

LUCRAREA NR. 5

RUTAREA DINAMICĂ

1. Obiective

Obiectivele acestui laborator sînt: familiarizarea cu rutarea dinamică și cu conceptul de metrică, înțelegerea diferenței între protocoale rutate și protocoale de rutare, între protocoale starea-legăturii și protocoale vector-distanță, familiarizarea cu sistemele autonome, studiul protocolului RIP, configurarea protocolului RIP.

2. Considerații teoretice

2.1 Utilizarea rutării dinamice

Un element important în stabilirea strategiei de rutare îl constituie alegerea modului de rutare în rețea: rutare statică sau dinamică. În cazul rutării statice, administratorul de rețea crează și modifică manual tabelele de rutare înainte de începerea rutării. Rutarea dinamică are la bază protocoale de rutare specializate care asigură comunicarea între rutere și distribuirea informației de rutare între rutere. Pentru ca un ruter să transmită traficul pe care-l primește trebuie să aibă informații stocate sub forma unor tabele de rutare, informații pe care le obține de la celelalte rutere conectate în rețea.

În rutarea dinamică, administratorul de rețea nu mai este obligat să populeze tabelele de rutare. Deasemenea, dacă se modifică topologia rețelei, protocoalele transmit informații despre această modificare. Timpul de convergență a rețelei depinde de timpul de convergență al protocoalelor de rutare utilizate.

2.2 Metrici de rutare

Una dintre cele mai importante funcții ale protocoalelor de rutare este evaluarea eficienței rutelor către o destinație. Într-o rețea cu rute redundante există mai multe căi pe care un pachet le poate parcurge de la sursă la destinație. În acest caz, tabela ruterului conține mai multe intrări pentru aceeași destinație și ruterul trebuie să decidă care este cea mai eficientă. De

aceea, în tabela de rutare s-a definit o coloana numită metrica consultată pentru evaluarea unei rute. Cu cât valoarea din coloana metrică este mai mică cu atât ruta este mai eficientă.

Metricile definite pentru protocoale sînt: lățimea de bandă, numărul de hopuri, întârzierea, încărcarea, gradul de încredere.

Numărul de hopuri măsoară distanța în numărul de rețele pe care datagrama le traversează. De fiecare dată cînd un ruter transmite un pachet într-un segment de rețea se număra un hop. Calea cea mai bună conține cel mai mic număr de hopuri.

Protocoalele care iau în considerare capacitatea unui link au ca metrică lățimea de bandă (măsurată în bps); în calcul sînt luate în considerare capacitățile tuturor link-urilor din calea de la sursă la destinație, fiind aleasă calea cu cea mai mare capacitate cumulativă.

Întârzierea reprezintă timpul necesar unui ruter pentru procesarea, stocarea în buffer și transmiterea unei datagrame. Protocoalele care au ca metrică întârzierile calculează întârzierile tuturor link-urilor din calea de la sursă la destinație, fiind aleasă calea cu cea mai mică întârziere cumulativă.

Deși gradul de încredere al unui link poate fi alocat de administrator ca o valoare fixă, acesta e măsurat într-un anumit interval de timp. Ruterele analizează link-urile, problemele raportate – link-uri căzute, erori de interfață, datagrame pierdute. Link-urile cu multe probleme au un grad de încredere scăzut; scara utilizată e de la 255 la 1, cu 255 cel mai mare grad de încredere.

Încărcarea unui link este o valoare variabilă, măsurată prin gradul de utilizare a capacității unui link într-un interval de timp. Valoarea 255 indică o ocupare de 100% a capacității. Valorile apropiate de 255 semnaleză congestia în rețea, valorile mici indică un trafic moderat.

2.3 Protocoale rutate și protocoale de rutare

Prin conceptul de protocol rutat se înțelege orice protocol de nivel rețea care furnizează suficiente informații în pachetul de la nivel rețea astfel încît pachetul poate fi transmis de la un host la altul pe baza schemei de adresare. Protocoalele rutate definesc formatul și cîmpurile pachetelor. Protocolul IP este un exemplu de protocol rutat.

Prin conceptul de protocol de rutare se înțelege un protocol care oferă suport unui protocol rutat, mecanisme de distribuire a informației de rutare. Protocoalele de rutare facilitează comunicarea între rutere pentru actualizarea și menținerea tabelor de rutare. Exemple de protocoale de rutare: RIP - Routing Information Protocol, IGRP - Interior Gateway Routing Protocol, EIGRP - Enhanced Interior Gateway Routing Protocol, OSPF - Open shortest Path First.

2.4 Protocoale vector-distanță și protocoale starea-legăturii

2.4.1 Protocoale vector-distanță

Denumirea vector-distanță se referă la faptul că rutele sînt transmise sub forma unor vectori (distanță, direcție), în care distanța este metrica iar direcția este hopul (ruterul) următor. Un ruter învață de la ruterele vecine și la rîndul lui transmite rute pe care le cunoaște; rutarea de acest gen se numește și “rutare prin rumoare”.

În clasa protocoalelor vector-distanță intră: RIP - Routing Information Protocol în rețelele IP, XNS RIP - Xerox, IPX RIP în rețele Novell, IGRP - Internet Gateway Routing Protocol protocol Cisco, DNA Phase IV protocol DEC, RTMP - Routing Table Maintenance Protocol protocol AppleTalk. .

Ruterele cu protocoale vector-distanță transmit rutelor vecine toată informația din tabela de rutare la intervale periodice de timp. Aceste intervale variază de la 10 secunde la 90 de secunde.

Ruter vecin desemnează un ruter cu care există o legătură directă. Protocolul vector-distanță transmite informația rutelor vecine și apoi depinde de acestea să transmită informația la rîndul lor rutelor vecine. Acesta este motivul pentru care se spune că rutarea vector-distanță utilizează actualizări hop-cu-hop.

Un ruter își anunță prezența în rețea prin transmiterea unui mesaj de broadcast. Ruterele vecine care utilizează același protocol de rutare vor intercepta mesajul.

2.4.2 Protocoale starea-legăturii

Protocoalele de tip link-state construiesc o bază de date cu întreaga topologie a rețelei și calculează drumul cel mai scurt pe baza unui algoritm

de tip Dijkstra (SPF - shortest path first). Pentru actualizarea tabelelor de rutare se trimite într-o primă etapă întreaga tabelă de rutare către toate ruterele ce rulează același protocol de rutare, aceasta realizându-se prin folosirea în câmpul destinație a unei adrese logice de multicast specifice fiecărui protocol în parte. După această etapă de trimitere a tuturor informațiilor, numită și flooding, actualizările se vor efectua doar la apariția unei schimbări în topologie, iar pachetele de actualizare vor conține doar informații despre rutele modificate, această metodă de actualizare numindu-se actualizare incrementală.

Principala problemă a acestor protocoale este că fiecare dintre rutere va trebui să construiască arborele topologic, și apoi să extragă rutele ca drumuri optime în acest arbore, iar acest proces necesită resurse de memorie și procesor adesea semnificative. În plus, efortul configurării unui protocol bazat pe starea conexiunii este adesea mult mai mare decât cel necesar pentru a configura un protocol bazat pe vectori de distanță. Cu toate acestea, datorită inițierii procesului de actualizare odată cu apariția modificărilor în topologie, precum și datorită folosirii adresării multicast, și deci a propagării informațiilor de actualizare în întreaga rețea, timpul de convergență pentru protocoalele link-state este semnificativ mai redus decât pentru cele distace-vector.

În clasa protocoalelor starea-legăturii intră: OSPF - Open Shortest Path First pentru rețelele IP, IS-IS - Intermediate System to Intermediate System pentru CLNS și IP, DNA Phase V protocol DEC, NLSP - NetWare Link Services Protocol protocol Novell.

2.4.3 Compararea tipurilor de rutare

Tabelul 5.1 *Vector-distanță versus starea-legăturii*

Rutarea vector-distanță	Rutarea starea-legăturii
Ruterele transmit informația de rutare doar vecinilor direcți	Ruterele transmit informația de rutare tuturor ruterele din topologie
Informația conține o estimare a costurilor căilor spre rețele	Informația conține valoarea exactă a costului căilor spre rețele adiacente
Informația e transmisă regulat (la intervale fixe de timp)	Informația e transmisă când apar modificări în rețea
Un ruter determină hopul următor prin execuția algoritmului Bellman-Ford	Un ruter construiește topologia și apoi utilizează un algoritm pentru determinarea hopului următor

2.5 Sisteme autonome

Un sistem autonom reprezintă:

- un grup de rutere care schimbă între ele informație de rutare utilizând același protocol de rutare;
- un set de rutere și rețele aflate în administrația unei organizații.

Protocoloalele utilizate pentru comunicarea în cadrul unui sistem autonom se numesc protocoale de rutare internă, protocoalele utilizate pentru comunicarea între sisteme automate se numesc protocoale de rutare externă. Protocoloalele de rutare internă descoperă căi între rețele, protocoalele de rutare externă descoperă căi între sisteme autonome. RIP și OSPF sînt protocoale de rutare internă, EGP - Exterior Gateway Protocol, IDRP - InterDomain Routing Protocol sînt protocoale de rutare externă.

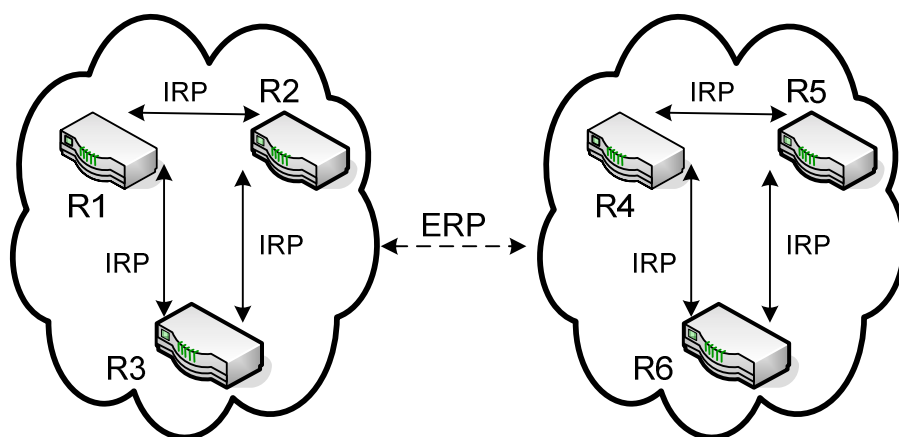


Figura 5.1 Sisteme autonome, protocoale de rutare internă și externă

2.6 Protocolul RIP (Routing Information Protocol)

Documentul RFC 1058 definește protocolul RIPv1 (Routing Information Protocol). RIP folosește drept metrică numărul de hopuri sau rutere până la rețeaua destinație. Pentru a se evita efectele negative ale buclelor logice a fost stabilită o metrică maximă, astfel încât orice informație despre o rută cu o metrică mai mare de 15 este ignorată.

Actualizările se fac implicit la 30 de secunde, acest timp reprezentând un compromis între timpul de convergență și cantitatea de bandă utilizată pentru actualizări. Astfel timpul de convergență la RIP în cel mai defavorabil caz este de 7 minute jumătate, calificând RIP în categoria protocoalelor de rutare internă cu o convergență scăzută. În măsura în care se impune un timp de convergență mai mic, perioada de actualizare poate fi redusă, ducând astfel și la un consum mai ridicat de lățime de bandă.

Documentul RFC 2453 definește protocolul RIPv2 care oferă suport pentru VLSM (Variable Length Subnet Mask), autentificarea rutei și transmiterea în maniera multicast a actualizărilor rutelor. Actualizările sînt tot la 30 de secunde, se pastrează limita de 15 hopuri și utilizarea portului 520 UDP. RIPv2 se utilizează în rețele de dimensiuni mici în care e necesar VLSM.

Actualizările se fac transmițând toate informațiile de rutare și nu doar cele ce s-au modificat de la ultima actualizare, dar sunt trimise folosindu-se adrese de difuzare, adică pachetele de actualizare vor ajunge doar la ruterele adiacente, deoarece în mod implicit ruterele filtrează pachetele de broadcast.

Fiecare ruter ce primește un pachet de actualizare va incrementa metrica fiecărei rute conținute în pachet cu 1, iar apoi pentru fiecare dintre rute va încerca să determine dacă nu există deja o rută cu o metrică mai bună către aceeași destinație în tabela de rutare.

Protocolul RIP previne intrarea în buclă infinită de rutare prin definirea unui număr maxim (15) de hopuri între sursă și destinație. Dacă în urma actualizărilor, metrica unei căi ajunge la valoarea infinită (16), rețeaua destinație este considerată inaccesibilă. Dezavantajul acestei caracteristici de stabilitate îl constituie limitarea la 16 hopuri a diametrului rețelei în care se configurează RIP.

Timere RIP: de actualizare, de route-timeout și route-flush. Timerele de actualizare măsoară timpul între două transmisii succesive de informație de rutare (30 de secunde). Fiecare intrare în tabela de rutare are asociat un route-timeout, cînd acesta expiră ruta e marcată ca fiind invalidă dar este menținută în tabelă pînă la expirarea lui route-flush.

Protocolul implementează mecanismul de spargere a orizontului (split horizon) pentru a bloca transmiterea informațiilor de rutare incorecte.

Să considerăm topologia din figura 5.2 în care toate link-urile au costul 1. B menține o distanță de 2 pînă la rețeaua 5, cu următorul hop nodul D. A și C mențin o distanță de 3 pînă la rețeaua 5 cu următorul hop nodul B. În cazul în care ruterul D cade se desfășoară următorul scenariu:

1. nodul B determină că rețeaua 5 este inaccesibilă prin nodul D și setează o distanță de 4 pe baza informațiilor primite de la A și C (B primește informații de la A și C: rețeaua 5 este la distanță 3 de noi). În următorul interval de actualizare, B transmite informația nodurilor A și C;
2. A și C își actualizează tabela de rutare (rețeaua 5 e accesibilă și are asociată metrica $5 = 4 + 1$ (4 plus 1 pentru B));
3. în următorul interval de actualizare, B primește distanța 5 și trece în tabela de rutare distanța 6.

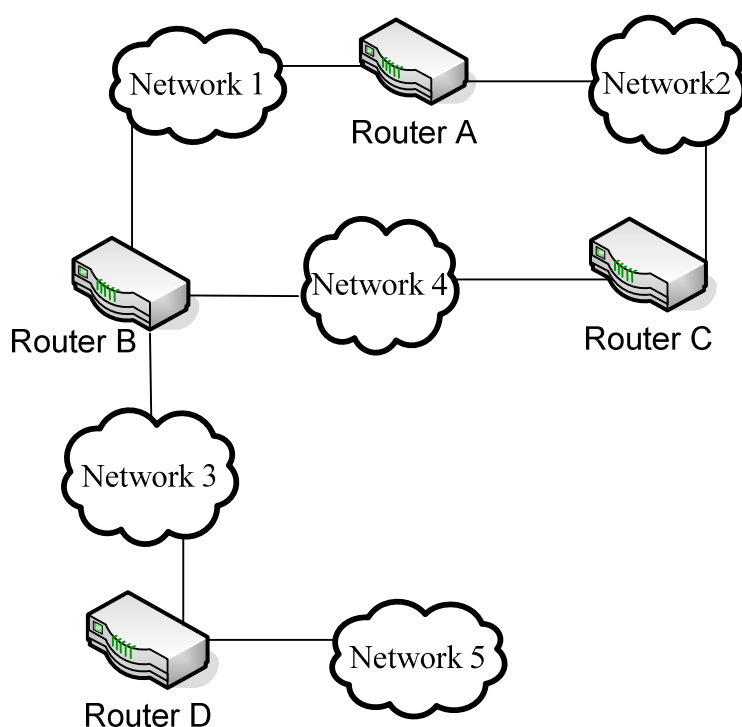


Figura 5.2 *Scenariu*

Acest pattern continuă pînă cînd valoarea distanței ajunge la 16 și nodul decide că rețeaua destinație este inaccesibilă. Problema numărării la infinit este declanșată de neînțelegerea între nodurile B și A (B și C) - fiecare crede că poate ajunge la rețeaua 5 prin celalalt.

Regula de “spargere” a orizontului impune ca un ruter să nu transmită informația de rutare înapoi, în direcția din care a primit-o, dacă ruterul de la care a primit această informație este mai aproape de destinație.

Regula de otrăvire a sursei impune ca un ruter să transmită vecinilor actualizări cu metrica 16 pentru rutele învățate de la ei.

3. Desfășurarea lucrării

Se va testa utilizarea protocolului RIP într-o configurația de rețea ca cea din figura 5.3.

3.1 Configurați calculatoarele cu adresele IP indicate.

3.2 Configurați interfețele ruterelor utilizând calculatoarele ca și console. Nu uitați rata ceasului!

3.3 Configurați o ruta statică între Router1 și Router2.

3.4 Verificați comanda `ping` între Computer1 și Computer2. Executați `tracert -d adresa_IP` pentru vizualizarea rutei.

3.5 Configurați protocolul RIP pe rutere: rute dinamice între Router1 și Router3 respectiv Router2 și Router3.

3.6 Verificați comanda `ping` între Computer1 și Computer2. Utilizați `tracert` pentru a stabili ce ruta a fost aleasă.

3.7 Dezactivați ruta statică între Router1 și Router2.

3.8 Verificați comanda `ping` între Computer 1 și Computer 2. Executați `tracert -d ip_address` pentru vizualizarea rutei.

3.9 Configurați RIP astfel încât să existe o rută dinamică între Router1 și Router2.

3.10 Verificați comanda `ping` între Computer1 și Computer2.

Executați `tracert -d ip_address` pentru vizualizarea rutei. Ce rută a fost aleasă? De ce?

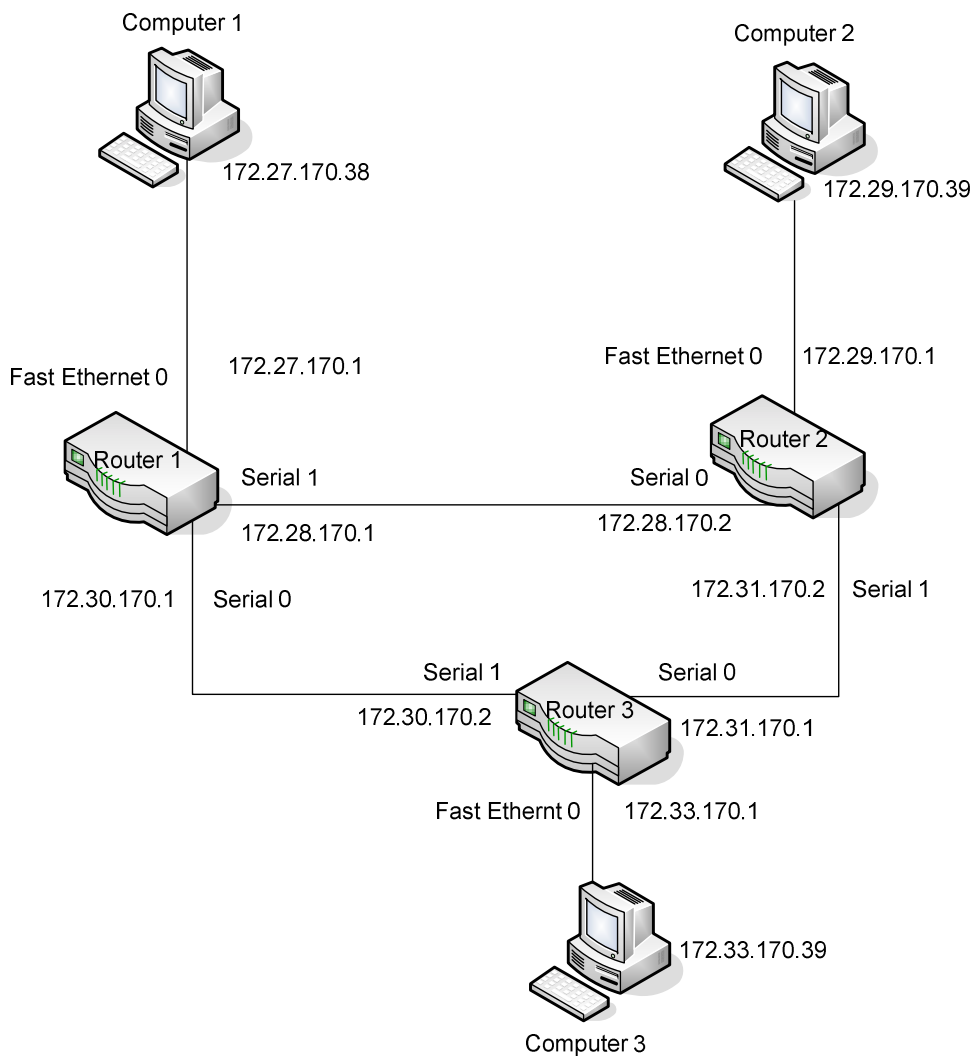


Figura 5.3 Configurația laboratorului

Notițe