

## Capitolul 3: Introducere în protopoale de rutare dinamice

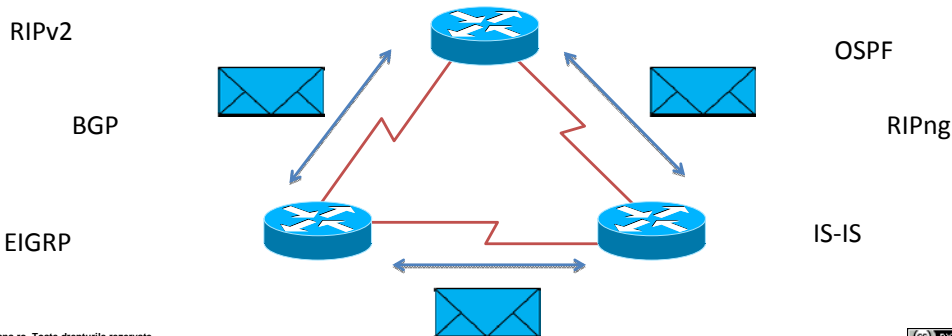


## Funcțiile unui protocol de rutare dinamic

- Un protocol de rutare are multiple funcții:

- partajarea dinamică a informațiilor între rutere
- adaptarea la schimbările din topologia rețelei
- determinarea căii cele mai bune spre fiecare destinație

Update-uri  
de rutare  
→ metrică



Un protocol de rutare este un set de procese, algoritmi și mesaje folosite pentru a schimba informații de rutare și a popula tabela de rutare cu căile cele mai bune către destinație.

Protocolele de rutare dinamice determină calea optimă în fiecare rețea folosind algoritmi interni, care apoi este adăugată în tabela de rutare. Există posibilitatea ca același protocol de rutare să furnizeze mai multe rute către aceeași destinație. Pentru ierarhizarea acestora, fiecare rută are asociată o valoare numerică numită metrică, ce indică o apreciere calitativă a drumului până la destinație.

Un rol important al protocolelor de rutare dinamice este adaptarea la schimbările apărute în topologie fără intervenția administratorului. În momentul în care este detectată o modificare a rețelei, protocolul de rutare actualizează rapid informațiile proprii și în același timp își anunță vecinii de schimbările survenite.

## Protocoale de rutare - soluții

- Un protocol de rutare eficient și scalabil trebuie să aducă soluții la următoarele probleme:
  - cum se pot menține informațiile mereu actualizate în tabelele de rutare?
  - cum se determină cea mai bună cale spre o destinație?
  - cât de repede poate să propage protocolul o modificare apărută în rețea?
  - cât de repede poate să găsească protocolul o cale alternativă spre o destinație?



Principalele probleme care trebuie rezolvate de protocoalele dinamice de rutare și care determină o anumită ierarhizare în privința performanțelor acestora sunt reprezentate de:

- modul în care se pot menține informațiile mereu actualizate în tabelele de rutare prin schimbări periodice de mesaje sau prin procese declanșate de modificări în topologie
- determinarea celei mai bune căi către destinație prin utilizarea unui algoritm intern în funcție de anumiți parametri
- viteza de propagare a unei modificări apărute în rețea și anume diminuarea timpului necesar pentru a anunța o eventuală schimbare către celelalte rutere din topologie, dar totodată și viteza de determinare a unei căi alternative spre destinație în urma procesării datelor generate de modificările apărute în topologie

## Protocoale de rutare - componente

- Structuri de date
  - tabele sau baze de date salvate în RAM
- Algoritmul intern
  - folosit pentru a determina cea mai bună spre destinație
- Mesajele protocolului de rutare
  - interschimbate de vecini
  - folosite pentru:
    - descoperirea vecinilor direct conectați
    - transmiterea de update-uri cu informații de rutare



Toate protocoalele de rutare au același scop: să determine cea mai bună cale spre fiecare rețea destinație, și în același timp să mențină informațiile de rutare actualizate de fiecare dată când în rețea are loc o schimbare. Astfel, pentru a permite ruterele să învețe dinamic rețelele nou conectate, dar și să găsească rute alternative, un protocol de rutare este alcătuit din mai multe seturi de componente dintre care se pot menționa următoarele:

- Structuri de date, reprezentând metode stocare eficientă a informațiilor
- Algoritmul folosit de protocolul de rutare, utilizat pentru a determina cea mai optimă către destinație (exemplu: algoritmul lui Dijkstra)
- Mesajele protocolului de rutare, prin care sunt descoperiți vecinii configurați cu același protocol de rutare, dar prin care este menținută o consistență a informațiilor despre topologie

## Recapitulare: Rutarea statică

- Avantaje:
  - consum minim de resurse (CPU, memorie, lățime de bandă)
  - ușor de configurat și de depanat (în rețele mici)
  - comportament complet previzibil
- Dezavantaje:
  - orice modificare se execută exclusiv manual (nu scalează)
  - nu detectează nicio schimbare în rețea
- Utilizări generale:
  - rute către diverse zone de rețea unde nu rutează protocoale de rutare
  - rute default la marginea rețelei



Rutele statice sunt introduse manual de administrator, spre deosebire de rutele dinamice care sunt generate de un protocol de rutare. O rută statică apare în tabela de rutare doar dacă interfața de ieșire asociată acesteia este activă. Spre deosebire de rutarea dinamică, rutarea statică nu folosește resurse adiționale de lățime de bandă, timp de procesor sau memorie necesare funcționării protocoalelor de rutare.

Un alt avantaj al rutării statice este efortul redus necesar pentru configurarea și administrarea rețelelor de dimensiuni mici, în care implementarea unui protocol dinamic de rutare ar însemna un consum inutil de resurse.

Dimensiunile rețelelor actuale nu permit folosirea exclusivă a rutării statice, dar sunt situații în care folosirea rutelor statice este necesară, cu scopul de a fi redistribuite apoi în protocoalele de rutare interne, sau de a se asigura conectivitatea în cazul rețelelor de tip „stub”.

## Clasificarea protocoalelor de rutare

- **Definiție:** Un sistem autonom (AS) reprezintă un grup de rutere aflate sub o administrație comună.
  - în multe cazuri, într-un AS rulează un singur protocol de rutare
  - un AS poate aparține unei companii, ISP, și este identificat printr-un număr de 16 sau 32 de biți
- **Clasificarea în funcție de AS-uri:**
  - protocoale **IGP** (Interior Gateway Protocol)
    - rutează doar în interiorul unui AS
    - RIPV2, IS-IS, OSPF, EIGRP
  - protocoale **EGP** (Exterior Gateway Protocol)
    - rutează informații între AS-uri
    - BGP

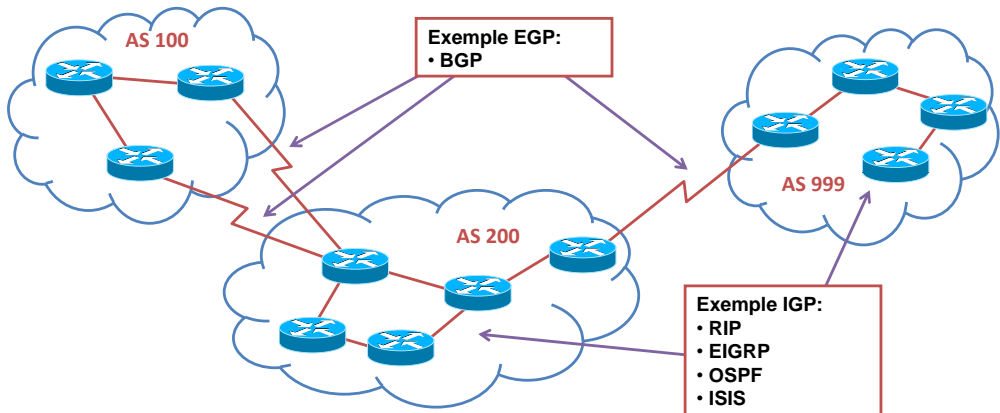
Datorită dimensiunii actuale a Internetului, toate protocoalele rutate trebuie să suporte o schemă de adresare ierarhică. Astfel rețelele pot fi grupate în sisteme autonome. Un sistem autonom (AS) reprezintă o infrastructură de rețea aflată sub o administrație comună. O administrație comună se referă la un set comun de protocoale de rutare, un set de politici de securitate și de criterii de decizie, întâlnită în cazul rețelelor interne ale companiilor sau în cazul ISP-urilor.

Protocoalele de rutare pot fi clasificate ca protocoale interioare (IGP) sau exterioare (EGP) după modul de funcționare în raport cu AS-ul. Protocoalele de rutare IGP promovează rețele în interiorul unui AS, în timp ce protocoalele de rutare EGP schimbă informații între AS-uri.

La ora actuală, BGP-ul este singurul protocol de rutare EGP utilizat în internet, fiind un protocol de tip „path vector” (specifică AS-urile prin care trebuie să treacă un pachet până la destinație).

## IGP & EGP

- Distincția între IGP-uri și EGP-uri se face pe baza scalabilității
  - un ruter ce rulează BGP trebuie să suporte întreaga tabelă de rutare a Internetului (~ 300.000 de rute)



O cerință esențială pentru un protocol de tip EGP este puterea sporită de procesare a unor tabele semnificativ mai mari decât cele întâlnite în interiorul unui AS. O tabelă de rutare în Internet, care este schimbată între două rutere de graniță din sisteme autonome diferite, poate cuprinde aproximativ 180.000 de rute.

O altă caracteristică a protocolelor de tip EGP este cea de flexibilitate, BGP-ul folosind un algoritm complex de comparare a două sau mai multe rute.

Pe de altă parte, cerințele de convergență pentru un EGP sunt destul de reduse, datorită faptului că legăturile de nucleu sunt foarte stabile. Astfel, timpul de convergență pentru BGP este de ordinul orelor mai degrabă decât al minutelor.

## Distance-vector și Link-state

- Distance vector:
  - rutele sunt descrise prin *distanță și direcție*
  - ruterele nu au o vedere completă a topologiei
  - folosesc update-uri periodice
  - update-urile nu țin cont de informațiile deja trimise anterior
  - se trimit tabelele de rutare întregi
- Link state:
  - ruterele dețin o vedere topologică completă a rețelei
  - ruterele vecine mențin adiacențe
  - update-urile sunt trimise doar când e necesar
  - update-urile pot descrie doar modificările apărute în rețea
- BGP e un caz special, considerat un protocol „path vector”

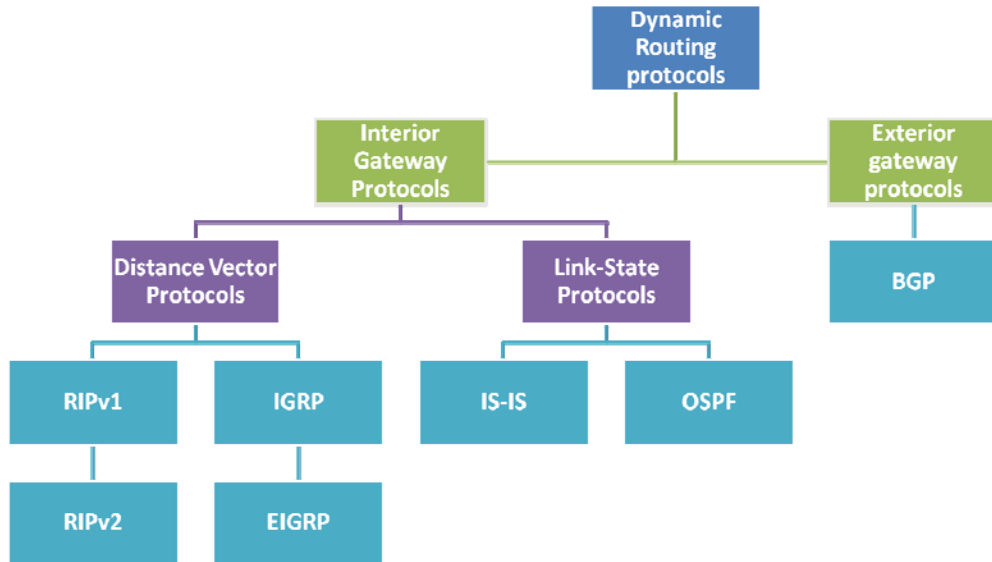
Protocoalele de rutare IGP pot fi clasificate în funcție de modul de învățare a rutelor în protocoale de rutare Distance-vector și Link-state.

Primele protocoale de rutare IGP de tip distance vector folosesc algoritmi de tip Bellman-Ford pentru a-și construi tabela de rutare. Acestea promovează rutele ca vectori de distanță și direcție, trimitând periodic ruterele vecine întreaga tabelă de rutare. Distanța poate fi numărul de hopuri (RIP), iar direcția unei rute, adresa IP a echipamentului „next-hop”.

Protocoalele de rutare link-state rezolvă câteva din limitările protocoalelor distance-vector. Ruterele trimit informații care sunt propagate către toate ruterele din topologie pe măsură ce se modifică stările link-urilor. Fiecare ruter calculează căile optime către fiecare destinație, creând un arbore de cost minim cu el însuși ca rădăcină și având astfel o imagine de ansamblu asupra rețelei.



## Clasificare (rezumat)

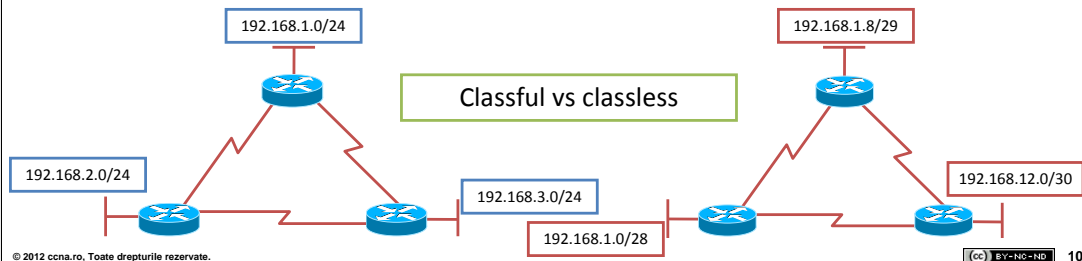


Caracteristicile relevante ale unui protocol de rutare determină dacă acesta este:

- distance-vector, link-state sau hibrid – în funcție de modul de învățare al rutelor
- interior sau exterior – în funcție de utilizare în rețele private sau Internet
- classless (permite CIDR) sau classful – permite agregarea de rute (supernetare) în schimbul de informații dintre rutere
- capabil să proceseze măști de rețea de lungime fixă sau variabilă (VLSM) – VLSM permite conservarea de adrese într-o rețea
- uniform sau ierarhic – determină scalabilitatea de adresare în rețele extinse IPv4 sau IPv6, protocoale de rutare noi fiind utilizate în rețelele IPv6 (RIPng)

## Classful vs classless

- Protocoale de rutare **classful**
  - NU TRIMIT masca de rețea în update-urile de rutare
  - exemple: RIPv1, IGRP
  - cu ce mască va introduce ruterul o astfel de rută în tabelă?
- Protocoale de rutare **classless**
  - TRIMIT masca de rețea în update-urile de rutare
  - exemple: RIPv2, EIGRP, OSPF, ISIS, BGP
  - ce avantaje prezintă un astfel de comportament?



La începutul anilor '90, o rută nu conținea mască de rețea atașată, întregul proces de rutare fiind classful. Astfel, o primă clasificare a protocoalelor de rutare se poate face în funcție de tipul procesului de rutare, classful sau classless. Odată cu trecerea timpului, dezvoltarea Internetului a dus la utilizarea tabelor de rutare classless.

În procesul de rutare classful, se evaluează primul octet din adresa IP destinație extrasă din antetul unui pachet ajuns la ruter, determinându-se astfel clasa de adrese și implicit masca de rețea.

În mod implicit, există și situația în care o rută face parte din același supernet classful al rețelei de pe interfața de pe care a fost primită, caz în care ruterul îi va atribui masca configurată pe propria sa interfață.

Protocoalele classless au multiple avantaje, cum ar fi: în tabela de rutare pot apărea rute discontinue, masca de rețea este inclusă în update-urile de rutare, suportă anunțarea rețelelor de tip VLSM.

## Convergența unui protocol de rutare

- Un protocol de rutare este în stare de **convergență** atunci când toate tabelele de rutare au o stare consistentă în raport cu topologia
- Timpul de convergență este un factor important în alegerea unui protocol de rutare
  - reflectă viteza cu care protocolul răspunde la modificările din rețea
  - timpii de convergență mari pot crea (temporar) erori logice în rețea:
    - bucle de rutare
    - black-hole routing
    - rutare suboptimală
- Convergență lentă: RIP, IGRP
- Convergență rapidă: OSPF, EIGRP, ISIS



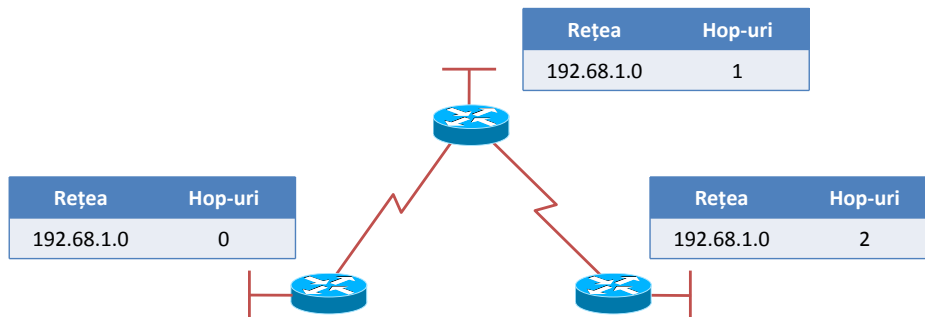
O rețea este convergentă atunci când tabelele de rutare conțin informații consistente despre rutele spre toate destinațiile existente. În general, o rețea este inutilizabilă sau dificil de controlat în timp ce converge, de aceea protocoalele de rutare urmăresc să atingă un timp de convergență cât mai mic. În cazul unei convergențe lente, datorită inconsistenței tabelelor de rutare, în rețea pot apărea erori logice: bucle de rutare, „black-hole routing” sau rutare suboptimală. Convergența depinde de fiecare ruter în parte, deoarece ruterele ce participă într-un protocol de rutare trebuie să calculeze independent căile cele mai bune spre toate destinațiile cunoscute.

Astfel, protocoalele de rutare pot fi clasificate în funcție de viteza de convergență:

- RIP și IGRP au un timp de convergență redus
- EIGRP și OSPF au un timp de convergență rapid

## Metrica unei rute

- Metrica reprezintă o valoare folosită de protocoalele de rutare pentru a decide care rute sunt mai „bune” decât altele
- Exemplu simplu de metrică bazată pe hop-count:
  - metoda este implementată în RIP



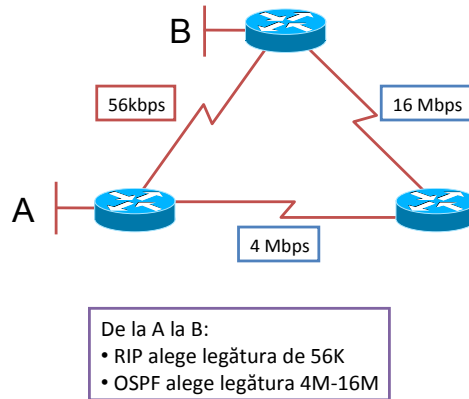
Un protocol de rutare poate să furnizeze două sau mai multe rute către aceeași destinație și astfel este necesară specificarea unui mecanism de comparare a rutelor între ele. În acest scop este folosită metrica.

Metrica unei rute reprezintă un număr, rezultat din aprecierea calității unui drum spre o anumită destinație conform unor criterii, specific fiecărui protocol de rutare. Astfel, nu are sens compararea metricilor unor rute obținute prin protocoale de rutare diferite.

În funcție de algoritmul intern de determinare a căii optime caracteristic fiecărui protocol, metrica este aleasă depinzând de parametrii mai mult sau mai puțin complecși. RIP spre exemplu folosește o metrică simplă ce determină numărul de rutere pe care ruta respectivă le-a traversat înainte de a ajunge în punctul curent. Alte protocoale mai avansate pot folosi metrice complexe care să includă: lățimea de bandă („bandwidth”) sau încărcarea unei legături („load”).

## Tipuri de metrice

- Protocoalele de rutare folosesc diferite valori pentru a măsura rutele:
  - hop count
  - bandwidth
  - cost
  - delay
  - load
  - reliability
  - diverse combinații ale acestora



Așadar, există diverse tipuri de metrice utilizate de protocoalele de rutare:

- Hop count: numărul de rutere traversate de un pachet până la destinație
- Bandwidth: este preferată calea către destinație cu cea mai mare lățime de bandă
- Load: capacitatea traficului utilizat pe o legătură
- Delay: timpul necesar transmiterii unui semnal peste o legătură
- Reliability: probabilitatea unei legături de a deveni inactivă
- Cost: o valoare care indică preferința pentru o anumită rută

Metrica folosită de RIP este cea a numărului de hop-uri, în timp ce EIGRP folosește o metrică compusă având următoarele componente: „bandwidth”, „delay”, „reliability” și „load”.

## Metrica în tabela de rutare

- Metrica este relevantă doar în raport cu protocolul de rutare care a generat-o
- Ruterele compară doar metricile aceluiași protocol de rutare

```
Router# show ip route
Gateway of last resort is not set

    172.16.0.0 255.255.0.0 is variably subnetted, 4 subnets, 2 masks
O       172.16.0.21 255.255.255.255
        [110/51] via 172.18.1.2, 00:04:56, Serial0/0/0.100
R       172.16.0.12 255.255.255.255
        [110/2] via 172.18.1.6, 00:04:56, Serial0/1/0
C       172.16.0.11 255.255.255.255 is directly connected, Loopback0
O       172.16.1.4 255.255.255.252
        [110/113] via 172.18.1.6, 00:04:56, Serial0/1/0
```

Protocoalele de rutare determină cea mai bună cale către destinație folosind ruta cu cea mai mică valoare a metricii.

Metrica asociată cu fiecare rută poate fi vizualizată în tabela de rutare utilizând comanda **show ip route**, fiind a doua valoare dintre parantezele drepte din dreptul fiecărei înregistrări. În output se pot observa rute învățate prin protocoale de rutare diferite. În cazul protocolului de rutare RIP, rețeaua destinație 172.18.1.12 se află la două hopuri distanță. Metrica are rol de apreciere a calității unui drum către o anumită destinație doar în cadrul unui anumit protocol de rutare, fiind inutilă compararea metricilor a două rute învățate prin protocoale de rutare diferite. Ierarhizarea diferitelor protocoale de rutare se realizează cu ajutorul primei valori menționate între parantezele drepte din dreptul fiecărei rute – distanța administrativă. La fel ca și în cazul metricii, va fi preferat protocolul de rutare care are asociată o valoare mai mică a distanței administrative.

## „show ip route [address]”

- Comanda **show ip route** poate afișa informații specifice pentru fiecare rută
- Pentru a afișa doar rutele unui anumit protocol se folosește:
  - **show ip route [protocol]**

```
Router#show ip route 10.0.0.1

Routing entry for 10.0.0.1/32
  Known via "isis", distance 115, metric 20, type level-1
  Redistributing via isis
  Last update from 223.191.255.251 on Fddi1/0, 00:00:13 ago
  Routing Descriptor Blocks:
  * 10.22.22.2, from 223.191.255.247, via Serial2/3
    Route metric is 20, traffic share count is 1
    223.191.255.251, from 223.191.255.247, via Fddi1/0
    Route metric is 20, traffic share count is 1
```

Comanda **show ip route** poate fi apelată împreună cu un parametru suplimentar care specifică o adresă IP a unei rețele destinație din cadrul tabelii de rutare astfel: **show ip route [address]**. Din outputul afișat, se pot observa informații detaliate despre ruta primită ca parametru:

- Adresa IP a rețelei, împreună cu masca de rețea atașată
- Modul de învățare a rutei, în cazul de față, prin protocolul dinamic de rutare IS-IS, împreună cu informații specifice protocolului menționat
- Distanța administrativă și metrica rutei
- Interfața de ieșire prin care se vor trimite pachetele a căror IP destinație aparține rețelei definite în cadrul rutei

## Load balancing

- Equal-cost load balancing
  - abilitatea unui ruter de a folosi multiple căi spre aceeași destinație, dacă acestea au metrice egale
  - majoritatea protocoalelor de rutare pot introduce căi multiple în tabela de rutare
- Unequal-cost load balancing
  - rutele spre destinație pot avea metrice diferite, dar într-un interval predefinit

```
Router#show ip route
Gateway of last resort is not set

R    192.168.6.0 [120/1] via 192.168.2.1, 00:00:24, Serial0/0/0
      [120/1] via 192.168.4.1, 00:00:26, Serial0/0/1
```

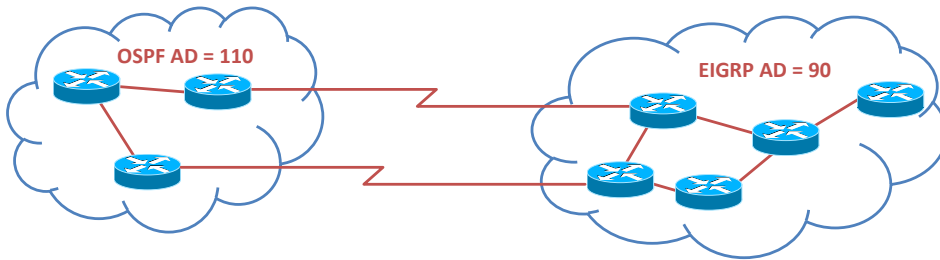
După procesul de construire a tabelii de rutare există posibilitatea existenței mai multor rute către aceeași destinație având atașate aceeași metrică. În acest caz, ruterul va distribui traficul în mod egal către toate interfețele care au asociate rute cu metrice egale. Procesul desfășurat este cunoscut sub denumirea de „equal-cost load balancing” și poate fi vizualizat în tabela de rutare atunci când două sau mai multe rute au asociate aceleași rețele destinație.

O caracteristică importantă a anumitor protocoale de rutare cum este protocolul EIGRP este faptul că poate echilibra traficul pe mai multe rute de cost diferit, ținând cont de o valoare a variației metricilor folosite în procesul de „load balancing”. Mai precis, vor fi introduse în tabela de rutare toate rutele cu metrica mai mică decât valoarea obținută prin înmulțirea valorii variației cu metrica rutei de cost minim.



## Distanța administrativă (AD) a unei rute

- Distanța administrativă este o valoare ce reprezintă gradul de „preferință” pentru originea unei anumite rute
  - rutele statice, direct conectate și cele dinamice au valori AD diferite
  - valoarea mai mică este preferată
  - metricele sunt comparate doar pentru rute cu AD-uri egale



Două sau mai multe protocoale de rutare diferite pot să furnizeze câte o cale către aceeași destinație, cu aceeași adresă și mască de rețea atașată. Criteriul principal de diferențiere a rutelor generate este reprezentat de distanța administrativă (AD). Ruta cu valoarea distanței administrative mai mică este preferată și introdusă în tabela de rutare. Așadar, distanța administrativă este o valoare ce reprezintă gradul de „preferință” pentru originea unei anumite rute. În practică, aceasta determină o ierarhie bine definită a tuturor modurilor posibile prin care o rută poate fi dobândită.

În cadrul aceluiași protocol de rutare, diferențierea rutelor se va face pe baza valorii metricii, ruta cu o valoare mai mică a acesteia fiind introdusă în tabela de rutare.

## AD în tabela de rutare

- Fiecare rută dinamică afișează [AD / metrică]
- Valorile aparțin intervalului [0..255]
- Doar rutele direct conectate au AD = 0 (și nu poate fi schimbat)
- Un AD = 255 indică o rută ce nu va fi inclusă niciodată în tabela de rutare

```
R1#show ip route

Gateway of last resort is not set

  10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
D       10.7.7.2/32 [90/21024000] via 10.1.1.1, 00:14:05, Serial0
D       10.7.7.0/24 [90/21024000] via 10.1.1.1, 00:14:05, Serial0
C       10.1.1.0/24 is directly connected, Serial0
C       10.1.1.1/32 is directly connected, Serial0
C       192.168.0.0/24 is directly connected, Ethernet0
```

În cadrul tabelii de rutare, distanța administrativă este reprezentată de prima valoare numerică afișată între paranteze drepte. Acest număr va aparține intervalului 0-255, valoarea cea mai mică fiind preferată pentru alegerea rutei către destinație. Astfel, ruta cu distanța administrativă 0, asociată rutelor direct conectate, va fi tot timpul aleasă înaintea altor rute existente. Pe de altă parte, o rută cu o valoare AD („administrative distance”) egală cu 255 nu va fi instalată niciodată în tabela de rutare. După rutele direct conectate vor fi preferate rutele statice, deoarece acestea au atașată o distanță administrativă implicită egală cu 1, iar apoi protocoalele dinamice de rutare.

## Valori AD standard

Sursa rutei	Distanța administrativă
Direct conectată	0
Statică	1
Rută sumarizată EIGRP	5
Rută BGP externă	20
Rută EIGRP internă	90
IGRP	100
OSPF	110
ISIS	115
RIP	120
Rută EIGRP externă	170
Rută BGP internă	200

Valorile standard pot fi modificate... doar la CCNP 😊

Distanțele administrative pentru cele mai utilizate protocoale de rutare sunt precizate în tabelul alăturat. Se remarcă valorile pentru următoarele tipuri de rute:

- Direct conectate: 0
- Statice: 1
- RIP: 120
- EIGRP: 90
- OSPF: 110

În funcție de valorile AD-urilor se pot efectua următoarele concluzii: AD RIP > AD EIGRP deoarece EIGRP este mai performant decât RIP, AD IGRP > AD EIGRP deoarece EIGRP a fost dezvoltat ca o îmbunătățire a IGRP. Rutele externe vor avea o distanță administrativă mai mare datorită faptului că sunt provenite din alte domenii de rutare.

## Rute direct conectate și statice

- Atenție la rutele statice date prin interfața de ieșire!
  - ele apar în tabela de rutare ca fiind „directly connected”
  - dar au AD = 1

```
R1#show ip route
S    192.168.0.0/24 is directly connected, Serial0/0

R1#show ip route 192.168.0.0
Routing entry for 192.168.0.0/24
Known via "static", distance 1, metric 0 (connected)
Routing Descriptor Blocks:
* directly connected, via Serial0/0
  Route metric is 0, traffic share count is 1
```

Rutele statice sunt introduse manual de către administratorul de rețea cu scopul de a configura o cale optimă către destinație. Valoarea implicită a distanței administrative a rutelor statice este 1, deci, după rutele direct conectate care au distanța administrativă egală cu 0 sunt preferate rutele statice.

Rutele statice pot fi configurate utilizând adresa IP „next-hop” sau interfața de ieșire, în ambele cazuri având distanța administrativă implicită, și anume 1. Totuși, în cazul rutelor statice configurate folosind interfața de ieșire, valoarea distanței administrative nu este afișată la introducerea comenzii **show ip route**. Rutele direct conectate vor apărea în tabela de rutare imediat după configurarea adreselor IP pe interfețe și activarea acestora. Distanța administrativă fiind 0, va fi întotdeauna ruta preferată.

În cazul în care o rută statică configurată nu apare în tabela de rutare se va verifica dacă interfața de ieșire este configurată corect și activată.

## Rezumat

- Funcțiile protocoalelor de rutare
- Interior Gateway Protocols (IGP)
- Exterior Gateway Protocols (EGP)
- Classful și classless
- Distance-vector și Link-state
- Distanța administrativă (AD)
- Metrica

1. Prin ce metode se pot menține informațiile mereu actualizate în tabela de rutare?
2. Care sunt categoriile de diferențiere a protocoalelor de rutare
3. Care sunt erorile logice care pot apărea într-o rețea în cazul unui timp mare de convergență?
4. Care este criteriul principal de diferențiere a protocoalelor de rutare? Dar a rutelor în cazul unui anumit protocol ?
5. Care este distanța administrativă a rutelor direct conectate? Dar a rutelor configurate static?