivaren genom att använda ett utskriftskommando. Använd följande kommando vid ledtexten för `mailx`:

```
| nummer lp
```

(Symbolen `|` kallas för *programkoppling*.) Skriv följande kommando och tryck på Return om du vill skriva ut en kopia av brev nummer 2:

```
& |2 lp
```

Om du inte anger ett brevnnummer skrivs det aktuella brevet ut. Om du vill ha mer information om hur du använder kopplingsoperatoren, kan du läsa "Omdirigera dataflöde" i kapitel 2 "Grundläggande SunOS-kommandon".

Sända brev

Om du vill sända brev med `mailx` behöver du veta mottagarens användarnamn. Om mottagaren finns på ett annat system kan du behöva veta namnet på det systemet. För att ta reda på denna information kan du använda kommandona `who`, `finger` eller `rusers`.

Om du anger kommandot `who` visas alla användare som är inloggade på samma filhanterare som du är inloggad på. Listan innehåller användarnamn, terminaltyp samt datum och tid för inloggning. Exempel:

```
$ who
helmer      tty15      feb 20 10:22
susanne     tty04      feb 20 10:37
torvald     tty07      feb 20 11:49
stefan      tty06      feb 20 12:02
```

Om du anger kommandot `finger` visas samma information som med kommandot `who`, men informationen är mer detaljerad. Vilken information som visas beror på hur systemadministratören har definierat kommandot. Informationen kan se ut så här:

```
$ finger
```

Login	Name	TTY	Idle	When
helmer	Helmer Bryd	tty15	43	tors 10:22
susanne	Susanne Ek	tty04		tors 10:37
torvald	Torvald Kulle	tty07	12	tors 11:49
stefan	Stefan Strand	tty06	22	tors 12:02

Kommandot `rusers` ger dig information om de användare som är inloggade på det lokala nätverket. I kapitel 9 "Arbeta i nätverk" kan du läsa mer om hur du använder kommandot `rusers`.

När du har fått den information du behöver, följer du stegen nedan för att sända ett brev.

1. Skriv kommandot `mailx` följt av användarens adress:

```
$ mailx användarnamn@systemnamn
```

användarnamn är mottagarens namn för inloggning och *systemnamn* är namnet på mottagarens system.

- Om du redan har startat `mailx` behöver du bara skriva `m` vid ledtexten för `mailx`, och sedan mottagarens användarnamn och systemnamn:

```
& m användarnamn@systemnamn
```

- Om du vill sända samma brev till flera mottagare, skiljer du varje adress åt med ett mellanslag eller ett kommatecken, till exempel:

```
$ mailx henrik@storstad marianne@uppland sten@uranus
```

Du kan också skriva så här:

```
$ mailx henrik@storstad,marianne@uppland,sten@uranus
```

2. När du trycker på Return blir du ombedd att ange ett ämne. Ange ämnet och tryck på Return.
3. Skriv själva brevet. När du vill skapa en ny rad trycker du på Return. En rad som bryts betraktas bara som en separat rad om du tryckt på Return i slutet av raden.

Observera – Varje textrad i brevet kan bestå av upp till 256 tecken. När du överskrider gränsen kommer din skärm att låsa sig. Om detta inträffar trycker du på Ctrl-C för att ta bort brevet.

4. När du har skrivit klart brevet trycker du på Return för att flytta markören till en ny rad. Tryck sedan på Ctrl-D för att sända ditt brev.

Brev som inte kan sändas

Om du anger en felaktig användaradress när du sänder ett brev, visas följande meddelande:

```
användarnamn@systemnamn...User unknown
```

Brevet läggs tillbaka i din inkorg. Nästa gång du anger kommandot `mailx`, anges det i brevhuvudet att du har fått tillbaka post enligt exemplet nedan:

```
N 1 Mailer-Daemon Fri Jan 3 11:13 8/49 Returned mail: User unknown
```

När ett brev inte kan sändas, sparas det också i filen `dead.letter` i din hemkatalog.

Avbryta ett osänt brev

Du kan avbryta ett brev när som helst *innan* du sänder det genom att trycka på Ctrl-C två gånger.

Lägga till kopior och dolda kopior

Innan du sänder ett brev kan du ange att kopior ska sändas till andra än huvudadressaterna. Du kan också sända dolda kopior. (Mottagarna kan läsa adresserna för kopiorna, men inte adresserna för de dolda kopiorna.)

Du kan till exempel sända kopior eller dolda kopior till dig själv för att få en kopia av brevet.

Du kan sända en kopia på tre olika sätt:

- Använd ett textredigeringsprogram för att redigera filen `.mailrc` (i din hemkatalog) och infoga följande rad:

```
set askcc
```

Programmet `mailx` tar nu fram ledtexten för kopian (`Cc:`) efter ledtexten för ämnet. Ange adresserna på de användare du vill sända kopior till. Skilj adresserna åt med mellanslag.

- När du har skrivit själva brevet trycker du på Return (innan du trycker på Ctrl-D) för att flytta markören till en ny rad. Skriv sedan följande kommando:

```
~c adress(er)
```

Om du vill använda den här metoden för att sända kopior till flera mottagare, skiljer du adresserna åt med mellanslag. Exempel:

```
~c henrik@storstad berra@veteranbilar bosse@butter
```

- Fältet `Cc:` skapas med kommandot `~h`. Då visas hela brevhuvudet. `~h` visar ledtexterna `To:`, `Subject:`, `Cc:` och `Bcc:` (kopia och dold kopia), en i taget. Tomma rader kan fyllas i och ifyllda rader kan skrivas om på nytt. Precis som med andra kommandon med tecknet `~` kan du alltid använda kommandot `~h` på en ny rad.

Observera – `~c`, `~h` och andra kommandon med tecknet `~` beskrivs i avsnittet "Kommandon med tecknet `~`" i det här kapitlet.

Infoga en kopia av ett brev eller en fil

Du kan infoga en kopia av ett brev som finns i din inkorg i ett annat brev som du just skriver. Du kan också infoga en kopia av en textfil.

Infoga ett brev

Kommandot för att infoga ett brev är:

`~m nummer`

nummer är numret på brevet som du vill infoga. Om du till exempel vill sända ett brev till en annan användare och infoga en kopia av brev nummer 3 från inkorgen, gör du så här:

- 1. Ange kommandot `~m 3` på en ny rad och tryck på Return.**
- 2. Meddelandet Interpolerar: 3 (fortsätt) visas.**
- 3. Du kommer inte att se texten i brev nummer 3, men mottagaren kommer givetvis att göra det. Du kan sedan fortsätta med ditt brev efter texten (fortsätt) eller sända det som det är.**
- 4. Om du vill se hela brevet, inklusive det brev du har fogat in, anger du kommandot `~p`.**

Infoga en fil

Du kan också infoga en kopia av en textfil i brevet. Använd följande kommando när du skriver ett brev:

`~r filnamn`

Om du till exempel vill infoga filen `planering` i det aktuella brevet skriver du:

```
~r planering
```

Svara på ett brev

Du kan svara på ett brev genom att ange följande kommando vid ledtexten för `mailx`:

`r nummer`

(Om du utelämnar brevs nummer svarar `mailx` på det aktuella brevet.) Om du till exempel vill svara på brev nummer 2, anger du följande kommando:

```
& r 2
```

`mailx` adresserar automatiskt ditt brev och lägger till fältet `Re: Subject:` som motsvarar originalraden `Subject:`. Svara på brevet på vanligt sätt.

`R` är en variant av svarskommandot som sänder dina svar till alla mottagare av originalbrevet liksom till avsändaren. Använd bara detta kommando om det är absolut nödvändigt för att undvika att sända onödiga brev.

Observera – Du kan infoga ett brev i ett svar enligt beskrivningen i föregående avsnitt. Om du vill infoga en kopia av det brev du svarar på anger du bara kommandot `~m` utan något brevnnummer.

Spara och återfå brev

Förutom att sända och ta emot brev kan du spara och återfå brev senare. I `mailx` kan du spara brev genom att infoga dem i vanliga textfiler. Du kan också infoga brev i specialfiler som kallas mappar. Båda metoderna beskrivs nedan.

`mailx` skiljer mellan att spara och kopiera brev. Om du sparar ett brev tas det bort från inkorgen och infogas i en fil eller mapp. Om du kopierar ett brev finns brevet kvar i inkorgen samtidigt som en kopia infogas i en fil eller en mapp.

Spara brev i och kopiera brev till filer

Om du vill spara ett brev i en fil skriver du följande kommando vid ledtexten för mailx:

```
s nummer filnamn
```

nummer är numret på det brev du vill spara och *filnamn* är namnet på den fil du vill spara brevet i. Om du till exempel vill spara brev nummer 3 i en fil som heter ~/utpost/ekonomi, skriver du:

```
& s 3 ~/utpost/ekonomi
```

(Kom ihåg att tecknet ~ i en sökväg representerar din hemkatalog.)

Du kan också spara flera brev på en gång i samma fil. Om du till exempel vill spara brev 3, 5, 6, 7 och 8 i ~/utpost/ekonomi, skriver du:

```
& s 3 5-8 ~/utpost/ekonomi
```

Om filen du anger inte finns, skapas den av mailx. Om filen finns, läggs brevet till i slutet av filen.

Om du sparar en fil tas den bort från inkorgen. En asterisk (*) visas vid brevhuvudet för de brev som har sparats.

Om du vill spara brevet i en fil och samtidigt låta det ligga kvar i inkorgen, kopierar du brevet enligt följande:

```
& c 3 ~/utpost/ekonomi
```

Spara brev i och kopiera brev till mappar

Du kan slippa skriva den fullständiga sökvägen till filer om du sparar brev i eller kopierar brev till en mapp för post. Mappar är specialfiler som sparas i en katalog med mappar.

Fördelarna med att spara eller kopiera brev till mappar är att breven automatiskt läggs i samma katalog, där du enkelt kan komma åt dem utan att du behöver ange långa sökvägar.

Definiera en katalog för mappar

Om du vill använda mappar måste du först definiera en katalog för mappar. Gör så här:

1. Skapa först en katalog med kommandot `mkdir`.

Om du till exempel vill att din katalog ska heta `Brev`, skriver du så här:

```
$ mkdir Brev
```

2. Använd sedan ett redigeringsprogram för att redigera filen `.mailrc` (som innehåller tillägg till `mailx`) i din hemkatalog och definiera en sökväg för katalogen med mappar.

Här behöver du redigera variabeln `set folder` för att inkludera den fullständiga sökvägen till katalogen med mappar. Exempel:

```
set folder=/home/stockholm/vera/Brev
```

Du kan också använda C-skalets kortkommando `~` för att specificera hemkatalogen.

```
set folder=~ /Brev
```

Du kan nu placera brev i katalogen med mappar. (De ändringar du har gjort i filen `.mailrc` kommer att aktiveras nästa gång du startar `mailx`.)

Ange mappnamn i kommandon

Du kan använda samma kommandon för att spara brev i eller kopiera brev till filer och mappar med undantag av att mappnamnet föregås av ett plustecken (+) i stället för en sökväg. Plustecknet talar om för `mailx` att mappen finns i katalogen med mappar (`Brev`).

Om du till exempel vill spara brev nummer 3 i en mapp som kallas `projekt`, skriver du:

```
& s 3 +projekt
```

`mailx` tolkar det här kommandot som att brev nummer 3 ska sparas i `~/Brev/projekt`. (Om mappen inte finns skapas den.)

Kopiera brevet till en mapp genom att skriva:

```
& c 3 +projekt
```

Sända brev direkt till filer eller mappar

Du kan sända en kopia av ett brev direkt till en fil eller en mapp. Om du vill sända en kopia till en mapp skriver du mappens namn i fältet `Cc:` eller `Bcc:`. Du gör på samma sätt när du sänder en kopia till en fil, men du måste då skriva den fullständiga sökvägen.

Läsa brev i filer och mappar

Om du vill läsa brev som finns sparade i en fil använder du följande kommando:

```
mailx -f filnamn
```

Om du följer exemplet ovan kan du läsa filen `~/memo/finans` genom att skriva följande kommando:

```
$ mailx -f ~/memo/finans
```

Du kan läsa brev som finns sparade i en mapp med ett liknande kommando: du skriver `+` i stället för en sökväg. Om du till exempel vill läsa brev i mappen `projekt`, skriver du:

```
$ mailx -f +projekt
```

Det här kommandot startar `mailx` i den fil eller mapp du har angett. Endast brevhuvudena visas. Välj ett brev som du vill läsa genom att skriva dess nummer vid ledtexten för `mailx` och trycka på Return.

Du kan också arbeta med mappar med post i programmet `mailx`. Om du vill se en lista över dina mappar skriver du följande vid ledtexten för `mailx`:

```
& folders
```

Om du vill gå från inkorgen till en mapp, använder du följande kommando:

```
& folder +mappnamn
```

Om du vill återgå till inkorgen, skriver du följande vid ledtexten:

```
& %
```

Om du vill återgå till föregående mapp, skriver du:

```
& #
```

Använda `vi` tillsammans med `mailx`

Du kan använda redigeringsprogrammet `vi` för att skriva brev när du använder programmet `mailx`. Du kan därmed korrigera misstag och lägga till eller ta bort text innan du sänder dina brev. Om du inte är bekant med programmet `vi` kan du läsa mer i kapitel 6 "Redigeringsprogrammet `vi`".

I `mailx` kan du använda standardkommandona för `vi` för att infoga, ta bort och ändra text.

Gör så här om du vill skriva ett brev med `vi`:

1. Ge kommandot `mailx` tillsammans med en adress vid ledtexten för `mailx` (&) eller kommandoledtexten (\$).
2. Ange det aktuella ämnet i fältet `Subject:`. Tryck på Return.

3. Starta `vi` genom att ge kommandot `~v` på en ny rad. En tom fil, som ligger i katalogen `/tmp`, visas.
4. Använd `vi`-kommandona för att skriva och redigera brevet.
5. När du är klar avslutar du `vi` med kommandot `:wq` eller `ZZ`.

När du har avslutat `vi` visas meddelandet (fortsätt). Du kan då antingen lägga till ytterligare text till brevet (utanför `vi`) eller sända det genom att trycka på `Ctrl-D`.

Aliasnamn

Aliasnamn är ett antal användarnamn som grupperats under ett namn.

Använd aliasnamn när du ofta sänder brev till samma personer. Om du till exempel ofta skickar post till `henrik@storstad`, `greger@viking` och `marianne@uppland`, kan du skapa aliasnamnet `kamrater`. Varje gång du skickar post till `kamrater`, kommer alla tre personerna att få post.

Du kan ange aliasnamn i följande filer:

- `.mailrc`
- `/etc/aliases`

Aliasnamn som definieras i filen `.mailrc` fungerar på ett annat sätt än aliasnamn som definieras i `/etc/aliases`. Skillnaderna beskrivs i tabell 7-1 i slutet av det här avsnittet.

Definiera aliasnamn i `.mailrc`

Tänk på följande när du definierar aliasnamn i filen `.mailrc`:

- Aliasnamn i `.mailrc` är *privata*, vilket innebär att endast du kan använda dem. Om du till exempel definierar aliasnamnet `kamrater` i `.mailrc` och en annan användare försöker sända post till `kamrater`, kommer felmeddelandet `unknown user` att visas.
- När posten har skickats visas namnen på de personer som du har skickat post till. Om du till exempel sänder post till `kamrater`, kommer din post att sändas till rätt personer precis som om du hade skrivit alla mottagares namn separat. Mottagarna kan inte se att du har använt ett aliasnamn.

Filen `.mailrc` finns i din hemkatalog. Filen innehåller ett antal inställningar som kontrollerar `mailx` och postverktyget i OpenWindows.

Om du vill lägga till ett aliasnamn i `.mailrc`, skriver du följande:

```
$ vi ~/.mailrc
```

Observera – Du kan använda vilket redigeringsprogram du vill för att redigera filen `.mailrc`. Exemplet ovan visar hur du använder redigeringsprogrammet `vi` för att redigera filen. Om du inte är bekant med detta redigeringsprogram kan du läsa kapitel 6 "Redigeringsprogrammet `vi`".

Varje aliasnamn skrivs på en rad i filen. Det innebär att aliasnamnet kan fortsätta på en annan rad, men att det inte kan innehålla vagnreturer. Varje aliasnamn bör därför innehålla följande information (åtskilda med mellanrum):

- ordet `alias`
- aliasnamnet (det får bara bestå av ett ord)
- mottagarna (användarnamn och systemnamn) i aliasnamnet (åtskilda med mellanslag)

Exemplet nedan visar två aliasnamn. Det första (`kamrater`) innehåller tre mottagarnamn. Det andra (`fotboll`) innehåller åtta. I `fotboll` kan du se hur raderna bryts på skärmen. Detta går bra så länge det inte finns några vagnreturer i namnen.

```
alias kamrater henrik@storstad greger@viking marianne@uppland
alias fotboll eva@elit sten@uranus helmer@venus janne@pluto
henrik@storstad bo@kairo marianne@uppland stina@egypten
```

Om du vill sända post till personer som definierats med aliasnamn i med `.mailrc`, skriver du bara aliasnamnet. Skriv *inte* systemnamnet. Skriv till exempel följande meddelande:

```
$ mail kamrater
Subject: Lunch på fredag?

Kära arbetskamrater! Vad sägs om att åka in till stan och äta lunch
på fredag? Hör av er till mig om ni är intresserade.
```

Det meddelande som visas för mottagarna ser ut så här (observera att adresserna till alla i gruppen visas i fältet `To`):

```
To: henrik@storstad greger@viking marianne@uppland
Subject: Lunch på fredag?

Kära arbetskamrater! Vad sägs om att åka in till stan och äta lunch
på fredag? Hör av er till mig om ni är intresserade.
```

Definiera aliasnamn i `/etc/aliases`

Tänk på följande när du definierar aliasnamn i `/etc/aliases`:

- Aliasnamn i filen `/etc/aliases` är *allmänna*. Det betyder att om du definierar aliasnamnet `fotboll`, kan vem som helst använda aliasnamnet och sända post till `fotboll@ditt_systemnamn`.
- När du har skickat posten visas *inte* namnet på de olika mottagarna. Om du till exempel skickar post till `fotboll@systemnamn`, är det just detta som visas för mottagarna. Mottagarna får reda på vilket aliasnamn som har använts, men *inte* vilka andra namn som förknippas med aliasnamnet.

Formatet på aliasnamn som skapas i `/etc/aliases` skiljer sig från aliasnamn som har skapats i `.mailrc`. Varje aliasnamn i `/etc/aliases` bör ha följande format:

- Aliasnamnet följt av ett kolon (`:`)
- Mottagarna (användarnamn och systemnamn), skilda åt med kommatecken. Observera att aliasnamn *inte* behöver skrivas på en enstaka rad.

Om du vill redigera filen `/etc/aliases`, måste du först bli superanvändare. Om du måste ange ett lösenord för att bli superanvändare, får du kontakta någon som känner till lösenordet.

Skriv följande för att bli superanvändare i systemet:

```
$ su
Password:
#
```

Observera att kommandoledtexten ändras när du blir superanvändare.

Följande exempel visar hur aliasnamnet `fotboll@dalsland` läggs till standardfilen `/etc/aliases`.

```
# vi /etc/aliases
##
#Aliases can have any mix of upper and lower case on the left-
#hand side,
#but the right-hand side should be proper case (usually lower)
#
#   >>>>>>>>>The program "newaliases" will need to be run after
#   >> NOTE >>this file is updated for any changes to
#   >>>>>>>>>show through to sendmail.
#
#@(##)aliases 1.10 89/01/20 SMI
##
# Following alias is required by the mail protocol, RFC 822
# Set it to the address of a HUMAN who deals with this system's
# mail problems.
Postmaster: root

# Alias for mailer daemon; returned messages from our MAILER-
# DAEMON
# should be routed to our local Postmaster.
MAILER-DAEMON: postmaster

# Aliases to handle mail to programs or files, eg news or vacation
# decode: "|/usr/bin/uudecode"
nobody: /dev/null

# Sample aliases:
```



```
# Alias for distribution list, members specified here:
#staff:wnj,mosher,sam,ecc,mckusick,sklower,olson,rwh@ernie

# Alias for distribution list, members specified elsewhere:
#keyboards: :include:/usr/jfarrell/keyboards.list

# Alias for a person, so they can receive mail by several names:
#epa:eric

#####
# Local aliases below #
#####
fotball@dalsland: eva@elit sten@uranus helmer@venus janne@pluto
henrik@storstad bo@kairo marianne@uppland stina@egypten
:wq          (avsluta vi och spara filen /etc/aliases)
# exit      (gå ur superanvändarläget)
$
```

Du kan använda vilket redigeringsprogram du vill för att redigera filen `/etc/aliases`. Exemplet ovan visar hur du använder `vi`. Om du inte är bekant med `vi`, kan du läsa kapitel 6 "Redigeringsprogrammet `vi`".

Observera att nummertecknet (`#`) som visas i filen `/etc/aliases` har placerats där för att göra raderna till kommentarer. Nummertecknet förhindrar att systemet använder aliasnamnen som riktiga aliasnamn.

Placera inte nummertecknen framför aliasnamn som du vill lägga till en fil, om du inte vill kommentera bort ett aliasnamn.

Om du vill sända ett brev med ett aliasnamn som definierats i filen `/etc/aliases`, anger du ett aliasnamn och ett systemnamn. Skriv till exempel följande:

```
$ mail fotboll@dalsland
Subject: Helgens program
```

```
Låt oss träffas i parken vid den stora eken och diskutera planerna
för helgkonferensen. Det vore bra om du kunde vara där vid
fyratiden.
```

Mottagarna får då ett meddelande som ser ut så här:

```
To: fotboll@dalsland
Subject: Helgens program
```

```
Låt oss träffas i parken vid den stora eken och diskutera planerna
för helgkonferensen. Det vore bra om du kunde vara där vid
fyratiden.
```

Observera att det är aliasnamnet som visas i fältet `TO:` och inte de olika mottagarnas namn.

När du sänder post med hjälp av ett aliasnamn av den här typen, bör du se till att även inkludera systemnamnet. Om du definierar ett aliasnamn som du kallar riddare på systemet neptunus, bör du sända posten till `riddare@neptunus`.

Tabell 7-1 innehåller en översikt över skillnaderna mellan aliasnamn som har skapats i `.mailrc` och aliasnamn som har skapats i `/etc/aliases`.

Tabell 7-1 Jämförelse mellan aliasnamn i `.mailrc` och `/etc/aliases`

	<code>.mailrc</code>	<code>/etc/aliases</code>
Måste jag vara superanvändare för att ändra i filen?	nej	ja
Sänd meddelandet till:	<i>aliasnamn</i>	<i>aliasnamn@systemnamn</i>
Kan mottagarlistan läsas av mottagarna?	ja	nej
Ska namnen skiljas åt med kometecknen?	nej	ja
Ska alla namn stå på en rad?	ja	nej
Kan någon annan använda aliasnamnet?	nej	ja

Om du vill ha mer information om aliasnamn, skriver du `man aliases` eller `man addresses` vid systemledtexten.

Kommandon med tecknet ~

Du kan använda kommandon med tecknet ~ för att utföra en mängd olika funktioner när du skriver brev. Dessa kommandon består av tecknet ~ följt av ett enskilt tecken. Följande tabell beskriver några av de vanligaste kommandona med tecknet ~. Vissa av kommandona har redan beskrivits i det här kapitlet.

Observera – Om du vill infoga ett ~-tecken i ett brev skriver du två ~-tecken efter varandra. Endast ett tecken kommer att visas.

Tabell 7-2 Kommandon med tecknet ~ (mailx)

Kommando	Funktion
~! <i>kommando</i>	Startar ett skalkommando.
~.	Motsvarar Ctrl-D som innebär att du markerar slutet på filen.
~?	Visar en lista över kommandon med tecknet ~.
~b <i>användarnamn</i>	Lägger till användarnamn i listan över dolda kopior (Bcc:).
~c <i>användarnamn</i>	Lägger till användarnamn i listan över kopior (Cc:).
~d	Flyttar innehållet i filen <code>dead.letter</code> till det aktuella brevet.
~f <i>nummer</i>	Vidarebefordrar det angivna brevet. Kommandot är endast giltigt när du sänder ett meddelande samtidigt som du läser post.
~h	Flyttar markören till fälten Subject, To, Cc och Bcc.
~m <i>nummer</i>	Infogar text från ett specifikt brev till det aktuella brevet. Kommandot är endast giltigt när du sänder ett meddelande samtidigt som du läser post.
~p	Skriver ut det meddelande som visas på skärmen.
~q	Motsvarar två tryckningar på tangenterna Ctrl-C. Om det aktuella brevet inte är tomt sparas innehållet i <code>dead.letter</code> .
~r <i>filnamn</i>	Infogar text från en specifik fil.
~s <i>sträng</i>	Ändrar raden som anger ämne till <i>sträng</i> .
~t <i>namn</i>	Lägger till ett specifikt namn i fältet To.
~w <i>filnamn</i>	Sparar det aktuella brevet utan brevhuvud i en specifik fil.
~x	Avslutar <code>mailx</code> . Fungerar på samma sätt som ~q förutom att meddelandet inte sparas i filen <code>dead.letter</code> .

Om du behöver hjälp: andra `mailx`-kommandon

I `mailx` finns två hjälpkommandon som visar listor över kommandon och funktioner. När du befinner dig i kommandoläget kan du skriva `? vid` ledtexten för `mailx (&)` för att se en lista över kommandon som kan användas i det läget. När du befinner dig i inmatningsläget (till exempel när du skriver ett brev), kan du ge kommandot `~?` för att se en lista över kommandon med tecknet `~`.

Direkthjälpen (`man`) innehåller mer teknisk information om `mailx`. Om du vill se direkthjälpen ger du kommandot:

```
$ man mailx
```

Du kan också läsa *man Pages(1): User Commands*.

Skriva ut



Skrivarhanteringen LP (delsystemet *line printer*) är den del av SunOS som du använder för att hantera utskrifter. Den har många olika funktioner varav de flesta inte beskrivs i den här handboken. Det här kapitlet beskriver bara hur du utför följande grundläggande utskriftsfunktioner med hjälp av skrivarhanteringen:

- Begära en utskrift genom att sända en fil till skrivaren
- Kontrollera en skrivare
- Avbryta en begäran om utskrift

I *User Accounts, Printers, and Mail Administration* finns en fullständig beskrivning av skrivarhanteringen.

Begära en utskrift

Du kan skriva ut en fil genom att ange kommandot `lp` på kommandoraden. När du begär en utskrift på det här sättet placeras din begäran i skrivarens kö och dess id-nummer visas. Därefter visas ledtexten igen.

Skriva ut på standardskrivaren

När en skrivare har ställts in som standardval i skrivarhanteringen kan du begära utskrifter på följande sätt utan att behöva ange skrivarens namn:

```
$ lp filnamn
```

där *filnamn* är namnet på den fil du vill skriva ut.

Den angivna filen placeras i kön till standardskrivaren, och *utskriftens id* visas.

Så här skriver du till exempel ut filen `/etc/passwd`:

```
$ lp /etc/passwd
Utskriftens id är tryckaren-8 (1 fil)
$
```

I *User Accounts, Printers, and Mail Administration* kan du läsa mer om hur du anger standardskrivare.

Skriva ut genom att ange skrivarnamn

Oavsett om någon standardskrivare finns angiven eller inte kan du skriva ut på alla skrivare som ditt system är inställt för. Så här skriver du ut genom att ange skrivarnamn:

```
$ lp -d skrivarnamn filnamn
```

där *skrivarnamn* är namnet på den skrivare som du vill skriva ut på och *filnamn* är namnet på den fil du vill skriva ut.

Den angivna filen placeras i den angivna skrivarens kö och utskriftens id visas.

Så här skriver du till exempel ut filen `/etc/passwd` på skrivaren `pressen`:

```
$ lp -d pressen /etc/passwd
Utskriftens id är pressen-9 (1 fil)
$
```

Om du försöker skriva ut på en skrivare som ditt system inte är inställt för visas ett meddelande. Följande exempel visar ett sådant meddelande:

```
$ lp -d sol /etc/passwd
UX:lp: FEL: Skrivarhanteringen LP känner inte till skrivaren
"sol".
$
```

I *User Accounts, Printers, and Mail Administration* kan du läsa om hur du ställer in ditt system för en skrivare. I avsnittet "Kontrollera skrivaren" i det här kapitlet kan du läsa om hur du tar reda på vilka skrivare du kan använda på ditt system.

Begära meddelande om att utskriften är färdig

När du har skrivit ut en stor fil kanske du vill att skrivarhanteringen ska meddela när utskriften är färdig. Du kan begära att skrivarhanteringen meddelar dig på två sätt:

- genom att sända ett brev med elektronisk post
- genom att skriva ett meddelande i ditt systemfönster

Du får en ett brev med elektronisk post om du skriver tillägget `-m` efter utskriftskommandot:

```
$ lp -m filnamn
```

Du får ett meddelande i systemfönstret om du skriver tillägget `-w` efter utskriftskommandot:

```
$ lp -w filnamn
```

Filnamn är namnet på den fil du vill skriva ut.

Skriva ut flera exemplar

Du kan skriva ut fler än ett exemplar av en fil. Den skrivs ut det antal gånger som du anger med `-n` efter kommandot `lp`. Utskriften betraktas som en enda, och bara ett försättsblad skrivs ut.

Skriv så här om du vill ha flera exemplar av utskriften:

```
$ lp -n antal filnamn
```

där *antal* är det antal exemplar du vill ha och *filnamn* är namnet på det fil som ska skrivas ut.

Så här skriver du till exempel ut fyra exemplar av filen `/etc/passwd`:

```
$ lp -n4 /etc/passwd
Utskriftens id är tryckaren-9 (1 fil)
$
```

Översiktlig tabell över tillägg till kommandot `lp`

Du kan variera dina utskrifter genom att ange tillägg till kommandot `lp`. Du kan ange form, teckensnitt, filter, sidhuvud, titelsida osv. Tabell 8-1 ger en översikt av de tillägg som används mest med kommandot `lp`. Du kan använda tilläggen var för sig eller kombinera dem i vilken ordning som helst på kommandoraden. När du kombinerar tillägg skriver du ett mellanslag mellan varje tillägg. Det ska stå ett bindestreck (-) framför varje tillägg.

Så här anger du till exempel skrivare, begäran om brev och att sex exemplar av filen ska skrivas ut:

```
$ lp -d skrivarnamn -m -n6 filnamn
```

där *skrivarnamn* är namnet på skrivaren och *filnamn* är namnet på den fil du ska skriva ut.

Tabell 8-1 Översikt av vanliga tillägg till kommandot `lp`

Tillägg	Beskrivning
<code>-d</code>	Destination. Bestämmer vilken skrivare utskriften ska göras på.
<code>-m</code>	Mail. Bestämmer att ett brev ska sändas till användaren när utskriften är färdig.
<code>-n</code>	Number. Bestämmer hur många exemplar av en fil som ska skrivas ut.
<code>-t</code>	Title. Bestämmer en titel på utskriften som skrivs på försättsbladet.
<code>-o nobanner</code>	Option. Titelsidan skrivs inte ut vid utskriften.
<code>-h</code>	Header. Ett sidhuvud sätts på varje sida.
<code>-c</code>	Copy. Filen kopieras före utskriften.
<code>-w</code>	Write. Du får ett meddelande när utskriften är färdig.

I man *Pages(1): User Commands* finns en fullständig lista över de tillägg du kan använda med kommandot `lp`.

Kontrollera skrivaren

Använd kommandot `lpstat` för att kontrollera skrivarhanteringen. Du kan kontrollera dina utskrifters plats i skrivarkön, ta reda på vilka skrivare du kan använda och ta reda på dina utskrifters id i fall du vill ta bort dem ur skrivarkön eller avbryta dem.

Kontrollera dina skrivare

Så här skriver du om du vill kontrollera dina utskrifter i skrivarkön:

```
$ lpstat
```

En lista över de filer du har sänt till skrivaren visas.

I följande exempel står en fil i kö för att skrivas ut på skrivaren tryckaren, som är ansluten till systemet dataavd:

```
$ lpstat
tryckaren-10      fredrik      1261   maj 12 17:34 på dataavd
$
```

Kommandot `lpstat` visar varje utskrifts id, användare, storlek samt datum och tid när utskriften sändes till skrivaren.

Ta reda på vilka skrivare som är tillgängliga

Så här tar du reda på vilka skrivare du kan använda i ditt system:

```
$ lpstat -s
```

Köhanterarens status visas, och därefter namnet på den standardvalda skrivaren och en lista över de system och skrivare som du kan använda.

I följande exempel heter systemet `elm`. Köhanteraren är i gång, skrivaren tryckaren är standardvald och det finns två skrivare i nätverket, tryckaren och pressen:

```
$ lpstat -s
Köhanteraren är igång
Systemets standardskrivare: tryckaren
system för tryckaren: dataavd
system för pressen: ekonomiavd
$
```

Visa all statusinformation

Tillägget `-t` till `lpstat` ger en kort lista över skrivarhanterings status.

Så här gör du för att listan med statusinformation ska visas:

```
$ lpstat -t
```

All tillgänglig statusinformation visas.

I följande exempel finns det inga utskrifter i skrivarkön. Hade det funnits hade deras status också visats:

```
$ lpstat -t
Köhanteraren är igång
Systemets standardskrivare: tryckaren
system för pressen: ekonomiavd
tryckaren accepterar utskrifter sedan onsdag, 2 januari kl
18:20:10 MET 1994
pressen accepterar utskrifter sedan måndag, 4 mars kl 15:53:47
MET 1994
skrivaren tryckaren är sysslolös. aktiverad sedan onsdag, 2
januari kl 18:20:22 MET 1994. tillgänglig.

skrivaren pressen är sysslolös. aktiverad sedan måndag, 4 mars kl
15:53:47 MET 1994. tillgänglig.
$
```

Visa skrivarnas status

Du kan ta fram statusinformation för enskilda skrivare genom att ange tillägget `-p` till `lpstat`. Med tillägget får du veta om skrivaren håller på att skriva ut eller inte, när den sattes i gång eller stängdes av och om den kan skriva ut.

Så här begär du information om alla skrivares status:

```
$ lpstat -p
```

I följande exempel är två skrivare i gång, tillgängliga och lediga. Om det hade funnits utskrifter i skrivarnas köer hade det också visats:

```
$ lpstat -p
skrivaren tryckaren är sysslolös. aktiverad sedan onsdag, 2
januari kl 18:20:10 MET 1994. tillgänglig.
skrivaren pressen är sysslolös. aktiverad sedan onsdag, 2 januari
kl 18:20:10 MET 1994. tillgänglig.
$
```

Så här får du information om en enskild skrivares status:

```
$ lpstat -p skrivarnamn
```

där *skrivarnamn* är namnet på en skrivare.

Visa skrivares inställningar

Om du vill veta hur en skrivare är inställd anger du tilläggen `-p` och `-l` med `lpstat`. Detta är särskilt användbart för att ta reda på skrivartyp och typ av innehåll.

Så här skriver du för att visa inställningarna för alla skrivare anslutna till ett system:

```
$ lpstat -p -l
```

Skrivarhanteringen visar en tabell med all information om inställningar för varje skrivare.

I följande exempel är alla fält tomma utom fältet för skrivaren tryckarens typ av innehåll och skrivartyp.

```
$ lpstat -p tryckaren -l
skrivaren tryckaren är sysslolös. aktiverad sedan onsdag, 2
januari kl 18:20:10 MET 1994. tillgänglig.
  Innehållstyp: PS
  Skrivartyp:   PS
  Beskrivning:
  Tillåtna användare:
                (alla)
  Tillåtna blanketter:
                (ingen)
  Banner krävs inte
  Teckensnitt:
                (inget)
  Standard för täthet:
  Standard för sidstorlek:

$
```

Översiktlig tabell över tillägg till kommandot `lpstat`

Du kan begära olika typer av statusinformation för skrivare med kommandot `lpstat`. Tabell 8-2 ger en översikt över de vanligaste tilläggen för kommandot. Använd tilläggen var för sig eller kombinera dem i vilken ordning som helst på kommandoraden. När du kombinerar tillägg skriver du ett mellanslag mellan varje tillägg. Det ska stå ett bindestreck (-) framför varje tillägg.

Om du till exempel vill visa en lång statuslista för en enskild skrivare skriver du så här:

```
$ lpstat -p skrivarnamn -l
```

där *skrivarnamn* är den skrivare vars status du vill kontrollera.

Tabell 8-2 Översikt av vanliga tillägg till kommandot `lpstat`.

Tillägg	Beskrivning
-a	Accept. Visar om en skrivare tar emot utskrifter.
-c	Class. Klasser och deras medlemmar visas.
-d	Destination. Standardskrivare visas.
-f	Forms. Blanketter visas.
-o	Output. Utmatningens status visas.
-p [<i>lista</i>][-D][-l]	Printer/Description/Long list. Skrivarnas status visas.
-r	Request. Köhanterarens status visas.
-R	Utskriftens plats i kön visas.
-s	Status. Statusöversikt visas.
-S	Sets. Teckenuppsättningar visas.
-u [<i>användarnamn</i>]	User. En användares utskrifter visas.
-v	Enheter visas.

I *man Pages(1): User Commands* finns en fullständig lista över tillägg.

Avbryta en utskrift

Använd kommandot `cancel` för att ta bort en utskrift ur skrivarkön eller avbryta en utskrift. För att göra detta måste du ange utskriftens id. Utskriftens id innehåller alltid namnet på skrivaren, ett bindestreck och utskriftens nummer. När du sänder utskriften till skrivaren visas dess id. Om du inte kommer ihåg det skriver du `lpstat` och trycker på Return. Bara den som sände utskriften till skrivaren eller den som är inloggad som `root` eller `lp` kan avbryta en utskrift.

Avbryta en utskrift genom att ange id-nummer

Så här avbryter du en utskrift:

```
$ cancel utskrifts-id
```

där *utskrifts-id* är utskriftens id-nummer.

Ett meddelande visas som talar om att utskriften är avbruten. Nästa utskrift i kön påbörjas.

I följande exempel avbryts två utskrifter:

```
$ cancel pinecone-3 pinecone-4
utskriften "pinecone-3" har avbrutits
utskriften "pinecone-4" har avbrutits
$
```

Avbryta en utskrift genom att ange skrivarnamn

Du kan också avbryta den utskrift som just pågår genom att ange skrivarens namn i stället för utskriftens id:

```
$ cancel skrivarnamn
```

där *skrivarnamn* är namnet på den skrivare dit du sändt din utskrift.

Ett meddelande visas som talar om att utskriften är avbruten. Nästa utskrift i kön påbörjas.

I följande exempel har en pågående utskrift avbrutits:

```
$ cancel tryckaren  
utskriften "tryckaren-3" har avbrutits  
$
```

Systemadministratören kan logga in som `root` eller `lp` och avbryta en pågående utskrift genom att använda skrivarnamnet som argument för kommandot `cancel`.

Arbeta i nätverk



Ett *nätverk* består av ett antal datorer som kopplats samman så att de kan kommunicera med varandra. Om din arbetsstation är kopplad till ett nätverk har du möjlighet att använda de resurser som finns på andra datorer i nätverket när du har loggat in. Du kan också logga in på andra arbetsstationer i nätverket eller ge kommandon som påverkar andra arbetsstationer.

Det här kapitlet beskriver följande:

- allmänna nätverksbegrepp
- hur du loggar in på en annan arbetsstation i nätverket
- hur du kopierar filer från andra arbetsstationer
- hur du utför kommandon på andra arbetsstationer
- hur du tar fram statusinformation om andra arbetsstationer

Om den dator du arbetar med inte är ansluten till ett nätverk har du kanske ingen nytta av informationen i det här kapitlet. Det kan ändå vara bra att skumma igenom kapitlet för att få en uppfattning om vilka fördelar det kan medföra att vara ansluten till ett nätverk.

Viktiga begrepp

Om två datorer är anslutna till varandra via ett nätverk kan de utbyta information. Nätverken kan vara av olika storlek. *Lokala nätverk* är nätverk med ett begränsat antal användare som sitter inom ett begränsat område, till exempel inom en byggnad eller på ett våningsplan. *Globala nätverk* är nätverk som sträcker sig över hela jorden. Mellantinget mellan dessa två typer av nätverk är *regionala nätverk*.

Ett nätverk som består av ett antal sammankopplade nätverk kallas för ett *stornätverk* eller *internet*. Din arbetsstation kan vara dels en del av ett lokalt nätverk, dels en del av ett stornätverk där andra liknande nätverk är anslutna. Eftersom du som användare normalt inte ser skillnaderna mellan ett lokalt nätverk och ett stornätverk används begreppet "nätverk" i den här handboken för alla typer av nätverk.

För att information ska kunna skickas mellan de olika datorerna i ett nätverk används ett *nätverksprotokoll* som ser till att informationen överförs på rätt sätt. Ett speciellt protokoll behövs för att koppla samman flera lokala nätverk till ett stornätverk.

Logga in på en annan arbetsstation (rlogin)

Med kommandot `rlogin` kan du logga in på andra arbetsstationer i nätverket.

Om du vill logga in på en annan arbetsstation i nätverket från din egen arbetsstation skriver du följande kommando:

```
$ rlogin arbetsstation
```

arbetsstation är namnet på den andra arbetsstationen.

Om en ledtext med en uppmaning att ange ett lösenord visas skriver du lösenordet för den andra arbetsstationen och trycker på Return. Om din arbetsstations namn finns i filen /etc/hosts.equiv på den andra arbetsstationen behöver du inte ange något lösenord.

```
$ rlogin gul
Password: (skriv lösenord)
Last login: mån jan 6 09:37:55 from svart
Sun Microsystems, Inc. SunOS 5.1 October 1992
(Följande kommandon utförs på gul.)
$ pwd
/home/janneb
$ logout
Connection closed.
$
```

Logga in på en annan arbetsstation utan att ha en hemkatalog

I exemplet ovan loggade användaren janneb in på arbetsstationen gul i katalogen /home/janneb. Kommandot pwd visade att katalogen /home/janneb var aktuell. Om du loggar in på en arbetsstation där du inte har en hemkatalog visas ett meddelande om detta och du hamnar i rotkatalogen (/).

```
$ rlogin gul
Password:
No directory! Logging in with home=/
Last login: Fri Jan 3 10:21:59 from svart
Sun Microsystems, Inc. SunOS 5.1 October 1992
(Följande kommandon utförs på gul.)
$ pwd
/
$ logout
Connection closed.
$
```

Använda `rlogin` för att logga in med någon annans användarnamn

Ibland kan du behöva logga in på en annan arbetsstation med någon annans användarnamn. Då kan du använda `rlogin` med tillägget `-l`. Detta gäller också om du arbetar vid någon annans arbetsstation och vill logga in med ditt användarnamn på din egen arbetsstation. Kommandot har följande syntax:

```
rlogin arbetsstation -l användarnamn
```

I följande exempel loggar användaren `janneb` vid arbetsstationen `svart` in på arbetsstationen `gul` som `jennyl`.

```
$ rlogin gul -l jennyl
Password:
Last login: Wed Jan 8 07:12:25 from svart
Sun Microsystems, Inc. SunOS 5.1 October 1992
(Följande kommandon utförs på gul.)
$ pwd
/home/jennyl
$ logout
Connection closed.
$
```

Lägg märke till att du hamnar i den andra personens hemkatalog efter inloggningen.

Använda `rlogin` för att logga in på en arbetsstation vars namn du inte känner till

Om du försöker logga in på en arbetsstation vars namn inte finns i värddatorns databas visas följande meddelande:

```
$ rlogin brun
brun: unknown host
$
```

Logga ut från en arbetsstation som anslutits med `rlogin`

Normalt loggar du ut från en arbetsstation som anslutits med `rlogin` genom att skriva `logout` när du är klar med arbetet vid den andra arbetsstationen. Om du av någon anledning inte kan logga ut på detta sätt kan du avbryta förbindelsen genom att skriva `~.` i början av en ny rad. Då avbryts förbindelsen och du återkommer till din egen arbetsstation.

Om du har loggat in på en eller flera arbetsstationer genom att gå via en annan arbetsstation kan du logga ut från alla arbetsstationer genom att använda `~..`. Då återkommer du till din egen arbetsstation:

```
$ rlogin uppland
Password:
Last login: Fri Jan 10 09:14:43 from gotland
Sun Microsystems, Inc. SunOS 5.1 October 1992
(Följande kommando utförs på uppland.)
$ ~. (Det är inte säkert att ~-tecknet visas på skärmen.)
Connection closed.
$
```

Om du vill återgå till en mellanliggande `rlogin`-förbindelse skriver du två `~`-tecken följt av en punkt (`~~.`) enligt följande:

```
$ rlogin uppland
Password:
Last login: Tue Jan 7 08:12:49 from gotland
Sun Microsystems, Inc. SunOS 5.1 October 1992
(Följande kommando utförs på uppland.)
$ rlogin dalsland
Password:
Last login: Tue Jan 7 10:17:40 from uppland
Sun Microsystems, Inc. SunOS 5.1 October 1992
(Följande kommando utförs på dalsland.)
$ ~~. (Det är inte säkert att ~-tecknen visas på skärmen.)
Connection closed.
$
```

Göra en paus i en rlogin-förbindelse

Om du tillfälligt vill göra en paus i en rlogin-förbindelse skriver du ett ~-tecken och trycker på Ctrl-Z. rlogin-processen stoppas och du återkommer till den arbetsstation där du loggade in.

När du vill återuppta förbindelsen skriver du fg. Du kan också skriva ett procenttecken (%) följt av den stoppade processens nummer. (Om du inte anger något nummer återupptas den senast stoppade processen.)

```

$ rlogin gul
Password:
Last login: Tue Jan 7 08:12:49 from svart
Sun Microsystems, Inc. SunOS 5.1 October 1992
(Följande kommando utförs på gul.)
~^Z (Det är inte säkert att ^Z visas på skärmen.)
Stopped
(Följande kommandon utförs på svart.)
$ pwd
/home/janneb
$ %
rlogin gul

(Följande kommando utförs på gul.)
$ logout
Connection closed.
$

```

Om du skriver två ~-tecken och trycker på Ctrl-Z görs en paus i den aktuella rlogin-förbindelsen och du återkommer till föregående rlogin-förbindelse.

Kontrollera vilken arbetsstation du arbetar med (whoami)

Om du har loggat in på flera arbetsstationer i nätverket från din arbetsstation och använt olika användarnamn kanske du behöver kontrollera vilken arbetsstation du är inloggad på. Om du skriver whoami visas namnet på den arbetsstation du är inloggad på och det användarnamn du loggat in med.

Om du vill ha mer information om kommandot rlogin kan du skriva man rlogin vid kommandoledtexten eller läsa handboken *man Pages(1): User Commands*.

Kopiera filer mellan olika arbetsstationer (r_{cp})

Med kommandot `rcp` kan du kopiera filer från en arbetsstation till en annan. Din behörighet kontrolleras i filerna `/etc/hosts.equiv` och `/etc/passwd` på den andra arbetsstationen. Syntaxen för `rcp` liknar den för `cp`.

Observera – Om du ska kopiera underkataloger och deras innehåll från en arbetsstation till en annan ska du använda kommandot `rcp -r`.

Kopiera från en annan arbetsstation till din egen

Om du vill kopiera filer från en annan arbetsstation till din egen skriver du följande kommando:

```
rcp arbetsstation:källa mål
```

arbetsstation är namnet på den andra arbetsstationen, *källa* är namnet på den eller de filer du vill kopiera och *mål* är sökvägen till den katalog på din arbetsstation där filerna ska placeras.

I följande exempel kopieras filen `/home/dalsland/dok/brev` från arbetsstationen `dalsland` till katalogen `/tmp` på den egna arbetsstationen (svart):

```
$ rcp dalsland:/home/dalsland/dok/brev /tmp
$
```

Du kan också använda förkortningar och jokertecken när du använder `rcp`. Om du till exempel vill kopiera alla filer vars namn slutar med `.dok` från hemkatalogen (`~`) för användaren `knuts` på arbetsstationen `gul` till den aktuella katalogen (`.`) på din egen arbetsstation (svart) skriver du följande kommando:

```
~$ rcp gul:~knuts/*.dok .
~$
```

Kopiera från din egen arbetsstation till en annan

Om du vill kopiera filer från din egen arbetsstation till en annan skriver du följande kommando:

```
rcp källa arbetsstation:mål
```

källa är den eller de filer du vill kopiera, *arbetsstation* är namnet på den andra arbetsstationen och *mål* är sökvägen till den katalog på den andra arbetsstationen där filerna ska placeras.

I följande exempel kopieras filen *visby* från katalogen `~/landskap/gotland` på din arbetsstation till katalogen `~sturet/städer` på arbetsstationen *gul* (kom ihåg att `~` är din hemkatalog och att `~sturet` är hemkatalogen för användaren *sturet*):

```
$ rcp ~/landskap/gotland/visby gul:~sturet/städer
$
```

Du kan läsa mer om kommandot `rcp` och dess tillägg i *man Pages(1): User Commands*.

Ge kommandon på andra arbetsstationer (r_{sh})

Med kommandot `rsh` kan du ge kommandon som utförs på en annan arbetsstation i nätverket utan att behöva logga in på den arbetsstationen. Detta kan spara tid om du vet att du bara ska göra en sak på den andra arbetsstationen.

Skriv följande kommando om du vill ge ett kommando på en annan arbetsstation i nätverket:

```
rsh arbetsstation kommando
```

Exemplet nedan visar hur du kan kontrollera innehållet i katalogen `/home/labanh/brev` på arbetsstationen *gul*:

```
$ rsh gul ls /home/labanh/brev
lisa          kalle          vera
gunilla      martin         sven
$
```


Precis som när du använder kommandona `rlogin` och `rcp` kontrolleras din behörighet i filerna `/etc/hosts.equiv` och `/etc/passwd` när du använder kommandot `rsh`.

Du kan läsa mer om kommandot `rsh` och dess tillägg i *man Pages(1): User Commands*.

Kontrollera användarinformation (`rusers`)

Med kommandot `rusers` kan du kontrollera vilka användare som loggat in på andra arbetsstationer i nätverket. Om du skriver kommandot utan några tillägg visas en lista över arbetsstationerna i nätverket och vilka som loggat in på dem. Listan kan se ut så här:

```
$ rusers
gul          susannep
svart        janneb
brun         alicer
lila         stens
dalsland     lindab      johnj      karinam
gotland      kallep
norrbotten   georgt
uppland      tryggveo
$
```

Lägg märke till att tre olika användare har loggat in på arbetsstationen `dalsland`.

Om du vill ta fram information om en viss arbetsstation skriver du kommandot `rusers` följt av namnet på arbetsstationen enligt följande:

```
$ rusers dalsland
dalsland     lindab      johnj      karinam
$
```

Om du använder tillägget `-l` visas mer detaljerad information. Förutom arbetsstationens namn och användarnamnet visas även terminalnamnet, tidpunkten då användaren loggade in, hur lång tid varje användare arbetat (om tiden överstiger en minut) och namnet på den arbetsstation användaren loggade in från:

```
$ rusers -l dalsland
lindab      dalsland:ttyd8      feb 10 08:12      5:29
johnj       dalsland:console    feb 10 09:16
karinam     dalsland:ttyp0      feb 10 11:56      36 (gotland)
$
```

Du kan också använda tillägget `-l` utan att ange namnet på en arbetsstation.

Du kan läsa mer om kommandot `rusers` och dess tillägg i *man Pages(1): User Commands*.

Anpassa din arbetsmiljö

10 

Med operativsystemet SunOS har du stora möjligheter att anpassa din arbetsmiljö. Du kan ändra ett stort antal *omgivningsvariabler* som finns i systemets *startfiler*. När du loggar in hämtas informationen i startfilerna och omgivningsvariablernas värden används för att anpassa systemet. Genom att ställa in omgivningsvariablerna kan du anpassa systemet så att det passar dina behov bättre och blir lättare att arbeta med.

Det här kapitlet beskriver hur du ändrar startfilerna och ställer in de vanligaste omgivningsvariablerna. Här beskrivs också hur du skapar aliasnamn för SunOS-kommandon, ändrar kommandoledtexten, ställer in standardbehörighet för filer och väljer teckensnitt för OpenWindows.

Startfiler

Vilka startfiler som används för att anpassa systemet beror på vilket skal systemadministratören angav som standardskal när systemet installerades. Bourne-skalet är standardskalet i SunOS, men du kan också använda C-skalet eller Korn-skalet. De olika skalerna använder sig av olika startfiler.

Om du inte vet vilket skal som är standardskal (eller *inloggningsskal*) gör du så här:

1. Skriv kommandot `echo $SHELL` och tryck på Return:

```
$ echo $SHELL
/bin/sh
```

2. Titta på resultatet av kommandot.

- `/bin/sh` – Bourne-skalet är inloggningsskal
- `/bin/csh` – C-skalet är inloggningsskal
- `/bin/ksh` – Korn-skalet är inloggningsskal

Oavsett vilket skal du använder körs normalt filen `/etc/profile`. Den filen ägs normalt av systemadministratören, men alla användare har behörighet att läsa innehållet i den.

När filen `/etc/profile` körts läses en eller flera *inställningsfiler för användaren* in. Dessa filer innehåller inställningar för användarens arbetsmiljö. Om du till exempel använder OpenWindows kontrolleras inställningsfilerna varje gång du öppnar ett nytt skal- eller kommandofönster.

Följande filer innehåller inställningar för användaren:

- `.profile` (för Bourne- och Korn-skalen)
- `.login` och `.cshrc` (för C-skalet)

Inställningsfilerna är placerade i din hemkatalog och med hjälp av dem kan du anpassa din arbetsmiljö så att den passar dina behov.

Omgivningsvariabler

I start- och inställningsfilerna finns ett antal omgivningsvariabler som anpassar din arbetsmiljö. Om du vill ändra dessa inställningar tillfälligt kan du ge kommandon vid kommandoledtexten. Vill du ändra inställningarna permanent ändrar du i stället filerna `.profile`, `.login` och `.cshrc`.

Gör så här om du vill ta fram en lista över omgivningsvariablernas aktuella inställningar:

1. Skriv kommandot `env` och tryck på Return:

```
$ env
HISTORY=100
HOME=/home/dalsland/janneb
HZ=100
LOGNAME=janneb
MAIL=/var/mail/janneb
MANSECTS=\1:1m:1c:1f:1s:1b:2:\3:3c:3i:3n:3m:3k:3g:3e:3x11:3xt:3
w:3b:9:4:5:7:8
PATH=/usr/bin
SHELL=/bin/sh
TERM=sun
TZ=EST5EDT
```

Observera – Du kan också använda kommandot `env` för att ta reda på vilket inloggningsskal som används. Detta visas efter variabeln `SHELL`. I exemplet ovan används `/bin/sh` (Bourne-skalet) som inloggningsskal.

Användarens inställningsfiler

Det här avsnittet beskriver de vanligaste omgivningsvariablerna. Flera av dem finns kanske redan i dina inställningsfiler. Som tidigare nämnts finns inställningsfilerna `.profile` för Bourne- och Korn-skalen och `.cshrc` för C-skalet) i din hemkatalog.

Observera – Du tar fram en lista över de dolda punktfilerna med kommandot `ls -la`.

I följande lista visas de vanligaste omgivningsvariablerna som kan infogas i inställningsfilerna. Omgivningsvariablernas syntax skiljer sig åt mellan olika skal.

- `CDPATH` – Anger vilka kataloger som ska sökas igenom om du anger namnet på en katalog utan att ange den fullständiga sökvägen.

- HISTORY – Anger hur många kommandon som ska vara tillgängliga med kommandot `history` (gäller bara C-skalet).
- HOME – Anger sökvägen till din hemkatalog. Om du anger kommandot `cd` utan argument kommer du till den katalog som anges här.
- LANG – Anger vilket språk som används. Du kan välja mellan engelska, japanska, tyska, franska, svenska och italienska.
- LOGNAME – Definierar användarnamnet. Standardvärdet för denna variabel är det användarnamn som specificerats i databasen `passwd` vid inloggningen. Du kan läsa mer om databasen `passwd` i *User Accounts, Printers, and Mail Administration*.
- LPDEST – Anger vilken skrivare som ska vara standardskrivare.
- MAIL – Anger sökvägen till din inkorg för post. Den finns normalt i katalogen `/var/mail/ användarnamn`, där *användarnamn* är ditt namn för inloggning. Du kan läsa mer om denna fil i kapitel 7 "Elektronisk post".
- MANSECTS – Anger vilka sektioner i direkthjälpen (`man`) som ska vara tillgängliga.
- PATH – Anger vilka kataloger som ska sökas igenom efter program när du ger ett kommando. Katalogerna anges i den ordning de ska sökas igenom. Om du ger ett kommando och katalogen där kommandot finns inte angetts i sökvägen kan inte kommandot utföras. Du måste då antingen ange den fullständiga sökvägen till kommandot eller gå till rätt katalog först.

Standardvärdet för den här variabeln definieras i filen `.profile` (Bourne- eller Korn-skalet) eller `.cshrc` (C-skalet) vid inloggningen.

- PS1 – Anger hur kommandoledtexten ska se ut. Standardledtexten för Bourne- och Korn-skalen är ett dollartecken (`$`) och för C-skalet ett procenttecken (`%`). Standardledtexten för superanvändare är ett nummertecken (`#`) i alla skalen.
- SHELL – Anger vilket skal som används av `vi` och andra verktyg.
- TERMINFO – Anger sökvägen till en terminal som inte stöds och som har lagts till databasen `terminfo`. Du behöver inte ställa in denna variabel om din terminal finns med i databasen. Du kan läsa mer om databasen `terminfo` i *User Accounts, Printers, and Mail Administration*.

- **TERM** – Anger vilken terminal som används. När du startar ett redigeringsprogram letar systemet efter en fil med det namn som anges med denna variabel. Filen innehåller information om den aktuella terminalen. Först söks de kataloger igenom som angetts med variabeln **TERMINFO** och sedan söks standardkatalogen `/usr/share/lib/terminfo` igenom. Om det inte finns någon fil med information om terminalen definieras terminalen som "dum", det vill säga som en terminal som bara kan användas för in- och utmatning.
- **TZ** – Anger tidszonen för systemets klocka.

Ställa in variabeln **PATH**

Omgivningsvariabeln **PATH** används för att lokalisera kommandon i katalogstrukturen i SunOS. Med ett värde för **PATH** kan du ange vika kataloger som systemet alltid ska söka igenom när du ger ett kommando.

Om du inte angett någon sökväg med kommandot **PATH** och vill kopiera en fil måste du ange hela sökvägen till kommandot: `/usr/bin/cp`. Om du däremot har angett sökvägen `/usr/bin` som värde för variabeln **PATH** behöver du bara skriva `cp`. Systemet letar nämligen efter kommandot `cp` i alla kataloger som angetts med variabeln **PATH** och utför kommandot när det hittas. Genom att använda variabeln **PATH** och ange de kataloger där de vanligaste SunOS-kommandona finns kan du förenkla arbetet vid datorn betydligt.

Om du använder Bourne- eller Korn-skalet definieras variabeln **PATH** i filen `.profile` i hemkatalogen. Kommandot har följande syntax:

```
PATH=.: /usr/bin: /hemkatalog/bin
```

hemkatalog är sökvägen till din hemkatalog.

Om du använder C-skalet definieras variabeln **PATH** i filen `.cshrc` i hemkatalogen. Kommandot har följande syntax:

```
set path=(. /usr/bin hemkatalog/bin)
```

hemkatalog är sökvägen till din hemkatalog.

Observera – I C-skalet kan du använda tecknet ~ för att ange sökvägen till hemkatalogen.

Om du använder C-skalet och ändrar variabeln `PATH` kan du använda kommandot `source` för att läsa in de nya inställningarna i det aktuella fönstret. Då behöver du inte logga ut:

```
exempel% source .cshrc
```

Om du använder Bourne- eller Korn-skalet kan du använda följande kommando för att läsa in de nya inställningarna i det aktuella fönstret. Då behöver du inte logga ut:

```
$ . .profile
```

Aliasnamn (gäller bara C-skalet)

Med hjälp av *aliasnamn* kan du förkorta kommandon som du använder ofta. Ett exempel: standardinställningen för kommandot `rm` är att ingen kontrollfråga visas innan filerna eller katalogerna tas bort. För att kontrollfrågan ska visas måste du använda tillägget `-i`. För att minska risken att ta bort filer av misstag kan du i C-skalet skapa ett aliasnamn för kommandot `rm -i` genom att lägga till följande rad i filen `.cshrc`:

```
alias rm 'rm -i'
```

Om denna rad finns med i filen `.cshrc` kommer kommandot `rm` att motsvara `rm -i`, vilket innebär att en kontrollfråga visas för varje fil som ska tas bort. (Etersom det är ett mellanslag mellan `rm` och `-i` måste kommandot skrivas inom enkla citationstecknen, annars kan inte C-skalet tolka texten efter mellanslaget.)

Om du vill att ändringarna ska träda i kraft omedelbart i det aktuella fönstret kan du använda kommandot `source`. Då läses filen `.cshrc` in och kommandona i den utförs:

```
exempel% source .cshrc
```

Ändra kommandoledtexten

Hur du gör för att ändra systemets kommandoledtext beror på om du använder Bourne-, Korn- eller C-skalet.

Bourne- och Korn-skalen

Om du använder Bourne- eller Korn-skalet kan du ändra kommandoledtexten med kommandot `PS1`. Här är tre exempel på hur detta kommando kan användas:

```
PS1=" : "  
PS1="värddator: "  
PS1="värddator{användarnamn}": "
```

- Det första kommandot ändrar kommandoledtexten till ett kolon (:) följt av ett mellanslag.
- Det andra kommandot skapar en kommandoledtext som består av namnet på värddatorn följt av ett kolon och ett mellanslag.
- Det tredje kommandot skapar en kommandoledtext som består av namnet på värddatorn följt av användarnamnet inom klamrar, ett kolon och ett mellanslag.

Prova gärna kommandona ovan och ändra kommandoledtexten. Ändringen kvarstår tills du ändrar ledtexten igen eller loggar ut.

Om du vill ändra kommandoledtexten permanent lägger du till kommandot i filen `.profile` i din hemkatalog. Då ändras ledtexten varje gång du loggar in eller startar ett nytt skal.

C-skalet

Om du använder C-skalet ändrar du kommandoledtexten med kommandot `set prompt`. Här är tre exempel på hur detta kommando kan användas:

```
set prompt="% "
set prompt="värddator\!: "
set prompt="värddator{användarnamn} }: "
```

- Det första kommandot ändrar kommandoledtexten till ett procenttecken (%) följt av ett mellanslag.
- Det andra kommandot skapar en kommandoledtext som består av namnet på värddatorn följt av kommandots ordningsnummer (värddator1, värddator2, värddator3 och så vidare).
- Det tredje kommandot skapar en kommandoledtext som består av namnet på värddatorn följt av användarnamnet inom klamrar, ett kolon och ett mellanslag.

Prova gärna kommandona ovan och ändra kommandoledtexten. Ändringen kvarstår tills du ändrar ledtexten igen eller loggar ut.

Om du vill ändra kommandoledtexten permanent lägger du till kommandot i filen `.cshrc` i din hemkatalog. Då ändras ledtexten varje gång du loggar in eller startar ett nytt skal.

Andra användbara variabler

Utöver ovanstående variabler finns det ett stort antal variabler som du kan ange i filen `.profile` eller `.cshrc`. I *man Pages(1): User Commands* finns en fullständig förteckning över dessa variabler. I det här avsnittet beskrivs några av de vanligaste variablerna.

Med variabeln `noclobber` kan du förhindra att filer oavsiktligt ersätts när du använder kommandot `cp` för att kopiera filer. Denna variabel kan bara användas i C-skalet. Lägg till följande kommando i filen `.cshrc`:

```
set noclobber
```

Med variabeln `history` kan du ange hur många kommandon som ska sparas i listan över använda kommandon. (Du kan sedan återanvända dessa kommandon.) Denna variabel kan bara användas i C-skalet. Lägg till följande kommando i filen `.cshrc`:

```
set history=100
```

Om du använder Bourne- eller Korn-skalet skriver du följande kommando i filen `.profile`:

```
HISTORY=100
```

Ställa in standardbehörighet till filer och kataloger

Med kommandot `umask` ställer du in standardbehörighet till de filer och kataloger du skapar. Om du till exempel inte vill att gruppmedlemmar och andra användare ska ha behörighet att ändra dina filer (`-rwxr-xr-x`) kan du ange detta i filen `.cshrc` eller `.profile`. Då ställs denna behörighet in för alla filer och kataloger du skapar.

Precis som när du använder kommandot `chmod` anger du behörigheten med en sifferkod. Sifferkoden för `umask` skiljer sig dock från den för `chmod`.

Om du använder sifferkoden 000 i kommandot `umask` kommer alla filer du skapar att få följande behörighet, vilket innebär att alla användare har behörighet att läsa och ändra men inte använda (köra) filerna:

```
rw-rw-rw- (koden 666 för kommandot chmod)
```

Alla kataloger kommer att få följande behörighet (alla användare har behörighet att läsa, ändra och använda katalogerna):

```
rwxrwxrwx (koden 777 för kommandot chmod)
```

För att beräkna sifferkoden för `umask` subtraherar du `chmod`-koden för den behörighet du vill ställa in från `chmod`-koden för den aktuella behörigheten. Resultatet är det värde du ska ange i kommandot `umask`.

Om du till exempel vill ändra standardbehörigheten från 666 (*rw-rw-rw-*) till 644 (*rw-r--r--*) subtraherar du 644 från 666. Resultatet, 022, är det värde du ska ange i kommandot `umask`.

```
umask 022
```

Siffrorna representerar ägaren, gruppen och andra användare, precis som för kommandot `chmod`.

- Den första siffran styr ägarens behörighet
- Den andra siffran styr behörigheten för gruppen
- Den första siffran styr behörigheten för alla andra användare

Tabell 10-1 visar vilken behörighet de olika siffrorna i sifferkoden motsvarar.

Tabell 10-1 Behörighetskoder för `umask`

umask-kod	Behörighet
0	<i>rwX</i>
1	<i>rw-</i>
2	<i>r-x</i>
3	<i>r--</i>
4	<i>-wX</i>
5	<i>-w-</i>
6	<i>--X</i>
7	<i>---</i> (ingen)

Du kan läsa mer om kommandot `umask` i *man Pages(1): User Commands*.

Välja teckensnitt för *OpenWindows*

Om du vill kan du ändra teckensnitt, storlek och stil för *OpenWindows*-programmen. De följande avsnitten beskriver hur du gör det.

Välja teckensnitt och storlek

Standardteckensnittet för text i OpenWindows-fönster är Lucida Sans, 12 punkter (medium), och standardteckensnittet för namnlistor är Lucida Sans Bold. Om du vill kan du välja andra teckensnitt och storlekar för text i fönster och namnlistor. Du kan ändra teckensnittet för ett enskilt fönster eller för samtliga program. Om du vill ändra teckensnittet för samtliga OpenWindows-program använder du fönstret Egenskaper för arbetsytan.

Teckensnitt med fast och proportionell teckenbredd

Teckensnitt brukar delas in i två grupper beroende på om de har *fast* eller *proportionell teckenbredd*. Om ett teckensnitt har fast teckenbredd tar alla tecken upp lika stor plats, medan tecknen i ett teckensnitt med proportionell teckenbredd tar upp olika stor plats beroende på om tecknet är brett eller smalt. Teckensnitt med proportionell teckenbredd är mer lättlästa. Vissa program (till exempel kommandofönstret, skalfönstret eller `xterm`, ett vanligt program för terminalemulering) fungerar dock bäst om man väljer ett teckensnitt med fast teckenbredd.

Välja mellan teckensnitt med fast och proportionell teckenbredd

Standardteckensnittet för kommandofönstret och skalfönstret är ett teckensnitt med proportionell teckenbredd. Även om detta teckensnitt är lättläst kan det uppstå problem med teckenjustering (när du gör mellanslag och använder tabbar) i ett skal- eller kommandofönster. Om sådana problem uppstår bör du välja ett teckensnitt med fast teckenbredd i fönstret. I de exempel som följer används teckensnitt med fast teckenbredd i skal- och kommandofönster och teckensnitt med proportionell teckenbredd i andra fönster och i namnlistor.

Välja teckensnitt för enskilda fönster

Det här avsnittet beskriver hur du startar ett program och väljer teckensnitt, stil och storlek för programmets fönster. Observera att du inte kan ändra teckensnittet i ett befintligt fönster – du måste starta om programmet för att se det nya teckensnittet. Du startar ett program genom att skriva programmets namn på kommandoraden.

Kommandot för att starta ett program och ange ett teckensnitt visas nedan. Förutom programnamnet anger du även tillägget `-fn`, teckensnittets namn och teckenstorleken. Tecknet `&` gör att systemets ledtext visas i fönstret och att du kan fortsätta att ge kommandon efter det att det första kommandot utförts.

```
$ program -fn teckensnitt-storlek &
```

Här följer ett antal exempel på hur du kan starta program och välja teckensnitt och storlek för programmets fönster.

- Följande kommando öppnar ett nytt kommandofönster med ett teckensnitt med proportionell teckenbredd: Lucida Sans Typewriter Bold. (Hur teckensnittets namn ska skrivas beskrivs i avsnittet "Ta fram en lista över tillgängliga teckensnitt" längre fram i det här kapitlet.

Eftersom ingen teckenstorlek anges används standardstorleken (12 punkter).

```
$ cmdtool -fn lucidasans-typewriter-bold &
```

- Följande kommando öppnar ett nytt skalfönster med teckensnittet Lucida Sans Typewriter Bold. Teckenstorleken ökas till 14 punkter.

Lägg märke till att fönstrets storlek ändras när du ändrar teckenstorleken.

```
$ shelltool -fn lucidasans-typewriter-bold-14 &
```

- Följande kommando öppnar ett nytt `xterm`-fönster med teckensnittet terminal-bold, 16 punkter:

```
$ xterm -fn terminal-bold-16 &
```

- Följande kommando startar ett nytt textverktyg med teckensnittet Helvetica Bold, 14 punkter:

```
$ textedit -fn helvetica-bold-14 &
```

Du kan använda tillägget `-fn` och ange valfritt teckensnitt och storlek. I avsnittet "Listan över tillgängliga teckensnitt" längre fram i det här kapitlet kan du läsa om hur du tar fram en lista över tillgängliga teckensnitt i OpenWindows.

Ändra teckensnitt permanent

Om du ofta väljer ett speciellt teckensnitt för ett visst program kan du ställa in OpenWindows så att teckensnittet väljs automatiskt. Det gör du genom att öppna fönstret Egenskaper för arbetsytan, välja kategorin Menyn Program och lägga till ett nytt, anpassat menyalternativ på menyn Arbetsyta. Då slipper du skriva ett långt kommando med tillägg för teckensnittet varje gång du vill starta programmet. Om du till exempel ofta vill starta textverktyget med en större teckenstorlek än den normala kan du lägga till följande kommando på menyn Program:

```
textedit -fn lucidasans-typewriter-14
```

Om du vill ha möjlighet att starta ett program med olika teckensnitt och storlekar kan du lägga till flera menyalternativ för programmet på menyn Program. Du kanske vill ha möjlighet att starta textverktyget med tre olika teckenstorlekar, 12, 14 eller 18 punkter. Lägg då till följande kommandon på menyn Program:

```
textedit -fn lucidasans-typewriter-12  
textedit -fn lucidasans-typewriter-14  
textedit -fn lucidasans-typewriter-18
```

När du har anpassat menyn Program på detta sätt kan du starta textverktyget med en av ovanstående teckenstorlekar genom att välja önskat alternativ på menyn.

Observera – Du bör inte avsluta de kommandon som du lägger till på menyn Program med ett `&`-tecken.

Ta fram en lista över tillgängliga teckensnitt

Om du vill prova några andra teckensnitt än de som använts i exemplen ovan kan du ta fram en lista över tillgängliga teckensnitt och storlekar.

Listan över tillgängliga teckensnitt

Du tar fram listan över tillgängliga teckensnitt genom att ge kommandot `xlsfonts` vid ledtexten i ett skal- eller kommandofönster. Det är bäst att använda kommandofönstret, eftersom du då kan rulla innehållet i fönstret och på så sätt se hela listan – den är nämligen mycket lång.

Observera – Listan som visas när du ger kommandot `xlsfonts` är mycket lång. Totalt finns det ungefär 400 teckensnitt. Om din lista inte innehåller så många teckensnitt ska du tala med systemadministratören. Eventuellt har inte alla teckensnitt installerats i systemet.

Varje teckensnitt har ett långt och ett kort namn. Det här är till exempel det långa namnet på teckensnittet Lucida Sans Typewriter:

```
-b&h-lucida sans typewriter-medium-r-normal-sans-12-120-72-72-m-0-iso8859-1
```

I `xlsfonts`-listan visas först teckensnittets långa namn och sedan det korta. Använd det korta namnet när du ska välja teckensnitt för fönster i `OpenWindows`.

När du har bestämt vilket teckensnitt du vill använda följer du instruktionerna i avsnittet "Välja teckensnitt och storlek" på sidan 155 för att starta ett program och välja teckensnittet.

Byta till OpenWindows 3.3 eller senare version



Användarmiljön i Solaris är OpenWindows. Om du använder äldre version av OpenWindows än 3.3, till exempel OpenWindows 2, eller någon annan användarmiljö, till exempel SunView, kanske du vill byta till OpenWindows 3.3. Den här bilagan beskriver hur du gör det.

Observera – Till skillnad från tidigare versioner av OpenWindows finns det i OpenWindows 3.3 inte något stöd för SunView. Om du byter till OpenWindows 3.3 har du inte längre någon möjlighet att använda SunView.

SPARC *Byta till OpenWindows 3.3 från SunView*

Om du byter till OpenWindows från SunView ska du läsa igenom detta avsnitt noga.

SPARC *Filerna .defaults och .Xdefaults*

Om du vill använda samma inställningar i OpenWindows som du använt i SunView kan du konvertera filen `.defaults` i din hemkatalog till en `.Xdefaults`-fil (som används i OpenWindows). Om filen `.defaults` finns i din hemkatalog ska du använda programmet `convert_to_Xdefaults(1)` för att konvertera den.

Om du vill konvertera filen `.defaults` ger du följande kommando:

```
$ cd
$ $OPENWINHOME/bin/convert_to_Xdefaults .defaults
```

Då skapas filen `.Xdefaults` i din hemkatalog. Denna fil används för att anpassa OpenWindows-miljön när du startar programmet.

Byta till OpenWindows 3.3 från tidigare versioner

Om du använder en tidigare version av OpenWindows än 3.3 och vill byta till version 3.3, eller senare, ska du läsa det här avsnittet noga. Avsnittet handlar framför allt om att byta från OpenWindows 2.

Omgivningsvariabeln `OPENWINHOME`

Om du använder en tidigare version av OpenWindows än 3.3 kanske du har använt omgivningsvariabeln `OPENWINHOME`. Vi rekommenderar att du inte använder denna variabel i OpenWindows 3.3 eller senare version, varken i startfiler eller kommandon.

När du ger kommandot `openwin` för att starta OpenWindows får variabeln `OPENWINHOME` automatiskt värdet `/usr/openwin`. Du behöver därför inte tilldela variabeln ett värde.

Om variabeln `OPENWINHOME` tilldelas ett värde med hjälp av någon av filerna `.profile` eller `.cshrc` i din hemkatalog ska du ta bort den rad där detta görs eller göra den till en kommentar innan du startar OpenWindows 3.3 eller senare version. Gör så här:

1. **Öppna filen `.profile` eller `.cshrc` med ett redigeringsprogram, till exempel `vi`.**
2. **Gör den rad där variabeln står till en kommentar eller ta bort den helt.** Om du vill göra raden till en kommentar infogar du ett nummertecken (`#`) först på raden. Under punkt a nedan kan du se hur det ska se ut i filen `.profile` och under punkt b hur det ska se ut i filen `.cshrc`.

a. Så här ska raden se ut i filen `.profile`:

```
#OPENWINHOME=/usr/openwin
```

b. Så här ska raden se ut i filen `.cshrc`:

```
#setenv OPENWINHOME /usr/openwin
```

3. Spara filen och avsluta redigeringsprogrammet.

Filen `.xinitrc`

Här är några viktiga kommentarer om filerna `.xinitrc` och `$(OPENWINHOME)/lib/Xinitrc`:

1. I OpenWindows 2 skapade kommandofilen `openwin` automatiskt en kopia av `$(OPENWINHOME)/lib/Xinitrc`, placerade kopian i hemkatalogen och gav den namnet `.xinitrc`. Detta gäller inte i OpenWindows 3.3. Detta är en viktig skillnad av följande skäl:
 - a. Kommandofilen `openwin` använder standardfilen `$(OPENWINHOME)/lib/Xinitrc` om filen `.xinitrc` inte finns i hemkatalogen.
 - b. Det är viktigt att du använder den standardversion av filen `$(OPENWINHOME)/lib/Xinitrc` som levereras med OpenWindows 3.3 och senare versioner. (Längre fram i det här avsnittet får du veta hur du ska göra för att behålla eventuella ändringar som du gjort i filen `.xinitrc` för OpenWindows 2.)
2. Du behöver bara en version av fönsterhanteraren `olwm`, även om du använder flera bildskärmar.

Använda rätt startfil

Om du använder en äldre version av OpenWindows än 3.3 är det viktigt att du tar reda på om filen `.xinitrc` finns och i så fall om den modifierats. Filen `.xinitrc` är en startfil som finns i din hemkatalog och som kan innehålla användarspecifika inställningar.

Ge följande kommandon för att ta reda på om filen finns:

```
$ cd
$ ls -a .xinitrc
```

Beroende på vad resultatet blir gör du något av följande:

- Om filen `.xinitrc` inte finns (det vill säga om filens namn inte visas när du ger kommandot `ls -a`) behöver du inte göra någonting alls. Om filen `.xinitrc` inte finns i din hemkatalog används standardfilen vid start.
- Om filen finns (det vill säga om dess namn visas när du ger kommandot `ls -a`) men du inte gjort några ändringar i den eller om du inte vill spara ändringarna kan du ta bort den. Följ instruktionerna i steg 1 i avsnittet "Hantera startfilerna" nedan.
- Om filen finns (det vill säga om dess namn visas när du ger kommandot `ls -a`) och du har gjort ändringar i den följer du instruktionerna i steg 2 i avsnittet "Hantera startfilerna" nedan.

Hantera startfilerna

1. Om du vill ta bort filen `.xinitrc` från din hemkatalog ger du följande kommando:

```
$ rm .xinitrc
```

2. Om du vill behålla de ändringar du gjort i filen `.xinitrc` gör du så här:

- a. Flytta `.xinitrc` till `.xinitrc.spara`:

```
$ mv .xinitrc .xinitrc.spara
```

- b. Kopiera `$OPENWINHOME/lib/Xinitrc` till din hemkatalog och ge kopian namnet `.xinitrc`:

```
$ cp $OPENWINHOME/lib/Xinitrc $HOME/.xinitrc
```

c. **Kopiera de rader i filen `.xinitrc`. spara som du vill behålla och lägg in dem i filen `.xinitrc`.**



Varning – Lägg inte till kommandot `olwm` i filen `.xinitrc` – det finns redan där och får bara förekomma en gång i filen. Lägg inte heller till kommandot `svenv`. Du får inte heller ta bort raden med `$OPENWINHOME/lib/openwin-sys`.

Egenskaper för arbetsytan

I tidigare versioner av OpenWindows än 3.3 sparades de ändringar du gjorde i fönstret Egenskaper för arbetsytan i filen `.Xdefaults` i din hemkatalog. I OpenWindows 3.3 sparas dessa ändringar i filen `.OWdefaults` som också finns i din hemkatalog. Filen `.Xdefaults` kan fortfarande finnas kvar, men inställningarna i `.OWdefaults` har företräde framför inställningarna i `.Xdefaults`.

Filen `.Xdefaults` bör bara användas för sådana anpassningar som inte kan göras i fönstret Egenskaper för arbetsytan. Du kan till exempel göra anpassningar av program som inte körs under OpenWindows eller lägga till pre-processormakron i C. Använd ett redigeringsprogram, till exempel `vi`, för att göra dessa ändringar. Ändringar du gör i fönstret Egenskaper för arbetsytan påverkar inte inställningarna i filen `.Xdefaults`.

Även om du inte vill göra några ändringar i filen `.Xdefaults` i din hemkatalog behöver du inte ta bort den. Eftersom inställningarna i `.OWdefaults` har företräde framför inställningarna i `.Xdefaults` kan `.Xdefaults` ligga kvar utan att det gör något.

Anpassa menyn Arbetsyta

I OpenWindows 3.3 kan du använda fönstret Egenskaper för arbetsytan för att anpassa undermenyn Program på menyn Arbetsyta. I tidigare versioner av OpenWindows var du tvungen att redigera filen `.openwin-menu` i din hemkatalog för att göra detta.

Observera – Om filen `.openwin-menu` inte finns i din hemkatalog behöver du inte följa instruktionerna nedan – du kan direkt använda fönstret Egenskaper för arbetsytan för att anpassa menyn Arbetsyta.

Om filen `.openwin-menu` finns i din hemkatalog måste du följa instruktionerna nedan för att du ska kunna använda fönstret Egenskaper för arbetsytan för att anpassa menyn Arbetsyta.

Kontrollera om följande rad finns i filen `.openwin-menu`:

```
"Program"           MENU $OPENWINHOME/lib/openwin-menu-programs
```

Om raden finns tar du bort den och ersätter den med följande rad:

```
"Program"           INCLUDE openwin-menu-programs
```

Om raden inte finns infogar du ovanstående rad.

När du ersätter eller infogar denna rad läggs standardversionen av menyn Program in på menyn Arbetsyta och du kan anpassa den med hjälp av fönstret Egenskaper för arbetsytan.

Om alternativ som du inte behöver visas på menyn Arbetsyta kan du ta bort motsvarande rader i filen `.openwin-menu`.

Anpassa tangentbordet



Den här bilagan beskriver hur du anpassar tangentbordet. Bilagan beskriver bland annat hur du definierar specialtangenter och hur du anpassar tangentbordet om du är vänsterhänt. Här finns också information om hur du aktiverar och stänger av tangenten Compose.

I *Användarhandbok för Solaris* kan du läsa om hur du anpassar musknapparna, till exempel för att passa vänsterhänta.

Aktivera och stänga av tangenten Compose

x86 – Tangenten Compose är definierad som Ctrl-Shift-F1 på x86-system.

Om du inte använder tangenten Compose kan du stänga av den så att du inte behöver oroa dig om du råkar trycka på den av misstag. Ta först reda på tangentkoden för Compose (`Multi_key`):

```
$ xmodmap -pk | grep Multi_key
```

Då visas något i stil med:

```
nn 0xff20 (Multi_key)
```

Den viktiga informationen här är det tvåsiffriga talet i början av raden, vilket representeras av *nn*. Använd denna tangentkod när du lägger till följande rad i filen `.xinitrc`:

```
xmodmap -e 'keycode nn = NoSymbol'
```

Om du åter vill aktivera Compose-tangenten gör du den föregående raden i filen `.xinitrc` till en kommentarrad och startar om OpenWindows.

SPARC *Anpassa tangentbordet för vänsterhänta*

Det här avsnittet beskriver hur du använder en kommandofil (för tangentbord av typerna 4 och 5) för att definiera om tangentbordet så att de flesta special-tangenterna på höger respektive vänster sida av tangentbordet byter plats.

SPARC – Resten av det här kapitlet gäller bara för SPARC-system.

SPARC *Definiera om tangentbordet med en kommandofil*

Gör så här om du vill skapa och använda en kommandofil för att definiera om tangentbordet:

1. Skapa en fil med namnet `lefty.data` med hjälp av ett redigeringsprogram.

Filen kan ligga i vilken katalog som helst, men instruktionerna under punkt 4 måste utföras i samma katalog.

2. Skriv in texten som visas i tabell B-1 som börjar på sidan 167.

Raderna som börjar med utropstecken är bara kommentarer och inte kommandon.

3. Spara ändringarna och avsluta redigeringsprogrammet.

4. Skriv följande vid ledtexten:

```
$ xmodmap lefty.data
```

När du gör detta måste du stå i den katalog där kommandofilen finns.

5. Klicka på en musknapp när markören befinner sig på arbetsytan så att kommandofilen träder i kraft.

Nu har tangenterna definierats om för en vänsterhänt person.

Skriv följande i kommandofilen `lefty.data` (se steg 1).

Tabell B-1 Kommandofilen `lefty.data`

```
!  
! lefty.data  
!  
! Den här filen byter plats på funktionstangenterna på  
! tangentbordet Sun typ 4.  
! Om du vill använda filen ska du ska du skriva följande vid  
! ledtexten.  
!  
! xmodmap <filnamn>  
!  
! Kommentarererna nedan motsvarar tangentkoden som följer  
! direkt efter.  
!  
! L2 (Upprepa) byts mot R1 (Paus)  
! L3 (Egenskaper) byts mot R6 (KP_Multiply)  
! L4 (Ångra) byts mot R4 (KP_Equal)  
! L5 (Fram) byts mot R9 (KP_9)  
! L6 (Kopiera) byts mot R7 (KP_7)  
! L7 (Öppna) byts mot R12 (KP_6)  
! L8 (Klistra in) byts mot R10 (Left)  
! L9 (Sök) byts mot R15 (KP_3)  
! L10 (Klipp ut) byts mot R13 (KP_1)  
!  
! R3 (Break) byts mot L1 (Stopp)
```

Tabell B-1 Kommandofilen `lefty.data` (fortsättning)

```
! R2 (Print) byts mot R10 (vänsterpil)
! R5 (KP_Divide) byts mot R12 (högerpil)
!
! Linefeed byts mot Control-R
!
keycode 10 = R1           R1           Pause
keycode 28 = L2           L2           SunXK_Again
keycode 32 = R6           R6           KP_Multiply
keycode 54 = L3           L3           SunXK_Props
keycode 33 = R4           R4           KP_Equal
keycode 52 = L4           L4           SunXK_Undo
keycode 56 = R9           R9           KP_9           Prior
keycode 77 = L5           L5           SunXK_Front
keycode 58 = R7           R7           KP_7           Home
keycode 75 = L6           L6           SunXK_Copy
keycode 79 = Right       R12          KP_6
keycode 100 = L7          L7           SunXK_Open
keycode 80 = Left        R10          KP_4
keycode 98 = L8           L8           SunXK_Paste
keycode 102 = R15         R15          KP_3           Next
keycode 121 = L9          L9           SunXK_Find
keycode 104 = R13         R13          KP_1           End
keycode 119 = L10         L10          SunXK_Cut
keycode 30 = L1           L1           SunXK_Stop
keycode 29 = Left        R10          KP_4
keycode 53 = Right       R12          KP_6
keycode 118 = Control_R
add control = Control_R
```

SPARC *Byta tillbaka till det gamla tangentbordet*

Det finns två sätt att återställa tangentbordets funktion. Det första sättet är att avsluta OpenWindows och starta om det igen. Det andra är att skapa ännu en kommandofil som byter tillbaka till ett normalt tangentbord. Denna metod rekommenderas om du ofta kommer att byta mellan olika tangentbordsdefinitioner.

Gör så här för att skapa den andra kommandofilen:

- 1. Starta ett redigeringsprogram och skapa en fil med namnet `nolefty.data`.**
Filen måste ligga i samma katalog som filen `lefty.data`.
- 2. Skriv in texten som visas i tabell B-2 som börjar på sidan 170.**
Rader som börjar med ett utropstecken är bara kommentarer och inte kommandon.
- 3. Spara ändringarna och avsluta redigeringsprogrammet.**
- 4. Skriv följande vid ledtexten:**

```
$ xmodmap nolefty.data
```

För att ändringarna ska träda i kraft måste du stå i den katalog där kommandofilen finns.

Tabell B-2 Kommandofilen `nolefty.data`

```
!  
! nolefty.data  
!  
! Den här filen ger tillbaka funktionstangenterna på tangent-  
! bordet Sun typ 4 dess ursprungliga funktioner om du har bytt  
! plats på dem. Om du vill använda filen ska du skriva följande  
! vid ledtexten.  
!  
! xmodmap <filnamn>  
!  
! Återställ standardvärden för de vänstra tangenterna.  
!  
keycode 10 = L2          L2          SunXK_Again  
keycode 32 = L3          L3          SunXK_Props  
keycode 33 = L4          L4          SunXK_Undo  
keycode 56 = L5          L5          SunXK_Front  
keycode 58 = L6          L6          SunXK_Copy  
keycode 79 = L7          L7          SunXK_Open  
keycode 80 = L8          L8          SunXK_Paste  
keycode 102 = L9         L9          SunXK_Find  
keycode 104 = L10       L10         Sun_XK_Cut  
!  
! Återställ standardvärden för de högra tangenterna.  
!  
keycode 28 = R1          R1          Pause  
keycode 29 = R2          R2          Print  
keycode 30 = R3          R3          Scroll_Lock  Break
```

Tabell B-2 Kommandofilen `nolefty.data` (fortsättning)

```
keycode 52 = R4      R4      KP_Equal
keycode 53 = R5      R5      KP_Divide
keycode 54 = R6      R6      KP_Multiply
keycode 75 = R7      R7      KP_7      Home
keycode 77 = R9      R9      KP_9      Prior
keycode 98 = Left   R10     KP_4
keycode 100 = Right R12     KP_6
keycode 119 = R13   R13     KP_1      End
keycode 121 = R15   R15     KP_3      Next
!
! Återställ tangenten Line Feed och ta bort den från
! tangentöversikten.
!
remove control = Control_R
keycode 118 = Linefeed
```

Använda program över nätverk



Den här bilagan beskriver en avancerad funktion i OpenWindows som gör att du kan använda program som finns på en annan arbetsstation i nätverket.

Observera – De flesta användare behöver inte läsa den här bilagan. Om du vill utforska möjligheten att använda program över nätverk kan du fråga systemadministratören om det finns några sådana program.

För det mesta är alla program i OpenWindows som visas på din skärm (till exempel postverktyget och kalendern) program som finns på din arbetsstation. Men om din arbetsstation är en del av ett nätverk kan du använda program som finns på en annan arbetsstation så att de visas på din egen skärm. Att använda program på det här sättet sparar kapacitet på din arbetsstation.

Den här bilagan beskriver de enklaste sättet att använda ett program på en annan arbetsstation så att det visas på din egen skärm. Eftersom datormiljöer kan vara olika kan det hända att de här instruktionerna inte helt stämmer med din miljö. Avsnittet "Mer om säkerhet" ger ytterligare information om hur komplicerat det kan vara att använda program över nätverk.

För att kunna använda program på andra arbetsstationer på det sätt som beskrivs nedan krävs följande:

- Du måste ha behörighet till den andra arbetsstationen.
- Den andra arbetsstationen måste kunna NFS-montera din hemkatalog.
- Programmet och de bibliotek som behövs måste vara installerade på den andra arbetsstationen eller *värddatorn*.

Om du har några frågor när det gäller de här kraven kan du ställa dem till systemadministratören.

Använda `rlogin` för att använda ett program över nätverk

Grundförutsättningen när du ska använda ett program på en annan arbetsstation över ett nätverk är att dina omgivningsvariabler i ditt skal på den andra arbetsstationen är rätt inställda:

- `HOME` ska vara inställd på din hemkatalog.
- `DISPLAY` ska vara inställd på skärmen på din arbetsstation.
- Om OpenWindows inte är installerat i gemensamma filsystem (`/usr/lib` eller `/usr/local`) ställer du in `LD_LIBRARY_PATH` på `$OPENWINHOME/lib`.

Här följer ett exempel på hur kommandoverktyget på en annan arbetsstation används med hjälp av kommandot `rlogin`. I det här exemplet är hemkatalogen monterad i den andra arbetsstationen på `/home/minkatalog`, och OpenWindows finns i `/usr/openwin` på den andra arbetsstationen. Ändra variablerna `minkatalog` och `minstation` så att de passar dig. Ersätt också `cmdtool` med namnet på det program du tänker använda.

```
$ rlogin annanstation
...
(Följande kommandon utförs på den andra arbetsstationen.)
...
$ HOME=/home/minkatalog
$ DISPLAY=minstation:0
$ LD_LIBRARY_PATH=/usr/openwin/lib
$ /usr/openwin/bin/cmdtool &
```


När du har skrivit den sista raden och tryckt på Return visas kommandofönstret på din skärm. Fastän du använder programmet precis som du skulle ha gjort med vilket program som helst på din skärm finns själva programmet på den andra arbetsstationen.

Det finns ingen särskild anledning att använda just kommandoverktyget på det här sättet, eftersom det redan finns på din egen arbetsstation och inte använder någon väldigt stor del av kapaciteten där. Men exemplet visar hur du på din egen arbetsstation kan använda ett program som finns på en annan arbetsstation.

Mer om säkerhet

Det här avsnittet beskriver några grundläggande principer när det gäller nätverkssäkerhet som kan vara bra att tänka på när du använder program över nätverket. De handlar om:

- att kontrollera behörigheten till användarkonton och till värddatorer
- behörighetsprotokollen `MIT-MAGIC-COOKIE-1` och `SUN-DES-1`
- när och hur en hanterares behörighetskontroll ska ändras
- hur ett program på den egna arbetsstationen eller en annan arbetsstation används med ett annat användarnamn

Vem ska läsa det här avsnittet?

De standardvalda säkerhetsinställningarna i OpenWindows 3.3 och senare versioner behöver inte ändras om inte program används på följande sätt:

- Du använder program som är länkade med versioner av `Xlib` eller `libcps` som är äldre än OpenWindows 2 eller `X11R4`.
- Du använder program som är länkade till biblioteken i OpenWindows 2 och du vill använda behörighetsprotokollet `SUN-DES-1`.
- Du använder ett program på en hanterare.

Behörighetskontroll

Behörighetskontroll är ett sätt att avgöra vilka klienter eller program som har behörighet till X11-hanteraren. Endast klienter med behörighet får ansluta sig till hanteraren. Om behörighet saknas visas ett felmeddelande och klienten avslutas.

Det finns två olika typer av behörighetskontroll: en som ger behörighet till en användares konto och en som ger behörighet till en värddator eller annan arbetsstation. Om inte tillägget `-noauth` läggs till `openwin` är båda behörighetskontrollerna aktiva. Du kan läsa mer om det här i avsnittet "Styra behörigheten till hanteraren" i det här kapitlet.

Kontobehörighet

En användare kan uttryckligen få kontobehörighet på en värddator. Användarens klient sänder behörighetsdata till hanteraren. Om dessa data passar ihop med hanterarens behörighetsdata får användaren tillgång till hanteraren.

Behörighet till en värddator

Kontrollen av behörighet till en värddator är mer allmän än kontobehörighet. Med den kan du ge behörighet till en viss värddator, och alla användare på den kan då ansluta sig till en hanterare. Det här är en mer begränsad behörighetskontroll eftersom alla användare på värddatorn har tillgång till hanteraren om värddatorn har det.

Program som är länkade med versioner av `xlib` eller `libcps` som är äldre än `OpenWindows 2` eller `X11R4` känner inte igen den nyare formen av behörighetskontroll. Det är oftast därför man nöjer sig med kontroll av behörigheten till en värddator. För att programmen ska kunna ansluta sig till hanteraren måste användaren antingen använda behörighetskontroll till värddatorn eller länka om med nyare versioner av `xlib` och `libcps`.

Observera – Om det är möjligt bör klienter som är länkade med äldre versioner av `xlib` eller `libcps` länkas om med nyare versioner av de här biblioteken. Då kan de anslutas till hanteraren med den nyare formen av behörighetskontroll.

Behörighetsprotokoll

Det går att använda två behörighetsprotokoll i den här versionen av `OpenWindows`: `MIT-MAGIC-COOKIE-1` och `SUN-DES-1`. De använder olika behörighetsdata men kontrollerar behörighet på samma sätt. Hanteraren

använder aldrig mer än ett protokoll åt gången. Protokollet MIT-MAGIC-COOKIE-1, som kontrollerar kontobehörighet, är standardvalet i OpenWindows.

MIT-MAGIC-COOKIE-1

Behörighetsprotokollet MIT-MAGIC-COOKIE-1 har utvecklats av Massachusetts Institute of Technology. När en hanterare startas skapas en teckensekvens för hanteraren och den användare som startade systemet. Varje gång någon försöker ansluta sig till hanteraren sänder användarens klient teckensekvensen till hanteraren som en del av anslutningspaketet. Teckensekvenserna jämförs, och om de stämmer överens tillåts anslutningen.

SUN-DES-1

Behörighetsprotokollet SUN-DES-1, som har utvecklats av Sun Microsystems, bygger på Secure RPC (Remote Procedure Call), och för att kunna använda det behövs programmet DES (Data Encryption Software).

Behörighetsinformationen utgörs av en användares datoroberoende namn i nätverket. Den här informationen kodas och sänds till hanteraren som en del av anslutningspaketet. Hanteraren tolkar informationen, och om namnet i nätverket känns igen tillåts anslutningen.

Det här protokollet ger bättre säkerhet än MIT-MAGIC-COOKIE-1. Det är omöjligt för en annan användare att använda ditt datoroberoende namn i nätverket för att ansluta sig till en hanterare, men det är möjligt för en annan användare att använda teckensekvensen för protokollet MIT-MAGIC-COOKIE-1.

I avsnittet "Ge behörighet med SUN-DES-1" i det här kapitlet beskriver hur du kan ge andra användare behörighet till din hanterare genom att lägga till deras namn i nätverket på din hanterares behörighetslista.

Byta standardvalt behörighetsprotokoll

Det standardvalda behörighetsprotokollet, MIT-MAGIC-COOKIE-1, kan bytas till SUN_DES-1. Det går också att vara helt utan kontroll av kontobehörighet. Du ändrar standardvalet genom att ange tillägg till kommandot `openwin`. Om du till exempel vill ändra standardvalet från MIT-MAGIC-COOKIE-1 till SUN-DES-1, startar du OpenWindows så här:

```
$ openwin -auth sun-des
```

Om du måste använda programmet OpenWindows utan kontroll av kontobehörighet anger du tillägget `-noauth`:

```
$ openwin -noauth
```



Varning – Säkerheten blir sämre när du använder `-noauth`. Det är detsamma som att använda OpenWindows med bara kontroll av behörigheten till en värddator; hanteraren avaktiverar kontrollen av kontobehörighet. Vem som helst som kan använda programmen på din arbetsstation får behörighet till din hanterare.

Styra behörigheten till hanteraren

Om inte tillägget `-noauth` används till `openwin` (läs avsnittet "Byta standardvalt behörighetsprotokoll" här ovan), är kontrollen av både kontobehörighet och behörighet till värddatorn aktiva. Hanteraren kontrollerar först kontobehörigheten och sedan behörigheten till värddatorn. Det standardvalda säkerhetsinställningarna använder MIT-MAGIC-COOKIE-1 för att kontrollera kontobehörighet och en tom lista för att kontrollera behörigheten till värddatorn. Eftersom listan är tom är bara kontrollen av kontobehörighet effektiv. Om du anger tillägget `-noauth` stängs kontrollen av kontobehörighet av och den lokala värddatorn läggs till i listan över värddatorer.

Det finns två program som du kan använda för att ändra hanterarens behörighetskontroll: `xhost` och `xauth`. Programmen hämtar information från två binära filer som behörighetsprotokollet skapat.

Filerna innehåller sessionsspecifika behörighetsdata. Den ena filen ska bara användas av hanteraren. Den andra finns i användarens \$HOME-katalog:

- `.Xauthority` fil för klientbehörighet

Använd programmet `xhost` för att ändra i listan över värddatorer i hanteraren. Du kan lägga till eller ta bort värddatorer från listan. Om du startar med de standardvalda inställningarna – en tom värddatorbaserad behörighetslista – och använder `xhost` för att lägga till namnet på en arbetsstation sänker du säkerhetsnivån. Hanteraren ger behörighet till den värddator som du la till på listan och till alla användare som anger det standardvalda behörighetsprotokollet. I avsnittet ”Behörighet till en värddator” förklaras varför kontrollen av behörigheten till en värddator ger en lägre säkerhetsnivå.

Programmet `xauth` hämtar data till behörighetsprotokollet i klientfilen `.Xauthority`. Du kan hämta den här informationen i din `.Xauthority`-fil så att en annan användare kan lägga den till sin `.Xauthority`-fil och på så sätt få behörighet till din hanterare eller till den hanterare som du ansluter dig till.

I avsnittet ”Ge behörighet med MIT-MAGIC-COOKIE-1” finns exempel på hur `xhost` och `xauth` kan användas.

Fil för klientbehörighet

Filen för klientbehörighet är `.Xauthority`. Den innehåller poster som ser ut så här:

<code>connection-protocol</code>	<code>auth-protocol</code>	<code>auth-data</code>
----------------------------------	----------------------------	------------------------

Standardinställningen i `.Xauthority` är att MIT-MAGIC-COOKIE-1 är *auth-protocol*, och att poster som endast visas lokalt är *connection-protocol* och *auth-data*. På värddatorn *anyhost* kan till exempel filen `.Xauthority` innehålla följande poster:

<code>anyhost:0</code>	<code>MIT-MAGIC-COOKIE-1</code>	<code>82744f2c4850b03fce7ae47176e75</code>
<code>localhost:0</code>	<code>MIT-MAGIC-COOKIE-1</code>	<code>82744f2c4850b03fce7ae47176e75</code>
<code>anyhost/unix:0</code>	<code>MIT-MAGIC-COOKIE-1</code>	<code>82744f2c4850b03fce7ae47176e75</code>

När klienten startar läses en post som stämmer överens med *connection-protocol* från *.Xauthority*, och *auth-protocol* och *auth-data* sänds till hanteraren som en del av anslutningspaketet. Standardinställningen är att *xhost* sänder tillbaka tomma värddatorlistor och talar om att behörighetskontrollen är igång.

Om du har bytt behörighetsprotokoll från det standardvalda till *SUN-DES-1* innehåller posterna i *.Xauthority SUN-DES-1* som *auth-protocol* och namn i nätverket som *auth-data*. Användarkoden har den här formen:

```
unix.användar-id@NIS-domännamn
```

På värddatorn *vård* kan till exempel filen *.Xauthority* innehålla följande poster, där *unix.15339@EBB.Eng.Sun.COM* är användarens datorberoende kod:

```
anyhost:0      SUN-DES-1      "unix.15339@EBB.Eng.Sun.COM"  
localhost:0    SUN-DES-1      "unix.15339@EBB.Eng.Sun.COM"  
anyhost/unix:0 SUN-DES-1      "unix.15339@EBB.Eng.Sun.COM"
```

Observera – Om du inte vet vad du har för namn i nätverket kan du fråga systemadministratören.

Ge behörighet med MIT-MAGIC-COOKIE-1

Om du använder behörighetsprotokollet *MIT-MAGIC-COOKIE-1* ger du en annan användare behörighet till din hanterare så här:

- Lägg över en post som motsvarar *hostname:0* till en fil genom att ge kommandot *xauth* på hanterardatorn.**

I det här exemplet är *hostname* *vård*, och filen är *xauth.info*:

```
$ $OPENWINHOME/bin/xauth nextract - vård:0 > $HOME/xauth.info
```

- Sänd filen till den användare som vill ha behörighet (använd postverktyget, *rcp* eller någon annan överföringsmetod).**

Observera – Det är säkrare att använda postverktyget för att sända filen med behörighetsinformation än att använda `rcp`. Om du ändå använder `rcp` ska du *inte* placera filen i en katalog som är lättillgänglig för en annan användare.

3. Den andra användaren måste lägga posten i sin `.Xauthority`-fil.

I det här exemplet lägger *userhost* `xauth.info` i sin `.Xauthority`-fil:

```
$ $OPENWINHOME/bin/xauth nmerge - < xauth.info
```

Observera – `auth-data` är sessionsspecifik, så den gäller bara så länge som hanteraren inte startas om.

Ge behörighet med `SUN-DES-1`

Om du använder behörighetsprotokollet `SUN-DES-1` ger du en annan användare behörighet till din hanterare så här:

1. Presentera den nya användaren för hanteraren genom att använda `xhost` på hanterardatorn.

I det här exemplet ska den nya användaren *någon* få behörighet till *minvärd*:

```
$ xhost + någon@minvärd
```

2. Den nya användaren måste ange `xauth` för att lägga posten i sin `.Xauthority`-fil.

I det här exemplet är användarens datorberoende kod `unix.15339@EBB.Eng.Sun.COM`. Observera att det här kommandot ska skrivas i en följd utan vagnretur. Efter tecknet `@` skriver du mellanslag och därefter resten av kommandot.

```
$ echo 'add minvärd:0 SUN-DES-1 "unix.15339@EBB.Eng.Sun.COM"' |  
$OPENWINHOME/bin/xauth
```

Eftersom raden ovan är så lång fortsätter den på en ny rad. Men när den visas på din skärm får den plats.

Använda klienter från någon annans arbetsstation eller som en annan användare på den egna arbetsstationen

X-klienten använder värdet på omgivningsvariabeln `DISPLAY` för att ta reda på namnet på den hanterare som de ska ansluta sig till.

Så här använder du klienter på någon annans arbetsstation eller som en annan användare på din egen arbetsstation:

1. Ge en annan användare behörighet på hanterararbetsstation.

Beroende på vilket behörighetsprotokoll du använder följer du instruktionerna i något av avsnitten "Ge behörighet med MIT-MAGIC-COOKIE-1" eller "Ge behörighet med SUN-DES-1".

2. Ställ in `DISPLAY` genom att ange namnet på den värddator där hanteraren finns.

I det här exemplet heter värddatorn `annan`:

```
$ DISPLAY=annan:0
```

3. Starta klientprogrammet genom att skriva dess namn.

```
$ klientprogram&
```

Klienten visas på den andra arbetsstationen, `annan`.

SPARC-system och DNI (DECnet Interworking)



Den här bilagan beskriver kommunikation mellan OpenWindows och DECwindows™ via transportprotokollet NSP DECnet.

SPARC – Hela det här kapitlet gäller bara för SPARC-system. Dessutom går det bara att kommunicera med DECnet med DNI 8.x.

Det finns två tänkbara sätt att använda DNI:

- Du kan köra en X11-klient på en VAX-dator med operativsystemet VMS® och visa klientfönstret på en arbetsstation med OpenWindows.
- Du kan köra en X11-klient på en arbetsstation med OpenWindows och visa klientfönstret på ett VAX-system.

Dessa två sätt beskrivs i den här bilagan. Bilagan inleds dock med ett avsnitt som beskriver hur DNI ska vara inställt.

Inställningar för att kommunicera med DECnet

Gör så här om du vill göra inställningar för att kommunicera med DECnet:

1. Upprätta en förbindelse via DNI.

OpenWindows-hanteraren och klientbiblioteken använder en dynamiskt laddbar version av DNI-biblioteket `libdni`. För att det ska kunna startas måste du ändra variabeln `DNI_X_ENABLE` till den katalog där `libdni.so` finns installerat.

I exemplet antas att du installerat DNI med `pkgadd` i standardkatalogen:

```
$ DNI_X_ENABLE=/opt/SUNWconn/dni/lib
```

2. Starta OpenWindows-hanteraren.

OpenWindows-hanteraren stöder som standard behörighetssystemet `MIT-MAGIC-COOKIE`. Det systemet utgår från användarna och inte från arbetsstationerna, vilket innebär att det är ett vissa användare som har rätt att ansluta sig till hanteraren och inte vissa arbetsstationer. I standardläget ger kommandot `xhost` bara en tom lista som resultat. Den talar om att behörighetssystemet är påkopplat. Det går att stänga av behörighetssystemet och i stället använda behörighetssystemet för tidigare versioner av OpenWindows-hanteraren. Du ger bara kommandot `openwin` med tillägget `-noauth`.

```
$ openwin -noauth
```

3. Be ägaren till arbetsstationen med OpenWindows att ge kommandot `xhost` så att VAX-datorn får behörighet att skapa en X11-förbindelse till OpenWindows-hanteraren.

För att X11-klienter ska kunna anslutas till OpenWindows-hanteraren via DNI, måste nodadresserna för DECnet ha kopplingar till nodnamnen för DECnet. Det här gör du genom att skapa och initiera NCP-databasen, vilket också måste göras på VAX-datorn.

```
$ xhost decvax::
```

De två kolontecknen anger det logiska namnet för DECNet.

Visa en fjärransluten klient på en arbetsstation med OpenWindows

Du kan köra X11-klienter med VMS genom att logga in på VAX-systemet med DNI-kommandot `dnilogin`. Ställ först in variabeln `DISPLAY` på den lokala arbetsstationen så att den är X11-hanteraren på den fjärranslutna arbetsstationen. Kör sedan en X11-klient genom att skriva klientens namn. I vårt exempel kallas den `x11_klient`. I *VMS DECwindows User's Guide, Running Applications Across the Network* finns mer information om operativsystemet VMS.

Ett exempel:

```
$ dnilogin decvax
.
.
.
$ define DECW$DISPLAY OW_machine::0
$ spawn/nowait run x11_klient
```

Visa en fjärransluten klient på en VAX-dator

Du kan köra X11-klienter på en arbetsstation med OpenWindows och sedan visa dem på en DECwindows-hanterare genom att ställa in variabeln `DISPLAY` så att den är det fjärranslutna VAX-systemet.

Innan du kan köra någon X11-klient måste du installera de teckensnitt som används för OpenWindows i DECwindows-hanteraren. Dessa teckensnitt finns i X11R4-utgåvan från MIT och i den valbara teckensnittsupsättningen till OpenWindows version 3.3. Gör så här om du vill installera teckensnitten i DECwindows-hanteraren:

- 1. Installera de valbara teckensnitten till OpenWindows (eller teckensnitten i X11R4-utgåvan från MIT) i arbetsstationen med OpenWindows.**
- 2. Följ instruktionerna i *OpenWindows Server Programmer's Guide* för hur du installerar teckensnitt.**

3. Kopiera teckensnitten till en katalog i VAX-systemet.

```
$ cd $OPENWINHOME/share/src/fonts/misc  
$ dnicp *.bdf `decvax:[vaxdir]`
```

4. Installera teckensnittet för skärmen på VAX-datorn.

Som resultat får du filer som till exempel `olcursor.decw$font;1` och `olglyph10.decw$font;1` och så vidare.

```
$ font olcursor.bdf  
$ font olglyph10.bdf  
...
```

5. Kopiera teckensnitten till katalogen `sysfont`:

```
$ set def sys$sysroot:[sysfont.decw.user_cursor16]  
$ copy [vaxdir]olcursor.decw$font;1 *
```

Observera – Du måste ha loggat in som "system" på VAX-datorn för att kunna kopiera teckensnitten till katalogen `sysfont`.

6. Du måste utföra stegen 2-4 för resten av skärmteckensnitten och för **Lucida-teckensnitten** i `$OPENWINHOME/share/src/fonts/75dpi` och `$OPENWINHOME/share/src/fonts/100dpi`.

Observera – Lucida-teckensnitten ska installeras i `sys$sysroot:[sysfont.decw.user_75dpi]` och `sys$sysroot:[sysfont.decw.user_100dpi]`.

Nedan följer en lista över de teckensnitt som måste vara installerade för att du ska kunna köra DeskSet-programmen för OpenWindows. Om du använder standardteckensnitten för programmen behöver du bara installera dessa teckensnitt. Men om du vill kan du installera ännu fler teckensnitt.

- olcursor.bdf
- olglyph10.bdf
- olglyph12.bdf
- olglyph14.bdf
- olglyph19.bdf
- luBS08.bdf
- luBS10.bdf
- luBS12.bdf
- luBS14.bdf
- luRS08.bdf
- luRS10.bdf
- luFS12.bdf
- lutBs12.bdf
- lutRS10.bdf
- lutRS12.bdf

7. Starta om DECwindows-hanteraren.

8. Du kan ta reda på om teckensnitten är riktigt installerade genom att ta fram en lista över de teckensnitt som finns tillgängliga i DECwindows-hanteraren:

```
$ DISPLAY=decvax::0
$ xlsfonts | grep Sun      (skärmteckensnitt)
$ xlsfonts | grep Lucida
```

9. Se till att OpenWindows-arbetsstationen har rätt att använda DECwindows-hanteraren. Det gör du genom att använda menyn Security i DECwindows sessionshanterare.

10. Starta ett X11-program (till exempel ett DeskSet-program för OpenWindows).

```
$ DISPLAY=decvax::0
$ mailtool
```

Observera – Variabeln `DNI_X_ENABLE` måste vara inställd på sökvägen för DNI-biblioteket `libdni` (se steg 1 i avsnittet "Inställningar för att kommunicera med DECnet" i det här kapitlet).

Om ett felmeddelande visas, till exempel det i exemplet nedan, måste du installera det teckensnitt som anges i DECwindows-hanteraren för att programmet ska fungera.

```
XView warning: Cannot load font '-b&h-lucida-medium-r-*-*  
*-80-*-*-*-*-*' (Font package)
```

Felmeddelandet innebär att teckensnittet `luRS10.bdf` måste installeras.

Mer information om teckensnitt finns i *OpenWindows Server Programmer's Guide*.

Installera modem och skrivare



Installera modem i systemet

Med hjälp av ett *modem* är kan systemet ta emot och sända information över telefonnätet. Du kan använda Administration Tool för att installera modem i systemet. Administration Tool har ett grafiskt användargränssnitt och är därför enkelt att använda.

1. Anslut modemmet.

Gör så här om du vill ansluta modemmet till systemet:

- Anslut modemmet till telefonnätet och till systemet med respektive kablar. (Om du har ett modemkort installerar du det först i systemenheten.)
- Ställ in överföringshastighet, port och andra parametrar på modemmet (genom att ställa in vippkontakter eller med hjälp av ett program).
- Anslut modemets nätkabel eller nätenhet ett vägguttag, om det behövs.

In dokumentationen som följer med modemmet och i installationsanvisningarna till systemet kan du läsa om du behöver gör något mer.

2. Bli superanvändare.

Om du inte tillhör UNIX sysadmin group (GID 14), måste du bli superanvändare på systemet för att få använda Administration Tool.

Använd kommandot `su` för att bli superanvändare:

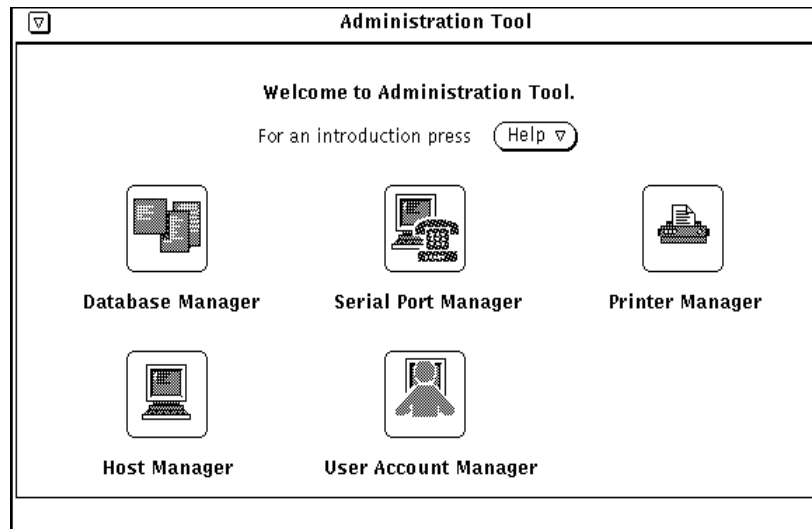
```
$ su
Password:      (ange superanvändarens lösenord, om det efterfrågas)
#
```

3. Starta Administration Tool.

Ge följande kommando:

```
# /usr/bin/admintool &
#
```

Administration Tool visas efter ett litet tag.



Figur E-1 Administration Tool

4. Starta Serial Port Manager.

Dubbelklicka på ikonen för Serial Port Manager för att starta verktyget. Ett nytt fönster visas. Där finns inställningar och reglage med vilka du hanterar modem och terminaler.

5. Följ instruktionerna i direkthjälpen.

Du tar fram direkthjälpen för Serial Port Manager genom att välja Help... i grundfönstret för verktyget. Direkthjälpen ger dig alla instruktioner du behöver för att installera modem i systemet.

Mer information om modem

I handboken *Peripherals Administration* (som är en del av *Solaris 2.4 System Administrator AnswerBook*) kan du läsa mer om hur du installerar och konfigurerar modem.

Installera en lokal skrivare i systemet

Det är enklast att använda Administration Tool för att installera en skrivare i systemet. Följande instruktioner beskriver hur du installerar en skrivare som du ska använda lokalt. Hur du installerar skrivare för användning i nätverk kan du läsa i handboken *Peripherals Administration* (som är en del av *Solaris 2.4 System Administrator AnswerBook*).

Gör så här om du vill installera en lokal skrivare i systemet:

1. Anslut skrivaren.

Gör så här om du vill ansluta modem till systemet:

- Anslut skrivarens kabel till systemet.
- Ställ in överföringshastighet, port och andra parametrar på skrivaren.
- Anslut skrivarens nätkabel eller nätenhet ett vägguttag.

Du kan använda Administration Tool för att installera skrivaren i systemet före eller efter det att du ansluter skrivaren till systemet och slår på strömmen. I handboken som följer med skrivaren kan du läsa vilka inställningar du kan göra på skrivaren och vilken typ av kabel du ska använda. Vanligtvis ansluter du skrivarens kabel till en serieport, men i vissa fall, beroende på vilken typ av skrivare du installerar, kan du i stället använda en parallellport.

2. Bli superanvändare.

Om du inte tillhör UNIX sysadmin group (GID 14), måste du bli superanvändare på systemet för att få använda Administration Tool.

Använd kommandot `su` för att bli superanvändare:

```
$ su
Password:      (ange superanvändarens lösenord, om det efterfrågas)
#
```

3. Starta Administration Tool.

Ge följande kommando:

```
# /usr/bin/admintool &
#
```

Administration Tool visas efter ett litet tag (se figur E-1 på sidan 190).

4. Starta Printer Manager.

Dubbelklicka på ikonen för Printer Manager för att starta verktyget. Ett nytt fönster visas. Där finns inställningar och reglage med vilka du hanterar skrivare.

5. Följ instruktionerna i direkthjälpen.

Du tar fram direkthjälpen för Print Manager genom att välja Help... i grundfönstret för verktyget. Direkthjälpen ger dig alla instruktioner du behöver för att installera skrivaren i systemet.

Mer information om skrivare

I handboken *Peripherals Administration* (som är en del av *Solaris 2.4 System Administrator AnswerBook*) kan du läsa mer om hur du installerar och konfigurerar skrivare.

Sakregister

Symboler

- . (symbol) 40
- .. (symbol) 33

A

- absolut behörighet 48–50
 - se även* behörighet
- aliasnamn 115, 150
 - se även* mailx (program)
- anpassa
 - musknappar 165
- användarnamn 1
- apropos (kommando) 24
- arbetskatalog, visa 32
- arbetspass 1

B

- bakgrundsjobb 22
- bdiff (kommando) 39
- behörighet
 - .Xauthority (fil) 179–180
 - ange standardval 153–154
 - behörighetskontroll
 - definition 175
 - jokertecknet * 47
 - MIT-MAGIC-COOKIE-1

- (behörighetsprotokoll) 177
- noauth (tillägg) 176
- omfattning 42
- protokoll 176–181
- ställa in absolut 48–50
- till filer och kataloger
 - att använda 42
 - att läsa 42
 - att ändra 42
- typer av 42
- xauth (program) 181
- ändra 45–47
- behörighetsprotokoll, *se* behörighet
- bildskärmar
 - starta OpenWindows på flera bildskärmar 13
 - visa OpenWindows på två skärmar bredvid varandra 14
 - visa OpenWindows på två skärmar ovanpå varandra 15
- Bourne-skalet
 - startfiler 145–146
- brev
 - dolda kopior 108
 - kopior 108
 - läsa 102–103
 - skriva ut 105
 - spara 110–113
 - svara på 110

sända 105–107
flera mottagare 106
buffertar 81
byta katalog 32–34
byta namn
filer 28
kataloger 35

C

cancel (kommando) 132
cat (kommando) 29
cd (kommando) 26, 32–34
CDPATH (omgivningsvariabel) 147
chmod 153
chmod (kommando) 45, 48
citationstecken 57
Compose-tangenten, aktivera/stänga
av 165
convert_to_Xdefaults (program) 159
cp (kommando) 27, 36
.cshrc (fil) 146–153
C-skäl
aliasnamn 150
startfiler 145–146
tidigare kommandon 20
Ctrl-tangenten xvi

D

date (kommando) 18
dead.letter (katalog) 107
.defaults (fil) 159
df (kommando) 64
diff (kommando) 37
diff3 (kommando) 39
direkthjälpen (man) 2, 24, 148
DISPLAY (omgivningsvariabel) 182
ditroff (program) 67
dolda kopior 108
du (kommando) 64
resultat genom sort 64

E

eko 2
elektronisk post, se mailx (program)
email, se mailx (program)
env (kommando) 147
/etc/aliases (fil) 117
/etc/hosts.equiv (fil) 141, 143
/etc/passwd (fil) 141, 143, 145
/etc/profile (fil) 146
ex (kommando) 70
ex-kommandon 82–85

F

fil
ange standardbehörighet 153–154
behörighet 42–50
.cshrc 146
.defaults 159
definition 25–26
/etc/aliases 117
/etc/hosts.equiv 141, 143
/etc/passwd 141, 143, 145
/etc/profile 146
infoga ett brev 109
jämföra 36
kommando 27
kontrollera typ 30
lista över dolda 44
.login 146
.mailrc 115
mbox 98
.profile 146
skapa ny 27
skriva ut 124
spara brev i 111
söka 39–42
titta på innehållet 29
vars namn börjar med punkt 44
.Xauthority 179–180
.Xdefaults 159
.xinitrc 161
ändra behörighet 45–47
file (kommando) 30

`find` (kommando) 39–42
`finger` (kommando) 106
flytta
 filer 28
 kataloger 35
 vi-text 81
 med `ex`-kommandon 84

G

globala nätverk 136
`grep` (kommando)
 metatecken 56–57
 som filter 52
 strängar 53
 syntax för 51–52
 textsträngar 54–56

H

hemkatalog 26, 32
 definiera 148
 gå till med tecknet tilde 32
 `rlogin` (kommando) 137
`history` (kommando) 20, 148
`HISTORY` (omgivningsvariabel) 148
hjälpinformation på skärmen 23
`HOME` (omgivningsvariabel) 148

I

identifikationsnummer för en process, *Se*
 process-id
inkorg för post 98
 ange sökväg 148
 se även `mailx` (program)
inloggningsnamn
 definiera 148
inloggningsskal 2–3, 145
inmatningsläget 69
inställningar för användare 146
internet (stornätverk) 136

J

jokertecken 27
`JUSTERA` (musknapp) xv
jämföra filer 36

K

katalog
 ange standardbehörighet 153–154
 behörighet 42–50
 byta 32–34
 byta namn 35
 `dead.letter` 107
 definiera för mappar 112
 definition 26
 flytta 35
 hem 32
 hierarki 31
 kontrollera utrymme 64
 kopiera 35
 rot 31
 skapa ny 34
 sökväg 30
 ta bort 36
 ändra behörighet 45–47
`kill` (kommando) 63
klocka 149
kommando
 användning 24
 apropos 24
 `bdiff` 39
 `cancel` 132
 `cat` 29
 `cd` 32–34
 `chmod` 45, 48
 `umask` 153
 `copy` 111
 `cp` 27, 36
 `date` 18
 `df` 64
 `diff` 37
 `diff3` 39
 `du` 64
 `env` 147

ex 70
file 30
find 39-42
finger 106
för att infoga text 76
för att klistra in 81
för att kopiera en rad 80
för tom rad 76
ge på en annan arbetsstation 142-143
gemener och versaler 18
grep 51-58
history 20, 148
kill 63
logname 19
lp 123
lpstat 127-131
ls 27
man 2
med tecknet tilde 121
mkdir 34, 112
more 29
mv 28
nohup 23
omdirigera dataflöde 21-22
omdirigera resultat 22
openwin 12
passwd 61
ps 62
pwd 26, 31
rcp 141-142
rlogin 136-140
rm 28
rmdir 36
rsh 142-143
rusers 143
skriva flera 19
skriva långa 19
sort 65
syntax 24
söka efter nyckelord 24
tillägg 21
touch 27
umask 153, 154
upprepa tidigare 19-20
upprepa vi 81
utföra i bakgrunden 22
whatis 24
who 105
xauth 181
xmodmap 165
kommandotext 3
definiera 148
ändra 151-153
kommandoläget i vi 69
kommentera bort text 119
konto 1
konventioner, typografiska xiv
kopiera
brev till en fil 111
brev till mappar 112-113
filer 27
från en annan arbetsstation 141
kataloger 36
rader från en vi-fil till en annan 91
till en annan arbetsstation 142
vi-text 80
vi-text med ex-kommandon 83
kopior 108
Korn-skalet
startfiler 145-146
körbara filer 25

L

LANG (omgivningsvariabel) 148
ledning
brev till kommandot lp 105
resultatet genom grep 52
ledtext 17
line printer (delsystem), se utskrift
logga in 1-2
starta OpenWindows 11
logga in i systemet 1, 17
på en annan arbetsstation 136-140
som någon annan 138
logga ut 3
.login (fil) 146
logname (kommando) 19
LOGNAME (omgivningsvariabel) 148

lokala nätverk 136
lp (kommando) 123
 tabell över tillägg 127
LP, skrivarhantering, *se* utskrift
LPDEST (omgivningsvariabel) 148
lpstat (kommando) 127–131
 syntax 131
 tabell över tillägg 131
ls (kommando) 27, 43
 lista över dolda filer 44
 tillägg för långt format 43–44
lösenord 1
 byta 60
 tidsbegränsade 61
 välja 59

M

MAIL (omgivningsvariabel) 148
.mailrc (fil) 115
mailx (program)
 aliasnamn
 definiera 115
 i .mailrc-fil 115–117
 i /etc/aliases 117–120
 jämförelsetabell 120
 använda kommandot vi 114
 avsluta 101
 avsluta ett osänt brev 107
 brev som inte kan sändas 107
 copy (kommando) 111
 få hjälp 122
 infoga en fil 109
 infoga ett brev 109
 inkorg 98
 läsa brev 100, 102–103
 mappar 111–114
 mbox (fil) 98
 spara brev 111–113
 starta 98
 svara på brev 110
 ta bort brev 104–105
 tildekommandon 121
 version 100
 visa lista över brev 103

man (kommando) 2, 23
MANSECTS (omgivningsvariabel) 148
mapp för post 111
 se även mappar
mappar 111–114
mbox (fil) 98
 se även mailx (program)
MENY (musknapp) xv
metatecken 54
MIT-MAGIC-COOKIE (behörighet) 184
mkdir (kommando) 34, 112
modem
 (definition) 189
more (kommando) 29
mus
 anpassa för vänsterhänta 165
musknappar
 JUSTERA xv
 MENY xv
 VÄLJ xv
mv (kommando) 28

N

namn för inloggning 1
-noauth (tillägg till openwin) 12, 178
nohup (kommando) 23
nroff (program) 67
nätverk
 använda program över DECnet 183
 definition 135
 stora xiii

O

omdirigera resultatet av kommandon 22
omgivningsvariabler
 CDPATH 147
 DISPLAY 182
 HISTORY 148
 HOME 148
 i startfiler 146–150
 LANG 148

LOGNAME 148
LPDEST 148
MAIL 148
MANSECTS 148
PATH 148, 149
PS1 148
SHELL 148
TERM 149
TERMINFO 148
TZ 149
openwin (kommando)
-dev (tillägg) 13-14, 15
-dev tillägg 15
-noauth (tillägg) 12, 176, 178
tillägg för enhet 13
OpenWindows
använda flera bildskärmar 13
behörighet med MIT-MAGIC-
COOKIE 184
logga in och logga ut 11
postverktyget 97
startfiler 146

P

parallellport 191
passwd (kommando) 61
PATH (omgivningsvariabel) 148
processer
avbryta 63
pågående 62
process-id 62-63
processortid 62
.profile (fil) 146-153
program
convert_to_Xdefaults 159
ditroff 67
mailx 98
nroff 67
på en annan dator 174
troff 67
vi 67
view 67
över nätverk med DNI (DECnet) 183

ps (kommando) 62
tillsammans med kommandot
grep 63
PS1 (omgivningsvariabel) 151
PS1 (omgivningvariabel) 148
punktfiler 44
pwd (kommando) 26, 32

R

rcp (kommando) 141-142
redigeringsprogrammet vi, se vi
(redigeringsprogram)
regionala nätverk 136
rekursiv kopiering 36
relativ sökväg 34
rlogin (kommando) 136-140
avbryta en förbindelse 139
göra paus i en förbindelse 140
rm (kommando) 29, 36
rmdir (kommando) 36
rotanvändare 118
rotkatalogen 31
rsh (kommando) 142-143
rusers (kommando) 106, 143

S

serieport 191
SHELL (omgivningsvariabel) 148
skriva ut
se även lp
brev 105
flera exemplar 126
se även utskrift
till en viss skrivare 124-125
vi-filer 73
skrivare
port
parallel- eller serie- 191
skrivarhanteringen LP, se utskrift
små och stora bokstäver i
redigeringsprogrammet vi 73,

-
- 86
 - sort (kommando) 65
 - språk 148
 - standardbehörighet, ställa in 153–154
 - standardledtext 17
 - standardskrivare 124
 - standardval
 - filbehörighet 153–154
 - katalog 26
 - menyalternativ xvi
 - skal 145
 - skrivare 148
 - start
 - tangenten Compose, aktivera/stänga av 165
 - starta
 - information till OpenWindows 2-användare 160
 - på flera bildskärmar 14–15
 - specialalternativ 12
 - startfiler 145–146
 - statusrad 69
 - stornätverk (internet) 136
 - sträng 51
 - SunLink, över nätverk 185
 - SunOS-kommandon xiv
 - SunView 159
 - convert_to_Xdefaults (program) 159
 - filer
 - .default 159
 - .Xdefaults 159
 - systemadministratör 1
 - söka
 - med `find` 39–42
 - med `grep` 51–58
 - sökväg 30
 - relativ 34
 - T**
 - ta bort
 - brev 104–105
 - filer 29
 - kataloger 36
 - vi-text 79
 - vi-text med `ex`-kommandon 85
 - tangentbord
 - anpassa för vänsterhänta 166
 - byta tillbaka för högerhänta 169
 - tecken för omdirigering 22
 - teckensnitt
 - för program över nätverk 187
 - ta fram lista 158
 - ändra standardval 155
 - teckensträng 85
 - TERM (omgivningsvariabel) 149
 - TERMINFO (omgivningsvariabel) 148
 - textfiler 25
 - textsträngar 54
 - tidigare kommandon 19–20
 - tildekommandon 122
 - tillägg till kommandon 21
 - noauth 176
 - touch (kommando) 27
 - troff (program) 67
 - typografiska konventioner xiv
 - TZ (omgivningsvariabel) 149
 - U**
 - umask (kommando) 153, 154
 - upprepa kommandon 19–20
 - utrymme på skivminnen 64
 - utskrift
 - avbryta 132–133
 - kontrollera 127–130
 - se även skriva ut
 - ta reda på skrivartyp 130
 - till standardskrivaren 124
 - tillgängliga skrivare 128
 - utskrifts-id 132
 - utskrifts-id 132
 - utskriftskö 129

V

variabler

- ställa in för redigeringsprogrammet
vi 91

VAX, som värd för X11-klient 183

vi (redigeringsprogram)

- ange antal gånger att upprepa 81
- använda i programmet mailx 114
- avsluta 71-72
- bläddra igenom filer 75
- bryta en rad 78
- buffertar 81
- byta plats på tecken 78
- byta ut
 - en rad 77
 - ett ord 77
 - tecken 77

definition 67

ex-kommando för att

- flytta 84
- kopiera 83
- ta bort 85

flytta text 81

förflyttning i 73-76

gå till en viss rad 89

infoga andra filer 89

infoga text 76

inmatningsläget 69-70

kommando

- för att infoga text 76
- för att klistra in 81
- för att kopiera en rad 80
- för tom rad 76

kommandoläget 69, 70

kopiera text 80

markör 68

- flytta 73

och ex (kommando) 70

och ex-kommandon 82-85

redigera flera filer samtidigt 90-91

rulla i 75

sammanfoga rader 78

sammanställning av grundläggande
kommandon 92-95

skapa en fil 68

skriva ut filer 73

små och stora bokstäver i en
sökning 86

statusrad 68-69

ställa in variabler 91

sök och ersätt 85-88

ta bort

- en rad 79
- ett ord 79
- ett tecken 79
- text 79-80

upprepa infogningar 81

upprepa kommandon 81

visa nummer 82

ångra ändringar 78

återfå förlorad information 92

ändra text 77-78

view (program) 67

visa

användare

- på andra arbetsstationer 143-144
- på din filhanterare 105

arbetskatalog 32

filbehörighet 43

innehåll

- i en fil 29

- i en katalog på en annan
arbetsstation 142

lista över brev i inkorgen 103

skrivarstatus 128-130

utrymme

- i en katalog 64-65
- på skivminnen 64

vilken arbetsstation du arbetar
med 140

VÄLJ (musknapp) xv

vänster hand

använda musen 165

använda tangentbordet 166

W

whatis (kommando) 24

who (kommando) 105

X

xauth (kommando) 181
.Xauthority (fil) 179–180
.Xdefaults (fil)
 SunView 159
.xinitrc (fil) 161
xmodmap
 rita om tangentbordet 166
xmodmap (kommando) 165

Ö

överordnad katalog 33

