UNIVERSITATEA TEHNICĂ CLUJ-NAPOCA

DEPARTAMENTUL ELECTROTEHNICĂ ŞI MĂSURĂRI

Cursul 10 PROTOCOALELE NIVELULUI OSI 3 (INTERNET) IP ŞI ICMP

Introducere curs	În acest curs sunt prezentate protocoalele nivelului OSI3 (Internet).

Obiective curs	• IP (Internet Protocol).
	  • Antetul pachetului IP.
	  • Formarea şi transferul unui pachet în rețea.
	• Internet Control Message Protocol (ICMP).
	• Adresarea IP.
	  • Formatul adresei IP
	  • Masca de rețea
	  • Adresa de broadcast.

Durată medie de studiu individual	Durata medie de studiu individual: 100 minute. Acest interval de timp presupune parcurgerea continutului cursului şi rezolvarea testelor de autoevaluare

Cuprins:

10.1. IP (Internet Protocol). 1
10.2. Internet Control Message Protocol (ICMP). 7
10.3. Adresarea IP. 9
TESTE DE AUTOEVALUARE 13
RĂSPUNSURI 14
Bibliografie 14

În cele ce urmează se vor analiza componentele de nivel OSI 3 (rețea), **IP** (Internet Protocol) şi ICMP *(Internet Control Messages Protocol)*, din suita **TCP/IP**. Se vor prezenta informațiile de control înscrise în antetele datagramelor, modul în care se formează acestea şi câteva elemente legate de depanarea protocolului [1].

Cunoaşterea în detaliu a acestui protocol, ne permite să înțelegem mai bine procesele de transmitere şi rutare a pachetelor în rețea.

Datorită faptului că **TCP/IP** a fost conceput în mod compact, nivelul Internet (echivalent cu cel de **Rețea**, OSI 3) nu este absolut independent de nivelele superioare, în antetul pachetelor **IP** se găsesc şi informații referitoare la acestea. Cunoscând procesul de asamblare a unei datagrame şi cum este transmisă de-a lungul nivelelor, se poate înțelege mai bine cum funcționează componentele, de nivel înalt, ale suitei. O analiză finală, în paralel a capitolelor corespunzătoare, este binevenită.

10.1. IP (Internet Protocol).

Protocolul de Internet (**IP**) reprezintă protocolul de transfer, corespunzător nivelului OSI 3 (rețea) şi al nivelului Internet, din modelul de referință TCP/IP. El face parte din categoria

protocoalelor "rutabile", adică poate fi rutat de-a lungul unor căi formate din porţi de acces, datorită informaţiilor conţinute în antet.

Din această categorie mai face parte şi protocolul **IPX** (*Internetwork Packet Exchange*), dezvoltat cu succes de firma Novell, la sfârşitul anilor '80. Acesta era folosit la transferul pachetelor în reţelele locale (**LAN**) bazate pe sisteme de operare, din familia NetWare. Deşi **IPX** este un protocol eficient pentru reţelele bazate pe server de fişiere, acesta a început să fie abandonat, în favoarea **TCP/IP**. Începând cu versiunea NetWare 5.0 firma Novell foloseşte **TCP/IP** ca protocol "nativ" (inclus în nucleul sistemului de operare ca protocol principal, nu auxiliar) de reţea. Până la această versiune **TCP/IP** era folosit doar pentru "tunelare" (încapsularea pachetelor **IPX** în pachete **IP**) la transmiterea datelor între serverul de fişiere şi terminale, sau între reţele.

Principalele sarcini ale IP sunt legate de adresarea datagramelor (care la acest nivel se numesc pachete) între terminale şi gestionarea fragmentării şi reasamblării lor. Formatul pachetelor acestui protocol are bine definit cu ajutorul antetului (*packet header*), care conţine informaţiile necesare despre dimensiunea şi conţinutul datelor transferate. Totodată, aceste informaţii au un rol major în rutarea pachetelor între porţile de acces (determinarea căii potrivite între terminalul sursă şi cel destinaţie).

IP nu are mecanisme specifice de control al fluxului de date şi nici nu garantează livrarea pachetelor la destinaţie. Din diferite motive, pachetele pot întârzia, s-au chiar pierde, pe drum.

IP nu foloseşte mijloace de verificare a integrităţii datelor transmise, cum ar fi sume ciclice de control, ci doar pentru verificarea integrităţii informaţiilor de control conţinute de antet. Sarcinile de verificare şi recuperare a datelor, sau de control al fluxului, revin nivelului transport (OSI 4).

În procesul de rutare, informaţiile conţinute de antetul **IP** sunt folosite, de protocoalele de rutare, ca să determine calea cea mai bună spre destinaţie, fără să poată verifica până la sfârşit dacă acest lucru este adevărat.

Pe baza unor algoritmi de complexitate ridicată, se decide spre care nod (ruter) din reţea trebuie trimis pachetul, considerat ca următorul pas (*next hop*) din calea spre destinaţie. După trimiterea pachetului spre acel nod, presupus a fi cel mai potrivit, nu mai există nici un control asupra „destinului” acelui pachet. De fapt, fiecare poartă de acces (ruter) decide doar pasul următor. Aparent, **IP** nu gestionează prea bine expedierea pachetelor, dar există componente, care în timpul manipulării pachetelor, generează mesaje de eroare în cazul în care acestea apar.

S-a văzut mai înainte cum sunt împărţite (fragmentate) datele în bucăţi mai mici pentru a putea fi transmise prin reţea, iar apoi reasamblate la destinaţie.

IP permite o dimensiune maximă a pachetului de 65535 octeţi, care este o cantitate mai mare decât poate manevra, în general, un protocol de nivel inferior (OSI 1 sau2). De aici apare şi necesitatea fragmentării datelor, proces pe care **IP** îl îndeplineşte în mod automat. Sunt situaţii când datagrame venite de la nivelul superior sunt împărţite în mai multe datagrame (pachete) mai mici, după principiul studiat mai înainte.

Când o datagramă, conţinând date dintr-un şir fragmentat, ajunge la un terminal destinaţie, nivelul IP al acesteia porneşte un cronometru incrementat cu timpul de reasamblare (*reassembly timer*). Dacă toate "piesele" datagramei respective sunt recepţionate în acel timp, aceasta este reasamblată, prelucrată şi datele trimise apoi nivelului superior. Dacă după trecerea timpului limită nu se pot recepţiona toate piesele, cele recepţionate deja sunt abandonate şi se cere retransmiterea completă a lor.

Terminalul destinaţie cunoaşte acest interval de timp şi ordinea în care sunt transmise fragmentele, datorită existenţei unor câmpuri dedicate acestora în antetul pachetului **IP**. Totuşi, şansa pe care o au datagramele fragmentate, de a ajunge la destinaţie, este mult mai mică decât a celor nefragmentate. De aceea, cele mai multe aplicaţii încearcă să evite pe cât posibil fragmentarea.

IP este un protocol neorientat spre conexiune (*connectionless*), ceea ce înseamnă că nu contează, la acest nivel, care este terminalul sursă sau destinaţie, pachetele "călătorind" în reţea doar în baza informaţiilor de rutare. Cu alte cuvinte, între terminalul sursă şi cel destinaţie, nu se stabileşte nici un fel de negociere de început, sau de sfârşit, de transmisie. De acestea se ocupă nivele superioare (Transport, Sesiune şi Aplicaţie, OSI 4,5,7).

Informaţiile conţinute de antet sunt folosite, la acest nivel, doar pentru rutare, restul fiind folosite de componenta de transport (**TCP** sau **UDP**, nivelul OSI 4). Versiunea 6 a protocolului **IP** (IPv6) poate manevra mai multe informaţii, aşa cum se va putea vedea mai târziu în acest capitol [1].

Antetul pachetului IP.

Procesul de asamblare a pachetului este asemănător cu cel folosit de tehnologia Ethernet când formează (încapsulează) cadrele, prin adăugarea la datele de transmis a informațiilor de control. Deosebirea constă în faptul că acestea din urmă sunt specifice IP.

Când componentele de la nivelul **Legătură** (Ethernet) primesc un pachet IP, acestea îl încapsulează, adăugând un antet care conține, printre altele, adresele nivelului **Legătură** (OSI 2) de 48 biți (MAC) ale sursei şi destinației, alături de secvența de control a cadrului (FCS).

Pentru a putea analiza formatul datagramei utilizate de **IP** trebuie specificate câteva elemente. Unitatea de transfer a protocolului de Internet (**IP PDU**) este denumită uneori datagramă, sau mai des, pachet. În cadrul pachetului, unitatea cu care se operează, atât pentru informațiile de control cât şi pentru celelalte protocoale sau servicii încapsulate în câmpul de date, este cuvântul. Acesta are dimensiunea de 32 biți (patru octeți). Lungimea unui pachet este întotdeauna un multiplu de 32 biți.

Lungimea maximă a antetului **IP** este de şase cuvinte (6 x 32 = 192 = 24 x 8 biți) adică 24 octeți, atunci când sunt incluse toate câmpurile opționale, iar cea minimă de cinci (5 x 32 = 160 = 20 x 8 biți) adică 20 octeți.

Pentru a înțelege semnificația tuturor câmpurilor trebuie să se precizeze că IP nu este dependent de sistemul hardware, dar orice variantă de IP trebuie să fie compatibilă, în jos, cu versiunile anterioare.

Formatul antetului **IP** este prezentat schematic în **figura 10.1**, iar semnificația câmpurilor este descrisă în detaliu în cele ce urmează.

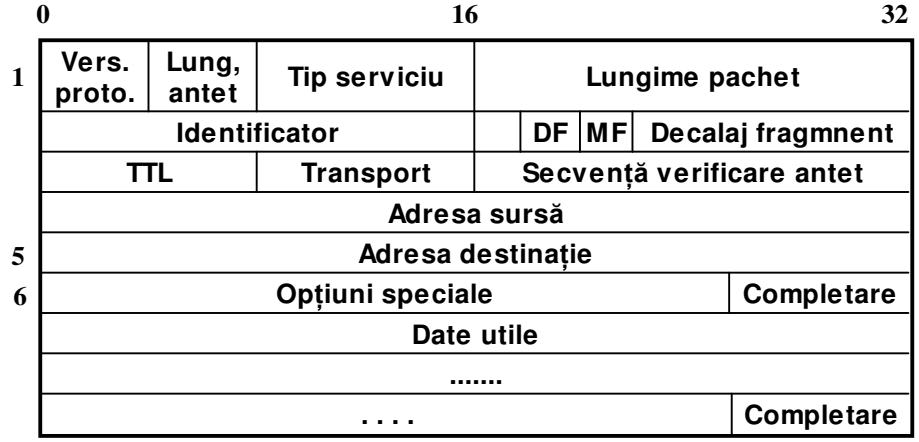

Fig. 10.1. Formatul antetului **IP**.

Numărul versiunii ***(Version Number)*** este un câmp cu dimensiunea de patru biți, care defineşte versiunea protocolului folosit de sistemul de operare. Acest număr este necesar pentru ca sistemul de operare, care primeşte pachetul, să ştie cum să decodifice restul antetului (care se poate schimba în funcție de versiune).

Versiunea cea mai des întâlnită la acest moment este 4, dar s-a trecut deja la noua generație (*IPng Next Generation*, sau IPv6). Pentru rețelele locale versiunea 4 este suficient de cuprinzătoare şi utilă. Sistemul de operare care recepționează pachetul de date trebuie mai întâi să verifice numărul versiunii pentru a şti de la început dacă poate să manevreze pachetul sau nu. În cazul că poate, va interpreta informațiile din antet, va prelucra datagrama, va extrage conținutul şi îl va trimite spre nivelul superior iar dacă nu, o va ignora complet.

Lungimea antetului *(Header Length)* este un câmp cu dimensiunea de patru biți care reprezintă lungimea totală a antetului exprimată în cuvinte (de câte 32 biți). Poate lua valoarea 5 (20 octeți), sau 6 (24 octeți) în cazul în care se folosesc şi informațiile opționale. Pentru a şti unde se termină antetul (şi implicit de unde încep datele), sistemul de operare trebuie să cunoască dimensiunea acestuia. Trebuie remarcat faptul că nu există elemente de delimitare între antet şi câmpul de date, lungimea antetului folosind-se pentru a calcula locul în care începe acesta.

Tipul serviciului *(Type of Service)* este un câmp cu dimensiunea de 8 biți (un octet) prin care se indică cum trebuie procesată corect datagrama. Semnificația biților este prezentată în **figura 10.2.** Primii trei biți indică prioritatea (importanța) datagramei, valoarea zecimală a câmpului variind între 0

(pachete cu prioritate normală) şi 7 (pachete de control), valoarea mai mare având prioritate mai mare. Teoretic, în cazul rutării unor pachete care aşteaptă într-o coadă la ieşirea dintr-o poartă de acces, cele care au prioritate mai mare (valoare mai mare a numărului scris în acest câmp) trebuie trimise primele. Practic , în cele mai multe cazuri de aplicare ale TCP/IP, se ignoră acest câmp. Există excepţii atunci când ruterele aplică reguli de asigurare a calităţii serviciilor (QOS - *Quality Of Services*).

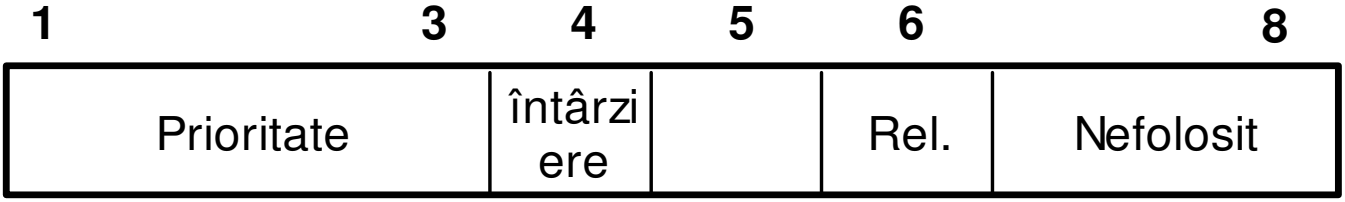

Fig. 10.2. Componentele câmpului de indicare a tipului serviciului.

Următorii trei biţi reprezintă: un bit de control al întârzierii, ratei de transfer şi al siguranţei, iar ultimii doi nu sunt folosiţi. Dacă primul bit este 1, se impun măsuri care să determine o întârziere mică, o rată mare de transfer şi o siguranţă ridicată a căii de transmitere a pachetului, iar bitul are valoarea 0 pentru situaţii normale. De fapt, la fel ca şi în cazul câmpului precedent, în situaţiile practice se ignoră rolul acestui bit, toate datagramele fiind tratate la fel. Sistemele de operare uzuale nu se complică să folosească şi să interpreteze aceşti biţi, deşi semnificaţia lor ar putea fi utilă în optimizarea unei reţele.

Lungimea pachetului *(Datagram Length, Packet Length*) este un câmp de 16 biţi care indică lungimea totală, în octeţi, a datagramei, câmp de date plus antet. Dimensiunea câmpului de date se poate obţine prin diferenţa dintre lungimea totală a datagramei şi cea a antetului. Dimensiunea maximă a pachetului **IP** poate să fie de 65.535 octeţi.

Identificator de bloc *(Identification).* Acest câmp conţine un număr, creat de sursă pentru a identifica, în mod unic, fragmentele aparţinând aceluiaşi bloc de date. Acest număr este folosit la recepţie, în procesul de reasamblare a blocurilor, pentru a nu amesteca între ele fragmente provenite de la blocuri diferite.

Atributele *(Flags)* reprezintă un câmp de 3 biţi, dintre care primul nu este folosit, iar următorii doi au rol în controlul procesului de fragmentare. Primul dintre aceştia este numit **DF** (*Don't Fragment* - "a nu se fragmenta"), iar următorul **MF** (*More Fragments* - "sosesc mai multe fragmente").

Dacă **DF** este 1, datagrama nu poate fi fragmentată în nici un caz. Dacă o componentă a nivelului **IP** nu poate trimite datagrama fără să o fragmenteze, atunci ea va fi abandonată şi acea componentă va trimite un mesaj de eroare spre dispozitivul sursă.

Dacă **MF** este 1, datagrama curentă va fi urmată de mai multe pachete care conţin fragmente (numite uneori subpachete) şi care trebuie reasamblate. Ultimul fragment trimis va avea bitul **MF** egal cu 0, ceea ce înseamnă că s-a terminat succesiunea curentă de fragmente, pentru ca terminalul destinaţie să nu mai aştepte alte fragmente. Deoarece ordinea în care sosesc fragmentele ar putea să nu fie aceeaşi cu cea de la expediere, atributul **MF** se foloseşte împreună cu câmpul următor **FO** (*Fragment Offset* - indicator al decalajului dintre fragmente).

Indicatorul decalajului dintre fragmente *(Fragment Offset).* Dacă bitul **MF** este 1 (se anunţă fragmentarea unei datagrame mari), indicatorul decalajului dintre fragmente arată poziţia unui fragment şi permite reasamblarea datagramei. Aceasta se face în succesiunea ei firească, indiferent de ordinea de sosire a fragmentelor. Decalajele sunt exprimate faţă de începutul mesajului. Câmpul are 13 biţi, deci decalajul va fi calculat în unităţi de un octet (8 biţi), corespunzător cu lungimea maximă a pachetului de 65.535 octeţi. Utilizat împreună cu identificatorul de bloc, componenta **IP** de pe terminalul destinaţie poate să reasambleze fragmentele indiferent de ordinea în care acestea ajung la aceasta.

Timpul de viaţă al unui pachet *(TTL - Time to Live)* este un câmp care indică timpul, în secunde, în care un pachet poate "călători" prin reţea înainte de a fi abandonat. Acesta este fixat de terminalul sursă în timpul asamblării datagramei. De obicei valoarea acestuia este de 15 sau 30 secunde. Standardele **TCP/IP** specifică faptul că valoarea acestui câmp trebuie diminuată, cu cel puţin o secundă, de către fiecare nod care procesează pachetul, chiar dacă timpul de procesare este mai mic de o secundă.

Când un pachet este recepționat de o poartă de acces, timpul de sosire este înregistrat, iar timpul cât acesta aşteaptă să fie procesat îi scurtează durata de viață. Dacă poarta de acces este supraîncărcată şi nu poate trimite pachetul într-un timp suficient de scurt, acesta va fi abandonat. Când valoarea câmpului **TTL** devine 0, pachetul trebuie abandonat de nodul curent, iar acesta trimite un mesaj **ICMP**, de eroare, spre terminalul sursă. Aceasta poate astfel să trimită din nou pachetul. Regula impusă de **TTL** are ca scop să împiedice circulația nedefinită a pachetelor în rețea în cazul apariției buclelor, sau a altor anomalii, de rutare.

Protocolul de transport *(Transport Protocol)* este un câmp prin care se identifică numeric protocolul de nivel superior. Acest număr este definit de InterNIC pentru fiecare dintre cele aproape 50 de protocoale de transport cunoscute. Pentru **IP** valoarea este 0, pentru **ICMP 1**, iar pentru **TCP**, 6. Pentru protocoalele implementate pe un terminal, aceste numele pot fi găsite în fişierul protocol. Conținutul unui astfel de fişier se poate vedea mai jos.

```
# This file contains the Internet protocols as defined by RFC 1060
# (Assigned Numbers).
#
# Format:
#
# <protocol name> <assigned number> [aliases...] [#<comment>]

ip      0    IP       # Internet protocol
icmp    1    ICMP     # Internet control message protocol
ggp     3    GGP      # Gateway-gateway protocol
tcp     6    TCP      # Transmission control protocol
egp     8    EGP      # Exterior gateway protocol
pup     12   PUP      # PARC universal packet protocol
udp     17   UDP      # User datagram protocol
hmp     20   HMP      # Host monitoring protocol
xns-idp 22   XNS-IDP  # Xerox NS IDP
rdp     27   RDP      # "reliable datagram" protocol
rvd     66   RVD      # MIT remote virtual disk
```

Pentru celelalte protocoale (multe dintre ele aproape nefolosite!) se pot consulta documentele RFC.

Suma de control a antetului *(Header Checksum)* este un câmp de 16 biți care reprezintă secvența de control a antetului, dar nu şi a datelor din pachet. Deoarece câmpul **TTL** este decrementat la fiecare trecere a pachetului printr-o poartă de acces, cu această ocazie suma de control trebuie recalculată şi înscrisă în antet. Algoritmul de calcul, a sumei de control, foloseşte complementul față de unu a celor 16 biți din fiecare cuvânt. Acesta este un algoritm rapid şi destul de eficient, dar nu reuşeşte să recupereze datele în cazul pierderii unui întreg cuvânt de 16 biți. Datorită faptului că secvența de control folosită pentru verificarea datelor la nivelul Transport (OSI 4), atât de **TCP** cât şi de **UDP**, este mai complexă, integritatea datelor conținute de pachet nu este afectată.

Adresa sursă şi adresa destinație *(Sending Address / Destination Address).* Aceste câmpuri de 32 biți conțin adresa IP (logică) maşinii sursă şi a maşinii destinație, Aceste câmpuri se stabilesc la crearea pachetului şi nu sunt modificate pe parcurs, decât în cazul adreselor din rețele private, care se conectează în exterior prin NAT (Network Address Translation - translatarea adreselor în rețea, proces de schimbare a adresei sursă la ieşirea dintr-o poartă de acces care conectează o rețea privată la una publică).

Opțiuni speciale, este un câmp opțional, compus din câteva coduri cu lungime variabilă. Dacă se folosesc mai multe opțiuni, acestea sunt aşezate consecutiv în antetul **IP**. Toate opțiunile sunt definite de un octet, împărțit în trei câmpuri: 1 bit atribut de copiere, 2 biți definesc clasa opțiunii şi 5 biți reprezintă numărul opțiunii. Atributul de copiere este folosit pentru a specifica cum trebuie utilizată opțiunea, pentru cazul în care este necesară fragmentarea pachetului, la trecerea printr-o poartă de acces. Când acest bit este 0 opțiunea trebuie copiată doar pe următoarea datagramă (nu şi pe următoarele), iar dacă este 1 opțiunea se copiază pe toate datagramele (fragmentele). Valoarea zecimală a biților care definesc clasa şi tipul opțiunii este prezentată în **tabelul 10.1.** În acest moment se folosesc doar două clase de opțiuni, din cele patru

posibile de-a fi definite cu doi biţi. Dacă valoarea este 0 opţiunea se aplică pachetelor de date şi de control a reţelei. Valoarea 2 se foloseşte pentru pachete folosite la depanare, sau pentru administrare. Valorile 1 şi 2 nu se folosesc.

Tabelul 10.1. Valoarea câmpurilor pentru opţiunile speciale

Clasa opţiunii	Numărul opţiunii	Descrierea opţiunii
0	0	Marks the end of the options list
0	1	No option (used for padding)
0	2	Security options (military purposes only)
0	3	Loose source routing
0	7	Activates routing record (adds fields)
0	9	Strict source routing
2	4	Timestamping active (adds fields)

Sunt interesante opţiunile care permit înregistrarea rutei şi a marcajelor de timp. Acestea sunt folosite pentru a înscrie, în corpul datagramei, "o foaie de parcurs" a datagramei de-a lungul căii prin reţea, informaţii necesare pentru diagnosticare şi depanare. Timpul se exprimă în milisecunde scurse de la miezul nopţii (GMT - Ora Universală Greenwich), ceea ce presupune necesitatea unei bune sincronizări a ceasului fiecărui sistem. Cum acest lucru nu este foarte riguros, chiar în cazul utilizării serverelor de timp (în nici un caz nu se pot sincroniza la milisecundă ceasurile porţilor de transfer), aceste informaţii sunt îndoielnice.

Există două tipuri de rute, indicate în câmpul de opţiuni speciale, *loose* şi *strict*. Ruta liberă *(Loose routing)* reprezintă o listă de adrese pe care pachetul trebuie să le parcurgă, dar este permisă orice rută care trece prin aceste adrese. Ruta strictă *(Strict routing)* nu permite nici o deviere de la lista de adrese. Dacă ruta nu poate fi respectată, pachetul este abandonat. Ruta strictă este folosită la testarea rutelor, dar nu şi la transmiterea pachetelor normale, existând riscul ca acestea să se piardă, deoarece nu pot respecta ruta. În cazul reţelelor complexe, rutele sunt determinate dinamic şi se schimbă de la un moment la altul, respectarea unei rute alese „a priori", fiind aproape imposibilă.

Completarea cu zerouri (*Padding*). Conţinutul acestui câmp este variabil în funcţie de opţiunile selectate şi se foloseşte pentru a completa cu zerouri antetul pachetului, sau câmpul **Date**, pentru ca acestea să aibă un număr întreg de cuvinte.

Formarea şi transferul unui pachet în reţea.

Pentru a înţelege mai uşor modul în care nivelele **TCP/IP** (OSI 4 şi 3) împachetează şi trimit datele de la un terminal la altul, trebuie să analizăm un exemplu simplu. Când o aplicaţie trimite o datagramă spre reţea, sunt parcurse câteva etape simple.

În prima etapă, se construieşte pachetul **IP,** în conformitate cu lungimea prescrisă de opţiunile de implementare, ale suitei **TCP/IP,** pe sistemul respectiv. Se calculează suma de control pentru câmpul de date şi apoi se construieşte antetul **IP**. După aceea, se determină următorul pas *(next hop)* din ruta spre terminalul destinaţie, direct spre acel terminal, dacă el se află în acelaşi domeniu de coliziuni, sau spre prima poartă de acces, în caz contrar. În cazul unei rutări speciale, se vor folosi opţiunile suplimentare de rutare din antet, iar dacă nu există restricţii de rută, aceste opţiuni vor fi eliminate. În final, datagrama este predată nivelului inferior, pentru încapsulare, în vederea transmiterii în reţea.

Când o datagramă parcurge reţeaua, fiecare poartă de acces execută, asupra ei, o serie de teste. După ce nivelul inferior (*Network Access* - după modelul **TCP/IP**, nivelul Legătură OSI 2) decapsulează datagrama (cadrul) şi trimite conţinutul acesteia nivelului superior (**IP**), care verifică integritatea datagramei. Dacă suma de control nu se potriveşte, datagrama (pachetul) este abandonat şi poarta de acces trimite spre sursă un mesaj de eroare. Dacă suma se potriveşte, câmpul **TTL** este decrementat şi apoi comparat cu zero. În cazul în care durata de viaţă a expirat, datagrama se abandonează şi se transmite spre sursă un mesaj de eroare. După determinarea următorului pas *(next hop)*, în funcţie de adresa destinaţie sau de informaţiile de rutare din câmpul opţiunilor speciale, se reconstruieşte antetul **IP** cu o nouă valoare a timpului de viaţă (**TTL**) şi a sumei de control.

Dacă este necesară fragmentarea, datorită creşterii lungimii datagramei sau a unei limitări a acesteia din partea sistemului de operare sau al nivelelor inferioare, datagrama va fi împărţită şi bucăţile vor fi asamblate (încapsulate) fiecare cu câte un antet, care poartă informaţiile de control necesare. Dacă este ne-

voie de rută specială, sau de marcaje de timp, acestea vor fi adăugate. Reorganizată în acest fel, datagrama va fi trimisă înapoi nivelului inferior **Legătură** (OSI 2) pentru a-şi urma, mai departe, calea spre destinaţie.

Când datagrama ajunge, în sfârşit, la destinaţie, sistemul de operare îi verifică, cu ajutorul sumei de control, integritatea şi dacă mesajul a fost fragmentat pe drum. Dacă există mai multe fragmente, sistemul le aşteaptă pe toate un anumit interval de timp. Dacă acestea nu sosesc, sau sistemul nu le poate reasambla în acest interval, fragmentele sunt abandonate şi se trimite spre sursă un mesaj de eroare. Dacă fragmentele pot fi reasamblate, antetul **IP** este eliminat, mesajul refăcut şi conţinutul său trimis nivelului superior. Dacă este nevoie de mesaj de răspuns, acesta va fi generat şi trimis înapoi spre sursă.

Când într-o datagramă se adaugă informaţii speciale, de rutare sau marcaje de timp, lungimea acesteia poate să crească, ceea ce înseamnă că pe drum pot avea loc procesele de fragmentare. Faptul că **IP** poate să opereze automat cu această problemă, având mecanismele necesare de decizie, este încă un argument important pentru folosirea acestui protocol în reţelele de mare complexitate.

10.2. Internet Control Message Protocol (ICMP).

Aşa cum s-a putut vedea la începutul acestui capitol, în timpul transferului unei datagrame prin reţea, între terminalul sursă şi cel destinaţie, pot să apară probleme de rutare, întârziere, sau de fragmentare. Un pachet trimis poate să nu găsească destinaţia, să întârzie mai mult decât îi permite **TTL**, sau fragmente dintr-un mesaj se pot pierde pe drum. Acest protocol a fost dezvoltat, ca parte a suitei **TCP/IP**, pentru a se putea verifica starea reţelei, de fapt a căii dintre sursă şi destinaţie.

ICMP (*Internet Control Message Protocol* - protocol de transmitere a mesajelor de control) este un sistem care permite raportarea erorilor. Ca parte a suitei de protocoale de **Reţea** (OSI 3), specifice nivelului Internet (**IP**), el trebuie inclus în oricare implementarea **IP**. Există unele excepţii legate de securitate, deoarece permite "conversaţii de control" între două terminale, sau noduri din reţea. Mesajele generate de **ICMP** sunt tratate, de-a lungul reţelei, ca orice pachet **IP**, dar sunt interpretate corespunzător de aplicaţiile aflate la nivelul superior. Aşa cum se va vedea la adresarea porturilor în **TCP** şi **UDP**, **ICMP** foloseşte un port dedicat [1].

În majoritatea situaţiilor, mesajele de eroare trimise de **ICMP** sunt "întoarse" terminalului sursă, deoarece în antetul pachetului sursă sunt trecute doar adresele sursei şi destinaţiei. Deoarece mesajul de eroare nu reprezintă nimic pentru terminalul destinaţie, sursa este singurul recipient valabil la care trebuie să ajungă mesajele de eroare. În funcţie de acestea, terminalul sursă poate determina tipul defectului şi poate lua măsurile corespunzătoare.

Mesajul **ICMP** suferă o dublă încapsulare, acestuia i se adaugă mai întâi un antet propriu **ICMP** şi apoi antetul standard al pachetului **IP**, aşa cum se poate vedea în **figura 10.3.** Formatul antetului **ICMP** este diferit în funcţie de tipul mesajului, toate începând cu aceleaşi trei câmpuri: tipul mesajului, codul mesajului şi suma de control. În **figura 10.4** se poate vedea formatul unui mesaj **ICMP**.

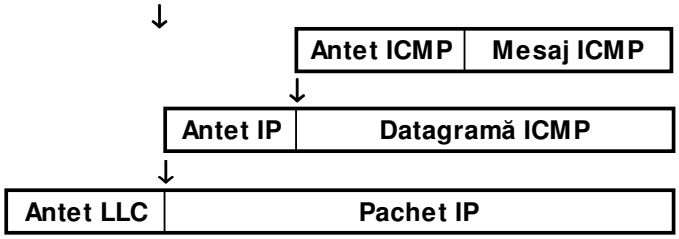

Fig. 10.3. Dubla încapsulare a unui mesaj **ICMP**.

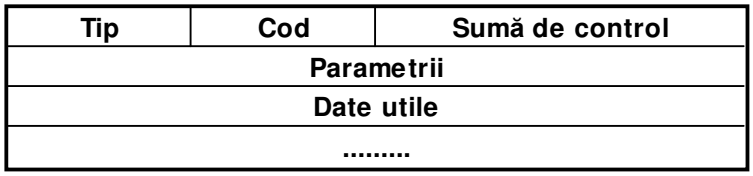

Fig. 10.4. Formatul unui mesaj **ICMP**.

În general, orice mesaj **ICMP** care raportează o problemă de livrare, include antetul şi primii 64 biți de date din pachetul la care a apărut problema. Există două motive pentru care se face acest lucru. Primul, se permite sursei să compare fragmentul cu pachetul original, iar al doilea, pentru a afla din antet tipul protocolului implicat în problemă.

Tipul mesajului este un câmp de 8 biți şi poate avea valorile prezentat în **tabelul 10.2**.

Tabelul 10.2. Valorile posibile ale câmpului tipul mesajului **ICMP**.

Valoare	Semnificație	Descriere	Observații
0	Echo Reply	Răspuns cu ecou	
3	Destination Not Reachable	Destinația nu este accesibilă	
4	Source Quench	Sursă	
5	Redirection Required	Cerere de redirectare	
8	Echo Request	Cerere de ecou	
11	Time to Live Exceeded	TTL depăşit	
12	Parameter Problem	Mesaj de eroare	
13	Timestamp Request	Se cer marcaje de timp	
14	Timestamp Reply	Răspuns cu marcaje de timp	
15	Information Request	Se cer informații	nu mai este folosit
16	Information Reply	Răspuns cu informații	nu mai este folosit
17	Address Mask Request	Se cere mască de adrese	
18	Address Mask Reply	Răspuns cu mască de adrese	

Câmpul **cod mesaj** detaliază informațiile conținute în câmpul tip mesaj, iar suma de control se calculează la fel ca la antetul **IP**.

Formatul mesajului **ICMP** este diferit pentru fiecare tip de mesaj în parte, **figura 10.5** prezentând formatul fiecărui tip de antet. Unele mesaje se interpretează direct, aşa cum este de exemplu **„destinație imposibil de atins”** *(Destination Unreachable)*, iar altele trebuie să influențeze decizii de manipulare a pachetelor. Un astfel de exemplu este acesta: **„datagrama trebuie neapărat fragmentată”** *(Datagram must be fragmente),* iar dacă acest lucru nu este posibil, datorită atributului **DF** activ, mesajul trebuie interpretat de aplicația de la nivelul superior şi luată măsura potrivită.

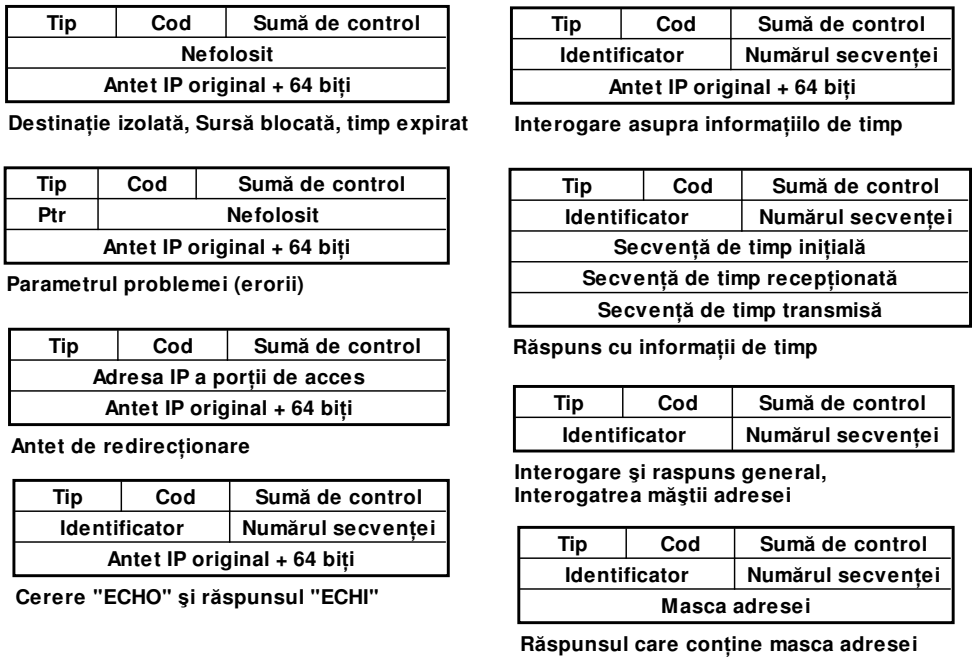

Fig. 10.5. Formatul antetului din mesajul ICMP.

Mesajul ICMP de **„temperare a sursei”** (*Source Quench)* este folosit pentru a controla rata de transmitere a datagramei, deşi este o metodă foarte rudimentară de control al fluxului. Când un

terminal primeşte un astfel de mesaj, ar trebui să-şi reducă rata de transmitere a datelor, până la o valoare la care acest mesaj încetează.

De obicei, mesajul este generat de către o poartă de acces (ruter), sau un terminal, care are tamponul de recepţie a datelor plin, procesează lent pachetele, sau din alte motive. Dacă tamponul este plin, terminalul emite acest tip de mesaje, la fiecare abandonare a unei datagrame. Pentru a coborî rata de transmisie a pachetelor, unele variante de implementare a sistemului emit aceste mesaje când tamponul depăşeşte o anumită fracţiune din capacitate, ceea ce poate permite golirea tamponului înaintea primirii unei noi datagrame.

Mesajele de redirecţionare *(Redirection Required)* se trimit unei porţi de acces (ruter), dintr-o cale, atunci când este găsită o rută mai bună. De exemplu, dacă o poartă tocmai a primit un pachet, dar analizând tabelele de rutare găseşte că există o rută mai bună, trimite spre sursă un mesaj de redirectare, care conţine adresele **IP** ale acestei rute. În antetul acestui mesaj se trece, în câmpul cod al antetului o valoare, indicându-se condiţiile în care se aplică modificarea rutei.

0 - înseamnă că toate datagramele trimise spre reţeaua destinaţie trebuie redirecţionate;

1 - se redirecţionează doar cele provenite de la o anumită maşină;

2 - trebuie redirecţionate doar datagramele care sunt destinate unei reţele, care foloseşte aceleaşi tipuri de servicii (indicate în antetul IP);

3 - redirecţionarea are loc doar pentru acelaşi terminal şi pentru acelaşi tip de serviciu.

Mesajul „**Parametru eronat"** *(Parameter Problem)* este folosit oricând apare o eroare, semantică sau sintactică, în antetul **IP**. Aceasta se poate întâmpla atunci când se folosesc opţiuni cu argumente greşite. Când se trimite un astfel de mesaj, înapoi spre sursă, câmpul parametri, din componenţa mesajului de eroare, conţine un indicator *(pointer)* al octetului, din antetul IP, care a cauzat anomalia.

Cererile de ecou *(echo request)*, sau **răspunsurile la mesaje** *(reply messages)* sunt des folosite în scopuri de depanare. Când se trimite o cerere şi destinaţia, sau calea spre aceasta, este inactivă, o poartă de acces de pe acea cale trimite spre sursă un răspuns corespunzător. Această pereche **cerere/răspuns** este utilă pentru identificarea problemei, a porţii de acces defecte sau a mediului de propagare întrerupt.

Simpla utilizare a unui mesaj **ICMP** este de fapt o verificare a reţelei, deoarece fiecare dispozitiv, aflat de-a lungul căii, trebuie să interpreteze corect antetul şi să transmită mai departe mesajul.

Orice defecţiune de-a lungul căii se poate regăsi într-o formă sau alta în mesajele de eroare. Exemplul cel mai cunoscut de folosire a unui sistem de perechi **cerere/răspuns** este utilitarul **ping**. Acesta trimite o serie de cereri, sub forma unor pachete de date de o dimensiune dată, iar apoi aşteaptă întoarcerea răspunsului. Prin măsurarea timpului scurs de la transmitere cu cel de primire a răspunsului şi prin numărarea pachetelor pierdute, se poate determina calitatea legăturii. Prin combinarea acestor informaţii cu cele oferite prin utilizarea opţiunii **„rută strictă"** *(Strict Routing)* se pot identifica tronsoanele de reţea afectate de congestie. În mod asemănător se pot trimite cereri unui grup de adrese dintr-o reţea. În funcţie de mesajele de răspuns trimise de acestea, se pot compara modurile de reacţie ale maşinilor.

Utilitarul **traceroute** foloseşte mesaje **ICMP** pentru a determina starea şi performanţele conexiunii, pentru fiecare tronson de reţea, dintre sursă şi destinaţie.

10.3. Adresarea IP.

Protocolul Internet (IP) oferă, pentru terminalele care utilizează acest protocol, o soluţie de **identificare unică** şi **ierarhică,** în reţea. Unicitatea se referă la cazul apartenenţei terminalului la o reţea publică, sau în cadrul unei reţele private. O **reţea publică** este caracterizată prin faptul că terminalele sunt accesibile reciproc, cel puţin teoretic (fără să se ţină seama de restricţii administrative de acces), din orice parte a ei. O **reţea privată** este o reţea izolată, din punct de vedere fizic de alte reţele, sau este conectată prin procedee de translatare, sau ascundere, a adreselor. Pachetele care se transmit în interiorul unei reţele private nu sunt rutate spre reţeaua

publică. În cazul reţelelor private, se poate folosi aceeaşi adresă, pentru mai multe terminale, dacă acestea nu aparţin aceleiaşi reţele [1, 17].

Formatul adresei IP.

Deoarece scopul acestui paragraf este de a face o analiză principială a unui mod de adresare ierarhică, se va studia doar cazul versiunii 4 a protocolului Internet (IPv4). Această versiune utilizează pentru adresare un cuvânt de 32 biţi, în care sunt identificate atât terminalul, cât şi reţeaua din care face parte acesta. Primii biţi, mai semnificativi, reprezintă adresa reţelei, iar următorii (până la 32) identifică terminalul, în cadrul reţelei, aşa cum se poate vedea în **figura 10.6**. Apar astfel două câmpuri: adresa de reţea *(Network)*, pentru biţii egali cu 1 din masca de reţea şi identificatorul terminalului în cadrul reţelei *(Host)*, pentru biţii din mască egali cu 0.

Adresa unui terminal este unic determinată dacă se cunoaşte adresa de reţea şi identificatorul de terminal. Aceste două componente se folosesc împreună, fiind nedespărţite. Deşi semnificaţie adresei se interpretează în forma sa binară, pentru o administrare mai uşoară pentru operatori, se utilizează formatul „zecimal cu punct". Prin aceasta, se împarte numărul binar, compus din 32 biţi, în 4 octeţi reprezentaţi în format zecimal şi despărţiţi între ei printr-un punct, **a.b.c.d** unde **a** este primul octet, **b** al doilea, **c** al treilea, iar **d** al patrulea.

Notă: Trebuie specificat faptul că unii biţi dintr-un octet pot aparţine câmpului „Reţea", iar ceilalţi câmpului „Terminal". Valoarea zecimală a octetului nu ţine seama de acest lucru, ci doar de valoarea biţilor. Ca urmare a acestui lucru, din valoarea zecimală a octetului nu se poate determina frontiera care separă cele două câmpuri.

Masca de reţea.

Deoarece, cele două entităţi, câmpul „Reţea" şi câmpul „Terminal", nu sunt reprezentate distinct, separarea lor se face cu ajutorul măştii de reţea. Masca de reţea *(netmask)* este un filtru logic care permite, prin operaţii binare de conjuncţie logică (funcţia ŞI), să extragă din adresa unui terminal, adresa reţelei din care acesta face parte. Deoarece adresa de reţea este aceeaşi, pentru terminale aparţinând aceleiaşi reţele, acestea din urmă o folosesc pentru a determina dacă un terminal destinaţie se află în reţeaua lor, sau nu. Dacă după aplicarea măştii de reţea, pe adresa sursă şi pe cea destinaţie, se obţin rezultate diferite, înseamnă că terminalul destinaţie se află în altă parte, pachetul de date fiind trimis porţii de acces (ruter), spre a fi trimis mai departe în reţea.

Adresa de *broadcast*.

Adresa de broadcast se foloseşte pentru transmiterea de mesaje întregii comunităţi de terminale dintr-o reţea. Pachetele trimise cu adresă destinaţie de *broadcast* vor fi acceptate de toate terminalele, alături de mesajele care sosesc cu adresă destinaţie dedicată *(unicast)*. Adresa de broadcast se obţine din adresa de reţea, la care toţi biţii din câmpul terminal sunt egali cu 1.

Exemplu de adresare IP.

Pentru exemplificare, se consideră o reţea privată, formată din 35 terminale ***($2^5 = 32 < 35 < 64 = 2^6$)***, a cărei adresă de reţea este 192.168.120.0 şi are masca 255.255.255.192. Dacă se aplică, în formatul binar, masca de reţea şi se completează cu 1 câmpul din adresă unde masca are biţii egali cu 0, se obţine adresa de broadcast, aşa cum se poate vedea în exemplul prezentat în **figura 10.6**.

Octet	a	.	b	.	c	.	d
Adresa IP	192	.	168	.	120	.	28
Format (binar)	11000000	.	10101000	.	01111000	.	00 011100
Masca (binar)	11111111	.	11111111	.	11111111	.	11 000000
Câmpuri			Reţea				Terminale
Masca	255	.	255	.	255	.	192
Reţea (binar)	11000000	.	10101000	.	01111000	.	00 000000
Adresa Reţea	192	.	168	.	120	.	0
Broadcast (bin.)	11000000	.	10101000	.	01111000	.	00 111111

Broadcast	192	.	168	.	120	.	63	

Fig. 10.6 Formatul adresei IPv4: adresa reţelei, masca şi adresa de *broadcast*.

Adresele, cuprinse între adresa de reţea 192.168.120.0 şi cea de *broadcast* 192.168.120.63, se pot aloca terminalelor din reţea, inclusiv porţii de acces (ruter), dacă este cazul. Numărul maxim de adrese, incluzând adresa reţelei, a terminalelor şi adresa de *broadcast*, depinde de numărul de biţi din câmpul „Terminal", în acest caz 6, fiind egal cu 2^6. Numărul de adrese care pot fi alocate terminalelor este $2^6 - 2 = 64 - 2 = 62$. Generalizând problema, Dacă notăm cu R numărul de biţi din câmpul „Reţea" şi cu T numărul de biţi din câmpul „Terminal" $(R + T = 32)$, numărul maxim de terminale într-o reţea $N_T = 2^T - 2$.

Alocarea adreselor.

Adresarea într-o reţea publică are ca scop identificarea unică a terminalelor şi de aceea trebuie ţinută o evidenţă strictă a acestora. De această problemă se ocupă **IANA** *(Internet Assigned Numbers Authority),* o autoritate administrativă înfiinţată pentru a controla alocarea diferitelor numere de identificare, în Internet [1].

Clase de adrese.

În momentul conceperii protocolului Internet (IP) s-a considerat că în lume se vor dezvolta câteva reţele mari şi s-a alocat, pentru identificarea lor, primul octet din adresă. Această „viziune" s-a dovedit mai târziu că a fost o mare eroare, cele 256 de combinaţii posibile fiind insuficiente pentru cererea, în continuă creştere, a pieţei. Mai târziu, prin documentul RFC 791 emis în 1981, s-au definit 3 clase de dimensiuni de reţea, A, B, C. La acestea s-au adăugat încă două C şi D, pentru utilizări speciale, aşa cum sunt specificate în **tabelul 10.3**.

Tabelul 10.3.

Clasa	Primul octet		Octeţi alocaţi		Număr maxim		Prefix
	Binar	Zecimal	Reţea	Term.	Reţele	Term.	
A	0xxxxxxx	1-126	a.	b.c.d	126 (2^7-2)	16M (2^{24}-2)	/8
B	10xxxxxx	128-191	a.b.	c.d	16k (2^{14})	65k (2^{16}-2)	/16
C	110xxxxx	192-223	a.b.c.	d	2M (2^{21}-1)	254 (2^8-2)	/24
D	1110xxxx	224-239	utilizată pentru transmisii multicast				/4
E	1111xxxx	240-254	rezervată pentru utilizări viitoare				/4
loop	01111111	127	uz intern, 127.0.0.1 – adresa locală				

Între adresele specificate anterior există un set de adrese care pot fi utilizate pentru reţele private. Pachetele care sunt transmise de terminale sursă, având aceste adrese, nu sunt rutate spre reţelele publice şi de aceea pot fi folosite în mai multe reţele, dacă acestea nu sunt conectate între ele. Clasele de adrese private sunt prezentate, alături de caracteristicile lor, în **tabelul 10.4.**

Tabelul 10.4.

Clasa	Dim. maximă a reţelei – prefix	Adresă de început	Adresă de sfârşit	Masca de reţea a clasei
A	24 biţi	10.0.0.0	10.255.255.255	255.0.0.0
B	20 biţi	172.16.0.0	172.31.255.255	255.240.0.0
C	16 biţi	192.168.0.0	192.168.255.255	255.255.0.0
Notă:	Aceste adrese pot fi împărţite în subreţele, cu dimensiunea necesară.			

Există situaţii în care este nevoie de anumite adrese speciale, adresări colective (multicast), interfeţe locale virtuale (loop), sau pentru experimente sau teste. Câteva dintre cele mai importante clase de adrese de acest tip sunt prezentate în **tabelul 10.5**.

Tabelul 10.5.

Clasa	Domeniul de utilizare	Adresă/prefix	Adresă de sfârşit	Masca de reţea a clasei
-	Ruta implicită*	0.0.0.0/0	255.255.255.255	0.0.0.0
A	Adresa nulă	0.0.0.0/8	0.255.255.255	255.0.0.0
A	Adrese locale	127.0.0.0/8	127.255.255.255	255.0.0.0
B	Adrese automate	169.254.0.0/16	169.254.255.255	255.255.0.0
C	Experimente	192.0.2.0/24	192.0.2.255	255.255.255.0
D	Multicast	224.0.0.0/4	239.0.0.0	240.0.0.0
E	Rezervate	240.0.0.0/4	255.255.255.255	240.0.0.0
Notă:	*Aceasta adresă are doar un rol simbolic, reprezentând „toate adresele”			

Împărţirea în subreţele.

După cum se poate vedea în tabelele de mai sus, dimensiunea claselor predefinite nu este întotdeauna potrivită pentru reţelele reale. De aceea, este nevoie ca aceste clase să fie împărţite în domenii mai mici, sau dimpotrivă, din două clase cu dimensiune mai mică să se creeze un domeniu de reţea, mai mare. Acest lucru se poate face prin operaţia care se numeşte împărţire în **subreţele** *(subnetworking)*. Împărţirea în subreţele are, de cele mai multe ori, un rol administrativ prin care se poate controla şi gestiona mai uşor o reţea. Acest lucru este necesar dacă trebuie controlat accesul la anumite resurse, sau anumite „tratamente” diferite făcute asupra unor grupuri de terminale [2].

Trebuie remarcat faptul că pentru a împărţi un domeniu de adrese, în două părţi egale, trebuie să se adauge în mască, spre dreapta, un bit egal cu 1. Aceste operaţii de „deplasare” a biţilor se numesc „împrumuturi” şi se pot face în ambele sensuri. Când se împrumută biţi din câmpul „Terminal” care se adaugă la câmpul „Reţea”, are loc o operaţie de **împărţire**, dimensiunea reţelei se înjumătăţeşte cu fiecare bit deplasat spre dreapta. În cazul când se împrumută biţi din câmpul „Reţea”, care se adaugă la câmpul „Terminal”, are loc o operaţie de **agregare**, dimensiunea reţelei se dublează, cu fiecare bit deplasat spre stânga.

TESTE DE AUTOEVALUARE

1. Nivelul (nivelele) OSI la care operează protocolul Internet sunt:

   a) 2-legătură, 3-rețea şi 4-transport;
   b) 3-rețea şi 4-transport;
   c) 4-transport;
   d) 3-rețea;
   e) 3-rețea şi 5-sesiune.

2. Lungimea minimă a antetului IP este de:
   a) 32 biți;
   b) 4 octeți;
   c) 5 x 32 biți;
   d) 20 octeți;
   e) 128 biți

3. Într-o rețea cu adresa 192.168.22.64 şi masca 255.255.255.192 are adresa de broadcast:
   a) 192.168.22.255;
   b) 192.168.22.128;
   c) 192.168.22.127;
   d) 192.168.22.192;
   e) 192.168.22.63;

I. Alegeți răspunsul corect:

1. IP are mecanisme specifice de control al fluxului de date şi garantează livrarea pachetelor la destinație.

   □ adevărat □ fals

2. Câmpul „lungime pachet”, din componeța antetului pachetului IP, are lungimea de 65535 octeți.

   □ adevărat □ fals

3. ICMP este un protocol de transmitere a mesajelor de control, oferind un sistem care permite raportarea erorilor. Pachetul ICMP se încapsulează în pachete IP şi nu sunt folosite pentru tranferul datelor utile.

   □ adevărat □ fals

RĂSPUNSURI

I. Varianta de răspuns corectă:

1. d
2. c, d
3. c

II. Alegeți răspunsul corect:

1. fals
2. fals
3. adevărat

Bibliografie

1. **Parker, T**., *Teach Yourself TCP/IP in 14 Days - Second Edition*, Sams Publishing, 1996
2. **Hart, R**., „*IP Sub-Networking Mini-Howto*”, v1.1, Distribuție Linux, 2001