2.3. Comaparație intre modelul ISO-OSI si TCP/IP

Modelul OSI şi modelul TCP/IP sunt ambele modele de referinţă folosite pentru a descrie procesul de transmitere a datelor.

Deși atât ISO-OSI cât și TCP încearcă să definească/modeleze același lucru, și anume procesul de comunicare între două unități aflate la distanță se pune firesc întrebarea: care din ele este mai bun? Se va face o analiză privind asemănările și deosebirile care leagă cele două modele.

O importantă asemănare între ISO-OSI și TCP/IP o constituie **faptul că ambele sunt modele conceptuale ale procesului de comunicare**. Din păcate această asemănare simplă și evidentă conține și o primă deosebire fundamentală: ISO-OSI **este general, permițând explicarea oricărui proces de comunicare**, în timp ce TCP/IP-ul **nu reuşeşte** să modeleze perfect decât procesul de comunicare folosit în Internet.

În figura 2.3. se prezintă comparația intre cele două modele teoretice.

*Fig. 2.3. Comparație intre modelul ISO-OSI SI modelul TCP / IP*

Aplicație

Transport

 Protocoale

Acces la mediu

Rețea

 Rețele

 Niveluri flux

 de date

Aplicație

Prezentare

Sesiune

Transport

Rețea

Legătura de date

Fizic

 Niveluri

 aplicație

Noi ca utilizatori finali folosim în mod curent numai cu nivelul aplicație, însă dacă dorim de dezvoltăm o rețea trebuie să avem o cunoaștere detaliată a nivelurilor.

Este adevărat că majoritatea utilizatorilor nu știu mai nimic despre protocoale de rutare sau alte detalii.

De ce nu se poate studia doar un model care este considerat cel mai bun?

* Modelul OSI este folosit pentru dezvoltarea standardelor de comunicație pentru echipamente și aplicații ale diferiților producători. Unii specialiști în domeniu îl preferă pentru analize mai atente dar și ca fundament în orice discuție legată de rețele de calculatoare.
* Pe de altă parte este adevărat că TCP/IP este folosit pentru suita de protocoale TCP/IP și este mai folositor pentru că este implementat în lumea reală.

*Asemănări*

* Ambele au nivelul aplicație, deși fiecare conține servicii diferite;
* Ambele au nivelurile rețelei și transportului comparabile
* Ambele folosesc tehnologia de tip packet switching

Administratorii de rețea trebuie să le cunoască pe amândouă pentru a putea lua cea mai bună decizie cu privire la modelul ce îl va implementa.

*Deosebiri*

* TCP/IP combină în nivelul său Aplicație nivelurile Aplicație, Prezentare şi Sesiune din modelul OSI.
* TCP/IP combină nivelul Legătură de date şi nivelul Fizic din modelul OSI într-un singur nivel numit Acces Reţea.
* TCP/IP pare a fi mai simplu deoarece are mai puține niveluri. Protocoalele TCP/IP reprezintă standardele pe baza cărora s-a dezvoltat Internetul.
* Rețelele tipice nu sunt construite pe baza protocoalelor OSI, deși modelul OSI este considerat ca ghid.
* TCP / IP folosește protocolul UDP care nu garantează întotdeauna livrarea de pachete precum face nivelul transport din modelul OSI.

Este indicat a se folosi ambele modele atât OSI, dar și protocoalele TCP/IP.

Trebuie să vedem protocolul TCP ca pe un protocol al nivelului Transport din modelul OSI, IP ca pe un protocol al nivelului Rețea din modelul OSI, și Ethernet ca o tehnologie a nivelelor Legătură de date și Fizic din modelul OSI.

***Concluzii:***

Avantajele oferite de împărțirea rețelelor în niveluri sunt:

* Standardizarea componentelor rețelelor, permițând astfel crearea acestora de către diverși producători.
* Permiterea comunicării între tipuri diferite de componente software și hardware dezvoltate de diverși producători.
* Previne ca schimbările apărute într-un nivel să nu afecteze celelalte niveluri, permițând astfel dezvoltarea rapidă a acestora.
* Fenomenul de comunicare în rețea este descompus în părți mai mici și implicit mai simple. Comunicarea prin rețea devine mai puțin complexă, înțelegerea și învățarea modului în care informația este trimisă și primită devenind mai ușor de făcut.

Studierea acestor niveluri permite să înțelegem cum circulă pachetele de date de la o rețea la alta și ce echipamente operează în fiecare nivel în momentul când informația circulă prin el. Astfel managementul problemelor care pot apărea în cursul fluxului pachetului de date se poate face mai ușor.

**Exemplu de transfer de date de la un calculator sursa la unul destinație**

Cel mai bun exemplu este modul in care putem citi o pagina web aflata pe un calculator situat la mare distanta:

1. utilizatorul lansează un program pentru vizualizarea paginilor web (browser) – care este asociat nivelului aplicație;
2. browser-ul este entitatea aplicație care va "negocia" pentru noi obținerea paginii de la calculatorul sursă;
3. nivelul aplicație va identifica existenta resursei cerute de client (clientul este browser-ul, care îl reprezintă pe utilizator în această "tranzacție") și a posesorului acesteia (serverul - înțeles ca fiind entitatea ce oferă resursa ceruta nu calculatorul central al unei rețele; în cazul nostru avem de-a face cu un server de web). Se realizează autentificarea serverului (se verifica daca partenerul este într-adevăr cine pretinde ca este) li se stabilește daca acesta este disponibil;
4. nivelul sesiune va stabili o conexiune între procesul client și procesul server;
5. nivelul transport se va ocupa de întreținerea conexiunii și de corectarea erorilor netratate la nivelul rețea;
6. nivelul rețea va asigura transferul datelor în secvențe (pachete), stabilind drumul acestora între server și client.

Teoretic conform celor descrise mai sus lucrurile se ar trebui să se desfășoare foarte simplu. Însă în realitate lucrurile se desfășoară ceva mai complicat ca în figura 2.4. Datele sosesc prin intermediul mediului de comunicație ca un flux de biți. La nivelul legăturii de date, biții sunt transformați în cadre, iar la nivelul rețea în pachete. În cele din urmă, datele ajung la nivelul aplicație unde sunt preluate de browser și ne sunt prezentate. Fiecare nivel adaugă sau șterge o parte din informațiile de control atașate datelor de celelalte nivele.

**Construirea datelor.** Utilizatorul scrie mesajul ce trebuie trimis care va fi procesat în straturile superioare pentru a avea un format care să poată fi trimis în rețea .

**Segmentare datelor.** Se face la nivelul transport (nivelul 4). În felul acesta se garantează ca datele vor ajunge în siguranța de la un calculator la altul.

**Adaugarea adreselor de retea**. La acest nivel informației i se adăugă un “header”. Pentru ca mesajul să ajungă la destinație acest „header” trebuie să conțină informații esențiale precum:

* adresa logica 3 catre care va fi expediat pachetul;
* adresa logica sursei;
* infomații despre următoarea mașină căreia i se va livra pachetul (next hop)

**Adugarea headerului de nivel “Legăturii de date”.** La acest nivel se va adăuga un header care conține informații cu privire la următoarea mașină care va primi acea informație. Acest pachet de date poartă denumirea de “frame”. Sunt două tipuri de transmisii de frame-uri:

* spre exemplu dacă sunt într-o rețea A și trimit informația în aceeași rețea IP-ul destinației va fi al mașinii către care trimit, MAC-ul de asemeni
* dacă se trimit date într-o altă rețea IP-ul va fi al destinației iar MAC-ul va fi al default gateway-ul din rețeaua A în care se află expeditorul.

**Convertirea frame-ului intr-o secventa de biti ( 0 si 1).** Așa circulă informația în mediul de propagare. La acest nivel este un sistem de sincronizare a celor două mașini

Același parcurs îl are informația odată ce a atins destinația dar în sens contrar de la stratul 1 pana la 7. In sensul asta trebuie precizat ca fiecare strat comunica cu echivalentul sau din mașina cu care s-a stabilit o conexiune. Acest tip de comunicare se numește comunicare peer-to peer 1 și implică folosirea unor PDU-uri ( prodocol data units). Pentru nivelul 4 PDU-ul este segmentul, pentru nivelul 3 packet-ul iar pentru nivelul 2 frame-ul, (ca în figura 2.5).

*Fig. 2.4. Modul în care sunt transmise datele de la calculatorul
sursă la calculatorul destinație*

Aplicație

Prezentare

Sesiune

Transport

Rețea

Legătura de date

Fizic

Date

Date

Date

**Date**

**Date**

**Date**

**Antet de rețea**

**Datee**

**Antet de rețea**

**Date**

**Antet Frame**

**Final Frame**

100110000001111101011

Calculator sursa

Calculator destinație

Aplicație

Prezentare

Sesiune

Transport

Rețea

Legătura de date

Fizic

Sensul în care circulă informația

Sensul în care circulă informația

Sensul în care circulă informația

În figura 2.5. este prezentată “discuția virtuală” dintre sursă și destinație în care nivelul de la calculatorul sursă discută cu același nivel de la calculatorul destinație.

*Fig. 2.5. Comunicația peer – to -peer (de la egal la egal)*

Aplicație

Prezentare

Sesiune

Transport

Rețea

Legătura de date

Fizic

Calculator sursa

Calculator destinație

Aplicație

Prezentare

Sesiune

Transport

Rețea

Legătura de date

Fizic

Segmente de date

Pachete de date

Frame-uri

Biți

Date

Date

Date