

UNIVERSITATE DIN BUCUREŞTI
FACULTATEA : *MATEMATICĂ ŞI INFORMATICĂ*
DEPARTAMENTUL : *INFORMATICĂ*



FMI – CTI REȚELE



FMI – CTI REȚELE

***FUNDAMENTELE
REȚELELOR DE
CALCULATOARE***

Contents

Introducere în Lumea Rețelelor.....	10
CAPITOLUL 1. EXPLORAREA REȚELELOR	13
<i>Introducere.....</i>	13
1.2 <i>Rețelistica în prezent</i>	13
1.3 <i>Schimbarea modului în care învățăm.....</i>	15
1.4 <i>Schimbarea modului în care comunicăm.....</i>	16
1.5 <i>Schimbarea modului în care muncim.....</i>	18
1.6 <i>Schimbarea modului în care ne jucăm</i>	18
1.7 <i>Furnizarea de resurse în Rețele</i>	19
1.8 <i>LANs, WANs, și Internetul - Componentele Rețelelor.....</i>	22
1.9 <i>LANs and WANs</i>	27
1.10 <i>Internetul</i>	29
1.10.1 <i>Conectarea la Internet</i>	30
1.11 <i>Rețea ca o Platformă - Rețele convergente</i>	33
1.12 <i>Rețea de încredere.....</i>	34
1.13 <i>Schimbarea mediului de rețea - Tendințe de rețea.....</i>	42
1.14 <i>Conceptul – integrează propriul dispozitiv (Bring Your Own Device -BYOD)</i>	43
1.15 <i>Comunicație Video</i>	43
1.16 <i>Cloud Computing</i>	45
1.17 <i>Tehnologii pentru Rețele de casă</i>	47
1.18 <i>Wireless Internet Service Provider (WISP)</i>	48
1.19 <i>Securitatea rețelei</i>	49
1.20 <i>Arhitecturi de Rețea</i>	50
1.21 <i>Concluzii Capitolul 1</i>	51
CAPITOLUL 2. CONFIGURAREA IOS PEN DE REȚEA	53
<i>Introducere</i>	53
2.1 <i>IOS Bootcamp</i>	54
2.2 <i>Accesarea unui dispozitiv Cisco IOS.....</i>	57
2.3 <i>Navigarea prin IOS</i>	59
2.3.1 <i>Structura de bază a comenzi IOS</i>	63
2.4 <i>Noțiuni de bază – Nume de gazdă</i>	70
2.5 <i>Exemplu pentru Setarea Numelui.....</i>	71
2.5.1 <i>ConFig.rea Numelor Gazdelor prin intermediul IOS</i>	71
2.5.2 <i>Limitarea Accesului la ConFig.țiile Dispozitivelor</i>	72
2.5.3 <i>Salvarea ConFig.țiilor</i>	75
2.5.4 <i>ConFig.țiile de rezervă cu capturarea textului</i>	77
2.6 <i>Schemele Adreselor . Porturi și Adrese</i>	78

2.6.1 Adresarea Dispozitivelor.....	79
2.6.2 Verificarea Connectivității.....	82
2.7 Concluzii Capitolul 2	84
CAPITOLUL 3. PROTOCOALE DE REȚEA	86
Introducere.....	86
3.1 Conceptele fundamentale referitoare la aceste reguli și funcții	86
3.2 Reguli de comunicare.....	87
3.2.1 Regulile	87
3.2.2 Stabilirea regulilor.....	88
3.2.3 Codarea mesajului	89
3.2.4 Formatarea și încapsularea mesajului	90
3.2.5 Dimensiunea Mesajului	91
3.2.6 Opțiuni de livrare a mesajului.....	94
3.3.1 Protocole	95
3.3.2 Suite de protocole.....	98
3.4 Organizațiile de emitere a standardelor	103
3.5 Modele de referință	106
3.6 Deplasarea datelor prin intermediul Rețelelor	109
3.6.1 Încapsularea datelor	109
3.7 Adresa de rețea	112
3.8 Adresa de legătură de date.....	113
3.9 Accesarea resurselor de la distanță	115
3.10 Concluzii Capitolul 3	117
Introducere.....	118
4.1 Protocole de nivel fizic	118
4.2 Scopul nivelului fizic	119
4.3 Principiile fundamentale ale Nivelului 1.....	122
4.3.1 Componete fizice	122
4.3.2 Codificarea.....	122
4.3.3 Semnalizarea	123
4.4 Medii de comunicații de rețea.....	125
4.4.1 Cabluri de cupru	125
4.5 Protocoalele de la Nivelul Legătura de Date.....	140
4.5.1 Scopul nivelului legătură de date	140
4.5.2 Structura frameului de nivel 2	142
4.5.3 Standarde de Nivel 2	143
4.5.4 Controlul Accessului la Mediu - Media Access Control	144

<i>4.6 Ethernet</i>	154
<i>4.7 Point-to-Point Protocol</i>	155
<i>4.8 802.11 Wireless</i>	156
<i>4.9 Concluzii Capitolul 4</i>	157
CAPITOLUL 5. TEHNOLOGIA ETHERNET	159
<i>Introducere</i>	159
<i>5.1 Protocolul Ethernet</i>	159
<i>5.1.1 Ethernet Operation</i>	159
<i>5.1.2 Încapsularea datelor</i>	160
<i>5.1.3 Structura adresei MAC</i>	162
<i>5.2 Atributele frame-ului Ethernet</i>	164
<i>5.3 Ethernet MAC</i>	166
<i>5.4 ADRESELE DE TIP : MAC și IP</i>	170
<i>5.5 Address Resolution Protocol – ARP</i>	172
<i>5.5.1 Rezolvarea adreselor IPv4 în adresarea MAC</i>	172
<i>5.5.2 Menținerea tabelei ARP</i>	173
<i>5.5.3 Crearea frame-ului</i>	174
<i>5.5.4 ARP</i>	175
<i>5.5.5 LAN bazate pe Switchuri</i>	179
<i>5.6 Fix sau Modular</i>	185
<i>5.7 Switchuri de Layer 3</i>	187
<i>5.8 Concluzii Capitolul 5</i>	190
CAPITOLUL 6. NIVELUL REȚEA	192
<i>Introducere</i>	192
<i>6.1 Network Layer Protocols</i>	192
<i>6.1.1 Caracteristici ale protocolului IP</i>	198
<i>6.1.2 Pachetul IPv4</i>	201
<i>6.1.3 IPv6 Packet</i>	204
<i>6.1.4 Rutarea – Cum rutează un Host</i>	209
<i>6.1.5 Tabela de rutare a Routerului</i>	213
<i>6.2 Routerele – Anatomia unui Router</i>	222
<i>6.4.1 Bootarea Routerului</i>	226
<i>6.4.2 ConFig.rea Routerului – Setările Inițiale de ConFig.re</i>	230
<i>6.5 Concluzii Capitolul 6</i>	236
CAPITOLUL 7. NIVELUL TRANSPORT	238
<i>Introducere</i>	238
<i>7.1 Protocolele de la nivelul transport – Transportul datelor</i>	238
<i>7. 1.1 Urmărirea conversațiilor individuale</i>	239

7.2 Introducere în lumea protocoalelor de comunicație : TCP și UDP	245
7.2.1 Transmission Control Protocol (TCP)	245
7.2.2 User Datagram Protocol (UDP)	247
7.3 Utilizarea ambelor protocoale TCP și UDP	251
7.4 TCP și UDP procesează segmentări diferențiate.....	252
7.5 Comunicații TCP	253
7.6 Încrederea și controlul fluxului	261
7.6.1 Confirmarea Recepționării Segmentelor	262
7.6.2 Manevrarea Segmentelor Pierdute	263
7.6.3 Controlul fluxului	265
7.6.4 Reducerea ferestrei de lucru - Window Size	266
7.7 Comunicații UDP	267
7.8 TCP sau UDP, aceasta este intrebarea.....	269
7.9 Concluzii Capitolul 7	271
CAPITOLUL 8. IP ADDRESSING	273
Introducere	273
8.1 Adresele de Rețea IPv4 – Structura Adreselor IPv4	273
8.1.1 Notația Pozițională	274
8.4 Masca de Rețea pentru adresa de tip : IPv4	277
8.5 Prefixele de rețea	278
8.6 Adresa de rețea	279
8.7 Adresa de broadcast.....	279
8.8 Adresele de host	280
8.8.1 Prima adresă de host.....	280
8.8.2 Ultima adresă de host	280
8.9 Operația ȘI LOGIC – ANDing	281
8.10 Addresele IPv4 Unicast, Multicast și Broadcast.....	282
8.10.1 Atribuirea static	282
8.10.2 Atribuirea dinamică	283
8.11 Tipuri de adrese IPv4	286
8.11.1 Adrese private	287
8.11.2 Adrese publice	287
8.11.3 Blocul de adrese Clasă A	289
8.11.4 Blocul de adrese Clasă B	289
8.11.5 Blocul de adrese Clasă C	289
8.11.6 Adresarea de tip Classless	290
8.12 IANA și RIRs	290

<i>8.13 ISPs</i>	291
<i>8.13.1 Servicii ISP</i>	291
<i>8.13.2 ISP Tiers</i>	292
<i>8.14 Adresele de Rețea IPv6</i>	293
<i>8.14.1 Necesitatea de adresare IPv6</i>	293
<i>8.14.2 Internetul pentru toate lucrurile</i>	293
<i>8.14.3 Adresarea IPv6</i>	295
<i>8.14.4 Tipuri de adrese IPv6</i>	298
<i>8.15 ConFig.rea Routerului</i>	302
<i>8.15 ConFig.rea Hostului</i>	302
<i>8.15.1 Stateless Address AutoconFig.ation (SLAAC)</i>	303
<i>8.15.2 DHCPv6</i>	304
<i>8.15.3 IDul Interfeței</i>	305
<i>8.15.4 EUI-64</i>	305
<i>8.15.5 ID de Interfață Generat Aleator</i>	306
<i>8.15.6 Asignarea Adresei de Link-Local Dinamic</i>	307
<i>8.15.7 Static Link-Local Address</i>	307
<i>8.15.8 Adresele IPv6 Multicast</i>	309
<i>8.16 Verificarea Connectivității</i>	311
<i>8.16.1 ICMP</i>	311
<i>8.16.2 Router Solicitation și Router Advertisement Messages</i>	312
<i>8.17 Testare și Verificare</i>	314
<i>8.18 Concluzii Capitolul 8</i>	316
CAPITOLUL 9. SUBNETAREA REȚELELOR IP	318
<i>Introducere</i>	318
<i>9.1 Subnetarea Rețelelor IPv4 – Segmentarea Rețelelor</i>	318
<i>9.1.1 Subnetarea IP este FUNDAMENTALĂ</i>	320
<i>9.1.2 Subnetarea unei Rețele IPv4</i>	321
<i>9.1.3 Determinarea Măștii de Rețea</i>	330
<i>9.1.4 Beneficiile Maștilor cu Lungime Variabilă (Variable Length Subnet Masking-VLSM)</i>	333
<i>9.2 Scheme de Adresare – Proiectarea Structurii</i>	338
<i>9.3 Considerații pentru Proiectarea IPv6</i>	340
<i>9.3.1 Subnetarea unei Rețele IPv6</i>	340
<i>9.4 Concluzii Capitolul 9</i>	342
CAPITOLUL 10. NIVELUL APLICAȚIE	344
<i>Introducere</i>	344
<i>10.1 Protocolele de la nivelul aplicație</i>	344

<i>10.2 Nivelul Aplicație</i>	345
<i>10.3 Nivelul Prezentare</i>	345
<i>10.4 Nivelul Sesiune</i>	346
<i>10.5 Cum Protocolele Aplicație Interacționează cu Aplicațiile Utilizator</i>	347
<i>10.5.1 Rețelele P2P</i>	347
<i>10.5.2 Protocole și servicii bine-cunoscute de la nivelul aplicație</i>	350
<i>10.6 Furnizarea Serviciilor de Adresare IP</i>	355
<i>10.7 Furnizarea Serviciilor de Partajare de Fișiere</i>	360
<i>10.8 Mesajele pot fi auzite în întreaga lume</i>	362
<i>10.9 Concluzii Capitolul 10</i>	367
CAPITOLUL 11. ESTE O REȚEA	368
<i>Introducere</i>	368
<i>11.1 Creare și Dezvoltare. Echipamentele în Rețelele Mici</i>	368
<i>11.2 Serviciile și Caracteristicile Sistemului de Operare</i>	370
<i>11.2.1 Aplicațiile de rețea</i>	373
<i>11.3 Păstrarea rețelei în siguranță - Măsuri de securitate a dispozitivelor de rețea</i>	378
<i>11.3.1 Vulnerabilități și atacuri de rețea</i>	380
<i>11.3.2 Atacuri de recunoaștere</i>	381
<i>11.3.3 Atacuri de acces</i>	382
<i>11.3.4 Negarea serviciilor</i>	383
<i>11.3.5 Combaterea atacurilor de rețea</i>	384
<i>11.3.6 Autentificarea</i>	385
<i>11.3.7 Autorizarea</i>	386
<i>11.3.8 Contorizarea</i>	386
<i>11.4 Securizarea dispozitivelor</i>	388
<i>11.4.1 Additional Password Security</i>	389
<i>11.5 Acces de la distanță prin SSH</i>	390
<i>11.6 Performanța de bază a rețelei</i>	391
<i>11.7 Comenzile show</i>	395
<i>11.8 Comenzile pe Host și IOS</i>	399
<i>11.9 Verificarea interfețelor de pe router</i>	401
<i>11.10 Verificarea interfețelor de pe switch</i>	401
<i>11.11 Gestionarea fișierelor de config.re IOS</i>	402
<i>11.11.1 Back up și restaurarea fișierelor de config.re</i>	404
<i>11.11.2 Restaurarea config.țiiilor text</i>	404
<i>11.11.3 Backupul Config.țiiilor cu TFTP</i>	405
<i>11.11.4 Restaurarea config.țiiilor cu TFTP</i>	405

<i>11.11.5 ConFig.ții de backup cu un drive USB flash</i>	406
<i>11.11.6 Restaurarea conFig.ților cu un drive flash USB</i>	406
<i>11.11.7 Servicii de rutare integrate</i>	407
<i>11.12 Wireless</i>	409
<i>11.12.1 Service Set Identifier (SSID).....</i>	409
<i>11.12.2 Canalul wireless</i>	409
<i>11.12.3 Wired Equivalency Protocol (WEP).....</i>	410
<i>11.12.4 Wi-Fi Protected Access (WPA)</i>	410
<i>11.13 ConFig.rea routerului integrat</i>	411
<i>11.13.1 Accesarea și conFig.rea unui Linksys Router</i>	411
<i>11.13.2 ConFig.rea unui client wireless</i>	412
<i>11.14 Concluzii Capitolul 11</i>	413
BIBLIOGRAFIE	415

Introducere în Lumea Rețelelor

Bun venit la cursul de introducere în lumea rețelelor. Scopul acestui curs este de a prezenta concepțele fundamentale și tehnologiile de bază din acest domeniu. Materialele de curs online vor ajuta în dezvoltarea abilităților necesare planificării și implementării unor rețele mici într-o gamă de aplicații. Competențele specifice vizate sunt descrise la începutul fiecărui capitol.

În această conjunctură poate fi folosit telefonul mobil, tableta, laptopul sau desktopul pentru a accesa cursul, pentru a vedea rezultatele de pe platforma online, pentru a citi sau revizui textul și pentru a practica folosind mediile interactive.

Prin parcursul acestui curs, studenții se alătură la o comunitate globală legată de tehnologii și scopuri comune. Școli, colegii, universități și alte entități din peste 160 de țări participă la astfel de cursuri. O vizualizare a comunității globale Networking Academy este disponibilă pe <http://www.academynetspace.com>.

Platforma de învățare NetSpace este o parte importantă a experienței generale a cursului pentru studenți și profesori din Networking Academy. Materialele de curs online includ cursuri text și mijloace interactive legate de curs, activități de simulare prin softuri dedicate, laboratoare pe echipamente reale, laboratoare pe echipamente de la distanță și tipuri diferite de chestionare. Toate aceste materiale oferă un feedback important pentru a ajuta în evaluarea progresului fiecărui student.

Materialul din acest curs cuprinde o gamă largă de tehnologii care facilitează modul în care oamenii lucrează, trăiesc, se joacă și învăță prin comunicarea vocală, video sau partajarea altor tipuri de resurse. Rețelistica și Internetul afectează oamenii în mod diferit, în diferite zone ale lumii. Materialul de pe platforma online a fost realizat prin participarea profesorilor din întreaga lume și este important ca cei care participă la acest curs să lucreze individual și împreună cu colegii pentru a face cunoștințele dobândite aplicabile în situația locală din zona în care se află.

E-doing este o filozofie de design care aplică principiul că oamenii învăță cel mai bine prin practică. Curriculum include activități practice încorporate, interactive pentru a ajuta stimularea învățării, pentru a crește reținerea de cunoștințe și pentru a face înțeaga experiență de învățare mult mai bogată – ceea ce va face înțelegerea conținutului mult mai ușoară.

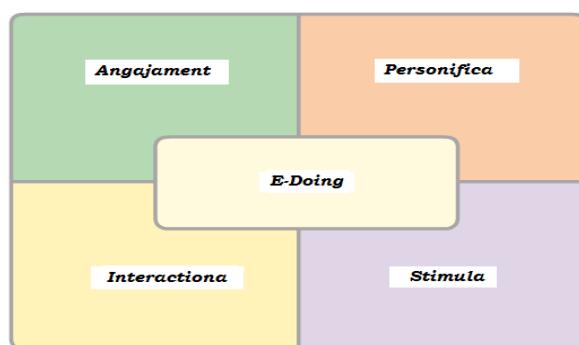


Fig. 1. E-Doing.

Într-o lecție tipică, după ce se învăță despre un subiect pentru prima dată, se va verifica înțelegerea cu ajutorul unor instrumente interactive. Dacă există noi comenzi de invățat, se vor verifica cu ajutorul unui verificator de sintaxă înaintea folosirii lor pentru configurația sau depanarea unei rețele în softuri specifice.

Packet Tracer poate, de asemenea, oferi o practică suplimentară în orice moment prin crearea unor activități proprii sau prin testarea competitivității abilităților cu colegii. Evaluările competențelor în Packet Tracer și laboratoare integrate oferă un feedback bogat al abilităților

dorite pentru a fi dobândite și care reprezintă o bună practică pentru toate capitolele, pentru etapele de verificare și pentru examenele finale.

Un scop important în educație este acela ca fiecare student să-și imbogățească cunoștințele, prin extinderea a ceea ce poate face și ce nu. Este important să se realizeze faptul că materialele și profesorul doar vă pot ușura procesul. Fiecare student trebuie să se angajeze în învățarea noilor competențe.

Profesioniștii în domeniul rețelisticii, de obicei, au un "Engineering Journals" în care pot scrie lucrurile pe care le observă și le învață, cum ar fi modul în care sunt utilizate protocolele și comenzi. Menținerea unor astfel de jurnale crează o referință ce poate fi folosită în diverse situații ICT. Scrierea reprezintă o modalitate de consolidare a învățării, în strânsă corelație cu citirea, vizualizarea și practica.

O conFig.re de probă pentru implementarea unei tehnologii ar putea include comenzi software necesare, scopul acestor comenzi, variabile de comandă și o diagramă de topologie care indică conținutul pentru utilizarea comenzilor și pentru conFig.rea tehnologiei.

Packet Tracer este un instrument de învățare despre rețele care suportă un număr mare de simulări logice și fizice. Oferă, de asemenea, instrumente de vizualizare care ajută la înțelegerea modului de lucru în interiorul rețelei.

Activitățile pre-efectuate din Packet Tracer constau în simulări ale rețelei, jocuri, activități și provocări care oferă multiple experiențe în învățare. Aceste instrumente vor ajuta la dezvoltarea și la înțelegerea modului în care datele circulă într-o rețea.

De asemenea, se poate folosi Packet Tracer pentru a fi create propriile experimente și scenarii de rețea. În timp, este de preferat să fie utilizat Packet Tracer – nu mai pentru experimentarea activităților pre-construite, dar și pentru ca fiecare student să devină autor, explorator sau experimentator.

Materialele de curs online dispun de activități Packet Tracer care se vor lansa pe computer care rulează sisteme de operare Windows sau Linux, în cazul în care Packet Tracer este instalat.

Jocurile Educaționale – Jocurile multi-utilizator Packet Tracer permit unui student sau unei echipe să concureze cu alți studenți pentru a vedea cine poate completa o serie de sarcini de rețea mai rapid. Este un mod excelent de a verifica abilitățile dobândite prin parcurgerea activităților din Packet Tracer și din laboratoare.

Cisco Aspire este un joc pentru un singur jucător, cu simulare strategică de sine stătătoare. Jucătorii își testează abilitățile din rețea prin completarea contractelor într-un oraș virtual. Networking Academy Edition este dezvoltată în mod specific pentru a ajuta în pregătirea pentru examenul de certificare CCENT. Încorporează, de asemenea, abilități de afaceri și de comunicare pe care angajatorii ICT le caută la candidații unui loc de muncă.

Evaluările performanțelor de bază – Evaluările bazate pe performanța Networking Academy făcute în activitățile din Packet Tracer de-a lungul timpului sunt acum integrate cu un motor de evaluare online ce va da automat rezultatele și va oferi un feedback imediat. Acest feedback ajută la identificarea mai precisă a cunoștințelor și abilităților însușite și ce anume trebuie remediat. Există de asemenea întrebări în chestionarele de capitol și examenele ce utilizează activități în Packet Tracer pentru a oferi un feedback adițional în progresul studentului.

Acest curs se axează pe învățarea fundamentelor rețelelor. În acest curs, se vor învăța abilitățile practice și conceptuale care construiesc fundamentul de înțelegere activităților de bază dintr-o rețea.

Pentru aceste deprinderi se vor efectua următoarele:

- Examinare om versus comunicarea peste rețea și se vor vedea paralele dintre ele.
- Parcurgerea celor două modele principale folosite pentru planificarea și implementarea rețelelor : OSI și TCP/IP
- Abordarea pe nivele a rețelelor
- Examinarea nivelelor, din stivele OSI vs TCP/IP, în detaliu pentru a înțelege funcțiile și serviciile lor
- Familiarizarea cu dispozitivele de rețea diverse și cu schemele de adresare din rețea
- Descoperirea modurilor de comunicare folosite pentru a transporta datele peste rețea.
La sfârșitul acestui curs studentul devine capabil să construiască LANuri, să realizeze configurații de bază pentru rutere și switchuri și să implementeze scheme de adresare IP.

CAPITOLUL 1. EXPLORAREA REȚELELOR

Introducere

Omenirea se află acum într-un punct critic - de cotitură - în folosirea tehnologiei pentru a se extinde și atribui noi abilități de comunicare. Globalizarea Internetului a avut un succes mai rapid decât și-ar fi putut imagina cineva. Maniera în care interacțiunile personale, sociale, comerciale și politice au loc se schimbă rapid pentru a ține pasul cu evoluția rețelei globale. În următoarea etapă de dezvoltare umană, inovatorii vor folosi Internetul ca un punct de start al eforturilor lor – creare de noi produse și servicii dezvoltate special pentru a profita de capacitatele rețelei. Deoarece dezvoltatorii își extind limitele cât de mult este posibil, capacitatele rețelelor interconectate care formează Internetul vor avea un rol important în succesul acestor proiecte.

Acest capitol introduce platforma rețelelor de date, de care relațiile noastre sociale și de afaceri depind din ce în ce mai mult. Materialul reprezintă bazele de explorare a serviciilor, tehnologiilor și problemelor întâlnite de profesioniștii în rețelistică în timpul proiectării, construirii și gestionării rețelelor moderne.

La începutul și sfârșitul fiecărui capitol sunt prezentate acitivăți practice. Unele activități pot fi completate individual (acasă sau în laborator), iar unele dintre acestea vor necesita interacțiuni de grup sau cu “comunitatea de învățare”. Profesorul va face tot posibilul pentru ca studenții să participe la toate aceste activități introductory.

Aceste activități vor ajuta la îmbunătățirea înțelegerii prin oferirea unei oportunități de vizualizarea unor concepte abstracte care vor fi învățate în timpul cursului. Se cere creativitate și entuziasm pentru realizarea acestor activități.

Prin parcurgerea acestui capitol ne propunem să atingem următoarele obiective țintă :

- *Descrierea modului în care rețelele au impact asupra vieții noastre de zi cu zi (interacțiunea umană, învățăm, muncim, jocuri).*
- *Descrierea modului în care rețelele suportă comunicația.*
- *Descrierea rolului rețelelor de date.*
- *Descrierea conceptului de rețea convergentă.*
- *Identificarea componentelor-cheie ale oricărei rețele de date.*
- *Compararea echipamentelor și tehnologiilor LAN vs WAN.*
- *Explicarea structurii de bază a Internetului.*
- *Explicarea utilizării echipamentelor de rețea.*
- *Descrierea caracteristicilor în amenințarea securității arhitecturii de rețea:*
 - toleranță la erori,
 - scalabilitate,
 - calitatea serviciilor și de securitate.

Tehnologiile ajută la crearea unei lumi în care :

- *Granițele naționale*
- *Distanțele geografice*
- *Limitele fizice* - devin din ce în ce mai puțin relevante în viața de zi cu zi.

1.2 Rețelistică în prezent

Dintre toate elementele necesare pentru existența umană, nevoia de interacțiune cu alții se află imediat după nevoia noastră de ne a menține viață. Comunicarea este aproape la fel de importantă pentru noi ca încrederea în aer, apă, mâncare și adăpost.

Metodele pe care le folosim pentru a comunica sunt în continuă schimbare și evoluție. Întrucât noi am fost odată limitați la interacțiunile față-în-față, descoperirile tehnologice au extins semnificativ raza de acțiune a comunicațiilor noastre. De la picturile din peșteri la presa scrisă, radio și televiziune, fiecare nouă dezvoltare a îmbunătățit și a consolidat capacitatea noastră de conectare și comunicare cu cei care ne încjoară.

Crearea și interconectarea rețelelor puternice de date a avut un efect profund asupra comunicației și a devenit platforma peste care comunicațiile moderne își desfășoară activitatea.

În lumea de astăzi, prin intermediul utilizării rețelelor, oamenii sunt conectați cum nu au mai fost niciodată. Oamenii cu idei pot comunica instant cu alții pentru a transforma aceste idei în realitate. Evenimentele de știri și descoperirile sunt cunoscute în lumea întreagă în câteva secunde. Indivizii pot chiar să se conecteze și să se joace cu prietenii, separați de oceane sau continente.

Rețelele conectează oamenii și promovează comunicarea nereglementată. Oricine se poate conecta și poate partaja resurse, oricără de mare ar fi distanță între ei.

Să ne imaginăm o lume fără Internet. Nu mai există Google, YouTube, Mesagerie Instant, Facebook, Wikipedia, jocurile online, Netflix, iTunes și accesul ușor la informațiile curente. Nu mai există siteuri de comparare a prețurilor, evitarea de cozi prin cumpărarea online sau căutarea rapidă a unor numere de telefoane sau direcții de pe hartă pentru diverse locații, prin intermediul unui simplu click.

Cât de diferită ar fi viața noastră fără aceste lucruri? Aceasta este lumea pe care am trăit-o acum 15, 20 de ani. Dar, cu trecerea anilor, rețelele de date au fost extinse încet și au avut ca scop îmbunătățirea calității vieții oamenilor de pretutindeni.

În cursul zilei, resursele disponibile în Internet ne pot ajuta în:

- Postarea și partajarea informațiilor, fotografiilor, imaginilor cu prietenii sau cu toată lumea.
- Accessarea și informarea despre activitatea educațională.
- Comunicarea cu familia, cu prietenii, sau colegi folosind aplicații de e-mail, mesagerie instant sau apeluri telefonice prin intermediul Internetului.
- Participare la jocuri online cu prietenii.
- Vizualizarea videoclipurilor, filmelor sau episoadelor de televiziune la cerere.
- Decizia cum să ne îmbrăcăm, ca urmare a vizualizării condițiilor meteo curente.
- Găsirea celei mai puțin aglomerate rute către destinația dorită, afișarea imaginilor video de vreme și trafic prin camere web.
- Verificarea contului bancar și efectuarea de plăți electronice.

Inovatorii au găsit metode de utilizare a Internetului intes în fiecare zi. Deoarece dezvoltatorii își întind limitele cât de mult posibil, capacitatea Internetului și rolul Internetului în viețile noastre se vor extinde din ce în ce mai mult. Să luăm în considerare schimbările care au avut loc în ultimii 25 de ani, conform imaginii prezentate în Fig. 2.

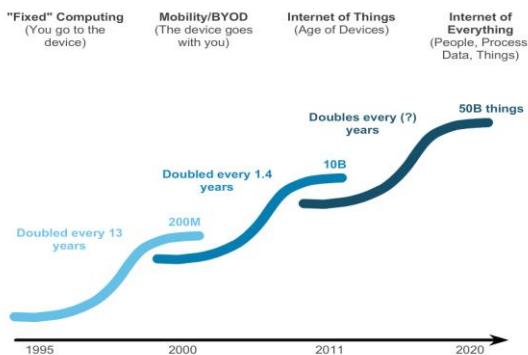


Fig. 1.1. Evoluția utilizatorilor din Internet.

Acum să luăm în considerare schimbările care vor avea loc în următorii 25 de ani. Acest viitor deține conceptul de Internet of Everything (IoE).

IoE unește oameni, procese, date și lucruri pentru a face conexiunile din rețea mai relevante și mai prețioase. Transformă informațiile în acțiuni și creează noi capacitați, experiențe mai bogate și oportunități economice fără precedent pentru oameni, afaceri și țări.

Ce altceva se crede că vom fi capabili să facem cu ajutorul rețelei ca o platformă ?

Progresele în tehnologiile de rețea rezultă, probabil, cei mai semnificativi agenți de schimbare din lumea de astăzi. Aceștia ajută la crearea unei lumi în care granițele naționale, distanțele geografice și limitările fizice devin mai puțin relevante și prezintă obstacole în continuă scădere.

Internetul a schimbat modul în care interacțiunile sociale, comerciale, politice și personale au loc. Natura imediată a comunicațiilor din Internet încurajează crearea comunităților globale. Comunitățile globale permit interacțiunea socială, care depinde de locație sau de fusul orar. Crearea de comunități online pentru schimbul de idei și informații are potențialul de creștere a oportunităților de productivitate din întreaga lume.

Cisco se referă la acest lucru ca fiind o rețea umană. Rețeaua umană concentrează impactul Internetului și rețelelor asupra oamenilor și afacerilor.

În ce mod ne afectează rețeaua umană ?



Fig. 1.2. Rețea umană.

Rețelele și Internetul au schimbat tot ceea ce facem, de la modul de învățare, la modul în care comunicăm, cum lucrăm sau cum ne jucăm.

1.3 Schimbarea modului în care învățăm

Comunicarea, colaborarea și angajamentul sunt blocurile fundamentale de construcție ale educației. Instituțiile se străduiesc continuu să sporească aceste procese pentru a maximiza propagarea de cunoștințe. Metodele de învățare tradiționale oferă, în primul rând, două surse de expertiză prin care studentul poate obține informații: manualul și profesorul. Aceste două surse sunt limitate, ambele în formatul și în calendarul de prezentare.

Rețelele au schimbat modul în care învățăm. Rețelele de încredere și cuprindătoare suportă și îmbogățesc experiențele de învățare ale studentului. Acestea oferă material de învățare într-un număr mare de formate, cum ar fi activități interactive, evaluări și feedback. Conform cu imaginile prezentate în Fig. 4, rețelele acum :

- Suportă crearea de clase virtuale.
- Oferă video "la-cerere".
- Permite spații de colaborare pentru învățare.
- Permite învățarea mobilă.



Fig. 1.3. Noi cai de învățare.

Accesul la instruirea de calitate superioară nu mai este limitată studentilor care trăiesc în zone la mare depărtare față de locul în care instruirea este livrată. Învățarea online de la distanță a îndepărtat barierele geografice și a îmbunătățit oportunitățile studentului. Cursurile online (e-learning) pot fi acum livrate prin intermediul unei rețele. Aceste cursuri pot conține date (text, linkuri), voce și imagini video disponibile studenților în orice moment al zilei din orice loc. Grupurile online de discuții și “panouri” de mesaje permit unui student să colaboreze cu profesorul, cu alți studenți din clasă și chiar cu studenți din întreaga lume. Cursuri combinate pot combina cursuri ținute de profesor cu cursuri online pentru a prelua ceea ce este mai bun din ambele metode disponibile.

În plus, pentru beneficiile studentului, rețelele au îmbunătățit managementul și administrarea cursurilor. Unele dintre aceste funcții online includ integrarea studentului, evaluarea continuă și urmărirea progresului.

1.4 Schimbarea modului în care comunicăm

Globalizarea Internetului a introdus noi forme de comunicare care permit indivizilor să creeze informații care pot fi accesate de către o audiență globală.

Unele forme de comunicare includ:

- Mesageria instant (IM)/Text – IM și text permit comunicare instant, în timp real, între doi sau mai mulți oameni. Multe dintre aplicațiile IM și text încorporează caracteristici precum transferul de fișier. Aplicațiile IM pot oferi caracteristici suplimentare precum comunicare video sau prin voce.
- Social media – Social media constă în siteuri interactive unde oameni și comunități creează și partajează conținut generat de utilizator, cu prietenii, familia, colegii și/sau întreaga lume.
- Instrumente de colaborare – instrumentele de colaborare oferă oamenilor oportunitatea de a lucra împreună și de a partaja documente. Fără constrângerea unei locații sau a unui fus orar, indivizi conectați la un sistem partajat, pot vorbi între ei, adesea prin intermediul unui video.

interactiv, în timp real. Prin intermediul rețelei, pot partaja text și grafice și pot edita documente împreună. Cu ajutorul instrumentelor de colaborare întotdeauna disponibile, organizațiile se pot îndrepta rapid către partajarea de informații și să-și urmărească obiectivele. Distribuția largă a rețelelor de date înseamnă că oamenii din locații aflate la distanță pot contribui în mod egal cu oamenii de la centrele cu populație mare.

- Blogurile – blogurile sunt pagini de web care sunt ușor de actualizat și de editat. Spre deosebire de siteurile comerciale, care sunt create de experți în comunicații profesionale, blogurile oferă oricărei persoane o metodă de comunicare a gândurilor proprii către o audiencă globală, fără cunoștințe tehnice despre web design. Există bloguri cu aproape orice subiect, se pot gândi și comunități de oameni ce se formează adesea în jurul autorilor de bloguri populare.
- Wikis – sunt pagini web pe care grupuri de oameni le pot edita și vizualiza împreună. Spre deosebire de blog, care este mai mult un jurnal personal, individual, wiki este o creație de grup. Acestea fiind spuse, Wikis ar putea fi supus unei editări și verificări mai extinse. Ca și blogurile, wikis pot fi create în trepte, de oricine, fără ajutorul unei sponsorizări a unei întreprinderi comerciale mari. Wikipedia a devenit o resursă cuprinzătoare – o enciclopedie online – de subiecte publicate în mod public. Organizațiile private și indivizii pot de asemenea să-și construiască propriile wikis pentru a expune cunoștințe colectate cu privire la un subiect particular. Multe afaceri folosesc wikis ca un instrument intern de colaborare. Cu ajutorul internetului, oameni din toate colțurile lumii pot participa în wikis și pot adăuga perspective proprii și cunoaștere cu privire la o resursă partajată.
- Podcasting - podcasting este un mediu bazat pe audio care permite inițial oamenilor să înregistreze audio și să convertească fișiere audio pentru utilizarea lor. Podcasting permite oamenilor să trimită înregistrări proprii la o audiencă extinsă. Fișierele audio sunt plasate pe un website (blog sau wiki) de unde alții le pot descărca și le pot asculta pe propriile calculatoare, laptopuri și alte dispozitive mobile.
- Partajarea de fișiere Peer-to-Peer (P2P)-Partajarea de fișiere Peer-to-Peer (P2P) permite oamenilor să partajeze fișiere între ei, fără a necesita stocarea și descărcarea lor dintr-un server central. Utilizatorul se alătură rețelei P2P prin simpla instalare a softwareului P2P, lucru ce le permite să localizeze și să partajeze fișiere cu alții oameni, în rețeaua P2P. Digitalizarea pe scară largă a fișierelor media, precum fișiere video și de muzică, a crescut interesul în partajarea P2P. Partajarea de fișiere P2P nu a fost primită bine de toată lumea. Multi oameni s-au îngrijorat de violarea de legi cu privire la materialele cu drepturi de autor.



Fig. 1.4. Forme de comunicare

1.5 Schimbarea modului în care muncim

În lumea afacerilor, rețelele de date au fost inițial folosite de către afaceriști pentru a înregistra și pentru a gestiona informații financiare interne, informații ale clientului și sistemele de salarizare ale angajatului. Rețelele de afacere au evoluat la permiterea transferului mai multor tipuri diferite de servicii de informații, inclusive e-mail, video, mesagerie și telefonie.

Utilizarea rețelelor pentru a oferi instruire eficientă și rentabilă a angajatului crește în acceptare. Oportunitățile online de învățare pot scădea consumul de timp și călătoriile costisitoare, în timp ce asigură faptul că toți angajații sunt instruiți adecvat pentru a-și efectua joburile lor într-o manieră sigură și productivă.

Există multe povești de succes care ilustrează metode inovatoare în care rețelele au fost folosite pentru a ne oferi mai mult succes la locul de muncă. Unele dintre aceste scenarii sunt disponibile prin intermediul site-ului web Cisco la adresa <http://www.cisco.com>.



Fig. 1.5. Rețele în lumea afacerilor

1.6 Schimbarea modului în care ne jucăm

Adoptarea pe scară largă a Internetului de către industrii de călătorie și divertisment sporește abilitatea de a ne bucura și partaja mai multe forme de relaxare, indiferent de locație. Este posibilă explorarea unor locuri în mod interactiv, la care înainte puteam numai să visăm la vizitarea lor, precum și vizualizarea unei destinații înaintea unei călătorii. Călătorii pot posta online detaliile și fotografiile din aventurile lor pentru ca alții să le vadă.

În plus, Internetul este utilizat pentru forme tradiționale de divertisment. Ascultăm artiști înregistrări, previzualizăm sau vizualizăm imagini în mișcare, putem citi cărți întregi și descărca material pentru un acces offline ulterior. Evenimente de sport online și concerte pot fi experimentate în timp real sau înregistrate pentru o vizualizare la cerere.

Rețelele permit crearea de noi forme de divertisment, cum ar fi jocurile online. Jucătorii participă în orice tip de competiție online pe care dezvoltatorii de jocuri și-o pot imagina. Noi concurăm cu prieteni și amici din jurul lumii în aceeași manieră precum cea în care ei ar fi fost în aceeași cameră cu noi.

Chiar și activitățile offline sunt imbunătățite folosind servicii de colaborare în rețea. Comunitățile globale cu interes în acest subiect au crescut rapid. Noi împărtim experiențe comune și hobbiurile dincolo de cartierul nostru local, oraș sau regiune. Fanii de sport împărtășesc opinii și lucruri despre echipele lor favorite. Colecționarii oferă colecții valoroase și primesc feedback adecvat despre ele.

Magazinele online și siteurile de licitație oferă oportunitatea de cumpărare, vânzare și schimb a tuturor tipurilor de mărfuri.

Indiferent de forma de relaxare pe care o preferăm în rețea umană, rețelele de date convergente ne imbunatășesc experiența.



Fig. 1.6. Forme de relaxare

1.7 Furnizarea de resurse în Rețele

Rețelele sunt de toate dimensiunile. Pot varia de la o rețea simplă ce constă din două computere la o rețea ce conectează milioane de dispozitive.

Rețelele simple instalate în case particulare permit schimbul de resurse, cum ar fi imprimante, documente, fotografii și muzică între câteva computere locale.



Fig. 1.7. Rețele mici de casă

Rețelele office de acasă sau rețele office mici sunt adesea setate de indivizi care lucrează de acasă sau de la un birou de la distanță și care necesită să se conecteze la o rețea corporativă sau la alte resurse centralizate. În plus, mulți antreprenori independenți folosesc biroul de acasă și rețelele de birou mici pentru a face publicitate și a vinde produse, a comanda consumabile și de a comunica cu clienții. Comunicarea în rețea este de obicei mai eficientă și mai puțin costisitoare decât formele tradiționale de comunicare, cum ar fi serviciul poștal sau lungi apeluri telefonice la distanță.



Fig. 1.8. Rețele pentru birou de acasă sau birouri mici

În afaceri și organizații mari, rețelele pot fi utilizate pe o scară mai largă pentru a permite angajaților să furnizeze consolidare, stocare și accesul la informații cu privire la serverele de rețea. Rețelele permit, de asemenea, comunicații rapide cum ar fi e-mail, mesagerie instant și colaborare între angajați. Pentru beneficiile organizatorice interne, multe organizații își folosesc rețelele pentru a oferi produse și servicii clienților prin intermediul conectării lor la Internet.



Fig. 1.9. Rețele medii spre mari

Internetul este cea mai mare rețea existentă. De fapt, termenul de Internet înseamnă “o rețea de rețele”. Internetul este practic o colecție de rețele publice și private interconectate, cum ar fi cele descrise anterior. Rețelele pentru afaceri, rețelele office mici, chiar și rețelele de acasă, de obicei, oferă o conexiune comună la Internet.



Fig. 1.10. Rețele în lumea largă WAN

Este incredibil cât de repede Internetul a devenit o parte integrată a rutinei noastre zilnice.

Toate computerele conectate la o rețea care participă direct la comunicarea de rețea sunt clasificate drept hosturi sau dispozitive finale. Hosturile pot trimite și primi mesaje în rețea. În rețelele moderne, dispozitivele finale pot fi un client, un server sau ambele. Softwareul instalat pe computer determină care rol este jucat de respectivul computer.

Serverele reprezintă hosturi care au software dedicat care le permite să ofere informații, cum ar fi e-mail sau pagini web, către alte hosturi din rețea. Fiecare serviciu necesită un software de server separat. De exemplu, un host necesită un software de server web pentru a oferi servicii web în rețea.

Clienții sunt hosturi care au software instalat ce le permite să facă o cerere și să afișeze informații obținute de la server. Un exemplu de software client este un browser web, precum Internet Explorer.

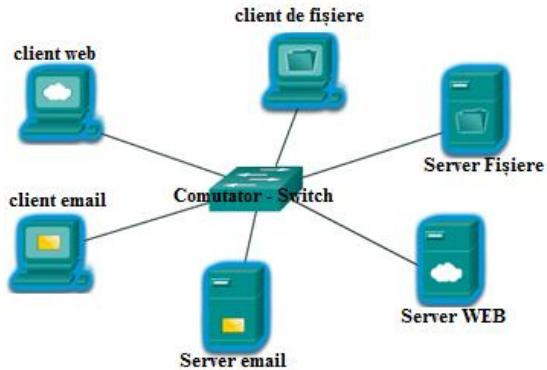


Fig. 1.11.Client de web și Server de web

Un computer cu software de server poate oferi servicii simultan la unul sau mai mulți clienți. În plus, un singur computer poate rula mai multe tipuri de software de server. Într-o casă sau o afacere mică, ar putea fi necesar ca un computer să fie un server de fișiere, server web și server de e-mail.

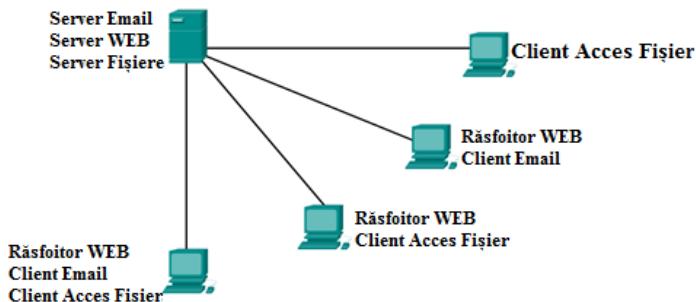


Fig. 1.11. Servere și echipamente cu servicii multiple

Un singur computer poate, de asemenea, să ruleze mai multe tipuri de software client. Trebuie să existe software client pentru orice serviciu necesar. Cu mai mulți clienți instalați, un host se poate conecta la mai multe servere în același timp. De exemplu, un utilizator își poate verifica e-mail și poate vizualiza o pagină web, în timp ce folosește mesageria instant și ascultă radio prin intermediul Internetului.

Softwaerul de server și client, în mod normal, rulează pe computere separate, însă este posibil ca un singur computer să aibă ambele roluri în același timp. În afaceri mici și în case, mai multe computere funcționează ca servere și client în rețea în același timp. Acest tip de rețea se numește de la egal la egal (peer-to-peer).

Cea mai simplă rețea peer-to-peer constă din două computere conectate direct folosind o conexiune cablată sau wireless.

Mai multe calculatoare pot fi conectate pentru a crea o rețea peer-to-peer mai largă, dar este necesar un dispozitiv de rețea, cum ar fi un hub, pentru a interconecta computerele.

Principalul dezavantaj al mediului peer-to-peer este acela că performanța hostului poate fi încetinită în cazul în care se comportă ca server și client în același timp.

În afacerile mari, având în vedere potențialul mare de trafic în rețea, este adesea necesară alocarea de echipamente dedicate (servere) pentru a suporta numărul mare de servicii necesare.

Avantajele retelelor de la egal la egal:

- Complexitate redusa
- Usor de configurat
- Cost redus deoarece nu sunt necesare echipamente dedicate
- Pot fi utilizate pentru procese transfer de fisiere sau share de imprimanta



Dezavantajele retelelor de egal la egal :

- Nu sunt sigure
- Nu sunt scalabile
- Nu pot fi administrate centralizat
- Toate echipamentele pot actiona fie ca servere fie clienti ceea ce poate reduce drastic performanta retelei

Fig. 1.12.

1.8 LANs, WANs, și Internetul - Componentele Rețelelor

Calea pe care masajul o parcurge de la sursă la destinație poate fi la fel de simplă, precum un cablu ce conectează un calculator cu altul, sau complexă precum o rețea care se întinde peste lumea întreagă. Această infrastructură de rețea este platforma care suportă rețeaua, oferă un canal stabil și de încredere peste care comunicațiile au loc.

Infrastructură de rețea conține trei categorii de componente:

- Dispozitive
- Mediile de comunicație
- Servicii

Dispozitivele și mediile de comunicație reprezintă elementele fizice, sau componenta hardware, ale rețelei. Hardware reprezintă adesea componenta vizibilă a platformei de rețea, cum ar fi laptop, PC, switch, router, punct de acces wireless sau cablajul folosit pentru a conecta dispozitivele. Ocazional, unele componente ar putea să nu fie vizibile. În cazul mediului wireless, mesajele sunt transmise prin aer folosind o frecvență radio invizibilă sau unde infraroșii.

Componentele de rețea sunt utilizate pentru a oferi servicii și procese. Sunt programe de comunicații, numite software, care rulează pe dispozitivele din rețea. Un serviciu de rețea oferă informații ca răspuns la o cerere. Serviciile includ mai multe aplicații cunoscute din rețea pe care oamenii le folosesc în fiecare zi, cum ar fi servicii de e-mail și servicii web. Procesele oferă funcționalitatea care direcționează și transferă mesajele prin rețea. Procesele sunt mai puțin vizibile pentru ochiul uman, acestea fiind critice pentru operațiile din rețele.

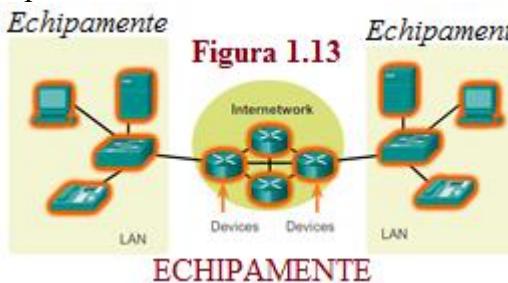


Figura 1.13

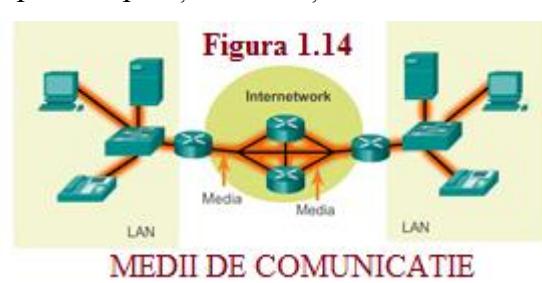
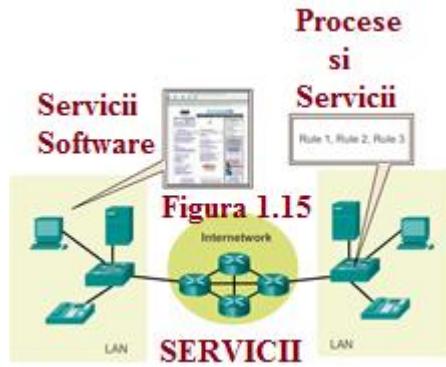


Figura 1.14

MEDIU DE COMUNICATIE

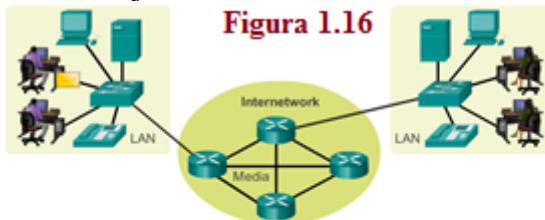


Dispozitivele din rețea cu care oamenii sunt familiari sunt numite dispozitive finale, sau hosturi. Aceste dispozitive formează interfață dintre utilizatori și rețeaua de comunicare ce stă la bază.

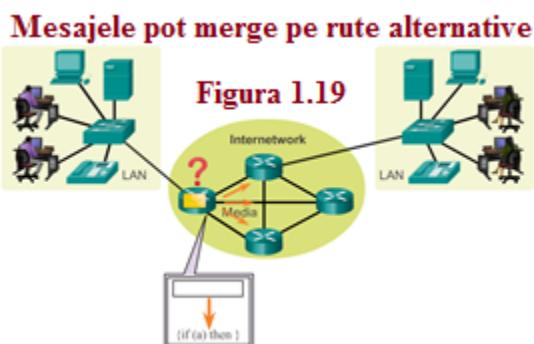
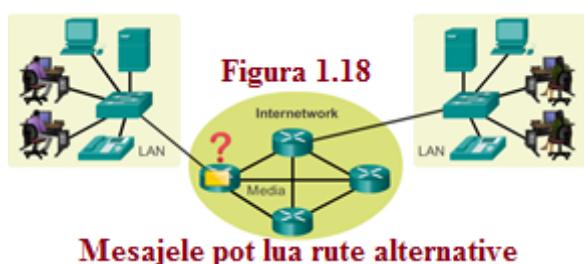
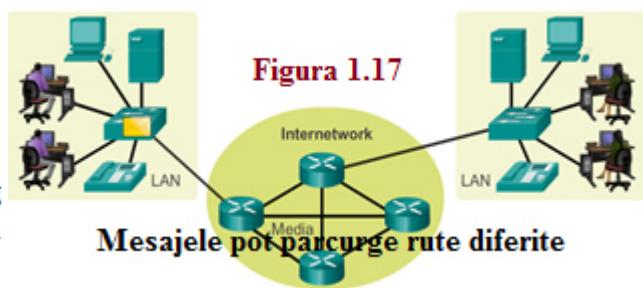
Unele exemple de dispozitive finale sunt:

- Computere (stații de lucru, laptopuri, servere de fișiere, servere web etc.).
- Imprimante de rețea.
- Voce peste protocolul de Internet (VoIP).
- Camera de securitate
- Dispozitive mobile (cum ar fi smartphone, tablete, PDA, cititoare de debit/credit wireless și scanare de coduri de bare)

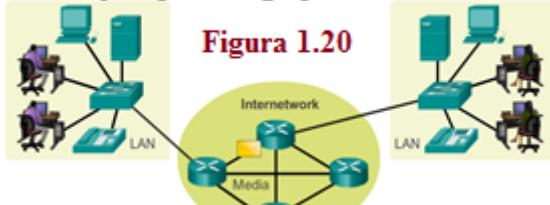
Un dispozitiv host este fie o sursă, fie o destinație a mesajului transmis prin rețea, conform animației. Pentru a distinge un host de celălalt, fiecare host din rețea este identificat de o adresă. Atunci când un host inițiază o comunicare, folosește adresa hostului destinației pentru a specifica unde mesajul trebuie să fie transmis.



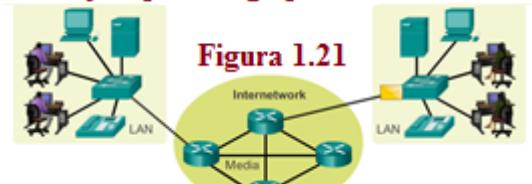
Datele initiate de terminalul gazda, curg peste echipamentele intermediare, pana ajung la terminalul destinatie.



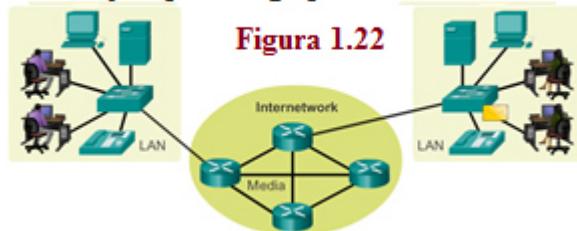
Mesajele pot merge pe rute alternative



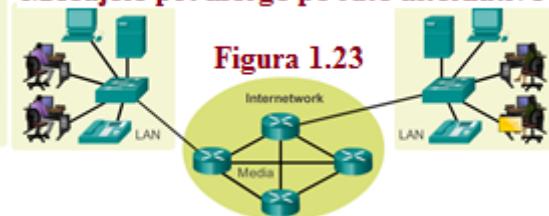
Mesajele pot merge pe rute alternative



Mesajele pot merge pe rute alternative



Mesajele pot merge pe rute alternative



Dispozitivele intermediare interconectează dispozitivele finale. Aceste dispozitive oferă conectivitate și funcționează “în background” pentru a se asigura de faptul că datele traversează rețea în mod corect. Dispozitivele intermediare conectează hosturi individuale la rețea și pot conecta mai multe rețele individuale pentru a forma legătura între rețele diverse ”*internetwork*”.

Exemple de dispozitive de rețea intermediare sunt:

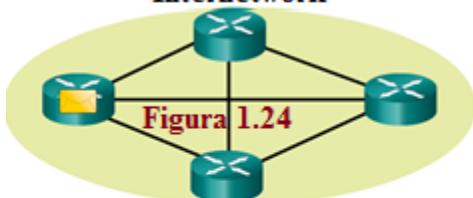
- *Network Access (switchuri și puncte de acces wireless)*
- *Internetworking (routere)*
- *Securitate (firewalluri)*

Managementul datelor să cum sunt transferate în rețea reprezintă de asemenea un rol de dispozitiv intermediar. Aceste dispozitive folosesc adresa hostului destinație, împreună cu informații despre interacțiunile din rețea, pentru a determina calea pe care mesajele ar trebui să o parcurgă prin rețea.

Procesele ce rulează pe dispozitivele intermediare de rețea efectuează următoarele funcții:

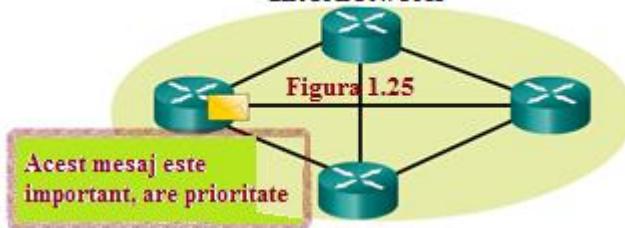
- Regenerează și retransmit semnalele de date.
- Mențin informații cu privire la căile existente în rețea și internetwork.
- Notifică alte dispozitive cu privire la erori și eșecuri de comunicare.
- Direcționează datele pe o cale alternativă în cazul în care există un eșec de legătură(link).
- Clasifică și direcționează mesajele în concordanță cu prioritățile QoS.
- Permit și resping fluxul de date, bazându-se pe setările de securitate.

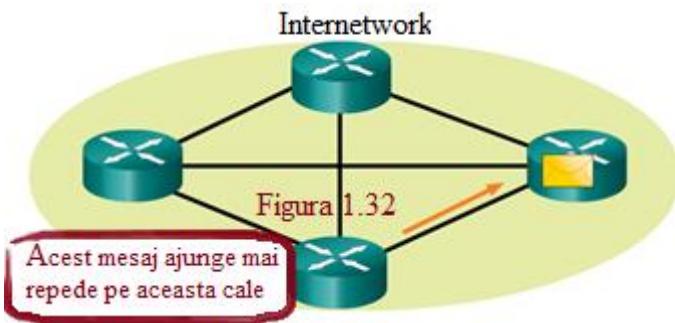
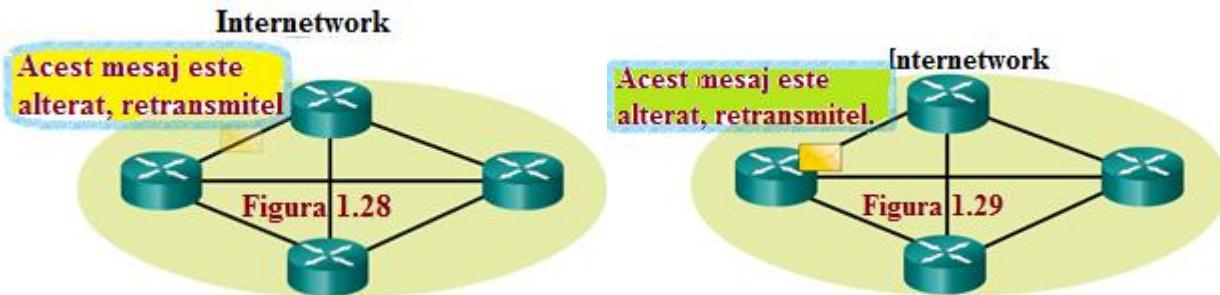
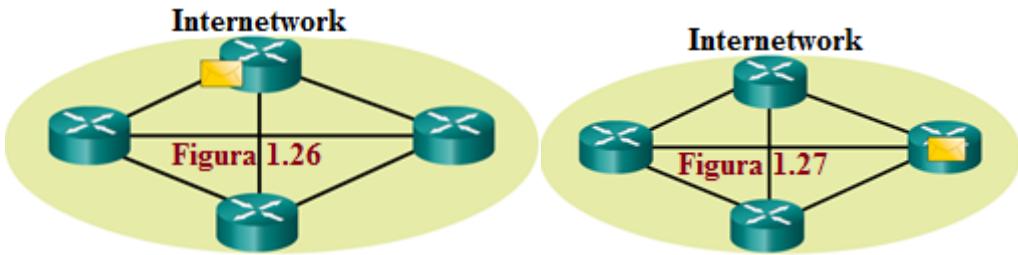
Internetwork



Echipamentele intermediare pot direcționa datele peste rețea, dar nu le pot genera sau să le schimbe continutul.

Internetwork





Comunicările din rețea sunt efectuate peste un mediu de comunicații. Mediul de comunicație oferă canalul peste care mesajul este transmis de la sursă la destinație.

Rețelele moderne folosesc, în primul rând, trei tipuri de medii pentru a interconecta dispozitivele și pentru a oferi calea peste care datele pot fi transmise. Aceste medii sunt:

- *Cabluri cu fire metalice.*
- *Cabluri cu fibre de plastic sau sticlă (cablu fibră optică).*
- *Transmisie fără fir (wireless).*

Codarea semnalului care trebuie să aibă loc pentru ca mesajul să fie transmis este diferită pentru fiecare tip de mediu. În cazul firelor metalice, datele sunt codate în impulsuri electrice care corespund unor modele specifice. Transmisiile prin fibra optică se bazează pe pulsuri de lumină,

fie în interval de lumină vizibilă, fie în infraroșu. În transmisiile wireless, modelele de unde electromagnetice descriu diferite valori de biți.

Diferite tipuri de medii din rețea au diferite caracteristici și beneficii. Nu toate mediile au aceleași caracteristici și nu toate au același scop. Criteriile în alegerea mediului de comunicare sunt:

- *Distanța pe care mediu poate transmite cu succes un semnal.*
- *Mediul în care mediul de comunicare va fi instalat.*
- *Cantitatea de date și viteza cu care trebuie transmise.*
- *Costul mediului și instalării.*



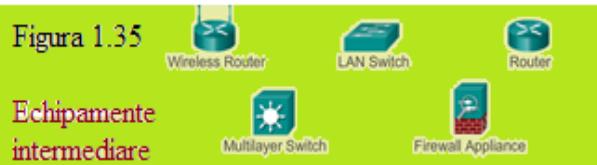
Fig. 1.33. Mediile de comunicație

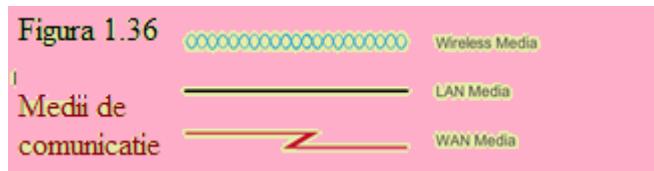
În cazul transmiterii unor informații complexe precum afișarea tuturor dispozitivelor și mediul într-un internetwork mare, este de ajutor reprezentarea vizuală. O diagrame oferă un mod mai ușor de înțelegere a modului în care dispozitivele dintr-o rețea mare sunt conectate. O astfel de diagrame folosește simboluri de reprezentare a diferitelor dispozitive și conexiuni care alcătuiesc o rețea. Acest tip de "Fig." a unei rețele este cunoscută ca fiind *diagrama topologiei*.

Ca orice alt limbaj, limbajul rețelei folosește un set comun de simboluri de reprezentare a diferitelor dispozitive finale, dispozitive de rețea, mediu. Abilitatea de recunoaștere a reprezentărilor logice a componentelor fizice de rețea este critică pentru a fi capabil de a vizualiza organizarea și operarea unei rețele. Prin intermediul acestui curs și laboratoarelor aferente, vom învăța modul în care aceste dispozitive operează și modul în care pot fi efectuate sarcini de configurație de bază a acestor dispozitive.

Pe lângă aceste reprezentări, terminologia specializată este folosită pentru modul în care aceste dispozitive și mediile se conectează între ele. Termenii importanți de reținut sunt:

- *Network Interface Card* – Un NIC sau adaptor LAN, oferă o conexiune fizică de la rețea la PC sau alte dispozitive. Cablul ce conectează PC-UL la dispozitivul de rețea se introduce direct în NIC.
- *Portul fizic* – Un conector sau ”outlet” pe un dispozitiv de rețea unde cablul este conectat la un host sau la alte dispozitive de rețea.
- *Interfață* – Porturi specializate ale unui dispozitiv de rețea ce se conectează la rețele individuale. Deoarece routerele sunt utilizate pentru a interconecta rețele, porturile de pe un router sunt referite ca interfețe.





Diagramale de topologie sunt obligatorii pentru oricine lucrează cu o rețea. Oferă o hartă vizuală a modului în care rețeaua este conectată.

Există două moduri de diagrame de topologie ce includ:

- *Diagrame de topologie fizică* – identifică locația fizică a dispozitivelor intermediare , porturile configurate și instalarea de cablu.
- *Diagrame de topologie logică* - identifică dispozitive, porturi , schema de adresare IP.

Figura 1.37 Topologia Fizica

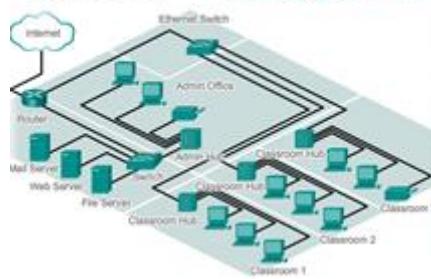
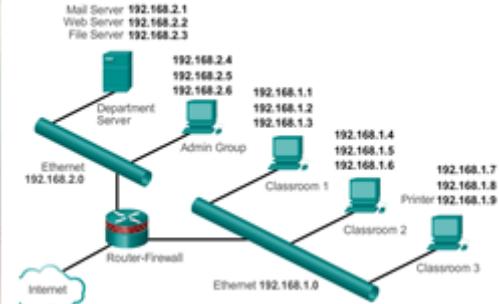


Figura 1.38 Topologia Logica



1.9 LANs and WANs

Infrastructurile de rețea pot varia foarte mult în ceea ce privește:

- Dimensiunea ariei de acoperire
- Numărul de utilizatori conectați
- Numărul și tipurile de servicii disponibile

Fig. de mai jos ilustrează două dintre cele mai comune tipuri de infrastructuri de rețea:

- *Local Area Network (LAN)* - O infrastructură de rețea care oferă acces la utilizatori și dispozitivele finale într-o arie geografică mică.
- *Wide Area Network (WAN)* - O infrastructură de rețea care oferă acces la alte rețele peste o arie geografică largă.

Alte tipuri de rețele includ:

- *Metropolitan Area Network (MAN)* – O infrastructură de rețea care acoperă o arie mai mare decât LAN, dar mai mică decât WAN (exemplu : un oraș). MANs sunt administrate în mod obișnuit de o singură entitate precum o organizație mare.
- *Wireless LAN (WLAN)* – Este similar cu LAN, dar utilizatorii și echipamentele finale sunt conectate wireless într-o arie geografică mică.
- *Storage Area Network (SAN)* - O infrastructură de rețea dezvoltată pentru a suporta servere de fișiere și pentru a oferi stocarea de date, recuperarea și replicarea lor. Implică servere high-end, mai multe matrici de discuri (numite blocuri) și tehnologie de interconectare Fiber Channel.

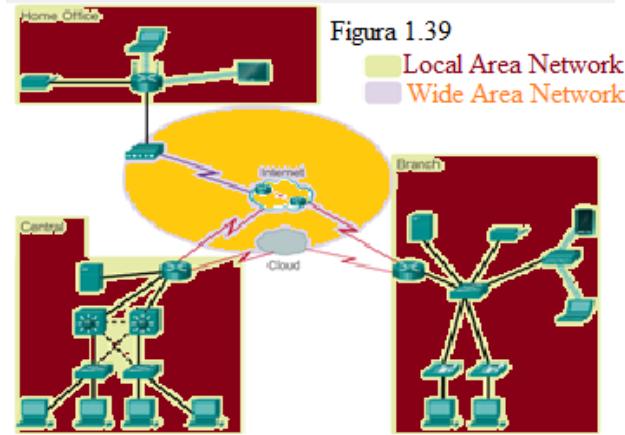


Figura 1.39

Local Area Network
Wide Area Network

LANurile reprezintă o infrastructură de rețea care acoperă o arie geografică mică. Caracteristicile specifice LANurilor sunt:

- *LANurile interconectează dispozitive finale într-o arie limitată cum ar fi acasă, la școală, clădire de birouri sau campus.*
- *Un LAN este de obicei administrat de o singură organizație sau de un singur individ. Controlul administrativ care guvernează politicile de acces și de control este pus în aplicare la nivelul de rețea.*
- *LANurile furnizează lățime de bandă de viteză mare la dispozitivele finale interne sau dispozitivele intermediare.*

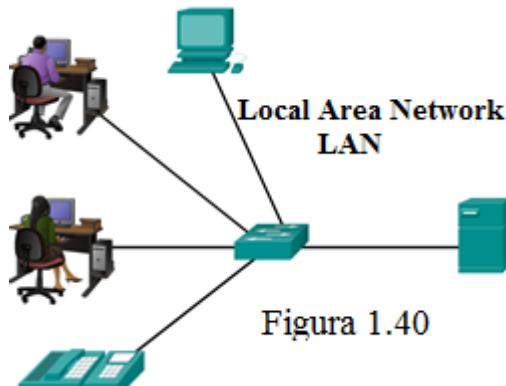


Figura 1.40

WANurile sunt infrastructuri de rețea care acoperă o arie geografică largă. WANurile sunt gestionate, în mod obișnuit, de SP (Service Providers) sau de ISP (Internet Service Providers). Caracteristicile specifice WANurilor includ:

- *WANurile interconectează LANuri, pe o arie geografică mare cum ar fi orașe, state, provincii, țări sau continente.*
- *WANurile sunt, de obicei, administrate de mai mulți furnizori de servicii.*
- *WANurile furnizează, de obicei, legături de viteză mai mică între LANuri.*

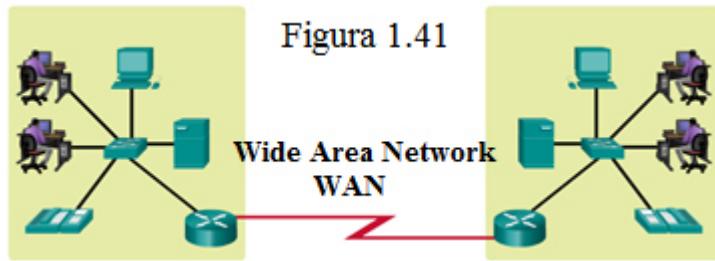


Figura 1.41

1.10 Internetul

Deși există beneficii în folosirea unui LAN sau WAN, mulți indivizi necesită să comunice cu o resursă dintr-o altă rețea, în afara rețelei locale a casei, campusului sau a organizației. Acest lucru se realizează cu ajutorul Internetului.

Cum este arătat și în Fig., Internetul este o colecție de rețele interconectate din întreaga lume (internetworks sau internet pe scurt), care comunică unele cu altele pentru schimbul de informații realizat cu ajutorul standardelor comune. Prin intermediul firelor de telefon, cablurilor optice, transmisilor wireless și legăturilor prin satelit, utilizatorii de Internet pot face schimb de informații într-o mare varietate de forme.

Internetul este un conglomerare de rețele și nu este deținut de nici-un individ sau grup. Asigurarea de comunicare eficientă de-a lungul infrastructurilor diverse necesită aplicarea unor tehnologii și standarde comune și consistente recunoscute, precum și cooperarea între multe agenții de administrare a rețelei. Există organizații care au fost dezvoltate pentru scopul ajutării menținerii structurii și standardizării protocoalelor și proceselor din Internet. Aceste organizații includ Internet Engineering Task Force (IETF), Internet Corporation for Assigned Names and Numbers (ICANN), și Internet Architecture Board (IAB), plus multe altele.

Notă : Termenul de internet (cu literă mică “i”) este utilizat pentru a descrie multiple rețele interconectate. Atunci când ne referim la sistemul global de rețele de calculatoare interconectate sau la World Wide Web, este folosit termenul Internet (cu literă mare “I”).

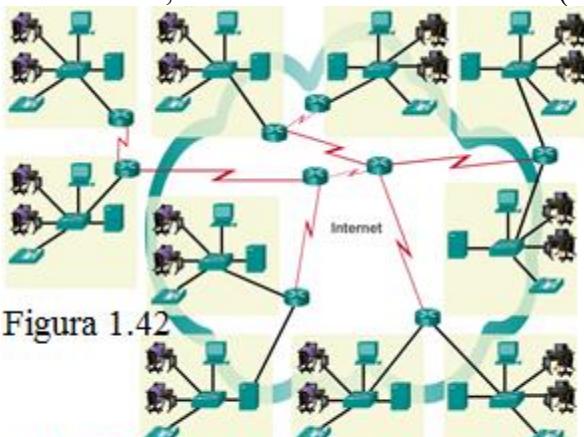


Figura 1.42

LANurile și WANurile pot fi conectate și obținem internetwork

Există doi alți termeni care sunt similari cu termenul Internet:

- *Intranet*
- *Extranet*

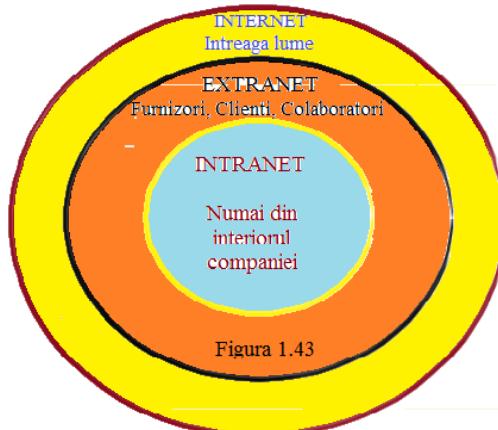
Intranet este un termen utilizat, de obicei, pentru a ne referi la o conexiune privată de LANuri și WANuri care aparțin unei organizații, și care, este dezvoltată pentru a fi accesibilă

numai membrilor organizației, angajaților sau altor oameni care au autorizație. Intraneturile sunt în esență un internet care este, în mod obișnuit, accesibil numai din interiorul organizației.

Organizațiile pot publica pagini web pe un intranet despre evenimente interne, politici de sănătate și siguranță, buletine informative de personal și numere de telefon ale personalului. De exemplu, școlile pot avea intranet care include informații cu privire la programa clasei, curriculum online și forumuri de discuții. Intraneturile, în mod obișnuit, ajută la eliminarea hârtiilor și crește viteza fluxurilor de lucru. Intranetul poate fi accesibil personalului și din afara organizației prin utilizarea unor conexiuni securizate din rețea internă.

O organizație ar putea utiliza un extranet pentru a oferi acces securizat și sigur indivizilor care lucrează pentru diferite organizații, dar necesită datele companiei. Exemple de extraneturi includ:

- *O companie ce oferă acces furnizorilor/contractorilor externi.*
- *Un spital ce oferă un sistem de booking doctorilor pentru ca ei să-și programeze pacienții.*
- *Un birou local de educație ce oferă informații cu privire la personalul și bugetul școlilor din cartierul său.*



1.10.1 Conectarea la Internet

Există multe moduri diferite de conectare a utilizatorilor și organizațiilor la Internet.

Utilizatorii de acasă, teleworkers (muncitorii de la distanță) și birourile mici necesită, în mod obișnuit, o conexiune la ISP pentru a accesa Internetul. Opțiunile de conectare variază mult între ISP și locația geografică. Oricum, alegerile populare includ cabluri de bandă largă, DSL, WANuri wireless și servicii mobile.

Organizațiile, în mod obișnuit, necesită acces la alte locuri corporate și la Internet. Conexiuni rapide sunt necesare pentru a suporta servicii de afaceri ce includ telefoane bazate pe IP, conferințe video și stocarea datelor într-un centru de date.

Interconexiuni business-class sunt, de obicei, furnizate de către SP (furnizorii de servicii). Serviciile populare de business-class includ business DSL, liniile "închiriate" și Metro Ethernet.

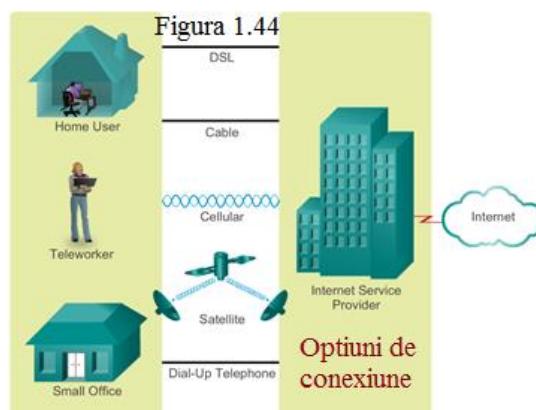
Opțiuni de conexiune comune pentru utilizatorii dintr-un birou mic sau dintr-un birou de casă, includ Fig. 1.44:

- *Cablu – Ofertă, de obicei, de către furnizorii de servicii de televiziune, semnalul de date de Internet este transmis pe același cablu coaxial care transmite televizunea prin cablu. Furnizează o lățime de bandă mare, întotdeauna, pe conexiune de Internet. Un modem de cablu special separă semnalul de date de Internet de celelalte semnale transmise pe cablu și furnizează o conexiune Ethernet pe un calculator gazdă sau LAN.*

- **DSL** – furnizează o lățime de bandă mare, disponibilă întotdeauna, printr-o conexiune la Internet. Necessită un modem special de mare viteză ce separă semnalul DSL de semnalul de telefonie și oferă o conexiune Etherenet pe un calculator gazdă sau LAN. DSL rulează pe o linie de telefonie, cu linia împărțită în trei canale.
 - Un canal este utilizat pentru apeluri de telefonie. Acest canal permite ca un individ să primească apeluri de telefonie fără deconectarea de la Internet.
 - Al doilea canal este un canal de descărcare mai rapid, utilizat pentru a primi informații din Internet.
 - Al treilea canal este utilizat pentru trimiterea sau încărcarea de informații. Acest canal este, de obicei, mai încet decât canalul de descărcare. Calitatea și viteza conexiunii DSL depinde în mare parte de calitatea liniei de telefonie și de distanță de la biroul central al companiei de telefonie. Cu cât locația se află mai departe de biroul central, cu atât conexiunea este mai slabă.
- **Celular** – Accesul la Internet Cellular folosește o rețea de telefonie mobilă pentru a se conecta. Ori de câte ori este acces la un semnal celular, se poate avea acces la Internet cellular. Performanța va fi limitată de către capacitatele telefonului sau turnului de telefonie la care este conectat. Disponibilitatea accesului la Internetul cellular este un beneficiu real în acele arii în care nu ar exista conectivitate la Internet sau pentru acei ce sunt continuu în mișcare.
- **Satelit** – Serviciul de satelit este o bună opțiune pentru case și birouri care nu au acces la DSL sau cablu. Antenele de satelit au nevoie de o linie clară de vedere a satelitului și, de aceea, ar putea fi dificil de accesat în zone puternic împădurite sau locuri cu alte obstacole deasupra lor. Vitezele vor varia în funcție de contract, deși acestea sunt în general bune. Costurile de echipament și instalare pot fi ridicate (aflăm de la furnizorul de oferte speciale) cu o taxă lunară stabilită ulterior. Disponibilitatea accesului la Internet prin satelit este un beneficiu real în acele arii în care altfel nu ar avea nici-o conexiune la Internet.
- **Dial-up Telephone** – O opțiune ieftină care utilizează orice linie telefonică și un modem. Pentru a se conecta la ISP, un utilizator apelează numărul de telefon de acces ISP. Lățimea de bandă scăzută oferită printr-o conexiune prin modem dial-up este de obicei insuficientă pentru transferul de date mari, deși este utilă pentru accesul mobil în timpul unei călătorii. O conexiune prin modem dial-up ar trebui să fie luată în considerare numai atunci când opțiuni de conexiune mai rapide nu sunt disponibile.

Mai multe locuințe și birouri mici sunt mai degrabă conectate direct cu ajutorul cablurilor de fibră optică. Acest lucru permite ca un furnizor de servicii de Internet să ofere viteze de lățime de bandă mare și suportă mai multe servicii precum Internet, telefonie și televiziune.

Alegerea conexiunii variază în funcție de locația geografică și furnizorul de servicii disponibil.



Opțiunile de conexiune corporative diferă în funcție de opțiunile utilizatorului de acasă. Afacerile pot necesita lătime de bandă mai mare, lătime de bandă dedicată sau servicii de administrare. Opțiunile de conexiune disponibile diferă în funcție de numărul de furnizori de servicii disponibili în zona de acoperire.

Fig. 1.45 ilustrează opțiuni comune de conexiune pentru organizații, ce includ:

- **Dedicated Leased Line** – Aceasta este o conexiune dedicată clientului de la furnizorul de servicii. Liniile „închiriate” sunt în esență circuite rezervate care conectează geografic birouri separate pentru networkingul privat de voce și/sau date. Circuitele sunt în mod normal închiriate cu o rată lunară sau anuală ce are tendința să fie mare (scumpă). În America de Nord, circuite comune de linie „închiriată” includ T1 (1.54 Mb/s) și T3 (44.7 Mb/s) care în alte părți ale lumii sunt disponibile ca E1 (2 Mb/s) și E3 (34 Mb/s).
- **Metro Ethernet** - Metro Ethernet este disponibil în mod normal de la un furnizor la client peste o conexiune dedicată de cupru sau fibră oferind viteze de lătime de bandă de la 10 Mb/s la 10 Gb/s. Ethernet over Copper (EoC) este mai economic decât serviciul Ethernet peste fibra optică în multe cazuri, destul de disponibil pe scară largă, și atinge viteze mai mari decât 40 Mbps. Însă, Ethernet over Copper este limitat de distanță. Serviciul Fiber optic Ethernet oferă cele mai rapide conexiuni disponibile la un preț economic pe megabit. Din nefericire, există încă multe arii unde acest serviciu este indisponibil.
- **DSL** – Business DSL este disponibil în mai multe formate. O alegere populară este Symmetric Digital Subscriber Lines (SDSL) care este similar cu Asymmetric Digital Subscriber Line (ADSL), dar oferă aceleași viteze de încărcare și de descărcare. ADSL este proiectat pentru a transmite lătime de bandă de rate diferite ale fluxului de descărcare decât ale celui de încărcare. De exemplu, un client ce are acces la Internet ar putea avea rate ale fluxului de descărcare în intervalul 1.5 Mbps - 9 Mbps, pe când lăimea de bandă a fluxului de încărcare se află în intervalul 16, 640 kbps. Transmisiile ADSL funcționează la distanțe mai mari decât 18,000 feet (5,488 metrii) peste un singur cablu de cupru cu perechi răsucite (torsadate).
- **Satellite** – serviciul satelit poate oferi o conexiune atunci când o soluție cablată nu este disponibilă. Antenele satelit necesită o linie clară de vedere către satelit. Conexiunile au tendința de a fi mai încete și mai puțin de încredere decât competiția terestră, ceea ce le face mai puțin atractive decât alternativele.

Opțiuni comune de conexiune pentru organizații

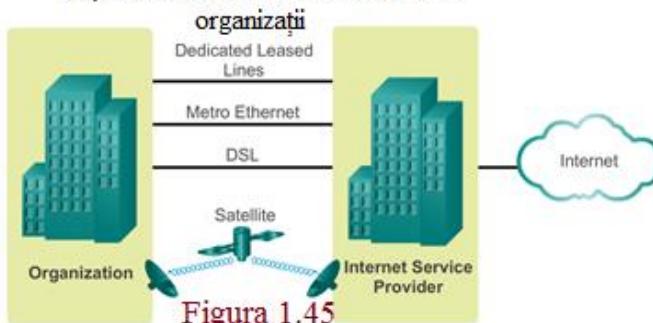


Figura 1.45

Alegerea conexiunii variază în funcție de locația geografică și de furnizorul de servicii disponibil.

Packet Tracer este un program software flexibil, interactiv, disponibil de acasă, care ajută în studierea rețelelor și pentru Cisco Certified Network Associate (CCNA). Packet Tracer permite experimentarea comportamentului rețelei, modelelor de rețea construite și permite punerea întrebării „dacă”. În această activitate, explorăm o rețea relativ complexă care pune în lumină câteva dintre caracteristicile Packet Tracer. Prin efectuarea acestui lucru, învățăm modul de

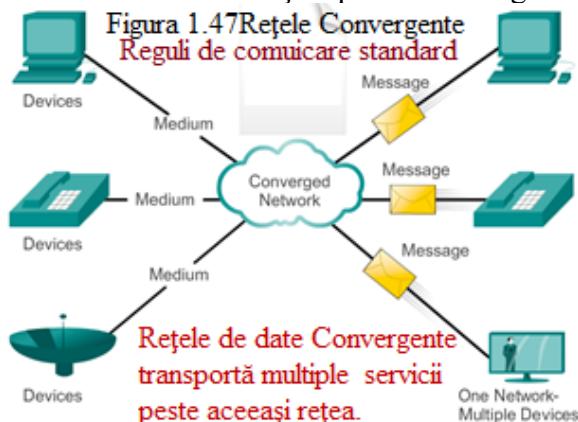
accesare Help și a tuturialelor, de asemenea, modul de trecere între mai multe modele și spații de lucru. În final, explorăm modul în care Packet Tracer servește ca un instrument de modelare pentru reprezentările de rețea.

1.11 Rețeaua ca o Platformă - Rețele convergente

Rețelele moderne evoluează continuu pentru a răspunde noilor cerințe. Rețelele de date de la început erau limitate în schimbul de informații între sistemele de calcul conectate. Rețelele tradiționale de telefon, radio și televiziune erau gestionate separat de rețelele de date. În trecut, fiecare dintre aceste servicii necesitau o rețea dedicată, cu canale de comunicare diferite și tehnologii diferite pentru a transmite un semnal de comunicare particular. Fiecare serviciu avea propriul set de reguli și standarde pentru a asigura succesul comunicării.

Să ne gândim la o școală construită acum 40 de ani. Atunci, clasele erau cablate pentru rețelele de date, rețeaua de telefonie și rețeaua video pentru televiziune. Aceste rețele separate au fost disparate; acest lucru înseamnă că ele nu puteau comunica între ele, conform cu Fig. 1.46 :

Evoluțiile tehnologice permit consolidarea acestor tipuri diferite de rețele într-o singură platformă numită "rețea convergentă". Spre deosebire de rețelele dedicate, rețelele convergente sunt capabile de transmiterea de voce, fluxuri de date, text și grafică între mai multe tipuri de dispozitive, pe același canal de comunicare și aceeași structură de rețea, conform cu Fig. de mai jos. Formele anterioare de comunicare distincte și separate converg într-o platformă comună.



Pe o rețea convergentă există multe puncte de contact și multe dispozitive specializate cum ar fi computere personale, telefoane, TV și tablete, dar ele aparțin aceleiași infrastructuri de rețea. Această infrastructură de rețea utilizează același set de reguli, acorduri și standarde de implementare.

Convergența diferitelor tipuri de rețele de comunicare într-o singură platformă reprezintă prima fază în construirea unei rețele inteligente de informații. Ne aflăm acum în faza evoluției rețelelor. Următoarea fază va fi consolidarea nu numai a tipurilor diferite de mesaje într-o singură rețea, dar și consolidarea aplicațiilor care generează, transmit și securizează mesajele în dispozitivele de rețea integrate.

Nu numai că vocea și video vor fi transmise pe aceeași rețea, ci și dispozitivele care efectuează "telephone switching" și "video broadcasting" vor fi aceleași dispozitive ce rutează mesajele în rețea. Platforma de comunicații rezultată va oferi funcționalitate aplicației de înaltă calitate la un cost redus.

Ritmul în care dezvoltarea de noi aplicații interesante pentru rețele convergente are loc poate fi atribuit la creșterea rapidă și expansiunea Internetului. Cu numai aproximativ 10 miliarde

(din 1.5 trilioane) de lucruri conectate acum global, există un potențial vast de conectare a lucrurilor neconectate prin intermediul IoE. Această expansiune a creat un public global pentru orice mesaj, produs sau serviciu ce poate fi transmis.

Mecanica de bază și procesele care conduc această creștere explozivă au ca rezultat o arhitectură de rețea care este capabilă atât în suportarea schimbărilor, cât și în creștere. Ca platformă tehnologică de sprijin pentru viață, învățare, lucru și joacă în rețeaua umană, arhitectura de rețea a Internetului trebuie să se adapteze constant cerințelor de schimbare pentru o calitate superioară a serviciului și securității.



Figura 1.48 Rețele Inteligente

1.12 Rețea de încredere

Rețelele trebuie să suporte un interval larg de aplicații și servicii, precum și să opereze peste multe tipuri diferite de cabluri și dispozitive, ce alcătuiesc infrastructura fizică. Termenul de arhitectură de rețea, în acest context, se referă la tehnologiile ce suportă infrastructura, serviciile și regulile programate – protocolele – care deplasează mesajele prin rețea.

În timp ce rețelele evoluează, descoperim că acestea păstrează patru caracteristici de bază de care arhitecturile de bază au nevoie pentru a răspunde cerințelor utilizatorilor:

Toleranța la defecte – Așteptarea este accea că Internetul să fie întotdeauna disponibil pentru milioane de utilizatori care se bazează pe el. Acest lucru necesită o arhitectură de rețea construită astfel încât să fie tolerantă la erori. O rețea tolerantă la erori este accea ce limitează impactul unui eșec, astfel un număr scăzut de dispozitive sunt afectate în cazul unui eșec. Este de asemenea construită astfel încât permite o recuperare rapidă în cazul în care un astfel de eșec are loc. Aceste rețele depind de mai multe căi de legătură între sursă și destinație a unui mesaj. În cazul în care o cale „pică”, mesajele pot fi instant transmise printr-o altă legătură. Faptul că există mai multe căi spre o destinație, este cunoscut ca redundanță.

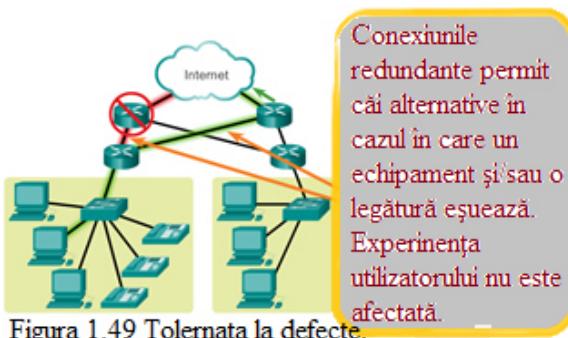
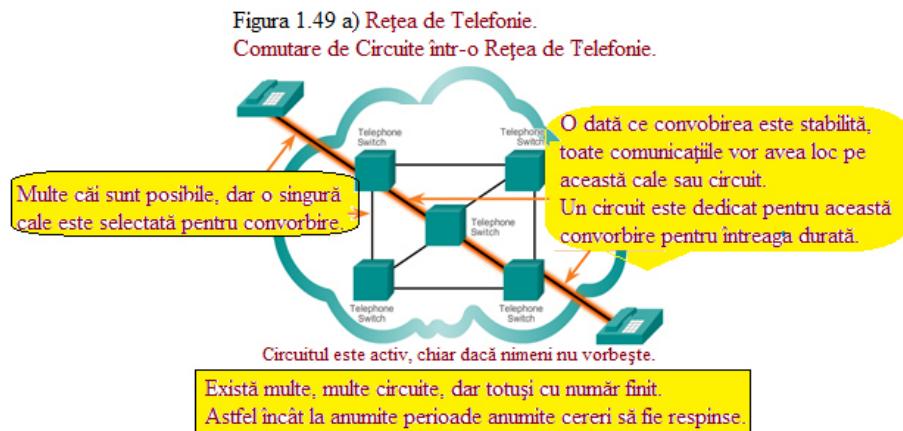


Figura 1.49 Tolerația la defecte.

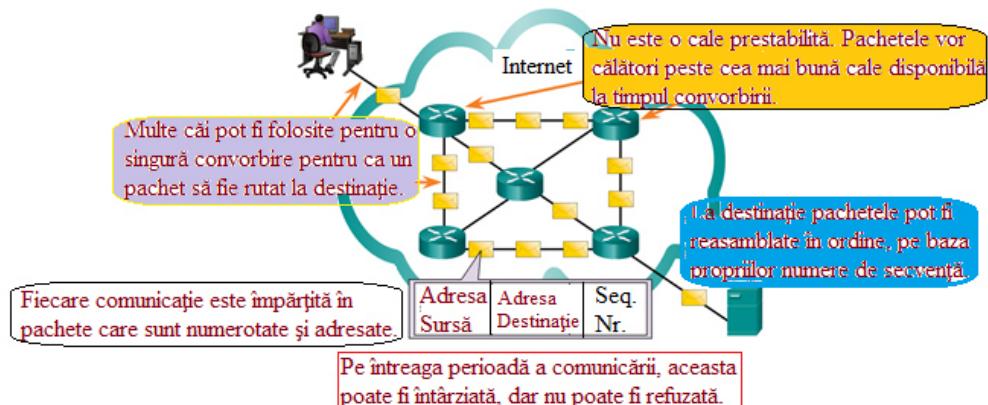
Rețele orientate pe conexiune bazate pe comutare de circuite – Pentru a înțelege necesitatea redundanței, putem să analizăm modul în care sistemele de telefonie funcționează. Atunci când o persoană telefonează folosind un set de telefonie tradițional, apelul în primul rând trece printr-un proces de configurație. Acest proces identifică locațiile de comutare telefonice dintre persoana care efectuează apelul (sursa) și setul de telefonie ce recepționează apelul (destinația). O cale temporară, sau un circuit, a fost creată pentru perioada apelului telefonic. În cazul în care legătura sau dispozitivul din circuit „pică”, apelul este „aruncat – drop down”. Pentru reconectare, un nou apel trebuie să fie efectuat, cu un circuit nou. Acest proces de conectare este referit ca proces de comutare de circuit și este ilustrat în Fig. .



- Multe rețele bazate pe comutare de circuite oferă prioritate conexiunilor de circuit existente în detrimentul noilor cereri de circuit. După ce un circuit este stabilit, chiar dacă nici-o comunicare nu are loc între persoane la nici-un capăt al apelului, circuitul rămâne conectat și resursele sunt în uz până când una dintre părți deconectează apelul. Deoarece există atât de multe circuite ce pot fi create, este posibil să primim un mesaj că toate circuitele sunt ocupate și că un apel nu poate avea loc. Costul de creare a mai multor căi alternative cu o capacitate suficientă pentru a suporta un număr mare de circuite simultane și tehnologiile necesare pentru recrearea dimineață a circuitelor „picăte” în cazul unui eșec, este motivul pentru care tehnologia bazată pe comutare de circuite nu a fost optimă pentru Internet.

Rețele bazate pe comutare de pachete – În căutarea unei rețele care să fie mai tolerantă în cazul unor erori, dezvoltatorii timpurii de Internet au cercetat rețelele bazate pe comutare de pachete. Premisa pentru acest tip de rețea este aceea că un singur mesaj poate fi „spart” în mai multe blocuri de mesaj, cu fiecare bloc de mesaj conținând informații de adresare ce indică punctul de origine și destinația finală. Utilizând aceste informații încorporate, aceste blocuri de mesaj, numite pachete, pot fi transmise prin rețea de-a lungul unor căi variate și pot fi reasamblate în mesajul original atunci când ajung la destinație, conform imaginii următoare.

Figura 1.49 b) Rețea de Date.
Comutare de Pachete într-o Rețea de Date.



- Dispozitivele din rețea sunt, în mod obișnuit, în necunoștință de conținutul pachetelor individuale. Vizibilă este numai adresa destinației finale. Aceste adrese sunt referite, de obicei, ca adrese IP, reprezentate într-un format zecimal cum ar fi "192.168.10.1". Fiecare pachet este trimis independent de la o locație la alta. În fiecare locație, o decizie de rutare este luată conform căreia se alege ce cale este utilizată pentru a transmite pachetul spre destinația finală. Acest lucru poate fi comparat cu scrierea unui mesaj mare pentru un prieten folosind zece cărți poștale. Fiecare carte poștală are adresa destinație a receptorului. În modul de transmitere a cărților poștale prin sistemul poștal, adresa destinație este utilizată pentru a determina următoarea cale pe care cartea poștală trebuie să o urmeze. Eventual, vor fi transmise la adresa de pe cărțile poștale.
- În cazul în care calea anterioară utilizată nu mai este disponibilă, funcția de rutare poate alege în mod dinamic următoarea cale cea mai bună, disponibilă. Deoarece mesajul este transmis în bucăți, mai repede decât un singur mesaj complet, puține pachete pot fi pierdute și retransmise la destinație printr-o cale diferită. În multe cazuri, dispozitivul destinație este în necunoștință de faptul că are loc un eșec sau o rerutare. Folosind analogia cu cartea poștală, dacă una dintre cărțile poștale nu mai este "pe drum", numai acea carte poștală trebuie retransmisă.
- Necesitatea pentru un singur circuit rezervat "end-to-end" nu există într-o rețea ce se bazează pe comutare de pachet. Orice bucata dintr-un mesaj poate fi transmisă prin rețea utilizând orice cale disponibilă. În plus, pachetele ce conțin bucăți din mesaje de la diferite surse pot "călători" prin rețea în același timp. Prin oferirea unei metode ce utilizează dinamic căile redundante, fără intervenția unui utilizator, Internetul a devenit o metodă de comunicare tolerantă la erori. Prin analogie, cărțile poștale călătoresc prin sistemul poștal împreună cu alte cărți poștale, scrisori și cutii. De exemplu, una dintre cărțile poștale poate fi plasată într-un avion, împreună cu alte cutii și scrisori ce sunt transportate la destinația finală.
- Deși rețelele bazate pe comutare de pachete reprezintă principala infrastructură pentru Internetul de astăzi, există unele beneficii pentru un sistem orientat pe conexiune, cum ar fi sistemul de telefonie "circuit-switched". Deoarece resursele din locații de comutare diferite sunt dedicate pentru a oferi un număr finit de circuite, calitatea și consistența mesajelor transmise printr-o rețea orientată pe conexiune pot fi garantate. Un alt beneficiu este acela că furnizorul unui serviciu poate percepe o taxă utilizatorilor unei

rețele pentru perioada de timp în care conexiunea este activă. Capacitatea de a percepe utilizatorilor o taxă pentru conexiunile active din rețea este o premisă fundamentală a industriei serviciului de telecomunicație.

Scalabilitate – Mii de noi utilizatori și furnizori de servicii se conectează la Internet în fiecare săptămână. Pentru ca Internetul să suporte o astfel de creștere rapidă, trebuie să fie scalabil. O rețea scalabilă se poate extinde rapid pentru a suporta noi utilizatori și aplicații fără a avea un impact asupra performanței serviciului transmis utilizatorilor existenți. Fig. prezintă o vizinătură de ansamblu referitoare la structura Internetului.

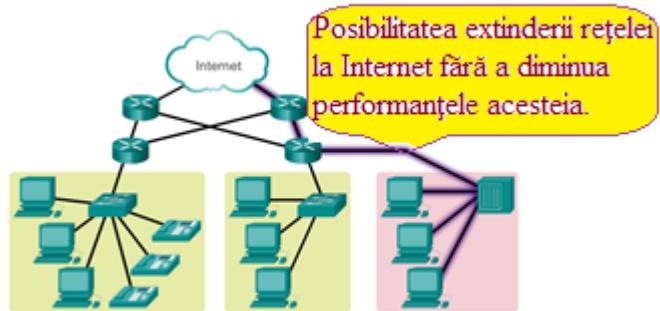


Figura 1.50 Scalabilitatea

În centrul Internetului se află ISP-urile de nivel 1 = Tier1, care furnizează conexiuni naționale și internaționale. Aceste ISP-uri sunt tratate ca fiind egale unele cu altele.

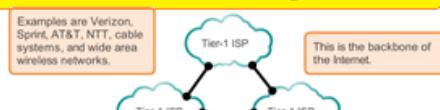


Figura 1.50 a) Nivelul 1

ISP-urile de nivel 2 = Tier2, sunt de obicei mai mici și furnizează servicii locale. În mod usual Tier 2 plătesc la Tier 1 pentru conectivitate în Internet.

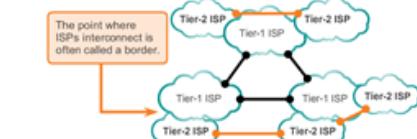


Figura 1.50 b) Nivelul 2

ISP-urile de nivel 3 = Tier3, sunt furnizorii servicii direct către utilizatori. Tier 3 sunt în mod usual conectați la ISP Tier 2 și plătesc acestora pentru conectivitate în Internet.

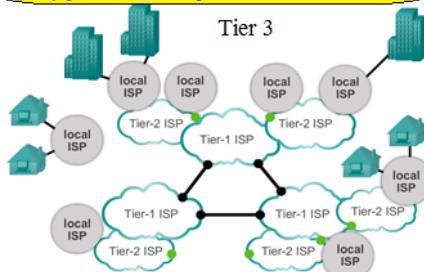


Figura 1.50 c) Nivelul 3

Faptul că Internetul este capabil să se extindă, fără a afecta serios performanța experimentată de utilizatorii individuali, este o funcție a dezvoltării de protocoale și tehnologii de bază a modului în care este construit. Internetul are o structură ierarhică, pe nivele de adresare, pentru serviciile de conectivitate. Ca rezultat, traficul din rețea care este destinat pentru servicii locale sau regionale nu necesită traversarea unui punct central de distribuție. Serviciile comune pot fi dublate în regiuni diferite, păstrând astfel traficul în afara rețelelor backbone de nivel înalt.

Scalabilitatea se referă, de asemenea, la abilitatea de acceptare a unor noi aplicații și produse. Deși nu există o singură organizație care reglementează Internetul, rețelele individuale

care oferă conexiune la Internet cooperează pentru a urma protocolele și standardele acceptate. Aderarea la standarde permite producătorilor de hardware și software să se concentreze pe dezvoltarea de produse și imbunătățiri în ariile de performanță și capacitate, știind faptul că noile produse se pot integra și îmbunătăți infrastructura existentă.

Infrastructura actuală de Internet, deși extrem de scalabilă, s-ar putea să nu poată să țină pasul întotdeauna cu ritmul de cerințe al utilizatorului. Noi protocole și structuri de adresare sunt în curs de dezvoltare pentru a întâlni rata de creștere, la care se adaugă aplicațiile și serviciile din Internet.

Calitatea serviciului – Calitatea serviciului este de asemenea o cerință în continuă creștere a rețelelor zilelor noastre. Noi aplicații disponibile utilizatorilor peste rețele, cum ar fi transmisiile în timp real de voce sau video, aşa cum se poate observa în Fig. de mai jos, crează așteptări mai mari pentru calitatea serviciilor oferite. Calitatea serviciilor poate fi mai ușor de înțeles dacă încercăm să ne uităm la un video cu interruperi și pauze constante.

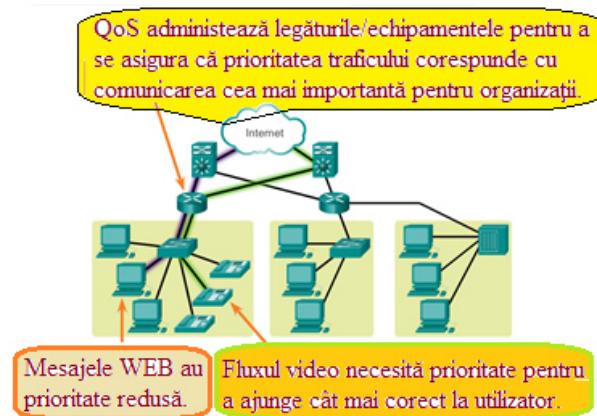


Figura 1.51 Calitatea Serviciilor – QoS

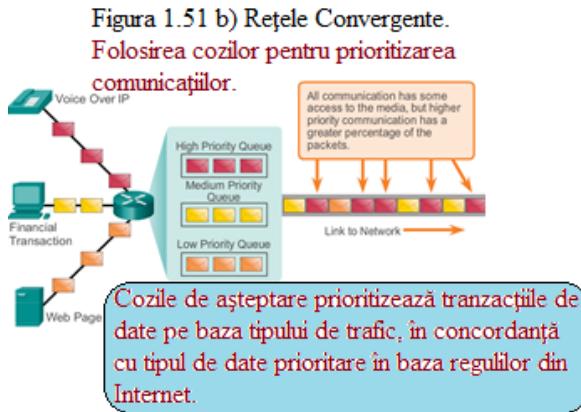
Figura 1.51 a) Rețele Convergente.



Rețelele trebuie să ofere servicii previzibile, măsurabile și uneori, garantabile. Arhitectura de rețea care are la bază comutare de pachete nu garantează faptul că toate pachetele care compun un mesaj particular vor ajunge în același timp, în ordinea corectă și faptul că vor ajunge cu siguranță.

Rețelele, de asemenea, necesită mecanisme de gestiune a traficului de rețea aglomerată. Lățimea de bandă de rețea reprezintă măsura capacitatii de transmitere a datelor în rețea. Cu alte cuvinte, se evaluatează ce cantitate de informații poate fi transmisă într-o anumită perioadă de timp. Lățimea de bandă de rețea se măsoară în numărul de biți ce pot fi transmiși într-o secundă, sau biți pe secundă. Atunci când comunicații simultane au loc în rețea, cererea de lățime de bandă de rețea își poate depăși disponibilitatea, creând astfel aglomerare de rețea. Rețea are mai mulți biți de transmis decât lățimea de bandă a canalului de comunicare poate transmite.

În multe cazuri, atunci când volumul de pachete este mai mare decât poate fi transportat în rețea, dispozitivele rețin pachetele în memorie până când resursele devin disponibile pentru a le transmite, conform imaginii următoare.



Reținerea de pachete provoacă întârzieri deoarece noile pachete nu pot fi transmise până când pachetele anterioare nu au fost procesate. Dacă numărul de pachete reținute continuă să crească, memoria se umple și pachetele sunt "aruncate".

Realizarea QoS necesară prin gestionarea parametrilor de întârziere și pierderile de pachete într-o rețea devine secretul unei soluții de calitate a aplicației end-to-end de succes. Un mod de realizare a acestui lucru este prin clasificare. Pentru a crea clasificări QoS ale datelor, folosim o combinație a caracteristicilor de comunicare și importanța relativă atribuită aplicației, conform de imaginii de mai jos.

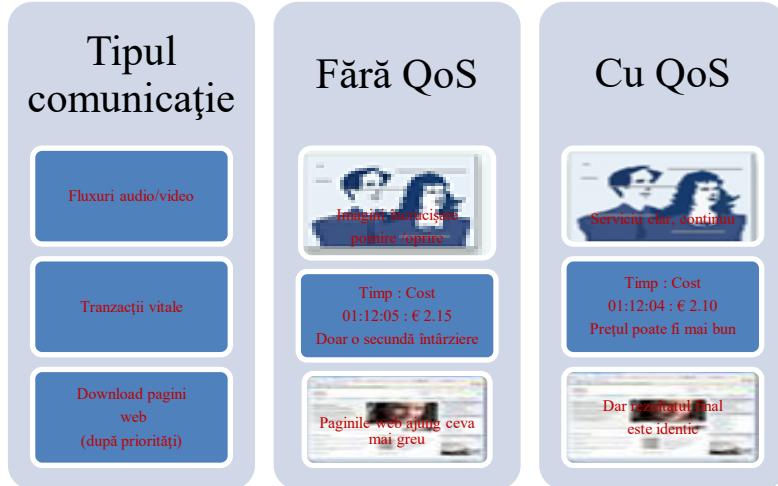


Fig. 1.51 c) Calitatea serviciilor contează

Tratăm toate datele din aceeași clasificare în conformitate cu aceleași reguli. De exemplu, comunicarea sensibilă la timp, cum ar fi transmisiile de voce, va fi clasificată diferit de comunicarea tolerantă la întârziere, cum ar fi transferul de fișiere.

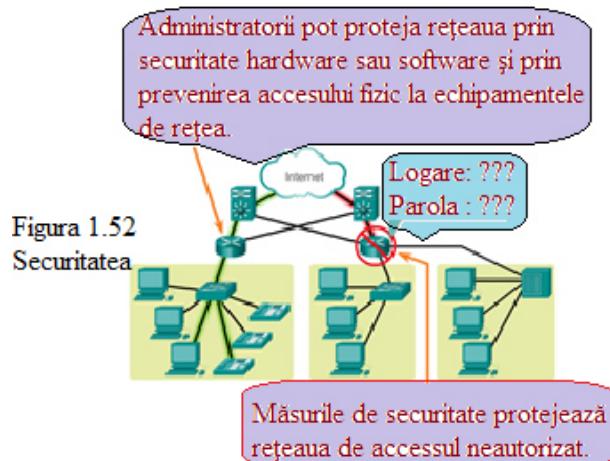
Exemple de decizii de prioritate pentru o organizație ar putea include:

- Comunicare sensibilă la timp – crește prioritatea serviciilor precum telefonie sau distribuție video.
- Comunicare non-sensibilă la timp – descrește prioritatea pentru recuperarea paginii web sau e-mail.
- De importanță mare pentru organizație – crește prioritatea pentru controlul producției sau tranzacțiile de date din afacere.

- Comunicare nedorită – descrește prioritatea sau blochează activitatea nedorită, cum ar fi partajarea de fișiere peer-to-peer sau divertismentul live.

Securitatea – Internetul a evoluat de la un internetwork bine controlat de organizații educaționale și guvernamentale la un mod global accesibil de transmisie a comunicațiilor personale și de afacere. Ca rezultat, cerințele de securitate ale rețelei s-au schimbat. Infrastructura de rețea, serviciile și datele conținute de dispozitivele atașate rețelei sunt active, personale și de afacere. Compromiterea integrității acestor active ar putea avea consecințe serioase, precum:

- *Întreruperi de rețea care împiedică comunicații și tranzacții să aibă loc, cu pierdere consecventă în afacere.*
- *Proprietatea intelectuală (idei de cercetare, brevete sau schițe) furată și utilizată de către un competitor.*
- *Informații personale sau private compromise sau făcute publice fără acordul utilizatorilor.*
- *Dezorientarea și pierderea de fonduri personale sau de afacere.*
- *Pierderea de date importante care necesită o muncă semnificativă pentru a le înlocui, sau care sunt neinlocuibile.*



Există două tipuri de aspecte ale securității de rețea care trebuie luate în considerare :

- **Securitatea infrastructurii de rețea** - Securizarea unei infrastructuri de rețea include securizarea fizică a dispozitivelor care oferă conectivitate la rețea și prevenirea accesului neautorizat la softwareul de gestiune aflat pe ele.
- **Securitatea informațiilor** - Securitatea informațiilor se referă la protejarea informațiilor conținute de pachetele transmise în rețea și informațiilor stocate pe dispozitivele atașate rețelei. Măsurile de securitate într-o rețea ar trebui:
 - *Să prevină dezvăluirea neautorizată.*
 - *Să prevină furtul de informații.*



Figura 1.52 a) Tranzacții Neautorizate.

Fig. 1.63.

- Să prevină modificarea neautorizată a informațiilor.
- Să prevină Denial of Service (DoS).

Pentru a realiza scopurile securității de rețea, există trei cerințe importante, conform figurii:



Figura 1.52 b) Infrastructuri și date private.

- *Asigurarea confidențialității* – Confidențialitatea datelor înseamnă că numai beneficiarii autorizați și destinați – indivizi, procese sau dispozitive – pot accesa și citi datele. Acest lucru se realizează printr-un sistem puternic de autentificare a utilizatorului, introducerea de parole greu de ghicit și cerința ca utilizatorii să le schimbe în mod frecvent. Criptarea datelor, pentru ca numai beneficiarul destinat să le poată citi, este de asemenea o parte a confidențialității.
- *Menținerea integrității comunicării* – Integritatea datelor înseamnă asigurarea faptului că informațiile nu au fost alterate în timpul transmisiei, de la sursă la destinație. Integritatea datelor poate fi compromisă atunci când informațiile sunt corupte – intenționat sau accidental. Integritatea datelor este posibilă să mențină prin cerința validării expeditorului, cât și utilizarea de mecanisme pentru validarea faptului că pachetul nu a fost schimbat în timpul transmisiei.
- *Asigurarea disponibilității* – Disponibilitatea înseamnă asigurarea accesului de încredere și rapid la serviciile de date pentru utilizatorii autorizați. Dispozitivele firewall de rețea, împreună cu softwareul antivirus pentru server sau desktop, pot asigura încrederea sistemului și putere pentru a detecta, respinge și pentru a face față unor asemenea atacuri. Construirea unei infrastructuri de rețea complet redundantă, cu puține puncte de eșec, poate reduce impactul acestor amenințări.

1.13 Schimbarea mediului de rețea - Tendințe de rețea

Atunci când privim la modul în care Internetul a schimbat atât de multe lucruri pe care oamenii le fac zilnic, este greu de crezut că a fost accesibil pentru mulți oameni doar de aproximativ 20-25 de ani. A transformat modul în care indivizii și organizațiile comunică. De exemplu, înainte ca Internetul să devină atât de răspândit, organizațiile și afacerile mici se bazau mult pe marketingul printat pentru a-și face clienții conștienți de produsele lor. A fost dificil pentru afaceri să determine ce case erau potențiali clienți, deci afacerile se bazau pe programe de marketing printat. Aceste programe erau scumpe și variate în eficacitate. Putem compara acest lucru cu modul în care clienții sunt atrași astăzi. Multe afaceri au o prezență în Internet de unde clienții pot învăța despre produsele lor, pot citi despre părerile asupra acelui produs ale altor clienți și pot comanda produse direct de pe site. Bloggerii fac parteneriate cu întreprinderile pentru a evidenția și aproba produse și servicii. Cele mai multe dintre aceste plasări de produse se adresează potențialului consumator, și nu unor mase de oameni. Fig. următoare prezintă mai multe previziuni pentru Internet din viitorul apropiat.

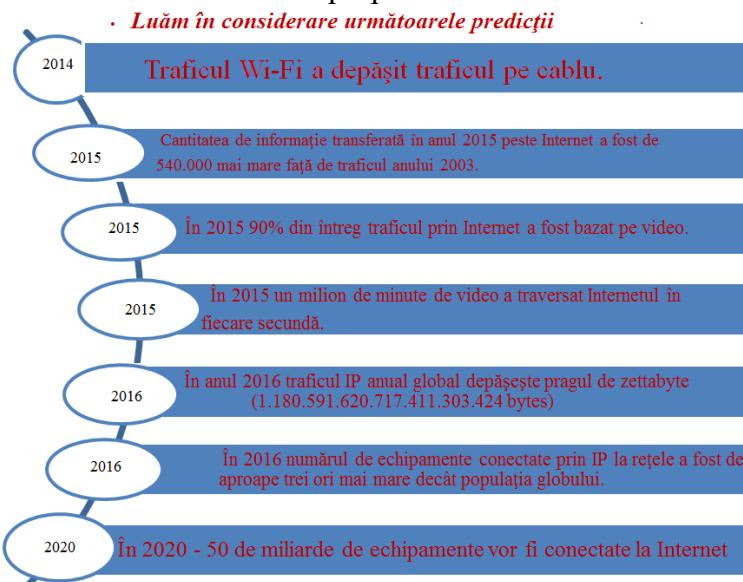


Figura 1.53 Predictii

Deoarece noi tehnologii și dispozitive apar pe piață, întreprinderile și consumatorii trebuie să continuie să se adapteze la acest mediu în continuă schimbare. Rolul rețelei este transformat pentru a permite conexiunile dintre oameni, dispozitive și informații. Există mai multe noi tendințe de rețea care vor afecta organizațiile și consumatorii. Unele dintre tendințele de top includ:

- *Orice dispozitiv, orice conținut, în orice mod.*
- *Colaborarea online.*
- *Video.*
- *Cloud computing.*

Aceste tendințe sunt interconectate și vor continua să se dezvolte bazându-se una pe alta în următorii ani. Următoarele subiecte vor acoperi aceste tendințe mai detaliat.

De rețineut că noile tendințe sunt vizate și proiectate în fiecare zi. Cum se va schimba Internetul în următorii 10 ani ? 20 de ani ?

1.14 Conceptul – integrează propriul dispozitiv (Bring Your Own Device -BYOD)

Conceptul de orice dispozitiv, orice conținut, în orice mod este o tendință globală importantă care necesită schimbări semnificative în modul în care dispozitivele sunt utilizate. Această tendință este cunoscută sub numele de Bring Your Own Device (BYOD).

BYOD este despre utilizatorii finali care trebuie să aibă libertatea de a utiliza instrumentele personale pentru a accesa informații și pentru a comunica prin intermediul unei rețele a campusului sau organizațiilor. O dată cu creșterea dispozitivelor de consum și scăderea costului, se așteaptă ca angajații și studenții să aibă unele dintre cele mai avansate instrumente de rețea și computing pentru utilizare personală. Aceste instrumente personale includ laptopuri, notebookuri, tablete, smartphoneuri și e-readers. Acestea pot fi dispozitive achiziționate de către companie sau școală, de către individ, sau ambele.

BYOD înseamnă orice dispozitiv, cu orice proprietar, utilizat oriunde. De exemplu, în trecut, un student care necesită acces la rețea campusului sau la Internet trebuia să utilizeze unul dintre calculatoarele de la școală. Aceste dispozitive erau limitate în mod normal și erau văzute ca instrumente de lucru efectuat numai în sala de clasă sau în bibliotecă. Conectivitatea extinsă la accesul mobil sau de la distanță la rețea campusului oferă studenților o flexibilitate extraordinară și mai multe oportunități de învățare.

BYOD este o tendință cu influență care a atins sau va atinge fiecare organizație IT.

Colaborare online

Indivizii vor să se conecteze la rețea, nu numai pentru acces la aplicațiile de date, dar și pentru colaborarea dintre ei. Colaborarea este definită ca “actul de lucru împreună cu altul sau alții la un proiect comun.”

Pentru afaceri, colaborarea este o prioritate critică și strategică. Pentru a rămâne competitive, organizațiile trebuie să răspundă la trei întrebări elementare de colaborare :

- *Cum aduc pe toată lumea pe aceeași lungime de undă ?*
- *Cu un buget și personal scăzut, cum vor pune în balanță resursele pentru a fi în mai multe locuri în același timp ?*
- *Cum pot ele să mențină relații față-în-față cu o rețea în creștere de colegi, clienți, parteneri și amici într-un mediu care este dependent 24 de ore de conectivitate ?*

Colaborarea este de asemenea o prioritate în educație. Studenții au nevoie să colaboreze unii cu alții pentru a învăța, pentru a dezvolta abilități de echipă folosite în munca în echipă și pentru a lucra împreună în proiecte bazate pe echipă.

Un mod de răspuns la aceste întrebări și de îndeplinire a acestor cereri din mediul de astăzi este prin intermediul instrumentelor de colaborare online. În mediile de lucru tradiționale, precum și în mediile BYOD, indivizii au la dispoziție servicii de conferințe audio și video în eforturile lor de colaborare.

Abilitatea de colaborare online schimbă procesele de afacere. Noi și extinse instrumente de colaborare permit indivizilor să colaboreze într-un mod rapid și ușor, indiferent de locația fizică. Organizațiile au mai multă flexibilitate în modul în care sunt organizate. Indivizii nu mai sunt restrictionați de locațiile fizice. Conoștința de specialitate este mai ușor de accesat decât înainte. Extinderile de colaborare permit organizațiilor să-și îmbunătățească colectarea de informații, inovația și productivitate. Fig. pune în evidență unele dintre beneficiile colaborării online.

Instrumentele de colaborare oferă angajaților, profesorilor, clienților și partenerilor un mod de conectare instant, de interacțiune și de desfășurare a activității, prin intermediul oricărui canal de comunicare preferat, precum și un mod de realizare a obiectivelor.

1.15 Comunicație Video

O altă tendință din rețelistică, critică pentru efortul de comunicare și colaborare, este transmisia video. Transmisia video este utilizată pentru comunicații, colaborare și divertisment.

Apelurile video au devenit din ce în ce mai populare, facilitând comunicațiile ca parte a rețelei umane. Apelurile video pot fi efectuate din orice loc cu o conexiune la Internet, inclusiv de acasă sau de la muncă.

Apelurile video și conferințele video se dovedesc a fi foarte puternice pentru procesele de vânzări și pentru efectuarea de afaceri. Video este un instrument util pentru efectuarea de afaceri de la distanță, atât local, cât și global. Astăzi, întreprinderile utilizează video pentru a transforma modul în care se fac afacerile. Video ajută întreprinderile să creeze un avantaj competitiv, oferă costuri scăzute și reduc impactul mediului prin reducerea nevoii de călătorie. Fig. arată tendința video în comunicație.

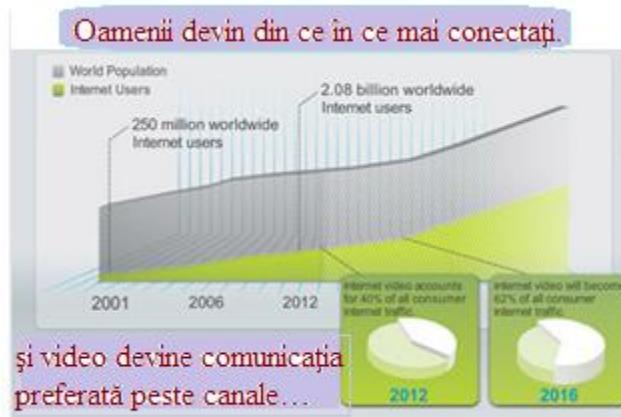


Figura 1.54 Evoluția traficului.

Atât consumatorii, cât și întreprinderile conduc această schimbare. Video devine o cerință cheie pentru colaborarea eficientă pentru organizații ce se extind dincolo de granițele geografice și culturale. Utilizatorii video cer acum abilitatea de a vedea orice conținut, de pe orice dispozitiv, de oriunde.

Afacerile, de asemenea, recunosc rolul video în rețeaua umană. Creșterea mediului de comunicare și noile utilizări ale acestuia conduc la nevoie de integrare audio și video în multe forme de comunicare. Conferința audio va coexista cu cea video. Instrumentele de colaborare proiectate pentru a lega angajații distribuiți vor integra video-desktop pentru a aduce echipele "mai aproape".

Există multe "drivere" și beneficii pentru includerea unei strategii de utilizare video. Fiecare organizație este unică. Mixul exact și natura "driverelor" de adaptare video vor varia de la organizație la organizație și în funcție de activitatea de afacere. Marketingul, de exemplu, s-ar putea axa pe globalizare și pe schimbarea rapidă a "gusturilor" consumatorilor; pe când, "Chief Information Officer (CIO)" s-ar putea axa pe reducerea costurilor prin reducerea costurilor de călătorie pentru angajații ce necesită să se întâlnească față în față. Fig. de mai jos listeză unele dintre "driverele" pentru organizații utilizate în dezvoltarea și implementarea unei strategii de soluție video.

Drivers for implementing a video strategy:

- **A global workforce and need for real-time collaboration** - Create collaborative teams that span corporate and national boundaries, and geographies.
- **Reducing costs and green IT**- Avoiding travel reduces both cost and carbon emissions.
- **New opportunities for IP convergence**- Converging video applications, such as high-definition video collaboration, video surveillance systems, and video advertising signage onto a single IP network.
- **Media explosion**- Plummeting cost of video cameras and a new generation of high-quality, low-cost devices have turned users into would-be movie producers.
- **Social networking** - The social networking phenomenon can be as effective in business as it is in a social setting. For example, employees are increasingly filming short videos to share best practices with colleagues, and to brief peers about projects and initiatives.
- **Demands for universal media access** - Users are demanding to be able to access rich-media applications wherever they are, and on any device. Participation in video conferencing, viewing the latest executive communications, and collaborating with co-workers are applications that will need to be accessible to employees, regardless of their work location.

55

O alta tendință în video este video-la-cerere și streaming live video. Furnizarea video peste rețea permite o nouă vizualizare de filme și programe de televiziune atunci când dorim și de unde dorim.

1.16 Cloud Computing

Cloud computing reprezintă utilizarea de resurse (hardware și software) ce ne sunt livrate sub forma de servicii peste rețea. O companie utilizează hardware și software în cloud și plătește o taxă pentru serviciu.

Computerele locale nu mai trebuie să facă toată “munca grea” atunci când vine vorba de rularea de aplicații de rețea. Rețeaua de computere ce alcătuiește cloudul se ocupă de acest lucru în locul lor. Cerințele hardware și software ale utilizatorului scad. Computerul utilizatorului trebuie să interacționeze cu cloudul prin software, software ce ar putea fi un browser web, iar rețeaua cloudului “are grijă” de restul.

Cloud computing este o altă tendință ce schimbă modul în care accesăm și stocăm datele. Cloud computing cuprinde orice serviciu pe bază de abonament sau orice serviciu pay-per-use, în timp real, peste Internet. Cloud computing permite stocarea de fișiere personale, chiar și o copie de rezervă a întregului hard disk pe serverele din Internet. Aplicațiile precum procesarea word sau editarea foto pot fi accesate cu ajutorul cloudului.

Pentru afaceri, cloud computing extinde capacitatea IT fără cerințele unei investiții într-o nouă infrastructură, fără educarea noului personal sau fără cerința unei noi licențe software. Aceste servicii sunt disponibile la cerere și sunt livrate în mod economic pe orice dispozitiv, din orice colț al lumii, fără a compromite securitatea sau funcționarea acestuia.

Termenul de “cloud computing” se referă de fapt la computing bazat pe web. Online banking, magazine de vânzare online și descărcările online de muzică sunt exemple de cloud computing. Aplicațiile cloud sunt, în mod normal, livrate utilizatorului prin intermediul unui browser web. Utilizatorii nu necesită să aibă software instalat pe dispozitivele lor finale. Acest lucru permite multor tipuri de dispozitive să se conecteze la cloud.

Cloud computing oferă următoarele beneficii potențiale:

- *Flexibilitate organizațională* - Utilizatorii pot accesa informațiile în orice moment din orice loc cu ajutorul unui browser web.
- *Agilitate și desfășurare rapidă* – Departamentul IT se poate axa pe livrarea de instrumente pentru a extrage, analiza și partaja informațiile și cunoștințele din bazele de date, fișiere, și oameni.

45

- *Costul redus al infrastructurii* – Tehnologia este mutată la un furnizor de cloud, eliminând costul hardware și al aplicațiilor.
- *Reorientarea resurselor IT* – Economiile de cost ale aplicațiilor și hardware pot fi utilizate în altă parte.
- *Crearea de noi modele de afaceri* – Aplicațiile și resursele sunt ușor accesibile, deci companiile pot reacționa rapid la nevoile consumatorului. Acest lucru ajută la setarea unor strategii de promovare a inovației în timpul intrării potențiale pe noi piete.

Există patru tipuri importante de cloud, conform figurii.

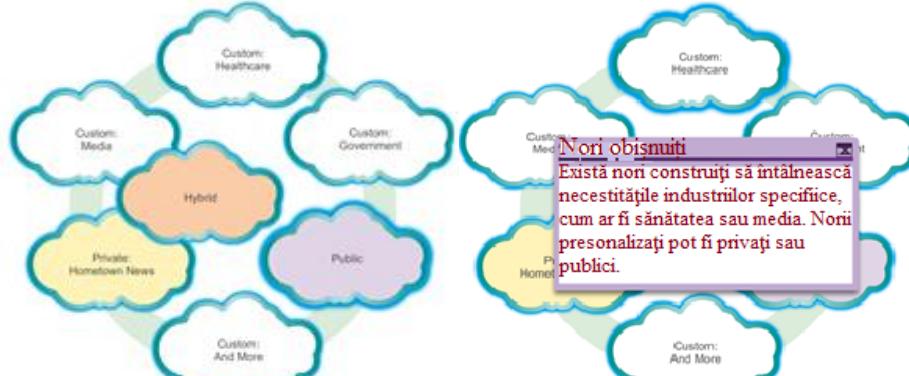


Figura 1.56 Tipuri de nori

Nori obișnuiti
Există non construi să întâlnească necesitățile industriilor specifice, cum ar fi sănătatea sau media. Norii personalizați pot fi privați sau publici.

Figura 1.56 a) Nori personalizați

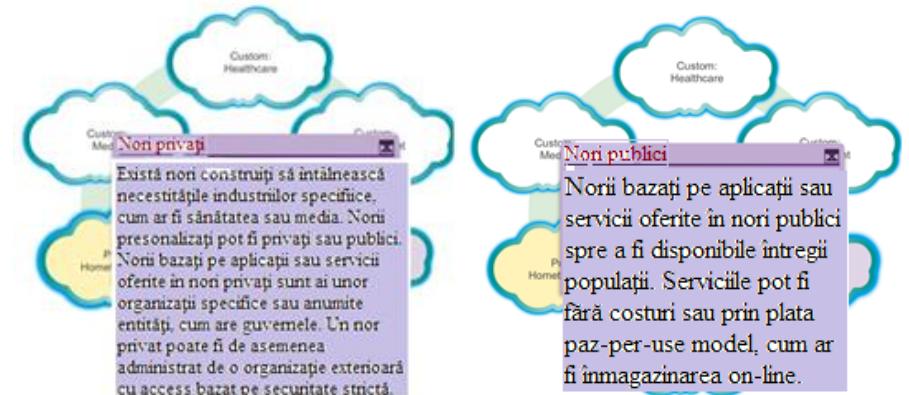


Figura 1.56 b) Nori privați.

Nori publici
Norii bazăți pe aplicații sau servicii oferite în nori publici sunt disponibile într-o populație largă. Serviciile pot fi fără costuri sau prin plată pe bază de utilizare.

Figura 1.56 c) Nori publici

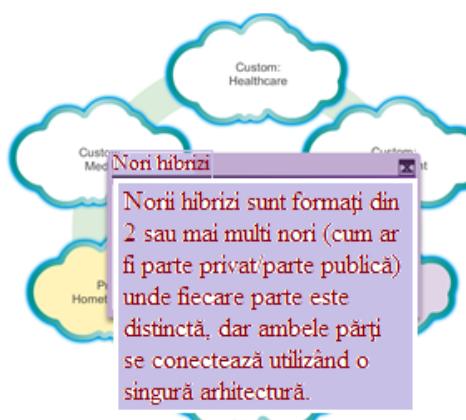


Figura 1.56 d) Nori hibrizi

Fig. 1.70.

Cloud computing este posibil datorită centrelor de date. Un centru de date este o facilitate utilizată pentru stocarea sistemelor de calculatoare și componentele asociate lor, inclusiv:

- *Conexiunile de comunicare de date redundante.*
- *Servere virtuale de mare viteză (referite ca "server farms" sau "server clusters").*
- *Sisteme de stocare redundante (în mod normal utilizează tehnologie SAN).*
- *Surse de alimentare redundante sau de rezervă.*
- *Controale de mediu (exemplu: aer condiționat, stingător de incendiu etc.).*
- *Dispozitive de securitate.*

Un centru de date poate ocupa o încăpere dintr-o clădire, mai multe etaje dintr-o clădire, sau chiar o întreagă clădire. Centrele de date moderne se folosesc de cloud computing și de virtualizare pentru a gestiona eficient tranzacțiile mari de date. Virtualizarea este crearea unei versiuni virtuale a unui lucru, cum ar fi o platformă hardware, sistem de operare, dispozitiv de stocare sau resurse de rețea. În timp ce un computer fizic este un dispozitiv real distinct, o mașină virtuală constă într-un set de fișiere și programe ce rulează pe un sistem fizic real. Spre deosebire de multitasking, ce implică rularea mai multor programe pe același OS, virtualizarea rulează mai multe sisteme de operare (OS) diferite în paralel, pe un singur CPU. Acest lucru reduce drastic costurile și cheltuielile generale administrative.

Centrele de date sunt, în mod normal, foarte costisitor de construit și întreținut. Din acest motiv, organizațiile mari utilizează centre de date construite în privat pentru a-și stoca datele și pentru a furniza servicii utilizatorilor. De exemplu, un spital mare ar putea deține un centru de date separat, în care înregistrările pacienților sunt reținute electronic. Organizațiile mai mici, care nu își permit să întrețină propriul centru de date privat, pot reduce costurile totale de proprietate prin închirierea unui server și stocarea serviciilor de la o organizație mai mare de centru de date în cloud.

1.17 Tehnologii pentru Rețele de casă

Powerline networking este o tendință în curs de dezvoltare pentru rețea de acasă ce utilizează cablurile electrice existente pentru a conecta dispozitive, conform imaginii de mai jos. Conceptul de "fără fire noi" înseamnă abilitatea de conectare a unui dispozitiv la rețea oriunde există o priză electrică. Utilizând aceleasi cabluri ce oferă electricitate, powerline networking trimite informații prin trimiterea de date pe anumite frecvențe similare cu cele utilizate de tehnologia DSL.

Prin utilizarea unui *"HomePlug standard powerline adapter"*, dispozitivele se pot conecta la LAN de oriunde există o priză electrică. Powerline networking este utilizat în mod special atunci când puncte de acces wireless nu pot fi folosite sau nu ajung la toate dispozitivele din casă. Powerline networking nu este conceput pentru fi un înlocuitor pentru cablarea dedicată pentru rețelele de date. Însă, este o alternativă atunci când cablurile de rețea de date sau comunicațiile wireless nu sunt o opțiune viabilă.

Conecțarea la Internet este vitală în tehnologia inteligentă de acasă. DSL și cablul sunt tehnologiile utilizate, de obicei, pentru a conecta casele și afacerile mici la Internet. Însă, wireless poate fi o altă opțiune în multe domenii.

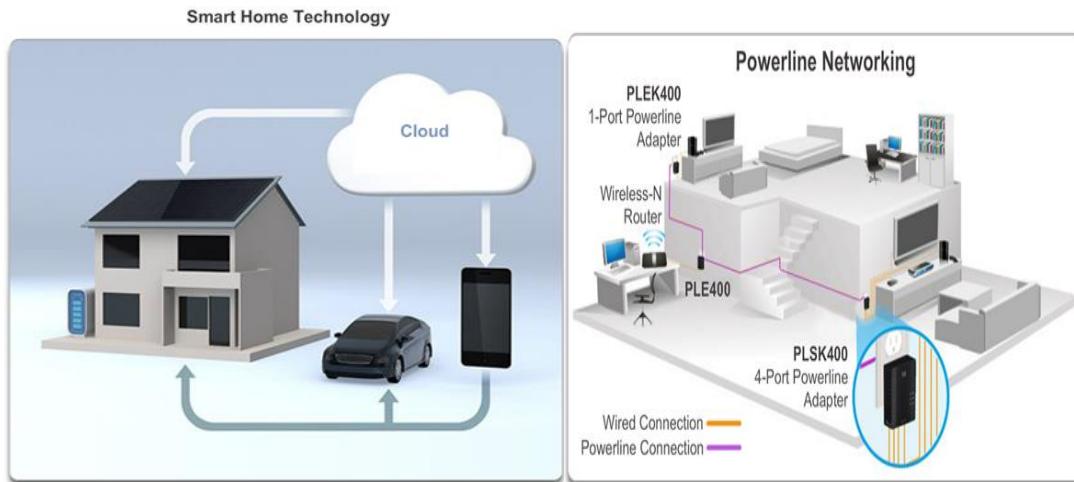


Fig. 1.71.

Fig. 1.72.

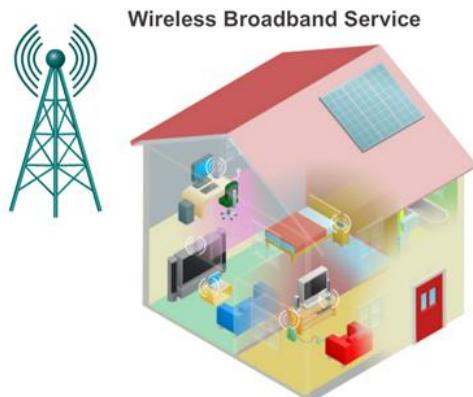


Fig. 1.73.

1.18 Wireless Internet Service Provider (WISP)

Wireless Internet Service Provider (WISP) este un ISP ce conectează abonații la un punct de acces desemnat sau la un hot spot ce utilizează tehnologii wireless similare cu cele utilizate în WLANs (wireless local area networks). WISPs sunt întâlnite mai mult în mediile rurale unde serviciile DSL sau prin cablu nu sunt disponibile.

Deși un turn de transmisie separat ar putea fi instalat pentru antenă, de obicei, este comun faptul ca antena să fie atașată la o structură mai înaltă existentă cum ar fi un turn de apă sau un turn radio. O antenă este instalată pe acoperișul abonatului în raza de acțiune a transmisiilor WISP. Unitatea de acces a abonatului este conectată la o rețea cablată din interiorul casei. Din punctul de vedere a utilizatorului, setările nu sunt cu mult mai diferite decât a serviciului DSL sau prin cablu. Principala diferență este aceea că realizarea conexiunii de la casă la ISP este wireless și nu prin cablu fizic.

Wireless Broadband Service - O altă soluție wireless pentru domiciliu și afaceri mici este wireless de bandă largă. Această soluție utilizează aceeași tehnologie celulară folosită pentru a accesa Internetul cu un telefon mobil sau cu o tabletă. O antenă este instalată în afara casei oferind fie conexiune wireless, fie cablată pentru dispozitivele din casă. În multe arii, wireless de bandă largă este în concurență directă cu serviciile DSL și prin cablu.

1.19 Securitatea rețelei

Securitatea rețelei este o parte integrală din rețelelistică, indiferent dacă rețeaua este limitată la un mediu de acasă cu o singură conexiune la Internet sau dacă este mare, ca în cazul unei corporații cu mii de utilizatori. Implementarea securității rețelei trebuie să țină cont de mediul înconjurător, cât și de instrumentele și cerințele rețelei. Trebuie să fie capabilă să securizeze datele, în timp ce menține calitatea serviciului așteptată în rețea.

Securizarea unei rețele implică protocoale, tehnologii, dispozitive, instrumente și tehnici pentru a securiza datele și pentru a combate amenințările. Multe amenințări externe de securitate a rețelei de astăzi sunt răspândite în Internet. Cele mai comune amenințări externe din rețea includ:

- *Virusi, viermi și cai Troieni* – software rău intenționat și cod aleator rulat pe un dispozitiv al utilizatorului.
- *Spyware and adware* – software instalat pe un dispozitiv al utilizatorului care colectează în secret informații despre utilizator.
- *Atacuri Zero-day, numite și atacuri zero-hour* – un atac ce are loc în prima zi în care o vulnerabilitate devine cunoscută.
- *Atacuri ale hackerilor* – un atac al unei persoane cu cunoștințe despre un dispozitiv sau resurse de rețea.
- *Atacuri Denial of service* – atacuri destinate să încetinească sau să “prăbușească” aplicații și procese pe un dispozitiv de rețea.
- *Interceptarea datelor și furtul de date* – un atac de capturare a informațiilor private de la o rețea a unei organizații.
- *Furtul de identitate* – un atac ce fură credențialele de logare ale unui utilizator pentru a accesa datele private.

Este de asemenea, la fel de importantă considerarea amenințărilor interne. Au existat multe studii ce arată că multe breșe de securitate au loc datorită utilizatorilor interni ai unei rețele. Acest lucru poate fi atribuit la dispozitivele furate sau pierdute, folosirea greșită, accidentală, de către angajați și, în mediul de afacere, chiar angajații rău intenționați. O dată cu dezvoltarea strategiilor BYOD, datele corporative sunt mult mai vulnerabile. Mai mult, atunci când dezvoltăm o politică de securitate, este important să ne adresăm atât amenințărilor interne de securitate, cât și celor externe.

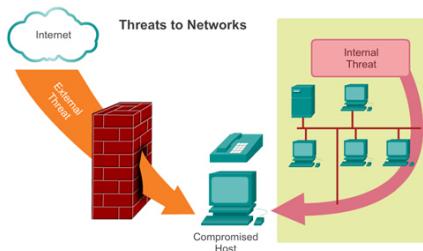


Fig. 1.73.

Nu există o soluție unică pentru protejarea rețelei împotriva varietăților de amenințări existente. Din acest motiv, securitatea ar trebui să fie implementată pe mai multe nivele, folosind mai mult decât o soluție. Dacă o componentă de securitate nu identifică și protejează rețeaua, altele vor fi implementate pentru a face acest lucru.

O implementare de securitate a rețelei de domiciliu este mai degrabă de bază. Este implementată în general pe dispozitivele host de conectare, precum și în punctul de conexiune la Internet și se poate bază chiar și pe serviciile contractate de la ISP.

Implementarea securității de rețea pentru o rețea corporativă constă în mod normal din mai multe componente construite în rețea pentru a monitoriza și filtra traficul. În mod ideal, toate componentele funcționează împreună, acest lucru minimizând întreținerea și imbunătățind securitatea.

Componentele de securitate ale rețelei pentru o locuință sau pentru un birou mic ar trebui să includă cel puțin:

- *Antivirus și antispyware* – pentru a proteja dispozitivele utilizatorului împotriva softwareului rău intenționat.
- *Firewall filtering* – pentru a bloca accesul neautorizat la rețea. Acest lucru ar putea include un sistem firewall bazat pe gazdă care este implementat pentru a preveni accesul neautorizat la dispozitivul gazdei sau un serviciu de filtrare pe routerul locuinței pentru a preveni accesul neautorizat din exterior la rețea.

În plus față de cele menționate mai sus, rețelele mai mari și rețelele corporative adesea au alte cerințe de securitate:

- *Sisteme firewall dedicate* – pentru a oferi capacitați firewall mai avansate care pot filtra o cantitate mare de trafic cu o granularitate mai mare.
- *Liste de control al accesului (ACL)* – pentru a favoriza accesul filtrat și pentru redirecționarea traficului.
- *Sisteme de prevenție a intruziunii (IPS)* – pentru a identifica amenințări ce se răspândesc rapid, cum ar fi atacuri zero-day sau zero-hour.
- *Rețele virtuale private (VPN)* – pentru a furniza acces securizat utilizatorilor de la distanță.

Cerințele de securitate de rețea trebuie să țină cont de mediul rețelei, precum și de aplicațiile variate și de cerințele computaționale. Ambele medii, de afacere și mediul de locuințe, trebuie să fie capabile să-și securizeze datele, în timp ce mențin calitatea așteptată a serviciului în funcție de tehnologia utilizată. În plus, soluția de securitate implementată trebuie să fie adaptabilă la creștere și la tendințele de schimbare ale rețelei.

Studiul amenințărilor de securitate a rețelei și a tehnicielor de reducere începe cu o înțelegere precisă a bazelor infrastructurii de rutare și switching, utilizate pentru a organiza serviciile de rețea.



Fig. 1.74.

1.20 Arhitecturi de Rețea

Rolul rețelei s-a schimbat de la o rețea bazată pe date, la un sistem ce permite conexiuni ale oamenilor, dispozitivelor și informațiilor într-un mediu de rețea convergentă. Pentru ca rețelele să funcționeze eficient și să crească în acest tip de mediu, rețeaua trebuie să fie construită în funcție de o arhitectură de rețea standard.

Arhitectura de rețea se referă la dispozitivele, conexiunile și produsele integrate pentru a suporta tehnologiile și aplicațiile necesare. O tehnologie de rețea bine-planificată ajută la asigurarea conexiunii oricărui dispozitiv, în orice combinație de rețele. O dată cu asigurarea conectivității crește și eficiența costului prin integrarea de securitate și management al rețelei și se imbunătățesc procesele de afacere. La bază tuturor arhitecturilor de rețea, și de fapt, la bază Internetului, sunt routere și switchuri. Routerele și switchurile transportă comunicațiile de date, voce și video, permit accesul wireless și asigură securitatea.

Construirea de rețele care suportă nevoile utilizatorilor de astăzi și nevoile și tendințele din viitor începe cu o înțelegere precisă a bazelor infrastructurii de rutare și comutare (switching). După ce o infrastructură de rețea de rutare și switching de bază este construită, indivizii și afacerile mici și organizațiile își pot extinde rețeaua în timp, adăugând caracteristici și funcționalități într-o soluție integrată.



Fig. 1.76.

În timp ce utilizarea acestor rețele extinse, integrate crește, crește și nevoia de pregătire pentru indivizii care implementează și gestionează soluțiile de rețea. Această pregătire trebuie să înceapă cu fundamentele rutării și comutării.

1.21 Concluzii Capitolul 1

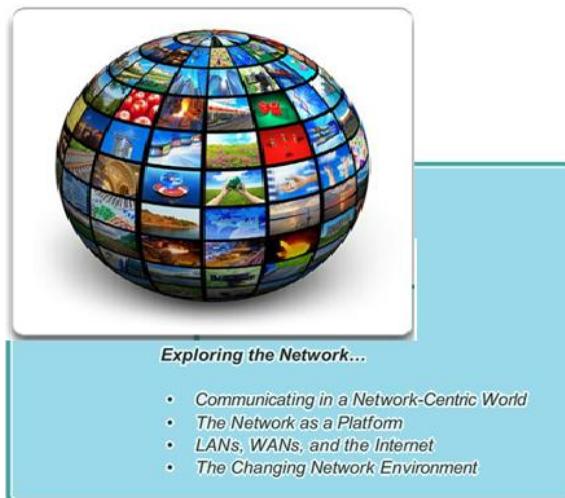


Fig. 1.77.

Rețelele și Internetul au schimbat modul în care oamenii comunică, învață, lucrează și se distrazează.

Rețelele sunt de toate dimensiunile. Pot varia de la simple rețele alcătuite din două computere, la rețele ce conectează milioane de dispozitive.

Internetul este cea mai mare rețea existentă. De fapt, termenul de Internet înseamnă “rețea de rețele”. Internetul furnizează serviciile ce permit să ne conectăm și să comunicăm cu familiile, prietenii, la locul de munca, și să comunicăm interesele noastre.

Infrastructura de rețea este platforma ce suportă rețeaua. Oferă canalul stabil și de încredere prin intermediul căruia comunicarea are loc. Este alcătuită din componente de rețea, cum ar fi dispozitivele finale, dispozitivele intermediare și mediul de rețea.

Rețelele trebuie să fie de încredere. Acest lucru înseamnă că rețelele trebuie să fie tolerate la erori, scalabile, să ofere servicii de calitate și să asigure securitatea informațiilor și resurselor din rețea. Securitatea rețelei este o parte integrată a rețelisticiei, indiferent dacă o rețea este limitată la un mediu restrâns (domiciliu) cu o singură conexiune la Internet sau dacă este o rețea mare ce cuprinde mii de utilizatori. Nici-o soluție nu poate proteja rețeaua de varietatea de amenințări existente. Din acest motiv, securitatea ar trebui să fie implementată pe mai multe nivele, folosind mai mult decât o singură soluție de securitate.

Infrastructura de rețea poate varia mult în funcție de dimensiune, numărul de utilizatori și numărul și tipurile de servicii suportate. Infrastructura de rețea trebuie să crească și să se adapteze pentru a suporta modul în care rețeaua este utilizată. Platforma de rutare și switching este bază oricărei infrastructuri de rețea.

Intelligent Networks Are Bringing the World Together

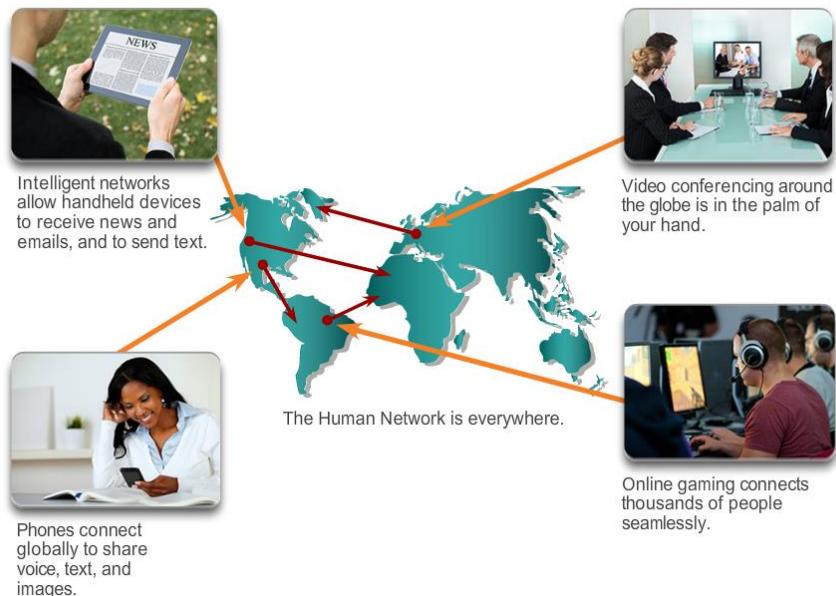


Fig. 1.78.

CAPITOLUL 2. CONFIGURAREA IOS PENTRU REȚEA

Introducere

Rețele de domiciliu interconectează o multitudine de dispozitive finale cum ar fi: calculatoare, laptopuri, telefoane, smart TVuri, media playere de rețea conform DLNA, precum Xbox 360 sau PlayStation 3, și multe altele.

Toate aceste dispozitive finale sunt conectate de obicei la un router local de casă. Routerele de casă prezintă, de fapt, 4 dispozitive într-unul singur:

- *Router* – Trimit mai departe pachetele de date și receptionează pachetele de la Internet.
- *Switch* – Conectează dispozitivele folosind cabluri de rețea.
- *Punct de acces wireless* – Constă dintr-un transmisiștor radio capabil de a conecta dispozitivele, în mod fără fir.
- *Firewall* – Securizează traficul careiese și îl restricționează pe cel care intră.

În rețelele mai mari, de afaceri, ce conțin mult mai multe dispozitive și cu un trafic mult mai ridicat, aceste dispozitive sunt de obicei independente, de sine stătătoare, oferind servicii dedicate. Dispozitivele finale, precum calculatoarele și laptopurile, sunt conectate la switchuri folosind cabluri de rețea. Pentru a trimite pachete dincolo de rețea locală, switchurile sunt conectate la routere. Alte dispozitive din rețea includ: puncte de acces wireless și dispozitive de securitate dedicate, precum firewallurile.

Fiecare dispozitiv diferă prin componente hardware, utilizare și capacitate. Dar în toate cazurile, sistemul de operare este cel care permite hardwareului să funcționeze.

Sistemele de operare sunt folosite de către toate dispozitivele finale, ale utilizatorilor și cele de rețea, conectate la Internet. Dispozitivele finale ale utilizatorilor includ dispozitive precum telefoanele mobile, tabletele, calculatoarele, și laptopurile. Dispozitivele de rețea, sau dispozitivele intermediare, sunt dispozitive folosite pentru a transporta datele în rețele, și includ switchuri, routere, puncte de acces wireless și firewalluri. Sistemul de operare de pe un dispozitiv de rețea este cunoscut sub numele de sistem de operare de rețea - Internetwork Operating System (IOS).

Acest capitol face referire la o topologie simplă, ce constă din 2 switchuri și 2 calculatoare, pentru a prezenta cum se utilizează Cisco IOS.

După parcurgerea acestui capitol se deprind următoarele capabilități:

- *Explicarea scopului IOS.*
- *Explicarea modului de accesare și navigare cu IOS pentru a configura dispozitivele de rețea.*
- *Descrierea structurii comenzi din IOS.*
- *Configarea numelui unui dispozitiv IOS folosind CLI.*
- *Utilizarea de comenzi IOS pentru a limita accesul la configurația dispozitivelor.*
- *Utilizarea comenzi IOS pentru a salva configurația curentă.*
- *Explicarea modului de comunicare al dispozitivelor într-o rețea.*
- *Configarea adreselor IP a unui dispozitiv gazdă.*
- *Verificarea conexivității dintre două dispozitive finale.*

2.1 IOS Bootcamp

Toate dispozitivele finale și dispozitivele de rețea conectate la Internet necesită un sistem de operare (SO) pentru a le ajuta să-și execute funcțiile.

În momentul în care un calculator este pornit, acesta încarcă SO, de obicei de pe o unitate de disc, în memoria RAM. Portiunea din codul SO care interacționează direct cu partea hardware a calculatorului este cunoscută sub numele de kernel. Portiunea care interacționează cu aplicațiile și cu utilizatorul este cunoscută sub numele de shell. Utilizatorul poate interacționa cu shellul folosind fie interfață pentru comenzi (CLI), fie interfață grafică (GUI).

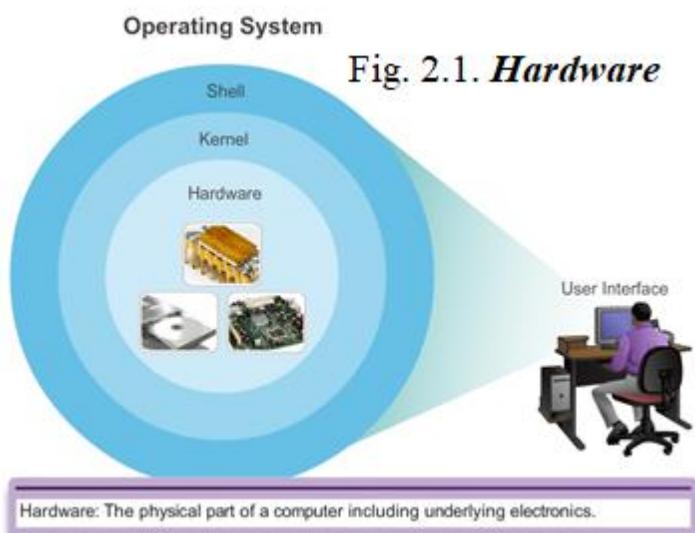
Când este folosită CLI, utilizatorul interacționează direct cu sistemul intr-un mediu bazat pe text, prin introducerea comenzilor de la tastatură la prompter. Sistemul execută comanda și oferă de obicei rezultate textuale. Interfața GUI permite utilizatorului să interacționeze cu sistemul intr-un mediu care folosește imagini și text. Acțiunile sunt executate prin interacțiunea cu imaginile de pe ecran. GUI este mai ușor de utilizat și necesită mai puține cunoștințe legate de structura comenzii pentru a utiliza sistemul. Din acest motiv, mulți utilizatori se bazează pe mediile GUI. Majoritatea sistemelor de operare oferă atât GUI, cât și CLI.

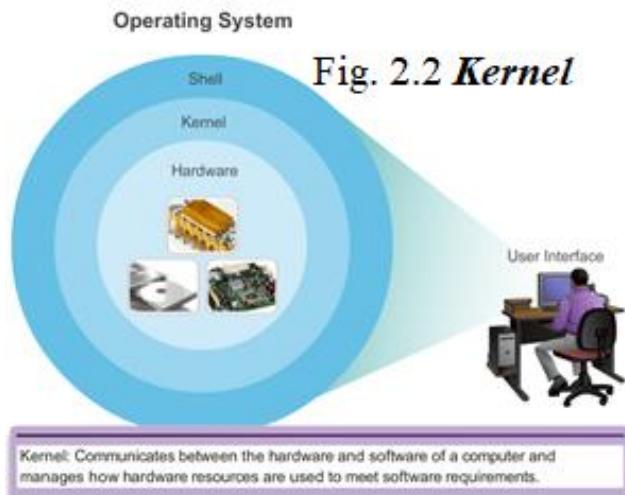
Majoritatea sistemelor de operare de pe dispozitivele finale sunt accesate folosind GUI, inclusiv MS Windows, MAC OS X, Linux, Apple iOS, Android și altele.

Sistemul de operare de pe routerele de casă este numit de obicei firmware. Cea mai utilizată metodă pentru a configura astfel de router este folosirea unui browser web pentru a accesa o interfață GUI. Majoritatea routerelor de casă permit actualizarea firmwareului pe măsură ce vulnerabilități în securitate sunt descoperite.

Dispozitivele de rețea din infrastructură folosesc un sistem de operare de rețea. Sistemul de operare de rețea folosit de dispozitivele Cisco se numește Cisco Internetwork Operating System (IOS). Cisco IOS este un termen general pentru colecția de sisteme de operare de rețea folosite pe dispozitivele de rețea Cisco. Cisco IOS este folosit de majoritatea dispozitivelor Cisco indiferent de tipul sau dimensiunea lor. Cea mai utilizată metodă de accesare a acestor dispozitive este prin CLI.

Acest capitol prezintă o topologie bazată pe switchuri a unei companii mici. Topologia conține 2 switchuri și 2 calculatoare și va fi folosită pentru a demonstra utilizarea Cisco IOS folosind CLI.





Sistemele de operare de rețea sunt asemănătoare cu sistemele de operare de pe calculatoare. Un sistem de operare execută un număr de funcții tehnice “behind the scenes” care permit utilizatorului să:

- Utilizeze un mouse.
- Vizualizeze rezultate pe monitor.
- Introducă de comenzi text.
- Selecteze opțiuni dintr-o fereastră de dialog.

Funcțiile “behind the scenes” ale switchurilor și routerelor sunt similare. IOSul de pe un switch sau un router oferă o interfață tehnicianului de rețea. Tehnicianul poate introduce comenzi pentru a configura/programa dispozitivul sau să execute anumite funcții. Detaliile operaționale ale IOS variază pe dispozitive, în funcție de scopul dispozitivului și de caracteristicile suportate.

Cisco IOS este un termen care cuprinde sisteme de operare diferite care rulează pe diferite dispozitive de rețea. Există multe variații distincte ale Cisco IOS:

- IOS pentru switchuri, routere, și alte dispozitive de rețea Cisco.
- Versiuni numerotate ale IOS pentru un anumit dispozitiv de rețea Cisco.
- Seturi de caracteristici IOS care oferă pachete diferite de servicii.

La fel cum un PC poate rula Microsoft Windows 8 și un MacBook poate rula OS X, un dispozitiv de rețea Cisco rulează o anumită versiune de Cisco IOS. Versiunea de IOS este dependență de tipul dispozitivului folosit și de caracteristicile necesare. Cu toate că majoritatea

dispozitivelor au un IOS și un set de caracterisitici implicate, este posibilă actualizarea versiunii IOS și setului de caracterisitici, în vederea obținerii unor caracteristici adiționale.

Fișierul ce conține IOS are câțiva megaocteți și este stocat într-o zonă semi-permanentă de memorie numită **flash**. Fig. afișează un compact flash card. Memoria flash oferă spațiu de stocare nevolatil. Astfel, conținutul memoriei nu este pierdut în momentul în care dispozitivul nu mai este alimentat cu energie electrică. Cu toate acestea, conținutul poate fi schimbat sau suprascris dacă este nevoie. Acest lucru permite actualizarea versiunii IOSului sau adăugarea de noi caracteristici fără înlocuirea hardwaerului. În plus, memoria flash poate fi folosită pentru a stoca mai multe versiuni de IOS.

În multe dispozitive Cisco, IOSul este copiat în memoria RAM în momentul pornirii acestora. Apoi IOSul este rulat din RAM când dispozitivul funcționează. RAMul are multe funcții, inclusiv stocarea de date necesare dispozitivului pentru a susține operațiile rețelei. Rularea IOSului din RAM îmbunătățește performanța dispozitivului, dar RAM este considerată o memorie volatilă deoarece datele se pierd într-un ciclu de alimentare. Un ciclu de alimentare reprezintă oprirea și apoi pornirea dispozitivului, în mod intenționat sau accidental.

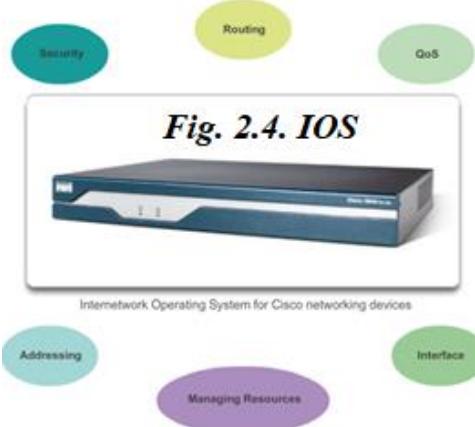
Cantitatea de memorie flash și RAM necesară unui IOS variază dramatic. Pentru planificarea și administrarea rețelei, este importantă determinarea cerințelor de flash și RAM pentru fiecare dispozitiv, inclusiv conFig.ările maxime de flash și RAM. Este posibil ca o versiune nouă de IOS să necesite mai multă memorie RAM și flash decât poate fi instalată pe anumite dispozitive.

IOSul de pe routerele și switchurile profesionale (ex. Cisco) efectuează funcții de care administratorii rețelelor depind pentru a face rețelele să funcționeze într-un anumit fel. Funcții majore efectuate de către routere și switchuri includ:

- *Asigurarea securității rețelei.*
- *Adresarea IP a interfețelor fizice și virtuale.*
- *ConFig.area interfețelor, în mod independent, pentru a optimiza conectivitatea mediului respectiv.*
- *Rutarea.*
- *Permiterea tehnologiilor QoS.*
- *Susținerea tehnologiilor pentru administrarea rețelei.*

Fiecare caracteristică sau serviciu are asociată o colecție de comenzi de conFig.re care permit unui tehnician de rețea să o implementeze.

Serviciile oferite de Cisco IOS sunt deobicei accesate folosind comenzi prin linia consolă sau prin comenzi directe - CLI.



2.2 Accesarea unui dispozitiv Cisco IOS

Există mai multe moduri de accesare a mediului CLI. Cele mai comune metode sunt:

- *Consola*
- *Telnet sau SSH*
- *Portul AUX*

Consola – Portul consolă este un port de administrare care oferă acces "out-of-band" la dispozitivele Cisco. Accesul out-of-band se referă la accesul printr-un canal dedicat administrării, folosit în scopul întreținerii dispozitivelor. Avantajul folosirii portului consolă este acela că dispozitivul este accesibil chiar dacă nici-un serviciu de rețea nu a fost config.t, ca în cazul configurării inițiale a dispozitivului de rețea. Când este efectuată config.rea inițială, un calculator care rulează un terminal este conectat la portul consolă al dispozitivului printr-un cablu special. Comenzi de config.re a switchului sau routerului sunt introduse prin calculatorul respectiv.

Portul consolă poate fi folosit și în momentul în care serviciile de rețea au cedat și accesul de la distanță la dispozitivul Cisco nu este posibil. Dacă acest lucru se întâmplă, o conexiune prin acest port permite determinarea stării dispozitivului. Implicit, consola transmite mesaje legate de inițializarea dispozitivului, de depanare și de erori. După ce tehnicienul de rețea s-a conectat la dispozitiv, el poate executa orice comandă de config.re necesară, folosind sesiunea consolă.

În cazul multor dispozitive cu IOS, accesul la consolă nu necesită nici-o formă de securitate, în mod preșă. Dar consola trebuie config.tă cu parole pentru a împiedica accesul neautorizat. În cazul în care parola este pierdută, există un set special de proceduri pentru a evita parola, accesând direct dispozitivul. De asemenea, dispozitivul ar trebui să se afle într-o cameră încuiată sau într-un rack pentru a împiedica accesul fizic neautorizat.



Telnet – Telnet este o metodă pentru inițierea de la distanță a unei sesiuni CLI pe un dispozitiv, printr-o interfață virtuală, printr-o rețea. Spre deosebire de conexiunea consolă, sesiunile Telnet necesită ca serviciile de rețea de pe dispozitiv să fie active. Dispozitivul trebuie să aibă măcar o interfață activă config.tă cu o adresă Internet, cum ar fi o adresă IPv4. Dispozitivele Cisco IOS includ un proces server Telnet care permite utilizatorilor să introducă comenzi de config.re printr-un client Telnet. În afară de acest proces server Telnet, dispozitivul Cisco IOS conține și un client Telnet. Aceasta permite administratorului de rețea să acceseze, din interfață CLI a dispozitivului, un alt dispozitiv care are un server proces Telnet.

SSH – Protocolul SSH oferă o autentificare de la distanță similară cu cea Telnet, dar folosește servicii de rețea mai sigure. SSH oferă o autentificare mai puternică decât Telnet și transportă datele sesiunii în mod criptat. Acest lucru păstrează IDul utilizatorului, parola și detaliile legate de sesiunea curentă într-un format confidенtial. Ca o bună practică, se cere utilizarea SSH în locul Telnetului.

Majoritatea versiunilor de Cisco IOS includ un server SSH. În unele dispozitive, acest serviciu este activat în mod implicit, iar în alte dispozitive se cere activarea manuală a serverului SSH. Dispozitivele cu IOS includ și un client SSH care poate fi folosit pentru a iniția sesiuni SSH cu alte dispozitive.

AUX – O metodă mai veche de inițiere a unei sesiuni CLI de la distanță, este prîntr-o conexiune telefonică dialup folosind un modem conectat la portul auxiliar (AUX) al routerului, care este evidențiat în figură. Asemănătoare conexiunii consolă, metoda AUX este, de asemenea, o conexiune out-of-band și nu necesită nici-un serviciu de rețea config.t sau disponibil pe dispozitiv. În cazul în care serviciile de rețea eșuează, este posibil ca un administrator să acceseze, de la distanță, switchul sau routerul prîntr-o linie telefonică.

Portul AUX poate fi folosit și local, precum portul consolă, cu o conexiune directă la un calculator ce rulează un program de emulare a terminalului. Cu toate acestea, portul consolă este preferat în locul portului AUX pentru depanare, deoarece afișează mesaje de pornire, de depanare și de eroare, în mod implicit.

Notă: Switchurile Cisco Catalyst nu au o conexiune auxiliară.



**Fig. 2.6.
Portul AUXILIAR**

Există multe programe de emulare a unui terminal, disponibile pentru conectarea la un dispozitiv de rețea fie prîntr-o conexiune serială la portul consola sau prîntr-o conexiune Telnet/SSH.

Câteva dintre acestea sunt:

- PuTTY



- Tera Term



*Fig. 2.8
Interfața TeraTerm*

- SecureCRT
- HyperTerminal
- OS X Terminal

Aceste programe permit mărirea productivității prin ajustarea dimensiunii ferestrei, schimbarea dimensiunii fontului, și schimbarea culorilor.

2.3 Navigarea prin IOS

După ce un tehnician de rețea se conectează la un dispozitiv, îl poate configura. Acesta trebuie să treacă prin diferite moduri ale IOSului. Modurile Cisco IOS sunt asemănătoare între switchuri și routere. CLI folosește o structură ierarhică pentru aceste moduri.

În ordine ierarhică, de la cel de bază până la cel mai specializat, modurile majore sunt:

- *Modul utilizator (User EXEC)*
- *Modul privilegiat (Privileged EXEC)*
- *Modul de configurație globală*
- *Alte moduri specifice de configurație, precum modul de configurație al interfețelor.*

Fiecare mod are un prompter distinct și este folosit pentru a realiza anumite sarcini cu un anumit set de comenzi care sunt disponibile doar în acel mod. De exemplu, modul de configurație globală permite configurația setărilor unui dispozitiv, setări care îl afectează în întregime, cum ar fi schimbarea numelui dispozitivului. Totuși, un alt mod este necesar pentru configurația setărilor de securitate pentru un anumit port al unui switch, spre exemplu. În acest caz, tehnicianul de rețea trebuie să acceseze modul de configurație al interfeței pentru acel port. Toate comenziile introduse în acel mod se aplică doar aceluui port.

Structura ierarhică poate fi configurația să ofere securitate. Diferite autentificări pot fi implementate pentru fiecare mod. Acest lucru controlează nivelul de acces pe care personalul rețelei îl poate avea.

Fig. afișează structura modurilor IOS cu prompterul și trăsăturile lor.

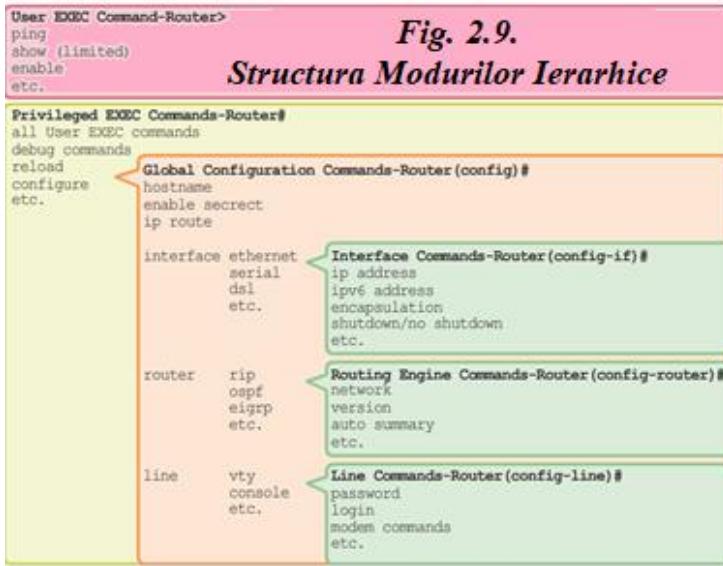


Fig. 2.9.
Structura Modurilor Ierarhice

Cele două moduri principale sunt modul utilizator "user EXEC" și modul privilegiat "privilege EXEC". Ca o măsură de securitate, softwareul Cisco IOS separă sesiunile EXEC în două niveluri de acces. După cum se poate vedea în figură, modul privilegiat EXEC are un nivel mai mare de autoritate în ceea ce privește lucrurile pe care le poate efectua utilizatorul asupra dispozitivului.

Modul User EXEC – Modul "user EXEC" are capabilități limitate, dar este folositor pentru operațiile de bază. Acest mod se află la bază structurii ierarhice. El este primul mod întâlnit în momentul accesării CLI a unui dispozitiv IOS.

Modul "user EXEC" permite un număr limitat de comenzi de monitorizare de bază. Acest mod este numit deobicei view-only - doar pentru vizualizare. Modul "user EXEC" nu permite executarea comenziilor care pot schimba configurația dispozitivului.

În mod implicit, accesarea modului "user-EXEC" de la consolă, nu necesită nici-o autentificare, dar, este o practică bună configurației în timpul configurației inițiale.

Modul "user EXEC" este identificat prin promptul CLI care se termină cu simbolul >. Aceasta este un exemplu care arată simbolul > din prompt:

Router>

Modul de config.re Privilegiat – Executarea comenziilor de config.re și de administrare necesită ca administratorul rețelei să folosească modul privilegiat EXEC sau un mod mai specific din ierarhie. Acest lucru înseamnă că utilizatorul trebuie să intre mai întâi în modul utilizator EXEC, și de acolo, să acceseze modul privilegiat EXEC.

Modul privilegiat EXEC poate fi identificat prin promptul care se termină cu simbolul #.

Router>enable

Rouer#

În mod implicit, modul privilegiat EXEC nu necesită o autentificare. Este o practică bună configurației autentificării.

Modul de config.re Globală – Modul de config.re globală și toate celelalte moduri de config.re mai specifice pot fi accesate doar din modul privilegiat EXEC.

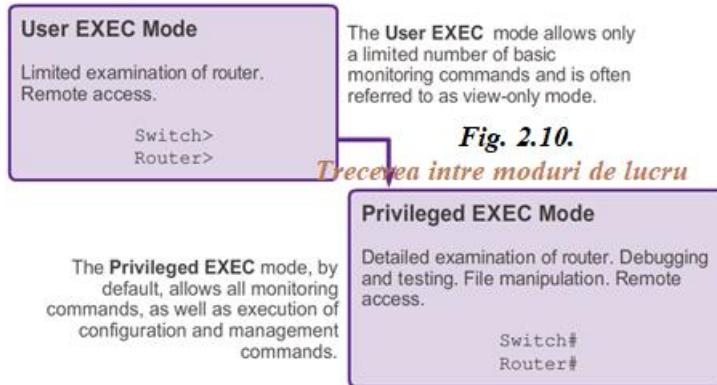


Fig. 2.10.

Treceerea intre moduri de lucru

Modul de config.re principal este numit config.re globală sau “global config”. Din modul de config.re globală, modificările de config.ție sunt efectuate, ele afectând funcționarea întregului dispozitiv. Modul de config.re globală este accesat înainte de a fi accesate alte moduri de config.re mai specifice.

Următoarea comandă CLI este folosită pentru a trece din modul privilegiat EXEC în modul de config.re globală și pentru a permite introducerea comenziilor de config.re:

`Router# configure terminal`

După executarea comenzi, promptul se schimbă pentru a afișa faptul că routerul se află în modul de config.re globală.

`Router(config)#`

Moduri de config.re Specifice – Din modul de config.re globală, utilizatorul poate accesa diferite submoduri de config.re. Fiecare dintre acestea permit config.rea unei anumite părți sau funcții a IOS-ului dispozitivului. Lista de mai jos prezintă câteva dintre ele:

- **Modul interfață** – config.rea unei interfețe (Fa0/0, S0/0/0)
- **Modul linie** – config.rea unei linii fizice sau virtuale (consola, AUX, VTY)

Fig. afișează prompturile pentru câteva dintre aceste moduri. Pentru a ieși dintr-un mod specific de config.re și a reveni la modul global de config.re, se introduce comanda `exit`. Pentru a părăsi complet modul de config.re și a reveni la modul privilegiat EXEC, se introduce comanda `end` sau se apasă concomitent tastele `Ctrl-Z`.

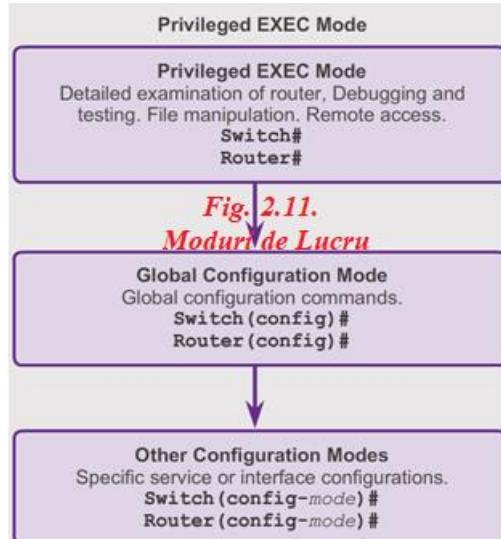


Fig. 2.11.

Moduri de Lucru

Prompturile Comenzilor – Când se utilizează CLI, modul poate fi identificat prin promptul liniei de comandă, acesta fiind unic modului respectiv. În mod implicit, fiecare prompt începe cu numele dispozitivului. După nume, restul promptului indică modul. De exemplu, promptul modului global de config.re de pe un router va fi:

Router(config)#

Pe măsura ce sunt introduse comenzi și modurile sunt schimbate, promptul se schimbă pentru a reflecta contextul curent, după cum se poate observa și în Fig..

Fig. 2.12 Structura prompturilor IOS

```

Router>ping 192.168.10.5
Schimbarea promptului denota modul de lucru curent
Router#show running-config

Router(config)#Interface FastEthernet 0/0

Router(config-if)#ip address 192.168.10.1 255.255.255.0

```

Mutarea între modul user EXEC și modul privilegiat EXEC – Comenziile **enable** și **disable** sunt folosite în CLI pentru a trece din modul utilizator EXEC în modul privilegiat EXEC, și invers.

Pentru a accesa modul privilegiat EXEC, este folosită comanda **enable**. Modul privilegiat EXEC este numit deobicei **modul enable**.

Sintaxa pentru introducerea comenzi **enable** este:

Router>**enable**

Această comandă este executată fără a fi nevoie de un alt argument sau cuvânt cheie. După apăsarea tastei ENTER, promptul se schimbă în:

Router #

Simbolul # de la sfârșitul promptului indică faptul că Routerul se află acum în modul privilegiat EXEC.

Dacă autentificarea cu parolă este config.tă pentru modul privilegiat EXEC, aceasta va trebui introdusă în dreptul promptului IOS. De exemplu:

Router > **enable**

Password: xxxxxxxxxxxx(*nu sunt afișate caracterele-metoda blank*)

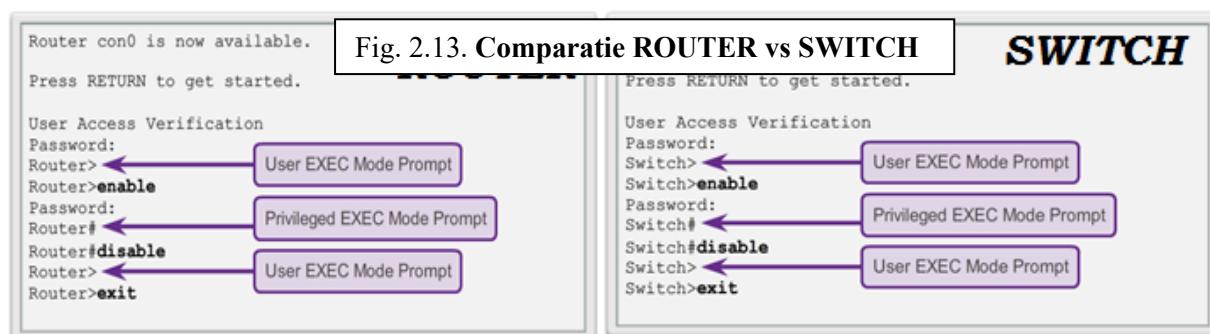
Router #

Comanda **disable** este folosită pentru a trece din modul privilegiat EXEC în modul utilizator EXEC. De exemplu:

Router # **disable**

Router >

După cum se poate observa în Fig., comenziile pentru accesarea modului privilegiat EXEC și pentru revenirea la modul utilizator EXEC, pe un switch Cisco, sunt identice cu cele folosite pe un Router Cisco.



Mutarea de la modul de configurație globală la submoduri – Pentru a ieși din modul global de config.re și a reveni la modul privilegiat EXEC, se introduce comanda **exit**.

Introducerea comenții **exit** în modul privilegiat EXEC încheie sesiunea curentă. Astfel, după introducerea comenții **exit** în modul privilegiat EXEC, va fi afișat ecranul de la inițierea unei sesiuni consolă. Aici trebuie apăsată tasta Enter pentru a accesa modul utilizator EXEC.

Pentru a trece din orice submod al modului de config.re globală în modul anterior, care se află cu o poziție mai sus în ierarhia modurilor, se introduce comanda **exit**. Sintaxa de mai jos ilustrează trecerea din modul utilizator EXEC în modul privilegiat EXEC, apoi accesarea modului de config.re globală, modului de config.re a interfeței, revenirea la modul de config.re globală și apoi la modul privilegiat EXEC folosind comanda **exit**.

Router>**enable**

Router#**configure terminal**

Enter config.tion commands, one per line. End with CTRL/Z.

Router(config)#**interface gigabitethernet 0/0**

Router(config-if)#**exit**

Router(config)#**exit**

Router#

Pentru a reveni din orice submod al modului privilegiat EXEC la modul privilegiat EXEC, se introduce comanda **end** sau se folosește combinația de taste **Ctrl+Z**. Sintaxa următoare pune în evidență revenirea din modul de config.re **interface gigabitethernet 0/0** la modul privilegiat EXEC, folosind comanda **end**.

Router#**configure terminal**

Enter config.tion commands, one per line. End with CTRL/Z.

Router(config)#**interface gigabitethernet 0/0**

Router(config-if)#**end**

Router#

Pentru a trece din orice submod al modului de config.re globală în alt submod “immediat” al modului de config.re globală, se tastează sintaxa corespunzătoare care în mod normal ar fi fost introdusă din modul de config.re globală. Următoarea sintaxă ilustrează trecerea din modul de config.re linie, Router(config-line)#, în modul de config.re interfață, Router(config-if)®, fără a ieși din modul de config.re linie.

Router#**configure terminal**

Enter config.tion commands, one per line. End with CTRL/Z.

Router(config)#line console 0

Router(config-line)#**interface gigabitethernet 0/0**

Router(config-if)#**end**

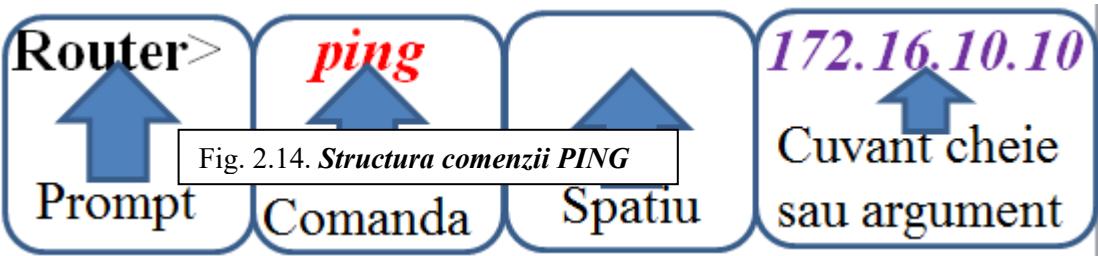
Router#

2.3.1 Structura de bază a comenzi IOS

Structura comenzi – Un echipament cu IOS suportă multe comenzi. Fiecare comandă IOS are un format specific sau o sintaxă și poate fi executată doar dintr-un anumit mod. Sintaxa generală a unei comenzi este comanda urmată de anumite cuvinte cheie sau argumente. Unele comenzi includ un subset de cuvinte cheie și argumente care oferă funcții suplimentare. Comenzile sunt folosite pentru a executa o acțiune, iar cuvintele cheie sunt folosite pentru a identifica unde și cum trebuie executată comanda.

După cum se poate observa în sintaxa dată:

Router>**ping 172.16.10.10**



comanda este reprezentată de cuvântul (cuvintele) inițial introdus în linia de comandă după prompt. Comenzile nu sunt case-sensitive. După comandă urmează unul sau mai multe cuvinte cheie și argumente. După introducerea fiecărei comenzi complete, inclusiv cuvintele cheie și argumentele, se apasă tasta Enter pentru a trimite comanda interpretorului de comenzi.

Cuvintele cheie descriu parametrii specifici, interpretorului de comenzi. De exemplu, comanda **show** este folosită pentru a afișa informații despre echipament. Această comandă are diferite cuvinte cheie care trebuie folosite pentru a defini rezultatul exact ce trebuie afișat. De exemplu:

Router# show running-config

Comanda **show** este urmată de cuvântul cheie **running-config**. Cuvântul cheie specifică faptul că rezultatul ce trebuie afișat este configurația curentă. **Convențiile**

comenzilor din IOS – O comandă poate necesita unul sau mai multe argumente. Spre deosebire de un cuvânt cheie, un argument nu este un cuvânt predefinit. Un argument este o valoare sau o variabilă definită de utilizator. Pentru a determina cuvintele cheie și argumentele unei comenzi, trebuie urmărită sintaxa comenzi. Sintaxa oferă modelul sau formatul care trebuie folosit în momentul introducerii comenzi.

De exemplu sintaxa comenzi **description** este:

Router(config-if)# description Legătura cu LAN 10

Există o serie de convenții ce trebuie respectate pentru a obține rezultatul dorit. Astfel, textul îngroșat indică comenzile și cuvintele cheie care trebuie introduse întocmai, iar textul italic indică un argument pentru care trebuie furnizată o valoare. Pentru comanda **description**, argumentul este un sir de caracter. Lungimea maximă a sirului de caractere este 80.

În consecință, când este aplicată o descriere unei interfețe folosind comanda **description**, este introdusă o linie precum cea de mai jos:

Router(config-if)# description Legătura cu R2

Comanda este **description** și argumentul definit de utilizator este "Legătura cu R2".

Următoarele exemple demonstrează câteva convenții folosite pentru a documenta și utiliza comenzi IOS.

Pentru comanda **ping**:

Sintaxa:

Router> ping IP-address

Exemplu cu valoare:

Router> ping 172.16.10.10

Comanda este **ping**, iar argumentul definit de utilizator este **172.16.10.10**.

În mod asemănător, sintaxa pentru a introduce comanda **traceroute** este:

Sintaxa:

Router> traceroute IP-address

Exemplu cu valoare:

Router> traceroute 192.168.10.254

Comanda este **traceroute**, iar argumentul definit de utilizator este **192.168.10.254**.

Cisco IOS Command Reference este o colecție online de documente ce descriu în detaliu comenzi IOS folosite pe dispozitivele Cisco. Command Reference este sursa fundamentală de

informații pentru o anumită comandă IOS, la fel cum dicționarul este sursa fundamentală de informații despre un anumit cuvânt.

Command Reference este o resursă fundamentală pe care inginerii de rețea o folosesc pentru a verifica diferite caracteristici ale unei comenzi IOS. Unele dintre cele mai comune caracteristici sunt:

- **Sintaxa** – cea mai detaliată versiune a sintaxei unei comenzi poate fi găsită.
- **Default** – felul în care o comandă este implementată pe un dispozitiv cu o configurație implicită.
- **Mod** – modul de configurație al dispozitivului de pe care este introdusă comanda.
- **Istoric** – descrieri referitoare la implementarea comenzi relative la versiunile IOS.
- **Reguli de folosire** – reguli ce descriu exact modul de implementare al comenzi.
- **Exemple** – exemple folosite care ilustrează scenarii tipice care folosesc respectiva comandă.

Pentru a naviga prin Command Reference și pentru a găsi o anumită comandă trebuie parcursi pașii de mai jos:

Pasul 1. Se accesează www.cisco.com.

Pasul 2. Se apasă pe **Support**.

Pasul 3. Se apasă pe **Networking Software (IOS & NX-OS)**.

Pasul 4. Se apasă pe **15.2M&T** (de exemplu).

Pasul 5. Se apasă pe **Reference Guides**.

Pasul 6. Se apasă pe **CommandReferences**.

Pasul 7. Se apasă pe tehnologia particulară care cuprinde comanda referită.

Pasul 8. Se apasă pe legătura din stânga, care corespunde alfabetic comenzi referite.

Pasul 9. Se apasă pe legătura comenzi.

De exemplu, comanda **description** poate fi găsită la *Cisco IOS Interface and Hardware Component Command Reference*, la legătura pentru ordinea alfabetică D-E.

Notă: Versiuni PDF complete ale referințelor comenzi pentru o anumită tehnologie pot fi descărcate de la legăturile de pe pagina accesată după **Pasul 7** de mai sus.

IOSul are multe forme de asistență disponibile:

- Context-Sensitive Help
- Command Syntax Check
- Hot Keys and Shortcuts

Context-Sensitive Help – Ajutorul sensibil la context oferă o listă de comenzi și argumentele asociate acestora, în contextul modului curent. Pentru a accesa acest ajutor sensibil la context, se introduce semnul întrebării, ?, în dreptul unui prompt. Va urma un răspuns imediat, fără a fi necesară apăsarea tastei Enter.

O utilizare a ajutorului sensibil la context este aceea de a obține o listă cu comenzi disponibile. Aceasta poate fi folosită în momentul în care nu este cunoscută sintaxa unde se verifică dacă IOSul suportă o anumită comandă într-un anumit mod.

Spre exemplu, pentru a vedea toate comenzi disponibile la nivelul utilizator EXEC, se introduce semnul întrebării, ?, după promptul Router>.

O altă folosire a ajutorului sensibil la context este aceea de a obține o listă cu comenzi sau cuvinte cheie care încep cu un anumit caracter sau cu anumite caractere. După introducerea unei secvențe de caractere, dacă un semn de întrebare este introdus imediat, fără spațiu, IOSul va afișa o listă cu comenzi sau cuvintele cheie ale contextului respectiv, care încep cu caracterele introduse.

De exemplu, dacă se introduce **sh?** se obține o listă de comenzi care încep cu secvență de caractere **sh**.

Un tip final de ajutor sensibil la context este folosit pentru a determina opțiunile, cuvintele cheie, sau argumentele care corespund unei anumite comenzi. Când se introduce o comandă, se introduce un spațiu urmat de un "?" pentru a determina ce trebuie sau ce poate fi introdus în continuare.

După cum se poate observa pentru sintaxa dată, după introducerea comenzi:

clock set 19:50:00,

putem introduce semnul întrebării pentru a determina opțiunile sau cuvintele cheie adiționale, disponibile pentru aceasta comandă.

Router#cl? - prezintă o listă de comenzi sau cuvinte cheie care încep cu caracter **cl**.
clear clock

Router#clock set ? – IOS afișează ce argumente sau cuvinte cheie trebuie să urmeze.

hh:mm:ss Current Time

Router#clock set 12:51:50 ? – prezintă ce argumente sau cuvinte cheie sunt necesare.
<1-31> Day of the month MONTH Month of the year

Router#clock set 12:51:50 10 April 2015

Command Syntax Check – Când o comandă este executată prin apăsarea tastei Enter, interpretorul liniei de comandă analizează comanda de la stânga la dreapta pentru a determina ce acțiune trebuie efectuată. IOSul oferă în general doar feedback negativ, după cum se poate observa în sintaxa de mai jos. Dacă interpretorul înțelege comanda, acțiunea va fi executată, iar CLI va reveni la promptul corespunzător. Totuși, dacă interpretorul nu înțelege comanda introdusă, va oferi feedback, descriind ce este în neregulă cu comanda.

Router#clock se – IOS întoarce un mesaj de ajutor care solicită argumentele sau cuvintele cheie.

% Incomplete command.

Router#clock set 12:59:00

% Incomplete command.

Router#a - IOS întoarce un mesaj de ajutor care spune că nu sunt inserate suficiente caractere.

% Ambiguous command: "c"

Router#clock set 12:59:00 10 6

% Invalid input detected at '^' marker. – IOSul întoarce simbolul ^ care indică faptul că interpretorul nu poate decripta comanda.

De aici se poate trage concluzia că pot fi 3 tipuri diferite de mesaje de eroare:

- Comandă ambiguă
- Comandă incompletă
- Comandă incorectă

Comanda **clock set** este o comandă IOS ideală pentru a experimenta mesajele de ajutor, create după verificarea sintaxei comenziilor, după cum se poate observa.

Hot Keys and Shortcuts – CLI din IOS oferă combinații de taste și scurtături care facilitează configarea, monitorizarea și depanarea. Următoarele taste sunt considerate mai importante:

- **Săgeata Jos** – permite deplasarea înainte prin comenzi introduse anterior
- **Săgeata Sus** - permite deplasarea înapoi prin comenzi introduse anterior
- **Tab** – Completează o comandă sau un cuvânt cheie scris parțial
- **Ctrl-A** – Mută cursorul la începutul liniei
- **Ctrl-E** - Mută cursorul la sfârșitul liniei
- **Ctrl-R** – Afisează din nou o linie

- **Ctrl-Z** – Părăsește modul de config.re și revine la modul utilizator EXEC
- **Ctrl-C** – Părăsește modul de config.re și intrerupe comanda curentă
- **Ctrl-Shift-6** – Permite întreruperea unui proces IOS precum **ping** sau **traceroute**

Pentru o înțelegere mai bună este de preferat examinarea unora dintre acestea în detaliu.

Tab - Tasta **Tab** este folosită pentru a completa restul unei comenzi sau parametrii abreviați dacă abreviația conține suficiente litere pentru a identifica în mod unic comanda sau parametrul. Când o parte suficient de mare din comandă a fost introdusă pentru a asigura unicitatea, se apasă tasta **Tab**, iar apoi CLI va afișa restul comenzi.

Aceasta este o tehnică bună de folosit doar în momentul în care se învață deoarece permite vizualizarea întregului cuvânt folosit pentru o comandă sau un cuvânt cheie.

Ctrl-R - Afisează din nou linia introdusă anterior. De exemplu, IOSul trimite un mesaj către CLI în timp ce se scrie o linie. Se poate folosi **Ctrl-R** pentru a afișa din nou linia, evitând reintroducerea ei.

În acest exemplu, un mesaj cu privire la o interfață care a cedat apare în mijlocul comenzi.

Router# **show mac**

```
16w4d: %LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to
down
```

```
16w4d: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/1, changed state to down
```

Pentru a afișa din nou linia pe care se folosește combinația de taste Ctrl-R:

Router# **show mac**

Ctrl-Z - Părăsește orice mod de config.re și revine în modul privilegiat EXEC. Deoarece IOSul are o structură ierarhică de moduri, se poate ajunge cu multe niveluri mai jos. În loc să fie părăsit fiecare mod în parte, se folosește combinația de taste Ctrl-Z pentru a reveni direct în modul privilegiat EXEC.

Săgețile în sus și în jos - Aceste taste permit accesul la istoricul comenzi introduse. Softwareul Cisco IOS stochează multe comenzi introduse anterior pentru ca acestea să poată fi refolosite. Memoria tampon este folositoare pentru reintroducerea comenziilor fără a le retasta. Sunt disponibile secvențe de chei pentru a parcurge aceste comenzi salvate. Tasta **Săgeata Sus** (**Ctrl-P**) folosită pentru a afișa comenzi introduse anterior. De fiecare dată când tasta este apăsată, următoarea comandă mai veche va fi afișată. Tasta **Săgeata Jos** (**Ctrl-N**) pentru a înainta în istoric și pentru a afișa comenzi mai recente.

Ctrl-Shift-6 - Secvența de ieșire va întrerupe orice proces activ. Când un proces IOS este lansat din CLI, precum **ping** sau **traceroute**, comanda va rula până când va fi completă sau va fi întreruptă. Cât timp procesul este activ, CLI nu poate fi folosit. Pentru a-l întrerupe și pentru a interacționa cu CLI, se apasă simultan tastele **Ctrl-Shift-6**.

Ctrl-C - Aceasta întrerupe o comandă și părăsește modul de config.re. Este folositoare după introducerea unei comenzi care trebuie opriță.

Abrevierea comenzielor sau cuvintelor cheie - Comenzi și cuvintele cheie pot fi abreviate cu numărul minim de caractere care le identifică în mod unic. De exemplu, comanda **configure** poate fi abreviată cu **conf** deoarece **configure** este singura comandă care începe cu **conf**. Abreviația **con** nu va funcționa deoarece există mai multe comenzi care încep cu **con**.

De asemenea, și cuvintele cheie pot fi abreviate.

Spre exemplu, comanda **show interfaces** poate fi abreviată astfel:

```
Router# show interfaces
```

```
Router# show int
```

Pot fi abreviate ambele comenzi și cuvintele cheie astfel:

```
Router# sh int
```

Pentru a verifica și depăsa funcționarea rețelei, trebuie să analizăm funcționarea dispozitivelor. Comanda de bază pentru examinare este comanda **show**.

Există multe variante ale acestei comenzi. Pe măsură ce se deprinde experiență în lucru cu IOSul, va deveni relativ ușoară folosirea și interpretarea rezultatelor comenziilor **show**. Se folosește comanda **show ?** pentru a obține o listă cu comenzi disponibile într-un anumit context sau mod.

O comandă **show** tipică poate oferi informații despre configurație, despre funcționare, și despre stările părților unui router sau switch Cisco.

În acest curs, ne vom concentra asupra comenziilor **show** de bază.

O comandă **show** des utilizată este **show interfaces**. Aceasta afișează statistici pentru toate interfețele dispozitivului. Pentru a vedea statisticile unei anumite interfețe, se introduce comanda **show interfaces** urmată de tipul acelei interfețe și de numărul portului/slotului. De exemplu:

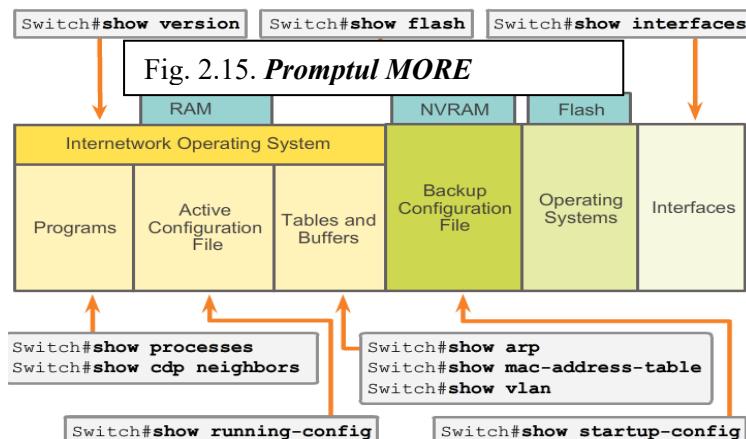
```
Router# show interfaces gigabitethernet 0/1
```

Alte comenzi **show** des utilizate de tehnicienii de rețea sunt:

show startup-config – afișează configurația salvată, localizată în NVRAM

show running-config – afișează conținutul fișierului de configurație activ

Promptul More - Când o comandă oferă un rezultat prea mare, care nu poate fi afișat pe un singur ecran, promptul – **More** – apare în josul ecranului. Când acest prompt apare, se apasă tasta **Space** pentru a vedea următoarea parte din rezultat. Pentru a afișa doar următoarea linie, se apasă tasta **Enter**. Dacă orice altă tasta este apăsată, afișarea rezultatului va fi întreruptă și se revine la prompt.



IOS **show** commands can provide information about the configuration, operation, and status of parts of a Cisco switch or router.

Una dintre cele mai folosite comenzi pe un switch sau router este:

```
Router# show version
```

Această comandă afișează informații legate de versiunea curentă de IOS, de dispozitiv și de hardware. Dacă este accesat un router sau un switch de la distanță, comanda **show version** este

un instrument util pentru a se afla informații despre dispozitivul la care suntem conectați. Câteva dintre informațiile pe care această comandă le oferă sunt:

- **Versiune software** – versiunea softwareului IOS (stocat în flash).
- **Versiune bootstrap** – versiunea bootstrapului (stocat în Boot ROM).
- **Timpul de funcționare al sistemului** – timpul scurs de la ultimul reboot.
- **Informații despre repornirea sistemului** – metoda de repornire (spre exemplu: power cycle, cedare).
- **Numele imaginii software** – numele fișierului IOS stocat în flash.
- **Tipul routerului și tipul procesorului** – modelul routerului și tipul procesorului.
- **Tipul memoriei și alocarea (shared/main)** - Main Processor RAM și Shared Packet I/O buffering.
- **Caracteristici software** – seturile de protocole/caracteristici acceptate.
- **Interfețe hardware** – interfețele disponibile ale dispozitivului.
- **Registru de configurație** – stabilește specificațiile bootup, setările de viteză ale consolii, și alți parametrii.

Outputul de mai jos afișează rezultatul pentru un echipament Router Cisco 2901:

```

Physical Config CLI
IOS Command Line Interface
ROUTER#show version
Router#show version
Cisco IOS Software, C2900 Software (C2900-UNIVERSALK9-M), Version 15.1(4)M4,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2011 by Cisco Systems, Inc.
Compiled Thu 5-Jan-12 15:41 by pt_team

ROM: System Bootstrap, Version 15.1(4)M4, RELEASE SOFTWARE (fc1)
cisco2901 uptime is 13 hours, 44 minutes, 39 seconds
System returned to ROM by power-on
System image file is "flash0:c2900-universalk9-mz.SPA.151-1.M4.bin"
Last reload type: Normal Reload

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
that you are licensed to use any specific algorithm or key size.
Import, export, distribution or use of Cisco cryptographic products
by third party manufacturers, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/w处处/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.
Cisco CISCO2901/K9 (revision 1.0) with 491520K/32768K bytes of memory.
Processor board ID FTX152400KS

```

Outputul de mai jos afișează rezultatul pentru un echipament switch Cisco 2960 :

```

Physical Config CLI
IOS Command Line Interface
Switch>show version
Switch#show version
Cisco IOS Software, C2960 Software (C2960-LANBASE-M), Version 12.2(25)FX, RELEASE
SOFTWARE (fc1)
Copyright (c) 1986-2005 by Cisco Systems, Inc.
Compiled Wed 12-Oct-05 22:05 by pt_team

ROM: C2960 Boot Loader (C2960-HBOOT-M) Version 12.2(25r)FX, RELEASE SOFTWARE (fc4)
System returned to ROM by power-on
Cisco WS-C2960-24TT (RC32300) processor (revision C0) with 21039K bytes of memory.

24 FastEthernet/IEEE 802.3 interface(s)
2 Gigabit Ethernet/IEEE 802.3 interface(s)

63488K bytes of flash-simulated non-volatile configuration memory.
Base ethernet MAC Address : 0001.966C.0550
Motherboard assembly number : 73-9882-06
Power supply part number : 341-0097-02
Motherboard serial number : FOC10324BMJ
Power supply serial number : DCA102133JA
Model revision number : B0
Motherboard revision number : C0
Model number : WS-C2960-24TT
System serial number : FOC103321EY
--More-- |
```

2.4 Noțiuni de bază – Nume de gazdă

Switchurile și routerele au multe asemănări. Ele suportă un sistem de operare modular asemănător, suportă structuri de comenzi de sintaxă similară, și suportă multe comenzi identice. Mai mult, ambele dispozitive au aceeași pașii de configurație inițială la implementarea într-o rețea.

Totuși, un switch Cisco IOS este unul dintre cele mai simple dispozitive care pot fi configurate într-o rețea. Acest lucru se datorează faptului că nu există configurații necesare înaintea funcționării dispozitivului. În forma să de bază, un switch poate fi pornit fără nici-o configurație, dar va schimba date între dispozitivele conectate.

Un switch este de asemenea unul dintre dispozitivele fundamentale folosite în crearea rețelelor mici. Prin conectarea a două calculatoare la un switch, acele calculatoare vor avea instant conectivitate între ele.

Datorită acestor motive, restul capitolului se va concentra asupra creării unei rețele mici, cu două calculatoare conectate printr-un switch configurat cu setări inițiale. Setările inițiale includ setarea numelui switchului, limitarea accesului la configurația dispozitivului, configurația mesajelor banner și salvarea configurației.



Fig. 2.16. Front of SWITCH

Când configurăm un dispozitiv de rețea, unul dintre primii pași este setarea unui nume unic pentru dispozitiv - **hostname**. Hostnameurile apar în prompturile CLI, pot fi folosite în diferite procese de autentificare între dispozitive și ar trebui folosite în diagramele topologilor.

Hostnameurile sunt configurate pe dispozitivul de rețea activ. Dacă numele dispozitivului nu este configurat explicit, un nume implicit va fi folosit de Cisco IOS. Numele implicit pentru un switch Cisco IOS este "Switch".

Dacă ne imaginăm că o rețea ar avea multe switchuri, toate având numele implicit "Switch", cum se poate observa în Fig. următoare:

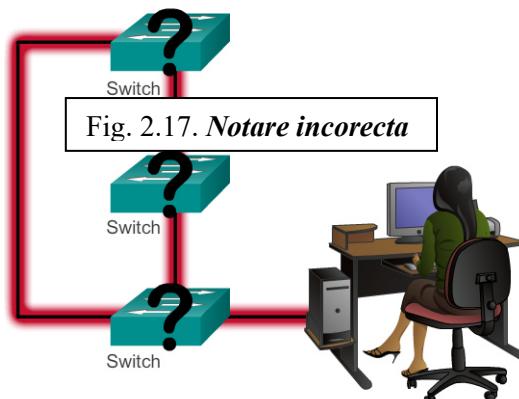


Fig. 2.17. Notare incorecta

acest lucru ar crea confuzie în configurația și administrarea rețelei. Când este accesat un dispozitiv de la distanță prin SSH, este important să avem o confirmare că suntem conectați la dispozitivul corect. Dacă toate dispozitivele ar fi lăsate cu numele implicit, ar fi foarte dificil să identificăm dacă suntem conectați la dispozitivul corect.

Alegând numele inteligent, este mai ușor de ținut minte, de discutat, de documentat, și de identificat dispozitivele de rețea. Denumirea dispozitivelor într-un mod consistent și folositor, necesită stabilirea unor convenții la nivelul companiei sau, cel puțin, la nivelul locației. Este o practică bună crearea convențiilor de nume în același timp cu schema de adresare pentru a permite o continuitate în organizare.

Câteva reguli pentru convențiile de nume sunt că numele trebuie:

- *Să înceapă cu o literă.*
- *Să nu conțină spații.*
- *Să se termine cu o literă sau o cifră.*
- *Să conțină doar litere, cifre și cratime.*
- *Să aibă mai puțin de 64 de caractere.*

Hostnameurile folosite în IOSul dispozitivului păstrează forma exactă a caracterelor (litere mici sau majuscule). Astfel, putem scrie un nume în orice fel. Acest lucru este în contrast cu majoritatea convențiilor de nume din Internet, unde literele mici și mari nu sunt tratate identic.

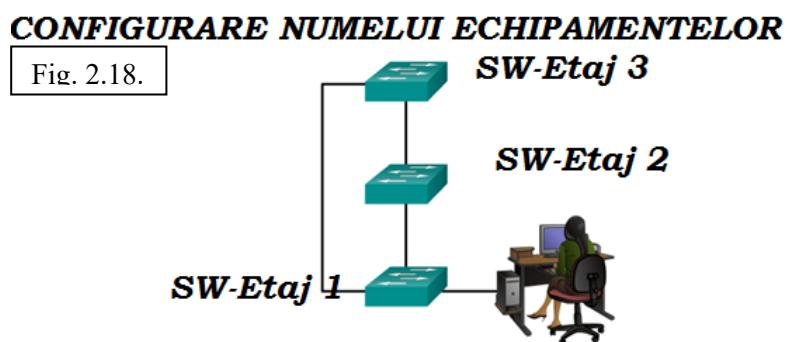
Hostnameurile permit identificarea lor de către administratorii de rețea, într-o rețea sau în Internet.

2.5 Exemplu pentru Setarea Numelui

Vom folosi un exemplu cu 3 switchuri conectate împreună într-o rețea, aflate la 3 etaje diferite.

Pentru a crea o convenție de nume pentru switchuri, se iau în considerare locațiile și scopurile dispozitivelor.

De exemplu, pentru Fig. de mai jos:



pentru cele 3 switchuri am dat numele SW-Etaj1, Sw-Etaj 2, respectiv Sw-Etaj 3.

În documentația rețelei, am include aceste nume, și motivele alegerii lor, pentru a asigura o continuitate în convențiile de nume, pe măsură ce mai multe dispozitive sunt adăugate.

Odată ce convenția de nume a fost identificată, următorul pas este aplicarea numelor dispozitivelor folosind CLI.

2.5.1 ConFig.rea Numelor Gazdelor prin intermediul IOS

Din modul EXEC privilegiat, se accesează modul de config.re global prin introducerea comenzi **configure terminal**:

Router# **configure terminal**

După ce comanda a fost executată, promptul va fi schimbat în:

Router(config)#

După cum am putut observa în Fig. precedentă, în modul de config.re global, se introduce numele gazdei:

Router(config)# **hostname R-Floor-1**

După ce comanda este executată, promptul va fi schimbat în:

R-Floor-1 (config)#+

Se poate observa faptul că numele echipamentului apare în prompt. Pentru a părăsi modul de config.re global, se folosește comanda **exit**.

Trebuie să ne asigurăm mereu că documentația este actualizată de fiecare dată când un dispozitiv este adăugat sau modificat. Astfel dispozitivele pot fi identificate din documentație, după locație, scop și adresă.

Notă: Pentru a anula efectele unei comenzi se adaugă cuvântul cheie **no** în fața comenzi.

Spre exemplu pentru a șterge numele unui dispozitiv, se folosește sintaxa:

```
R-Floor-1 (config)# no hostname
```

```
Router(config)#
```

Se poate observa faptul că această comandă, **no hostname**, a făcut ca Routerul să revină la numele implicit, "Router".

2.5.2 Limitarea Accesului la ConFig.țiiile Dispozitivelor

Limitarea accesului fizic la dispozitivele de rețea, prin plasarea lor în dulapuri și în rackuri încuiate, este o practică bună; totuși, parolele reprezintă apărarea primară împotriva accesului neautorizat la dispozitivele de rețea. Fiecare dispozitiv, inclusiv routerele de casă, ar trebui să aibă conFig.te parole pentru a limita accesul. De asemenea, tot o bună practică este reprezentată prin întărirea securității prin solicitarea unui nume de user împreună cu parola. Deocamdată, vom prezenta precauții de securitate de bază folosind doar parole.

După cum am putut observa anterior, IOSul folosește moduri ierarhice pentru a realiza securitatea dispozitivului. Ca parte a acestei constrângeri de securitate, IOSul poate accepta multe parole pentru a permite diferite privilegii de acces la dispozitiv.

Parolele utilizate în coFig.țiiile de bază sunt:

- **Enable password** – limitează accesul la modul privilegiat EXEC.
- **Enable secret** – criptat, limitează accesul la modul privilegiat EXEC.
- **Console password** – limitează accesul la dispozitiv utilizând conexiunea consolă.
- **VTY password** – limitează accesul la dispozitiv prin Telnet.

Ca o practică bună, este de preferat să fie folosite parole de autentificare diferite pentru fiecare dintre aceste niveluri de acces. Cu toate că logarea cu multe parole diferite este incomodă, este o precauție necesară pentru a proteja în mod corespunzător infrastructura rețelei de accesul neautorizat.

În plus, sunt folosite parole puternice care nu sunt ușor de ghicit. Folosirea parolelor slabe sau ușor de ghicit continuă să fie o problemă de securitate în multe ipostaze ale lumii afacerilor.

De aceea, sunt luate în considerare aceste puncte cheie în momentul alegerii parolelor:

- *Folosirea parolelor ce au mai mult de 8 caractere.*
- *Folosirea unei combinații de litere mici și majuscule, numere, caractere speciale, și/sau secvențe numerice.*
- *Evitarea folosirii aceleiași parole pentru toate dispozitivele.*
- *Evitarea folosirii cuvintelor comune precum password sau administrator, deoarece sunt ușor de ghicit.*

Notă: În majoritatea laboratoarelor ce completează acest curs, sunt folosite parole simple precum. Aceste parole sunt considerate slabe și ușor de ghicit, și trebuie evitate într-un mediu de lucru. Astfel de parole sunt utilizate aici pentru eficiență în laborator sau pentru a ilustra exemple de conFig.re.

Pentru a securiza accesul la modul privilegiat EXEC, este folosită comanda **enable secret password**. O versiune mai veche și mai puțin sigură a acestei comenzi este comanda **enable password password**. Cu toate că oricare dintre acestea poate fi folosită pentru autentificare

înainte de permiterea accesului la modul privilegiat EXEC, este recomandată folosirea comenții **enable secret**. Comanda **enable secret** oferă o securitate sporită deoarece parola este criptată.

Exemplu de comandă pentru setarea parolelor:

```
Router(config)# enable secret ctifmi
```

Exemplul din figură ilustrează cum o parolă nu este solicitată la prima utilizare a comenții **enable**. Apoi comanda **enable secret ctifmi** este config.tă și accesul la modul privilegiat EXEC devine securizat. Se poate observa că din motive de securitate, parola nu este afișată în momentul introducerii ei.

```
R-Etaj1>enable
R-Etaj1#
R-Etaj1#configure terminal
R-Etaj1(config)# enable secret ctifmi
R-Etaj1(config)#exit
R-Etaj1#
R-Etaj1>enable
Password:
R-Etaj1#
```

Portul consolă al dispozitivelor de rețea trebuie să fie securizat, măcar prin solicitarea unei parole puternice. Acest lucru reduce şansele ca personalul neautorizat să obțină acces la dispozitiv prin conectarea fizică la dispozitiv prin un cablu.

Următoarele comenzi sunt folosite în modul de config.re global pentru a seta o parolă pentru linia de consola:

```
Router(config)# line console 0
Router (config-line)# password fmicti
Router (config-line)# login
```

Din modul de config.re global, comanda **line console 0** este folosită pentru a accesa modul de config.re linie pentru consolă. Cifra 0 este folosită pentru a reprezenta prima (și în multe cazuri singura) interfață de consolă.

A doua comandă, **password fmicti** specifică parola pentru linia consolă.

Comanda **login** configurează Routerul pentru a solicita autentificarea în momentul logării. Când logarea este activă și o parolă este setată, utilizatorul consolei va trebui să introducă o parolă înainte de a obține acces la CLI.

VTY Password – Liniile VTY permit accesul la un dispozitiv prin serviciul Telnet. În mod implicit, majoritatea modelelor de switchuri suportă până la 16 liniilor vty, numerotate de la 0 la 15. Numărul de liniile vty suportă pe un router variază în funcție de tipul routerului și de versiunea IOS. Totuși, 16 este cel mai comun număr de liniile vty config.te. Aceste liniile sunt numerotate de la 0 la 4, în mod implicit, cu toate că liniile adiționale pot fi config.te. O parolă trebuie config.tă pentru toate liniile vty disponibile. Aceeași parolă poate fi setată pentru toate conexiunile, totuși, este de preferat ca fiecare linie să aibă o parolă unică pentru a oferi o rezervă pentru accesul administrativ la dispozitiv, dacă celelalte conexiuni sunt ocupate.

Exemple de comenzi folosite pentru a seta o parolă pe liniile vty:

```
Router(config)# line vty 0 15
Router (config-line)# password fmicti
Router (config-line)# login
```

În mod implicit, IOSul include comanda **login** pe liniile VTY. Acest lucru împiedică accesul la dispozitiv, prin Telnet, fără autentificare. Dacă, din greșală, comanda **no login** este folosită, comandă care elimină necesitatea autentificării, persoane neautorizate se pot conecta la dispozitiv prin Telnet. Aceasta ar fi un risc de securitate major.

Sintaxa de mai jos ilustrează securizarea modului utilizator EXEC pe liniile consolă și Telnet.

```
R-Etaj1(config)#line console 0
R-Etaj1(config-line)#password ctifmi
R-Etaj1(config-line)#login
R-Etaj1(config-line)#exit
R-Etaj1(config)#line vty 0 15
R-Etaj1(config-line)#password fmicti
R-Etaj1(config-line)#login
R-Etaj1(config-line)#exit
```

O altă comandă folosită impiedică apariția parolelor în text clar, în momentul vizualizării fișierelor de conFig.re. Comanda este **service password-encryption**.

Această comandă conduce la criptarea parolelor în momentul configurării lor. Comanda **service password-encryption** aplică o criptare slabă tuturor parolelor necriptate. Această criptare se aplică doar parolelor din fișierul de conFig.re, nu și parolelor care sunt trimise în rețea. Scopul acestei comenzi este împiedicarea indivizilor neautorizați să poată vedea parolele din fișierul de conFig.re în text clar.

Dacă se introduce comanda **show running-config** sau **show startup-config** înainte de a folosi comanda **service password-encryption**, parolele necriptate vor fi vizibile. Comanda **service password-encryption** poate fi executată apoi și criptarea va fi aplicată paralelor. Odată ce criptarea a fost aplicată, eliminarea serviciului de criptare nu anulează criptarea.

Cu toate că solicitarea paralelor este o cale de a preveni accesul neautorizat la rețea, este vitală oferirea unei metode pentru declararea faptului că doar personalul autorizat ar trebui să încerce să obțină acces la dispozitiv. Pentru a face acest lucru, se adaugă un banner în ieșirea dispozitivului.

Bannerele pot fi o parte importantă a unui proces legal în cazul în care o persoană este acuzată de accesarea ilegală a unui dispozitiv. Câteva sisteme legale nu permit urmărirea penală, sau monitorizarea utilizatorilor, dacă o notificare nu este vizibilă.

Conținutul exact al unui banner depinde de legile locale și de politicile corporatiste. Câteva exemple de informații de inclus într-un banner:

- *"Accessul permis doar personalului strict autorizat."*
- *"Activitatea este monitorizată, pătrunderea ilegală se pedepsește drastic."*
- *"Pot fi luate cele mai aspre măsuri legale în cazul folosirii neautorizate."*

Deoarece bannerele pot fi văzute de oricine încearcă să acceseze dispozitivul, mesajul trebuie formulat foarte atent. Orice formulare care sugerează o invitație la logare nu este adecvată. Dacă o persoană afectează rețea după ce a obținut acces neautorizat, tragerea acesteia la răspundere va fi dificil de dovedit dacă există un banner interpretat ca o invitație.

Crearea bannerelor este un proces simplu; totuși, bannerele ar trebui folosite în mod adecvat. Când un banner este folosit, acesta nu trebuie să invite pe nimeni să configureze un dispozitiv. Ar trebui să detalieze faptul că doar personalul autorizat poate accesa dispozitivul. Mai mult, bannerul poate include opriri programate ale sistemului și alte informații care afectează toți utilizatorii rețelei.

IOSul oferă multe tipuri de bannere. Un banner comun este mesajul zilei (message of the day - MOTD). Este des folosit pentru notificări legale deoarece este afișat tuturor terminalelor conectate.

Pentru a conFig. MOTD se folosește comanda **banner motd** din modul de conFig.re global.

Comanda **banner motd** necesită folosirea unor separatori pentru a identifica conținutul mesajului. Comanda **banner motd** este urmată de un spațiu și de un separator. Apoi, una sau mai multe linii de text sunt introduse, reprezentând mesajul bannerului. A doua apariție a

separatorului indică sfârșitul mesajului. Separatorul poate fi orice caracter care nu se regăsește în mesaj. Din acest motiv, simboluri precum '#' sunt folosite.

Sintaxa configurării MOTD, din modul de conFig.re global este:

```
Router(config)# banner motd # Serverul intră în menenanță vineri la ora 16:00 ! #
```

Odată ce comanda a fost executată, bannerul va fi afișat la fiecare încercare de a obține accesul la dispozitiv, până când bannerul este eliminat.

Exemplul ilustrează conFig.rea unui banner folosind ca separator simbolul '#'. Se poate observa cum este afișat bannerul în momentul accesării echipamentului.

```
R-Etaj1(config)#banner motd # Serverul intră în menenanță vineri ora la 16:00 ! #
R-Etaj1(config)#end
R-Etaj1#
%SYS-5-CONFIG_I: Configured from console by console
R-Etaj1#exit
R-Etaj1 con0 is now available
Press RETURN to get started.
Serverul intră în menenanță vineri ora la 16:00 !
R-Etaj1>
```

2.5.3 Salvarea ConFig.ților

Fișierul de conFig.re curentă reflectă conFig.ția curentă a IOSului dispozitivului. Conține comenzi folosite pentru a determina modul de funcționare al dispozitivului în rețea. Modificarea conFig.ției curente afectează funcționarea unui dispozitiv imediat.

Fișierul de conFig.re curentă este stocat în memoria de lucru a dispozitivului, sau în RAM. Astfel, fișierul de conFig.re curentă este activ temporar, cât timp dispozitivul este pornit. Totuși, dacă alimentarea dispozitivului este întreruptă sau dispozitivul este repornit, toate schimbările de conFig.re vor fi pierdute, dacă nu au fost salvate.

După efectuarea unor schimbări asupra fișierului de conFig.re curentă, pot fi luate în considerare următoarele opțiuni:

- *Se aduce dispozitivul la conFig.ția să inițială.*
- *Se șterg toate conFig.țile de pe dispozitiv.*
- *ConFig.ția schimbăță, devine noua conFig.ție de pornire.*

Fișierul de conFig.ție de pornire reflectă conFig.ția ce va fi folosită de dispozitiv după repornire. Fișierul este stocat în NVRAM. Când un dispozitiv de rețea a fost conFig.t și conFig.ția curentă a fost modificată, este importantă salvarea acelor schimbări în fișierul de conFig.re de pornire. Astfel, schimbările nu se vor pierde în cazul întreruperii alimentării sau în cazul repornirii intenționate.

Înainte de comiterea schimbărilor, se folosesc comenzi **show** adecvate pentru a verifica funcționarea dispozitivului. Comanda **show running-config** poate fi folosită pentru a vedea fișierul de conFig.re curentă. După verificarea corectitudinii schimbărilor, se folosește **comanda copy running-config startup-config** în modul privilegiat EXEC. Comanda pentru salvarea conFig.ției curente în fișierul de conFig.re de pornire este:

```
Router# copy running-config startup-config
```

După executare, fișierul de conFig.re curentă actualizează fișierul de conFig.re de pornire.

Dacă schimbările efectuate asupra conFig.ției curente nu produc efectul dorit, poate fi necesară restaurarea conFig.ției anterioare a dispozitivului. Presupunând că nu a fost suprascrisă conFig.ția de pornire cu schimbările, se poate înlocui conFig.ția curentă cu cea de pornire. Acest

lucru poate fi realizat cel mai bine prin repornirea dispozitivului, folosind comanda **reload** din modul privilegiat EXEC.

Când este inițiată o repornire, IOSul detectează schimbările din config.ția curentă care nu au fost salvate în config.ția de pornire. În aceste condiții va fi afișat un prompt pentru a întreba dacă se dorește salvarea schimbărilor. Pentru a renunța la schimbări, se introduce **n** sau **no**.

Un prompt suplimentar va apărea pentru a confirma repornirea. Pentru a confirma, se apasă tasta Enter. Apăsarea oricărei alte taste va întrerupe procesul.

Exemplu:

```
"R-Etaj1# reload
System config.tion has been modified. Save? [yes/no]: n
Proceed with reload? [confirm]
*Apr 17 01:34:15.758: %SYS-5-RELOAD: Reload requested by console. Reload Reason:
Reload Command.
System Bootstrap, Version 12.3(8r)T8, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2004 by cisco Systems, Inc. PLD version 0x10 GIO ASIC version 0x127
c1841 processor with 131072 Kbytes of main memory
Main memory is configured to 64 bit mode with parity disabled"
```

Dacă anumite schimbări nedорite sunt salvate în config.ția de pornire, este necesară ștergerea tuturor config.țiilor. Astfel, config.ția de pornire trebuie ștersă, iar dispozitivul trebuie repornit.

Config.ția de pornire este ștersă prin comanda **erase startup-config**.

Pentru a șterge fișierul de config.re de pornire se folosește comanda **erase NVRAM:startup-config** sau **erase startup-config** în modul privilegiat EXEC:

```
Router# erase startup-config
```

După lansarea comenzii, echipamentul va solicita o confirmare:

```
R-Etaj1#erase startup-config
```

Erasing the nvram filesystem will remove all config.tion files! Continue? [confirm]

Confirmarea este răspunsul implicit. Pentru a confirma și șterge fișierul de config.re de pornire, se apasă tasta Enter. Apăsarea oricărei alte taste va întrerupe procesul.

Atenție: Mare atenție în momentul utilizării comenzii **erase**. Această comandă poate fi folosită pentru a șterge orice fișier de pe dispozitiv. Utilizarea necorespunzătoare a comenzii poate șterge IOSul sau alt fișier critic.

Pe un switch este necesar să fie folosită și comanda **delete vlan.dat**, în afară de comanda **erase startup-config**, pentru a aduce dispozitivul la config.ția implicită (asemănătoare cu o resetare la setările din fabrică):

```
Switch# delete vlan.dat
```

Delete filename [vlan.dat]?

Delete flash:vlan.dat? [confirm]

```
Switch# erase startup-config
```

Erasing the nvram filesystem will remove all config.tion files! Continue?

[confirm]

[OK]

Erase of nvram: complete

```
Switch#
```

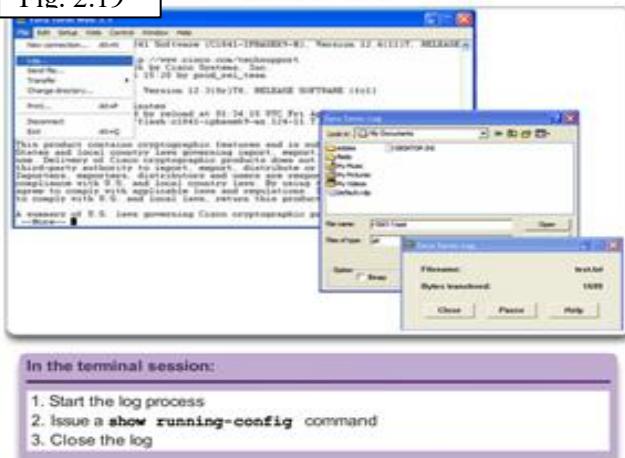
După ștergerea config.ției de pornire din NVRAM (și ștergerea fișierului vlan.dat în cazul unui switch), se repornește dispozitivul pentru a șterge fișierul de config.re curentă din RAM. Apoi, dispozitivul va încărca config.ția de pornire implicită care a venit odată cu dispozitivul, în config.ția curentă.

2.5.4 ConFig.țile de rezervă cu capturarea textului

În plus față de salvarea conFig.ției curente în conFig.ția de pornire, fișierele de conFig.re pot fi salvate și arhivate într-un document text. Această secvență de pași asigură disponibilitatea unei copii a fișierelor de conFig.re, pentru modificare sau refolosire ulterioară.

Salvarea configurației într-un fișier text cu Tera Term

Fig. 2.19



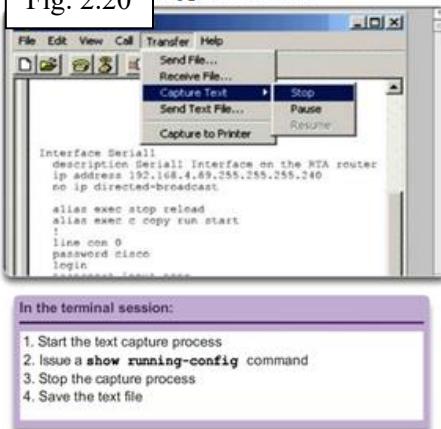
Pașii sunt:

- În meniul File, se apasă pe **Log**.
- Se alege locația. Tera Term va începe capturarea textului.
- După ce capturarea a fost pornită, se execută comanda **show running-config** sau **show startup-config** din modul privilegiat EXEC. Textul afișat în fereastra terminalului va fi stocat în fișierul ales.
- După completarea capturării, se apasă pe **Close** în fereastra TeraTerm: Log.
- Se deschide fișierul rezultat pentru a verifica dacă a fost corrupt sau nu.

Asemănător, fișierele pot fi salvate și arhivate într-un document text, folosind un HyperTerminal.

Salvarea configurației într-un fișier text cu HyperTerminal

Fig. 2.20



Restaurarea ConFig.ților Text – Un fișier de conFig.re poate fi copiat din memorie într-un dispozitiv. Când este copiat în terminal, IOSul execută fiecare linie din textul de conFig.re ca pe o comandă. Fișierul va necesita probabil modificări înainte de copiere. Este recomandată schimbarea parolelor criptate în text clar și ștergerea parametrului, fie numărul 5, fie 7, care

indică faptul că parola este criptată. Textul ce nu reprezintă o comandă, cum ar fi “--More--”, și mesajele IOS trebuie șterse.

Mai mult, în CLI, dispozitivul trebuie să fie în modul de conFig.re globală pentru a receptiona comenzi din fișierul text copiat.

În cazul utilizării TeraTerm, pașii sunt:

- *Se modifică textul pentru a șterge non-comenzile, și se salvează.*
- *În meniul **File**, se apasă pe **Send file**.*
- *Se localizează fișierul ce trebuie copiat în echipament și se apasă **Open**.*
- *TeraTerm va copia fișierul în echipament.*

Textul din fișier va fi aplicat sub formă de comenzi în CLI și va deveni conFig.ția curentă a echipamentului. Aceasta este o metodă convenabilă pentru conFigarea manuală a unui în echipament.

2.6 Schemele Adreselor . Porturi și Adrese

Utilizarea adreselor IP, fie IPv4, fie IPv6, este modalitatea principală de localizare a dispozitivelor și de stabilire a comunicațiilor între sursă și destinație în Internet. De fapt, în orice rețea, adresele IP sunt esențiale pentru dispozitive, pentru a comunica de la sursă la destinație și înapoi.

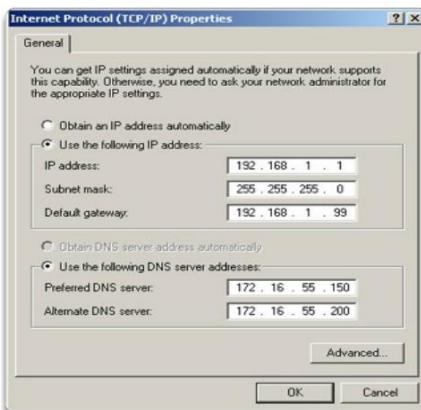
Fiecare dispozitiv final dintr-o rețea trebuie să fie conFig.t cu adrese IP. Câteva exemple de dispozitive finale sunt:

- *Calculatoarele (stațiile de lucru, laptopurile, serverele de fișiere, serverele web).*
- *Imprimantele de rețea.*
- *Telefoanele VoIP.*
- *Camerele de securitate.*
- *Telefoanele mobile.*
- *Dispozitivele portabile (precum scannerele de coduri de bare wireless).*

Structura unei adrese IPv4 este numită notația zecimală punctată și este reprezentată cu 4 numere zecimale cuprinse între 0 și 255. Adresele IPv4 sunt numere atribuite dispozitivelor individuale conectate la rețea. Acestea au o natură logică, în sensul că oferă informații despre locația dispozitivelor.

Odată cu adresa IP, este necesară și o mască de rețea. O mască de rețea este un tip special de adresă IPv4 care, împreună cu adresa IP, determină o anumită subrețea a unei rețele mari, din care face parte dispozitivul.

Adresele IP pot fi atribuite atât porturilor fizice, cât și interfețelor virtuale de pe dispozitive. O interfață virtuală înseamnă că nu există nici-o componentă hardware în dispozitiv, asociată cu aceasta.



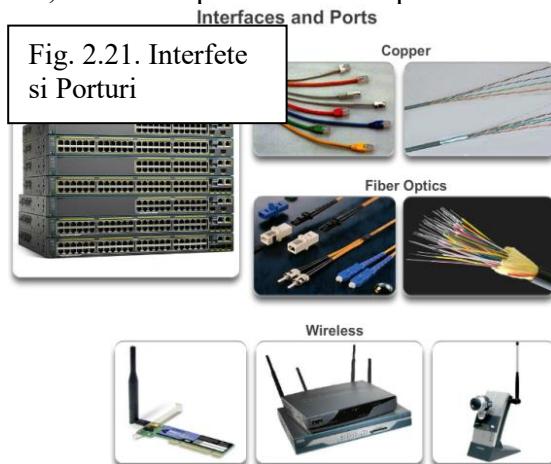
Comunicațiile în rețea depind de interfețele dispozitivelor finale, de interfețele dispozitivelor de rețea, și de cablurile care le conectează.

Fiecare interfață fizică are specificații, sau standarde, care o definesc; un cablu ce conectează interfață trebuie să fie proiectat pentru a îndeplini standardele fizice ale interfeței. Tipuri de medii de rețea includ cabluri de cupru twisted-pair, fibra optică, cabluri coaxiale, sau wireless. Fiecare mediu de rețea are anumite caracteristici și avantaje. Nu toate mediile de rețea au aceleași caracteristici și nu sunt adecvate pentru aceleași scopuri. Unele diferențe dintre tipurile de medii includ:

- *Distanța pe care un mediu de comunicație poate transmite un semnal cu succes.*
- *Mediul în care mediu de rețea va fi instalat.*
- *Volumul de date și viteza cu care poate fi transmis*
- *Costul mediului de comunicație și al instalației.*

Nu numai că fiecare legătură din Internet necesită un anumit tip de mediu de rețea, dar fiecare legătură necesită o anumită tehnologie de rețea. Ethernet este cea mai comună tehnologie LAN folosită astăzi. Porturile Ethernet se găsesc în dispozitivele finale, în switchuri, și în alte dispozitive de rețea care se pot conecta fizic la rețea, folosind un cablu. Pentru ca un cablu să conecteze dispozitive folosind un port Ethernet, FastEthernet, sau GigabitEthernet, el trebuie să aibă atașat conectorul corect, respectiv un conector RJ-45.

Switchurile Cisco IOS au porturi fizice la care dispozitivele se pot conecta, dar au și una sau mai multe interfețe virtuale (Switch Virtual Interface – SVI). Acestea sunt interfețe virtuale, deoarece nu există nici-o componentă hardware asociată cu ele; o SVI este creată prin software. Interfața virtuală oferă un mijloc de administrare de la distanță a unui switch, prin rețea folosind IPv4. Fiecare switch are o SVI, în mod implicit. SVIul implicit este interfața VLAN1.



2.6.1 Adresarea Dispozitivelor

Pentru a accesa un echipament intermidiar de nivel 2, sau de nivel 3 de la distanță, o adresă IP și o mască de rețea trebuie config. te pe o interfață a echipamentului:

- **Adresa IP** – Împreună cu masca de rețea, identifică în mod unic dispozitivul final în rețea
- **Masca de rețea** – Delimitiază adresa de rețea de adresele IP asignabile.

Pentru început studiem protocolul IPv4, iar mai târziu vom studia și protocolul IPv6.

În timp se dobândesc cunoștințele necesare înțelegerei semnificației adreselor IP, dar pentru moment scopul este de a învăța să config. rapid echipamentul pentru acces de la distanță. Pentru a realiza acest lucru vom prelua comenzile necesare pentru a activa conectivitatea IP atât pe un Switch și apoi pe un Router, folosind adresarea IP :

Pentru a configura un Router care să poată fi accesat de la distanță se parcurg pașii:

Pas 1. Se lansează comanda: #interface GigabitEthernet 0/0 – Folosită pentru a accesa modul de configurație al interfeței, din modul de configurație globală.

Pas 2. Se face o descriere a legăturii: #description "Legătura cu LAN 10"

Pas 3. Se asignează adresa: # ip address 192.168.10.1 255.255.255.0

Pas 4. Se activează interfața prin intermediul comenzi: no shutdown – Se trece interfață în starea activă.

Pentru a configura un Switch care să poată fi accesat de la distanță se parcurg pașii. ConFig.rea adresei IP și masca de rețea pentru switch (aceasta este una dintre posibilele combinații de adresă IP și mască de rețea):

Pas 1. Se lansează comanda: #interface vlan 1 – Folosită pentru a accesa modul de configurație al interfeței, din modul de configurație globală.

Pas 2. Se face o descriere a legăturii: #description "Legătura cu LAN 10"

Pas 3. Se asignează adresa: # ip address 192.168.10.2 255.255.255.0 – Configurează adresa IP și masca de rețea pentru switch (aceasta este una dintre posibilele combinații de adresă IP și mască de rețea)

Pas 4. Se activează interfața prin intermediul comenzi: no shutdown – Se trece interfață în starea activă.

După configurația acestor comenzi, echipamentele au toate elementele IP necesare pentru comunicarea în rețea.

Notă: Echipamentele vor avea nevoie de unul sau mai multe porturi fizice configurate, precum și linii VTY, pentru a completa configurația ce permite administrarea de la distanță a acestora. Doar multă practică va conduce la înțelegerea acestor pași de lucru.

Pentru ca un dispozitiv final să comunice într-o rețea, el trebuie configurat cu adresa IP corectă. Asemănător cu exemplele prezentate orice alt dispozitiv final trebuie configurat cu o adresă IP și o mască de rețea, informație ce este configurată prin setările PC-ului.

Toate aceste setări trebuie configurate pe un dispozitiv final pentru a se putea conecta, în mod corespunzător, la rețea. Aceste informații sunt configurate în setările de rețea ale PC-ului. În afară de adresa IP și de masca de rețea, este posibilă configurația „portii implicate – default gatewayului” și a serverului DNS.

Subliniem faptul că adresa default gateway este adresa IP a interfeței echipamentului folosit drept ieșire din rețea locală pentru trafic. Default gatewayul este o adresă IP atribuită de obicei de administratorul de rețea și este folosită când traficul trebuie să ajungă în altă rețea, respectiv adresa asignată interfeței routerului din exemplul de mai sus.

Adresa serverului DNS este adresa IP a serverului DNS, care este folosită pentru a translata adrese IP în adrese web, precum www.fmi.unibuc.ro. Toate dispozitivele din Internet au atribuite și sunt localizate printr-o adresă IP. Totuși, pentru oameni, este mai ușor de ținut minte nume în loc de numere. De aceea, siteurile web folosesc nume pentru simplitate. Serverul DNS este folosit pentru a menține maparea dintre adresele IP și numele diverselor dispozitive.

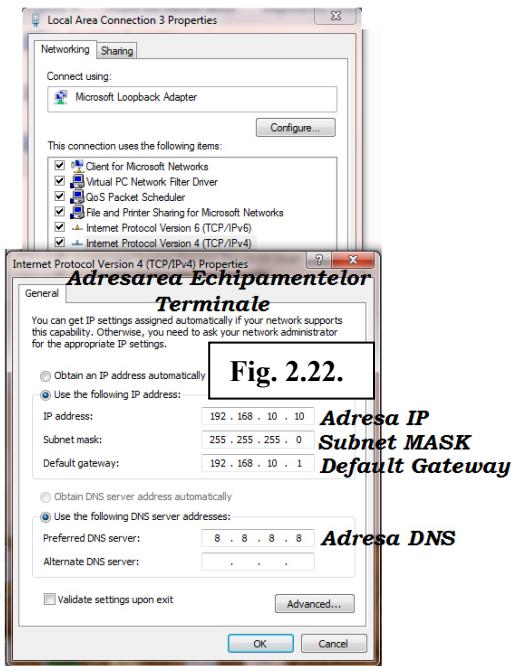


Fig. 2.22.

Adresa IP poate fi introdusă manual în PC, sau prin Dynamic Host Configuration Protocol DHCP. Serviciul DHCP permite dispozitivelor finale să aibă informațiile IP configurate automat.

DHCP este o tehnologie folosită în aproape toate rețelele de afaceri. Cea mai bună cale prin care se poate înțelege popularitatea DHCP-ului este prin considerarea muncii suplimentare care ar trebui făcută în lipsa lui.

DHCP permite configurația automată a adreselor IPv4 pentru fiecare dispozitiv final, într-o rețea cu DHCP activat. Să ne imaginăm cantitatea de timp consumată, dacă de fiecare dată când ne conectăm la rețea, ar trebui să introducem manual adresa IP, masca de rețea, default gateway-ul, și serverul DNS. Dacă se înmulțește această cantitate cu numărul de utilizatori și cu numărul dispozitivelor din rețea vom înțelege problema și utilitatea acestui serviciu.

DHCP este un exemplu de tehnologie în plin apogeu. Unul dintre scopurile principale ale oricărei tehnologii este de a ușura efectuarea sarcinilor pe care administratorul de rețea sau inginerul de sistem vor să le facă sau trebuie să le facă. Cu DHCP, utilizatorul final intră într-o zonă servită de o anumită rețea, introduce un cablu Ethernet sau activează o conexiune wireless, și va primi imediat toate informațiile IPv4 necesare pentru a comunica cu rețea respectivă.

După cum se poate observa în Fig. X, pentru a configura DHCP pe un PC cu Windows, trebuie doar selectată opțiunea “Obtain an IP address automatically” și de asemenea opțiunea “Obtain DNS server address automatically”. În urma acestor setări PC-ului îi vor fi atribuite informații dintr-o zonă de adrese IP, și informațiile IP asociate lui vor fi setate pe serverul DHCP.

Este posibilă afișarea setărilor de configurație IP pe un PC cu Windows prin folosirea comenzi **ipconfig** în promptul de comandă. Rezultatul va afișa adresa IP, masca de rețea, și gateway-ul pe care PC-ul le-a primit de la serverul DHCP.

Afișarea adresei IP a unui PC cu SO Windows prin introducerea comenzi are forma:

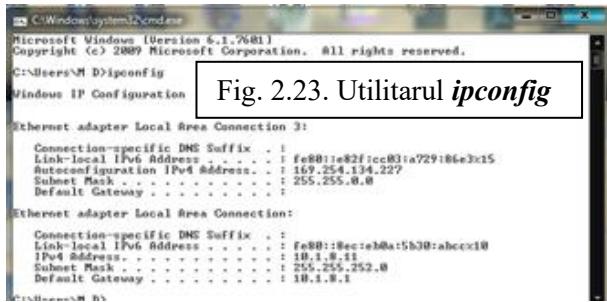


Fig. 2.23. Utilitarul *ipconfig*

Dacă o adresă IP statică (manuală) este definită pentru un dispozitiv de rețea, spre exemplu, o imprimantă, iar apoi este instalat un server DHCP, conflicte legate de adrese IP identice pot apărea între un dispozitiv de rețea și un PC care obține automat o adresă IP de la serverul DHCP. Conflictul poate apărea și în cazul în care este definită manual o adresă IP statică pentru un dispozitiv de rețea, în timp ce serverul DHCP nu este disponibil din cauza unor probleme; după rezolvarea problemelor, serverul DHCP devine disponibil în rețea și conflictul apare.

Pentru a rezolva un conflict de adrese IP se face convertirea dispozitivului de rețea cu adresa IP statică într-un client DHCP; sau pe serverul DHCP se exclude adresa IP statică a dispozitivului final din rândul de adrese alocate de DHCP.

A doua soluție necesită drepturi de administrator pe serverul DHCP și experiență cu configurația DHCP pe un server.

De asemenea, pot fi întâlnite conflicte de adrese IP când sunt definite manual adrese IP pe un dispozitiv final într-o rețea care folosește doar adrese IP statice. În acest caz, trebuie să se determine adresele IP disponibile dintr-o subrețea și să fie configurate corespunzător. Acest caz ilustrează importanța menținerii unei documentații detaliate a rețelei de către administrator, inclusiv atribuirea adreselor IP pentru dispozitivele finale.

Notă: De obicei adresele IP statice sunt folosite pentru servere, imprimante și alte dispozitive dedicate în rețele de afaceri mici și mijlocii, în timp ce dispozitivele angajaților folosesc adrese IP alocate de către DHCP.



Fig. 2.24. Output server DHCP

2.6.2 Verificarea conectivității

2.6.2.1 Testarea unei interfețe virtuale (Loopback)

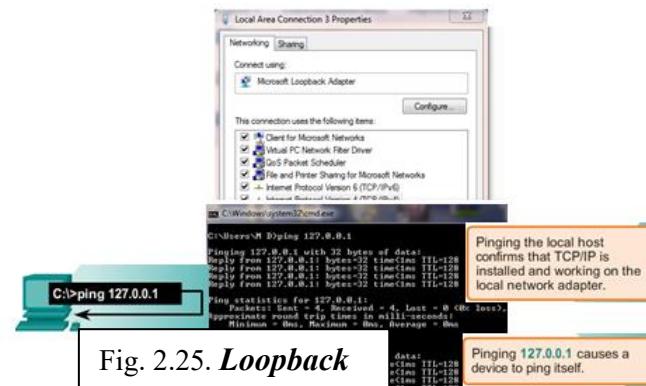


Fig. 2.25. *Loopback*

Fig. arată primul pas din secvență de testare. Comanda **ping** este folosită pentru a verifica configurația IP internă de pe o gazdă locală. Acest test este realizat prin folosirea comenzi **ping** pe o adresă rezervată numită loopback (127.0.0.1). Adresa loopback, 127.0.0.1, este definită de protocolul TCP/IP ca o adresă rezervată care trimit pachetele înapoi la gazdă.

Comenzile **ping** sunt introduse în linia comandă a gazdei locale, folosind sintaxa:

C:\> **ping 127.0.0.1**

Rezultatul acestei comenzi ar arăta astfel:

Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times în milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms

Rezultatul indică faptul că 4 pachete de test, a către 32 de octeți fiecare, au fost trimise și primite de la gazda 127.0.0.1 într-un timp mai mic de 1ms. Această cerere **ping** reușită verifică faptul că placa de rețea, driverele, și implementarea stivei TCP/IP funcționează corect.

În același mod în care au fost folosite comenzile și utilitarele pentru verificarea configurației unei gazde, se pot folosi comenzile pentru a fi verificate interfețele dispozitivelor intermediare. IOSul oferă comenzi pentru verificarea funcționalității interfețelor unui router sau ale unui switch.

2.6.2.2 Verificarea Interfețelor Switchului

Examinăm două switchuri S1 și S2, folosind comanda **show ip interface brief** pentru a verifica starea interfețelor switchurilor, aşa cum se poate observa:

Asignarea IP pe interfața VLAN

Enter the command to verify the interface configuration on S1.

Interface	IP-Address	OK?	Method	Status
FastEthernet0/1	unassigned	YES	manual	up
FastEthernet0/2	unassigned	YES	manual	up
<output omitted>				
Vlan1	192.168.10.2	YES	manual	up

You are now on S2. Enter the command to verify the interface configuration on S2.

Interface	IP-Address	OK?	Method	Status
FastEthernet0/1	unassigned	YES	manual	up
FastEthernet0/2	unassigned	YES	manual	up
<output omitted>				
Vlan1	192.168.10.3	YES	manual	up

Fig. 2.26.

Adresa IP atribuită interfeței VLAN1 pe S1 este 192.168.10.2. Adresa IP atribuită interfeței VLAN1 pe S2 este 192.168.10.3. Interfețele fizice F0/1 și F0/2 ale S1 sunt operaționale, precum și interfețele fizice F0/1 și F0/2 ale S2.

2.6.2.3 Testarea conectivității între PC și Switch

Comanda **ping** poate fi folosită pe un PC, ca pe un dispozitiv cu IOS și are aceeași formă ca pentru oricare alt echipament terminal sau intermediar. Un exemplu este prezentat mai jos cu un **ping** de la PC1 către adresa IP a interfeței VLAN1 a S1, 192.168.10.2, care ar trebui să funcționeze.

Testarea conectivității între PC și Switch

C:\> ping 192.168.10.2

Fig. 2.27.

```
Pinging 192.168.10.2 with 32 bytes of data:  
Reply from 192.168.10.2: bytes=32 time=838ms TTL=35  
Reply from 192.168.10.2: bytes=32 time=820ms TTL=35  
Reply from 192.168.10.2: bytes=32 time=883ms TTL=36  
Reply from 192.168.10.2: bytes=32 time=828ms TTL=36  
  
Ping statistics for 192.168.10.2:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 820ms, Maximum = 883ms, Average = 842ms
```

2.6.2.4 Testarea conectivității între două echipamente terminale

Adresa IP a PC1 este 192.168.10.10, cu masca de rețea 255.255.255.0, și default gateway 192.168.10.1., iar adresa IP a PC2 este 192.168.10.11, cu masca de rețea 255.255.255.0, și default gateway 192.168.10.1.

Un **ping** de la PC1 către PC2 ar trebui să funcționeze. Un **ping** reușit de la PC1 la PC2 verifică conectivitatea end-to-end într-o rețea !

Testarea conectivității între două echipamente terminale

Enter the command to verify connectivity to PC2 at '192.168.10.11'.

C:\> ping 192.168.10.11

Fig. 2.8.

```
Pinging 192.168.10.11 with 32 bytes of data:  
Reply from 192.168.10.11: bytes=32 time=838ms TTL=35  
Reply from 192.168.10.11: bytes=32 time=820ms TTL=35  
Reply from 192.168.10.11: bytes=32 time=883ms TTL=36  
Reply from 192.168.10.11: bytes=32 time=828ms TTL=36  
  
Ping statistics for 192.168.10.11:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 820ms, Maximum = 883ms, Average = 842ms
```

2.7 Concluzii Capitolul 2

IOS este un termen ce cuprinde un număr de sisteme de operare, care rulează pe diferite dispozitive de rețea. Tehnicianul poate introduce comenzi pentru a configura, sau programa, dispozitivul pentru a efectua diverse funcții în rețea. Routerele și switchurile cu IOS efectuează funcții pe care profesioniștii de rețea se bazează pentru a face rețelele să funcționeze într-un anumit mod.

Serviciile oferite de IOS sunt în general accesate folosind o interfață cu linie de comandă (CLI), care este accesată fie prin portul consolă, portul AUX, sau prin telnet sau SSH. Odată conectați la CLI, tehnicienii de rețea pot schimba configurația dispozitivelor cu IOS. IOS este

proiectat ca un sistem de operare modal, ceea ce înseamnă că un tehnician de rețea trebuie să navigheze prin multe moduri ierarhice ale IOS. Fiecare mod suportă diferite comenzi IOS.

IOS Command Reference este o colecție de documente online care descriu în detaliu comenzile IOS folosite pe dispozitive, precum routere sau switchuri cu IOS.

Routerele și switchurile cu IOS suportă un sistem de operare modal similar, suportă structuri de comenzi similare, și suportă multe comenzi identice. Mai mult, ambele dispozitive au pași de configurație inițială identici, când sunt implementate într-o rețea.

Acest capitol a introdus concepte destre IOS, a detaliat diversele moduri ale IOS și a examinat structura de bază a comenzilor, ce este folosită pentru a-l configura.. De asemenea, a parcurs setările inițiale ale unui switch cu IOS, inclusiv setarea unui nume, limitarea accesului la configurația dispozitivului, configurația mesajelor banner, și salvarea configurației.

Următorul capitol explorează modul în care pachetele sunt transmise într-o rețea și va introduce regulile comunicării pachetelor.

Structura Ierarhica a Modurilor CLI



CAPITOLUL 3. PROTOCOALE DE REȚEA

Introducere

Din ce în ce mai mult rețelele ne conectează. Oamenii comunică online de pretutindeni. Conversațiile dintr-o sală de clasă se împart în sesiuni de mesagerie instant și dezbatere online ce continuă în școală. Noi servicii sunt dezvoltate zilnic pentru a utiliza rețea.

Decât să dezvolte sisteme unice și separate de livrare a fiecărui serviciu nou, industria de rețelistică a adoptat un cadru de dezvoltare ce permite designerilor să întreagă platformele curente de rețele și să le gestioneze. În același timp, acest cadru este utilizat pentru a facilita dezvoltarea de noi tehnologii pentru a suporta nevoi de comunicații viitoare și îmbunătățiri tehnologice.

În centrul acestui cadru de dezvoltare este utilizarea modelelor generale acceptate ce descriu regulile și funcțiile rețelei.

În acest capitol, vom învăța despre aceste modele, precum și standardele ce fac posibilă funcționarea rețelei și modul în care comunicarea are loc peste rețea.

3.1 Conceptele fundamentale referitoare la aceste reguli și funcții

Pornim în această activitate de la un fapt cotidian. Să presupunem că o persoană tocmai a achiziționat un nou autoturism pentru uz personal. După ce conduce mașina o săptămână, află că aceasta nu funcționează corect.

După ce discută problemele cu mai multe cunoștințe, decide să apeleze la o soluție de reparare de automobile ce este recomandată de către amici. Este singura posibilitate de reparare localizată în apropierea să.

Când ajunge la punctul de reparare, observă că mecanicii vorbesc altă limbă. Are dificultăți în explicarea problemelor de performanță ale autoturismului, dar mecanicii solicită aceste informații pentru a-și putea face treaba. În aceste condiții nu este sigur că îl poate conduce înapoi acasă pentru a căuta alte opțiuni.

Trebuie să găsească o modalitate de lucru cu posibilitatea de reparare pentru a se asigura de faptul că autoturismul este reparat.

Cum va comunica cu mecanicii din firmă ?

Cerință. Să se proiecteze un model de comunicare pentru asigurarea faptului că mașina este reparată în mod corespunzător.



3.2 Reguli de comunicare

3.2.1 Regulile

O rețea poate fi complexă precum dispozitivele conectate în Internet sau simplă precum două calculatoare conectate direct unul cu celălalt cu un singur cablu și “fără nimic între ele”. Rețelele pot varia în dimensiune, formă și funcție. Oricum, simpla conexiune fizică dintre dispozitivele finale nu este suficientă pentru permiterea comunicației. Pentru ca aceasta să aibă loc, dispozitivele trebuie să știe cum să comunice.

Oamenii fac schimb de idei prin intermediul multor metode de comunicare diferite. Oricum, indiferent de metoda aleasă, toate metodele de comunicare au trei elemente în comun.

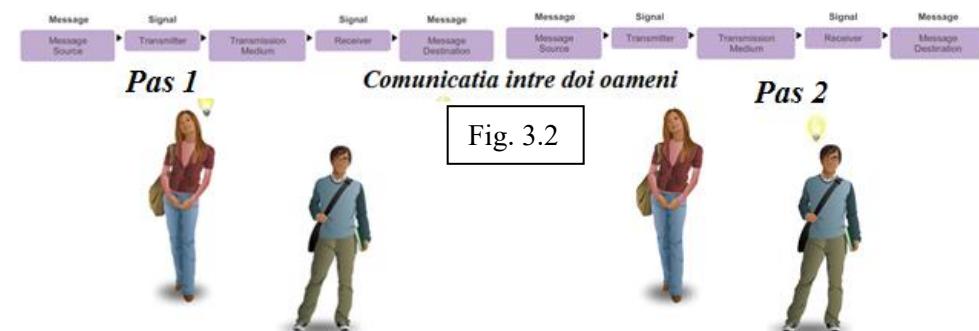
1. Primul element este **sursa** mesajului, sau expeditorul. Sursele mesajului sunt oameni sau dispozitive electronice care necesită să transmită un mesaj către alți indivizi sau dispozitive.

2. Al doilea element este **destinația** – entitatea care primește mesajul și îl interpretează.

3. Al treilea element, numit **canal=mediul de comunicație**, constă din mediul ce oferă o cale prin care mesajul călăorește de la sursă la destinație.

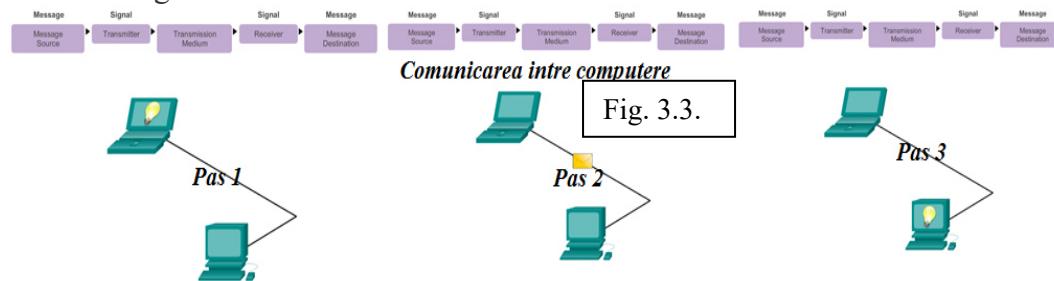
Comunicarea începe cu un mesaj, sau o informație, care trebuie să fie transmisă de la o sursă la o destinație. Transmiterea acestui mesaj, indiferent dacă este efectuată față-în-față sau peste rețea, este guvernată de **reguli** numite **protocole**. Aceste protocoale sunt specifice tipului de metodă de comunicare ales. În comunicarea personală de zi cu zi, regulile folosite de oameni pentru a comunica peste mediu, cum ar fi un apel telefonic, nu sunt neapărat aceleași cu protocoalele folosite de alt mediu, cum ar fi transmiterea unei scrisori.

De exemplu, să luăm în considerare doi oameni ce comunică față-în-față, ca în Fig. xxx1.



Înainte de a comunica, aceștia trebuie să stabilească modul în care comunică. Dacă comunicarea se face prin voce, trebuie să stabilească în ce limbă. Apoi, când au un mesaj de împărtit, trebuie să fie capabili să formeze mesajul într-un mod în care să fie înțeles. De exemplu, dacă cineva utilizează limba engleză, dar cu o structură săracă a frazei, mesajul poate fi ușor neînțeles. Fiecare dintre aceste sarcini descriu protocoale stabilite pentru realizarea comunicării. Acest lucru este valabil și la comunicare prin calculator, ilustrată în Fig. xxx2.

Pentru o înțelegere mai rapidă și mai ușoară să ne gândim la modul în care multe reguli și protocoale diferite guvernează toate metodele diferite de comunicare existente în lumea de astăzi.



3.2.2 Stabilirea regulilor

Înaintea comunicării dintre ei, indivizii trebuie să stabilească reguli sau înțelegeri ce guvernează conversația. De exemplu, considerând Fig. xxx1, protocolele sunt necesare pentru o comunicare eficientă. Protocolele utilizate sunt specifice caracteristicilor metodei de comunicare, inclusive caracteristicilor sursei, destinației și canalului. Aceste reguli, sau protocole, trebuie să fie următe pentru ca mesajul să fie livrat cu succes și înțeles. Există mai multe protocole disponibile pentru a guverna comunicarea umană de succes. O dată ce există o înțelegere cu privire la metoda de comunicare (față-în-față, telefon, scrisoare, fotografie) protocolele stabilite trebuie să țină cont de următoarele cerințe:

- Un **expeditor = sursa** și un **receptor = destinația** identificat.
- Limba și gramatica comune.
- Viteza și timpul de livrare.
- Cerințele de confirmare sau răspuns.

Protocolele utilizate în comunicațiile din rețea împart multe dintre trăsăturile fundamentale cu acele protocole folosite pentru guvernarea conversațiilor umane, conform Figurii xxx2. Pentru a identifica sursa și destinația, protocolele de rețea și computer definesc detaliile modului în care mesajul este transmis de-a lungul rețelei pentru a întâlni cerințele de mai sus. Deși există mai multe protocole ce trebuie să interacționeze, protocolele comune de computer includ:

- *Codarea mesajului.*
- *Formatarea și încapsularea mesajului.*
- *Dimensiunea mesajului.*
- *Sincronizarea mesajului.*
- *Opțiunile de livrare ale mesajului.*

Fiecare dintre acestea vor fi discutate detaliat pe întreaga desfășurare a cursului.

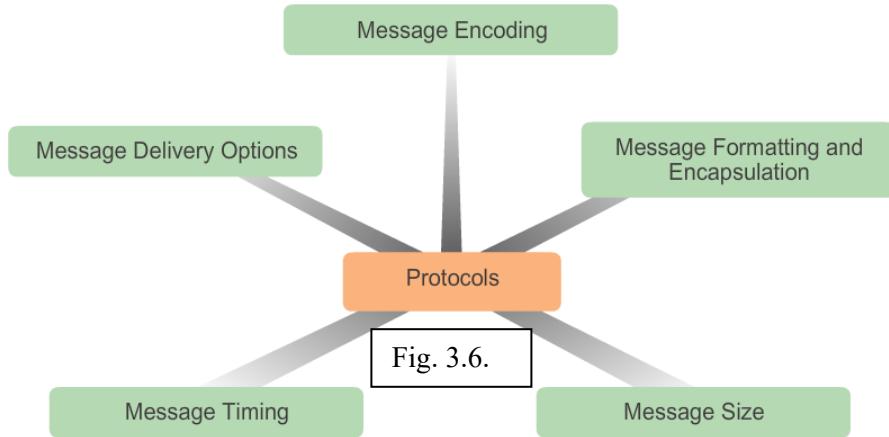
humans communication between govern rules.

It is very difficult to understand messages that are not correctly formatted and do not follow the established rules and protocols. A estrutura da gramática, da língua, da pontuação e da sentença faz a configuração humana compreensível por muitos indivíduos diferentes.

Fig. 3.4.

Rules govern communication between humans. It is very difficult to understand messages that are not correctly formatted and do not follow the established rules and protocols. The structure of the grammar, the language, the punctuation and the sentence make the configuration humanly understandable for many different individuals.

Fig. 3.5.

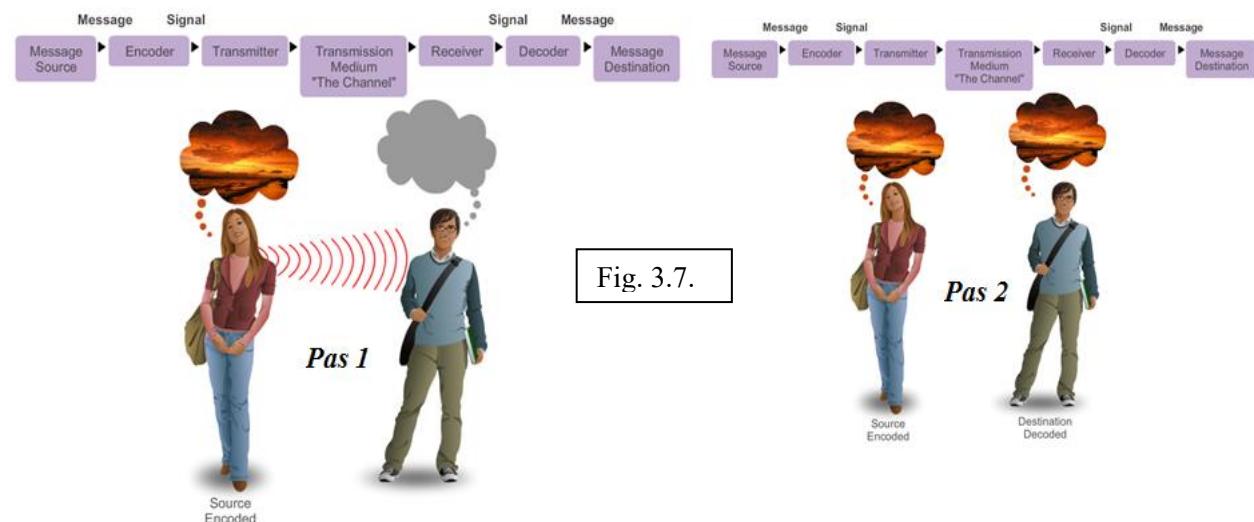


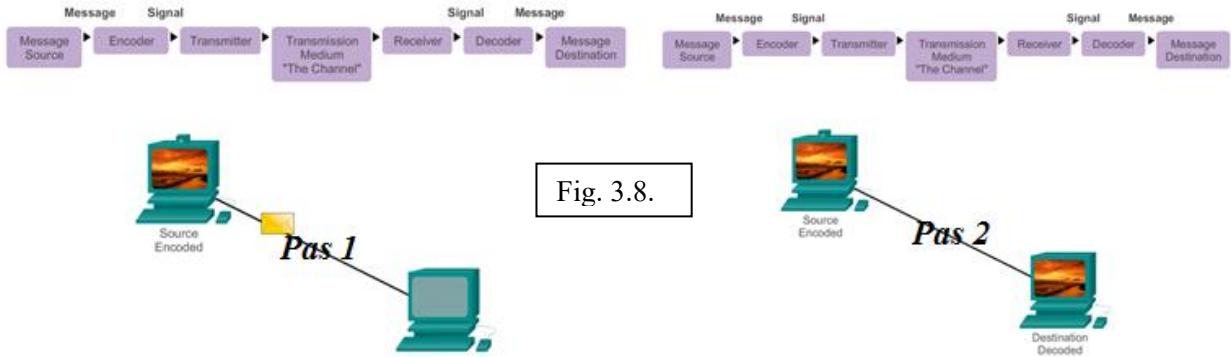
3.2.3 Codarea mesajului

Unul dintre primii pași din transmiterea unui mesaj este codarea lui. Codarea este procesul de conversie a informației într-o altă formă admisibilă pentru transmisie. Decodarea este procesul invers folosit pentru interpretarea informației.

Să ne imaginăm o persoană ce își planifică o excursie de vacanță cu un prieten și își sună prietenul pentru a discuta detaliile privind locul în care să meargă, conform Figurii xxx1. Pentru a comunica mesajul, expeditorul trebuie mai întâi să convertească, sau să codifice, gândurile sale și percepțiile despre locație în cuvinte. Cuvintele sunt vorbite în telefon utilizând sunetele și inflexiunile din limba vorbită care este utilizată pentru a transmite mesajul. La celălalt capăt al liniei de telefon, persoana ascultă descrierea, primește și decodifică sunetele pentru a vizualiza Fig. descrisă de către expeditor.

Codarea are loc, de asemenea, și în comunicarea prin intermediul computerului, conform Figurii xxx2. Codarea dintre hosturi trebuie să se facă într-o formă adecvată mediului de comunicare. Mesajele transmîte prin rețea sunt mai întâi convertite în biți de către hostul expeditor. Fiecare bit este codat în şabloane de sunete, unde luminoase sau impulsuri electrice, în funcție de mediul de rețea peste care biții sunt transmiși. Destinația trebuie să primească și să decodifice semnalele pentru a interpreta mesajul.





3.2.4 Formatarea și încapsularea mesajului

Atunci când un mesaj este transmis de la sursă la destinație, trebuie să folosească o structură sau un format specific. Formatele mesajului depind de tipul de mesaj și de canalul utilizat pentru a transmite mesajul.

Scrierea de scrisori este una dintre cele mai comune forme de comunicare scrisă de rasa umană. De secole, formatul convenit pentru scrisorile personale nu s-a schimbat. În multe culturi, o scrisoare personală conține următoarele elemente:

- *Un identificator al receptorului.*
- *Un salut sau o întâmpinare.*
- *Conținutul mesajului.*
- *O frază de încheiere.*
- *Un identificator al expeditorului.*

Pentru a avea un format corect, multe scrisori personale trebuie să fie închise, sau încapsulate, într-un plic pentru a fi livrate, conform Figurii xxx1. Plicul conține adresa expeditorului și cea a receptorului pe el, fiecare localizată într-un loc adecvat pe plic. Procesul de plasare a unui format de mesaj (scrisoarea) în interiorul altui format de mesaj (plicul) se numește încapsulare. Decapsularea are loc la inversarea procesului, atunci când destinatarul scoate scrisoarea din plic.

Un om care scrie scrisori utilizează un format adecvat pentru a se asigura de faptul că scrisoarea este livrată și înțeleasă de către destinatar. În același mod, un mesaj transmis prin intermediul rețelei de calculatoare urmează reguli specifice de formatare pentru a fi livrate și procesate. Așa cum și scrisoarea este încapsulată într-un plic pentru livrare, și mesajele din computer sunt încapsulate. Fiecare mesaj electronic este încapsulat într-un format specific, numit frame (cadru), înainte de a fi transmis peste rețea. Un cadru se comportă ca un plic; oferă adresa destinației și adresa sursei, conform Figurii xxx2.

Conținutul și formatul unui cadru sunt determinate de tipul mesajului ce urmează transmis și de canalul peste care este livrat. Mesajele care nu sunt formatare corect nu sunt livrate cu succes sau procesate de către gazda destinație.



Recipient (destination) Location address	Sender (source) Location address	Salutation (start of message indicator)	Recipient (destination) identifier	Content of Letter (encapsulated data)	Sender (source) identifier	End of Frame (End of message indicator)
Envelope Addressing		Encapsulated Letter				
1400 Main Street Canton, Ohio 44203	4085 SE Pine Street Ocala, Florida 34471	Dear	Jane	I just returned from my trip. I thought you might like to see my pictures.	John	

Fig. 3.10.

Destination (physical / hardware address)	Source (physical / hardware address)	Start Flag (start of message indicator)	Recipient (destination identifier)	Sender (source identifier)	Encapsulated Data (bits)	End of Frame (end of message indicator)
Frame Addressing		Encapsulated Message				

Fig. 3.11.

3.2.5 Dimensiunea Mesajului

O altă regulă a comunicării este dimensiunea. Atunci când oamenii comunică unii cu ceilalți, mesajele trimise sunt, în mod normal, sparte în părți mai scurte sau în propoziții. Aceste propoziții sunt limitate în dimensiune pentru ca persoana destinatară să le proceseze dintr-o dată, conform Figurii xxx1. O conversație individuală poate fi alcătuită din multe propoziții scurte

pentru a asigura faptul că fiecare parte a mesajului este primită și înțeleasă. Să ne gândim cum ar fi să citim acest curs dacă totul ar apărea sub formă unei propoziții lungi; nu ar fi ușor de citit și mai ales de înțeles.

De asemenea, atunci când un mesaj lung este transmis de la un host la altul peste rețea, este necesară “spargerea” mesajului în bucăți mai mici, conform Figurii xxx2. Regulile ce guvernează dimensiunea bucăților, sau cadrelor, transmise prin rețea sunt foarte stricte. Pot fi, de asemenea, diferite în funcție de canalul folosit. Cadrele care sunt prea lungi sau prea scurte nu sunt livrate.

Restricțiile de dimensiune ale cadrelor necesită ca hostul sursă să “spargă” un mesaj lung în bucăți individuale pentru a întâlni cerințele de dimensiune minimă și maximă. Acest lucru este cunoscut ca **segmentare**. Fiecare segment este încapsulat într-un frame separat cu informații despre adresă și este transmis peste rețea. La hostul destinație, mesajele sunt decapsulate și unite pentru a fi procesate și interpretate.



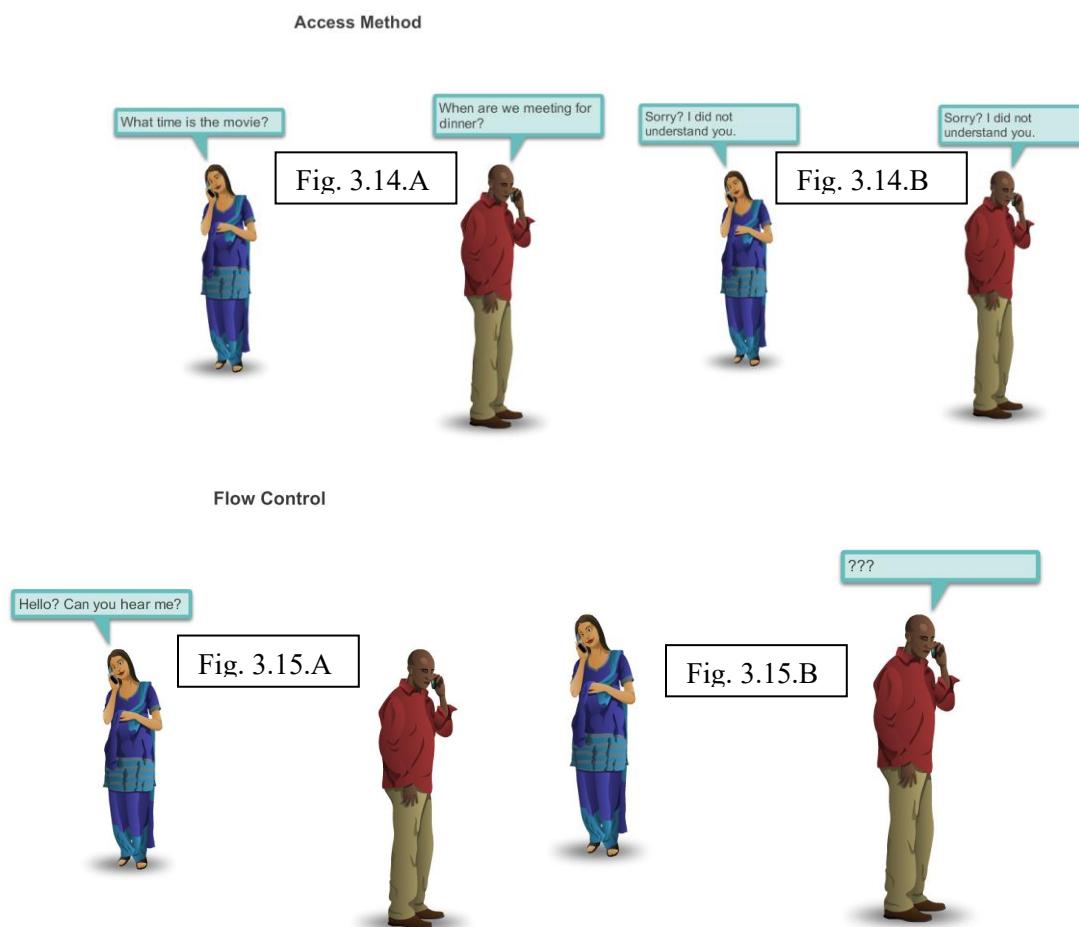
Sincronizarea mesajului – Un alt factor ce influențează modul în care mesajul este receptat și înțeles este sincronizarea. Oamenii folosesc sincronizarea pentru a determina când să vorbească, cât de repede sau cât de încet și cât de mult să aștepte un răspuns. Acestea sunt regulile necesare sincronizării.

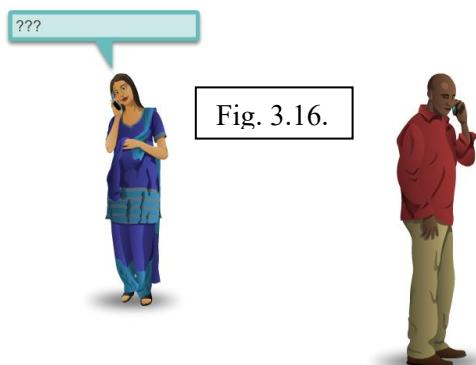
Metoda de acces – Metoda de acces determină momentul în care cineva poate transmite un mesaj. Aceste reguli de timp sunt bazate pe mediu. De exemplu, oricine este capabil să vorbească

oricând are ceva de spus. În acest mediu, o persoană trebuie să aștepte până când nimeni altcineva nu vorbește, înainte de a primi acceptul de a vorbi. Dacă doi oameni vorbesc în același timp, o coliziune de informații are loc și este necesar ca cei doi să se retragă și să înceapă din nou, conform Figurii xxx1. De asemenea, este necesar pentru calculatoare să definească o metodă de acces. Hosturile dintr-o rețea au nevoie de o metodă de acces pentru a ști când să înceapă să transmită mesaje și cum să răspundă în cazul în care apare o eroare.

Controlul fluxului – Timpul afectează, de asemenea, câtă cantitate de informații poate fi transmisă și viteza cu care poate să se transmită. Dacă o persoană vorbește prea rapid, este dificil pentru celalătă persoană să asculte și să înțeleagă mesajul, conform Figurii xxx2. Persoana receptor trebuie să îi ceară expeditorului să încetinească. În comunicația din rețea, un host expeditor poate transmite mesaje cu o rată mai mare decât hostul destinatar poate să le primească și să le proceseze. Hosturile sursă și destinație utilizează controlul fluxului pentru a negocia timpul corect pentru o comunicare de succes.

Timpul de răspuns – Dacă o persoană pune o întrebare și nu aude un răspuns într-o perioadă de timp, persoana presupune că nu există nici-un răspuns și reacționează conform acestui fapt, la fel ca în Fig. xxx3. Persoana ar putea repeta întrebarea sau să meargă mai departe cu conversația. Hosturile din rețea au de asemenea reguli care specifică cât timp să se aștepte răspunsurile și ce acțiune să se efectueze în cazul în care are loc un "timeout" de răspuns.

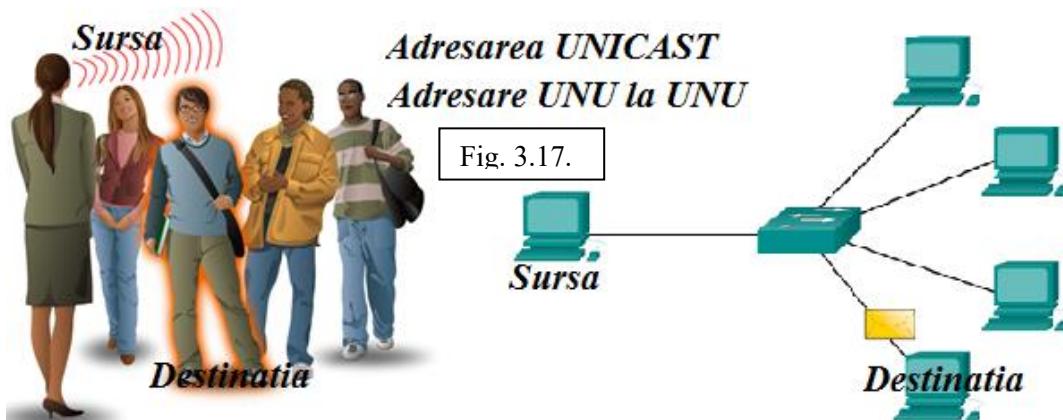




3.2.6 Opțiuni de livrare a mesajului

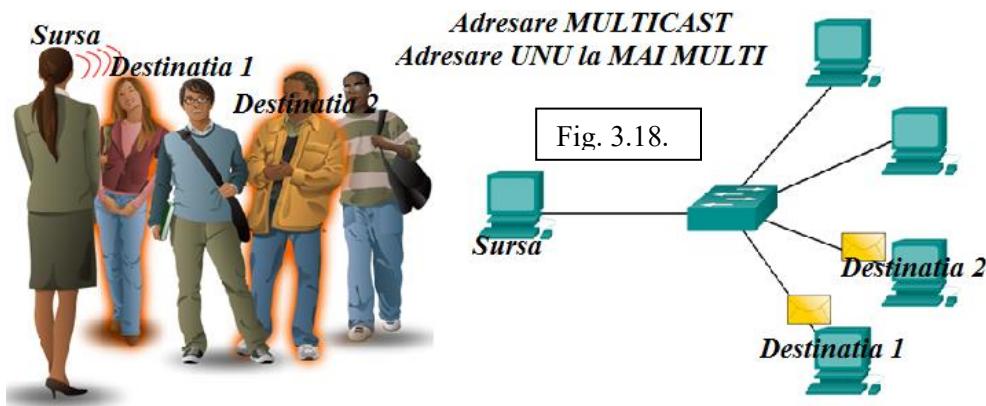
Un mesaj ar putea necesită să fie livrat în mai multe moduri.

Opțiunea 1. Adresarea Unicast – atunci când o persoană vrea să comunice informații către un singur individ. O conversație dintre doi oameni este un exemplu de livrare unu-la-unu. O opțiune de livrare unu-la-unu ne este referită ca unicast, însemnând că există o singură destinație a mesajului.

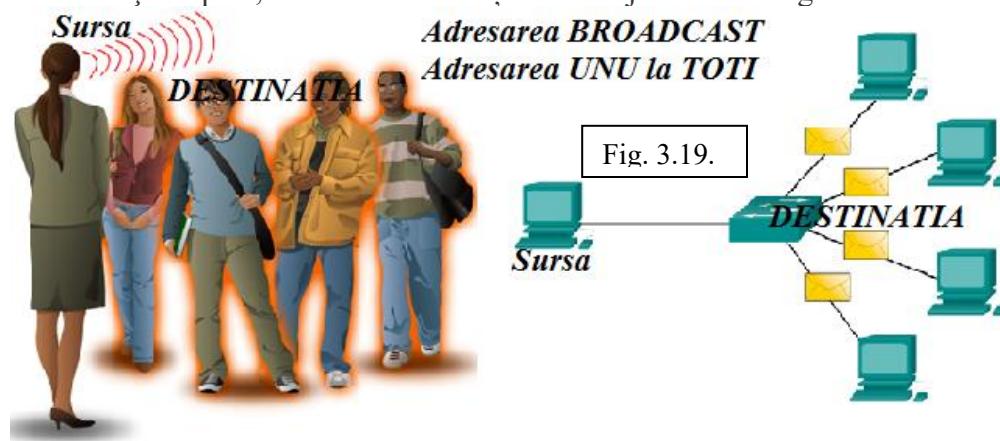


Opțiunea 2. Adresarea Multicast – atunci când o persoană dorește să transmită informații către un grup de persoane în același timp sau chiar tuturor oamenilor din aceeași arie. Atunci când un grup de destinatari necesită să recepționeze același mesaj simultan, o livrare de mesaj unu-la-mai mulți sau unu-la-toți este necesară. Atunci când un host necesită să transmită mesajele utilizând o opțiune de livrare unu-la-mai mulți, se numște multicast. Multicastul este transmiterea unui mesaj la un grup de hosturi destinație, în mod simultan.

Există de asemenea momente când expeditorul unui mesaj necesită să se asigure de faptul că mesajul a fost livrat cu succes la destinație. În aceste cazuri, este necesar ca destinatarul să trimită un "acknowledgement" expeditorului. Dacă nu este necesar un acknowledgement, opțiunea de livrare este numită "unacknowledged".



Opțiunea 3. Adresarea Broadcast – se realizează acunci când toate hosturile din rețea necesită să primească mesajul în același timp. Forma de adresare Broadcast reprezintă opțiunea de livrare unu-la-toți. În plus, hosturile au cerințe de mesaje acknowledged sau unacknowledged.



3.3 Standarde și protocole de rețea

3.3.1 Protocole

Ca și în comunicarea umană, protocole diverse de rețea și computerele trebuie să fie capabile să interacționeze și să lucreze împreună pentru a se realiza o comunicare peste rețea cu succes. Un grup de protocole inter-relaționale necesare pentru a îndeplini o funcție de comunicare se numește set (suită) de protocole. Seturile de protocole sunt implementate de către hosturi și dispozitivele de rețea în software, hardware sau ambele.

Unul dintre cele mai bune moduri de vizualizare a modului în care protocolele dintr-o suită interacționează este vizualizarea interacțiunii ca o stivă. O stivă de protocole arată modul în care protocolele individuale dintr-o suită sunt implementate. Protocolele sunt văzute în termeni de nivele, cu fiecare serviciu de nivel superior depinzând de funcționalitatea definită de către protocolele aflate în nivelele inferioare. Nivelele inferioare ale stivei se ocupă de mutarea datelor peste rețea și oferirea de servicii pentru nivele superioare, ce sunt axate pe conținutul mesajului transmis. Așa cum se poate observa Fig. xxx, putem folosi nivelele pentru a descrie activitatea ce are loc în exemplul de comunicare față-în-față. La nivelul de bază, nivelul fizic, avem doi oameni, fiecare ce poate spune cuvinte cu voce tare. La al doilea nivel, nivelul de reguli, avem o înțelegere de a vorbi aceeași limbă. La nivelul superior, nivelul de conținut, există cuvintele vorbite. Acesta este conținutul comunicării.

Protocolele = Reguli care Guverneaza Comunicatiile

Fig. 3.20.

Cum comunicam ?

Continutul Nivelului

Suita Protocolelor pentru Conversatie

- 1. Foloseste un limbaj comun Regulile Nivelului**
- 2. Asteapta initializarea**
- 3. Semnalaizeaza terminarea**



Suita de Protocole reprezinta seturi de reguli care impreuna ajuta la rezolvarea problemelor

Când asistă la această conversație, oamenii nu văd nivelele plutind în spațiu. Utilizarea de nivele este un model ce oferă un mod de împărțire în mod convenabil a unei sarcini complexe în părți și de descriere a modului în care acestea lucrează.

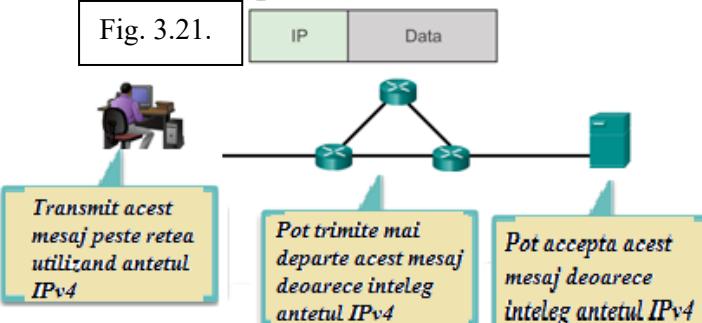
La nivel uman, unele regule de comunicare sunt formale și altele sunt înțelese simplu prin punerea în practică. Pentru ca dispozitivele să comunice cu succes, un set de protocole de rețea trebuie să descrie precis cerințele și interacțiunile. Protocolele de rețea definesc un format comun și un set de regule pentru schimbul de mesaje între dispozitive. Unele dintre protocolele de rețea sunt IP, HTTP, DHCP etc.

Figurile xxx ilustrează protocolele de rețea ce descriu următoarele procese:

- Modul în care mesajul este formatat sau structurat, conform Figurii xxx1

Scopul Protocolelor

Fig. 3.21.



- Procesele prin care dispozitivele de rețea împart informații despre căile de acces cu alte rețele, conform Figurii xxx2.

Procesele prin care echipamentele de retea împart informații despre caile de acces și alte informații despre retele

Totii suntem de acord că, dacă una sau mai multe din căile de mai jos devine inactivă, vom informa toate dispozitivele conectate

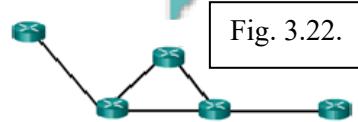
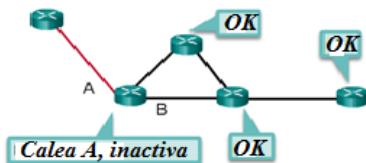


Fig. 3.22.



- Modul în care și când mesajele de sistem și eroare sunt transmise între dispozitive, conform Figurii xxx3.

Cum și când mesaje de eroare și de sistem sunt transmise între dispozitive

Convenim cu totii că mesajele de eroare vor avea un număr unic de identificare

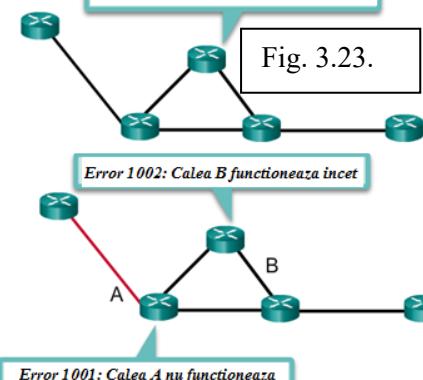
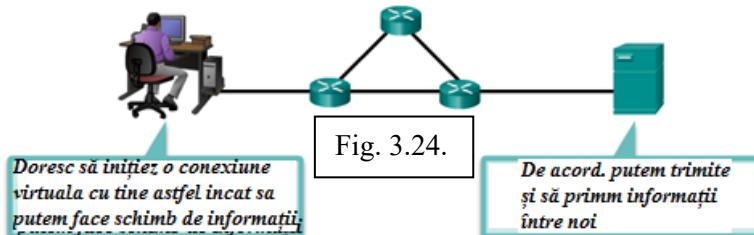


Fig. 3.23.

- Setarea și terminarea sesiunilor de transfer de date, conform Figurii xxx4.

Inițierea și închiderea sesiunii de transfer de date

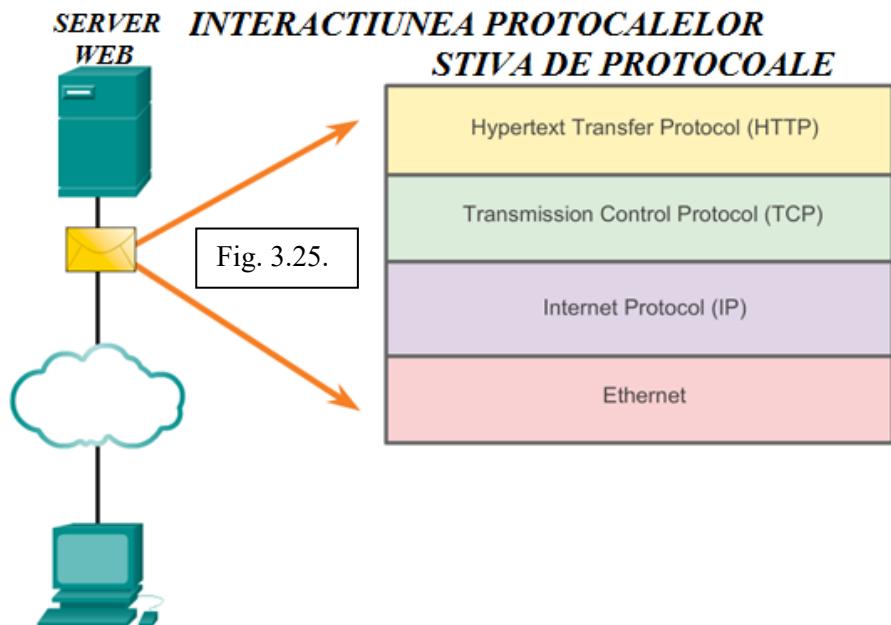


De exemplu, IP definește modul în care un pachet de date este livrat în rețea sau printr-o rețea de la distanță. Informațiile din protocolul IPv4 sunt transmise într-un format specific pentru ca destinatarul să poată să le interpreteze corect. Acest lucru nu este cu mult diferit de protocolul folosit pentru a adresa un plic atunci când este transmisă o scrisoare. Informațiile trebuie să preia un format adecvat sau scrisoarea nu poate fi livrată la destinație prin oficiul poștal.

Un exemplu de utilizare a suitei de protocoale în comunicațiile de rețea este interacțiunea dintre un server web și un client web. Această interacțiune folosește un număr de protocoale și

standarde în procesul de schimb de informații dintre aceastea. Protocole diferite lucrează împreună pentru a asigura că mesajul este receptat și înțeles de ambele părți. Exemple de aceste protocole sunt:

- **Protocolul aplicație** - Hypertext Transfer Protocol (HTTP) este un protocol ce guvernează modul în care un server web și un server client interacționează. HTTP definește conținutul și formatul cererilor și răspunsurilor schimbate între client și server. Softwareul client și server implementează HTTP ca parte a aplicației. HTTP se bazează pe alte protocoale ce guvernează modul în care mesajele sunt transmise între server și client.
- **Protocolul transport** - Transmission Control Protocol (TCP) este protocolul de transport care gestionează conversațiile individuale dintre serverele web și serverele clienți. TCP divide mesajele HTTP în bucăți mici, numite segmente. Aceste segmente sunt trimise între procesele de server web și client ce rulează pe hostul destinație. TCP este de asemenea responsabil de dimensiunea și rata cu care mesajele sunt transmise între server și client.
- **Protocolul de Internet** – IP este responsabil de preluarea segmentelor de la TCP, încapsularea lor în pachete și atribuirea acestora adrese adecvate și livrarea lor pe cea mai bună cale spre destinație.
- **Protocole de acces la rețea** – Protocolele de acces la rețea descriu două funcții importante, comunicarea peste o legătură de date și transmiterea fizică a datelor peste mediul de rețea. Protocolele de management al legăturii de date iau pachetele de la IP și le formatează pentru a fi transmise peste mediu. Standardele și protocolele de la mediul fizic guvernează modul în care semnalele sunt transmise și interpretate de către clienții destinații. Un exemplu de protocol de acces la rețea este Ethernet.



3.3.2 Suite de protocoale

Se poate afirma că - suita de protocoale reprezintă un set de protocoale ce lucrează împreună pentru a oferi servicii de comunicare de rețea complexe. O suita de protocoale ar putea fi specificată de către o organizație de standarde sau dezvoltată de către un furnizor.

Protocolele IP, HTTP și DHCP sunt toate părți ale suitei de protocoale Internet, cunoscută ca Transmission Control Protocol/IP (TCP/IP). Suta de protocoale TCP/IP este un standard deschis,

însemnând că aceste protocole sunt disponibile public sau oricărui furnizor ce este capabil să implementeze aceste protocole pe hardwareul sau softwareul propriu.

Un protocol bazat pe standard este un proces sau un protocol ce a fost aprobat de către industria de rețele și aprobat de către organizația de standarde (International Organization for Standardization – ISO). Utilizarea standardelor în dezvoltarea și implementarea protocolelor asigură faptul că produsele de diferite fabricații pot interacționa cu succes. Dacă un protocol nu este observant atent de către un fabricant particular, echipamentul lor sau softwareul ar putea să nu fie capabil să comunice cu succes cu produsele de la alți fabricanți.

În comunicațiile de date, de exemplu, dacă un capăt al conversației utilizează un protocol ce guvernează comunicația într-un singur sens și celălalt capăt adoptă un protocol ce descrie comunicarea în ambele sensuri, cu siguranță, datele nu vor fi schimbate între ei.

Unele protocole sunt particulare, particulare în acest context, înseamnă că o companie sau un furnizor controlează definirea unui protocol și funcționalitatea să. Unele protocole particulare pot fi utilizate de către organizații diferite cu permisiunea proprietarului. Altele pot fi implementate numai pe echipamente fabricate de către furnizorul proprietar. Exemple de protocole proprietare sunt AppleTalk și Novell Netware.

Multe companii ar putea chiar să lucreze împreună pentru a crea un protocol particular. Nu este neobișnuit pentru un furnizor (sau un grup de furnizori) să dezvolte un protocol particular pentru a întări nevoile clientilor și pentru a ajuta mai târziu ca un protocol particular să devină un standard deschis. De exemplu, Ethernet a fost un protocol dezvoltat de către Bob Metcalfe în XEROX Palo Alto Research Center (PARC) în 1970. În 1979, Bob Metcalfe și-a format propria companie, 3COM și a lucrat cu Digital Equipment Corporation (DEC), Intel și Xerox pentru a promova standardul DIX pentru Ethernet. În 1985, Institute of Electrical and Electronics Engineers (IEEE) a publicat standardul IEEE 802.3 aproape identic cu Ethernet. Astăzi, 802.3 este standardul comun utilizat în LANuri. Un alt exemplu, cel mai recent, CISCO a dezvoltat protocolul EIGRP cu un RFC informațional pentru a răspunde nevoilor clientilor ce doresc utilizarea unui protocol într-o rețea multivendor.

Industria Standardelor și Suitele de Protocole

TCP/IP	ISO	AppleTalk	Novell Netware
HTTP DNS DHCP FTP	ACSE ROSE TRSE SESE	AFP	NDS
TCP UDP	TP0 TP1 TP2 TP3 TP4	ATP AEP NBP RTMP	SPX
IPv4 IPv6 ICMPv4 ICMPv6	CONP/CMNS CLNP/CLNS	AARP	IPX
Ethernet	PPP	Frame Relay	ATM
			WLAN

Fig. 3.26.

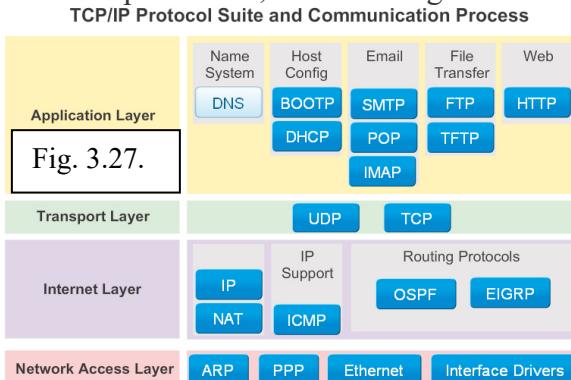
Suita IP este o suită de protocole necesară pentru transmiterea și receptia de informații utilizând Internetul. Este cunoscută ca suita TCP/IP deoarece primele două protocole de rețea definite pentru acest standard au fost TCP și IP. Standardul deschis TCP/IP a înlocuit alte suite de protocole particulare ale unui furnizor, cum ar fi Apple's AppleTalk și Novell's Internetwork Packet Exchange/Sequenced Packet Exchange (IPX/SPX).

Prima rețea cu comutare de pachete și predecesoare a Internetului de astăzi a fost Advanced Research Projects Agency Network (ARPANET), înființată în 1969 prin conectarea de computere mainframe în patru locații. ARPANET a fost fondată de către U.S. Department of

Defense pentru utilizarea să în universități și laboratoare de cercetare. Bolt, Beranek and Newman (BBN) a fost contractorul care a făcut o mare parte din dezvoltarea inițială a ARPANET, inclusiv crearea primului router cunoscut ca Interface Message Processor (IMP).

În 1973, Robert Kahn și Vinton Cerf au început să lucreze împreună la TCP pentru a dezvolta următoarea generație - ARPANET. TCP a fost dezvoltat pentru a înlocui Network Control Program (NCP) de la ARPANET. În 1978, TCP a fost divizat în două protocole: TCP și IP. Mai târziu, alte protocole au fost adăugate la suita TCP/IP cum ar fi Telnet, FTP, DNS și multe altele.

În prezent, suita include zeci de protocole, conform Figurii 3.27.



Ele sunt organizate în nivele utilizând modelul de protocol TCP/IP. Protocolele TCP/IP sunt incluse în nivelul Internet la nivelul aplicație atunci când ne referim la modelul TCP/IP. Protocolele de nivel inferior - legătură de date sau nivelul de acces la rețea - sunt responsabile de livrarea pachetului IP peste mediul fizic. Aceste protocole de nivel inferior sunt dezvoltate de organizații de standard, cum ar fi IEEE.

Suia de protocol TCP/IP este implementată ca stiva TCP/IP la ambele hosturi, destinație și sursă, pentru a oferi livrarea aplicațiilor peste rețea. Protocolele Ethernet 802.3 sunt utilizate pentru a transmite pachetul IP peste mediul fizic dintr-un LAN.

Figurile xxx2 și 3 demonstrează procesul complet de comunicare utilizând un exemplu de server web ce transmite date către un client:

1. Pagina web Hypertext Markup Language (HTML) de pe server este **data** ce trebuie transmisă.
2. Headerul protocolului aplicație HTTP este adăugat la începutul datelor HTML. Headerul conține informații variate, inclusiv versiunea HTTP folosită de server și un cod de stare indicând faptul că are informații pentru clientul web.
3. Protocolul nivelului aplicație HTTP livrează datele paginii web formatată HTML către nivelul transport. Protocolul nivelului transport TCP este folosit pentru a gestiona conversația individuală dintre clientul web și server.
4. Apoi, informațiile IP sunt adăugate la începutul informațiilor TCP. IP desemnează adresele IP adecvate ale sursei și destinației. Aceste informații sunt cunoscute ca un pachet IP.
5. Protocolul Ethernet adaugă informații la ambele capete ale pachetului IP, cunoscut sub forma de cadru al legăturii de date. Acest cadru este livrat la cel mai apropiat router în calea spre clientul web. Acest router înălătură informațiile Ethernet, analizează pachetul IP, determină cea mai bună cale pentru pachet, inserează pachetul într-un nou cadru și îl trimite către următorul router vecin spre destinație. Fiecare router înălătură și adaugă noi informații despre legătura de date înainte de a transmite pachetul.
6. Aceste date sunt acum transmise prin internetwork, ce constă din dispozitive intermediare și medii de comunicare.
7. Clientul primește cadrele de legătură de date ce conțin datele și fiecare header de protocol este procesat și apoi înălăturat în ordinea inversă în care au fost adăugate. Informațiile Ethernet sunt

procesate și înălțurate, urmate de informațiile protocolului IP, informațiile TCP și în final, informațiile HTTP.

8. Informațiile paginii web sunt apoi transmise la softwareul de browser web al clientului.

Protocol Operation of Sending and Receiving a Message

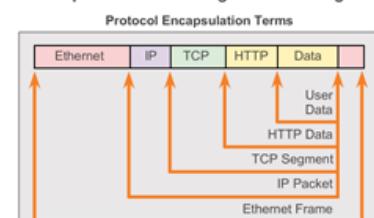


Fig. 3.28.A

Protocol Encapsulation Terms



Fig. 3.28.B



Fig. 3.28.A

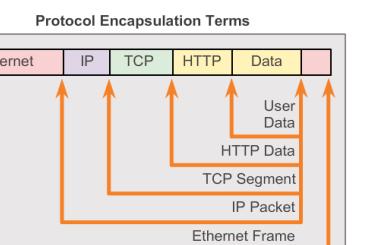


Fig. 3.28.C

Protocol Encapsulation Terms

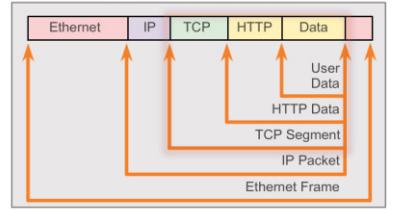
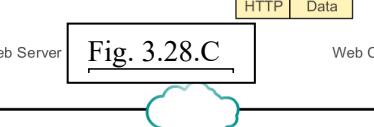


Fig. 3.28.D



Protocol Encapsulation Terms

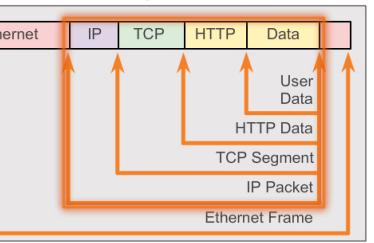


Fig. 3.28.E

Protocol Encapsulation Terms

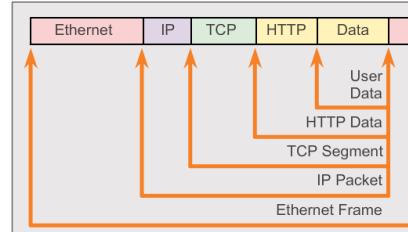
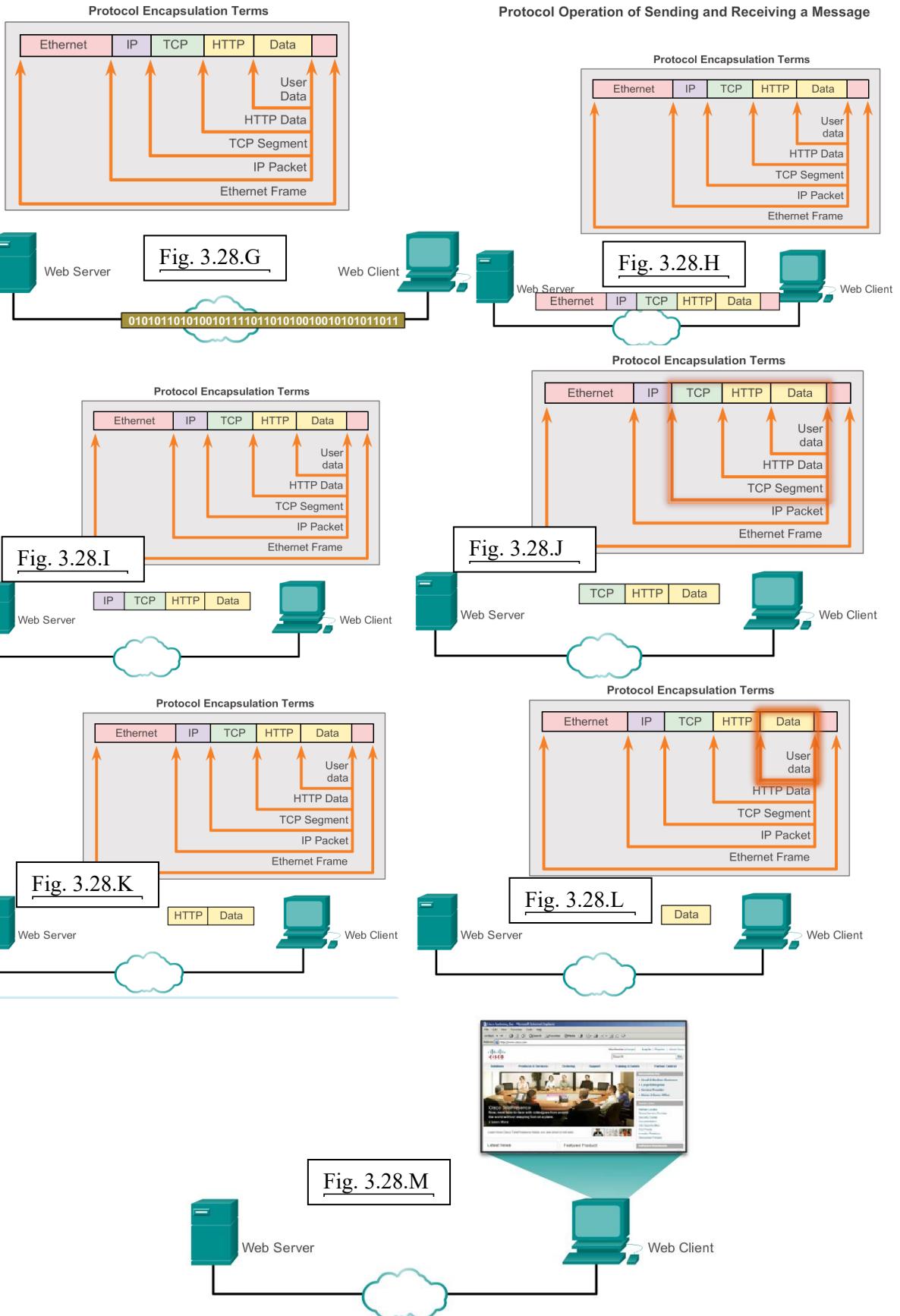


Fig. 3.28.F





3.4 Organizațiile de emitere a standardelor

Standardele deschise încurajează competiția și lucrurile inovatoare. De asemenea garantează că niciun produs nu poate acapara întreaga piață sau că acesta are un avantaj nedrept în fața competiției. Un exemplu bun al acestui lucru este atunci când se achiziționează un router wireless pentru domiciliu. Există multe alegeri diferite disponibile de la o varietate de furnizori, toate cu standarde deschise încorporate cum ar fi IPv4, DHCP, 802.3(Ethernet) și 802.11 (Wireless LAN). Aceste standarde deschise permit unui client ce rulează un sistem de operare Apple's OS X să descarce o pagină web de la un server web ce rulează sistemul de operare Linux. Acest lucru este realizabil datorită faptului că ambele sisteme de operare implementează protocoale de standarde deschise, cum ar fi cele din suita TCP/IP.

Organizațiile de standarde sunt importante în gestionarea unui Internet deschis cu protocoale și specificații accesibile care pot fi implementate de orice furnizor. O organizație de standarde ar putea proiecta un set de reguli proprii sau în alte cazuri, ar putea selecta un protocol privat ca bază pentru standard. Dacă este utilizat un protocol privat, se implică de obicei și furnizorul care a creat protocolul.

Organizațiile de standarde sunt de obicei organizații non-profit, neutre cu privire la furnizor, stabilite pentru dezvoltarea și promovarea conceptului de standarde deschise.

Organizațiile de standarde includ:

- *The Internet Society (ISOC)*.
- *The Internet Architecture Board (IAB)*.
- *The Internet Engineering Task Force (IETF)*.
- *The Institute of Electrical and Electronics Engineers (IEEE)*.
- *The International Organization for Standardization (ISO)*.

Fiecare dintre aceste organizații vor fi prezentate detaliat în următoarele pagini.

Internet Society (ISOC) este responsabilă de promovarea dezvoltării deschise, evoluția și utilizarea Internetului în lume. ISOC facilitează devoltarea de standarde și protocoale deschise pentru infrastructura tehnică din Internet, inclusiv supravegherea Internet Architecture Board (IAB).

Internet Architecture Board (IAB) este responsabilă de managementul general și de dezvoltarea de standarde din Internet. IAB oferă supravegherea arhitecturii pentru protocoale și proceduri utilizate de către Internet. IAB este alcătuită din 13 membrii, inclusiv Internet Engineering Task Force (IETF). Membrii IAB sunt indivizi independenți și nu reprezentanți ai unei companii, agenții sau alte organizații.



Misiunea IEFT este de a dezvolta, actualiza și gestiona tehnologiile TCP/IP. Una dintre responsabilitățile cheie ale IEFT este de a produce documente Request for Comments (RFC), ce reprezintă un memorandum ce descrie protocoale, procese și tehnologii pentru Internet. IEFT constă din grupuri de lucru, mecanismul principal pentru dezvoltarea specificațiilor și orientărilor IEFT. Grupurile de lucru sunt echipe mici, iar după ce obiectivele grupului sunt terminate, grupul este dizolvat. Internet Engineering Steering Group (IESG) este responsabil pentru managementul tehnic al IEFT și pentru procesul de standarde Internet.

Internet Research Task Force (IRTF) este axat pe cercetare pe termen lung legată de Internet și protocoale TCP/IP, aplicații, arhitectură și tehnologii. În timp ce IETF se axează pe probleme pe termen scurt ale creării de standarde, IRTF constă din grupuri de cercetare pentru eforturi de dezvoltare pe termen lung. Unele dintre grupurile de cercetare curente sunt Anti-Spam Research Group (ASRG), Crypto Forum Research Group (CFRG), Peer-to-Peer Research Group (P2PRG), și Router Research Group (RRG).

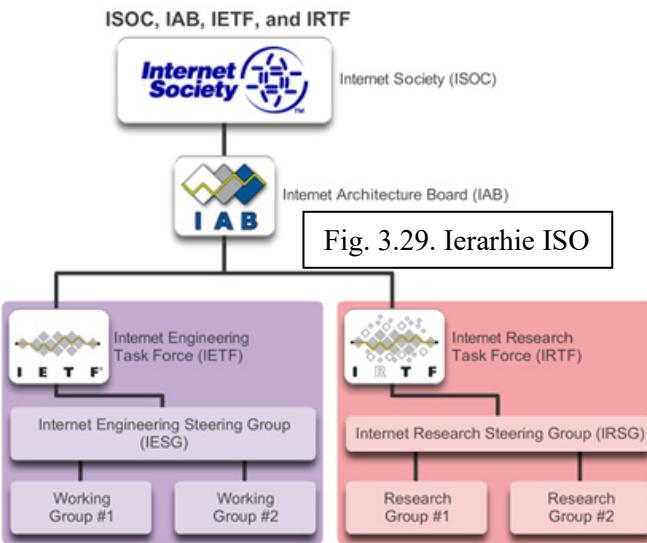


Fig. 3.29. Ierarhie ISO

Institute of Electrical and Electronics Engineers este o organizație profesională pentru acele domenii electronice și de inginerie electrică ce sunt dedicate pentru inovări tehnologice și pentru crearea de standarde. În 2012, IEEE constă din 38 de societăți, a publicat 130 de jurnale și a sponsorizat peste 1.300 de conferințe din întreaga lume. IEEE are peste 1.300 de standarde și proiecte aflate în curs de dezvoltare.



IEEE are peste 400.000 de membrii în mai mult de 160 de țări. Mai mult de 107.000 dintre acești membrii sunt membrii studenți. IEEE oferă oportunități educaționale și de carieră pentru a promova aptitudinile și cunoașterea industriei electronice.

IEEE este una dintre organizațiile importante ce produce standarde din lume. Creaază și menține standarde ce afectează o gamă largă de industrii, inclusiv cele de putere și energie, sănătate, telecomunicații și rețelistică. Familia de standarde IEEE 802 se ocupă de rețelele din aria locală și de rețelele din aria metropolitană, cablare și wireless. Ca și în Fig. , fiecare standard IEEE constă dintr-un grup de lucru – Working Group (WG) – responsabil de crearea și îmbunătățirea standardelor.

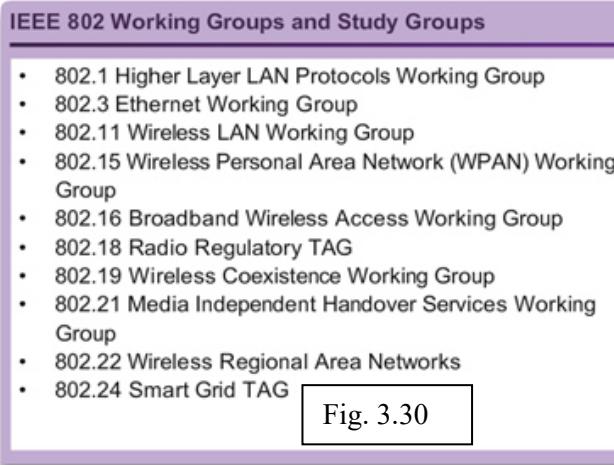


Fig. 3.30

Standardele IEEE 802.3 și IEEE 802.11 sunt standarde importante în rețelistică. Standardul IEEE 802.3 definește Media Access Control (MAC) pentru Ethernet prin cablu. Această tehnologie este pentru LANuri, dar are și aplicații WAN. Standardul 802.11 definește un set de standarde pentru implementarea WLANurilor. Acest standard definește Open Systems Interconnection (OSI) fizic și legătura de date MAC pentru comunicațiile wireless.

ISO, International Organization for Standardization, este cel mai mare dezvoltator din lume de standarde internaționale pentru o varietate de produse și servicii. ISO nu este un acronim pentru numele organizației; mai degrabă termenul ISO se bazează pe cuvântul gresesc “isos” ce înseamnă egal. International Organization for Standardization a ales termenul ISO pentru a-și afirma poziția să de egalitate în toate țările.



În rețelistică, ISO este bine cunoscut pentru modelul de referință Open Systems Interconnection (OSI). ISO a publicat modelul de referință OSI în 1984 pentru a dezvolta un cadru pe nivele pentru protocolele de rețea. Obiectivul original al acestui proiect a fost nu numai să creeze un model de referință, dar mai mult și să servească ca bază pentru o suită de protocole utilizate în Internet. A fost cunoscut ca suita de protocole OSI. Oricum, având în vedere creșterea popularității suitei TCP/IP, dezvoltată de Robert Kahn, Vinton Cerf și alții, suita de protocol OSI nu a fost aleasă ca suita de protocole pentru Internet. În schimb, suita de protocole TCP/IP a fost selectată. Suta de protocole OSI a fost implementată pe echipamentul de telecomunicații și poate fi încă întâlnită în rețelele de telecomunicații.

Este de preferat să fie cunoscute majoritatea produselor ce utilizează standarde ISO. Extensia de fișier ISO este utilizată la multe imagini CD pentru a evidenția faptul că standardul ISO 9660 este folosit pentru sistemul de fișiere. ISO este, de asemenea, responsabil pentru crearea de standarde pentru protocolele de rutare.



Standardele de rețea implică multe alte organizații de standarde. Unele dintre cele mai întâlnite sunt:

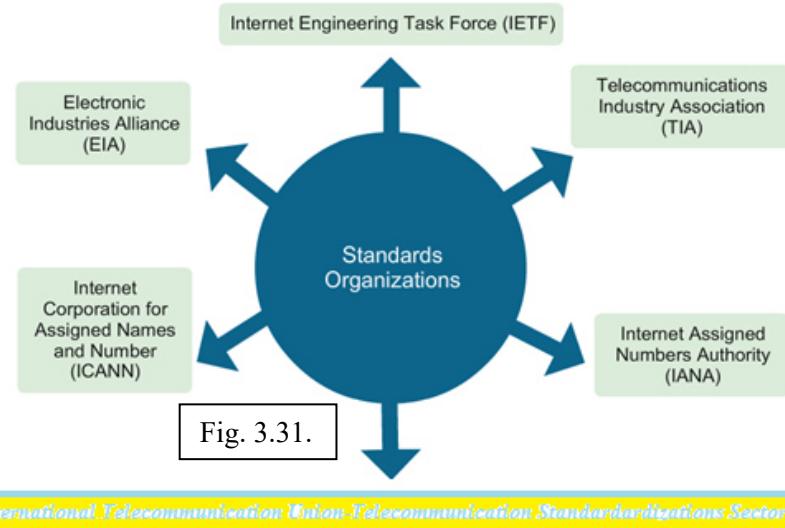
- **EIA - Electronic Industries Alliance (EIA)**, cunoscută anterior ca Electronics Industries Association, este o organizație de comerț și standarde internaționale pentru organizațiile electronice. EIA este bine cunoscută pentru standardele sale legate de cabluri, conectori și rackuri de 19 inch utilizate pentru montarea echipamentului de rețea.
- **TIA - Telecommunications Industry Association (TIA)** este responsabilă de dezvoltarea de standarde de comunicație într-o varietate de domenii, cum ar fi echipamente radio, turnuri celulare, dispozitive VoIP, comunicații satelit și altele. Multe dintre standardele lor sunt produse în colaborare cu EIA.
- **ITU-T - International Telecommunications Union-Telecommunication Standardization Sector (ITU-T)** este una dintre cele mai mari și mai vechi organizații de standarde de comunicație. ITU-T definește standarde pentru comprimarea video, Internet Protocol Television (IPTV) și comunicații broadband, cum ar fi DSL. De exemplu, atunci când este apelată o altă țară, codurile țării ITU sunt folosite pentru a efectua conexiunea.
- **ICANN - The Internet Corporation for Assigned Names and Numbers (ICANN)** este o organizație non-profit cu sediul în Statele Unite, care coordonează alocarea de adrese IP, managementul numelor de domenii utilizate de DNS și identificatorii de protocol sau numerele de port utilizate de protocolele TCP și UDP. ICANN crează politici și are responsabilitate generală pentru aceste sarcini.



- **IANA** - *Internet Assigned Numbers Authority (IANA)* este un departament al ICANN responsabil de supravegherea și gestionarea alocării de adrese IP, managementul numelor de domenii și identificatorii de protocol pentru ICANN.



Familiarizarea cu organizațiile care dezvoltă standarde utilizate în rețea ajută la o înțelegere mai bună asupra modul în care aceste standarde crează un Internet deschis, neutru la furnizor, și oferă posibilitatea de învățare despre noi standarde în curs de dezvoltare.



International Telecommunication Union-Telecommunication Standardizations Sector (ITU-T)

3.5 Modele de referință

Un model pe nivele, cum ar fi modelul TCP/IP, este adesea utilizat pentru a ajuta vizualizarea interacțiunii dintre protocoale diferite. Un model pe nivele descrie funcționarea protocoalelor ce au loc la fiecare nivel, precum și interacțiunea protocoalelor cu nivelele superioare și inferioare fiecărui nivel.

Există beneficii în utilizarea modelului pe nivele pentru a descrie protocoalele de rețea și operațiile. Utilizarea unui model pe nivele:

- Ajută la proiectarea protocolului, deoarece protocoalele ce operează la un nivel specific au definite informații după care acționează și o interfață definită pentru nivelele superioare și inferioare.
- Menține competiția deoarece produsele de la furnizori diferite pot lucra împreună.
- Previne schimbări tehnologice sau în capacitate la un nivel pentru a nu afecta alte nivele superioare și inferioare.
- Oferă un limbaj comun pentru a descrie funcțiile și caracteristicile de rețea.

Există două tipuri de bază ale modelelor de rețea:

- **Modelul de protocol** – Acest model îmbină structura unei suite particulare de protocol. Setul ierarhic de protocoale relaționale într-o suită reprezintă în mod normal toate funcționalitățile necesare pentru interfațarea rețelei umane cu rețeaua de date. Modelul TCP/IP este un model de protocol, deoarece descrie funcțiile ce au loc la fiecare nivel de protocoale din suita TCP/IP.
- **Modelul de referință** – Acest model oferă consistență în toate tipurile de protocoale și servicii prin descrierea a ceea ce trebuie să fie efectuat la un nivel particular, dar nu menționează modul în care trebuie să fie realizat. Un model de referință nu are intenția să fie o specificație de implementare sau să ofere un nivel de detaliu pentru definirea cu

precizie a serviciilor din arhitectura de rețea. Principalul obiectiv al modelului de referință este ajutarea la înțelegerea clară a funcțiilor și proceselor implicate.

Modelul OSI este cel mai cunoscut model de referință. Este utilizat pentru proiectarea rețelei de date, specificațiile de funcționare și depanare.

Ca și în Fig. , modelele TCP/IP și OSI sunt principalele modele utilizate când vine vorba de funcționalitatea rețelei. Proiectanții protocolelor de rețea, serviciilor sau dispozitivelor își pot crea propriile modele pentru reprezentarea produselor. La final, proiectanții trebuie să le prezinte industriei prin legarea produsului sau serviciului de modelul OSI sau de modelul TCP/IP, sau de ambele.

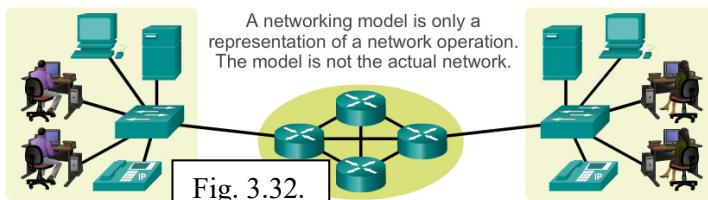
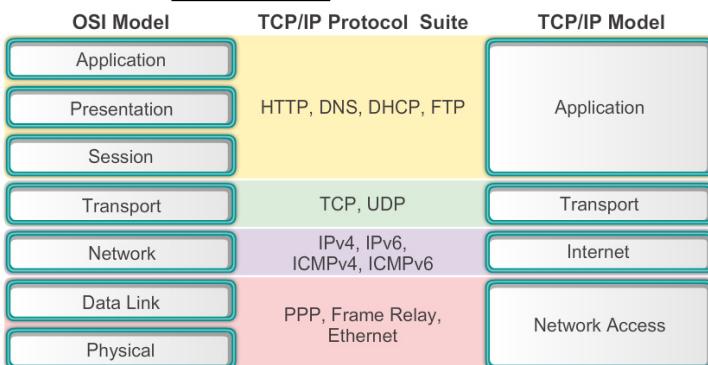


Fig. 3.32.



Inițial, modelul OSI a fost proiectat de către ISO pentru a oferi un cadru pe care să se construiască o suită de protocole de sistem, deschisă. Viziunea a fost ca acest set de protocole să fie utilizat pentru dezvoltarea unei rețele internaționale care să nu fie dependentă de sisteme proprietare.

În final, viteza de dezvoltare a Internetului bazat pe TCP/IP, a condus la adoptarea acestuia ca protocol de internet, ceea ce a cauzat neacceptarea suitei de protocole OSI. Deși câteva dintre protocolele dezvoltate cu specificațiile OSI sunt utilizate pe scară largă astăzi, modelul OSI pe șapte nivele a adus contribuții majore la dezvoltarea altor protocole și produse pentru toate tipurile de rețele noi.

Modelul OSI oferă o listă cuprinzătoare de servicii și funcții care au loc la fiecare nivel. De asemenea, descrie interacțiunea fiecărui nivel cu nivelele direct inferioare și superioare lui. Deși conținutul acestui curs este structurat în jurul modelului de referință OSI, prezentarea se axează și pe protocolele identificate în modelul de protocol TCP/IP.

Notă: Spre deosebire de nivelele modelului TCP/IP ce sunt referite numai după nume, nivelele modelului OSI sunt mai mult referite după număr, decât după nume. De exemplu, nivelul fizic este referit cu Nivelul 1 din modelul OSI.

Modelul OSI



Fig. 3.34.

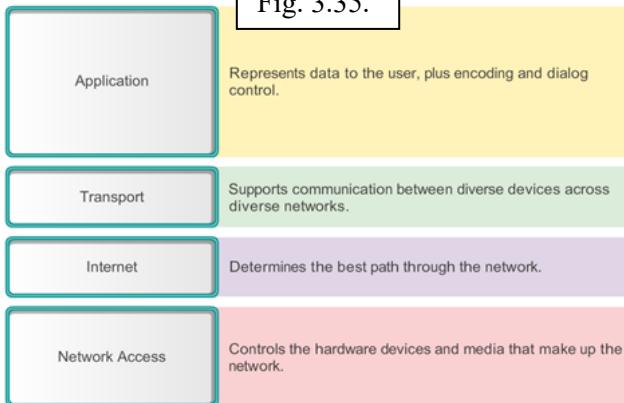
Modelul de protocol TCP/IP pentru comunicațiile internetwork a fost creat la începutul anului 1970 și este uneori numit modelul Internet. Ca și în Fig. , definește patru categorii de funcții ce au loc într-o comunicație cu succes. Arhitectura suitei de protocoale TCP/IP urmează structura acestui model. Din acest motiv, modelul Internet este referit adesea ca modelul TCP/IP.

Multe modele de protocoale descriu o stivă de protocoale specifică furnizorului. Oricum, datorită faptului că modelul TCP/IP este un standard deschis, nu doar o singură companie controlează definiția acestui model. Definițiile standardului și protocoalelor TCP/IP sunt discutate pe un forum public și definite într-un set disponibil public de RFCuri. RFC conține specificațiile formale ale protocoalelor de comunicare și resursele ce descriu utilizarea protocoalelor.

RFCurile conțin de asemenea documente organizaționale și tehnice despre Internet, inclusiv specificațiile tehnice și documentele de politică produse de către IEFT.

Modelul TCP/IP

Fig. 3.35.



Protocoalele ce alcătuiesc suita de protocoale TCP/IP pot fi descrise în termenii modelului de referință OSI. În modelul OSI, nivelul de acces la rețea și nivelul aplicație ale modelului TCP/IP sunt divizate pentru a descrie funcțiile distinct ce au loc la aceste nivele.

La nivelul de acces la rețea, suita de protocoale TCP/IP nu specifică ce protocoale să fie utilizate atunci când se transmite peste un mediu fizic; descrie numai transferul de la nivelul Internet la protocoalele de rețea fizice. Nivelele 1 și 2 OSI discută procedurile necesare de acces ale căilor fizice și media pentru a transmite datele peste rețea.

Precum în Fig. 3.35, comparațiile critice dintre cele două modele de rețea au loc la nivelele 3 și 4 OSI. Nivelul 3 OSI, nivelul rețea, este aproape folosit unilateral pentru a descrie gama de

procese ce au loc în toate rețelele de date pentru a adresa și ruta mesajele într-un internetwork. IP este protocolul suitei TCP/IP ce include funcționalitatea descrisă la Nivelul 3 OSI.

Nivelul 4, nivelul transport al modelului OSI, descrie servicii și funcții generale care oferă o livrare ordonată și de încredere a datelor de la sursă la destinație. Aceste funcții includ confirmarea, recuperarea erorilor și sevențierea. La acest nivel, protocolele TCP/IP TCP și UDP oferă funcționalitatea necesară.

Nivelul aplicație TCP/IP include un număr de protocole ce oferă funcționalitate specifică a unei variații de aplicații de utilizator. Nivelele OSI 5, 6, 7 sunt utilizate ca referințe pentru dezvoltatorii de software de aplicații și furnizori pentru a produce produse ce operează în rețele.

Comparație între Modelul OSI și Modelul TCP/IP

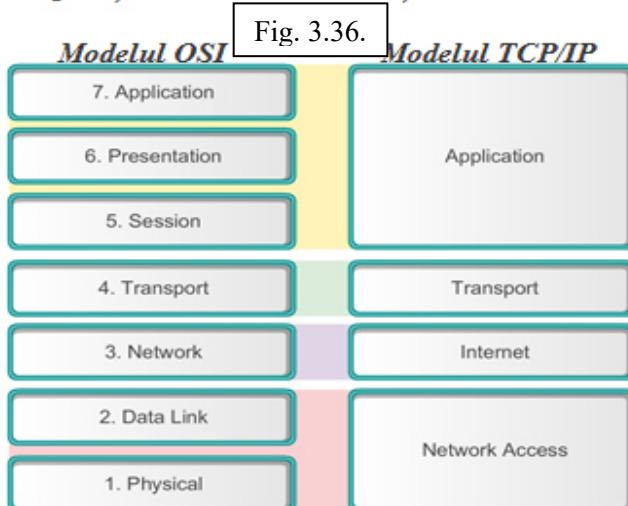


Fig. 3.36.

3.6 Deplasarea datelor prin intermediul Rețelelor

3.6.1 Încapsularea datelor

În teorie, o singură comunicație, cum ar fi un video muzical sau un mesaj prin e-mail, poate fi trimisă prin rețea de la o sursă la o destinație ca un șir neîntrerupt, masiv de biți. Dacă mesajele ar fi transmise așa, ar însemna că nici-un alt dispozitiv nu va fi capabil să trimită sau să primească mesaje în aceeași rețea atât timp cât transferul de date este în desfășurare. Aceste șiruri mari de date vor avea ca rezultat întârzieri semnificative. Mai mult, dacă o legătură din infrastructură de rețea interconectată "pică" în timpul transmisiei, mesajul complet va fi pierdut și retransmis în întregime.

O abordare mai bună este divizarea datelor în bucăți mai mici, ușor gestionabile, peste rețea. Această divizare a fluxului de date în bucăți mici se numește segmentare. Segmentarea mesajelor are două beneficii importante:

- Prin trimitera de bucăți mici individuale de la sursă la destinație, mai multe conversații diferite pot avea loc în același timp în rețea. Procesul utilizat de intercalare a bucăților din conversații separate în rețea se numește multiplexare.

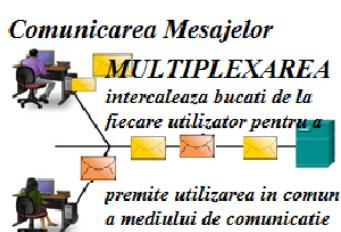
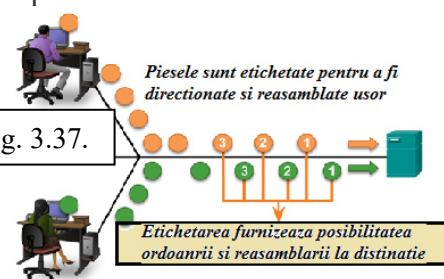


Fig. 3.37.



- Segmentarea poate crește încrederea comunicațiilor din rețea. Pieselete separate ale fiecărui mesaj trebuie să parcurgă aceeași cale în rețea de la sursă la destinație. Dacă o cale particulară devine aglomerată cu traficul de date sau devine indisponibilă, bucățile individuale ale mesajului pot fi direcționate spre destinație pe căi alternative. Dacă o parte din mesaj nu ajunge la destinație, numai părțile lipsă trebuie să fie retransmise.

Dezavantajul în utilizarea segmentării și multiplexării pentru transmiterea de mesaje prin rețea este nivelul de complexitate adăugat procesului. Presupunem că avem de trimis o scrisoare de 100 de pagini, dar fiecare plic poate conține o singură pagină. Procesul de adresare, etichetare, expediere, primire și deschidere a celor 100 de plicuri ar fi consumator de timp pentru ambele persoane, expeditorul și destinatarul.

În comunicațiile din rețea, fiecare segment al mesajului trebuie să treacă printr-un proces similar pentru a se asigura de faptul că primește destinația corectă și că va putea fi reasamblat în conținutul original al mesajului.

Multe tipuri de dispozitive din rețea participă la asigurarea faptului că bucățile mesajului ajung la destinație într-un mod de încredere.

Deoarece datele sunt transmise prin stiva de protocoale în drumul lor prin mediul de rețea, multe protocoale adaugă informații la fiecare nivel. Acest lucru este cunoscut sub numele de proces de încapsulare.

Forma pe care o bucată de date o ia la fiecare nivel se numește PDU. În timpul încapsulării, fiecare nivel încapsulează PDU primit de la nivelul anterior în concordanță cu protocolul utilizat. La fiecare etapă din proces, un PDU are un nume diferit pentru a reflecta noile funcții. Deși nu există o convenție de nume universal pentru PDU, în acest material, PDUrile sunt numite în concordanță cu protocoalele suitei TCP/IP:

- Data** – Termenul general pentru PDU utilizat la nivelul aplicație.
- Segment** – PDU de la nivelul transport.
- Packet** – PDU de la nivelul Internet.
- Frame** – PDU de la nivelul acces la rețea.
- Bits** – Un PDU utilizat atunci când se transmit datele fizic prin mediu.

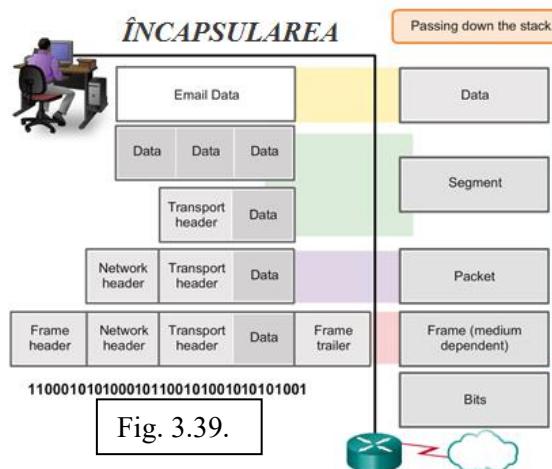
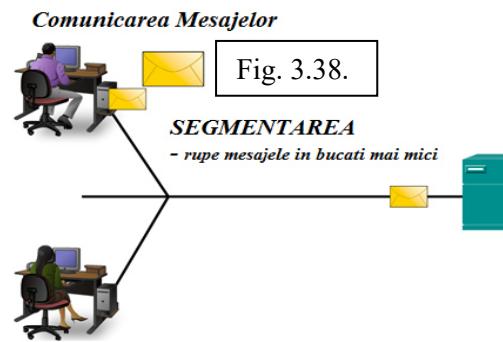


Fig. 3.39.



Încapsularea datelor este procesul care adaugă un header de protocol cu informații suplimentare datelor înaintea transmisiei. În multe forme ale comunicațiilor de date, datele originale sunt încapsulate sau înfășurate în mai multe protocoale înainte de a fi transmise.

Atunci când sunt trimise mesaje în rețea, stiva de protocoale de pe un host operează de sus în jos. În exemplul cu serverul web, putem folosi modelul TCP/IP pentru a ilustra procesul de transmitere a unei pagini HTML la un client.

Protocolul de nivel aplicație, HTTP, începe procesul prin livrarea datelor paginii web HTML formatare la nivelul transport. Acolo, datele aplicației sunt “sparte” în segmente TCP. Fiecare segment TCP primește un tabel, numit header, ce conține informații despre procesul ce rulează pe computerul destinație ce trebuie să primească mesajul. Conține, de asemenea, și informații ce permit ca procesul destinație să reasambleze datele în formatul original.

Nivelul transport încapsulează datele paginii web HTML în segment și îl trimită la nivelul Internet, unde protocolul IP este implementat. Aici, întregul segment TCP este încapsulat într-un packet IP, ce adaugă un alt tabel, numit header IP. Headerul IP conține adresele IP sursă și destinație, precum și informațiile necesare pentru transmiterea pachetului la procesul destinație corespunzător.

Apoi, pachetul IP este transmis la nivelul de acces la rețea unde este încapsulat cu un header și un trailer pentru cadru. Fiecare header al cadrului conține o adresă fizică sursă și destinație. Adresa fizică identifică unic dispozitivul din rețea locală. Trailerul conține informații de verificare de erori. În final, biții sunt codificați în mediu prin NIC.

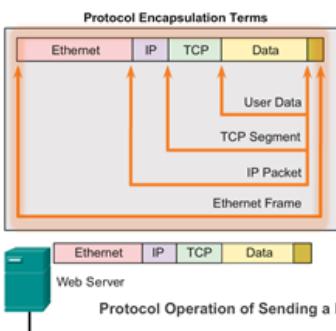


Fig. 3.40.

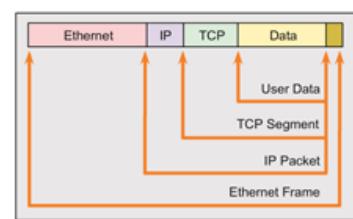
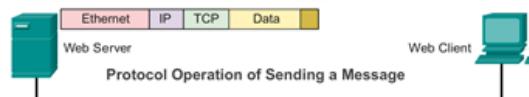


Fig. 3.41.



Procesul este inversat pe hostul destinație și se numește decapsulare. Decapsularea este procesul utilizat de către un dispozitiv destinație pentru a îndepărta unul sau mai multe headere de protocol. Datele sunt decapsulate și mutate înapoi în stivă de jos în sus, până la aplicația utilizatorului final.

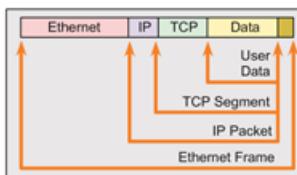


Fig. 3.42.

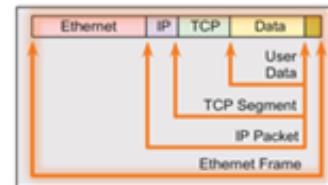
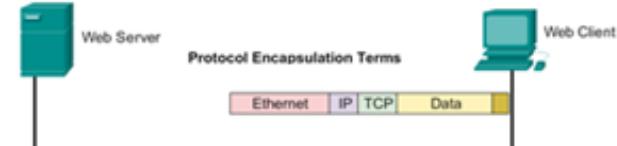


Fig. 3.43.



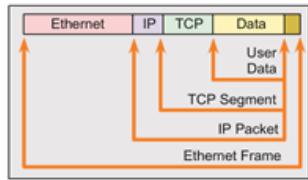


Fig. 3.44.

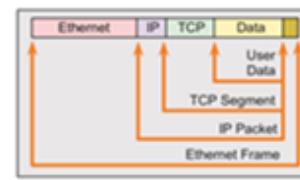


Fig. 3.45.



Fig. 3.46.

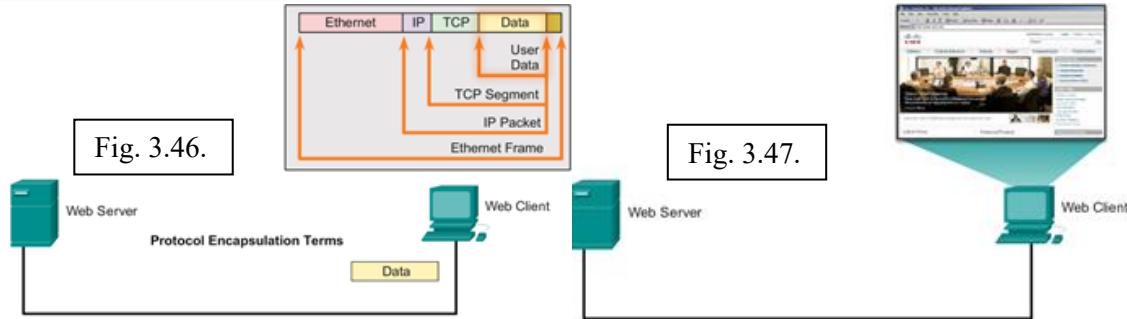


Fig. 3.47.

Accesarea resurselor locale – Modelul OSI descrie procesele de codare, formatare, segmentare și încapsularea datelor pentru transmisia peste rețea. Nivelul rețea și nivelul legătură de date sunt responsabile de livrarea datelor de la dispozitivul sursă sau expeditor, la dispozitivul destinație sau destinatar. Protocolele de la ambele nivele conțin adresele sursă și destinație, dar adresele lor au diferite scopuri.

3.7 Adresa de rețea

La Nivelul Rețea, sau Nivelul 3, se regăsește adresa logică care conține informații necesare pentru livrarea packetului IP de la dispozitivul sursă la dispozitivul destinație. O adresă IP de nivel 3 are două părți, prefixul de rețea și partea de host. Prefixul de rețea este folosit de către routere pentru a transmite packetul în rețea adecvată. Partea de host este utilizată de către ultimul router din cale – drumul dintre sursă și destinație - pentru a livra pachetul la destinație.

Un packet IP conține două adrese IP:

- **Adresa IP sursă** – adresa IP a dispozitivului expeditor.
- **Adresa IP destinație** – adresa IP a dispozitivului destinație. Adresa IP destinație este utilizată de către routere pentru a transmite pachetul la destinația să.

Adresele de la nivelul rețea, sau adresele IP, indică rețea și adresa de host sursă și destinație. Partea de rețea a adresei va fi aceeași; numai partea de host sau de dispozitiv a adresei va fi diferită.

- Adresa IP sursă – adresa IP a dispozitivului sursă, computerul client PC1 : 192.168.1.110.
- Adresa IP destinație – adresa IP a dispozitivului destinație, FTP server: 192.168.1.9.

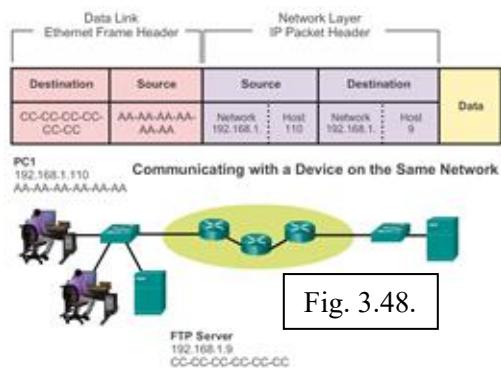


Fig. 3.48.

3.8 Adresa de legătură de date

Adresele fizice ale nivelului legătură de date, sau Nivelul 2, au un rol diferit. Rolul adreselor de la nivelul legătură de date este de a livra frameul de la o interfață de rețea la o altă interfață de rețea din aceeași rețea. Înainte ca un packet IP să fie transmis peste o rețea cablată sau wireless, trebuie să fie încapsulat într-un frame pentru a putea fi transmis peste mediul fizic, adică rețeaua propriu-zisă. LANurile Ethernet și LANurile wireless sunt două exemple de rețele ce au medii fizice diferite, fiecare având propriul său tip de protocol de legătură de date.

Pachetul IP este încapsulat într-un frame la nivelul legătură de date pentru a fi transmis la rețeaua destinație. Adresele sursă și destinație de la nivelul legătură de date sunt adăugate, conform Figurii xxx:

- **Adresa legătură de date sursă** – adresa fizică a dispozitivului care trimite pachetul. Inițial, aceasta este adresa NIC, sursă a pachetului IP.
- **Adresa legătură de date destinație** – adresa fizică a interfeței de rețea sau routerul următor sau interfață de rețea a dispozitivului destinație.

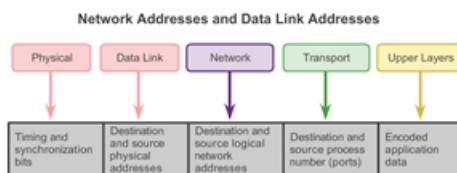


Fig. 3.49.

Pentru a înțelege modul în care comunicarea este realizată cu succes în rețea, este important să înțelegem rolurile adreselor de la nivelul rețea și de la nivelul legătură de date, atunci când un dispozitiv comunică cu alt dispozitiv din aceeași rețea. În acest exemplu, avem un computer client PC1 ce comunică cu un server de fișiere, FTP server, în aceeași rețea IP.

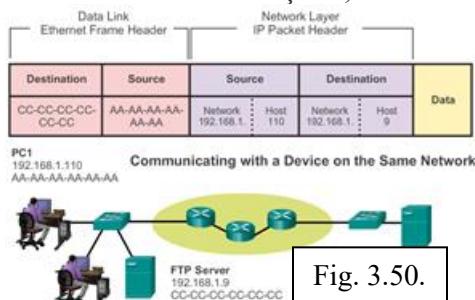


Fig. 3.50.

Atunci când expeditorul și destinatarul packetului IP sunt din aceeași rețea, frameul data link este trimis direct la dispozitivul destinatar. Pe o rețea Ethernet, adresele legătură de date sunt cunoscute ca adrese MAC Ethernet. Adresele MAC sunt adrese pe 48 de biți ce sunt încorporate fizic pe Ethernet NIC. O adresă MAC este cunoscută și ca adresa fizică sau **"burned-in address"** (BIA).

- **Adresa MAC destinație** – atunci când dispozitivul destinatar este în aceeași rețea cu dispozitivul sursă, aceasta este adresa legătură de date a dispozitivului destinatar. În exemplul nostru, adresa MAC destinație este adresa MAC a serverului FTP: CC-CC-CC-CC-CC-CC.

Adresele sursă și destinație sunt adăugate la frameul Ethernet. Frameul cu pachetul IP încapsulat poate fi acum transmis de la PC1 direct la serverul FTP.

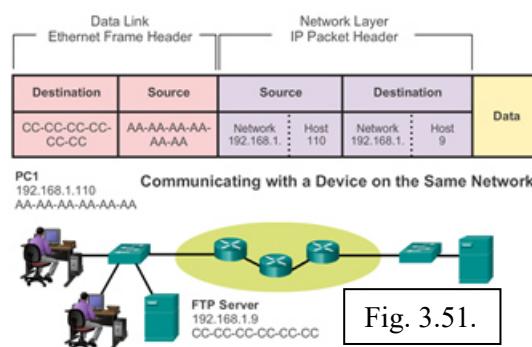
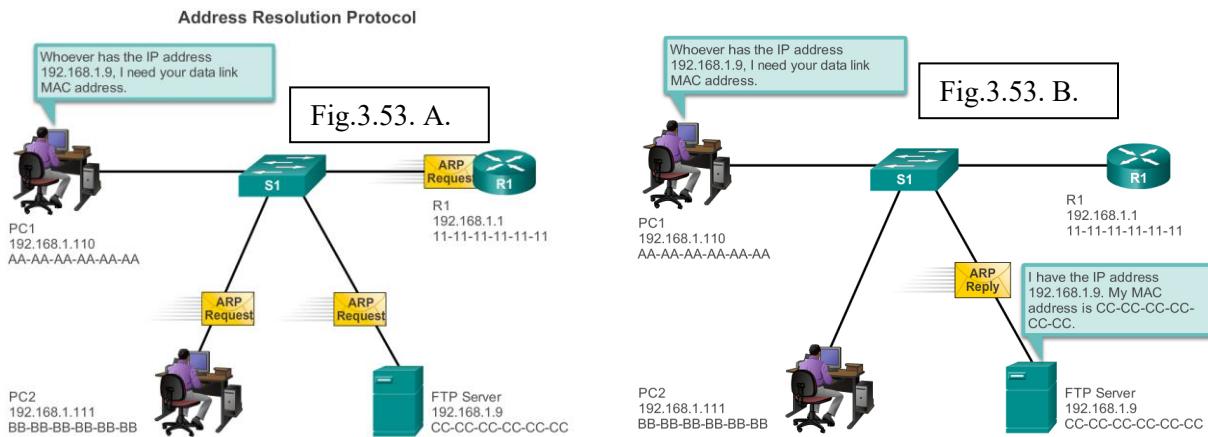


Fig. 3.51.

Ar trebui să fie clar acum că pentru a transmite date de la un host la altul în același LAN, hostul sursă trebuie să cunoască adresele logică și fizică ale hostului destinație. O dată ce sunt cunoscute, poate crea un frame și îl poate trimite pe mediul de rețea. Hostul sursă poate învăța adresa IP destinație în mai multe moduri. De exemplu, ar putea învăța adresa IP prin utilizarea Domain Name System (DNS), sau ar putea cunoaște adresa IP destinație datorită faptului că a fost introdusă manual în aplicație, ca atunci când un utilizator specifică adresa IP a serverului FTP destinație. Dar cum un host determină adresa MAC a altui dispozitiv ?

Multe aplicații de rețea se bazează pe adresa IP logică a destinației pentru a identifica locația hosturilor cu care comunică. Adresa MAC este necesară pentru a livra pachetul IP încapsulat în frameul Ethernet peste rețea, spre destinație.

Hostul sursă utilizează un protocol numit Address Resolution Protocol (ARP) pentru a descoperi adresa MAC a oricărui host din aceeași rețea locală. Hostul sursă trimite un mesaj ARP Request către întregul LAN. ARP Request este un mesaj broadcast. ARP Request conține adresa IP a dispozitivului destinație. Fiecare dispozitiv din LAN examinează ARP Request pentru a vedea dacă acesta conține propria să adresa IP. Numai dispozitivul ce conține adresa IP din ARP Request răspunde cu un ARP Reply. ARP Reply include adresa MAC asociată cu adresa IP din ARP Request.



3.9 Accesarea resurselor de la distanță

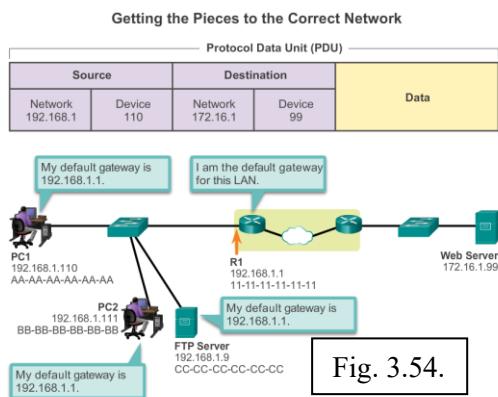
Metoda utilizată de host pentru a trimite mesaje la o destinație dintr-o rețea de la distanță diferă de modul în care un host trimite mesaje la o destinație din aceeași rețea locală. Atunci când un host necesită să trimită un mesaj la un alt host din aceeași rețea, va transmite mesajul direct. Un host va utiliza ARP pentru a descoperi adresa MAC a hostului destinație. Include adresa IP destinație în headerul pachetului și îl încapsulează într-un frame ce conține adresa MAC a destinației, apoi îl transmite.

Atunci când un host necesită să transmită un mesaj la o rețea aflată la distanță, trebuie să folosească routerul, cunoscut și ca echipament *"default gateway"*. Default gateway este adresa IP (de preferat cea mai mică adresă IP din range address) a unei interfețe de pe un router din aceeași rețea cu hostul sursă.

Este important ca adresa default gateway să fie config.ă pe fiacare host din rețea locală. Dacă nu este config.ă nici-o adresă default gateway în setările hostului, sau dacă este specificată o adresă default gateway greșită, mesajele adresate hosturilor din rețele de la distanță nu vor putea fi trimise.

În Fig. xxx, hosturile din LAN utilizează R1 ca default gateway cu adresa 192.168.1.1 config.ă în setările TCP/IP. Dacă destinația unui PDU este într-o rețea diferită, hosturile trimitem PDU la default gateway al routerului pentru ca acesta să transmită mai departe.

Dar care sunt rolurile adreselor de la nivelele rețea și legătura de date atunci când un dispozitiv comunică cu un dispozitiv dintr-o rețea de la distanță? În exemplul următor există un computer client, PC1, ce comunică cu un server, numit Web Server, dintr-o rețea IP diferită.



Adrese de rețea – Adresele IP indică rețeaua și adresele sursă și destinație ale dispozitivului. Atunci când expeditorul pachetului se află într-o rețea diferită de destinatar, adresele IP sursă și destinație vor reprezenta hosurile din rețele diferite. Acest lucru va fi indicat de către partea de rețea a dresei IP a hostului destinație.

- **Adresa IP sursă** – adresa IP a dispozitivului sursă, computerul client PC1: 192.168.1.110.
- **Adresa IP destinație** – adresa IP a dispozitivului destinație , serverul, Web Server: 172.16.1.99.

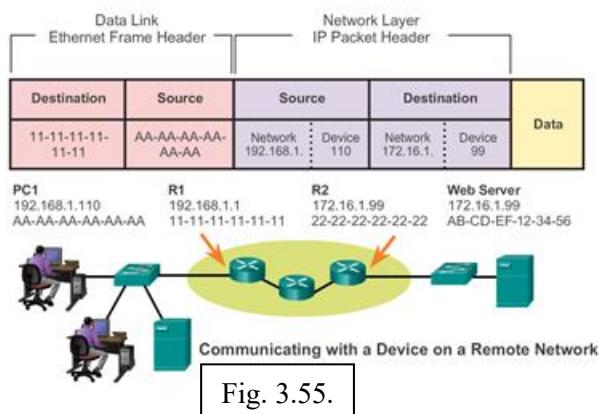
Adrese de legătură de date – Atunci când expeditorul și destinatarul pachetului IP se află în rețele diferite, frameul legăturii de date nu poate fi transmis direct la hostul destinație deoarece hostul nu este direct accesibil în rețeaua sursă. Frameul Ethernet trebuie să fie trimis la un alt dispozitiv cunoscut ca router sau default gateway. În exemplul nostru, default gateway este R1. R1 are o interfață și o adresă IP aflată în aceeași rețea cu PC1. Acest lucru permite ca PC1 să acceseze routerul direct.

- **Adresa MAC sursă** – adresa MAC Ethernet a dispozitivului sursă, PC1. Adresa MAC a interfeței Ethernet de pe PC1 este AA-AA-AA-AA-AA-AA.
- **Adresa MAC destinație** – atunci când dispozitivul destinație se află într-o rețea diferită de dispozitivul sursă, adresa MAC destinație este adresa MAC a default gateway sau a routerului. În exemplu, adresa MAC destinație este adresa MAC a interfeței Ethernet de pe R1 care este în aceeași rețea cu PC1, adică 11-11-11-11-11-11.

Frameul Ethernet cu pachetul IP încapsulat poate fi transmis la R1. R1 transmite mai departe pachetul spre destinație, către Web Server. Acest lucru ar putea însemna că R1 îl transmite către alt router sau direct la Web Server dacă destinația este într-o rețea conectată la R1.

"Ne punem întrebarea cum determină dispozitivul sursă adresa MAC a routerului ?"

Fiecare dispozitiv cunoaște adresa IP a routerului prin adresa IP default gateway configată în setările sale TCP/IP. Adresa default gateway este adresa interfeței routerului conectată la aceeași rețea locală ca și dispozitivul sursă. Toate dispozitivele din rețea folosesc adresa default gateway pentru a transmite mesaje către router. După ce hostul cunoaște adresa IP default gateway, poate folosi ARP pentru a determina adresa MAC a default gatewayului respectiv. Adresa MAC a default gatewayului este apoi introdusă în frame.



Această activitate de simulare ajută la înțelegerea fluxului de trafic și conținuturile pachetelor de date aşa cum călătoresc ele într-o rețea complexă. Comunicațiile vor fi examineate în trei locuri diferite ce simulează rețele tipice de casă sau de afaceri.

3.10 Concluzii Capitolul 3

Rețelele de date sunt sisteme alcătuite din dispozitive finale, dispozitive intermediare și medii ce conectează dispozitivele. Pentru ca să aibă loc comunicarea, aceste dispozitive trebuie să știe cum să comunice.

ACESTE dispozitive trebuie să țină cont de protocolele și regulile de comunicare. TCP/IP este un exemplu de suită de protocole. Multe protocole sunt create de o organizație de standarde, cum ar fi IEFT sau IEEE. Institute of Electrical and Electronics Engineers este o organizație profesională pentru cei care sunt în domeniile electrice, electronice și de inginerie. ISO, International Organization for Standardization, este cel mai mare dezvoltator din lume de standarde internaționale pentru o varietate largă de produse și servicii.

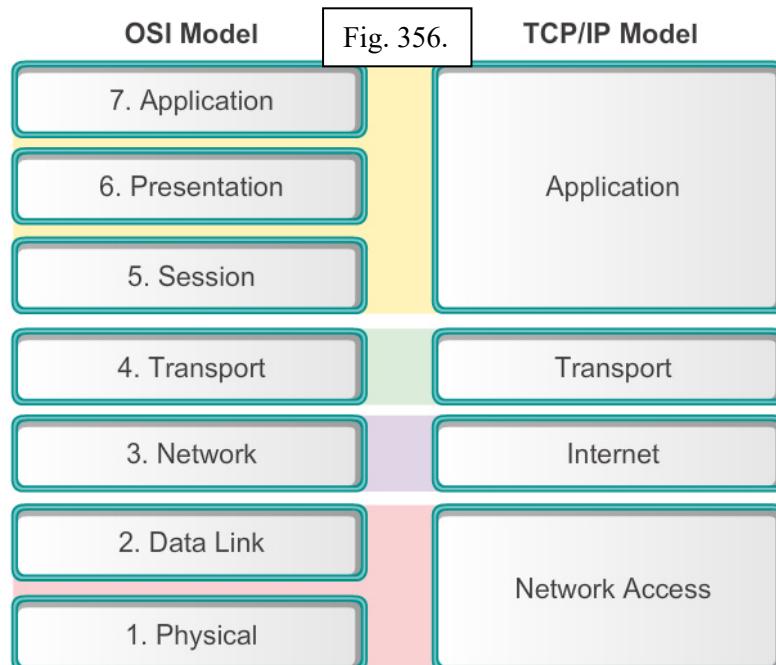
Cele mai utilizate modele din lume sunt modelele OSI și TCP/IP. Asocierea protocolelor ce stau la bază regulilor de comunicații cu nivele diferite ale acestor modele este utilă în determinarea a ce dispozitive și servicii sunt aplicate într-un anumit punct din călătoria datelor peste LANuri sau WANuri.

Datele ce parcurg stiva modelului OSI sunt segmentate în bucăți și încapsulate cu adrese și alte câmpuri. Procesul este inversat prin decapsularea bucătilor și pargurgerea inversă a stivei de protocol la destinație. Modelul OSI descrie procesele de codare, formatare, segmentare și încapsulare a datelor pentru transferul lor peste rețea.

Suita de protocole TCP/IP este un protocol de standard deschis care a fost aprobat de către industria de rețea și de către o organizație de standarde. Suita de protocole Internet este o suită de protocole necesară pentru transferul și primirea de informații cu ajutorul Internetului.

Protocol Data Units (PDUs) sunt numite în concordanță cu protocolele suitei TCP/IP : data, segment, pachet, frame și biți.

Aplicarea modelelor permite indivizilor, companiilor și asociațiilor comerciale analizarea rețelelor actuale și planificarea lor în viitor.



CAPITOLUL 4.ACCESS LA REȚEA

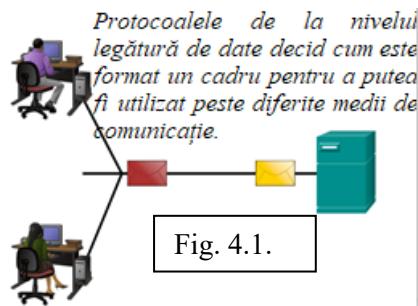
Introducere

Pentru a suporta comunicația noastră, modelul OSI divide funcțiile rețelei de date în nivele. Fiecare nivel lucrează cu nivelele inferioare și superioare lui pentru a transmite date. Două nivele din modelul OSI sunt foarte apropiate, care conform modelului TCP/IP sunt în esență un singur nivel. Aceste două nivele sunt nivelul legătură de date și nivelul fizic.

Pe dispozitivul sursă, nivelul legătură de date are rolul de a pregăti datele pentru transmisie și control al modului în care acestea acceseză mediul fizic. Oricum, nivelul fizic controlează modul în care datele sunt transmise pe mediul fizic prin codarea cifrelor binare ce reprezintă datele în semnale.

La dispozitivul destinatar, nivelul fizic primește semnalele din mediu. După decodarea semnalului și transpunerea înapoi în date, nivelul fizic transmite datele la nivelul legătură de date pentru acceptare și procesare.

Acest capitol începe cu funcțiile generale ale nivelului fizic, standardele și protocolele ce gestionează transmisia datelor prin mediul local. Introduce de asemenea și funcțiile nivelului legătură de date și protocolele asociate lui.



4.1 Protocole de nivel fizic

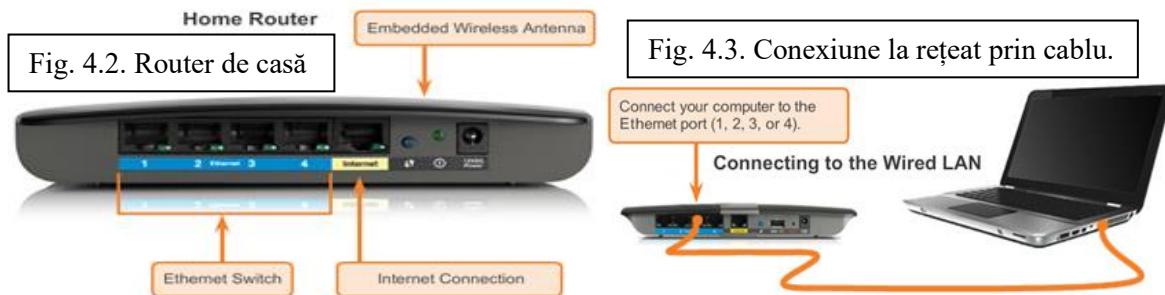
Indiferent dacă se conectează o imprimantă locală la rețeaua de casă sau la un site web dintr-o altă țară, înainte ca orice comunicație de rețea să aibă loc, trebuie să fie stabilită mai întâi o conexiune fizică la o rețea locală. O conexiune fizică poate fi o conexiune cablată folosind un cablu sau o conexiune wireless folosind unde radio.

Tipul conexiunii fizice utilizate este total dependent de configurația rețelei. De exemplu, în mai multe birouri corporative, angajații au desktop sau laptop ce sunt conectate fizic, prin cablu, la un switch partajat. Acest tip de setare este o rețea cablată, în care datele sunt transmise prin cablul fizic.

În plus față de conexiunile cablate, unele instituții ar putea oferi de asemenea și conexiuni wireless pentru laptopuri, tablete și telefoane mobile. Cu dispozitivele wireless, datele sunt transmise prin intermediul undelor radio. Utilizarea conectivității wireless a devenit mai populară odată ce utilizatorii și întreprinderile aferente au descoperit avantajele oferite de serviciile wireless. Pentru a oferi conectivitate wireless, o rețea trebuie să încorporeze un punct de acces wireless (Wireless Access Point – WAP) pentru dispozitivele ce se vor conecta.

Dispozitivele switch și WAP sunt de obicei două dispozitive dedicate separate într-o implementare de rețea. Există de asemenea și dispozitive ce oferă ambele conectivitate, cablată și wireless. În multe cazuri, de exemplu, indivizi implementează servicii de rutare integrate într-un router (home integrated service routers – ISRs), conform Figurii xxx. ISRs oferă o componentă de switching cu porturi multiple, ce permite ca dispozitive multiple să se conecteze la LAN prin

cabluri, conform Figurii xxx. În plus, multe ISRs includ un WAP, ce permite dispozitive wireless să se conecteze la aceeași rețea.



Placa de rețea (Network Interface Cards – NICs) conectează un dispozitiv la rețea. Ethernet NICs sunt utilizate pentru o conexiune cablată, iar WLAN (Wireless Local Area Network) NICs sunt utilizate pentru wireless. Un dispozitiv final ar putea include unul sau ambele tipuri de NICs. O imprimantă de rețea, de exemplu, ar putea avea numai un Ethernet NIC și să se conecteze la rețea printr-un cablu Ethernet. Alte dispozitive, precum tabletele sau smartphonurile, ar putea conține numai un WLAN NIC și trebuie să folosească o conexiune wireless.

Nu toate conexiunile fizice sunt egale, în termenii nivelului de performanță, atunci când sunt conectate la o rețea.

De exemplu, un dispozitiv wireless va întâmpina variații în performanță în funcție de distanță să față de punctul de wireless acces. Cu cât dispozitivul este mai departe de punctul de acces, cu atât semnalul wireless este mai slab. Acest lucru poate însemna o lățime de bandă mai mică sau nici-o conexiune wireless. Fig. xxx arată că un wireless range extender poate fi utilizat pentru a regenera semnalul wireless în alte părți ale casei, care sunt departe față de punctul de acces wireless. Ca alternative, o conexiune cablată nu va scădea în performanță dar, este limitată deplasarea și în general necesită o poziție statică.

Toate dispozitivele wireless trebuie să împartă accesul la undele radio pentru conectarea la punctul de acces wireless, lucru ce implică o performanță de rețea scăzută cu cât crește numărul de dispozitive wireless ce accesează rețea în mod simultan. Un dispozitiv cablat nu necesită să împartă accesul la rețea cu alte dispozitive. Fiecare dispozitiv cablat are un canal separat de comunicații peste propriul său cablu Ethernet. Acest lucru este important atunci când sunt luate în considerare unele aplicații, cum ar fi online gaming, streaming video și conferințe video, ce necesită lățime de bandă dedicată mai mare decât alte aplicații.

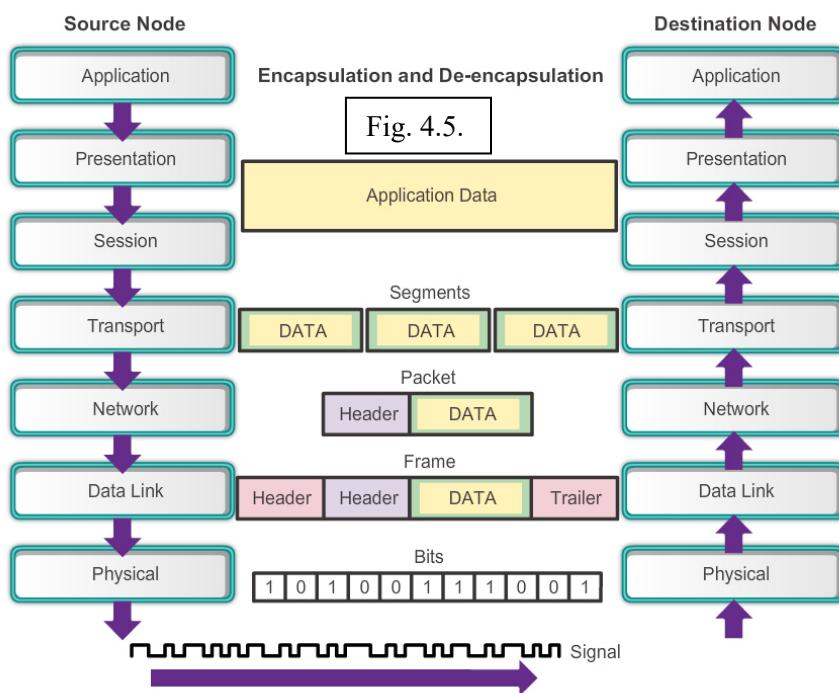


4.2 Scopul nivelului
Fig. 4.4. Conexiune la o rețea fără fir cu un spațiu de extindere

Nivelul fizic din stiva de protocoale OSI oferă mijloace de transport a bițiilor ce alcătuiesc cadrul (frame) de la nivelul legătură de date prin mediul de rețea. Nivelul acceptă un frame complet de la nivelul legătură de date și îl codifică într-o serie de semnale ce sunt transmise peste mediul local. Biții codificați ce alcătuiesc un frame sunt primiți de către un dispozitiv final sau unul intermediar.

Procesul prin către trec datele de la un nod sursă la un nod destinație este:

- Datele utilizatorului sunt segmentate de către nivelul transport, plasate în pachete la nivelul rețea și încapsulate în frameuri de către nivelul legătură de date.
- Nivelul fizic codifică frameurile și crează semnale electrice, optice sau unde radio ce reprezintă biții din fiecare frame.
- Aceste semnale sunt apoi transmise prin mediu pe rând.
- Nivelul fizic de la nodul destinație primește aceste semnale individuale din mediu, le convertește înapoi în reprezentările binare și le transmite către nivelul legătură de date sub forma unui frame complet.



Există trei forme de bază ale mediului de rețea. Nivelul fizic produce reprezentările și grupările de biți pentru fiecare tip de mediu:

- **Cablu de cupru :** Semnale sunt modele de pulsuri electrice.
- **Cablu de fibră optică :** Semnalele sunt modele de lumină.
- **Wireless :** Semnalele sunt modele de transmisii de microunde.

Fig. xxx evidențiază exemple de cablu, fibra optică și wireless.

Pentru a permite interoperabilitate la nivel fizic, toate aspectele acestor funcții sunt guvernate de către organizațiile de standarde.

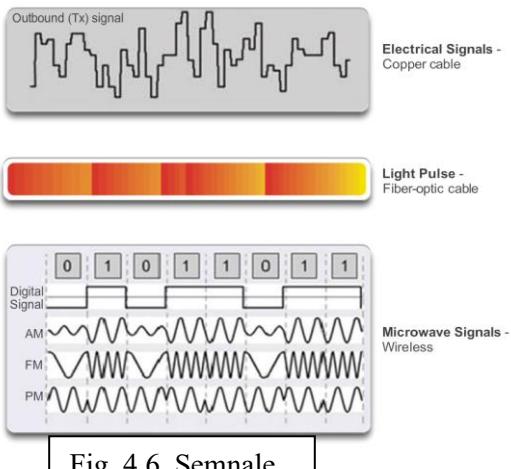


Fig. 4.6. Semnale.

Protocolele și operațiile nivelelor superioare OSI sunt realizate în software proiectat de către ingineri software și specialiști din domeniul IT. De exemplu, serviciile și protocolele din suita TCP/IP sunt definite de către the Internet Engineering Task Force (IETF) în RFCs, conform Figurii xxx.

Nivelul fizic constă din circuite electrice, mediu și conectori dezvoltăți de către ingineri. Prin urmare, este necesar ca standardele ce guvernează aceste componente hard să fie definite de către organizații electrice și de inginerie relevante în comunicații.

Există multe organizații naționale și internaționale diferite, organizații guvernamentale de reglementare și companii private implicate în stabilirea și gestionarea standardelor de la nivelul fizic. De exemplu, standardele hardware, media, de codare și transmisie de la nivelul fizic sunt definite și guvernează de către:

- *International Organization for Standardization (ISO).*
- *Telecommunications Industry Association/Electronic Industries Association (TIA/EIA).*
- *International Telecommunication Union (ITU).*
- *American National Standards Institute (ANSI).*
- *Institute of Electrical and Electronics Engineers (IEEE).*
- *National telecommunications regulatory authorities including the Federal Communication Commission (FCC) în USA și European Telecommunications Standards Institute (ESTI).*

În plus față de acestea, există de obicei grupuri regionale de cablare precum CSA (Canadian Standards Association), CENELEC (European Committee for Electrotechnical Standardization) și JSA/JSI (Japanese Standards Association), elaborând specificații locale.

Fig. 4.7. listează principalii contribuitori și unele dintre standardele de la nivelul fizic relevante.

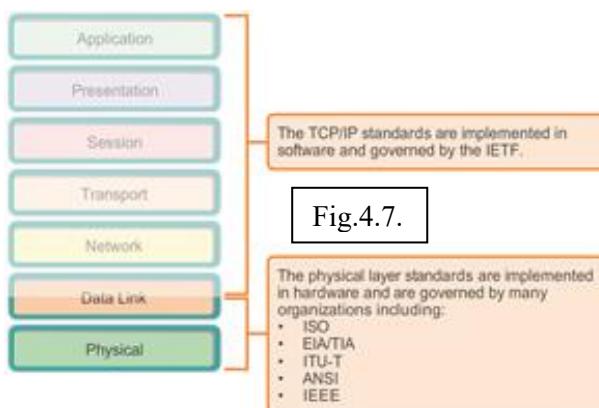


Fig.4.7.

Standard Organization	Networking Standards
ISO	<ul style="list-style-type: none"> ISO 8877: Officially adopted the RJ connectors (e.g., RJ-11, RJ-45). ISO 11801: Network cabling standard similar to EIA/TIA 568.
EIA/TIA	<ul style="list-style-type: none"> TIA-568-C: Telecommunications cabling standards, used by nearly all voice, video, and data networks. TIA-569-B: Commercial Building Standards for Telecommunications Pathways and Spaces. TIA-598-C: Fiber optic color coding. TIA-942: Telecommunications Infrastructure Standard for Data Centers.
ANSI	568-C: RJ-45 pinouts. Co-developed with EIA/TIA.
ITU-T	G.992: ADSL
IEEE	<ul style="list-style-type: none"> 802.3: Ethernet 802.11: Wireless LAN (WLAN) & Mesh (Wi-Fi certification) 802.15: Bluetooth

Fig.4.8.

4.3 Principiile fundamentale ale Nivelului 1

Standardele nivelului fizic adreseaza trei domenii funcționale:

4.3.1 Componente fizice

Componentele fizice sunt dispozitivele electronice hardware, medii de comunicație și alți conectori care transmit și transportă semnalele pentru a reprezenta biți. Componentele hardware precum adaptorii de rețea (NICs), interfețele și conectorii, materialele de cablu și modelele de cablu sunt toate specificate în standarde asociate cu nivelul fizic. Porturile și interfețele variate pe un router Cisco 1941 sunt de asemenea exemple de componente fizice cu pini și conectori specifici rezultați din standarde.

4.3.2 Codificarea

Codificarea este o metodă de conversie a unui flux de biți de date într-un cod predefinit. Codurile sunt grupuri de biți utilizate pentru a oferi un model previzibil ce poate fi recunoscut atât de expeditor, cât și de destinatar. În cazul rețelisticiei, codificarea este un model de tensiune sau curent utilizat pentru a reprezenta biți: 0 și 1.

Pentru crearea codurilor de date, metodele de codificare de la nivelul fizic ar putea de asemenea să ofere coduri pentru scopuri de control, cum ar fi identificarea începutului și sfârșitului unui frame.

Metode comune de codificare utilizate în rețea sunt :

- Codificarea Manchester :** Un 0 este reprezentat prîntr-o tranziție a tensiunii HIGH-LOW și un 1 este reprezentat prîntr-o tranziție a tensiunii LOW-HIGH. Acest tip de codificare este utilizat în versiunile mai vechi de Ethernet, RFID și Near Field Communication.
- Non-Return to Zero (NRZ):** Aceasta este o metodă comună de codificare a datelor ce are două stări "zero" și "unu" și nici-o poziție neutră sau de repaus. Un 0 poate fi reprezentat de către un nivel de tensiune din mediu și un 1 poate fi reprezentat de către un alt nivel de tensiune din mediu.

Notă: Ratele de date mai rapide necesită o codificare mai complexă, cum ar fi 4B/5B, însă, explicarea acestor metode nu intră în scopul acestui curs.

4.3.3 Semnalizarea

Nivelul fizic trebuie să genereze semnale electrice, optice sau wireless ce reprezintă 1 și 0 din mediul de comunicație. Metoda de reprezentare a bițiilor se numește metoda de semnalizare. Standardele nivelului fizic trebuie să definească ce tip de semnal reprezintă un 1 și ce tip de semnal reprezintă un 0. Acest lucru poate fi la fel de simplu precum schimbarea unui nivel al unui semnal electric sau al unui puls optic. De exemplu, un puls lung poate reprezenta un 1, iar un puls scurt poate reprezenta un 0.

Acest lucru este similar modului în care codul Morse este utilizat în comunicații. Codul Morse este o altă metodă de semnalizare ce utilizează o serie de tonuri, lumini sau clickuri pornit-oprit (on-off) pentru a transmite text peste cablurile telefonice sau între navele marine.

Semnalele pot fi transmise în unul dintre cele două moduri:

- **Asincron** : *Semnalele de date sunt transmise fără un semnal de ceas asociat. Spațiul de timp dintre caracterele de date sau blocuri poate avea o durată arbitrară, acest lucru însemnând că spațiul nu este standardizat. Frameurile necesită indicațioare steag de start și de stop.*
- **Sincron** : *Semnalele de date sunt transmise cu un semnal de ceas care are loc la durațe de timp uniform distribuite, numite "bit time".*

Există multe moduri de transmisie a semnalelor. O metodă comună de transmisie a datelor este utilizarea tehnicii de modulare. Modularea este procesul prin care caracteristica unei unde (semnalul) modifică o altă undă (transportatorul). Următoarele tehnici de modulare au fost utilizate global în transmiterea datelor pe mediu:

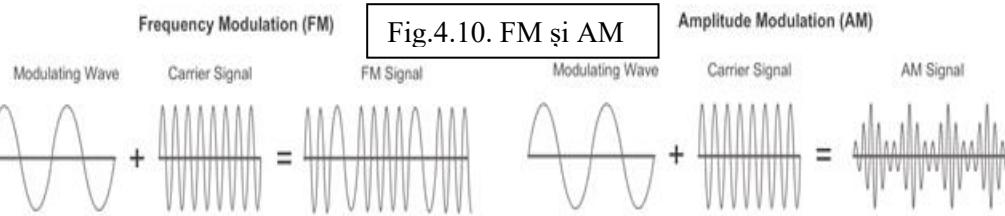
- **Modularea frecvenței (FM)**: *O metodă de transmisie în care frecvența transportatorului variază în funcție de semnal.*
- **Modularea în amplitudine (AM)**: *O tehnică de transmisie în care amplitudinea transportatorului (purtătorului) variază în funcție de semnal.*
- **Pulse-coded modulation (PCM)**: *O tehnică în care un semnal analog, cum ar fi o voce, este convertit într-un semnal digital prin eșantionarea amplitudinii semnalului și prin exprimarea diferențelor amplitudinii ca număr binar. Rata de eșantionare trebuie să fie cel puțin de două ori mai mare decât frecvența din semnal.*

Natura semnalelor ce reprezintă biții pe mediu va depinde de metoda de semnalizare utilizată. Unele metode ar putea utiliza un atribut al semnalului pentru a reprezenta un singur 0 și un alt atribut pentru a reprezenta un singur 1.

Fig. 4.9. ilustrează modul în care tehniciile AM și FM sunt utilizate pentru a transmite un semnal.

Media	Physical Components	Frame Encoding Technique	Signalling Method
Copper cable	<ul style="list-style-type: none"> • UTP • Coaxial • Connectors • NICs • Ports • Interfaces 	<ul style="list-style-type: none"> • Manchester Encoding • Non-Return to Zero (NRZ) techniques • 4B/5B codes are used with Multi-Level Transition Level 3 (MLT-3) signaling • 8B/10B • PAM5 	<ul style="list-style-type: none"> • Changes in the electromagnetic field • Intensity of the electromagnetic field • Phase of the electromagnetic wave

Fig.4.9. Tehnicile AM și FM



Medii fizice diferite suportă transferul de biți la viteze diferite. Transferul de date este de obicei discutat în termenii de "bandwidth" și "throughput".

Bandwidth reprezintă capacitatea mediului de a transporta datele. Digital bandwidth măsoară cantitatea de date ce poate fi transportată de la un loc la altul într-un anumit timp. Bandwidth este măsurată, de obicei, în kilobits per second (kb/s) sau megabits per second (Mb/s).

Lățimea de bandă concretă a rețelei este determinată de către o combinație de factori:

- *Proprietățile mediului fizic.*
- *Tehnologiile alese pentru semnalizarea și detectarea semnalelor de rețea.*

Proprietățile mediului fizic, tehnologiile curente și legile fizice joacă un rol important în determinarea lățimii de bandă disponibilă.

Unit of Bandwidth	Abbreviation	Equivalent
Bits per second	bps	1 bps = fundamental unit of bandwidth
Kilobits per second	kbps	1 kbps = 1,000 bps = 10^3 bps
Megabits per second	Mbps	1 Mbps = 1,000,000 bps = 10^6 bps
Gigabits per second	Gbps	1 Gbps = 1,000,000,000 bps = 10^9 bps
Terabits per second	Tbps	1 Tbps = 1,000,000,000,000 bps = 10^{12} bps

Tabelul arată unitățile utilizate în mod normal pentru măsurarea lățimii de bandă.

Throughput reprezintă mărimea transferului de biți prin mediu într-o perioadă de timp.

Având în vedere un număr de factori, throughput, de obicei, nu corespunde lățimii de bandă specificată în implementările de la nivelul fizic. Mai mulți factori influențează throughput, precum:

- *Cantitatea de trafic.*
- *Tipul de trafic.*
- *Latența creată de numărul de dispozitive de rețea întâlnite de la sursă la destinație.*

Latența se referă la catitatea de timp, inclusiv întârzierile, necesară pentru ca datele să călătorescă de la un punct dat la altul.

Într-o internetwork sau rețea cu multiple segmente, throughput nu poate fi mai rapid decât cea mai înceată legătură sau cale de la sursă la destinație. Chiar dacă toate sau cele mai multe segmente au lățime de bandă mare, este necesar doar un segment în cale cu throughput scăzut pentru a crea o gătuire (bottleneck) a throughputului din întreaga rețea.

Există mai multe teste online de viteză care pot arăta throughputul unei conexiuni Internet. Fig. xxx oferă rezultatele pentru un test de viteză.

Notă: Există o a treia măsurătoare a transferului datelor utilizabile, cunoscută ca **goodput**. **Goodput** reprezintă măsura datelor utilizabile transferate într-o anumită perioadă de timp. Goodput este throughput minus traffic overhead pentru stabilirea sesiunilor, acknowledgements și încapsulare.



Fig. 4.12.

Nivelul fizic produce reprezentarea și gruparea bițiilor în tensiuni, frecvențe radio sau pulsuri de lumină. Mai multe organizații de standarde au contribuit la definiția proprietăților fizice, electrice și mecanice ale mediului disponibil pentru diferite comunicații de date. Aceste specificații garantează faptul că respectivele cabluri și conectori vor funcționa corect pe diferite implementări ale nivelului legătură de date.

Ca un exemplu, standardele pentru mediul de cupru sunt definite pentru:

- *Tipul de cabluri de cupru utilizate.*
- *Lățimea de bandă a comunicației.*
- *Tipul de conectori utilizați.*
- *Pini și codurile de culoare ale conexiunilor la mediu.*
- *Distanța maximă a mediului.*

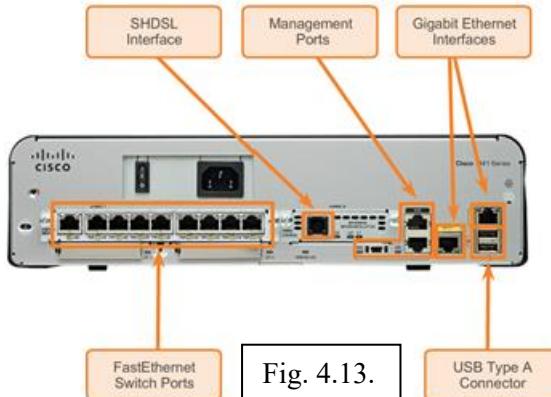


Fig. 4.13.

4.4 Mediile de comunicații de rețea

4.4.1 Cabluri de cupru

Rețelele utilizează mediul de cupru deoarece nu este scump, este ușor de instalat și are rezistență scăzută la curentul electric. Însă, mediul de cupru este limitat de distanță și de interferență semnalului.

Datele sunt transmise prin cabluri de cupru ca pulsuri electrice. Un detector din interfață de rețea a dispozitivului destinație trebuie să primească un semnal care poate fi decodificat cu succes pentru a corespunde semnalului transmis. Însă, cu cât semnalul călătorește mai mult, cu atât se deteriorează mai mult, fenomen numit atenuarea semnalului. Din acest motiv, tot mediul de cupru trebuie să urmeze limitări de distanță stricte specificate de către standarde.

Valorile de timp și tensiune ale pulsurilor electrice sunt de asemenea sensibile la interferențele dintre două surse:

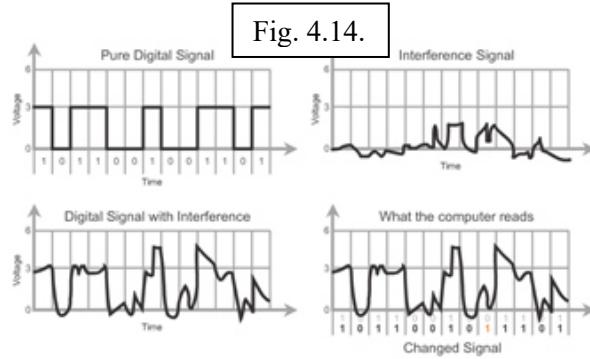
- **Interferența electromagnetică (EMI) sau interferența frecvenței radio (RFI)** – semnalele EMI și RFI pot corupe și deforma semnalele de date transportate de mediul de cupru. Sursele potențiale ale EMI și RFI includ undele radio și dispozitivele electomagnetic precum lumini fluorescente sau motoare electrice, conform Figurii xxx de mai jos.
- **Crosstalk** – este o perturbare cauzată de către câmpurile electrice sau magnetice ale unui semnal dintr-un cablu adiacent. În circuitele telefonice, crosstalk poate rezulta într-o parte a altelui conversații dintr-un circuit adiacent. Mai exact, atunci când curentul electric curge printr-un cablu, crează un mic câmp magnetic circular în jurul cablului care poate fi preluat de către un alt cablu adiacent.

Pentru a contracara efectele negative ale EMI și RFI, unele tipuri de cabluri de cupru sunt înfășurate într-un scut metalic și necesită conexiuni adecvate de pregătire elementară.

Pentru a contracara efectele negative ale crosstalkului, diverse tipuri de cabluri de cupru au perechi de fire răsucite împreună ce anulează în mod eficient crosstalkul.

Susceptibilitatea cablurilor de cupru la zgomotul electric poate fi limitată de asemenea de:

- Selectarea tipului sau categoriei de cablu cea mai potrivită pentru un mediu de rețea dat.
- Proiectarea unei infrastructuri de cablu pentru evitarea surselor potențiale și cunoscute de interferență din structura clădirii.
- Utilizarea tehniciilor de cablare ce includ manipularea și terminarea corectă a cablurilor.



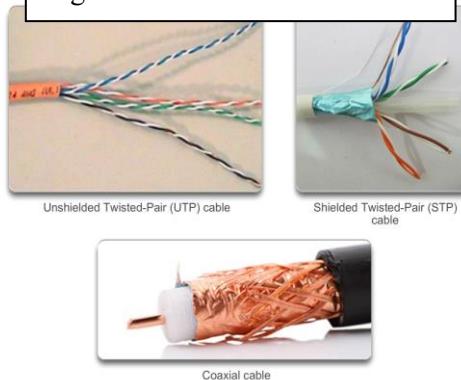
Există trei principale tipuri de mediu de cupru utilizate în rețea:

- **Unshielded Twisted-Pair (UTP).**
- **Shielded Twisted-Pair (STP).**
- **Coaxial.**

Acste cabluri sunt utilizate pentru interconectarea nodurilor dintr-un LAN și dispozitivele de infrastructură precum switchuri, routere și puncte de acces wireless. Fiecare tip de conexiune și dispozitivele corespunzătoare au cerințe de cablare specifice de către standardele nivelului fizic.

Diferite standarde ale nivelului fizic specifică utilizarea de conectori diferiți. Aceste standarde specifică dimensiunile mecanice ale conectorilor și proprietățile electrice acceptate ale fiecărui tip. Mediul de rețea utilizează conectori și prize modulare pentru a oferi conectare și deconectare ușoară. De asemenea, un singur tip de conector fizic poate fi utilizat pentru mai multe tipuri de conexiuni. De exemplu, conectorul RJ-45 este utilizat pe scară largă în LANuri cu un tip de mediu și în WANuri cu alt tip de mediu.

Fig. 4.15. Cabluri de conexiune.

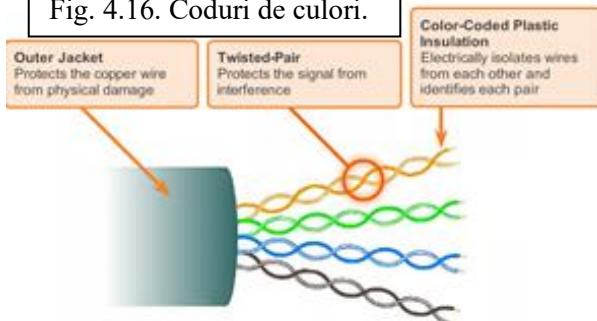


Cablul Unshielded Twisted-Pair (UTP) este cel mai cunoscut mediu de rețea. Cablul UTP, terminat cu conectori RJ-45, este utilizat pentru interconectarea hosturilor de rețea cu dispozitivele intermediare de rețea, cum ar fi switchuri și routere.

În LANuri, cablul UTP constă din patru perechi de fire colorate ce au fost răsucite împreună și apoi încorporate într-un scut din plastic flexibil ce protejează împotriva daunelor fizice minore. Această răsucire a firelor ajută la protejarea împotriva interferenței cu semnal de la diverse surse.

Conform Figuri 4.16., codurile de culoare identifică perechile individuale și firele din perechi și ajută la finalizarea corectă a cablului.

Fig. 4.16. Coduri de culori.



Shielded Twisted-Pair (STP) oferă o protecție mai bună împotriva zgomotului decât UTP. Cu toate acestea, în comparație cu cablul UTP, cablul STP este semnificativ mai scump și dificil de instalat. Ca și cablul UTP, STP utilizează un conector RJ-45.

Cablul STP combină tehniciile de protecție ale EMI și RFI și firele răsucite pentru a contracara crosstalkul. Pentru a câștiga întregul beneficiu al protecției, cablurile STP sunt terminate cu conectori de date STP special protejați. În cazul în care cablul este construit neadecvat, scutul ar putea funcționa ca o antenă și recepționa semnale nedorite.

Tipuri diferite de cabluri STP cu diferite caracteristici sunt disponibile. Există două variații comune ale STP:

- Cablul STP ce protejează întregul pachet cu fire cu folie, eliminând practic orice interferență (mai comun).
- Cablul STP ce protejează întregul pachet de fire, precum și perechile de fire individuale cu folie, eliminând toate interferențele.

Cablul STP prezentat utilizează patru perechi de fire, fiecare înfășurată într-o folie scut, ce sunt apoi înfășurate într-o țesătură metalică sau folie.

Pentru mulți ani, STP a fost structura de cablu specificată pentru utilizarea în instalații de rețele Token Ring. O dată cu declinul Token Ring cererea pentru STP s-a diminuat. Cu toate acestea, noul standard 10GB pentru Ethernet are o prevedere pentru utilizarea cablului STP ce oferă o reînoire a interesului pentru STP.

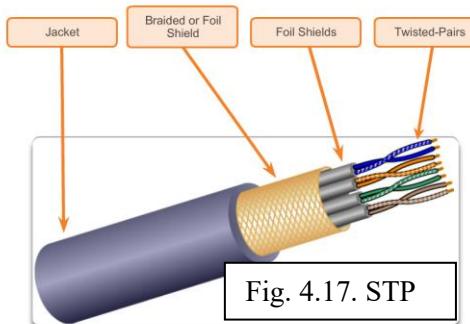


Fig. 4.17. STP

Cablul coaxial, sau coax pe scurt, își ia numele din faptul că există două conductoare ce împart aceeași axă. Ca și în Fig. xxx, cablul coaxial constă din:

- Un conductor de cupru utilizat pentru a transmite semnalele electrice.

- *Conductorul de cupru este înconjurat de un strat de izolație flexibilă din plastic.*
- *Materialul izolator este înconjurat de o impletitură de cupru, sau folie metalică, ce acționează ca un al doilea fir în circuit și ca scut pentru conductorul interior. Acest al doilea nivel, sau scut, reduce de asemenea cantitatea de interferențe electromagnetice exterioare.*

Notă: Există tipuri diferite de conectori utilizați cu cablul coaxial.

Cablul coaxial a fost utilizat în televiziune prin cablu capabil să transmită într-o singură direcție. A fost de asemenea utilizat în rețelele Ethernet de la început.

Deși cablul UTP a înlocuit cablul coaxial în rețelele Ethernet moderne, designul cablului coaxial a fost adaptat pentru a fi utilizat în:

- **Rețele wireless:** Cablurile coaxiale atașează antenele la dispozitivele wireless. Cablul coaxial poartă energie de radiofreqvență (RF) între antene și echipamentul radio.
- **Rețele de Internet prin cablu:** Furnizorii de serviciu prin cablu, în prezent, convertesc sistemele lor one-way în sisteme two-way pentru a oferi conectivitate la Internet clienților lor. Pentru a furniza aceste servicii, porțiuni de cablu coaxial și elemente de suport de amplificare sunt înlocuite cu cablu de fibră-optică. Însă, conexiunea finală de la locația clientului și cablul din interiorul locației clientului este încă un cablu coaxial. Această combinație de fibră și cablu coaxial se numește hybrid fiber coax (HFC).

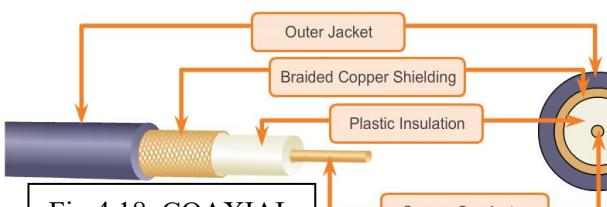


Fig.4.18. COAXIAL



Toate trei tipurile de medii de cupru sunt sensibile la foc și la dezastrele electrice.

Pericolele de incendiu există deoarece izolația cablurilor și scuturile pot fi imflamabile sau pot produce vapori toxici atunci când se încălzesc sau sunt arse. Autoritățile sau organizațiile de construcție pot prevedea standarde de siguranță pentru instalările hardware și de cabluri.

Pericolele electrice sunt o problemă potențială deoarece firele de cupru pot conduce electricitate în moduri nedorite. Acest lucru ar putea supune personalul și echipamentul la o serie de pericole electrice. De exemplu, un dispozitiv de rețea defect poate conduce curenți către șasiul altor dispozitive de rețea. În plus, cablarea rețelei poate prezenta nivele de tensiune nedorite atunci când este utilizată pentru conectarea dispozitivelor ce au surse de alimentare cu diferite potențiale de masă. Asemenea situații sunt posibile atunci când cablul de cupru este utilizat pentru conectarea rețelelor din diferite clădiri sau de la etaje diferite sau clădiri ce utilizează facilități de alimentare diferite. Cablul de cupru ar putea conduce tensiuni cauzate de fulgere de lumină la alte dispozitive.

Rezultatul tensiunilor și curenților nedoriți poate fi defectarea dispozitivelor de rețea și a computerelor conectate sau rănirea personalului. Este important ca acest cablu de cupru să fie instalat adevarat și în concordanță cu specificațiile relevante și codurile clădirii, pentru a evita potențialele pericole și situații de defectare.

Fig. 4.19. arată practici de cablare adecvate pentru evitarea potențialelor pericole electrice și de incendiu.

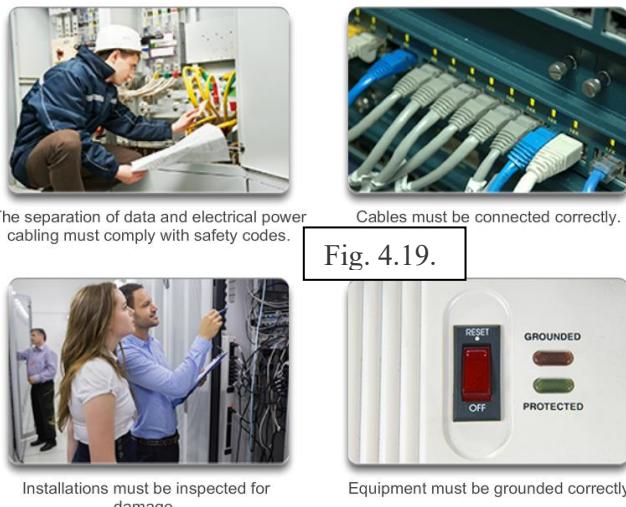


Fig. 4.19.

4.4.1.1 Cablarea UTP

Atunci când sunt utilizate ca mediu de rețea, cablurile UTP constau din patru perechi de fire colorate ce au fost răsucite împreună și apoi învelite într-un scut flexibil de plastic. Cablul UTP de rețea are patru perechi de fire de cupru de ecartament 22 sau 24. Un cablu UTP are un diametru extern de aproximativ 0.43 cm (0.17 inches) și dimensiunea mică a să poate fi avantajoasă în timpul instalării.

Cablul UTP nu utilizează protecție pentru combaterea efectelor EMI și RFI. În schimb, proiectanții cablului au descoperit că pot limita efectul negativ al crosstalkului prin:

- **Neutralizare (anulare):** Proiectanții pun fire într-un circuit. Atunci când două fire dintr-un circuit electric sunt aduse unul lângă celălalt, câmpurile lor magnetice sunt opuse unui față de celălalt. Prin urmare, cele două câmpuri magnetice se anulează unul pe celălalt și astfel, anulează orice semnal exterior EMI și RFI.
- **Variată numărului de răsuciri pe pereche:** Pentru a îmbunătăți și mai mult efectul anular, designerii variază numărul de răsuciri al fiecărei perechi de fire dintr-un cablu. Cablul UTP trebuie să urmeze specificații precise ce dau detalii cu privire la numărul de răsuciri sau impletituri ce sunt permise pe metru sau pe cablu. Se poate remarcă în Fig. 4.20 că perechile portocaliu/alb-portocaliu sunt răsucite mai puțin decât perechile albastru/alb-albastru.

Cablul UTP Fig. 4.20 se bazează numai pe efectul de anulare produs de perechile de fire torsadate pentru limitarea degradării semnalului și oferă autoprotecție eficientă pentru perechile de fire din mediul de rețea.

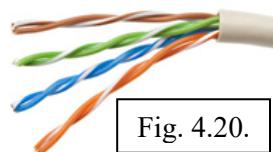


Fig. 4.20.

Cablurile UTP se conformează standardelor stabilite de către TIA/EIA. Mai precis, TIA/EIA-568A specifică standardele de cablare comerciale pentru instalații LAN și este standardul cel mai comun utilizat în mediile de cablare LAN. Unele dintre elementele definite sunt:

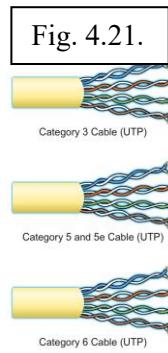
- *Tipurile de cablu.*
- *Lungimile cablului.*
- *Conecțori.*
- *Terminarea cablului.*
- *Metode de testare a cablului.*

Caracteristicile electrice ale cablului de cupru sunt definite de către *"Institute of Electrical and Electronics Engineers (IEEE)"*. IEEE evaluează cablul UTP în funcție de performanță să. Cablurile sunt plasate în categorii în funcție de abilitatea lor de a transporta rate de lățime de bandă mai mari. De exemplu, cablul de Categoria 5(Cat5) este utilizat în instalații 100BASE-TX FastEthernet. Alte categorii includ cablurile de Categoria 5 (Cat5e), Categoria 6 (Cat6) și Categoria 6a.

Cablurile din categorii mai mari sunt proiectate și construite pentru a suporta rate de date mai mari. Deoarece noi tehnologii Ethernet de viteză gigabit au fost dezvoltate și preluate, Cat5e este acum tipul de caplu minim acceptat, iar Cat6 este tipul recomandat pentru noile instalări din clădiri.

Fig. 4.21 ilustrează categoriile diferite ale cablării UTP.

Notă: Unii producători produc cabluri ce depășesc specificațiile TIA/EIA Category 6a și fac parte din Categoria 7.



Cabul UTP este în mod normal terminat cu un standard ISO 8877 specificat ca un conector RJ-45. Acest conector este utilizat pentru o serie de specificații de nivel fizic, printre care se află și Ethernet. Standardul TIA/EIA 568 descrie codurile de culoare ale firelor pentru cablurile Ethernet.

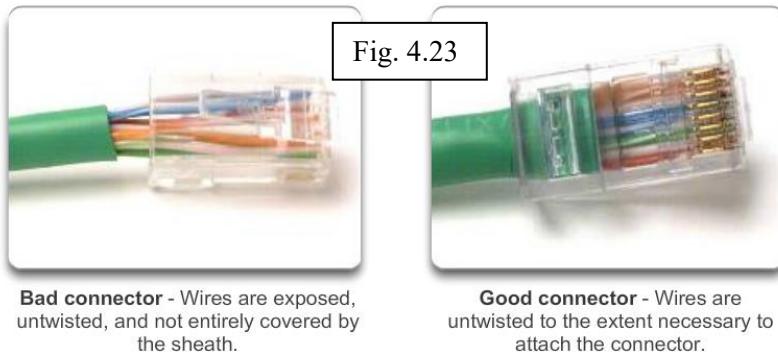
Așa cum este prezentat în Fig. 4.22, conectorul RJ-45 este componenta “masculină” exterioară, “zimțată”, de la capătul cablului. “Socket” este componenta feminină dintr-un dispozitiv de rețea, priza de portiune mică de pe perete sau patch panel.



De fiecare dată când cablul de cupru este terminat există posibilitatea pierderii de semnal sau introducerea de zgomot în circuitul de comunicație. Atunci când este terminat neadecvat, fiecare cablu este o sursă potențială de scădere a performanței nivelului fizic. Este esențial ca

toate terminațiile mediului de cupru să fie de calitate superioară pentru a asigura performanțe optime ale tehnologiilor actuale și viitoare.

Fig. 4.23 evidențiază un exemplu de cablu UTP terminat în mod neadecvat și un cablu UTP terminat în mod adekvat.



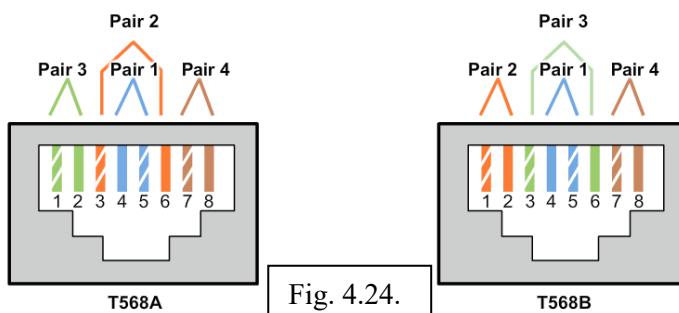
Anumite situații ar putea necesita cabluri UTP care să fie terminate în mod diferit în funcție de diverse convenții.

Următoarele sunt tipurile de cablu principale obținute prin utilizarea convențiilor de terminare:

- **Ethernet Straight-Through** : Cel mai comun tip de cablu de rețea. Este utilizat în mod normal pentru a interconecta un host la un switch sau un switch la un router.
- **Ethernet Crossover**: Un cablu obișnuit utilizat pentru interconectarea dispozitivelor similare. De exemplu, pentru a conecta un switch la un switch, un host la un host sau un router la un router.
- **Rollover**: Un cablu proprietar Cisco utilizat pentru conectarea la portul de consolă a unui router sau switch.

Utilizarea incorectă a unui cablu, crossover sau straight-through, între dispozitive ar putea să nu afecteze dispozitivele, însă comunicarea sau conectivitatea dintre dispozitive nu va avea loc. Aceasta este o eroare comună în laborator și verificarea dacă sunt efectuate conexiunile corect ar trebui să fie prima acțiune de depanare în cazul în care conectivitatea nu este realizată.

Fig. xxx arată tipurile de cablu UTP, standardele și aplicația tipică a acestor cabluri. Identifică de asemenea perechile de fire individuale pentru standardele TIA 568A și TIA 568B.



Cable Type	Standard	Application
Ethernet Straight-through	Both ends T568A or both ends T568B	Connects a network host to a network device such as a switch or hub.
Ethernet Crossover	One end T568A, other end T568B	<ul style="list-style-type: none"> Connects two network hosts Connects two network intermediary devices (switch to switch, or router to router)
Rollover	Cisco proprietary	Connects a workstation serial port to a router console port, using an adapter.

După instalare, un tester de cablu UTP ar trebui să fie utilizat pentru a testa următorii parametrii:

- *Harta distribuției firelor (wire map).*
- *Lungimea cablului.*
- *Pierdere de semnal în timpul atenuării.*
- *Crosstalk.*

Este recomandat să se verifice dacă toate cerințele de instalare UTP sunt indeplinite.

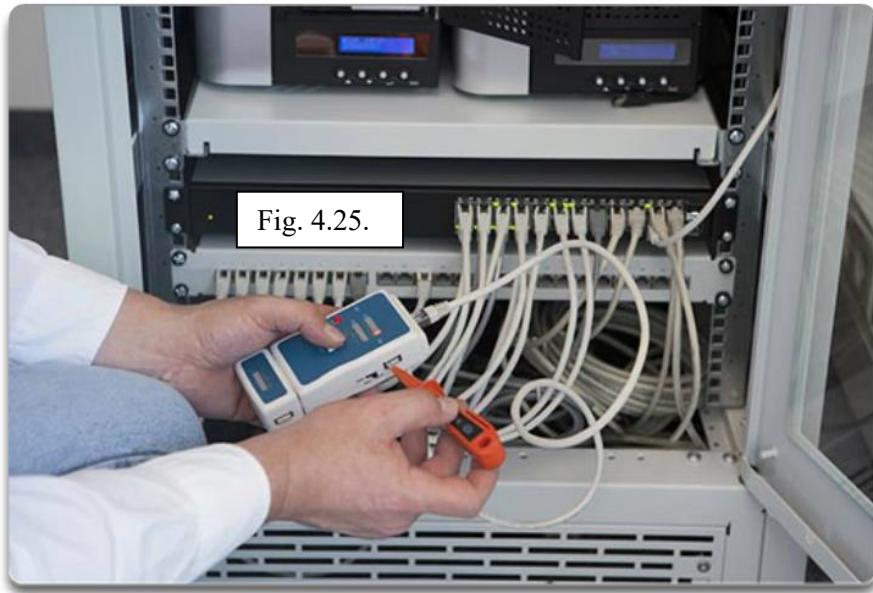


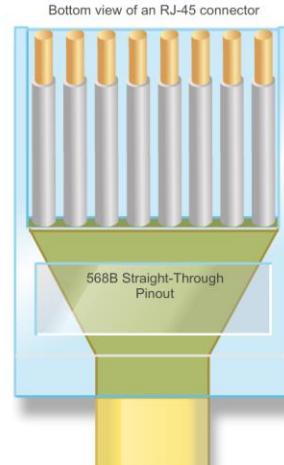
Fig. 4.25.

Correctly align the wire colors to build a UTP 568B, straight-through cable pinout.

Drag each wire color to its correct placement on the RJ-45 image in the graphic.



Fig. 4.26.



Bottom view of an RJ-45 connector

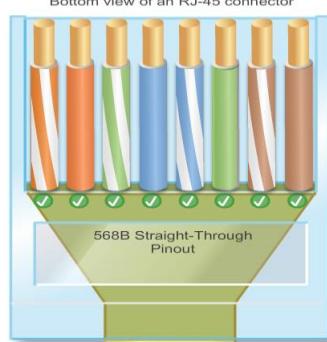


Fig. 4.27.

4.4.1.2 Cablarea cu Fibră Optica

Cablul de fibră optică a devenit foarte popular pentru interconectarea dispozitivelor de rețea din infrastructură. Permite transmisia datelor peste distanțe mari și lățimi de bandă mai mari decât orice alt mediu de rețea.

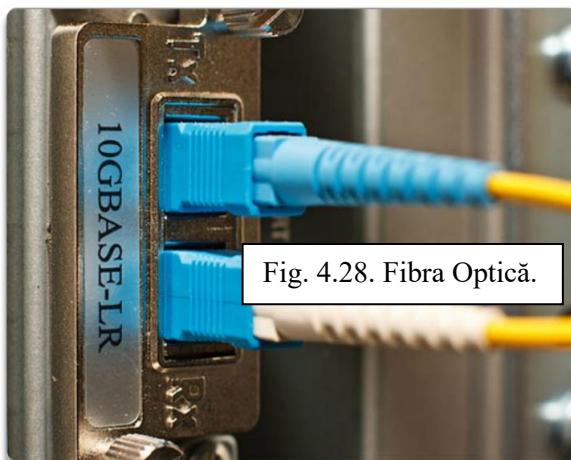
Fibra optică este un fir flexibil, dar foarte subțire și transparent de sticlă foarte pură, nu cu mult mai mare decât un fir de păr uman. Biții sunt codificați pe fibră în impulsuri luminoase. Cablul de fibră optică ca un ghid de undă, sau “conductă de lumină”, pentru a transmite lumina între două capete cu o pierdere minimă de semnal.

Spre deosebire de cablurile de cupru, cablul de fibră optică poate transmite semnale cu o atenuare mai mică și imune complet la EMI și RFI.

Cablarea cu fibra optică este acum utilizată în patru tipuri de industrii:

- **Rețele de întreprindere:** Fibra este utilizată pentru aplicații de cablare backbone și interconectează dispozitive de infrastructură.
- **Rețele FTTH și de acces:** Fiber-to-the-home (FTTH) este utilizată pentru a oferi servicii always-on broadband pentru case și întreprinderi mici. FTTH suportă acces la Internet de mare viteză, cum ar fi telecommuting, telemedicine și video la cerere.
- **Rețele “de distanțe mari” :** Furnizorii de servicii utilizează rețele terestre de fibră optică de “distanțe mari” pentru a conecta țări și orașe. Rețelele în mod normal variază de la câteva zeci la câteva mii de kilometri și utilizează sisteme de până la 10Gb/s.
- **Rețele submarine:** Cabluri de fibră optică speciale sunt utilizate pentru a oferi soluții de încredere de mare viteză și capacitate mare, capabile de supraviețuire în medii subacvatice dure pe distanțe transoceane.

Noi ne axăm pe utilizarea fibrei în întreprinderi.



Deși o fibra optică este foarte subțire, este compusă din două feluri de sticlă și un scut exterior protector. Aceste componente sunt:

- **Nucleul:** Conține sticlă pură și este partea fibrei unde este purtată lumina.
- **Scheletul:** Sticla ce înfășoară nucleul și se comportă ca o oglindă. Pulsurile de lumină se propagă în nucleu pe când scheletul reflectă pulsurile de lumină. Aceasta păstrează pulsurile conținute în nucleul de fibră într-un fenomen cunoscut sub numele de reflexie totală internă.
- **Mantaua:** Este de obicei, o manta PVC ce protejează nucleul și scheletul. Ar putea să asemenea să conțină materiale de întărire și un buffer al cărui scop este protecția sticlei împotriva zgârieturilor și umidității.

Deși sensibile la îndoiri accentuate, proprietățile nucleului și ale scheletului au fost alterate la un nivel molecular pentru a le face foarte puternice. Fibra optică a fost testată printr-un proces riguros de fabricație pentru rezistență la un minim de *100,000 pounds per square inch*. Fibra optică este suficient de rezistentă pentru a rezista în tipul instalării și implementării în condiții dure de mediu în rețelele din toată lumea.

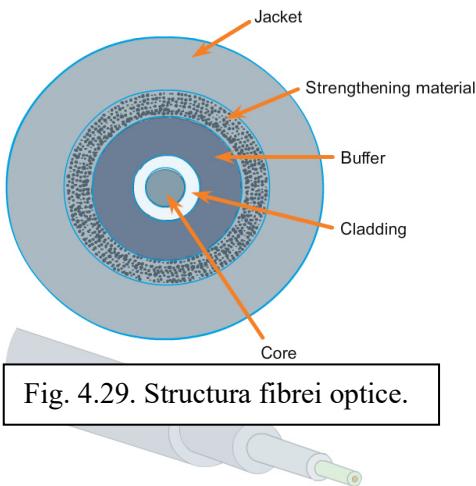


Fig. 4.29. Structura fibrei optice.

Pulsurile de lumină reprezentate de datele transmise sub formă de biți pe mediu sunt generate de :

- *Lasere.*
- *Light emitting diodes (LEDs).*

Dispozitivele electronice semi-conductoare numite fotodiode detectează pulsurile electrice și le convertește în tensiuni ce pot fi reconstruite în frameurile de date.

Notă: Lumina de laser transmisă peste cablul de fibră optică poate afecta ochiul uman. Trebuie avut grijă să se evite privirea în capătul unei fibre optice active.

Cablurile de fibră optică pot fi clasificate în mare în două tipuri:

- **Single-mode fiber (SMF):** Conține un nucleu foarte mic și utilizează tehnologie laser foarte scumpă pentru a transmite o singură rază de lumină. Este populară în situații ce implică o distanță lungă, se întinde pe sute de kilometri, cum este necesar în telefonie și aplicații de cablu TV.
- **Multimode fiber (MMF):** Conține un nucleu mare și utilizează emițători LED pentru a transmite pulsuri de lumină. Mai exact, lumina de la un LED intră în fibra multimode din unghiuri diferite. Este populară în LANuri deoarece poate fi alimentată de leduri de cost scăzut. Ofere o lățime de bandă până la 10Gb/s pe distanțe de până la 550 m.

Figurile 4.30 și 4.31 evidențiază caracteristicile fibrei multimode și single-mode. Una dintre cele mai evidente diferențe dintre fibra multimode și cea single-mode este cantitatea de dispersie. Dispersia se referă la răspândirea pulsului de lumină în timp. Cu cât este mai mare dispersia, cu atât pierderea de putere a semnalului este mai mare.

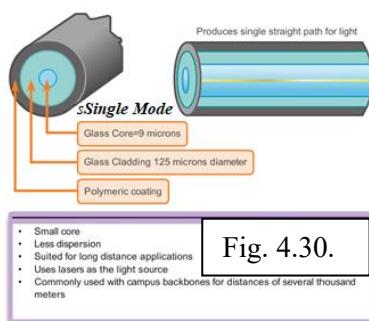


Fig. 4.30.

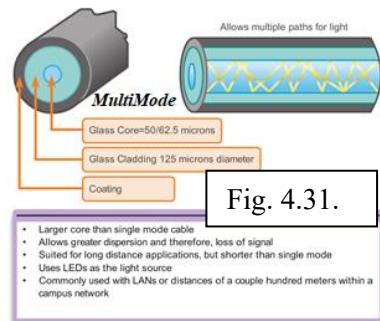


Fig. 4.31.

- Larger core than single mode cable
- Allows greater dispersion and therefore, loss of signal
- Suited for long distance applications, but shorter than single mode
- Uses LEDs as the light source
- Commonly used with LANs or distances of a couple hundred meters within a campus network

Un conector de fibră optică termină fibra optică. Sunt disponibili mai mulți conectori de fibră optică. Principalele diferențe dintre tipurile de conectori sunt dimensiunile și metodele de cuplare mecanice. În general, organizațiile se standardizează pe un tip de conector, în funcție de echipamentul utilizat de ei în general, sau se standardizează pe tipul fibrei (unul pentru MMF, unul pentru SMF). Având în vedere toate generațiile de conectori, astăzi sunt disponibili aproximativ 70 de conectori.

Ca și în Fig. xxx 1, există trei conectori de fibră optică cei mai populari:

- **Straight-Tip (ST):** Un conector stil baionetă utilizat pe scară largă cu fibra multimode.
- **Subscriber Connector (SC):** Numit uneori și conector standard sau conector pătrat. Este un conector utilizat pe scară largă în LAN și WAN ce utilizează un mecanism push-pull pentru a asigura o inserție pozitivă. Acest tip de conector este utilizat cu fibra multimode și single-mode.
- **Lucent Connector (LC):** Numit uneori conector mic sau conector local, se află în creștere rapidă în popularitate datorită dimensiunii sale reduse. Este utilizat cu fibra single-mode și suportă și fibra multimode.

Notă: alți conectori de fibră precum Ferrule Connector (FC) și Sub Miniature A (SMA) nu sunt populari în LAN sau WAN. Conectorii învechiți includ conectorii biconic (obsolete) și D4. Acești conectori nu intră în scopul acestui capitol.

Deoarece lumina poate călători numai într-o singură direcție peste fibra optică, două fibre sunt necesare pentru a suporta operația full duplex. Prin urmare, patch cablurile de fibră optică împreunează două cabluri de fibră optică și le termină cu o pereche de conectori standard de fibră optică. Unii conectori de fibră acceptă ambele fibre, de transmitere și de recepție, într-un singur conector, numit conector duplex, ilustrat în Fig. xxx 1.

”Fiber patch cords” sunt necesare pentru interconectarea dispozitivelor de infrasctructură. De exemplu, Fig. xxx 2 ilustrează mai multe patch cords comune:

- *SC-SC multimode patch cord.*
- *LC-LC single-mode patch cord.*
- *ST-LC multimode patch cord.*
- *SC-ST single-mode patch cord.*

Cablurile de fibră ar trebui să fie protejate cu un capac de plastic mic atunci când nu sunt utilizate.

Se poate observa de asemenea utilizarea culorii pentru a distinge single-mode și multimode patch cords. Motivul este acela ca standardul TIA-598 recomandă utilizarea unei mante de culoare galbenă pentru cablurile de fibră single-mode și de culoare portocalie pentru cablurile de fibră multimode.

Fig. 4.32. Conectori pentru Fibra Optică

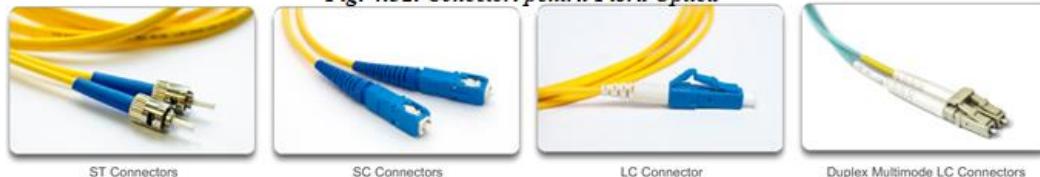


Fig. 4.33. Cabluri de Conexiune pentru Fibra Optică



Terminarea și îmbinarea cap la cap a cablajului de fibră optică necesită un echipament și un training special. Terminarea incorectă a mediului de fibră optică va avea ca rezultat diminuarea distanțelor de semnal sau eșecul transmisiei complet.

Trei tipuri de erori comune de terminare și îmbinare a fibrei optice sunt:

- **Lipsa de aliniere:** *Mediile de fibră optică nu sunt aliniate precis atunci când sunt îmbinate.*
- **Decalaj la sfârșit:** *Cablul nu prezintă "atingere" (îmbinare) complet la îmbinare sau conexiune.*
- **Terminarea cablului:** *Capătul final al cablului nu este bine lustruit sau prezintă mizerie.*

Un test rapid și ușor poate fi făcut prin luminarea cu o lanternă la un capăt al fibrei în timp ce observăm celălalt capăt al fibrei. Dacă lumina este vizibilă, fibra este capabilă de transfer al luminii. Deși acest lucru nu asigură performanța fibrei, este un mod rapid și ieftin de depistare a fibrei stricate.

Este recomandat ca un tester optic, cum ar fi cel din Fig. xxx, să fie utilizat să testeze cablurile de fibră optică. Un Optical Time Domain Reflectometer (OTDR) reprezintă un instrument pentru a testa fiecare segment din fibră optică. Dispozitivul injectează un puls de testare de lumină în cablu și măsoară "back scatter" și reflexia luminii detectată ca o funcție de timp. OTDR va calcula distanță aproximativă la care aceste defecte sunt detectate de-a lungul lungimii cablului.

Fig. 4.34. Optical Time Domain Reflectometer



Există multe avantaje în utilizarea cablului de fibră optică în comparație cu cele de cupru.

Având în vedere că fibrele utilizate în mediu de fibră optică nu sunt conductoare electrice, mediul este imun la interferența electromagnetică și nu va conduce curenți electrici nedoriți datorită problemelor de împământare. Deoarece fibrele optice sunt subțiri și au pierdere de semnal relativ redusă, pot funcționa pe lungimi mult mai mari decât mediul de cupru, fără necesitatea regenerării de semnal. Unele specificații de nivel fizic pentru fibra optică permit ca lungimile să ajungă să atingă mai mulți kilometri.

Probleme de implementare a mediului de fibră optică:

- *Mai costisitor (de obicei) decât mediul de cupru pentru aceeași distanță (dar pentru o capacitate mai mare)*
- *Echipamente și componente diferite necesare pentru terminarea și îmbinarea infrastructurii de cablu.*
- *Manipularea mai atentă decât în cazul mediului de cupru.*

În prezent, în multe medii enterprise, fibra optică este utilizată în cablarea backbone pentru conexiuni punct-la-punct cu trafic ridicat între facilități de distribuție de date și pentru interconectarea clădirilor din campusurile cu mai multe clădiri. Deoarece fibra optică nu conduce electricitate și are pierdere de semnal scăzută, este potrivită pentru aceste utilizări.

Fig. 4.35 pune în lumină unele dintre aceste diferențe.

Implementation Issues	UTP Cabling	Fiber-optic Cabling
Bandwidth supported	10 Mb/s – 10 Gb/s	10 Mb/s – 100 Gb/s
Distance	Relatively short (1 – 100 meters)	Relatively high (1 – 100,000 meters)
Immunity to EMI and RFI	Low	High (Completely immune)
Immunity to electrical hazards	Low	High (Completely immune)
Media and connector costs	Lowest	Highest
Installation skills required	Lowest	Highest
Safety precautions	Lowest	Highest

Fig. 4.35. Parametrii Tehnici pentru Cabluri

4.4.1.3 Mediul de comunicație Wireless

Mediul wireless conduce semnale electromagnetice ce reprezintă digiții binari ai comunicațiilor de date prin utilizarea frecvențelor radio sau microunde.

Ca mediu de rețea, mediul wireless nu este restricționat la conductori sau căi de conducție, așa cum sunt mediile de cupru și fibră optică. Mediul wireless oferă cele mai bune opțiuni de mobilitate dintre toate mediile. Prin urmare, numărul de dispozitive ce permit wireless este în continuă creștere. Din aceste motive, wireless a devenit alegerea pentru rețelele de domiciliu. Deoarece opțiunile de lățime de bandă cresc, wireless câștigă popularitate rapidă în rețelele de întreprinderi.

Fig. 4.36 evidențiază simboluri variate legate de wireless.

Wireless are unele domenii de interes, precum:

- **Aria de acoperire:** Tehnologiile wireless de comunicații de date lucrează bine în medii deschise. Cu toate acestea, anumite materiale de construcții utilizate în clădiri sau structuri, precum și terenul local, vor limita aria de acoperire.
- **Interferență:** Wireless este sensibil la interferență și poate fi perturbat de către dispozitive comune precum telefoanele fără fir de uz casnic, unele tipuri de lumini fluorescente, cupoare cu microunde și alte comunicații wireless.
- **Securitatea:** Acoperirea comunicațiilor wireless nu necesită accesarea unui cablu fizic din mediu. Prin urmare, dispozitivele și utilizatorii ce nu au acces autorizat la rețea pot câștiga acces la transmisie. În consecință, securitatea rețelei este o componentă importantă a administrării de rețea wireless.

Deși tehnologia wireless crește în popularitate pentru conectivitate, cuprul și fibra sunt cele mai populare medii de la nivelul fizic pentru implementările de rețea.



Fig. 4.37. Simboluri Wi-Fi

IEEE și standardele din industria de comunicații pentru comunicații de date wireless acoperă nivelele fizic și legătură de date.

Standarde comune de comunicații de date ce se aplică la mediul wireless sunt:

- **Standard IEEE 802.11:** Tehnologia Wireless LAN (WLAN), referită ca Wi-Fi, utilizează un sistem non-deterministic ca proces de acces la mediu Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA).
- **Standard IEEE 802.15:** Standardul Wireless Personal Area Network (WPAN), cunoscut ca "Bluetooth", utilizează un proces de dispozitiv pentru a comunica peste distanțe de la 1m la 100m.
- **Standard IEEE 802.16:** Cunoscut ca Worldwide Interoperability for Microwave Access (WiMAX), utilizează o topologie point-to-multipoint pentru a oferi acces wireless de bandă largă.

Fig. xxx ilustrează unele diferențe dintre mediile wireless.

Notă: Alte tehnologii wireless precum comunicațiile prin satelit sau celular pot de asemenea oferi conectivitate de rețea. Oricum, aceste tehnologii wireless nu intră în scopul acestui capitol.

În fiecare dintre exemplele de mai sus, specificații de nivel fizic sunt aplicate la domenii ce includ:

- Datele de codare a semnalului radio.
- Frecvența și puterea transmisiei.
- Recepția semnalului și cerințele de codificare.
- Designul și construcția antenei.

Notă: Wi-Fi este un simbol comercial al Wi-Fi Alliance. Wi-Fi este utilizat cu produse certificate ce aparțin dispozitivelor WLAN care se bazează pe standardele IEEE 802.11.

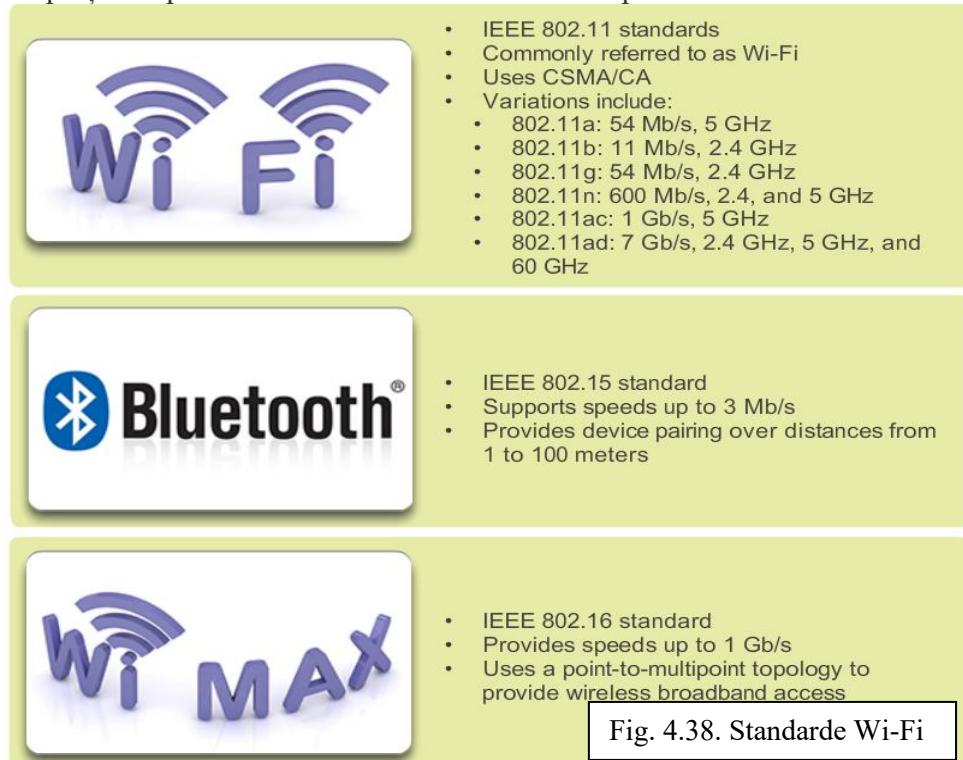


Fig. 4.38. Standarde Wi-Fi

O implementare de date wireless comună este permiterea dispozitivelor să se conecteze fără fir într-un LAN. În general, un LAN Wireless necesită următoarele dispozitive de rețea:

- **Wireless Access Point (AP):** Strâng semnalele wireless de la utilizator și se conectează, de obicei printr-un cablu de curpru, la infrastructura de rețea bazată pe

cupru existentă, cum ar fi Ethernet. Routerele wireless de domiciliu sau din întreprinderile mici integrează funcțiile unui router, switch și punct de acces într-un singur dispozitiv, la fel cum este ilustrat în Fig. xxx.

- **Wireless NIC adapters:** Oferă capacitate de comunicare wireless la fiecare host din rețea.

Pe măsură ce tehnologia s-a dezvoltat, un număr de standarde WLAN bazate pe Ethernet au apărut. Trebuie să fim atenți la achiziționarea dispozitivelor wireless pentru a asigura compatibilitate și interoperabilitate.

Beneficiile tehnologiilor wireless de comunicații de date sunt evidente, în special în ceea ce privește spațiile de cablare costisitoare și confortul mobilității hostului. Însă, administratorii de rețea trebuie să dezvolte și să aplique politici și procese stricte de securitate pentru a proteja LANurile wireless împotriva accesului neautorizat și împotriva daunelor.



Fig. 4.39. Cisco Linksys EA6500 802.11ac Wireless Router

Diferite standarde 802.11 au evoluat de-a lungul anilor. Acestea includ:

- **IEEE 802.11a:** Operează la frecvența de 5GHz și oferă viteze mai mari de 54Mb/s. Deoarece acest standard operează la frecvențe mari, are o arie de acoperire mai mică și este mai puțin eficient în penetrarea structurilor din clădiri. Dispozitivele ce operează sub acest standard nu sunt interoperabile cu standardele 802.11b și 802.11g descrise mai jos.
- **IEEE 802.11b:** Operează la frecvența de 2.4GHz și oferă viteze mai mari de 11Mb/s. Dispozitivele ce implementează acest standard au o arie de acoperire mai mare și penetreză mai bine structurile clădirilor decât dispozitivele bazate pe 802.11a.
- **IEEE 802.11g:** Operează la frecvența de 2.4GHz și oferă viteze mai mari de 54Mb/s. Dispozitivele ce implementează acest standard operează la aceeași frecvență radio și au aceeași arie de acoperire ca 802.11b, dar aceeași lățime de bandă ca 802.11a.
- **IEEE 802.11n:** Operează la frecvența de 2.4GHz sau 5GHz. Vitezele de date așteptate sunt de la 100Mb/s la 600Mb/s și au o distanță de acoperire de până la 70m. Este compatibil cu dispozitivele 802.11a/b/g.
- **IEEE 802.11ac:** Poate opera simultan cu frecvențele de 2.4GHz și 5.5GHz, oferind viteze de până la 450Mb/s și 1.3Gb/s (1300Mb/s). Este compatibil cu dispozitivele 802.11a/b/g/n.
- **IEEE 802.11ad:** Cunoscut și sub numele de "WiGig". Utilizează o soluție tri-band Wi-Fi folosind 2.4 GHz, 5 GHz, și 60 GHz și oferă teoretic viteze de până la 7Gb/s.

Fig. 4.40 pune în evidență unele dintre aceste diferențe.

Standard	Maximum Speed	Frequency	Backward Compatible
802.11a	54 Mb/s	5 GHz	No
802.11b	11 Mb/s	2.4 GHz	No
802.11g	54 Mb/s	2.4 GHz	802.11b
802.11n	600 Mb/s	2.4 GHz or 5 GHz	802.11a/b/g
802.11ac	1.3 Gb/s (1300 Mb/s)	2.4 GHz and 5 GHz	802.11a/b/g/n
802.11ad	7 Gb/s (7000 Mb/s)	2.4 GHz, 5 GHz, and 60 GHz	802.11a/b/g/n/ac

Fig. 4.40. Diferențe între standarde Wi-Fi

Atunci când se lucrează în Packet Tracer (mediu de laborator sau o setare corporativă), trebuie bine știut cum se selectează cablul adecvat și modul în care se conectează dispozitivele.

4.5 Protocolele de la Nivelul Legătura de Date

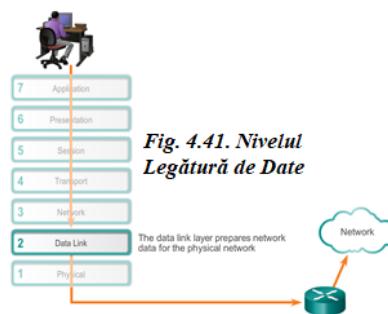
4.5.1 Scopul nivelului legătură de date

Nivelul de acces la rețea din stiva TCP/IP are echivalente în stiva OSI nivelele :

- *Legătura de date (Nivelul 2).*
- *Fizic (Nivelul 1).*

Așa cum este prezentat în Fig. 4.41, nivelul legătură de date este responsabil de schimbul de cadre dintre nodurile dintr-un mediu de rețea fizic. Permite nivelelor superioare să acceseze mediu și să controleze modul în care datele sunt plasate și recepționate în mediu.

Notă: Notația de nivel 2 pentru dispozitivele de rețea conectate într-un mediu comun este “un nod”.



Nivelul legătură de date îndeplinește aceste două servicii de bază:

- *Acceptă pachetele de la nivelul 3 și le împachetează în unități de date numite cadre.*
- *Controlează controlul accesului la mediu și efectuează detecția de erori.*

Nivelul legătură de date separă în mod eficient tranzitiiile ce au loc în timp ce pachetul este transferat de la procesele de comunicație de la nivelele superioare. Nivelul legătură de date primește pachetele și le direcționează de la un protocol de nivel superior, în acest caz IPv4 sau IPv6. Acest protocol de nivel superior nu trebuie să știe ce mediu de comunicație va fi utilizat.

Notă: În acest capitol, mediul și mijlocul nu se referă la conținutul digital și la multimedia, cum ar fi audio, animație, televiziune și video. Mediul se referă la materialul ce transferă semnalele de date, cum ar fi cablul de cupru sau fibra optică.

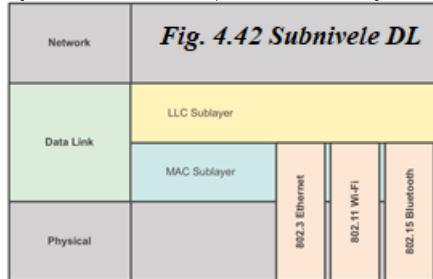
Nivelul legătură de date este divizat în două subnivele:

- **Logical Link Control (LLC):** *Acest nivel superior definește procesul software ce furnizează servicii la protocolele de la nivelul rețea. Adaugă informații în cadru ce identifică ce protocol de la nivelul rețea este utilizat pentru cadru. Aceste informații permit ca protocole multiple de nivel 3, cum ar fi IPv4 și IPv6, să utilizeze aceeași interfață de rețea și mediu.*
- **Media Access Control (MAC):** *Acest nivel inferior definește procesele de acces la mediu efectuate de către hardware. Oferă adresare și delimitare de date la nivelul legătură de date în concordanță cu cerințele de semnal fizic ale mediului și tipului de protocol de la nivel de legătură de date utilizat.*

Separarea nivelului legătură de date în subnivele permite ca un tip de cadru definit de către nivelul superior să acceseze tipuri diferite de medii definite de către nivelul inferior. Acest lucru se întâmplă în multe tehnologii LAN, inclusiv Ethernet.

Fig. xxx ilustrează modul în care nivelul legătură de date este împărțit în subnivelele LLC și MAC. LLC comunica cu nivelul rețea în timp ce nivelul MAC permite tehnologii diferite de acces la rețea. De exemplu, nivelul MAC comunica cu tehnologia Ethernet LAN pentru a

transmite și primi cadre peste cabluri de cupru sau de fibră optică. Subnivelul MAC comunică de asemenea cu tehnologii wireless precum Wi-Fi și Bluetooth pentru a transmite și primii cadre.



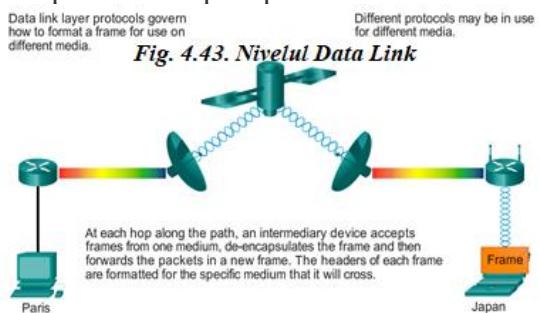
Protocolele de nivel 2 specifică încapsularea unui pachet într-un cadru și tehniciile pentru punerea sau luarea pachetului încapsulat de pe fiecare mediu. Această tehnică se numește metoda de control al accesului la mediu.

În călătoria pachetelor de la hostul sursă la hostul destinație, ele traversează în mod normal diferite rețele fizice. Aceste rețele fizice pot consta din diferite tipuri de medii fizice precum cabluri de cupru, fibre optice și wireless ce constă din semnale electromagnetice, frecvențe radio și microunde și legături prin satelit.

Pachetele nu au o modalitate pentru a accesa direct aceste medii diferite. Este rolul nivelului legătură de date din stiva OSI să pregătească pachetele de la nivelul rețea pentru transmisie și să controleze accesul la mediul fizic. Metodele de control al accesului la mediu descrise de către protocolele de la nivelul legătură de date definesc procesele prin care dispozitivele de rețea pot accesa mediul și transmit cadre în diverse medii de rețea.

Fără nivelul legătură de date protocolele de nivel rețea precum IP ar trebui să ia măsuri pentru conectarea la orice tip de mediu ce poate exista în drumul de livrare. Mai mult, IP ar trebui să se adapteze de fiecare dată când o nouă tehnologie de rețea sau mediu este dezvoltat. Acest proces ar împiedica inovarea și dezvoltarea de protocol sau mediu de rețea. Acesta este un motiv cheie de utilizare a unei abordări pe nivele în rețea.

Deși cele două hosturi comunică utilizând IP în mod exclusiv, este necesar ca numeroase protocole de la nivelul legătură de date să fie utilizate pentru a transporta pachetele IP peste diferite tipuri de LANuri și WANuri. Fiecare tranzitie de la un router poate necesita un protocol de nivel legătură de date diferit pentru transport pe un mediu nou.



Metode diferite de control de acces la mediu pot fi necesare în timpul unei singure comunicații. Fiecare mediu de rețea pe care pachetele îl traversează de la un host local la un host de la distanță poate avea caracteristici diferite. De exemplu, un Ethernet LAN constă din mai multe hosturi ce au acces la mediul de rețea pe o bază ad hoc. Legăturile seriale constau dintr-o conexiune directă între două dispozitive, peste care datele "curg" secvențial ca biți într-un mod ordonat.

Interfețele routerului încapsulează pachetul în frameul adecvat și este utilizată o metodă de control de acces la mediu potrivită pentru accesarea fiecărei legături. În orice schimb de

pachete de nivel rețea, pot există numeroase nivele de legătură de date și tranziții de mediu. La fiecare hop de-a lungul căii, un router realizează următoarele operații :

- *Accepta un frame dintr-un mediu.*
- *Decapsulează frameul.*
- *Reîncapsulează pachetul într-un nou frame.*
- *Transmite noul frame adecvat la mediul segmentului de rețea fizică respectiv.*

Routerul din Fig. xxx are o interfață Ethernet pentru a se conecta la LAN și o interfață serială pentru a se conecta la WAN. În procesarea frameului de către router, acesta va utiliza serviciile de la nivelul legătură de date pentru a primi frameul de la un mediu, îl decapsulează în PDU de Nivel 3, îl reîncapsulează într-un nou frame și îl plasează pe mediul noii legături din rețea.



4.5.2 Structura frameului de nivel 2

Nivelul legătură de date pregătește un pachet pentru transportul prin mediul de la sursă la destinație, prin încapsularea să cu un header și un trailer pentru a crea un frame. Descrierea unui frame este un element cheie al oricărui protocol de la nivelul legătură de date.

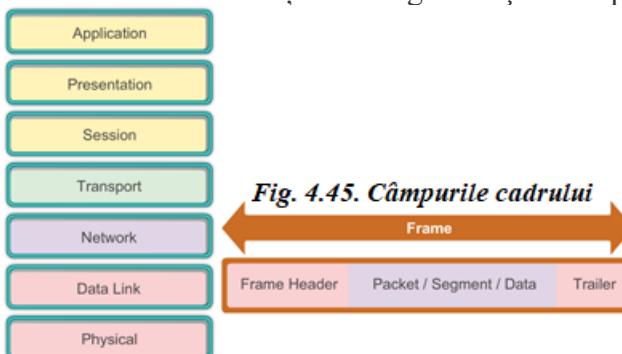
Protocolele de la nivelul legătură de date necesită informații de control pentru a permite ca protocolele să funcționeze. Informațiile de control, în mod normal, răspund la:

- *Ce noduri comunică între ele ?*
- *Când comunicarea dintre noduri individuale începe și când se termină ?*
- *Ce erori au loc în timpul în care nodurile comunică ?*
- *Ce noduri vor comunica apoi ?*

Spre deosebire de alte PDUuri prezentate anterior în acest curs, cadrul de la nivelul legătură de date include:

- **Header:** Conține informații de control, cum ar fi adresarea, și este localizat la începutul PDU-ului.
- **Data:** Conține headerul IP, headerul de la nivelul transport și application data.
- **Trailer:** Conține informații de control pentru detecția erorilor și este adăugat la sfârșitul PDU-ului.

Aceste elemente ale frameului sunt evidențiate în Fig. 4.45 și vor fi prezentate în detaliu.



Atunci când datele traversează mediul, sunt convertite în fluxuri de biți, sau în 1 și 0. Dacă un nod primește fluxuri mari de biți, cum va determina unde un frame începe și unde se termină sau ce biți reprezintă adresa?

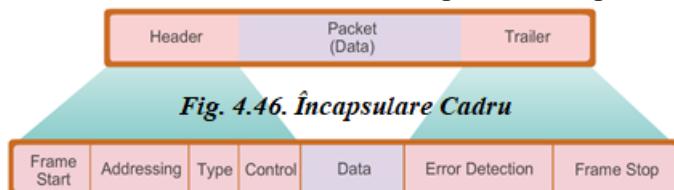
Încadrarea împarte fluxul în grupări descifrabile, cu informații de control inserate în header și trailer ca valori în câmpuri diferite. Acest format oferă semnalelor fizice o structură ce poate fi primită de către noduri și decodată în pachete la destinație.

La fel ca în Fig. xxx, tipurile generale de câmpurile din frame sunt:

- **Frame start and stop indicator flags:** Utilizat de către subnivelul MAC pentru a identifica începutul și sfârșitul unui frame.
- **Adresarea:** Utilizată de către subnivelul MAC pentru a identifica nodurile sursă și destinație.
- **Tipul:** Utilizat de către LLC pentru a identifica protocolul de Nivel 3.
- **Control:** Identifică servicii speciale de control al fluxului.
- **Data:** Conține frame payload (headerul pachetului, headerul segmentului și data).
- **Detectia erorilor:** Incluse după data pentru a forma trailerul, aceste câmpuri de frame sunt utilizate pentru detectia erorilor.

Nu toate protocolele includ toate aceste câmpuri. Standardele pentru un protocol specific de legătură de date definesc formatul real al frameului.

Notă: Exemple de formate ale frameurilor vor fi prezentate spre sfârșitul acestui capitol.



4.5.3 Standarde de Nivel 2

Spre deosebire de protocolele nivelelor superioare din suita TCP/IP, protocolele de la nivelul legătură de date nu sunt definite în general de către Request for Comments (RFCs). Deși Internet Engineering Task Force (IETF) gestionează protocolele și serviciile funcționale pentru suita de protocole TCP/IP în nivelele superioare, IETF nu definește funcțiile și funcționarea nivelului de acces la rețea din modelul TCP/IP.

Mai exact, serviciile și specificațiile de la nivelul legătură de date sunt definite de către mai multe standarde ce se bazează pe o varietate de tehnologii și medii la care se aplică protocolele. Unele dintre aceste standarde integrează atât serviciile de nivel 2, cât și pe cele de nivel 1.

Protocolele și serviciile funcționale de la nivelul legătură de date sunt descrise de către:

- Organizații de inginerie ce stabilesc standarde și protocole publice și deschise.
- Companii de comunicații ce stabilesc și utilizează protocole proprietare pentru a profita de noi progrese în tehnologie sau de oportunități de pe piață.

Organizații de inginerie ce definesc protocole și standarde deschise ce se aplică la nivelul legătură de date:

- Institute of Electrical and Electronics Engineers (IEEE).
- International Telecommunication Union (ITU).
- International Organization for Standardization (ISO).
- American National Standards Institute (ANSI).

Fig. 4.47 evidențiază diverse organizații de standarde și unele dintre cele mai importante protocole de nivel legătură de date ale lor.

Standard Organization	Networking Standards
IEEE Fig. 4.47. Organizații de emisie a Standardelor	<ul style="list-style-type: none"> 802.2: Logical Link Control (LLC) 802.3: Ethernet 802.4: Token bus 802.5: Token ring 802.11: Wireless LAN (WLAN) & Mesh (Wi-Fi certification) 802.15: Bluetooth 802.16: WiMax
ITU-T	<ul style="list-style-type: none"> G.992: ADSL G.8100 - G.8199: MPLS over Transport aspects Q.921: ISDN Q.922: Frame Relay
ISO	<ul style="list-style-type: none"> HDLC (High Level Data Link Control) ISO 9314: FDDI Media Access Control (MAC)
ANSI	<ul style="list-style-type: none"> X3T9.5 and X3T12: Fiber Distributed Data Interface (FDDI)

4.5.4 Controlul Accessului la Mediu - Media Access Control

4.5.4.1 Topologii

Reglementarea plasării frameurilor de date pe mediu este controlată de către subnivelul de control al accesului la mediu.

Controlul accesului la mediu este echivalent cu regulile de trafic ce guvernează intrarea autoturismelor pe o șosea. Absența unui control de acces la mediu ar fi echivalent cu ignorarea de către vechicule a traficului și intrarea lor pe șosea fără a ține cont de alte vechicule. Însă, nu toate șoselele și intrările sunt la fel. Traficul poate intra pe o șosea prin fuziune, prin aşteptare a rândului la un semn de stop sau prin vizualizarea luminii de semnalizare. Un conducător auto urmează diferite seturi de reguli în funcție de fiecare tip de intrare.

În același mod, există diferite moduri de gestionare a plasării frameurilor pe mediu. Protocolele de la nivelul legătură de date definesc regulile de acces la medii diferite. Unele metode de control al accesului la rețea utilizează procese controlate cu strictețe pentru a asigura faptul că frameurile sunt plasate în siguranță pe mediu. Aceste metode sunt definite de către protocole sofisticate, ce necesită mecanisme ce introduc overhead în rețea.

Împreună cu diferite implementări ale protocolelor de la nivelul legătură de date, există metode diferite de control al accesului la mediu. Aceste tehnici de control al accesului la mediu definesc dacă și când nodurile “împart” mediul.

Metoda reală de control al accesului la mediu utilizată depinde de:

- Topologie:** Modul în care conexiunile dintre noduri apar la nivelul legătură de date.
- Partajarea mediului:** Modul în care nodurile “împart” mediul. Partajarea mediului poate fi punct-la-punct, ca în conexiunile WAN, sau partajata ca în rețelele LAN.



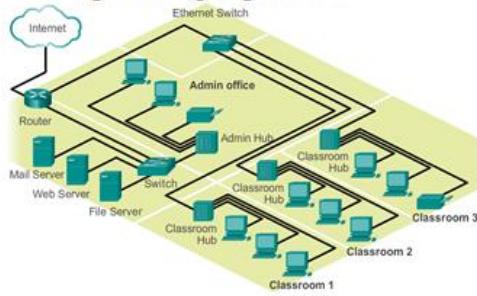
Topologia unei rețele reprezintă aranjarea sau relația dispozitivelor de rețea și interconexiunile dintre ele. Topologiile LAN și WAN pot fi vizualizate în două moduri:

- Topologie fizică:** Se referă la conexiunile fizice și identifică modul în care dispozitivele finale și dispozitivele de infrastructură, cum ar fi routere, switchuri și puncte de acces wireless, sunt interconectate. Topologiile fizice sunt de obicei punct-la-punct sau stea. De văzut Fig. 4.49.

- **Topologie logică:** Se referă la modul în care o rețea transferă frameurile de la un nod la următorul nod. Această aranjare constă în conexiuni virtuale între nodurile dintr-o rețea. Aceste căi de semnal logice sunt definite de către protocoalele de la nivelul legătură de date. Topologia logică a legăturilor point-to-point este relativ simplă în timp ce mediul partajat oferă metode de control al accesului la mediu deterministică și nedeterministică. A se vedea Fig. 4.50.

Nivelul legătură de date “vede” topologia logică a unei rețele atunci când controlează accesul datelor în mediu. Topologia logică influențează tipul de network framing și control al accesului la mediu utilizat.

Fig. 4.49. Topologie Fizică



Mail server 192.168.2.1
Web server 192.168.2.2
File server 192.168.2.3

Ethernet 192.168.2.0

192.168.2.4
192.168.2.5
192.168.2.6

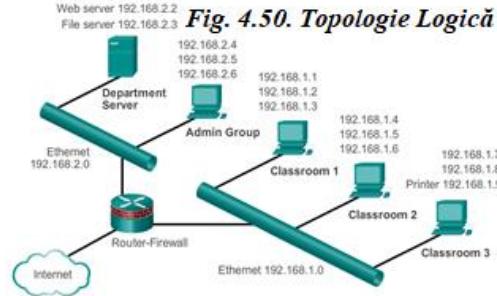
192.168.1.1
192.168.1.2
192.168.1.3

192.168.1.4
192.168.1.5
192.168.1.6

192.168.1.7
192.168.1.8
192.168.1.9

Printer 192.168.1.0

Fig. 4.50. Topologie Logică



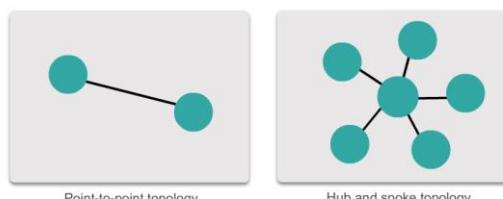
4.5.4.2 Topologii WAN

WANurile sunt de obicei interconectate utilizând următoarele topologii fizice:

- **Point-to-Point:** Aceasta este cea mai simplă topologie ce constă dintr-o legătură permanentă între două puncte. Din acest motiv, aceasta este o topologie WAN foarte populară.
- **Hub and Spoke:** O versiune WAN a topologiei stea în care un punct central interconectează locurile marginale prin legături point-to-point.
- **Mesh:** Această topologie oferă disponibilitate ridicată, dar necesită ca fiecare sistem să fie interconectat cu toate celelalte sisteme. Prin urmare, costurile fizice și administrative pot fi semnificative. Fiecare legătură este de fapt o legătură point-to-point cu celălalt nod. Variații ale acestei topologii includ un mesh parțial în care unele, însă nu toate, dispozitive sunt interconectate.

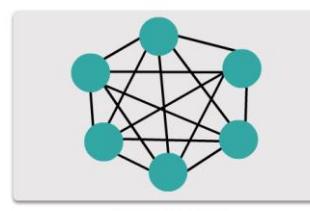
Cele trei tipuri de topologii fizice WAN comune sunt ilustrate în Fig. xxx.

Topologiile fizice point-to-point conectează direct două noduri, aşa cum este ilustrat în Fig. xxx.



Point-to-point topology

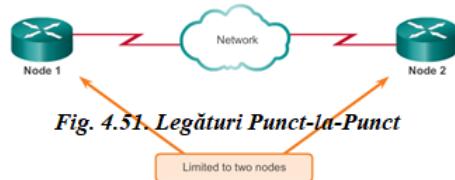
Hub and spoke topology



Full mesh topology

În acest aranjament, două noduri nu trebuie să împartă mediul cu alte hosturi. În plus, un nod nu trebuie să facă nici-o determinare cu privire dacă un frame este destinat pentru el sau pentru un alt nod. Prin urmare, protocolele de legătură de date logice pot fi foarte simple, iar toate frameurile din mediu pot călători la sau de la cele două noduri. Frameurile sunt plasate pe mediu de comunicație de către un nod de la un capăt și sunt scoase din mediu de către nodul de la celălat capăt al circuitului point-to-point.

Protocolele de la nivelul legătură de date oferă procese de control al accesului la mediu mai sofisticate pentru topologii point-to-point, dar acest lucru nu ar mai adăuga **"overhead"** de protocol ce nu este necesar.



Nodurile terminale ce comunică în rețea point-to-point pot fi conectate fizic printr-un număr de dispozitive intermediare. Oricum, utilizarea dispozitivelor fizice în rețea nu afectează topologia logică.

Ca și în Fig. 4.51, nodurile sursă și destinație pot fi conectate indirect unul cu celălalt, peste o distanță geografică. În unele cazuri, conexiunea logică dintre noduri formează ceea ce se numește un circuit virtual. Un circuit virtual este o conexiune logică creată într-o rețea între două dispozitive de rețea. Două noduri de la orice capăt al circuitului virtual fac schimb de frameuri între ele. Acest lucru are loc chiar dacă frameurile sunt direcționate prin intermediul dispozitivelor intermediare. Circuitile virtuale sunt construcții importante de comunicări logice utilizate de către tehnologiile de nivel 2.

Metoda de acces la mediu utilizată de către protocolul de legătură de date este determinată de către topologia point-to-point și nu de cea fizică. Acest lucru înseamnă că acea conexiune point-to-point logică dintre două noduri nu trebuie să fie neapărat între nodurile fizice la fiecare capăt al unei singure legături fizice.

Fig. 4.52 ilustrează conexiunile fizice dintre două routere.

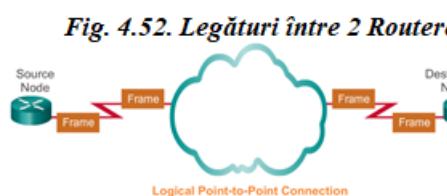


Fig. 4.53. Conexiuni fizice Punct-la-Punct

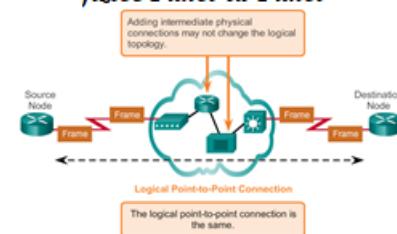


Fig. 4.53 ilustrează o topologie point-to-point. În rețelele point-to-point Fig. 4.54., datele pot "curge" în unul dintre cele două moduri:

- **Comunicație half-duplex**: Ambele dispozitive pot transmite și primi informații prin mediul de comunicație, dar nu o pot face în mod simultan. Ethernet a stabilit reguli arbitrare pentru rezolvarea conflictelor generate de situații în care mai mult de o stație încearcă să transmită în același timp. Fig. 4.55 ilustrează comunicația half-duplex.
- **Comunicația full-duplex**: Ambele dispozitive pot transmite și primi informații prin mediul de comunicație în același timp. Nivelul legătură de date presupune că mediul este disponibil pentru transmisie la ambele noduri, în orice moment. Prin urmare, nu este necesară existența unei reguli arbitrare la nivelul legătură de date. Fig. 4.56. ilustrează comunicarea full-duplex.

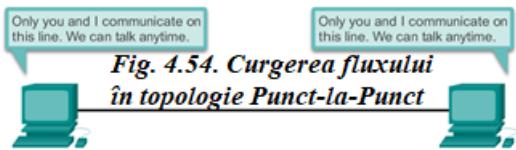
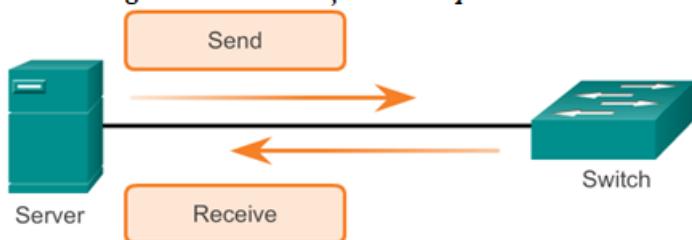


Fig. 4.55. Comunicație Half-Duplex



Fig. 4.56. Comunicație Full-Duplex



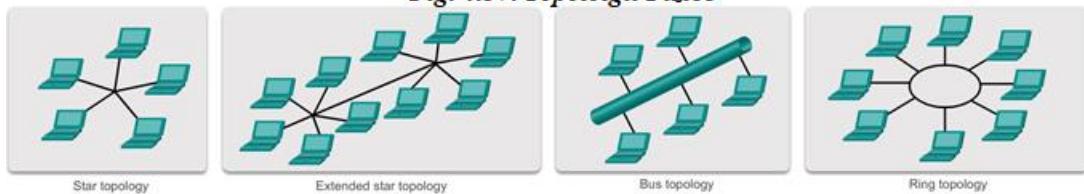
4.5.4.4 Topologii LAN

Topologia fizică definește modul în care sistemele sunt interconectate fizic. În LANurile cu mediu partajat, dispozitivele finale pot fi interconectate prin utilizarea următoarelor topologii fizice:

- **Steauă:** Dispozitivele finale sunt conectate la un dispozitiv intermediar central. Topologiile steauă inițiale interconectau dispozitivele finale prin intermediul huburilor. Însă, topologiile steauă actuale utilizează switchuri. Topologia steauă este cea mai comună topologie LAN fizică deoarece este ușor de instalat, foarte scalabilă (ușor de inserat) și ușor de depanat.
- **Steauă extinsă sau hibridă:** Aceasta este o combinație de alte topologii cum ar fi rețele star interconectate între ele printr-o topologie bus.
- **Bus:** Toate sistemele finale sunt legate între ele și terminate într-o anumită formă la fiecare capăt. Dispozitivele de infrastructură precum switchurile nu sunt necesare pentru interconectarea dispozitivelor finale. Topologiile bus erau utilizate în rețelele Ethernet vechi deoarece erau ieftin de utilizat și ușor de instalat.
- **Ring:** Sistemele finale sunt conectate la vecinul respectiv lor, formând un inel. Spre deosebire de topologia bus, inelul nu trebuie să fie finalizat. Topologiile inel erau utilizate în rețelele vechi Fiber Distributed Data Interface (FDDI). Mai exact, rețelele (FDDI) folosesc un al doilea inel pentru imbunătățiri de performanță sau toleranță la defecțiuni.

Fig. 4.57 ilustrează modul în care dispozitivele finale sunt interconectate în LANuri.

Fig. 4.57. Topologii Fizice



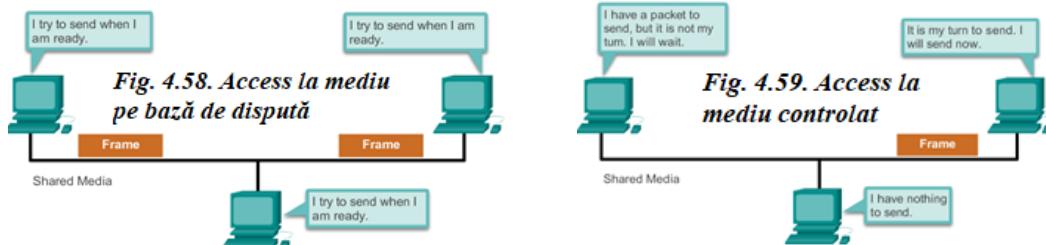
Topologia logică a unei rețele este legată strâns de mecanismul utilizat pentru gestionarea accesului la rețea. Metodele de acces oferă procedurile de gestionare a accesului la rețea astfel încât toate stațiile să aibă acces. Atunci când mai multe entități împart același mediu, unele mecanisme trebuie să fie introduse pentru a controla accesul. Aceste metode sunt aplicate în rețele pentru a stabili accesul la mediu.

Unele topologii de rețea împart un mediu comun cu mai multe noduri. În fiecare moment, trebuie să fie un număr de dispozitive în aşteptare să trimită și să primească date utilizând mediul de rețea. Există reguli ce guvernează modul în care aceste dispozitive împart mediul.

Există două metode de control al accesului la rețea pentru mediul partajat:

- **Acces pe bază de “dispută”:** *Toate nodurile “concurează” pentru utilizarea mediului, dar au un plan în cazul coliziunilor. Fig. 4.58 ilustrează accesul pe bază de “dispută”.*
- **Acces controlat:** *Fiecare nod are propriul timp în care să utilizeze mediul. Fig. 4.59 ilustrează accesul controlat.*

Protocolul de la nivelul legătură de date specifică metoda de control al accesului la mediu ce va oferi balanță adecvată dintre controlul frameului, protecția frameului și network overhead.



Atunci când se utilizează o metodă non-deterministică bazată pe dispută, un dispozitiv de rețea poate încerca să acceseze mediul oricând are date de transmis. Pentru a preveni haosul complet în mediu, această metodă utilizează un proces Carrier Sense Multiple Access (CSMA) pentru a detecta mai întâi dacă pe mediu este un semnal.

Dacă este detectat un semnal de la un alt nod pe mediu, înseamnă că un alt dispozitiv transmite. Dacă dispozitivul ce încearcă să transmită “vede” că mediul este ocupat, va aștepta și va încerca iar, după o scurtă perioadă de timp. Dacă nu este detectat nici-un semnal, dispozitivul transmite datele. Rețelele wireless și Ethernet utilizează control al accesului la mediu bazat pe “dispută”.

Este posibil ca procesul CSMA să eșueze și două dispozitive să transmită în același timp, creându-se o coliziune. Dacă acest lucru are loc, datele transmise de către ambele dispozitive vor fi corupte și necesită să fie retransmise.

Metodele de control al accesului la mediu bazate pe “dispută” nu au overhead al metodelor de acces controlat. Un mecanism de detectare al cui îi este rândul să acceseze mediul nu este necesar. Însă, sistemele bazate pe “dispută” nu sunt potrivite în condițiile unei utilizări intense ale mediului. Cu cât crește numărul de noduri, probabilitatea unui acces la mediu cu succes fără coliziune scade. În plus, mecanismele de recuperare necesare pentru corectarea erorilor în timpul acestor coliziuni diminuează și mai mult throughputul.

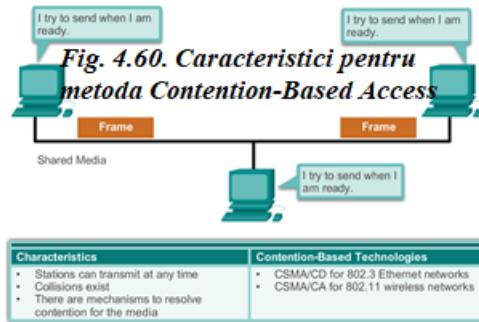
CSMA este de obicei implementat împreună cu o metodă de rezolvare a “disputei” din mediu. Cele două metode comune utilizate sunt:

- **Carrier Sense Multiple Access with Collision Detection (CSMA/CD):** *Dispozitivul final monitorizează mediul pentru prezența unui semnal de date. Dacă un semnal de date este absent și prin urmare mediul este liber, dispozitivul transmite datele. Dacă semnalele sunt apoi detectate, arată faptul că un alt dispozitiv trimite în același timp, toate dispozitivele încearcă să transmită și încearcă mai târziu. Formele tradiționale de Ethernet utilizează această metodă.*

- **Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA):** Dispozitivul final examinează mediul de prezență unui semnal de date. Dacă mediul este liber, dispozitivul transmite o notificare în mediul cu intenția să de utilizare a mediului. De îndată ce primește o permisiune de transmisie, dispozitivul transmite datele. Această metodă este utilizată de către tehnologiile de rețea wireless 802.11.

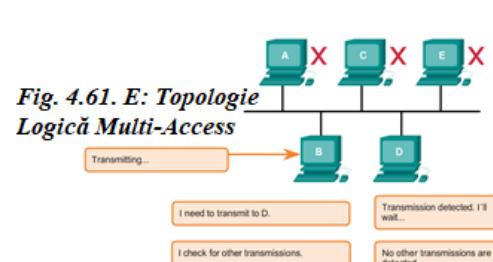
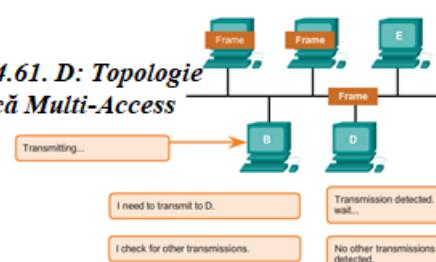
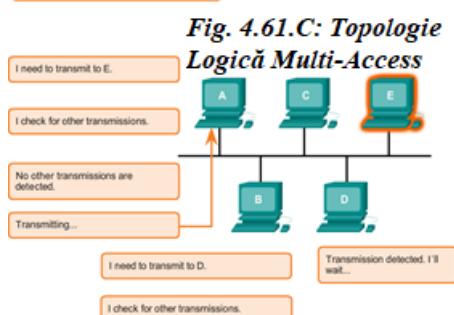
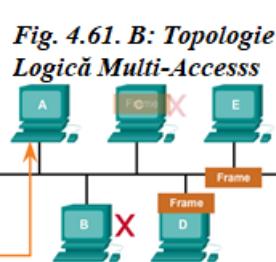
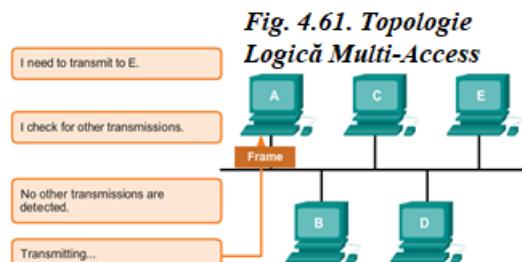
Fig. 4.60 ilustrează următoarele:

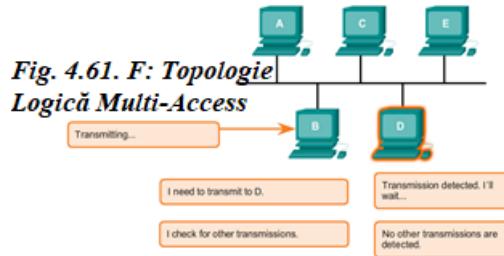
- Modul în care metodele de acces bazate pe “dispută” funcționează.
- Caracteristici ale metodelor de acces bazate pe “dispută”.
- Exemple de metode de acces bazate pe “dispută”.



O topologie multi-acces logică permite unui număr de noduri să comunice utilizând aceleași medii partajate. Datele de la un singur nod pot fi plasate pe mediul la un moment dat. Fiecare nod vede toate frameurile care sunt pe mediul, dar numai nodul la care frameul este adresat poate procesa conținutul frameului.

Având multe noduri care partajează accesul la mediul la nivel legătură de date necesită o metodă de control al accesului la mediul pentru a reglementa transmisia de date și, prin urmare, reduce coliziunile între semnale diferite.





Atunci când se utilizează metoda de acces controlat, dispozitivele de rețea își așteaptă timpul, în secvență, să acceseze mediul. Dacă un dispozitiv nu necesită să acceseze mediul, oportunitatea trece la următorul dispozitiv. Procesul acesta este facilitat prin utilizarea unui token. Un dispozitiv final achiziționează tokenul și pasează un frame pe mediul și nici-un alt dispozitiv nu poate să facă același lucru până când frameul nu a ajuns și nu a fost procesat la destinație, eliberând tokenul.

Notă: Această metodă este cunoscută și ca acces programat sau deterministic.

Deși accesul controlat este bine ordonat și oferă throughput previzibil, metodele deterministicice pot fi ineficiente, deoarece un dispozitiv trebuie să-și aștepte rândul până când poate utiliza mediul.

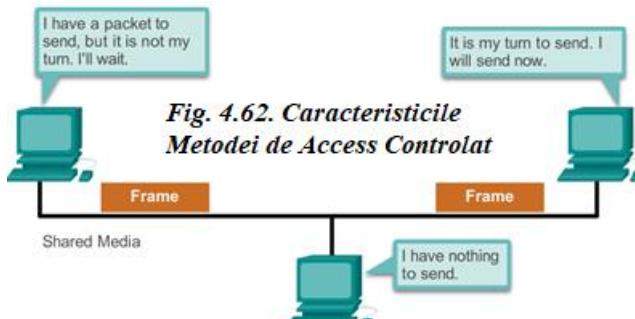
Exemple de acces controlat:

- *Token Ring (IEEE 802.5).*
- *Fiber Distributed Data Interface (FDDI) ce este bazat pe IEEE 802.4 token bus protocol.*

Notă: Ambele metode de control al accesului la mediul sunt considerate învechite.

Fig. 4.62 ilustrează următoarele:

- *Modul în care metodele de acces controlat funcționează.*
- *Caracteristici ale metodelor de acces controlat.*
- *Exemple de metode de acces controlat.*



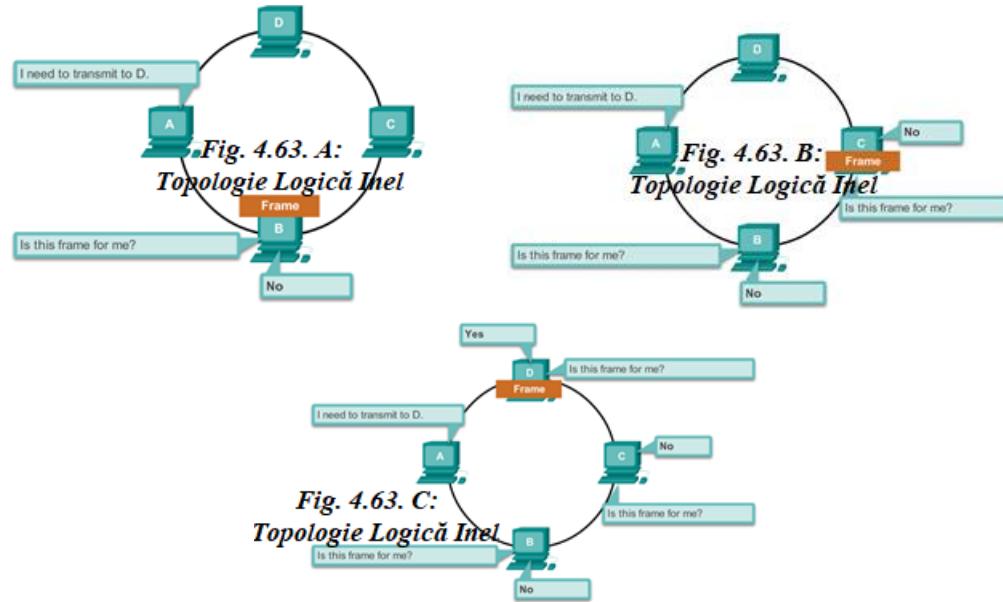
Characteristics	Controlled Access Technologies
<ul style="list-style-type: none"> • Only one station transmits at a time • Devices wishing to transmit must wait their turn • No collisions • May use a token passing method 	<ul style="list-style-type: none"> • Token Ring (IEEE 802.5) • Fiber Distributed Data Interface (FDDI)

Într-o topologie ring logică, fiecare nod din rând primește un frame. Dacă frameul nu îi este adresat nodului, nodul îl pasează la următorul nod. Acest lucru permite unui inel să utilizeze o tehnică de control al accesului la mediul controlat numită **"token passing"**.

Nodurile dintr-o topologie ring logică înălătură frameul din inel, examinează adresa și îl trimit mai departe dacă nu îi este adresat. Într-un inel, toate nodurile din inel (între nodul sursă și destinație) examinează frameul.

Există mai multe tehnici de control al accesului la mediul ce pot fi utilizate cu un inel logic, în funcție de nivelul de control necesar. De exemplu, numai un singur frame la un moment dat este transmis pe mediul. Dacă nu există date în transmisie, un semnal (cunoscut ca un token) poate fi plasat pe mediul și un nod poate plasa frameul de date în mediul numai atunci când are tokenul.

Atenție nivelul legătură de date “vede” o topologie ring logică. Topologia fizică cablată reală poate fi o altă topologie.



4.5.4.5 Frameul legăturii de date

Deși există mai multe protocoale de nivel legătură de date diferite ce descriu frameurile de la nivelul legătură de date, fiecare tip de frame are trei părți de bază:

- *Header*.
- *Data*.
- *Trailer*.

Toate protocoalele de la nivelul legătură de date încapsulează PDU de nivel 3 în câmpul de date al frameului. Însă, structura frameului și câmpurile conținute în header și trailer variază în funcție de protocol.

Protocolul de la nivelul legătură de date descrie caracteristicile necesare pentru transportul pachetelor prin diferite medii. Aceste caracteristici ale protocolului sunt integrate în încapsularea frameului. Atunci când frameul ajunge la destinația să și protocolul de la nivelul legătură de date preia frameul din mediu, informațiile încadrate sunt citite și înlăturate.

Nu există nici-o structură de frame ce îndeplinește nevoile tuturor mijloacelor de transport de date peste toate tipurile de mediu. În funcție de mediu, cantitatea de informații de control necesară în frame variază pentru a întâlni cerințele de control de acces la mediu ale mediului și ale topologiei logice.

Ca și în Fig. 4.64, un mediu fragil (sensibil) necesită mai mult control. Însă, un mediu protejat, ca și în Fig. 4.65, necesită mai puține controale.



Headerul frameului conține informații de control specificate de către protocolul de la nivelul legătură de date pentru topologia logică specifică și pentru mediul utilizat.

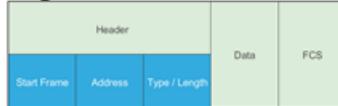
Informația de control a frameului este unică pentru fiecare tip de protocol. Este utilizată de către protocolul de nivel 2 pentru a furniza caracteristicile cerute de către mediul de comunicație.

Fig. 4.66 ilustrează câmpurile din headerul frameului Ethernet:

- **Câmpul “Start Frame”**: Indică începutul frameului.
 - **Câmpurile “Source and Destination Address”**: Indică nodurile sursă și destinație din mediu.
 - **Câmpul “Type”**: Indică serviciul de nivel superior conținut în frame.
- Protocolele diferite de la nivelul legătură de date ar putea utiliza câmpuri diferite de cele menționate. De exemplu, alte câmpuri ale frameului protocolului de nivel 2 ar putea fi:
- **Câmpul “Priority/Quality of Service”**: Indică un tip particular de serviciu de comunicație pentru prelucrare.
 - **Câmpul “Logical connection control”**: Utilizat pentru a stabili o conexiune logică între noduri.
 - **Câmpul “Physical link control”**: Utilizat pentru a stabili legătura la mediu.
 - **Câmpul “Flow control”**: Utilizat pentru a porni și opri traficul pe mediu.
 - **Câmpul “Congestion control”**: Indică congestie în mediu.

Deoarece scopurile și funcțiile protocolelor de la nivelul legătură de date sunt legate de topologiile specifice și de mediu, fiecare protocol trebuie să fie examinat pentru a căpăta o înțelegere detaliată a structurii frameului său. Când protocolele vor fi discutate în acest curs, vor fi explicate mai multe informații cu privire la structura frameului.

Fig. 4.66 Rolul Headerului



Nivelul legătură de date oferă adresarea ce este utilizată pentru transportul unui frame printr-un mediu local partajat. Adresele de dispozitiv de la acest nivel sunt referite ca adrese fizice. Adresarea de la nivelul legătură de date este conținută în interiorul headerului frameului și specifică nodul destinație din rețea locală. Headerul frameului ar putea conține de asemenea și adresa sursă a frameului.

Spre deosebire de adresele logice de nivel 3, ce sunt ierarhice, adresele fizice nu indică în ce rețea dispozitivul este localizat. Mai degrabă, adresa fizică este o adresă unică specifică a dispozitivului. Dacă dispozitivul este mutat într-o altă rețea sau subnet, va funcționa cu aceeași adresă fizică de nivel 2.

O adresă ce este specifică dispozitivului și non-ierarhică nu poate fi utilizată pentru a localiza un dispozitiv din rețelele mari sau din Internet. Acest lucru ar fi la fel cu încercarea găsirii unei singure case în întreaga lume, fără a avea nimic mai mult decât un număr de casă și un nume de stradă. Adresele fizice, însă, pot fi utilizate pentru a localiza un dispozitiv dintr-o arie limitată. Din acest motiv, adresa de la nivelul legătură de date este utilizată numai pentru livrarea locală. Adresele de la acest nivel nu au nici-un înțeles în afara rețelei locale. Putem compara acest lucru cu nivelul 3, unde adresele din headerul pachetului sunt transportate de la hostul sursă la hostul destinație, indiferent de numărul de hopuri din calea parcursă.

În cazul în care datele trebuie să treacă într-un alt segment de rețea, un dispozitiv intermediar, precum un router, este necesar. Routerul trebuie să accepte frameul bazându-se pe adresa fizică și să decapsuleze frameul pentru a examina adresa ierarhică, sau adresa IP. Folosind adresa IP, routerul este capabil să determine locația rețelei a dispozitivului destinație și cea mai bună cale pentru a ajunge la acesta. O dată ce știe unde să trimită pachetul, routerul crează un nou frame pentru pachet, iar noul frame este trimis pe următorul segment spre destinația finală.

Fig. 4.67 pune în evidență cerințele adresei de nivel 2 în topologiile multi-access și point-to-point.

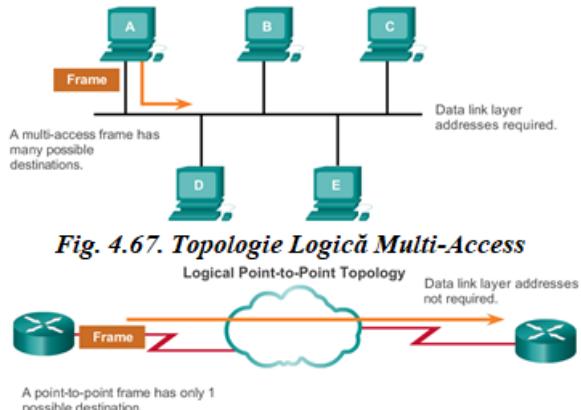


Fig. 4.67. Topologie Logică Multi-Access

Logical Point-to-Point Topology

Protocolele de la nivelul legătură de date adaugă un trailer la sfârșitul fiecărui frame. Trailerul este utilizat pentru a determina dacă un frame ajunge fără eroare. Acest proces se numește detectia erorii și este îndeplinit prin plasarea unei sume matematice sau logice de biți ce cuprind frameul în trailer. Detectia erorii este adăugata la nivelul legătură de date deoarece semnalele din mediu pot fi afectate de interferență, distorsiuni sau pierderi ce ar schimba substanțial valorile de biți ce sunt reprezentate de acele semnale.

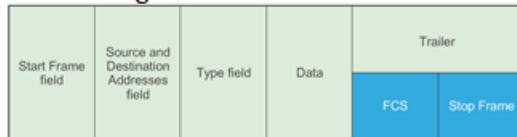
Un nod de transmisie crează un rezumat logic al conținutului frameului. Aceasta este cunoscut ca valoarea Cyclic Redundancy Check (CRC). Această valoare este plasată în câmpul Frame Check Sequence (FCS) al frameului pentru a reprezenta conținutul frameului.

Atunci când frameul ajunge la destinație, nodul receptor calculează propriul rezumat logic, sau CRC, al frameului. Nodul destinație compară cele două valori CRC. Dacă cele două valori sunt egale, frameul este considerat nealterat. Dacă valoarea CRC din FCS diferă de valoarea CRC calculată de către nodul destinatar, frameul este “aruncat”.

Prin urmare, câmpul FCS este utilizat pentru a determina dacă au avut loc erori în transmisia și primirea frameului. Mecanismul de detectie a erorii oferit prin utilizarea câmpului FCS descooperă multe erori ce au loc în mediu.

Există întotdeauna o mică posibilitate ca un frame cu un rezultat CRC bun să fie de fapt corupt. Erorile din biți ar putea să se anuleze unele pe celelalte atunci când CRC este calculat. Protocolele de nivel superior vor fi necesare pentru a detecta și corecta această pierdere de date.

Fig. 4.68. Remorca Cadrului



Într-o rețea TCP/IP, toate protocolele de nivel 2 lucrează cu IP la nivel 3 OSI. Însă, protocolul real de nivel 2 utilizat depinde de topologia logică a rețelei și de implementarea nivelului fizic. Având în vedere gama largă de medii fizice utilizate în gama de topologii din rețea, există un număr mare de protocole de nivel 2 în uz.

Fiecare protocol efectuează controlul accesului la mediu pentru topologii logice de nivel 2 specificate. Acest lucru înseamnă că un număr de dispozitive de rețea diferite pot acționa ca noduri ce funcționează la nivelul legătură de date atunci când se implementează aceste protocole. Aceste dispozitive includ network adapter sau network interface cards (NICs) de la computere sau interfețele de la routere sau switchurile de nivel 2.

Protocolul de nivel 2 utilizat pentru o topologie de rețea particulară este determinat de către tehnologia utilizată pentru implementarea topologiei. Tehnologia este, la rândul ei,

determinată de către mărimea rețelei – în termenii numărului de hosturi sau scopului geografic – și de serviciile ce trebuie să fie furnizate în rețea.

Un LAN utilizează în mod normal o tehnologie de lățime de bandă mare ce este capabilă să suporte un număr mare de hosturi. Aria geografică relativ restrânsă a LANului (o singură clădire sau un campus compus din mai multe clădiri) și densitatea mare de utilizatori face această tehnologie convenabilă cu privire la cost.

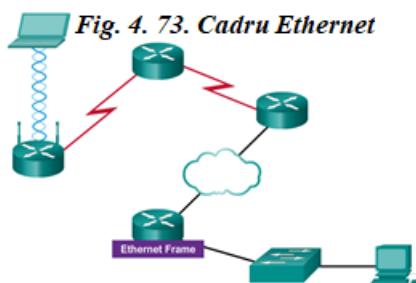
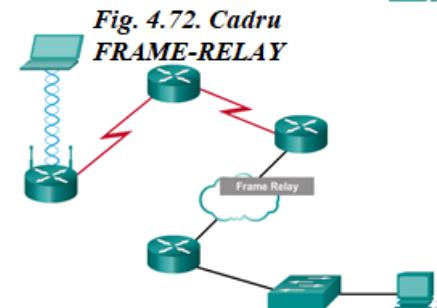
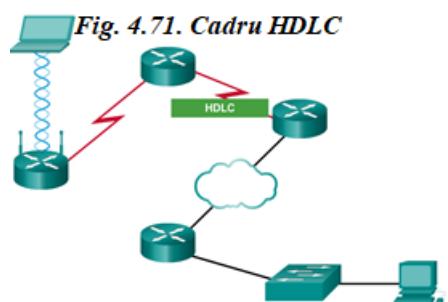
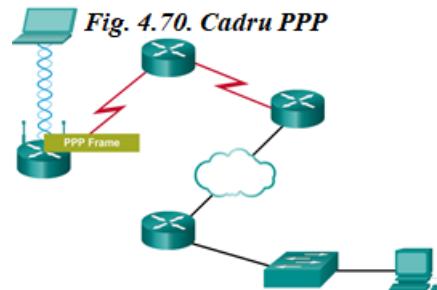
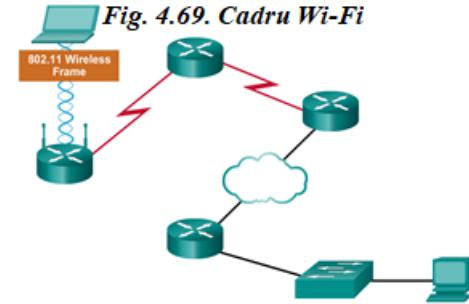
Însă, utilizarea unei tehnologii cu lățime de bandă mare nu este de obicei convenabilă în ceea ce privește costul pentru WANuri ce acoperă arii geografice mari (orașe sau mai multe orașe, de exemplu). Costul legăturilor fizice de mare distanță și tehnologiile utilizate pentru a transmite semnalele peste acele distanțe rezultă în mod normal într-o capacitate scăzută de lățime de bandă.

Diferențele de lățime de bandă rezultă în mod normal din utilizarea de protocoale diferite pentru LANurile și WANurile.

Protocole comune de la nivelul legătură de date:

- *Ethernet*
- *Point-to-Point Protocol (PPP)*
- *802.11 Wireless*

Alte protocole acoperite de curriculum sunt High-Level Data Link Control (HDLC) și Frame Relay.



4.6 Ethernet

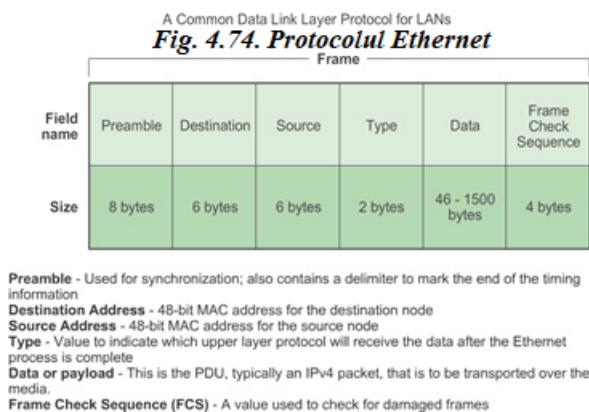
Ethernet este tehnologia LAN dominantă. Este o familie de tehnologii de rețea ce sunt definite de către standardele IEEE 802.2 și 802.3.

Standardele Ethernet definesc protocolele de nivel 2 și tehnologiile de nivel 1. Ethernet este cea mai utilizată tehnologie LAN din lume și suportă lățimi de bandă de 10 Mbps, 100 Mbps, 1 Gbps (1,000 Mbps), sau 10 Gbps (10,000 Mbps).

Formatul de bază al frameului și subnivelele IEEE ale nivelelor 1 și 2 din OSI rămân la fel în toate formele de Ethernet. Însă, metodele de detectare și plasare a datelor pe mediu variază în funcție de implementările diferite.

Ethernet oferă un "acknowledged connectionless service" peste un mediu partajat folosind CSMA/CD ca metodă de acces la mediu. Mediul partajat necesită ca headerul frameului Ethernet să utilizeze o adresă de nivel de legătură de date pentru a identifica nodurile sursă și destinație. Ca în cele mai multe protocole LAN, această adresă se numește adresa MAC a nodului. O adresă MAC Ethernet este alcătuită din 48 de biți și în general este reprezentată într-un format hexazecimal.

Fig. xxx ilustrează mai multe câmpuri ale frameului Ethernet. La nivelul legătură de date, structura frameului este aproape identică pentru toate vitezele tehnologiei Ethernet. Însă, la nivelul fizic, versiuni diferite ale Ethernet plasează biții pe mediu în mod diferit. Ethernet este discutat mai detaliat în următorul capitol.



4.7 Point-to-Point Protocol

Un alt protocol de la nivelul legătură de date este Point-to-Point Protocol (PPP). PPP este un protocol utilizat pentru livrarea frameurilor între două noduri. Spre deosebire de multe protocole de la nivelul legătură de date ce sunt definite de către organizațiile de inginerie electrică, standardul PPP este definit de către RFCs. PPP a fost dezvoltat ca un protocol WAN și rămâne alegerea pentru implementarea multor WANuri seriale. PPP poate fi utilizat pe diferite medii fizice, inclusiv twisted pair, liniile de fibră optică și transmisiile prin satelit, dar și pentru conexiunile virtuale.

PPP utilizează o arhitectură pe nivele. Pentru a se adapta diferitelor tipuri de medii, PPP stabilisce conexiuni logice, numite sesiuni, între două noduri. Sesiunea PPP ascunde mediul fizic de bază de protocolul PPP superior. Aceste sesiuni oferă de asemenea PPP cu o metodă pentru încapsularea mai multor protocole peste o legătură punct-la-punct. Fiecare protocol încapsulat peste o legătură își stabilisce propria sesiune PPP.

PPP permite de asemenea ca cele două noduri să negocieze opțiuni în cadrul sesiunii PPP. Acest lucru include autentificare, comprimare și multilink (utilizarea mai multor conexiuni fizice).

Fig. 4.75. Protocolul Punc-la-Punct

Frame						
Field name	Flag	Address	Control	Protocol	Data	FCS
Size	1 byte	1 byte	1 byte	2 bytes	variable	2 or 4 bytes

Flag - A single byte that indicates the beginning or end of a frame. The flag field consists of the binary sequence 01111110.
Address - A single byte that contains the standard PPP broadcast address. PPP does not assign individual station addresses.
Control - A single byte that contains the binary sequence 00000011, which calls for transmission of user data in an unsequenced frame.
Protocol - Two bytes that identify the protocol encapsulated in the data field of the frame. The most up-to-date values of the protocol field are specified in the most recent Assigned Numbers Request For Comments (RFC).
Data - Zero or more bytes that contain the datagram for the protocol specified in the protocol field.
Frame Check Sequence (FCS) - Normally 16 bits (2 bytes). By prior agreement, consenting PPP implementations can use a 32-bit (4-byte) FCS for improved error detection.

4.8 802.11 Wireless

Standardul IEEE 802.11 utilizează același standard 802.2 LLC și aceeași schemă de adresare pe 48 de biți ca alte LANuri 802. Însă, există multe diferențe la subnivelul MAC și nivelul fizic. Într-un mediu wireless, mediul necesită considerații speciale. Nu există nici-o conectivitate fizică definită; prin urmare, factorii externi ar putea interfera cu transferul de date și este dificilă controlarea accesului. Pentru a rezolva aceste provocări, standardele wireless au controale în plus.

Standardul IEEE 802.11 este referit adesea ca Wi-Fi. Este un sistem bazat pe "dispută" ce utilizează procesul de acces la mediu CSMA/CA. CSMA/CA specifică o procedură de backoff random pentru toate nodurile care așteaptă să transmită. Cea mai probabilă oportunitate pentru o dispută pe mediu este chiar imediat ce mediul devine disponibil. Făcând nodurile să aștepte (back off) pentru o perioadă aleatoare, se reduce mult posibilitatea unei coliziuni.

Rețelele 802.11 utilizează de asemenea "**data link acknowledgements**" pentru a confirma că un frame este primit cu succes. Dacă stația expeditoare nu detectează acknowledgement frame, fiindcă frameul de date original sau acknowledgment nu a fost primit nealterat, frameul este retransmis. Acest acknowledgment explicit combată interferența și alte probleme legate de radio.

Alte servicii suportate de către 802.11 sunt autentificarea, asocierea (conectivitatea la un dispozitiv wireless) și intimitățile (criptare).

Ca și în Fig. xxx, un frame 802.11 conține următoarele câmpuri:

- **Câmpul Protocol Version:** Versiunea frameului 802.11 utilizat.
- **Câmpurile Type și Subtype :** Identifică una dintre cele trei funcții și subfuncții ale frameului : control, data și management.
- **Câmpul To DS:** Setat 1 în frameurile de date destinate pentru sistemul de distribuție (dispozitivele din structura wireless).
- **Câmpul From DS :** Setat 1 pentru frameurile de date ce ies din sistemul de distribuție.
- **Câmpul More Fragments :** Setat 1 pentru frameurile ce au alt fragment.
- **Câmpul Retry :** Setat 1 dacă frameul este o retransmisie a unui frame mai vechi.
- **Câmpul Power Management :** Setat 1 pentru a indica faptul că un nod va fi în modul power-save.
- **Câmpul More Data :** Setat 1 pentru a indica unui nod aflat în modul power-save faptul că mai multe frameuri sunt buffered pentru respectivul nod.
- **Câmpul Wired Equivalent Privacy (WEP) :** Setat 1 dacă frameul conține informații criptate WEP pentru securitate.
- **Câmpul Order :** Setat 1 într-un data type frame ce utilizează Strictly Ordered service class (nu necesită reordonare).

- **Câmpul Duration/ID:** În funcție de tipul de frame, reprezintă fie timpul, în microsecunde, necesar să transmită frameul, fie o identitate asociată (AID) pentru stația ce transmite frameul.
- **Câmpul Destination Address (DA):** Adresa MAC a nodului destinație finală din rețea.
- **Câmpul Source Address (SA):** Adresa MAC a nodului sursă a frameului.
- **Câmpul Receiver Address (RA):** Adresa MAC ce identifică dispozitivul wireless ce este destinatarul imediat al frameului.
- **Câmpul Fragment Number :** Indică numărul pentru fiecare fragment al unui frame.
- **Câmpul Sequence Number:** Indică numărul de secvență atribuită frameului; frameurile retransmise sunt identificate prin duplicarea sequence numbers.
- **Câmpul Transmitter Address (TA):** Adresa MAC ce identifică dispozitivul wireless ce transmite frameul.
- **Câmpul Frame Body:** Conține informația ce este transmisă; pentru framele de date, de obicei un pachet IP.
- **Câmpul FCS :** Conține cyclic redundancy check (CRC) pe 32 de biți al frameului.

Fig. 4.76. Protocolul 802.11

pentru LAN Wi-Fi

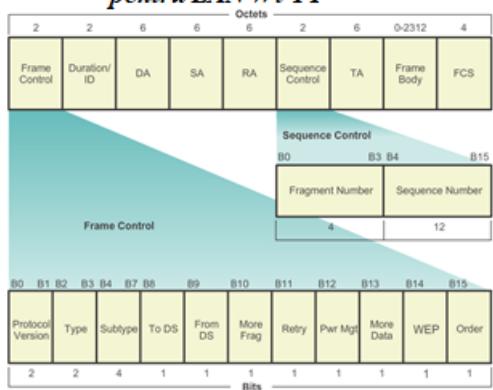
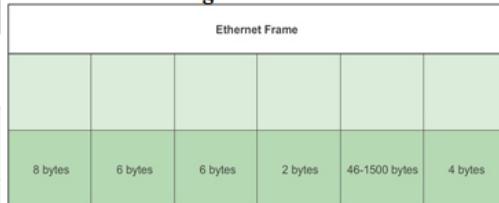
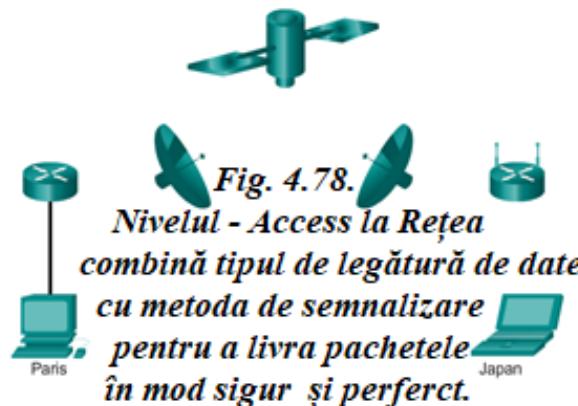


Fig. 4.77. Cadrul Ethernet



4.9 Concluzii Capitolul 4



Nivelul de acces la rețea TCP/IP este echivalent cu nivelul legătură de date OSI (Nivelul 2) și cu nivelul fizic (Nivelul 1).

Nivelul fizic OSI oferă metodele de transport al bițiilor, ce alcătuiesc un frame de la nivelul legătură de date, pe mediul de rețea. Componentele fizice sunt dispozitive hardware electronice, mediul și alți conectori ce transmit și transportă semnalele ce reprezintă biții. Componentele hardware precum network adapters (NICs), interfețele și conectorii, materiale de cablu și modelele de cablu sunt toate specificate în standarde asociate cu nivelul fizic. Standardele de la

nivelul fizic adresează trei arii funcționale: componentele fizice, tehnica de codificare de frame și metoda de semnalizare.

Utilizarea mediului adecvat este o parte importantă din comunicațiile din rețea. Fără o conexiune fizică adecvată, fie cablată sau wireless, comunicațiile dintre oricare două dispozitive nu vor avea loc.

Comunicația cablată constă din mediul de cupru și cablul de fibră.

- Există trei tipuri principale de cabluri de cupru utilizate în rețea: unshielded-twisted pair (UTP), shielded-twisted pair (STP) și cablul coaxial. Cablul UTP este cel mai cunoscut mediu de cupru din rețea.
- Cablul de fibră optică a devenit foarte popular pentru interconectarea dispozitivelor de rețea din infrastructură. Permite transmisia datelor peste distanțe mari cu bandwidth mai mare (data rates) decât orice alt mediu de rețea. Spre deosebire de cablurile de cupru, cablul de fibră optică poate transmite semnale cu o atenuare mai redusă și este complet imun la EMI și RFI.

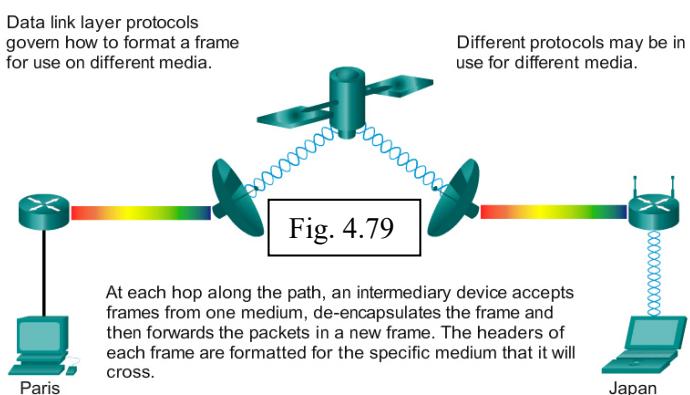
Mediul wireless transportă semnale electromagnetice ce reprezintă digiți binari ai comunicațiilor de date, folosind frecvențe radio sau microunde.

Numărul de dispozitive ce permit wireless continuă să crească. Din aceste motive, wireless a devenit mediu ales pentru rețelele de domiciliu și câștigă rapid popularitate în rețelele de întreprindere.

Nivelul legătură de date este responsabil pentru schimbul de frameuri între noduri peste un mediu de rețea fizic. Permite nivelelor superioare să acceseze mediu și controlează modul în care datele sunt plasate și primite prin mediu.

Împreună cu implementările diferite ale protocolelor de la nivelul legătură de date, există diferite metode de controlare a accesului la mediu. Aceste tehnici de control al accesului la mediu definesc modul în care nodurile împart mediu. Metoda reală de control al accesului la mediu utilizată depinde de topologie și de partajarea mediului. Topologiile LAN și WAN pot fi fizice sau logice. Topologia logică influențează tipul de network framing și controlul accesului la mediu utilizat. WANurile sunt interconectate de obicei utilizând topologii fizice point-to-point, hub and spoke, sau mesh. În LANurile cu mediu partajat, dispozitivele finale pot fi interconectate utilizând topologii fizice de tip stea, bus, inel sau extended star (hybrid).

Toate protocolele de la nivelul legătură de date încapsulează PDU de Nivel 3 cu câmpul de date al frameului. Însă, structura frameului și câmpurile conținute în header și trailer variază în funcție de protocol.



CAPITOLUL 5. TEHNOLOGIA ETHERNET

Introducere

Nivelul fizic OSI oferă metodele de transport al biților ce alcătuiesc un frame de la nivelul legătură de date prin mediul de rețea.

Ethernet este în prezent tehnologia LAN predominantă în lume. Ethernet funcționează la nivelul legătură de date și nivelul fizic. Standardele de protocol Ethernet definesc mai multe aspecte ale comunicațiilor din rețea, inclusiv formatul frameului, dimensiunea frameului, timing și codare. Atunci când mesajele sunt transmise între hosturi într-o rețea Ethernet, hosturile formatează mesajele în formatul de frame specificat de către standarde. Frameurile sunt de asemenea numite Protocol Data Units (PDUs).

Deoarece Ethernet este alcătuit din standarde la aceste nivele inferioare, acesta ar putea fi mai bine înțeles făcând referință la modelul OSI. Modelul OSI separă funcționalitățile nivelului legătură de date de adresare, framing și accesare a mediului de standardele de nivel fizic ale mediului. Standardele Ethernet definesc protocolele de nivel 2 și tehnologiile de nivel 1. Deși specificațiile Ethernet suportă medii diferite, lățimi de bandă diferite și alte variații de nivel 1 și 2, formatul de frame de bază și schema de adresare sunt aceleași pentru toate tipurile de Ethernet.

Acest capitol analizează caracteristicile și funcționarea tehnologiei Ethernet, aşa cum a evoluat de la mediul partajat, tehnologia de comunicații de date bazate pe "dispută" pe lățimea de bandă de astăzi, respectivă tehnologia full-duplex.



5.1 Protocolul Ethernet

5.1.1 Ethernet Operation

Ethernet este tehnologia LAN cea mai utilizată în lume din zilele noastre.

Ethernet funcționează la nivelul legătură de date și la nivelul fizic. Este o familie de tehnologii de rețea ce sunt definite de către standardele IEEE 802.2 și 802.3. Ethernet suportă lățimi de bandă de:

- 10 Mb/s.
- 100 Mb/s.
- 1000 Mb/s (1 Gb/s).
- 10.000 Mb/s (10 Gb/s).
- 40.000 Mb/s (40 Gb/s).
- 100.000 Mb/s (100 Gb/s).

Așa cum este ilustrat și în Fig. xxx 1, standardele Ethernet definesc protocolele de nivel 2 și tehnologiile de nivel 1. Pentru protocolele de nivel 2, cu toate standardele 802 IEEE,

Ethernet se bazează pe două subnivele separate ale nivelului legătură de date pentru a funcționa, subnivelele Logical Link Control (LLC) și Media Access Control (MAC).

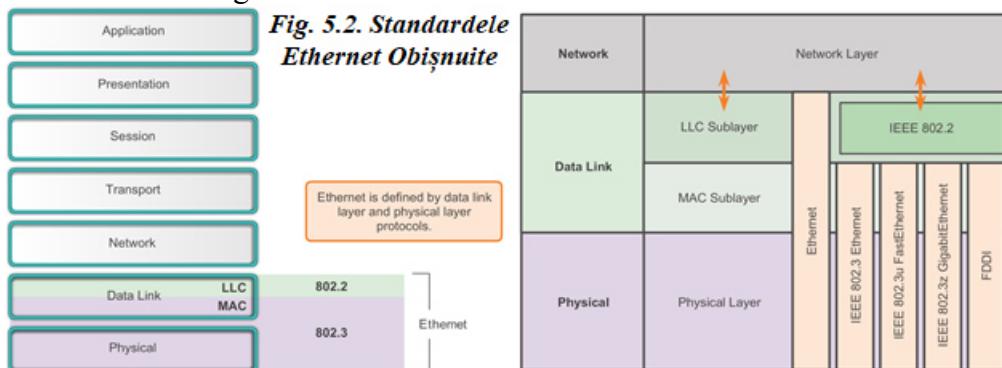
5.1.1.1 Subnivelul LLC

Subnivelul LLC Ethernet se ocupă de comunicarea dintre nivelele superioare și nivelele inferioare. Acest lucru se realizează în mod normal între networking software și device hardware. Subnivelul LLC preia datele de la protocolul de rețea, ce sunt de obicei un pachet IPv4, și adaugă informații de control ce ajută livrarea pachetului la nodul destinație. LLC este utilizat pentru a comunica cu nivelele superioare ale aplicației și trecerea pachetului la nivelele inferioare pentru livrare.

LLC este implementat în software și implementarea să este independentă de hardware. Într-un computer, LLC poate fi considerat driver software al NIC. NIC driver este un program ce interacționează direct cu hardwareul de pe NIC pentru a transmite datele între subnivelul MAC și mediul fizic.

5.1.1.2 Subnivelul MAC

MAC constituie subnivelul inferior al nivelului legătură de date. MAC este implementat pe hardware, în mod normal în placă de rețea a computerului. Specificațiile sunt înscrise în standardele IEEE 802.3. Fig. 5.2 listează standarde IEEE Ethernet comune.



Așa cum este ilustrat și în Fig. 5.3, sunbivelul MAC Ethernet are două responsabilități elementare:

- *Încapsularea datelor.*
- *Controlul accesului la mediu.*

5.1.2 Încapsularea datelor

Procesul de încapsulare a datelor include asamblarea frameului înaintea transmisiei și decapsularea frameului după primirea unui frame. În formarea frameului, nivelul MAC adaugă un header și un trailer la PDU-ul de nivel rețea.

Încapsularea datelor oferă trei funcții principale:

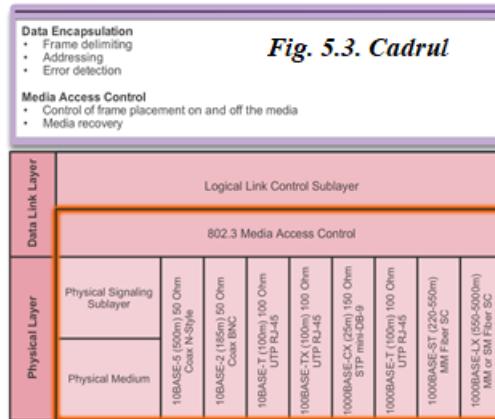
- *Delimitarea frameului:* Procesul de framing oferă delimitări importante ce sunt utilizate pentru a identifica un grup de biți ce alcătuiesc un frame. Acest proces oferă sincronizarea între nodurile de sursă și destinație.

- *Adresarea*: Procesul de încapsularea oferă de asemenea adresare pentru nivelul legătură de date. Fiecare header Ethernet adaugă în conținutul frameului adresa fizică (adresa MAC) ce permite unui frame să fie livrat la nodul destinație.
- *Detectia erorii*: Fiecare frame Ethernet conține un trailer cu un cyclic redundancy check (CRC) al conținutului frameului. După primirea unui frame, nodul destinatar crează un CRC pentru a-l compara cu cel din frame. Dacă cele două CRC calculate sunt identice, frameul a fost primit fără erori.

Utilizarea de frameuri ajută la transmiterea de biți în forma în care sunt plasați pe mediu și la gruparea bițiilor la nodul destinație.

Controlul accesului la mediu – A două responsabilitate a subnivelului MAC este controlul accesului la mediu. Controlul accesului la mediu este responsabil de plasarea cadrelor pe mediu și înlăturarea lor din mediu. Cum și numele său sugerează, controlează accesul la mediu. Subnivelul comunică direct cu nivelul fizic.

Topologia logică de bază Ethernet este magistrală mulți-access; prin urmare, toate nodurile (dispozitivele) de pe un singur segment de rețea împart mediul. Ethernet este o metodă bazată pe “dispută” de rețea. Reamintim că o metodă bazată pe dispută, sau o metodă non-deterministică, înseamnă că orice dispozitiv poate încerca să transmită date pe mediul partajat oricând are date de transmis. Însă, în cazul în care doi oameni încearcă să vorbească simultan, dacă mai multe dispozitive dintr-un singur mediu încearcă să trimită date simultan, datele se vor “ciocni” rezultând date corupte, neutilizabile. Din acest motiv, Ethernet oferă o metodă de controlare a modului în care nodurile împart accesul prin utilizarea tehnologiei Carrier Sense Multiple Access (CSMA).



Procesul CSMA este utilizat pentru a detecta mai întâi dacă mediul transferă un semnal. Dacă un semnal este pe mediu de la un alt nod, înseamnă că un alt dispozitiv transmite. Atunci când dispozitivul încearcă să transmită “vede” că mediul este ocupat, va aștepta și va încerca din nou după o scurtă perioadă de timp. Dacă nu este detectat nici-un semnal, dispozitivul își transmite datele. Este posibil ca procesul CSMA să nu funcționeze corect și două dispozitive să transmită în același timp. Acest lucru se numește coliziune a datelor. Dacă are loc, datele transmise de cele două dispozitive vor fi corupte și va fi necesară retransmiterea lor.

Metodele de control al accesului la mediu bazate pe “dispută” nu necesită mecanisme de depistarea a cărui dispozitiv îi este rândul să acceseze mediul; prin urmare, ele nu au overhead al metodelor cu acces controlat. Însă, sistemele bazate pe “dispută” nu se pretează bine la utilizarea intensivă a mediului. O dată cu creșterea numărului de noduri și a utilizării, probabilitatea de acces la mediu cu succes, fără coliziune, scade. În plus, mecanismele de recuperare necesare pentru a corecta erorile în timpul acestor coliziuni diminuează și mai mult throughputul.

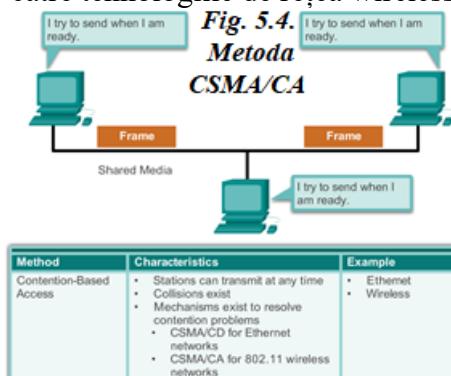
Așa cum este ilustrat și în Fig. xxx, CSMA este implementat în mod normal împreună cu o metodă de rezolvare a “disputei” de mediu. Cele două metode comune utilizate sunt:

➤ **CSMA/Collision Detection** - În CSMA/Collision Detection (CSMA/CD), dispozitivele monitorizează mediul pentru prezența unui semnal de date. În cazul în care un semnal de date este absent, indicând că mediul este liber, dispozitivul transmite datele. Dacă semnalele sunt apoi detectate, arătând că un alt dispozitiv a transmis în același timp, toate dispozitivele încetează transmisia și încearcă mai târziu. Formele tradiționale de Ethernet au fost dezvoltate pentru a utiliza această metodă.

Încorporarea pe scară largă a tehnologiilor de switching în rețelele moderne a înlocuit mult nevoia inițială pentru CSMA/CD în rețelele cu arie locală de acțiune. Aproape toate conexiunile dintre dispozitivele dintr-un LAN de astăzi sunt conexiuni full-duplex - un dispozitiv este capabil să trimită și să primească date simultan. Acest lucru înseamnă că deși rețelele Ethernet sunt dezvoltate cu tehnologia CSMA/CD, cu dispozitivele intermediare de astăzi, coliziunile nu au loc și procesele utilizate de către CSMAC/CD sunt într-adevăr inutile.

Însă, conexiunile wireless într-un mediu LAN încă trebuie să ia în considerare coliziunile. Dispozitivele LAN wireless utilizează metoda de acces la mediu CSMA/Collision Avoidance (CSMA/CA).

➤ **CSMA/Collision Avoidance** - În CSMA/CA, dispozitivele examinează mediul de prezență unui semnal de date. În cazul în care mediul este liber, dispozitivul transmite o notificare în mediul cu intenția să de-a-l utilizeze. Dispozitivul transmite apoi datele. Această metodă este utilizată de către tehnologiile de rețea wireless 802.11.



Așa cum s-a menționat și mai devreme, topologia de bază logică a Ethernet este o magistrală multi-access. Fiecare dispozitiv de rețea este conectat la același mediu partajat și toate nodurile primesc toate frameurile transmise. Problema este aceea că dacă toate dispozitivele primesc fiecare frame, cum poate fiecare dispozitiv să identifice dacă este destinatarul fără overhead necesar pentru procesare și fără decapsularea frameului pentru a obține adresa IP? Problema a devenit mai mare în rețele cu volum mare de trafic unde sunt transmise multe frameuri.

Pentru a preveni overhead excesiv implicat în procesarea fiecărui frame, un identificator unic numit adresa MAC a fost creat pentru a identifica nodurile sursă și destinație reale într-o rețea Ethernet. Indiferent de ce Ethernet este utilizat, adresarea MAC oferă o metodă pentru identificarea de dispozitiv la nivelul inferior al modelului OSI. După cum este bine cunoscut, adresarea MAC face parte dintr-un PDU de nivel 2. O adresă MAC Ethernet este o valoare pe 48 de biți exprimată ca 12 săgeți hexazecimale (4 biți pe săgeță hexazecimal).

5.1.3 Structura adresei MAC

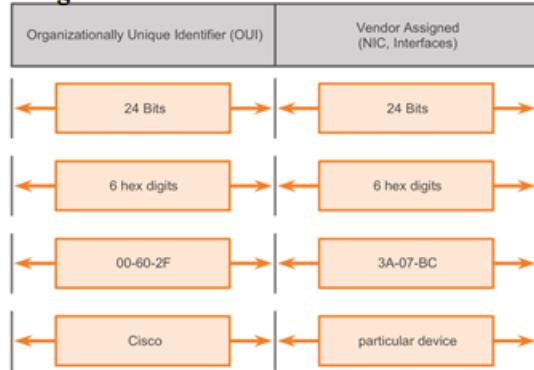
Adresa MAC trebuie să fie unică la nivel global. Valoarea adresei MAC este un rezultat direct al regulilor IEEE cerute furnizorilor pentru a asigura adrese unice la nivel global pentru fiecare dispozitiv Ethernet. Regulile stabilite de către IEEE cer ca orice furnizor ce vinde

dispozitive Ethernet să se înregistreze cu IEEE. IEEE stabilește furnizorului un cod de 24 de biți, numit Organizational Unique Identifier (OUI).

IEEE cere unui furnizor să urmeze două reguli simple, ilustrate și în Fig. xxx:

- *Toate adresele MAC atribuite unui NIC sau altui dispozitiv Ethernet, trebuie să utilizeze OUI atribuit furnizorului ca primii trei bytes.*
- *Toate adresele MAC cu același OUI trebuie să aibă atribuite o valoare unică (cod de furnizor sau număr serial) în ultimii trei bytes.*

Fig. 5.5. Structura Adresei MAC-Ethernet



Adresa MAC este adesea referită ca o adresă "burned-in address (BIA)" deoarece, din punct de vedere istoric, această adresă este "arsă" în ROM (Read-Only Memory) pe NIC. Acest lucru înseamnă că adresa este codată în ROM chip permanent - nu poate fi schimbată de software.

Notă: Pe sistemele moderne de operare sau NICuri, este posibilă schimbarea adresei MAC în software. Acest lucru este util atunci când se încearcă câștigarea accesului la o rețea ce filtrează bazându-se pe BIA- prin urmare, filtrarea sau controlul traficului bazat pe adresa MAC nu este la fel de sigur.

Adresele MAC sunt asignate la stații de lucru, servere, imprimante, switchuri și routere - orice dispozitiv care trebuie să trimită sau să primească date din rețea. Toate dispozitivele conectate la un LAN Ethernet au interfețe adresate prin MAC. Diferiți producători de hardware și software ar putea reprezenta adresa MAC în formate hexazecimale diferite. Formatele de adresă ar putea fi exprimate astfel:

- 00-05-9A-3C-78-00
- 00:05:9A:3C:78:00
- 0005.9A3C.7800

Atunci când un computer pornește, primul lucru pe care îl face placa de rețea este să copieze adresa MAC din ROM în RAM. Atunci când un dispozitiv transmite un mesaj într-o rețea Ethernet, atașează informații de header la pachet. Informațiile din header conțin adresele MAC sursă și destinație. Dispozitivul sursă trimite datele prin rețea.

Fiecare NIC din rețea vizualizează informațiile, la subnivelul MAC, pentru a vedea dacă adresa MAC destinație din frame corespunde adresei MAC fizică a dispozitivului stocată în RAM. Dacă nu corespunde, dispozitivul "aruncă" frameul. Atunci când frameul ajunge la destinație, unde MACul din frame corespunde cu cel de pe NIC, NIC transferă frameul la nivelele superioare din stiva OSI, unde procesul de decapsulare are loc.

I need to send information to H3.

Fig. 5.6. A: Transferul de Cadre

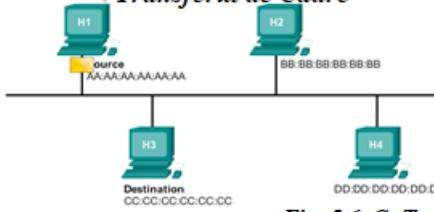
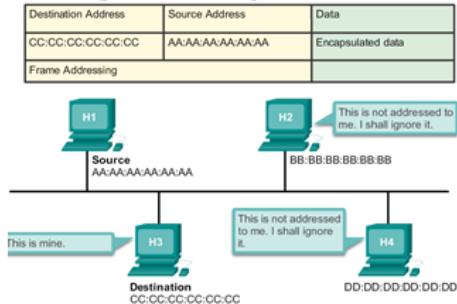


Fig. 5.6. B: Transferul de Cadre

Destination Address	Source Address	Data
CC:CC:CC:CC:CC:CC	AA:AA:AA:AA:AA:AA	Encapsulated data
Frame Addressing		

Fig. 5.6. C: Transferul de Cadre



5.2 Atributele frame-ului Ethernet

De la crearea Ethernet în 1973, standardele au evoluat pentru a specifica versiuni mai rapide și mai flexibile ale tehnologiei. Această abilitate a Ethernet să se îmbunătățească în timp este unul dintre principalele motive pentru care a devenit aşa de popular. Versiunile inițiale ale Ethernet erau relativ încete, 10Mbps. Ultimile versiuni ale Ethernet funcționează la 10 Gigabits per second sau mai rapid. Fig. 5.7 pune în evidență schimbările din versiuni diferite ale Ethernet.

La nivelul legătură de date, structura frameului este aproape identică pentru toate vitezele Ethernet. Structura frameului Ethernet adaugă headere și trailer în jurul PDU-ului de nivel 3 pentru a încapsula mesajul ce trebuie transmis.

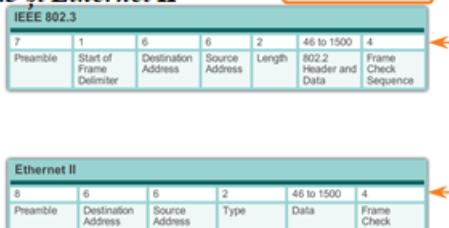
Atât headerul, cât și trailerul, Ethernet au mai multe secțiuni de informații ce sunt utilizate de protocolul Ethernet. Fiecare secțiune a frameului se numește câmp. Precum în Fig. 5.7, există două stiluri de Ethernet framing:

- Standardul IEEE 802.3 Ethernet ce a fost actualizat de mai multe ori pentru a include noi tehnologii.
- Standardul DIX Ethernet ce este referit ca Ethernet II.

Diferențele dintre stilurile de cadre sunt minime. Cea mai semnificativă diferență dintre cele două standarde este adăugarea unui Start Frame Delimiter (SFD) și schimbarea câmpului Type în câmpul Length în 802.3.

Ethernet II este formatul de frame Ethernet utilizat în retelele TCP/IP.

Fig. 5.7. Comparări între structura cadrele și dimensiunea câmpurilor pentru protocoalele 802.3 și Ethernet II



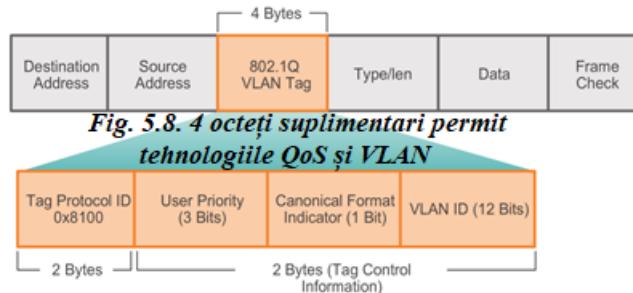
Ambele standarde, Ethernet II și IEEE 802.3, definesc dimensiunea de frame minimă la 64 bytes și maximă la 1518 bytes. Acestea includ toți biții de la câmpul Destination MAC Address al câmpului Frame Check Sequence (FCS). Câmpurile Preamble și Start Frame Delimiter nu sunt incluse atunci când se descrie dimensiunea unui frame.

Orice frame cu o dimensiune mai mică decât 64 de bytes este considerat "collision fragment" sau "runt frame" și este automat "aruncat" de către stațiile ce îl primesc.

Standardul IEEE 802.3ac, conceput în 1998, extinde dimensiunea maximă permisă a frameului la 1522 bytes. Dimensiunea frameului a crescut pentru a susține o tehnologie numită Virtual Local Area Network (VLAN). VLANs sunt create într-o rețea de switchuri și vor fi prezentate într-un curs viitor. De asemenea, mai multe tehnologii de quality of service (QoS) folosesc câmpul User Priority pentru a implementa mai multe nivele ale serviciului, cum ar fi serviciu de prioritate pentru traficul de voce. Fig. xxx pune în evidență câmpurile conținute în eticheta protocolului 802.1Q VLAN.

Dacă dimensiunea unui frame transmis este mai mică decât cea minimă sau mai mare decât cea maximă, dispozitivul destinatar aruncă frameul. Frameurile aruncate sunt de obicei rezultatul coliziunilor sau a altor semnale nedorite și prin urmare sunt considerate invalide.

La nivelul legătură de date structura frameului este aproape identică. La nivelul fizic variații diferite ale Ethernet diferă în metoda lor de detectare și plasarea a datelor pe mediu.



Principalele câmpuri ale frameului Ethernet sunt:

- **Câmpurile Preamble și Start Frame Delimiter:** Câmpurile Preamble (7 bytes) și Start Frame Delimiter (SFD), numit și Start of Frame (1 byte), sunt utilizate pentru sincronizarea dintre dispozitivele sursă și destinație. Acești primi 8 bytes ai frameului sunt utilizati pentru a capta atenția nodurilor destinație. În esență, primii bytes transmit destinatarilor să se pregătească să primească un nou frame.
- **Câmpul Destination MAC Address:** Acest câmp de 6 bytes este identificatorul pentru destinația dorită. Așa cum este bine știut, această adresă este utilizată de către Nivelul 2 pentru a ajuta dispozitivele să determine dacă un frame este adresat lor. Adresa din frame este comparată cu adresa MAC a dispozitivului. Dacă cele două sunt identice, dispozitivul acceptă frameul.
- **Câmpul Source MAC Address:** Acest câmp de 6 bytes identifică NICul sau interfața de origine a frameului.
- **Câmpul length:** Pentru orice standard IEEE 802.3 mai vechi decât 1997, câmpul length definește dimensiunea exactă a câmpului de date al frameului. Acesta este utilizat mai târziu ca parte a FCS pentru asigurarea faptului că mesajul a fost primit cu succes. În caz contrar, scopul câmpului este de a descrie ce protocol de nivel superior este prezent. Dacă valoarea a doi octeți este egală sau mai mare decât 0x0600 hexadecimale sau 1536 zecimal, conținutul câmpului Data este decodat în funcție de protocolul indicat. Dacă valoarea este egală sau mai mică decât 0x05DC hexadecimale sau 1500 zecimal, câmpul length este utilizat pentru a indica folosirea formatului de frame IEEE 802.3. Astfel sunt diferențiate frameurile Ethernet II și 802.3.

- **Câmpul Data:** Acest câmp (46 - 1500 bytes) conține datele încapsulate de la un nivel superior, de obicei un PDU de nivel 3, sau un pachet IPv4. Toate frameurile trebuie să aibă cel puțin 64bytes. Dacă un pachet mic este încapsulat, biți adiționali numiți "pad" sunt adăugați pentru a crește dimensiunea frameului la dimensiunea minimă.
- **Câmpul Frame Check Sequence:** Câmpul Frame Check Sequence (4 bytes) este utilizat pentru a detecta erorile dintr-un frame. Utilizează o valoare Cyclic Redundancy Check (CRC). Dispozitivul sursă include rezultatul unui CRC în câmpul FCS al frameului. Dispozitivul destinație primește frameul și generează un CRC pentru a detecta erori. Dacă cele două calcule corespund, nu a avut loc nici-o eroare. Altfel, este o informare că datele au fost schimbate; prin urmare, frameul este aruncat. O schimbare în date poate fi rezultatul unei întreruperi a semnalelor electrice ce reprezintă biți.

Fig. 5.9. IEEE 802.3

7 Preamble	1 Start of Frame Delimiter	6 Destination Address	6 Source Address	2 Length	46 to 1500 802.2 Header and Data	4 Frame Check Sequence
---------------	-------------------------------	--------------------------	---------------------	-------------	-------------------------------------	---------------------------

5.3 Ethernet MAC

Folosirea adresei MAC este unul dintre cele mai importante aspecte ale tehnologiei LAN Ethernet. Adresele MAC folosesc numere hexazecimale.

Hexazecimal este un cuvânt utilizat atât ca substantiv, cât și ca adjecțiv. Cât este utilizat ca un substantiv, înseamnă sistemul de numerație hexazecimal. Hexazecimal oferă un mod convenabil de reprezentare a valorilor binare. Așa cum zecimal este un sistem de numere în bază 10 și binar este un sistem de numere în bază 2, hexazecimal este un sistem în bază 16.

Sistemul de numere în bază 16 utilizează numerele de la 0 la 9 și literele de la A la F. Fig. xxx 1 prezintă valorile zecimale și hexazecimale ale codurilor binare de la 0000 la 1111. Este mai ușor pentru noi să exprimăm o valoare ca un digit hexazecimal, decât prin patru biți binari.

Având în vedere că 8 biți (1 byte) este o grupare binară comună, codurile binare de la 00000000 la 11111111 pot fi reprezentate în hexazecimal de la 00 la FF. Zerourile de la început sunt afișate întotdeauna pentru a completa reprezentarea de 8 biți. De exemplu, valoarea binară 0000 1010 este reprezentată în hexazecimal ca 0A.

Notă: Este importantă diferențierea dintre valorile hexazecimale și cele zecimale în ceea ce privește cifrele de la 0 la 9, așa cum este evidențiat și în Fig. xxx 1.

Reprezentarea valorilor hexazecimale – Hexazecimal este de obicei reprezentat în text de valoarea precedată de 0x (de exemplu 0x73) sau cu un indice 16. Mai puțin comun, poate fi urmată de către un H, de exemplu 73H. Însă, deoarece textul cu indice nu este recunoscut în linia de comandă sau în mediile de programare, reprezentarea tehnică hexazecimală este precedată de 0x (zero x). Prin urmare, exemplele de mai sus vor fi reprezentate ca 0xA sau respectiv 0x73.

Hexazecimal este utilizat pentru a reprezenta adresele Ethernet MAC și adresele IPv6.

Conversile hexazecimale – Conversile numerice dintre valorile zecimale și hexazecimale sunt simple, însă înmulțirea sau împărțirea rapidă cu 16 nu este întotdeauna convenabilă. Dacă asemenea conversii sunt necesare, ar fi mai ușoară convertirea valorilor zecimale sau hexazecimale în binar, urmată apoi de conversia valorii binare în valoarea zecimală, respectiv hexazecimală.

Prin practică, este posibilă recunoașterea tipelor de biți binari ce corespund valorilor zecimale sau hexazecimale. Fig. 5.10 arată aceste tipuri pentru valorile de 8 biți selectate.

Fig. 5.10. Hexadecimal Numbering
Decimal and Binary equivalents of 0 to F Hexadecimal

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Fig. 5.11. Hexadecimal Numbering
Selected Decimal, Binary, and Hexadecimal equivalents

Decimal	Binary	Hexadecimal
0	0000 0000	00
1	0000 0001	01
2	0000 0010	02
3	0000 0011	03
4	0000 0100	04
5	0000 0101	05
6	0000 0110	06
7	0000 0111	07
8	0000 1000	08
10	0000 1010	0A
15	0000 1111	0F
16	0001 0000	10
32	0010 0000	20
64	0100 0000	40
128	1000 0000	80
192	1100 0000	C0
202	1100 1010	CA
240	1111 0000	F0
255	1111 1111	FF

Pe un host cu Sistem de Operare Windows, comanda **ipconfig /all** poate fi utilizată pentru a identifica adresa Mac de pe un adaptor Ethernet. În Fig. 5.12, se observă Physical Address (MAC) a computerului ca fiind 00-18-DE-C7-F3-FB. Ceva asemănător se poate testa și pe sistemul pe care se lucrează.

În funcție de dispozitiv și de sistemul de operare, vor fi reprezentări diferite ale adreselor MAC, ca în Fig. 5.13. Routerele și switchurile utilizează forma XXXX.XXXX.XXXX, unde X este un caracter hexazecimal.

C:\>ipconfig/all

Ethernet adapter Local Area Connection:

```

Connection-specific DNS Suffix . : example.com
Description . . . . . : Intel(R) Gigabit Network Connection
Physical Address . . . . . : 00-18-DE-C7-F3-F8
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . . : Yes
IPv4 Address. . . . . : 192.168.1.67 (Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : Monday, November 26, 2012 12:14:48 PM
Lease Expires . . . . . : Saturday, December 01, 2012 12:15:02 AM
Default Gateway . . . . . : 192.168.1.254
DHCP Server . . . . . : 192.168.1.254
DNS Servers . . . . . : 192.168.1.254

```

Fig. 5.12. Adresa MAC=fizică.

Fig. 5.13. Reprezentare adrese MAC

With Dashes 00-60-2F-3A-07-BC

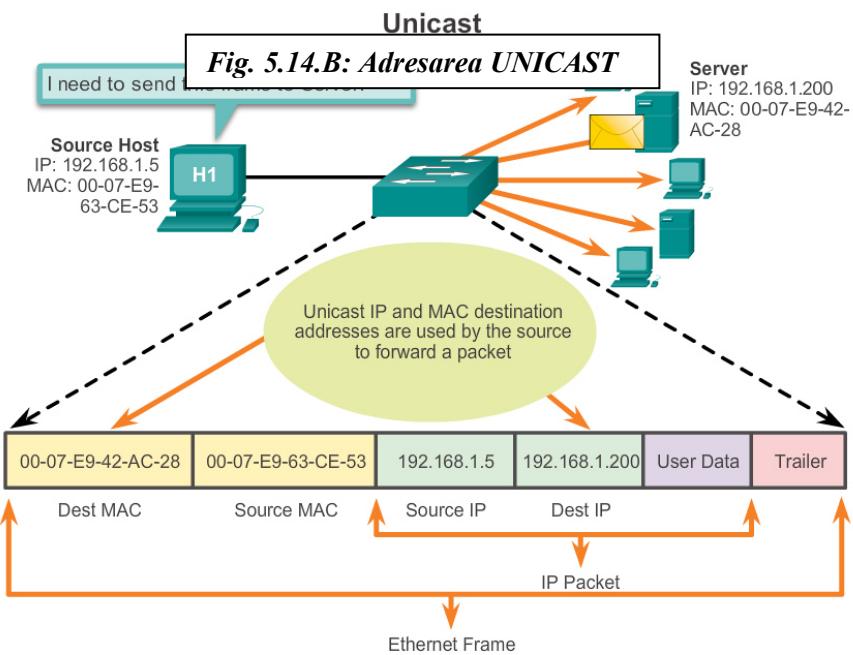
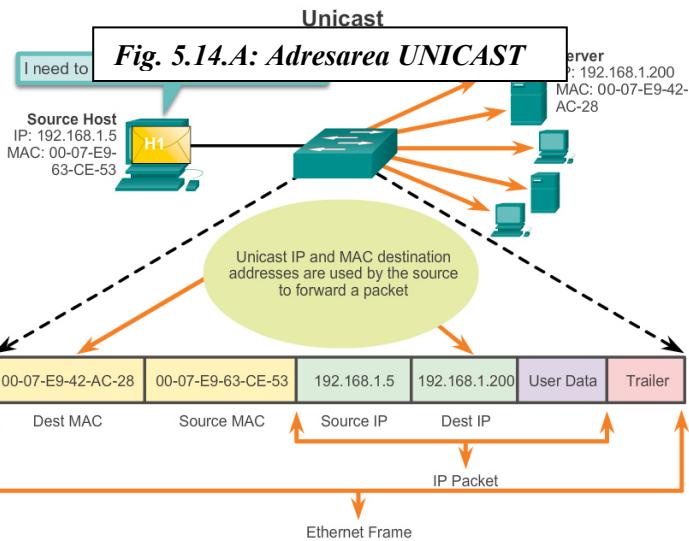
With Colons 00:60:2F:3A:07:BC

With Periods 0060.2F3A.07BC

În Ethernet, adrese MAC diferite sunt utilizate pentru comunicațiile unicast, broadcast și multicast.

O adresă MAC unică este adresa unică utilizată atunci când un frame este trimis de la un singur dispozitiv către un singur dispozitiv destinație.

În exemplul din Fig. xxx, un host cu adresa IP 192.168.1.5 (sursă) cere o pagină web de la serverul cu adresa IP 192.168.1.200. Pentru ca un pachet unicast să fie transmis și primit, o adresă IP destinație trebuie să fie existentă în headerul pachetului IP. O adresă MAC destinație trebuie de asemenea să fie prezentă în headerul frameului Ethernet. Adresa IP și adresa MAC se combină pentru a livra datele la un host destinație corespunzător.



Un pachet broadcast conține o adresă IP destinație care are numai 1 în partea de host. Acest lucru înseamnă că toate hosturile din rețeaua locală (domeniul de broadcast) vor primi și procesa pachetul. Mai multe protocoale de rețea, precum DHCP și Address Resolution Protocol (ARP), utilizează broadcast. Modul în care ARP utilizează broadcast pentru a mări adresele de nivel 2 în adrese de nivel 3 este discutat mai departe în acest capitol.

Ca și în Fig. 5.15 de mai jos, o adresă IP broadcast pentru o rețea necesită o adresă MAC broadcast corespondentă în frameul Ethernet. În rețelele Ethernet, adresa MAC broadcast este FF-FF-FF-FF-FF-FF.

Fig. 5.16. A: Adresarea BROADCAST

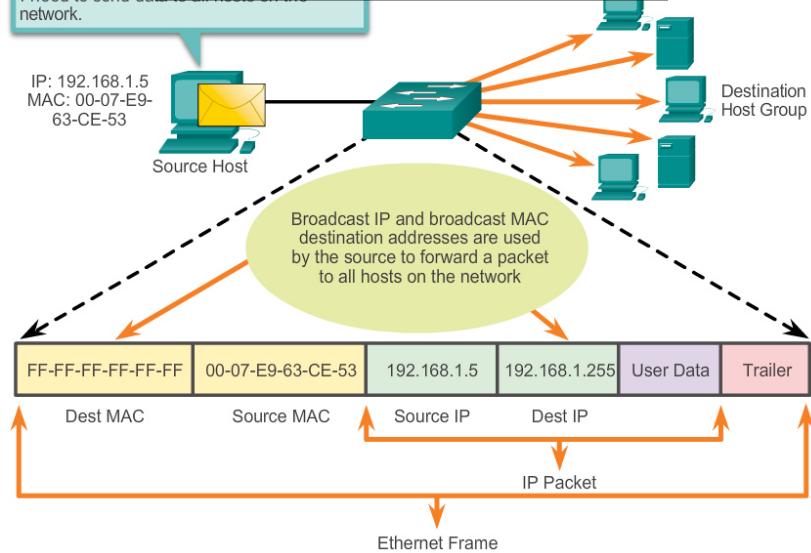
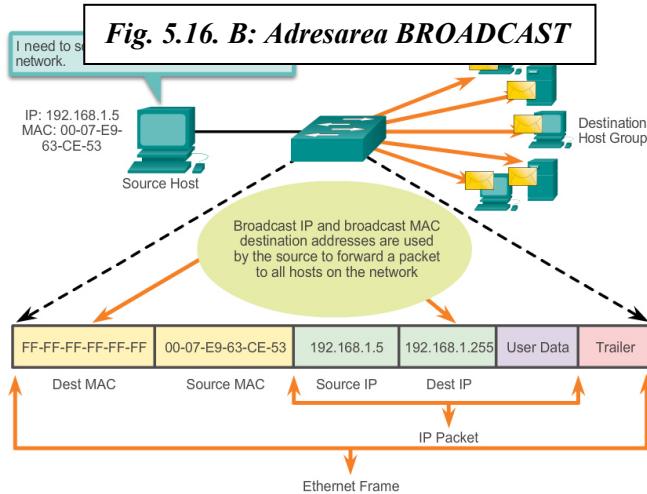


Fig. 5.16. B: Adresarea BROADCAST

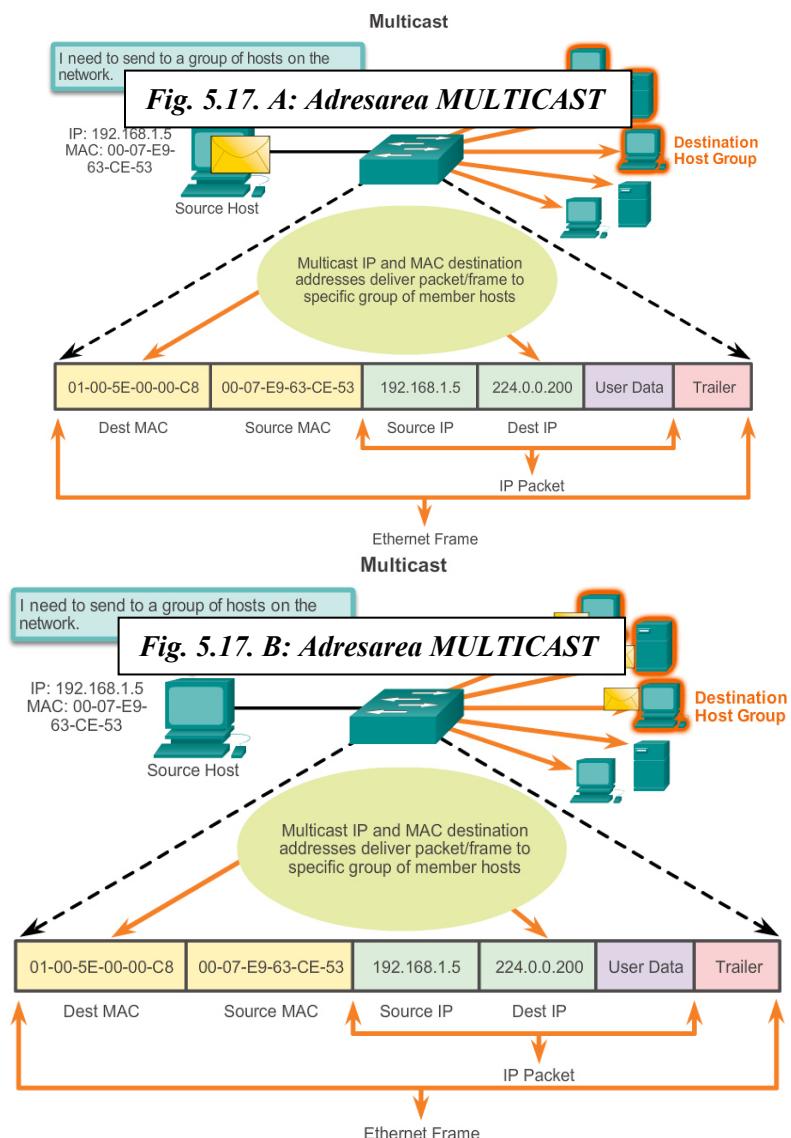


Adresele multicast permit unui dispozitiv sursă să trimită un pachet la un grup de dispozitive. Dispozitivele ce aparțin unui grup multicast au atribuite un grup de adrese IP. Intervalul adreselor multicast este de la 224.0.0.0 la 239.255.255.255. Deoarece adresele multicast reprezintă un grup de adrese (numit uneori un grup de hosturi), pot fi folosite numai ca destinație a unui pachet. Sursa va fi întotdeauna o adresă unică.

Adresele multicast vor fi utilizate în “**remote gaming**”, unde mai mulți jucători sunt conectați de la distanță, dar joacă același joc. O altă utilizare a adreselor multicast este în învățarea la distanță prin intermediul conferințelor video, unde mai mulți studenți sunt conectați la aceeași clasă.

Ca și adresele unicast și broadcast, adresa IP multicast necesită o adresă MAC corespondentă pentru a livra frameurile într-o rețea locală. Adresa MAC multicast este o valoare specială care începe cu 01-00-5E în hexadecimale. Partea rămasă din adresa MAC este creată prin convertirea celor 23 de biți inferiori din adresa de grup IP multicast în 6 caractere hexazecimale.

Un exemplu, din Fig.5.17, este adresa hexazecimală multicast 01-00-5E-00-00-C8.



5.4 ADRESELE DE TIP : MAC și IP

Există două adrese importante asignate unui dispozitiv host:

- *Adresa fizică (adresa MAC).*
- *Adresa logică (adresa IP).*

Ambele, adresa MAC și adresa IP, lucrează împreună pentru a identifica un dispozitiv din rețea. Procesul de utilizare a adresei MAC și adresei IP pentru a localiza un computer este similar cu procesul de utilizare a unui nume și adresă pentru un individ pentru a transmite/primi o scrisoare.

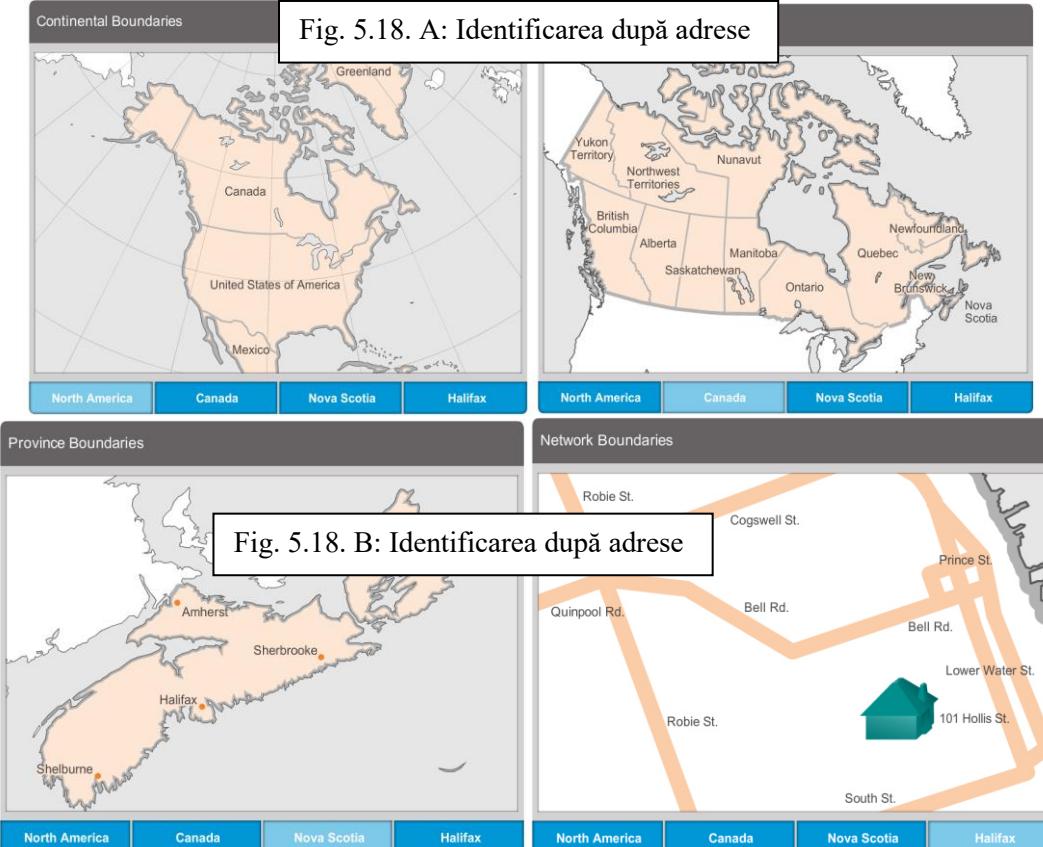
Numele unei persoane, de obicei, nu se schimbă. Adresa unei persoane, pe de altă parte, este legată de locul în care trăiește și se poate schimba.

Similar cu numele unei persoane, adresa MAC a unui host nu se schimbă; este atribuită fizic pe NICul hostului și este cunoscută ca adresa fizică. Adresa fizică rămâne aceeași indiferent de locul în care este plasat hostul.

Adresa IP este similară cu adresa unei persoane. Această adresă este bazată pe locul în care hostul se află. Utilizând această adresă, este posibil pentru un frame să determine locația unde frameul ar trebui să fie transmis. Adresa IP, sau adresa de rețea, este cunoscută ca adresa logică

deoarece este asignată local. Este atribuită la fiecare host de către un administrator de rețea, bazându-se pe rețeaua locală unde hostul este conectat. Fig. 5.18 demonstrează natura ierarhică din localizarea unui individ bazată pe adresa logică.

Ambele, adresa MAC și adresa IP logică, sunt necesare pentru ca un computer să comunice într-o rețea ierarhică, la fel cum și adresa și numele unui persoane sunt necesare pentru a livra scrisoarea.



Un dispozitiv sursă va trimite un pachet în funcție de o adresă IP. Unul dintre cele mai comune moduri în care un dispozitiv sursă determină adresa IP a unui dispozitiv destinație este prin intermediul Domain Name Service (DNS), în care o adresă IP este asociată unui nume de domeniu. De exemplu, www.google.com este egal cu 8.8.8. Această adresă IP va primi pachetul de la locația de rețea a dispozitivului destinație. Această adresă IP este folosită de către routere pentru a determina cea mai bună cale spre destinație. Deci, pe scurt, adresarea IP determină comportamentul end-to-end al unui pachet IP.

Însă, de-a lungul fiecărei legături dintr-o cale, un pachet IP este încapsulat într-un frame specific pentru o tehnologie particulară de legătură de date asociată acelei legături, cum ar fi Ethernet.

În rețelele Ethernet, adresele MAC sunt utilizate pentru a identifica, la un nivel inferior, sursa și destinația. Atunci când un host dintr-o rețea Ethernet comunică, el trimită frameurile ce conțin adresa MAC proprie ca sursă și adresa MAC a destinatarului ca destinație. Toate hosturile ce primesc frameul citesc adresa MAC destinație. Dacă adresa MAC destinație corespunde adresei MAC configurată pe host, atunci hostul va procesa mesajul.

Fig. xxx 1 arată modul în care un pachet de date, conținând informații de adresă IP, este încapsulat cu frameul de la nivelul legătură de date ce conține informații de adresă MAC.

Fig.5.19 arată modul în care frameurile sunt încapsulate, bazându-se pe tehnologia legăturii respective.

Cum sunt asociate adresele IP ale pachetelor IP dintr-un flux de date cu adresele MAC din fiecare legătură de-a lungul căii spre destinație?

Acest lucru se realizează printr-un proces numit Address Resolution Protocol (ARP).

Destination MAC Address BB:BB:BB:BB:BB:BB	Source MAC Address AA:AA:AA:AA:AA:AA	Source IP Address 10.0.0.1	Destination IP Address 192.168.1.5	Data	Trailer
Fig. 5.19. A					

A switch examines MAC addresses.

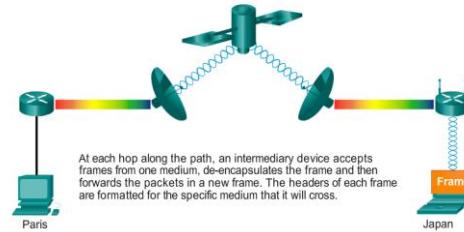
Destination MAC Address BB:BB:BB:BB:BB:BB	Source MAC Address AA:AA:AA:AA:AA:AA	Source IP Address 10.0.0.1	Destination IP Address 192.168.1.5	Data	Trailer
Fig. 5.19. B					

A router examines IP addresses.

The Data Link Layer
Fig. 5.19. C

Data link layer protocols govern how to format a frame for use on different media.

Different protocols may be in use for different media.



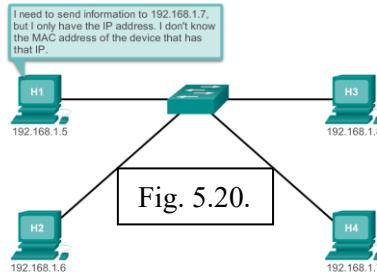
5.5 Address Resolution Protocol – ARP

Amintim faptul că fiecare nod dintr-o rețea IP are o adresă MAC și o adresă IP. Pentru a transmite datele, nodul trebuie să utilizeze ambele adrese. Nodul trebuie să utilizeze propriile adrese MAC și IP în câmpurile de sursă să ofere ambele adrese MAC și IP pentru destinație. În timp ce adresa IP destinație va fi oferită de către nivelul OSI superior, nodul expeditor necesită să găsească adresa MAC destinație pentru respectiva legătură Ethernet. Acesta este scopul ARP.

ARP se bazează pe anumite tipuri de mesaje broadcast Ethernet și unicast Ethernet, numite ARP requests și ARP replies.

Protocolul ARP oferă două funcții de bază:

- Rezolvarea adreselor IPv4 în adresarea MAC.
- Menținerea unui tabel de mapare.



5.5.1 Rezolvarea adreselor IPv4 în adresarea MAC

Pentru ca un frame să fie plasat pe mediul de comunicație al unui LAN, trebuie să aibă adresa MAC destinație. Atunci când un pachet este trimis la nivelul legătură de date pentru a fi încapsulat într-un frame, nodul face referire la un tabel din memoria proprie pentru a afla adresa de la nivelul legătură de date ce este mapată la adresa IPv4 destinație. Tabelul este numit tabel ARP sau ARP cache. Tabelul ARP este stocat în memoria RAM a dispozitivului.

Fiecare rând al tabelului ARP leagă o adresă IP de o adresă MAC. Numim această relație dintre cele două valori o mapare – înseamnă că putem localiza o adresă IP din tabel și descoperi adresa MAC corespunzătoare. Tabelul ARP salvează temporar maparea pentru dispozitivele din LANul local.

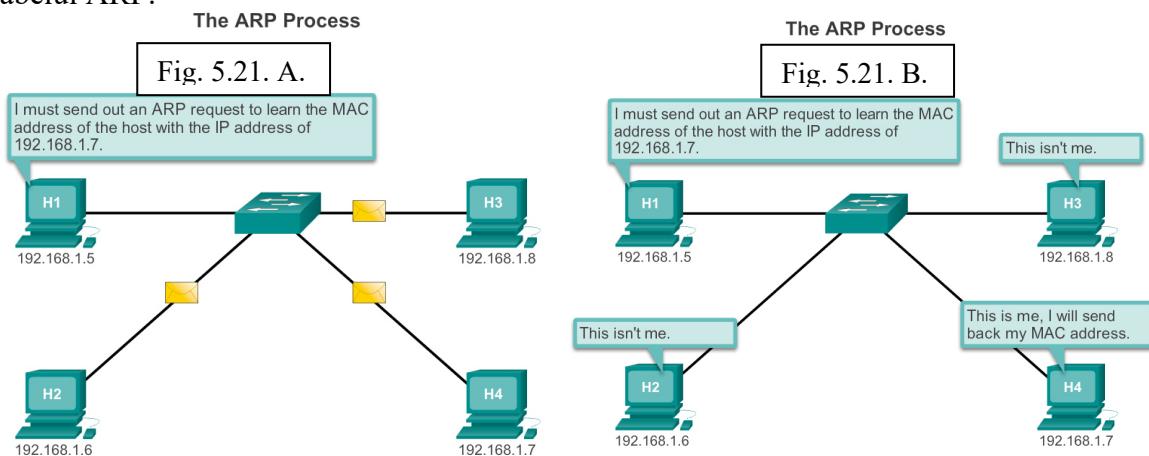
Pentru a începe procesul, un nod de transmisie încearcă să localizeze adresa MAC mapată la o destinație IPv4. Dacă această mapare este găsită în tabel, nodul utilizează adresa MAC ca destinație în frame și încapsulează pachetul IPv4. Frameul este apoi codat în mediul de rețea.

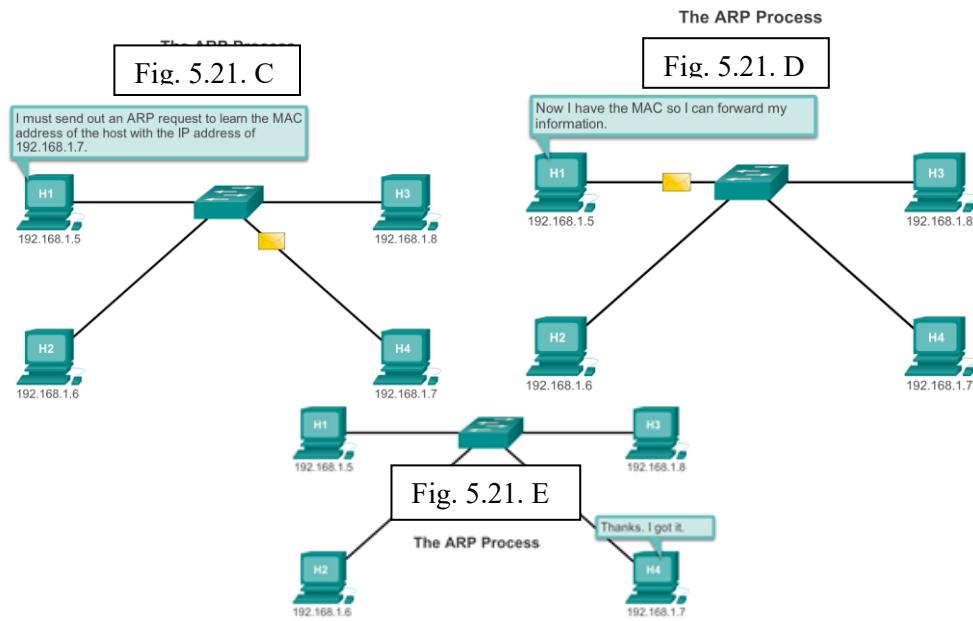
5.5.2 Menținerea tabelaiei ARP

Tabelul ARP este menținut în mod dinamic. Există două moduri în care un dispozitiv poate avea adresele MAC. Un mod este de a monitoriza traficul ce are loc pe segmentul de rețea locală. Atunci când un nod primește frameuri din mediul, poate înregistra adresele sursă IP și MAC ca o mapare în tabelul ARP. Atunci când frameurile sunt transmise în rețea, dispozitivul populează tabelul ARP cu perechi de adrese.

Un alt mod de primire a unei perechi de adrese este transmiterea unui ARP request, așa cum este prezentat în Fig. xxx. ARP request este un broadcast de nivel 2 către toate dispozitivele din LANul Ethernet. ARP request conține adresa IP a hostului destinație și adresa MAC broadcast, FFFF.FFFF.FFFF. Din moment ce este un broadcast, toate nodurile din LANul Ethernet îl vor primi și se vor uita la conținutul său. Nodul a cărui adresă IP corespunde cu adresa IP din ARP request va da un răspuns. Răspunsul va fi un frame unicast ce include adresa MAC ce corespunde adresei IP din cerere. Acest răspuns este apoi utilizat pentru a introduce un nou rând în tabela ARP al nodului sursă.

Intrările din tabelul ARP sunt marcate temporar în același mod în care intrările din tabelul MAC sunt marcate temporar în switchuri. Dacă un dispozitiv nu primește un frame de la un anumit dispozitiv înainte ca marca de timp să expire, intrarea pentru acest dispozitiv este ștearsă din tabelul ARP.





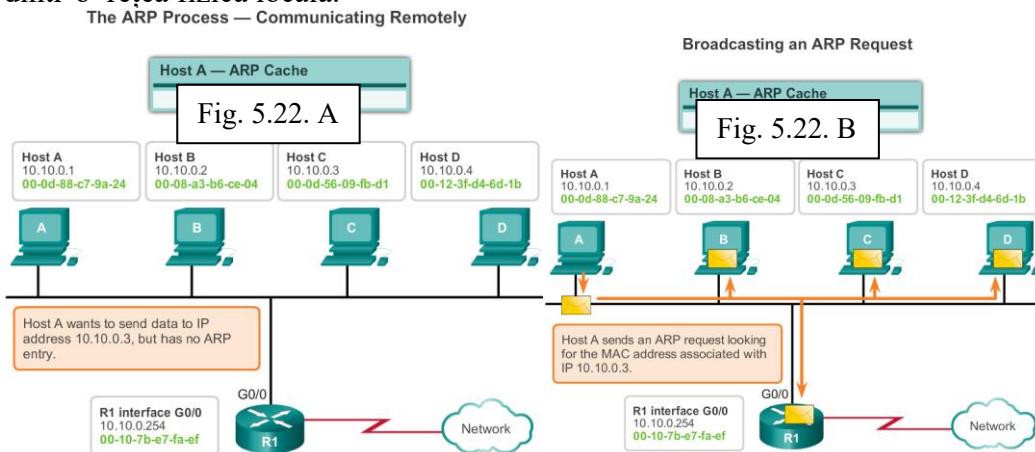
5.5.3 Crearea frame-ului

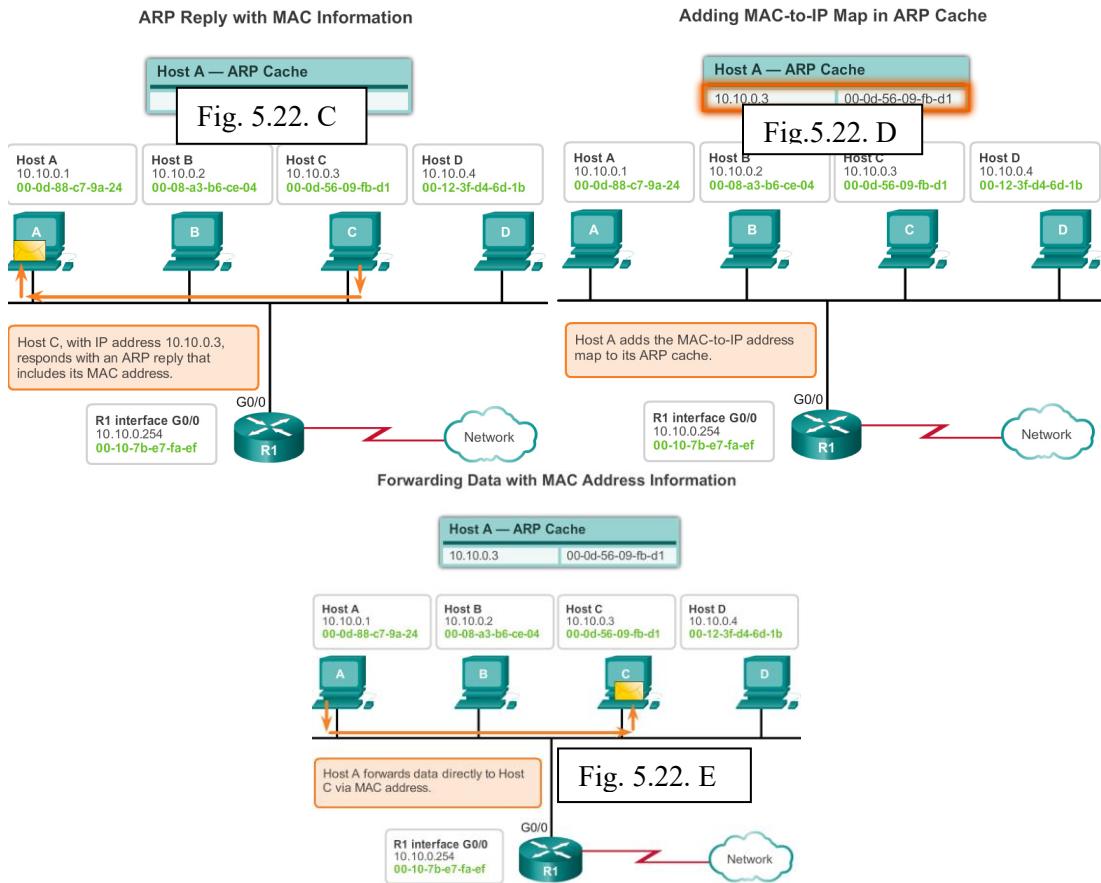
Ce face un nod atunci când necesită să creeze un frame, iar tabelul ARP nu conține o mapare a unei adrese IP la o adresă MAC destinație ? *Generează un ARP request !*

Atunci când ARP primește o cerere de mapare a unei adrese IPv4 într-o adresă MAC, caută maparea din tabelul ARP propriu. Dacă nu este găsită nici-o intrare, încapsularea unui pachet IP eșuează și procesele de nivel 2 notifică ARP că necesită o mapare. Procesele ARP apoi trimit un ARP request pentru a descoperi adresa MAC a dispozitivului destinație din rețea locală. Dacă un dispozitiv ce primește cererea are adresa IP destinație, răspunde cu un ARP reply. O mapare este creată în tabelul ARP. Pachetele pentru adresa IPv4 respectivă pot fi acum încapsulate în frameuri.

Dacă nici-un dispozitiv nu răspunde la ARP request, pachetul este aruncat deoarece un frame nu poate fi creat. Acest eșec de încapsulare este raportat la nivelele superioare ale dispozitivului. Dacă dispozitivul este un dispozitiv intermediar, cum ar fi un router, nivelele superioare ar putea alege să răspundă hostului sursă cu o eroare într-un pachet ICMPv4.

Prin vizualizarea Figurilor 1-5 putem observa procesul utilizat pentru a găsi o adresă MAC a nodului dintr-o rețea fizică locală.





5.5.4 ARP

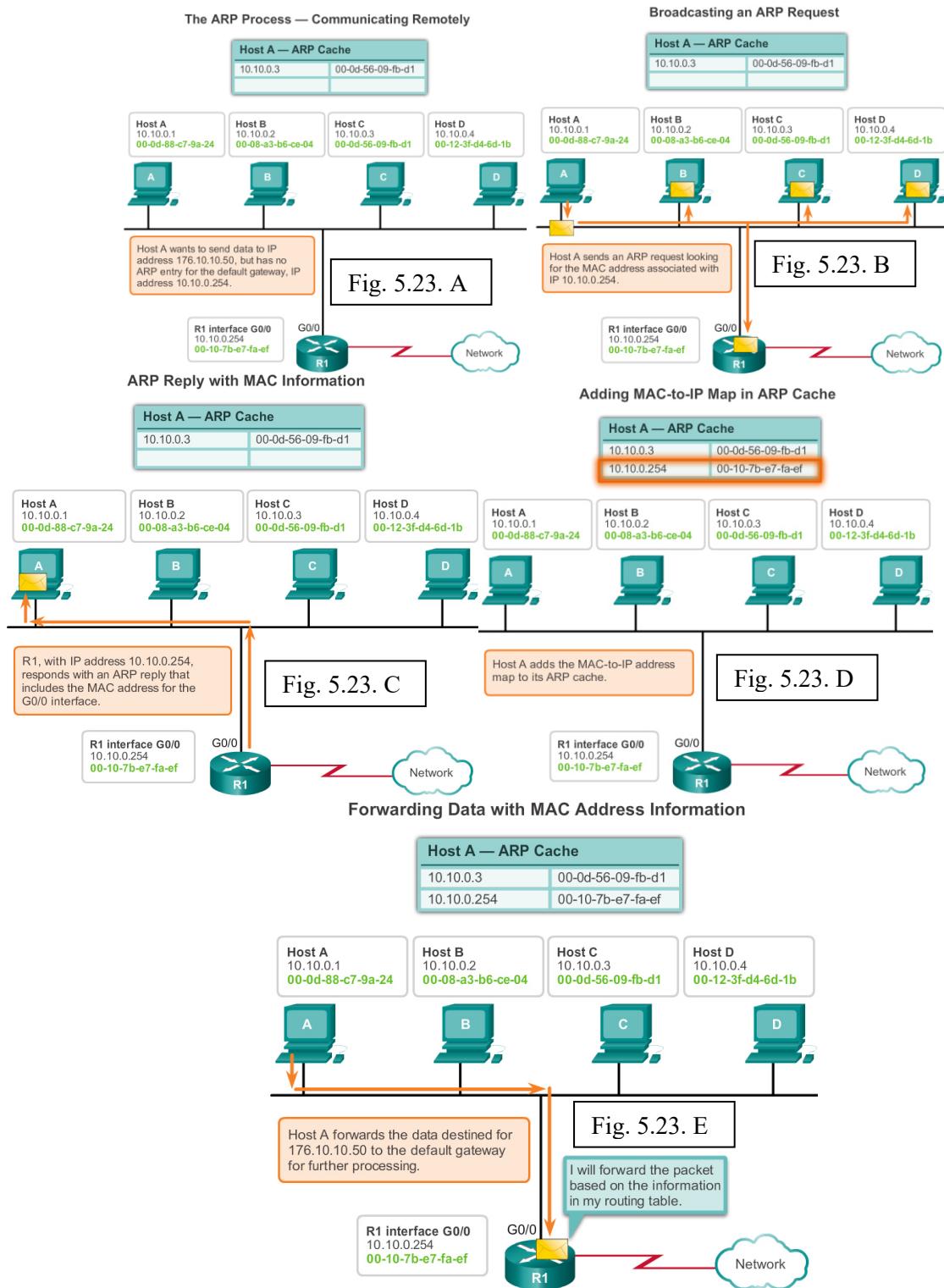
Toate frameurile trebuie să fie livrate la un nod dintr-o segment de rețea locală. Dacă hostul destinație IPv4 se află în rețeaua locală, frameul va utiliza adresa MAC a propriului dispozitiv ca adresă MAC destinație.

Dacă hostul destinație IPv4 nu se află în rețeaua locală, nodul sursă necesită să transmită frameul interfeței routerului care este “gateway” sau următorului hop utilizat pentru a atinge destinația respectivă. Nodul sursă va utiliza adresa MAC a gatewayului ca adresă destinație pentru frameurile ce conțin un pachet IPv4 adresat la hosturile din alte rețele.

Adresa de gateway a interfeței routerului este stocată în configurația IPv4 a hosturilor. Atunci când un host crează un pachet pentru o destinație, compară adresa IP destinație și adresa IP proprie pentru a determina dacă cele două adrese IP sunt localizate în aceeași rețea de nivel 3. Dacă hostul destinatar nu se află în aceeași rețea, sursa utilizează procesul ARP pentru a determina o adresă MAC pentru a insera adresa MAC a routerului ce servește ca gateway.

În cazul în care adresa MAC de gateway nu este în tabel, procesul ARP normal va trimite un ARP request pentru a primi adresa MAC asociată cu adresa IP a interfeței routerului.

Prin vizualizarea Figurilor 1-5 putem observa procesul utilizat pentru obținerea adresei MAC a gateway.

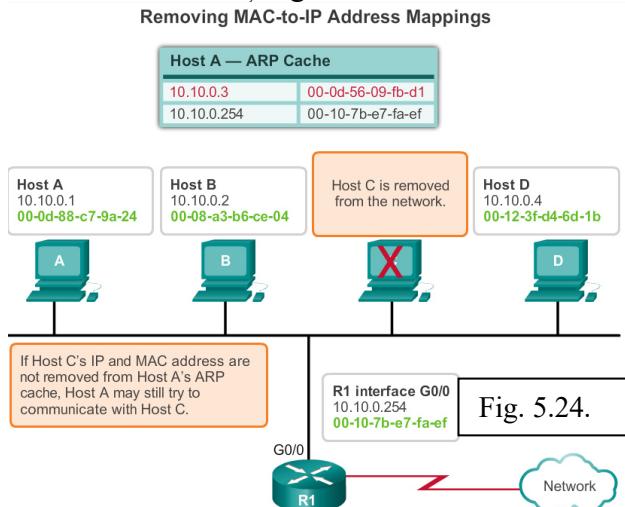


Pentru fiecare dispozitiv, un ARP cache timer sterge intrările ARP ce nu au fost utilizate de o perioadă precizată de timp. Perioadele diferă în funcție de dispozitiv și sistemul de operare. De exemplu, unele sisteme de operare Windows stochează intrările din tabelul ARP pentru două minute. Dacă o intrare este utilizată în această perioadă de timp, ARP timer pentru intrarea respectivă este extins la 10 minute.

Comenziile pot fi de asemenea utilizate pentru a șterge manual toate sau unele intrări din tabelul ARP. După ce o intrare a fost ștersă, procesul pentru trimitera unui ARP request și primirea unui ARP reply trebuie să aibă loc din nou pentru a introduce maparea în tabelul ARP.

Fiecare dispozitiv are o comandă specifică sistemului de operare pentru a șterge conținutul tabelului ARP. Aceste comenzi nu invocă execuția ARP în nici-un mod. Ele elimină doar intrările tabelului ARP. Serviciul ARP este integrat în protocolul IPv4 și implementat de către dispozitiv. Funcționarea să este transparentă utilizatorilor și aplicațiilor de nivel superior.

Ca și în Fig. 5.24, este uneori necesară ștergerea unei intrări din tabelul ARP.



Spre exemplu pe un router Cisco, comanda **show ip arp** este utilizată pentru a afișa tabelul ARP, conform Fig. 5.25.

Router#show ip arp					
Protocol	Address	Age (min)	Hardware Addr	Type	Interface
Internet	172.16.233.229	-	0000.0c59.f892	ARPA	Ethernet0/0
Internet	172.16.233.218	-	0000.0c07.ac00	ARPA	Ethernet0/0
Internet	172.16.168.11	-	0000.0c63.1300	ARPA	Ethernet0/0
Internet	172.16.168.254	9	0000.0c36.6965	ARPA	Ethernet0/0

Fig. 5.25. Tabela ARP

Pe un PC cu Windows 7, comanda **arp -a** este utilizată pentru a afișa tabelul ARP, conform Fig. 5.26.

C:\>arp -a			
Fig. 5.26. Tabela ARP pe HOST			
Interface: 192.168.1.67 --- 0xa			
Internet Address	Physical Address	Type	
192.168.1.254	64-0f-29-0d-36-91	dynamic	
192.168.1.255	ff-ff-ff-ff-ff-ff	static	
224.0.0.22	01-00-5e-00-00-16	static	
224.0.0.251	01-00-5e-00-00-fb	static	
224.0.0.252	01-00-5e-00-00-fc	static	
255.255.255.255	ff-ff-ff-ff-ff-ff	static	
Interface: 10.82.253.91 --- 0x10			
Internet Address	Physical Address	Type	
10.82.253.92	64-0f-29-0d-36-91	dynamic	
224.0.0.22	01-00-5e-00-00-16	static	
224.0.0.251	01-00-5e-00-00-fb	static	
224.0.0.252	01-00-5e-00-00-fc	static	
255.255.255.255	ff-ff-ff-ff-ff-ff	static	

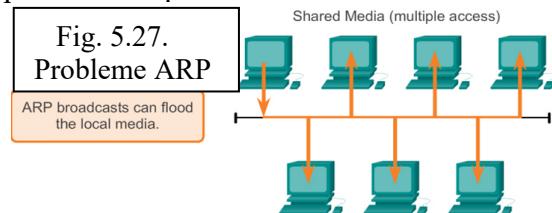
5.6.4.1 Probleme ARP

Fig. 5.27 ilustrează două probleme potențiale ale ARP.

➤ **Overhead pe mediu** – Ca orice frame broadcast, un ARP request este primit și procesat de către fiecare dispozitiv din rețea locală. Într-o rețea tipică de afacere, aceste broadcast ar avea probabil impact minim asupra performanței rețelei. Însă, dacă un număr mare de dispozitive urmează să fie alimentate și toate încep să accesze serviciile de rețea în același timp, ar putea exista o anumită reducere în performanță pentru o perioadă scurtă de timp. De exemplu, dacă toți studenții dintr-un laborator se conectează la computerele din sală și încercă să accesze internetul în același timp, pot apărea întârzieri. Însă, după ce dispozitivele transmit broadcasturile ARP inițiale îi învață adresele MAC necesare, orice impact asupra rețelei va fi minimizat.

➤ **Securitatea** – În unele cazuri, utilizarea ARP poate duce la un risc potențial de securitate. ARP spoofing sau ARP poisoning este o tehnică utilizată de orice atacator pentru a introduce asocierea greșită de adrese MAC într-o rețea prin emiterea de ARP request fals. Un atacator falsifică adresa MAC a unui dispozitiv și apoi frameurile pot fi transmise la o destinație greșită.

ConFig.rea manuală de asociere ARP statice este un mod de prevenire a ARP spoofing. Adresele Mac autorizate pot fi configurate pe unele dispozitive de rețea pentru a restricționa accesul de rețea numai la respectivele dispozitive listate.



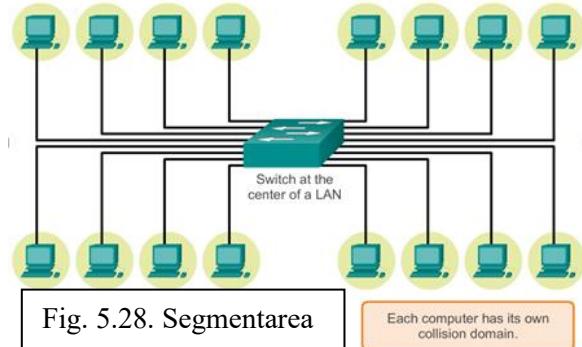
Problemele ARP:

- Securitatea*
- Broadcast și overhead peste mediile de comunicație*
- Tabela de mesaje ARP poate furniza o adresă MAC incorectă, care atunci poate permite atacul prin folosirea acesteia, atac cunoscut sub denumirea de spoof.*

Problemele de securitate și broadcast legate de ARP pot fi combătute cu switchuri moderne. Spre exemplu, switchurile Cisco suportă mai multe tehnologii de securitate dezvoltate în special pentru reducerea problemelor Ethernet legate de broadcast, în general și de ARP, în particular.

Switchurile oferă segmentarea unui LAN, divizând LANul în mai multe domenii de coliziune. Fiecare port dintr-un switch reprezintă un domeniu de coliziune separat și oferă full media bandwidth pentru nodul sau nodurile conectate la respectivul port. Deși switchurile nu previn în mod explicit broadcasturile să se propage la dispozitivele conectate, izolează comunicațiile Ethernet unicast astfel încât ele să fie "auzite" doar de dispozitivele sursă și destinație. Deci, dacă există un număr mare de ARP requests, fiecare ARP reply va fi numai între două dispozitive.

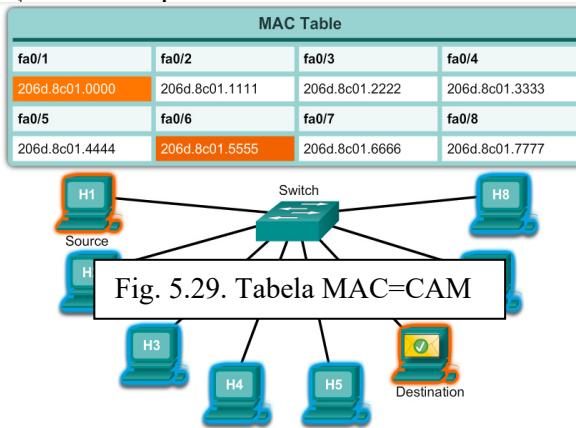
În ceea ce privește reducerea tipurilor diferite de atacuri broadcast, la care rețelele Ethernet sunt predispuse, inginerii de rețea implementează tehnologii de securitate pe Switchurile Cisco, cum ar fi liste de control al accesului de specialitate și securizarea portului.



5.5.5 LAN bazate pe Switchuri

Reamintim faptul că topologia logică a unei rețele Ethernet este o magistrală multi-access în care dispozitivele împart accesul la același mediu de comunicație. Această topologie logică determină modul în care hosturile dintr-o rețea văd și procesează frameurile trimise și primite din rețea. Însă, topologia fizică a multor rețele Ethernet de astăzi este star sau extended-star. Acest lucru înseamnă că pe cele mai multe rețele Ethernet, dispozitivele finale sunt în mod normal conectate, într-o legătură point-to-point, la un switch de nivel 2.

Un switch de nivel 2 efectuează comutare și filtrare bazându-se numai pe adresele MAC de nivel 2 (nivelul legătură de date OSI). Un switch este complet transparent protocolelor de rețea și aplicațiilor de utilizator. Un switch de nivel 2 construiește un tabel de adrese MAC ce este utilizat pentru a lua decizii de transmitere. Switchurile de nivel 2 depind de routere pentru a transmite datele între subrețele IP independente.



Switchurile utilizează adresele MAC pentru a direcționa comunicațiile de rețea prin intermediul **"switch fabric"** la portul corespunzător spre nodul destinație. Switch fabric, constă din circuitele integrate și programarea transporturilor de însoțire, ce permite să fie controlate căile de date prin intermediul switchului. Pentru ca un switch să știe ce port să folosească pentru a transmite un frame unicăst, trebuie mai întâi să învețe ce noduri există la fiecare dintre porturile sale.

Un switch determină modul de tratare a frameurilor de date prin utilizarea tabelei de adrese MAC. Un switch își construiește propria tabelă de adrese MAC prin înregistrarea adreselor MAC ale nodurilor conectate la fiecare port al său. O dată ce o adresă MAC pentru un nod specific de la un port specific este înregistrată în tabelul de adrese, switchul știe să transmită traficul destinat pentru nodul respectiv prin portul mapat nodului pentru transmisii ulterioare.

Atunci când este primit un frame de date de către un switch și adresa MAC destinație nu este în tabel, switchul transmite frameul pe toate porturile, exceptie făcând portul prin care a fost

primit. Atunci când nodul destinație răspunde, switchul înregistrează adresa MAC a nodului în tabelul de adrese de la câmpul de adresă sursă a frameului. În rețelele cu mai multe switchuri interconectate, tabelele de adrese MAC înregistrează mai multe adrese MAC pentru porturile ce conectează switchurile ce reflectă dincolo de nod. În mod normal, porturile switchurilor utilizate pentru a interconecta două switchuri au mai mulți adrese MAC înregistrate în tabelul de adrese MAC.

Următorul algoritm descrie acest proces astfel :

Pasul 1. Switchul primește un frame broadcast de la PC1 pe portul 1.

Pasul 2. Switchul introduce adresa MAC sursă și portul de switch ce primește frameul în tabelul de adrese.

Pasul 3. Deoarece adresa destinație este un broadcast, switchul transmite frameul la toate porturile, în afara celui de la care a primit frameul.

Pasul 4. Dispozitivul destinație răspunde la broadcast cu un frame unicast adresat lui PC1.

Pasul 5. Switchul introduce adresa MAC sursă a PC2 și numărul de port al portului de switch care a primit frameul în tabela de adrese. Adresa destinație a frameului și portul asociat sunt înregistrate în tabelul de adrese MAC.

Pasul 6. Switchul poate să transmită acum frameuri între dispozitivele sursă și destinație fără flooding, deoarece are intrări în tabelul de adrese ce identifică porturile asociate.

Notă: Tabelul de adrese MAC se mai numește și **"Content Addressable Memory"** tabela (CAM). Deși termenul CAM este destul de cunoscut, pentru scopul acestui curs, ne vom referi la tabel ca tabelul de adrese MAC.



Fig. 5.30. Învățarea adresei MAC-P1.

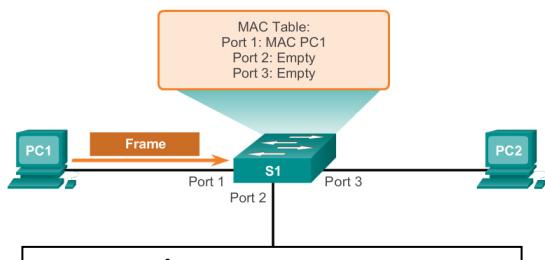


Fig. 5.30. Învățarea adresei MAC-P2.

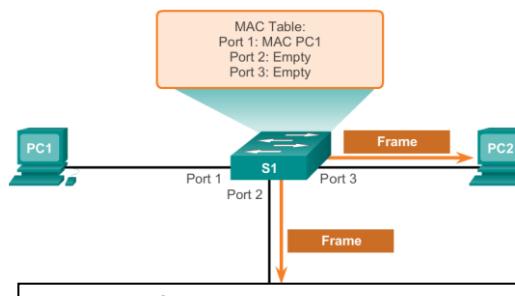


Fig. 5.30. Învățarea adresei MAC-P3.

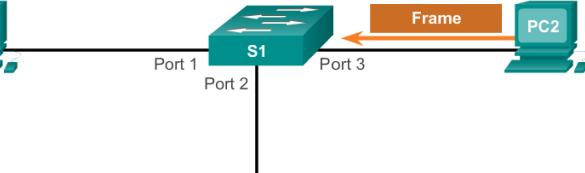


Fig. 5.30. Învățarea adresei MAC-P4.

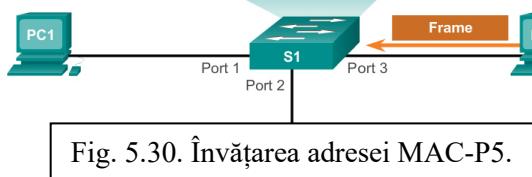


Fig. 5.30. Învățarea adresei MAC-P5.

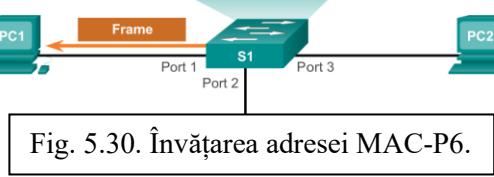


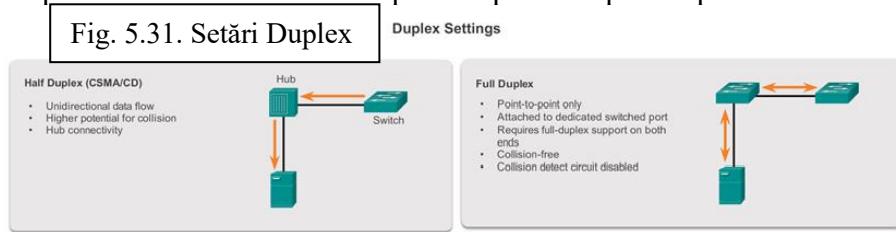
Fig. 5.30. Învățarea adresei MAC-P6.

Deși transparent pentru protocolele de rețea și aplicațiile de utilizator, switchurile pot funcționa în mai multe moduri, care pot avea efecte atât pozitive, cât și negative, atunci când se transmit frameuri Ethernet într-o rețea. Una dintre setările de bază ale switchului este setarea duplex pentru fiecare port individual conectat la fiecare dispozitiv de rețea. Un port de pe switch trebuie să fie config. t pentru a îndeplini setările de duplex ale tipului de mediu. Există două tipuri de setări duplex pentru comunicațiile dintr-o rețea Ethernet; half-duplex și full-duplex.

Half Duplex – Comunicarea half duplex se bazează pe un flux de date unidirecțional, unde datele transmise și primite nu au loc în același timp. Acest lucru este similar cu modul în care walkie-talkies și radio cu două direcții funcționează astfel încât o singură persoană poate vorbi la un moment dat. Dacă cineva vorbește în timp ce o altă persoană vorbește, are loc o coliziune. Ca rezultat, comunicarea half-duplex implementează CSMA/CD pentru a ajuta reducerea potențialului de coliziuni și pentru detectarea în cazul în care are loc. Comunicațiile half-duplex au probleme de performanță cu privire la timpul constant de așteptare, deoarece datele pot fi transmise într-o singură direcție la un moment dat. Conexiunile half-duplex sunt întâlnite de obicei în hardwareul vechi, precum huburi. Nodurile atașate huburilor ce împart conexiunea la un port de switch trebuie să funcționeze în mod half-duplex deoarece dispozitivele finale trebuie să fie capabile să detecteze coliziunile. Nodurile pot funcționa într-un mod half-duplex dacă NIC nu poate fi config. t pentru operații full duplex. În acest caz, portul de la switch are de asemenea modul half-duplex. Datorită acestor limitări, comunicația full duplex a înlocuit half duplex în hardwareul actual.

Full Duplex – În comunicația full duplex, fluxul de date este bidirecțional, deci datele pot fi transmise și primite în același timp. Suportul bidirecțional îmbunătățește performanța prin reducerea timpului de așteptare dintre transmisii. Mai multe tehnologii - Ethernet, Fast Ethernet și Gigabit Ethernet în care funcționează NICs vândute astăzi oferă capacitate full duplex. În modul full duplex, circuitul de detecție de coliziuni este dezinstalat. Frameurile transmise de către două noduri conectate nu pot să producă coliziune deoarece nodurile finale utilizează două circuite separate în cablul de rețea. Fiecare conexiune full duplex utilizează un singur port. Conexiunile full duplex necesită un switch ce suportă full duplex sau o conexiune directă între două noduri, fiecare suportând full duplex. Nodurile conectate direct la un port de switch dedicat cu NICs ce suportă full duplex ar trebui să fie conectate la porturile de switch ce sunt config. te să funcționeze în modul full-duplex.

Fig. 5.31 prezintă cele două setări duplex disponibile pe echipamentul modern de rețea.



Un Cisco Catalyst switch suportă trei setări duplex:

- *Opțiunea full setează modul full-duplex.*
- *Opțiunea half setează modul half-duplex.*
- *Opțiunea auto setează autonegotierea modului duplex. Cu această opțiune activată, două porturi comunică pentru a decide modul cel mai bun de funcționare.*

Pentru porturile Fast Ethernet și 10/100/1000, opțiunea implicită este cea auto. Pentru porturile 100BASE-FX, cea full. Porturile 10/100/1000 funcționează fie în modul half-duplex, fie

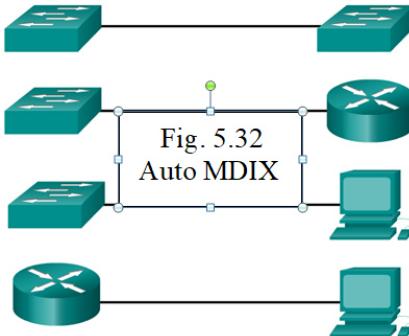
full-duplex atunci când sunt setate la 10 sau 100 Mb/s, însă atunci când sunt setate la 1,000 Mb/s, funcționează numai în modul full-duplex.

Pentru a avea setarea corectă duplex, este necesar tipul de cablu corect definit pentru fiecare port. Conexiunile dintre dispozitive specifice, precum switch la switch, switch la router, switch la host și router la host necesită utilizarea unor tipuri de cabluri specifice (crossover sau straight-through). În schimb, mai multe dispozitive switch suportă acum comanda de config.re de interfață mdix auto în CLI pentru a permite caracteristica automatic medium-dependent interface crossover (auto-MDIX).

Atunci când caracteristica auto-MDIX este activată, switchul detectează tipul de cablu necesar pentru conexiunile de cupru Ethernet și configurează interfețele corespunzătoare acestora. Prin urmare, pot fi folosite atât cabluri crossover, cât și straight-through pentru conexiuni la un port de cupru 10/100/1000 de pe un switch, indiferent de tipul de dispozitiv ce se află la celălalt capăt al conexiunii.

Caracteristica auto-MDIX este activată implicit pe switchurile ce rulează Cisco IOS Release 12.2(18)SE, sau IOS mai nou. Pentru versiunile dintre Cisco IOS Release 12.1(14)EA1 și 12.2(18)SE, caracteristica auto-MDIX este dezactivată implicit.

MDIX auto detects the type of connection required and configures the interface accordingly.

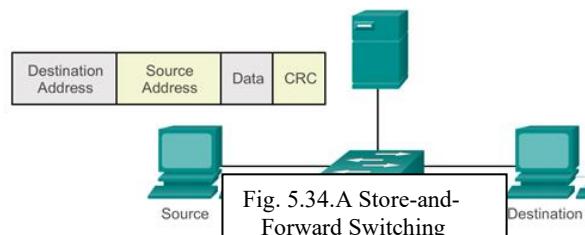
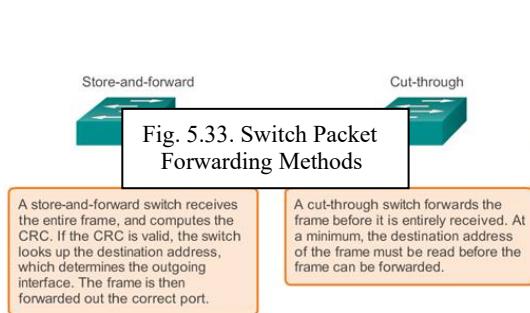


În trecut, switchurile utilizau una dintre următoarele metode de transmitere pentru comutarea de date dintre porturile de rețea:

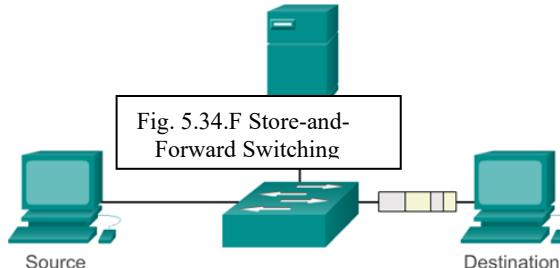
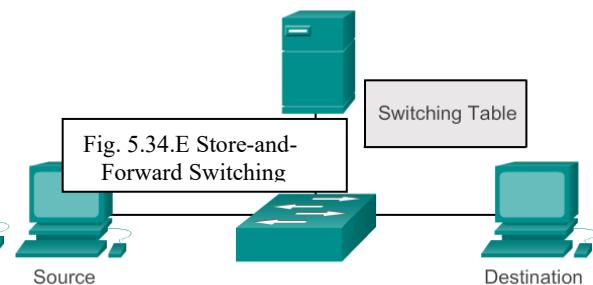
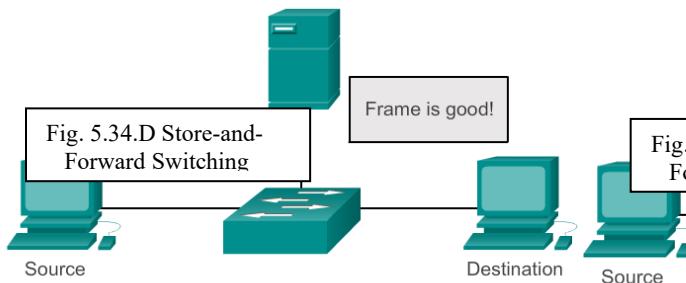
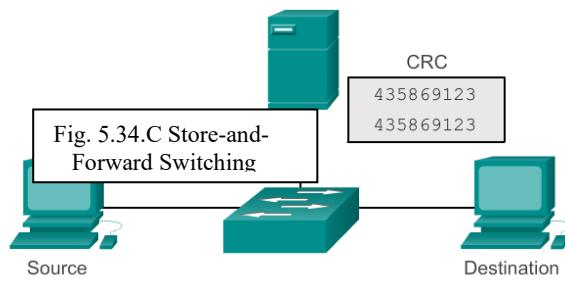
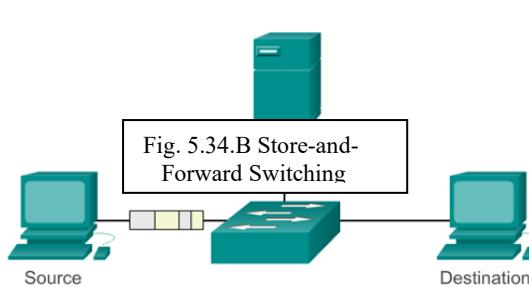
- *Store-and-forward switching.*
- *Cut-through switching.*

În store-and-forward switching, atunci când switchul primește un frame, stochează datele în buffere până când frameul complet a fost primit. În timpul procesului de stocare, switchul analizează frameul pentru informații despre destinația să. În acest proces, switchul efectuează de asemenea și o verificare de eroare utilizând partea de trailer de Cyclic Redundancy Check (CRC) a frameului Ethernet.

CRC folosește o formulă matematică, bazată pe numărul de 1 din frame, pentru a determina dacă frameul primit are o eroare. După confirmarea integrității frameului, frameul este transmis pe portul adecvat spre destinație. Atunci când o eroare este detectată în frame, switchul aruncă frameul. Aruncarea frameului cu erori reduce cantitatea de lățime de bandă consumată de datele corupte. Comutare Store-and-forward este necesar pentru analiza Quality of Service (QoS) pe rețelele convergente în care este necesară clasificarea frameurilor pentru prioritizarea traficului. De exemplu, voce prin fluxuri de date IP trebuie să aibă prioritate față de traficul web de navigare.



A store-and-forward switch receives the entire frame, and computes the CRC. If the CRC is valid, the switch looks up the destination address, which determines the outgoing interface. The frame is then forwarded out the correct port.



În metoda de comutare cut-through, switchul se ocupă de date o dată ce sunt primite, chiar dacă transmisia nu este completă. Switchul stochează atât cât este necesar din frame pentru a citi adresa MAC destinație astfel încât să determine pe care port să transmită datele. Adresa MAC destinație este localizată în primii 6 bytes din frame, ulterior preambulului. Switchul caută adresa MAC destinație în tabelul său, determină portul corespunzător și transmite frameul spre destinație prin intermediul portului respectiv. Switchul nu efectuează nici-o detectie de eroare asupra frameului. Datorită faptului că switchul nu trebuie să aștepte primirea să completă și deoarece nu efectuează nici-o detectie de eroare, cut-through switching este mai rapid decât store-and-forward switching. Însă, deoarece switchul nu efectuează nici-o verificare de eroare, transmite frameuri alterate prin rețea. Frameurile alterate consumă lățime de bandă în timp ce sunt transmise. NICul destinație aruncă frameurile alterate.

Există două variante de cut-through switching:

- **Fast-forward switching:** Fast-forward switching oferă cel mai scăzut nivel de latență. Fast-forward switching transmite pe loc un pachet după ce îi citește adresa destinație.

Deoarece fast-forward switching începe transmiterea înainte ca pachetul să fie primit, pot există momente când pachetele sunt transmise cu erori. Acest lucru se întâmplă rar, iar adaptorul de rețea destinație aruncă pachetul alterat după primire. În modul fast-forward, latența este măsurată de la primul bit primit la primul bit transmis. Fast-forward switching este metoda cut-through tipică de switching.

- **Fragment-free switching:** În fragment-free switching, switchul stochează primii 64 bytes ai frameului înainte de a-l transmite mai departe. Fragment-free switching poate fi văzută ca un compromis dintre store-and-forward switching și fast-forward switching. Motivul pentru care fragment-free switching stochează numai primii 64 de bytes din frame este acela că cele mai multe erori de rețea și coliziunii au loc în primii 64 de bytes. Fragment-free switching încearcă să îmbunătățească fast-forward switching prin efectuarea unei verificări de erori în primii 64 bytes ai frameului pentru a se asigura de faptul că o coliziune nu a avut loc înaintea transmiterii frameului. Fragment-free switching este un compromis dintre latența mare și integritatea ridicată a store-and-forward switching, și latența scăzută și integritatea redusă a fast-forward switching.

Fig. 5.35 prezintă un exemplu de cut-through switching.

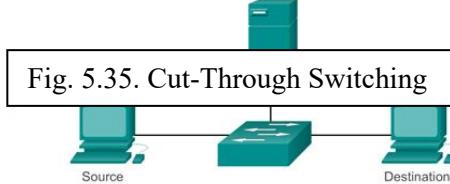


Fig. 5.35. Cut-Through Switching

A cut-through switch forwards the frame before it is entirely received. At a minimum, the destination address of the frame must be read before the frame can be forwarded.

Unele switchuri sunt configurate pentru a efectua cut-through switching pe bază per-port până când un prag de eroare definit de utilizator este atins și apoi se schimbă automat în store-and-forward. În cazul în care rata de eroare scade sub prag, portul se schimbă automat înapoi în cut-through switching.

Un switch analizează unele sau toate informațiile dintr-un pachet înainte să îl trimită la hostul destinație. Un switch Ethernet ar putea utiliza o tehnică de buffering pentru a stoca frameurile înaintea transmiterii lor. Buffering ar putea de asemenea să fie folosit atunci când portul destinație este ocupat, iar switchul stochează frameul până când poate fi transmis.

Așa cum este prezentat și în Fig. 5.36, există două metode de buffering de memorie: bazată pe port și memorie partajată.

Fig. 5.36. Port-Based and Shared Memory Buffering

Port-based memory	In port-based memory buffering, frames are stored in queues that are linked to specific incoming and outgoing ports.
Shared memory	Shared memory buffering deposits all frames into a common memory buffer, which all the ports on the switch share.

Buffering de memorie bazat pe port – În port-based memory buffering, frameurile sunt stocate în cozi ce sunt legate de porturi specifice de primire sau transmitere. Un frame este transmis la portul de ieșire atunci când frameurile precedente lui la coada au fost transmise cu succes. Este posibil ca un singur frame să întârzie transmisia tuturor frameurilor din memorie datorită unui port destinație ocupat. Această întârziere are loc chiar dacă celelalte frameuri pot fi transmise la porturi destinație libere.

Buffering de memorie partajat – Shared memory buffering stochează toate frameurile într-un buffer comun de memorie pe care toate porturile din switch îl împart. Cantitatea de memorie buffer necesară pentru un port este alocată dinamic. Frameurile din buffer sunt legate dinamic la portul destinație. Acest lucru permite pachetului să fie primit pe un port și apoi transmis pe alt port, fără a se muta la o coadă diferită.

Switchul păstrează o hartă de frame a legăturilor de port ce arată unde un pachet necesită să fie transmis. O legătură din hartă este ştearsă după ce frameul a fost transmis cu succes. Numărul de frameuri stocate în buffer este restricționat de dimensiunea întregului buffer de memorie și nu este limitat la un singur buffer de port. Acest lucru permite frameurilor mai mari să fie transmise, cu mai puține frameuri aruncate. Este foarte important acest lucru în metoda asymmetric switching. Asymmetric switching permite mai multe rate de date diferite pe mai multe porturi diferite. Acest lucru permite ca lățime de bandă mai mare să fie dedicată la anumite porturi, cum ar fi un port conectat la un server.

5.6 Fix sau Modular

Atunci când alegem un switch, este importantă înțelegerea caracteristicilor cheie ale opțiunilor de switch disponibile. Acest lucru înseamnă că este necesară decizarea dacă sunt sau nu necesare caracteristici precum Power over Ethernet (PoE) și decizarea cu privire la “rata de expediere” preferată.

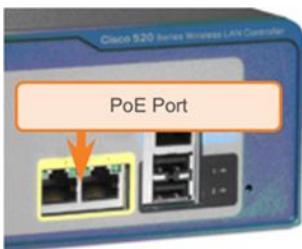
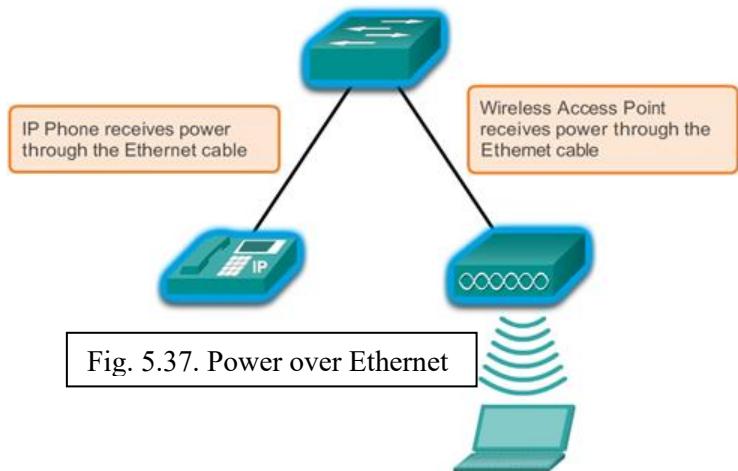
La fel ca și în Fig. 5.37, PoE permite unui switch să livreze putere unui dispozitiv, cum ar fi IP phones sau unele puncte de acces wireless, peste cablul Ethernet existent. Acest lucru permite mai multă flexibilitate instalării.

Forwarding rates definesc capacitatele de procesare ale unui switch prin menținerea cantității de date ce poate fi procesată pe secundă de către switch. Liniile de producție de switch sunt clasificate în funcție de forwarding rates. Switchurile entry-level au forwarding rates mai scăzute decât switchurile enterprise-layer. Alte considerații pot fi dacă dispozitivul este stackable sau non-stackable, precum și grosimea unui switch (exprimată în numărul de unități/rack), densitatea portului sau numărul de porturi disponibile pentru un singur switch. Densitatea portului a unui dispozitiv poate varia în funcție dacă dispozitivul este un dispozitiv cu conFig.ție fixă sau un dispozitiv modular.

Switchuri cu conFig.ție fixă – Switchurile cu conFig.ție fixă sunt, cum este de așteptat, fixe în conFig.ția lor. Acest lucru înseamnă că nu se pot adăuga noi caracteristici sau opțiuni switchului, în afara celor existente. Acest model particular determină caracteristicile și opțiunile disponibile. De exemplu, dacă se achiziționează un 24-port gigabit fixed switch, nu pot fi adăugate porturi suplimentare atunci când este nevoie. Există mai multe opțiuni de conFig.re diferite ce variază în funcție de numărul și tipul de porturi incluse.

Switchuri modulare – Switchurile modulare oferă mai multă flexibilitate în conFig.ția lor. Switchurile modulare, în general, sunt cu șasiuri de dimensiuni diferite ce permit instalarea de numere diferite ale cardurilor de linie modulare. Cardurile de linie conțin porturile. Cardul de linie se potrivește în șasiul switchului la fel cum cardurile de expansiune se potrivesc într-un PC. Cu cât este mai mare șasiul, cu atât mai multe module poate suporta. Așa cum se vede și în Fig. xxx există mai multe dimensiuni de șasiu diferite. Dacă este cumpărat un switch cu un 24-port line card, se pot adăuga cu ușurință un 24/port line card suplimentar, pentru a aduce numărul total de porturi la 48.

Fig. xxx 2 arată exemple de switchuri cu conFig.ție fixă, modulare și stackable.



PoE ports look the same as any switch port. Check the model of the switch to determine if the port supports PoE.



Fig. 5.38. PoE porturi

PoE ports on wireless access point look the same as any switch port. Check the model of the wireless access point to determine if the port supports PoE.

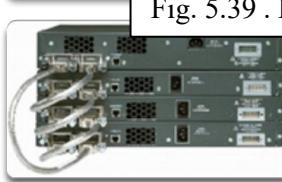
Switch Form Factors



Fixed Configuration Switches
Features and options are limited to those that originally come with the switch.



Modular Configuration Switches
The chassis accepts line cards that contain the ports.



Stackable Configuration Switches
Stackable switches, connected by a special cable, effectively operate as one large switch.

Fig. 5.39 . Modele de Switchuri

Liniile de producție switchuri Cisco sunt dezvoltate global, în mare parte datorită flexibilității oferite de acestea pentru opțiuni add-on. Nu numai că Cisco IOS are cel mai bogat set de caracteristici disponibile în raport cu orice alt sistem de operare de rețea, dar IOS este potrivit pentru fiecare dispozitiv de rețea Cisco, switchurile în particular.

Pentru a evidenția opțiunile disponibile, care sunt prea multe pentru a fi listate, ne axăm pe switchurile Catalyst 3560. Switchurile Catalyst 3560 au porturi Switch Form-Factor Pluggable (SFP) ce suportă un număr de module de emisie SPF. Există o lista de module SPF suportate de către unul sau mai multe tipuri de switchuri 3560:

Fast Ethernet SFP Modules –

- *100BASE-FX (multimode fiber-optic (MMF)) for 2 kilometers (km).*
- *100BASE-LX10 (single-mode fiber-optic (SMF)) for 2km.*
- *100BASE-BX10 (SMF) for 10 km.*
- *100BASE-EX (SMF) for 40 km.*

- *100BASE-ZX (SMF) for 80 km.*

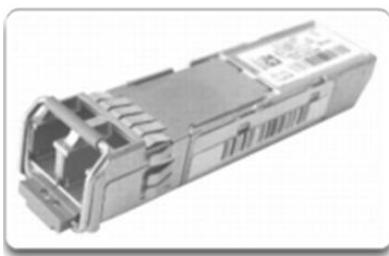
Gigabit Ethernet SFP Modules –

- *1000BASE-SX 50/62.5 µm (MMF) up to 550/220 m.*
- *1000BASE-LX/LH (SMF/MMF) up to 10/0.550 km.*
- *1000BASE-ZX (SMF) up to 70 km.*
- *1000BASE-BX10-D&1000BASE-BX10-U (SMF) up to 10 km.*
- *1000BASE-T (copper wire transceiver).*

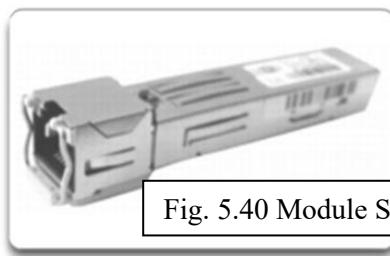
10 Gigabit Ethernet SFP Modules –

- *10G-SR (MMF) up 400 m.*
- *10G-SR-X (MMF) up to 400 m (supporting extended temperature range).*
- *10G-LRM (MMF) up to 220 m.*
- *FET-10G (MMF) up to 100 m (for Nexus fabric uplinks).*
- *10G-LR (SMF) up to 10 km.*
- *10G-LR-X (SMF) up to 10 km (supporting extended temperature range).*
- *10G-ER (SMF) up to 40 km.*
- *10G-ZR (SMF) up to 80 km.*
- *Twinax (copper wire transceiver) up to 10 m.*
- *Active Optical up to 10 m (for intra/inter-rack connections).*

Modulele 40 Gigabit Ethernet și 100 Gigabit Ethernet sunt suportate pe dispozitivele high-end Cisco, cum ar fi Catalyst 6500, CRS router, ASR 9000 series router și Nexus 7000 series switch.



Cisco Optical Gigabit Ethernet SFP



Cisco 1000BASE-T Copper SFP



Cisco 2-channel 1000BASE-BX Optical SFP

Fig. 5.40 Module SFP

5.7 Switchuri de Layer 3

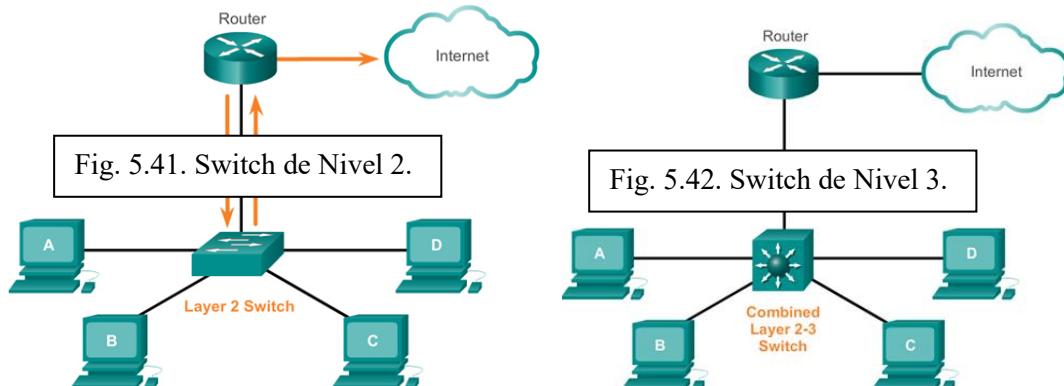
Pentru a determina diferenți factori de formă pentru switchuri, ar putea fi necesară alegerea dintre switchul LAN de nivel 2 și switchul de nivel 3.

Reamintim că switchul LAN de nivel 2 efectuează comutare și filtrare bazându-se numai pe adresele MAC de nivel 2 (nivelul legătură de date OSI) și depinde de routere să transmită datele dintre subrețele IP independente (Fig. xxx 1).

Ca și în Fig. xxx 2, un switch de nivel 3, cum ar fi Catalyst 3560, funcționează similar cu un switch de nivel 2, cum ar fi Catalyst 2960, însă în schimb utilizează numai informații de adresă MAC de nivel 2 pentru deciziile de transmitere, un switch de nivel 3 utilizează de asemenea și informațiile de adresă IP. În schimb să învețe numai ce adrese MAC sunt asociate porturilor sale, un switch de nivel 3 poate învăța de asemenea ce adrese IP sunt asociate interfețelor sale. Acest lucru permite switchului de nivel 3 să direcționeze traficul în rețea bazându-se pe informațiile de adresă IP.

Switchurile de nivel 3 sunt de asemenea capabile să efectueze funcții de rutare de nivel 3, reducând necesitatea routerelor dedicate dintr-un LAN. Deoarece switchurile de nivel 3 au

hardware specializat de switching, ele pot de obicei ruta datele la fel de rapid precum efectuează switchul.



Dispozitivele Cisco ce suportă Layer 3 switching utilizează Cisco Express Forwarding (CEF). Această metodă de transmitere este destul de complexă, însă din fericire, ca orice tehnologie bună, se efectuează în mare parte “în spate=background”. În mod normal, foarte puțină configurație CEF este necesară pe un dispozitiv Cisco.

Practic, CEF decouplează interdependența strictă dintre luarea decizilor de nivel 2 și de nivel 3. Ceea ce face transmiterea pachetelor IP înceată este constantă de referință “du-te – vino” dintre construcțiile de nivel 2 și 3 dintr-un dispozitiv de rețea. Deci, în măsura în care structurile de date de nivel 2 și nivel 3 pot fi decuplate, transmiterea este mai rapidă.

Cele două componente principale ale CEF sunt:

- *Forwarding Information Base (FIB)*.
- *Tabelele de adiacență*.

FIB este similar conceptual cu o tabelă de rutare. Un router utilizează tabela de rutare pentru a determina calea cea mai bună spre rețeaua destinație, bazându-se pe partea de rețea a adresei IP destinație. Cu CEF, informațiile stocate anterior în route cache sunt, în schimb, stocate în mai multe structuri de date pentru CEF switching. Structurile de date oferă căutarea optimizată pentru expediere eficientă a pachetului. Un dispozitiv de rețea utilizează tabela de căutare FIB pentru a lua decizii de switching bazate pe destinație, fără a avea acces la tabela de rutare.

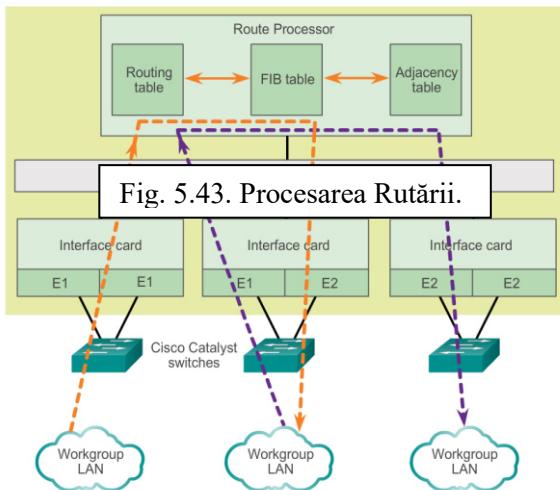
FIB este actualizat atunci când au loc schimbări în rețea și conține toate rutele cunoscute la un moment dat.

Tabelele de adiacență mențin adresele next-hop de nivel 2 pentru toate intrările FIB.

Separarea informațiilor de accesibilitate (în tabela FIB) și informațiilor de expediere (în tabela de adiacență) oferă un număr de beneficii:

- *Tabela de adiacență poate fi construită separat de cea FIB, permitând construirea amândorură fără ca nici-un pachet să fie comutat.*
- *Antetul MAC rescris utilizat pentru a expedia un pachet nu este stocat în intrările cache, deci rezcrierea schimbărilor dintr-un header MAC nu necesită invalidarea intrărilor din cache.*

CEF este activat implicit pe cele mai multe dispozitive Cisco ce efectuează comutare de nivel 3.



Dispozitivele de rețea Cisco suportă un număr de tipuri diferite de interfețe de nivel 3. O interfață de nivel 3 este una ce suportă transmiterea de pachete IP la destinație bazându-se pe adresa IP.

Cele mai importante tipuri de interfețe de nivel 3 sunt:

- **Switch Virtual Interface (SVI)** – *Interfață logică de pe un switch asociată cu virtual local area network (VLAN).*
- **Routed Port** – *Port fizic pe un switch de nivel 3 config.t pentru a se comporta ca un port de router.*
- **Layer 3 EtherChannel** – *Interfață logică pe un dispozitiv Cisco asociată cu un pachet de porturi de rutare.*

Un SVI pentru default VLAN (VLAN1) trebuie să fie activat pentru a oferi conectivitate IP host la switch și pentru a permite administrarea switchurilor de la distanță. SVIs trebuie de asemenea să fie config.t pentru a permite rutare între VLANuri. SVIs sunt interfețe logice config.t pentru VLANuri specifice; pentru a ruta între două sau mai multe VLANuri, fiecare VLAN trebuie să aibă o SVI separată activată.

Porturile de rutare permit ca switchurile Cisco (nivel 3) să se comporte ca routere. Fiecare port de pe un asemenea switch poate fi config.t ca un port pe o rețea IP independentă.

Layer 3 EtherChannels sunt utilizate pentru a "împacheta" legăturile Ethernet de nivel 3 dintre dispozitivele Cisco pentru a agrega lățimea de bandă, de obicei pe uplinks.

Notă: În plus față de SVIs și L3 EtherChannels, există și alte interfețe logice pe dispozitivele Cisco, cum ar fi interfețele loopback și interfețele tunnel.

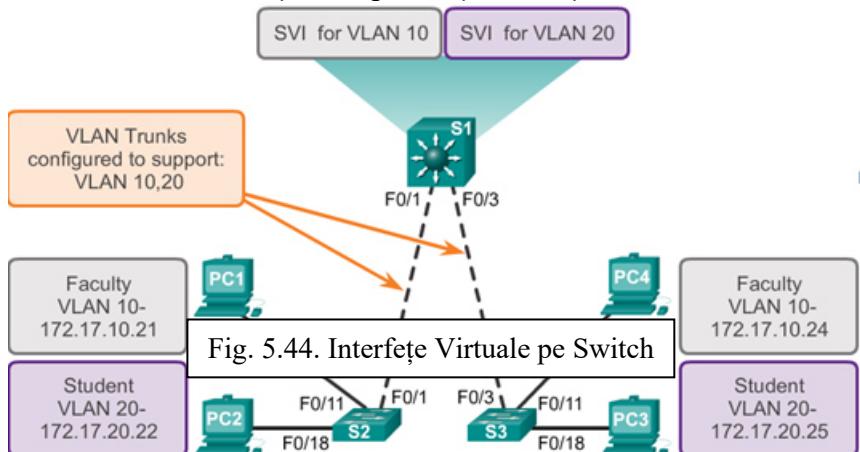


Fig. 5.44. Interfețe Virtuale pe Switch

Un port de switch poate fi configurațional să fie un port de rutare de nivel 3 și să se comporte ca o interfață obișnuită de router. Mai precis, un port de router:

- Nu este asociat cu un VLAN particular.
- Poate fi configurațional cu un protocol de rutare de nivel 3.
- Este o interfață de nivel 3 și nu suportă protocol de nivel 2.

Configurarea porturilor de router se face prin punerea interfeței în modul de nivel 3 cu ajutorul comenzi de configurație ”**no switchport**”, apoi, se atribuie o adresă IP portului.

Mai multe despre funcțiile de rutare în următorul capitol.

```

S1(config)#interface f0/6
S1(config-if)#no switchport
S1(config-if)#ip address 192.168.200.1 255.255.255.0
S1(config-if)#no shutdown
S1(config-if)#end
S1#
*Mar 1 00:15:40.115: %SYS-5-CONFIG_I: Configured from console by console
S1#show ip interface brief
Interface      IP-Address      OK? Method Status      Protocol
Vlan1         unassigned      YES unset administratively down down
FastEthernet0/1  unassigned      YES unset down      down
FastEthernet0/2  unassigned      YES unset down      down
FastEthernet0/3  unassigned      YES unset down      down
FastEthernet0/4  unassigned      YES unset down      down
FastEthernet0/5  unassigned      YES unset down      down
FastEthernet0/6  192.168.200.1 YES manual up      up
FastEthernet0/7  unassigned      YES unset up       up
FastEthernet0/8  unassigned      YES unset up       up
<output omitted>

```

Fig. 5.45. ConFig.rea
Portului de Rutare

5.8 Concluzii Capitolul 5

Ethernet uses end and intermediary devices to identify and deliver frames through networks.

Fig. 5.46.



Ethernet este cea mai utilizată tehnologie LAN de astăzi. Este o familie de tehnologii de rețea ce sunt definite de către standardele IEEE 802.2 și 802.3. Standardele Ethernet definesc protocolele de nivel 2 și tehnologiile de nivel 1. Pentru protocolele de nivel 2, împreună cu toate standardele 802IEEE, ca să funcționeze Ethernet se bazează pe două subnivele separate ale nivelului legătură de date, subnivelele LLC și MAC.

La nivelul legătură de date, structura frameului este aproape identică pentru toate vitezele Ethernet. Structura frameului Ethernet adaugă headere și trailer la PDU de nivel 3 pentru a încapsula mesajul ce este transmis.

Există două stiluri de Ethernet framing: standardul IEEE 802.3 Ethernet și standardul DIX Ethernet, numit și Ethernet II. Cea mai importantă diferență dintre cele două standarde este adăugarea unui Start Frame Delimiter (SFD) și schimbarea câmpului Type în campul Length în 802.3. Ethernet II este formatul de frame Ethernet utilizat în rețelele TCP/IP. Ca o implementare a standardelor IEEE 802.2/3, frameul Ethernet oferă adresare MAC și verificare a erorilor.

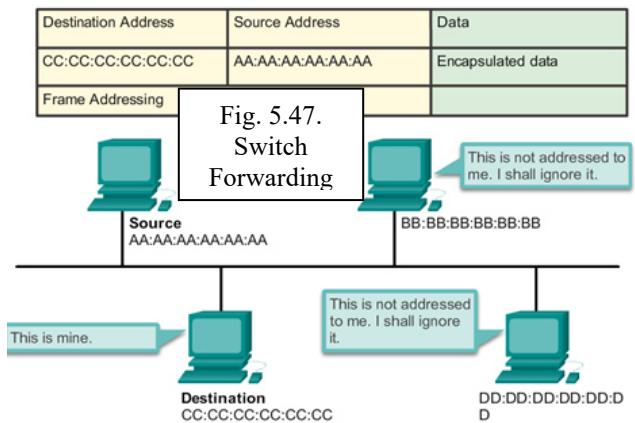
Adresarea de nivel 2 oferită de către Ethernet suportă comunicații unicast, multicast și broadcast. Ethernet utilizează Address Resolution Protocol pentru a determina adresele MAC ale destinaților și pentru a le măpa cu adrese cunoscute de la nivelul rețea.

Fiecare nod dintr-o rețea IP are o adresa MAC și una IP. Nodul trebuie să folosească propriile adrese MAC și IP în câmpurile sursă și trebuie să ofere adresele MAC și IP pentru destinație. Pe când adresa IP a destinației va fi oferită de către un nivel OSI superior, nodul expeditor trebuie să găsească adresa MAC a destinației pentru o legătură Ethernet dată. Aceasta este obiectivul ARP.

ARP se bazează pe anumite tipuri de mesaje broadcast Ethernet și mesaje unicat Ethernet, numite ARP requests și ARP replies. Protocolul ARP rezolvă adresele IPv4 în adrese MAC și menține o tabelă de mapări.

În cele mai multe rețele Ethernet, dispozitivele finale sunt de obicei conectate, într-o bază punct-la-punct, la un switch LAN de nivel 2. Switchul de nivel 2 efectuează comutare și filtrare bazându-se numai pe adresele MAC de nivel 2 OSI. Un switch de nivel 2 construiește o tabelă de adrese MAC utilizată pentru luarea decizilor de expediere. Switchurile de nivel 2 depind de routere pentru a transfera datele între subrețele IP independente.

Switchurile de nivel 3 sunt de asemenea capabile de efectuare a funcțiilor de rutare de nivel 3, reducând nevoia de routere dedicate dintr-un LAN. Deoarece switchurile de nivel 3 au hardware de switching specializat, pot ruta datele la fel de repede precum pot efectua funcția de switch.



CAPITOLUL 6. NIVELUL REȚEA

Introducere

Aplicațiile și serviciile de rețea de pe un dispozitiv pot comunica cu aplicațiile și serviciile ce rulează pe alt dispozitiv final.

Se pune întrebarea : ”Cum comunică aceste date eficient în întreaga rețea ?”

Protocolele de la nivelul rețea din modelul OSI specifică adresarea și procesele ce permit ca datele de la nivelul transport să fie împachetate și transportate. Încapsularea de la nivelul rețea permite datelor să fie livrate la destinația din rețea (sau dintr-o altă rețea) cu un overhead minim.

Acest capitol se axează pe rolul nivelului rețea. Examinează modul în care divide rețelele în grupuri de hosturi pentru a gestiona fluxul de pachete de date dintr-o rețea. Abordează de asemenea și modul în care comunicația dintre rețele are loc. Comunicația dintre rețele se numește rutare.

În weekend decizi să îți vizitezi un coleg de facultate ce este bolnav. Cunoști strada la care locuiește, însă nu ai mai fost niciodată în acest oraș. În loc să cauți adresa pe hartă, decizi să întrebi localnicii din oraș după ce te dai jos din tren. Oamenii pe care i-ai întrebat sunt de ajutor, dar au un obicei interesant, în loc să îți explice întreaga rută spre destinație, îți spun :

”Mergi pe această cale și cum ajungi la următoarea răscruce, întrebă pe cineva pentru următoarele indicații.”

Oarecum uimit de această aparentă ciudătenie, urmezi aceste instrucțiuni și în final ajungi, de la răscruce la răscruce, din drum în drum, la casa prietenului tău.

Răspunde la următoarele întrebări:

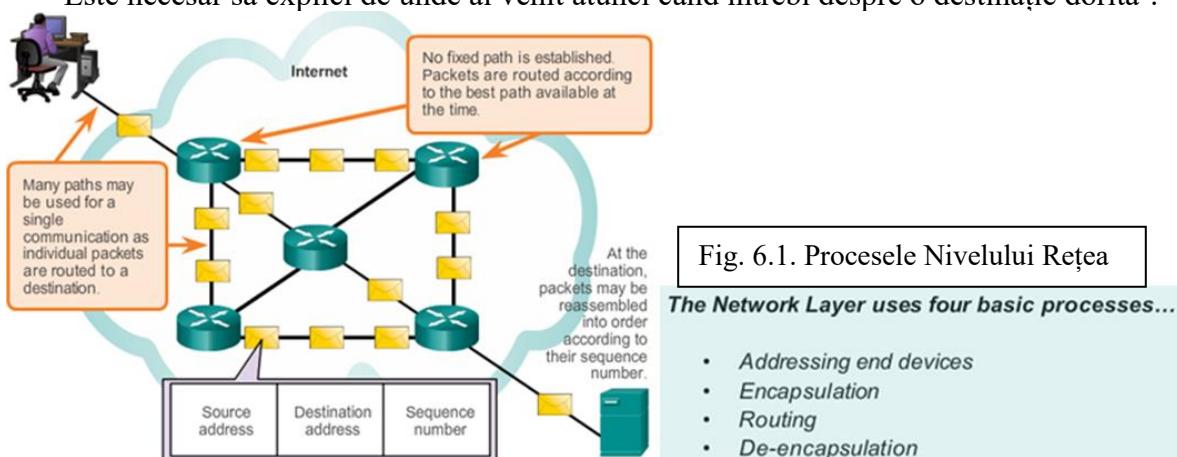
-Ar fi fost o diferență semnificativa în cazul în care și-ar fi spus despre întreaga cale sau despre cea mai mare parte din cale în schimb să fi direcționat la fiecare răscruce ?

-Ar fi fost mai util să întrebi despre adresa specifică a străzii sau doar despre numele străzii ?

- Ce s-ar fi întâmplat dacă persoana pe care o întrebai nu știa unde este strada destinație sau te direcționa spre un drum incorrect ?

-Presupune că în drumul de întoarcere acasă, vei alege să rogi localnicii să te îndrumă. Ar fi garantat că te-ar fi direcționat pe același traseu pe care ai venit la casa prietenului tău ?

Este necesar să explici de unde ai venit atunci când întrebai despre o destinație dorită ?



6.1 Network Layer Protocols

Nivelul rețea, sau nivelul 3 OSI, oferă servicii ce permit dispozitivelor finale să schimbe date în rețea. Pentru a realiza acest transport end-to-end, nivelul rețea utilizează patru procese de bază:

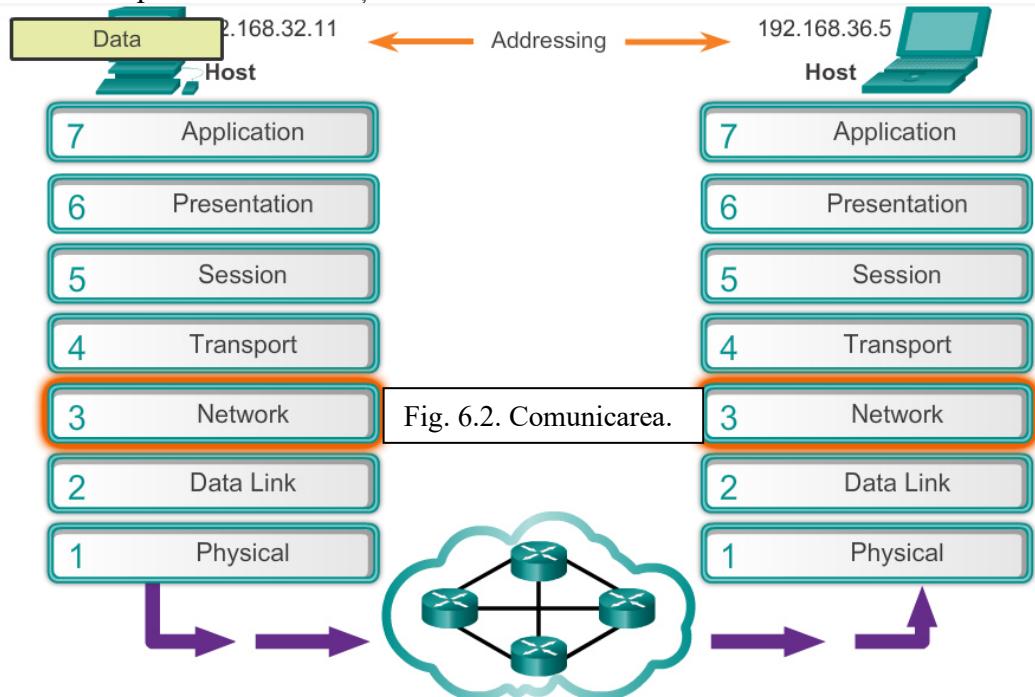
1. **Adresarea dispozitivelor finale** – la fel cum un telefon are un număr de telefon unic, dispozitivele finale trebuie să fie configurați cu o adresă IP unică pentru identificarea lor din rețea. Un dispozitiv final cu o adresă IP configurată se numește host.

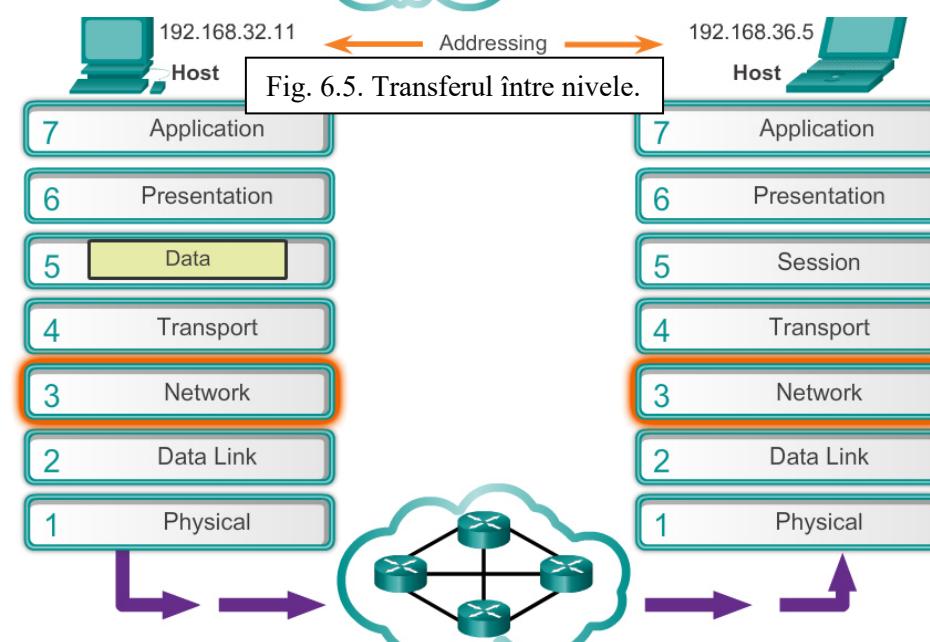
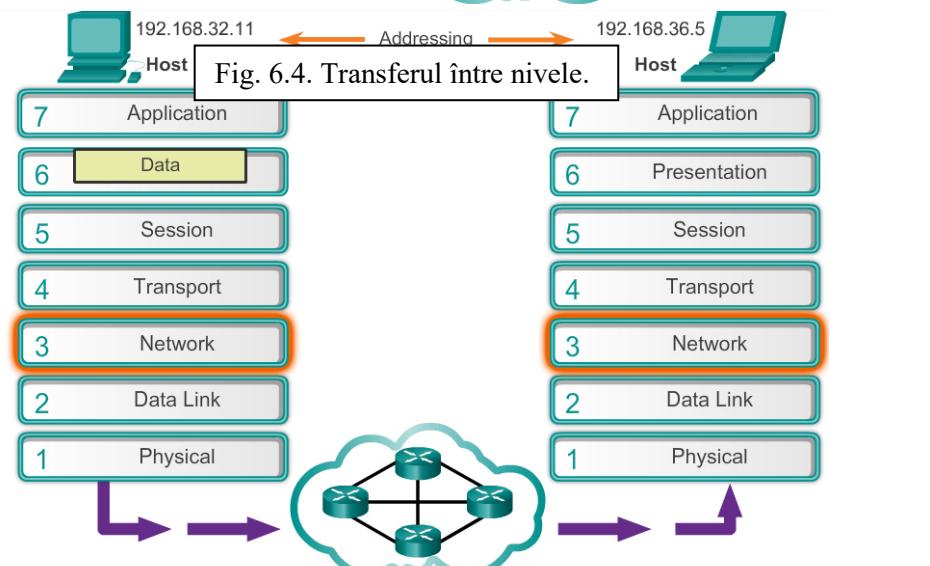
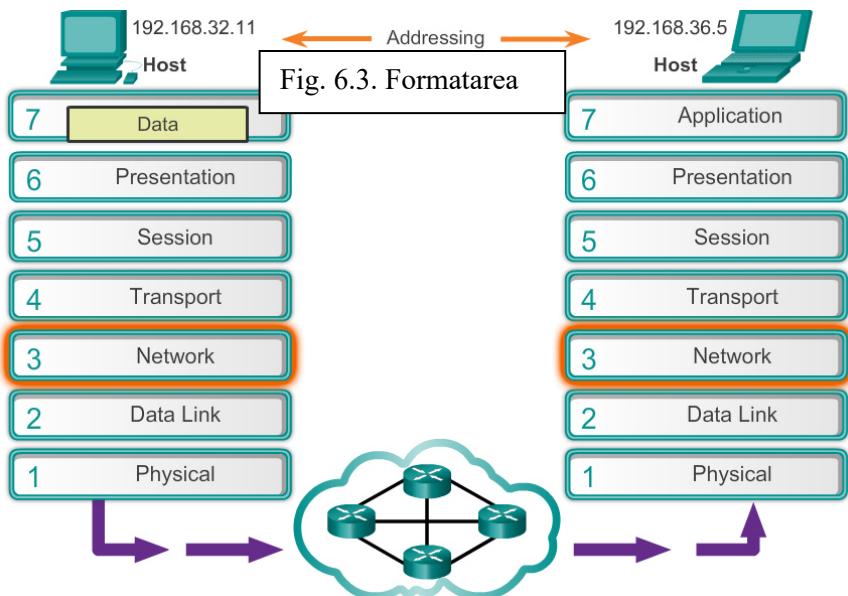
2. **Încapsularea** – nivelul rețea primește un PDU de la nivelul transport. Într-un proces numit încapsulare, nivelul rețea adaugă informații de header IP, cum ar fi adresa sursă IP și adresa destinație IP a hosturilor. După adăugarea headerului, PDU rezultat se numește pachet.

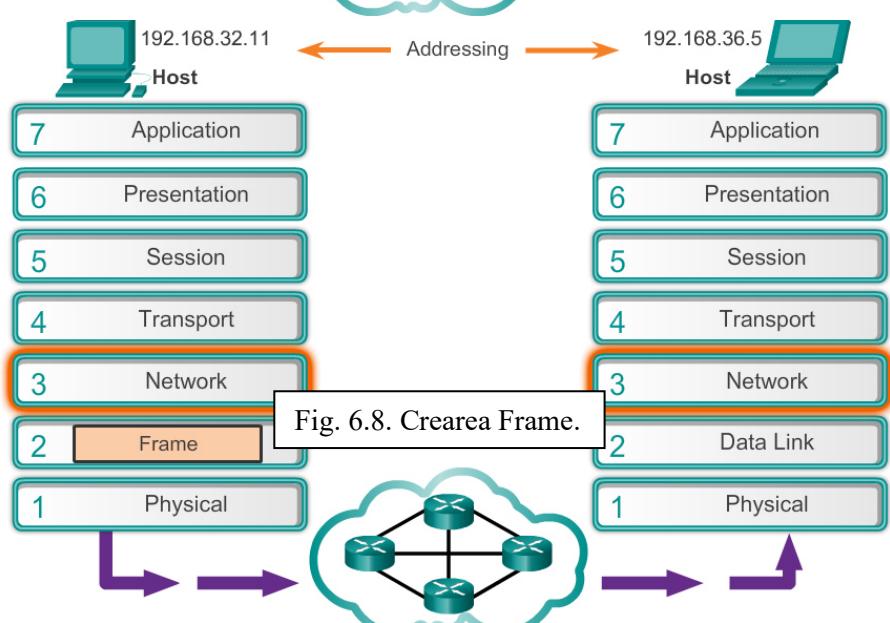
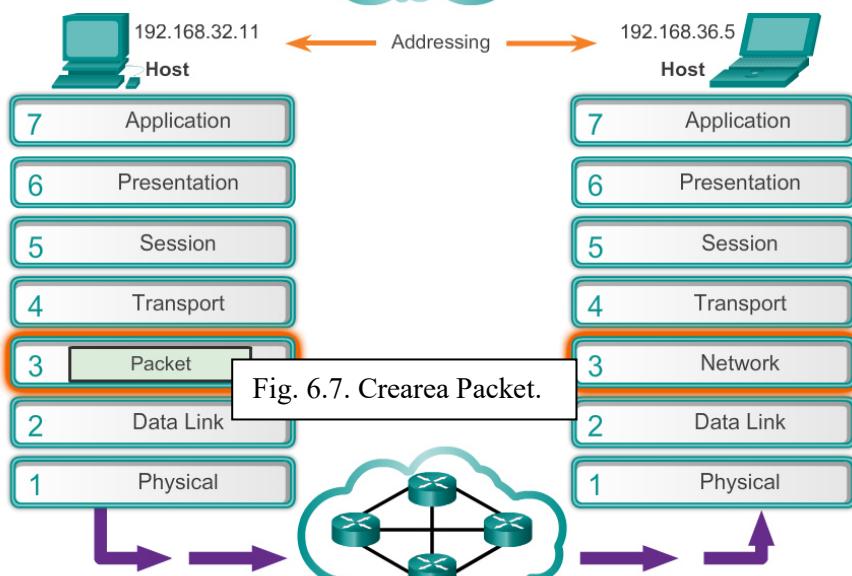
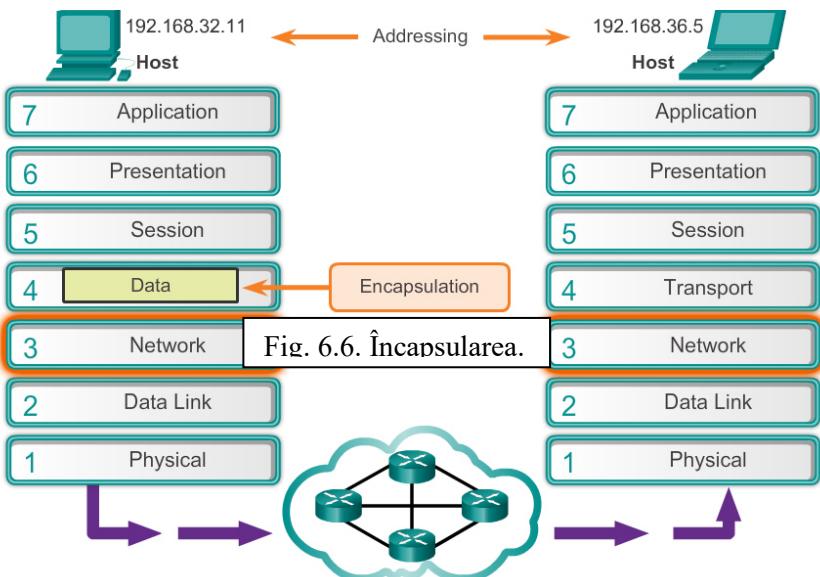
3. **Rutarea** – nivelul rețea oferă servicii de direcționare a pachetelor la un host destinație sau într-o altă rețea. Pentru că călători în alte rețele, pachetul trebuie să fie procesat de către router. Rolul routerului este de a selecta căi pentru a direcționa pachetele spre destinație într-un proces numit rutare. Un pachet ar putea traversa mai multe dispozitive intermediare înaintea ajungerii la destinația să. Fiecare echipament pe care pachetul trece prin el către destinație se numește hop.

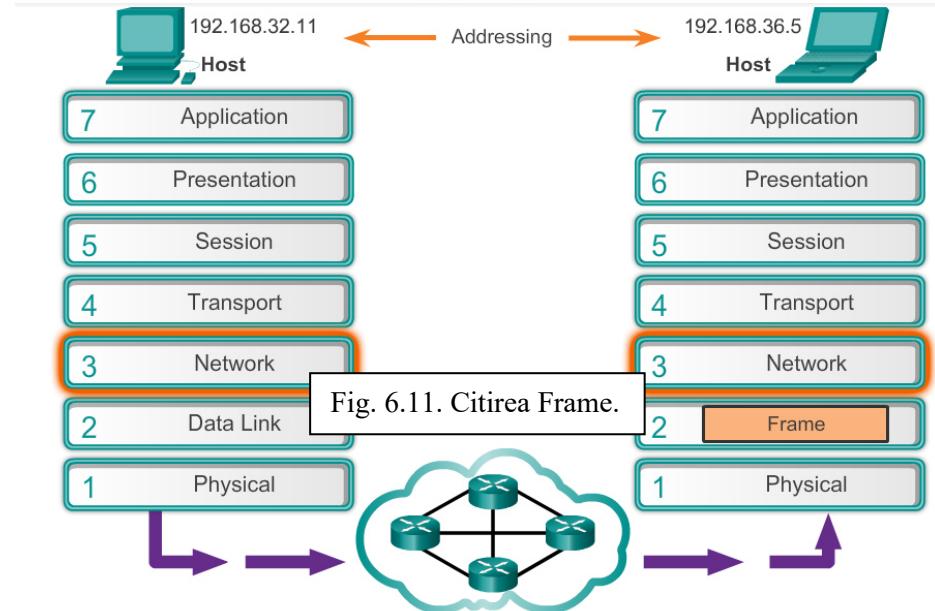
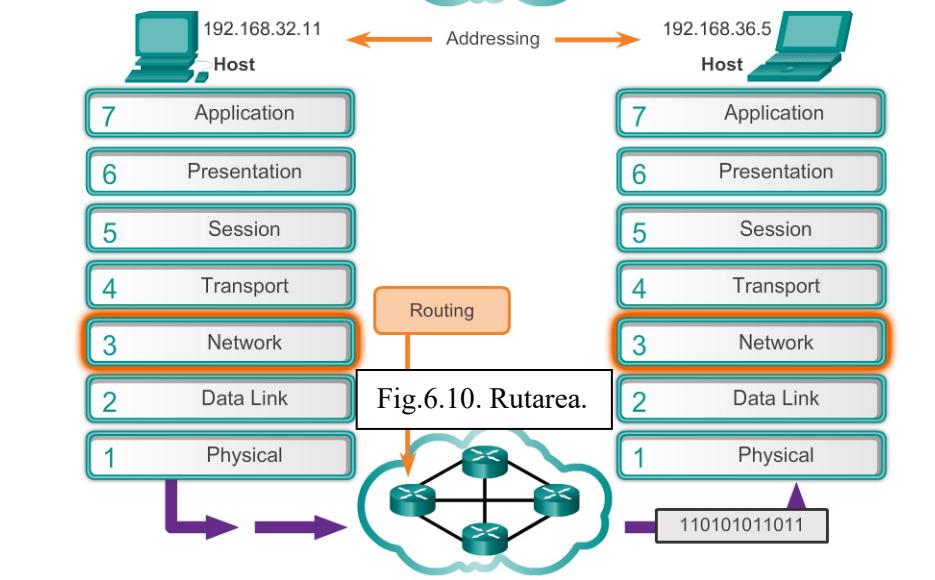
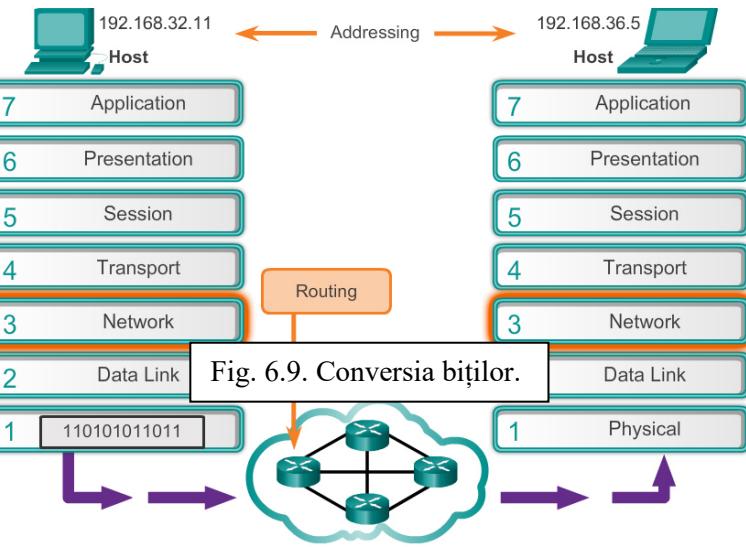
4. **Decapsularea** – atunci când pachetul ajunge la nivelul rețea de la hostul destinație, hostul verifică headerul IP al pachetului. Dacă adresa IP din header corespunde cu adresa să, headerul IP este înlăturat din pachet. Acest proces se numește decapsulare. După ce pachetul este decapsulat de nivelul rețea, PDU de nivel 4 rezultat este transmis la serviciul adecvat de la nivelul transport.

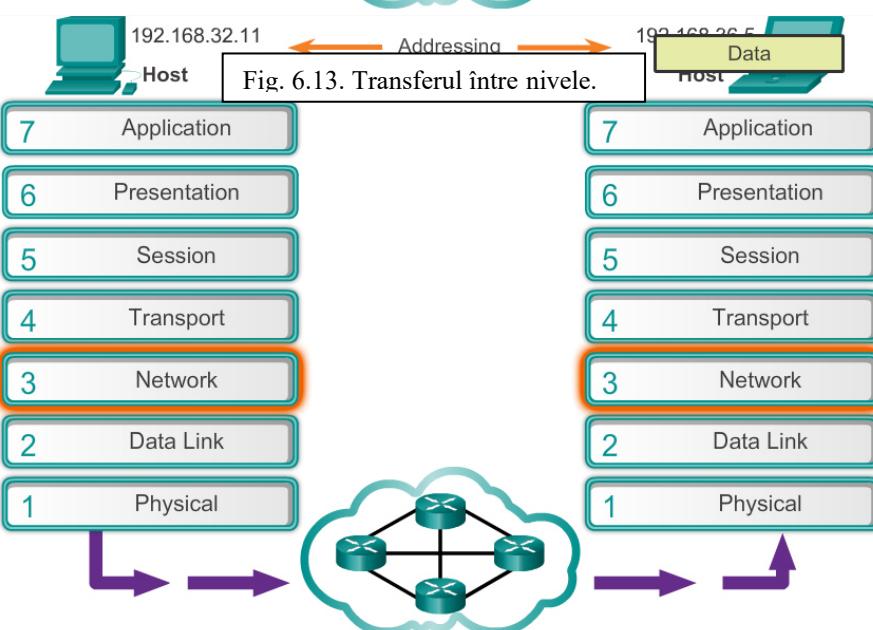
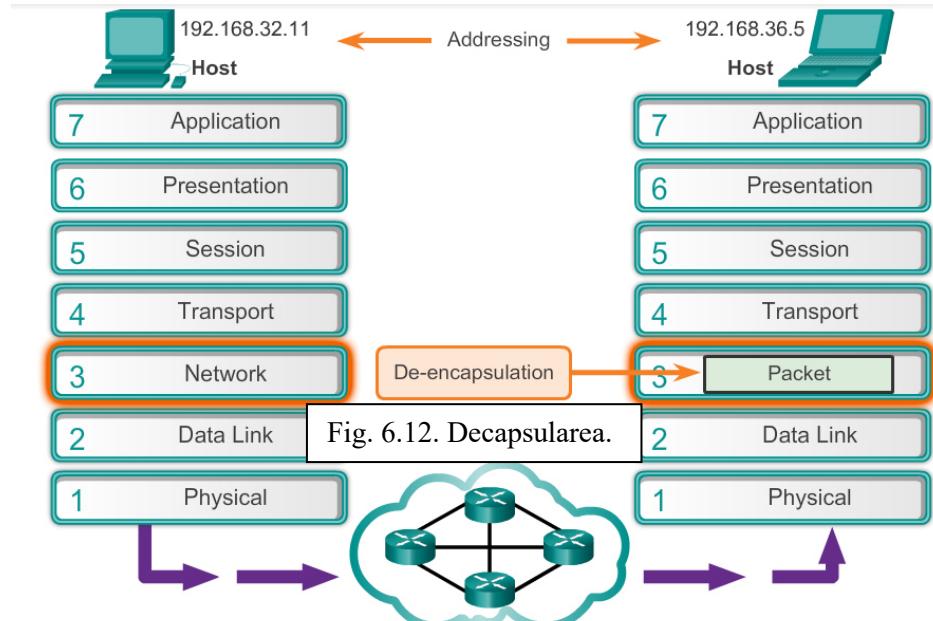
Spre deosebire de nivelul transport, nivelul 4 OSI, ce gestionează transportul datelor dintre procesele ce rulează pe fiecare host, protocolele de la nivelul rețea specifică structura pachetului și procesarea utilizată pentru transportul datelor de la un host la altul. Operarea fără a ține cont de datele transportate în fiecare pachet permite nivelului rețea să transporte datele pentru mai multe tipuri de comunicații dintre mai multe hosturi.









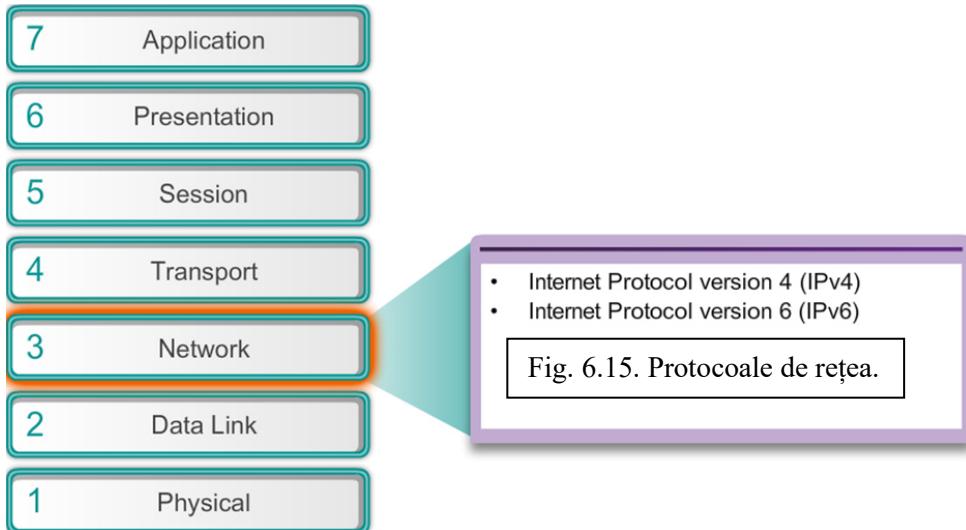


Există mai multe protocoale de nivel rețea; însă, numai următoarele două sunt implementate de obicei, aşa cum este evidențiat și în Fig. 6.14. :

- *Internet Protocol version 4 (IPv4).*
- *Internet Protocol version 6 (IPv6).*

Alte protocoale de nivel rețea utilizate global sunt:

- *Novell Internetwork Packet Exchange (IPX).*
- *AppleTalk.*
- *Connectionless Network Service (CLNS/DECNet).*



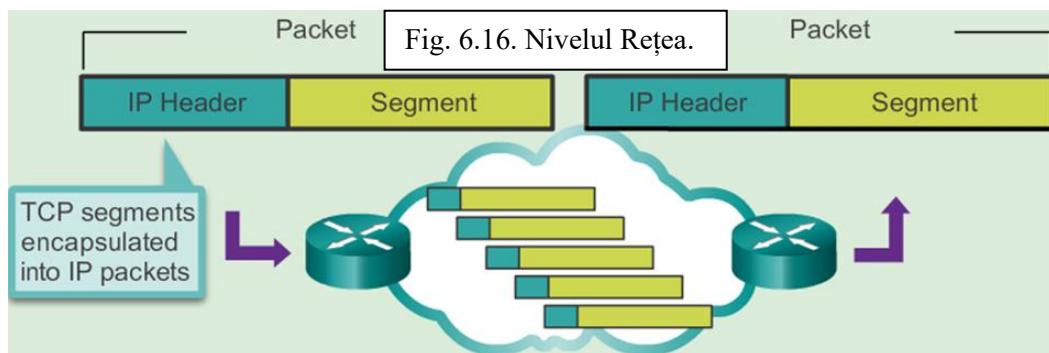
6.1.1 Caracteristici ale protocolului IP

IP este serviciul de la nivelul rețea implementat de către suita de protocole TCP/IP.

IP a fost dezvoltat ca un protocol cu overhead scăzut. Oferă numai funcții necesare pentru livrarea unui pachet de la o sursă la o destinație, peste un sistem interconectat de rețele. Protocolul nu a fost dezvoltat pentru a urmări și gestiona fluxul de pachete. Aceste funcții, dacă sunt necesare, sunt efectuate de către alte protocole de la alte nivele.

Caracteristicile de bază ale IP sunt:

- **Connectionless** - Nu este stabilită nici-o conexiune cu destinația înaintea expedierii pachetelor da date.
- **Best Effort (unreliable)** – Livrarea pachetului nu este garantată.
- **Media Independent** – Funcționarea este independentă de mediul care transportă datele.

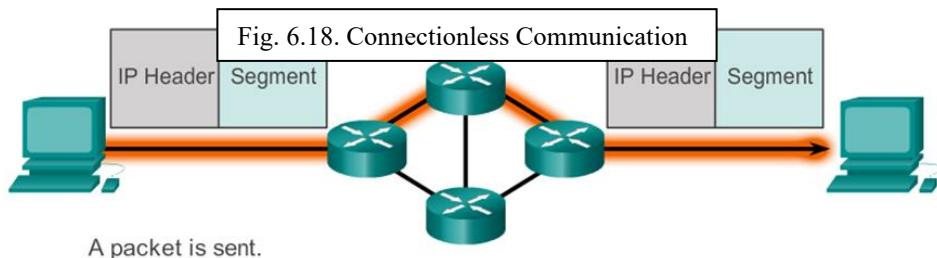
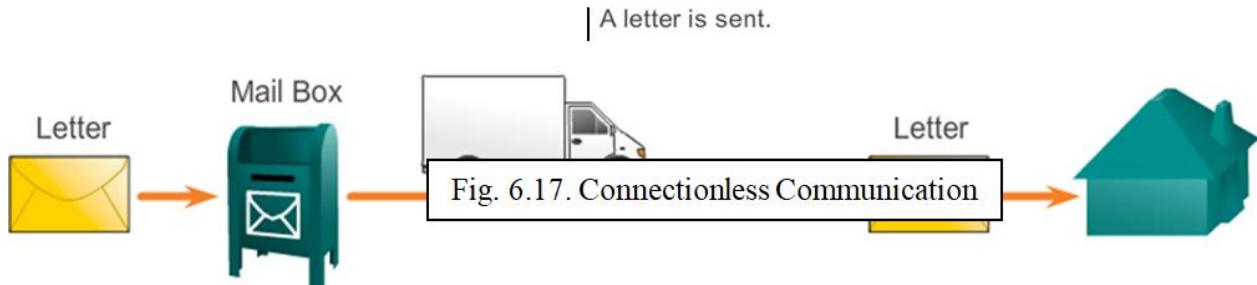


Rolul nivelului rețea este de a transporta pachete între hosturi cu overhead cât mai scăzut în rețea, pe cât posibil. Nivelul rețea nu se ocupă de, sau nu este conștient de, tipul de comunicații conținut în pachet. IP este connectionless, însemnând că nu este creată nici-o conexiune dedicată end-to-end înaintea transferului de date. Comunicațiile connectionless sunt similare cu expedierea unei scrisori cuiva fără a notifica destinatarul înainte.

Așa cum este arătat și în Fig. xxx, serviciul poștal utilizează informațiile dintr-o scrisoare pentru a livra scrisoarea la destinatar. Adresa de pe plic nu oferă informații despre faptul că destinatarul este prezent, ori când scrisoare ajunge, sau dacă destinatarul poate citi scrisoarea. De fapt, serviciul poștal nu este conștient de informațiile conținute în pachetul livrat și, prin urmare, nu poate oferi nici-un mecanism de corecție a oricărei erori.

Comunicațiile de date connectionless funcționează pe același principiu.

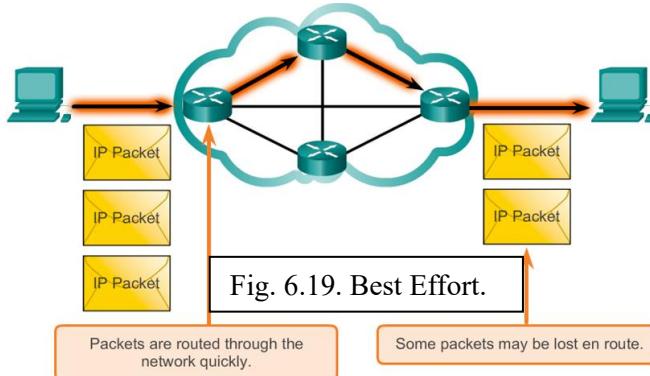
IP este connectionless și, prin urmare, nu necesită schimb inițial de informații de control pentru a stabili o conexiune end-to-end înaintea livrării pachetului. IP nu necesită câmpuri adiționale în headerul PDU pentru a gestiona o conexiune stabilită. Acest proces reduce mult overheadul IP. Însă, cu nici-o conexiune end-to-end prestabilită, expeditorii nu sunt conștienți de faptul că dispozitivele destinație sunt prezente și funcționale atunci când expediază pachetele sau de faptul că dispozitivele destinație primesc pachetul sau dacă sunt capabile să îl acceseze și să îl citească. Fig. 6.17 evidențiază un exemplu de comunicare connectionless.



IP este de obicei referit ca un protocol unreliable sau best-effort delivery. Acest lucru nu înseamnă ca IP funcționează corespunzător uneori și nu funcționează bine alteori, sau că este un protocol de comunicații de date “sărac”. Unreliable înseamnă doar că IP nu are capacitatea să gestioneze și să recupereze pachetele alterate sau nelivrate. Acest lucru este datorită faptului că pachetele IP sunt transmise cu informații despre locația de livrare și nu conțin nici-o informație ce poate fi procesată pentru a informa expeditorul de faptul că livrarea s-a efectuat cu succes. Nu există inclusiv sincronizarea datelor în headerul pachetului pentru a urmări ordinea livrării pachetului. Nu există nici-un acknowledgments de livrare a pachetului cu IP și nu există nici control de eroare pentru a urmări dacă pachetele au fost livrate fără să fie alterate. Pachetele ar putea ajunge la destinație alterate sau deloc. Bazându-se pe informațiile oferite în headerul IP, nu există nici-o capacitate pentru retransmiterea pachetului în cazul în care au loc asemenea erori.

În cazul în care pachetele lipsă sau care nu sunt în ordinea dorită crează probleme pentru aplicații, serviciile de nivel superior, precum TCP, trebuie să rezolve aceste probleme. Acest lucru permite ca IP să funcționeze foarte eficient. În cazul în care reliability overhead erau incluse în IP, comunicațiile ce nu necesitau conexiuni sau încredere vor fi consumatoare de lățime de bandă și vor rezulta întârzieri produse de acest overhead. În suita TCP/IP, nivelul transport poate utiliza fie TCP, fie UDP, în funcție de nevoie de încredere în comunicație. Lăsând decizia de încredere nivelului transport face IP mai adaptabil și consiliabil pentru diferite tipuri de comunicare.

Fig. 6.19 arată un exemplu de comunicații IP. Protocolele orientate pe conexiune, cum ar fi TCP, necesită ca datele de control să fie schimbate pentru a stabili conexiunea. Pentru a gestiona informațiile despre conexiune, TCP necesită de asemenea câmpuri adiționale în headerul PDU.

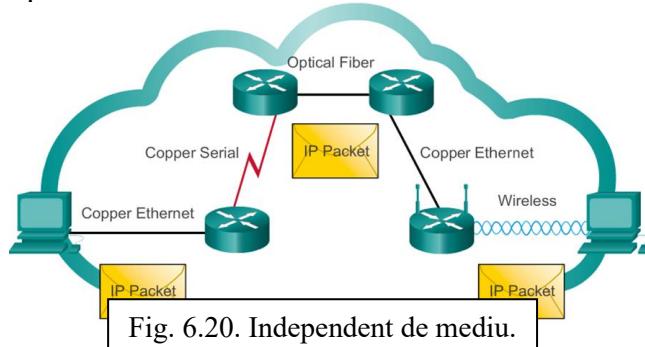


Nivelul rețea nu este de asemenea încărcat cu caracteristici ale mediului pe care sunt transportate pachetele. IP funcționează independent de mediul ce transportă datele la nivele inferioare ale stivei de protocole. Ca și în Fig. xxx, orice pachet IP individual poate fi comunicat electric pentru cablu, ca semnale optice peste fibra optică sau ca semnale radio pe mediul wireless.

Este responsabilitatea nivelului legătură de date OSI de a lua un pachet IP și de pregătirea sa pentru transferul prin mediul de comunicații. Acest lucru înseamnă că transportul pachetelor IP nu este limitat la un mediu particular.

Există, însă, o caracteristică importantă a mediului pe care nivelul rețea o ia în considerare: dimensiunea maximă a PDU-ului pe care fiecare mediu o poate transporta. Această caracteristică se numește maximum transmission unit (MTU). O parte a comunicațiilor de control dintre nivelul legătură de date și nivelul rețea este stabilirea dimensiunii maxime pentru un pachet. Nivelul legătură de date transferă valoarea MTU la nivelul rețea. Nivelul rețea determină apoi dimensiunea maximă a pachetului.

În unele cazuri, un dispozitiv intermediar, de obicei un router, trebuie să împartă un pachet atunci când îl transmite de pe un mediu pe altul cu o valoare MTU mai mică. Acest proces se numește fragmentare a pachetului.

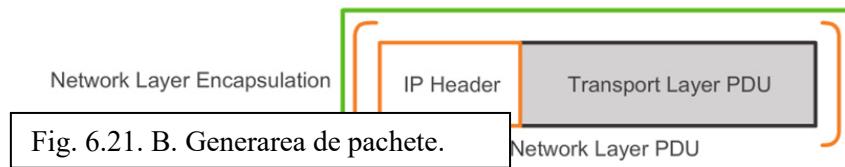
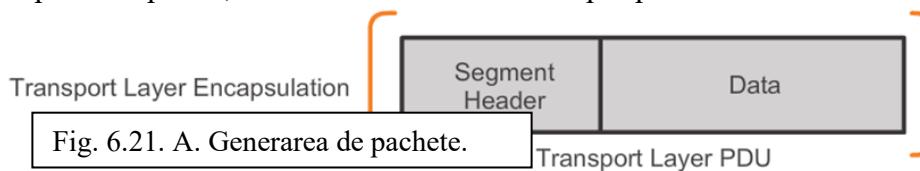


IP încapsulează, sau împachetează, segmentul de la nivelul transport prin adăugarea unui header IP. Acest header este utilizat pentru a livra pachetul la hostul destinație. Headerul IP rămâne din momentul în care pachetul pleacă de la nivelul rețea a sursei până când ajunge la nivelul rețea de la hostul destinație.

Fig. xxx arată procesul de creare a PDU de la nivelul transport. Fig. 2 arată procesul de creare a PDU de nivel rețea.

Procesul de încapsulare a datelor nivel cu nivel permite serviciilor de la nivele diferite să se dezvolte fără a afecta alte nivele. Acest lucru înseamnă că segmentele de la nivelul transport pot fi împachetate de IPv4 sau IPv6 sau de orice nou protocol ce poate fi dezvoltat în viitor.

Ruterele pot implementa aceste protocole diferite de la nivelul rețea să funcționeze concurențial pe rețea de la și la aceleași hosturi sau de la și la hosturi diferite. Rutarea efectuată de către aceste dispozitive intermediare ia în considerare numai conținutul headerului de pachet ce încapsulează segmentul. În toate cazurile, porțiunea de date a pachetului, care este PDU de nivel transport încapsulat, rămâne neschimbată în timpul proceselor de la nivelul rețea.



Connectionless	Best Effort	Media Independent
No contact is made with the destination host before sending a packet.	Packet delivery is not guaranteed.	Will adjust the size of the packet sent depending on what type of network access will be used.
Will send a packet even if the destination host is not able to receive it.	Does not guarantee that the packet will be delivered fully without errors.	Fiber optics cabling, satellites, and wireless can all be used to route the same packet.

Fig. 6.22. Metode de livrare.

6.1.2 Pachetul IPv4

IPv4 este utilizat încă din 1983 când a fost dezvoltat în Advanced Research Projects Agency Network (ARPANET), predecesorul Internetului. Internetul este bazat în mare parte pe IPv4, care este încă cel mai utilizat protocol de nivel rețea.

Un pachet IPv4 are două părți:

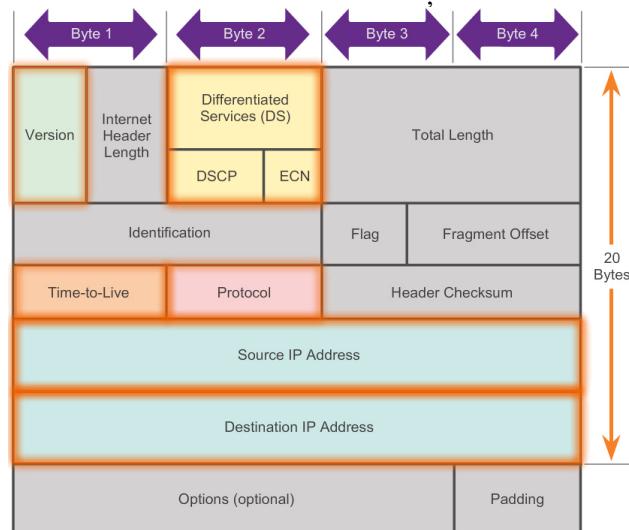
- **IP Header** – Identifică caracteristicile pachetului.
- **Payload** – Conține informații de segment de nivel 4 și datele.

Așa cum este arătat și în Fig. xxx, un header de pachet IPv4 constă din câmpuri ce conțin informații importante despre pachet. Aceste câmpuri conțin numere binare ce sunt examineate de către procesul de nivel 3. Valorile binare ale fiecărui câmp identifică setările diferite ale pachetului IP.

Câmpuri importante din headerul IPv4 sunt:

- 1) **Versiunea** – Conține o valoare binară de 4 biți ce identifică versiunea pachetului IP. Pentru pachetele IPv4, acest câmp este întotdeauna 0100.
- 2) **Differentiated Services (DS)** – Numit și câmpul Type of Service (ToS), câmpul DS este un câmp de 8 biți utilizat pentru a determina prioritatea fiecărui pachet. Primii 6 biți identifică valoarea Differentiated Services Code Point (DSCP) ce este utilizată de către mecanismul QoS. Ultimii 2 biți identifică valoarea explicit congestion notification (ECN) ce poate fi folosită pentru a preveni aruncarea pachetelor în timpul congestiei de rețea.
- 3) **Time-to-Live (TTL)** – Conține o valoare binară de 8 biți ce este folosită pentru a limita timpul de viață al unui pachet. Este specificat în secunde, dar nu este referit adesea ca hop count. Expeditorul pachetului setează valoarea TTL și este decrementată cu 1 de fiecare dată când pachetul este procesat de către un router, sau hop. Dacă valoarea TTL ajunge la 0, routerul aruncă pachetul și trimită un mesaj Internet Control Message Protocol (ICMP) Time Exceeded la adresa IP sursă. Comanda traceroute utilizează acest câmp pentru a identifica routerele utilizate între sursă și destinație.
- 4) **Protocol** – Această valoare binară de 8 biți indică tipul de payload de date pe care pachetul îl conține, ceea ce permite ca nivelul rețea să transfere datele protocolului adecvat de nivel superior. Valorile comune sunt ICMP (0x01), TCP (0x06), și UDP (0x11).
- 5) **Adresa IP sursă** – Conține o valoare binară de 32 de biți ce reprezintă adresa IP sursă a pachetului.
- 6) **Adresa IP destinație** – Conține o valoare binară de 32 de biți ce reprezintă adresa IP destinație a pachetului.

Două dintre cele mai comune referite câmpuri sunt adresele IP sursă și destinație. Aceste câmpuri identifică de unde pachetul provine și unde se duce. În mod normal, aceste adrese nu se schimbă în drumul de la sursă la destinație.



Câmpurile rămase sunt folosite pentru a identifica și valida pachetul sau pentru reordonarea pachetelor fragmentate.

Câmpurile utilizate pentru identificarea și validarea pachetului sunt:

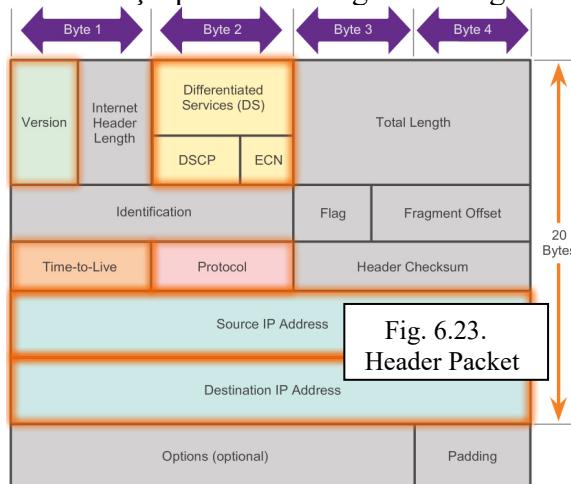
- i. **Internet Header Length (IHL)** - Conține o valoare binară de 4 biți ce identifică numărul de 32 de biți în header. Valoarea IHL variază în funcție de câmpurile Options și Padding. Valoarea minimă a acestui câmp este 5 (i.e., $5 \times 32 = 160$ bits = 20 bytes) și valoarea maximă este 15 (i.e., $15 \times 32 = 480$ bits = 60 bytes).

ii. **Total Length** - Numit și Pachet Length, acest câmp definește dimensiunea întreagă a pachetului, inclusiv headerul și datele, în bytes. Dimensiunea minimă a pachetului este 20 bytes (20-byte header + 0 bytes data și cea maximă 65,535 bytes).

iii. **Header Checksum** - Câmpul de 16 biți este folosit pentru verificarea de erori în headerul IP. Checksumul headerului este recalculată și comparată cu valoarea din câmpul checksum. Dacă valorile nu corespund, pachetul este aruncat.

Un router ar putea să trebuiască să fragmenteze un pachet atunci când îl trasmite dintr-un mediu pe altul ce are o valoare MTU mai mică. Atunci când are loc acest lucru, fragmentarea are loc și pachetul IPv4 utilizează următoarele câmpuri pentru a ține evidența fragmentelor:

- **Identification** – Acest câmp de 16 biți identifică în mod unic fragmentul unui pachet IP original.
- **Flags** – Acest câmp de 3 biți identifică modul în care pachetul este fragmentat. Este utilizat împreună cu câmpurile Fragment Offset și Identification pentru a ajuta la reconstrucția fragmentului în pachetul original.
- **Fragment Offset** – Acest câmp de 13 biți identifică ordinea în care trebuie plasat un fragment de pachet în reconstrucția pachetului original nefragmentat.



Wireshark este un instrument util de monitorizare a rețelei pentru oricine lucrează cu rețele și poate fi folosit în mai multe laboratoare din cursurile Cisco Certified Network Associate (CCNA) pentru analiza datelor și depanări. Poate fi utilizat pentru a vizualiza valori conținute în câmpurile din headerul IP.

Cele trei Figure conțin capturi din diferite pachete IP:

- Fig. 6.24 arată conținutul pachetului numărul 2 din captura aceasta. Sursă este 192.168.1.109 și destinația este 192.168.1.1. Fereastra din mijloc conține informații despre headerul IPv4, precum dimensiunea sa, dimensiunea totală și orice flag ce este setat.
- Fig. 6.25 arată conținutul pachetului numărul 8 din această captură. Acesta este un pachet HTTP. De remarcat prezența informațiilor dincolo secțiunea TCP.
- Fig. 6.26 arată conținutul pachetului numărul 16. Acest pachet este un **ping** request de la 192.168.1.109 la 192.168.1.1. De remarcat că nu există informații TCP sau UDP deoarece este un pachet Internet Control Message Protocol (ICMP).

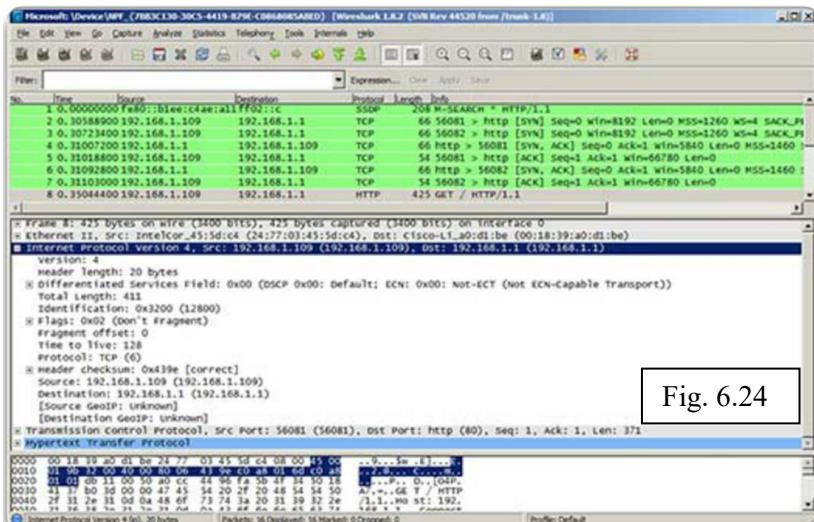


Fig. 6.24

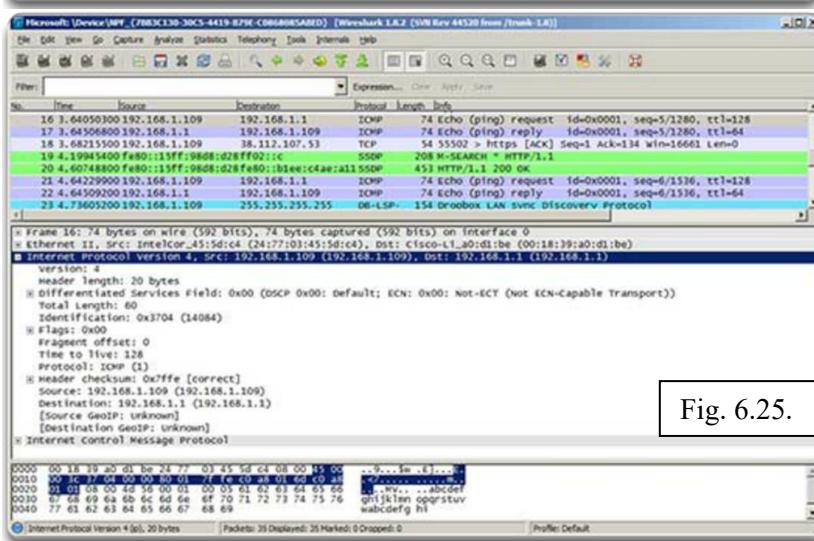


Fig. 6.25.

Fig. 6.26. Câmpurile Headerelor.

Version	Differentiated Services	Internet Header Length
Always set to 0100 for IPv4	Identifies the priority of each packet	Identifies the number of 32-bit words in the header
Time-to-Live	Protocol	Total Length
Commonly referred to as hop count	Identifies the upper-layer protocol to be used next	Maximum value is 65,535 bytes
Source IP Address	Destination IP Address	Header Checksum
Identifies the IP address of the sending host	Identifies the IP address of the recipient host	Error-checks the IP header – if incorrect, the packet is discarded

6.1.3 IPv6 Packet

Pe parcursul anilor, IPv4 a fost actualizat pentru a răspunde la noi provocări. Însă, chiar cu schimbări, IPv4 are încă trei probleme importante:

A. **Epuizarea adreselor IP** – IPv4 are un număr limitat de adrese IP publice unice disponibile. Deși există aproximativ 4 miliarde de adrese IPv4, numărul mare de noi dispozitive, conexiunile mereu disponibile și creșterea potențial a regiunilor mai puțin dezvoltate au crescut nevoia de mai multe adrese.

B. **Extinderea tableei de rutare Internet** – O tabelă de rutare este utilizată de către routere pentru a determina cele mai bune căi. Deoarece numărul de servere (noduri) conectate la Internet crește, numărul de routere crește și el. Aceste căi IPv4 consumă o cantitate mare de memorie și de resurse de processor ale routerelor din Internet.

C. **Blocarea conectivității end-to-end** - Network Address Translation (NAT) este o tehnologie implementată în rețelele IPv4. NAT oferă un mod pentru dispozitive de a împărți o singură adresă IP publică. Însă, datorita faptului că adresa IP publică este partajată, adresa IP a unui host de rețea intern este ascunsă. Acest lucru poate fi o problemă pentru tehnologiile ce necesită o conectivitate end-to-end.



Fig. 6.27.

La începutul anilor 1990, Internet Engineering Task Force (IETF) a fost preocupat de problemele IPv4 și a început să caute o soluție. Această activitate a condus la dezvoltarea IPv6. IPv6 depășește limitele lui IPv4 și este un accesoriu puternic cu caracteristici ce se potrivesc mai bine cererilor de rețea actuale și previzibile.

Îmbunătățirile aduse de IPv6 sunt:

- **Creșterea spațiului de adresare** – *adresele IPv6 sunt bazate pe adresare ierarhică pe 128 de biți, spre deosebire de IPv4 cu 32 de biți. Aceast lucru crește foarte mult numărul de adrese IP disponibile.*
- **Manipulare a pachetului îmbunătățită** – *hederul IPv6 a fost simplificat, având mai puține câmpuri. Aceast lucru îmbunătățește manipularea pachetui de către routerele intermediare și oferă de asemenea suport pentru extinderi și opțiuni pentru creșterea scalabilității.*
- **Elimină nevoie de NAT** – *Cu un număr atât de mare de adrese IPv6 publice, NAT nu mai este necesar. Clientii, de la întreprinderile mari la o singură locuință, pot avea o adresă de rețea IPv6 publică. Aceast lucru evită unele dintre problemele de aplicație induse de către NAT, experimentate de către aplicațiile ce necesită conectivitate end-to end.*
- **Securitate integrată** – IPv6 suportă implicit autentificare și capabilități private. Cu IPv4, caracteristici suplimentare trebuie să fie implementate pentru a realiza acest lucru.

Spațiul de adrese IPv4 pe 32 de biți oferă aproximativ 4.294.967.296 adrese unice. Dintre acestea, numai 3.7 miliarde sunt asignabile, deoarece sistemul de adresare IPv4 separă adresele în clase și rezervă adrese pentru multicasting, testare și alte utilizări specifice.

Așa cum este arătat și în Fig. xxx, spatial de adrese IPv6 oferă 340.282.366.920.938.463.463.374.607.431.768.211.456, or 340 undecillion de adrese, care este aproximativ echivalent cu fiecare bob de nisip de pe Pământ.

Legend

- Yellow box: There are 4 billion IPv4 addresses
- Green box: There are 340 undecillion IPv6 addresses

Number Name	Scientific Notation	Number of Zeros	Fig. 6.28. Numărul de Adrese IPv6.
1 Thousand	10^3	1,000	
1 Million	10^6	1,000,000	
1 Billion	10^9	1,000,000,000	
1 Trillion	10^{12}	1,000,000,000,000	
1 Quadrillion	10^{15}	1,000,000,000,000,000	
1 Quintillion	10^{18}	1,000,000,000,000,000,000	
1 Sextillion	10^{21}	1,000,000,000,000,000,000,000	
1 Septillion	10^{24}	1,000,000,000,000,000,000,000,000	
1 Octillion	10^{27}	1,000,000,000,000,000,000,000,000,000	
1 Nonillion	10^{30}	1,000,000,000,000,000,000,000,000,000,000	
1 Decillion	10^{33}	1,000,000,000,000,000,000,000,000,000,000,000,000	
1 Undecillion	10^{36}	1,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000	

Una dintre cele mai importante îmbunătățiri de design ale IPv6 peste IPv4 este simplificarea headerului IPv6.

Headerul IPv4 constă din 20 de octeți și 12 câmpuri de bază, fără câmpul Options și câmpul Padding.

Headerul IPv6 constă din 40 de octeți (chiar mai mult în funcție de lungimea adreselor IPv6 sursă și destinație) și 8 câmpuri de header (3 câmpuri de bază IPv4 și 5 câmpuri de header adiționale).

Fig. 1 arată structura headerului IPv4. Ca și în Fig. , pentru IPv6, unele câmpuri au rămas la fel, unele câmpuri din headerul IPv4 nu mai sunt folosite și unele câmpuri și-au schimbat numele și poziția.

Un nou câmp a fost adăugat la IPv6 și nu este folosit de IPv4. Headerul IPv6 simplificat este arătat în Fig. 2.

Headerul IPv6 simplificat oferă mai multe avantaje peste IPv4:

- *Eficiența de rutare mai bună pentru performanță și forwarding-rate scalability.*
- *Nu este cerere pentru procesare de checksums.*
- *Mecanisme mai eficiente de extensie de header și simplificate (spre deosebire de câmpul Options IPv4).*
- *Un câmp Flow Label pentru procesare per-flow cu nici-o necesitate de deschidere a pachetului interior de transport pentru a identifica fluxurile de trafic diferite.*

Legend

- Yellow box: - Field names kept from IPv4 to IPv6
- Light Green box: - Name and position changed in IPv6
- Grey box: - Fields not kept in IPv6

Fig. 6.29. Header IPv4.

Version	IHL	Type of Service	Total Length			
Identification		Flags	Fragment Offset			
Time-to-Live	Protocol	Header Checksum				
Source Address						
Destination Address						
Options		Padding				

Legend

- Yellow box: - Field names kept from IPv4 to IPv6
- Light Green box: - Name and position changed in IPv6
- Green box: - New field in IPv6

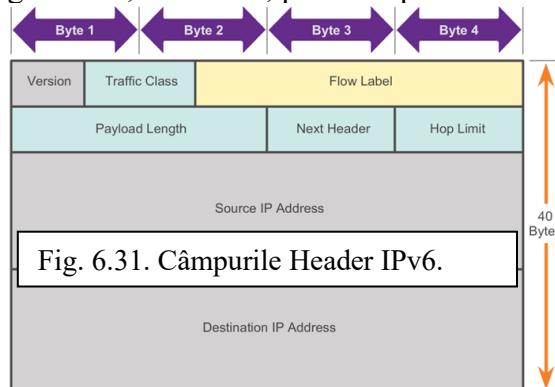
Fig. 6.30. Header IPv6.

Version	Traffic Class	Flow Label	
Payload Length		Next Header	Hop Limit
Source IP Address			
Destination IP Address			

Câmpurile incluse în headerul pachetului IPv6 sunt:

- **Version** – Acest câmp conține o valoare binară de 4 biți ce identifică versiunea pachetului IP. Pentru IPv6, acest câmp este întotdeauna 0110.
- **Traffic Class** – Acest câmp de 8 biți este echivalent câmpului IPv4 Differentiated Services (DS). Conține de asemenea o valoare Differentiated Services Code Point (DSCP) pe 6 biți ce este folosită pentru clasificarea pachetelor și o valoare de 2 biți Explicit Congestion Notification (ECN) folosită pentru controlul congestiei de trafic.
- **Flow Label** - Acest câmp de 20 de biți oferă un serviciu special pentru aplicațiile din timp real. Foarte fi utilizat pentru a informa routerele și switchurile să mențină aceeași cale pentru fluxul de pachete astfel încât pachetele nu vor fi reordonate.
- **Payload Length** – Acest câmp de 16 biți este echivalent câmpului Total Length din headerul IPv4. Definește întreaga dimensiune a pachetului, inclusiv headerul și extensiile opționale.
- **Next Header** – Acest câmp de 8 biți este echivalent câmpului IPv4 Protocol. Indică tipul de payload de date conținute de pachet, permitând ca nivelul rețea să transfere datele la protocolul de nivel superior adecvat. Acest câmp este de asemenea folosit dacă există hedere opționale de extindere adăugate la pachetul IPv6.
- **Hop Limit** – Acest câmp de 8 biți înlocuiește câmpul TTL IPv4. Această valoare este decrementată cu 1 de fiecare router ce expediază pachetul. Atunci când număratoare ajunge la 0, pachetul este aruncat și un mesaj ICMPv6 este transmis la hostul expeditor, indicând faptul că pachetul nu a ajuns la destinație.
- **Adresa sursă** – Acest câmp de 128 de biți identifică adresa IPv6 sursă.
- **Adresa destinație** – Acest câmp de 128 de biți identifică adresa IPv6 destinație.

Un pachet IPv6 ar putea conține de asemenea headere de extindere (EH), ce oferă informații opționale de nivel rețea. EH sunt opționale și plasate între hedereul IPv6 și payload. EH sunt utilizate pentru fragmentare, securitate, pentru suportul de mobilitate etc.



La vizualizarea capturilor Wireshark IPv6, remarcăm faptul că headerul IPv6 are mai puține câmpuri decât headerul IPv4. Acest lucru face ca headerul IPv6 să fie mai ușor și mai rapid de procesat pentru routere.

Adresele IPv6 arată foarte diferit. Datorită adreselor mai mari de 128 de biți IPv6, sistemul de numerație hexazecimal este utilizat pentru a simplifica reprezentarea adresei. Adresele IPv6 utilizează coloane pentru intrări separate într-o serie de blocuri hexazecimale de 16 biți.

Fig. 6.32 arată conținutul pachetului numărul 46 din captura Wireshark. Pachetul conține mesajul inițial al TCP 3-way handshake dintre hostul IPv6 și serverul IPv6. De observat valorile secțiunii de header extins IPv4. De văzul faptul că acesta este un pachet TCP și nu conține nici-o altă informație în spatele secțiunii TCP.

Fig. 6.33 arată conținutul pachetului numărul 49. Pachetul conține mesajul inițial Hyper Text Transfer Protocol (HTTP) **GET** către server. De văzul faptul că este un pachet HTTP și că acesta conține acum informații în spatele secțiunii TCP.

Fig. 6.34 arată conținutul pachetului 1. Pachetul este un mesaj ICMPv6 Neighbor Solicitation. De observat faptul că nu există nici-o informație TCP sau UDP.

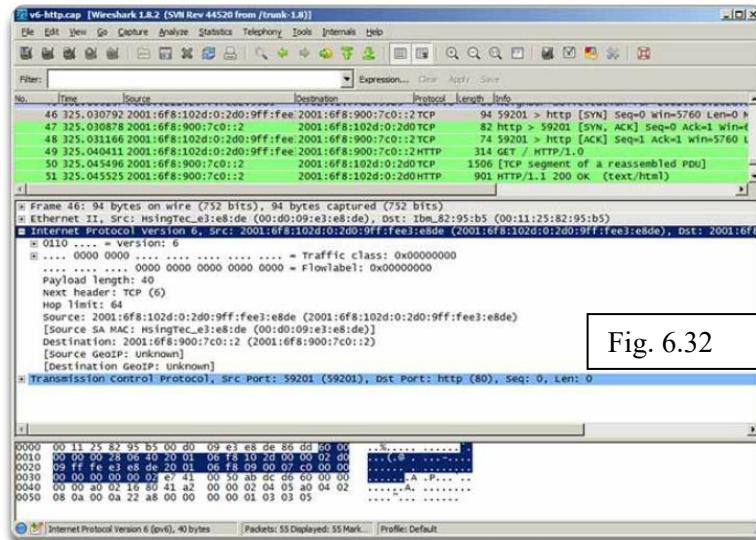


Fig. 6.32

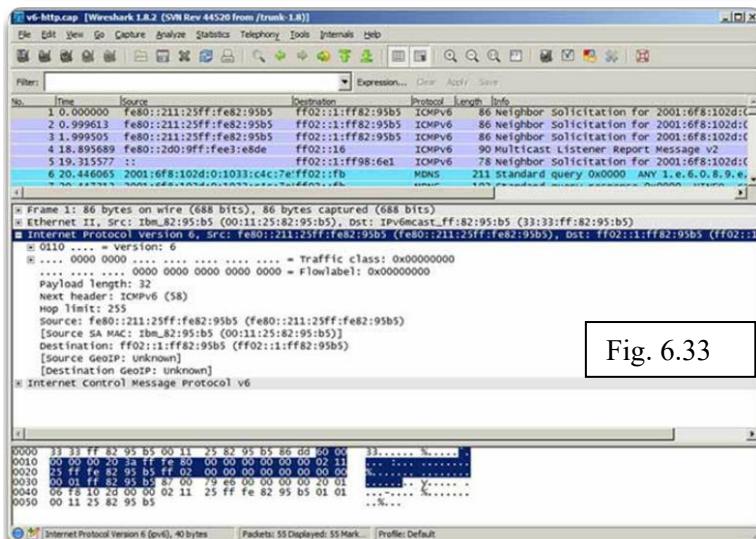


Fig. 6.33

Version	Payload Length
Is always set to 0110	Identifies the packet fragment size
Traffic Class	Next Header
Classifies packets for congestion control	Identifies the application type to the upper-layer protocol
Flow Label	Hop Limit
Can be set to use the same pathway flow so that packets are not reordered upon delivery	When this value reaches 0, the sender is notified that the packet was not delivered

Fig. 6.34. Câmpurile Header IPv6.

6.1.4 Rutarea – Cum rutează un Host

Un alt rol al nivelului rețea este de a direcționa pachete între hosturi. Un host poate trimite un pachet către:

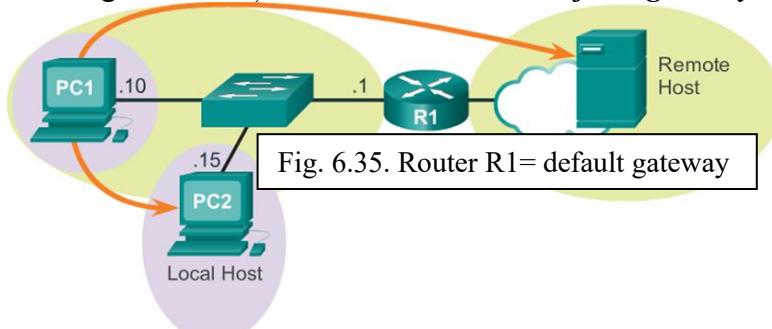
- **El însuși** – Este o adresă IP specială 127.0.0.1, numită și interfață loopback. Această adresă loopback este asignată automat unui host atunci când rulează TCP/IP. Abilitatea unui host de a-și trimite pachete lui însuși folosind funcționalitatea rețelei este utilă pentru scopuri de testare. Orice IP din rețeaua 127.0.0.0/8 se referă la un host local.
- **Host local** – Acesta este un host din aceeași rețea cu hostul expeditor. Hosturile împart aceeași adresă de rețea.
- **Host de la distanță** – Acesta este un host aflat într-o rețea de la distanță. Hosturile nu împart aceeași adresă de rețea.

Dacă un pachet este destinat unui host local sau unui host de la distanță este determinat de combinația de adresă IP și masca de rețea a dispozitivului sursă comparată cu adresa IP și masca de rețea a dispozitivului destinație.

Într-o rețea de domiciliu sau de afaceri ar putea exista mai multe dispozitive wireless sau cablate interconectate împreună prin intermediul unui dispozitiv intermediar cum ar fi un switch LAN sau și un WAP. Acest dispozitiv intermediar oferă interconexiuni între hosturile locale din rețeaua locală. Hosturile locale se pot găsi unele pe celelalte și împart informații fără nevoie altor dispozitive suplimentare. Dacă un host trimite un pachet la un dispozitiv conFig.t cu un IP de rețea ca cel al hostului, pachetul este trimis direct pe interfața hostului, prin intermediul dispozitivului intermediar, la dispozitivul destinație.

Desigur, în cele mai multe situații noi vrem ca dispozitivele noastre să fie capabile să comunice în afara segmentului local de rețea: cu alte locuințe, yone de afaceri și Internet. Dispozitivele aflate în afara segmentului local de rețea sunt numite hosturi de la distanță. Atunci când un dispozitiv sursă trimite un pachet la un host de la distanță, este necesar ajutorul routerelor și al rutării.

Definiție. *Rutarea este procesul de identificare a celei mai bune căi spre destinație.* Routerul conectat la un segment de rețea local este referit ca **default gateway**.



Default gateway este dispozitivul ce rutează traficul de la rețeaua locală la dispozitivele aflate în rețele de la distanță. Într-un mediu de locuință sau de afaceri mici, default gateway este adesea utilizat pentru a conecta rețeaua locală la Internet.

Dacă un host transmite un pachet la un dispozitiv cu un IP de rețea diferit, hostul trebuie să transmită pachetul prin intermediul dispozitivului intermediar la default gateway. Acest lucru se întâmplă deoarece un host nu menține informații de rutare, din exteriorul rețelei locale, pentru a ajunge la destinații de la distanță. Default gateway face acest lucru. Default gateway, care este de cele mai multe ori un router, menține o tabelă de rutare. O tabelă de rutare este un fișier de date din RAM utilizat pentru a stoca informații despre rețelele direct conectate, cât și despre

rețelele de la distanță învățate de către dispozitiv. Un router utilizează informații din tabela de rutare pentru a determina calea cea mai bună spre destinații.

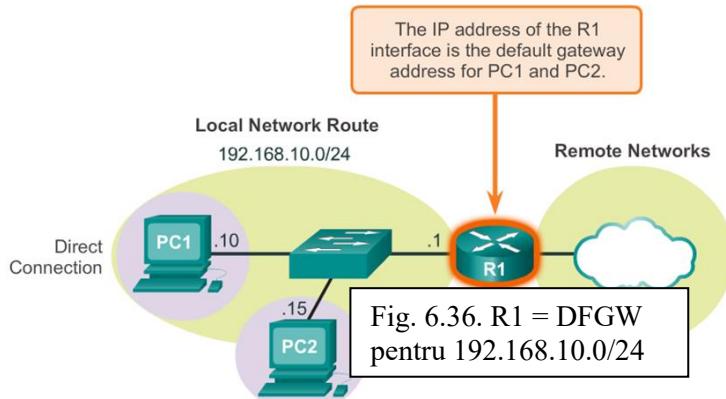
Cum ține evidența un host dacă să transmită sau nu pachetele spre Default gateway ?

Hosturile trebuie să-și mențină propriile tabele de rutare locale pentru a se asigura de faptul că pachetele de la nivelul rețea sunt direcționate la rețelele destinație corecte. Tabela locală a unui host conține de obicei:

- **Conexiunea directă** – Aceasta este o ruta cu interfață loopback (127.0.0.1).
- **Ruta de rețea locală** – Rețeaua la care este conectat hostul este populată automat în tabela de rutare a hostului.
- **Ruta default locală** – Ruta default reprezintă ruta pe care trebuie să o ia pachetele pentru a ajunge la toate adresele de rețea de la distanță. Ruta default este creată atunci când o adresă de default gateway este prezentă pe host. Adresa de default gateway este adresa IP a interfeței de rețea a routerului conectat la rețeaua locală. Adresa de default gateway poate fi configurată pe host manual sau învățată dinamic.

Este important de ținut minte că ruta default, default gateway, este utilizată numai atunci când un host trebuie să transmită pachete la o rețea de la distanță. Nu este necesară, nici măcar nu trebuie să fie configurată, dacă pachetele transmise sunt destinate în rețeaua locală.

De exemplu, considerăm o imprimantă/scanner de rețea. Dacă imprimanta de rețea are o adresă IP și o mască de rețea configurată, hosturile locale pot trimite documente către imprimantă pentru a fi printate. În plus, imprimanta poate expedia documente scanate la orice host local. Atât timp cât imprimanta este utilizată local, o adresă de default gateway nu este necesară. De fapt, prin neconfigurarea unei adrese de default gateway pe imprimantă, este interzis accesul la Internet, ce ar putea reprezenta o alegere înteleaptă de securitate. Nici-un acces la Internet înseamnă nici-un risc de securitate. Cu toate că dispozitivele, precum imprimantele, ar putea oferi capacitatea de efectuare a unor actualizări automate prin intermediul Internetului, este mai ușor, de obicei și mai sigură, efectuarea respectivelor actualizări printr-un upload local de la un host local securizat, cum ar fi un PC.



Pe un host Windows, comenziile **route print** sau **netstat -r** pot fi folosite pentru a afișa tabela de rutare a hostului. Ambele comenzi generează același output. Outputul ar putea părea copleșitor la început, însă este foarte ușor de înțeles.

Introducând comanda **route print** sau comanda **netstat -r**, se afișează trei secțiuni legate de conexiunile curente de rețea TCP/IP:

➤ *Lista de interfețe* – Listă adresele Media Access Control (MAC) și numărul de interfață asignat al fiecărei interfețe de rețea de pe host, inclusiv Ethernet, Wi-Fi și adaptorul Bluetooth.

➤ *Tabela de rutare IPv4* – Listă toate routerele IPv4 cunoscute, inclusiv conexiunile directe, rețeaua locală și ruturile implicate locale.

➤ *Tabela de rutare IPv6* - Listeață toate routerele IPv6 cunoscute, inclusiv conexiunile directe, rețeaua locală și rutele implicate locale.

Notă: Outputul comenzi variază, în funcție de modul în care hostul este config.t și de tipurile de interfețe existente.

Fig. arată secțiunea din tabela de rutare IPv4 a outputului. De remarcat faptul că outputul este divizat în 5 coloane ce identifică:

- **Destinatiile de rețea** – Listeață toate rețelele ce pot fi accesate.
- **Masca de rețea** – Listeață o mască de rețea ce informează hostul de modul în care să determine portiunile de rețea și de host ale adresei IP.
- **Gateway** – Listeață adresa utilizată de către computerul local pentru a ajunge la o destinație de rețea de la distanță. Dacă o destinație este accesată direct, va fi afișat “on-link” în această coloană.
- **Interfață** – Listeață adresa interfeței fizice folosită pentru a transmite pachetul la gateway, ce este folosit pentru a ajunge la rețeaua destinație.
- **Metrica** – Listeață costul fiecărei rute și este folosită pentru a determina cea mai bună rută spre o destinație.



C:\Users\PC1>netstat -r					
<Output omitted>					
IPv4 Route Table					
=====					
Active Routes:					
Network Destination	Netmask	Gateway	Interface	Metric	
0.0.0.0	0.0.0.0	192.168.10.1	192.168.10.10	25	
127.0.0.0	255.0.0.0	On-link	127.0.0.1	306	
127.0.0.1	255.255.255.255	On-link	127.0.0.1	306	
127.255.255.255	255.255.255.255	On-link	127.0.0.1	306	
192.168.10.0	255.255.255.0	On-link	192.168.10.10	281	
192.168.10.10	255.255.255.255	On-link	192.168.10.10	281	
192.168.10.255	255.255.255.255	On-link	192.168.10.10	281	
224.0.0.0	240.0.0.0	On-link	127.0.0.1	306	
224.0.0.0	240.0.0.0	On-link	192.168.10.10	281	
255.255.255.255	255.255.255.255	On-link	127.0.0.1	306	
255.255.255.255	255.255.255.255	On-link	192.168.10.10	281	

<Output omitted>

Fig. 6.37. Comanda **netstat**

Pentru simplificarea outputului, rețelele destinație pot fi grupate în cinci secțiuni identificate de zonele evidențiate din Fig. :

0.0.0.0 – Ruta default locală; toate pachetele cu destinații ce nu corespund nici unei alte adrese specificate în tabela de rutare sunt transmise la gateway. Prin urmare, toate rutele ce nu se potrivesc nici unei destinații sunt transmise la gateway cu adresa IP 192.168.10.1 (R1) ce are ieșire prin interfață cu adresa IP 192.168.10.10. De remarcat faptul că adresa destinație finală specificată în pachet nu se schimbă; hostul știe numai să transmită pachetul la gateway pentru procesări ulterioare.

127.0.0.0 – 127.255.255.255 – Aceste adrese loopback se referă la conexiunea directă și oferă servicii pe hostul local.

192.168.10.0 - 192.168.10.255 – Aceste adrese se referă la host și la rețeaua locală. Toate pachetele cu adrese destinație ce se află în această categorie vor ieși pe interfață 192.168.10.10.

192.168.10.0 – Reprezintă toate computerele din rețeaua 192.168.10.x.

192.168.10.10 – Adresa hostului local.

192.168.10.255 - Adresa de broadcast; trimite mesaje tururilor din ruta de rețea locală.

224.0.0.0 – Acestea sunt adrese speciale multicast de clasă D rezervate pentru utilizare prin interfața loopback (127.0.0.1) sau adresa IP de host (192.168.10.10).

255.255.255.255 – Ultimile două adrese reprezintă valorile de adresă IP broadcast limitate pentru utilizarea prin interfață loopback (127.0.0.1) sau adresa IP de host (192.168.10.10). Aceste adrese pot fi folosite pentru găsirea unui server DHCP, înainte ca IP local să fie determinat.

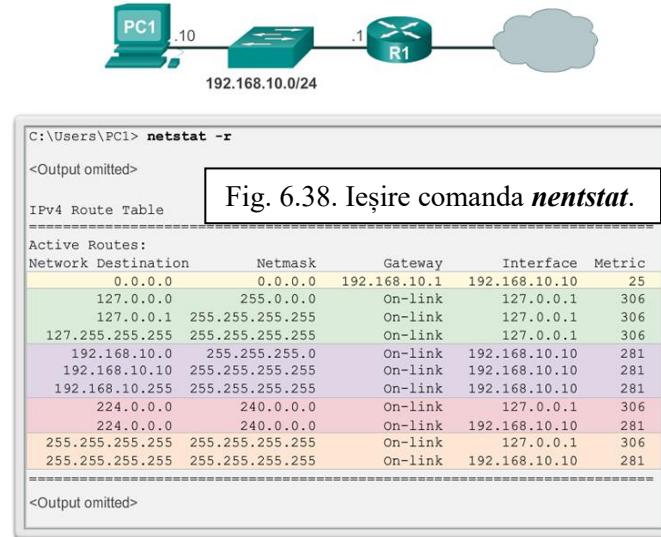


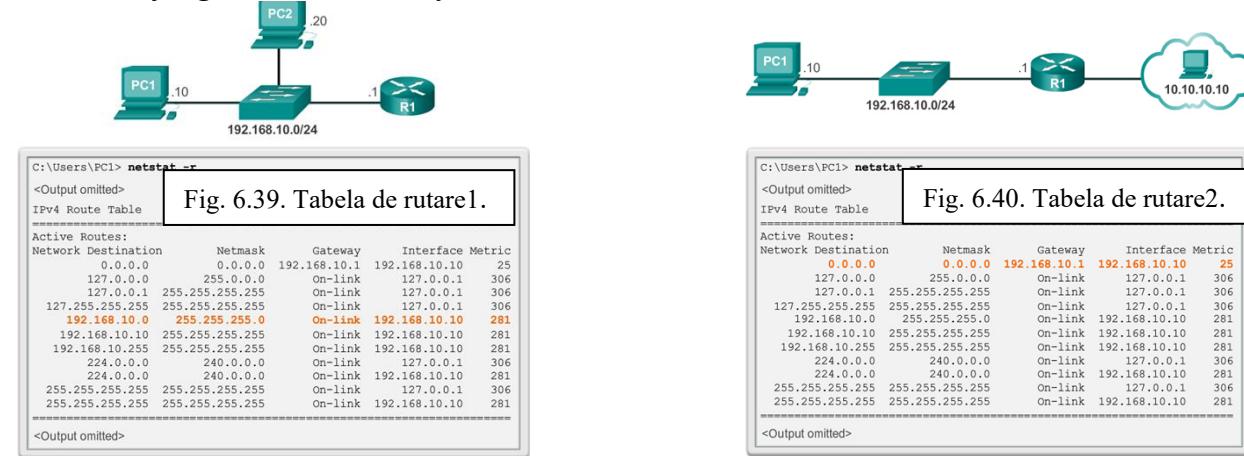
Fig. 6.38. Ieșire comanda **netstat**.

De exemplu, dacă PC1 vrea să trimită un pachet la 192.168.10.20, va urma pașii:

- 1) *Va consulta tabela de rutare IPv4.*
- 2) *Va potrivi adresa IP destinație cu intrarea 192.168.10.0 Network Destination pentru a descoperi că hostul se află în aceeași rețea (On-link).*
- 3) *PC1 va transmite pachetul la destinația finală folosind interfața sa locală (192.168.10.10).*

Dacă PC1 vrea să transmită un pachet la un host de la distanță aflat la 10.10.10.10, va urma pașii:

- i. *Va consulta tabela de rutare IPv4.*
- ii. *Va remarcă faptul că nu există nici-o potrivire pentru adresa IP destinație.*
- iii. *Va alege ruta default locală (0.0.0.0) pentru a descoperi faptul că trebuie să transmită pachetul la adresa de gateway 192.168.10.1.*
- iv. *PC1 transmite apoi pachetul la gateway pentru utilizarea interfeței locale (192.168.10.10). Dispozitivul de gateway determină următoarea cale pentru ca pachetul să ajungă la destinația sa finală 10.10.10.10.*



Ieșirea tabelei de rutare IPv6 diferă în hederele de coloană și format având în vedere adrese IPv6 mai lungi.

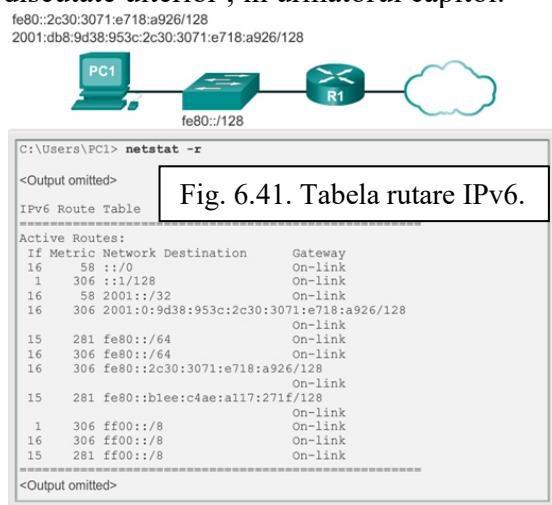
Secțiunea tabelei de rutare IPv6 arată patru coloane ce identifică:

- If** – Listeză numerele de interfețe de la secțiunea Interface List a comenzi **netstat -r**. Numerele de interfață corespund cu interfață de rețea de pe host, inclusiv Ethernet, Wi-Fi, și adaptorul Bluetooth.
- Metrica** – Listeză costul fiecărei rute spre o destinație. Numerele mai mici indică rutele preferate.
- Destinația de rețea** – Listeză rețelele ce pot fi accesate.
- Gateway** – Listeză adresele folosite de către hostul local pentru a transmite pachetele la o destinație de rețea de la distanță. **On-link** indică faptul că hostul este conectat la aceasta.

De exemplu, în Fig. este arătată secțiunea de IPv6 Route generată de către comanda **netstat -r** ce dezvăluie următoarele destinații :

- **::/0** – IPv6 echivalent pentru ruta default locală.
- **::1/128** – Este echivalent adreselor loopback IPv4 și oferă servicii pe hostul local.
- **2001::/32** – Aceasta este prefixul de rețea unicast globală.
- **2001:0:9d38:953c:2c30:3071:e718:a926/128** - Aceasta este adresa IPv6 unicast globală a computerului local.
- **fe80::/64** - Aceasta este adresa de rețea pentru rutare locală și reprezintă toate computerele din rețeaua IPv6 cu legătură locală.
- **fe80::2c30:3071:e718:a926/128** - Aceasta este legătura locală cu adresa IPv6 a computerului local.
- **ff00::/8** - Acestea sunt adrese multicast rezervate de clasă D echivalente IPv4 224.x.x.x.

Notă : Interfețele din IPv6 au de obicei două adrese IPv6: o adresă de legătură locală și o adresă unicast globală. De asemenea, remarcăm faptul că nu există nici-o adresă de broadcast în IPv6. Adresele IPv6 vor fi discutate ulterior , în următorul capitol.



6.1.5 Tabela de rutare a Routerului

Atunci când un host trimite un pachet la un alt host, va folosi propria tabelă de rutare pentru a determina unde să trimită pachetul. Dacă destinația se află la o rețea de la distanță, pachetul este trimis la adresa dispozitivului gateway.

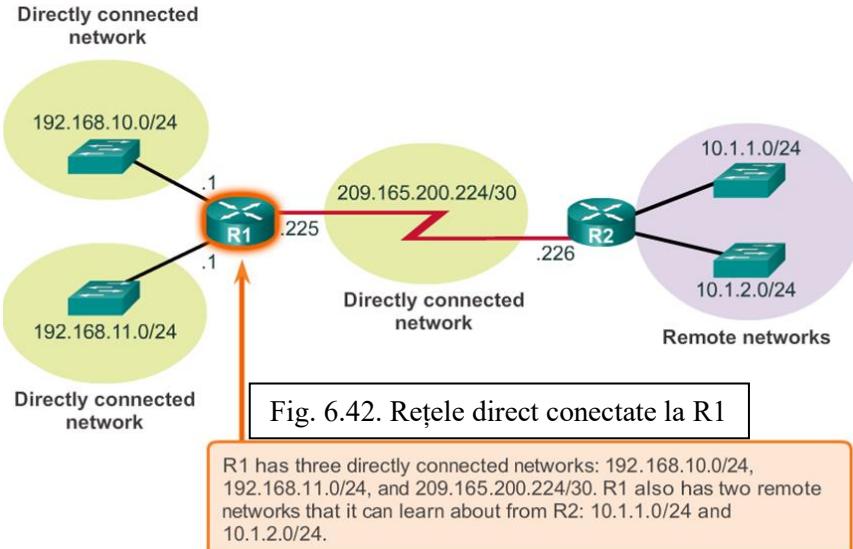
Întrebare : Ce se întâmplă atunci când un pachet ajunge la o interfață de pe router ?

Răspuns: Routerul se uită la tabela sa de rutare pentru a determina unde să trimită pachetele.

Tabela de rutare a unui router conține informații despre:

Rutele direct conectate – Aceste rute sunt de la interfețele routerului active. Routerul adaugă o rută direct conectată atunci când o interfață este config.ată cu o adresă IP și este activată. Fiecare dintre interfețele routerului este conectată la un segment de rețea diferit. Routerele mențin informații despre segmentele de rețea la care sunt conectate în tabela de rutare.

Rutele de la distanță – Aceste routere sunt ale rețelelor de la distanță conectate la alte routere. Rutele de la aceste rețele pot fi config.ate fie manual pe routerul local de către administratorul de rețea, fie dinamic prin activarea pe routerul local schimbul de informații cu alte routere, prin folosirea protoocoalelor de rutare dinamice.



O tabelă de rutare a unui host include numai informații despre rețelele direct conectate. Un host necesită un default gateway pentru a transmite pachetele la o destinație de la distanță. Tabela de rutare a unui router conține informații similare, dar poate identifica și rețelele de la distanță specifice.

Tabela de rutare a unui router este similară cu tabela de rutare a unui host. Ambele identifică:

- Rețeaua destinație.
- Metrica asociată cu rețeaua destinație.
- Gateway pentru a ajunge la rețeaua destinație.

Pe un router Cisco cu IOS, comanda **show ip route** poate fi folosită pentru a afișa tabela de rutare a routerului. Un router oferă de asemenea informații suplimentare de rutare, cum ar fi modul în care a fost învățată o rută, când a fost actualizată ultima oară și ce interfață specifică este utilizată pentru a ajunge la o destinație predefinită.

Atunci când un pachet ajunge la interfață routerului, routerul examinează hederul pachetului pentru a determina rețeaua destinație. Dacă rețeaua destinație se potrivește cu o rută din tabela de rutare, routerul transmite pachetul folosind informațiile specificate în tabela de rutare. Dacă există două sau mai multe rute spre aceeași destinație, metrica este folosită pentru a decide ce rută apare în tabela de rutare.

Fig. arată tabela de rutare a R1 dintr-o rețea simplă. Spre deosebire de tabela de rutare a unui host, nu există titluri de coloane ce identifică informațiile conținute într-o intrare din tabela de rutare. Prin urmare, este important să învățăm înțelesul tipurilor diferite de informații incluse în fiecare intrare.

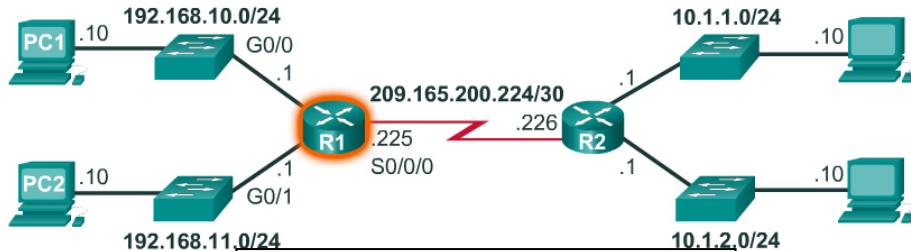


Fig. 6.43. Comanda *show ip route*

```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
      IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route
Gateway of last resort is not set
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
D        10.1.1.0/24 [90/2170112] via 209.165.200.226, 00:00:05,
          Serial0/0/0
          10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05

Serial0/0/0
D        10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05,
          Serial0/0/0
          192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C          192.168.10.0/24 is directly connected, GigabitEthernet0/0
L          192.168.10.1/32 is directly connected, GigabitEthernet0/0
          192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C          192.168.11.0/24 is directly connected, GigabitEthernet0/1
L          192.168.11.1/32 is directly connected, GigabitEthernet0/1
          209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks
C          209.165.200.224/30 is directly connected, Serial0/0/0
L          209.165.200.225/32 is directly connected, Serial0/0/0
R1#
```

Două intrări din tabela de rutare sunt create automat atunci când o interfață de rutare activă este config. cu o adresă IP și masca de rețea. Fig. arată intrările din tabela de rutare de pe R1 pentru rețeaua direct conectată 192.168.10.0. Aceste intrări au fost adăugate automat în tabela de rutare atunci când interfața GigabitEthernet 0/0 a fost config. și activată. Intrările conțin următoarele informații:

Ruta Sursă - este etichetată "A" în Fig. . Identifică modul în care ruta a fost învățată. Interfețele direct conectate au două coduri pentru sursa rutei.

C - Identifică o rețea direct conectată. Rețele direct conectate sunt create automat atunci când o interfață este config. cu o adresă IP și este activată.

L - Identifică faptul că este o rută de legătură locală. Rutele cu legături locale sunt create automat atunci când o interfață este config. cu o adresă IP și este activată.

Rețeaua Destinație – Rețeaua destinație este etichetată "B" în Fig. . Identifică adresa rețelei de la distanță.

Interfața de ieșire – Interfața de ieșire este etichetată “C” în Fig. . Identifică interfața de ieșire utilizată atunci când sunt transmise pachetele la rețeaua destinație.

Un router în mod normal are mai multe interfețe configurate. Tabela de rutare stocăreză informații despre rutele direct conectate și despre cele aflate la distanță. Sursa rutei identifică modul în care ruta a fost învățată. De exemplu, coduri comune pentru rețelele de la distanță sunt:

S – Identifică faptul că ruta a fost creată manual de către un administrator de rețea pentru a ajunge la o rețea specifică. Este cunoscută ca ruta statică.

D – Identifică faptul că ruta a fost învățată dinamic de la un alt rută ce folosește Enhanced Interior Gateway Routing Protocol (EIGRP).

O - Identifică faptul că ruta a fost învățată dinamic de la un alt rută ce folosește protocolul de rutare Open Shortest Path First (OSPF).

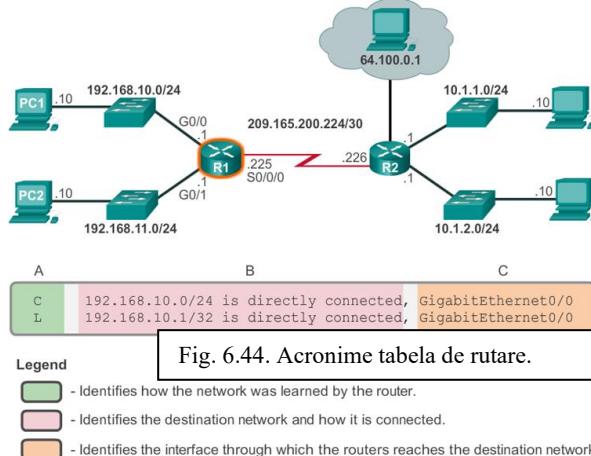


Fig. 6.44. Acronime tabela de rutare.

Fig. afișează o intrare din tabela de rutare a lui R1 pentru ruta de la rețeaua de la distanță 10.1.1.0. Intrarea identifică următoarele informații:

Sursa rutei – identifică modul în care ruta a fost învățată.

Rețeaua destinație – identifică adresa rețelei de la distanță.

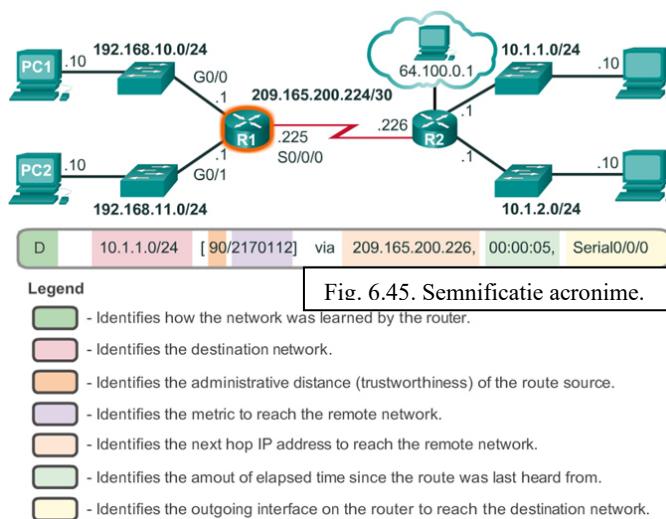
Distanța administrativă – identifică încrederea sursei rutei.

Metrica – identifică valoarea atribuită pentru a ajunge la rețea destinație. Valorile mai mici indică rutele preferate.

Next-hop – identifică adresa IP a routerului următor pentru a transmite pachetul.

Route timestamp – identifică atunci când ruta a fost ultima oară actualizată.

Interfața de ieșire – identifică interfața de ieșire folosită pentru a transmite un pachet spre destinația sa finală.



Un next-hop este adresa dispozitivului ce va procesa pachetul ulterior. Pentru un host dintr-o rețea, adresa de gateway (interfața routerului) este next-hop pentru toate pachetele ce trebuie să fie transmise într-o altă rețea. În tabela de rutare a unui router, fiecare rută spre o rețea de la distanță listează un next-hop.

Atunci când un pachet destinat pentru o rețea de la distanță ajunge la router, routerul potrivește rețeaua destinație cu o rută din tabela de rutare. Dacă găsește o potrivire, routerul transmite pachetul la adresa IP a routerului next-hop folosind interfața identificată de către intrarea rutei.

Un router next-hop este gateway pentru rețelele de la distanță.

De exemplu, în Fig. , un pachet ce ajunge la R1 destinat fie pentru rețeaua 10.1.1.0 , fie pentru 10.1.2.0 este transmis la adresa next-hop 209.165.200.226 folosind interfața Serial 0/0/0.

Rețelele direct conectate la un router nu au adresa next-hop deoarece un router poate transmite pachete direct la hosturile din aceste rețele folosind interfața aleasă.

Pachetele nu pot fi transmise de către router fără o rută spre rețeaua destinație în tabela de rutare. Dacă o rută reprezintă că rețeaua destinație nu există în tabela de rutare, pachetul este aruncat (nu este transmis mai departe).

Însă, la fel cum un host poate utiliza un default gateway să transmită un pachet la o destinație necunoscută, un router poate să fie configurt să utilizeze o rută default statică pentru a crea Gateway of Last Resort. Gateway of Last Resort va fi explicată în detaliu în cursul de Fundamentele Rutării.

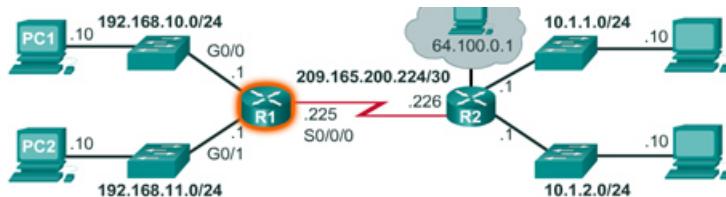


Fig. 6.46. Gateway of Last Resort

```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile,
      B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
      IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set
```

```
Serial0/0/0
D    10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05,
Serial0/0/0
      192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.10.0/24 is directly connected, GigabitEthernet0/0
L      192.168.10.1/32 is directly connected, GigabitEthernet0/0
      192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.11.0/24 is directly connected, GigabitEthernet0/1
L      192.168.11.1/32 is directly connected, GigabitEthernet0/1
      209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks
C      209.165.200.224/30 is directly connected, Serial0/0/0
L      209.165.200.225/32 is directly connected, Serial0/0/0
R1#
```

Presupunem că PC1 cu adresa IP 192.168.10.10 vrea să transmită un pachet la un alt host din aceeași rețea. PC1 va verifica tabela de rutare IPv4 bazându-se pe adresa IP destinație. Apoi, PC1 va descoperi că hostul se află în aceeași rețea și va transmite pachetul pe interfața sa (**on-link**).

Notă: R1 nu este implicat în transferul pachetului. Dacă PC1 trimită un pachet la orice altă rețea decât cea locală, va folosi serviciile routerului R1 și va transmite pachetul la ruta default locală proprie (192.168.10.1).

Următoarele exemple ilustrează modul în care un host și un router iau decizii de rutare a pachetului prin consultarea tabelelor de rutare respective:

Exemplul 1. PC1 vrea să verifice conectivitatea la propriul default gateway local 192.168.10.1 (interfața routerului):

1. PC1 consultă tabela de rutare IPv4 bazându-se pe adresa IP destinație.
2. PC1 va descoperi că hostul se află în aceeași rețea și va transmite pachetul **ping** pe interfața sa (**on-link**).
3. R1 primește pachetul pe interfața Gigabit Ethernet 0/0 (G0/0) și se uită la adresa IP destinație.
4. R1 își consultă tabela de rutare.
5. R1 potrivește adresa IP destinație cu intrarea din tabela de rutare L 192.168.10.1/32 și descoperă că această rută este spre interfața locală proprie.
6. R1 deschide restul pachetului IP și răspunde în consecință.

Exemplul 2: PC1 vrea să transmită un pachet la PC2 (192.168.11.10):

- A. PC1 consultă tabela de rutare IPv4 și descoperă că nu există nici-o potrivire.
- B. PC1 folosește, în consecință, rețeaua 0.0.0.0 și trimită pachetul folosind ruta default locală (192.168.10.1).
- C. R1 primește pachetul pe interfața Gigabit Ethernet 0/0 (G0/0) proprie și se uită la adresa IP destinație (192.168.11.10).
- D. R1 consultă tabela proprie de rutare și potrivește adresa IP destinație cu intrarea din tabela de rutare C 192.168.11.0/24.
- E. R1 trimite pachetul mai departe pe interfața direct conectată Gigabit Ethernet 0/1.
- F. PC2 primește pachetul și își consultă tabela de rutare IPv4.
- G. PC2 descoperă că pachetul îi este adresat, îl deschide și răspunde în consecință.

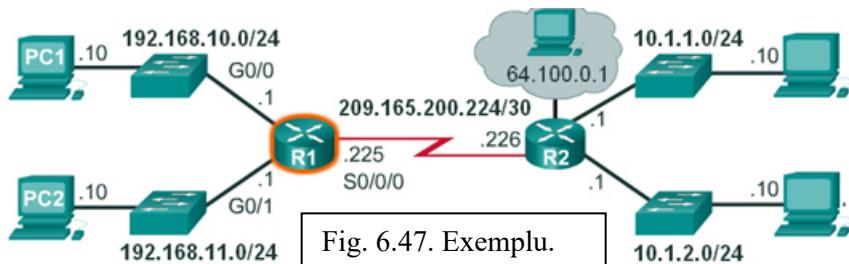
Exemplul 3: PC1 vrea să transmită un pachet la adresa 209.165.200.226:

- i. PC1 consultă tabela de rutare IPv4 și descoperă că nu există nici-o potrivire.
- ii. PC1 folosește, în consecință, rețeaua 0.0.0.0 și trimită pachetul folosind ruta default locală (192.168.10.1).
- iii. R1 primește pachetul pe interfața Gigabit Ethernet 0/0 (G0/0) proprie și se uită la adresa IP destinație (209.165.200.226).
- iv. R1 consultă tabela proprie de rutare și potrivește adresa IP destinație cu intrarea din tabela de rutare C 209.165.200.224/30
- v. R1 transmite pachetul prin interfața direct conectată Serial 0/0/0.

Exemplul 4: PC1 vrea să transmită un pachet la hostul cu adresa IP 10.1.1.10:

- a) PC1 consultă tabela de rutare IPv4 și descoperă că nu există nici-o potrivire.
- b) PC1 folosește, în consecință, rețeaua 0.0.0.0 și trimită pachetul folosind ruta default locală (192.168.10.1).
- c) R1 primește pachetul pe interfața Gigabit Ethernet 0/0 (G0/0) proprie și se uită la adresa IP destinație (10.1.1.10).

- d) R1 consultă tabela proprie de rutare și potrivește adresa IP destinație cu intrarea din tabela de rutare D 10.1.1.0/24.
- e) R1 descoperă că trebuie să transmită pachetul la adresa next-hop 209.165.200.226.
- f) R1 consultă din nou tabela proprie de rutare și potrivește adresa IP destinație cu intrarea din tabela de rutare 209.165.200.224/30.
- g) R1 transmite pachetul prin interfața direct conectată Serial 0/0/0.



```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile,
      B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
      IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is 209.165.200.226 to network 0.0.0.0
```

```
Serial0/0/0
D    10.1.2.0/24 [90/2170112] via 209.165.200.226, 01:13:55,
     Serial0/0/0
     192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.10.0/24 is directly connected, GigabitEthernet0/0
L      192.168.10.1/32 is directly connected, GigabitEthernet0/0
     192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.11.0/24 is directly connected, GigabitEthernet0/1
L      192.168.11.1/32 is directly connected, GigabitEthernet0/1
     209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks
C      209.165.200.224/30 is directly connected, Serial0/0/0
L      209.165.200.225/32 is directly connected, Serial0/0/0
S*    0.0.0.0/0 [1/0] via 209.165.200.226
```

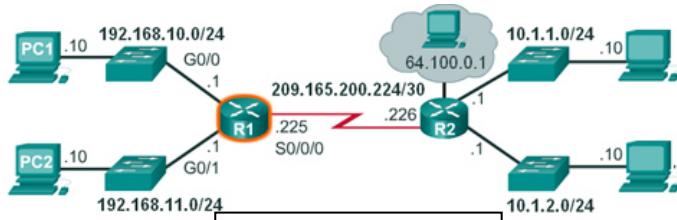


Fig. 6.48. Exemplu

```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile,
      B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
      IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is 209.165.200.226 to network 0.0.0.0

          10.0.0.0/0 is variably subnetted, 2 subnets, 2 masks

```

```
          Serial0/0/0
D        10.1.2.0/24 [90/2170112] via 209.165.200.226, 01:13:55,
          Serial0/0/0
          192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C        192.168.10.0/24 is directly connected, GigabitEthernet0/0
L        192.168.10.1/32 is directly connected, GigabitEthernet0/0
          192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C        192.168.11.0/24 is directly connected, GigabitEthernet0/1
L        192.168.11.1/32 is directly connected, GigabitEthernet0/1
          209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks
C        209.165.200.224/30 is directly connected, Serial0/0/0
L        209.165.200.225/32 is directly connected, Serial0/0/0
S*       0.0.0.0/0 [1/0] via 209.165.200.226
```

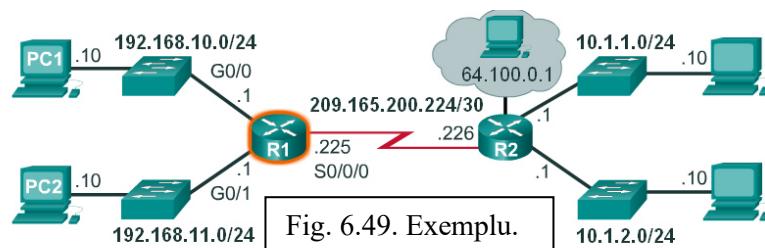


Fig. 6.49. Exemplu.

```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile,
      B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
      IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is 209.165.200.226 to network 0.0.0.0

          10.0.0.0/0 is variably subnetted, 2 subnets, 2 masks
```

```

D      10.1.1.0/24 [90/2170112] via 209.165.200.226, 01:13:55,
      Serial0/0/0
D      10.1.2.0/24 [90/2170112] via 209.165.200.226, 01:13:55,
      Serial0/0/0
      192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.10.0/24 is directly connected, GigabitEthernet0/0
L      192.168.10.1/32 is directly connected, GigabitEthernet0/0
      192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.11.0/24 is directly connected, GigabitEthernet0/1
L      192.168.11.1/32 is directly connected, GigabitEthernet0/1
      209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks
C      209.165.200.224/30 is directly connected, Serial0/0/0
L      209.165.200.225/32 is directly connected, Serial0/0/0
S*    0.0.0.0/0 [1/0] via 209.165.200.226

```

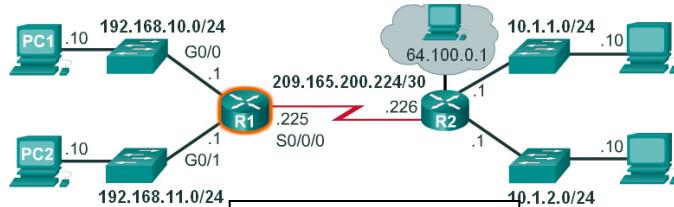


Fig. 6.50. Exemplu.

```

R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile,
       B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
       IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 209.165.200.226 to network 0.0.0.0

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks

```

```

      10.1.1.0/24 [90/2170112] via 209.165.200.226, 01:13:55,
      Serial0/0/0
D      10.1.2.0/24 [90/2170112] via 209.165.200.226, 01:13:55,
      Serial0/0/0
      192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.10.0/24 is directly connected, GigabitEthernet0/0
L      192.168.10.1/32 is directly connected, GigabitEthernet0/0
      192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.11.0/24 is directly connected, GigabitEthernet0/1
L      192.168.11.1/32 is directly connected, GigabitEthernet0/1
      209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks

```

```

Serial0/0/0
D      10.1.2.0/24 [90/2170112] via 209.165.200.226, 01:13:55,
      Serial0/0/0
      192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.10.0/24 is directly connected, GigabitEthernet0/0
L      192.168.10.1/32 is directly connected, GigabitEthernet0/0
      192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C      192.168.11.0/24 is directly connected, GigabitEthernet0/1
L      192.168.11.1/32 is directly connected, GigabitEthernet0/1
      209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks
C      209.165.200.224/30 is directly connected, Serial0/0/0
L      209.165.200.225/32 is directly connected, Serial0/0/0
S*    0.0.0.0/0 [1/0] via 209.165.200.226

```

6.2 Routerele – Anatomia unui Router

Există mai multe tipuri de routere de infrastrucțură disponibile. Rutele Cisco sunt concepute pentru a corespunde nevoilor:

Branch – Teleworkers, afaceri mici și filiale de dimensiune medie. Includ Cisco 800, 1900, 2900 și 3900 Integrated Series Routers (ISR) G2 (a 2-a generație).

WAN – Afaceri, organizații și întreprinderi de dimensiune mare. Includ Cisco Catalyst 6500 Series Switches și Cisco Aggregation Service Router (ASR) 1000.

Furnizor de serviciu – Furnizori de servicii mari. Includ Cisco ASR 1000, Cisco ASR 9000, Cisco XR 12000, Cisco CRS-3 Carrier Routing System și 7600 Series routers.

Certificările CCNA își îndreaptă atenția spre familia de routere branch. Fig. arată familia de routere Cisco 1900, 2900 și 3900 ISR G2.

Indiferent de funcționalitatea lor, complexitatea sau dimensiunea lor, toate modelele de routere sunt în esență computere. Precum computerele, tabletele și dispozitivele smart, rutele necesită:

- *Operating systems (OS).*
- *Central processing units (CPU).*
- *Random-access memory (RAM).*
- *Read-only memory (ROM).*

Un router are de asemenea memorie specială ce include Flash și nonvolatile random-access memory (NVRAM).

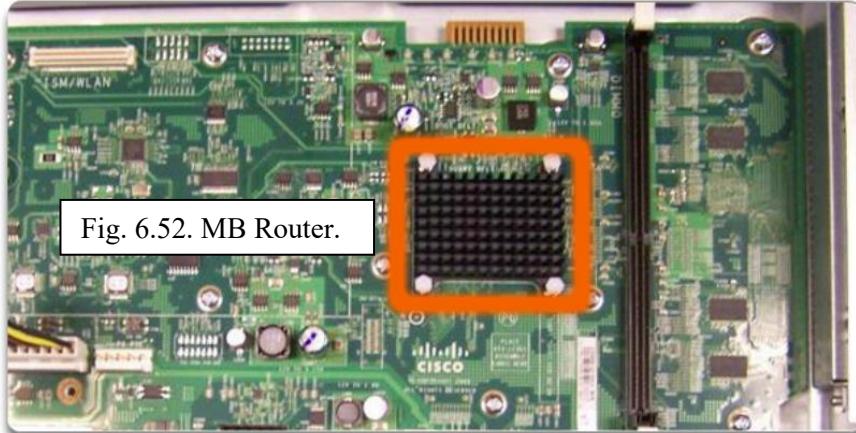
Fig. 6.51 Router.



Ca toate computerele, tabletele și dispozitivele smart, dispozitivele Cisco necesită un CPU pentru a executa instrucțiuni OS, cum ar fi inițializarea sistemului, funcții de rutare și funcții de switch.

CPU necesită un OS pentru a oferi funcții de rutare și switch. Cisco Internetwork Operating System (IOS) este softwareul de sistem folosit de cele mai multe dispozitive Cisco indiferent de mărime sau tipul de dispozitiv. Este folosit de routere, switchuri LAN, puncte de acces wireless mici, routere mari cu zeci de interfețe și multe alte dispozitive.

Fig. 6.52. MB Router.



Un router are acces la patru tipuri de memorie: RAM, ROM, NVRAM și Flash.

RAM – este folosit pentru a stoca aplicații și procesă diferite servicii, cum ar fi:

- IOS - IOSul este copiat în RAM în timpul procesului de bootup.
- Running configuration file - Acesta este fișierul de config.re ce stocă comenzile de config.re folosite actual de IOS. Este cunoscut de asemenea ca running-config.
- IP routing table - Acest fișier stocă informații despre rețelele direct conectate și despre cele de la distanță. Este utilizat pentru a determina cea mai bună cale pentru a transmite pachetele.
- ARP cache - Acest cache conține mapările adresa IP - adresa MAC, similar cu Address Resolution Protocol (ARP) cache pe un PC. ARP cache este utilizat pe routere ce au interfețe LAN, cum ar fi interfețe Ethernet.
- Packet buffer - Pachetele sunt stocate temporar într-un buffer atunci când sunt primite pe o interfață sau înaintea ieșirii lor pe o interfață.

Ca și computerele, routerele Cisco folosesc dynamic random-access memory (DRAM). DRAM este un tip comun de memorie RAM ce stocă instrucțiunile și datele necesare pentru execuția lor de către CPU. Spre deosebire de ROM, RAM este o memorie volată și necesită alimentare de curent continuă pentru a-și menține informațiile. Pierde tot conținutul său atunci când routerul nu mai este alimentat sau este restartat.

Implicit, routerele 1941 au 512MB de DRAM sudată pe placa de sistem principală (onboard) și un slot de modul de memorie in-line (DIMM) pentru upgradeuri de memorie de până la 2.0 GB suplimentari. Modelele Cisco 2901, 2911 și 2921 au 512 MB de onboard DRAM. De reținut faptul că prima generație de ISRuri și alte routere Cisco mai vechi nu aveau onboard RAM.

ROM – Routerele Cisco folosesc ROM pentru a stoca:

- Bootup instructions - Oferă instrucțiunile de startup.
- Basic diagnostic software - Efectuează power-on self-test (**POST**) pe toate componente.
- Limited IOS - Oferă o versiune limitată de rezervă a OS, în cazul în care routerul nu poate încărca întreagul IOS.
- ROM este firmware incorporat pe un circuit integrat din interiorul routerului și nu își pierde conținutul atunci când routerul este restartat sau nu mai este alimentat.

NVRAM – NVRAM este folosit de către Cisco IOS ca un depozit permanent pentru fișierul de config.re startup(startup-config). Ca și ROM, NVRAM nu își pierde conținutul atunci când routerul nu mai este alimentat.

Flash Memory – Memoria Flash este o memorie computer non-volată folosită ca un depozit permanent pentru IOS și alte fișiere legate de sistem. IOS este copiat din flash în RAM în timpul procesului de bootup.

Routerele Cisco 1941 au două sloturi externe Compact Flash. Fiecare slot poate suportă densități de stocare de mare viteză ce pot fi upgradeate la 4Gb în densitate.

Memory	Volatile / Non-Volatile	Stores
RAM	Volatile	<ul style="list-style-type: none">• Running IOS• Running configuration file• IP routing and ARP tables• Packet buffer
ROM	Non-Volatile	<ul style="list-style-type: none">• Bootup instructions• Basic diagnostic software• Limited IOS
NVRAM	Non-Volatile	<ul style="list-style-type: none">• Startup configuration file
Flash	Non-Volatile	<ul style="list-style-type: none">• IOS• Other system files

Deși există mai multe tipuri și modele diferite de routere, fiecar router are aceleasi componente hardware generale.

Fig. arată interiorul unui router Cisco 1841, prima generație ISR.

De remarcat faptul că Fig. evidențiază și alte componente dintr-un router, cum ar fi sursa de alimentare, ventilatorul de răcire, scuturi termice și un modul AIM, care nu reprezintă scopul acestui capitol.

Notă: Un profesionist de rețea ar trebui să fie familiar și să înțeleagă funcțiile principalelor componente interne ale unui router, decât să știe locația exactă a acestor componente în interiorul unui router specific. În funcție de model, aceste componente sunt localizate în diferite locuri din interiorul routerului.

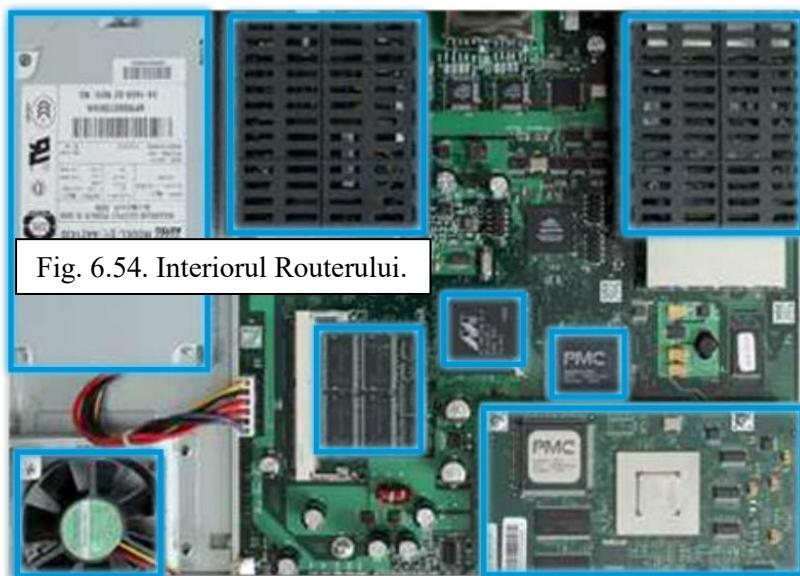


Fig. 6.54. Interiorul Routerului.

Un router Cisco 1941 include următoarele componente:

- Porturi de consolă – două porturi de consolă pentru configurație inițială și accesul de management CLI folosind un port normal RJ-45 și un nou conector USB Type-B (mini-B USB).
- Port AUX – un port RJ-45 pentru accesul de management de la distanță; este similar portului de consolă.
- Două interfețe LAN – două interfețe Gigabit Ethernet pentru accesul la LAN.
- Enhanced high-speed WAN interface card (EHWIC) slots – două sloturi ce oferă modularitate și flexibilitate prin permiterea routerului să suporte tipuri diferite de module de interfață, inclusiv Serial, digital subscriber line (DSL), switch port și wireless.

Cisco 1941 ISR de asemenea are sloturi de stocare pentru a suporta capacitați extinse. Sloturile de memorie Dual-compact flash sunt capabile să suporte un card de 4 GB de compact flash pentru a crește spațiul de stocare. Două porturi host USB sunt incorporate pentru spațiu de stocare suplimentar și pentru capacitate token securizată.

Compact flash poate stoca Cisco IOS software image, fișierele de logare, fișierele de configurație, fișierele HTML, fișierele de backup sau oricare alte fișiere necesare pentru sistem. Implicit, numai slotul 0 este populat cu un card de compact flash din fabricație și este, implicit, locația boot.

Fig. 6.55 identifică locația acestor conexiuni și sloturi.

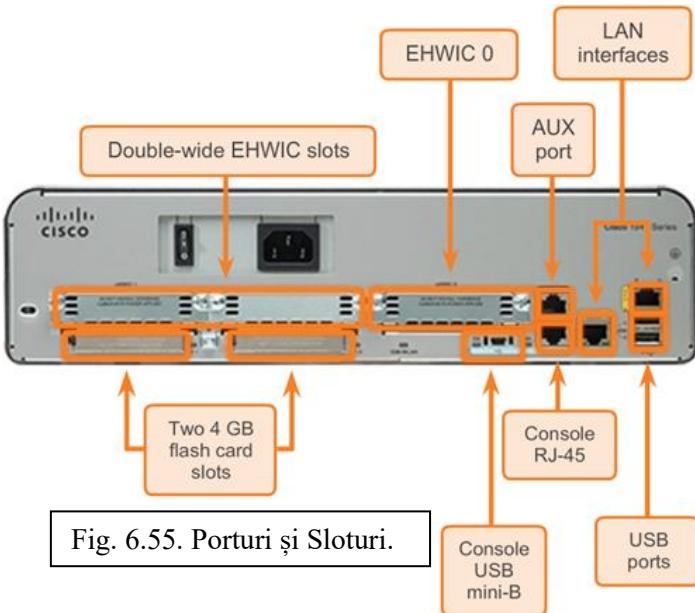


Fig. 6.55. Porturi și Sloturi.

Dispozitivele Cisco, routerele și switchurile de obicei interconectează mai multe dispozitive. Din acest motiv, aceste dispozitive au mai multe tipuri de porturi și interfețe. Aceste porturi și interfețe sunt folosite pentru a conecta cablurile la dispozitiv.

Conexiunile unui router Cisco pot fi grupate în două categorii:

- *Porturi de management* – Acestea sunt porturile de consolă și auxiliare folosite pentru a configura, gestiona și depăra routerul. Spre deosebire de interfețele LAN și WAN, porturile de management nu sunt utilizate pentru transmiterea pachetelor.
- *Interfețe inband de router* – Acestea sunt interfețele LAN și WAN configurate cu adresa IP pentru a transporta traficul de utilizator. Interfețele Ethernet sunt cele mai cunoscute conexiuni LAN, iar conexiunile WAN includ interfețe seriale și DSL.

Fig. pune în evidență porturile și interfețele unui router Cisco 1941 ISR G2.

Precum multe dispozitive de rețea, dispozitivele Cisco folosesc indicatori LED pentru a oferi informații de status. O interfață LED indică activitatea interfeței corespunzătoare. Dacă un LED este stins atunci când interfața este activă și conectată corect, ar putea fi o indicație a faptului că există o problemă cu acea interfață. Dacă o interfață este foarte ocupată, LEDul este întotdeauna aprins.

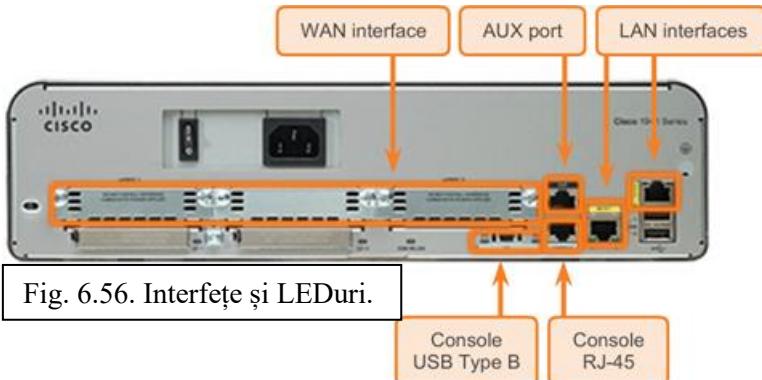


Fig. 6.56. Interfețe și LEDuri.

Similar cu un switch Cisco, există mai multe moduri de acces la mediul CLI de pe un router Cisco. Cele mai comune metode sunt:

- *Consola* – Folosește un serial de viteză scăzută sau o conexiune USB pentru a oferi conexiune directă, out-of-band management access la un dispozitiv Cisco.

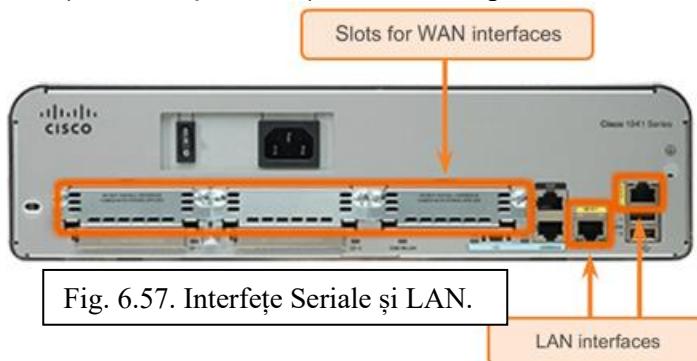
- *Telnet sau SSH* – Două metode pentru accesarea de la distanță a sesiunii CLI printr-o interfață de rețea activă.
- *Portul AUX* – Folosit pentru management de la distanță a unui router utilizând o linie de telefonie dial-up și un modem.
Porturile de consola și AUX sunt localizate pe router.

În plus față de aceste porturi, routerele au interfețe de rețea pentru a primi și transmite pachete IP. Routerele au mai multe interfețe ce sunt folosite pentru a conecta mai multe rețele. În mod normal, interfețele conectează mai multe tipuri de rețele, ceea ce înseamnă că tipuri diferite de mediu și conectori sunt necesari.

Fiecare interfață de pe router este un membru sau un host dintr-o rețea IP diferită. Fiecare interfață trebuie să fie configurată cu o adresă IP și o mască de rețea a fiecărei rețele diferite. Cisco IOS nu permite ca două interfețe active de pe același router să se afle în aceeași rețea.

Interfețele routerului sunt grupate în două categorii:

- Interfețe LAN Ethernet – Folosite pentru conectarea de cabluri între router și dispozitivele LAN, cum ar fi computere sau switchuri. Această interfață poate fi folosită pentru a conecta routerele între ele. Mai multe convenții pentru numirea interfețelor Ethernet sunt populare: Ethernetul vechi, FastEthernet și Gigabit Ethernet. Numele folosit depinde de tipul și modelul dispozitivului.
- Interfețe WAN seriale – Folosite pentru conectarea routerelor la rețele externe, de obicei peste o distanță geografică mare. Similar cu interfețele LAN, fiecare interfață WAN are propria adresă IP și mască de rețea, ce se identifică ca un membru pentru o rețea specifică. Fig. 6.57 arată interfețele LAN și interfețele seriale de pe un router.



6.4.1 Bootarea Routerului

Detaliile operaționale IOS variază pe dispozitivele diferite de internetworking, în funcție de scopul dispozitivului și de caracteristica setată. Însă, IOSul pentru routere oferă următoarele:

- *Adresare.*
- *Interfețe.*
- *Rutare.*
- *Securitate.*
- *Qos.*
- *Managementul resurselor.*

Fișierul IOS constă din mai mulți megabytes în dimensiune și este similar cu IOS de pe switchuri și este stocat în memoria flash. Folosind flash permite IOSului să fie upgradat la cele mai noi versiuni sau să aibă noi caracteristici adăugate. În timpul procesului de bootup, IOS este

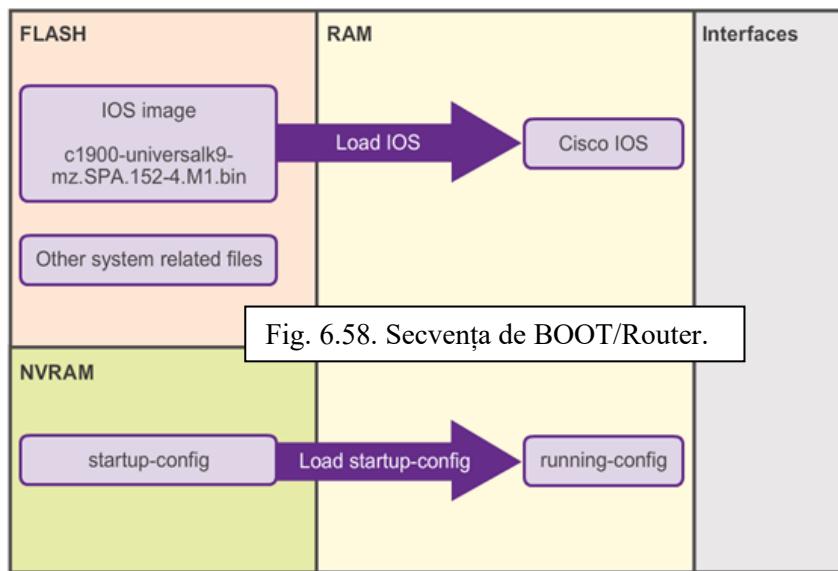
copiat din memoria flash în RAM. DRAM este mult mai rapid decât flash; prin urmare, copierea IOS în RAM crește performanța dispozitivului.



Așa cum este arătat și în Fig. , un router încarcă următoarele două fișiere în RAM atunci când are loc procesul de boot:

- Fișierul cu imaginea IOS – IOS facilitează funcționarea de bază a componentelor hardware ale dispozitivului. Fișierul de imagine IOS este stocat în memoria flash.
- Fișierul de config.re startup – Fișierul de config.re startup conține comenzi necesare pentru configurația inițială a unui router și crează fișierul de config.re running stocat în RAM. Fișierul de config.re startup este stocat în NVRAM. Toate schimbările de config.re sunt stocate în fișierul de config.re running și sunt implementate imediat de către IOS.

Configurația running este modificată atunci când administratorul de rețea efectuează configurația dispozitivului. Atunci când sunt efectuate schimbările în fișierul running-config, ar trebui salvate în NVRAM în fișierul de config.re startup, în cazul în care routerul este restartat sau își pierde alimentarea.



Exista trei faze principale în procesul de bootup, arătate în Fig. 6.58:

- Efectuarea **POST** și încărcarea programului de bootstrap.
- Localizarea și încărcarea softwareului Cisco IOS.
- Localizarea și încărcarea fișierului de config.re și intrarea în modul de setare.

1. Efectuarea **POST** și încărcarea programului de bootstrap (Fig. 6.59)

Power-On Self Test (**POST**) este un proces obișnuit ce are loc pe aproape toate computerele în timpul procesului de bootup. Procesul Power-On Self Test (**POST**) este folosit pentru a testa hardwareul routerului. Atunci când routerul este alimentat, softwareul de pe ROM efectuează **POST**. În timpul acestui test, routerul execută diagnosticarea din ROM pe mai multe componente hardware, inclusiv CPU, RAM și NVRAM. După ce **POST** a fost finalizat, routerul execută programul de bootstrap.

După **POST**, programul de bootstrap este copiat din ROM în RAM. O dată aflat în RAM, CPU execută instrucțiunile din programul de bootstrap. Principala sarcină a programului de bootstrap este de a localiza IOSul și încărcarea sa în RAM.

Notă: În acest moment, dacă avem conexiune prin consolă la router, începem să vedem ieșirea pe ecran.

2. Localizarea și încărcarea softwareului IOS (Fig. 6.60)

IOS este de obicei stocat în memoria flash și este copiat în RAM pentru execuția sa de către CPU. În timpul auto-decompresiei fișierului cu imaginea IOS, un string de semnale # vor fi afișate.

Dacă imaginea IOS nu este localizată în flash, routerul ar putea să o caute într-un server TFTP. Dacă nu poate fi localizată o imagine completă IOS, o versiune limitată de IOS este copiată din ROM în RAM. Această versiune de IOS este folosită pentru ajutarea diagnosticării oricărei probleme și poate fi folosită pentru încărcarea unei versiuni complete de IOS în RAM.

3. Localizarea și încărcarea fișierului de config.re și intrarea în modul de setare (Fig. 6.61).

Programul de bootstrap caută fișierul de config.re startup în NVRAM. Acest fișier are parametrii și comenzi anterioare de config.re salvate. Dacă există, este copiat în RAM ca fișierul de config.re running, running-config. Fișierul running-config conține adresele de interfață, încep procesele de rutare, configuraază parolele routerului și definește alte caracteristici ale routerului.

Dacă fișierul startup-config nu există în NVRAM, routerul ar putea să îl caute într-un server TFTP. Dacă routerul detectează faptul că are o legătură activă la un alt router config.t, trimite la legătură activă o căutare broadcast pentru un fișier de config.re.

Dacă un server TFTP nu este găsit, routerul afișează promptul modului de config.re. Modul de config.re este redat printr-o serie de întrebări afișate utilizatorului pentru informații de bază de config.re. Modul de config.re nu este folosit pentru a introduce configuri complexe pe router și nu este folosit de obicei de administratorii de rețea.

Notă: Modul de config.re nu este folosit în acest curs pentru a configura routerul. Când se afișează modul de config.re, întotdeauna răspundem cu nu. Dacă m răspuns da și apare modul de config.re, apăsăm oricând combinația de taste **Ctrl+C** pentru a termina procesul de config.re.

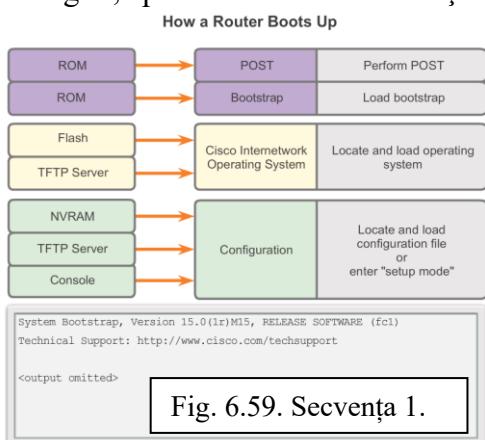


Fig. 6.59. Secvența 1.

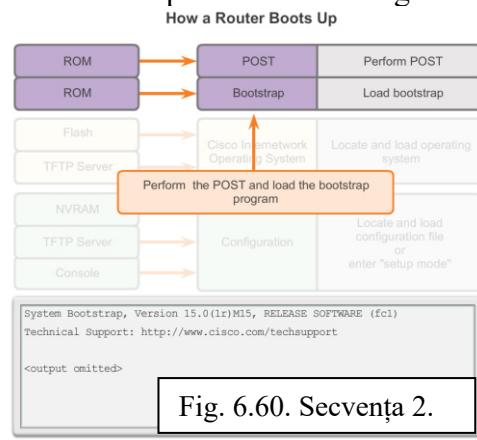
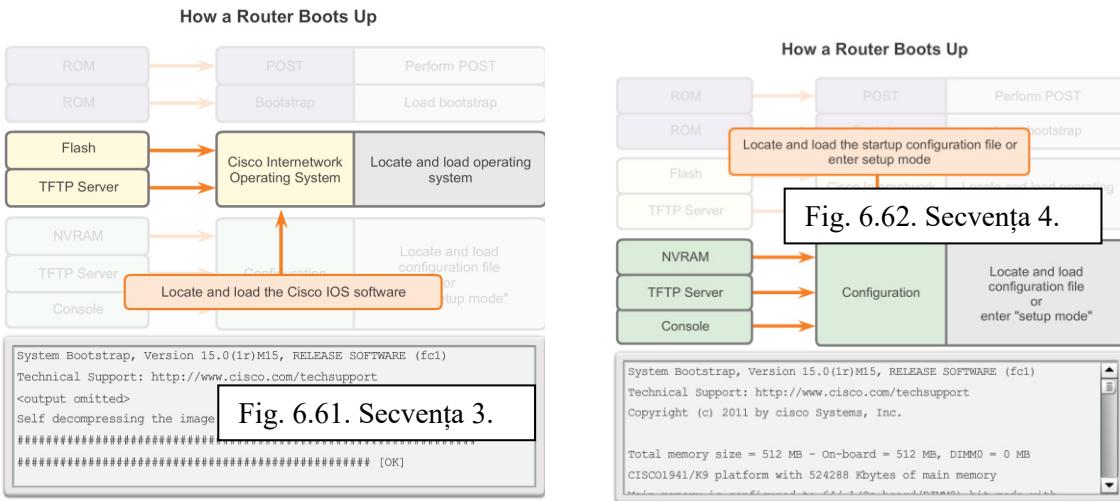


Fig. 6.60. Secvența 2.



Putem folosi comanda **show version** pentru a verifica și depana unele componente de bază hardware și software ale routerului. Comanda afișează informații cu privire la versiunea softwareului IOS ce rulează pe router, versiunea programului de bootstrap și informații despre configurația hardware, inclusiv cantitatea de memorie a sistemului.

Ieșirea comenzi **show version** include:

- *Versiunea IOS* – Versiunea softwareului IOS din RAM folosită de către router.
- *Programul de bootstrap ROM* – Afișează versiunea softwareului de sistem bootstrap, stocat în ROM, ce a fost inițial folosit pentru bootarea routerului.
- *Locația IOS* – Afișează unde este localizat programul de bootstrap și unde este încărcat IOSul și numele complet al imaginii IOS.
- *CPU și cantitatea de RAM* – Prima parte a acestei linii afișează tipul de CPU de pe router. Ultima parte a acestei linii afișează cantitatea de DRAM. Unele serii de routere, cum ar fi Cisco 1941 ISR, folosesc o fracțiune de DRAM ca memorie de pachet. Memoria de pachet este folosită pentru buffer de pachete. Pentru a determina cantitatea totală de DRAM a routerului, adunăm ambele numere.
- *Interfețe* – Afișează interfețele fizice de pe router. În acest exemplu, Cisco 1941 ISR are două interfețe Gigabit Ethernet și două interfețe seriale de viteză scazută.
- *Cantitatea de NVRAM și Flash* – Aceasta este cantitatea de NVRAM și cantitatea de memorie flash a routerului. NVRAM este folosită pentru a stoca fișierul de startup-config și flash este folosit pentru a stoca permanent IOSul.

Ultima linie a comenzi **show version** afișează valoarea curentă config.tă a registrului software de config.re în hexazecimal. Dacă există o sau două valoare afișată în paranteze, arată valoarea registrului de config.re folosită în timpul următorului reload.

Registrul de config.re are mai multe utilizări, inclusiv recuperarea de parolă. Setarea din fabrică implicită pentru registrul de config.re este 0x2102. Aceasta valoare indică faptul că routerul încearcă să încarce imaginea software IOS din memoria flash și încarcă fișierul de config.re startup din NVRAM.

```

Router#show version
Cisco IOS Software, C1900 Software (C1900-UNIVERSALK9-M),
Version 15.2(4)M1, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Thu 26-Jul-12 19:34 by prod_rel_team

ROM: System Bootstrap, Version 15.0(1r)M15,
RELEASE SOFTWARE (fc1)

Router uptime is 10 hours, 9 minutes
System returned to ROM by power-on
System image file is
"flash0:c1900-universalk9-mz.SPA.152-4.M1.bin"
Last reload type: Normal Reload
Last reload reason: power-on

<Output omitted>

Cisco CISCO1941/K9 (revision 1.0)
with 446464K/77824K bytes of memory.
Memory available for reuse: 446464K/446464K

```

Technology Package License Information for Module:'c1900'			
Technology	Technology-package	Technology-package	
Current	Type	Next reboot	
ipbase	ipbasek9	Permanent	ipbasek9
security	None	None	None
data	None	None	None

Configuration register is 0x2142
(will be 0x2102 at next reload)

Fig. 6.63. Comenzi de verificare.

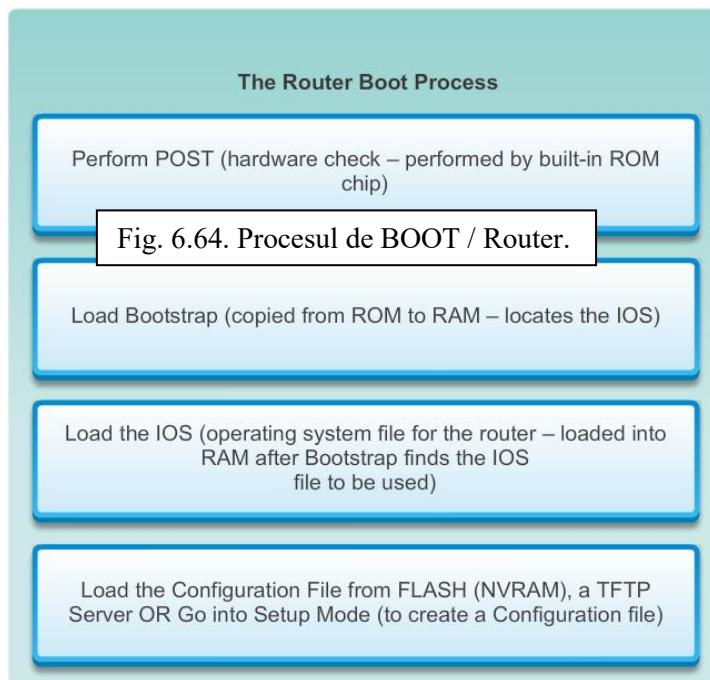


Fig. 6.64. Procesul de BOOT / Router.

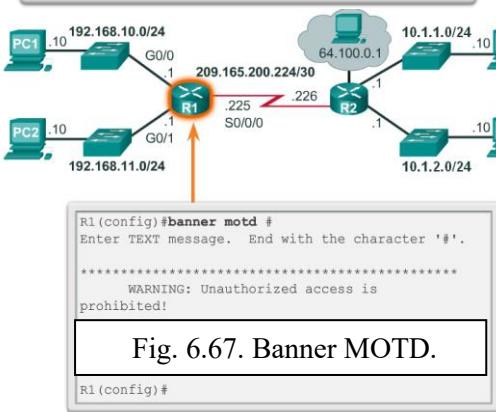
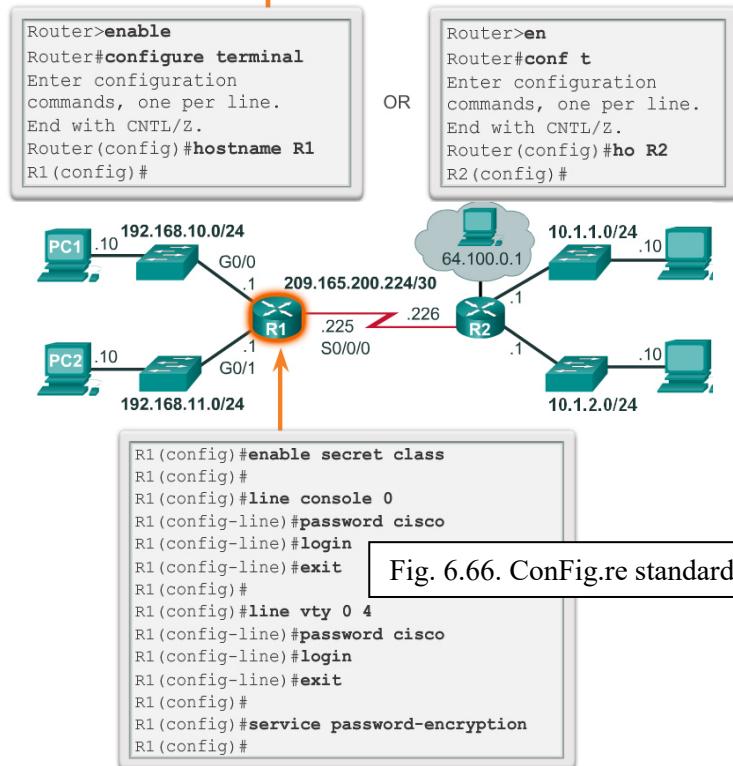
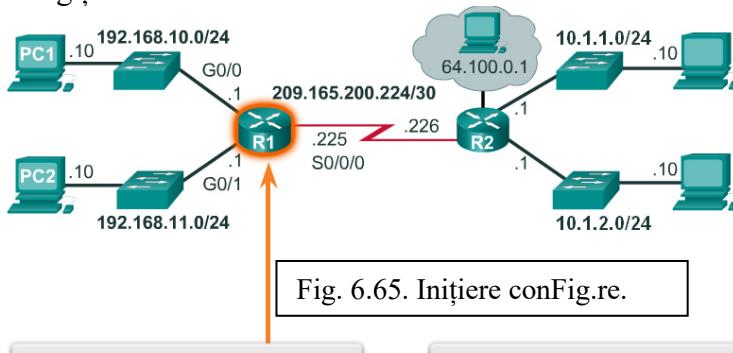
6.4.2 ConFig.rea Routerului – Setările Inițiale de ConFig.re

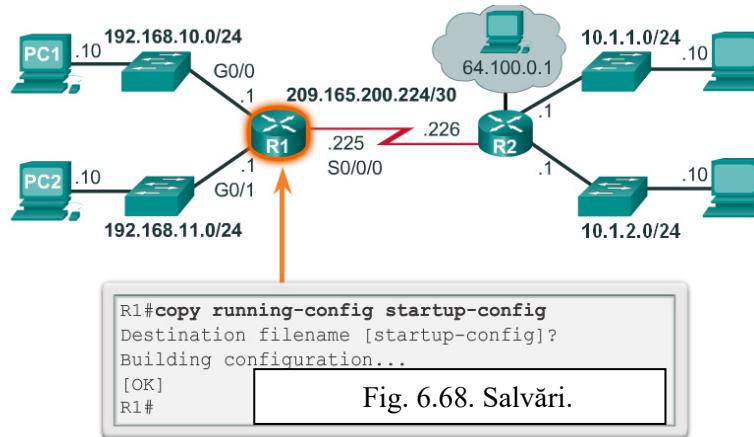
Routerele și switchurile au mai multe similarități. Ambele suportă un sistem similar de operare, structuri similare de comandă și suportă mai multe comenzi la fel. În plus, ambele dispozitive au aceeași pași inițiali de config. re atunci când sunt puse în aplicare într-o rețea.

Similar cu config. rea unui switch, următorii pași trebuie să fie efectuați atunci când se configurează setările inițiale de pe un router:

- Atribuirea unui nume de dispozitiv prin folosirea comenții **hostname** în modul de config. re global.
- Setarea parolelor.
- Securizarea accesului la modul privilegiat prin folosirea comenții **enable secret**.
- Securizarea accesului la modul Exec folosind comanda **login** pe portul de consolă și comanda **password** pentru a seta parola.

- Securizarea accesului virtual în mod similar cu securizarea modului de acces Exec, însă se face pe portul Virtual Telnetetype (VTY).
- Folosirea comenzi de config.re globală **service password-encryption** pentru a preveni ca parolele să fie afișate în text clar în fișierul de config.re.
- Oferearea unei notificări legale prin folosirea comenzi de config.re globală **banner motd** (message of the day [MOTD]).
- Salvarea config.ției folosind comanda **copy running-config startup-config**.
- Verificarea config.ției folosind comanda **show run**.





6.4.2.1 ConFig.rea Interfețelor

Pentru ca routerele să fie accesibile, interfețele routerului trebuie să fie conFig.re. Prin urmare, pentru a activa o anumită interfață, intrăm în modul de conFig.re a interfeței folosind comanda **interface type-and-number** în modul de conFig.re global.

Există mai multe tipuri de interfețe disponibile pe routere. În acest exemplu, routerul Cisco 1941 este echipat cu două interfețe Gigabit Ethernet și un card de interfață serială WAN ce constă din două interfețe; interfețele sunt numite astfel:

- *Gigabit Ethernet 0/0 (G0/0).*
- *Gigabit Ethernet 0/1 (G0/1).*
- *Serial 0/0/0 (S0/0/0).*
- *Serial 0/0/1 (S0/0/1).*

Pentru a activa o interfață de router, configuriăm următoarele:

- Adresa IPv4 și masca de rețea – *Configurăm adresa IPv4 și masca de rețea folosind comanda de conFig.re de interfață ip address subnet-mask.*
- Activăm interfața – *Implicit, interfețele LAN și WAN nu sunt activate. Interfețele trebuie să fie activate folosind comanda no shutdown.* Aceasta este similară cu alimentarea interfeței. Interfața trebuie să fie conectată la un alt dispozitiv (un hub, un switch, sau un alt dispozitiv) pentru ca nivelul fizic să fie activ.

Deși nu este necesar, este bine să configuriăm o descriere pe fiecare interfață pentru a ajuta la documentarea informațiilor de rețea. Textul de descriere este limitat la 240 de caractere. Pe rețele de producție, o descriere poate fi folosită în depanare prin oferirea de informații despre tipul de rețea la care interfață este conectată și dacă este alt router în rețea. Dacă interfața se conectează la un ISP sau purtător de servicii, este de ajutor să introducem conexiunea la a treia parte și informații de contact.

Fig. 6.69 arată conFig.ția interfețelor LAN conectate la R1.

Notă: Abrevierile de comandă sunt folosite pentru conFig.rea Gigabit Ethernet 0/1.

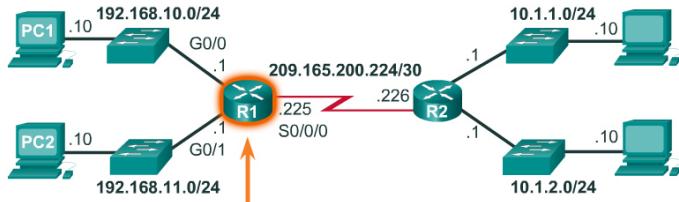


Fig. 6.69. ConFig.re Interfețe.

```
R1#conf t
Enter configuration commands, one per line.
End with CNTL/Z.
R1(config)#
R1(config)#interface gigabitethernet 0/0
R1(config-if)#ip address 192.168.10.1 255.255.255.0
R1(config-if)#description Link to LAN-10
R1(config-if)#no shutdown
%LINK-5-CHANGED: Interface GigabitEthernet0/0,
changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0 changed state to up

R1(config-if)#exit
R1(config)#
R1(config)#int g0/1
R1(config-if)#ip add 192.168.11.1 255.255.255.0
R1(config-if)#des Link to LAN-11
R1(config-if)#no shut
%LINK-5-CHANGED: Interface GigabitEthernet0/1,
changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/1, changed state to up
R1(config-if)#exit
R1(config)#

```

Există mai multe comenzi ce pot fi folosite să se verifice conFig.rea interfeței. Cea mai utilă dintre acestea este comanda **show ip interface brief**. Ieșirea generată de aceasta afișează toate interfețele, adresa IP respectivă și stările actuale. Interfețele conFig.te și conectate ar trebui să afișeze un status “up” și protocol “up”. Orice altceva va indica o problemă fie în conFig.re, fie în cablare.

Putem verifica conectivitatea unei interfețe prin folosirea comenzi **ping**. Routerele trimit cinci pinguri consecutive și măsoară timpurile minime, medii și maxime de dus-întors. Semnul de exclamare marchează verificarea conectivității.

Fig. 1 afișează ieșirea comenzi **show ip interface brief**, care descoperă că interfețele LAN și legăturile WAN sunt toate operaționale. De remarcat faptul că pingul generează cinci semne de exclamare, verificând conectivitatea cu R2.

Alte comenzi de verificare a interfeței sunt:

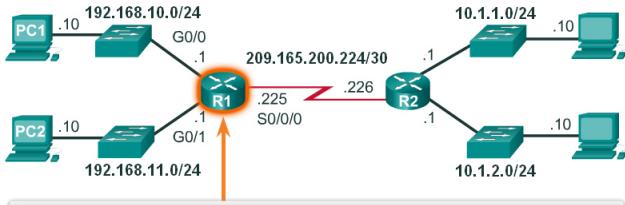
show ip route - Afisează conținutul tabelei de rutare stocată în RAM.

show interfaces - Afisează statistici pentru toate interfețele de pe dispozitiv.

show ip interface - Afisează statistici IPv4 pentru toate interfețele de pe router.

Fig. 6.70 afișează ieșirea comenzi **show ip route**. De remarcat intrările de rețele direct conectate și intrările de interfețe cu legătură locală.

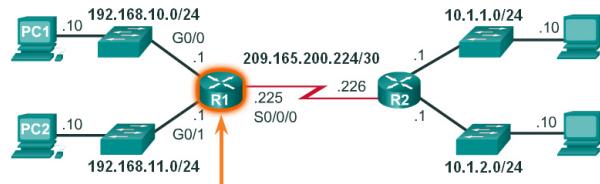
Să nu uităm să salvăm conFig.rea folosind comanda **copy running-config startup-config**.



```
R1#show ip interface brief
Interface          IP-Address      OK? Method Status
GigabitEthernet0/0  192.168.10.1   YES  manual up
GigabitEthernet0/1  192.168.11.1   YES  manual up
Serial0/0/0          209.165.200.225 YES  manual up
Serial0/0/1          unassigned     YES  NVRAM administratively down
Vlan1               unassigned     YES  NVRAM administratively down
R1#
R1#ping 209.165.200.226
Type escape sequence: [Ctrl-C]
Sending 5, 100-byte ICMP Echos to 209.165.200.226, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5),
round-trip min/avg/max = 1/2/9 ms
R1#
```

Fig. 6.70. Comenzi de control.

```
Serial0/0/1          unassigned     YES  NVRAM administratively down
Vlan1               unassigned     YES  NVRAM administratively down
R1#
R1#ping 209.165.200.226
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.226,
timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5),
round-trip min/avg/max = 1/2/9 ms
R1#
```



```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP,
       M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF,
       IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1,
       L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
Gateway of last resort is not set
```

Fig. 6.71. SHOW IP ROUTE.

```
Gateway of last resort is not set

192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C    192.168.10.0/24 is directly connected, GigabitEthernet0/0
L    192.168.10.1/32 is directly connected, GigabitEthernet0/0
192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C    192.168.11.0/24 is directly connected, GigabitEthernet0/1
L    192.168.11.1/32 is directly connected, GigabitEthernet0/1
209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks
C    209.165.200.224/30 is directly connected, Serial0/0/0
L    209.165.200.225/32 is directly connected, Serial0/0/0
R1#
```

6.4.2.2 ConFig.rea Default Gateway

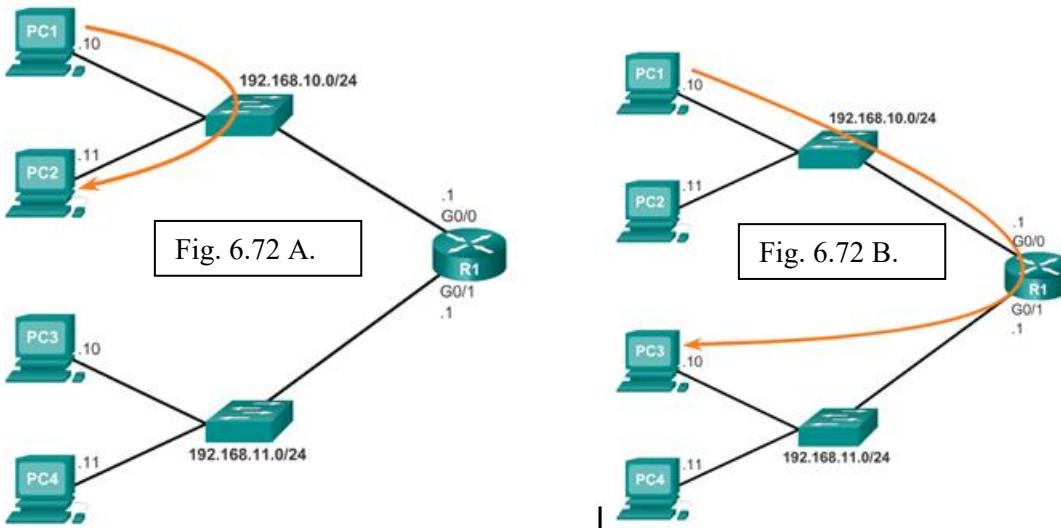
Cele mai multe routere au, cel puțin, două interfețe. Fiecare interfață este conFig.tă cu o adresă IP separată și cu o mască de rețea.

Pentru ca un dispozitiv final să comunice peste rețea, trebuie să fie conFig.t cu informațiile de adresă IP corecte, inclusiv adresa de default gateway. Default gateway este folosită numai atunci când hostul dorește să trimită un pachet la un dispozitiv dintr-o altă rețea. Adresa de default gateway este în general adresa de pe interfața routerului atașată la rețeaua locală a hostului. Pe când pe router nu contează ce adresă este conFig.tă, adresa IP a dispozitivului host și adresa de pe interfața routerului trebuie să fie ambele din aceeași rețea.

Fig. afișează o topologie cu un router și două interfețe separate. Fiecare interfață este conectată la o rețea separată. G0/0 este conectată la 192.168.10.0, pe când ce G0/1 la rețeaua 192.168.11.0. Fiecare dispozitiv host este conFig.t cu adresa adecvată de default gateway.

În Fig. 6.72 A, PC1 trimite un pachet la PC2. În acest exemplu, default gateway nu este folosită; PC1 adresează pachetul cu adresa IP a PC2 și îl trimită direct la PC2 prin intermediul switchului.

În Fig. 6.72. B, PC1 trimite un pachet la PC3. În acest exemplu, PC1 adresează pachetul cu adresa IP a PC3, dar îl transmite la router. Routerul acceptă pachetul, își adresează tabela de rutare și determină interfața de ieșire adecvată pentru adresa destinație și apoi trimită pachetul pe interfața adecvată spre PC3.



Un default gateway este utilizat de către toate dispozitivele ce folosesc un router să determine cea mai bună cale spre o destinație de la distanță. Dispozitivele finale necesită adrese de default gateway, la fel și cele intermediare, cum ar fi switchul.

Informațiile de adresa IP de pe un switch sunt necesare numai pentru a gestiona de la distanță switchul. Cu alte cuvinte, pentru a fi capabili să accesăm switchul de la distanță, switchul trebuie să aibă o adresă IP de accesare prin Telnet. Dacă switchul este accesat numai de dispozitivele din rețeaua locală, numai o adresă IP este necesară.

ConFig.rea unei adrese IP pe un switch se face pe switch virtual interface (SVI):

- S1(config)#interface vlan1
- S1(config-vlan)#description Legatura cu reteaua 192.168.10.0/24
- S1(config-vlan)#ip address 192.168.10.50 255.255.255.0
- S1(config-vlan)#no shut

Însă, dacă switchul trebuie să fie accesibil de către dispozitivele dintr-o rețea diferită, switchul trebuie să fie conFig.t cu o adresă de default gateway deoarece pachetele provenite de la

switch sunt manipulate ca și pachetele provenite de la un dispozitiv host. Prin urmare, pachetele ce provin de la switch și sunt destinate pentru un dispozitiv din aceeași rețea sunt transmise direct la dispozitivul respectiv. Pachetele provenite de la switch și destinate pentru un dispozitiv de pe o rețea de la distanță trebuie să fie trimise la default gateway pentru determinarea traseului.

Pentru a configura un default gateway pe un switch folosim următoarea comandă de configurație globală:

```
S1(config)#ip default-gateway 192.168.10.1
```

Fig. 6.73 arată un administrator de rețea ce se conectează la un switch dintr-o rețea de la distanță. Pentru ca switchul să transmită pachete de răspuns administratorului, trebuie să fie configurația **ip default gateway**.

O concepție greșită comună este aceea că switchul folosește propria adresă de default gateway pentru a determina unde să transmită pachetele provenite de la hosturile conectate la switch și destinate hosturilor dintr-o rețea de la distanță. Însă, adresa IP și informația de default gateway sunt folosite numai pentru pachetele ce provin de la switch. Pachetele ce provin de la hosturile conectate la switch trebuie să aibă deja informații de default gateway configurație pentru a comunica în rețelele de la distanță.

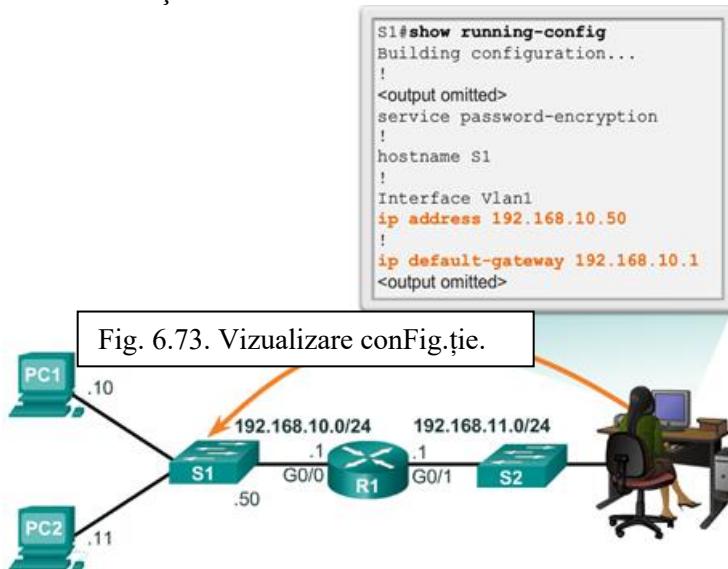


Fig. 6.73. Vizualizare configurație.

Pentru ca un dispozitiv să comunice peste mai multe rețele, trebuie să configuriăm o adresă IP, o mască de rețea și un default gateway. Default gateway este folosit atunci când hostul vrea să transmită un pachet la un dispozitiv dintr-o rețea diferită. Adresa de default gateway este în general adresa de pe interfață routerului atașată la rețea locală la care este conectat hostul. În această activitate, vom finaliza documentarea rețelei. Vom verifica apoi documentația rețelei prin testarea conectivității end-to-end și a problemelor de dapanare. Metoda de depanare folosită constă din următorii pași:

- Verificarea documentației de rețea și utilizarea de teste pentru izolare problemelor.
- Determinarea unei soluții adecvate pentru o problemă dată.
- Implementarea soluției.
- Testarea pentru verificarea faptului că problema este rezolvată.
- Documentarea soluției.

6.5 Concluzii Capitolul 6

Nivelul rețea sau nivelul 3 OSI oferă servicii ce permit dispozitivelor finale să schimbe date în rețea. Pentru a realiza acest transport end-to-end, nivelul rețea folosește patru procese de bază: adresare IP pentru dispozitive finale, încapsulare, rutare și decapsulare.

Internetul se bazează mult pe IPv4, cel mai utilizat protocol de nivel rețea la nivel global. Un pachet IPv4 conține header IP și payload. Însă, IPv4 are un număr limitat de adrese IP unice publice disponibile. Acest lucru a condus la dezvoltarea IPv6. IPv6 simplifică headerul și oferă mai multe avantaje peste IPv4, inclusiv eficiență de rutare mai bună, headere de extensie simplificate și capacitate pentru procesare per-flow. În plus, adresele IPv6 sunt bazate pe adresa ierarhică de 128 de biți, spre deosebire de IPv4 pe 32 de biți. Acest lucru crește dramatic numărul de adrese IP disponibile.

În plus față de adresarea ierarhică, nivelul rețea este responsabil și de rutare.

Hosturile necesită o tabelă de rutare locală pentru a asigura faptul că pachetele sunt direcționate la rețeaua destinație corectă. Tabela locală a unui host conține de obicei conexiunea directă, ruta de rețea locală și ruta default locală. Ruta default locală este ruta către default gateway.

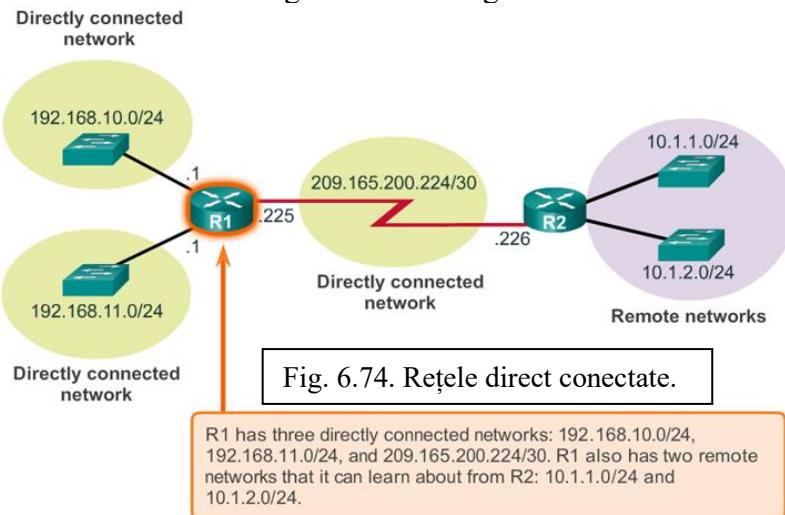
The default gateway is the IP address of a router interface connected to the local network. When a host needs to forward a packet to a destination address that is not on the same network as the host, the packet is sent to the default gateway for further processing.

Default gateway este adresa IP a interfeței routerului conectată la rețeaua locală. Atunci când un host are nevoie să transmită un pachet la o adresa destinație ce nu se află în aceeași rețea cu hostul, pachetul este trimis la default gateway pentru procesare ulterioară.

Atunci când un router, ca și default gateway, primește un pachet, examinează adresa IP destinație a rețelei destinație. Tabela de rutare a unui router stochează informații despre rutele direct conectate și rutele de la distanță la rețelele IP. Dacă routerul are o intrare în tabela sa de rutare pentru rețeaua destinație, routerul transmite pachetul. Dacă nu există nici-o intrare, routerul ar putea transmite pachetul la propria sa ruta implicită, dacă este config.tă, sau va arunca pachetul.

Intrările din tabela de rutare pot fi config.tă manual pe fiecare router pentru a oferi rutare statică sau routerele pot comunica între ele informații de rute în mod dinamic, folosind un protocol de rutare.

Pentru a routerele să fie accesibile, interfața routerului trebuie să fie config.tă. Pentru a activa o interfață specifică, intrăm în modul de config.re interfață prin introducerea comenzi interface type-and-number din modul global de config.re.



CAPITOLUL 7. NIVELUL TRANSPORT

Introducere

Rețelele de date și Internetul suportă rețeaua umană prin furnizarea de comunicații de încredere între oameni. Pe un singur dispozitiv, oamenii pot folosi mai multe aplicații și servicii cum ar fi e-mail, web și mesagerie instant pentru a trimite mesaje sau a prelua informații. Aplicațiile cum ar fi clienții de e-mail, browserele web și clienții de mesagerie instant permit oamenilor să folosească computere și rețelele pentru a trimite mesaje și a afla informații.

Datele de la fiecare dintre aceste aplicații sunt împachetate, transportate și livrate la aplicația adecvata de pe dispozitivul destinație. Procesele descrise în nivelul de transport OSI acceptă datele de la nivelul aplicație și le pregătește pentru adresarea lor de la nivelul rețea. Nivelul transport *pregătește* datele pentru transmisia peste rețea. Un computer sursă comunică cu un computer destinație pentru a decide cum să împartă data în **segmente**, cum să se asigure de faptul că segmentele nu se pierd și cum să verifice faptul că segmentele au ajuns la destinație. Atunci când ne gandim la nivelul aplicație, ne gandim la un departament de transport ce pregătește o singură ordine de livrare pentru mai multe pachete.

În acest capitol, vom examina rolul nivelului transport în încapsularea datelor aplicației pentru utilizarea lor de către nivelul rețea. Nivelul transport îndeplinește de asemenea și următoarele funcții:

- *Permite mai multor aplicații, cum ar fi e-mail și rețeaua socială, să comunice peste rețea în același timp, pe un singur dispozitiv.*
- *Asigură faptul că, dacă este necesar, toate datele sunt primite în ordine și într-un mod de încredere de la aplicația corectă.*
- *Folosește mecanisme de tratare a erorilor.*

Obiective țintă

La finalizarea acestui capitol, vom fi capabili să:

- Explicăm necesitatea nivelului transport.
- Identificăm rolul nivelului transport în oferirea transferului end-to-end a datelor dintre aplicații.
- Descriem rolul celor două protocole de la nivelul transport din TCP/IP: TCP și UDP.
- Explicăm funcțiile cheie ale nivelului transport, inclusiv încrederea, adresarea pe bază de port și segmentarea.
- Explicăm modul în care protocolele UDP și TCP se ocupă de funcțiile cheie.
- Identificăm când este necesar să folosim TCP sau UDP și să oferim exemple de aplicații pentru fiecare protocol.

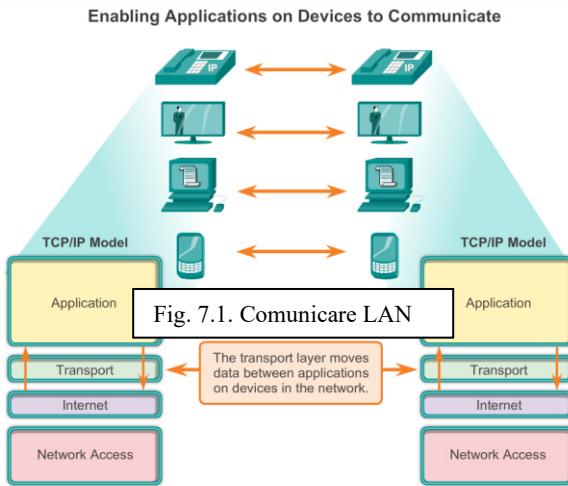
7.1 Protocolele de la nivelul transport – Transportul datelor

Nivelul transport este responsabil de stabilirea unei sesiuni temporare de comunicație între două aplicații și de livrarea datelor dintre ele. O aplicație generează date ce trebuie transmise de la o aplicație de pe hostul sursă la o aplicație de pe hostul destinație, indiferent de tipul de host destinație, tipul de mediu peste care datele trebuie să călătorească, calea luată de date, congestia unei legături sau de dimensiunea rețelei. Ca și în Fig. , nivelul transport este legătură dintre nivelul aplicație și nivelele inferioare responsabile de transportul de rețea.

Nivelul transport oferă o metodă de livrare a datelor prin rețea încruntându-le în mod ce ne asigură că datele pot fi reasamblate la destinație. Nivelul transport oferă segmentarea datelor și controlul necesar pentru reasamblarea acestor segmente în streamuri diferite de comunicație. În TCP/IP, aceste procese de segmentare și reasamblare pot fi realizate prin utilizarea a două protocoale foarte diferite de la nivelul transport : Transmission Control Protocol (TCP) și User Datagram Protocol (UDP).

Principalele responsabilități ale protocoalelor de la nivelul transport sunt:

- *Urmărirea comunicării individuale dintre aplicațiile de pe hosturile sursă și destinație.*
- *Segmentarea datelor pentru administrare și reasamblarea datelor segmentate în fluxuri ale datelor aplicație de la destinație.*
- *Identificarea aplicației adevărate pentru fiecare stream de comunicație.*



7.1.1 Urmărirea conversațiilor individuale

La nivelul transport, fiecare set particular de date ce călătoresc între o aplicație sursă și o aplicație destinație se numește conversație (Fig. 1). Un host ar putea avea mai multe aplicații ce comunică în rețea simultan. Fiecare dintre aceste aplicații comunică cu una sau mai multe aplicații de pe unul sau mai multe hosturi de la distanță. Este responsabilitatea nivelului transport să mențină și să urmărească aceste conversații multiple.

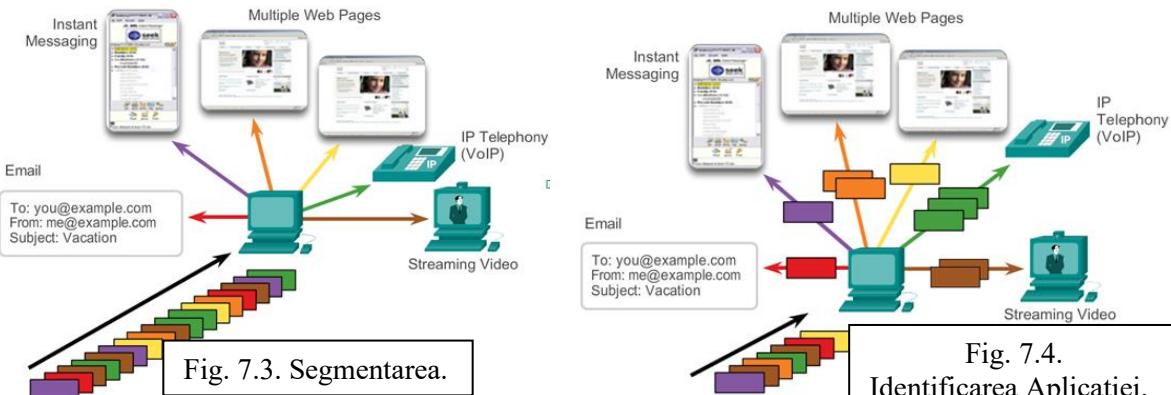
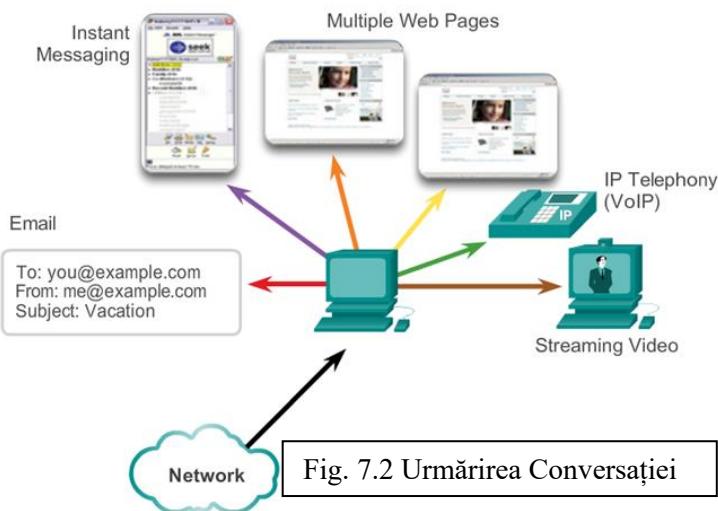
7.1.1.1 Segmentarea datelor și reasamblarea segmentelor

Datele trebuie să fie pregătite pentru a fi transmise pe mediu în piese gestionabile. Cele mai multe rețele au o limită a cantității de date ce poate fi inclusă într-un singur pachet. Protocolele de la nivelul transport au servicii ce segmentează datele aplicației în blocuri de date ce au o dimensiune adevărată (Fig. 2). Serviciul include încapsularea necesară pentru fiecare piesă de date. Un header, folosit pentru reasamblare, este adăugat la fiecare bloc de date. Acest header este folosit pentru urmărirea streamului de date.

La destinație, nivelul transport trebuie să fie capabil să reconstruiască piesele de date încruntându-le într-un stream de date complet ce este folosit de nivelul aplicație. Protocolele de la nivelul transport descriu modul în care informațiile din headerul nivelului transport sunt folosite pentru reasamblarea pieselor de date în fluxuri pentru a fi transmise la nivelul aplicație.

7.1.1.2 Identificarea aplicațiilor

Pot exista mai multe aplicații sau servicii ce rulează pe fiecare host din rețea. Pentru a transmite fluxurile de date la aplicațiile adecvate, nivelul transport trebuie să identifice aplicația țintă (Fig. 3). Pentru a realiza acest lucru, nivelul transport atribuie fiecării aplicații un identificator. Acest identificator se numește numărul de port. Fiecare proces software ce are nevoie să acceseze rețea are atribuit un număr de port unic în acel host. Nivelul transport folosește porturile pentru a identifica aplicația sau serviciul.



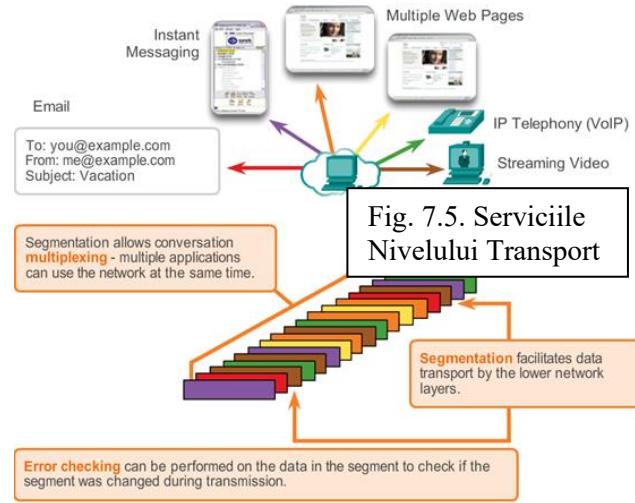
7.1.1.3 Multiplexarea conversației

Trimitera unor tipuri de date (de exemplu, un stream video) în rețea, ca un stream complet de comunicație, poate folosi toată lățimea de bandă disponibilă și poate împiedica alte comunicații să aibă loc în același timp. Face dificilă recuperarea în caz de eroare și retrasmisia datelor alterate.

Fig. arată că segmentarea datelor în piese mai mici ce permite ca mai multe comunicații diferite, de la mai mulți utilizatori, să fie intercalate (multiplexate) în aceeași rețea. Segmentarea datelor de către protocoalele de la nivelul transport oferă de asemenea mijloace de a trimite și a primi date atunci când rulează mai multe aplicații în același timp pe un computer.

Fără segmentare, numai o aplicație poate primi datele. De exemplu, cu un streaming video, mediul va fi complet ocupat de către un stream de comunicație, în schimb să fie partajat mediul. Nu putem primi emailuri, chat sau mesagerie instant sau nu putem vizualiza pagini web în timp ce vizualizăm imagini video.

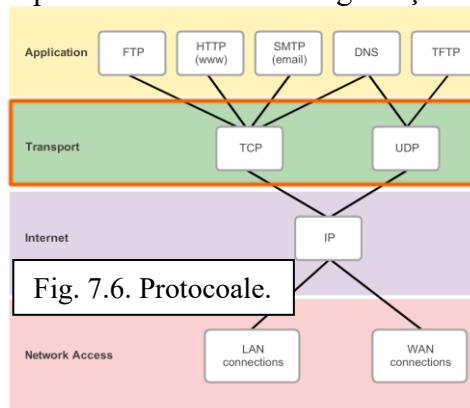
Pentru a identifica fiecare segment de date, nivelul transport adaugă segmentelor un header ce conține date binare. Acest header conține câmpuri de biți. Valorile din aceste câmpuri permit ca protocole de nivel transport diferite să efectueze diferite funcții în gestionarea comunicațiilor de date.



Nivelul transport este de asemenea responsabil de gestionarea cerințelor se siguranță ale unei conversații. Aplicații diferite au cerințe diferite de siguranță a transportului.

IP se ocupă numai de structura, adresarea și rutarea pachetelor. IP nu specifică modul în care livrarea sau transportul de pachete are lor. Protocolele de transport specifică modul în care se transferă mesajele între hosturi. TCP/IP oferă două protocole de la nivelul transport, Transmission Control Protocol (TCP) și User Datagram Protocol (UDP), aşa cum este evidențiat și în Fig. . IP folosește aceste protocole de transport pentru a permite hosturilor să comunice și să transfere datele.

TCP este considerat protocolul de la nivelul transport de încredere, cu caracteristici complete, ce asigură faptul că toate datele ajung la destinație. În opozitie, UDP este un protocol de la nivelul transport foarte simplu ce nu oferă nici-o siguranță.



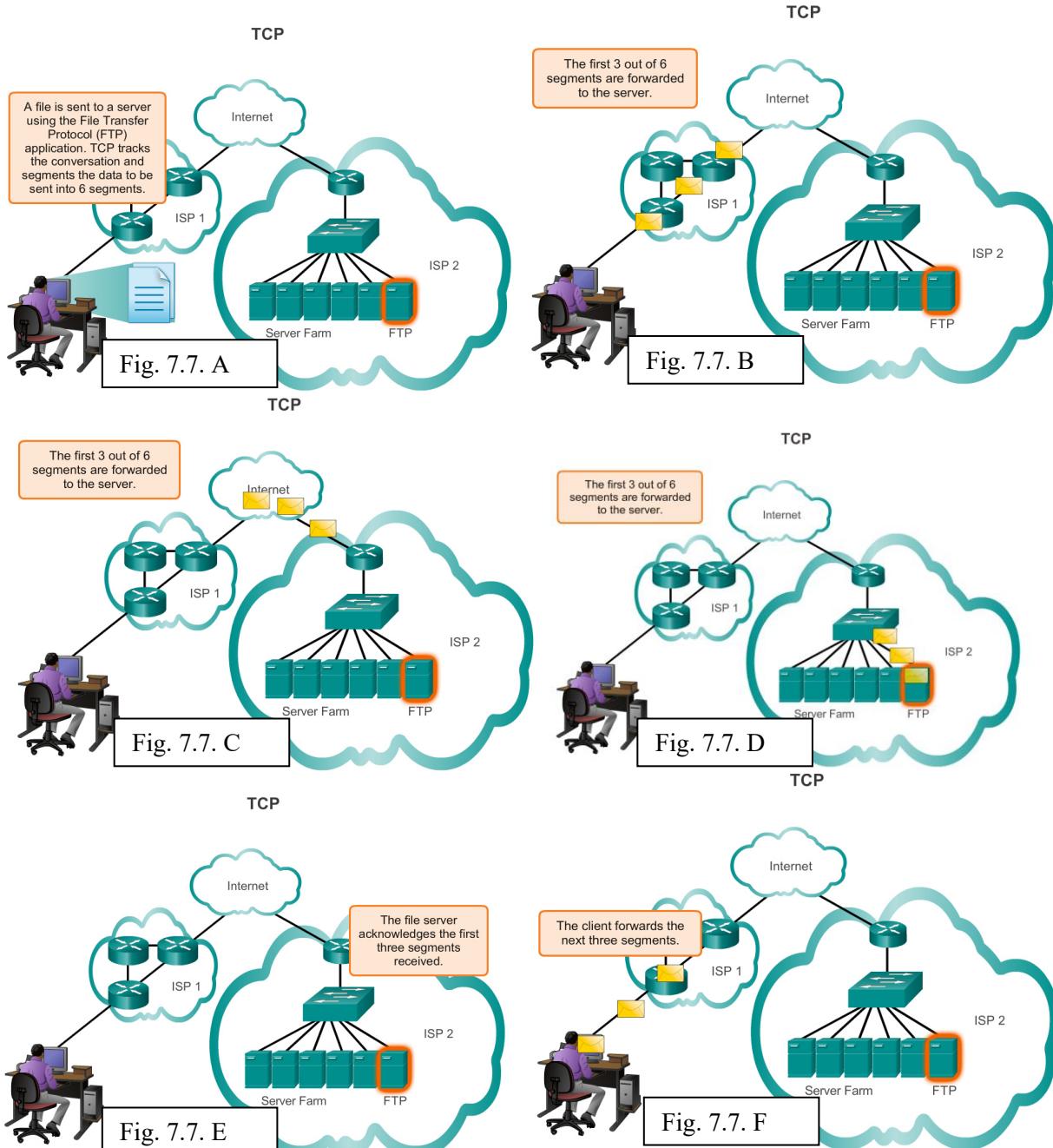
Așa cum a fost menționat mai sus, TCP este considerat un protocol de transport de încredere, ceea ce înseamnă ca TCP include procese ce asigură livrarea de încredere între aplicații prin intermediul folosirii unei livrări acknowledged. TCP este similar cu trimitera pachetelor ce sunt urmărite de la sursă la destinație. Dacă o comandă FedEx este spartă în mai multe transporturi, un client poate verifica online ordinea livrării.

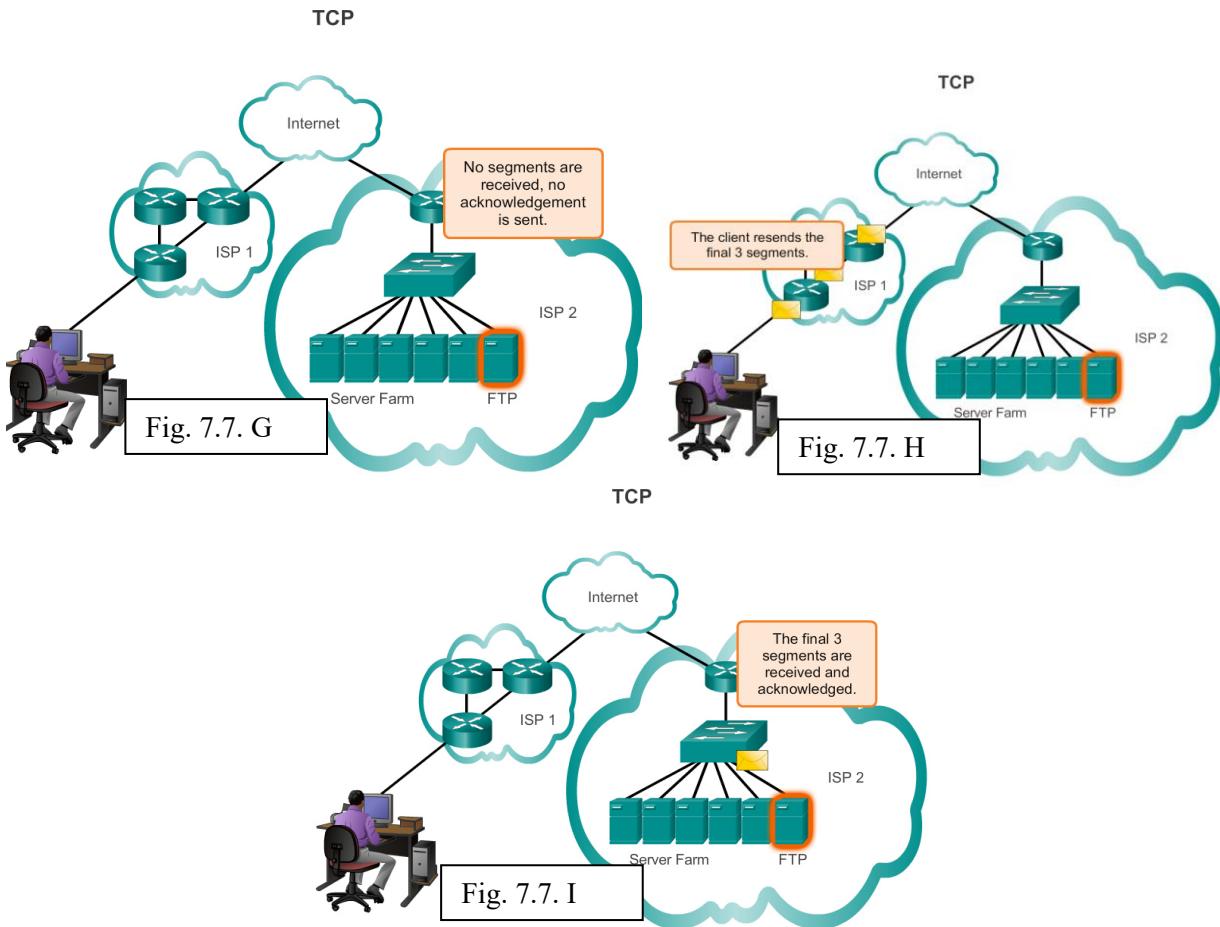
Cu TCP, cele trei operații de încredere sunt:

- *Urmărirea segmentelor de date transmise.*
- *Confirmarea datelor primite.*
- *Retransmiterea datelor neconfirmate.*

TCP “sprise” un mesaj în pieșe mai mici numite segmente. Segmentele sunt numerotate în secvențe ce sunt trimise la procesul IP pentru asamblarea în pachete. TCP urmărește numărul de segmente ce au fost transmise la un anumit host de la o anumită aplicație. Dacă expeditorul nu a primit o confirmare într-o anumită perioadă de timp, presupune că segmentele au fost pierdute și le retransmite. Numai partea mesajului pierdută este retransmisă, nu întregul mesaj. La hostul destinație, TCP este responsabil de reasamblarea segmentelor mesajului și livrarea lor la aplicație. File Transfer Protocol (FTP) și Hypertext Transfer Protocol (HTTP) sunt exemple de aplicații ce folosesc TCP pentru asigurarea livrării datelor.

Acstea procese de încredere plasează overhead suplimentar pe resursele rețelei în timpul proceselor de confirmare, urmărire și retransmisie. Pentru a suporta aceste procese de încredere, mai multe date de control sunt schimbate între sursă și destinație. Aceste informații de control sunt conținute într-un header TCP.

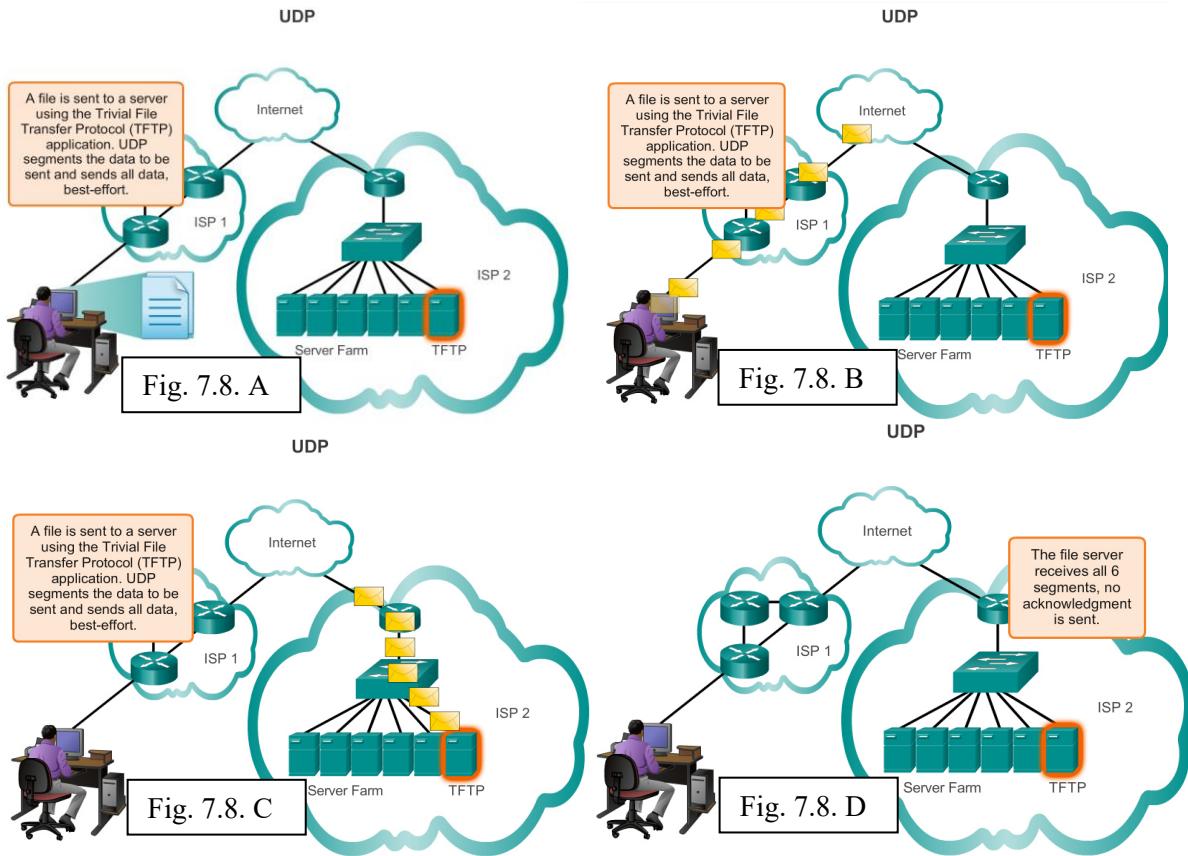




Funcțiile de încredere TCP oferă mai multe comunicații puternice între aplicații, însă de asemenea induc overhead suplimentar și posibile întârzieri în transmisie. Există un compromis între valoarea de încredere și sarcina pusă pe resursele de rețea. Overheadul ridicat pentru asigurarea încrederei pentru anumite aplicații poate reduce utilitatea aplicației și chiar să fie dăunator pentru aplicație. În aceste cazuri, UDP este un protocol de transport mai bun.

UDP oferă doar funcțiile de bază pentru livrarea segmentelor de date între aplicațiile adecvate, cu puțin overhead și verificare a datelor. UDP este cunoscut ca un protocol de livrare best-effort. În contextul rețelei, livrarea best-effort este de neîncredere deoarece nu există nici-o confirmare a faptului că datele sunt primite la destinație. Cu UDP, nu există procese de nivel transport care să informeze expeditorul de succesul livrării.

UDP este similar cu plasarea unei scrisori obișnuite, neînregistrate, în mail. Expeditorul scrisorii nu este cunoscut atunci când un destinatar este disponibil să primească scrisoarea, iar oficial poștal nu este responsabil de urmărirea scrisorii sau informarea expeditorului dacă scrisoarea nu a ajuns la destinația sa finală.



Ambele protocole, TCP și UDP, sunt protocole de transport valide. În funcție de cerințele aplicației, fie unul, fie amândouă protocoalele de transport sunt folosite. Dezvoltatorii de aplicații trebuie să aleagă ce tip de protocol de transport este adekvat, în funcție de cerințele aplicațiilor.

Pentru unele aplicații, segmentele trebuie să ajungă într-o secvență specifică pentru a fi procesate cu succes. În alte aplicații, datele trebuie să fie primite în întregime înainte de a fi considerate utile. În ambele cazuri, TCP este folosit ca protocol de transport. De exemplu, aplicațiile cum ar fi bazele de date, browsere web și clienții de e-mail necesită ca toate datele să fie primite la destinație în formatul original. Orice date care lipsesc pot cauza o comunicație alterată, care este incompletă sau nu poate fi citită. Prin urmare, aceste aplicații folosesc TCP. Overheadul de rețea suplimentar este considerat necesar pentru aceste aplicații.

În alte cazuri, o aplicație poate tolera unele pierderi de date în timpul transmisiei peste rețea, însă întârzierile în transmisie sunt inacceptabile. UDP este o alegere mai bună pentru aceste aplicații deoarece este necesar un overhead de rețea mai scăzut. UDP este preferat de aplicații cum ar fi streaming video, audio și Voice over IP (VoIP). Confirmările ar încetini livrarea și retrasmisia nu este dorită.

De exemplu, dacă unul sau două segmente ale unui stream video nu ajung, se crează o întrerupere în stream. Acest lucru ar putea apărea ca o întrerupere în Fig. , însă poate fi chiar nedetectată de utilizator. Pe de altă parte, Fig. din streamul video ar putea fi degradat mult dacă dispozitivul destinație trebuie să țină cont de datele lipsă și întârzie streamul până când sunt retrasmise. În acest caz, este mai bine să se facă cea mai bună alegere posibilă cu segmentele primite și să se renunțe la încredere.

Internet radio este un alt exemplu de aplicație ce utilizează UDP. Dacă unele mesaje sunt pierdute în timpul călătoriei prin rețea, nu sunt retrasmise. Dacă puține pachete lipsesc, ascultătorul ar putea auzi o întrerupere mică în sunet. Dacă TCP ar fi fost folosit și pachetele

pierdute ar fi retransmise, transmisia va face o pauză la receptor până când se primesc, iar întreruperea ar fi mai semnificativă.

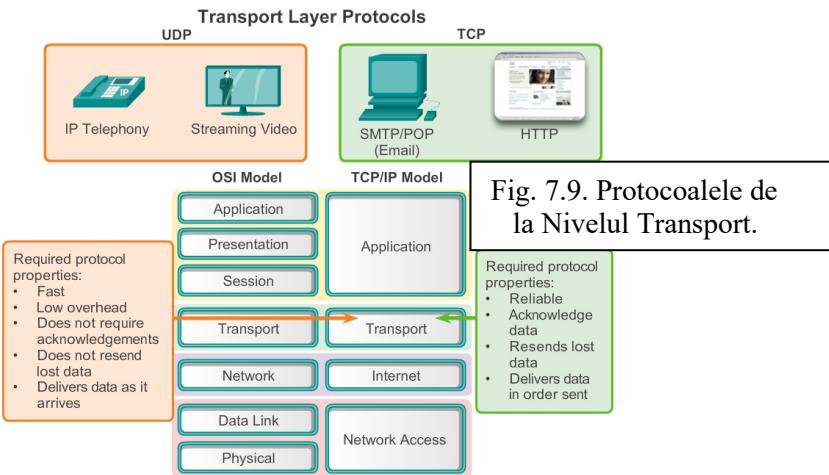


Fig. 7.9. Protocolele de la Nivelul Transport.

7.2 Introducere în lumea protocolelor de comunicație : TCP și UDP

Pentru a înțelege diferențele dintre TCP și UDP, este important să înțelegem modul în care fiecare protocol implementează funcții specifice de încredere și modul în care urmăresc comunicațiile.

7.2.1 Transmission Control Protocol (TCP)

TCP a fost descris în RFC 793. Pentru a suporta funcțiile de bază ale segmentării și reasamblării datelor, TCP oferă:

- *Conversații orientate pe conexiune pentru stabilirea sesiunilor.*
- *Livrare de încredere.*
- *Reconstrucție ordonată a datelor.*
- *Controlul fluxului.*

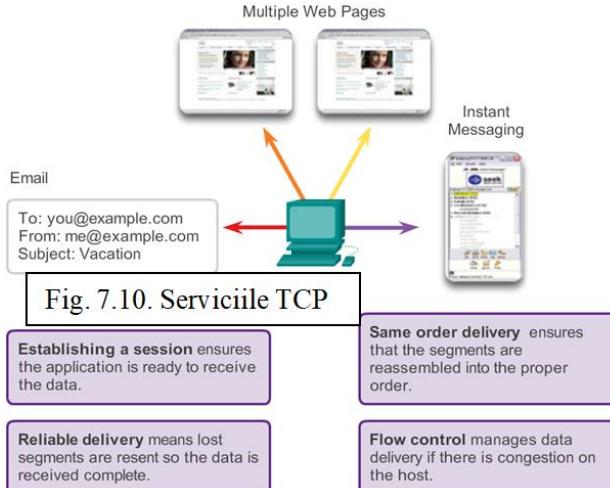
Stabilirea unei sesiuni – TCP este un protocol orientat pe conexiune. Un protocol orientat pe conexiune este acela care negociază și stabilește o conexiune permanentă (sau o sesiune) între sursă și destinație, înaintea livrării. Stabilirea sesiunii pregătește dispozitivele să comunice între ele. Prin intermediul stabilirii sesiunii, dispozitivele negociază cantitatea de trafic ce poate fi transmisă într-o perioadă de timp și datele de comunicație dintre cele două pot fi gestionate atent. Sesiunea este terminată numai după ce toate comunicațiile sunt complete.

Livrare de încredere – TCP poate implementa o metodă pentru a asigura livrarea de încredere a datelor. În termenii de rețea, încredere înseamnă asigurarea faptului că fiecare piesă de date transmisă de către sursă ajunge la destinație. Din aceste motive, este posibil ca orice piesă să devină alterată sau pierdută complet în timpul transmisiei prin rețea. TCP asigură faptul că toate piezelor ajung la destinație prin retransmiterea datelor pierdute sau alterate.

Livrare în aceeași ordine – Deoarece rețelele pot oferi mai multe rute care au diferite viteze de transmisie, datele pot ajunge într-o ordine gresita. Prin numerotarea și secvențierea segmentelor, TCP asigură faptul că aceste segmente sunt reasamblate într-o ordine adecvata.

Controlul fluxului – Hosturile de rețea au resurse limitate, cum ar fi memorie sau lățime de bandă. Deoarece TPC este cunoscut că aceste resurse sunt suprasolicitante, poate cere ca aplicațiile

expeditoare să reducă rata fluxului de date. Acest lucru se realizează de către TCP prin reglarea cantității de date transmisă de către sursă. Controlul fluxului previne pierderea de segmente din rețea și evită necesitatea retransmisiiei.



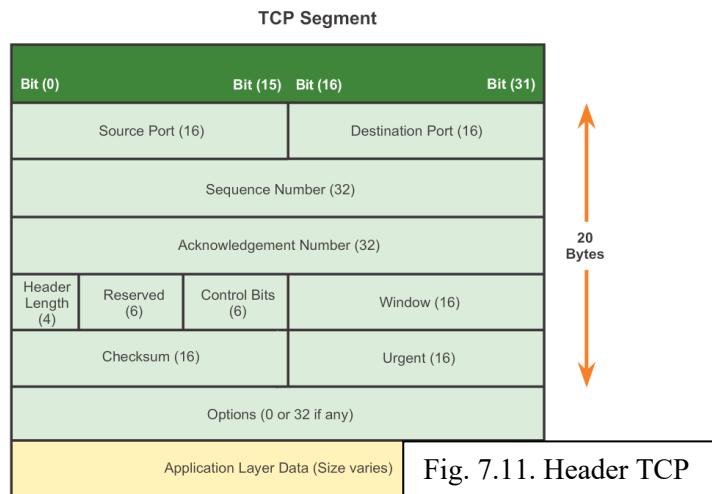
O dată ce TCP a stabilit o sesiune, este capabil să urmărească conversația din sesiune. Datorită abilității TCP de a urmări conversațiile, este considerat un protocol stateful. Un protocol stateful este un protocol ce urmărește starea sesiunii de comunicare. De exemplu, atunci când datele sunt transmise utilizând TCP, expeditorul se așteaptă ca destinația să confirme datele primite. TCP urmărește ce informații au fost trimise și ce informații au fost confirmate. Dacă datele nu sunt confirmate, expeditorul presupune că datele nu au ajuns și le retransmite. Sesiunea dinamică începe cu stabilirea sesiunii și se termină atunci când sesiunea este încheiată cu terminarea sesiunii prin confirmarea primirii tututor segmentelor.

Notă: Menținerea acestei informații de stare necesită resurse ce nu sunt necesare pentru un protocol stateless, cum ar fi UDP.

TCP suportă overhead suplimentar pentru a îndeplini aceste funcții. Așa cum este arătat și în Fig. , fiecare segment TCP are 20 bytes de overhead în headerul ce încapsulează data de la nivelul aplicație. Acest overhead este considerabil mai mare decât la un segment UDP, ce are numai 8 bytes de overhead. Overheadul suplimentar include:

- **Sequence number (32 bits)** - Folosit pentru scopuri de reasamblare a datelor.
- **Acknowledgement number (32 bits)** - Indică datele ce au fost primite.
- **Header length (4 bits)** - Cunoscut ca data offset. Indică lungimea headerului segmentului TCP.
- **Reserved (6 bits)** - Acest câmp este rezervat pentru viitor.
- **Control bits (6 bits)** - Include coduri de biți, sau flaguri, ce indică scopul și funcția segmentului TCP.
- **Window size (16 bits)** - Indica numărul de segmente ce poate fi acceptat la un moment dat.
- **Checksum (16 bits)** - Folosit pentru verificarea de eroare a headerului de segment și a datelor.
- **Urgent (16 bits)** - Indică dacă datele sunt urgente.

Exemple de aplicații ce folosesc TCP sunt browserele web, e-mail și transferul de fișiere.

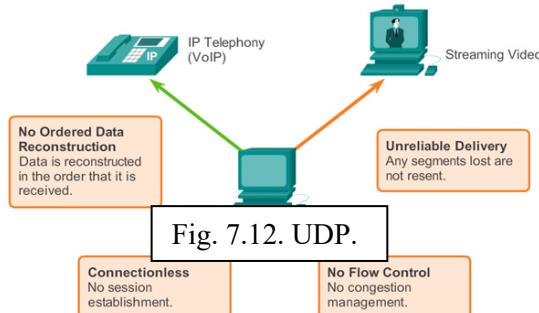


7.2.2 User Datagram Protocol (UDP)

UDP este considerat un protocol de transport best-effort, descris în RFC 768. UDP este un protocol de transport “ușor” ce oferă aceleași segmentări și reasamblări de date ca TPC, dar fără încredere și flow control ale TCP. UDP este un protocol atât de simplu încât este descris de obicei în termenii a ceea ce nu face în comparație cu TCP.

Ca și în Fig. , următoarele caracteristici descriu UDP:

- **Connectionless** – UDP nu stabilește o conexiune între hosturi înainte ca datele să fie trimise și primeite.
- **Livrare de neîncredere** – UDP nu oferă servicii pentru a asigura faptul că datele vor fi trimise într-un mod de încredere. Nu există procese în UDP prin care expeditorul trebuie să retrasmă datele pierdute sau alterate.
- **Reconstrucție a datelor neordonată** – Câteodată datele sunt primeite într-o ordine diferită decât sunt trimise. UDP nu oferă nici-un mecanism de reasamblare a datelor în secvență originală. Datele sunt livrate aplicației în ordinea în care sunt primeite.
- **Nu există control al fluxului** – Nu există mecanisme în UDP pentru controlul cantității de date transmise de către sursă pentru a evita “copleșirea” nodului destinație. Sursa trimite datele. Dacă resursele de la destinație devin suprasolicitante, hostul destinație cel mai probabil aruncă datele transmise până când resursele devin disponibile. Spre deosebire de TCP, UDP nu are niciun mecanism de reatransmisie automată a datelor aruncate.



Deși UDP nu include mecanismele de încredere și control al fluxului ca TCP, ca și în Fig. de mai jos, overheadul de livrare a datelor scăzut al UDP îl face un protocol ideal de transport pentru aplicațiile ce tolerează unele pierderi de date. Pieselete de comunicație din UDP sunt numite datagrams. Aceste datagrams sunt trimise ca best effort de către protocolul de la nivelul transport.

Unele aplicații ce folosesc UDP sunt Domain Name System (DNS), video streaming și Voice over IP (VoIP).

Una dintre cele mai importante cerințe de livrare pentru live video și voce peste rețea este continuitatea datelor de a “curge” rapid. Aplicațiile video și voce pot tolera unele pierderi de date cu un efect minim sau nedetectabil și sunt perfect potrivite pentru UDP.

UDP este un protocol stateless, însemnând că nici clientul, nici serverul nu sunt obligați să urmărească starea sesiunii de comunicație. Așa cum este arătat și în Fig. , UDP nu se ocupă de controlul fluxului sau de încredere. Datele pot fi pierdute sau primite într-o secvență diferită de cea inițială fără existența nici unui mecanism UDP de recuperare sau reordonare a datelor. Dacă încrederea este necesară atunci când se folosește UDP ca protocol de transport, trebuie gestionată de către aplicație.

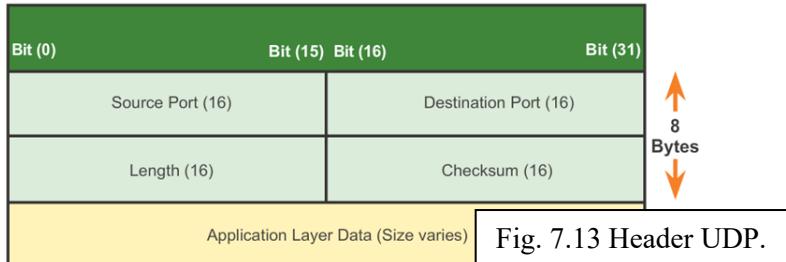


Fig. 7.13 Header UDP.

Nivelul transport trebuie să fie capabil să separe și să gestioneze comunicații multiple cu nevoi diferite de transport. De exemplu, considerăm un utilizator conectat la o rețea pe un dispozitiv final. Utilizatorul primește și trimite în mod simultan e-mail și mesaje instant, vede websites și efectuează un apel Voice over IP (VoIP). Fiecare dintre aceste aplicații trimite și primește date din rețea în același timp, indiferent de cerințele diferite de încredere. În plus, datele de la apelul telefonic nu sunt direcționate la browserul web și textul de la mesageria instant nu apare în e-mail.

Pentru încredere, utilizatorii necesită ca emailul și pagina web să fie primite complet și prezente în întregime pentru ca informațiile să fie considerate utilizabile. Întârzierile ușoare din încărcarea emailului sau paginii web sunt în general acceptabile, atât timp cât produsul final este vizualizat complet și corect. În acest exemplu, rețeaua gestionează retrimiterea și înlocuirea informațiilor lipsă și nu afișează produsul final până când totul este primit și corect asamblat.

În contrast, lipsa ocazională a unor părți mici dintr-o conversație de telefonie poate fi considerată acceptabilă. Chiar dacă unele părți mici sau puține cuvinte sunt aruncate, se poate deduce partea lipsă din context sau poate fi rugăță cealaltă persoană să repete ce a spus. Acest lucru este de preferat întârzierilor apărute în cazul în care rețeaua trebuia să gestioneze și să retrimită segmentele lipsă. În acest exemplu, utilizatorul, nu rețeaua, gestionează retrimiterea și înlocuirea informației lipsă.

Așa cum este arătat în Fig. , pentru ca TCP și UDP să gestioneze aceste conversații simultane cu cerințe variate, serviciile bazate pe TCP și UDP trebuie să urmărească diversele aplicații ce comunică. Pentru a diferenția segmentele și datagramele pentru fiecare aplicație, TCP și UDP au câmpuri de header ce pot identifica unic aceste aplicații. Acești identificatori unici sunt numere de port.

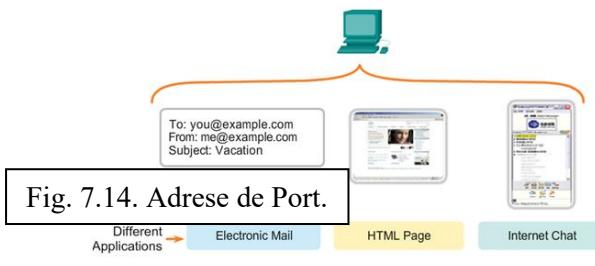


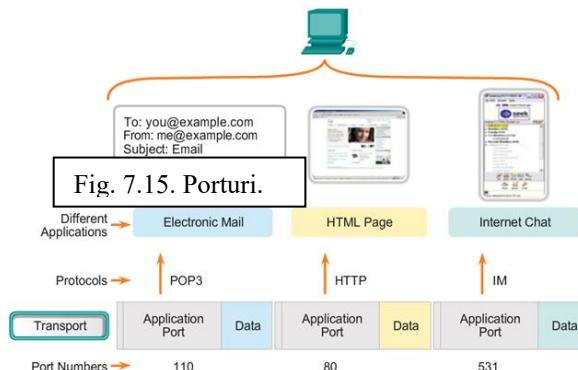
Fig. 7.14. Adrese de Port.

În headerul fiecărui segment sau datagram, există un port sursă și un port destinație. Numărul de port sursă este numărul pentru această comunicație asociat cu aplicația de origine de pe hostul local. Așa cum este arătat și în Fig. , numărul de port destinație este numărul pentru această comunicație asociat cu aplicația destinație de pe hostul de la distanță.

Atunci când un mesaj este livrat folosind fie TCP, fie UDP, protocolele și serviciile necesare sunt identificate printr-un număr de port. Un port este un identificator numeric din fiecare segment folosit pentru urmărirea unei anume conversații și serviciile destinație necesare. Fiecare mesaj trimis de un host conține un port destinație și un port sursă.

Portul destinație – Clientul plasează un număr de port destinație în segment pentru a “spune” serverului destinație ce serviciu este cerut. De exemplu, portul 80 se referă la HTTP sau serviciu web. Atunci când un client specifică portul 80 ca port destinație, serverul care primește mesajul știe că serviciile web sunt cerute. Un server poate oferi unul sau mai multe servicii simultan. De exemplu, un server poate oferi servicii web pe portul 80 în același timp în care oferă conexiune FTP stabilită pe portul 21.

Portul sursă – Numărul de port sursă este generat automat de către dispozitivul expeditor pentru a identifica o conversație dintre două dispozitive. Acest lucru permite ca mai multe conversații să aibă loc simultan. Cu alte cuvinte, un dispozitiv poate trimite mai multe cereri de serviciu HTTP către un server web în același timp. Conversațiile separate sunt urmărite în funcție de porturile sursă.



Porturile sursă și destinație sunt plasate în segment. Segmentele sunt apoi încapsulate într-un pachet IP. Pachetul IP conține adresa IP sursă și destinație. Combinarea de adrese IP sursă și destinație cu numerele de port destinație și sursă se numește socket. Socketul este folosit pentru a identifica serverul și serviciul cerut de către client. În fiecare zi, mii de hosturi comunică cu milioane de servere diferite. Aceste comunicații sunt identificate de socketuri.

Aceasta este combinația de număr de port al nivelului transport cu adresa IP a nivelului rețea ce identifică unic un proces de aplicație particular ce rulează pe un dispozitiv host individual. Această combinație se numește socket. O pereche socket, alcătuită din adresele IP și numerele de port sursă și destinație, este unică și identifică o anumita conversație dintre două hosturi.

Un socket de client ar putea arăta astfel, cu 1099 reprezentând numărul de port sursă:

192.168.1.5:1099

Socketul de pe un server web ar putea fi:

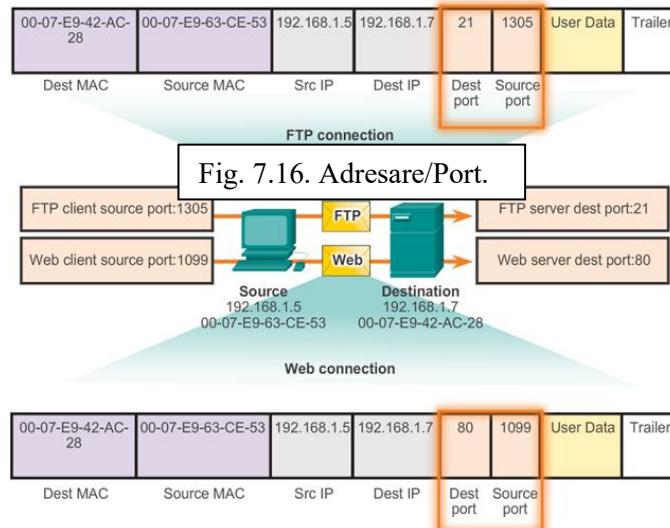
192.168.1.7:80

Împreună, aceste două socketuri se combină pentru a forma perechea socket:

192.168.1.5:1099, 192.168.1.7:80

Cu crearea de socketuri, punctele finale de comunicare sunt cunoscute astfel încât datele pot să călătorescă de la o aplicație de pe un host la o aplicație de pe alt host. Socketurile permit ca mai multe procese ce rulează pe un client să se diferențieze între ele, iar multiplele conexiuni de pe un proces server să se evidențieze între ele.

Un port sursă al unei cereri de client este generat aleator. Acest port acționează ca o adresă de întoarcere a aplicației cerute. Nivelul transport urmărește portul și aplicația ce a inițiat cererea astfel încât atunci când un răspuns este returnat, poate să fie trimis la aplicația corectă. Numărul de port al aplicație solicitată este folosit ca numărul de port destinație în răspunsul de la server.



Internet Assigned Numbers Authority (IANA) atribuie numerele de port. IANA este o asociație de standardizare responsabilă de atribuirea diferitelor standarde de adresare.

Există trei spații de numere de porturi, așa cum este evidențiat și în Fig 7.17.

- **Porturi bine-cunoscute (Numerele de la 0 la 1023)** – Aceste numere sunt rezervate pentru servicii și aplicații. Sunt de obicei folosite pentru aplicații cum ar fi HTTP (web server), Internet Message Access Protocol (IMAP)/Simple Mail Transfer Protocol (SMTP) (e-mail server) și Telnet. Prin definiția acestor porturi bine-cunoscute pentru aplicațiile server, aplicațiile client pot fi programate să ceară o conexiune la acel port specific și la serviciul său asociat.
- **Porturi înregistrate (Numerele de la 1024 la 49151)** - Aceste numere sunt rezervate pentru procesele și aplicațiile de utilizator. Aceste procese sunt aplicațiile individuale pe care un utilizator a ales să le instaleze, spre deosebire de aplicațiile cunoscute ce au număr de port bine-cunoscut. Atunci când nu sunt folosite pentru o resursă de server, aceste porturi ar putea fi folosite dinamic selectate de către un client ca port sursă.
- **Porturi private sau dinamice (Numerele de la 49152 la 65535)** – Cunoscute și ca porturi efemere, acestea sunt în mod normal asignate dinamic aplicațiilor de client atunci când clientul inițiază o conexiune la un serviciu. Portul dinamic este folosit în mod normal pentru a identifica aplicația client în timpul comunicației, în timp ce clientul folosește portul bine-cunoscut pentru a identifica și se conecta la serviciul cerut serverului. Nu este obișnuit pentru un client să se conecteze la un serviciu folosind un port dinamic sau privat (deși unele programe de partajare de fișiere peer-to-peer folosesc aceste porturi).

Fig. 7.17. Spații de adrese pentru porturi.

Port Number Range	Port Group
0 to 1023	Well-known Ports
1024 to 49151	Registered Ports
49152 to 65535	Private and/or Dynamic Ports

Fig. 7.19 afișează unele porturi bine-cunoscute și înregistrate din TCP. Fig. 7.20 afișează unele porturi bine-cunoscute și înregistrate din UDP, iar Fig. 7.21 prezintă unele porturi utilizate de ambele protocole.

7.3 Utilizarea ambelor protocole TCP și UDP

Unele aplicații ar putea folosi atât TCP, cât și UDP. De exemplu, overheadul scăzut al UDP permite DNS să răspundă la mai multe cereri client foarte rapid. Uneori, însă, trimiterea informațiilor cerute ar putea necesita încrederea TCP. În acest caz, numărul de port bine-cunoscut, 53, este folosit de ambele, TCP și UDP, cu acest serviciu.

O listă curentă de numere de port și aplicațiile asociate poate fi găsită pe website-ul IANA.

Port Number Range	Port Group
0 to 1023	Well-known Ports
1024 to 49151	Registered Ports
49152 to 65535	Private and/or Dynamic Ports

Legend

Registered TCP Ports:	Well-known TCP Ports:
1863 MSN Messenger	21 FTP
2000 Cisco SCCP (VoIP)	23 Telnet
8008 Alternate HTTP	25 SMTP
8080 Alternate HTTP	80 HTTP
	143 IMAP
	194 Internet Relay Chat (IRC)
	443 Secure HTTP (HTTPS)

Fig. 7.19. Porturi utilizate de TCP

Port Number Range	Port Group
0 to 1023	Well-known Ports
1024 to 49151	Registered Ports
49152 to 65535	Private and/or Dynamic Ports

Legend

Registered UDP Ports:	Well-known UDP Ports:
1812 RADIUS Authentication Protocol	69 TFTP
5004 RTP (Voice and Video Transport Protocol)	520 RIP
5040 SIP (VoIP)	

Fig. 7.20. Porturi utilizate de UDP

Port Number Range	Port Group
0 to 1023	Well-known Ports
1024 to 49151	Registered Ports
49152 to 65535	Private and/or Dynamic Ports

Legend

Registered TCP/UDP Common Ports:	Well-known TCP/UDP Common Ports:
1433 MS SQL	53 DNS
2948 WAP (MMS)	161 SNMP
	531 AOL Instant Messenger, IRC

Fig. 7.20. Porturi utilizate de TCP/UDP

Uneori este necesar să știm ce conexiune TCP activă este deschisă și rulează pe un host. **Netstat** este o utilitarul de rețea important ce poate fi folosit pentru a verifica acele conexiuni. **Netstat** listează protocolul folosit, adresa locală și numărul de port, adresa de la distanță și numărul de port și starea conexiunii.

Conexiunile TCP inexplicabile pot prezenta o amenințare de securitate importantă, deoarece pot indica faptul că cineva sau ceva este conectat la hostul local. În plus, conexiunile TCP ce nu sunt necesare pot consuma resurse de sistem valoroase, astfel încetind performanța hostului. **Netstat** ar putea fi folosit pentru a examina conexiunile deschise pe un host atunci când performanța pare să fie compromisă.

Mai multe opțiuni utile sunt disponibile pentru comanda **netstat**.

C:\> netstat				
Active Connections				
Proto	Local Address	Foreign Address	State	
TCP	kenpc:3126	192.168.0.2:netbios-ssn	ESTABLISHED	
TCP	kenpc:3158	207.138.126.152:http	ESTABLISHED	
TCP	kenpc:3159	207.138.126.169:http	ESTABLISHED	
TCP	kenpc:3160	207.138.126.169:http	ESTABLISHED	
TCP	kenpc:3161	sc.msn.com:http	ESTABLISHED	
TCP	kenpc:3166	www.cisco.com:http	ESTABLISHED	

C:\> netstat				
Active Connections				
Proto	Local Address	Foreign Address	State	
TCP	kenpc:3126	192.168.0.2:netbios-ssn	ESTABLISHED	
TCP	kenpc:3158	207.138.126.152:http	ESTABLISHED	
TCP	kenpc:3159	207.138.126.169:http	ESTABLISHED	
TCP	kenpc:3160	207.138.126.169:http	ESTABLISHED	
TCP	kenpc:3161	sc.msn.com:http	ESTABLISHED	
TCP	kenpc:3166	www.cisco.com:http	ESTABLISHED	

Fig. 7.22. Utilizarea comenzi netstat

C:\> netstat				
Active Connections				
Proto	Local Address	Foreign Address	State	
TCP	kenpc:3126	192.168.0.2:netbios-ssn	ESTABLISHED	
TCP	kenpc:3158	207.138.126.152:http	ESTABLISHED	
TCP	kenpc:3159	207.138.126.169:http	ESTABLISHED	
TCP	kenpc:3160	207.138.126.169:http	ESTABLISHED	
TCP	kenpc:3161	sc.msn.com:http	ESTABLISHED	
TCP	kenpc:3166	www.cisco.com:http	ESTABLISHED	

Fig. 7.23. Verificare după nume

C:\> netstat				
Active Connections				
Proto	Local Address	Foreign Address	State	
TCP	kenpc:3126	192.168.0.2:netbios-ssn	ESTABLISHED	
TCP	kenpc:3158	207.138.126.152:http	ESTABLISHED	
TCP	kenpc:3159	207.138.126.169:http	ESTABLISHED	
TCP	kenpc:3160	207.138.126.169:http	ESTABLISHED	
TCP	kenpc:3161	sc.msn.com:http	ESTABLISHED	
TCP	kenpc:3166	www.cisco.com:http	ESTABLISHED	ESTABLISHED

Fig. 7.24. Verificare conexiunii.

Într-un capitol anterior a fost explicitat modul în care **protocol data units** (PDUs) sunt construite prin trecerea de date de la aplicație la nivelele inferioare pentru a crea un PDU ce este apoi transmis peste mediu de comunicație. La hostul destinație, acest proces este inversat până când datele ajung la aplicație.

Unele aplicații transmit cantități mari de date, în unele cazuri, mai mulți gigabytes. Este nepractic să transmitem toate aceste date într-o singură bucată mare. Nici-un alt trafic de rețea nu ar mai putea fi transmis până când datele nu sunt livrate. O piesă mare de date poate dura minute sau ore să fie livrată. Prin urmare, dacă există erori, întregul fișier de date va fi pierdut sau va trebui retransmis. Dispozitivele de rețea nu ar avea buffere de memorie atât de mari încât să stocheze aceasta mare cantitate de date atunci când este transmisă sau primită. Această limită variază în funcție de tehnologia de rețea și de mediul fizic utilizat.

Divizarea datelor aplicației în segmente asigură faptul că datele sunt transmise în limitele mediului și faptul că datele de la diferite aplicații pot fi multiplexate pe mediu.

7.4 TCP și UDP procesează segmentări diferențiate

Așa cum este arătat și în Fig. , fiecare header de segment TCP conține un număr de secvență ce permite funcțiilor nivelului transport de pe hostul destinație săreasambleze segmentele în ordinea în care au fost transmise. Acest lucru asigură faptul că aplicația destinație are datele în forma exactă dorită de expeditor.

Însă serviciile ce folosesc UDP urmăresc de asemenea conversațiile dintre aplicații; nu se preocupă de ordinea în care informațiile au fost transmise sau de menținerea unei conexiuni. Nu există nici-un număr de secvență în headerul UDP. UDP este cu design mai simplu și generează overhead mai puțin decât TCP, având ca rezultat un transfer de date mai rapid.

Informațiile ar putea ajunge într-o ordine diferită față de cea în care au fost transmise, datorită faptului că pachetele diferite ar putea urma diferite căi prin rețea. O aplicație ce folosește UDP trebuie să tolereze faptul că datele ar putea să nu ajungă în ordinea în care au fost transmise.

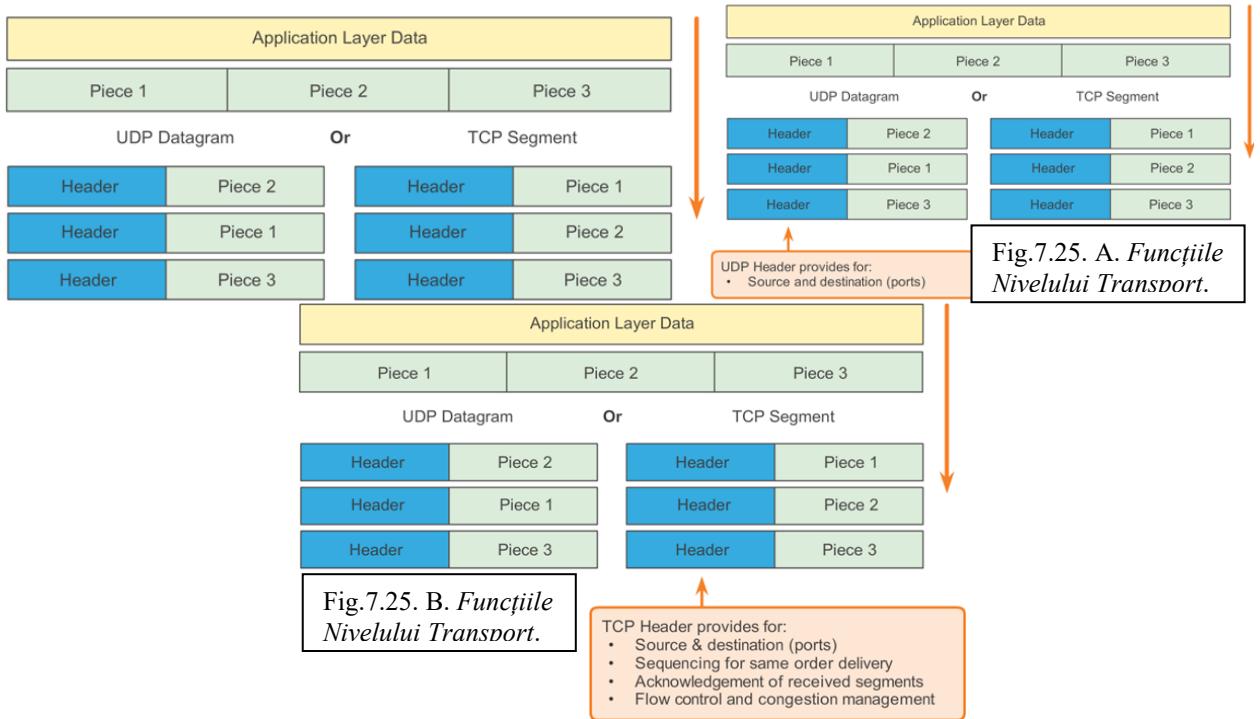


Fig.7.25. A. Funcțiile Nivelului Transport.

Fig.7.25. B. Funcțiile Nivelului Transport.

7.5 Comunicații TCP

Diferența cheie dintre TCP și UDP este încrederea. Încrederea comunicației TCP este obținuta prin folosirea sesiunii orientată pe conexiune. Înainte ca un host ce folosește TCP să trimită datele unui alt host, TCP inițiază un proces pentru a crea o conexiune cu destinația. Această conexiune stateful permite urmărirea unei sesiuni sau streamului de comunicare dintre hosturi. Acest proces asigură faptul că fiecare host este conștient și pregătit pentru streamul de comunicare. O conversație TCP necesită stabilirea unei sesiuni dintre hosturi în ambele direcții.

După ce o sesiune a fost stabilită, transferul de date începe și destinația trimite confirmări sursei pentru segmentele primite. Aceste confirmări formează baza încrederii din sesiunea TCP. Atunci când sursa primește o confirmare, știe că datele au fost livrate cu succes și poate renunța la urmărirea datelor. Dacă sursa nu a primit o confirmare într-un anumit interval de timp, retransmite datele destinației.

O parte a overdeahului suplimentar generat de folosirea TCP este traficul de rețea generat de confirmări și retransmisii. Stabilirea sesiunilor crează overhead în forma de segmente adiționale schimbate între hosturi. Există de asemenea overhead suplimentar pe hosturile individuale creat de necesitatea urmăririi a căror segmente așteaptă confirmare și de procesul de retransmisie.

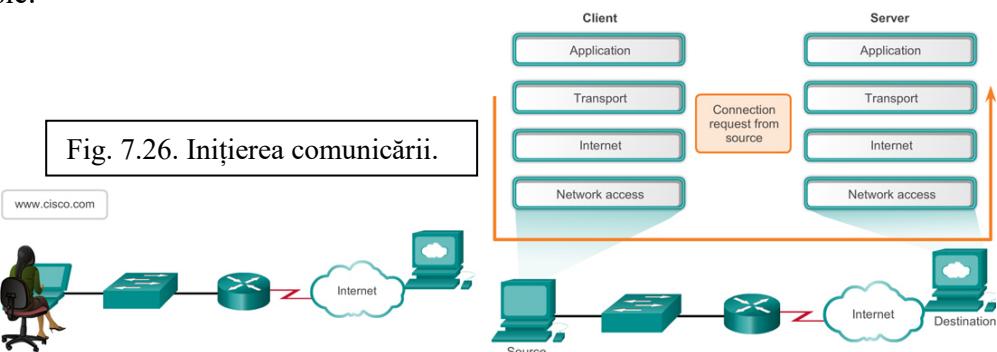
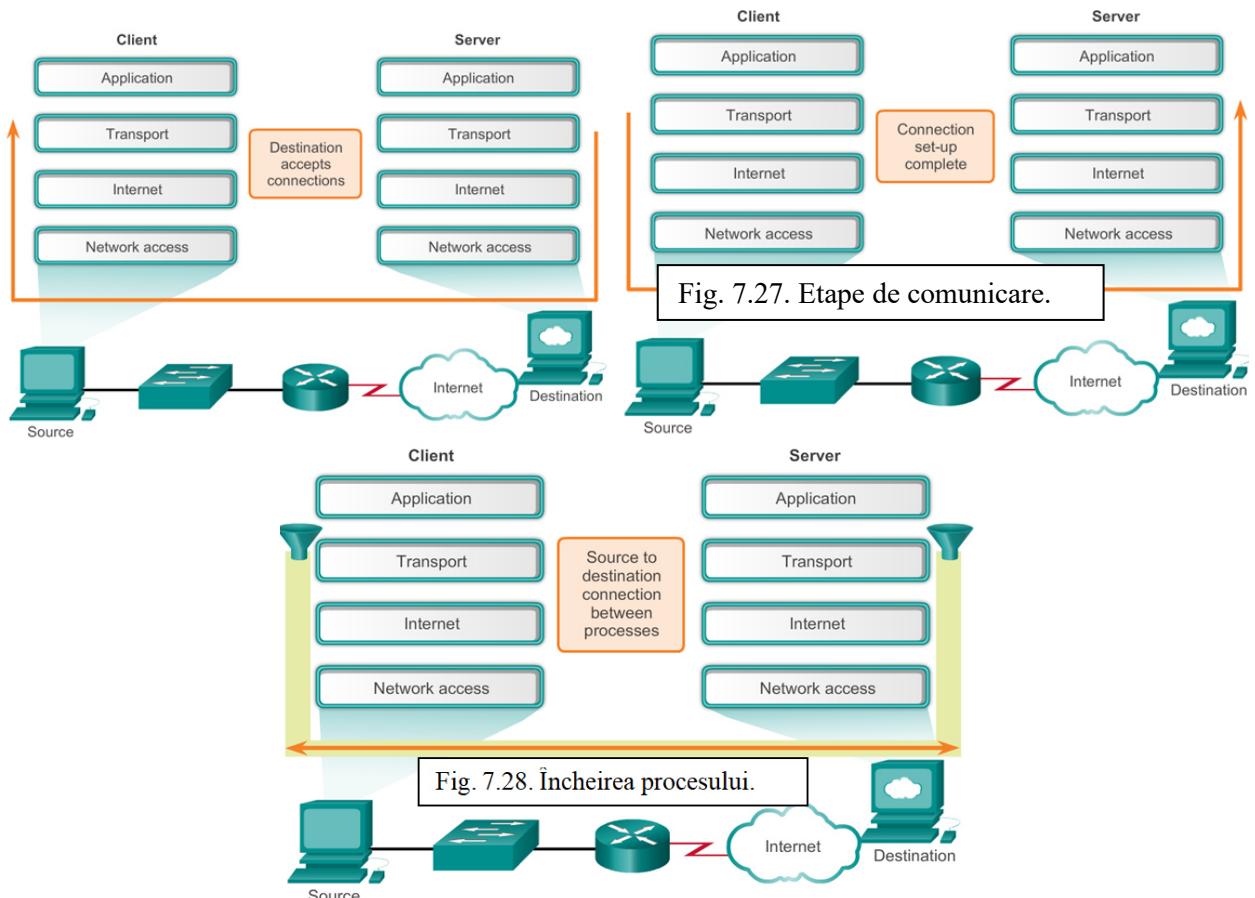


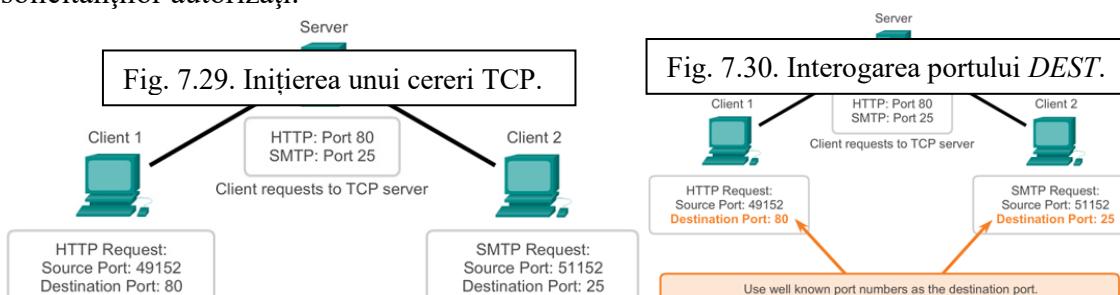
Fig. 7.26. Inițierea comunicării.

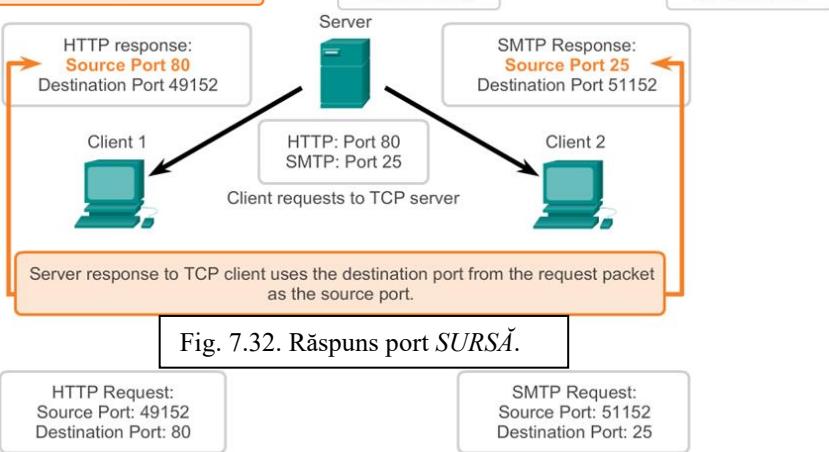
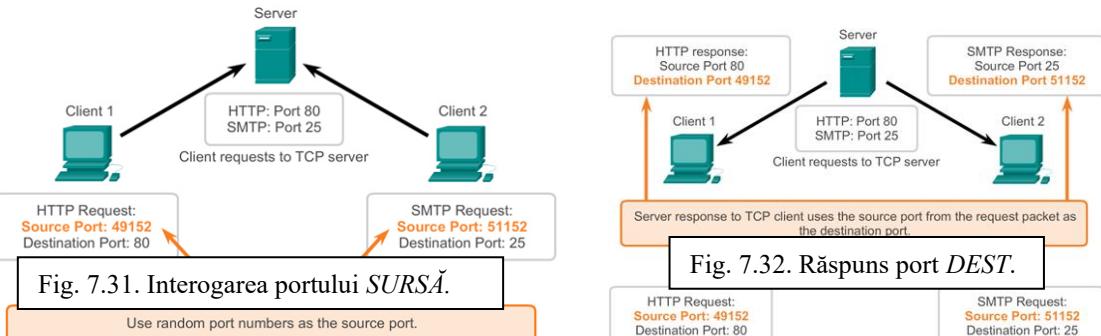


Procesele aplicație rulează pe servere. Un singur server ar putea rula mai multe procese aplicație în același timp. Aceste procese așteptă până când un client inițiază comunicația cu o cerere de informații sau alte servicii.

Fiecare proces aplicație ce rulează pe server este configuraționat cu un număr de port, implicit sau manual de către un administrator de sistem. Un server individual nu poate avea două servicii ce au atribuit același număr de port cu aceleași servicii de nivel transport. Un host ce rulează o aplicație de server web și o aplicație de transfer de fișier nu poate avea configurație același port (de exemplu, portul TCP 8080). O aplicație activă de server ce are atribuit un anumit port este considerată deschisă, ceea ce înseamnă că nivelul transport acceptă și procesează segmente adresate portului respectiv. Orice cerere de client primită adresată socketului corect este acceptată și datele sunt trimise aplicației server. Pot exista mai multe porturi deschise simultan pe un server, unul pentru fiecare aplicație activă de pe server. Este obișnuit pentru un server să ofere mai mult de un serviciu în același timp, cum ar fi un server web și un server FTP.

Un mod de îmbunătățire a securității pe un server este restricționarea accesului la server numai pe acele porturi asociate cu serviciile și aplicațiile ce ar trebui să fie accesibile solicitanților autorizați.





În unele culturi, atunci când două persoane se întâlnesc, de multe ori se salută între ei prin strângerea de mâină. Acțiunea de strângere de mâină este înțeleasă de ambele părți ca un semnal de prietenie. Conexiunile dintr-o rețea sunt similare. Prima strângere de mâină cere sincronizarea. A doua strângere de mâină confirmă cererea de sincronizare inițială și sincronizează parametrii de conexiune în direcția opusă. A treia strângere de mâină este o confirmare folosită pentru a informa destinatarul că ambele părți confirmă faptul că o conexiune a fost stabilită.

Atunci când două hosturi comunică folosind TCP, o conexiune este stabilită înainte ca datele să fie schimbată. După ce comunicarea este completă, sesiunile sunt închise și conexiunea este terminată. Mecanismele de sesiune și conexiune permit funcția de încredere TCP.

Hosturi urmăresc fiecare segment de date dintr-o sesiune și schimbă informații despre ce date sunt primite folosind informațiile din headerul TCP. TCP este un protocol full-duplex, în care fiecare conexiune reprezintă două fluxuri de comunicație într-o singură direcție, sau sesiuni. Pentru a stabili conexiunea, hosturile efectuează three-way handshake. Biții de control din headerul TCP indică progresele și statusul conexiunii. Three-way handshake:

- Stabilește faptul că dispozitivul destinație este prezent în rețea.
- Verifică faptul că dispozitivul destinație are un serviciu activ și că acceptă cererile pe numărul de port destinație pe care clientul încearcă să îl folosească pentru sesiune.
- Informează dispozitivul destinație de faptul că sursa intenționează să stabilească o sesiune de comunicare pe numărul de port respectiv.

În conexiunile TCP, clientul stabilește conexiunea cu serverul. Cei trei pași în stabilirea conexiunii sunt:

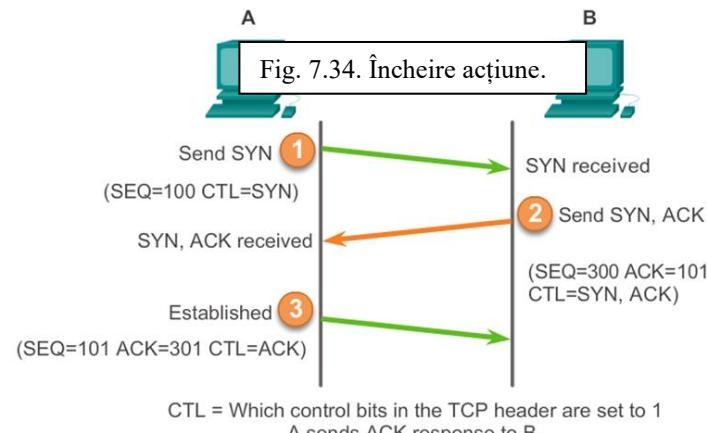
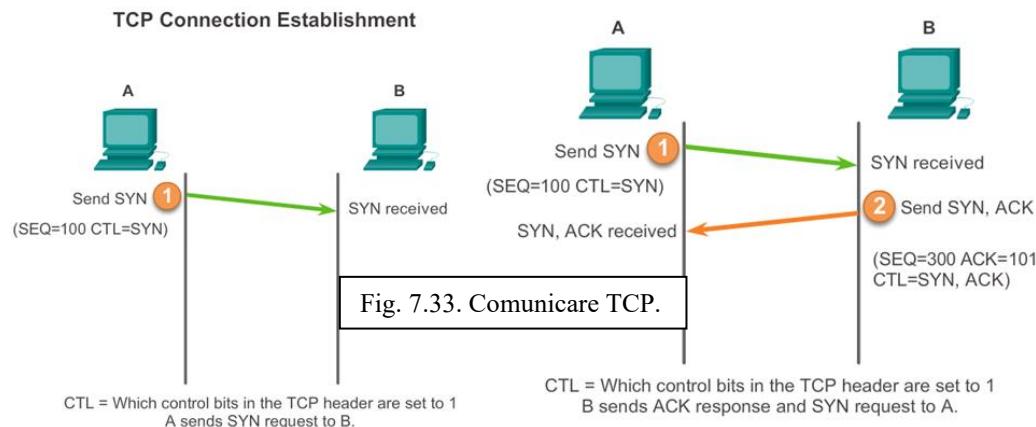
Pasul 1. Clientul solicită o sesiune de conexiune client-server cu serverul.

Pasul 2. Serverul recunoaște sesiunea de conexiune client-server și cere o sesiune de conexiune client-server.

Pasul 3. Clientul recunoaște sesiunea de conexiune client-server.

Pentru a înțelege procesul three-way handshake, observăm valorile diferite pe care cele două hosturi le schimbă între ele. În headerul segmentului TCP, există șase câmpuri de 1 bit ce conțin informații de control folosite pentru gestionarea proceselor TCP. Acele câmpuri sunt:

- **URG** - Urgent pointer field significant.
- **ACK** - Acknowledgement field significant.
- **PSH** - Push function.
- **RST** - Reset the connection.
- **SYN** - Synchronize sequence numbers.
- **FIN** - No more data from sender.



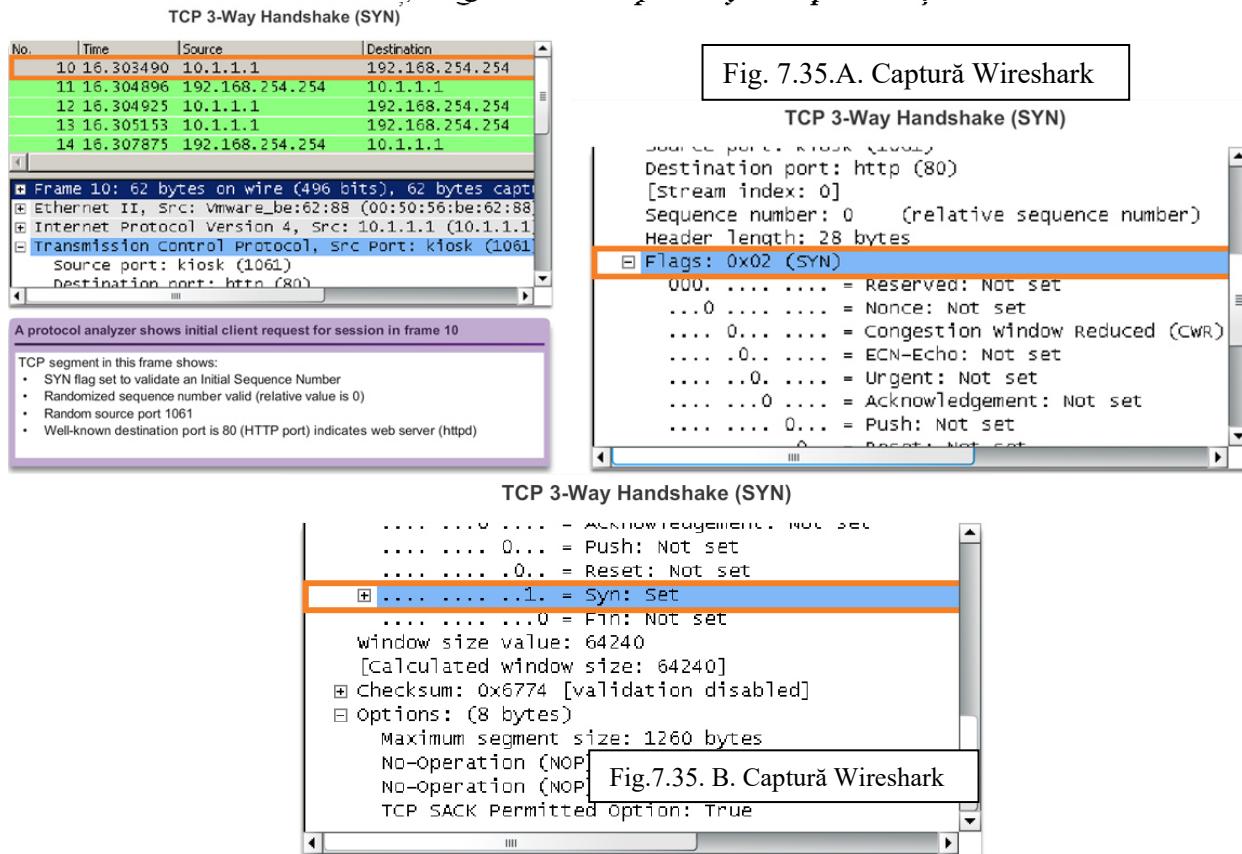
Folosind outputul softwareului de analiză, cum ar fi ieșirile Wireshark, putem examina acțiunea TCP 3-way handshake:

Pasul 1. Clientul solicită o sesiune de comunicare client-server cu serverul.

Un client TCP începe three-way handshake prin trimiterea unui “*synchronize sequence number*” (SYN) control flag set, indicând o valoare inițială în câmpul sequence number din header. Această valoare pentru numărul de secvență, numită și “*initial sequence number*” (ISN), este aleasă aleator și este folosită pentru a începe urmărirea fluxului de date de la client la server pentru această sesiune. ISN din header pentru fiecare segment crește cu unu pentru fiecare byte de date trimise de la client la server în timpul conversației.

Așa cum este evidențiat și în Fig. , outputul analizorului de protocol arată SYN control flag și numărul de secvență respectiv.

SYN control flag este setat, iar numărul de secvență este 0. Deși analizorul de protocol din grafic indică valorile relative pentru numele de secvență și acknowledgement, valorile reale sunt numere binare de 32 de biți. Fig. 7.35 arată patru bytes reprezentați în hexazecimal.



Pasul 2. Serverul recunoaște sesiunea de conexiune client-server și cere o sesiune de conexiune client-server.

Serverul TCP trebuie să confirme primirea segmentului SYN de la client și să stabilească sesiunea client-server. Pentru a face acest lucru, serverul trimite un segment înapoi la client cu acknowledgement (ACK) flag indicând faptul că numărul de confirmare este semnificativ. Cu acest flag setat în segment, clientul îl recunoaște ca o confirmare a faptului că serverul a primit SYN de la clientul TCP.

Valoarea câmpului numărul de confirmare este egală cu ISN plus 1. Acest lucru stabilește o sesiune de la client la server. ACK flag rămâne setat pentru balanța sesiunii. Reamintim faptul că reala conversație dintre client și server constă de fapt din două sesiuni într-o singură direcție: una de la client la server și alta de la server la client. În acest pas doi al three-way handshake, serverul trebuie să inițieze răspunsul clientului. Pentru a începe această sesiune, serverul folosește SYN flag în același mod în care a fost folosit de client. Setează SYN control flag în header pentru a stabili o sesiune de la server la client. SYN flag indică faptul că valoarea inițială a câmpului numărului de secvență este în header. Această valoare este folosită pentru urmărirea fluxului de date din sesiune de la server la client.

Așa cum este evidențiat și în Fig.7.36 de mai jos, ieșirea analizorului de protocol arată faptul că ACK și SYN control flags sunt setate și numele de secvență și confirmare respective sunt afișate.

Fig. 7.36. A. Setarea steagurilor.

TCP 3-Way Handshake (SYN, ACK)

No.	Time	Source	Destination
10	16.303490	10.1.1.1	192.168.254.254
11	16.304896	192.168.254.254	10.1.1.1
12	16.304925	10.1.1.1	192.168.254.254
13	16.305153	10.1.1.1	192.168.254.254
14	16.307875	192.168.254.254	10.1.1.1

A protocol analyzer shows server response in frame 11

- ACK flag set to indicate a valid Acknowledgement number
- Acknowledgement number response to initial sequence number as relative value of 1
- SYN flag set to indicate the Initial Sequence Number for the server to client session
- Destination port number of 1061 to correspond to the clients source port
- Source port number of 80 (HTTP) indicating the web server service (httpd)

TCP 3-Way Handshake (SYN, ACK)

Source port: http (80)
destination port: kiosk (1061)
[Stream index: 0]
Sequence number: 0 (relative sequence number)
Acknowledgement number: 1 (relative ack number)
Header length: 28 bytes
Flags: 0x12 (SYN, ACK)

000. = Reserved: Not set
...0 = Nonce: Not set
.... 0.... . = Congestion Window Reduced (CWR)
.... .0. = ECN-Echo: Not set
.... .0. = Urgent: Not set

TCP 3-Way Handshake (SYN, ACK)

.... 0... = Push: Not set
....0... = Reset: Not set
....1. = Syn: Set
....0 = Fin: Not set
window size value: 5840
[calculated window size: 5840]
Checksum: 0x4159 [validation disabled]
Options: (8 byte)
[SEQ/ACK analysis] Fig. 7.36. B. Setarea steagurilor.
[This is an ACK to the segment in frame: 10]
[The RTT to ACK the segment was: 0.001406000 sec]

Pasul 3. Clientul recunoaște sesiunea de conexiune client-server.

În final, clientul TCP răspunde cu un segment ce conține un ACK ce reprezintă răspunsul la TCP SYN trimis de către server. Nu există date de utilizator în acest segment. Valoarea din câmpul numărului de confirmare ISN plus 1 față de cel primit de la server. După ce ambele sesiuni sunt stabilite între client și server, toate segmentele suplimentare schimbă în această conversație vor avea ACK flag setat.

Așa cum este evidențiat și în Fig.7.37, de mai jos, ieșirea analizorului de protocol arată ACK control flag setat și numerele de secvență și confirmare relative.

Securitatea poate fi adăugată în rețeaua de date prin:

- Refuzarea stabilirii de sesiuni TCP.
- Permiterea stabilirii sesiunilor numai pentru anumite servicii.
- Permiterea traficului ce face parte din sesiunile deja stabilite.

Acste măsuri de securitate pot fi implementate pentru toate sesiunile TCP sau numai pentru anumite sesiuni.

Fig. 7.37. A. Setare steag ACK

TCP 3-Way Handshake (ACK)

No.	Time	Source	Destination
10	16.303490	10.1.1.1	192.168.254.254
11	16.304896	192.168.254.254	10.1.1.1
12	16.304925	10.1.1.1	192.168.254.254
13	16.305153	10.1.1.1	192.168.254.254
14	16.307875	192.168.254.254	10.1.1.1

A protocol analyzer shows client response to session in frame 12

The TCP segment in this frame shows:

- ACK flag set to indicate a valid Acknowledgement number
- Acknowledgement number response to initial sequence number as relative value of 1
- Source port number of 1061 to correspond
- Destination port number of 80 (HTTP) indicating the web server service (httpd)

TCP 3-Way Handshake (ACK)

Source port: kiosk (1061)
destination port: http (80)
[Stream index: 0]
Sequence number: 1 (relative sequence number)
Acknowledgement number: 1 (relative ack number)
Header length: 20 bytes
Flags: 0x10 (ACK)

000. = Reserved: Not set
...0 = Nonce: Not set
.... 0.... . = Congestion Window Reduced (CWR)
.... .0. = ECN-Echo: Not set
.... .0. = Urgent: Not set

TCP 3-Way Handshake (ACK)

```

.... .1 .... = Acknowledgement: set
.... .0... = Push: Not set
.... .0.. = Reset: Not set
.... .0...
.... .0...
window size value: 64240
[calculated window size: 64240]
>window size scaling factor: -2 (no window scaling)
 Checksum: 0x89fc [validation disabled]
 [SEQ/ACK analysis]
[This is an ACK to the segment in frame: 11]
[The RTT to ACK the segment was: 0.00029000 sec]

```

Fig. 7.37. B. Setare steag ACK

Pentru a închide o conexiune, Finish (FIN) control flag trebuie să fie setat în headerul segmentului. Pentru a finaliza fiecare sesiune TCP, un two-way handshake este folosit, constând dintr-un segment FIN și un segment ACK. Prin urmare, pentru a termina o conversație suportată de TCP, patru schimburi sunt necesare pentru finalizarea ambelor sesiuni.

Notă: În aceasta explicație, termenii de client și server sunt folosiți ca o referință pentru simplitate, dar procesul de finalizare poate fi inițiat de oricare dintre cele două hosturi care au o sesiune deschisă:

Pasul 1. Atunci când clientul nu mai are date de transmis, trimite un segment cu FIN flag setat.

Pasul 2. Serverul trimite un ACK pentru a confirma primirea FIN de la client la server.

Pasul 3. Serverul trimite un FIN la client, pentru a termina sesiunea de la server la client.

Pasul 4. Clientul răspunde cu un ACK pentru a confirma FINul de la server.

Atunci când clientul nu mai are date de transmis, setează FIN flag în headerul segmentului. Apoi, serverul trimite un segment normal ce conține date cu ACK flag setat, folosind numărul de confirmare, confirmând faptul că toți bytes de date au fost primiți. Când toate segmentele au fost confirmate, sesiunea este închisă.

Sesiunea din direcția opusă este închisă cu ajutorul același proces. Receptorul indică faptul că nu mai există date de transmis prin setarea FIN flag în headerul segmentului trimis sursei. O confirmare indică faptul că toți bytes de date au fost primiți și sesiunea este închisă.

Este de asemenea posibilă finalizarea conexiunii prin three-way handshake. Atunci când clientul nu mai are date de transmis, trimite un FIN serverului. Dacă serverul nu mai are date de transmis, poate răspunde cu FIN și ACK setate, combinând doi pași într-un pas. Clientul apoi răspunde cu un ACK.

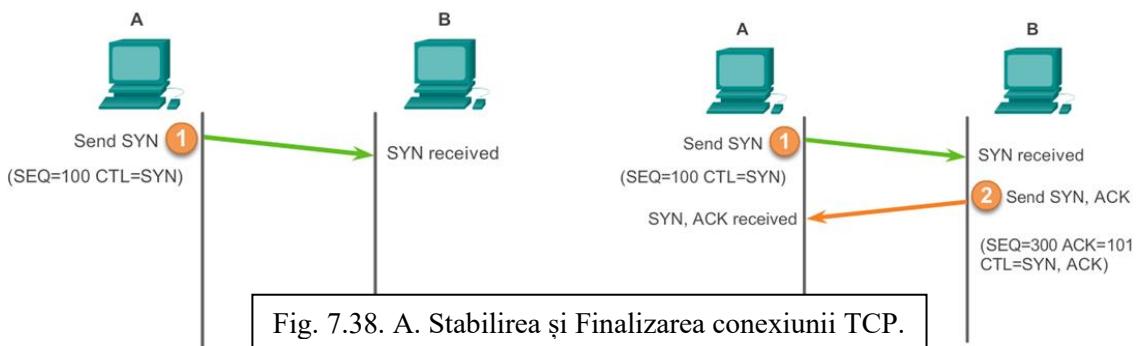


Fig. 7.38. A. Stabilirea și Finalizarea conexiunii TCP.

CTL = Which control bits in the TCP header are set to 1
A sends SYN request to B.

CTL = Which control bits in the TCP header are set to 1
B sends ACK response and SYN request to A.

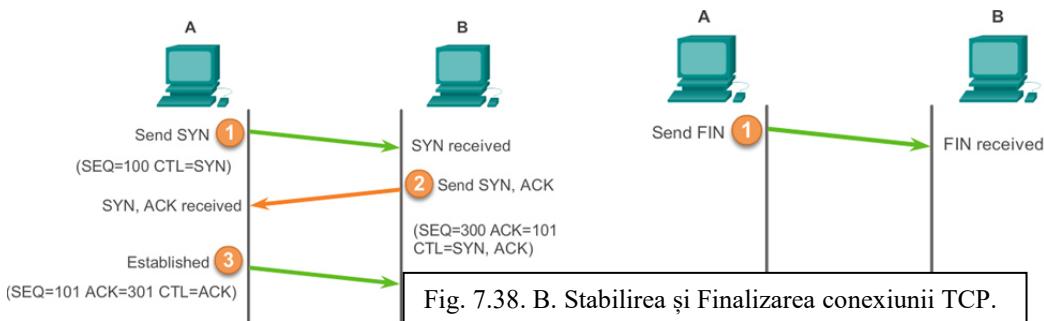


Fig. 7.38. B. Stabilirea și Finalizarea conexiunii TCP.

CTL = Which control bits in the TCP header are set to 1
A sends ACK response to B.

A sends FIN request to B.

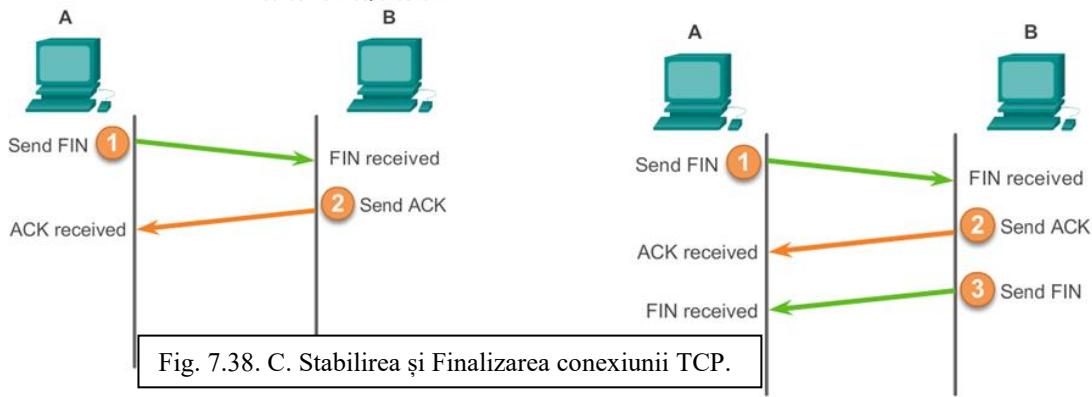


Fig. 7.38. C. Stabilirea și Finalizarea conexiunii TCP.

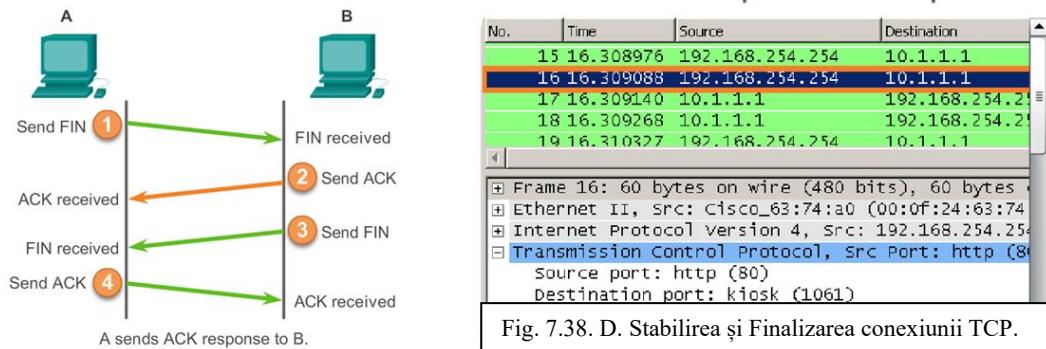


Fig. 7.38. D. Stabilirea și Finalizarea conexiunii TCP.

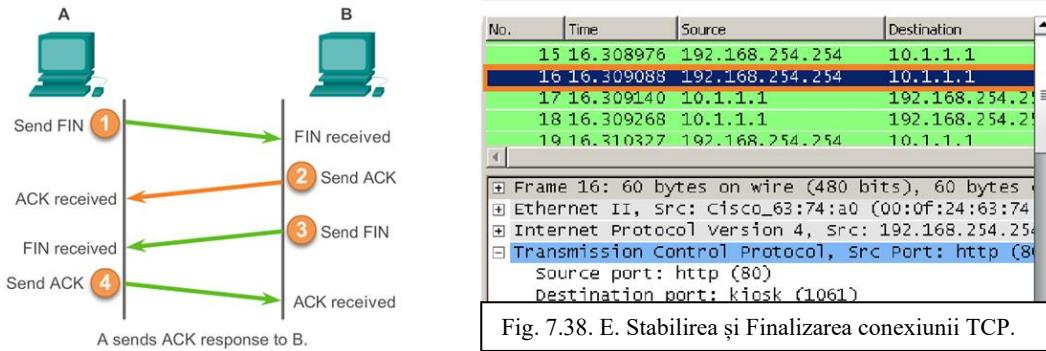


Fig. 7.38. E. Stabilirea și Finalizarea conexiunii TCP.

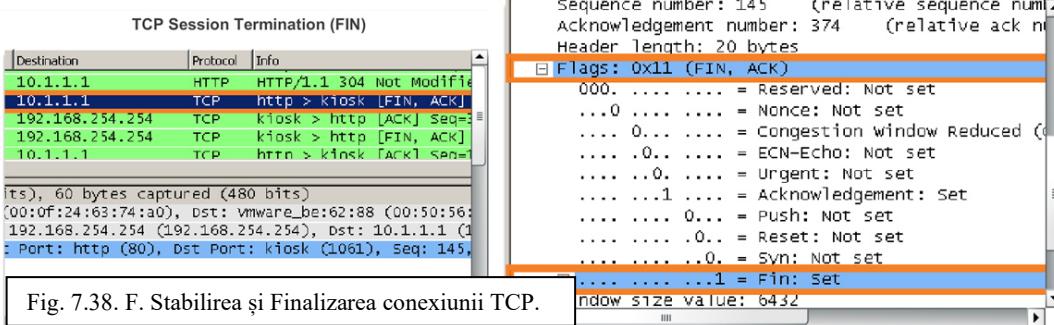
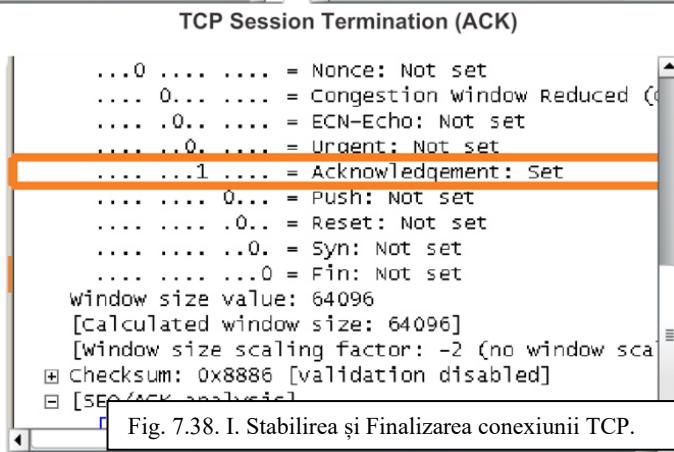
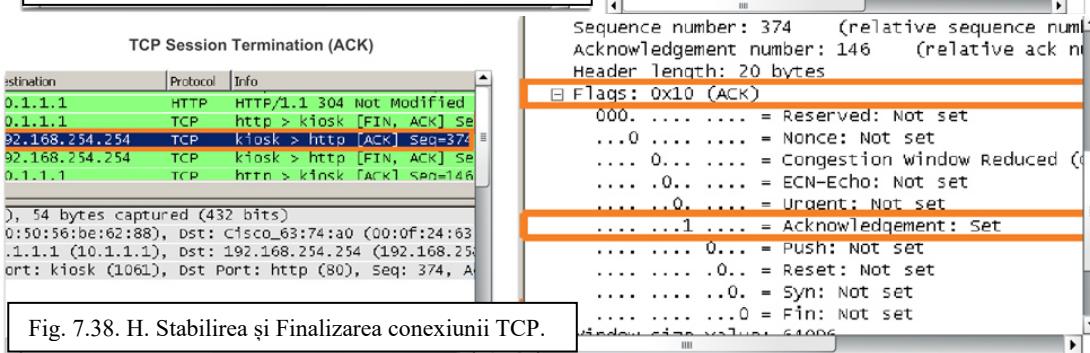
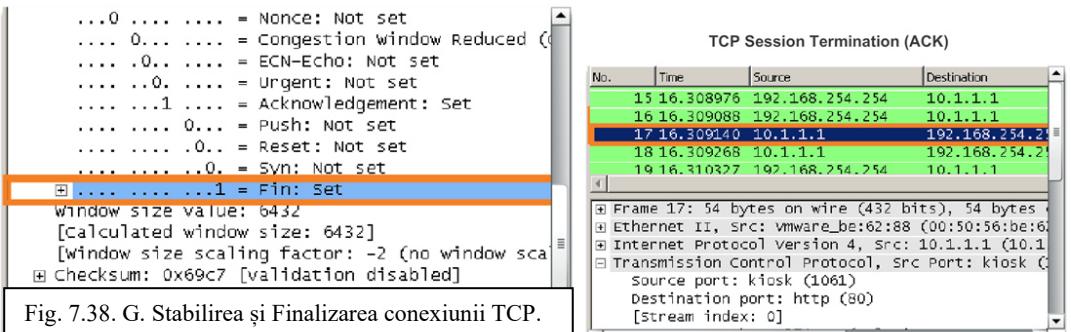


Fig. 7.38. F. Stabilirea și Finalizarea conexiunii TCP.



7.6 Încrederea și controlul fluxului

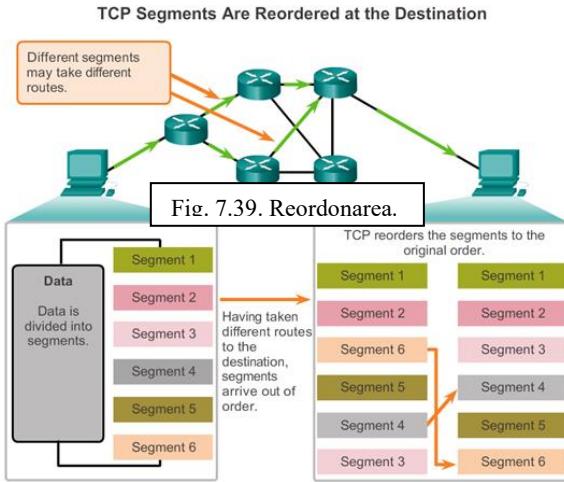
Atunci când serviciile trimit date folosind TCP, segmentele ar putea ajunge la destinație într-o altă ordine. Pentru ca mesajul original să fie înțeles de către destinatar, datele din segmente sunt reasamblate în ordinea originală. Numerele de secvență sunt atribuite în headerul fiecărui pachet pentru a indeplini acest lucru.

În timpul instalării sesiunii, un ISN este setat. Acest ISN reprezintă valoarea de start pentru bytes din această sesiune ce sunt transmiși la aplicația destinatară. În timp ce datele sunt transmise în timpul sesiunii, numărul de secvență crește cu unu împreună cu bytes transmiși. Această urmărire de byte de date permite fiecărui segment să fie identificat unic și confirmat. Segmentele lipsă pot fi identificate.

Numerele de secvență de segment oferă încredere prin indicarea modului în care segmentele primite să fie reasamblate și ordonate.

Procesul destinație TCP plasează datele de la un segment într-un buffer. Segmentele sunt plasate în ordinea adecvată a numărului de secvență și sunt trimise la nivelul aplicație atunci când sunt reasamblate. Oricare segmente care au ajuns cu un număr de secvență non-contiguous sunt

păstrate pentru procesare ulterioară. Apoi, atunci când segmentele cu bytes lipsă ajung, aceste segmente sunt procesate în ordine.



7.6.1 Confirmarea Recepționării Segmentelor

Una dintre funcțiile TCP este asigurarea faptului că fiecare segment ajunge la destinație. Serviciile TCP de pe hostul destinație confirmă datele ce au fost primite de la aplicația sursă.

SEQ și ACK sunt folosite împreună pentru a confirma primirea bytes de date conținuți în segmentele transmise. Numărul SEQ indică numărul relativ de bytes ce au fost transmisi în sesiune, inclusiv bytes din segmentul curent. TCP folosește numărul ACK trimis înapoi la sursă pentru a indica urmatorul byte pe care destinatarul îl așteaptă. Acesta se numește “*expectational acknowledgement*”.

Sursa este informată de faptul că destinația a primit toți bytes până la acest stream de date, dar nu include byteul indicat de către numărul ACK. Hostul expeditor ar trebui să trimită un segment ce folosește un număr de secvență egal cu numărul ACK.

Ne reamintim faptul că fiecare conexiune constă de fapt din două sesiuni într-o singură direcție. Numerele SEQ și ACK sunt schimbate în ambele direcții.

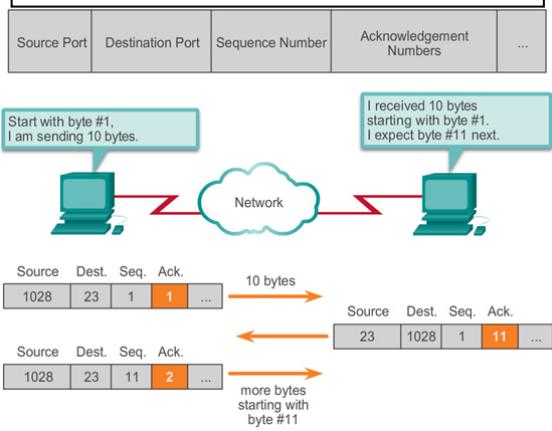
În exemplul din Fig. , hostul din stânga trimite date la hostul din dreapta. El trimite un segment ce conține 10 bytes de date pentru această sesiune și un SEQ egal cu 1 în header.

Hostul destinatar primește segmentul la Nivelul 4 și determină faptul că SEQ=1 și că are 10 bytes de date. Hostul apoi trimite un segment înapoi la hostul din stânga pentru a confirma primirea datelor. În acest segment, hostul setează ACK =11 pentru a indica faptul că următorul byte de date așteptat în această sesiune este byte cu numărul 11. Atunci când hostul expeditor primește acest ACK, poate trimite următorul segment ce conține date pentru această sesiune, începând cu byte numărul 11.

Luând în considerare acest exemplu, dacă hostul din stânga trebuie să aștepte ACK după fiecare 10 bytes, rețea va avea mult overhead. Pentru a reduce acest overhead al confirmărilor, mai multe segmente de date pot fi transmise și confirmate cu un singur mesaj TCP din direcția opusă. Acest ACK conține un număr ACK bazat pe numărul total de bytes primiți în sesiune. De exemplu, începând cu un număr de secvență 2000, dacă 10 segmente de 1.000 de bytes au fost primite, un număr ACK de 12001 va fi transmis sursei.

Cantitatea de date pe care o sursă o poate transmite înainte ca un ACK să fie primit se numește window size, ce este un câmp în headerul TCP ce permite gestionarea datelor pierdute și controlul fluxului.

Fig. 7.40. Acknowledgement of TCP Segments



7.6.2 Manevrarea Segmentelor Pierdute

Indiferent de cât de bine este dezvoltată rețeaua, pierderile de date au loc; prin urmare, TCP oferă metode de gestionare a pierderilor de segmente. Printre acestea există un mecanism de retrasmisare a segmentelor cu date neconfirmate.

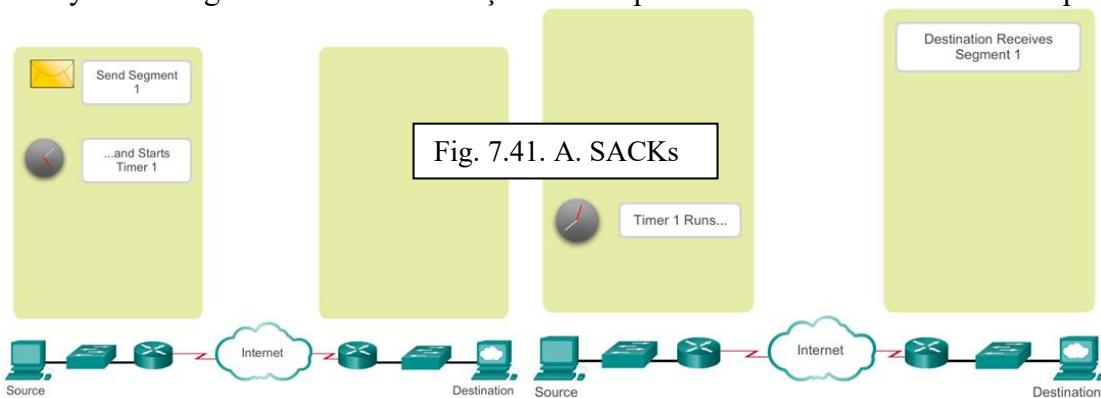
Un serviciu de pe hostul destinație ce folosește TCP confirmă numai datele pentru bytes din secvența conținută. Dacă unul sau mai multe segmente lipsesc, numai datele din prima secvență continuă de bytes sunt confirmate. De exemplu, dacă segmentele cu numerele de la 1500 la 3000 și 3400 la 3500 au fost primite, numărul ACK va fi 3001. Acest lucru se întâmplă datorită faptului că există segmente cu numere SEQ de la 3001 la 3399 ce nu au fost primite.

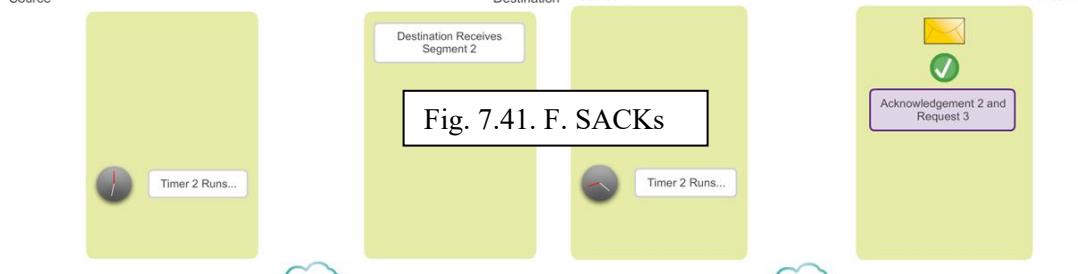
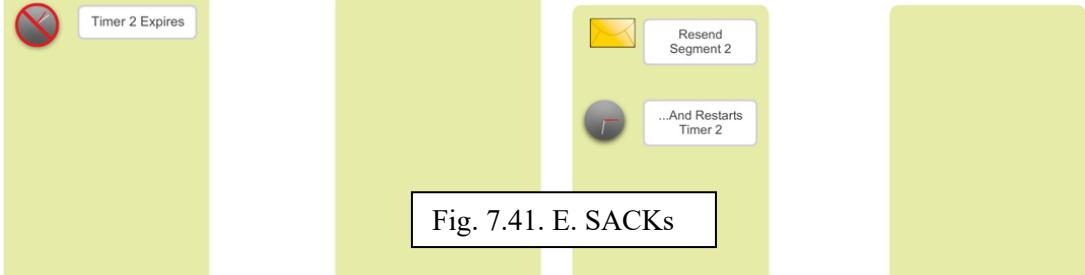
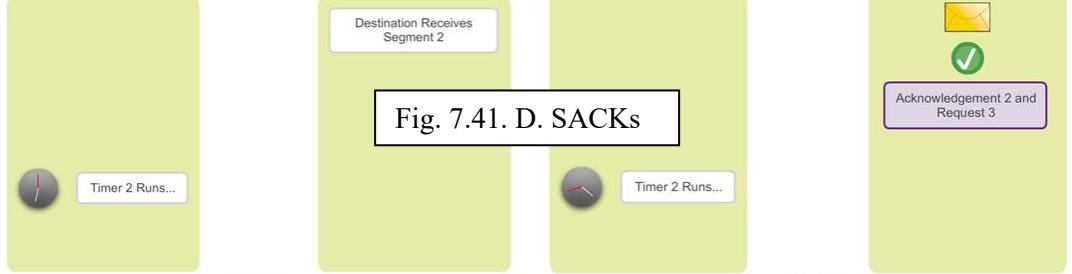
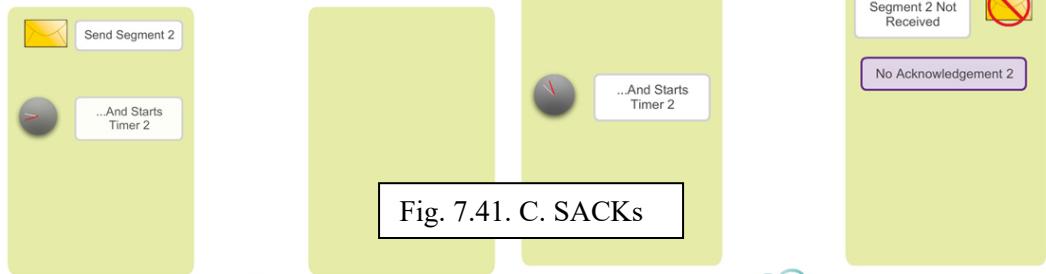
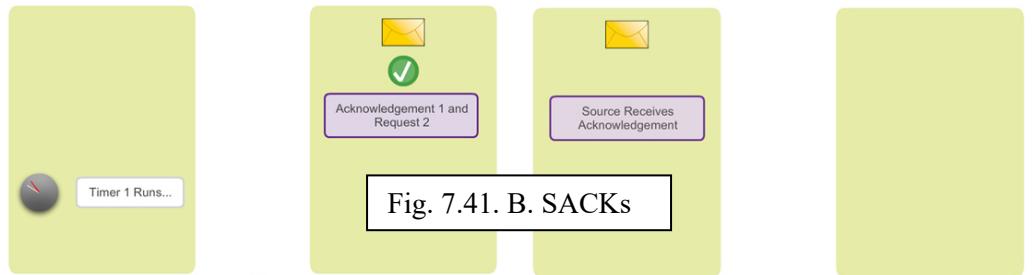
Atunci când TCP de pe hostul sursă nu a primit o confirmare după o perioadă de timp predefinită, se întoarce la ultimul număr ACK primit și retrasmite datele din acel punct. Procesul de retransfer nu este specificat de către Request for Comments (RFC), însă implementat de TCP.

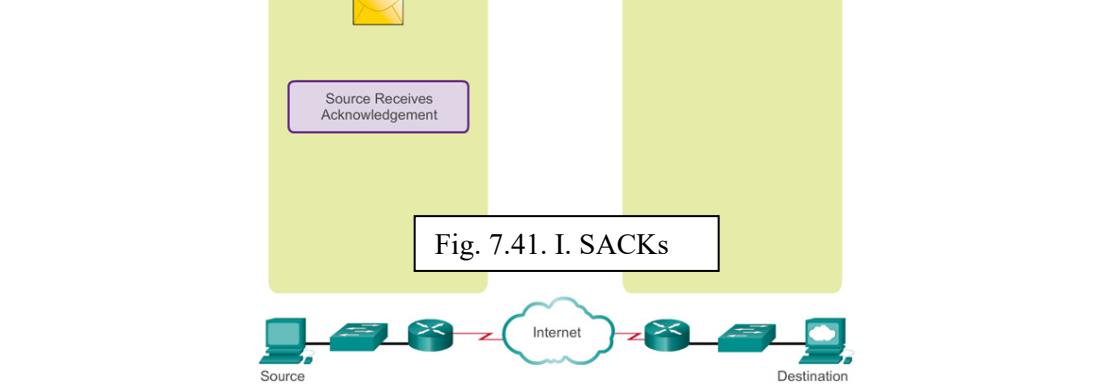
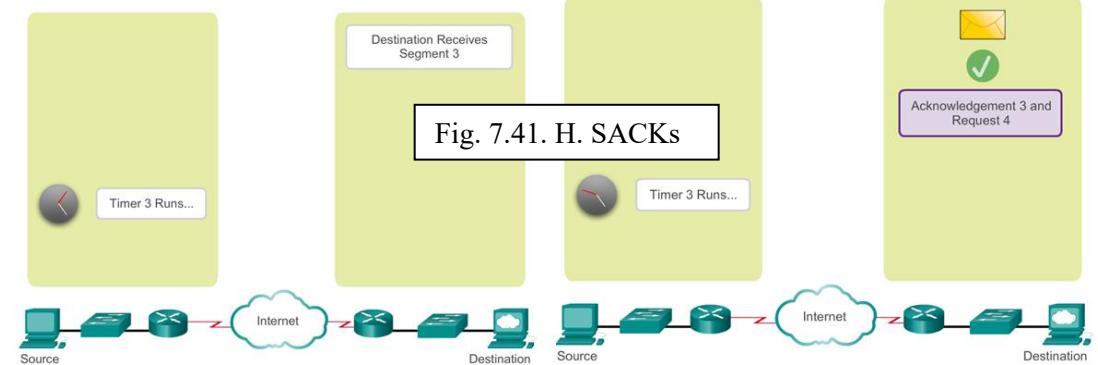
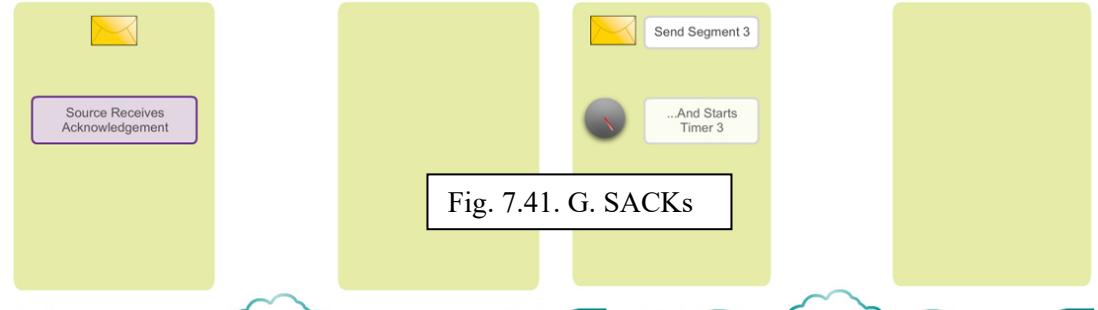
Pentru o implementare tipică TCP, un host ar putea transmite un segment, copiază segmentul într-o coadă de retrasmisare și pornește un cronometru. Atunci când confirmarea datelor este primită, segmentul este șters din coadă. Dacă nu se primește confirmarea până la expirarea timpului, segmentul este retrasmis.

Hosturile de astăzi ar putea de asemenea folosi o caracteristică optională numită confirmări selective (SACKs). Dacă ambele hosturi suportă SACKs, este posibil ca destinația să confirme bytes din segmentele discontinue și hostul ar putea să transmită numai datele lipsă.

Fig. 7.41. A. SACKs







7.6.3 Controlul fluxului

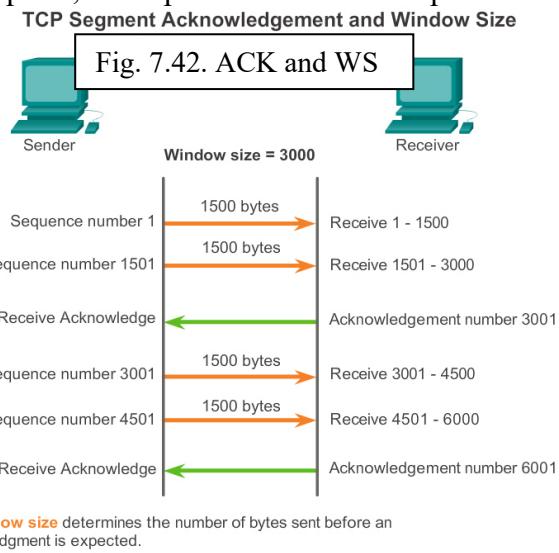
TCP oferă de asemenea mecanisme de control al fluxului. Controlul fluxului ajută la menținerea încrederii transmisiei TCP prin ajustarea cantității de flux de date dintre sursă și destinație pentru o sesiune dată. Controlul fluxului este realizat prin limitarea cantității de segmente de date transmise o dată și prin cererea de confirmări de primire înainte de a mai trimite altele.

Pentru a realiza controlul fluxului, primul lucru pe care TCP îl determină este cantitatea de segmente de date pe care dispozitivul destinație o poate accepta. Headerul TCP include un câmp de 16 biți numit windows size. Acesta este un număr de bytes pe care dispozitivul destinație al unei sesiuni TCP este capabil să îi accepte și să îi proceseze o dată. Window size inițial este convenit în timpul pornirii sesiunii prin three-way handshake între sursă și destinație. După convenție, dispozitivul sursă trebuie să limiteze contitatea de segmente de date transmise la destinație, în funcție de windows size. Numai după ce dispozitivul sursă a primit o confirmare a faptului că segmentele de date au fost primeite, poate continua trimiterea datelor pentru sesiune.

În timpul întârzierii în primirea confirmării, expeditorul nu mai trimită și alte segmente. În perioadele în care rețeaua este congestionață sau resursele de la hostul destinație sunt încărcate, întârzierea poate crește. Cu cât această întârziere crește mai mult, cu atât eficiența ratei de transmisie de date pentru această sesiune scade. Încetinirea în transmisia datelor pentru fiecare sesiune ajută la reducerea conflictelor de resurse din rețea și de la destinație atunci când mai multe sesiuni rulează.

Observăm Fig. pentru o reprezentare simplificată a window size și acknowledgements. În acest exemplu, window size inițial pentru o sesiune TCP este setată la 3000 bytes. Atunci când expeditorul a transmis 3000 de bytes, așteaptă o confirmare a acestor bytes înainte de a mai transmite segmente în sesiunea aceasta. După ce a primit confirmarea de la destinatar, poate transmite alți 3000 de bytes.

TCP folosește pentru a încerca să gestioneze rata de transmisie a fluxului maxim pe care rețeaua și destinația îl poate suporta, în timp ce se minimizează pierderea și retrasmisia.



7.6.4 Reducerea ferestrei de lucru - Window Size

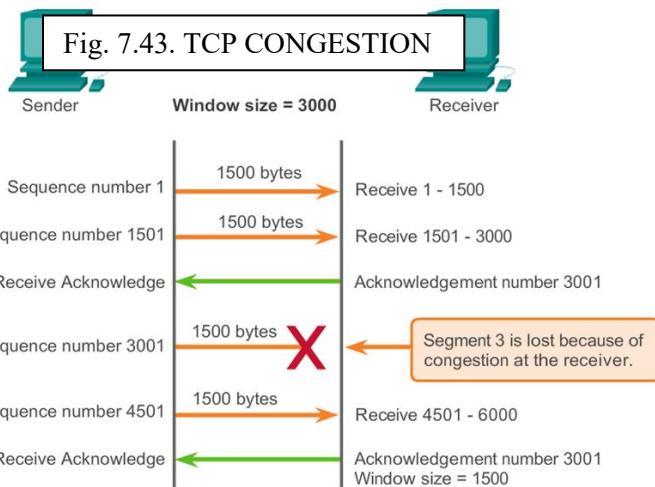
Un alt mod de control al fluxului de date este utilizarea **Window Size Dinamic**. Atunci când resursele de rețea sunt constrânse, TCP poate reduce window size prin cererea ca segmentele primite să fie confirmate mai des. Acest lucru reduce în mod eficient rata de transmisie deoarece sursa așteaptă ca datele să fie confirmate mai des.

Hostul destinație trimite valoarea window size hostului expeditor pentru a indica numărul de bytes pe care îi poate primi, astfel încât să negocieze window size cu sursa, dacă destinația necesită să încetinească rata de comunicare deoarece are o memorie de buffer limitată, de exemplu, poate trimite o valoare mai mică de window size sursei, ca parte din ack.

Așa cum este arătat și în Fig. , dacă hostul destinație este încărcat, poate răspunde sursei cu un segment în care specifică un window size redus. În această Fig. , există o pierdere a unei dintre segmente. Destinatarul a schimbat câmpul de windows din headerul TCP a segmentului din această conversație de la 3.000 la 1.500, ceea ce va determina ca sursa să reducă window size la 1.500.

După o perioadă de transmisie cu nici-o pierdere de date sau resurse încărcate, destinatarul începe să crească câmpul window, ceea ce va reduce overheadul din rețea datorită confirmărilor mai puține ce trebuie să fie transmise. Window size continuă să crească până când apare o pierdere de date, ce va determina scăderea window size.

Creșterea și scăderea dinamică a window size este un proces continuu în TCP. În rețelele foarte eficiente, window sizes pot deveni foarte mari deoarece datele nu sunt pierdute. În rețele în care infrastructura de bază este sub “stres”, window size rămâne cel mai probabil mică.



7.7 Comunicații UDP

UDP este un protocol simplu ce oferă funcții de bază de nivel transport. Are overhead mult mai scăzut decât TCP deoarece nu este orientat pe conexiune și nu oferă mecanisme sofisticate de retransmisie, secvențiere și control al fluxului ce oferă încredere.

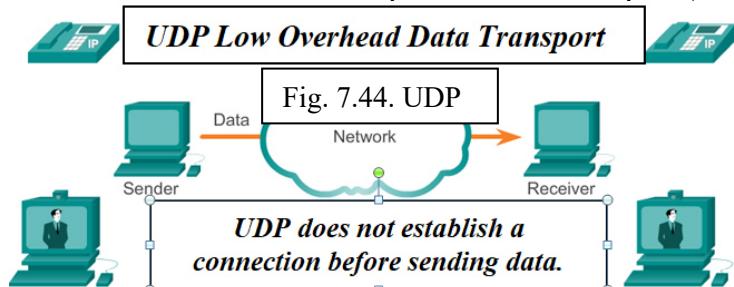
Acest lucru nu înseamnă că aplicațiile ce folosesc UDP sunt întotdeauna de neîncredere sau că UDP este un protocol inferior. Înseamnă numai că aceste funcții nu sunt oferite de către protocolul de la nivelul transport și trebuie să fie implementate în altă parte, dacă sunt necesare.

Deși cantitatea totală de trafic UDP dintr-o rețea normal este relativ mică, protocolele cheie de la nivelul aplicație ce folosesc UDP sunt:

- Domain Name System (DNS).
- Simple Network Management Protocol (SNMP).
- Dynamic Host Configuration Protocol (DHCP).
- Routing Information Protocol (RIP).
- Trivial File Transfer Protocol (TFTP).
- IP telephony or Voice over IP (VoIP).
- Online games.

Unele aplicații, cum ar fi online games sau VoIP, pot tolera unele pierderi de date. Dacă aceste aplicații folosesc TCP, pot întâlni întârzieri mari în timpul detecție TCP a pierderii de date și retransmisia lor. Întârzierile ar fi mai dăunătoare performanței aplicației decât unele pierderi mici de date. Unele aplicații, cum ar fi DNS, ar putea trimite cererea din nou dacă nu primesc un răspuns; prin urmare, nu necesită TCP pentru garantarea livrării mesajului.

Overheadul scăzut al UDP îl face de dorit pentru asemenea aplicații.

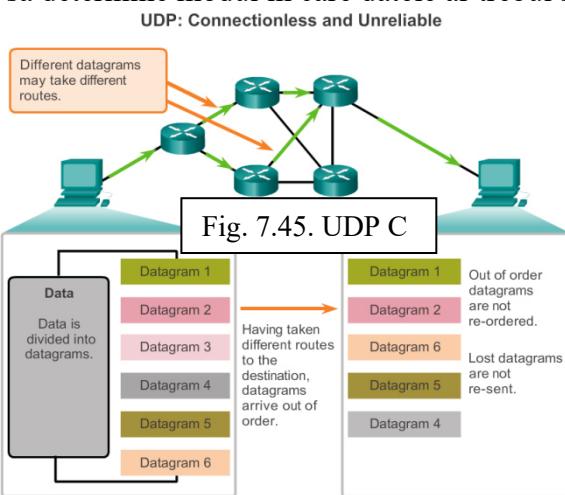


Deoarece UDP este connectionless, sesiunile nu sunt stabilite înainte de a avea loc comunicația, spre deosebire de TCP. UDP se spune că este pe bază de tranzacție; atunci când o aplicație are date de transmis, pur și simplu le transmite.

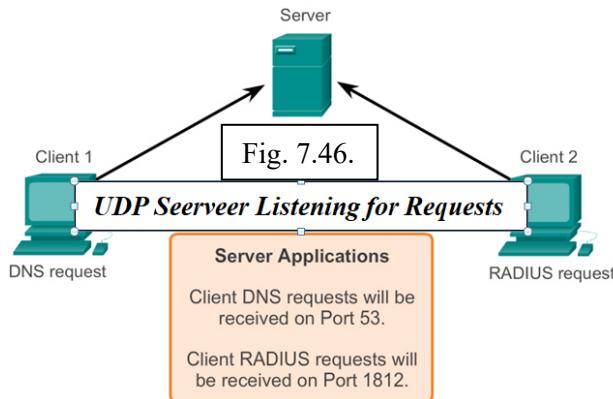
Mai multe aplicații ce folosesc UDP trimit cantități mici de date care încap într-un singur segment. Însă, unele aplicații trimit cantități mari de date ce trebuie să fie împărțite în mai multe segmente. PDU-ul UDP se numește datagram, deși termenii de segment și datagram sunt uneori folosiți alternativ pentru a descrie un PDU de nivel transport.

Atunci când mai multe datagrams sunt transmise la destinație, pot urma căi diferite și pot ajunge într-o ordine greșită. UDP nu urmărește numere de secvență, spre deosebire de TCP. UDP nu are nici-o modalitate de reordonare a datagrams în ordinea în care au fost transmise, așa cum este evidențiat și în Fig. .

Prin urmare, UDP reasamblează pur și simplu datele în ordinea în care au fost primite și le transmite la aplicație. Dacă secvența de date este importantă pentru aplicație, aplicația trebuie să identifice secvența bună și să determine modul în care datele ar trebui să fie procesate.



Ca și aplicațiile bazate pe TCP, aplicațiile server bazate pe UDP au atribuite numere de port bine-cunoscute sau înregistrate. Atunci când aceste aplicații sau procese rulează pe un server, acceptă datele ce corespund cu numărul de port atribuit. Atunci când UDP primește un datagram destinat pentru unul dintre aceste porturi, trime datele de aplicație la aplicația adecvată, în funcție de numărul de port.

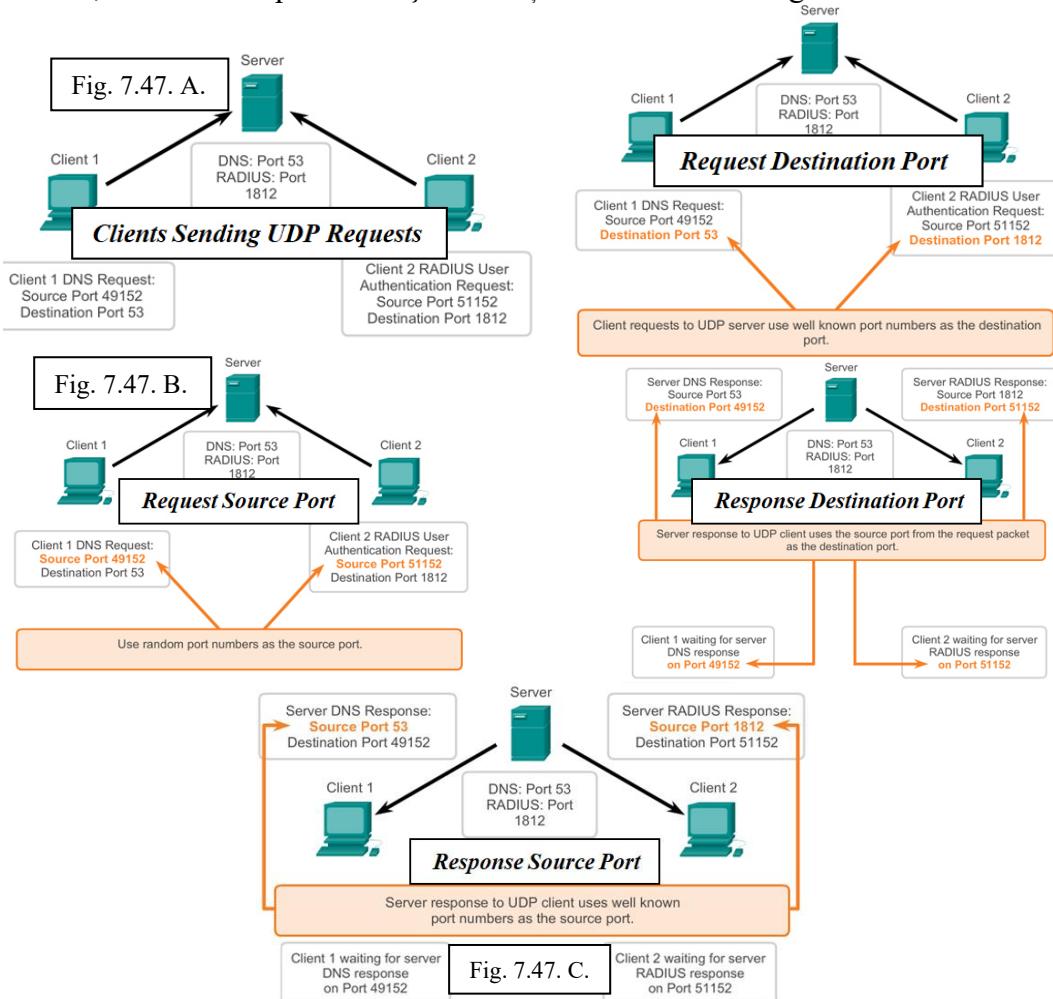


La fel ca TCP, comunicația client/server este inițiată de către aplicația client care cere date de la un proces server. Procesul client UDP selectează aleator un număr de port din intervalul numerelor de port dinamice și îl folosește ca port sursă al conversației. Portul destinație este de obicei un număr de port bine-cunoscut sau înregistrat atribuit procesului server.

Numerele aleatoare de port sursă ajută de asemenea securitatea. Dacă există un structură previzibilă pentru selecția portului destinație, un intrus poate cu usurință simula accesul la un client prin încercarea conectării la numărul de port ce este cel mai probabil deschis.

Deoarece nu există sesiune creată de UDP, de îndată ce datele sunt gata de transmis și porturile sunt identificate, UDP poate forma datagrams și le poate transmite la nivelul rețea pentru a fi adresate și trimise în rețea.

După ce un client a selectat porturile sursă și destinație, aceeași pereche de porturi este folosită în headerul tuturor datagrams folosite în tranzacție. Pentru ca datele să se întoarcă de la server la client, numerele de port sursă și destinație din headerul datagram sunt inversate.



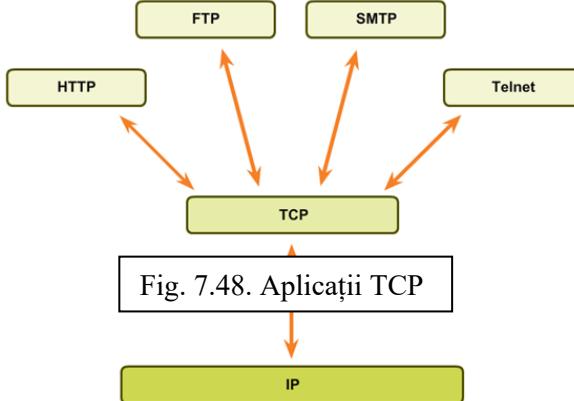
7.8 TCP sau UDP, aceasta este întrebarea

Multe aplicații necesită încredere și alte servicii oferite de TCP. Acestea sunt aplicațiile ce pot tolera unele întârzieri sau pierdere de performanță, având în vedere overheadul TCP.

Acest lucru face TCP cel mai potrivit pentru aplicațiile care necesită transport de încredere și care pot tolera unele întârzieri. TCP este un exemplu excepțional al modului în care nivele diferite ale suitei de protocol TCP/IP au roluri specifice. Deoarece protocolul de transport TCP se ocupă de toate sarcinile asociate cu segmentarea de date în segmente, încredere, control al fluxului și reordonare a segmentelor, eliberează nivelul aplicație de toate acestea. Aplicația poate pur și simplu trimite datele la nivelul transport și să folosească servicii TCP.

Aşa cum este prezentat şi în Fig. de mai jos, unele exemple de aplicaţii bine-cunoscute ce folosesc TCP sunt:

- *Hypertext Transfer Protocol (HTTP)*.
- *File Transfer Protocol (FTP)*.
- *Simple Mail Transfer Protocol (SMTP)*.
- *Telnet*.



Exista trei tipuri de aplicaţii ce se pretează cel mai bine pentru UDP:

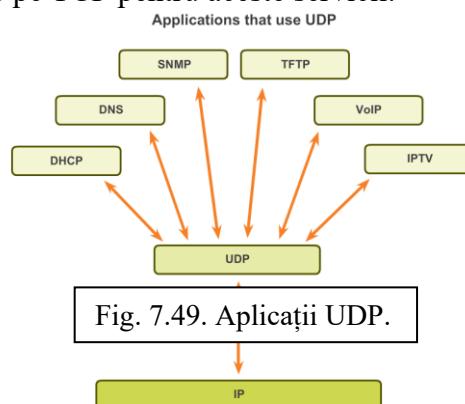
- *Aplicaţiile ce pot tolera unele pierderi de date, însă necesită întârziere scăzută sau deloc.*
- *Aplicaţiile cu tranzacţii simple de cerere şi răspuns.*
- *Comunicaţii unidirecţionale în care încrederea nu este necesară sau poate fi gestionată de către aplicaţie.*

Multe aplicaţii video şi multimedia, cum ar fi VoIP şi Internet Protocol Television (IPTV), folosesc UDP. Aceste aplicaţii pot tolera unele pierderi de date cu efect scăzut sau nedetectabil. Mecanismele de încredere ale TCP introduc unele întârzieri ce pot fi semnificative în calitatea video sau audio primită.

Alte tipuri de aplicaţii potrivite pentru UDP sunt acelea care folosesc tranzacţii simple cerere/răspuns. Acest lucru se întâmplă când un host trimite o cerere şi ar putea sau nu să primească un răspuns. Aceste tipuri de aplicaţii sunt:

- *DHCP*.
- *DNS - May also use TCP*.
- *SNMP*.
- *TFTP*.

Unele aplicaţii se ocupă de încredere. Aceste aplicaţii nu necesită servicii TCP şi folosesc mai bine UDP ca protocol de transport. TFTP este un exemplu de acest tip de protocol. TFTP are propriile mecanisme de control al fluxului, detecţia erorii, confirmării şi recuperare în caz de eroare. Nu trebuie să se bazeze pe TCP pentru aceste servicii.



7.9 Concluzii Capitolul 7

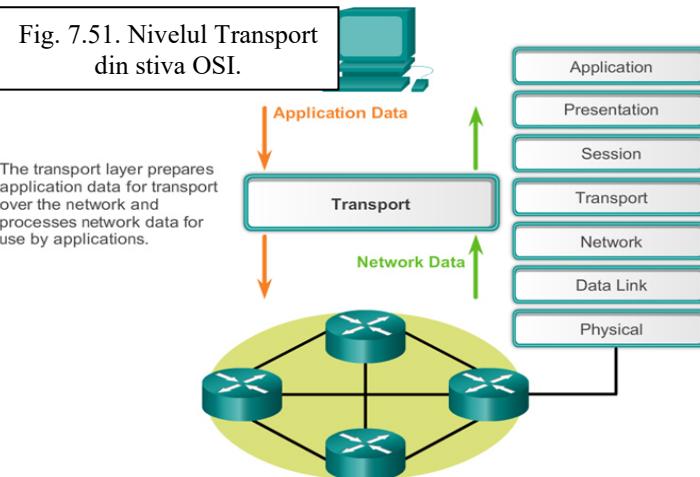


Fig. 7.50.

TCP and UDP are transport layer protocols instrumental in ensuring that...

- Network communications with different levels of importance are sent/received according to their levels of importance.
- The type of data will affect whether TCP or UDP will be used as the method of delivery.
- Timing is a factor and will affect how long it takes to send/receive TCP/UDP data transmissions.

În timpul „călătoriei” datelor prin rețea, acestea sunt împărțite în piese mai mici și identificate în aşa fel încât să poată fi reasamblate. Fiecare dintre aceste piese are atribuit un nume specific (PDU) și se asociază cu un anumit nivel. Modul de simulare Packet Tracer permite utilizatorului să vadă fiecare dintre protocoale și PDU-ul asociat. Pașii descriși mai jos conduc utilizatorul prin procesul de cerere de servicii folosind mai multe aplicații disponibile pe un client PC.



Nivelul transport oferă servicii legate de transport prin:

- Divizarea datelor primite de la o aplicație în segmente.
- Adaugarea unui header pentru a identifica și gestiona fiecare segment.
- Utilizarea informațiilor din header pentru reasamblarea segmentelor în datele de aplicatie.
- Transmiterea datelor reasamblate la aplicația corectă.

UDP datagrams și TCP segments au headere adăugate în partea din față a datelor ce includ un număr de port destinație și un număr de port sursă. Aceste numere de port permit ca datele să fie direcționate la aplicația corectă ce rulează pe computerul destinație.

TCP nu transmite date în rețea până când nu cunoaște faptul că destinația este pregătită să le primească. TCP apoi gestionează fluxul de date și retransmite orice segmente de date care nu au fost confirmate. TCP folosește mecanisme de handshaking, cronometrare, mesaje de confirmare și windowing dinamic pentru a îndeplini funcția de încredere. Acest proces de

încredere, însă, impune overhead pe rețea în termeni de headere de segment mult mai mari și trafic mai mare între sursă și destinație.

Dacă datele de aplicație necesită să fie livrate prin rețea rapid, sau dacă lățimea de bandă a rețelei nu suportă overheadul mesajelor de control ce sunt schimbată între sursă și destinație, UDP va fi protocolul de nivel transport preferat de către dezvoltator. UDP nu urmărește sau nu confirmă primirea datagramelor la destinație – doar transferă datagrams primite la nivelul aplicație, în forma în care au fost primite – și nu retrimit datagrams pierdute. Însă, acest lucru nu înseamnă neapărat că nu este de încredere comunicarea; ar putea exista mecanisme în protocoalele de la nivelul aplicație și servicii ce procesează datagrams pierdute sau întârziate, dacă aplicația are aceste cerințe.

Dezvoltatorul de aplicație decide protocolul de nivel transport cel mai potrivit cerințelor aplicației. Este important de ținut minte faptul că toate nivelele joacă un rol în comunicațiile din rețeaua de date și influențează performanța rețelei.

CAPITOLUL 8. IP ADDRESSING

Introducere

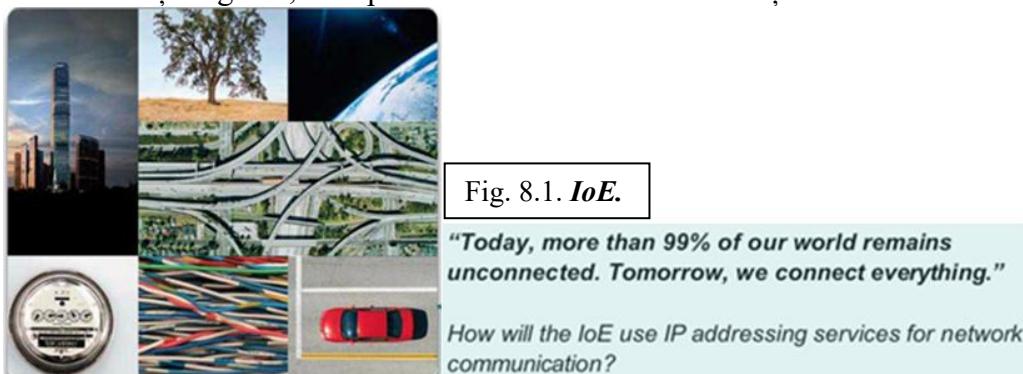
Adresarea este o funcție cheie a protocolelor de nivel rețea ce permite comunicarea datelor între hosturi, indiferent dacă sunt în aceeași rețea sau în rețele diferite. Protocolele Internet Protocol version 4 (IPv4) și Internet Protocol version 6 (IPv6) oferă adresare ierarhică pentru pachetele ce conțin date.

Proiectarea, implementarea și gestionarea unui plan eficient de adresare IP permite ca rețelele să funcționeze eficient și productiv.

Acet capitol examinează în detaliu structura adreselor IP și aplicațiile lor în construcția și testarea rețelelor și subrețelelor IP.

Internetul pentru toți (The Internet of Everything – IoE)

Dacă natura, traficul, transportul, crearea de rețele și explorarea spațiului depind de schimbul de informații digitale, cum pot fi identificate acele informații de la sursă la destinație ?



8.1 Adresele de Rețea IPv4 – Structura Adreselor IPv4

Pentru a înțelege funcționarea dispozitivelor dintr-o rețea, trebuie să ne uităm la adresele și celealte date în modul în care dispozitivele o fac – în notație binară. Notația binară este o reprezentare a informațiilor folosind zero și unu. Computerele comunică prin date binare. Datele binare pot fi folosite pentru a reprezenta mai multe forme de date. De exemplu, atunci când scriem scrisori cu tastatura, literele apar pe ecran într-o formă pe care o putem înțelege și citi; însă, computerul traduce fiecare literă într-o serie binară pentru stocare și transport. Pentru a traduce acele litere, computerul folosește American Standard Code for Information Interchange (ASCII).

Folosind ASCII, litera "A" este reprezentată în binar ca 01000001, pe când litera "a" este 01100001.

Nu este în general necesar ca oamenii să se preocupe de conversia binară a literelor, dar este necesară înțelegerea utilizării notației binare pentru adresarea IP. Fiecare dispozitiv dintr-o rețea trebuie identificat unic, folosind o adresă binară. În rețelele IPv4, această adresă este reprezentată folosind un string de 32 de biți (0 și 1). La nivelul rețea, pachetele includ această informație unică de identificare atât pentru sistemul sursă, cât și pentru destinație. Prin urmare, într-o rețea IPv4, fiecare pachet include o adresă sursă pe 32 de biți și o adresă destinație de 32 de biți în headerul de nivel 3.

Pentru mulți oameni, un string de 32 de biți este dificil de interpretat și chiar mai dificil de ținut minte. Prin urmare, reprezentăm adresele IPv4 folosind formatul zecimal în schimbul celui binar. Acest lucru înseamnă că ne uităm la fiecare octet ca la un număr zecimal de la 0 la 255. Pentru a înțelege acest lucru trebuie să avem anumite aptitudini în conversia binar-zecimal.

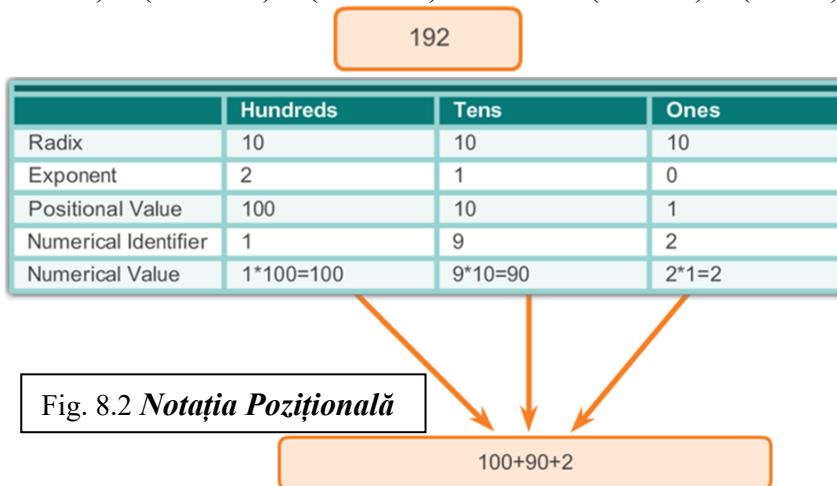
8.1.1 Notația Pozițională

Învățarea conversiei binar-zecimală necesită o înțelegere a bazei sistemului de numerație matematic numit *"positional notation"*. Notarea pozițională înseamnă că un digit reprezintă diferite valori în funcție de poziția pe care o ocupă. Într-un sistem de notație pozițională, baza numărului se numește radix. În sistemul de bază 10, radix este 10. În sistemul binar, radix este 2. Termenul de radix și bază pot fi folosiți în mod alternativ. Mai exact, valoarea pe care o reprezintă un digit este valoarea multiplicată de puterea bazei, sau radix, reprezentată de poziția pe care o ocupă digitul. Unele exemple vor ajuta în clarificarea modului în care acest sistem funcționează.

Pentru numărul zecimal 192, valoarea reprezentată de 1 este $1 * 10^2$. 1 este ceea cea la care ne referim poziția sutelor. Notația pozițională se referă la această poziție ca poziție în $baza^2$ deoarece baza, sau radix, este 10 și puterea este 2. 9 reprezintă $9 * 10^1$.

Folosind notația pozițională în sistemul numeric în baza 10, 192 este:

$$192 = (1 * 10^2) + (9 * 10^1) + (2 * 10^0) \text{ sau } 192 = (1 * 100) + (9 * 10) + (2 * 1)$$



În IPv4, adresele sunt numere binare de 32 de biți. Însă, pentru o mai ușoară utilizare de către oameni, tipurile binare ce reprezintă adresele IPv4 sunt exprimate în zecimal. Acest lucru este mai întâi realizat prin separarea fiecărui octet (8 biți) al tiparului binar de 32 de biți printr-un punct. Aceasta se numește octet deoarece fiecare număr zecimal reprezintă un byte sau 8 biți.

Adresa binară 11000000 10101000 00001010 00001010 este reprezentată în zecimal de numărul = 192.168.10.10.

Dar cum sunt determinate echivalentele zecimale ?

8.2 Sistemul de numerație binar

În sistemul de numerotare binar, radix este 2. Prin urmare, fiecare poziție reprezintă puterea în creștere a lui 2. În numerele binare pe 8 biți, pozițiile reprezintă următoarele cantități:

$$2^7 = 128, 2^6 = 64, 2^5 = 32, 2^4 = 16, 2^3 = 8, 2^2 = 4, 2^1 = 2, 2^0 = 1.$$

Sistemul de numerotare binar are numai două cifre: 0 și 1.

Atunci când interpretăm un byte ca un număr zecimal, avem cantitatea pe care poziția o reprezintă dacă cifra este 1 și nu avem cantitatea dacă cifra este 0, aşa cum este arătat și în Fig.1.

Fig. 8.2 ilustrează reprezentarea numărului 192 în binar. Un 1 într-o anumită poziție înseamnă că adăugăm valoarea respectivă la total. Un 0 înseamnă că nu adăugăm respectiva valoare. Numărul binar 11000000 are un 1 în poziția 2^7 (valoarea zecimală 128) și un 1 în poziția 2^6 (valoarea zecimală 64). Biții rămași sunt toți 0, deci nu mai adăugăm valorile zecimale respective. Rezultatul $128+64$ este 192, valoarea zecimală pentru numărul binar 11000000.

Exemplul 1: Un octet ce conține toți de 1: 11111111

Un 1 în fiecare poziție înseamnă că adăugăm valoarea pentru fiecare poziție la total. Toți de 1 înseamnă că valorile fiecărei poziții sunt incluse în total, deci, valoarea tuturor de 1 dintr-un octet este 255.

$$128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255$$

Exemplul 2. Un octet ce conține toți 0: 00000000

Un 0 în fiecare poziție înseamnă că nu adăugăm valoarea pentru fiecare poziție la total. Un 0 în fiecare poziție conduce la un total de 0.

$$0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 = 0$$

O combinație diferită de 0 și 1 va conduce la o valoare zecimală diferită.

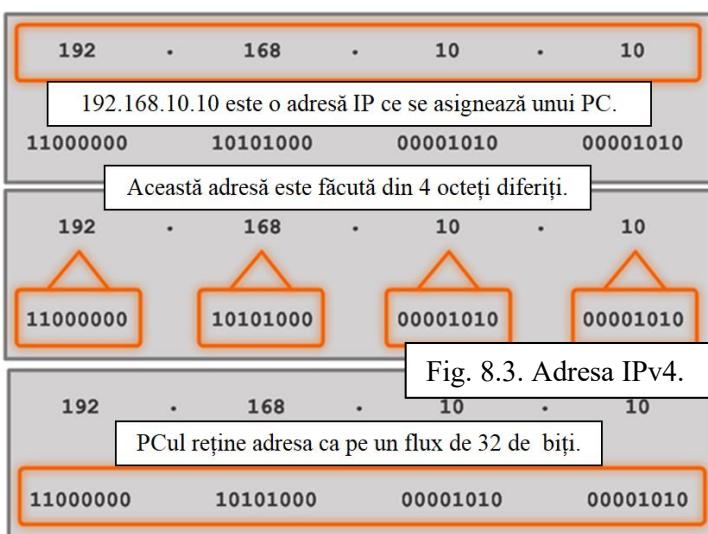


Fig. 8.3. Adresa IPv4.

Radix	2	2	2	2	2	2	2	2
Exponent	7	6	5	4	3	2	1	0
Octet Bit Values	128	64	32	16	8	4	2	1
Binary Address	1	1	0	0	0	0	0	0
Binary Bit Values	128	64	0	0	0	0	0	0

Add the binary bit values.
128 + 64 = 192

Legend:
■ - 1 in this position means add the octet bit value to the total
■ - 0 in this position means 0 is added to the total

Fig. 8.4. Calculul Binar.

Fiecare octet este alcătuit din 8 biți și fiecare bit are o valoare, fie 0, fie 1. Cele patru grupuri de 8 biți au același set de valori valide în intervalul de la 0 la 255 inclusiv. Valoarea fiecărui bit, de la dreapta la stânga este 1, 2, 4, 8, 16, 32, 64 și 128.

Determinarea valoarii octetului prin adăugarea valorilor pozițiilor în cazul în care este un 1 binar.

- Dacă este un 0 într-o poziție, nu adăugăm valoarea.
- Dacă toți biții sunt 0, 00000000, valoarea octetului este 0.
- Dacă toți biții sunt 1, 11111111, valoarea octetului este $255(128+64+32+16+8+4+2+1)$.
- Dacă cei 8 biți sunt de valori diferite, valorile sunt adăugate. De exemplu, octetul 00100111 are valoarea 39 ($32+4+2+1$).

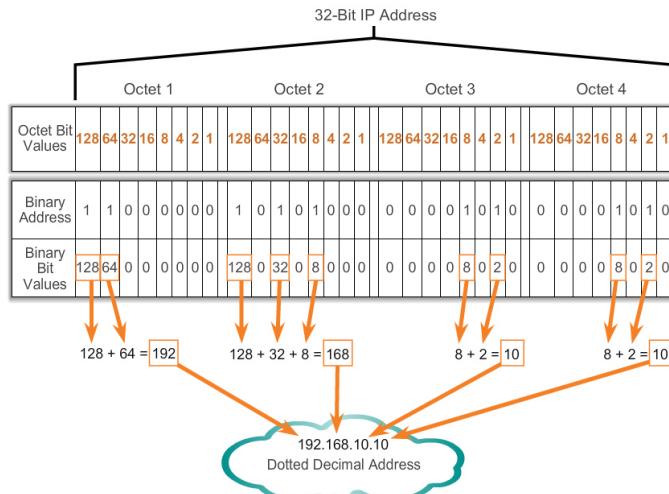
Deci valoarea fiecărui octet dintre cei 4 poate varia de la 0 la un maxim de 255.

Folosind adresa IPv4 pe 32 de biți, 1100.0000/1010.1000/0000.1010/0000.1010, convertim reprezentarea binară la formatul zecimal punctat folosind următorii pași:

Pasul 1. Divizăm cei 32 de biți în 4 octeți.

Pasul 2. Convertim fiecare octet în zecimal.

Pasul 3. Adăugăm un “.” între fiecare valoare zecimală.

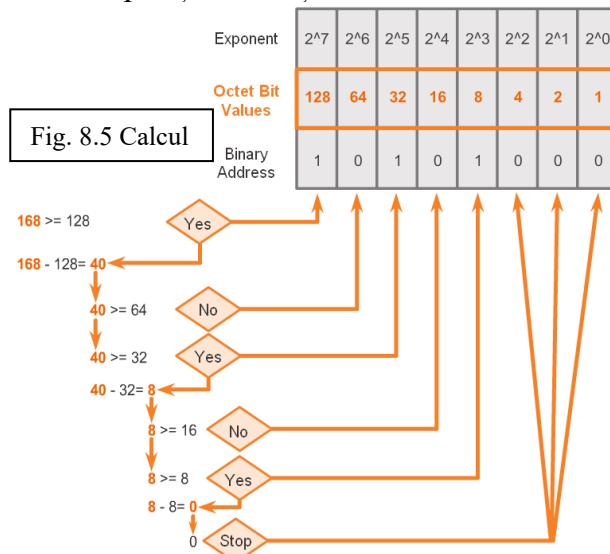


Pentru a fi capabil să convertim binar în zecimal este de asemenea necesară înțelegerea modului de conversie zecimal în binar.

Deoarece reprezentăm adresele IPv4 folosind formatul zecimal punctat, este necesară examinarea procesului de conversie a 8 biți din binar în valorile zecimale de la 0 la 255 pentru fiecare octet dintr-o adresă IPv4.

Pentru a începe procesul de conversie, începem prin determinarea numărului zecimal care este egal sau mai mare decât cea mai mare valoare zecimală reprezentată de cel mai semnificativ bit. În cea mai mare poziție, determinăm dacă numărul este egal sau mai mare de 128. Dacă numărul este mai mic decât 128, punem 0 în poziția respectivă pentru valoarea zecimală 128 și trecem la poziția bitului pentru valoarea zecimală 64.

Dacă numărul din poziția bitului pentru valoarea zecimală 128 este mai mare sau egal cu 128, adăugăm 1 în poziția bitului pentru valoarea zecimală 128 și scădem 128 din numărul ce este convertit. Apoi convertim rezultatul operației de mai sus cu următoarea valoare mai mică, 64. Continuăm procesul pentru toate pozițiile de biți rămasă.



Urmărim pașii de conversie din imagini pentru a vedea cum o adresă IPv4 este convertită în binar.

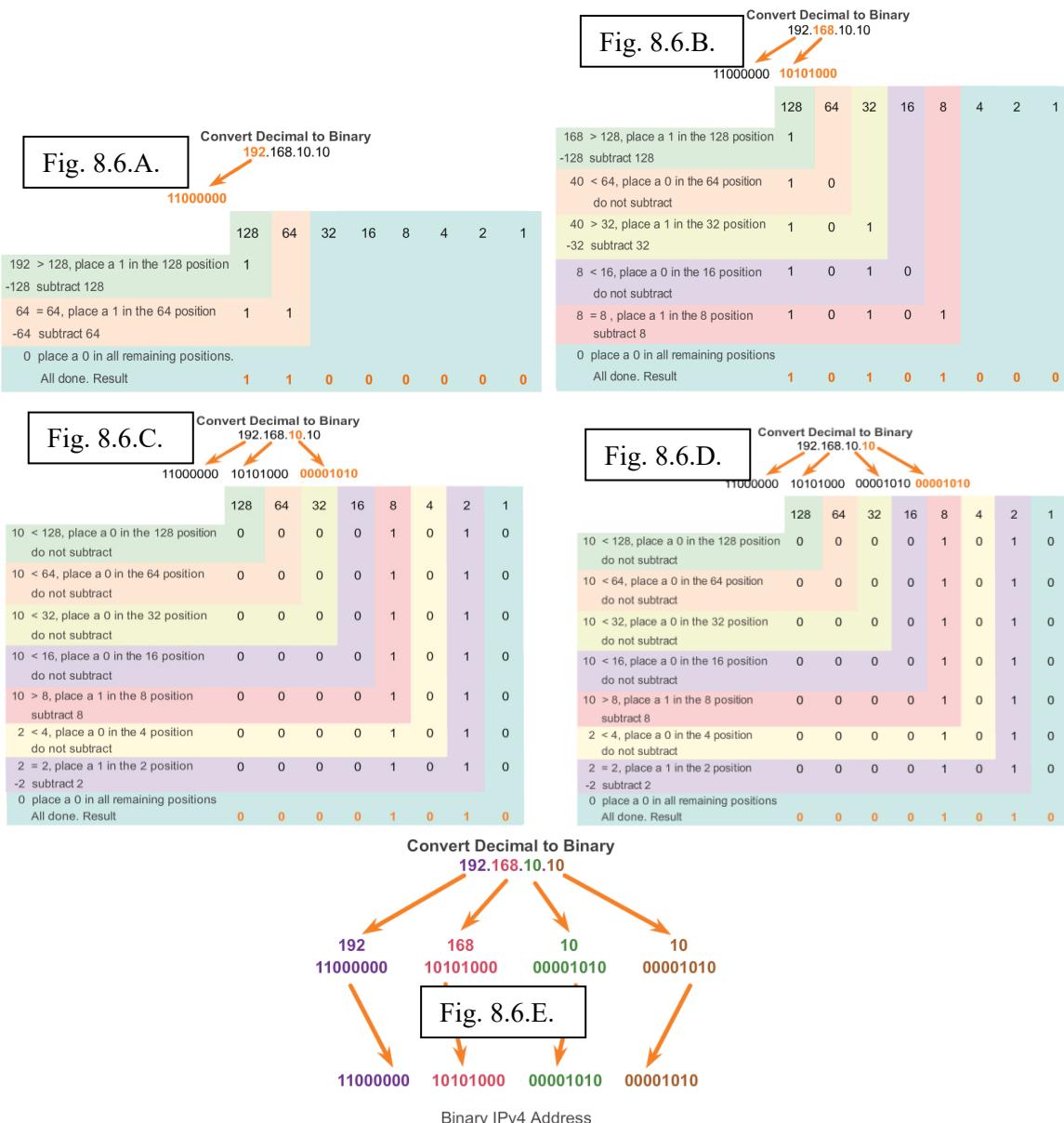
Fig. 8.6.A. Convertim 192 în binar.

Fig. 8.6.B. Convertim 168 în binar.

Fig. 8.6.C. Convertim 10 în binar.

Fig. 8.6.D. Convertim 10 în binar.

Fig. 8.6.E. Combinăm octetii convertiti începând cu primul octet.



8.4 Masca de Rețea pentru adresa de tip : IPv4

Înțelegerea notației binare este importantă atunci când determinăm dacă două hosturi se află în aceeași rețea. Reamintim faptul că o adresă IP este o adresă ierarhică alcătuită din două părți: o parte de rețea și o parte de host. Atunci când determinăm partea de rețea vs partea de host, este necesar să ne uităm la streamul de 32 de biți și nu la valoarea zecimală. În streamul de 32 de biți, o parte de biți alcătuiesc rețea și o parte din biți alcătuiesc hostul.

Biții dintr-o parte de rețea a unei adrese trebuie să fie identici pentru toate dispozitivele din aceeași rețea. Biții din partea de host a unei adrese trebuie să fie unici pentru a identifica un anumit host din rețea. Indiferent dacă numerele zecimale ale două adrese IPv4 se potrivesc, dacă

două hosturi au același bit-pattern în partea de rețea specificată a streamului de 32 de biți, acele două hosturi sunt în aceeași rețea.

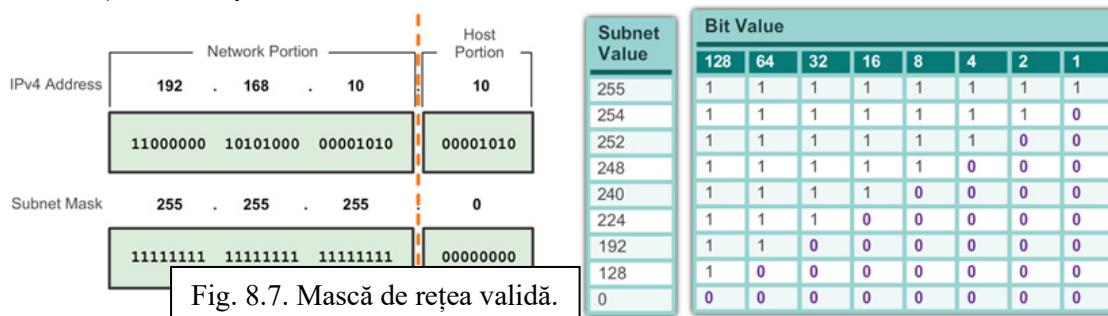
Întrebare : Cum știu hosturile care parte a 32 de biți este de rețea și care de host ?

Răspuns : Aceasta este sarcina măștii de rețea.

Atunci când un host IP este config. t, o mască de rețea este atribuită împreună cu adresa IP. Ca și adresa IP, masca de rețea este de 32 de biți. Masca de rețea indică care parte a adresei IP este rețea și care parte este hostul.

Masca de rețea este comparată cu adresa IP de la stânga la dreapta, bit cu bit. 1 din masca de rețea reprezintă partea de rețea; 0 reprezintă partea de host. Ca și în Fig. 1, masca de rețea este creată prin plasarea unui 1 binar în fiecare poziție de bit ce reprezintă partea de rețea și plasarea fiecărui 0 binar în fiecare poziție de bit ce reprezintă partea de host. De reținut faptul că masca de rețea nu conține de fapt partea de rețea sau host a unei adrese IP, ci doar specifică computerului unde să se uite pentru acele părți într-o adresă IPv4 dată.

Similar adreselor IPv4, masca de rețea este reprezentată în format zecimal punctat pentru o utilizare mai ușoară. Masca de rețea este config. tă pe un dispozitiv host, în concordanță cu adresa IPv4, și este necesară pentru ca hostul să determine din care rețea face parte. Fig. 2 arată măștile de rețea valide pentru un octet IPv4.



8.5 Prefixele de rețea

Lungimea unui prefix este un alt mod de exprimare a măștii de rețea. Lungimea unui prefix este numărul de biți de 1 din masca de rețea. Este scrisă în notația comprimată cu ajutorul simbolului slash, un “/” urmat de numărul de biți de 1. De exemplu, dacă masca de rețea este 255.255.255.0, există 24 de biți de 1 în varianta binară a măștii de rețea, deci lungimea prefixului este de 24 de biți sau /24. Prefixul și masca de rețea sunt modalități diferite de reprezentare a aceluiași lucru – partea de rețea a unei adrese.

Rețelele nu au întotdeauna atribuit un prefix /24. În funcție de numărul de hosturi din rețea, prefixul atribuit poate fi diferit. Având un număr diferit de prefix acesta determină diferit adresa de broadcast pentru fiecare rețea și numărul de hosturi posibil existente în aceea rețea.

Fig.8.8 ilustrează prefixe diferențiate folosind aceeași adresă 10.1.1.0. În Fig. 8.8 se ilustrează prefixele de la /24 la /26, precum și prefixele /27 și /28.

De reținut faptul că adresa de rețea poate rămâne aceeași, dar spațiul adreselor pentru hosturi și adresa de broadcast sunt diferențiate pentru lungimi de prefix diferențiate. În Fig. 8.8, putem vedea faptul că numărul de hosturi ce pot fi adresate în rețea se schimbă.

Dotted Decimal		Significant bits shown in binary
Network Address	10.1.1.0/24	10.1.1.00000000
First Host Address	10.1.1.1	10.1.1.00000001
Last Host Address	10.1.1.254	10.1.1.11111110
Broadcast Address	10.1.1.255	10.1.1.11111111
Number of hosts:	$2^8 - 2 = 254$ hosts	

Network Address	10.1.1.0/25	10.1.1.00000000
First Host Address	10.1.1.1	10.1.1.00000001
Last Host Address	10.1.1.126	10.1.1.01111110
Broadcast Address	10.1.1.127	10.1.1.01111111
Number of hosts:	$2^7 - 2 = 126$ hosts	

Network Address	10.1.1.0/26	10.1.1.00000000
First Host Address	10.1.1.1	10.1.1.00000001
Last Host Address	10.1.1.62	10.1.1.00111110
Broadcast Address	10.1.1.63	10.1.1.00111111
Number of hosts:	$2^6 - 2 = 62$ hosts	

Fig. 8.8. Prefixe pentru rețele.

Dotted Decimal		Significant bits shown in binary
Network Address	10.1.1.0/27	10.1.1.00000000
First Host Address	10.1.1.1	10.1.1.00000001
Last Host Address	10.1.1.30	10.1.1.00011110
Broadcast Address	10.1.1.31	10.1.1.00011111
Number of hosts:	$2^5 - 2 = 30$ hosts	

Network Address	10.1.1.0/28	10.1.1.00000000
First Host Address	10.1.1.1	10.1.1.00000001
Last Host Address	10.1.1.14	10.1.1.00001110
Broadcast Address	10.1.1.15	10.1.1.00001111
Number of hosts:	$2^4 - 2 = 14$ hosts	

Există trei tipuri de adrese în spațiul de adresă a fiecărei rețele IPv4:

- *Adresa de rețea.*
- *Adresa de broadcast.*
- *Adresele de host.*

8.6 Adresa de rețea

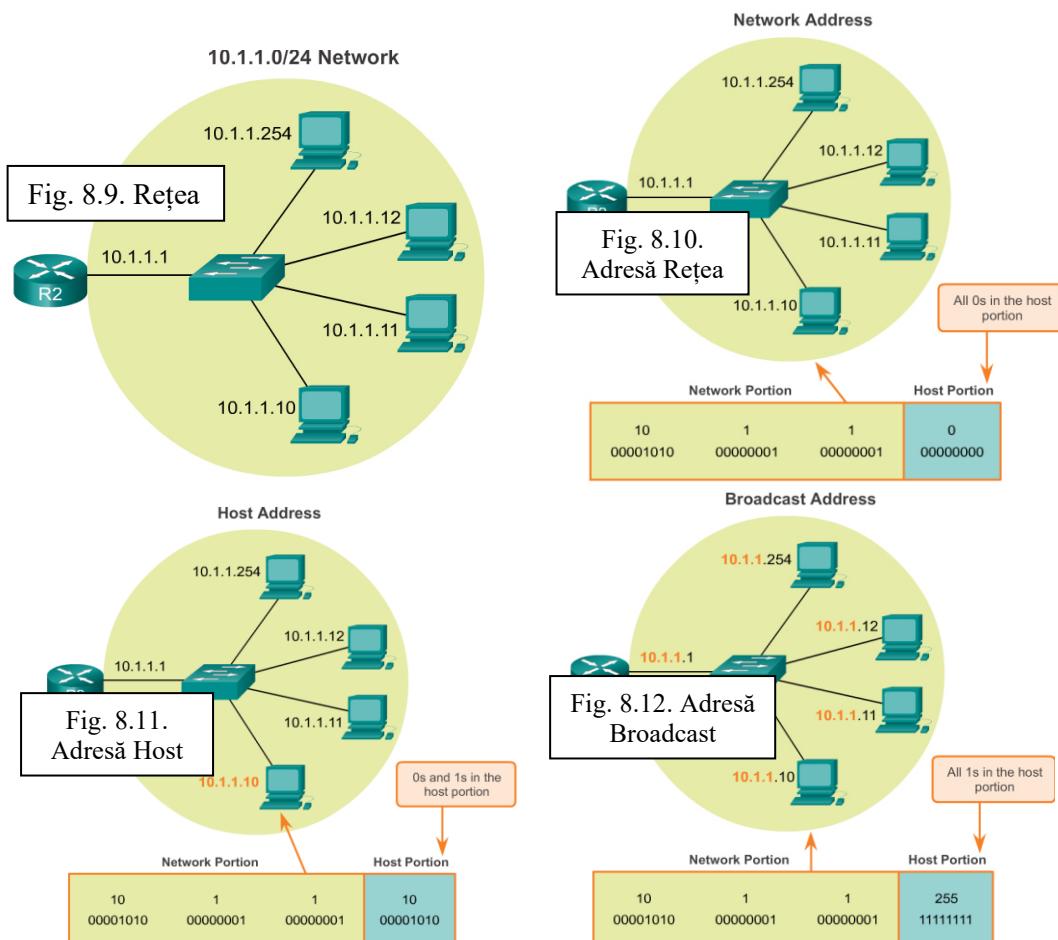
Adresa de rețea este o modalitate standard de referire la o rețea. Masca de rețea sau lungimea prefixului pot fi de asemenea folosite atunci când ne referim la o adresă de rețea. De exemplu, rețeaua din Fig. 1 poate fi referită ca rețeaua 10.1.1.0, rețeaua 10.1.1.0 255.255.255.0 sau rețeaua 10.1.1.0/24. Toate hosturile din rețeaua 10.1.1.0/24 vor avea aceeași biți în partea de rețea.

Așa cum este evidențiat și în Fig. 8.8, într-un spațiu de adrese IPv4 a unei rețele, prima adresă este rezervată adresei de rețea. Această adresă are 0 pentru fiecare host în partea de host a adresei. Toate hosturile din rețea împart aceeași adresă de rețea.

8.7 Adresa de broadcast

Adresa de broadcast IPv4 este o adresă specială pentru fiecare rețea ce permite comunicarea tuturor hosturilor din rețeaua respectivă. Pentru a transmite date la toate hosturile dintr-o rețea în același timp, un host poate transmite un singur pachet ce este adresat adresei de broadcast a rețelei și fiecare host din rețea care va primi pachetul îl va procesa conținutul.

Adresa de broadcast folosește cea mai mare adresă din spațiul de adrese al rețelei. Aceasta este adresa în care biții din partea de host sunt toți 1. Toți 1 într-un octet din format binar este egal cu 255 din format zecimal. Prin urmare, pentru rețeaua 10.1.1.0/24, în care ultimul octet este folosit pentru partea de host, adresa de broadcast va fi 10.1.1.255. De reținut faptul că partea de host nu va fi întotdeauna un octet întreg. Această adresă este referită și ca directed broadcast.



8.8 Adresele de host

Fiecare dispozitiv final necesită o adresă unică pentru a comunica în rețea. În adresele IPv4, valorile dintre adresa de rețea și adresa de broadcast pot fi atribuite dispozitivelor finale dintr-o rețea. Așa cum este arătat și în Fig. 3, aceasta adresa are orice combinatie de 0 și 1 în partea de host a adresei, însă nu poate conține toți biții de 0 sau toți biții de 1.

Pentru a ne asigura că toate hosturile dintr-o rețea au atribuită o adresă IP unică din spațiul adreselor de rețea, este importantă identificarea primei adrese de host și a ultimei adrese de host. Hosturile dintr-o rețea pot avea atribuite adrese IP din acest spațiu.

8.8.1 Prima adresă de host

Ca și în Fig. 1, partea de host a primei adrese de host va conține toți biții de 0 cu 1 bit pentru bitul cel mai din dreapta. Această adresă este întotdeauna cu 1 mai mare decât adresa de rețea. În acest exemplu, prima adresă asignabilă din rețeaua 10.1.1.0/24 este 10.1.1.1. Este comun multor scheme de adresare să se utilizeze prima adresă de host pentru ruter echivalentă cu adresa de default gateway.

8.8.2 Ultima adresă de host

Partea de host a ultimei adrese de host va conține toți biții de 1 cu 0 pentru bitul cel mai din dreapta. Această adresă este întotdeauna cu 1 mai mică decât adresa de broadcast. Ca și în Fig. 2, ultima adresă asignabilă a rețelei 10.1.1.0/24 este 10.1.1.254.

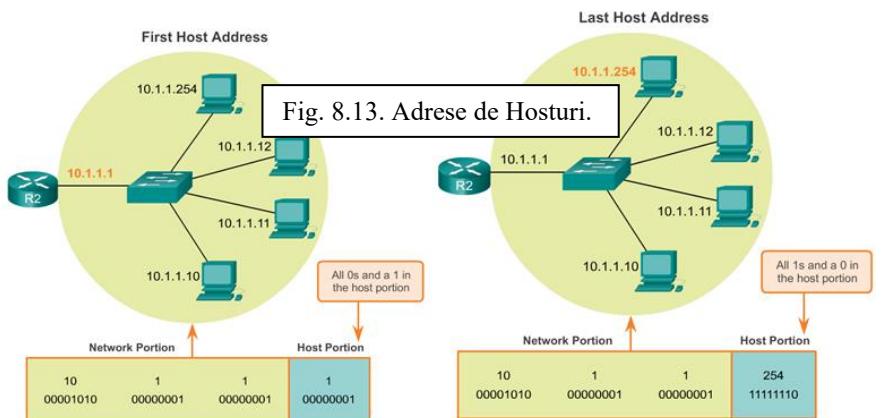


Fig. 8.13. Adrese de Hosturi.

Atunci când o adresă IP este atribuită unui dispozitiv, dispozitivul respectiv folosește masca de rețea pentru a determina adresa de rețea de care aparține. Adresa de rețea este adresa ce reprezintă toate dispozitivele din aceeași rețea.

Atunci când trimitem date de rețea, dispozitivele folosesc această informație pentru a determina dacă pot trimite pachetele local sau dacă trebuie să trimită pachetele la default gateway pentru o livrare la distanță. Atunci când un host trimite un pachet, compară partea de rețea a propriei adrese IP cu partea de rețea a adresei IP destinație, în funcție de masca de rețea. Dacă biții de rețea corespund, hosturile sursă și destinație se află în aceeași rețea și pachetul poate fi livrat local. Dacă nu corespund, hostul sursă trimite pachetul la default gateway pentru a fi transmis într-o altă rețea.

8.9 Operația ȘI LOGIC – ANDing

ANDing este una dintre cele trei operații de bază folosite în logica digitală. Celalalte două sunt OR și NOT. Cele trei sunt folosite în rețele de date și AND este folosită pentru determinarea adresei de rețea. Prin urmare, ne vom limita la AND logic. AND logic este compararea a doi biți ce conduc la următoarele rezultate:

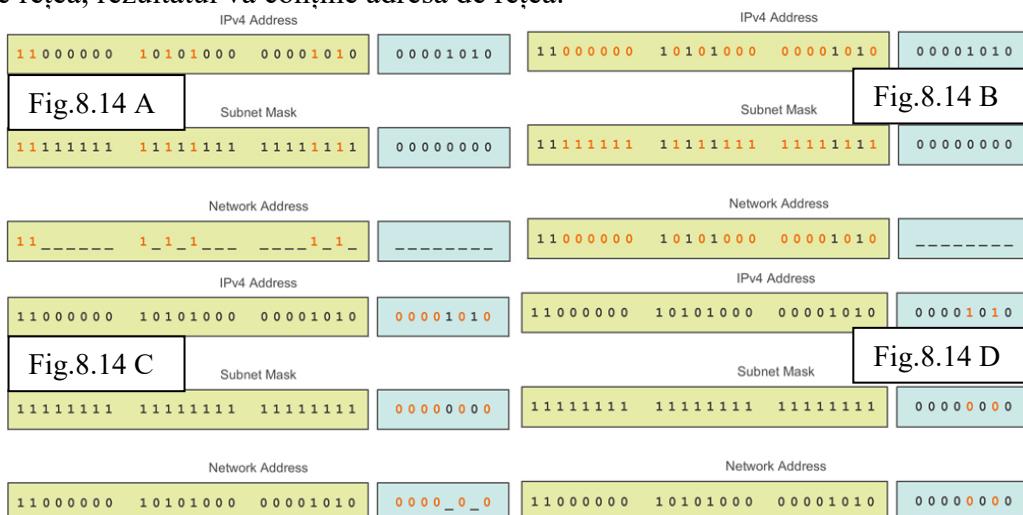
$$1 \text{ AND } 1 = 1 \text{ (Fig.8.14 A)}$$

$$0 \text{ AND } 1 = 0 \text{ (Fig.8.14 B)}$$

$$0 \text{ AND } 0 = 0 \text{ (Fig.8.14 C)}$$

$$1 \text{ AND } 0 = 0 \text{ (Fig.8.14 D)}$$

Adresa de host IPv4 este logic ANDed, bit cu bit, cu masca de rețea proprie pentru a determina adresa de rețea asociată hostului. După efectuarea operației ANDing între adresa și masca de rețea, rezultatul va conține adresa de rețea.



Orice bit de adresă ANDed cu o valoare de 1 din masca de rețea își va păstra valoarea originală din adresă. Deci, un 0 (din adresa IPv4) AND 1 (din masca de rețea) rezultă 0. Un 1 (din adresa IPv4) AND 1 (din masca de rețea) rezultă 1. Orice ANDed cu 0 rezultă 0. Aceste proprietăți ale ANDing sunt folosite cu masca de rețea pentru “a masca biții” de host dintr-o adresă IPv4. Fiecare bit din adresă este ANDed cu bitul corespunzător din masca de rețea.

Deoarece toți biții din masca de rețea ce reprezintă biții de host sunt 0, partea de host a rețelei rezultate este 0. Reamintim faptul că o adresă IPv4 cu toți 0 în partea de host reprezintă adresa de rețea.

De asemenea, toți biții din masca de rețea ce reprezintă biții de rețea sunt 1. Atunci când fiecare dintre acești 1 este ANDed cu bitul corespunzător din adresă, biții rezultați sunt identici cu biții originali de adresă.

Ca și în Fig. , biții de 1 din masca de rețea vor rezulta în partea de rețea a adresei de rețea având aceeași biți ca partea de rețea a hostului. Partea de host a adresei de rețea va rezulta în toți de 0.

Pentru o adresă IP dată și masca sa, ANDing poate fi folosită pentru a determina cărei subrețele aparțină adresa, precum și ce alte adrese aparțin aceleiași subrețele. Reamintim faptul că dacă două adrese se află în aceeași rețea sau subrețea, sunt considerate locale unele cu celelalte și prin urmare, pot comunica direct unele cu celelalte. Adresele ce nu se află în aceeași rețea sau subrețea sunt considerate la distanță și trebuie să existe un dispozitiv de nivel 3 (cum ar fi un router sau switch de nivel 3) între ele pentru a comunica.

În verificarea/depanarea rețelei, adesea trebuie să determinăm două hosturi din aceeași rețea locală. Trebuie să facem această determinare din perspectiva dispozitivelor de rețea. Având în vedere conFig.rea inadecvată, un host ar putea să se afle într-o rețea în care nu ar trebui de fapt. Acest lucru poate duce la o operațiune “dezordonată” doar dacă nu este depistată prin examinarea procesului de ANDing de către host.

IPv4 Address	192	.	168	.	10	.	10
Fig.8.15. ANDing	11000000	10101000	00001010		00001010		
Subnet Mask	255	.	255	.	255	.	0
	11111111	11111111	11111111		00000000		
Network Address	192	.	168	.	10	.	0
	11000000	10101000	00001010		00000000		

8.10 Adresele IPv4 Unicast, Multicast și Broadcast

În mai multe rețele de date, cea mai mare populație de hosturi constă în dispozitive finale cum ar fi PCuri, tablete, smartphoniuri, imprimante și telefoane IP. Deoarece acestea reprezintă cel mai mare număr de dispozitive dintr-o rețea, cel mai mare număr de adrese ar trebui să fie alocate acestora. Aceste hosturi au atribuite adrese IP din spațiul de adrese disponibile în rețea. Aceste adrese IP pot fi atribuite static sau dinamic.

8.10.1 Atribuirea statică

Într-o atribuire statică, administratorul de rețea trebuie să configureze manual informațiile de rețea pentru host. Fig. 1 arată “fereastra” pentru proprietățile adaptorului de rețea. Pentru a configura o adresă statică IPv4, alegem IPv4 din ecranul de adaptor de rețea, apoi introducem adresa statică, masca de rețea și adresa pentru default gateway. Fig. 2 arată configurația statică de bază: adresa IP de host, masca de rețea și adresa default gateway.

Există mai multe avantaje în adresarea statică. De exemplu, sunt utile pentru imprimante, servere și alte dispozitive de rețea ce nu își schimbă locația deoarece și trebuie să fie accesibile clientilor din rețea în funcție de o adresă IP fixă. Dacă hosturile accesează în mod normal un server de la o anumită adresă IP, vor apărea probleme în cazul în care respectiva adresă este schimbată. În plus, atribuirea statică a informațiilor de adresare poate oferi control suplimentar asupra resurselor de rețea. De exemplu, este posibilă crearea de filtre de acces în funcție de traficul de la și la adresa IP respectivă. Însă, adresarea statică poate fi consumatoare de timp pentru fiecare host.

Atunci când utilizăm adresarea IP statică, este necesară menținerea unei liste de adrese IP atribuite fiecărui dispozitiv. Acestea sunt adrese permanente și nu sunt de obicei reutilizate.

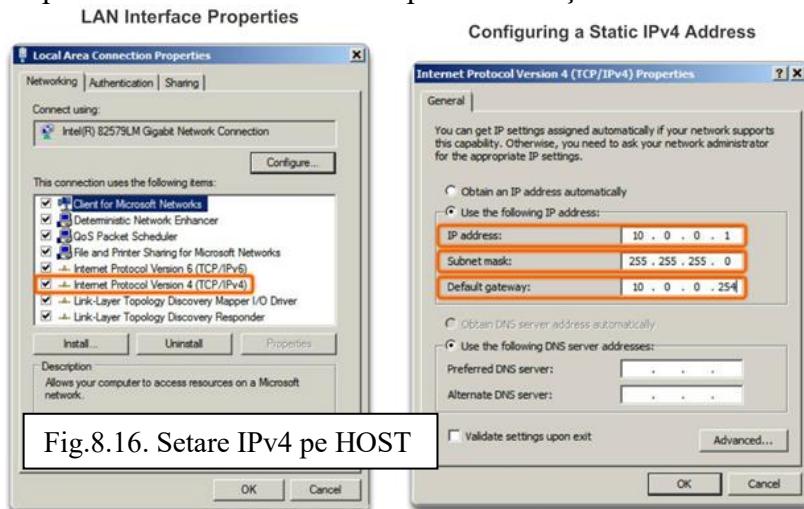


Fig.8.16. Setare IPv4 pe HOST

8.10.2 Atribuirea dinamică

Pe rețelele locale este adesea cazul în care populația de utilizatori se schimbă frecvent. Noi utilizatori vin cu laptopuri și solicită noi conexiuni. Alții au stații de lucru noi sau alte dispozitive de rețea, cum ar fi smartphoneuri, ce necesită conexiune. În loc ca un administrator de rețea să atribuie adrese IP pentru fiecare stație de lucru o adresă statică, este mai ușoară atribuirea adreselor IP în mod automat. Acest lucru se realizează prin folosirea unui protocol numit Dynamic Host Configuration Protocol (DHCP), arătat în Fig.8.17.

DHCP permite atribuirea automată a informațiilor de adresare cum ar fi adrese IP, masca de rețea, adresa default gateway și alte informații de configurație. Un server DHCP necesită ca un block de adrese, numit address pool, să fie folosit pentru atribuirea adreselor spre clienții DHCP dintr-o rețea. Adresele atribuite acestui pool ar trebui să fie planificate astfel încât să excludă orice adresă statică folosită de alte dispozitive.

DHCP este în general metoda preferată de atribuire a adreselor IPv4 hosturilor din rețelele mari deoarece reduce sarcina personalului de suport al rețelei și elimină virtual erorile de introducere.

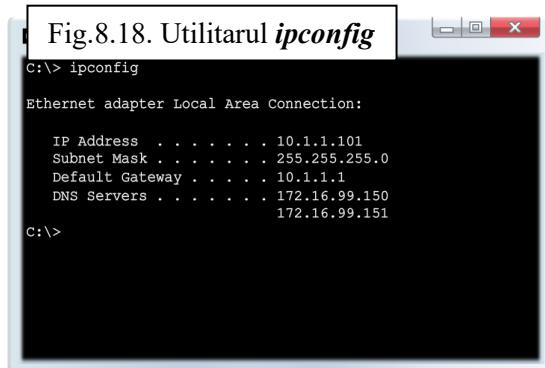
Un alt avantaj al DHCP este acela că o adresă nu este atribuită permanent unui host, însă este "închiriată" pentru o anumită perioadă de timp. Dacă hostul nu este alimentat sau înălțurat din rețea, adresa se întoarce în pool pentru refolosire. Această caracteristică este utilă în mod special pentru utilizatorii mobili care vin și pleacă dintr-o rețea.

Dacă DHCP este activat pe un dispozitiv host, comanda **ipconfig** poate fi folosită pentru vizualizarea informațiilor de adresare IP atribuite de către serverul DHCP, aşa cum se observă și în Fig.8.18.

Assigning a Dynamic IPv4 Address



Verifying a Dynamic IPv4 Address



Într-o rețea IPv4, hosturile pot comunica în unul dintre cele trei moduri:

- **Unicast** - Procesul de trimitere a unui pachet de la un host la un anumit host.
- **Multicast** - Procesul de trimitere a unui pachet de la un host la un anumit grup de hosturi, posibil din rețelele diferite.
- **Broadcast** - Procesul de trimitere a unui pachet de la un host la toate hosturile din aceeași rețea.

Acste trei tipuri de comunicare sunt folosite pentru scopuri diferite în rețelele de date. În toate cele trei cazuri, adresa IPv4 a hostului sursă este plasată în headerul pachetului ca adresa sursă.

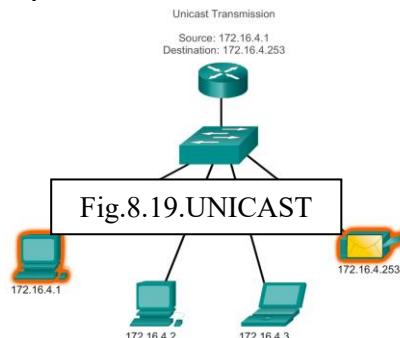
8.10.2.1 Traficul Unicast

Comunicarea unicast este folosită pentru comunicarea normală host-la-host în rețelele client/server și peer-to-peer. Pachetele unicast folosesc adresele dispozitivului destinație ca adresă destinație și pot fi dirigate printr-o internetwork.

Într-o rețea IPv4, adresele unicast aplicate pe un dispozitiv final se numește adresă de host. Pentru comunicarea unicast, adresele atribuite celor două dispozitive finale sunt folosite ca adrese sursă și destinație. În timpul procesului de încapsulare, hostul sursă plasează adresa sa IPv4 în headerul de pachet unicast ca adresa sursă și adresa IPv4 a hostului destinație în header ca adresa destinație. Indiferent dacă destinația specificată este un unicast, broadcast sau multicast, adresa sursă a oricărui pachet este întotdeauna o adresă unicast a hostului sursă.

Notă: În acest curs, toate comunicațiile dintre dispozitive sunt comunicații unicast, doar dacă nu este specificat un alt mod.

Adresele de host IPv4 sunt adrese unicast și sunt în spațiul de adrese de la 0.0.0.0 la 255.255.255.255. Însă, în spațiul de adrese sunt mai multe adrese rezervate pentru anumite scopuri. Aceste adrese cu scopuri speciale vor fi discutate mai târziu, în acest capitol.



8.10.2.2 Traficul Broadcast

Traficul de broadcast este folosit pentru a transmite pachete la toate hosturile din rețea folosind adresa broadcast a rețelei. Cu un broadcast, pachetul conține o adresă IP destinație cu toți de 1 în partea de rețea. Acest lucru înseamnă că toate hosturile din rețeaua locală (domeniul de broadcast) vor primi și se vor uita la pachet. Mai multe protocoale de rețea, cum ar fi DHCP, folosesc broadcasturi. Atunci când un host primește un pachet trimis la o adresă de rețea broadcast, procesează pachetului în modul în care ar fi adresat adresei sale unicast.

Unele exemple de folosire a transmisiei broadcast sunt:

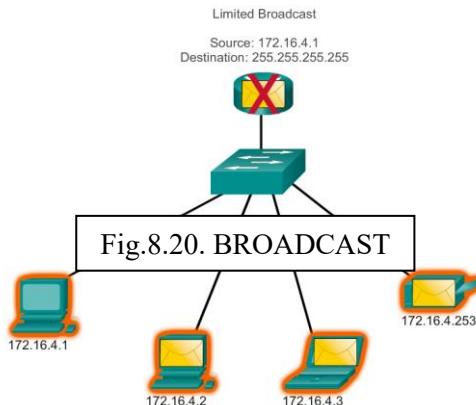
- *Maparea adreselor de nivel superioar la adresele de nivel inferior.*
- *Cererea unei adrese.*
- *Spre deosebire de unicast, unde pachetele pot fi dirigate printr-o internetwork, pachetele broadcast sunt de obicei restricționate la rețeaua locală. Această restricție este dependentă de configurația gateway router și de tipul de broadcast. Există două tipuri de broadcast: broadcast dirijat (controlat) și broadcast limitat.*

Directed Broadcast – Un broadcast dirijat este trimis la toate hosturile dintr-o anumită rețea. Acest tip de broadcast este util pentru transmiterea unui broadcast tuturor hosturilor dintr-o rețea non-locală. De exemplu, pentru ca un host din exteriorul rețelei 172.16.4.0/24 să comunice cu toate hosturile din rețea, adresa destinație a pachetului va fi 172.16.4.255. Deși routerele nu transmit directed broadcasts implicit, pot fi configurate să facă acest lucru.

Limited Broadcast – Broadcastul limitat este folosit pentru comunicarea limitată hosturilor din rețeaua locală. Aceste pachete folosesc întotdeauna o adresă IPv4 destinație 255.255.255.255. Routerele nu transmit un broadcast limitat. Din acest motiv, o rețea IPv4 este referită ca un domeniu de broadcast. Routerele formează bariera pentru un domeniu de broadcast.

Ca exemplu, un host din rețeaua 172.16.4.0/24 va transmite un broadcast tuturor hosturilor din rețea folosind un pachet cu o adresă destinație 255.255.255.255.

Atunci când un pachet este de tip broadcast, folosește resursele rețelei și face ca fiecare destinatar din rețea să proceseze pachetul. Prin urmare, traficul de broadcast ar trebui să fie limitat astfel încât să nu afecteze performanța rețelei sau a dispozitivelor. Deoarece routerele separă domeniile de broadcast, împărțirea rețelelor cu broadcast excesiv va îmbunătăți performanța rețelei.



8.10.2.3 Traficul Multicast

Transmisia multicast este proiectată pentru a conserva lățimea de bandă a unei rețele IPv4. Reduce traficul prin permiterea unui host să trimită un singur pachet la un anumit set de hosturi ce fac parte dintr-un grup multicast. Pentru a ajunge la mai multe hosturi destinație cu ajutorul comunicării unicast, o sursă va trebui să trimită un pachet adresat fiecărui host. Cu multicast,

hostul sursă poate trimite un singur pachet ce poate ajunge la mii de hosturi destinație. Responsabilitatea internetwork este de a reproduce fluxurile multicast într-o manieră

Unele exemple de transmisie multicast sunt:

- *Video și audio broadcasts.*
- *Schimbul de informații de rutare din protocole de rutare.*
- *Distribuția de software.*
- *Gaming de la distanță.*

8.10.2.3.1 Adrese Multicast

IPv4 are un bloc de adrese rezervate grupurilor de adresare multicast. Acest spațiu de adrese este de la 224.0.0.0 la 239.255.255.255. Spațiul de adrese multicast este divizat în diferite tipuri de adrese: adrese rezervate de legătură locală și adrese de domeniu global. Un tip suplimentar de adresă multicast sunt adreselor cu scop administrativ, numite și adrese cu scop limitat.

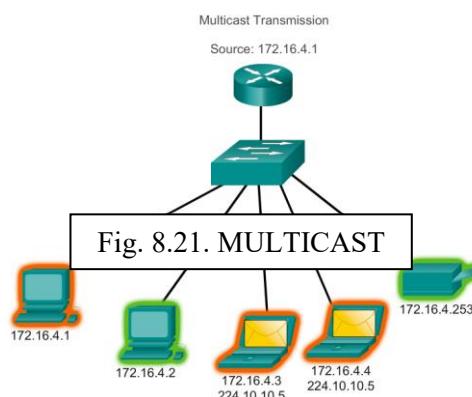
Adresele multicast IPv4 de la 224.0.0.0 la 224.0.0.255 sunt adrese rezervate de legătură locală. Aceste adrese sunt folosite pentru grupuri multicast dintr-o rețea locală. Un router conectat la rețeaua locală recunoaște aceste pachete ca fiind adresate unui grup multicast cu legătură locală și nu le transmite niciodată mai departe. O utilizare tipică a adreselor rezervate cu legătură locală este în protocolele de rutare ce folosesc transmisii multicast pentru schimbul de informații de rutare.

Adrese de domeniu global sunt de la 224.0.1.0 la 238.255.255.255. Ele pot fi folosite pentru multicast de date pe Internet. De exemplu, 224.0.1.1 a fost rezervată pentru Network Time Protocol (NTP) pentru a sincroniza ceasurile dispozitivelor de rețea.

8.10.2.3.2 Clienții Multicast

Hosturile ce primesc anumite date multicast sunt numite clienți multicast. Clienții multicast sunt servicii necesare unui program client pentru a se înscrie într-un grup multicast.

Fiecare grup multicast este reprezentat de o adresă destinație multicast IPv4. Atunci când un host se înscrie la un grup multicast, hostul procesează pachetele adresate acestei adrese multicast și cele adresate adresei unică alocată unic.



8.11 Tipuri de adrese IPv4

Deși cele mai multe adrese IPv4 sunt publice destinate utilizării în rețelele accesibile pe Internet, există blocuri de adrese folosite în rețele ce necesită un acces la Internet limitat sau deloc. Aceste adrese se numesc adrese private.

8.11.1 Adrese private

Blocurile de adrese private sunt:

- *10.0.0.0 la 10.255.255.255 (10.0.0.0/8).*
- *172.16.0.0 la 172.31.255.255 (172.16.0.0/12).*
- *192.168.0.0 la 192.168.255.255 (192.168.0.0/16).*

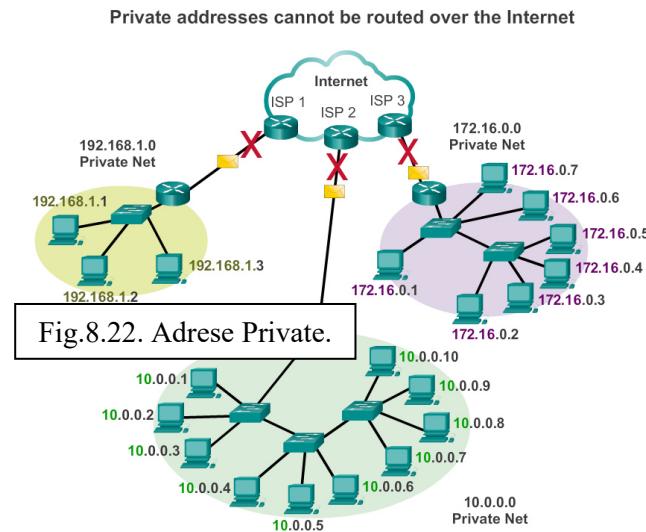
Adresele private sunt definite în RFC 1918, Address Allocation for Private Internets, și ne sunt uneori referite ca adrese RFC 1918. Blocurile private din spațiul de adrese, aşa cum sunt arătate și în Fig. , sunt folosite în rețelele private. Hosturile ce nu necesită acces la internet pot folosi adrese private. Însă într-o rețea privată, hosturile necesită o adresă IP unică din spațiul privat.

Hosturile din rețele diferite pot folosi aceleași adrese din spațiul privat. Pachetele ce folosesc aceste adrese ca sursă sau destinație nu ar trebui să apară în internetul public. Routerul sau dispozitivul firewall de la perimetru acestor rețele private trebuie să blocheze sau să traducă aceste adrese. Chiar dacă aceste pachete sunt intenționate pentru Internet, routerele nu ar trebui să aibă rute pentru transmiterea lor la rețeaua privată corespunzătoare.

În RFC 6598, IANA a rezervat un alt grup de adrese numit spațiu de adrese partajat. Similar spațiului de adrese publice RFC 1918, adresele de spațiu de adrese partajate nu sunt routabile global. Însă, aceste adrese sunt folosite numai pentru rețelele ale furnizorilor de servicii. Blocul de adrese partajate este 100.64.0.0/10.

8.11.2 Adrese publice

Marea majoritate a adreselor din spațiul de adrese unicast de host IPv4 sunt adrese publice. Aceste adrese sunt dezvoltate pentru a fi folosite la hosturile ce sunt accesibile public din Internet. Chiar în aceste blocuri de adrese IPv4, există mai multe adrese desemnate pentru alte scopuri speciale.



Există anumite adrese ce nu pot fi atribuite hosturilor. Există de asemenea și adrese speciale ce pot fi asignate hosturilor, însă cu restricții a modului în care acele hosturi pot interacționa în rețea.

Adrese de Rețea și Broadcast – Cum a fost mentionat mai devreme, fiecare prima și ultima adresa dintr-o rețea nu pot fi atribuite hosturilor. Acestea sunt adrese de rețea și de broadcast.

Adresa Loopback – Una dintre adresele rezervate este adresa IPv4 loopback 127.0.0.1. Loopback este o adresă specială pe care hosturile o folosesc pentru direcționarea traficului către

ele însăși. Adresa de loopback crează o metodă mai scurtă pentru ca aplicațiile și serviciile TCP/IP ce rulează pe același dispozitiv să comunice unele cu celelalte. Prin utilizarea adresei de loopback în schimbul adresei de host IPv4 atribuită, două servicii de pe același host pot trece la nivelele inferioare ale stivei TCP/IP. Putem de asemenea să dăm **ping** la adresa loopback pentru a testa configurația TCP/IP de pe hostul local.

Deși numai adresa 127.0.0.1 este folosită, adresele de la 127.0.0.0 la 127.255.255.255 sunt rezervate. Orice adresă din acest bloc va bucla înapoi la hostul local. Nici-o adresă din acest bloc nu ar trebui să apară într-o rețea.

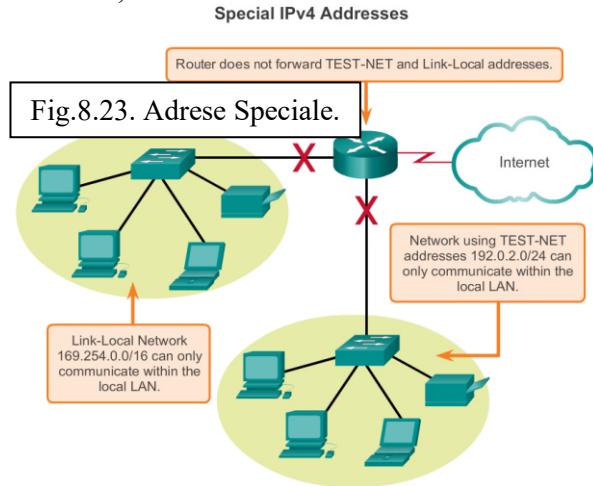
Adresa de tip Link-Local – Adresele IPv4 din blocul de adrese de la 169.254.0.0 la 169.254.255.255 (169.254.0.0/16) sunt desemnate ca adrese de legătură locală. Aceste adrese pot fi atribuite automat hostului local de către sistemul de operare în medii în care nu există nici-o configurație IP. Acestea pot fi folosite într-o rețea mică peer-to-peer sau pentru un host ce nu poate să obțină automat o adresă de la un server DHCP.

Comunicarea ce folosește adrese IPv4 de legătură locală este potrivită numai pentru comunicarea cu dispozitivele conectate la aceeași rețea, așa cum este evidențiat și în Fig. . Un host nu trebuie să trimită un pachet cu o adresă destinație IPv4 de legătură locală routerului pentru expediere lui mai departe și de aceea, ar trebui ca time to live (TTL) al acestor pachete să fie setat la 1.

Adresele de legătură locală nu oferă servicii în afara rețelei locale. Însă, mai multe aplicații client/server și peer-to-peer vor funcționa în mod corespunzător cu adrese IPv4 de legătură locală.

Adrese de tip TEST-NET – Blocul de adrese de la 192.0.2.0 la 192.0.2.255 (192.0.2.0/24) este păstrat pentru scopuri didactice și de învățare. Aceste adrese pot fi folosite în documentație și exemple de rețea. Spre deosebire de adresele experimentale, dispozitivele de rețea vor accepta aceste adrese în configurațiile lor. Putem întâlni deseori aceste adrese folosite în domain names example.com or example.net în RFCs, furnizor și documentație de protocol. Adresele din acest bloc nu ar trebui să apară în Internet.

Adrese de tip Experimental – Adresele din blocul de la 240.0.0.0 la 255.255.255.254 sunt rezervate pentru utilizări viitoare (RFC 3330). Actual, aceste adrese pot fi folosite pentru scopuri de cercetare sau experimentare, însă nu pot fi folosite într-o rețea IPv4. Însă, conform RFC 3330, ele pot, din punct de vedere tehnic, să fie convertite în adrese utilizabile în viitor.



Din punct de vedere istoric, RFC1700, Assigned Numbers, au grupat spațiile de adrese unicast în dimensiuni specifice numite adrese de clasă A, clasă B și clasă C. Defineste de

asemenea și adrese de clasă D (multicast) și E (experimental). Clasele de adrese unicast A, B și C definesc rețele de dimensiune precisă și blocuri de adresa specifice pentru aceste rețele. Unei companii sau organizații i s-a atribuit o întreaga rețea dintr-un bloc de adresă de clasă A, clasă B sau clasă C. Această utilizare a spațiului de adrese se numește adresare classful.

8.11.3 Blocul de adrese Clasă A

Un bloc de adrese de clasă A a fost proiectat pentru a suporta rețelele de mărime foarte mare, cu mai mult de 16 milioane de adrese de host. Adresele IPv4 de clasă A folosesc un prefix fix /8 cu primul octet indicând adresa de rețea. Ceilalți trei octeți sunt folosiți pentru adresele de host. Toate adresele de clasă A necesită ca cel mai semnificativ bit al octetului de ordine cea mai mare să fie zero. Acest lucru înseamnă că există numai 128 de posibile rețele de clasa A, de la 0.0.0.0/8 la 127.0.0.0/8. Chiar dacă adresele de clasă A rezervă jumătate pentru spațiul de adresa, datorită limitei de 128 de rețele, pot fi alocate numai aproximativ la 120 de companii și organizații.

8.11.4 Blocul de adrese Clasă B

Spațiul de adrese de clasă B a fost proiectat pentru a suporta nevoile rețelelor de dimensiune moderată spre mare, cu aproximativ 65.000 de hosturi. O adresă IP de clasă B folosește cei doi octeți de ordine mai mare pentru a indica adresa de rețea. Ceilalți doi octeți specifică adresele de host. Ca și clasa A, spațiul de adrese pentru clasele de adresă ramase trebuie să fie rezervate. Pentru adresele de clasă B, cei mai semnificativi doi biți ai octetului de ordine cea mai mare sunt 10. Acest lucru restricționează blocul de adrese pentru clasa B de la 128.0.0.0/16 la 191.255.0.0/16. Clasa B a avut o alocare de adrese ușor mai eficientă decât clasa A deoarece împarte în mod egal 25% din totalul de spațiu de adrese IPv4 la aproximativ 16.000 de rețele.

8.11.5 Blocul de adrese Clasă C

Spațiul de adrese de clasă C a fost cel mai frecvent disponibil dintre clasele de adrese istorice. Acest spațiu de adrese a fost proiectat pentru a oferi adrese pentru rețele mici cu un maxim de 254 hosturi. Blocurile de adrese de clasă C folosesc un prefix /24. Acest lucru înseamnă că o rețea de clasă C folosește numai ultimul octet ca adresa de host cu trei octeți de ordine cea mai mare folosiți pentru a indica adresa de rețea. Blocurile de adrese de clasă C rezervă spațiu de adrese prin utilizarea unei valori fixe de 110 pentru cei mai semnificativi 3 biți ai octetului cu ordinea cea mai mare. Acest lucru restricționează blocul de adrese pentru clasa C de la 192.0.0.0/24 la 223.255.255.0/24. Deși ocupă numai 12.5% din totalul spațiului de adrese IPv4, poate oferi adrese pentru 2 milioane de rețele.

Limitări ale Sistemului Claselor de Bază – Nu toate cerințele organizațiilor se potrivesc într-o singură rețea. Alocarea classful a spațiului de adrese adesea risipește multe adrese, ceea ce epuizează disponibilitatea adreselor IPv4. De exemplu, o companie ce are o rețea de 260 de hosturi va avea nevoie de o adresă de clasa B cu mai mult de 65.000 de adrese.

Deși acest sistem classful a fost abandonat la sfârșitul anilor 1990, putem vedea resturi din el în rețelele de astăzi. De exemplu, atunci când atribuim o adresă IPv4 unui computer, sistemul de operare examinează adresa atribuită pentru a determina dacă este de clasă A, clasă B sau clasă C. Sistemul de operare apoi ia în considerare prefixul folosit de clasa respectivă și atribuie masca de rețea implicită.

8.11.6 Adresarea de tip Classless

Sistemul folosit astăzi se numște adresare classless. Numele formal este Classless Inter-Domain Routing (CIDR). Alocarea classful a adreselor IPv4 a fost foarte ineficientă, permitând numai lungimi de prefix /8, /16 sau /24, fiecare dintr-un spațiu de adrese diferit. În 1993, IETF a creat un nou set de standarde ce permit furnizorilor de servicii să aloce adrese IPv4 cu orice lungime de prefix în loc să fie adresate numai adrese din clasa A, B sau C.

IETF a știut că CIDR a fost numai o soluție temporară și că un nou protocol IP trebuie să fie proiectat pentru a întâmpina creșterea rapidă a numărului de utilizatori de Internet. În 1994, IETF a început să lucreze la găsirea unui succesor pentru IPv4, ce a devenit IPv6.

Fig.8.24 arată spațiile de adrese de tip Classless, iar Fig.8.25 arată spațiile de adrese de tip classful.

Fig.8.24. Adresare Classless

11111111.00000000.00000000.00000000	/8 (255.0.0.0) 16,777,214 host addresses
11111111.10000000.00000000.00000000	/9 (255.128.0.0) 8,388,606 host addresses
11111111.11000000.00000000.00000000	/10 (255.192.0.0) 4,194,302 host addresses
11111111.11100000.00000000.00000000	/11 (255.224.0.0) 2,097,150 host addresses
11111111.11110000.00000000.00000000	/12 (255.240.0.0) 1,048,574 host addresses
11111111.11111000.00000000.00000000	/13 (255.248.0.0) 524,286 host addresses
11111111.11111100.00000000.00000000	/14 (255.252.0.0) 262,142 host addresses
11111111.11111110.00000000.00000000	/15 (255.254.0.0) 131,070 addresses
11111111.11111111.00000000.00000000	/16 (255.255.0.0) 65,534 host addresses
11111111.11111111.10000000.00000000	/17 (255.255.128.0) 32,766 host addresses
11111111.11111111.11000000.00000000	/18 (255.255.192.0) 16,382 host addresses
11111111.11111111.11100000.00000000	/19 (255.255.224.0) 8,190 host addresses
11111111.11111111.11110000.00000000	/20 (255.255.240.0) 4,094 host addresses
11111111.11111111.11111000.00000000	/21 (255.255.248.0) 2,046 host addresses
11111111.11111111.11111100.00000000	/22 (255.255.252.0) 1,022 host addresses
11111111.11111111.11111110.00000000	/23 (255.255.254.0) 510 host addresses
11111111.11111111.11111111.00000000	/24 (255.255.255.0) 254 host addresses
11111111.11111111.11111111.10000000	/25 (255.255.255.128) 126 host addresses
11111111.11111111.11111111.11000000	/26 (255.255.255.192) 62 host addresses
11111111.11111111.11111111.11100000	/27 (255.255.255.224) 30 host addresses
11111111.11111111.11111111.11110000	/28 (255.255.255.240) 14 host addresses
11111111.11111111.11111111.11111000	/29 (255.255.255.248) 6 host addresses
11111111.11111111.11111111.11111100	/30 (255.255.255.252) 2 host addresses
11111111.11111111.11111111.11111110	/31 (255.255.255.254) 0 host addresses
11111111.11111111.11111111.11111111	/32 (255.255.255.255) "Host Route"

Fig.8.25. Adresare Classful

Address Class	1st octet range (decimal)	1st octet bits do	parts of address		Number of possible networks and hosts per network
			and binary		
A	1-127**	00000000-01111111	N.H.H.H	255.0.0.0	128 nets (2^7) 16,777,214 hosts per net (2^{24-2})
B	128-191	10000000-10111111	N.N.H.H	255.255.0.0	16,384 nets (2^{14}) 65,534 hosts per net (2^{16-2})
C	192-223	11000000-11011111	N.N.N.H	255.255.255.0	2,097,150 nets (2^{21}) 254 hosts per net (2^{8-2})
D	224-239	11100000-11101111	NA (multicast)		
E	240-255	11110000-11111111	NA (experimental)		

Note: All zeros (0) and all ones (1) are invalid hosts addresses.

Pentru ca o companie sau organizație să aibă hosturi de rețea, cum ar fi servere web, accesibile din Internet, organizația trebuie să aibă un bloc de adrese publice atribuite. Reamintim faptul că adresele publice trebuie să fie unice și utilizarea acestor adrese publice este reglementată și alocată fiecărei organizații în mod separat. Acest lucru este adevărat pentru adresele IPv4 și IPv6.

8.12 IANA și RIRs

Internet Assigned Numbers Authority (IANA) (<http://www.iana.org>) gestionează alocarea de adrese IPv4 și IPv6. Înainte de mijlocul anilor 1990, toate spațiile de adrese IPv4 au fost gestionate direct de către IANA. La acel moment, spațiul rămas de adrese IPv4 a fost alocat la

diverse alte registre pentru a-l gestiona pentru diferite scopuri sau arii regionale. Aceste companii se numesc Regional Internet Registries (RIRs).

Principalele registre regionale sunt:

- AfriNIC (African Network Information Centre) - Africa Region <http://www.afrinic.net>
- APNIC (Asia Pacific Network Information Centre) - Asia/Pacific Region <http://www.apnic.net>
- ARIN (American Registry for Internet Numbers) - North America Region <http://www.arin.net>
- LACNIC (Regional Latin-American and Caribbean IP Address Registry) - Latin America and some Caribbean Islands <http://www.lacnic.net>
- RIPE NCC (Reseaux IP Europeans) - Europe, the Middle East, and Central Asia <http://www.ripe.net>

8.13 ISPs

RIRs sunt responsabile de alocarea de adrese IP pentru Internet Service Providers (ISPs). Cele mai multe companii sau organizații obțin blocurile de adrese IPv4 de la un ISP. Un ISP va furniza în general un număr mic de adrese IPv4 utilizabile (6 sau 14) clienților ca parte a serviciilor lor. Blocuri mai mari de adrese pot fi obținute în funcție de justificare nevoilor și la costuri de serviciu suplimentare.

Într-un sens, ISP împrumută sau închiriază aceste adrese organizației. Dacă alegem să ne schimbăm conectivitatea la Internet la alt ISP, noul ISP ne va oferi adrese din blocuri de adresă oferite lor și ISP anterior ne ia înapoi blocurile împrumutate nouă pentru a le împrumuta altui client.

Adresele IPv6 pot fi obținute de la ISP sau în unele cazuri direct de la RIR. Adresele IPv6 și dimensiunile tipice de bloc de adresă vor fi discutate ulterior în acest capitol.



8.13.1 Servicii ISP

Pentru a avea acces la serviciile din Internet, trebuie să ne conectăm rețeaua noastră de date la Internet prin folosirea unui Internet Service Provider (ISP).

ISPs au propriul set de rețele de date interne pentru a gestiona conectivitatea la Internet și pentru a oferi serviciile asociate. Împreună cu alte servicii pe care un ISP de oferă în general clienților sunt serviciile DNS, serviciile de e-mail și un website. În funcție de nivelul de servicii necesar și disponibil, clienții folosesc diferite nivele ale unui ISP.

8.13.2 ISP Tiers

ISPs sunt dezvoltăți ierarhic în funcție de nivelul lor de conectivitate la Internet. Fiecare nivel inferior obține conectivitate la Internet printr-o conexiune de la un nivel superior ISP, aşa cum este evidențiat și în Fig. .

8.13.2.1 Tier 1

Așa cum este arătat și în Fig.8.27.A, la nivelul superior al ierarhiei ISP este nivelul 1 ISPs. Aceste ISPs sunt ISPs mari internaționale sau naționale care sunt direct conectate la Internet. Clienții lor sunt fie ISPs de nivel inferior, fie companii și organizații mari. Deoarece ele sunt în vârful conectivității la Internet, construiesc conexiuni și servicii de înaltă încredere. Printre tehnologiile folosite pentru suportul acestei încrederi sunt conexiuni multiple la Internet.

Principalul avantaj al clientilor de nivel 1 ISPs sunt încrederea și viteza. Deoarece acești clienti au acces la o conexiune departe de Internet, există mai puține oportunități de eșec sau blocaje de trafic. Dezavantajul pentru clienții acestui nivel 1 ISP este costul ridicat.

8.13.2.2 Tier 2

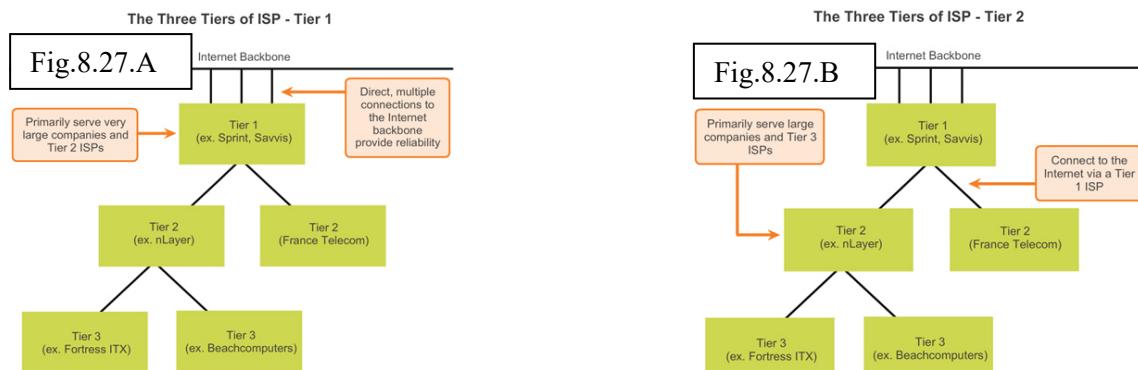
Așa cum se observă și în Fig.8.27.B, nivelul 2 ISPs achiziționează serviciul lor Internet de la nivelul 1 ISPs. Nivelul 2 ISPs se axează pe clienții business. Nivelul 2 ISPs de obicei oferă mai multe servicii decât celelalte nivele de ISPs. Aceste nivele 2 ISPs tind să aibă resursele IT necesare pentru a funcționa propriile servicii cum ar fi DNS, e-mail, servere și web servere. Alte servicii pe care nivelul 2 ISPs le poate oferi sunt dezvoltarea și gestionarea de website, e-commerce/e-business și VoIP.

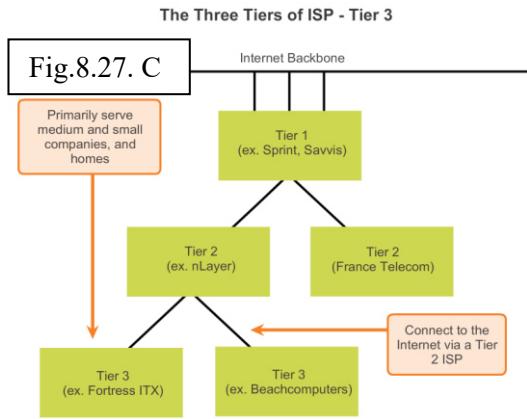
Principalul dezavantaj al nivelului 2 ISPs, în comparație cu nivelul 1 ISPs, este accesul mai lent la Internet. Deoarece nivelul 2 ISPs au cu o conexiune mai departată de Internet, tind să aibă încredere mai mică decât nivelul 1 ISPs.

8.13.2.3 Tier 3

Așa cum se observă și în Fig.8.27. C, nivelul 3 ISPs achiziționează serviciul lor Internet de la nivelul 2 ISPs. Acestea se axează pe vânzare și piețele locale dintr-o anumită zonă. Clienții de nivel 3 de obicei nu necesită multe servicii oferite de clienții de nivel 2. Principala lor necesitate este conectivitatea și suportul.

Acesti clienți adesea au puțină sau nici-o expertiză în rețea sau computer. Nivelul 3 ISPs adesea oferă conectivitate la Internet ca o parte a contractelor de serviciu de computer sau rețea pentru clienții lor. Deși ele pot avea lățime de bandă mai redusă și încredere mai scăzută decât ale furnizorilor de nivel 2 și 1, sunt de obicei alegeri bune pentru companii mici sau de dimensiune medie.





8.14 Adresele de Rețea IPv6

IPv6 este proiectat pentru a fi succesorul lui IPv4. IPv6 are un spațiu de adrese mai mare de 128 de biți, oferind 340 undecillion de adrese (numărul 340, urmat de 36 de zerouri). Însă, IPv6 este mai mult decât un spațiu de adrese mai mari. Atunci când IETF a început dezvoltarea unui succesor pentru IPv4, a profitat de aceasta ocazie pentru a îmbunătăți limitările IPv4 și pentru a include îmbunătățiri suplimentare. Un exemplu este Internet Control Message Protocol version 6 (ICMPv6), ce include rezoluție de adresă și auto-Config.re de adresă ce nu se găsesc în ICMPv4. ICMPv4 și ICMPv6 vor fi discutate ulterior în acest capitol.

8.14.1 Necessitatea de adresare IPv6

Epuizarea spațiului de adrese IPv4 a fost factorul motivant pentru trecerea la IPv6. O dată ce Africa, Asia și alte arii din lume s-au conectat mai mult la Internet, nu mai există destule adrese IPv4 pentru a susține această creștere. Luni, ianuarie 31, 2011, IANA a alocat ultimile două blocuri de adresa IPv4 /8 pentru Regional Internet Registries (RIRs). Diferite proiecții arată că toate cele cinci RIRs vor rămâne fără adrese IPv4 între anii 2015 și 2020. În acel moment, adresele IPv4 rămase vor fi alocate la ISP.

IPv4 are teoretic maxim 4,3 miliarde de adrese. Adresele private RFC 1918 în combinație cu Network Address Translation (NAT) au avut un rol esențial în încetinirea epuizării spațiului de adrese IPv4. NAT are limitări care împiedică strict comunicațiile peer-to-peer.

8.14.2 Internetul pentru toate lucrurile

Internetul de azi este semnificativ mai diferit de Internetul din ultimele decenii. Internetul de azi este mai mult decât e-mail, pagini web și transferul de fișiere dintre computere. Internetul evoluat devine un Internet de lucruri. Computerele, tabletele și smartphoneurile nu vor mai fi singurele dispozitive ce accesează Internetul. Dispozitivele “de mâine” echipate cu senzori vor include dispozitivele de la automobile și dispozitivele biomédicale la ecosistemele naturale și aparatele de uz casnic. Să ne imaginăm o întâlnire cu un client programat automat pe o aplicație calendar ce începe cu o oră înaintea programului normal. Acest lucru ar putea fi o problemă semnificativă, în special dacă uităm să verificăm calendarul sau să schimbăm alarmă de la ceas. Acum să ne imaginăm că aplicația calendar comunică această informație direct alarmei ceasului și automobilului. Mașina se va încărzi automat pentru a topi gheata de pe parbriz înainte de a intra în mașină și ne redirecționează la întâlnire.

Cu o populație, ce accesează Internetul, tot mai mare, un spațiu de adrese IPv4 limitat, problemele cu NAT și cerințele de conectare la Internet a numeroase noi lucruri, a venit timpul pentru a începe tranziția la IPv6.

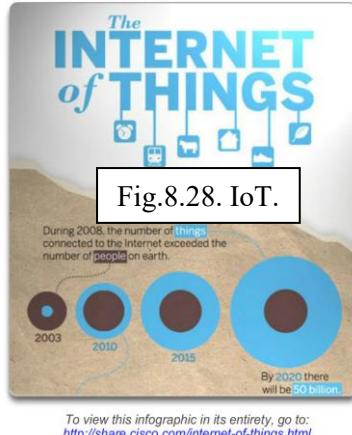


Fig.8.28. IoT.

Nu există o dată de trecere la IPv6. În viitor, IPv4 și IPv6 vor coexista. Tranzitia poate dura ani. IETF a creat diverse protocoale și instrumente pentru a ajuta administratorii de rețea să migreze rețelele lor la IPv6. Tehnicile de migrare pot fi divizate în trei categorii:

- **Dual Stack** - Așa cum se observă în Fig.8.29, dual stack permite ca IPv4 și IPv6 să coexiste în aceeași rețea. Dispozitivele Dual stack rulează stivele de protocol IPv4 și IPv6 simultan.
- **Tunneling** - Așa cum se observă în Fig.8.30, tunneling este o metodă de transport a pachetului IPv6 peste o rețea IPv4. Pachetul IPv6 este încapsulat în interiorul unui pachet IPv4, în mod similar altor tipuri de date.
- **Translation** – Așa cum se observă în Fig.8.31, Network Address Translation 64 (NAT64) permite dispozitivelor cu IPv6 să comunice cu dispozitivele cu IPv4 prin folosirea unei tehnici de traducere similară cu NAT pentru IPv4. Un pachet IPv6 este interpretat ca un pachet IPv4 și invers.

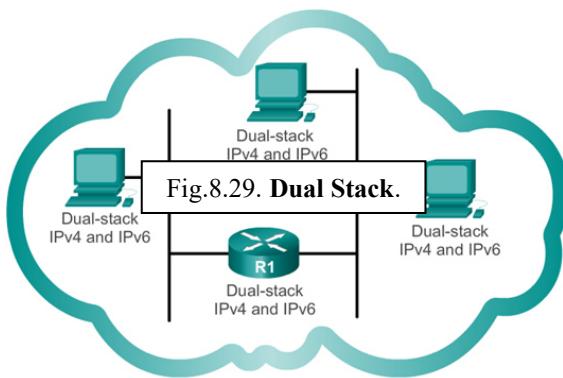


Fig.8.29. Dual Stack.

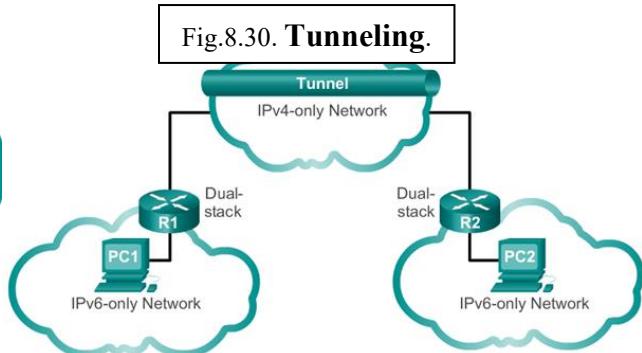


Fig.8.30. Tunneling.

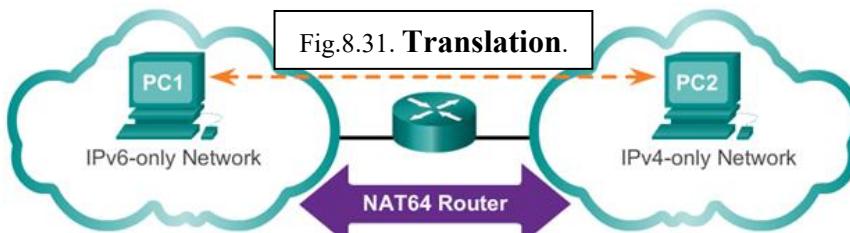


Fig.8.31. Translation.

8.14.3 Adresarea IPv6

Spre deosebire de adresele IPv4 ce sunt exprimate în notație zecimală punctată, adresele IPv6 sunt reprezentate cu ajutorul valorilor hexazecimale. În Wireshark, hexazecimal este folosit pentru reprezentarea valorilor binare din frameuri și pachete. Hexazecimal este de asemenea folosit pentru a reprezenta adresele Ethernet Media Access Control (MAC).

Numerație în Hexadecimal – Hexazecimal ("Hex") este un mod convenabil de reprezentare a valorilor binare. La fel cum zecimal este un sistem de numărare în baza zece și binar în baza doi, hexazecimal este un sistem de numerație în baza 16.

Sistemul de numerație în baza 16 folosește numerele de la 0 la 9 și literele de la A la F. Fig. 1 arată valorile echivalente în zecimal, binar și hexazecimal. Există 16 combinații unice de patru biți de la 0000 la 1111. Hexazecimal este sistemul de numerație perfect de folosit deoarece orice patru biți pot fi reprezentați cu o singură valoare hexazecimală.

Înțelegerea Bytes – Fiind dat faptul că 8 biți reprezintă o grupare binară comună, cifrele binare de la 00000000 la 11111111 pot fi reprezentate în hexazecimal cu valori de la 00 la FF. Zerourile de început pot fi afișate pentru a completa reprezentarea pe 8 biți. De exemplu, valoarea binară 0000 1010 este afișată în hexazecimal ca 0A.

Notă: Este importantă distingerea valorilor hexazecimale de valorile zecimale în ceea ce privește caracterele de la 0 la 9.

Reprezentarea Valorilor Hexadecimale – Hexadecimal este de obicei reprezentat în text de valoarea precedată de 0x (de exemplu 0x73) sau cu indice 16. Mai puțin comun, poate fi urmată de un H, de exemplu 73H. Însă, deoarece textul indice nu este recunoscut în linia de comandă sau în mediile de programare, reprezentarea tehnică a hexazecimal este cu 0x. Prin urmare, exemplele de mai sus vor fi 0xA și respectiv 0x73.

Conversiile în Hexadecimal – Conversiile numerice dintre zecimal și hexazecimal sunt simple, însă divizarea și multiplicarea cu 16 nu este întotdeauna convenabilă.

Cu multă practică, este posibilă recunoașterea tiparelor de biți binare ce corespund valorilor zecimale și hexazecimale. Fig.8.32, arată aceste tipare pentru valorile de 8 biți selectate.

Representing Hexadecimal Values			Fig.8.32. Tipare pe 8 BIȚI			Hexadecimal Conversions of Binary Octets		
Hexadecimal	Decimal	Binary	Hexadecimal	Decimal	Binary	Hexadecimal	Decimal	Binary
0	0	0000	00	0	0000 0000	00	0	0000 0000
1	1	0001	01	1	0000 0001	01	1	0000 0010
2	2	0010	02	2	0000 0011	03	3	0000 0100
3	3	0011	04	4	0000 0101	05	5	0000 0110
4	4	0100	06	6	0000 0111	07	7	0000 1000
5	5	0101	08	8	0000 1001	0A	10	0000 1010
6	6	0110	0F	15	0000 1111	10	16	0001 0000
7	7	0111	20	32	0010 0000	40	64	0100 0000
8	8	1000	80	128	1000 0000	C0	192	1100 0000
9	9	1001	CA	202	1100 1010	F0	240	1111 0000
A	10	1010	FF	255	1111 1111			
B	11	1011						
C	12	1100						
D	13	1101						
E	14	1110						
F	15	1111						

Adresele IPv6 au o lungime de 128 de biți și sunt scrise ca un string de valori hexazecimale. Fiecare 4 biți sunt reprezentăți de o singură valoare hexazecimală; rezultă un total de 32 de valori hexazecimale. Adresele IPv6 nu sunt case sensitive și pot fi scrise și cu litere mici și cu litere mari.

Format Preferat – Așa cum este arătat în Fig.8.33, formatul preferat pentru scrierea unei adrese IPv6 este $x:x:x:x:x:x$, cu fiecare “ x ” constând din patru valori hexazecimale. Atunci când ne referim la 8 biți dintr-o adresă IPv4 folosim termenul de octet. În IPv6, un hextet este termenul neoficial folosit pentru referirea la un segment de 16 biți sau 4 valori hexazecimale. Fiecare “ x ” este un hextet, 16 biți sau 4 valori hexazecimale.

Formatul preferat înseamnă faptul că adresa IPv6 este scrisă folosind toate cele 32 de valori hexazecimale. Acest lucru nu înseamnă că este metoda ideală de reprezentare a adresei IPv6. În următoarele pagini, vom vedea două reguli de ajutor în reducerea numărului de valori necesare pentru reprezentarea unei adrese IPv6.

Fig. 8.34, are exemple de adrese IPv6 în format preferat.

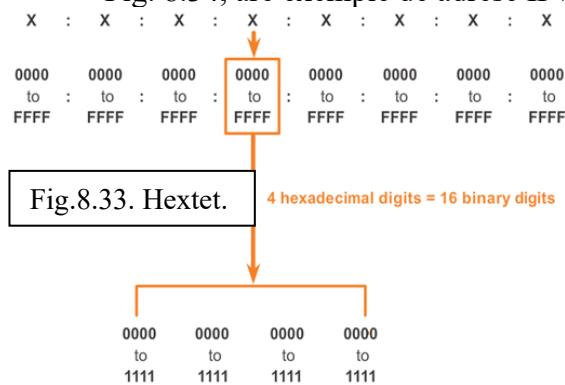


Fig.8.33. Hextet.

Fig.8.34. Ex. Format Preferat.

2001 : 0DB8 : 0000 : 1111 : 0000 : 0000 : 0000 : 0200
2001 : 0DB8 : 0000 : 00A3 : ABCD : 0000 : 0000 : 1234
2001 : 0DB8 : 000A : 0001 : 0000 : 0000 : 0000 : 0100
2001 : 0DB8 : AAAA : 0001 : 0000 : 0000 : 0000 : 0200
FE80 : 0000 : 0000 : 0000 : 0123 : 4567 : 89AB : CDEF
FE80 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001
FF02 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001
FF02 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001 : FF00 : 0200
0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001
0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000

Prima regula de reducere a notăției adreselor IPv6 este că orice 0 din față din fiecare secțiune de 16 biți sau hextet să fie omis. De exemplu:

- 01AB poate fi reprezentat ca 1AB.
- 09F0 poate fi reprezentat ca 9F0.
- 0A00 poate fi reprezentat ca A00.
- 00AB poate fi reprezentat ca AB.

Această regulă se aplică numai zerourilor de început, altfel adresa va fi ambiguă. De exemplu, hextetul “ABC” poate fi fie “0ABC”, fie “ABC0”.

Fig. 8.35 A-D arată mai multe exemple a modului în care omiterea zerourilor de început poate fi folosită pentru a reduce dimensiunea unei adrese IPv6. Pentru fiecare exemplu, este evidențiat formatul preferat. Vedem cum omiterea de zerouri din exemple rezultă într-o reprezentare mai mică a adresei.

Preferred	2001:0DB8:0000:1111:0000:0000:0000:0200
No leading 0s	2001: DB8: 0:1111: 0: 0: 0: 200

Fig. 8.35. A. Omiterea 0.

Preferred	2001:0DB8:0000:A300:ABCD:0000:0000:1234
No leading 0s	2001: DB8: 0:A300:ABCD: 0: 0:1234

Fig. 8.35. B. Omiterea 0.

Preferred	2001:0DB8:000A:1000:0000:0000:0000:0100
No leading 0s	2001: DB8: A:1000: 0: 0: 0: 100

Preferred	FE80:0000:0000:0000:0123:4567:89AB:CDEF
No leading 0s	FE80: 0: 0: 0: 123:4567:89AB:CDEF

Preferred	FF02:0000:0000:0000:0000:0000:0000:0001
No leading 0s	FF02: 0: 0: 0: 0: 0: 0: 0: 1

Fig. 8.35. C. Omiterea 0.

Preferred	FF02:0000:0000:0000:0000:0001:FF00:0200
No leading 0s	FF02: 0: 0: 0: 0: 0: 1:FF00: 200

Preferred	0000:0000:0000:0000:0000:0000:0000:0001
No leading 0s	0: 0: 0: 0: 0: 0: 0: 0: 1

Fig. 8.35. D. Omiterea 0.

Preferred	0000:0000:0000:0000:0000:0000:0000:0000
No leading 0s	0: 0: 0: 0: 0: 0: 0: 0: 0

A doua regulă de reducere a notăției adreselor IPv6 este aceea că simbolul “::” poate înlocui un singur șir continuu de unu sau mai multe segmente de 16 biți ce conțin toți biții de 0.

Simbolul “::” poate fi folosit o dată într-o adresă, altfel va fi mai mult decât o singură adresă ca rezultat. Atunci când este folosită cu tehnica de omitere a zerourilor de început, notăția adresei IPv6 poate fi redusă foarte mult. Acest lucru se numește format compresat.

Adresa incorrectă : 2001:0DB8::ABCD::1234.

Possible extinderi de adrese comprimate ambiguë:

- 2001:0DB8::ABCD:0000:0000:1234.
- 2001:0DB8::ABCD:0000:0000:0000:1234.
- 2001:0DB8:0000:ABCD::1234.
- 2001:0DB8:0000:0000:ABCD::1234.

Fig.8.36. A-D arată mai multe exemple a modului în care utilizarea simbolului “::” și a tehnicii de omitere a zerourilor de început pot reduce dimensiunea unei adrese IPv6.

Preferred	2001:0DB8:0000:1111:0000:0000:0000:0200
No leading 0s	2001: DB8: 0:1111: 0: 0: 0: 0: 200
Compressed	2001:DB8:0:1111::200



Fig.8.36.A. Utilizare două puncte duble

Preferred	FE80:0000:0000:0000:0123:4567:89AB:CDEF
No leading 0s	FE80: 0: 0: 0: 123:4567:89AB:CDEF
Compressed	FE80::123:4567:89AB:CDEF

Preferred	FF02:0000:0000:0000:0000:0000:0000:0001
No leading 0s	FF02: 0: 0: 0: 0: 0: 0: 0: 1
Compressed	FF02::1

Fig.8.36.B. Utilizare două puncte duble

Preferred	FF02:0000:0000:0000:0000:0001:FF00:0200
No leading 0s	FF02: 0: 0: 0: 0: 1:FF00: 200
Compressed	FF02::1:FF00:200

Preferred	0000:0000:0000:0000:0000:0000:0000:0001
No leading 0s	0: 0: 0: 0: 0: 0: 0: 0: 1
Compressed	::1

Preferred	0000:0000:0000:0000:0000:0000:0000:0000
No leading 0s	0: 0: 0: 0: 0: 0: 0: 0: 0
Compressed	::

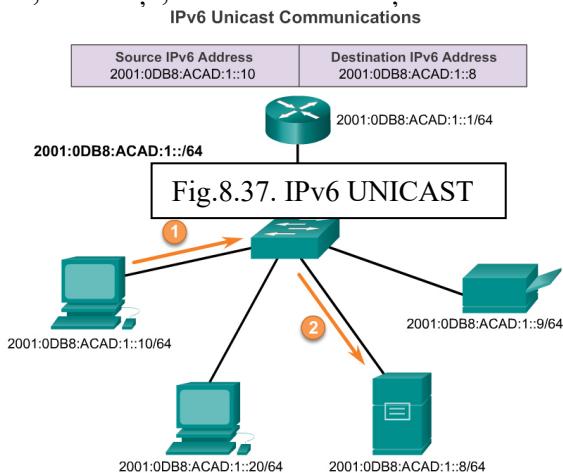
Fig.8.36.D. Utilizare două puncte duble

8.14.4 Tipuri de adrese IPv6

Există trei tipuri de adrese IPv6:

- **Unicast** – O adresă unicast IPv6 identifică unic o interfață de pe dispozitivul ce permite IPv6. Așa cum se observă și în Fig. , o adresă IPv6 sursă trebuie să fie o adresă unicast.
- **Multicast** – O adresă multicast IPv6 este folosită pentru a trimite un singur pachet IPv6 la mai multe destinații.
- **Anycast** - O adresă anycast IPv6 este orice adresă unicast IPv6 ce poate fi atribuită la mai multe dispozitive. Un pachet trimis la o adresă anycast este rutat la dispozitivul cel mai apropiat ce are respectiva adresă. Adresele anycast nu intră în scopul acestui capitol.

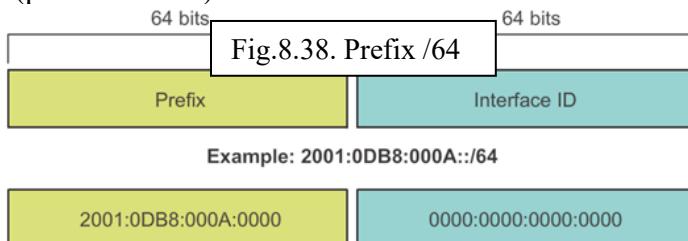
Spre deosebire de IPv4, IPv6 nu are o adresă de broadcast. Însă, există o adresă multicast IPv6 pentru toate nodurile ce, în esență, conduce la același rezultat.



Reamintim faptul că prefixul, sau partea de rețea, al unei adrese IPv4 poate fi identificat prin masca de rețea în zecimal sau lungimea prefixului (notăția cu slash). De exemplu, o adresă IP 192.168.1.10 cu masca de rețea în zecimal 255.255.255.0 este echivalentă cu 192.168.1.10/24.

IPv6 folosește lungimea prefixului pentru a reprezenta partea de prefix a adresei. IPv6 nu folosește notație zecimală punctată a măștii de rețea. Lungimea de prefix este folosită pentru a indica partea de rețea a unei adrese IPv6 folosind IPv6 adresa/lungimea prefixului.

Lungimea prefixului poate varia de la 0 la 128. O lungime tipică de prefix pentru IPv6 pentru LANuri și pentru cele mai multe tipuri de rețele este /64. Acest lucru înseamnă că prefixul sau partea de rețea a adresei are 64 de biți în lungime, iar ceilalți 64 de biți sunt folosiți pentru ID-ul adresei interfeței (partea de host).



O adresă unicast IPv6 identifică unic o interfață de pe un dispozitiv ce permite IPv6. Un pachet trimis la o adresă unicast este primit de interfața ce are atribuită respectiva adresă. Similar cu IPv4, o adresă sursă IPv6 trebuie să fie o adresă unicast. Adresa destinație IPv6 poate fi fie o adresă unicast, fie o adresă multicast.

Exista șase tipuri de adrese unicast IPv6:

1) **Global unicast** – O adresă unicast globală este similară cu o adresa IPv4 publică. Acestea sunt adrese rutabile în Internet, unice global. Adresele unicast globale pot fi configurate static sau atribuite dinamic. Există unele diferențe importante în modul în care un dispozitiv primește adresa sa IPv6 dinamic, spre deosebire de DHCP pentru IPv4.

2) **Link-local** – Adresele de legătură locală sunt folosite pentru comunicarea cu alte dispozitive din aceeași legătură locală. Cu IPv6, termenul de legătură se referă la subrețea. Adresele de legătură locală sunt limitate la o singură legătură. Unicitatea lor trebuie să fie confirmată numai pe respectiva legătură deoarece ele nu sunt rutabile în afara acestei legături. Cu alte cuvinte, routerele nu vor transmite pachetele mai departe ce au o adresă destinație sau sursă de legătură locală.

3) **Loopback** – Adresa de loopback este folosită de un host pentru a-și trimite un pachet și nu poate fi atribuită pe o interfață fizică. Similar cu o adresă loopback IPv4, putem da **ping** la o adresă loopback IPv6 pentru a testa configurația TCP/IP de pe hostul local. Adresa loopback IPv6 este alcătuită numai din zerouri cu excepția ultimului bit, reprezentat ca ::1/128 sau ::1 în formatul comprimat.

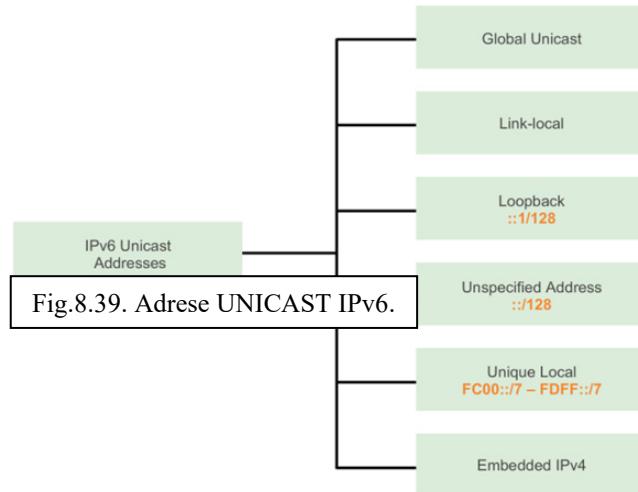
4) **Unspecified address** – O adresă nespecificată este o adresă formată numai din zerouri reprezentată în formatul comprimat ca ::/128 sau numai "::". Nu poate fi atribuită unei interfețe și este folosită numai ca o adresă sursă a unui pachet IPv6. O adresă nespecificată este folosită ca o adresă sursă atunci când dispozitivul nu are încă o adresă IPv6 permanentă sau atunci când sursa unui pachet este irelevantă pentru destinație.

5) **Unique local** – Adresele IPv6 unice locale au unele similarități cu adresele private RFC 1918 pentru IPv4, însă există și diferențe semnificative. Adresele IPv6 unice locale sunt folosite pentru adresarea locală dintr-un loc sau dintr-un număr limitat de locuri. Aceste adrese nu ar trebui să fie rutabile în global IPv6. Adresele IPv6 unice locale se găsesc în spațiul de adrese de la FC00::/7 la FDFF::/7.

În IPv4, adresele private sunt combinate cu NAT/PAT pentru a oferi o traducere "mai mulți la unul" a adreselor private la publice. Acest lucru se realizează datorită disponibilității limitate a spațiului de adresă IPv4. Mai multe locuri folosesc, de asemenea, natura privată a adreselor RFC 1918 în ajutorul securizării și ascunderii rețelei lor de potențialele riscuri de securitate. Însă, acest lucru nu a reprezentat niciodată scopul acestor tehnologii și IEFT întotdeauna a recomandat ca acestea să ia măsuri de precauție de securitate adecvate pe routerul ce "întâmpină" Internetul. Deși IPv6 oferă această adresare specifică, nu are scopul de a fi utilizat pentru ascunderea dispozitivelor interne de Internetul IPv6. IEFT recomandă ca accesul să fie limitat pe dispozitive cu ajutorul unor măsuri de securitate adecvate, de bună practică.

Notă: Specificația originală IPv6 definește adresele site-local pentru un scop similar, folosind spațiul de prefix FEC0::/10. Au existat mai multe ambiguități în specificație și adresele site-local au fost depreciate de către IEFT în favoarea adreselor unice local.

Adresele IPv4 integrate – Ultimul tip de adresă unicast este adresa încorporată IPv4. Aceste adrese sunt folosite pentru ajutarea tranziției de la IPv4 la IPv6. Adresele încorporate în IPv4 nu reprezintă scopul acestui curs.



O adresă de legătură locală IPv6 permite unui dispozitiv să comunice cu alte dispozitive de pe aceeași legătură și numai în respectiva legătură (subrețea). Pachetele cu o sursă sau destinație de legătură locală nu pot fi rutate în afara legăturii de unde provin.

Spre deosebire de adreselor de legătură locală IPv4, adrese de legătură locală IPv6 au un rol important în diferite aspecte ale rețelei. Adresa unicast globală nu este o necesitate; însă, orice interfață de rețea trebuie să aibă o adresă de legătură locală.

Dacă o adresă de legătură locală nu este configată manual pe o interfață, dispozitivul va crea automat propria adresă fără comunicarea cu un server DHCP. Hosturile ce permit IPv6 crează o adresă de legătură locală IPv6 chiar dacă dispozitivul nu a fost asignat cu o adresă IPv6 unic平 globală. Acest lucru permite ca dispozitivele IPv6 să comunice cu alte dispozitive IPv6 din aceeași subrețea. Acest lucru include comunicare cu default gateway (router).

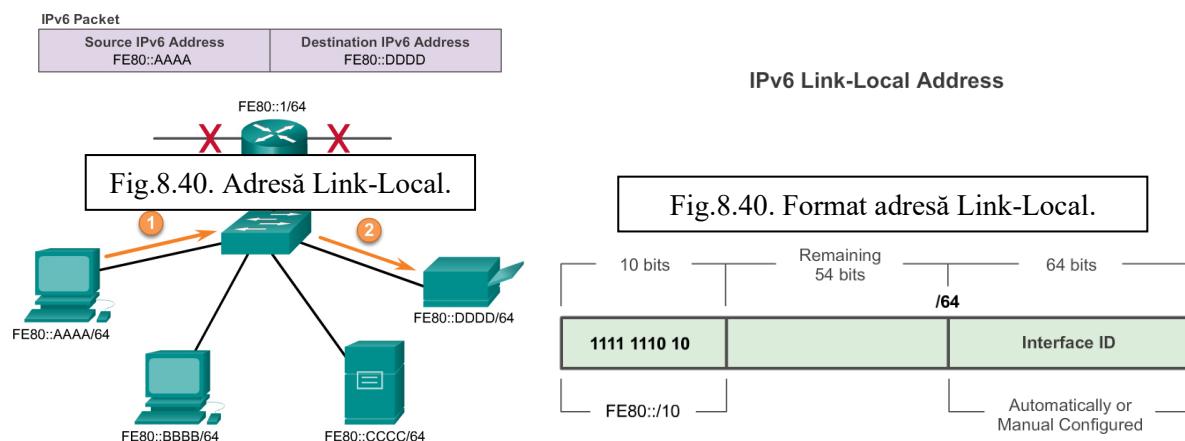
Adresele de legătură locală IPv6 sunt în spațiul FE80::/10. /10 ce indică faptul că primii 10 biți sunt 1111 1110 10xx xxxx. Primul hextet are spațiul de la 1111 1110 1000 0000 (FE80) la 1111 1110 1011 1111 (FEBF).

Fig.8.40 arată un exemplu de comunicare prin utilizarea adreselor de legătură locală IPv6. Fig.8.41 arată formatul unei adrese de legătură locală IPv6.

Adresele de legătură locală IPv6 sunt folosite și de protocoalele de rutare IPv6 în schimbul de mesaje ca adrese next-hop din tabela de rutare IPv6. Adresele de legătură locală IPv6 sunt discutate mai în detaliu în acest curs.

Notă: În mod normal, adresa de legătură locală a routerului și nu cea unicast globală este folosită ca default gateway pentru alte dispozitive din legătură (subrețea).

IPv6 Link-Local Communications



8.14.4.1 Adresele IPv6 Unicast

Adresele unicast globale IPv6 sunt unice global și rutabile pe Internet IPv6. Aceste adrese sunt echivalente adreselor publice IPv4. The Internet Committee for Assigned Names and Numbers (ICANN), operatorul pentru Internet Assigned Numbers Authority (IANA), alocă blocurile de adresă IPv6 către cinci RIRs. Actual, numai adresele unicast globale cu primii trei biți de 001 sau 2000::/3 au fost atribuite. Reprezintă numai 1/8 din totalul spațiului de adrese IPv6 disponibile, excludând doar o foarte mică parte de alte tipuri de adrese unicast și multicast.

Notă: Adresa 2001:0DB8::/32 a fost rezervată pentru scopuri de documentare, inclusiv pentru utilizarea în exemple.

Fig.8.42 arată structura și spațiul unei adrese unicast globale.

O adresă unicast globală are trei părți:

- *Prefix de rutare global.*
- *ID-ul de subrețea.*
- *ID-ul de interfață.*

8.14.4.2 Prefixul Global de Rutare

Prefixul de rutare global este prefixul, sau rețeaua, partea de adresă ce este atribuită de către furnizor, cum ar fi un ISP, clientului sau unei locații. Actual, RIRs a atribuit un prefix de rutare global /48 clienților. Acesta include pe oricine de la rețelele de întreprindere la gospodării individuale. Există mai mult decât necesar spațiu de adrese pentru alți clienți.

Fig.8.43 arată structura unei adrese unicast globale folosind un prefix de rutare global /48. Prefixele /48 sunt cele mai cunoscute prefixe de rutare globale și vor fi discutate mai mult pe exemplele din acest curs.

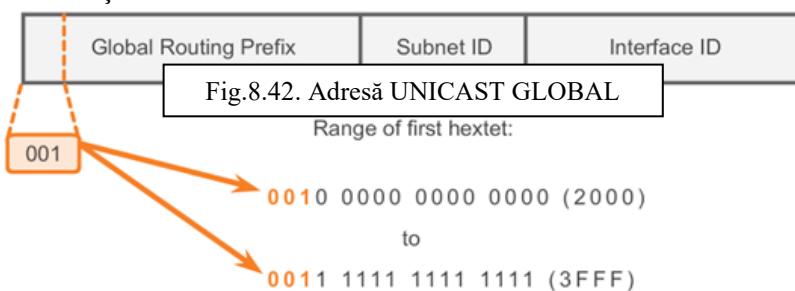
De exemplu, adresa IPv6 2001:0DB8:ACAD::/48 are un prefix ce indică faptul că primii 48 de biți (3 hexteți) (2001:0DB8:ACAD) este prefixul sau partea de rețea a adresei. “::” înseamnă că restul adresei conține zerouri.

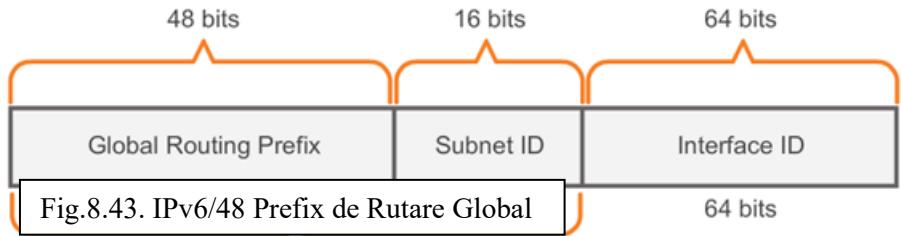
IDul Subrețelei - ID-ul de subrețea este folosit de o organizație pentru identificare subrețelelor din cadrul locației.

IDul Interfeței – Id-ul de interfață IPv6 este echivalent cu partea de host a unei adrese IPv4. Termenul de ID de interfață este folosit deoarece un singur host poate avea mai multe interfețe, fiecare având una sau mai multe adrese IPv6.

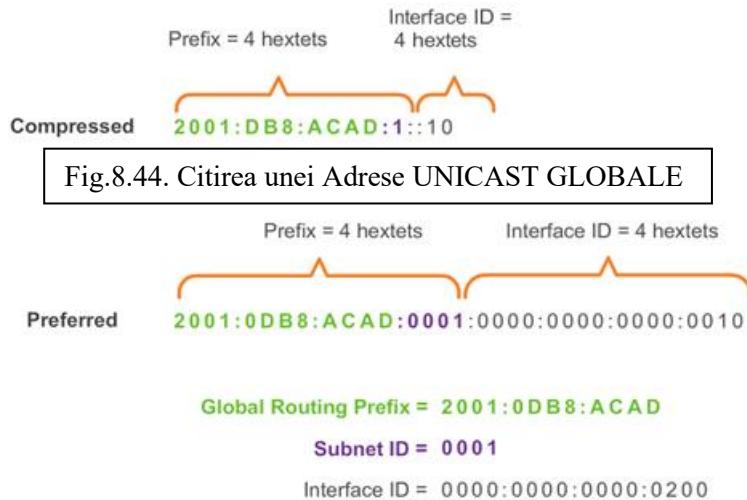
Notă: Spre deosebire de IPv4, în IPv6, adresa cu toți biții de 0 poate fi atribuită unui dispozitiv deoarece nu există adrese de broadcast în IPv6. Însă, adresa cu toți biții de 0 este rezervată ca adresă anycast Subnet-Router și ar trebui să fie atribuită numai routerelor.

O modalitate simplă de citire a celor mai multe adrese IPv6 este numărarea numărului de hexteți. Așa cum este arătat și în Fig. 3, într-o adresă unicast globală /64 primii patru hexteți sunt partea de rețea a adresei, cu patru hexteți indicând IDul subrețelei. Ceilalți patru hexteți reprezintă IDul de interfață.





A /48 routing prefix + 16 bit Subnet ID = /64 prefix.



8.15 ConFig.rea Routerului

Cele mai multe comenzi de conFig.re și verificare din IPv6 din IOS sunt similare cu cele din IPv4. În multe cazuri, singura diferență este utilizarea **ipv6** în locul **ip** din interiorul comenziilor.

Comanda **interface** de conFig.re a unei adrese unicast globale IPv6 pe o interfață este **ipv6 address ipv6-address/prefix-length**.

De reținut faptul că nu există spațiu dintre *ipv6-address* și *prefix-length*.

Exemplul de ConFig.re va folosi topologia din Fig. 1 și următoarele subrețele IPv6:

- 2001:0DB8:ACAD:0001:/64 (or 2001:DB8:ACAD:1::/64).
- 2001:0DB8:ACAD:0002:/64 (or 2001:DB8:ACAD:2::/64).
- 2001:0DB8:ACAD:0003:/64 (or 2001:DB8:ACAD:3::/64).

Așa cum se observă și în Fig. 2, comenziile necesare pentru conFig.rea unei adreselor unicast globale IPv6 pe o interfață GigabitEthernet 0/0 de pe R1 sunt:

Router(config)#interface GigabitEthernet 0/0

Router(config-if)#description Legatura la Rețea 2001:db8:acad:1::0/64

Router(config-if)#ipv6 address 2001:db8:acad:1::1/64

Router(config-if)#no shutdown

8.15 ConFig.rea Hostului

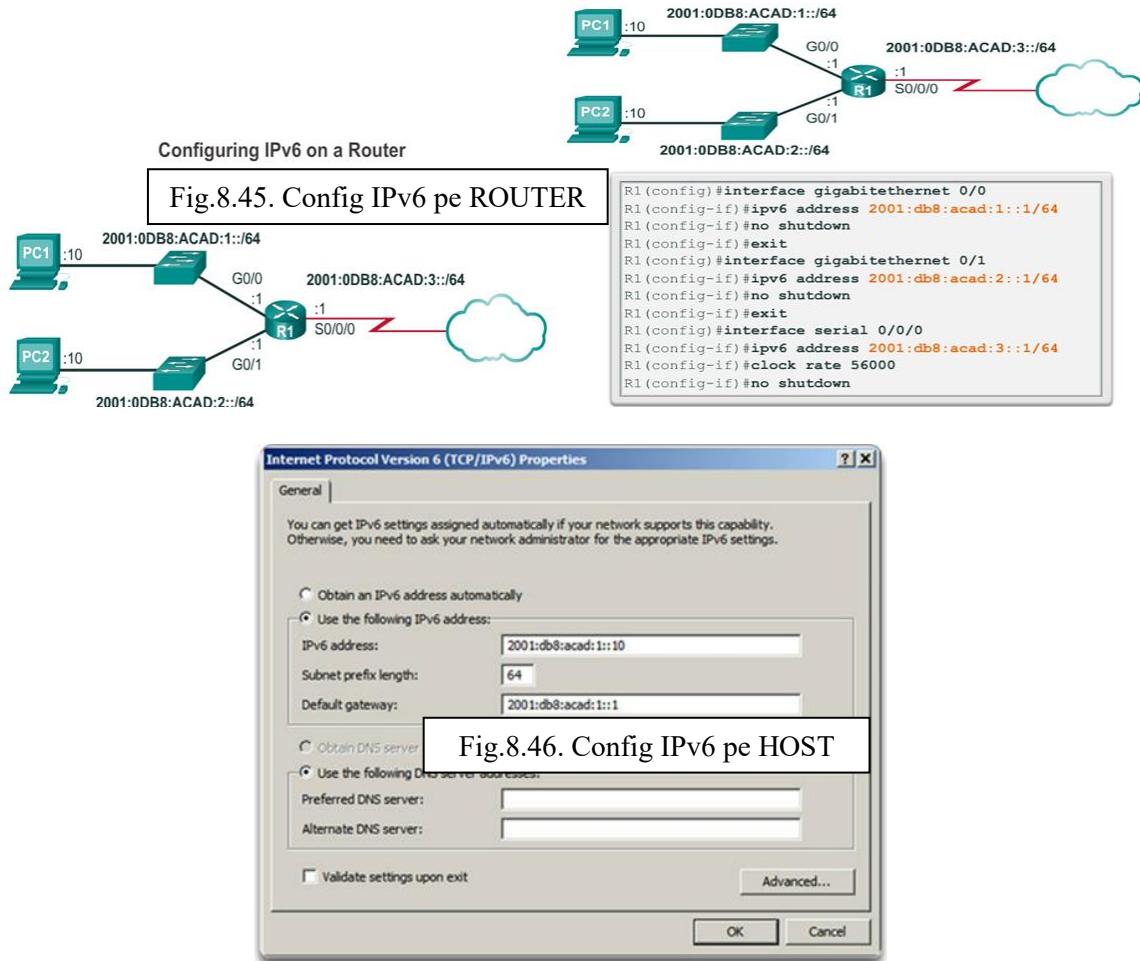
ConFig.rea manuală a unei adrese IPv6 pe un host este similară cu cea din IPv4.

Așa cum se observă și în Fig. 3, adresa de default gateway conFig.tă pentru PC1 este 2001:DB8:ACAD:1::1, adresa unicast globală a interfeței R1 GigabitEthernet este din aceeași rețea.

Ca și în IPv4, configurația adreselor în mod static pe clienți nu scalează bine în medii mari. Din acest motiv, mulți administratori de rețea dintr-o rețea IPv6 vor activa atribuirea dinamică a adreselor IPv6.

Există două moduri în care un dispozitiv poate obține o adresa unică globală IPv6 în mod automat:

- Stateless Address AutoConfiguration (SLAAC).
- DHCPv6.



8.15.1 Stateless Address AutoConfiguration (SLAAC)

Stateless Address AutoConfiguration (SLAAC) este o metodă ce permite unui dispozitiv să-și obțină prefixul, lungimea prefixului și informații de adresă de default gateway de la un router IPv6 fără utilizarea unui server DHCP. Folosind SLAAC, dispozitivele se bazează pe mesajele ICMPv6 Router Advertisement (RA) ale routerului local pentru a obține informațiile necesare.

Routerele IPv6 trimit periodic mesaje ICMPv6 Router Advertisement (RA) tuturor dispozitivelor activate cu IPv6 din rețea. Implicit, routerele Cisco trimit mesaje RA la fiecare 200 de secunde tuturor adreselor de grup multicast IPv6. Un dispozitiv IPv6 din rețea nu trebuie să aștepte mesajele periodice RA. Un dispozitiv poate trimite un mesaj Router Solicitation (RS) routerului, folosind adresa de grup multicast all-routers IPv6. Atunci când un router IPv6 primește un mesaj RS, va răspunde imediat cu un RA.

Chiar dacă o interfață de pe un router Cisco poate fi config.ă cu o adresă IPv6, nu îl face un router IPv6. Un router IPv6 este un router ce:

- Trimit pachete IPv6 între rețele.
- Poate fi config.ă cu rute statice IPv6 sau cu un protocol dinamic de rutare IPv6.
- Trimit mesaje ICMPv6 RA.

Rutarea IPv6 nu este activată implicit. Pentru a activa un router ca un router IPv6, comanda de config.ă globală **ipv6 unicast-routing** trebuie să fie folosită.

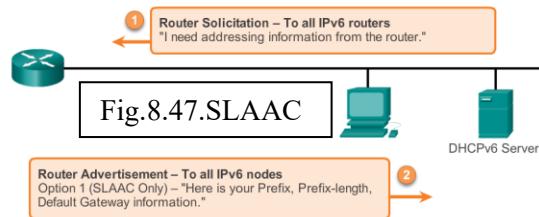
Notă: Routerele Cisco sunt activate ca routere IPv4 implicit.

Mesajul ICMPv6 RA conține prefixul, lungimea prefixului și alte informații pentru dispozitivul IPv6. Mesajul ICMPv6 RA informează de asemenea dispozitivul IPv6 de modul cum să-și obțină informațiile de adresare. Mesajul RA poate conține una dintre următoarele opțiuni, astăzi cum se observă și în Fig. :

- **Opțiunea 1 - SLAAC Only** – Dispozitivul ar trebui să folosească prefixul, lungimea prefixului și informațiile de adresa de default gateway conținute în mesajul RA. Nici-o altă informație nu este disponibilă de la serverul DHCPv6.
- **Opțiunea 2 – SLAAC and DHCPv6** – Dispozitivul ar trebui să folosească prefixul, lungimea prefixului și informațiile de adresa de default gateway în mesajul RA. Nici-o altă informație nu este disponibilă de la serverul DHCPv6, cum ar fi adresa de server DNS. Dispozitivul, prin intermediul procesului normal de descoperire și cerere la serverul DHCPv6, obține această informație suplimentară. Acest lucru este cunoscut ca stateless DHCPv6 deoarece serverul DHCPv6 nu trebuie să aloce sau să țină evidența nici-unei atribuiri de adresă IPv6, însă oferă numai informații suplimentare cum ar fi adresa de server DNS.
- **Opțiunea 3 – DHCPv6 only** – Dispozitivul nu trebuie să folosească informațiile din mesajul RA pentru informațiile de adresare. În schimb, dispozitivul va folosi procesul normal de descoperire și cerere la un server DHCPv6 pentru a obține toate informațiile sale de adresare. Acestea includ o adresă unică globală IPv6, lungimea prefixului, o adresa de default gateway și adresele serverelor DNS. În acest caz, serverul DHCPv6 se comportă ca un server stateful DHCP, similar lui DHCP pentru IPv4. Serverul DHCPv6 alocă și ține evidența adreselor IPv6, deci nu atribuie aceeași adresă IPv6 la mai multe dispozitive.

Routerele trimit mesaje ICMPv6 RA folosind adresa de legătură locală ca adresă sursă IPv6. Dispozitivele folosind SLAAC utilizează adresa de legătură locală a routerului ca adresa lor de default gateway.

Router Solicitation and Router Advertisement Messages



8.15.2 DHCPv6

Dynamic Host Configuration Protocol pentru IPv6 (DHCPv6) este similar DHCPului pentru IPv4. Un dispozitiv poate primi automat informațiile sale de adresare, inclusiv adresa unică globală IPv6, lungimea prefixului, o adresa de default gateway și adresele serverelor DNS folosind serviciile unui server DHCPv6.

Un dispozitiv poate primi toate sau unele informații de adresare IPv6 de la un server DHCPv6, în funcție de ce opțiune este specificată în mesajul ICMPv6 RA (opțiune SLAAC și DHCPv6 sau opțiunea numai DHCPv6). În plus, OS al hostului ar putea alege să ignore orice se află în mesajul RA de la router și să obțină adresa IPv6 proprie și alte informații direct de la un server DHCPv6.

Înainte de implementarea dispozitivelor IPv6 într-o rețea este o ideea bună de a verifica dacă un host observă opțiunile din mesajul ICMPv6 RA al routerului.

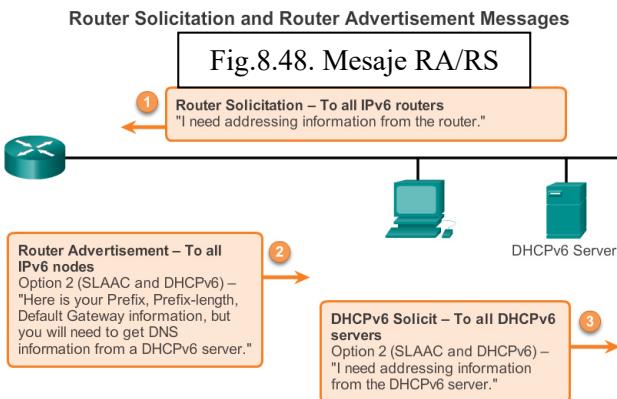
Un dispozitiv își poate obține adresa unică globală IPv6 în mod dinamic și poate fi configurat cu mai multe adrese statice IPv6 pe aceeași interfață. IPv6 permite ca mai multe adrese IPv6, aparținând aceleiași rețele IPv6, să fie configurate pe aceeași interfață.

Un dispozitiv poate fi de asemenea configurat cu una sau mai multe adrese IPv6 de default gateway. Pentru mai multe informații în ceea ce privește decizia luată în ceea ce privește ce adresă este folosită ca o adresă sursă IPv6 sau care adresă default gateway este utilizată, de documentat RFC 6724, Default Address Selection pentru IPv6.

8.15.3 IDul Interfeței

În cazul în care clientul nu folosește informațiile conținute în mesajul RA și se bazează pe DHCPv6, serverul DHCPv6 va oferi întreaga adresă unică globală IPv6, inclusiv prefixul și IDul interfeței.

Însă, dacă opțiunea 1 sau opțiunea 2 este folosită, clientul nu obține partea reală de ID de interfață a adresei din aceste procese. Dispozitivul client trebuie să determine propriul ID de interfață de 64 de biți, fie prin folosirea procesului EUI-64, fie prin generarea unui număr aleator de 64 de biți.



8.15.4 EUI-64

IEEE definește Extended Unique Identifier (EUI) sau procesul modificat EUI-64. Acest proces folosește o adresă Ethernet MAC de 48 de biți a clientului și inserează alti 16 biți în mijlocul adresei MAC de 48 de biți pentru a crea un ID de interfață de 64 de biți.

Adresele Ethernet MAC sunt de obicei reprezentate în hexazecimal și sunt alcătuite din două părți:

- **Organizationally Unique Identifier (OUI)** – OUI este un cod de furnizor de 24 de biți (6 cifre hexazecimale) atribuit de către IEEE.
- **Device Identifier** – Identificatorul dispozitivului este o valoare unică de 24 de biți dintr-un OUI.

Un ID de interfață EUI-64 este reprezentat în binar și este alcătuit din trei părți:

- OUI de 24 de biți de la adresa MAC a clientului, însă al 7-lea bit (the Universally/Locally (U/L) bit) este inversat. Acest lucru înseamnă că dacă al 7-lea bit este un 0 devine 1 și invers.
- Valoarea introdusă FFFE de 16 biți (în hexazecimal).
- Identifierul dispozitivului de 24 de biți de la adresa MAC a clientului.

Procesul EUI-64 este ilustrat în Fig. 1, folosind adresa MAC GigabitEthernet a lui R1: FC99:4775:CEE0.

Pasul 1. Împărțim adresa MAC între OUI și identifierul de dispozitiv.

Pasul 2. Inserăm valoarea hexazecimală FFFE, reprezentată în binar ca 1111 1111 1111 1110.

Pasul 3. Convertim primele două valori hexazecimale ale OUI în binar și inversăm al 7-lea bit. În acest exemplu, 0 de pe bitul 7 este schimbat în 1.

Rezultatul este un ID de interfață generat de EUI-64 : FE99:47FF:FE75:CEE0.

Notă: Utilizarea bitului U/L și motivele de inversare a valorii sale sunt discutate în RFC 5342.

Avantajul EUI-64 este că adresa Ethernet MAC poate fi folosită pentru a determina ID-ul interfeței. Permite de asemenea administratorilor de rețea să urmărească ușor o adresă IPv6 și un dispozitiv ce folosește adresa unică MAC. Însă, acest lucru a ridicat probleme de confidențialitate în rândul multor utilizatori. Sunt îngrijorați de faptul că pachetele lor pot fi urmărite la computerul fizic. Având în vedere aceste îngrijorări, poate fi folosit în schimb un ID de interfață generat aleator.

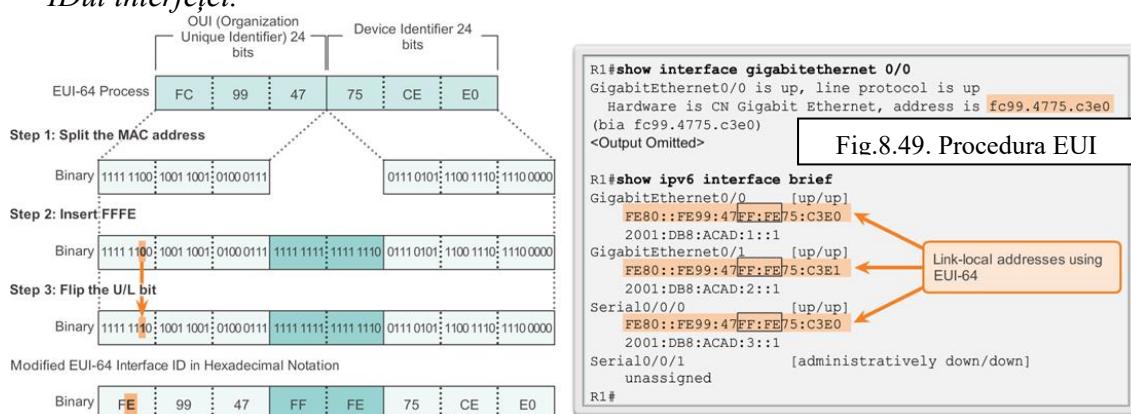
8.15.5 ID de Interfață Generat Aleator

În funcție de sistemul de operare, un dispozitiv poate utiliza un ID de interfață generat aleator în schimbul adresei MAC și a procesului EUI-64. De exemplu, începând cu Windows Vista, Windows folosește un ID de interfață generat aleator în schimbul unuia creat cu EUI-64. Windows XP și sistemele de operare Windows de dinaintea lui, foloseau EUI-64.

Un mod simplu de identificare a faptului că o adresă a fost cel mai probabil creată cu EUI-64, FFFE este localizat în mijlocul ID de interfață, așa cum se observă și în Fig. 2.

După ce ID-ul interfeței este stabilit, fie prin procesul EUI-64, fie prin generarea aleatoare, poate fi combinat cu un prefix IPv6 pentru a crea o adresă unică globală sau o adresă de legătură locală:

- **Adresa unică globală** – Atunci când folosim SLAAC, dispozitivul își primește prefixul de la ICMPv6 RA și îl combină cu ID-ul de interfață.
- **Adresa de legătură locală** – Un prefix de legătură locală începe cu FE80::/10. Un dispozitiv folosește în mod normal FE80::/64 ca prefix/lungime de prefix, urmat de către ID-ul interfeței.



Atunci când folosește SLAAC (numai SLAAC sau SLAAC cu DHCPV6), un dispozitiv își primește prefixul și lungimea prefixului de la ICMPv6 RA. Deoarece prefixul adresei a fost atribuit de către mesajul RA, dispozitivul trebuie să ofere numai partea de ID de interfață a adresei sale. Așa cum spuneam mai devreme, IDul de interfață poate fi generat automat prin procesul EUI-64 sau, în funcție de sistemul de operare, generat aleator. Utilizând informațiile din mesajul RA și IDul de interfață, dispozitivul poate stabili adresa sa unicast globală.

După ce o interfață unicast globală este atribuită unei interfețe, dispozitivul IPv6 va genera automat adresa sa de legătură locală. Dispozitivele activate IPv6 trebuie să aibă, cel puțin, adresa de legătură locală. Reamintim faptul că adresa de legătură locală IPv6 permite unui dispozitiv să comunice cu alte dispozitive activate IPv6 din aceeași subrețea.

Adresele de legătură locală sunt folosite pentru mai multe scopuri, cum ar fi:

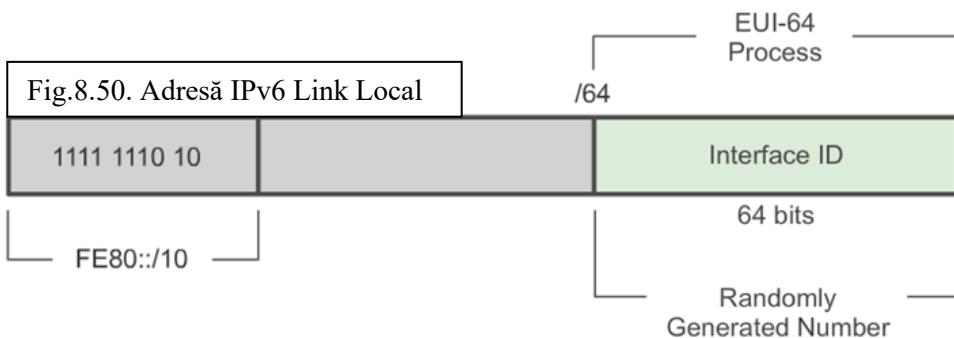
- *Un host folosește adresa de legătură locală a routerului local pentru adresa sa IPv6 de default gateway.*
- *Routerele schimbă mesaje de protocol de rutare dinamic folosind adresele de legătură locală.*
- *Tabelele de rutare ale routerelor folosesc adresa de legătură locală pentru a identifica routerul next-hop atunci când trimit pachete IPv6.*

O adresă de legătură locală poate fi stabilită dinamic sau configată manual ca o adresă statică de legătură locală.

8.15.6 Asignarea Adresei de Link-Local Dinamic

Adresa de legătură locală este creată dinamic folosind prefixul FE80::/10 și IDul de interfață.

Implicit, IOSul routerelor folosesc EUI-64 pentru a genera IDul de interfață pentru toate adresele de legătură locală de pe interfețele IPv6. Pentru interfețele seriale, routerul folosește adresa MAC a unei interfețe Ethernet. Reamintim faptul că o adresă de legătură locală trebuie să fie unică numai pe legătură respectivă sau rețea. Însă, un pas înapoi în utilizarea atribuirii dinamice a adresei de legătură locală este lungimea sa, ce face dificilă identificare și ținerea evidenței adreselor atribuite.



8.15.7 Static Link-Local Address

Configarea adresei de legătură locală manual oferă abilitatea de creare a unei adrese recunoscute și ușor de ținut minte.

Adresele de legătură locală pot fi configurate manual prin folosirea acelorași comenzi de interfață folosită pentru crearea adreselor unicast globale IPv6, dar cu un parametru suplimentar: **Router(config-if)#ipv6 address link-local-address link-local**

Fig.8.50 arată faptul că o adresă de legătură locală are un prefix din spațiul de adrese de la FE80 la FEBF. Atunci când o adresă începe cu acest hextet, parametrul de legătură locală trebuie să urmeze adresa.

Fig.8.51 arată configurația unei adrese de legătură locală prin folosirea comenzi **ipv6 address interface**. Adresa de legătură locală FE80::1 este folosită pentru a face mai ușor de recunoscut faptul că aparține routerului R1. Aceeași adresă IPv6 de legătură locală este configurată pe toate interfețele lui R1. FE80::1 poate fi configurată pe fiecare legătură deoarece este unică pe respectiva legătură.

Similar cu R1, routerul R2 va fi configurat cu FE80::2 ca adresa IPv6 de legătură locală pe toate interfețele sale.

```

R1(config)#interface gigabitethernet 0/0
R1(config-if)#ipv6 address fe80::1 ?
  link-local  Use link-local address

R1(config-if)#ipv6 address fe80::1 link-local
R1(config-if)#exit
R1(config)#interface gigabitethernet 0/1
R1(config-if)#ipv6 address fe80::1 link-local
R1(config-if)#exit
R1(config)#interface serial 0/0/0
R1(config-if)#ipv6 address fe80::1 link-local
R1(config-if)#

```

R1#show ipv6 interface brief	
GigabitEthernet0/0	[up/up]
FE80::1	
2001:DB8:ACAD:1::1	
GigabitEthernet0/1	[up/up]
FE80::1	
2001:DB8:ACAD:2::1	
Serial0/0/0	[up/up]
FE80::1	
2001:DB8:ACAD:3::1	
Serial0/0/1	[administratively down/down]
unassigned	

R1#

Statically configured link-local addresses

Fig.8.51. Config IPv6 Link Local

Așa cum se observă în Fig.8.51, comanda de verificare a configurației de interfață IPv6 este similară cu comanda folosită în IPv4.

Comanda **show interface** afișează adresa MAC a interfețelor Ethernet. EUI-64 folosește adresa MAC pentru a genera ID-ul de interfață pentru adresele de legătură locală. În plus, comanda **show ipv6 interface brief** afișează detalii pentru fiecare interfață. Afisajul [up/up] din aceeași linie indică starea interfeței de nivel 1/2. Acest lucru este la fel cu Status și Protocol din comanda IPv4.

De remarcat faptul că fiecare interfață are două adrese IPv6. A doua adresă pentru fiecare interfață este adresa unică globală configurată. Prima adresă, cea care începe cu FE80, este adresa unică de legătură locală pentru interfață. Reamintim faptul că adresa unică de legătură locală este adăugată automat interfeței atunci când este atribuită o adresă unică globală.

De asemenea, este de remarcat faptul că adresa de legătură locală de pe Serial 0/0/0 a lui R1 este aceeași cu cea de pe interfață GigabitEthernet 0/0. Interfețele seriale nu au adresa MAC și, prin urmare, IOSul folosește adresa MAC a primei interfețe Ethernet disponibile. Acest lucru este posibil deoarece interfețele de legătură locală trebuie să fie unice doar pe respectiva legătură.

Adresa de legătură locală de pe interfață routerului este de obicei adresa default gateway pentru dispozitivele din respectiva legătură sau rețea.

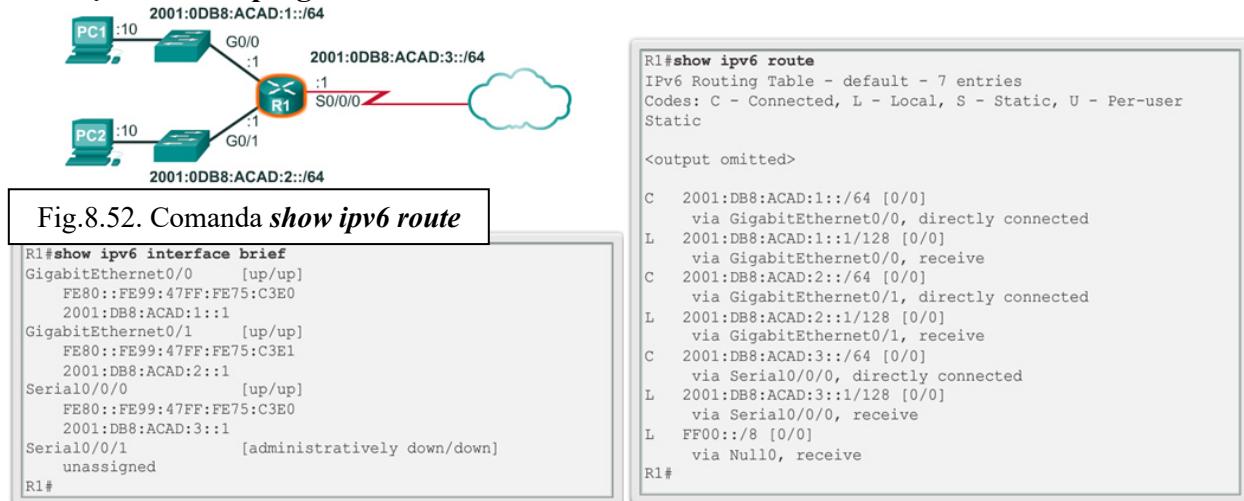
Așa cum se poate observa în Fig.8.52, comanda **show ipv6 route** poate fi folosită pentru a verifica faptul că rețelele IPv6 și adresele de interfețe specifice au fost instalate în tabela de rutare IPv6. Comanda **show ipv6 route** va afișa numai rețelele IPv6, nu și pe cele IPv4.

În tabela de rutare, un C în dreptul unei rute indică faptul că respectiva rută este o rețea direct conectată. Atunci când o interfață de router este configurată cu o adresă unică globală și este în starea "up/up", prefixul și lungimea prefixului IPv6 sunt adăugate în tabela de rutare IPv6 ca rute conectate.

Adresa unică globală IPv6 configurată pe interfață este de asemenea instalată în tabela de rutare ca rută locală. Ruta locală are un prefix /128. Rutele locale sunt folosite de tabela de rutare pentru procesarea eficientă a pachetelor cu o adresă destinație a adresei de interfață a routerului.

Comanda **ping** pentru IPv6 este identică ca cea pentru IPv4, cu excepția faptului că o adresă IPv6 este folosită. Așa cum se observă în Fig.8.53, comanda este folosită pentru verificare conectivității de nivel 3 dintre R1 și PC1. Atunci când de pe un router se dă **ping** către o adresă

de legătură locală, IOSul va cere utilizatorului interfața de ieșire. Deoarece adresa destinație de legătură locală poate fi pe una sau mai multe legături sau rețele, routerul are nevoie să știe pe care interfață să trimită ping.



```
R1#ping 2001:db8:acad:1::10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:1::10, timeout
is 2 seconds:
!!!!!
Success rate is 100 percent (5/5)
R1#
```

Fig.8.53. Comanda *ping*

8.15.8 Adresele IPv6 Multicast

Adresele multicast IPv6 sunt similare cu adresele multicast IPv4. Reamintim faptul că o adresă multicast este folosită pentru a trimite un singur pachet la una sau mai multe destinații (grup multicast). Adresele multicast IPv6 au prefixul FF00::/8.

Notă: Adresele multicast pot fi numai adrese destinație și nu adrese sursă.

Există două tipuri de adrese IPv6 multicast:

- *Assigned multicast*.
- *Solicited node multicast*.

8.15.8.1 Assigned Multicast

Adresele multicast alocate sunt adrese de multicast rezervate pentru grupuri predefinite de dispozitive. O adresă multicast alocată este o singură adresă folosită pentru a ajunge la un grup de dispozitive ce rulează un protocol sau serviciu comun. Adresele multicast alocate sunt folosite cu anumite protocole, cum ar fi DHCPv6.

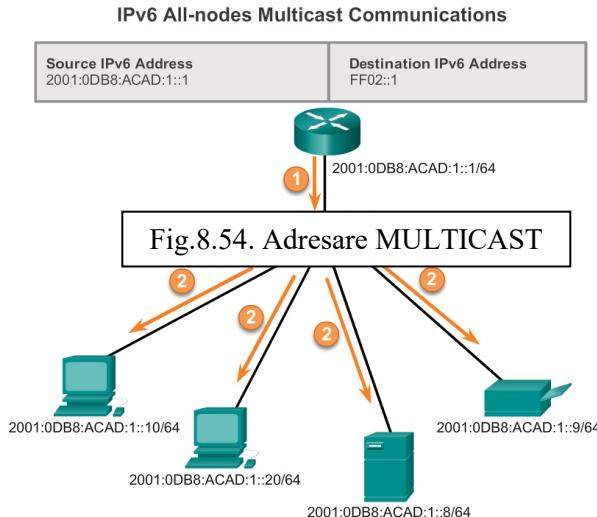
Două grupuri comune multicast alocate IPv6 sunt:

- **FF02::1 All-nodes multicast group** – Acesta este un grup multicast din care fac parte toate dispozitivele activate cu IPv6. Un pachet trimis la acest grup este primit și procesat de către toate interfețele IPv6 din legătură sau rețea. Aceasta are același efect ca o adresă de broadcast din IPv4. Fig. arată un exemplu de comunicare ce folosește all-nodes multicast address. Un router IPv6 trimite mesaje Internet Control Message Protocol version 6 (ICMPv6) RA tuturor nodurilor din

grupul multicast. Mesajul RA informează toate dispozitivele activate IPv6 din rețea despre informațiile de adresare, cum ar fi prefixul, lungimea prefixului și default gateway.

- **FF02::2 All-routers multicast group** – Aceasta este un grup multicast din care fac parte toate routerele IPv6. Un router devine un membru al grupului atunci când este activat ca router IPv6 cu comanda de configurație globală **ipv6 unicast-routing**. Un pachet trimis la acest grup este primit și procesat de către toate routerele IPv6 din legătură sau rețea.

Dispozitivele activate IPv6 trimit mesaje ICMPv6 Router Solicitation (RS) către all-routers multicast address. Mesajul RS cere un mesaj RA de la routerul IPv6 pentru a ajuta dispozitivul în configurația sa de adresă.



Un **"solicited-node"** multicast este similar cu **"all-nodes"** multicast address. Reamintim faptul că all-nodes multicast address este în esență același lucru cu un broadcast IPv4. Toate dispozitivele din rețea trebuie să proceseze traficul trimis la adresă. Pentru a reduce numărul de dispozitive ce trebuie să proceseze traficul, folosim o adresă solicited-node multicast address.

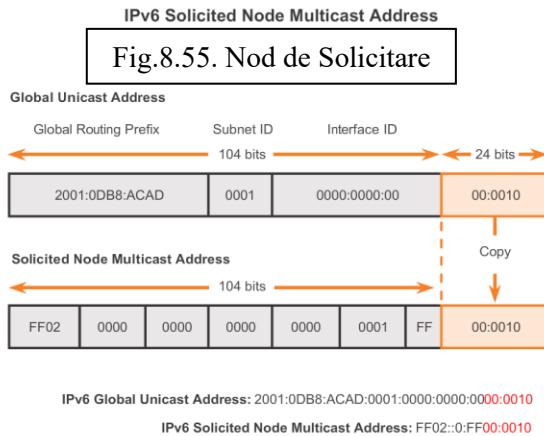
O solicited-node multicast address este o adresă ce are aceeași ultimi 24 de biți cu adresa unică globală IPv6 a dispozitivului. Singurele dispozitive ce trebuie să proceseze pachetele sunt acele dispozitive ce au aceeași ultimi 24 de biți, în extrema dreptă a ID-ului lor de interfață.

O adresă IPv6 solicited-node multicast address este creată automat atunci când este atribuită o adresă unică globală sau una unică de legătură locală. IPv6 solicited-node multicast address este creată prin combinarea prefixului special FF02:0:0:0:0:FF00::/104 cu extrema dreaptă de 24 de biți a adresei unice.

Solicited-node multicast address este alcătuită din două părți:

- **FF02:0:0:0:0:FF00::/104 multicast prefix** – Aceștia sunt primii 104 biți ai all solicited-node multicast address.
- **Least significant 24-bits** – Aceștia sunt ultimii 24 de biți din extrema dreaptă a solicited-node multicast address. Acești biți sunt copiați din cei 24 de biți de extrema dreaptă ai adresei unice globale sau unice de legătură locală a dispozitivului.

Este posibil ca mai multe dispozitive să aibă aceeași solicited-node multicast address. Deși rar, acest lucru se poate întâmpla atunci când dispozitivele au aceeași 24 de biți în extrema dreaptă a ID-urilor de interfață. Acest lucru nu crează nici-o problemă deoarece dispozitivul va procesa în continuare mesajul încapsulat, ce va include adresa completă IPv6 a dispozitivului.



8.16 Verificarea Connectivității

8.16.1 ICMP

Deși IP nu este un protocol de încredere, suita TCP/IP asigură ca mesajele să fie trimise în cazul în care au loc anumite erori. Aceste mesaje sunt trimise folosind serviciile ICMP. Scopul acestor mesaje este de a oferi feedback despre problemele legate de procesarea pachetelor IP în anumite condiții, nu pentru a face IP de încredere. Mesajele ICMP nu sunt necesare și sunt adesea nepermise în rețea din motive de securitate.

ICMP este disponibil pentru IPv4 și IPv6. ICMPv4 este protocolul de mesagerie pentru IPv4. ICMPv6 oferă aceleași servicii pentru IPv6, însă include funcționalități adiționale. În acest curs, termenul de ICMP va fi folosit pentru ambele, ICMPv4 și ICMPv6.

Tipurile de mesaje ICMP și motivele pentru care sunt transmise sunt multe. Vom discuta unele dintre aceste multe mesaje.

Mesajele ICMP comune ambelor, ICMPv4 și ICMPv6, includ:

- *Confirmarea hostului.*
- *Destinație sau serviciu fără acoperire.*
- *Timp depășit.*
- *Redirecționarea rutei.*

Confirmarea hostului – Un ICMP Echo Message poate fi folosit pentru a determina dacă un host este operațional. Hostul local trimite un ICMP Echo Request unui host. Dacă hostul este disponibil, hostul destinație răspunde cu un mesaj Echo Reply. Această utilizarea a mesajelor ICMP Echo este baza utilitarului **ping**.

Destinație sau serviciu fără acoperire – Atunci când un host sau gateway primește un pachet ce nu poate fi livrat, poate folosi un mesaj ICMP Destination Unreachable pentru a notifica sursa de faptul că destinația sau serviciul nu este disponibil. Mesajul va include un cod ce indică de ce pachetul nu poate fi livrat.

Unele dintre codurile de destinație inaccesibile pentru ICMPv4 sunt:

- 0 – *net inaccesibil.*
- 1 - *host inaccesibil.*
- 2 - *protocol inaccesibil.*
- 3 - *port inaccesibil.*

Notă: ICMPv6 are coduri similare, însă puțin diferite pentru mesajele de destinație inaccesibilă.

Timp depășit – Un mesaj ICMPv4 Time Exceeded este folosit de către un router pentru a indica faptul că un pachet nu poate fi expediat mai departe deoarece câmpul TTL al pachetului a fost decrementat până la 0. Dacă un router primește un pachet și decrementează câmpul TTL din pachetul IPv4 la zero, aruncă pachetul și trimite un mesaj ICMPv4 Time Exceeded hostului sursă.

ICMPv6 trimite de asemenea un mesaj Time Exceeded în cazul în care routerul nu poate transmite un pachet IPv6 deoarece pachetul “a expirat”. IPv6 nu are un câmp TTL; folosește câmpul de limită de hop pentru a determina dacă pachetul a expirat.

Redirecționarea rutelor – Un router ar putea folosi mesajul ICMP Redirect Message pentru a notifica hosturile din rețea de faptul că o rută mai bună este disponibilă pentru o destinație particulară. Acest mesaj poate fi folosit numai atunci când hostul sursă se află pe aceeași rețea fizică cu ambele gatewayuri.

Ambele, ICMPv4 și ICMPv6, folosesc mesaje de redirecționare de rută.

ICMPv4 Ping to a Remote Host

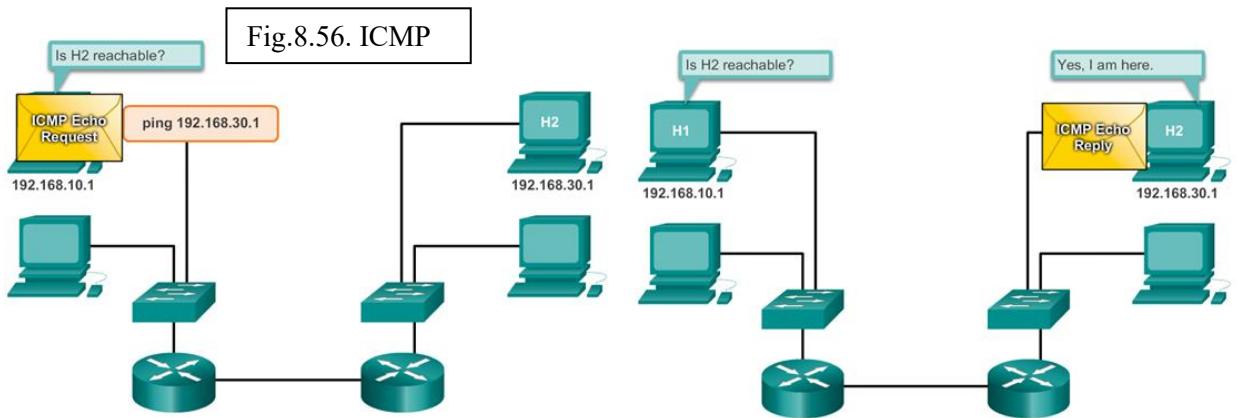


Fig.8.56. ICMP

Mesajele informaționale și de eroare aflate în ICMPv6 sunt foarte similare cu mesajele de control și eroare implementate de către ICMPv4. Însă, ICMPv6 are noi caracteristici și funcționalități îmbunătățite, neîntâlnite în ICMPv4.

ICMPv6 include patru noi protocoale ca parte a Neighbor Discovery Protocol (ND sau NDP):

- *Router Solicitation message.*
- *Router Advertisement message.*
- *Neighbor Solicitation message.*
- *Neighbor Advertisement message.*

8.16.2 Router Solicitation și Router Advertisement Messages

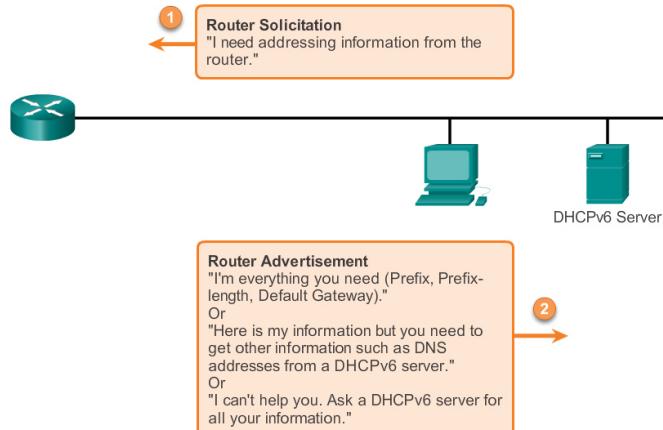
Dispozitivele activate cu IPv6 pot fi împărțite în două categorii, routere și hosturi. Mesajele Router Solicitation și Router Advertisement sunt trimise între hosturi și routere.

- **Mesajul Router Solicitation (RS):** Atunci când un host este configurat pentru a-și obține informațiile de adresare automat cu ajutorul Stateless Address Autoconfiguration (SLAAC), hostul va trimite un mesaj RS routerului. Mesajul RS este trimis ca un mesaj multicast IPv6 all-routers.
- **Mesajul Router Advertisement (RA):** Mesajele RA sunt trimise de către routere pentru a furniza informații de adresă hosturilor folosind SLAAC. Mesajul RA poate conține informații de adresare pentru host, cum ar fi prefixul și lungimea prefixului. Un router va transmite un mesaj RA periodic sau ca răspuns pentru un mesaj RS. Implicit, routerele

Cisco trimite mesaj RA la fiecare 200 de secunde. Mesajele RA sunt trimise la adresa multicast IPv6 all-nodes. Un host ce folosește SLAAC va transmite propriul default gateway la adresa de legătură locală a routerului ce transmite mesajul RA.

Router Solicitation and Router Advertisement Messages

Fig.8.57. Mesaje RS/RA



ICMPv6 Neighbor Discovery Protocol include două tipuri de mesaje suplimentare, mesaje Neighbor Solicitation (NS) și Neighbor Advertisement (NA).

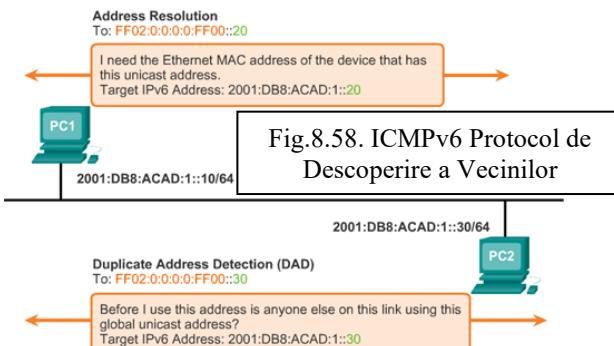
Mesajele Neighbor Solicitation (NS) și Neighbor Advertisement (NA) sunt folosite pentru:

- *Rezoluția adresei.*
- *Detectarea adresei dupicat (DAD).*

Rezoluția adresei – Rezoluția de adresă este folosită atunci când un dispozitiv de pe LAN știe adresa unicast IPv6 a unei destinații, dar nu știe adresa Ethernet MAC. Pentru a determina adresa MAC a destinației, dispozitivul va transmite un mesaj NS adresei de nod solicitată. Mesajul va include adresa IPv6 cunoscută. Dispozitivul ce are adresa IPv6 cunoscută va răspunde cu un mesaj NA ce conține adresa Ethernet MAC.

Detectarea adresei dupicat – Atunci când un dispozitiv are atribuită o adresă unicast globală sau una unicast de legătură locală, este recomandat ca DAD să fie efectuat pe adresa pentru asigurarea de faptul că este unică. Pentru a verifica unicitatea unei adrese, dispozitivul va trimite un mesaj NS cu propria adresă IPv6 ca adresa IPv6 țintă. Dacă alt dispozitiv din rețea are această adresă, va răspunde cu un mesaj NA. Acest mesaj NA va înștiința sursa de faptul că adresa este folosită. Dacă un mesaj NA nu este primit într-o anumită perioadă de timp, adresa unicast este unică și utilizabilă.

Notă: DAD nu este necesar, însă RFC 4861 recomandă ca DAD să fie efectuat pe adresele unicast.



8.17 Testare și Verificare

Ping este utilitarul de testare ce folosește meseje ICMP echo request și echo reply pentru a testa conectivitatea dintre hosturi. **Ping** funcționază cu ambele tipuri de adrese, IPv4 și IPv6.

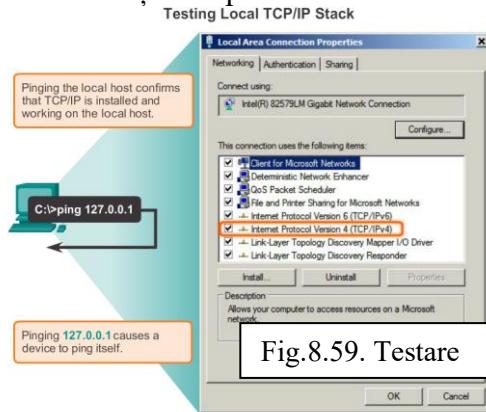
Pentru a testa conectivitatea cu un alt host dintr-o rețea, un echo request este trimis adresei de host prin folosirea comenzi **ping**. Dacă hostul de la adresa specificată primește echo request, răspunde cu un echo reply. Cu fiecare echo reply primit, **ping** oferă feedback în timpul în care cererea a fost trimisă și confirmarea a fost primită. Acest lucru poate fi privit ca o măsură a performanței rețelei.

Ping are o valoare de timeout pentru reply. Dacă un răspuns nu a fost primit în acest timp, **ping** oferă un mesaj ce indică faptul că un răspuns nu a fost primit. Acest lucru indică de obicei că există o problemă, dar și faptul că anumite caracteristici de securitate ce blochează mesajele **ping** au fost activate în rețea.

După ce toate cererile au fost trimise, utilitarul **ping** oferă un rezumat ce include rata de succes și timpul mediu de dus-întors de la destinație.

Pinging în adresa de Local Loopback – Există unele cazuri speciale de testare și verificare pentru care putem utiliza **ping**. Un caz este testarea configurației interne de IPv4 sau IPv6 pe hostul local. Pentru a efectua acest test, folosim **ping** pe adresa de loopback local 127.0.0.1 pentru IPv4 (:1 pentru IPv6). Testarea IPv4 loopback este evidențiată în Fig. de mai jos.

Un răspuns de la 127.0.0.1 pentru IPv4 (:1 pentru IPv6), indică că un IP este instalat adecvat pe host. Acest răspuns provine de la nivelul rețea. Acest răspuns nu este însă o indicație a faptului că adresele, măștile sau gateways sunt corecte. Nu indică nimic despre statusul nivelului inferior al stivei de rețea. Testează pur și simplu IPul la nivelul de rețea IP. Dacă primim un mesaj de eroare, este o indicație a faptului că TCP/IP nu este operational pe host.

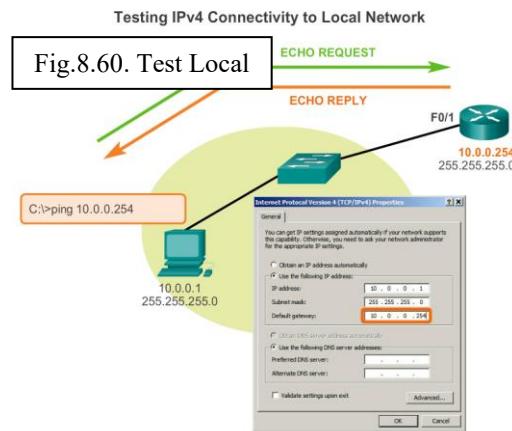


Putem folosi **ping** pentru a testa de asemenea abilitatea unui host de a comunica cu rețeaua locală. Acest lucru se realizează în general prin efectuarea de **ping** spre gateway a hostului. Un **ping** spre gateway indică faptul că un host și interfața routerului sunt operaționale pe rețeaua locală.

Pentru acest test, adresa de gateway este cea mai utilizată deoarece routerul este, în mod normal, operațional. Dacă adresa de gateway nu răspunde, un **ping** poate fi trimis la adresa IP a altui host din rețeaua locală ce este cunoscut ca fiind operațional.

Dacă gateway sau alt host răspunde, hostul local poate comunica cu succes peste rețeaua locală. Dacă gateway nu răspunde, dar un alt host răspunde, se poate indica faptul că este o problema cu interfața routerului (gateway).

O posibilitate este aceea că o adresă greșită de gateway a fost configurată pe host. O altă posibilitate este ca interfața routerului să fie funcțională, însă să aibă securitate aplicată ce împiedică procesarea și răspunsul la cererile de **ping**.

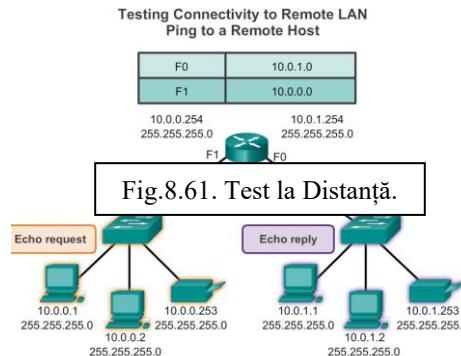


Ping poate fi de asemenea folosit pentru a testa abilitatea unui host local de a comunica prin internetwork. Hostul local poate da **ping** către un host IPv4 funcțional de pe o rețea de la distanță, aşa cum se poate vedea și în Fig. .

Dacă acest **ping** este cu succes, funcționarea unei mari bucăți din internetwork poate fi verificată. Un **ping** cu succes în internetwork confirmă comunicarea din rețeaua locală, funcționalitatea routerului (gateway) și funcționalitatea tuturor celorlalte routere ce se află în calea dintre rețeaua locală și rețeaua hostului de la distanță.

În plus, funcționalitatea hostului de la distanță poate fi verificată. Dacă hostul de la distanță nu poate comunica în exteriorul rețelei sale locale, nu va putea răspunde.

Notă: Mai mulți administratori de rețea limitează sau interzic intrarea mesajelor ICMP în rețeaua întreprinderii; prin urmare, lipsa unui răspuns **ping** poate fi provocată de restricțiile de securitate.



Ping este folosit pentru testarea comunicării dintre două hosturi, însă nu oferă informații despre detalii ale dispozitivelor dintre hosturi. Traceroute (**tracert**) este o utilitarul ce generează o listă de hopuri ce au fost atinse cu succes de-a lungul traseului. Lista poate oferi informații importante de verificare și depanare. Dacă datele ajung la destinație, **tracert** listează interfața fiecărui router din calea celor două hosturi. Dacă datele ajung doar până la un anumit hop din drum, adresa ultimului router ce a răspuns poate oferi o indicație cu privire la locul în care sunt găsite probleme sau restricții de securitate.

Round Trip Time (RTT) – Folosirea lui traceroute oferă **Round Trip Time (RTT)** pentru fiecare hop din cale și indică dacă un hop nu răspunde cu succes. **Round trip time** este timpul necesar pentru ca un pachet să ajungă la hostul de la distanță sau pentru ca un răspuns să ajungă la sursă. Simbolul “(*)” indică un pachet pierdut sau fără răspuns.

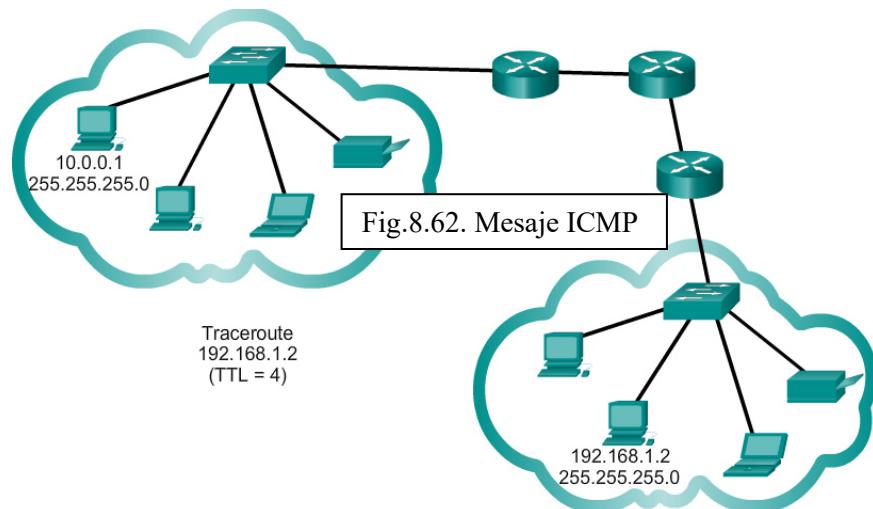
Acstea informații pot fi folosite pentru a localiza un router cu probleme din cale. Dacă afișajul arată tempi de răspuns mari sau pierderi de date de la un anumit hop, este o indicație a faptului că resursele routerului sau conexiunile sale pot fi "stresate".

IPv4 Time-to-Live (TTL) and IPv6 Hop Limit – Traceroute folosește o funcție de câmp TTL din IPv4 și Hop limit din IPv6 din headerele de nivel 3, împreună cu mesajul ICMP time exceeded.

Prima secvență de mesaje trimise de la traceroute vor avea un câmp TTL cu valoarea 1. Acest lucru face ca TTL să ajungă la 0 în pachetul IPv4 la primul router. Acest router răspunde apoi cu un mesaj ICMPv4. Traceroute are acum adresa primului hop.

Traceroute crește apoi câmpul TTL progresiv (2, 3, 4...) pentru fiecare secvență de mesaje. Acest lucru oferă **tracert** cu adresa fiecărui hop din cale. Câmpul TTL continuă să crească până când ajunge la destinație sau este incrementat până la un anumit maxim predefinit.

O dată ajuns la destinație, hostul răspunde fie cu un mesaj ICMP port unreachable, fie cu un mesaj ICMP echo reply message, în locul mesajului ICMP time exceeded.



8.18 Concluzii Capitolul 8

În acest capitol a fost prezentat modul în care întreprinderile mici și mijlocii sunt conectate la rețele în grupuri. Internet of Everything a fost de asemenea introdus în activitate de modelare de la început.



Adresele IP sunt ierarhice în rețea, subrețea și părțile de host. O adresă IP poate reprezenta o rețea completă, un host specific, sau adresa de broadcast a rețelei.

Înțelegerea notației binare este importantă atunci când se stabilește dacă două hosturi sunt în același rețea. Bițiile din partea de rețea a adresei IP trebuie să fie identici pentru toate dispozitivele din același rețea. Mască de rețea sau prefixul este folosit pentru a determina partea de rețea a unei adrese IP. Adresele IP pot fi atribuite fie static, fie dinamic. DHCP permite atribuirea dinamică a informațiilor de adresare cum ar fi adresa IP, masca de rețea, default gateway sau alte informații de configurație.

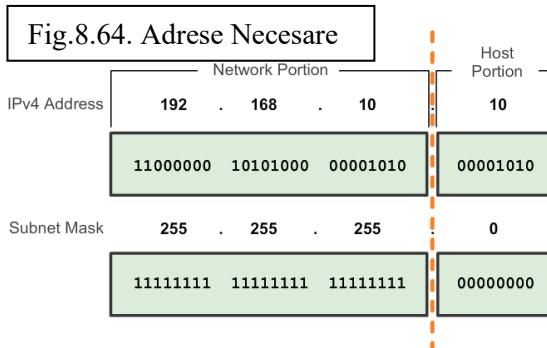
Hosturile IPv4 pot comunica într-unul dintre cele trei moduri diferite: unicast, broadcast și multicast. De asemenea, blocurile de adrese folosite în rețele care necesită acces limitat sau inexistent la Internet sunt numite adrese private. Blocurile de adrese IPv4 private sunt: 10.0.0.0/8, 172.16.0.0/12 și 192.168.0.0/16.

Epuizarea spațiului de adrese IPv4 este factorul motivant pentru trecerea la IPv6. Fiecare adresă IPv6 are 128 de biți, în comparație cu o adresă IPv4 de 32 de biți. IPv6 nu folosește notație de mască de rețea zecimală punctată. Lungimea prefixului este folosită pentru a indica partea de rețea a unei adrese IPv6 folosind următorul format: IPv6 address/prefix length.

Există trei tipuri de adrese IPv6: unicast, multicast și anycast. O adresă IPv6 de legătură locală permite unui dispozitiv să comunique cu alte dispozitive activate IPv6 din același legătură locală sau numai pe respectiva legătură (subrețea). Pachetele cu o adresă sursă sau destinație de legătură locală nu pot fi rutate în afara legăturii de unde este original pachetul. Adresele IPv6 de legătură locală sunt în spațiul FE80::/10.

ICMP este disponibil atât pentru IPv4, cât și pentru IPv6. ICMPv4 este protocolul de mesagerie pentru IPv4. ICMPv6 oferă aceleași servicii pentru IPv6, însă include funcționalități suplimentare.

După ce este implementată, o rețea IP trebuie să fie testată pentru a verifica conectivitatea sa și performanța funcțională a sa.



CAPITOLUL 9. SUBNETAREA REȚEELOR IP

Introducere

Proiectarea, implementarea și gestionarea unui plan de adresare IP eficient asigură faptul că rețelele pot funcționa eficient și corect. Acest lucru este valabil mai ales o dată cu creșterea numărului de conexiuni ale hosturilor la o rețea. Înțelegerea structurii ierarhice ale adresării IP și modul în care se poate modifica această ierarhie pentru îndeplinirea eficientă a cerințelor de rutare este o parte importantă a planificării unei scheme de adresare IP.

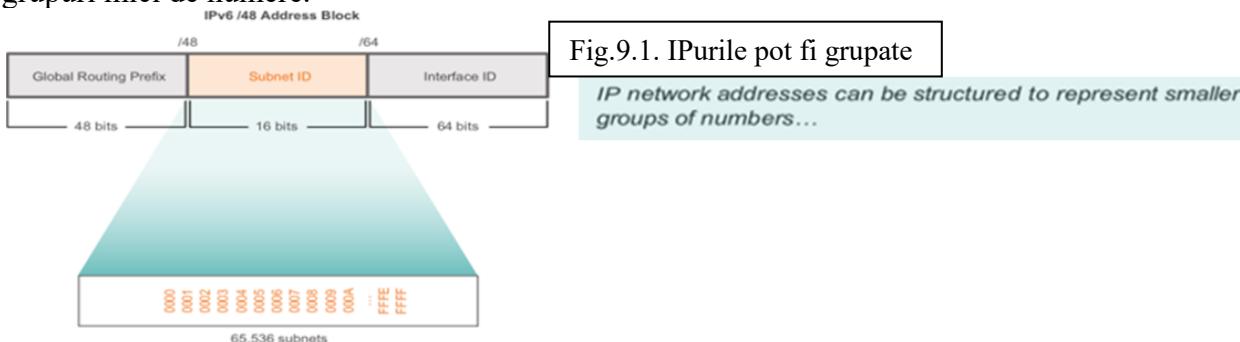
În adresarea originală IPv4, există două nivele de ierarhie: o rețea și un host. Aceste două nivele de adresare permit grupuri de bază de rețea care facilitează routarea de pachete la o rețea destinație. Un router transmite pachete în funcție de partea de rețea a unei adrese IP; o dată ce este localizată rețeaua, partea de host a adresei permite identificarea dispozitivului destinație.

Însă, o dată cu creșterea rețelelor, numeroase organizații adaugă sute, chiar mii de hosturi în rețelele lor, astfel încât cele două nivele de ierarhie devin insuficiente.

Împărțirea unei rețele adaugă un nou nivel la ierarhia rețelei, creând, în esență, trei niveluri: o rețea, o subrețea și un host. Introducerea unui nivel suplimentar în ierarhie creează subgrupuri suplimentare într-o rețea IP ceea ce facilitează livrarea mai rapidă de pachete și adaugă filtrarea prin ajustarea minimizării traficului "local".

Acest capitol examinează, în detaliu, crearea și atribuirea de adrese IP de rețea și subrețea prin utilizarea măștii de rețea.

În acest capitol, se va învăța cum să se grupeze dispozitivele în subrețele, sau grupuri mai mici de rețea, dintr-o rețea mai mare. Adresele IP de rețea pot fi structurate prin reprezentarea în grupuri mici de numere.



9.1 Subnetarea Rețeelor IPv4 – Segmentarea Rețelelor

În implementările de rețea la început, era comun organizațiilor să aibă toate computerele și alte dispozitive conectate la o singură rețea IP. Toate dispozitivele din organizație aveau atribuită o adresă de rețea IP cu un ID de rețea corespunzător. Acest tip de configurație este cunoscut ca o proiectare de rețea plată. Într-o rețea mică, cu un număr limitat de dispozitive, o proiectare de rețea plată nu reprezintă o problemă. Însă, odată cu creșterea rețelei, acest tip de configurație poate crea mari probleme.

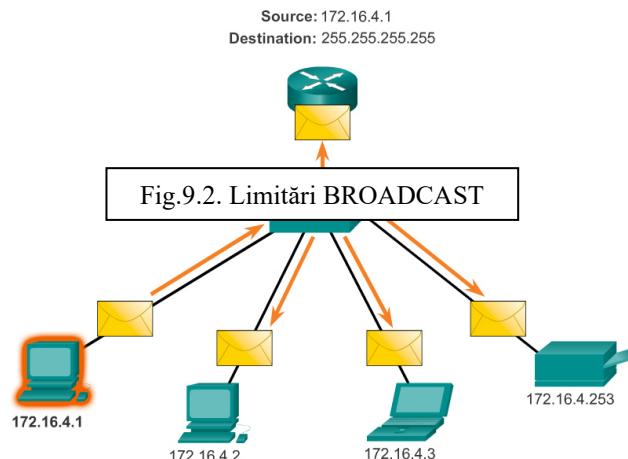
Într-un LAN Ethernet, dispozitivele folosesc adresarea broadcast pentru a localiza serviciile și dispozitivele necesare. Reamintim că solicitările de tip comunicație broadcast sunt trimise tuturor hosturilor dintr-o rețea IP. Dynamic Host Configuration Protocol (DHCP) este un exemplu de serviciu de rețea ce depinde de adresarea broadcast. Dispozitivele trimit comunicării de tip broadcast prin rețea pentru a localiza serverul DHCP. Într-o rețea mare, acest lucru ar putea crea o cantitate mare de trafic ce încetinește operațiile de rețea. În plus, deoarece o comunicație broadcast este adresată tuturor dispozitivelor, toate dispozitivele trebuie să accepte și să

proceseze traficul, având ca rezultat cerințe de procesare crescute. Dacă un dispozitiv trebuie să proceseze o cantitate mare de cerințe broadcast, ar putea încetini chiar și operațiile de dispozitiv. Din aceste motive, rețelele mari trebuie să fie segmentate în subrețele mai mici, localizate în grupuri mai mici de dispozitive și servicii.

Acest proces de segmentare a unei rețele, prin divizarea lor în spații de rețea mai mici, se numește **subnetare**, iar rețelele astfel obținute se numesc **subrețele**. Administratorii de rețea pot grupa dispozitivele și serviciile în subrețele ce sunt determinate de o locație geografică (cum ar fi campusul unei universități), de o unitate organizațională (cum ar fi departamentul IT), după tipuri de dispozitive (imprimante, servere, echipamente WAN) sau orice altă diviziune ce dă un sens rețelei. Subnetarea poate reduce traficul global de rețea și poate îmbunătăți performanța rețelei.

Notă: O subrețea este echivalentă cu o rețea și acești termeni pot fi folosiți în mod alternativ. Multe rețele reprezintă o subrețea a unui bloc mai mare de adrese.

Limited Broadcast

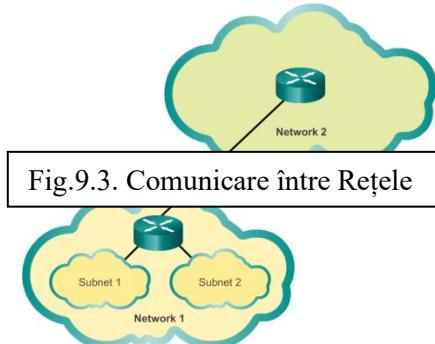


Un router este necesar pentru ca dispozitivele din rețele diferite să poată comunica. Dispozitivele dintr-o rețea folosesc interfață routerului atașată la LANul lor ca default gateway. Traficul destinat unui dispozitiv dintr-o rețea de la distanță va fi procesat de către router și trimis spre destinație. Pentru a determina dacă traficul este local sau la distanță, routerul folosește masca de rețea.

Într-un spațiu de rețea subnetat, acest lucru merge în același mod. Ca și în Fig. , subnetarea creează mai multe rețele logice dintr-un singur bloc de adrese sau adresa de rețea. Fiecare subrețea este tratată ca un spațiu de rețea separat. Dispozitivele din aceeași subrețea trebuie să aibă o adresă, mască de rețea și default gateway ce corespund subrețelei din care fac parte.

Traficul nu poate fi transmis între subrețele fără utilizarea unui router. Fiecare interfață a routerului trebuie să aibă o adresă de host IPv4 ce aparține rețelei sau subrețelei la care este conectată interfața routerului.

Communicating between Networks



9.1.1 Subnetarea IP este FUNDAMENTALĂ

Așa cum se observă și în Fig. , planificarea subrețelelor de rețea necesită examinarea nevoilor utilizării rețelei organizației și modul în care subrețelele vor fi structurate. Efectuarea unui studiu de cerință de rețea este punctul de plecare. Acest lucru înseamnă privirea întregii rețele și determinarea secțiunilor principale ale rețelei și modul în care vor fi segmentate. Planul de adresare include deciderea necesităților pentru fiecare subrețea în ceea ce privește dimensiunea, numărul de hosturi, cum adresele de host vor fi atribuite, ce hosturi vor necesita adrese IP statice și ce hosturi pot folosi DHCP pentru obținerea informațiilor de adresare a lor.

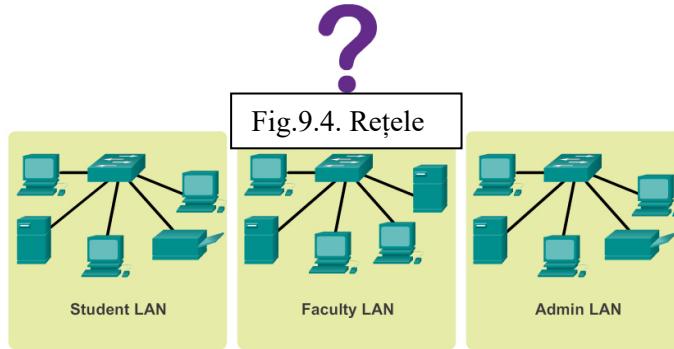
Dimensiunea subrețelei presupune planificarea numărului de hosturi ce necesită adrese IP de host în fiecare subrețea a rețelei private subnetată. De exemplu, într-un design de rețea de campus trebuie să luăm în considerare câte hosturi sunt necesare în LANul administrativ, câte în LAN facultății și câte în LANul de student. Într-o rețea de domiciliu, o considerare ar putea fi realizată prin numărul de hosturi din Main House LAN și numărul de hosturi din Home Office LAN.

Așa cum am discutat mai devreme, spațiul de adrese IP folosit într-un LAN este alegerea administratorului de rețea și necesită considerații atente pentru asigurarea faptului că sunt destule adrese de host disponibile pentru hosturile actuale cunoscute și pentru o extindere viitoare. Amintim faptul că spațiile de adrese private IP sunt:

- *10.0.0.0 cu o masca de rețea 255.0.0.0.*
- *172.16.0.0 cu o masca de rețea 255.240.0.0.*
- *192.168.0.0 cu o masca de rețea 255.255.0.0.*

Cunoascând cerințele de adresare IP vom determina spațiul sau spaile de adrese de host ce le vom implementa. Subnetarea spațiului de adresare privată IP selectat va oferi adresele de host necesare pentru îndeplinirea cerințelor rețelei.

Adresele pulice folosite pentru conectarea la Internet sunt de obicei alocate de la un furnizor de servicii - *ISP*. Deși se aplică aceleași principii pentru subnetare, nu este în general responsabilitatea administratorului de rețea al organizației.



Creem standarde de atribuire a adresei IP din fiecare spațiu de subrețea. De exemplu:

- *Imprimantele și serverele vor avea atribuite adrese IP statice.*
- *Utilizatorul va primi adresele IP de la servere DHCP ce folosesc subrețele /24.*
- *Routerele au atribuite primele adrese de host din range.*

Doi dintre factorii cei mai importanți ce conduc la determinarea blocului de adresare privată necesari sunt numărul de subrețele necesare și maximul numărului de hosturi necesare pe fiecare subrețea. Fiecare dintre aceste blocuri de adresă va permite alocarea de hosturi adecvată în funcție de dimensiunea dată a unei rețele și de hosturile cerute actual și în viitorul apropiat. Cerințele de spațiu de adrese IP vor determina spațiul sau rangeurile hosturilor.

În următoarele exemple vom vedea subnetarea în funcție de blocurile de adrese ce au masca de rețea 255.0.0.0, 255.255.0.0 și 255.255.255.0.



Fig.9.5. Planificare.

9.1.2 Subnetarea unei Rețele IPv4

Fiecare adresa de rețea are un spațiu valid de adrese de host. Toate dispozitivele din aceeași rețea vor avea o adresă de host IPv4 din rețea și o mască de rețea comună (sau prefix de rețea).

Prefixul și masca de rețea sunt moduri diferite de reprezentare a același lucru – partea de rețea a unei adrese.

Subrețele IPv4 sunt create prin folosirea unuia sau a mai multor biți ca biți de rețea. Acest lucru se realizează prin extinderea măștii și împrumutarea câtorva biți din partea de host a adresei pentru a crea biți de rețea suplimentari. Cu cât sunt împrumutați mai mulți biți, mai multe subrețele pot fi definite. Pentru fiecare bit împrumutat, numărul de subrețele disponibile se dublează. De exemplu, dacă 1 bit este împrumutat, pot fi create 2 subrețele. Dacă 2 biți sunt împrumutați, se crează 4, dacă 3 biți sunt împrumutați se crează 8 și aşa mai departe. Însă, cu fiecare bit împrumutat, mai puține adrese de host sunt disponibile pe subrețea.

Biții pot fi împrumutați numai din partea de host a adresei. Partea de rețea a adresei este alocată de către furnizorul de servicii și nu poate fi schimbată.

Notă: În exemplele din imagini, numai ultimul octet este arătat în binar datorită faptului că numai biții din partea de host sunt împrumutați.

Așa cum se poate observa în Fig.9.6.A, rețeaua 192.168.1.0/24 are 24 de biți în partea de rețea și 8 biți în partea de host, ceea ce indică masca de rețea 255.255.255.0 sau notația /24. Fără subnetare, rețeaua suportă o singură interfață LAN. Dacă este necesar un LAN suplimentar, rețeaua trebuie să fie subnetată.

Așa cum se poate observa în Fig.9.6.B, 1 bit este împrumutat de la cel mai semnificativ bit din partea de host, pentru a extinde partea de host la 25 de biți. Acest lucru crează 2 subrețele identificate prin folosirea unui 0 pe bitul împrumutat pentru prima rețea și un 1 pentru a doua rețea. Masca de rețea pentru ambele rețele folosește un 1 în poziția bitului împrumutat pentru a indica faptul că acest bit este acum parte din zona de rețea.

Așa cum se poate observa în Fig.9.6.C, atunci când convertim octetul binar în zecimal observăm faptul că prima adresă de subrețea este 192.168.1.0 și a doua adresă de subrețea este 192.168.1.128. Deoarece un bit a fost împrumutat, masca de rețea pentru fiecare subrețea este 255.255.255.128 sau /25.

Address	192	168	1	0000	0000
Mask	255	255	255	0000	0000
Fig.9.6.A - Rețea					
Network Portion					Host Portion

Original	192.	168.	1.	0	000	0000	Network: 192.168.1.0/24
Mask	255.	255.	255.	0	000	0000	Mask: 255.255.255.0

Fig.9.6.B- Reprezentare în zecimal

Borrowing 1 bit creates 2 subnets with the same mask.

Net 0	192.	168.	1.	0	000	0000	Network: 192.168.1.0/25
Mask	255.	255.	255.	1	000	0000	Mask: 255.255.255.128

Net 1	192.	168.	1.	1	000	0000	Network: 192.168.1.128/25
Mask	255.	255.	255.	1	000	0000	Mask: 255.255.255.128

Borrow 1 bit from the host portion of the address.

Original 192. 168. 1. 0 000 0000 1 Network
 Mask 255. 255. 255. 0 000 0000

Fig.9.6.C.Împrumut de Biți

The borrowed bit value is 0 for the Net 0 address.

Net 0	192.	168.	1.	0	000	0000
-------	------	------	----	---	-----	------

The borrowed bit value is 1 for the Net 1 address. 2 Subnets

Net 1	192.	168.	1.	1	000	0000
-------	------	------	----	---	-----	------

The new subnets have the SAME subnet mask.

Mask	255.	255.	255.	1	000	0000
------	------	------	------	---	-----	------

În exemplul anterior, rețeaua 192.168.1.0/24 a fost subnetată pentru a crea două subrețele:

- 192.168.1.0/25.
- 192.168.1.128/25.

În Fig.9.7.A, observăm faptul că routerul R1 are două segmente de LAN atașate la interfețele sale GigabitEthernet. Subrețelele vor fi folosite pentru segmentele atașate la aceste interfețe. Pentru a servi drept gateway pentru dispozitivele din LAN, fiecare interfață a routerului trebuie să aibă atribuită o adresă IP din spațiul de adrese valide pentru subrețea calculată. Este o practică comună utilizarea primei sau a ultimei adrese disponibile dintr-un spațiu de adrese pentru adresa de interfață a routerului.

Prima subrețea, 192.168.1.0/25, este folosită pentru rețeaua atașată la GigabitEthernet 0/0 și a doua subrețea 192.168.1.128/25, este folosită pentru rețeaua atașată la GigabitEthernet 0/1. Pentru a atribui o adresă IP pe fiecare dintre cele două interfețe, este necesară determinarea spațiului de adrese valide pentru fiecare subrețea.

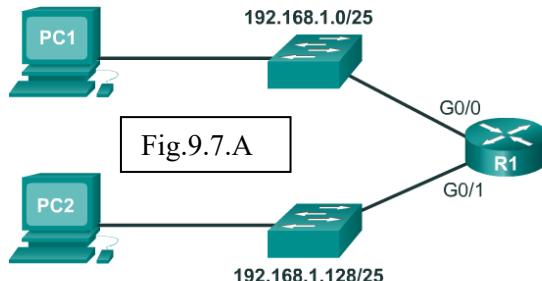
Următoarele sunt liniile directoare pentru fiecare dintre subrețele:

- **Adresa de rețea** – toți biții de 0 în partea de host a adresei.
- **Adresa primului host** – toți biții de 0 plus un bit de 1 cel mai din dreapta în partea de host a adresei.
- **Adresa ultimului host** – toți biții de 1 plus un bit de 1 cel mai din dreapta în partea de host a adresei.
- **Adresa de broadcast** – toți biții de 1 în partea de host a adresei.

Așa cum se poate vedea în Fig.9.7.B, prima adresă de host pentru rețeaua 192.168.1.0/25 este 192.168.1.1, iar ultima adresă de host este 192.168.1.126. Fig.9.7.C arată faptul că prima adresă de host pentru rețeaua 192.168.1.128/25 este 192.168.1.129, iar ultima adresă de host este 192.168.1.254.

Pentru a atribui prima adresă de host din fiecare subrețea interfeței routerului pentru respectiva subrețea, folosim comanda **ip address** în modul de configurație interfață, așa cum este arătat în Fig.9.7.D. De remarcat faptul că fiecare subrețea folosește masca de rețea 255.255.255.128 pentru a indica că partea de rețea a adresei este de 25 de biți.

O configurație de host pentru rețeaua 192.168.1.128/25 este prezentată în Fig.9.7.E. De remarcat faptul că adresa IP de gateway este adresa configurată pe interfața G0/1 a R1, 192.168.1.129, iar masca de rețea este 255.255.255.128.



Address Range for 192.168.1.128/25 Subnet

Network Address	192. 168. 1. 1 . 000 0000	= 192.168.1.128
First Host Address	192. 168. 1. 1 . 000 0001	= 192.168.1.129
Last Host Address	192. 168. 1. 1 . 111 1110	= 192.168.1.254
Broadcast Address	192. 168. 1. 1 . 111 1111	= 192.168.1.255

Fig.9.7.C

Network Address	192. 168. 1. 0 . 000 0000	= 192.168.1.0
First Host Address	192. 168. 1. 0 . 000 0001	= 192.168.1.1
Last Host Address	192. 168. 1. 0 . 111 1110	= 192.168.1.126
Broadcast Address	192. 168. 1. 0 . 111 1111	= 192.168.1.127

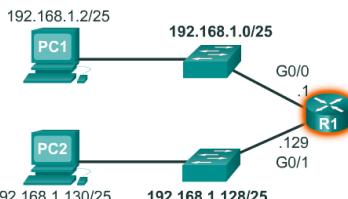
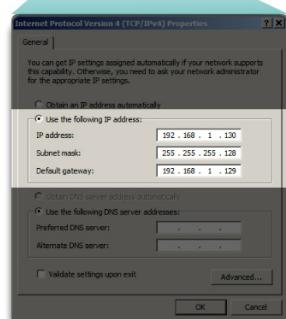
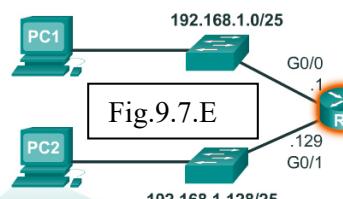


Fig.9.7.D

```
R1(config)#interface gigabitethernet 0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.128
R1(config-if)#exit
R1(config)#interface gigabitethernet 0/1
R1(config-if)#ip address 192.168.1.129 255.255.255.128
```



Calcularea Subrețelelor – Folosim următoarea formulă pentru a calcula numărul de subrețele : 2^n (unde n este numărul de biți împrumutați).

Așa cum se observă și în Fig. 1, pentru exemplul 192.168.1.0/25, calculul arată astfel: $2^1 = 2$ subrețele

Calcularea Numărului de Hosturi – Folosim următoarea formulă pentru a calcula numărul de hosturi pe subrețea : 2^n (unde n este numărul de biți rămași în câmpul de host).

Așa cum se poate observa și în Fig. 2, pentru exemplul 192.168.1.0/25, calculul arată astfel : $2^7 = 128$ adrese

din care doar 126 pot fi atribuite echipamentelor (*adresa de rețea și broadcast nu pot fi asignate*).

Deoarece hosturile nu pot folosi adresa de rețea sau broadcast dintr-o subrețea, 2 dintre aceste adrese nu sunt asignabile. Acest lucru înseamnă că fiecare subrețea are 126 de adrese de host valide.

Deci în acest exemplu, împrumutând 1 bit de host rezultă crearea a 2 subrețele și fiecare subrețea are un număr de 126 de hosturi asignabile.

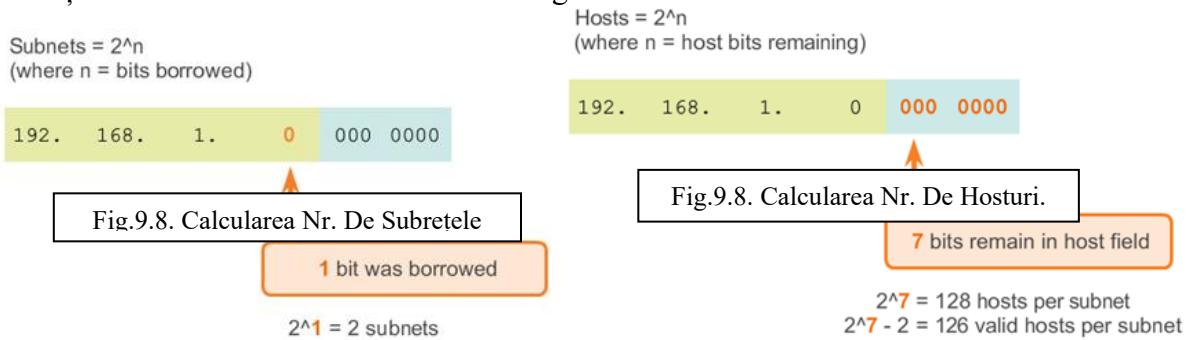


Fig.9.8. Calcularea Nr. De Subretele

Fig.9.8. Calcularea Nr. De Hosturi.

Să considerăm o rețea care necesită trei subrețele.

Utilizând același bloc de adrese 192.168.1.0/24, biții de host trebuie să fie împrumutați pentru a crea cel puțin 3 subrețele. Împrumutarea unui singur bit oferă 2 subrețele. Pentru a oferi mai multe rețele, mai mulți biți de host trebuie să fie împrumutați. Calculând numărul de subrețele create dacă împrumutăm 2 biți folosind formula **2ⁿ numărul de biți împrumutați** rezultă $2^2 = 4$ subrețele.

Împrumutarea a 2 biți crează 4 subrețele, așa cum se observă și în Fig.9.10.A.

Reamintim faptul că masca de rețea trebuie să fie schimbată pentru a reflecta biții împrumutați. În acest exemplu, când 2 biți sunt împrumutați, masca se extinde cu 2 biți în ultimul octet. În zecimal, masca se reprezintă ca 255.255.255.192, deoarece ultimul octet în binar este 1100 0000.

Pentru a calcula numărul de hosturi, examinăm ultimul octet. După împrumutarea a doi biți pentru subrețea, există 6 biți rămași.

Aplicăm formula de calculare de host , așa cum este prezentată în Fig.9.10.B.

$$2^6 = 64$$

De reținut faptul că dacă toți biții sunt 0 din partea de host a adresei este adresa de rețea, iar toți biții de 1 în partea de host rezultă adresa de broadcast. Prin urmare, există numai 62 de adrese de host care sunt disponibile pentru fiecare subrețea.

Așa cum se poate observa în Fig.9.10.C, prima adresă de host pentru prima subrețea este 192.168.1.1 și ultima adresă de host este 192.168.1.62. Fig.9.10.D prezintă spațiile de adrese pentru subrețelele 0-2. Reamintim faptul că fiecare host trebuie să aibă o adresă IP validă din spațiul definit pentru respectivul segment de rețea. Rețeaua atribuită interfeței routerului va determina cărui segment îi aparține un host.

Fig.910.E prezintă un exemplu de config.re. În această config.re, prima rețea este atribuită interfeței GigabitEthernet 0/0, a doua interfeței GigabitEthernet 0/1, iar a treia interfeței Serial 0/0/0.

Folosind un plan de adresare general, prima adresă de host este atribuită interfeței routerului. Hosturile din fiecare subrețea vor utiliza adresa interfeței routerului ca adresă de default gateway.

- *PC1 (192.168.1.2/26) va utiliza 192.168.1.1 (adresa interfeței G0/0 a R1) ca adresă de default gateway.*
- *PC2 (192.168.1.66/26) va utiliza 192.168.1.65 (adresa interfeței G0/1 a R1) ca adresă de default gateway.*

Notă: Toate dispozitivele din aceeași subrețea vor avea o adresă de host IPv4 din spațiul de adrese de host și va folosi aceeași mască de rețea.

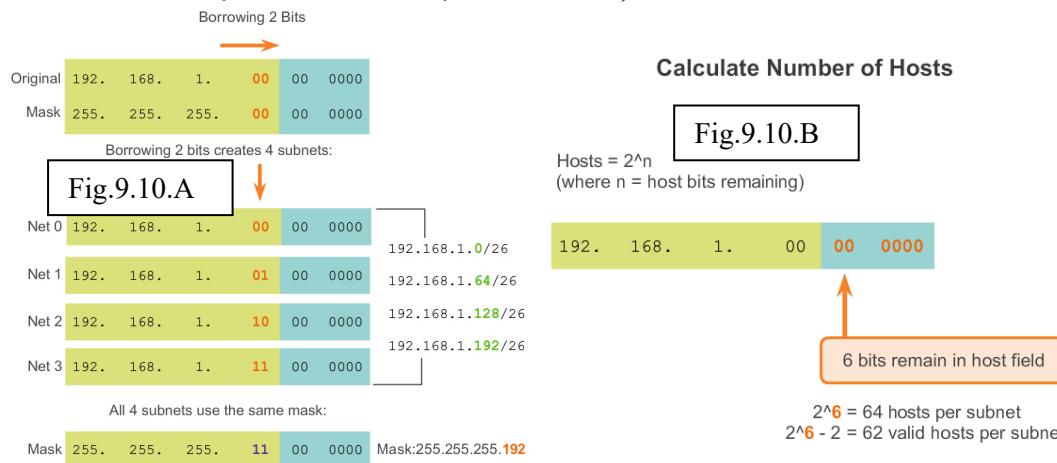
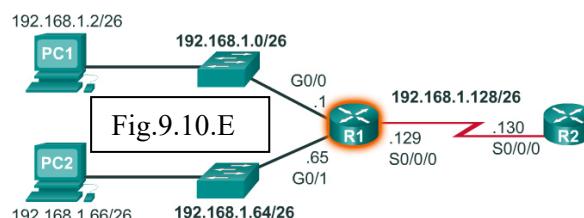


Fig.9.10.D Address Ranges Nets 0 - 2

Address Range for 192.168.1.0/26 Subnet	
Network Address	192.168.1.00000000 = 192.168.1.0
First Host Address	192.168.1.00000001 = 192.168.1.1
Last Host Address	192.168.1.01111110 = 192.168.1.62
Broadcast Address	192.168.1.01111111 = 192.168.1.63

Fig.9.10.C



```
R1(config)#interface gigabitethernet 0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.192
R1(config-if)#exit
R1(config)#interface gigabitethernet 0/1
R1(config-if)#ip address 192.168.1.65 255.255.255.192
R1(config-if)#exit
R1(config)#interface serial 0/0/0
R1(config-if)#ip address 192.168.1.129 255.255.255.192
```

Să considerăm o rețea ce necesită cinci subrețele, aşa cum se poate vedea în Fig.9.11.A.

Utilizând același bloc de adrese 192.168.1.0/24, biții de host trebuie să fie împrumutați pentru a crea cel puțin 5 subrețele. Împrumutarea a 3 biți oferă 4 subrețele, aşa cum am observat în exemplul anterior. Pentru a oferi mai multe rețele, mai mulți biți trebuie să fie împrumutați. Calculăm numărul de subrețele create dacă 3 biți sunt împrumutați, folosind formala :

$$2^3 = 8 \text{ subrețele.}$$

Așa cum se poate vedea în Fig.9.11.B și Fig.9.11.C, împrumutarea a 3 biți crează 8 subrețele. Atunci când 3 biți sunt împrumutați, masca de rețea este extinsă cu 3 biți în ultimul octet (/27), rezultând masca de rețea 255.255.255.224. Toate dispozitivele din aceste subrețele vor utiliza masca de rețea 255.255.255.224 sau /27.

Pentru a calcula numărul de hosturi, examinăm ultimul octet. După împrumutarea a 3 biți pentru subrețea, rămân 5 biți de host.

Aplicăm formula de calculare de host $2^5 = 32$, însă scădem 2 pentru adresa de rețea (ce are toți de 0 în partea de host) și adresa de broadcast (ce are toți de 1 în partea de host).

Subrețelele sunt atribuite segmentelor de rețea necesare pentru topologie, aşa cum se poate observa în Fig.9.11.D.

Folosind un plan de adresare general, prima adresă de host este atribuită interfeței routerului. Hosturile din fiecare subrețea vor utiliza adresa interfeței routerului ca adresă de default gateway.

- PC1 (192.168.1.2/27) va folosi adresa 192.168.1.1 ca adresa să de default gateway.
- PC2 (192.168.1.34/27) va folosi adresa 192.168.1.33 ca adresa să de default gateway.
- PC3 (192.168.1.98/27) va folosi adresa 192.168.1.97 ca adresa să de default gateway.
- PC4 (192.168.1.130/27) va folosi adresa 192.168.1.129 ca adresa să de default gateway.

5 Subnets Required

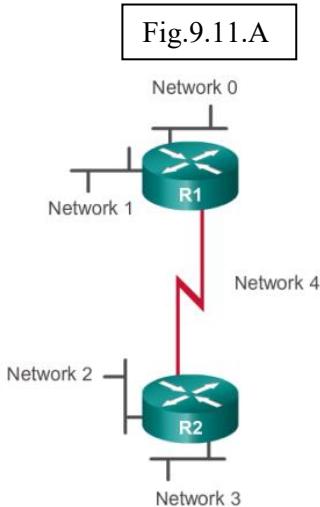
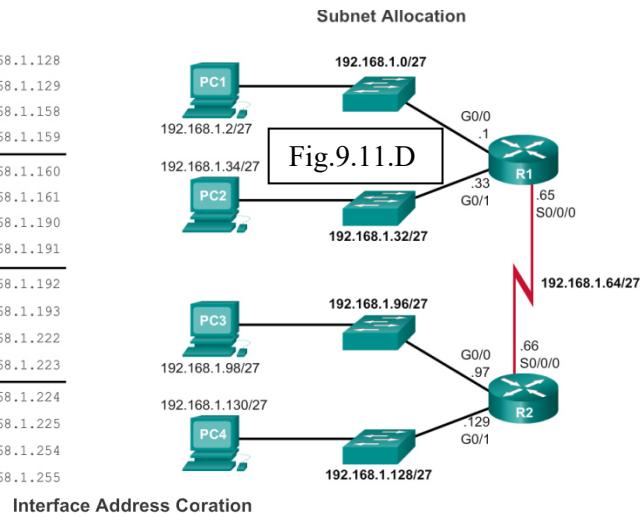


Fig.9.11.B

Net 0	Network	192.	168.	1.	000	0	0000	192.168.1.0
	First	192.	168.	1.	000	0	0001	192.168.1.1
	Last	192.	168.	1.	000	1	1110	192.168.1.30
	Broadcast	192.	168.	1.	000	1	1111	192.168.1.31
Net 1	Network	192.	168.	1.	001	0	0000	192.168.1.32
	First	192.	168.	1.	001	0	0001	192.168.1.33
	Last	192.	168.	1.	001	1	1110	192.168.1.62
	Broadcast	192.	168.	1.	001	1	1111	192.168.1.63
Net 2	Network	192.	168.	1.	010	0	0000	192.168.1.64
	First	192.	168.	1.	010	0	0001	192.168.1.65
	Last	192.	168.	1.	010	1	1110	192.168.1.94
	Broadcast	192.	168.	1.	010	1	1111	192.168.1.95
Net 3	Network	192.	168.	1.	011	0	0000	192.168.1.96
	First	192.	168.	1.	011	0	0001	192.168.1.97
	Last	192.	168.	1.	011	1	1110	192.168.1.126
	Broadcast	192.	168.	1.	011	1	1111	192.168.1.127

	Network	192.	168.	1.	100	0	0000	192.168.1.128
Net 4	First	192.	168.	1.	100	0	0001	192.168.1.129
	Last	192.	168.	1.	100	1	1110	192.168.1.158
	Broadcast	192.	168.	1.	100	1	1111	192.168.1.159
	Network	192.	168.	1.	101	0	0000	192.168.1.160
Net 5	First	192.	168.	1.	101	0	0001	192.168.1.161
	Last	192.	168.	1.	101	1	1110	192.168.1.190
	Broadcast	192.	168.	1.	101	1	1111	192.168.1.191
	Network	192.	168.	1.	110	0	0000	192.168.1.192
Net 6	First	192.	168.	1.	110	0	0001	192.168.1.193
	Last	192.	168.	1.	110	1	1110	192.168.1.222
	Broadcast	192.	168.	1.	110	1	1111	192.168.1.223
	Network	192.	168.	1.	111	0	0000	192.168.1.224
Net 7	First	192.	168.	1.	111	0	0001	192.168.1.225
	Last	192.	168.	1.	111	1	1110	192.168.1.254
	Broadcast	192.	168.	1.	111	1	1111	192.168.1.255

Fig.9.11.C



Interface Address Correlation

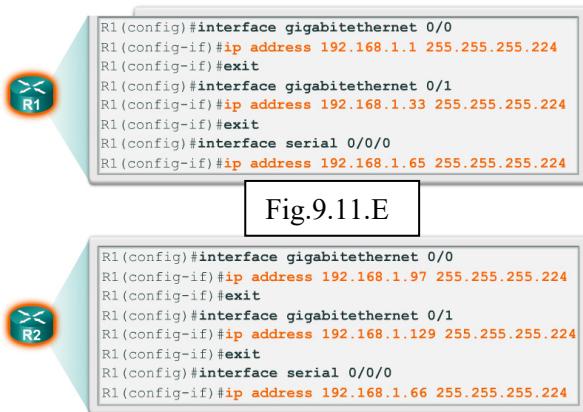


Fig.9.11.E

În exemplele anterioare, am considerat o internetwork ce necesită 3 subrețele și una ce necesită 5 subrețele. Pentru a realiza acest obiectiv de creare a patru subrețele am împrumutat doi biți din 8 biți de host disponibili dintr-o adresă IP ce are masca de rețea 255.255.255.0 sau /24. Masca de rețea rezultată a fost 255.255.255.192, iar un număr total de 4 subrețele posibile au fost create. Aplicând formula de calculare a hosturilor ($2^6 - 2$) am determinat că fiecare dintre acele 4 subrețele pot avea 62 de adrese de host atribuite nodurilor.

Pentru a obține 5 subrețele, am împrumutat 3 biți din 8 biți de host disponibili dintr-o adresă IP ce are masca de rețea 255.255.255.0 sau /24. Prin împrumutarea celor 3 biți din partea de host a adresei IP a internetwork existente. Pentru a calcula numărul de subrețele, trebuie să ne uităm la numărul de biți de host disponibili pentru a-i folosi în formula de calculare 2^x numărul de biți împrumutați - 2. Folosind adresa IP a ultimului exemplu, 192.168.10.0/24, avem 8 biți de host. Pentru a crea 100 de subrețele trebuie să împrumutăm 7 biți.

Calculăm numărul de subrețele dacă 7 biți sunt împrumutați : $2^7 = 128$ subrețele.
Însă, împrumutând 7 biți va rămâne un sigur bit de host și dacă vom aplica formula de calcul de host, va rezulta existența niciunui host în respectivele subrețele. Calculăm numărul de hosturi în cazul în care rămâne un singur bit $2^1 = 2$, apoi scădem 2 pentru adresa de rețea și cea de broadcast; rezultă 0 hosturi ($2^1 - 2 = 0$).

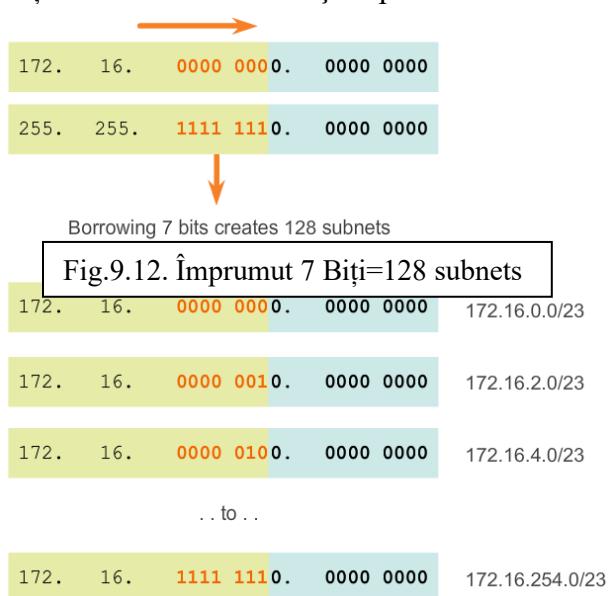
Într-o situație ce necesită un număr mai mare de subrețele, este necesară o rețea ce are mai mulți biți de host de împrumutat, cum ar fi o adresă IP cu o masca de rețea implicită /16 sau 255.255.0.0.

Adresele ce au un spațiu de adrese 128 - 191 în primul octet au o mască implicită 255.255.0.0 sau /16. Adresele din acest spațiu de adrese au 16 biți în partea de rețea și 16 în partea de host. Acești 16 biți sunt biți disponibili pentru împrumutul necesar creării de subrețele.

Utilizând o nouă adresă IP din blocul de adrese 172.16.0.0/16, trebuie să împrumutăm biți de host pentru a crea cel puțin 100 de subrețele. Începând de la stânga la dreapta, cu primul bit de host disponibil, vom împrumuta câte un bit o dată până când ajungem la numărul de biți necesari pentru a crea 100 de subrețele. Împrumutând un bit, vom crea două subrețele, împrumutând 2 biți vom crea 4 subrețele, 3 biți crează 8 subrețele și aşa mai departe. Calculând numărul de subrețele create dacă împrumutăm 7 biți cu ajutorul formulei $2^7 = 128$ de subrețele.

Împrumutând 7 biți se crează 128 de subrețele, aşa cum este evidențiat în Fig.9.12.

Reamintim faptul că masca de rețea trebuie să fie schimbată pentru a reflecta biții împrumutați. În acest exemplu, când 7 biți sunt împrumutați, masca se extinde cu 7 biți în al treilea octet. În decimal, masca este reprezentată ca fiind 255.255.254.0 sau /23, deoarece al treilea octet este 11111110 în binar și al patrulea octet este 00000000 în binar. Subnetarea se va face în al treilea octet, cu biții de host în al treilea și al patrulea octet.



Pentru a calcula numărul de hosturi, examinăm al treilea și al patrulea octet. După împrumutarea de 7 biți pentru subnet, rămân 3 biți în al treilea octet și 8 biți în al patrulea octet.

Aplicăm formula de calculare de host, aşa cum este efectuat și în Fig.9.13.

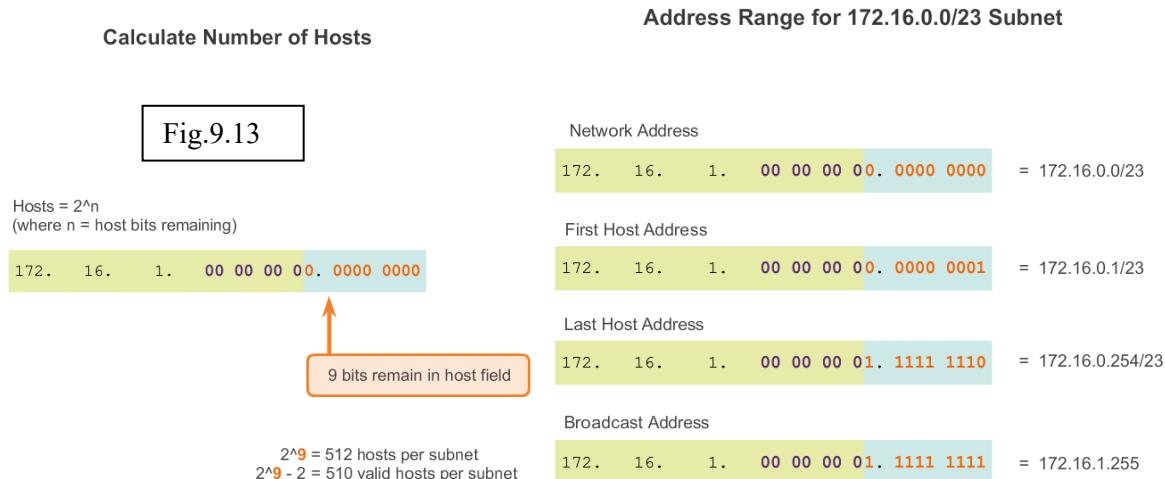
$$2^9 = 512$$

De reținut faptul că atunci când sunt toți biții de 0 în partea de host a adresei este adresa de rețea, iar când sunt toți biții de 1 în partea de host a adresei este adresa de broadcast. Prin urmare, există numai 510 adrese de host disponibile pentru fiecare subrețea.

Așa cum se poate observa și în Fig. 2, prima adresă de host pentru prima subrețea este 172.16.0.1, iar ultima adresă de host este 172.16.1.254. Amintim faptul că fiecare host trebuie să aibă o adresă IP validă din spațiul definit pentru fiecare segment. Subrețea atribuită interfeței routerului va determina cărui segment aparține un host.

Reminder:

Biți pot fi împrumutați numai din partea de host a adresei. Partea de rețea a adresei este alocată de către furnizorul de servicii și nu poate fi schimbată. Deci, organizațiile ce necesită un număr mare de subrețele trebuie să comunice nevoia lor la ISP pentru ca ISP-ul să aloce o adresă IP cu o mască implicită cu destui biți astfel încât să se poată crea subrețelele dorite.



Există unele organizații, cum ar fi furnizorii mici de servicii, ce ar putea avea nevoie de mai multe de 100 de subrețele. De exemplu, o organizație ce necesită 1000 de subrețele. Pentru a crea subrețele trebuie să împrumutăm biți din partea de host a adresei IP din internetwork existentă. Ca și înainte, pentru a calcula numărul de subrețele este necesar să ne uităm la numărul de biți de host disponibili. O astfel de situație necesită ca adresa IP atribuită de către ISP să aibă destui biți disponibili pentru calculul a 1000 de subrețele. Adresele IP ce au spațiul între 1-126 în primul octet au masca implicită 255.0.0.0 sau /8. Acest lucru înseamnă că există 8 biți în partea de rețea și 24 de biți în partea de host disponibili pentru împrumut.

Folosind blocul de adrese 10.0.0.0/8, biți de host trebuie să fie împrumutați pentru a crea cel puțin 1000 de subrețele. Începând de la stânga la dreapta, cu primul bit de host disponibil, vom împrumuta un bit o dată până când ajungem la numărul de biți necesari pentru a crea 1000 de subrețele. Prin calcularea numărului de subrețele rezultate în cazul în care 10 biți sunt împrumutați, rezultă $2^{10} = 1024$ subrețele.

Împrumutul a 10 biți rezultă 1024 subrețele, așa cum se poate vedea în Fig.9.14.

Reamintim faptul că masca de rețea trebuie să se schimbe astfel încât să reflecte biții împrumutați. În acest exemplu, când sunt împrumutați 10 biți, masca de rețea se extinde cu 10 biți în al treilea octet. În zecimal, masca este reprezentată ca fiind 255.255.192.0 sau /18, deoarece al treilea octet din masca de rețea este 11000000 în binar și al patrulea octet este 00000000 în binar. Subnetarea va fi efectuată în al treilea octet, însă să nu uităm de biții de host din al treilea și al patrulea octet.

Pentru a calcula numărul de hosturi, examinăm al treilea și al patrulea octet. După împrumutul a 10 biți pentru subnet, rămân 6 biți în al treilea octet și 8 biți în al patrulea octet, rezultând 14 biți rămași.

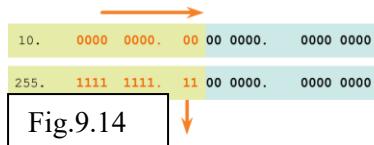
Aplicăm formula de calcul de host, așa cum este efectuată și în Fig.9.15.

$$2^{14} - 2 = 16382$$

Prima adresă de host pentru prima subrețea este 10.0.0.1 și ultima adresă de host este 10.0.63.254. De reținut faptul că fiecare host trebuie să aibă o adresă IP validă din spațiul definit pentru fiecare segment. Subrețea atribuită interfeței routeurului va determina cărui segment aparține un host.

Notă: Toate dispozitivele din aceeași subrețea vor avea o adresă de host IPv4 din spațiul de adrese de host și vor utiliza aceeași mască de rețea.

Calculate Number of Hosts



Borrowing 10 bits creates 1024 subnets

10. 0000 0000. 00 00 0000. 0000 0000	10.0.0.0/18
10. 0000 0000. 01 00 0000. 0000 0000	10.0.64.0/18
10. 0000 0000. 11 00 0000. 0000 0000	10.0.192.0/18
10. 0000 0001. 00 00 0000. 0000 0000	10.1.0.0/18
.. to ..	
10. 1111 1111. 10 00 0000. 0000 0000	10.255.128.0/18

Address Range for 10.0.0.0/18 Subnet

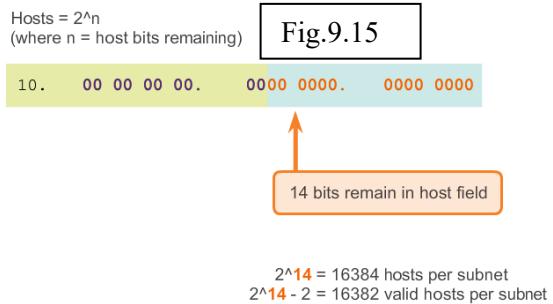


Fig.9.15

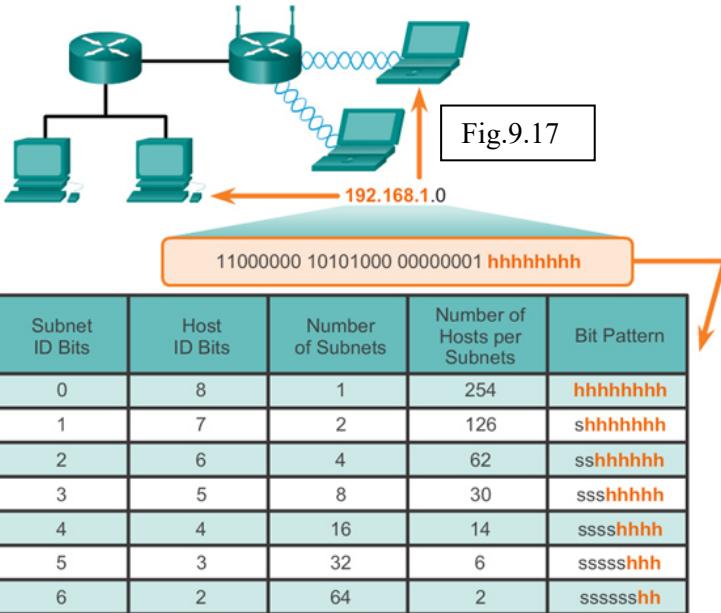
Network Address	Fig.9.16
10. 00 00 00 00. 0000 0000. 0000 0000	= 10.0.0.0/18
First Host Address	
10. 00 00 00 01. 0000 0000. 0000 0001	= 10.0.0.1/18
Last Host Address	
10. 00 00 00 0F. 0011 1111. 1111 1110	= 10.0.63.254/18
Broadcast Address	
10. 00 00 00 10. 0011 1111. 1111 1111	= 10.0.63.255/18

9.1.3 Determinarea Măștii de Rețea

Decizia cu privire la câți biți de host să împrumutăm pentru a crea subrețele este o decizie importantă de planificare. Există două considerații atunci când planificăm subrețelele: numărul de adrese de host necesare pentru fiecare rețea și numărul de subrețele individuale necesare. Fig. arată posibilitatiile de subnetare pentru rețeaua 192.168.1.0. Selectia numărului de biți pentru subnetID afectează atât numărul de posibile subrețele cât și numărul de adrese de host din fiecare subrețea.

De remarcat faptul că este o relație inversă între numărul de subrețele și numărul de hosturi. Cu cât sunt împrumutați mai mulți biți pentru crearea de subrețele, cu atât rămân mai puțini biți de host disponibili; prin urmare, mai puține hosturi pe subrețea. Dacă sunt necesare mai multe adrese, sunt necesari mai mulți biți, având ca rezultat mai puține subrețele.

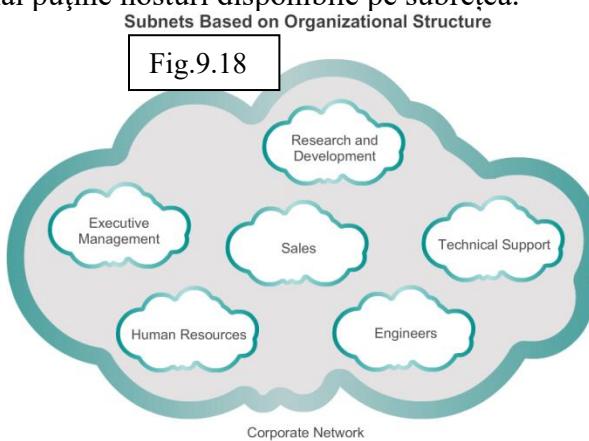
Numărul de Hosturi – Atunci când împrumutăm biți pentru a crea mai multe subrețele, pastrăm destui biți de host pentru subrețeaua mai mare. Numărul de adrese de host necesare în subrețeaua mai mare va determina câți biți trebuie să rămână în partea de host. Formula 2^n (unde n este numărul de biți de host rămasi) este folosită pentru a calcula câte adrese vor fi disponibile pe fiecare subrețea. Reamintim faptul că două dintre aceste adrese nu pot fi folosite, deci numărul utilizabil de adrese se calculează ca $2^n - 2$.



Uneori un anumit număr de subrețele este necesar cu accent mai redus pe numărul de adrese de host pe subrețea. Acest lucru ar putea fi cazul în care o organizație alege să separe traficul de rețea în funcție de structura internă sau de departament. De exemplu, o organizație ar putea alege să pună toate dispozitivele de host utilizate de către angajați din departamentul de inginerie într-o rețea și toate dispozitivele de host folosite de management într-o rețea separată. În acest caz, numărul de subrețele este important în determinarea numărului de biți împrumutați.

Reamintim faptul că numărul de subrețele create atunci când biții sunt împrumutați poate fi calculat cu ajutorul formulei 2^n (unde n este numărul de biți împrumutați). Nu există nevoie de scădere a vreunei rețele rezultante deoarece sunt toate utilizabile.

Cheia este echilibrarea numărului de subrețele necesare și numărului de hosturi necesare pentru cea mai mare subrețea. Mai mulți biți împrumutați pentru crearea de subrețele suplimentare înseamnă mai puține hosturi disponibile pe subrețea.

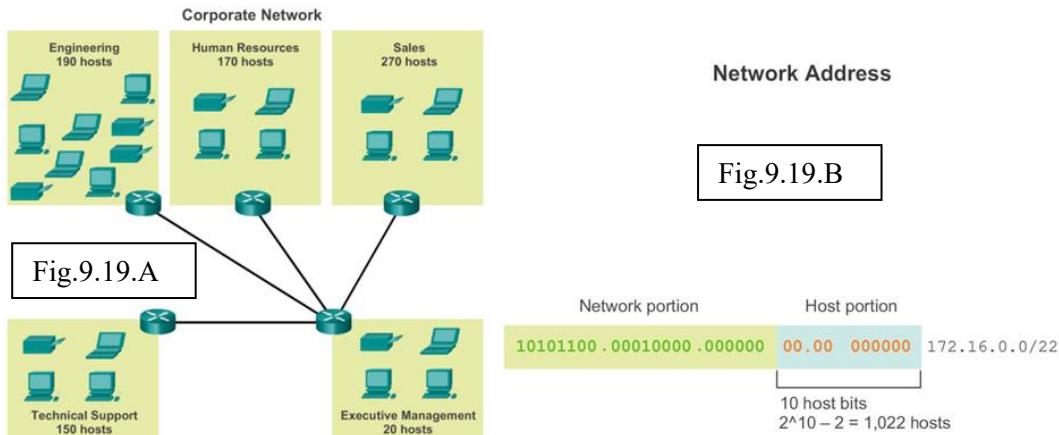


Orice rețea dintr-o organizație este proiectată pentru a suporta un număr finit de hosturi. Subnetarea de bază prevede suficiente subrețele pentru a adapta rețelele și pentru a oferi suficiente adrese de host pe subrețea.

Unele rețele, cum ar fi legăturile point-to-point WAN, necesită numai două hosturi. Alte rețele, cum ar fi un LAN dintr-o clădire mare sau departamentar ar putea necesita să susțină sute de hosturi. Administratorii de rețea trebuie să elaboreze schema de adresare internetwork pentru a fixa numărul de hosturi maxim pentru fiecare rețea.

Determinarea Numărului Total de Hosturi – Mai întâi, luăm în considerare numărul total de hosturi necesare în întreaga internetwork corporativă. Un bloc de adrese destul de mare trebuie să fie folosit pentru a cuprinde toate dispozitivele din toate rețelele corporate. Aceste dispozitive includ dispozitivele de utilizator, servere, dispozitive intermediare și interfețele de router.

Considerăm exemplul unei internetwork corporativ ce trebuie să cuprindă un număr total de 800 de hosturi în cinci locații - Fig.9.19.A. În acest exemplu, furnizorul de servicii a alocat următoarea adresă de rețea 172.16.0.0/22 (10 biți de host). Așa cum se poate observa și în Fig.9.19.B, aceasta va oferi 1.022 adrese de host ce cuprind mai mult decât nevoile de adresare pentru această internetwork.



Determinarea Numărului și Dimensiunii Rețelelor – Apoi, considerăm numărul de subrețele necesare și numărul de adrese de host necesare pentru fiecare subrețea. Având în vedere topologia de rețea ce cuprinde 5 segmente de LAN și 4 conexiuni internetwork între routere, sunt necesare 9 subrețele. Cea mai mare subrețea necesită 40 de hosturi. Atunci când proiectăm o schemă de adresare, trebuie să anticipăm creșterea în ambele domenii - numărul de subrețele și numărul de hosturi pe subrețea.

Adresa de rețea 172.16.0.0/22 are 10 biți de host. Deoarece cea mai mare subrețea necesită 40 de hosturi, un minim de 6 biți de host trebuie să fie împrumutați. Acest lucru este determinat de formula $2^6 - 2 = 62$ hosturi. Cei 4 biți rămași pot fi folosiți pentru a aloca subrețele. Folosind formula de determinare a subrețelelor, rezultă 16 subrețele $2^4 = 16$. Deoarece internetwork din exemplu necesită 9 subrețele, îndeplinim cerințele și permite o anumită creștere ulterioară.

Atunci când sunt împrumutăți 4 biți nouă mască de rețea este 255.255.255.192 sau /26.

Așa cum se poate observa și în Fig.9.20.A, folosind o lungime de prefix /26, pot fi determinate 16 adrese de subrețea. Numai partea de subnet a adresei este incrementată. Cei 22 de biți originali ai adresei de rețea nu se pot schimba și partea de host va conține numai biți de 0.

Notă: De remarcat faptul că deoarece partea de subrețea se află în octetul trei și patru, una dintre aceste două valori va oscila în adresele de subrețea.

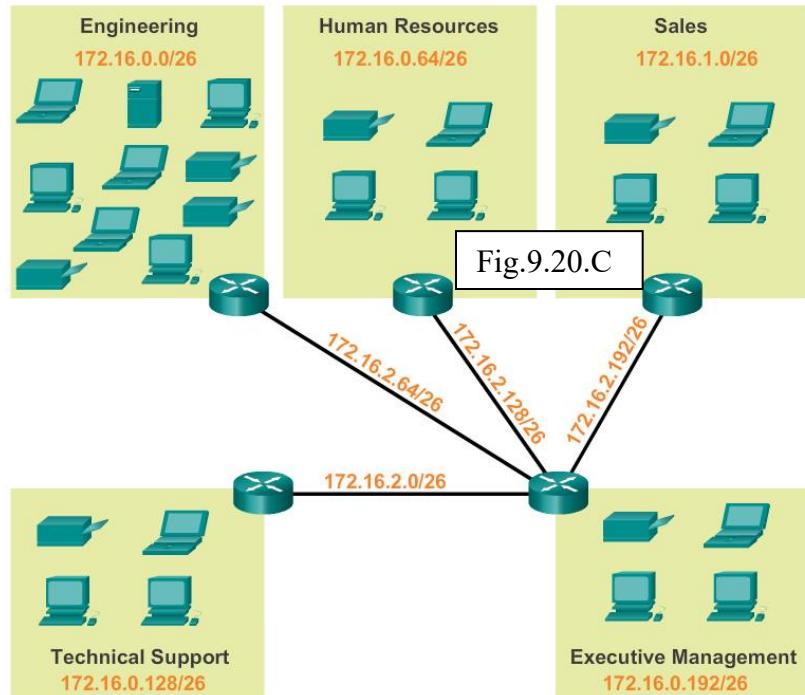
Așa cum se poate observa și în Fig.9.20.B, rețeaua originală 172.16.0.0/22 era o singură rețea cu 10 biți de host ce oferea 1.022 adrese utilizabile pentru atribuirea hosturilor. Prin împrumutarea a 4 biți, 16 subrețele (de la 0000 la 1111) pot fi create. Fiecare subrețea are 6 biți de host sau 62 de adrese de host utilizabile pe subrețea.

Așa cum se poate observa și în Fig.9.20.C, subrețelele pot fi atribuite segmentelor de LAN și conexiunilor router-to-router.

Subnets and Addresses

Fig.9.20.B

Subnet Scheme				Subnets and Addresses			
Fig.9.20.A							
10101100.00010000.00000000 00.00 000000 172.16.0.0/22				0	10101100.00010000.00000000 00.00 000000 172.16.0.0/26		
1	10101100.00010000.00000000 00.01 000000 172.16.0.64/26	1	10101100.00010000.00000000 00.10 000000 172.16.0.128/26	2	10101100.00010000.00000000 00.11 000000 172.16.0.192/26	3	10101100.00010000.00000000 01.00 000000 172.16.1.0/26
2	10101100.00010000.00000000 00.10 000000 172.16.0.128/26	4	10101100.00010000.00000000 01.01 000000 172.16.1.64/26	5	10101100.00010000.00000000 01.10 000000 172.16.1.128/26	6	10101100.00010000.00000000 01.10 000000 172.16.1.128/26
3	10101100.00010000.00000000 00.11 000000 172.16.0.192/26						
4	10101100.00010000.00000000 01.00 000000 172.16.1.0/26						
5	10101100.00010000.00000000 01.01 000000 172.16.1.64/26						
6	10101100.00010000.00000000 01.10 000000 172.16.1.128/26						
Nets 7 – 14 not shown				Nets 7 – 14 not shown			
15	10101100.00010000.00000000 11.10 000000 172.16.3.128/26	15	10101100.00010000.00000000 11.10 000000 172.16.3.128/26	16	10101100.00010000.00000000 11.11 000000 172.16.3.192/26		
16	10101100.00010000.00000000 11.11 000000 172.16.3.192/26						
4 bits borrowed from host portion to create subnets				2^4 = 16 subnets 2^6 - 2 = 62 hosts per subnet			
172.16.0.0/22							



9.1.4 Beneficiile Maștilor cu Lungime Variabilă (Variable Length Subnet Masking-VLSM)

Folosind subnetarea tradițională, același număr de adrese este alocat pentru fiecare subrețea. Dacă toate subretelele au aceleași cerințe cu privire la numărul de hosturi, aceste blocuri de adrese de dimensiune fixă sunt eficiente. Însă, adesea nu se întâmplă aşa.

De exemplu, topologia din Fig.9.21.A necesită șapte subretele, una pentru fiecare dintre cele patru LANuri și una pentru fiecare dintre cele trei conexiuni WAN dintre routere. Folosind subnetarea tradițională cu adresa dată 192.168.20.0/24, 3 biți pot fi împrumutați din partea de host a ultimului octet pentru a îndeplini cerința de subnetare pentru șapte subretele. Așa cum se poate observa în Fig.9.21.B, împrumutarea a trei biți crează 8 subretele cu 5 biți de host și 30 de hosturi utilizabile pe subrețea. Această schemă crează subretelele necesare și îndeplinește cerința de host pentru cel mai mare LAN.

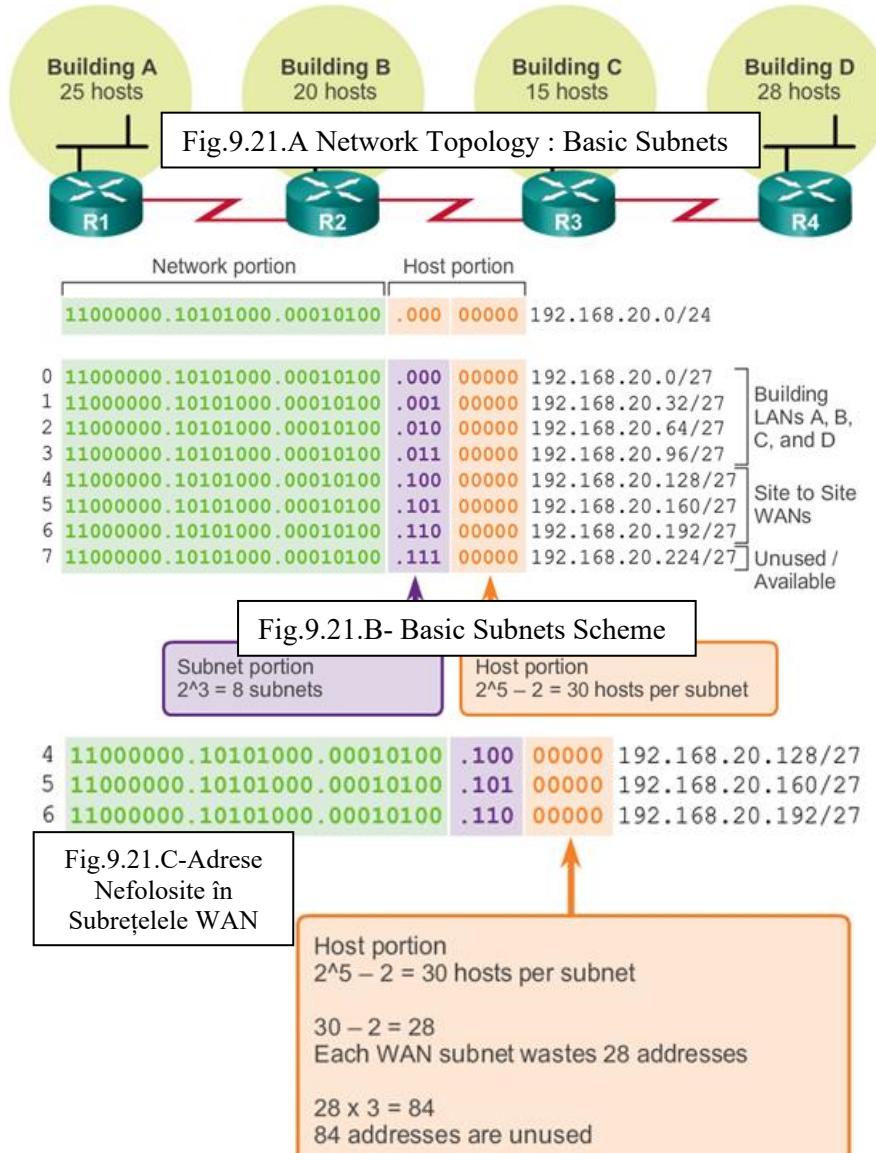
Deși această subnetare tradițională îndeplinește cerințele celor mai mari LAN și împarte spațiul de adrese într-un număr adecvat de subrețele, are ca rezultat o mare pierdere de adrese utilizabile.

De exemplu, numai două adrese sunt necesare pentru fiecare subrețea dintre cele trei legături LAN. Deoarece fiecare subrețea are 30 de adrese utilizabile rămân 28 de adrese în fiecare dintre aceste subrețele. Așa cum se poate observa în Fig.9.21.C, rezultă 84 de adrese nefolosite (28×3).

Mai mult, acest lucru limitează creșterea viitoare prin reducerea numărului total de subrețele disponibile. Această utilizare ineficientă a adreselor este caracteristică subnetării tradiționale ale rețelelor de tip classful.

Aplicarea unei scheme de subnetare tradițională la acest scenariu nu este foarte eficientă și este risipitoare. De fapt, acest exemplu este un bun model de exemplificare a modului în care subnetarea poate fi folosită pentru maximizarea utilizării de adrese.

Subnetarea unui rețele, sau folosirea Variable Length Subnet Mask (VLSM), a fost concepută pentru a evita pierderea de adrese.

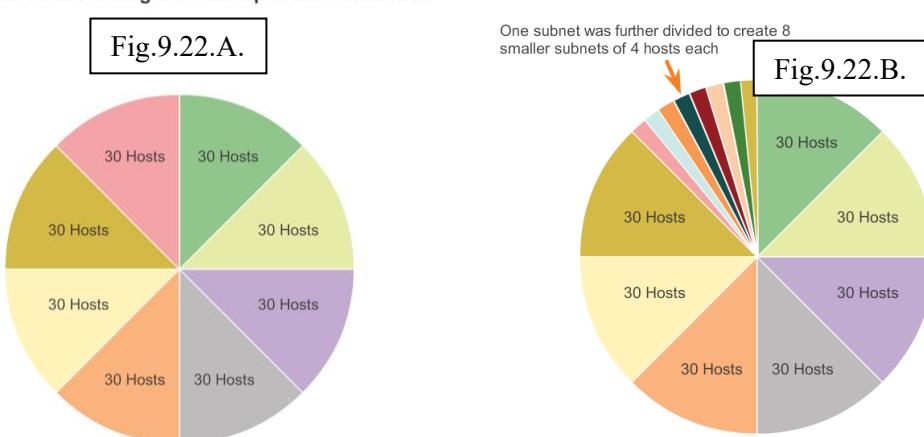


În toate exemplele anterioare de subnetare, remarcăm faptul că aceeași mască de subrețea a fost aplicată tuturor subrețelelor. Acest lucru înseamnă că fiecare subrețea are același număr de adrese de host disponibile.

Așa cum se ilustrează în Fig.9.22.A, subnetarea tradițională crează subrețele de dimensiuni egale. Fiecare subrețea dintr-o schemă tradițională folosește aceeași mască de rețea. Așa cum este evidențiat în Fig.9.22.B, VLSM permite ca un spațiu de rețea să fie divizat în părți inegale. Cu VLSM, masca de rețea variază în funcție de câți biți trebuie să fie împrumutați pentru o anumită subrețea, astfel rezultând partea "variabilă" a VLSM.

Subnetarea cu VLSM este similară cu cea tradițională în care biții sunt împrumutați pentru a crea subrețele. Formulele de calcul a numărului de hosturi pe subrețea și a numărului de subrețele create se aplică și aici. Diferența este că această subnetare nu este o activitate "single pass". Cu VLSM, rețeaua este subnetată mai întâi și apoi subrețelele sunt subnetate din nou. Acest proces poate fi repetat de mai multe ori pentru a crea subrețele de dimensiuni diferite.

Traditional Subnetting Creates Equal Sized Subnets Subnets of Varying Sizes



Pentru a înțelege mai bine procesul VLSM, să ne întoarcem la exemplul anterior.

În exemplul anterior, prezentat în Fig.9.23.A, rețeaua 192.168.20.0/24 a fost subnetată în 8 subrețele de dimensiuni egale; șapte din 8 au fost alocate. Patru subrețele au fost folosite pentru LANuri și trei pentru conexiunile WAN dintre routere. Reamintim faptul că risipa spațiului de adrese a fost în subrețelele folosite pentru conexiunile WAN deoarece acele subrețele necesitau numai două adrese utilizabile: una pentru fiecare interfață a routerului. Pentru a evita această risipă, VLSM poate fi folosit pentru a crea subrețele mai mici pentru conexiunile WAN.

Pentru a crea subrețele mai mici pentru conexiunile WAN, una dintre subrețele va fi divizată. În Fig.9.23.B, ultima subrețea 192.168.20.224/27, va fi subnetată mai departe.

Reamintim că atunci când numărul de adrese de host necesare este cunoscut, formula $2^n - 2$ (unde n este egal cu numărul de biți de host rămas) poate fi folosită. Pentru a oferi două adrese utilizabile, trebuie să rămână 2 biți de host în partea de host.

$$2^2 - 2 = 2$$

Deoarece există 5 biți de host în spațiul de adrese 192.168.20.224/27, 3 biți pot fi împrumutați, rămânând 2 biți în partea de host.

Calculele din acest moment sunt exact la fel cu cele folosite pentru subnetarea tradițională. Biții sunt împrumutați și spațiile de adrese pentru subrețele sunt determinate.

Așa cum se evidențiază în Fig.9.23.B, schema de subnetare VLSM reduce numărul de adrese pe subrețea la o dimensiune adecvată pentru WANuri. Subnetizarea subrețelei 7 pentru WANuri permite ca subrețelele 4, 5, 6 să fie disponibile pentru rețele viitoare și multe alte subrețele disponibile pentru WANuri.

	11000000.10101000.00010100.00000000	192.168.20.0/24	
0	11000000.10101000.00010100.00000000	192.168.20.0/27	
1	11000000.10101000.00010100.00100000	192.168.20.32/27	LANs A, B, C, D
2	11000000.10101000.00010100.01000000	192.168.20.64/27	
3	11000000.10101000.00010100.01100000	192.168.20.96/27	
4	11000000.10101000.00010100.10000000	192.168.20.128/27	Unused / Available
5	11000000.10101000.00010100.10100000	192.168.20.160/27	
6	11000000.10101000.00010100.11000000	192.168.20.192/27	
7	11000000.10101000.00010100.11100000	192.168.20.224/27	

Fig.9.23.A

	11000000.10101000.00010100.00000000	192.168.20.0/24	
0	11000000.10101000.00010100.00000000	192.168.20.0/27	
1	11000000.10101000.00010100.00100000	192.168.20.32/27	LANs A, B, C, D
2	11000000.10101000.00010100.01000000	192.168.20.64/27	
3	11000000.10101000.00010100.01100000	192.168.20.96/27	
4	11000000.10101000.00010100.10000000	192.168.20.128/27	
5	11000000.10101000.00010100.10100000	192.168.20.160/27	Unused / Available
6	11000000.10101000.00010100.11000000	192.168.20.192/27	
7	11000000.10101000.00010100.11100000	192.168.20.224/27	

Fig.9.23.B Schema de Subnetare VLSM

Folosind VLSM, segmentele LAN și WAN pot fi adresate fără nici-o “risipă” inutilă.

Hosturile din fiecare dintre LANuri vor fi atribuite cu o adresă de host validă din spațiul de adresare pentru respectiva subrețea și vor avea masca /27. Fiecare dintre cele patru routere vor avea o interfață LAN cu o subrețea /27 și una sau mai multe interfețe seriale cu o subrețea /30.

Folosind o schema de adresare generală, prima adresă de host IPv4 a fiecărei subrețele este atribuită interfeței LAN a routerului. Interfețele WAN ale routerelor sunt asignate cu adrese IP și masca /30.

Fig.9.24 și respective Fig.9.25 arată configurația de interfață pentru fiecare dintre routere.

Hosturile din fiecare subrețele vor avea o adresă de host IPv4 din spațiul de adrese de host pentru respectiva subrețea și o mască de rețea adecvată. Hosturile vor folosi adresa interfeței routerului LAN atașat ca adresă de default gateway.

- Hosturile din clădirea A(192.168.20.0/27) vor folosi adresa 192.168.20.1 ca adresă de default gateway.
- Hosturile din clădirea B(192.168.20.32/27) vor folosi adresa 192.168.20.33 ca adresă de default gateway.
- Hosturile din clădirea C(192.168.20.64/27) vor folosi adresa 192.168.20.65 ca adresă de default gateway.
- Hosturile din clădirea D(192.168.20.96/27) vor folosi adresa 192.168.20.97 ca adresă de default gateway.

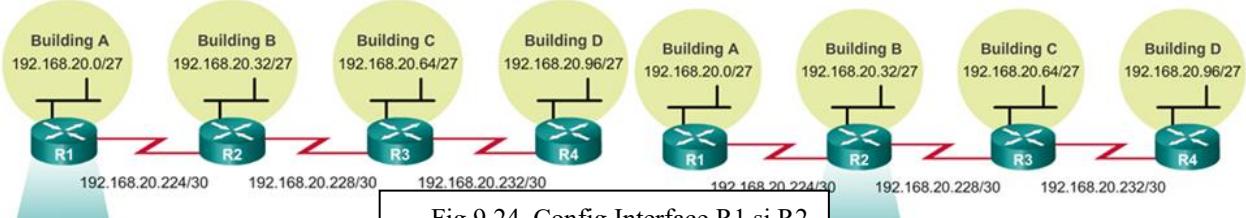


Fig.9.24. Configurație R1 și R2

```
R1(config)#interface gigabitethernet 0/0
R1(config-if)#ip address 192.168.20.1 255.255.255.224
R1(config-if)#exit
R1(config)#interface serial 0/0/0
R1(config-if)#ip address 192.168.20.225 255.255.255.252
R1(config-if)#end
R1#
```

```
R2(config)#interface gigabitethernet 0/0
R2(config-if)#ip address 192.168.20.33 255.255.255.224
R2(config-if)#exit
R2(config)#interface serial 0/0/0
R2(config-if)#ip address 192.168.20.226 255.255.255.252
R2(config-if)#exit
R2(config)#interface serial 0/0/1
R2(config-if)#ip address 192.168.20.229 255.255.255.252
R2(config-if)#end
R2#
```

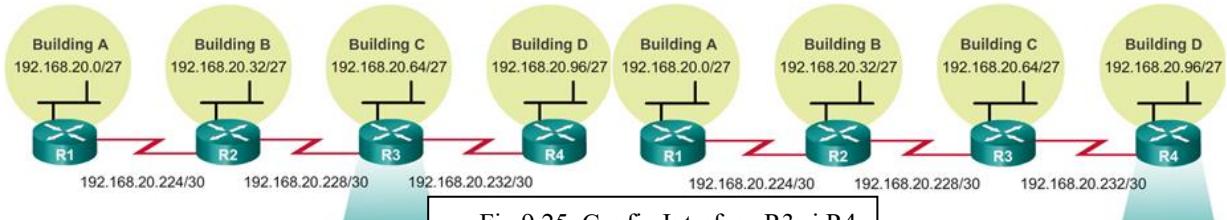


Fig.9.25. Config Interface R3 si R4

```
R3(config)#interface gigabitethernet 0/0
R3(config-if)#ip address 192.168.20.65 255.255.255.224
R3(config-if)#exit
R3(config)#interface serial 0/0/0
R3(config-if)#ip address 192.168.20.230 255.255.255.252
R3(config-if)#exit
R3(config)#interface serial 0/0/1
R3(config-if)#ip address 192.168.20.233 255.255.255.252
R3(config-if)#end
R3#

```

```
R4(config)#interface gigabitethernet 0/0
R4(config-if)#ip address 192.168.20.97 255.255.255.224
R4(config-if)#exit
R4(config)#interface serial 0/0/0
R4(config-if)#ip address 192.168.20.234 255.255.255.252
R4(config-if)#end
R4#

```

Planificarea de adrese poate fi de asemenea îndeplinită cu ajutorul unei varietăți de instrumente. O metodă este folosirea diagramei VLSM pentru identificarea a ce blocuri de adrese sunt disponibile pentru utilizare și care sunt deja atribuite. Acest lucru ajută la prevenirea atribuirii adreselor ce au fost deja alocate. Folosind rețea din exemplul anterior, diagrama VLSM poate fi folosită pentru planificarea atribuirii de adrese.

Examinarea Subrețelelor cu masca /27 – Așa cum se poate vedea în Fig.9.26, atunci când folosim subnetarea tradițională primele șapte blocuri de adrese au fost alocate pentru LANuri și WANuri. Reamintim faptul că schema prezintă 8 subrețele cu 30 de adrese utilizabile (/27). Deși această schemă funcționa pentru segmentele LAN, există o mare pierdere de adrese în segmentele WAN.

Atunci când proiectăm o schemă de adresare pe o rețea nouă, blocurile de adrese pot fi atribuite într-un mod ce minimizează pierderea și păstrează blocuri neutilizate de adrese continue.

Asignarea Blocurilor de Adrese cu VLSM – Așa cum se poate observa în Fig.9.27, pentru folosirea spațiului de adrese cât mai eficient, sunt create subrețele /30 pentru legăturile WAN. Pentru a păstra blocurile de adrese nefolosite împreună, ultima subrețea /27 a fost subnetată la rândul ei în subrețele /30. Primele trei subrețele au fost atribuite legăturilor WAN.

- .224 /30 spațiul de adrese de host de la 225 la 226: legătura WAN dintre R1 și R2.
- .228 /30 spațiul de adrese de host de la 229 la 230: legătura WAN dintre R2 și R3.
- .232 /30 spațiul de adrese de host de la 233 la 234: legătura WAN dintre R3 și R4.
- .236 /30 spațiul de adrese de host de la 237 la 238: disponibile pentru utilizare.
- .240 /30 spațiul de adrese de host de la 241 la 242: disponibile pentru utilizare.
- .244 /30 spațiul de adrese de host de la 245 la 246: disponibile pentru utilizare.
- .248 /30 spațiul de adrese de host de la 249 la 250: disponibile pentru utilizare.
- .252 /30 spațiul de adrese de host de la 253 la 254: disponibile pentru utilizare.

Proiectarea schemei de adresare în acest fel lasă 3 subrețele neutilizate /27 și 5 subrețele neutilizate /30.

Basic Subnetting of 192.168.20.0/24

Fig.9.26

	/27 Network	Hosts
Building A	.0	.1 - .30
Building B	.32	.33 - .62
Building C	.64	.65 - .94
Building D	.96	.97 - .126
WAN R1 – R2	.128	.129 - .158
WAN R2 – R3	.160	.161 - .190
WAN R3 – R4	.192	.193 - .222
Unused	.224	.225 - .254

VLSM Subnetting of 192.168.20.0/24

	/27 Network	Hosts
Bldg A	.0	.1 - .30
Bldg B	.32	.33 - .62
Bldg C	.64	.65 - .94
Bldg D	.96	.97 - .126
Unused	.128	.129 - .158
Unused	.160	.161 - .190
Unused	.192	.193 - .222
	.224	.225 - .254

	/30 Network	Hosts
WAN R1-R2	.224	.225 - .226
WAN R2-R3	.228	.229 - .230
WAN R3-R4	.232	.233 - .234
Unused	.236	.237 - .238
Unused	.240	.241 - .242
Unused	.244	.245 - .246
Unused	.248	.249 - .250
Unused	.252	.253 - .254

9.2 Scheme de Adresare – Proiectarea Structurii

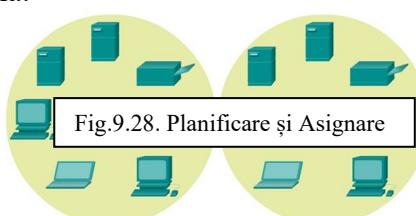
Așa cum se poate observa și în Fig.9.28, alocarea spațiului de adrese de la nivelul rețea dintr-o rețea corporativă trebuie să fie bine concepută. Atribuirea de adrese nu trebuie să fie aleatoare. Există trei principale considerații atunci când vine vorba de planificarea de alocare a adreselor.

- **Prevenirea duplicării de adrese** – *Fiecare host dintr-o internetwork trebuie să aibă o adresă IP unică. Fără o planificare și documentare adecvată, o adresă poate fi atribuită la mai mult de un host, rezultând probleme de acces pentru ambele hosturi.*
- **Asigurarea și controlul accesului** – *Unele hosturi, cum ar fi serverele, oferă resurse hosturilor interne, cât și celor externe. Adresa de nivel 3 atribuită unui server poate fi folosită pentru controlul accesului la serverul respectiv. Însă, dacă adresa este atribuită aleator și nu este bine documentată, controlul accesului este mai dificil.*
- **Monitorizarea securității și performanței** – În mod similar, securitatea și performanța hosturilor de rețea și a rețelei întregi trebuie să fie monitorizate. Ca parte a procesului de monitorizare, traficul de rețea este examinat pentru adresele ce generează sau primesc pachete în mod excesiv. Dacă există o planificare și documentare a adresării de rețea, dispozitivele de rețea problematice pot fi găsite ușor.

Asignarea Adresării într-o Rețea – Într-o rețea, există diferite tipuri de dispozitive, cum ar fi:

- Utilizatori finali.
- Servere și periferice.
- Hosturile accesibile din Internet.
- Dispozitive intermediare.
- Gateway.

La elaborarea unei scheme de adresare IP, este recomandată în general să existe un pattern a modului în care adresele sunt alocate pentru fiecare tip de dispozitiv. Acest lucru aduce beneficii administratorilor atunci când adaugă și scot dispozitive, filtrează traficul în funcție de IP, dar și simplifică documentația.



Un plan de adresare de rețea ar putea include diferite spații de adrese din fiecare subrețea, pentru fiecare tip de dispozitiv.

Adresele pentru Clienți – Din cauza provocărilor asociate cu managementul adreselor statice, dispozitivele utilizatorilor finali sunt adesea atribuite cu adrese în mod dinamic, folosind Dynamic Host Configuration Protocol (DHCP). DHCP este în general metoda preferată de atribuire a adreselor IP hosturilor din rețelele mari deoarece reduce sarcina personalului ce administrează rețeaua și elimină virtual erorile de introducere.

Un alt beneficiu al DHCP este că o adresă nu este atribuită permanent unui host, ci este numai “închiriată” pentru o perioadă de timp. Dacă trebuie să schimbăm schema de subnetare din rețeaua noastră, nu trebuie să reatribuim static adresele de host individuale. Cu DHCP, trebuie numai să reconfigurăm serverul DHCP cu noile informații. După realizarea acestui lucru, hosturile trebuie doar să își reînnnoiască automat cererile de adresare IP.

Adresele pentru Servere și Periferice – Orice resursă de rețea, cum ar fi un server sau o imprimantă, ar trebui să aibă o adresă IP statică, așa cum se vede și în Fig. . Hosturile client accesează aceste resurse folosind adresele IP ale dispozitivelor. Prin urmare, adresele IP previzibile pentru fiecare dintre aceste servere și periferice sunt necesare.

Servelele și perifericele sunt un punct central pentru traficul de rețea. Există mai multe pachete trimise la și de la adresele IPv4 ale acestor dispozitive. La monitorizarea traficului de rețea cu un utilitar precum Wireshark, un administrator de rețea ar trebui să fie capabil să identifice rapid aceste dispozitive. Folosind un sistem de numerotare consistent pentru aceste dispozitive face identificarea mai ușoară.

Adresele pentru Hosturile care sunt Accessibile din Internet – În cele mai multe internetworkuri, doar câteva dispozitive sunt accesibile de hosturile din exteriorul întreprinderii. În cea mai mare parte, aceste dispozitive sunt servere de unele tipuri. Pe toate dispozitivele dintr-o rețea care oferă resurse de rețea, adresele IP ar trebui să fie statice.

În cazul serverelor accesibile prin Internet, fiecare dintre acestea trebuie să aibă o adresă din spațiul public asociat. În plus, variații în adresa a unui astfel de dispozitiv îl va face inaccesibil din Internet. În multe cazuri, aceste dispozitive sunt pe o rețea cu adrese private. Acest lucru înseamnă că routerul sau firewallul din perimetru rețelei trebuie să fie configurați să traducă adresa internă a serverului într-o adresă publică. Datorită acestei configurații suplimentare în dispozitivul intermediar, este mai important ca acest dispozitiv să aibă o adresă previzibilă.

Adresele pentru Echipamentele Intermediare – Dispozitivele intermediare sunt de asemenea un punct de concentrare pentru traficul de rețea. Aproape tot traficul din sau dintre rețele este pasat prin intermediul dispozitivelor intermediare. Prin urmare, aceste dispozitive de rețea oferă o locație potrivită pentru managementul, monitorizarea și securitatea rețelei.

Dispozitivele intermediare au asignate adrese de nivel 3, fie pentru managementul de dispozitiv, fie pentru funcționarea lor. Dispozitivele, cum ar fi huburi, switchuri și puncte de acces wireless, nu necesită adrese IPv4 pentru a funcționa ca dispozitive intermediare. Însă, dacă trebuie să le accesez pentru configurație, monitorizare sau funcția de depanare, ele trebuie să fie asignate cu adrese.

Deoarece trebuie să știm cum să comunicăm cu dispozitivele intermediare, ele ar trebui să aibă adrese previzibile. Prin urmare, adresele lor sunt în mod normal atribuite manual. În plus, adresele acestor dispozitive ar trebui să fie dintr-un spațiu de adrese diferit din blocul de rețea decât adresele dispozitivelor de utilizator.

Adresele pentru Gateway (Routere și Firewalluri) – Spre deosebire de dispozitivele intermediare menționate, routerele și dispozitivele firewall au o adresă IP atribuită pe fiecare interfață. Fiecare interfață se află pe o rețea diferită și servește drept gateway pentru hosturile din respectiva rețea. În mod normal, interfața routerului folosește fie cea mai mare adresă , fie cea

mai mică adresă din rețea. Această atribuire ar trebui să fie uniformă peste toate rețelele din întreprindere astfel încât personalul să știe întotdeauna gatewayul rețelei, indiferent de rețea.

Interfețele routerelor și a firewallurilor sunt un punct important în traficul ce intră și ieșe din rețea. Deoarece hosturile din fiecare rețea folosesc o interfață de router sau dispozitiv firewall drept gateway din rețea, multe pachete trec prin aceste interfețe. Prin urmare, aceste dispozitive joacă un rol important în securitatea rețelei prin filtrarea pachetelor în funcție de adresele IP sursă/destinație. Gruparea diferitelor tipuri de dispozitive în grupuri de adresare logice face ca atribuirea și funcționarea filtrării de pachete să fie mai eficientă.

Network: 192.168.1.0/24		
Use	First	Last
Host Devices	.1	.229
Servers	.230	.239
Printers	.240	.249
Intermediary Devices	.250	.253
Gateway (router LAN interface)	.254	

Fig.9.30 Spațiu de Adresare IP.

9.3 Considerații pentru Proiectarea IPv6

9.3.1 Subnetarea unei Rețele IPv6

Subnetarea IPv6 necesită o abordare diferită față de subnetarea IPv4. Principalul motiv este acela că în IPv6 există foarte multe adrese, motiv pentru care subnetarea este complet diferită. Un spațiu de adrese IPv6 nu este subnetat pentru a conserva adrese; mai degrabă, este subnetat pentru suportul designului logic ierarhic al rețelei. Pe când subnetarea IPv4 este pentru gestionarea deficitului de adrese, subnetarea IPv6 este pentru construirea unei ierarhii de adresare în funcție de numărul de routere și de rețele suportate.

Reamintim faptul că un bloc de adrese IPv6 cu un prefix /48 are 16 biți pentru subnet ID, așa cum se poate observa și în Fig.9.31A. Subnetarea utilizând subnet ID de 16 biți conduce la 65.536 /64 posibile subrețele și nu necesită împrumutul niciunui bit din ID-ul de interfață sau din partea de host a adresei. Fiecare subrețea IPv6 /64 conține aproximativ 18 adrese quintillion (o mie la puterea 6), în mod evident, mai mult decât va fi vreodată nevoie într-un singur segment de rețea IP.

Subrețelele create din subnet ID sunt ușor de reprezentat deoarece nu este necesară conversia în binar. Pentru a determina următoarea subrețea disponibilă, doar numărăm în hexazecimal. Așa cum se vede în Fig.9.31.B, acest lucru înseamnă numărarea în hexazecimal în partea de subnet ID.

Prefixul de routare global este același pentru toate subrețele. Numai cvartetul subnet ID este incrementat pentru fiecare subrețea.

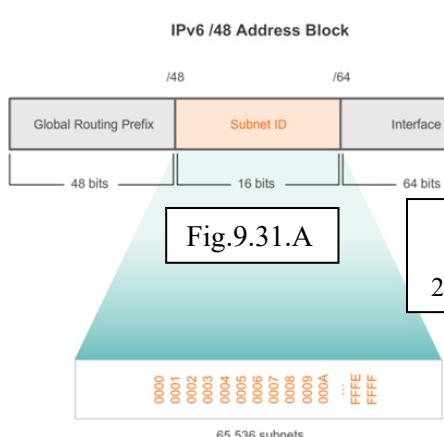
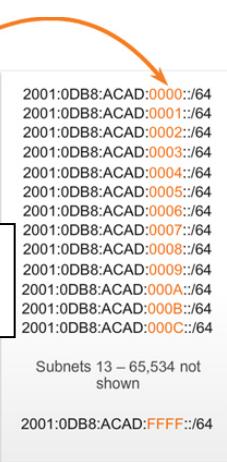


Fig.9.31.B Blocul de Adrese pentru 2001:0BB8:ACAD:/48



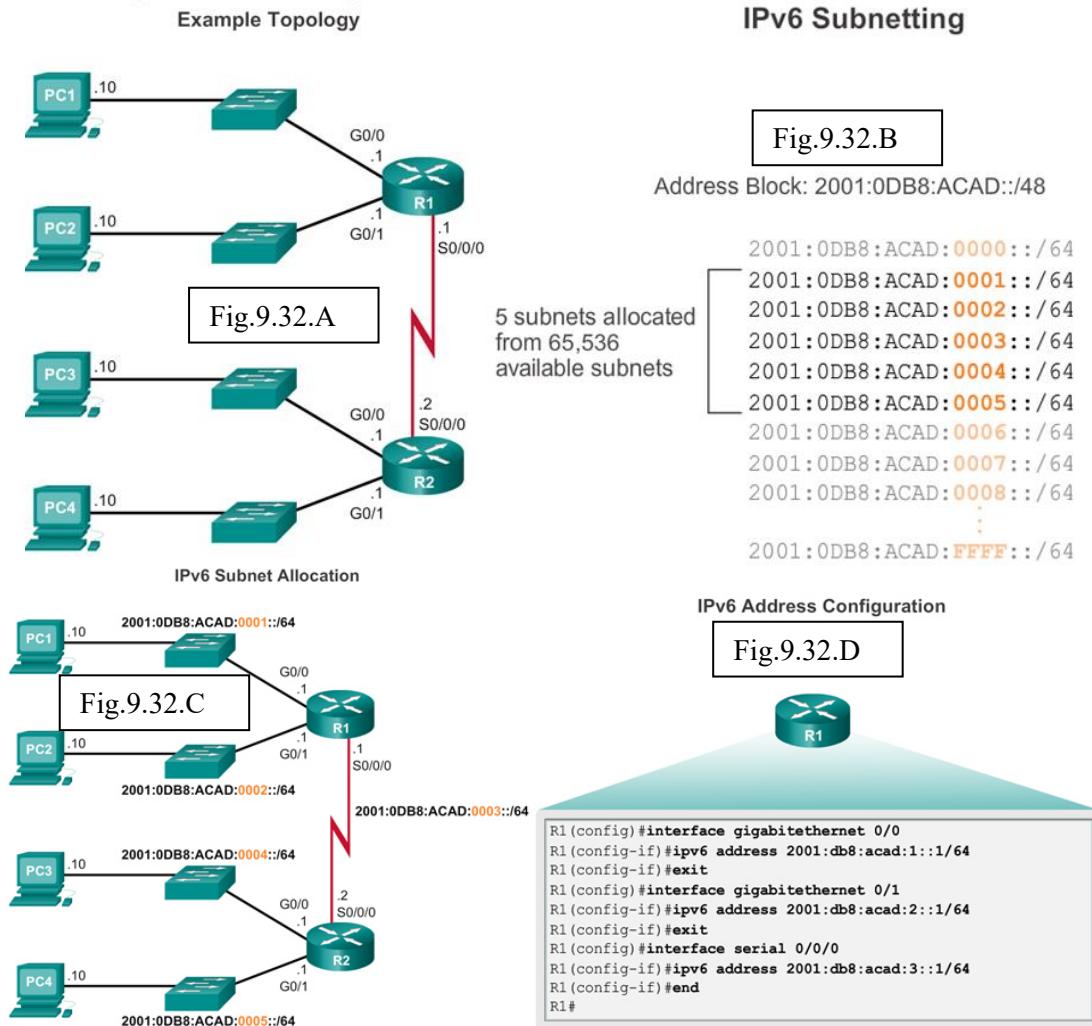
Cu mai mult de 65.000 de subrețele de unde să alegem, sarcina administratorului de rețea devine una de proiectare a unei scheme logice de adresare a rețelei.

Așa cum se poate observa în Fig.9.32.A, topologia exemplului va necesita subrețele pentru fiecare LAN, cât și pentru legătura WAN dintre R1 și R2. Spre deosebire de exemplul pentru IPv4, cu IPv6 subrețea din legătura WAN nu va fi subnetată în continuare. Deși acest lucru ar putea “risipi” adrese, nu reprezintă o preocupare atunci când folosim IPv6.

Așa cum se poate observa în Fig.9.32.B, alocarea de 5 subrețele IPv6, cu subnet ID de la 0001 la 0005 va fi folosită pentru acest exemplu. Fiecare subrețea /64 va oferi mai multe adrese decât vor fi vreodată necesare.

Așa cum se poate observa în Fig.9.32.C, fiecare segment LAN și legătura WAN au atribuite o subrețea /64.

Similar configurării de IPv4, Fig.9.32.D prezintă faptul că fiecare interfață a routerului a fost configurată cu o subrețea diferită IPv6.



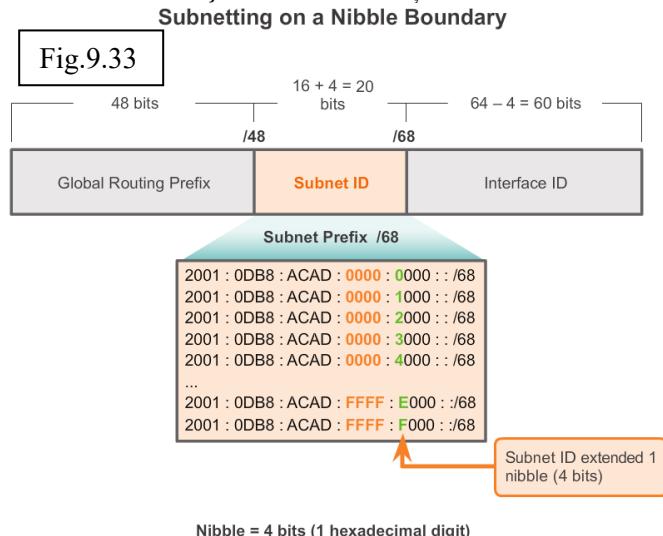
Similar împrumutului de biți din partea de host a unei adrese IPv4, biții IPv6 pot fi împrumutați din ID-ul interfeșei pentru a crea subrețele IPv6 suplimentare. Acest lucru este de obicei efectuat din motive de securitate pentru a crea mai puține hosturi pe subrețea și nu neapărat pentru crearea de subrețele suplimentare.

Atunci când extindem subnet ID prin împrumutarea de biți din interface ID, cea mai bună practică este aceea de a subneta pe “nibble boundary”. Un nibble este o cifră hexazecimală sau 4 biți. Ca și în Fig. , prefixul de subnet /64 este extins cu 4 biți sau 1 nibble la /68. Făcând acest lucru, reducem dimensiunea interface ID cu 4 biți, de la 64 la 60 de biți.

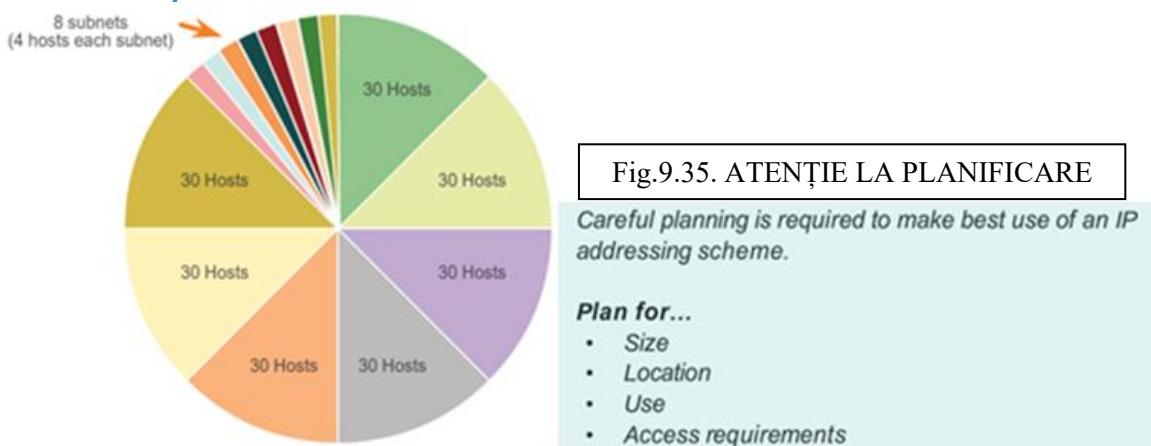
Subnetarea pe “nibble boundaries” înseamnă că folosim numai măști de rețea aliniate nibble. Începând cu /64, măștile de rețea aliniate nibble sunt /68, /72, /76, /80 etc.

Subnetarea pe “nibble boundaries” crează subrețele prin folosirea valorii hexazecimale suplimentare. În exemplu, noul subnet ID constă din 5 valori hexazecimale, începând de la 00000 la FFFF.

Este posibilă subnetarea în interiorul unei “nibble boundary”, într-o cifră hexazecimală, însă nu este recomandat sau necesar. Subnetarea în interiorul nibble nu are avantajul determinării ușoare a prefixului din interface ID. De exemplu, dacă este folosită o lungime de prefix /66, primii doi biți vor fi parte a subnet ID și următorii 2 biți din interface ID.



9.4 Concluzii Capitolul 9



Prin parcurgerea acestui material un tehnician de rețea devine familiar cu implementările de adresare IPv4 și IPv6, gata să preiea o infrastructură de rețea existentă și să aplice cunoștințele și abilitățile sale pentru a finaliza conFig.ția.

Așa cum se poate observa în Fig. , procesul de segmentare a rețelei, prin divizarea sa în mai multe spații de rețea mai mici, se numește subnetare.

Fiecare adresă de rețea are un range valid de adrese de host. Toate dispozitivelor atașate la aceeași rețea vor avea o adresă IPv4 de host pentru respectiva rețea și o mască de rețea comună sau prefix de rețea comun. Traficul nu poate fi livrat între subrețele fără utilizarea unui router. Pentru a determina dacă traficul este local sau la distanță, routerul folosește masca de subrețea.

Prefixul și masca de subrețea sunt moduri diferite de reprezentare a același lucru – partea de rețea a unei adrese.

Subrețele IPv4 sunt create prin folosirea unuia sau a mai multor biți ca biți de rețea. Doi factori foarte importanți ce vor duce la determinarea blocului de adresă IP cu masca de rețea sunt numărul de subrețele necesare și numărul maxim de hosturi necesare pe subrețea. Este o relație inversă între numărul de subrețele și numărul de hosturi. Cu cât sunt împrumutați mai mulți biți pentru a crea subrețele, cu atât rămân mai puțini biți disponibili; prin urmare, mai puține hosturi pe subrețea.

Formula 2^n (unde n este numărul de biți de host rămași) este folosită pentru calcularea numărului de adrese ce vor fi disponibile în fiecare subrețea. Însă, adresa de rețea și adresa de broadcast dintr-un range nu sunt utilizabile; prin urmare, pentru a calcula numărul de adrese utilizabile, formula 2^{n-2} este necesară.

Subnetarea unei subrețele, sau folosirea Variable Length Subnet Mask (VLSM) a fost proiectată pentru evitarea risipei de adrese.

Subnetarea IPv6 necesită o abordare diferită decât subnetarea IPv4. Un spațiu de adrese IPv6 nu este subnetat pentru conservarea adreselor; mai degrabă este subnetat pentru suportul designului logic, ierarhic al rețelei. Deci, subnetarea IPv4 este pentru gestionarea deficitului de adrese, iar subnetarea IPv6 este pentru construirea unei ierarhii de adresare în funcție de numărul de routere și de rețele suportate.

Planificarea cu atenție este necesară pentru cea mai bună utilizarea a spațiului de adrese disponibile. Cerințele de dimensiune, locație, folosire și acces sunt considerații în procesul de planificare de adresare.

După implementare, o rețea IP trebuie să fie testată pentru verificarea conectivității și a performanței funcționale.

Original	192.	168.	1.	0	000	0000	Network: 192.168.1.0/24
Mask	255.	255.	255.	0	000	0000	Mask: 255.255.255.0

Fig.9.36. ATENȚIE LA SUBNETARE

Borrowing 1 bit creates 2 subnets with the same mask.



Net 0	192.	168.	1.	0	000	0000	Network: 192.168.1.0/25
Mask	255.	255.	255.	1	000	0000	Mask: 255.255.255.128
Net 1	192.	168.	1.	1	000	0000	Network: 192.168.1.128/25
Mask	255.	255.	255.	1	000	0000	Mask: 255.255.255.128

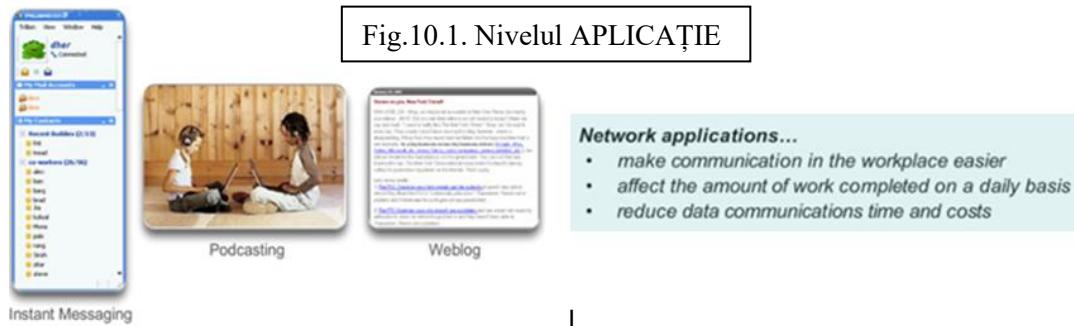
CAPITOLUL 10. NIVELUL APLICAȚIE

Introducere

Experimentăm Internetul prin intermediul World Wide Web atunci când jucăm jocuri online, vorbim cu prietenii, comunicăm prin e-mail și facem cumpărături de pe siteuri web. Aplicațiile, cum ar fi cele folosite pentru oferirea serviciilor menționate, oferă interfață umană peste bazele rețelelor. Ele ne permit să trimitem și să primim date cu ușurință. În mod normal le putem accesa și folosi fără a ști cum funcționează. Însă, pentru profesioniștii de rețea, este important să știe modul în care o aplicație este capabilă să formateze, transmită și interpreteze mesajele ce sunt trimise și primite prin rețea.

Vizualizarea mecanismelor care permit comunicarea în întreaga rețea se face mai ușor dacă vom folosi cadrul de lucru stratificat al modelului OSI.

În acest capitol, vom explora rolul nivelului aplicație și modul în care aplicațiile, serviciile și protocoalele din nivelul aplicație fac posibilă comunicarea complexă în rețelele de date.



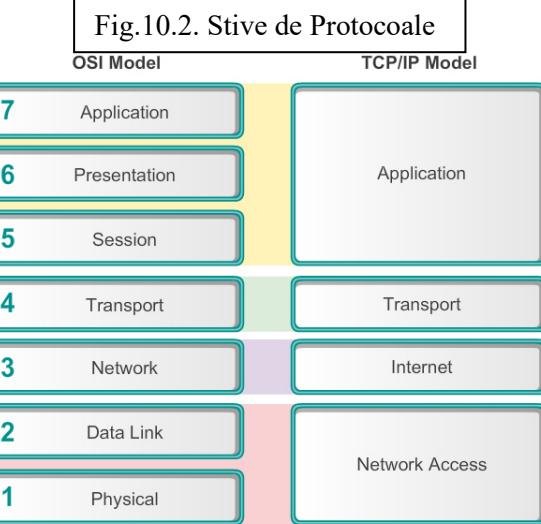
10.1 Protocolele de la nivelul aplicație

Așa cum se poate observa și în Fig. , profesioniștii de rețea folosesc modelele OSI și TCP/IP pentru a transmite documente tehnice verbale și scrise. Specialiștii de rețea pot utiliza aceste metode pentru a descrie comportamentul protocoalelor și aplicațiilor.

În modelul OSI, datele sunt pasate de la nivel la nivel, începând cu nivelul aplicație de pe hostul sursă și sunt procesate până la nivelul fizic unde sunt transmise pe canalul de comunicație la hostul destinație unde datele sunt procesate de la nivelul fizic la nivelul aplicație de pe hostul destinație.

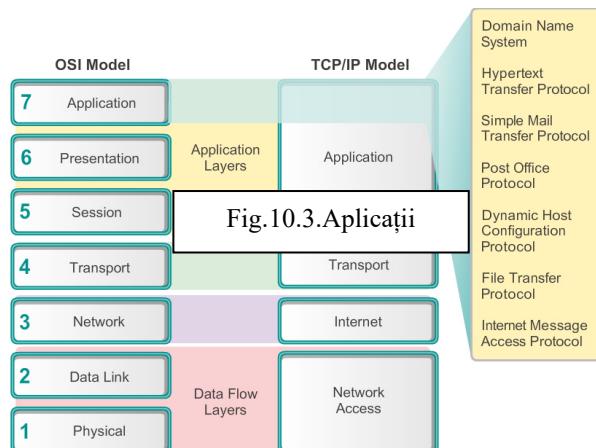
Nivelul aplicație este în vârful ambelor modele, OSI și TCP/IP. Nivelul aplicație TCP/IP include un număr de protocoale ce oferă funcționalitate specifică unei varietăți de aplicații de utilizator. Această funcționalitate a protocoalelor de la nivelul aplicație TPC/IP se potrivește cu cadrul de lucru al celor trei nivele de vârf al modelului OSI: aplicație, prezentare și sesiune. Nivelele modelului OSI 5, 6 și 7 sunt folosite ca referință pentru dezvoltatorii și furnizorii de aplicații software pentru a produce produse, cum ar fi pagini web necesare pentru accesul la rețele.

Comparing the OSI Model and TCP/IP Models



10.2 Nivelul Aplicație

Nivelul aplicație este cel mai aproape de utilizatorul final. Așa cum se poate observa în Fig. , acesta este nivelul ce oferă interfață dintre aplicațiile utilizate pentru comunicarea și baza rețelei peste care mesajele noastre sunt transmise. Protocolele de la nivelul aplicație sunt folosite pentru schimbul de date dintre programele ce rulează pe sursă și destinație. Există multe protocoale de nivel aplicație și noi protocoale ce sunt în continuă dezvoltare. Unele dintre cele mai cunoscute protocoale de nivel aplicație sunt HyperText Transfer Protocol (HTTP), File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP), Internet Message Access Protocol (IMAP) și Domain Name System (DNS) protocol.



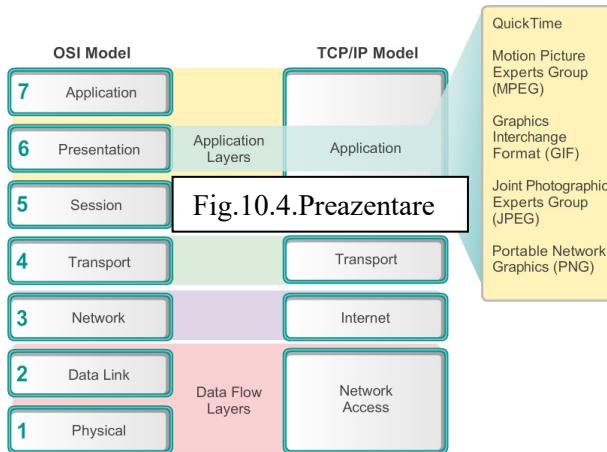
10.3 Nivelul Prezentare

Nivelul prezentare are trei funcții principale:

- Formatează, sau prezintă, datele de la dispozitivul sursă într-o formă compatibilă pentru primirea lor pe dispozitivul sursă.
- Compresează datele într-un mod ce pot fi decomprimate de către dispozitivul destinație.
- Criptează datele pentru transmisie și descripează datele pe dispozitivul destinație.

Aşa cum se poate observa în Fig. , nivelul prezentare formează datele de la nivelul aplicație și setează standarde pentru formatele de fișier. Unele standarde bine-cunoscute pentru video sunt QuickTime și Motion Picture Experts Group (MPEG). QuickTime este o specificație de computer Apple pentru video și audio și MPEG este un standard pentru compresia și codarea video și audio.

Printre cele mai cunoscute formate de imagini grafice care sunt folosite în rețele sunt formatele Graphics Interchange Format (GIF), Joint Photographic Experts Group (JPEG) și Portable Network Graphics (PNG). GIF și JPEG sunt standarde de compresie și codare pentru imaginile grafice. PNG a fost dezvoltat pentru a adresa anumitor limitări ale formatului GIF și eventual pentru înlocuirea sa.



10.4 Nivelul Sesiune

Aşa cum sugereaza şi numele, funcţiile nivelului sesiune sunt de a crea şi menţine dialogurile dintre aplicaţiile sursă şi destinaţie. Nivelul sesiune gestionează schimbul de informaţii pentru iniţierea dialogurilor, le ține active şi restabileşte sesiunile dacă sunt întrerupte pentru o perioadă mai mare de timp.

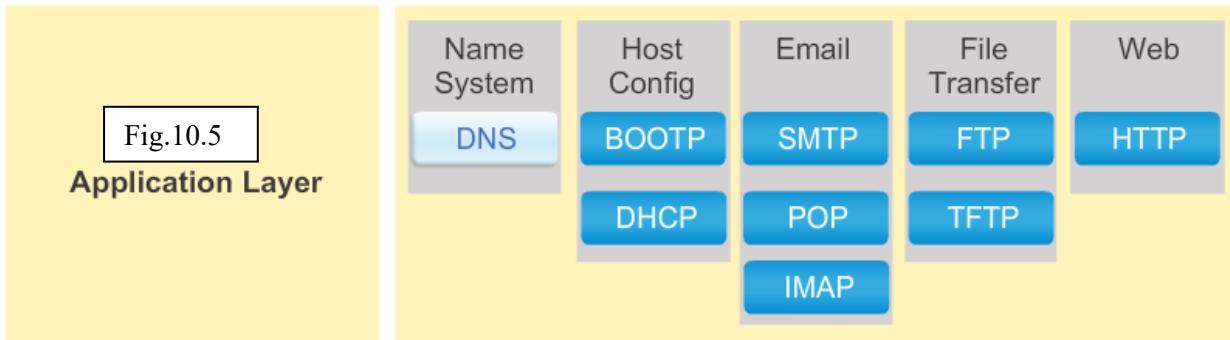
Pe când modelul OSI separă funcţia individuală de aplicație, prezentare și sesiune, aplicațiile TCP/IP cele mai cunoscute și implementate încorporează funcționalitatea acestor trei nivele.

Protocolele aplicație TCP/IP specifică formatul și informațiile de control necesare pentru mai multe funcții de comunicație prin Internet. Printre aceste protocoale TPC/IP sunt:

- **Domain Name System (DNS)** – Acest protocol rezolvă numele Internet în adrese IP.
- **Telnet** – Acesta este folosit pentru a oferi acces de la distanță la servere și dispozitive de rețea, fără securitatea transmisiei.
- **Simple Mail Transfer Protocol (SMTP)** - Acest protocol transferă mesaje mail și anexele acestora.
- **Dynamic Host ConFig.tion Protocol (DHCP)** - Protocol folosit pentru atribuirea unei adrese IP, mască de rețea, default gateway și adrese de server DNS unui host.
- **Hypertext Transfer Protocol (HTTP)** - Acest protocol transferă fișiere ce alcătuiesc paginile web ale World Wide Web.
- **File Transfer Protocol (FTP)** - Un protocol folosit pentru schimbul de fișiere interactiv între sisteme.
- **Trivial File Transfer Protocol (TFTP)** - Acest protocol este folosit pentru transferul de fișiere active neorientat pe conexiune.

- **Bootstrap Protocol (BOOTP)** - *Acest protocol este un precursor al protocolului DHCP. BOOTP este un protocol de rețea folosit pentru a obține informații de adresă IP în timpul bootup.*
- **Post Office Protocol (POP)** - *Un protocol folosit de către clienții de e-mail pentru a preluă e-mail de la un server de la distanță și a-l plasa pe hostul client.*
- **Internet Message Access Protocol (IMAP)** - *Acesta este un alt protocol pentru preluarea de e-mail și distribuirea acestuia pe hostul client.*

Protocolele de la nivelul aplicație sunt utilizate de către dispozitivele sursă și destinație în timpul unei sesiuni de comunicare. Pentru ca aceste comunicații să fie cu succes protocolele de nivel aplicație implementate pe hostul sursă și destinație trebuie să fie compatibile.



10.5 Cum Protocolele Aplicație Interacționează cu Aplicațiile Utilizator

La accesarea informațiilor pe un dispozitiv de rețea, fie că este un PC, laptop, tabletă, smartphone sau alt dispozitiv conectat la rețea, datele ar putea să nu fie stocate fizic pe dispozitiv. În acest caz, o cerere de acces a respectivelor informații trebuie să fie efectuată dispozitivului pe care sunt stocate datele. În modelul peer-to-peer (p2p), datele sunt accesate de la un dispozitiv peer fără utilizarea unui server dedicat.

Modelul de rețea P2P include două părți: rețelele P2P și aplicațiile p2p. Ambele părți au caracteristici similare, însă în practică lucrează diferit.

10.5.1 Rețelele P2P

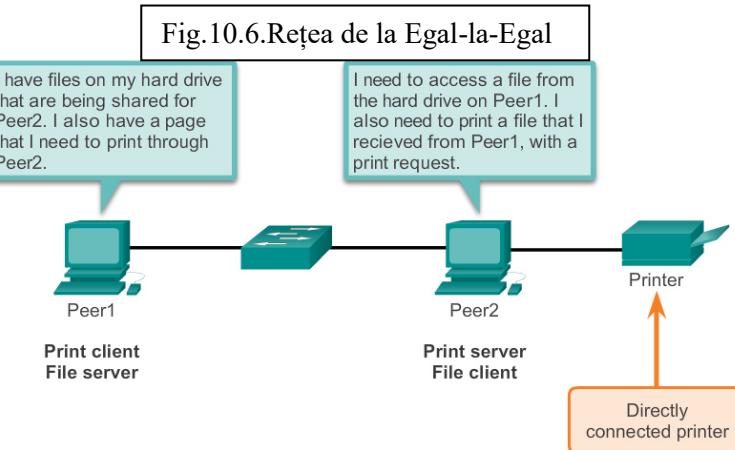
Într-o rețea P2P, două sau mai multe computere sunt conectate printr-o rețea și împart resurse (cum ar fi imprimante și fișiere) fără existența unui server dedicat. Fiecare dispozitiv final conectat (cunoscut ca un peer) poate funcționa atât ca server, cât și ca un client. Un computer își poate asuma rolul de server pentru o tranzacție, în timp ce este un client pentru altă tranzacție. Rolarile clientului și serverului sunt setate pe bază de cerere.

Un exemplu este o rețea de domiciliu simplă cu două computere, aşa cum se arată în Fig. . În acest exemplu, Peer2 are o imprimantă atașată direct prin USB și este setat să împartă imprimanta în rețea astfel încât și Peer1 să poată printa. Peer1 este setat să împartă un drive sau folder în rețea. Acest lucru permite ca Peer2 să acceseze și să salveze fișierele din folderul partajat. În plus față de împărțirea de fișiere, o rețea ca aceasta va permite utilizatorilor să activeze jocuri de rețea sau să împartă o conexiune la Internet.

Rețelele P2P descentralizează resursele dintr-o rețea. În schimbul localizării datelor pe servere dedicate pentru a fi partajate, datele pot fi localizate oriunde și pe orice dispozitiv conectat. Multe sisteme de operare actuale suportă partajarea de fișiere și printarea fără necesitatea unui software de server suplimentar. Însă, rețelele P2P nu folosesc conturi de utilizator centralizate sau servere de acces pentru a gestiona permisiunea. Prin urmare, este dificilă asigurarea securității și politicilor de acces în rețelele ce conțin mai multe computere.

Conturile de utilizator și drepturile de acces trebuie să fie setate individual pe fiecare dispozitiv peer.

Peer-to-Peer Networking

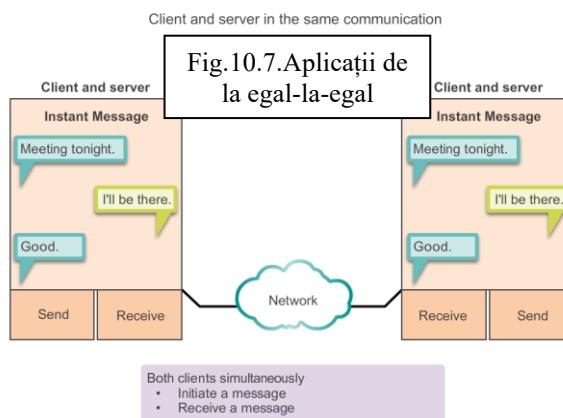


O aplicație **p2p** permite unui dispozitiv să funcționeze și ca server precum ca și client în aceeași comunicație, așa cum se poate observa și în Fig. . În acest model, fiecare client este un server și fiecare server este un client. Ambele pot iniția o comunicație și sunt considerate egale în procesul de comunicare. Însă, aplicațiile **p2p** necesită ca fiecare dispozitiv final să ofere o interfață de utilizator și să ruleze un serviciu de background. Atunci când lansăm o anumită aplicație **p2p**, se încarcă interfața de utilizator necesară și serviciile de background; cu alte cuvinte, dispozitivele pot comunica direct.

Unele aplicații **p2p** folosesc un sistem hybrid în care partajarea de resurse este descentralizată, însă indecșii ce pointează la locațiile resursei sunt stocați într-un director centralizat. Într-un sistem hybrid, fiecare peer accesează un server indexat pentru a afla locația unei resurse stocată pe un alt peer. Serverul indexat de asemenea ajută la conectarea celor două peer, însă după conectare, comunicația are loc între perechi fără comunicație suplimentară cu serverul indexat.

Aplicațiile **p2p** pot fi utilizate în rețelele P2P, rețelele client/server și peste Internet.

Peer-to-Peer Applications



Cu aplicațiile **p2p**, fiecare computer din rețea ce rulează aplicația funcționează ca un client sau ca un server pentru alte computere din rețea ce rulează aplicația. Aplicații **p2p** sunt:

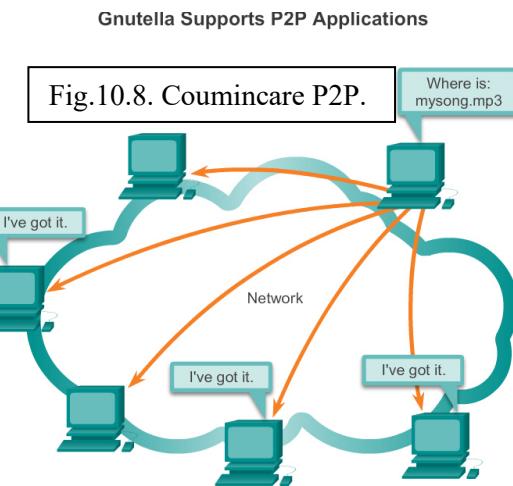
- *eDonkey*.
- *eMule*.
- *Shareaza*.
- *BitTorrent*.

- *Bitcoin*.
- *LionShare*.

Unele aplicații **p2p** sunt bazate pe protocolul Gnutella. Ele permit oamenilor să împartă fișierele de pe hard diskurile lor cu alți utilizatori. Așa cum se vede în Fig. , softwareul de client compatibil Gnutella permite utilizatorilor să se conecteze la servicii Gnutella peste Internet și să localizeze și acceseze resurse partajate de alte perechi Gnutella. Multe aplicații client sunt disponibile pentru accesarea rețelei Gnutella, cum ar fi BearShare, Gnuclous, LimeWire, Morpheus, WinMX și XoloX.

Pe când Gnutella Developer Forum gestionează protocolul de bază, furnizorii de aplicații adesea dezvoltă extensii pentru a face protocolul să funcționeze mai bine pe aplicațiile lor.

Multe aplicații **p2p** nu folosesc o bază de date centrală pentru a stoca toate fișierele disponibile pe perechi. În schimb, dispozitivele din rețea își “spun” între ele că fișierele sunt disponibile la cerere și folosesc protocolul și serviciile de partajare de fișier pentru a sprijini localizarea resurselor.



În modelul client-server, dispozitivul care cere informațiile este numit client, iar cel ce răspunde la cerere se numește server. Procesele client și server sunt considerate la nivelul aplicație. Clientul începe schimbul prin cererea datelor de la server, ce răspunde prin trimiterea unuia sau a mai multor fluxuri de date la client. Protocolele de la nivelul aplicație descriu formatul cererilor și răspunsurilor dintre clienți și servere. Pentru transferul de date real, acest schimb necesită și autentificarea de utilizator și identificarea fișierului de date ce trebuie transferat.

Un exemplu de rețea client-server este utilizarea unui serviciu de mail al ISP de a trimite, primi și stoca e-mail. Clientul de e-mail de pe un computer de domiciliu inițiază o cerere la serverul de e-mail al ISP pentru orice mail necitit. Serverul răspunde prin trimiterea emailului cerut de către client.

Deși datele sunt în mod normal descrise ca și cum ar “curge” de la server la client, unele datele “curg” întotdeauna de la client la server. Fluxul de date poate fi egal în ambele direcții, sau poate fi chiar mai mare în direcția de la client la server. De exemplu, un client poate să transfere un fișier de la server pentru scopuri de stocare. Așa cum se poate observa în Fig. , transferul de date de la un client la un server se numește **upload**, iar transferul de date de la un server la client **download**.

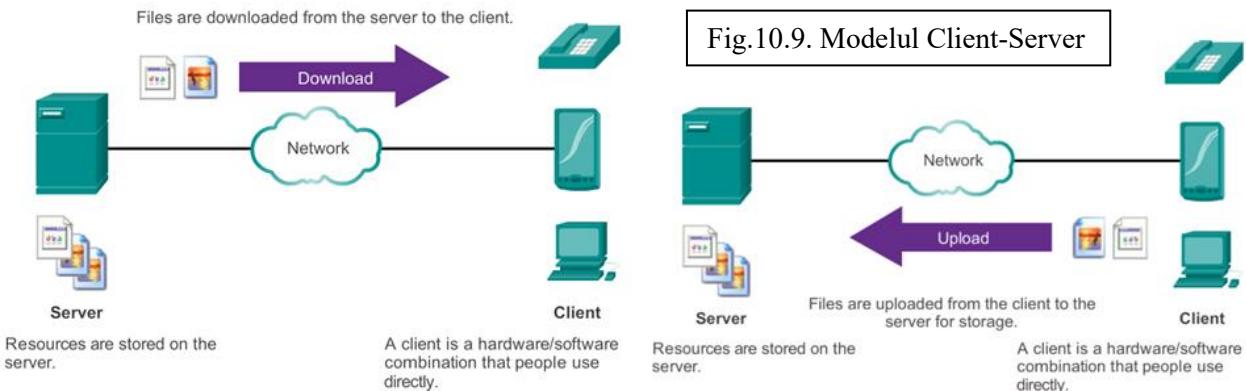


Fig.10.9. Modelul Client-Server

10.5.2 Protocole și servicii bine-cunoscute de la nivelul aplicație

Există zeci de protocole de nivel aplicație, însă într-o zi obișnuită se folosesc probabil numai cinci sau șase. Trei protocole de nivel aplicație ce sunt implicate în activitatea de zi cu zi sunt:

- *Hypertext Transfer Protocol (HTTP)*.
- *Simple Mail Transfer Protocol (SMTP)*.
- *Post Office Protocol (POP)*.

Acstea protocole de nivel aplicație fac posibilă căutarea în web și trimitera și primirea de e-mail. HTTP este folosit pentru a permite utilizatorilor să se conecteze pe siteuri web din Internet. SMTP este folosit pentru a permite utilizatorilor să trimită e-mail. POP este folosit pentru permiterea utilizatorilor să primească e-mail.

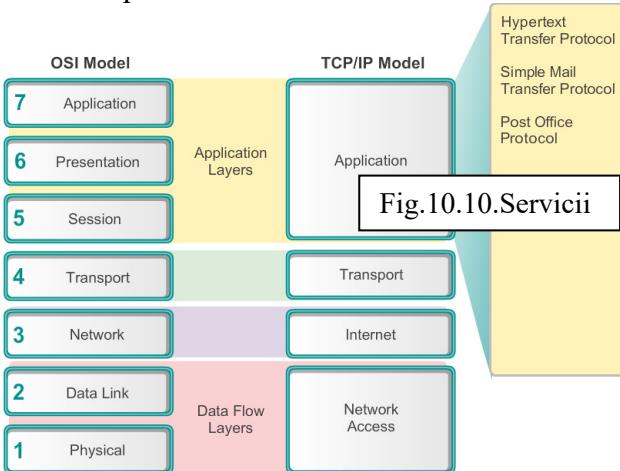


Fig.10.10.Servicii

Atunci când o adresă web sau *Uniform Resource Locator* (URL) este introdusă într-un browser web, browserul web stabilește o conexiune cu serviciul web ce rulează pe serverul ce folosește protocolul HTTP. URLs și *Uniform Resource Identifier* (URIs) sunt numele ce mulți oameni le asociază cu adrese web.

<http://www.fmi.cti-ro/index.html> URL este un exemplu de URL ce este asociat unei anumite resurse; o pagină web numită **index.html** pe un server ce se identifică ca **fmi.cti-ro**.

Browserele web sunt tipurile de aplicații client de pe un computer folosite pentru conectarea la World Wide Web și pentru a accesa resursele stocate pe un server web. Ca multe procese server, serverul web rulează ca serviciu de background și face tipuri diferite de fișiere să fie disponibile.

Pentru a accesa conținutul, clienții web se leagă la server și cer resursele dorite. Serverul răspunde cu resursele și, după primire, browserul interpretează datele și le prezintă utilizatorului.

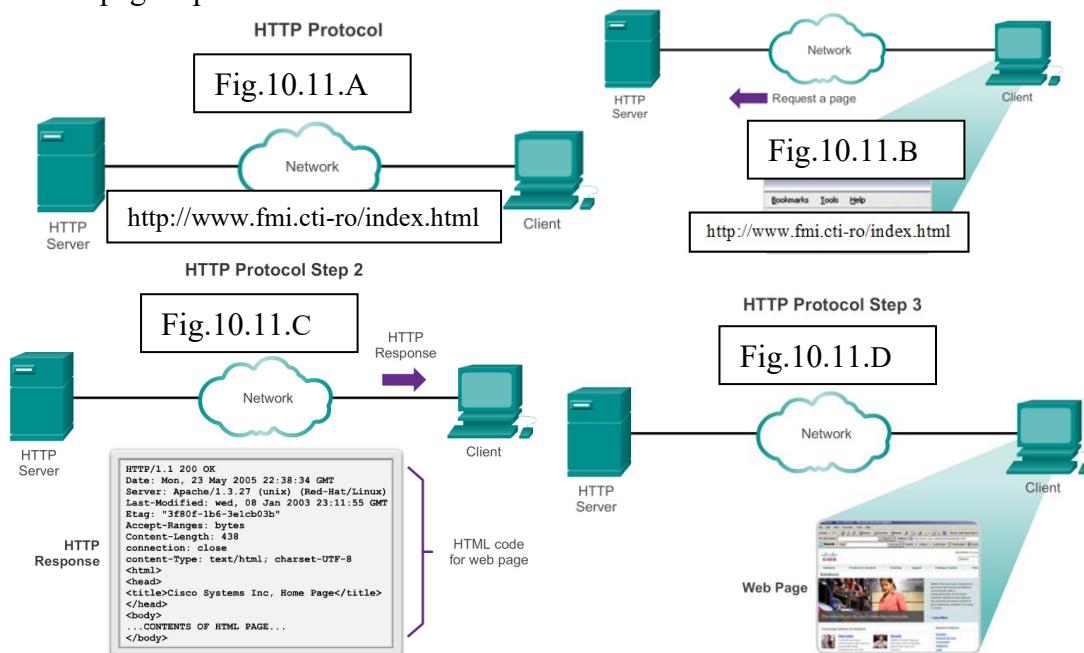
Browserele pot interpreta și prezenta mai multe tipuri de date (cum ar fi text clar sau Hypertext Markup Language, limbajul în care paginile web sunt alcătuite). Alte tipuri de date, însă, ar putea necesita alt serviciu sau program, referit nouă în mod normal ca plug-ins sau add-ons. Pentru a ajuta browserul să determine ce tip de fișier primește, serverul specifică ce tip de date conține fișierul.

Pentru a înțelege mai bine cum interacționează browserul și clientul, putem examina modul în care o pagină web este deschisă într-un browser. Pentru acest exemplu, folosim <http://www.fmi.cti-ro/index.html> URL.

Prima dată, aşa cum se poate observa și în Fig.10.11.A, browserul interpretează trei părți ale URL:

1. **http (protocolul sau schema).**
2. **www. fmi.cti-ro (numele serverului).**
3. **index.html (numele de fișier specific cerut).**

Așa cum se poate vedea în Fig.10.11.B, browserul apoi verifică un server de nume ce convertește **www. fmi.cti-ro** într-o adresă numerică, utilizată pentru conectarea la server. Utilizând cerințele HTTP, browserul trimită o cerere **GET** serverului și cere fișierul **index.html**. Serverul, cum se poate vedea în Fig.10.11.C, trimite un cod HTML pentru această pagină browserului. La final, aşa cum se poate vedea în Fig.10.11.D, browserul deschide codul HTML și formează pagina pentru fereastra de browser.



HTTP este folosit peste World Wide Web pentru transferul de date și este unul dintre cele mai utilizate protocoale de aplicație de astăzi. A fost dezvoltat inițial pentru a simplifica publicarea și preluarea paginilor HTML; însă flexibilitatea HTTP l-a facut să fie o aplicație vitală din sistemele distribuite de colaborare pentru informație.

HTTP este un protocol bazat pe cerere/răspuns. Atunci când un client, în mod normal un browser web, trimită o cerere la un server web, HTTP specifică tipul de mesaj pentru comunicație. Trei tipuri comune de mesaj sunt **GET**, **POST** și **PUT** (Fig.).

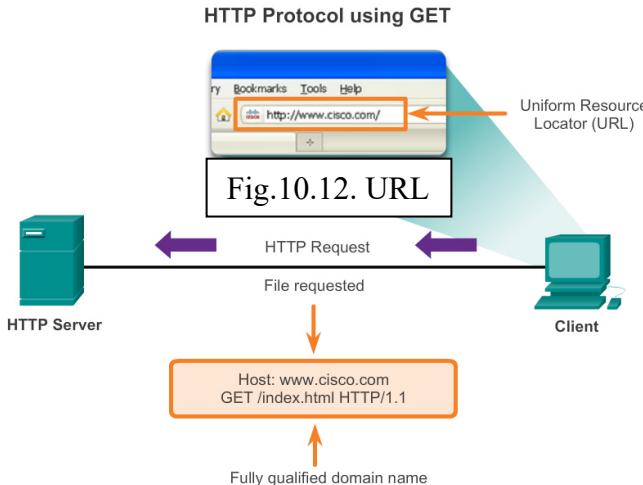
GET este o cerere de date a clientului. Un client (browser web) trimită un mesaj **GET** serverului web pentru a cere pagini HTML. Atunci când serverul primește cererea **GET**, răspunde cu o linie de status, cum ar fi **HTTP/1.1 200 OK** și cu un mesaj propriu. Mesajul de la server ar putea include fișierul HTML cerut, dacă este disponibil, sau poate conține o eroare sau un mesaj de informare, cum ar fi "The location of the requested file has changed."

POST și **PUT** sunt folosite pentru încărcarea fișierelor de date pe serverul web. De exemplu, atunci când utilizatorul introduce datele într-o formă încorporată unei pagini web (cum ar fi cazul în care se completează o cerere de ordine), mesajul **POST** este trimis serverului web. Datele pe care utilizatorul le-a prezentat în formă sunt incluse în mesajul **POST**.

PUT încarcă resursele sau conținutul serverului web. De exemplu, dacă un utilizator încearcă să încarce un fișier sau o imagine pe un website, un mesaj **PUT** este trimis de la client la server cu fișierul atașat sau cu imaginea.

Deși HTTP este remarcabil flexibil, nu este un protocol sigur. Mesajele cerute trimit informații serverului în text clar ce poate fi interceptate și citite. În mod similar, răspunsurile serverului, în mod normal pagini HTML, sunt de asemenea necriptate.

Pentru comunicarea securizată prin Internet, protocolul HTTP Secure (HTTPS) este folosit pentru accesarea și postarea informațiilor de server web. HTTP Secure (HTTPS) poate folosi autentificarea și criptarea pentru securizarea datelor care circulă de la client la server. HTTPS specifică reguli suplimentare pentru transferarea datelor de la nivelul aplicație la nivelul transport. HTTPS folosește același proces cerere client-răspuns server ca HTTP, însă streamul de date este criptat cu Secure Socket Layer (SSL) înainte de a fi transferat peste rețea. HTTPS crează timp de procesare și încarcare suplimentar pe server datorită criptării și decriptării traficului.



Unul dintre principalele servicii oferite de către un ISP este gazdă de e-mail (hosting). Serviciul Email a revoluționat modul în care oamenii comunică prin viteza și simplitatea sa. Pentru a rula pe un computer sau alte dispozitive finale, e-mail necesită multe aplicații și servicii.

Email este o metodă store-and-forward pentru trimiterea, stocarea și preluarea mesajelor electronice dintr-o rețea. Mesajele e-mail sunt stocate în baze de date sau servere de mail. ISPiștii adesea gestionează servere de mail ce suportă mai multe conturi de clienți diferite.

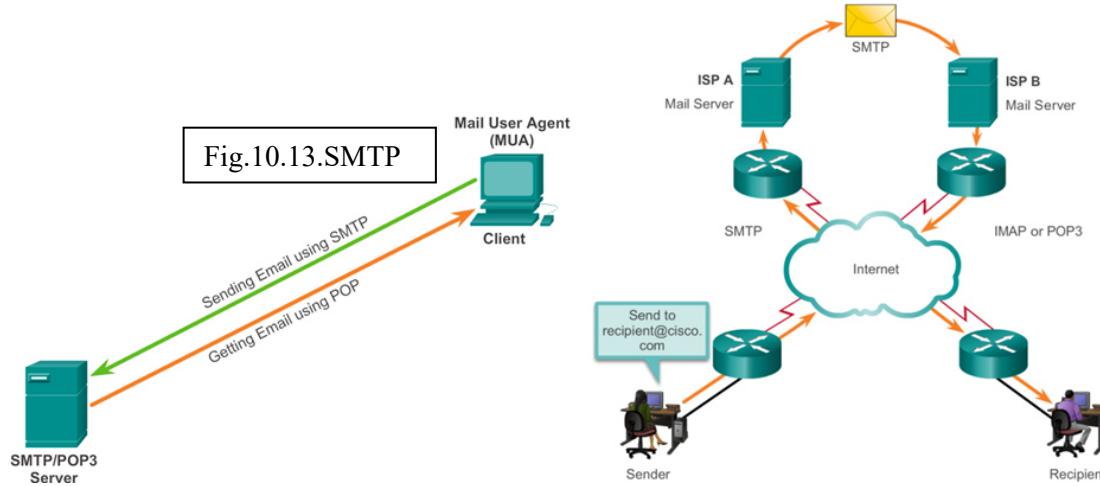
Clienții de e-mail comunică cu serverele de mail pentru a trimite și primi e-mail. Serverele de mail comunică cu alte servere de mail pentru a transporta mesajele de la un domeniu la altul. Un client e-mail nu comunică direct cu alt client e-mail atunci când trimite un e-mail. În schimb, ambii clienți se bazează pe serverul de mail pentru transportarea mesajelor. Acest lucru se întâmplă chiar și atunci când ambii clienți sunt în același domeniu.

Clienții de e-mail trimit mesaje serverului de e-mail config.t în setările de aplicație. Atunci când serverul primește mesajul, verifică dacă domeniul este localizat în baza de date locală. Dacă nu este, trimit o cerere DNS pentru a determina adresa IP a serverului de mail pentru domeniul destinație. E-mailul este apoi transmis mai departe serverului adecvat.

E-mail suportă trei protocoale separate de funcționare: Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP) și Internet Message Access Protocol (IMAP). Procesul de la

nivelul aplicație ce trimite mailul folosește SMTP. Aceasta este cazul de trimitere de la un client la un server, sau de la un server la altul.

Un client preia e-mailul, însă folosind unul dintre cele două protocoale de nivel aplicație: POP sau IMAP.

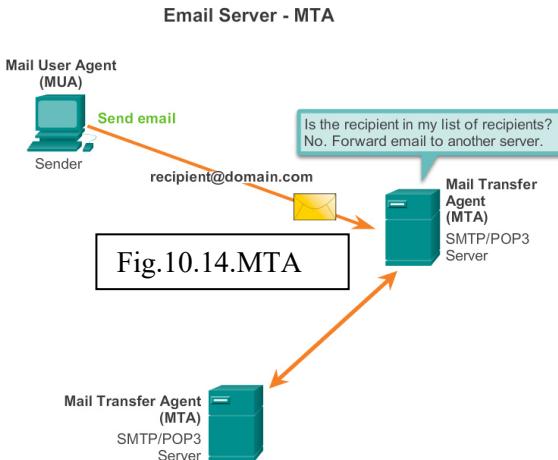


Simple Mail Transfer Protocol (SMTP) transferă mailul în mod eficient și de încredere. Pentru ca aplicațiile SMTP să lucreze eficient, mesajul de mail trebuie să fie transmis adecvat și procesele SMTP trebuie să ruleze pe ambele echipamente, client și server.

Formatele de mesaj SMTP necesită un header de mesaj și un corp al mesajului. Pe când corpul mesajului poate conține orice cantitate de text, headerul mesajului trebuie să aibă o adresă a destinatarului e-mailului corect formatată și o adresă a sursei. Orice alte informații de header sunt opționale.

Atunci când un client trimite e-mail, procesul SMTP client se conectează cu un proces SMTP server pe portul bine cunoscut 25. După realizarea conexiunii, clientul încearcă să trimită e-mailul serverului prin intermediul conexiunii. Atunci când serverul primește mesajul, fie plasează mesajul în contul local, dacă destinația este locală, fie îl transmite mai departe folosind procesul de conexiune SMTP la alt server de mail pentru livrare.

Serverul de e-mail destinație ar putea să nu fie online sau să fie ocupat atunci când mesajele de e-mail sunt transmise. Prin urmare, mesajele SMTP sunt într-o "coadă" pentru a fi transmise mai târziu. Periodic, serverul verifică coada pentru mesaje și încearcă să le trimită din nou. Dacă mesajul nu este livrat după o perioadă predefinită de timp, se întoarce la expeditor ca nelivrabil.

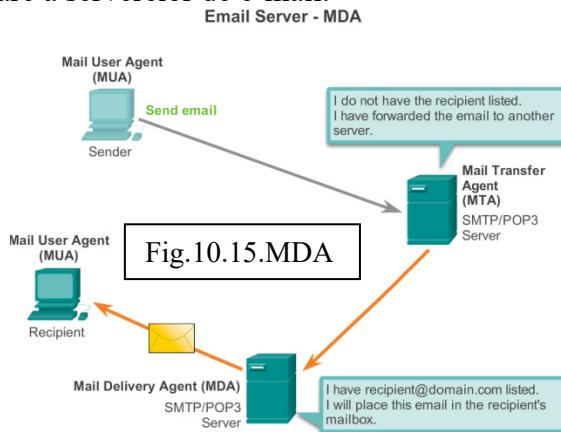


Post Office Protocol (POP) permite unei stații de lucru să preia mailul de pe un server de mail. Cu POP, mailul este descărcat de la server la client și apoi este șters de pe server.

Serverul activează serviciul POP prin “ascultarea pasivă” pe portul 110 TCP pentru cereri de conexiune de la client. Atunci când un client vrea să folosească serviciul, trimite o cerere de stabilire a unei conexiuni TCP la server. Atunci când conexiunea este stabilită, serverul POP transmite “un salut”. Clientul și serverul POP apoi schimbă comenzi și răpsunsuri între ele până când conexiunea este închisă sau abandonată.

Deoarece mesajele de e-mail sunt descărcate pe client și șterse de pe server, nu există o locație centralizată unde mesajele de e-mail sunt păstrate. Deoarece POP nu stocă mesajele, nu este preferat pentru afacerile mici ce necesită o soluție centralizată de backup.

POP3 este preferat pentru un ISP, deoarece ameliorează responsabilitatea de gestionare a unei cantități mari de stocare a serverelor de e-mail.

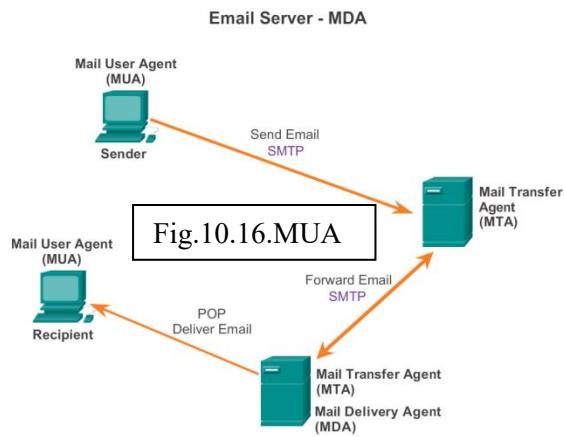


Internet Message Access Protocol (IMAP) este un alt protocol ce descrie o metodă de primire a mesajelor de e-mail. Însă, spre deosebire de POP, atunci când un utilizator se conectează la un server IMAP, copii ale mesajelor sunt descărcate pe aplicația client. Mesajele originale sunt păstrate în server până când sunt șterse manual. Utilizatorii vizualizează copii ale mesajelor pe softwareul lor de client e-mail.

Utilizatorii pot crea o ierarhie de fișiere pe server pentru a organiza și stoca mailul. Acea structură de fișier este duplicată pe clientul e-mail. Atunci când un utilizator decide să șteargă un mesaj, serverul sincronizează acțiunea și șterge mesajul de pe server.

Pentru afacerile mici și mijlocii, există multe avantaje ale utilizării IMAP. IMAP poate oferi stocare pe termen lung a mesajelor de e-mail pe serverele de mail și permite backup centralizat. Permite de asemenea angajaților să acceseze mesajele de e-mail de pe locații multiple, folosind dispozitive diferite sau softwareuri diferite de client. Structura folderului de mailbox pe care un utilizator se așteaptă să o vadă este disponibilă indiferent de modul în care utilizatorul accesează mailbox.

Pentru un ISP, IMAP ar putea să nu fie protocolul preferat. Poate fi scump pentru achiziționare și gestionare a spațiului de disk astfel încât să suporte un număr mare de e-mailuri stocate. În plus, dacă clienții doresc ca mailboxes să fie susținute în mod curent, poate crește costurile ISP.



10.6 Furnizarea Serviciilor de Adresare IP

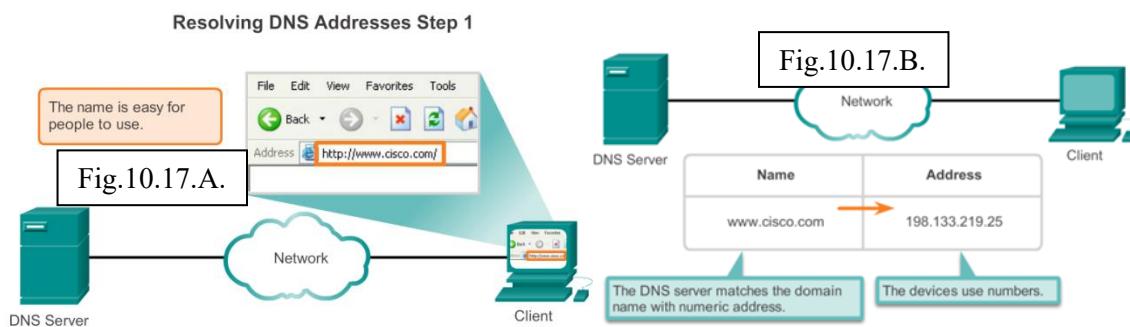
În rețelele de date, dispozitivele sunt etichetate cu adrese IP numerice pentru a transmite și primi date peste rețele. Mulți oameni nu pot ține minte aceste adrese numerice. Numele de domeniu au fost create pentru convertirea adresei numerice într-un nume simplu, recunoscut.

În Internet, aceste nume de domeniu, cum ar fi <http://www.fmi.unibuc.ro>, sunt mult mai ușor de reținut pentru oameni decât **198.133.219.25**, ce este adresa numerică reală pentru acest server. Dacă facultatea decide să schimbe adresa numerică a www.fmi.unibuc.ro, este transparentă utilizatorului deoarece numele de domeniu rămâne la fel. Noua adresă este legată simplu la numele de domeniu existent și conectivitatea este menținută. Atunci când rețelele erau mici, era simplă sarcina de gestiune a mapării dintre numele de domeniu și adresele reprezentate. O dată cu creșterea rețelelor și a numărului de dispozitive, sistemul manual a devenit inaplicabil.

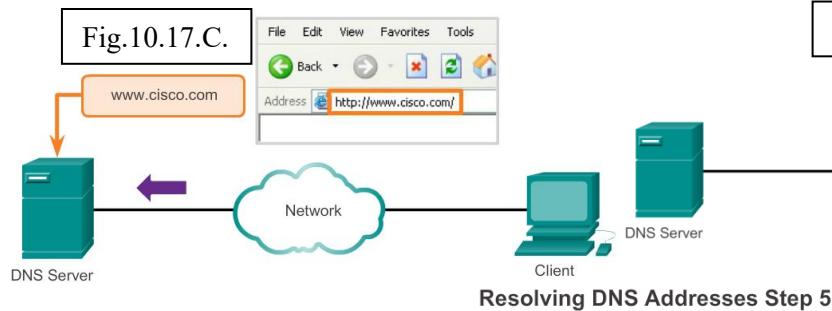
Domain Name System (DNS) a fost creat pentru ca numele de domeniu să fie asociat acestor rețele. DNS folosește un set distribuit de servere pentru a rezolva numele asociate cu adresele numerice.

Protocolul DNS definește un serviciu automat ce asociază numele resurselor cu adresele de rețea numerice necesare. Include formatul pentru cerere, răspuns și date. Comunicațiile protocolului DNS folosesc un singur format numit mesaj. Acest format de mesaj este folosit pentru toate tipurile de cereri de client și răspunsuri de server, mesaje de eroare și transferul de informații ale resurselor dintre servere.

Fig. 10.17.A /Fig.10.17.E prezintă pașii pentru rezolvarea cererilor DNS.

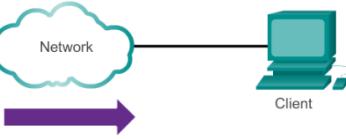


Resolving DNS Addresses Step 3



Resolving DNS Addresses Step 4

Fig.10.17.D.



Resolving DNS Addresses Step 5

Fig.10.17.E.



Un server DNS oferă rezoluție de nume folosind *Berkeley Internet Name Domain* (BIND), sau daemon de nume. BIND a fost dezvoltat inițial de către patru studenți ai University of California Berkley la începutul anilor 1980. Așa cum se poate observa în Fig.10.17C , formatul mesajului DNS folosit de către BIND este cel mai cunoscut format DNS din Internet.

Serverul DNS stochează diferite tipuri de înregistrări de resurse folosite pentru rezolvarea numelor. Aceste înregistrări conțin nume, adresa și tipul de înregistrare.

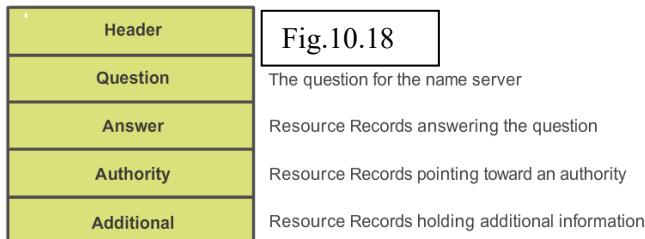
Unele dintre aceste înregistrări sunt:

- **A – O adresă de dispozitiv final.**
- **NS – Un nume de server autoritar.**
- **CNAME – Un nume canonic (sau Fully Qualified Domain Name) pentru un alias; folosit atunci când mai multe servicii au o singură adresă de rețea, însă fiecare serviciu are propria intrare în DNS.**
- **MX - Mail exchange record; mapează un nume de domeniu la o listă de servere de mail exchange pentru respectivul domeniu.**

Atunci când un client face o interogare, procesul BIND al serverului se uită mai întâi în propriile înregistrări pentru a rezolva numele. Dacă nu este capabil să rezolve numele folosind propriile înregistrări stocate, contactează alte servere pentru rezolvarea numelui.

Cererea ar putea să fie pasată mai multor server, ceea ce poate lua mai mult timp și poate consuma lățime de bandă. După ce se găsește o potrivire și este transmisă la serverul original, serverul stochează temporar adresa ce corespunde numelui în memoria cache.

Dacă același nume este cerut din nou, primul server poate returna adresa prin folosirea valorii stocate în name cache. Caching reduce traficul de cerere de date DNS și volumul de lucru al serverelor de pe nivelele de mai sus din ierarhie. Serviciul Client DNS de pe PCurile Windows optimizează performanța rezoluției numelui DNS prin stocarea numelor rezolvate anterior în memorie. Comanda **ipconfig /displaydns** afișează toate înărările DNS stocate pe un sistem PC cu sistem de operare Windows.



Protocolul DNS folosește un sistem ierarhic pentru a crea o bază de date spre a oferi rezoluție de nume. Ierarhia arată ca un copac întors cu rădăcina în vârf și ramurile în jos. DNS folosește numele de domeniu pentru a forma ierarhia.

Structura de nume este împărțită în zone mici, gestionabile. Fiecare server DNS menține un fișier specific de baze de date și este responsabil numai de gestionarea mapărilor de nume în IP pentru respectiva mica parte a întregii structuri DNS. Atunci când un server DNS primește o cerere de traducere de nume ce nu corespunde zonei sale DNS, serverul DNS transferă cererea la alt server DNS din zona adecvată pentru traducere.

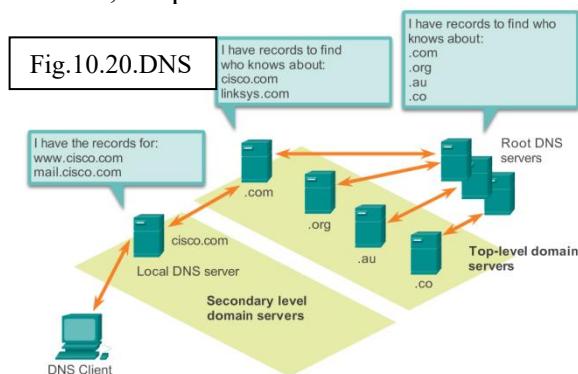
Notă: DNS este scalabil deoarece rezoluția de hostname este împărțită mai multor servere.

Domeniile diferite de la nivelul de sus reprezintă fie tipul de organizație, fie țara de origine. Exemple de domenii de la nivelul superior sunt:

- **.au – Australia.**
- **.co – Colombia.**
- **.com – o afacere sau industrie.**
- **.jp – Japan.**
- **.org – o organizație non-profit.**

După domeniile de la nivelul superior sunt numele de domeniu de nivel doi, iar sub ele sunt alte domenii de nivele inferioare. Fiecare nume de domeniu este o cale în jos în copac începând de la rădăcină. De exemplu, aşa cum se poate vedea în Fig. , serverul rădăcină DNS ar putea să nu știe exact unde înregistrarea pentru serverul de e-mail, **mail.fmi.unibuc.ro**, este localizată, însă menține o înregistrare pentru domeniul .ro în domeniul de nivel înalt. Deci, serverele din domeniul .ro ar putea să nu aibă o înregistrare pentru **mail.fmi.unibuc.ro**, însă au o înregistrare pentru domeniu. Serverele din domeniul **fmi.unibuc.ro** au o înregistrare (o înregistrare MX mai precis) pentru **mail.fmi.unibuc.ro**.

DNS se bazează pe ierarhia de servere descentralizate pentru a stoca și menține aceste înregistrări de resurse. Înregistrările de resurse listeză numele de domeniu pe care serverul le poate rezolva și serverele alternative ce pot procesa cererea. Dacă un server dat are înregistrările de resurse ce corespund nivelului sau de domeniului din ierarhie, se numește autoritar pentru respectivele înregistrări. De exemplu, un server de nume din domeniul **cisco.netacad.net** nu va fi autoritar pentru înregistrarea **mail.cisco.com** deoarece respectiva înregistrare este ținută într-un server de nivel de domeniu ridicat; în special serverul de nume din domeniul **cisco.com**.



DNS este un serviciu client/server; însă, diferă de alte servicii client/server. Pe când alte servicii folosesc un client ce este o aplicație (cum ar fi web browser, e-mail client), clientul DNS rulează ca un serviciu de sine stătător. Clientul DNS, uneori numit DNS resolver, suportă rezoluția de nume pentru alte aplicații de rețea și alte servicii ce au nevoie de el.

La configurația unui dispozitiv de rețea, în general oferim una sau mai multe adrese de server DNS pe care clientul DNS le poate folosi pentru rezoluția de nume. În mod normal ISP oferă adresele folosite pentru serverele de DNS. Atunci când o aplicație de utilizator cere să se conecteze la un dispozitiv de la distanță în funcție de nume, clientul DNS cere unuia dintre serverele de nume să rezolve numele într-o adresă numerică.

Sistemele de operare de pe computer au de asemenea un utilitar numit **nslookup** ce permite utilizatorului să ceară manual serverelor de nume să rezolve un hostname dat. Acest utilitar poate fi utilizat de asemenea pentru depanarea problemelor de rezoluție de nume și pentru identificarea stării curente ale serverelor de nume.

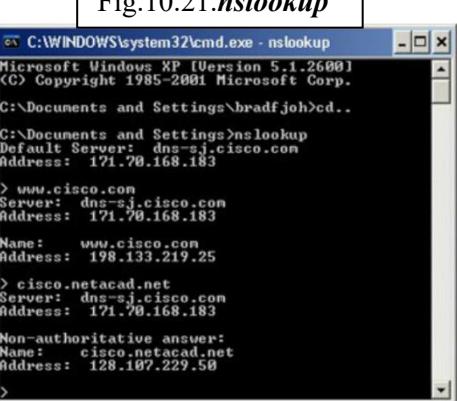
În Fig. , atunci când comanda **nslookup** este efectuată, serverul DNS implicit config.t pentru hostul respectiv este afișat. În acest exemplu, serverul DNS este dns-google.com ce are adresa 8.8.8.8.

Numele unui host sau domeniu poate fi introdus în promptul **nslookup**. În prima interogare din Fig. , o cerere este efectuată pentru **fmi.unibuc.ro**. Serverul de nume oferă adresa 198.133.219.25.

Interrogările arătate în Fig. sunt numai teste simple. Utilitarul **nslookup** are multe opțiuni disponibile pentru testare și verificare extinsă a procesului DNS. La final, introducem exit pentru a ieși din utilitarul **nslookup**.

Using nslookup

Fig.10.21.nslookup



Serviciul Dynamic Host Configuration Protocol (DHCP) permite dispozitivelor dintr-o rețea să obțină adresele IP și alte informații de la un server DHCP. Acest serviciu automatizează atribuirea adreselor IP, măștilor de rețea, de gateway și alți parametrii de rețea. Aceasta se numește adresare dinamică. Alternativa adresării dinamice, este adresarea statică. La utilizarea adresării statice, administratorul de rețea introduce manual informațiile de adresă IP pe hosturile de rețea.

DHCP permite unui host să obțină o adresă IP în mod dinamic atunci când se conectează la rețea. Serverul DHCP este conectat și o adresă este cerută. Serverul DHCP alege o adresă dintr-un range de adrese configurate numit pool și o atribuie ("încărca") hostului pentru o perioadă de timp setată.

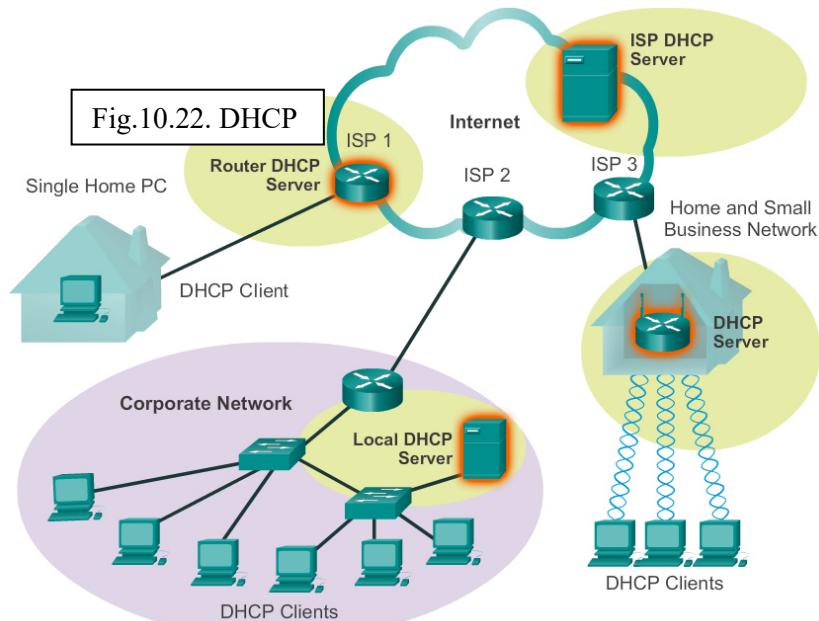
În rețelele locale mari, sau unde populația de utilizatori se schimbă frecvent, DHCP este preferat pentru atribuirea adreselor. Utilizatorii noi pot veni cu laptopuri și obțin o conexiune; alții utilizatori pot avea stații de lucru noi ce trebuie să fie conectate. În loc ca administratorul de rețea să atribuie adrese pentru fiecare stație de lucru, mult mai eficient este ca atribuirea de adrese IP să se facă automat prin DHCP.

Adresele distribuite prin DHCP nu sunt atribuite permanent hosturilor, ci sunt numai închiriate pentru o anumită perioadă de timp. Dacă hostul nu mai este alimentat sau este înălăturat din rețea, adresa se întoarce în pool pentru reutilizare. Acest lucru este util în mod special pentru utilizatorii mobili ce vin și pleacă din rețea. Utilizatorii pot să se mute frecvent dintr-o locație în alta și să restabilească conexiuni la rețea. Hostul poate obține o adresă IP după ce este efectuată conexiunea hardware, printr-un LAN wireless sau cablat.

DHCP face posibilă conectarea la Internet prin wireless în aeroporturi sau cafenele. Atunci când un dispozitiv wireless intră într-un hotspot, dispozitivul client DHCP contactează serverul DHCP local printr-o conexiune wireless și serverul DHCP atribuie o adresă IP dispozitivului.

Așa cum se vede în imagini, mai multe tipuri de dispozitive pot fi servere DHCP atunci când rulează un serviciu software DHCP. Serverul DHCP din cele mai multe rețele mijlocii spre mari este de obicei un server local dedicat bazat pe PC. În rețelele de domiciliu, serverul DHCP este localizat pe routerul local ce conectează rețeaua de domiciliu la ISP. Hosturile locale primește informații de adresă IP direct de la routerul local. Routerul local primește o adresă IP de la serverul DHCP de la ISP.

DHCP poate presupune un risc de securitate deoarece orice dispozitiv conectat la rețea poate primi o adresă. Riscul face securitatea fizică un factor determinant al alegerii între a utiliza adresarea manuală sau statică. Ambele, adresarea manuală și cea statică, au loc în designul de rețea. Multe rețele folosesc ambele forme de adresare, adresarea manuală și cea statică. DHCP este folosit pentru hosturi de uz general, cum ar fi dispozitivele utilizatorului final; adresarea statică este folosită pentru dispozitive de rețea, cum ar fi gateways, switches, servere și imprimante.



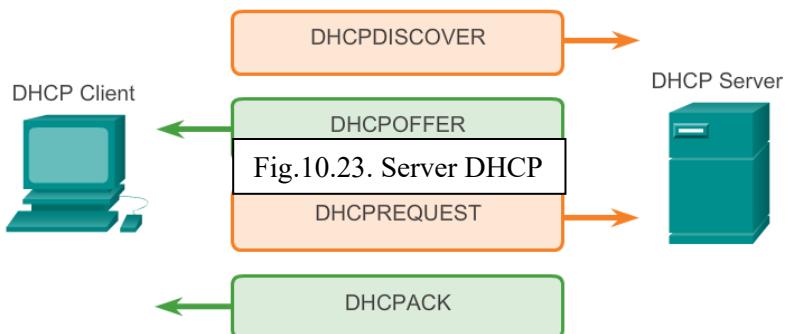
Fără DHCP, utilizatorii trebuie să introducă manual adresa IP, masca de rețea și alte setări de rețea pentru a se conecta la rețea. Serverul DHCP menține un pool de adrese IP și închiriază o adresă oricărui client activat DHCP atunci când clientul este alimentat. Deoarece adresele IP sunt mai degrabă diminice (închiriate) decât statice (permanent atribuite), adresele care nu mai sunt folosite sunt automat returnate în pool pentru realocare. Așa cum se vede în Fig. , atunci când un dispozitiv activat DHCP se conectează la rețea, clientul trimite un mesaj broadcast de descoperire DHCP (DHCPDISCOVER) pentru a identifica orice server DHCP disponibil în rețea. Un server DHCP răspunde cu un mesaj de ofertă DHCP (DHCPOFFER), ce oferă o închiriere clientului.

Mesajul oferit conține adresa IP și masca de rețea ce vor fi atribuite, adresa IP a serverului DNS și adresa IP a default gateway. Oferta de închiriere include de asemenea și durata închirierii.

Clientul ar putea primi mai multe mesaje DHCP OFFER dacă există mai multe servere DHCP în rețeaua locală; prin urmare, trebuie să aleagă între ele și trimite un mesaj de cerere DHCP (DHCP REQUEST) ce identifică serverul explicit și oferta de închiriere pe care clientul a acceptat-o. Un client ar putea alege de asemenea să ceară o adresă pe care a mai avut-o alocată de către server.

Presupunând că adresa IP cerută de către client sau oferită de către server este încă disponibilă, serverul întoarce un mesaj de confirmare DHCP (DHCP ACK) ce confirmă clientului faptul că închirierea este finalizată. Dacă oferta nu mai este disponibilă, poate datorită faptului că alt client a închiriat-o sau a apărut timeout, serverul selectat răspunde cu un mesaj de confirmare negativ DHCP (DHCP NAK). Dacă un mesaj DHCP NAK este întors, procesul de selecție trebuie să înceapă din nou cu un nou mesaj DHCP DISCOVER transmis. După ce clientul a închiriat, trebuie să reînnoiască cererea înainte de expirarea închirierii, printr-un alt mesaj DHCP REQUEST.

Serverul DHCP asigură faptul că toate adresele IP sunt unice (aceeași adresa IP nu poate fi atribuită la două dispozitive diferite din rețea simultan). Folosirea DHCP, permite administratorilor de rețea să reconfigureze ușor adresele IP de client fără a face schimbări manuale asupra clientilor. Multă furnizori de Internet folosesc DHCP pentru alocarea adreselor clientilor lor ce nu necesită o adresă statică.



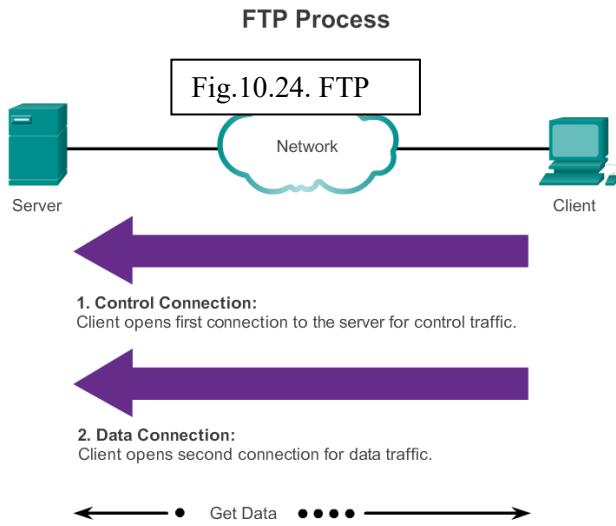
10.7 Furnizarea Serviciilor de Partajare de Fișiere

File Transfer Protocol (FTP) este alt protocol comun de la nivelul aplicație. FTP a fost dezvoltat pentru a permite transferul de date dintre un client și un server. Un client FTP este o aplicație ce rulează pe un computer și este utilizată pentru a trimite și a prelua date de la un server ce rulează un FTP daemon (FTPD).

Așa cum se poate observa și în Fig. , pentru un transfer cu succes al datelor, FTP necesită două conexiuni între client și server, una pentru comenzi și răspunsuri, alta pentru transferul real de fișiere:

- Clientul stabilește prima conexiune cu serverul pentru traficul de control, constând din comenzi de client și răspunsuri ale serverului.
- Clientul stabilește a doua conexiune cu serverul pentru transferul real de fișiere. Această conexiune este creată de fiecare dată când există date de transmis.

Transferul datelor poate avea loc în ambele direcții. Clientul poate descărca date de la server sau clientul poate încărca date pe server.



Server Message Block (**SMB**) este un protocol client-server de partajare de fișiere, dezvoltat de IBM la sfârșitul anilor 1980 pentru a descrie structura resurselor de rețea partajate, cum ar fi directoare, fișiere, imprimante și porturi seriale. Este un protocol cerere-răspuns.

Protocolul **SMB** descrie accesul la sistemul de fișier și modul în care clienții pot cere fișiere. Descrie de asemenea comunicarea interproceselor de protocol **SMB**. Toate mesajele **SMB** împart un format comun. Acest format folosește un header de dimensiune fixă, urmat de un parametru de dimensiune variabilă și de componentă de date.

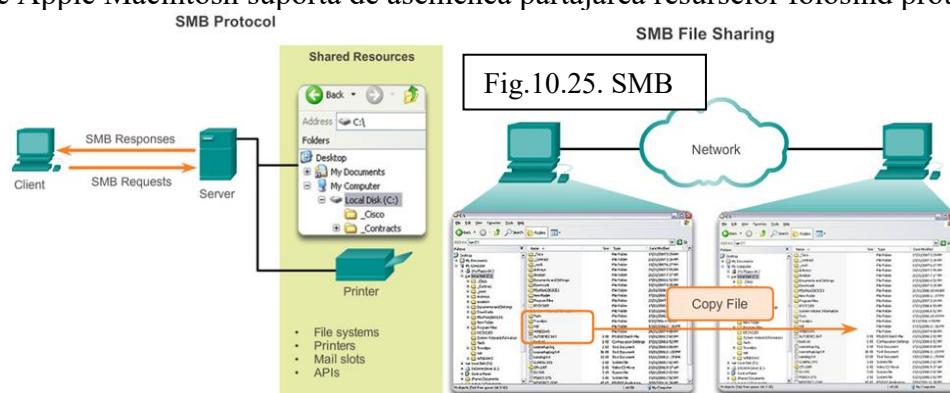
Mesajele **SMB** pot:

- Începe, autentifica și termină sesiuni.
- Controla accesul la fișier și imprimantă.
- Permite unei aplicații să trimită sau să primească mesaje la sau de la un alt dispozitiv.

Serviciile **SMB** de printare și partajare de fișier au devenit sprijinul principal al rețelisticiei Microsoft. O dată cu introducerea seriilor software Microsoft 2000, Microsoft a schimbat structura de bază a utilizării **SMB**. În versiunile anterioare ale produselor Microsoft, serviciile **SMB** foloseau un protocol non-TCP/IP pentru implementarea rezoluției de nume. Începând cu Windows 2000, toate produsele ulterioare folosesc DNS, ce permite ca protocolele TCP/IP să suporte direct partajarea de resursă **SMB**, aşa cum se vede în Fig. 1. Procesul de schimb de fișier **SMB** dintre PC-urile Windows este evidențiat în Fig. 2.

Spre deosebire de partajarea de fișier suportată de File Transfer Protocol (FTP), clienții stabilesc o conexiune pe termen lung cu serverele. După ce este stabilită conexiunea, utilizatorul client poate accesa resursele de pe server ca și cum ar fi local, pe hostul client.

Sistemele de operare LINUX și UNIX oferă de asemenea o metodă de partajare a resurselor cu rețelele Microsoft, folosind o versiune a **SMB** numită SAMBA. Sistemele de operare Apple Macintosh suportă de asemenea partajarea resurselor folosind protocolul **SMB**.



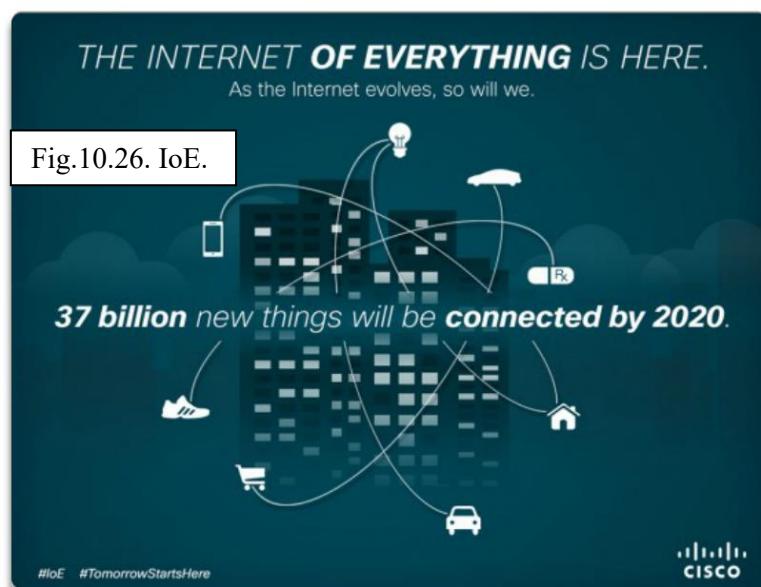
10.8 Mesajele pot fi auzite în întreaga lume

Nivelul aplicație este responsabil de accesarea directă a proceselor de bază pe care le gestionează și de livrarea comunicațiilor prin rețea. Acest nivel servește ca sursă și destinație a comunicațiilor peste rețele, indiferent de tipul de rețea de date utilizat. De fapt, progresele cu privire la modului în care ne conectăm la rețea au un efect direct al tipului de aplicații dezvoltate.

Tendințe ca Bring Your Own Device (BYOD), accesul de oriunde, virtualizarea și conexiunile machine-to-machine (m2m) au făcut loc unui noi tip de aplicații. Este estimat că aproximativ 50 miliarde de dispozitive să fie conectate în 2020. Numai în anul 2010, mai mult de 350.000 de aplicații au fost dezvoltate cu mai mult de trei milioane de descărări. Toate acestea conduc la o lume de conexiuni intuitive între oameni, procese, date și lucruri din rețea.

Utilizarea de smart-tagging și conectivitate avansată pentru a digitaliza produsele neinteligente – de la biciclete la sticle, frigidere și mașini – și conectarea lor la Internet, vor permite oamenilor și companiilor să interacționeze în moduri noi și aproape inimaginabile. Obiectele vor fi capabile să primească și să transmită informații utilizatorilor și altor obiecte conectate. Așa cum se poate vedea în Fig. , acest nou val din dezvoltarea Internetului este numit **Internet of Things** !

Peste 100 de milioane de automate, autoturisme, alarme de fum și alte dispozitive deja împart informații automat în zilele noastre, Fig. prezintă analiștii de market de la **Berg Insight** care se asteaptă să crească la 360 de milioane până în 2016. Astăzi, fotocopiatore cu un modul M2M pot cere automat hârtie și toner nou sau pot alerta tehnicienii de un defect – chiar și să le spună ce piese să aducă.



Explozia masivă de aplicații se datorează în mare parte geniului de abordare pe nivele pentru procesarea datelor dintr-o rețea. Mai exact, păstrarea funcionalității nivelului aplicație separat de funcționarea transportului de date, permite ca protocolele de la nivelul aplicație să fie schimbată și noi aplicații să fie dezvoltate, fără ca dezvoltatorul să se îngrijoreze de mecanismele de transport al datelor peste Internet. Aceasta este funcția altor nivele și prin urmare, altor dezvoltatori.

Așa cum se arată în Fig. , atunci când o aplicație trimie o cerere la o aplicație server, mesajul este construit de către nivelul aplicație, însă pasează datele la funcionalitățile altor nivele de pe client pentru livrare. În călătoria lor prin stivă, fiecare nivel încapsulează datele cu un header ce conține protocolele de comunicație pentru respectivul nivel. Aceste protocole, ce

sunt implementate și pe hostul sursă și pe cel destinație, interacționează pentru a oferi livrare end-to-end a aplicațiilor peste rețea.

Protocoloale ca HTTP, de exemplu, suportă livrarea paginilor web la dispozitivele finale. Acum că am învățat toate nivelele și funcțiile lor diferite, putem urmări o cerere de client a unei pagini web de la serverul web pentru a vedea cum aceste funcționalități independente lucrează împreună.

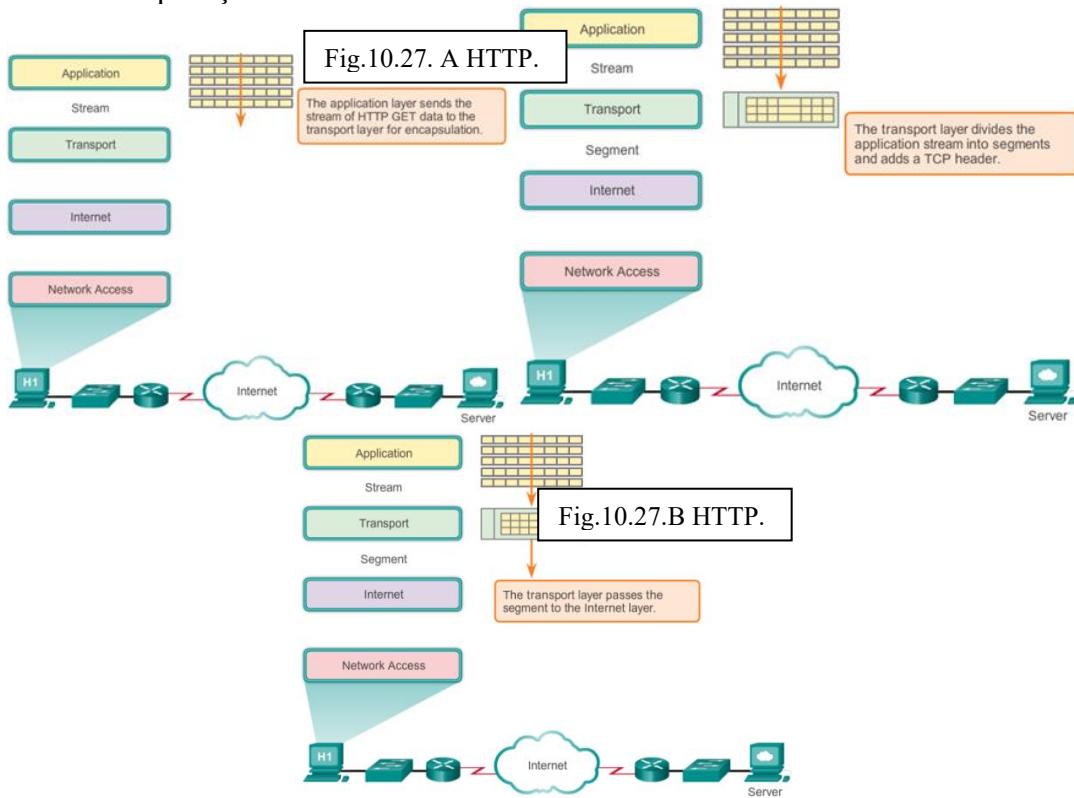
Folosind modelul TCP/IP, un proces complet de comunicare include șase pași:

PASUL 1. Crearea datelor

Primul pas este crearea datelor la nivelul aplicație al dispozitivului sursă. În acest caz, După lansarea cererii clientului web, cunoscută ca HTTP **GET**, datele respective vor fi codate, comprimate și criptate dacă este necesar. Aceasta este sarcina protocolului nivelului aplicație din modelul TCP/IP – însă acesta include funcționalitatea descrisă de către nivelele aplicație, prezentare și sesiune ale modelului OSI. Nivelul aplicație trimite datele ca un stream la nivelul transport.

PASUL 2. Segmentarea și încapsularea inițială

Următorul pas este segmentarea și încapsularea datelor în parcurgerea stivei de protocole. La nivelul transport, mesajul HTTP **GET** va fi împărțit în mai multe piese mai mici și ușor gestionabile și fiecare parte a mesajului va avea un header de nivel transport atașată ei. În interiorul headerului de la nivelul transport sunt indicatori a modului în care se poate reconstrui mesajul. Include de asemenea și un identificator, număr de port 80. Acesta este utilizat pentru a spune serverului destinație că mesajul este destinat pentru aplicația de server web. Un port sursă generat aleator este adăugat pentru a asigura faptul că și clientul poate urmări comunicația și că va fi transmisă la aplicația client corectă.



PASUL 3. Adresarea

La acest pas, identificatori de adresă sunt adăugați la segmente. Așa cum aici sunt mai multe nivele de protocole ce preparam datele pentru transferul la destinație, există mai multe nivele de adresare pentru asigurarea livrării. Rolul nivelului rețea este adăugarea adresării ce permite transferul de date de la hostul sursă la hostul ce le va utiliza. Nivelul rețea realizează acest lucru prin încapsularea fiecărui segment cu un header de pachet IP. Headerul de pachet IP conține adresele IP ale dispozitivelor sursă și destinație (adresa IP a dispozitivului destinație este de obicei determinată printr-un proces anterior numit *domain name service*). Combinarea de adresă sursă IP și destinație cu numărul de port sursă și destinație se numește socket. Socketul este folosit pentru a identifica serverul și serviciul cerut de către client.

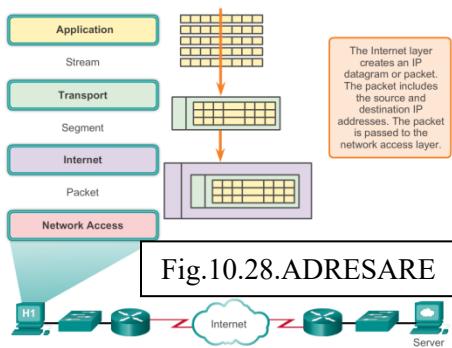


Fig.10.28.ADRESARE

PASUL 4. Pregătirea pentru transport

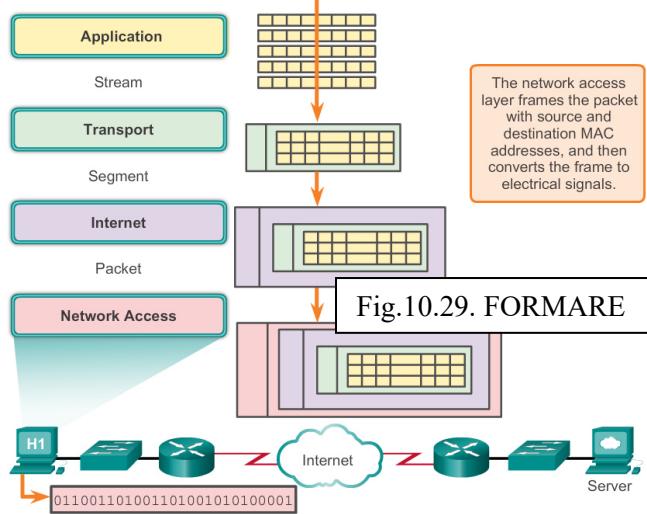
După ce este adăugată adresa IP, pachetul este pasat la nivelul acces la rețea pentru generarea datelor pe mediu, așa cum se poate vedea în Fig. . Pentru a se realiza acest lucru, nivelul acces la rețea trebuie mai întâi să pregătească pachetul pentru transmisie prin plasarea lui într-un frame cu un header și trailer. Acest frame include adresa de host fizică a sursei, adresa fizică a next hop din calea sa spre destinație. Acest lucru este echivalent cu funcționalitatea nivelului 2, sau nivelul legătură de date, a modelului OSI. Nivelul 2 se ocupă de livrarea mesajelor pe o rețea locală. Adresa de nivel 2 este unică pe rețea locală și reprezintă adresa dispozitivului final pe mediul fizic. Într-un LAN ce folosește Ethernet, această adresă se numește adresa Media Access Control (MAC). O dată ce nivelul acces la rețea a pregătit frameul cu adresele sursă și destinație, codifica frameul în biți și apoi în impulsuri electrice sau luminoase ce sunt transmise pe mediul de comunicație.

PASUL 5. Transportul datelor

Datele sunt transportate prin internetwork, ce constă din mediu și orice dispozitive intermediare. În drumul mesajului încapsulat prin rețea poate parcurge mai multe tipuri de medii de comunicație diferite de rețea. Nivelul de acces la rețea specifică tehnicele de plasare și scoatere a frameului din fiecare mediu, cunoscute ca metode de control al accesului la mediu.

Dacă hostul destinație se află în aceeași rețea cu hostul sursă, pachetul este livrat între cele două hosturi din mediul local fără nevoie unui router. Însă, dacă hostul destinație și cel sursă se află în rețele diferite, pachetul ar putea traversa mai multe rețele, pe mai multe tipuri diferite de medii de comunicație, peste mai multe routere. În călătoria prin rețea, informațiile conținute în frame nu sunt alterate.

La limita fiecărei rețele locale, un dispozitiv de rețea intermediar, de obicei un router, decapsulează frameul pentru a citi adresa de host destinație conținută în headerul pachetului. Routerele folosesc partea de identificator de rețea a adresei pentru a determina ce cale să folosească pentru a ajunge la hostul destinație. O dată ce este determinată calea, routerul încapsulează pachetul într-un nou frame și îl trimită la următorul hop din drumul spre dispozitivul destinație.



PASUL 6. Livrarea datelor la aplicația destinație corectă

La final, la dispozitivul destinație, frameul este primit. Decapsularea și reasamblarea datelor are loc, în timpul călătoriei datelor în stiva dispozitivului destinație. Datele sunt pasate continuu la nivelele superioare, de la nivelul de acces la rețea la nivelul rețea, nivelul transport până la nivelul aplicație unde vor fi procesate.

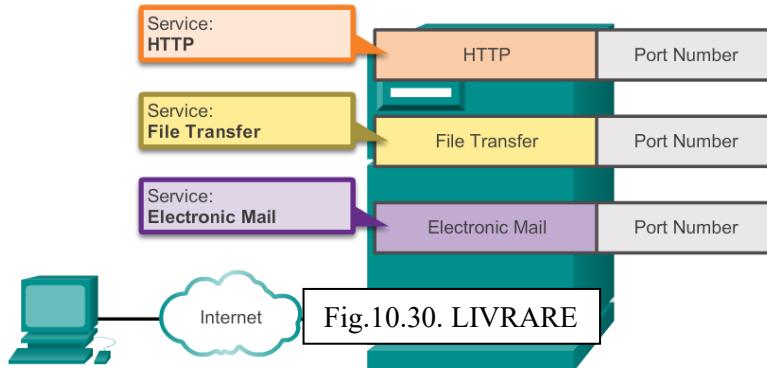
Dar cum poate fi sigur dispozitivul că este identificat procesul aplicație correct?

Așa cum se poate vedea în Fig., reamintim faptul că la nivelul transport, informațiile conținute în headerul PDU identifică procesul specific sau serviciul ce rulează pe dispozitivul de host destinație ce va procesa datele. Hosturile, fie că sunt clienți, fie că sunt servere din Internet, pot rula mai multe aplicații de rețea simultan. Oamenii ce utilizează PCuri au de obicei un client de e-mail ce rulează în același timp cu un browser web, program de mesagerie instant, streaming media și eventual un joc. Toate aceste programe ce rulează separat sunt exemple de procese individuale.

Vizualizarea unei pagini web implică cel puțin un proces de rețea. Apasarea pe un hyperlink face ca un browser web să comunice cu un server web. În același timp, în background, un client de e-mail poate transmite și primi e-mailuri și un coleg sau prieten poate trimite un mesaj instant.

Să ne imaginăm un computer ce are numai o interfață de rețea conectată la el. Toate fluxurile de date create de către aplicațiile ce rulează pe un PC intră și ies prin respectiva interfață, însă mesajele instant nu apar în mijlocul documentelor de procesor word, iar e-mailurile nu apar într-o interfață a unui joc.

Acest lucru se datorează faptului că procesele individuale ce rulează pe hosturile sursă și destinație comunică între ele. Fiecare aplicație sau serviciu este reprezentat la nivelul 4 printr-un număr de port. Un dialog unic între dispozitive este identificat cu o pereche de numere de port destinație și sursă de nivel 4, reprezentative pentru cele două aplicații ce comunică. Atunci când datele sunt primite pe host, numărul de port este examinat pentru a determina ce aplicație sau proces este optimă pentru date.



O resursă distractiva pentru a te ajuta să vizualizezi conceptele de rețea este filmul de animație "Warriors of the Net" de la TNG Media Lab. Înainte vizualizării video, există câteva lucruri de luat în calcul. Mai întâi, în termenii conceptelor invătate în acest capitol, gandeste-te când în video este într-un LAN sau WAN sau intranet sau Internet; ce sunt dispozitivele finale în comparație cu cele intermediare; cum modelele OSI și TCP/IP se aplică; ce protocoale sunt implicate.

Apoi, pe când numerele de port 21, 23, 25, 53 și 80 ne sunt referite explicit în video, adresele IP sunt referite implicit – poti vedea unde? Unde în video ar putea fi implicate adresele MAC?

La sfârșit, deoarece toate animatiile au de obicei simplificări în ele, există o eroare în video. Pe la minutul 5, afirmația "What happens when Mr. IP doesn't receive an acknowledgement, he simply sends a replacement packet." este făcută. Aceasta nu este o funcție a nivelului 3 IP, ce este un protocol de livrare best effort și unreliable, ci mai degrabă o funcție a protocolului TCP de nivel transport.

Descarcati video de la <http://www.warriorsofthe.net>.



10.9 Concluzii Capitolul 10



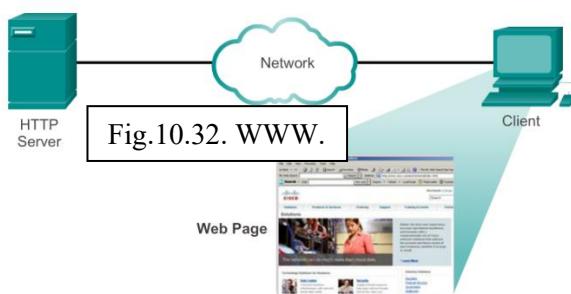
Nivelul aplicație este responsabil de accesarea directă a proceselor de bază pe care le gestionează și livrarea comunicațiilor din rețeaua umană. Acest nivel servește ca sursă și destinație a comunicațiilor din rețelele de date. Aplicațiile, serviciile și protocoalele nivelului aplicație permit utilizatorilor să interacționeze cu rețeaua de date într-un mod ce are înțeles și eficiență.

- *Aplicațiile sunt programe de computer în care utilizatorul interacționează și care inițiază procesul de transfer de date la cererea utilizatorului.*
- *Serviciile sunt programe de background ce oferă conexiune între nivelul aplicație și nivelele inferioare ale modelului de rețea.*
- *Protocoalele oferă o stivă de reguli stabilite și procese ce asigură serviciile ce rulează pe un dispozitiv particular ce poate trimite și primi date de la un rangu de dispozitive de rețea diferite.*

Livrarea de date peste rețea poate fi cerută de la un server de către un client sau între dispozitive ce funcționează într-un aranjament **p2p**, unde relația client/server este stabilită, în funcție de ce dispozitiv este destinație și sursă la un moment dat. Mesajele sunt schimbate între serviciile de nivel aplicație de pe fiecare dispozitiv final în concordanță cu specificațiile de protocol stabilite și de utilizarea acestor relații.

Protocoale cum ar fi HTTP, de exemplu, suportă livrarea paginilor web la dispozitivele finale. SMTP și POP suportă trimiterea și primirea de e-mail. **SMB** și FTP permit utilizatorilor să împartă fișiere. Aplicațiile P2P fac mai ușor pentru consumatori să împartă mediul într-un mod distribuit. DNS rezolvă numele lizibile umane utilizate pentru referirea resurselor de rețea în adrese numerice folosite de către rețea. Norii sunt locații upstream de la distanță ce stochează datele și aplicațiile de host pentru ca utilizatorii să nu necesite aşa multe resurse locale și pentru ca utilizatorii să acceseze conținutul de pe diferite dispozitive, din orice locație.

Toate aceste elemente funcționează împrumă la nivelul aplicație. Nivelul aplicație permite utilizatorilor să lucreze și să se joace împreună peste Internet.



CAPITOLUL 11. ESTE O REȚEA

Introducere

Până în acest moment al cursului, am examinat serviciile ce pot fi oferite, de către o rețea de date, rețelei umane, caracteristicile fiecărui nivel al modelului OSI și funcțiile protoocoalelor TCP/IP și am descris nivelul Ethernet în detaliu, tehnologie LAN universală. Următorul pas este de a învăța asamblarea acestor elemente împreună într-o rețea funcțională și care trebuie menținută activă.

Notă: Studenții pot lucra individual, în perechi sau întreaga clasă poate completa această activitate împreună.

11.1 Creare și Dezvoltare. Echipamentele în Rețelele Mici

Majoritatea afacerilor sunt afaceri mici. Deci, nu este o surpriză faptul că majoritatea rețelelor sunt rețele mici.

În rețelele mici proiectarea rețelei este de obicei simplă. Numărul și tipul de dispozitive din rețea sunt reduse semnificativ în comparație cu o rețea mai mare. Topologiile de rețea pentru rețelele mici implică de obicei un singur router și unul sau mai multe switchuri. Rețelele mici ar putea avea de asemenea puncte de acces wireless (posibil construite într-un router) și telefoane IP. Ca și conexiune la Internet, în mod normal o rețea mică are o singură conexiune WAN oferită printr-o tehnologie DSL, cablu sau o conexiune Ethernet.

Gestionarea unei rețele mici necesită aceleași aptitudini ca în gestionarea unei rețele mari. Cea mai mare parte a muncii este axată pe gestionarea și depanarea echipamentului existent, cât și pe securitatea dispozitivelor și a informațiilor din rețea. Administrarea unei rețele mici este efectuată fie de un angajat al companiei, fie de o persoană angajată prin contract de către companie, în funcție de dimensiunea afacerii și de tipul de afacere.

O rețea tipică pentru o afacere mică este prezentată în Fig. 11.1.

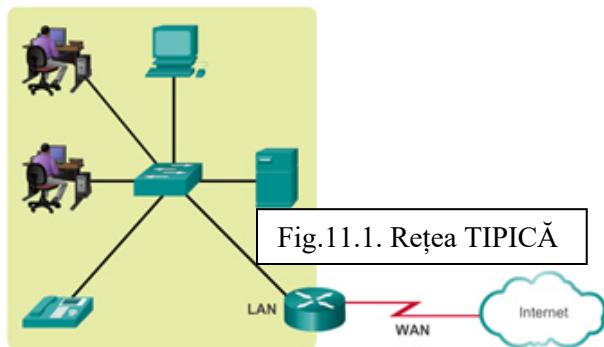


Fig. 11.1. Rețea tipică pentru o afacere mică

Pentru a îndeplini cerințele utilizatorului, chiar și rețelele mici necesită planificare și proiectare. Planificarea asigură faptul că toate cerințele, factorii de cost și opțiunile de dezvoltare sunt luate în considerare.

Una dintre primele considerații de proiectare atunci când implementăm o rețea mică este tipul de dispozitive intermediare folosite pentru suportul rețelei. La alegerea tipului de dispozitive intermediare există un număr de factori ce trebuie să fie luați în considerare, aşa cum se poate observa și în Fig. 11.2.

Costul – Costul este în mod normal unul dintre cei mai importanți factori atunci când alegem echipamentul pentru o rețea de dimensiune mică. Costul unui switch sau router este determinat de capacitatea și caracteristicile sale. Capacitatea dispozitivului include numărul și tipuri de porturi disponibile și viteza de backplane. Alți factori ce au impact asupra costului sunt capabilitățile de management ale rețelei, tehnologiile de securitate integrate și tehnologiile optionale de switching avansate. Un alt element cheie ce afectează costul este cătă redundanță să încorporăm în rețea – include dispozitivele, porturile de pe un dispozitiv și cablajul prin cupru sau fibră optică.

Viteza și tipurile de porturi/interfețe – Alegerea numărului și tipurilor de porturi de pe un router sau switch este o decizie critică. Întrebările ce trebuie să fie puse includ :

- “Să comandăm porturi destule pentru nevoile de astăzi sau să luăm în considerare cerințele de creștere ?”
- “Avem nevoie de un amestec de viteză UTP ?”
- “Avem nevoie de ambele tipuri de porturi, UTP și de fibră ?”

Computerele mai noi au încorporate Plăci de Rețea (NIC) de 1 Gbps. Porturile de 10 Gbps sunt deja incluse în unele stații de lucru și servere. Deși sunt mai costisitoare, alegerea dispozitivelor de nivel 2 ce se pot adapta vitezelor în creștere permit rețelei să evolueze fără înlocuirea dispozitivelor centrale.

Extensibilitatea – Dispozitivele de rețea au configurații fizice atât fixe cât și modulare.

Configurațiile fixe au un număr și tip de porturi sau interfețe specific.

Dispozitivele modulare au sloturi de expansiune ce oferă flexibilitate pentru adăugarea de noi module o dată cu extinderea cerințelor. Multe dispozitive modulare au la început un număr de porturi fixe, cât și de sloturi de expansiune. Switchurile sunt disponibile cu porturi suplimentare speciale pentru legături de mare viteză optionale. De asemenea, deoarece routerele pot fi utilizate pentru conectarea unor numere și tipuri diferite de rețele, trebuie să avem grijă să alegem modulele și interfețele adecvate pentru mediul respectiv.

Întrebări ce trebuie să fie puse sunt:

- “Comandăm dispozitive cu module upgradabile ?”
- “Ce tip de interfețe WAN, dacă există, sunt necesare pe router (routere) ?”



Fig.11.2. FACTORI

11.2 Serviciile și Caracteristicile Sistemului de Operare

În funcție de versiunea sistemului de operare, un dispozitiv de rețea poate suporta anumite caracteristici și servicii, cum ar fi :

- **Securitate – Security.**
- **Calitatea Serviciilor – QoS.**
- **Voce peste Protocolul de Internet – VoIP.**
- **Comutare la nivelul 3 din stiva OSI - Layer 3 switching.**
- **Translatarea Adreselor de Rețea – NAT.**
- **ConFig.rea dinamică a gazdelor – DHCP.**

Routerele pot fi costisitoare din punct de vedere al costurilor, în funcție de interfețele și caracteristicile necesare. Modulele suplimentare, cum ar fi cel de fibră optică, cresc costul dispozitivelor de rețea.

La implementarea unei rețele mici, este necesară planificarea spațiului de adresare IP. Toate hosturile dintr-un internetwork trebuie să aibă o adresă unică. Chiar și într-o rețea mică, atribuirea de adrese din rețea nu trebuie să fie întâmplătoare. Schema de adresare IP trebuie să fie planificată, documentată și menținută în funcție de tipul de dispozitive ce primesc adresa.

Exemple pentru diferite tipuri de dispozitive ce vor face parte din proiectarea alocării IP :

- *Dispozitivele finale pentru utilizatori.*
- *Servere și periferice.*
- *Hosturi accesibile din Internet.*
- *Dispozitivele intermediare.*

Planificarea și documentarea schemei de adresare IP ajută administratorul să urmărească tipurile de dispozitive. De exemplu, dacă toate serverele au atribuite o adresă de host din rangeul de la 50 la 100, este ușoară identificarea traficului de server prin adresa IP. Acest lucru poate fi foarte util la depanarea problemelor traficului de rețea cu ajutorul unui analizor de protocol.

În plus, pentru administratori devine mai ușor să controleze accesul la resursele dintr-o rețea în funcție de adresa IP cu folosirea unei scheme de adresare IP deterministă. Acest lucru poate fi important pentru hosturile ce oferă resurse într-o rețea internă cât și într-o rețea externă. Serverele de web sau e-commerce joacă acest rol. Dacă adresele pentru aceste resurse nu sunt planificate și documentate, securitatea și accesibilitatea la dispozitive nu sunt controlate eficient. Dacă un server are o adresă atribuită aleator, blocarea accesului la această adresă este dificilă, iar clienții ar putea să nu fie capabili să localizeze această resursă.

Fiecare dintre aceste tipuri de dispozitive diferite ar trebui să aibă alocat un bloc logic de adrese din spațiul de adresă al rețelei.



Fig. 11.3. Planificare și asignare adrese IP în funcție de echipament.

O altă parte importantă a designului de rețea este fiabilitatea. Chiar întreprinderile mici se bazează adesea pe rețeaua lor puternică pentru activitatea economică. Un eșec al rețelei poate fi foarte costisitor. Pentru a menține un grad ridicat de fiabilitate, redundanța este necesară în proiectarea rețelei. Redundanța ajută la eliminarea punctelor unice de eșec. Există mai multe moduri de a realiza redundanță într-o rețea. Redundanța poate fi realizată prin instalarea echipamentelor după, dar poate fi de asemenea realizată prin furnizarea de legături de rețea după pentru zonele critice, așa cum se arată în Fig. 11.4.

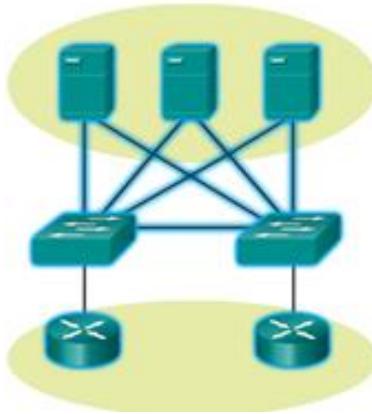


Fig. 11.4 Redundanță într-o "fermă de servere"

Cu cât este mai mică rețeaua, cu atât este mai mică șansa ca redundanța de echipament să fie accesibilă. Prin urmare, un mod normal de introducere a redundanței este utilizarea de conexiuni redundante switchului între mai multe switchuri de rețea și între switchuri și routere.

De asemenea, serverele adesea au mai multe NICuri ce permit conexiuni redundante la unul sau mai multe switchuri. Într-o rețea mică, serverele sunt dezvoltate adesea ca servere web, servere de fișiere sau servere de e-mail.

Rețelele mici de obicei oferă un punct unic de ieșire spre Internet prin una sau mai multe porți implicite (default gateway). Cu un singur router în topologie, singura redundanță în termenii de căi de nivel 3 este stabilită prin utilizarea a mai multor interfețe Ethernet de pe un router. Însă, dacă routerul "pică", întreaga rețea își pierde conectivitatea la Internet. Din acest motiv, ar putea fi un sfat bun pentru o întreprindere mică să plătească pentru un cont de opțiune de cost-scăzut pentru un al doilea furnizor de servicii pentru "backup".

Utilizatorii se așteaptă la acces imediat la e-mailuri și la fișierele ce le paratajează sau actualizează. Pentru a ajuta asigurarea acestei disponibilități, dezvoltatorul de rețea ar trebui să parcurgă următorii pași :

Pasul 1. Securizarea serverelor de fișier și e-mail într-o locație centralizată.

Pasul 2. Protejarea locației față de accesul neautorizat prin implementarea măsurilor de securitate fizică și logică.

Pasul 3. Crearea de redundanță în "server farm" pentru a asigura faptul că dacă un dispozitiv "pică", fișierele nu sunt pierdute.

Pasul 4. ConFig.rea cailor redundante spre servere.

În plus, rețelele moderne folosesc adesea unele forme de voce și video peste IP pentru comunicarea cu clienții și partenerii de afaceri. Acest tip de rețea convergentă este implementată ca o soluție integrată sau ca o formă suplimentară de date brute suprapuse peste o rețea IP. Administratorul de rețea ar trebui să ia în considerare tipurile variate de trafic și tratarea lor în designul de rețea. Routerul (routerele) și switchul (switchurile) dintr-o rețea mică ar trebui să fie

conFig.te pentru a suporta traficul în timp real, cum ar fi voce sau video, într-o manieră distinctă în comparație cu traficul de date. De fapt, un design de rețea bun va clasifica traficul cu atenție în funcție de prioritate, aşa cum se vede în Fig. 11.5.

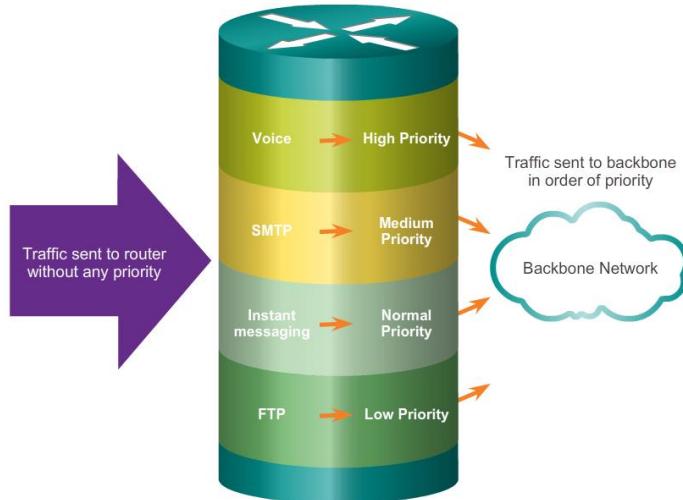


Fig. 11.5 Prioritizarea traficului

Clasele de trafic pot fi :

- *Transferul de fișiere.*
- *E-mail.*
- *Voce.*
- *Video.*
- *Mesagerie.*
- *Tranzacții.*

În final, scopul unui design de rețea bun, chiar și pentru o rețea mică, este de a asigura productivitatea pentru angajați și de a minimiza defecțiunile în timp ale rețelei.

Rețeaua este la fel de utilă ca aplicațiile ce sunt pe ea. Așa cum se vede în Fig. 11.6, la nivelul aplicație există două forme de programe software sau procese ce oferă acces la rețea: aplicațiile de rețea și serviciile de nivel aplicație.

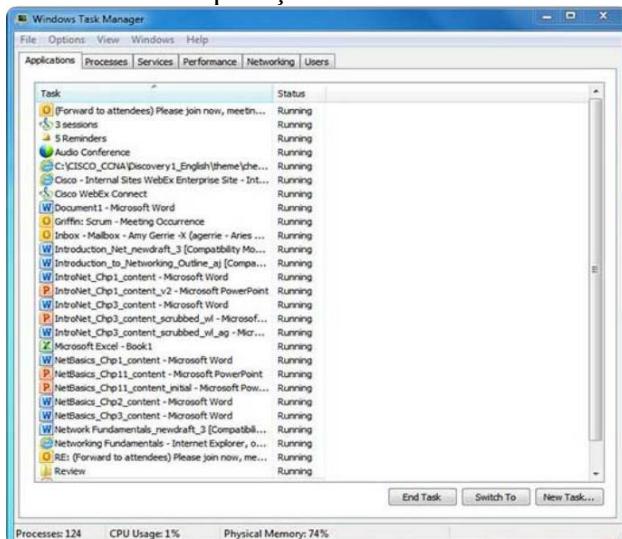


Fig. 11.6 Procese la nivelul aplicație

11.2.1 Aplicațiile de rețea

Aplicațiile sunt programe software folosite pentru comunicarea peste rețea. Unele aplicații end-user sunt "network-aware", ceea ce înseamnă că ele implementează protocoale de nivel aplicație și sunt capabile să comunice direct peste nivelele inferioare ale stivei de protocoale. Clientii de e-mail și browserele web sunt exemple de acest tip de aplicație.

11.2.1.1 Serviciile de nivel aplicație

Alte programe ar putea avea nevoie de asistență a serviciilor de nivel aplicație pentru a utiliza resursele de rețea, cum ar fi transferul de fișiere sau "spooling" de imprimare prin rețea. Deși transparent pentru un angajat, aceste servicii sunt programe ce interfațează cu rețeaua și pregătesc datele pentru transfer. Tipuri diferite de date, fie text, grafice sau video, necesită servicii de rețea diferite pentru a asigura faptul că sunt pregătite corect pentru procesare de către funcțiile ce au loc la nivelele inferioare ale modelului OSI.

Fiecare aplicație sau serviciu de rețea folosește protocoale ce definesc standarde și formate de date ce vor fi utilizate. Fără protocoale, rețeaua de date nu ar avea un mod comun de formatare și direcționare a datelor. Pentru a înțelege funcționarea serviciilor diferite de rețea, este necesar să devem familiarizați cu protocoalele de bază ce guvernează funcționarea lor.

Cea mai mare parte a muncii unui tehnician, fie într-o rețea mică, fie într-o rețea mare, va fi în modul de implicare cu protocoalele de rețea. Protocoalele de rețea suportă aplicațiile și serviciile folosite de către angajații dintr-o rețea mică. Protocoale comune de rețea sunt:

- *DNS.*
- *Telnet.*
- *IMAP, SMTP, POP (e-mail).*
- *DHCP.*
- *HTTP.*
- *FTP.*

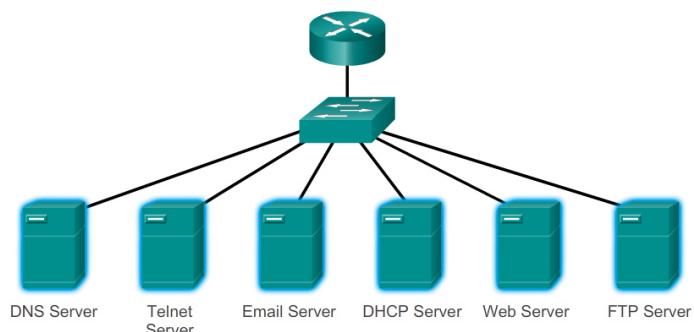


Fig. 11.7 Servere pentru protocoale comune de rețea

Acstea protocoale de rețea cuprind setul de instrumente fundamental pentru un profesionist în rețelistică. Fiecare dintre aceste protocoale definesc :

- *Procesele de la fiecare capăt al unei sesiuni de comunicare.*
- *Tipurile de mesaje.*
- *Sintaxa mesajelor.*
- *Înțelesul câmpurilor informaționale.*
- *Modul în care mesajele sunt trimise și răspunsul așteptat.*
- *Interacțiunea cu nivelul următor inferior.*

Multe întreprinderi au stabilit o politică de folosire a versiunilor securizate ale acestor protocole, acolo unde este posibil, cum ar fi protocolele : HTTPS, SFTP și SSH.

Suplimentar față de protocolele comune de rețea descrise anterior, întreprinderile moderne, chiar și cele mici, folosesc în mod normal aplicații în timp real pentru comunicarea cu clienții și partenerii de business. Deoarece o companie mică nu ar fi capabilă să justifice costul unei soluții de tip "Enterprise Cisco Telepresence", există alte aplicații în timp real, ca cele prezentate în Fig. 11.8, ce sunt convenabile din punct de vedere al prețului și justificabile pentru organizațiile din întreprinderile mici.



Fig. 11.8 Aplicații de comunicare în timp real

Aplicațiile în timp real necesită o planificare mai amănunțită și servicii dedicate (în comparație cu alte tipuri de date) pentru a asigura prioritatea livrării traficului de voce și video. Acest lucru înseamnă că administratorul de rețea trebuie să asigure echipamentul adecvat instalat în rețea și faptul că dispozitivele de rețea sunt configurate pentru a asigura prioritatea livrării. Fig. 11.9 prezintă elementele unei rețele mici ce suportă aplicații în timp real.



Fig. 11.9 Echipamentele de rețea mai puțin văzute de utilizatori

11.2.1.2 Infrastructură

Pentru a suporta aplicațiile în timp real existente și propuse, infrastructura trebuie să cuprindă caracteristicile fiecărui tip de trafic. Dezvoltatorul de rețea trebuie să determine dacă switchurile și cablajul existent pot suporta traficul ce va fi adăugat în rețea. Cablarea ce poate suporta transmisii la viteze gigabit ar trebui să fie capabilă să transporte traficul generat și să nu solicite nici-o schimbare în infrastructură. Alte switchuri ar putea să nu suporte tehnologia "Power over Ethernet" (PoE). Cablarea învechită ar putea să nu suporte cerințele de lățime de bandă. Switchurile și cablarea vor trebui actualizate pentru a suporta aceste aplicații.

11.2.1.3 VoIP

Tehnologia VoIP este implementată într-o organizație ce folosește încă telefoanele tradiționale. VoIP utilizează routere voice-enabled. Aceste routere convertesc vocea analogică din semnalele de telefonie tradițională în pachete IP. După ce semnalele sunt convertite în pachete IP, routerul trimite aceste pachete între locațiile respective. VoIP este mai puțin costisitor decât o soluție integrată de telefonie IP, însă calitatea comunicației lor nu îndeplinește aceleași standarde. Soluțiile de video și VoIP pentru întreprinderile mici pot fi realizate, de exemplu, cu Skype sau versiuni non-enterprise ale Cisco WebEx.

11.2.1.4 IP Telephony

În telefonia IP, telefonul IP efectuează conversia voce-în-IP. Routerele voice-enabled nu sunt necesare într-o rețea cu o soluție integrată de telefonie IP. Telefoanele IP folosesc un server dedicat pentru controlul apelului și signaling. Există acum mulți furnizori cu soluții dedicate de telefonie IP pentru rețelele mici.

11.2.1.5 Aplicații în timp real

Pentru a transporta streaming media eficient, rețeaua trebuie să fie capabilă să suporte aplicații ce necesită livrare sensibilă la întârziere. Real-Time Transport Protocol (RTP) și Real-Time Transport Control Protocol (RTCP) sunt două protocole ce suportă această cerință. RTP și RTCP permit controlul și scalabilitatea resurselor de rețea prin mecanisme de "Quality of Service" (QoS) încorporate. Aceste mecanisme QoS oferă instrumente prețioase pentru minimizarea problemelor de latență pentru aplicațiile de streaming în timp real.

11.2.1.6 Creșterea rețelelor mari

Creșterea este un proces natural pentru multe întreprinderi mici, iar rețelele lor trebuie să crească și ele. Un administrator de rețea pentru o întreprindere mică lucrează fie reactiv, fie proactiv, în funcție de conducătorii companiei, ce includ adesea și administratorul de rețea. Ideal, administratorul de rețea are destul timp de gândire pentru a lua decizii inteligente cu privire la creșterea rețelei o dată cu creșterea companiei.

Pentru scalarea unei rețele, mai multe elemente sunt necesare:

- **Documentarea rețelei** – topologia fizică și logică.
- **Inventarul echipamentelor** – listarea dispozitivelor folosite sau incluse în rețea.
- **Bugetul** – bugetul IT detaliat, inclusiv bugetul de achiziționare de echipamente pe an.
- **Analiza traficului** – protocole, aplicații, servicii și cerințele de trafic respective ar trebui să fie documentate.

Acste elemente sunt utilizate pentru a informa “luarea de decizii” ce însoțesc scalarea unei rețele mici.



Fig. 11.10 Scalarea rețelelor

Suportul și creșterea unei rețele mici necesită să fim familiari cu protocolele și aplicațiile de rețea ce rulează peste rețea. În timp ce un administrator de rețea are mai mult timp într-un mediu de rețea mică de analizare individuală a utilizării rețelei pentru fiecare dispozitiv de rețea, o abordare mai holistică cu unele tipuri de analizoare de protocole bazate pe hardware sau software este recomandată.

Așa cum se poate observa în Fig.11.11, analizatoarele de protocole permit unui profesionist de rețea să compileze rapid informațiile statistice cu privire la fluxurile de date dintr-o rețea.

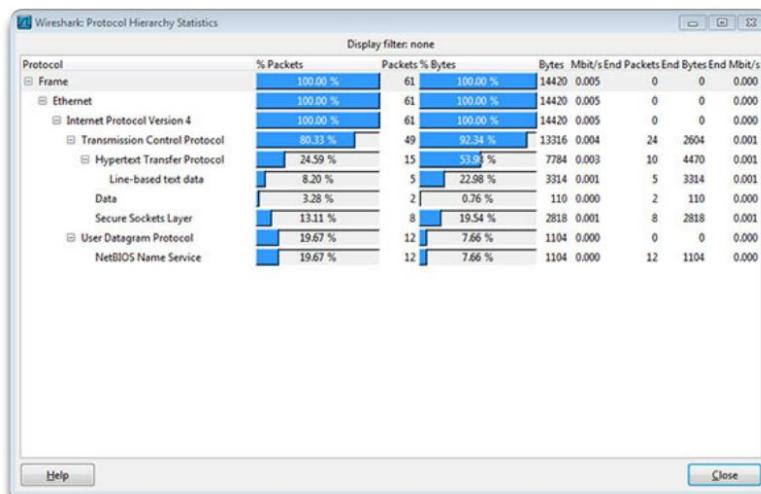


Fig. 11.11 Analizor protocole de rețea

La încercarea determinării modului în care gestionăm traficul, în special o dată cu creșterea rețelei, este important să înțelegem tipul de trafic ce traversează rețeaua, cât și fluxul de trafic curent. Dacă tipurile de trafic sunt necunoscute, analizorul de protocol va ajuta la identificarea traficului și a sursei sale.

Pentru a determina structurile fluxului de trafic, este important să :

- Captăm traficul în perioadele de utilizare de vârf pentru a primi o reprezentare bună a diferitelor tipuri de trafic.
- Efectuăm captura pe diferite segmente de rețea deoarece unele tipuri de trafic vor fi locale, pe un singur segment particular.

Informațiile adunate de către analizorul de protocol sunt analizate în funcție de sursă și destinația traficului, cât și de tipul de trafic transmis. Această analiză poate fi utilizată pentru a lăsa decizii cu privire la gestionarea traficului cât mai eficientă. Acest lucru se poate realiza prin reducerea fluxurilor de trafic inutil sau prin schimbarea patternurilor de flux prin mutarea pe un server, de exemplu.

Uneori, simpla realocare a unui server sau serviciu pe un alt segment de rețea îmbunătățește performanța rețelei și îndeplinește nevoile crescute de trafic. Alte ori, optimizarea performanței rețelei necesită reproiectarea rețelei și intervenția asupra traficului.

Pentru a înțelege tendințele de schimbare a traficului, un administrator de rețea trebuie să fie conștient de cum se schimbă utilizarea rețelei. Așa cum se poate observa în Fig. 11.12, un administrator de rețea dintr-o rețea mică are abilitatea de a obține "capturi de imagine" IT a aplicațiilor utilizate de angajat pe o perioadă de timp. Aceste "snapshots" includ de obicei informații cum ar fi:

- *Versiunea OS+OS.*
- *Aplicațiile non-rețea.*
- *Aplicațiile de rețea.*
- *Utilizarea CPU.*
- *Utilizarea driverului.*
- *Utilizarea RAM.*



Fig. 11.12 Procese Software

Documentarea prin intermediul snapshots pentru angajații dintr-o rețea mică pe o perioadă de timp va informa administratorul de rețea de creșterea cerințelor de protocole și fluxurile de date asociate. De exemplu, pot exista unii angajați ce folosesc resurse off-site cum ar fi mediul social pentru a poziționa mai bine o companie cu privire la marketing. Atunci când au

început să lucreze pentru companie, acești angajați se axau mai puțin pe reclama pe Internet. Această schimbare în utilizarea resurselor ar putea face ca administratorul de rețea să schimbe alocarea de resurse de rețea în mod corespunzător.

Este responsabilitatea administratorului de rețea să urmărească utilizarea rețelei și cerințele fluxului de trafic și să implementeze modificări de rețea pentru a optimiza productivitatea angajatului o dată cu creșterea rețelei și a întreprinderii.

11.3 Păstrarea rețelei în siguranță - Măsuri de securitate a dispozitivelor de rețea

Fie cablate, fie wireless, rețelele de calculatoare sunt esențiale pentru activitățile de zi cu zic. Indivizii și organizațiile depind de computerele și de rețelele lor. Pătrunderea unei persoane neautorizate poate avea ca rezultat intreruperi de rețea costisitoare și pierderi de informații. Atacurile la o rețea pot fi devastatoare și pot rezulta pierderi de timp și bani, datorită deteriorării sau furtului de informații importante sau bunuri.

Intrușii pot câștiga acces la rețea printr-o vulnerabilitate software, atacuri hardware sau prin ghicirea parolei și a numelui de utilizator ale unei persoane. Intrușii care câștigă acces prin modificarea software sau explorarea vulnerabilităților software se numesc hackeri.

După ce un hacker câștigă acces la rețea, pot apărea patru tipuri de amenințări:

- *Furtul de informații.*
- *Furtul de identitate.*
- *Pierdere/manipularea de date.*
- *Întreruperea de servicii.*

Chiar și în rețelele mici, este necesar să luăm în considerare amenințările de securitate și vulnerabilitățile atunci când planificăm implementarea unei rețele.

Când ne gândim la securitatea rețelei, sau securitatea computerului, ne putem imagina atacatorii ce explorează vulnerabilități software. O vulnerabilitate egală în importanță este securitatea fizică a dispozitivelor, aşa cum se poate observa și în Fig. 11.13 Un atacator poate refuza utilizarea resurselor de rețea dacă acele resurse pot fi compromise fizic.



Fig. 11.13 Tipuri de amenințări

Cele patru clase de amenințări fizice sunt:

- **Amenințări hardware** – deteriorarea fizică a serverelor, routerelor, switchurilor, cablării și stațiilor de lucru.
- **Amenințările de mediu** – temperaturile externe (prea frig sau prea cald) sau umiditatea extremă (prea uscat sau prea umed).
- **Amenințările electrice** – vârfurile de tensiune, insuficientă tensiune de alimentare, putere necondiționată (zgomot) și pierderea totală de putere.

- **Amenințări de întreținere** - manipulare necorespunzătoare a componentelor electrice (descărcare electrostatică), pierdere a pieselor de schimb critice, cablare slabă și etichetare slabă.

Unele dintre aceste probleme trebuie să fie abordate într-o politică organizațională. Unele dintre ele reprezintă subiectul unei bune conduceri și un management adecvat într-o organizație.

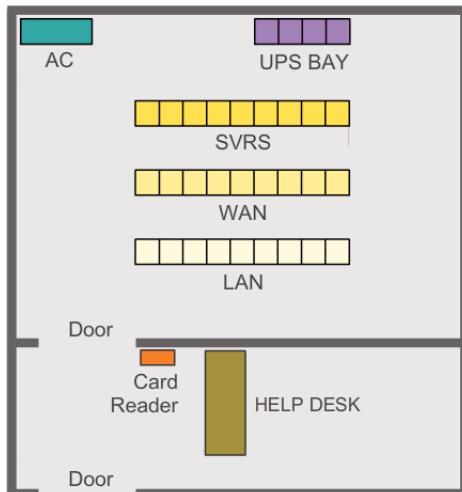


Fig. 11.14 Plan pentru securizarea fizică pentru a limita distrugerea echipamentelor
Trezi factori de securitate a rețelei sunt :

- **Vulnerabilitatea**-reprezintă gradul de slăbiciune ce este inherent în orice rețea și dispozitiv. Include routerele, switchurile, desktopurile, serverele și chiar dispozitivele de securitate.
- **Amenințările** - includ oameni interesați și calificați în profitarea de fiecare slăbiciune de securitate. Asemenea indivizi continuă să caute noi slăbiciuni și breșe de securitate.
- **Atacurile** - Amenințările sunt realizate printr-o varietate de instrumente, scripturi și programe pentru a lansa atacuri asupra rețelei și dispozitivelor de rețea. În mod normal, dispozitivele de rețea predispușe atacurilor sunt "endpoints", cum ar fi serverele și computerele desktop.

Există trei vulnerabilități principale sau slăbiciuni:

- **Tehnologică**, evidențiată în Fig. 11.15 a

Network security weaknesses:	
TCP/IP protocol weakness	<ul style="list-style-type: none"> • Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), and Internet Control Message Protocol (ICMP) are inherently insecure. • Simple Network Management Protocol (SNMP) and Simple Mail Transfer Protocol (SMTP) are related to the inherently insecure structure upon which TCP was designed.
Operating system weakness	<ul style="list-style-type: none"> • Each operating system has security problems that must be addressed. • UNIX, Linux, Mac OS, Mac OS X, Windows Server 2012, Windows 7, Windows 8 • They are documented in the Computer Emergency Response Team (CERT) archives at http://www.cert.org
Network equipment weakness	Various types of network equipment, such as routers, firewalls, and switches have security weaknesses that must be recognized and protected against. Their weaknesses include password protection, lack of authentication, routing protocols, and firewall holes.

Fig. 11.15 a Vulnerabilitate Tehnologică

▪ De conFig.re, evidențiată în Fig. 11.15 b

Configuration Weakness	How the weakness is exploited
Unsecured user accounts	User account information may be transmitted insecurely across the network, exposing usernames and passwords to snoopers.
System accounts with easily guessed passwords	This common problem is the result of poorly selected and easily guessed user passwords.
Misconfigured Internet services	A common problem is to turn on JavaScript in Web browsers, enabling attacks by way of hostile JavaScript when accessing untrusted sites. IIS, FTP, and Terminal Services also pose problems.
Unsecured default settings within products	Many products have default settings that enable security holes.
Misconfigured network equipment	Misconfigurations of the equipment itself can cause significant security problems. For example, misconfigured access lists, routing protocols, or SNMP community strings can open up large security holes.

Fig. 11.15 b Vulnerabilitate de conFig.re

▪ Politica de securitate, evidențiată în Fig. 11.15 c

Policy Weakness	How the weakness is exploited
Lack of written security policy	An unwritten policy cannot be consistently applied or enforced.
Politics	Political battles and turf wars can make it difficult to implement a consistent security policy.
Lack of authentication continuity	Poorly chosen, easily cracked, or default passwords can allow unauthorized access to the network.
Logical access controls not applied	Inadequate monitoring and auditing allow attacks and unauthorized use to continue, wasting company resources. This could result in legal action or termination against IT technicians, IT management, or even company leadership that allows these unsafe conditions to persist.
Software and hardware installation and changes do not follow policy	Unauthorized changes to the network topology or installation of unapproved applications create security holes.
Disaster recovery plan is nonexistent	The lack of a disaster recovery plan allows chaos, panic, and confusion to occur when someone attacks the enterprise.

Fig. 11.15 c Vulnerabilitate prin politici de securitate

Toate cele trei vulnerabilități sau slăbiciuni pot duce la atacuri variate, inclusiv atacuri cu cod rău intenționat sau atacuri de rețea.

11.3.1 Vulnerabilități și atacuri de rețea

Atacurile cu cod rău intenționat includ un număr de tipuri de programe de computer ce au fost create cu intenția de a produce pierderi sau deteriorarea de date. Principalele trei tipuri de atacuri cu cod rău intenționat sunt :

- **Virusii** - Un virus este un software rău intenționat ce este atașat altui program pentru a executa o funcție particulară nedorită pe o stație de lucru. Un exemplu este un program ce este atașat la command.com (interpretorul primar pentru sistemele Windows) și șterge anumite fișiere și infectează orice alte versiuni ale command.com pe care le poate găsi.

Virusii necesită în mod normal un mecanism de livrare, un vector, cum ar fi un fișier zip sau unele fișiere executabile atașate unui e-mail, pentru a transporta codul de virus de la un sistem la altul. Elementul cheie ce distinge un vierme de computer de un virus de computer este faptul că interacțiunea umană este necesară pentru a facilita impreăștirea unui virus.

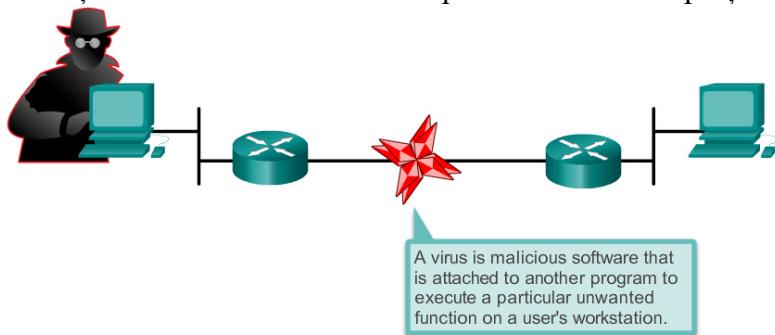


Fig. 11.16 Atac cu viruși

- **Cai Troieni** - Un cal Trojan este diferit numai prin faptul că întreaga aplicație a fost scrisă să arate ca altceva, însă este un instrument de atac. Un exemplu de cal Trojan este o aplicație

software ce rulează un simplu joc pe o stație de lucru. În timp ce utilizatorul este ocupat cu jocul, calul Troian trimite o copie a să fiecărei adrese din rangeul de adrese a utilizatorului. ceilalți utilizatori primesc jocul și îl joacă, astfel împărțind calul Troian tuturor adreselor lor.

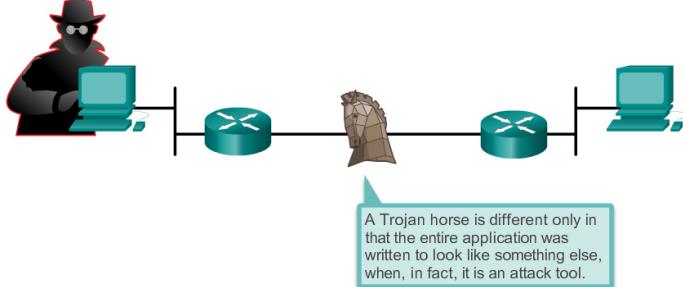


Fig. 11.17 Atac cu cai trojan

- **Viermii** – Viermii reprezintă programe ”self-contained” ce atacă un sistem și încearcă să exploreze o anumită vulnerabilitate de pe țintă. După explorarea cu succes a vulnerabilității, viermele copiază programul său de pe hostul atacat la noul sistem explorat pentru a reîncepe ciclul. Anatomia unui atac cu vierme este:
 - *Vulnerabilitatea permisă* – Un vierme se instalează prin explorarea unei vulnerabilități cunoscute în sisteme, cum ar fi utilizatorii naivi ce deschid atașamente executabile neverificate din emailuri.
 - *Mecanism de propagare* – După câștigarea accesului la un host, un vierme se copiază pe hostul respectiv, apoi caută noi ținte.
 - *Payload* – După ce un host este infectat cu un vierme, atacatorul are acces la host, de obicei ca un utilizator privilegiat. Atacatorii pot folosi o exploatare locală pentru a crește nivelul lor privilegiat la administrator.

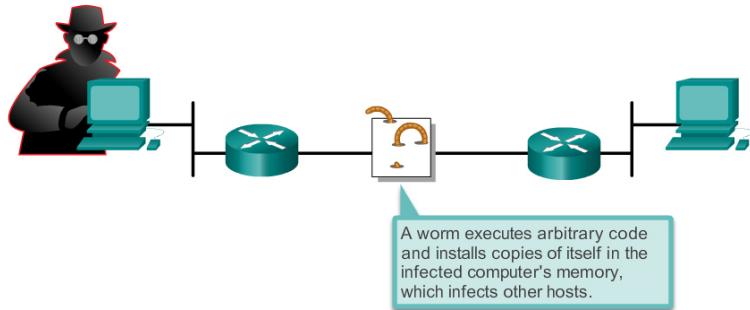


Fig. 11.18 Atac cu viermi

În plus față de atacurile cu cod rău intenționat, este posibil ca rețelele să cadă pradă mai multor atacuri de rețea. Atacurile de rețea pot fi clasificate în trei principale categorii :

1. **Atacuri de recunoaștere** – *descoperirea și mapări neautorizate ale sistemelor, serviciilor sau vulnerabilităților.*
2. **Atacuri de acces** – *manipularea neautorizată a datelor, accesului la sistem sau privilegiilor de utilizator.*
3. **Denial of service** – *dezactivarea sau coruperea rețelei, sistemelor sau serviciilor.*

11.3.2 Atacuri de recunoaștere

Atacatorii externi pot folosi instrumente Internet, cum ar fi utilizarea ”nslookup” și ”whois” pentru a determina ușor spațiul de adrese IP atribuit unei întreprinderi sau entități. După ce este determinat spațiul de adrese IP, un atacator poate da **ping** adreselor IP disponibile pentru a identifica adresele active. Pentru automatizarea acestui pas, un atacator ar putea folosi un

instrument de tip ”**ping sweep**”, cum ar fi ”**fping**” sau ”**gping**”, ce dă **ping** sistematic tuturor adreselor de rețea dintr-un range dat sau subrețea. Acest lucru este similar cu urmărirea unei secțiuni din cartea de telefon și apelarea fiecărui număr pentru a vedea cine răspunde.

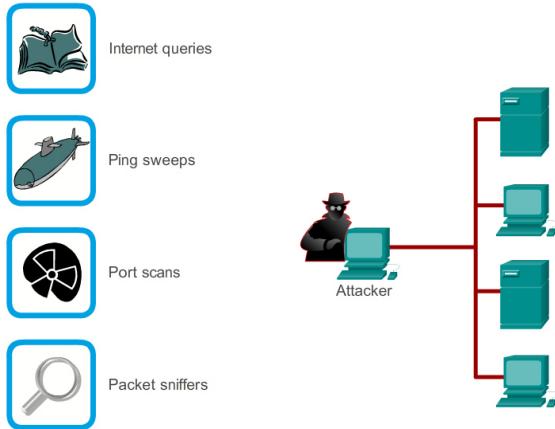


Fig. 11.19 Atacuri de recunoaștere

11.3.3 Atacuri de acces

Atacurile de acces explorează vulnerabilități cunoscute în serviciile de autentificare, servicii FTP și servicii web pentru a câștiga acces la conturi web, baze de date confidențiale și alte informații sensibile. Atacurile de acces pot fi clasificate în patru tipuri :

A. Unul dintre cele mai cunoscute tipuri de atacuri este atacul de parolă. Atacurile de parolă pot fi implementate folosind un packet sniffer pentru a obține conturile de utilizator și parolele transmise în text clar. Atacurile de parolă pot fi încercări repetitive de logare la o resursă comună, cum ar fi un server sau router, pentru a identifica un cont de utilizator, o parolă sau ambele. Aceste încercări repetitive sunt numite atacuri de tip dicționar sau atacuri brute-force.

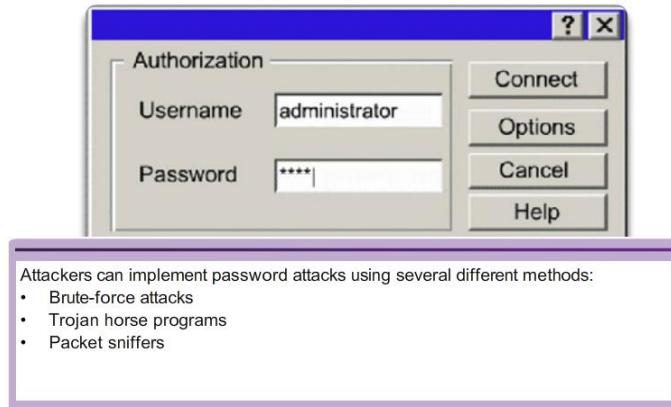
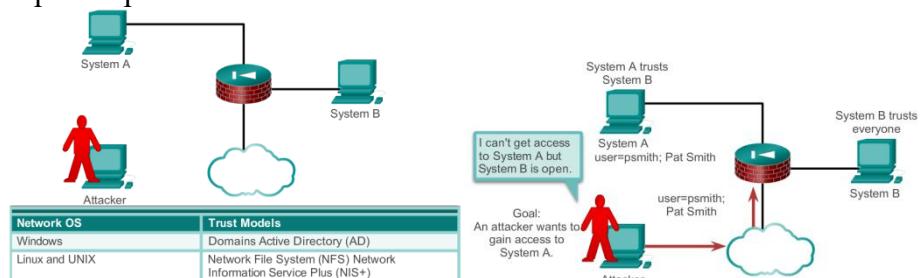


Fig. 11.20 Password attack

B. Atac prin exploatarea încrederii



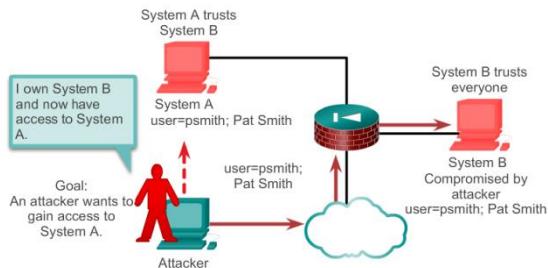


Fig. 11. 21 Trust exploitation

C. Atac prin redirectarea porturilor

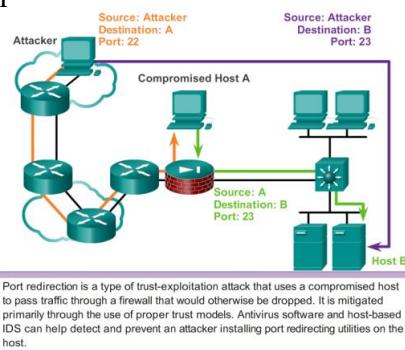


Fig. 11. 22 Port redirection

D. Atac prin om la mijloc

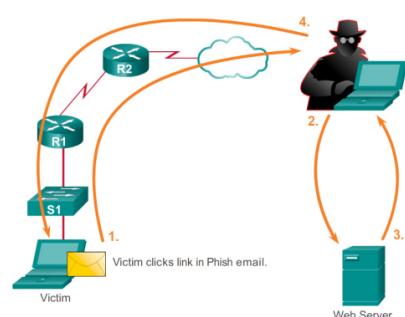
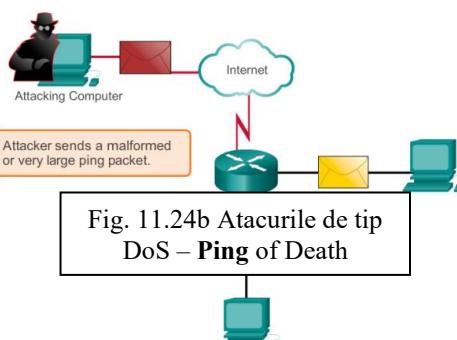
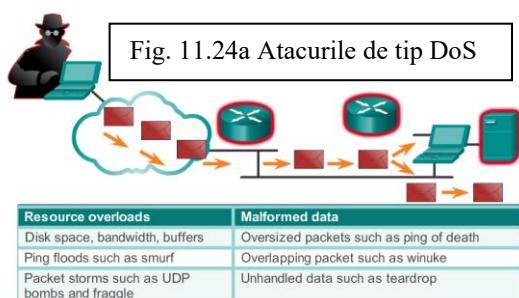


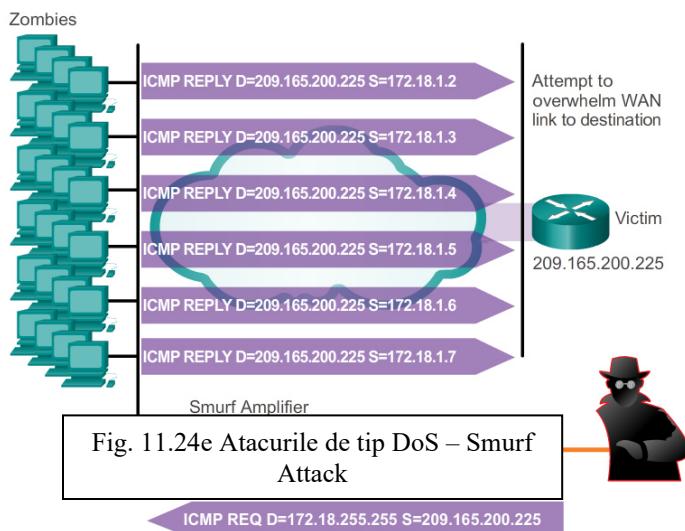
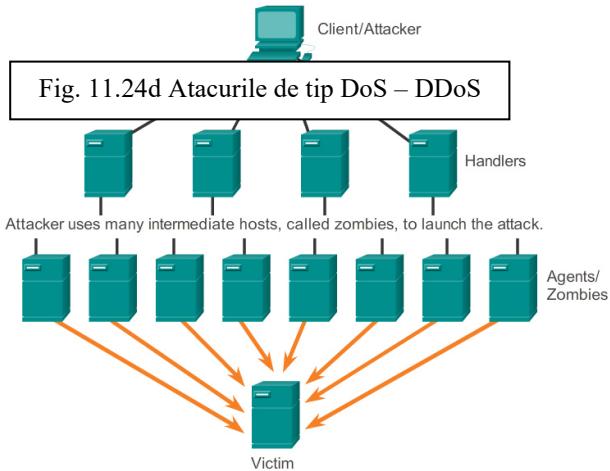
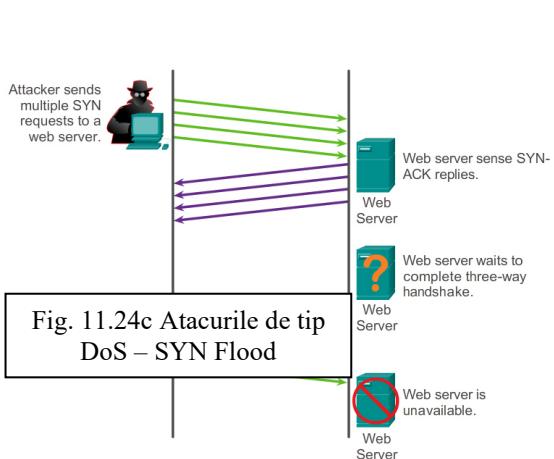
Fig. 11. 23 Man în – the – middle

11.3.4 Negarea serviciilor

Atacurile DoS sunt cele mai mediatizate forme de atac și se află printre cele mai dificile de eliminat. Chiar și într-o comunitate de atacator, atacurile DoS sunt considerate banale și de formă rea deoarece necesită prea puțin efort de executare. Însă datorită ușurinței lor de implementare și a prejudiciului potențial adus, atacurile DoS merită o atenție specială din partea administratorilor de rețea.

Atacurile DoS iau mai multe forme. Ele blochează oamenii autorizați la accesul la un serviciu prin consumarea resurselor sistemului.





11.3.5 Combaterea atacurilor de rețea

Software antivirus poate detecta mulți viruși și aplicații de tip cai Troieni și poate preveni împrăștierea acestora în rețea. Softwareul antivirus poate fi instalat la nivel de utilizator și la nivel de rețea.

Menținerea la curent cu cele mai recente dezvoltări în aceste tipuri de atacuri poate de asemenea conduce la o protecție eficientă împotriva acestor atacuri. O dată cu apariția unui nou virus sau aplicații Trojan, întreprinderile trebuie să-și actualizeze ultimile versiuni ale softwareului antivirus.

Combaterea atacului de tip vierme necesită multă silință din partea personalului de administrare a rețelei și sistemului. Următorii pași sunt recomandați pentru combaterea atacului de tip vierme:

- *Izolarea* – Stoparea răspândirii viermelui în rețea. Compartimentăm părțile neinfecțate ale rețelei.
- *Inocularea* – Începem să “peticim” toate sistemele și, dacă este posibil, scanăm sistemele de vulnerabilități.
- *Carantina* – Detectăm fiecare mașină infectată în rețea. Deconectăm, înlăturăm sau blocăm mașinile infectate din rețea.

- *Tratament* – Curățăm și “peticim” fiecare sistem infectat. Unii viermi necesită reinstalare completă de sistem pentru a curăța sistemul.

Cel mai eficient mod de combatere a atacului de tip vierme este prin descărcarea actualizărilor de securitate de la furnizorul de sistem de operare și peticirea tuturor sistemelor vulnerabile. Acest lucru este dificil de realizat cu sisteme necontrolate în rețea locală. Administrarea a numeroase sisteme implică crearea unei imagini software standard (sistemul de operare și aplicații acreditate ce sunt autorizate pentru folosirea pe sistemele client) ce este adăugată pe sisteme noi sau actualizate. Însă, cerințele de securitate se schimbă și sistemele deja adăugate ar putea necesita instalarea ”patches” de securitate actualizate.

O soluție de management a ”patches” de securitate critice este crearea unui server central de ”patch” cu care toate sistemele trebuie să comunice după o anumită perioadă de timp, aşa cum este prezentată în figura 11.25. Orice ”patches” ce nu sunt aplicate pe un host sunt descărcate automat de pe serverul de patch și instalate fără alte intervenții.

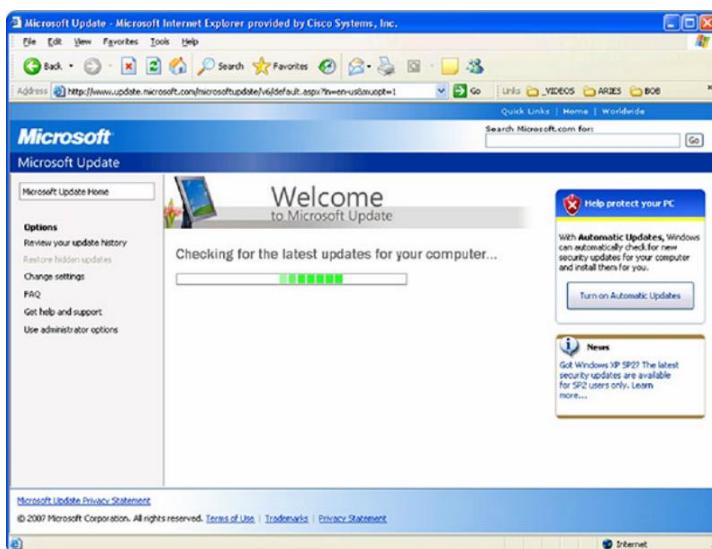


Fig. 11.25 Actualizările SO

Serviciile de securitate de rețea de tip triplu **A - Authentication, Authorization, and Accounting=AAA**) oferă cadrul de muncă inițial pentru setarea controlului accesului la un dispozitiv de rețea. AAA este un mod de control a persoanelor ce au permis accesul la o rețea (autentificare), ce pot face în timp ce se află în rețea (autorizare) și să vadă acțiunile efectuate în acest timp (contabilizare). AAA oferă un grad mai mare de scalabilitate decât comenzi de autentificare de pe consolă, AUX, VTY și din modul privilegiat.

11.3.6 Autentificarea

Utilizatorii și administratorii trebuie să dovedească că sunt cine spun. Autentificarea poate fi stabilită prin utilizarea unei combinații nume de utilizator-parolă, întrebări de provocare și răspuns, token cards și alte metode. De exemplu: “Sunt utilizatorul ‘student’. Știu parola pentru a dovedi că sunt utilizatorul ‘student’.”

Într-o rețea mică, autentificarea locală este adesea utilizată. Cu autentificarea locală, fiecare dispozitiv își menține propria bază de date de combinații nume de utilizator/parolă. Însă, când sunt mai multe conturi de utilizator într-o bază de date locală de pe dispozitiv, gestionarea respectivelor conturi de utilizator devine complexă. În plus, o dată cu creșterea rețelei și cu numărul de dispozitive introduse în rețea, autentificarea locală devine dificil de menținut și nu

este scalabilă. De exemplu, dacă există 100 de dispozitive de rețea, toate conturile de utilizator trebuie să fie adăugate pe toate cele 100 de dispozitive.

Pentru rețelele mari, o soluție mai scalabilă este autentificarea externă. Autentificarea externă permite tuturor utilizatorilor să fie autentificați printr-un server extern de rețea. Cele mai populare două opțiuni pentru autentificarea externă a utilizatorilor sunt RADIUS și TACACS+:

- **RADIUS** este un standard deschis cu utilizare scăzută a resurselor CPU și a memoriei. Este folosit de un range de dispozitive de rețea, cum ar fi switchuri, routere și dispozitive wireless.
- **TACACS+** este un mecanism de securitate ce permite autentificarea modulară, autorizarea și contabilitatea serviciilor, care folosește un TACACS+ daemon ce rulează pe un server de securitate.

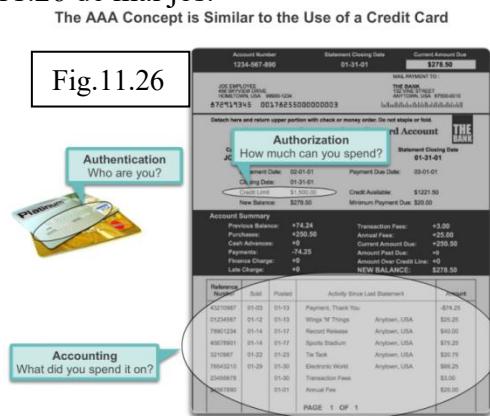
11.3.7 Autorizarea

După ce utilizatorul este autentificat, serviciile de autorizare determină ce resurse poate utilizatorul să le acceseze și ce operații îi sunt permise. Un exemplu este: "Utilizatorul 'student' poate accesa serverXYZ folosind numai Telnet."

11.3.8 Contorizarea

Contorizarea ține evidența înregistrărilor efectuate de către utilizator, inclusiv ce a accesat, cantitatea de timp în care resursa a fost accesată și orice schimbări efectuate. Contabilitatea ține evidența a modului în care sunt utilizate resursele de rețea. Un exemplu este "Utilizatorul 'student' a accesat serverXYZ folosind Telnet pentru 15 minute."

Conceptul AAA este similar utilizării unui card de credit. Cardul de credit identifică cine îl poate folosi, cât de mult poate cheltui și ține evidența a ce lucruri a comparat utilizatorul, așa cum se poate observa în Fig.11.26 de mai jos.



Pentru a proteja computerele individuale și serverele atașate la rețea, este important să controlăm traficul ce călătorește din și prin rețea.

Un firewall este unul dintre cele mai eficiente instrumente de securitate disponibile pentru protejarea utilizatorilor de rețea interni de amenințările externe. Un firewall este între două sau mai multe rețele și controlează traficul dintre ele și ajută de asemenea la prevenirea accesului neautorizat. Produsele firewall folosesc tehnici variate de determinare a accesului permis și interzis dintr-o rețea.

Acstea tehnici sunt:

- **Filtrarea de pachete** – Previne sau permite accesul în funcție de adresele IP sau MAC.
- **Filtrarea de aplicații** – Previne sau permite accesul unor anumite tipuri de aplicație, în funcție de numerele de port.
- **Filtrarea URL** – Previne sau permite accesul la websiteuri în funcție de anumite URLuri sau cuvinte cheie.
- **Stateful packet inspection (SPI)** – Pachetele ce vin trebuie să fie răspunsuri legitime la cereri de la hosturile interne. Pachetele nesolicită sunt blocate, cu excepția cazului în care sunt permise prin specificare. ISP poate de asemenea include capacitatea de recunoaștere și filtrare a anumitor tipuri de atacuri cum ar fi DoS.

Produsele firewall ar putea suporta una sau mai multe dintre aceste capacitați de filtrare. În plus, firewalurile efectuează adesea Network Address Translation (NAT). NAT traduce o adresă IP internă sau un grup de adrese într-o adresă IP publică, externă ce este trimisă în rețea. Acest lucru permite ca adresele IP interne să fie ascunse de utilizatorii externi.

Produsele firewall vin în diferite forme, aşa cum se poate vedea și în Fig.11.27 :

- **Appliance-based firewalls** – Un firewall bazat pe dispozitiv este un firewall ce este construit într-un dispozitiv dedicat hardware numit și dispozitiv de securitate.
- **Server-based firewalls** – Un firewall bazat pe server constă dintr-o aplicație firewall ce rulează pe network operating system (NOS) cum ar fi UNIX sau Windows.
- **Integrated firewalls** – Un firewall integrat este implementat prin adăugarea funcționalității firewall unui dispozitiv existent, cum ar fi un router.
- **Personal firewalls** - Firewalurile personale se află pe computerele hosturilor și nu sunt dezvoltate pentru implementări LAN. Ele pot fi disponibile implicit de la OS sau pot fi de la un furnizor extern.



O rețea securizată este la fel de puternică precum legătura cea mai slabă a sa. Amenințările de profil înalt discutate cel mai mult în media sunt amenințările externe, cum ar fi viermii sau atacurile DoS. Însă securizarea rețelei interne este la fel de importantă ca securizarea perimetrului unei rețele. Rețeaua internă este alcătuită din puncte de lucru finale, unele fiind arătate în Fig. . Un endpoint, sau host, este un sistem individual computer sau dispozitiv ce se comportă ca un client de rețea. Endpointurile comune sunt laptopurile, desktopurile, serverele, smart phoneurile și tabletele. Dacă utilizatorii nu își instalează securitatea pe dispozitivele lor, nici-o cantitate de precauții de securitate nu va garanta o rețea sigură.

Securizarea dispozitivelor finale este una dintre cele mai provocatoare sarcini ale unui administrator de rețea deoarece implică natura umană. O companie ar putea avea politici bine documentate puse la punct și angajații trebuie să fie conștienți de acele reguli. Angajații trebuie să fie instruiți de utilizarea adecvată a rețelei. Politicile includ adesea utilizarea de software

antivirus și prevenirea intruziunilor pe host. Mai multe soluții complexe de securizare a endpointului se bazează pe controlul accesului la rețea.

Securitatea endpoint necesită de asemenea securizarea dispozitivelor de nivel 2 din infrastructura de rețea pentru a preveni atacurile de nivel 2 cum ar fi MAC address spoofing, atacurile de overflow a tabelei de adrese MAC și atacurile LAN storm. Acest lucru este cunoscut ca fiind combatere a atacului.



11.4 Securizarea dispozitivelor

O parte a securității de rețea este securizarea dispozitivelor, inclusiv dispozitivele finale și intermediare, cum ar fi dispozitivele de rețea.

Atunci când un nou sistem de operare este instalat pe dispozitiv, setările de securitate sunt la valorile implicate. În multe cazuri, nivelul de securitate este inadecvat. Pentru routerele Cisco, caracteristica Cisco AutoSecure poate fi folosită pentru a ajuta la securizarea sistemului, aşa cum este descrisă în Fig. . Există pași simpli ce ar trebui să fie urmați ce se aplică celor mai multe sisteme de operare:

- *Numele de utilizator și parolele default ar trebui schimbată imediat.*
- *Accesul la resursele sistemului ar trebui să fie restricționat numai la indivizi ce sunt autorizați să utilizeze respectivele resurse.*
- *Orice servicii și aplicații inutile ar trebui închise sau dezinstalate, atunci când este posibil.*

Toate dispozitivele ar trebui să fie actualizate cu patches de securitate imediat ce sunt disponibile. Adesea, dispozitivele de la producător au stat într-un depozit pentru o anumită perioadă de timp și nu au instalate cele mai noi patches. Este important, înainte de punerea în funcțiune, să actualizăm orice software și să instalăm orice patches de securitate.

Locking Down Your Router

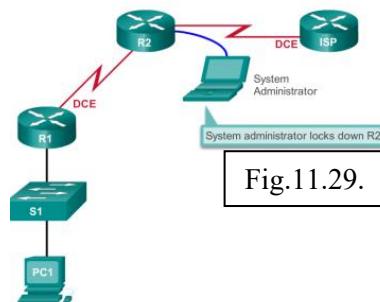


Fig.11.29.

Pentru a proteja dispozitivele de rețea, este important să utilizăm parole puternice. Există mai multe linii de ghidare standard de urmat:

- *Folosim o parolă cu lungime de cel puțin 8 caractere, de preferat 10 sau mai multe caractere. Cu cât este mai lungă parola, cu atât este mai bună.*
- *Facem parolele complexe. Includem o combinație de litere mari cu litere mici, numere, simboluri și spații, dacă este permis.*

- Evită parolele bazate pe repetare, cuvinte comune din dicționar, litere sau numere în segvență, nume de utilizator, nume ale rудelor sau ale animalului, informații biografice, cum ar fi date de naștere, numere ID, numele strămoșilor sau alte piese identificabile ușor.
 - Scriem un cuvânt incorrect în mod deliberat. De exemplu, Smith = Smyth = 5mYth sau Security = SecurIty.
 - Schimbăm parolele des. Dacă o parolă este compromisă în necunoștință, fereastra oportunității pentru atacator de utilizarea a parolei este limitată.
 - Nu notăm parolele și le lăsăm în locuri evidente, cum ar fi pe birou sau monitor.
- Fig. arată exemple de parole puternice și slabe.

Pe routerele Cisco, spațiile de început sunt ignorate pentru parole, însă spațiile după primul caracter nu. Prin urmare, o metodă de creare a unei parole puternice este utilizarea spațiilor în parolă și crearea unei fraze alcătuită din mai multe cuvinte. Aceasta se numește pass phrase. O pass phrase este adesea mai ușor de ținut minte decât o parolă simplă. Este de asemenea mai lungă și mai greu de ghicit.

Administratorii ar trebui să se asigure de faptul că parolele puternice sunt folosite în rețea. Un mod de realizarea al acestui lucru este utilizarea instrumentelor de atac “brute force” folosite de atacatori pentru a verifica puterea parolei.

Weak and Strong Passwords

Fig.11.30. PASS

Weak Password	Why it is weak
secret	Simple dictionary password
smith	Mother's maiden name
toyota	Make of a car
bob1967	Name and birthday of a user
Blueleaf23	Simple words and numbers

Strong Password	Why it is strong
b67n42d39c	Combines alphanumeric characters
12^h u4@1p7	Combines alphanumeric characters, symbols, and also includes a space

La implementarea dispozitivelor este important să urmăm toate îndrumările de securitate setate de către organizație. Acestea includ numirea dispozitivelor într-o manieră ce permite documentarea ușoară și urmărirea, dar și care menține o anumită formă de securitate. Nu este înțelept să oferim prea multe informații cu privire la utilizarea dispozitivului în numele de host. Există multe alte măsuri de bază de securitate ce ar trebui să fie urmate.

11.4.1 Additional Password Security

Parolele puternice sunt utile atunci când sunt secrete. Există mai mulți pași ce pot fi urmați pentru a ajuta ca parolele să rămână secrete. Folosirea comenzii din config.rea globală **service password-encryption** previne indivizii neautorizați de a vedea parolele în text clar în fișierul de config.re, aşa cum se arată în Fig. . Această comandă are ca efect criptarea tuturor parolelor necriptate.

În plus, pentru a ne asigura că toate parolele au minimul de lungime specificată, folosiți comanda **security passwords min-length** în modul de config.re global.

Un alt mod prin care hackerii pot învăța parolele se face prin simplele atacuri brute-force, încercând mai multe parole până când una se potrivește. Este posibil să prevenim acest tip de atac prin blocarea încercărilor de logare la un dispozitiv dacă un număr setat de eșecuri au loc într-o anumită perioadă de timp.

```
Router(config)# login block-for 120 attempts 3 within 60
```

Această comandă va bloca încercările de logare pentru 120 de secunde, dacă au existat trei încercări cu eşec în 60 de secunde.

Banners – Un mesaj banner este similar cu un semn de încălcare a legii. Sunt importante pentru că ar putea avea urmări în justiție, într-o instanță de drept, pe oricine accesează sistemul inadecvat. Ne asigurăm că mesajele de banner respectă politicile de securitate ale organizației.

```
Router(config)# banner motd #Vineri la ora 14:00 serverul va intra în revizie !#
```

Exec Timeout – O altă recomandare este setarea de executive timeauturi. Prin setarea exec timeout, spunem dispozitivului să deconecteze automat utilizatorii de pe o linie după ce au fost inactivi pentru o anumită perioadă specificată prin valoarea exec timeout. Exec timeauturile pot fi configurate pe liniile console și vty, precum și porturi auxiliare.

```
Router(config)# line vty 0 15
```

```
Router(config-vty)# exec-timeout 10 55
```

Această comandă va deconecta utilizatorii după 10 minute și 55 de secunde.

```
Router(config) #service password-encryption
Router(config) #security password min-length 8
Router(config) #login block-for 120 attempts 3 within 60
Router(config) #line vty 0 4
Router(config-vty) #exec-timeout 10
Router(config-vty) #end
Router#show running-config
-more-
!
line vty 0 4
password 7 03095A0F034F38435B49150A1819
exec-timeout 10
login
```

11. 5 Acces de la distanță prin SSH

Protocolul de gestionare a dispozitivelor de la distanță este Telnet. Telnet nu este securizat. Datele conținute într-un pachet Telnet sunt transmise necriptate. Folosind un instrument ca Wireshark, este posibil ca cineva să “intercepteze” o sesiune Telnet și să obțină o informație de parolă. Din acest motiv, este recomandat să activăm SSH pe dispozitive pentru accesul securizat de la distanță. Este posibilă configurația unui dispozitiv să suporte SSH în patru pași, așa cum se poate vedea și în Fig. .

Pasul 1. Ne asigurăm că routerul are un nume de host unic și apoi configurăm numele IP de domeniu al rețelei folosind comanda **ip domain-name Seria25.com** în modul de configurație globală.

Pasul 2. Chei secrete one-way trebuie să fie generate de către router pentru a cripta traficul SSH. Cheia este folosită pentru criptarea și decriptarea datelor. Pentru a crea o cheie criptată, folosim comanda **crypto key generate rsa general-keys modulus-modulus-size=1024** în modul de configurație global. Înțelesul specific al diferențelor părți ale acestei comenzi este complex și nu prezintă scopul acestui curs, însă acum doar notăm faptul că modulus determină dimensiunea cheii și poate fi configurația de la 360 biți la 2048 de biți. Cu cât este mai mare modulus, cu atâtă

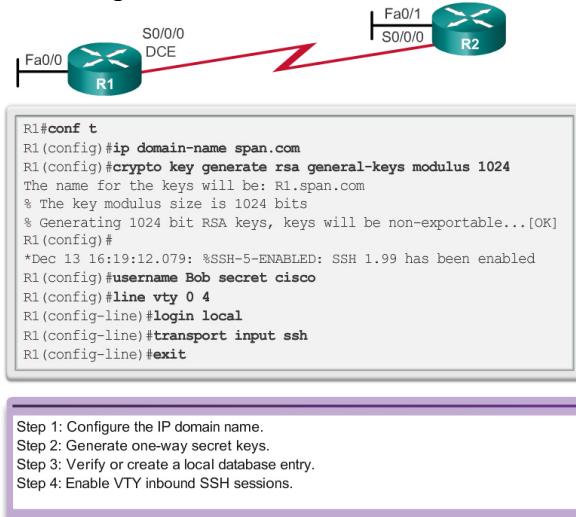
cheia este mai sigură, însă durează mai mult criptarea și decriptarea informațiilor. Lungimea minimă recomandată a modulus este de 1024 de biți.

Router(config)# crypto key generate rsa (general-keys modulus 1024)

Pasul 3. Creem o bază de date locală cu intrare de nume de utilizator folosind comanda **username name secret secret** din modul de conFig.re global.

Pasul 4. Activăm sesiuni SSH pe vty folosind comenziile pe linia vty **login local** și **transport input ssh**.

Serviciul SSH de pe router poate fi acum accesat folosind un software client SSH.



11.6 Performanța de bază a rețelei

Ping – După ce rețeaua a fost pusă în aplicare, un administrator de rețea trebuie să fie capabil să testeze conectivitatea la rețea pentru a se asigura de faptul că funcționează corect. În plus, este o bună idee ca administratorul de rețea să documenteze rețeaua.

Comanda ping – Comanda **ping** este un mod eficient de testare a conectivității. Testarea ne este referită ca testare a stivei de protocoale deoarece comanda **ping** merge de la nivelul 3 al modelului OSI la nivelul 2, apoi la nivelul 1. **Ping** folosește protocolul ICMP pentru a verifica conectivitatea.

Comanda **ping** nu găsește întotdeauna cu precizie natura unei probleme, însă ajută la identificarea sursei problemei, un pas important în depanarea unei probleme de rețea.

Comanda **ping** oferă o metodă de verificare a stivei de protocoale și a conFig.ției adresei IPv4 de pe host, dar și testează conectivitatea cu hosturile locale sau de la distanță, aşa cum se poate observa în Fig. . Există instrumente suplimentare ce pot oferi mai multe informații decât **ping**, cum ar fi Telnet sau Trace route, ce vor fi discutate detaliat mai târziu.

Indicatorii Ping din IOS – Un **ping** efectuat de pe IOS va avea unul dintre indicatorii pentru fiecare ICMP echo trimis. Indicatorii cei mai cunoscuți sunt:

- **!** – indică primirea unui mesaj de răspuns ICMP echo.
- **.** – indică faptul că timpul de așteptare a expirat pentru un mesaj de răspuns ICMP echo
- **U** – un mesaj de ICMP unreachable a fost primit.

"!" (semnul de exclamare) indică faptul că **ping** a fost completat cu succes și a verificat conectivitatea de nivel 3.

".." poate indica probleme în comunicare. Ar putea indica faptul că o problemă de conectivitate are loc undeva în cale. Ar putea de asemenea indica faptul că un router din cale nu are o rută spre destinație și nu a trimis un mesaj ICMP de destinație unreachable. De asemenea ar putea indica faptul că **ping** a fost blocat de către securitatea de dispozitiv.

"U" indică faptul că un router în cale nu are o rută spre adresa destinație sau faptul că cererea **ping** a fost blocată și a avut ca răspuns un mesaj ICMP unreachable.

Testarea loopback – Comanda **ping** este folosită pentru a verifica configurația IP internă de pe hostul local. Reamintim faptul că acest test este realizat prin utilizarea comenzi **ping** asupra unei adrese rezervate numită loopback (127.0.0.1). Aceasta verifică funcționarea adecvată a stivei de protocol de la nivelul rețea până la nivelul fizic și înapoi, fără punerea unui semnal real pe mediu de comunicare.

Comenzile **ping** sunt introduse pe linia de comandă.

Introducem comanda **ping loopback** cu sintaxa următoare:

C:\>ping 127.0.0.1

Răspunsul pentru o astfel de comandă va prezenta ceva de forma:

Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Statisticile pentru **Ping** la adresa 127.0.0.1:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms

Rezultatul indică faptul că patru pachete de 32 de byte au fost trimise și returnate de host 127.0.0.1 într-un timp de mai puțin de 1 ms. TTL (Time-to-Live) definește numărul de hopuri pe care pachetul **ping** le-a străbătut până a fost aruncat.

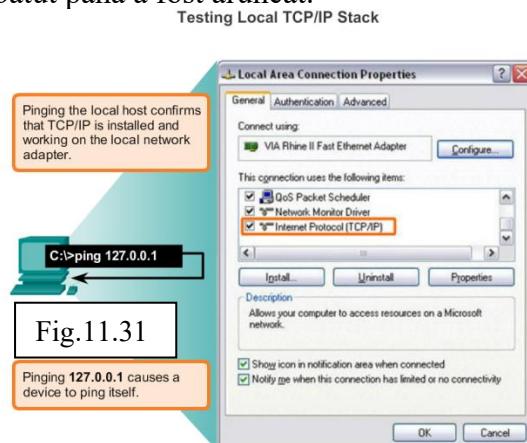


Fig.11.31

IOS de pe echipamentele intermediare oferă un mod "extins" al comenzi **ping**. Acest mod este realizat prin introducerea comenzi **ping** în modul EXEC privilegiat, fără o adresa IP destinație. O serie de prompturi sunt prezentate, așa cum se poate vedea și în exemplul de mai jos. Prin apăsarea tastei Enter acceptăm valorile indicațiile implicate. Exemplul de mai jos ilustrează modul în care putem forța adresa sură pentru un **ping** să fie la 10.1.1.1 (de văzut R2 din Fig.); adresa sursă pentru un **ping** standard va fi 209.165.200.226. Prin efectuarea acestui lucru, administratorul de rețea poate verifica de la distanță (de pe R2) faptul că R1 are ruta 10.1.1.0/24 în tabela sa de rutare.

R2# **ping**

Protocol [ip]:

Target IP address: **192.168.10.1**

Repeat count [5]:

Datagram size [100]:

Timeout în seconds [2]:

Extended commands [n]: **y**

Source address or interface: **10.1.1.1**

Type of service [0]:

Set DF bit în IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 36/97/132 ms

Introducerea unei perioade mai mari de timeout decât cea implicită permite ca problemele eventuale de latență să fie detectate. Dacă testul **ping** este cu succes pentru o valoare mai mare, o conexiune există între hosturi, însă latența poate fi o problema în rețea.

De reținut faptul că introducerea "y" pe promptul "Extended commands" oferă mai multe opțiuni utile pentru depanare.

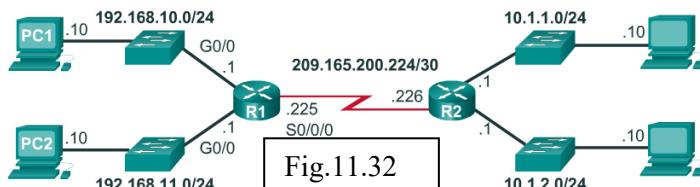


Fig.11.32

Unul dintre cele mai eficiente instrumente de monitorizare și depanare a performanței rețelei este stabilirea unor linii de bază (baseline) de rețea. O "baseline" este un proces de studiere a rețelei la intervale regulate pentru a asigura faptul că rețeaua funcționează corect. O baseline de rețea este mai multe decât un simplu raport ce detaliază "sănătatea" rețelei la un anumit punct. Crearea unei baseline eficiente de performanță a rețelei este realizată în timp. Măsurarea performanței la momente de timp diferite și încărcarea vor ajuta la crearea unei imagini mai bune asupra performanței generale de rețea.

Ieșirea comenzielor de rețea poate contribui cu date pentru baseline de rețea.

O metodă de începere a unei baseline este copierea rezultatelor comenziilor **ping**, **tracert** executate sau a altor comenzi relevante într-un fișier text. Aceste fișiere text pot fi "ștampilate" cu data și salvate într-o arhivă pentru revederea ulterioară a acestora.

O utilizare eficientă a informațiilor stocate este compararea rezultatelor în timp (Fig. 3). Printre elementele de luat în considerare sunt mesajele de eroare și timpii de răspuns de la host la host. Dacă există o creștere considerabilă în timpii de răpsuns, poate exista o problemă de latență la adresa.

Importanța creării documentației nu poate fi accentuată destul. Verificarea conectivității host-la-host, problemele de latență și soluții ale problemelor identificate pot ajuta un administrator de rețea în păstrarea funcționării rețelei cât mai eficient posibilă.

Rețelele corporative ar trebui să aibă baselines extinse; mai extinse decât putem explica în acest curs. Instrumente software de grad profesional sunt disponibile pentru stocarea și menținerea informațiilor baseline. În acest curs, acoperim numai unele tehnici de bază și discutăm scopul acestor baselines.

Capturarea ieșirii comenzi **ping** poate fi realizată din promptul IOS, aşa cum se poate vedea în Fig. 11.33.

Run the same test

FEB 8, 2013 08:14:43

```
C:\>ping 10.66.254.159
Pinging 10.66.254.159 with 32 bytes of data:
Reply from 10.66.234.159: bytes=32 time<1ms TTL=128
Ping statistics for 10.66.254.159:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

MAR 17, 2013 14:41:06

```
C:\>ping 10.66.254.159
Pinging 10.66.254.159 with 32 bytes of data:
Reply from 10.66.234.159: bytes=32 time<6ms TTL=128
Ping statistics for 10.66.254.159:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 6ms, Maximum = 6ms, Average = 6ms
```

At different times

FEB 8, 2013 08:14:43

```
C:\>ping 10.66.254.159
Pinging 10.66.254.159 with 32 bytes of data:
Reply from 10.66.234.159: bytes=32 time<1ms TTL=128
Ping statistics for 10.66.254.159:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

MAR 17, 2013 14:41:06

```
C:\>ping 10.66.254.159
Pinging 10.66.254.159 with 32 bytes of data:
Reply from 10.66.234.159: bytes=32 time<6ms TTL=128
Ping statistics for 10.66.254.159:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Compare values

FEB 8, 2013 08:14:43

```
C:\>ping 10.66.254.159
Pinging 10.66.254.159 with 32 bytes of data:
Reply from 10.66.234.159: bytes=32 time<1ms TTL=128
Ping statistics for 10.66.254.159:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

MAR 17, 2013 14:41:06

```
C:\>ping 10.66.254.159
Pinging 10.66.254.159 with 32 bytes of data:
Reply from 10.66.234.159: bytes=32 time<6ms TTL=128
Ping statistics for 10.66.254.159:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Router Ping Capture - Saving to a text file

Fig.11.33

dsh - HyperTerminal

File Edit View Call Transfer Help

Send File... Receive File... Capture Text... Stop

Send Text File... Pause

Capture to Printer Resume

Interface Serial1 description Serial1 Interface on the RTA router ip address 192.168.4.89.255.255.240

In the terminal session:

1. Start the text capture process.
2. Issue a **ping <ip address>** command.
3. Stop the capture process.
4. Save the text file.

Tracert – Trace are ca răspuns o listă de hopuri prin care un pachet este routat în rețea. Forma comenzi depinde de locul unde este efectuată. Atunci când efectuăm comanda trace dintr-un computer Windows, folosind tracert. Atunci când efectuăm comanda trace de pe CLI a unui router, folosim traceroute, aşa cum se poate observa și în Fig. 1.

Ca și comenzi **ping**, comenzi **trace** sunt introduse în linia de comandă și necesită o adresa IP ca argument.

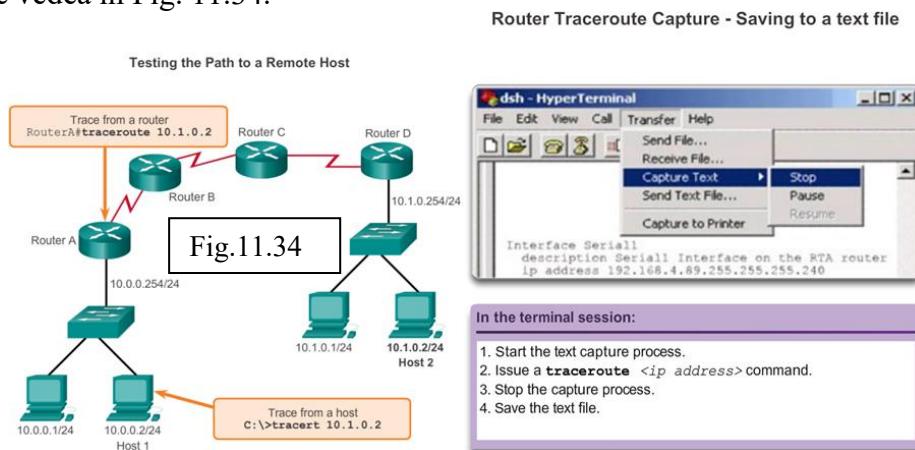
Presupunând că această comandă va fi efectuată de pe un computer Windows, folosim forma **tracert**:

C:>**tracert 10.1.0.2**

```
Tracing route to 10.1.0.2 over a maximum of 30 hops
1 2 ms 2 ms 2 ms 10.0.0.254
2 * * * Request timed out.
3 * * * Request timed out.
4 ^C
```

Singurul răspuns cu succes a fost de la gateway de pe Router A. Cererile Trace la următorul hop au expirat, însemnând că routerul next hop nu a răspuns. Rezultatele trace indică faptul că eșecul este undeva în internetwork, în exteriorul LANului.

Capturarea ieșirii traceroute poate fi de asemenea efectuată de pe promptul routerului, aşa cum se poate vedea în Fig. 11.34.



11.7 Comenzi show

Comenzi IOS CLI **show** afișează informații relevante cu privire la configurația și funcționarea dispozitivului.

Tehnicienii de rețea utilizează comenzi **show** intens pentru vizualizarea fișierelor de config, verificarea statusului interfețelor și proceselor de pe dispozitiv și verificarea statusului operațional al dispozitivului. Comenzi **show** sunt disponibile fie că dispozitivul a fost config.t cu CLI, fie cu Cisco Configuration Professional.

Statusul a aproape fiecărui proces sau funcție a routerului poate fi afișat cu ajutorul comenzi **show**. Unele dintre cele mai populare comenzi **show** sunt:

- **show running-config** (Fig. 11.35).
- **show interfaces** (Fig. 11.36).
- **show arp** (Fig. 11.37).
- **show ip route** (Fig. 11.38).
- **show protocols** (Fig. 11.39).
- **show version** (Fig. 11.40).

Show running-config

```
R1#show running-config
<Output omitted>
Building configuration...
Current configuration : 1063 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname R1
enable secret 5 $1$6w9$vdvpVM6zV10E6tSyLdkR5/
no ip domain lookup
!
interface FastEthernet0/0
description LAN 192.168.1.0 default gateway
ip address 192.168.1.1 255.255.255.0
duplex auto
speed auto
!
interface FastEthernet0/1
no ip address
shutdown
duplex auto
speed auto
!
interface Serial0/0/0
description WAN link to R2
ip address 192.168.2.1 255.255.255.0
encapsulation ppp
clock rate 64000
no fair-queue
!
interface Serial0/0/1
no ip address
shutdown
!
interface Vlan1
no ip address
!
router rip
version 2
network 192.168.1.0
network 192.168.2.0
!
banner motd ^CUnauthorized Access Prohibited^C
!
ip http server
!
line con 0
password cisco
```

Fig.11.35

```
!
interface Serial0/0/1
no ip address
shutdown
!
interface Vlan1
no ip address
!
router rip
version 2
network 192.168.1.0
network 192.168.2.0
!
banner motd ^CUnauthorized Access Prohibited^C
!
ip http server
!
line con 0
password cisco
login
line aux 0
line vty 0 4
password cisco
login
```

```
R1#show interfaces
<Output omitted>
FastEthernet0/0 is up, line protocol is up
Hardware is GT96K FE, address is 001b.5325.256e
(hia 001b.5325.256e)
Internet address is 192.168.1.1/24
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Full-duplex, 100Mb/s, 100BaseTX/FX
ARF type: ARPA, ARP Timeout 04:00:00
Last input 00:01:17, output 00:00:01, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes);
Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
196 packets input, 31850 bytes
Received 181 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog
```

Fig.11.36

```
0 input packets with dribble condition detected
392 packets output, 35239 bytes, 0 underruns
0 output errors, 0 collisions, 3 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out

FastEthernet0/1 is administratively down,
line protocol is down

Serial0/0/0 is up, line protocol is up
Hardware is GT96K Serial
Internet address is 192.168.2.1/24
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation PPP, LCP Listen, loopback not set
Keepalive set (10 sec)
Last input 00:00:02, output 00:00:03, output hang never
Last clearing of "show interface" counters 00:51:52
Input queue: 0/75/0/0 (size/max/drops/flushes);
Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
```

```
Hardware is GT96K Serial
Internet address is 192.168.2.1/24
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation PPP, LCP Listen, loopback not set
Keepalive set (10 sec)
Last input 00:00:02, output 00:00:03, output hang never
Last clearing of "show interface" counters 00:51:52
Input queue: 0/75/0/0 (size/max/drops/flushes);
Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
401 packets input, 27437 bytes, 0 no buffer
Received 293 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
389 packets output, 26940 bytes, 0 underruns
0 output errors, 0 collisions, 2 interface resets
0 output buffer failures, 0 output buffers swapped out
6 carrier transitions
DCD=up DSR=up DTR=up RTS=up CTS=up

Serial0/0/1 is administratively down, line protocol is down
```

Show arp

Fig.11.37

```
R1#show arp
Protocol Address Age (min) Hardware Addr Type Interface
Internet 172.17.0.1 - 001b.5325.256e ARPA
FastEthernet0/0
Internet 172.17.0.2 12 000b.db04.a5cd ARPA
FastEthernet0/0
```

Show ip route

Fig.11.38

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, Serial0/0/0
R    192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:24, Serial0/0/0
```

Show protocols	Fig.11.39	Show version	Fig.11.40
<pre>R1#show protocols Global values: Internet Protocol routing is enabled FastEthernet0/0 is up, line protocol is up Internet address is 192.168.1.1/24 FastEthernet0/1 is administratively down, line protocol is down FastEthernet0/1/0 is up, line protocol is down FastEthernet0/1/1 is up, line protocol is down FastEthernet0/1/2 is up, line protocol is down FastEthernet0/1/3 is up, line protocol is down Serial0/0/0 is up, line protocol is up Internet address is 192.168.2.1/24 Serial0/0/1 is administratively down, line protocol is down Vlan1 is up, line protocol is down</pre>		<pre>R1#show version <Output omitted> Cisco IOS Software, 1841 Software (C1841-ADVISERVICESK9-M), Version 12.4(10b), RELEASE SOFTWARE (fc3) Technical Support: http://www.cisco.com/techsupport Copyright (c) 1986-2007 by Cisco Systems, Inc. Compiled Fri 19-Jan-07 15:15 by prod_rel_team ROM: System Bootstrap, Version 12.4(13r)T, RELEASE SOFTWARE (fc1) R1 uptime is 43 minutes System returned to ROM by reload at 22:05:12 UTC Sat Jan 5 2008 System image file is "flash:c1841-adviserervicesk9-mz.124-10b.bin" Cisco 1841 (revision 6.0) with 174080K/22528K bytes of memory. Processor board ID FTX1111WQF 6 FastEthernet interfaces 2 Serial(sync/async) interfaces 1 Virtual Private Network (VPN) Module DRAM configuration is 64 bits wide with parity disabled. 191K bytes of NVRAM. 62720K bytes of ATA CompactFlash (Read/Write) Configuration register is 0x2102</pre>	

După ce fișierul de configurare startup este încărcat și routerul bootează cu succes, comanda **show version** poate fi utilizată pentru verificarea și depanarea unor componente software și hardware de bază folosite în procesul de bootup. Ieșirea comenzi **show version** include:

- *Versiunea Cisco IOS utilizată.*
- *Versiunea softwareului bootstrap de sistem, stocat în memoria ROM ce a fost inițial utilizat pentru a boota routerul.*
- *Numele de fișier complet al imaginii Cisco IOS și unde este localizat programul de bootstrap.*
- *Tipul de CPU de pe router și cantitatea de RAM. Ar putea fi necesară actualizarea cantității de RAM atunci când se actualizează softwareul IOS.*
- *Numărul și tipul de interfețe fizice de pe router.*
- *Cantitatea de NVRAM. NVRAM este utilizat pentru a stoca fișierul startup-config.*
- *Cantitatea de memorie flash a routerului. Ar putea fi necesară actualizarea cantității de flash atunci când se actualizează softwareul IOS.*
- *Valoarea curentă config.tă a registrului software de config.ție în hexazecimal.*

Registrul de config.ție spune routerului cum să booteze. De exemplu, setările implicite din fabrică pentru registrul de config.ție este 0x2102. Această valoare indică faptul că routerul încearcă să încarce o imagine IOS din flash și încarcă fișierul startup de config.re din NVRAM. Este posibilă schimbarea registrului de config.re și prin urmare, schimbarea locului în care routerul se uită pentru imaginea IOSului și fișierul startup de config.re în timpul procesului de bootup. Dacă există o a doua valoare în paranteze, precizează valoarea registrului de config.ție din următoarea reîncărcare a routerului.

Fig.11.41

```

Router#show version
Cisco Internetwork Operating System
Software
IOS(tm)2500 Software (C2500-I-L), Version
12.0(17a), RELEASE SOFTWARE (fc1)
Copyright (c)1986-2002 by cisco
Systems, Inc.
Compiled Mon 11-Feb-02 05:55 by kellythw
Image text-base:0x00001000
ROM:system Bootstrap, Version
11.0(10c), SOFTWARE
BOOTFLASH :3000 Bootstrap Software (IGS-
BOOT-R), Version 11.0(10c), RELEASE
SOFTWARE (fc1)
System image file is "flash:c2500-i-
1.120-17a.bin"
Cisco 2500 (68030 processor(revision N)
With 2048K/2048K bytes of memory.
processor bord ID 08860060,with hardware
revision 00000000
Bridging software.
X.25 software,version 3.0.0.
1 Ethernet/IEEE 802.3 interface(s)
2 Serial network interface(s)
32K bytes of non-volatile Configuration
memory.
8192K bytes of processor board system
flash (Read ONLY)
Configuration register is 0x2102
Router#

```

Comanda **show version** de pe un switch afișează informații cu privire la versiunea software încărcată curent, împreună cu informații hardware și de dispozitiv. Unele dintre informațiile afișate în această comandă sunt:

- **Software version** - *Versiunea IOS software.*
- **Bootstrap version** - *Versiunea Bootstrap.*
- **System up-time** – *Timpul scurs de la ultimul reboot.*
- **System restart info** – *Metoda de restart (e.g., ciclu energetic, crash).*
- **Software image name** – *Numele fișier IOS.*
- **Switch platform and processor type** – *Numărul modelului și tipul procesorului.*
- **Memory type (shared/main)** – *RAM principal al procesorului și shared packet I/O buffering.*
- **Hardware interfaces** – *Interfețele disponibile pe switch.*
- **Configuration register** – *Setează specificațiile de bootup, viteza de consolă și parametrii aferenți.*

Fig.11.42 arată un exemplu de ieșire tipică a comenzi **show version** afișată de către un switch.

<pre> Switch#show version Cisco IOS Software, C2960 Software (C2960-LANBASE-M), Version 12.2(25)SEE2, RELEASE SOFTWARE (fc1) Copyright (c) 1986-2006 by Cisco Systems, Inc. Compiled Fri 28-Jul-06 04:33 by yenanh Image text-base: 0x00003000, data-base: 0x00AA2F34 ROM: Bootstrap program is C2960 boot loader BOOTLDR: C2960 Boot Loader (C2960-HBOOT-M) Version 12.2(25r)SEE1, RELEASE SOFTWARE (fc1) Switch uptime is 2 minutes System returned to ROM by power-on System image file is "flash:c2960-lanbase-mz.122-25.SEE2/c2960- lanbase-mz.122-25.SEE2.bin" cisco WS-C2960-24TT-L (PowerPC405) processor (revision B0) with 61440K/4088K bytes of memory. Processor board ID FOC1107Z9ZN Last reset from power-on 1 Virtual Ethernet interface 24 FastEthernet interfaces 2 Gigabit Ethernet interfaces The password-recovery mechanism is enabled. </pre>	<pre> 64K bytes of flash-simulated non-volatile configuration memory. Base ethernet MAC Address : 00:1B:53:03:17:00 Motherboard assembly number : 73-10390-03 Power supply part number : 341-0097-02 Motherboard serial number : FOC11071TTJ Power supply serial number : AZS110605RU Model revision number : B0 Motherboard revision number : C0 Model number : WS-C2960-24TT-L System serial number : FOC1107Z9ZN Top Assembly Part Number : 800-27221-02 Top Assembly Revision Number : C0 Version ID : V02 CLEI Code Number : COM3L00BRA Hardware Board Revision Number : 0x01 Switch Ports Model SW Version SW Image ----- ----- ----- * 1 26 WS-C2960-24TT-L 12.2(25)SEE2 C2960-LANBASE-M Configuration register is 0xF </pre>
--	---

Fig.11.42

11.8 Comenzile pe Host și IOS

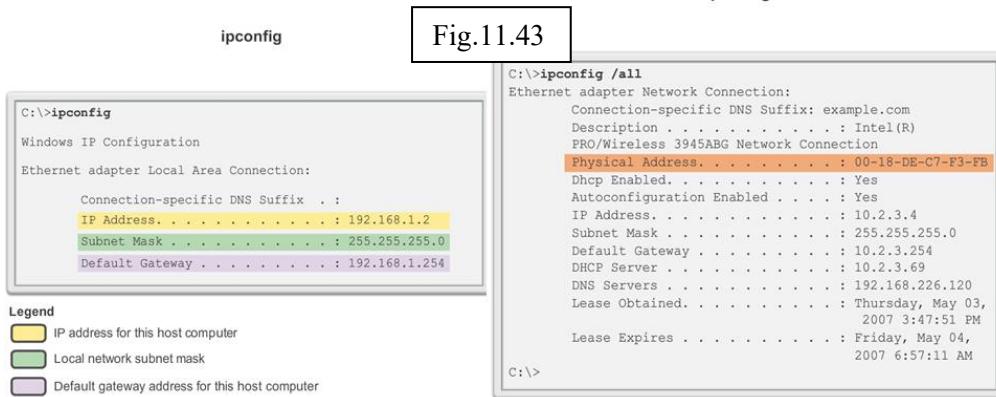
Așa cum se poate vedea în Fig. 1, adresa IP a default gateway de pe hostului poate fi vizualizată prin efectuarea comenzi **ipconfig** pe linia de comandă a unui computer Windows.

Un instrument de examinare a adresei MAC de pe computerul personal este **ipconfig /all**. De remarcat faptul că în Fig. 11.43, adresa MAC a computerului este afișată împreună cu un număr de detalii cu privire la adresarea de nivel 3 a dispozitivului.

În plus, producătorul interfeței de rețea de pe computer poate fi identificat prin partea OUI a adresei MAC. Aceasta poate fi căutat pe Internet.

Serviciul de Client DNS de pe PCurile Windows optimizează performanța rezoluției de nume DNS prin stocarea numelor rezolvate anterior în memorie. Comanda **ipconfig /displaydns** afișează toate intrările DNS stocate pe un sistem de computer Windows.

ipconfig /all



```
C:\>ipconfig /all
Windows IP Configuration

Ethernet adapter Network Connection:
  Connection-specific DNS Suffix . :
    IP Address . . . . . : 192.168.1.2
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.254

C:\>
```

Fig.11.43

Comanda **arp** permite crearea, editarea și afișarea mapărilor adreselor fizice cu adrese IPv4 cunoscute. Comanda **arp** este executată din Windows command prompt.

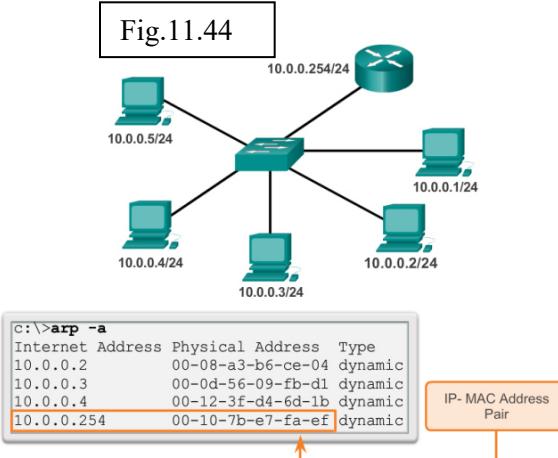
Pentru a executa comanda **arp**, pe command prompt al hostului, introducem C:\host1>**arp -a**

Așa cum se poate observa în Fig. , comanda **arp -a** listează toate dispozitivele aflate curent în ARP cacheul hostului, ce include adresa IPv4, adresa fizică și tipul de adresare (static/dinamic) pentru fiecare dispozitiv.

Cache poate fi “curățat” prin utilizarea comenzi **arp -d** în cazul în care administratorul de rețea vrea să repopuleze cache cu informații actualizate.

Notă: ARP cache conține numai informații de la dispozitivele ce au fost accesate recent. Pentru a ne asigura de faptul că ARP cache este populat, **pinguim** un dispozitiv pentru a avea o intrare în ARP table.

Learning About the Nodes on the Network



Examinăm ieșirea comenții **show cdp neighbors** din Fig.11.45, cu topologia din Fig.11.46. Remarcăm faptul că a strâns unele informații detaliate cu privire la R2 și switchul conectat pe interfață Fast Ethernet a R3.

CDP este un protocol proprietar Cisco ce rulează la nivelul legătură de date. Deoarece CDP rulează la nivelul legătură de date, două sau mai multe dispozitive de rețea, cum ar fi routere ce suportă diferite protocoale de nivel rețea, pot învăța unele despre celelalte, chiar dacă nu există o conectivitate de nivel 3.

Atunci când un dispozitiv Cisco bootează, CDP este pornit în mod implicit. CDP descoperă automat dispozitive Cisco vecine ce rulează CDP, indiferent de ce protocol de nivel 3 sau sătă rulează. CDP schimbă informații de dispozitiv hardware și software cu vecinii CDP direct conectați.

CDP oferă următoarele informații cu privire la fiecare dispozitiv vecin CDP:

- **Identifierii de dispozitiv** – De exemplu, *hostname config.t* pe un switch.
- **Lista de adresa** – O adresă de nivel rețea pentru fiecare protocol suportat.
- **Identifier de port** – Numele portului local sau de la distanță în forma unui string de caractere ASCII, cum ar fi *ethernet0*.
- **Lista de capacitați** – De exemplu, dacă dispozitivul respectiv este un router sau switch.
- **Platforma** – Platforma hardware a dispozitivului; de exemplu, un router din seria Cisco 1841.

Comanda **show cdp neighbors detail** afișează adresa IP a unui dispozitiv vecin. CDP va afișa adresa IP a vecinului indiferent dacă putem da **ping** sau nu vecinului. Această comandă este foarte utilă atunci când routerele Cisco nu pot accesa legătura lor comună. Comanda **show cdp neighbors detail** va ajuta la determinarea dacă unul dintre vecini are o eroare de *config.tie IP*.

Pentru situațiile de descoperire de rețea, cunoașterea adresei IP a vecinului CDP este adesea singura informație necesară pentru a accesa de la distanță prin Telnet respectivul dispozitiv.

Din motive evidente, CDP reprezintă un risc de securitate. Deoarece unele versiuni IOS trimit advertisements CDP implicit, este important să simă dezactivăm CDP.

Pentru a dezactiva CDP global, utilizăm comanda **no cdp run** în modul de *config.re* global. Pentru a dezactiva CDP pe o interfață, utilizăm comanda pe interfață **no cdp enable**.

```
R3#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge,
                  B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP,
                  r - Repeater, P - Phone

Device ID Local Intrfce Holdtme Capability Platform Port ID
S3      Fas 0/0        151   S I       WS-C2950  Fas 0/6
R2      Ser 0/0/1      125   R         1841     Ser 0/0/1

R3#show cdp neighbors detail
Device ID: R2
Entry address(es):
  IP address : 192.168.1.2
Platform: Cisco 1841, Capabilities: Router Switch IGMP
Interface: Serial0/0/1, Port ID (outgoing port): Serial0/0/1
Holdtime : 161 sec

Version :
Cisco IOS Software, 1841 Software (C1841-ADVIPSERVICESK9-M),
Version 12.4(10b), RELEASE SOFTWARE (fc3)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Fri 19-Jan-07 15:15 by prod_rel_team

advertisement version: 2
VTP Management Domain: ''

-----
Device ID: S3
Entry address(es):
Platform: cisco WS-C2950-24, Capabilities: Switch IGMP
Interface: FastEthernet0/0, Port ID (outgoing port):
FastEthernet0/11
Holdtime : 148 sec

Version :
Cisco Internetwork Operating System Software
IOS (tm) C2950 Software (C2950-I6Q4L2-M), Version 12.1(9)EA1,
RELEASE SOFTWARE (fc1)
```

Fig.11.45

```

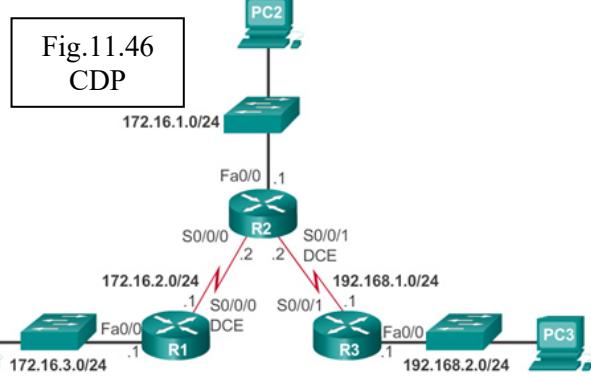
Entry address(es):
Platform: cisco WS-C2950-24, Capabilities: Switch IGMP
Interface: FastEthernet0/0, Port ID (outgoing port):
FastEthernet0/11
Holdtime : 148 sec

Version :
Cisco Internetwork Operating System Software
IOS (tm) C2950 Software (C2950-I6Q4L2-M), Version 12.1(9)EA1,
RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2002 by cisco Systems, Inc.
Compiled Wed 24-Apr-02 06:57 by antonino

advertisement version: 2
Protocol Hello: OUI=0x00000C, Protocol ID=0x0112; payload
len=27, value=0000000FFFFFFF0
10231FF00000000000000AB769F6C0FF0000
VTP Management Domain: 'CCNA3'
Duplex: full

R3#

```



În același mod în care comenzi și utilitarele sunt utilizate pentru verificarea unei configurații de host, comenzi pot fi utilizate pentru verificarea interfețelor de pe dispozitivele intermediare. IOSul oferă comenzi de verificare a funcționării interfețelor routerului și switchului.

11.9 Verificarea interfețelor de pe router

Una dintre cele mai frecvente comenzi utilizate este comanda **show ip interface brief**. Această comandă oferă o ieșire mai compactă decât comanda **show ip interface**. Oferă un rezumat al informațiilor cheie pentru toate interfețele de rețea de pe router.

În Fig. 1 este arătată topologia utilizată în acest exemplu.

Ieșirea **show ip interface brief** afișează toate interfețele de pe router, adresa IP atribuită pe fiecare interfață, dacă există, și statusul funcțional al interfeței.

Conform ieșirii, interfața FastEthernet 0/0 are o adresa IP 192.168.254.254. Ultimele două coloane din această linie arată statusul de nivel 1 și 2 al acestei interfețe. Cuvântul “**up**” în coloana de Status arată că această interfață este operațională la nivel 1. Cuvântul “**up**” în coloana de protocol indică faptul că protocolul de nivel 2 este funcțional.

De asemenea, remarcăm faptul că interfața Serial 0/0/1 nu a fost activată. Acest lucru este indicat prin **administratively down** din coloana Status.

Ca pe orice dispozitiv final, putem verifica conectivitatea de nivel 3 cu comenzi **ping** și **traceroute**. În acest exemplu ambele comenzi, **ping** și **traceroute**, arată o conectivitate cu succes.

11.10 Verificarea interfețelor de pe switch

Comanda **show ip interface brief** poate fi utilizată de asemenea pentru a verifica statusul interfețelor de pe switch. Adresa IP pentru switch este aplicată pe o interfață VLAN. În acest caz, interfața VLAN1 are atribuită adresa IP 192.168.254.250, a fost activată și este operațională.

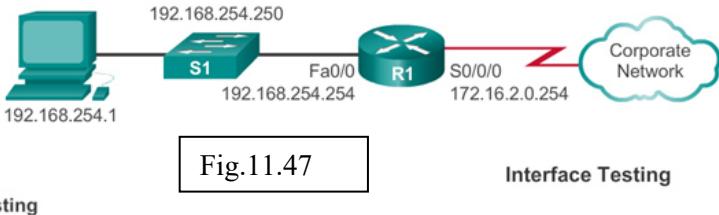
Acest output afișează și faptul că interfața FastEthernet0/1 este down. Acest lucru indică faptul că fie niciun dispozitiv nu este conectat la interfață, fie că dispozitivul care este conectat la această interfață are o interfață de rețea ce nu este funcțională.

Ieșirea arată și faptul că FastEthernet0/2 și FastEthernet0/3 sunt operaționale. Acest lucru este indicat prin faptul că ambele coloane, Status și Protocol, sunt **up**.

Switchul își poate testa de asemenea conectivitatea de nivel 3 cu ajutorul comenziilor **show ip interface brief** și **traceroute**. În acest exemplu, ambele comenzi, **ping** și **traceroute**, arată o conectivitate cu succes.

Este important de reținut faptul că o adresă IP nu este necesară pentru ca un switch să își efectueze sarcina de frame forwarding de nivel 2. O adresă IP este necesară numai atunci când

switchul va fi gestionat prin rețea cu ajutorul Telnet sau SSH. Dacă administratorul de rețea planuiește o conectare de la distanță pe switch dintr-o locație exterioară LANului local, atunci un default gateway trebuie să fie configurat.



```
R1# show ip interface brief
Interface      IP-Address      OK? Method Status      Protocol
FastEthernet0/0 192.168.254.254  YES NVRAM up        up
FastEthernet0/1 unassigned       YES unset  down      down
Serial0/0/0     172.16.0.254   YES NVRAM up        up
Serial0/0/1     unassigned       YES unset  administratively down
down

R1# ping 192.168.254.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.254.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R1# traceroute 192.168.0.1
Type escape sequence to abort.
Tracing the route to 192.168.0.1
 1 172.16.0.253 8 msec 4 msec 8 msec
 2 10.0.0.254 16 msec 16 msec 8 msec
 3 192.168.0.1 16 msec * 20 msec

S1# show ip interface brief
Interface      IP-Address      OK? Method Status      Protocol
Vlan1          192.168.254.250  YES manual up        up
FastEthernet0/1 unassigned       YES unset  down      up
FastEthernet0/2 unassigned       YES unset  up        up
FastEthernet0/3 unassigned       YES unset  up        up
<output omitted>

S1# ping 192.168.254.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.254.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

S1# traceroute 192.168.0.1
Type escape sequence to abort.
Tracing the route to 192.168.0.1
 1 192.168.254.254 4 msec 2 msec 3 msec
 2 172.16.0.253 8 msec 4 msec 8 msec
 3 10.0.0.254 16 msec 16 msec 8 msec
 4 192.168.0.1 16 msec * 20 msec
```

11.11 Gestionarea fișierelor de config.re IOS

Pentru a implementa și securiza o rețea mică, în sarcina administratorului de rețea intră și gestionarea fișierelor de config.re. Gestionarea fișierelor de config.re este importantă pentru scopurile de recuperare și backup în cazul producerii unui eveniment de defecțiune a unui dispozitiv.

Cisco IOS File System (IFS) oferă o singura interfață tuturor sistemelor de fișier utilizate de un router, inclusiv:

- *Sistemele de fișier de memorie flash.*
- *Sistemele de fișier de rețea (TFTP și FTP).*
- *Orice alte endpoint de citire sau scriere a datelor, cum ar fi NVRAM, config.ția running, ROM și altele.*

Cu Cisco IFS, toate fișierele pot fi vizualizate și clasificate (fișier imagine, text și așa mai departe), inclusiv fișierele de pe serverele de la distanță. De exemplu, este posibilă vizualizarea unui fișier de config.re de pe un server de la distanță pentru a verifica faptul că este fișierul corect de config.re, înainte de încărcarea acestuia pe router.

Cisco IFS permite administratorului să se deplaceze în directoare diferite și să listeze fișierele într-un director și să creeze subdirectoare în memoria flash sau pe un disk. Directoarele disponibile depind de dispozitiv.

Fig.11.48 afișează ieșirea comenzi **show file systems**, ce listează toate sistemele de fișiere disponibile pe un router Cisco 1941, în acest exemplu. Această comandă oferă informații utile, cum ar fi cantitatea de memorie liberă și disponibilă, tipul de sistem de fișier și permisiunile sale. Permisiunile includ read only (ro), write only (wo) și read and write (rw), arătate în coloana Flags din ieșirea comenzi.

Deși există multe sisteme de fișiere listate, cele ce prezintă interes pentru noi sunt sistemele de fișier tftp, flash și nvram.

De remarcat faptul că sistemul de fișier flash are un asterisk ce îl precede. Acest lucru indică faptul că flash este sistemul de fișier implicit actual. Bootable IOS este localizat în flash; prin urmare simbolul # este anexat la flash indicând faptul că acesta este un disk bootabil.

Fișierul Sistemului Flash – Fig.11.49 listează conținutul sistemului de fișier default actual, care în acest caz este flash, indicat prin asterisks precedente listate în Fig. anterioară. Există mai multe fișiere localizate în flash, însă ne interesează în mod special ultima listare. Aceasta este numele imaginii de fișier curent Cisco IOS ce rulează în RAM.

Fișierul Sistemului NVRAM – Pentru a vedea conținutul NVRAM, trebuie să schimbăm sistemul de fișier default curent folosind comanda **cd** (change directory), aşa cum se poate vedea în Fig.11.50. Comanda **pwd** (present working directory) verifică faptul că vedem directorul NVRAM. Comanda **dir** (directory) listează conținutul NVRAM. Deși există mai multe fișiere de config.re listate, un interes specific este asupra fișierului startup-conFig.ation.

File Systems					Flash
Fig.11.48					Fig.11.49
<pre>Router#show file systems File Systems:</pre>					<pre>Router#dir Directory of flash0:/</pre>
<pre>Size(b) Free(b) Type Flags Prefixes - - opaque rw archive: - - opaque rw system: - - opaque rw tmpsys: - - opaque rw null: - - network rw tftp: * 256487424 183234560 disk rw flash0: flash:#</pre>					<pre>1 -rw- 2903 Sep 7 2012 06:58:26 +00:00 cpconfig- 19xx.cfg 2 -rw- 3000320 Sep 7 2012 06:58:40 +00:00 cpexpress.tar 3 -rw- 1038 Sep 7 2012 06:58:52 +00:00 home.shtml 4 -rw- 122880 Sep 7 2012 06:59:02 +00:00 home.tar 5 -rw- 1697952 Sep 7 2012 06:59:20 +00:00 securedesktop- ios-3.1.1.45-k9.pkg 6 -rw- 415956 Sep 7 2012 06:59:34 +00:00 ssclient-win- 1.1.4.176.pkg 7 -rw- 67998028 Sep 26 2012 17:32:14 +00:00 c1900- universalk9- mz.SPA.152-4.M1.bin</pre>
					256487424 bytes total (183234560 bytes free)
NVRAM					
Fig.11.50					
<pre>Router#cd nvram: Router#pwd nvram:/ Router#dir Directory of nvram:/</pre>					
<pre>253 -rw- 1156 <no date> startup-config 254 ---- 5 <no date> private-config 255 -rw- 1156 <no date> underlying-config 1 -rw- 2945 <no date> cwpw_inventory 4 ---- 58 <no date> persistent-data 5 -rw- 17 <no date> ecfm_ieee_mib 6 -rw- 559 <no date> IOS-Self-Sig#1.cer</pre>					
262136 bytes total (254779 bytes free)					

Cu sistemul de fișiere flash de pe switch Cisco 2960, putem copia fișierele de config.re și înregistra (încărca și descărca) imagini software.

Comanda de vizualizarea a sistemelor de fișiere de pe un switch Catalyst este aceeași ca cea de pe un router Cisco: **show file systems**, aşa cum este arătat și în Fig.11.51.

Mai multe comenzi UNIX de bază sunt suportate pe switchurile și routerele Cisco: **cd** pentru schimbarea unui sistem de fișiere sau directore, **dir** pentru a afișa directoarele de pe un sistem de fișier și **pwd** pentru a afișa directorul curent.

Switch#show file systems					
File Systems:					
Size(b)	Free(b)	Type	Flags	Prefixes	
*	32514048	20887552	flash	rw	flash:
-	-	opaque	rw	vb:	
-	-	opaque	ro	bs:	
-	-	opaque	rw	system:	
-	-	opaque	rw	tmpsys:	
65536	48897	nvram	rw	nvram:	
-	-	opaque	ro	xmodem:	
-	-	opaque	ro	ymodem:	
-	-	opaque	rw	null:	
-	-	opaque	ro	tar:	
-	-	network	rw	tftp:	
-	-	network	rw	rcp:	
-	-	network	rw	http:	
-	-	network	rw	ftp:	
-	-	network	rw	scp:	
-	-	network	rw	https:	
-	-	opaque	ro	cns:	

Fig.11.51

11.11.1 Back up și restaurarea fișierelor de config.re

Fișierele de config.re pot fi salvate/arhivate într-un fișier text utilizând Tera Term.

Cum se poate vedea și în Fig. , pașii sunt:

- **Pasul 1.** Din meniul File, apăsăm **Log**.
- **Pasul 2.** Alegem locația de salvare a fișierului. Tera Term va începe să capteze text.
- **Pasul 3.** După ce captura a început, executăm comanda **show running-config** sau **show startup-config** sau în promptul EXEC privilegiat. Textul afișat în fereastra de terminal va fi direcționat în fișierul ales.
- **Pasul 4.** După ce captura este completă, alegem **Close** din Tera Term: Log window.
- **Pasul 5.** Vizualizăm fișierul pentru a verifica faptul că nu a fost corupt.

11.11.2 Restaurarea config.ților text

O config.ție poate fi copiată de pe un fișier pe dispozitiv. La copierea de pe un fișier text pe o fereastră terminal, IOS execută fiecare linie a config.ției text ca o comandă. Acest lucru înseamnă ca fișierul va necesita editare pentru a ne asigura de faptul că parolele criptate sunt în text clar și de faptul că textul non-comandă precum "--More--" și mesajele IOS sunt înlăturate. Acest proces va fi prezentat în laborator.

Pe CLI, dispozitivul trebuie să fie setat în modul de config.re globală pentru a primi comenzi din fișierul text ce sunt copiate în fereastra terminal.

Folosind Tera Term, pașii sunt:

- **Pasul 1.** Din meniul file, click **Send file**.
- **Pasul 2.** Localizăm fișierul ce va fi copiat în dispozitiv și apăsăm **Open**.
- **Pasul 3.** Tera Term va copia fișierul în dispozitiv.

Textul din fișier va fi aplicat ca și comandă în CLI și va deveni running-config.ion de pe dispozitiv. Aceasta este o metodă convenabilă pentru config.rea manuală a unui router.

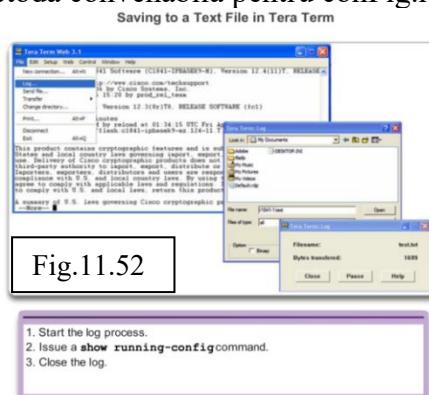


Fig.11.52

1. Start the log process.
2. Issue a **show running-config** command.
3. Close the log.

11.11.3 Backupul ConFig.ților cu TFTP

Copii ale fișierelor de conFig.re ar trebui să fie stocate ca fișiere de backup în cazul apariției unei probleme. Fișierele de conFig.re pot fi stocate pe un server Trivial File Transfer Protocol (TFTP) sau pe un drive USB. Un fișier de conFig.re ar trebui să fie de asemenea inclus în documentația de rețea.

Pentru a salva conFig.ția running sau conFig.ția startup pe un server TFTP folosim fie comanda **copy running-config tftp**, fie comanda **copy startup-config tftp**, arătate în Fig. . Urmăm pași de mai jos pentru a copia conFig.ția running pe un server TFTP:

- **Pasul 1.** Introducem comanda **copy running-config tftp**.
- **Pasul 2.** Introducem adresa IP a hostului pe care fișierul de conFig.re va fi stocat.
- **Pasul 3.** Introducem numele ce va fi atribuit fișierului de conFig.re.
- **Pasul 4.** Apăsăm tasta Enter pentru a confirma fiecare alegere.

11.11.4 Restaurarea conFig.ților cu TFTP

Pentru a restaura conFig.ția running sau conFig.ția startup de pe un server TFTP folosim fie comanda **copy tftp running-config**, fie comanda **copy tftp startup-config**. Urmăm acești pași pentru a restaura conFig.ția running de pe un server TFTP:

- **Pasul 1.** Introducem comanda **copy tftp running-config**.
- **Pasul 2.** Introducem adresa IP a hostului pe care fișierul de conFig.re va fi stocat.
- **Pasul 3.** Introducem numele ce va fi atribuit fișierului de conFig.re.
- **Pasul 4.** Apăsăm tasta Enter pentru a confirma fiecare alegere.

```
Router#copy running-config tftp
Remote host []? 131.108.2.155
Name of configuration file to write[tokyo-config]?tokyo.2
Write file tokyo.2 to 131.108.2.155? [confirm]
Writing tokyo.2 !!!!!!! [OK]
```

Caracteristica de stocare Universal Serial Bus (USB) permite anumitor modele de routere Cisco să suporte drivere flash USB. Caracteristica USB flash oferă o capacitate de stocare secundară opțională și un dispozitiv suplimentar de boot. Imaginele, conFig.țile și alte fișiere pot fi copiate pe sau de pe un Cisco USB de memorie flash cu aceeași încredere prin care se stochează și restaurează fișierele cu ajutorul cardului Compact Flash. În plus, routerele cu servicii integrate modulare pot boota orice imagine IOS Software salvată pe memoria USB flash.

Modulele flash Cisco USB sunt disponibile în versiuni de 64MB, 128 MB și 256MB.

Pentru a fi compatibil cu un router Cisco, un drive USB flash trebuie să fie formatat într-un format FAT16. Dacă nu, comanda show file systems va afișa o eroare ce indică un sistem de fișier incompatibil.

Următorul exemplu este pentru utilizarea comenzii dir pe un sistem de fișier USB:

```
Router# dir usbflash0:
Directory of usbflash0:/
1 -rw- 30125020 Dec 22 2032 05:31:32 +00:00 c3825-entservicesk9-mz.123-14.T
63158272 bytes total (33033216 bytes free)
```

Ideal, USB flash poate păstra mai multe copii ale IOSului și mai multe conFig.ții de router. USB flash permite unui administrator să mute mai ușor și să copieze respectivele fișiere IOS și conFig.țile de la router la router, de mai multe ori, iar procesul de copiere poate avea loc de mai multe ori mai rapid decât peste un LAN sau WAN. Reținem faptul că IOS ar putea să nu

recunoască dimensiunea adecvată a USB flash, însă acest lucru nu înseamnă neapărat faptul că flash este neacceptat. În plus, porturile USB de pe un router sunt de obicei USB 2.0, cum se vede și în Fig.11.53.

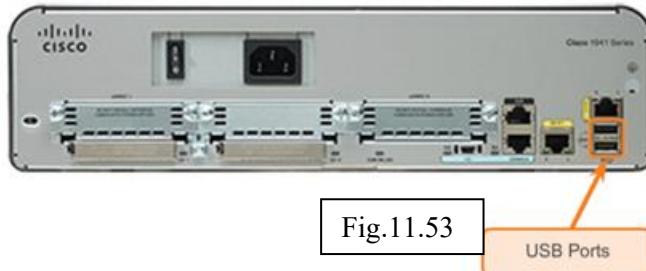


Fig.11.53

11.11.5 Configurații de backup cu un drive USB flash

La copierea pe un port USB, este o bună idee de efectuare a comenzi **show file systems** pentru verificarea faptului că USB drive este acolo și confirmă numele, aşa cum se poate vedea în Fig. 11.54.

Apoi, folosim comanda **copy run usb flash0:/** pentru a copia fișierul de config.re pe driveul flash USB. Ne asigurăm că utilizăm numele lui flash drive, aşa cum este indicat în sistemul de fișier. Slash este opțional, însă indică directorul rădăcină al driveului flash USB.

IOS va solicita numele de fișier. Dacă fișierul există deja pe drive flash USB, routerul va solicita suprascriere, aşa cum se poate vedea în Fig. 11.55

Folosim comanda **dir** pentru a vedea fișierul pe driveul USB și folosim comanda **more** pentru a vedea conținutul, aşa cum se poate vedea în Fig.11.56.

11.11.6 Restaurarea config.ților cu un drive flash USB

Pentru a copia înapoi fișierul, va fi necesară editarea fișierului USB R1-Config cu un editor de text pentru a-l face un fișier config valid; altfel, există o multime de intrări ce reprezintă comenzi invalide și nici-o interfață nu va fi ridicată.

R1#copy usbflash0:/R1-Config running-config

Destination filename [running-config]?

```
R1#show file systems
File Systems:
  Size(b)  Free(b)   Type   Flags  Prefixes
  -        -         opaque  rw     archive:
  -        -         opaque  rw     system:
  -        -         opaque  rw     tmpsys:
  -        -         opaque  rw     null:
  -        -         network rw     tftp:
  -        -         disk    rw     flash0: flash:#*
  -        -         disk    rw     flash1:
  262136   249270   nvram   rw     nvram:
  -        -         opaque  wo     syslog:
  -        -         opaque  rw     xmodem:
  -        -         opaque  rw     ymodem:
  -        -         network rw     rpc:
  -        -         network rw     http:
  -        -         network rw     ftp:
  -        -         network rw     scp:
  -        -         opaque  ro     tar:
  -        -         network rw     https:
  -        -         opaque  ro     cns:
  4050042880 3774152704  usbflash rw     usbflash0:
```

Fig.11.54

```
R1#copy running-config usbflash0:
Destination filename [running-config]? R1-Config
5024 bytes copied in 0.736 secs (6826 bytes/sec)
```

Copying to USB flash drive, and no file pre-exists.

Fig.11.55

```
R1#copy running-config usbflash0:
Destination filename [running-config]? R1-Config
%Warning: There is a file already existing with this name
Do you want to over write? [confirm]
5024 bytes copied in 1.796 secs (2797 bytes/sec)
```

Copying to USB flash drive, and the same configuration file already exists on the drive.

```

R1#dir usbflash0:/  

Directory of usbflash0:/  

  1  drw-    0 Oct 15 2010 16:28:30 +00:00 Cisco  

  16 -rw-  5024 Jan  7 2013 20:26:50 +00:00 R1-Config  

4050042880 bytes total (3774144512 bytes free)  

R1#more usbflash0:/R1-Config  

!  

! Last configuration change at 20:19:54 UTC Mon Jan  7 2013 by  

admin version 15.2  

service timestamps debug datetime msec  

service timestamps log datetime msec  

no service password-encryption  

!  

hostname R1  

!  

boot-start-marker  

boot-end-marker  

!  

logging buffered 51200 warnings  

!  

no aaa new-model  

!
no ipv6 cef

```

Fig.11.56

11.11.7 Servicii de rutare integrate

Utilizarea rețelei nu se limitează la întreprinderi mici și organizații mari.

Un alt mediu ce profită din ce în ce mai mult de tehnologia de rețea este locuința. Rețelele de domiciliu sunt utilizate pentru oferirea conectivității și împărțirea Internetului printre mai multe sisteme de computer personale și laptopuri din locuințe. De asemenea permit indivizilor să profite de multiple servicii, cum ar fi partajarea imprimării la o imprimantă de rețea, stocarea centralizată a fotografiilor, muzicii și filmelor pe un dispozitiv network attached storage (NAS); de asemenea permite altor dispozitive de utilizator, cum ar fi tablete, telefoane mobile și chiar electrocasnicelor, precum un televizor, să aibă acces la serviciile de Internet.

O rețea de domiciliu este foarte asemănătoare cu o rețea de întreprindere mică. Însă, unele rețele de domiciliu și mai multe rețele de întreprindere mică nu necesită dispozitive de volum mare, cum ar fi un router dedicat și switchuri. Dispozitivele de dimensiune mai mică, atât timp cât oferă aceeași funcționalitate de routare și switching, reprezintă singura necesitate. Din acest motiv, multe rețele de domiciliu și întreprindere mici utilizează serviciul unui dispozitiv multi-function.

Pentru scopul acestui curs, dispozitivele multi-function vor fi referite ca routere integrate.

Un router integrat este ca și cum am avea mai multe dispozitive diferite conectate împreună. De exemplu, conexiunea dintre un switch și router are loc, însă are loc intern. Atunci când un pachet este trimis de la un dispozitiv la altul, în aceeași rețea, switchul integrat va trimite automat pachetul la dispozitivul destinație. Dacă un pachet este trimis la un dispozitiv de pe o rețea de la distanță, switchul integrat va transmite pachetul conexiunii routerului intern. Routerul intern va determina apoi cea mai bună cale și va transmite pachetul în mod corespunzător căii determinate.

Multe routere integrate oferă capacitați atât de conexiune wireless, cât și switching cablat, și servesc ca access point (AP) în rețeaua wireless, aşa cum se poate vedea și în Fig.11.57. Conectivitatea wireless este o modalitate polulară, flexibilă și eficientă din punct de vedere a costului pentru locuințe și întreprinderi pentru oferirea serviciilor de rețea dispozitivelor finale.

Imaginiile listează unele avantaje și considerații comune pentru utilizarea wireless.

Pentru a suporta rutarea, switching și conectivitatea wireless, multe caracteristici suplimentare pot fi disponibile pe un router integrat, cum ar fi : serviciu DHCP, un firewall și chiar network attached storage services.

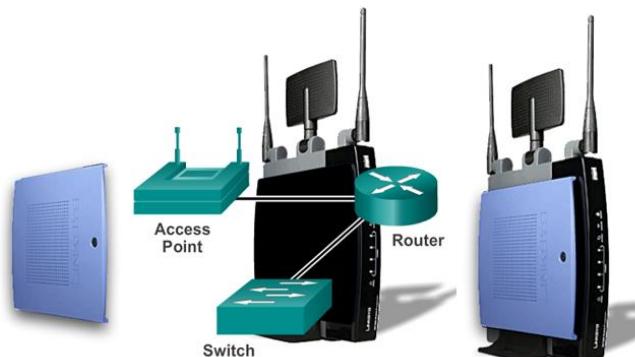
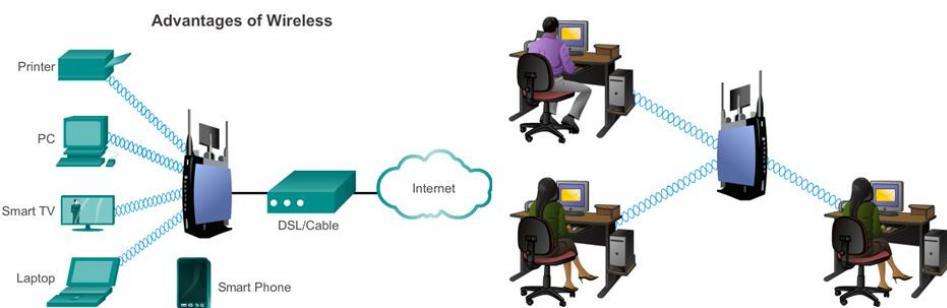


Fig.11.57

Limitations of Wireless



Routerele integrate pot varia de la dispozitive mici destinate pentru locuințe și aplicații de întreprindere mică la dispozitive mai puternice ce suportă filiale de întreprindere.

Un exemplu de acest tip de router integrat este un Linksys wireless router, arătat în Fig.. Acest tip de router integrat este simplu în design și nu are în mod normal componente separate. Acest lucru reduce costul dispozitivului. Însă, în cazul producerii unui eșec, nu este posibilă înlocuirea unei singure componente stricte. Prin urmare, se crează un singur punct de eșec și nu sunt optimizate pentru oricare funcție.

Un alt exemplu de router integrat este Cisco integrated services router sau ISR. Familia de produse Cisco ISR oferă un rangu mare de produse, inclusiv cele destinate pentru medii de small office și home office, cât și cele destinate pentru rețele mari. Multe dintre ISR oferă modularitate și au componente separate pentru fiecare funcție, cum ar fi o componentă de switch și o componentă router. Acest lucru permite componentelor individuale să fie adăugate, înlocuite și actualizate dacă este necesar.

Toate routerele integrate permit setări de bază de configurație cum ar fi parole, adrese IP și setări DHCP, ce sunt aceleași fie că dispozitivul este utilizat pentru a conecta hosturi prin cablu, fie prin wireless. Însă, dacă utilizăm funcționalitatea wireless, parametrii de configurație suplimentari sunt necesari, cum ar fi setări pentru modul wireless, SSID și canalul wireless.

Linksys: Model WRT300N2

Linksys: Model WRT300N2



Fig.11.58

Front View

The Linksys is a simplified, low-cost device that carries out the functionality of multiple network devices (switch, router, wireless access point).

Light emitting diodes (LEDs) indicate the connection status of each port.

Click the LEDs for a description.

Rear View

When connecting a local network using a multifunction device it is important that all local devices are connected to the switch ports.

Click the ports for a description.

11.12 Wireless

Modul wireless se referă la setările wireless IEEE 802.11 standard pe care rețeaua le va vedea. Există patru modificări ale standardului IEEE 802.11 ce descriu caracteristici diferite pentru comunicațiile wireless; ele sunt IEEE 802.11a, IEEE 802.11b, IEEE 802.11g și IEEE 802.11n. Fig. 1 listează mai multe informații cu privire la fiecare standard.

Multe routere wireless integrate suportă 802.11b, 802.11g și 802.11n. Cele trei tehnologii sunt compatibile, însă toate dispozitivele din rețea trebuie să funcționeze cu același standard comun tuturor dispozitivelor. De exemplu, dacă un router 802.11n este conectat la un laptop cu 802.11n, rețeaua va funcționa ca un standard 802.11n. Însă, adăugată o imprimantă wireless 802.11b la rețea, ambele, routerul și laptopul, se vor întoarce să folosească standardul mai înalt 802.11b pentru a comunica toate echipamentele. Prin urmare, păstrarea dispozitivelor wireless mai vechi în rețea va face ca întreaga rețea să funcționeze mai lent. Este important să reținem acest lucru pentru atunci când ne decidem dacă pastrăm sau nu dispozitivele wireless mai vechi.

11.12.1 Service Set Identifier (SSID)

Pot exista mai multe alte rețele wireless în aria de acoperire a unei rețele. Este important ca dispozitivele wireless să se conecteze la WLAN corect. Acest lucru se realizează prin utilizarea unui Service Set Identifier (SSID).

SSID este cu nume case-sensitive, alpha-numeric pentru rețeaua wireless din locuință. Numele poate avea până la 32 de caractere în lungime. SSID este utilizat pentru a spune dispozitelor wireless căruia WLAN aparțin și cu ce alte dispozitive pot comunica. Indiferent de tipul de instalare WLAN, toate dispozitivele wireless dintr-un WLAN trebuie să fie configurate cu același SSID pentru a comunica.

11.12.2 Canalul wireless

Canalele sunt create prin divizarea spectrumului RF disponibil. Fiecare canal este capabil să transporte o conversație diferită. Aceast lucru este similar cu modul în care canalele de televiziune sunt transmise pe un singur mediu. Mai multe APs pot funcționa în apropierea unuia față de altul, atât timp ce utilizează canale diferite de comunicare.



Măsuri de securitate ar trebui să fie planificate și configurate înaintea conectării AP la rețea sau ISP.

Așa cum se poate observa în Fig. 1, unele dintre măsurile de securitate de bază sunt:

- Schimbarea valorilor default pentru SSID, numele de utilizator și parole.
- Dezactivarea broadcast SSID.
- Configurarea criptării folosind WEP sau WPA.

Criptarea este procesul de transformare a datelor astfel încât și dacă sunt interceptate, sunt inutile.

11.12.3 Wired Equivalency Protocol (WEP)

WEP este o caracteristică avansată de securitate ce criptează traficul de rețea care circulă prin intermediul aerului. WEP folosește chei preconFig.te pentru a cripta și decripta datele, aşa cum se poate vedea în Fig. 2.

O cheie WEP este introdusă ca un string de numere și litere și este în general de 64 de biți sau 128 de biți. În unele cazuri, WEP suportă și chei de 256 de biți. Pentru a simplifica crearea și introducerea acestor chei, mai multe dispozitive includ o opțiune de Passphrase. Passphrase este un mod ușor de reținere a cuvântului sau frazei utilizată pentru a genera automat o cheie.

Pentru ca WEP să funcționeze, AP, cât și toate dispozitivele wireless ce au permisiune de acces la rețea trebuie să aibă aceeași cheie WEP introdusă. Fără această cheie, dispozitivele nu vor fi capabile să înțeleagă transmisia wireless.

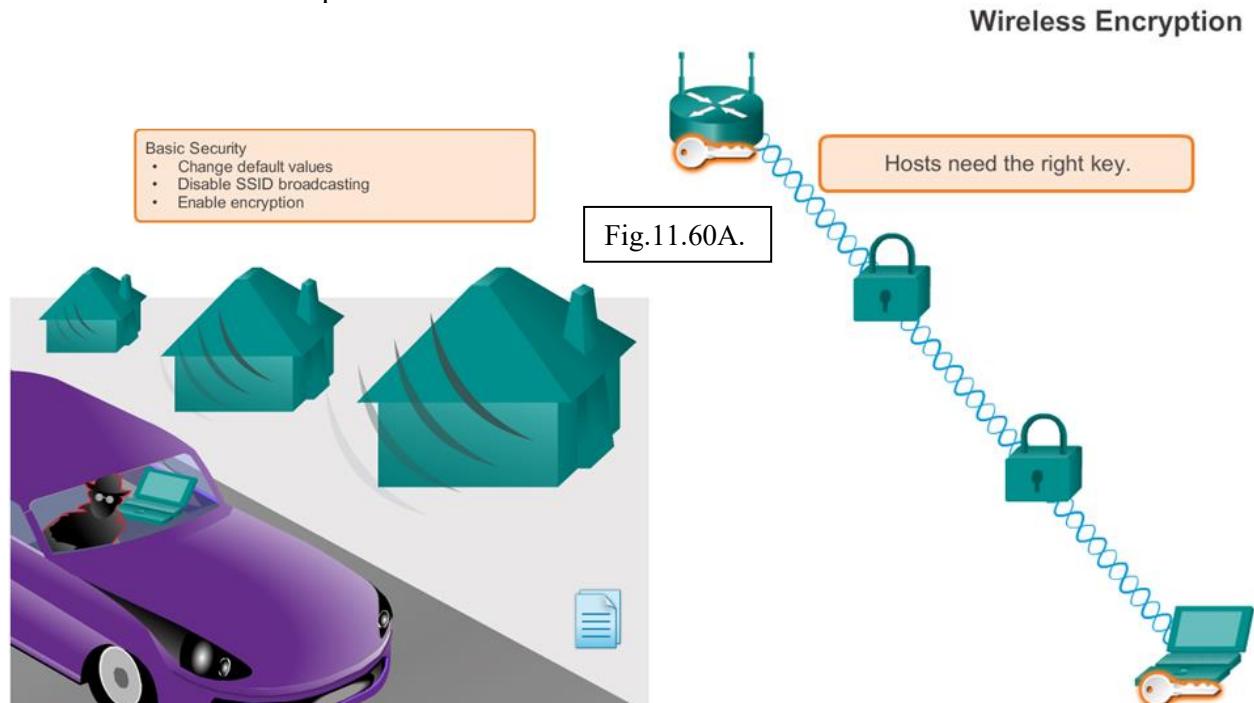
Există unele slăbiciuni cu WEP, inclusiv utilizarea unei chei statice pe toate dispozitivele ce au activat WEP. Există aplicații disponibile atacatorilor ce pot fi folosite pentru descoperirea cheii WEP. Aceste aplicații sunt disponibile în Internet. O dată ce atacatorul a aflat cheia, are acces complet la toate informațiile transmise.

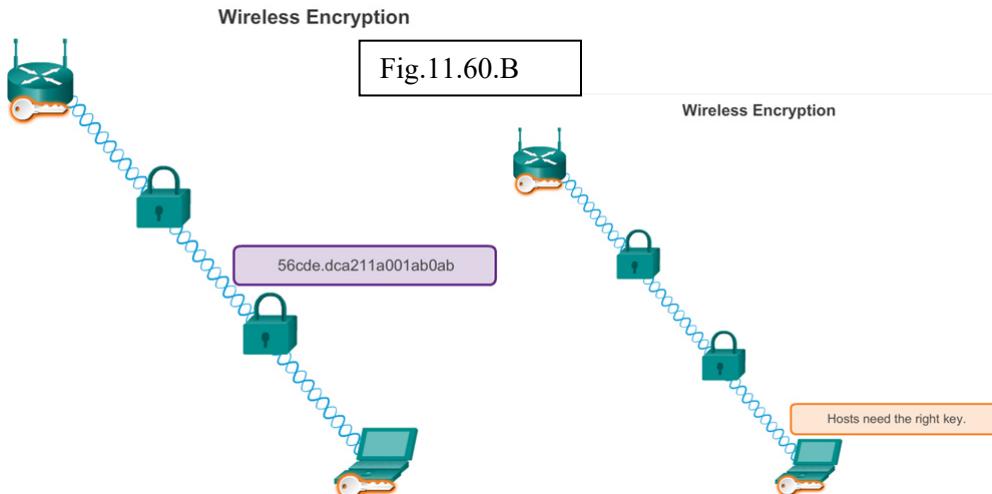
O modalitate de a combate această vulnerabilitate este schimbarea frecventă a cheii. Un alt mod este utilizarea unei forme mai avansate și securizate de criptare numită Wi-Fi Protected Access (WPA).

11.12.4 Wi-Fi Protected Access (WPA)

WPA utilizează de asemenea chei de criptare de 64 de biți până la 256 de biți. Însă, WPA, spre deosebire de WEP, generează chei noi dinamice de fiecare dată când un client stabilește o conexiune cu AP. Din acest motiv, WPA este considerat mai sigur decât WEP deoarece este mult mai dificil de "spart".

Există mai multe implementări de securitate ce pot fi configurate pe un AP wireless, inclusiv filtrarea de adresă MAC, autentificarea și filtrarea traficului. Însă, aceste implementări de securitate nu intră în scopul acestui curs.





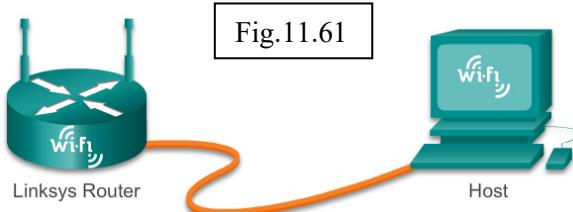
11.13 ConFig.rea routerului integrat

Un router Linksys wireless este un dispozitiv comun utilizat în rețele de locuință și de întreprindere mici și va fi utilizat în acest curs pentru demonstrarea conFig.ărilor de bază pe un router integrat. Un dispozitiv Linksys tipic oferă de la cinci la opt porturi Ethernet pentru conectivitate cablată, pentru a acționa ca un wireless access point. Dispozitivul Linksys acționează de asemenea ca un server DHCP și ca un mini-webserver ce suportă web bazându-se pe graphical user interface (GUI).

11.13.1 Accesarea și conFig.rea unui Linksys Router

Inițial, accesăm routerul prin cablarea unui computer la unul dintre porturile routerului LAN Ethernet, aşa cum se poate vedea în Fig. . O dată cablat, dispozitivul conectat automat va obține informații de adresare IP, inclusiv o adresă de default gateway, de la routerul integrat. Adresa de default gateway este adresa IP a dispozitivului Linksys. Verificăm setările de rețea ale computerului folosind comanda **ipconfig /all** pentru a obține această adresă. Acum putem introduce adresa IP într-un browser web de pe computer pentru a accesa web-based conFig.ation GUI.

Dispozitivul Linksys are un conFig.ări implicită ce permite switching și servicii de rutare de bază. Este de asemenea conFig.ări implicită ca un server DHCP. Sarcini de conFig.ări de bază, cum ar fi schimbarea numelui de utilizator și a parolei implicite, schimbarea adresei default IP a Linksys și chiar rangeurile de adresă IP DHCP implicită, ar trebui să fie efectuate înaintea ca AP să fie conectat la o rețea live.



Pentru a activa conectivitate wireless, modul wireless, SSID, canalul RF și orice mecanism de Securitate Wi-Fi, trebuie să fie configurați.

Mai întâi, selectăm modul de wireless corect, aşa cum se poate vedea în Fig.11.62. La selectarea modului sau standardului wireless, fiecare mod include o anumită cantitate de

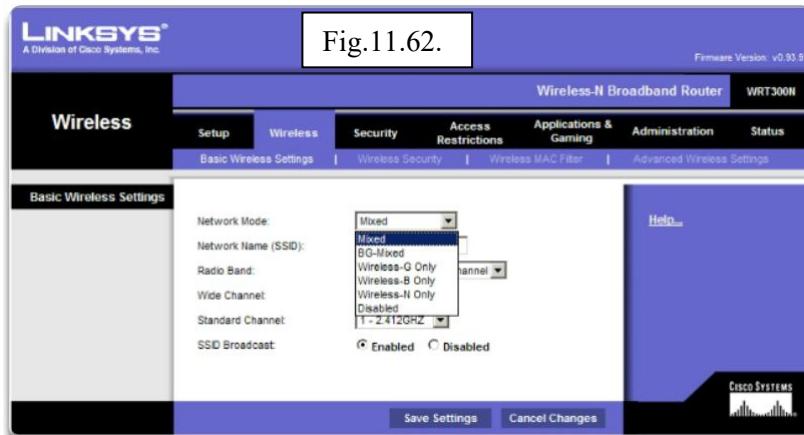
overhead. Dacă toate dispozitivele din rețea folosesc același standard, selectarea modului asociat cu respectivul standard limitează cantitatea de overhead suportată. De asemenea, crește securitatea prin interzicerea dispozitivelor cu standarde diferite să se conecteze. Însă, dacă dispozitivele ce utilizează standarde diferite trebuie să acceseze rețea, poate fi selectat modul mixed. Performanța rețelei va scădea având în vedere overhead suplimentar al tuturor modurilor suportate.

Apoi, setăm SSID. Toate dispozitivele ce doresc să participe la WLAN trebuie să utilizeze același SSID. Din motive de securitate, SSID default ar trebui să fie schimbat. Pentru a permite detectia ușoară a WLAN de către clienți, SSID este broadcast implicit. Este posibilă dezactivarea caracteristicii de broadcast a SSID. Dacă SSID nu este broadcast, clienții wireless vor trebui să aibă această valoare configată manual.

Alegerea canalului RF utilizat pe routerul integrat trebuie să se facă relativ cu alte rețele wireless din împrejurime.

Rețelele wireless adiacente trebuie să utilizeze canale ce nu se suprapun pentru a optimiza throughputul. Multe puncte de acces oferă acum o alegere de a permite routerului să localizeze automat canalul cel mai puțin congestionat.

La final, alegem mecanismul de criptare preferat și introducem o cheie sau passphrase.



11.13.2 Configurarea unui client wireless

Un host wireless, sau un client, este definit ca orice dispozitiv ce conține wireless NIC și software de client wireless. Acest software de client permite ca hardware să participe în WLAN. Dispozitivele includ: unele telefoane mobile, laptopuri, desktop PCuri, imprimante, televizoare, tablete etc.

Pentru ca un client wireless să se conecteze la WLAN, setarea de configurație de client trebuie să corespundă cu cea de pe routerul wireless. Acest lucru include SSID, setările de securitate și informațiile de canal (dacă respectivul canal a fost setat automat). Aceste setări sunt specificate în softwareul de client.

Softwareul de client wireless utilizat poate fi software integrat în sistemul de operare al dispozitivului sau poate fi un utilitar software wireless, de sine stătător, descarcabil, destinat pentru interacțiunea cu wireless NIC.

O dată ce softwareul de client este configurație, verificăm legătura dintre client și AP.

Deschidem ecranul Link Information wireless pentru a afișa informații cum ar fi: connection data rate, statusul conexiunii și canalul wireless utilizat, aşa cum se poate vedea în Fig.. Caracteristica Link Information, dacă este disponibilă, afișează puterea semnalului curent și calitatea semnalului wireless.

În plus, pentru a verifica statusul conexiunii, verificăm faptul că datele pot fi într-adevăr transmise. Unul dintre cele mai comune teste de verificare a transmisiei de date cu succes este testul **ping**. Dacă **ping** e cu succes, transmisia datelor este posibilă.



11.14 Concluzii Capitolul 11

Pentru a îndeplini cerințele de utilizator, chiar și rețelele mici necesită planificare și design, aşa cum se poate vedea în Fig.11.64. Planificarea asigură faptul că toate cerințele, factorii de cost și opțiunile de implementare sunt luate în considerare. O parte importantă a designului de rețea este încrederea, scalabilitatea și disponibilitatea.



When planning any network consider...

- Cost
- Ports
- Speed
- Expandability
- Manageability

Fig.11.64

Suportarea și creșterea unei rețele mici necesită ca noi să fim familiari cu protocolele și aplicațiile de rețea ce rulează în rețea. Analizoarele de protocol permit unui profesionist de rețea să compileze rapid informații statistice cu privire la fluxurile de trafic dintr-o rețea. Informațiile adunate de către analizorul de protocol sunt analizate în funcție de sursa și destinația traficului, cât și în funcție de tipul de trafic transmis. Această analiză poate fi folosită de către un tehnician de rețea pentru a lua decizii despre modul în care să gestioneze traficul mai eficient. Protocolele de rețea comune includ: DNS, Telnet, SMTP, POP, DHCP, HTTP și FTP.

Este important să luăm în considerare amenințările de securitate și vulnerabilitățile atunci când planificăm o implementare de rețea. Toate dispozitivele de rețea trebuie să fie securizate.

Acestea includ routerele, switchurile, dispozitivele de utilizator final și chiar dispozitivele de securitate. Rețelele trebuie să fie protejate împotriva softwareului rău intenționat, cum ar fi virusii, caii Troieni și vermii. Softwareul antivirus poate detecta mulți viruși și aplicații cai Troian și poate preveni împrăștierea lor în rețea. Cel mai eficient mod de combatere a atacului vierme este descărcarea actualizărilor de securitate de la un furnizor de sistem de operare și “peticirea” tuturor sistemelor vulnerabile.

Rețelele trebuie să fie de asemenea protejate de atacurile de rețea. Atacurile de rețea pot fi clasificate în trei mari categorii: de recunoaștere, atacuri de acces și denial of service. Există multe moduri de protecție a unei rețele de atacurile de rețea.

- Serviciile de securitate de rețea Authentication, Authorization, și Accounting (AAA, sau “triple A”) oferă cadrul primar al setării controlului accesului pe un dispozitiv de rețea. AAA este un mod de control asupra celor ce au permis accesul la rețea (autentificare), ce pot face în acest timp (autorizare) și vizualizarea acțiunilor efectuate în timpul accesării rețelei (contabilitate).
- Un firewall este unul dintre cele mai eficiente instrumente de securitate disponibile pentru protejarea utilizatorilor interni din rețea de amenințările externe. Un firewall se află între două sau mai multe rețele și controlează traficul dintre ele și ajută la prevenirea accesului neautorizat.
- Pentru a proteja dispozitivele de rețea, este important să utilizăm parole puternice. De asemenea, atunci când accesăm dispozitivele de rețea de la distanță, este recomandat să activăm SSH, în schimb Telnet nesecurizat.

După ce rețeaua a fost implementată, un administrator de rețea trebuie să fie capabil să monitorizeze și să mențină conectivitatea rețelei. Există mai multe comenzi disponibile pentru acest lucru. Pentru testarea conectivității rețelei cu destinații locale și de la distanță, sunt utilizate comenzi precum **ping**, **telnet** și **traceroute**.

Pe dispozitivele cu IOS, comanda **show version** poate fi utilizată pentru a verifica și depana unele componente hardware și software de bază folosite în timpul procesului de bootup. Pentru a vizualiza informații pentru toate interfețele de pe un router, comanda **show ip interface** este utilizată. Comanda **show ip interface** poate fi de asemenea utilizată pentru vizualizarea unui output mai compact decât al comenzi **show ip interface**. **Cisco Discovery Protocol** (CDP) este un protocol proprietar Cisco ce rulează la nivelul legătură de date. Deoarece CDP operează la nivelul legătură de date, două sau mai multe dispozitive de rețea Cisco, cum ar fi routerele ce suportă protocoale diferite de nivel rețea, pot învăța unele despre celelalte chiar dacă nu există conectivitate de nivel 3.

Fișierele de configurație IOS, cum ar fi startup-config sau running-config, ar trebui să fie arhivate. Aceste fișiere pot fi salvate într-un fișier text sau stocate pe un server TFTP. Unele modele de routere au de asemenea un port USB și un fișier ce poate fi încărcat pe un drive USB. Dacă este necesar, aceste fișiere pot fi copiate pe router și/sau switch de pe serverul TFTP sau driverul USB.

Utilizarea rețelei nu este limitată la întreprinderi mici și organizații mari. Un alt mediu ce profită din ce în ce mai mult de tehnologia de rețea este locuința. O rețea de domiciliu este similară cu o rețea de întreprindere mică. Însă, multe rețele de domiciliu (și multe rețele de întreprindere mică) nu necesită dispozitive de volum ridicat, cum ar fi routere dedicate sau switchurile. În schimb, multe rețele de domiciliu utilizează un singur dispozitiv multi-function. Pentru scopul acestui curs, dispozitivele multi-function vor fi referite ca routere integrate. Multe routere integrate oferă capabilități atât de conexiune wireless, cât și de switching cablat și servesc ca access point (AP) în rețeaua wireless. Pentru a permite conectivitate wireless, modul wireless, SSID, canalul RF și orice mecanism de securitate dorit trebuie să fie implementate.

BIBLIOGRAFIE

- [1] A. Tanenbaum, *Rețele de calculatoare (ediția a patra)*, Byblos, Tg.Mureș, 2003
- [2] R. Stevens, B. Fenner, A. Rudoff, *UNIX Network Programming* Volume 1, Third Edition: The Sockets Networking API, Addison Wesley, 2003
- [3] M. Popa, *Bazele modelării Rețelelor de Calculatoare*, Ed. Universității din București,, București, 2004.
- [4] T. Thomas, *Primi pași în Securitatea Rețelelor*, ciscopress.com, 2005.
- [5] S. Buraga, G. Ciobanu, *Atelier de programare în rețele de calculatoare*, Polirom, Iași, 2001: <http://www.infoiasi.ro/~lrc/>
- [6] D. Comer, *Internetworking With TCP/IP*, vol.I: Principles, Protocols, and Architecture (2nd edition), Prentice Hall, New Jersey, 1991.
- [7] D. Comer, D. Stevens, Internetworking with TCP/IP: vol.III: Client-Server Programming and Applications, Prentice Hall, New Jersey, 1993.
- [8] S. Androusellis-Theotokis, D. Spinellis, A Survey of Peer-to-Peer Content Distribution Technologies, ACM Computing Surveys, 36(4):335-371, December 2004
- [9] M. Mallick, Mobile and Wireless Design Essentials, John Wiley & Sons, 2003
- [10] S. Raab et al., Mobile IP Technology and Applications, Cisco Press, 2005
- [11] J. Doyle, CCIE Professional Development: Routing TCP/IP, Volume I, Macmillan Technical Publishing, 1998
- [12] K. Robbins, S. Robbins, UNIX Systems Programming: Communication, Concurrency, and Threads, Prentice Hall PTR, 2003
- [13] R. Stevens, TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley Longman, 1994 – sursele pot fi gasite la adresa <http://www.kohala.com/start/tcpipiv1.tar.Z>
- [14] R. Stevens, TCP/IP Illustrated, Volume 2: The Implementation, Addison-Wesley Longman, 1995
- [15] R. Stevens, TCP/IP Illustrated, Volume 3: TCP for Transaction, HTTP, NNTP, and the UNIX Domain Protocols, Addison-Wesley Longman, 1996
- [16] D. Acostăchioie, Programare C și C++ pentru Linux, Polirom, Iași, 2002: <http://www.biosfarm.ro/~dragos/prg>
- [17] D. Acostăchioie, Securitatea sistemelor Linux, Polirom, Iași, 2003: <http://www.biosfarm.ro/~dragos/sec>
- [18] J. Martin, J. Leben, TCP/IP Networking, Prentice Hall, New Jersey, 1994
- [19] A. Abbas, Grid Computing: A Practical Guide to Technology and Applications, Charles River Media, 2004
- [20] D. Acostăchioie, S. Buraga, Utilizare Linux, Polirom, Iași, 2004: <http://www.infoiasi.ro/~linux/>
- [21] D. Acostăchioie, Administrarea și conFig.rea sistemelor Linux (ediția a treia), Polirom, Iași, 2006: <http://www.biosfarm.ro/~dragos/admin>
- [22] M. Ben-Ari, Principles of Concurrent Programming, Prentice Hall International, 1982
- [23] S. Dixit, R. Prasad (eds.), Wireless IP and Building the Mobile Internet, Artech House, 2003
- [24] A. Grama et al., Introduction to Parallel Computing (2nd edition), Addison Wesley, 2003
- [25] E. Hall, Internet Core Protocols: The Definitive Guide, O'Reilly, 2000
- [26] G. Held, Ethernet Networks (4th edition), John Wiley & Sons, 2003
- [27] A. Jones, J. Ohlund, Network Programming for Microsoft Windows, Microsoft Press, 1999
- [28] A. Kshemkalyani, M. Singhal, Distributed Computing. Principles, Algorithms, and Systems, Cambridge University Press, 2008
- [29] C. McNab, Network Security Assessment, O'Reilly, 2004
- [30] D. Naik, Internet. Standards and Protocols, Microsoft Press, 1998
- [31] K. Robbins, S. Robbins, Unix Systems Programming: Communication, Concurrency, and Threads, Prentice Hall PTR, New Jersey, 2003
- [32] M. Rockkind, Advanced UNIX Programming, Prentice Hall, New Jersey, 1985
- [33] N. Shi (ed.), Mobile Commerce Applications, Idea Group Publishing, 2004

- [34] R. Stevens, Advanced UNIX Programming in the UNIX Environments, Addison-Wesley, Reading MA, 1992
- [35] A. Tanenbaum, Modern Operating Systems, Addison-Wesley, Reading MA, 2001
- [36] A. Wells, Grid Application Systems Design, CRC Press, 2008
- [37] * * *, Peer-to-Peer Application Development, Hungry Minds, 2002
- [38] Utilitare pentru shell (traducere de Adrian Haisan)
- [39] Utilitare pentru rețea (tutorial tradus de Diana Popovici)
- [40] TCP/IP și rețelele (tutorial tradus de Ionuț Lucaș)
- [41] Principii UNIX și Internet (HOWTO) (traducere de Florin Bandaș)
- [42] ConFig.rea TCP/IP (traducere de Luminița Moruz și Gabriel Manolache)
- [43] Specificația protocolului IP (traducere de Raluca Gordân)
- [44] Împărțirea în subrețele (traducere de Corina Rotaru)
- [45] Protocole de rutare (o prezentare de Ecaterina Valică și Raluca Moroșan)
- [46] Specificația protocolului ICMP (traducere de Valentin Cilibiu)
- [47] Specificația protocolului RIP (traducere de Leontina Munteanu și Andreea Tutoveanu)
- [48] Specificații formale pentru Exterior Gateway Protocol (traducere de Daniel Onacă)
- [49] Specificația protocolului TCP (traducere de Cătălin Bulancea)
- [50] Specificația protocolului TELNET (traducere de Raluca Motrescu)
- [51] Specificația protocolului FTP (traducere efectuată de Adrian Haisan și Cătălin Mihai Apostu)
- [52] Specificația protocolului TFTP (traducere de Raluca Lăcătușu)
- [53] Specificația protocolului POP3 (traducere de Adina Slușer)
- [54] Fire de execuție în Linux (tutorial tradus de Ana-Maria Cucu)
- [55] Introducere în firele de execuție POSIX (traducere de Bogdan Manolache)
- [56] Programarea rețea în Windows – Winsock FAQ (traducere de Cristian Baciu și Sabin Nemțisor)
- [57] Introducere în MySQL (traducere de Carmen Leonte)
- [58] NFS și NIS (traducere de Ana-Roxana Tubultoc)
- [59] Principles of system administration
- [60] Booting, Init, and Shutdown
- [61] Users and Logins
- [62] System Names and Access Permissions
- [63] Filesystems and Disks
- [64] Managing Processes
- [65] Managing Resources
- [66] TCP/IP and Networks
- [67] Configuring TCP/IP
- [68] TCP/IP Utilities