

Curs 11

1. Retea

- Alcatuit din noduri (calculatoare) numite host-uri
- Legaturi fizice: cabluri de retea, canale wireless
- Legaturi logice: calea prin intermediul legaturilor fizice dintr-un nod sursa si unul destinatie
- O retea poate fi conectata la una sau mai multe retele

2. Protocol de comunicatie

- Modelul OSI (Open Systems Interconnection)
 - 7 Nivele: fizic, data link, retea, transport, sesiune, prezentare, aplicatie
- Model alternativ, protocoalele internet
 - Combina utlimele trei nivele OSI (sesiune, prezentare, aplicatie) intr-un singur nivel: *_aplicatie_*
 - Combina nivelul fizic si data link intr-un singur nivel: *_link_*
- Functioneaza pe principiul stivei de protocoale:
 - Protocolul de nivel cel mai inalt apeleaza la serviciile protocolului imediat anterior
 - La baza, datele trimise prin protocoalele *_data link_* sunt trimise prin retea fizica
- Exemple:
 - Data link (retele LAN): Ethernet, Wireless
 - Retea: IP
 - Transport de date: TCP, UDP, SCTP
 - Sesiune: RPC
 - Prezentarea datelor: criptare, compresie, XDR, ASN.1
 - Aplicatie: HTTP, DNS, FTP, SSH
- Socket: abstractizeaza comunicatia la nivel de aplicatie

3. Internet Protocol (IP)

- Fiecare host este identificat prin una sau mai multe adrese IP
- Versiuni: IPv4 (32 biti) si IPv6 (128 biti)
- Anumite adrese sunt folosite pentru adresare privata
 - Ex: 192.168.1.1, 127.0.0.1

4. TCP (Transport Control Protocol)

- Impacheteaza/dezpacketeaza mesaje in pachete
- Pe scurt, este cel mai safe protocol:
- Asigura livrarea pachetelor in ordine la destinatie
- Rezolva automat probleme de rutare
- Asigura retransmisia automata a pachetelor pierdute/defecte

5. DNS (Domain Name System)

- Inlocuieste IP-uri cu nume usor de tinut minte
- Agenda centralizata de perechi (nume, ip)
- Ex: fmi.unibuc.ro -> 193.226.51.6

6. Socket

- Abstractie de nivel SO
- Defineste un endpoint format din:
 - Adresa: adresa IP a unui host
 - Port: intrarea a host-ului
 - Port server: identifica un serviciu
 - Port client: identifica procesul client
- O conexiune se face intre 2 socketuri/endpointuri folosind un protocol comun
- Ex: <TCP, 192.168.1.6:4444, 193.226.51.6:80>

- TCP Sockets:
 - Beneficiaza de safety-ul protocolului [[#TCP (Transport Control Protocol)|TCP]]
 - Din el se citesc octeti, nu mesaje intregi
- UDP Sockets:
 - Canal *_fara garantii_* (fara safety)
 - Mesajele se pot pierde
 - Mesajele se citesc in intregime

7. World Wide Web (www)

- Foloseste HyperText Transfer Protocol (HTTP) si portul 80
- Resursele sunt identificate prin URL
- Acces criptat prin HTTPS pe portul 443
- Exemplu comincatie HTTP:

...

GET/HTTP/1.1

Host: <host>

...

8. Email - IMAP, POP

- Acces la posta
- Post Office Protocol (POP)
 - Face o copie locala a mesajelor si le sterge de pe serverul de mail
- Internet Message Access Protocol (IMAP)
 - Opereaza asupra mesajelor direct pe server
 - Poate tine o copie (partiala) local

9. Email - SMTP

- Expediere mesaje
- Simple Message Transfer Protocol (SMTP)
 - Trimite un mesaj catre un destinatar
 - Este un protocol vechi, nu cere expeditor si nici autentificare care duce la spam si spoofing
- Solutii anti-spam:
 - Reverse DNS: legatura IP -> nume
 - Autentificare: utilizator si parola; criptare
 - Anti-email spoofing (SPF, DKIM, DMARC)
 - Baze de date cu IP-uri de spammeri

10. FTP (File Transfer Protocol)

- Transfer de fisiere
- Autentificare si criptare FTPS

11. Peer-to-peer

- Transfer de fisiere distribuit fara server
- Toti clientii transmit si primesc date
- Mod eficient de transfer pentru date mari
- Ex: Torrent

12. SSH (Secure SHell)

- Protocol sigur de conectare la alt calculator
- Folosit peste tot in programare si administrare
- Autentificare parola, cheie asimetrica, etc