

## Capitolul 2: Concepte introductive de switching



## Generații de Ethernet

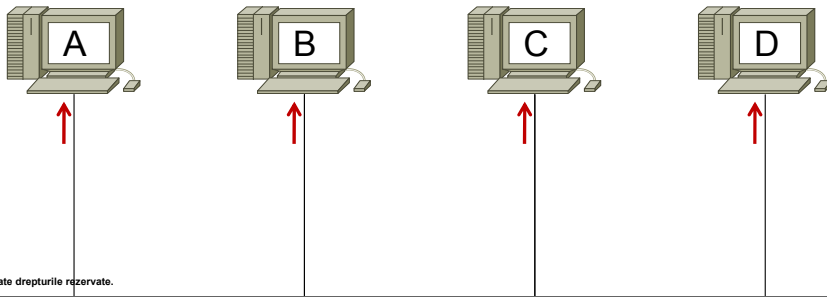
		Bandwidth	CSMA/CD	Duplex	Tip cablu
Ethernet	~1980	10Mb/s	Da	Half/Full	Coaxial, torsadat, fibră
Fast Ethernet	1995	100Mb/s	Da	Half/Full	Torsadat, fibră
Gigabit Ethernet	1999	1Gb/s	Da	Half/Full	Torsadat, fibră
10 Gigabit Ethernet	2002	10Gb/s	Nu	Full	Torsadat, fibră
40/100 Gigabit Ethernet	2010	40/100Gb/s	Nu	Full	Torsadat, fibră

Ethernet-ul a fost proiectat în anii 1970 la centrul de cercetare PARC® (Palo Alto Research Center), la momentul respectiv divizie a companiei XEROX®, folosind ca sursă de inspirație sistemul ALOHAnet al universității din Hawaii. Promovarea acestuia ca standard a început în 1979, când DEC® (Digital Equipment Corporation), Intel® și XEROX® au colaborat în acest sens. Primul standard, numit "DIX" (de la Digital/Intel/Xerox) a fost publicat în 1980. Acesta specifica o viteză de 10 Mb/s și adrese sursă și destinație pe 48 de biți. Standardul oficial de Ethernet (IEEE 802.3) a fost lansat în 1983.

De atunci, dezvoltarea și îmbunătățirea acestui standard au continuat, urmărind obținerea de viteze din ce în ce mai mari și a unei eficiențe cât mai ridicate a transmisiei datelor, ultima versiune specificând viteze de 40 și 100 Gb/s.

# CSMA/CD

- Carrier Sense

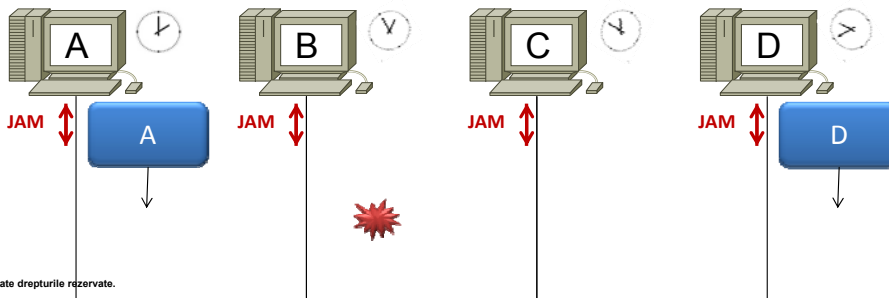


CSMA/CD este un protocol de control a accesului la mediul de comunicație, implementat la nivelul Access la rețea (stiva TCP/IP) / în subnivelul MAC al nivelului Legătură de date (stiva OSI), care ajută la împărțirea lărgimii de bandă între echipamente, reducând probabilitatea ca două dintre acestea să transmită simultan și este folosit doar în cazul comunicațiilor half-duplex. Această variantă a protocolului nu garantează eliminarea totală a coliziunilor.

Carrier Sense: Toate echipamentele care doresc să transmită trebuie mai întâi să verifice dacă mediul este liber. Dacă nu este detectat nici un semnal, rezultă că nici un alt echipament nu comunică în acel moment și pot începe să transmită. Altfel, se va aștepta o anumită perioadă, după care se va relua acest proces. După finalizarea transmisiei se va reveni la starea de verificare a mediului. Această "ascultare" are loc și în timpul transmisiei pentru a detecta eventualele semnale simultane de la alte stații (coliziuni).

## CSMA/CD

- **Multiple Access**
- **Collision Detection**
  - jam signal
  - random backoff

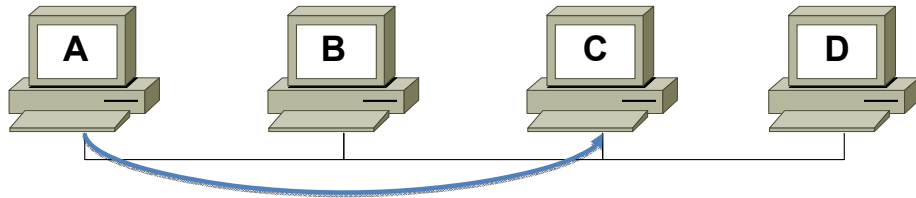


**Multiple Access:** Mai multe echipamente comunică peste același mediu.

**Collision Detection:** Coliziunile sunt detectate datorită creșterii amplitudinii semnalului la producerea lor, moment în care echipamentele implicate în coliziune (care transmiteau) emit un semnal de bruijaj (Jam Signal). Astfel, toate echipamentele conectate la mediu sunt informate de producerea unei coliziuni și pornesc un algoritm de backoff. Acesta presupune oprirea transmisiei pentru o durată aleatoare (Random Backoff), care va fi mai lungă pentru echipamentele care au produs coliziunea. După expirarea acestei perioade, fiecare echipament revine în starea de "ascultare a mediului".

## Comunicația Unicast

- Un singur receptor

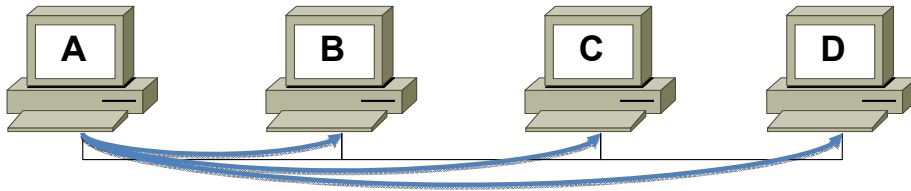


Comunicația la nivelul 2 (stiva OSI) / 1 (stiva TCP/IP) poate avea loc în 3 moduri diferite: unicast, broadcast și multicast.

În cazul comunicației unicast, cadrul este trimis de un host unui destinatar unic. Altfel spus, la schimbul de informații participă un singur emițător și un singur receptor. Acest mod de transmisie este cel predominant în rețelele locale și în Internet. Printre protocoalele care folosesc acest tip de comunicație amintim: HTTP, FTP, SMTP, Telnet.

## Comunicația Broadcast

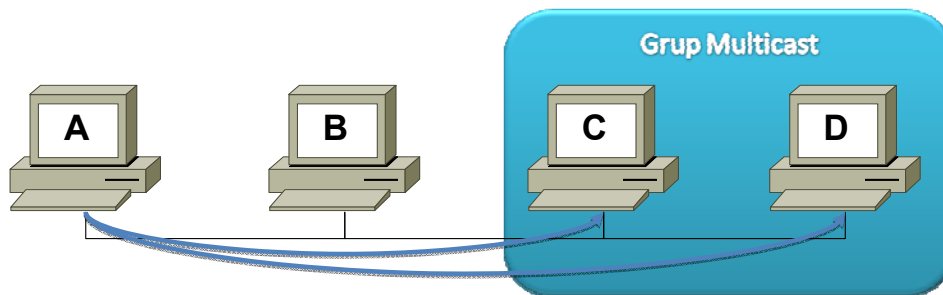
- Cadru trimis tuturor adreselor
- Adresă MAC destinație - FF:FF:FF:FF:FF:FF



În cazul comunicației broadcast, cadrul transmis de un host este destinat tuturor celorlalte host-uri care aparțin domeniului de broadcast al emițătorului. Orice echipament care primește un broadcast pe una dintre interfețele sale este obligat să-l proceseze. Această formă de comunicație este necesară în situațiile în care se dorește trimiterea aceluiași mesaj tuturor dispozitivelor din aceeași rețea (exemplu: mesajul de query al protocolului ARP).

## Comunicația Multicast

- Cadru trimis unui grup specific
- Adresă MAC destinație – paritatea primului octet este impară



În comunicația de tip multicast, destinația cadrului este o grupare specifică de echipamente (clienți). Singura formă de multicast este one-to-many: o singură stație trimite cadre către un grup de stații. Cadrele au întotdeauna în câmpul Adresă sursă o adresă unicast. Adresa de multicast nu poate să apară decât în câmpul Adresă destinație. Clienții unei transmisii multicast trebuie să fie membrii unui grup logic de multicast (multicast group) pentru a primi cadre. Exemple de fluxuri de date multicast pot fi transmisiile video și / sau audio asociate aplicațiilor de comunicare și colaborare.

## Cadrul Ethernet (1)

- Dimensiune cuprinsă (în general) între 64 și 1518 octeți

7	1	6	6	2	46 - 1500	4
Preambul	Delimitator început de cadru	Adresă Destinație	Adresă Sursă	Lungime/ Tip	Antet 802.2 și Date	FCS

Înainte de a explica rolul fiecărui câmp al cadrului Ethernet, vă reamintim faptul că acesta este rezultatul încapsulării PDU-ului de nivel 3 (stiva OSI), prin adăugarea unui header și a unui trailer.

**Preambul:** Acești 7 octeți conțin un șir alternativ de "0" și "1", cu rolul de a permite receptorului să detecteze apariția unui nou cadru pe mediu.

**Delimitator început de cadru:** Are valoarea "10101011", trecerea de la "10" la "11" semnalizând terminarea părții de sincronizare și începutul cadrului propriu-zis.

În concluzie, acești primi 8 octeți sunt folosiți pentru sincronizarea receptorului cu emițătorul.

**Adresă Destinație:** Adresa MAC a interfeței destinație.

**Adresă Sursă:** Adresa MAC a interfeței sursă.



## Cadrul Ethernet (2)

- Dimensiune cuprinsă (în general) între 64 și 1518 octeți



7	1	6	6	2	46 - 1500	4
Preambul	Delimitator început de cadru	Adresă Destinație	Adresă Sursă	Lungime/ Tip	Antet 802.2 și Date	FCS

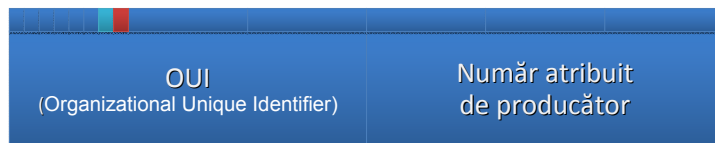
**Lungime / Tip:** Definește dimensiunea câmpului de date, fie prin specificarea lungimii efective a acestuia (valori mai mici decât 1536), fie prin specificarea protocolului de nivel superior încapsulat (valori mai mari sau egale ca 1536). Este utilizat și la verificarea integrității cadrului primit.

**Antet 802.2 și Date:** Conțin PDU-ul de nivel superior. Lungimea minimă a unui cadru este de 64 octeți (ajută la detecția coliziunilor); dacă aceasta nu este atinsă, este folosit câmpul "Pad" pentru mărirea dimensiunii până la limita minimă.

**FCS:** Câmp folosit la detecția erorilor prin utilizarea tehnicii de CRC (cyclic redundancy check). Conține CRC-ul calculat de emițător, care, dacă nu coincide cu CRC-ul calculat de destinatar la primirea cadrului, duce la aruncarea acestuia.

## Adresa MAC

- Adresă de tip BIA (**B**urned-**i**n **A**ddress)
- Bitul  (0) din primul octet -> Broadcast/Multicast
- Bitul  (1) din primul octet -> Adresă administrată local



Adresa MAC este înscrisă definitiv ("arsă") în chip-ul ROM al interfeței de rețea, este reprezentată pe 48 de biți folosind 12 "cifre" hexazecimale și este compusă din 2 părți:

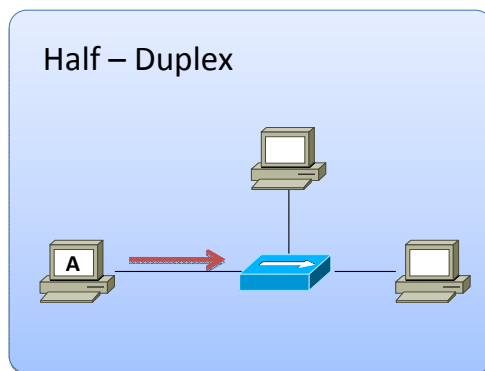
OUI Identifică, folosind 24 de biți, producătorul interfeței de rețea. Alocarea de OUI-uri este reglementată și gestionată de IEEE. Suplimentar, cei mai puțin semnificativi doi biți ai primului octet din cadrul adresei destinație determină următoarele:

Bitul (0): Dacă este setat, cadrul este destinat fie unui grup de multicast, fie broadcast.

Bitul (1): Dacă este setat, numărul atribuit de producător din adresa MAC a echipamentului sursă este administrat local.

Număr atribuit de producător: Identifică în mod unic interfața de rețea folosind cei 24 de biți rămași. Poate fi cea înscrisă din fabrică sau modificată prin software.

## Half – Duplex

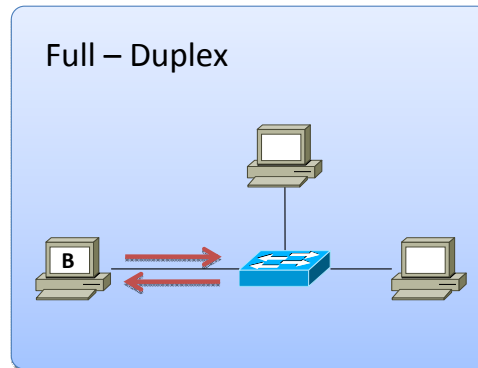


- Interfața poate să fie configurată în modurile **full**, **half** sau **auto**

După modul în care are loc propriu-zis comunicația din punctul de vedere al secvențierii fluxurilor de date, o putem clasifica în:

**Half-Duplex:** Cele două fluxuri de date (emițător → receptor și invers) nu au loc simultan, în caz contrar producându-se o coliziune. Pentru a diminua probabilitatea apariției coliziunilor și pentru a le detecta în momentele în care apar, comunicația de tip half-duplex implementează mecanismul CSMA/CD. Din păcate însă, timpii de așteptare introduși de acest mecanism duc la scăderea eficienței, motiv pentru care conexiunile half-duplex sunt întâlnite doar la echipamentele vechi, ca de exemplu hub-urile (nodurile conectate la un hub au funcționarea impusă în acest mod), switch-urile, interfețele de rețea. Din cauza acestor limitări, comunicația half-duplex a fost înlocuită cu cea full-duplex.

## Full – Duplex



- Interfața poate să fie configurată în modurile **full**, **half** sau **auto**

**Full-Duplex:** Cele două fluxuri de date (emițător → receptor și invers) au loc simultan, astfel încât se pot trimite și primi date în același timp. Deoarece există suport dedicat pentru traficul bidirecțional concomitent, este eliminată complet problema coliziunilor și nu mai este necesară folosirea mecanismului CSMA/CD, iar timpii de așteptare între transmisiile sunt micșorați, îmbunătățind eficiența.

Comparativ, o configurație half-duplex oferă o eficiență de 50 – 60 % din lărgimea de bandă disponibilă, în timp ce o configurație full-duplex oferă eficiență 100 % în ambele sensuri.

Pentru a putea funcționa în unul dintre cele două moduri, atât emițătorul cât și receptorul trebuie să fie configurate în mod identic (half- sau full-duplex). În cazul în care sunt configurate în modul auto, cele două interfețe vor "negocia" modul de funcționare.

## Tabela MAC (1)

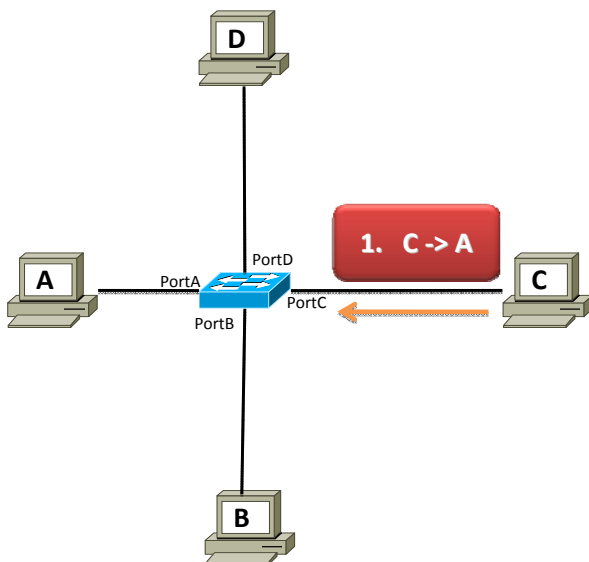


Tabela MAC

PortC: C

Tip operații

Broadcast

Switch-urile direcționează traficul primit pe un port de la nodul sursă către portul aferent nodului destinație folosind adresele MAC. Pentru a determina însă portul de ieșire corect, switch-ul trebuie să cunoască ce noduri se află "în spatele" fiecărui port în parte. Excepția o reprezintă traficul broadcast, care va fi replicat automat pe toate porturile în afara celui de pe care a venit.

Asocierile port – adresă MAC destinație sunt stocate de un switch în tabela MAC, pe măsură ce sunt determinate. Unui singur port îi pot fi asociate multiple adrese MAC, în cazul în care printr-un singur port se poate ajunge la mai multe noduri (exemplu: switch-uri interconectate).

Când un switch primește un cadru cu adresa MAC a destinatarului necunoscută (nu se află în tabela MAC), acesta este transmis pe toate porturile, cu excepția celui pe care a primit cadrul, și se reține în tabelă adresa MAC a nodului sursă (în cazul în care nu exista deja).

## Tabela MAC (2)

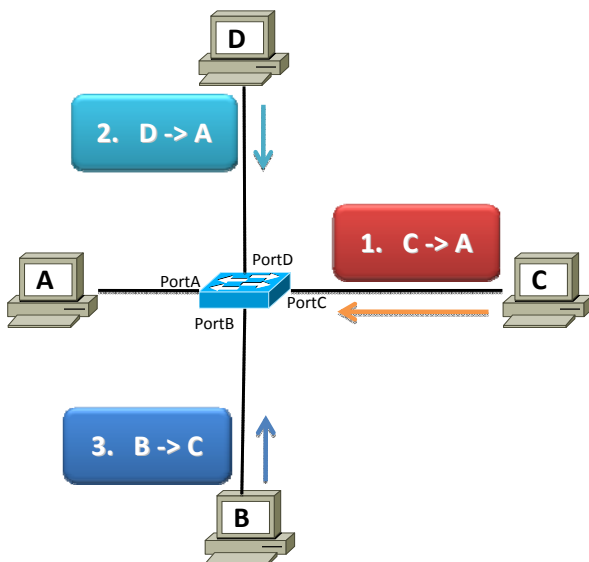


Tabela MAC	
PortC:	C
PortD:	D
PortB:	B

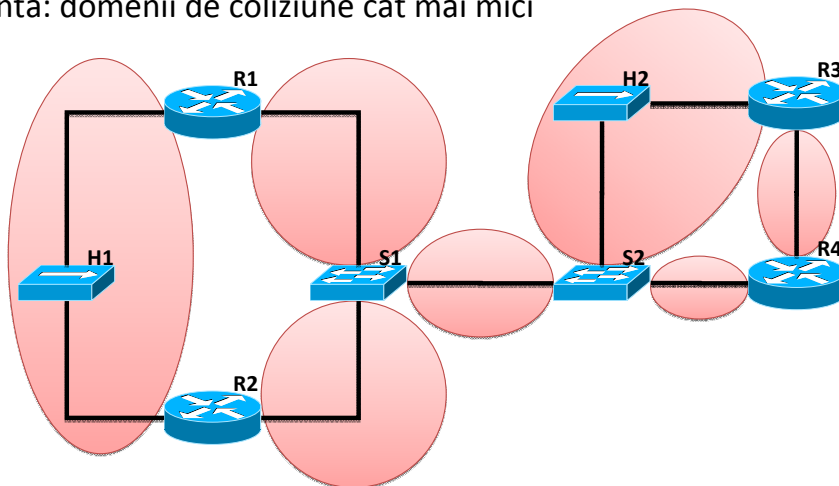
Tip operații	
Broadcast	
Broadcast	
Unicast	

În momentul în care nodul destinație din prima fază răspunde sursei inițiale, adresa acestuia este introdusă în tabelă.

În momentul în care un switch-ul din figură primește de la nodul C un cadru cu destinația A (a cărei adresă MAC nu se află în tabelă), acesta este transmis pe toate celelalte porturi (broadcast, etapa 1 din figură), iar adresa MAC a nodului C este asociată portului C. Când nodul A va răspunde, switch-ul va introduce adresa MAC a acestuia în tabelă și o va asocia portului A. Presupunând totuși că nodul A nu a răspuns înainte ca nodul D să trimită cadre cu destinația A, switch-ul va recurge din nou la broadcast la primirea acestora (etapa 2 din figură) (altfel adresa MAC a nodului A ar fi fost deja asociată portului A, și s-ar fi transmis unicast). La primirea unui cadru de la nodul B cu destinația C, cadrul va fi transmis unicast pe portul aferent deoarece adresa MAC C se află în tabela MAC(etapa 3 din figură).

## Domenii de Coliziune

- Switch-urile delimitează domeniile de coliziune
- Țintă: domenii de coliziune cât mai mici



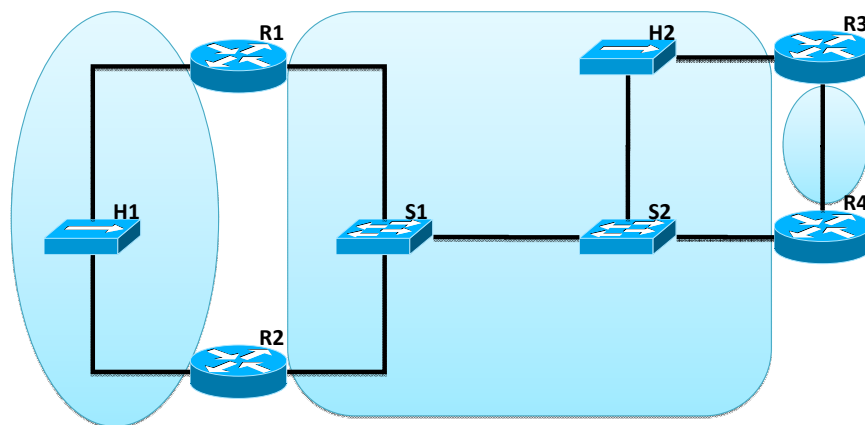
Un domeniu de coliziune este un segment al unei rețele în care pot avea loc coliziuni între cadrele transmise pe un mediu partajat (half – duplex, de exemplu cel creat de un hub). Altfel spus, este o zonă din rețea de unde provin cadre care pot produce coliziuni.

Toate nodurile conectate la un hub aparțin aceluiași domeniu de coliziune, deoarece toate împart același mediu de comunicație. În schimb, când un nod este conectat la un switch, acesta din urmă creează o conexiune dedicată pentru a separa traficul asociat nodului de restul traficului, iar domeniul de coliziune este limitat la acesta. Astfel, porturile unui switch aparțin unor domenii de coliziune distincte.

Proiectarea corectă a unei rețele presupune, printre altele, micșorarea domeniilor de coliziune, mărind astfel eficiența și procentajul lărgimii de bandă neutilizate.

## Domenii de Broadcast

- Switch-urile nu delimitează domeniile de broadcast
- Țintă: domenii de broadcast cât mai mici



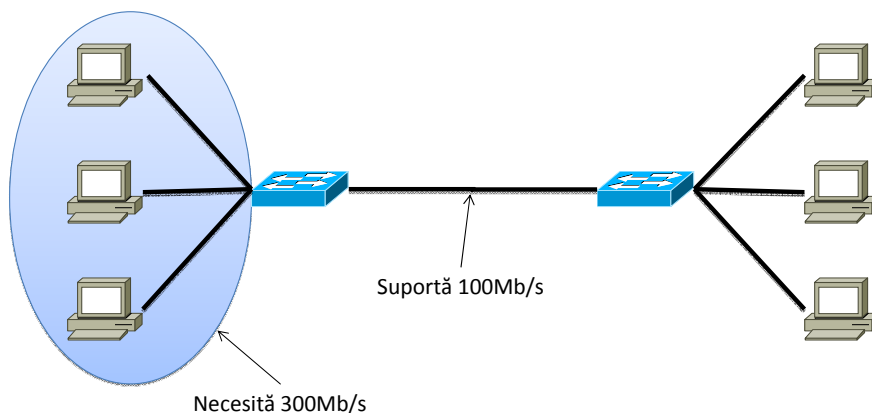
Domeniul de broadcast reprezintă partea unei rețele în care dacă un nod transmite un cadru broadcast, toate celelalte noduri în vor primi. De exemplu, toate nodurile conectate la un switch aparțin aceluiași domeniu de broadcast și, evident, nodurile conectate folosind două sau mai multe switch-uri în cascadă vor aparține aceluiași domeniu de broadcast.

Broadcast-urile la nivel 2 (stiva OSI) pot fi filtrate doar de un echipament de layer 3 (exemplu: ruter) sau folosind VLAN-uri. Utilizarea VLAN-urilor este explicată în capitolul următor.



## Congestii

- Fiecare dispozitiv din cale introduce latență



Latența este definită ca timpul necesar unui cadru sau unui pachet să ajungă de la sursă la destinația finală și este cauzată din cel puțin 3 motive:

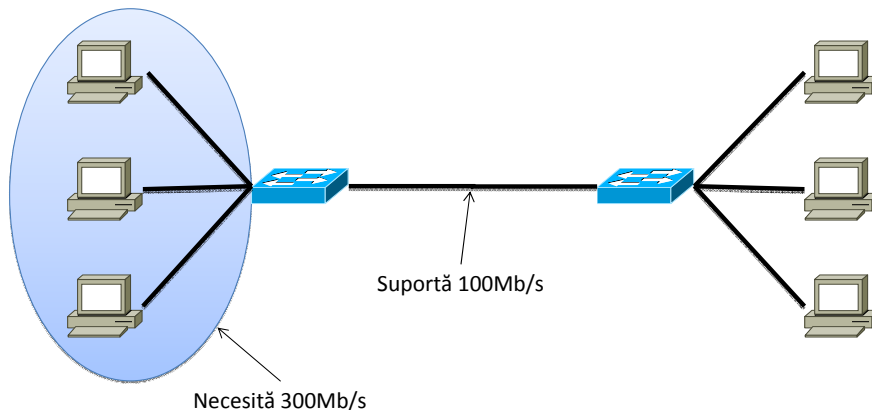
**Timpul necesar interfeței** de rețea a sursei să emită ("plaseze") pulsurile pe mediu și cel necesar interfeței de la destinație să le interpreteze.

**Durata necesară semnalului electric** să parcurgă efectiv secțiunile de cablu folosite la interconectare.

**Timpul necesar "traversării" echipamentelor** de rețea intermediare. De exemplu, timpul necesar unui switch (operații la nivelul 2 OSI) este mai mic decât cel necesar unui ruter (operații la nivelurile 2 și 3 OSI).

## Congestii

- Rețeaua trebuie să evite apariția bottleneck-urilor



Congestiile apar în momentele în care lărgimea de bandă a unui mediu este mai mică decât cantitatea de date care ar trebui să traverseze mediul în unitatea de timp considerată. Cele mai des întâlnite motive ale congestiilor sunt:

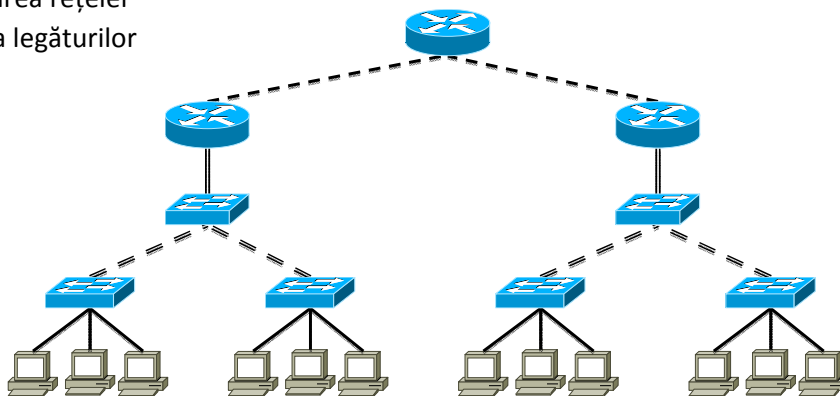
- Echipamentele moderne (PC-uri, periferice, etc.) pot să proceseze și să trimită date la viteze mult mai mari decât în trecut.
- Creșterea volumului de trafic efectuat în mod uzual, depășind volumul pentru care a fost proiectat segmentul de rețea.
- Traficul de tip broadcast (de exemplu cereri ARP, DHCP, etc.) și / sau domenii de broadcast extinse.

Congestiile apar și în cazul unor topologii asimetrice, în care un server este conectat pe unul dintre porturile unui switch, restul porturilor oferind conectivitate pentru 20 de clienți care trimit date la capacitatea maximă a conexiunii. În cazul în care porturile switchului sunt de viteză egală se ajunge la o congestie a legăturii pentru server. Aceasta

## Latență și Congestii

### ▪ Soluții:

- design ierarhic
- segmentarea rețelei
- agregarea legăturilor



Pentru a evita creșterea latențelor și producerea congestiilor, într-un LAN se recomandă luarea următoarelor măsuri:

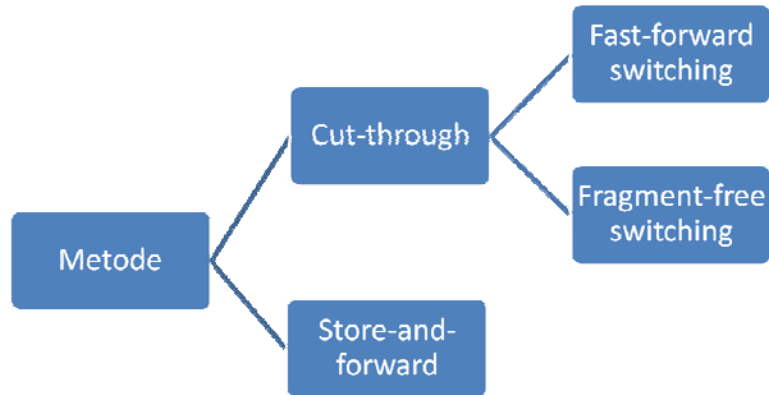
**Folosirea unui design ierarhic** la proiectarea rețelei. Modelul clasic de ierarhizare conține 3 niveluri: "core", "distribution" și "access".

**Segmentarea rețelei** în domenii de coliziune și de broadcast multiple și reduse ca dimensiune, prin folosirea de rutere și switch-uri. În rețelele moderne nu se mai folosesc bridge-uri și hub-uri.

**Dimensionarea corectă** a echipamentelor intermediare, atât din punctul de vedere al vitezelor porturilor, cât și din cel al circuitelor interne. De exemplu, un switch folosit în partea de "core" cu 24 de porturi Gigabit full-duplex trebuie să conțină circuite interne ("switch fabric") care să ofere o lățime de bandă de aproximativ 48 Gb/s.

**Înlocuirea interfețelor** de rețea cu unele care suportă viteze mai mari și / sau agregarea de legături. Agregarea de legături reprezintă utilizarea mai multor legături fizice între două switch-uri ca o singură

## Metode de forwarding



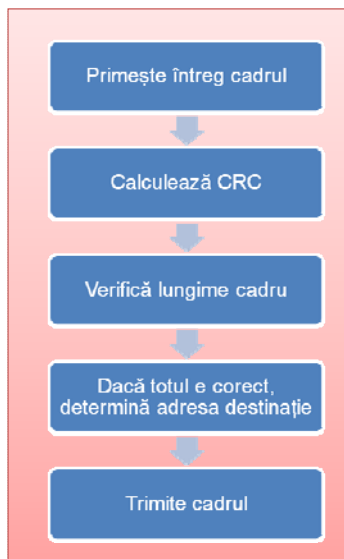
Switch-urile pot funcționa în moduri diferite, fiecare mod având avantaje și dezavantaje față de celelalte.

Principalele două metode folosite la forwarding-ul cadrelor de pe un port pe altul sunt "cut-through" și "store-and-forward" switching.

Mai mult, procesul de switching se poate desfășura fie simetric, fie asimetric, poate avea loc atât la nivelul 2 al stivei OSI, cât și la nivelul 3, iar memoria tampon poate fi ori rezervată pentru fiecare port în parte ori comună pentru toate porturile.

Toate aceste opțiuni trebuie înțelese și avute în vedere la proiectarea, îmbunătățirea sau extinderea unei rețele.

## Store-and-Forward Switching



Dacă un switch implementează metoda "store-and-forward", atunci secvențierea operațiilor care au loc între primirea și trimiterea unui cadru are loc conform diagramei din imagine:

1. Cadrul venit pe un port este stocat într-un buffer până când este primit în totalitate.
2. Se calculează codul de CRC al cadrului primit și se determină lungimea pachetului, valoare care se compară cu cea trimisă de sursă în antetul Ethernet. Dacă cel puțin una dintre 2 valori nu coincide, pachetul este aruncat.
3. Dacă pachetul nu a fost aruncat, se determină adresa MAC a destinației.
4. Cadrul este trimis pe portul aferent adresei MAC a destinației.

## Store-and-Forward Switching - Dezavantaje



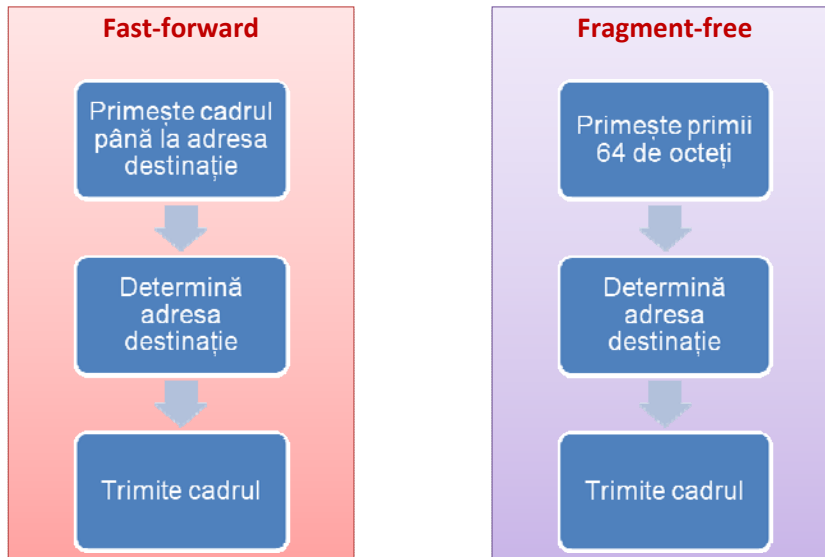
Performanțe scăzute

Cerințe mari de sistem

Dezavantajele acestei metode sunt legate de performanțe, deoarece switch-ul trebuie să memoreze întregul cadru înainte de a-l verifica și de a-l trimite, rezultând astfel latențe mai mari. În cazul în care există mai multe switch-uri de-a lungul traseului cadrului, cadrul va fi verificat de fiecare dintre ele, iar performanțele rețelei vor scădea. Un alt dezavantaj ar fi faptul că un astfel de switch necesită buffer-e de dimensiuni mai mari și mai multe cicluri CPU pentru a realiza aceste verificări decât unul care implementează metoda "cut-through".

Cu toate acestea, marea majoritate a switch-urilor moderne folosesc această metodă deoarece este necesară implementării tehnicii de QoS, iar evoluția hardware-ului folosit a făcut ca diferența din punct de vedere al latenței introduse să devină ne semnificativă.

## Cut-Through Switching



În comparație cu metoda "store-and-forward", cea "cut-through" diferă prin următoarele aspecte:

În cazul "fast-forward" cadrul este primit până la adresa destinație, iar apoi este trimis fără a se verifica integritatea sa. Această variantă oferă cea mai mică latență și este cea obișnuită în switching-ul "cut-through".

În cazul "fragment-free" switching se primesc primii 64 de octeți (erorile apar de obicei în acest interval) și se verifică integritatea acestora. Dacă nu se detectează nici o eroare cadrul este trimis spre destinație. Această variantă reprezintă un compromis între switching-ul "store-and-forward" și cel "cut-through fast-forward" din punct de vedere al latențelor introduse și al integrității cadrelor.

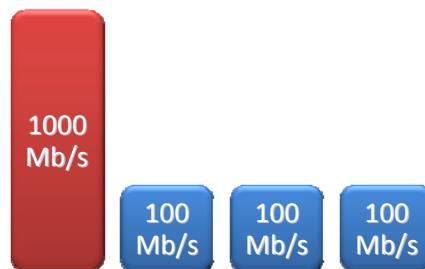
Switch-urile care implementează această metodă sunt folosite de obicei pentru aplicații și în sisteme HPC (high performance computing), care necesită latențe inter-proces foarte mici.

## Switching simetric/asimetric

- Simetric:
  - toate porturile au același bandwidth



- Asimetric:
  - porturile au bandwidth diferit



Împărțirea procesului de switching în simetric și asimetric se realizează după modul în care este alocată lărgimea de bandă porturilor:

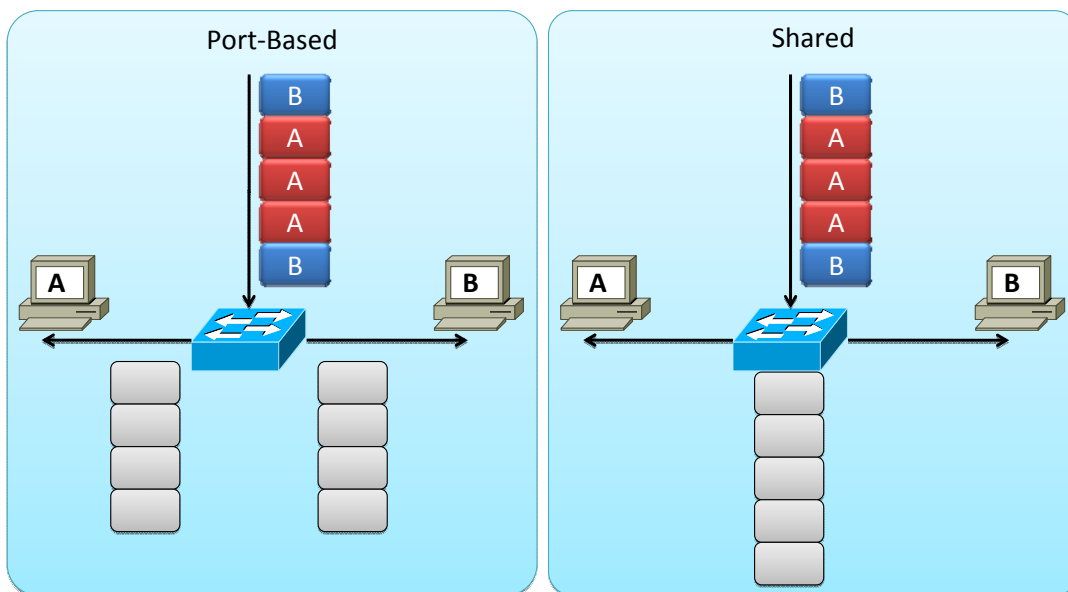
**Simetric:** Toate porturile au alocată aceeași lărgime de bandă. Aceste switch-uri sunt optimizate pentru trafic distribuit.

**Asimetric:** Toate porturile cu excepția unuia au alocată aceeași lărgime de bandă (porturi pentru clienți). Portul rămas are o lărgime de bandă alocată mult mai mare și este dedicat conectării unui server. Aceste switch-uri sunt optimizate pentru scenarii client-server. Pentru a se putea realiza în mod optim transmisia la două viteze diferite, cadrele sunt stocate integral în memoria tampon și transmise pe porturi la momentul potrivit.

Majoritatea switch-urilor moderne sunt asimetrice datorită flexibilității oferite.



## Memory Buffering



Stocarea cadrelor în memoria tampon are loc fie în timpul procesului de switching (parțial sau total, în funcție de metodă), fie în situația în care portul de ieșire este blocat din cauza unei congestii. Există două moduri în care se realizează această stocare temporară, și anume:

**Port-Based:** Fiecare port are o coadă de dimensiune fixă asociată. Ca urmare, un cadru aflat în coadă va fi trimis doar după ce toate cadrele de dinaintea sa au fost trimise. Astfel, este posibil ca un cadru să le întârzie pe toate cele care îi urmează (de exemplu, cazul portului ocupat din cauza unei congestii).

**Shared:** Există o singură zonă de memorie, folosită în comun de toate port-urile, cantitatea de memorie alocată fiecăruia fiind ajustată dinamic. Cadrele aflate în memorie sunt asociate porturilor de ieșire corespunzătoare. Numărul cadrelor aflate în memorie este limitat doar de dimensiunea acesteia și nu are o valoare maximă pentru fiecare port în parte.

## Layer 2 vs Layer 3 Switching (1)



**Switching nivel 2**



**Switching nivel 3**

Un switch de layer 2 va folosi doar informația ce se regăsește la acest nivel în procesul de switching. Astfel, în tabela sa MAC fiecărui port îi vor fi asociate una sau mai multe adrese MAC și nimic mai mult.

În schimb, un switch de layer 3 va analiza atât nivelul 2, cât și nivelul 3 în procesul de switching. Practic, asocierile din tabela MAC vor fi extinse prin adăugarea adresei IP aferente fiecărei adrese MAC și, implicit, fiecărui port.

Vom ilustra această diferență dintre cele 2 tipuri de switch-uri în exemplul următor:

## Layer 2 vs Layer 3 Switching (2)



Catalyst 2960

**Switching nivel 2**



Catalyst 3750

**Switching nivel 3**

Să presupunem că avem host-uri din două rețele diferite (LAN 1, LAN 2) conectate la același switch de layer 2. În momentul în care un host din LAN 1 trimite un broadcast, acesta va fi retransmis de switch pe toate celelalte porturi ale sale, indiferent dacă acestea aparțin rețelei LAN 1 sau LAN 2. În această situație, host-urile din LAN 2 vor primi și procesa inutil broadcast-ul la nivelul 2, urmând să îl arunce în timpul procesării la nivelul 3. Această problemă poate fi rezolvată prin înlocuirea switch-ului cu unul de layer 3, care va ține cont de faptul că broadcast-ul are ca sursă un echipament din LAN 1 și îl va trimite mai departe doar pe celelalte porturi ale sale care aparțin LAN 1.

Suplimentar, față de un switch de layer 2, un switch de layer 3 poate să efectueze și rutare de pachete (la fel de repede precum realizează switching-ul datorită implementării în hardware a acestor funcții).

## Layer 3 Switch vs Ruter

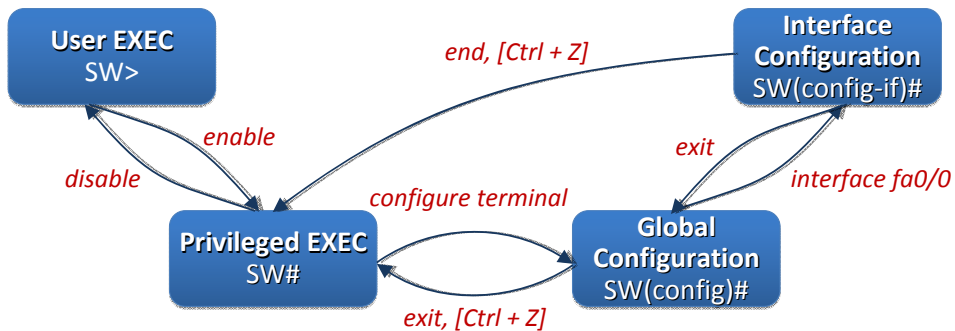
	Switch de nivel 3	ruter
Suport pentru WIC		Da
Rutare de nivel 3	Da	Da
Protocoale avansate de rutare		Da
Rutare la viteza interfeței	Da	Da

Cu toate că un switch de layer 3 este capabil să realizeze și rutarea pachetelor, acesta nu va putea elimina nevoia unui ruter în anumite situații. Enumerăm doar câteva dintre facilitățile oferite de rutare în plus față de majoritatea switch-urilor:

- Un ruter poate rula protocoale complexe de rutare (exemplu: BGP).
- Un ruter oferă suport mult mai flexibil pentru WIC-uri (WAN interface cards).
- Un ruter poate stabili conexiuni pentru remote-access (exemplu: VPN).

Anumite switch-uri pot oferi toate funcționalitățile enunțate mai sus (ex: Catalyst 6500).

# Modurile CLI din IOS



Reamintim faptul că sistemul de operare Cisco IOS este organizat ierarhic în moduri de operare, fiecare mod având propriul domeniu de operare și fiind folosit în vederea realizării unor operații specifice cu ajutorul comenzilor disponibile în acesta. Modurile principale sunt (în ordine top - down):

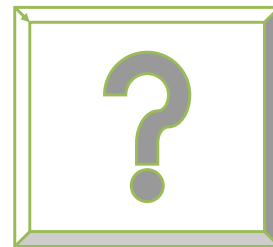
- User executive mode
- Privileged executive mode
- Global configuration mode
- Other specific configuration mode (exemplu: Interface configuration mode)

Comenzile folosite pentru a trece dintr-un mod de operare în altul sunt prezentate în imaginea de mai sus.

## Căutare ajutor



- Introducerea în CLI a caracterului "?"
- *cisco.com*
- Forumul de ajutor de pe *ccna.ro* din cadrul cursului



Sistemul de operare IOS oferă ajutor în două direcții:

### Denumirea unei comenzi

În cazul în care se cunosc doar primele caractere dintr-o comandă, acestea pot fi introduse, după care se apasă tasta "?" (fără a se lasă spațiu între caractere și "?"). În acest moment vor fi afișate toate comenzile care încep cu șirul respectiv de caractere și sunt disponibile în modul de operare în care ne aflăm.

### Sintaxa unei comenzi

În cazul în care denumirea unei comenzi este cunoscută, însă parametrii necesari completării acesteia sunt necunoscuți, se apelează tot la simbolul "?". Introducerea acestora are ca efect afișarea tuturor parametrilor care (mai) pot fi folosiți. Dacă este afișat și "<cr>", atunci nu mai este nevoie de nici un alt parametru pentru a asigura funcționarea comenzii.

# Erori IOS



Eroare	Exemplu	Cauză	Soluție
Ambiguous command	SW# a	IOS-ul nu poate determina cu exactitate ce comandă să execute	Completarea numelui comenzii
Incomplete command	SW# show	Sunt necesari parametri suplimentari pentru execuție	Adăugarea parametrilor lipsă
Invalid input	SW(config-if) #ip address 172.16.10.0.0 255.255.255.0	IOS-ul nu poate parse sintactic comanda	Analizarea formatului comenzii

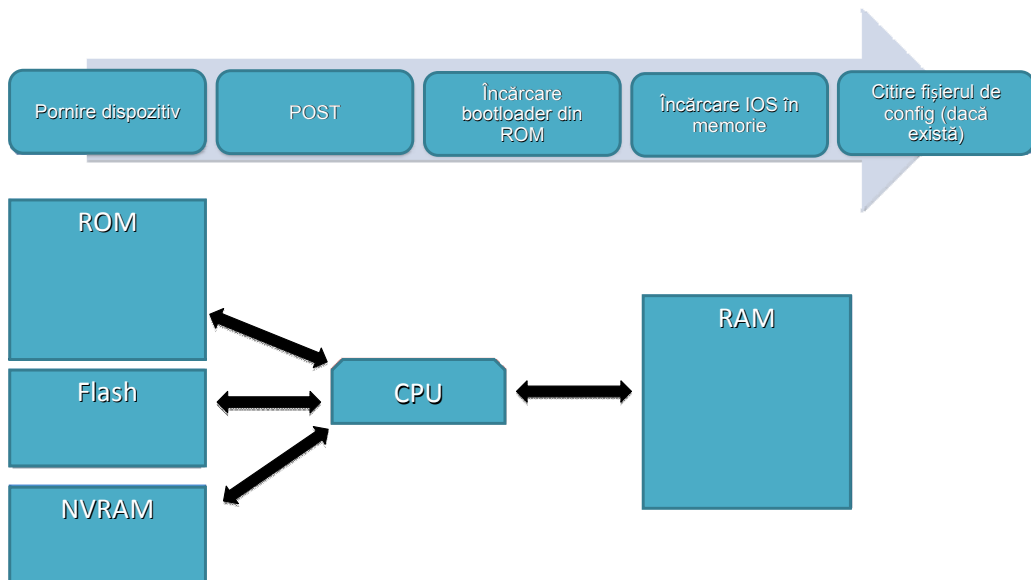
În Cisco IOS întâlnim trei tipuri de erori semnalate de sistemul de operare cu funcții de a ajuta utilizatorul să detecteze comanda dorită.

Astfel, când este semnalată eroarea "Ambiguous command" sistemul de operare nu poate determina cu exactitate care este comanda dorită. Această eroare apare cel mai adesea când comanda este executată sub o formă prescurtată care pentru IOS poate coincide cu mai multe comenzi. Pentru a rezolva această eroare este necesară scrierea numelui comenzii sub formă întreagă.

Eroarea "Incomplete command" apare atunci când o comandă este executată fără toți parametrii necesari. Soluția în acest caz este adăugarea unor parametri suplimentari.

Eroarea "Invalid input" apare atunci când comanda executată este scrisă incorect și sistemul nu o poate interpreta. Pentru a soluționa această eroare trebuie reanalizat, de obicei din punct de vedere sintactic, formatul comenzii.

## Procesul de boot



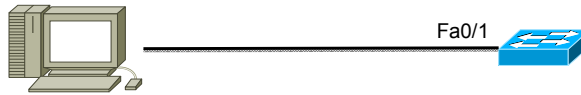
Secvența după care se desfășoară procesul de boot este următoarea:

- Se încarcă boot-loader-ul din memoria non-volatilă (NVRAM)
- Boot-loader-ul
  - inițializează CPU-ul la nivel scăzut
  - realizează procedura de POST (Power-On Self Test)
  - inițializează sistemul de fișiere
  - încarcă sistemul de operare IOS în memoria volatilă (RAM)
  - Sistemul de operare este căutat mai întâi în flash, iar apoi, dacă nu este găsit, este căutat pe un server tftp, existent în rețea. În cazul în care și a doua încercare eșuează, se va încarca un SO minimal din ROM.
- Sistemul de operare rulează folosind configurația găsită în fișierul "config.txt" din memoria flash, dacă acesta există. Altfel va folosi o configurație "default".



## Configurare conectivitate IP <sup>(1)</sup>

- Este necesară folosirea unei interfețe VLAN



```
Switch# configure terminal
Switch(config)# interface vlan 50
Switch(config-if)# ip address 192.168.10.2
255.255.255.0
Switch(config-if)# no shutdown
```

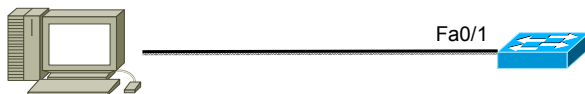
Pentru a administra un switch de la distanță este necesar ca acel switch să aibă asignată o adresă IP. Acest IP este adăugat unei interfețe virtuale numită Virtual LAN (VLAN) și apoi această interfață este asignată unuia sau mai multor port-uri ale switch-ului.

În mod implicit switch-ul este administrat prin VLAN-ul 1, dar acesta poate fi schimbat cu orice VLAN dorit.

Pentru a configura o adresă IP și o mască de rețea pe VLAN-ul de administrare al unui switch, trebuie să accesăm modul de configurare al unei interfețe VLAN folosind comanda **interface vlan id\_vlan** unde **id\_vlan** este numărul VLAN-ului dorit. În acest mod configurăm adresa IP și masca de rețea folosind comanda **ip address adresă mască** și pornim interfața folosind comanda **no shutdown**.

## Configurare conectivitate IP (2)

- Este necesară folosirea unei interfețe VLAN



```
Switch(config-if)# interface fastethernet 0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 50
Switch(config-if)# end
```

În final, trebuie ca VLAN-ul de management nou creat să fie adăugat cel puțin unei interfețe a switch-ului prin care acesta să poată fi administrat. Pentru aceasta, intrăm în modul de configurare al unei interfețe folosind comanda **interface fastethernet/serial număr\_identificare\_interfață** și adăugăm VLAN-ul folosind comenzile **switchport mode access** și **switchport access vlan id\_vlan** unde **id\_vlan** este numărul VLAN-ului.

Vom discuta mai multe despre VLAN-uri în capitolul următor.

## Configurare default gateway

- Necesară pentru a putea comunica și cu alte rețele



```
Switch# configure terminal
Switch(config)# ip default-gateway 192.168.10.1
Switch(config)# end
```

Vrem să configurăm switch-ul astfel încât să putem trimite pachete IP și către rețele diferite de cea în care se află. Pentru aceasta folosim default gateway-ul. Switch-ul trimite pachete IP cu adrese destinație ce nu fac parte din rețeaua sa către default gateway.

Pentru a configura un default-gateway pe switch se folosește comanda **ip default gateway adresă** unde adresa reprezintă adresa ip a unui ruter aflat în aceeași rețea cu switch-ul. Ruter-ul va trimite pachetele primite de la switch mai departe către destinație.

## Configurare duplex și viteză



```
Switch# configure terminal
Switch(config)# interface fastethernet 0/1
Switch(config-if)# duplex auto
Switch(config-if)# speed auto
Switch(config-if)# end
```

Folosind comenzile **duplex** *tip* și **speed** *valoare* putem specifica manual modul de funcționare al interfeței (half duplex, full duplex sau auto), cât și viteza port-urilor switch-urilor pentru a evita autonegocierea cu alte echipamente de rețea.

Ambele comenzi se execută din modul **interface**, și pentru a le salva în NVRAM este necesară comanda **copy running-config startup-config** sau comanda **write**.

## Configurare interfață web



```
Switch# configure terminal
Switch(config)# ip http authentication enable
Switch(config)# ip http server
Switch(config)# end
```

Switch-urile moderne Cisco au numeroase utilitare ce necesită ca echipamentul să fie configurat ca HTTP server. Printre acestea se numără interfața web pentru echipamente Cisco, Cisco Security Device Manager, aplicații IP Phone și aplicații Cisco IOS Telephony Service.

Pentru a controla cine poate accesa serverul HTTP se poate configura opțional și autentificare.

Comenzile utile pentru configurarea switch-ului ca server HTTP se execută din modul `configure terminal` și sunt **“ip http authentication enable”** și **“ip http server”**. Pentru ca setarea să devină permanentă este necesară salvarea configurației în NVRAM folosind ***copy running-config startup-config*** sau ***write***.

## Înregistrări MAC statice



```
Switch# configure terminal
Switch(config)# mac-address-table static
                 a8:12:34:56:78:90 vlan 10 interface fa0/1
Switch(config)# exit
Switch# show mac-address-table
```

Switch-urile folosesc tabela CAM pentru a determina pe ce porturi să trimită pachetele primite. Într-o tabelă CAM se pot înregistra adrese MAC atât static cât și dinamic.

Adresele MAC dinamice sunt adrese MAC învățate de switch care după o anumită perioadă de timp (proces numit “aging”) dispar din tabelă dacă nu sunt folosite.

Adresele MAC statice sunt adăugate de administrator pe anumite porturi. Adresele adăugate static nu dispar niciodată din tabela CAM și sunt trimise mereu pe portul pe care ele au fost setate. Adresele de tip static oferă administratorului control total asupra accesului la rețea.

Pentru a crea o mapare statică se folosește comanda **mac-address-table static mac\_sursă vlan id\_vlan interface fastethernet/serial id\_interfață** din modul configure terminal.

Pentru a vizualiza MAC-urile aflate în tabela CAM se folosește

## Comenzi show utile

Comandă	Efect
<code>show interfaces [interface-id]</code>	Informație despre o interfață
<code>show startup-config</code>	Afișează configurația încărcată la pornire
<code>show running-config</code>	Afișează configurația activă
<code>show flash:</code>	Afișează conținutul memoriei flash
<code>show version</code>	Informații despre hardware-ul și software-ul de sistem
<code>show history</code>	Istoria comenzilor efectuate
<code>show ip interface [interface-id]</code>	Informații de nivelul 3 despre o interfață
<code>show cdp neighbors</code>	Vecinii descoperiți prin CDP
<code>show mac-address-table</code>	Afișează tabela de forwarding

Pentru a verifica corectitudinea configurațiilor realizate, sunt utilizate comenzile de tip **show**.

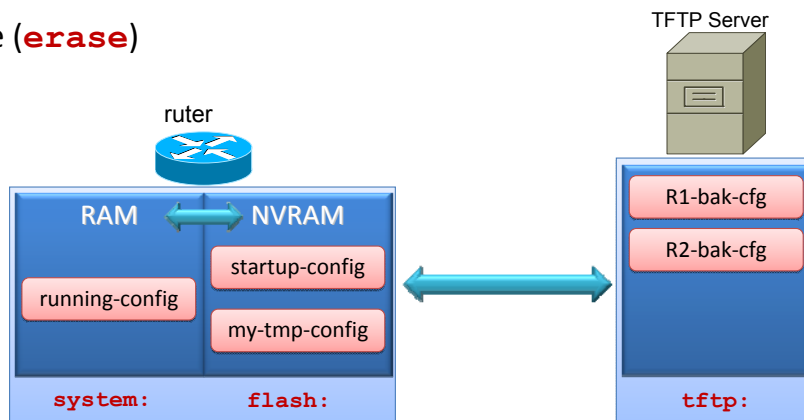
Comanda **show** se execută din modul Privileged EXEC și poate fi folosită cu diverși parametrii în funcție de configurația pe care vrem să o vizualizăm.

O comandă importantă **show** este comanda **show running-config**. Cu aceasta afișăm configurațiile ce rulează în acest moment pe switch. Pentru a afișa configurația pe care switch-ul o încarcă la boot-are se folosește comanda **show startup-config**.

O altă comandă utilă este **show interfaces [fastethernet/serial] [id\_interfață]** unde parametrii aflați între [] sunt opționali. Dacă adăugăm interfața dorită, această comandă ne arată starea și alte informații despre interfața respectivă. Dacă nu specificăm interfața, comanda ne arată datele pentru toate interfețele.

## Manipularea configurațiilor

- Backup/restaurare (**copy**)
  - server TFTP
  - NVRAM
- Ștergere (**erase**)



Echipamentele Cisco permit salvarea configurațiilor în mai multe moduri.

Dacă ne dorim să păstrăm mai multe fișiere de configurare inițială putem opta să salvăm una dintre configurații în flash folosind comanda **copy startup-config flash:nume\_fișier**. Salvând mai multe variante ale configurației inițiale putem să revenim în orice moment la o configurație anterioară funcțională.

Pentru a reveni la o configurație anterioară salvată în flash nu trebuie decât să copiem configurația peste cea actuală folosind comanda **copy flash:nume\_fișier startup-config** și apoi să resetăm switch-ul folosind comanda "reload" din modul EXEC.

O altă metodă de a salva fișiere de configurare este folosirea unui server TFTP. Putem realiza acest lucru deoarece IOS-ul Cisco conține un client TFTP care permite conectarea la un server TFTP ce se află în aceeași rețea cu echipamentul.



## Configurare acces consolă



```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# password cisco
Switch(config-line)# login
Switch(config-line)# end
```

Prin portul de consolă al unui echipament Cisco, se pot realiza orice configurații. Din acest motiv securizarea acestui port este benefică și necesară.

Pentru a realiza acest lucru setăm autentificarea pe portul de consolă folosind o parolă de login.

Pentru a seta această parolă intrăm în modul de configurare **config-line** folosind comanda **line console 0** din modul **global config**. Pentru a determina autentificarea prin parolă se folosește comanda **password parolă**. Pentru ca unui utilizator să îi fie necesară parola la conectarea prin portul de consolă trebuia ca la final să folosim comanda **login**.

Pentru a reveni la conectare fără parolă trebuie ca din modul config-line să executăm comenzile **no password** și **no login**.

## Configurare acces terminal virtual



```
Switch# configure terminal
Switch(config)# line vty 0 4
Switch(config-line)# password cisco
Switch(config-line)# login
Switch(config-line)# end
```

Porturile vty ale unui switch permit accesarea acestuia de la distanță. Prin porturile de vty se pot realiza orice fel de configurații. Astfel, accesul fizic la echipament nu este neapărat necesar și este foarte important ca porturile vty să fie securizate.

Pentru a oferi un minim de securitate acestor porturi putem să setăm o parolă pentru accesul prin liniile de vty. Pe un switch Cisco pot fi numeroase porturi de vty pentru ca mai mulți administratori să poată configura switch-ul în același timp. Astfel, pentru a securiza switch-ul, trebuie ca toate porturile să necesite introducerea parolei.

Pentru a intra în modul de configurare a liniilor vty se folosește comanda **line vty *primul\_port ultimul\_port*** unde între *primul\_port* și *ultimul\_port* este intervalul port-urilor de acces la distanță. Pentru a seta apoi parola, se procedează idem ca la securizarea portului de consolă, folosind comanda **password *parolă*** urmată apoi de comanda **login**.

## Configurare parolă EXEC

- Protejează accesul la modul Privileged Exec

```
Switch# configure terminal
; dacă se dorește afișarea în text clar:
Switch(config)# enable password cisco
; dacă se dorește afișarea hash-ului:
Switch(config)# enable secret cisco
Switch(config)# end
```

În modul Privileged Exec, orice utilizator poate configura toate opțiunile disponibile pe echipament, inclusiv folosirea comenzilor de tip **show** pentru observarea parolelor necriptate. Din acest motiv este foarte importantă securizarea accesului la acest mod.

Comanda **enable password parolă** permite setarea unei parole pentru restricționarea accesului la modul Privileged Exec. Folosind această comandă parola este salvată necriptat în running și startup config. Astfel, folosind comenzi de tip **show**, parola poate fi citită din fișierele de configurare. Din aceasta cauză Cisco a introdus comanda **enable secret parolă** ce salvează parola sub formă de hash în fișierele de configurare.

## Criptarea parolelor

- Comanda criptează parolele în text clar din configurații
- Folosește criptare type 7
- Parolele astfel criptate sunt foarte ușor de spart

```
Switch# show running-config
...
  password cisco
...
Switch# configure terminal
Switch(config)# service password-encryption
Switch(config)# end
Switch# show running-config
...
  password 1511021F0725
...
```

Când configurăm parole în Cisco IOS CLI, implicit toate parolele (exceptând enable secret) sunt salvate în format clear text în fișierele de configurare startup-config și running-config.

Folosind comanda **service password-encryption** toate parolele din sistem sunt stocate sub forma criptată. Imediat ce comanda a fost executată din modul global config, toate parolele salvate în fișierele de configurare sunt convertite într-o formă criptată.

## Configurare bannere

- Login banner

```
Switch(config)# banner login "Accesul interzis!"
```

- MOTD banner

```
Switch(config)# banner motd "Test congestie joi"
```

Cisco IOS permite configurarea unor mesaje ce apar oricărei persoane ce se autentifică pe echipament.

Aceste mesaje se numesc banner login sau banner motd ( Message Of The Day).

Pentru a configura un mesaj ce va aparea înainte de autentificarea cu username si parolă, se folosește comanda **banner login mesaj** . Mesajul trebuie scris între ghilimele.

Comanda **banner motd mesaj** configurează un mesaj ce va apărea la accesarea echipamentului de la distanță.

## Configurare Telnet

- Telnet

```
Switch(config)# line vty 0 4  
Switch(config-line)# transport input telnet
```

Un switch Cisco poate fi accesat de la distanță prin două metode: Telnet și SSH.

Telnet este un protocol popular deoarece majoritatea sistemelor de operare din ziua de azi conțin un client de Telnet preinstalat. Este metoda originală ce este suportată pe toate echipamentele Cisco, însă este nesecurizată deoarece protocolul transmite toate datele necriptat.

În mod implicit, echipamentele pot fi accesate folosind Telnet, însă, pentru a specifica explicit acest lucru se poate folosi comanda **transport input telnet** din modul de configurare al liniilor de vty.

## Configurare SSH

- SSH

```
Switch(config)# ip domain-name somedomain.com
Switch(config)# crypto key generate rsa
Switch(config)# ip ssh version 2
Switch(config)# line vty 0 4
Switch(config-line)# transport input ssh
```

SSH a devenit protocolul preferat de conectare la distanță pe echipamentele Cisco, deoarece acesta adresează problema securității introdusă de folosirea protocolului Telnet. Comunicația SSH între client și server este criptată. Actualmente echipamentele Cisco suportă atât SSH versiunea 1 cât și SSH versiunea 2. Se recomandă folosirea SSH v2 atunci când este posibil deoarece criptarea folosită este mai puternică decât în cazul versiunii anterioare.

SSH poate folosi diferite standarde de criptare a datelor (DES, 3DES). Pentru a implementa SSH este necesară generarea unor chei RSA. RSA folosește o cheie publică și o cheie privată pentru a realiza autentificarea. Algoritmii de autentificare vor fi studiați în detaliu la CCNA4.

## Atacuri de securitate (1)

- MAC address flooding
- DHCP spoofing

### MAC adres flooding

Se generează și se trimite unui switch trafic "fals" folosind un număr foarte mare de adrese MAC sursă pentru a umple tabela MAC a switch-ului. În momentul în care aceasta a ajuns la limita superioară, switch-ul se va comporta ca un hub, permițând unui atacator conectat la switch să "vadă" tot traficul care trece prin acesta.

### DHCP spoofing

Se activează un server DHCP "pirat" în rețeaua țintă, care să răspundă cererilor DHCP înaintea server-ului legitim, configurând astfel host-urile după dorința atacatorului. Din acest moment, tot traficul destinat gateway-ului din acea rețea poate fi redirectat către atacator.



## Atacuri de securitate (2)



- DHCP starvation
- Atacuri folosind CDP
- Brute Force
- DoS

### **DHCP starvation**

Se trimite un număr mare de cereri DHCP false, epuizând astfel spațiul de adrese IP disponibil.

### **Brute Force**

Spargerea unei parole prin generarea tuturor combinațiilor posibile, începând cu cele uzuale.

### **DOS (Denial Of Service)**

Atacuri care au ca rezultat blocarea accesului la o resursă sau un serviciu.

## Port Security

- Limitează adresele MAC permise pe un port
- Pe o interfață se pot configura
  - un grup de adrese MAC valide
  - o singură adresă MAC validă
  - comportamentul portului când o regulă este încălcată
- Adrese MAC sigure
  - statice
  - dinamice
  - sticky

Adresele MAC sigure pot fi statice ( configurate manual, salvate în tabela CAM și în fișierul de configurație în mod automat ), dinamice ( învățate și salvate în tabela MAC în mod dinamic ) și "sticky" ( învățate și salvate în tabela MAC și în fișierul de configurație în mod dinamic ).

Următoarele 2 situații se consideră o încălcare a regulilor de securitate:

- Se depășește numărul maxim de adrese MAC definit pentru o interfață a switch-ului.
- O adresă MAC învățată sau configurată pe o interfață securizată este depistată pe o altă interfață securizată din același VLAN.

## Port Security

- Limitează adresele MAC permise pe un port
- Pe o interfață se pot configura
  - un grup de adrese MAC valide
  - o singură adresă MAC validă
  - comportamentul portului când o regulă este încălcată
- Adrese MAC sigure
  - statice
  - dinamice
  - sticky

Interfețele unui switch pot fi configurate să se comporte într-unul dintre modurile următoare:

- Protect: La atingerea pragului maxim de adrese MAC pentru o interfață, cadrele primite pe aceasta cu adresa MAC sursă necunoscută vor fi aruncate. Nu are loc o informare a faptului că a avut loc o încălcare a regulilor de securitate.
- Restrict: La fel ca modul "protect", cu diferența că are loc o informare a faptului că a avut loc o încălcare a regulilor de securitate.
- Shutdown: Acesta este modul implicit. În cazul apariției unei încălcări a regulilor de securitate, interfața în cauză este trecută imediat în modul "error-disabled". În plus, are loc o și informare a faptului că a avut loc o încălcare a regulilor de securitate. Readucerea interfeței în starea normală se face prin oprirea ("shutdown") și repornirea acesteia ("no shutdown").

## Configurare Dynamic Port Security



```
Switch# configure terminal
Switch(config)# interface fastethernet 0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# end
```

Cu ajutorul port-security pe un switch se pot specifica adresele MAC ce sunt permise pe fiecare port sau acțiuni specifice când o adresă MAC neautorizată încearcă să se conecteze pe acel port.

Pentru a activa port-security se intră în modul interfeței dorite folosind comanda **interface fastethernet id\_interfață**, se trece portul în modul access folosind comanda **switchport mode access** și se introduce comanda **switchport port-security**.

Adresele MAC permise pe fiecare port pot fi specificate în 3 moduri:

1. Static, configurate manual folosind comanda **switchport port-security mac-address adresa\_mac**.
2. Dinamic, adrese învățate automat de switch ce sunt salvate numai în tabela de adresare. Acestea se șterg la resetarea switch-ului.
3. Sticky, adresele sunt învățate automat de switch, iar apoi acestea pot fi salvate în startup-config.

## Configurare Sticky Port Security

```
Switch# configure terminal
Switch(config)# interface fastethernet 0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security
maximum 50
Switch(config-if)# switchport port-security
mac-address sticky
Switch(config-if)# end
```

Pentru a configura port security de tip sticky, primul pas constă în trecerea interfeței în mod access folosind comanda **switchport mode access** urmată de comanda de activare a port security **switchport port-security**.

Pentru a specifica numărul maxim de adrese ce vor fi învățate pe acest port vom utiliza comanda **switchport port-security maximum număr\_de\_adrese**.

Pentru a activa port security de tip sticky folosim comanda **switchport port-security mac-address sticky**. Odată executată această comandă, switch-ul va învăța primele 50 de adrese ce se vor conecta pe acel port, numai acelea având access. Pentru a reține acele adrese este necesară salvarea configurației curente în startup-config.

## Rezumat

- Ethernet
- Procesul de switching
- Configurări de bază



1. Ce reprezintă un atac de tip Denial of Service?
2. Care este tipul de switching folosit pe echipamentele Cisco actuale?
3. Ce echipamente delimitează domeniile de coliziune?
4. La ce nivel al stivei OSI operează protocolul CSMA/CD?
5. Din câți biți este formată adresa MAC?