

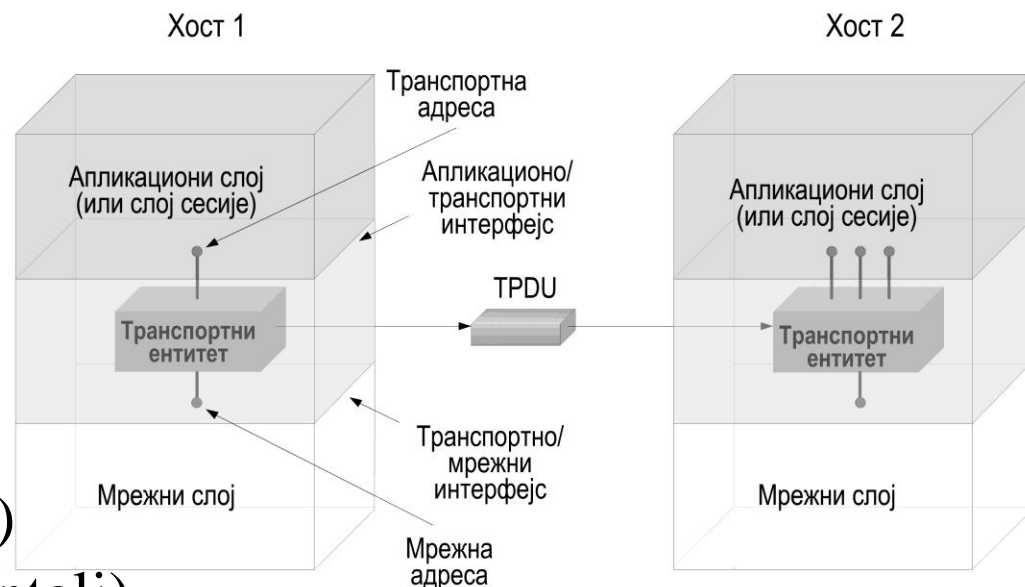
XI - Funkcije transportnog sloja

- Predstavlja **ključni sloj** u konceptu arhitekture mrežnih protokola.
- Protokoli nižih slojeva su **lakši za razumevanje** i manje složeni
- On se nalazi između sloja **aplikacije** (sloj sesije/OSI) i **mrežnog sloja**
- Obezbeđuje **efikasan, pouzdan i ekonomičan** prenos podataka između krajnjih korisnika-*sa kraja na kraj* koji se ostvaruje na mrežnom nivou
- Da prihvati podatke **aplikacije izvorišta** i dostavi ih **aplikaciji odredišta** starajući se o **prenosu, kontroli i ispravljanju grešaka** pri prenosu i **garantovanjem isporuke**.

- Koristi usluge koje mu **obezbeđuje mrežni sloj** kako bi izvršio svoje funkcije.

- Postoji **logička povezanost** mrežnog, transportnog i aplikacionog sloja:

1. **aplikacija-mreža** (po vertikali)
2. **dva udaljena hosta** (po horizontali)



XI - Funkcije transportnog sloja

- Podrška za transportni sloj je **realizovana na nivou OS računara** s tim da sam transportni sloj nije eksplicitno definisan već se realizuje kroz **podršku za protokole tog sloja**.
- Svaka aplikacija može se projektovati tako **da koristi direktno uslugu transportnog sloja** bez posredstva sloja sesije i prezentacije.
- Pored prihvatanja podataka od aplikativnog sloja transportni sloj ima zadatak i **da te podatke prevede u oblik pogodan za transport**.
- Znatno kompleksnija funkcija ovog sloja jeste da podatke prenese korišćenjem nižih slojeva **kojima često nedostaju pomenute funkcionalnosti** vezane za garantovanje isporuke i kontrolu grešaka.
- Usluga koju pruža transportni sloj veoma je slična usluzi koju pruža mrežni sloj pa se **postavlja pitanje zašto postoje dva sloja** ?
- Softver koji realizuje usluge transportnog sloja **nalazi se na računaru samog korisnika**, dok se softver koji realizuje usluge mrežnog sloja najčešće **nalazi na ruterima** koji su kod računarskih mreža širih geografskih područja deo opreme operatera.

XI - Funkcije transportnog sloja

- ✓ Ostvarivanje **virtuelne veze** za prenos podataka
- ✓ Održavanje **višestrukih komunikacija** između aplikacija na obe strane.
- ✓ **Prevođenje podataka** u (uglavnom binarni) format pogodan za prenos.
- ✓ **Podela podataka** radi efikasnijeg iskorišćenja komunikacionog kanala.
Segmentacija kod pošiljaoca a **reasembliranje** na prijemnoj strani
- ✓ Isporuka podataka **u identičnom obliku** u kom su poslani.
- ✓ **Kontrola toka** (*Flow control*)-omogućavanje **optimalne brzine prenosa** u zavisnosti od mogućnosti i trenutnog opterećenja mreže (**propusna moć kanala, učestalošću grešaka** na kanalu i **prihvatnoj moći** primaoca).
- ✓ **Connection oriented**-uspostavljanje i održavanje komunikacione sesije
- ✓ Pouzdan prenos (**Reliable delivery**)- izgubljeni ili oštećeni segmenti se detektuju i ponovo šalju pa se **garantuje pouzdan prenos** svih podataka
- ✓ **Održavanje redosleda segmenata** - različitim putevima segmenti mogu stići u izmenjenom redosledu, ali prijemna strana ima zadatak da rekonstruiše originalni redosled **bez obzira na redosled prijema**
- ✓ **Identifikovanje aplikacija** koje šalju podatke - na strani pošiljaoca
- ✓ **Prosleđivanje podataka** označenim aplikacijama - na prijemnoj strani

XI - Funkcije transportnog sloja

- Segmentacija i reasembliranje - količina podataka koja se šalje gotovo uvek je veća od veličine osnovne jedinice za slanje **PDU** (*Protocol Data Unit*). Transportni sloj deli informacije u blokove podataka odgovarajuće veličine. Na odredištu, transportni sloj reasemblira podatke pre nego ih prosledi na odgovarajuću odredišnu aplikaciju ili uslugu
- Multipleksiranje komunikacija - na istom čvoru može biti pokrenuto više aplikacija. Svakoj od ovih aplikacija dodeljena je adresa-**broj porta**, na osnovu koga transportni sloj određuje kojoj aplikaciji pripada podatak
- Mehanizam pouzdanog prenosa i redosleda segmenata - uvođenje posebnog polja *sequence number* koje je jedinstveno definisalo svaki segment i omogućilo: praćenje poslatih paketa, potvrda primljenih paketa (*acknowledgement*), retransmisija nepotvrđenih segmenata.
- Kao posledica javljalo se **dodatno opterećenje** (*overhead*) transportnog sloja koje utiče na smanjivanje performansi prenosa

XI - Funkcije transportnog sloja

- IP je **odgovoran za komunikaciju** između računara (*host-host*)
- IP isporučuje poruku od izvornog do odredišnog računara.
- Često nije dovoljno samo isporučiti poruku odredišnom računaru, već je **treba i predati odgovarajućem procesu** na odredišnom računaru
- Drugim rečima, konačno odredište poruke nije računar, **već aplikacioni proces na odredišnom računaru** (*proces-proces komunikacija*).
- Proces-proces komunikacija se uobičajno ostvaruje shodno **klijent-server paradigmi**: proces na lokalnom hostu - *client* traži uslugu procesa na udaljenom računaru - *server*.
- Tipično, oba procesa (klijent i server) **imaju isto ime**.
- OS podržavaju **više-korisnička**(*multiuser*) i **više-programaska okruženja**
- Da bi se uspostavila komunikacija između odgovarajućeg para procesa nepohodno je definisati: **lokalni host - proces** i **udaljeni host - proces**
- Lokalni i udaljeni host su **definisani IP adresama**, a lokalni i udaljeni proces definišu se identifikatorom **adresa servisa** ili **broj porta**(*port*)
- Ovaj poslednji korak u isporuci poruka predstavlja **odgovornost protokola transportnog sloja** koji poseduje adresnu jedinicu **port**.

XI - Funkcije transportnog sloja

➤ Port je određen **16-bitnim numeričkim parametrom**(0-65535) i uloga mu je da adresira izvorni/odredišni entitet aplikativnog sloja-aplikaciju

Portovi se mogu podeliti na:

1. Privilegovane - imaju opseg od **0-1023**, pravo na njihovo otvaranje uglavnom ima samo OS. Na ovim portovima se nalaze najčešće korišćeni servisi (FTP, SSH, Telnet, DNS i sl.).

2. Registrovane - kreću se u opsegu od **1024 - 49151** i na njima se zvanično koriste servisi novijeg datuma.

3. Dinamički ili kratkotrajni - kreću se u opsegu od **49152 - 65535** i njih nije moguće registrovati a uglavnom služe za klijentske komponente klijent/server softvera.

➤ Za identifikaciju klijentskog procesa koristi se broj porta koji se zove ***efmerni*** ("kratko-živući") ili privremeni broj porta koji su veći od 1023

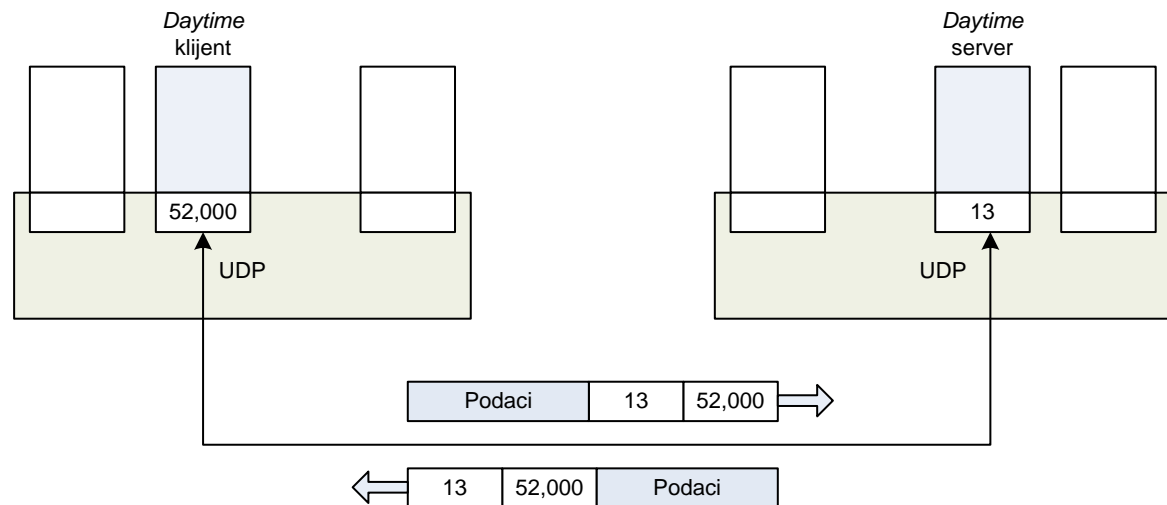
➤ Klijetnskom procesu se prilikom pokretanja dodeljuje **jedan od neiskorišćenih privremenih portova** na klijentskom računaru, kojeg on koristi za **svolju identifikaciju prilikom obraćanja serverskom procesu.**

➤ Kada klijentski proces završi sa radom, njegov **broj porta se oslobađa**

XI - Funkcije transportnog sloja

- Aplikaciju koja se izvršava na serveru **takođe identifikuje jedinstveni broj porta**, ali u ovom slučaju on se **ne može proizvoljno izabrati**.
- Ako bi serverski računar dodeljivao portove svojim serverskim procesima na slučajan način, klijentski procesi koji se izvršavaju na klijentskim računarima **ne bi znali preko kog porta da zatraže uslugu** udaljenog serverskog procesa.
- Kod TCP/IP za standardizovane servise koriste se univerzalni, tzv. **dobro-poznati brojevi porta**.
- **Svi klijentski procesi znaju** dobro-poznati port odgovarajućeg servisa.

Primer: za identifikaciju *Daytime* klijentskog procesa **može** se koristiti broj porta 52 000, ali za *Daytime* serverski proces **mora** da se koristi broj porta 13.

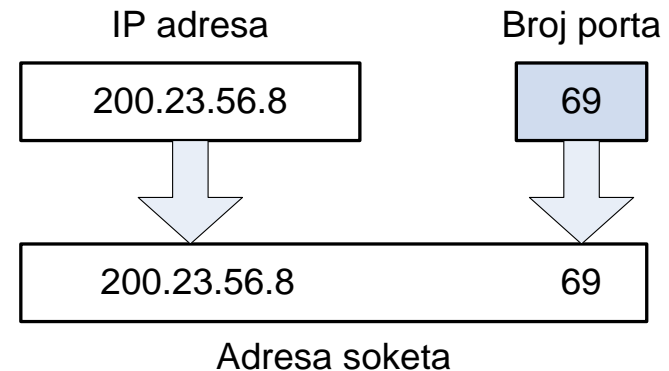


XI - Dobro poznati brojevi portova

Port	Naziv protokola	Skraćenica
20	prenos podataka	FTP
21	kontrolne poruke	FTP
22	<i>Secure Shell</i>	SSH
23	<i>Telnet</i>	
25	<i>Simple Mail Transfer Protocol</i>	SMTP
49	Prijavljivanje na sistem	Login
53	<i>Domain Name System</i>	DNS
69	<i>Trivial File Transport Protocol</i>	TFTP
70	<i>Gopher</i>	Gopher
79	Pregled informacija o korisnicima	Finger
80	<i>Hyper Text Transfer Protocol</i>	HTTP
110	<i>Post Office Protocol verzije 3</i>	POP3
119	<i>USENET</i> vesti	NNTP
123	<i>Network Time Protocol</i>	NTP
143	<i>Internet Message Access Protocol 4</i>	IMAP4
161	<i>Simple Network Management Protocol</i>	SNMP
194	<i>Internet Relay Chat</i>	IRC
389	<i>Lightweight Directory Access Protocol</i>	LDAP
443	HTTP obezbeđen putem TLS/SSL	HTTPS
517	Razgovor sa drugim korisnikom	Talk
860	Internet SCSI	iSCSI
631	<i>Internet Printing Protocol</i>	IPP
989	prenos podataka obezbeđen putem TLS/SSL	FTP
990	kontrolne poruke obezbeđen putem TLS/SSL	FTP

XI - Kompozitna adresna jedinica

- Kompozitna adresna jedinica transportnog i mrežnog sloja jeste **socket**.
- *Socket* (mrežni *socket* ili *socket*) **sastoji se od sledećih komponenti:**
 - ✓ IP adresa izvorišta
 - ✓ Port izvorišta
 - ✓ Protokol transportnog sloja
 - ✓ Port odredišta
 - ✓ IP adresa odredišta



- Podrška za *socket*-e se u OS najčešće realizuje **pomoću gotovih sistemskih biblioteka**.
- Neke od najpopularnijih *socket* biblioteka su **Berkeley socket** za UNIX i **Winsock** za MS Windows operativne sisteme.
- Osim upotrebe u računarskim mrežama *socket*-i se **mogu koristiti i kod aplikacija** koje se izvršavaju na lokalnom računaru.
- **Grafički sistem X Window System** na UNIX-u traži *socket* da bi radio
- Kombinacija IP adrese i broja porta se naziva **adresom soketa**
- Klijentska adresa *socket*-a **jednoznačno definiše klijentski**, kao što serverska adresa *socket*-a **jednoznačno definiše serverski proces**.

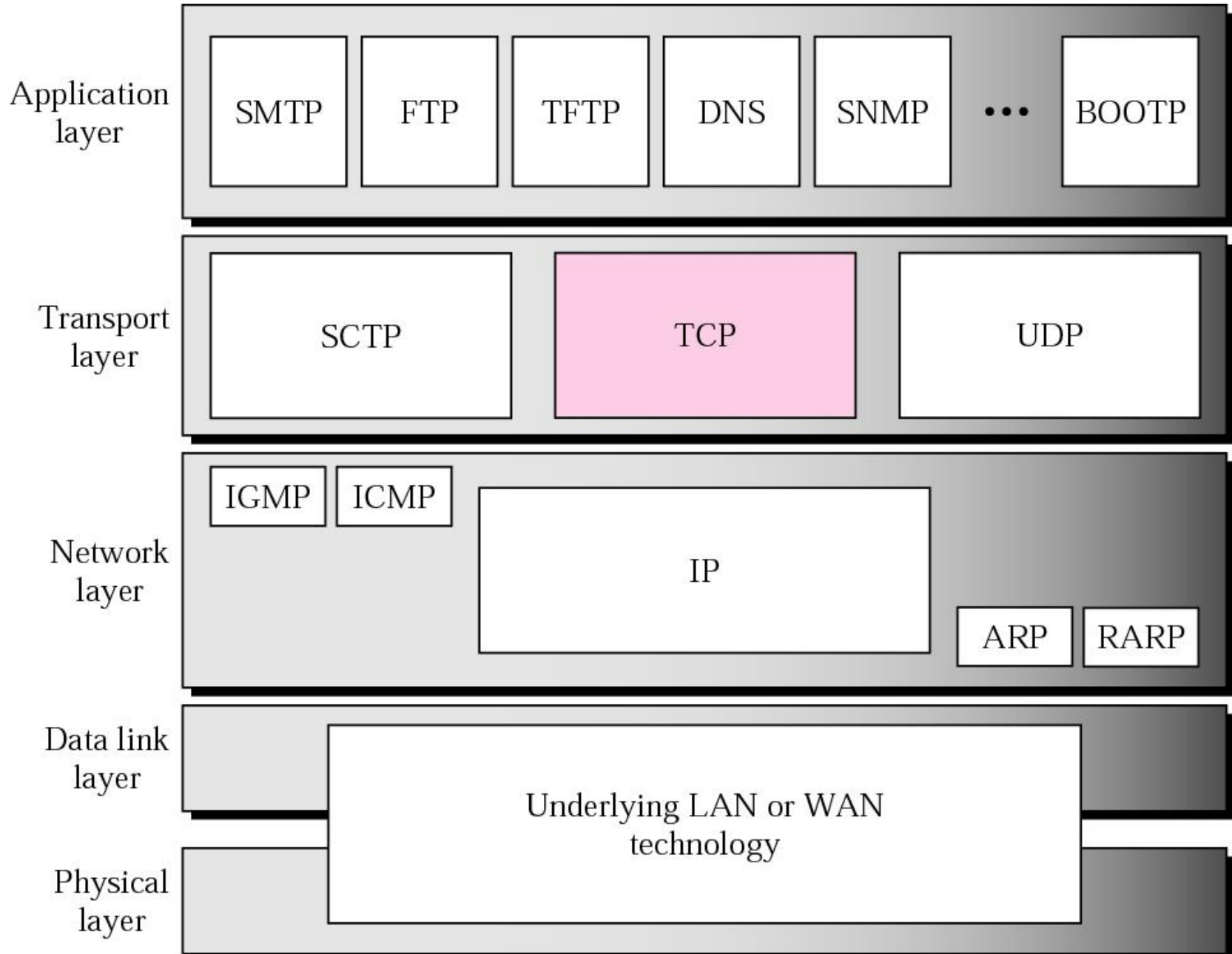
XI - Kompozitna adresna jedinica

- Da bi se koristile usluge transportnog sloja neophodan je par **klijentska /serverska adresa socket-a**.
- Ova četiri podatka su **deo IP zaglavlja i UDP/TCP zaglavlja**.
- IP zaglavlje sadrži **IP adrese** a UDP/TCP zaglavlje sadrži **brojeve porta**
- OSI i TCP/IP referentni modeli omogućavaju **razvoj različitih transportnih protokola** a danas se najčešće koriste **Transmission Control Protocol (TCP)** i **User Datagram Protocol (UDP)**.
- TCP protokol je protokol **sa uspostavom veze** koji omogućava **pouzdan prenos podataka** putem ostvarivanja virtuelne veze, **kontrole grešaka, kontrole redosleda segmenata i prilagođavanje brzine slanja podataka** prijemnoj moći odredišta.
- UDP protokol ne poseduje ove funkcionalnosti jer je to protokol **bez uspostave veze i bez ikakvih garancija** da su podaci stigli na odredište
- Aplikacije kao što su **sistem imena domena (DNS)**, ili **protokol za nadzor i upravljanje (SNMP)**, koriste UDP protokol.
- TCP protokol se koristi kod servisa **kod kojih je neophodna tačnost** na račun performansi a UDP protokol **u obrnutim situacijama**.

XI - Protokoli transportnog sloja

- **TCP** - Transmission Control Protocol;
- **UDP** - User Datagram Protocol;
- **SPX** - Sequenced Packet Exchange;
- **SCTP** - Stream Control Transmission Protocol;
- **ATP** - AppleTalk Transaction Protocol;
- **CUDP** - Cyclic UDP;
- **DCCP** - Datagram Congestion Control Protocol;
- **FCP** - Fiber Channel Protocol;
- **NBF** - NetBIOS Frames Protocol;
- **SST** - Structured Stream Transport

XI - Protokoli transportnog sloja



XI-Transmission Control Protocol - TCP

- TCP je protokol dizajniran da za podatke koristi nizove bajtova i obezbedi pouzdan prenos podataka u oba smera (*full-duplex*).
- Protokol je pogodan za rad na komunikacionim medijumima visoke pouzdanosti (UTP, optički kablovi) a pokazuje slabije performanse na medijumima sa velikim % grešaka u prenosu (bežična komunikacija).
- TCP protokol je jedan od najčešće korišćenih protokola na transportnom nivou kada su u pitanju Internet i klasične lokalne mreže.
- Ovaj protokol je već godinama u upotrebi i razlog tome je pre svega optimalan rad na Ethernet tehnologiji.
- TCP protokol je inicijalno definisan u dokumentu RFC793
- Parametri TCP protokola određeni su u skladu sa ograničenjima komunikacionih kanala (MTU parametar je ograničen na 536 bajtova)
- Savremena okruženja za razvoj distribuiranih aplikacija najčešće imaju razvijenu podršku za korišćenje TCP-a i/ili protokola višeg nivoa
- To znači da ovaj protokol predstavlja *de facto standard* za razvoj distribuiranog softvera koji zahteva pouzdan prenos podataka.
- Najpopularniji servisi Interneta (WEB, E-mail, FTP) koriste TCP

XI - TCP segmenti

- Osnovna jedinica prenosa podataka kod TCP protokola je **segment**.
- Segment se sastoji od **zaglavlja** (generiše i interpretira sam protokol) i **aplikativnih podataka koje** generiše i preuzima aplikativni sloj
- Aplikativni podaci **nisu obavezan deo segmenta** i izostavljeni su kod kontrolnih segmenata (zaduženih za uspostavljanje i prekid veze).
- Zaglavlje TCP segmenta se sastoji **od polja fiksne dužine**
- Bitska dužina zaglavlja je **5x32 b (20 B)** ukoliko nisu uključene opcije
- Zaglavlje **može biti praćeno opcionim delom**.
- Posle opcionog dela može se poslati **65535-40=65515 B podataka**.
- Segment bez podataka je moguć i obično se koristi **kao poruka potvrde ili poruka za upravljanje**.
- Osim osnovnog dela zaglavlja ono **može sadržati i dodatne opcije**.
- **Svaka opcija zaglavlju dodaje reč od 32 bita**.
- Ako opcija **ne ispunjava 32 bita** preostali bitovi se **dopunjavaju nulama**
- Neke od najvažnijih opcija TCP protokola su MSS (*Maximum Segment Size*), WSOPT (*Window Scale Option*), SACK (*Selective ACK*) i *SACK Permitted*.

XI - Zaglavlje TCP protokola

Source port number (16 b) - port preko koga se vrši šalju segmenti

Destination port number (16 b) - port preko koga se primaju segmenti

Sequence number (32 b) - broj koji označava redni broj segmenta

Acknowledgment number (32 b) - broj koji služi za utvrđivanje koji paketi su regularno isporučeni na odredište.

Header lenght (4 b) - broj reči dužine od 4B koje se nalaze u zaglavlju (podrazumevana vrednost je 5).

Reserved (4 b) – za buduće proširenje

Indikatori (8 bita) :

CWR-*Congestion Window Reduced*

ECE-*Explicit Congestion Notification*

URG - urgentan segment (*urgent*)

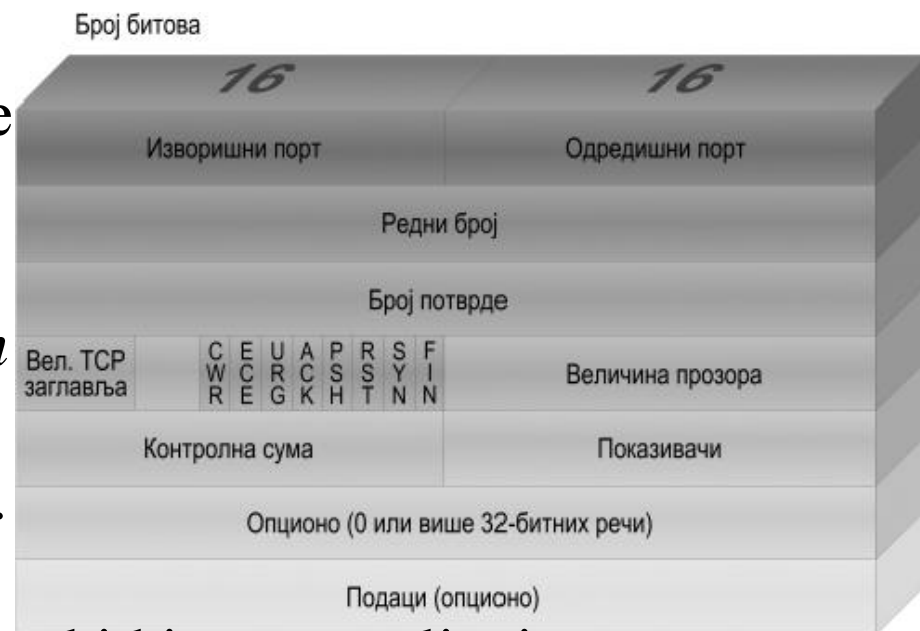
ACK-potvrda prijema (*acknowledge*).

PSH - (push).

RST - označava segment koji inicira prekid i uspostavljanje veze.

SYN- označava segment koji inicira uspostavljanje veze

FIN - označava segment koji inicira prekid veze.



XI - Zaglavlje TCP protokola

Window size (16 bita) - veličina okvira tj. broj bajtova koje odredište može da prihvati preko segmenata koji su potvrđeni. Koristi se za kontrolu toka. Sadrži broj bajtova podataka, počev od rednog broja koji je naveden u polju „potvrđni broj” koje primalac može da prihvati. Kada je vrednost ovog polja jednaka 0 ukazuje da primalac ne može da prihvati više podataka.

TCP checksum (16 bita) - kontrolna suma koja se odnosi na zaglavlje i podatke i koristi se za proveru da li je segment izmenjen tokom transporta

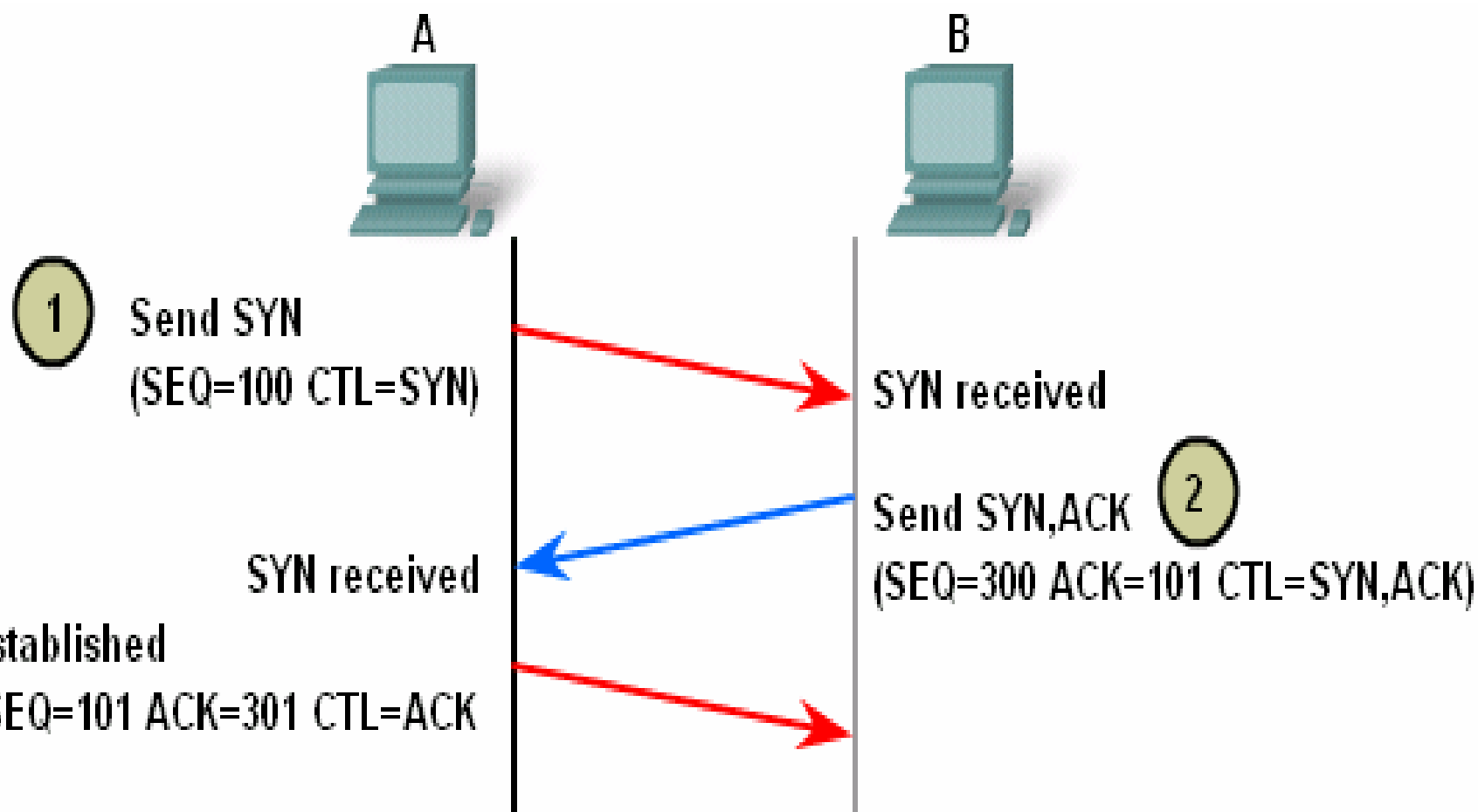
Urgent pointer (16 bita) - koristi se u sprezi sa URG indikatorom i upućuje na poslednji bajt urgentnih podataka. Kada se ova vrednost doda na redni broj segmenta dobija se vrednost poslednjeg bajta u sekvenci podataka koje treba hitno isporučiti aplikaciji.

Opcije (promenljive veličine) - koristi se za dodatne usluge koje nisu uključene u regularno zaglavlje. Jedna od opcija je, na primer, veličina segmenta koja se koristi pri prenosu. Trebalo bi koristiti što veće segmente jer se na taj način prenosi više podataka a zaglavlje od 20 bajtova se lakše amortizuje.

XI - Uspostavljanje TCP veze

1. Klijent serveru šalje segment sa **SYN** indikatorom koji sadrži broj **porta servera** na koji klijent želi da se poveže i **ISN** klijenta.
 2. Server odgovara na SYN zahtev klijenta segmentom koji **sadrži ACK indikator sa ISN-om klijenta uvećanim za jedan**. Segment takođe sadrži SYN indikator sa **ISN (Initial Sequence Number) servera**.
 3. Klijent odgovara na SYN zahtev servera šaljući segment sa **ACK indikatorom koji sadrži ISN servera uvećan za jedan**.
- Ova tri koraka se nazivaju “**rukovanje**” (*handshake*) i ukoliko ne dođe do greške u njima, veza je uspostavljena.
 - Strana koja inicira uspostavljanje veze izvršava **aktivno uspostavljanje veze** (*active open*) dok strana koja prihvata uspostavljanje veze izvršava **pasivno uspostavljanje veze** (*passive open*).
 - Jednom uspostavljena veza ostaje aktivna dok god se **ne zahteva njen prekid** ili dok se jedna od strana **ne izgubi evidenciju o njoj**.
 - To znači da su u periodima kada se veza ne koristi za prenos podataka **moćni prekidi na svim nižim slojevima** (uključujući i fizički).

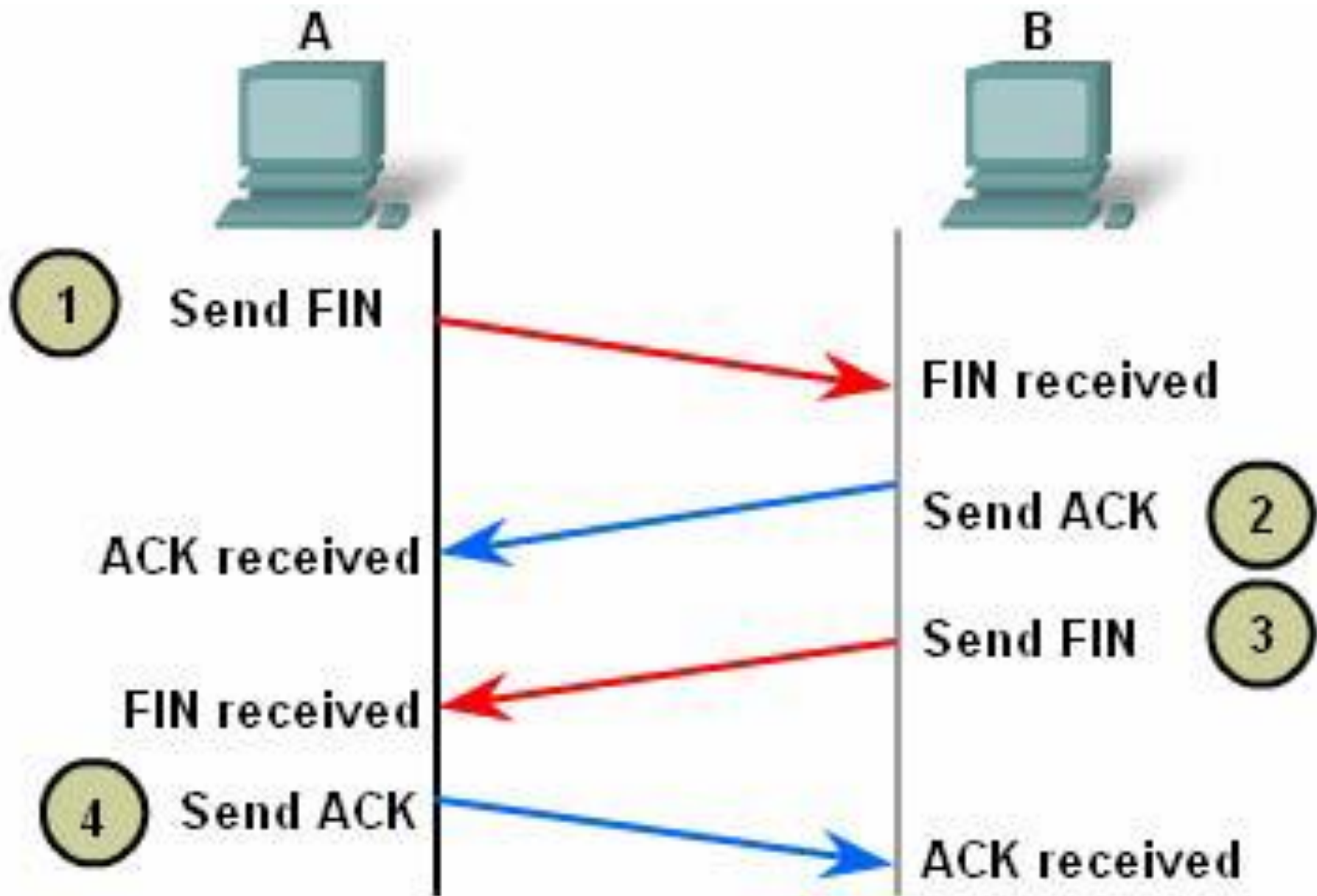
XI - Uspostavljanje i prekidanje veze



XI - Prekidanje TCP veze

- Kod prekida uspostavljene veze potrebno je **da obe strane dobiju informaciju da je veza prekinuta** i da dalji prenos podataka nije moguć.
- Prekid veze **može inicirati svaka od strana**, bez obzira na to koja strana je inicirala uspostavljanje veze.
- Za prekid veze se koriste **FIN** indikatori u TCP zaglavlju a redosled je:
 - 1.Strana koja inicira prekid **šalje TCP segment sa uključenim FIN indikatorom**.
 - 2.Primalac odgovara segmentom sa **uključenim ACK indikatorom**.
 - 3.Za potpuni prekid veze **primalac šalje segment sa uključenim FIN indikatorom**.
 - 4.Inicijator prekida šalje segmenta sa **uključenim ACK indikatorom**.
- Strana koja inicira prekid veze izvršava **aktivan prekid veze** (*active close*) dok strana koja prihvata prekid veze izvršava **pasivan prekid veze** (*passive close*).
- Ukoliko jedna strana **inicira prekid veze** (pošalje FIN i primi ACK) a druga zadrži vezu, takva veza se naziva **polu-zatvorenom** (*half-close*) i njome se nadalje **moгу slati samo podaci u jednom smeru**.

XI - Uspostavljanje i prekidanje veze



XI - Adresiranje

- Kada aplikacioni proces želi da uspostavi vezu sa drugim procesom **potrebno je da označi proces**, tj. da definiše njegovu transportnu adresu
- Termin koji se koristi je **pristupna tačka usluge transportnog sloja** (TSAP-*TransportServiceAccessPoint*)-na Internetu se nazivaju portovi
- Na mrežnom sloju se ove tačke (adrese mrežnog sloja) nazivaju **pristupne tačke usluge mrežnog sloja** NSAP (IP adrese su NSAP-ovi).
- Aplikacioni procesi na strani klijenta i servera **moгу da se priključe na pristupnu tačku transportne usluge** (TSAP) da bi uspostavili vezu sa udaljenom pristupnom tačkom transportne usluge.
- Ove veze se realizuju **preko pristupnih tačaka mrežne usluge** (NSAP)
- Svaka adresa korisnika je **određena parom** (stanica, port).
- Port je **promenljiv i predstavlja određenog korisnika**.
- Identifikacija transportne celine **nije potrebna** pošto obično **postoji samo jedna celina za određeni tip protokola**.
- Port se nalazi **u transportnom zaglavlju** i na odredištu ga može upotrebiti odredišni transportni sloj.

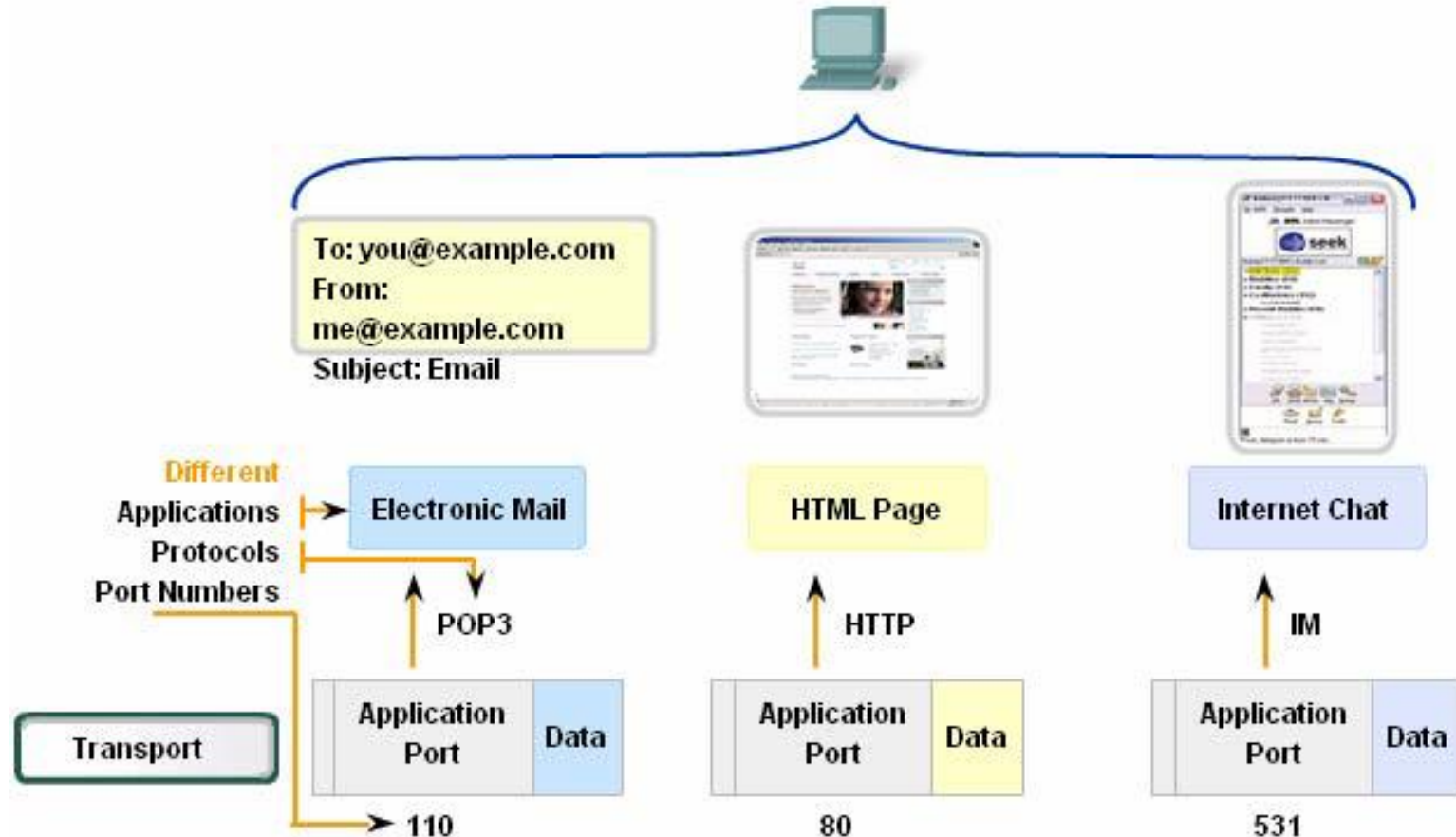
XI - Adresiranje

Kako inicijator veze zna adresu odredišnog transportnog korisnika?

1. **Korisnik zna unapred odredišnu adresu.** U osnovi to je funkcija konfigurisanja sistema kod koje postoji proces koji se isključivo bavi prikupljanjem statističkih podataka za određeni broj korisnika.
2. Neke od opšte korišćenih usluga proglašene su „**poznatim adresama**” kao što su FTP, SMTP i neki drugi standardni protokoli;
3. **Obezbeđeni su serveri imena.** Korisnik zahteva uslugu koristeći neko generičko ili globalno ime. Zahtev se šalje ka serveru imena koji **sadrži bazu adresa** iz koje pronalazi traženu adresu i šalje je korisniku. Ova usluga je korisna za aplikacije koje se često koriste ali povremeno menjaju svoju lokaciju. Na primer neki proces može se prebaciti sa jednog računara na drugi u okvir lokalne računarske mreže da bi se obezbedilo ravnomerno opterećenje.
4. U nekim slučajevima **odredišni korisnik je proces koji se po zahtevu treba pokrenuti, odnosno izazvati.** Korisnik inicijator šalje zahtev ka poznatim adresama. Proces - server na tim adresama pokreće nove procese i šalje inicijatoru adresu novopokrenutog procesa.

XI - Multipleksiranje

- **multipleksiranje nagore** (*Upward multiplexing*) - ukoliko više transportnih korisnika deli jednu virtuelnu vezu uspostavljenu između mrežnih celina,
- **multipleksiranje nadole** (*Downward multiplexing*) - ukoliko jedna transportna veza koristi više veza na mrežnom sloju.



XI - Kontrola toka podataka

- **Jednostavan mehanizam** na sloju veze a **složen** na transportnom sloju:
 - kontrola toka na transportnom sloju **uključuje međusobnu interakciju** transportnih korisnika, transportnih celina i usluge mrežnog sloja,
 - **kašnjenje u prenosu između transportnih celina** može biti veliko a to znači da postoji **kašnjenje i u razmeni informacija o kontroli toka**,
 - kašnjenje u prenosu paketa između transportnih celina je u opštem slučaju **znatno duže od vremena potrebnog za njegovo slanje**,
 - sa obzirom na to da transportni sloj **funkcioniše preko nižih slojeva** kašnjenje može da bude veoma promenljivo pa je prilično **otežano korišćenje časovnika** za ponovno slanje izgubljenih podataka.
- Prijemna transportna celina može na primer da rezerviše memorijski prostor za **sve veze zajedno, a ne za svaku pojedinačno**.
- Svaka transportna celina ima **određenu količinu prostora za čuvanje** pristiglih podataka, odnosno **veličinu bafera**.
- TPDU ili segmenti koji pristižu **smeštaju se u bafer**.
- Ako prijemna transportna celina ne može da održi korak sa dotokom segmenata, **baferi se prepunjuju** pa može da dođe do gubitka podataka

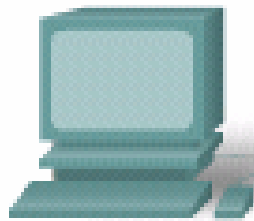
XI - Kontrola toka podataka

- Da bi se sprečila pojava „**prepunjenih bafera**” prijemna transportna celina mora da preduzme korake za zaustavljanje ili usporavanje dotoka
- Da bi rešila ovaj problem prijemna transportna celina može:
 1. da **ne reaguje** ni na koji način,
 2. da **odbije da nadalje prima segmente** (TPDU) od mrežnog sloja; nije poželjno koristiti, jer dolazi do retransmisije paketa
 3. da **koristi protokol klizajućeg prozora** sa fiksnim otvorom prozora; tehnika koja je primenjena kod protokola HDLC kod koje se koristi potvrda koja dovodi do pomeranja (klizanja) prozora. Sa pouzdanom uslugom mrežnog sloja tehnika klizajućeg prozora **radi sasvim dobro**.
 4. da koristi **mehanizam kredita**; ovde transportna celina pored potvrde o paketima koje je uspešno primila **šalje novi parametar - kredit**. On predstavlja otvor prozora ali može da se menja u toku trajanje veze.

XI - Kontrola toka podataka

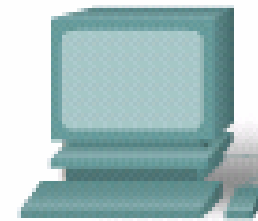
- Polje *Window* u *TCP* zaglavlju-**označava ukupan broj bajtova** koji se mogu poslati pre nego što se čeka na potvrdu
- Prozor” određene veličine **se pomera po nizu aplikativnih podataka** koji se šalju. Zato se mehanizam prozora često naziva i ***Sliding Window***
- Manji prozor-**više čeka, sporije**, a veći prozor - **manje čeka, brže slanje**
- *TCP* **dinamički uspostavlja veličinu prozora** - *Dynamic Window*
- Kada se popuni prozor: **obustavlja se sa slanjem novih segmenata, čeka se potvrda prethodnih poslatih segmenata, potvrđeni segmenti se oslobađaju i oslobađa se prostor u prozoru za slanje novih segmenata**
- Inicijalno se obe strane dogovore o veličini prozora
 - U slučaju gubitka paketa ili greške vrši se **retransmisija** i zahteva se **smanjenje veličine prozora**
 - U slučaju da **nema grešaka** pošiljalac postepeno može da **povećava veličinu prozora**

XI - Kontrola toka podataka

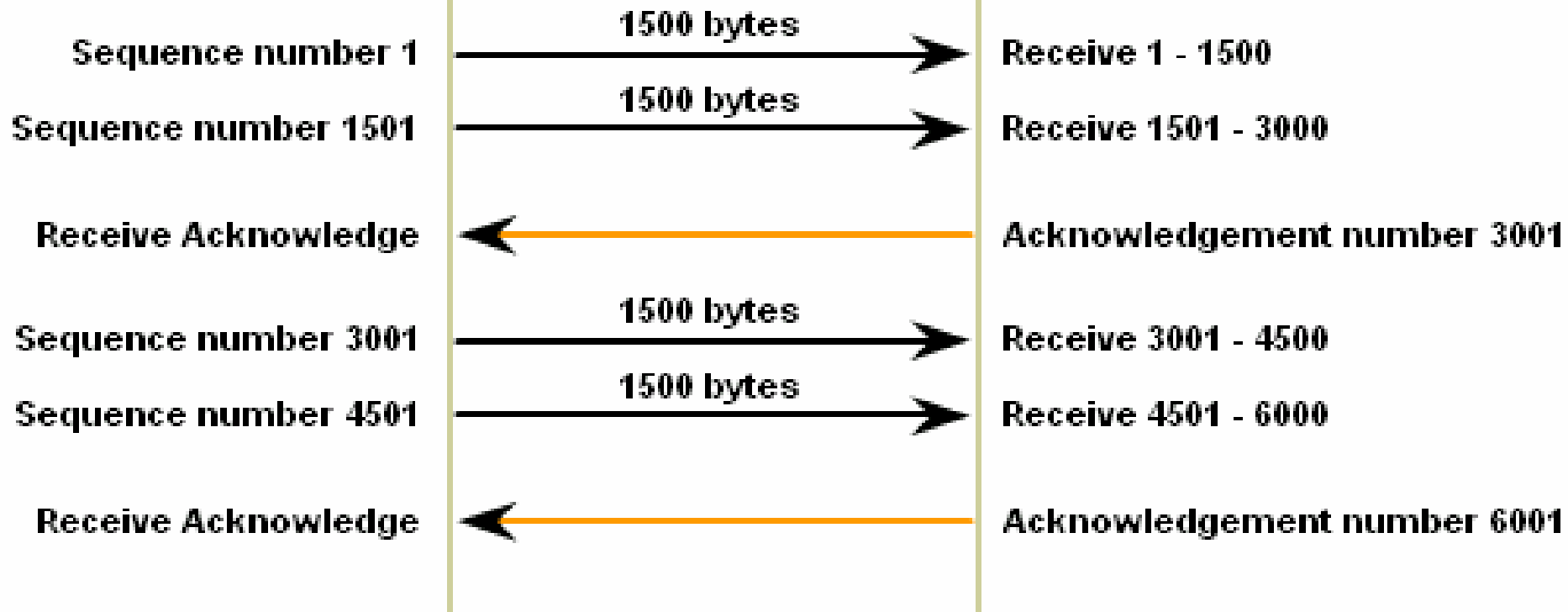


Sender

Window size = 3000



Receiver



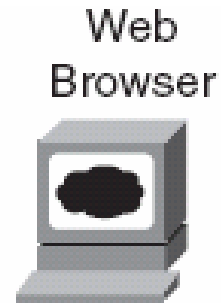
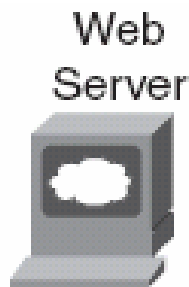
XI - Pouzdanost i performanse

- Performanse TCP protokola su uglavnom **znatno slabije** u poređenju sa protokolima koji rade bez uspostavljanja veze (UDP protokolom).
- TCP protokol **ne podržava** *broadcasting* i *multicasting*
- Glavna karakteristika TCP protokola je **pouzdanost** koja se postiže:
 - ✓ Podatke koje aplikacija dostavlja transportnom sloju TCP **deli u segmente koje šalje pojedinačno** - smanjuje se jedinica nad kojom se vrši kontrola i smanjuje se mogućnost i cena ispravljanja greške.
 - ✓ TCP **zahteva potvrdu** da je svaki od poslanih segmenata isporučen.
 - ✓ TCP **eliminiše iste segmente** koje je mrežni sloj greškom dostavio
 - ✓ TCP **reorganizuje primljene segmente po izvornom redosledu** bez obzira na redosled kojim ih mrežni sloj dostavlja.
 - ✓ TCP prilikom slanja segmenta **generiše kontrolne parametre vezane za zaglavlje i sadržaj** koji mu pomažu da kontroliše ispravnost segmenata.
 - ✓ TCP **prilagođava frekvenciju slanja prihvatnoj moći primaoca** čime sprečava odbacivanje segmenata i smanjuje mogućnost greške.

XI – Pouzdanost i performanse

- Svaki poslani segment **zahteva potvrdu** (ACK) da je uspešno pristigao
- Potvrda se **čeka određeno vreme** – *timeout* gde se za svaki poslani segment startuje **poseban tajmer**
- Vreme tajmera **nešto je veće od vremena putovanja segmenta** od izvorišta do odredišta i nazad – RTT (*Round Trip Time*)
- Ako **nema grešaka**, primalac će da **pošalje potvrdu** za poslednji bajt poslednjeg segmenta tj. potvrdu da su svi segmenti uspešno primljeni
- Pošiljalac **nastavlja sa slanjem novih segmenata** i čeka novu potvrdu da su segmenti uspešno pristigli bez grešaka
- Ako primalac prepozna **da je izostao jedan prethodni segment**, onda **potvrđuje prijem poslednjeg ispravnog** segmenta u nizu
- Ako **istekne tajmer nekog segmenata**, a ne stigne njegova potvrda, sprovodi se **retransmisija samo tog segmenta**, u nadi da su naredni segmenti uspešno pristigli
- Kada primalac dobije retransmitovani segment, **popuniće “rupu” u nizu** primljenih podataka i potvrdiće poslednji ispravan segment

XI - Oporavak od greške



1000 Bytes of Data, Sequence = 1000

1000 Bytes of Data, Sequence = 2000

1000 Bytes of Data, Sequence = 3000

No Data, Acknowledgment = 2000

1000 Bytes of Data, Sequence = 2000

No Data, Acknowledgment = 4000

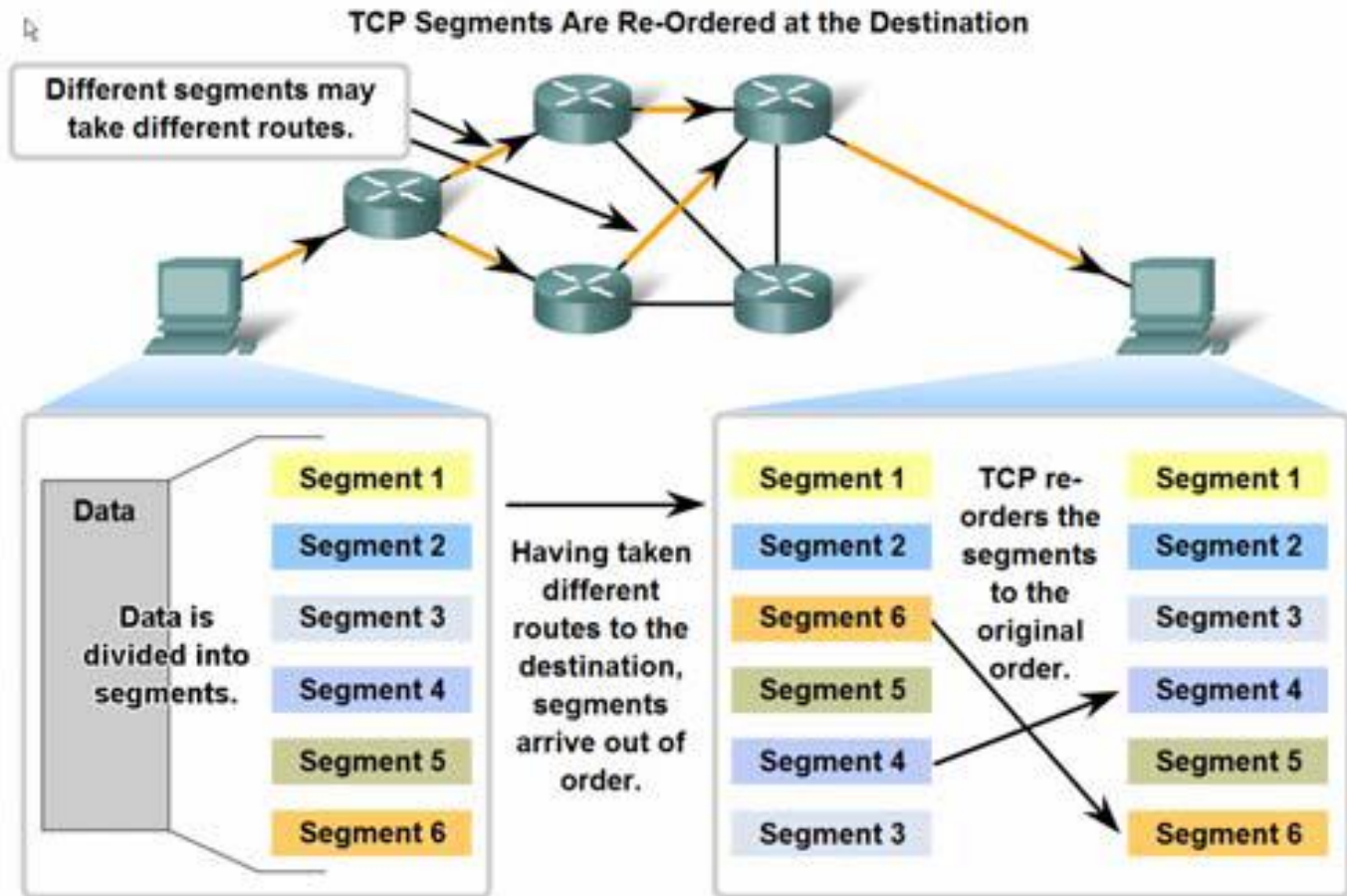
He Lost the Segment with Sequence = 2000. Resend It!

I Probably Lost One. ACK What I Got in Order!

I Just Got 2000-2999, and I Already Had 3000-3999. Ask for 4000 Next.

XI-Rekonstrukcija redosleda segmenata

- U slučaju **više putanja između izvorišta i odredišta** različiti segmenti uzimaju različite putanje (*Load Balancing*)
- Može da dođe do **permutacije segmenata** – promene redosleda prijema
- Prijemna strana će da rekonstruiše originalni redosled na osnovu rednog broja segmenta (*Sequence Number*)



XI - User Datagram Protocol (UDP)

- UDP ne omogućava pouzdan prenos podataka putem ostvarivanja virtuelne veze, kontrole grešaka, kontrole redosleda segmenata i ne prilagođava brzinu slanja podataka prijemnoj moći odredišta.
- Namena UDP protokola je prenos sa što manjim vremenskim neslaganjima između generisanja podataka na strani izvorišta i prijema podataka na odredištu.
- Glavne primene UDP-a su kod protočnog prenosa glasa i video materijala (Internet telefonija, video konferencije, računarske igrice,...)
- Prednost UDP protokola u odnosu na TCP jeste mogućnost *broadcast* slanja podataka - istovremenog slanja podataka svim članovima mreže
- Jedinica za prenos podataka UDP protokola je datagram.
- Struktura UDP datagrama je znatno jednostavnija od strukture TCP segmenata jer je izostavljena većina kontrolnih informacija.
- Nedostatak kontrolnih informacija čini UDP protokol znatno efikasnijim u smislu manjeg opterećenja komunikacionog kanala kontrolnim podacima i manjeg opterećenja primaoca datagrama u smislu jednostavnijeg procesiranja datagrama.

XI - User Datagram Protocol (UDP)

Source port number (16 bita)

Destination port number (16 bita)

Length (16 bita)

Checksum (16 bita)

Изворишни порт (16 битова)	Одредишни порт (16 битова)
Дужина (16 битова)	Контролна сума (16 битова)
Подаци...	

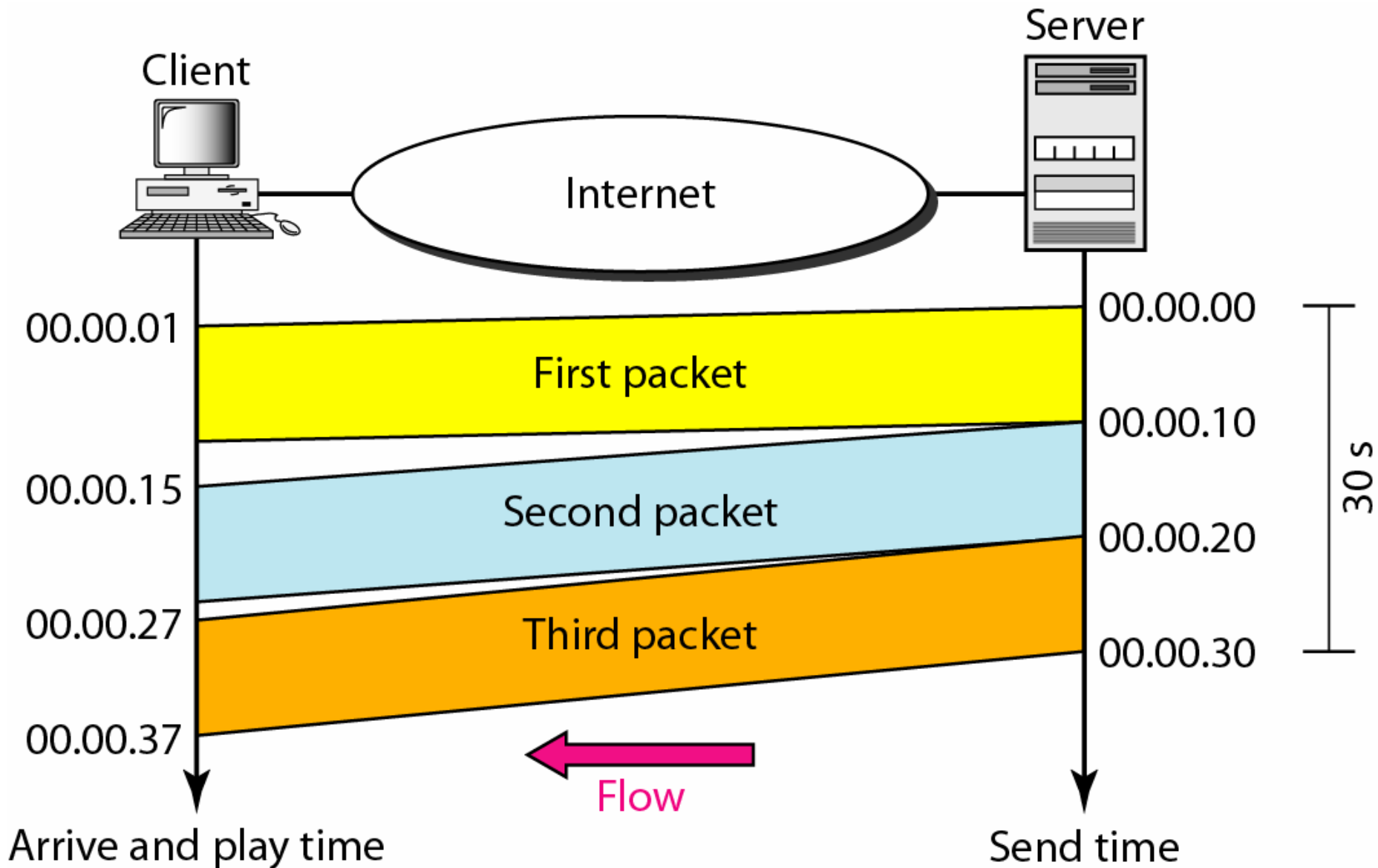
Karakteristike UDP protokola

- ✓ **Connectionless** - ne uspostavlja konekciju pre slanja podataka i svaki datagram se nezavisno prenosi
- ✓ **Unreliability** – nepouzdan jer se izgubljeni ili oštećeni paketi (datagram) se ne mogu retransmitovati, nema kontrole i provere redosleda pristizanja paketa a ako je potrebno, aplikacija koja koristi UDP mora da implementira ove funkcije
- ✓ **Low overhead**-brz, jednostavan, ne zahteva velike mrežne i CPU resurse
- **Neotpornost UDP protokola** na greške pri prenosu, moguće je nadomestiti funkcionalnostima u aplikativnom sloju.
- Neke aplikacije koje koriste UDP **primenjuju ovakav pristup** (TFTP).
- Korišćenjem UDP protokola aplikacije uglavnom očekuju **maksimalne performanse prenosa** bez obzira na greške

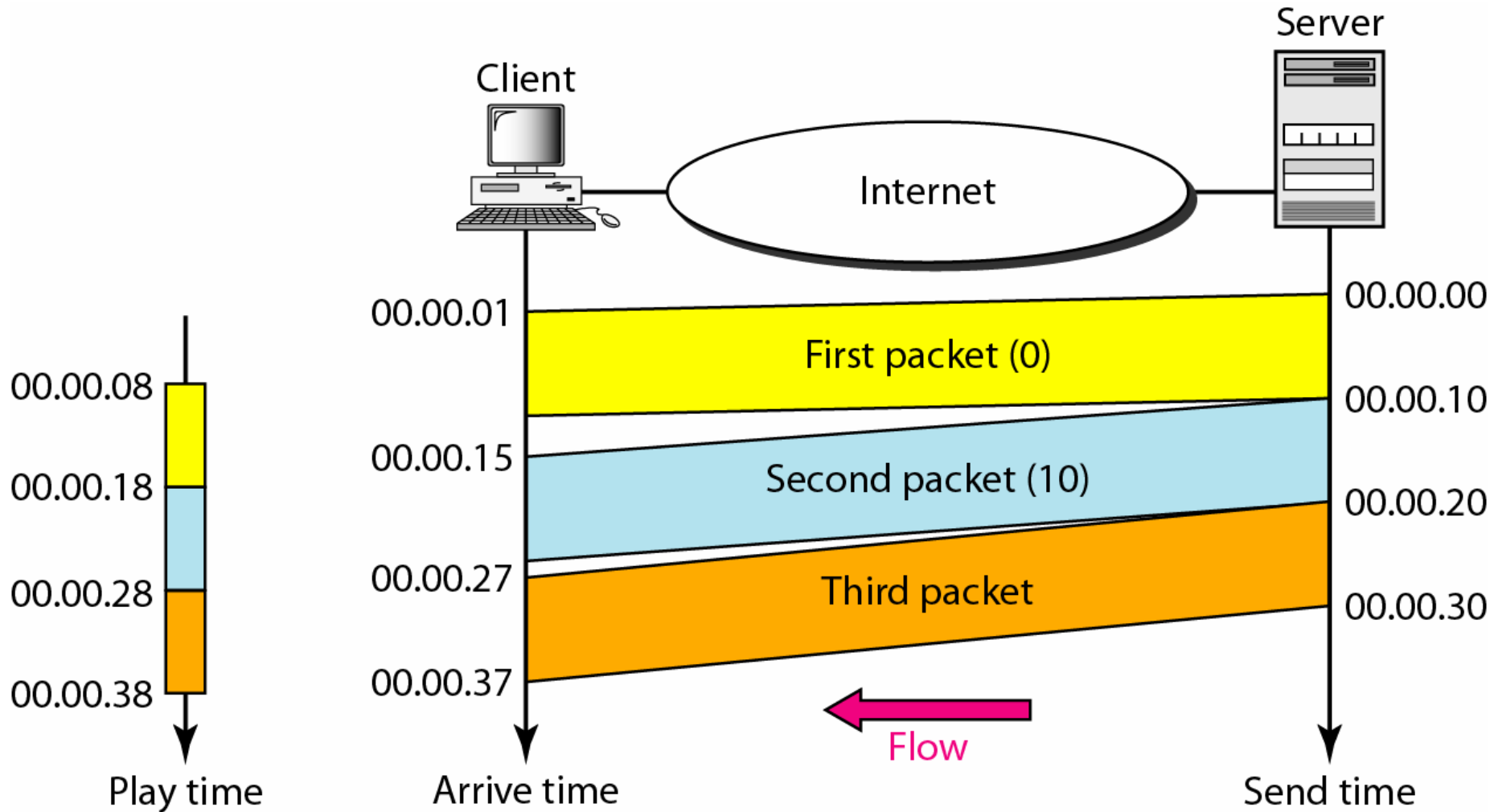
XI - Primena UDP protokola

- Mnoge aplikacije **ne zahtevaju pouzdanost prenosa** i UDP funkcije su dovoljne (npr. periodičan prenos manje količine podataka)
 - DNS - *Domain Name System*
 - SNMP - *Simple Network Management Protocol*
 - DHCP - *Dynamic Host Configuration Protocol*
 - RIP - *Routing Information Protocol*
 - TFTP - *Trivial File Transfer Protocol*
- Nekim aplikacijama bi TCP **predstavljao problem** jer ima **veliko zaglavlje**, a podaci se kontinualno prenose **u malim količinama**, kontrola greške, retransmisija, dinamički prozor **moгу da uspore podatke** i degradiraju prenos podataka kod: *Real-Time* saobraćaj, *VoIP–Voice Over IP*, *Video Streaming – IP TV*.
- Ovakve aplikacije imaju **stroge zahteve** koji moraju da se ispune:
 - **malo kašnjenje** (*delay*) – *do 200 ms za interaktivan razgovor*
 - **mala varijacija kašnjenja** - džiter (*jitter*) *+/- 30 ms*
 - **manji gubitak paketa se može tolerisati**
- Ako se izgubi neki paket, **retransmisija bi narušila kašnjenje i džiter**

XI - Primena UDP protokola



XI - Primena UDP protokola



XI - Sequenced Packet Exchange (SPX)

- SPX protokol je protokol transportnog sloja koji se koristi kao podrazumevani transportni protokol kod Novell NetWare OS.
- Ovaj protokol se koristi u kombinaciji sa IPX protokolom.
- SPX protokol omogućava pouzdan prenos podataka sa ostvarivanjem veze i najbliže se može uporediti sa TCP protokolom.
- SPX protokol je razvijen na osnovu *Sequenced Packet Protocol* (SPP)
- Početkom 90-ih godina XX veka, IPX/SPX kombinacija protokola je u tom periodu predstavljala jedno od najpopularnijih rešenja za LAN-ove u okviru Novell-ovih mreža
- Pojavom TCP/IP stek protokola ovi protokoli odlaze u istoriju, pa čak i *NetWare OS* počev od verzije 5 podržavaju i komunikaciju putem TCP/IP protokola.

XI-Stream Control Transmission Protocol

- SCTP je protokol koji **ima sličnosti sa TCP i sa UDP** protokolom.
- SCTP **omogućava pouzdan prenos podataka** brzinom koja je prilagođena prihvatnoj moći odredišta.
- SCTP **omogućava isporuku podataka na više odredišta** istovremeno.
- Njegova prevashodna namena je da omogući prenos PSTN (*Public Switched Telephone Networks*) signalnih poruka preko IP mreže.
- SCTP je **relativno mlad protokol** tako da se danas retko koristi
- Podrška za ovaj protokol u OS-a je **uglavnom u eksperimentalnoj fazi**.
- Ovaj protokol je pomenut zbog svoje važnosti kao protokol koji u budućnosti **može objediniti prednosti TCP i UDP** protokola.

Hvala na pažnji !!!



Pitanja

? ? ?