

Uvod u veb i internet tehnologije

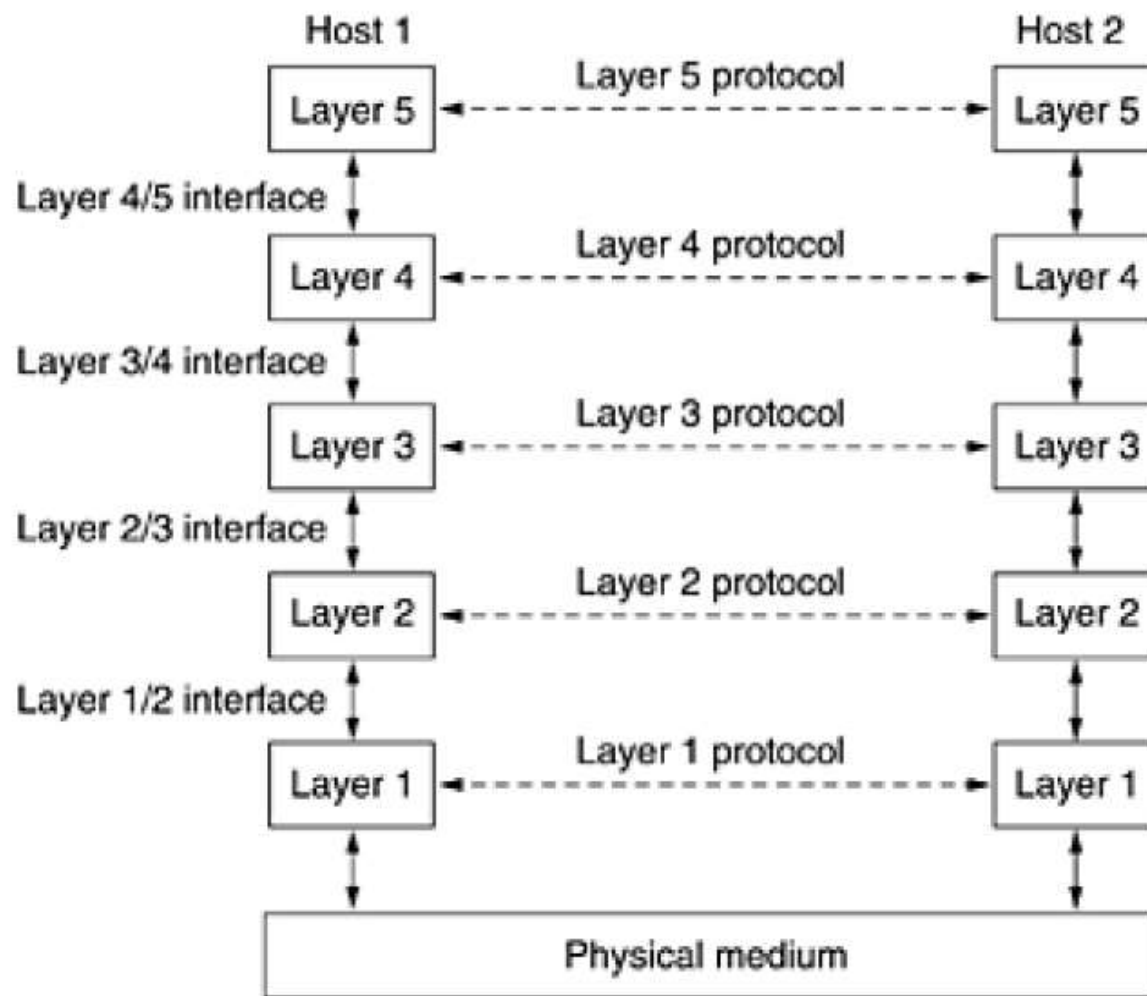




Slojevi kod računarskih mreža



Slojevitost mreža

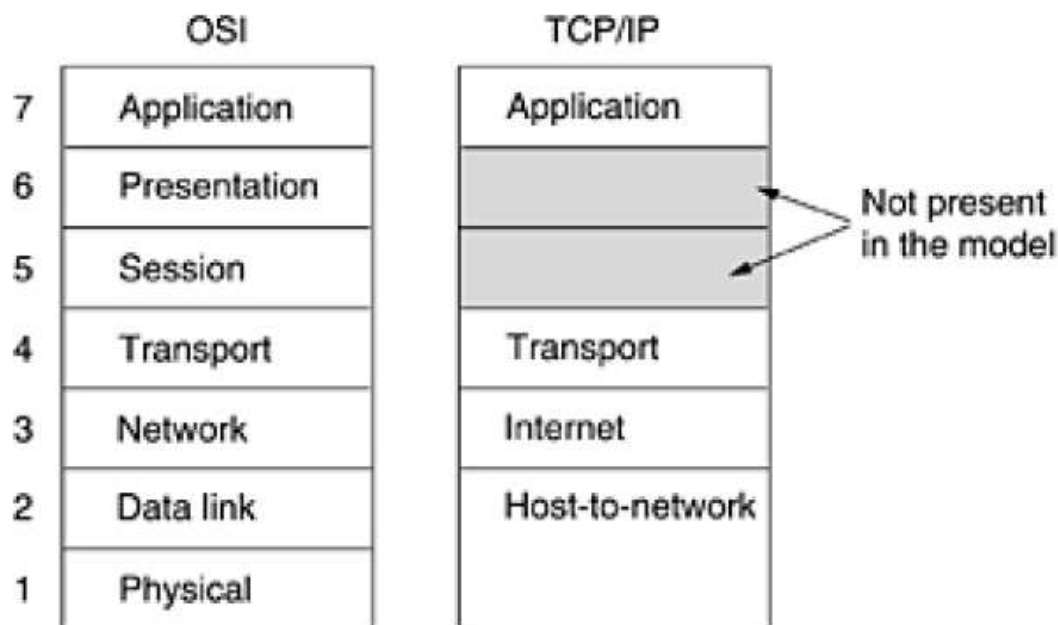


Prikaz komunikacije i protokola komunikacije



Slojevitost mreža (2)

- Istorijski, mreže se obično posmatraju u okviru dva referentna modela:
 - **OSI model** (Open Systems Interconnection) sa 7 slojeva, standardizovan od strane ISO
 - **TCP/IP model** sa 4 sloja, prisutan u okviru Interneta





Slojevi TCP/IP modela

Sledi opis uloga najznačajnijih slojeva u okviru **TCP/IP** modela:

- 1) Sloj „**host-prema-mreži**“ ili „pristup mreži“ („host-to-network“ layer) obezbeđuje kanal komunikacije na najnižem nivou
- 2) **Međumrežni** sloj (internet layer) - bavi se povezivanjem više računara u mrežu
- 3) **Transportni** sloj (transport layer) - ima zadatak da prihvata podatke sa viših slojeva, deli ih na manje jedinice (pakete), šalje te pakete na odredište korišćenjem nižih slojeva
- 4) **Aplikacioni** sloj (application layer) - definiše protokole koje direktno koriste korisničke aplikacije u okviru svoje komunikacije



Protokoli i slojevi

OSI sloj	TCP/IP sloj	Jedinica	Protokol
aplikativni sloj (mrežni procesi vezani za aplikaciju)	aplikativni sloj	podatak	HTTP, FTP, Telnet, DNS, DHCP, POP/SMTP, NNTP
sloj prezentacije (enkripcija i kodiranje podataka)		podatak	MIME, TLS, SSL
sloj sesije (uspostavljanje sesije krajnjih korisnika)		podatak	SSH, Named Pipes, PPTP
transportni sloj (veza, pouzdanost, transport)	transportni sloj	segment datatgram	TCP, UDP, SCTP, DCCP
mrežni sloj (logičko adresiranje i rutiranje)	međumrežni sloj	paket	IP (IPv4, IPv6), ICMP, ARP, RARP
sloj veze podataka (fizičko adresiranje, pristup medijumu)	sloj pristupa mreži	okvir	PPP, HDLC, Frame Relay
fizički sloj (prenos signala)		bit	Token Ring, RS-232, T1, E1, POTS, OTN, DSL, 802.11a/b/g/n PHY, 802.15.x PHY, Ethernet, USB, Bluetooth, Firewire (IEEE 1394)



Slojevi kod računarskih mreža “host-prema-mreži”



Protokoli i slojevi

OSI sloj	TCP/IP sloj	Jedinica	Protokol
aplikativni sloj (mrežni procesi vezani za aplikaciju)	aplikativni sloj	podatak	HTTP, FTP, Telnet, DNS, DHCP, POP/SMTP, NNTP
sloj prezentacije (enkripcija i kodiranje podataka)		podatak	MIME, TLS, SSL
sloj sesije (uspostavljanje sesije krajnjih korisnika)		podatak	SSH, Named Pipes, PPTP
transportni sloj (veza, pouzdanost, transport)	transportni sloj	segment datagram	TCP, UDP, SCTP, DCCP
mrežni sloj (logičko adresiranje i rutiranje)	međumrežni sloj	paket	IP (IPv4, IPv6), ICMP, ARP, RARP
sloj veze podataka (fizičko adresiranje, pristup medijumu)	sloj pristupa mreži	okvir	PPP, HDLC, Frame Relay
fizički sloj (prenos signala)		bit	Token Ring, RS-232, T1, E1, POTS, OTN, DSL, 802.11a/b/g/n PHY, 802.15.x PHY, Ethernet, USB, Bluetooth, Firewire (IEEE 1394)



Sloj „host-prema-mreži”

Sloj „Host-to-network“ obezbeđuje kanal komunikacije.

- Na najnižem nivou, obezbeđuje postojanje komunikacionog kanala i mogućnost slanja i primanja pojedinačnih bitova kroz komunikacioni kanal
- Na najnižem nivou u okviru ovog sloja nema kontrole grešaka
- Na višem nivou se međumrežnom sloju obezbeđuje postojanje pouzdanog kanala komunikacije u kome se:
 - greške automatski detektuju i ispravljaju (error control)
 - automatski se vodi računa o brzini slanja podataka kako se ne bi desilo da brzi uređaji zagušuju sporije (flow control)
- Ukoliko se koristi zajednički kanal komunikacije, na ovom sloju se vrši kontrola pristupa uređaja komunikacionom kanalu (medium access control)



Sloj „host-prema-mreži” (2)

- Ovde se, gledano na najnižem nivou, dobija zadatak da se preko komunikacionog medijuma prenese sekvenca bitova
- U tom najnižem nivou komunikacije se proučava mehanizam slanja pojedinačnih bitova od jednog do drugog uređaja kroz komunikacioni medijum
- Najniži nivo komunikacije karakteriše potreba za velikom efikasnošću
- Način komunikacije na tom najnižem nivou zavisi od tipa komunikacionog medijuma - žičana ili bežična veza, koja vrsta kablova je u pitanju i sl.
- U okviru lokalne mreže komunikacija se zasniva na tehnologijama:
 - Ethernet (žičano povezivanje)
 - Wi-Fi (bežično povezivanje)
- Brzina prenosa podataka u ovakvim mrežama veća od 1Gbps



Sloj „host-prema-mreži” (3)

- Ovaj sloj od uređaja koji rade na međumrežnom sloju dobija zadatak da se paket (u IP terminologiji, taj paket se naziva **IP datagram**) prenese:
 - sa jednog rutera na drugi
 - sa jednog uređaja na drugi u okviru lokalne mreže
- Taj zadatak se realizuje tako što se IP datagram se obmotava dodatnim podacima i kreiraju se **okviri** (frame)





Sloj „host-prema-mreži” (4)

Potrebno je sprečiti izmenu podataka prilikom mrežnog prenosa (preskakanje bitova, izmena bitova, ponavljanje, ...)

- Na kraj okvira dodaje se sekvenca za proveru okvira:
 - omogućava primaocu da proveri da li je došlo do greške
 - neke greške se mogu ispraviti
- Moguće je detektovati i ispraviti složenije greške korišćenjem sekvenci od više bitova, kodiranih kodovima za otkrivanje i ispravljanje grešaka
- Na ovom sloju koriste se MAC adrese
 - Predstavljaju se pomoću 48 bita
 - Zapisuju se u obliku 6 dvocifrenih heksadekadnih brojeva (primer: 2c:d4:44:a8:be:3b)
- Na početak okvira dodaju se MAC adresa primaoca i pošiljaoca
- Ako se u okviru nalaze IP datagrami, tada okvir sadrži i IP adrese primaoca i pošiljaoca, ali one se na ovom nivou ne analiziraju



Slojevi kod računarskih mreža međumrežni sloj



Protokoli i slojevi

OSI sloj	TCP/IP sloj	Jedinica	Protokol
aplikativni sloj (mrežni procesi vezani za aplikaciju)	aplikativni sloj	podatak	HTTP, FTP, Telnet, DNS, DHCP, POP/SMTP, NNTP
sloj prezentacije (enkripcija i kodiranje podataka)		podatak	MIME, TLS, SSL
sloj sesije (uspostavljanje sesije krajnjih korisnika)		podatak	SSH, Named Pipes, PPTP
transportni sloj (veza, pouzdanost, transport)	transpotni sloj	segment datatgram	TCP, UDP, SCTP, DCCP
mrežni sloj (logičko adresiranje i rutiranje)	međumrežni sloj	paket	IP (IPv4 , IPv6), ICMP, ARP , RARP
sloj veze podataka (fizičko adresiranje, pristup medijumu)	sloj pristupa mreži	okvir	PPP, HDLC, Frame Relay
fizički sloj (prenos signala)		bit	Token Ring, RS-232, T1, E1, POTS, OTN, DSL, 802.11a/b/g/n PHY, 802.15.x PHY, Ethernet, USB, Bluetooth, Firewire (IEEE 1394)



Međumrežni sloj

Međumrežni sloj (internet layer) - bavi se povezivanjem više računara u mrežu

- Osnovni zadatak u okviru ovog sloja je **rutiranje** (routing), tj. određivanja putanja paketa koji putuju kroz mrežu kako bi se odredio efikasan način da stignu na svoje odredište
 - Kako bi se odredila putanja, neophodno je uvođenje sistema adresiranja
- Ukoliko se povezuju heterogene mreže (sa različitim shemama adresiranja), na ovom sloju se vrši prevođenje adresa
 - Na primer, na nižim slojevima se obično koriste fizičke MAC adrese, a na višim IP adrese
- Svaki čvor u mreži uključen u komunikaciju mora da implementira mrežni protokol, da razume odredišnu adresu i da na osnovu ovoga odluči kome će da prosledi primljenu poruku
 - Najpoznatiji protokol ovog sloja je koji se koristi u okviru Interneta je **Internet Protocol (IP)**



IP protokol međumrežnog sloja

- Internet protokol (Internet Protocol - IP) je protokol koji se koristi za komunikaciju u okviru mrežnog sloja Interneta
- Dve osnovne verzije ovog protokola su IPv4 i IPv6
- Iz istorijskih razloga i veće preglednosti u nastavku će detaljnije biti opisana IPv4 verzija IP protokola
- Osnovni zadatak ovog protokola je da pokuša da dopremi (tj. rutira) paket od izvora do odredišta u okviru mreže sa paketnim komutiranjem, isključivo na osnovu navedene adrese, bez obzira da li su izvor i odredište u okviru iste mreže ili između njih postoji jedna ili više drugih mreža
- Protokol ne daje nikakve garancije da će paketi zaista i biti dopremljeni, ne daje garancije o ispravnosti dopremljenih paketa, ne garantuje da će paketi biti dopremljeni u istom redosledu u kojem su poslani i slično
 - Garancije ovog tipa obezbeđuju se na višim slojevima komunikacije



IP protokol međumrežnog sloja (2)

- Pri prosleđivanju paketa sa transportnog sloja na ovaj sloj dodaju se:
 - адреса pošiljaoca,
 - адреса primaoca, ...
- IP datagram - ide od pošiljaoca do primaoca, preko serije rutera





Hijerarhijska struktura IP adresa

- IP adrese su strukturirane **hijerarhijski**: adresa se deli na bitove koji adresiraju mrežu (vodeći) i bitove koji adresiraju uređaj u okviru mreže
- Paket se dostavlja:
 - korišćenjem lokalnog mrežnog saobraćaja
 - šalje se van mreže "u svet" - preko određenog rutera koji se naziva **izlazna kapija** (gateway)
- Svi uređaji iz iste mreže dele zajednički početak IP adrese
 - Primer: od 147.91.67.0 do 147.91.67.255 - ista prva 24 bita, razlikuju se poslednjih 8



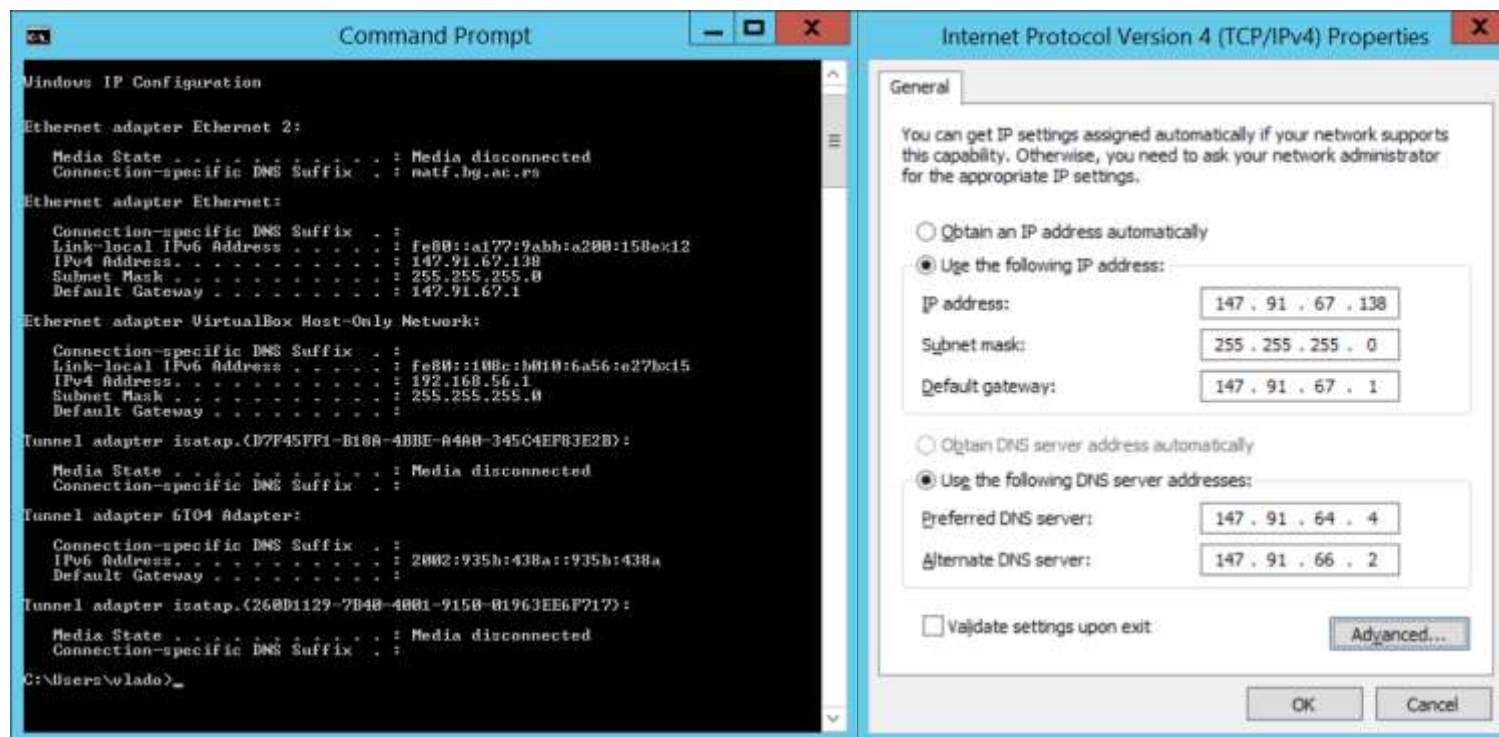
Hijerarhijska struktura IP adresa (2)

- Ranije su IP adrese bile deljene na klase (A, B, C, D, E) i svaka klasa je definisala broj bita za prvi i broj bita za drugi deo deo IP adrese.
 - Adrese klase A (prvi bit u zapisu je 0 - između 0.0.0.0 i 27.255.255.255) su bile dodeljivane jako velikim mrežama (8+24 bita - 128 mreža sa mogućih preko 16.7 miliona korisnika)
 - Adrese klase B (počinje sa 10 - između 128.0.0.0 i 191.255.255.255) su bile dodeljivane srednjim mrežama (16+16 bita - preko 16 hiljada mreža sa mogućih 65536 korisnika)
 - Adrese klase C (počinje sa 110 - između 192.0.0.0 i 223.255.255.255) su bile dodeljivane malim mrežama (24+8 bita - preko dva miliona mreža sa mogućih 256 korisnika).
- Vremenom se pokazalo da ovakva organizacija nije skalabilna
 - Obično su mreže kompanija imale potrebu za više od 256 uređaja, tako su uzimale adrese klase B, pa je veliki broj adresa ostajao nedodeljen



Hijerarhijska struktura IP adresa (3)

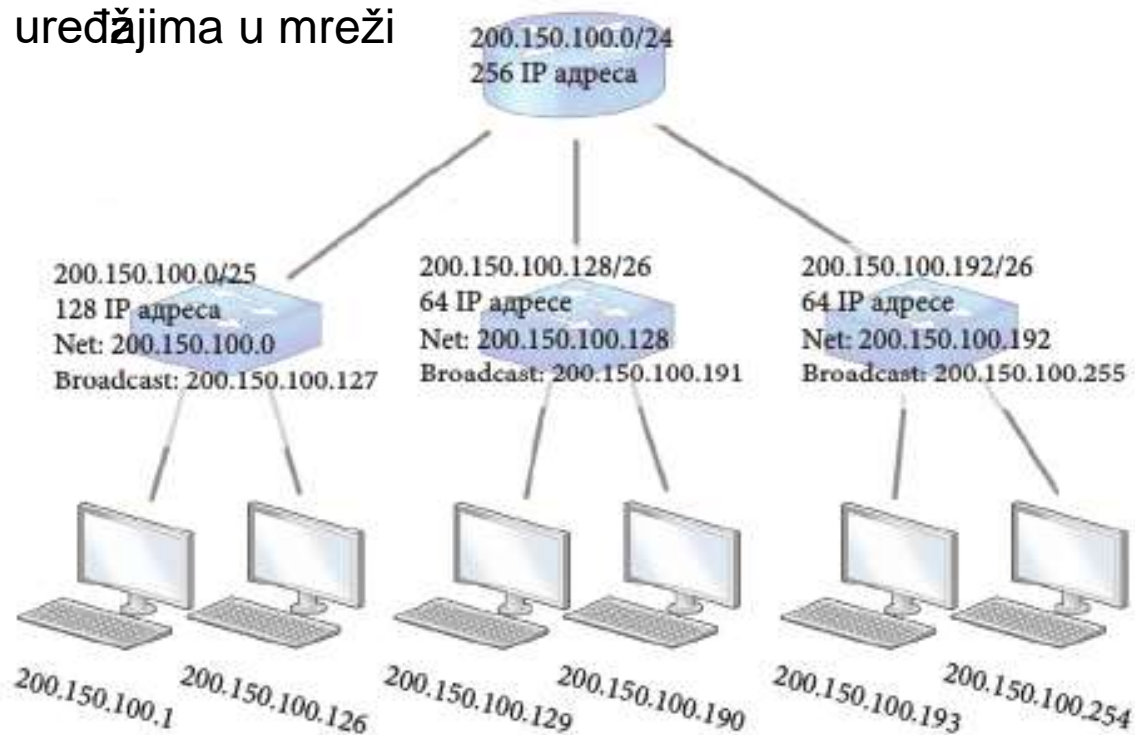
- Dva načina zapisa skalabilnog zapisa IP adresa:
 - CIDR notacija - adresa 147.91.67.138/24
 - Maska podmreže (subnet mask) - uz adresu 147.91.67.138 navodi se maska podmreže 255.255.255.0 (24 jedinice i 8 nula)





Povezivanje uređaja u lokalnoj mreži

- U okviru svake mreže postoje dve adrese sa specijalnom namenom:
 - prva adresa (250.150.100.0) smatra se adresom mreže
 - poslednja adresa (250.150.100.255) - adresa za javno emitovanje (broadcast address) - svaka poruka poslata na tu adresu dostavlja se svim uređajima u mreži

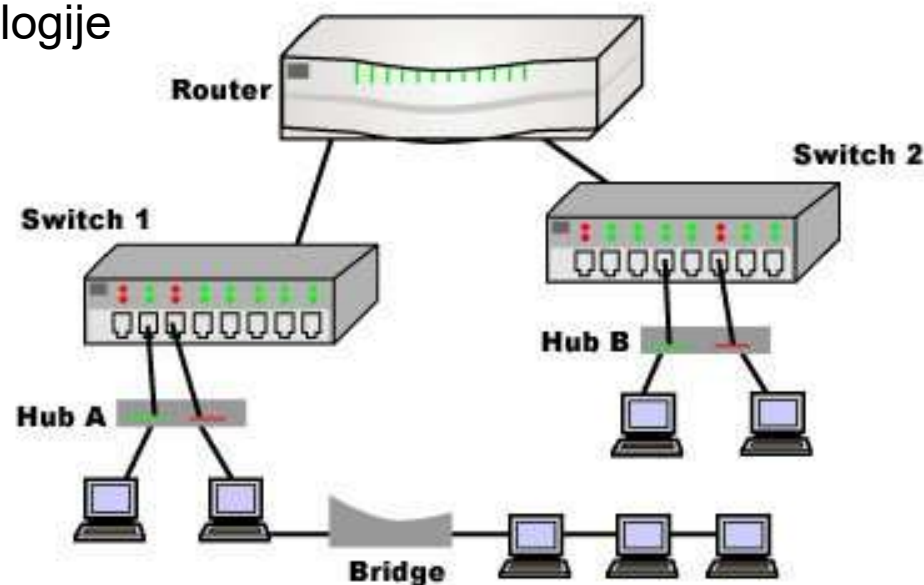




Povezivanje uređaja u lokalnoj mreži (2)

Elementi mrežnog hardvera koji se koriste:

- **Ruter** (router) - kompleksniji uređaj namenjen povezivanju raznorodnih mreža i povezivanju mreža sa Internetom
 - Obično ima javnu IP adresu koju deli cela mreža
 - Koristi IP adrese za prosleđivanje paketa, što dopušta mrežnu komunikaciju po različitim protokolima
 - Prosleđuje pakete na osnovu softvera, dok svič radi hardverski
 - Podržava različite WAN tehnologije
 - Radi na međumrežnom sloju





Povezivanje uređaja u lokalnoj mreži (3)

Elementi mrežnog hardvera koji se koriste:

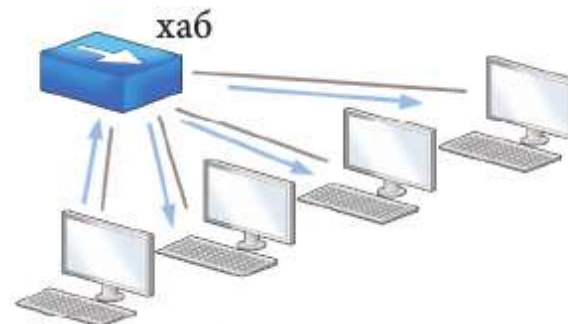
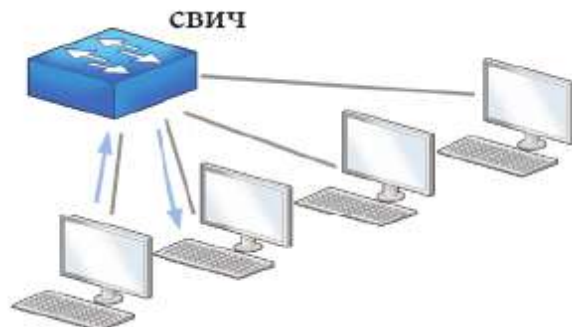
- Svič (switch) - povezuje dve ili više nezavisnih mreža
 - Postavljanjem sviča između povezanih uređaja poruka se prosleđuje samo uređaju kome je namenjena (efikasnija komunikacija)
 - Svič čuva tabelu koja preslikava MAC adrese priključenih uređaja na redne brojeve priključaka
 - Tabela se gradi i održava automatski tokom komunikacije
 - Podržava veći broj ulaznih i izlaznih portova
 - Vršiti kontrolu greške pre prosleđivanja paketa
 - U zavisnosti od tipa, realizuju prosleđivanje na nivou „host-prema-mreži“ (zasnovano na MAC adresama) i na međumrežnom nivou (zasnovano na IP adresama)
 - Pakete prosleđuje samo mreži u kojoj se nalazi primalac
 - Kod velikih mreža se svičevi koriste umesto habova za povezivanje računara u podmrežama



Povezivanje uređaja u lokalnoj mreži (4)

Elementi mrežnog hardvera koji se koriste:

- **Hab** (hub) - dobijene poruke prosleđuje svim priključenim uređajima
 - Postavljanje haba između povezanih uređaja - primljeni paketi se prosleđuju svim uređajima povezanim na njega (jednostavno, ali je verovatnoća sudara velika)
 - Ne može kontrolisati propuštanje paketa koje šalje povezanim uređajima
 - Ne može odrediti najbolji put za slanje paketa
 - Nisu efikasni
 - Koriste se u malim mrežama, sa niskim nivoom komunikacije
 - Radi na nivou sloja „host-prema-mreži“ – nisko, najbliže fizičkom sloju





Povezivanje uređaja u lokalnoj mreži (5)

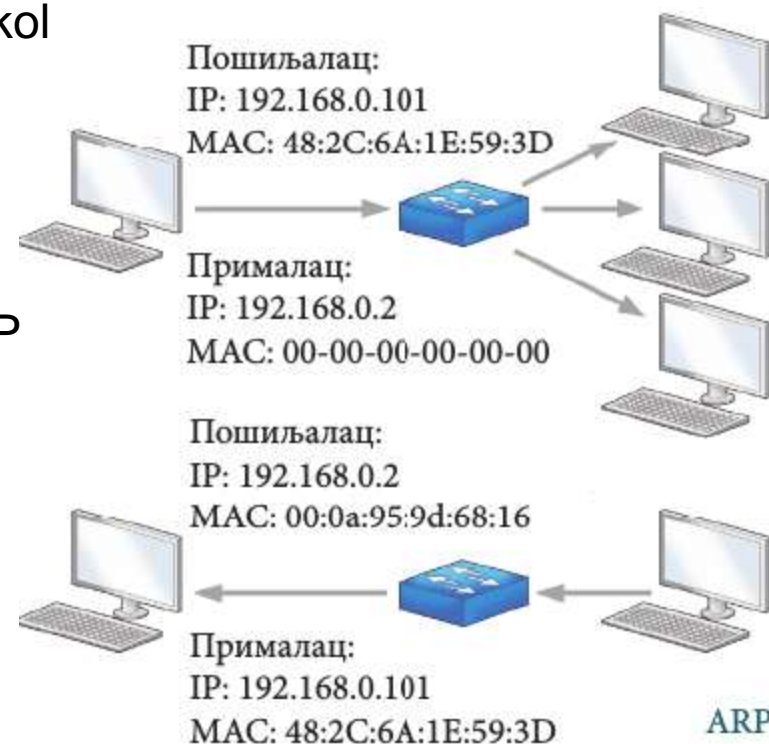
Elementi mrežnog hardvera koji se koriste:

- **Most** (bridge) - povezuje lokalnu mrežu sa drugom lokalnim mrežom koja koristi isti protokol
 - Ima jedinstveni ulazni i jedinstveni izlazni port
 - Kontrolise propuštanje paketa na mreži na osnovu MAC adrese odredišta – ne šalje sve pakete bez kontrole
 - Pakete prosleđuje samo mreži u kojoj se nalazi primalac
 - Radi na nivou sloja „host-prema-mreži“



Povezivanje uređaja u lokalnoj mreži (6)

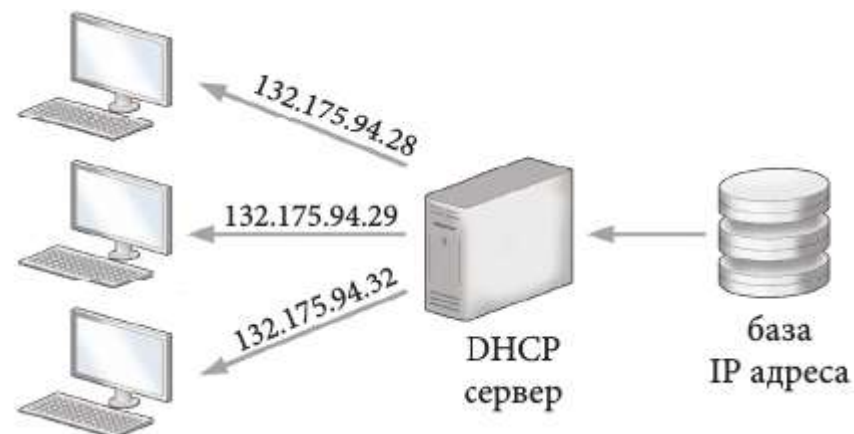
- Kako uređaj koji zna IP adresu primaoca određuje MAC adresu na koju prosleđuje IP datagram?
 - na osnovu mrežne maske utvrđuje da li je primalac u istoj mreži; ako jeste šalje njemu, ako nije šalje izlaznoj kapiji
 - u oba slučaja zna IP adresu uređaja u lokalnoj mreži
 - za dobijanje adrese koristi se protokol razrešavanja adresa (address resolution protocol, ARP)
 - javno se emituje ARP zahtev sa IP adresom
 - uređaj sa tom IP adresom šalje ARP odgovor sa svojom MAC adresom





IP adrese i DHCP

- Dinamičke IP adrese se dodeljuju korišćenjem specijalizovanog protokola za dinamičku konfiguraciju (Dynamic Host Configuration Protocol - **DHCP**)
- Specijalizovani server (tzv. DHCP server) je zadužen za skup IP adresa koje određuje administrator mreže i na zahtev uređaja koji se priključuje na mrežu dodeljuje mu neku u tom trenutku slobodnu adresu
- Server se može konfigurisati tako da dodeljuje bilo koju slobodnu IP adresu, ili uvek istu adresu koja se određuje na osnovu MAC adrese uređaja koji zahteva IP adresu, i slično





Javne i privatne IP adrese

- Da ne bi došlo do nestašice IPv4 adresa uvode se privatne adrese:
 - 10.0.0.0/8 (od 10.0.0.0 do 10.255.255.255) - 16.7 miliona adresa
 - 172.16.0.0/12 (od 172.16.0.0 do 172.31.255.255) - milion adresa
 - 192.168.0.0/16 (od 192.168.0.0 do 192.168.255.255) - 65536 adresa
- Privatne adrese se koriste samo za lokalnu mrežnu komunikaciju
- Prilikom pristupa Internetu:
 - ruter (izlazna kapija) menja lokalnu adresu svojom (javnom) adresom
 - primalac odgovor šalje nazad ruteru, a on menja adresu privatnom adresom uređaja koji je poslao zahtev i prosleđuje odgovor





Javne i privatne IP adrese (2)

- Ovaj proces se naziva preslikavanja mrežnih adresa (network address translation - NAT)
- Korišćenje NAT-a prilikom slanja paketa:
 - U slučaju da ruter detektuje odredišnu adresu iz opsega adresa privatne mreže sa kojom je povezan, jasno je da je paket namenjen za lokalnu komunikaciju i šalje se jedinstvenom uređaju sa navedenom lokalnom adresom
 - Ako je odredišna adresa javna, ruter adresu pošiljaoca zamenjuje svojom adresom (globalno jedinstvenom) i paket prosledjuje na odredište.



Javne i privatne IP adrese (3)

- Korišćenje NAT-a prilikom prijema paketa:
 - U slučaju dolaznog paketa, nije odmah jasno na koju privatnu adresu je potrebno poslati paket koji je pristigao
 - Kako bi se ovo razrešilo, lokalna adresa se pakuje i postaje sastavni deo paketa koji se šalje
 - Ruter, pre prosledivanja dolaznog paketa, vrši njegovo raspakivanje i određivanje lokalne adrese
- Sve ovo narušava osnovne principe i koncepte IP protokola, pa se zato NAT smatra prelaznim rešenjem problema nestašice IP adresa, dok ne zaživi IPv6



Rutiranje

- U većim mrežama postoji veliki broj povezanih rutera
- Uloga rutera: na osnovu IP adrese primaoca i na osnovu tabela koje su zapisane u njihovoj memoriji (tabela rutiranja) odrediti kome od povezanih čvorova treba proslediti paket da bi efikasno stigao do cilja
- Tabele rutiranja sadrže spisak mrežnih adresa različitog nivoa hijerarhije i za svaku od njih kom uređaju treba dostaviti paket
- Primer: Neka je u tabeli rutiranja rutera

```
0.0.0.0/0 via 200.170.10.10  
200.0.0.0/8 via 200.100.5.20  
200.160.0.0/16 is directly connected, Serial0/1
```

- Ako ruter primi paket namenjen adresi 200.150.100.23, on se dostavlja preko rutera 200.100.5.20
- Šablonom 0.0.0.0/0 zadaje se gde proslediti paket ako adresa nije prepoznata na neki drugi način
- Traži se najpreciznije poklapanje sa šablonom - poklapanje sa najvećim brojem bitova



Rutiranje (2)

- Kvalitet rutiranja zavisi od tabela rutiranja
- Tabele rutiranja se mogu graditi statički ili dinamički



Slojevi kod računarskih mreža aplikativni sloj HTTP protokol



Protokoli i slojevi

OSI sloj	TCP/IP sloj	Jedinica	Protokol
aplikativni sloj (mrežni procesi vezani za aplikaciju)	aplikativni sloj	podatak	HTTP , FTP, Telnet, DNS, DHCP, POP/SMTP, NNTP
sloj prezentacije (enkripcija i kodiranje podataka)		podatak	MIME, TLS, SSL
sloj sesije (uspostavljanje sesije krajnjih korisnika)		podatak	SSH, Named Pipes, PPTP
transportni sloj (veza, pouzdanost, transport)	transportni sloj	segment datatgram	TCP, UDP, SCTP, DCCP
mrežni sloj (logičko adresiranje i rutiranje)	međumrežni sloj	paket	IP (IPv4, IPv6), ICMP, ARP, RARP
sloj veze podataka (fizičko adresiranje, pristup medijumu)	sloj pristupa mreži	okvir	PPP, HDLC, Frame Relay
fizički sloj (prenos signala)		bit	Token Ring, RS-232, T1, E1, POTS, OTN, DSL, 802.11a/b/g/n PHY, 802.15.x PHY, Ethernet, USB, Bluetooth, Firewire (IEEE 1394)



Aplikacioni sloj

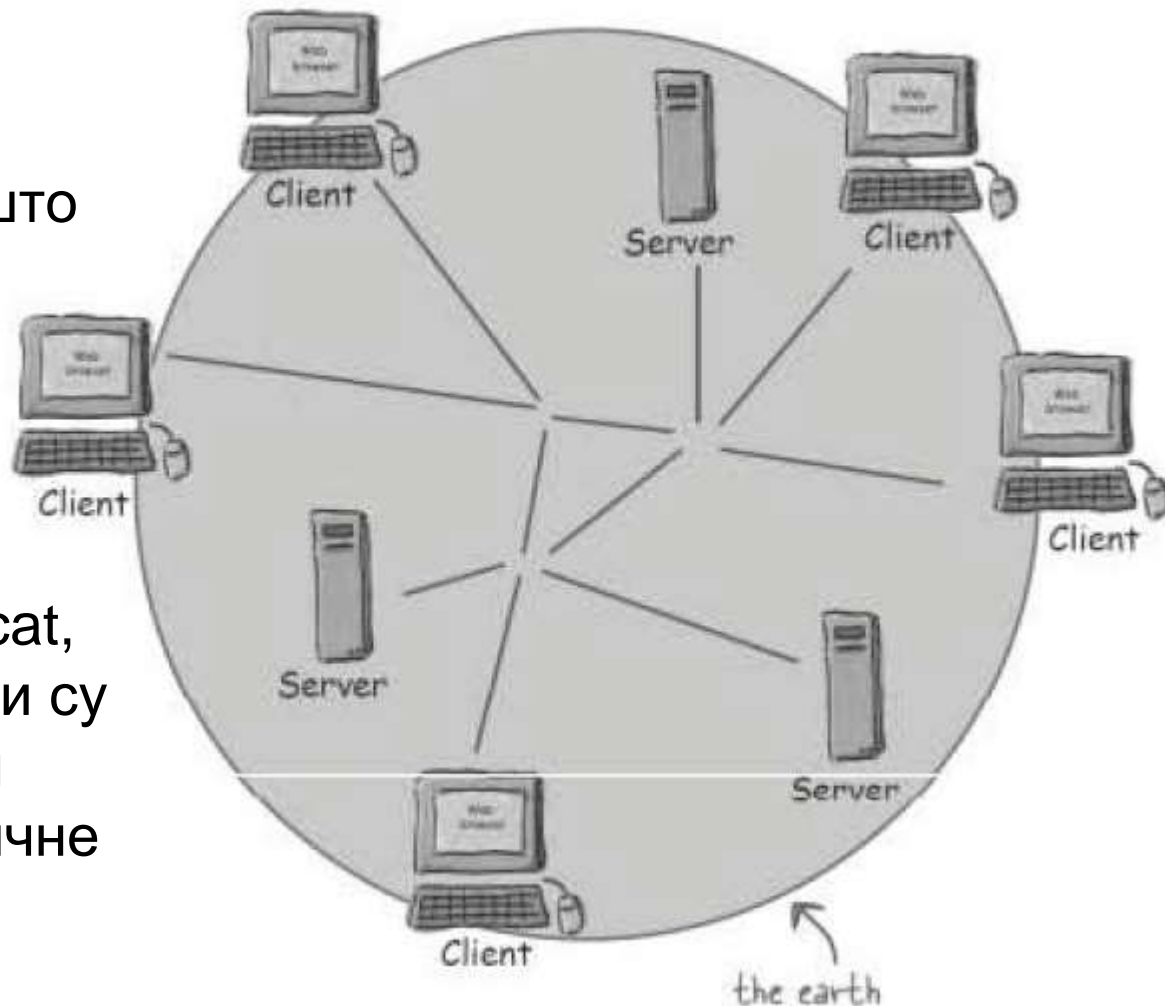
Aplikacioni sloj (application layer) - definiše protokole koje direktno koriste korisničke aplikacije u okviru svoje komunikacije

- Ovi protokoli su prilagođeni specifičnim zahtevima aplikacija
- Aplikacioni protokoli u protokoli kojima dva programa tj. dve aplikacije komuniciraju
- Najkorišćeniji protokoli ovoga sloja u okviru Interneta su
 - HyperText Transfer Protocol (**HTTP**) koji se koristi za prenos veb stranica
 - **SMTP**, **POP3**, **IMAP** koji se koriste u za prenos elektronske pošte
 - File Transfer Protocol (**FTP**) koji se koristi za prenos datoteka
 - itd.



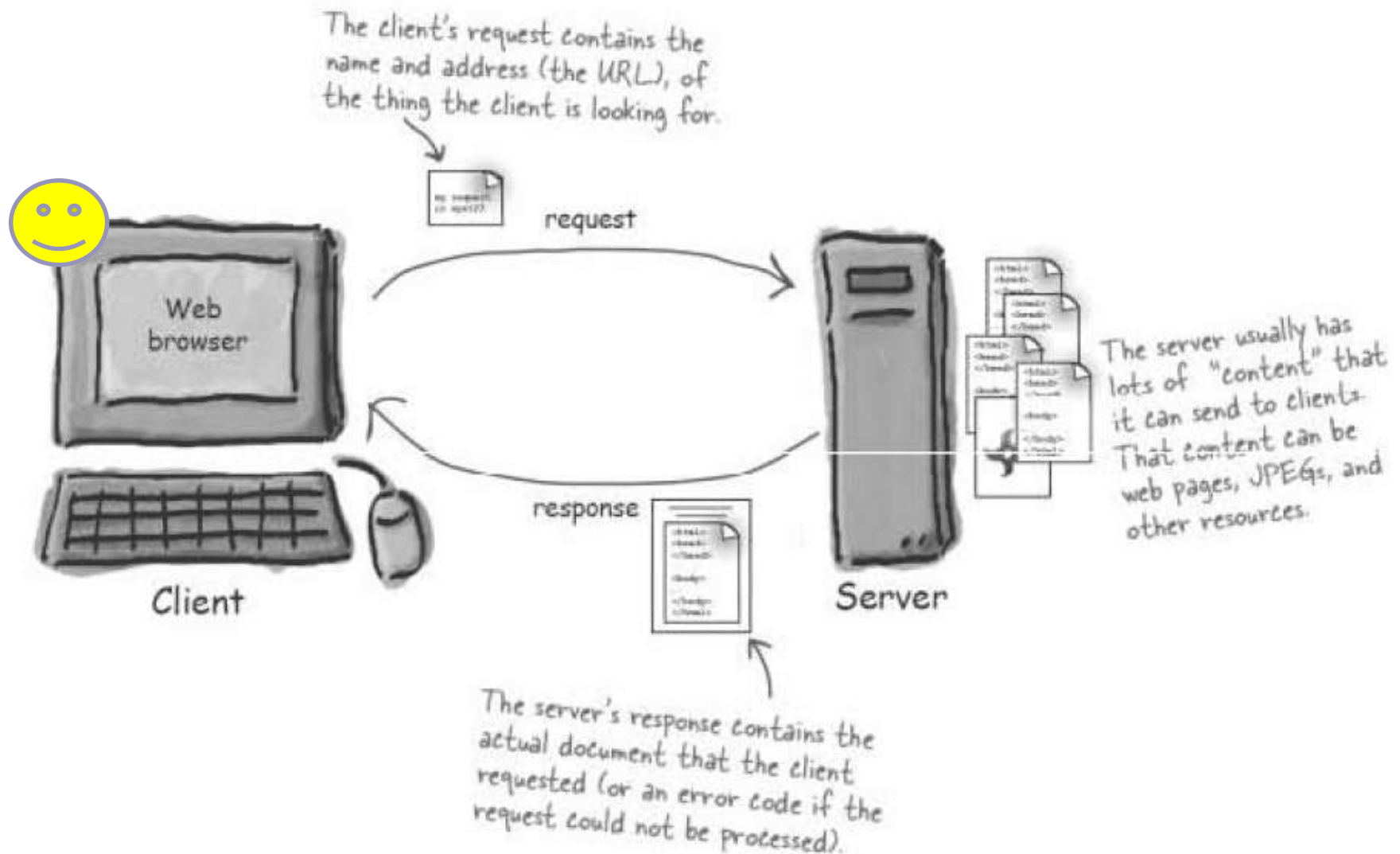
HTTP протокол и веб

- Веб се састоји од огромног броја клијената са прегледачима (као што су Chrome, Mozilla, Yandex, Safari итд.) и од сервера (који користе веб сервере као што су Apache, JBoss, Tomcat, Microsoft IIS итд.) који су међусобно повезани кроз жичане и безжичне мреже.





Функционисање веб сервера



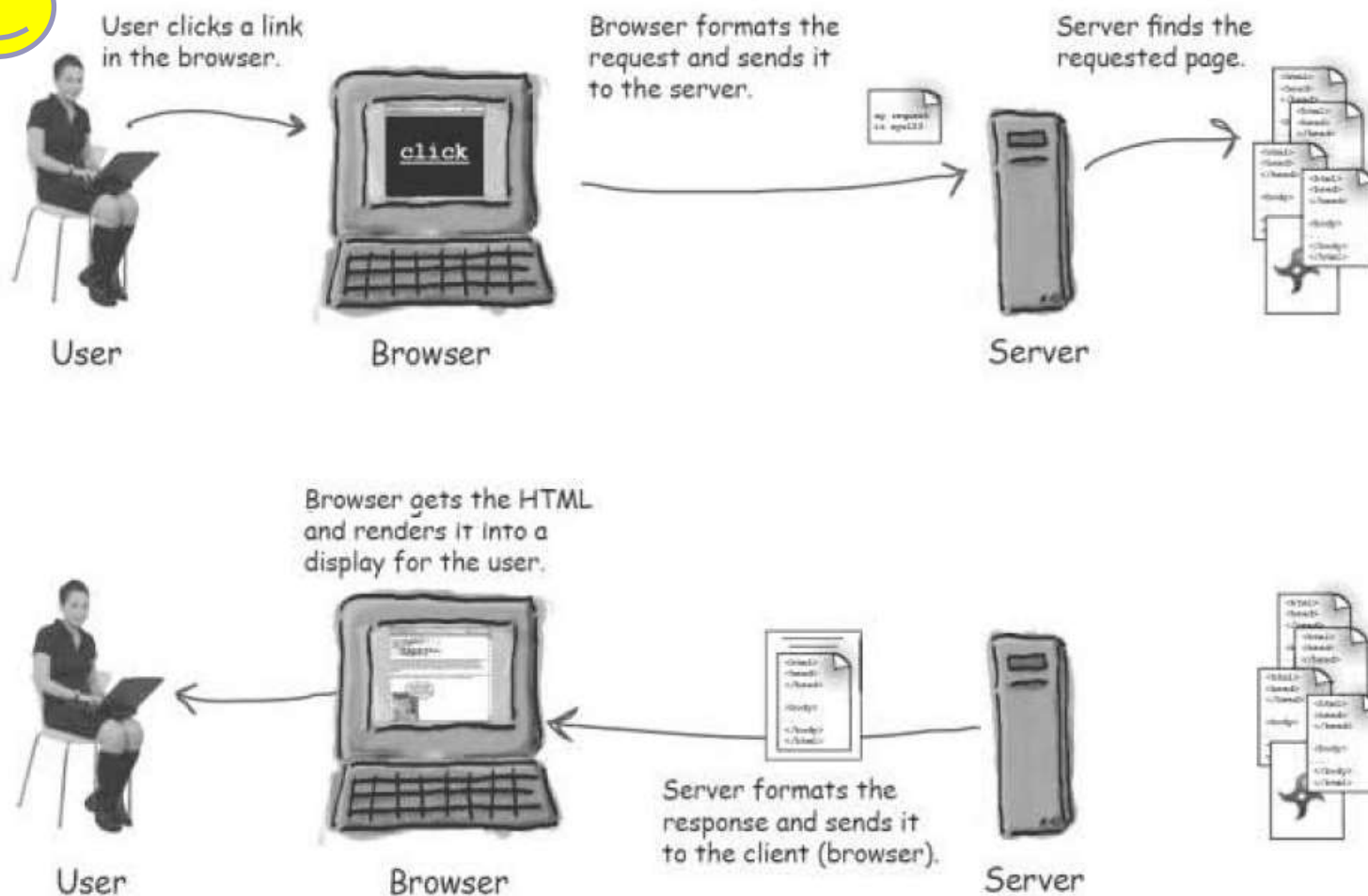


Функционисање веб сервера (2)

- Корисник преко веб прегледача шаље захтев за ресурсом
- Веб сервер прихвата захтев, проналази тражени ресурс и њега шаље кориснику
 - Ресурс може бити HTML страна, слика, PDF документ или нешто четврто – што год да је у питању, клијент захтева ресурс, а сервер шаље клијенту ресурс који је захтеван
 - У случају када нема захтеваног ресурса, генерише се грешка „404 Not found“
- Термин „сервер“ означава и сам рачунар (тј. хардвер) и програм који представља веб сервер (тј. софтвер)
 - Ако из контекста није јасно да ли се ради о хардверу или софтверу, то ће бити додатно истакнуто



Функционисање веб клијента





Функционисање веб клијента (2)

- Када се говори о клијенту, има се у виду корисник, али и веб прегледач – апликација коју корисник користи за сурфовање. Корисник преко веб прегледача шаље захтев за ресурсом
 - Прегледач је софтвер (нпр. Netscape, Chrome, Mozilla, Yandex, Safari, Edge, Opera и сл.) који комуницира са веб сервером. Осим послова комуникације, прегледач треба да интерпретира HTML код и да исцрта веб стране за корисника
 - Клијент може бити и ма који други програм кји комунира са сервером коришћењем HTTP протокола (curl, postman,...)
- Ако се експлицитно не наведе другачије, надаље ће термин „клијент“ ће обухватити и софтвер (тј. прегледач) и човека (тј. корисника)
 - Другим речима, клијент је апликација - прегледач која обавља оно што корисник захтева да се уради



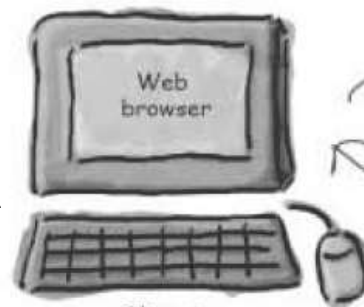
Клијент и сервер користе HTML и HTTP

- Када сервер шаље одговор на захтев, он прегледачу обично шаље садржај датог типа, тако да прегледач може да прикаже добијени одговор.
- Често је одговор који сервер шаље клијенту секвенца знакова који представљају документ у **HTML** формату. Тај HTML документ потом бива приказан од стране прегледача
 - Језик за означавање HTML (прецизније, HTML 5), је описан у претходним предавањима
- Највећи део конверзације између клијената и сервера се реализује коришћењем **HTTP** протокола
 - Клијент тада шаље HTTP захтев, а сервер одговара са HTTP одговором.
 - Када сервер пошаље HTML страну клијенту, он то ради коришћењем HTTP протокола.



HTTP протокол

- HTTP протокол се извршава преко TCP/IP протокола
- То је мрежни протокол са карактеристикама које се односе на веб, али он се ослања на TCP/IP протокол ради обезбеђења потпуног преноса захтева и одговора са једног места на друго
- Суштина HTTP конверзације је једноставна секвенца захтев/одговор: прегледач **захтева** а сервер **одговара**



Client

HTTP request

HTTP response



Server

Key elements of the **request** stream:

- ▶ HTTP method (the action to be performed)
- ▶ The page to access (a URL)
- ▶ Form parameters (like arguments to a method)

Key elements of the **response** stream:

- ▶ A status code (for whether the request was successful)
- ▶ Content-type (text, picture, HTML, etc.)
- ▶ The content (the actual HTML, image, etc.)



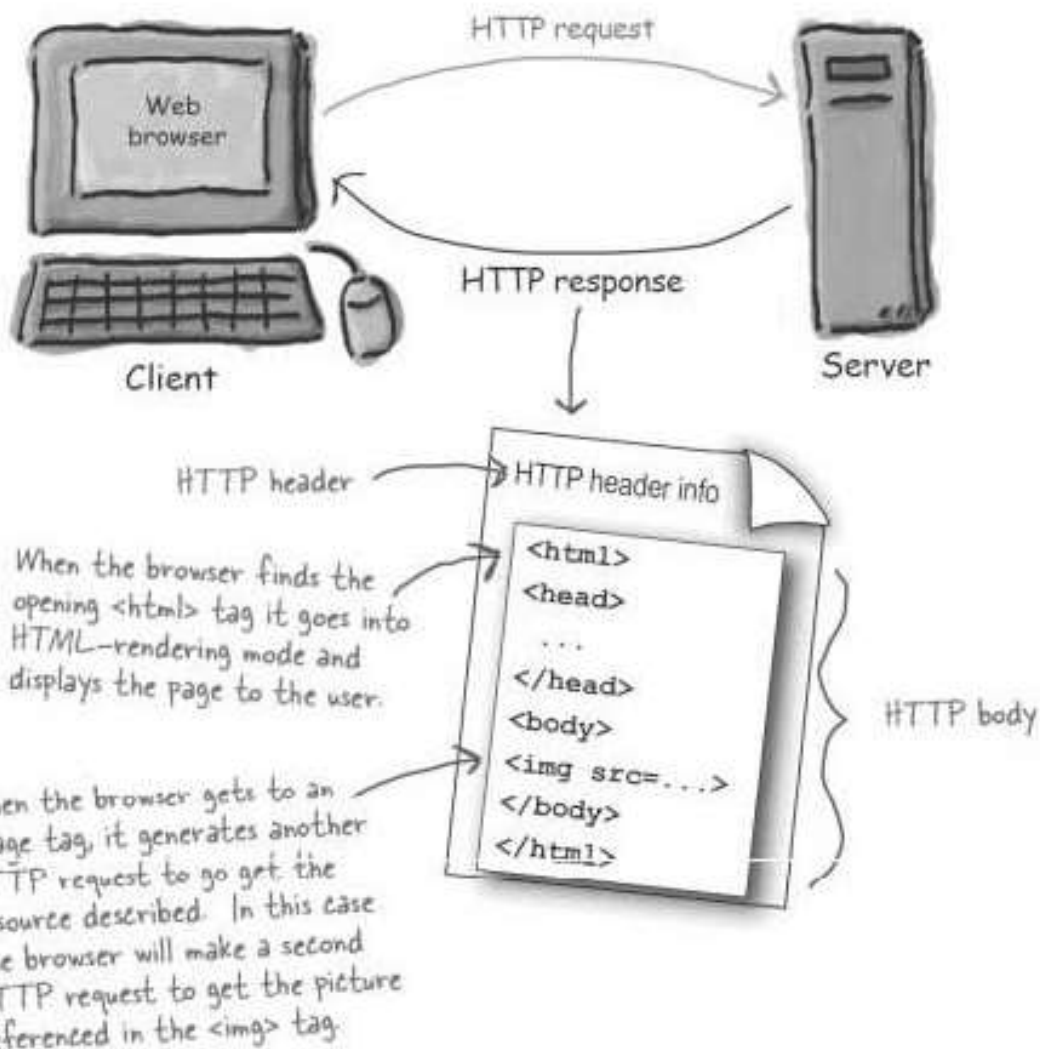
HTTP протокол (2)

- Сам HTTP протокол је описан IETF документом RFC 2616
- Веб сервери Apache, JBoss, Tomcat, Microsoft IIS и сл. су примери сервера који обрађују HTTP захтеве
- Прегледачи Netscape, Chrome, Mozilla, Yandex, Safari, Edge, Opera и сл. обезбеђују кориснику да генерише HTTP захтев, упути га према серверу и на адекватан начин прикаже HTTP одговор који добије од сервера
-
- Карактеристике HTTP протокола:
 1. HTTP не одржава конекцију (connectionless)
 2. HTTP је независтан од медијума (media independent)
 3. HTTP не подржава стања (stateless)



HTML је део HTTP одговора

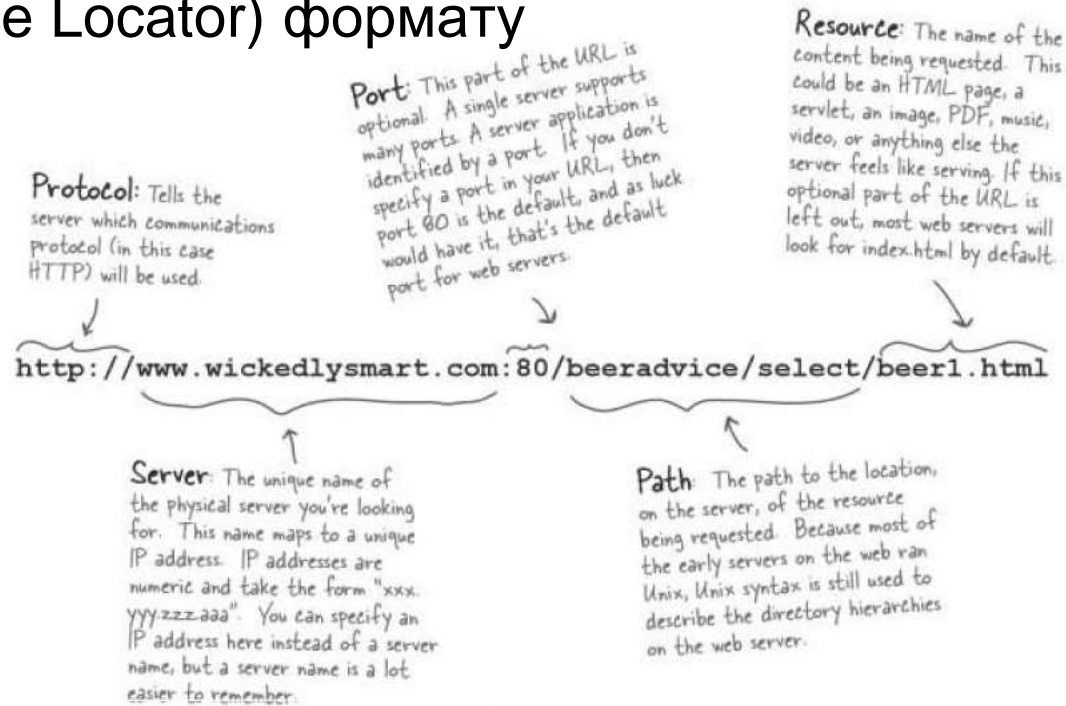
- HTTP одговор може садржавати HTML
- HTTP додаје информације о заглављу на почетак садржаја који се враћа као одговор, какав год садржај био у питању
- Прегледач користи информације из заглавља као помоћ у процесирању HTML садржаја
- Дакле, HTML се може посматрати као садржај уметнут у HTTP одговор





URL

- Сваки ресурс на вебу има своју јединствену адресу, у URL (Uniform Resource Locator) формату



Not shown:

Optional Query String

Remember, if this was a GET request, the extra info (parameters) would be appended to the end of this URL, starting with a question mark "?", and with each parameter (name/value pair) separated by an ampersand "&".



HTTP метод

- HTTP захтев садржи назив метода у свом заглављу
- Назив метода говори серверу о каквој се врсти захтева ради и како ће бити форматиран остатак поруке
- HTTP протокол подржава следеће методе:
 - GET - користи се за преузимање информација са датог сервера на основу дате адресе. Захтеви који користе метод GET треба само да прибављају податке, а никако не треба да их мењају
 - HEAD - је врло сличан GET методу, са тим што се тело поруке не враћа клијенту (враћа се само статусна линија и заглавље). Метод се може користити ради утврђивања да ли је линк измењен у односу на претходно стање - измењено стање се тестира упоређивањем информација послатих у заглављу захтева са информацијама из заглавља генерисаног одговора
 - POST - се користи за захтев да се пошаљу подаци HTTP серверу коришћењем HTML форме



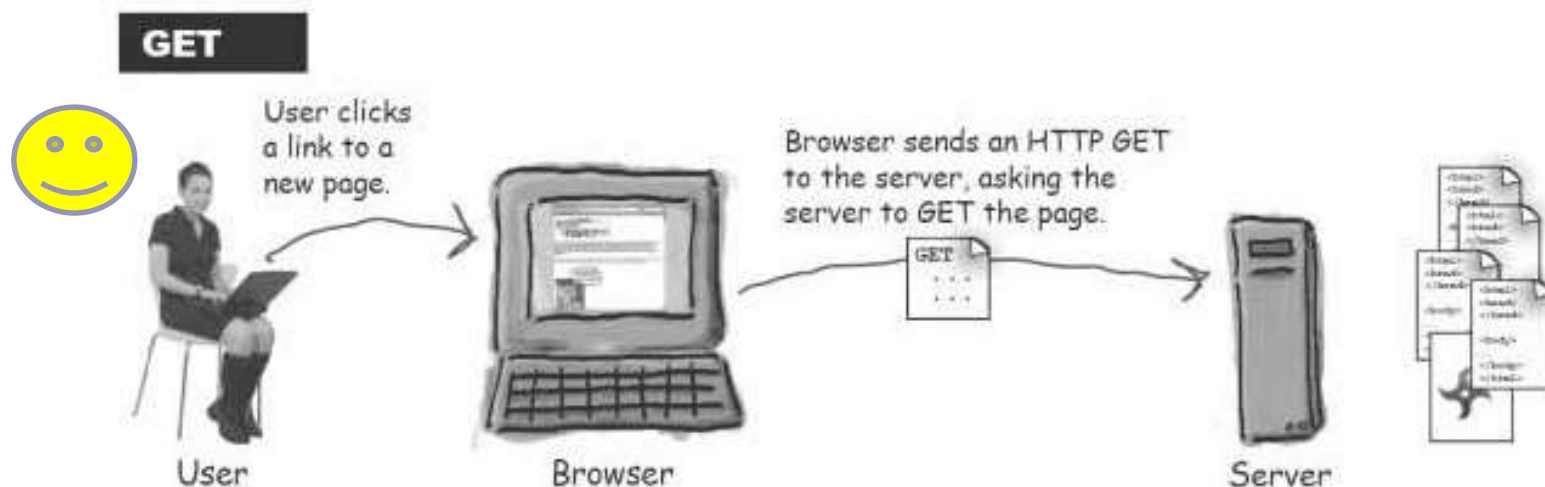
HTTP метод (2)

- PUT - користи се за захтев HTTP серверу да се подаци послати у оквиру захтева сместе на месту наведеног ресурса
- DELETE - користи се за захтев серверу да се уклони наведени ресурс
- TRACE – извршава тестирање повратне поруке дуж путање којом се захтев креће према циљном ресурсу
- OPTIONS – описује опције комуникације за циљни ресурс
- CONNECT – обезбеђује тунелску комуникацију према серверу одређеним са датом адресом



Методи GET и POST

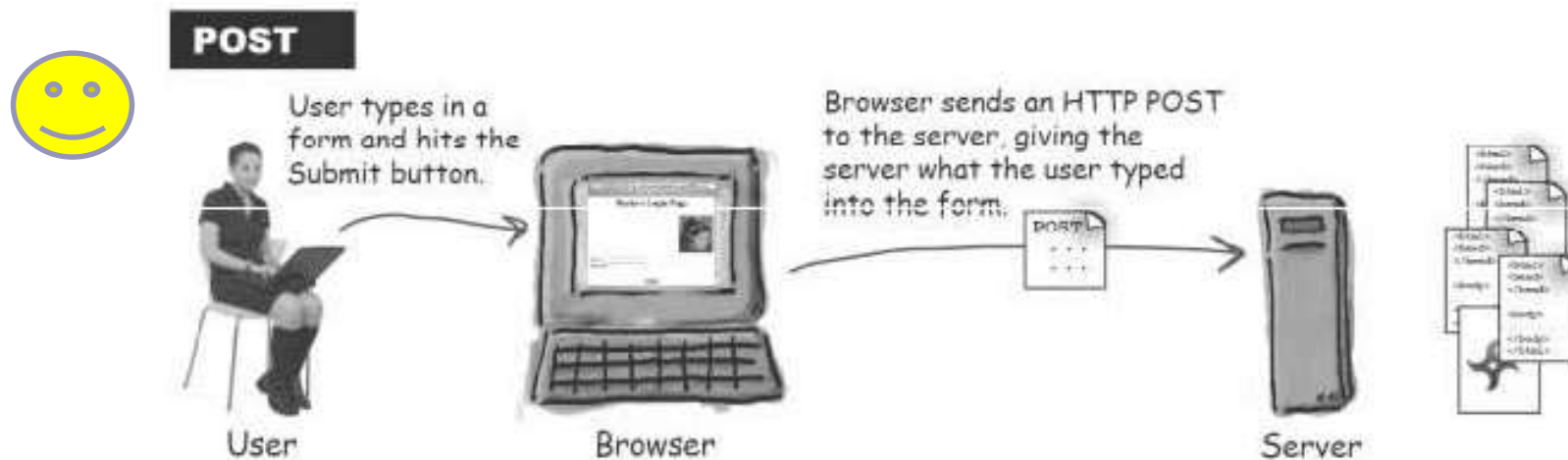
- Метод GET је најједноставнији HTTP метод
- Метод GET тражи од сервера да прибави ресурс и да га врати позиваоцу
- Ресурс може бити HTML страна, PDF документ, JPG слика ...
- Сврха метода GET је да се добије ресурс од сервера





Методи GET и POST (2)

- Метод POST је моћнији од метода GET
- Коришћењем овог метода може се захтевати нешто од и истовремено слати податке на сервер (па сервер може процесирати прispеле податке)





Методи GET и POST (3)

- Подаци се могу слати на сервер и помоћу метода GET и помоћу метода POST
 - Укупан број знакова који се помоћу могу послати метода GET је много мањи од броја знакова који се могу послати преко метода POST, а то горње ограничење зависи од врсте веб сервера и прегледача
 - Подаци који се шаљу коришћењем метода GET се налепљују на адресу у адресној линији прегледача, па је све што се тим путем шаље на сервер директно видљиво кориснику (и самим тим подложније манипулацији)
 - Корисник не може поставити маркер на страницу где је садржај форме прослеђен методом POST, а може ако је за прослеђивање коришћен метод GET

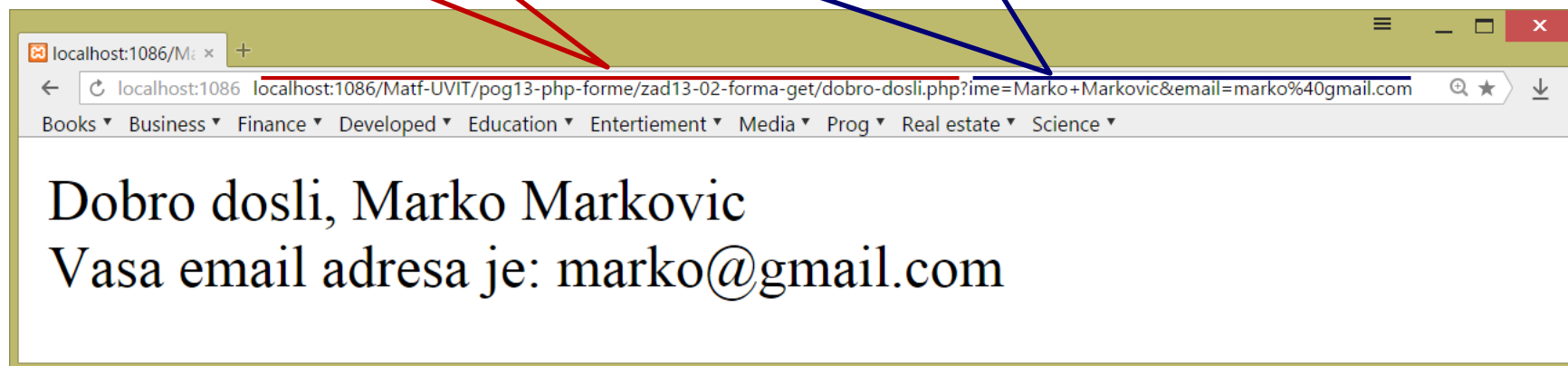


Методи GET и POST (4)

- Илустрација слања података на сервер и помоћу метода GET

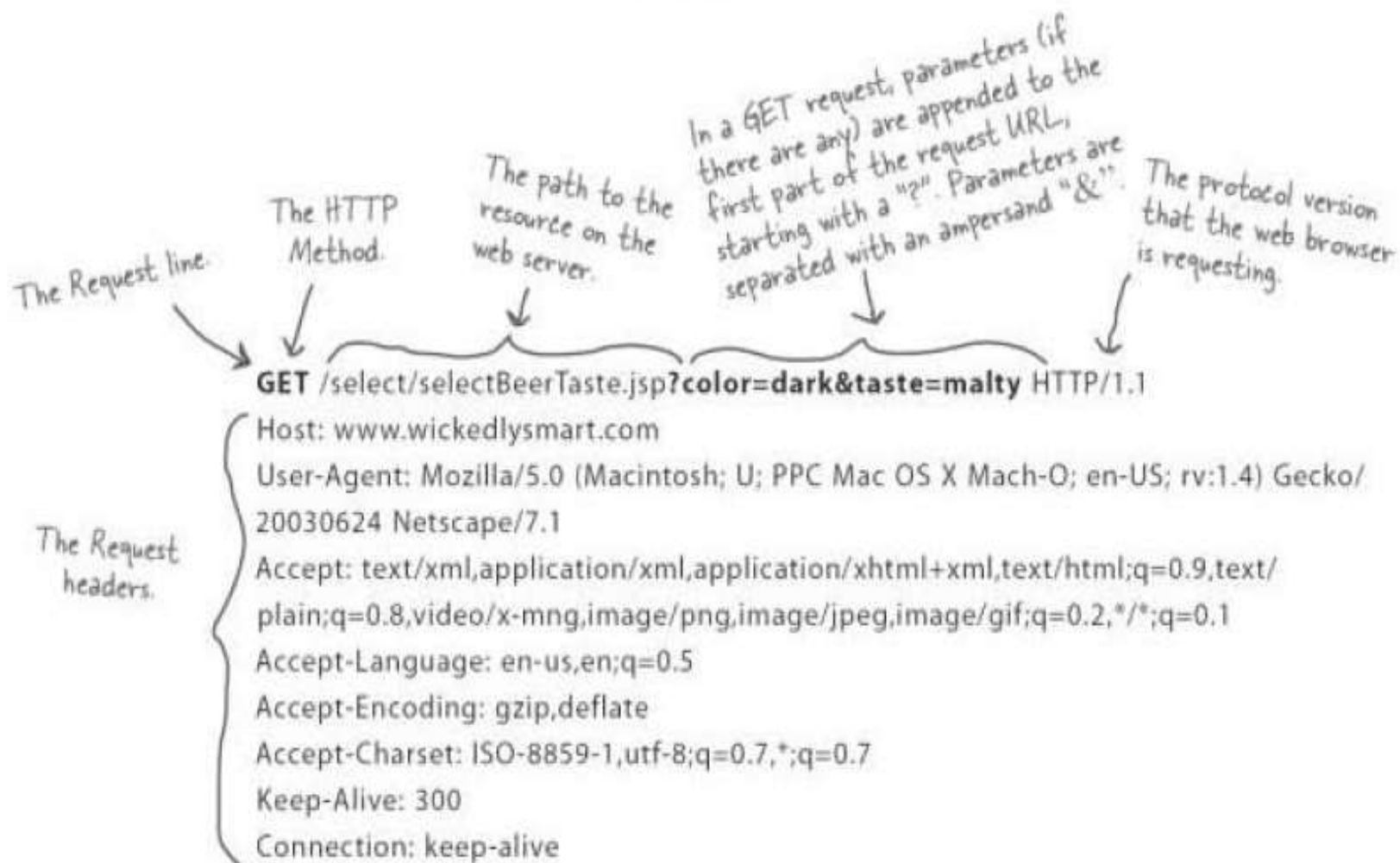
Оригинални URL без додатних параметара

Симбол ? раздваја путању од параметара, а симбол & раздваја параметре.
Параметри се дефинишу у облику име = вредност



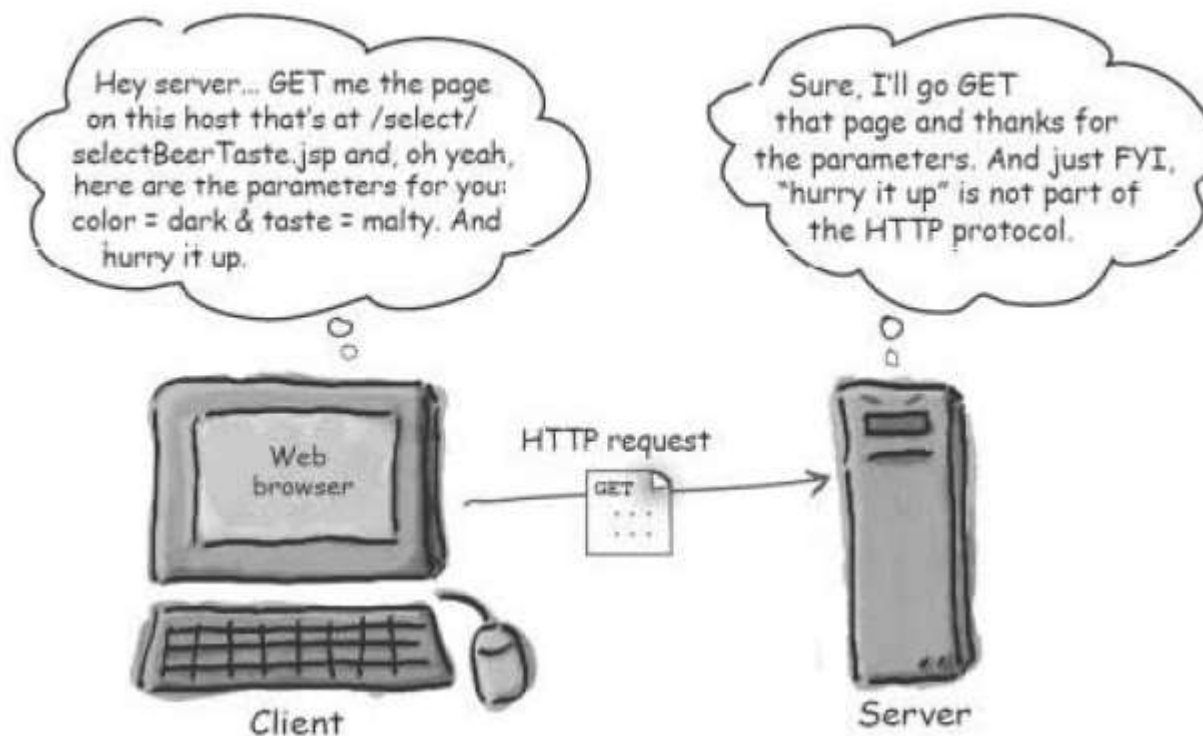


Анатомија GET захтева



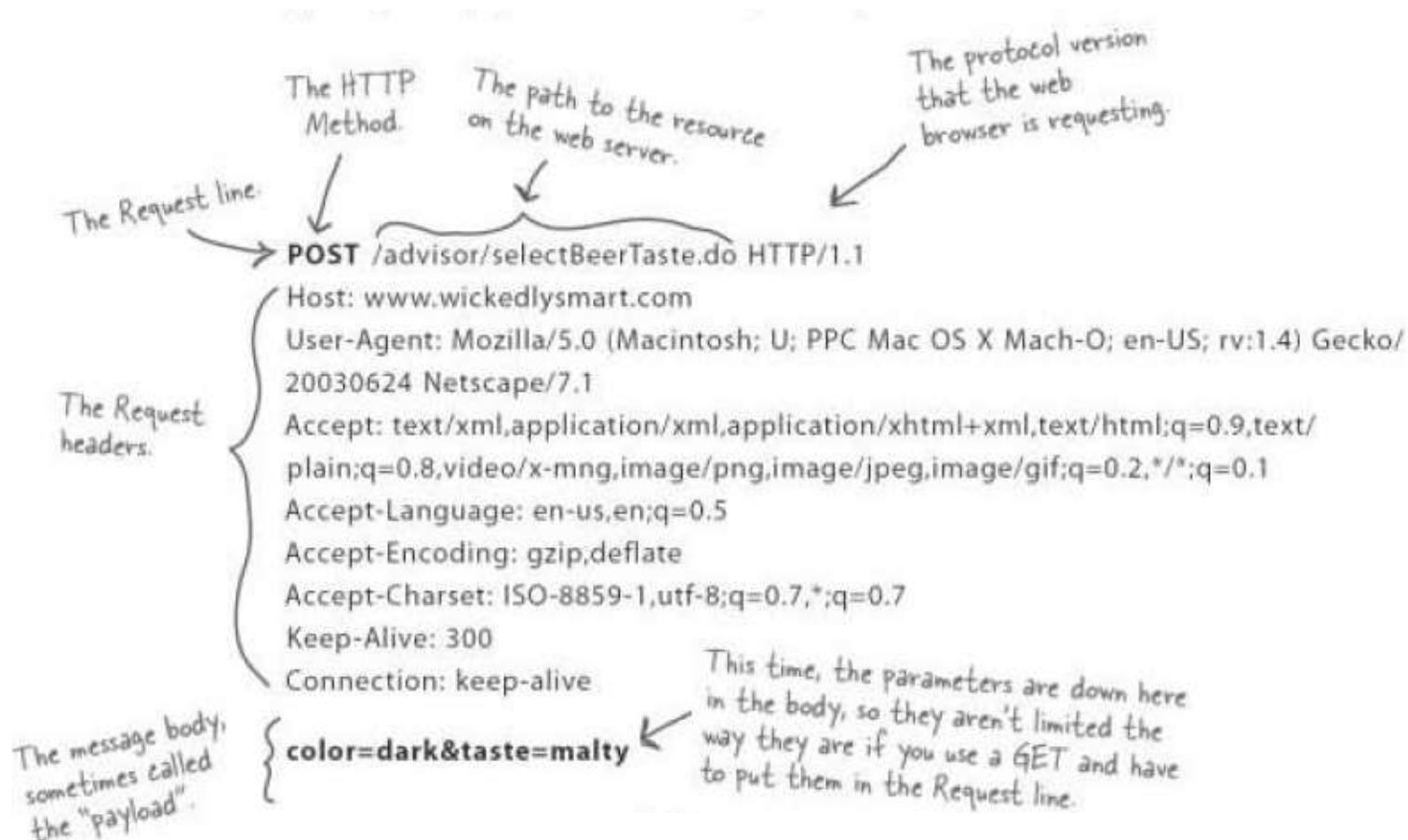


Анатомија GET захтева (2)





Анатомија POST захтева





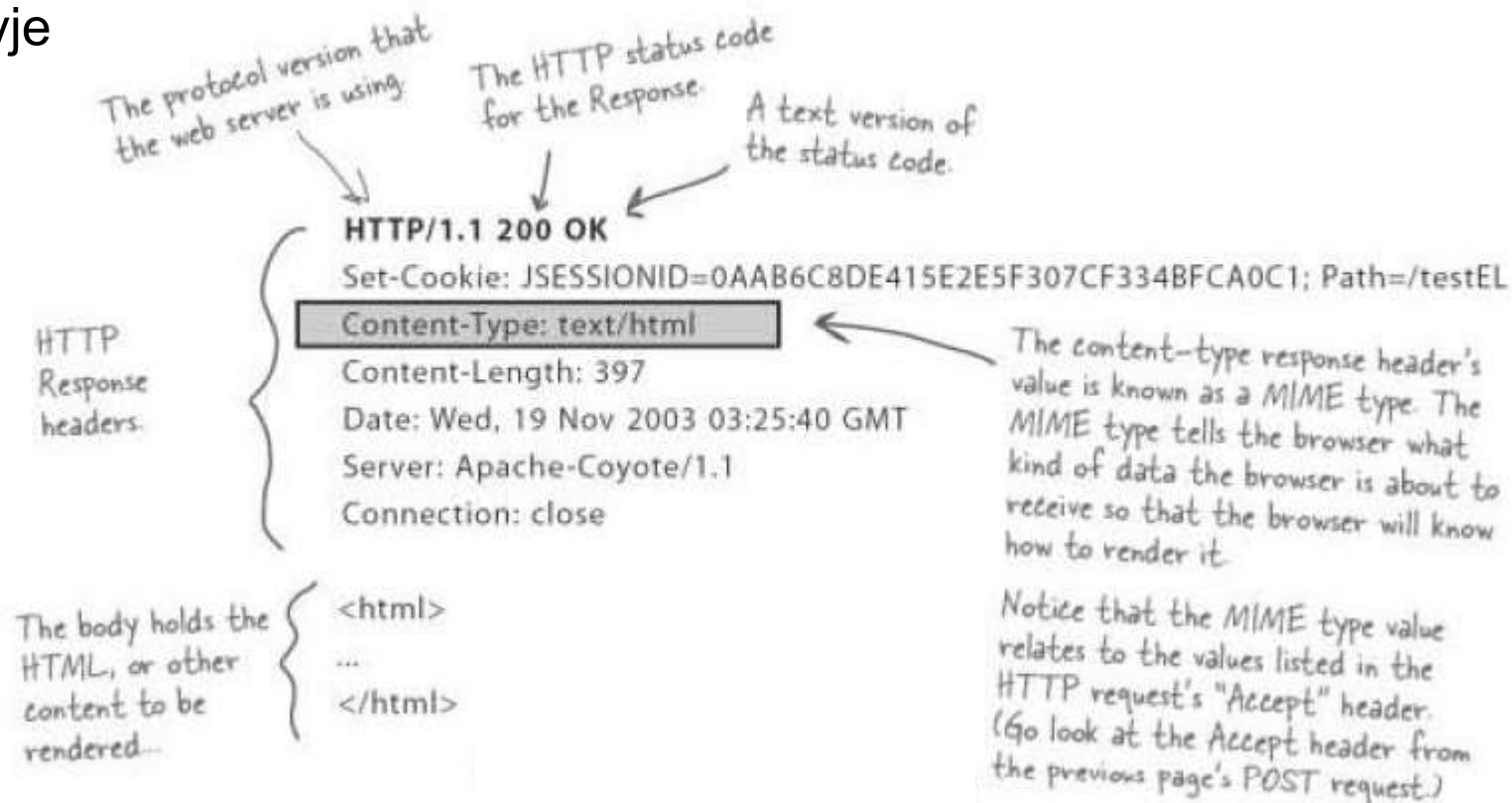
Анатомија POST захтева (2)





Анатомија HTTP одговора

- HTTP одговор садржи заглавље и тело
 - Информације у заглављу говоре прегледачу који је протокол коришћен, да ли је захтев био успешан и која врста садржаја се налази у телу захтева, а тело садржи сам садржај који прегледач приказује



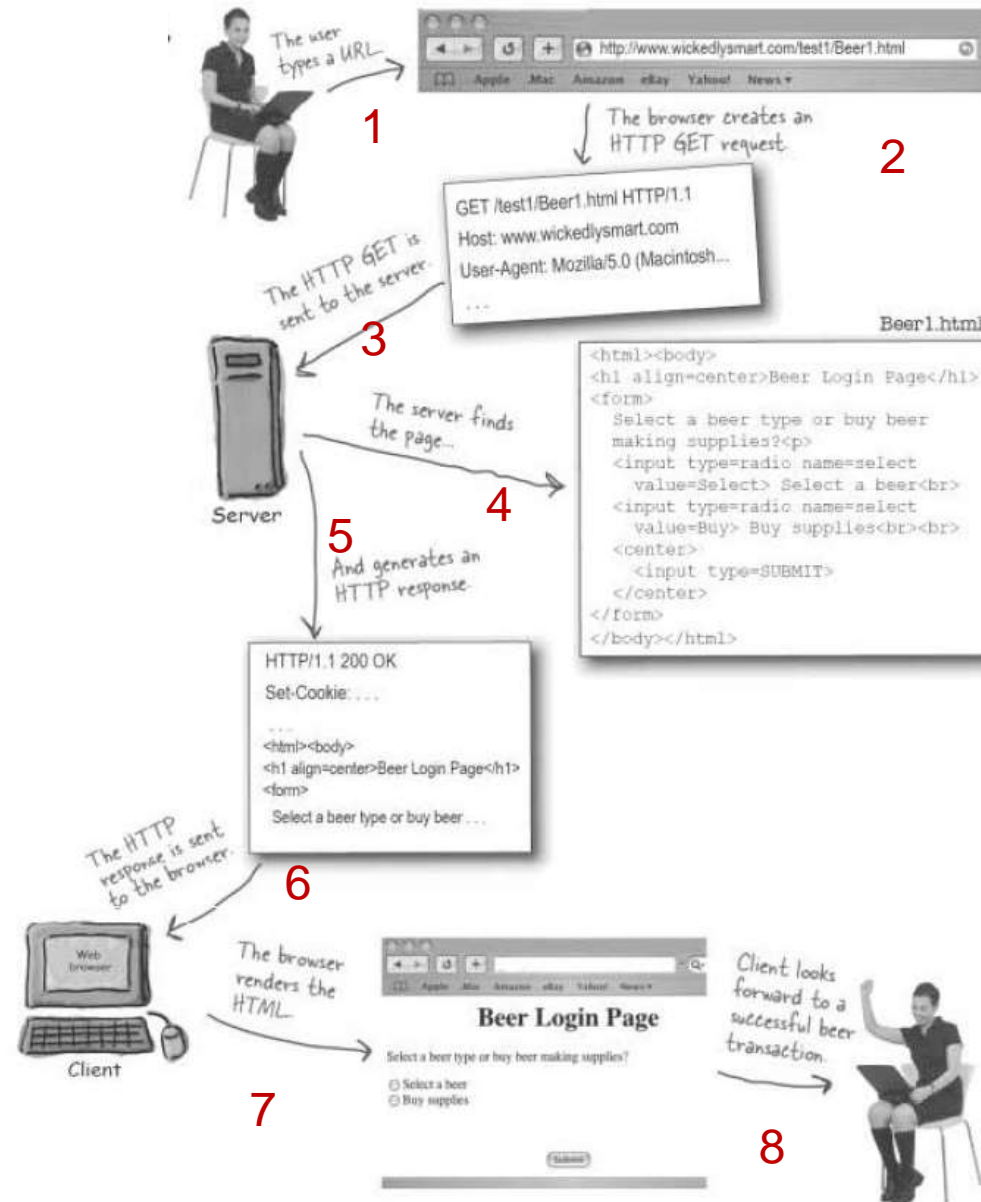


Анатомија HTTP одговора (2)





HTTP захтев и HTTP одговор





Статусни кодови HTTP одговора

- Статусни код HTTP одговора је троцифрен број. Прва цифра статусног кода HTTP одговора специфицира о којој се од пет класа одговора ради
- Статусни кодови су прошириви и HTTP клијенти нису обавезни да разумеју значење свих статусних кодова
- Побројани статусни кодови су део HTTP/1.1 стандарда (документ RFC 7231), осим ако се не истакне да је другачије



Статусни кодови HTTP одговора (2)

- 1xx: Information – Захтев је примљен и процес се наставља

1xx: Information

Message	Description
100 Continue	Only a part of the request has been received by the server, but as long as it has not been rejected, the client should continue with the request.
101 Switching Protocols	The server switches protocol.



Статусни кодови HTTP одговора (3)

- 2xx: Successful – захтев је успешно примљен, схваћен и прихваћен

2xx: Successful

Message	Description
200 OK	The request is OK.
201 Created	The request is complete, and a new resource is created .
202 Accepted	The request is accepted for processing, but the processing is not complete.
203 Non-authoritative Information	The information in the entity header is from a local or third-party copy, not from the original server.
204 No Content	A status code and a header are given in the response, but there is no entity-body in the reply.
205 Reset Content	The browser should clear the form used for this transaction for additional input.
206 Partial Content	The server is returning partial data of the size requested. Used in response to a request specifying a <i>Range</i> header. The server must specify the range included in the response with the <i>Content-Range</i> header.



Статусни кодови HTTP одговора (4)

- 3xx: Redirection – морају се предузети додатне акције да би се компетирао захтев

3xx: Redirection

Message	Description
300 Multiple Choices	A link list. The user can select a link and go to that location. Maximum five addresses .
301 Moved Permanently	The requested page has moved to a new url .
302 Found	The requested page has moved temporarily to a new url .
303 See Other	The requested page can be found under a different url .
304 Not Modified	This is the response code to an <i>If-Modified-Since</i> or <i>If-None-Match</i> header, where the URL has not been modified since the specified date.
305 Use Proxy	The requested URL must be accessed through the proxy mentioned in the <i>Location</i> header.
306 <i>Unused</i>	This code was used in a previous version. It is no longer used, but the code is reserved.
307 Temporary Redirect	The requested page has moved temporarily to a new url.



Статусни кодови HTTP одговора (5)

- 4xx: Client Error – захтев садржи некоректну синтаксу или не може бити испуњен

4xx: Client Error

Message	Description		
400 Bad Request	The server did not understand the request.	410 Gone	The requested page is no longer available .
401 Unauthorized	The requested page needs a username and a password.	411 Length Required	The "Content-Length" is not defined. The server will not accept the request without it .
402 Payment Required	<i>You can not use this code yet.</i>	412 Precondition Failed	The pre condition given in the request evaluated to false by the server.
403 Forbidden	Access is forbidden to the requested page.	413 Request Entity Too Large	The server will not accept the request, because the request entity is too large.
404 Not Found	The server can not find the requested page.	414 Request- url Too Long	The server will not accept the request, because the url is too long. Occurs when you convert a "post" request to a "get" request with a long query information .
405 Method Not Allowed	The method specified in the request is not allowed.	415 Unsupported Media Type	The server will not accept the request, because the mediatype is not supported .
406 Not Acceptable	The server can only generate a response that is not accepted by the client.	416 Requested Range Not Satisfiable	The requested byte range is not available and is out of bounds.
407 Proxy Authentication Required	You must authenticate with a proxy server before this request can be served.	417 Expectation Failed	The expectation given in an Expect request-header field could not be met by this server.
408 Request Timeout	The request took longer than the server was prepared to wait.		
409 Conflict	The request could not be completed because of a conflict.		



Статусни кодови HTTP одговора (7)

- 5xx: Server Error – сервер није успео да испуни наизглед ваљан захтев

5xx: Server Error

Message	Description
500 Internal Server Error	The request was not completed. The server met an unexpected condition.
501 Not Implemented	The request was not completed. The server did not support the functionality required.
502 Bad Gateway	The request was not completed. The server received an invalid response from the upstream server.
503 Service Unavailable	The request was not completed. The server is temporarily overloading or down.
504 Gateway Timeout	The gateway has timed out.
505 HTTP Version Not Supported	The server does not support the "http protocol" version.



Protokol aplikativnog sloja - HTTP

- Hypertext Transfer Protocol (HTTP), je protokol aplikacionog sloja koji predstavlja osnovu veba
- HTTP je implementiran u okviru dve vrste programa:
 - klijentskim programima, najčešće pregledačima veba
 - serverskim programima, najčešće veb serverima
- Ovi programi međusobno komuniciraju razmenom HTTP poruka
- HTTP definiše strukturu ovih poruka i način na koji klijenti i serveri razmenjuju ove poruke
- Neki od osnovnih pojmova veba:
 - Veb je distribuirana aplikacija zasnovana na veb stranicama
 - Veb strane se sastoje od objekata – hipertekstualnih datoteka opisanih na jeziku HTML, slika u raznim formatima (npr. JPG, PNG, GIF), Java apleta i sli.
 - Svaki pojedinačni objekat ima jedinstvenu adresu u obliku tzv. URI (Uniform Resource Identifier)



Protokol aplikativnog sloja – HTTP (2)

- Važno je naglasiti da nakon slanja odgovora, server ne održava tj. ne koristi apsolutno nikakve informacije o klijentu, odnosno da je HTTP protokol bez stanja (stateless protocol)
- HTTP zahtevi i odgovori se navode u precizno specificiranom obliku

```
GET /~filip/uvit/ HTTP/1.1
Host: www.matf.bg.ac.rs
User-Agent: Mozilla/5.0 Firefox/3.5.8
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
```

primer HTTP zahteva

```
metod putanja verzija
polje: vrednost
...
polje: vrednost

sadrzaj
```

opšti format HTTP zahteva



Protokol aplikativnog sloja – HTTP (3)

- HTTP zahtev se šalje nakon što je uspostavljena TCP konekcija sa nekih host računarom
- U prvoj liniji, navodi se ime metoda, putanja (na serveru) do objekta koji se zahteva i verzija HTTP protokola
- Najčešće korišćeni metodi su GET, POST i donekle HEAD
- HTTP zahtev sadrži i niz polja i njihovih vrednosti kojima klijent serveru saopštava neke relevantne informacije:
 - Host: - obavezno polje u HTTP/1.1 i sadrži ime hosta na koji se šalje zahtev
 - User-Agent: - ovim se identifikuje klijentski softver koji šalje zahtev
 - Accept: - ovim klijent navodi vrstu sadržaja (MIME tip) koju priželjkuje
 - Accept-Language: - ovim klijent navodi jezik koji priželjkuje
 - Accept-Charset: - ovim klijent navodi kodnu stranu koju priželjkuje
 - Connection: - ovim se navodi da li se želi perzistentna (keep-alive) ili jednokratna (close) TCP konekcija.



Protokol aplikativnog sloja – HTTP (4)

- Niz polja u HTTP zahtevu:
 - If-modified-since: - ovim klijent serveru naglašava da mu objekat pošalje samo ako je bio modifikovan od datuma navedenog u ovom polju (ukoliko objekat nije modifikovan, on se ne šalje ponovo već klijent prikazuje verziju koja mu je prethodno bila dostavljena i koja je sačuvana je u kešu)
- Nakon prijema HTTP zahteva, server šalje HTTP odgovor

```
HTTP/1.1 200 OK
Date: Sun, 07 Mar 2010 14:53:05 GMT
Server: Apache/2.2.9 (Unix) mod_ssl/2.2.9 OpenSSL/0.9.8h PHP/5.2.6
X-Powered-By: PHP/5.2.6
Content-Length: 2905
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
...
```

primer HTTP odgovora



Protokol aplikativnog sloja – HTTP (5)

```
verzija kod status  
polje: vrednost  
...  
polje: vrednost  
  
sadrzaj
```

opšti format HTTP odgovora

- Kod i status su u odgovoru najčešće nešto od sledećeg:
 - 200 OK - Zahtev je uspešan i informacija se vraća u okviru odgovora
 - 301 Moved Permanently - Zahtevani objekat je premešten na lokaciju koja je navedena u polju Location: i klijentski program može automatski da pošalje novi zahtev na dobijenu lokaciju
 - 304 Not Modified - Zahtevani objekat nije promenjen od datuma navedenog u zahtevu i nema ga potrebe ponovo slati
 - 400 Bad Request - Server nije uspeo da razume zahtev
 - 404 Not Found - Zahtevani objekat nije naden na serveru



Protokol aplikativnog sloja – HTTP (6)

- Kod i status su u odgovoru najčešće nešto od sledećeg:
 - 500 Internal Server Error - Došlo je do neke interne greške u radu serverskog programa
 - 505 HTTP Version Not Supported - Server ne podržava verziju HTTP protokola
- Kodovi koji počinju sa 2 govore o tome da je sve proteklo kako treba, kodovi koji počinju sa 3 obaveštavaju korisnika o nekoj redirekciji, kodovi koji počinju sa 4 govore o nekoj grešci u zahtevu (grešci koju je napravio klijent), a kodovi koji počinju sa 5 govore o nekoj grešci na strani servera



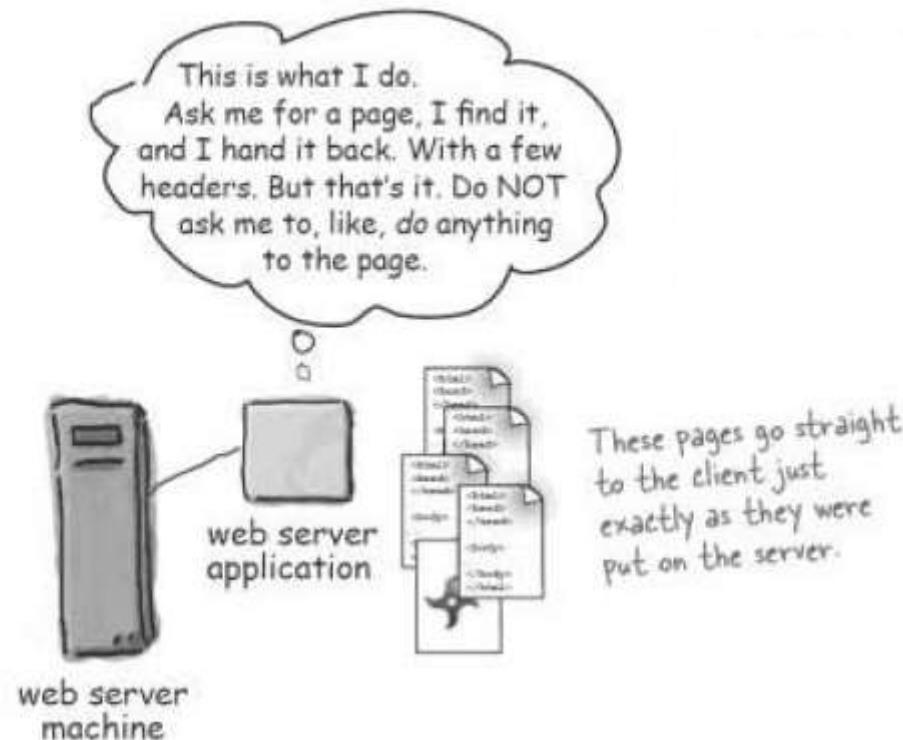
Protokol aplikativnog sloja – HTTP (7)

- Neka od najčešće navedenih polja u HTTP odgovorima su:
 - Date: - tačno vreme kada je odgovor poslat
 - Server: - identifikacija veb server programa koji je poslao odgovor
 - Content-Type: - vrsta sadržaja (MIME tip) poslata u okviru odgovora
 - Content-Length: - dužina sadržaja u bajtovima
 - Last-Modified: - vreme kada je sadržaj poslednji put modifikovan na serveru



Статичке веб стране

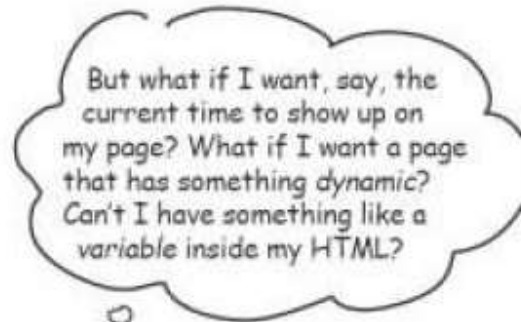
- Статичка веб страна се налази у директоријуму на веб серверу
 - Веб сервер такву страну само пронађе и проследи клијенту, баш онакву каква је на серверу
 - Сваки од клијената добија потпуно исти садржај као одговор





Статичке веб стране (2)

- Шта радити када треба обезбедити да се веб страна динамички мења?



```
<html>
<body>
The current time is [insertTimeOnServer].
</body>
</html>
```

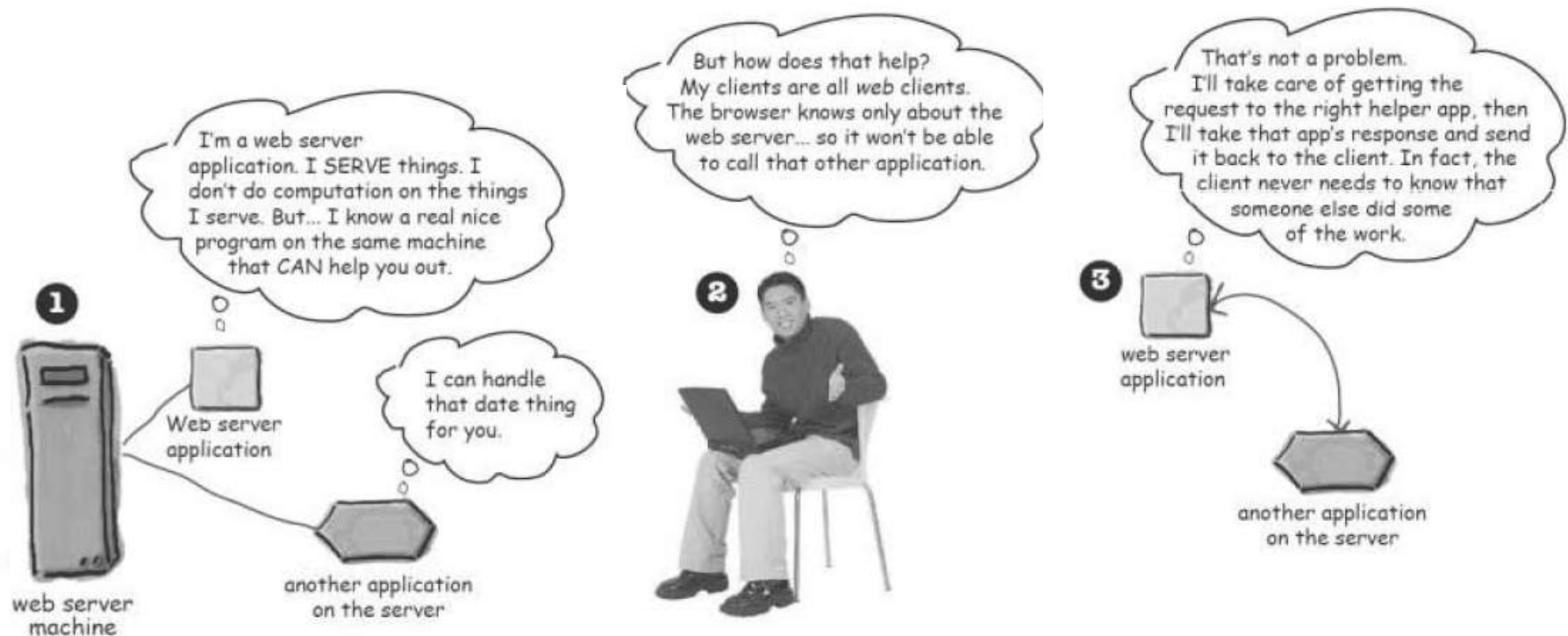
What if we want to stick something variable inside the HTML page?





Динамичке веб стране

- Сам веб сервер опслужује само статичке стране, али се може користити посебна помоћна апликација, са којом комуницира веб сервер, а која креира динамички садржај





Динамичке веб стране (2)

- Динамички садржај може бити било шта: датум и време са сервера, списак датотека у директоријуму, случајно изабрана слика итд.
- Динамички садржај не постоји све док не стигне захтев
- По приспећу захтева, помоћна апликација „креира“ HTML а онда веб сервер тај HTML „спакује“ у одговор

Уместо статичног

```
<html>
<body>
The current time is
always 4:20 PM
on the server
</body>
</html>
```

Треба да се добије динамичан садржај

```
<html>
<body>
The current time is
[insertTimeOnServer]
on the server
</body>
</html>
```



Динамичке веб стране (3)

- Када корисник проследи серверу податке са форме, тада је за процесирање прослеђених података (чување података у бази, ради генерисање одговора на основу података прослеђених уз захтев итд.) неопходно коришћење помоћне апликације
- Када сервер препозна да се захтев односи на помоћну апликацију, тада и прослеђене параметре проследи помоћној апликацији, па та апликација генерише одговор за који се потом проследи клијенту



Slojevi kod računarskih mreža transportni sloj



Protokoli i slojevi

OSI sloj	TCP/IP sloj	Jedinica	Protokol
aplikativni sloj (mrežni procesi vezani za aplikaciju)	aplikativni sloj	podatak	HTTP, FTP, Telnet, DNS, DHCP, POP/SMTP, NNTP
sloj prezentacije (enkripcija i kodiranje podataka)		podatak	MIME, TLS, SSL
sloj sesije (uspostavljanje sesije krajnjih korisnika)		podatak	SSH, Named Pipes, PPTP
transportni sloj (veza, pouzdanost, transport)	transportni sloj	segment datatgram	TCP, UDP , SCTP, DCCP
mrežni sloj (logičko adresiranje i rutiranje)	međumrežni sloj	paket	IP (IPv4, IPv6), ICMP, ARP, RARP
sloj veze podataka (fizičko adresiranje, pristup medijumu)	sloj pristupa mreži	okvir	PPP, HDLC, Frame Relay
fizički sloj (prenos signala)		bit	Token Ring, RS-232, T1, E1, POTS, OTN, DSL, 802.11a/b/g/n PHY, 802.15.x PHY, Ethernet, USB, Bluetooth, Firewire (IEEE 1394)



Transportni sloj

Transportni sloj (transport layer) - ima zadatak da prihvata podatke sa viših slojeva, deli ih na manje jedinice (pakete), šalje te pakete na odredište korišćenjem nižih slojeva

- Obično se na ovom sloju razlikuju dve vrste protokola: protokoli sa uspostavljanjem konekcije i protokoli bez uspostavljanja konekcije
 - Protokoli koji zahtevaju uspostavljanje konekcije garantuju da će poslati podaci zaista i stići na odredište u istom redosledu u kojem su i poslati
 - Protokoli bez uspostavljanja konekcije ne daju ovakve garancije, ali je prenos podataka obično brži
- Za razliku od protokola mrežnog sloja koji moraju da budu implementirani u svakom čvoru lanca komunikacije, protokoli transportnog sloja moraju biti implementirani jedino na krajnjim čvorovima komunikacije (host čvorovima)



Transportni sloj (2)

- Ruteri (uredaji koji posredno učestvuju u komunikaciji prenošenjem paketa) obično nisu svesni detalja transportnih protokola
- Transportni protokoli se, dakle, mogu smatrati protokolima kojim komuniciraju dva host računara
- S obzirom da na istom host računaru obično postoji više različitih programa koji imaju potrebu za komunikacijom (svaki korišćenjem zasebnog aplikacionog protokola, ali zajedničkim korišćenjem transportnog protokola), zadatak transportnih protokola je i da vrše tzv. **multipleksovanje**
 - Multipleksovanje se obično ostvaruje kroz koncept **portova** koji predstavljaju brojeve na osnovu kojih se određuje kom programu pokrenutom na host računaru pripada paket primljen na transportnom sloju
- Najkorišćeniji protokoli ovog sloja (koji se koriste u okviru Interneta) su **Transfer Control Protocol (TCP)** i **User Datagram Protocol (UDP)**



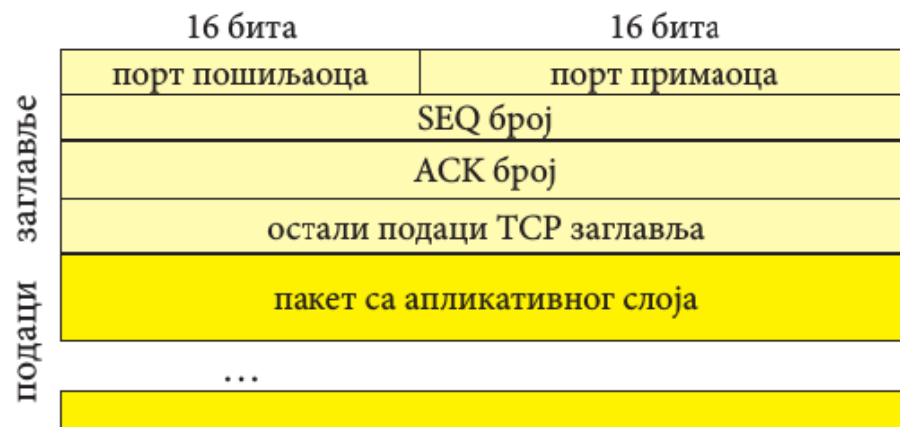
Protokoli transportnog sloja

- Poruka se deli na **pakete** koji se nezavisno šalju (komutiranje paketa)
- Više delova iste poruke može paralelno da putuje kroz mrežu
- Svaki paket se dopunjuje informacijama potrebnim za njegovu dostavu
- Na transportnom sloju paketi se nazivaju **segmenti**
- Komunikacija se organizuje ne samo kao komunikacija između dva uređaja, već između dva programa koji se na njima izvršavaju
- Paket mora da sadrži informaciju o uređaju i softveru koji paket prima i koji paket šalje
- Na transportnom nivou se paketima dodaju identifikatori softvera - portovi, a adrese uređaja tek na mrežnom sloju



TCP protokol transportnog sloja

- TCP (Transmission Control Protocol) je protokol transportnog sloja u okviru Interneta koji pre komunikacije vrši uspostavljanje pouzdane konekcije između dva hosta
 - Kanal komunikacije je dvosmeran (eng. full duplex)
 - Konekcija se uspostavlja tako što klijent i server razmene tri poruke (three way handshake):
 - Klijent traži uspostavljanje konekcije, server potvrđuje da prihvata konekciju i konačno klijent potvrđuje da je konekcija uspostavljena
 - Prava komunikacija može da započne tek nakon što je konekcija uspostavljena, što može da traje neko vreme





TCP protokol transportnog sloja (2)

- TCP garantuje pouzdanost prenosa podataka (reliable transfer) čime se garantuje da će paketi koji su poslani biti primljeni (i to u istom redosledu u kojem su poslani). S obzirom da niži mrežni slojevi ne garantuju dostavu paketa:
 - TCP protokol mora da se stara o tome da paketi koji zalutaju automatski budu ponovno poslani, kao i da na prihvatnoj strani automatski permutuje primljene pakete tako da odgovaraju redosledu slanja
 - Da bi ovo moglo da bude realizovano, uvodi se potvrda prijema paketa (acknowledgment), tj. nakon prijema jednog ili više paketa, vrši se slanje poruke pošaljiocu koja govori da su ti paketi zaista primljeni
 - Pošaljioc, na osnovu ovoga, može da odluči da ponovno pošalje paket koji je ranije već bio poslat, u slučaju da u određenom vremenskom periodu ne dobije potvrdu prijema



TCP protokol transportnog sloja (3)

- TCP uvodi kontrolu i korekciju grešaka (error correction)
 - Ovo je dodatna slaba provera (vrši se samo kontrola parnosti), jer se pretpostavlja da se jača provera (obično CRC) vrši na nižim slojevima
 - Ipak, u praksi se pokazuje da ova provera ima smisla i uspeva da uoči i ispravi veliki broj grešaka koje promaknu ostalim kontrolama
- TCP uvodi i kontrolu brzine protoka (flow control)
 - Njom se kontroliše brzina slanja kako se ne bi desilo da brzi uređaji šalju pakete brzinom većom od one kojom spori uređaji mogu da ih prime (npr. brz računar koji šalje podatke na spor mobilni telefon)



TCP protokol transportnog sloja (4)

- Važna odlika TCP protokola je da vrši kontrolu zagušenja (congestion control)
 - Pojava zagušenja se javlja kada više čvorova pokušava da pošalje podatke kroz mrežu koja je već na granicama svoje propusne moći
 - U takvim situacijama, dešava se da brzina komunikacije u celoj mreži opada za nekoliko redova veličina
 - Naime, broj izgubljenih paketa se višestruko povećava jer unutrašnji čvorovi mreže (ruteri) ne mogu da prihvate nove pakete zato što su im prihvatni baferi prepuni
 - TCP pokušava da detektuje ovakve situacije i da u tim slučajevima uspori sa slanjem paketa dok se mreža ne rastereti
 - Jedna od tehnika koje se koriste u cilju smanjenja zagušenja je da se pri početku komunikacije paketi šalju sporije (slow-start), a da se brzina slanja postepeno povećava kada se utvrdi da paketi zaista i stižu na odredište



TCP protokol transportnog sloja (5)

- Činjenica da TCP protokol da vrši kontrolu zagušenja je jedan od razloga zbog čega TCP spada u grupu sporijih protokola
- Stoga se TCP ne koristi se za aplikacije kod kojih je brzina prenosa presudna



UDP protokol transportnog sloja

- **UDP** (User Datagram protocol) je protokol transportnog sloja u okviru Interneta koji ne vrši uspostavljanje konekcije pre započinjanja komunikacije
 - Prilikom korišćenja UDP protokola ne vrši se potvrda prijema poslatih paketa, tako da se komunikacija može smatrati nepouzdanom
 - Osnovni razlozi korišćenja UDP protokola su, pre svega, njegova brzina - zbog toga se uglavnom koristi od strane aplikacija koje imaju potrebu za komunikacijom u realnom vremenu (real time), kao što su npr. audio-video prenosi, internet telefonija, igrice i sl.
 - Takođe, UDP se koristi za aplikacione protokole koji daju elementarne mrežne usluge i vrše kontrolu mreže (npr. DHCP, DNS, SNMP)



Sistem imena domena

- IP adrese su pogodne za korišćenje od strane računara, ali nisu pogodne za ljudsku upotrebu
- Stoga je uveden je sistem imena domena (domain name system - DNS) – adrese hostova (servera) zadaju se u tekstualnom obliku
- DNS se smatra „telefonskim imenikom” Interneta, koje imenima domena dodeljuje razne informacije (najčešće IP adrese)
 - Na primer, već pomenuti studentski server Matematičkog fakulteta u Beogradu ima domen `alas.matf.bg.ac.rs`
- Domeni su hijerarhijski organizovani i čitaju se s desna na levo
 - Na primer, domen `rs` označava Republiku Srbiju, `ac.rs` označava akademsku mrežu u Srbiji, `bg.ac.rs` njen čvor u Beogradu, `matf.bg.ac.rs` označava Matematički fakultet, dok `alas.matf.bg.ac.rs` označava konkretan studentski server koji tako imenovan u čast velikog matematičara Mihaila Petrovića Alasa
- Domeni najvišeg nivoa mogu biti bilo nacionalni (kao u navedenom primeru), bilo generički (npr. `.com`, `.org`, `.net`), a novom regulativom je liberalizovano korišćenje domena najvišeg nivoa



Sistem imena domena (2)

- Domeni se koriste u okviru jedinstvenih lokatora resursa na Vebu (URL), u okviru adresa elektronske pošte, itd.
- Prilikom preslikavanja domena u adrese, koriste se usluge distribuirane DNS baze podataka
 - Specijalizovani DNS serveri čuvaju delove ove baze
 - Ovi serveri su hijerarhijski organizovani i njihova hijerarhija uglavnom prati hijerarhiju domena

```

Command Prompt

DNS Suffix Search List. . . . . : matf.bg.ac.rs

Ethernet adapter Ethernet 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . : matf.bg.ac.rs
    Description . . . . . : Intel(R) PRO/1000 PM Network Connection
    Physical Address. . . . . : 00-24-21-5F-EE-06
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix . : 
    Description . . . . . : Intel(R) 82566DM-2 Gigabit Network Connection
    Physical Address. . . . . : 00-24-21-5F-EE-05
    DHCP Enabled. . . . . : No
    Autoconfiguration Enabled . . . . : Yes
    Link-local IPv6 Address . . . . . : fe80::a177:9abb:a200:158e%12<Preferred>
    IPv4 Address. . . . . : 147.91.67.138<Preferred>
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 147.91.67.1
    DHCPv6 IAID . . . . . : 201335841
    DHCPv6 Client DUID. . . . . : 00-01-00-01-1C-29-0A-22-00-24-21-5F-EE-05

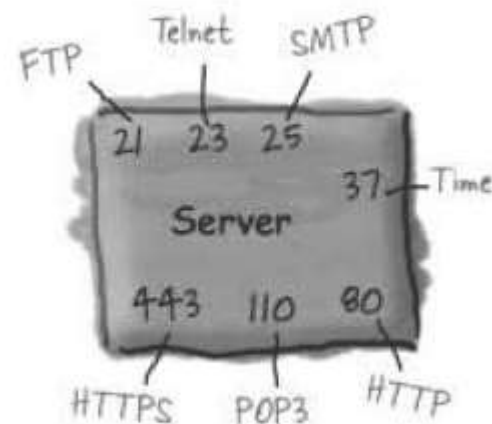
    DNS Servers . . . . . : 147.91.64.4
                           147.91.66.2
    NetBIOS over Tcpip. . . . . : Enabled
  
```



TCP порт

- Различите апликације користе различите портове
 - HTTP сервер подразумевано користи порт 80
 - Telnet сервер подразумевано користи порт 23
 - FTP сервер подразумевано користи порт 21
 - POP3 порт 110, а SMTP порт 25
 - Сервер за време порт 37
- Порт представља једнозначни идентификатор – број између 0 и 65535
- Порт представља логичку везу између конкретног софтвера и хардвера на ком се тај софтвер извршава
- Порт не представља место на које се прикључује уређај, већ број који означава приступну тачку за комуникацију са сервером

Well-known TCP port numbers for common server applications



Using one server app per port, a server can have up to 65536 different server apps running.



TCP порт (2)

- Ако не би било броја порта, онда сервер не би знао са којом клијентском апликацијом комуницира, нити по ком се протоку та комуникација реализује
 - Могло би се догодити, на пример, да прегледач, уместо да комуницира са веб сервером, покушава успостави комуникацију са сервером електронске поште
- TCP портови између 0 и 1023 су резервисани за познате сервисе и не препоручује се да се ти портови користе за „нове“ серверске програме
- Ако се извршавају серверски програми на рачунарској мрежи компаније, тада са администратором система треба проверити који су портови слободни, а који заузети



TCP/IP programski interfejs

- Većina savremenih operativnih sistema i programskih jezika daje direktnu podršku za korišćenje Internet (TCP/IP familije) protokola
- Podrška za korišćenje ovih protokola u okviru programa se realizuje kroz koncept soketa (socket)
- Soket je apstrakcija kojom se programeru predstavlja kanal komunikacije (zasnovan bilo na TCP bilo na UDP protokolu)
- Programer piše podatke u soket ili čita podatke iz soketa, obično na sličan način kao da je u pitanju obična datoteka, a operativni sistem se brine o svim aspektima stvarne mrežne komunikacije



TCP/IP programski interfejs (2)

```
import java.io.*;
import java.net.*;
class TCPClient {
    public static void main(String argv[]) throws Exception {
        String userSentence, modifiedSentence;
        BufferedReader inFromUser =
            new BufferedReader(new InputStreamReader(System.in));
        Socket clientSocket = new Socket("hostname", 6789);
        DataOutputStream outToServer =
            new DataOutputStream(clientSocket.getOutputStream());
        BufferedReader inFromServer =
            new BufferedReader(new InputStreamReader(
                clientSocket.getInputStream()));
        userSentence = inFromUser.readLine();
        outToServer.writeBytes(userSentence + '\n');
        modifiedSentence = inFromServer.readLine();
        System.out.println("FROM SERVER: " + modifiedSentence);
        clientSocket.close();
    }
}
```



TCP/IP programski interfejs (3)

```
import java.io.*;
import java.net.*;
class TCPServer {
    public static void main(String argv[]) throws Exception {
        String clientSentence, modifiedSentence;
        ServerSocket welcomeSocket = new ServerSocket(6789);
        while(true) {
            Socket connectionSocket = welcomeSocket.accept();
            BufferedReader inFromClient =
                new BufferedReader(new InputStreamReader(
                    connectionSocket.getInputStream()));
            DataOutputStream outToClient =
                new DataOutputStream(connectionSocket.getOutputStream());
            clientSentence = inFromClient.readLine();
            modifiedSentence = clientSentence.toUpperCase() + '\n';
            outToClient.writeBytes(modifiedSentence);
        }
    }
}
```



Slojevi kod računarskih mreža aplikativni sloj protokol FTP



Protokoli i slojevi

OSI sloj	TCP/IP sloj	Jedinica	Protokol
aplikativni sloj (mrežni procesi vezani za aplikaciju)	aplikativni sloj	podatak	HTTP, FTP , Telnet, DNS, DHCP, POP/SMTP, NNTP
sloj prezentacije (enkripcija i kodiranje podataka)		podatak	MIME, TLS, SSL
sloj sesije (uspostavljanje sesije krajnjih korisnika)		podatak	SSH, Named Pipes, PPTP
transportni sloj (veza, pouzdanost, transport)	transportni sloj	segment datatgram	TCP, UDP, SCTP, DCCP
mrežni sloj (logičko adresiranje i rutiranje)	međumrežni sloj	paket	IP (IPv4, IPv6), ICMP, ARP, RARP
sloj veze podataka (fizičko adresiranje, pristup medijumu)	sloj pristupa mreži	okvir	PPP, HDLC, Frame Relay
fizički sloj (prenos signala)		bit	Token Ring, RS-232, T1, E1, POTS, OTN, DSL, 802.11a/b/g/n PHY, 802.15.x PHY, Ethernet, USB, Bluetooth, Firewire (IEEE 1394)



Protokol aplikativnog sloja - FTP

- File Transfer Protocol (FTP) je protokol za prenos datoteka između računara
 - Protokol datira još od 1970-tih i doba ranog Interneta, ali se i danas koristi
 - U okviru tipične FTP sesije, korisnik sedi za jednim host računarom i želi da prenosi datoteke na ili sa drugog host računara
 - FTP koristi TCP kao protokol za komunikaciju nižeg nivoa
 - FTP protokol ostvaruje dve TCP konekcije za prenos datoteka. Jedna konekcija (obično na portu 21) se koristi za prenos kontrolnih informacija, a druga (obično na portu 20) za prenos samih podataka
 - Za svaku datoteku, otvara se nova konekcija za prenos podataka, koja se automatski zatvara kada se završi prenos datoteka
 - Za to vreme kontrolna konekcija sve vreme ostaje otvorena
 - Za razliku od HTTP protokola, tokom FTP sesije server mora da čuva određene podatke o korisniku (na primer, tekući direktorijum) tj. FTP je protokol koji čuva stanje (statefull protocol)



Protokol aplikativnog sloja – FTP (2)

- Kontrole koje se izdaju serveru putem kontrolne konekcije se zapisuju u čitljivom ASCII obliku:
 - USER username - koristi se za slanje identifikacije korisnika serveru
 - PASS password - koristi se za slanje lozinke korisnika serveru
 - LIST - koristi se kako bi se serveru poslala poruka da pošalje listu datoteka u tekućem direktorijumu
 - RETR filename - koristi se kako bi se sa servera (iz tekućeg direktorijuma) prenela datoteka sa datim imenom na klijent (u tekući direktorijum)
 - STOR filename - koristi se kako bi se sa klijenta (iz tekućeg direktorijuma) prenela datoteka sa datim imenom na server (u tekući direktorijum)



Protokol aplikativnog sloja – FTP (3)

- Server obično na zahteve klijenta otvara konekciju za prenos podataka, a istovremeno preko kontrolne konekcije šalje statusne poruke ili poruke o greškama, kao što su:
 - 331 Username OK, password required
 - 25 Data connection already open; transfer starting
 - 425 Can't open data connection
 - 452 Error writing file



Slojevi kod računarskih mreža aplikativni sloj protokoli POP3, SMTP i IMAP



Protokoli i slojevi

OSI sloj	TCP/IP sloj	Jedinica	Protokol
aplikativni sloj (mrežni procesi vezani za aplikaciju)	aplikativni sloj	podatak	HTTP, FTP, Telnet, DNS, DHCP, POP/SMTP , NNTP
sloj prezentacije (enkripcija i kodiranje podataka)		podatak	MIME, TLS, SSL
sloj sesije (uspostavljanje sesije krajnjih korisnika)		podatak	SSH, Named Pipes, PPTP
transportni sloj (veza, pouzdanost, transport)	transportni sloj	segment datagram	TCP, UDP, SCTP, DCCP
mrežni sloj (logičko adresiranje i rutiranje)	međumrežni sloj	paket	IP (IPv4, IPv6), ICMP, ARP, RARP
sloj veze podataka (fizičko adresiranje, pristup medijumu)	sloj pristupa mreži	okvir	PPP, HDLC, Frame Relay
fizički sloj (prenos signala)		bit	Token Ring, RS-232, T1, E1, POTS, OTN, DSL, 802.11a/b/g/n PHY, 802.15.x PHY, Ethernet, USB, Bluetooth, Firewire (IEEE 1394)



Protokoli aplikativnog sloja – SMTP, POP3 i IMAP

- Pre nego opisa pojedinačnih protokola, ukratko o osnovnim principima funkcionisanja elektronske pošte:
 - Za slanje elektronske poruka sa računara pošaljioca na računar primaoca, potrebno je da u komunikaciju budu uključeni i server elektronske pošte pošiljaoca, kao i server elektronske pošte primaoca
 - 1. Pošiljaoc sa svog računara dostavlja poruku svom serveru, od kog se zahteva da poruku dostavi serveru primaoca i smesti je u poštansko sanduče primaoca
 - 2. Server pošiljoca nastavlja da brine o dostavljanju poruke tj. vrši komunikaciju sa serverom primaoca i pokušava da dostavi poruku sve dok ili ne uspe ili dok ne ustanovi da dostavljanje poruke nije moguće
 - 3. U slučaju da dostavljanje poruke nije uspelo, server obično obaveštava pošiljaoca da dostavljanje nije uspelo
 - 4. Kada se poruka uspešno dostavi na server primaoca, ona se smešta u njegovo poštansko sanduče gde je smeštena sve dok primaoc ne proverí svoju poštu i ne poželi da pročita dobijenu poruku
 - 5. U tom trenutku potrebno je dostaviti poruku sa servera primaoca do njegovog ličnog računara



Protokoli aplikativnog sloja – SMTP, POP3 i IMAP (2)

- Simple Mail Transfer Protocol (SMTP) je standardni protokol za slanje pošte

```

Server: 220 smtp.example.com ESMTP Postfix
Client: HELO relay.example.org
Server: 250 Hello relay.example.org, I am glad to meet you
Client: MAIL FROM:<bob@example.org>
Server: 250 Ok
Client: RCPT TO:<alice@example.com>
Server: 250 Ok
Client: RCPT TO:<theboss@example.com>
Server: 250 Ok
Client: DATA
Server: 354 End data with <CR><LF>.<CR><LF>
Client: From: "Bob Example" <bob@example.org>
      To: Alice Example <alice@example.com>
      Cc: theboss@example.com
      Date: Tue, 15 Jan 2008 16:02:43 -0500
      Subject: Test message

      Hello Alice.
      This is a test message with 5 header fields and
      5 lines in the message body.
      Your friend,
      Bob
      .
Server: 250 Ok: queued as 12345
Client: QUIT
Server: 221 Bye
  
```

Primer SMTP sesije između klijenta koji šalje poštu i servera koji je prima, kako bi je dalje prosledio



Protokoli aplikativnog sloja – SMTP, POP3 i IMAP (3)

- Post Office Protocol (POP) je jednostavni protokol za preuzimanje poruka sa servera, pri čemu se prilikom preuzimanja poruke obično brišu sa servera
- Preuzete poruke se čuvaju na klijentskom računaru, koji nakon preuzimanja poruka više ne mora da ima pristup Internetu
- POP protokol koristi TCP konekciju na portu 110.

Primer POP3 sesije
između
klijenta i servera

```

Server: +OK POP3 server ready <1896.697170952@dbc.mtview.ca.us>
Client: APOP mrose c4c9334bac560ecc979e58001b3e22fb
Server: +OK mrose's maildrop has 2 messages (320 octets)
Client: STAT
Server: +OK 2 320
Client: LIST
Server: +OK 2 messages (320 octets)
      1 120
      2 200
      .
Client: RETR 1
Server: +OK 120 octets
      <the POP3 server sends message 1>
      .
Client: DELE 1
Server: +OK message 1 deleted
Client: RETR 2
Server: +OK 200 octets
      <the POP3 server sends message 2>
Client: QUIT
Server: +OK dewey POP3 server signing off (maildrop empty)
  
```



Protokoli aplikativnog sloja – SMTP, POP3 i IMAP (4)

- Osnovne komande koje klijentski softver izdaje u POP3 protokolu su:
 - APOP - ovim se vrši autorizacija klijenta navođenjem njegovog korisničkog imena i kriptovane lozinke.
 - STAT - statistika o stanju poštanskog sandučeta
 - LIST - lista poruka
 - RETR - primanje poruke sa navedenim rednim brojem
 - DELE - brisanje poruke sa navedenim rednim brojem
 - QUIT - prekidanje sesije



Protokoli aplikativnog sloja – SMTP, POP3 i IMAP (5)

- Internet Message Access Protocol (IMAP) je znatno napredniji protokol za primanje pošte. On je prevashodno namenjen korisnicima koji su mobilni tj. koji svojoj pošti pristupaju sa različitih računara
 - Kako bi ovakvi korisnici imali mogućnost pristupa svim svojim porukama, nije poželjno brisati ih sa servera (iz poštanskog sandučeta) prilikom preuzimanja
 - Klijenti za elektronsku poštu na lokalnim računarima obično omogućavaju korisnicima sortiranje poruka, organizovanje u fascikle, pretragu i sli.
 - IMAP protokol je projektovan tako da se ovakva funkcionalnost obezbedi tako što se korisnicima omogući da ove funkcije izvode direktno u svom poštanskom sandučetu na serveru
 - Mana ovog pristupa je što se zahteva da korisnici imaju pristup Internetu sve vreme dok rade sa svojom elektronskom poštom
 - Određeni broj veb aplikacija za rad sa elektronskom poštom je zasnovan na IMAP protokolu



Zahvalnica

Delovi materijala ove prezentacije su preuzeti iz:

- Skripte iz predmeta Uvod u veb i internet tehnologije, na Matematičkom fakultetu Univeziteta u Beogradu, autor prof. dr Filip Marić
- Prezentacija iz predmeta Uvod u veb i internet tehnologije, na Matematičkom fakultetu Univeziteta u Beogradu, autor dr Vesna Marinković
- Skripte iz predmeta Informatika na Univerzitetu Milano Bicocca, autor dr Dario Pescini