

Računalniška omrežja

Transportni sloj

doc. dr. Peter Rogelj (peter.rogelj@upr.si)

Vsebina

- Uvod in storitve transportnega sloja.
- Elementi transportnih protokolov
 - Naslavljanje
 - Vzpostavljanje povezave
 - Prekinitev povezave
 - Nadzor pretoka
 - Multipleksiranje
- Internetni transportni protokoli
 - UDP
 - TCP
 - SCTP
- Mrežno programiranje

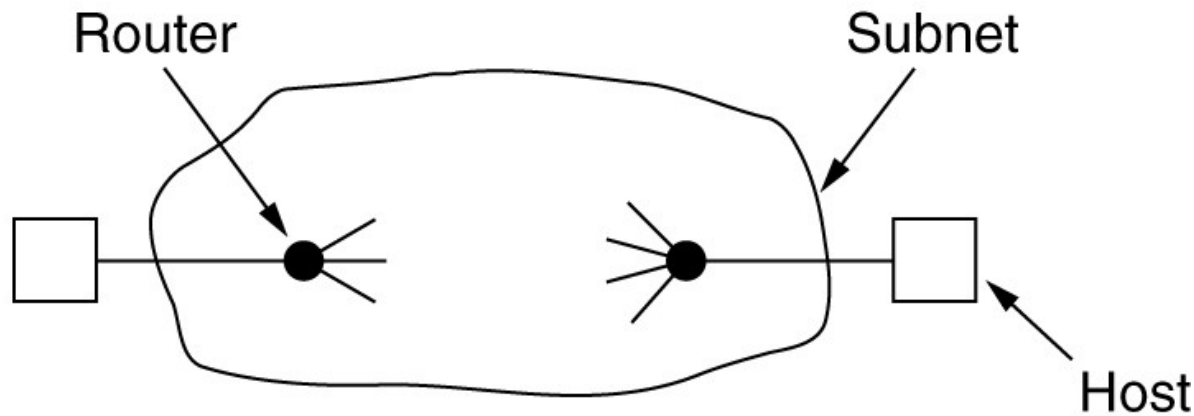
Uvod

Transportni sloj

Transportni sloj

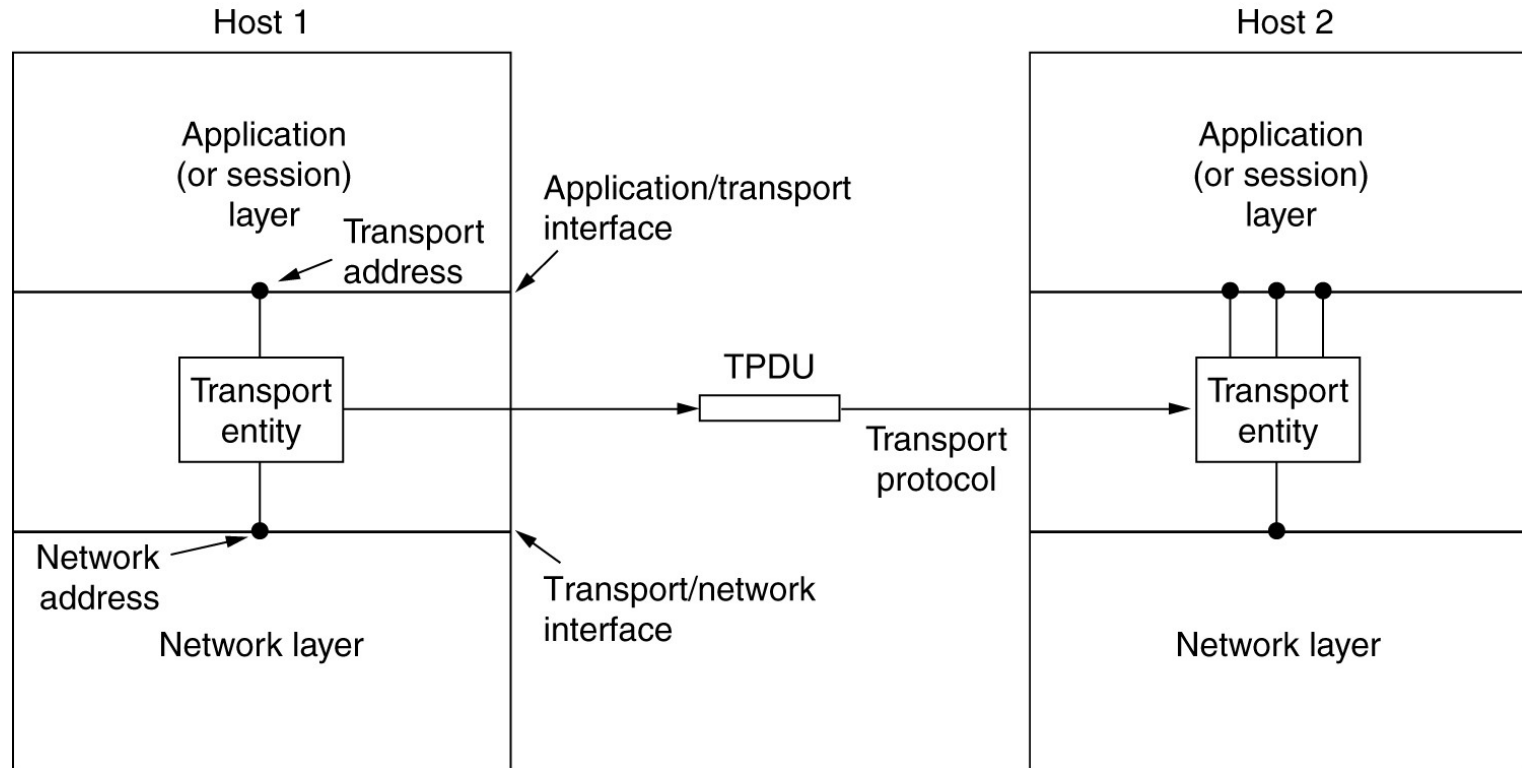
- Transportni (tudi prenosni) sloj zagotavlja zanesljiv in učinkovit prenos podatkov preko omrežja od izvirnega do ponornega vozlišča.
 - Poveča zanesljivost mrežnega sloja.
- Prvi sloj 'od konca do konca' (ang. end-to-end).
 - Deluje izključno na končnih vozliščih in ni prisoten na usmerjevalnikih.

Okolje transportnega sloja



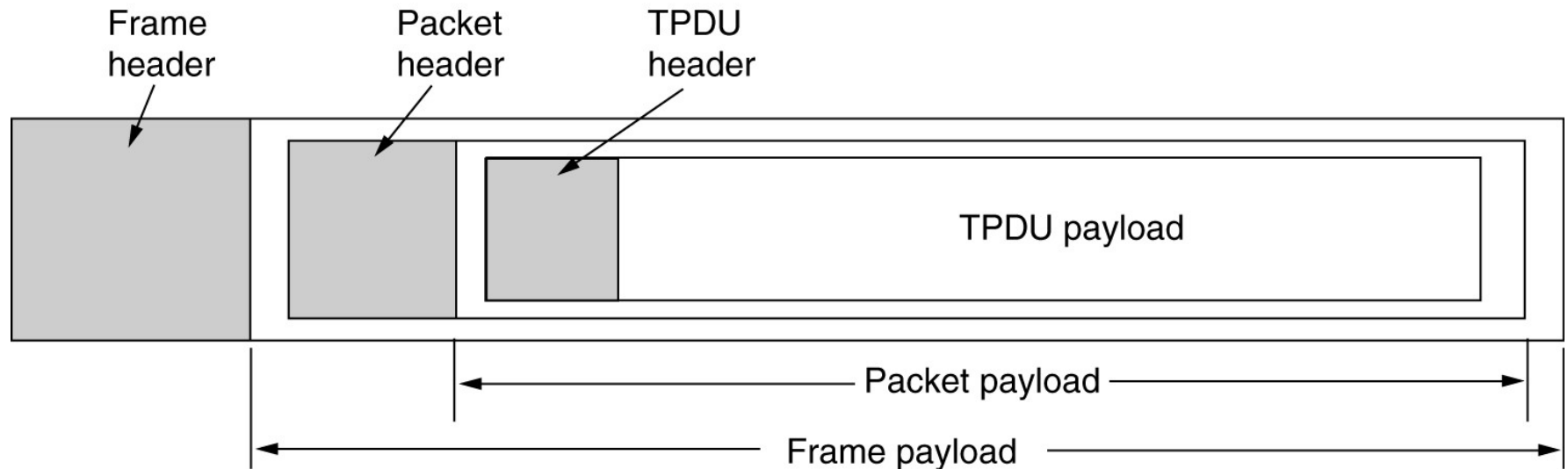
- Transportni sloj se ukvarja s prenosom podatkov med končnima vozliščema.
- Uporablja storitve mrežnega sloja, vendar ga ne zanimata pot in način prenosa po omrežju!

Transportni sloj



- TPDU: podatkovna enota transportnega sloja (ang. Transport Protocol Data Unit)
- Vmesnik do aplikacijskega sloja določa transportne naslove.

Podatkovne enote



- TPDU: podatkovna enota transportnega sloja
- Paket (ang. Packet): podatkovna enota mrežnega sloja
- Okvir (ang. Frame): podatkovna enota linijskega sloja

Storitve transportnega sloja

- **Nepovezavna transportna storitev**
 - Posredovanje podatkovnih enot (sporočil) od pošiljatelja do prejemnika.
 - Nezanestljiva storitev.
 - Tipična uporaba za pretočne vsebine ter storitve, ki ne zahtevajo velike zanesljivosti.
- **Povezavna transportna storitev**
 - Zanesljiva storitev.
 - Zahteva vzpostavitev transportne povezave med končnima vozliščema.
 - Tipična uporaba za storitve, ki zahtevajo visoko zanesljivost.

Elementi transportnih storitev

- Elementi nepovezavne storitve
 - SEND data (pošiljanje sporočila)
 - RECEIVE data (prejemanje sporočila)
- Elementi povezavne storitve
 - LISTEN (čakaj na povezavo drugega vozlišča)
 - CONNECT (aktivna vzpostavitev povezave)
 - SEND data (pošiljanje podatkov)
 - RECEIVE data (prejemanje podatkov)
 - DISCONNECT (prekinitev povezave)

Uporaba mrežnega sloja

- Omrežje nudi storitev prenosa paketov, ki je lahko povezavna ali nepovezavna.
- Datagramska omrežja z dinamičnim usmerjanjem imajo naslednje omejitve:
 - Vrstni red prejetih paketov je lahko različen od vrstnega reda pošiljanja.
 - Izražena lastnost hranjenja omrežja: paket se v omrežju lahko hrani več sekund preden prispe do cilja!
 - Mrežni sloj pogosto ne nudi zanesljive storitve (ni preverjanja pravilnosti prenosa).

Elementi transportnih protokolov

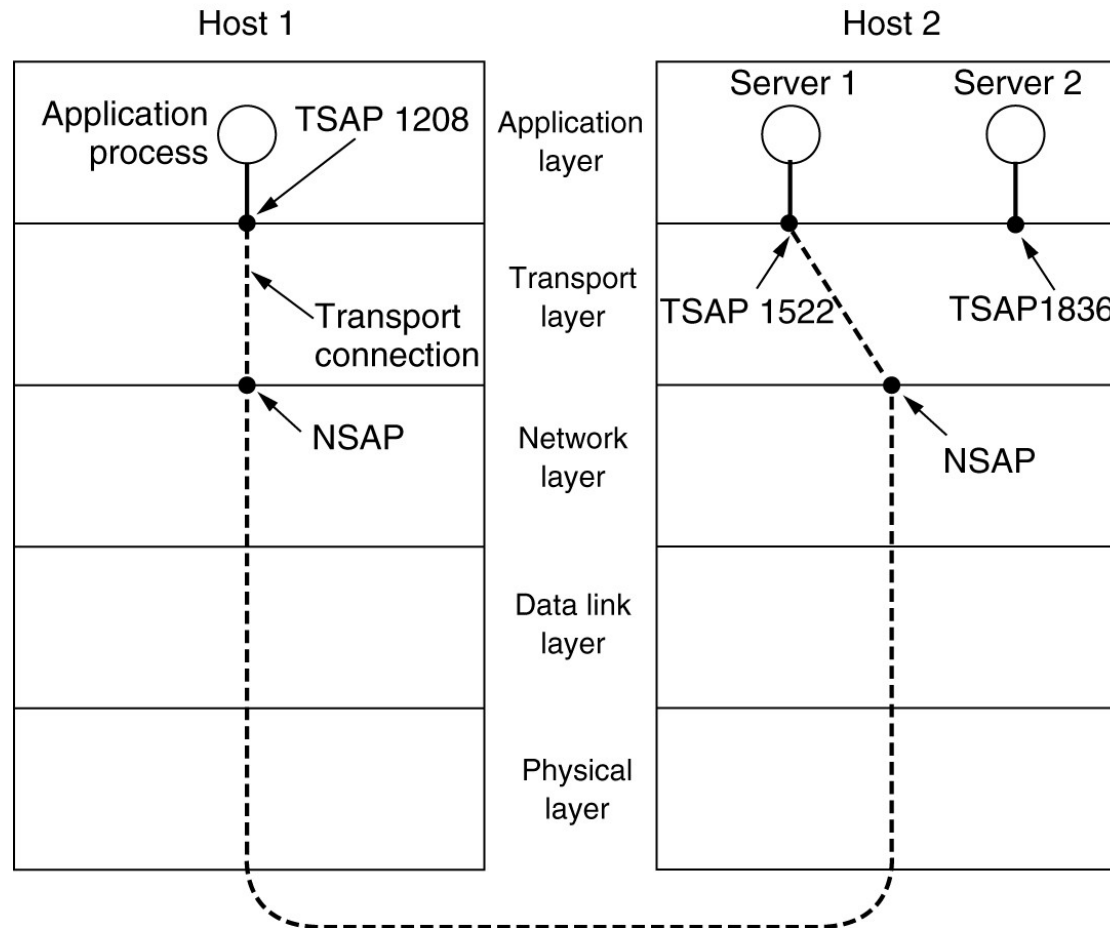
Elementi transportnih protokolov

- Naslavljanje
- Vzpostavljanje povezave
- Prekinitev povezave
- Nadzor pretoka
- Multipleksiranje

Naslavljanje

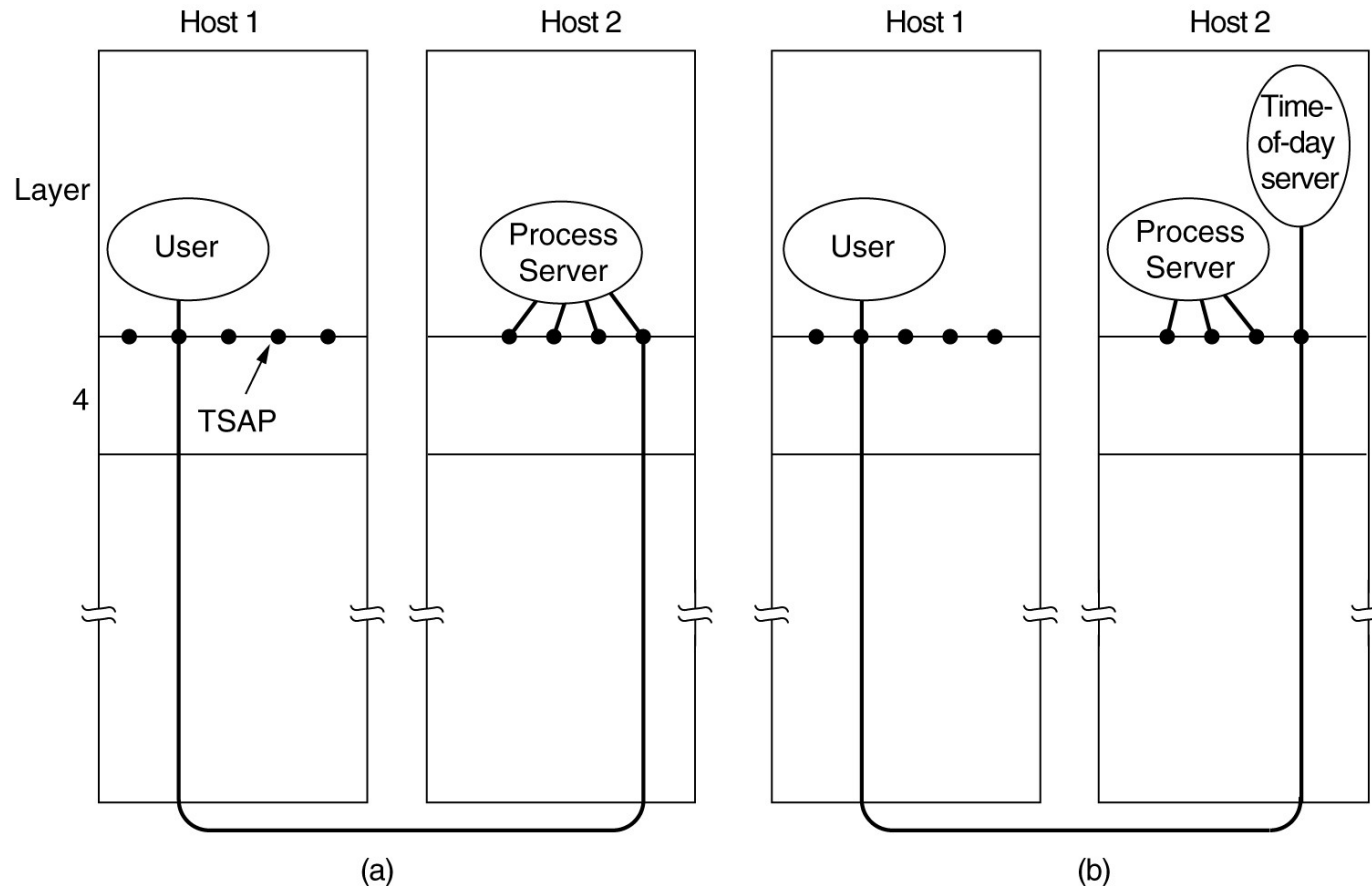
- Ko neka aplikacija želi komunicirati z oddaljeno aplikacijo, mora poznati nejen naslov!
- TSAP - transportni naslov aplikacije (ang. Transport Service Access Point)
 - V omrežju Internet so to vrata (ang. port).
 - V omrežju ATM se imenuje AAL-SAP.
- TSAP je nalogen NSAP – mrežni naslov (ang. Network Service Access Point)
 - NSAP v omrežju Internet je IP naslov.
- TSAP in NSAP skupaj določata končni naslov (ang. endpoint address)
 - V omrežju Internet IPnaslov:vrata, npr: 88.200.63.148:80

TSAP, NSAP, povezave



Primer mrežne povezave med aplikacijama na dveh vozliščih.

Posredovalni strežnik



Posredovalni strežnik čaka na vzpostavitev povezave (a), ki jo preda strežniku (b)

Vzpostavljanje povezave

- Kako?
 - ☐ Odjemalec pošlje
CONNECTION REQUEST TPDU
 - ☐ Strežnik odgovori s
CONNECTION ACCEPTED TPDU
 - ☐ Je to zadosti?

- Kaj če omrežje izgubi, shrani ali podvoji pakete?

Črni scenarij

- Uporabnik vzpostavi povezavo z banko ter pošlje zahtevo za nakazilo visoke vsote denarja...
Prekine povezavo.
- Žal se vsi paketi slučajno podvojijo in shranijo na omrežju ter zakasnjeni prispejo do banke.
- Vzpostavi se nova povezava in opravi novo nakazilo. Banka ne ve, da gre za isto nakazilo!
- Kako zanesljivo preprečiti takšno situacijo?

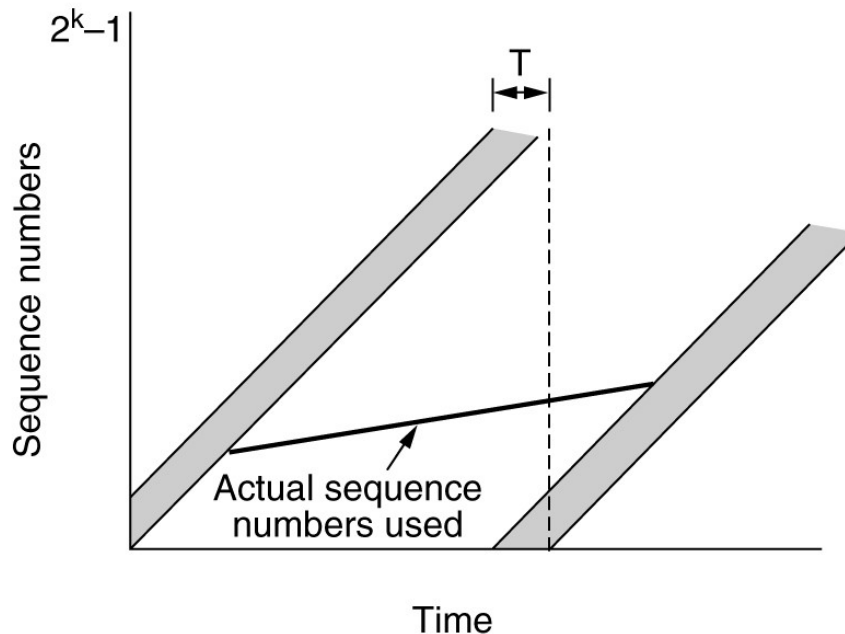
Izbrana rešitev (1)

- Vzpostavitev mehanizma s katerim preprečimo dolgotrajno hranjenje paketov – omejena življenska doba paketov.
- Možnosti:
 - Omejeno načrtovanje podmrežja.
 - Vsak paket vsebuje števec hopov.
 - Časovno žigosanje paketov.

Izbrana rešitev (2)

- Vsak TPDU vsebuje zaporedno številko.
- Števec zaporednih številk mora biti tako velik, da se v življenski dobi paketa zaporedna številka ne ponovi.
- TPDU s staro zaporedno številko se zavrže.
- Enako je potrebno obravnavati tudi potrditve (ACK).

Zaporedne številke in čas

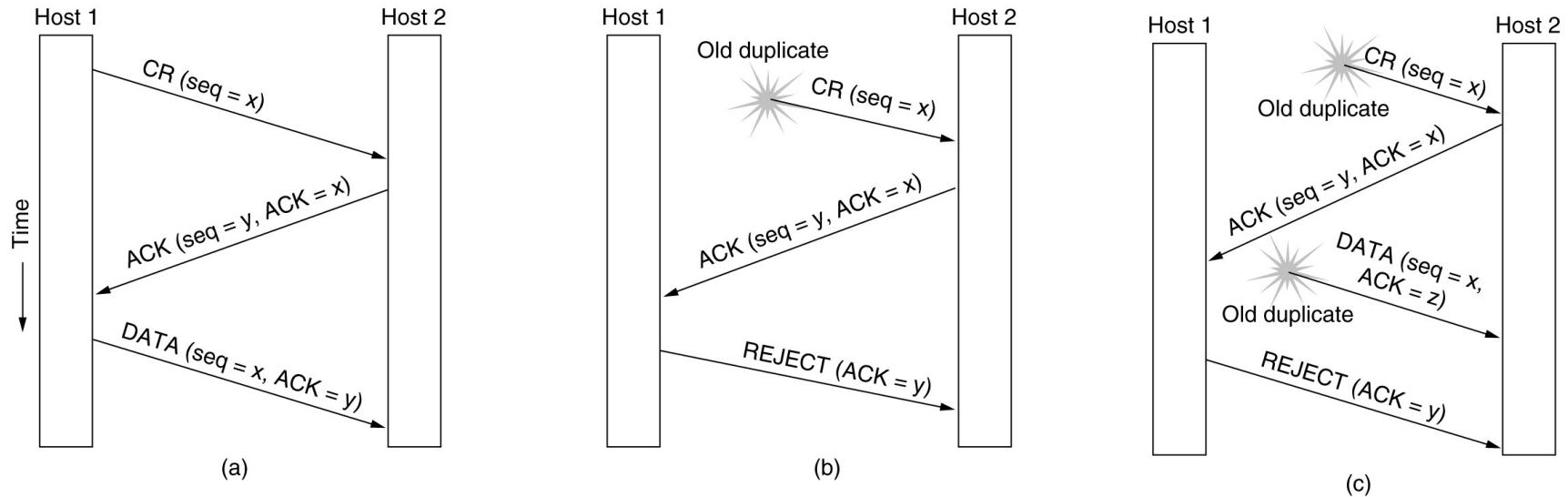


Dovoljeni so le paketi v omejenem pasu v prostoru časa in zaporednih števil.

Izbrana rešitev (3)

- Težava se pojavi ob zrušitvi strežnika (crash) ...
 - Končna vozlišča se morata dogovoriti o začetni zaporedni številki.
- Rešitev je v trikratnem rokovanju (ang. three-way handshake)
 - Vozlišče 1 izbere zaporedno številko x in jo pošlje s CONNECTION REQUEST TPDU.
 - Vozlišče 2 odgovori z ACK TPDU, kjer potrdi x in napove svojo začetno zaporedno številko y.
 - Vozlišče 1 potrdi y v prvem podatkovnem TPDU.

Trikratno rokovanje



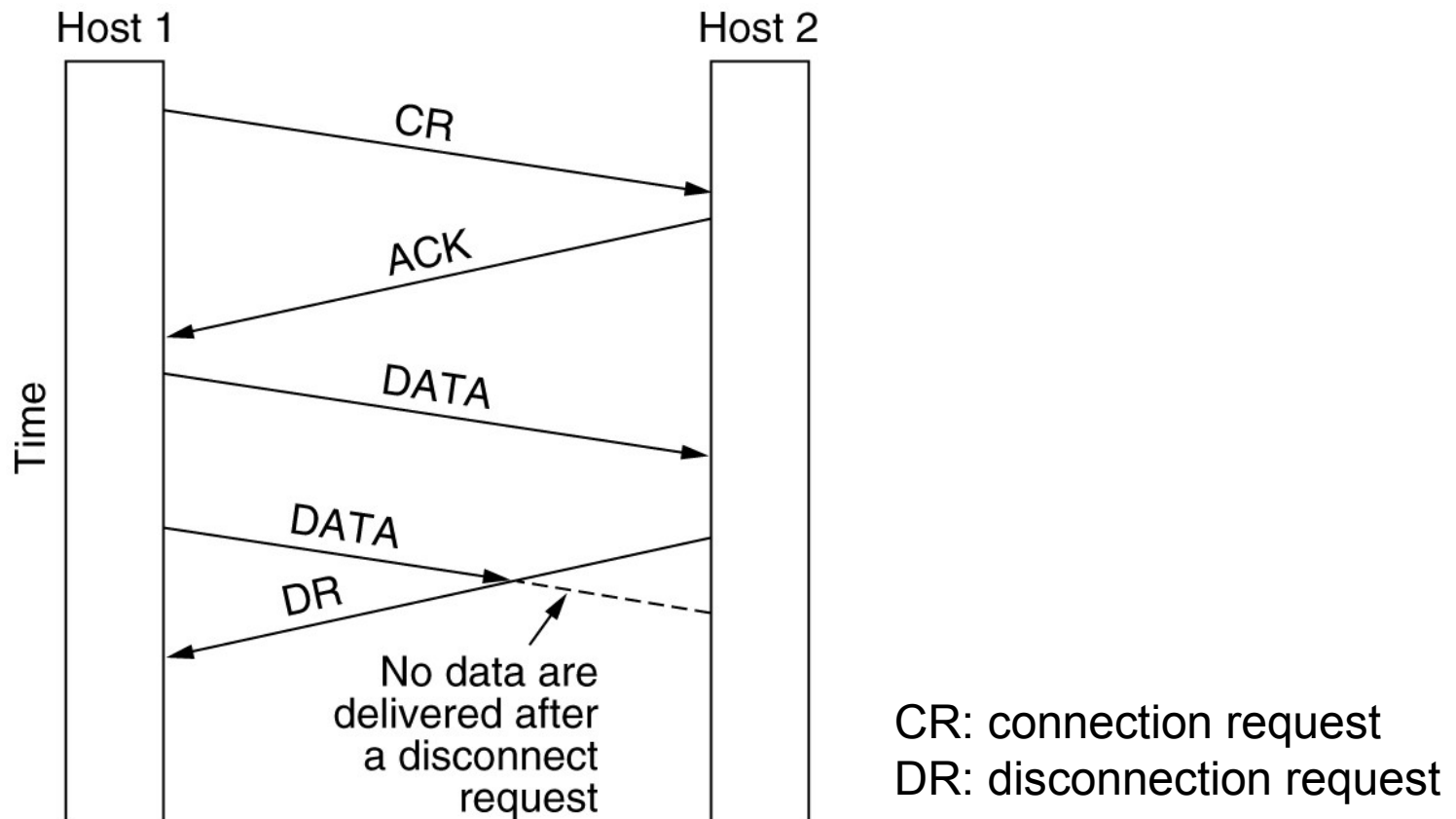
- a) Običajna vzpostavitev povezave.
- b) Situacija, če se pojavi star CONNECTION REQUEST TPDU.
- c) Podvojena CONNECTION REQUEST in ACK.

Prekinitev povezave

- Obstajata dva načina prekinjanja povezave:
 - Asimetrična prekinitev
 - Analogija s telefonskim sistemom – povezava se prekine, ko eden od udeležencev odloži.
 - Simetrična prekinitev
 - Povezava se obravnava kot dve ločeni enosmerni povezavi, kjer je vsako od njih treba prekiniti ločeno.

Asimetrična prekinitev

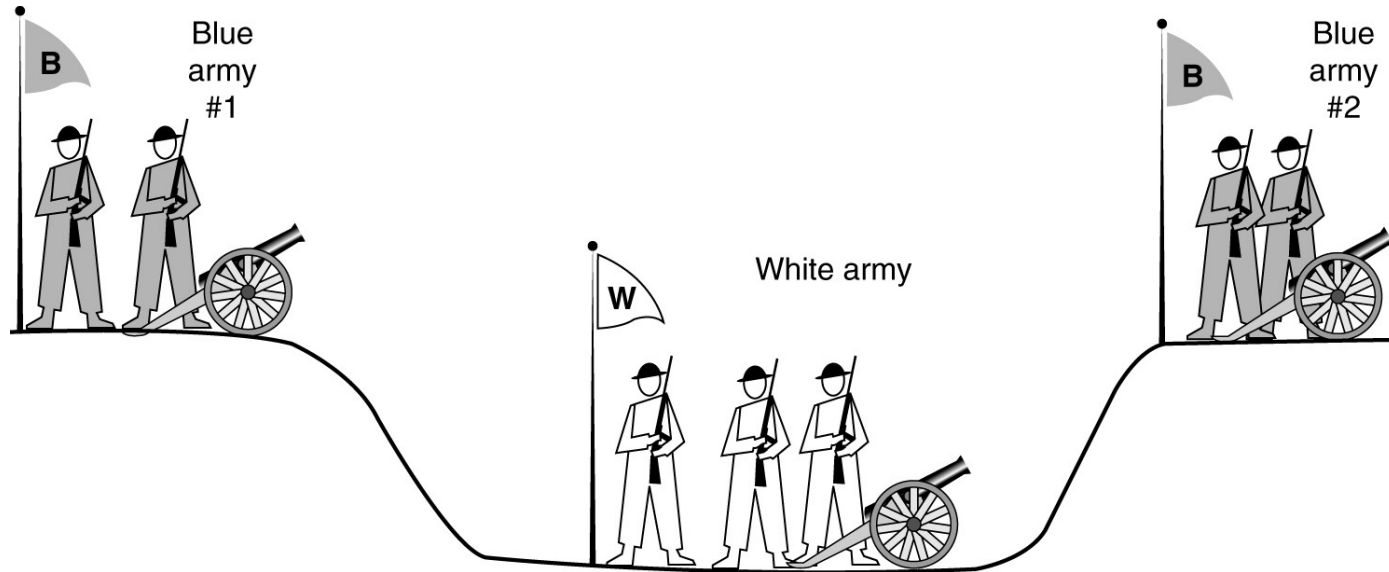
- Asimetrična prekinitev lahko povzroči izgubo podatkov!



Asimetrična prekinitev

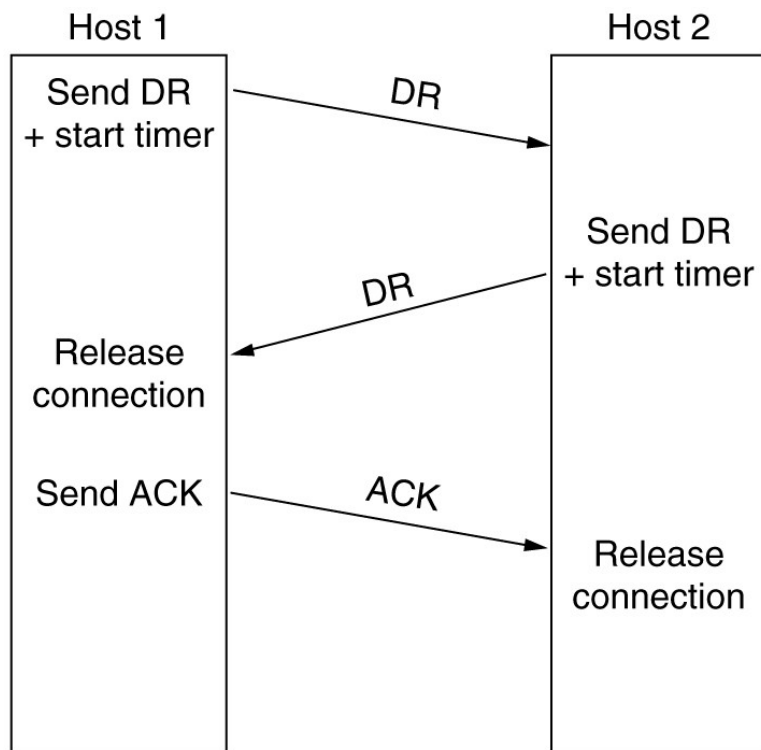
- Potrjevanje prekinitve?
 - ☐ Vozlišče V1 predlaga prekinitev.
 - ☐ Vozlišče V2 se s prekinitvijo strinja in jo potrdi.
- Ali se bo prekinitev zgodila?
 - ☐ Kaj če se potrditev izgubi? Potem V1 ne bovedel, da lahko prekine – ne bo prekinil, da ne pride do izgube podatkov.
 - ☐ Naj potrdi prejem potrdila?
 - ☐ ...
 - ☐ Gre za t.i. problem dveh armad.

Problem dveh armad



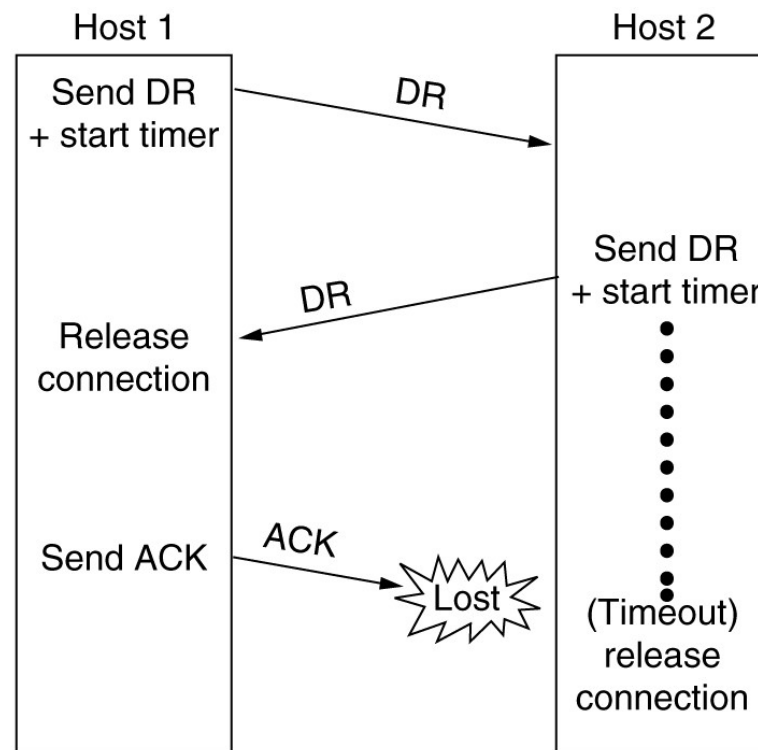
- Modra armada lahko zmaga le, če oba njena dela hkrati napadeta belo armado.
- Kako naj se dogovorita za napad. Komunikacija je možna le preko glasnika, ki ga lahko bela armada ujame... Nezanesljiv prenos.
- Protokol, ki bi rešil problem, ne obstaja!
- Analogija s prekinitvijo (napad = prekinitev).

Prekinitv v treh korakih



(a)

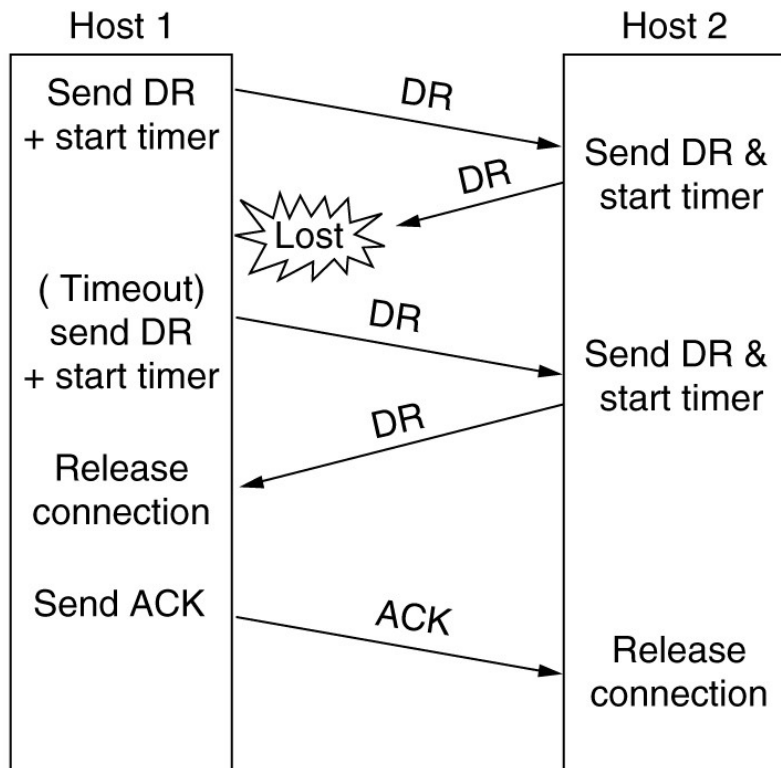
Običajen potek



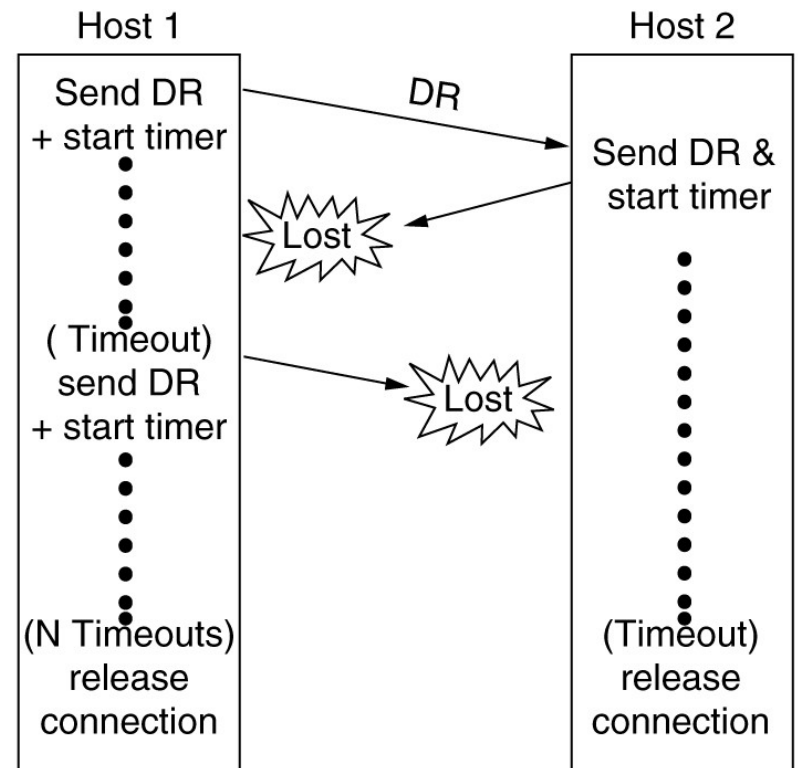
(b)

Izguba potrditve

Prekinitv v treh korakih



(c)



(d)

Prekinitev v primeru izgube zahteve za prekinitev

Simetrična prekinitev

- Ko vozlišče nima več podatkov za pošiljanje, prekine povezavo.
- Po prekinitvi še naprej lahko prejema podatke od drugega vozlišča.
- Prekinitev je popolna, ko prekineta obe vozlišči.

Nadzor pretoka

- Če je storitev mrežnega sloja nezanesljiva, mora pošiljatelj hraniti vse poslane TPDU, dokler ne prejme njihove ptditve.
- Ker pošiljatelj ne ve, ali bo prejemnik lahko prejel TPDU, ga mora hraniti (tudi v primeru zanesljivega prenosa) dokler ne prejme potrditve.
- Vprašanje prejemnikovega medpomnilnika!
 - Dinamično dodeljevanje medpomnilnika je povezano z dinamičnim določanjem širine okna pri nadzoru pretoka.

Nadzor pretoka

- Nadzor pretoka poteka na enak način kot smo ga obravnavali pri linijskem sloju!

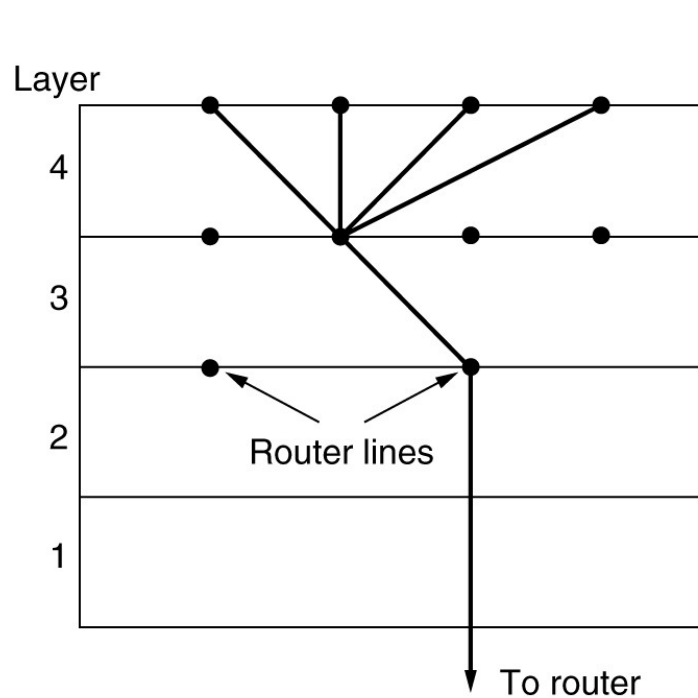
Multipleksiranje

- Multipleksiranje pogovorov se dogaja v več slojih mrežne arhitekture.
 - Fizični sloj: FDM, TDM...
 - Linijski sloj: uporaba ene povezave za več (hkratnih) pogovorov.
 - Mrežni sloj: uporaba več povezav (hkrati) za en pogovor.
 - Transportni sloj: Hkratna komunikacija več aplikacij istega vozlišča ali uporaba več komunikacij za eno vozlišče.

Multipleksiranje

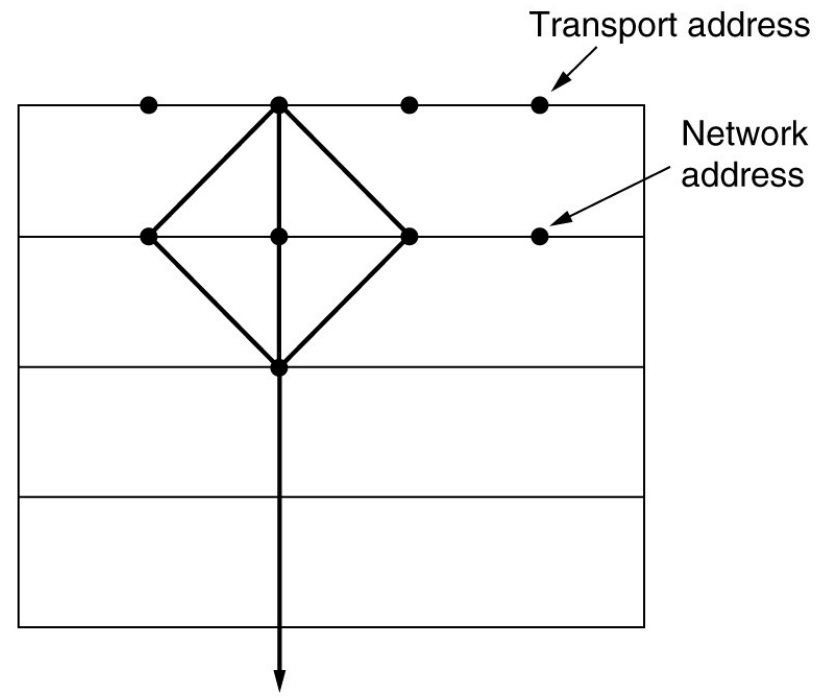
- Potreba po multipleksiranju v transportnem sloju ima torej lahko dva vzroka:
 - Če je vozlišče dostopno prek enega mrežnega naslova, morajo prek njega potekati vsi pogovori.
 - Transportni sloj omogoča multipleksiranje z uporabo več transportnih naslovov.
 - Gre za multipleksiranje 'navzgor'.
 - Če uporabnik potrebuje večjo pasovno širino, kot jo omogoča ena povezava:
 - Transportni sloj omogoča aplikaciji vzpostavitev več povezav.
 - Gre za multipleksiranje 'navzdol'.

Multiplexiranje



(a)

Multipleksiranje 'navzgor'

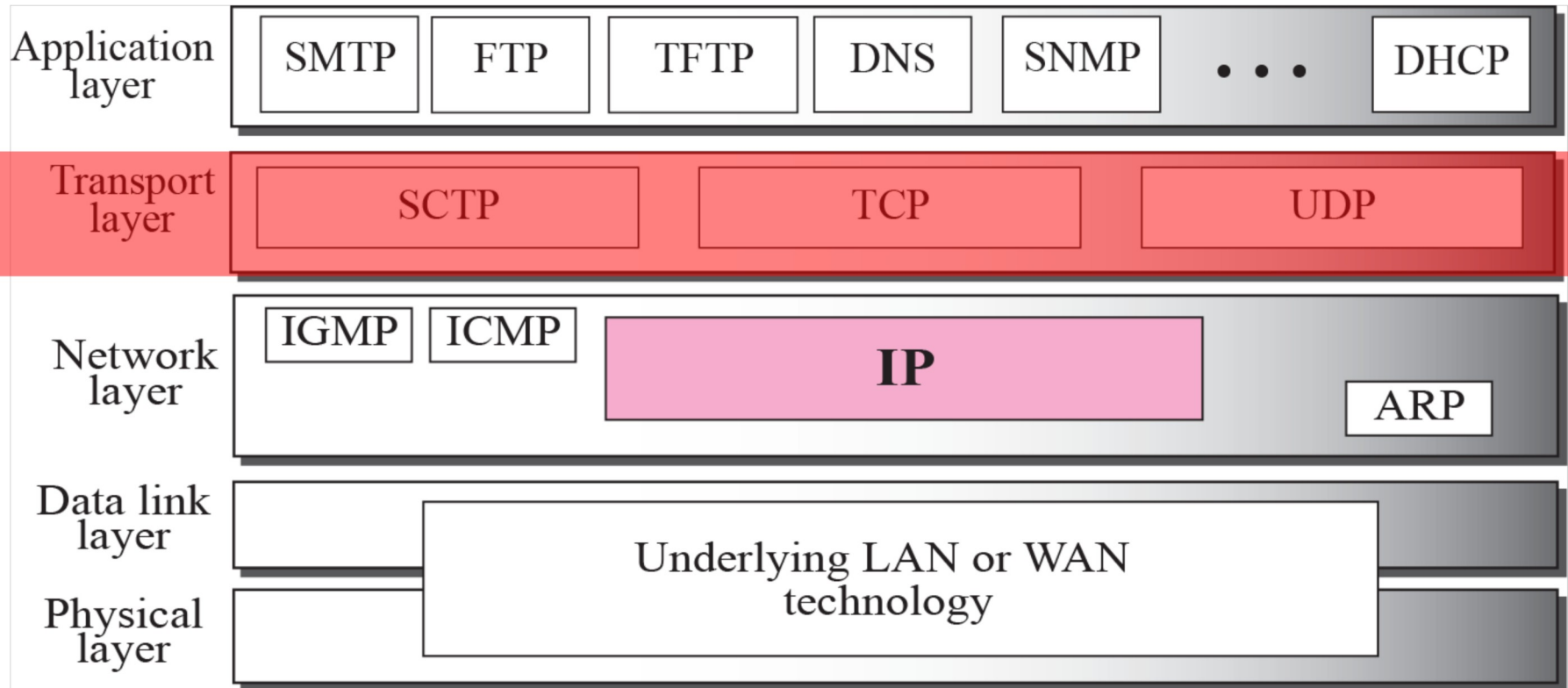


(b)

Multipleksiranje 'navzdol'

Internetni transportni protokoli

Arhitektura TCP/IP



Transportni protokoli

■ TCP

- Nudi zanesljivo povezavno storitev.

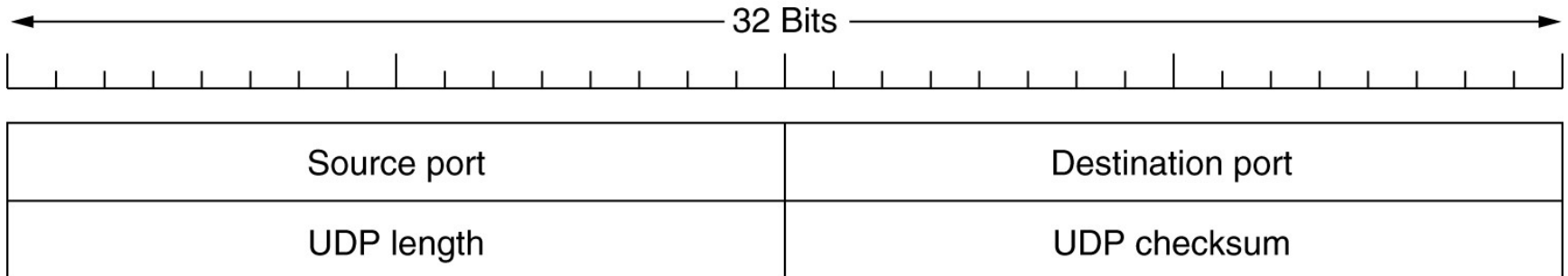
■ UDP

- Nudi nezanesljivo nepovezavno storitev.
- Brez odvečnih paketov (vzopostavljanje povezave ali potrjevanje).
- Možnost razpršenega oddajanja.

■ SCTP

- Uveden leta 2002, za prenos pretočnih vsebin.
- Konkuriranje uveljavljenim protokolom je težavno – za pretočne vsebine se še vedno večinoma uporablja UDP.

UDP



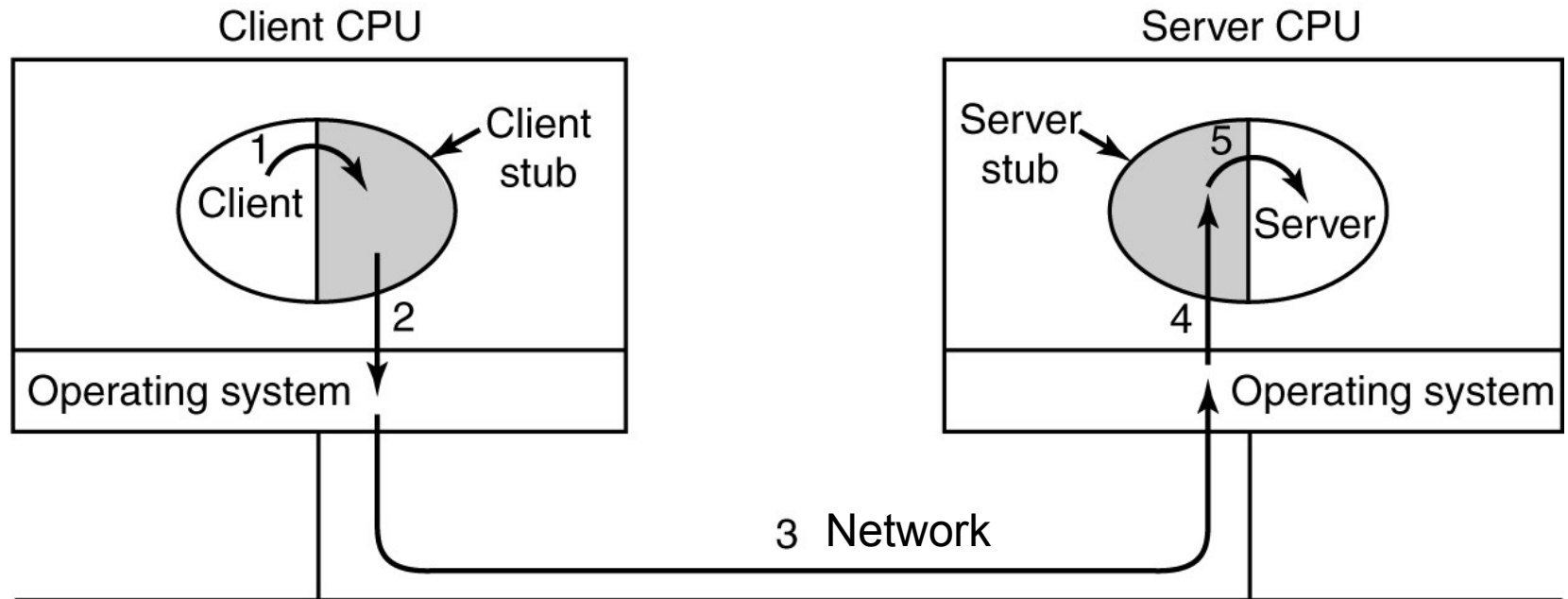
■ Glava UDP:

- Source/destination port: naslova izvornih in ponornih vrat.
- UDP length: dolžina celotnega sporočila (glava in podatki)
- UDP checksum: (opcijsko za IPv4) temelji na pseudo glavi, ki vsebuje podatke iz IP protokola.

UDP

- Enostaven protokol.
- Primeren za storitve, kjer je zagotavljanje kvalitete nepotrebno ali nezaželeno (pretočne vsebine).
- Priročen za klice oddaljenih postopkov – RPC (ang. Remote Procedure Call).
 - Oddaljen postopek se izvede na oddaljenem vozlišču.
 - Klic postopka analogen klicu funkcije/metode na lokalnem računalniku.
 - Posredovane parametrov in rezultatov po omrežju (npr. z UDP).

RPC



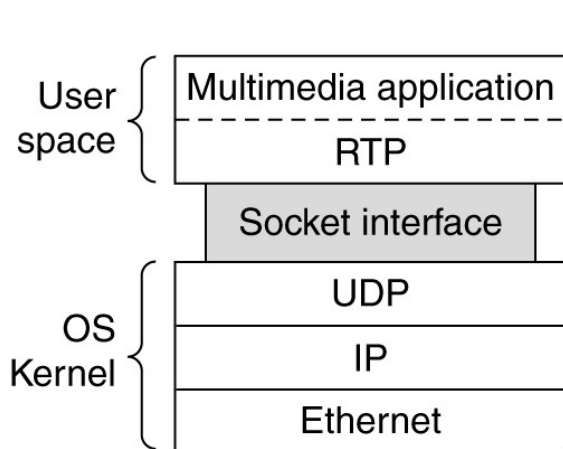
- Odjemalec in strežnik sta zgrajena z uporabo knjižnice za komunikacijo (ang. stub code):
 - Odjamalec kliče oddaljen postopek kot da se (navidezno) izvaja lokalno.
 - Strežnik izvaja postopek kot da bi ga (navidezno) klicala lokalna procedura.

RPC - omejitve

- Nezmožnost posredovanja podatkov s kazalci.
- Težavnost ugotavljanja tipov podanih parametrov.
- Nezmožnost uporabe globalnih spremenljivk.
- ...

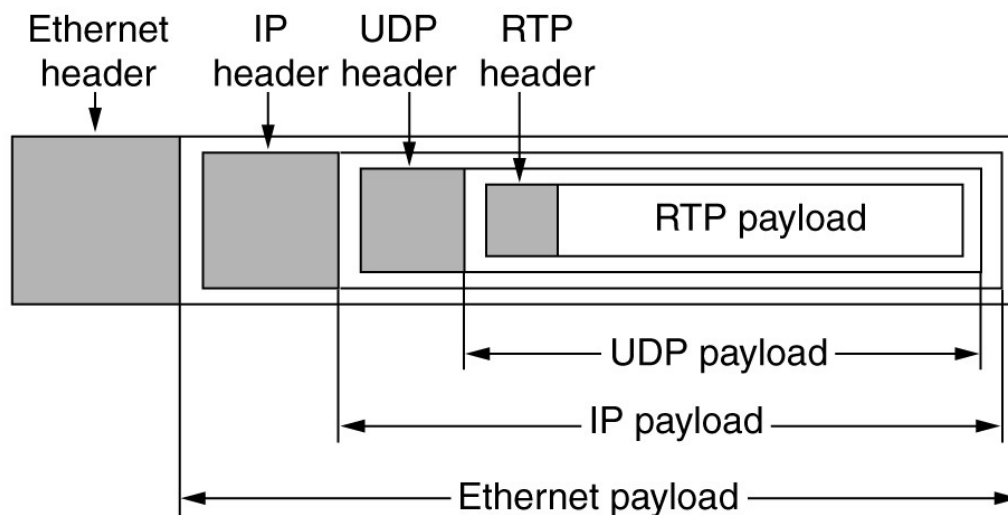
- Protokol prenosa v realnem času (ang. Real-Time Transport Protocol).
 - Splošen protokol aplikacijskega sloja za opravljanje transportnih storitev.
 - Opravlja nalogo multipleksiranja več tokov (pretočne vsebine).
 - Številčenje paketov, ob izgubi paketa je možna interpolacija (nikakor ponovni prenos).
 - Ni krmiljenja pretoka in kontrole napak.

RTP



(a)

Umestitev RTP v mrežno arhitekturo.



(b)

Vgnezdenje RTP paketov

RTCP

- RTCP (Real-Time Control Protocol) dopolnjuje RTP z možnostno nadzora:
 - Povratne informacije o pretoku, možnost vplivanja na način kodiranja...
 - Sihnronizacija...

TCP

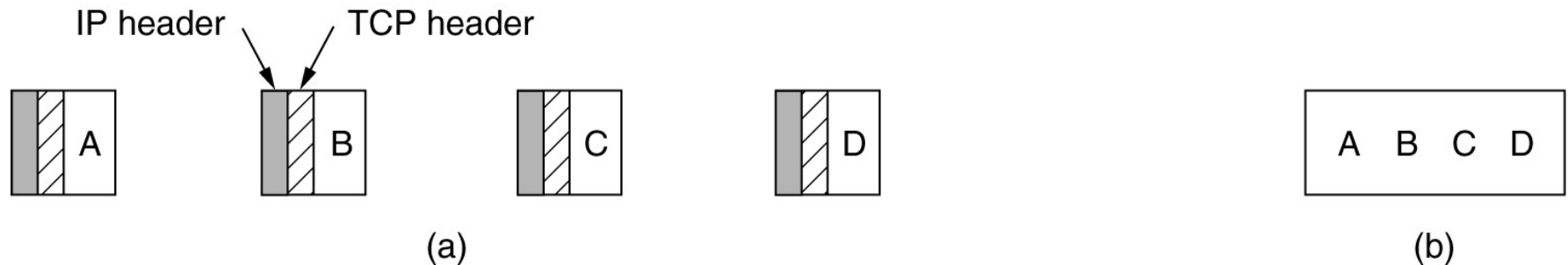
- Ang. Transmission Control Protocol.
- Glavni transportni protokol Interneta.
- Zanesljiv protokol z zaporedno dostavo podatkov.
- Za delovanje TCP protokola skrbi TCP entiteta
 - Realizirana kot del jedra sistema, knjižnica ali uporabniški proces
 - Sprejema uporabnikove tokove podatkov, jih deli na dele, ki ne presegajo 46KB (pogosto 1460B – za Ethernet).
 - Na sprejemni strani rekonstruira izvirne tokove in jih nudi uporabniku (aplikaciji).

Vtičnica - socket

- Vtičnice (sockets) so končne točke (endpoints), ki jih TCP storitev nudi pošiljatelju in prejemniku.
- Naslov vtičnice je določen z IP naslovom in transportnim naslovom, ki ga imenujemo vrata (port).
- Povezava mora biti vzpostavljena eksplicitno med **dvema** podanima končnima točkama.
 - Vsaka vtičnica je lahko uporabljena za več povezav hkrati.
 - Povezava se identificira z noslovi obeh končnih točk.

Povezave in podatkovni tokovi

- Vse povezave so dvosmerne (full duplex).
- Povezave nudijo podatkovni tok v bajtih in ne v sporočilih! Meje sporočil se ne ohranjajo!



Primer: štiri sporočila, ki se prenašajo v ločenih 512B segmentih (a) lahko prejemnik prebere v enem branju, kot eno 2048B sporočilo (b).

Povezave in podatkovni tokovi

- Ko aplikacija pošlje podatke TCP entiteti, se ti lahko
 - pošljejo takoj ali
 - začasno shranijo in pošljejo ko se nabere dovolj podatkov za napolnitev TCP okvirja.
- Aplikacija lahko zahteva takojšnjo oddajo podatkov (PUSH flag).
 - Tudi podatki poslani s PUSH zastavico se na strani prejemnika lahko združijo v enotno sporočilo!
- Urgentni podatki (kontrolne informacije pomembne za delovanje programov) se lahko pošljejo prioriteto (URGENT flag)
 - TCP entiteta preneha akumulirati podatke in jih takoj pošlje prejemniku.
 - Na strani prejemnika urgentni podatki prekinejo delovanje aplikacije. Ta lahko prebere/obdela prejete podatke. Konec urgentnih podatkov je označen.

TCP protokol

- Vsak bajt prenešen preko TCP ima svojo 32 bitno zaporedno številko (sequence number).
 - V sodobnih omrežjih se števec lahko obrne zelo hitro!
 - Pri 10Gb/s dejanskem prenosu podatkov se to zgodi v 3,4s
- TCP segment sestoji iz
 - glave dolžine 20B (+ opcije) ter
 - podatki dolžine 0 ali več.
- Velikost podatkov v segmentu je omejena z:
 - dopustno velikostjo podatkov v IP paketu (65515 B),
 - MTU (maximum transfer unit), v praksi 1500 za Ethernet.

TCP glava

| Bit offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
|------------|------------------------------|---|---|---|----------|---|---|---|-------------|-------------|-------------|-------------|-------------|-------------|------------------|-------------|------------------|----|----|----|----|----|----|----|---------|----|----|----|----|----|----|----|
| 0 | Source port | | | | | | | | | | | | | | | | Destination port | | | | | | | | | | | | | | | |
| 32 | Sequence number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 64 | Acknowledgment number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 96 | Data offset | | | | Reserved | | | | C W R | E C E | U R G | A C K | P S H | R S T | S S Y N | F I N | Window Size | | | | | | | | | | | | | | | |
| 128 | Checksum | | | | | | | | | | | | | | | | Urgent pointer | | | | | | | | | | | | | | | |
| 160 | Options (if Data Offset > 5) | | | | | | | | | | | | | | | | | | | | | | | | padding | | | | | | | |
| ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

- Source/Destination port: številka izvornih in ponornih vrat.
 - Številke vrat pomembnih aplikacij ureja www.iana.org (do 1024)
- Sequence number: zaporedna številka (prvega naslednjega bajta)
- Acknowledgment number: številka potrditve - zaporedna številka prvega naslednjega pričakovanega bajta.
- Data offset: dolžina glave v mnogokratnikih 32b.
- Reserved: ni v uporabi in mora biti 0.

TCP glava

| Bit offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
|------------|------------------------------|---|---|---|----------|---|---|---|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------------|----|----|----|----|----|----|----|----|----|---------|----|----|----|----|----|
| 0 | Source port | | | | | | | | | | | | | | | | Destination port | | | | | | | | | | | | | | | |
| 32 | Sequence number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 64 | Acknowledgment number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 96 | Data offset | | | | Reserved | | | | C W R | E C E | U R G | A C K | P S H | R S T | S Y N | F I N | Window Size | | | | | | | | | | | | | | | |
| 128 | Checksum | | | | | | | | | | | | | | | | Urgent pointer | | | | | | | | | | | | | | | |
| 160 | Options (if Data Offset > 5) | | | | | | | | | | | | | | | | | | | | | | | | | | padding | | | | | |
| ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

■ Zastavice:

- ☐ CWR (Congestion Window Reduced): mehanizem nadzora preobremenitev.
- ☐ ECE (ECN-Echo): končno vozlišče omogoča ECN nadzor preobremenitev.
- ☐ URG (Urgent): aktiven je kazalec urgentnih podatkov.
- ☐ ACK (Acknowledgment): veljavna številka potrditve.
- ☐ PSH (Push): podatki poslani z zahtevo po takojšnji oddaji (PUSH zastavica).
- ☐ RST (Reset): zavrnitev neveljavnega segmenta.
- ☐ SYN (Synchronize): Sinhronizacija zaporednih številok pti vzpostavitvi povezave.
- ☐ FIN (Finish): Končanje prenosa – prekinitev povezave.

TCP glava

| Bit offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
|------------|------------------------------|---|---|---|----------|---|---|---|-------------|-------------|-------------|-------------|-------------|-------------|------------------|-------------|------------------|----|----|----|----|----|----|----|----|----|---------|----|----|----|----|----|
| 0 | Source port | | | | | | | | | | | | | | | | Destination port | | | | | | | | | | | | | | | |
| 32 | Sequence number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 64 | Acknowledgment number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 96 | Data offset | | | | Reserved | | | | C W R | E R E | U R G | A C K | P S H | R S T | S S Y N | F I N | Window Size | | | | | | | | | | | | | | | |
| 128 | Checksum | | | | | | | | | | | | | | | | Urgent pointer | | | | | | | | | | | | | | | |
| 160 | Options (if Data Offset > 5) | | | | | | | | | | | | | | | | | | | | | | | | | | padding | | | | | |
| ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

- Window size: dinamična velikost okna – koliko bajtov sme biti poslano od zadnjega potrjenega bajta.
 - 0 – sprejemnik zaseden in ni pripravljen sprejemati podatkov! Pošiljanje se lahko nadaljuje po ponovni potrditvi zadnjega segmenta z navedeno velikostjo okna >0.
- Checksum: preverjanje veljavnosti glave, podatkov in pseudoglave s podatki IP paketa.
 - Kršitev pravil ločenosti omrežnih slojev!

TCP glava

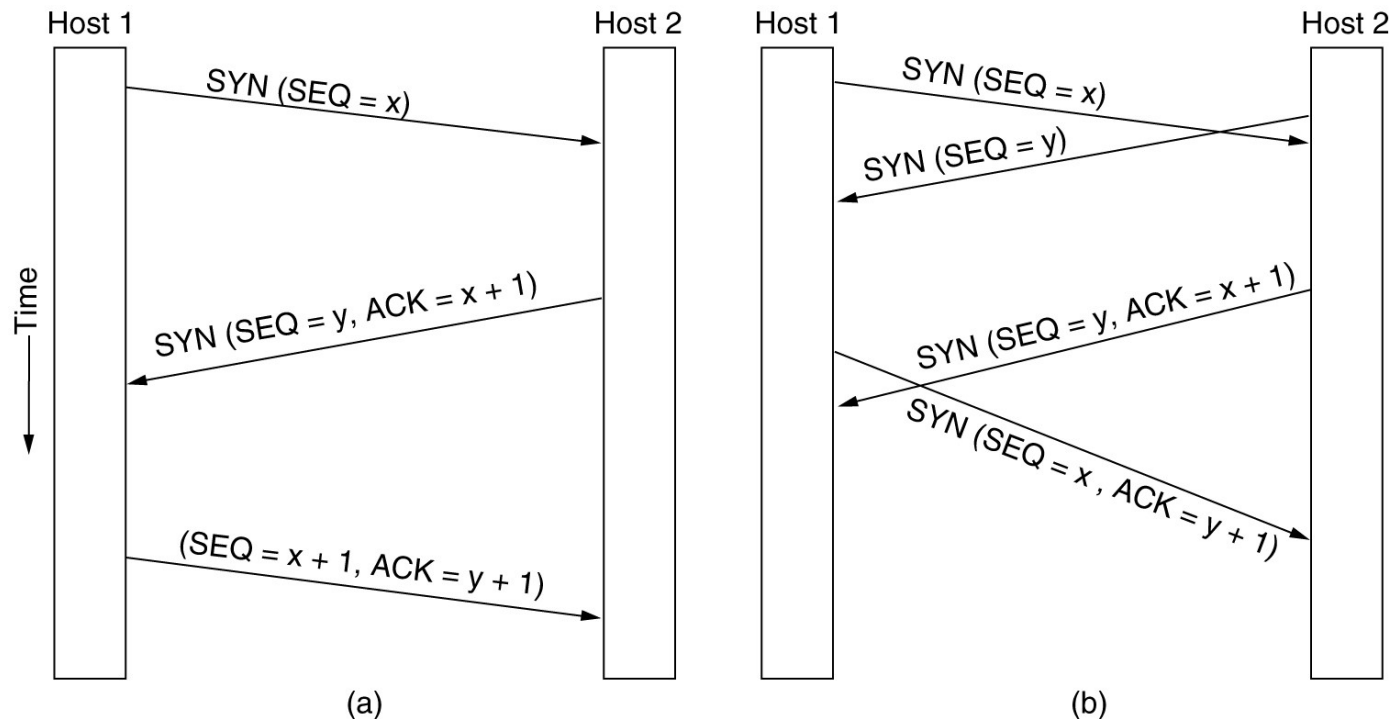
| Bit offset | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
|------------|------------------------------|---|---|---|----------|---|---|---|-------------|-------------|-------------|-------------|-------------|-------------|------------------|-------------|------------------|----|----|----|----|----|----|----|----|----|---------|----|----|----|----|----|
| 0 | Source port | | | | | | | | | | | | | | | | Destination port | | | | | | | | | | | | | | | |
| 32 | Sequence number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 64 | Acknowledgment number | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 96 | Data offset | | | | Reserved | | | | C W R | E R E | U R G | A C K | P S H | R S T | S S Y N | F I N | Window Size | | | | | | | | | | | | | | | |
| 128 | Checksum | | | | | | | | | | | | | | | | Urgent pointer | | | | | | | | | | | | | | | |
| 160 | Options (if Data Offset > 5) | | | | | | | | | | | | | | | | | | | | | | | | | | padding | | | | | |
| ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

- Urgent pointer: kazalec na zadnji urgentni bajt podatkov.
- Options: opsijske razširitve, npr.:
 - sporočanje največje velikosti podatkovnega polja,
 - skaliranje okna (window size) v primeru potrebe – pri višjih hitrostih prenosa in večjih zakasnitvah.
 - Implementacija SRP (selective repeat) namesto običajnega GBN (go-back-n).
- Padding: dopolnitev glave na velikost $n \times 32$ bitov.

TCP vzpostavljajanje povezave

- Vzpostavitev povezave poteka s trikratnim rokovanjem (three-way handshake):
 - Strežnik aktivno čaka na zahtevo odjemalca (LISTEN, ACCEPT).
 - Odjemalec zahteva povezavo (CONNECT)
 - Navede naslov končne točke strežnika
 - Določi max velikost prejetega segmenta
 - Odpošlje TCP segment s SYN=1 in ACK=0.
 - Strežnik odgovori s TCP segmentom:
 - Z RST=1 za zavrnitev zahteve, ali
 - SYN=1 za sprejetje zahteve.
 - Odjemalec potrdi povezavo s potrditvijo ACK v prvem podatkovnem segmentu.

TCP vzpostavljajanje povezave

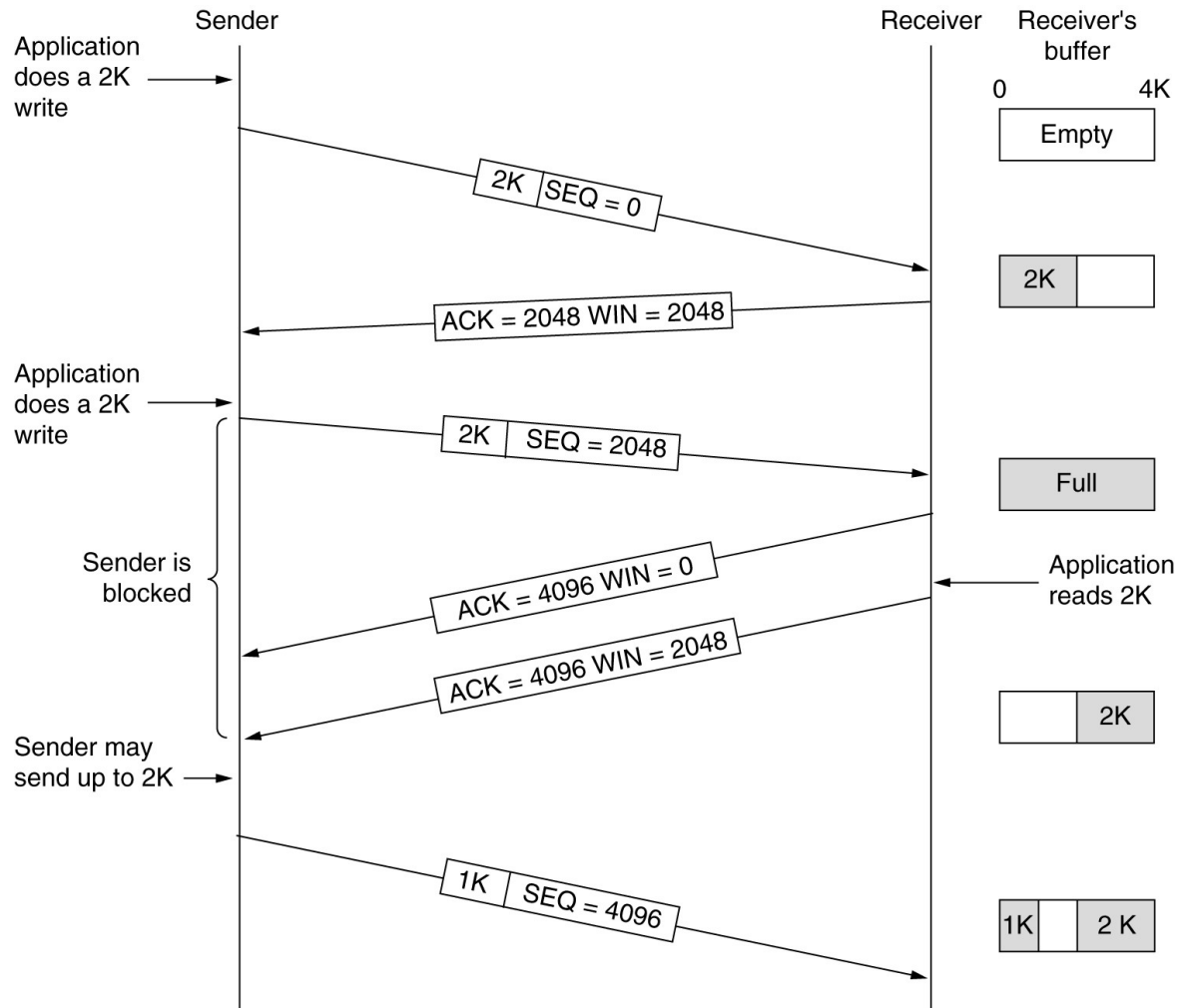


Vzpostavljajanje TCP povezave (a) običajna uspešna vzpostavitev in (b) hkratno vzpostavljajanje povezave z obeh končnih točk.

TCP prekinitev povezave

- TCP uporablja simetrično prekinitev povezave.
 - Dvosmerno (full duplex) povezavo si lahko predstavljamo kot dve enosmerni povezavi.
 - Za prekinitev pošiljanja v vsako od smeri je potrebno povezavo zapreti ločeno (FIN=1).
 - V vsaki smeri je prekinitveni segment potrjen (ACK).
 - Če se potrditev izgubi, se povezava vseeno prekine po dveh življenskih ciklih paketov (v izogib problema dveh armad).

TCP upravljanje pretoka



Mehanizem upravljanja pretoka TCP.

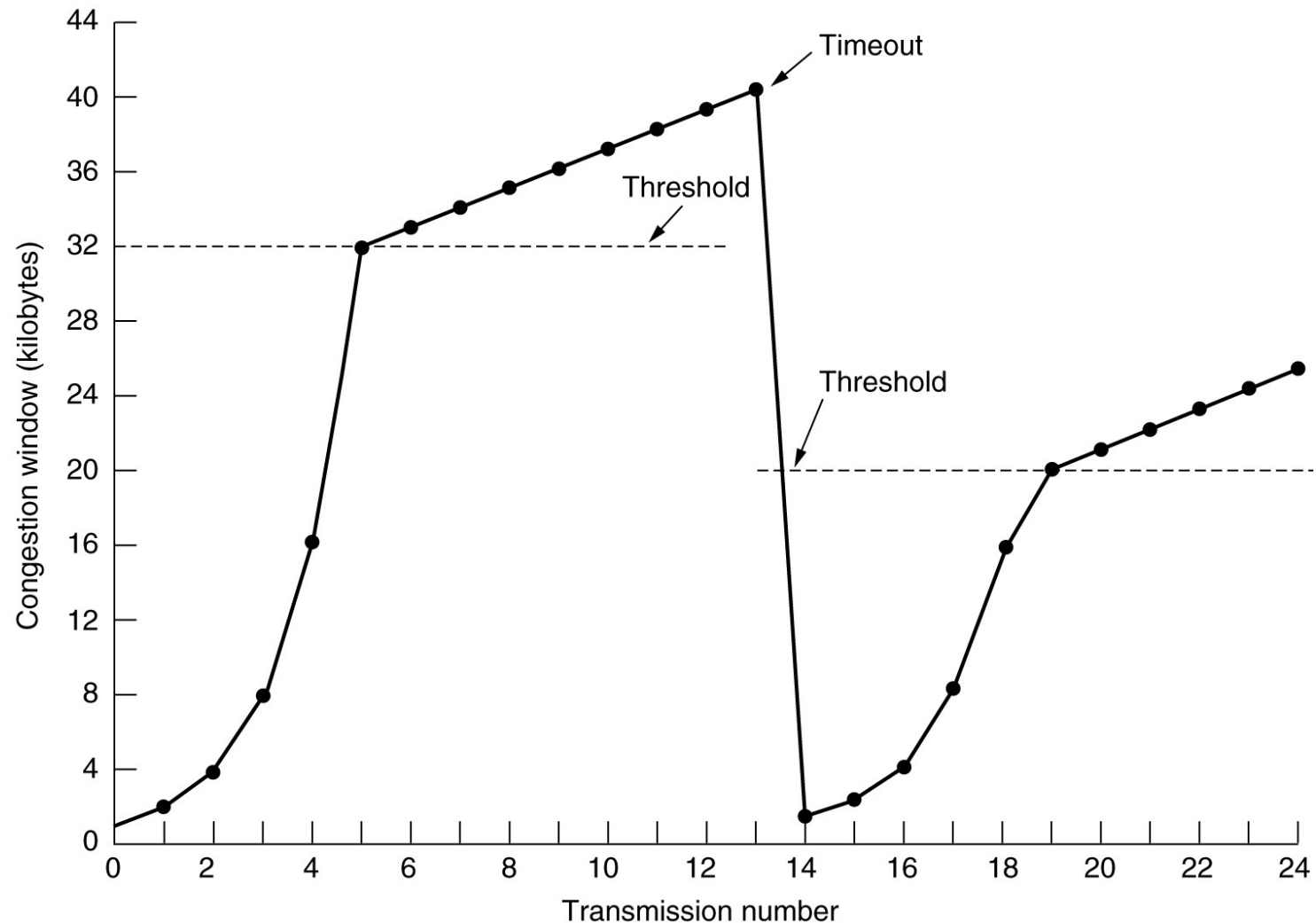
TCP upravljanje preobremenitev

- Za upravljanje preobremenitev se uporablja mehanizem spremenljive širine okna.
 - Z zmanjšanjem širine okna se zmanjša pretok.
 - Vsak oddajnik vzdrži dve širini oken in vedno uporabi manjšega od obeh:
 - okno določeno od sprejemnika
 - preobremenitveno okno (congestion window).
- Predpostavlja se, da je pretek časovnika vedno posledica preobremenitev... (je to res?)

TCP preobremenitveno okno

- Mehanizem 'slow start':
 - Ob vzpostavitvi povezave oddajnik za preobremenitveno okno vzame vrednost enega največjega dovoljenega segmenta.
 - Če je segment potrjen (pred iztekom časovnika), podvoji velikost preobremenitvenega okna.
 - Če pride do izteka časovnika, se velikost okna ustali na prejšnji vrednosti (slow start).
- Dodatno se velikost okna ureja z mehanizmom mejne vrednosti.
 - Prvo mejno vrednost se določi z mehanizmom 'slow start'.
 - Od tu naprej se velikost okna povečuje linearno, po eno največjo velikost segmenta.
 - Če (ko) pride do izteka časovnika, se mejna vrednost postavi na polovico trenutne velikosti okna in se ponovno začne z mehanizmom 'slow start'.

TCP preobremenitveno okno



Postopek določanja širine preobremenitvenega okna.

TCP upravljanje časovnika

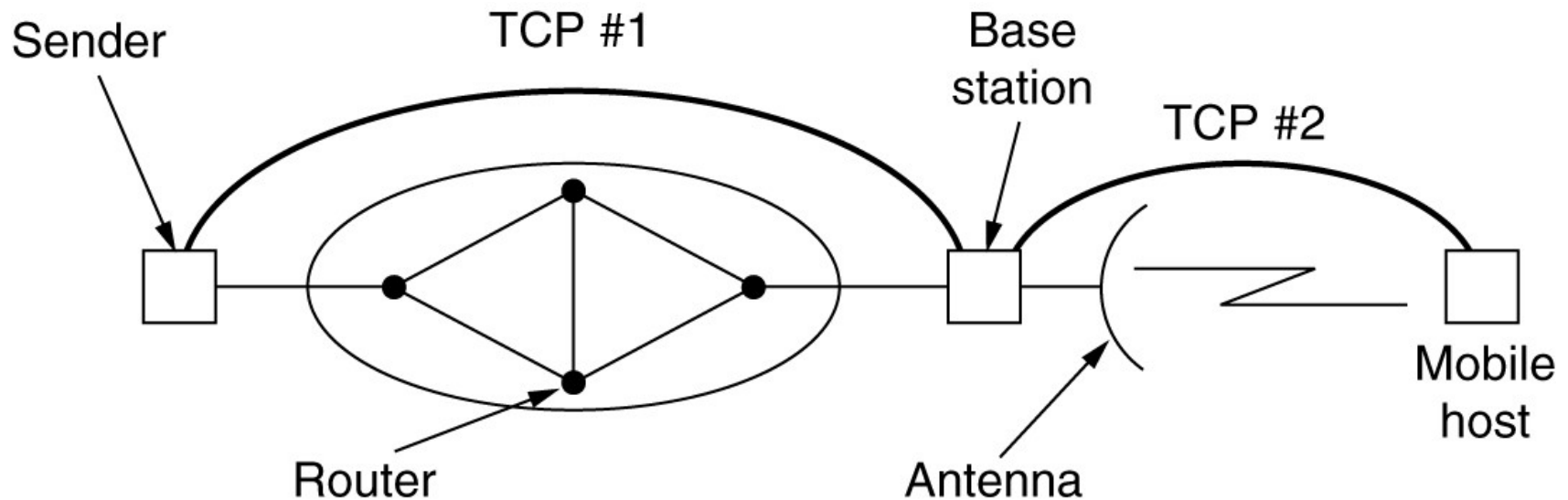
- TCP uporablja več časovnikov, najpomembnejši pa je “retransmission timer”.
 - Časovnik starta ob pošiljanju TCP segmenta.
 - Če TCP segment ni potrjen do izteka časovnika, se segment pošlje ponovno.
- Določitev časovnega intervala je težavna in se določa dinamično.
 - Na osnovi ocene časa za pot podatkov v obe smeri (round-trip time - RTT).
 - Običajno se za časovnik uporabi interval $2 \times \text{RTT}$, obstajajo pa tudi kompleksnejši mehanizmi.

TCP in brezžična omrežja

- V primeru brezžičnih omrežij ne drži predpostavka, da je iztek časovnika (le) posledica preobremenitev.
 - V tem primeru je upravljanje preobremenitev oteženo.
 - Ena od rešitev je razdelitev povezave na dva segmenta:
 - segment od postaje do brezžične dostopne točke,
 - segment od brezžične dostopne točke dalje.

Rešitev izboljša upravljanje preobremenitev, vendar krši pravila transportnega sloja – sloj nič več ne povezuje končnih vozlišč!

TCP in brezžična omrežja



Delitev povezave na dva segmenta
(možna uporaba za TCP in UDP)

SCTP

- Stream Control Transmission Protocol.
- Uveden leta 2002.
- Namenjen prenosu pretočnih vsebin (to vlogo naj bi prevzel od UDP).
- Je sporočilno orientiran (kot UDP)
- Omogoča zanesljiv sekvenčni transport, z nadzorom preobremenitev (kot TCP).
- Mnogi domači mrežni usmerjevalniki tega protokola še ne podpirajo!

SCTP - okvir

| Bits | 0–7 | 8–15 | 16–23 | 24–31 |
|------|------------------|---------------|------------------|-------|
| +0 | Source port | | Destination port | |
| 32 | Verification tag | | | |
| 64 | Checksum | | | |
| 96 | Chunk 1 type | Chunk 1 flags | Chunk 1 length | |
| 128 | Chunk 1 data | | | |
| ... | ... | | | |
| ... | Chunk N type | Chunk N flags | Chunk N length | |
| ... | Chunk N data | | | |

Mrežno programiranje

Berkely sockets

- Za uporabo TCP in UDP protokola po IP omrežju je bila razvit aplikacijski programski vmesnik (API) znan kot 'Berkely sockets.'
 - Izdan leta 1983, programski jezik C, operacijski sistem Unix.
 - Postal je defacto standard.
 - Na njem temelji vrsta drugih knjižnic za različne operacijske sisteme in programske jezike.