

Lucrarea de laborator №5

Rutare statică și protocoale de rutare dinamică

Scopul lucrării constă în formarea abilităților practice de configurare statică și dinamică în Cisco Packet

Tracer a tabelelor de rutare ale routerelor din componența rețelelor

Obiective:

- Prezentarea conceptului de rutare și de tabel de rutare pe router
- Realizarea rutării statice pe routerele unei rețele
- Configurarea protocoalelor de rutare dinamică RIP, EIGRP și OSPF
- Configurarea protocolului OSPF pe mai multe domenii
- Prezentarea conceptului de redistribuire a rutelor între sistemele autonome
- Configurarea protocolului de rutare dinamică BGP între sistemele autonome

Indicații metodice privind realizarea lucrării

Prin rutare se înțelege procesul prin care routerul determină automat în baza tabelului său de rutare traseul pe care trebuie transmis un pachet de date recepționat pentru a ajunge la destinație.

Tabelul de rutare include anumite rute până la subrețelele rețelei inițiale. Fiecare rută (pe lângă IP-ul și masca rețelei destinație) definește interfața routerului prin care se intră în rețeaua destinație.

Inițial, la conectarea routerului în rețea, în tabelul de rutare al acestuia sunt scrise adresele subrețelor direct conectate la acesta. Pentru a avea posibilitate de transmis date către celelalte rețele, în tabelul de rutare este necesar să se definească rute corespunzătoare până la acestea. Definirea rutelor se poate face manual sau în mod automatizat. Procedul prin care se definesc manual rutele pe fiecare router al rețelei este numit rutare statică, iar procedul automatizat este realizat printr-un protocol de rutare dinamică.

Rutarea statică

Vom ilustra mai întâi cum se configurează manual o rută statică (numită recursivă) și rutele statice implicite (care specifică IP-ul interfeței routerului către care se va transmite pachetul de date, atunci când tabelul de rutare al routerului curent nu include un traseu până la rețeaua destinație).

Sintaxa aplicată la definirea unei rute statice recursive este următoarea:

```
Router(config)# ip route network-address subnet-mask ip-address
```

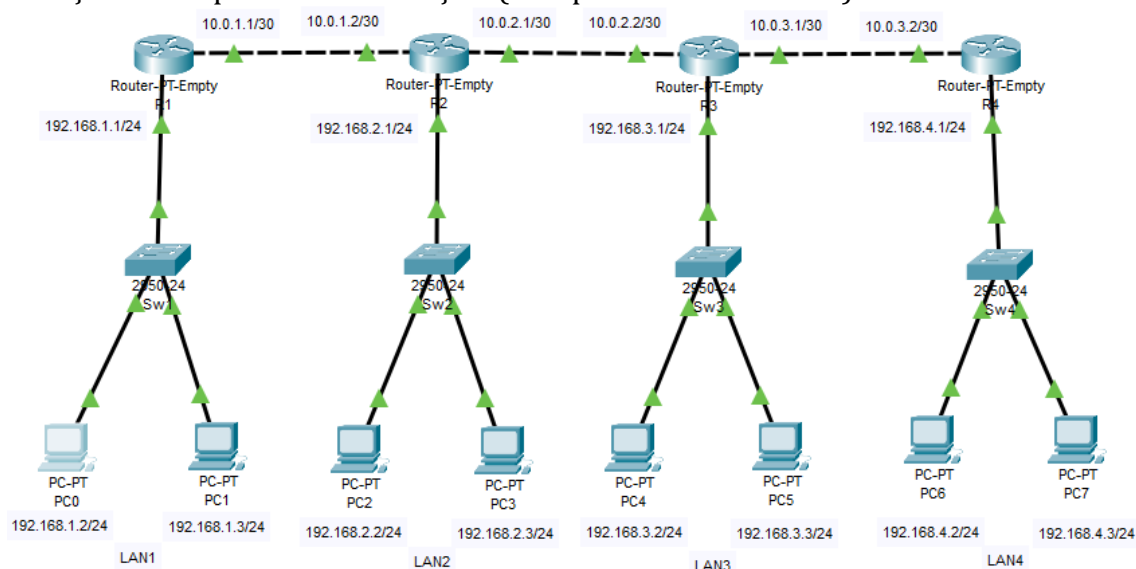
unde network-address și subnet-mask sunt adresa și masca subrețelei destinație, iar ip-address este IP adresa interfeței de intrare în următorul router în calea către subrețeaua destinație.

Sintaxa aplicată la definirea unei rute statice implicite este următoarea:

```
Router(config)# ip route 0.0.0.0 0.0.0.0 {ip-address}
```

unde 0.0.0.0 0.0.0.0 este IP-ul și masca destinație, iar ip-address – adresa IP a interfeței de intrare în următorul router în calea către subrețeaua destinație.

Să examinăm rețeaua compusă din 4 subrețele (în raport cu switch-urile)



Dacă dăm un ping de pe PC0 către PC6, routerul R1 ne va anunța că „Destination host unreachable”

```
C:\>ping 192.168.4.2

Pinging 192.168.4.2 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.4.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Routerul R0 nu știe cum să transmită pachetul până la subrețeaua LAN4 => trebuie să-i spunem acestui router cum să ajungă la subrețeaua LAN4, adică să-i indicăm ruta!

În acest sens, pentru a accede în subrețeaua LAN4 urmează să se transmită pachetul către routerul R2, iar lui R2 îi „vom spune” că pentru a ajunge în LAN4 urmează să transmită pachetul lui R3.

La fel lui R3 îi „vom spune” că pentru a ajunge în LAN4 urmează să transmită pachetul lui R4, iar acesta din urmă deja „știe ce trebuie să facă” în subrețeaua (domeniu broadcast) ce este conectată direct la el.

În Cisco Packet Tracer vom utiliza comanda *ip route*

| |
|--|
| Pentru routerul R1: |
| Router>en Router#conf ter Router(config)#ip route 192.168.4.0 255.255.255.0 10.0.1.2 |
| Pentru routerul R2: |
| Router>en Router#conf ter Router(config)#ip route 192.168.4.0 255.255.255.0 10.0.2.2 |
| Pentru routerul R3: |
| Router>en Router#conf ter Router(config)#ip route 192.168.4.0 255.255.255.0 10.0.3.2 |

Totuși, dacă dăm din nou un ping de pe PC0 către PC6, vom fi refuzați. De ce?

În modul de simulare vedem că pachetul ajunge până la destinație, dar când se întoarce – se blochează la routerul R4. Aceasta înseamnă că trebuie de indicat și calea înapoi, de la destinație la sursă:

| |
|--|
| Pentru routerul R4: |
| Router>en Router#conf ter Router(config)#ip route 192.168.1.0 255.255.255.0 10.0.3.1 |
| Pentru routerul R3: |
| Router#conf ter Router(config)#ip route 192.168.1.0 255.255.255.0 10.0.2.1 |
| Pentru routerul R2: |
| Router>en Router#conf ter Router(config)#ip route 192.168.1.0 255.255.255.0 10.0.1.1 |

În acest moment, ping-ul către PC6 trece cu succes:

```
C:\>ping 192.168.4.2

Pinging 192.168.4.2 with 32 bytes of data:

Reply from 192.168.4.2: bytes=32 time<1ms TTL=124
Reply from 192.168.4.2: bytes=32 time<1ms TTL=124
Reply from 192.168.4.2: bytes=32 time=6ms TTL=124
Reply from 192.168.4.2: bytes=32 time=1ms TTL=124

Ping statistics for 192.168.4.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 6ms, Average = 1ms
```

Am scris ruta de la prima la a patra subrețea

Dar analog trebuie de scris rutele între oricare două subrețele

Mai trebuie de completat rutele dintre subrețelele: de la 1 la 2, de la 1 la 3, de la 2 la 3, de la 3 la 2, de la 4 la 3, de la 4 la 2

Astfel, pentru a asigura accesul la aceste subrețele, am fost nevoiți să scriem rutele indicate

Dacă se închide accesul la istoricul comenzilor introduse în linia de comandă a routerului, atunci pot fi

scrise următoarele două comenzi pentru a evita o astfel de situație:

```
Router(config)#line console 0
Router(config-line)#exec-timeout ?
<0-35791> Timeout in minutes
Router(config-line)#exec-timeout 0 0
```

Cum putem vizualiza rutele pe care le-am scris? În tabelul de rutare! Aceasta din urmă poate fi vizualizată pe fiecare router (tabelul de rutare al routerului respectiv!), folosind comanda *show ip route* din modul privilegiat sau comanda *do show ip route* din modul de configurare globală:

De exemplu, pe routerul R2 avem:

```
Router>en
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/30 is subnetted, 2 subnets
C 10.0.1.0 is directly connected, FastEthernet9/0
C 10.0.2.0 is directly connected, FastEthernet8/0
S 192.168.1.0/24 [1/0] via 10.0.1.1
C 192.168.2.0/24 is directly connected, FastEthernet7/0
S 192.168.4.0/24 [1/0] via 10.0.2.2

Router#conf ter
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#do show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/30 is subnetted, 2 subnets
C 10.0.1.0 is directly connected, FastEthernet9/0
C 10.0.2.0 is directly connected, FastEthernet8/0
S 192.168.1.0/24 [1/0] via 10.0.1.1
C 192.168.2.0/24 is directly connected, FastEthernet7/0
S 192.168.4.0/24 [1/0] via 10.0.2.2
```

Rutele notate cu S sunt cele statice, adică cele setate manual pe routerul respectiv

Rutele notate cu C sunt cele conectate direct la routerul curent

Analog pot fi vizualizate rutele și pe celelalte routere

Ceea ce am realizat acum se numește rutare statică!

Dar în rețelele organizațiilor cu mai mult de 5 routere se folosește rutarea dinamică!

Rutarea dinamică

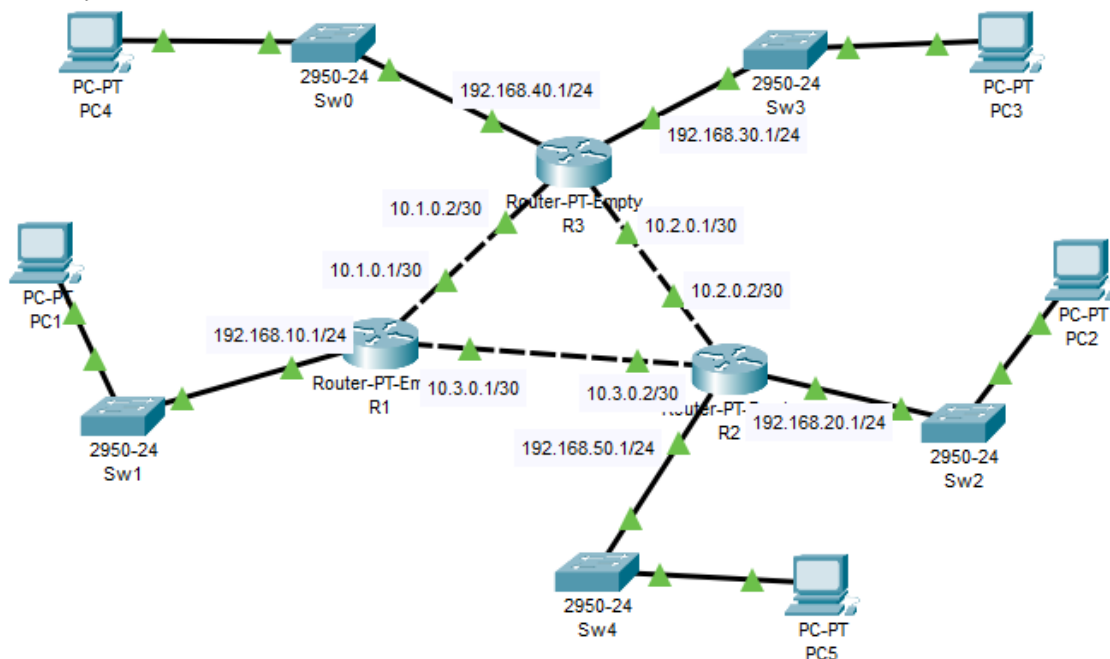
Protocoalele de rutare dinamică sunt de trei tipuri:

- Protocoale bazate pe vectori distanță
Se calculează în mod iterativ costurile rutelor în condițiile în care nu se dispune de informație completă despre rețea. Tabelul de rutare se construiește iterativ în câteva etape
Vom examina două protocoale de acest tip: RIP (Routing Information Protocol) și EIGRP
- Protocoale bazate pe analiza stării link-ului
Routerele mai întâi examinează în întregime rețeaua, după care, după ce este disponibilă informație completă despre rețea, sunt calculate costurile rutelor
Vom examina un protocol de acest tip - OSPF

- Protocoale de rutare ierarhică (numite și protocoale bazate pe distanța dintre sistemele autonome)
– vom examina un protocol de acest tip - BGP

Configurarea protocolului RIP cu Cisco Packet Tracer

Să considerăm rețeaua următoare:



Vom configura pe routerule R1, R2 și R3 ale acesteia protocolul de rutare dinamică RIP:

```
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#network 10.1.0.0
R1(config-router)#network 192.168.10.0
R1(config-router)#network 10.3.0.0
R1(config-router)#no auto-summary
R1(config-router)#exit
R1(config)#do wr
```

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#network 192.168.20.0
R2(config-router)#network 192.168.50.0
R2(config-router)#network 10.3.0.0
R2(config-router)#network 10.2.0.0
R2(config-router)#no auto-summary
R2(config-router)#exit
R2(config)#do wr
```

```
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#network 10.1.0.0
R3(config-router)#network 10.2.0.0
R3(config-router)#network 192.168.30.0
R3(config-router)#network 192.168.40.0
R3(config-router)#no auto-summary
R3(config-router)#exit
R3(config)#do wr
```

În rezultat, pe fiecare router au fost stabilite rutele RIP până la toate subrețelele rețelei inițiale. De exemplu, în tabelul de rutare a lui R1 avem:

```
10.0.0.0/30 is subnetted, 3 subnets
C    10.1.0.0 is directly connected, GigabitEthernet9/0
R    10.2.0.0 [120/1] via 10.3.0.2, 00:00:03, GigabitEthernet8/0
    [120/1] via 10.1.0.2, 00:00:28, GigabitEthernet9/0
C    10.3.0.0 is directly connected, GigabitEthernet8/0
C    192.168.10.0/24 is directly connected, GigabitEthernet7/0
R    192.168.20.0/24 [120/1] via 10.3.0.2, 00:00:03, GigabitEthernet8/0
R    192.168.30.0/24 [120/1] via 10.1.0.2, 00:00:28, GigabitEthernet9/0
R    192.168.40.0/24 [120/1] via 10.1.0.2, 00:00:28, GigabitEthernet9/0
```

```
R 192.168.50.0/24 [120/1] via 10.3.0.2, 00:00:03, GigabitEthernet8/0
```

Dacă se dă ping între oricare două host-uri din rețea, atunci ne putem convinge că avem conexiune între acestea. La fel, cu comanda *tracert* putem vizualiza ruta până la destinație. De exemplu, avem:

```
C:\>tracert 192.168.20.10
Tracing route to 192.168.20.10 over a maximum of 30 hops:
  1    1 ms         11 ms         0 ms         192.168.10.1
  2    0 ms         0 ms         4 ms         10.3.0.2
  3    0 ms         0 ms         0 ms         192.168.20.10
Trace complete.
```

Agregarea rutei reduce cantitatea de informații din tabelele de rutare. Dacă utilizați RIP versiunea 2, puteți dezactiva agregarea automată, specificând comanda *no auto-summary* în modul de configurare a routerului. Dezactivați agregarea automată dacă trebuie să efectuați rutare între subrețele care nu sunt conectate direct. Dacă agregarea automată este dezactivată, atunci sunt arătate și subrețelele.

Configurarea protocolului EIGRP cu Cisco Packet Tracer

Pe exemplul aceleiași rețele vom arăta cum se configurează protocolul EIGRP:

```
R1(config)#router eigrp 1
R1(config-router)#network 10.1.0.0 0.0.0.3
R1(config-router)#network 192.168.10.0 0.0.0.255
R1(config-router)#network 10.3.0.0 0.0.0.3
R1(config-router)#no auto-summary
R1(config-router)#exit
R1(config)#do wr
```

```
R2(config)#router eigrp 1
R2(config-router)#network 10.2.0.0 0.0.0.3
R2(config-router)#network 10.3.0.0 0.0.0.3
R2(config-router)#network 192.168.50.0 0.0.0.255
R2(config-router)#network 192.168.20.0 0.0.0.255
R2(config-router)#no auto-summary
R2(config-router)#exit
R2(config)#do wr
```

```
R3(config)#router eigrp 1
R3(config-router)#network 10.1.0.0 0.0.0.3
R3(config-router)#network 10.2.0.0 0.0.0.3
R3(config-router)#network 192.168.30.0 0.0.0.255
R3(config-router)#network 192.168.40.0 0.0.0.255
R3(config-router)#no auto-summary
R3(config-router)#exit
R3(config)#do wr
```

Pe lângă adresele rețelelor direct conectate la router, sunt indicate și măștile inverse ale acestora.

Configurarea protocolului OSPF cu Cisco Packet Tracer

Pe exemplul aceleiași rețele vom arăta cum se configurează protocolul OSPF:

```
R1(config)#router ospf 1
R1(config-router)#network 10.1.0.0 0.0.0.3 area 0
R1(config-router)#network 10.3.0.0 0.0.0.3 area 0
R1(config-router)#network 192.168.10.0 0.0.0.255 area 0
R1(config-router)#exit
R1(config)#do wr
```

```
R2(config)#router ospf 1
R2(config-router)#network 10.3.0.0 0.0.0.3 area 0
R2(config-router)#network 10.2.0.0 0.0.0.3 area 0
R2(config-router)#network 192.168.50.0 0.0.0.255 area 0
R2(config-router)#network 192.168.20.0 0.0.0.255 area 0
R2(config-router)#exit
R2(config)#do wr
```

```
R3(config)#router ospf 1
R3(config-router)#network 192.168.40.0 0.0.0.255 area 0
R3(config-router)#network 192.168.30.0 0.0.0.255 area 0
R3(config-router)#network 10.1.0.0 0.0.0.3 area 0
R3(config-router)#network 10.2.0.0 0.0.0.3 area 0
R3(config-router)#exit
R3(config)#do wr
```

Pe lângă adresele rețelelor direct conectate la router, sunt indicate măștile inverse ale acestora și domeniul

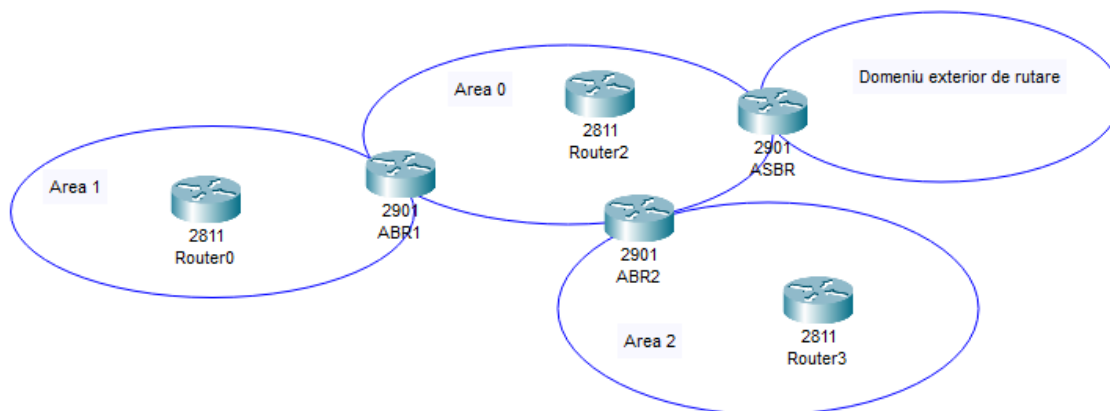
căruia aparțin.

Utilizarea domeniilor în OSPF

Protocolul OSPF solicită esențial resursele procesorului și memoria operativă.

Pentru a soluționa problema de solicitare consistentă a resurselor routerului (procesorul și memoria operativă) a fost propusă divizarea pe domenii, care includ un număr mai mic de routere. Routerelor dintr-un domeniu determină acolo rutele, după care transmit aceste rute routerelor din alte domenii, mai exact routerelor ABR (de la frontiera fiecărui domeniu), care le distribuie în interiorul domeniului.

Dacă există mai multe domenii, atunci numai decît trebuie să fie domeniul Area 0 astfel încât celelalte domenii să transmită datele anume prin Area 0 => aceasta este necesar pentru ca să nu se formeze cicluri între domenii.



În OSPF se consideră următoarele tipuri de routere:

Router ABR – router amplasat la frontiera dintre două domenii, astfel încât o interfață a acestuia este conectată la Area 0, iar celelalte – la alte domenii

Router de interior – toate interfețele sunt conectate la dispozitive din același domeniu

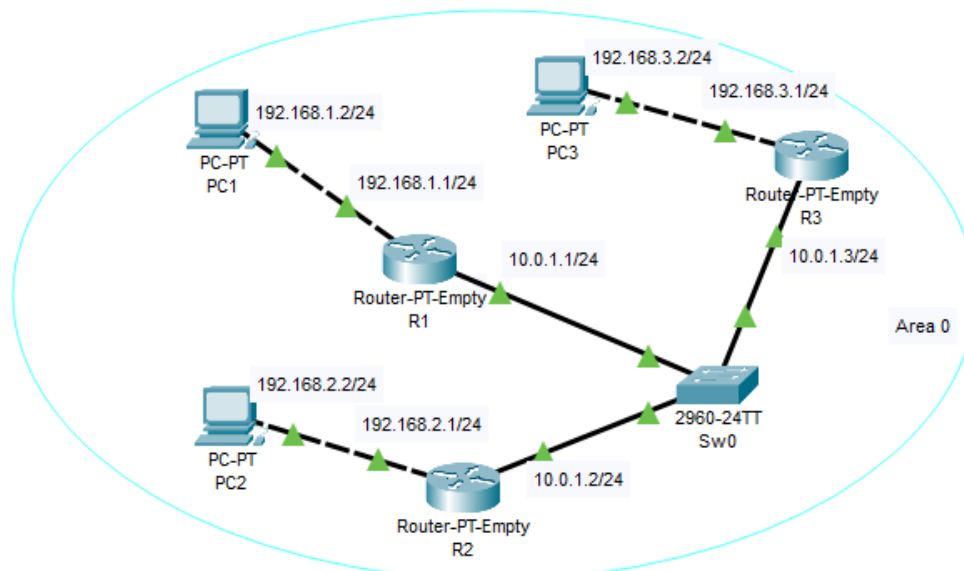
Router backbone – toate interfețele routerului sunt conectate la dispozitive din domeniul Area 0

Router ASBR – router amplasat la frontiera dintre două sisteme autonome

Construirea unei rețele ce constă din trei domenii OSPF, folosind Cisco Packet Tracer

Este necesar ca fiecare router dintr-un domeniu să „știe” cum să transmită pachete de date către routerelor din celelalte domenii

Mai întâi construim primul domeniu (a se vedea Figura de mai jos):



Configurăm pe interfețele fiecărui router IP adresele indicate în Figură. La fel, configurăm pe host-uri IP-urile indicate și masca /24.

Pe fiecare router din Area 0 configurăm protocolul de rutare dinamică OSPF:

| |
|--|
| Pe routerul R1: |
| R1(config)#router ospf 1 |
| R1(config-router)#network 192.168.1.0 0.0.0.255 area 0 |
| R1(config-router)#network 10.0.1.0 0.0.0.255 area 0 |

```
R1(config-router)#exit
R1(config)#do wr
```

Pe routerul R2:

```
R2(config)#router ospf 1
R2(config-router)#network 192.168.2.0 0.0.0.255 area 0
R2(config-router)#network 10.0.1.0 0.0.0.255 area 0
R2(config-router)#exit
R2(config)#do wr
```

Pe routerul R3:

```
R3(config)#router ospf 1
R3(config-router)#network 192.168.3.0 0.0.0.255 area 0
R3(config-router)#network 10.0.1.0 0.0.0.255 area 0
R3(config-router)#exit
R3(config)#do wr
```

Pentru a vedea rutele generate de OSPF pe routerul R2, vom folosi comanda *do show ip route*:

```
R2(config)#do show ip route
```

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

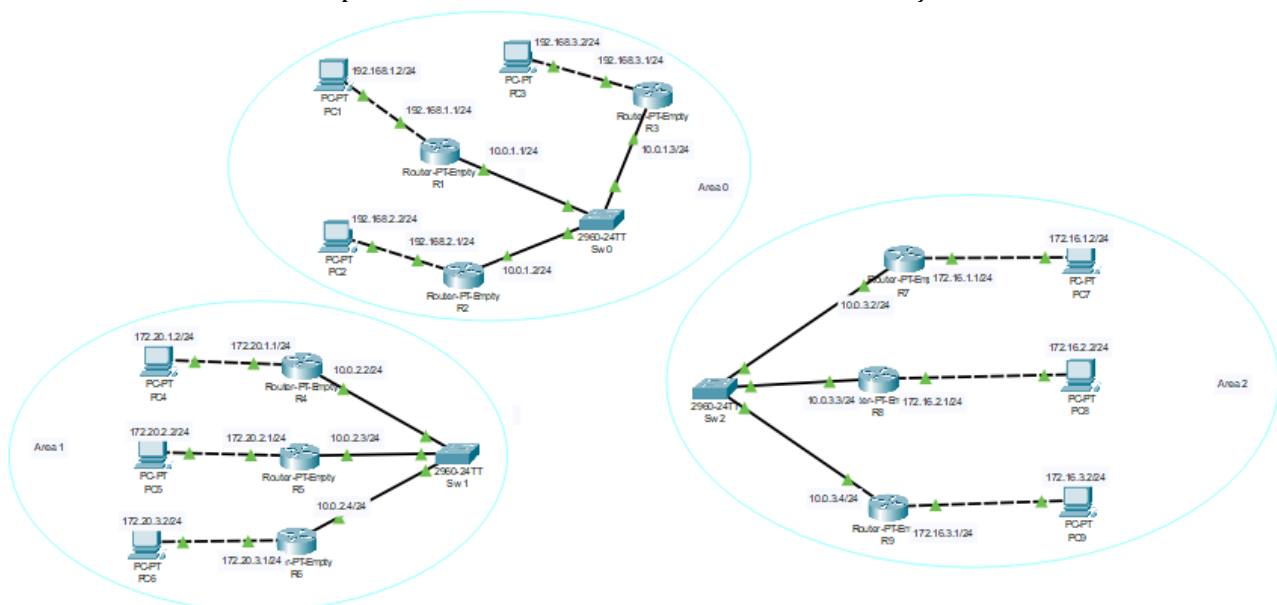
Gateway of last resort is not set

```
10.0.0.0/28 is subnetted, 1 subnets
C    10.0.1.0 is directly connected, GigabitEthernet9/0
O    192.168.1.0/24 [110/2] via 10.0.1.1, 00:05:23, GigabitEthernet9/0
C    192.168.2.0/24 is directly connected, GigabitEthernet8/0
O    192.168.3.0/24 [110/2] via 10.0.1.3, 00:02:22, GigabitEthernet9/0
```

Vedem în tabelul de rutare al routerului R2 două rute generate de OSPF, care sunt notate cu litera O

Analog putem vedea tabelele de rutare pentru routerule R1 și R3

Construim încă două domenii, Area 1 și Area 2, și configurăm dispozitivele corespunzătoare (a se vedea Figura de mai jos). IP-urile necesare pentru configurări sunt indicate în Figură. La fel, configurăm protocolul de rutare dinamică pe routerule din fiecare domeniu – Area 1 și Area 2.



De exemplu, pe routerul R4 (din Area 1) vom configura protocolul OSPF în modul următor:

```
Pe routerul R4:
R4(config)#router ospf 1
R4(config-router)#network 172.20.1.0 0.0.0.255 area 1
```

```

R4(config-router)#network 10.0.2.0 0.0.0.255 area 1
R4(config-router)#exit
R4(config)#do wr

```

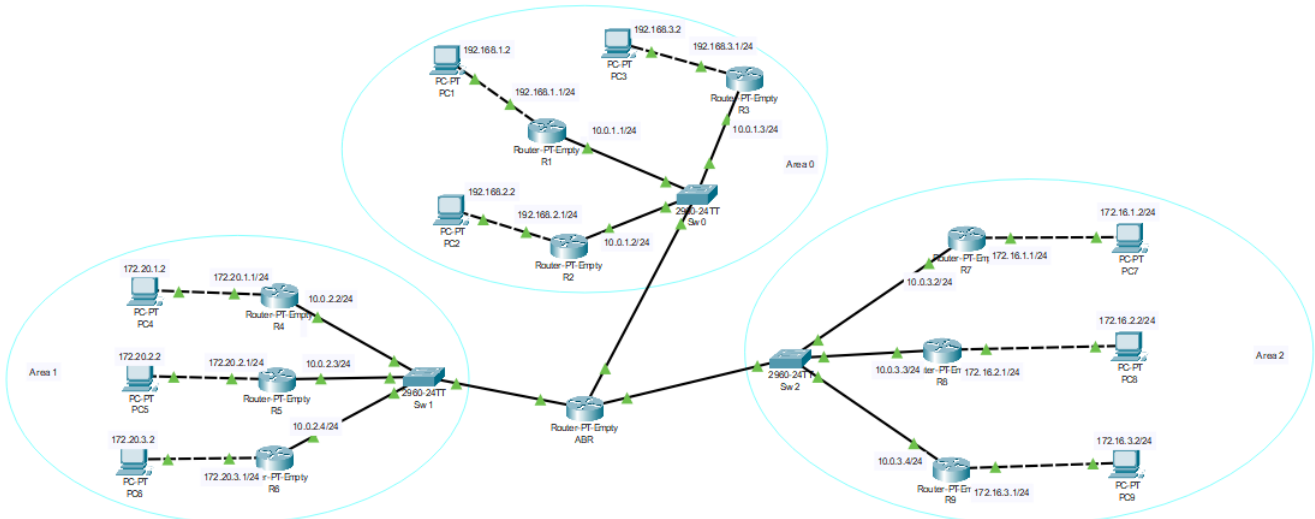
iar pe routerul R8 (din Area 2) astfel:

```

Pe routerul R8:
R8(config)#router ospf 1
R8(config-router)#network 10.0.3.0 0.0.0.255 area 2
R8(config-router)#network 172.16.2.0 0.0.0.255 area 2
R8(config-router)#exit
R8(config)#do wr

```

Pentru a conecta cele trei domenii este necesar un router ABR cu trei interfețe. Vom insera un router *PT Empty*, la care vom adăuga trei module de interfață

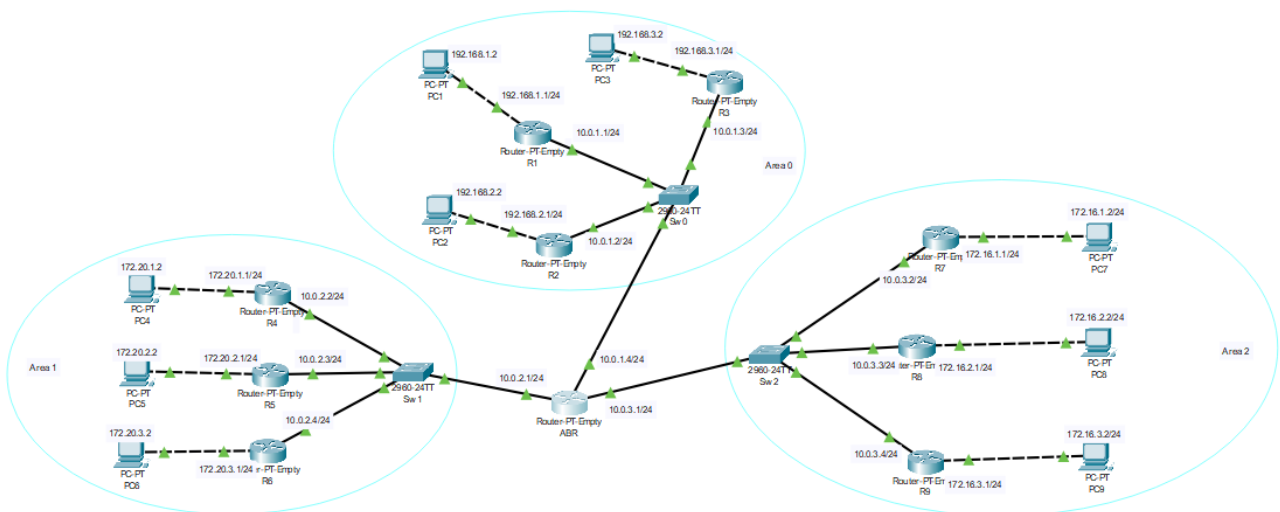


Configurăm routerul ABR (a se vedea în Figură IP adresele corespunzătoare interfețelor acestuia), după care configurăm protocolul de rutare dinamică OSPF pe acesta:

```

ABR(config)#router ospf 1
ABR(config-router)#network 10.0.1.0 0.0.0.255 area 0
ABR(config-router)#network 10.0.2.0 0.0.0.255 area 1
ABR(config-router)#network 10.0.3.0 0.0.0.255 area 2
ABR(config-router)#exit
ABR(config)#do wr

```



Rolul routerelor de tip DR, BDR și DROTHER

În acest moment în tabelul de rutare al routerului ABR sunt scrise toate rutele din rețea:

```

ABR(config)#do show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area

```



```
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
```

```
Gateway of last resort is not set
```

```
10.0.0.0/24 is subnetted, 3 subnets
C 10.0.1.0 is directly connected, GigabitEthernet8/0
C 10.0.2.0 is directly connected, GigabitEthernet7/0
C 10.0.3.0 is directly connected, GigabitEthernet9/0
172.16.0.0/24 is subnetted, 3 subnets
O 172.16.1.0 [110/2] via 10.0.3.2, 00:01:31, GigabitEthernet9/0
O 172.16.2.0 [110/2] via 10.0.3.3, 00:01:31, GigabitEthernet9/0
O 172.16.3.0 [110/2] via 10.0.3.4, 00:01:31, GigabitEthernet9/0
172.20.0.0/24 is subnetted, 3 subnets
O 172.20.1.0 [110/2] via 10.0.2.2, 00:01:01, GigabitEthernet7/0
O 172.20.2.0 [110/2] via 10.0.2.3, 00:01:01, GigabitEthernet7/0
O 172.20.3.0 [110/2] via 10.0.2.4, 00:01:01, GigabitEthernet7/0
O 192.168.1.0/24 [110/2] via 10.0.1.1, 00:01:15, GigabitEthernet8/0
O 192.168.2.0/24 [110/2] via 10.0.1.2, 00:01:15, GigabitEthernet8/0
O 192.168.3.0/24 [110/2] via 10.0.1.3, 00:01:15, GigabitEthernet8/0
```

Să vedem dacă au fost transmise rutele din domeniile Area 1 și Area 2 în tabelul de rutare al routerului R2 din Area 0:

```
R2(config)#do show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
        i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
        * - candidate default, U - per-user static route, o - ODR
        P - periodic downloaded static route
```

```
Gateway of last resort is not set
```

```
10.0.0.0/24 is subnetted, 2 subnets
C      10.0.1.0 is directly connected, GigabitEthernet9/0
O IA    10.0.3.0 [110/2] via 10.0.1.4, 00:06:41, GigabitEthernet9/0
O      192.168.1.0/24 [110/2] via 10.0.1.1, 00:45:57, GigabitEthernet9/0
C      192.168.2.0/24 is directly connected, GigabitEthernet8/0
O      192.168.3.0/24 [110/2] via 10.0.1.3, 00:46:07, GigabitEthernet9/0
```

Răspunsul este nu! Aceasta se întâmplă, deoarece între routere nu este construită corect relația ierarhică (care este routerul central și care sunt restul!)

Routerul ABR aparține (prin interfețele corespunzătoare) celor 3 domenii Area 0, Area 1 și Area 2

Routerele din același domeniu schimbă între ele mesaje LSA printr-un router, numit router central sau router DR.

Atunci când sunt conectate, routerele sunt implicate într-un proces de alegere (election) a routerului DR și a adjunctului său – numit BDR.

După ce este ales routerul central DR, celelalte routere din limitele unui domeniu broadcast transmit acestuia mesaje LSA, iar routerul DR, după ce a adunat această informație, o transmite înapoi fiecărui router din domeniul broadcast, dar și routerelor din celelalte domenii broadcast ale rețelei.

În rețeaua examinată în calitate de router DR ar fi trebuit să fie ales routerul ABR, dar în realitate nu el a fost ales. Putem vedea aceasta, folosind comanda *show ip ospf neighbor* pe routerul R2:

```
R2(config)#do show ip ospf neighbor
```

| Neighbor ID | Pri | State | Dead Time | Address |
|--------------------|-----|--------------|-----------|----------|
| Interface | | | | |
| 192.168.1.1 | 1 | FULL/BDR | 00:00:37 | 10.0.1.1 |
| GigabitEthernet9/0 | | | | |
| 192.168.3.1 | 1 | FULL/DR | 00:00:37 | 10.0.1.3 |
| GigabitEthernet9/0 | | | | |
| 10.0.3.1 | 1 | 2WAY/DROTHER | 00:00:32 | 10.0.1.4 |
| GigabitEthernet9/0 | | | | |

În calitate de router DR a fost ales routerul R3 (prin interfața sa 10.0.1.3), iar adjunctul său BDR (acesta devine DR în cazul în care iese din funcțiune routerul DR inițial) este R1.

Router DR trebuie să fie cel ABR prin interfața sa 10.0.1.4, iar acest lucru poate fi realizat dacă proprietatea *priority* a acestuia va fi făcută mai mare decât pentru celelalte routere din acel domeniu broadcast (R1, R2, R3).

Prin definiție prioritățile tuturor routerelor sunt egale cu 1. Dacă vom face ca prioritatea routerului ABR să fie 100 (de exemplu!), acesta va fi reales în calitate de router DR.

ABR routerul stă la frontiera domeniului Area 0. Proprietatea *priority* este setată în raport cu domeniul Area 0, iar pentru aceasta este necesar să se seteze prioritatea (folosind comanda *ip ospf priority 100*) în raport cu interfața lui ABR ce corespunde domeniului Area 0. Analog, trebuie de setat prioritatea 100 în raport cu interfețele lui ABR comune cu domeniile Area 1 și, respectiv, Area2. În linia de comandă pentru routerul ABR scriem:

```
ABR(config)#int gig 8/0
ABR(config-if)#ip ospf priority 100
ABR(config-if)#int gig 7/0
ABR(config-if)#ip ospf priority 100
ABR(config-if)#int gig 9/0
ABR(config-if)#ip ospf priority 100
ABR(config-if)#exit
ABR(config)#do clear ip ospf process
Reset ALL OSPF processes? [no]: y
```

Prin comanda *clear ip ospf process* se reinițializează procesul de alegere (election) a routerului DR. Este necesar de confirmat „yes” inițializarea procesului.

După ce rulează o serie de acțiuni, putem vizualiza din nou pe routerul R2 cine este noul router DR:

```
R2(config)#do show ip ospf neighbor
```

| Neighbor ID | Pri | State | Dead Time | Address | Interface |
|-------------|-----|--------------|-----------|----------|--------------------|
| 192.168.1.1 | 1 | FULL/BDR | 00:00:31 | 10.0.1.1 | GigabitEthernet9/0 |
| 192.168.3.1 | 1 | 2WAY/DROTHER | 00:00:31 | 10.0.1.3 | GigabitEthernet9/0 |
| 10.0.3.1 | 100 | FULL/DR | 00:00:36 | 10.0.1.4 | GigabitEthernet9/0 |

La fel, de exemplu, pe routerul R5 din Area 1 avem:

```
R5(config)#do show ip ospf neighbor
```

| Neighbor ID | Pri | State | Dead Time | Address |
|--------------------|-----|--------------|-----------|----------|
| Interface | | | | |
| 172.20.1.1 | 1 | FULL/DROTHER | 00:00:36 | 10.0.2.2 |
| GigabitEthernet8/0 | | | | |
| 172.20.3.1 | 1 | FULL/DROTHER | 00:00:36 | 10.0.2.4 |
| GigabitEthernet8/0 | | | | |
| 10.0.3.1 | 100 | FULL/DR | 00:00:39 | 10.0.2.1 |
| GigabitEthernet8/0 | | | | |

iar pe routerul R7 din Area 2, respectiv:

```
R7(config)#do show ip ospf neighbor
```

| Neighbor ID | Pri | State | Dead Time | Address |
|--------------------|-----|--------------|-----------|----------|
| Interface | | | | |
| 172.16.2.1 | 1 | FULL/BDR | 00:00:36 | 10.0.3.3 |
| GigabitEthernet8/0 | | | | |
| 172.16.3.1 | 1 | 2WAY/DROTHER | 00:00:33 | 10.0.3.4 |
| GigabitEthernet8/0 | | | | |
| 10.0.3.1 | 100 | FULL/DR | 00:00:32 | 10.0.3.1 |
| GigabitEthernet8/0 | | | | |

Astfel, pentru fiecare din cele trei domenii routerul ABR a devenit router DR.

Să vedem ce s-a întâmplat cu tabelul de rutare pe R2:

```
R2(config)#do show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
```

```
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
```

Gateway of last resort is not set

```
10.0.0.0/24 is subnetted, 3 subnets
C 10.0.1.0 is directly connected, GigabitEthernet9/0
O IA 10.0.2.0 [110/2] via 10.0.1.4, 00:00:45, GigabitEthernet9/0
O IA 10.0.3.0 [110/2] via 10.0.1.4, 00:00:45, GigabitEthernet9/0
172.16.0.0/24 is subnetted, 3 subnets
O IA 172.16.1.0 [110/3] via 10.0.1.4, 00:00:45, GigabitEthernet9/0
O IA 172.16.2.0 [110/3] via 10.0.1.4, 00:00:45, GigabitEthernet9/0
O IA 172.16.3.0 [110/3] via 10.0.1.4, 00:00:45, GigabitEthernet9/0
172.20.0.0/24 is subnetted, 3 subnets
O IA 172.20.1.0 [110/3] via 10.0.1.4, 00:00:35, GigabitEthernet9/0
O IA 172.20.2.0 [110/3] via 10.0.1.4, 00:00:35, GigabitEthernet9/0
O IA 172.20.3.0 [110/3] via 10.0.1.4, 00:00:45, GigabitEthernet9/0
O 192.168.1.0/24 [110/2] via 10.0.1.1, 00:00:45, GigabitEthernet9/0
C 192.168.2.0/24 is directly connected, GigabitEthernet8/0
O 192.168.3.0/24 [110/2] via 10.0.1.3, 00:00:45, GigabitEthernet9/0
```

Prin simbolul IA (Inter Area) sunt notate rutele din alte domenii, care au fost recepționate prin routerul ABR

Verificăm conexiunea între host-urile din diferite domenii:

Dăm un ping de pe host-ul PC2 către PC5 din Area 1 și către PC9 din Area 2:

```
C:\>ping 172.20.2.2

Pinging 172.20.2.2 with 32 bytes of data:

Request timed out.
Reply from 172.20.2.2: bytes=32 time<1ms TTL=125
Reply from 172.20.2.2: bytes=32 time=1ms TTL=125
Reply from 172.20.2.2: bytes=32 time<1ms TTL=125

Ping statistics for 172.20.2.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>ping 172.16.3.2

Pinging 172.16.3.2 with 32 bytes of data:

Request timed out.
Reply from 172.16.3.2: bytes=32 time<1ms TTL=125
Reply from 172.16.3.2: bytes=32 time<1ms TTL=125
Reply from 172.16.3.2: bytes=32 time<1ms TTL=125

Ping statistics for 172.16.3.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Concluzia: este conexiune între host-urile din diferite domenii prin intermediul rutelor generate de către protocolul OSPF

A fost realizată o economie de resurse prin aceea că routerele R1, R2 și R3 nu știu despre rutele existente pe routerele din celelalte două domenii (și reciproc), dar acestea transmit date prin routerul ABR, care știe toate rutele generate de routerele din rețea

Menționăm că pe lângă routerul DR există în domeniul broadcast și routerul BDR, precum și routere DROTHER. Routerul BDR va deveni router DR dacă routerul DR curent va ieși din funcțiune. Routerul DROTHER este acel router care nu este nici DR, nici BDR. Dacă vor ieși din funcțiune și DR și BDR, atunci routerele DROTHER vor organiza re alegerea unui nou router DR și BDR.

Redistribuirea rutelor între protocoalele de rutare dinamică

Deseori în diferite sisteme autonome (domenii de rutare) rulează diferite protocoale de rutare dinamică. Chiar și în cadrul unei rețele compuse pe anumite segmente ar putea să fie configurate diferite protocoale de rutare.

În acest caz, pentru a asigura transmiterea pachetelor de date între oricare două dispozitive din rețea, se aplică procedeul de redistribuire a rutelor, care permite ca rutele generate de un protocol să fie transmise spre utilizare unui alt protocol de rutare (se asigură comunicare între protocoalele de rutare). Protocolul de rutare care a recepționat rutele redistribuite, le etichetează pe acestea din urmă ca fiind rute externe. Între două domenii de rutare trebuie să existe cel puțin un punct de redistribuire - un router pe care vor rula ambele protocoale de rutare din cele două domenii.

Este posibil de realizat redistribuirea rutelor unui protocol în același protocol ce rulează în alt domeniu de rutare. La fel, pot fi redistribuite rutele statice într-un protocol de rutare dinamică.

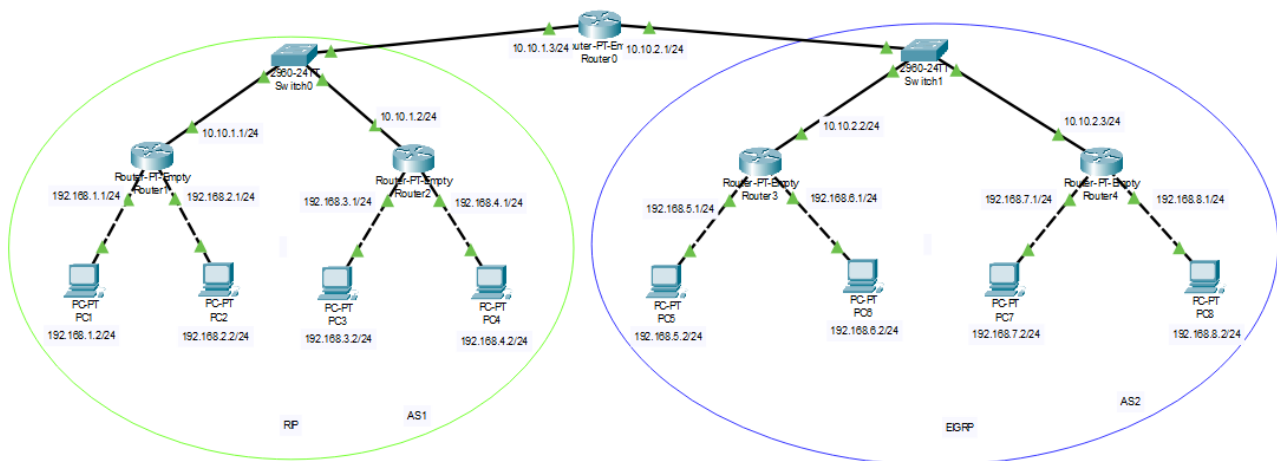
Pot fi redistribuite acele rute care se conțin în tabelul de rutare al routerului.

Metrica de rutare reprezintă componenta de bază în redistribuirea rutelor. Cu excepția lui EIGRP, celelalte protocoale utilizează o metrică unică. Rutelor redistribuite trebuie să li se asocieze manual o metrică pe care o „înțelege” protocolul ce recepționează aceste rute.

Redistribuirea rutelor între RIP și EIGRP

Amintim că RIP și EIGRP sunt protocoale bazate pe vectori de distanță, care utilizează în calitate de metrică numărul de hop-uri (routere) intermediare până la rețeaua destinație.

Să considerăm rețeaua următoare:



În cele 5 subrețele din AS1 este configurat protocolul de rutare dinamică RIP, iar în AS2 – protocolul EIGRP. Vom redistribui rutele între AS1 și AS2 prin intermediul routerului Router0. În acest sens, mai întâi vom configura pe Router0 atât protocolul RIP, cât și EIGRP, declarând pentru RIP rețeaua 10.10.1.0/24, iar pentru EIGRP – rețeaua 10.10.2.0/24:

```
Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 10.10.1.0
Router(config-router)#no auto-summary
Router(config-router)#exit
Router(config)#router eigrp 1
Router(config-router)#network 10.10.2.0 0.0.0.255
Router(config-router)#no auto-summary
Router(config-router)#do wr
```

Pe Router0 se redistribuie rutele RIP în EIGRP și reciproc:

```
Router(config)#router eigrp 1
Router(config-router)#redistribute rip metric 10000 100 255 1 1500
Router(config-router)#exit
```

```
Router(config)#router rip
Router(config-router)#redistribute eigrp 1 metric 1
Router(config-router)#exit
Router(config)#do wr
```

EIGRP implică cinci valori atunci când redistribuie rutele din alte protocoale (în loc de *redistribute rip* poate fi *redistribute ospf*, *redistribute static* ș.a.): lățimea de bandă (bandwidth), întârzierea (delay), fiabilitatea (reliability), încărcarea (load) și, respectiv, MTU.

În rezultatul execuției comenzilor menționate mai sus, tabelele de rutare pentru Router1 și Router2 vor include toate rutele din configurația de rețea:

```
10.0.0.0/24 is subnetted, 2 subnets
C    10.10.1.0 is directly connected, GigabitEthernet7/0
R    10.10.2.0 [120/1] via 10.10.1.3, 00:00:08, GigabitEthernet7/0
C    192.168.1.0/24 is directly connected, GigabitEthernet9/0
C    192.168.2.0/24 is directly connected, GigabitEthernet8/0
R    192.168.3.0/24 [120/1] via 10.10.1.2, 00:00:14,
GigabitEthernet7/0
R    192.168.4.0/24 [120/1] via 10.10.1.2, 00:00:14,
GigabitEthernet7/0
R    192.168.5.0/24 [120/1] via 10.10.1.3, 00:00:08,
GigabitEthernet7/0
R    192.168.6.0/24 [120/1] via 10.10.1.3, 00:00:08,
GigabitEthernet7/0
R    192.168.7.0/24 [120/1] via 10.10.1.3, 00:00:08,
GigabitEthernet7/0
R    192.168.8.0/24 [120/1] via 10.10.1.3, 00:00:08,
GigabitEthernet7/0
```

La fel, și tabelele de rutare pentru Router3 și Router 4:

```
10.0.0.0/24 is subnetted, 2 subnets
D EX 10.10.1.0 [170/258560] via 10.10.2.1, 00:08:15,
GigabitEthernet9/0
C    10.10.2.0 is directly connected, GigabitEthernet9/0
D EX 192.168.1.0/24 [170/258560] via 10.10.2.1, 00:08:14,
GigabitEthernet9/0
D EX 192.168.2.0/24 [170/258560] via 10.10.2.1, 00:08:14,
GigabitEthernet9/0
D EX 192.168.3.0/24 [170/258560] via 10.10.2.1, 00:08:14,
GigabitEthernet9/0
D EX 192.168.4.0/24 [170/258560] via 10.10.2.1, 00:08:14,
GigabitEthernet9/0
C    192.168.5.0/24 is directly connected, GigabitEthernet7/0
C    192.168.6.0/24 is directly connected, GigabitEthernet8/0
D    192.168.7.0/24 [90/7680] via 10.10.2.3, 00:08:14,
GigabitEthernet9/0
D    192.168.8.0/24 [90/7680] via 10.10.2.3, 00:08:14,
GigabitEthernet9/0
```

Dacă dăm un ping între oricare două host-uri din rețea, vom vedea că există conexiune între acestea. Redistribuirea EIGRP într-un alt proces EIGRP nu necesită careva conversie de metrică, deci nu este necesar să se definească vreo metrică (de exemplu, cea predefinită) pe parcursul redistribuirii.

Pentru redistribuirea în EIGRP din alte protocoale se va utiliza una din variantele indicate mai jos:

```
router eigrp 1
 redistribute static
 redistribute ospf 1
 redistribute rip
 default-metric 10000 100 255 1 1500
```

Pentru redistribuirea în RIP din alte protocoale se va utiliza una din variantele indicate mai jos:

```
router rip
 redistribute static
 redistribute eigrp 1
 redistribute ospf 1
 default-metric 1
```

Redistribuirea între EIGRP și OSPF

În cele 5 subrețele din AS1 este configurat protocolul de rutare dinamică EIGRP, iar în AS2 – protocolul OSPF.

Vom redistribui rutele între AS1 și AS2 prin intermediul routerului Router0. În acest sens, mai întâi vom configura pe Router0 atât protocolul EIGRP, cât și OSPF, declarând pentru EIGRP rețeaua 10.10.1.0/24, iar pentru OSPF – rețeaua 10.10.2.0/24:

```

Router(config)#router eigrp 1
Router(config-router)#network 10.10.1.0 0.0.0.255
Router(config-router)#no auto-summary
Router(config-router)#exit
Router(config)#do wr
Router(config)#router ospf 1
Router(config-router)#network 10.10.2.0 0.0.0.255 area 0
Router(config-router)#exit
Router(config)#do wr

```

Pe Router0 se redistribuie rutele EIGRP în OSPF și reciproc:

```

Router(config)#router eigrp 1
Router(config-router)#redistribute ospf 1 metric 10000 100 255 1 1500
Router(config-router)#exit

Router(config)#router ospf 1
Router(config-router)#redistribute eigrp 1 metric 100 subnets
Router(config-router)#exit
Router(config)#do wr

```

În rezultatul execuției comenzilor menționate mai sus, tabelele de rutare pentru Router1 și Router2 vor include toate rutele din configurația de rețea:

```

10.0.0.0/24 is subnetted, 2 subnets
C      10.10.1.0 is directly connected, GigabitEthernet7/0
D EX   10.10.2.0 [170/284160] via 10.10.1.3, 05:28:14,
GigabitEthernet7/0
C      192.168.1.0/24 is directly connected, GigabitEthernet9/0
C      192.168.2.0/24 is directly connected, GigabitEthernet8/0
D      192.168.3.0/24 [90/7680] via 10.10.1.2, 06:05:02,
GigabitEthernet7/0
D      192.168.4.0/24 [90/7680] via 10.10.1.2, 06:05:02,
GigabitEthernet7/0
D EX 192.168.5.0/24 [170/284160] via 10.10.1.3, 05:28:14,
GigabitEthernet7/0
D EX 192.168.6.0/24 [170/284160] via 10.10.1.3, 05:28:14,
GigabitEthernet7/0
D EX 192.168.7.0/24 [170/284160] via 10.10.1.3, 05:28:14,
GigabitEthernet7/0
D EX 192.168.8.0/24 [170/284160] via 10.10.1.3, 05:28:14,
GigabitEthernet7/0

```

La fel, și tabelele de rutare pentru Router3 și Router 4:

```

10.0.0.0/24 is subnetted, 2 subnets
O E2   10.10.1.0 [110/100] via 10.10.2.1, 05:27:19,
GigabitEthernet9/0
C      10.10.2.0 is directly connected, GigabitEthernet9/0
O E2 192.168.1.0/24 [110/100] via 10.10.2.1, 05:27:19,
GigabitEthernet9/0
O E2 192.168.2.0/24 [110/100] via 10.10.2.1, 05:27:19,
GigabitEthernet9/0
O E2 192.168.3.0/24 [110/100] via 10.10.2.1, 05:27:19,
GigabitEthernet9/0
O E2 192.168.4.0/24 [110/100] via 10.10.2.1, 05:27:19,
GigabitEthernet9/0
O     192.168.5.0/24 [110/2] via 10.10.2.2, 05:55:46,
GigabitEthernet9/0
O     192.168.6.0/24 [110/2] via 10.10.2.2, 05:55:46,
GigabitEthernet9/0
C      192.168.7.0/24 is directly connected, GigabitEthernet7/0
C      192.168.8.0/24 is directly connected, GigabitEthernet8/0

```

Dacă dăm un ping între oricare două host-uri din rețea, vom vedea că există conexiune între acestea.

Pentru a redistribui rute statice, RIP sau EIGRP pe un router pe care este configurat protocolul OSPF se vor utiliza comenzile

```

router ospf 1
redistribute static metric 200 subnets
redistribute rip metric 200 subnets
redistribute eigrp 1 metric 100 subnets

```


Metrica OSPF este costul definit ca 10^8 / lăţimea de bandă a link-ului în biţi/sec. De exemplu, costul OSPF al Ethernet este $10: 10^8/10^7 = 10$

Dacă nu este specificată o metrică, OSPF atribuie o valoare implicită de 20 la redistribuirea rutelor din toate protocoalele, cu excepţia rutelor din protocolul BGP (Border Gateway Protocol), care obţin metrica 1.

Nu trebuie să definim o metrică sau să utilizăm metrica implicită atunci când redistribuim un proces OSPF într-un alt proces OSPF (de exemplu, *redistribute ospf 2 subnets*).

Redistribuirea rutelor statice printre rutele protocoalelor de rutare dinamică

În cele 5 subreţele din AS1 este realizată rutarea statică, iar în AS2 – protocolul OSPF.

Vom redistribui rutele între AS1 şi AS2 prin intermediul routerului Router0.

Nu putem redistribui rutele unui protocol de rutare dinamică în cele statice. Pentru aceasta, pe fiecare router din AS1 (în care se configurează static rutele!) vom defini ca statice rutele din AS2, toate cu IP-ul interfeţei de intrare 10.10.1.3 al Router0 (din partea AS1!).

Pe Router0 configurăm atât rutele statice, cât şi cele OSPF:

```
Router(config)#ip route 192.168.1.0 255.255.255.0 10.10.1.1
Router(config)#ip route 192.168.2.0 255.255.255.0 10.10.1.1
Router(config)#ip route 192.168.3.0 255.255.255.0 10.10.1.2
Router(config)#ip route 192.168.4.0 255.255.255.0 10.10.1.2
Router(config)#router ospf 1
Router(config-router)#network 10.10.2.0 0.0.0.255 area 0
```

Pe Router0 se redistribuie rutele statice în OSPF:

```
Router(config)#router ospf 1
Router(config-router)#redistribute static metric 200 subnets
Router(config-router)#exit
```

În rezultatul execuţiei comenzilor menţionate mai sus, tabelele de rutare pentru Router1 şi Router2 vor include toate rutele din configuraţia de reţea:

```
10.0.0.0/24 is subnetted, 2 subnets
C    10.10.1.0 is directly connected, GigabitEthernet7/0
S    10.10.2.0 [1/0] via 10.10.1.3
C    192.168.1.0/24 is directly connected, GigabitEthernet9/0
C    192.168.2.0/24 is directly connected, GigabitEthernet8/0
S    192.168.3.0/24 [1/0] via 10.10.1.2
S    192.168.4.0/24 [1/0] via 10.10.1.2
S    192.168.5.0/24 [1/0] via 10.10.1.3
S    192.168.6.0/24 [1/0] via 10.10.1.3
S    192.168.7.0/24 [1/0] via 10.10.1.3
S    192.168.8.0/24 [1/0] via 10.10.1.3
```

La fel, şi tabelele de rutare pentru Router3 şi Router 4:

```
10.0.0.0/24 is subnetted, 1 subnets
C    10.10.2.0 is directly connected, GigabitEthernet9/0
O E2 192.168.1.0/24 [110/200] via 10.10.2.1, 00:13:54,
GigabitEthernet9/0
O E2 192.168.2.0/24 [110/200] via 10.10.2.1, 00:13:54,
GigabitEthernet9/0
O E2 192.168.3.0/24 [110/200] via 10.10.2.1, 00:13:54,
GigabitEthernet9/0
O E2 192.168.4.0/24 [110/200] via 10.10.2.1, 00:13:54,
GigabitEthernet9/0
C    192.168.5.0/24 is directly connected, GigabitEthernet7/0
C    192.168.6.0/24 is directly connected, GigabitEthernet8/0
O    192.168.7.0/24 [110/2] via 10.10.2.3, 00:48:17,
GigabitEthernet9/0
O    192.168.8.0/24 [110/2] via 10.10.2.3, 00:48:04,
GigabitEthernet9/0
```

Dacă dăm un ping între oricare două host-uri din reţea, vom vedea că există conexiune între acestea.

Redistribuirea rutelor între două sisteme autonome cu acelaşi protocol de rutare

Vom ilustra printr-un exemplu modul în care are loc redistribuirea rutelor dinamice între două procese OSPF (din două sisteme autonome).

Vom redistribui rutele între AS1 şi AS2 prin intermediul routerului Router0. În acest sens, mai întâi vom configura pe Router0 două procese OSPF (1 şi 2):

```
Router(config)#router ospf 1
```

```
Router(config-router)#network 10.10.1.0 0.0.0.255 area 0
Router(config-router)#exit
Router(config)#router ospf 2
Router(config-router)#network 10.10.2.0 0.0.0.255 area 0
Router(config-router)#exit
```

Pe Router0 se redistribuie rutele OSPF 1 în OSPF 2 și reciproc:

```
Router(config)#router ospf 1
Router(config-router)#redistribute ospf 2 subnets
Router(config-router)#exit
```

```
Router(config)#router ospf 2
Router(config-router)#redistribute ospf 1 subnets
Router(config-router)#exit
```

În rezultatul execuției comenzilor menționate mai sus, tabelele de rutare pentru Router1 și Router2 vor include toate rutele din configurația de rețea:

```
10.0.0.0/24 is subnetted, 2 subnets
C      10.10.1.0 is directly connected, GigabitEthernet7/0
O E2   10.10.2.0 [110/20] via 10.10.1.3, 00:04:27,
GigabitEthernet7/0
C      192.168.1.0/24 is directly connected, GigabitEthernet9/0
C      192.168.2.0/24 is directly connected, GigabitEthernet8/0
O      192.168.3.0/24 [110/2] via 10.10.1.2, 00:15:17,
GigabitEthernet7/0
O      192.168.4.0/24 [110/2] via 10.10.1.2, 00:15:17,
GigabitEthernet7/0
O E2   192.168.5.0/24 [110/20] via 10.10.1.3, 00:04:16,
GigabitEthernet7/0
O E2   192.168.6.0/24 [110/20] via 10.10.1.3, 00:04:16,
GigabitEthernet7/0
O E2   192.168.7.0/24 [110/20] via 10.10.1.3, 00:04:16,
GigabitEthernet7/0
O E2   192.168.8.0/24 [110/20] via 10.10.1.3, 00:04:16,
GigabitEthernet7/0
```

La fel, și tabelele de rutare pentru Router3 și Router 4:

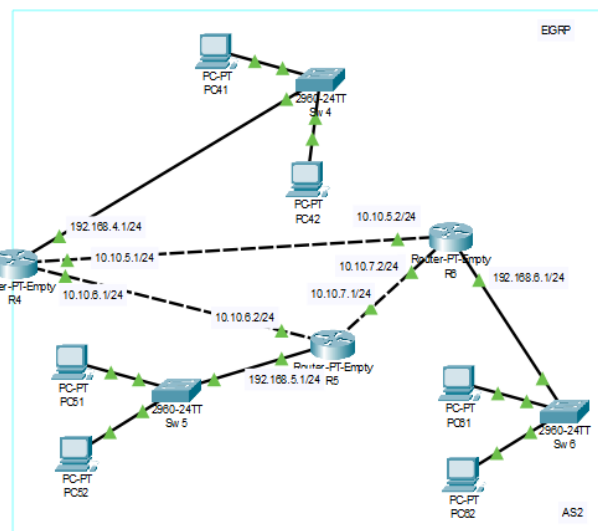
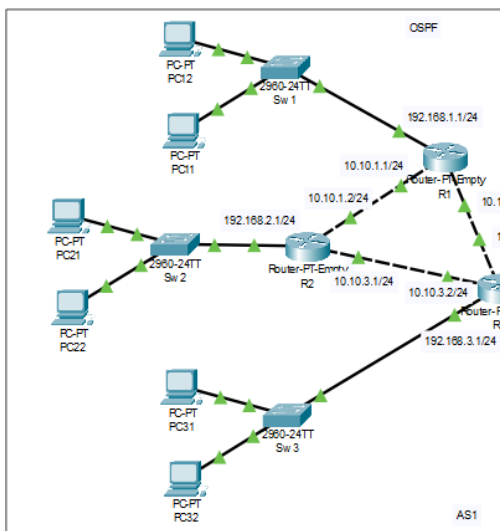
```
10.0.0.0/24 is subnetted, 2 subnets
O E2   10.10.1.0 [110/20] via 10.10.2.1, 00:02:46,
GigabitEthernet9/0
C      10.10.2.0 is directly connected, GigabitEthernet9/0
O E2   192.168.1.0/24 [110/20] via 10.10.2.1, 00:02:46,
GigabitEthernet9/0
O E2   192.168.2.0/24 [110/20] via 10.10.2.1, 00:02:46,
GigabitEthernet9/0
O E2   192.168.3.0/24 [110/20] via 10.10.2.1, 00:02:46,
GigabitEthernet9/0
O E2   192.168.4.0/24 [110/20] via 10.10.2.1, 00:02:46,
GigabitEthernet9/0
C      192.168.5.0/24 is directly connected, GigabitEthernet7/0
C      192.168.6.0/24 is directly connected, GigabitEthernet8/0
O      192.168.7.0/24 [110/2] via 10.10.2.3, 00:10:02,
GigabitEthernet9/0
O      192.168.8.0/24 [110/2] via 10.10.2.3, 00:09:52,
GigabitEthernet9/0
```

Protocolul de rutare dinamică BGP

BGP (Border Gateway Protocol) este un protocol de rutare între sistemele autonome, care este conceput pentru a oferi rutare fără bucle între organizații (a se vedea Figura de mai jos). În continuare este explicată implementarea Cisco IOS a configurării protocolului BGP.

De regulă, BGP este utilizat pentru a conecta o rețea locală la o rețea externă cu scopul de a obține acces la Internet sau pentru a realiza conectarea la alte organizații. Când este realizată conexiunea la o organizație externă, sunt create sesiuni de peering extern BGP (eBGP). Deși BGP este denumit protocol de gateway exterior (EGP), multe rețele din cadrul organizațiilor au devenit atât de complexe, încât BGP poate fi utilizat pentru a simplifica rețeaua internă utilizată în cadrul organizației. În cadrul aceleiași organizații se schimbă informații de rutare prin sesiuni interne de peering BGP (iBGP).

Vom ilustra cum se configurează protocolul BGP pe exemplul următoarei rețele:



În AS1 pe routerele R1, R2 și R3 este configurat protocolul OSPF. Pe routerul R3 la configurarea protocolului OSPF se indică doar rețelele ce se referă la AS1 (rețeaua 10.10.4.0 asigură legătura dintre AS1 și AS2!):

```
router ospf 1
 network 10.10.3.0 0.0.0.255 area 0
 network 10.10.2.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
 exit
```

Pe routerul R3 este configurat protocolul BGP:

```
router bgp 1 (routerul R3 se află în AS1)
 neighbor 10.10.4.2 remote-as 2 (routerul de legătură R4 se află în AS2)
 (În continuare se indică adresele și măștile rețelelor din AS1:)
 network 192.168.1.0 mask 255.255.255.0
 network 192.168.2.0 mask 255.255.255.0
 network 192.168.3.0 mask 255.255.255.0
 network 10.10.1.0 mask 255.255.255.0
 network 10.10.2.0 mask 255.255.255.0
 network 10.10.3.0 mask 255.255.255.0
 exit
```

În AS2 pe routerele R4, R5 și R6 este configurat protocolul EIGRP. Pe routerul R4 la configurarea protocolului EIGRP se indică doar rețelele ce se referă la AS2 (rețeaua 10.10.4.0 asigură legătura dintre AS1 și AS2!):

```
router eigrp 1
 network 192.168.4.0
 network 10.10.5.0 0.0.0.255
 network 10.10.6.0 0.0.0.255
 no auto-summary
 exit
```

Pe routerul R4 este configurat protocolul BGP:

```
router bgp 2 (routerul R4 se află în AS2)
 neighbor 10.10.4.1 remote-as 1 (routerul de legătură R3 se află în AS1)
 (În continuare se indică adresele și măștile rețelelor din AS2:)
 network 192.168.4.0 mask 255.255.255.0
 network 192.168.5.0 mask 255.255.255.0
 network 192.168.6.0 mask 255.255.255.0
 network 10.10.5.0 mask 255.255.255.0
 network 10.10.6.0 mask 255.255.255.0
 network 10.10.7.0 mask 255.255.255.0
 exit
```

La moment, în tabelul de rutare al routerului R3, datorită protocolului BGP, sunt prezente toate rutele din rețea:

```
R3(config)#do show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
```

```
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
```

Gateway of last resort is not set

```
10.0.0.0/24 is subnetted, 7 subnets
O    10.10.1.0 [110/2] via 10.10.3.1, 01:50:23, GigabitEthernet7/0
      [110/2] via 10.10.2.1, 01:50:23, GigabitEthernet8/0
C    10.10.2.0 is directly connected, GigabitEthernet8/0
C    10.10.3.0 is directly connected, GigabitEthernet7/0
C    10.10.4.0 is directly connected, GigabitEthernet6/0
B    10.10.5.0 [20/0] via 10.10.4.2, 00:00:00
B    10.10.6.0 [20/0] via 10.10.4.2, 00:00:00
B    10.10.7.0 [20/0] via 10.10.4.2, 00:00:00
O    192.168.1.0/24 [110/2] via 10.10.2.1, 01:50:23, GigabitEthernet8/0
O    192.168.2.0/24 [110/2] via 10.10.3.1, 01:50:23, GigabitEthernet7/0
C    192.168.3.0/24 is directly connected, GigabitEthernet9/0
B    192.168.4.0/24 [20/0] via 10.10.4.2, 00:00:00
B    192.168.5.0/24 [20/0] via 10.10.4.2, 00:00:00
B    192.168.6.0/24 [20/0] via 10.10.4.2, 00:00:00
```

La fel, în tabelul de rutare al routerului R4, datorită protocolului BGP, sunt prezente toate rutele din rețea:

```
R4(config)#do show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
```

Gateway of last resort is not set

```
10.0.0.0/24 is subnetted, 7 subnets
B    10.10.1.0 [20/0] via 10.10.4.1, 00:00:00
B    10.10.2.0 [20/0] via 10.10.4.1, 00:00:00
B    10.10.3.0 [20/0] via 10.10.4.1, 00:00:00
C    10.10.4.0 is directly connected, GigabitEthernet6/0
C    10.10.5.0 is directly connected, GigabitEthernet7/0
C    10.10.6.0 is directly connected, GigabitEthernet8/0
D    10.10.7.0 [90/3072] via 10.10.5.2, 01:55:17, GigabitEthernet7/0
      [90/3072] via 10.10.6.2, 01:55:17, GigabitEthernet8/0
B    192.168.1.0/24 [20/0] via 10.10.4.1, 00:00:00
B    192.168.2.0/24 [20/0] via 10.10.4.1, 00:00:00
B    192.168.3.0/24 [20/0] via 10.10.4.1, 00:00:00
C    192.168.4.0/24 is directly connected, GigabitEthernet9/0
D    192.168.5.0/24 [90/5376] via 10.10.6.2, 01:55:17, GigabitEthernet8/0
D    192.168.6.0/24 [90/5376] via 10.10.5.2, 01:55:17, GigabitEthernet7/0
```

Pe routerul R3 sunt redistribuite rutele BGP1 în OSPF:

```
router ospf 1
 redistribute bgp 1 subnets
exit
```

Pe routerul R4 sunt redistribuite rutele BGP2 în EIGRP:

```
router eigrp 1
 redistribute bgp 2 metric 10000 100 255 1 1500
exit
```

În rezultat, toate rutele din rețea sunt disponibile pe fiecare router. De exemplu, pe routerul R2 avem:

```
R2(config)#do show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
```

* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

Gateway of last resort is not set

```
10.0.0.0/24 is subnetted, 6 subnets
C    10.10.1.0 is directly connected, GigabitEthernet8/0
O    10.10.2.0 [110/2] via 10.10.1.1, 02:29:49, GigabitEthernet8/0
      [110/2] via 10.10.3.2, 02:29:49, GigabitEthernet7/0
C    10.10.3.0 is directly connected, GigabitEthernet7/0
O E2  10.10.5.0 [110/20] via 10.10.3.2, 01:27:33, GigabitEthernet7/0
O E2  10.10.6.0 [110/20] via 10.10.3.2, 01:27:33, GigabitEthernet7/0
O E2  10.10.7.0 [110/20] via 10.10.3.2, 01:27:33, GigabitEthernet7/0
O    192.168.1.0/24 [110/2] via 10.10.1.1, 02:29:49, GigabitEthernet8/0
C    192.168.2.0/24 is directly connected, GigabitEthernet9/0
O    192.168.3.0/24 [110/2] via 10.10.3.2, 02:29:49, GigabitEthernet7/0
O E2 192.168.4.0/24 [110/20] via 10.10.3.2, 01:27:33, GigabitEthernet7/0
O E2 192.168.5.0/24 [110/20] via 10.10.3.2, 01:27:33, GigabitEthernet7/0
O E2 192.168.6.0/24 [110/20] via 10.10.3.2, 01:27:33, GigabitEthernet7/0
```

Se poate verifica cu ping că există conexiune între oricare două host-uri din rețea. La fel, se poate vedea traseul de la sursă la destinație, folosind traceroute, de exemplu, de pe PC21 obținem:

```
C:\>tracert 192.168.6.2
```

Tracing route to 192.168.6.2 over a maximum of 30 hops:

| | | | | |
|---|------|------|------|-------------|
| 1 | 0 ms | 0 ms | 3 ms | 192.168.2.1 |
| 2 | 0 ms | 0 ms | 0 ms | 10.10.3.2 |
| 3 | 0 ms | 0 ms | 0 ms | 10.10.4.2 |
| 4 | 0 ms | 0 ms | 0 ms | 10.10.5.2 |
| 5 | 0 ms | 0 ms | 0 ms | 192.168.6.2 |

Trace complete.

Cerințe pentru realizarea lucrării de laborator №5

Se consideră configurația de rețea din Figura 1.

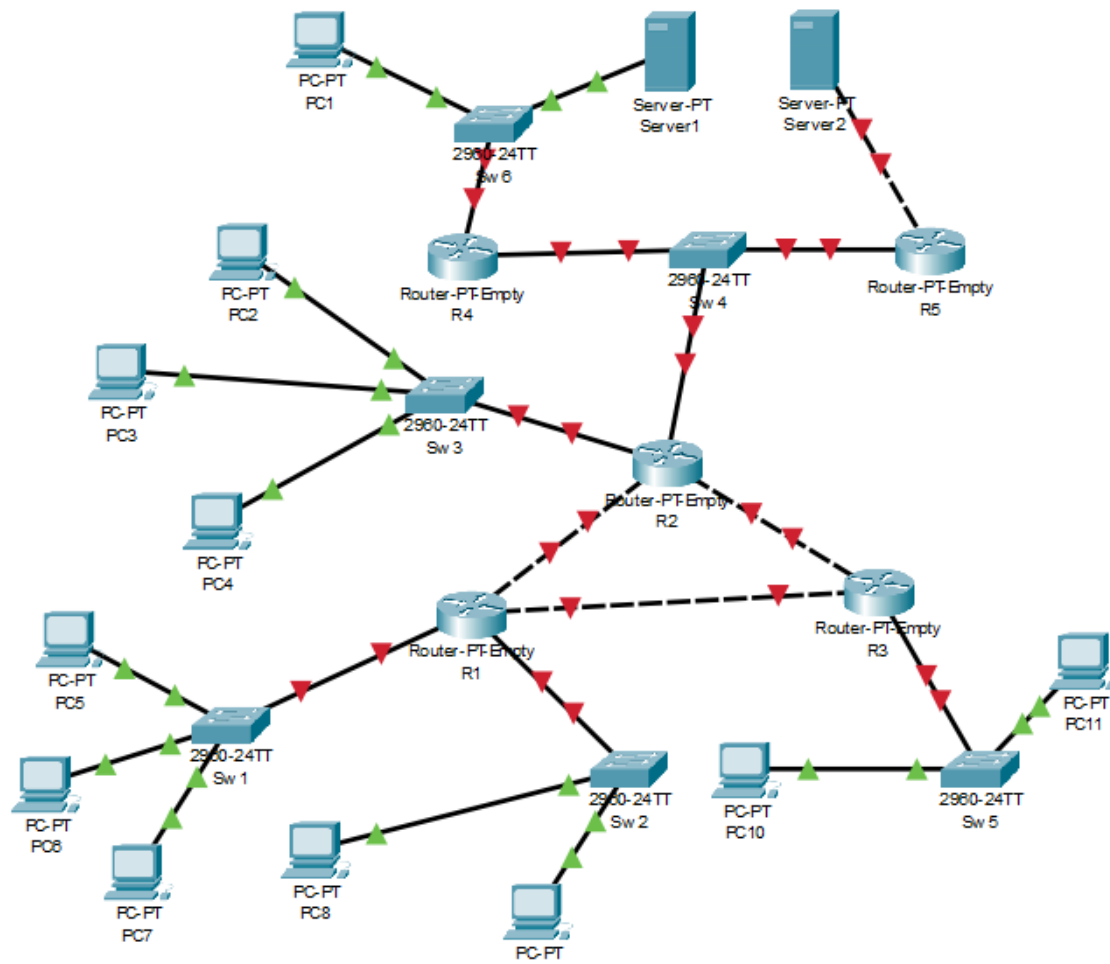


Figura 1

1. Folosind adresa de rețea
 - a) $192.168.5+k.14+k/24$,
 - b) $172.16.4+k.254-k/20$,
 - c) $10.10.16+k.0/18$,

(k - numărul de ordine al studentului în registrul grupei)

elaborați o schemă de IPv4 adrese pentru subrețelele rețelei dacă se știe că în fiecare subrețea sunt nu mai mult de 14 host-uri. Se va utiliza configurația de rețea din Figura 1 pentru a forma trei sisteme autonome AS1, AS2 și AS3, astfel încât

- dispozitivelor din AS1 li se vor atribui IP adrese ce aparțin schemei elaborate la punctul a).
- dispozitivelor din AS2 li se vor atribui IP adrese ce aparțin schemei elaborate la punctul b).
- dispozitivelor din AS3 li se vor atribui IP adrese ce aparțin schemei elaborate la punctul c).

În Cisco Packet Tracer, efectuați următoarele:

2. Folosind rutarea statică, configurați tabelele de rutare ale routerelor R1, R2, R3, R4 și R5 din sistemul autonom AS1. Salvați configurația de rețea realizată în fișierul **Nume_Prenume_Grupa_Retea4a.pkt**
3. Folosind protocolul de rutare dinamică
 - ✓ RIP (pentru k un număr din mulțimea {1,3,5,7,9,11,13,15,17,19,21,23,25}),
 - ✓ EIGRP (pentru k un număr din mulțimea {2,4,6,8,10,12,14,16,18,20,22,24}),
 configurați tabelele de rutare ale routerelor R1, R2, R3, R4 și R5 din sistemul autonom AS2. Salvați configurația de rețea realizată în fișierul **Nume_Prenume_Grupa_Retea4b.pkt**
4. Routerule sistemului autonom AS3 sunt divizate în două domenii, Area 0 și Area 1. Folosind protocolul de rutare dinamică OSPF cu *două domenii*, Area0 și Area 1 (a se vedea Figura 2), configurați tabelele de rutare ale routerelor R1, R2, R3, R4 și R5 din sistemul autonom AS3, astfel încât să fie asigurată

conexiune între oricare două dispozitive ale rețelei inițiale. Salvați configurația de rețea realizată în fișierul **Nume_Prenume_Grupa_Retea4c.pkt**

5. Se consideră rețeaua formată din trei sisteme autonome AS1, AS2 și AS3 (a se vedea Figura 3), care sunt conectate între ele prin routerul R0. Subrețelelor ce conectează routerul R0 cu AS1, AS2 și AS3 li se vor atribui corespunzător IP adresele 10.1.k.0/24, 10.1.k+1.0/24 și 10.1.k+2.0/24. În AS1 este configurată rutarea statică realizată la punctul 2. În AS2 este configurată rutarea dinamică realizată la punctul 3, iar în AS3 - rutarea dinamică realizată la punctul 4.

5.1. Realizați o redistribuire a rutelor între AS1, AS2 și AS3 fără a utiliza protocolul BGP. Salvați configurația de rețea realizată în fișierul **Nume_Prenume_Grupa_Retea4d.pkt**

5.2. Realizați o redistribuire a rutelor între AS1, AS2 și AS3, folosind protocolul de rutare dinamică BGP. Salvați configurația de rețea realizată în fișierul **Nume_Prenume_Grupa_Retea4e.pkt**

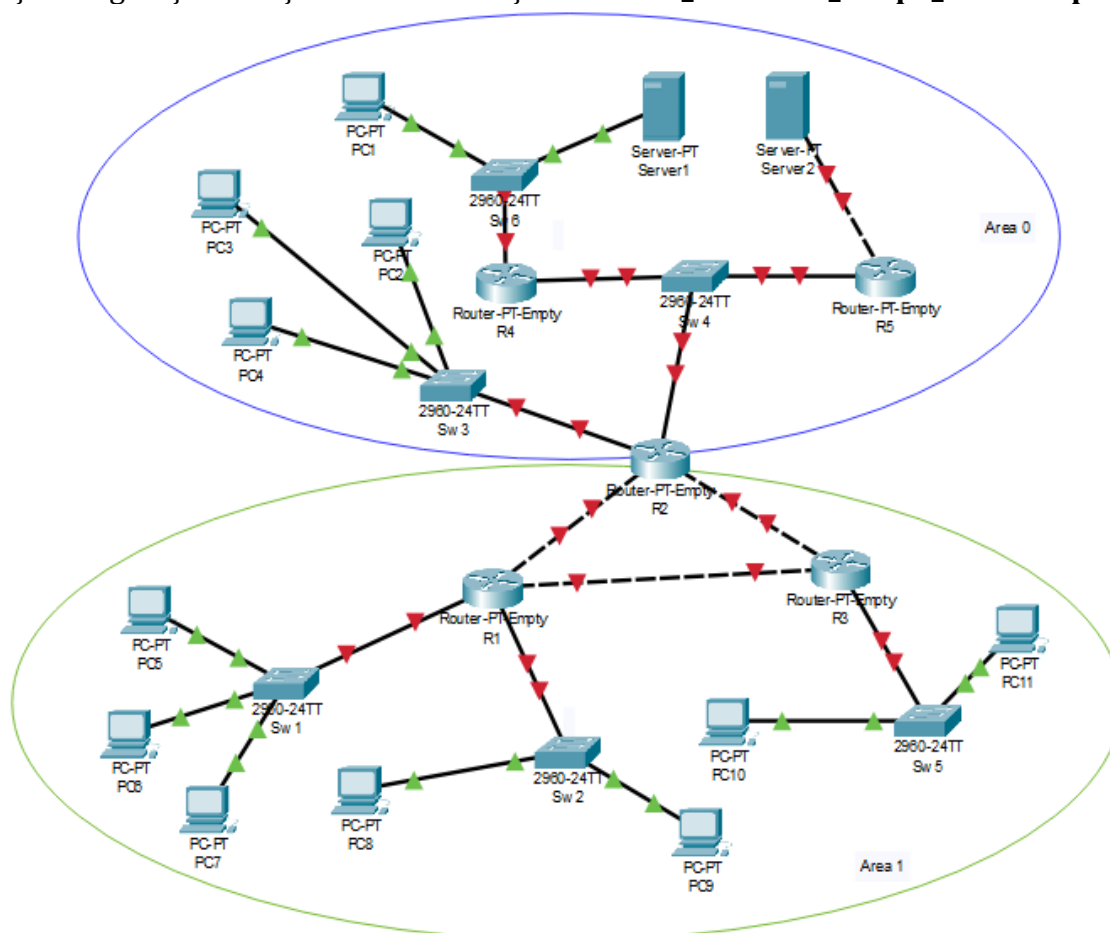


Figura 2

6. După realizarea fiecăruia din punctele 2,3,4 și 5, verificați conexiunea dintre dispozitive, folosind comanda *ping*. Folosind comanda *tracert*, generați trasee între două host-uri aleatoare din rețea. În sistemul autonom examinat întrerupeți o legătură dintre două routere conectate direct și analizați cum s-au modificat conținuturile tabelului de rutare ale routerelor din rețea. Includeți rezultatele în darea de seamă.

Realizați o dare de seamă asupra lucrului efectuat, care să conțină răspunsuri explicite la fiecare punct formulat în cerințe.

Încărcați fișierul cu darea de seamă și fișierele .pkt în mapa *Lucrarea de laborator N5* din pagina dedicată cursului de Rețele de Calculatoare a platformei educaționale moodle.usm.md.

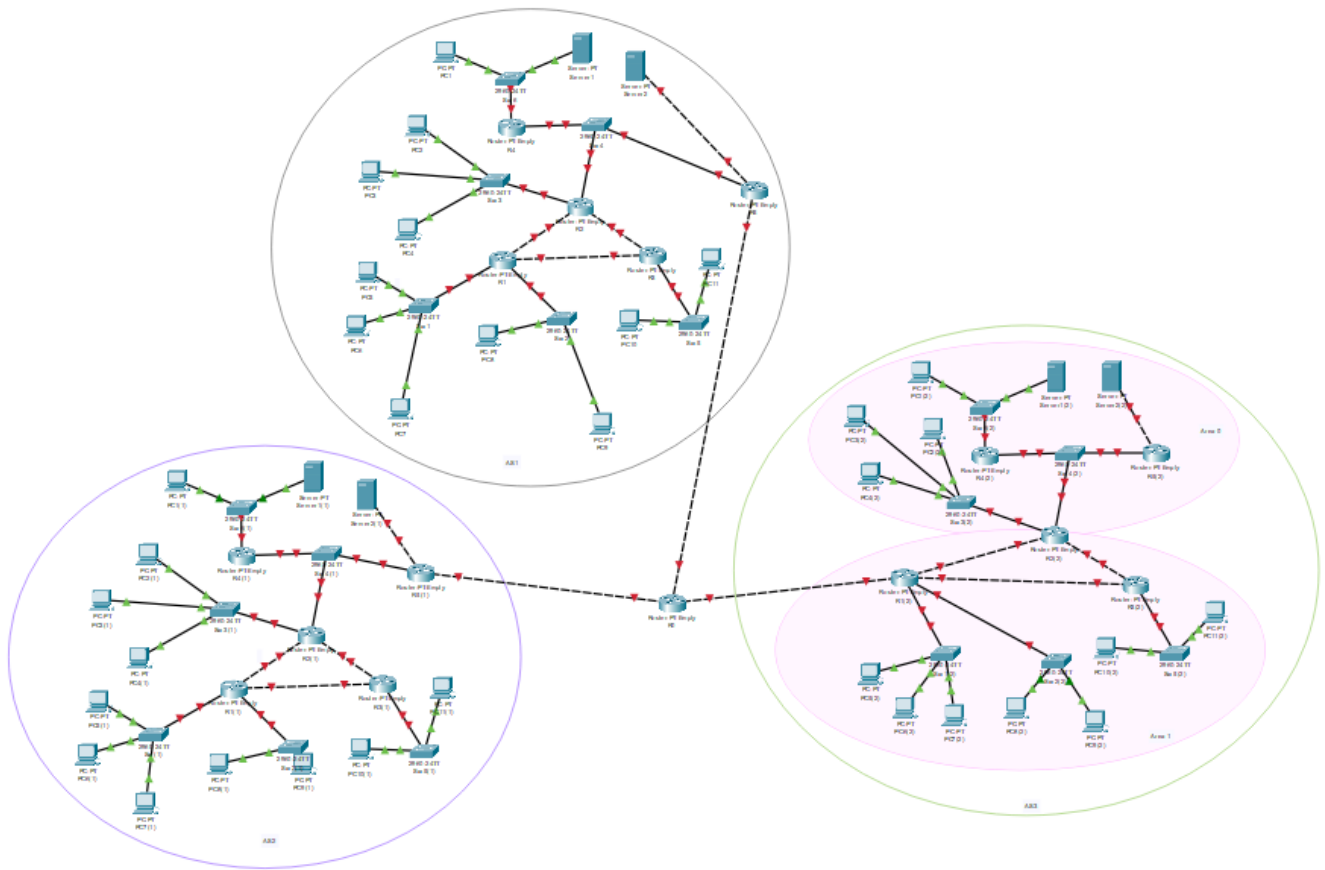


Figura 3