

Glava 5: Mrežni nivo

5.1 Uvod

5.2 IP (*Internet Protocol*)

- Format datagrama
- IP adresiranje

5.3 Rutiranje

- *Link state*
- *Distance Vector*
- Hijerarhijsko rutiranje
- Protokoli rutiranja

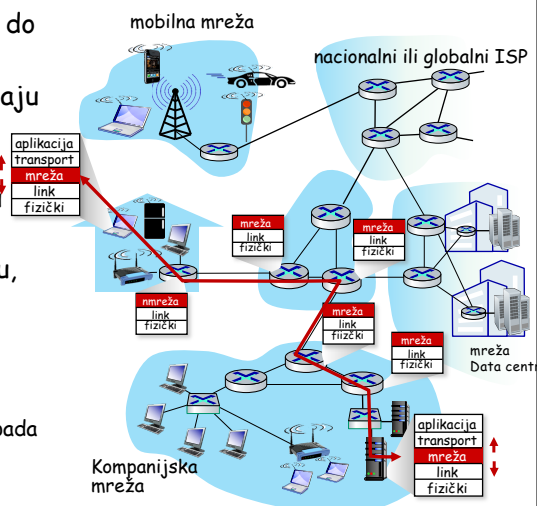
5.4 Ruter

Mrežni nivo 5-1

1

Mrežni nivo

- Prenos segmenta od pošiljaoca do odredišta
- Na strani koja šalje enkapsuliraju se segmenti u datagrame
- Na strani prijema predaja segmenata transportnom nivou
- Protokoli mrežnog nivoa su implementirani u *svakom* hostu, ruteru
- Ruter
 - ispituje polja zaglavlja svakog IP datagrama kojeg prosleđuje
 - prosleđuje paket na izlaz koji pripada odgovarajućoj ruti od izvora do destinacije



Mrežni nivo 5-2

2

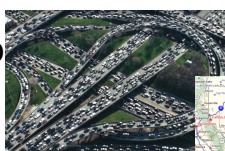
Ključne funkcije mrežnog nivoa

- **prosleđivanje:** prenos paketa sa ulaza rutera na odgovarajući izlaz
- **rutiranje:** izbor rute kojom se paketi prenose od izvora do destinacije.

- *Algoritmi rutiranja*

analogija:

- **rutiranje:** proces planiranja putovanja
- **prosleđivanje:** proces prolaska kroz jednu raskrscopicu



prosleđivanje



rutiranje

Mrežni nivo 5-3

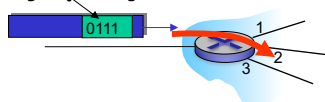
3

Mrežni nivo: ravan podataka, ravan kontrole

Ravan podataka

- Lokalna funkcija rutera
- Determiniše kako se datagram koji dolazi na ulazni port rutera prosleđuje na izlazni port
- Funkcija prosleđivanja

Destinaciona adresa u zaglavlju datagrama



Kontrolna ravan

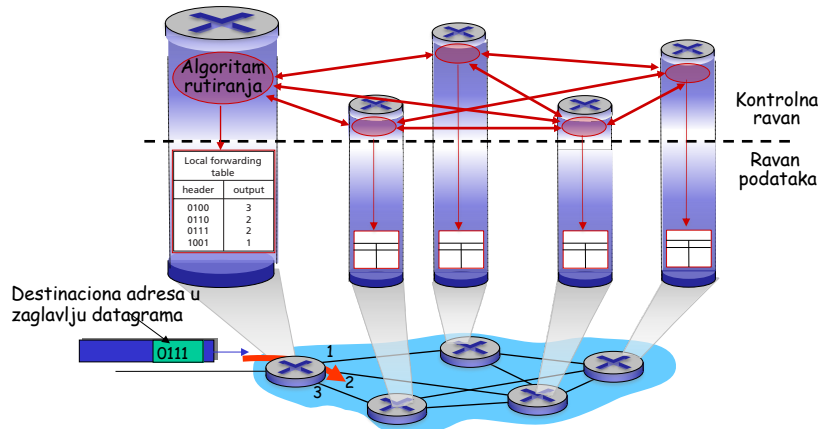
- Mrežna logika
- Određuje kako se datagram rutira duž putanje od kraja do kraja od izvorišnog do odredišnog hosta
- Dva pristupa:
 - *Tradicionalni algoritmi rutiranja:* implementirani u ruterima
 - *Software Defined Networking (SDN):* implementirani u udaljenim serverima

Mrežni nivo 5-4

4

Distribuirana kontrolna ravan

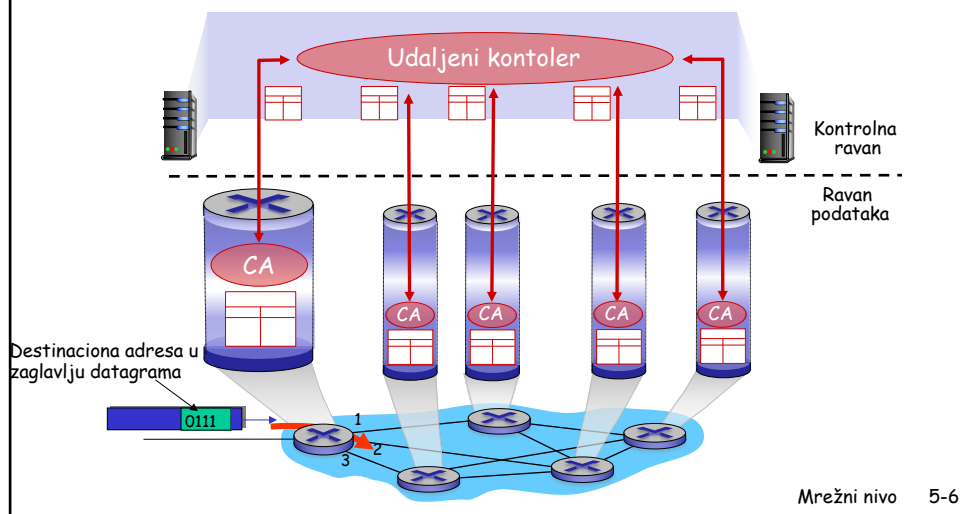
Individualni algoritmi rutiranja se izvršavaju samostalno u svakom ruteru i interaguju u kontrolnoj ravni



5

Centralizovana kontrolna ravan

Udaljeni kontroler interaguje sa lokalnim kontrolnim agentima (CAs)



6

Mrežni servisni model

Pitanje: Koji *servisni model* nudi “kanal” koji transportuje datagrame od pošiljaoca do prijemnika?

Primjer servisa za individualne datagrame:

- Garantovana predaja
- Garantovana predaja sa kašnjenjem manjim od određene vrijednosti (npr. 100ms)

Primjer servisa za tok datagrama:

- Redosledna predaja datagrama
- Garantovani minimalni protok toka
- Ograničene promjene u međupaketskim intervalima
- Nivo zaštite

Mrežni nivo 5-7

7

Modeli servisa mrežnog nivoa:

Mrežna Arhitektura	Model Servisa	Garantovani QoS				"Congestion Feedback"
		Brzina	Gub.	Red.	Tajm.	
Internet	<i>best effort</i>	bez	ne	ne	ne	ne (preko gubitaka)
ATM	CBR	konstantna brzina	da	da	da	nema zagušenja
ATM	VBR	garantov. brzina	da	da	da	nema zagušenja
Internet	Intserv	da	da	da	da	ne
Internet	Diffserv	moгуće	mog.	mog.	ne	

Mrežni nivo 5-8

8

Glava 5: Mrežni nivo

5.1 Uvod

5.2 IP (*Internet Protocol*)

- Format datagrama
- IP adresiranje

5.3 Rutiranje

- *Link state*
- *Distance Vector*
- Hijerarhijsko rutiranje
- Protokoli rutiranja

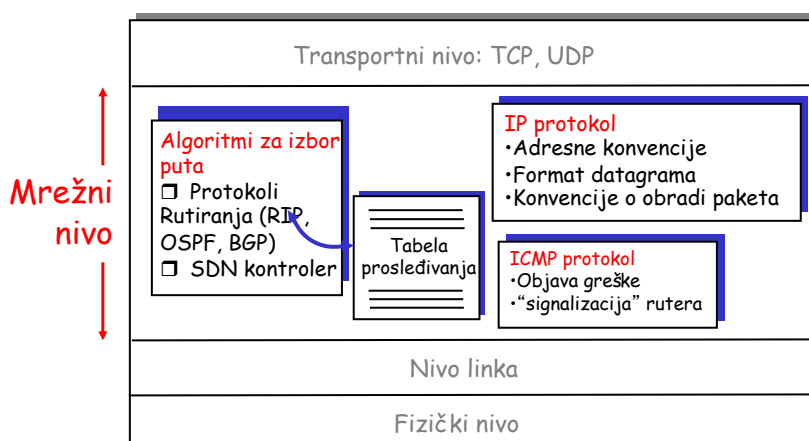
5.4 Ruter

Mrežni nivo 5-9

9

Internet mrežni nivo

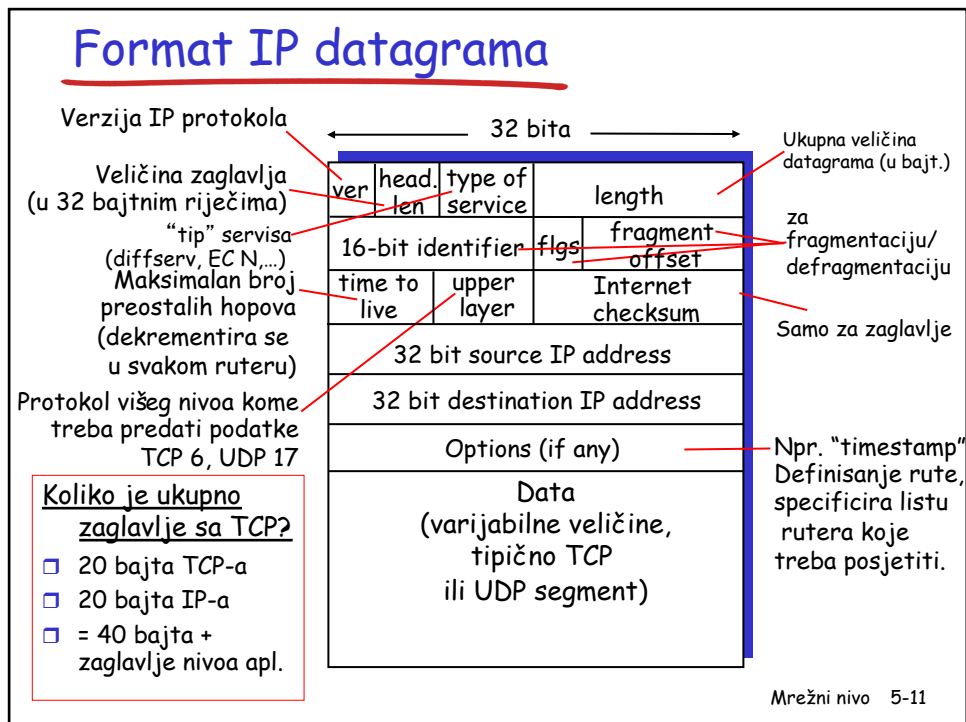
Host, ruter funkcije mrežnog nivoa:



Mrežni nivo 5-10

10

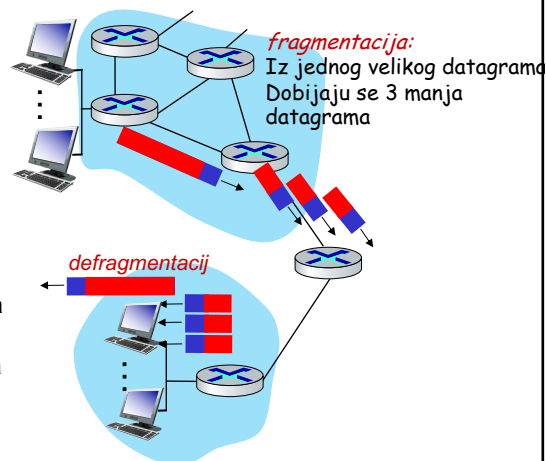
Format IP datagrama



11

IP Fragmentacija & Defragmentacija

- Mrežni linkovi imaju MTU (*max.transfer size*) ili najveći mogući okvir nivoa linka.
 - Različiti tipovi linkova, različiti MTU-ovi
- veliki IP datagram se dijeli ("fragmentira") u okviru mreže
 - jedan datagram postaje više datagrama
 - "defragmentira" se samo na konačnoj destinaciji
 - IP biti zaglavljaja se koriste za identifikaciju redosleda vezanog za fragment



Mrežni nivo 5-12

12

IP fragmentacija, defragmentacija

Primjer:

- Datagram od 4000B
- MTU = 1500B

dužina =4000	ID =x	fragflag =0	offset =0
-----------------	----------	----------------	--------------

Jedan veliki datagram se dijeli na više manjih datagrama

1480 B u polju podataka

offset = 1480/8

dužina =1500	ID =x	fragflag =1	offset =0
-----------------	----------	----------------	--------------

dužina =1500	ID =x	fragflag =1	offset =185
-----------------	----------	----------------	----------------

dužina =1040	ID =x	fragflag =0	offset =370
-----------------	----------	----------------	----------------

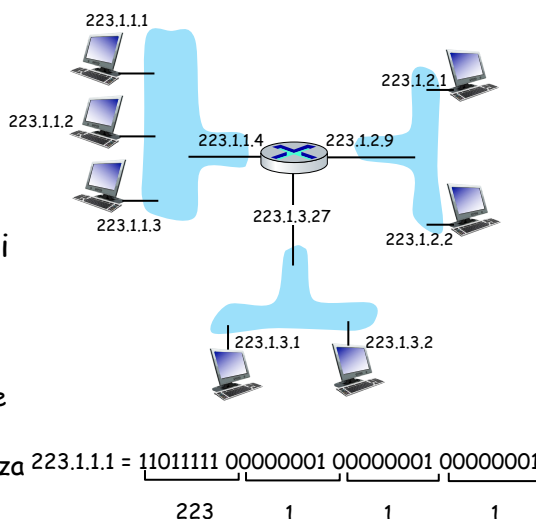
Mrežni nivo 5-13

13

IP Adresiranje: uvod

- IP adresa: 32-bitni identifikator interfejsa hosta ili rutera
- interfejs: veza između host/rutera i fizičkog linka

- ruteri tipično imaju više interfejsa
- i host može imati više interfejsa
- IP adrese su vezane za svaki interfejs

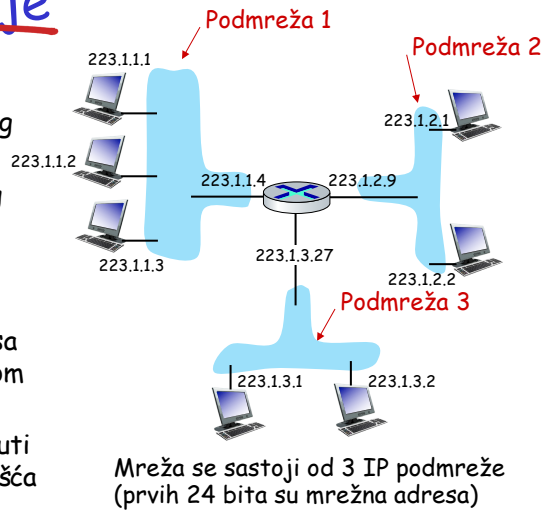


Mrežni nivo 5-14

14

IP Adresiranje

- IP adresiranje:
 - Mrežni dio (biti višeg reda)
 - Dio hosta (biti nižeg reda)
- Šta je mreža? (iz perspektive IP adrese)
 - Interfejsi uređaja sa istim mrežnim dijelom IP adrese
 - mogu fizički dosegnuti jedni druge bez učešća rutera



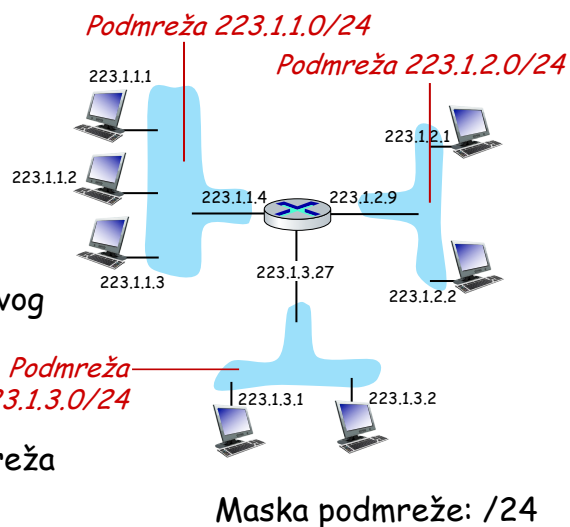
Mrežni nivo 5-15

15

Podmreža

Napomena

- Da bi odredili podmreže, treba razdvojiti svaki interfejs od njegovog hosta ili rutera, kreirajući ostrva izolovanih mreža.
- Svaka izolovana mreža se zove **podmreža**.

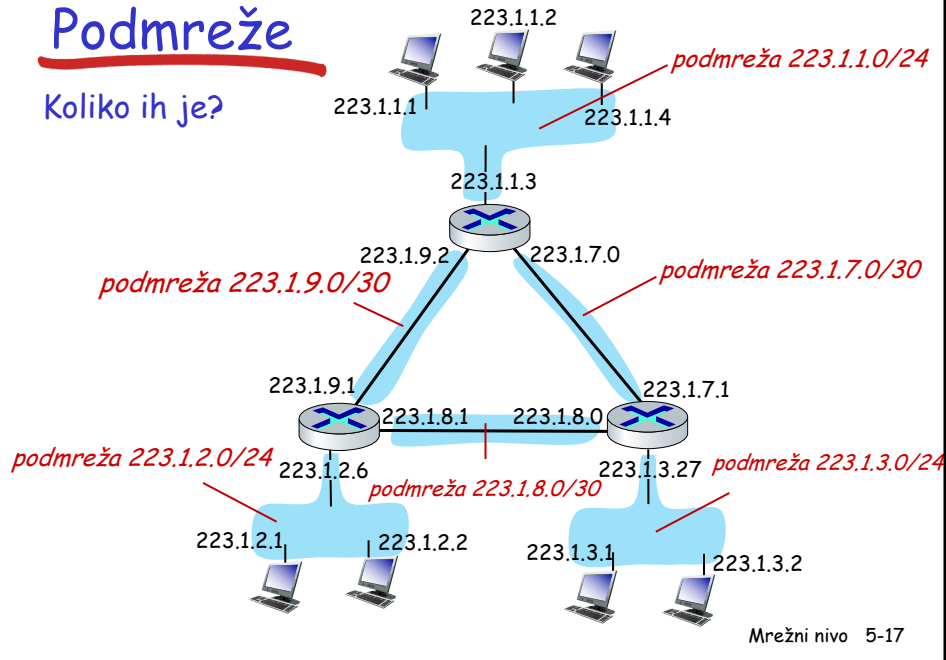


Mrežni nivo 5-16

16

Podmreže

Koliko ih je?

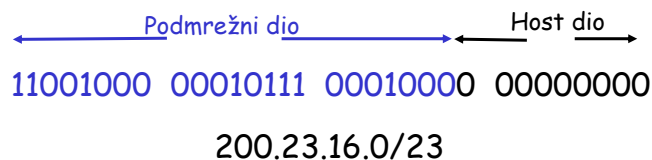


17

IP adresiranje: CIDR

CIDR: Classless InterDomain Routing

- Podmrežni dio adrese je proizvoljne veličine
- Format adrese: a.b.c.d/x, gdje je x broj bita u mrežnom dijelu adrese



Mrežni nivo 5-18

18

IP adrese: kako dobiti IP adresu?

Kako *host* dobija IP adresu?

- od strane sistem administratora
 - Winl: control-panel->network->configuration->tcp/ip->properties
 - UNIX: /etc/rc.config
- dinamički od DHCP servera
 - DHCP (*Dynamic Host Configuration Protocol*)
 - *plug-and-play*

Mrežni nivo 5-19

19

DHCP: Dynamic Host Configuration Protocol

Cilj je omogućiti hostu dinamičko dobijanje adresa, prilikom povezivanja na mreću, od DHCP servera

- Može obnoviti adresu koju je već koristio
- Omogućava "reuse" adresa (host zadržava adresu dok je uključen)
- Olakšava pristup mobilnim korisnicima koji se pridružuju mreži

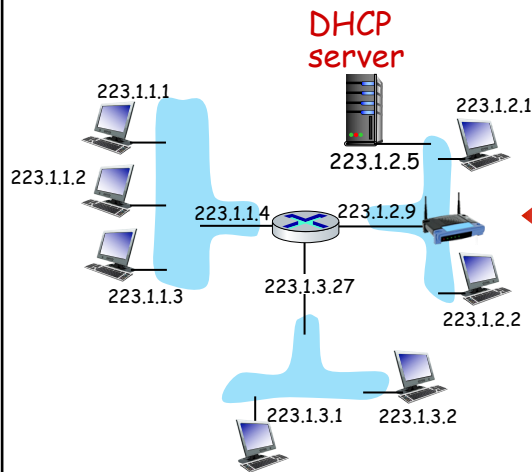
Pregled DHCP:

- host svima šalje "DHCP discover" poruku (UDP segment na port 67)
- DHCP server odgovara "DHCP offer" porukom
- host zahtijeva IP adresu "DHCP request" porukom
- DHCP server šalje adresu "DHCP ack" porukom

Mrežni nivo 5-20

20

DHCP klijent-server scenario



Uobičajeno, DHCP server, smješten unutar rutera, opslužuje sve podmreže na koje je ruter povezan

dolazećem DHCP klijentu treba adresa u ovoj mreži

Mrežni nivo 5-21

21

DHCP klijent-server scenario

DHCP server: 223.1.2.5

DHCP discover

Novi klijent

```
src : 0.0.0.0, 68
dest.: 255.255.255.255,67
yiaddr: 0.0.0.0
transaction ID: 654
```

DHCP offer

```
src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddr: 223.1.2.4
transaction ID: 654
Lifetime: 3600s
```

DHCP request

```
src: 0.0.0.0, 68
dest:: 223.1.2.5, 67
yiaddr: 223.1.2.4
transaction ID: 655
Lifetime: 3600s
```

DHCP ACK

```
src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddr: 223.1.2.4
transaction ID: 655
Lifetime: 3600s
```

vrijeme

Mrežni nivo 5-22

22

DHCP nudi više od IP adrese

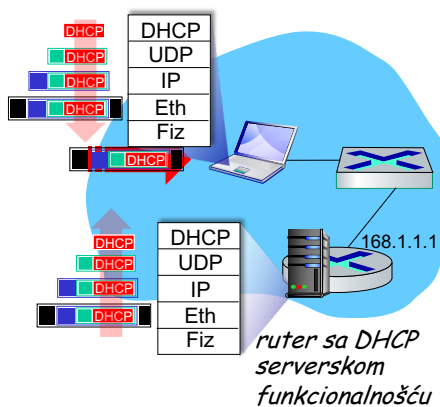
DHCP obezbeđuje više od same alokacije IP adrese u pod mreži:

- Adresu *gateway* rutera pod mreže
- Ime i IP adresu DNS servera
- *Subnet* masku (indiciira mrežni dio adrese)

Mrežni nivo 5-23

23

DHCP: primjer

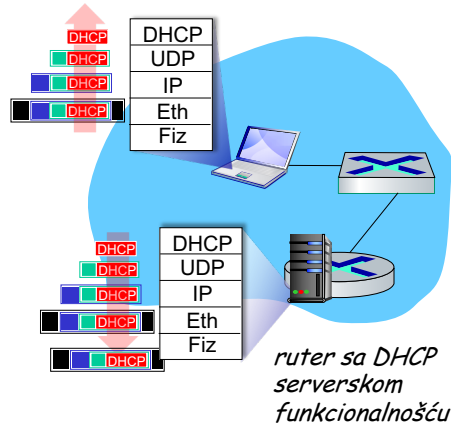


- Laptopu je potrebna IP adresa, adresa gateway-a i adresa DNS servera
- koristi DHCP
- DHCP zahtjev se enkapsulira u UDP segment, pa u IPdatagram, a zatim u IEEE 802.3 Ethernet fejm
- Ethernet fejm se šalje svim interfejsima u LAN-u i prima od strane DHCP servera
- Obavlja se suprotan proces enkapsulaciji

Mrežni nivo 5-24

24

DHCP: primjer



- DHCP server kreira DHCP potvrdu koja sadrži klijentsku IP adresu, IP adresu *gateway* rutera, ime i IP adresu DNS servera
- Frejm se prosleđuje do klijenta koji ga otvara
- Klijentu je poznata IP adresa, ime i IP adresa DNS servera, IP adresa *gateway* rutera

Mrežni nivo 5-25

25

IP adrese: kako dobiti IP adresu?

P: Kako mreža dobija podmrežni dio IP adrese?

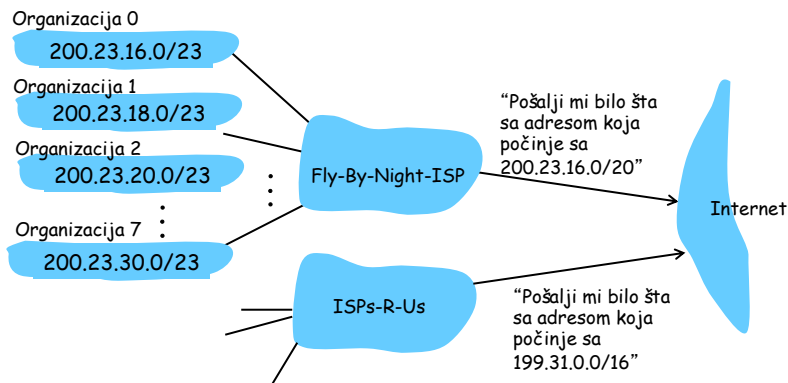
ISP-ov blok	<u>11001000</u>	<u>00010111</u>	<u>00010000</u>	00000000	200.23.16.0/20
Organizacija 0	<u>11001000</u>	<u>00010111</u>	<u>00010000</u>	00000000	200.23.16.0/23
Organizacija 1	<u>11001000</u>	<u>00010111</u>	<u>00010010</u>	00000000	200.23.18.0/23
Organizacija 2	<u>11001000</u>	<u>00010111</u>	<u>00010100</u>	00000000	200.23.20.0/23
...
Organizacija 7	<u>11001000</u>	<u>00010111</u>	<u>00011110</u>	00000000	200.23.30.0/23

Mrežni nivo 5-26

26

Hijerarhijsko adresiranje: agregacija ruta

Hijerarhijsko adresiranje dozvoljava efikasno oglašavanje informacije potrebne za rutiranje:

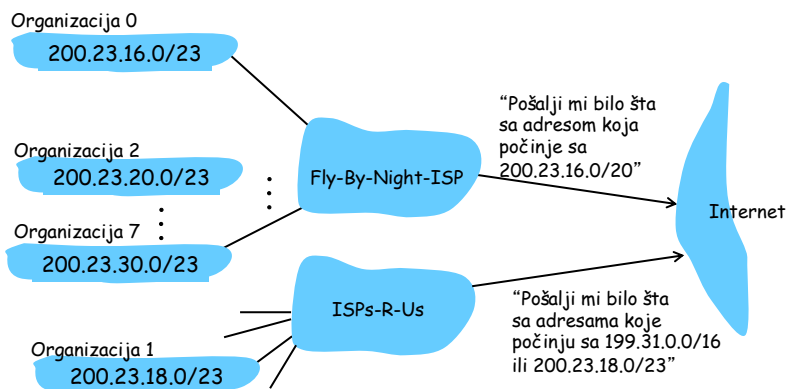


Mrežni nivo 5-27

27

Hijerarhijsko adresiranje: specifičnije rute

ISPs-R-Us ima više specifičnih ruta do Organizacije 1



Mrežni nivo 5-28

28

ICANN

Kako ISP dobija svoj blok adresa?

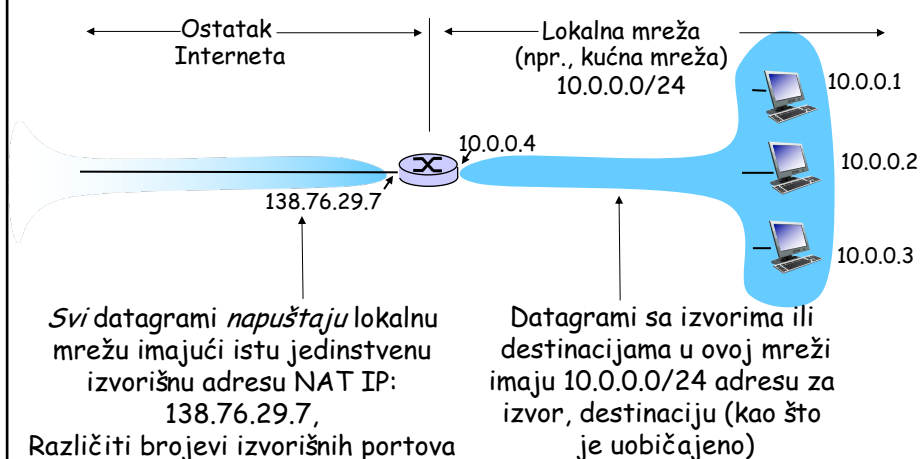
ICANN (Internet Corporation for Assigned Names and Numbers)

- Alocira IP adrese preko 5 regionalnih registara (ARIN, RIPE, APNIC, LACNIC i AFRINIC)
- Upravlja DNS root serverima
- Dodjeljuje imena domena
- Razrešava sporove
- Poslednje IPv4 adrese je dodijelila 2011. godine

Mrežni nivo 5-29

29

NAT: Network Address Translation



Mrežni nivo 5-30

30

NAT: Network Address Translation

- Lokalna mreža koristi samo jednu javnu IP adresu
 - Nema potrebe za dodjelom opsega adresa od strane ISP (samo jedna IP adresa se koristi za sve uređaje)
 - Mogu mijenjati adrese uređaja u lokalnim mrežama bez obavještanja "ostatka svijeta"
 - Mogu mijenjati ISP bez mijenjanja adresa uređaja u lokalnim mrežama
 - Uređaji unutar mreže se eksplicitno ne adresiraju, na vidljiv način "ostatku svijeta" (plus u smislu zaštite).

Mrežni nivo 5-31

31

NAT: Network Address Translation

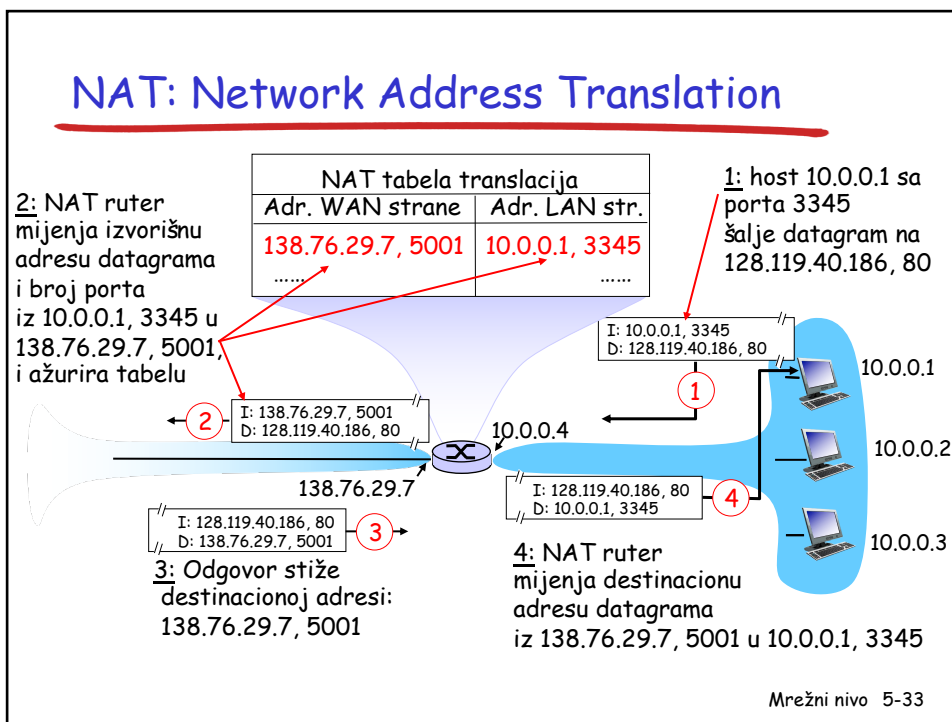
Implementacija: NAT ruter mora:

- *odlazni datagrami: zamijeniti* (izvorišnu IP adresu, broj porta) svakog odlaznog datagrama sa (NAT IP adresom, novim brojem porta)
 - ... udaljeni klijenti/serveri će odgovoriti korišćenjem (NAT IP adrese, novi broj porta) kao adrese destinacije.
- *zapamtiti (u NAT tabeli translacija)* svaki (izvorišna IP adresa, broj porta) i (NAT IP adresa, novi broj porta) u vidu translacionog para
- *dolazeći datagrami: zamijeniti* (NAT IP adresu, novi broj porta) u polju destinacije svakog dolaznog datagrama sa odgovarajućim (izvorišna IP adresa, broj porta) smještenim u NAT tabeli

Mrežni nivo 5-32

32

NAT: Network Address Translation



33

NAT: Network Address Translation

- ❑ 16-bitno polje broja porta:
 - 65536 simultanih veza sa jednom adresom sa LAN strane!
- ❑ NAT je kontraverzan:
 - Ruteri bi trebali vršiti obradu samo do nivoa 3
 - Narušava prirodu od kraja do kraja
 - NAT mora biti uzet u obzir od strane dizajnera aplikacija, npr., P2P aplikacija
 - Oskudica adresa se ublažila i prije upotrebe IPv6
 - Broj porta se posredno koristi za adresiranje računara
- ❑ NAT se intenzivno koristi u praksi

Mrežni nivo 5-34

34

IPv6

- Inicijalna motivacija: 32-bitni adresni prostor će vrlo brzo u potpunosti biti dodijeljen.
- Dodatna motivacija:
 - Format zaglavlja pomaže obradi/prosleđivanju
 - Promjene zaglavlja uključuju QoS
- IPv6 format datagrama:
 - Zaglavlje fiksne-dužine od 40B
 - Nije dozvoljena fragmentacija

Mrežni nivo 5-35

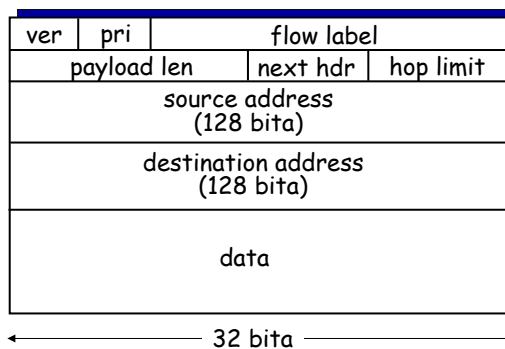
35

IPv6 zaglavlje (nastavak)

Priority: identifikuje prioritet između datagrama u "toku"

Flow Label: identifikuje datagrame u istom "toku".
(koncept "toka" nije precizno definisan).

Next header: identifikuje protokola višeg nivoa za podatke



Mrežni nivo 5-36

36

Druge izmjene u odnosu na IPv4

- ❑ **Checksum:** potpuno uklonjena kako bi se smanjila obrada na svakom hopu
- ❑ **Options:** dozvoljene, ali van zaglavlja, indicirano sa "Next Header" poljem
- ❑ **ICMPv6:** nova verzija ICMP
 - dodatni tipovi poruka, npr. "Packet Too Big"
 - funkcija upravljanja multicast grupama

Mrežni nivo 5-37

37

IPv6 adresiranje

Format:

- ❑ RFC 4291 (Februar 2006)
- ❑ 128 bita
- ❑ Predstavlja se u vidu 8 grupa po četiri heksadecimalna broja
- ❑ X:X:X:X:X:X:X
- ❑ 111111000011010 0100001010111001 0000000000011011
0000000000000000 0000000000000000 0001001011010000
0000000001011011 0000011010110000
- ❑ FE1A:42B9:001B:0000:0000:12D0:005B:06B0
- ❑ FE1A:42B9:001B:0:0:12D0:005B:06B0 (grupa od četiri 0 se može prikazati jednom 0)
- ❑ FE1A:42B9:1B::12D0:5B:6B0 (više susjednih grupa od četiri 0 se prikazuje sa ::, koja se može pojaviti samo jednom)
- ❑ 2001:4C::50:0:0:741
- ❑ 2001:004C::0050:0000:0000:0741
- ❑ 2001:004C:0000:0000:0050:0000:0000:0741

Telekomunikacione mreže 6-38

38

IPv6 adresiranje

IPv6 prefiks:

- ❑ Slično kao kod IPv4: IPv6adresa/dužina prefiksa
- ❑ 200C:001b:1100:0:0:0:0:0/40 ili 200C:1b:1100::/40
- ❑ Koristi se CIDR rutiranje

Telekomunikacione mreže 6-39

39

Glava 5: Mrežni nivo

5.1 Uvod

5.2 IP (Internet Protocol)

- ❑ Format datagrama
- ❑ IP adresiranje

5.3 Rutiranje

- ❑ *Link state*
- ❑ *Distance Vector*
- ❑ Hijerarhijsko rutiranje
- ❑ Protokoli rutiranja

5.4 Ruter

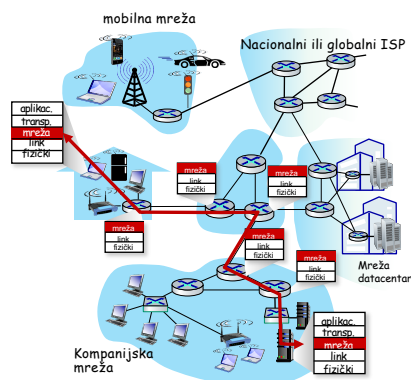
Mrežni nivo 5-40

40

Protokoli rutiranja

Protokol rutiranja određuju rutu između izvorišnog i odredišnog hosta koja sadrži jedan ili više rutera.

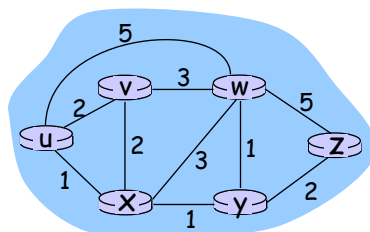
- Put predstavlja niz rutera preko kojih se paket može prenijeti od izvora do destinacije
- Ruta je put koji je po nekom kriterijumu najbolji (npr. Najmanji broj rutera=
- Rutiranje je jedan od 10 najvećih izazova na Internetu.



Mrežni nivo 5-41

41

Abstrakcija pomoću grafa



Graf: $G = (N, E)$

$N = \text{skup rutera} = \{ u, v, w, x, y, z \}$

$E = \text{skup linkova} = \{ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$

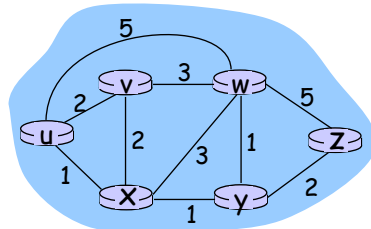
Napomena: Abstrakcija pomoću grafa je korisna i u drugim mrežnim kontekstima.

Primjer: P2P, gdje je N skup peer-ova, a E skup TCP konekcija

Mrežni nivo 5-42

42

Abstrakcija pomoću grafa: troškovi



• $c(x,x')$ = težinski faktor (cost) linka (x,x')

- npr., $c(w,z) = 5$

• težinski faktor može biti uvijek 1, ili recipročan protoku, ili recipročan zagušenju

Težinski faktor puta $(x_1, x_2, x_3, \dots, x_p) = c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$

Pitanje: Koji je put sa najmanjim težinskim faktorom između u i z ?

Algoritam rutiranja je algoritam koji pronalazi put sa najmanjim težinskim faktorom

Mrežni nivo 5-43

43

Klasifikacija algoritama rutiranja

Globalna ili decentralizovana informacija?

Globalna:

- ❑ svi ruteri posjeduju kompletnu topologiju mreže sa informacijama o težinskim faktorima linkova
- ❑ *link state algoritmi*

Decentralizovani:

- ❑ ruter poznaje fizički povezane susjede i težinske faktore linkova do susjeda
- ❑ iterativni proces izračunavanja, razmjena informacija sa susjedima
- ❑ *distance vector algoritmi*

Statički ili dinamički?

Statički:

- ❑ Rute se sporo mijenjaju

Dinamički:

- ❑ Rute se mijenjaju mnogo brže
 - periodični *update*
 - kao odgovor na promjene težinskih faktora linkova

Mrežni nivo 5-44

44

Link-State algoritam rutiranja

Dijkstra algoritam

- Centralizovan algoritam kod koga su mrežna topologija i težinski faktori linkova poznati svim čvorištima
 - Dobijeni preko *link state broadcast* poruke
 - Sva čvorišta imaju istu informaciju
- Proračunava puteve najmanjih težinskih faktora od jednog čvorišta ("izvor") do svih ostalih čvorova
 - generiše **tabelu rutiranja** za to čvorište
- iterativni: poslije k iteracija, poznat je put sa najmanjim težinskim faktorom do k destinacija

Notacija:

- $c(x,y)$: težinski faktor linka od čvorišta x do y koji ima beskonačnu vrijednost ukoliko čvorišta nijesu susjedi
- $D(A)$: trenutna vrijednost težinskog faktora puta od izvorišta do destinacije A
- $p(A)$: sledeće čvorište duž puta od izvorišta do čvorišta A , koje je susjed A
- N' : skup čvorišta čiji su najniži težinski faktori puta poznati

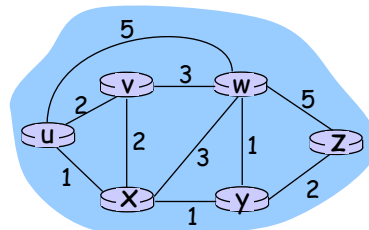
Mrežni nivo 5-45

45

Dijkstra Algoritam (na čvoru u)

1 Inicijalizacija:

- 2 $N' = \{u\}$
- 3 Za sva čvorišta A
- 4 Ako je A susjed čvorištu u
- 5 tada $D(A) = c(u,A)$
- 6 else $D(A) = \infty$



7

8 Petlja

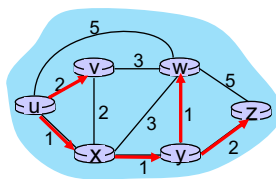
- 9 Pronaći B koje nije u N' tako da je $D(B)$ minimalno
- 10 dodati B skupu N'
- 11 update $D(A)$ za sve A susjede B koji nijesu u N' :
 $D(A) = \min(D(A), D(B) + c(B,A))$
- 12 /* novi težinski faktori za A su ili stari težinski faktori za A ili su poznati
- 14 najmanji težinski faktori puta do B plus težinski faktori od B do A */
- 15 dok sva čvorišta ne postanu članovi skupa N'

Mrežni nivo 5-46

46

Dijkstra algoritam

Step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	u,x	2,u	4,x	1,u	2,x	∞
2	u,x,y	2,u	3,y	1,u	2,x	4,y
3	u,x,y,w	2,u	3,y	1,u	2,x	4,y
4	u,x,y,w	2,u	3,y	1,u	2,x	4,y
5	u,x,y,w,z	2,u	3,y	1,u	2,x	4,y



Inicijalizacija (korak 0)

Za sve A: ako je A susjed tada je $D(A) = c_{u,A}$

Pronađi A sa minimalnim $D(A)$ koji nije u N'
dodaj A u N'

ažurirati $D(B)$ za sve B susjede A koji nisu u N' :

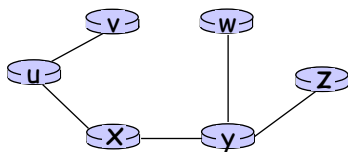
$$D(B) = \min(D(B), D(A) + c_{A,B})$$

Mrežni nivo 5-47

47

Dijkstra algoritam (čvorište u)

Rezultantna shortest-path topologija iz čvorišta u:



Rezultantna tabela prosleđivanja u čvorištu u:

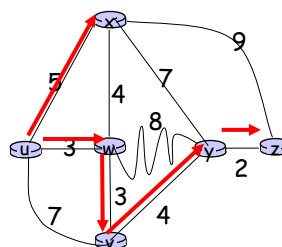
destinacija	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Mrežni nivo 5-48

48

Dijkstra algoritam: primjer

Korak	N'	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(y), p(y)$	$D(z), p(z)$
0	u	7,u	3,u	5,u	∞	∞
1	uw	6,w	5,u	11,w	∞	
2	uwx	6,w	u	11,w	14,x	
3	uwxv			10,v	14,x	
4	uwxvy				12,y	
5	uwxvyz					



Napomene:

- Pronaći najkraći put praćenjem prethodnih čvorišta
- Linkovi mogu biti prekinuti

Mrežni nivo 5-49

49

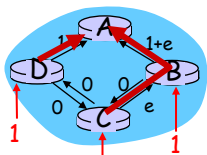
Dijkstra's algoritam, diskusija

Kompleksnost algoritma: n čvorišta

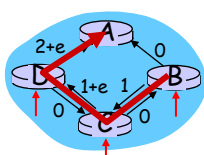
- Tokom svake od n iteracija je potrebno provjeriti sva čvorišta koja nijesu u N'
- $n*(n+1)/2$ komparacija tako da je kompleksnost proporcionalna n^2
- Moguće su efikasnije implementacije čija su kompleksnosti proporcionalne $n \log n$

Kompleksnost poruka

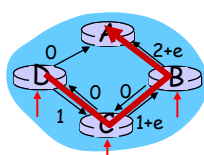
- Svaki ruter mora poslati svim ruterima stanje svojih linkova
- Poruka svakog rutera mora preći n linkova



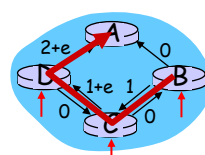
inicijalno



izračunavanje



izračunavanje



izračunavanje

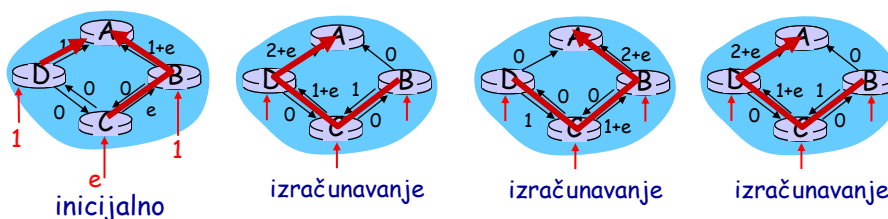
Mrežni nivo 5-50

50

Dijkstra's algoritam, diskusija

Moguće su oscilacije:

- npr., težinski faktor linka = količina prenešenog saobraćaja



Mrežni nivo 5-51

51

Distance Vector algoritam (1)

Bellman-Ford jednačina (dinamičko programiranje)

Neka je

$D_x(y) :=$ težinski faktor puta sa najmanjim troškovima od x do y

Tada

$$D_x(y) = \min_v \{ c(x,v) + D_v(y) \}$$

Gdje se \min_v uzima u odnosu na sve susjede rutera x

Mrežni nivo 5-52

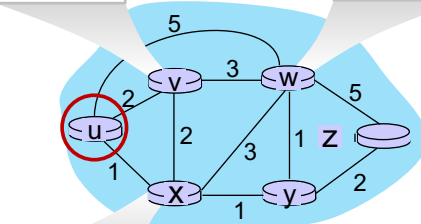
52

Distance Vector algoritam (2)

Neka susjedi x, v, w , znaju za destinaciju z .

$$D_v(z) = 5$$

$$D_w(z) = 3$$



$$D_x(z) = 3$$

Bellman-Ford jednačina

$$D_u(z) = \min \{ c_{u,v} + D_v(z), c_{u,x} + D_x(z), c_{u,w} + D_w(z) \}$$

$$= \min \{ 2 + 5, 1 + 3, 5 + 3 \} = 4$$

Čvorište koje dostigne minimum je sledeći korak (hop) u najkraćem putu → tabela prosleđivanja

Mrežni nivo 5-53

53

Distance Vector algoritam (3)

- $D_x(y)$ predstavlja procjenu najmanjeg težinskog faktora od x do y
- *Distance vector*: $D_x = [D_x(y): y \in N]$
- Čvorište x poznaje težinske faktore do svakog svog susjeda v : $c(x,v)$
- Čvorište x nadzire $D_x = [D_x(y): y \in N]$
- Čvorište x takođe nadzire *distance vector-e* svojih susjeda
 - Za svakog susjeda v , x nadzire $D_v = [D_v(y): y \in N]$

Mrežni nivo 5-54

54

Distance vector algoritam (4)

Osnovna ideja:

- Svako čvorište periodično šalje procjenu svog *distance vector*-a svojim susjedima
- Kada čvorište x primi novu DV estimaciju od svog susjeda v , ažurira svoj sopstveni DV korišćenjem B-F jednačine:

$$D_x(y) \leftarrow \min_v \{c(x,v) + D_v(y)\} \quad \text{za svako čvorište } y \in N$$

- U većem broju slučajeva, pod normalnim okolnostima, procjena $D_x(y)$ konvergira stvarnom najmanjem težinskom faktoru $d_x(y)$

Mrežni nivo 5-55

55

Distance Vector algoritam (5)

Iterativni, asinhron:

svaka lokalna iteracija je uzrokovana:

- lokalni težinski faktori linka su promjenljivi
- porukama od susjeda: najmanji težinski faktori puta su promijenjeni

Distribuiran:

- svako čvorište obavještava susjeda *samo* kada se njegov put sa najmanjim težinskim faktorom promijeni
 - susjedi informišu susjede ako je to potrebno
 - nema akcije bez dobijanja obavještenja

Svako čvorište:

čeka poruku od susjeda o promjeni težinskih faktora linka

Preračunava *distance vector* tabelu

Ako se najmanji težinski faktori puta do bilo koje destinacije promijene, *obavještava* susjede

Mrežni nivo 5-56

56

$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\} = \min\{2+0, 7+1\} = 2$
 $D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\} = \min\{2+1, 7+0\} = 3$

Tabela čvorišta x
 Tež. fakt. do

	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

Tabela čvorišta y
 Tež. fakt. do

	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

Tabela čvorišta z
 Tež. fakt. do

	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

(Note: In the original image, the values 0, 2, 7 in the x-table; 2, 0, 1 in the y-table; and 7, 1, 0 in the z-table are circled in red. Arrows indicate the flow of information between these tables.)

→ vrijeme
 Mrežni nivo 5-57

57

Distance Vector: promjene težinskih faktora linka

Promjene težinskih faktora linka:

- Čvorište detektuje lokalne promjene težinskih faktora linka
- ažuriranje *distance* tabele
- Ako se težinski faktori promijene na putu sa najmanjim težinskim faktorom, obaviještava susjede

U trenutku t_0 , y detektuje promjenu težinskog faktora linka, ažurira njegov DV, i informiše susjede.

“dobre vijesti brzo putuju” U trenutku t_1 , z prima update od y i ažurira svoju *distance* tabelu. Izračunava novi najmanji težinski faktor do x i šalje svojim susjedima svoj DV.

U trenutku t_2 , y prima od z *update* i ažurira svoju *distance* tabelu. Y-ov najmanji težinski faktor se ne mijenja i stoga y ne šalje nikakvu poruku ruteru z.

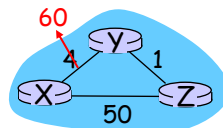
Mrežni nivo 5-58

58

Distance Vector: promjene težinskih faktora linka

Promjene troškova linka:

- dobre vijesti se brzo prenose
- loše vijesti se sporije prenose - problem "brojanje do ∞ !"
- 44 iteracije do stabilizacije algoritma



Tehnika split-horizon

- Update ruta šalje težinske faktore ruta koje se mogu doseći preko drugih portova.

Poissoned reverse (lažno rastojanje):

- Ako Z rutira preko Y do X :
 - Z govori Y da je njegova (Z-ova) udaljenost do X beskonačna (tako da Y ne bi rutirala do X preko Z)
- Da li će to riješiti problem brojanja do beskonačnosti?

Mrežni nivo 5-59

59

Poređenje LS i DV algoritama

Kompleksnost poruke

- **LS:** sa n čvorišta, šalje se n^2 poruka
- **DV:** razmjena samo između susjeda
 - Konvergencija varira u vremenu

Brzina konvergencija

- **LS:** $O(n^2)$ algoritam zahtijeva $O(n^2)$ poruka
 - Mogu imati oscilacije
- **DV:** konvergencija varira u vremenu
 - Može biti petlji
 - Problem brojanja do ∞

Robustnost: šta se dešava kada ruter otkaže?

LS:

- Čvorište može objaviti težinski faktor neispravnog linka
- Svako čvorište proračunava svoju sopstvenu tabelu

DV:

- DV čvorište može objaviti težinski faktor neispravnog linka
- Tabelu svakog čvorišta koriste drugi
 - Greška se prenosi kroz mrežu

Mrežni nivo 5-60

60

Skalabilno rutiranje

Prethodna analiza je bila - idealizacija

- Svi ruteri su identični
- *flat* mreža

... *praksa je drugačija*

veličina: nekoliko stotina miliona destinacija:

- ne mogu se sve destinacije smjestiti u tabele rutiranja!
- razmjena tabela rutiranja može oboriti linkove!
- LS može izazvati potiskivanje saobraćaja na račun *broadcasta* tabela
- DV teško može konvergirati

administrativna autonomija

- internet = mreža svih mreža
- svaki mrežni administrator želi
 - kontrolu rutiranja u svojoj mreži
 - sakriti mrežnu organizaciju od ostalih

Mrežni nivo 5-61

61

Internet pristup skalabilnom rutiranju

- grupiše ruteru u regione, "autonomni sistemi" (AS)
- ruteri u istom AS izvršavaju isti protokol rutiranja
 - "intra-AS" protokol rutiranja se slično ponaša objašnjenim idealizovanim modelima
 - ruteri u različitim AS mogu izvršavati različite intra-AS protokole rutiranja

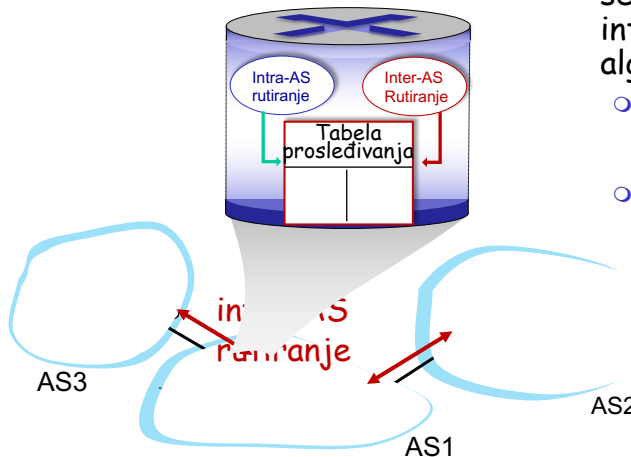
Gateway ruter

- Prosleđuje datagrame van AS

Mrežni nivo 5-62

62

Međupovezivanje AS-ma



- Tabela prosleđivanja se konfigurira i sa intra- i sa inter-AS algoritmom rutiranja
 - Intra-AS setuje sadržaje za interne destinacije
 - Inter-AS & Intra-AS setuju sadržaje za eksterne destinacije

Mrežni nivo 5-63

63

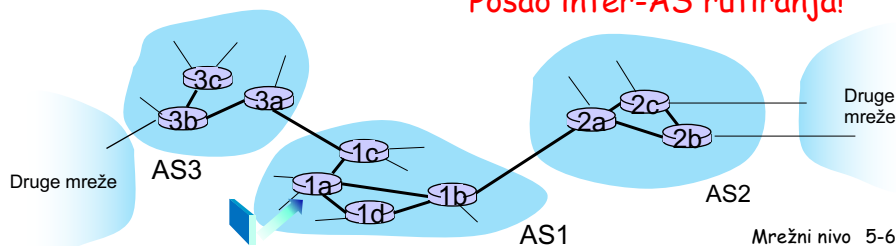
Inter-AS zadaci

- Neka ruter u AS1 primi datagram za koji je destinacija van AS1
 - Ruter bi trebao prosljediti paket prema gateway ruteru. Kojem?

Ruteri AS1 treba:

1. da nauče koje su destinacije dostižne preko AS2, a koje preko AS3
2. da prosljede tu informaciju o mogućnosti doseganja do svih rutera u AS1

Posao inter-AS rutiranja!



Mrežni nivo 5-64

64

Intra-AS Rutiranje

- ❑ Poznato kao **Interior Gateway Protocols (IGP)**
- ❑ Najpoznatiji Intra-AS protokoli rutiranja:
 - RIP: Routing Information Protocol
 - OSPF: Open Shortest Path First
 - IGRP: Interior Gateway Routing Protocol (vlasništvo kompanije Cisco)
 - IS-IS: Intermediate system to intermediate system

Mrežni nivo 5-65

65

OSPF (Open Shortest Path First)

- ❑ Interior Gateway Protocol (IGP)
- ❑ "open": javno dostupan
- ❑ Verzija 2 (RFC 2328) iz 1998
- ❑ Verzija 3 (RFC2740) iz 1999 podržava IPv4 i IPv6
- ❑ Koristi se u velikim kompanijskim mrežama zbog brze konvergencije, rješavanja problema petlji i balansiranja saobraćaja, dok operatori koriste IS-IS koji je pogodan za stabilne mreže
- ❑ Koristi "Link State" algoritam
 - LS širenje paketa
 - Mapa topologije na svakom čvorištu
 - Proračun rute korišćenjem Dijkstra algoritma
 - Broadcast svakih 30min
- ❑ OSPF oglašavanja nose po jednu informaciju po susjednom ruteru
- ❑ Širenje oglašavanja preko **čitavog AS** ("flooding")
 - Nose se u OSPF porukama direktno preko IP (a ne preko TCP ili UDP) pri čemu potrebne kontrole obavlja OSPF
- ❑ Radi smanjenja saobraćaja može se koristiti koncept DR (*designated router*) i *multicast* tabele.

Mrežni nivo 5-66

66

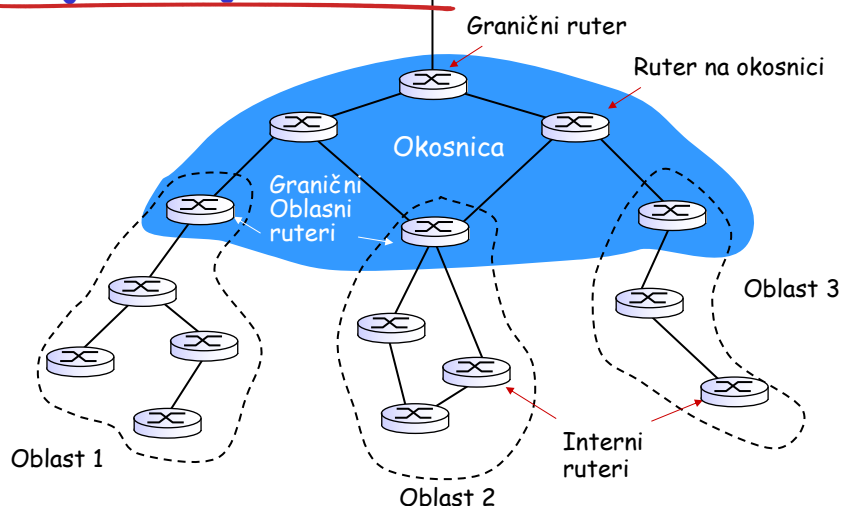
Napredne OSPF karakteristike (nema ih RIP)

- **Sigurnost:** za sve OSPF poruke se mora znati izvor (prevencija malicioznih aktivnosti) pri čemu se koriste lozinke ili MD5 kodiranje
- **Više** puteva sa istim troškovima je dozvoljeno (samo jedan put u RIP)
- Za svaki link, više metrika troškova za različiti **TOS** (npr., troškovi satelitskog linka su podešeni na "nisko" za *best effort*, visoko za servis u realnom vremenu)
- Integrisana uni- i **multicast** podrška:
 - *Multicast* OSPF (MOSPF) koristi istu bazu podataka o topologiji kao OSPF
- **Hijerarhijski** OSPF u velikim domenima.

Mrežni nivo 5-67

67

Hijerarhijski OSPF



Mrežni nivo 5-68

68

Hijerarhijski OSPF

- **Hijerarhija u dva nivoa:** lokalna mreža (oblast) i okosnica.
 - Oglašavanja o stanju linka samo u lokalnoj mreži (oblasti)
 - Svako čvorište ima detaljnu topologiju mreže; samo poznaje najkraći put do mreža u drugim mrežama.
- **Ruter na granici lokalne mreže:** “sumira” rastojanja do mreža u sopstvenoj zoni odgovornosti i to oglašava drugim ruterima na granicama lokalnih mreža.
- **Ruteri okosnice:** izvršavaju OSPF rutiranje samo na okosnici.
- **Granični ruteri:** povezivanje na druge AS.

Mrežni nivo 5-69

69

Internet inter-AS rutiranje: BGP

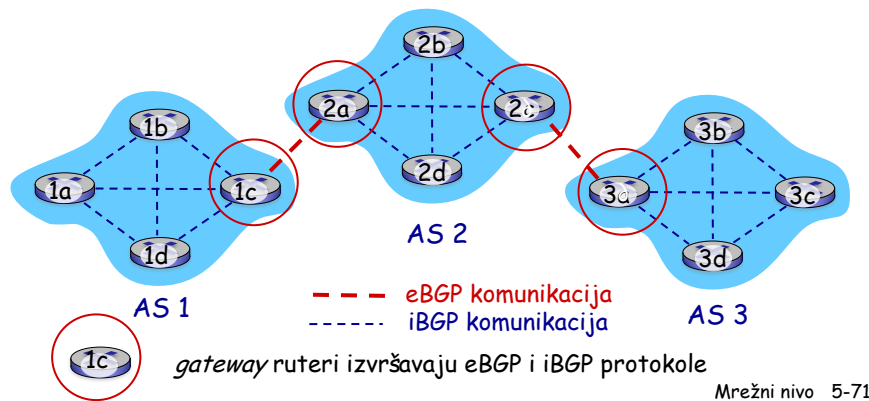
- **BGP (Border Gateway Protocol):** *de facto* standard
- Verzija 4 (RFC1771) iz 1994. je doživjela preko 20 korekcija, pri čemu je zadnja RFC4271 (iz 2006.)
- CIDR i agregacija ruta
- Naslijedio EGP čime je napravljena potpuna decentralizacija Interneta
- Mogu ga koristiti i kompanije kada OSPF nije dovoljno dobar i kada se radi o *multihomed* mreži (bolja redundansa).
- BGP omogućava svakom AS:
 1. Dobijanje informacije o dostiznosti sa susjednih AS-ova.
 2. Prosleđivanje prethodne informacije svim ruterima u okviru AS.
 3. Utvrđivanje “dobre” rute do podmreža baziranih na informaciji o dostiznosti i politici.
- Dozvoljava podmreži oglašavanje svog prisustva ostatku Interneta: “*Ovdje sam*”

Mrežni nivo 5-70

70

BGP osnove

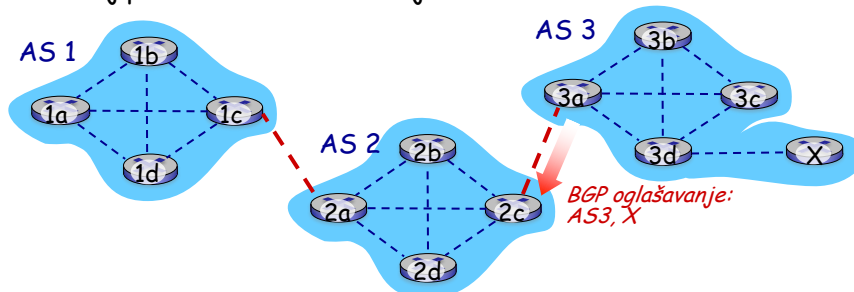
- Parovi rutera (BGP peer-ovi) razmjenjuju informaciju rutiranja preko semi-permanentne TCP konekcije (port 179): **BGP sesije**
- Svakih 60s šalje **keep alive** poruku
- Napomena:** BGP sesije ne odgovaraju fizičkim linkovima.
- Kada AS2 oglasi prefiks do AS1, AS2 **obećava** da će proslijediti bilo koji datagram koji je adresiran do tog prefiksa preko sebe.
 - AS2 može agregirati prefikse u oglašavanjima



71

Distribuirana informacija o dostižnosti

- Sa eBGP sesijom između 3a i 1c, AS3 šalje informaciju o dostižnosti prefiksa do AS1.
- 1c može tada koristiti iBGP za distribuciju ove nove informacije o dostizanju prefiksa do svih rutera u AS1
- 1b može tada ponovo oglasiti novu informaciju o dostizanju do AS2 preko 1b-2a eBGP sesije
- Kada ruter stekne znanje o novom prefiksu, kreira sadržaj za taj prefiks u tabeli rutiranja.



72

Atributi puta & BGP rute

- Kada oglašava prefiks, oglašavanje uključuje BGP attribute.
 - prefix + atributi = "ruta"
- Dva važna atributa:
 - **AS-PATH**: sadrži AS-ove preko kojih je oglašavanje prefiksa prošlo: AS67 AS17
 - **NEXT-HOP**: Indicira specifični interni-AS ruter do *next-hop* AS. (Može biti više linkova od trenutne AS do *next-hop* AS.)
- Kada *gateway* ruter primi oglašavanje rute, koristi **politiku importovanja** za potvrdu/odbijanje.

Mrežni nivo 5-73

73

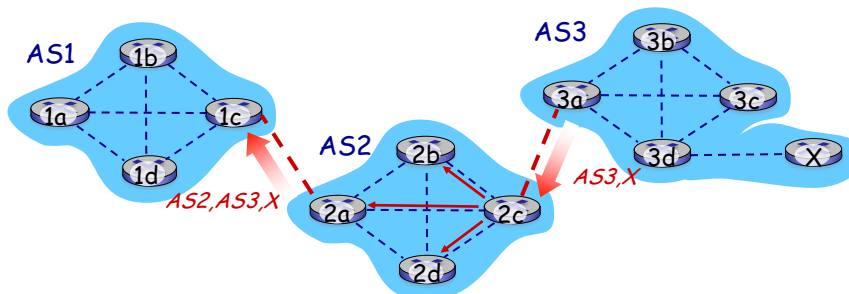
BGP izbor rute

- Ruter može naučiti više od jedne rute do istog prefiksa. Ruter mora odabrati rutu.
- Pravila eliminacije:
 1. Vrijednost atributa lokalne reference: odluka politike
 2. Najkraći AS-PATH
 3. Najbliži NEXT-HOP ruter: "vrući krompir" rutiranje
 4. Dodatni kriterijum

Mrežni nivo 5-74

74

BGP oglašavanje puta

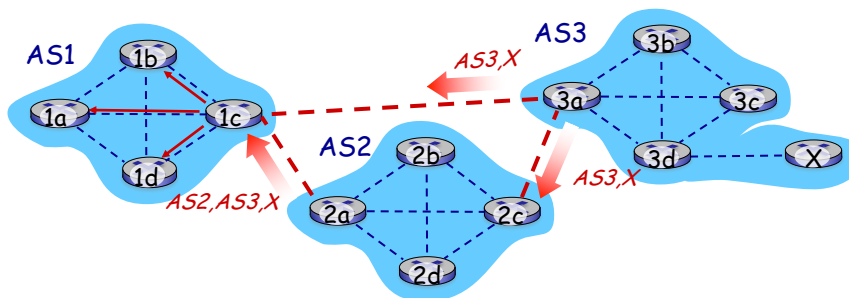


- AS2 ruter 2c dobija oglašavanje puta **AS3,X** (preko eBGP) od AS3 rutera 3a
- Bazirano na AS2 politici, AS2 ruter 2c prihvata put **AS3,X**, prosleđuju (preko iBGP) do svih AS2 rutera
- Bazirano na AS2 politici, AS2 ruter 2a oglašava (preko eBGP) put **AS2, AS3, X** do AS1 rutera 1c

Mrežni nivo 5-75

75

BGP oglašavanja ruta



gateway ruter može naučiti **više** puteva do destinacija:

- AS1 gateway ruter 1c uči put **AS2,AS3,X** od 2a
- AS1 gateway ruter 1c uči put **AS3,X** od 3a
- Bazirano na politici, AS1 gateway ruter 1c bira put **AS3,X**, i oglašava put **kroz AS1 preko iBGP**

Mrežni nivo 5-76

76

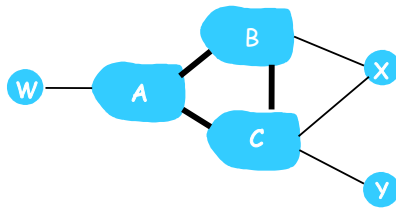
BGP poruke

- BGP poruke se razmjenjuju preko TCP.
- BGP poruke:
 - **OPEN**: otvara TCP vezu sa peer i vrši identifikaciju pošiljaoca
 - **UPDATE**: oglašava novi put (ili odbacuje stari)
 - **KEEPALIVE** održava vezu u odsustvu UPDATE poruka, a potvrđuje i OPEN zahtjev
 - **NOTIFICATION**: izvještava o greškama u prethodnoj poruci, a takođe se koristi za raskidanje veze

Mrežni nivo 5-77

77

BGP politika rutiranja

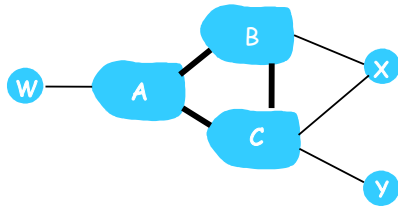


- A,B,C su **mreže provajdera**
- x,w,y su korisnici (mreža provajdera)
- x je **dual-homed**: povezan na dvije mreže
 - x ne želi da se saobraćaj rutira od B preko x do C
 - .. tako x neće oglašavati B rutu do C

Mrežni nivo 5-78

78

BGP: kontroliš ko rutira do tebe



- A oglašava B put Aw
- B oglašava X put BAw
- Da li će B oglašavati C put BAw?
 - Neće! B ne dobija “profit” za rutiranje CBAw pošto w i C nisu B-ovi korisnici
 - B želi da prinudi C da rutira do w preko A
 - B želi da rutira *samo* do/od njegovih korisnika!

Mrežni nivo 5-79

79

Zašto različito Intra- i Inter-AS rutiranje ?

Politika:

- Inter-AS: administrator želi kontrolu nad načinom rutiranja saobraćaja i time ko rutira kroz njegovu mrežu.
- Intra-AS: jedan administrator, nema potrebe za političkim odlukama

Veličina:

- hijerarhijsko rutiranje čuva veličinu tabele, smanjuje saobraćaj koji se odnosi na ažuriranje

Performanse:

- Intra-AS se može fokusirati na performanse
- Inter-AS politika može dominirati u odnosu na performanse

Mrežni nivo 5-80

80

Glava 5: Mrežni nivo

5.1 Uvod

5.2 IP (Internet Protocol)

- Format datagrama
- IP adresiranje

5.3 Rutiranje

- *Link state*
- *Distance Vector*
- Hijerarhijsko rutiranje
- Protokoli rutiranja

5.4 Ruter

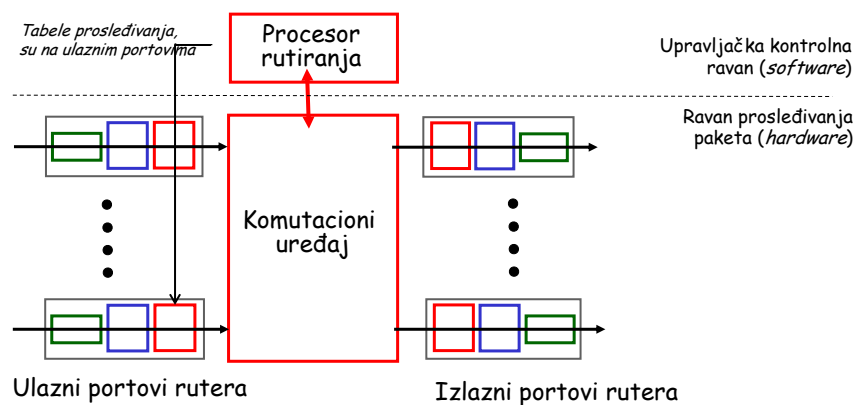
Mrežni nivo 5-81

81

Pregled arhitekture rutera

Dvije ključne funkcije rutera:

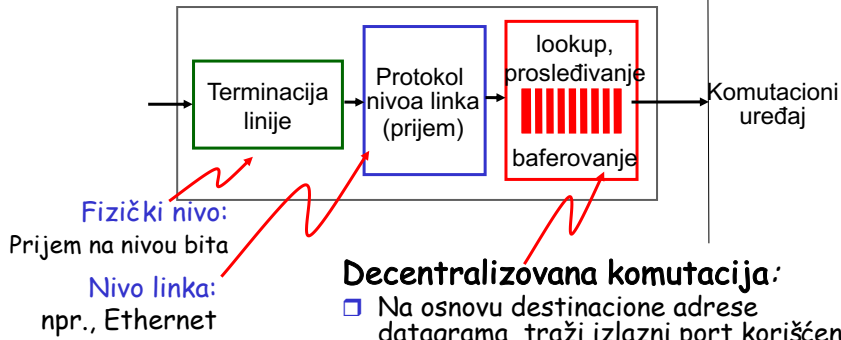
- Izvršava algoritme/protokole rutiranja (RIP, OSPF, BGP)
- *Prosleđuje (komutira) datagrame sa ulaznog na izlazni link*



Mrežni nivo 5-82

82

Funkcije ulaznog porta



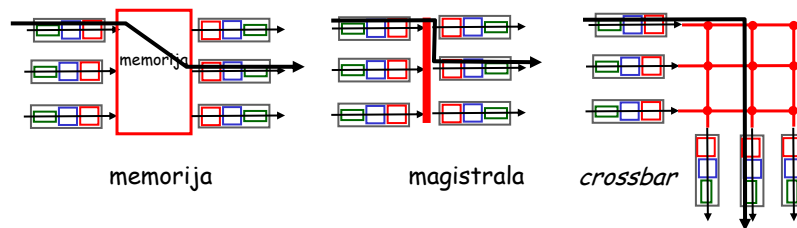
Decentralizovana komutacija:

- Na osnovu destinacione adrese datagrama, traži izlazni port korišćenjem tabele rutiranja u memoriji ulaznog porta
- Cilj: kompletirati obradu na ulaznom portu u skladu sa brzinom na linku
- Red čekanja (bafer): ako datagrami pristižu brže nego što je brzina prosleđivanja u Komutacionom uređaju

Mrežni nivo 5-83

83

Tri tipa komutacionih uređaja



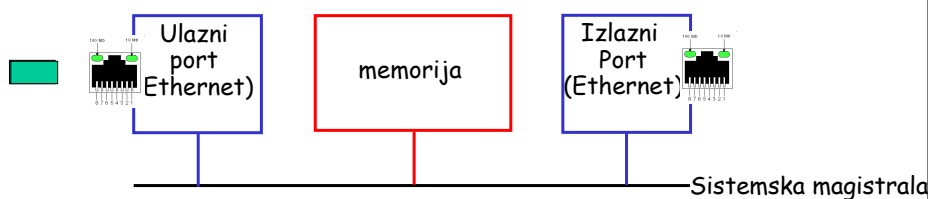
Mrežni nivo 5-84

84

Komutacija preko zajedničke memorije

Prva generacija rutera:

- tradicionalni računari sa komutacijom pod direktnom kontrolom CPU
- paketi se smještaju u memoriju sistema
- brzina ograničena brzinom memorije (svaki datagram se mora dva puta prenijeti preko magistrale)
- Cisco Catalyst *switch* serije 8500 (specifično rješenje)

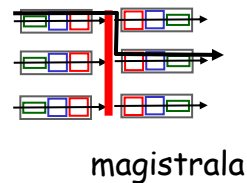


Mrežni nivo 5-85

85

Komutacija preko zajedničke magistrale

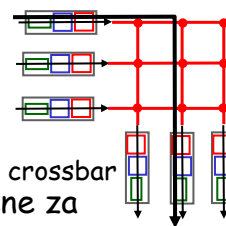
- Datagram se sa memorije ulaznog porta do memorije izlaznog porta prenosi preko zajedničke magistrale bez učešća procesora
- **Kolizija na magistrali:** brzina komutacije je ograničena kapacitetom magistrale
- Cisco 5600:
 - 32 Gb/s magistrala,
 - pristupni i kompanijski ruteri (neregionalne ili na okosnici)



Mrežni nivo 5-86

86

Prostorni komutatori

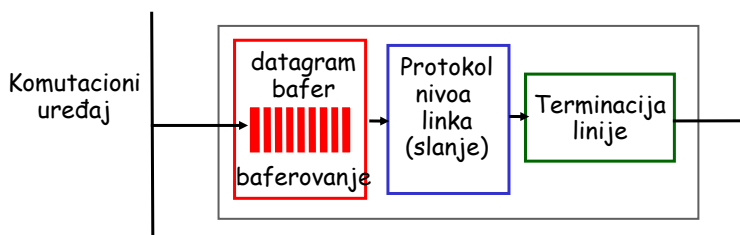


- Komutacione strukture su inicijalno razvijene za povezivanje procesora u multiprocesorsku arhitekturu
- Prevazilazi ograničenja kapaciteta magistrale
- Nudi više puteva između skupa ulaza i skupa izlaza
- Crossbar topologija
- Napredan dizajn: fragmentacija datagrama u ćelije fiksne dužine, komutiranje ćelija kroz uređaj.
- Cisco 12000:
 - komutira do 60Gb/s kroz komutacionu matricu
- Banyan, Clos, paralelni...
- Komutiraju pakete fiksne dužine

Mrežni nivo 5-87

87

Izlazni portovi



- **Baferovanje** se zahtijeva kada datagrami stižu iz uređaja većom brzinom nego što je brzina prenosa datagrama
- **Disciplina raspoređivanja (Scheduling)** bira za prenos datagrama u redovima čekanja

Mrežni nivo 5-88

88

Veličina bafera?

- RFC 3439 (*rule of thumb*) pravilo: srednja veličina bafera je jednaka “prosječno” RTT (npr. 250ms) pomnoženo sa kapacitetom linka C

$$RTT \cdot C$$

- npr., $C = 10\text{Gb/s}$ link: 298MB

- Neke preporuke ukazuju da su moguće i manje memorije: za N tokova, potrebna veličina bafera je

$$\frac{RTT \cdot C}{\sqrt{N}}$$

Mrežni nivo 5-89

89

Nivo mreže

Ispitna pitanja

1. Objasniti ključne funkcije mrežnog nivoa
2. Nacrtati IPv4 datagram i objasniti funkcije njegovih ključnih polja
3. Objasniti i na konkretnom primjeru ilustrovati funkcionisanje DHCP protokola
4. Objasniti i na konkretnom primjeru ilustrovati funkcionisanje NAT-a
5. Nacrtati IPv6 datagram i objasniti funkcije njegovih ključnih polja
6. Dijkstra algoritam
7. Belman Fordov algoritam
8. OSPF
9. BGP
10. Ruter

Mrežni nivo 90

90