

LPI level 1 sertifiseringskurs



**Linux
Professional
Institute**

©2007 Linpro AS

Innholdsfortegnelse

1 LPI Level 1 sertifiseringskurs.....	1
1.1 Beskrivelse, målsetning og forkunnskaper.....	1
1.1.1 Mål for kurset.....	1
1.1.2 Innhold.....	1
1.2 Presentasjonsrunde.....	3
1.3 Kurs i Linpro.....	4
2 Introduksjon til LPI-sertifisering.....	5
2.1 Introduksjon / oversikt.....	5
2.2 Bok.....	6
2.3 Level 1-oversikt.....	8
2.4 Level 2-oversikt.....	9
2.5 Slik blir du sertifisert.....	10
3 Maskinvare og arkitektur (101).....	11
3.1 Maskinvare og arkitektur.....	11
3.2 Filer og kommandoer.....	12
3.3 SCSI-enheter, filer og kommandoer.....	13
3.4 Oppsett av ulike tilleggskort i PC.....	14
3.5 Konfigurere USB-enheter.....	15
4 Installasjon og pakkehåndtering (101).....	17
4.1 Installasjon og pakkehåndtering.....	17
4.2 Diskenes partisjonsmuligheter.....	18
4.3 Partisjoner og størrelser.....	19
4.4 Installere boot-manager.....	20
4.5 Bygge/kompilere og installere programmer fra kildekode.....	21
4.6 Pakkehåndtering/administrasjon i Debian.....	22
4.7 dselect.....	23
4.8 apt-get.....	24
4.9 Søking etter pakker i Debian.....	25
4.10 Vanlige RPM-kommandoer.....	26
5 GNU & Unix-kommandoer (101).....	27
5.1 GNU & Unix-kommandoer.....	27
5.2 Filtre.....	28
5.3 Filhåndtering.....	29
5.4 Omdirigering og rørlegging.....	30
5.5 Starte, overvåke og drepe prosesser.....	31
5.6 Arbeide med tekstfiler og regulære uttrykk.....	32
6 Enheter, Linux filsystemer og FHS (101).....	33
6.1 Enheter, Linux filsystemer og FHS.....	33
6.2 Montering, avmontering og plassbegrensning (quota).....	34
6.3 Eierskap, rettigheter og linking.....	35
6.4 FHS (Filesystem Hierarchy Standard).....	36
6.5 whatis / apropos.....	37
7 Vindussystemet X (101).....	39
7.1 Vindussystemet X.....	39
7.2 Installer og tilpass vindushåndterer og miljø.....	40

Innholdsfortegnelse

8 Kjernen (102).....	41
8.1 Kjernen.....	41
8.2 Konfigurere og bygge kjerne/kjernemoduler.....	42
8.3 Starte, initialisere og stoppe systemet.....	43
9 Utskrift (102).....	45
9.1 Utskrift.....	45
10 Dokumentasjon (102).....	47
10.1 Dokumentasjon.....	47
10.2 Lokal dokumentasjon.....	47
10.3 Dokumentasjon på Internett.....	47
10.4 Mer om manualsider.....	48
11 Shell, skripting, programmering og kompilering.....	49
11.1 Shell, skripting, programmering og kompilering.....	49
11.2 Variabler i bash.....	50
11.3 Testing og løkker.....	51
12 Administrative oppgaver (102).....	53
12.1 Administrative oppgaver (systemadministrasjon).....	53
12.2 Bruker- og gruppeadministrasjon.....	54
12.3 Brukermiljø og miljøvariabler.....	55
12.4 Loggfiler.....	56
12.5 Kjøre jobber i framtida: at.....	57
12.6 Repetere kjøring av jobb på angitte tidspunkter: cron.....	58
12.7 Sikkerhetskopiering.....	59
12.8 Systemtid / systemklokka.....	60
13 Essensielt om nettverk (102).....	61
13.1 Essensielt om nettverk.....	61
13.2 Vanlige nettverkskommandoer.....	62
13.3 Oppsett og feilsøking.....	63
13.4 Sette opp Linux som PPP-klient.....	64
14 Nettverkstjenester (102).....	65
14.1 Nettverkstjenester.....	65
14.2 inetd, xinetd og TCP wrappers.....	66
14.3 Grunnleggende oppsett av Sendmail.....	67
14.4 Grunnleggende Apache-oppsett.....	68
14.5 Administrere NFS og Samba.....	69
14.6 Grunnleggende DNS-oppsett.....	70
14.7 Konfigurere Secure Shell (OpenSSH).....	71
15 Sikkerhet (102).....	73
15.1 Sikkerhet.....	73
15.2 Kommandoer relatert til sikkerhet.....	74
16 Avslutning og oppsummering.....	75
16.1 Avslutning og oppsummering.....	75
16.2 Sluttoppgaver (så langt dere rekker).....	76

1 LPI Level 1 sertifiseringskurs

Slide 1

1.1 Beskrivelse, målsetning og forkunnskaper

Linux Professional Institute er en organisasjon som tar mål av seg å være ledende på Linux-sertifisering på tvers av de ulike distribusjonene. Det ble startet som en non-profit organisasjon i oktober 1999, med mange store i ryggen, f.eks. IBM, HP, Linuxcare og Novell.

LPI Level 1-sertifiseringen ser ut til å feste seg som en industristandard, en sertifisering flere av de store aktørene stiller seg bak.

Dette kurset er både en repetisjon av de viktigste tingene vi må huske før vi tar sertifiseringstestene, og så tar vi veldig kort for oss ting vi ikke har lært tidligere, men som vi bør kunne før vi tar testene. Siden testene ofte kan være litt overfladiske, er det en del emner vi bare såvidt "sneier borti". Noen emner er grundigere behandlet i LPI Level 2.

1.1.1 Mål for kurset

Målet med kurset er at deltakerne skal være i stand til å ta LPI-eksamenene 101 og 102, som tilsammen utgjør LPI Level 1. (Testene taes på et Prometric- eller VUE-testsenter.)

Forkunnskaper

For å kunne ta dette kurset, må du ha kunnskaper tilsvarende kurset Linux/UNIX grunnkurs eller Linux/UNIX oppfriskning, Nettverk & TCP/IP og Linux systemadministrasjon I.

1.1.2 Innhold

- Introduksjon til LPI-sertifisering
- Maskinvare og arkitektur
- Installasjon og pakkehåndtering
 - ◆ Bootloader-konfigurasjon
 - ◆ RPM
 - ◆ DPKG / APT
- GNU & Unix-kommandoer
 - ◆ Kort repetisjon av de vi har lært
 - ◆ Noen nye kommandoer
- Enheter, Linux filsystemer og FHS
- Vindussystemet X
- Kjernen
 - ◆ Kompilering, prinsippet
 - ◆ Moduler
- Utskrift
- Dokumentasjon
- Shell, skripting, programmering og kompilering
- Administrative oppgaver

- ◆ Sikkerhetskopiering
- Essensielt om nettverk
- Nettverkstjenester
 - ◆ Kort om NFS
 - ◆ Kort repetisjon om Samba (SMB/CIFS)
 - ◆ Kort om sendmail
 - ◆ Kort om PPP
- Sikkerhet
- Sluttoppgaver og avslutning

1.2 Presentasjonsrunde

- Hva du heter og hva du jobber med?
- Hvor mye relevant erfaring har du?
- Hva ønsker du å sitte igjen med?
- Spesielle emner du ønsker å høre (mer) om?

Dersom du har spesielle forventninger til kurset er dette et bra tidspunkt å nevne det.

1.3 Kurs i Linpro

- Grunnleggende kurs:
 - ◆ Linux/UNIX grunnkurs
 - ◆ Nettverk og TCP/IP
 - ◆ Linux systemadministrasjon I
 - ◆ LPI level 1 sertifiseringskurs
- Videregående kurs:
 - ◆ Linux/UNIX shellprogrammering
 - ◆ Linux systemadministrasjon II
 - ◆ LPI level 2 sertifiseringskurs
- Linux påbygning:
 - ◆ Linux systemadministrasjon III
 - ◆ Linux sikkerhet
 - ◆ DNS
 - ◆ LDAP
 - ◆ Samba
- Flere programmeringskurs (Perl, Python, Ruby, PHP, XML, SQL)
- Skolelinux
- Linux på skrivebordet
- Spesialtilpassede kurs, workshops

2 Introduksjon til LPI-sertifisering

Slide 4

2.1 Introduksjon / oversikt

LPI-sertifiseringsløp/tester:

VUE/Prometric #	Prometric-navn på test
117101	LPI Level 1 Exam 101
117102	LPI Level 1 Exam 102
117201	LPI Level 2 Exam 201
117202	LPI 202 Advanced Linux

LPI Level 3 er nå tilgjengelig.

Den siste her burde vel hatt navnet "LPI Level 2 Exam 202", men har altså ikke det (dersom det ikke er korrigert i nyere tid).

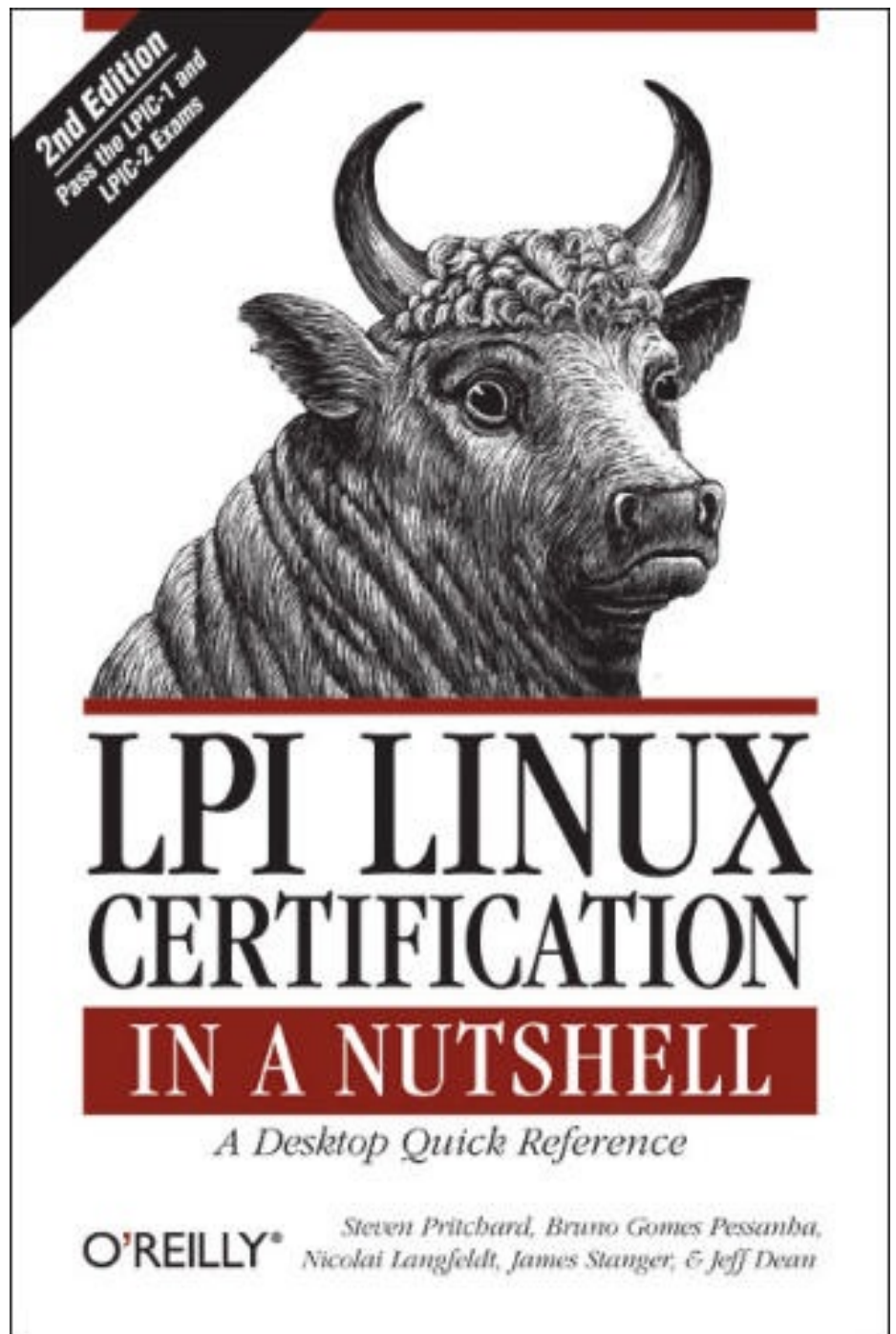
I en periode var det to ulike versjoner av 101-testen. Det er det slutt på.

Når du skal ta test registrerer du deg først hos LPI og så bestiller du testen fra et Pearson VUE eller Thomson Prometric-godkjent test-senter, f.eks. Visma Ajourit, Masterminds, HP, IBM eller Siemens.

2.2 Bok

Kurset baserer seg på boka "LPI Linux Certification in a Nutshell" (Steven Pritchard, Bruno Pessanha, Nicolai Langfeldt, James Stanger, Jeff Dean. O'Reilly 2006, ISBN: 0596005288).

Tips: Legg merke til "On the exam"-boksene i boka



Ikke stol på at innholdet i boka alltid stemmer med innholdet i testene. Autoritativ kilde til informasjon er til enhver tid: <http://www.lpi.org/>

Gå direkte til <http://www.lpi.org/en/lpic.html> for å finne oversikten over hvilke emner som er med og i hvilke tester.

Historisk har vi i Linpro måttet endre innholdet i kurset flere ganger siden LPIC-testene kom. Flere bøker om testene har også blitt utdatert, ikke nødvendigvis fordi innholdet i testene er så veldig oppdatert og nytt,

men de har blitt omorganisert flere ganger, og temaer er flyttet mellom de ulike testene. Forhåpentligvis stemmer innholdet i kursdokumentasjonen her godt med testene, men sjekk alltid før du går opp til test.

Når det står referert til kapitler i boka i resten av kurset, er det altså denne boka vi henviser til. I boka er det dessuten eksempel–eksamener som minner om de virkelige testene.

Både kurset og boka følger LPIC–testene nøye. Boka går mer i dybden på temaene enn det testene gjør.

2.3 Level 1–oversikt

LPI Level 1 Exam 101 tests basic capabilities in these areas:

- Hardware & Architecture
- Linux Installation & Package Management
- GNU & Unix commands
- Devices, Linux Filesystems, Filesystem Hierarchy Standard
- The X Window System

LPI Level 1 Exam 102 tests basic capabilities in these areas:

- Kernel
- Boot, Initialization, Shutdown and Runlevels
- Printing
- Documentation
- Shells, Scripting, Programming, and Compiling
- Administrative Tasks
- Networking Fundamentals
- Networking Services
- Security

Sjekk alltid LPI sine websider for siste versjon før du går opp til test:

<http://www.lpi.org/en/lpic.html>

(Fra menyene går du inn på linken "Certification" og deretter på "The LPIC Program".)

2.4 Level 2–oversikt

LPI Level 2 Exam 201 tests basic capabilities in these areas:

- Linux Kernel
- System Startup
- Filesystems
- Hardware
- System Maintenance
- System Customization and Automation
- Troubleshooting

LPI Level 2 Exam 202 tests basic capabilities in these areas:

- Networking
- Mail & News
- DNS
- Web Services
- Network Client Management
- System Security
- Network Troubleshooting

2.5 Slik blir du sertifisert

Først må du registrere deg på <http://www.lpi.org>

Deretter kontakter et Pearson VUE eller Thomson Prometric–godkjent test–senter, f.eks. Visma Ajourit, Masterminds, HP, Azlan, IBM eller Siemens.

Der bestiller du testen du ønsker å ta (enten via telefon eller web). Deretter sjekker du om innholdslisten til LPI har endret seg, før du forbereder deg godt og går og tar testen.

LPI har en egen "retake policy" som du finner på hjemmesiden deres. Der heter det blant annet at dersom du stryker første gangen, må du vente en uke før du får prøve på nytt. Neste gang må du vente lenger. De har altså en karanteneperiode når du stryker på tester.

Retake policy: (Revidert 19. april 2004)

- Anyone who takes an LPI exam once must wait one week before re–taking.
- Anyone who takes an LPI exam a second (and subsequent) time must wait 90 days before re–taking.
- Anyone who passes an LPI exam may not retake that exam for at least two years.

(Sjekk <http://www.lpi.org/en/policies.html>

Der ser du også at testene er regnet som aktive i 10 år.)

Adressen til der du registrerer deg hos LPI:

<https://www.lpi.org/en/register.html>

3 Maskinvare og arkitektur (101)

Slide 9

3.1 Maskinvare og arkitektur

Innenfor denne kategorien er målet at vi skal kjenne grunnleggende PC/ix86–arkitektur. Vi må vite hva BIOS er, og så må vi f.eks. kjenne til uttrykket LBA, hva det betyr.

LBA

"**Logical Block Addressing**, it is a translation method to allow your computer to support larger hard drives. Before you add a large hard drive (over 504 MB), be sure you have support for LBA translation in your system BIOS – the older BIOS don't support cylinders numbering above 1024. LBA does not use standard CHS parameters. For instance, if your drive has say 2000 cylinders and 16 sectors, LBA translation will make your software think your drive has 1000 cylinders, and 32 heads. Recent developments now allow drives as large as 8.4GB – soon to be broken again."

I boka er dette gjennomgått i kapittel 3: "Hardware and Architecture"

Det kan være lurt å se i boka på tabellene for vanlige I/O–adresser, IRQ og DMA for serieporter, parallellporter og diskettstasjoner. Hvilke IRQ brukes f.eks. vanligvis av ttyS0?

Vi forventes å vite at serieporter kan dele IRQ (interrupt request, avbruddsordrelinje). Det kan være en fordel å ha erfaring med å løse I/O–, IRQ– eller DMA–konflikter. (DMA = direkte minneaksess)

3.2 Filer og kommandoer

Filer vi skal kjenne til innholdet i:

```
/proc/ioports
/proc/interrupts
/proc/dma
/proc/pci
```

Vi skal kjenne til oppsett av modem og lydkort, typisk ISA-greier fra i gamle dager ... Vi må da vite at de trenger unike IRQer, I/O-områder og DMA-adresser. Vi bør kjenne til kommandoene:

```
isapnp(8)
pnpdump(8)
sndconfig
lspci(8)
setserial(8)
```

Linux Device	Windows Name	Typical IRQ	I/O Address
/dev/ttyS0	COM1	4	0x03f8
/dev/ttyS1	COM2	3	0x02f8
/dev/ttyS2	COM3	4	0x3e8
/dev/ttyS3	COM4	4	0x02e8
/dev/lp0	LPT1	7	0x0378–0x037f
/dev/lp1	LPT2	5	0x0278–0x027f
/dev/fd0	A:	6	0x03f0–0x03f7
/dev/fd1	B:	6	0x0370–0x0377

Her anbefaler vi at du leser gjennomgangen i boka grundig dersom du ikke har erfaring med ISA-enheter og slikt.

3.3 SCSI-enheter, filer og kommandoer

Vi bør kjenne til SCSI-enheter, hvordan de konfigureres med SCSI-ID osv. Filer:

```
/proc/scsi/  
scsi_info
```

Det er verdt å merke seg at det i senere revisjoner av testen nå heter "non-IDE devices". Dette kommer av at f.eks. SATA også går gjennom SCSI-laget i kjernen (og da heter `/dev/sdX`).

Dersom du ikke har erfaring med oppsett og konfigurering av SCSI-kontrollere og enheter, anbefaler vi at du leser gjennom bolken i boka.

3.4 Oppsett av ulike tilleggskort i PC

"Setup different PC expansion cards"

Vi skal kjenne til:

```
/proc/interrupts  
/proc/ioports  
/proc/pci  
pnpdump(8)  
isapnp(8)  
lspci(8)
```

Dersom du ikke har erfaring med oppsett og konfigurering av tilleggskort, anbefaler vi at du leser gjennom bolken i boka. `isapnp` er nå tatt vekk fra deler av testen, men ikke alle stedene, så vi bør fortsatt lese om bruken av kommandoen.

3.5 Konfigurere USB-enheter

- xHCI modules
- lspci
- lsusb
- /etc/usbmgr/
- usbmodules
- /etc/hotplug
- udev configuration files, utilities and documentation

Det siste punktet her, om `udev`, har kommet til etter at boka ble skrevet, og det er heller ikke helt klart for oss hvor mye som ev. blir forventet av kandidatene her. Temaet er nytt med Linux 2.6-kjernen og du kan lese mer om det her: <http://en.wikipedia.org/wiki/Udev>
<http://www.kernel.org/pub/linux/utils/kernel/hotplug/udev.html>

Boka har en kort og enkel gjennomgang. Det er lurt å kjenne til hva "uhci", "ohci" og "ehci" er.

Tidligere var det bare listet opp at vi burde kjenne til `usb-uhci.o` og `usb-ohci.o`. Nå står det listet at vi bør kjenne til "xHCI modules". USB-målene ser ut til å være noe oppdatert etter at boka ble skrevet.

4 Installasjon og pakkehåndtering (101)

Slide 14

4.1 Installasjon og pakkehåndtering

Vi må vite hvordan filsystemet er bygd og ha en forståelse for hvorfor og hvordan vi partisjonerer. Dette finner vi i kapittel 4 i boka: "Linux Installation and Package Management. Ellers finner vi stoff om partisjonering i kapittelet om installasjon av Linux i kurset Linux systemadministrasjon I.

Vi bør ha en idé om hvorfor partisjonering er lurt både for både sikkerhet og hastighet. En gammel tommelfinger-regel for å bestemme hvor mye virtuelt minne (swap) vi bør sette opp, er at du må ha like mye eller litt mer enn det du har av fysisk minne.

Vi bør også være kjent med Filesystem Hierarchy Standard (FHS). Her ser vi hva ulike monteringspunkt (kataloger) er beregnet til, og hvilke som er valgfrie.

- IDE/EIDE-hardisker (litt annerledes med devfs i 2.4-kjernen)
 - ♦ /dev/hda – Første harddisk på første kontroller
 - ♦ /dev/hdb – Andre harddisk på første kontroller
 - ♦ /dev/hdc – Første harddisk på andre kontroller
 - ♦ /dev/hdd – Andre harddisk på andre kontroller
- SCSI-disker (og SATA-disker)
 - ♦ /dev/sd[a-h] – Avhengig av SCSI-ID og plassering på SCSI-bussen
- RAID-systemer kan ha navn av typen
 - ♦ /dev/rd/c0d0p1 (kontroller, disk, partisjon)

Legg merke til at SCSI-disker blir nummerert fortløpende etter hvilken SCSI-ID de har. Slik at hvis du peller ut harddisken som står som nummer en, så har det som var disk 2 plutselig et annet device-navn. Vær nøye med at du vet hvilken disk som inneholder /.

I 2.4-kjernen kan du bruke devfs. devfs holder styr på scsi-id'ene til div. disk, cdrom, tapestreamere og slikt og sørger for at uansett hvor de står på kabelen får de riktig navn og nummer.

4.2 Diskenes partisjonsmuligheter

- Hver disk kan ha fire primærpartisjoner
- For å få flere enn fire partisjoner må vi opprette en extended partisjon
- Extended-partisjonen opptar en primærpartisjon, og inneholder de logiske partisjonene
- Du kan ha 59 logiske partisjoner på en IDE disk, 63 partisjoner totalt
- SCSI-disker kan ha opptil 15 partisjoner
- Det er mulig å ha flere, men da må du recompile kjernen og flytte litt på noen devicefiler
- Partisjonene 1 til 4 er primærpartisjoner. Logiske partisjoner starter alltid på 5

4.3 Partisjoner og størrelser.

- Hvor mange og hvilke størrelser partisjonene skal ha avgjøres av hva maskinen skal brukes til.
 - ♦ Maskiner uten vanlige brukere trenger gjerne en litt mindre /home. Legg merke til at en del gamle distribusjoner bruker /home/ftp og /home/httpd som rot for ftp og webserverinstallasjonene sine.
 - ♦ Filservere kan gjerne ha flere /home, må ha egen katalogstruktur under /home, som /home/a , /home/b
 - ♦ Mailservere bør ha en gedigen /var, man kan også ha /var/mail som en egen partisjon
 - ♦ De fleste servere bør ha /var/tmp som separat partisjon
 - ♦ /tmp skal alltid være en egen partisjon! Alt der bør slettes ved reboot. Tmpfs er en god ting.
 - ♦ Minst mulig skrijving på /
- De vanligste partisjonene i egnede størrelser:

Mountpoint	Størrelse (MB)	Bruksområde
/	150+++	Konfigurasjonsfiler, kjerne, roots hjemmområde
/tmp	450	Plass til temporære filer for hvem som helst
/var	2000	Logger, spool.
/var/tmp	1000	Temporære filer med en forventet levetid til lenge over neste boot
/usr	4000	Programmer og biblioteker. /usr/share brukes til delte datafiler
/home	10000	Brukernes hjemmeområder
/local	10000	Programmer som kan deles over NFS, ting man har kompilert lokalt.
Swap	*	Virtuelt minne

*) Fysisk minne + swap > virtuelt minne alle programmene som skal kjøre samtidig trenger.

Det er en gammel tommelfinger-regel å ha virtuelt minne (swap) som tilsvarer to ganger fysisk minne. I praksis er ikke dette nødvendigvis det vi anbefaler i dag.

4.4 Installere boot-manager

Vi bør være fortrolige med å installere og konfigurere både LILO og GRUB, og ellers vite hvordan de fungerer, hva som er forskjellen (fordeler/ulempen) med å installere bootloaderen i henholdsvis superblokka eller MBR (Master Boot Record).

Filer vi må kjenne til:

```
/boot/grub/grub.conf
/boot/grub/menu.lst
grub-install
/etc/lilo.conf
lilo
```

Vi skal også kjenne til "first stage boot loader". Det er forklart kort i boka hvordan LILO/GRUB jobber i to "stages". Stage 1 legges i f.eks. MBR og har som eneste oppgave å laste stage 2.

Dersom vi velger å installere f.eks. LILO til superblokka, vet vi at det betyr at vi installerer til en begynnelsen av en partisjon (som inneholder et filsystem, og altså en superblokk), f.eks. `/dev/sda1`. Dersom vi installerer til MBR, er jo det begynnelsen av disken, f.eks.: `/dev/sda`.

4.5 Bygge/kompilere og installere programmer fra kildekode

Vi må beherske komprimering, dekomprimering og pakking, slik vi lærte i kapittelet om filverktøy i kurset Linux/UNIX grunnkurs. Kommandoer vi bør kunne mestre:

```
gunzip
gzip
bzip2
tar
```

I tillegg husker vi at vi bruker typisk

```
cd katalog
./configure
make
make install
```

for å bygge og kompilere programmer (GNU Automake).

I tillegg bør vi være kjent med hvordan vi håndterer delte biblioteker, kjenne kommandoer og filer og variabler som

```
ldd
ldconfig
/etc/ld.so.conf
LD_LIBRARY_PATH
```

Vi går gjennom et eksempel sammen. Vi henter ned et program, genererer `Makefile`, ser på targets, bygger/kompilerer og installerer. Så ser vi på hvilke biblioteker binæren(e) er kompilert mot. *Notér gjerne hva vi gjør underveis, selv om det viktigste er forståelse for hva som skjer.*

Boka har en fin bolk om samme tema.

4.6 Pakkehåndtering/administrasjon i Debian

Først var det kun én eksamensvariant, i hovedsak med RPM. Så var det i en periode mulig å velge mellom to ulike varianter, den ene med DPKG, den andre med RPM. I dag er begge deler i en eksamen.

Filer du bør kjenne til hva inneholder:

```
/etc/dpkg/dpkg.cfg
/var/lib/dpkg/*
/etc/apt/apt.conf
/etc/apt/sources.list
```

og så bør du kunne kommandoene:

```
dpkg
dselect
dpkg-reconfigure
apt-get
alien
```

Noen vanlige dpkg-kommandoer:

```
dpkg -i <filnavn.deb>
    installerer pakke(r)
dpkg -r <pakke> eller dpkg --purge <pakke>
    fjerner pakke(r)
dpkg -l "*" | grep ^ii og dpkg --get-selections
    viser installerte pakker
dpkg -I <pakke>
    gir informasjon om pakke(r)
dpkg -L <pakke>
    Liste ut filene installert av pakke(r)
dpkg -S <filnavn>
    Søke etter fil(er) for å se hvilken pakke de(n) tilhører
```

4.7 dselect

Du må også kjenne til `dselect`.

Det er bra forklart med hjelp og tips i programmet, men her er noen tips:

- Vær forsiktig med space/mellomromstasten
- +/- for å velge eller velge bort en pakke
- ? for å få hjelp
- Skråstrek '/' for å søke, '\' for å søke etter neste treff
- Ctrl+C for å gå rett ut når du har gjort noe dumt

`dselect` er i praksis erstattet av `aptitude` i dag, men skrivende stund er ikke dette gjenspeilet i LPIC1-testen.

4.8 apt-get

Den viktigste apt-kommandoen er `apt-get`. Her er noen eksempler på hvordan du bruker den:

`apt-get update`

Oppdaterer fillistene sine, finner ut hva som er siste versjoner av alt

`apt-get upgrade`

Oppgraderer alle pakkene dine til siste versjon

`apt-get dist-upgrade`

Samme som over, men installerer i tillegg nye ting som kommer

`apt-get install <pakke1 pakke2 ...>`

Installerer en eller flere pakker, løser avhengigheter og spør om det er greit å installere ev. pakker som må inn for at det du ba om skal virke

`apt-get remove <pakke1 pakke2 ...>`

Fjerner en eller flere pakker

`apt-get clean`

Tømmer lageret av `.deb`-filer

4.9 Søking etter pakker i Debian

- `dselect select` og `/` for å søke etter pakker, `^C` for å avbryte og gå rett ut
- `dpkg -l '*pakkenavn*'` for å lete etter pakker
- `apt-cache search <stikkord>` for å lete etter pakker
- `apt-cache show <pakkenavn>` for å få full beskrivelse av ei pakke når du vet pakkenavnet

Vi tar med litt ekstra om `dpkg` her (mer enn eksamen omfatter, men det blir veldig oppstykket å bare se på deler).

4.10 Vanlige RPM-kommandoer

Dersom du velger RPM-versjonen av eksamen 101, så bør du kjenne til vanlig bruk av RPM, de vanligste opsjonene:

- `rpm -i`, `rpm -Uvh`, `rpm -F`
- `rpm -qa`, `rpm -qc`, `rpm -qd`
- `rpm -e`

I tillegg bør du kjenne til

```
/etc/rpmrc  
/usr/lib/rpm/*  
rpm  
grep
```

Eksempler på bruk:

- `rpm -Uvh pakke-versjon.rpm`
- `rpm -qa foo*`
- `rpm -qa | grep -i stikkord`
- `rpm -ql pakkenavn`
- `cd /katalog ; rpm -Fvh *.rpm`

Sjekk boka for flere eksempler og forklaringer.

5 GNU & Unix–kommandoer (101)

Slide 24

5.1 GNU & Unix–kommandoer

Vi skal generelt være vant til å jobbe på kommandolinjen, og vi skal kunne vanlige kommandoer og mekanismer. De fleste har vi lært i kurset Linux/UNIX grunnkurs.

Kommandoer vi skal kunne:

```
.  
bash  
echo  
env  
exec  
export  
pwd  
set  
unset
```

Filer:

```
~/.bash_history  
~/.profile
```

Kapittel 5 i boka, "GNU and Unix Commands", tar for seg de ulike kommandoene og mekanismene vi skal kjenne.

Notér gjerne her hva kommandoene gjør og filene inneholder.

5.2 Filtre

Et program som leser fra `stdin` og skriver til `stdout`, kan vi kalle et filter. Disse bør vi kjenne til:

```
cat
cut
expand
unexpand
fmt
head
tail
hexdump
nl
paste
pr
sed
sort
split
join
tac
tr
uniq
wc
```

Tidligere var også `od` ("octal dump") med her, men nå ser det ut til at `hexdump` har tatt over (tilsvarer "`od -x`", på noen systemer er de det samme programmet).

5.3 Filhåndtering

Kommandoer vi bør mestre:

```
ls
cp
mv
rm
mkdir
rmdir
touch
find
file
```

I tillegg skal vi skjønne jokernotasjon ("file globbing"), dvs. bruk av *, ? og lister (f.eks. [a-z] og [^123]) i filnavn/katalognavn.

Vi skal kunne kopiere og slette filer og kataloger, også rekursivt (dvs. kataloger med innhold).

File globbing er omtalt som "Wildcards" i boka.

5.4 Omdirigering og rørlegging

Vi skal kjenne `tee` og `xargs` i tillegg til omdirigeringer med `<`, `<<`, `|`, `>`, `>>` og `` `` ("back ticks"). Oppsummering/eksempler:

```
kommando < innfil
```

```
$ cat <<EOT
>Hei
>på deg
>EOT
Hei
på deg
```

```
kommando > utfil
kommando >> utfil
```

```
kommando1 | kommando2
`kommando`
```

```
ls | xargs echo "Innholdet i katalogen: "
kommando | tee fil | less
```

`xargs` sender linjene som argumenter til den etterfølgende kommandoen. Du kan også styre hvilke argumenter som skal inn hvor.

`tee` tar imot data fra standard inn og lagrer til fil, men sender samtidig videre til standard ut.

5.5 Starte, overvåke og drepe prosesser

Kommandoer/konsepter vi bør mestre:

&
bg
fg
jobs
kill
nohup
ps
top
nice
renice
killall

Dette ble gjennomgått i kurset Linux/UNIX systemadministrasjon I. Se ev. gjennomgangen i boka for repetisjon.

5.6 Arbeide med tekstfiler og regulære uttrykk

I tillegg bør vi kunne arbeide med tekststrømmer og tekstfiler:

```
grep
regexp
sed

vi
    /, ?
    h, j, k, l
    G, H, L
    i, c, d, dd, p, o, a
    ZZ, :w!, :q!, :e!
    :!
```

Boka har en kort gjennomgang av både `sed`, `grep`, regulære uttrykk ("regexp") og `vi`. Før eksamen kan det være lurt å kunne `vi` litt i fingrene, sørge for at du husker de kommandoene som er nevnt ovenfor. Repeter gjerne kapittelet om regulære uttrykk og jokenotasjon fra kurset Linux/UNIX grunnkurs eller Linux/UNIX shellprogrammering.

6 Enheter, Linux filsystemer og FHS (101)

Slide 30

6.1 Enheter, Linux filsystemer og FHS

I dette kapittelet (kapittel 6 i boka), skal vi kunne lage partisjoner og filsystemer, vedlikeholde/sjekk filsystemene, montere og avmontere, sette og se på plassbegrensninger for brukere/grupper (quota), beherske filrettigheter og eierskap, lage linker (harde/symbolske) og finne filer og kataloger (kjenne til plasseringer).

For å laga partisjoner og filsystem:

```
fdisk  
mkfs  
mkswap
```

For å vedlikeholde og sjekke integriteten til filsystemet:

```
du  
df  
fsck  
e2fsck  
mke2fs  
debugfs  
dumpe2fs  
tune2fs
```

6.2 Montering, avmontering og plassbegrensning (quota)

For å kontrollere montering og avmontering av filsystemer:

```
/etc/fstab  
mount  
umount
```

Mount/fstab-opsjoner: `hide/unhide`, `user/users` osv.

For å administrere quota:

```
quota  
  
edquota  
  
repquota  
  
quotaon
```

Noter gjerne eksempler på bruk av quota-kommandoene over.

Oppgave: Slå opp i boka eller i manualsiden eller et annet sted, og skriv noter hva du typisk vil ha i `/etc/fstab` for å slå på quota for henholdsvis brukere eller grupper.

Bruk boka (s. 144 og utover) for å finne ut hva kommandoene gjør.

6.3 Eierskap, rettigheter og linking

For å styre tilgang til filer/kataloger ved hjelp av rettigheter eller flagg:

```
chmod  
umask  
chattr  
(lsattr)
```

Administrere eierskap:

```
chmod  
chown  
chgrp
```

Linking av filer (minner om snarveier i Windows og alias i Mac OS):

```
ln
```

Vi bruker stort sett symbolske linker (`ln -s`). Uten denne opsjonen får vi "harde" linker, dvs. at fila/katalogen får et nytt navn direkte, i stedet for en liten fil som peker til en annen fil/katalog.

6.4 FHS (Filesystem Hierarchy Standard)

Dette er også nevnt i kapitlet om installasjon og pakkehåndtering, men vi skal altså kjenne til FHS. Legg merke til hva ulike monteringspunkt (kataloger) er beregnet til, og hvilke som er valgfrie (optional).

Kurset Linux Systemadministrasjon I tar også for seg FHS.

Grunnen til at vi bør kjenne FHS, er at vi skal kunne plassere filer der de hører hjemme.

For å lokalisere filer:

`find`

`locate`

`slocate`

`updatedb`

`whereis`

`which`

`/etc/updatedb.conf`

6.5 whatis / apropos

I tillegg kan det være kjekt å kjenne til bruk av `whatis` og `apropos`.

`whatis` gjør det samme som `man -f`, dvs. viser en beskrivelse av hvilke manualsider som matcher det eksakte ordet du oppgir, og viser da en kort beskrivelse av alle de matchende sidene.

`apropos` gjør det samme som kommandoen `man -k`, dvs. søker etter nøkkelord for å vise relevante manualsider.

7 Vindussystemet X (101)

Slide 35

7.1 Vindussystemet X

Siste tema vi skal beherske, er X, eller The X Windows System. Dvs. at vi skal være i stand til å installere og konfigurere XFree86/Xorg og en fontserver.

Vi bør kjenne til kommandoene

```
xorgcfg  
xorgconfig  
XF86Setup  
xf86config  
xvidtune
```

Og vi bør kjenne oppbygningen/innholdet i filene

```
/etc/X11/xorg.conf  
/etc/X11/XF86Config  
.Xresources
```

Vi bør kjenne til hvordan man konfigurerer en display manager:

```
/etc/X11/xdm/*  
/etc/X11/gdm/*  
/etc/X11/kdm/*
```

Debian GNU/Linux har ikke eget runlevel for å bestemme om X skal starte eller ikke. Det har RedHat, med slekt og venner.

Vi bør da vite hva runlevel er, og kjenne til fila:

```
/etc/inittab
```

7.2 Installer og tilpass vindushåndterer og miljø

Vi bør kjenne til filene:

```
.xinitrc  
(.xsession)  
.Xdefaults
```

I tillegg bør vi kjenne til bruken av:

```
xhost  
(xauth)  
$DISPLAY
```

Det som står i parentes, er nyttig å kunne, men ikke lista opp hos LPI som nødvendig kunnskap for dette temaet.

8 Kjernen (102)

Slide 37

8.1 Kjernen

Vi skal beherske det å arbeide med kjernemoduler ("Manage/Query kernel modules runtime").

Disse filene og kommandoene skal vi kjenne innholdet i, eller kva de gjør. (Sjekk kapittel 13, "Kernel", side 259 i boka):

```
/lib/modules/kernel-version/modules.dep
```

```
modules.conf & conf.modules
```

```
depmod
```

```
insmod
```

```
lsmod
```

```
rmmod
```

```
modinfo
```

```
modprobe
```

```
uname
```

(Notér gjerne forklaring, stikkord eller eksempler over.)

8.2 Konfigurere og bygge kjerne/kjernemoduler

Rekonfigurere, bygge og installere, kjerne og/eller kjernemoduler:

```
/usr/src/linux/*  
/usr/src/linux/.config  
/lib/modules/kernel-version/*  
/boot/*  
make  
make targets: all, config, menuconfig, xconfig, gconfig  
               oldconfig, modules, install, modules_install,  
               depmod, rpm-pkg, binrpm-pkg, deb-pkg
```

Kommandoen `make` opererer med ulike *targets*. Dvs. at vi definerer mål/nøkkelord i `Makefile`. Der definerer vi hva som skal skje og avhengigheter.

Linux-kjerna (versjonene før 2.6) har bl.a. følgende *targets*:

- config
- menuconfig
- xconfig
- oldconfig
- dep
- modules
- install
- modules_install

8.3 Starte, initialisere og stoppe systemet

Vi må vite hva som skjer under oppstart, rekkefølgen ting skjer, og hvordan kjernen overgir kontrollen til `init` som igjen starter prosesser basert på `/etc/inittab`, som definerer ulike *runlevels*. For hvert runlevel overlates kontrollen til individuelle shellscript som starter tjenester, konfigurerer enheter osv.

Dette finner vi i kurset Linux systemadministrasjon I, i kapittelet om hvordan Linux er oppbygd. (Det står òg forklart i kapittel 14, side 277 i boka.)

Vi bør òg ha en formening om hva som defineres i filene:

```
/var/log/messages  
/etc/modules.conf eller /etc/conf.modules
```

Vi bør kjenne disse kommandoene:

```
dmesg  
shutdown  
init
```

I tillegg bør vi være kjent med LILO og GRUB, fordeler og ulemper med hver av dem, og hvordan vi konfigurerer de. Dette er vi også innom i installasjonsbolken i 101 ...

9 Utskrift (102)

Slide 40

9.1 Utskrift

Vi skal kunne konfigurere/sette opp CUPS og kjenne de ulike konfigurasjonsverktøyene (web/kommandolinje/andre programmer).

Vi bør kjenne til oppbygningen av og nøkkelord i `/etc/printcap`.

Vi skal kunne administrere jobber og skriverkøer med kommandoene:

```
lpc
lpq
lprm
lp
```

Så skal vi kunne skrive ut og se på køen:

```
CUPS configuration files, tools and utilities
lpc
lpq
mpage
a2ps
```

Spesifikasjonen sier at vi må kunne installere og konfigurere "local and remote printers", og nevner da:

```
CUPS configuration files, tools and utilities
/etc/printcap
/var/spool/cups/
/var/spool/lpd/*/
lpd
```

Det meste her er forklart i kapittelet om utskrift i kurset Linux systemadministrasjon II.

LPI ser ut til å ha oppdatert spesifikasjonene sine rett etter at boka kom ut og nå har de putta inn litt noen få linjer om CUPS. Hvor omfattende kunnskap som trengs, er uklart, siden de bare skriver "CUPS configuration files, tools and utilities" og `/var/spool/cups/`.

Trolig er det i alle fall lurt å vite om filene:

```
/etc/cups/client.conf
/etc/cups/cupsd.conf
/etc/cups/printers.conf
/etc/cups/ppd/navn-på-skriverkø.ppd
```

Videre har du kommandoene `lpadmin`, `lpoptions` og `lpinfo`. Videre har du i CUPS (Common Unix Printing System) tilgang til webgrensesnitt for aadministrasjon og alt på `http://localhost:631`

10 Dokumentasjon (102)

Slide 41

10.1 Dokumentasjon

For å kunne jobbe effektivt i Linux, må vi vite hvordan vi finner dokumentasjon og svar på spørsmål. Dette er vi innom i flere av kursene våre, det er òg et eget kapittel i boka (kapittel 16, s. 304).

10.2 Lokal dokumentasjon

```
$MANPATH
man
apropos
whatis
/usr/share/doc/*
```

10.3 Dokumentasjon på Internett

- LDP, <http://ldp.linux.no>
- Websider til "tredjepart"
- Newsgroups / Usenet
- E-postlister

I tillegg bør vi vite om følgende måter å gi meldinger til brukerne på:

```
/etc/issue
/etc/issue.net
/etc/motd
(wall)
```

LDP sin internasjonale hovedside er: <http://www.tldp.org/>

Under `/usr/share/doc/pakkenavn` (RedHat/Debian m.fl.) eller `/usr/share/doc/packages/pakkenavn` (SUSE) finner vi dokumentasjon/eksempelfiler for de ulike pakkene som er installert (typisk README-filer og slikt).

I tillegg finner vi som regel HOWTOS og slikt under `/usr/share/doc`.

10.4 Mer om manualsider

Kommandoen `man` går igjen flere steder. Vi bør kjenne til hvordan manualsidene er organisert i seksjoner, der f.eks. seksjon 1 er for vanlige programmer (bl.a. i `/bin` og `/usr/bin`) og seksjon 8 er for systembinærer/programmer som typisk krever at du er `root` (bl.a. `/sbin` og `/usr/sbin`). Seksjon 5 på Linux er for filformater/konfigurasjonsfiler

Vi bør kjenne til bruken av `more` og `less`, og at en av disse typisk brukes av `man` ved visning av sider (kan typisk defineres i variabelen `PAGER`).

Vi bør kjenne til `man -k/apropos` og `man -f/whatis`. I tillegg kan det være kjekt å bruke `man -d` for debugging dersom du ikke finner det du hadde håpet på.

Dette emnet er grundig behandlet i kursene Linux/UNIX grunnkurs og Linux systemadministrasjon I, men du finner også det du trenger i boka, kapittel 16.

På Linux finner vi dessuten ofte mer info i `info`-sidene (eksisterer side om side med manualsidene), men det er ikke nevnt her i LPIC1-spesifikasjonen.

11 Shell, skripting, programmering og kompilering

Slide 43

11.1 Shell, skripting, programmering og kompilering

Tittelen på kapittelet er lit overdreven i forhold til det begrensede innholdet. Emnene her har vi vært innom i kursene Linux/UNIX grunnkurs, Linux systemadministrasjon I og Linux/UNIX Shellprogrammering (for de som har vært på det). Relevant kapittel i boka er kapittel 17 (side 315).

Det første vi må ha oversikt over, er oppstartsfiler/konfigurasjonsfiler for `bash`:

```
~/.bash_profile  
~/.bash_login  
~/.profile  
~/.bashrc  
~/.bash_logout  
~/.inputrc
```

Vi bør kjenne til `function()` i `bash`.

Boka har en god presentasjon av hvordan vi lager funksjoner i `bash`.

11.2 Variabler i bash

Vi må vite hva miljøvariabler er, og hvordan vi definerer de. Kommandoer:

- export
- env
- set (shell built-in)
- unset (shell built-in)
- \$PATH

Videre bør vi kjenne til sekvenser og lister:

```
for i in `seq 1 100`; do echo $i; done

for variabel in heisann hoppsann falle ralle ra; do
    echo "Variabelen inneholder: $variabel"
done

for file in *.txt ; do cp $file ${file}.bak ; done
```

I bash kan også benytte C-lik syntaks som denne (i stedet for `seq`, men det er ikke med i LPIC-1 spesifikasjonen):

```
for ((i=0; i<100; i++)); do echo $i; done
```

Med opsijonen `-w` til `seq` kan du få f.eks. tre-sifrede tall som teller oppover:

```
for i in `seq -w 1 100`; do mkdir dir$i; done
```

11.3 Testing og løkker

For å kunne skrive shellscripts, er det en fordel å beherske løkke-konstruksjoner og testing:

- `while`
- `for`
- `test`

I tillegg må vi kunne `chmod` for å gjøre skriptet kjørbart.

Dette skal være kjent materiale fra tidligere kurs, men se gjerne gjennom kapittelet i boka.

12 Administrative oppgaver (102)

Slide 46

12.1 Administrative oppgaver (systemadministrasjon)

Temaet er gjennomgått i kurset Linux systemadministrasjon I. Noen av filene og kommandoene er ikke gjennomgått der. I boka finner du kapittel 18, "Administrative Tasks", på side 341.

Kapittelet inneholder:

- Bruker- og gruppeadministrasjon
- Brukermiljø og miljøvariabler
- Loggfiler
- Kjøre jobber i framtida: at og cron
- Sikkerhetskopiering
- Systemtid/systemklokka

12.2 Bruker- og gruppeadministrasjon

Systemfiler:

```
/etc/passwd  
/etc/shadow  
/etc/group  
/etc/gshadow
```

Kommandoer:

```
chage  
gpassword  
groupadd  
groupdel  
groupmod  
passwd  
useradd  
userdel  
usermod
```

Du bør ha oversikt over hva som ligger i hvilke konfigfiler og hva som er poenget med *shadow*-filer for passord.

Tidligere var disse kommandoene også med:

```
grpconv  
grpunconv  
pwconv  
pwunconv
```

Slå gjerne opp i manualsider eller boka og notér hva de ulike kommandoene gjør og hva som er i de ulike filene.

12.3 Brukermiljø og miljøvariabler

Filer:

- `/etc/profile`
- `/etc/skel`

Kommandoer:

- `env`
- `export`
- `set`
- `unset`

I tillegg er det fint å vite at `bash` gjerne har `/etc/bashrc` eller `/etc/bash.bashrc` i tillegg til `/etc/profile`. (Selv om dette neppe kommer opp i eksamensspørsmålene.)

12.4 Loggfiler

Både for sikkerhetsformål og administrative formål er det viktig å kunne arbeide med loggfiler.

For å konfigurere `syslogd`, loggetjenesten, editerer vi fila

```
/etc/syslog.conf
```

. (Denne så vi detaljert på i kurset Linux systemadministrasjon I.

Loggene finner vi typisk i `/var/log`-katalogen, enten direkte, eller i egne underkataloger.

Vi bør også skjønne prinsippet med loggrotering. Både RedHat og Debian GNU/Linux bruker `logrotate` som standard.

Ellers er

```
tail -f
```

en av de vanligste kommandoen for en systemadministrator.

Spørsmål:

Hvorfor er `tail -f` så vanlig? Hva gjør kommandoen?

`/etc/logrotate.conf` leser typisk filene i `/etc/logrotate.d/`.

12.5 Kjøre jobber i framtida: at

Kommandoen `at` bruker vi for å kjøre enkeltjobber på angitte tidspunkter. Eksempel:

```
$ at teatime
> echo hei
> ^D
$
```

Spørsmål:

Hvor kommer resultatet av kommandoen (gjelder både `cron` og `at`)?

Kommandoer:

- `at`
- `atq`
- `atrm`

Filer:

- `/etc/at.deny`
- `/etc/at.allow`

12.6 Repetere kjøring av jobb på angitte tidspunkter: cron

Mekanismen cron brukes hvis man ønsker å utføre en kommando gjentatte ganger (f.eks hver natt), og fungerer ved at en demon, `crond`, starter kommandoer ved behov. For å spesifisere hvilke tidspunkter og hvilken kommandoer det er snakk om bruker man kommandoen `crontab`:

```
crontab -l
    list ut crontab-filen (brukerens egen hvis ikke -u er angitt)
crontab -e
    editor crontab-filen med editoren spesifisert i variabelen EDITOR
crontab -r
    (remove) fjern hele crontab-filen
```

En slik crontab-fil er bygget opp av linjer, hvor hver linje er bygget opp som følger:

```
minutt time dag-i-måned måned dag-i-uke kommando
0-59 0-23 0-31          0-12 0-7
# Send en melding hver julaften kl. 17:00
0 17 24 12 * mail -s "Jul" dash%%Nå ringes julen inn.%
```

Filer:

```
/etc/anacrontab
/etc/crontab
/etc/cron.allow
/etc/cron.deny
/var/spool/cron/*
```

12.7 Sikkerhetskopiering

Kommandoer vi bør mestre:

- `cpio`
- `dd`
- `dump`
- `restore`
- `tar`

Gode rutiner for sikkerhetskopiering, bygger på analyse av hva som bør taes backup av og hvor ofte – og gjennom det en strategi for hvordan backup-media skal roteres (bygd på hvor ofte du tar full, inkrementell og/eller differensiell backup). For LPI level 1–sertifisering, bør du stort sett kjenne til syntaks og bruk av kommandoene som er nevnt ovenfor.

`tar` og `dd` skal være kjent fra tidligere kurs, men se gjerne i boka for en gjennomgang av hvordan vi bruker både disse kommandoene og `dump/restore` og `cpio`.

12.8 Systemtid / systemklokka

Kommandoer

- date
- hwclock
- ntpd
- ntpdate

Filer:

/usr/share/zoneinfo

/etc/timezone

/etc/localtime

/etc/ntp.conf

/etc/ntp.drift

Bruk gjerne manualsidene og boka for å finne ut hva de ulike filene inneholder og hva de ulike kommandoene gjør, gjerne med eksempel på bruk.

13 Essensielt om nettverk (102)

Slide 54

13.1 Essensielt om nettverk

Dette har et eget kapittel (19), "Networking Fundamentals", i boka, side 376). Ellers er det gjennomgått i kursene Nettverk og TCP/IP og Linux systemadministrasjon I.

Vi skal kjenne til de vanligste standard-portene for TCP og UDP, som definert i `/etc/services`. Spesifikt nevnt fra LPI sin side, er port 20, 21, 23, 25, 53, 80, 110, 139, 143, 161.

Vi skal kjenne til hva `/etc/services` inneholder.

Vi skal også skjønne CIDR-notasjon.

I tillegg skal vi kunne litt om forskjellen på IPv4 og IPv6.

13.2 Vanlige nettverkskommandoer

- ftp
- telnet
- host
- ping
- dig
- traceroute
- whois

Ellers skal vi kunne drive generell feilsøking og oppsett.

13.3 Oppsett og feilsøking

Vanlige filer:

- /etc/HOSTNAME og /etc/hostname
- /etc/hosts
- /etc/networks
- /etc/host.conf
- /etc/resolv.conf
- /etc/nsswitch.conf

Kommandoer:

```
ifconfig
ifup & ifdown
route
(ip)
dhcpcd, dhclient, pump
host
hostname
domainname
dnsdomainname
netstat
ping
traceroute
tcpdump
```

I tillegg bør vi kjenne til `init`-skriptene for nettverk som kjøres ved oppstart, typisk `/etc/init.d/network` eller `/etc/init.d/networking`.

`ip` fra pakken "iproute" tar over for `ifconfig` og `route` og har nye muligheter, men i skrivende stund er ikke LPI C-1-spesifikasjonen oppdatert med denne.

13.4 Sette opp Linux som PPP-klient

Filer:

```
/etc/ppp/options.*  
/etc/ppp/peers  
/etc/ppp/ip-up  
/etc/ppp/ip-down  
/etc/wvdial.conf
```

Kommandoer:

- pppd
- wvdial

Les gjennom denne bolken i boka, fra side 400. Temaet kan virke litt avskrekkende og kanskje irrelevant. Men dersom du skjønner hovedflyten, hvordan pppd bruker chatscripts og hva disse inneholder, så holder det i massevis.

14 Nettverkstjenester (102)

Slide 58

14.1 Nettverkstjenester

Innhold:

- `inetd` og `xinetd`
- Grunnleggende om Sendmail
- Grunnleggende Apache
- NFS og Samba
- Grunnleggende DNS
- Secure Shell (OpenSSH)

Temaene er omtalt i systemadministrasjonskursene våre. I tillegg er det kapittel 20 i boka, "Networking Services" (side 410).

14.2 inetd, xinetd og TCP wrappers

Vi skal kunne konfigurere og administrere `xinetd`, `inetd` og relaterte tjenester. Filer vi da bør kjenne til:

```
/etc/hosts.allow  
/etc/hosts.deny  
/etc/services  
/etc/xinetd.conf  
/etc/xinetd.log  
/etc/xinetd.d/  
/etc/inetd.conf
```

Vi skal altså kjenne til konfigurasjon av `inetd` og `xinetd`.

Spørsmål: Vet du forskjellen(e) på disse to?

14.3 Grunnleggende oppsett av Sendmail

Vi skal kjenne til litt grunnleggende bruk og oppsett av `sendmail`, noe vil være likt i andre MTAer.

Filer:

- `/etc/aliases` eller `/etc/mail/aliases`
- `/etc/mail/*`
- `~/.forward`

Kommandoer:

- `mailq`
- `sendmail`
- `newaliases`

LPIC–1–spesifikasjonen ser ut til å være oppdatert etter at boka kom ut og lister nå at vi skal kjenne:

```
Configuration files, documentation and commands for:
    postfix, qmail, exim and sendmail
/etc/mail/*
~/.forward
sendmail emulation layer commands
newaliases
```

`aliases`-fila og kommandoer som `newaliases` og `mailq` er stort sett felles. Faktisk tilbyr de fleste MTAer kommandoen `sendmail` også.

Sendmail har hovedkonfigurasjonsfilen `sendmail.cf`, men denne genererer vi fra en M4 makrofil, typisk `sendmail.mc`. (Ofte kan du bare endre `/etc/mail/sendmail.mc` og deretter skrive "make" i katalogen `/etc/mail/`.)

Postfix har konfigurasjonsfilene `master.cf` og `main.cf` (vi jobber mest med den sistnevnte). Disse kan typisk ligge i `/etc/mail/` eller i `/etc/postfix/`. Exim har typisk konfigurasjonsfilen `exim.conf` i enten `/etc/` eller `/etc/exim/`. QMail har typisk flere konfigurasjonsfiler og er heller laget for å være en direkte "sendmail replacement", slik postfix er det.

Gå gjennom filene og kommandoene ovenfor og skriv ned hva de gjør. (Bruk boka eller manualsidene.

14.4 Grunnleggende Apache–oppsett

Konfigurasjonsfila heter `httpd.conf` og ligger typisk i katalogen `/etc/apache` eller `/etc/apache2` (sistnevnte står i LPI-spesifikasjonen nå).

For å stoppe, starte og kontrollere tjenesten, bruker vi `apachectl` eller `apache2ctl` (f.eks. med argumentet "restart").

Selve webserver–prosessen heter:

`httpd`

Du skal også kjenne til hvordan du kan få Apache til å starte automatisk ved neste oppstart.

Denne gjennomgangen er veldig upresis og overfladisk, men det er også testen på dette punktet.

Man kan selvsagt også stoppe og starte Apache med init–skriptene som følger med installasjonen/distribusjonen.

14.5 Administrere NFS og Samba

Vi skal kunne administrere NFS og Samba (dvs. `smbd` og `nmbd`)

Filer:

- `/etc/exports`
- `/etc/fstab`
- `/etc/smb.conf`

Kommandoer:

- `mount`
- `umount`

Trolig får du ingen spørsmål som går i dybden her, men det kan være lurt å lese kjapt gjennom denne bolken i boka.

14.6 Grunnleggende DNS–oppsett

Vi skal kjenne til grunnleggende oppsett av DNS.

Det inkluderer filene:

- `named.conf` eller `named.boot`
- `/etc/hosts`
- `/etc/resolv.conf`
- `/etc/nsswitch.conf`

Og så har vi selve BIND–prosessen:

- `named`

Det skal ikke være nødvendig å gå inn på sonefil–syntaks og detaljer i `named.conf` til denne testen.

Du vil se at `named.conf` typisk kan være plassert under `/etc/bind/`.

14.7 Konfigurere Secure Shell (OpenSSH)

Vi skal kunne konfigurere og bruke SSH.

Vi bør da kjenne filene:

- `/etc/hosts.allow`
- `/etc/hosts.deny`
- `/etc/nologin`
- `/etc/ssh/sshd_config`
- `/etc/ssh_known_hosts`
- `/etc/sshrc`

Og vi bør kjenne programmene:

- `sshd`
- `ssh-keygen`

Noen av konfigurasjonfilene her ser ut til å være feil i forhold til hvor du vil finne dem i virkeligheten, men de er tatt med slik de er listet opp i LPI sin spesifikasjon for LPIC-1.

15 Sikkerhet (102)

Slide 65

15.1 Sikkerhet

Vi skal kjenne til en del grunnleggende filer og kommandoer innen sikkerhet. Vi har vært innom temaet i kursene Linux videregående og Linux systemadministrasjon. Det er et eget "Security"-kapittel i boka (kapittel 20, side 446).

Vi bør kjenne til *TCP wrappers*, dvs. `tcpd` og filene `/etc/hosts.allow` og `/etc/hosts.deny` og bruk av disse.

Vi bør vite hvordan vi finner SUID/SGID-filer med `find`, og hvorfor det er interessant fra et sikkerhets-perspektiv. I tillegg bør vi kjenne til *password aging*.

Filer:

`/proc/net/ip_*`

Under `/proc/net/` finner vi filer som starter med `ip_` i filnavnet. Her får vi typisk ut informasjon fra kjernen om brannmur og ruting. Brannmurkoden heter Netfilter, kommandoen er `iptables`.

Det er i tillegg interessante filer for nettverket under `/proc/sys/net/ipv4/ip_*`. I disse kan vi endre systemets oppførsel, f.eks. slå av og på ruting, enten ved å skrive data til filene her direkte, eller ved kommandoen `sysctl`. (Dette får du trolig ikke på eksamen.)

15.2 Kommandoer relatert til sikkerhet

Kommandoer:

```
find
passwd
socket
iptables
netstat
nmap
ulimit
```

`nmap` er en såkalt "port scanner" som finner ut hvilke porter som er åpne og halvåpne på en eller flere maskiner.

Vi skal kjenne kommandoen `ulimit` som vi bruker for å sette begrensninger for prosesser (forbruk av minne, filstørrelse m.m.). Logisk sett burde vi da også kjenne til at vi typisk definerer `ULIMIT` i `/etc/login.defs` (uten at dette er nevnt i LPIC-1-spesifikasjonen). For mer info om `ulimit`, prøv `help ulimit`.

I tillegg bør vi beherske `quota` (som omtalt tidligere) og kommandoen `usermod`

16 Avslutning og oppsummering

Slide 67

16.1 Avslutning og oppsummering

Da har vi kommet til veis ende. Spørsmål vi gjerne vurderer er da om vi har nådd målsetningen for kurset.

Noen som har spørsmål sånn på tampen?

Før du drar hjem, er det fint om du fyller ut kursevalueringen på <http://kurseval.linpro.no/>

Ved din tilbakemelding kan vi stadig forbedre oss.

Har du spørsmål kommentarer til kursinnholdet i etterkant av kurset, kan du sende oss e-post på kurs-hjelp@linpro.no

Har du andre spørsmål eller henvendelser om kurs eller videreutdanning, ta gjerne kontakt på ***kurs@linpro.no***.

Siste versjon av vår kurskalender skal du alltid kunne finne under <http://www.linpro.no/kurs>

16.2 Sluttoppgaver (så langt dere rekker)

1. Sortér innholdet av filen `/etc/passwd` alfabetisk med hensyn på Fullt navn/gecos-feltet. Lagre resultatet til `testkatalog/passwd_sortert` (Hint: `man sort`).
2. Bytt eier av `passwd_sortert` til `root`. Gjør at fila bare blir lesbar av eier og gruppe. (Hint: `chown` og `chmod`)
3. Start et shell som superbruker (`root`). På hvilke måter kan du gjøre det?
4. Dersom du ønsker å starte en prosess automatisk ved oppstart, hvordan kan du gjøre det? Her finst det flere måter, og det finnes spesielle kommandoer for ulike Linux-distribusjoner som hjelper deg. F.eks. har vi `/usr/sbin/update-rc.d` i Debian/GNU Linux. I RedHat m/slekt og venner har vi typisk `chkconfig`. Tillegg finnes det ulike grafiske verktøy. **Men kan du beskrive hvordan du gjør det manuelt?** (Er samme prinsipp i alle såkalte System V-UNIX-varianter.)
5. Samarbeid med sidemannen, logg inn på hverandres maskiner med SSH, overfør filer fram og tilbake med SCP.

Dersom du rakk alle oppgavene over, bruk `man` og finn ut hvordan `alias` virker.

Når dere er ferdige for dagen, kan dere slette alle katalogene og filene dere lagde med `rm -r`, ev. ta med `f`-opsjonen.

