Rețele de Calculatoare și Internet

▶Proiect

Protocolul BGP
(Border Gateway Protocol)

Coordonator:

Prof.Dr.Ing. Ștefan Stăncescu

Masterand:

Deaconescu Claudia-Daniela

Cuprins

Introducere BGP rutarea ierarhică……3
Capitolul 1 Algoritmi și probleme de rutare……..…….……....4
1.1 Algoritmi de rutare și metrica acestora……..4
1.2 Comparații de performanță pentru diferite configurații (RIPv1 și RIPv2)…..……..8
1.2.1 CIDR și VLSM rutarea dinamică……..9
1.3 Topologia generic ierarhică în Internet a protocolului BGB……12
1.4 Distance Vector Protocol (DVP)……..14
1.5 Ierarhia și mesajele BGP, neighbor acquisition……16
1.5.1 Evoluția protocolului BGP și metrica de rutare a algoritmului……17
Capitolul 2 Traficul si problemele BGP, rezolvarea acestora……..……...…..20
2.1 Schimbul de informații de rutare în cadrul protocolului BGP……20
2.2 Configurarea politicilor BGP locale și sincronizarea…… 21
2.3 Problemele BGP și rezolvarea acestora ……..23
2.3.1 Legăturile de la egal la egal (peer establishment)……..23
2.3.2 Rute lipsă (missing routes)……26
2.3.3 Selecția incompletă a traseului……...26
2.3.4 Probleme legate de bucle și convergență……29
2.3.5 Bouncing problem……...30
2.4. Securitatea……...31
Concluzii……33
Bibliografie……35

Introducere

- BGP rutarea ierarhică -

Protocolul BGP (Border Gateway Protocol) este "motorul" Internetului curent, cu alte cuvinte conceptul de Internet este privit ca o suma de Sisteme Autonome (Autonomous System-AS) care schimbă rute prin intermediul protocolului BGP, iar în interiorul AS-urilor se vor folosi alte protocoale de rutare interne. Scopul acestui procedeu este de a proteja toate rețelele interne care nu trebuie cunoscute în exterior, prin sumarizarea rețelelor ce pot fi cunoscute în rețele cât mai puține și cu o mască de rețea cât mai scurtă.

O reţea cu o singură ieşire se mai numeşte single-homed network sau stub network. O reţea cu mai multe ieşiri, care satisface definiţia AS, se cheamă multi-homed. Fiecare AS cere şi primeşte un număr de AS unic.

BGP poate fi folosit în interiorul unui AS, în care caz se numeşte Interior BGP (IBGP), sau între AS-uri, în care caz se numeşte EBGP. De notat că IBGP şi EBGP nu sînt 2 protocoale diferite; BGP va detecta automat, pe baza numărului AS a vecinului, dacă rulează în mod interior sau exterior. 2 rutere care comunică între ele prin BGP realizează o conexiune TCP pe portul 179. Prin comparaţie, OSPF foloseste doar IP, iar RIP trimite updates folosind UDP pe portul 520.

Conținutul mesajelor BGP include mai multe câmpuri dintre care un header, care are în componență patru tipuri de mesaje și anume: OPEN- pentru stabilire a conexiunii, KEEPALIVE – un mesaj de 19 octeţi, care reprezintă lungimea minimă a headerului+tipului, fără alte date care este utilizat pentru menținerea în funcție a conexiunii, NOTIFICATION pentru semnalarea unei erori și UPDATE pentru informare.

Mesajele UPDATE sînt cele mai importante, căci conţin cele mai multe date şi pe baza lor BGP crează o imagine (un graf) al întregii reţele, pe baza căruia se stabileşte ruta optimă către orice destinaţie.

Prin specificarea AS-number BGP cunoaşte AS-ul propriu; în continuare, în funcţie de AS-number specificat pentru fiecare neighbor, BGP determină singur dacă respectivul este în acelaşi AS (şi deci protocolul va rula ca IBGP) sau în alt AS (EBGP).

În afara acestei configuraţii obligatorii, BGP suportă opţiuni de configurare foarte complexe, în funcţie de politicile pe care administratorii doresc să le implementeze (în general prin manipularea atributelor rutelor); vom studia doar câteva din posibilităţi pe baza unor scenarii.

Capitolul 1- Algoritmi și probleme de rutare

1.1 Algoritmi de rutare și metrica acestora

Protocolul de rutare este alcătuit dintr-un algoritm plus un mecanism de schimb de mesaje între nodurile rețelei. Rețeaua se poate abstractiza printr-un graf orientat/neorientat.
Problema de rutare: gasirea unui arbore cu cost minim de la orice nod considerat ca sursa către orice nod considerat ca destinatie: costul oricarei cai = minim (SPT = Shortest path tree") și cost total al arborelui = minim (arbori de tip "Steiner").
Criterii de cost/metrica: metrica simpla (1 criteriu), compusă și mai multe criterii.
Exemple de metrici:
- statice: nr. de noduri traversate, 1/Banda_link, cost administrativ, cost total al unui arbore,
- dinamice (costul se modifica odata cu incracarea rețelei cu trafic): grad de incarcare a unui link, intârziere pe link/ ruta, număr mediu de erori de transmisie per link.

Moduri de lucru ale protocolului:

- Proactiv:
Algoritmul lucreaza in permanenta ("background") si caluleaza rute;
Info de rutare exista in permanenta in tabelul de fwd.
Rutele pot sa nu mai fie actuale daca intervalul temporal de calcul e mare in comparatie cu modificarile de topologie sau costuri (ex. retele cu mobilitate)
Daca perioada de calcul este f. atunci rezultă un "overhead" cu trafic de control
- Reactiv ("on demand")
Ruta se calulează la cererea unei stații
De ex. în rețele ad-hoc
- De tip mixt
Alg. poate rula în stil proactiv în anumite condiții și în mod reactiv în altele
Clasificare d.p.d.v al cunoașterii topologiei:
- vectori de distanță ("distance vector")
un nod nu cunoaște topologia completă a rețelei ci numai informații de tip "distanțe" între anumite puncte.
- cu stări ale link-urilor ("link state")

Probleme suplimentare în calculul rutelor:
- mobilitatea;
- inteferențe – în radio;
- refacerea rutelor deteriorate
- problema link-urilor asimetrice

- problema cailor diferite în cele două sensuri.

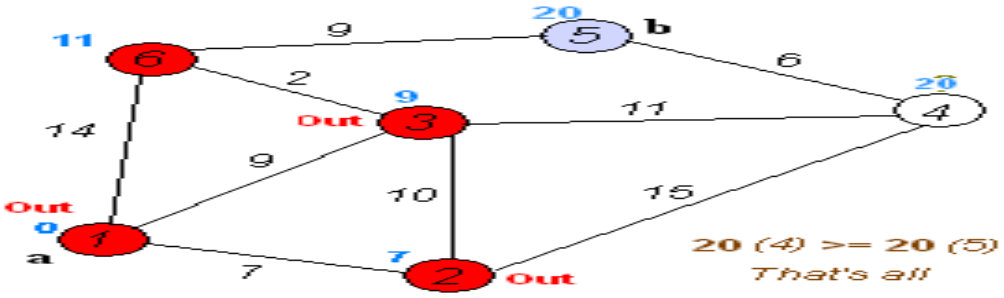
Caracteristici dorite ale protocolului: convegența, optimalitate, complexitate redusă, robustete, extensibilitate, echitate, fiabilitate, etc. (unele sunt contradictorii)

<u>Algoritmii de rutare pot fi grupați în două clase mari:</u>

1. Algoritmii **neadaptivi** nu își bazează deciziile lor de rutare pe alte masurători sau aproximări a topologiei curente sau a traficului. Decizia luată pentru a folosi o ruta intre I si J este computerizată în avans, într-un mediu off-line si downloadată în rutere când se instalează rețeaua. Această procedură este cateodată numită rutare statică. Deoarece nu raspunde atunci când cade un link din rețea, rutarea statică este des folosită in situații în care ruta aleasă este clară si vizibilă.
2. Algoritmii **adaptivi,** în coparație cu cei neadaptivi, își modifică deciziile de rutare pentru a reflecta schimbări în topologie, iar cateodată chiar schimbări în trafic. Acești algoritmi dinamici diferă în funcție de unde isi iau informațiile, când își schimbă rutele si ce metrică este folosită pentru optimizare.

A. <u>Primul Algoritm cu calea cea mai scurtă- Shortest Path First Algorithm</u>

Algoritmul SPF este schema de rutare standardizata a Internetului din ziua de azi. El minimizeaza o singura metrica aditivă care este asignată ca un cost către fiecare link. Aceasta metrică poate fi simplu numarată ca un hop-count sau întârzierea propagarii. Un lucru important asupra schemei SPF este că nu definește rute în cazul in care câteva căi către aceeași pereche sursă - destinație au cumulat aceeași metrică. În loc de optimizarea unei metrici secunde, în acest caz, SPF alege o cale aleator. Această proprietate nefericită este adresată de către o subsecvență a schemei de rutare. Complexitatea implementarii algoritmului SPF este bazată pe algoritmul Floyd care gasește cea mai scurtă cale între toate nodurile este $O(v^3)$. Forma căii între un nod si toate celelalte poate fi găsită cu algoritmul Dijkstra care are complexitatea $O(v^2)$.



B. <u>Lărgimea primului algoritm cu calea cea mai scurtă- Widest Shortest Path First Algorithm</u>

Algoritmul WSPF (Widest Shortest Path First) este similar cu schema SPF. Singura diferență este că o metrică secundă este considerată, daca prima metrica duce către căi alternative. În acest caz calea cu maximul minim al capacitații este aleasă. Capacitatea nu reprezintă o metrică aditivă. Capacitatea unei căi este determinată de link-ul "bottleneck" care este minimul capacitatii tuturor link-urilor din cale. A

doua metrică este testată cu o singura verificare adițională, care nu schimbă complexitatea algoritmului SPF.

C. Primul algoritm cu calea cea mai scurtă a lărgimii- Shortest Widest Path First Algorithm

Algoritmul SWPF (Shortest Widest Path First) este o altă variantă a schemei SPF. Prima metrică este maximul minim al capacității pe o cale. A doua metrică este propagarea întârzierii sau se aplică hop-count-ul. Ținta algoritmul SWPF este de a imprăștia trafic încărcat în întreaga rețea. O dată ce prima metrică nu este aditivă, implementarea SWPF este mai complicată decât SPF sau WSPF. Se propune o dublă aplicare a algoritmului Dijkstra pentru fiecare pereche sursă - destinație. Apoi, următorii pași se fac independent pentru fiecare pereche sursă-destinație.

D. Algoritmul cu căi maxime- Maximally Disjoint Paths Algorithm

Algoritmul MDJ (Maximally Disjoint Paths) diferă de schemele de rutare descrise anterior deoarece el gasește mai mult de o cale pentru fiecare pereche sursă – destinație. El generează o forma de arbore de la nodul sursă către fiecare nod destinație bazat pe un algortm simplu SWPF. Apoi, o pereche cu un număr maxim de bucați este calculată prin adaugarea mai târziu de link-uri comune la metrica de rutare. Acest pas este repetat pentru fiecare sursă posibilă. Schema MDJ este eficient aplicată în rețelele MPLS care implementează protecția căii. Aici se folosește și a doua cale. Algoritmul MDJ este aplicat static. Complexitatea lui este $O(v^2*e)$, unde e>=v.

RIP (Routing Information Protocol) version 1

Cu trecerea anilor, protocoalele de rutare au evolat datorită creșterii necesității de rețele mai mari și mai complexe. Primul protocol utilizat a fost protocolul RIP (Routing Information Protocol). Deși este primul protocol apărut și pe lângă el au apărut si alte protocoale mult mai eficiente, acesta încă se bucură de 'popularitate' deoarece este simplu și foarte raspândit. Deși este un protocol de rutare relativ simplu, acesta ne poate ajuta pentru întelegerea mai ușoara a altor protocoale de rutare.
În tabelul de mai jos am exemplificat o clasificare a protocoalelor de rutare. Se observă că protocolul RIPv1 este bazat pe algoritmul Bellman-Ford (Distance Vector Alghoritm).

	Interior Gateway Protocols				Exterior Gateway Protocols
	Distance Vector Routing Protocols		Link State Routing Protocols		Path Vector
Classful	RIP	IGRP			EGP
Classless	RIPv2	EIGRP	OSPFv2	IS-IS	BGPv4
IPv6	RIPng	EIGRP for IPv6	OSPFv3	IS-IS for IPv6	BGPv4 for IPv6

[sursa 7]

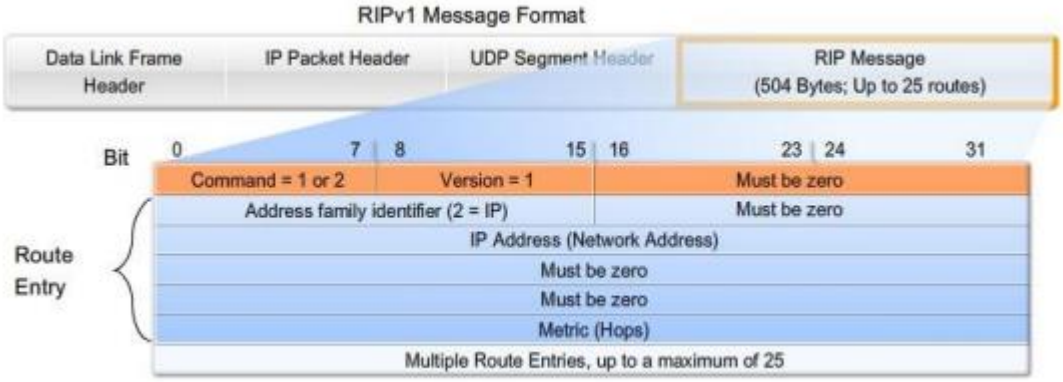
Caracteristici:

Protocolul RIP are următoarele caracteristici:
- RIP este un protocol de rutare bazat pe algoritmul “distance vector” - RIP folosește numărarea de hopuri ca fiind singura metrica pentru alegerea căii de transmisie a pachetelor - Nu se poate ajunge la ruterele care se afla la o distanță mai mare 15 hopuri - Mesajele de broadcast sunt la interval de 30 de secunde”[1]

Formatul mesajului RIP:

RIP Headear “Trei câmpuri sunt specificate în cei patru biți ai porțiunii de headear. Câmpul “Command” specifică tipul mesajului, câmpul “Version” este setat la 1 pentru RIP version 1, iar al treilea câmp, “Must be zero”, ne lasă loc pentru viitoarele expansiuni ale protocolului.”[1]

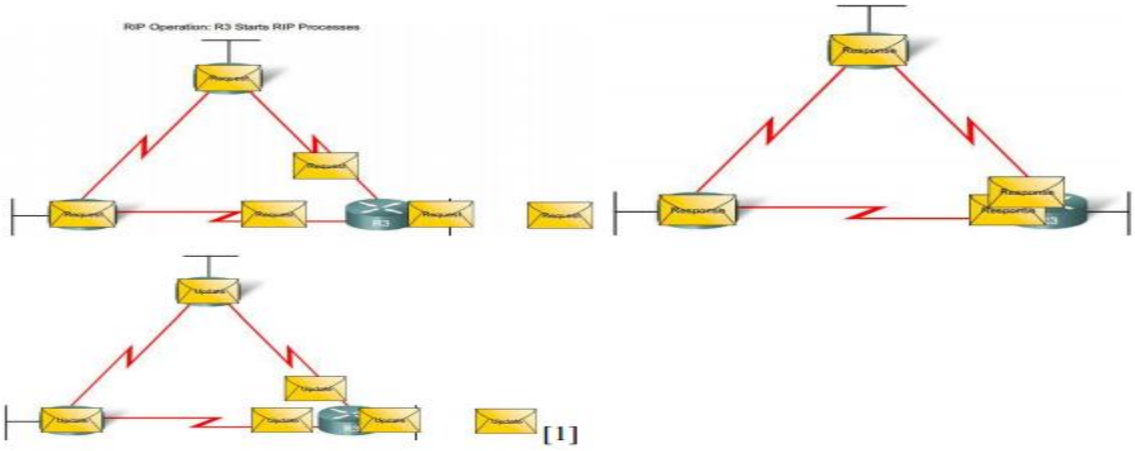
“Porțiunea mesajului numită “route entry” conține trei câmpuri cu următoarele semnificații: “Address family identifier” (este setata la 2 pentru IP, iar setarea 0 este in cazul in care ruterul are nevoie de o tabelă de rutare completă), adresa IP si metrica. Aceasta portiune din route entry reprezinta o ruta de destinație cu metrica sa. Un update al lui RIP poate contine pana la 25 de rute de intrare. Dimensiunea maxima este de 504 bytes, fara sa include headerele de IP sau de UDP. “[1]



[sursa 7]

RIP Request/Response Process

RIP foloseste două tipuri de mesaje specificate in câmpul “Command”: “Request message” si “Response message”. “Descrierea procesului: fiecare interfata configurată RIP trimite un mesaj de tip ‘request’ la pornire, cerând ca toți vecinii RIP să trimită tabela lor de rutare completă. Un mesaj de tip ‘response’ este trimis înapoi de către vecinii ‘enabled’. Când ruterul care a facut cererea primește mesajul ‘response’, evaluează fiecare rută de intrare. Daca ruta de intrare este nouă, atunci o instalează în tabela sa de rutare. Daca ruta există în tabelă, ea este înlocuită doar în cazul în care are un număr de hopuri (metrica) mai mică. Dar și ruterul care a trimis cererea de ‘request’ trimite un ‘update’ vecinilor pentru a le trimite tabela sa de rutare. În figurile de mai jos am exemplificat acest proces:”



1.2 Comparații de performată pentru diferite configurații

RIP este un protocol 'classfull' – RIPv1 nu trimite informatii cu privire la subnet mask in mesajele de update. Prin urmare, un ruter foloseste subnet mask-ul configurat pe interfata locala, sau aplica subnet mask-ul default specific clasei adresei IP.

Distanta administrative (AD) specifica protocolului RIP este de 120. Daca comparam RIP-ul cu alte protocoale de rutare in ceea ce priveste distanta administrativă, acesta este cel mai putin preferat. IS-IS, OSPF, IGRP si EIGRP are distante administrative default mai mici.

Limitarile protocolului

"Protocolul nu rezolva toate problemele posibile de rutare, ci are si niste limitari:

- Protocolul este limitat la retele a caror lungimi nu depasesc 15 hopuri. De aceea nu este recomandat in retele mari.
- Protocolul depinde de "counting to infinity" pentru a rezolva anumite situatii neobisnuite. Daca sistemul de retele are cateva sute de retele si o bucla de rutare a fost formata continandu-le pe toate, rezolutia buclei necesitand mai mult timp sau latime de banda. O astfel de bucla va consuma mult din latimea de banda a retelei inainte ca ea sa fie rezolvata. Se crede ca in realitate, acest lucru nu ar fi o problema exceptand liniile lente. Chiar si atunci, problema ar fi putin neobisnuita, deoarece sunt luate masuri pentru a preveni aceste probleme in majoritatea cazurilor.
- Acest protocol foloseste metrici fixate pentru a compara rute alternative. Acest lucru nu este recomandat pentru situatii in care ruterele trebuie sa fie alese pe baza parametrilor de timp real cum ar fi intarzierea masurata, reliabilitatea sau incarcarea. Evident ca aceste extensii pentru acceptarea acestor tipuri de metrici sunt pentru a introduce instabilitatea pentru care protocolul nu este proiectat pentru a-i face fata. "

1.1.2 CIDR și VLSM rutarea dinamică

CIDR (Classless InterDomain Routing) se referă la renunţarea la alocarea de adrese conform împărţirii pe clasele A,B,C. În primii ani ai Internetului, anumite organizaţii sau companii cum ar fi, IBM, HP, Apple, MIT şi altele au primit clase A (16 milioane de adrese) pe care în mod evident nu le folosesc integral; este dificilă acum retragerea şi realocarea acestor adrese. De aceea, se folosesc subneturi de lungime variabilă şi se alocă fiecărui client/ fiecărei organizaţii atîtea adrese cîte are nevoie. De exemplu, o organizaţie care are nevoie de 1000 de adrese primeşte 4 reţele de lungime /24 (rezultînd un supernet de lungime /22 dacă cele 4 reţele sînt reţele clasă C contingue); aceste reţele pot fi şi subreţele ale unei clase A de exemplu, în acest caz combinarea a 4 subneturi /24 într-un subnet formează tot un subnet /22, denumirea de supernet fiind de obicei rezervată "încălcării" unei limite de clasă. CIDR este sinonim cu adresarea şi rutarea classless; implementarea sa a necesitat introducerea obligatorie a măştii de reţea asociată oricărei adrese, şi transmiterea acesteia şi de către protocoalele de rutare; de exemplu, RIP v1 a trebuit înlocuit cu RIP v2.

Inainte de 1981, adresele IP foloseau doar primii 8 biti pentru a arata portiunea de network, limitand Internetul – pe atunci cunoscut sub numele de ARPANET – la 255 de retele. Bineinteles ca s-a ajuns la concluzia ca numarul de retele este mult prea mic. In 1981, RFC 791 a modificat IPv4 la o adresa de 32 de biti pentru a permite existenta a trei clase de retele: clasa A, clasa B si clasa C. Clasa A de adresa utilizeaza 8 biti pentru network, clasa B foloseste 16 biti si clasa C utilizeaza 24 de biti. Acest format a devenit cunoscut ca fiind o adresare IP classful. Desi problema celor 256 de retele a fost rezolvata a aparut o alta problema: spatiul pentru adresele IP incepea sa fie prea mic. Ca raspuns la aceasta problema, IETF-ul (Internet Engineering Task Force) a introdus CIDR-ul (Classless Inter-Domain Routing) care utilizeaza VLSM-ul (Variable Length Subnet Masking) pentru a ajuta in conservarea spatiului de adrese. O data cu introducerea CIDR-ului si VLSM-ului, administratorii de retea au fost nevoiti sa utilizeze skilluri suplimentare pt subnetizare.VLSM-ul reprezinta o simpla subnetizare a subnet-ului. Reteaua poate fi subnetizata pe mai multe nivele. In concordanta cu subnetizarea, a devenit posibila sumarizarea unei colestii mai mari de retele classful intr-o ruta agregata, sau supernet.
"Conform ISC in 08/1981 existau 213 host-uri ajungand ca in 07/2012 sa fie 908.585.739." [2]

RIP (Routing Information Protocol) version 2

"RIP Version 2 (RIPv2) este definit in RFC 1723. Este primul protocol de rutare classless. Desi RIPv2 este un protocol de rutare potrivit pentru unele medii, el are o popularitate scazuta in comparatie cu alte protocoale, ca de exemplu EIGRP, OSPF si IS-IS, care ofera mai multe posibilitati si sunt mai scalabile."[1] Desi sunt mai putin populare decat alte protocoale de rutare, ambele versiune de RIP sunt potrivite in anumite situatii. Desi RIP nu are capacitatile multor protocoale ulterioare, simplitatea si utilizarea pe scara larga in multiple sisteme de operare il face candidatul ideal pentru retele mici si omogene – in special in mediile UNIX. RIPv1 este un protocol "classful". RIPv2 este un protocol "classless" adica include subnet mask-ul in update-urile de rutare. Din aceasta cauza, RIPv2 este mai compatibil cu mediile de rutare moderne. "RIPv2 este de fapt o imbunatatire a caracteristicilor lui RIPv1. Cateva din aceste imbunatatiri sunt:

- Adresa urmatorului hop este inclusa in update-urile de rutare.
- Foloseste o adresa de multicast pentru trimiterea update-urilor.
- Optiunea de autentificare este valabilă."

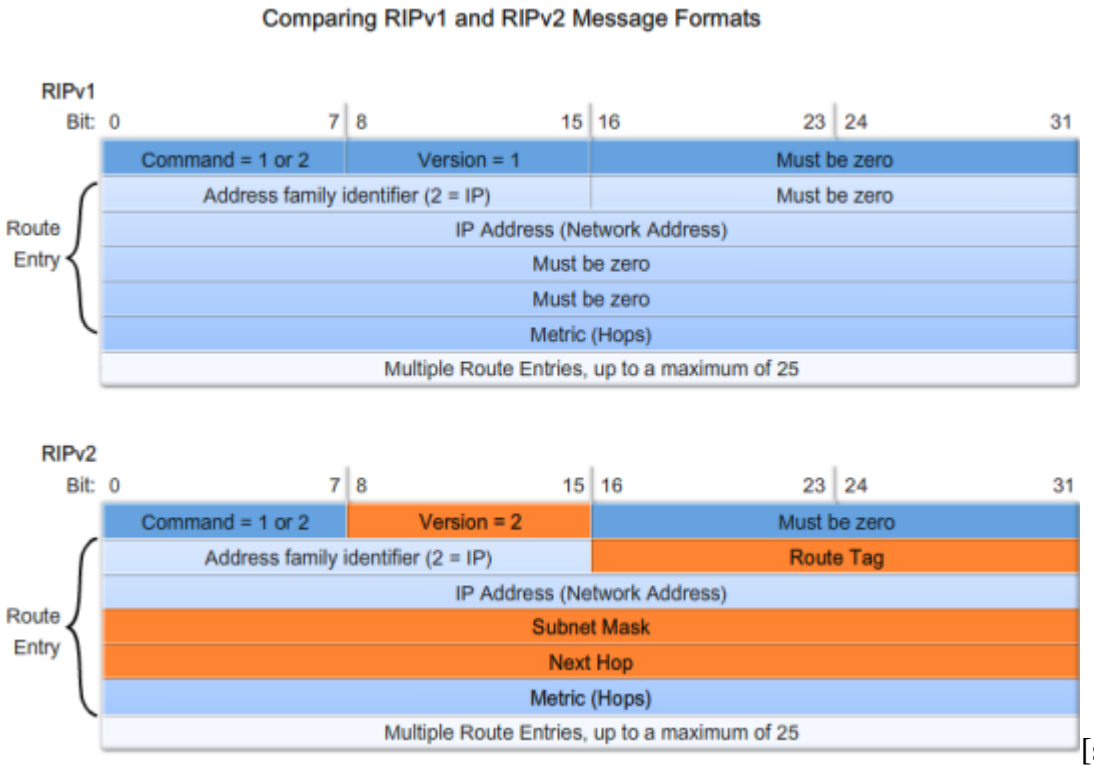
La fel ca RIPv1, RIPv2 este un protocol bazat pe algoritmul distance vector. "Atat RIPv1 cat si RIPv2 impartasesc urmatoarele caracteristici si limitari:

- Utilizarea lui "holddown" si a altor timere care ajuta pentru prevenirea buclelor de rutare.
- Folosesc "split horizon" sau "split horizon with poison reverse" tot pentru prevenirea buclelor de rutare.
- Folosesc "triggered updates" atunci cand apare o schimbare in topologie pentru convergenta mai rapida.
- Numarul maxim de hopuri este limitat la 15, iar numarul de hopuri egal cu 16 sa reprezinte o retea la care nu se poate ajunge.

1.6.1 Formatul mesajului la RIPv2

RIPv2 este definit in RFC 1723. La fel ca versiunea 1, RIPv2 este incapsulat intr-un segment UDP utilizand portul 520 si poate contine pana la 25 de rute. Desi RIPv2 are acelasi mesaj de baza ca si RIPv1, are adaugate inca doua extensii semnificante.

In figura de mai jos avem o comparatie intre formatul mesajului de la RIPv1 si RIPv2:



[sursa 7]

Probleme comune întâlnite la RIPv2

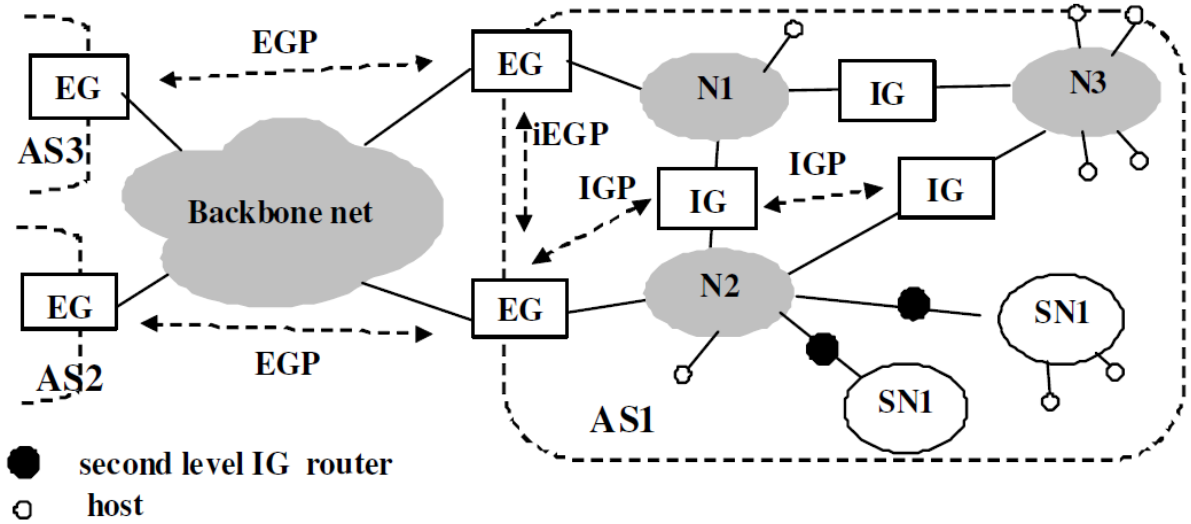
- Una din problemele care pot aparea este versiunea: nu a fost moficata din 1 sa fie 2. - O alta sursa de probleme pot fi datorate declararii gresite a retelei. Declarea retelei face doua lucruri: 1.permite protocolului de rutare sa trimita si sa primeasca update-uri pe orice interfata locala care apartine retelei 2.include retelele locale in update-urile de rutare din tabelele vecinilor - O alta sursa de probleme poate fi datorata auto-sumarizarii rutelor.

Autentificarea

"Majoritatea protocoalelor de rutare trimit update-uri despre rutare si alte informatii utilizand IP-ul. O problema de securitate a oricarui protocol de rutare o reprezinta posibilitatea de acceptare de update-uri invalide. Sursa acestor update-uri invalide poate fi un atacator care incearca sa distruga reteaua sau sa captureze pachete de date prin determinarea ruterului sa trimita update-uri catre destinatii gresite. O alta sursa pentru update-uri invalide poate fi reprezentata de un ruter configurat gresit."[1] "Indiferent de motiv, este bine sa se practice autentificarea informatiilor de rutare transmise intre rutere. Protocoalele RIPv2, EIGRP, OSPF, IS-IS si BGP pot fi configurate sa autentifici informatiile despre rutare. Aceasta practica asigura ruterele ca vor accepta informatii despre rutare numai de la alte rutere care au fost configurate cu aceasi parola sau informatii despre autentificare. Autentificarea nu encripteaza tabela de rutare."

1.3 Topologia generic ierarhică în Internet a protocolului BGP

Ierarhia este alcătuită generic din: rețeaua backbone, sisteme autonome, rețele, hosturi. Sistemele autonome reprezintă partea logică a rețelelor IP mai mari administrate de către o singură autoritate. Acestea ar cuprinde în mod normal în cadrul unei organizații internetwork, și ar fi desemnate ca atare pentru a urmări comunicarea peste rețelele IP publice cu AS-uri aparținând altor organizații. Este obigatoriu să se înregistreze o organizație de tip internetwork ca un AS în vederea utilizării acestor servicii publice de tip IP.
Adresele IP sunt niște adrese globale echivalente cu adresele NSAP, diferite de NPA (Network Point of Attachment) care depind de subrețea particular.



[sursa 2]

Figura 5 Arhitectura generică a protocolului BGP

Border Gateway Protocol (BGP) este un protocol de rutare inter-autonom. Un sistem autonom este o rețea sau un grup de rețele sub o administrare comună și cu politici comune de rutare. BGP este folosit pentru a face schimb de informații de rutare în Internet și este protocolul utilizat între furnizorii de servicii Internet (ISP). Rețelele de tip client, cum ar fi universitățile și corporații, de obicei folosesc un interior Gateway Protocol (IGP), cum ar fi RIP sau OSPF pentru schimbul de informații de rutare în cadrul rețelelor lor. Clienții care se conectează la ISPs și ISP folosesc protocolul BGP pentru a face schimb de rute clienți și ISP. . Atunci când se utilizează BGP între sisteme autonome (AS), protocolul

este numit BGP extern (EBGP). În cazul în care un furnizor de servicii utilizează BGP pentru a face schimb de rute în cadrul unui sistem autonom, atunci protocolul se referă la BGP interior (IBGP).

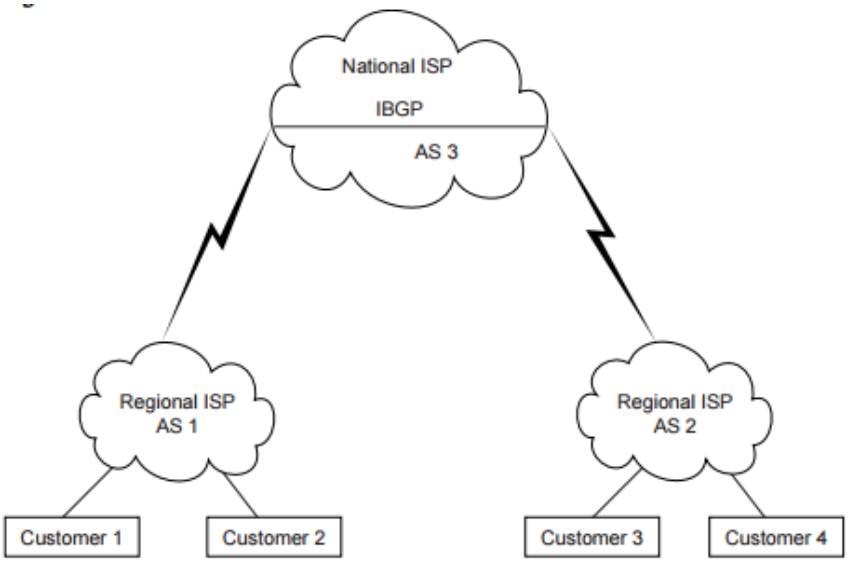


Figura 6

[sursa 3]

BGP este un protocol de rutare foarte robust și scalabil , după cum reiese din faptul că BGP este protocolul de rutare utilizat în Internet. Numărul tabelelei de rutare în Internet pentru BGP era mai mare de 90.000 de rute. Pentru a atinge scalabilitate la acest nivel, BGP utilizează mai mulți parametri de rutare, numite atribute, pentru a defini politicile de rutare și de a menține un mediu stabil de rutare. In plus față de atributele BGP, rutarae clasificată interdomeniu (CIDR) este folosită de BGP pentru a reduce dimensiunea tabelelor de rutare în Internet. De exemplu, să presupunem că un ISP deține adresa IP bloc 195.10.xx din spațiul de adrese tradiționale de clasa C. Acest bloc este format din 256 de clasă blocuri de adrese C, 195.10.0.x prin 195.10.255.x.

Să presupunem că ISP atribuie un bloc de clasa C la fiecare dintre clienții săi. Fără CIDR, ISP-ul ar face publice 256 blocuri de adrese din clasa C către vecinii săi BGP. Cu CIDR, BGP poate prin supernet al spațiul de adrese să facă public un singur bloc, 195.10.xx Acest bloc are aceeași mărime ca un bloc de adrese tradiționale de clasă B. Distincțiile de clasă devin folosite prin CIDR, permițând o reducere semnificativă în tabelele de rutare BGP. Vecinii BGP fac schimb de informații de rutare complete atunci când conexiunea TCP între vecini este mai întâi stabilită. Atunci când sunt detectate modificări în tabelul de rutare, routere BGP trimit la vecinii lor numai acele rute care s-au schimbat. Rutere BGP nu trimite actualizări periodice de rutare, precum și actualizările de rutare BGP fac publică numai calea optimă la o rețea de destinație.

1.4 Distance Vector Protocol

Protocolul Distance Vector

Asa cum implica si numele, rutele sunt considerate ca niste vectori cu lungime(distanta) si directie. Distanta este definite ca metrica (numarul de noduri de care trebuie sa teaca un pachet pentru a ajunge la destinatie) si directia este urmatorul nod sau interfata de iesire.
Protocolul de rutare Distance Vector foloseste algoritmul Bellman-Ford de rutare, numit dupa cel care l-a dezvoltat(Bellman, 1957; Ford si Fulkerson, 1962) pentru a determina cea mai buna ruta. A fost algoritmul original ARPANEET si a fost introdus in Internet sub numele de RIP. Desi acest algoritm acumuleaza suficiente date pentru a mentine convergenta unei retele, acesta nu permite routerelor sa stie topologia exacta a intra-retelei. Routerul primeste informatii despre rutare doar de la vecini.

Caracteristicile protocolului de rutare DV (distance vector):

- actualizări periodice
- actualizările de rutare RIP includ întregul tabel de rutare
- vecinii sunt definiți ca ruterele care distribuie o legătură și sunt configurați să utilizeze același protocol

• Procesul de descoperire de rețea pentru protocolul DV de rutare:

- rute direct conectate
- ruterele vor schimba între ele informații de rutare
- convergența este atinsă atunci când toate ruterele din rețea au aceeași informație de rețea.

Distance vector protocol menține tabela de rutare în timp ce RIP va trimite actualizări periodice. RIP utilizează patru timere diferite pentru a garanta că infoormațiile sunt exacte și convergente. EIGRP trimite trigger updates. DV poate să producă bucle în rețea, buclele de rutare sunt o condiție în care pachetele traversează în mod continuu rețeaua..

Mecanismul utilizat pentru a minimiza buclele de rutare include definirea pasului curent (hop count) maxim., divizarea orizontale (split horizon), actualizarile declanșate, hold down timers etc.
Condițiile care duc la buclele de rutare includ:

- rute statice incorect configurate
- convergența lentă

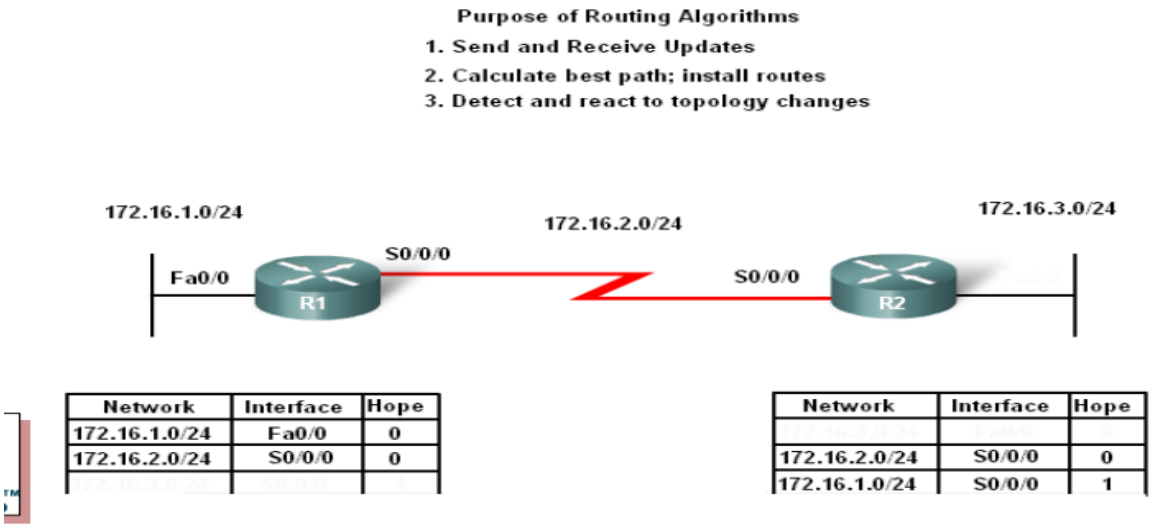
Toate acestea pot să infuenteze consumul excesiv de bandă, degradarea convergenței rețelei, actualizările de rutare pot fi pierdute sau nu pot fi procesate

Avantejele DVP:

- impementarea simplă și mentenanța
- cerințe de resurse scăzute

Dezavantaje DVP:

- convergența lentă
- scalabilitate limitată
- bucle de rutare



Distance vecor protocol , ce presupene acest algoritm [sursa 7]

Rutarea dinamică a fost utilizată în rețele încă din anii 1980. Cum rețelele au evoluat și au devenit din ce în ce mai complexe, au fost implementate mai multe protocoale de rutare.
Avantajele protocolului de rutare dinamica sunt:
• Administratorul nu necesita foarte multa implicare in configurarea retelei atunci cand se adauga sau se sterg conexiuni spre alte retele.
• Protocoalele reactioneaza automat atunci cand topologia retelei se schimba.
• Configuratia este mai putin inclinata spre erori.
• Aceste protocoale sunt mai scalabile; complexitatea unei retele nu reprezinta o problema.

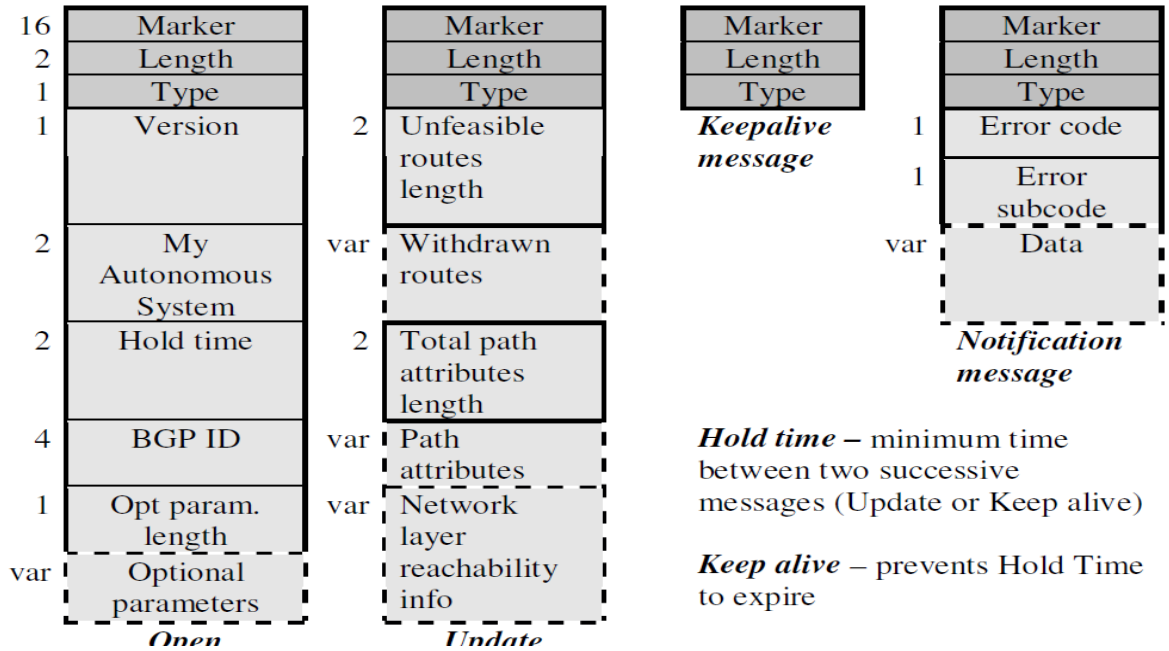
Dezavantajele protocolului de rutare dinamica:

• Resursele routerelor sunt mai intens folosite.

• Este necesara o cunoastere mai aprofundata in vederea verificarii si rezolvarii de probleme in vederea configurarii retelei..

1.5. Ierarhia și mesajele BGP, neighbor acquisition

Formatul mesajelor BGP



[sursa 2]

Neighbors acquisition

Conceptul de neighbors acquisition este atunci când două rutere vecine sunt de accord să facă schimb de informații de rutare în mod regulat, necesare deoarece un singur ruter nu ar putea să ia parte. Un ruter trimite cererea iar celălalt răspunde. Cu alte cuvinte se poate cunoaște existența mai multor rutere și totodată necesitatea de a face schimb de informații stabilite la momentul configurării sau prin intervenție activă.

Routerul deschide conexiunea TCP cu vecinul, va trimite un mesaj open, se va identifica AS-ul care a trimis mesajul și se va da adresa IP, se include hold time așa cum este stabilit de cel care trimite, destinatarul se va pregăti să deschidă comunicarea cu vecinii, se va calcula timpul de menținere, timpul maxim între mesajele keepalive, și se va genera raspunsul cu keepalive.

Detaliu keepalive apare numai la nivelul antetului, este sufficient pentru a preveni expirarea hold time(timpul de menținere al legăturii). Dacă timpul de așteptare expiră se va genera o schimbare topologică.

1.5.1 Evoluția protocolului BGP și metrica de rutare a algoritmului

Metrica algoritmului de rutare depinde de următorii parametrii

- Bandă
- Întarziere
- Cost
- Unitatea maximă de transmisie (MTU)
- Încărcare
- Fiabilitate

Rețelele din interiorul unui sistem autonomy (AS) împărtășesc în mod normal aceeași strategie a protocolului de rutare și este în mod tipic sub controlul aceleiași administarări de rețea.

Un protocol IGP (Interior Gateway Protocol), există ăn interiorul unui sistem autonomy (AS). Un IGP este responsabil pentru toate funcțiile de rutare și determinările rutelor din interiorul unui AS.

Câteva dintre cunoscutele protocoale IGP amintim:

- RIP
- IGRP
- EIGRP
- IS-IS
- OSPF

Un protocol de rutare exterior va efectua funcțiile între sistemele autonome. Un EGP este responsabil pentru toate funcțiile de rutare și determinarea rutelor între AS-uri. Câteva dintre binele cunoscute protocoale EGP amintim:

- IDRP
- EGP
- BGP

O funcție de rutare de bază este de a trimite un pachet de la o interfață de intrare către o interfață curentă de ieșire. Rutarea dinamică impică protocoalele de rutare care rulează în rețea și determină rutele de la o sursă către celelalte distinații. Un AS este o colecție de rețele sub administrarea unui singur domeniu.

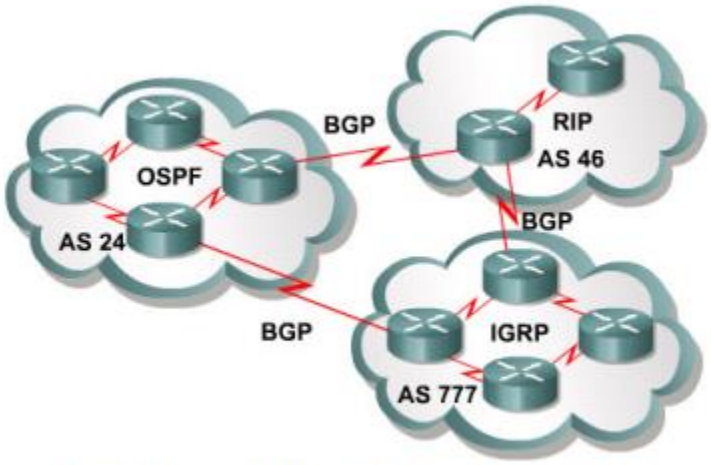


Fig. 2 Exemplu de rețea multi-AS (sursa: [1])

Evoluția protocolului BGP este urmatoarea:
EGP Version 1 (RFC-829) 1982
EGP Version 2 (RFC-904) 1984
BGP Version 1 (RFC-1105) 1989
BGP Version 2 (RFC-1163) 1990
BGP Version 3 (RFC-1267) 1991
BGP Version 4 (RFC-1771) 1995

Beneficiul major al BGP-4 a fost Acela că ajută la rezolvarea epuizării adreselor IP suportând CIDR (Classless Inter-Domain Routing).

Un alt impediment cu care se confruntă Internetul a fost creșterea exponențială a tabelelor de rutare, care în 1995 erau de 30000 de rute. Versiunea 4 de BGP suportă agregarea rutelor ceea ce a rezultat într-o reducere semnificativă a tabelei de rutare în ultimul deceniu. Mai mult decât atât tabela de rutare a fost dublată. Prin apariția prefixelor IP.

Avantaje și dezavantaje între BGP, OSPF, RIPv1 și RIPv2

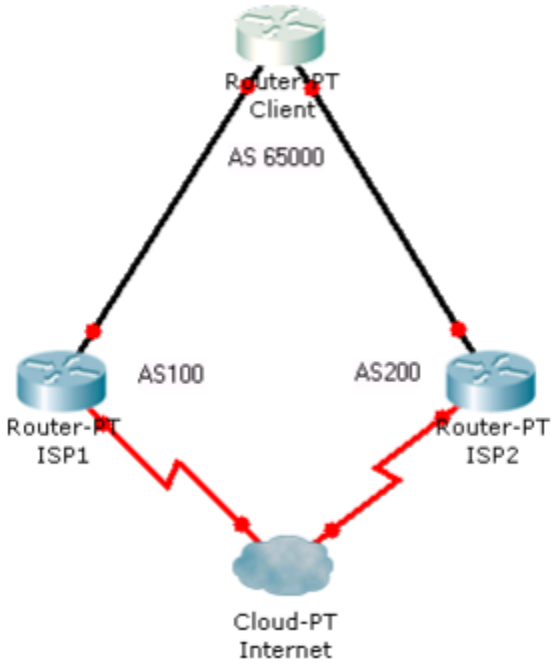
Protocolul OSPF (Open Shortest Path First) este un protocol de rutare dinamică de tip link-state, spre deosebire de RIP şi IGRP care erau de tip distance-vector. Aceasta înseamnă ca toate ruterele cunosc topologia completă a reţelei şi pot lua decizii de rutare fără riscul de a genera bucle în reţea. Metrica folosită se bazeaza pe un cost care este de obicei banda disponibilă pe fiecare legătura, în timp ce la RIP metrica era dată de numărul de hopuri (aceasta înseamnă ca, la RIP, este preferata o cale directă între 2 noduri pe o legătura seriala de viteza mică în locul unei cai indirecte, cu un nod intermediar, dar pe o legătura Gigabit Ethernet). Un alt avantaj al OSPF faţă de RIP este numărul mai mare de hopuri suportate (255 faţă de 15) ceea ce permite funcţionarea în reţele de dimensiuni medii/mari. Ca şi RIP, OSPF este un protocol bazat pe un standard (open) astfel încît este posibilă interoperarea între mai multi producători de rutere. Algoritmul pentru determinarea distanţei minime între oricare 2 noduri (shortest path) este algoritmul Dijkstra. RIP şi OSPF sînt protocoale de rutare interioare, adică sînt folosite în interiorul unui domeniu aflat sub o administraţie comună. Un astfel de domeniu se numeşte AS (Autonomous System). Rutele, protocoalele, politicile de rutare şi celelate aspecte tehnice şi administrative se definesc în mod unitar şi sînt sub controlul respectivului AS. Prin comparaţie, există şi protocoale de rutare exterioare (de exemplu BGP) care sînt folosite între AS diferite.

Din punct de vedere al funcţionării, OSPF este semnificativ mai complex decît RIP. Mesajele de rutare schimbate între rutere se numesc LSA (Link State Advertisements). Un ruter trimite în LSA informaţii precum starea legăturilor sale cu celelalte rutere conectate (link states). Pe baza acestora fiecare ruter din reţea crează o tabelă (link state database). Aceste tabele trebuie să fie identice pentru toate ruterele din reţea (sau, după cum se va vedea, măcar din cadrul unei arii). Aplicînd algoritmul SPF asupra acestei tabele, fiecare ruter obţine căile de cost minim către toate destinaţiile din tabelă, ceea ce formează tabela de rutare (routing table). Pentru a reduce numărul de LSA în reţele de mari dimensiuni se foloseşte conceptul de arie (Area) care reprezintă o anumită parte a reţelei (Cisco recomandă ca într-o arie să nu fie mai mult de 50 de rutere, dar se pot face şi arii mult mai mici – depinde de viteza legăturilor disponibile, precum şi de puterea de calcul a ruterelor). Împărţirea în arii se face astfel încît să existe o arie numită Backbone Area – aria la care se leagă toate celelalte arii, dacă există. Aceasta se numerotează obligatoriu Area 0, celalte arii putînd fi numerotate oricum. Informaţiile de link state sînt identice numai pentru ruterele din interiorul unei arii. Între două arii nu se transferă informaţii de link state referitoare la ruterele din interiorul fiecarei arii, ci doar la ruterele de margine (Area Border Routers), ceea ce face OSPF scalabil în AS-uri mari.

Capitolul 2- Traficul și problemele BGP, rezolvarea acestora

2.1 Schimbul de informații de rutare în cadrul protocolului BGP

BGP menţine o tabelă cu reţele IP (sau "prefixe") care arată calea folosită pentru a ajunge la reţeaua respectivă prin diferitele sisteme autonome (AS)
Se presupune existența a trei rutere R1, R2 și R3, unde R1 va constitui tabela de rutare pentru AS1 (sistemul autonom 1) utilizând OSPF, R1 va actualize și mesajele eronate către R3 din AS2. As-Path identifică AS1, iar next-hop va reprezenta adresa IP a R1, NLRI este lista tuturor subrețelelordin AS1. Cu alte cuvinte dacă R2 ar avea relații cu vecinul R3 in AS3 atunci R2 va transmite informații de la R1 către R3 în mesajele update. AS-Path este lista id-urilor {AS2,AS1. R3 decide dacă este ruta preferată și transmite către rutele vecine.



[sursa 2]

BGP nu foloseşte aceleaşi metrici ca protocoalele de rutare folosite în interiorul ASurilor, ci ia decizii bazîndu-se pe cale şi pe politicile de rutare ale sistemului autonom din care face parte.
Reţelele IP de mari dimensiuni folosesc BGP inclusiv în interiorul reţelei, de exemplu pentru a lega mai multe subreţele suficient de mari pentru ca protocolul de rutare OSPF să-şi atingă limitele. Alt caz de utilizare îl reprezintă conectarea mai multor puncte de prezenţă ale unui singur furnizor de acces Internet (acest caz este descris în RFC 1998).

Border Gateway Protocol este un protocol de rutare unic, deoarece, spre deosebire de celelalte protocoale de rutare, stabileşte şi menţine conexiuni între ruterele vecine folosind protocolul TCP. În cazul ruterelor aflate în ASuri diferite, o conexiune BGP poate fi stabilită doar dacă ruterele sunt direct conectate. Legătura se realizează pe portul TCP 179, fiind menţinută prin mesaje periodice de 19 octeţi (intervalul implicit este de 60 de secunde).

Vecinii eBGP trebuie să fie direct conectaţi pentru a fi realizată conexiunea BGP, dar există şi excepţii. De exemplu, implementările Cisco au opţiunea "multihop", care permite realizarea de conexiuni eBGP către rutere nelegate direct. Această limitare nu există pentru iBGP. Pentru a asigura rutarea între toate nodurile din reţea care rulează BGP poate fi folosit un protocol de rutare IGP (OSPF, RIP etc.).

În mod normal, un ruter iBGP menţine sesiuni cu toate celelalte rutere iBGP din AS, formând o topologie logică full-mesh (fiecare cu fiecare). Acest lucru este necesar deoarece, pentru a preveni formarea de cicluri de rutare, iBGP nu transmite rute învăţate prin iBGP altor vecini care rulează iBGP. Dacă se doreşte ca ruterele iBGP să schimbe rute BGP între ele, este necesară configurarea de reflectori de rute (route reflector) sau confederaţii.

Cînd un ruter află despre o ruta nouă prin protocolul eBGP, va seta adresa următorului hop la adresa ruterului vecin eBGP de la care a aflat ruta respectivă. Când se primesc rute din interiorul ASului, adresa următorului hop rămîne neschimbată.

Un factor care cauzează mărirea dimensiunii tabelei de rutare este echilibrarea încărcării pentru reţele cu mai multe legături externe. Dacă un ISP îşi publică reţeaua către toţi vecinii BGP, una sau mai multe dintre legături pot fi congestionate, pe când celelalte sunt subutilizate. Acest lucru se întâmplă în cazul în care toţi vecinii consideră acele legături ca optime. Ca şi celelalte protocoale de rutare, BGP nu detectează congestia.

Pentru a evita această problemă, administratorii reţelelor respective împart blocul de adrese pe care îl administrează în sub-blocuri şi publică pe fiecare legătură BGP un alt bloc de adrese. Acest lucru duce la mărirea numărului de intrări din tabelele BGP.

2.2 Configurarea politicilor BGP locale și sincronizarea.

Un router BGP într-un ISP (Internet Service Provider) poate avea mai multe rute alternative pentru a ajunge la o anumită destinație. În absența unei politici de rutare, router-ul ar alege ruta de lungime minimă (cost minim), într-un mod arbitrar de a elimina legăturile dintre rutele cu aceeași lungime. Cu toate acestea, în scopul de a oferi programatorilor un control mai mare asupra traseului de selecție, s-au adăugat mai multe atribute suplimentare, permițând ca un router să-și modifice deciziile sale bazate pe valorile acestor atribute. Rezultatul final este procesul de decizie BGP, constând dintr-o listă ordonată de atribute cu care rutele sunt comparate, așa cum se arată în tabelul 1.

Step	Attribute	Controlled by local or neighbor AS?
1.	Highest LocalPref	local
2.	Lowest AS path length	neighbor
3.	Lowest origin type	neither
4.	Lowest MED	neighbor
5.	eBGP-learned over iBGP-learned	neither
6.	Lowest IGP cost to border router	local
7.	Lowest router ID (to break ties)	neither

Tabela 1 Pașii în decizia protocolului BGP [sursa 7]

Router-ul parcurge lista, comparând fiecare atribut din această lista cu privire la cele două rute. În cazul în care rutele au valori diferite pentru atributul, router-ul îl alege pe cel care are atributul dorit, în caz contrar se trece la compararea cu atributul următor din listă. Traseul ales este utilizat de router pentru a transmite pachete.

Există diferite locații în cazul în care un atribut rută poate fi stabilit prin politici:

a. La nivel local, de exemplu, Local-Pref este o valoare întreagă stabilită și transmisă prin intermediul sistemului autonom local AS și filtrată înainte de a fi transmisă la ISP-uri vecine.

b. Vecinii, de exemplu, atributul MED este utilizat în mod obișnuit de două AS-uri conectate prin mai multe link-uri pentru a indica care link pereche ar trebui să fie utilizate pentru a ajunge la AS publicat de atributul MED, și nu este folosit pentru a compara rute prin doua next-hopuri AS diferite.

c. Neither (Nimic): de exemplu unele atribute în cazul în care traseul a fost învățat printr-un vecin extern BGP (eBGP) sau de la un router intern vorbind BGP (iBGP), sunt stabilite prin protocol și nu pot fi schimbate.

Există trei clase de care pot fi folosite pentru a controla importul și exportul politicilor:

1. Preferințele influențate de rutele BGP vor fi alese pentru fiecare prefix destinație

Schimbarea preferințelor se face prin adăugarea / ștergerea / modificarea atributelor rută în BGP. Tabelul 1 arată atributele care pot fi modificate în timpul importului pentru a controla preferințele la nivel local, și care poat fi modificată și în timpul exportului pentru a arăta cât de mult un vecin preferă o ruta.

2. Filtrarea elimina anumite rute luate în considerare și controlează de asemenea care vor fi exportate.

Filtrarea poate fi aplicată atât înainte (la intrare filtrării) cât și după preferință (la ieșire filtrării). Filtrarea se face prin instruirea routere pentru a ignora atributele potrivite cu anumite valori specificate sau intervalelor.

3. Etichetarea permite unui operator să asocieze o stare suplimentară cu o rută, care poate fi folosită pentru a coordona deciziile luate de către un grup de routere într-un sistem autonom, sau de a împărtăși contextul dincolo de limitele AS. Mecanismul cheie este atributul community, un șir de caractere de lungime variabilă este utilizat pentru a eticheta rutele. Acest atribut comunitate este un mecanism extrem

de expresiv, pentru a sprijini o gamă largă de politici complexe, care sunt dificil de exprimat prin alte mijloace. De exemplu, o valoare de community ar putea afecta modul în care setează router-ul atributul Loca-lPref, în timp ce altul ar putea determina ruta să fie aleasă de un alt router. Cu toate acestea, expresivitatea oferă potential pentru configurațiile greșite, aceasta este amplificată de faptul că utilizarea atributelor community nu este standardizată.

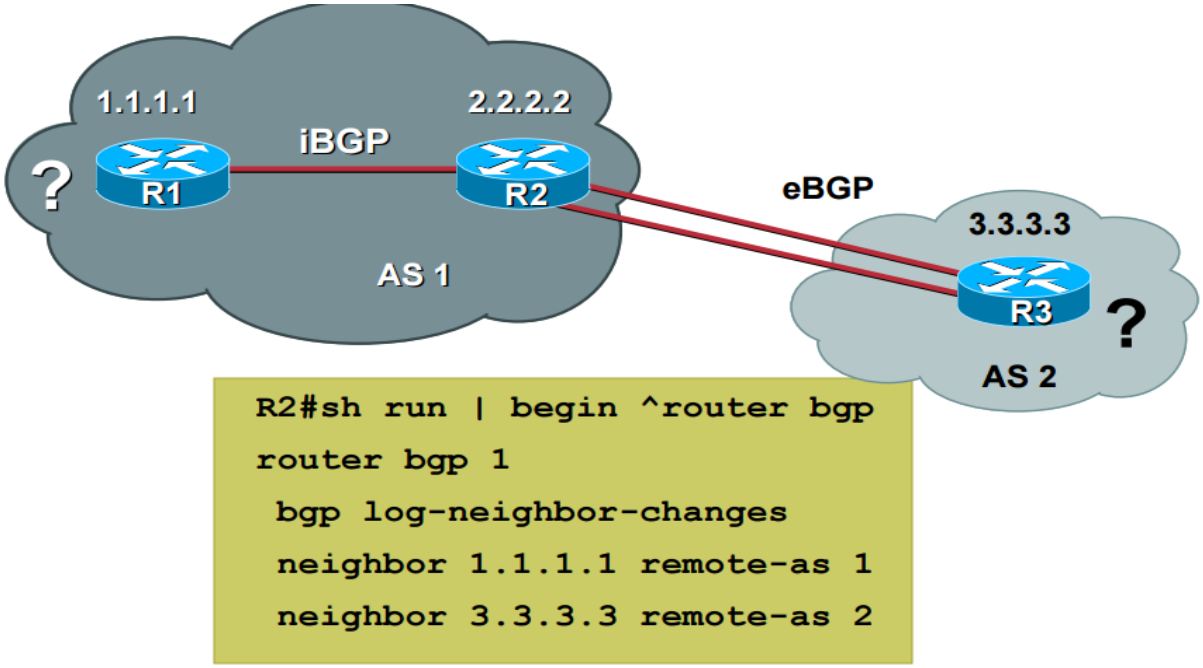
O implementare ISP pune în aplicare politicile sale prin aplicarea comenzilor de configurare la rutere. Aceste configurații constau în mod tipic dintr-un set de liste de preferință, filtrate și reguli de etichetare, o listă pentru fiecare sesiune de routere se face cu un router vecin prin BGP. Deși limba configurată diferită între furnizori, un cheie primitivă, care este adesea furnizată este un traseu hartă, o construcție de limbaj folosit pentru a modifica atributele de rută și definește condițiile care determina ce rute sunt exportate către vecini. Se compune din două părți: un set de condiții care indică atunci când harta este invocată (de exemplu, refixul este o valoare specificată, sau calea AS se potrivește cu o expresie specifică), precum și măsurile care trebuie luate în cazul în care publicarea se potrivește cu condițiile (de exemplu, modificarea unui atribut specificat, sau alegerea rutei).

2.3 Probleme BGP și rezolvarea acestora

2.3.1 Legătura de la egal la egal (Peer establishment)

Primul pas în a rezolva o problemă este aceea de a identifica acea cauză ce a dus la producerea ei. Problemele posibile sunt configurarea eronată și erorile umane.

În cele ce urmează este prezentată diagrama pentru legătura de la egal la egal (peer establishment):



[sursa 3]

În AS1 se folosește protocolul de rutare iBGP iar legătura între AS1 și AS2 se face prin intermediul protocolului eBGP. În cazul acestei legături de la egal la egal router-ul R2 este un intermediar în vederea schimbului de pachete de rutare între vecinii R1 (1.1.1.1) și R3 (3.3.3.3).

```
R2#show ip bgp summary
BGP router identifier 2.2.2.2, local AS number 1
BGP table version is 1, main routing table version 1
Neighbor   V   AS MsgRcvd MsgSent   TblVer   InQ OutQ Up/Down State
1.1.1.1    4    1       0       0        0     0    0  never  Active
3.3.3.3    4    2       0       0        0     0    0  never  Idle
```

[sursa 3]

Starea routerelor vecine poate fi schimbată între Idle, Active and Connect de către programator. Următorul pas este de a se vedea dacă conectivitatea IP a fost stabilită. Se va verifica TCP pentru a stabili care conexiune a fost acceptată:

```
R2#show tcp brief all
TCB      Local Address   Foreign Address    (state)
005F2934 *.179           3.3.3.3.*           LISTEN
0063F3D4 *.179           1.1.1.1.*           LISTEN
```

[sursa 3]

Se va aștepta realizarea conexiunii pe portul TCP 179 pentru a configura adresa de interconectare (peering address)- care permite schimbul de trafic între cele două routere (R1 și R2). În acest caz adreasa de interconectare va fi: 4.4.4.4:26385.

```
R2#debug ip tcp transactions
TCP special event debugging is on
R2#
TCP: sending RST, seq 0, ack 2500483296
TCP: sent RST to 4.4.4.4:26385 from 2.2.2.2:179
```

[sursa 3]

Se încearcă remote de către router-ul R1 deschiderea sesiunii de la adresa 4.4.4.4

```
R2#debug ip bgp
BGP debugging is on
R2#
BGP: 1.1.1.1 open active, local address 4.4.4.5
BGP: 1.1.1.1 open failed: Connection refused by remote host
```

[sursa 3]

Se încearcă remote de către router-ul R1 deschiderea sesiunii de la adresa 4.4.4.5

```
R2#sh ip route 1.1.1.1
Routing entry for 1.1.1.1/32
  Known via "static", distance 1, metric 0 (connected)
  * directly connected, via Serial1
      Route metric is 0, traffic share count is 1

R2#show ip interface brief | include Serial1
Serial1      4.4.4.5      YES manual   up      up
```

[sursa 3]

Adresa sursei este interfața de ieșire spre destinație, dar adresa de interconectare (peering) în acest caz, utilizează interfețe loopback. Forțează ambele rutere către adresa sursă prin interfețele corecte.

```
R2#
router bgp 1
 neighbor 1.1.1.1 remote-as 1
 neighbor 1.1.1.1 update-source Loopback0
 neighbor 3.3.3.3 remote-as 2
 neighbor 3.3.3.3 update-source Loopback0
```

Problema care apare este că iBGP rulează între interfețele de loopback ale ruterului pentru stabilitate, dar această configurașie de loopback lipsește de pe ruter, iBGP va eșua în ceea ce privește

stabilirea conexiunii. Pentru rezolvarea acestei probleme se vor folosi alte protocoale de rutare interne și nu iBGP.

2.3.2 Rute lipsă

Problema comună care apare este atunci când sunt puse prefixe în tabele BGP:
- tabela BGP nu este tabela de rutare (RIB- Routing Information Base) precum este tabela OSPF, rute statice tabela IS-IS etc este doar utilizată în vederea organizării tabelei RIB. Fiecare protocol de rutare are o prioritatate sau o distanță diferită . Pentru a obține un prefix în BGP va trebui să ruleze într-un alt proces de rutare.

Declararea rețelei:

```
R1# show run | include 200.200.0.0
 network 200.200.0.0 mask 255.255.252.0
```

[sursa 3]

Rețeaua nu se află în tabelă:

```
R1# show ip route 200.200.0.0 255.255.252.0
% Network not in table
```

[sursa 3]

BGP are mereu nevoie de o rută incompletă RI, se va agrega o adresă în tabele BGP nu in tabela de rutare RIB

“show ip route x.x.x.x y.y.y.y longer” // se vor găsi component rutelor din RIB
“show ip bgp x.x.x.x y.y.y.y longer” // se vor găsi rutele componente din BGP

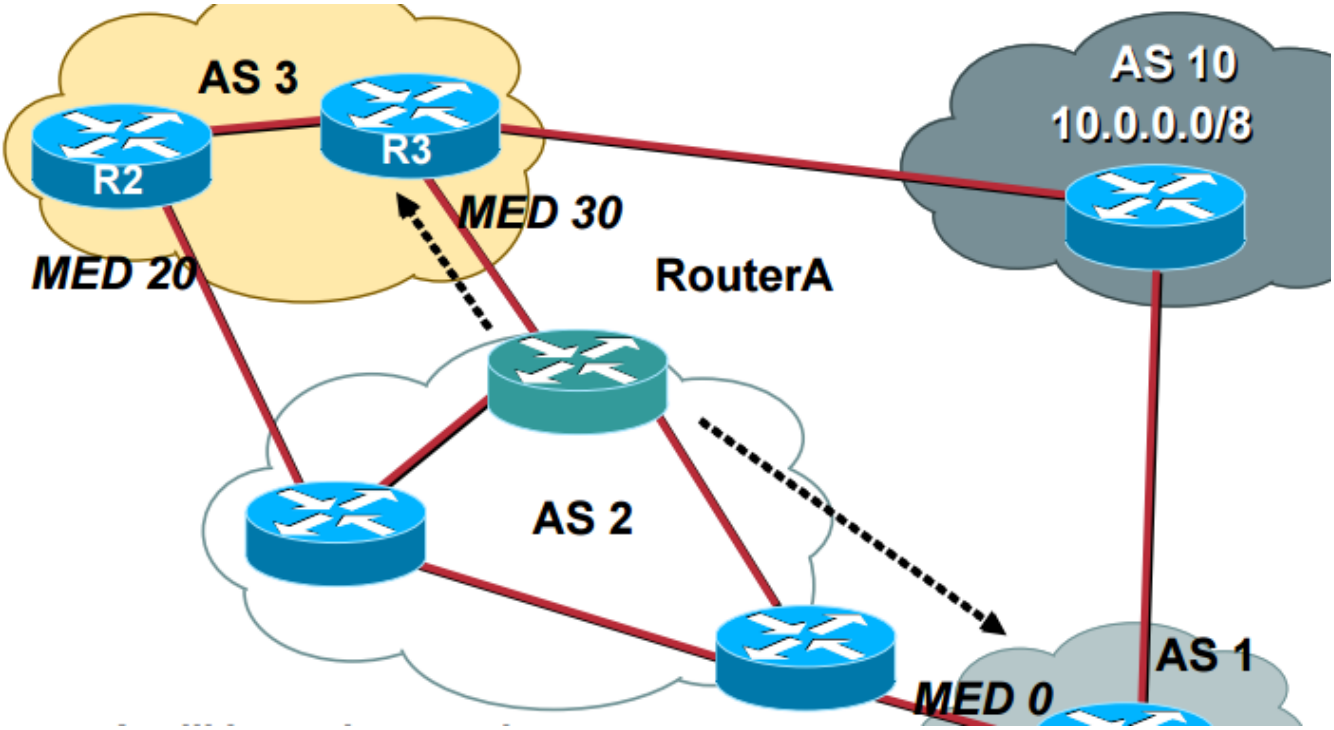
2.3.3 Selecția traseului incompletă

De exemplu, default-ul în Cisco IOS este de a compara prefixurile în ordinea în care sosesc (de la cele mai recente către cele mai vechi). Acest lucru poate avea ca rezultat selecția incompletă a traseului. Ca și rezultat este acela că cea mai bună cale aleasă va fi diferită la fiecare resetare BGP.

Selecția incomplete a traseului poate cauza probleme cu privire la buclele de rutare și anume protocolul trimite în mod continuu actualizări într-o încercare de a aduce modificări la nivelul traficului.

Dificil de prins și de rezolvat în CISCO IOS, comanda de configurarea a determinării erorilor este utilizată pentru a activa în mod constant toate ruterele din AS-ul respectiv. Calea cea mai bună este recalculate odată ce comanda este reintrodusă.

Fie următoarea configurație:



[sursa 3]

Router-ul A va avea trei părți:

MEDs de la AS3, nu va fi comparat cu MEDs de la AS1
RouterA va selecta, uneori, calea de la R1 cel mai bine și poate selecta, de asemenea, calea de la R3

Exemplu 1

```
RouterA#sh ip bgp 10.0.0.0
BGP routing table entry for 10.0.0.0/8, version 40
Paths: (3 available, best #3, advertised over iBGP, eBGP)
  3 10
    2.2.2.2 from 2.2.2.2
      Origin IGP, metric 20, localpref 100, valid, internal
  3 10
    3.3.3.3 from 3.3.3.3
      Origin IGP, metric 30, valid, external
  1 10
    1.1.1.1 from 1.1.1.1
      Origin IGP, metric 0, localpref 100, valid, internal, best
```

[sursa 3]

Initial State Path 1 beats Path 2 – Lower MED
Path 3 beats Path 1 – Lower Router-ID

```
RouterA#sh ip bgp 10.0.0.0
BGP routing table entry for 10.0.0.0/8, version 40
Paths: (3 available, best #3, advertised over iBGP, eBGP)
  1 10
    1.1.1.1 from 1.1.1.1
      Origin IGP, metric 0, localpref 100, valid, internal
  3 10
    2.2.2.2 from 2.2.2.2
      Origin IGP, metric 20, localpref 100, valid, internal
  3 10
    3.3.3.3 from 3.3.3.3
      Origin IGP, metric 30, valid, external, best
```

[sursa 3]

Se returnează 1.1.1.1 și căile sunt reordonate
Path 1 beats Path 2 – Lower Router-ID și Path 3 beats Path 1 – External vs Internal
Căile sunt ordonate de vecinul din AS, cea mai bună cale pentru fiecare vecin din AS este selectată. Cea mai bună cale va fi consistentă, deoarece căile vor fi plasate într-o ordine deterministă

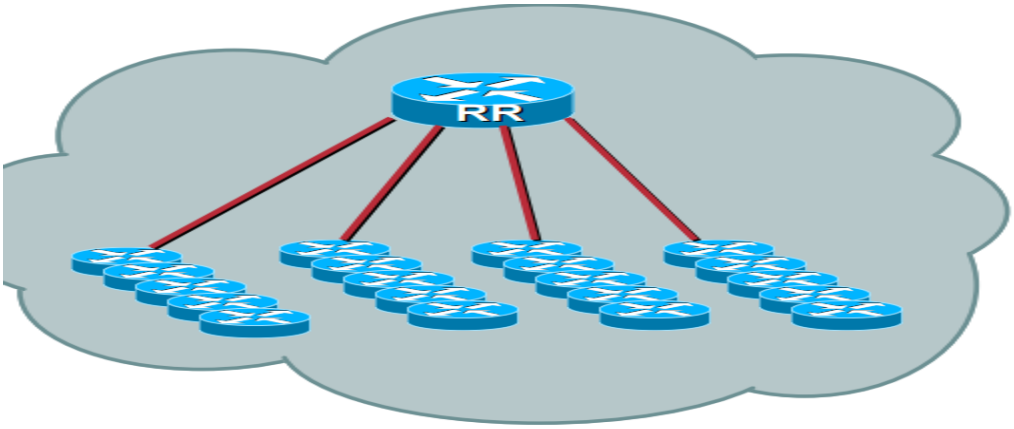
Se va obține rezultatul:

```
RouterA#sh ip bgp 10.0.0.0
BGP routing table entry for 10.0.0.0/8, version 40
Paths: (3 available, best #1, advertised over iBGP, eBGP)
  1 10
    1.1.1.1 from 1.1.1.1
      Origin IGP, metric 0, localpref 100, valid, internal, best
  3 10
    2.2.2.2 from 2.2.2.2
      Origin IGP, metric 20, localpref 100, valid, internal
  3 10
    3.3.3.3 from 3.3.3.3
      Origin IGP, metric 30, valid, external
```

[sursa 3]

Calea 1 este cea mai bună pentru AS1, Calea 2 va fi mai bună decât Calea3 pentru AS3-Lower MED
Calea 1 va fi mai bună decât Calea2- Lower Router-ID
Se va utiliza mereu "bgp deterministic-med". Această comandă va trebui să fie activă pe parcursul întregului interval de timp al precesului desfăsurat în cadrul rețelei.
Dacă este activat numai pe o porțiune din buclele de rutare de rețea, problemele de convergență pot deveni mai severe. Ca rezultat, comportamentul implicit nu poate fi modificată, astfel încât butonul trebuie să fie configurat de către utilizator.

2.3.4 Probleme legate de bucle și convergență



[sursa 3]

Se poate remarca faptul că din cauza compexității configurației, BGP nu este convergent în această situație.

Se vor utiliza ACLs cand se activează comanda de debug. Se asigură faptul că logarea BGP este activată și totodată MEDs-rile deterministe sunt active. Dacă întrega tabelă are probleme se alege un singur prefix și se va încerca rezolvarea lui.

Un ACL permite stabilirea de reguli prin care să fie permis sau interzis accesul pachetelor cu anumite adrese IP, fie integral, fie numai către anumite destinaţii, porturi sau protocoale. Practic, folosind ACL se poate crea un firewall folosind un ruter Cisco.

Exemple de reguli: anumite hosturile nu vor avea acces la siteurile de web dintr-o listă

• calculatoarele pe care lucrează studenţii vor avea acces numai la anumite site-uri web

• nici un host din exteriorul domeniului nu va avea acces Telnet la hosturile din interiorul domeniului, dar reciproca va fi permisă Specificarea hosturilor afectate de o regulă se face cu perechea adresa_IP, wildcard_mask.

Un wildcard mask prezintă similitudini cu un netmask, adică permite specificarea căror biţi din adresa IP li se aplică o regulă şi cărora nu li se aplică. Mai precis, se selectează/verifică acei biţi din adresa IP care au "0" în mască şi ignoră pe cei care au "1" în mască, oarecum invers faţă de netmask. Dar, wildcard masks sînt mai generale şi nu sînt neapărat egale cu inversul netmask.

2.3.5 Bouncing problem

Starea inițială este reprezentată de faptul că toate cele trei noduri sunt conectate direct la ruta R. Conținutul tabelei de rutare arată care căi care sunt legate de ruta R pe fiecare AS în fiecare etapă.

Se arată faptul că, chiar și într-o astfel de topologie simplă, creșterea căilor AS și a numărului mare de actualizări sunt generate din cauza problemei convergenței. În acest caz, mecanismul AS path nu împiedică un AS de la a învăța o cale nouă, sau invalidă de la vecinii săi.

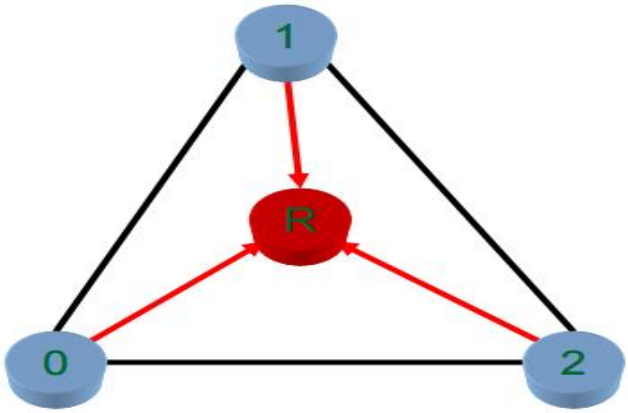
Impresia este că adoptarea vectorului cale în BGP poate agrava în mod exponențial problema robustă (bouncing problem). Raționamentul este că ruta explorată în BGP este monoton în creștere.

Se presupune că în cel mai rău caz, pot exista întârzieri suficient de lungi din cauza link-ului / sau procesării cozii. Toate acestea pot avea ca rezultat o ordonare a mesajelor astfel încât BGP va explora toate căile posibile de toate lungimile posibile. Limita superioară de convergență BGP ar fi O ((n-1)!)

Să presupunem că mesajul de întârziere este delimitat de timerul MinRouteAdver pe fiecare AS. În conformitate cu cea mai bună situație, după ce fiecare timer MinRouteAdver expiră informațiile de rutare vor fi agregate și numai căi de nivel superior vor fi analizate ulterior. Astfel AS path va crește strict până când toate căile au fost explorate.

Chiar dacă acest model este oarecum supra-simplificat, am putea avea o anumită intuiție care să aibă un anumit sens al rezultatelor. Acest lucru ne ajută să înțelegem și convergența lentă a BGP.

Se va identifica ruta preferată de fiecare nod, R are o problemă astfel va trimie mesaje de retragere la fiecare dintre vecinii săi.



[sursa 4]

Nodurile vor alege două căi, se vor detecta buclele după care se vor alege din nou două căi. Procesul va dura până se vor genera toate posibilitățiile.

Rutarea ierarhică este o componentă de bază în cadrul rețelelor, interconectarea în internet este un process complex. Abilitatea rețelei de a extinde gama de servicii este legată de modurile de peering, rutarea internet devine mai puțin ierarhică și condusă mai puțin de politici decât de performanță.

2.4 Securitatea

Un sistem autonom AS este extrem de vulnerabil la informații false în actualizările BGP. Prin trimiterea de informații false, un furnizor ISP poate subverta obiectivele de rutare ale vecinilor, cauzând routerelor supraîncărcarea și eșuarea conexiunii, sau să fie degradată calitatea serviciilor.

Informațile false pot avea o influență semnificativă în procesul de rutare într-un sistem autonom, chiar dacă sursa informației este de mai multe hopuri AS. Astfel de informații sunt uneori generate de bug-uri ale routerelor sau configurații eronate. Ar putea fi, de asemenea, generate cu rea intenție de către un ISP vecin, care concurează pentru clienți și, prin urmare, are un interes legitim în a face clienții ISP nemulțumiși de serviciul oferit. De aceea un ISP ar putea dori să își exercite programare defensivă pentru a se proteja împotriva atacurilor.

Eliminând rutele invalide (prin filtrare): furnizorii ISP-ar putea dori să protejeze clienții lor de învățare rutelor invalide prin efectuarea controalelor pentru a se asigura că acele conținuturi de actualizare sunt valide înainte de a le propaga pe plan intern.

Protejarea integrității politicilor de rutare (prin atribute rescriere): Un ISP poate dori să împiedice un vecin AS de a avea o influență nejustificată asupra deciziilor sale de rutare, încălcând acordul lor peering. În caz contrar, ISP-ul ar putea fi pacalit in transmiterea traficului pe o distanță mare peste rețeaua backbone în numele vecinilor.

Asigurarea infrastructurii de rețea (prin exportul filtrării): Un ISP poate dori să împiedice entități externe să acceseze anumite resurse interne prin configurarea politicilor sale de export, care filtrează BGP pentru destinații care nu ar trebui să fie accesibile în exterior.

Blocarea atacurilor de tip denial-of-service (prin filtrare și amortizare): Atacurilor de tip denial-of-service pot degrada calitatea serviciilor prin suprasolicitarea ruterele cu mesaje suplimentare de actualizare BGP sau consumă cantități excesive al link-ului de lățime de bandă.

Concluzii

BGP este folosit, în principal la scară largă, din cauza complexității de implementare. Acesta este utilizat, deci, între Provideri de Internet, în domeniul de Internet public datorită securității crescute a protocolului. Are ca scop, la fel ca celelalte protocoale de rutare, să găsească acea cale optimă Sursă-Destinație din punct de vedere al traficului.

Protocolul BGP (Boarder Gateway Protocol) este "motorul" Internetului curent, cu alte cuvinte conceptul de Internet este privit ca o suma de Sisteme Autonome (Autonomous System-AS) care schimbă rute prin intermediul protocolului BGP.

Prin specificarea AS-number BGP cunoaşte AS-ul propriu; în continuare, în funcţie de AS-number specificat pentru fiecare neighbor, BGP determină singur dacă respectivul este în acelaşi AS (şi deci protocolul va rula ca IBGP) sau în alt AS (EBGP).

BGP suportă opţiuni de configurare foarte complexe, în funcţie de politicile pe care administratorii doresc să le implementeze (în general prin manipularea atributelor rutelor).

Dacă comparăm RIP-ul cu alte protocoale de rutare în ceea ce privește distanța administrativă, acesta este cel mai puțin preferat. IS-IS, OSPF, IGRP și EIGRP au distanțe administrative default mai mici.

BGP utilizează mai mulți parametri de rutare, numite atribute, pentru a defini politicile de rutare și de a menține un mediu stabil de rutare. În plus față de atributele BGP, rutarea clasificată interdomeniu (CIDR) este folosită de BGP pentru a reduce dimensiunea tabelelor de rutare în Internet.

Perechile de rutere BGP comunică între ele stabilind conexiuni TCP. Operarea în acest mod oferă comunicaţie sigură şi ascunde toate detaliile reţelelor traversate.

BGP este la bază un protocol bazat pe vectori distanţă, dar destul de diferit de majoritatea celorlalte protocoale. În loc să menţină doar costul până la fiecare destinaţie, fiecare ruter BGP memorează calea exactă folosită. Similar, în loc să trimită periodic fiecărui vecin costul său estimat către fiecare destinaţie posibilă, fiecare ruter BGP comunică vecinilor calea exactă pe care o foloseşte.

BGP este un protocol robust și scalabil și moştenește abilitatea sistemelor autonome de a putea alege rutele şi de a-şi implementa regulile de rutare fără a trebui să depindă de o autoritate centrală.

BGP are şi câteva neajunsuri. În primul rând necesită configuraţie manuală excesivă.

BGP 4 are suport numai pentru Ipv4 , însă o versiune "multiprotocol" este în dezvoltare. Fiind necesară o politică de rutare se implementează soluţii ca: BGP tunnelling, Source Demand Routing, IDPR şi MPLS.

Din punctul meu de vedere BGP a adus ca îmbunătățire fundamental în cadrul explorării unei rețele distribuite (la nivelul cărora se aferă servere intermediare), utilizarea de trasee mult mai rapide și mult mai puțin aglomerate.

Consider că în cadrul acestui protocol există o mai mare fiabilitate de comunicații prin găsirea de rute alternative atunci când rutele prestabilite (default) suferă întreruperi.

În viitorul apropiat se urmărește dezvoltarea tot mai mare a protocolului MPLS, care la nivel Layer 3 utilizează BGP. Cu alte cuvinte se utilizează BGP în cadrul MPLS pentru ca operatorii de rețea să ofere o gamă largă de servicii.

Extensiile protocolului BGP pot fi folosite pentru a gestiona o cale MPLS.
BGP parcurge rute împreună cu etichetele MPLS care sunt mapate pe acea rută. Mesajul de actualizare BGP conține informațiile etichetelor de mapare MPLS despre ruta respectivă. Ruterele se asigură că pot trimite rute cu etichete MPLS. Verificarea între routere se face atunci când etichetele MPLS sunt adăugate la actualizările mesajelor de ieșire ale BGP.

Bibliografie

[1]- Andrew Tanenbaum. Computer Networks. 4th ed. Prentice Hall, 2003, (Andrew Tanenbaum. Retele de calculatoare. Ed. a 4-a. Editura BYBLOS, 2003).

[2]-Network and Services- E. Borcoci, UPB 2014-2015

[3]- https://www.nanog.org/meetings/nanog33/presentations/troubleshooting.pd

[4]-http://disi.unitn.it/locigno/ didattica/AdNet/10-11/02- Routing_S.pdf

[5]-http://www.cs.cornell.edu/ courses/cs519/2003sp/slides/ 10_routingprotocols.pdf

[6]-http://robotics.eecs.berkeley. edu/~wlr/228a02/Lecture% 20Slides/routing3.pdf

[7]- https://www.cs.princeton.edu/~jrex/papers/policies.pdf

[8]- A. FARREL "The Internet and Its Protocols A Comparative Approach" Ed, Morgan Kaufmann, 2004

[9]- Keith Knightson, Naotaka Morita, "NGN Architecture: Generic Principles, Functional Architecture, and Implementation", IEEE Communications Magazine , October 2005 [

[10]- Martin W. Murhammer, et.al., TCP/IP Tutorial and Technical Overview, 200