



6. LEKCIJAS MATERIĀLS

Saturs

Partīcijas (loģiskie diski), to veidošana.....	1
Sistēmas klonēšana lokāli un caur tīklu.....	3
Skriptu rakstīšana un palaišana.....	5
Attālināta administrēšana, uzraudzība.....	6

Partīcijas (loģiskie diski), to veidošana.

Loģisko disku veidošana nozīmē viena fiziskā diska sadalīšanu vairākos loģiskajos diskos. Partīcija ir blakusesošu bloku kopums, kas tiek uzskatīts par neatkarīgu disku. Loģisko disku tabula satur indeksus katrai loģiskajai vienībai.

Kāpēc būtu vajadzīgas vairākas partīcijas? Datu iekapsulēšanai. Piemēram, ja dati vienā partīcijā tiek bojāti, tad citas partīcijas paliek neskartas. Nodrošina efektīvāku diska telpas izmantošanu. Ir iespējams formatēt partīciju mainot bloka izmēru. Ja, piemēram, dati ir daudz mazu failu (mazāki par 1kb) un partīcijā izmantoti 4kb lieli bloki, tad uz katra faila tiek zaudēti 3kb. Respektīvi, tiek zaudēta aptuveni puse no bloka izmēra uz katru failu. Tādējādi pie formatēšanas vērtīgi norādīt aptuveno bloka izmēru, kas sevišķi svarīgi, ja ir ļoti daudz failu. Ierobežots datu pieaugums. Nošķeļot operētājsistēmas datus no lietotāja datiem, nodrošinās mums sistēmas stabilitāku darbību. Jo situācijās, kad lietotājs pārpilda diska telpu un netiek izmantotas kvotas, operētājsistēma var pārtraukt savu darbību diska vietas trūkuma dēļ.

Partīcijas veidojot jāņem vērā sekojoši ierobežojumi. Partīcijas nedrīkst pārklāties. Var tikt zaudēti dati un notikt citādas dīvainas lietas. Nevajadzēt būt tukšai vietai starp blakusesošajām partīcijām. Protams, ja tāds tukšums ir, nekas slikts nevar notikt. Vienīgi nepamatoti tiek zaudēta diska vieta. Diskam nav jābūt sadalītam loģiskajos diskos līdz galam. Var atstāt tukšu vietu, kur varēs izveidot loģiskos disku vēlāk. Partīcijas nevar pārvietot, bet izmantojot speciālu programmatūru iespējams mainīt to izmēru un veikt kopēšanu. Kursā aplūkotā *fdisk* utilīta neprot veikt šīs papildus darbības.

Partīciju veidošana nepieciešama, lai organizētu fiziskā cietā diska sadalīšanu vairākos loģiskos diskos, kur katram var būt sava failsistēma. Rezultātā tiek iegūti neatkarīgi diski, uz kuriem var tikt glabāti vienkārši dati vai instalētas dažādas operētājsistēmas. Tāda paša nosaukuma partīciju administrēšanas rīks pieejams arī uz Microsoft Windows sistēmām. No komerciālajām aplikācijām loģisko disku pārvaldībai bieži izmanto *Partition Magic*, kuram ir grafiskais lietotāja interfeiss un bagāts iespēju klāsts. Taču *fdisk* priekšrocības ir vienkāršums un spēja to darbināt no komandrindas. Turpmāk aplūkota tiks *fdisk* versija, kas darbojas uz Linux operētājsistēmām. Izmantoti uz Linux tiek arī šādi rīki, kā *parted* un *diskdrake*.



Datoram parasti ir divi IDE kontrolieri, tas nozīmē iespēju pieslēgt četrus cietos diskus, kuri iekārtu direktorijā apzīmējas sekojoši:

/dev/hda – pirmā IDEs kontroliera disks nr.1

/dev/hdb – pirmā IDEs kontroliera disks nr.2

/dev/hdc – otrā IDEs kontroliera disks nr.1

/dev/hdd – otrā IDEs kontroliera disks nr.2

Ja piemēram uz /dev/hda ir izveidotas četras partīcijas, tad tās apzīmēsies šādi:

/dev/hda1

/dev/hda2

/dev/hda3

/dev/hda4

Ar SATA un SCSI diskkiem situācija analogiska, vienīgi iekārtu faili parasti apzīmējas ar

/dev/sda1

/dev/sda2

utt.

Lai darbinātu *fdisk* rīku, komandrindā jāraksta *fdisk iekartasvars*. piem.,

fdisk /dev/hda

fdisk /dev/sda

```
[root@localhost etc]# fdisk /dev/sda

The number of cylinders for this disk is set to 30401.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
 1) software that runs at boot time (e.g., old versions of LILO)
 2) booting and partitioning software from other OSs
    (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): m
Command action
 a  toggle a bootable flag
 b  edit bsd disklabel
 c  toggle the dos compatibility flag
 d  delete a partition
 l  list known partition types
 m  print this menu
 n  add a new partition
 o  create a new empty DOS partition table
 p  print the partition table
 q  quit without saving changes
 s  create a new empty Sun disklabel
 t  change a partition's system id
 u  change display/entry units
 v  verify the partition table
 w  write table to disk and exit
 x  extra functionality (experts only)

Command (m for help): █
```

Lai iegūtu informāciju par izvēlēta diska loģiskajiem diskkiem programmā *fdisk* izvēlas komandu *p* (*print the partition table*).



Command (m for help): p

Disk /dev/sda: 250.0 GB, 250058268160 bytes
255 heads, 63 sectors/track, 30401 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x9f849f84

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	3824	30716248+	7	HPFS/NTFS
/dev/sda2	*	3825	7649	30720000	83	Linux
/dev/sda3		7649	8159	4096000	83	Linux

Command (m for help): █

Pārējo biežāk izmantoto komandu īss skaidrojums:

n – jaunas partīcijas izveidošana

d – partīcijas izdzēšana

a – norāda no kuras partīcijas ielādēsies sistēma

w – svarīgi izpildīt, kad viss izdarīts un vēlamies izmaiņas partīciju tabulā saglabāt

Kad partīcijas ir izveidotas, tām nepieciešama noformatēšana, lai reāli uz tām varētu izvietot datus. Formatēšanas procesā var tikt uzstādīti partīcijai dažādi failsistēmu tipi. Windows sistēmām tā parasti ir *FAT32* vai *NTFS* failsistēma, savukārt Linux sistēmām *ext3*, *ext4* un *swap* failsistēmas.

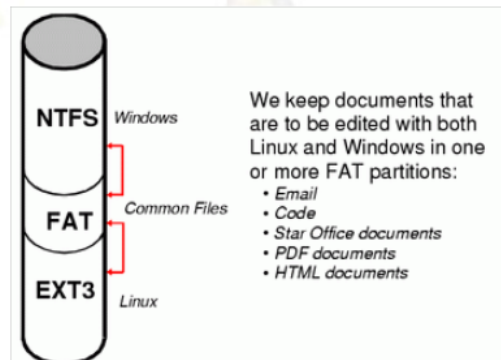
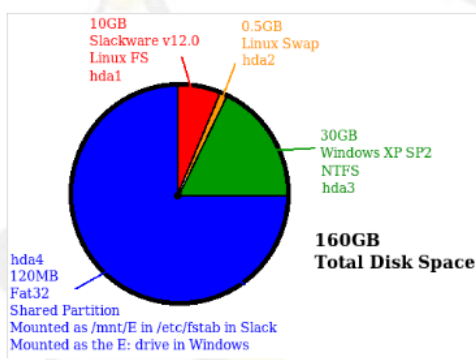
Lai veiktu formatēšanu no komandrindas izmanto, piemēram, komandas:

mkfs.ext3 /dev/hdb1

mkfs.ext4 /dev/sda1

mkfs.ntfs /dev/sda2

mkfs.vfat /dev/hda1



Sistēmas klonēšana lokāli un caur tīklu.

Svarīga lieta administrējot datortīklus, ir iespēja ātri, attālināti un vienkārši atjaunot vai dublēt diska partīcijas un uz tām esošas operētājsistēmas. Atvērtā koda pasaulē ir divi projekti, kurus ērti izmantot šim mērķim:

G4U (Ghost for Unix) - <http://www.feyrer.de/g4u/>

G4L (Ghost for Linux) - <http://freshmeat.net/projects/g4l/>



G4U ir vienkāršāks un darbojas praktiski jebkurā situācijā ar jebkuru cieto disku. *G4L* ir lietotājam draudzīgāks, jo sagatavoti speciāli skripti un pseidografika, kas atvieglo darbu ar to. Problēmas, kas bija saistītas ar SATA disku atpazīšanu, ir novērstas.

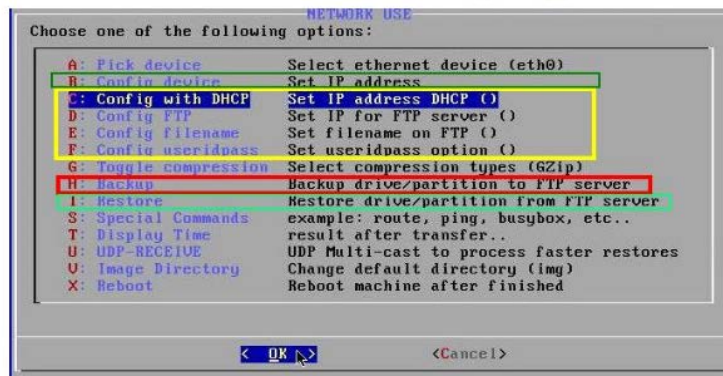
Programmas *G4L* darbības pamatā ir **FTP serveris** un **ncftppget**, **ncftpput**, **ntfscclone** un **dd** rīki. Informācijas rezerves kopiju **sūtīšanai/saņemšanai uz/no** servera izmanto **ncftpput/ncftppget**. **dd** kopē uz cietā diska esošos datus kā vienkāršu bitu virkni, neatkarīgi no failsistēmas. Lai nebūtu pārāk liels datu apjoms, to kombinēt kopā ar dažādiem arhivēšanas rīkiem. Šāda pieeja darbojas vienmēr, arī tad ja parādās jaunas failsistēmas. Ja nepieciešams darboties ar *NTFS* failsistēmas partīciju, tad noder **ntfscclone**, kas darbosies ātrāk kā **dd**. Papildus Internetā var palasīt informāciju par **UDPCast** risinājumu, kas izmanto *broadcast* kadrus, informācijas sūtīšanai.



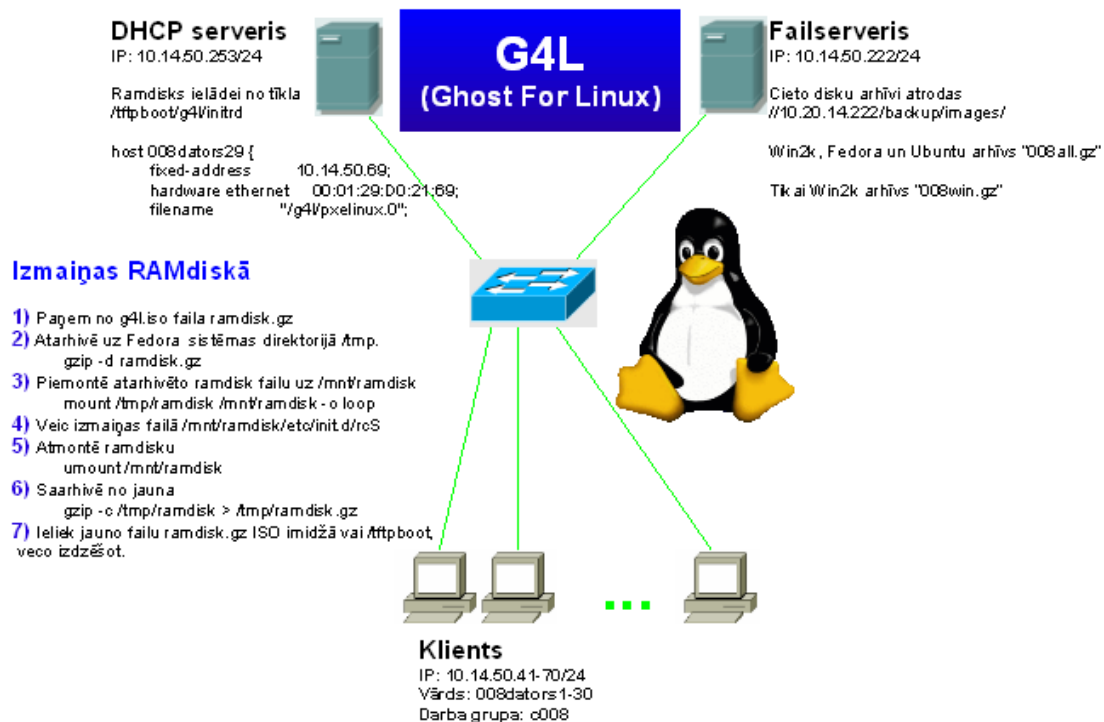
```
wd2 wd3 sd0 sd1 sd2 sd3.
-----
Welcome to g4u Harddisk Image Cloning V2.1!
-----
Commands:
* Upload disk-image to FTP:  [GZIP=1] uploaddisk serverIP [image] [disk]
* Upload partition to FTP:  [GZIP=1] uploadpart serverIP [image] [disk+part]
* Install harddisk from FTP: slurpdisk serverIP [image] [disk]
* Install partition from FTP: slurppart serverIP [image] [disk+part]
* Copy disks locally:       copydisk disk0 disk1
* Copy partitions locally:  copypart disk+part0 disk+part1
* List all disks:           disks
* List partitions:         parts disk
* See all devices:         dmesg
* This screen:             help

[disk] defaults to wd0 for first IDE disk, [disk+part] defaults to wd0d for
the whole first IDE disk. Use wd1 for second IDE disk, sd0 for first SCSI
disk, etc. Default image for slurpdisk is 'rwd0d.gz'.

Enjoy!                                     Send comments to hubert@feyrer.de
                                           Donate at paypal@feyrer.de !
                                           http://www.feyrer.de/g4u/
-----
g4u>
```



Tā kā *G4L* ir atvērtā koda projekts, ir ērti veikt izmaiņas sistēmas darbībā pastāvīgi un nodrošināt pilnīgi automātisku un attālinātu darbību konkrētai situācijai.



Skriptu rakstīšana un palaišana.

Skriptu nepieciešamība ir sevišķi noderīga izmantojot tīklā Linux sistēmas serverus. Dažādu automatisko procesu veidošanai un darbināšanai, skripti ļauj praktiski realizēt ļoti dažādus scenārijus. Skriptu sintakse šajā kursā netiks aplūkota. Skatīt piemēru! Idejiski ko līdzīgu var nākties realizēt eksāmenā ar tām komandām kuras kursa ietvaros ir aplūkotas.

Piemērs:

Palaižam teksta redaktoru pie viena izveidojot failu: **vi tiikls.sh**

```
#!/bin/sh
```

```
# Mans Linux skripts
```

```
clear
```

```
echo "Sveiki! Tīkla servisi tiks parstarteti!"
```

```
/etc/init.d/network restart
```

Saglabājam izmaiņas: **wq**

Piešķiram tiesības: **chmod 777 tiikls.sh**

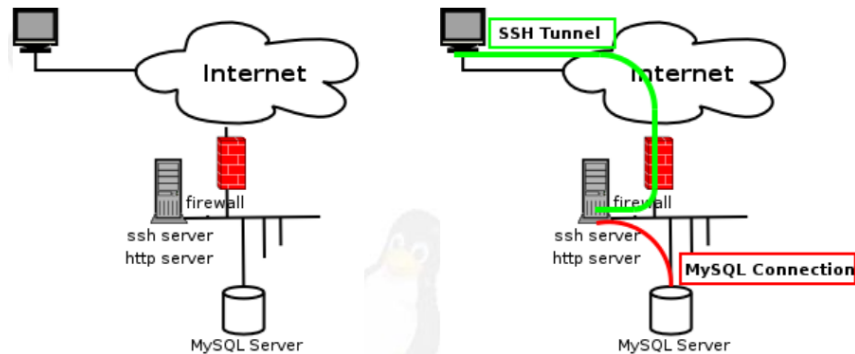
Izveidoto skriptu var palaiž ar **./tiikls.sh**

Vai noderīga ir arī loģiskās saites (symlink) izveidošana: **ln -s tiikls.sh /usr/local/bin/tiikls**

Skriptu var palaist no jebkuras vietas failsistēmas kokā ar komandu **tiikls**



Attālināta administrēšana, uzraudzība.



ssh – Drošs, tiek izmantota šifrēšana. Biežāk izmantotais komandrindas rīks. Ātra darbība.

```
192.168.10.1 - PuTTY
login as: root
root@192.168.10.1's password:
root:~# uname -a
Linux UMLlinux 2.6.23-rc1 #9 Fri Aug 10 00:42:14 CEST 2007 i686 unknown
root:~# set | grep SSH
SSH_CLIENT='192.168.4.71 2443 22'
SSH_TTY=/dev/pts/2
root:~# route -n
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.4.0      192.168.10.3   255.255.255.0   UG  0      0      0 eth0
192.168.10.0     0.0.0.0        255.255.255.0   U    0      0      0 eth0
root:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr DA:39:54:BB:6A:24
          inet addr:192.168.10.1  Bcast:192.168.10.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1925 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1524 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:166223 (162.3 KiB)  TX bytes:335076 (327.2 KiB)
          Interrupt:5
root:~#
```

telnet – autorizācijas informācija tiek pārsūtīta atklātā tekstā, tāpēc labāk neizmantot, vai arī jātaisa šifrēts tunelis pirms *telnet* izmantošanas.

```
cx Telnet 172.16.208.174
=====
vpn-server
=====

vpn-test login: tedy
Password:
Last login: Tue Jun 17 20:17:03 WIT 2008 from 172.16.209.120 on pts/1
Linux vpn-test 2.6.24-16-generic #1 SMP Thu Apr 10 13:23:42 UTC 2008 i686

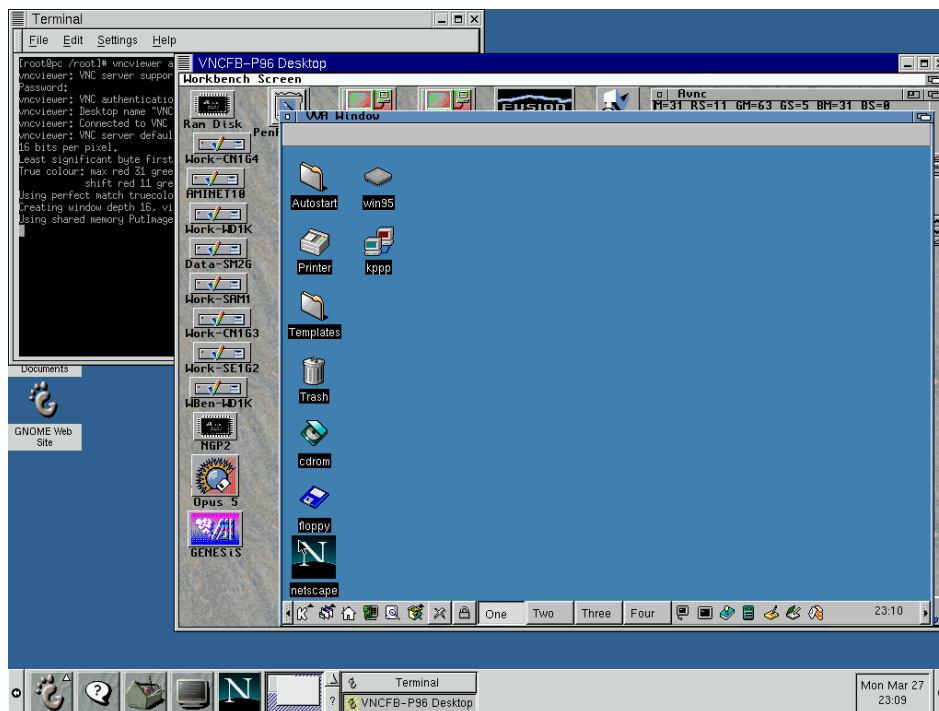
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
tedy@vpn-test:~$ uname -a
Linux vpn-test 2.6.24-16-generic #1 SMP Thu Apr 10 13:23:42 UTC 2008 i686 GNU/Linux
tedy@vpn-test:~$
```




vnc – grafisks attālinātās administrēšanas rīks, kas rada attālinātā datora darba virsmu. Arī šeit šifrēšanai labāk izmantot šifrētu tuneli (*SSH*).



Izmantotā literatūra:

- Christopher Negus, *Linux Bible*, Wiley Publishing, 802 lpp, 2005
- Trevor Kay, *Linux+ Certification Bible*, Hungry Minds, 669 lpp, 2002
- Klimas Family, *Linux Newbie Administrator Guide*, 134 lpp, 1998
- Bill McCarty, *Learning Red Hat Linux*, 3rd Edition, O'Reilly, 336 lpp, 2003