

Računarske mreže
(2OER5O03)

Rutiranje i IP adrese

Auditivne vežbe



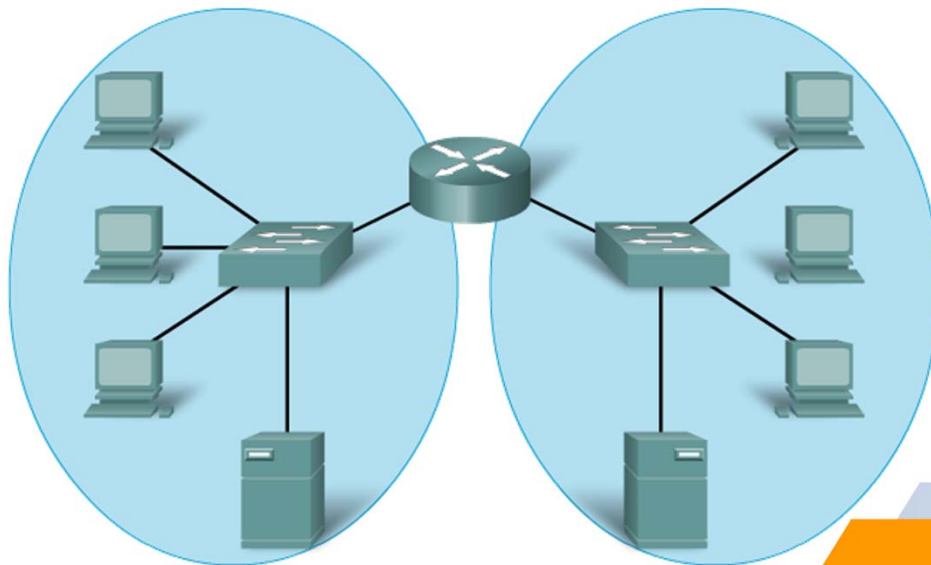
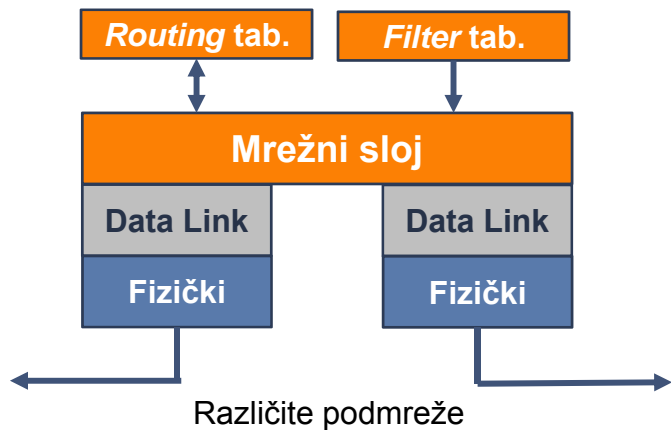
1

Rutiranje

Struktura i funkcija rutera

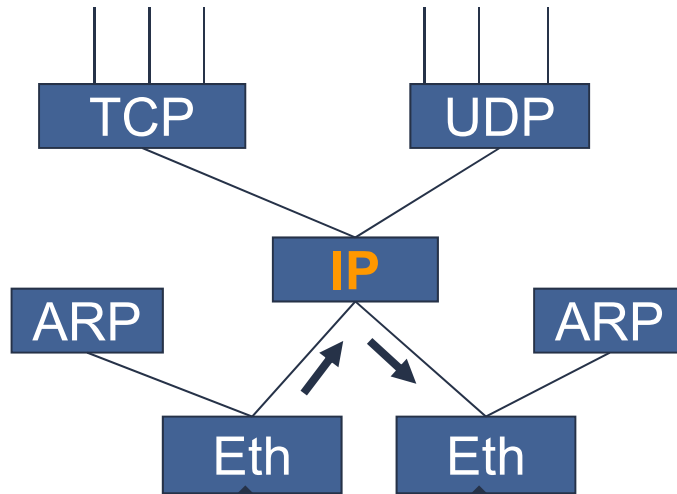
Šta su ruteri?

Ruteri su uređaji koji funkcionišu na **mrežnom** sloju OSI modela i povezuju dve ili više IP mreža, ili povezuju segmente jedne mreže.



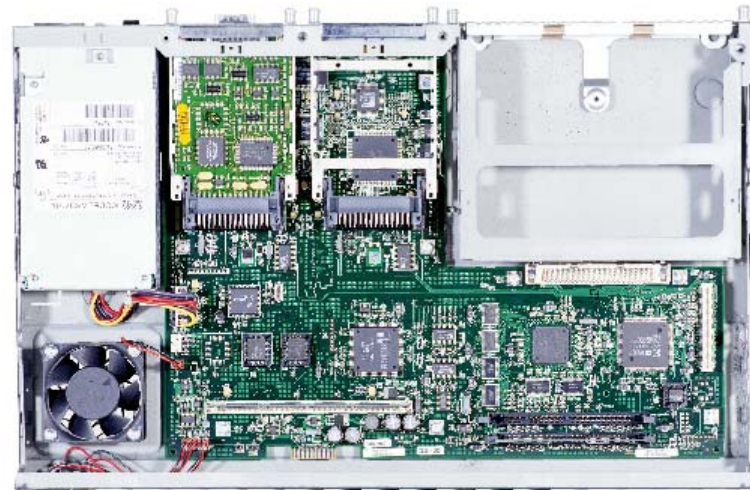
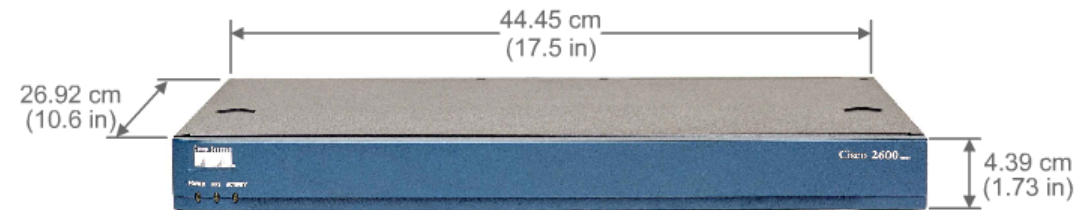
TCP/IP stek rutera

Aplikacije

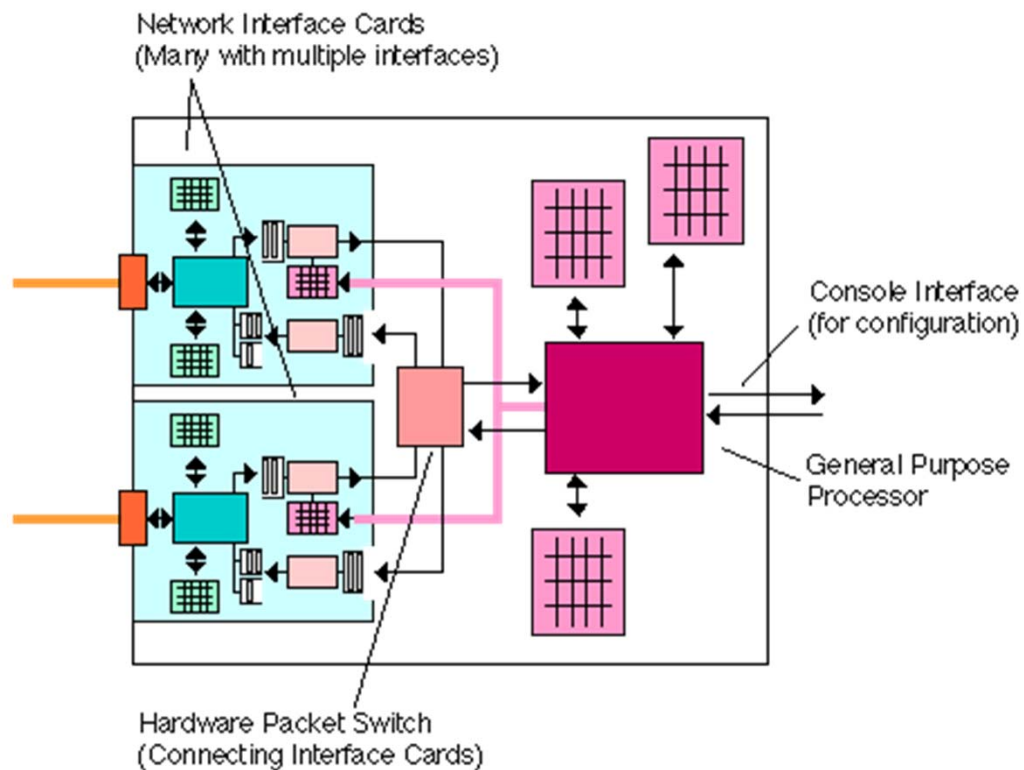


Etheret kabl

Kako izgledaju ruteri?



Blok diagram rutera

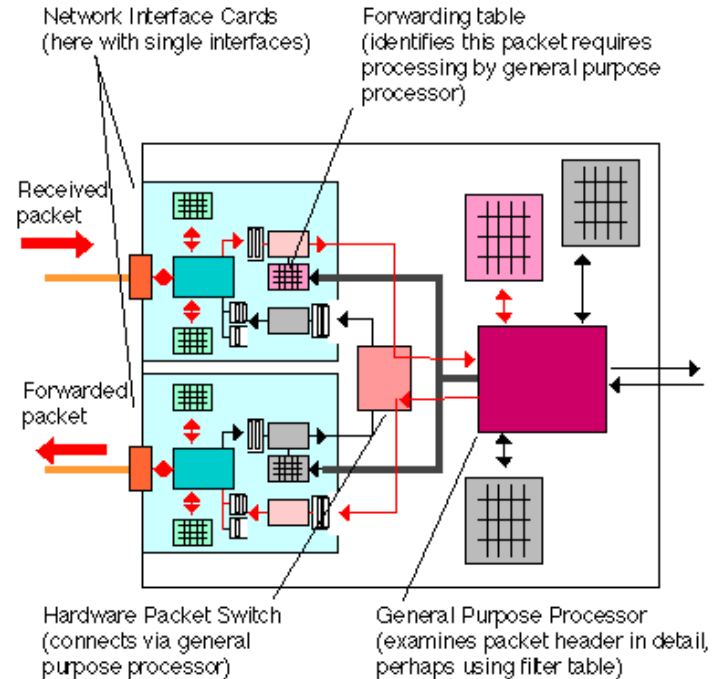


Grupe zadataka koje obvalja ruter

- Process switching
- Fast switching
- Rutiranje
- Filtriranje
- Alokacija propusnog opsega (QoS)

Process switching

- Spor i složen proces
- Omogućuje funkcionalnost koja ne postoji u ostalim režimima (npr. fragmentaciju paketa, primena pravila *firewall*-a, itd.)
- Softverski kontrolisan proces (lako se može unaprediti - *update*)
- U praksi se mnogo manje paketa prosleđuje na ovan način nego korišćenjem drugih metoda



Process switching –1.deo

■ Preuzimanje paketa

- ▷ primljeni paket se proverava
- ▷ paket se smešta u deljivu memoriju
- ▷ poziva se interapt za Switching Processor (SwP)

■ Opsluživanje iterapta

- ▷ SwP zapisuje adresu paketa u memoriji i aktivira proces koji utvrđuje gde paket treba proslediti (*forwarding*)
- ▷ interapt se završava

Process switching - 2 .deo

■ Switching

- ▷ nakon opsluživanja svih interapta *switching* procesor prelazi na obradu paketa
- ▷ ako paket sadrži neku “opciju” one se obrađuju
- ▷ izvlači se odredišna adresa
- ▷ ako je paket namenjen ruteru, prosleđuje se odgovarajućem zadatku
- ▷ ako je odredište drugi računar, pretražuje se *forwarding* tabela da bi se pronašla adresa rutera sledećeg skoka i nalazi se odgovarajući (izlazni) interfejs

Process switching - 3 .deo

■ Switching (nastavak)

- ▷ traži se MAC adresa sledećeg rutera u ARP keš tabeli
- ▷ ako ne postoji, aktivira se ARP a paket smešta u red čekanja
- ▷ proverava da li je veličina paketa dozvoljena na izlaznoj mreži (po potrebi radi fragmentaciju) i smešta podatke u *forwarding cache*
- ▷ dodaje se MAC adresa i referenca na paket smešta u listu paketa za slanje u odgovarajućoj mrežnoj kartici
- ▷ Switching procesor obaveštava odgovarajući procesor mrežne kartice da postoji paket koji čeka u redu

Process switching - 4 .deo

■ Slanje

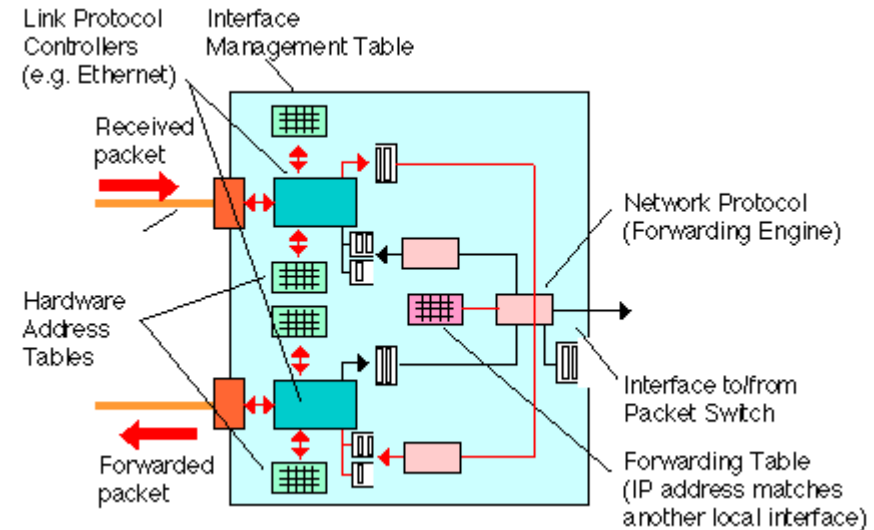
- ▷ kada je proces za slanje mrežne kartice slobodan (svi paketi pre datog u redu su već poslati) paket se šalje
- ▷ procesor mrežne kartice poziva interrupt switching procesora da ga obavesti da je paket poslat

■ Opsluživanje interapta (*switching proc.*)

- ▷ oslobađa se memorija
- ▷ switching procesor ažurira statistiku u *router management information base*.

Fast switching (brzo prosleđivanje)

- Koristi se kada se više paketa prosleđuje sa iste izvorne na istu odredišnu adresu
- Samo prvi paket prosleđuje se *process switching*-om, rezultat se kešira i na osnovu njega se vrši *fast switching*
- Korišćenje keširanih podataka (određena IP adresa, broj porta, MAC adresa, itd.) značajno ubrzava prosleđivanje

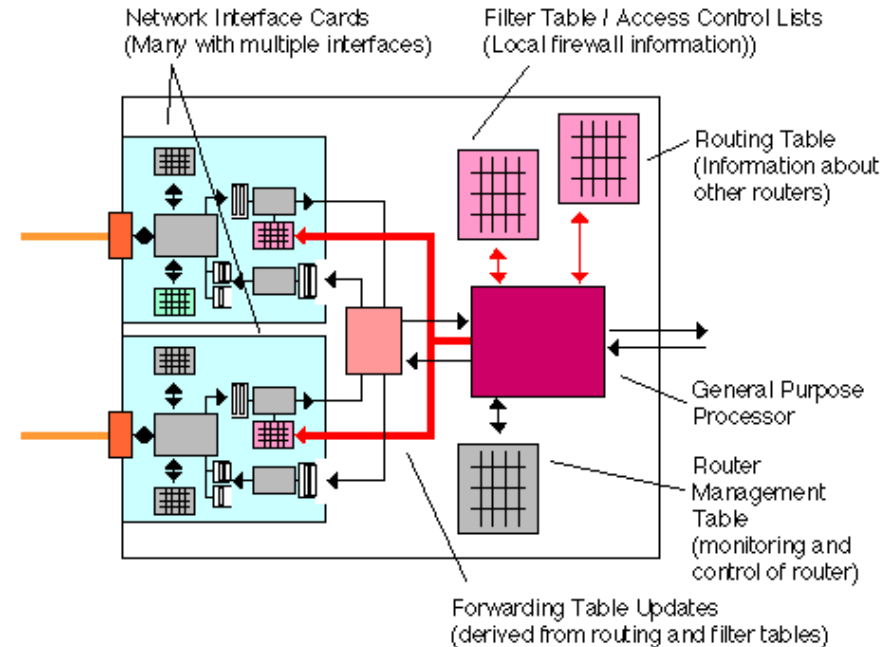


Fast switching

- Preuzimanje paketa (isto kao kod process switching-a)
- Opsluživanje interapta
 - ▷ Switching procesor (SwP) proverava “opcije”
 - ▷ SwP proverava da li u forwarding kešu postoji ulaz koji odgovara određenoj IP adresi (ako ne postoji prelazi se na process switching)
 - ▷ Ako je potrebna fragmentacija prelazi se na process switching
 - ▷ dodaje se MAC adresa i referenca na paket smešta u listu paketa za slanje u odgovarajućoj mrežnoj kartici
 - ▷ Switching procesor obaveštava odgovarajući procesor mrežne kartice da postoji paket koji čeka u redu
 - ▷ Povratak iz interapta
- Slanje (isto kao kod process switching-a)
- Opsluživanje interapta (isto kao kod process switching-a)

Route procesor (centralni procesor)

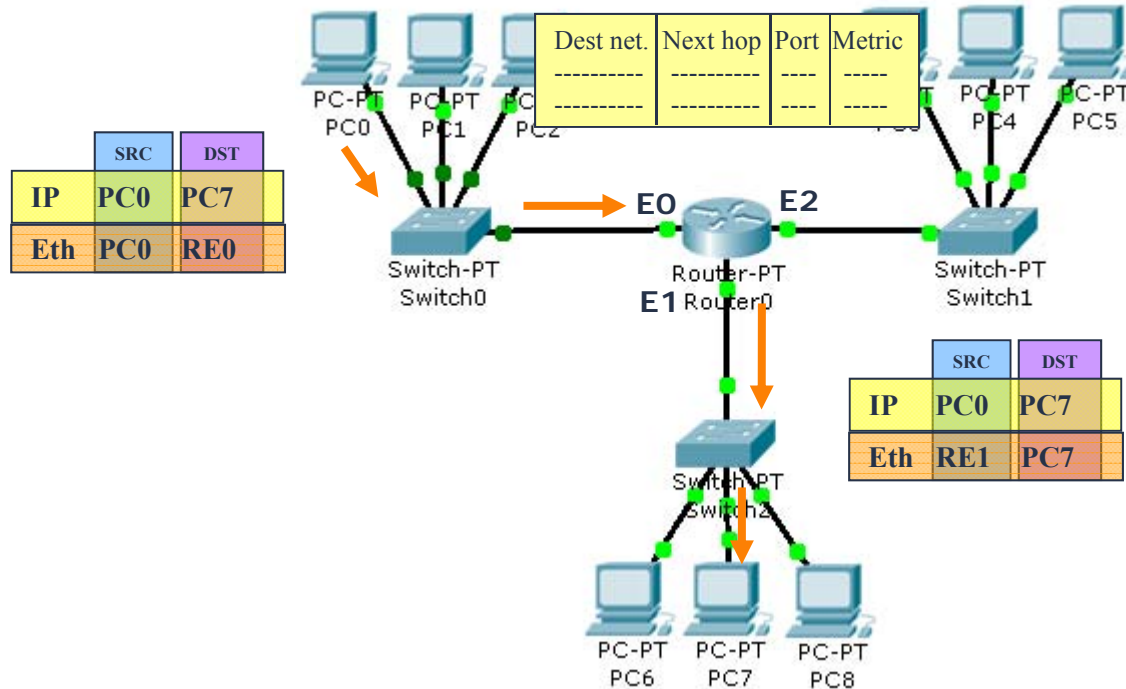
- Konfiguriše mrežne kartice
- Prikuplja podatke potrebne za upravljanje od mrežnih kartica i smešta ih u **Router Management Table**
- Direktno je povezan sa paket *switch*-om
- Obrađuje pakete namenjene samom ruteru
- Upravlja **routing** tabelom, **filter** tabelom i **forwarding** tabelom



Tabele

- **Forwarding tabela** – koristi *switching* proces i za njeno punjenje potrebno je obraditi sledeće dve tabele:
- **Routing tabela** – sadrži liste adresa mreža, rutera i interfejsa (puni je *routing* protokol)
- **Filter tabela** – sadrži pravila za odbacivanje paketa (*firewall*), često se naziva i Access Control List (ACL)

Rutiranje



ARP

Address Resolution Protocol

| IP address | Ethernet address |
|------------|-------------------|
| 223.1.2.1 | 08-00-39-00-2F-C3 |
| 223.1.2.3 | 08-00-5A-21-A7-22 |
| 223.1.2.4 | 08-00-10-99-AC-54 |



Sender IP Address 223.1.2.1
Sender Enet Address 08-00-39-00-2F-C3

Target IP Address 223.1.2.2
Target Enet Address <blank>



| IP address | Ethernet address |
|------------|-------------------|
| 223.1.2.1 | 08-00-39-00-2F-C3 |
| 223.1.2.2 | 08-00-28-00-38-A9 |
| 223.1.2.3 | 08-00-5A-21-A7-22 |
| 223.1.2.4 | 08-00-10-99-AC-54 |

Sender IP Address 223.1.2.2
Sender Enet Address 08-00-28-00-38-A9

Target IP Address 223.1.2.1
Target Enet Address 08-00-39-00-2F-C3

IP adrese

IP adrese

- IP adresa na jedinstveni način identifikuje hosta na mreži
- **IPv4** definiše adresu kao **32**-bitni broj
- **IPv6** definiše adresu kao **128**-bitni broj
- Zbog jednostavnijeg pisanja i pamćenja najčešće se javljaju u tzv. **dotted decimal** notaciji
- Sastoji se od **adrese mreže** i **adrese hosta**. Svi hostovi na istoj mreži moraju imati identičan mrežni deo adrese, i moraju se razlikovati u delu koji definiše adresu hosta.
- Koji bitovi u adresi predstavljaju adresu mreže, a koji adresu hosta definiše – **mrežna maska**

Ekstrakcija adrese mreže

192 . 168 . 1 . 23

11000000.10101000.00000001.00010111

255 . 255 . 255 . 0

11111111.11111111.11111111.00000000

&

192 . 168 . 1 . 0

11000000.10101000.00000001.00000000

Određivanje adrese mreže

Obzirom da se bitovi koji sačinjavaju mrežnu adresu nalaze uvek na početku i da su kontinualni (ne mogu se mešati bitovi koji čine adresu mreže i oni koji čine adresu hosta) maska se može zapisati i u skraćenom obliku, npr. 255.255.255.0 se može napisati kao /24, što znači da prvih 24 bita predstavlja adresu mreže.

Primer

Odrediti adresu mreže i adresu hosta date IP adrese

10.106.248.11/20 \Rightarrow **mreža: 10.106.240.0** **host: 0.0.8.11**

(**00001010 . 01101010 . 11111000 . 00001011**)

Klase IP adresa

- **A** – 1.0.0.0 – 126.255.255.255
- **B** – 128.0.0.0 – 191.255.255.255
- **C** – 192.0.0.0 – 223.255.255.255
- **D** – 224.0.0.0 – 239.255.255.255

| Klasa | Početni bitovi | Opseg adresa | Mreža | Host |
|-------------|----------------|-----------------------------|-------|------|
| A | 0 | 1.0.0.0 - 126.255.255.255 | 7b | 24b |
| B | 10 | 128.0.0.0 - 191.255.255.255 | 14b | 16b |
| C | 110 | 192.0.0.0 - 223.255.255.255 | 21b | 8b |
| D/Multicast | 1110 | 224.0.0.0 - 239.255.255.255 | - | - |

Privatne adrese

- 10.0.0.0 – 10.255.255.255
- 172.16.0.0 – 172.31.255.255
- 192.168.0.0 – 192.168.255.255

Povratna adresa

- 127.0.0.x

Podmreža

- Povećavaju iskorišćenost IP adresa
- Olakšavaju administraciju
- Smanjuju saobraćaj
 - ▷ smanjuju kolizione domene
 - ▷ smanjuju *broadcast* domene
- Odvajaju se ruterima


Kako se dobijaju adrese podmreža?

- Određeni broj bitova adrese hosta pridodaje se adresi mreže

| Mrežna adresa klase B – 147.10.0.0 | | | | | | |
|------------------------------------|---|----------|---|----------|---|----------|
| 10010011 | . | 00001010 | . | 00000000 | . | 00000000 |
| N | . | N | . | H | . | H |
| 10010011 | . | 00001010 | . | 00000000 | . | 00000000 |
| N | . | N | . | sN | . | sN H |

Pozajmljivanjem 12 bitova, umesto 65534 hostova u okviru jedne mreže klase B, dobijamo 4096 podmreža sa po 14 hostova.

Kako se računa broj podmreža?

- Ukupan broj podmreža: **2 broj pozajmljenih bitova**
- Ukupan broj hostova: **2 preostali broj bitova**
- Broj korisnih podmreža: **2 broj pozajmljenih bitova - 2** 
- Broj korisnih hostova: **2 preostali broj bitova - 2**
- Adresa koja ima sve nule na poziciji bitova hosta nije validna adresa jer se radi o **adresi mreže** i ne može se dodeliti uređaju (npr. 128.11.0.0)
- Adresa koja ima sve jedinice na poziciji bitova hosta nije validna adresa jer se radi o **broadcast** adresi i ne može se dodeliti uređaju (npr. 128.11.255.255)
- Jedna od validnih adresa hostova mora se dodeliti interfejsu rutera, na koji je povezana data mreža, tako da je broj računara koji se može povezati zapravo **2 broj bitova hosta - 3**

Primeri adresa

Neka je zadata sledeća mreža: 172.12.16.0/20

- Adresa **mreže**: 172.12.16.0/20

- ▷ 10101100.00001100.0001**0000.00000000**

- Adresa **rutera**: 172.12.16.1/20

- 10101100.00001100.0001**0000.00000001**

- Adrese **hostova**: 172.12.16.2/20 – 172.12.31.254/20

- ▷ 10101100.00001100.0001**0000.00000010** – 0001**1111.11111110**

- **Broadcast** adresa: 172.12.31.255/20

- ▷ 10101100.00001100.0001**1111.11111111**

Nulta pod mreža i pod mreža “sve jedinice”

- Nekada je bilo zabranjeno korišćenje ovih pod mreža
- Nulta pod mreža poklapa se sa klasnom nad mrežom. (Kako razlikovati npr. 132.11.0.0/16 i 132.11.0.0/20)
- *Broadcast* adresa poslednje pod mreže poklapa se sa *broadcast* adresom nad mreže. To može da izazove kumulativnu razmenu paketa do isteka TTL-a.
- Ovi problemi postojali su kod *classfull routing* protokola, jer ne prenose masku uz *update*, pa su nulta i pod mreža “sve jedinice” bile izbegavane.
- Danas to ne bi trebalo da bude problem (osim ako se ne koristi RIPv1).

Ako je:

- mržna adresa: **59.0.0.0**
- potreban broj podmreža: **3103**
- svaka podmreža mora da ima bar **10** hostova
- Odrediti kako izgleda *subnet mask*!

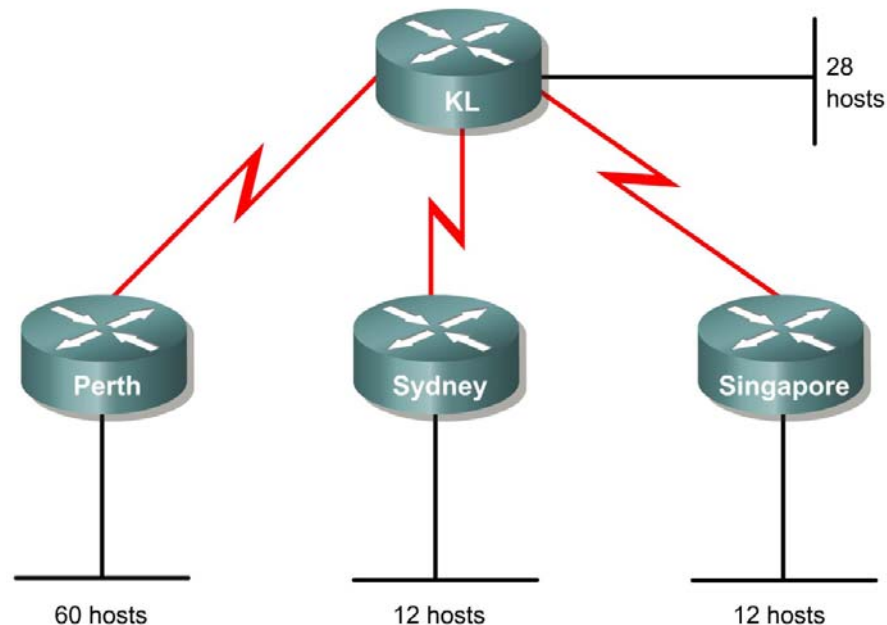
- Prvi broj koji je stepen br. 2 i veći je od 3103 je: **4096**
- Br. bitova za predstavljanje 4096 vrednosti je: **12** ($2^{12} = 4096$)
- Dakle, pozajmljujemo 12 bitova od adrese hosta
- Obzirom da je adresa klase A, i da je bilo 24b za adrese hostova, ostaje 12b, pa imamo: $2^{12} - 2 = 4094$ hostova po mreži
- Subnet mask dobijamo kada postavimo 1 na svim bitovima dodeljenim adresi mreži, a 0 bitovima adrese hosta, tj.
- 11111111.11111111.11110000.0000, odnosno dekadno
- **255.255.240.0**

Validne vrednosti za bajtove u maskama

- **0** – 0 0 0 0 0 0 0 0
- **128** – 1 0 0 0 0 0 0 0
- **192** – 1 1 0 0 0 0 0 0 (128+64)
- **224** – 1 1 1 0 0 0 0 0 (192+32)
- **240** – 1 1 1 1 0 0 0 0 (224+16)
- **248** – 1 1 1 1 1 0 0 0 (240+8) ili (255-7)
- **252** – 1 1 1 1 1 1 0 0 (255-3)
- **254** – 1 1 1 1 1 1 1 0 (255-1)
- **255** – 1 1 1 1 1 1 1 1

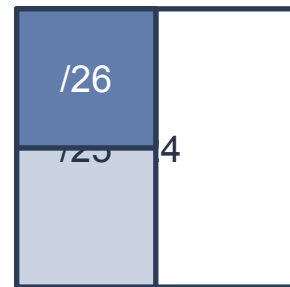
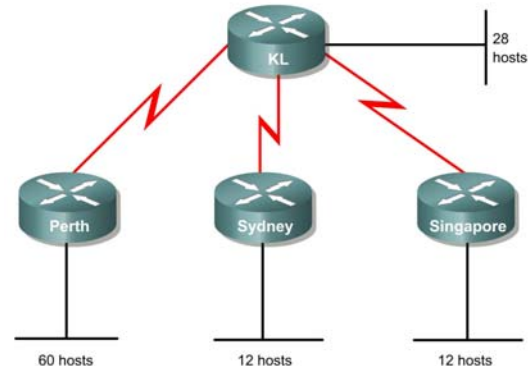
Zadatak

Za mrežu prikazanu na slici, odrediti adresnu šemu, tako da iskorišćenost adresa bude što veća, ako je na raspolaganju sledeća adresa klase C: 192.168.10.0/24



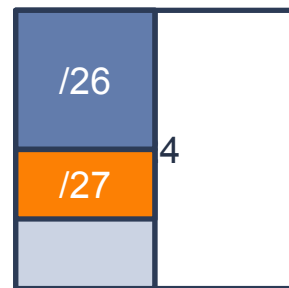
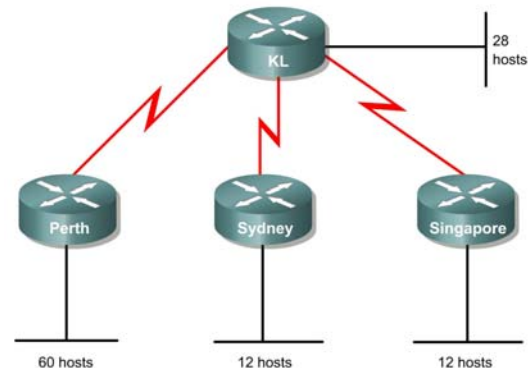
Rešenje

- Pronalazi se podmreža sa najviše hostova i za nju se alocira prostor
- **Perth** zahteva **60** hostova
- **60** (za hostove) + **2** (nevalidne, mreža + broadcast) + **1** (ruter) = **63**,
- Prvi sledeći stepen dvojke \geq od 63 je **64**, tj. **2^6**
- Dakle, **6** bitva je potrebno za adresu hosta, a sve ostalo može biti adresa mreže
- Pa je potrebno kreirati podmrežu sa prefiksom **/26**, jer je adresa 32-bitna ($32 - 6 = 26$)
- **Pert:** **192.168.10.0/26**
 - ▷ adr. rutera: 192.168.10.1
 - ▷ adr. hostova: 192.168.10.2 – 192.168.10.62
 - ▷ broadcast adr.: 192.168.10.63
- Ovim smo zauzeli četvrtinu raspoloživog adresnog prostora.



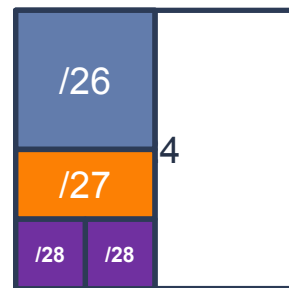
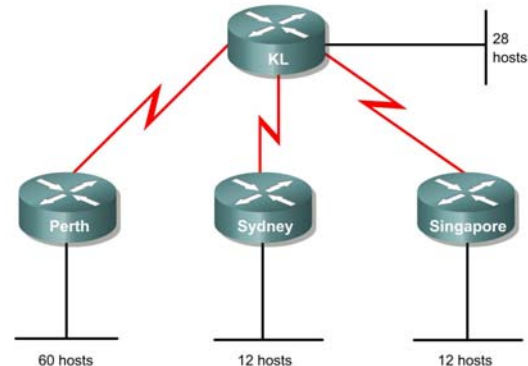
Rešenje

- Sledeća po veličini je **KL** sa **28** hostova.
- Za 28 hostova ($28+3=31 \leq 32 = 2^5$) potrebno nam je **5** bitova u delu za adr. hosta
- Formiramo podmrežu **/27**
- **KL: 192.168.10.64/27**
 - ▷ adr. rutera: 192.168.10.65
 - ▷ adr. hostova: 192.168.10.65 – 192.168.10.94
 - ▷ broadcast adr.: 192.168.10.95



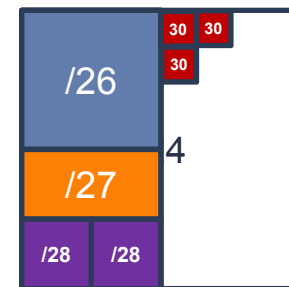
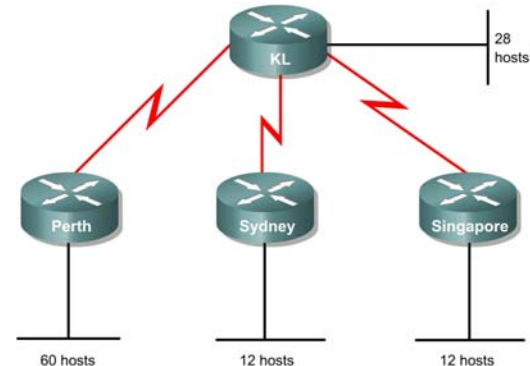
Rešenje

- **Sydney i Singapore** imaju po **12** hostova
- $12 + 3 = 15 \leq 16 = 2^4 \Rightarrow /28$
- **Sydney: 192.168.10.96/28**
 - ▷ adr. rutera: 192.168.10.97
 - ▷ adr. hostova: 192.168.10.98 – 192.168.10.110
 - ▷ broadcast adr.: 192.168.10.111
- **Singapore: 192.168.10.112/28**
 - ▷ adr. rutera: 192.168.10.113
 - ▷ adr. hostova 192.168.10.114 – 192.168.10.126
 - ▷ broadcast adr.: 192.168.10.127

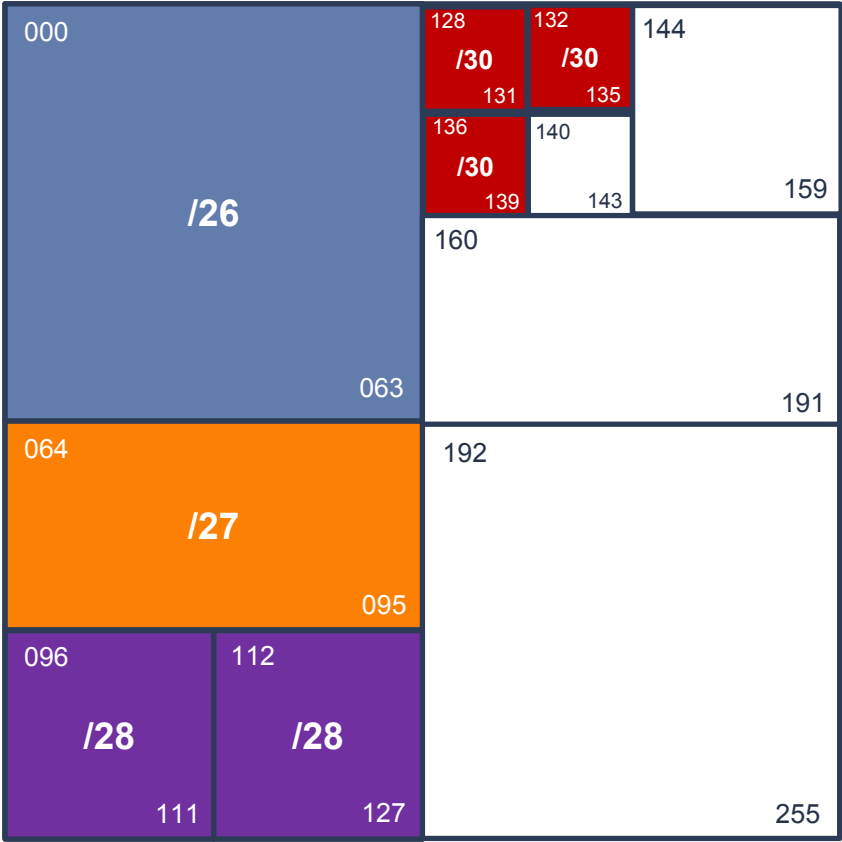
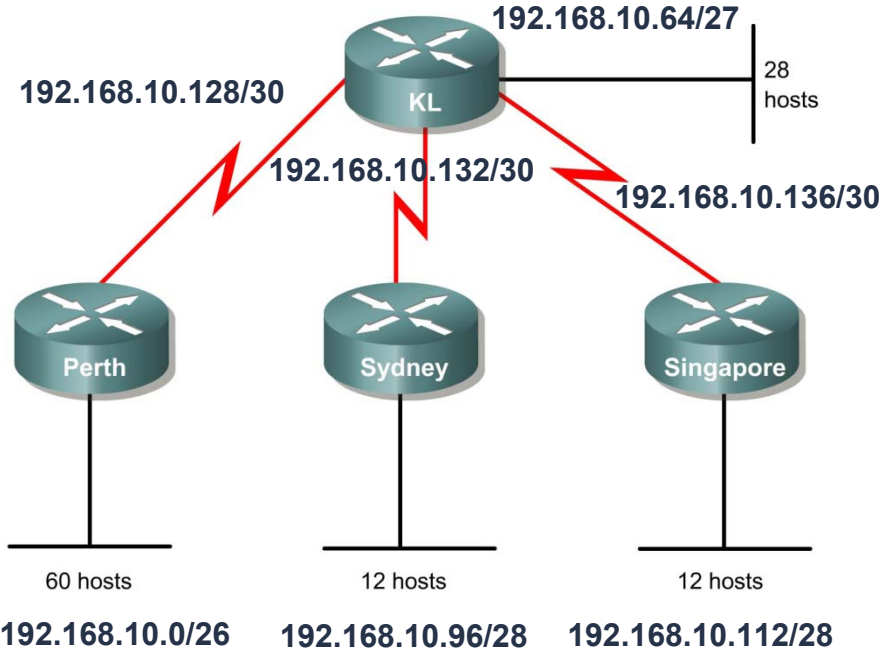


Rešenje

- Ostaju još 3 point-to-point veze (mreže) sa prefiksom **/30** ($2+2=4 \leq 4 = 2^2 \Rightarrow$ **/30**)
 - ▷ **192.168.10.128/30**,
 - ▷ **192.168.10.132/30** i
 - ▷ **192.168.10.136/30**



Rešenje

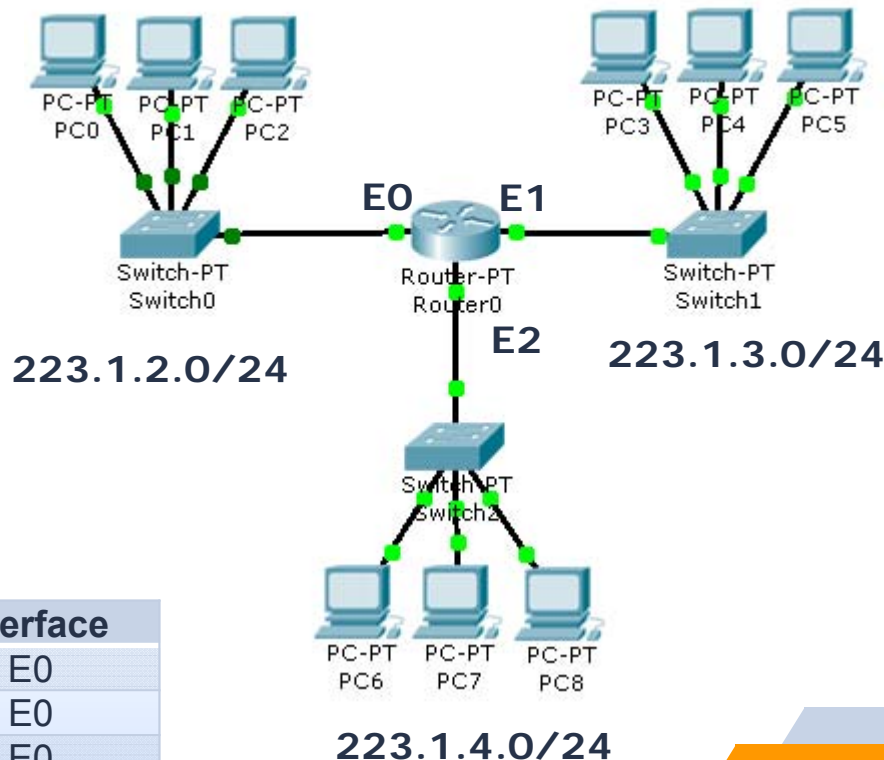


Routing tablele

U svakom čvoru nalazi se *routing* tabela na osnovu koje se određuje gde se preusmerava saobraćaj

| Network | Mask | D/I | Next | Interface |
|-----------|---------------|-----|------|-----------|
| 223.1.2.0 | 255.255.255.0 | D | < > | E0 |
| 223.1.3.0 | 255.255.255.0 | D | < > | E1 |
| 223.1.4.0 | 255.255.255.0 | D | < > | E2 |

Router

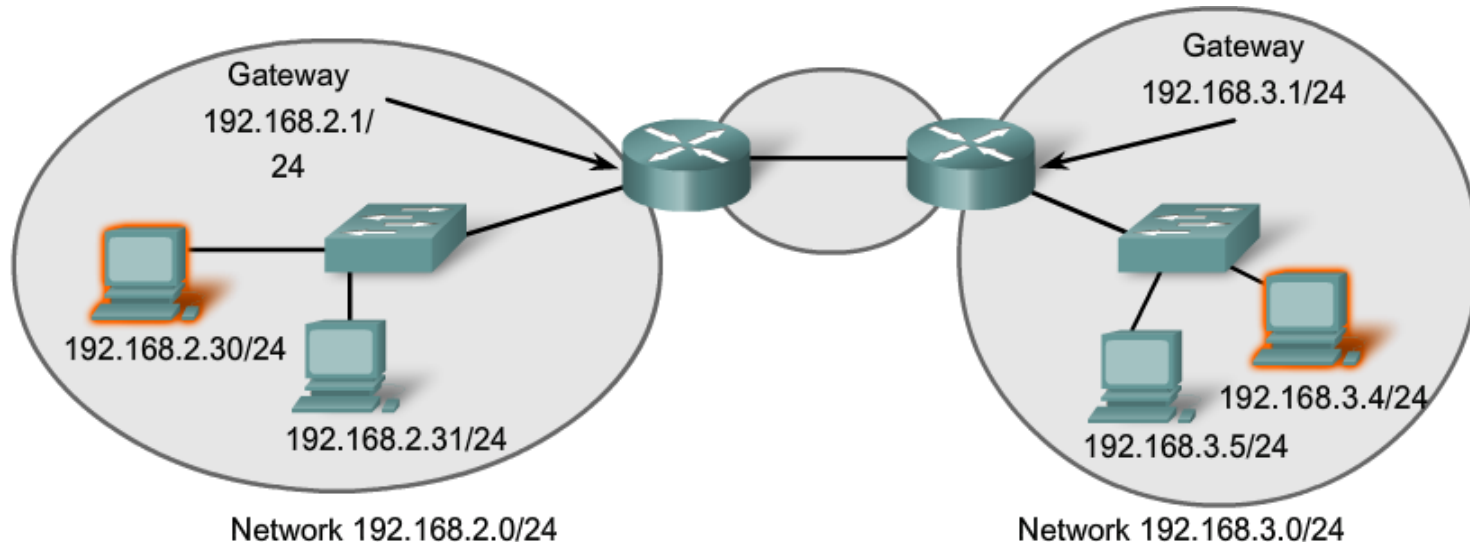


PC0

| Network | Mask | D/I | Next | Interface |
|-----------|---------------|-----|-----------|-----------|
| 223.1.2.0 | 255.255.255.0 | D | < > | E0 |
| 223.1.3.0 | 255.255.255.0 | I | 223.1.2.1 | E0 |
| 223.1.4.0 | 255.255.255.0 | I | 223.1.2.1 | E0 |

Default Gateway

Da bi se pojednostavilo prosleđivanje paketa računarima koji nisu u lokalnoj mreži, na *host*-ovima se definišu adrese *default gateway*-a. To su adrese interfejsa rutera povezanog na istu lokalnu mrežu kao i *host* koga konfiguriramo.

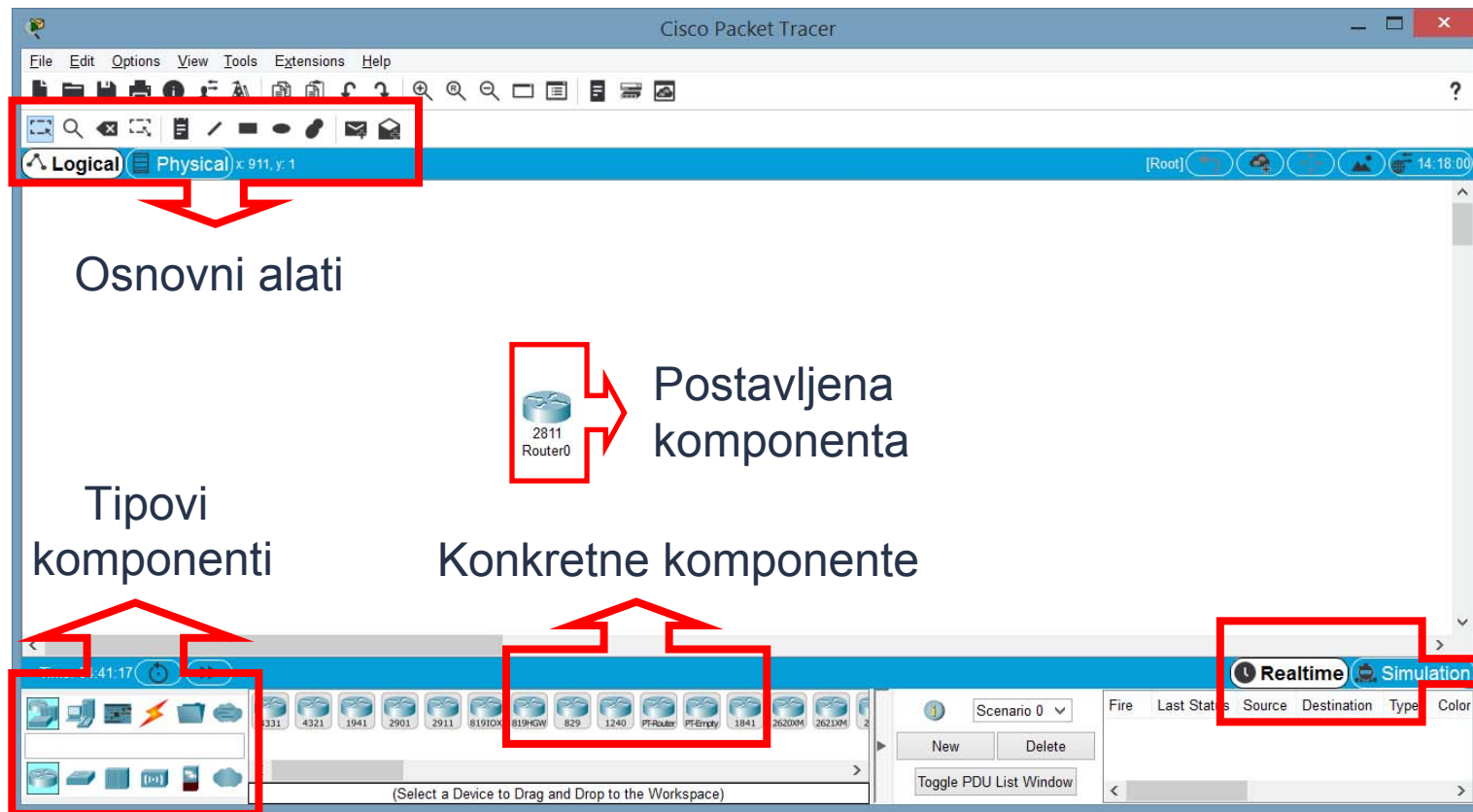


Neki korisni mrežni servisi

- Ipconfig
- Arp
- Ping
- Tracert

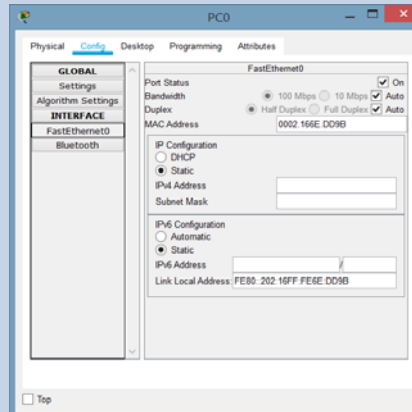
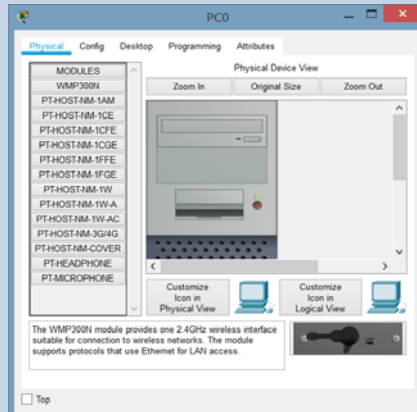
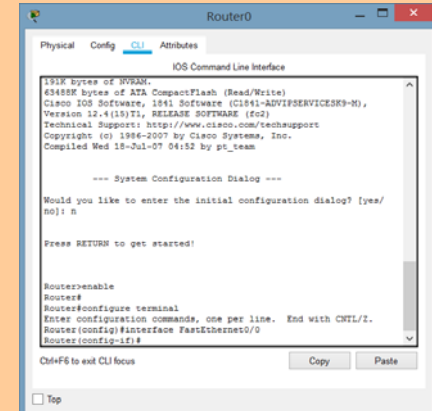
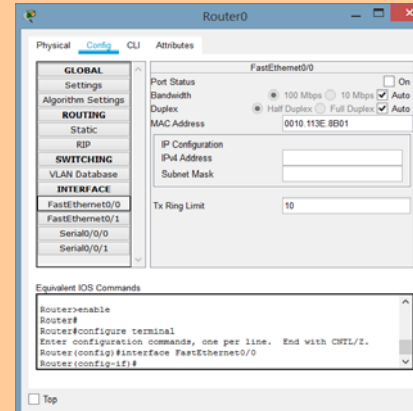
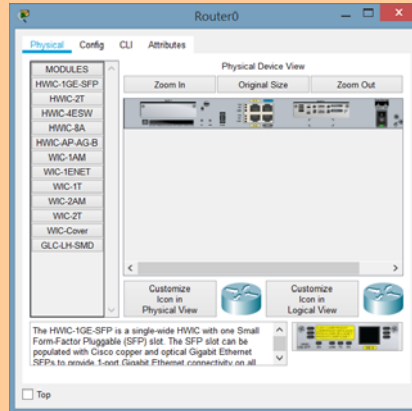
Packet Tracer

Packet Tracer



Realtime
ili
simulacioni
mod

Packet Tracer



Packet Tracer

Simulation Panel

Event List

| Vis. | Time(sec) | Last Device | At Device | Type |
|------|-----------|-------------|-----------|------|
| | 0.000 | -- | PC1 | ICMP |
| | 0.001 | PC1 | Switch3 | ICMP |

Reset Simulation

☒ Constant Delay

Captured to: 0.001 s

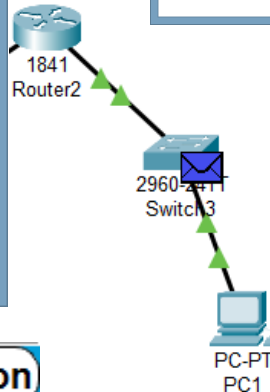
Play Controls

Event List Filters - Visible Events

ICMP

Edit Filters

Show All/None



PDU Information at Device: Switch3

OSI Model

Inbound PDU Details

Outbound PDU Details

At Device: Switch3
Source: PC1
Destination: 192.168.4.1

In Layers

Layer7

Layer6

Layer5

Layer4

Layer3

Layer 2: Ethernet II Header 0001.438E.6237 >> 0005.5E17.EC02

Layer 1: Port FastEthernet0/1

Out Layers

Layer7

Layer6

Layer5

Layer4

Layer3

Layer 2: Ethernet II Header 0001.438E.6237 >> 0005.5E17.EC02

Layer 1: Port(s): FastEthernet0/23

1. FastEthernet0/23 sends out the frame.

Challenge Me

PDU Information at Device: Switch3

OSI Model

Inbound PDU Details

Outbound PDU Details

PDU Formats

EthernetII

0 4 8 16 20 24 Bytes

PREAMBLE: 101010..10

SF D

DEST ADDR: 0005.5E17.EC02

SRC ADDR: 0001.438E.6237

TYPE: 0x0800

DATA (VARIABLE LENGTH)

FCS: 0x00000000

IP

0 4 8 16 20 24 Bits

VER: 4

IHL: 5

DSCP: 0x00

TL: 128

ID: 0x001a

FLAGS: 0x0

FRAG OFFSET: 0x000

TTL: 128

PRO: 0x01

CHKSUM

SRC IP: 192.168.4.2

DST IP: 192.168.4.1

DATA (VARIABLE LENGTH)

ICMP