

# Osnove korištenja operacijskog sustava Linux

## 01. Uvod

Antun Aleksa, Josip Žuljević  
Nositelj: dr. sc. Stjepan Groš

Sveučilište u Zagrebu  
Fakultet elektrotehnike i računarstva

15.10.2016

# Sadržaj

# Operacijski sustav (1)

- ▶ Operacijski sustav ima nekoliko primarnih zadaća:
  - Pristup uređajima
    - Kontrolira svaki zahtjev za pristup uređajima
    - Određuje koji zahtjev ima prioritet
  - Upravljanje korisnicima
    - Razlikuje zahtjeve korisnika
- ▶ Ovakav opis skriva puno detalja, ali definira dvije najbitnije uloge operacijskog sustava

## Operacijski sustav (2)

- ▶ Sklopovlje – mnoštvo kompleksnih uređaja
  - Pisanje aplikacija za samo jedan je komplicirano
- ▶ OS preuzima detalje
  - Korisnik (u teoriji) treba znati samo što želi
  - OS zna kako pristupiti određenom uređaju
- ▶ Primjer: pisanje podataka na tvrdi disk
  - Korisnik/aplikacija uputi zahtjev za brisanje datoteke
  - OS primi zahtjev i dalje odlučuje što sa njime

## Operacijski sustav (3)

- ▶ OS je posrednik između aplikacije i sklopovlja
  - korisnik → aplikacija → OS → uređaj
- ▶ Aplikacijama nikada nije dopušteno izravno pristupanje uređajima
  - Moglo bi doći do kolizije
    - Tome služi operacijski sustav
  - Primjer iznimke je DOS

# Osnovni pojmovi (1)

- ▶ Kernel – jezgra sustava
  - Ono što nazivamo Linux je jezgra
    - Linux u širem smislu je jezgra + aplikacije
- ▶ Sklopovska podrška izvršavanju OS-a
  - Nadzorni način rada, MMU, ...

# Unix porodica operacijskih sustava (1)

- ▶ Većina se grubo može podijeliti u dvije skupine
  - Windows bazirani
  - Unix bazirani
- ▶ Suprotno očekivanjima, većina su Unix bazirani
- ▶ Unix – prvi višekorisnički sustav
  - Nastao 1970-ih u Bell Labs laboratoriju
  - Prvotno potpuno besplatan

## Unix porodica operacijskih sustava (2)

- ▶ Nastaju razne inačice
  - AIX, Solaris (OpenSolaris), HP-UX, ...
- ▶ 1983. Unix je komercijaliziran
- ▶ Dvije glavne grupe
  - BSD verzije
  - System V Release 4 verzije
- ▶ Linux **nije** Unix, Unix je zaštićeni znak
  - Spada u grupu tzv. unixoida (Linux, \*BSD, ...)



# Razvoj Linuxa (1)

- ▶ Distribuirani razvoj
- ▶ Tisuće programera diljem svijeta
  - Velik broj nezavisnih, gledano pojedinačno
  - Još veći pod sponzorstvom
    - Google, IBM, Oracle, Intel, ...
  - Pišu se moduli, zakrpe, dokumentacija
  - Izmjene se predlažu odgovornim osobama
  - Cijeli kernel mozete vidjeti na GitHub-u ([link](#))

## Razvoj Linuxa (2)

- ▶ Linux je podijeljen na podsustave
- ▶ Svaki podsustav ima tzv. “održavatelja” (engl. *maintainer*)
  - Održavatelj odlučuje o izmjenama (više-manje)
- ▶ Linus Torvalds ima najveći autoritet
  - Izmjene danas rijetko idu izravno preko njega
- ▶ Značajni održavatelji
  - Andrew Morton
  - Greg Kroah-Hartman

# Razvoj Linuxa (3)

- ▶ Model razvoja nekada
  - Parne verzije stabilne, neparne razvojne
    - 1.0, 1.2, 2.0, 2.2, 2.4 stabilne verzije
    - 1.1, 1.3, 2.1, 2.3, 2.5 nestabilne verzije
- ▶ `www.distrowatch.com`

# Licence (1)

- ▶ Licence (EULA – end-user licence agreement) su vrlo bitne jer određuju prava i obaveze korisnika
- ▶ Niz različitih tipova licenci
  - Komercijalne, “shareware”, otvoreni kod, ...
- ▶ Licence otvorenog koda su bitne za razvoj Linuxa
- ▶ GPL (1,2,3), LGPL, Apache, BSD

# Distribucije (1)

- ▶ Labavo definirano, distribucija je Linux kernel + skup programa
- ▶ Većina operacijskih sustava i njihovih alata dolaze u kompletu
  - 1 izdavač – 1 operacijski sustav sa 1 “distribucijom”
    - Microsoft/Windows, Apple/Mac OS X, FreeBSD
- ▶ Linux ima stotine distribucija
  - specijalizirane ili opće namjene

## Distribucije (2)

- ▶ Tri najveće grane distribucija
  - Debian, Red Hat, Slackware
- ▶ Najočitije razlike su grafičko sučelje i instalacijski sustav
  - GNOME, KDE, Xfce, Awesome, ...
  - apt, yum, pacman, ...
- ▶ Distribucije su konačni proizvodi, operacijski sustavi u najširem smislu

# Terminal

- ▶ “Uređaj” koji prima znakove i prikazuje ispis
  - Nekada su terminali bili fizički uređaji
  - Danas su aplikacije koje oponašaju fizičke terminale
- ▶ Postoji nekoliko emulatora terminala na Unix sustavima
  - xterm, rxvt, gnome-terminal, ...
- ▶ Terminali upravljaju unosom i ispisom znakova
- ▶ Ljuska interpretira značenje znakova

# Komunikacija sa računalom (1)

- ▶ Dva temeljna načina komunikacije
  - Kroz grafičko sučelje i putem ljuske/komandne linije
- ▶ Ljuska je aplikacija(!) koja prihvća korisnikove naredbe i izvršava ih
  - Ljuska olakšava komunikaciju sa sustavom
- ▶ Ljuska označava spremnost za prihvćanje naredbi prikazivanjem naredbenog retka (engl. *command prompt*)



## Komunikacija sa računalom (2)

- ▶ Kada ljuska pokrene naredbu, čeka da se njeno izvršavanje završi
- ▶ Za to vrijeme ljuska ne prikazuje naredbenu liniju!
  - Moguće **prisilno** zaustaviti naredbu, Ctrl+C
- ▶ Primjer:
  - Pokrenuti naredbu cat

# Sustav pomoći

- ▶ Linux ima sustav pomoći
  - Naredba `man`
    - Dostupna na svakom Unixu
  - Naredba `info`
    - Dostupna sa GNU alatima
  - U direktoriju `/usr/share/doc` dosta materijala
- ▶ Sadrži opise i načine korištenja naredbi, funkcija i konfiguracijskih datoteka

# Naredba man (1)

- ▶ Korištenje naredbe man
  - Pregled neke upute
    - `man <ime naredbe>`
    - `man <sekcija> <ime naredbe>`
  - Pretraživanje stranica
    - `man -k <ključna riječ>`
    - `man -f <ime datoteke>`
- ▶ Primjer: Pregledavanje upute za naredbu man
  - `man man`

## Naredba man (2)

- ▶ Standardni dijelovi man stranice
  - NAME – ime i kratki opis
  - SYNOPSIS – mogući načini korištenja
  - DESCRIPTION – dulji opis što naredba radi
  - OPTIONS – opcije koje naredba prihvaća
  - ENVIRONMENT – varijable okruženja (o njima kasnije)
  - AUTHOR – autor stranice/naredbe
  - SEE ALSO – koje su vezane naredbe

## Naredba man (3)

- ▶ Opcionalni dijelovi se stavljaju unutar [ i ]
- ▶ Ponavljanje se označava sa ...
- ▶ Izlazak iz pregledavanja upute
  - malo slovo q

# Prikaz sadržaja direktorija

- ▶ Naredba `ls` (engl. *list*)
  - Prva akcija nakon pokretanja ljske na nepoznatom sustavu
  - bez parametara: popis svih datoteka i direktorija, poredan abecednim redom, odozgo prema dolje te s lijeva na desno
- ▶ **Linux razlikuje velika i mala slova**

# Naredbe (1)

- ▶ Naredbe se pokreću na sljedeći način  
    <naredba><opcije><arugmenti>
- ▶ Argumenti označavaju nad čime se vrši naredba
  - Često datoteke
  - Ponekad nisu potrebni
    - Primjer 1s
  - Sve piše u man stranicama

## Naredbe (2)

- ▶ Opcije utječu na ponašanje naredbe
  - duge opcije (engl. *long options*) počinju s "--"
  - kratke opcije (engl. *short options*) počinju s "-"
- ▶ Naredbe obično imaju različite opcije za isto ponašanje
  - Primjer: `man -k regex` **ili** `man --apropos regex`
- ▶ Duge opcije su često opisne zbog lakšeg razumijevanja



# Naredba ls (1)

- ▶ Naredba `ls` prihvaća niz opcija
- ▶ Često korištena opcije je `-l` (engl. *long*)
  - Ispisuje detalje o datotekama i direktorijima
- ▶ Primjer: Izvršiti sljedeću naredbu

```
$ ls -l /bin/sh
```

```
lrwxrwxrwx 1 root root 4 2010-04-29 10:44 /bin/sh -> bash
```

## Naredba ls (2)

- ▶ Stupci kod korištenja `ls` naredbe s `-l` opcijom
  - Tip datoteke, dozvole, broj referenci, vlasnik, grupa, veličina u oktetima, vrijeme zadnje promijene, ime datoteke
- ▶ Često `ls` implicitno dodaje opciju `--color`

## Naredba ls (3)

- ▶ Naredbi `ls` moguće je zadati i ime datoteke/direktorija

- niz odvojen prazninama (označen s ... u man)

- ▶ Primjer

```
$ ls -l /bin/sh
```

```
lrwxrwxrwx 1 root root 4 2010-04-29 10:44 /bin/sh -> bash
```

- ▶ Opcija `-h` (engl. *human readable*)

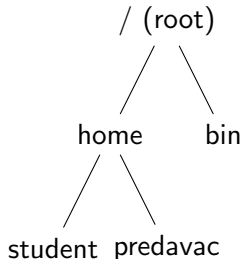
- U kombinaciji s `-l` ispisuje veličine u čitljivijem formatu

# Skrivene datoteke

- ▶ Skrivene datoteke su skrivene u svrhu ljepšeg/jednostavnijeg ispisa, te kako se ne bi slučajno oštetile nepažnjom korisnika
- ▶ Na Unix/Linux operacijskom sustavu datoteke čija imena započinju točkom su **skrivene datoteke**
  - Ne ispisuju se prilikom izvršavanja naredbe `ls`, osim ako to ne zatražimo

# Direktoriji

- ▶ Direktoriji su organizirani kao stablo
- ▶ U Unix operacijskom sustavu nema diskova
  - Sve je **jedno** stablo direktorija s **jednim** korijenom



# Trenutni direktorij

- ▶ Trenutni (tekući, radni) direktorij je direktorij u kojem se nalazimo
  - Korisnik ili aplikacija
- ▶ Možemo saznati koji je radni direktorij naredbom `pwd` (engl. *print working directory*)

# Matični direktorij

- ▶ Svaki korisnik ima matični (engl. *home*) direktoriji
  - Služi za spremanje osobnih podataka i postavki
  - Korisnik ima ovlasti za pisanje i brisanje unutar svog matičnog direktorija
  - Izvan matičnog direktorija **nije dozvoljeno** pisanje korisnicima koji nisu administratori
- ▶ Neposredno nakon pokretanja ljuške, trenutni direktorij je matični direktorij

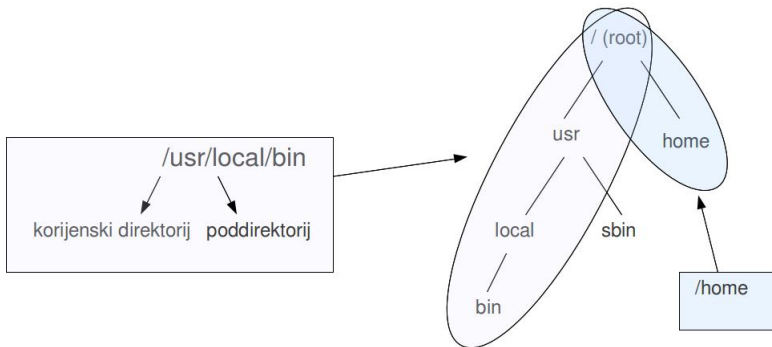
# Relativna i apsolutna staza (1)

- ▶ Položaj datoteke ili direktorija na Unix/Linux sustavu može se zadati
  - Apsolutno - počinje znakom "/"  
Primjer: `/home/student/okos1`
  - Relativno  
Primjer: `student/okos1`
- ▶ Relativna staza ovisi o trenutnom direktoriju
  - Može se zadati samo za direktorije koji su niže u hijerarhiji
  - Ako ste u direktoriju `/home`, apsolutna staza za **student/okos1** je **`/home/student/okos1`**



## Relativna i apsolutna staza(2)

- ▶ Znak “/” označava korijenski direktorij
  - Kod staze znači da je sljedeći direktorij poddirektorij prethodnog



# Stvaranje direktorija

- ▶ Stvaranje novog direktorija obavlja se naredbom `mkdir`
  - Argument je relativno ili apsolutno ime direktorija koji se stvara
- ▶ Naredba `mkdir` prihvaća opciju `-p`
  - Stvara sve potrebne poddirektorije ako ne postoje

# Prikaz sadržaja direktorija

- ▶ Naredba `ls` ispisuje sadržaj direktorija
- ▶ Opcijom `-d` ne izlistava se sadržaj direktorija već informacije o samom direktoriju

# Promjena direktorija (1)

- ▶ Promjena direktorija obavlja se naredbom `cd` (engl. *change directory*)
- ▶ Novi direktorij se zadaje
  - relativno u odnosu na tekući direktorij
  - apsolutno u odnosu na korijenski direktorij
- ▶ Naredba `cd` bez argumenata vraća u matični direktorij

# Posebni direktoriji (1)

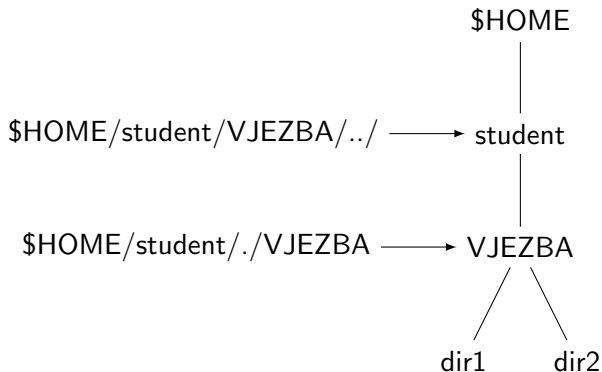
- ▶ Svaki direktorij sadrži dva posebna direktorija
  - .. roditeljski direktorij (engl. *parent directory*)
  - . tekući direktorij (engl. *current directory*)
- ▶ Primjeri
  - `cd .`
    - Mijenja direktorij u tekući direktorij, efektivno nema promjene
  - `cd ..`
    - Mijenja trenutni direktorij u direktorij iznad

## Posebni direktoriji (2)

- ▶ Koriste se za relativno *adresiranje* direktorija
- ▶ Ne mogu ići van korijenskog direktorija, tj.  
    `/../..` je isto što i `/`
- ▶ Česta pogreška korisnika DOS-a
  - Upisivanje `cd..` (bez razmaka)
  - Na Unixu to je posebna naredba (`.` može biti sastavni dio imena)

## Posebni direktoriji (3)

- Posebni direktoriji se mogu koristiti u imenima datoteka



# Skripte

- ▶ Skripta je niz naredbi zapisan u tekstualnu datoteku, a za čije izvršavanje nije potreban compiler
- ▶ Inače bi korisnik morao unostit svaku naredbu zasebno u ljsku
- ▶ Kratica za skriptu je .sh, te se pokreće naredbom `bash <ime_skripte>`



# Primjer skripte

- ▶ `#!/bin/bash`  
`mkdir radno`  
`echo pozdrav_ekipa`  
`ls -lh`

# Pregled naredbi

<code>man &lt;ime naredbe&gt;</code>	opis određene naredbe
<code>ls</code>	popis svih datoteka i direktorija
<code>pwd</code>	trenutni (radni) direktorij
<code>mkdir</code>	stvaranje novog direktorija
<code>cd</code>	promjena trenutnog direktorija
<code>bash &lt;ime skripte&gt;</code>	pokretanje skripte

# Upravljanje verzijama datoteke

- ▶ `git init`
- ▶ `git add .`
- ▶ `git commit -am 'initialized project'`
- ▶ `git push origin master`
- ▶ `log`, `reset`, `pull`, `fetch`, `merge`, `remote`, `checkout`
- ▶ privatni i javni ključevi
- ▶ Kratak reference na najbitnije naredbe u git-u

# Upravljanje paketima

- ▶ paket je program
- ▶ yum, apt, pacman, zypper ...
- ▶ Primjer naredbi za apt:
  - apt-get update
  - apt-get install git

# Literatura

- ▶ <http://www.troubleshooters.com/linux/info.htm>
- ▶ [http://www.schweikhardt.net/man\\_page\\_howto.html#q3](http://www.schweikhardt.net/man_page_howto.html#q3)
- ▶ <http://www.debian.org/doc/debian-policy/ch-docs.html>
- ▶ [http://www.unix.org/what\\_is\\_unix/history\\_timeline.html](http://www.unix.org/what_is_unix/history_timeline.html)
- ▶ <http://distrowatch.com/dwres.php?resource=major>