

"Vispārējās izglītības pedagogu tālākizglītība" (vienošanās Nr.2010/0062/1DP/1.2.1.2.3/09/IPIA/VIAA/003)

### 1.modulis/DM/IM/1-1lidz5-Linux\_komandrindas\_serveris.pdf

"Novitātes mācību priekšmetu tehniskā atbalsta sistēmā"

# Linux komandrindas servera ieviešana skolā

## **Kaspars Krampis**

Rīga, 2011



Materiāls licencēts ar Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported licenci http://creativecommons.org/licenses/by-nc-sa/3.0/

#### Saturs

Priekšvārds	3
Ubuntu Linux uzstādīšana	4
Ievads komandrindas saskarnē	13
Failu apskates rīki	18
Sistēmas rīki	19
Linux programmkanāli	22
Rakstīšana failā	23
Failu rediģēšana	24
Komandrindas rīku pamācības	
BASH čaulas iespējas	25
Failu lejupielādēšana no tīmekļa	
Administratīvu funkciju veikšana	
Programmatūras uzstādīšana un atjaunināšana	
Manipulācija ar pakotnēm	
Serveru programmatūras uzstādīšana	34
Mājaslapu, DNS un Samba serveru uzstādīšana	
Linux pārvaldības rīks Webmin	

# Attēli

Attēls 1: Valodas izvēle	
Attēls 2: Uzstādīt serveri	5
Attēls 3: Operētājsistēmas valodas izvēle	6
Attēls 4: Tīkla iestatījumu automātiska konfigurēšana	7
Attēls 5: Servera nosaukuma uzstādīšana	7
Attēls 6: Laika joslas izvēle	7
Attēls 7: Cietā diska sadaļu veidošana	8
Attēls 8: Cietā diska sadaļu apstiprināšana	
Attēls 9: Linux operētājsistēmai atvēlētas diskas vietas iestatīšana	9
Attēls 10: Diska sadaļu formatēšanas apstiprināšana	9
Attēls 11: Mājas mapes šifrēšanas iestatīšana	10
Attēls 12: Automātisko atjauninājumu iestatīšana	10
Attēls 13: Papildu programmatūras uzstādīšanas izvēlne	
Attēls 14: Sāknētāja uzstādīšana	
Attēls 15: Uzstādīšana pabeigta	
Attēls 16: Ubuntu Linux pieslēgšanās logs	12
Attēls 17: Teksta redaktors nano	24
Attēls 18: PuTTY galvenais logs	
Attēls 19: Vai attāllinātais dators ir drošs?	
Attēls 20: SSH sesija izmantojot PuTTY	37
Attēls 21: Servera programmatūras uzstādīšanas izvēle	

## Priekšvārds

*Linux* ir datora operētājsistēma, kuras aizsākumi meklējami 1991. gadā, kad somu students Linus Torvalds radīja *Linux* kodolu – operētājsistēmas galveno komponenti. Viņš nolēma par to neprasīt samaksu, bet savu veikumu piedāvāt ikvienam bez maksas. Turklāt, viņš ne tikai atļāva izmantot *Linux* kodolu bez maksas, bet arī deva līdzi tā pirmkodu (programmēšanas valodā pierakstītu tekstu), lai citi to varētu uzlabot. Tas būtu tāpat kā pavārs kopā ar gatavu ēdienu iedotu līdzi arī recepti. Sākotnēji aprīkota tikai ar komandrindas saskarnes rīkiem *Linux* kļuva populāra operētājsistēma sistēmu administratoru vidū dēļ tās pielāgojamības un plašām pielietojuma iespējām.

Turpmākajās nodaļās lasītājs tiks iepazīstināts ar to, kā uzstādīt un uzturēt nelielu *Linux* serveri izmantojot komandrindas rīkus bez grafiskās vides uzstādīšanas. *Linux* vidē visas administratīvās darbības var veikt komandrindā. Mūsdienās lielāko daļu no tām var paveikt arī izmantojot grafiskos rīkus. Komandrindas izmantošanai ir savas priekšrocības, piemēram, tā kā grafiskās vides saskarne ir resurs ietilpīga, lai to darbinātu nepieciešams jaudīgāks dators ar lielāku operatīvās atmiņas apjomu un jaudīgāku procesoru.

Materiāla apguvei priekšzināšanas serveru administrēšanā nav nepieciešamas, tomēr tiek pieņemts, ka lasītājam ir vismaz minimāla pieredze *Linux* lietošanā, ka viņš/viņa zina kā darboties *Linux* vidē un zina kā veikt pamatdarbības šajā sistēmā.

Par pamatu tiks izmantots *Ubuntu Linux 10.04 Lucid Lynx Server* distributīvs (*Linux* paveids) izmantojot anglisku saskarni. Ja grafiskajā vidē latviešu valodas saskarnes izmantošana ir pati par sevi saprotama, tad komandrindā labāk izmantot anglisko saskarni, jo komandas tāpat jāraksta izmantojot to angliskos nosaukumus un šobrīd latviskota ir ļoti neliela daļa no komandrindas, tādēļ pagaidām labāk izmantot anglisko versiju, iespējams ar laiku arī komandrindā varēs darboties latviskajā saskarnē. *Ubuntu Linux* distributīvs izvēlēts vairāku iemeslu dēļ:

- 1. Ubuntu Linux ir šobrīd populārākais Linux distributīvs ar plašu lietotāju kopienu;
- 2. Latvijā darbojas *Ubuntu Linux* lietotāju kopiena Ubuntu-lv (<u>www.ubuntu.lv</u>);
- 3. *Ubuntu Linux 10.04 Server* ir tā saucamā LTS (*Long Term Support*) versija, kas tiks nodrošināta ar programmatūras atjauninājumiem līdz 2015. gada aprīlim;
- 4. u.c.

Kaut arī piemēros izmantota *Ubuntu Linux* operētājsistēma, aprakstītās darbības lielākoties ir savietojamas ar praktiski ikvienu citu *Linux* paveidu, jo darbības principi ir līdzīgi.

Materiālā apskatīti sekojoši uzdevumi:

1. *Ubuntu Linux* uzstādīšana un pārvaldīšana, lietotāju kontu izveidošana, papildu programmatūras uzstādīšana;

- 2. failu servera (SAMBA), tīmekļa (WEB) un DNS (Bind) serveru uzstādīšana;
- 3. administrēšanas rīka Webmin uzstādīšana

Piezīme: šis mācību materiāls ir papildinājums 2010. gadā tapušajam *Linux* servera ieviešana, L. Trukšāns, K. Krampis. Materiāls pieejams: <u>http://profizgl.lu.lv/mod/book/view.php?id=16455</u> (viesa parole: *alfa1*).

## Ubuntu Linux uzstādīšana

*Ubuntu Linux server* uzstādīšanai ir paredzēts *server* disks. Tas lejupielādējams <u>www.ubuntu.com</u> mājaslapā sadaļā *server*. Ja dators aprīkots ar 64-bit procesoru, tad servera vajadzībām ieteicams uzstādīt 64-bit *Linux* versiju. Savukārt, ja datora pamatdarbības būs lietotāju funkciju nodrošināšana izmantojot grafisko vidi, tad iespējams labāk izmantot 32-bit versiju, jo vēljoprojām ne visas lietojumprogrammas ir pielāgotas darbam 64-bit vidē. Serveru programmatūra gan ir pilnvērtīgi pielāgota darbam ar 64-bit sistēmām un spēj tās izmantot.

Instalācijas vednis sastāv no vairākām izvēlnēm, kur jāizvēlas sistēmas valoda, pielietojums, laika josla, jāveido diska sadaļas un jāveic citas darbības. Turpmākajās lappusēs tiks aprakstīta *server* versijas uzstādīšana.

Ievietojot *Ubuntu Linux server* disku diskdzinī un iestartējot datoru no tā, tiek parādīta sekojoša valodas izvēlne:

	Lan	guage	
Amharic	Gaeilge	Nederlands	Wolof
Arabic	Galego	Norsk bokmål	中文(简体)
Asturianu	Gujarati	Norsk nynorsk	中文(繁體)
Беларуская	עברית	Punjabi(Gurmukhi)	
Български	Hindi	Polski	
Bengali	Hrvatski	Português do Brasil	
Bosanski	Magyar	Português	
Català	Bahasa Indonesia	Română	
Čeština	Italiano	Русский	
Cymraeg	日本語	Sámegillii	
Dansk	ქართული	Slovenčina	
Deutsch	Қазақ	Slovenščina	
Dzongkha	Khmer	Shqip	
Ελληνικά	한국어	Српски	
English	Kurdî	Svenska	
Esperanto	Lietuviškai	Tamil	
Español	Latviski	Thai	
Eesti	Македонски	Tagalog	
Euskara	Malayalam	Türkçe	
Suomi	Marathi	Українська	
Français	Nepali	Tiếng Việt	
1 Help F2 Language F3	3 Keymap  F4 Modes	F5 Accessibility F6 O	ther Options

#### Attēls 1: Valodas izvēle

Kā jau iepriekš minēts, tiks izmantota angliskā saskarne. Valodu var mainīt izmantojot kursorus uz tastatūras un apstiprināt ar *Enter* taustiņu. Piezīme: komandrindas saskarnē datora pele netiek izmantota, tādēļ to droši var palikt malā, lai netraucē. Pierodot vairāk strādāt ar

tastatūru, darbības var paveikt arvien ātrāk. Atcerieties – tastatūrai ir 101 taustiņš, bet datorpelei tikai 3.

Nākamajā solī tiek piedāvāts izvēlēties starp paredzētajiem pielietojumiem:



Noklusētais ir uzstādīt *Ubuntu* serveri (*Install Ubuntu Server*). Ubuntu aprīkots arī ar mākoņskaitļošanas tehnoloģijām un *Ubuntu* izstrādātāji ir ievērojami atvieglojuši datora sagatavošanu darbam "mākonī" izveidojot *Ubuntu Enterprise Cloud* risinājumu un daļēji automatizējuši tā uzstādīšanu. Vēl ir iespējams pārbaudīt vai kompaktdisks ierakstījies korekti un viss tā saturs ir nolasāms (*Check disc for defects*). *Server* instalācijas disku var izmantot arī, lai pārbaudītu datora operatīvo atmiņu vai pat, lai "ārstētu" uz datora jau uzstādītu sistēmu, kas nedarbojas.

Nākamajā solī jānorāda valoda (vēlreiz):



Attēls 3: Operētājsistēmas valodas izvēle

Nākamajos soļos jāizvēlas atrašanās vieta. Vēlams norādīt Latviju (*other -> Europe -> Latvia*). Izvēloties sev tuvāko atrašanās vietu tiek uzstādīta laika josla, lai datora pulkstenis rādītu pareizu laiku (serveru programmatūrai tas ir jo īpaši svarīgi). Vadoties pēc atrašanās vietas tiek izvēlēts arī tuvākais programmatūras spoguļserveris – dators, kas satur *Ubuntu Linux* pieejamo programmatūru, ko varēs uzstādīt izmantojot komandrindas rīkus, neapmeklējot izstrādātāju mājaslapas un lejupielādējot no tām. Plašāk par programmatūras uzstādīšanu skatīt nodaļu failu rediģēšana.

Tastatūras izvēlnes solī ieteicams izvēlēties izkārtojumu ar roku un atteikties no automātiskas atpazīšanas (*Detect keyboard layout? <No>*). Parasti datoriem ir standarta amerikāņu 101 taustiņu tastatūra (USA -> USA izvēlnes).

Nākamajos soļos no kompaktdiska tiek ielādētas papildu komponentes, kas nepieciešamas uzstādīšanai.

Ja datortīkls, kam pieslēgts dators ir aprīkots ar DHCP serveri, tad *Ubuntu* to noskaidros un būs spējīgs darboties tīklā bez papildu iestatījumu ievadīšanas. Ja tāda nav, tad IP adreses parametrus jāievada ar roku:



Attēls 4: Tīkla iestatījumu automātiska konfigurēšana

Nākamais solis ir datora vārda izvēle:



Attēls 5: Servera nosaukuma uzstādīšana

Nākamajā solī tiek prasīts apstiprināt vai automātiski noteiktā laika josla ir īstā. Ja iepriekšējos soļos izvēlēta Latvija, tad tiek piedāvāta pareiza *Europe/Riga* laika josla:



#### Attēls 6: Laika joslas izvēle

Nākamajā solī jānorāda kā tiks izmantots cietais disks, kā veidot diska sadaļas (partīcijas). *Server* versijai pēc noklusējuma tiek piedāvāts izmantot LVM loģisko sējumu pārvaldnieku un veidot t.s. dinamisko disku, ko pēc tam būs iespējams dinamiski mainīt palielinot vai samazinot atsevišķus sējumus jau uzstādītai un strādājošai sistēmai. Ja datorā ir vairāki cietie diski, tad īpaši jāpievērš uzmanība, kurā tiks uzstādīta sistēma:



Attēls 7: Cietā diska sadaļu veidošana

NB! Izmantojot visu disku (*use entire disk*) tā saturs tiks dzēsts pilnībā un neatgriezeniski!



Attēls 8: Cietā diska sadaļu apstiprināšana

Izmantojot LVM sējumus, instalācijas laikā iespējams norādīt cik lielu daļu no cietā diska izmantot. Var norādīt visu disku vai kādu daļu no tā. Ja norāda daļu, tad ar laiku iespējams dinamiski palielināt kādu no sējumiem, ja tajā trūkst vieta. Iesākumā var norādīt visu disku (atstāt maksimālo vērtību):



Attēls 9: Linux operētājsistēmai atvēlētas diskas vietas iestatīšana

Diska sadaļu veidošanas pēdējā solī vēlreiz tiek pārprasīts vai tiešām vēlaties turpināt un uzstādīt *Ubuntu Linux*, nodzēšot visu līdzšinējo diska saturu:



Attēls 10: Diska sadaļu formatēšanas apstiprināšana

Pēc šī soļa beigšanas tiek sākta t.s. bāzes sistēmas jeb minimāli nepieciešamās *Linux* programmatūras uzstādīšana. Tas parasti neaizņem vairāk kā 5-10 minūtes. Kad tas ir paveikts, tiek veidots sistēmas lietotājs, kas paredzēts administratoram. Jāievada lietotāja pilnais vārds, lietotājvārds (kas tiks izmantots, lai pieslēgtos sistēmai) un parole. Paroli vēlams izvēlēties vismaz 8 simbolus garu ar vismaz vienu lielo burtu un ciparu.

Nākamajā solī tiek piedāvāts šifrēt lietotāja mājas mapi. Datu šifrēšana ir noderīga un var pasargāt no jūsu privāto datu nokļūšanas nevēlamās rokās. Ja tiek izvēlēts šifrēt datus, jāņem vērā, ka mājas mape tiek šifrēta ar iepriekš izvēlēto paroli. Tas nozīmē, ka ikviens, kas to zinās, varēs piekļūt datiem. Tas nozīmē arī to, ja parole tiks aizmirsta, tad datiem, kas atrodas mājas mapē visticamākais vairs nekad nevarēs piekļūt. Varbūtība, ka šifrēto mapi varēs "atlauzt" ir ļoti neliela, turklāt tas var prasīt ievērojamus skaitļošanas un finansiālus resursus (piesaistot speciālistus) bez garantēta rezultāta:



Attēls 11: Mājas mapes šifrēšanas iestatīšana

Iesākumā ieteicams neuzstādīt mājas mapes šifrēšanu. Tas vairāk ir aktuāls gadījumos, kad tiek glabāti īpaši slepeni dati vai arī pastāv bažas par datora fizisko drošību – t.i., ka to var fiziski aiznest. Ja nevēlamai personai ir pieeja datoram un dati nav šifrēti, tad failiem var piekļūt arī bez lietotāja paroles zināšanas, piemēram, pieslēdzot cieto disku pie sava datora kā ārējo iekārtu.

Nākamajā solī iespējams izvēlēties vai sistēmas drošības atjauninājumus uzstādīt automātiski vai nē:



Attēls 12: Automātisko atjauninājumu iestatīšana

Drošības atjauninājumu uzstādīšana ir ļoti svarīga un tāpēc ieteicams atzīmēt, lai tie tiktu automātiski uzstādīti (*Install security updates automatically*). Pieredzējuši administratori mēdz būt skeptiski pret šādiem automātiskiem atjauninājumiem un izvēlas sekot līdzi

programmatūras versijām paši. Tā ir brīva izvēle, galvenais lai drošības atjauninājumi tiktu regulāri uzstādīti. Plašāk par atjauninājumu uzstādīšanu skatīt nodaļu Programmatūras uzstādīšana un atjaunināšana.

*Ubuntu server* versija aprīkota ar rīku tasksel, kas piedāvā uzstādīt programmatūru pēc paredzētā uzdevuma, piemēram, automātiski uzstādīt tīmekļa serveri un visu tam nepieciešamo papildu programmatūru (datubāžu serveri un tīmekļa programmēšanas valodas atbalstu). Uzstādītās komponentes tiek konfigurētas un ir izmantojamas uzreiz pēc uzstādīšanas:



Attēls 13: Papildu programmatūras uzstādīšanas izvēlne

Programmatūras uzstādīšana plašāk apskatīta nodaļās programmatūras uzstādīšana un atjaunināšana un manipulācija ar pakotnēm, tāpēc šajā solī papildu programmatūra pēc veicamā uzdevuma netiks veikta.

Nākamajā solī pēc programmatūras uzstādīšanas jāizvēlas sāknētāja (*boot loader*) uzstādīšana. Sāknētājs ir datora cietā diska pirmajos baitos ierakstīta īpaša programma, kas nepieciešama, lai varētu iestartēt (sāknēt) datorā uzstādīto (vai uzstādītās) operētājsistēmu (vai operētājsistēmas). *Linux* operētājsistēma izmanto GRUB sāknētāju. Ja *Ubuntu Linux* ir vienīgā operētājsistēma, tad GRUB droši var uzstādīt t.s. *master boot record* (MBR) ierakstā. Tas nozīmē, ka ieslēdzot datoru, sākumā tiks ielādēts GRUB, kas iestartēs operētājsistēmu. Ja datorā ir vairākas operētājsistēmas, tad arī GRUB var droši uzstādīt MBR ierakstā. Šajā gadījumā ieslēdzot datoru, tiks parādīta GRUB izvēlne, kas piedāvās izvēlēties kuru no pieejamajām operētājsistēmām iestartēt:



Attēls 14: Sāknētāja uzstādīšana

Ar to *Ubuntu Linux* instalēšana ir pabeigta. Pēdējā solī tiek atgādināts pēc pārstartēšanas izņemt instalācijas disku no diskdziņa, lai instalācijas process netiktu ielādēts no jauna:



Attēls 15: Uzstādīšana pabeigta

Pēc pārstartēšanas tiek ielādēta jaunā *Ubuntu Linux 10.04 Server* operētājsistēma. Lai pieteiktos jāievada instalācijas laikā izvēlētais lietotājvārds un parole:



Attēls 16: Ubuntu Linux pieslēgšanās logs

NB! *Linux* vidē ievadot lietotāja paroli tās garums netiek rādīts. T.i. Netiek rādītas par ierastās "zvaigznītes" burtu vietā. Tas tiek veikts, lai paaugstinātu sistēmas drošības līmeni. Pat ja kāds nepamanīts vēros servera ekrānu pieslēgšanās brīdī, nevarēs noskaidrot pat paroles garumu.

## levads komandrindas saskarnē

```
Pieslēdzoties, tiek parādīta informācija par serveri un komandrinda ar mirgojošu kursoru.
Sistēma ir gatava apstrādāt lietotāja ievadītās komandas:
     Linux serveris 2.6.32-24-generic #39-Ubuntu SMP Wed Jul 28 06:07:29 UTC
2010 i686 GNU/Linux
     Ubuntu 10.04.1 LTS
    Welcome to Ubuntu!
     * Documentation: https://help.ubuntu.com/
     System information as of Wed Jan 26 11:51:30 EET 2011
     System load: 0.0
                                                             67
                                       Processes:
                   12.5% of 5.37GB
     Usage of /:
                                      Users logged in:
                                                             1
                                       IP address for lo:
    Memory usage: 47%
                                                             127.0.0.1
     Swap usage:
                   0%
                                       IP address for eth0: 10.0.2.15
```

Graph this data and manage this system at <a href="https://landscape.canonical.com/">https://landscape.canonical.com/</a>

```
user@serveris:~$
```

NB! Šeit un turpmāk komandrindas saskarnes ievads un izvads tiks rakstīts šādā vienplatuma fontā.

Datorus var pārvaldīt izmantojot divas t.s. lietotāju saskarnes: grafisko lietotāju saskarni (*graphical user interface* jeb GUI) un komandrindas lietotāju saskarni (*command line interface* jeb CLI). Grafiskajā saskarnē komandas parasti tiek dotas izmantojot peles kursoru (piemēram, MS *Windows*), savukārt komandrindas saskarnē ievadot komandas izmantojot klaviatūru (piemēram, MS-DOS, *Linux*). Arī mūsdienu *Windows* operētājsistēmas versijās ir iekļauta komandrindas saskarne (*command prompt*), tomēr ar to nevar pilnvērtīgi administrēt sistēmu un tai ir ļoti ierobežot funkcionalitāte. Turpretim, *Linux* komandrindas saskarne ir jaudīgs rīks ar kuru var pārvaldīt sistēmu un nereti pat izdarīt vairāk nekā grafiskajā vidē. Sākotnēji *Linux* sistēmām CLI bija vienīgā pieejamā saskarne un GUI tika izveidota vēlāk (grafiskā saskarne arī ir programma), tādējādi ne visiem komandrindas rīkiem ir izveidoti attiecīgie grafiskie rīki, kas nodrošinātu to pašu funkcionalitāti. Mūsdienās, izmantojot *Linux*, lietotājs var paralēli darboties gan GUI, gan CLI un izvēlēties, ko labāk veikt grafiskajā vidē, bet ko komandrindā. Šī materiāla ietvaros tiks apskatīta servera administrēšana komandrindā.

Katru lietotāja ievadīto komandu apstrādā t.s. komandrindas interpretators jeb čaula (*shell*). Tā ir programma, kas darbojas kā starpnieks starp lietotāju un operētājsistēmu. *Linux* 

pieejami dažādi komandrindas interpretatori, bet populārākais no tiem ir BASH (*Bourne Again Shell*). Arī *Ubuntu Linux* kā noklusēto čaulu izmanto BASH.

Komandrinda parasti izskatās šādi:

user@serveris:~\$

kur user ir lietotājvārds, serveris ir datora nosaukums, tālāk seko adrese direktoriju kokā, kurā šobrīd "atrodas" lietotājs (~ zīme norāda uz mājas mapi). \$ zīme norāda uz lietotāja statusu - \$, ja lietotājs darbojas ar ierobežotām tiesībām (tā jābūt ikdienas darbā), #, ja lietotājs darbojas kā administrators (*root*).

Lai izpildītu komandu, tā jāievada un jānospiež *Enter* taustiņš. Parasti komandas tiek rakstītas šādā formā:

command -option file

komanda -papildu parametri (neobligāti) fails

Ir komandas, kuras neprasa papildu parametrus, pietiek ievadīt tikai komandas nosaukumu, piemēram, komanda ls:

user@serveris:~\$ ls

user@serveris:~\$

Nekas netika izdrukāts. Kāpēc tā? Tāpēc, ka aktīvajā direktorijā nav failu, precīzāk, nav neapslēpto (*hidden*) failu. *Linux* filozofija nosaka, ja komanda izpildās veiksmīgi, tā papildu ziņojumus neveic un neizvada, piemēram, "komanda ls izpildījās, šajā direktorijā nav failu". Klusēšana zelts — tā varētu dēvēt šo pieeju. Komanda ls izpildījās veiksmīgi, redzamu failu nav, tika atgriezta komandrinda. Komandu ls var izpildīt ar papildu parametriem, piemēram, parametru - a, kas nosaka, ka tiek izdrukāti arī slēptie faili:

```
user@serveris:~$ ls -a
. .. .bash_history .bash_logout .bashrc .cache .profile
user@serveris:~$
```

*Linux* vidē slēptajiem failem nav īpaša atribūta, kas norādītu to statusu, tā vietā tiek izmantots punkta simbols. Ja faila (vai mapes) nosaukums sākas ar punktu, tad tas ir slēpts. Atsevišķi redzamie punkts (.) un divpunkte (..) ir direktoriju koka apzīmējumi. Punkts vienmēr norāda uz aktīvo direktoriju, savukārt divpunkte uz direktoriju, kas atrodas vienu līmeni augstāk. Dotajā mapē ir vairāki slēptie faili, piemēram, .bash\* faili ir BASH interpretatora konfigurācijas un palīg faili.

Komandām parasti ir vairāki papildu parametri un tos var kombinēt kopā. Piemēram, komandai 1s ir parametrs -1 (no *long*), kas drukā papildu informāciju par direktorijā esošajiem failiem, ja to saliek kopā ar iepriekš apskatīto - **a**, tad iegūst šādu rezultātu:

```
user@serveris:~$ ls -la
total 28
drwxr-xr-x 3 user user 4096 2011-01-26 13:02 .
```

```
drwxr-xr-x 3 root root 4096 2010-12-26 12:10 ..
-rw----- 1 user user 620 2011-01-26 11:51 .bash_history
-rw-r--r-- 1 user user 220 2010-12-26 12:10 .bash_logout
-rw-r--r-- 1 user user 3103 2010-12-26 12:10 .bashrc
drwx----- 2 user user 4096 2010-12-26 12:13 .cache
-rw-r--r-- 1 user user 675 2010-12-26 12:10 .profile
user@serveris:~$
```

Redzama tāda informācija kā faila izmērs baitos, tā īpašnieks, kad tas pēdējo reizi labots,

u.c.

Pirms tiek apskatītas citas *Linux* komandas, jāiepazīstas ar direktoriju struktūru. Ja *Windows* operētājsistēmā parasti sastopami vairāki direktoriju koki, piemēram, C:, D:, kur katrs atbilst fiziskam diskam vai tā sadaļai (partīcijai), tad *Linux* vidē visa direktoriju struktūra glabājas vienotā kokā. Augstāko punktu apzīmē ar / zīmi un to sauc par saknes direktoriju (*root directory*). Jebkura cita direktorija, kas atrodas sistēmā ir saknes apakšdirektorija. Piemēram, komandai 1s kā faila parametru var rakstīt direktoriju (ja tā netiek rakstīta, tad tiek drukāta aktīvā):

```
user@serveris:~$ ls /
bin
             initrd.img
                         media
       dev
                                proc
                                       selinux
                                                tmp
                                                     vmlinuz
boot
       etc
             lib
                         mnt
                                 root
                                       srv
                                                usr
cdrom
      home
            lost+found opt
                                 sbin
                                       sys
                                                var
```

Saknes direktorija parasti satur ap 20 apakšdirektorijām, svarīgākās no tām:

- bin, sbin pamatkomandas, kas paredzētas lietotājam (bin) un administratoram (sbin)
- boot sistēmas sāknēšanas faili, kodols
- dev fiziskās (vai speciālās) ierīces
- etc sistēmas un servisu konfigurācijas faili
- home lietotāju mājas mapes
- root administratora (root) mājas mape
- usr lietotāju programmatūra, šīs mapes apakšdirektoriju struktūra atgādina saknes direktoriju, to mēdz dēvēt par sekundāro hierarhiju
- var mainīgi faili (variable), piemēram, žurnālfaili, pagaidu (temp) faili, u.c.

Linux direktoriju struktūra plašāk aprakstīta Filesystem Hierarchy Standard<sup>1</sup>

Lai pārvietotos pa direktorijām, tiek izmantota komanda cd (*change directory*). Kā parametru var rakstīt vai nu absolūtu adresi vai relatīvu pret šā brīža direktoriju. Absolūta adrese vienmēr sākas ar / zīmi un norāda pilno ceļu līdz mapei, piemēram:

```
user@serveris:~$ cd /
user@serveris:/$
```

<sup>1</sup> Filesystem Hierarchy Standard - http://www.pathname.com/fhs/

Ievērojiet, ~ zīme nomainījās pret /, tas nozīmē, ka šobrīd aktīvā ir saknes direktorija.

Ievadot relatīvas adreses parametru, komanda cd, "meklēs" tādu adresi aktīvajā direktorijā:

user@serveris:/\$ cd home

user@serveris:/home\$

Komanda cd bez parametriem aizvedīs uz mājām jeb mājas mapi, kas parasti ir /home/lietotājvārds (izņemot root lietotāju, kam ir /root)

user@serveris:/home\$ cd

user@serveris:~\$

Lai noskaidrotu absolūto ceļu līdz aktīvajai direktorijai paredzēta komanda pwd:

user@serveris:~\$ pwd

/home/user

Lai pārvietotos pa direktorijām var izmantot arī . . norādi uz mapi, kas atrodas vienu līmeni augstāk par aktīvo:

user@serveris:~\$ cd .. user@serveris:/home\$ pwd /home

Failu kopēšanai un dzēšanai paredzētas komandas Cp (*copy*) un rm (*remove*). Piemēram nokopēt failu ar nosaukumu fails1 uz failu fails2:

```
user@serveris:~$ ls
fails1
```

user@serveris:~\$ cp fails1 fails2

```
user@serveris:~$ ls
fails1 fails2
```

Komanda Cp ievēro jau iepriekš aprakstīto "klusēšana zelts" pieeju. Kopēšanas darbība notika veiksmīgi, papildu paziņojumi netiek veikti.

Komanda rm dzēš failu:

user@serveris:~\$ rm fails1

```
user@serveris:~$ ls
fails2
```

Komanda rm nepārprasa vai tiešām vēlamies dzēst failu — jābūt uzmanīgiem! Tomēr šai komandai ir papildu parametrs, kas nodrošina pārvaicāšanu:

user@serveris:~\$ rm -i fails1

rm: remove regular file `fails1'? y

Šādi var nodrošināties pret nejaušu failu izdzēšanu. Ja vēlaties, lai failu dzēšana vienmēr tiktu pārvaicāta, iespējams izveidot t.s. aizstājvārdu (alias), ko nodrošina BASH vide. Ievadot sekojošu rindiņu .bashrc failā mājas mapē:

```
alias rm='rm -i'
```

tiek noteikts, ka katru reizi, kad komandrindā tiks ievadīta komanda rm, tā tiks aizstāta ar komandu rm -i. Teksta failu labošanai ieteicams izmantot teksta redaktoru nano (par to plašāk lasīt nodaļu failu rediģēšana). NB! Lai izveidotais aizstājvārds darbotos ir no jauna jāpiesakās sistēmā, jo .bashrc fails tiek ielasīts vienu reizi, pieslēgšanās brīdī. Komanda exit (vai ctrl+d tastatūras kombinācija) aizver šā brīža sesiju un pēc tam var pieslēgties no jauna.

Jauna mape tiek veidota ar komandu mkdir: user@serveris:~\$ mkdir manamape user@serveris:~\$ ls fails1 fails2 manamape Failu pārvietošana (un arī pārsaukšana) tiek veikta ar komandu mv: user@serveris:~\$ mv fails1 manamape user@serveris:~\$ ls fails2 manamape user@serveris:~\$ ls manamape fails1

Fails ar nosaukumu fails1 tika pārcelts uz mapi manamape. Komanda 1s bez papildu parametriem izdrukā aktīvās direktorijas saturu, bet izmantojot faila parametru (šajā gadījumā relatīvu adresi), tiek izdrukāts mapes manamape saturs. Varētu arī rīkoties šādi:

```
user@serveris:~$ cd manamape
user@serveris:~/manamape$ ls
fails1
```

Faila pārsaukšana pēc būtības ir "faila pārvietošana uz citu nosaukumu". Tāpēc tiek izmantota tā pati komanda mv:

```
user@serveris:~/manamape$ mv fails1 1fails
user@serveris:~/manamape$ ls
1fails
```

Lai darbības tiktu pārjautātas, komandām mv un cp arī var izveidot aizstājvārdus .bashrc failā.

Direktoriju dzēšanai tiek izmantota jau iepriekš apskatītā komanda rm vai komanda rmdir, tomēr ir dažas nianses, kas jāievēro dzēšot direktorijas. Var dzēst tikai tukšas direktorijas, tāpēc ir vairākas iespējas kā rīkoties. Izdzēst visus direktorijā esošos failus un tad izmantot rmdir komandu vai izmantot -r parametru (r no rekursīvs) lai dzēstu direktoriju un visus failus (vai apakšdirektorijas), kas tajā atrodas:

```
user@serveris:~$ ls manamape
1fails
user@serveris:~$ ls tuksamape
```

```
user@serveris:~$ rmdir manamape
rmdir: failed to remove `manamape': Directory not empty
user@serveris:~$ rmdir tuksamape
```

```
user@serveris:~$ rm -r manamape
rm: descend into directory `manamape'? y
rm: remove regular empty file `manamape/1fails'? y
rm: remove directory `manamape'? y
```

Ikdienā ieteicams izmantot komandu rm. rmdir parasti tiek izmantota reti vai netiek vispār. Ja direktorijā ir daudz failu vai apakšdirektoriju, tad pārvaicāšanas režīmā var nākties atbildēt y daudzas reizes. Ja esat droši par direktorijas saturu un pilnu tā dzēšanu, tad šādos gadījumos var pielietot papildu parametru -f (no *force* jeb uzspiest), kas ļauj ignorēt pārvaicāšanas parametru:

user@serveris:~\$ rm -rf manamape

## Failu apskates rīki

Vairums konfigurācijas un žurnāl failu *Linux* sistēmā tiek glabāti vienkārša teksta (*plaintext*) formā, t.i. cilvēklasāmi. Vienkāršā tekstā pierakstīti faili ir ērti lasāmi, apstrādājami un tālākizmantojami, tāpēc *Linux* sistēmā tie plaši pielietoti un pieejams plašs klāsts rīku, kas paredzēti darbam ar šādiem failiem. Piemēram, failu apskates rīki – cat, less, head un tail.

Rīks Cat komandrindā izdrukā visu faila saturu, izmantošana: Cat failanosaukums (vai cat /adrese/failanosaukums, ja izmanto absolūto adresi). Ja fails ir liels, tad ar Cat izdrukājot visu tā saturu, var gadīties, ka izlasīt var tikai tā beigu daļu, jo komandrindas izvada atmiņā visam failam vietas nepietiek. Tādos gadījumos var izmantot komandu less, kas atver failu interaktīvā režīmā, ar iespēju to "šķirstīt" ar *PageUp* un *PageDown* tastatūras taustiņiem. Lai aizvērtu failu, jānospiež taustiņš q (no vārda *quit*, beigt).

Lai izdrukātu faila pirmās vai pēdējās 10 rindas, paredzētas komandas head un tail. Ja jādrukā mazāk vai vairāk par noklusētajām 10 rindām, izmanto parametru -n x, kur x ir rindu skaits, piemēram:

```
user@serveris:/etc$ head -n 4 /etc/hosts
127.0.0.1 localhost
127.0.1.1 serveris
```

# The following lines are desirable for IPv6 capable hosts

Komanda tail tiek izmantota arī lai "sekotu" faila beigām. Piemēram, žurnālfailam, kas nemitīgi tiek papildināts. Tam paredzēts parametrs - f (no angļu val. *follow*, sekot):

tail -f mansfails Lai apturētu "sekošanu" jānospiež Ctrl+c.

### Sistēmas rīki

Lai uzzinātu informāciju par sistēmas stāvokli, brīvo diska vietu, atmiņas aizpildījumu un citiem parametriem, *Linux* paredzēti vairāki rīki, daži no tiem ir df, free, w, top, du, ipconfig, u.c.

Komanda df paredzēta, lai noskaidrotu disku (cieto, kompakt, ārējo, tīkla) aizpildījumu. Pēc noklusējuma izmēri tiek drukāti 1 kilobaita blokos un tie ne visiem ir ērti saprotami, tādēļ ierasts, ka df komandu izpilda ar parametru -h (*human readable* jeb cilvēkam lasāmā formātā):

user@serveris:~\$ df	- h				
Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda1	28G	18G	8,9G	67%	/
none	491M	252K	491M	1%	/dev
none	497M	1,6M	495M	1%	/dev/shm
none	497M	116K	497M	1%	/var/run
none	497M	Θ	497M	0%	/var/lock

Cietie diski *Linux* sistēmā tiek apzīmēti kā /dev/sdX iekārtas, kur X ir alfabēta burti sākot ar a (pirmajam diskam), b (otrajam diskam), utt. Katram diskam var būt sadaļas (partīcijas), tās tiek numurētas: **1**, **2**, . . . , **n**, piemēram, sda1 ir pirmā diska pirmā partīcija. Dotajā piemērā datoram ir 28GB (gigabaitu) liela partīcija ar aizņemtiem 18GB un brīviem 8.9GB, aizpildījums 67% un tā piemontēta (pieslēgta) kā / (saknes direktorija). Pārējie ieraksti norāda uz virtuālām failsistēmām, kas nodrošina sistēmas funkcijas, šobrīd tās detalizētāk netiks apskatītas.

Lai noskaidrotu kādas direktorijas izmēru izmanto komandu du (*disk usage*, diska izlietojums), līdzīgi kā komandai df, iesaku izmantot parametru -h, lai izmēri tiktu attēloti ērtāk saprotamā formā:

```
user@serveris:~$ du -h /home/
4.0K /home/user/.cache
36K /home/user
40K /home/
```

Tiek izdrukāta informācija par katru apakšdirektoriju, kas atrodas norādītajā direktorijā un kopīgais izlietojums. Ja informācija par apakšdirektorijām nav nepieciešama, tad var izmantot parametru - S (*summarize*, summēt), tad tiks izdrukāts tikai kopīgais apjoms, t.i. tikai pēdējā rinda no iepriekš apskatītā piemēra:

```
user@serveris:~$ du -hs /home/
40K /home/
```

Komanda free paredzēta, lai noskaidrotu operatīvās vai virtuālās atmiņas apjomu vai aizpildījumu. Līdzīgi kā ar komandu df, pēc noklusējuma izmēri tiek drukāti citās mērvienībās nekā ierasts. Lai arī šajā gadījumā skaitļi ir jau nedaudz saprotami, ieteicams izmantot -m parametru, lai izmēri tiktu drukāti megabaitos:

	user@serveris:-	-\$ free -m						
		total		used		free	shared	buffers
cach	ed							
	Mem:	118		54		63	0	8
27								
	-/+ buffers/cad	che:	18		99			
	Swap:	307	0	:	307			

Serverim pieejami 118MB operatīvās atmiņas (RAM), no tiem 18MB ir aizņemti, bet 99 brīvi (jāskatās -/+ buffers/cache rindiņa, jo tajā attēlotie patiesie rādītāji). Ir pieejama arī 307MB virtuālā atmiņa (swap) un tā šobrīd nav aizņemta nemaz.

Komanda w (no *who*, kurš/kuri) izdrukā šobrīd sistēmā pieteikušos lietotāju sarakstu, ko tie šobrīd dara (kāda komanda tiek izpildīta), sistēmas noslodzi, laiku u.c. parametrus:

user@serveris:~\$ w

17:48:25 up 1 min, 1 user, load average: 0.57, 0.36, 0.14 USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT user pts/0 127.0.0.1 17:47 0.00s 0.40s 0.01s w

Komanda top nodrošina *Linux* procesu novērošanu un pārvaldīšanu. Tas ir interaktīvs rīks, kas attēlo šobrīd aktīvākos procesus, to procesora, operatīvās atmiņas un citu resursu izlietojumu:

load average: 0.00, 0.00, 0.15 top - 11:19:38 up 1:04, 1 user, 1 running, 64 sleeping, Tasks: 65 total, 0 stopped, 0 zombie Cpu(s): 0.1%us, 0.4%sy, 0.0%ni, 99.0%id, 0.2%wa, 0.0%hi, 0.2%si, 0.0%st 121348k total, 56444k used, 64904k free, 8204k buffers Mem: Swap: 315384k total, 0k used, 315384k free, 30516k cached

	PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	
	COMMAN	ND	813	user	20	0	2408	1016	796	R	0.1	0.8
	0:00.0	02 top			1	root	20	0	2644	1416 1	L032	S
	0.0	1.2	0:00.9	96 init	:		2	root	20	0	0	0
	0	S	0.0	0.0	0:00.0	02 kthr	read					
	6	root	20	0	0	0	0	S	0.0	0.0	0:04.2	26
events	5											
	7	root	20	0	0	0	0	S	0.0	0.0	0:00.0	00
cpuset	Ľ											
	8	root	20	0	0	0	0	S	0.0	0.0	0:00.0	00
khelpe	er											

	9	root	20	0	0	0	0	S	0.0	0.0	0:00.00
netns											

Dotais dators ir bijis ieslēgts 1h un 4 minūtes, noslodze pēdējo 5, 15, 30 minūšu laikā ir bijusi 0.00, 0.00 un 0.15. Kopā atvērti 65 procesi, no kuriem šobrīd darbojas tikai 1, pārējie ir t.s. miega (*sleep*) režīmā. Procesora noslodze ir tuvu nullei un brīvie procesora resursi if 99%. Pieejama arī operatīvās atmiņas informācija un interaktīvs procesu saraksts, kas pēc noklusējuma tiek kārtots pēc procesora (%CPU) noslodzes apgrieztā kārtībā (aktīvākie procesi augšā). Šobrīd aktīvākais process ir top (redzamais top rīks). Tas darbojas ar procesa identifikatoru (PID) 813, to izpilda lietotājs user, tam ir neitrāla (0) prioritāte jeb NI (nice) vērtība, tas nozīmē, ka process tiks pie tāda paša apjoma resursiem kā citi procesi. 0 ir noklusētā nice vērtība. Iespējamā skala ir no -20..+ 19, negatīvas vērtības nozīmē augstāku prioritāti (process is not nice, process nav "labs" pret citiem processiem), pozitīva zemāku prioritāti. VIRT, RES un SHR norāda atmiņas izlietojumu. VIRT – aizņemtās virtuālās jeb swap atmiņas apjoms, RES – aizņemtās fiziskās (RAM) atmiņas apjoms, SHR – potenciāli koplietojamais (shared) atminas apjoms, ko process varētu kopīgi izmantot vienlaicīgi ar citiem procesiem, ja sistēmā tādi tiku izpildīti. S – status, procesiem mēdz būt vairāki statusi, biežāk sastopamie ir S jeb guļošs (*sleepinq*) – process šobrīd nav aktīvs un R jeb darbojas (*runninq*). Vēl iespējami statusi ir T – process ir apstādināts, Z – process ir t.s. *zombie* stāvoklī, šis stāvoklis ir līdzīgs *"not responding"* stāvoklim grafiskās vides programmām un D – nepārtraucams gulēšanas režīms (uninterruptible sleep) var gadīties, ja process darbojas un gaida uz kādu resursu, piemēram, cieto disku, lai turpinātu darbu. Tajā brīdī procesam ir piespiedu gaidīšanas režīms. %CPU ir procesora resursu patēriņš dotajā brīdī procentos, %MEM ir atmiņas resursu patēriņš procentos no kopīgā atmiņas apjoma. TIME+ ir procesora laiks, ko process ir patērējis kopš tā izpildīšanas brīža. COMMAND ir komanda ar kādu process ir izpildīts. Pēc noklusējuma top atjaunina rādījumus ik pēc 3 sekundēm, to var mainīt ar S taustiņu. Ar taustiņiem Shift+< vai Shift+> var mainīt kolonnu pēc kuras tiek kārtoti procesi, pēc noklusējuma tā ir %CPU kolona. Rīkam top ir plaša funkcionalitāte, papildu informācija par tā izmantošanu pieejama nospiežot taustiņu h (*help*, palīdzība). Rīku aizver ar tausiņu **q** (*quit*, beigt).

Tīkla interfeisu pārvaldību nodrošina ifconfig rīks. Ar to var ieslēgt vai izslēgt tīkla karti, iestatīt tīkla parametrus. Tas parasti notiek sistēmas iestartēšanas laikā un pēc tam ar to nav jānodarbojas. Tomēr ifconfig rīks dažkārt noder, lai noskaidrotu tīkla parametrus vai to, ka tīkla kabelis ir iesprausts vai nē. Bez papildu parametriem ifconfig izdrukā informāciju par visām šobrīd aktīvajām tīkla iekārtām:

```
user@serveris:~$ ifconfig
```

```
eth0
```

Link encap:Ethernet HWaddr 08:00:27:be:4c:bf inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0 inet6 addr: fe80::a00:27ff:febe:4cbf/64 Scope:Link UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets:2266 errors:0 dropped:0 overruns:0 frame:0 TX packets:2410 errors:0 dropped:0 overruns:0 carrier:0 collisions:0 txqueuelen:1000 RX bytes:163256 (163.2 KB) TX bytes:1801033 (1.8 MB)

lo Link encap:Local Loopback inet addr:127.0.0.1 Mask:255.0.0.0 inet6 addr: ::1/128 Scope:Host UP LOOPBACK RUNNING MTU:16436 Metric:1 RX packets:0 errors:0 dropped:0 overruns:0 frame:0 TX packets:0 errors:0 dropped:0 overruns:0 carrier:0 collisions:0 txqueuelen:0 RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

Tīkla kartes *Linux* sistēmā tiek apzīmētas ar ethX, kur X ir skaitlis sākot ar 0, piemēram, eth0, eth1, utt. Pirmā tīkla karte ir eth0. Iekārta lo ir t.s. atgriezeniskais interfeiss (*loopback interface*), kas vienmēr norāda atpakaļ uz šo pašu datoru.

Iekārtai eth0 ir IP adrese (*inet addr*) 10.0.2.15, tā darbojas – UP un kopš ieslēgšanas brīža saņēmusi (RX) 163.2KB (kilobaitus) datu, bet nosūtījusi (TX) 1.8MB (megabaitus). HWaddr ir tā saucamā MAC jeb tīkla kartes fiziskā adrese.

#### Linux programmkanāli

Linux filozofija balstās uz to, ka katra komanda dara vienu lietu, bet dara to labi. Parasti šīs komandas kā ievad un izvad informāciju izmanto parastu tekstu (*plain text*), tas ļauj šīs komandas savienot virknē, kur vienas komandas izvad informācija tiek izmantota kā nākamās komandas ievad informācija. Viens no jaudīgākajiem *Linux* komandrindas rīkiem ir t.s. programmkanāli (*pipes*, no angļu val. trubas). Ar to palīdzību iespējams savienot vairākas komandas izveidojot jaunas, kas veic nepieciešamo darbību. Komandu savienošanai izmanto | simbolu. Piemēram, iepriekš apskatītās komandas ls (izdrukāt direktorijas saturu) izvads ir parasts teksts (var iedomāties, ka tas ir teksta fails, kas tajā brīdī glabājas tikai datora operatīvajā atmiņā, bet ne cietajā diskā), kas tiek izdrukāts uz ekrāna. Tā vietā, lai to drukātu uz ekrāna, izmantojot programmkanālus, šis fails tiks pāradresēts komandai grep kā ievad informācija. Komanda grep failā atrod noteiktu simbolu virkni un izdrukā visu rindu, kas satur šo virkni. Piemēram:

grep ziema mansfails

Failā mansfails tiks meklēts vārds ziema un izdrukātas visas rindas, kas to satur. Savukārt, savienojot komandu ls ar komandu grep var iegūt šādu rezultātu:

```
user@serveris:~$ ls -a | grep bash
.bash_history
```

```
.bash_logout
```

.bashrc

Komanda ls -a izdrukā visus failus (tai skaitā slēptos), kas atrodas dotajā direktorijā (šajā gadījumā lietotāja mājas direktorijā). Komanda grep bash meklē vārdu bash komandas ls -a izvadē un izdrukā atbilstošās rindas. Šī lietotāja mājas direktorijā ir 3 faili, kuru nosaukumi satur vārdu bash. Šo komandu varētu papildināt, piemēram, ar kārtošanas (sort) rīku, kas sakārto datus dilstošā secībā:

```
user@serveris:~$ ls -a | grep bash | sort -r
.bashrc
.bash_logout
.bash_history
```

Un pēc tam papildināt ar vēl vienu grep komandu, kas atlasa tikai tās rindas, kas saturs apakšsvītras \_ simbolu:

```
user@serveris:~$ ls -a | grep bash | sort -r | grep _
.bash_logout
.bash_history
```

Tā varētu turpināt bezgalīgi, pievienojot arvien papildu komandas, kas apstrādā iepriekšējās komandas izvad datus un sagatavo tos nākamajai komandai. Apskatītie piemēri ir salīdzinoši vienkārši, bet izmantojot šos pašus principus, iespējams izveidot jaudīgas komandas, kas pēc būtības ir nelielas programmas, kas veic konkrētu uzdevumu. Savā ziņā to var pat saukt par programmēšanu.

## Rakstīšana failā

Turpinot programmkanālu tēmu, tā vietā, lai iegūto rezultātu izdrukātu uz ekrāna, to var saglabāt failā. Rakstīšanai failā izmanto t.s. pāradresēšanas simbolus > un >>. Tie attiecīgi nosaka kādā veida rakstīšana tiks veikta: > nozīmē, ka failam (ja tāds eksistē) tiks rakstīts pāri (*overwrite*), dzēšot tā līdzšinējo saturu. Ja tāda faila nav, tas tiks izveidots. >> nozīmē, ka fails (ja tāds eksistē) tiks papildināts no beigām (*append*). Ja tāda faila nav, tas tiks izveidots:

```
user@serveris:~$ ls -a | grep bash | sort -r | grep _ > mansfails
Rezultāts saglabāts failā mansfails:
user@serveris:~$ cat mansfails
.bash_logout
.bash_history
Izpildot to pašu komandu papildināšanas režīmā:
user@serveris:~$ ls -a | grep bash | sort -r | grep _ >> mansfails
user@serveris:~$ cat mansfails
.bash_logout
.bash_logout
```

.bash\_history

Jau pazīstamās rindas ir ierakstītas failā 2 reizes.

## Failu rediģēšana

Lai labotu failus, *Linux* sistēmā pieejami vairāki komandrindas teksta redaktori, viens no ērtāk lietojamiem ir nano. Lai labotu esošu failu jāizpilda komanda nano ar faila nosaukumu kā parametru:

user@serveris:~\$ nano mansfails



Attēls 17: Teksta redaktors nano

Pa tekstu var pārvietoties izmantojot kursorus uz tastatūras, darbojas arī *PageUp*, *PageDown*, *Home* un *End* tastatūras taustiņi. Ekrāna apakšdaļā pieejams saraksts ar darbībām un tastatūras saīsnēm, kas tās izsauc. Piemēram, lai saglabātu (*WriteOut*), jāspiež Ctrl+o, lai aizvērtu (*Exit*) nano, Ctrl+x, lai meklētu (*Where Is*) – Ctrl+w, lai izgrieztu (*Cut Text*) rindu Ctrl+k un pēc tam to ievietotu (*UnCut Text*) – Ctrl+u. Papildu informācija par pieejamajām tastatūras saīsnēm atrodama nano pamācībā, to var atvērt nospiežot Ctrl+g. Ar nano var arī veidot jaunus failus, kā parametru ievadot jaunā faila nosaukumu, piemēram:

nano mansjaunaisfails

Darbojoties ar failiem *Linux* vidē, jāņem vērā tas, ka sistēma ir reģistrjūtīga, t.i. lielo un mazo burtu jūtīga. Fails, ar nosaukumu mansfails nav tas pats, kas fails ar nosaukumu Mansfails. Tie abi var atrasties vienā mapē:

```
user@serveris:~$ ls
mansfails Mansfails
```

Failiem ne vienmēr tiek piešķirts paplašinājums, jo *Linux* vidē parasti faila tips tiek noteikts pēc faila satura nevis tā paplašinājuma (piezīme - grafiskajā vidē tiek izmantoti arī paplašinājumi). Bet paplašinājumus, protams, var izmantot, lai labāk organizētu failus.

Piemēram, konfigurācijas failiem mēdz izmantot paplašinājumu .conf, žurnālfailiem .log, neskatoties uz to, ka gan vieni gan otri ir vienkārša teksta faili.

## Komandrindas rīku pamācības

Iepriekš apskatītajiem komandrindas rīkiem tika pieminēti tikai daži parametri, bet parasti tādu ir desmit vai pat vairāk. Piemēram, rīkam 1s ir vairāk kā 50 iespējamo parametru! Ja biežāk izmantotos parametrus var atcerēties, tad kā, lai iegaumē visus 50?! Tas nav jādara, jāzina kur meklēt informāciju par tiem. Teju vai katram komandrindas rīkam *Linux* vidē ir pieejama lietotāja pamācība jeb t.s. *man page*. Tajā pieejama pilna informācija par rīku, tā izmantošanu un pieejamajiem parametriem. Pamācības izsauc ar komandu man, kā parametru norādot rīku, kura lietošanas aprakstu ir vēlme apskatīties:

```
user@serveris:~$ man ls
NAME
ls - list directory contents
```

SYNOPSIS ls [OPTION]... [FILE]...

#### DESCRIPTION

List information about the FILEs (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor -sort.

Mandatory arguments to long options are mandatory for short options too.

Visas pamācības ir organizētas šādā veidā – augšdaļā ir komandas nosaukums un ko tā dara, tālāk seko kā tā jāizsauc. Pēc tam dots komandas plašāks apraksts. Aiz tā seko visu iespējamo parametru un to lietošanas apraksts. Lietotāja pamācība ir interaktīva, pa to var pārvietoties ar kursoriem uz tastatūras vai *PageUp/PageDown* taustiņiem. Lai meklētu jānospiež / taustiņš, jāievada meklējamā virkne un jāspiež *Enter*. Pamācību aizver ar q taustiņu. Pieejama arī pamācība par pamācības rīku:

user@serveris:~\$ man man

## BASH čaulas iespējas

BASH čaulā iekļautas komandas un tastatūras saīšņu atbalsts, kas atvieglo ikdienas darbu komandrindā. Piemēram, automātiska teksta papildināšana ar TAB taustiņu. Ievadot komandas, faila, vai pat parametra nosaukumu komandrindā, iesākto tekstu var automātiski papildināt nospiežot TAB taustiņu. Ja būs vairākas iespējas kā turpināt iesākto komandu, tad uz ekrāna tiks izdrukāts saraksts ar iespējamajiem variantiem, ja būs tikai viena iespēja, tad komanda tiks papildināta līdz galam:

```
user@serveris:~$ fr <TAB><TAB>
free fribidi from
user@serveris:~$ fr e <TAB>
user@serveris:~$ fr ee
```

Komandrindā ierakstot burtus fr un nospiežot TAB taustiņu 2 reizes, tiek parādītas visas komandas, kas sākas ar fr. Ja komandas papildināšanai ir vairākas iespējas, tad TAB taustiņš jāspiež 2 reizes, ja tikai viena iespēja, tad tā tiks automātiski papildināta, nospiežot TAB taustiņu vienu reizi. Ievadot burtu e un nospiežot TAB taustiņu vēlreiz, komanda tiek papildināta, jo ir tikai viena, kas sākas ar burtiem fre un tā ir komanda free.

Līdzīgi papildināšana darbojas norādot direktoriju vai failu nosaukumus, piemēram: user@serveris:~\$ cd /h <TAB> user@serveris:~\$ cd /home/ <TAB> user@serveris:~\$ cd /home/user/

Saknes (/) direktorijā ir tikai viena mape, kas sākas ar burtu h, savukārt /home mapē vispār ir tikai viena mape, tāpēc nospiežot taustiņu TAB tā tiek automātiski ierakstīta. TAB taustiņa izmantošana ne tikai ievērojami paātrina komandu ievadīšanu, bet arī samazina kļūdu pieļaušanas iespējas nepareizi ievadot komandas nosaukumu vai adresi.

Lai vēlreiz izsauktu nesen izpildītu komandu, var izmantot history (vēsture) rīku, kas izdrukā pēdējo izpildīto komandu sarakstu (atkarībā no iestatījumiem un iepriekš izpildīto komandu daudzuma saraksta garums var atšķirties):

```
120
     ls
121 cp fails2 fails1
122
    ls
123
    mv fails1 manamape
124
    ls
125
    ls manamape
126
    cd manamape
127
     ls
128
    mv fails1 1fails
129
    ls
```

Lai vēlreiz izpildītu kādu no redzamajām komandām jāievada izsaukuma zīme ar komandas numuru:

```
user@serveris:~$ !129
ls
mansfails Mansfails
Par citām BASH čaulas iespējām skatīt: man bash
```

## Failu lejupielādēšana no tīmekļa

Izmantojot grafiskās vides saskarni, failus no tīmekļa var lejupielādēt izmantojot pārlūkprogrammu, savukārt komandrindas saskarnē var attēlot tikai parastu tekstu, tāpēc

pārlūkprogrammas netiek plaši izmantotas. Bet tādas ir pieejamas, piemēram, lynx. Tomēr var gadīties, ka nepieciešams lejuepielādēt kādu failu no tīmekļa. Tam paredzēts rīks wget. Tas spēj strādāt gan ar HTTP (tai skaitā HTTPS), gan FTP protokoliem. Jāzina precīza faila adrese un tā jāievada kā parametrs:

user@serveris:~\$ wget ftp://ftp.linux.lv/pub/ubuntu/10.04.2/ubuntu-10.04.2-server-i386.iso --2011-01-10 14:28:47-- ftp://ftp.linux.lv/pub/ubuntu/10.04.2/ubuntu-10.04.2-server-i386.iso

=> `ubuntu-10.04.2-server-i386.iso'
Resolving ftp.linux.lv... 195.13.158.141
Connecting to ftp.linux.lv|195.13.158.141|:21... connected.
Logging in as anonymous ... Logged in!
==> SYST ... done. ==> PWD ... done.
==> TYPE I ... done. ==> CWD (1) /pub/ubuntu/10.04.2 ... done.
==> SIZE ubuntu-10.04.2-server-i386.iso ... 707500032
==> PASV ... done. ==> RETR ubuntu-10.04.2-server-i386.iso ... done.
Length: 707500032 (675M) (unauthoritative)

100%[=======] 707,500,032 3.51M/s in 2m 7s

2011-01-10 14:30:55 (5.32 MB/s) - `ubuntu-10.04.2-server-i386.iso' saved [707500032]

Fails ubuntu-10.04.2-server-i386.iso tiek saglabāts esošajā direktorijā. Šādi ar wget var lejupielādēt arī "mājas lapu", t.i. HTML valodā pierakstītu failu. Grafiskajā vidē pārlūkprogramma šo failu saprot un attēlo kā ierasts redzēt mājaslapas, savukārt, komandrindā ar iepriekš aprakstītajiem rīkiem var to apskatīties HTML pierakstā:

```
user@serveris:~$ wget http://www.lu.lv
--2011-01-10 14:31:17-- http://www.lu.lv/
Resolving www.lu.lv... 195.13.129.210
Connecting to www.lu.lv|195.13.129.210|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 285676 (279K) [text/html]
Saving to: `index.html'
```

100%[===================] 285,676 --.-K/s in 0.03s 2011-01-10 14:31:17 (9.29 MB/s) - `index.html' saved [285676/285676] Ar komandu less var apskatīties index.html faila saturu.

## Administratīvu funkciju veikšana

*Linux* vidē lietotāji pēc noklusējuma darbojas ar ierobežotām pieejas tiesībām kā t.s. nepriviliģētie lietotāji. Ja lietotājam ir jāveic kāda administratīva funkcija (piemēram, labot

sistēmas konfigurācijas failus vai uzstādīt programmatūru), tad pirms tam jāpārslēdzas uz administratora režīmu, jāizpilda attiecīgās darbības un tad jāpārslēdzas atpakaļ uz sākotnējo stāvokli. Šādu pārslēgšanos var veikt tikai tie sistēmas lietotaji, kuriem tas ir atļauts. Piemēram, pirmais uzstādītais lietotājs instalācijas laikā ir ar šādām privilēģijām.

Lai izpildītu komandu ar administratora (*root*) tiesībām, paredzēts rīks Sudo. Ja komandai sākumā pievieno Sudo, tā tiek izpildīta ar administratora tiesībām (lietotāju). Lai to izpildītu, lietotājam jāievada sava parole, un jābūt attiecīgām privilēģijām izmantot šo komandu:

user@serveris:~\$ sudo nano /etc/network/interfaces

[sudo] password for user:

Fails /etc/network/interfaces tiek atvērts ar administratora tiesībām, to var labot un saglabāt. Tiklīdz nano redaktors tiek aizvērts, lietotājs atgriežas savā bāzes privilēģiju līmenī un darbojas ar ierobežotām tiesībām. Rīks Sudo lietotāja paroli neprasīs biežāk kā reizi 15 minūtēs, bet izveidos sesiju, kuras laikā atcerēsies, ka lietotājs jau autentificējies ar savu paroli. Lai slēgtu Sudo sesiju, negaidot 15 minūtes, jāievada Sudo -k. Ar Sudo -i iespējams pārslēgties uz administratora režīmu, lai pirms katras komandas nebūtu jāraksta Sudo. Ievērojiet, ka komandrindā \$ zīmes vietā ir # un lietotājvārds ir root, tas norāda, ka darbojaties administratora režīmā. Lai pārietu atpakaļ uz ierobežotā lietotāja režīmu, jāievada

*Ubuntu Linux* vidē pēc noklusējuma Sudo var izmantot pirmais sistēmā uzstādītais lietotājs un tie lietotāji, kas ir pievienoti admin grupai (patiesībā pirmais lietotājs tiek automātiski pievienots šai grupai).

## Programmatūras uzstādīšana un atjaunināšana

Svarīga brīvās programmatūras veidošanas iezīme ir tas, ka izstrādājot jaunu programmatūru, tās veidotāji var izmantot jau iepriekš izstrādātu programmatūru kā jaunās komponentes un pievērsties inovācijām, nevis vēlreiz realizēt to, kas ir jau gatavs. Esošo brīvo programmatūru var iedomāties kā detaļas, kas jau ir pieejamas, bet trūkstošās tiek izveidotas no jauna. *Linux* vidē šādas "detaļas", no kurām kopā tiek "salikta" programma sauc par pakotnēm.

Pakotne - fails vai vairāku failu kopums, kas ir nepieciešams, lai izpildītu datora programmu vai lai jau iepriekš uzstādītai programmai pievienotu papildu funkcionalitāti.

Pakotnēm tiek noteiktas tā saucamās atkarības, t.i. prasības pēc citām pakotnēm. Piemēram, lai varētu uzstādīt vienu pakotni, pirms tam jāuzstāda kāda cita. Pēc analoģijas ar detaļām, lai saskrūvētu kopā divas plāksnes, vajag ne tikai skrūvi, bet arī uzgriezni. Lietotājam par atkarībām parasti nav jāsatraucas, *Linux* sistēmā iebūvētie rīki tās risina automātiski. Atšķirībā no citām sistēmām, *Linux* programmatūra katram distributīvam tiek glabāta centralizēti un lietotājam pieejama no pašas sistēmas, nevis meklējot to izstrādātāju mājaslapā. Šādas programmatūras glabātuves sauc par repozitorijiem. Tajos glabājas programmatūras, kas īpaši sagatavota konkrētajam *Linux* distributīvam un versijai. Pašā *Linux* sistēmā ir rīki, kas paredzēti darbam ar repozitorijiem. Ar tiem var lejupielādēt informāciju par pieejamo programatūru un versijām (to varētu salīdzināt ar satura rādītāju), atkarībām pēc citām pakotnēm un atrašanās vietu uz servera (URL). Ar šiem rīkiem var uzstādīt jaunu programmatūru, noņemt esošu, atjaunināt vienu, vairākas vai visas programmas, kam pieejamas jaunākas versijas. Uz jaunāku versiju iespējams atjaunināt pat visu operētājsistēmu.

*Ubuntu Linux* sistēmā komandrindā pieejami vairāki rīki darbam ar pakotnēm, plašāk pielietotais ir apt. Sākotnēji tas tika izstrādāts *Debian* sistēmai, un ir pieejams *Debian Linux* bāzētos distributīvos tai skaitā *Ubuntu*. apt sastāv no vairākiem komandrindas rīkiem, kas nodrošina darbu ar pakotnēm, tai skaitā apt-cache – meklēšanas rīks, apt-get – uzstādīšanas/noņemšanas rīks, u.c.

Pamatrīks darbam ar pakotņu pārvaldību *Ubuntu Linux* sistēmā ir apt-get. Ar to var lejupielādēt jaunāko pieejamo pakotņu sarakstu, uzstādīt jaunu programmatūru, atjaunināt vai noņemt jau uzstādīto.

#### Manipulācija ar pakotnēm

Jaunākās programmatūras saraksta atjaunināšana – sudo apt-get update,

Visu uzstādīto pakotņu atjaunināšana uz jaunākajām versijām – sudo apt-get upgrade,

Jaunas programmatūras uzstādīšana – sudo apt-get install pakotnes\_nosaukums,

Programmatūras noņemšana – sudo apt-get remove pakotnes\_nosaukums,

Jau uzstādītas programmatūras pārinstalēšana — sudo apt-get install --reinstall pakotnes\_nosaukums.

Lai uzstādītu kādu programmu, jāzina precīzs tās nosaukums, turklāt repozitoriju sistēmā šis nosaukums var būt nedaudz citādāks nekā paša produkta, piemēram, *OpenOffice* biroja programmatūras pakotnes nosaukums ir openoffice.org.

Lai noskaidrotu kādas pakotnes nosaukumu vai atrastu kādu pakotni zinot tās daļēju nosaukumu vai funkciju ko tā pilda, paredzēts apt-cache rīks. Tas izmanto lejupielādēto informāciju par pakotnēm un nodrošina meklēšanu pārbaudot pakotnes nosaukumu, aprakstu u.c. Informatīvos laukus. Piemēram, apt-cache search buggy komandas izvads:

user@serveris:~\$ apt-cache search buggy

```
dh-buildinfo - Debhelper addon to track package versions used to build a
package
    centerim-utf8 - A text-mode multi-protocol instant messenger client
    chimera2 - Web browser for X
    cupt - alternative front-end for dpkg -- console interface
    ddtc - Deal with ddts mails
    libxml-filter-reindent-perl - Perl module for reformatting whitespace
for pretty printing XML
    mcrypt - Replacement for old unix crypt(1)
    moon-buggy - Drive a car across the moon
    moon-buggy-esd - Drive a car across the moon (version with sound)
    netfilter-extensions-source - source for
                                                               kernel modules
                                                   netfilter
derived from patch-o-matic-ng
    torcs-data-cars - data files for TORCS game - Cars set
    uruk - Very small firewall script, for configuring iptables
    Pakotnes, kas aprakstā vai nosaukumā satur vārdu buggy (no anglu val. bagijs) ir 12, bet
```

pēc apraksta un pilnā nosaukuma var ērti noskaidrot, kura ir īstā. Šajā gadījumā, ja mērķis būtu uzstādīt komandrindas saskarnes spēlīti *Moon Buggy*, tad pēc meklēšanas rezultātu apskates varētu secināt, ka atbilstošās pakotnes nosaukums ir moon-buggy.

Lai apskatītos plašāku informāciju par izvēlēto pakotni, jāizmanto komanda apt - cache show:

```
user@serveris:~$ apt-cache show moon-buggy
Package: moon-buggy
Priority: optional
Section: universe/games
Installed-Size: 368
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Original-Maintainer: Christian T. Steigies <cts@debian.org>
Architecture: i386
Version: 1.0.51-8
Depends: libc6 (>= 2.4), libncurses5 (>= 5.6+20071006-3), debconf (>=
```

```
0.5) | debconf-2.0, dpkg (>= 1.15.4) | install-info
Conflicts: moon-buggy-esd, moon-buggy-pause, suidmanager (<< 0.50)
Filename: pool/universe/m/moon-buggy/moon-buggy_1.0.51-8_i386.deb
Size: 212386
MD5sum: 5c1004cd368c89a134651755551114fc
SHA1: 2ba0ef056f7d14e1c0d793ba263089aedc3dd2c8
SHA256: 103a83c85e722edd027dc08b82e1216ce553d36ec10baee36334be2289bab5b8
Description: Drive a car across the moon
Moon-buggy is a simple character graphics game, where you drive some
kind of car across the moon's surface. Unfortunately there are
dangerous craters there. Fortunately your car can jump over them!
```

Homepage: http://seehuhn.de/pages/moon-buggy Bugs: https://bugs.launchpad.net/ubuntu/+filebug Origin: Ubuntu Lai uzstādītu *Moon Buggy*, jāraksta sudo apt-get install moon-buggy: user@serveris:~\$ sudo apt-get install moon-buggy Reading package lists... Done Building dependency tree Reading state information... Done The following NEW packages will be installed: moon-buggy 0 upgraded, 1 newly installed, 0 to remove and 80 not upgraded. Need to get 212kB of archives. After this operation, 377kB of additional disk space will be used. Get:1 http://lv.archive.ubuntu.com/ubuntu/ lucid/universe moon-buggy 1.0.51-8 [212kB] Fetched 212kB in Os (1,018kB/s) Preconfiguring packages ... Selecting previously deselected package moon-buggy. (Reading database ... 24326 files and directories currently installed.) Unpacking moon-buggy (from .../moon-buggy\_1.0.51-8\_i386.deb) ... Processing triggers for install-info ... Processing triggers for man-db ... Setting up moon-buggy (1.0.51-8) ... Ar komandu moon-buqqy var palaist tikko uzinstalēto spēlīti un iemēģināt roku pie stūres bagijam, kas pārvietojas pa Mēnesi. Tikai uzmanīgi, spēle mēdz aizraut. Esmu brīdinājis! Lai atinstalētu *Moon Buggy*, jāizmanto komanda apt-get remove: user@serveris:~\$ sudo apt-get remove moon-buggy Reading package lists... Done Building dependency tree Reading state information... Done The following packages will be REMOVED: moon-buggy 0 upgraded, 0 newly installed, 1 to remove and 80 not upgraded. After this operation, 377kB disk space will be freed. Do you want to continue [Y/n]? Y (Reading database ... 24345 files and directories currently installed.) Removing moon-buggy ... Processing triggers for man-db ... Processing triggers for install-info ...

Lai atjauninātu visas uzstādītās pakotnes vispirms vēlams ielādēt jaunāko pakotņu sarakstu ar komandu apt-get update (daļa no izvada):

user@serveris:~\$ sudo apt-get update

Get:1 http://lv.archive.ubuntu.com lucid Release.gpg [189B]
Ign http://lv.archive.ubuntu.com/ubuntu/ lucid/main Translation-en\_US
Ign http://lv.archive.ubuntu.com/ubuntu/ lucid/restricted Translationen\_US

Ign http://lv.archive.ubuntu.com/ubuntu/ lucid/universe Translationen\_US

Ign http://lv.archive.ubuntu.com/ubuntu/ lucid/multiverse Translationen\_US

Hit http://security.ubuntu.com lucid-security/main Packages Hit http://security.ubuntu.com lucid-security/restricted Packages Hit http://security.ubuntu.com lucid-security/main Sources Hit http://security.ubuntu.com lucid-security/restricted Sources Hit http://security.ubuntu.com lucid-security/universe Packages Get:6 http://lv.archive.ubuntu.com lucid/restricted Packages [6,208B] Get:7 http://lv.archive.ubuntu.com lucid/main Sources [659kB] Hit http://security.ubuntu.com lucid-security/universe Sources Hit http://security.ubuntu.com lucid-security/multiverse Packages Hit http://security.ubuntu.com lucid-security/multiverse Sources Get:8 http://lv.archive.ubuntu.com lucid/restricted Sources [3,775B] Get:9 http://lv.archive.ubuntu.com lucid/universe Packages [5,448kB] Get:10 http://lv.archive.ubuntu.com lucid/universe Packages [5,448kB] Get:11 http://lv.archive.ubuntu.com lucid/universe Sources [3,165kB] Get:12 http://lv.archive.ubuntu.com lucid/multiverse Packages [180kB] Get:13 http://lv.archive.ubuntu.com lucid/multiverse Sources [119kB] Get:14 http://lv.archive.ubuntu.com lucid-updates/main Packages [457kB] http://lv.archive.ubuntu.com lucid-updates/universe Get:18 Packages

#### [188kB]

Get:19 http://lv.archive.ubuntu.com lucid-updates/universe Sources
[69.7kB]

Fetched 12.0MB in 18s (662kB/s)

Reading package lists... Done

Kad pakotņu saraksts ir ielādēts, var atjaunināt visas uzstādītās pakotnes:

user@serveris:~\$ sudo apt-get upgrade

Reading package lists... Done

Building dependency tree

Reading state information... Done

The following packages have been kept back:

landscape-common linux-generic linux-image-generic

The following packages will be upgraded:

apparmor apparmor-utils apt apt-transport-https apt-utils at basefiles bind9-host bsdutils bzip2 coreutils dmsetup dnsutils dpkg e2fslibs e2fsprogs fuse-utils grub-common grub-pc gzip ifupdown language-pack-en language-pack-en-base libapparmor-perl libapparmor1 libbind9-60 libblkid1 libbz2-1.0 libc-bin libc6 libc6-i686 libcomerr2 libdbus-1-3 libdevmapperevent1.02.1 libdevmapper1.02.1 libdns64 libfuse2 libgssapi-krb5-2 libisc60 libisccc60 libisccfg60 libk5crypto3 libkrb5-3 libkrb5support0 liblwres60 libplymouth2 libss2 libss10.9.8 libudev0 libuuid1 libwww-perl libxml2 linuxfirmware linux-image-2.6.32-24-generic login lvm2 man-db mount mountall openssl passwd plymouth plymouth-theme-ubuntu-text python-lazr.restfulclient rsyslog screen sudo tar tzdata udev update-manager-core upstart ureadahead util-linux uuid-runtime wget xkb-data

77 upgraded, 0 newly installed, 0 to remove and 3 not upgraded. Need to get 74.0MB of archives. After this operation, 1,253kB of additional disk space will be used. Do you want to continue [Y/n]?

Pēc apstiprināšanas (Y), no repozitorija tiek lejupielādētas un uzstādītas jaunākās pakotņu versijas. Atkarībā no atjaunojamo pakotņu daudzuma, interneta pieslēguma ātruma un datora jaudas, šis process var aizņemt sākot no dažām sekundēm līdz vairākiem desmitiem minūtēm.

Ja starp atjaunotajām pakotnēm ir arī *Linux* kodols (linux-image), tad, lai veiktās izmaiņas stātos spēkā, jāpārstartē dators. *Linux* kodola atjaunināšana iespējams ir vienīgā reize, kad nepieciešams pārstartēt sistēmu. Pēc servera programmatūras, piemēram, tīmekļa servera, atjaunināšanas *Linux* sistēma nav jāpārstartē, atliek pārstartēt tikai attiecīgo programmu. Tā kā *Linux* sistēmas ir stabilas un kodola atjauninājumi iznāk salīdzinoši reti, tad tās bez pārstartēšanas var darboties pat gadiem ilgi.

Dažkārt var gadīties tā, ka vēlamā programmatūra repozitorijos nav pieejama un ar apt get to nevar uzstādīt. Tad tā jāmeklē tīmeklī. Skatoties versiju, jāpievērš uzmanību, kādam *Linux* distributīvam un versijai tā ir domāta. Zinot to, ka uzstādītā *Ubuntu Linux* versija ir 10.04 (to var noskaidrot ar komandu lsb\_release -a), tad jāmeklē programmas versija, kas atbilst uzstādītajai *Linux* versijai. Tomēr ne vienmēr programmām būs atsevišķa versija tieši konkrētajam distributīva laidienam. Dažkārt var būt rakstītas, ka tā der, piemēram, *Ubuntu Linux* 9.10 un jaunākām, tātad arī 10.04. Vēl var gadīties, ka *Ubuntu* un *Debian Linux* distributīviem ir viena un tā pati versija. Tā kā *Ubuntu Linux* ir bāzēts uz *Debian*, tad ir programmas, kuru viena un tā pati versija darbojas gan *Ubuntu*, gan *Debian* vidēs, tomēr uz to nevajag paļauties un uztvert to kā izņēmumu! *Ubuntu* (un arī *Debian*) *Linux* pakotnēm ir paplašinājums . deb, pēc tā var atpazīt, ka tā ir šiem distributīviem paredzēta pakotne.

Kad vēlamā pakotne atrasta un lejupielādēta (ar wget), tad to var uzinstalēt ar komandu dpkg. Tā kā šī pakotne nav repozitorijos, tad tā jāuzstāda pašrocīgi kā parametru norādot pakotnes failu, tam paredzēts rīks dpkg:

sudo dpkg -i pakotne.deb

Visu uzstādīto pakotņu sarakstu var apskatīt ar komandu dpkg -l (daļa no izvada): user@serveris:~\$ dpkg -l Desired=Unknown/Install/Remove/Purge/Hold

```
Status=Not/Inst/Cfg-files/Unpacked/Failed-cfg/Half-inst/trig-
    1
aWait/Trig-pend
    // Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
    ||/ Name
                      Version
                                      Description
    ii adduser
                3.112ubuntu1
                                 add and remove users and groups
    ii apparmor
                2.5.1-Oubuntu0.10.04.3 User-space
                                                  parser
                                                          utility
                                                                   for
AppArmor
    ii apparmor-utils 2.5.1-Oubuntu0.10.04.3 Utilities
                                                       for
                                                            controlling
AppArmor
    ii apport
                1.13.3-0ubuntu2
                                 automatically generate crash reports for
deb
    ii apport-symptoms 0.9
                           symptom scripts for apport
    ii apt 0.7.25.3ubuntu9.3 Advanced front-end for dpkg
                           0.7.25.3ubuntu9.3 APT https transport
    ii apt-transport-https
    ii apt-utils 0.7.25.3ubuntu9.3 APT utility programs
```

## Serveru programmatūras uzstādīšana

Programmatūras pārvaldības pamati ir apskatīti, var uzstādīt serveru programmatūru. Šī kursa ietvaros tiks apgūta tīmekļa servera, domēna vārdu servera (DNS), Windows failu koplietošanas servera *Samba* un attālinātās piekļuves SSH servera uzstādīšana un konfigurēšana.

*Linux* sistēmu komandrindas saskarnē var pārvaldīt arī attālināti. Nav obligāti jāsēž pie fiziskā datora. Tas pat nav īpaši ērti, jo serveri bez grafiskās vides saskarnes ir grūtāk administrēt gadījumā, ja vajag uzmeklēt informāciju tīmeklī vai vienlaicīgi strādāt ar vairākiem rīkiem. Tāpēc viena no pirmajām programmām, ko administratori parasti uzstāda *Linux* serverim ir attālinātās piekļuves serveris SSH (no *Secure Shell* – drošā čaula), kas ļauj tam pieslēgties no jebkura datora, kam ir interneta pieslēgums un SSH klienta programmatūra (*Windows, Linux, Mac* vai pat viedtelefons). Kad tā uzstādīta, administrators var turpināt servera konfigurēšanu un instalēšanu jau no sava datora, kas parasti aprīkots ar grafisko vidi, pārlūkprogrammu un citiem ikdienā ierastiem rīkiem.

SSH servera pakotnes nosaukums ir openssh-server (to var noskaidrot ar jau iepriekš apskatīto rīku apt-cache):

user@serveris:~\$ sudo apt-get install openssh-server Reading package lists... Done Building dependency tree... Reading state information... Done The following extra packages will be installed: libwrap0 openssh-client tcpd Suggested packages:

ssh-askpass libpam-ssh keychain openssh-blacklist openssh-blacklistextra rssh molly-guard The following NEW packages will be installed: libwrap0 openssh-server tcpd The following packages will be upgraded: openssh-client 1 upgraded, 3 newly installed, 0 to remove and 0 not upgraded. Need to get 1,046kB/1,147kB of archives. After this operation, 1,065kB of additional disk space will be used. Do you want to continue [Y/n]? Y [Working] Get:1 http://lv.archive.ubuntu.com/ubuntu/ lucid-updates/main opensshclient 1:5.3p1-3ubuntu5 [761kB] [1 openssh-client OB/761kB 0%] Get:2 http://lv.archive.ubuntu.com/ubuntu/ lucid-updates/main opensshserver 1:5.3p1-3ubuntu5 [285kB] [2 openssh-server 24.8kB/285kB 8%] Fetched 1,046kB in 0s (3,761kB/s) Preconfiguring packages ... Reading database ... 24281 files and directories currently installed.) Preparing to replace openssh-client 1:5.3p1-3ubuntu4 (using .../opensshclient\_1%3a5.3p1-3ubuntu5\_i386.deb) ... Unpacking replacement openssh-client ... Selecting previously deselected package libwrap0. Unpacking libwrap0 (from .../libwrap0\_7.6.g-18\_i386.deb) ... Selecting previously deselected package openssh-server. .../openssh-server\_1%3a5.3p1-Unpacking openssh-server (from 3ubuntu5\_i386.deb) ... Selecting previously deselected package tcpd. Unpacking tcpd (from .../tcpd\_7.6.g-18\_i386.deb) ... Processing triggers for man-db ... Processing triggers for ureadahead ... ureadahead will be reprofiled on next reboot Processing triggers for ufw ... Setting up openssh-client (1:5.3p1-3ubuntu5) ... Setting up libwrap0 (7.6.q-18) ... Setting up openssh-server (1:5.3p1-3ubuntu5) ... Creating SSH2 RSA key; this may take some time .... Creating SSH2 DSA key; this may take some time ... \* Stopping OpenBSD Secure Shell server sshd [ OK ] ssh start/running, process 1134

Setting up tcpd (7.6.q-18) ...

Processing triggers for libc-bin ...

ldconfig deferred processing now taking place

Pēc uzstādīšanas SSH serveris tiek iestartēts. Pēc noklusējuma tas izmanto 22. pieslēgvietu (*port*), tātad, ja dators atrodas aiz ugunsmūra, tad jāatļauj pieeja 22. portam. Saziņai starp klientu un serveri SSH izmanto šifrētu kanālu, kas tiek izveidots saziņas sākumā. Visi dati (tai skaitā paroles), kas tiek pārsūtīti ir šifrēti. Lai pieslēgtos attālinātam SSH serverim, jāzina tā IP adrese (komanda ifconfig) un jābūt uzstādītam SSH klientam. *Linux* un *Mac OS X* sistēmās SSH klienta programmatūra ir uzstādīta pēc noklusējuma, atliek tikai atvērt komandrindas saskarnes termināli un pieslēgties ievadot lietotājvārdu un IP adresi:

lietotajs@darbstacija:~\$ ssh user@10.0.2.15

The authenticity of host '10.0.2.15 (10.0.2.15)' can't be established. RSA key fingerprint is d7:27:44:a6:8a:5f:99:fc:4d:e6:47:5a:48:8f:12:11. Are you sure you want to continue connecting (yes/no)? yes Warning: Permanently added '10.0.2.15' (RSA) to the list of known hosts. user@10.0.2.15's password: user@serveris:~\$

Šajā gadījumā attālinātā datora IP adrese ir **10.0.2.15** un lietotājvārds - user. Pirmo reizi pieslēdzoties datoram, tiek pārvaicāts vai dators, ar kuru tiek veidots savienojums ir drošs, ja jā, tad jāievada lietotāja parole un tālāk darbošanās notiek jau attālinātajā datorā. Visas komandas, kas tiek ievadītas tiek izpildītas tieši tāpat, ja lietotājs fiziski sēdētu pie tā datora un komandas ievadītu no datoram pieslēgtās tastatūras.

Lai pieslēgtos no darbstacijas, kas aprīkota ar Windows operētājsistēmu, tad nepieciešms lejupielādēt programmu PuTTY (mājslapa: http://www.chiark.greenend.org.uk/~sgtatham/putty/). Vajadzīgs tikai putty.exe, to var izpildīt uzreiz bez uzstādīšanas. Atverot *PuTTY*, galvenajā logā jāievada attālinātā datora IP adrese un jāspiež *Open*. Ja attālinātajam datoram plānojat pieslēgties regulāri vai mainīt kādus no pieslēguma iestatījumiem, tad šo pieslēgumu var saglabāt *Saved sessions* sadaļā un nākamreiz, atverot *Putty*, varēs ātrāk pieslēgties attālinātajam datoram izmantojot savus iestatījumus:

RuTTY Configuration	×
Category:	
Session     Logging     Terminal     Keyboard     Bel     Features     Window     Appearance     Behaviour     Translation     Selection     Colours     Connection     Data     Proxy     Telnet     Rlogin     SSH     SSH     Serial	Basic options for your PuTTY session         Specify the destination you want to connect to         Host Name (or IP address)       Port         10.0.2.15       [22]         Connection type:       Raw         C Raw       Telnet         Basic options for your want to connect to         Host Name (or IP address)       Port         10.0.2.15       [22]         Connection type:       Case         C Raw       Telnet         Basic options       Serial         Load, save or delete a stored session       Save         Default Settings       Load         Save       Delete         Close window on exit:       Close window on exit:         Always       Never
About	Open Cancel

Attēls 18: PuTTY galvenais logs

Tāpat kā izmantojot komandrindas SSH klientu, pirmo reizi pieslēdzoties attālinātajam datoram, tiek pārvaicāts vai dators, ar kuru tiek veidots savienojums ir drošs.



Attēls 19: Vai attāllinātais dators ir drošs?

Ja jā, tad jāievada lietotājvārds un parole un tālāk darbošanās notiek jau attālinātajā datorā. Tieši tāpat kā izmantojot *Linux* vides SSH klientu vai komandas ievadot no serverim pieslēgtās tastatūras.



Attēls 20: SSH sesija izmantojot PuTTY

## Mājaslapu, DNS un Samba serveru uzstādīšana

Plaši pielietota mājaslapu servera konfigurācija sastāv no 4 galvenajām komponentēm – *Linux* operētājsistēmas, *Apache* tīmekļa servera, **M**ySQL datu bāžu servera, kurā glabā mājaslapas datus un PHP programmēšanas valodas atbalsta, kas nodrošina dinamisku saturu. Šo programmatūras kopumu bieži apzīmē ar abbreviatūru LAMP. *Linux* operētājsistēma jau ir uzstādīta, tātad, lai realizētu mājaslapu serveri nepieciešams vēl uzstādīt *Apache*, MySQL un PHP. Tā kā katra no šīm komponentēm sastāv no vairākām pakotnēm un pēc uzstādīšanas var nākties konfigurēt, lai tās darbotos kopā, tad *Ubuntu* izstrādātāji ir izveidojuši biežāk lietoto serveru instalāciju šablonus, kas realizē kādu uzdevumu. Piemēram, lai uzstādītu pilnībā funkcionējošu LAMP serveri, atliek atzīmēt "uzstādīt LAMP serveri" un visas nepieciešamās komponentes tiek uzinstalētas un iedarbinātas. Šīm darbībām paredzēts rīks tasksel (no angļu val. *select task* – izvēlēties uzdevumu). Izstrādātāji ir izveidojuši plašu klāstu ar uzdevumu bāzētu programmatūras izvēli, tai skaitā šī kursa ietvaros apskatīto LAMP, DNS un failu serveru uzstādīšanai, izpildot komandu sudo tasksel, tiek parādīta izvēlne ar visiem iespējamiem uzdevumiem:

You can choose to install one or more collections of software.	ction following predefined
Choose software to install:	
<pre>[ ] DNS server [ ] Edubuntu server [ ] LAMP server [ ] Mail server [ ] OpenSSH server [ ] PostgreSQL database [ ] Print server [ ] Samba file server [ ] Tomcat Java server</pre>	
<0k>	

Attēls 21: Servera programmatūras uzstādīšanas izvēle

Ar tastatūtas taustiņiem var pārvietoties pa uzdevumu sarakstu un ar atstarpi atzīmēt uzstādīšanai, lai apstiprinātu izvēlētās programmatūras uzstādīšanu un pārietu uz  $\langle Ok \rangle$ , jāspiež *Tab* taustiņš, pēc tam *Enter*. Kā redzams, iepriekš uzstādītais SSH serveris pieejams arī šajā sarakstā un to varēja uzstādīt arī izmantojot tasksel. Šī kursa ietvaros tiks uzstādīti sekojošie 3 serveri:

- *LAMP server* tīmekļa serveris
- DNS server domēna vārdu sistēmas serveris
- Samba file server Windows failu koplietošanas serveris

Pēc izvēles apstiprināšanas atzīmētās izmaiņas tiek veiktas un attiecīgā programmatūra uzstādīta.

## Linux pārvaldības rīks Webmin

Webmin ir Linux operētājsistēmas adminstrēšanas rīks. Tas ir bāzēts uz tīmekļa tehnoloģijām — Webmin var izmantot ar jebkuru mūsdienu tīmekļa pārlūkprogrammu, piemēram, Mozilla Firefox. Webmin nav pieejams Ubuntu repozitorijos, tāpēc to nepieciešams lejupielādēt no Webmin mājaslapas <u>www.webmin.com</u> – kreisajā Download -> Debian package (šis ir viens no tiem retajiem gadījumiem, kad *Ubuntu* un *Debian* ir viena un tā pati pakotne) ir pieejama saite uz jaunāko Webmin versiju. Nokopējot adresi un ievietojot to kā parametru wget komandai komandrindā, *Webmin* var uzreiz lejupielādēt uz serveri:

user@serveris:~\$

http://prdownloads.sourceforge.net/webadmin/webmin\_1.530\_all.deb --2011-01-11

16:13:58--

16:13:58--

wget

http://prdownloads.sourceforge.net/webadmin/webmin\_1.530\_all.deb

Resolving prdownloads.sourceforge.net... 216.34.181.59

Connecting to prdownloads.sourceforge.net|216.34.181.59|:80... connected.

HTTP request sent, awaiting response... 301 Moved Permanently Location:

http://downloads.sourceforge.net/project/webadmin/webmin/1.530/webmin\_1.530\_a ll.deb [following]

--2011-01-11 http://downloads.sourceforge.net/project/webadmin/webmin/1.530/webmin\_1.530\_a

ll.deb

Resolving downloads.sourceforge.net... 216.34.181.59

Reusing existing connection to prdownloads.sourceforge.net:80.

HTTP request sent, awaiting response... 302 Found

Location:

http://netcologne.dl.sourceforge.net/project/webadmin/webmin/1.530/webmin\_1.5 30\_all.deb [following]

--2011-01-11

16:13:59--

http://netcologne.dl.sourceforge.net/project/webadmin/webmin/1.530/webmin\_1.5 30\_all.deb

Resolving netcologne.dl.sourceforge.net... 78.35.24.46

Connecting to netcologne.dl.sourceforge.net|78.35.24.46|:80... connected.

HTTP request sent, awaiting response... 200 OK Length: 14643982 (14M) [application/x-debian-package]

Saving to: `webmin\_1.530\_all.deb'

100%[=====>] 14,643,982 314K/s in 37s 2011-01-11 16:14:36 (384 KB/s) - `webmin\_1.530\_all.deb' saved [14643982/14643982]

Pirms Webmin uzstādīšanas vēlams apskatīties *Webmin* instalēšanas pamācību, kas atrodama mājaslapā (*Webmin installation -> Installing on Debian* <u>http://www.webmin.com/deb.html</u>). Izstrādātāji raksta, ka pirms *Webmin* uzstādīšanas, jāuzinstalē pakotnes, kas nepieciešamas, lai tas varētu strādāt:

sudo apt-get install perl libnet-ssleay-perl openssl libauthen-pam-perl
libpam-runtime libio-pty-perl apt-show-versions

Pēc šo pakotņu uzstādīšanas var instalēt Webmin: user@serveris:~\$ sudo dpkg -i webmin\_1.530\_all.deb Selecting previously deselected package webmin. (Reading database ... 24485 files and directories currently installed.) Unpacking webmin (from webmin\_1.530\_all.deb) ... Setting up webmin (1.530) ... Webmin install complete. You can now login to https://serveris:10000/ as root with your root password, or as any user who can use sudo to run commands as root.

Processing triggers for ureadahead ...

Lai pieslēgtos uzstādītajam *Webmin* rīkam, pārlūkprogrammā jāievada adrese <u>https://10.0.2.15:10000</u> (10.0.2.15 nomainot pret konkrētā servera IP adresi). Pieslēgšanās logā jāraksta savs lietotājsvārds un parole.

Par Webmin izmantošanu un serveru programmatūras konfigurēšanu, lūdzu, lasīt materiālu Linux servera ieviešana (L. Trukšāns, K. Krampis, Rīga, 2010). Materiāls pieejams: <u>http://profizgl.lu.lv/mod/resource/view.php?id=16454</u> (viesa piekļuves parole ir alfa1).