

Naloge za ponavljanje ukazov v Linux konzoli

- Ugotovi svojo trenutno lokacijo (v katerem imeniku se nahajaš)
 - **pwd**
- Izpiši uporabniško ime
 - **logname, whoami**

Izdelava imenikov(map)

- v imeniku Documents izdelaj imenik vaje_linux
 - **mkdir vaje_linux**
- ustvari podimenik teksti, za tem pa še imenik apj
 - **mkdir teksti**
- ustvari podimenik pra (absolutno)
 - **mkdir /home/uporabnik/Documents/vaje_linux/pr**
- v imeniku vaje_linux ustvari več podimenikov (imena imenikov: ena, dva, tri) hkrati
 - **mkdir ena dva tri**

Sprememba delovnega imenika

- preveri v katerem imeniku si in se pomakni v imenik vaje_linux
 - **pwd**
 - **cd vaje_linux**
- pomik v podimenik apj imenika vaje_linux
 - **cd vaje_linux/apj**
- pomakni se v imenik Documents (uporabi absolutni pomik)
 - **cd /home/uporabnik/Documents/**
- absolutni pomik v imenik z vašim uporabniškim imenom
 - **cd /home/uporabnik/**
- pomakni se v starševski imenik (en nivo višje)
 - **cd ..**
- pomakni se v domači imenik
 - **cd cd ~uporabniško_ime**
- pomakni se v korenksi imenik (root)
 - **cd /**

Ogled vsebine delovnega imenika(izpis imenikov in datotek)

- pomakni se v imenik z vašim uporabniškim imenom in izpiši vse imenike in datoteke
 - **cd /home/uporabnik/**
 - **ls -a ali ls --all**
- podrobno izpiši imenike in datoteke (tako, da vidiš lastnika imenika in datoteke)
 - **ls -l** podrobni izpis map
 - **ls -all** še bolj podrobno
- podrobno izpiši imenike in datoteke (tako, da vidiš ID lastnika in grupe imenika in datoteke)
 - **ls -n**
- podrobno izpiši imenike in datoteke (tako, da NE vidiš lastnika imenika in datoteke)
 - **ls -g**
- podrobno izpiši imenike in datoteke (tako, da NE vidiš grupe imenika in datoteke)

- **ls -o**
- podrobno izpiši imenike in datoteke (tako, da NE vidiš grupe in lastnika imenika in datoteke)
 - **ls -o -g**
- izpiši imenike in datoteke tako da je viden tip
 - **ls -F**
- izpiši datoteke in imenike tako, da vidiš njihovo velikost
 - **ls -s**
- Izpiši vse datoteke tako, da bo vsaka datoteka v svoji vrstici
 - **ls -l**
- Premakni se v imenik **/var/lib/dpkg/info** in izpiši vse datoteke, ki se začno s črko a (**Ubuntu**)
 - **ls a* /var/lib/dict ls e***
- Izpiši vse datoteke v imeniku **/usr/share/doc/netcat/examples/data/**, ki se končajo s končnico c
 - **ls /usr/share/doc/netcat/examples/data/*.c** (**Ubuntu**)
 - **ls /usr/share/doc/manual/opensuse-manual_en/manual** (**Open Suse**)
 - ***.css *.html**
 - ali
 - **cd /usr/share/doc/netcat/examples/data/ in ls *.c** (**Ubuntu**)
- Izpiši vse datoteke iz imenika **/sbin/**, ki se začno s črko f, končnica datoteke pa se konča s končnico ext in poljubnim znakom
 - **ls /sbin/f*.ext?** (**Open Suse**)

Brisanje (praznega) imenika

- Izbriši imenik z imenom apj
 - **rmdir apj**
- Iz podimenika vaje_linux izbriši vse tri podimenike (ena, dva tri) hkrati
 - **rmdir ena dva tri**
- Pomakni se v domači imenik in izbriši imenik teksti (absolutno)
 - **rmdir /home/uporabnik/Documents/vaje_linux/teksti/**

Ustvarjanje datoteke

- V imeniku pra ustvari datoteko test.txt in vanjo vpiši to je moj prvi document s preprostim urejevalnikom (*vim*)
 - če še nismo v pravem imeniku **cd /home/uporabnik/Documents/vaje_linux/pa/**
 - **touch test.txt**
 - **vim test.txt**
 - vpišemo besedilo (tipka INSERT), ko vpišemo besedilo tipka ESC
 - shranjevanje in izhod **:wq ENTER**
 - **V primeru, da nismo v načinu za editiranje**
 - **x** **pobriše znak pod kurzorjem**
 - **dw** **pobriše vse znake desno od kurzorja(do presledka) vključno z znakom kjer je kurzor**
 - **dd** **pobriše vrstico pod kurzorjem**
 - **db** **pobriše vrstico pred kurzorjem**
- Ustvari datoteko zagon.txt v mapi vaje_linux(absolutno)

- **touch /home/uporabnik/Documents/vaje_linux/zagon.txt**
- V mapi pra ustvari tri datoteke (ubuntu.html, stevilo.pas in dijaki.sql) hkrati.
 - Z urejevalnikom VI izdelaj:
 1. html stran s sledečo vsebino
 - *To je prva html stran izdelana z urejevalnikom VI v OS Ubuntu*

```
<html>
<body>
To je prva html stran izdelana z urejevalnikom VI v OS Ubuntu
</body>
</html>
```
 - **Zagon html datoteke**
 - *firefox ubuntu.html*
 - *konqueror ubuntu.html*

2. Pascalsko datoteko (program), ki izpiše prvih sto naravnih števil
Ubuntu

V primeru da nimamo nameščenega prevajalnika za Pascal **sudo apt-get install gpc** ali **Administracija->Upravitelj paketov Synaptic** poiščemo **gpc** in ga namestimo
Program prevedemo s sledečim ukazom **gpc stevilo.pas -o stevilo.exe**
Prevedeno datoteko poženemo z ukazom **./test.exe**

Suse

```
su
tar xvf fpc-2.2.0.i386-linux.tar
sh install.sh
```

```
Program stevilo;
Var i:integer;
begin
for i:=1 to 100 do
writeln(i);
end.
```

3. SQL skripto, ki izpiše vse podatke iz tabele dijaki

```
SELECT *
FROM DIJAKI;
```

- **touch ena.html dva.txt tri.c**

Pregled vsebine datoteke s pomočjo ukaza cat

- cat stevilo.pas

Urejanje(sortiranje) datoteke

- sort primer.txt
 - shranjevanje sortirane vsebine
 - sort primer.txt>sortiran_primer.txt
 - Prikaz določene vrstice z določeno vsebino
- grep besedilo primer.txt
- Prikaz določene vrstice z določeno vsebino (ignoriramo Case Sensitive)
- grep -i Besedilo primer.txt
- Poišči konkreten primer
- cat /home/... | grep path

Združevanje datotek

Napiši ukaz, ki združi datoreki vaja1.txt in vaja2.txt v datoteko združeno.txt

- cat vaja1.txt vaja2.txt>zdruzeno.txt
- Počisti vsebino zaslona
 - clear

- Dodajanje vsebine tekstovni datoteki
 - dodajanje na začetek
 - echo „prva vrstica“>primer.txt
 - dodajanje na konec
 - echo „druga vrstica“>>primer.txt
- Vsebino imenika XYZ zapiši v datoteko `directorylisting.txt`
 - `ls -l > directorylisting.txt`
- Vsebino imenika ZZZ dodaj v datoteko `directorylisting.txt`
 - `ls -l >> directorylisting.txt`
- Napiši ukaz, ki izhod ukaza `ls` preusmeri v datoteko `file_list.txt`, rezultatov ne izpiše na ekran.
 - `ls > file_list.txt`
- Napiši ukaz, ki urejeno (sortirano) vsebino datoteke `file_list.txt` izpiše na zaslon.
 - `sort < file_list.txt`
- Napiši ukaz, ki urejeno (sortirano) vsebino datoteke `file_list.txt` izpiše na zaslon in zapiše v datoteko `sorted_file_list.txt`
 - `sort < file_list.txt > sorted_file_list.txt`

Namesto zaporedja ukazov

- `ls >> file_list.txt`
- `sort < file_list.txt > sorted_file_list.txt`

lahko uporabimo cev in izhod `ls` preusmerimo v vhod `sort` brez 'vmesne' datoteke `file_list.txt`:

- `ls | sort > sorted_file_list.txt`

Zelo uporabno za izpisovanje vsebine direktorija z mnogo datotekami je naslednje:

- `ls -l | less`

V tem primeru rezultat `ls` preusmerimo v `less`. Z uporabo "`| less`" lahko pregledujemo rezultate vseh ukazov.

Ukazi programa less

<i>Ukaz</i>	<i>Rezultat</i>
Page Up ali b	Pomik za eno stran navzgor
Page Down ali presledek	Pomik za eno stran navzdol
G	Pojdi na konec datoteke
1G	Pojdi na začetek datoteke

<code>/characters</code>	Išči prvo naslednjo pojavitev niza, določenega s <code>characters</code>
<code>n</code>	Ponovi iskanje
<code>q</code>	Izhod iz less

izpis vseh modulov, ki so vključeni v jedro OS

- `lsmod`

izpiši vse vrstice, ki vsebujejo `pcnet` (modul, ki podpira delovanje omrežne kartice)

- `lsmod | grep pcnet` **(Ubuntu)**

konfiguracija mrežne kartice (seznam in konfiguracija obstoječih omrežnih vmesnikov), mrežna kartica `eth0`, lo povratni vmesnik `loopback` – ne omogoča komuniciranja z zunanjim svetom

- `ifconfig` (predpogoj je, da se prijavite kot super user (`su`))

Odstranitev modula `pcnet` z jedra

- `rmmod pcnet32` **(Ubuntu)**
- `ifconfig` – vidimo, da dostop iz tega računalnika ni več mogoč

Vključitev modula `pcnet` v jedro OS

- `modprobe pcnet32` **(Ubuntu)**
- `ifconfig`

izpiši količino prostega in glavnega in izmenjalnega pomnilnika

- `free -m` (-m pomeni v količini megabyte)
- `free -k` je enako kot samo ukaz `free`

Kako dolgo teče sistem neprekinjeno

- `uptime`

izpiši podatke o procesih

- `ps`
- `ps -aux`
 - `a` (vse procese)
 - `u` (informacijo o lastniku procesa)
 - `x` (proces, ki niso bili zagnani iz terminala)
- `ps -aux | grep vi`
- `kill pid` primer: `kill 21570`
Spremljanje dogajanja v časovnih intervalih
- `top`
 - `q` izhod

Brisanje datoteke

- Izbrisi datoteko `test.txt`
 - `rm test.txt`
- Izbrisi datoteko `zagon.txt` (absolutno)
 - `rm /home/uporabnik/Documents/vaje_linux/zagon.txt`
- Izbrisi datoteko `ena.html` iz imenika `pra` tako, da moraš potrditi brisanje
 - `rm -i ena.html`
- zbrisi datoteko `dva.pas` iz imenika `pra` tako, da ti pri izbrisu izpiše katero datoteko si izbrisal

- `rm -v dva.pas`
- Izbriši imenik `pra` in vso vsebino imenika
 - `rm -r pra/`
 - `rm -fr pra/`

Izdelava kopije **cp** *izvor ponor* (izdelava kopije, kjer je *izvor* ime izvorne datoteke, *ponor* ime nove kopije)

- V imeniku `vaje_linux` ustvari datoteko `vsebina.txt` in vanjo vpiši datum svojega rojstva, vsebino te datoteke prekopiraj v novo datoteko z imenom `vsebina2.txt`
 - `touch vsebina.txt`
 - `cp vsebina.txt vsebina2.txt`
- V mapi `vaje_linux` naredi imenik (`mat`). Datoteko `vsebina.txt` prekopiraj v imenik `mat`.
 - `cp vsebina.txt mat`
- Kopiraj datoteko(absolutno) `vsebina.txt` v svoj v delovni imenik
 - `cp vsebina.txt /home/uporabnik`
- Kopiraj imenik `vaje_linux` s celotno vsebino v svoj domači imenik
 - dostopi do imenika `Documents`
 - `cp -r vaje_linux/ /home/uporabnik`
- Kopiraj celoten imenik
 - **`cp -R mapa3 mapa1/`**

Orodje za arhiviranje datotek (**tar** - orodje za arhiviranje datotek)
tar [opcije] arhiv datoteke

opcije:

`-czvf` kreira nov kompresiran arhiv, ki vsebuje navedene datoteke

Primer: `tar -czvf arhiv.tar *`

`-xzvf` razširi kompresiran arhiv v imenik, ki ga določimo

`-cvf` kreira nov arhiv, ki vsebuje navedene datoteke

`-xvf` razširi arhiv v imenik, ki ga določimo

Primer: `tar -xvf fpc-2.2.0.i386-linux.tar`

Premikanje ali preimenovanje *izvora v ponor*

mv *izvor ponor*

- Preimenuj datoteko `vsebina.txt` v `roj_datum.txt`
 - **`mv vsebina.txt roj_datum.txt`**
- Premakni datoteko `roj_datum.txt` v imenik `Documents`
 - **`mv roj_datum.txt /home/uporabnik/Documents/`**
- Premakni mapo `vaje_linux` v svoj domači imenik
 - **`mv -r vaje_linux/ /home/uporabnik/`**

Razno

- ugotovi vrsto datoteke(poglej v kompresirane datoteke)
 - **`file -z datoteka`**
- izpiši vrsto datoteke
 - **`file -b datoteka`**
- prikaži polno pot ukaza

- **which ime_programa**
- konvertiraj in kopiraj datoteko
 - **dd medij datoteka**
- poišči datoteko z imenom log v imenikih in podimenikih
 - **find -name log -iname ("izključimo" Case Sensitive)**
- preveri količino praznega prostora na disku uporabimo
 - **df -h**
- oceni porabljeno mesto na disku (prikaži vsoto)
 - **du -c datoteka**
- oceni porabljeno mesto na disku (pretvori v drugi format)
 - **du -h datoteka**
- oceni porabljeno mesto na disku (prikaži v kilobytes)
 - **du -k datoteka**
 - k prikaže v kilobytes
 - m prikaže v megabytes
- poišči vse izvršne datoteke
 - **whereis -b**
- poišči vso "dokumentacijo"
 - **whereis -m**
- poišči vse izvorne kode
 - **whereis -s**
- poišči vse nenavadne vnose
 - **whereis -u**

Primer:

poišči vse datoteke v /usr/bin ki niso dokumentirane v /usr/man/man1 ampak imajo izvorno kodo v /usr/src

- **whereis -u**
-
- cd /usr/bin
- whereis -u -M /usr/man/man1 -S /usr/src -f *

whereis -m -u * asks for those files
in the current directory that have no documentation.

- poišči podrobnosti o ukazu whereis
 - man whereis
 - izhod iz man-a (help-a) - >q (Q)
 - več pomoči črka h

Prikaz elektronskega priročnika z navodili za vpisani *ukaz* (**man ukaz**)

- elektronski priročnik za ukaz ls
 - **man ls**
- elektronski priročnik za ukaz man
 - **man man**

Prikaži vsebino *datoteke* (**cat datoteka**)

- izpiši vsebine datoteke pismo.txt

- **cat pismo.txt**
- izpiši vsebino datoteke passwd (absolutno)
 - **cat /etc/passwd**

prikaz vsebine *datoteke*, izpisan po dolžini enega zaslona naenkrat (**more datoteka**)

- izpis vsebine datoteke pomoc
 - more pomoc
- izpiši vsebino datoteke stdio.h (absolutno)
 - **more /lib/stdio.h**
- **"preslednica"**
 - naprej
- **"b"**
 - nazaj
- **"q"**
 - konec
- Dostopi do skupnega omrežnega prostora na strežniku atena in preglej, če je tam slika z imenom igor.jpg
 - smb://atena/skupno
- Kreiranje uporabnikov
 - Prijavi se kot administrator (root) in kreiraj 3 nove uporabnike z up. imenom in geslom tone, tine in maja
 - **useradd -s/bin/bash -ptest -d/home/test -m test**
 - **-d** home directory
 - **-s** starting program (shell)
 - **-p** password
 - **-m** (Create the user's home directory)

Primer:

- **useradd -s/bin/bash -d/home/tone -m tone**
 - **sudo passwd tone**
- **useradd -s/bin/bash -d/home/tine -m tine**
 - **sudo passwd tine**
- **useradd -s/bin/bash -d/home/maja -m maja**
 - **sudo passwd maja**
- Spremeni gesla vsem trem prej narejenim uporabnikom
 - **passwd tone**
 - v primeru, da smo prijavljeni kot root
 - Vnos obstoječega gesla (lahko je rogerjevo ali root geslo)
 - Vnos novega gesla
 - Potrditev novega gesla
 - spreminjanje uporabniškega imena
 - usermod¹
 - usermod -l novo_ime staro_ime
 - Primer: usermod -l reks rex
- V primeru, da si prijavljen kot uporabnik tone, se prijavi kot uporabnik tine in obratno

¹ prijavljen moraš biti kot administrator

- **su**
 - vpiši root geslo, ko si prijavljen kot root, potem **su -tine**
- Izbriši toneta in tineta, tako, da izbrišeš tudi njuna domača imenika (prijavljeni morate biti kot root)
 - **userdel -r tone**
 - **userdel -r tine**
- Izbriši uporabnika maja, tako, da izbrišeš uporabnika nato pa še njegov domači imenik
 - **userdel maja**
 - **rm -r /home/maja**
- Odjavi se iz konzole
 - **exit**

Opravila in procesi. Zaradi večopravnega sistema, ki je vgrajen v OS Linux, je možno preko ukazne lupine nadzirati tudi opravila, ki tečejo v sistemu (job control). V ta namen obstaja ukaz:

- Izpiši vsa opravila, ki smo jih zagnali
 - **jobs**

Opravila pa so lahko tudi zelo kompleksni ukazi, ki lahko med svojim izvajanjem generirajo več procesov.

- Preglej vse procese, ki smo jih pognali (ki trenutno tečejo)
 - **ps**
 - Izpisani procesi pa so le del celotne zaposlenosti OS
- izpiše vse tekoče procese (vse procese, ki tečejo v OS)
 - **ps -aux**

Pri izpisu lahko opazimo, da je vsakemu procesu dodeljena svoja identifikacijska številka (prvi stolpec izpisa), ki je označena s PID (Proces IDentification). Ko proces teče, se operacijski sistem nanj sklicuje ravno preko te številke. Lahko pa se zgodi, da želimo katerega izmed procesov odstraniti iz vrste. V takem primeru moramo kot uporabnik zahtevati od OS, da proces izloči. Za izločanje procesov moramo najprej imeti dovoljenje, nato pa poznati tudi PID procesa. Načeloma imajo vsi uporabniki dovoljenje za izločanje svojih procesov, PID pa lahko dobijo z uporabo prejšnjega ukaza. Izločitev procesa pa izvedemo z uporabo ukaza: **kill 14325** - kjer je 14325 PID procesa, ki ga želimo odstraniti. Pri izločanju procesov moramo paziti na stanje procesa. Izločiti je namreč mogoče le procese, ki tečejo, ne pa tudi ustavljene procese.

- Izloči poljubni process
 - **kill 14325**

Sprememba lastništva datotek in imenikov (**Uporabniki in dovoljenja datotek**)

Zaradi večuporabnosti ponuja Linux mehanizem ščitenja datotek, ki je znan kot **dovoljenja datotek**. S tem omejuje dostopanje do posameznih datotek različnim uporabnikom. Uporabnike razdeljuje v tri "koše":

- 1 lastnik datotek,
- 2 skupina, ki ji pripada lastnik datotek,
- 3 ostali uporabniki.

Pregled dovoljenj do datotek nam omogoča ukaz **ls -l**, ki nam prikaže vsebino imenika v 8 stolpičih. V prvem so predstavljena dovoljenja, v drugem število povezav (links), v tretjem je izpisan lastnik, v četrtem lastnikova skupina, v petem, šestem in sedmem dolžina, datum in ura nastanka datoteke in v osmi koloni ime datoteke. Primer takega izpisa prikazuje spodnji izpis:

```
-rw-r--r-- 1 kralj_1 users 322 Mar 22 12:35 moj_spis
```

Dovoljenja dostopanja do datotek v prvem stolpcu so razporejena po vrsti:

- znak - predstavlja tip datotek
- naslednji trije znaki - dostop za lastnika
- naslednji trije znaki - dostop za lastnikovo skupino
- naslednji trije znaki - dostop za ostale

Kaj pomenijo posamezne črke?

V dovoljenjih za dostopanje so lahko prisotni le r, w in x ki pomenijo:

- r - dovoljenje branja
- w - dovoljenje pisanja

- x - dovoljenje izvajanja

Spremembe dovoljenj je možno spreminjati z ukazom `chmod`, ki mu pripišemo binarni ekvivalent dovoljenj.

`chmod 764` seminarska_naloga

Pomen:

7 - lastnik lahko bere, piše in izvaja datoteko,

6 - skupina lahko bere in piše v datoteko,

4 - ostali lahko datoteko le preberejo.

Binarno	Desetiško
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Primer: `chmod 754 info.sh`

rwX	r-X	r--
111	101	100
7	5	4

Primer: `chmod ugoa+-rwx file`

prvi parameter določa komu (u)ser, (g)roup, (o)thers, (a)ll želimo (+)dodati ali odvzeti (-) pravice (r)ead, (w)rite, e(x)ecute. V prvem delu navedemo eno ali več črk, ki določajo komu bomo pravice spreminjali, sledi +/- za dodajanje oziroma odvzemanje pravic in ena ali več črk, ki določijo pravice, ki jih bomo spremenili.

Primer: `chmod ug+x file`

pomeni, da smo u-uporabniku in g-skupini dali pravico do izvajanja datoteke file. Pravice za lastnika, skupino in vse ostale lahko podamo tudi numerično.

Pravice si predstavljamo kot zaporedje bitov (tako jih vidi tudi sistem):

```

rwx rwx rwx = 111 111 111
rw- rw- rw- = 110 110 110
rwx --- --- = 111 000 000
    
```

itd...

```

rwx = 111 dvojiško = 7
rw- = 110 dvojiško = 6
r-x = 101 dvojiško = 5
r-- = 100 dvojiško = 4

```

Če si pravice za vsakega (lastnik, skupina, in ostale) predstavljamo kot eno cifro, je to zelo prikladen način za določanje pravic. Če želimo datoteki `file` določiti pravice read in write za lastnika, ostalim pa ne želimo omogočiti dostopa do datoteke, to storimo tako:

```
[me@linuxbox me]$ chmod 600 file
```

CHOWN - spremeni lastnika datoteke in skupine

chown lastnik datoteka

CHGRP - spremeni lastništvo skupine

chgrp skupina datoteka

Zaščita datotek

Vsaka datoteka in direktorij v Unixu ima attribute, ki določajo pravice, ki jih imajo nad datoteko oziroma direktorijem lastnik datoteke, skupina, kateri datoteka pripada in vsi ostali. Vsakemu od naštetih določimo pravico do branja datoteke, do pisanje v datoteko in pravico do izvajanja (poganjanja kot program) datoteke.

Pravice za posamezno datoteko in direktorij si lahko ogledamo z ukazom `ls -l`:

```

[me@linuxbox me]$ ls -l some_file

-rw-rw-r-- 1 me  me  1097374 Sep 26 18:48 some_file

```

Iz izpisa lahko razberemo naslednje:

1. Lastnik datoteke "some_file" je uporabnik "me"
2. Uporabnik "me" lahko datoteko bere in v njo piše
3. Datoteka pripada skupini "me"
4. Vsi člani skupine "me" prav tako lahko berejo in pišejo datoteko
5. Vsi ostali lahko datoteko samo berejo

Podobno si lahko ogledamo še datoteko `bash`:

```

[me@linuxbox me]$ ls -l /bin/bash

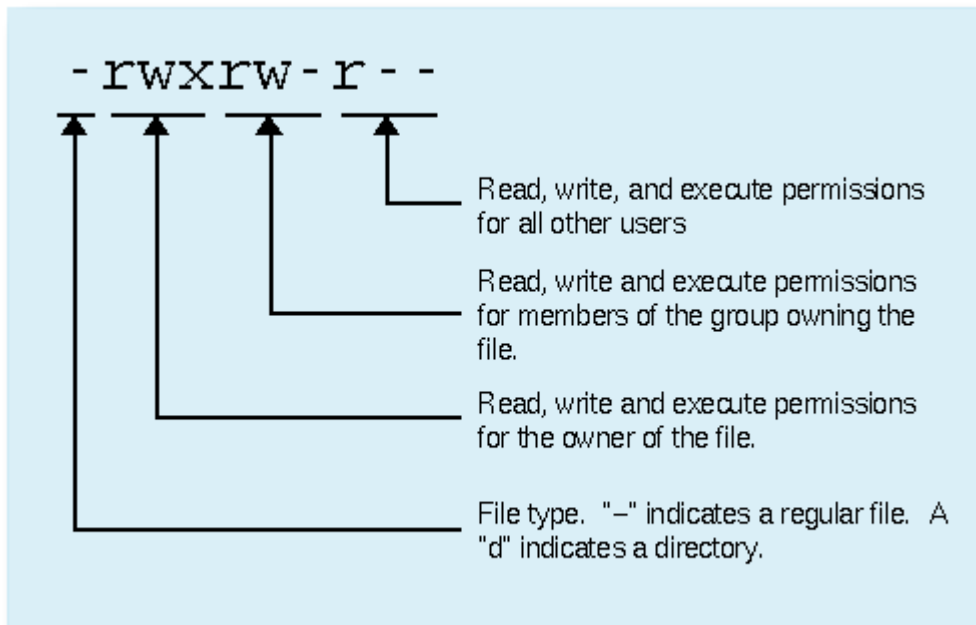
-rwxr-xr-x 1 root root  316848 Feb 27  2000 /bin/bash

```

Razberemo lahko naslednje:

- Lastnik `"/bin/bash"` je uporabnik `"root"` – 'superuser'
- `"root"` lahko datoteko bere, vanjo piše in jo izvaja (poganja)
- Datoteka pripada skupini `"root"`
- Člani skupine `"root"` lahko datoteko berejo, vanjo pišejo in jo izvajajo
- Vsi ostali lahko datoteko berejo in jo izvajajo

Na sliki je prikazano, kako se interpretira izpis s pravicami do datoteke. V izpisu je znak, ki določa tip datoteke, sledijo pa mu tri skupine znakov (r)ead (w)rite in e(x)ecute, ki določajo pravice do branja pisanja in izvajanja datoteke za lastnika, skupino in vse ostale.



Zaščita direktorijev

Direktorij je v resnici samo posebna vrsta datoteke, zato nastavitve pravic delujejo na enak način. To ne velja samo za pravico do izvajanja, ki določa pravico do izpisovanja direktorija. Kot primer je naštetih nekaj pogostih nastavitvev pravic za direktorije:

Vrednot	Pomen
777	<code>(rwxrwxrwx)</code> Ni omejitev. Vsak lahko izpiše vsebino, ustvarja nove datoteke in briše datoteke v direktoriju.
755	<code>(rwxr-xr-x)</code> Lastnik direktorija ima neomejene pravice. Vsi ostali lahko izpisujejo vsebino direktorija, vendar ne morejo ustvarjati novih datotek v njem oziroma jih brisati. To je običajna nastavitvev za direktorije, ki jih želimo deliti z drugimi uporabniki.
700	<code>(rwx-----)</code> Lastnik direktorija ima neomejene pravice. Vsi ostali nimajo nobenih pravic. Uporabno za direktorije, ki jih želimo skriti pred ostalimi

Kako začasno postati 'superuser'

Običajno smo iz varnostnih razlogov v sistem prijavljeni kot navaden uporabnik brez administratorskih pravic. Včasih imamo zato premalo pravic, da bi opravili določeno administracijo in želimo začasno postati 'superuser', da lahko administracijo opravimo. To naredim z ukazom `su` (superuser). Vnesti moramo samo še ustrezno geslo.

```
[me@linuxbox me]$ su
Password:
[root@linuxbox me]#
```

P izvedbi ukaza `su`, smo prijavljeni kot 'superuser'. Ko končamo z opravili, ki jih moramo opraviti kot 'superuser' sejo zaključimo z ukazom `exit` in zopet smo prijavljeni kot navadni uporabnik.

Sprememba lastnika datoteke

Lastnika datoteka je mogoče zamenjati z ukazom `chown`. Primer: lastnik datoteke `some_file` je uporabnik "me", želimo, da lastnik te datoteke postane uporabnik "you":

```
[me@linuxbox me]$ su
Password:
[root@linuxbox me]# chown you some_file
[root@linuxbox me]# exit
[me@linuxbox me]$
```

Za spremembo lastnika datoteke moramo biti prijavljeni kot 'superuser'. `chown` na enak način kot z datotekami deluje tudi z direktoriji.

Sprememba skupine, kateri pripada datoteka

Skupino, ki je lastnik določene datoteke, zamenjamo z ukazom `chgrp`. Ukaz uporabljamo podobno kot `chown`:

```
[me@linuxbox me]$ chgrp new_group some_file
```

Za izvajanje `chgrp` moramo biti lastnik datoteke oziroma direktorija.

Zaustavitev sistema

SHUTDOWN - pripravi računalnik na izklop

```
shutdown [opcije] čas [sporočilo]
```

opcije:

-h ustavi sistem

-n hitro ustavi sistem

- r ponovno zaženi sistem
- f hitro ponovno zaženi sistem
- c prekliči ustavitev sistema

čas:

podamo čas ali napišemo `now` za takoj

sporočilo:

sporočilo, ki je poslano vsem uporabnikom pred izklopom

HALT - pripravi računalnik na izklop

`halt`

REBOOT - pripravi računalnik na ponovni zagon

`reboot`

POWEROFF - pripravi računalnik na izklop

`poweroff`