

**Transporto lygis**

# Turinys

- Buferizavimas;
- Multipleksavimas;
- Atstatymas po sutrikimų;
- Transporto lygio protokolai internete:
  - UDP;
  - TCP;
- Eksploatacinių savybių užtikrinimas

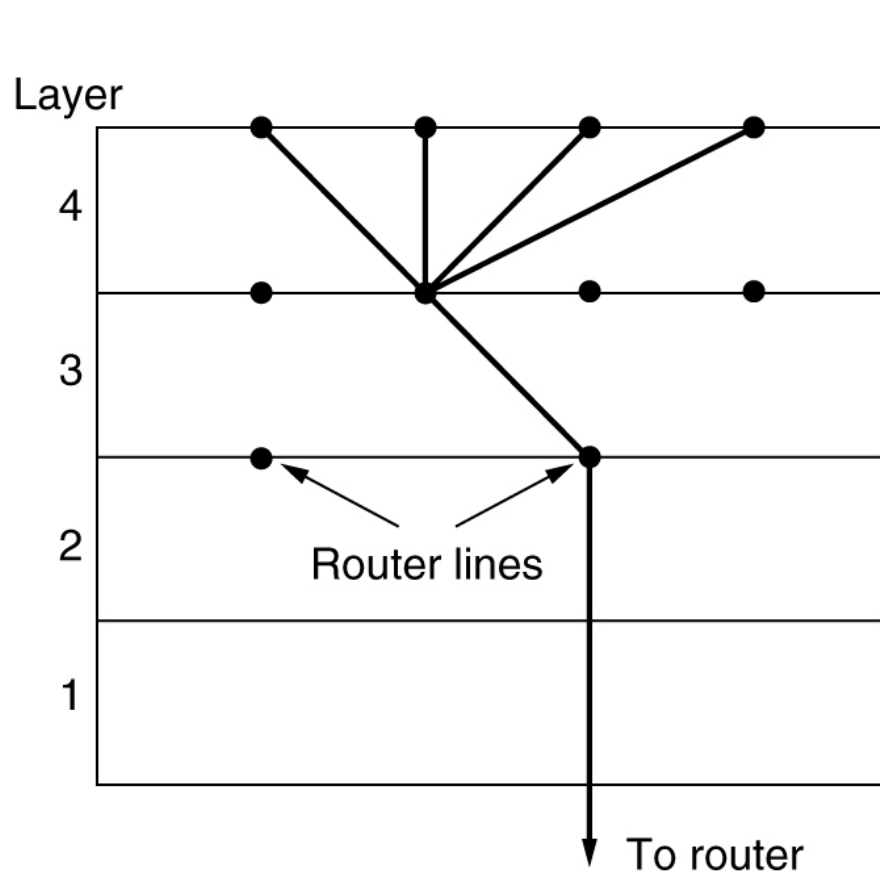
# Buferizavimas

- Paketų buferizavimas būtinas tiek siuntėjo tiek gavėjo pusėje
- 
- Buferių tipai:
  - Sujungti fiksuoto dydžio buferiai
  - Sujungti kintančio dydžio buferiai
  - Vienas didelis buferis susijungimui

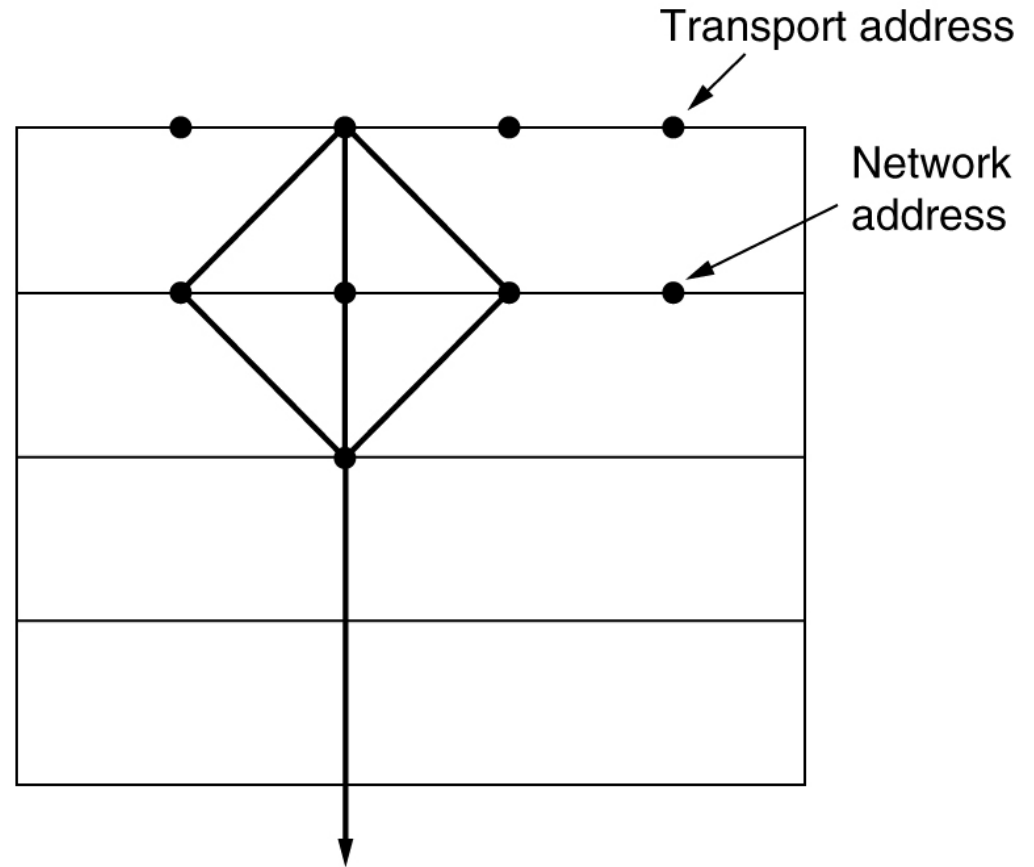
# Multipleksavimas

- Multipleksavimas nusileidimu
  - Pavyzdžiui, kainos sumažinimo tikslais, galima atvaizduoti kelis transporto sujungimus į vieną
- Multipleksavimas iš apačios į viršų
  - Kai kuriais atvejais pralaidumo padidrinimo tikslais galima vieną transporto sujungimą atvaizduoti į kelis tinklo sujungimus ir kiekvienam tinklo sujungimui turėti atskirą slankiojantį langą (angl. sliding window). Tuomet apdorojus vieną buferį galima eiti prie kito

# Multiplexing



(a)



(b)

# Atstatymas po sutrikimų

- Problema – atstatyti mašinos veikimą, įskaitant ir transporto lygį
  - Tarkim, serveris sutriko ir bando atstatyti funkcionavimą. Visų pirma, jis turi sužinoti iš kliento, koks buvo paskutinis nepatvirtintas TPDU ir paprašyti pakartotinai atsiųsti jį. Tuo tarpu klientas turi būti vienoje iš dviejų būsenų:
    - $S_1$  – nepatvirtintas TPDU
    - $S_0$  – visi TPDU buvo patvirtinti
- Ši problema negali būti išspręsta tik transporto lygio priemonėmis
  - Po TPDU išsaugojimo reikia informuoti apie tai taikomąją programą ir tik tada išsiųsti patvirtinimą. Atsistatant po sutrikimo reikia apklausti ne tik esybę transporto lygyje bet ir taikomąją programą.

# Atstatymas po sutrikimų

		Strategy used by receiving host					
		First ACK, then write			First write, then ACK		
Strategy used by sending host		AC(W)	AWC	C(AW)	C(WA)	W AC	WC(A)
	Always retransmit	OK	DUP	OK	OK	DUP	DUP
	Never retransmit	LOST	OK	LOST	LOST	OK	OK
	Retransmit in S0	OK	DUP	LOST	LOST	DUP	OK
	Retransmit in S1	LOST	OK	OK	OK	OK	DUP

OK = Protocol functions correctly

DUP = Protocol generates a duplicate message

LOST = Protocol loses a message

# **Transporto lygio protokolai Internete**



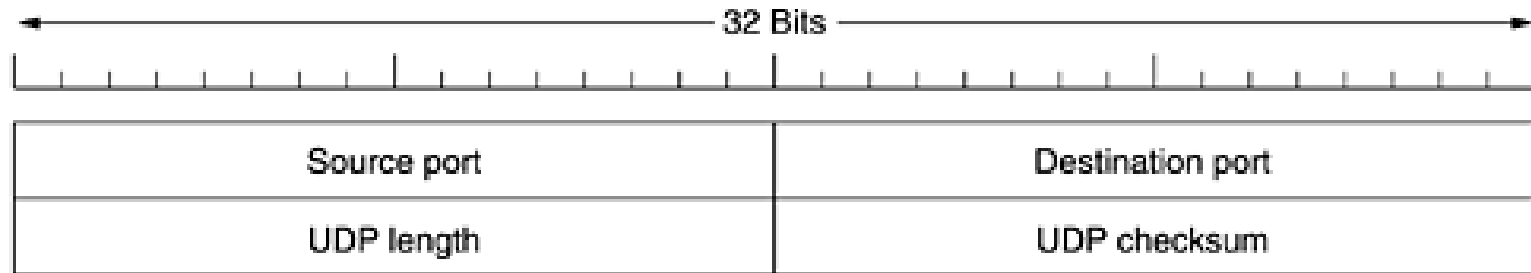
# Transporto protokolai internete: TCP ir UDP

- Internete yra du pagrindiniai transporto protokolai: TCP – orientuotas į sujungimus ir UDP – neorientuotas į sujungimus.
- UDP – yra beveik tas pats IP paketas su papildomais laukais antraštelėje. Todėl didesnis dėmesys bus skiriamas TCP protokolui.

# UDP protokolas

- Naudotojo datagramų protokolas – UDP (angl. User Datagram Protocol)
- Aprašomas RFC768

# UDP Antrašte



# TCP Protokolas

- Perdavimo valdymo protokolas – TCP (angl. Transmission Control Protocol)
- Aprašomas RFC793, vėliau ištaisytas RFC 1122.  
Papildymai aprašomi RFC 1323

# TCP Servisas

- TCP Servisas prieinamas per soketą
  - Soketas susideda iš host'o IP adreso ir 16 bitų lokalaus numerio hoste – portu. Portas yra TSAP TCP protokolui.
  - Kiekvienas sujungimas identifikuojamas soketu, tarp kurių yra tas sujungimas, pora.
  - Portai iki 1023 rezervuoti standartiniams servisams.
  - Visi TCP sujungimai dupleksiniai

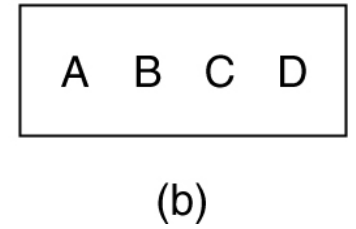
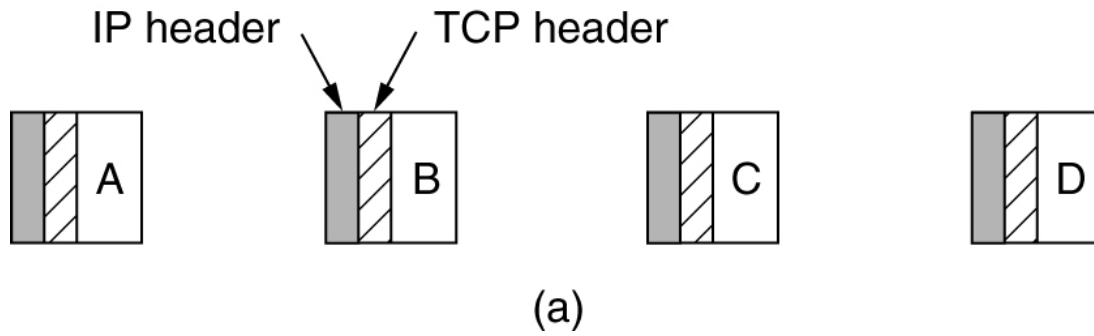
# TCP Servisas

- TCP Sujungimas palaiko “1-to-1” būdą.  
Nėra sujungimų “1-to-N”.
- TCP Pateikia baitų srautą, o ne pranešimų srautą.
- Požymis PUSH TCP paketo antraštėje – paketas turi būti perduodamas nedelsiant
- Požymis URGENT – visi šio sujungimo sukaupti duomenys turi būti perduoti iš karto. Kai skubūs duomenys pasiekia gavėją, gavėjas turi būti pertrauktas ir priimti duomenis.

# TCP Protokolas

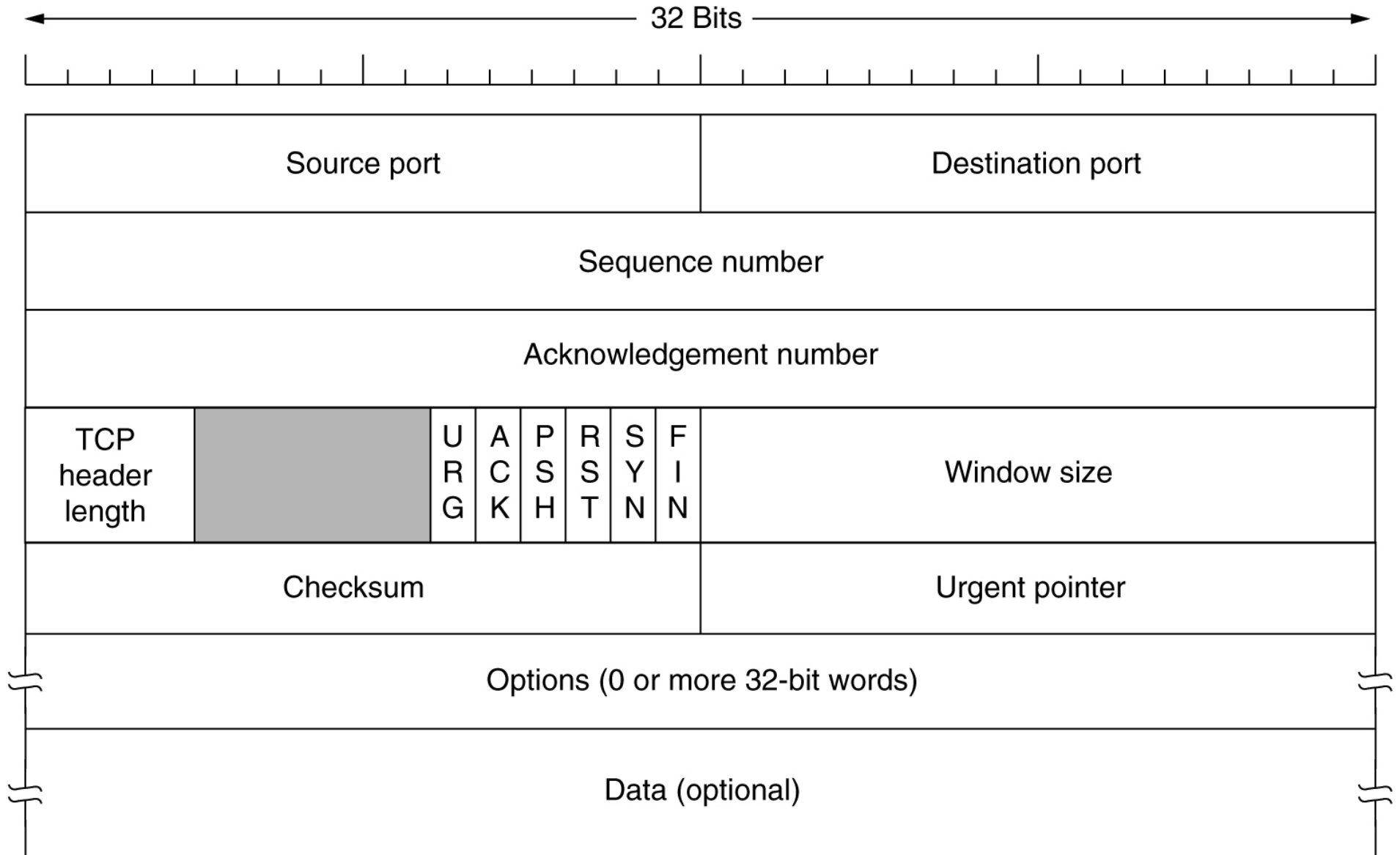
- Protokolas leidžia spręsti tokius uždavinius
  - Atstatyti segmentų tvarką
  - Šalinti segmentų dublikatus
  - Apibrėžti tinkamą užlaikymą segmento gavimo patvirtinimo time-out.
  - Patikimai užmegzti ir nutraukti sujungimą
  - Valdyti srautą
  - Valdyti perkrovimą

# Segmentavimas





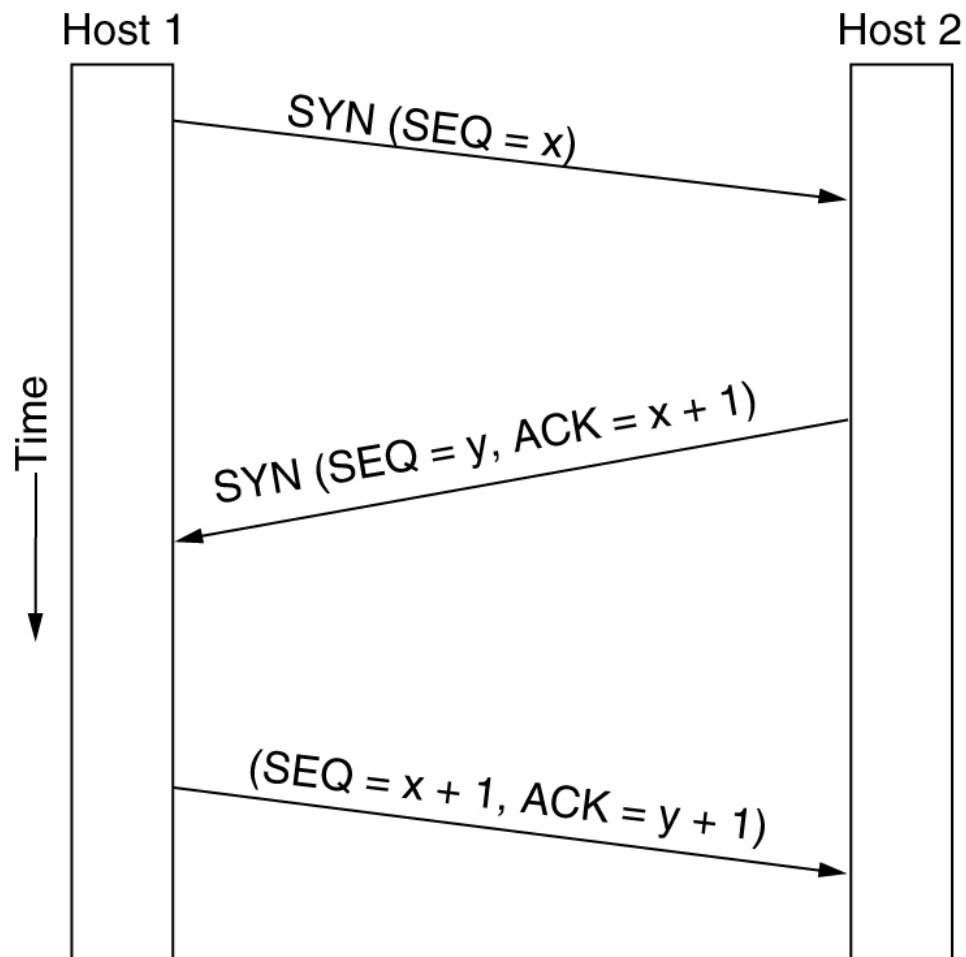
# TCP Antrašte



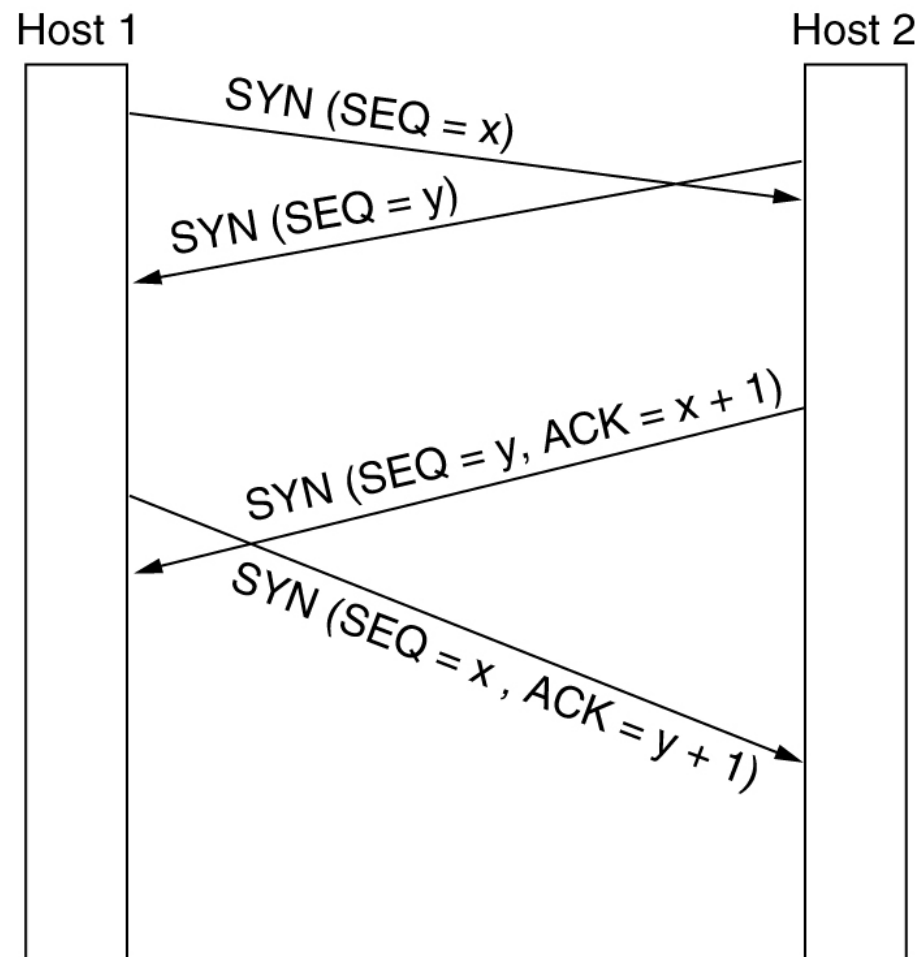
# Sujungimų valdymas TCP protokole

- TCP sujungimo užmezgimas vyksta pagal trigubo rankų paspaudimo (3-way handshake) protokolą
  - Požymiai SYN ir ACK segmento galvoje naudojami pažymėti pranešimams CONNECTION REQUEST ir CONNECTION ACCEPTED. Požymis RST naudojamas pažymėti REJECT.

# TCP Ryšio užmezgimas



(a)



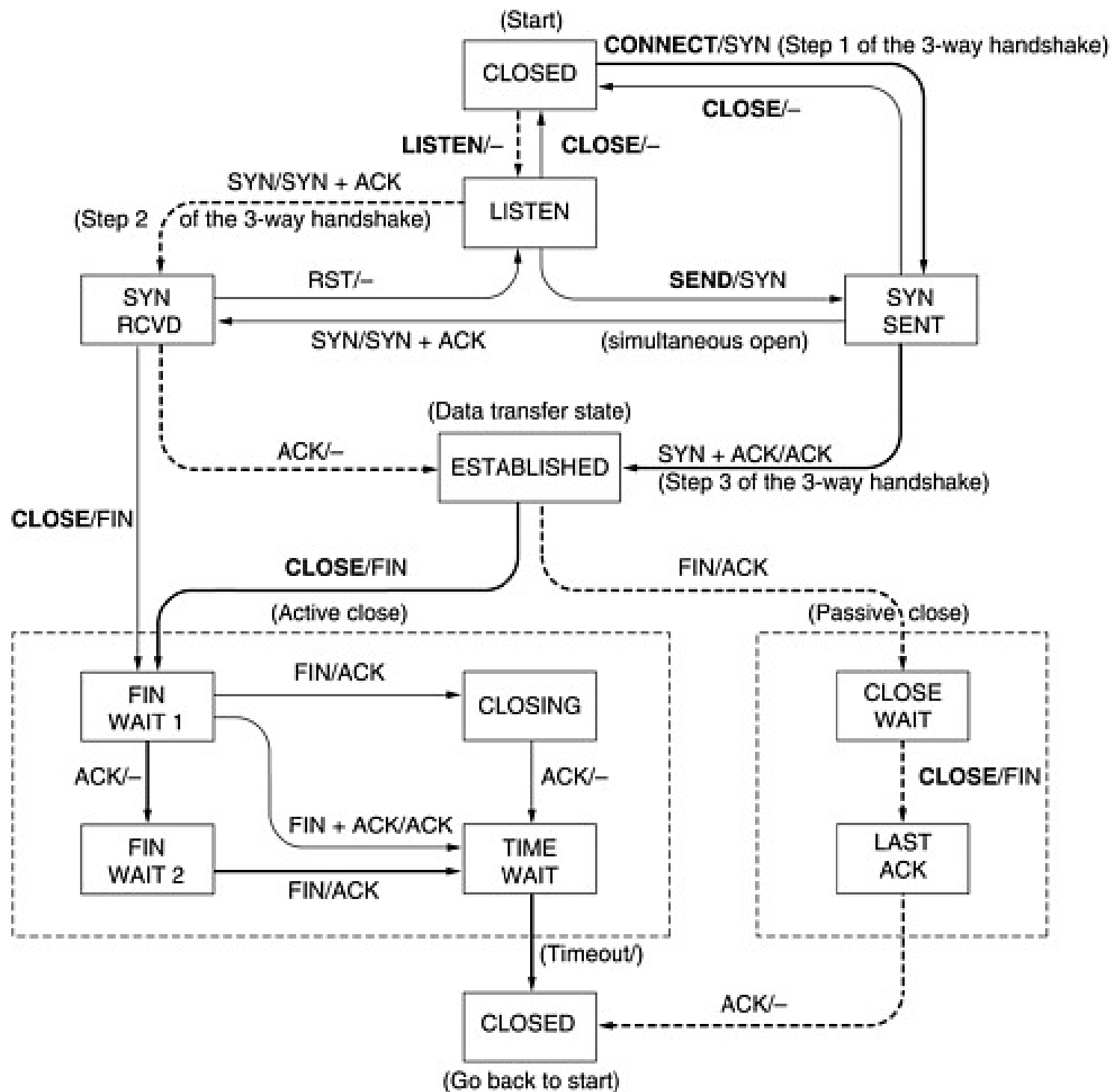
(b)

# Ryšio nutraukimas

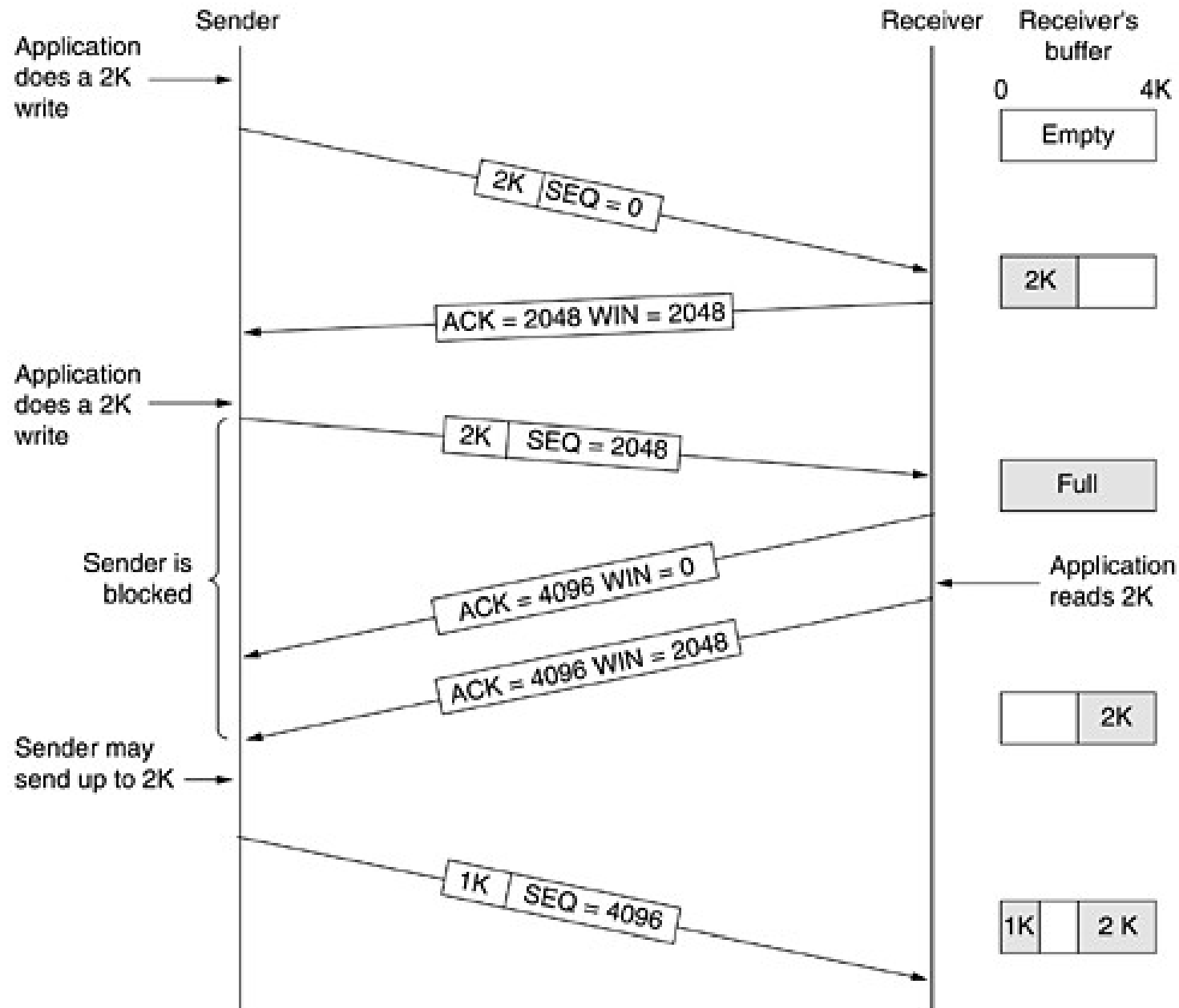
- Ryšiui nutraukti reikia išsiųsti FIN ir ACK segmentus į abi puses;
- Tam kad išvengti dviejų armijų problemos taimeriai yra panaudoti;
- Jei FIN paketas neatkeliauja per 2 maksimalius paketo gyvavimo laikotarpius ryšys yra nutraukiamas.

# TCP Susijungimų būsenos

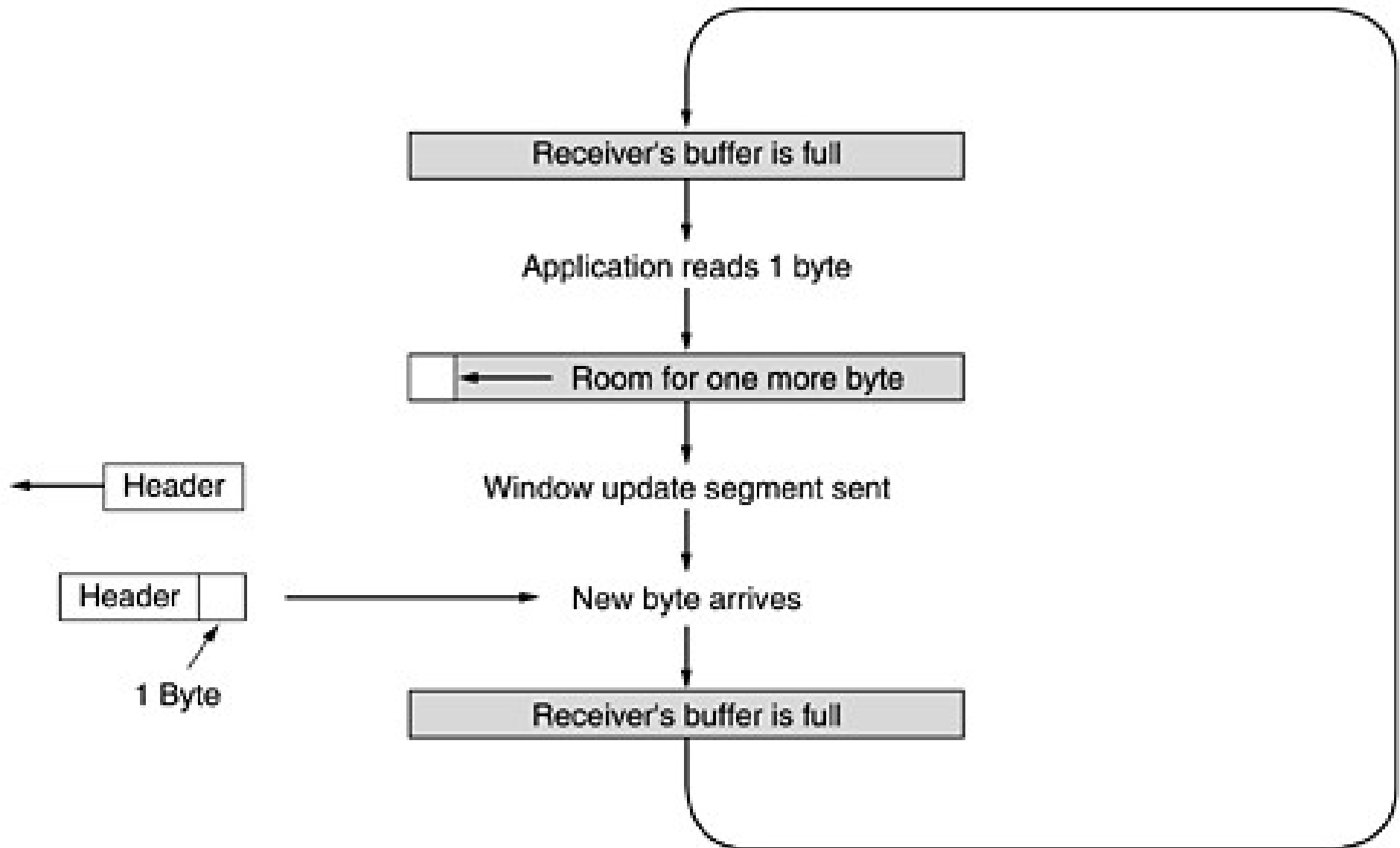
Būsena	Aprašymas
CLOSED	Nėra įvykdytas joks susijungimas
LISTEN	Serveris laukia prisijungimų
SYN RCVD	Gauta prisijungimo užklausa, laukiamas patvirtinimas
SYN SENT	Taikomoji programa kuria susijungimą
ESTABLISHED	Siuntimo būsena
FIN WAIT 1	Taikomoji programa nutraukia susijungimą
FIN WAIT 2	Kita pusė patvirtina susijungimo nutraukimą
TIMED WAIT	Laukiama kol visi paketai bus sunaikinti
CLOSING	Abi pusės bando uždaryti susijungimą
CLOSE WAIT	Kita pusė pradėjo ryšio nutraukimą
LAST ACK	Laukiama kol visi paketai bus sunaikinti



# TCP Siuntimo politika



# TCP Siuntimo politika





# **Eksploatacinių savybių užtikrinimas**

- Taisyklės:
  1. Procesoriaus sparta svarbiau nei tinklo;
  2. Sumažintas paketų skaičius – sumažintas programinės įrangos darbas;
  3. Sumažinti konteksto pasikeitimų skaičių;
  4. Sumažinti kopijavimų skaičių;
  5. Galima nusipirkti didesnę pralaidumą, bet ne mažesnę užlaikymą;
  6. Išvengti susigrūdimų yra geriau nei bandyti nuo jų atsistatyti;

# **Eksploatacinių savybių užtikrinimas**

7. Vengti time out'ų;

**Klausimai?**