

### 6.3.4 Linux

*Mottó: Linux is like a wigwam, no bills, no windows, Apache inside.  
A Linux olyan, mint az indián sátor, nincs számla, nincs ablaka, és az apacsok bent vannak.*

1999 derekán a multinacionális óriáscégek már olyan szervereket is szállítanak, amikben a Linux operációs rendszert installálják. Tapasztalatok és tesztek bizonyítják, hogy ez az egyik leggyorsabb hálózati operációs rendszer –ugyanakkor a használatát szabályozó licenz feltételei szerint szabadon terjeszthető. A Linux és idősebb testvére a UNIX a többfelhasználós operációs rendszerek családjába tartozik. Az **IEEE** az USA szabványügyi hivatalának, az **ANSI**-nak meghatározott egy UNIX szabványt. Ezt a szabványos UNIX „verziót” nevezték el **POSIX**-nek (Portable Operating System Interface for Computer Environments) és az **IEEE 1003** jelölést kapta. A szabvány meghatározza, hogy egy UNIX típusú operációs rendszernek hogyan kell működnie, definiálja a rendszerhívásokat, az interfészeket, stb. A Linux fejlesztői a kezdetektől követik a **POSIX** szabványt. Számos, a Linux alatt használt hálózati segédprogram és démon a **BSD** (Berkeley Software Distribution) UNIX csomagjából származik. Az a szerzői jogi szerződés, amellyel a U.C. Berkeley egyetem mindenki számára elérhetővé teszi a **FreeBSD** UNIX alaprendszert nem azonos a Free Software Foundation (FSF) termék, a GNU GPL tartalmával.

A Linux létrejöttét jóval megelőzően a UNIX első változata 1969-ben született az AT&T Bell Labs-nél. Mintegy öt évvel később a rendszer az akadémiai közösség és az egyetemek általánosan használt operációs rendszerévé vált. 1972-ben Dennis Ritchie kifejlesztette a C-nyelvet, majd egy évvel később a UNIX-ot újraírták C-ben. 1975-ben a Berkeleyyn is elkezdték a fejlesztést. 1983-ban az **AT&T** forgalomba hozta a System V Release 1-t, megjelent a **TCP/IP**, a következő évben hozták létre a POSIX szabványt. 1988-ban alapították az **OSF**-et (Open Systems Foundation), ami a nyílt rendszerek elterjedését volt hivatott segíteni. Tíz évvel ezelőtt jelent meg a PC-re Írt **SCO** (Santa Cruz Operation) UNIX **SVR4**. Ennek a sorozatnak újabb folytatása az **OpenServer R5** és a **UnixWare**. Közben 1981 augusztusában az IBM bemutatta a **PC**-t. Az **Intel** egyre nagyobb teljesítményű mikroprocesszorokat hozott forgalomba. Az évtized vége felé megnőtt az igény egy Intel platformon működő PC-s UNIX iránt. Ekkor kezdődött a Linux története. 1991 nyarán egy finn egyetemista, Linus Torvalds nem elégedett meg az akkoriban beszerezhető Minix által nyújtott szolgáltatásokkal és elhatározta, hogy létrehoz egy jobb operációs rendszert. Először Minix alatt fejlesztett **assembly**-ben, majd rövidesen áttért a **C nyelv**re. Ezután jelentős lépés történt: 1991 októberében Linus az **Interneten** meghirdette az első Linux verziót a. 0.02-est. **Önkéntes programozók** kapcsolódtak be a fejlesztésbe. Ez az **együtműködés**, a világ letehetségesebb programozóinak összefogása biztosítja azóta is azt a gyors sikert és népszerűséget, amit ez a mindenki operációs rendszere kivívott magának.

Magyarországon 1993-ban kezdett el terjedni a Linux, mert ekkorra kötötték be a felsőoktatási intézmények nagy részét az Internetbe. 1994 márciusában megjelent az **1.0.0** számú kernel. Innentől kezdve három részből álló verziószámot használnak. 1999 elején megjelent 2.0.0 változat, a mai napon 1999. június 24-én a KFKI FTP-szerverén az 1999. június 22.-i keltezésű, 2.3.8 verziószámú legújabb kernel patch már megtalálható! A Szabad Szoftver Alapítvány (FSF) forráskódjából és az igen sokféle alkalmazásból különféle rendszerek (disztribúciók) állíthatók össze. Ezek közül a legnépszerűbbek a RedHat, a Debian és a SuSE. A RedHat a 6.0, a Debian az 5.1, a SuSE pedig a 6.1 verziónál tart. Ezek szabadon letölthetők az Internetről, megvásárolhatók CD-n (Csak az előállítás költségét kell megfizetni.), de gyakran megtalálhatók szaklapok CD mellékletein is. A Debian disztribúció több mint 1500 programot tartalmaz. Mintegy háromszáz önkéntes fejlesztő folyamatosan, így gyorsan fejlődik. Az egyes programok felújíthatók (upgrade) anélkül, hogy az egyedi konfigurációs fájlok elvesznének, ill. hogy újraindítanánk a rendszert.

#### 6.3.4.1 Mire használható a Linux?

##### Ideális Internet kiszolgáló

- WWW szerver (Az Apache nevű web-szerver szoftver a legelterjedtebb a világon. A Microsoft Hotmail is ezt használja egy Sun szerveren!)
- FTP szerver (A gépünkön közérdeklődésre számot tartó, hasznos programokat és információkat, tananyagokat tárolhatunk gyors letöltés céljára.)
- Tűzfal (Véd a hackerektől, a betörőktől, akik szórakozásból, vagy információszerzés céljából, engedély nélkül akarnak belépni a hálózatunkba.)

##### Nagy teljesítményű LAN kiszolgáló

Ingyenes NetWare fájlserver, printszervert, ami kiegészítő szoftverek segítségével minden felhasználó részére elérhető lehet.

**Alkalmos korszerű irodai alkalmazások** (okmány és kiadványszerkesztők, táblázatkezelők, ill. prezentációkészítők) **futtatására.**

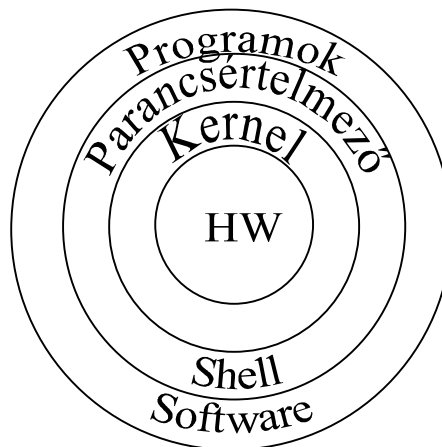
A Star Office, az Applixware és a Corel Office a Linux alatt használható változatai tartalmazzák a MS Office-ből ismert szoftvereszközöket. Az ApplixWare a SuSE disztribúcióban is megtalálható. A Star Office és az ApplixWare házi használatra ingyenes, hivatali használatra mindhárom jóval olcsóbb, mint a MS Office. A Linux operációs rendszer ezeken kívül természetesen alkalmas még szoftverfejlesztésre, tudományos számításokra, multimédiaalkalmazásokhoz, és játékokra is.

### 6.3.4.2 Milyen hardver kell a Linuxhoz?

Ha csak a **karakteres** felületet akarjuk használni egy kliens gépen, akkor elegendő egy 386SX processzoros gép 4 MB RAM-mal és egy 40-100 MB HDD-vel –de köztudott, hogy létezik olyan Linux verzió is ami XT-n is működőképes! A SuSE 6.1 verzióját számos változatban sikerült telepíteni. Az első változat mindössze 40 MB-ot foglalt el egy notebook gépen –beleértve néhány alprogramot, mint pl. vi, mc. A teljes változat egy Pentium166MMX processzoros, 16 MB RAM-mal és 1.2 GB HDD-vel felszerelt gépre került fel 4 db CD-ről. A teljes SuSE Linux változat karakteres felülettel való használatához egy Pentium processzoros, 32 MB RAM-mal és legalább 2,1 GB merevlemezzel felszerelt gép használatát ajánljuk. Ezzel persze a lehetőségek még mindig nem merültek ki, hiszen rendkívül sok szoftvert írtak Linuxra és az erre a célra átírt nagy szoftvereszközök száma is egyre gyarapszik. (Pl.: Oracle 8, Corel Office, stb.)

### 6.3.4.3 A Linux szerkezete

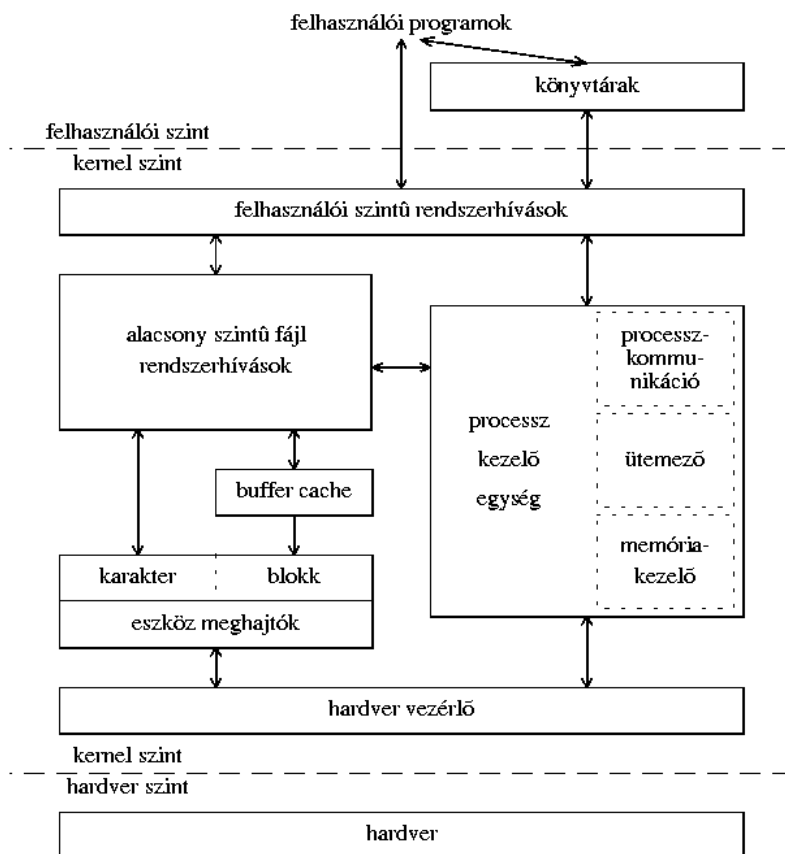
Minden disztribúció azonos felhasználói felületet (user interface) használ. Ehhez kapcsolódhat a nagyszámú felhasználói és segédprogram, amik többfelhasználós (multiuser), többfeladatos (multitasking) rendszer alkalmazását biztosítják.



30. ábra

A hardverhez kapcsolódó legbelső rész a **kernel**. Ez végzi a rendszer erőforrásainak megosztását és a futó processzek ütemezését, valamint egy „hordozható” interfészt biztosít az adott számítógépben lévő hardver felé. A Linux kernel nagyon gyorsan fejlődik, mind több új hardverhez található támogatás, azaz eszközmeghajtó. Az újabb változatok ezen kívül gyorsabbak, jobban kidolgozott a folyamatvezérlés, inkább mentesek a programhibáktól. Ezért a Linux felhasználók többsége rendszeresen cseréli a régebbi kernel verziót újabbra. Az új kernel verziók és kiegészítései (patch) C nyelvű forráskódban tölthetők le, pl. az [ftp.funet.fi](http://ftp.funet.fi) URL-ről, vagy más FTP helyről. Egy adott gép hardver felépítése eléggé különleges is lehet. Pl. előfordulhat, hogy szükségünk van egy nagy kapacitású DAT-ra, mellette DVD-re, ugyanakkor a hangkártyánk Gravis Ultrasound típusú, a monitorvezérlőnk AGP porton működik, a hálózati kártyánk pedig ismeretlen gyártmányú. A kiválasztott kernel forráskódot ezekhez a hardver eszközökhöz kell lefordítani, ahogy mondják, ezeknek az eszközöknek a támogatását be kell fordítani a kernelbe. A fordítást a **make config** paranccsal kezdeményezzük.

A gép indításakor az „Uncompressing Linux...” szavakkal kezdődik az a folyamat, amikor a kernel átveszi a hardver irányítását.



A kernel körül helyezkedik el a **shell** (héj, burok) ami biztosítja a kapcsolatot a felhasználó és a kernel között. Többféle lehet. A három legfontosabb: a Bourne shell - nevét a szerzőjétől kapta -, a Korn féle shell és a C-shell. A Bourne shell újabb változata Bourne Again Shell, a **bash**. Ha egy SuSE 6.1 Linux alatt beírjuk a **man bash** parancsot, akkor 60 oldalnyi leírást kapunk róla. Megtudjuk, hogy a GNU Bourne Again Shell az IEEE Posix Shell and Tools Specification (IEEE Working Group 1003.2) szabványos megvalósítása. A shell scriptek ("intelligens batch fájlok") megírásához, a shell script nyelv használatához a **Shell Grammar** ad segítséget. A parancssoros felületen kívül a Linux is rendelkezik **GUI**-val (Graphic User Interface), azaz grafikus felhasználói felülettel. Ez az **X-Window**. Egymás után jelennek meg az erre épülő különböző grafikus munkakörnyezetek, pl. a **KDE** (K Desktop Environment) és a **Gnome**.

#### 6.3.4.3.1 A fájlrendszer

A Linux újabb verziói a Second Extended, azaz **ext2** fájlrendszert alkalmazzák. Ez a fizikai diszket logikai részekre bontja. Mivel az MS DOS által is használt, partíciókra vonatkozó információkat a gép ROM BIOS-ának meg kell adni, ezért az ext2 fájlrendszer szintén a merevlemez első szektora, az MBS (Master Boot Sector) utolsó 64 bájtot használja a partíciókra vonatkozó adatok tárolására.

#### 6.3.4.3.2 A logikai lemez részei

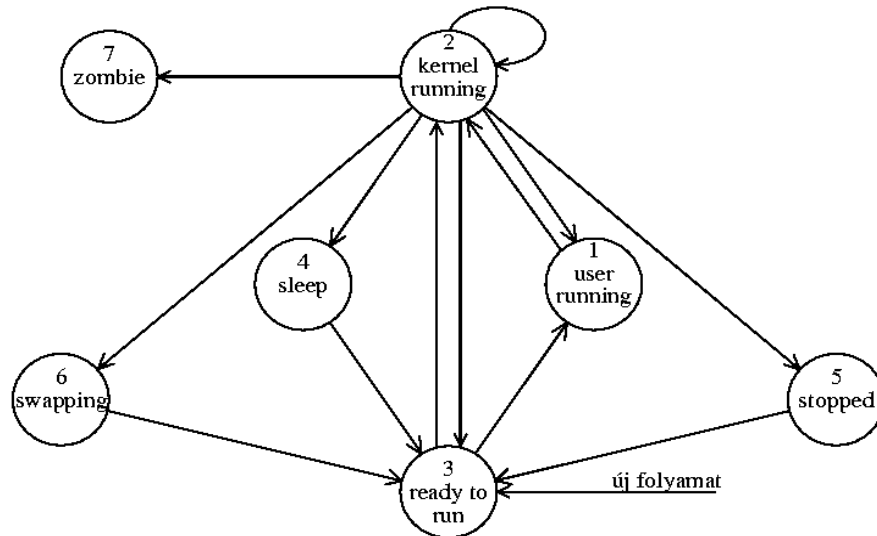
Mint az operációs rendszerek általában, a Linux is lehetővé teszi a fizikai háttértároló lemez logikai részegységekre való bontását. A lemezek részlemezre való bontásával ún. partíciókat kapunk. A Linux operációs rendszer könyvtárai egy vagy több partíción helyezkedhetnek el. A fájlrendszer az adatokat blokkoknak nevezett egységekben tárolja. A blokkok elején van egy szuperblokk, amelyben a blokkra vonatkozó összes információt tároljuk.

#### 6.3.4.3.3 Folyamatkezelés

Mint már említettük a Linux többfeladatos és többfelhasználós rendszer. Ebből következik, hogy akár egy felhasználó is egy időben több programot futtathat. Az elindított program a **processz**, azaz folyamat (**task**), más megfogalmazásban egy végrehajtható fájl „élő” változata. Az „életre keltett” folyamatok szekvenciálisan hajtódnak végre, azaz a felhasználó csak akkor kapja vissza a készenléti jelet, ha a végrehajtás befejeződött. Lehetőség van háttérprogram (background process) elindítására is. Ilyen esetben visszakapjuk az parancsfeldolgozó promptját és újabb parancsot adhatunk a rendszernek. Egy speciális háttérprogram a **démon** (daemon). Ezek nagy részét a Linux rendszer már a betöltődéskor elindítja. Számos démon fut a háttérben és figyelni pl. a lokális hálózatra belépőket, a nyomtatási kérélmeket, stb. Pl.: *inetd*, *ftpd*, *httpd*. A démonoknak átadott végrehajtá-

si kérelmek neve **job** (munka), ezekből jön létre a **queue**, (sor). A sorbarendezett feladatokat a rendszer egyenként hajtja végre.

#### 6.3.4.3.4 A folyamatok állapota



31. ábra

A Linux folyamatai a 31. ábra szerint hét féle állapotban lehetnek:

- |                   |   |  |
|-------------------|---|--|
| 1. User running   | = | A folyamat felhasználói módban fut.                  |
| 2. Kernel running | = | A folyamat kernel módban fut.                        |
| 3. Ready to run   | = | A folyamat futásra kész.                             |
| 4. Sleep          | = | A folyamat valamilyen eseményre várakozik.           |
| 5. Stopped        | = | A folyamat megállítható, majd újraindítható.         |
| 6. Swapping       | = | A rendszer egy memórialapot lemezre ír/betölt.       |
| 7. Zombie         | = | Exit után a folyamat elteszi a visszatérési értéket. |



A Linux egy folyamat állapotát a *task leíróban* tartja nyilván. A taskokra irányuló mutatókból egy tömb szervezhető. Ennek segítségével az összes folyamat állapota követhető. Ezt a 64 elemű tömböt task táblázatnak nevezik, ami így meghatározza a folyamatok lehetséges legnagyobb számát. Minden folyamat rendelkezik egy PID-je (Folyamatazonosító). A taskot ezzel lehet megcímezni, ill. ezzel lehet rájuk a rendszerhívásokban hivatkozni. Az éppen működő eljárásokat a **ps** paranccsal lehet ellenőrizni. Leállításukra a **kill** parancs szolgál.

#### 6.3.4.4 A memóriakezelés



A Linux kernel memória-nyilvántartó adatszerkezetei az Intel 80386 processzor memóriakezelésére épülnek. A gép fizikai memóriáját a Linux kernel kezeli, ezzel a felhasználónak nincsen semmi dolga. A többfelhasználós, többfeladatos operációs rendszerek az egyes memóriaterületekhez hozzáférési jogokat rendelnek. Ezeknek a területeknek a védelmét a mikroprocesszor támogatásával lehet megvalósítani. Az Intel először a 80386 típusú processzorába építette be a **protected mode** –ot (védett üzemmód). Ezt az üzemmódot alkalmazza a Linux is a memória szegmensekre osztására. Az üzemmód lényege az, hogy a mikroprocesszor szegmensregiszterében található adat, egy szegmenstáblában lévő szegmensleíróra mutat. Az ebben lévő címhez a szegmenscímet hozzáadva képződik a **folyamatos cím**. A folyamatos cím fizikai címmé alakítása a **lapozás**. A **demand paging** eljárás teszi lehetővé a memória-lapok, vagy a teljes memória háttértárolóra való mentését –azután ezeknek a szükség esetén való visszatöltését-, amivel a számítógép operatív tárának mérete látszólag megnövekszik. Ezekhez a műveletekhez a Linux által használt háttértárolókon egy külön partíció szolgál: a **swap**, aminek a mérete a fizikai memória nagyságával megegyezik, annak többszöröse, ill. nagyobb memóriaméret esetén azt 128 MB-ra egészíti ki.

#### 6.3.4.5 A futási szintek



Run level	Jelentés
0	Állj!
S	Egyfelhasználós üzemmód
1	Több felhasználó hálózat nélkül
2	Több felhasználó hálózattal
3	Több felh. hálózattal és XDM
4	Nincs felhasználva
5	Nincs felhasználva
6	Újraindítás

Az aktuális futási szint a **runlevel** paranccsal kérdezhető le. Pl.:

```
bitsy:~#runlevel
N3
```

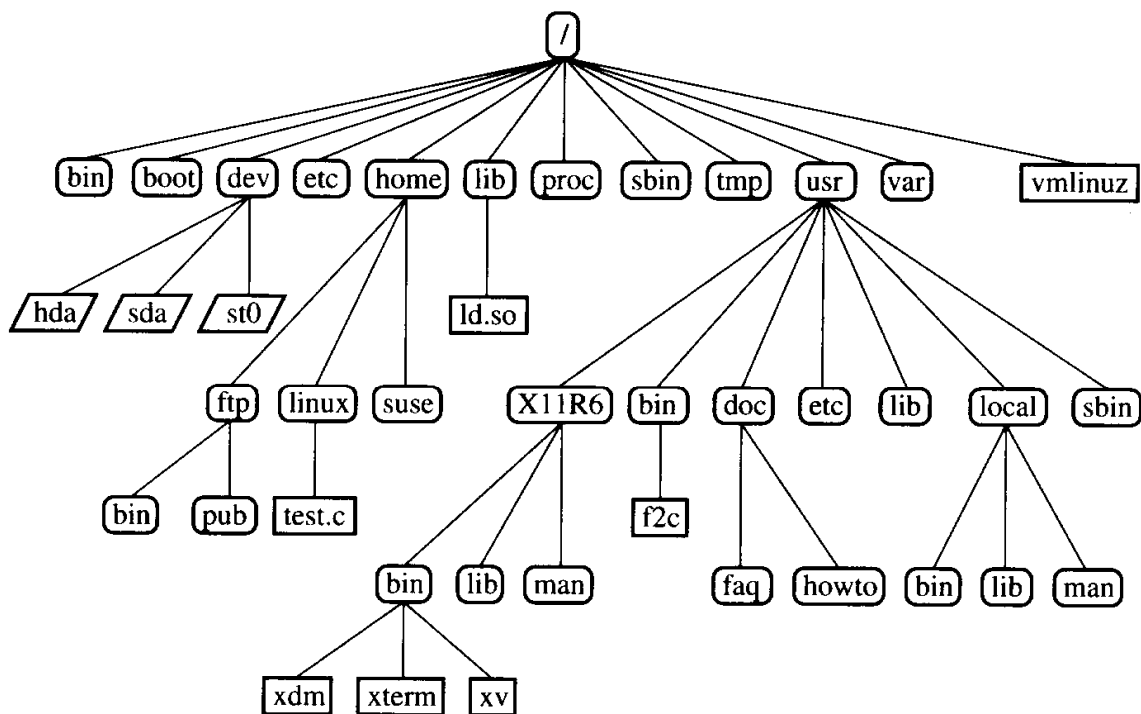
Tehát többfelhasználós hálózati szinten működik a rendszerünk, és az X-Window-t is futtatja. Ha most kiadjuk az **init 2** parancsot, akkor visszalépünk karakteres módba, azaz nem fog működni a grafikus felhasználói felület. Szervereken általában ezen a futási szinten dolgoznak a rendszergazdák. Ez a leggyorsabb és legmegbízhatóbb munkakörnyezet a rendszeradminisztráció elvégzésére. Az **init 0**-val állíthatjuk, az **init 6** paranccsal pedig újraindíthatjuk a rendszert. A **runlevel S** és a **runlevel s** egyenértékűek.

### 6.3.5 A SuSE Linux 6.1 könyvtárrendszere



A UNIX, ill. Linux operációs rendszer könyvtárstruktúrája is olyan mint egy fa, ami fejjel lefelé terebélyesedik. Számos könyvtárból áll amik nevei és tartalma egyes UNIX verziókban, ill. Linux disztribúciókban egymástól némiképp eltérnek. Pl. a weblap oldalait tartalmazó HTML fájlokat a Debian disztribúció a /var/www könyvtárban, míg a SuSE összeállítása a /usr/local/httpd/htdocs könyvtárban tárolja. A felhasználó szempontjából három kitüntetett könyvtár bír elsődleges fontossággal:

- root directory
- home directory
- current directory



32. ábra

#### 6.3.5.1 A fontosabb könyvtárak:

- /
- /home
- /dev
- /etc
- /usr/bin
- /sbin
- /usr/doc
- /usr/local/man
- /tmp
- /var
- /lib
- /proc

- A főkönyvtár, a könyvtárfa kiindulópontja
- Ehhez csatlakoznak a felhasználók könyvtárai
- A hardvereszközöket reprezentáló fájlok
- A legfontosabb konfigurációs fájlok
- Közhasználatú parancsok
- A superuser és a rendszerindító parancsok
- Dokumentációs fájlok
- Kézikönyv oldalak
- Ideiglenes fájlok
- Konfigurációs fájlok (Linkek az /usr-ből)
- Megosztott programkönyvtárak
- A processzek fájlrendszere

### 6.3.5.2 Mozgás a könyvtárrendszerben

Jelentkezzünk be a szerverünkre, mint rendszergazda! Hat különböző képernyőn –ún. virtuális konzolon dolgozhatunk, amelyek között az Alt-F1-Alt-F6 billentyűkombinációval kapcsolgathatunk. Válasszuk most ki a tty3 **virtuális konzolt** az Alt-F3 billentyűkkel! A monitoron ezt látjuk:

```

Welcome to SuSE Linux 6.1 (i386) – Kernel 2.2.5 (tty3)
bitsy login:                               A kettőspont után írjuk be:
root                                         Az enter leütése után megjelenik a
Password:                                   Ide írjuk be a jelszavunkat! Ezután jelenik meg a
Have a lot of fun...
Last login: Sun Jul 4 10:17:42 on tty3.
You have mail.
bitsy:~#

```

A prompt után írjuk be a **pwd** parancsot! Láthatjuk, hogy valójában melyik könyvtárban vagyunk.

```
/root
```

A helyi hálózatban csatlakoztatott most a *ty1* virtuális konzolon jelenkezzünk be a felhasználói nevünkkel! A készenléti jel most más lesz: `andy@bitsy:~> . pwd` parancsra a saját könyvtárunkba vezető útvonal íródik ki: `/home/andy`. Itt adjuk ki a **cd /etc** parancsot! A promptban megjelenik ha bejutottunk az `etc` könyvtárba `andy@bitsy:/etc >`. Ha itt paraméter nélkül adjuk ki a **cd** parancsot, akkor ismét visszajutunk a `home` könyvtárba. Hozzunk létre egy új könyvtárat!

```
mkdir public_html
```

Itt fogjuk majd tárolni a weblapunkat indító **index.html** állományt és a szükséges többi html-szövegfájlt, meg a grafikus fájlokat (képek). Egy floppylemezről másoljunk be pl. `.jpg` fájlokat, amik a weblapunkhoz szükségesek! A másolásra a `cp` parancs szolgál de kényelmesebben elvégezhetjük a műveletet a Midnight Commander segítségével. Ez nagyon hasonlít a már jól ismert Norton Commander programhoz. Indítása: **mc** (*Legyünk körültekintőek! A UNIX rendszerek különbséget tesznek a kis- és nagybetűk között!*)

### 6.3.6 Fájlinformációk

Jelentkezzünk be a szerverünkre *ty2* virtuális konzolon! Ha az Alt-F2 billentyűkombinációt alkalmazzuk erre máris itt a lehetőség. A *ty1*-*ty6* virtuális konzolok teszik lehetővé pl hat különböző program futtatását, amíg a *ty2*-n felhasználóként a fájlokat listázzuk, addig pl. a *ty3*-on mint rendszergazda egy segédprogramot telepíthetünk CD-ről! A *ty3*-ra az Alt-F3 billentyűkkel kapcsoltunk át.

```

bitsy login: andy
Password:*****
Have a lot of fun...
Last login: Sun Jul 4 18:41:25 on tty1
No mail.

```

Listázzuk ki a `home` könyvtárban lévő fájlokat!

```

andy@bitsy:~> ls
Mail
andy@bitsy:~> ls-l
total 1
drwx-----      2 andy   users      1024 Apr 20 14:46   Mail

```

A `/home/andy` könyvtár alatt egy `Mail` nevű alkönyvtár van.

```

andy@bitsy:~> cd mail
bash: cd: mail: No such file or directory
andy@bitsy:~> cd Mail
andy@bitsy:~/Mail>ls -l
total 5
-rw-----      1 andy   users      520      Mar 10 12:06 saved-messages
-rw-----      1 andy   users      1784     Apr 20 14:46 sent-mail
-rw-----      1 andy   users      1900     Mar 11 08:00 sent-mail-mar-1999. July

```

Ebben a listában látható a fájl típusa ( `-` normal), használati engedélye: `r` *read*, `w` *write*, a link számláló (1 link mutat a fájlra, tehát egyetlen fájlnevével lehet rá hivatkozni) a tulajdonos neve (`andy`) a csoport neve (`users`) a fájl

mérete bájtban, az utolsó módosítás ideje és végül a fájl neve. Ha az első helyen egy *d* karakter áll, akkor könyvtárral van dolgunk. Az *x* a végrehajthatóság (execute) jele.

-	<b>r</b>	<b>w</b>	-	<b>r</b>	-	<b>x</b>	<b>r</b>	-	-
Fájltípus		A fájl-tulajdonos jogai		A csoport jogai			Mások jogai		

### 6.3.7 A hardvereszközök jelölése

/dev/fd0	első floppy meghajtó
/dev/fd`	második floppy meghajtó
/dev/hda	első AT buszos HD
/dev/hda1-/dev/hda15	partíciók az első AT buszos HDn
/dev/sda	első SCSI merevlemez egység
/dev/sda1 - /dev/sda15	az első SCSI HD partíciói
/dev/sdb	második SCSI merevlemez e.
/dev/sgc	harmadik SCSI merevlemez e.
/dev/cdrom	link a CD-ROM meghajtóra
/dev/modem	link a modem portra
/dev/ttyS0 to /dev/ttyS3	0 –3 sz. soros portok
/dev/null	„adatnyelő” (nullperiféria)
/dev/tty1 - /dev/tty8	virtuális konzolok (AltF1-F8)

A hardver elemekre az eszköznevekkel hivatkozunk. Pl. ha egy floppy lemezzel akarunk fájlokat másolni, akkor először a rendszerrel tudatni kell, hogy az fd0 jelű eszközt is kezelni kell! ! (A mount és umount parancsok a háttértároló eszközön meglévő könyvtáraknak az állandó fájlrendszerhez való hozzákapcsolására, ill. az onnan való eltávolítására szolgálnak.)

```
mount /dev/fd0 /mnt
```

Ez azt jelenti, hogy a rendszer úgy tekinti, mintha az egyébként meglévő (de üres!) /**mnt** könyvtárban lennének a floppy diszken tárolt fájlok. Ha egy CD-ROM-ról akarunk programokat telepíteni, akkor hasonlóan kell eljárunk:

```
mount /dev/cdrom /cdrom
```

A CD-ROM –és bármilyen más, a fájlrendszerbe beillesztett háttértároló használata után az umount parancsot kell használnunk, amivel az eszközt kiiktatjuk a fájlrendszerből:

```
umount /dev/cdrom /cdrom
```

Példa egy másik eszköz, egy **PS-2 típusú egér indítására** Linux alatt:

```
gpm -t ps2 -m /dev/mouse &
```

Ha pedig modemmel akarunk kommunikálni, pl. be akarunk jelentkezni a vállalati szerverre, ezt megtehetjük a **dip** programmal a következőképpen:

```
dip -t          #Terminál emulátor indítása –párbeszédés módban
port/dev/ttyS1 #Ez jelenti azt, hogy a modem a DOS terminológia szerinti COM2 porton van.
speed 115200   #Az adatátviteli sebesség (a gép és a modem között) legyen 115200 bps
term          #Átkapcsolás a terminál emulátorra
ATDT 442253   #Hangkód tárcsázással összeköttetést létesítünk a szerverünkkel
```

### 6.3.8 Linux kezelési segédlet

A Linux operációs rendszer kezelésére számos parancs szolgál. Ezek közül a legfontosabbak:

<b>cat</b>	fájlok listázása
<b>cd</b>	könyvtárváltás
<b>chmod</b>	fájl hozzáférési jogosultságok változtatása
<b>chown</b>	Fájl tulajdonos megváltoztatása
<b>clear</b>	Képernyőtörlés
<b>cp</b>	Fájlmásolás
<b>date</b>	Dátum beállítása
<b>df</b>	Lemez foglaltság
<b>du</b>	Lemez használat könyvtáranként
<b>finger</b>	Felhasználó adatai

<b>grep</b>	Karakterlánc keresése
<b>less</b>	Szövegfájl kiírása előre-hátra lapozással
<b>kill</b>	Processz leállítás
<b>ln</b>	link létrehozás (egy fájlhoz több név )
<b>ls</b>	Fájlok listázása
<b>man</b>	Kézikönyv olvasása
<b>mkdir</b>	Könyvtár létrehozása
<b>more</b>	Szöveg tördelése
<b>mount</b>	Fájlrendszer felkötése létező könyvtárba
<b>mv</b>	fájl áthelyezése, átnevezése
<b>passwd</b>	Jelszó módosítás
<b>ps</b>	Processzek listázása
<b>pwd</b>	aktuális (munka-) könyvtár kiírása
<b>rm</b>	Fájltörlés
<b>rmdir</b>	Könyvtártörlés
<b>rpm</b>	programok telepítése rpm csomagból
<b>talk</b>	beszélgetés a felhasználók között
<b>umount</b>	a felkötött fájlrendszer lebontása
<b>who</b>	a szerverre bejelentkezett felhasználók
<b>write</b>	üzenetküldés

A **cd**, **kill**, **ls**, **mkdir**, **mount**, **umount**, **ps**, **pwd**, **runlevel** parancsok használatára a kapcsolódó témák tárgyalása során már bemutattunk példákat. Szükség van azonban még több parancs ismeretére!

### 6.3.9 A fájlműveletek parancsai

A **cp** parancs fájlok másolására szolgál. A parancsmondatban meg kell adni a forrásfájlt és a célfájlt vagy célkönyvtárat:

```
andy@bitsy:~/Mail>cp saved-messages /home/andy
```

A UNIX/Linux parancsok többsége nem ír ki semmit, ha sikeres volt a végrehajtás! **Az mv** parancs fájlok áthelyezésére és átnevezésére szolgál. Használata megegyezik a cp parancsával.

```
andy@bitsy:~/Mail>mv saved-messages messages.99.6
```

Az **rm** parancs fájlok törlésére szolgál.

```
andy@bitsy:~>rm messages.98.12
```

*Az eddigiekből láthatjuk, hogy a UNIX/Linux operációs rendszerben a 8+3 DOS fájlkonvenciótól eltérően használhatunk hosszú, több részből álló fájlneveket, amikben kis és nagybetűk, valamint számok és néhány írásjel is lehet.*

**Az rm** parancs fájlok törlésére szolgál. A törölt fájl többé már semmilyen módszerrel nem állítható helyre!

```
andy@bitsy:~>rm jegyzet
```

A parancs végrehajtása után a tartalomjegyzékből eltűnik a jegyzet nevű fájl. **A file** parancs és a fájlnev beírása után megkapjuk a kívánt információt.

```
andy@bitsy:~>file jegyzet
```

A válasz:

```
jegyzet: English text
```

**Szövegfájlkezelő** parancsok:

A **cat** parancssal fájlok tartalmát írathatjuk ki, ha a fájl nevét a parancs után írjuk:

```
andy@bitsy:~>cat jegyzet.uj.1
```

Ez egy igazi magyar jegyzet –csakis latin karakterekkel!

### Lapozás

Ha egy fájl hosszú, akkor a **cat** parancs az egészet kiküldi a képernyőre –de csak a végét láthatjuk. Ilyenkor használhatjuk a **more** és a **less** parancsokat. **A less mindkét irányban lapoz!**

```
andy@bitsy:~>cd /usr/man
andy@bitsy:~>ls
```



```
man1    man3    man5    man7    man9    whatis
man2    man4    man6    man8    mann
```

Olvassunk bele a whatis fájlba!

```
andy@bitsy:~>more whatis
```

Az enter billentyűvel lefelé görgethetjük a szöveget. A szövegolvasásból a **q** billentyűvel tudunk kilépni. Most használjuk a less parancsot!

```
andy@bitsy:~>less whatis
```

Ilyenkor görgetni a kurzormozgató, ill. a PgDn, PgUp billentyűkkel tudunk.

### Karaktorsor keresése

A **grep** parancs szolgál egy szó, vagy minta keresésére pl. egy fájlban. A /usr/man könyvtárban lévő whatis fájlban keressük meg a CGI karaktersort!

```
andy@bitsy:~>grep CGI whatis
```

Erre 24 olyan sort ír ki, ami a CGI betűket tartalmazza.

### Könyvtárak létrehozása és törlése

```
andy@bitsy:~>mkdir news
andy@bitsy:~>rmdir oldnews
```

### Engedélyek és tulajdonjogok

A **chmod** parancs a fájl-hozzáférési jogokat változtatja meg.

```
andy@bitsy:~>ls -l
-rw-r--r-- 1 andy users 57 Jul 21 09:01 jegyzet.uj.1
andy@bitsy:~>chmod go+rw jegyzet.uj.1
andy@bitsy:~>ls -l
-rw-rw-rw- 1 andy users 57 Jul 21 09:01 jegyzet.uj.1
```

Ezzel mindenki számára írhatóvá és olvashatóvá tettük a fájlt. A **–rw** használatával ezeket a jogokat elvehetjük. A **chown** parancs a fájl tulajdonosát és a felhasználói csoportot változtatja meg.

```
andy@bitsy:~>chown -cfv andras:tanar jegyzet.uj.1
owner of jegyzet.uj.1 changed to andras.tanar
```

A parancs után szoftverkapcsolókat használunk. A UNIX típusú operációs rendszerek megengedik több kapcsoló használatát is a **–** jel után! A **c** kapcsoló a változtatás kapcsolója, az **f** hatására a változtatás “csendben” történik, a **v** kapcsoló pedig az operációs rendszer visszajelzését váltja ki, közli, hogy mit hajtott végre.

### Szimbolikus linkek

Az **ln** parancs arra szolgál, hogy egy egy fájlra több, különböző néven is hivatkozhatunk, könnyebben elérhesük.

```
andy@bitsy:~>ln -s /usr/bin/arc archive
andy@bitsy:~>ls
archivejegyzet.uj.1
```

A könyvtárban megjelent a létrehozott archive nevű szimbolikus link.

### 6.3.10 A dokumentáció használata

A **man** parancs a manual szó rövidítése, ami magyarul kézikönyvet jelent. A man-t a UNIX rendszerek kezdettől fogva használják.

```
andy@bitsy:~>man talk
```

Az enter billentyű leütése után a következőket látjuk:

```
Reformatting talk (1), please wait
TALK(!)          UNIX Reference Manual          TALK(1)
NÉV
```

talk – beszélgetés egy másik felhasználóval

```
PÉLDA
talk miklos
```

## LEÍRÁS

A **talk** egy vizuális kommunikációs program, ami sorokat másol az egyik felhasználó termináljáról egy másik felhasználó termináljára. A beszélgetés kezdeményezésére a **talk miklos@bitsy.pszfs.hu** parancsmondattal kell válaszolni, s akkor a két részre osztott képernyő látható a beszélgető partnerek által beírt szöveg.

### 6.3.11 Egyéb parancsok

clear	képernyőtörlés
date	dátum lekérdezés
df	szabad hely a háttértárolón
du	a felhasznált hely a háttértárolón
who	egy szerverre bejelentkezett felhasználók felsorolása

### 6.3.12 A hálózat működésének ellenőrzésére szolgáló parancsok

Az **nslookup** suse.de parancsra megkapjuk az IP-címet: 194.212.123.200 . A **ping** 194.212.123.200 paranccsal meggyőződhetünk a hálózati gép bekapcsoltságáról és az összeköttetés minőségéről. A **traceroute** paranccsal kiírathatjuk a célszerverhez vezető hálózati útvonalat.

### 6.3.13 DOS és a Linux (UNIX) parancsok összehasonlítása

Művelet	DOS	Linux/UNIX
Fájlok listázása	dir	ls
Szövegfájl tartalmának kiírása	type	cat
Szövegfájl kiírása lapozással	type filenev more	more
Fájl másolása	copy	cp
Karakterlánc keresése	find	grep
Fájl átnevezése	ren	mv
Fájl törlése	del	rm
Könyvtár létrehozása	md (mkdir)	mkdir
Könyvtár törlése	rd (rmdir)	rmdir
Könyvtárváltás	cd (chdir)	cd
Fájlvédelem beállítása	attrib	chmod
Segítség a parancsokhoz	help	man
Dátum és idő lekérdezése	date, time	date
Szabad hely a háttértárolón	chkdsk	df
Fájl nyomtatása	print	lpr

### 6.3.14 A grafikus felhasználói felület

A UNIX típusú operációs rendszerek szabványos grafikus felhasználói felülete az X Window System. Ezt az egyik legismertebb informatikai nagyvállalat a **DEC** és az **MIT** az Athena Project keretében fejlesztette **X11** néven. Az első kiadás **X11R1** néven, **1987** szeptemberében már megjelent! A jelenleg legújabb, a hatodik kiadás megjelenése óta egy külön vállalkozás felelős a rendszer fejlesztéséért. Az X11 hálózatorientált termék. Az egyik –hálózatba kötött- gépen futó alkalmazások működése más, a helyi vagy a távolsági hálózatba kötött gépeken is látható. A PC számítógépekre készült ingyenes X-szerver implementáció, az **Xfree86**. Ezt egy 1992-ben alakult lelkes programozó csapat kezdte el fejleszteni. 1994-ben megalakult az Xfree86 Project, aminek a feladata a rendszer folyamatos fejlesztése és közreadása. A SuSE Linux sem rendelkezhetne e nélkül grafikus felülettel. A 6.1 verzióba az Xfree86 3.3.3.1 épült be, ui. az X11R6.3 a legfrissebb hivatalos X Window verzió. A SuSE Linux telepítő programja az általunk leírt módon a grafikus felhasználói felületet is telepíti. Ha a számítógépünk hardver elemei ismert gyártótól származnak (Pl. Intel mikroprocesszor, S3 VGA vezérlő, SoudBlaster hangkártya, stb.), akkor a telepítés befejezése után a sartx parancsra a grafikus felület máris elindul. Olyan esetben, ha pl. egy notebook gépre telepítünk, akkor még le kell futtatnunk a karakteres felületet használó **xf86config** nevű beállító programot, ami megkérdezi a különböző hardver elemek típusát, ill. gyártóját. Ha ezeket helyesen adjuk meg, akkor a grafikus felület könnyedén elindul. Ha már működik az X Window, akkor további finombeállításokat végezhetünk a **sax** nevű programmal. Ezt csak root-ként futtathatjuk! A rendelkezésünkre álló grafikus környezet a szokásos módon kezelhető és felfedezhető. Van minden, százféle háttér és képernyővédő, játék és segédprogram, Netscape, táblázatkezelő, kiadványszerkesztő, prezentációkészítő. Az ezeknél is újabb és izgalmasabb programok pedig az Internetről letölthetők, a szakfolyóiratok CD-mellékletein megtalálhatók. Csak azt kívánhatjuk, amit a SuSE verziót megalkotó német kollégák: **Have a lot of fun!** Sok izgalmas szórakozást! (Ez a fejezet a SuSE Linux 6.1 Installation, Configuration and First Steps c. kézikönyv Chapter 9, The X Window System kivonatatos fordítása.)

### 6.3.15 Rendszeradminisztráció

A Linux egy teljes értékű UNIX operációs rendszer. A felhasználók az operációs rendszerrel saját *shelleken* keresztül kommunikálnak. Gyakran szükség van arra is hogy új felhasználókat hozzunk létre, jelszavakat és konfigurációs fájlokat módosítsunk, új *scripteket* írjunk, új programokat telepítsünk. A rendszeradminisztrációs feladatok ellátásához szükségünk van a rendszergazda (root) jelszavára. A rendszergazda szinte mindent megváltoztathat a rendszerben ezért ezt a jelszót csak az ismeri, akinek tényleges feladata a rendszer kezelése (rendszergazda). Ha rendszergazdaként jelentkezünk be a szerverre, olyan shell-t kapunk, amiből kiadhatjuk az adminisztrációs parancsokat is. A készenléti jelet ilyenkor a # zárja le. (Pl.: bitsy:~#) A felhasználók promptja általában a \$ jellel végződik ([andy@bitsy~\\$](mailto:andy@bitsy~$)). Ha felhasználóként léptünk be a szerverünkre, akkor szükség esetén a **su** (super user) paranccsal kapjuk meg a root jogokat. A prompt ilyenkor megváltozik, hogy erre a helyzetre figyelmeztessen. A felhasználói jogokat az **exit**, vagy a Ctrl **d** paranccsal kapjuk vissza.

### 6.3.16 Rendszerbetöltés

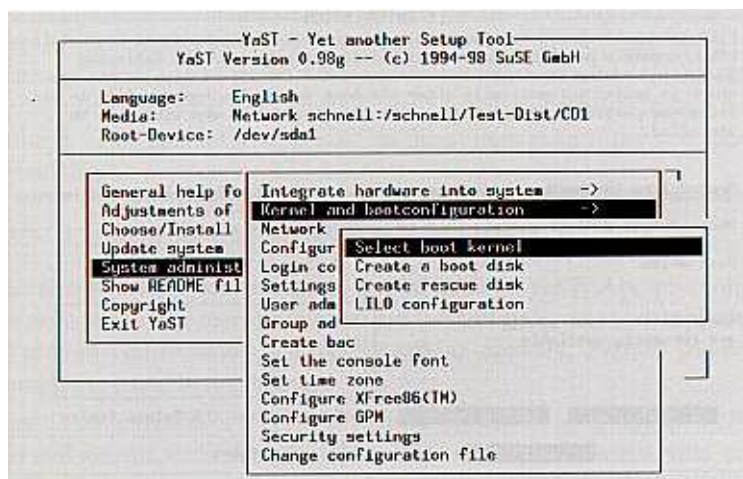
Betöltéskor az **/sbin** könyvtár **init**, „minden processzek atyja” végrehajtja az **/sbin/init.d** könyvtárban lévő **boot** scriptet. Ez elindítja a kernel-démont és betölti a szükséges modulokat. Az **/sbin/init.d/** könyvtárban található az azok a scriptek, amik a hardver elemek beállítását végzik. Az **init** bináris fájl részére az **/etc** -ben lévő **inittab** script írja le, hogy a különböző futási szintekhez mely scripteket kell beolvasnia. Ezután a partíciók ellenőrzése következik. Mindezt a monitoron követhetjük. Végül hozzákapcsolja (mount-olja) a könyvtárrendszert a főkönyvtárhoz, aminek a jele a / és aktivizálja a *swap partíciót*. A betöltés adatait beírja a **/var/log/boot.msg** fájlba és leállítja a kernel log demont. A SuSE Linuxban különleges fontossággal bír az **/sbin/SuSEconfig** fájl, amit bizonyos beállítások megváltoztatása esetén le kell futtatni. Ennek a disztribúciónak –a kiváló hardverfelismerő képességein kívül a másik nagy attrakciója az **/sbin** könyvtár progjamja, a **YaST**. (Yet another Setup Tool). Ennek a segítségével szinte minden beállítást és változtatást, úgyszólván a teljes rendszer- és hálózati adminisztrációt elvégezhetjük! A YaST megírja, ill. átírja helyettünk a konfigurációs fájlokat, amiket más, korábbi disztribúciókban mi írtunk meg! Segít új felhasználók és csoportok létrehozásában, karbantartásában, törlésében. Ez egy remek konfiguráló eszköz, azonban a haladóknak a Debian összeállítás megadja a szabadságot, ezzel rugalmasabb rendszer- és hálózati adminisztrációra ad lehetőséget –ám megköveteli a mélyebb ismereteket.

### 6.3.17 Hálózatadminisztráció

A UNIX/Linux rendszerek lehetővé teszik a TCP/IP protokollt használó hálózatokra való csatlakozást. Jelenleg már ez a legelterjedtebb hálózati protokoll, amit az Internet és a legtöbb helyi hálózat is használ. Ezt a protokollt eleve a különböző hardverplatformon működő és eltérő operációs rendszert alkalmazó számítógépek közötti kommunikációra tervezték a DARPA projekt keretében, még a hetvenes években. A protokollnak a (ISO/OSI) nyílt rendszerek működési modelljébe való illeszkedését az 5.1 részben már láthattuk. A TCP/IP további protokollokra bontható, mint pl.a TELNET, az FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), HTTP (Hyper-text Transfer Protocol), stb. További protokollok biztosítják a DNS (Domain name Service) és az NFS (Network File System) szolgáltatást. Az összes Internet Protokoll aktuális listáját az RFC 1011 tartalmazza. RFC=Request for Comment, A NIC, a Network Information Center dokumentumai.

#### 6.3.17.1 TCP/IP konfigurációs fájlok

Az **/etc** könyvtárban található az azok a konfigurációs fájlok, amik olyan hálózati információkat tartalmaznak, mint a gépek és domain-ek nevei, IP-címek, a hálózati kártyák adatai, a TCP és UDP port hozzárendelések, routing tábla. Ezeknek a tartalmát a SuSE Linuxban általában nem kell módosítanunk, mert a **yast** elvégzi a hálózati beállításokat is, ha a System Administration és Network Configuration menüpontokat választjuk, majd beírjuk a szükséges adatokat.



33. ábra

### 6.3.18 Hogyan tudhatunk meg még többet a Linuxról?

Egy Linux szerver üzemeltetése során számtalan problémával találkozhatunk. Ha tehetségünk engedi akár a kernel és a felhasználói programok fejlesztésében is részt vehetünk! Mint alkalmazók mindenképpen megéljük a