

Rețele de calculatoare

Host - orice dispozitiv parte din rețea ce poate primi/trimite date - fiecare host are o adresă unică

Hub - împarte conexiunea rețelei la mai multe calculatoare

Switch - grupează toate calculatoarele din rețea pentru a transmite datele către alt dispozitiv și e mai bun ca hub-ul deoarece duce datele de la sursă la destinație fără a le emite în rețea

Router - conectează LAN-ul la internet / conectează mai multe calculatoare la internet

Modem - conectează calculatorul la internet folosind o linie de telefonie. E un echipament intern pc-ului (pe placa de bază) folosit mai rar astăzi.

Utilizari :

- permite partajarea resurselor eliminând necesitatea prezentei fizice a echipamentului în locația unde se solicită datele
- se folosește în modelul client-server (serverul este un calculator central folosit pentru stocarea informațiilor și este administrat, iar clienții sunt calculatoarele ce solicită date de la sv)
- reprezintă un mediu de comunicare
- e-commerce

Avantaje

- back-up și roll-back de la server
- partajare software & hardware - se poate instala software direct pe server fără a mai fi nevoie să fie instalat pe fiecare mașină în parte. Idem hardware
- scalabilitate și fiabilitate (pe măsură ce rețeaua se extinde, viteza transferului de date scade și crește predispoziția apariției de erori, dar aceasta problema poate fi depășită cu switch-uri și routere

Arhitectura rețelei de calculatoare

a. Peer-to-peer

- toate calculatoarele sunt legate între ele și au privilegii și responsabilități egale (de preferat max 10 calculatoare)
- nu are server dedicat
- se alocă permisiuni fiecărui calculator pentru partajarea resurselor, dar pot apărea probleme dacă calculatorul cu resursa e picat

Avantaje

- costuri reduse
- dacă pică un pc celelalte nu sunt afectate
- usor de instalat

Dezavantaje

- Nu conține sistemul centralizat deci fără back-up
- Probleme de securitate întrucât fiecare calculator se

gestionează singur

b. Client-server

- creat pentru utilizatori (clienți) sa acceseze resurse (poze, cântece, videos)

- controller-ul central este cunoscut ca server
 - un server realizează toate operațiunile majore de la securitate la gestiunea rețelei

- server-ul este responsabil de gestiunea resurselor (fisiere, foldere, printere)

- dacă un calculator dorește sa comunice cu un altul atunci va trimite cererea către server, care va permite sau nu comunicarea

Avantaje

- o rețea client-server conține sistemul centralizat deci se pot recupera date
- contine un server dedicat care creste performanta sistemului
- o mai buna securitate deoarece resursele sunt gestionate de un singur server
- creste viteza de partajare a resurselor

Dezavantaje

- scump, server-ul are nevoie de memorie
- un sv are un NOS (Network Operating System) ca sa distribuie resursele către clienți
- necesita un admin de rețea

Componente

Componentele sunt părțile principale necesare pentru instalarea software-ului. NIC, switch, cablu, hub, router, modem.

În functie de tipul rețelei unele componente pot fi redundante

NIC - adresa MAC sau adresa fizica este unica și este pastrata în PROM (Programmable read-only memory)

HUB - dispozitiv fizic ce răspândește conexiunea între alte calculatoare.

Când un calculator solicita o informație de la rețea, acesta trimite solicitarea mai întâi către HUB care la rândul sau o distribuie la celalalte calculatoare. Toate vor verifica dacă solicitarea le apartine, în caz contrar

solicitarea este ignorata.

Tipuri de retele

- a. LAN
- b. MAN
- c. WAN
- d. PAN

e. InternetWork

- e data de conectarea a doua sau mai multe LAN sau MAN
- folosește protocolul internet
- modelul de referenda este Open System Interconnection (OSI)

Tipuri de InternetWork

1. Extranet

- rețea de comunicare bazată pe Transmission Control protocol și internet protocol.
- utilizatorii au nevoie de credentiale pentru acces
- nu poate avea doar un LAN, trebuie sa aibă măcar o conexiune

la rețeaua externa

2. Intranet

- rețea dedicata numai angajaților unei organizații

Avantaje intranet

- comunicare ieftină
- informatia e transmisa în timp real => eficienta
- arhitectura neutra deoarece un calculator se poate conecta cu un altul care are o alta arhitectura
- reduceri de cost

Topologia rețelei

Specifica structura rețelei anume cum sunt conectate componentele.
Exista doua tipuri : fizica și logica

a. Topologia fizica e reprezentarea geometrica a tuturor nodurilor dintr-o rețea

Exista 6 tipuri : Bus, Ring, Tree, Star, Mesh si Hibrida

1. Bus - toate stațiile sunt conectate la un cablu 'backbone'

- cand un nod vrea sa trimită un mesaj peste rețea acesta va trimite mesajul intregii rețele, toate stațiile primesc mesajul indiferent dacă le este adresat sau nu

- folosita în principal în rețele standard de tip

802.3(ethernet) și 802.4

- configurare simpla
- cea mai întâlnită metoda de acces pentru topologia bus e

CSMA (Carrier Sense Multiple Access)

CSMA - control de acces media folosit pentru a gestiona fluxul datelor astfel încât integritatea lor sa se păstreze (sa nu se piardă pachete)

i. CSMA CD (collision detection) metoda de acces Folosita pentru detectarea coliziunilor. Lucrează pe 'recuperare după Coliziune' deoarece odată ce se detectează coliziunea emițătorul va înceta transferul de date.

ii. CSMA CA (collision avoidance) - verifica daca mediul de transmisie este ocupat sau nu, dacă este ocupat emițătorul va aștepta pana când mediul se eliberează

Avantaje Bus

- cost redus pentru cabluri (toate sunt interconectate)
- viteze moderate prin cabluri coaxiale și împletite ce suporta pana la 10Mbps
- tehnologie prea bine cunoscuta cu dispozitive fizice ieftine
- limitarea dezastrului - nefuncționarea unui nod nu le impacteaza pe celelalte

Dezavantaje Bus

- prea mult cablu
- dacă se taie cablul (necesita echipament pentru depistare) pică toată rețeaua
- interferenta de semnal în situația în care doua noduri transmit mesaje simultan
- reconfigurare dificila - adăugarea de noi dispozitive încetinește rețeaua
- atenuarea semnalului - fix : sunt folosite relee (repeaters)

2. Topologie inel

- ca bus, dar cu capetele conectate
- un nod primește mesajul de la nodul precedent și trimite către următorul
- datele 'Curg' într-o singura direcție - clockwise / în forma unei bucle continue
- nu exista un capăt - fiecare nod e conectat cu altul
- cea mai întâlnită metoda de acces pentru inel este pasarea de token-uri de la un nod la altul. Token - cadru ce circula în rețea

Pasarea token-urilor: circula în rețea și e pasat de la un calculator la altul pana ajunge la destinație

- Expeditorul modifica token-ul prin alipirea adresei la datele mesajului
- Când adresa dispozitivului corespunde cu cea din token, dispozitivul destinatar va trimite o înștiințare către dispozitivul expeditor

3. Star

- Dispoziția nodurilor în forma stea se face prin interconectarea lor la un hub central
- Calculatorul central este cunoscut ca server, restul nodurilor clienți

Dezavantaje: nefuncționarea serverului

4. Copac = bus + star

5. Mesh = patrat cu diagonale (> 3 legături pe nod)

- se realizează după formula $NrCabluri = (n*(n-1))/2$

6. Hybrid = combinație din toate

Moduri de transmisie (mod de comunicare / direcționare)

- Fiecare canal de comunicare are o direcție asociată dată de modul de transmisie care se regăsește în stratul fizic al rețelei.

A. Simplex mode

- Unidirecțional, un nod poate doar trimite sau primi de la altul

- ex: stație radio - primește dar nu transmite, tastatura primește, monitorul emite

- folosit în organizații unde este nevoie doar să se trimită date

B. Half-duplex mode

- transmisiune în ambele direcții dar nu simultan

- toată banda este folosită când se transmite într-o anumită direcție

- ex : walkie-talkie - o persoană vorbește și una ascultă

- dezavantaj : cât timp un nod emite celălalt trebuie să aștepte până termină

C. Full-duplex mode

- mod de comunicare bidirecțional = ambele stații pot transmite și recepționa simultan

- construit din două simplex pentru fiecare direcție

- rapid, ex: rețea de telefonie mobilă - ambele persoane pot vorbi și asculta în același timp

Modele de rețea de calculatoare

Arhitectura stratificată

- dezvoltat de ISO pentru a împărți design-ul în părți mai mici

- fiecare strat de mai jos deservește stratul superior

- asigură independența între straturi punând serviciile din stratul inferior la dispoziția stratului superior fără a specifica cum sunt acestea implementate

- numărul de straturi, funcții, conținut al fiecărui strat diferă de la o rețea la alta

- serviciul este setul de acțiuni pe care un strat inferior îl oferă către stratul superior

- protocolul reprezintă un set de reguli despre conținut și ordinea mesajelor pe care un strat îl folosește pentru a schimba informații cu o entitate pereche

- interfața - modul în care se transfera informații
- în arhitectura stratificată *ex: stratul N al unei rețele va comunica cu stratul N al altei rețele, dar ele nu comunică direct ci pasează datele către stratul inferior până se ajunge la cel mai de jos strat
- sub stratul 1 se afla mediul fizic unde are loc comunicarea efectivă
- În arhitectura stratificată cerințele greu de gestionat sunt împărțite în părți mai mici (divide et impera)
- arhitectura = protocoale + straturi
- Avantaje: abordarea divide et impera, modularitatea straturilor, ușor de modificat și de testat

Modelul OSI (Open System Interconnection - ISO 1984)

- model de referință care descrie cum informația dintr-un software de pe un computer traversează mediul fizic pentru a ajunge într-un soft de pe alt pc
- 7 straturi fiecare cu funcția sa de rețea (task-urile sunt împărțite în 7 subtask-uri pt fiecare strat gestionate independent)
- Responsabilitatea host-ului = [Aplicație, Prezentare, Sesiune, Transport] = straturi superioare, implementate în software - folosite atât de utilizator cât și de aplicație
- Responsabilitatea rețelei = [Rețea, Data Link, Fizic] = straturi inferioare folosite în transportul datelor. Straturile Data Link și cel Fizic sunt implementate atât în soft cât și în echipamentul hardware. Stratul fizic se ocupa de plasarea datelor în mediul fizic de conexiune

1. Strat Aplicație - oferă serviciile utilizatorului

Utilizat ca o fereastră pentru ca utilizatorii și serviciile aplicațiilor să acceseze serviciile de rețea

Gestionează probleme ca transparență rețelei, alocarea resurselor șamd

Un strat de aplicație nu e o aplicație de sine stătătoare, dar realizează funcțiile uneia

Pune serviciile rețelei la dispoziția utilizatorului.

Funcții:

- *transfer fișiere, acces și management (FTAM) : stratul de aplicație permite unui utilizator să acceseze, să descarce și să gestioneze fișiere de pe alt computer aflat la distanță*
- *serviciu de corespondență (mail service) oferă facilitățile de trimitere și stocare a mail-urilor*
- *servicii de directoare*

2. Strat Prezentare - traduceri, compresii, criptare

-responsabil de sintaxa și semantica informațiilor transmise între două

sisteme

- funcționează ca traducător pentru o rețea
- e parte a sistemului de operare care convertește date de la un format de prezentare la altul
- cunoscut de asemenea ca strat de sintaxa

Funcții:

- *Traducere de la format-expeditor la format comun, apoi la format-destinatar*
- *Criptare - proces de amestecare a datelor folosind algoritmi matematici și chei*
- *Compresie - reduce numărul de biti transmise pe rețea: important la text, audio, video*

3. **Strat Sesiune** - realizează, menține și încheie sesiuni și sincronizează comunicarea dintre dispozitive

Funcții:

Controlul Dialogului - Creează un dialog între procese - poate fi half-duplex sau full-duplex

Stratul de sesiune Adaugă niște checkpoint-uri când transmite date într-o secvență. Dacă apare vreo eroare în mijlocul transmisiunii atunci aceasta se va relua de la checkpoint. Acest proces poartă numele de Sincronizare și recuperare

4. **Strat Transport** - corespondența mesaje între procese

- se asigură ca mesajele sunt transmise în ordinea în care au fost trimise și ca datele nu au fost duplicate
- responsabilitate principală este transferul complet al datelor
- primește 'bucăți' de date de la stratul sesiune și le convertește în unități mai mici numite segmente

2 Protocoale folosite:

i. Protocolul Controlului de Transmisie (TCP)

- *protocol standard ce menține conexiunea între gazde*
- *când se transmit date prin protocolul TCP, atunci protocolul împarte datele în segmente. Fiecare segment traversează internetul folosind rute multiple și ajung la destinație în ordine diferită. La destinație TCP le rearanjează în ordinea inițială*

ii. User Datagram Protocol (UDP) - un protocol neconsiderat fiabil deoarece la primirea pachetului receptorul nu emite o înștiințare iar expeditorul nu așteaptă după una

Funcții:

- *Adresarea Service-point: calculatoarele rulează mai multe programe simultan, de aceea transmiterea datelor nu se face doar de la un calculator la altul ci și de la un proces la altul. Stratul de transport Adaugă un header cu adresa numită adresa de punct-serviciu sau*

port-adresă. Responsabilitatea stratului rețea este să transmită datele de la un calculator la altul în timp ce responsabilitatea stratului de transport este transmiterea datelor de la un proces la altul

- Segmentare și reasamblarea : la segmentare fiecare segment

primește un număr pe baza căruia se va reconstitui mesajul

- Controlul fluxului și al erorilor

- Controlul conexiunii - oferă 2 servicii :

1. Connectionless - tratează fiecare segment ca un pachet individual și fiecare călătorește pe alta rută către destinație

2. Connection-oriented - deschide mai întâi o conexiune cu stratul de transport al stației destinație și toate pachetele călătoresc pe aceasta

5. **Strat Retea** - muta pachete de la sursă la destinație

- gestionează adresarea dispozitivelor, urmărind locația acestora

-determină cea mai bună cale de transfer a datelor / data-link fiind responsabil de rutare a pachetelor

-routerele sunt dispozitive specificate în acest strat

-protocoalele folosite pentru rutarea traficului de internet sunt cunoscute ca protocoale ale stratului de rețea : IP, IPv6

Funcții:

- internetworking - responsabilitate principală - oferă o conexiune logică între dispozitive

- adresare - Folosita pentru identificarea dispozitivului pe internet

- rutare - oferă cea mai bună cale

- pachetizare - stratul rețea primește 'Segmente' de date de la stratul transport și le împachetează mai departe corespunzător

6. **Strat Data Link** - transfer lipsit de erori al cadrelor de date; definește

formatul datelor transmise în rețea; responsabil de identificarea unică

a fiecărui dispozitiv dintr-o rețea locală; conține 2 substraturi : **Strat**

de control al legăturii logice - responsabil de transferul pachetelor

către stratul Rețea al receptorului; identifică adresa din antet și

Stratul de control al accesului media - transferă pachetele în rețea

Funcții:

Încadrare - transpune bitii din fluxul fizic neprelucrat în pachete

numite Cadre; Adaugă un Header și un Trailer pachetului. Header-ul conține adresele sursă și destinație;

Adresare fizică

Control de flux - cea mai importantă - e tehnica prin care rata

constantă de transfer a datelor se menține la ambele capete astfel

încât datele să nu fie corupte. Se asigură ca stația de transmisie cu o

putere de procesare mai mare nu depășește capacitățile de

procesare ale stației de Recepție

Control de erori - Adaugă o valoare calculată la Trailer numită CRC (Cyclic Redundancy Check) care se Adaugă la cadrul pachetului înainte de transmiterea către stratul fizic. Dacă apare vreo eroare, receptorul trimite înștiințarea de retransmisie a pachetelor corupte

Control de acces - când doua sau mai multe calculatoare sunt conectate la același canal de comunicare, protocoalele stratului data link determina ce calculator deține controlul legăturii la un moment dat

7. **Strat Fizic** - oferă mediul fizic prin care circula bitii; Principala funcție este de transfer de biți între noduri; Realizează/menține/încheie conexiunea fizică; oferă specificațiile de rețea mecanice, electrice și procedurale

Funcții:

Configurarea liniei (definește cum se conectează fizic doua sau mai multe dispozitive);

Transmisia de date (modul de transmisie simplex/half-duplex/ duplex) ;

Topologia - aranjamentul;

Semnalul - tipul de semnal folosit pentru transmiterea informațiilor

Modelul TCP/IP

- dezvoltat înaintea OSI și este destul de diferit fata de acesta
- 5 straturi : Aplicație, Transport, Rețea, Data-Link și Fizic
- primele 4 straturi oferă standarde fizice, interfața de rețea, internetworking și funcții de transport care corespund primelor 4 straturi din modelul OSI și acestea 4 sunt reprezentate în modelul TCP/IP ca un singur strat - cel de aplicație
- este un protocol ierarhic alcătuit din module interactive fiecare oferind funcționalități specifice

A. Stratul de acces la rețea (Network Access Layer)

- cel mai de jos din modelul TCP/IP
- combinație între stratul fizic și cel Data-Link regăsite în modelul OSI
- definește cum ar trebui să fie transmise fizic datele prin rețea
- responsabil în principal de transmiterea datelor între doua dispozitive din aceeași rețea
- funcțiile îndeplinite de acest strat includ încapsularea datagramelor IP în cadre(pachete) transmise prin rețea și maparea adreselor IP cu adresele fizice
- protocoalele folosite de acest strat sunt : ethernet, token ring, FDD, x.25 și frame relay

Stratul Internet

- al doilea strat al modelului TCP/IP, cunoscut și ca strat de rețea
- trimite pachete din orice rețea, acestea ajungând la destinație indiferent de ruta parcursă
- protocoalele folosite :

Protocolul IP : - cea mai importanta parte a modelului

- se ocupa de adresare IP, aceste adrese sunt folosite de straturile superioare pentru a identifica dispozitivul și pentru a oferi routare internetwork
- se ocupa de comunicarea gazda-gazda - determina calea pe care datele vor fi transmise
- încapsularea și formatarea : protocolul ip accepta date de la stratul de transport, încapsuleaza datele într-un mesaj numit și IP datagram și se asigura ca acestea ajung la destinație în siguranță
- fragmentare și reconstrucție : limita impusă asupra dimensiunii unei datagrame IP de protocolul stratului data-link se numește Maximum Transmission Unit (MTU). Dacă dimensiunea datagramei IP depășește aceasta limita atunci protocolul IP va împărți datagrama în unități mai mici astfel încât acestea sa poată călători prin rețeaua locală. Fragmentarea se poate face fie de expeditor sau de un router intermediar. La destinație fragmentele sunt asamblate din nou pentru a reconstitui mesajul original.
- routare : când expeditorul și destinatarul se afla în aceeași rețea locală LAN, MAN, WAN ne referim la o livrare directă a datagramelor IP. Când cele două stații sunt la mare depărtare ne referim la livrare indirectă care se poate realiza prin routare datagramei IP prin diferite alte dispozitive ca routerele.

Protocolul ARP (Address Resolution Protocol) - scoate adresa fizică din IP :

- cei doi termeni sunt asociați în principal cu protocolul ARP:

ARP request - când expeditorul dorește sa știe adresa fizică a dispozitivului acesta emite în rețea o solicitare ARP

ARP reply - toate dispozitivele atașate la rețea vor accepta aceasta solicitare dar numai destinatarul va recunoaște adresa IP și va trimite adresa fizică în forma unui răspuns ARP. Destinatarul va Adaugă adresa fizică în memoria sa cache și în header-ul datagramei.

Protocolul ICMP (Internet Control Message Protocol)

- mecanism folosit de gazde și routere pentru a trimite notificări despre problemele legate de datagrame înapoi către expeditor

- datagramele călătoresc din router în router până ajung la destinație. În cazul în care routerul nu poate direcționa datele din cauza unor condiții speciale cum ar fi nefuncționarea conexiunilor (router în flăcări / congestie de rețea) atunci se folosește protocolul ICMP pentru înștiințarea expeditorului ca datagrama nu poate fi livrată

- folosește cei doi termeni :

- ICMP Test - folosit pentru a determina dacă destinația este accesibilă

- ICMP Reply - folosit pentru a determina dacă dispozitivul destinație Răspunde sau nu

- responsabilitatea de bază este notificarea problemelor, nu rezolvarea lor, aceasta rămâne în grija expeditorului

- ICMP poate transmite mesaje doar spre destinație nu și către anumite routere intermediare deoarece datagramele conțin doar adresele expeditorului și destinatarului.

B. Stratul de transport

- se ocupa de fiabilitatea, controlul fluxului și corecția datelor trimise prin rețea

- 2 protocoale:

User Datagram Protocol (UDP)

- oferă un serviciu care nu necesită conexiune și livrarea transmisiei de la un capăt la altul

- nu este fiabil întrucât descoperă erorile dar nu le specifică

- UDP descoperă eroarea, iar ICMP Raportează la expeditor faptul ca datagrama a fost alterată

- continue următoarele câmpuri (în header = 8 octeți [fiecare câmp are 16 biți]) :

- i. Adresa portului sursă - este adresa programului aplicație care a creat mesajul

- ii. Adresa portului destinație - adresa programului aplicație care primește mesajul

- iii. Lungimea totală a datagramei exprimată în octeți

- iv. Checksum - câmp pe 16 biți folosit pentru detectarea erorilor

(UDP nu specifică exact care pachet a fost pierdut, conține doar checksum, nu și un ID pentru segmentul de date)

Transmission Control Protocol (TCP)

- oferă aplicațiilor servicii de transport

- Creează un circuit virtual între un expeditor și destinatar și e activ pe întreaga durată a transmisiunii

- e fiabil deoarece detectează erorile și retrimite pachetele alterate. Se asigură ca toate segmentele au fost recepționate înainte să

considere transmisiunea completa și să invalideze circuitul virtual.

- împarte la expediere mesajul în segmente și le atribuie un număr de ordine pentru reconstituire la Recepție

C. Stratul aplicație

- cel mai înalt nivel din modelul TCP/IP
- responsabil de gestiunea protocoalelor de nivel înalt și a probleme de reprezentare
- permite utilizatorului interacțiunea cu aplicația
- când acest strat al unei aplicații dorește să comunice cu stratul aplicației al altei stații el va trimite datele către stratul de transport
- nu toate aplicațiile fac parte din stratul aplicație ci doar cele care interacționează în sistemul de comunicare: ex text editor nu are treaba în timp ce browserul da deoarece folosește protocolul de aplicație HTTP

- protocoalele folosite în stratul aplicație :

1. HTTP - hypertext transfer protocol - permite accesul la date prin World Wide Web. Transferă datele în forma de text, audio, video; este cunoscut ca hiper deoarece dispune de eficiența de a lucra într-un mediu în care se sare rapid de la un document la altul

2. SNMP - simple network management protocol - framework de gestiune a dispozitivelor pe internet folosind suita TCP/IP

3. SMTP - simple mail transfer protocol - protocol de susținere a emailurilor - e folosit pentru a transmite date la o adresă de email

4. DNS - domain name system - sistemul de poreclire a adreselor IP

5. TELNET - terminal network - realizează conexiunea între calc local și cel aflat la distanță astfel încât terminalul local apare ca terminal al calculatorului de la distanță

6. FTP - file transfer protocol - transferă fișiere de la local la distanță