**Rețele de calculatoare – proiect**

**Aplicație pentru descoperirea unei topologii de rețea pe baza mecanismului de comunicație RIPv2**

**Profesor coordonator:** **Studenți:**

Nicolae-Alexandru Botezatu Cucoș Andreea, grupa 1307A

Pristanda Amalia-Maria, grupa 1307A

**Cuprins**

**Cap 1. Generalități**

**1**.1 RIPv2

**1.2** Concepte de bază

**1.3** Forma unui mesaj de tip RIPv2

**1.4** Cum funcționează

**1.5** Importanța topologiilor

**1.6** Tipuri de topologii

1.6.1 Topologia punct la punct

1.6.2 Topologia magistrală

1.6.3 Topologia inel

1.6.4 Topologia stea

1.6.5 Topologia mesh

1.6.6 Topologii mixte

**Cap 2. Descoperire topologie – RIPv2**

**2.1** Creare topologie

**2.2** Verificare funcționalitate

**Cap 3. Bibliografie**

**Capitolul 1: Generalități**

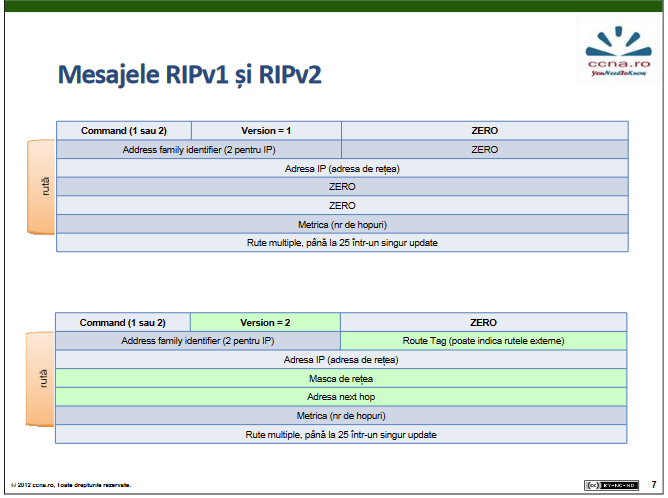
**1.1 RIPv2**

* Protocolul de Rutare a informatiei(RIP) este un protocol de rutare tip distanță-vector care presupune utilizarea ca metrică de rutare a numărului de pași de rutat, numit și hop count. Acest protocol previne apariția buclelor de rutare, utilizând o valoare limită maximă ca numar de pași de rutare pe calea de la sursă la destinație, această limită fiind în general 15
* RIP este utilizat în scopul schimbului de informații de rutare între routerele dinamice pe rețelele IP(Internet Protocol) sau IPX(Internetwork Packet Exchange). Acesta a fost conceput în anul 1980 pentru a fi utilizat împreună cu protocolul XNS(Xerox Network System), dar cel mai frecvent este utilizat în rețelele de internet TCP/IP de dimensiuni mici și mijlocii

**1.2 Concepte de bază**

* Întreaga tabelă de rutare este trimisă la toate routerele conectate din 30 în 30 de secunde
* RIPv2 aduce în plus față de RIPv1 posibilitatea de a transmite și informația de subrețea suportând astfel adresarea CIDR. Pentru a evita încărcarea inutilă a gazdelor ce nu sunt implicate în procesul de rutare, ruterele vor transmite tabela de rutare doar routerelor adiacente utilizând o adresă de tip multicast, 224.0.0.9
* Se utilizează update-uri periodice pentru a menține informația din tabelele de rutare corectă. Aceste update-uri sunt utilizate și în cazul în care un router nu mai funcționeaza în acest interval de 30 de secunde
* În momentul în care schimbăm topologia, folosim update-uri provocate pentru a propaga informația la noduri
* RIPv2 ne permite să realizăm autentificarea la nivel de interfață
* RIP realizează “routing by rumor” și este mult mai predispus la crearea buclelor față de celelate protocoale de rutare, deoarece acesta trimite întreaga sa tabelă la routerele adiacente. O dată primită aceasta tabelă, toate celelalte routere fac același lucru și, deoarece nu există o relație reală de vecin sau un calcul al rutelor, routerele au puține cunoștințe directe despre rețelele disponibile. La finalul acestui transfer, mesajul primit nu este identic cu cel inițial
* Datorită faptului ca acest protocol de rutare este de tip “distance vector” informația de rutare este determinată pe baza unui algoritm de tipul “shortest path”, spre exemplu algoritmul Bellman-Ford care determină calea optima de a ajunge la un nod distant ținând cont de câteva criterii de cost bine definite(numărul de hop-uri(noduri intermediare), lărgime bandă a conexiunii între 2 routere, etc.)
* Dezavantaj al protocoalelor de rutare: trimit update-urile folosind pachete IP deci există probleme de securitate. Un router poate să accepte update-uri invalide inițiate de la un atacator, un router care este incorect configurat sau echipamentul conectat la rețea, care rulează un protocol de rețea, fără stiința utilizatorului.
* O soluție pentru această problemă este să se folosească autentificarea între routerele care trimit informații, asigurând astfel că routerele vor accepta numai pachetele trimise de surse care cunosc datele de autentificare. Această autentificare se realizează la nivel de interfață și nu va cripta tabela în momentul trimiterii
* Configurarea autentificării in RIPv2 presupune crearea unui key chain cu nume si cu cel puțin o cheie(parola). Activarea are loc pe interfață. Un key-chain este un set de parole care sunt utilizate pentru autentificare. Pentru a nu întampina erori este necesar sa se folosească același set de parole pe interfețele asociate aceleiași legături. Parolele din key-chain sunt rotite periodic după o regulă cunoscută de ambele echipamente și pot fi transmise în clear-text sau folosind un hash md5.

**1.3 Forma unui mesaj de tip RIPv2**



* Command: indică tipul pachetului, cerere sau răspuns. Cererea presupune ca routerul să trimită întreg sau o parte din tabelul de rutare. Mesajele de răspuns conțin intrări de rută. Mesajul de răspuns este setat periodic sau ca răspuns la o solicitare.
* Version: specifică versiunea de RIP. Este 2 pentru RIPv2 si 1 pentru RIPv1;
* Address Family Identifier: specifică familia de adrese. RIP este conceput pentru a transporta informații de rutare pentru mai multe protocoale. Fiecare intrare are un identificator de familie de adresă pentru a indica tipul de adresa specificat. Identificatorul familiei de adrese pentru IP este 2. AFI este setat 0xFFF pentru prima intrare pentru a specifica faptul că restul intrării conține informații de autentificare;
* Route Tag este utilizat în scopul marcării rutelor care au fost importante(redistribuite) din alte protocoale de rutare. În momentul în care un router primește informații despre o rețea ca fiind importantă, acesta va conserva valoarea acestui câmp;
* IP adress: specifică adresa IP a destinației;
* Subnet Mask: conține informația de subrețea a destinației. Dacă acest câmp este 0 înseamnă ca nu a fost specificată nicio informație de subrețea la intrare;
* Next-hop: indică adresa IP a următoarei rețele intermediare (hop) către care sunt trimise pachetele. Daca acest câmp e setat la 0.0.0.0 adresa de la care se trimite update-ul este cel mai bun next-hop;
* Metric: indică câte routere intermediare au fost folosite pentru a ajunge la destinație.

Această valoare este între 1 si 15 pentru o rută validă sau 16 pentru o rută neaccesibilă sau infinită

**1.4 Cum funcționeză**

* După pornirea configurației nevolatile – memoria spune unui router care rulează protocolul RIP la ce rețea este conectat în mod direct. Ulterior această informație este încărcată in tabela sa de rutare;
* Routerele direct accesibile vor primi update-ul, își vor actualiza tabelele de rutare și vor genera propriile update-uri (actualizări de rutare) care vor reflecta modificările corespunzătoare;
* În cazul în care un update prezintă o valoare mai bună decât cea stocată în prezent în tabel, tabela de rutare trebuie actualizată cu această nouă informație. Această actualizare nu are grijă dacă expeditorul acesteia este, de asemenea, routerul care este în prezent selectat drept next-hop;
* Dacă o actualizare de rutare oferă o valoare mai rea decât cea stocată în prezent în tabel, tabela de rutare trebuie actualizată cu această informație dacă expeditorul acesteia este next-hop pentru această rețea(adresa next-hop din tabela este identică cu adresa sursă a update-ului). Actualizările de rutare cu o metrică mai rea sunt relevante doar dacă cineva de la acel router este menționat în intrarea efectivă în tabel;
* Mai pe scurt:
  + Fiecare router deține un director, care indică către alte rețele(intrările next-hop din tabelă) fără a cunoaște exact locația acestora;
  + Datagramele urmează aceste “indicatoare” și în cele din urmă ating obiectivele lor;
  + Informațiile despre aceste indicatoare se bazează pe actualizările de rutare.

**1.5 Importanța topologiilor**

Prin topologia unei rețele se înțelege modul de interconectare a calculatoarelor în rețea. Folosirea unei anumite topologii are influență asupra vitezei de transmitere a datelor, a costului de interconectare și a fiabilității rețelei

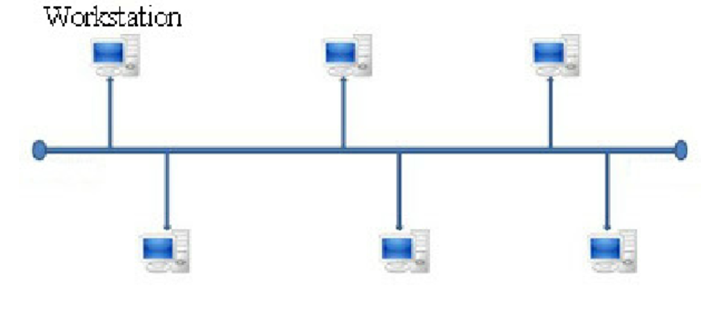
**1.6 Tipuri de topologii**

**1.5.1.** Topologia punct la punct

* cea mai simplă topologie, dar şi cu redundanţă minimă (redundanţa = 0, nu există nicio legătură de rezervă) → fiabilitate scazută

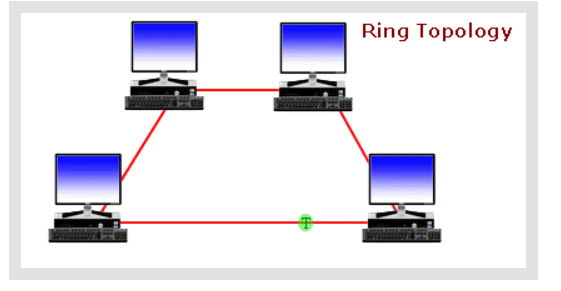
**1.5.2** Topologia magistrală (bus)

* se mai numește și magistrala liniară, deoarece există un singur cablu care leagă toate calculatoarele din rețea
* circulația pachetelor se face în ambele sensuri, fiecare calculator putând să transmită și să recepționeze
* cea mai folosită atunci când se realizează rețele locale de mici dimensiuni, iar performanțele nu trebuie să fie spectaculoase



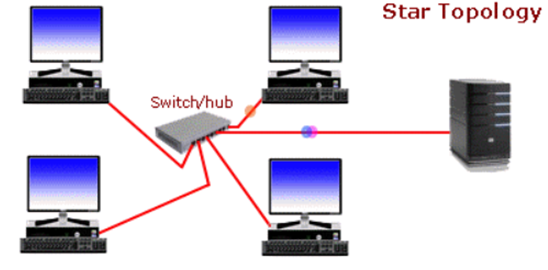
**1.5.3** Topologia inel

* conectează calculatoarele printr-un cablu în formă de buclă (nu există capete libere)
* defectarea unui calculator afectează întreaga rețea
* este o topologie activă în care calculatoarele regenerează semnalul și transferă datele în rețea, fiecare calculator funcționează ca un repetor, amplificând semnalul și transmițându-l mai departe; iar dacă îi este destinat îl copiază
* mesajul transmis de către calculatorul sursă este retras din bucla de către același calculator atunci când îi va reveni după parcurgerea buclei
* transmiterea datelor se face prin metoda jetonului (token passing). Cea mai cunoscută topologie inel este Token - ring de la IBM
* Token ring constă în controlul accesului la rețea prin pasarea unui jeton digital secvential de la o stație la alta. Când o stație primește accesul (jetonul), poate trimite date în rețea. Dacă stația nu are date de transmis, pasează mai departe jetonul următoarei stații și procesul se repetă



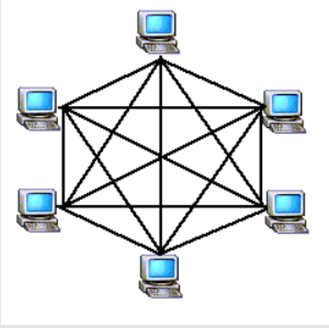
**1.5.4** Topologia stea

* folosește un calculator central care va fi conectat cu toate celelalte calculatoare prin cabluri directe
* toate transferurile de date se realizează prin intermediul calculatorului central
* dacă se folosește un calculator central de mare putere, atunci rețeaua va avea performanțe ridicate, însă defectarea acestuia duce la oprirea rețelei



**1.5.5** Topologia mesh

* cea mai complicată topologie, cu redundanţă maxima (există o legătură a unui nod cu toate celelalte noduri din reţea)→cel puţin 2 căi între oricare două noduri
* costul este ridicat(echipamente și cabluri)
* avantaj : număr mare de conexiuni -> fiabilitate ridicată



**1.5.6** Topologii mixte

* topologiile menţionate mai sus pot fi la rândul lor combinate şi rezultă sisteme deosebit de complexe aşa cum este cazul reţelei Internet (WAN), care este “o reţea de reţele”
* exemplu de reţea mixtă = topologia arbore (Tree), combină caracteristicile topologiilor punct-la-punct şi stea

**Capitolul 2: Descoperire topologie – RIPv2**

**2.1 Creare topologie**

Se vor utiliza socket-uri:

* socket = mecanism de comunicație in rețea care poate folosi orice protocol de comunicație(în general TCP/IP)
* fiecare nod va fi un socket

Pe baza modelului Client/Server:

* Procesul Server:
  + Oferă servicii de rețea
  + Acceptă cereri de la un proces client
  + Efectuează un anumit serviciu și returnează rezultatul
* Procesul Client:
  + Initializează comunicația cu serverul
  + Pentru ca un client să se poată conecta la server are nevoie de două informații adresa serverului și portul pe care ascultă aplicația la care se dorește conectarea
  + Cere un serviciu și așteaptă răspunsul serverului

**2.2 Verificare funcționalitate**

Utilizând VirtualBox:

* pentru a verifica funcționalitatea aplicației, aceasta va trebui rulată pe mai multe echipamente
* se poate crea o mașina virtuală, după care se clonează
* se realizeză setup-ul topologiei
* se copiază și se rulează aplicația pe fiecare mașina virtuală
* instanțele aplicației vor schimba date și vor descoperi topologia în maniera descrisă de RIPv2

**Capitolul 3: Bibliografie**

<https://ro.wikipedia.org/wiki/RIP>

<https://www.techrepublic.com/article/cisco-administration-101-know-the-basics-about-ripv2/>

<https://networkencyclopedia.com/routing-information-protocol-rip/>

<https://www.ict.tuwien.ac.at/lva/384.081/infobase/P40-RIP_v4-6.pdf>

<http://ham.elcom.pub.ro/rs/platf/rs-lab2.pdf>

<http://193.226.51.37/down/cisco/ccna2/C7.pdf>

http://calin.comm.pub.ro/Didactice/ARI/Notite%20curs/Prez/Ro/ARI\_R\_c4\_Topol.pdf

http://retele-de-calculatoare.weebly.com/21-topologia-retelelor.html

<https://www.scritub.com/stiinta/informatica/retele/Topologii-de-retea51637.php>

<https://www.yumpu.com/ro/document/read/39837221/4-retele-de-calculatoare>

https://wiki.dcae.pub.ro/index.php/Socket-uri\_de\_re%C8%9Bea