

SOCKET-URI

- comunicare între procese de pe calculatoare diferite (se poate și din același calculator)
- model client - server
- pt. a comunica în rețea se folosesc 2 tipuri de protocole
 1. UDP (User Datagram Protocol)
 - ↳ permite transferul fără conexiune a informațiilor
 2. TCP (Transmission Control Protocol)
 - ↳ permite transferul prin conexiune a informațiilor, este de încredere
- se pot folosi la:
 - transfer de date
 - comunicare în timp real
 - descărcare de fișieri
 - chat
 - jocuri online

TIPURI DE SOCKET-URI

Socket stream

- serviciu orientat către conexiune
- date receptate în ordinea transmisiei
- protocolul TCP
- analogie aparat telefonic

Socket datagram

- serviciu fără conexiune
- nu garantează receptarea datelor
- datele pot ajunge în altă ordine decât cea în care au fost transm.
- protocolul UDP
- analogie: cutia poștală

SOCKET-URI

- comunicare între procese de pe calculatoare diferite (se poate și din același calculator)
- model client - server
- pt. a comunica în rețea se folosesc 2 tipuri de protocole
 1. UDP (User Datagram Protocol)
 - ↳ permite transferul fără conexiune a informațiilor
 2. TCP (Transmission Control Protocol)
 - ↳ permite transferul prin conexiune a informațiilor, este de încredere
- se pot folosi la:
 - transfer de date
 - comunicare în timp real
 - descărcare de fișiere
 - chat
 - jocuri online

TIPURI DE SOCKET-URI

Socket stream

- serviciu orientat către conexiune
- date receptate în ordinea transmisiei
- protocolul TCP
- analogie aparat telefonic

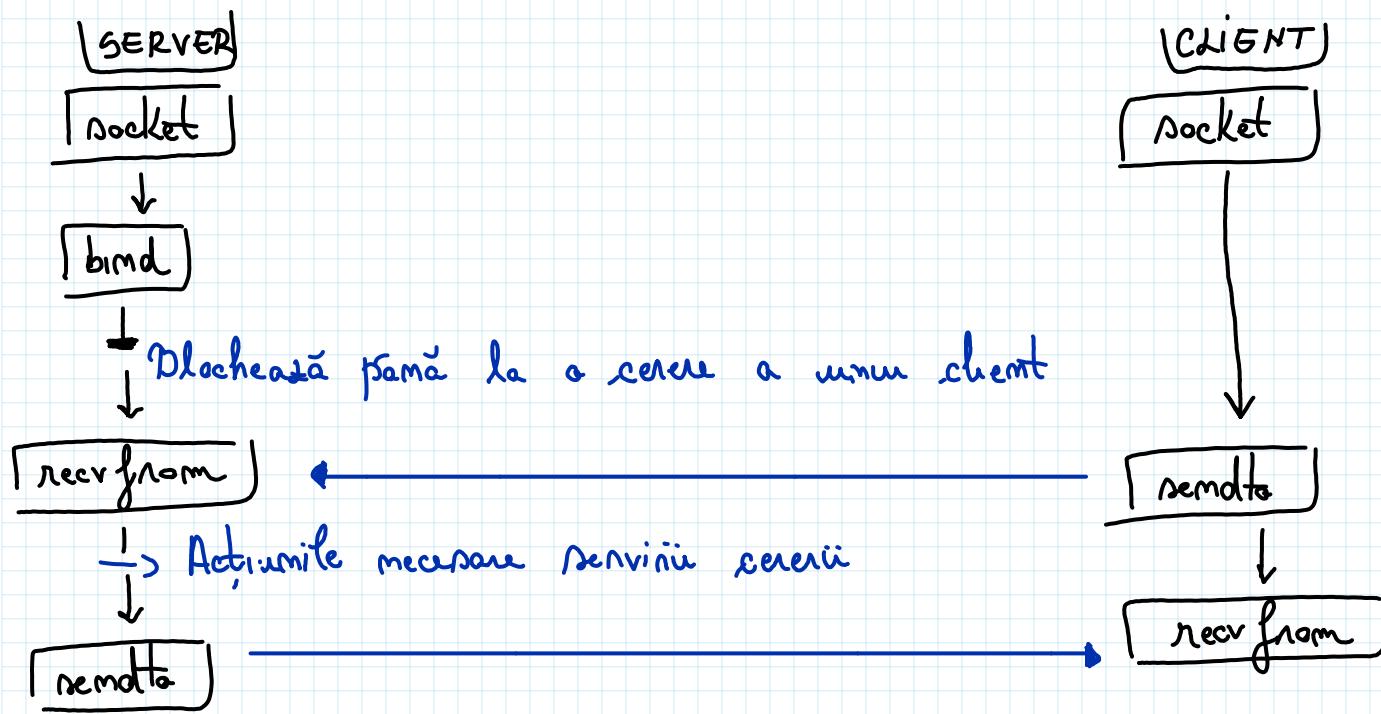
Socket datagram

- serviciu fără conexiune
- nu garantează receptarea datelor
- datele pot ajunge în altă ordine decât cea în care au fost transm.
- protocolul UDP
- analogie: cutia poștală

Pentru a stabili o conexiune folosind socket-uri, fiecare dispozitiv sau proces are o adresă IP în un port asociat. Adresa IP identifică dispozitivul, în timp ce portul indică un serviciu sau o aplicație.

ex IP → clădirea
port → camera

SOCKET DATAGRAM



Apeluri sistem

ip = '0 0 0 0'
port = 8888

s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

s == -1 => fail

s.bind((ip, port))

mesaj = "Salut"

s.sendto(mesaj, (ip, port))

data, clientAdres = s.recvfrom(buffer)

datele primite de la client (octete) de la tuplu (ip, port) octeti

BUFFER

= zonă de memorie temporară utilizată pt memorarea și manipularea datelor

- discul, rețeaua, dispozitive hardware

poate avea mai multe adrese IP
nu trebuie să fie ambele comunicări
căruia IP să poată căuta

are un port efemer
initială comunicarea
tbd. să fie IP
portul renunțului

SOCKET STREAM

SERVER



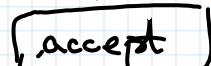
creare socket



port la care serverul așteaptă să fie contactat

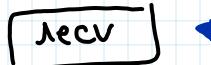


indica disponibilitatea de a primi cereri de conexiune pt
socketul cu descriptorul port; parametrul specific nr. maxim de solicitații care pot fi
blocate în timp ca se așteaptă ca serverul să le accepte



Blochează până la o cerere a unui client

STABILIRE CONEXIUNE (IP + PORT)



DATE PT CERERE

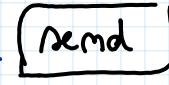
Actiunile necesare serverului



DATE DE RASPUNS



connect



recv

Apeluri sistem

IP = '0 0 0 0'

port = 8888

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

s == -1 → fail

s.bind((ip, port))

s.listen(n)

s.accept((ip, port))

clientSocket, clientAddr = s.accept()

mai tip de socket care
reprezintă conexiunea
formată cu clientul

AF_INET

↳ Address Family Internet

PF_INET

↳ Protocol Family Internet

→ indică utilizarea adn.

IPv4 în comunicarea
cu socket-uri

Când eșuează socketul?

- 1 Portul sau IP-ul e deja folosit
- 2 Lipsa permisiunilor
 - nu suntem portari sau IP
 - port < 1024 nu este folosită și poate fi blocată de rutare
3. Rutarea prea multor conexiuni simultană
 - depășirea limitelor de nerunză ale sistemului (limită de descriptori de fizici)
- 4 Setările imcorecte, ale firewall
 - se blochează conexiunea
- 5 Probleme de netea
 - server indisponibil
- 6 Deactivarea protocolului necesar
 - ex: Încercarea utilizării unui socket IPv6 pe un port IPv4
- 7 Erori de memorie sau depășirea bufferului

MEDIU DE TRANSMISIE

1. Fibre optice

- cablu coaxial
- cablu UTP (Unshielded Twisted Pair)

2. Fibra optică

- Single mode - distanțe lungi și viteză mare
 - un singur mod de propagare
- Multimode - distanțe mai scurte și în năelele locale
 - mai multă moduri de propagare

3. Wireless Media

- unde radio
- infraroșu (lumina de vedere directă)

4. Satellite Communication

- sateliți geostationari
 - înălțime fixă deasupra Pământului
 - comunicare la nivel global (ex. tv)
- sateliți non-geostationari
 - se află pe orbită
 - comunicare mobilă și internet

Cabluri

Cablu direct - dispozitive diferențiate

- ,alb portocaliu
- portocaliu
- alb verde
- albăstru
- alb ,albăstru
- verde
- alb maro
- maro

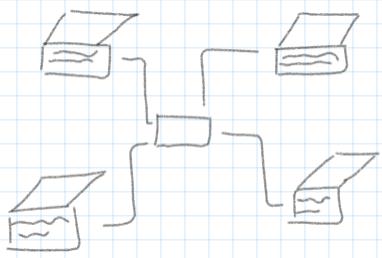
Cablu cross-over (se schimbă portocaliu cu verde) - dispozitive similare

- ,alb verde
- verde
- alb portocaliu
- albăstru
- alb ,albăstru
- portocaliu
- alb maro
- maro

TOPOLOGII

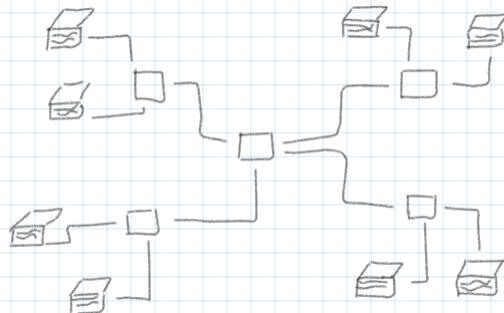
1 Star topology

- ↳ toate dispozitivele sunt conectate punct-um mod central (comutator / hub)
- ↳ dispozitivele sunt conectate în mod ind. prin modul central



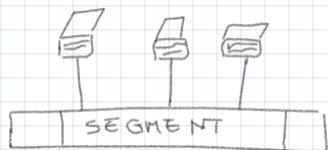
2 Extended star topology

- ↳ se folosește un dispozitiv central care e conectat la mai multe alte dispozitive



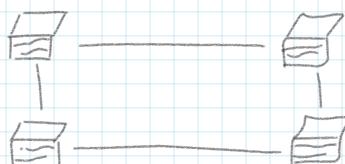
* 3 Bus Topology

- ↳ toate dispozitivele sunt conectate la un cablu comun
- ↳ datele de la un dispozitiv la altul sunt transmise prin magistrală și fiecare stație decide dacă să proceseze sau să ignore datele destinate altor stații



4 Ring Topology

- ↳ datele circulă de la o stație la alta pînă ajunge la destinație



NETWORK

- conexiune de dispozitive sau sisteme interconectate care permit schimburi de informații, date sau resurse
- poate fi fizică (componente hardware reale, calculatoare, switch-ură, rute, cabluri de rețea, sisteme Wi-Fi, imprimante, hub-ură)
- poate fi logică (modul în care dispozitivele comunică, deosebit de important cum se conectează: IP, subretele, tabele de routare, regulile de firewall, protocoale de comunicare)
- cuprinde: calculatoare, telefoane, servere, imprimante, rute

TIPI DE REȚELE

1. LAN (Local Area Network)

- rețea locală, restrânsă dintr-o casă / birou / clădire / campus
- dispozitivele din același locație se pot conecta și să partajeze informații
- adresa sunt gestionați de un router sau un switch pe o singură direcție

2. WAN (Wide Area Network)

- acoperă o zonă mult mai extinsă (orăș / ţară / planetă)
- mai lent decât LAN
- se bazează pe infrastructura publică de telecomunicări: linii telefoniice, cabluri de fibră optică, sateliți sau conexiuni de internet
- internetul în sine este un WAN

3. MAN (Metropolitan Area Network)

- LAN < MAN < WAN (zonă metropolitana, oraș, oraș mare)
- scop de a conecta dispozitive din diverse locații sau clădiri din același oraș
- viteza de transfer: LAN < MAN < WAN

4. PAN (Personal Area Network)

5. WLAN (Wireless Local Area Network)

PROTOCOALE DE COMUNICARE

1. TCP/IP (Transmission Control Protocol / Internet Protocol)

- asigură transmisarea fiabilă a datelor în controlul fluxului
- rapid

2. UDP (User Datagram Protocol)

- mai simplu, mai puțin fiabil
- livrare neasigurată
- viteză redusă

WWW

(World wide Web)

= rețea globală de informații
accesare și partajare
prin internet

3. HTTP (Hypertext Transport Protocol)

- transfer de pagini web și resurse asociate pe internet

4. HTTPS (Hypertext Transport / Protocol Secur)

- doar dacă datele sunt criptate

5. FTP (File Transfer Protocol)

- transfer de fișiere între calculatoare

6. SMTP (Simple Mail Transfer Protocol)

- transmitere și receptare de mailuri

7. POP3 (Post Office Protocol, vrs. 3) și IMAP (Internet Mail Access Prot.)

- folosita pentru accesarea și stocarea mailurilor
- POP3. descarcă mail-ul pe dispozitivul curent
- IMAP. le păstrează pe server și permite acceseul de pe mai multe dispozitive

8. DNS (Domain Name System)

- asociază numele de domeniu (ex: www exemplu.com) cu IP-ul

9. DHCP (Dynamic Host Configuration Protocol)

- atribuie automat adrese ip și alte informații de configurație a rețelei dispozitivelor care se conectează la o rețea)

10. SNMP (Simple Network Management Protocol)

- monitorizarea și gestionarea dispozitivelor de rețea (router, switch-uri, servere)
- permite administratorului să monitorizeze starea dispozitivelor și să facă modificări

11. SSH (Secure Shell)

- conexiunea securizată și criptată între 2 dispozitive

Poșta electronică (e-mail)

- = serviciu de trimisere / primire mesaje prin intermediul internetului
 - ↳ ex. Google, Yahoo, Microsoft etc
- utilizatorul primește și adresa de e-mail său (poate trimite mail unic în lume, independent de serviciu)
- se utilizează **SMTP** (Simple Mail Transfer Protocol)
 - ↳ utilizarea **TCP**
- beneficiu viteză
 - insanită de utilizare
 - cost redus
- dezavantaje spam
 - phishing
 - riscul de a avea emailul compromis

SPF (Sender Policy Framework)

- tehnica de autentificare pentru e-mail care permite receptorului să verifice autenticitatea proprietarului de poștă electronică
- permite proprietarilor de domeniu să specifice serverele de p.e. care sunt autorizate să trimită mesaje de e-mail în numele domeniului lor
- poate reduce spamul și atașurile
- receiverul scrie trimite DNS query-uri pentru a face verificări, dar e limitat la 10
 - ⇒ > 10 query-uri → !
 - ↳ există "extensiuni" pentru o mai bună funcționare

Protocol

- set de reguli pe care trebuie să le respecte 2 parteneri care comunică
- și **RFC** (Request For Comments). descriu de ce trebuie să facă comunitatea implementator, de client și sunul de server

Tipuri de trafic

- Unicast**
- comunicare 1:1 emițător - receptor (deobicei din același LAN)
 - conexiune TCP clasica

Broadcast - metodă de transmitere a informațiilor către toate disponibilele dintr-un LAN

- comunicare 1:tot
- servicii ARP, RARP, DHCP (prin UDP)

↳ nu se poate prim TCP (ar fi mai mult pe același descriptor de socket)

Multicast - comunicare 1:n (nu primește totuști mesajul)

- ex. 2 peis dintr-un LAN intră pe un ntb → 2x trafic unicat între device și server
- ABCD
 - ↳ A = maxim 223
 - ↳ → ABCD validă

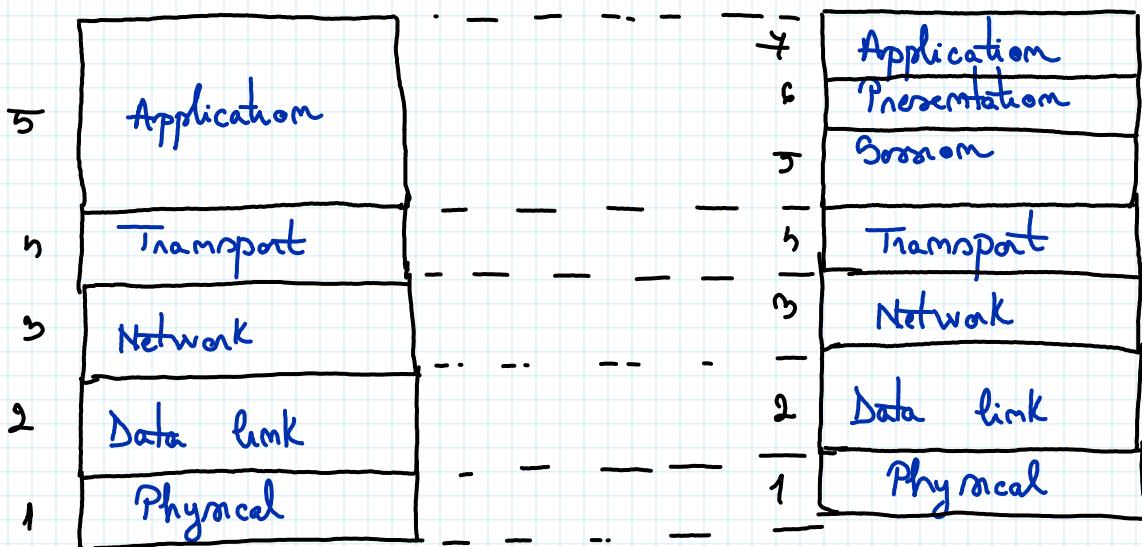
Anycast - comunicare 1:cel puțin 1 din mai multe

- ex. într-un LAN sunt mai multe servise DHCP. e important ca cel puțin 1 să răsp. cererii unui client

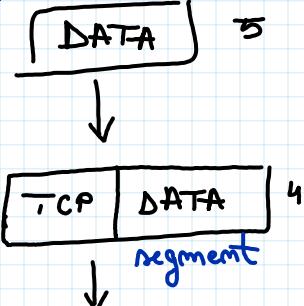
FF FF FF FF FF FF

= MAC de broadcast
 - utilizat pentru comunicarea cu toate echipamentele dintr-un LAN

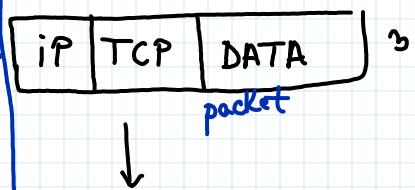
TCP / IP model



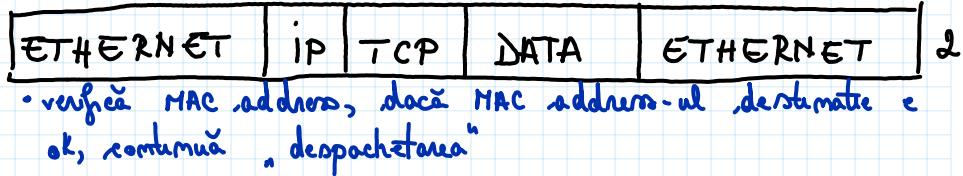
DATA 5



- se adaugă protocolul
- pt fiecare protocol se trimiț informații specifice $TCP = IP + port$



- verifică dacă ip dest e ok, sunt „despachetate” (procesarea)

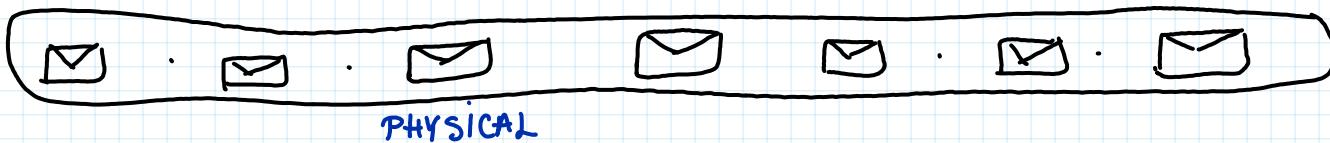


- verifică MAC address, dacă MAC address-ul destinatului e ok, continuă „despachetarea”



destination and source MAC address

error check and make sure the data has been received correctly



OSI model

OSI (Open System Interconnection) este un cadru conceptual utilizat pentru a descrie în întregime funcționalitatea rețelelor.

<p>Acesta e cel mai apropiat strat de utilizator. Au loc interacțiunile cu aplicația: navigare web, clientul de e-mail, server web, etc.</p>	<p>SMTP, FTP, Telnet</p>	<p>Application (7)</p>
<p>Se ocupă cu formatarea datelor și le face compatibile cu dispozitivele în aplicările din rețea. Comprimare, criptare și alte operații de prelucrare a datelor.</p>	<p>Format Data, Encryption</p>	<p>Presentation (6)</p>
<p>Stabilire, menținere și închidere semnale de comunicare între dispozitive</p>	<p>Start & Stop Session</p>	<p>Session (5)</p>
<p>Asigură comunicarea fizică și controlul flexibil între dispozitive (protocole)</p>	<p>TCP, UDP, Port Numbers</p>	<p>Transport (4)</p>
<p>Routarea datelor între rețele dif. Utilizarea adrese IP și a unui pește de către destinație corectă.</p>	<p>IP Address, Routers</p>	<p>Network (3)</p>
<p>Transmiterea datelor între dispozitive la secțiuni LAN și adăugarea MAC addresses și se efectuează verificarea de erori și asigurarea transmiterii fizice.</p>	<p>Mac Addresses, Switches</p>	<p>Data link (2)</p>
<p>Componente fizice ale rețelei (cablu, conector, semințe electrice sau optice) Se ocupă cu transmiterea bruto a datelor.</p>	<p>Cable, Network Interface, Cards, Hubs</p>	<p>Physical (1)</p>



1	Application
2	Presentation
3	Session
4	Transport
5	Network
6	Data link
7	Physical

Prin incapsulare, pachetul este "îmvelit" de primul dispozitiv în traius spre al doilea, care prelucră pachetul în ordine inversă (pe rând fiecare nivel)

4	Application
5	Presentation
6	Session
7	Transport
8	Network
9	Data link
1	Physical

Port numbers

adresa URL \rightarrow adresa IP $\xrightarrow{\hspace{1cm}}$ server

\downarrow
se adaugă portul
specific funcției
de aplicatie

ex: 10.54.16.1:80

dst port: 80 (HTTP)

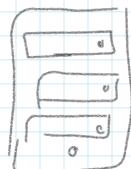
src port: 63139 (generat random)

Adresa IP date către computer
Port date către aplicatie



dst port: 80
src port: 63139

10.54.16.1:80
10.8.27.5:63139



dst port: 63139
src port: 80

\rightarrow specifică inclusiv tabel

Porturi

0 - 1023 \rightarrow well known

1024 - 49151 \rightarrow registered

49152 - 65535 \rightarrow dynamically assigned

Adresa IP

- identificator unic assignat fiecărui dispozitiv conectat la o rețea de calcul

IPv4

ex: 192. 168. 32. 152

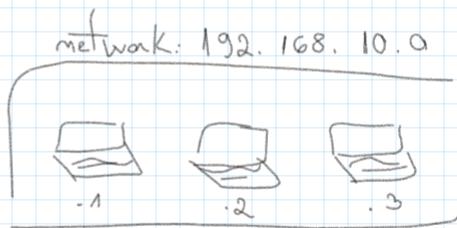
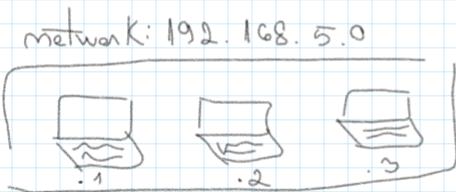
(adresa)

(nr. caser)

- împărțită în 2 prima parte NETWORK, și două părți: HOST

Subnet mask 255.255.255.0

ex:



192.168.5.3 → host
255.255.255.0

Class

Public

A 1000 - 126.255.255.255

Subnet 255.0.0.0 → hosts: 16777216 hosts

B

128.0.0.0 - 191.255.255.255

Subnet 255.255.0.0 → hosts: 65536

C

192.0.0.0 - 223.255.255.255

Subnet 255.255.255.0 → hosts: 256

PRIVATE

10.0.0.0 - 10.255.255.255

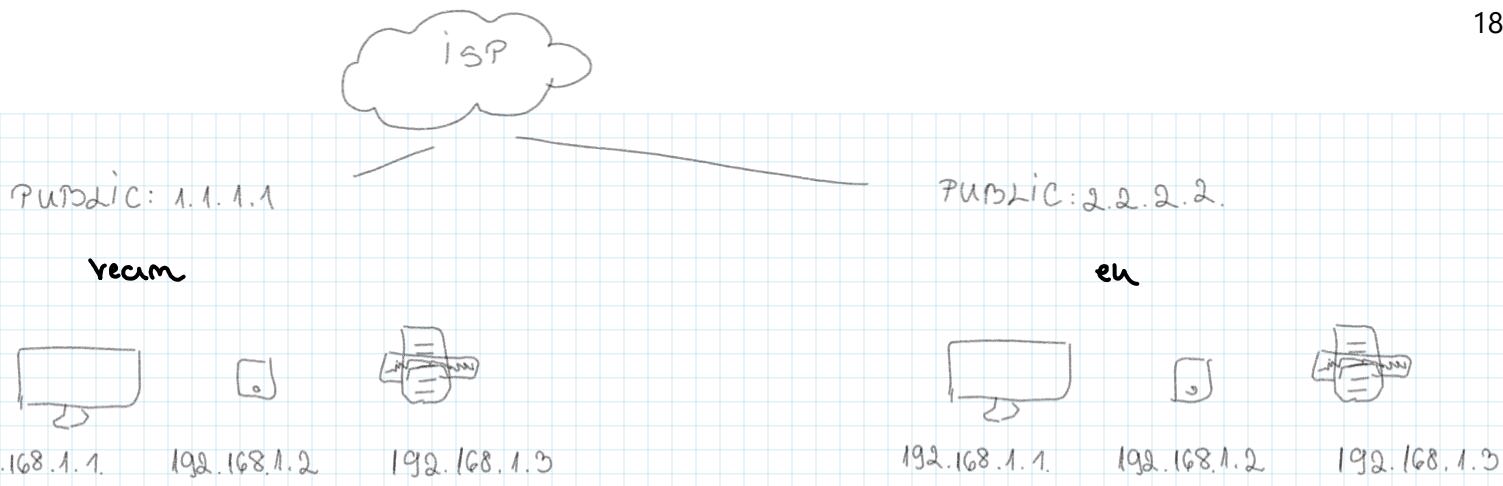
142.16.0.0 - 142.31.255.255

192.168.0.0 - 192.168.255.255

D - multicast addresses

E - experimental use

Public - doar prim internet, trebuie să fie unice



Adrese IP private (false)

- trebuie să fie ușoare să se transmită în propria rețea
(\hookrightarrow nu și în vecinul său) să aibă adrese IP
 - nu pot fi folosite pentru internet (să nu facă un ghicire de dupăcat)
- În general, la crearea contractului pentru internet primim și adresa IP
- avantaje permit economia de clase de addr. IP reale
securitatea
 - dezavantaje: trebuie SNAT ca să meangă internetul
nu se pot rupe servicii pe care să fie accesibile din alte
părți din internet fără DNAT
 - nu sunt mutabile

NAT nu înlocuiește adrese IP false cu ușoare reale
(se poate face cu false / real cu real)

Binary

ex: 192 168 0.1

1-on
0-off

10000000 10101000 00000000 00000001

$$192 - 128 = 64$$

128	65	52	16	8	5	2	1
0	0	0	0	0	0	0	0

Punem 1 în 0 astfel
încât , atunci , cînd
adunăm valoarele din
căsuarele la care am
pus 1 , să ne dea
mr. dorit. Începem
de la 750000 , cu cea
mai mare valoare.

$$168 - 128 = 40$$

$$1 - 1 = 0$$

$$h0 - 32 = 8$$

$$8 - 8 = 0$$

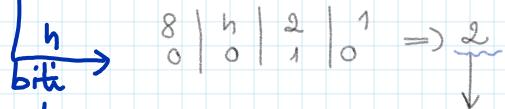
IPv6 Addresses

IPv6 - 32 bits (4 octets)

IPv6 - 128 bits (8 hexadecimals)

↳ Separati prim:

- format din caractere (cifre 0-9 sau litere A-F)



$$8 | h | 2 | 1 | 0 \Rightarrow 2$$

2001:0db8:0000:0000:a111:b222:c333:abcd

- nu se mai folosesc subnet mask

host

2001:0db8:0000:0000:a111:b222:c333:abcd / 64

network

primii 64 biti = network

când vede că e incomplet, se completează automat cu 0

2001:0db8::a111:b222:0:abcd

- înlocuiesc spațiul doar cu 0-vini
- pot fi folosiți și numărări hexadecimale nu ar fi să că înceapă de 0 să urmeze

pentru că nu se mai poate folosi numărul 0, să nu se completeze automat cu restul

Tipuri

1. Global Unicast

- ca în una publică v4 (pentru că sunt sute mii de rețele, nu mai avem nevoie de adrese private)

2001:0db8:0000:0000:a111:b222:c333:abcd

global prefix

subnet

host / interface ID

(64 biti)

- min. 48 biti

2000::/3 Publicly routable (începe cu 2 sau cu 3)

2. Unique Local

- ca în una privată v4

F000::/7 Routable in the LAN (începe cu F, urmat de C sau D)

A	10
B	11
C	12
D	13
E	14
F	15

3 Link Local

- comunică între nicio rețea și un rețele
- 169.254.x.x

FE80::/10 Not routable (începe cu FE)

4. Multicast

- se trimite unui grup de dispozitive care așteaptă în mod special același adresa (broadcast)

FF00::/8 Addresses for groups (începe cu FF)

5. Anycast

- angajarea unei adrese IP mai multor dispozitive
- informațiile sunt trimise celor mai apropiat dispozitiv cu adresa respectivă

2000::/5

MAC Addresses (Media Access Control)

- identificator unic assignat unei interfețe de rețea (NIC) (Network Interface Card)
- adrese fixe
- nu pot fi schimbate
- coduri (48 biti)

5. APPLICATION

4. TRANSPORT

3. NETWORK

2. DATA LINK

1. PHYSICAL

→ tehnologie pentru conectarea dispozitivelor la LAN

veloare unică atribuită de vendor

08 - 00 - 27 - EC - 10 - 61

OUI (Vendor)
(organizationally unique
identification)

Vendor

= codul de identificare
al furnizorului

- compania / producătorul
dispozitivului

Tipuri

1. UNICAST

↳ particulară, unică (ex. de răs)

2. MULTICAST

01 - 00 - 5E - 00 - 00 - 05

multicast prefix

- pentru aplicație / protocol
- se trimite tuturor dispozitivelor,elor care sunt conectate la același lan / segment

3. BROADCAST

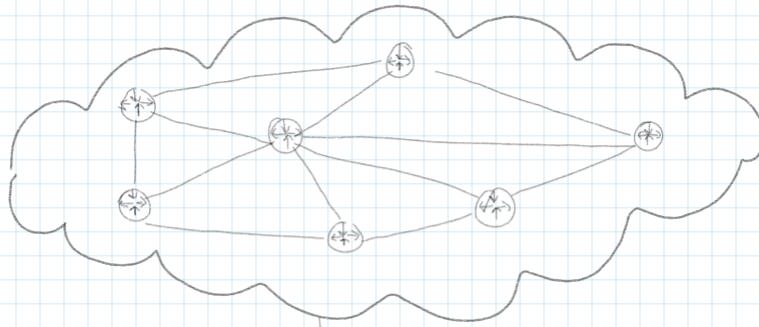
FF - FF - FF - FF - FF - FF

↳ se trimite tuturor dispozitivelor dintr-o rețea

Moduli de rețea

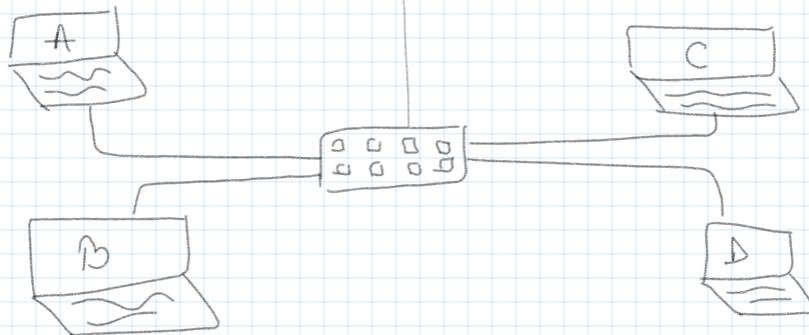
dimux / Apple 08 00 27: EC: 10 . G1
 Microsoft: 08 - 00 - 27 - EC - 10 - G1
 Cisco 08 00 27 EC 10 G1

Router



Layer 3
 IP Addresses
 → global communication

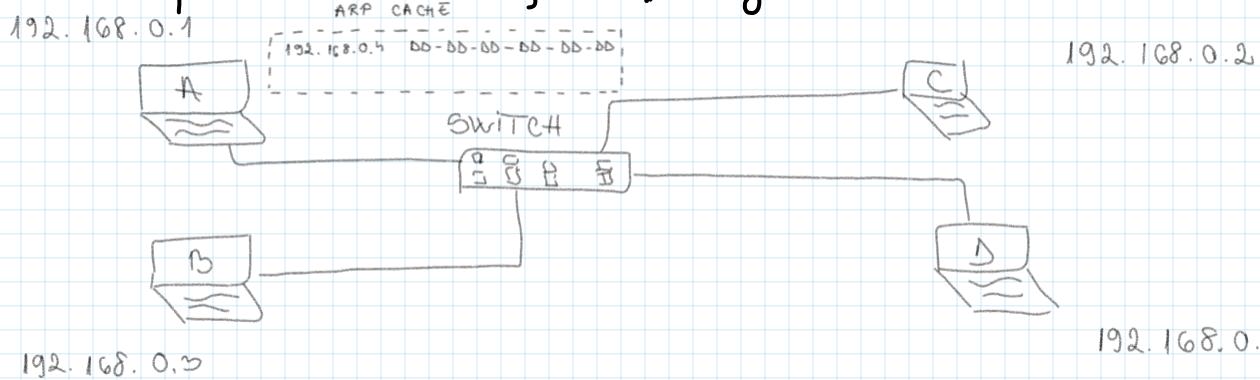
Switch



Layer 2
 MAC Addresses
 → local communication

ARP Address Resolution Protocol

→ descoperă adresa MAC și le „transformă” în adresa IP



A vrea să comunice cu D

Trimite un mesaj broadcast să vadă unde e IP 192.168.0.4

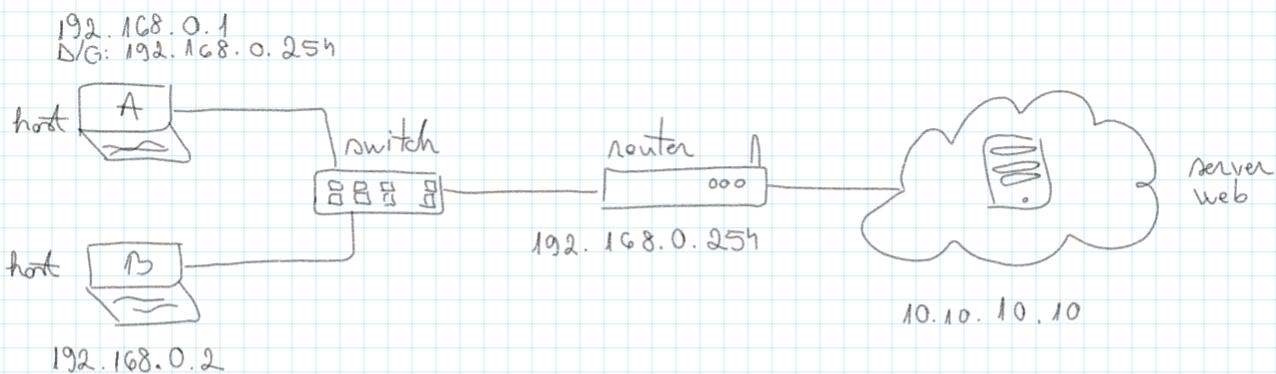
B și C nu au înțelesă

D îi trimite înapoi lui A adresa MAC

Se va salva în ARP cache pentru viitoare utilizări

Switch

- layer 2
- doar adr. MAC



A vrea să comunice cu serverul web

A verifică IP-ul s.w - ului și vede că nu face parte din aceeași rețea

A trimite cerere ARP în rețea: B negășă, routerul îi răspunde cu adresa MAC (ca A să poată ieși din rețea)

A acum A poate să trimită date cu MAC adresa-ului routerului, dar IP-ul serverului (mai departe se ocupă routerul).

Cererea ARP
se realizează
doar în
layer 2!

Default gateway = „ura” spre ieșirea din rețea (în funcție de IP)

ex: în acest caz, e routerul

RARP

(Reverse Address Resolution Protocol)

- protocol utilizat pt. atribuirea adreselor IP dispositivelor care nu pot stoca propriile adrese IP
- mod de funcționare
 - dispozitivul trimite adresa MAC în solicită una IP
 - un server RARP răspunde cu adresa cerută
- ⇒ află adresa IP pe baza adresei MAC
 - ↳ doar pe rețea o cunoaște
- se folosește de DHCP

VLAN

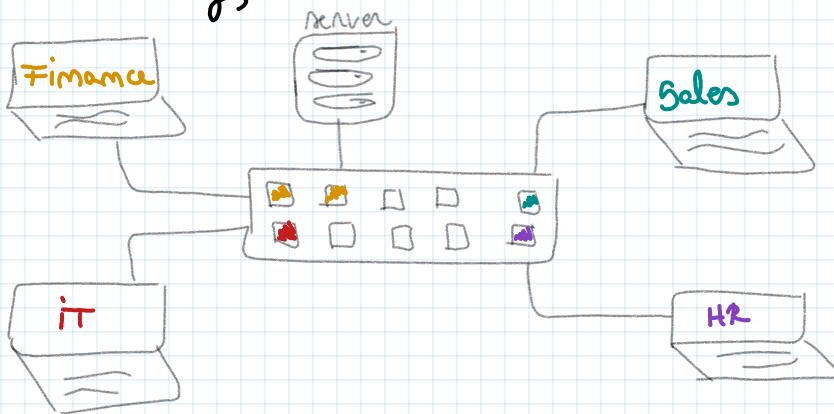
Virtual Local Area Network

- separă virtual LAN-urile

De ce nu folosim VLAN?

Broadcast traffic

- se separă virtual rețeaua
- traficul de date se comportă ca și cum ar fi împărțit fizic (adăugarea unor switch-uri / routare)
- se atribuie interfețe



În realitate nu folosesc numere, nu culori!

- se poate comunica doar în „interiorul” aceluiși VLAN
(finanțe și server pot comunica, păcă să fac parte din același VLAN)

Mod de funcționare

- VLAN initial VLAN1

↳ Toate interfețele pot comunica între ele

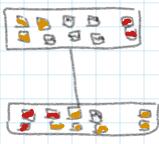


- Se pot adăuga maxim 4096 de VLAN-uri



- VLAN 1 (df)
- VLAN 10
- VLAN 20

- Putem să avem același VLAN-uri între mai multe switch-uri



→ necesită un trunk → tip special de interfață

pentru a

tag

! astăzi switch

Tag

- majoritatea dispozitivelor cu "stan" ce e un VLAN
↳ gestiunea de switch
- ⇒ comunicarea gestionată frame-uni



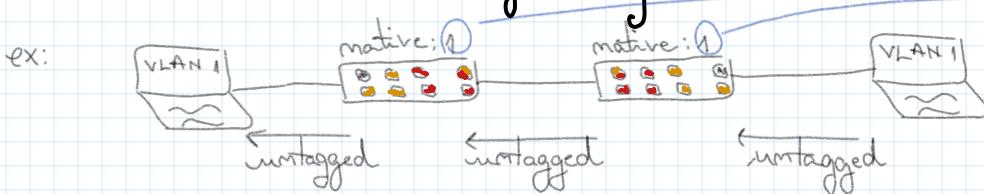
(preamble | sfld | dst. add | src. add | 802.1Q | type | data | fcs) → FRAME

- 5 octeți • TPID (tag protocol identifier)
↳ permitem să identifică frame-ul ca fiind 802.1q tagged
- TCI (tag control information)
 - 3 biti • 1 priorităție
 - 2. DEI (drop eligible indicator)
 - 3 id-ul VLAN-ului

Native VLANs

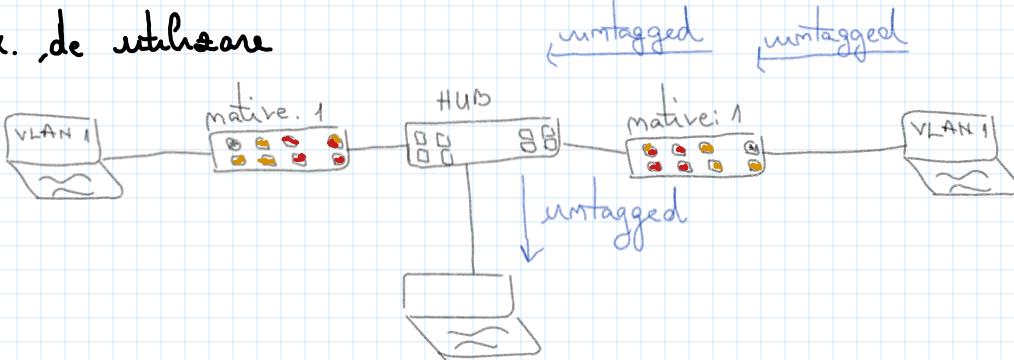
- trunk interface

- traversarea unui trunk fără tag



nu pot fi diferențiate, informația nu se mai adună la destinație

- ex. de utilizare



HUB

- nu pot scrie / citi tag-uri
- doar transmit frame-uri

STP Spanning Tree Protocol

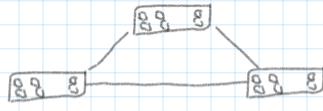
Tipuri de STP

- STP / 802.1D - original
- PVST+ - imbunătățire Cisco a STP prin adăugarea VLAN
- RSTP / 802.1w - imbunătățire STP cu o convergență mult mai rapidă
- Rapid PVST+ - imbunătățire Cisco a RSTP prin adăugarea VLAN

Utilizare

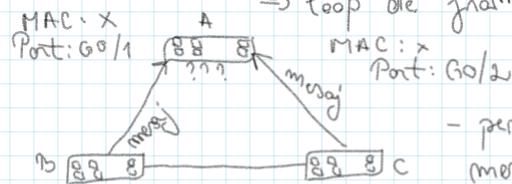
- prevene loop-urile, când sunt utilizate 2 sau mai multe switchuri

→ broadcast storm



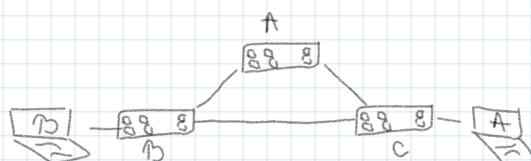
- de fiecare dată când se trim. un mesaj broadcast se trimite către toate switch. ⇒ loop de frame-uri

→ unitable MAC Address Tables



- pentru același mesaj de la switch. dif., se vor actualiza mac-tablele

→ duplicate frames



Host B → host A

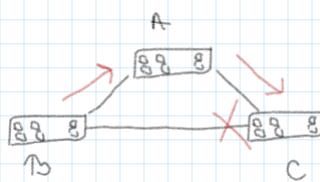
sw. B nu știe adresa lui h A

sw. C știe locația lui h A și îi trimite

sw. A știe că nu e pețntru el, deci trimite mai departe ⇒ ajunge la sw. C care îi trim. iar lui h A

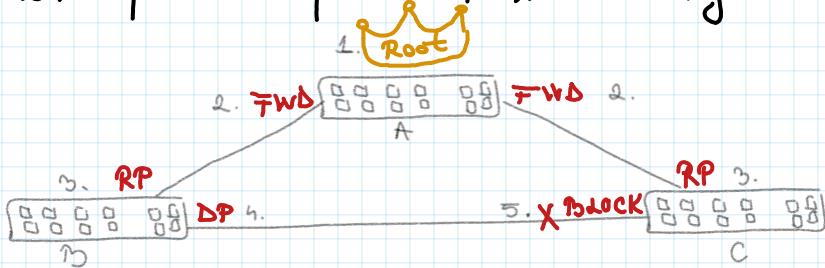
(= trimite la toată lumea)

- mod de prevenție: pentru a nu se creea loopuri, sw. care trimite informație va bloca anumite porturi



Mod de functionare

1. Elect a Root Bridge
2. Place root interfaces into a Forwarding state
3. Each non-root selects its Root port
4. Remaining links choose a Designated Port
5. All other ports are put into a Blocking state



Roles

Root Ports - the best port to reach the Root Bridge

Designated Ports - port with the best route to the Root Bridge on a link

Non-Designated Ports - all other ports that are in a blocking state

States

Disabled - a port that is shutdown

Blocking - a port that is blocking traffic

Listening - not forwarding traffic and not learning MAC addresses

Learning - not forwarding traffic but learning MAC addresses

Forwarding - sending and receiving traffic like normal

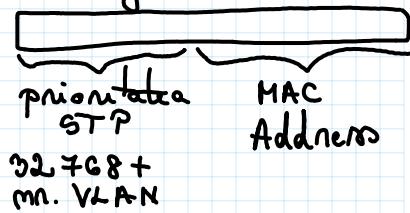
RYST

1 Root Bridge Election

- forward switch are use BPDUs

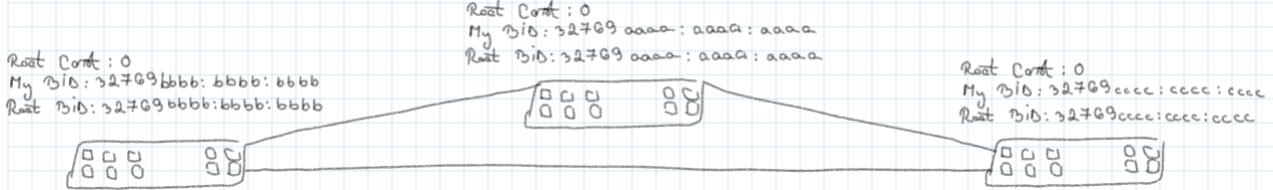
↳ root cost, route in local RIB

↳ Bridge ID

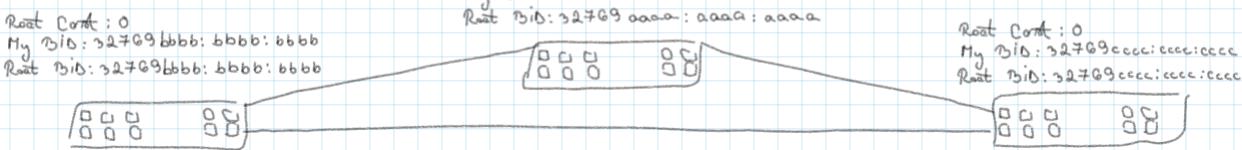


- devine root bridge switch-ul cu cel mai mic RID pe total

- la început, fiecare zw. re comandă rădăcina

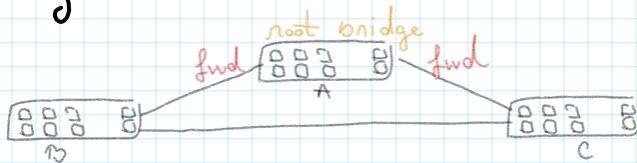


- apoi își trimit BPDU-urile între ele, iar cele care au BID mai mare re "conformă"



2. Root interface forwarding state

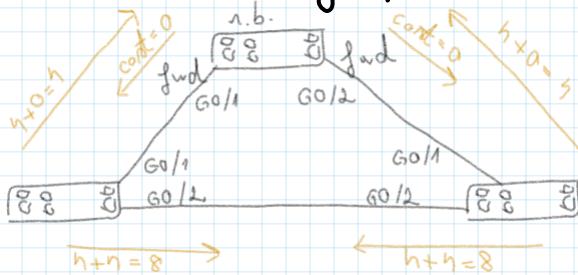
- toate porturile care se află în legătură cu rădăcina sunt în forwarding state



3 Non-roots choose the best path to the root bridge (reports)

- se bazează pe cantitatea porturilor,

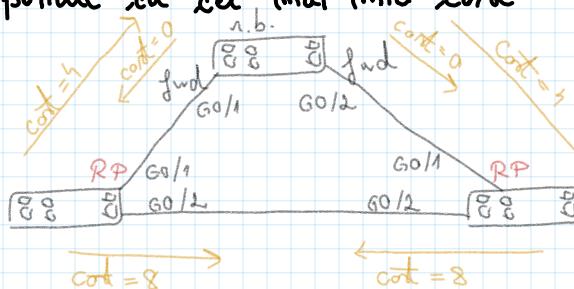
nr. porturilor care merg spre rădăcina



PORT COST		
Port Speed	Original	New
10 Mbps	100	2 000 000
100 Mbps	10	200 000
1 Gbps	4	20 000
10 Gbps	2	2000
100 Gbps	N/A	200
1 Tbps	N/A	20

- se alege portul cu cel mai mic cost

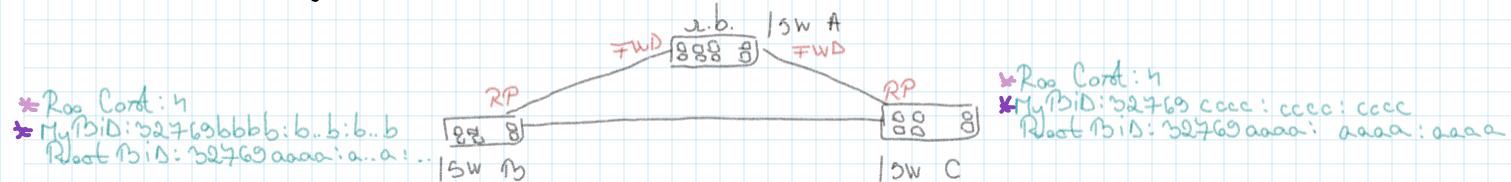
⇒



- dacă se fi făcut acelorași sort pe mai multe porturi, atunci se alege vecinul cu cel mai mic BiD - se verifică cea mai mică prioritate a portului
 - în caz de egalitate, se verifică cel mai mic nr. de port

4. Designated Ports

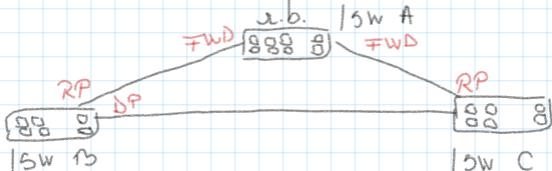
- se alege dintre cele care nu sunt repezis



- pară (se trece la un numătorul im case de egalitate).

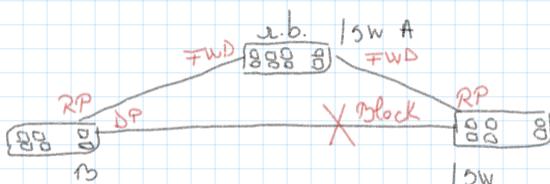
- se verifică cel mai mic cost *
- se verifică cel mai mic BiD *
- se verifică cel mai mic neighbor port priority
- se verifică cel mai mic neighbor port number

- în ex. de mai sus, port B devine demgated port



5. Blocking

- fiecare port care nu e rp (root port) sau dp (demgated port) este pus în blocking state



Timers - Default

Hello 2 sec → intervalul de timp în care RB creează și trimite mesajele hello (arașă reție toată "lumea" că funcțiile încă funcționează)

Max Age $10 \times \text{Hello}$ (20 sec.) → atât timp arată switchul pămă neactualizată că ceea ce nu e ok

Forward delay 15 sec → timp între linklayer state și learning state

STP states (ordinea efectuării)

Forwarding state → Blocking state → Listening state $\xrightarrow[rec]{15} Learning state \xrightarrow[rec]{15}$

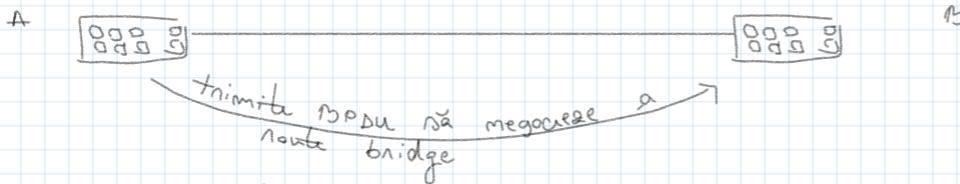
↳ durată multă → rezolvare rapidă spammig tree protocol

↳ în realitate, dacă avem un port conectat la un switch, portul va fi portocaliu către timp și abia apoi se face verde

Alt exemplu
Te conectezi la internet, nu merge, scoti cablul și îl bagi la loc (processul durează multă, tu îl întrenui și apoi tib. Nă începeă iar)

PortFast + BPDU Guard

STP - creat pentru evitarea loopurilor intre switch-uri



Dacă B nu are protectie, acceptă ceea cea = loop

Dacă B are BPDU guard enabled => B vede BPDU, realizează că e conectat la alt switch, blochează portul => Enabled

Comenzi

spanning-tree portfast

spanning-tree portfast default (apoi se dezactivează de pe porturile mediotite)

show spanning-tree summary (mă redem dacă portfast/bpdu e enabled/disabled)

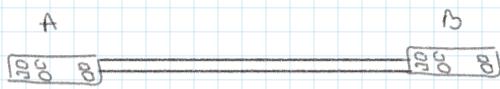
show spanning-tree interface fastEthernet 0/1 portfast (dacă portfast e activat pe interfață respectivă)

spanning-tree bpduguard enable

spanning-tree portfast bpduguard default

show running-config

Etherchannels



→ în mod normal, STP nu bloca unul dintre porturi



Se creează propria interfață logică și STP o primează direct în forwarding state

Se poate dubla capacitatea informațiilor. Dacă un cablu e necunoscut sau mai funcționează, se utilizează cel rămas.

Proprietăți

1. Se comportă ca o singură interfață

- permite statelor să "circule" în comunicație și în lipsa uneia dintre cabluri
- evitarea loopurilor (nu se trimită informația pe un cablu și nu se întoarce înapoi pe celălalt)

2. Pot fi până la 8 cabluri paralele

3. 3 metode de configurație statică, PAgP, LACP, recomandate

4. Facilitarea traficului de date

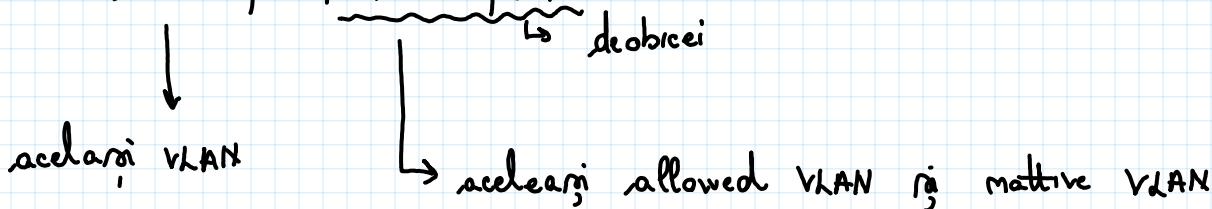
PAgP - Port Aggregation Protocol

LACP - Link Aggregation Control Protocol

Reguli pentru funcționare

Toate porturile în etherchannel - trebuie să aibă același

- duplex
- viteză
- acces port / trunk port



- STP interface settings (ex: port priority)

Keywords

0m / 0m	Static
Deminable / Demnable	PAgP
Deminable / Auto	PAgP
Active / Active	dACP
Active / Passive	LACP

retransmite información,
debe ser creada una
etherchannel

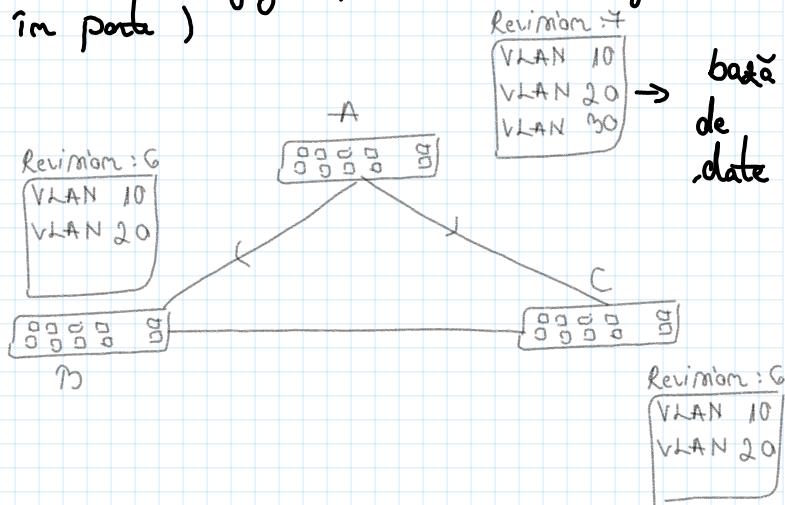
Channel Group = Port Channel = Etherchannel

VLAN Trunking Protocol (VTP)

- le permite switch-urilor să îmormătăceze configurațile VLAN (setul de trebuu introducute de mâmă la fiecare în portă)

Summary Advertisements

- se trimit automat la fiecare 5 minute
- nume VTP
- parola VTP
- revision nr.
- followers*



Subset Advertisements

- nume VTP
- toată informația VTP

Se fiecare dată când e modificată baza de date \Rightarrow newrev + 1
la fiecare 5 min se trimit de la fiecare sw. către toate restul summary advertisements

Se verifică nr. revizunii. Dacă un nr. are newrev mai mare, în funcție de followers* se trimit celalalte nr. subset advertisements (sabia acum se trimit informații despre VLAN - rămâne să fi fost trafic de date întrul)

Celalalte sw. își actualizează baza de date și se trimit summary adv

Moduri VTP

- Server - poate crea VLAN-uri
- trimit update-uri în adv. către bazele de date a VTP

- Client - nu poate crea VLAN-uri
- poate doar să trimită update-uri în adv. către bazele de date a VTP

- Transparent - poate crea doar VLAN-uri locale
- nu dă update-uri sau adv.
 - poate doar să trimită mai departe update-uri în adv. între alte sw, dar el le ignorează
 - nu are niciun impact asupra bazei de date VTP

Reguli

1. Link-urile trebuie să fie trunks
2. Toate switch-urile să aibă același VTP domain și revision

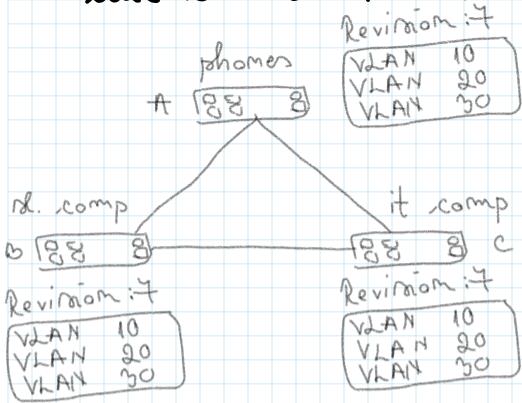
3. VTP password să fie la fel la toate
→ este optimă

I) nu se trimite
niciun mesaj pe
porturi de acces

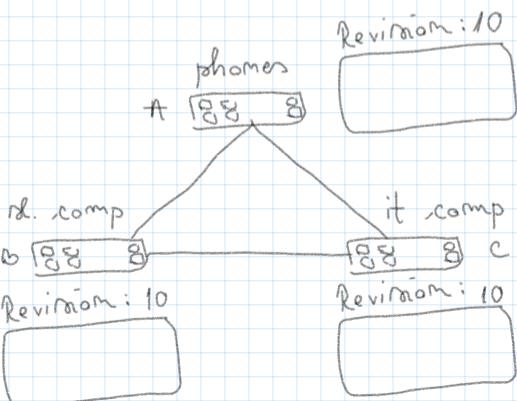
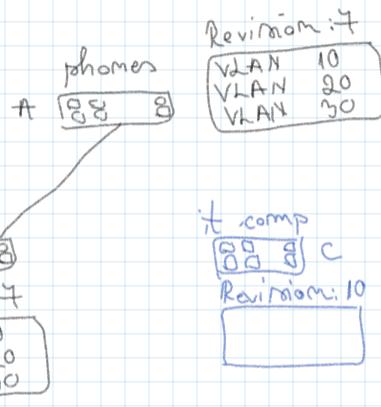
→ important pt. SW, ca să stie ce
mesaje să ascute și pe care
nu le ignore

DEZAVANTAJ

1. amintim că VTP-unile se actualizează după versiunea bazei de date cu cea mai mare valoare



deconectăm SW. C, facem experimente
pe el, și se adaugă la:



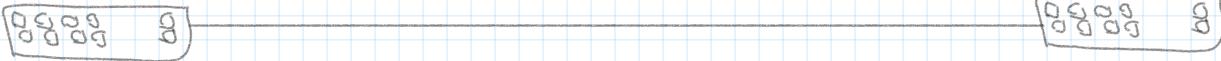
astfel, când conectăm înapoi
switch-urile, se vor actualiza.

Exemplu comenzi

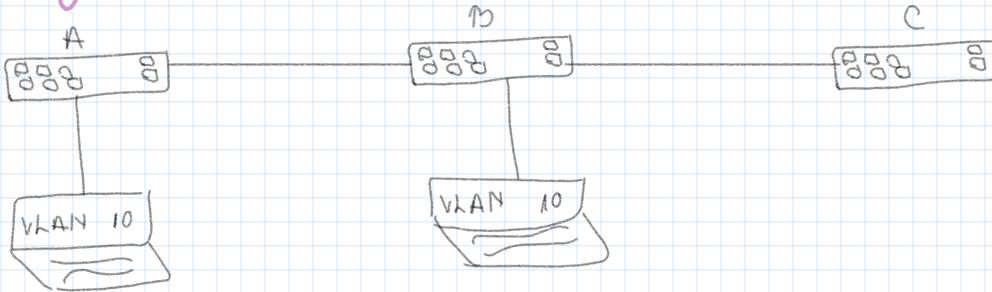
```
vtp mode server
vtp domain dama
vtp password dama
vtp pruning
interface fastEthernet 0/1
switchport trunk encapsulation dot1q
switchport mode trunk
```

enable
configure terminal

```
vtp mode client
vtp domain dama
vtp password dama
interface fastEthernet 0/1
switchport trunk encapsulation dot1q
switchport mode trunk
```



Pruning



- permite să „zică” pt ce VLAN -uri său ponturi

⇒ înloc ca A și B să îi trimitem informații anunță lun C (care le-ar ignora), C „spune” din start că nu acceptă VLAN 10 ⇒ A și B nu îl mai trimit lun C → salvare de rezurse

Rutare

= procesul prin care pachetele de date sunt redirecționate în rețele diferite

Tipuri

1. Rutare statică

- rute introduse manual de administrator \Rightarrow modificare și remunerare în cazul unor schimbări în rețea
 \rightarrow posibilitatea controlului strict a dimensiunii tabelelor de rutare

2. Rutare dinamică

- tabelele de rutare sunt construite automat cu ajutorul protocolelor de rutare: RIP, OSPF și BGP

permet noptenelor să comunice între ele și să schimbe informații despre rețeaua rețelei
 \Rightarrow noptenile pot lua decizii mai bune de rutare și pot adapta tabelele de rutare cînd se manifestă schimbări în rețea

Agregarea rutelor

- practică de grupare a unei trasee într-unul singur mai specific \Rightarrow reduce complexitatea în dimensiunile tabelelor de rutare

Rute implicite

- adresa IP a gateway-ului implicit

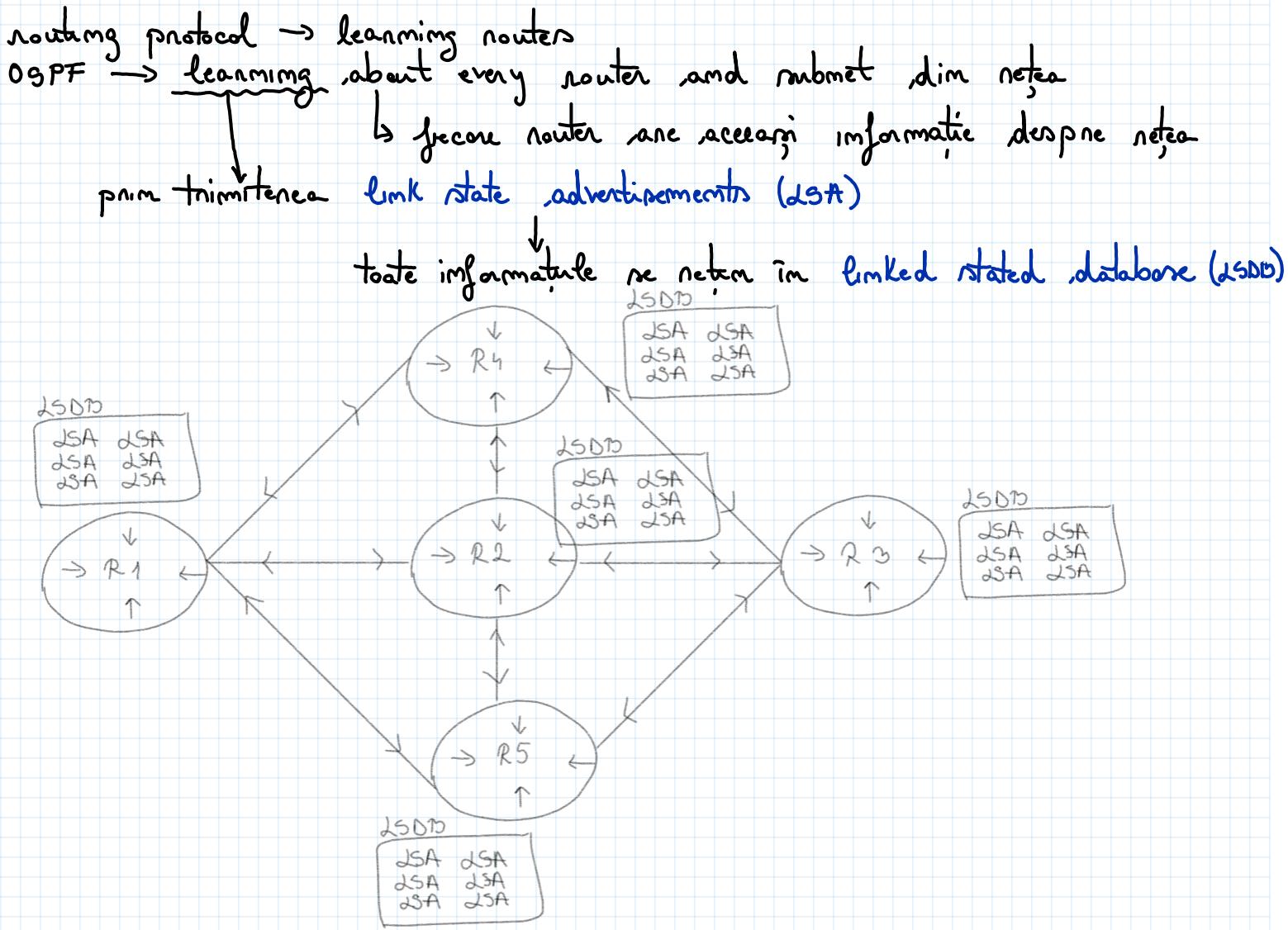
Tabelă de rutare = lista de rute disponibile peintru un router

- informații despre rețele disponibile, adrese IP și metrice (=costurile) fiecărui nod

\hookrightarrow minimă = comune și tot rutele necesare și conectarea rețelei la Internet / altă rețea

OSPF (Open Shortest Path First)

widely used and supported
IGP (Interior Gateway Protocol)
link-state routing protocol



Pasi

1. Become neighbours

- 2 rute dim sănătate rețea acceptă să creeze o relație de vecine (folosind protocoul OSPF)

2. Exchange database information

- vecini fac schimb de informații dim LSDB între ei

3. Choose the best routes

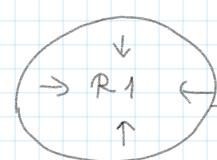
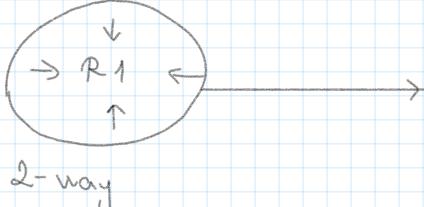
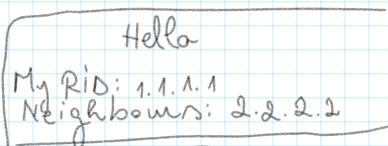
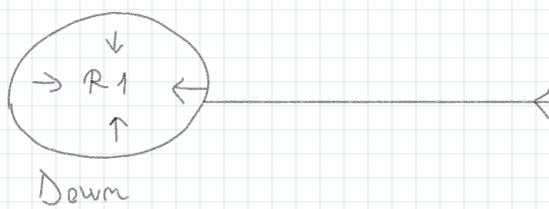
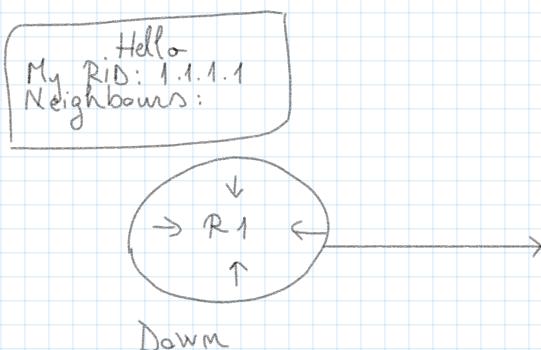
- fiecare router adaugă în routing table cele mai bune variante în funcție de informațiile dim LSDB

Router ID (RID)

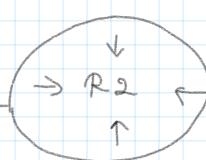
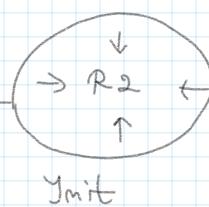
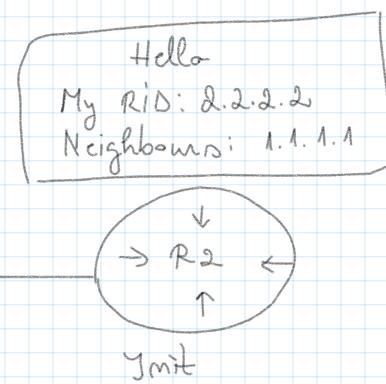
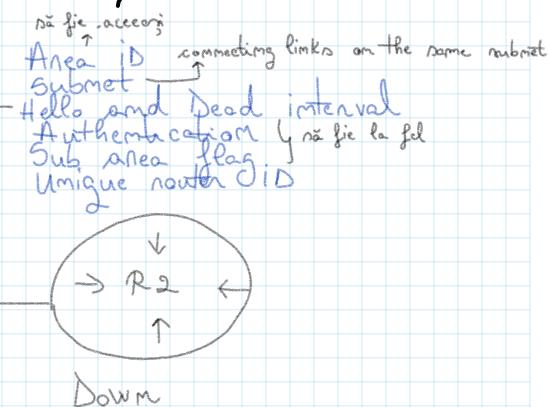
- mă. folosit pentru identificarea unui router individual
 - de forma unei adrese IPv4
 - se poate seta manual / automat

highest 'up' status loopback interface IP addn

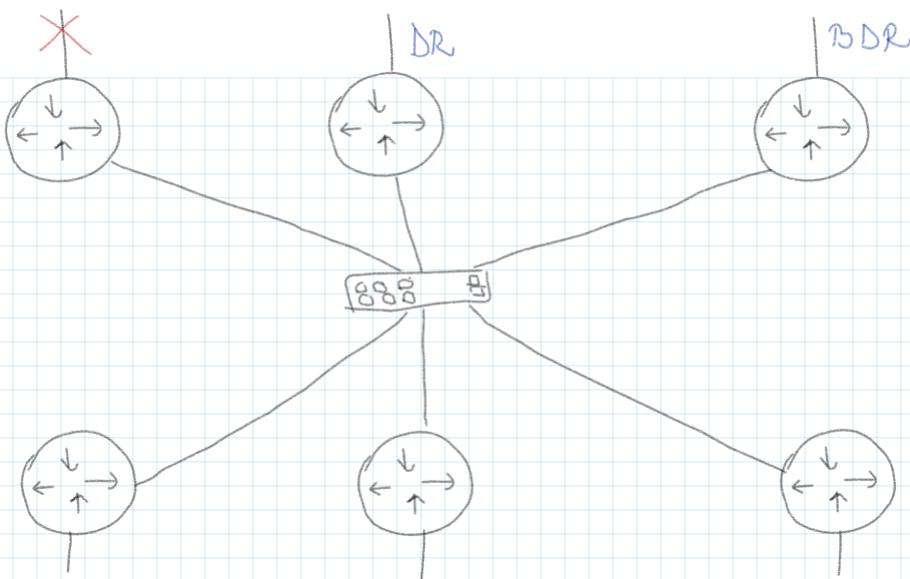
highest 'up' status non-loopback interface IP addr.



, acum sunt gata pentru schimb de informații



2-way



Designated router (DR)

- router care devine responsabil pt. gestionarea actualizării OSPF

Backup Designated Router (BDR)

- preia funcția de DR dacă aceasta este neașteptată

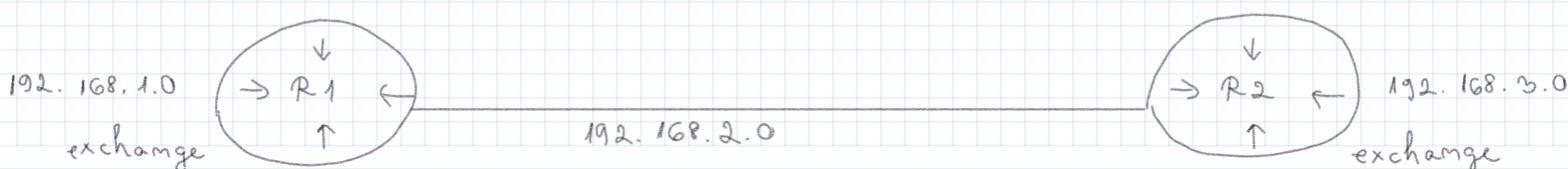
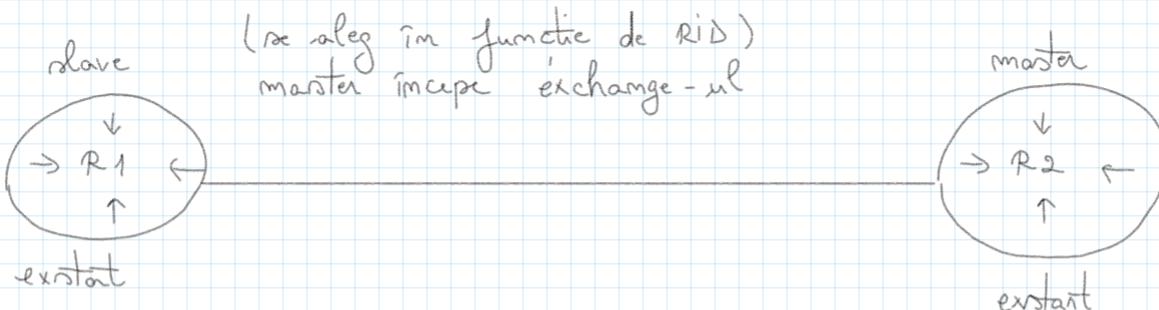
Pregătimem că un router estează 0 să se trimită modificării ca mesajele la toti vecinii și apoi de la fiecare la fiecare. \rightarrow avem nevoie de DR și BDR.

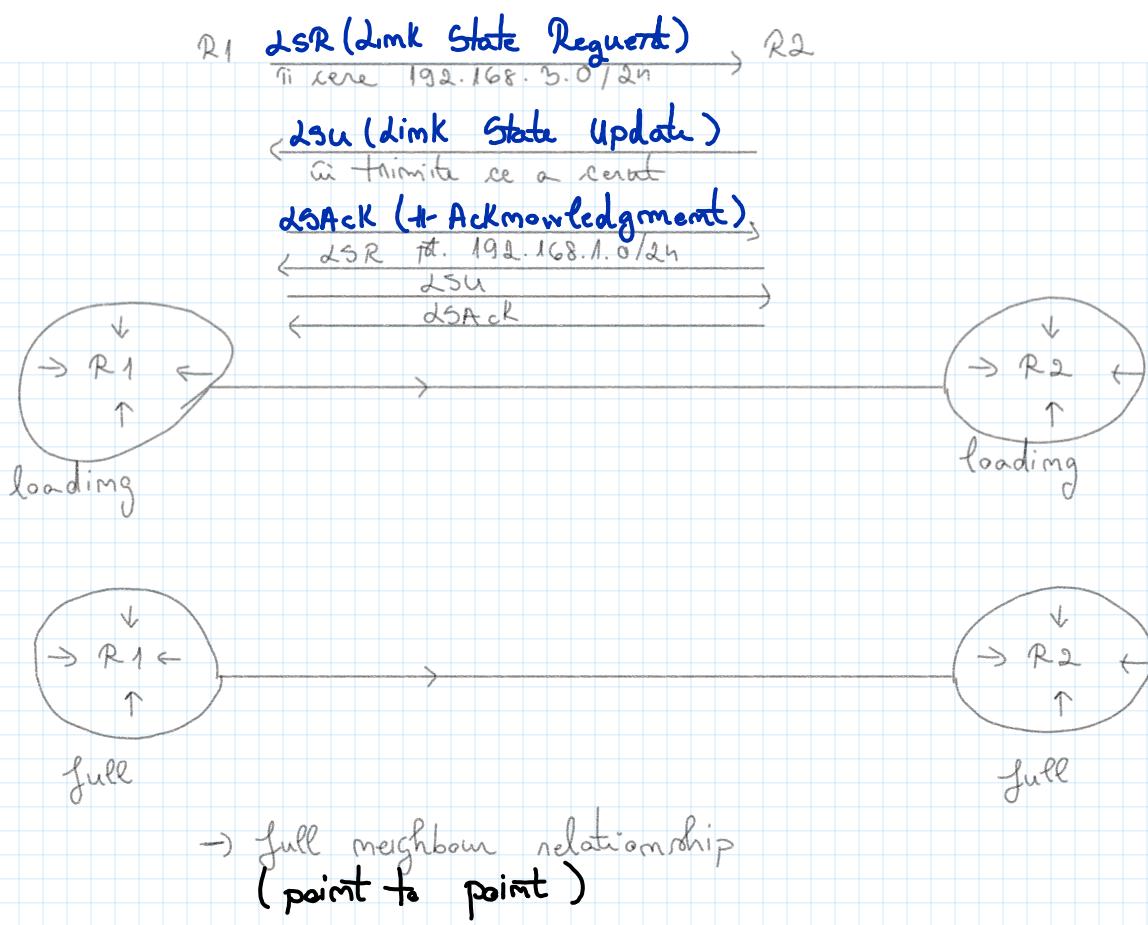
Această, dacă un router a șvadat, el trimită modificare la toti vecinii, dar toate routenele în afara de DR și BDR o ignoră. Astfel, se ocupă DR apoi să modifice routenele (care așteaptă ce să fie DR și BDR)

Cum se aleg DR și BDR?

1. Prioritatea OSPF (cea mai mare (default e 1, dar se poate seta))
2. Router ID-ul cel mai mare

În același segment, routenele devin full neighbours doar cu DR și BDR. Restul vecinilor rămân în 2-way state (bagă în seama doar updateurile venite de la DR/BDR)





Adăugarea celor mai bune nouti la routing table

OSPF Cost = value given to a link
 based on the bandwidth
 of the interface

default:
 100 000 Kbps Reference bandwidth / Interface bandwidth
 Kilobits/second

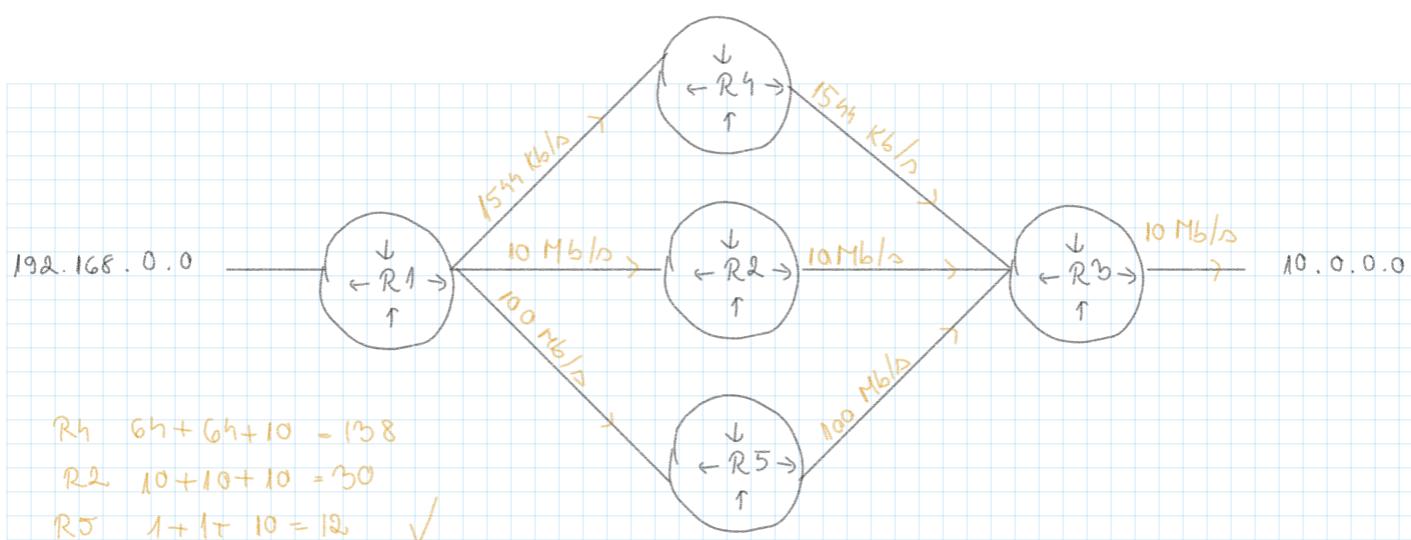
Interface	Default bandwidth	Cost
Serial	1500 Kb/s	64
Ethernet	10 000 Kb/s	10
FastEthernet	100 000 Kb/s	1

Bandwidth

- = capacitatea de transmitere a datelor pînă-n un medium de comunicatie
- se măsoară în bit/s sau multipluri de bit/s / secundă

Link

- = conexiunea fizică / logică ,dintre 2 dispozitive / moduri între-o rețea
- poate fi:
 - cablu fizic
 - conexiune fără fir
 - legătură logică între 2 routere / switch-uri

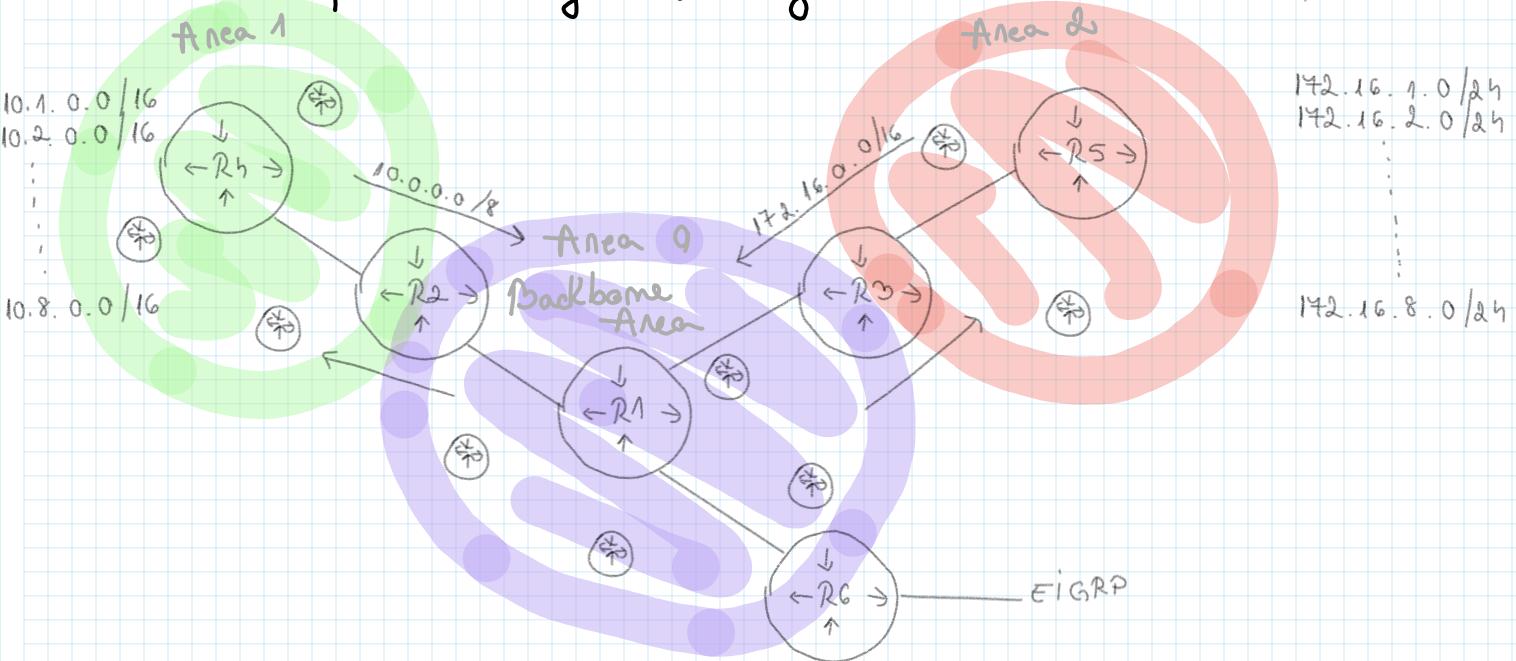


OSPF Multi Areas

- reduce dimensiunea DSDT
- "condensează" în routing tables
- update mesajelor într-o singură zonă

Zonă = grup de roțituri

Recomandat să fie max.
50 de roțituri



1. Area 0 (Backbone Area)

- toate celelalte roțituri să se conecteze la ea

2. Subnetting

- să se grupeze să fie de același tip
- ex: Toate care încep cu 172.16. ceva

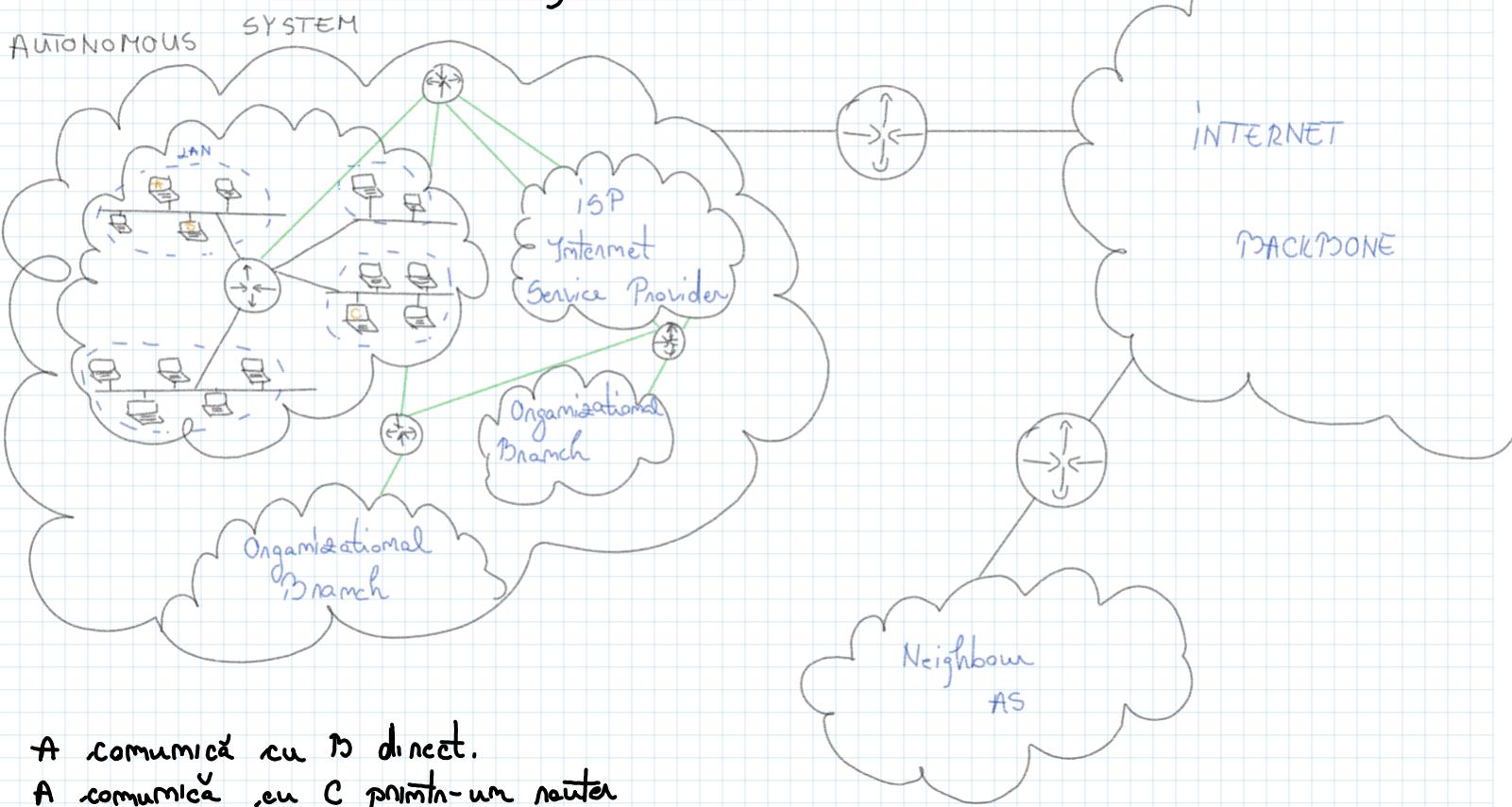
Backbone roțitor R1, R2, R3

Zonă Border Router (ABR) R2, R3 → intermediu

Internal roțitor R4, R5 → nu trebuie să aibă legături cu alte zonă

Autonomous System Boundary Router (ASBR) R6

BGP (Border Gateway Protocol)



A comunică cu B direct.

A comunică cu C printr-un router

↳ folosește diverse protocoale, nu poate include BGP

În același rețea \Rightarrow internal BGP

Router conectat la AS care se conectează la alt router conectat la alt AS \Rightarrow external BGP

sisteme autonome

vecini BGP (AS-URI)

comenzumi TCP între vecini

informații despre rețea

actualizări periodice

Path Vector Protocol \Rightarrow informații se transmit împreună cu ruta în sine \Rightarrow evitarea buclelor în rutare BGP

Autonomous System

- colectie de rute și rețele care au același administrator și au același politici de routare
- se identifică printr-un număr unic

RIP (Routing Information Protocol)

- = protocol de rutare cu vectori de distanță
- utilizarea "hop count" pentru alegerea rutelor

actualizări provocate prin UDP (RIP advertisements / updates)

hop count: hop = treceau printr-un router

ruta mai bună = cele mai puține hopsuri

primesc actualizări prin broadcast și își actualizează tabelele de rutare
valoare maximă: 15 (RIP v1) \Rightarrow ruta imacessibilă \Rightarrow evitarea contorizării la ∞
16 (RIP v2)

split horizon: nu trimit informații despre rutile învăluite prin prima interfață primă
aceeași interfață

hold down timer: perioadă de timp în care routerul
nu primește actualizări pe o rută,
dacă aceasta a devenit imacessibilă

actualizări + alte caracteristici \rightarrow convergență rapidă

- RIP v2 - subnetting
 - autentificare
 - adrese IPv6

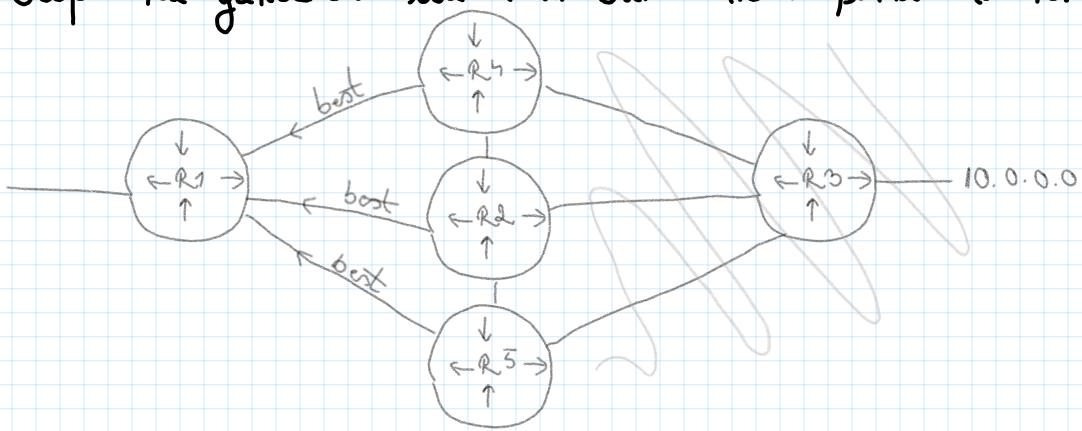
Placa de rețea - device
fiecare prim care se conectează calculatorul cu extenziile. Pe o placă
fisică de rețea se pot pune mai multe adrese IP \rightarrow interfețe

EIGRP

(Enhanced Interior Gateway Routing Protocol)

- released as an informational RFC 7868
- used within a single autonomous system (network independence)
- distance vector and link-state like features

Săptăm să găsească cea mai bună rută pâră la unicore router, din rețea



R1 săptăm să găsească rută pâră la router 10.0.0.0

Pentru el, "există" doar vecinii lui în acel casă, fiecare vecin are o drum acolo. Fiecare vecin încearcă să trimită cea mai bună rută găsită de el pâră la destinație. Apoi, R1 tot amintește vecinii, din stânga în tot ară.

1 Become neighbours

- Hello - 5 sec pt high bandwidth links
- 60 sec pt low bandwidth links
- se trim. non-stop de la unul la altul

Hold timer - 3 x intervalul lui Hello (15 sec)

- urat se antreaptă pâră sănătă că e "mort" celălalt, dacă nu trimită hello înapoi

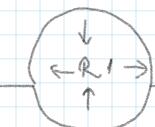
Multicast - 227.0.0.10

Requirements match?

Hello și Hold intervalele nu trebuie să fie!

B2: fie același
Autonomous System → AS number
Number (se aplică la
configurarea EIGRP pe rețea)
Subnet → nu trebuie să fie același rutmet
K-values → nu trebuie să fie același și calculat în
(nu se poate schimba)
Authentication → dacă se folosește, nu se potrivescă

192.168.0.0



10.0.0.0

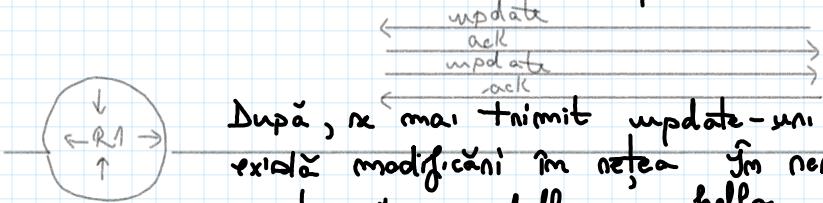
Dacă nu au îndeplinit condițiile ⇒ noutate vecine

2. Exchange routing information

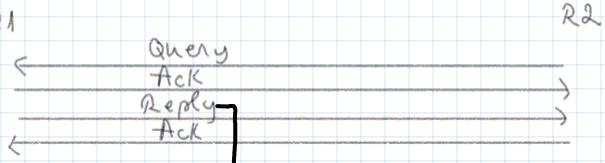
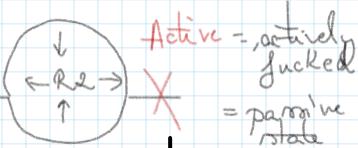
- nu foloseste UDP / TCP

- foloseste RTP (Reliable Transport Protocol)

- foloseste sequence numbers ca să identifice dacă mesajele au fost primite de vecin
- "duel" pt a evita loopurile



După ce mai trimit update-uri doar dacă există modificări în rețea și în mod, nu se continuă cu hello.



dacă generația e ok, dacă nu, trebuie eliminată din routing table

Dacă se întâmplă asta, va petrece un **Route Read Computation**: noul ruteur căută o nodă liberă până la obiectiv. Dacă nu găsește, încearcă pe vecini, dacă nu ei o nodă.

3 Choosing the best routes

• Metric Calculation Formula (Default)

$$((10^7 / \text{dealt Bandwidth}) + \text{Cumulative Delay}) * 256$$

value given to each outgoing link (microsec)

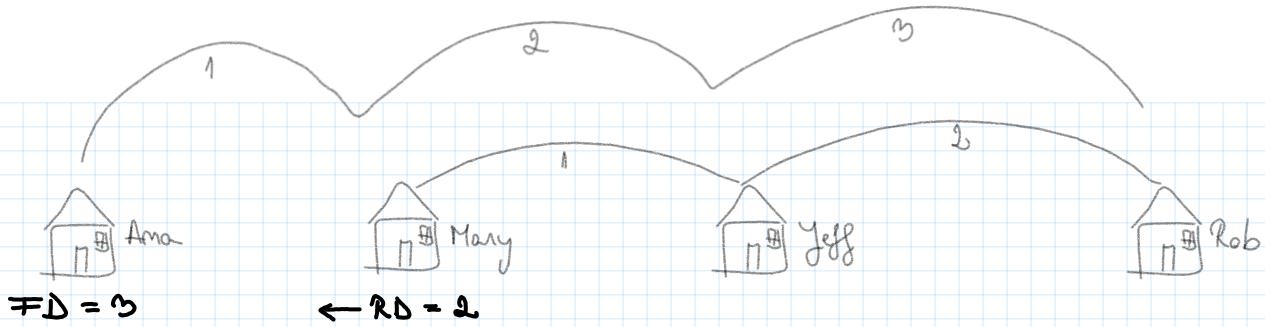


$$((10^7 / 100,000) + 10 + 10 + 10) * 256 = 33,280$$

• Reported Distance (RD) & Feasible Distance (FD)

↓
the metric pt o nodă
distanta a vecinului
(advertised distance)

↓
distanta raportată + distanța de la vecinul care me-a zis de nodă



Ama: Hei Mary, tu numai locuiești Rob?

Mary: Da, la 2 case de mine!

Ama: Super, slăcă ești vecina mea, îmdeamnă să la 3 case de mine!

• Successor vs Feasible Successor

↓
noda cu cel mai
bun metric pâră
la desfășurare
(pot fi mai mulți)

- backup route în caz că este succesor
- trebuie să aibă $RD < \text{Successor FD}$ (pentru evi-
tarea loopurilor) (nu poate folosi și dacă nu se
respectă imegalitatea, dar trebuie verificata loopurile)

• Comenzi

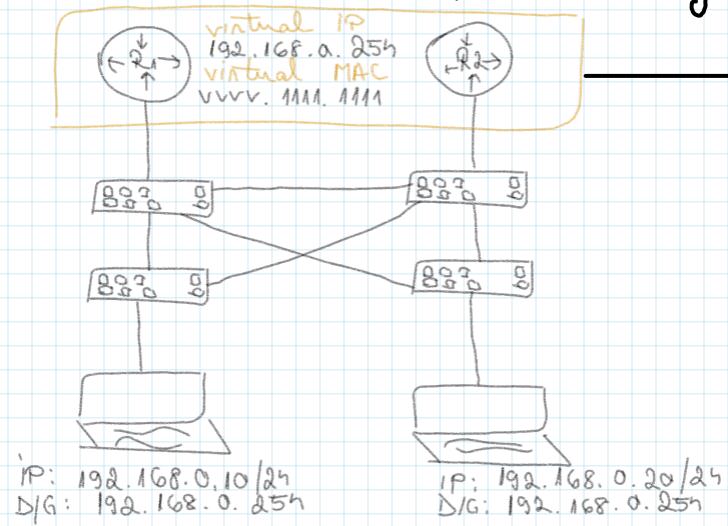
bandwidth x
show ip route eigrp 1 b 10.0.0.0
begin

show ip eigrp topology

→ schimbă valoarea în x

→ schimbă toate informațiile, inclusiv
s în FS (chiar și (FD, RD))

Fifth Hop Redundancy Protocol (FHRP)



se creează un grups pentru a evita erorile care să se petnece dacă un router nu poate să se conecteze la un singur router, care ar putea erau, dacă un dispozitiv ar emis arte, datele unui singur router

→ în cazul în care un router creează și cel rămas nu primește nicio reacție pentru adresa MAC (deci dispozitivele vor să se folosească de tabela de adrese), acesta va trimite un arp, ca dispozitivele să își actualizeze tab

FHRP

HSRP
(Hot Standby Router Protocol)

VRRP

(Virtual Router Redundancy Protocol)

GLBP

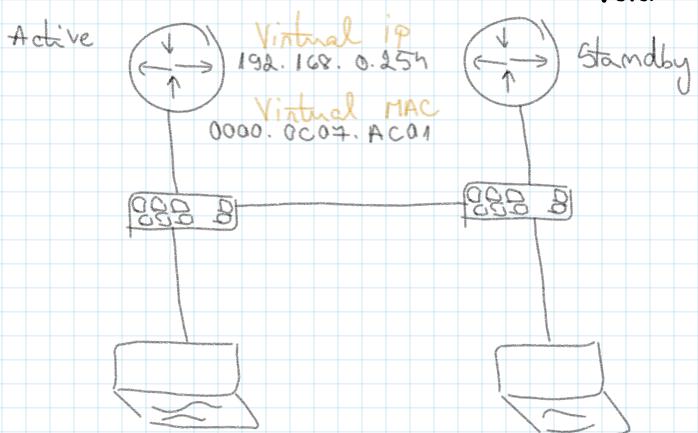
(Gateway Load Balancing Protocol)

HSRP

- dispozitivele trimisă și primesc multicast UDP hello packets la fiecare 3 sec

Venmion 1 22h.0 0 2

Venmion 2 22h.0 0 102



Active Router Election

Highest HSRP Priority
↓

Highest IP Address

Virtual IP - configurat să devină default gateway
 Virtual MAC - se generează automat

Version 1
 0000.0C07 ACXX
 ↓
 Group ID

Version 2
 0000.0C9F TXXX
 ↓
 Group ID

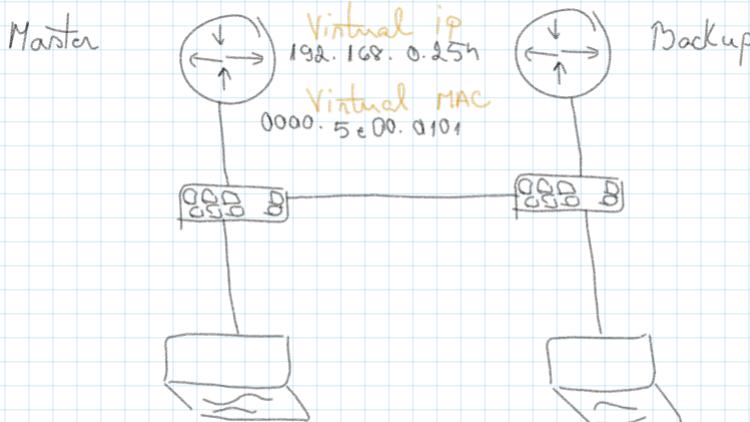
- dacă active router e înecă, standby devine automat active și îi sumează în pe rețea
 - ↳ dacă suntează revine, devine standby (repetă rețea care standby preiau dacă vrei să fie activ)

VRRP

- în loc de active și standby \Rightarrow master și backup
- unul dintre routere primește Virtual IP \Rightarrow IP Address Owner

Master Router Election

IP Address Owner
 ↓
 Highest priority
 ↓
 Highest IP Address



D/G: 192.168.0.254

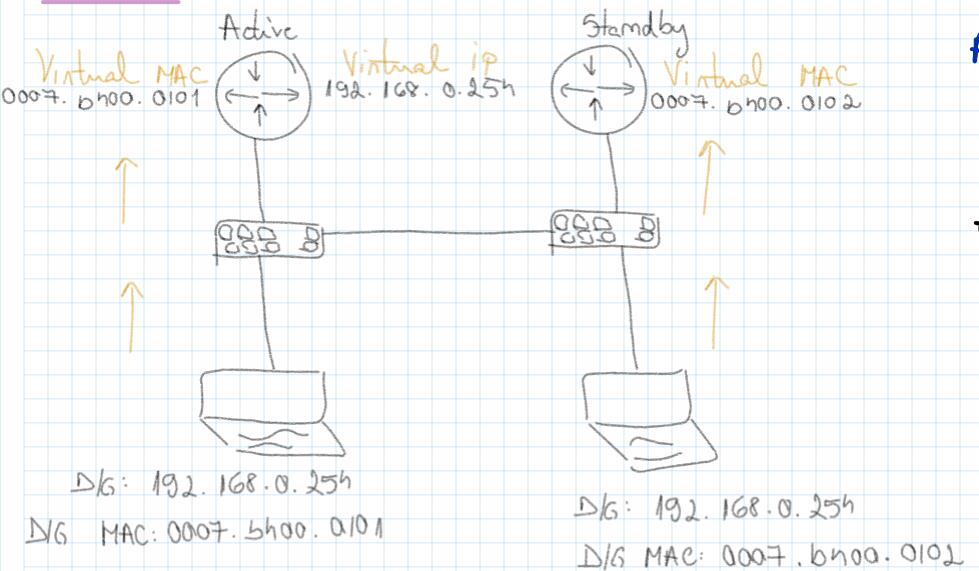
D/G: 192.168.0.254.

Virtual MAC: 0000 5e00. 01XX
 ↓ group id
 VRRP
 Virtual MAC

- doar master trimit mesaje în rețea
- VRRP Master devices trimit advertisement la adresa multicast 224.0.0.18 tot la 1 sec.

- dacă master ernează, se arreagați 3 sec. (3x adv timer) și urmă pac, apoi backup devine master
- ↳ dacă îmi revine, devine iar master automat

GLBP

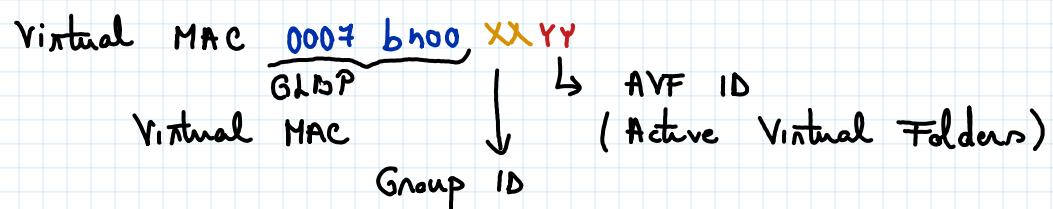


Active Router Election

Highest Priority

Highest IP Address

- fiecare router trimite "hello" rams ca să comunice între ele (multicast UDP)
 - la fiecare 3 sec
 - Multicast: 224.0.0.102
 - UDP Port 3222



- chiar, dacă host-urile au aceeași default gateway IP address, routerele pot să răspundă cu adrese MAC diferite => se pot folosi ambele routere ca să ne flindizeze traficul, înlocuindu-se reciproc în funcție de disponibilitate
- dacă ernează active, se arreagați 10 sec., apoi standby devine active și preia adresele MAC
 - ↳ dacă revine, revine ca standby, dar își ia adresa MAC înapoi

Router Roles Multicast Addrs. MAC addr. Format

HSRP	Cisco Proprietary	Active Standby	v1 224.0.0.2 v2. 224.1.1.102	0000 0C07 ACXX	One	Hello (L) 3 sec. Hold (L) 10 sec.
VRRP	RFC 5498	Master Backup	224.0.0.18	0000 5E00.01XX	One	Advertisement (L) 1 sec. Master Down (L) 3 sec.
GVRP	Cisco Proprietary	Active Standby	224.0.0.102	0007 b400 XXYY	Four Shared	Hello (L) 3 sec. Hold (L) 10 sec.

Network Address Translation (NAT)

The problem

- internetel. a tot, senerant m. n-an terminal ip, h-unile

The solution - Private Addresses

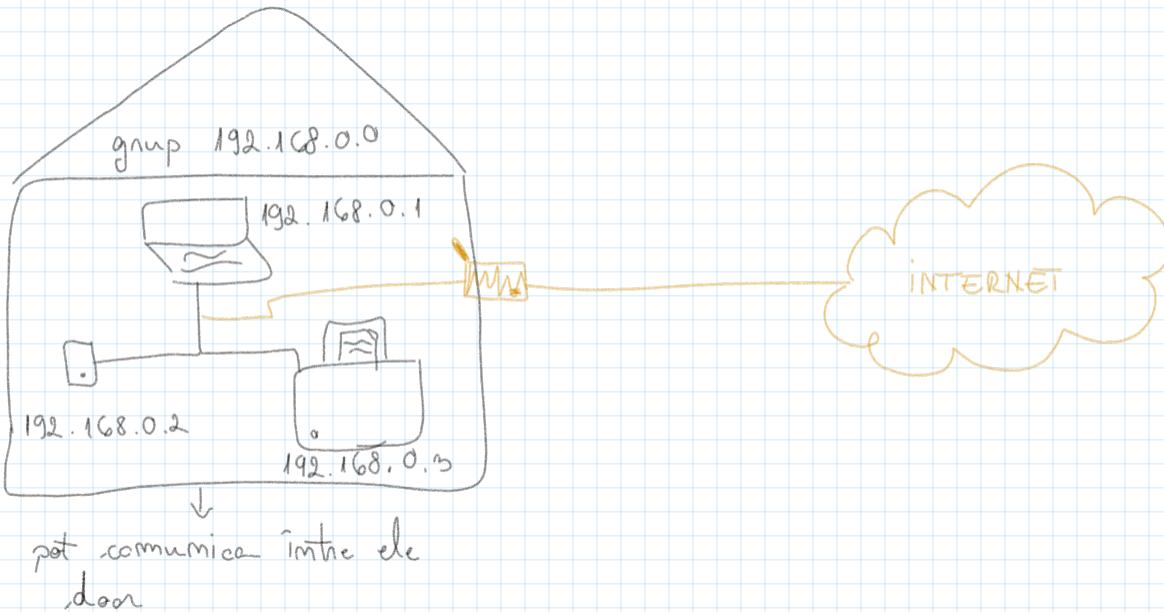
Se pot folosi doar în
rețele interne, nu au valoare
în internetul public

10.0.0.0 - 10.255.255.255.

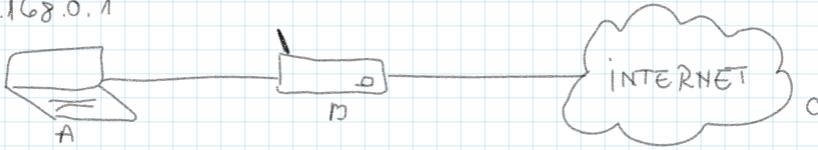
172.16.0.0 - 172.31.255.255

192.168.0.0 - 192.168.255.255

NAT converteste adrese private în adrese publice



192.168.0.1



Tipuri de NAT

1. Overload / PAT (Port Address Translation)

- cel mai popular

 $A \rightarrow B$ 

indica inclusiv aplicatia / tabul de care aparține

Source 192.168.0.1 8897 noutenul leDestinatorm 55.06.47.88.80 schimbă

creaza tabelul, apoi trimite datele

de obicei se păstrează dar
dacă e ocupat, se folosește
următorul liber

11.22.33.44 8899
55.06.47.88.80



Întrare	iese
192.168.0.1:8897	11.22.33.44 8897

 $C \rightarrow B$ 

Source: 55.06.47.88.80

Destinatorm 11.22.33.44 8897

55.06.47.88.80

192.168.0.1 8897



2. Dynamic

- funcționează asimetric, dar

$A \rightarrow B$ - noutenul alege prima adresa liberă găsită în
piscină

11.22.33.44 - 11.22.33.99

- se fac totușu pași de mai sus (cu schimbare, tabel în tot)

$C \rightarrow A$ - se fac totușu pași

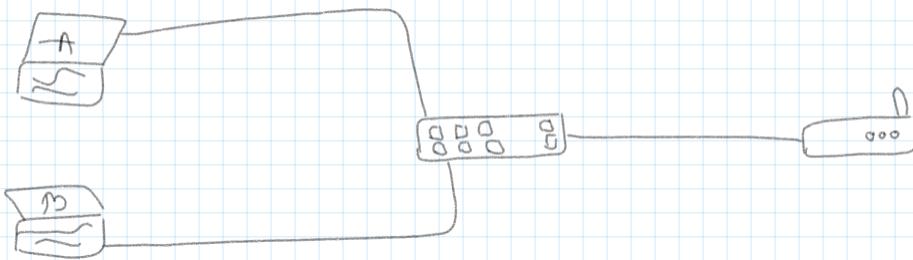
- după ce ajung datele înapoi la device, adrese ip minge înapoi în
piscină și va putea fi folosită iar

3 Static

- adresa privată și cca publică trebuie introduse manual
- în rest, funcționează la fel
- se folosesc mai mult porturi servere web (ex: http , unde portul < 80)

DHCP (Dynamic Host Configuration Protocol)

- assignează adrese IP unice device-urilor
- client / server → UDP Port Client 68
Server 67



Adresele lui A și B trebuie să fie unice, ca să meargă datele unde trebuie

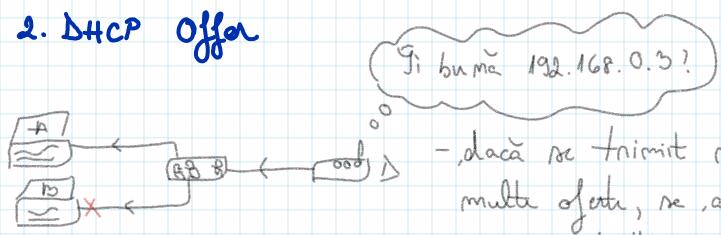
A și B vor fi DHCP clients
DHCP Server va fi un router sau un server

→ la ce urmează, se trimit mesaje broadcast, deci le primește tota lumea și cui nu i se adresează, le ignorează

1. DHCP Discover



2. DHCP Offer



- dacă se trimit mai multe oferte, se alege prima primată

3. DHCP Request

A găsește că o are

4. DHCP ACK

D: OK, fișă, săn, săn și vreau înapoi

D → A: adresa IP, subnet mask, default gateway și serverul DNS

Serviciul DHCP time apoi evidență

Dispozitivul trebuie să își recunoască adresa, astfel exprimă în mărgele înapoi în prima cu adresa.



⇒ evitarea nimănui adreșelor ip (dacă suntem / deconectăm un dispozitiv)

Syslog

Syslog Server

- toate dispozitivele din rețea îi trimit log information
UDP Port 514

- Beneficii
- verifică toate informațiile mai ușor, dintr-un singur loc
 - date retention (când se rezarcă un device, logurile se resetă)
 - se arhivează mai ușor

Cinco device le rețin în RAM



LOG information

= înregistrările / jurnalul
care capturează evenimentele activității sau mesajelor într-un sistem / aplicație

Jurnal de rețea

- informații precum - comenzi, decodări, trafic de date, erori de comunicare etc

Data retention

- politica și practicile în ceea ce privește păstrarea și stocarea datelor pentru o anumită perioadă de timp

Severity

= cat de urgent e log RMS

Timestamp / Sequence number

Aug 26 18:04:43.647 : %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up

număr
mesajului

Facility

Mnemonic

- cod pentru identificarea RMS-ului

Description
= mesajul log

Facility

0	Kern	Kernel logs
1	user	user-level logs
2	mail	mail system
3	daemon	system daemons
4	auth	security / authentication logs
5	syslog	logs generated internally by syslogd
6	lpr	line printer subsystem
7	news	network news subsystem
8	uucp	UUCP subsystem
9	cron	clock daemon
10	authpriv	security / authentication logs
11	ftp	FTP daemon
12	ntp	NTP subsystem
13	security	log audit
14	console	log alert
15	solaris-cron	clock daemon
16-23	local	local user

Severity

Fiecare grav 0 → 7 Nu este grav

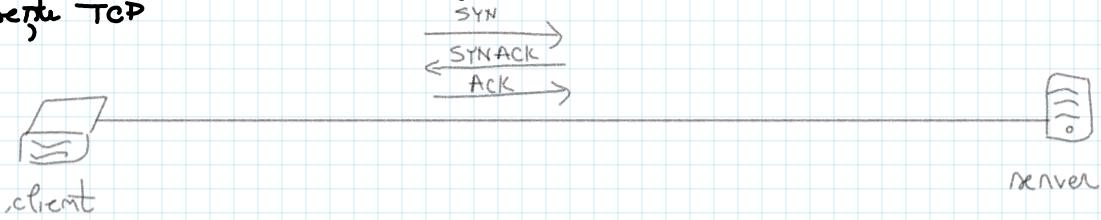
Code	Severity	Description
0	Emergency	System is unusable
1	Alert	Action must be taken immediately
2	Critical	Critical conditions
3	Error	Error conditions
4	Warning	Warning conditions
5	Notice	Normal but significant conditions
6	Informational	Informational messages
7	Debug	Debug-level messages

Când scriem log cu o anumită severitate, îți le dă pe toate de la severitatea saia în sus. dacă scri "Informational", îți dă tot ($6 \rightarrow 0$) pînă la Emergency

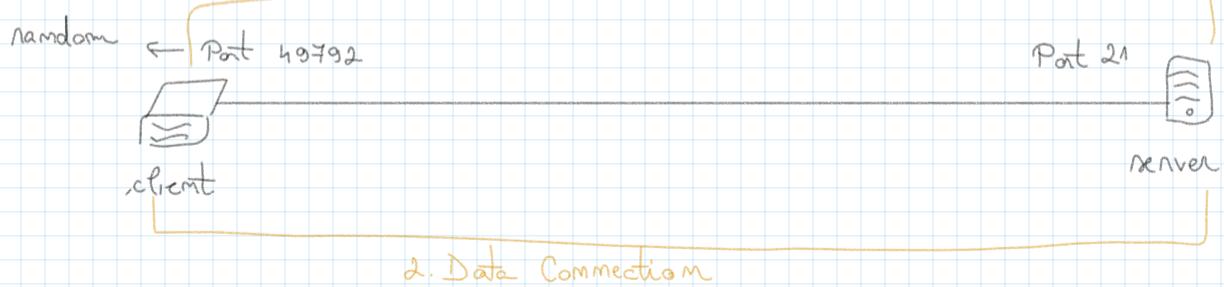
FTP

(File Transfer Protocol)

- = protocol folosit pentru transferul de fișiere între o rețea
- folosește TCP



1. Control Connection



Data Connection

1. Active: serverul face primul pas, având ca port număr 20, către un port generat dinamic random.
 - dacă există un firewall între client și server, cel mai probabil conexiunea mesajată către portul serverului nu va fi blocată
2. Pasiv: clientul face primul pas de pe portul număr generat random, către portul destinație 21
 - dacă există firewall, acesta nu blochează traficul, pentru că cel care a inițiat conexiunea a fost clientul
 - nu este necesar, deocamdată, să toate datele sunt transmise clar

FTPS (FTP Secure / FTP SSL)

- = extensie a FTP care supune utilizarea a TSL și SSL encryption
- ↓
protocol de criptare permitând menținerea datelor în siguranță și departe de hackeri
(îl privim ca și pe un tunel, care nu lasă datele să se vadă)
- nu trebuie confundat cu SFTP (SSH File Transfer Protocol)
↳ extensie a protocolului SSH

TFTP (Trivial File Transfer Protocol)

- = varianta mai "nedură" a FTP
- metodă simplă pentru un transfer de fișiere rapid și eficient
- folosește port UDP 69
- nu există autentificări, criptări, etc

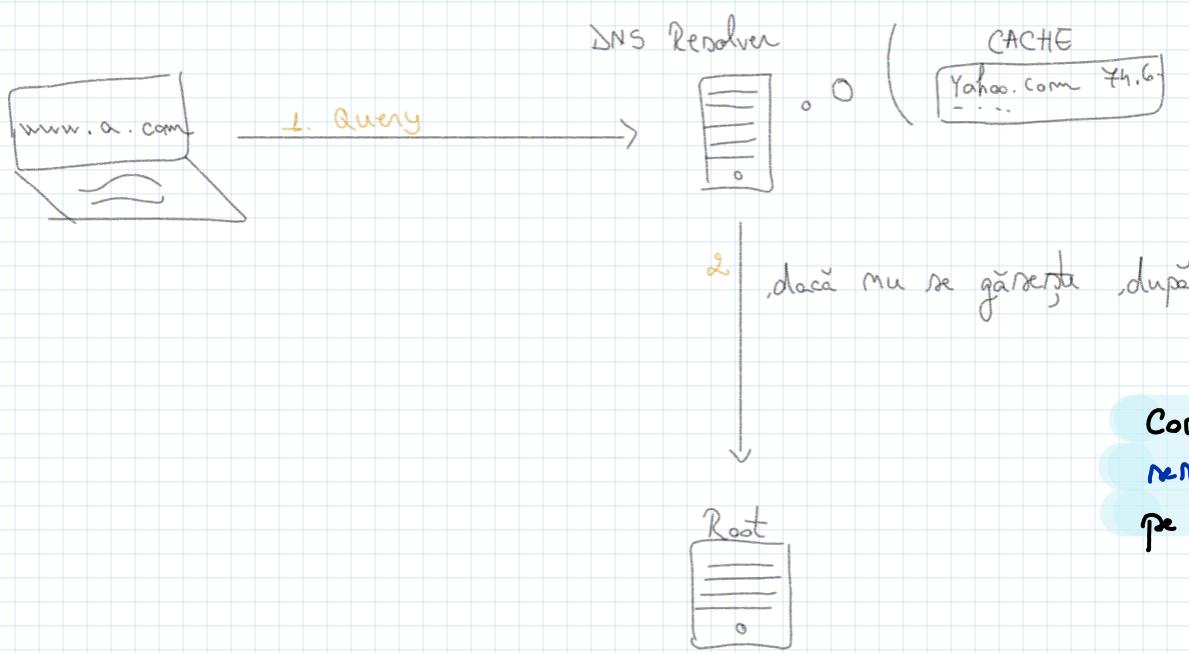
DNS Domain Name System

- preia un link și îl transformă în adresă IP (serverele web funcționează cu adrese IP)



- se verifică local cache pe computer și browser
- se verifică o local configuration file

,dacă nu există date, se trimită un query care cere o adresă IP pentru www.a.com



,dacă nu se găsește după 1

Comunicația între
serverele DNS se face
pe portul 53 prim **UDP**

Root Name Server

- primul din ierarhia DNS
 - primul pas pt. transformarea linkului în adresă IP
- ↳ există foarte multe, dar fiecare folosește 1 din 13 adrese IP

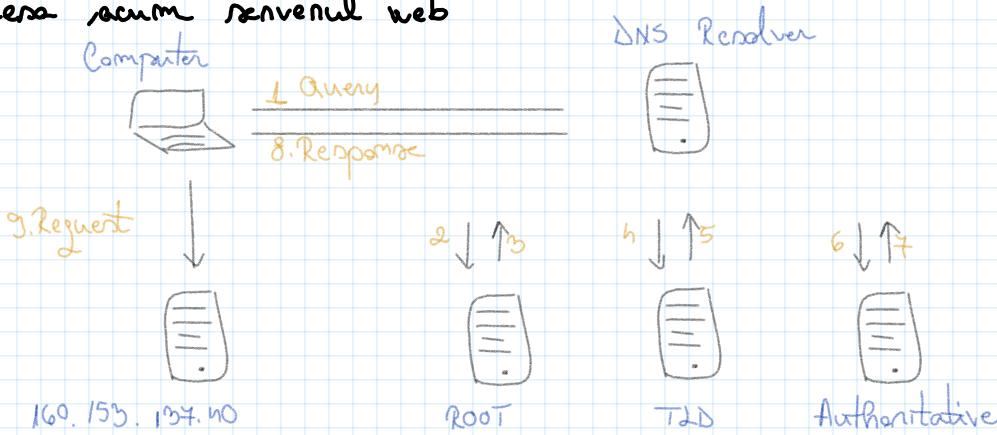
Rol: să găsească de la top level domeniul server (com, org etc)

TLD Name Server

- conține informații pentru domenii with a specific extension (com, net, org etc)
- tot nu știe adresa IP de care avem nevoie
- știe locația lui authoritative marne server

Authoritative Name Server

- ultimul pas în obținerea răspunsului cerut
- conține informații DNS pentru domeniile de care se ocupă
- trimite adresa IP lui DNS Resolver, care îl trimite computerului, care poate accesa acum serverul web



Lista de adrese IP pentru ROOT

198.41.0.4
 199.9.14.201
 192.53.4.12
 199.4.91.13
 192.203.230.10
 192.55.241
 192.112.36.4
 198.94.190.53
 192.36.148.14
 192.58.128.30
 193.0.14.129
 199.4.83.42
 202.12.24.33

Type A

- = înregistrare a unei IPuri (pentru un domeniu)
- pt IPv4, AAAA

Proxy Server

- server intermediar între un client și alte servere web
- performanță, securitate, confidențialitate (poate bloca accesul la diferite site-uri web, poate limita accesul la diferențe surse de date)
- poate ascunde adresa IP a unui client \Rightarrow navigare anonimă pe internet

Tipuri

Forward Proxy - client - servere web

- performanță și securitate conexiunii la internet

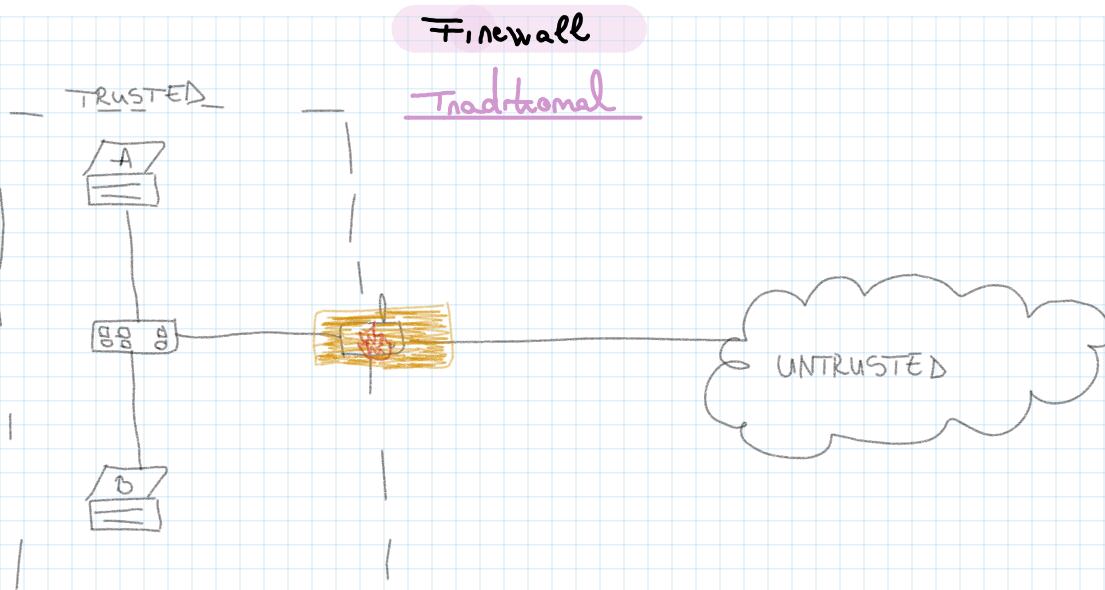
Reverse Proxy - client - unul / mai multe servere web

- performanță, securitate și confidențialitatea conexiunii la internet

Open Proxy - server intermediar utilizat de oricine pt a accesa internetul

\downarrow

în general pt navigare anonimă, ascunderea adresei IP



- cu scopul de a proteja rețelele trusted de cele untrusted
- by default, ele blochează tot traficul, dar vom să blocăm doar ce nu e bun

Firewall rules

SOURCE	DESTINATION	PORT	ACTION
Host A	any	HTTP	allow

Acum, A va putea să trimită mesaje cui vrea. B, de exemplu, va fi în continuare blocat de firewall.

Stateful firewalls - monitorizează conexiunile active
 ⇒ dacă lui A i-a fost permis să trimită date, e acceptat și traficul invers

NGFW's

(Next-Generation Firewalls)

Application level inspection (identifică și blochează)

IPS (Intrusion Prevention System) - patterns / signatures
 - anomalies

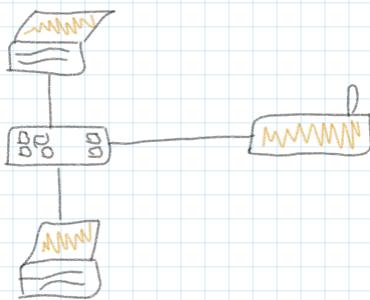
Threat Intelligence (updaters)

Features

URL filtering
 email scanning
 DLP (Data Loss Prevention)
 etc.

} UTM (Unified Threat Management)

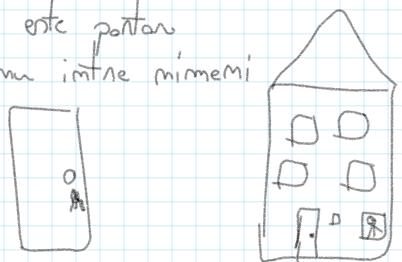
Software based firewalls



- dubă protecție, dacă am emișia vîme, dim ext
 - protecție, dacă vîme dim interior
- ex Windows Firewall

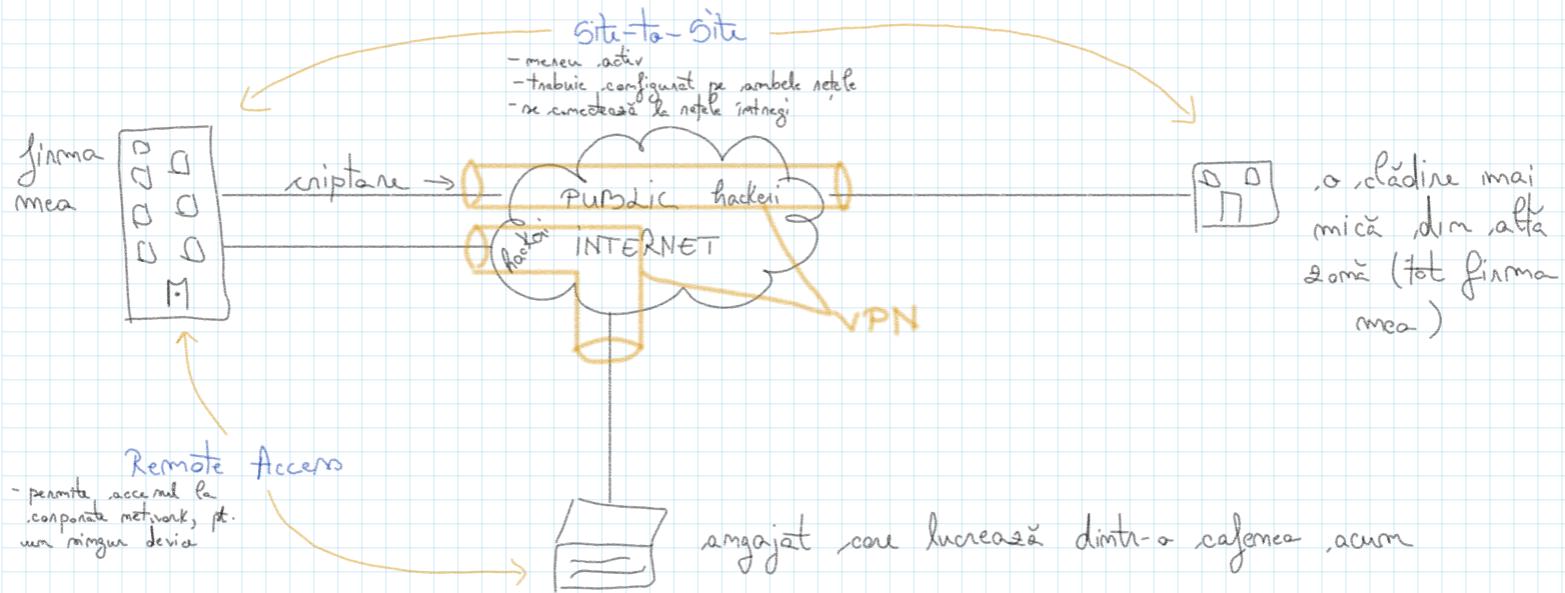
Că rai la cămin:

Teoretic nu pot intra persoanele năzădate, pă. că și în trebuie, caseta și este poartă
Dar mai tot ne închidem cu cheia usă de la cameră să nu nu intră nimic
care e în cămin deja (sau pă. mai doar me poartă)

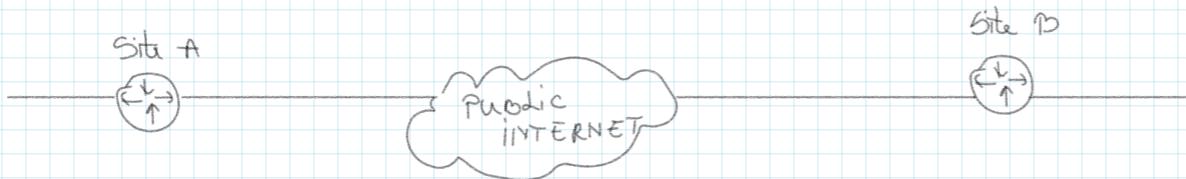


VPN (Virtual Private Network)

- se ocupă de livrarea în siguranță a datelor în rețea publică



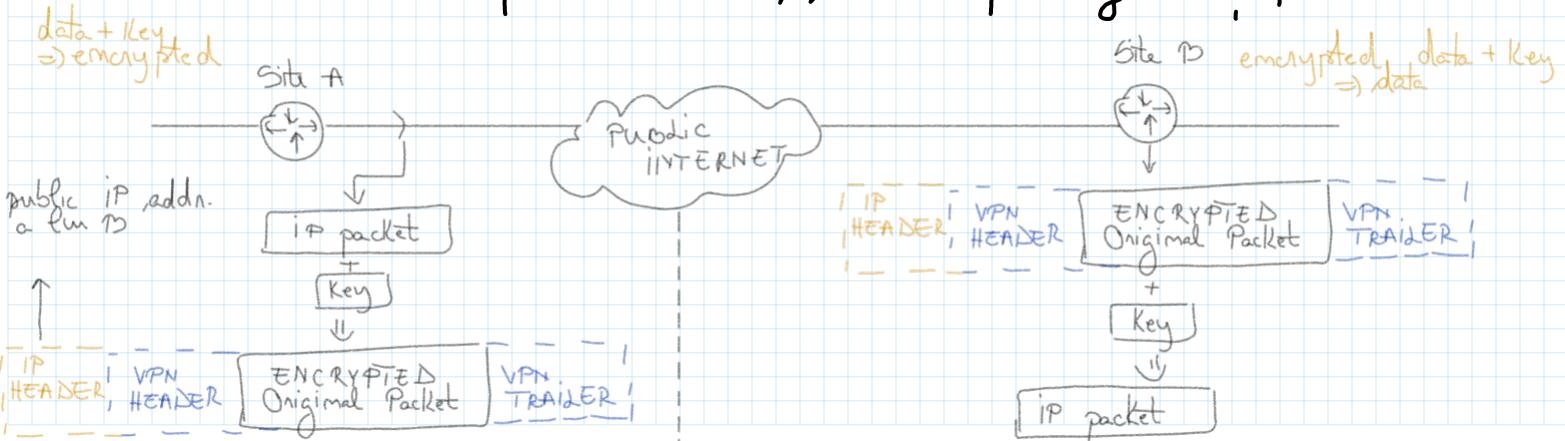
Site-to-Site VPN



- se configuriște deobicei pe un router / firewall pe ambele rate-uri

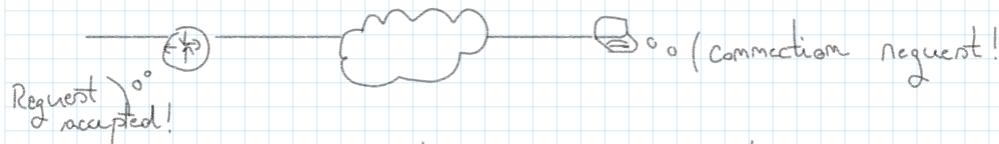
IPSEC VPN = framework / set of rules pt. creaarea VPN-ului în rețea (= pt. securizarea comunicării între o ip network)

- permite folosirea mai multor protocoale pt. fiecare VPN feature
- deobicei pt. rate-to-rate, dar se poate folosi și pt. remote access



Remote Access VPN

- permite conectarea unui singur device la o corporate network
- trebuie să secrete către host ca să te conectezi la rețea



Ex. de VPN Client Application: Cisco Anyconnect, OpenVPN

- se folosește în general **TLS (Transport Layer Security)**

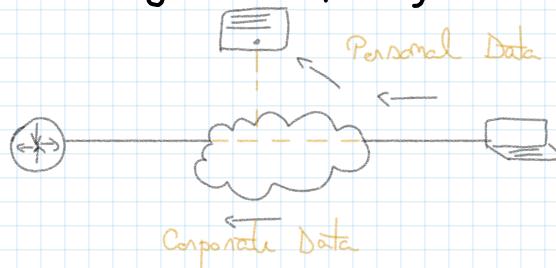
↳ succesor a lui SSL (Secure Sockets Layer)

- se folosește și pt traficul web, la conectarea la rețeaua http
- folosește în general portul 443, care e permis în general (e bine, pt. ca unele wifi-uri publice blochează porturile IPSEC)

Full tunnel - dacă te-ai conectat la VPN, tot traficul te va transmite la corporate network (în dacă răi pe fb de ex)

Split tunnel - doar traficul destinat ei va fi transmis către corporate network

- bandwidth saving + user privacy



ACL (Access Control List)

= rule-based lista filtrelor de routare în switch-uri pt identificarea trafiului

în funcție de ip addrs.
(source addrs, destination addrs)
și port numbers

- în general filtre pentru deny / allow traffic
- se mai folosesc pentru NAT și quality of service

10	deny	tcp	192.168.10.0	0.0.0.255	host	192.168.20.50	eq	ftp
20	deny	tcp	192.168.10.0	0.0.0.255	host	192.168.20.50	eq	telnet
30	permit	ip	host	192.168.10.0	host	192.168.20.50	eq	

Ondimea regulilor

- menge dim 10 îm 10 ca să poată revă înăuntru mai adângi între
- ondimea e foarte importantă, pentru că menge de reguli împreună nu încă un match pentru regulă. dacă găsește, aplică regulă și se oprește din căutare
- dacă nu se face niciun match \Rightarrow denied (ultima regulă e Implicit Deny)

Tipuri

Standard ACL

- numbers: 1-99
- expanded mrs 1300-1999 (\Rightarrow mai multe ACL / device)
- folosește doar source addrs. să identifice traficul

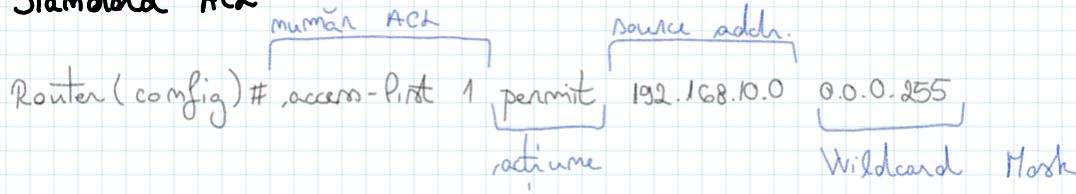
Extended ACL

- numbers 100-199
- expanded mrs. 2000 - 2699
- identifică traficul prin source addrs, destination addrs, protocol și port number

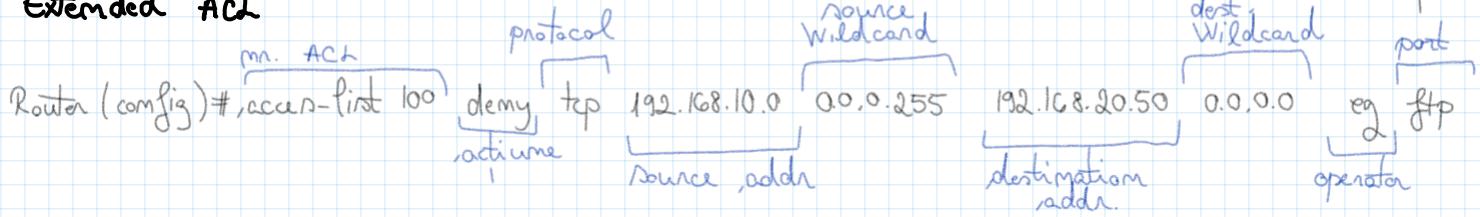
Named ACL

- permite ca standard și extended ACLs să primească nume ca să fie mai ușor de gestionat (ex: când sunt mai multe ACLs pe un device, să stiu care fiecare de ce se ocupă)

Standard ACL



Extended ACL



Named ACL

Tipuri: standard / extended

Router (config) # ip access-list extended drama

Router (config-ext-macl) # deny tcp 192.168.10.0 0.0.0.255 192.168.20.50 0.0.0.0

Router (config-ext-macl) # permit ip 192.168.10.0 0.0.0.255 192.168.20.50 0.0.0.0

!!! ft importantă, analiză

Wildcard Mask

- ca și o subnet mask inversată

0 = biti must match

1 = biti do not matter

Adresa IP

192 168 10 0 11000000 . 10101000 00001010 00000000

Wildcard Mask

0 0 0 255 00000000 . 00000000 00000000 11111111

→ match - urătu toate adresele ip între 192.168.10.0 și 192.168.10.255

Port Operator

gt = Greater Than

lt = less Than

neq = Not Equal

eq = Equal

range = Range Specified

Extended IP access list 101

```
10 deny tcp 192.168.10.0 0.0.0.255 host 192.168.20.50 eg ftp
20 deny tcp 192.168.10.0 0.0.0.255 host 192.168.20.50 eg telnet
30 permit ip host 192.168.10.0 host 192.168.20.50
```

Dacă Wildcard Mask are 32 de bîte / e 0.0.0.0, putem său Keyword-ul "host"

Standard IP access list 10

```
10 permit host 192.168.10.10
20 permit host 192.168.10.15
30 permit host 192.168.10.20
```

Extended IP access list 102

```
10 permit tcp any host 192.168.20.50 eg www
20 permit tcp any host 192.168.20.50 eg ftp
```

Potem folosi Keyword-ul "any", pentru a fi acceptata orice adresa IP

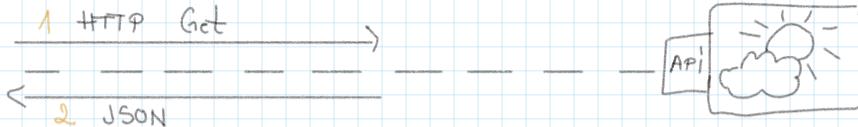
API

(Application Programming Interface)

= "interfață" către o aplicație

HTTP Methods**POST****GET****PUT****PATCH****DELETE****CRUD****CREATE****READ****UPDATE (replace)****UPDATE (modify)****DELETE**

exemplu:

aplicație mobilă
care vrea să vădezeaplicație pt. weare, cu fl. multe
pt. buton rezurse1: `https://api.openweathermap.org/data/2.5/weather?q={city name}&appid={API key}`

unic, al tău, ca aplicația nă ţine evidența, ceea ce înseamnă că nu îți spălăzi

2: "main":
 "temp": 9.53,
 "feels-like": 7.62,
 "temp_min": 4.78,
 "temp_max": 10.56,
 "pressure": 1016,
 "humidity": 61

y

De încrezut developers.google.com/youtube/v3

↳ e interesant, poti de ex să vezi căte like-uri, videodisponibile, subscrise etc. sau cum, cum

IP Addresses și NetMask

IP Address (IP, h)

- 32 de bîta \Rightarrow 4 octeta

00000001.00000010.00000011.00000100 $\xrightarrow{\text{scriem mai ușor}}$ 1.2.3.4
 1 2 3 4

Network Mask (NetMask, Mask)

- 32 de bîta \Rightarrow 4 octeta

11111111.11111111.11111111.00000000 \rightarrow 255.255.255.0
 24 de 1 8 de 0
 ↓ ↓
 124 2⁸ adrese IP în rețea

11111111.11111111.11111111.11000000 \rightarrow 255.255.255.192

26 de 1 \Rightarrow /26

6 de 0 \Rightarrow 2⁶ de adrese IP în rețea

11111111.11111111.11110000.00000000 \rightarrow 255.255.240.0

20 de 1 \Rightarrow /20

12 de 0 \Rightarrow 2¹², adrese IP în rețea

- amelioră diferențele dintre adresele IP ale device-urilor din aceeași rețea

Adresa de rețea

= NM AND ADRESĂ IP

Adresa de rețea

- adresa IP care identifică în mod unic o rețea
- permite adresa IP 192.168.1.10 cu masca 255.255.255.0, adresa de rețea e 192.168.1.0
- întotdeauna pară (broadcast = impară)
- nu se poate folosi

$$\begin{array}{r} 11111111.11111111.11110000.00000000 \\ \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\ 1 \quad 0 \quad 0 \quad 0 \end{array} \Rightarrow 2^7 + 2^6 + 2^5 + 2^4 = 128 + 64 + 32 + 16 = 240$$

$$2^7 + 2^6 + 2^5 + 2^4 =$$

$$128 + 64 + 32 + 16 =$$

$$192 + 48 = 240$$

Tot internetul: [0 0 0 0 \rightarrow 255 255 255 255] \rightarrow 2³² adrese IP în Internet
 NM = 0 0 0 0

NetMask-urile împart internetul în subrețele (intervale)

NM = 11 10 000 \rightarrow 2^x IP

[Start IP . . . End IP], , size = 2^x
 ↓ ↓
 Network Address (NA) Broadcast Address (BA)

Nu putem spune că o adresă IP e adresa de rețea fără să stăm netmask-ul!

Network Address (NA)

$$NA = IP \text{ AND } NM$$

Broadcast Address (BA)

$$BA = IP \text{ OR } (\text{NOT } NM)$$

Ex 1:

$$IP = 10.11.12.13$$

$$NM = 255.255.255.0 \quad /24$$

$$32-24=8 \Rightarrow 2^8=256 \text{ de IP-uri}$$

$$[NA \dots BA] \quad \text{mărime} = 256$$

NA

$$\begin{array}{r} 10.11.12.13 \text{ AND} \\ 255.255.255.0 \\ \hline 10.11.12.0 \end{array}$$

BA

$$\begin{array}{r} \text{NOT } NM = 0.0.0.255 \\ 10.11.12.13 \text{ OR} \\ 0.0.0.255 \\ \hline 10.11.12.255 \end{array}$$

$$\Rightarrow IP = 10.11.12.13$$

$$NM = 255.255.255.0$$

$$[10.11.12.0 \rightarrow 10.11.12.255], \text{ mărime} = 256$$

Ex 2:

$$IP = 10.11.12.13$$

$$NM = 255.255.255.248 \quad /29 \Rightarrow 2^5 \text{ IP-uri}$$

$$NM = 255.255.255.1111000$$

$$\begin{array}{r} NA = 10.11.12.00001101 \text{ AND} \\ 255.255.255.1111000 \\ \hline 10.11.12.00001000 \end{array}$$

$$NA = 10.11.12.8$$

$$\text{NOT } NM = 0.0.0.0.0000111$$

$$BA = 10.11.12.00001101 \text{ OR}$$

$$0.0.0.0.00000111$$

$$\hline 10.11.12.00001111$$

$$BA = 10.11.12.15$$

$$\Rightarrow IP = 10.11.12.13 \quad \text{și} \quad NM = 255.255.255.248 \quad /29$$

$$[10.11.12.8 \rightarrow 10.11.12.15], \text{ mărime} = 8$$

$$\begin{array}{r} \text{AND cu 1 în 0} \\ abcdefgh \text{ AND} \\ 11111111 \\ \hline abcde fgh \end{array}$$

$$\begin{array}{r} abcdefgh \text{ AND} \\ 00000000 \\ \hline 00000000 \end{array}$$

$$/32 \quad 255.255.255.255$$

- marcă de netea care demonstrează că nu există la un anumit IP, nu poate fi adăugat sau scăzut

0
1
0
1
0
0
1
0
0
1
0
1
1
0
0
0
1
0
1
0
1
0
0
1
1
0
1
1
1

Network Splitting

1.0.0.0 /24
 ↓
 network address

masca are 24 de 1 \Rightarrow 8 de 0 \Rightarrow NM = 255 255 255.0
 măre = 256

[1.0.0.0, , 1.0.0.255]

 256 valori

Impărțirea $[NA \dots BA] \Rightarrow [NA_1 \beta A_1][NA_2 \dots \beta A_2]$
 NM + 1

Exemplu:

[1.0.0.0 ... 1.0.0.255] /24
 $/24 = 11111111.11111111.11111111.00000000 = 225.225.225.0$
 $24 + 1 = 25$
 $/25 = 11111111.11111111.11111111.10000000 = 225.225.225.128$
 $\Rightarrow [1.0.0.0 \dots 1.0.0.127] [1.0.0.128 \dots 1.0.0.225] \quad măre = /25 = 2^7 = 128$

 128 128

 256

Cost general NA și BA:

m devices \rightarrow m IP addrs.

NA și BA nu se pot folosi pe dispozitive \Rightarrow m+2

Mac - $2^x \Rightarrow m+2 \leq 2^x$

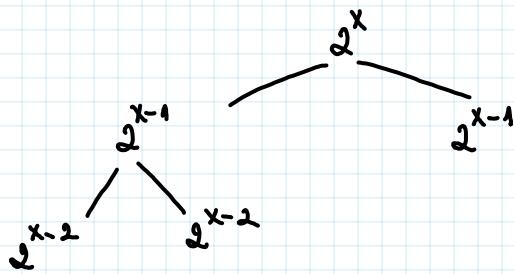
NM = $/(32-x)$, unde 11.10 00 avem x zero-uri în 32-x de 1

Cost general splitting:

1 10 0 x de 0

1 110 . 0 x-1 de 0

1 110 0 x-2 de 0



[NA	.	.	βA]	NM
[NA1	.	βA1]	[NA2	βA2]
[NA1	.	βA1]	[NA3 βA3]	[NA4.. βA5]
				NM + 2

Exemplu:

NA (Network IP Address) = 82.228.39.0

NM (Mask) = 225.225.225.0 (/24)

Subnetworks: N1: 40 IPs

N2: 40 IPs

N3: 16 IPs

N4: 20 IPs

N5: 4 IPs

N6: 3 IPs (între 3 routere)

N7: 2 IPs (-II-2)

N8: 2 IPs (-II-)

N9: 2 IPs (-II-)

N10: 2 IPs (între un router și un wireless router)

$$h_0 + 3 = h_3 < \textcircled{6h} = 2^6 \rightarrow 6 \text{ de } 0 \rightarrow 32 - 6 = /26$$

$$h_1 + 3 = h_4 < \textcircled{6h} = 2^6 \rightarrow /26$$

$$1C + 3 = 19 < \textcircled{32} \Rightarrow 2^5 \rightarrow /27$$

$$20 + 3 = 23 < \textcircled{32} \rightarrow /27$$

$$h + 3 = 7 < \textcircled{8} \rightarrow /29$$

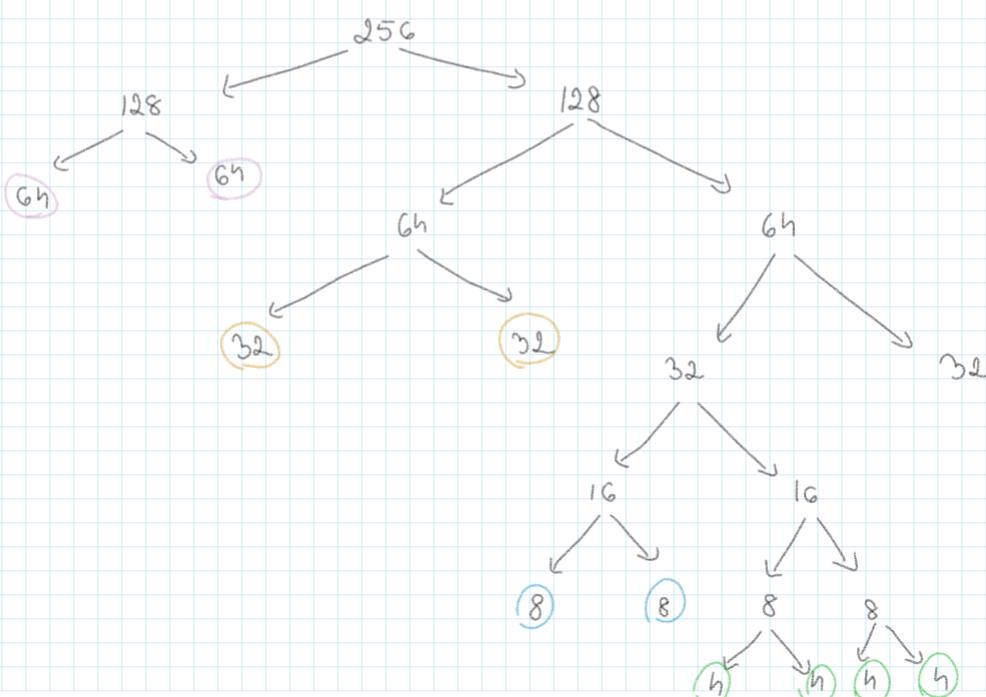
$$3 + 2 = 5 < \textcircled{8} \rightarrow /29$$

$$2 + 2 = h \leq \textcircled{h} \Rightarrow /30$$

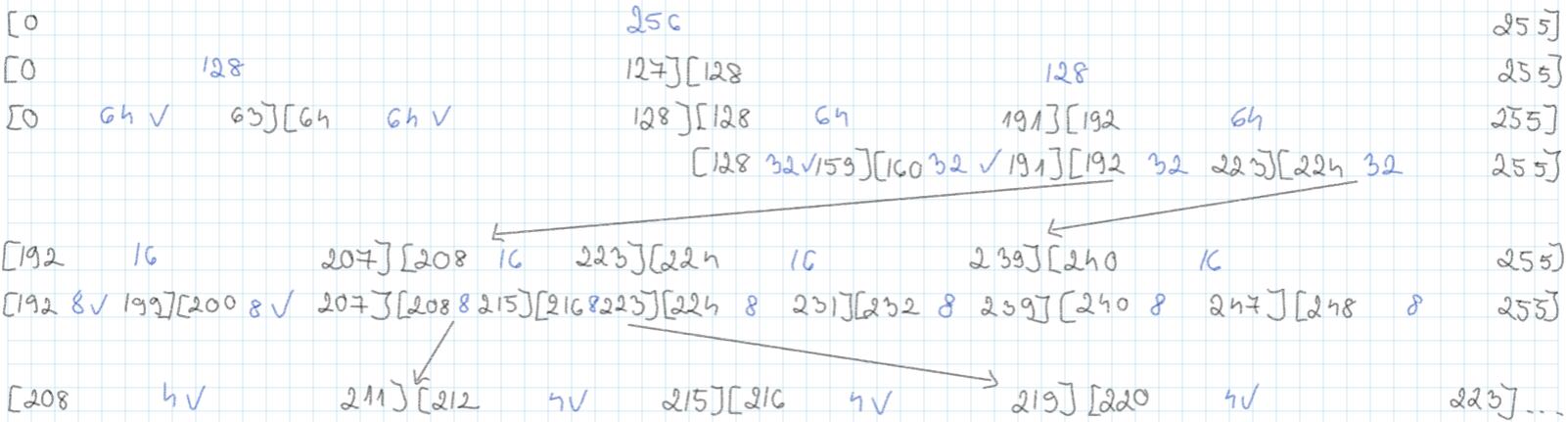
$$2 + 2 = h \leq \textcircled{h} \Rightarrow /30$$

$$2 + 2 = h \leq \textcircled{h} \Rightarrow /30$$

Verificare: $2 \times 6h + 2 \times 32 + 2 \times 8 + h \times h = 22h < 256 \quad \checkmark$



82. 228. 39. 0 /24



N1: 82. 228. 39. 0/24

NM: 255. 255. 255. 192

N2: 82. 228. 39. 64/24

NM: 255. 255. 255. 192

N3: 82. 228. 39. 128/24

NM: 255. 255. 255. 224

N4: 82. 228. 39. 160/24

NM: 255. 255. 255. 224

N5: 82. 228. 39. 192/24

NM: 255. 255. 255. 248

N6: 82. 228. 39. 200/24

NM: 255. 255. 255. 248

N7: 82. 228. 39. 208/24

NM: 255. 255. 255. 252

N8: 82. 228. 39. 212/24

NM: 255. 255. 255. 252

N9: 82. 228. 39. 216/24

NM: 255. 255. 255. 252

N10: 82. 228. 39. 220/24

NM: 255. 255. 255. 252

available: 82.228.39.224/24

NM: 255. 255. 255. 248

Dispozitivele vor primii IP-uri
începând cu prima valoare disponibilă (necesar, că N1 și N2 sunt
pe patru octeti)

ex: numărul dim N1 82.228.39.1
num. disp. dim N1: 82.228.39.2
etc
numărul dim N7. 82.228.39.161
etc.

Router

- dispozitiv care conectă 2 rețele diferite
- redirecționează pachete de date folosind adresele IP atribuite fiecărui dispozitiv într-un LAN
- preia pachete de date de la device-urile conectate la LAN și le redirecționează către și de pe internet către segmentele LAN, prim NAT

Switch

- dispozitiv care poate conecta mai multe dispozitive într-un LAN
- mai inteligente decât hub-urile (decât noilele mi)
 - ↳ nu ne deosebesc de ele, nu pot limita traficul de date către și de la fiecare port (\Rightarrow fiecare dispozitiv are o capacitate suficientă de bandwidth)
 - ↳ latime de bandă

Hub

- dispozitiv care poate conecta mai multe dispozitive într-un LAN
- cele mai simple dispozitive de rețea
- nu au capacitatea de a limita traficul de la și către fiecare port (\Rightarrow toate dispozitivele conectate la hub împart aceeași latime de bandă)

Default Gateway

- adresa IP a routerului care conectează un LAN la internet sau la altă rețea
- în general, e aceeași cu adresa IP a routerului
 - ↳ el directează datele între rețele