



OSNOVE LINUX-A ZA WEB DEVELOPERE

Uvod

- Nakon razvoja web prezentacije ili aplikacije, potrebno je istu objaviti – to znači da će biti dostupna svim internet korisnicima koji znaju njen URL* (*Uniform Resource Locator*), tj. putanju do određenog internet sadržaja, tj. do web prezentacije/aplikacije.
- Sintaksa URL je osmišljena da bi bila proširiva te je moguće prikazati adresu u bilo kom setu znakova

https://salapura.com/www/InternetProgramiranje/linux_za_web_developere-osnove.pdf

Iz adrese se vidi da se radi o PDF dokumentu

'[linux_za_web_developere-osnove.pdf](#)'

koji se nalazi na web serveru gdje se hostuje domena '[salapura.com](#)', na putanji '[/www/InternetProgramiranje/](#)' i da se dokumentu pristupa HTTPS protokolom ('[https://](#)')

Uvod...

- Elementi prezentacije (fajlovi, skriptovi, dokumenti, grafički elementi...) trebaju biti fizički smješteni* na web server
- Prostor za smještaj web aplikacije se može obezbjediti kroz web hosting, posvećeni server (dedicated server) ili VPS server (*Virtual Private Server*).

WEB HOSTING

- **Web hosting** (često zvan *shared hosting*), najčešći je oblik hostinga koji se nudi i u najvećem broju zadovoljava potrebe korisnika za smještaj vlastitih web stranica.
- Zakupom web hosting računa dobivaju se prostor i pristup odvojenom računu na serveru (tu su i drugi korisnici), pri čemu svi korisnici dijele ukupne resurse servera na kojeg su smješteni.
- Smještaj stranica na shared web hosting ima nekoliko bitnih prednosti u odnosu na druga dva rješenja (VPS ili posvećeni server):
 - sve je u startu spremno za korištenje
 - Veoma jednostavna instalacija web aplikacije
 - jednostavno kreiranje e-mail adresa
 - server je namijenjen isključivo za web hosting uslugu pa je posebno optimiziran za što bolje performanse
 - jednostavno upravljanje
 - povoljna cijena web hosting paketa

WEB HOSTING

- Nedostaci Web hostinga su:
 - nemogućnost naprednijeg podešavanja platforme (npr. MySQL)
 - ograničenje u resursima (svaki račun ima određene resurse – CPU/RAM)
 - potrebe naprednijih korisnika, poput web shopova s velikim brojem posjeta, brzo prerastu mogućnosti koje im pruža shared okruženje

DEDICATED SERVER

- Posvećeni server je potpuno različit od usluge web hostinga:
 - kod dedicated servera korisnik ima pristup ukupnim resursima,
 - ima puni pristup operativnom sistemu i
 - može server podesiti i koristiti po želji
- Ova vrsta hostinga sa sobom vuče i znatno višu cijenu
- Kreiran je za posebne korisnike, tj. korisnici koji imaju trebaju vlastiti resursi:
 - korisnici koji imaju zahtjevne web aplikacije s velikim brojem posjeta
 - Firme koje servere koriste za poslovne aplikacije i baze
 - korisnici koji trebaju specifične postavke ili virtualizaciju.

VPS SERVER

- VPS (*Virtual Private Server*) server je usluga koja je smještena između web hostinga i dedicated servera.
- Omogućava iskorištavanje posvećenih resursa (CPU, RAM, storage) i veoma je pogodna za korisnike koji prerastu web hosting okružje, ali i dalje ne trebaju snagu cijelog dedicanog servera.
- Korištenjem VPS dobijaju se mnoge prednosti koje pruža dedicated server, ali za mnogo manju cijenu.
- Uz VPS se dobija puni administratorski pristup pa je moguće server podesiti po želji (baš kao vlastiti server).

Uporedni pregled

	Web hosting	VPS server	Dedicated server
IP adresa	Ista IP adresa za web aplikacije	Vlastita javna IP adresa	Vlastita javna IP adresa
Vlastiti izbor OS	Ne (samo Linux)	Da	Da
Kontrola nad uslugom	Ograničena	Puna kontrola	Puna kontrola
Zasebni resursi	Ne	Da	Da
Instalacija vlastitih aplikacija	Ne	Da	Da
Resursi servera	Ograničeni na korisnički račun	Zakupljeni resursi	Puni resursi servera
Cijena	XY / godinu	XY / mjesec	10*XY / mjesec

VPS server - linux

- Postoji velika razlika u ponudi OS na VPS serverima: Linux je mnogo zastupljeniji i jeftiniji od sličnih opcija sa Windows OS*
- Za postavljanje i održavanje web aplikacije na VPS-u koji je zasnovan na Linux operativnom sistemu, potrebno je osnovno poznavanje administracije Linux servera.
- Administracija je olakšana specijalizovanim aplikacijama (ControlPanel, Plesk) koje često nisu besplatne (mogu koštati zamalo kao i iznajmljivanje samog VPS-a.
- Čak iako se obezbjedi aplikacija za administriranje servera, poželjno je i potrebno je da osoba koja upravlja web aplikacijom (web developer) ima osnovno znanje za rad sa Linux OS.

Linux – bash shell

- Rad sa linux operativnim sistemom omogućen je kroz *Bash Shell* (terminal) gdje korisnik može da:
 - Izvršava komande iz komandne linije
 - Pregleda istoriju prethodno izvršenih komandi
 - Kontrolira zadatke (*job control*)
 - Koristi *shell* funkciju i alijase (*alias*)
 - Izvršava aritmetičke operacije

Linux – bash script

- Datoteka (*file*) sa ekstenzijom `.sh`, koja sadrži niz komandi koje se izvršavaju redoslijedom kojim su navedene se naziva ***bash skripta***.
- Zbog načina tretiranja svega u okviru Unix OS (sekvence bajtova bez strukture), i fajlovi i uređaji (mrežne kartice, hard diskovi, particije, tastatura, štampač) se tretiraju kao objekti sličani fajlu.
- Svaka hardverska komponenta je predstavljena kao jedan specijalni fajl odgovarajućeg naziva naziv i lokacije (npr. memorija sistema je definisana kao fajl sa lokacijom `/dev/mem`, a tastatura i ekran su definisani u fajlu `/dev/tty1`).

Pristup Linux serveru – SSH protokol

- Za konekciju i komunikaciju sa udaljenim linux sistemom n se najčešće se korisiti **SSH protokol** (Secure Shell)
- SSH je kriptografski mrežni protokol za siguran pristup mrežnim resursima usluga preko nezaštićene mreže.
- SSH komande su šifrovane i sigurne jer su oba kraja veze (na strani klijenta i servera) autentifikovana upotrebom digitalnog sertifikata, dok su lozinke zaštićene kriptovanjem.
- Protokol radi po modelu klijent-server: povezivanje inicira SSH klijent koji se povezuje na SSH server.
- SSH klijent upravlja procesom uspostavljanja veze pri čemu za potvrdu identiteta SSH servera koristi kriptografiju javnog ključa (PKI).
- Nakon faze podešavanja, protokol SSH koristi snažne simetrične algoritme za šifrovanje kako bi se osigurala privatnost i integritet podataka koji se razmjenjuju između klijenta i servera.

Pristup Linux serveru – SSH protokol

- Za konekciju na udaljeni server je potrebno:
 - hostname IP
 - Korisničko ime
 - Lozinka
- Udaljenom serveru se pristupa korišćenjem odgovarajućih naredbi, pa u zavisnosti da li je *user* isti kao na serveru (ili nije), postoje dvije različite naredbe:
 - ukoliko je *user* isti kao i na serveru, koristi se:

```
ssh ip-adresa-udaljenog-servera
```
 - ako nije isti *user*, koristi se :

```
ssh naziv-usera-udaljenog-servera@ip-adresa-udaljenog-servera
```
- Nakon ovoga, unosi se lozinka, a za prekid sesije se koristi naredba:

```
exit
```

Aplikacije koje podržavaju SSH protokol

- Udaljenom serveru pristupa se kroz *bash* (terminal), postoji više windows aplikacija koje emuliraju *linux bash*:
 - Putty je najstarija/najpoznatija windows aplikacija za komunikaciju sa udaljenim sistemom koja koristi SSH protokol
 - PuttyTray
 - Kitty
 - MobaXterm
 - mRemoteNG
 - Smartty

Kopiranje i sinhronizacija fajlova - Rsync

- **Rsync** je brz i izuzetno prilagodljiv alat za kopiranje fajlova. Može kopirati fajlove lokalno, kao i sa drugog hosta.
- Nudi veliki broj opcija i omogućava fleksibilnu specifikaciju izbora datoteka koje treba kopirati.
- Poznat je po algoritmu prenosa “*delta*” koji smanjuje količinu podataka poslatih preko mreže tako što šalje samo razlike između izvornih datoteka i postojećih datoteka u odredištu.
- Postoje dva različita načina da Rsync da stupi u kontakt sa udaljenim sistemom: koristeći SSH ili direktno kontaktiranjem dinamičkog Rsync preko TCP-a.

```
rsync options source destination
```

Rad sa fajlovima u lokalu

Naredba	Objašnjenje
<code>rsync -zvh test.txt /home/test1/</code>	Fajl <i>test.txt</i> će biti kopiran ili sinhronizovan unutar foldera test1 čija je putanja /home/test1/
<code>rsync -t *.c foo:src/</code>	Prethodna naredba bi prenela sve datoteke koje odgovaraju obrazcu * .c iz trenutnog direktorijuma u direktorijum "src" na mašini "foo". Ako bilo koja od datoteka već postoji na udaljenom sistemu, onda se protokol rsync daljinskog ažuriranja koristi za ažuriranje datoteke slanjem samo razlika.
<code>rsync -avz foo:src/bar /data/tmp</code>	Rekurzivno kopiranje svih datoteka iz direktorijuma src/bar na mašini foo u direktorijum /data/tmp/bar na lokalnoj mašini. Datoteke se prenose u režimu "lqarchiverq", što osigurava da se u prenosu čuvaju simboličke veze, uređaji, atributi, dozvole, vlasništva itd. Pored toga, kompresija će se koristiti za smanjenje veličine dijelova podataka prenosa.
<code>rsync -avz foo:src/bar/ /data/tmp</code>	Dodatna kosa crtna na izvoru foo:src/bar/ omogućava da se izbjegne stvaranje dodatnog nivoa direktorijuma na odredištu.

Sinhronizacija sa lokalnog servera na udaljeni

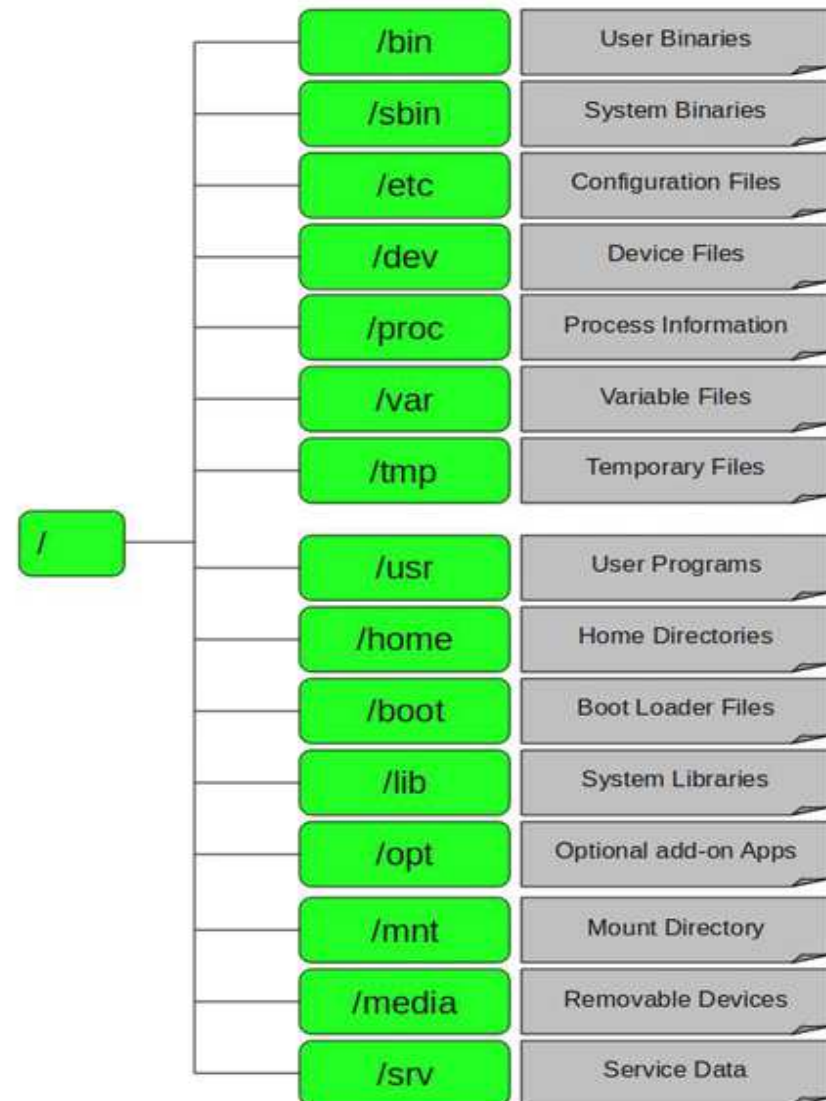
Naredba	SSH	Objašnjenje
<pre>rsync -avzh public_html/test.txt root@2.2.1.2:/home/</pre>	NE	fajl <i>test.txt</i> koji se nalazi u lokalnoj mašini je kopiran ili sinhronizovan na mašinu sa IP adresom 2.2.1.2 u direktorijum <i>home</i> .
<pre>rsync -avzhe ssh public_html/test.txt root@2.2.1.2:/home/</pre>	DA	fajl <i>test.txt</i> koji se nalazi u lokalnoj mašini je kopiran ili sinhronizovan na mašinu sa IP adresom 2.2.1.2 u direktorijum <i>home</i> koristeći SSH protokol.

Sinhronizacija sa udaljenog servera na lokalni

Naredba	SSH	Objašnjenje
<pre>rsync - avzh root@2.2.1.2:/home/test.txt /home/test/</pre>	NE	kao korisnik "root" kopira se sa servera čija je IP adresa 2.2.1.2 iz foldera <i>home</i> fajl test.txt na lokalni server u folder home/test/
<pre>rsync - avzhe root@2.2.1.2:/home/test.txt /home/test/</pre>	DA	kao korisnik "root" kopira se sa servera čija je IP adresa 2.2.1.2 iz foldera <i>home</i> fajl test.txt na lokalni server u folder home/test/ koristeći SSH protokol
<pre>sync -avze ssh --exclude '*txt' rroot@2.2.1.2:/home/test.txt home/test/</pre>	DA	kao korisnik "root" kopira se sa servera čija je IP adresa 2.2.1.2 iz foldera <i>home</i> sve osim ".txt" fajlova na lokalni server u folder home/test/ koristeći SSH protokol

Struktura Linux file sistema

- Slika prikazuje najčešću strukturu Linux file sistema
- Osnovni direktorijum je *Root* koji jedini nema roditeljski direktorijum. Označava se sa kosom crtom “/” i pristup ovome direktorijumu ima samo `root` korisnik.
- Ako je putanja nekog fajla navedena na način da počinje kosom crtom “/”, u pitanju je **apsolutna putanja** - putanja u odnosu na osnovni (`root`) direktorij.
- **Relativna putanja** se određuje u odnosu na trenutnu poziciju fajla i počinje sa “./” (trenutni folder).



Struktura Linux file sistema

- `/bin` – ovaj direktorij sadrži binarne izvršne komande, kao što su: ps, ls, ping, grep, cp....
- `/etc` - sadrži konfiguracione fajlove za sve programe, kao i shell skripte za započinjenje i gašenje aplikacija.
- `/dev` - Linux uređaje ,vidi kao fajlove, pa ovaj direktorijum sadrži više specijalnih fajlova koji predstavljaju uređaje. Npr. “`/dev/sda`” predstavlja prvi SATA disk u sistemu.
- `/home` - ovaj folder sadrži privatne i konfiguracione fajlove za svakog korisnika (oni imaju “*write*” pristup u vlastitom folderu, za pisanje u drugim folderima potrebna je `root` privilegija).
- `/media` – Folder sadrži podfoldere u kojima su prikazani prenosivi mediji ubačeni u računar.
- `/root` - Ovaj direktorijum je direktorijum `root` korisnika (ne nalazi se na “`/home/root`“, sa ostalim user-ima, već ovdje).
- `/srv` - Ovaj direktorijum sadrži podatke za usluge koje pruža sistem (npr. ako se koristite Apache HTTP server za web stranice, vjerovatno će se datoteke web sajta nalaziti unutar `/srv` foldera).
- `/tmp` – ovaj folder čuva privremene fajlove (fajlovi se brišu pri svakom pokretanju sistema)
- `/usr` - Ovaj direktorijum sadrži aplikacije i fajlove koje koriste korisnici, za razliku od aplikacija i fajlova koje koristi sistem (ne-esencijalne aplikacije se nalaze unutar direktorija “`/usr/bin`” da ne bi bile u folderu `/bin`, a u folder “`/usr/local`” se instaliraju lokalne aplikacije čime se sprječava da se nepotrebno “*zagadi*” ostatak sistema.
- `/var` - U ovaj direktorijum se snimaju promjenjivi (varijabilni) fajlovi - fajlovi za koje se očekuje da će im se sadržaj uvećavati. Najčešće su to:
 - system log files `/var/log`,
 - packages and database files `/var/lib`, ili
 - emails `/var/mail...`

Osnovne Linux komande

- Pomoć pri pisanju naredbi

U radu sa *Bash shell*-om se koristi:

- Tab na tastaturi - automatski dopuniti imena fajlova i direktorijuma.
- Simboli “?” i “*” su džoker znakovi (wildcard) u Linux-u (“?” mijenja jedan karakter, dok će “*” zameniti cijeli string).
- Kopiranje teksta u komandnoj liniji terminala vrši se naredbom Ctrl+Shift+C dok je paste se naredbom Ctrl+Shift+V

- Sintaksa svih Linux komandi se može predstaviti na sljedeći način:

```
[sudo] command [optional switch] [file or directory path]
```

Naredba	Objašnjenje
[command] --help	Izlistava opcije za korišćenje komande
man [command]	Prikazaju uputstvo (eng. “manual”) za traženu komandu.
sudo [command]	Komanda se izvršava sa administratorskim pravima

Bush prečice

Naredba	Objašnjenje
CTRL + c	Zaustavlja izvršavanje trenutne naredbe
CTRL + z	Sleep program
CTRL + a	Šalje kursor na početak naredbe
CTRL + e	Šalje kursor na kraj naredbe
CTRL + u	Briše sve karaktere od početka do kursora
CTRL + k	Briše sve karaktere od kursora do kraja linije
CTRL + r	Pretraga kroz korišćene naredbe (izlistavne sa ponovnim CTRL + R)
!!	Repeat last command
! <i>abc</i>	Pokreće poslednju naredbu koja počinje sa <i>abc</i>
! <i>abc</i> :p	Štampa u terminalu poslednju naredbu koja počinje sa <i>abc</i>
!*	Štampa sve argumente iz prethodne naredbe

Monitoring i informacije

Naredba	Objašnjenje
<code>ps</code>	Prikazuje sve trenutno aktivne procese koje je inicirao user
<code>ps -ef</code>	Prikazuje sve trenutno aktivne procese
<code>ps -ef grep imeprocesa</code>	Filtrira procese prema imenu procesa i prikazuje informacije o izabranom procesu u terminalu (linux pipeline: kada "ps -ef" izlista sve trenutne procese onda "grep" filtrira izlistane procese prema definisanom nazivu procesa)
<code>kill processID</code>	Ubija proces čiji ID je naveden
<code>killall processName</code>	Ubija proces čije ime je processName
<code>du -sh naziv_foldera/*</code>	Izlistava veličinu svih poddirektorijuma ili fajlova unutar trženog direktorijuma
<code>vmstat</code>	Prikazuje statistiku virtualne memorije
<code>cat /proc/meminfo</code>	Prikazuje podatke o memoriji
<code>cat /proc/cpuinfo</code>	Prikazuje podatke o procesoru
<code>free -h</code>	Prikazuje slobodnu i zauzetu memoriju (-h za human readable, -m za MB, -g za GB.)
<code>id</code>	Prikazuje id od usera i njegove grupe
<code>last</code>	Prikazuje listu poslednje logovanih usera na sistem
<code>ifconfig -a</code>	Prikazuje sve podatke o network interfejsu i IP adresi
<code>ping google.com</code>	Šalje kontrolnu poruku do traženog host-a i štampa odgovor
<code>dig google.com</code>	Prikazuje DNS informacije o traženom host-u
<code>netstat -nutlp</code>	Prikazuje sve tcp i udp portove koji oslušuju saobraćaj
<code>badblocks -s /dev/sda</code>	Testiranje hard diska

Rad sa file sistemom

Naredba	Objašnjenje
<code>pwd</code>	Štampa u terminalu putanju trenutnog direktorijuma
<code>mkdir test-direktorijum</code>	Pravi novi direktorijum <i>test-direktorijum</i>
<code>cd test-direktorijum</code>	Ulazak u direktorijum <i>test-direktorijum</i>
<code>cd ..</code>	Go up a directory
<code>ls</code>	Izlistava sve fajlove u trenutnom working direktorijumu
<code>ls -a</code>	Izlistava sve fajlove u trenutnom working direktorijumu uključujući i hidden
<code>ls -R</code>	Rekuzivno izlistava sve fajlove u trenutnom working direktorijumu uključujući sve poddirektorijume
<code>ls -t</code>	Izlistava sve fajlove u trenutnom working direktorijumu sortirane prema filteru last modified
<code>ls -Sa</code>	Izlistava sve fajlove u trenutnom working direktorijumu sortirane prema filteru size
<code>ls -l</code>	Izlistava sve fajlove u trenutnom working direktorijumu u punom formatu
<code>ls -1</code>	Izlistava sve fajlove u trenutnom working direktorijumu – svaki fajl u novu liniju
<code>ls -m</code>	Izlistava sve fajlove u trenutnom working direktorijumu, fajlovi su razdvojeni zarezom.
<code>touch novi_fajl.txt</code>	Kreira novi fajl pod nazivom <i>novi_fajl.txt</i>
<code>less neki_fajl</code>	Štampa sadržaj tekstualnog fajla <i>neki_fajl</i> u terminalu (izlazak sa "Q")
<code>cat neki_file1 neki_file2</code>	Štampa u terminalu tekstualni sadržaj fajlova (ukoliko ih ima više spaje sve u jedan ispis)
<code>file file1</code>	Štampa u terminalu podatke o fajlu <i>file1</i>
<code>cp pic.jpg pic2.jpg</code>	Kopira fajl u working direktorijumu <i>pic.jpg</i> i snima ga kao novi fajl pod imenom <i>pic2.jpg</i> isto u working direktorijumu
<code>cp /home/pic.jpg /home/backup/pic.jpg</code>	Iz direktorijuma home se kopira fajl <i>pic.jpg</i> i snima se u direktorijum backup kao novi fajl pod imenom <i>pic.jpg</i>
<code>cp -R /home/slike /home/backup</code>	Kopira ceo direktorijum slike sa svim poddirektorijumima u folder backup (prvi postaje poddirektorijum od foldera backup)
<code>cp *.jpg /home</code>	Kopira sve fajlove sa ekstenzijom .jpg u working direktorijumu i snima ih u home direktorijum sa istim nazivima
<code>mv /home/file1 /home/file2</code>	Rename fajl <i>file1</i> iz foldera home u <i>file2</i>
<code>mv /home/file1 /tmp/file2</code>	Move fajl <i>file1</i> iz foldera home u folder tmp i čuva ga kao fajl <i>file2</i>
<code>rm file1</code>	Briše fajl <i>file1</i>
<code>head file1</code>	Štampa u terminalu prvih 10 linija tekstualnog fajla <i>file1</i>
<code>tail file1</code>	Štampa u terminalu poslednjih 10 linija tekstualnog fajla <i>file1</i>
<code>tail -3 file1</code>	Štampa u terminalu poslednje 3 linije tekstualnog fajla <i>file1</i>

Pretraga - *search*

Naredba	Objašnjenje
<code>grep "neki tekst" /home/logovi.txt</code>	Pretraga teksta "neki tekst" unutar fajla <i>logovi.txt</i> . Ukoliko postoji rezultat, on se štampa u terminalu
<code>grep -r "neki tekst" /home/</code>	Rekurzivna pretraga teksta kroz sve fajlove u okviru home direktorijuma i njegovih poddirektorijuma
<code>grep -i "neki tekst" /home/</code>	Case insensitive pretraga teksta "neki tekst" kroz sve fajlove u okviru home direktorijuma
<code>find /home/ -name "test*"</code>	Pretraga fajlova u okviru home direktorijuma prema imenu (-name), a koje započinje sa <i>test</i>
<code>find /home/ -user "pera"</code>	Pretraga fajlova u okviru home direktorijuma prema useru (-user), čije ime započinje sa <i>pera</i>
<code>find /home/ -mmin 60</code>	Pretraga za svim fajlovima u okviru home direktorijuma koji su modifikovani u zadnjih 60 minuta
<code>find /home/ -amin 60</code>	Pretraga za svim fajlovima u okviru home direktorijuma kojima je pristupano (access) u zadnjih 60 minuta
<code>find /home/ -size 60M</code>	Pretraga za svim fajlovima u okviru home direktorijuma koji su veći od 60MB
<code>find /home/ -size +50M -size -100M</code>	Pretraga za svim fajlovima u okviru home direktorijuma koji su veći od 50MB a manji od 100MB
<code>find /home -type f -empty</code>	Pretraga za svim fajlovima u okviru home direktorijuma koji su prazni
<code>find /home -type d "test"</code>	Pretraga za direktorijumima u okviru home direktorijuma pod nazivom "test"
<code>find /home -type d -empty</code>	Pretraga za direktorijumima u okviru home direktorijuma koji su prazni

Prava pristupa - nivoi

- Svaki dokument/folder ima svoga vlasnika – najčešće je to korisnik koji je kreirao dokument/ folder.

Vlasnik određuje prava pristupa nad dokumentom/folderom.

- Svaki dokument/folder pripada određenoj grupi korisnika – najčešće je to ista grupa kojoj pripada i korisnik (ali ne mora biti)

Vlasnik određuje kojoj grupi pripada dokument/folder.

- Linux razlikuje 3 nivoa dodjeljivanja prava pristupa (ugo):

- u – pravo pristupa koje se odnosi na vlasnika (user)
- g – pravo pristupa koje se odnosi na grupu (group)
- o – pravo pristupa koje se odnosi na ostale (other)

- Za svaki od navedenih nivoa moguće je zasebno određivati prava pristupa

Prava pristupa - vrste

- Korisnicima mogu biti dodijeljena prava pristupa nad datotekama/folderima:
 - r - pravo čitanja (`read`)
 - w - pravo izmjene sadržaja (`write`)
 - x - pravo izvršavanja (`execute`).
- U zavisnosti od toga da li je u pitanju dokument ili folder, prava pristupa su:

	read	write	execute
datoteke	Sadržaj se može čitati (<code>more, cat</code>).	Sadržaj se može mijenjati (u datoteku se može pisati).	Datoteka se može izvršavati (može se koristiti kao naredba).
mape	Sadržaj se može čitati (<code>ls</code>).	Sadržaj se može mijenjati (<code>rm, cp, mv</code>).	Može postati radna mapa (<code>cd</code>).

- Prava pristupa se mogu pročitati upotrebom komande

```
ls -l          #(long listing format)
```

```
ls -a         #(do not ignore entries starting with.)
```

Prava pristupa

```
ls -al
total 76
```

drwxrwxr-x	13	anupam	anupam	4096	Aug 28 21:45	.
drwxr-xr-x	34	anupam	anupam	4096	Aug 28 20:52	..
drwxr-xr-x	2	anupam	anupam	4096	Aug 28 12:19	bin
drwxrwxr-x	5	anupam	anupam	4096	Aug 28 12:19	config
-rw-rw-r--	1	anupam	anupam	154	Aug 28 12:19	config.ru
drwxrwxr-x	3	anupam	anupam	4096	Aug 28 12:37	db
-rw-rw-r--	1	anupam	anupam	2871	Aug 28 12:19	Gemfile.lock
drwxrwxr-x	7	anupam	anupam	4096	Aug 28 21:45	.git
-rw-rw-r--	1	anupam	anupam	466	Aug 28 12:19	.gitignore
drwxrwxr-x	2	anupam	anupam	4096	Aug 28 12:29	log
drwxrwxr-x	2	anupam	anupam	4096	Aug 28 12:19	public
-rw-rw-r--	1	anupam	anupam	478	Aug 28 12:19	README.rdoc
drwxrwxr-x	8	anupam	anupam	4096	Aug 28 12:19	test
drwxrwxr-x	6	anupam	anupam	4096	Aug 28 12:33	tmp

1

2

3

4

5

6

1 – vrsta: **d** za direktorij, '-' za datoteke

drwxrwxr-x

-rw-rw-r--

i prava pristupa (rwx) za tri grupe (*ugo*)

rwx rwx rwx

2 - broj linkova (sistemski podatak)

3 - ime vlasnika

3 - ime grupe

4 - veličina u bajtovima

5 - datum i vrijeme zadnje modifikacije

6 - ime datoteke ili mape

Prava pristupa - primjeri

`rw-rwxrwx`

svi imaju pravo čitanja, pisanja i izvršavanja

`rw-r-xr-x`

vlasnik ima sva prava, a grupa i svi ostali imaju samo prava čitanja i izvršavanja

`rw-r-----`

vlasnik ima pravo čitanja i pisanja, grupa ima samo pravo čitanja, a ostali korisnici nemaju nikakvo pravo pristupa.

Dodjeljivanje prava pristupa

- Prava pristupa se dodjeljuju komandom `chmod`
- Komanda koristi simboličke i numeričke oznake

Simboličke oznake

- Simboličke oznake koje se koriste su `ugo` i `rxw`, uz operatore dodaj '+', oduzmi '-' i izjednači '='
- Opšti oblik komande je

```
chmod permisije [,permisije] datoteka
```

Permisije		
Ko	Vrsta promjene	Vrsta pristupa
u (vlasnik)	'+' dodaj	r čitanje
g (grupa)	'-' oduzmi	w pisanje
o (ostali)	'=' izjednači	x izvršavanje
a (svi)		

Dodjeljivanje prava pristupa

- Najjednostavniji način izmjene prava pristupa je navođenje 3 znaka:
 - na koga se promjena odnosi,
 - o kakvoj promjeni je riječ,
 - vrsta pristupa koja se mijenja

```
$ chmod a=r abc
$ ls -l abc
-r--r--r-- 1 kurs01 kurs 2332 Oct 21 2013 abc
$ chmod u+wx abc
$ ls -l abc
-rwxr--r-- 1 kurs01 kurs 2332 Oct 21 2013 abc
$ chmod g+x,o-r abc
$ ls -l abc
-rwxr-x--- 1 kurs01 kurs 2332 Oct 21 2013 abc
```

prije	komanda	poslije
rw- r-- ---	chmod u-w,g-r	r-- --- ---
	chmod a+x	rwX r-X --X
	chmod ug+x	rwX r-X ---
	chmod a=rx	r-X r-X r-X

Numeričke oznake prava pristupa

Opšti oblik komande je

```
chmod permisije [,permisije] datoteka
```

Svaka vrsta pristupa nosi brojčanu vrijednost

read	4
write	2
execute	1

Num. Vrijednost	Permisije	Permisije
4+2+1=7	čitanje, pisanje i izvršavanje	rxw
4+2+0=6	čitanje i pisanje	rw-
4+0+1=5	čitanje i izvršavanje	r-x
4+0+0=4	samo čitanje	r--
0+2+1=3	pisanje i izvršavanje	-wx
0+2+0=2	samo pisanje	-w-
0+0+1=1	samo izvršavanje	--x
0+0+0=0	ništa	---

Izmjena prava pristupa - `umask`

- Pri kreiranju datoteke određena su maksimalna prava pristupa `rw-rw-rw-`, a u slučaju foldera prava pristupa su `rw-rwxrwxrwx`, tj. svako može mijenjati sadržaj novih datoteka a u folderima svako može kreirati ili brisati datoteke.
- Naredbom `umask` moguće je promijeniti ova podrazumijevana prava pristupa dodijeljena prilikom kreiranja
- Prava pristupa se zadaju se maskom nizom od tri cifre – to su vrijednosti koje će se prilikom kreiranja datoteke ili foldera **oduzeti od maksimalnih** vrijednosti prava pristupa.
- Sintaksa komande je

```
umask [MASKA]
```
- Ako se ne zadaju argumenti, komanda ispisuje trenutna vrijednost maske.

Primjeri - umask

- U slučaju da se želi da pri kreiranju dokumenata vlasnici imaju sva prava, članovi grupe da nemaju pravo pisanja a ostali korisnici da nemaju pravo ni pisanja i čitanja dokumenta, komanda bi imala sljedeći oblik:

```
umask 026
```

- jer je

Maksimalne postavke	rw- rw- rw	6 6 6
Željene postavke	rw- r-- ---	6 4 0
Razlika		0 2 6

File permissions

Naredba	Objašnjenje
<code>chmod 644 file.html</code>	Vlasnik može da čita i piše (4+2), grupa može da čita, i ostali mogu da čitaju
<code>chmod u=rw example.jpg</code>	Vlasnik može da čita i piše (4+2)
<code>chmod -R 755 nekifolder</code>	Rekurzivno promeniti permisije u direktorijumu pod nazivom "nekifolder" i svim njegovim poddirektorijumima tako da vlasnik može da čita i piše i izvršava(4+2+1), grupa može da čita i izvršava (4+0+1), a ostali mogu da čitaju i izvršavaju (4+0+1)
<code>chmod 775 file</code>	Promena permisija Change mode of <i>file</i> to 775
<code>chmod -R 600 nekifolder</code>	Rekurzivna promena permisija na 600 za <i>nekifolder</i> i svih njegovih fajlova i podfoldera
<code>chown user:group nekifile</code>	Promena vlasništva nad fajlom <i>nekifile</i> na <i>group</i>