

Naučite

II izdanje

# Linux Shell SKRIPTOVANJE

Ganesh Naik

Iskoristite moć shell skriptova da biste rešili probleme u stvarnom svetu





Naučite

# Linux Shell SKRIPTOVANJE

**Ganesh Naik**

**II IZDANJE**

**Izdavač:**



**kompjuter  
biblioteka**

Obalskih radnika 4a, Beograd

**Tel: 011/2520272**

**e-mail:** kombib@gmail.com

**internet:** www.kombib.rs

**Urednik:** Mihailo J. Šolajić

**Za izdavača, direktor:**

Mihailo J. Šolajić

**Autor:** Ganesh Naik

**Prevod:** Slavica Prudkov

**Lektura:** Miloš Jevtović

**Slog:** Zvonko Aleksić

**Znak Kompjuter biblioteke:**

Miloš Milosavljević

**Štampa:** „Svetlost“, Čačak

**Tiraž:** 500

**Godina izdanja:** 2018.

**Broj knjige:** 505

**Izdanje:** Prvo

**ISBN:** 978-86-7310-528-4

# Learning Linux Shell Scripting Second Edition

Ganesh Naik

ISBN 978-1-78899-319-7

Copyright © 2018 Packt Publishing

All right reserved. No part of this book may be reproduced or transmitted in any form or by means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Autorizovani prevod sa engleskog jezika edicije u izdanju „Packt Publishing“, Copyright © 2018.

Sva prava zadržana. Nije dozvoljeno da nijedan deo ove knjige bude reprodukovan ili snimljen na bilo koji način ili bilo kojim sredstvom, elektronskim ili mehaničkim, uključujući fotokopiranje, snimanje ili drugi sistem presnimavanja informacija, bez dozvole izdavača.

**Zaštitni znaci**

Kompjuter Biblioteka i „Packt Publishing“ su pokušali da u ovoj knjizi razgraniče sve zaštitne oznake od opisnih termina, prateći stil isticanja oznaka velikim slovima.

Autor i izdavač su učinili velike napore u pripremi ove knjige, čiji je sadržaj zasnovan na poslednjem (dostupnom) izdanju softvera. Delovi rukopisa su možda zasnovani na predizdanju softvera dobijenog od strane proizvođača. Autor i izdavač ne daju nikakve garancije u pogledu kompletnosti ili tačnosti navoda iz ove knjige, niti prihvataju ikakvu odgovornost za performanse ili gubitke, odnosno oštećenja nastala kao direktna ili indirektna posledica korišćenja informacija iz ove knjige.

CIP - Каталогизација у публикацији  
Народна библиотека Србије, Београд,  
се добија на захтев

## O AUTORU

**Ganesh Naik** je autor, konsultant i korporativni trener za razvoj proizvoda koji se odnose na ugrađeni Android, ugrađeni Linux, IoT i mašinsko učenje. On ima više od 20 godina profesionalnog iskustva i mnoštvo završenih projekata u informacionim tehnologijama. Radio je kao korporativni trener za Indian Space Research Organization, “Intel”, GE, “Samsung”, “Motorola”, Penang Skills Development Center i razne kompanije u Singapuru i Indiji. Pokrenuo je kompaniju pod nazivom “Levana Technologies”, koja saraduje sa organizacijama za konsultacije i treninge.

*Zahvaljujem se mojoj supruzi Vishalakshi što mi je pružila vredne preporuke, podršku i neprekidnu motivaciju. Takođe se zahvaljujem mom kolegi Mansi Joshi, koji mi je obezbedio povratne informacije iz tehničke perspektive. Puno se zahvaljujem “Packtovom” timu u kome su bili Shrilekha Inani, Priyanka Deshpande i Prashant Chaudhari za pozitivnu i motivacionu podršku za pisanje ove knjige.*

## O RECENZENTU

Shawn Soloman je tehnološki veteran, sa širokim opsegom veština i sa ogromnim iskustvom na osnovu rada na polju tehnologije, dužeg od 20 godina. Iako je radio na poljima ISP, VoIP, obrazovanju, razvoju otvorenog koda i oporavku od otkaza sistema, skup njegovih veština se godinama prilagođavao i proširivao.

## „PACKT“ TRAŽI AUTORE KAO ŠTO STE VI

Ako ste zainteresovani da postanete autor za „Packt“, posetite stranicu [authors.packtpub.com](http://authors.packtpub.com) i prijavite se odmah. Mi smo radili sa hiljadama programera i profesionalaca da bismo im pomogli da podele svoj uvid sa globalnom tehničkom zajednicom. Možete da popunite osnovnu prijavu, da se prijavite za specifičnu temu za koje tražimo autore ili da nam pošaljete neku svoju ideju.

# UVOD

---

Shell skriptovi su osnovni deo svakog modernog operativnog sistema, kao što su Unix, Linux ili Windows. Jezik skriptovanja i njegova sintaksa mogu da variraju od OS-a do OS-a, ali osnovni principi ostaju isti. Ja sam se prvi put susreo sa Linux shell skriptovima u toku razvoja ugrađenih Linux proizvoda. Shell skriptovi su pokretali ceo proizvod, od osnovne procedure pokretanja sistema i evidencije korisnika, do potpunog pokretanja operativnog sistema. Još jedna situacija kada je potrebna upotreba skriptova je automatizacija regularnih aktivnosti, kao što je upravljanje verzijom i izdanjem izvornih kodova veoma složenih proizvoda, gde su 10.000 fajlova bili deo jednog projekta. Slično tome, još jedan veoma uobičajeni zahtev je automatizovanje administracije programa.

Prvo sam učio skriptove za rešavanje praktičnih problema i prilagođavanje već postojećih proizvoda. Ova knjiga je rezime onoga što sam naučio tokom godina o Linux shell skriptovanju kroz razvoj projekata, konsalting i korporativne treninge i sesije P&O.

U ovoj knjizi naučićete osnove shell skriptovanja za složene, prilagođene automatizacije. Zahvaljujući znanju stečenom u ovoj knjizi, čitalac će moći samouvereno da upotrebi sopstvene shell skriptove za rešavanje stvarnih problema. Ideja je da budemo što praktičniji i damo čitaocu uvid /kako izgleda skriptovanje u stvarnom svetu.

Ova knjiga obuhvata GNU Bourne Again Shell (BASH) skriptovanje. Možete da upotrebite znanje koje ste stekli čitanjem ove knjige za bilo koji shell bilo koje Unix ili Linux distribucije. Možda ćete morati da se pobrinete za nekoliko promena u sintaksi ako radite u drugim shellovima, kao što su Korn ili neki drugi. Možete da pročitate ovu knjigu od korice do korice ili samo nešto što vas interesuje. Međutim, ako imate pitanje kako da rešite neki određeni problem ili vam je potreban savet, veoma ćete lako pronaći odgovarajuće rešenje da biste uštedeli vreme i energiju. Poenta je u tome što je ova knjiga veoma posebna – knjiga u koju sam uložio celokupno svoje znanje. Međutim, analiza podataka može biti veoma komplikovana oblast i ponekad izgleda kao da ništa nema smisla. U takvim situacijama su se mnogi drugi (čak i ja) osećali kao da „stoje u mestu“. Nemojte odustajati, jer je „nagrada“ sjajna. Ako je „smetenjak“ kao što sam ja mogao da nauči analizu podataka, možete i vi. Samo napred!

## ZA KOGA JE OVA KNJIGA

Ova knjiga je namenjena čitaocima koji su vešti u radu na Linuxu i koji žele da nauče više o shell skriptovanju da bi poboljšali efikasnost i praktične veštine. Sledi nekoliko primera u kojima možete da primenite veštine koje se uče u ovoj knjizi:

- ▣ shell skriptovanje za automatizaciju zadataka, kao što je periodično kreiranje rezervne kopije
- ▣ administracija sistema
- ▣ održavanje baze podataka i kreiranje rezervne kopije
- ▣ obrada testa i generisanje izveštaja
- ▣ prilagođavanje pokretanja sistema
- ▣ razvoj ugrađenih Linux proizvoda

## ŠTA OBUHVATA OVA KNJIGA

U Poglavlju 1, “Početak rada i shell skriptovanje”, predstavljeni su različiti načini za pisanje i pokretanje shell skriptova. Takođe ćemo vam predstaviti načine za obradu fajlova i direktorijuma i upotrebu dozvola.

U Poglavlju 2, “Detaljnije o upravljanju procesima, kontroli posla i automatizaciji”, biće reči o osnovnom upravljanju procesima. Naučićete više o komandi ps i o upravljanju poslovima pomoću komandi kao što su jobs, fg, bg, kill i pkill. Kasnije ćete učiti o alatima za kontrolu procesa top, iostat, vmstat i sar.

U Poglavlju 3, “Upotreba obrade teksta i filtera u skriptovima”, upoznaćete upotrebu komandi more, less, head i tail. Takođe ćete učiti o alatima za obradu teksta, kao što su cut, paste, comm i uniq. Saznaćete šta su standardni ulaz, izlaz i standardna greška. Kasnije ćete učiti o meta karakterima i poklapanju obrazaca pomoću alatki VI i grep.

U Poglavlju 4, “Upotreba komandi”, saznaćete kako shell interpretira bilo koju komandu koja je uneta u komandnu liniju. Takođe ćete detaljnije učiti o zameni komande, separatorima i usmeravanjima.

U Poglavlju 5, “Istraživanje izraza i promenljivih”, biće reči o promenljivim uopšteno, a konkretno o promenljivim okruženja, što uključuje i način eksportovanja promenljivih okruženja, podešavanje, menjanje, promenljive read-only, argumente komandne linije i kreiranje i obradu nizova.

Poglavlje 6, “Dobri trikovi u shell skriptovanju”, posvećeno je ispravljanju grešaka, operatoru here i interaktivnim shell skriptovima za upotrebu unosa sa tastature i obradi fajla.

U Poglavlju 7, “Izvršavanje aritmetičkih operacija u shell skriptovima”, opisano je izvršavanje aritmetičkih operacija na različite načine, kao što je upotreba komandi declare,



let, expr i aritmetičkih izraza. Takođe ćete učiti o predstavljanju brojeva u različitim bazama, kao što su hex, octal i binary. Osim toga, u ovom poglavlju je opisana upotreba pomoćnog programa bc za izvršavanje aritmetičkih operacija sa decimalnim brojevima ili razlomcima.

Poglavlje 8, “Automatizacija donošenja odluka u skriptovima”, posvećeno je donošenju odluka u skriptovima upotrebom Testa i iskaza if-else i prebacivanjem slučaja. Takođe ćete naučiti kako da upotrebite selekciju pomoću petlje for, zajedno sa menijem.

U Poglavlju 9, “Automatizacija ponavljajućih zadataka”, govorimo o ponavljajućim zadacima, kao što je rutinska administracija pomoću petlje for, petlje while i petlje do while. Takođe ćete naučiti kako da kontrolišete petlje pomoću iskaza break i iskaza continue.

Poglavlje 10, “Upotreba funkcija”, posvećeno je funkcijama u shell skriptovima. Saznaćete kako da definišete i prikažete funkcije i kako se uklanja funkcija iz shella. Takođe ćete učiti o prosleđivanju argumenata u funkcije, deljenju podataka između funkcija, deklarisanju lokalnih promenljivih u funkciji, vraćanju rezultata iz funkcije i pokretanju funkcija u pozadini. Na kraju ćemo predstaviti upotrebu komandi source i .(dot). Koristićemo ove komande za upotrebu biblioteke funkcija.

U Poglavlju 11, “Upotreba naprednih funkcionalnosti u skriptovima”, tema je upotreba programskih klopki i signala. Takođe ćete učiti o kreiranju menija upotrebom pomoćnog programa dialog.

U Poglavlju 12, “Pokretanje sistema i prilagođavanje Linux sistema”, predstavimo pokretanje Linux sistema, od uključivanja, do prijave korisnika. Osim toga, naučićete kako da prilagodite okruženje Linux sistema.

Poglavlje 13, “Podudaranje obrazaca i regularnih izraza pomoću editora sed i awk”, posvećeno je regularnim izrazima i upotrebi editora sed (stream editor) i awk za obradu teksta. Naučićete kako da upotrebite različite komande i opcije, zajedno sa primerima za upotrebu editora sed i awk.

U Poglavlju 14, “Kreiranje rezervne kopije i ugrađivanje drugih jezika u shell skriptide”, govorimo o kreiranju rezervne kopije lokalno i preko mreže. Takođe ćete učiti o automatizaciji kreiranja rezervne kopije korišćenjem pomoćnog programa crontab. Ovladaćete ugrađivanjem drugih jezika u bash skriptide, kao što su Python, Ruby i Pearl.

U Poglavlju 15, “Administracija baze podataka pomoću shell skriptova”, saznaćete kako se pišu i izvršavaju MySQL komande u shell skriptu i kako se pišu i izvršavaju Oracle komande u shell skriptu. Zahvaljujući znanju stečenom u ovoj knjizi, moći ćemo da automatizujete često upotrebljavane zadatke administracije baze podataka.

## Da biste dobili maksimum iz ove knjige:

Svaki računar koji ima instaliran Linux OS biće dovoljan za učenje svih tema koje su opisane u ovoj knjizi. Za prvo izdanje smo koristili Ubuntu Linux distribuciju. Za ovo drugo izdanje upotrebili smo CentOS Linux distribuciju. Ja sam lično testirao sve komande i skripte u Ubuntu 16.04 i u CentOS 7.0 distribucijama.

Ako u toku obuke pronađete određeni pomoćni program koji nije instaliran u Ubuntu ili bilo koju drugu distribuciju koja je zasnovana na Debianu, unesite sledeću komandu da biste instalirali pomoćni program:

```
$ sudo apt-get update
$ sudo apt-get install package-name
```

Za pokretanje prethodnih komandi potrebna je dobra internet konekcija.

U CentOS ili bilo kojoj drugoj distribuciji koja je zasnovana na rpmu unesite sledeće komande:

```
$ sudo yum update
$ sudo yum install package-name
```

Ako imate internet konekciju, upotrebom ovih komandi možete da instalirate bilo koju komandu ili pomoćni program koji još nije instaliran.

## Preuzimanje fajlova primera koda

Fajlove sa primerima koda za ovu knjigu možete da preuzmete sa našeg sajta, na adresi:  
<http://bit.ly/2KswBOq>

Kada je fajl preuzet, raspakujte ili ekstrahujte direktorijum, koristeći najnoviju verziju:

- ▣ WinRAR/7-Zip za Windows
- ▣ Zipeg/iZip/UnRarX za Mac
- ▣ 7-Zip/PeaZip za Linux

## Upotrebljene konvencije

Postoji veliki broj konvencija teksta koje su upotrebljene u ovoj knjizi.

CodeInText - Ukazuje na reči koda u tekstu, nazive tabele baze podataka, nazive direktorijuma, nazive fajlova, ekstenzije fajlova, nazive putanje, skraćene URL-ove, korisnički unos i Twitter postove. Evo i primera: „Instalirajte preuzeti imidž fajl WebStorm-10\*.dmg kao drugi disk na vaš sistem.“

Blok koda je prikazan na sledeći način:

```
#!/bin/bash
# This is comment line echo „Hello World“
ls date
```

Kada želimo da privučemo pažnju na određeni deo bloka koda, relevantne linije ili stavke će biti ispisane zadebljanim slovima:

```
#!/bin/bash
# This is comment line
echo „Hello World“
ls date
```

Svaki unos komandne linije ili ispis će biti prikazan na sledeći način:

```
$ bash hello.sh
```

**Zadebljana slova** - Ukazuju na novi termin, važnu reč ili reči koje vidite na ekranu. Na primer, reči u menijima ili okvirima za dijalog prikazane su u tekstu na sledeći način. Evo i primera: „Selektujte **System info** iz panela **Administration**.“



Napomene ili važna obaveštenja prikazani su ovako.



Saveti i trikovi su prikazani ovako.

## POVRATNE INFORMACIJE

Povratne informacije od naših čitalaca su uvek dobrodošle.

**Osnovne povratne informacije** - Pošaljite e-mail na adresu [informatori@kombib.rs](mailto:informatori@kombib.rs)

i u naslovu poruke napišite naslov knjige. Ako imate bilo kakva pitanja o bilo kom aspektu ove knjige, pošaljite nam e-mail na adresu [informatori@kombib.rs](mailto:informatori@kombib.rs)

**Štamparske greške** - Iako smo preduzeli sve mere da bismo obezbedili tačnost sadržaja, greške su moguće. Ako pronađete grešku u ovoj knjizi, bili bismo vam zahvalni ako biste nam to prijavili. Posetite stranicu knjige:

1. <http://bit.ly/2ttSaUj>
2. **Kliknite:** Ostavite komentar
3. **Opišite** grešku.

**Piraterija** - Ako pronađete ilegalnu kopiju naših knjiga u bilo kojoj formi na Internetu, molimo vas da nas o tome obavestite i pošaljete nam adresu lokacije ili naziv web sajta. Stupite u kontakt sa nama na adresi [copyright@packtpub.com](mailto:copyright@packtpub.com) i pošaljite nam link ka sumnjivom materijalu.

## RECENZIJA

Zašto ne biste, kada pročitate i upotrebite ovu knjigu, napisali vaše mišljenje na sajtu sa kojeg ste je poručili? Potencijalni čitaoci mogu da vide i upotrebe vaše mišljenje da bi se odlučili za kupovinu, mi u “Packtu” možemo da razumemo šta mislite o našim proizvodima, a naši autori mogu da vide povratne informacije o svojoj knjizi.

Za više informacija o “Packtu” posetite sajt [packtpub.com](http://packtpub.com).

---

# Početak rada i shell skriptovanje

Ako koristite Linux, susrešćete se sa shellom. To je, obično, prvi program koji ćete koristiti. **Grafički korisnički interfejs (GUI)** postao je veoma popularan zbog lakoće upotrebe. Oni korisnici koji žele da iskoriste moć Linuxa upotrebiće shell program:

- Shell je program koji obezbeđuje korisniku direktnu interakciju sa operativnim sistemom. Hajde da pojasnimo faze u evoluciji Linux operativnog sistema. Linux je razvijen kao besplatna zamena otvorenog koda za Unix OS. Linux je razvijen kao besplatna zamena otvorenog koda za Unix OS. Hronologija je sledeća: Unix operativni sistem izdat je 1970. godine - razvili su ga Ken Thomson i Dennis Ritchie godinu dana ranije, a ponovo su ga napisali pomoću jezika C 1972. godine.
- Linus Torvalds je 1991. godine razvio Linux kernel za besplatni operativni sistem.

U ovom poglavlju obuhvaćene su sledeće teme:

- upoređivanje shellova
- upotreba shella
- učenje osnovnih Linux komandi
- prvi skript - Hello World
- kompajler i interpreter – razlike u procesima
- kada ne treba koristiti skriptove
- različiti direktorijumi
- efikasnija upotreba osnovnih komandi shella
- upotreba dozvola

## UPOREĐIVANJE SHELOVA

Na početku je Unix OS koristio shell program pod nazivom **Bourne shell**. Nakon toga, razvijani su mnogi shell programi za različite verzije Unixa. Slede kratke informacije o različitim shellovima:

- sh - Bourne shell
- csh - C shell
- ksh - Korn shell
- tcsh – poboljšani C shell
- bash - GNU Bourne Again shell
- zsh – ekstenzija za bash, ksh i tcsh
- pdksh – ekstenzija za ksh

Upoređivanje različitih shellova je predstavljeno u sledećoj tabeli.

FUNKCIJA	BOURNE	C	TC	KORN	BASH
Alijasi	Ne	Da	Da	Da	Da
Editovanje komandne linije	Ne	Ne	Da	Da	Da
Napredno podudaranje obrazaca	Ne	Ne	Ne	Da	Da
Završavanje naziva fajla	Ne	Da	Da	Da	Da
Stekovi direktorijuma (pushd i popd)	Ne	Da	Da	Ne	Da
Istorija	Ne	Da	Da	Da	Da
Funkcije	Da	Ne	Ne	Da	Da
Vezivanje ključa	Ne	Ne	Da	Ne	Da
Kontrola posla	Ne	Da	Da	Da	Da
Ispravljanje pravopisnih grešaka	Ne	Ne	Da	Ne	Da
Formatiranje upozorenja	Ne	Ne	Da	Ne	Da

U kodu vidimo da je, generalno, 95 odsto od sintakse svih ovih shellova slično. U ovoj knjizi mi ćemo pratiti Bash shell programiranje.

---

# ZADACI KOJE SHELL IZVRŠAVA

Kad god ukucamo neki tekst u Terminal shella, shell (`/bin/bash`) je odgovoran da pravilno izvrši komandu. Aktivnosti koje shell izvršava su sledeće:

- čitanje teksta i raščlanjavanje unete komande
- procenjivanje meta karaktera, kao što su zamenski znakovi, specijalni karakteri ili karakteri istorije
- obrada io-preusmeravanja, usmeravanja i pozadinska obrada
- obrada signala
- pokretanje programa za izvršenje

Opisaćemo prethodne teme u narednim poglavljima.

## UPOTREBA SHELLA

Započnite rad otvaranjem Terminala i upoznaćete bash shell okruženje:

1. Otvorite Linux Terminal i ukucajte:

```
$ echo $SHELL
/bin/bash
```

2. Prethodni izlaz u Terminalu ukazuje da je aktuelni shell `/bin/bash`, kao što je Bash shell:

```
$ bash -version
GNU bash, version 4.3.48(1)-release (x86_64-pc-linux-gnu)
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later
http://gnu.org/licenses/gpl.html
```

```
This is free software; you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
```

Sada ćemo upotrebiti reč `shell` da bismo označili samo Bash shell. Ako želimo da upotrebimo bilo koji drugi shell, on će biti konkretno pomenut po nazivu, kao što su `KORN` i drugi slični shellovi.

U Linuxu nazivi fajlova ispisani malim i velikim slovima se razlikuju; na primer, fajlovi `Hello` i `hello` su potpuno različiti. Po tome se Windows razlikuje, jer tamo nije važna veličina slova.

Ako je moguće, izbegavajte upotrebu razmaka u nazivima fajlova ili nazivima direktorijuma, kao što su:

- pogrešan naziv fajla - `Hello World.txt`
- pravilan naziv fajla - `Hello_World.txt` ili `HelloWorld.txt`

Pogrešno ispisani naziv fajla će izazvati grešku za određene pomoćne programe ili komande ili oni neće funkcionisati kako bi trebalo - na primer, pomoćni program `make`.

Dok kucate nazive fajlova ili nazive direktorijuma postojećih fajlova ili direktorijuma, upotrebite Linuxovu funkciju `tab completion`. U tom slučaju će Linux funkcionisati mnogo brže.

# UČENJE OSNOVNIH KOMANDI LINUXA

U sledećoj tabeli izlistane su neke osnovne Linux komande:

KOMANDA	OPIS
<code>\$ ls</code>	Ova komanda se koristi za proveru sadržaja direktorijuma.
<code>\$ pwd</code>	Ova komanda se koristi za proveru aktuelnog radnog direktorijuma.
<code>\$ mkdir work</code>	Mi ćemo raditi u posebnom direktorijumu pod nazivom <code>work</code> u direktorijumu <code>home</code> . Upotrebite ovu komandu za kreiranje novog direktorijuma pod nazivom <code>work</code> u aktuelnom direktorijumu.
<code>\$ cd work</code>	Ova komanda će promeniti radni direktorijum na novokreirani direktorijum <code>work</code> .
<code>\$ pwd</code>	Ova komanda može da se upotrebi za verifikaciju da li smo se pomerili u očekivani direktorijum.
<code>\$ touch hello.sh</code>	Ova komanda se koristi za kreiranje novog praznog fajla pod nazivom <code>hello.sh</code> u aktuelnom direktorijumu.
<code>\$ cp hello.sh bye.sh</code>	Ova komanda se koristi za kopiranje jednog fajla u drugi. Ona će kopirati fajl <code>hello.sh</code> kao fajl <code>bye.sh</code> .



KOMANDA	OPIS
\$ mv bye.sh welcome.sh	Ova komanda se koristi za promenu naziva fajla. Ona će promeniti naziv fajla <code>bye.sh</code> na <code>welcome.sh</code> .
\$ ll	Ova komanda će prikazati detaljne informacije o fajlu.
\$ mv welcome.sh .welcome.sh \$ ls	Hajde da prikazemo malo "magije". Promenićemo fajl, koristeći komandu <code>mv</code> , i pokrenućemo komandu <code>ls</code> . Sada komanda <code>ls</code> neće prikazati fajl <code>.welcome.sh</code> . Fajl je skriven. Svaki fajl ili direktorijum čiji naziv započinje tačkom (.) postaje skriven.
\$ ls -a	Ova komanda se koristi za prikaz skrivenih fajlova.
\$ rm .welcolme.sh	Ova komanda se koristi za brisanje fajla.

Ako izbrisemo bilo koji fajl iz GUI-a, on će biti pomeren u direktorijum svih izbrisanih fajlova `/home/user/.local/share/Trash/files/`.

## NAŠ PRVI SKRIPT - HELLO WORLD

Pošto ste naučili osnovne komande u Linux OS-u, sada ćemo napisati prvi shell skript, pod nazivom `hello.sh`. Možete da upotrebite bilo koji editor po izboru, kao što su `vi`, `gedit`, `nano`, `emacs`, `geany` i drugi slični. Ja najviše koristim editor `vi`:

1. Kreirajte novi fajl `hello.sh` na sledeći način:

```
#!/bin/bash
# This is comment line
echo "Hello World"
ls
date
```

2. Snimite fajl koji ste kreirali.

Linija `#!/bin/bash` se naziva šebeng linija. Kombinacija karaktera `# i !` se naziva čarobna sekvenca. Shell koristi ovu sekvencu za pozivanje željenog shella, kao što je u ovom slučaju `/bin/bash`. To bi uvek trebalo da bude prva linija u shell skriptu.

Sledećih nekoliko linija u shell skriptu su jasne:

- Svaka linija koja počinje sa # biće tretirana kao linija komentara. Izuzetak je prva linija `#!/bin/bash`.
- Komanda `echo` će odštampati `Hello World` na ekranu.
- Komanda `ls` će prikazati sadržaj direktorijuma u konzoli.
- Komanda `date` će prikazati aktuelni datum i vreme.

Možemo da izvršimo novi fajl pomoću sledećih komandi:

- Prva tehnika:

```
$ bash hello.sh
```

- Druga tehnika:

```
$ chmod +x hello.sh
```

Pokretanjem bilo koje od prethodnih komandi dodaćemo izvršne dozvole u novi fajl koji smo kreirali. Naučićete više o dozvolama fajla kasnije u ovom poglavlju.

```
$ ./hello.sh
```

Pokretanjem prethodne komande izvršićemo `hello.sh` kao izvršni fajl. Pomoću prve tehnike prosledili smo naziv fajla kao argumenat u `bash shell`.

Rezultat izvršenja fajla `hello.sh` će biti sledeći:

```
Hello World
hello.sh
Sun Jan 18 22:53:06 IST 2015
```

Pošto smo uspešno izvršili prvi skript, nastavice ćemo kreiranjem naprednijeg skripta `hello1.sh`. Kreirajte novi `hello.sh` skript na sledeći način:

```
#!/bin/bash
# This is the first Bash shell
# Scriptname : Hello1.sh
# Written by: Ganesh Naik
echo "Hello $LOGNAME, Have a nice day !"
echo "You are working in directory `pwd`."
echo "You are working on a machine called `uname -o`."
echo "List of files in your directory is : "
ls      # List files in the present working directory
echo "Bye for now $LOGNAME. The time is `date +%T`!"
```

Rezultat izvršenja fajla hello.sh će biti sledeći:

```
Hello student, Have a nice day !.  
Your are working in directory /home/student/work.  
You are working on a machine called GNU/Linux.  
List of files in your directory is :  
hello1.sh hello.sh  
Bye for now student. The time is 22:59:03!
```

Naučićete više o komandama LOGNAME, uname i drugim sličnim u nastavku knjige.

## KOMPAJLER I INTERPRETER – RAZLIKE U PROCESU

U razvoju bilo kog programa dve opcije su sledeće:

- **kompajliranje** - upotreba jezika zasnovanog na kompajleru, kao što su C, C++, Java i drugi slični
- **interpreter** - upotreba jezika zasnovanih na interpreteru, kao što je Bash shell skriptovanje

Kada koristimo jezik zasnovan na kompajleru, kompajliramo kompletan izvorni kod; kao rezultat, dobićemo binarni izvršni fajl. Zatim, izvršavamo binarni fajl da bismo proverili performansu programa.

Sa druge strane, kada razvijamo shell skript, kao što je program zasnovan na interpreteru, svaka linija programa je unos u Bash shell. Linije shell skripta su izvršene jedna po jedna, sekvencijalno. Čak i ako druga linija skripta ima grešku, shell interpreter će izvršiti prvu liniju.

## KADA NE TREBA KORISTITI SKRIPTOVE

Shell skriptovi imaju određene prednosti nad programima zasnovanim na kompajleru, kao što su C ili C++ jezik. Međutim, shell skriptovanje ima i određena ograničenja.

Slede neke prednosti:

- Skriptove je lako pisati.
- Skriptovi su brzi za pokretanje i za pronalaženje grešaka.
- Štedi se vreme u razvoju.
- Zadaci administracije su automatizovani.
- Nisu potrebni dodatno podešavanje ili alatke za razvoj ili testiranje shell skriptova.

Slede ograničenja shell skriptova:

- Svaka linija u shell skriptu kreira novi proces u operativnom sistemu. Kada izvršimo kompajlirani program, kao što je C, on se pokreće kao jedan proces za ceo program.
- Pošto svaka komanda kreira novi proces, shell skriptovi su spori, u poređenju sa kompajliranim programima.
- Shell skriptovi nisu pogodni ako su uključene teže matematičke operacije.
- Postoji problem sa međuplatformskim prenosom.

Ne možemo da koristimo shell skriptove u sledećim situacijama:

- gde su potrebne opširne operacije fajla
- gde su potrebne strukture podataka, kao što su povezane liste ili stabla
- gde treba da generišemo ili manipulišemo grafikama ili GUI-om
- gde nam je potreban direktan pristup hardveru sistema
- gde nam je potreban port ili priključak I/O
- gde treba da upotrebimo biblioteke ili interfejs sa zastarelim kodom
- gde se koriste vlasničke aplikacije zatvorenog koda (shell skriptovi otkrivaju izvorni kod da bi ceo svet mogao da ga vidi)

## RAZLIČITI DIREKTORIJUMI

Istražićemo strukturu direktorijuma u Linuxu, jer će nam to kasnije biti korisno:

- `/bin/` - Sadrži kormande koje koristi regularni korisnik.
- `/boot/` - Sadrži fajlove potrebne za pokretanje operativnog sistema i skladištenje.
- `/cdrom/` - Kada je priključen CD-ROM, njegovi fajlovi su ovde dostupni.
- `/dev/` - Fajlovi drajvera uređaja su uskladišteni u ovom direktorijumu. Oni će pokazati programe povezane sa hardverom koji su pokrenuti u kernelu.
- `/etc/` - Ovaj direktorijum sadrži konfiguracione fajlove i početne skriptove.
- `/home/` - Ovaj direktorijum sadrži početni direktorijum svih korisnika, osim administratora.
- `/lib/` - Fajlovi biblioteke su uskladišteni u ovaj direktorijum.
- `/media/` - Eksterni mediji, kao što je USB drajv, priključeni su u ovom direktorijumu.

- /opt/ - Opcioni paketi su instalirani u ovaj direktorijum.
- /proc/ - Sadrži fajlove koji pružaju informacije o kernelu i svakom procesu koji se pokreće u OS-u.
- /root/ - Ovo je početni direktorijum administratora.
- /sbin/ - Sadrži komande koje koriste administrator ili osnovni korisnik.
- /usr/ - Sadrži sekundarne programe, biblioteke i dokumentaciju o programima povezanim sa korisnikom.
- /var/ - Sadrži promenljive podatke, kao što su HTTP, TFTP, evidencije i drugo.
- /sys/ - Dinamički kreira sys fajlove.

## EFIKASNIJA UPOTREBA SHELLA – OSNOVNE KOMANDE

Potrebno je da naučite nekoliko komandi koje se često koriste, kao što su man, echo, cat i slične:

- Unesite sledeću komandu. Ona će vam prikazati različite tipove stranica sa uputstvima za rad, koje prikazuje komanda man:

```
$ man man
```

- Iz sledeće tabele možete da pretpostavite koji su različiti tipovi man stranica za istu komandu:

BROJ ODELJKA	OBLAST VAŽENJA
1	Komande korisnika
2	Pozivi sistema
3	Pozivi biblioteke
4	Specijalni fajlovi
5	Formati fajlova
6	Igrice
7	Razno
8	Administracija sistema
9	Rutine kernela

- Možemo da unesemo komandu `man` za prikaz odgovarajućih stranica sa uputstvima na sledeći način:

```
$ man 1 command
$ man 5 command
```

- Pretpostavimo da znate više o komandi `passwd`, koja se koristi za menjanje aktuelne lozinke korisnika. Možete da unesete komandu na sledeći način:

```
$ man command
man -k passwd // show all pages with keyword
man -K passwd // will search all manual pages content for pattern
"passwd"
$ man passwd
```

- Sledeća komanda će prikazati informacije o komandi `passwd`:

```
$ man 5 passwd
```

- Prethodna komanda će prikazati informacije o fajlu `passwd`, koji se nalazi u direktorijumu `/etc/`, kao što je `/etc/passwd`.

- Možemo da vidimo kratke informacije o komandi na sledeći način:

```
$ whatis passwd
```

- Ispis:

```
passwd (1ssl)      - compute password hashes
passwd (1)         - change user password
passwd (5)         - the password file
```

- Svaka komanda koju unesemo u Terminal ima izvršni binarni programski fajl koji je sa njom povezan. Možemo da proverimo lokaciju binarnog fajla na sledeći način:

```
$ which passwd
/usr/bin/passwd
```

- Prethodna linija ukazuje da se binarni fajl `passwd` nalazi u direktorijumu `/usr/bin/passwd`.
- Možemo da dobijemo kompletne informacije o lokaciji binarnog fajla i lokaciji stranice sa uputstvima bilo koje komande pomoću sledeće komande:

```
$ whereis passwd
```

- Ispis će biti sledeći:

```
passwd: /usr/bin/passwd /etc/passwd /usr/bin/X11/passwd
/usr/share/man/man1/passwd.1.gz /usr/share/man/man1/passwd.1ssl.gz
/usr/share/man/man5/passwd.5.gz
```

- Promenite prijavu korisnika i aktuelno korisničko ime:

```
$ whoami
```

- Ova komanda prikazuje korisničko ime prijavljenog korisnika:

```
$ su
```

- Komanda `su` (switch user) će korisnika učiniti administratorom, ali treba da znate lozinku administratora. Komanda `sudo` (superuser do) će pokrenuti komandu sa privilegijama administratora. Korisnik treba da bude dodat u listu `sudoers`.

```
# who am i
```

- Ova komanda će prikazati aktuelnog korisnika koji u tom momentu radi.

```
# exit
```

- Često će biti potrebno da kreirate nove komande od postojećih. Ponekad postojeće komande imaju složene opcije za pamćenje. U takvim slučajevima možemo da kreiramo nove komande na sledeći način:

```
$ alias ll='ls -l'
$ alias copy='cp -rf'
```

- Da bismo izlistali sve deklarisanе alijase, upotrebićemo sledeću komandu:

```
$ alias
```

- Da bismo uklonili alijas, upotrebićemo sledeću komandu:

```
$ unalias copy
```

- Možemo da proverimo detalje operativnog sistema, kao što su UNIX/Linux ili distribucija koja je instalirana, pomoću sledeće komande:

```
$ uname
```

- Rezultat

```
Linux
```

Biće prikazane osnovne informacije o OS-u (Unix naziv)

- Informacije o verziji Linux kernela će biti prikazane pomoću sledeće komande:

```
$ uname -r
```

- Ispis:

```
3.13.0-32-generic
```

- Da bismo dobili sve informacije o Linux mašini, upotrebićemo sledeću komandu:

```
$ uname -a
```

- Ispis:

```
Linux localhost.localdomain 3.10.0-693.el7.x86_64 #1 SMP Tue Aug 22  
21:09:27 UTC 2017 x86_64 x86_64 x86_64 GNU/Linux
```

- Sledeće komande će nam prikazati više informacija o Linux distribuciji:

```
$ cat /proc/version // detailed info about distribution  
$ cat /etc/*release  
# lsb_release -a
```

```
.
```

```
[student@localhost ~]$ lsb_release -a  
LSB Version: :core-4.1-amd64:core-4.1-noarch:cxx-4.1-amd64:cxx-4.1-noarch:desktop-  
-4.1-amd64:desktop-4.1-noarch:languages-4.1-amd64:languages-4.1-noarch:printing-4.1-  
amd64:printing-4.1-noarch  
Distributor ID: CentOS  
Description: CentOS Linux release 7.4.1708 (Core)  
Release: 7.4.1708  
Codename: Core
```

- Komanda `cat` se koristi za čitanje fajlova, a prikazana je na standardnom ispisu.



- Ponekad treba da kopiramo fajl ili direktorijum na više mesta. U takvim situacijama, umesto da iznova kopiramo originalni fajl ili direktorijum, možemo da kreiramo simboličke veze. U Windowsu to je slično funkciji za kreiranje prečice.

```
$ ln -s file file_link
```

- Da bismo otkrili tip fajla, možemo da upotrebimo komandni fajl. U Linuxu postoje različiti tipovi fajla. Neki primeri su sledeći:
  - regularni fajl (-)
  - direktorijum (d)
  - simbolička veza (l)
  - drajver znakovnog uređaja (c)
  - drajver blok uređaja (b)
  - fajl usmeravanja (p)
  - fajl priključka (s)
- Možemo da dobijemo informacije o fajlu upotrebom sledeće komande:

```
$ file file_name
```

- Štampanje teksta na ekranu za prikaz rezultata korisniku ili za upit detalja je veoma važna aktivnost.
- Sledeća komanda će kreirati novi fajl pod nazivom `file_name` pomoću komande `cat`:

```
$ cat > file_name
line 1
line 2
line 3
< Cntrl + D will save the file >
```

- Međutim, ova komanda se veoma retko koristi, jer mnogi moćni editori već postoje, kao što su `vi` ili `gedit`.
- Sledeća komanda će odštampati `Hello World` na konzoli. Komanda `echo` je veoma korisna za pisce shell skriptova:

```
$ echo "Hello World"

$ echo "Hello World" > hello.sh
```

- Komanda echo sa uključenim simbolom >menja sadržaj fajla. Ako sadržaj već postoji u fajlu, on će biti izbrisan i biće dodat novi. U situacijama u kojima treba da dodamo tekst u fajl možemo da upotrebimo komandu echo na sledeći način:

```
$ echo "Hello World" >> hello.sh    will append the text
```

- Sledeća komanda će kopirati znakovni niz Hello World u fajl hello.sh file:
- Sledeća komanda će prikazati sadržaj fajla na ekranu:

```
$ cat hello.sh
```

## UPOTREBA DOZVOLA

Slede tipovi dozvola:

- **dozvola za čitanje** - Korisnik može da čita ili da proveri sadržaj fajla.
- **dozvola za pisanje** - Korisnik može da edituje ili modifikuje fajl.
- **dozvola za izvršenje** - Korisnik može da izvrši fajl.

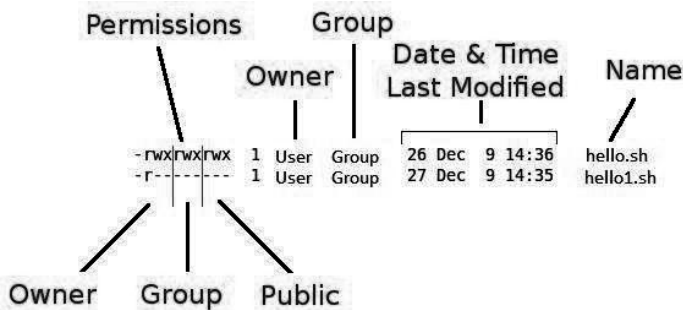
## Menjanje dozvola fajla

Slede komande za promenu dozvola fajla:

Da biste proverili dozvole fajla, unesite sledeću komandu:

```
$ ll file_name
```

Detalji o dozvolama fajla prikazani su u sledećem dijagramu:



U prethodnom dijagramu, kao što možete da vidite, dozvole su grupisane u dozvole vlasnika-korisnika, grupu i druge korisnike. Dozvole su za čitanje, pisanje i izvršenje. Možda će biti potrebno da promenimo dozvole za različite fajlove.

## Komanda `chmod`

Možemo da promenimo dozvole fajla ili direktorijuma na sledeća dva načina:

### Prva tehnika – simbolički metod

Sledeća komanda će dodati dozvole za čitanje/pisanje i izvršenje u fajl, gde je u za korisnika, g za grupu i o za ostale:

```
$ chmod ugo+rwX file_name
```

Alternativno, možemo da upotrebimo sledeću komandu:

```
$ chmod +rwX file_name
```

### Druga tehnika – numerički metod

Sledeća komanda će promeniti dozvole fajla pomoću oktalne tehnike:

```
$ chmod 777 file_name
```

Dozvola fajla `777` može da se razume kao `111 111 111`, što odgovara dozvolama `rwX.rwX.rwX`.

## Podešavanje `umask`

Videćete kako Linux odlučuje o standardnim dozvolama za novokreirani fajl ili direktorijum:

```
$ umask  
0002
```

Ako kreiramo novi direktorijum, iz dozvola `+rwx` će biti oduzeta dozvola `0002`. To znači da će za novokreirani direktorijum dozvole biti `775` ili `rwx rwx r-x`. Za novokreirani fajl dozvole fajla će biti `rw- rw- r--`. Prema standardnom podešavanju, za bilo koji novokreirani tekstualni fajl izvršni bit nikada neće biti podešen. Prema tome, novokreirani tekstualni fajl i direktorijum će imati različite dozvole, čak i ako je umask isti.

## Funkcija setuid

Još jedna veoma interesantna funkcionalnost je funkcija `setuid`. Ako je `setuid` bit podešen za skript, onda će se skript uvek pokretati sa privilegijama vlasnika, bez obzira koji korisnik pokreće skript. Ako administrator želi da pokrene skript koji je napisao on ili drugi korisnik, može da podesi ove bitove.

Vidite jednu od sledećih situacija:

```
$ chmod u+s file_name
$ chmod 4777 file
```

Dozvole fajla nakon bilo koje od prethodne dve komande će biti `drwsrwxrwx`.

## Funkcija setgid

Slično funkciji `setuid`, i funkcija `setgid` daje korisniku mogućnost da pokrene skriptove sa privilegijama grupe vlasnika, čak i ako ga izvrši bilo koji drugi korisnik:

```
$ chmod g+s filename
```

Alternativno, možete da upotrebite bilo koju od sledećih komandi:

```
$ chmod 2777 filename
```

Dozvole fajla nakon bilo koje od prethodne dve komande će biti `drwxrwsrwx`.

## Lepljivi bit

Lepljivi bit je veoma interesantna funkcionalnost. Pretpostavimo da u odeljenju administracije postoji 10 korisnika. Ako je jedan direktorijum podešen pomoću lepljivog bita, onda svi ostali korisnici mogu da kopiraju fajlove u taj direktorijum. Svi korisnici mogu da čitaju fajlove, ali samo vlasnik određenog fajla može da ga edituje ili da ga izbriše. Ostali korisnici mogu samo da čitaju, ali ne i da edituju ili modifikuju fajlove ako je podešen lepljivi bit:

```
$ chmod +t filename
```

Alternativno, možete da upotrebite sledeću komandu:

```
$ chmod 1777
```

Dozvole fajla nakon bilo koje prethodne dve komande će biti `drwxrwxrwt`.

## REZIME

U ovom poglavlju predstavili smo različite načine pisanja i pokretanja shell skriptova, načine za obradu fajlova i direktorijuma i upotrebu dozvola.

U sledećem poglavlju ćete učiti o upravljanju procesima, kontroli posla i automatizaciji.

