

**Sveučilište u Zagrebu**  
**PMF – Matematički odsjek**



# Mreže računala

Vježbe 07

**Zvonimir Bujanović**  
**Slaven Kožić**  
**Vinko Petričević**

# Klasični mrežni servisi i aplikacije

- Do sada smo upoznali mrežne aplikacije ssh i (s)ftp, te jednostavnije poput daytime i echo.
- Na ovim vježbama upoznat ćemo još nekoliko mrežnih servisa: SMTP, DNS i HTTP.
- Objasniti ćemo pozadinu i principe funkcioniranja protokola koje ti servisi koriste.

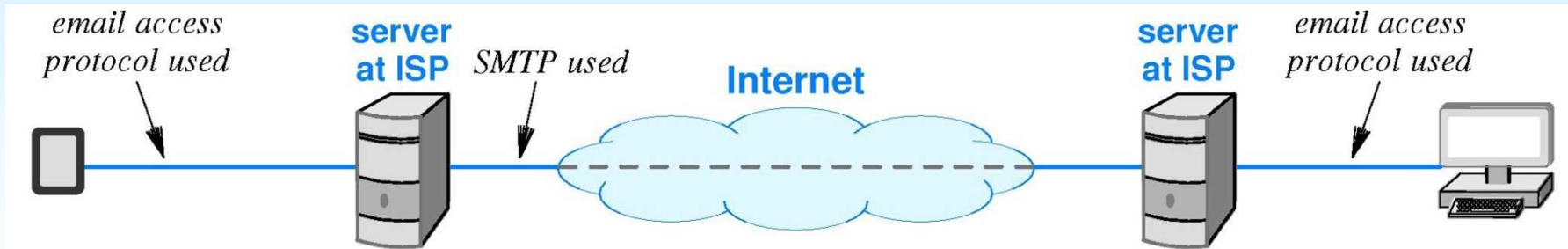
# telnet

- Primarna svrha: izvršavanje naredbi na udaljenom računalu; standardno koristi port 23. Nakon spajanja na taj port, te unosa korisničkog imena i lozinke, klijent je mogao izvršavati naredbe u UNIX ljuski na udaljenom računalu.
- telnet kroz mrežu šalje podatke neenkriptirano, pa je zamijenjen sigurnim protokolom ssh. Stoga je i port 23 obično zatvoren.
- Na ovim vježbama će nam telnet poslužiti za korištenje mrežnih servisa na drugim portovima.

# Zadatak 1

- Koristeći telnet, spojite se na port 7 računala cromath.math.hr.
- Koristeći telnet, spojite se na port 13 računala cromath.math.hr.
- Napomene:
  - Sve natipkano doslovno se šalje serveru.
  - Pomoću Ctrl + ] možemo pokrenuti razne komande unutar telnet-a, na primjer prekinuti konekciju ("close") i slično. Popis komandi dobivamo pomoću ?. Na komunikaciju sa serverom vraćamo pritiskom na Enter.

# Protokoli za elektroničku poštu



- "Server at ISP" za svakog korisnika čuva njegov poštanski sandučić (mailbox).
- Korisnik pristupa svom sandučiću preko web-browsera (npr. za Gmail) ili specijalizirane aplikacije (npr. MS Outlook). Za pristup sandučiću server koristi "email access protocol" poput POP3 ili IMAP.
- Za razmjenu mailova serveri koriste SMTP protokol.

# SMTP

- SMTP – Simple Mail Transfer Protocol
- SMTP je standard za prijenos i isporuku elektroničke pošte među računalima na Internetu. Razmjena e-mailova između polaznog i odredišnog poslužitelja odvija se putem SMTP protokola, bez obzira koristi li klijent webmail ili npr. MS Outlook.
- Svoje korijene SMTP vuče još iz ARPANET-a, a konkretan oblik dobiva 1982. godine kao RFC 821.
- Protokol je napisan isključivo kao tekstualni, za razmjenu ASCII datoteka. Za razmjenu binarnih datoteka koristi se enkodiranje 8-bitnog sadržaja u 7-bitni ASCII oblik (MIME).
- SMTP server standardno sluša na portu 25.
- Jedan od prvih SMTP servera koji su se pojavili bio je Sendmail, kasnije su slijedili Exim, Qmail, Postfix i drugi.

# Relaying

- Prilikom slanja e-maila, koristi se tzv. postupak prosljeđivanja (*relaying*) : korisnik se spaja na port 25 gdje sluša SMTP poslužitelj, te ako je korisnikova IP adresa u dozvoljenom rasponu adresa za koje se dopušta prosljeđivanje pošte poruka, poruka se preuzima od strane poslužitelja koji ju dalje šalje na odredište.
- Kako se zna koje računalo prima e-mail za koju domenu?
  - Svaka domena ima u svom DNS sustavu (vidi kasnije slide-ove) tzv. MX zapis (mail exchanger) koji pokazuje na računalo koje prima mail za tu domenu. Ime tog računala doznajemo pomoću naredbe "`host -t mx hostname`".
  - Ako želimo poslati e-mail na netko@math.hr, SMTP poslužitelj prvo pita DNS poslužitelj koje računalo prima mail za math.hr domenu, dobiva odgovor da je to mail.math.hr, spaja se na njega i isporučuje mu poruku na slijedeći način:

# Isporuka pošte

- Naredbom telnet spojimo se na port 25 poslužitelja mail.math.hr.
- Server odgovara “220 mail.math.hr ESMTP”.
- Šaljemo naredbu “HELO *hostname*” gdje je *hostname* ime našeg računala, kako bi se predstavili odredišnom poslužitelju.
- Server odgovara “250 mail.math.hr”.
- Šaljemo naredbu “MAIL FROM: naša@email.adresa” (pošiljatelj).
- Server odgovara “250 2.1.0 Ok”.
- Šaljemo naredbu “RCPT TO: netko@math.hr” (primatelj).
- Server odgovara “250 2.1.5 Ok”.
- Šaljemo naredbu “DATA” nakon čega slijedi tijelo poruke, kad smo gotovi u praznom redu napišemo točku i stisnemo enter.
- Prekidamo komunikaciju sa “quit”.



```
hrvoje@euler: ~  
File Edit View Terminal Tabs Help  
hrvoje@euler:~$ host -t mx math.hr  
math.hr mail is handled by 1 mail.math.hr.  
hrvoje@euler:~$ telnet mail.math.hr 25  
Trying 161.53.8.11...  
Connected to mail.math.hr.  
Escape character is '^]'.  
220 mail.math.hr ESMTP  
HELO hrvoje.org  
250 mail.math.hr  
MAIL FROM: hrvoje@hrvoje.org  
250 2.1.0 Ok  
RCPT TO: igor.jelaska@math.hr  
250 2.1.5 Ok  
DATA  
354 End data with <CR><LF>.<CR><LF>  
Ovo je demonstracija slanja poste...  
.  
250 2.0.0 Ok: queued as D45166C4115  
quit  
221 2.0.0 Bye  
Connection closed by foreign host.  
hrvoje@euler:~$
```

## Zadatak 2

- Koristeći isključivo naredbu telnet i gornji primjer kao referencu, direktno se spajajući na odredišni e-mail poslužitelj pošaljite e-mail kolege koji sjedi do Vas poruku sa svojim imenom, prezimenom i JMBAG brojem.
- Saznajte koje računalo prima e-mail za domene gmail.com, predsjednik.hr, iskon.hr.
- Zbog čega gmail.com ima više različitih računala koje primaju e-mail?

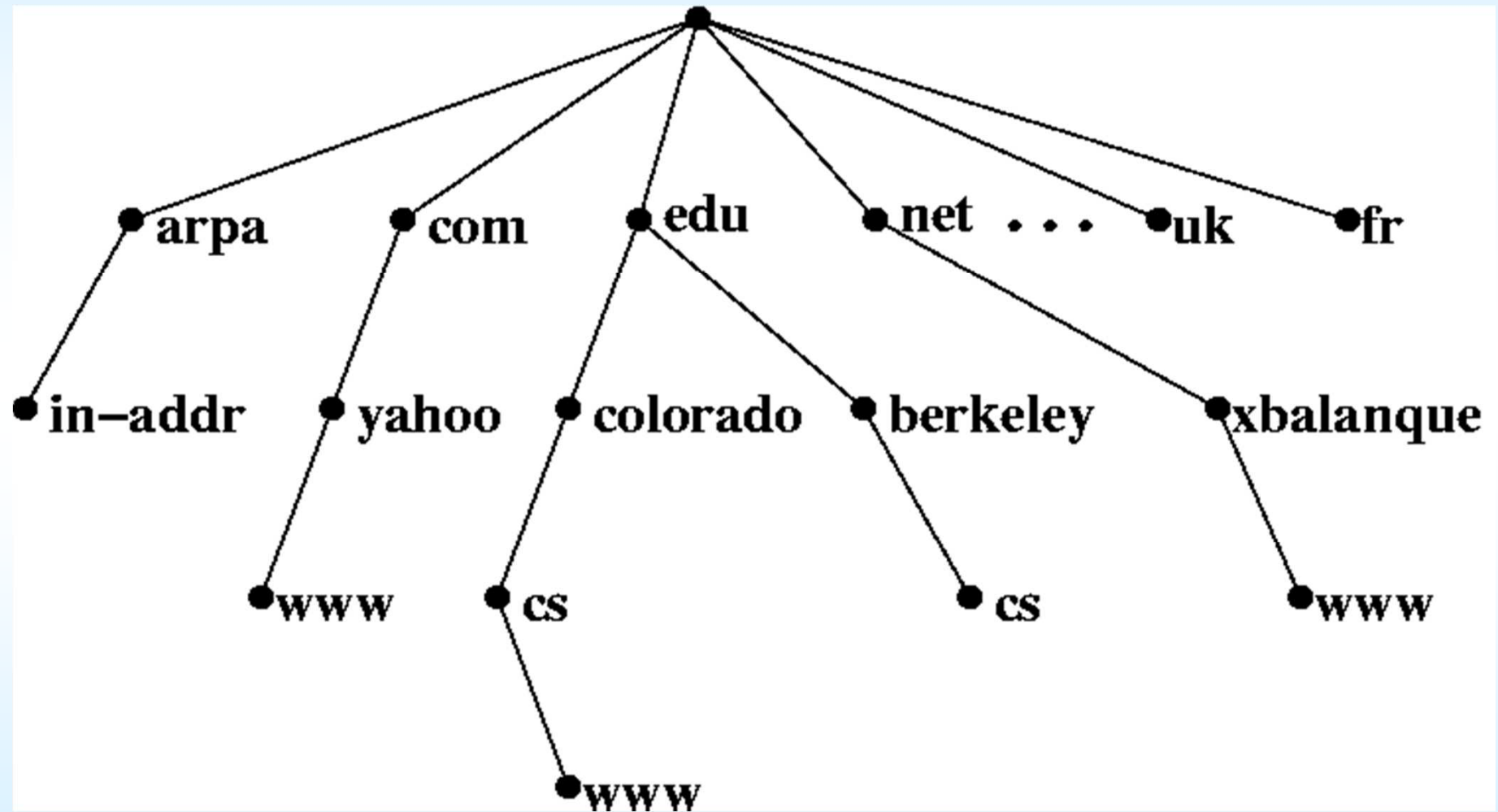
# DNS

- IP adrese računala nisu lako pamtljive, pa se za računala s mnogo korisnika koristi *hostname* – lako pamtljiv naziv (poput [www.google.com](http://www.google.com) ili [student.math.hr](http://student.math.hr)).
- Sva komunikacija na Internetu odvija se preko IP adresa (vidi: IP datagram!). Stoga je potrebna metoda konverzije *hostname*-a u IP adresu. To je zadaća *DNS-a* (*Domain name system*).
- Nastao je 1983. godine, nedugo nakon što Internet prelazi na korištenje TCP/IP protokola.
- Svojstva:
  - Računala mogu mijenjati IP adresu, a zadržati isti *hostname*.
  - Sistematizacija informacija na Internetu ovisno o vrsti domene (npr. com za komercijalne, edu za obrazovne institucije).
  - DNS je distribuirana baza podataka, svaki vlasnik domene odgovoran je i proizvoljno uređuje svoje domenske podatke.

# DNS

- Struktura podataka u DNS sustavu je u obliku stabla. Elementi tog stabla nazivaju se *domene*.
- Domena je riječ koja se može sastojati od slova, brojeva i crtice. Svaka domena može imati poddomene.
- TLD (Top Level Domena) – domena najvišeg nivoa je korijen stabla, postoji mnogo top level domena (.com, .net, .org, .edu, .hr, .de, itd.)
- Međunarodne komercijalno dostupne domene najvišeg nivoa su .com, .net, .org, .info, .biz.
- Svaka zemlja ima svoju domenu najvišeg nivoa čije ime odgovara kratici zemlje (.hr – Hrvatska, .de – Njemačka, .nl – Nizozemska, .at – Austrija...) za čije je održavanje obično nadležna neka državna agencija.
- Razina domene je njena visina u stablu, npr. domena druge razine je mit.edu, dok je lab.mit.edu domena treće razine; puno ime domene odgovara imenu svih čvorova od promatranog čvora do korijena odvojeno točkom.

# DNS

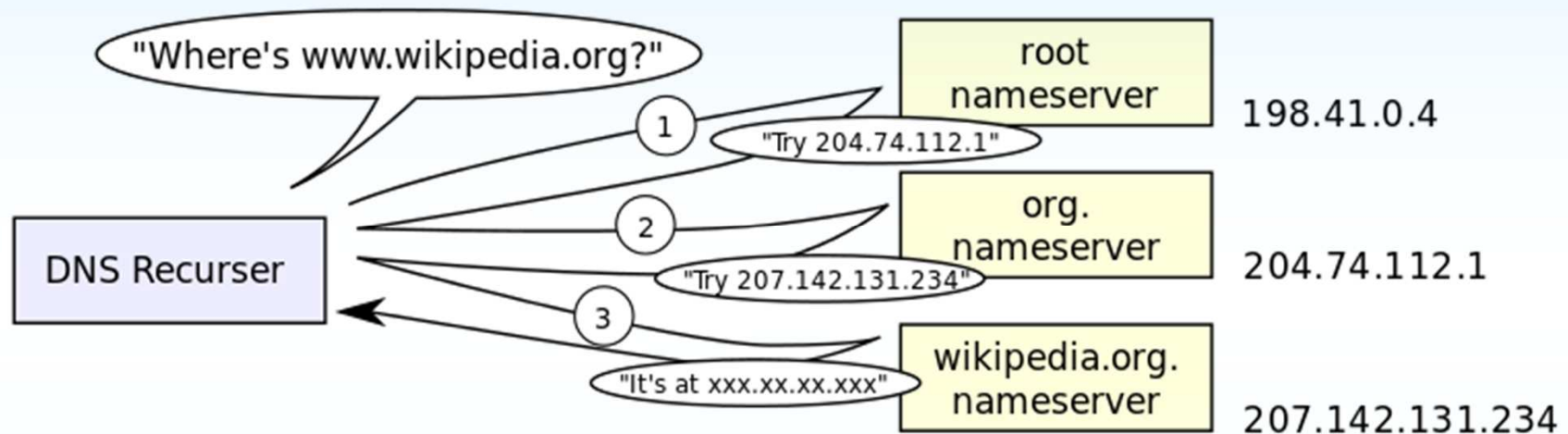


# Kako radi DNS?

- DNS obično koristi UDP protokol (svi ostali koje smo upoznali dosad koriste TCP!).
- Server standardizirano sluša na portu 53.
- Najpoznatiji DNS serveri su BIND, tinydns, maraDNS, itd.
- DNS upit radi ovako:
  1. Lokalni sistem je podešen da koristi unaprijed poznate vršne dns poslužitelje (*Root DNS Server*).
  2. Da bi se saznao podatak u DNS zapisu za neku domenu, potrebno je pitati njoj nadležan DNS poslužitelj.
  3. Upiti se vrše hijerarhijski: npr. ako želimo saznati IP adresu računala [www.google.com](http://www.google.com), prvo pitamo vršni DNS poslužitelj koji DNS poslužitelj je nadležan za .com domenu, zatim taj poslužitelj pitamo tko je nadležan za [google.com](http://google.com) domenu, te naposljetku pitamo poslužitelj nadležan za [google.com](http://google.com) domenu koja je IP adresa zapisa računala [www.google.com](http://www.google.com).

# Kako radi DNS?

- Nizom od 3 upita doznajemo IP adresu za `www.wikipedia.org`.



- DNS koristi *cache*-iranje da izbjegne ponavljanje upita.
- Za jednostavni DNS upit možemo koristiti alat "`host`".
- Za detalje oko pojedinih DNS upita možemo koristiti alat "`dig`":
  - npr. `dig www.math.hr` ili `dig iskon.hr mx`

# Vrste DNS zapisa

- **A zapis** – određeni *hostname* povezuje s IP adresom (npr za domenu math.hr A zapis “mail” sadrži IP adresu 161.53.8.11 pa računalo mail.math.hr ima IP adresu 161.53.8.11)
- **CNAME zapis** – pseudonim, pokazuje s jednog imena na drugo. Npr u zapisu www za domenu srce.hr, “www” je samo pseudonim za “regoc”, pravo ime računala www.srce.hr je regoc.srce.hr.
- **MX zapis** – (Mail eXchanger) sadrži informaciju koje računalo je zaduženo za primanje pošte za tu domenu. Npr, za domenu *iskon.hr* u mx zapisu stoji mx.iskon.hr, pa će se na to računalo isporučivati elektronska pošta za tu domenu.
- **NS zapis** – sadrži podatak koji su domenski poslužitelji nadležni za neku domenu.
- **SOA zapis** – (start of authority), specificira koji poslužitelji su ovlašteni davati autoritativne odgovore na domenske upite, serijski broj domenskih podataka, vrijednosti vezane za osvježavanje zapisa i dr.
- **PTR zapis** – određenu IP adresu povezuje s imenom (vidi idući slide).



# Reverzni DNS

- DNS omogućava i prevođenje IP adresa natrag u *hostname*, npr. ako želimo saznati koji je *hostname* računala s IP adresom 192.84.105.1
- Posebna domena, **in-addr.arpa** se koristi isključivo za tu namjenu.
- Upit za *hostname* koji pripada 192.84.105.1 se provodi ovako:
  - Generira se *hostname* 1.105.84.192.in-addr.arpa.
  - Provede se DNS upit za gornju adresu na način opisan ranije.
  - Pri tom upitu se traži DNS zapis tipa PTR. U tim zapisima je navedena IP adresa danog računala.
- Reverzni DNS upit možemo provesti pomoću:
  - **host IPadresa**
  - **dig -x IPadresa**

## Zadatak 3

- Pomoću naredbe `host` moguće je za određenu domenu izlistati određen tip zapisa (npr. A, PTR, SOA, MX, NS) pomoću parametra `-t`, npr. `host -t soa google.com`
- Koristeći `host` i `dig` saznajte slijedeće:
  - Koji su domenski poslužitelji nadležni za domenu `srce.hr`?
  - Koji poslužitelj prima elektronsku poštu za tu domenu?
  - Koja je IP adresa tog poslužitelja?
  - Sznajte ime poslužitelja čija je IP adresa `176.9.136.23`
  - Koji je domenski poslužitelj nadležan za prevođenje IP adresa oblika `161.53.103.*` u njihova imena?

# HTTP protokol

- HTTP (HyperText Transfer Protocol) – protokol za prijenos i isporuku podataka za World Wide Web.
- HTTP poslužitelj standardno sluša na portu 80.
- Web klijent se spaja na port 80, pomoću HTTP protokola dohvaća traženi sadržaj te ga formatira i ispisuje korisniku na zaslon.
- Najpoznatiji i najrašireniji HTTP poslužitelj na Internetu je *Apache*, podržan od strane velikog broja operacijskih sustava, otvorenog koda i besplatan za korištenje.
- Najpoznatije verzije HTTP protokola su HTTP/1.0 (svibanj 1996.) i HTTP/1.1 (lipanj 1999.)

# HTTP/1.0

- Klijent uspostavlja TCP vezu te šalje zahtjev za određenim sadržajem.
- Server vraća odgovor, koji se sastoji od željenog sadržaja ili poruke o grešci.
- Zahtjev se sastoji od:
  - Početne linije u kojoj specificiramo traženi sadržaj.
  - Linija zaglavlja (nisu nužne).
  - Prazne linije.
  - Opcionalnog tijela poruke (npr. datoteka koju šaljemo poslužitelju i sl.).

# HTTP/1.0

- Odgovor poslužitelja čine:
  - Statusna linija – verzija protokola, kod rezultata operacije i kratko objašnjenje koda, npr: HTTP/1.0 200 OK ili HTTP/1.0 404 Not Found
  - Linije zaglavlja, npr:

```
Date: Sat, 08 Dec 2007 22:34:04 GMT
Server: Apache/2.2.3 (Debian) PHP/5.2.0-8+etch7
Accept-Ranges: bytes
```
  - Tijela poruke, npr. HTML kod web stranice koju smo zatražili.

# Jednostavan primjer HTTP/1.0 komunikacije

- Želimo dohvatiti datoteku s adrese  
`http://www.nekoracunalo.com/putanja/datoteka.html`
- Na port 80 računala `www.somehost.com` šaljem zahtjev poput ovog:  
`GET /putanja/datoteka.html HTTP/1.0`  
[prazna linija]
- Dobivamo odgovor poput:  
`HTTP/1.0 200 OK`  
`Date: Sat, 10 Dec 2007 23:59:59 GMT`  
`Content-Type: text/html`  
`Content-Length: 1354`  
`<html> <body>`  
`<h1>Moja osobna web stranica!</h1>`  
(nastavlja se sadržaj datoteke) . . .  
`</body>`  
`</html>`

# HTTP

- Neki HTTP zahtjevi:
  - **GET** – klijent želi dohvatiti dokument; server odgovara slanjem statusnih informacija iza kojih slijedi kopija traženog dokumenta.
  - **HEAD** – klijent želi dohvatiti samo statusne informacije o nekom dokumentu; server odgovara slanjem istih (bez kopije dokumenta).
  - **POST** – klijent šalje podatke serveru, te želi da se postojeći dokument "proširi" tim podacima.
  - **PUT** – klijent šalje podatke serveru i želi da se ti podaci spreme na prosljeđenu adresu.

# HTTP

- Kodovi nekih HTTP odgovora:
  - 200 OK – zahtjev klijenta je uspio.
  - 400 Bad Request – zahtjev klijenta nije ispravan.
  - 404 Not Found – resurs kojeg klijent traži ne postoji.



# HTTP/1.1

- HTTP/1.0 za svaki element web stranice (za svaku pojedinu sliku, okvir i sl.) otvara novu TCP vezu, što opterećuje poslužitelj te usporava prijenos.
- HTTP/1.1 omogućuje prijenos više različitih elemenata traženog sadržaja putem jedne TCP veze (tzv. *persistent connection*).
- Ubrzava prikaz dinamički generiranih stranica omogućujući tzv. *chunked encoding*, tj. podjelu informacije koja se šalje u blokove poznate veličine čime se poslužitelju omogućuje da sa slanjem počne prije nego što je poznata konačna duljina prenesene informacije.
- Podrška za lokalno spremanje elemenata stranica (*cache*).
- Učinkovito korištenje IP adresa omogućujući tzv. virtualne poslužitelje – isti web poslužitelj može prikazivati web stranice za mnoštvo različitih domena, ovisno o zaprimljenom zahtjevu (npr zahtjev GET `http://www.nekadomena.hr` HTTP/1.1 prikazat će jednu web stranicu, dok zahtjev GET `http://www.nekadrugadomena.hr` HTTP/1.1 neku sasvim drugu bez obzira što se radi o istom poslužitelju).

## Zadatak 4

- Pomoću naredbe telnet dohvatite web-stranicu foruma `http://degiorgi.math.hr/forum/index.php`
- Što se dogodi kada umjesto GET koristite HEAD?
- Pomoću naredbe telnet pokušajte dohvatiti neku nepostojeću stranicu sa `degiorgi.math.hr`.
- Iz zaglavlja koje vam je poslužitelj poslao pročitajte o kojoj vrsti poslužitelja se radi (odgovor ćete naći u liniji koja počinje sa `Server:` )