# Antetu protocolului IPV4

# 

# Enumerati deficientele protocolului IPV4

Cea mai importanta deficienta a protocolului IPV4 este aceea ca numarul de adrese este insuficient pentru a face fata numarului de dispozitive conectate al Internet. Alta deficienta a acestui protocol faptul ca antetul are antetul foarte coplicat, avand 13 campuri dintre care unele nu sunt folosite (tip serviciu).

# Precizati cum rezolva IPV6 deficientele IPV4

IPV6 rezolva problemele versiunii IPV4 deorecere are: adrese mai lungi decat IPV4 (16 octeti fata de 4 la IPV4). Antetul protocolului IPV6 este simplificat fata de cel al versiunii precedente (7 campuri fata de 13). De asemenea protocolul IPV6 permite distingerea tipurilor de trafic, pentru a fi tratate distinct unele de altele, spre exemplu pachete care au diverse cerinte de livrare in timp real. O alta facilitate a protocolului IPV6 este faptul ca poate permite pachete cu lungime de aproximativ 4GB (2^32 – 1) numite jumbograme.

# Descrieti modul in care decurge fragmentarea in IPV4 si IPV6 (MTU).

La IPV4, cand un pachet este prea mare pentru urmatoarea legatura peste care va trece, poate fi fragmentat de expeditor (gazda sau ruter). Fragmentarea este posibila numai daca bit-ul DF (Don't fragment) nu este setat. Toate fragmentele cu exceptia ultimului au bitul MF (more fragments) setat pentru a indica faptul ca urmeaza sa mai soseasca fragmente din aceeasi datagrama. Daca bitul DF din antet este setat si este necesara fragmentarea lui, se trimite inapoi la sursa un pachet cu mersajul: “Destination Unreachable”.

Pentru IPv6, fragmentarea poate avea loc numai la nodul sursa, iar reasamblarea este facuta doar la nodul destinatie. Este folosit antetul extensiei de fragmentare.

# Descrieti relatia intre protocoalele IPV4 si ICMP, preziand la ce se foloseste ICMP

ICMP foloseste IPV4 pentru transmisie.

IPV4 foloseste ICMP pentru raportarea de erori.

# 5. NAT

NAT = translatarea adreselor de retea

Ideea de bază a NAT-ului este de a aloca fiecărei companii o singură adresă IP (sau cel mult un număr mic de adrese) pentru traficul Internet. În interiorul companiei, fiecare calculator primeşte o adresă IP unică, care este folosită pentru traficul intern. Totuşi, atunci când un pachet părăseşte compania şi se duce la ISP, are loc o translatare de adresă. Pentru a face posibil acest lucru, au fost declarate ca fiind private trei intervale de adrese IP. Companiile le pot folosi intern aşa cum doresc. Singura regulă este ca nici un pachet conţinând aceste adrese să nu apară pe Internet. Cele trei intervale rezervate sunt :

10.0.0.0 - 10.255.255.255/8 (16.777.216 gazde)

172.16.0.0 - 172.31.255.255/12 (1 .048.576 gazde)

192.168.0.0 - 192.168.255.255/16 (65.536 gazde).

# 6. Apreciati daca este un protocol eficient

Din punctul meu de vedere NAT este un protocol optim. Protocolul incalca una dintre regulile pe baza carora este construita stiva de prtocoale ISO/OSI, si anume faptul ca un nivel trebuie sa fi independent de celelalte, deoarece se foloseste de portul furnizat de protocoalele de la niveul transport. Folosind prtocolul NAT identificare unica a fiecarui calculator din internet cade. In ciuda acestor lucruri, nu ajutorul protocolului se mareste considerabil numarul de adrese ce pot fi folosite pentru identificare dispozitivelor conectate la internet.

# 7. Descrieti functionarea la trimiterea si apoi receptionarea unui pachet

La transmisie:

* Intra in unitatea NAT si adresa sura 10.x.yz este inlocuita cu adresa reala a companiei
* Campul TCP Port sursa este inlocuit cu un index in tabla de translatare a unitatii NAT (65.536 intrari) – contine adresa IP initiala si portul initial
* Recalculeaza si insereaza in pachet sumele de control ale antetelor IP si TCP

La primire:

* Extrage din antetul TCP portul sursa, il foloseste ca index in tabela de corespondenta a unitatii NAT
* Din intrare sunt extrase si inserate in pachet, adresa IP interna si portul sursa TCP initial
* Recalculeaza si insereaza in pachet sumele de control ale antetelor IP si TCP
* Pachetul este transferat ruterului companiei pentru transmitere normală folosind adresa 10.x.y.z.

1. **Dezavantaje NAT**

* În primul rând,NAT violează modelul arhitectural al IP-ului, care afirmă că fiecare adresă IP identifică unic o singură maşină din lume. Întreaga structură software a Internetului este construită pornind de la acest fapt. Cu NAT, mii de maşini pot folosi (şi folosesc) adresa 10.0.0.1.
* În al doilea rând, NAT schimbă natura Internetului de la o reţea fără conexiuni la un fel de reţea cu conexiuni.
* În al treilea rând, NAT încalcă cea mai fundamentală regulă a protocoalelor pe niveluri: nivelul k nu poate face nici un fel de presupuneri referitor la ceea ce a pus nivelul k+1 în câmpul de informaţie utilă. Principiul de bază este să se păstreze niveluri independente.
* În al patrulea rând, procesele din Internet nu sunt obligate să folosească TCP sau UDP.
* În al cincilea rând, anumite aplicaţii introduc adrese IP în corpul mesajului. Receptorul extrage aceste adrese şi le foloseşte. De vreme ce NAT nu ştie nimic despre aceste adrese, nu le poate înlocui, deci orice încercare de a le folosi de către cealaltă parte va eşua.

# Table de dirijare

Orice pachet conține o adresa IP a destinatarului, cu două părți.

Un pachet este transmis de la sursă la destinație trecând prin noduri intermediare (rutere), fiecare legând între ele cel puțin două rețele

Rol ruter – primește un pachet și

• îl livrează gazdei de destinație (dacă este in aceeași rețea). Toate nodurile care au aceeași adresa\_retea sunt situate in aceeasi retea fizica și pot comunica direct prin legătura de date (transmit cadre)

• altfel, il re-transmite (forward) către un alt nod NextHop. Folosește tabela de dirijare (rutare) care are intrări de forma

# Despre dirijari (clasificari)

1. Fara tabela de dirijare: inundare, hot potato
2. Cu tabele de dirijare-criterii:
   1. Adaptare la conditii de trafic: static, dinamica
   2. Locul unde se fac calculele: descentralizata, centralizata, distribuita
   3. Criterii de dirijare: calea cea mai scurta, intarzierea medie globala, folosirea eficienta a resurselor, echitabilitatea
   4. Informatii schimbate intre noduri: starea legaturii, vectorul distantelor
   5. Tipul retelei: uniforma, ierarhica

# Descrieti rolul dirijarii si precizati cand si cum efectiv revine in practica.

Dirijarea pachetelor de la masina sursa la masina destinatie este functia principal a nivelului retea. Este importanta atunci cand destainatia si sursa nu sunt in aceeasi retea.

# Cum functioneaza algoritmul de dirijare ad-hoc

Retelele de noduri care se afla (intamplator) aproape unul de celalalt sunt numite retele ad-

hoc.

Algoritmul de dirijare in acest tip de retele se numeste AODV (Ad-hoc On deman Distance

Vector). Reteaua este reprezentata printr-un graf (intre 2 puncte exista o muchie daca cele doua puncte compunica direct).

Fiecare nod contine:

1. Tabela de dirijare (destinatie, next hop, distanta)
2. Tabela history (identitatile cererilor precedente)
3. Tabela reverse route (cale spre sursa unui pachet)

* Daca un nod nu are destinatia in tabela sa trebuie sa o descopere → difuzeaza un pachet ROUTE REQUEST (identificat unic prin Source address + Request ID)
* Atunci cand un pachet ROUTE REQUEST soseste intr-un nod acesta:

→ verifica dupa duplicat in tabela history locala

→ trimite ROUTE REPLAY pe legatura inversa daca a gasit ruta noua sau incrementeaza hop-count si redifuzeaza ROUTE REQUEST si memoreaza informatia in reverse route tabele

# Ce structuri de date pastreaza ruterele din cadrul unei retele ad-hoc

Fiecare nod contine:

1. Tabela de dirijare (destinatie, next hop, distanta)
2. Tabela history (identitatile cererilor precedente)
3. Tabela reverse route (cale spre sursa unui pachet)

Acestea pot fi reprezentate de exemplu cu ajutorul matricilor sau a unor tabele de corespondenta.

# Ce actiuni executa un ruter dintr-o retea ad-hoc la primirea unui mesaj sau a unei confirmari

Verifica daca destinatia el este destinatia. Daca nu, verifica daca destinatia se gaseste in tabela sa de dirijare. Daca da, trimite pachetul mai departe. Daca nu trebuie sa descopere ruta, asa ca difuzeaza un pachet ROUTE REQUEST. Atunci cand primeste un pachet ROUTE REPLAY completeaza tabela de dirijare. Si trimite pachetul pe cale nou descoperita.

# Dirijarea folosind vectorul de distanta (RIP):

Algoritmul de dirijare cu vertori distanta presupune ca fiecare ruter mentine o tabela (de exemplu un vector) in care pastreaza cea mai buna distanta cunoscuta spre fiecare destinatie si linia care trebuie urmata. Aceste tabele sunt actualizate prin schimbul de informatii intre nourile vecine (o data la fiecare T msec fiecare ruter trimite spre fiecare vecin o lista a estimarilor proprii spre fiecare destinatie).

Tabela veche nu intervine in calcule. Problema numararii la infinit

Dirijarea folosind vectori distanta functioneaza in teorie, insa in practica are o limitare importanta: desi ea converge catre rezultatul corect ea o face foarte lent.

Solutii:

→ simple split horizon: omite rutele invatate de la un cevin in actualizarile trimise acestuia

# Dirijarea ierarhica

Ruterele un impartite in zone. Fiecare ruter cunoaste detaliile necesare pentru a dirija pachetele spre destinatie in cadrul regiunii sale, dar nestiind nimic despre organizarea interna a celorlalte regiuni.

# CIDR

# Dirijarea fara clase intre domenii

# Ideea de la baza CIDR, descrisă în RFC 1519, este de a aloca adresele IP rămase, în blocuri de dimensiune variabilă, fără a se ţine cont de clase.

* Reguli pentru alocare adresele
  + - lungimea blocului trebuie sa fie putere a lui 2
    - adresa de inceput a blocului de adresare alocat trebuie sa fie multiplu de dimensiunea acestuia
* Algoritmul de forwarding – poate fi aplicat tuturor adreselor, deci folosind CIDR vechile clase A,B,C nu mai sunt folosite pentru dirijare; pentru a accelera cautarea => trucuri de indexare
  + Intrare in tabela de intrare: <adresa\_retea, masca, next\_hop>
  + Adresa de retea: adresa\_ip & masca.
  + Fiecare intrare este extinsa cu o masca de 32 biti
  + Soseste un pachet IP, se extrage adresa IP.
  + Tabela de rutare este scanata intrare cu intrare, mascand adresa destinatie si comparand cu intrarea din tabela
  + Daca 2 intrari se potrivesc, cea care are in masca mai multi biti 1 castiga

# OSPF

Internet-ul este construit dintr-un număr mare de sisteme autonome. Fiecare AS este administrat de o organizaţie diferită şi poate folosi propriul algoritm de dirijare în interior.

Multe din AS-urile din Internet sunt foarte mari şi nu sunt simplu de administrat. OSPF le permite să fie divizate în zone numerotate, unde o zonă este o reţea sau o mulţime de reţele învecinate. Zonele nu se suprapun şi nu este necesar să fie exhaustive, în sensul că unele rutere pot să nu aparţină nici unei zone. O zonă este o generalizare a unei subreţele. În afara zonei, topologia şi detaliile sale nu sunt vizibile.

Un algoritm de dirijare în interiorul unui AS este numit protocol de porţi interioare;

→ protocol de dirijare pentru portile intrioare (in cadrul unui AS)

→ suporta 3 tipuri de conexiuni si retele:

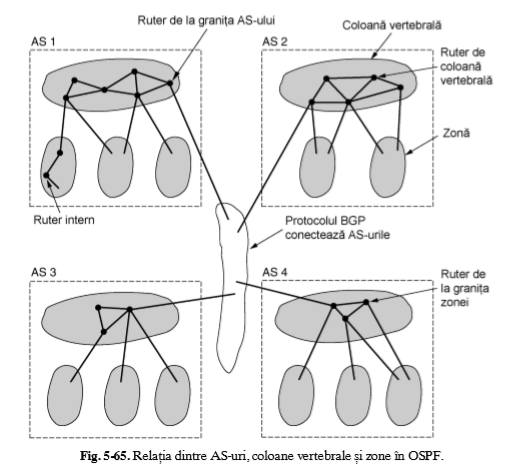
1. Linii punct-la-punct între exact două rutere.
2. Reţele multiacces cu difuzare (de exemplu, cele mai multe LAN-uri).
3. Reţele multiacces fără difuzare (de exemplu, cele mai multe WAN-uri cu comutare de pachete).

→ functioneaza prin abstractivizarea colectiei de retele, rutere si linii reale intr-un graf orientat in care fiecarui arc ii este atribuit un cost, apoi calculeaza cea mai scurta cale bazandu-se pe ponderile arcelor

→ reprezinta reteaua sub forma de graf si calculeaza cel mai scurt drum de la fiecare ruter la fiecare ruter.

→ divizeaza AS in zone (topologia si detaliile zonei nu sunt vizibile in afara ei).

Fiecare AS are o coloana vertebrala numita zona 0. Un ruter care conecteaza 2 sau mai multe zone, face parte din coloana vertebrala. Astfel este posibil sa se ajunga din orice zona in

orice alta zona din AS prin coloana vertebrlala.

→OSPF distinge 4 clase de rutere:

1. interne (integral in interiorul unei zone)
2. de la granita zonei (conecteaza 2 sau mai multe zone)
3. de coloana vertebrala
4. de granita AS (discuta cu ruterele din alte AS)

# BGP

++++++++++ALGORITM

→ Protocolul portilor de granita

→Din punctul de vedere al unui ruter BGP, lumea constă din AS-uri şi liniile care le conectează. Două AS-uri sunt considerate conectate dacă există o linie între două rutere de graniţă din fiecare. Dat fiind interesul special al BGP-ului pentru traficul în tranzit, reţelele sunt grupate în trei categorii.

1. Retelele ciot – care au doar o conexiune la graful BGP (nu pot fi folosite pentu traficul in tranzit pentru ca nu este nimeni la capatul celalalt).
2. Retelele multiconectate – pot fi folosite pentru trafiul in tranzit, cu exceptia a ceea ce le refuza
3. Retelele de tranzit – ex. Coloanele vertebrale, sunt doritoare sa manevreze pachetele altora, eventual cu unele restrictii, si de obicei pentru o plata

→ Perechile de rutere BGP comunica intre ele stabilind conexiuni TCP – ofera comunicatie sigura si ascunde toate detaliile retelelor traversate

→BGP are la baza un protocol bazat pe vectori distanta.

→Rezolva usor problema numara-la-infinit

→Tabelele de dirijare contin si rutere spre destinatie