



# LPIC-1. Examen 102. Objectiu 109

## LPI 109 Xarxes TCP/IP

[http://acacha.org/mediawiki/index.php/LPI\\_109.1](http://acacha.org/mediawiki/index.php/LPI_109.1)  
[http://acacha.org/mediawiki/index.php/LPI\\_109.2](http://acacha.org/mediawiki/index.php/LPI_109.2)  
[http://acacha.org/mediawiki/index.php/LPI\\_109.3](http://acacha.org/mediawiki/index.php/LPI_109.3)  
[http://acacha.org/mediawiki/index.php/LPI\\_109.4](http://acacha.org/mediawiki/index.php/LPI_109.4)



# Objectius

109.1. Fonaments dels protocols d'Internet	
	<ul style="list-style-type: none"><li>■ <b>Objectiu:</b> Els candidats han de demostrar que comprenen els fonaments de les xarxes <a href="#">TCP/IP</a></li><li>■ <b>Pes:</b> 4</li></ul>
	<p><b>Àrees Clau de Coneixement:</b></p> <ul style="list-style-type: none"><li>■ Demostrar que s'entenen les <a href="#">màscares de xarxa</a></li><li>■ Coneixements de les diferències entre adreces IP públiques i privades en format "<a href="#">dotted quad</a>"</li><li>■ Establir la ruta per defecte</li><li>■ Coneixement dels <a href="#">ports TCP i UDP més comuns</a> (20, 21, 22, 23, 25, 53, 80, 110, 119, 139, 143, 161, 443, 465, 993, 995)</li><li>■ Coneixement de les diferències i les característiques més importants dels protocols <a href="#">UDP</a>, <a href="#">TCP</a> i <a href="#">ICMP</a>.</li><li>■ Coneixement de les diferències més importants entre <a href="#">IPv4</a> i <a href="#">IPv6</a>.</li></ul>
	<p>La següent és una llista parcial de fitxers, termes i utilitats utilitzades:</p> <ul style="list-style-type: none"><li>■ <a href="#">/etc/services</a></li><li>■ <a href="#">ftp</a></li><li>■ <a href="#">telnet</a></li><li>■ <a href="#">host</a></li><li>■ <a href="#">ping</a></li><li>■ <a href="#">dig</a></li><li>■ <a href="#">traceroute</a></li><li>■ <a href="#">tracepath</a></li></ul>
	<p><b>Apunts:</b> <a href="#">LPI 109.1. Fonaments dels protocols d'Internet</a></p>



# Objectius

109.2. Configuració bàsica de la xarxa	
	<ul style="list-style-type: none"><li><b>Objectiu:</b> Els candidats han de ser capaços de consultar, canviar i verificar els paràmetres de configuració de xarxa en màquines client.</li><li><b>Pes:</b> 4</li></ul>
	<b>Àrees Clau de Coneixement:</b> <ul style="list-style-type: none"><li><a href="#">Configuració manual i configuració automàtica d'interfícies de xarxa</a></li><li><a href="#">Configuració TCP/IP bàsica de màquines</a></li></ul>
	La següent és una llista parcial de fitxers, termes i utilitats utilitzades: <ul style="list-style-type: none"><li><a href="#">/etc/hostname</a></li><li><a href="#">/etc/hosts</a></li><li><a href="#">/etc/resolv.conf</a></li><li><a href="#">/etc/nsswitch.conf</a></li><li><a href="#">ifconfig</a></li><li><a href="#">ifup</a></li><li><a href="#">ifdown</a></li><li><a href="#">route</a></li><li><a href="#">ping</a></li></ul>
	<b>Apunts:</b> <a href="#">LPI 109.2. Configuració bàsica de la xarxa</a>

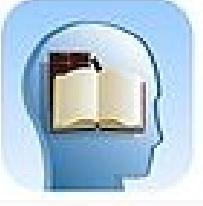


# Objectius

109.3. Resolució de problemes de xarxa bàsics	
	<ul style="list-style-type: none"><li><b>Objectiu:</b> Els candidats han de ser capaços de diagnosticar i solucionar problemes de xarxa a les màquines client.</li><li><b>Pes:</b> 4</li></ul>
	<b>Àrees Clau de Coneixement:</b> <ul style="list-style-type: none"><li><u>Configuració manual i automàtica de interfícies de xarxa i taules d'enrutament</u> incloent afegir, iniciar, aturar, tornar a iniciar, esborrar o reconfigurar interfícies de xarxa.</li><li><u>Modificar, consultar o configurar la taula de rutes i corregir de forma manual una ruta per defecte establerta de forma incorrecta.</u></li><li><u>Depurar problemes associats amb la configuració de la xarxa</u></li></ul>
	La següent és una llista parcial de fitxers, termes i utilitats utilitzades: <ul style="list-style-type: none"><li><u>ifconfig</u></li><li><u>ifup</u></li><li><u>ifdown</u></li><li><u>route</u></li><li><u>host</u></li><li><u>hostname</u></li><li><u>dig</u></li><li><u>netstat</u></li><li><u>ping</u></li><li><u>traceroute</u></li></ul>
	<b>Apunts:</b> <u>LPI 109.3. Resolució de problemes de xarxa bàsics</u>

# Objectius

## 109.4. Configuració del client DNS

	<ul style="list-style-type: none"><li><b>Objectiu:</b> Els candidats han de ser capaços de configurar <u>DNS</u> a una màquina client.</li><li><b>Pes:</b> 2</li></ul>
	<b>Àrees Clau de Coneixement:</b> <ul style="list-style-type: none"><li><u>Demostrar l'ús de DNS al sistema local</u></li><li><u>Modificar l'ordre en que és realitza la resolució de noms</u></li></ul>
	La següent és una llista parcial de fitxers, termes i utilitats utilitzades: <ul style="list-style-type: none"><li><u>/etc/hosts</u></li><li><u>/etc/resolv.conf</u></li><li><u>/etc/nsswitch.conf</u></li></ul>
	<b>Apunts:</b> <u>LPI 109.4. Configuració del client DNS</u>



# Arquitectures de comunicacions

## Arquitectures de comunicacions



# Protocol de comunicacions

És el conjunt de regles perfectament organitzades i convingudes per mutu acord que especifiquen l'intercanvi de dades o ordres durant la comunicació entre els nodes que formen part d'una xarxa

- ◆ Els protocols sovint són públics i són normatives o recomanacions d'associacions d'estàndards internacionals
- ◆ Un protocol (en societat o política) també és un conjunt de regles que cal observar pel que fa a l'etiqueta, les presidències, etc., en les cerimònies i en les relacions oficials
  - **Exemples:** Protocols IP, HTTP, TCP, FTP (la P és sempre de protocol)

**Atenció:** En la vida real els protocols no són obligatoris però en les comunicacions són **IMPRESCINDIBLES**

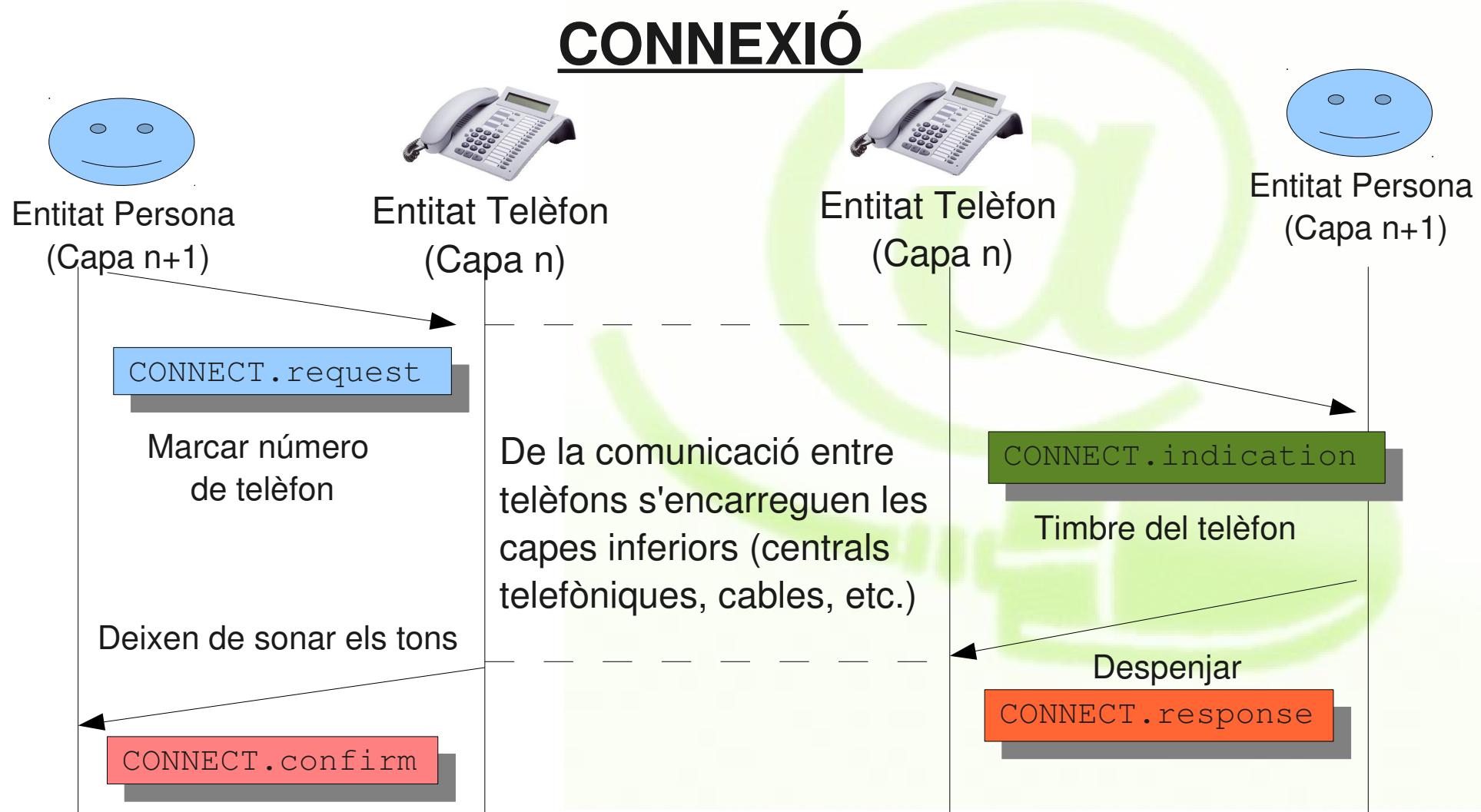
# Exemple de protocol. Telèfon

## → Passos d'una trucada

- ◆ Despenjar el telèfon a l'emissor
- ◆ Esperar to. Si no hi ha to tornar a penjar/despenjar
- ◆ Marcar el número de telèfon del receptor
- ◆ Si el receptor està ocupat sona el to de comunicant.
- ◆ Si el receptor no comunica, sonen els tons de trucada a l'emissor i el timbre al receptor.
- ◆ El receptor despenja el telèfon (accepta la comunicació)
- ◆ L'emissor i el receptor es comuniquen (comunicació full-duplex)
- ◆ L'emissor o el receptor finalitzen la comunicació penjant el telèfon

# Exemple de protocol. Telèfon

## Protocol de connexió telefònica





# Exemple de protocol telèfon

## ◆ Comentaris

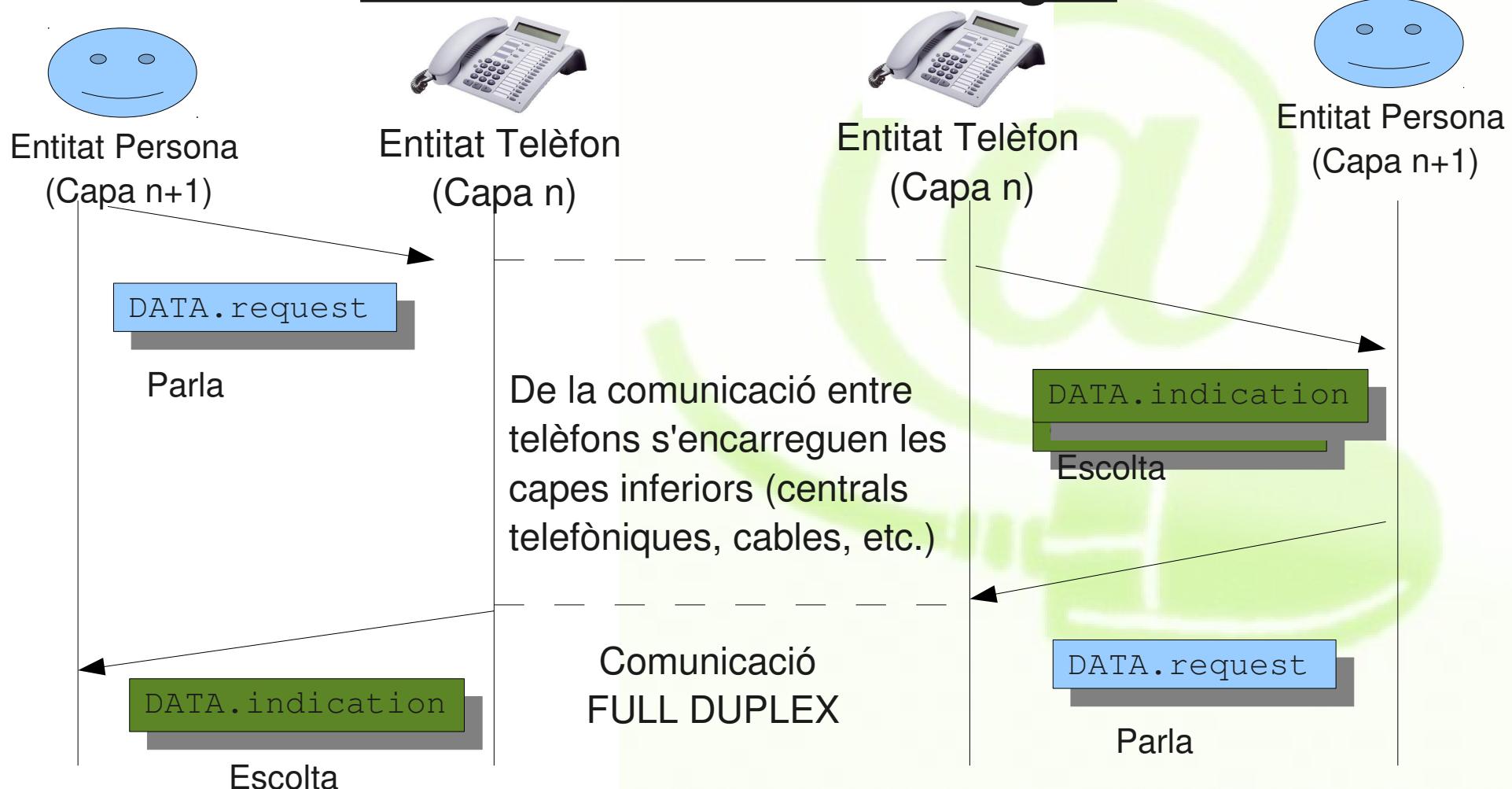
- ◆ El servei que hem vist és el servei de connexió (**CONNECT**) i utilitza totes les primitives fonamentals OSI per què **és un servei orientat a connexió i confirmat**.
- ◆ Fixeu-vos que les entitats (persones i telèfons) és comuniquen amb entitats iguals utilitzant protocols.
- ◆ Les persones (capa n+1) només necessitem saber com funcionen els telèfons (capa n). No ens cal conèixer la resta de capes
- ◆ El telèfons realment no es comuniquen directament entre si, si no que ho fan a través de les centrals telefòniques, cables, etc.



# Exemple de protocol. Telèfon

- **Intercanvi de missatges telefònics i desconexió**

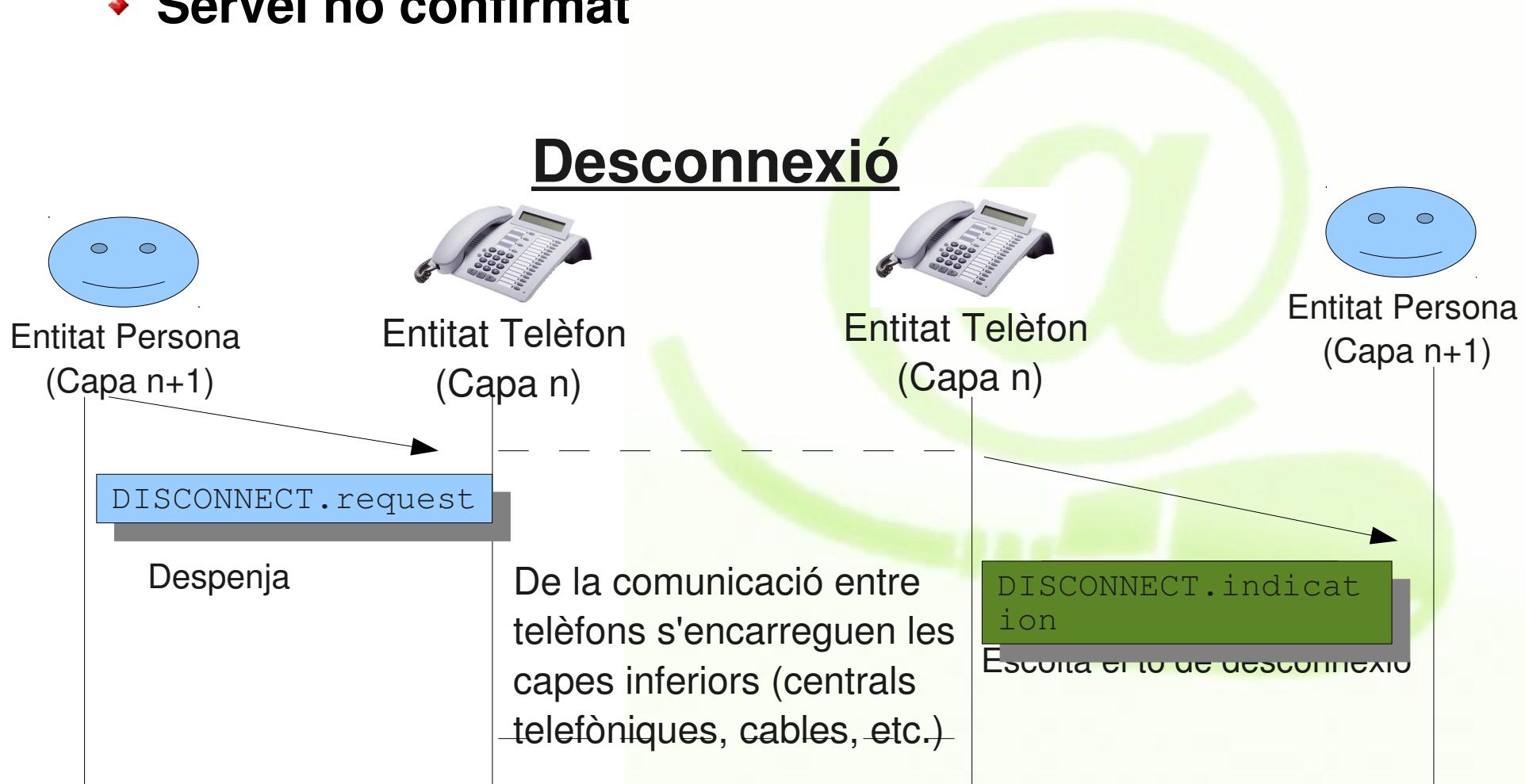
## Comunicació de missatges





# Exemple de protocol. Telèfon

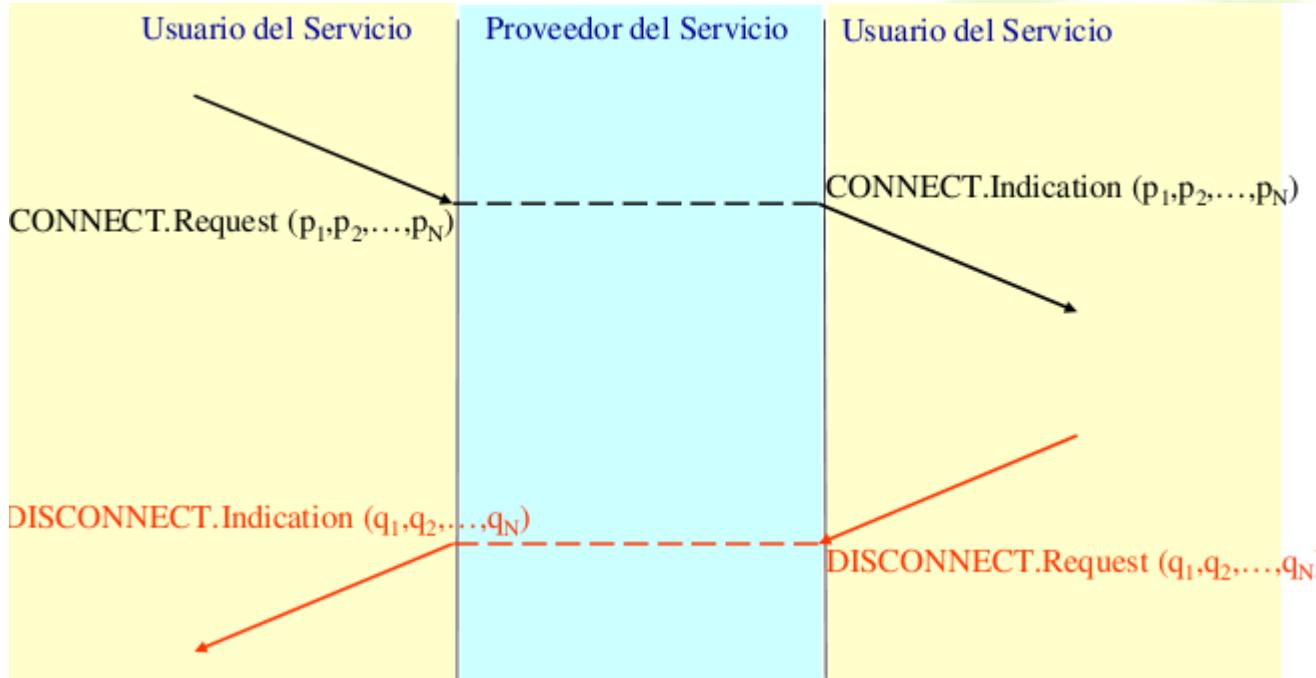
- Desconnexió telefònica
  - Servei no confirmat





# Exemples

- ◆ Que representa el següent diagrama?

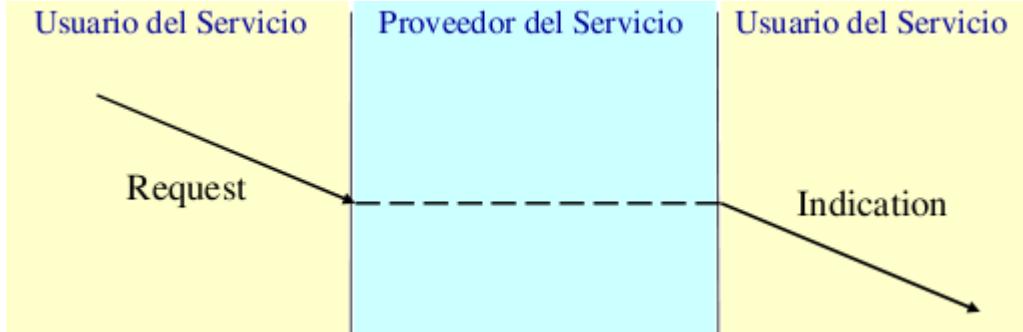


- ◆ Una connexió no acceptada per l'usuari remot (denegar trucada dels mòbils, comunicar...)

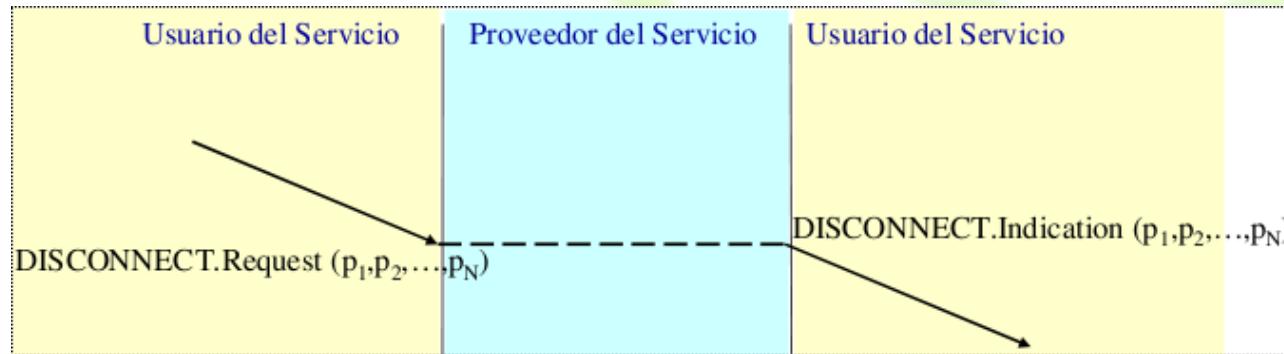


# Exemples

- ◆ Que representa el següent diagrama?



- ◆ Un servei qualsevol **NO CONFIRMAT**
- ◆ Que representa el següent diagrama?

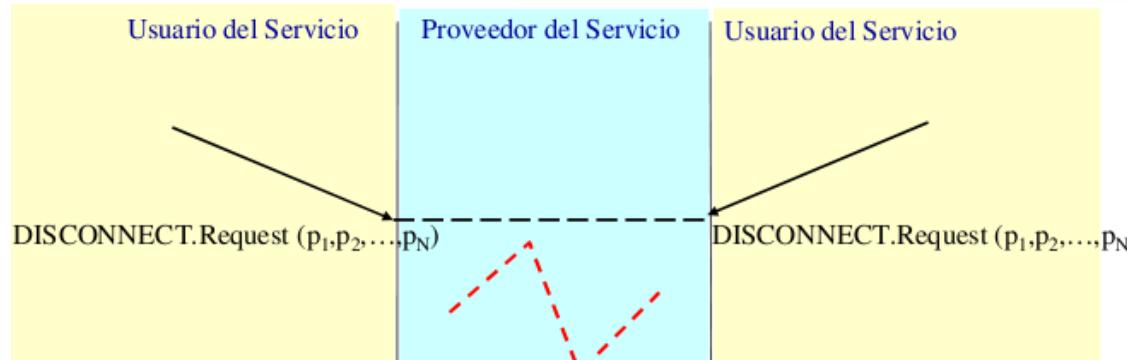


- ◆ Desconnexió iniciada per l'emissor

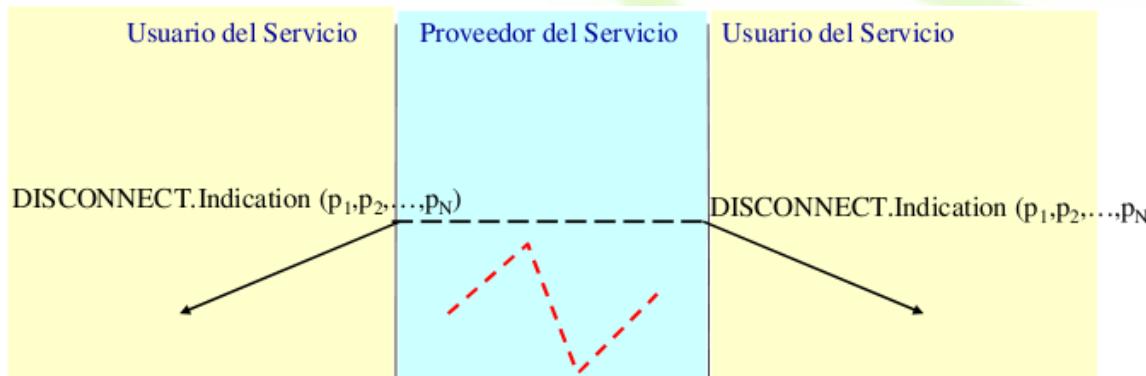


# Exemples

- ◆ Que representa el següent diagrama?



- ◆ Desconnexió iniciada per emissor i receptor
- ◆ Que representa el següent diagrama?

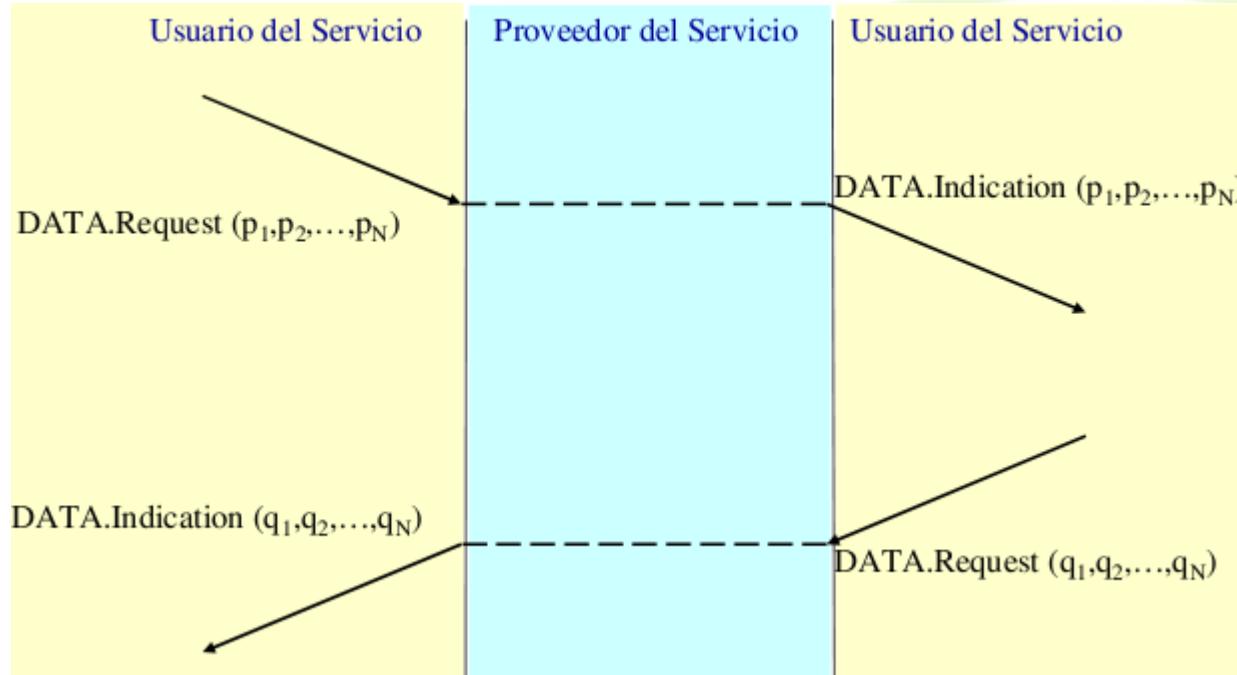


- ◆ Desconnexió iniciada pel proveïdor



# Exemples

- ◆ Que representa el següent diagrama?

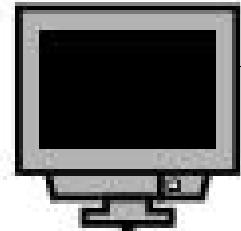


- ◆ Transferència de dades no confirmades

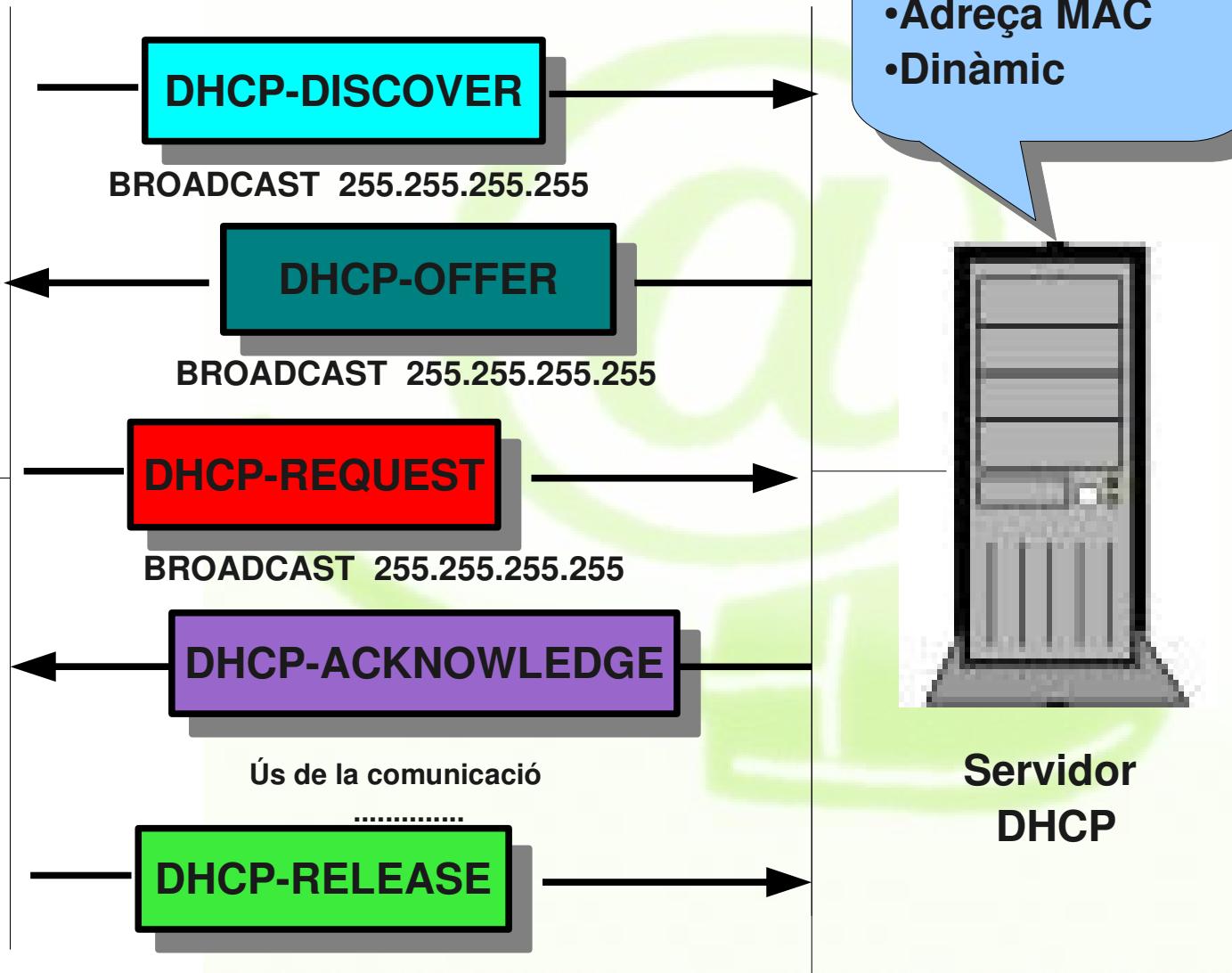
# Exemple de protocol. DHCP

## Funcionament del protocol

Oferta de configuració. Si l'accepta ho ha de notificar!



Client DHCP





# Funcions d'un protocol

- ❖ De protocols n'hi ha de molt variats (dels més simples als més sofisticats)
- ❖ Alguns exemples de funcions poden ser:
  - ❖ Detectar la connexió física existent
  - ❖ Establir una connexió (*handshake*)
  - ❖ Negociació de les característiques de la comunicació (velocitat, tassa d'error mínima, etc.)
  - ❖ Quan s'inicia o acaba un missatge i quin format té.
  - ❖ Que fer amb els missatges incorrectes o mal formats
  - ❖ Com acabar una connexió
  - ❖ Com detectar un final inesperat d'una connexió...



# Funcions d'un protocol

- ❖ **Els protocols determinen**
  - ◆ El **format** de les dades
  - ◆ La seva **temporització**
  - ◆ La seva **seqüènciació**
  - ◆ Com es dura a terme el control d'errors
- ❖ **Associacions que determinen els protocols**
  - ◆ **IEEE** (Institute of Electrical and Electronic Engineering)
  - ◆ **ANSI** (American National Standards Institute)
  - ◆ **TIA** (Telecommunications Industry Association)
  - ◆ **EIA** (Electronic Industries Alliance)
  - ◆ **ITU** (International Communications Union)



# Model arquitectònic de xarxa

És el conjunt organitzat de capes i protocols d'una xarxa

- ◆ L'objectiu d'aquesta organització de la xarxa és facilitar la tasca dels creadors de maquinari i programari per tal que puguin crear productes amb garanties de funcionament entre equips que segueixin les mateixes normes
- ◆ Les capes estan jerarquizades en capes superiors i capes inferiors. Cada capa ofereix els seus serveis a la capa superior.
- ◆ La gran majoria de models arquitectònics de xarxa actuals estan organitzats en capes



# Model arquitectònic de xarxa

## ◆ Capa o nivell

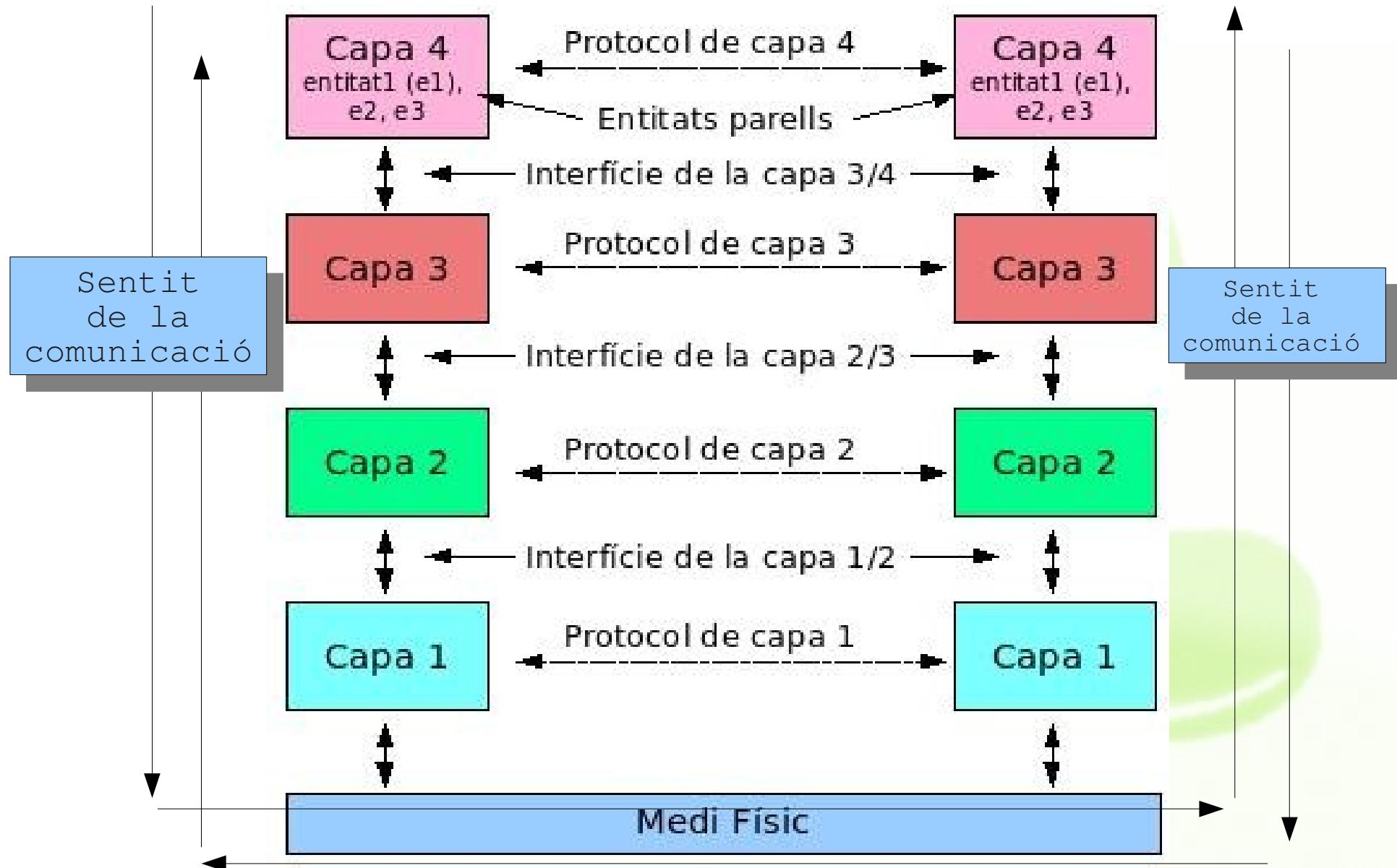
- ◆ En el disseny de xarxes s'utilitza sovint el concepte de capa o nivell com una forma d'estructurar les funcions i serveis que les xarxes proveeixen.
- ◆ Màxima: "**divideix i guanyaràs**". Per resoldre problemes grans sovint cal dividir el problema en parts petites.

## ◆ Interfície

Són les normes de comunicació entre dues capes consecutives

- ◆ La interfície és el conjunt de serveis i operacions que una capa inferior ofereix a una capa superior.
- ◆ Molts d'aquests conceptes **també són aplicables en altres àrees de la informàtica (programació – programació modular-, GUI, programació orientada a objectes, encapsulament)**

# Model arquitectònic de xarxa



# Models basats en nivells o capes

- ◆ La **capa n** ofereix una sèrie de **serveis** a la **capa n+1** a través de la seva **interfície**
- ◆ Els elements actius de cada capa s'anomenen **entitats**. Una entitat pot ser maquinari o programari
- ◆ Les **entitats d'una capa n d'un sistema** es comuniquen amb les **entitats de la capa n d'un altre sistema** conforme a una sèrie de regles preestablertes anomenades **protocol de capa n**.
- ◆ En un mateix sistema la comunicació entre dos capes es duu a terme a través d'una **interfície**.
- ◆ La implementació interna de cada capa depèn dels fabricants o desenvolupadors però sempre **s'ha de respectar la interfície i el protocol de capa**. A aquesta capacitat se l'anomena **Encapsulament (Caixa Negra)**.
- ◆ La primera capa és una excepció per què es comunica directament amb el medi físic.

# Models basats en nivells o capes

## ◆ Avantatges

- ◆ Permet que **diferents** tipus de maquinari i programari (de diferents empreses i desenvolupadors) es comuniquin entre si.
- ◆ Augmenta la **interoperabilitat** dels sistemes.
- ◆ Impedeix que els canvis en una capa afectin a la resta de capes (**encapsulament, caixa negra**)
- ◆ Facilita la **normalització** i **estandardització** dels components de xarxa
- ◆ Divideix la complexitat d'una comunicació a través d'una xarxa en parts més petites.
- ◆ És fàcil incorporar millores i nous serveis per què no afecten a totes les capes.

# Avantatges i Inconvenients

## → **Inconvenients**

- ◆ Cada capa que s'introdueix augmenta la complexitat del sistema.
- ◆ El model TCP/IP té menys capes i es va orientar més a les capes baixes (físic/enllaç i xarxa) i va deixar menys normativitzat les capes altes (transport i aplicació).
- ◆ És impossible cobrir tots els casos amb capes. Cas de les subcapes MAC i subcapa d'enllaç del nivell dos en xarxes amb medi compartit (P. ex Ethernet).

# Encapsulament

## ❖ Capes

- ❖ Cada capa té dades + informació de control

## ❖ Model de Caixa Negra. Cada capa és una caixa negra

- ❖ Per què és bo no saber com resol els problemes (dona serveis) una capa inferior?
  - Per què així no podem fer suposicions al nivell superior que després no es compleixen si canviem la caixa negra.
  - Vist d'un altre manera, si ens canvien la caixa negra per un altre que és diferent per dins però igual per fora ens donàrem compte?
  - **La resposta és NO** i això es precisament el que volem. Si canvia la targeta de xarxa l'ordinador hauria de funcionar igualment no? O si canviem un Switch per un altre la xarxa ha de continuar funcionant.





# Història

## • Inicis

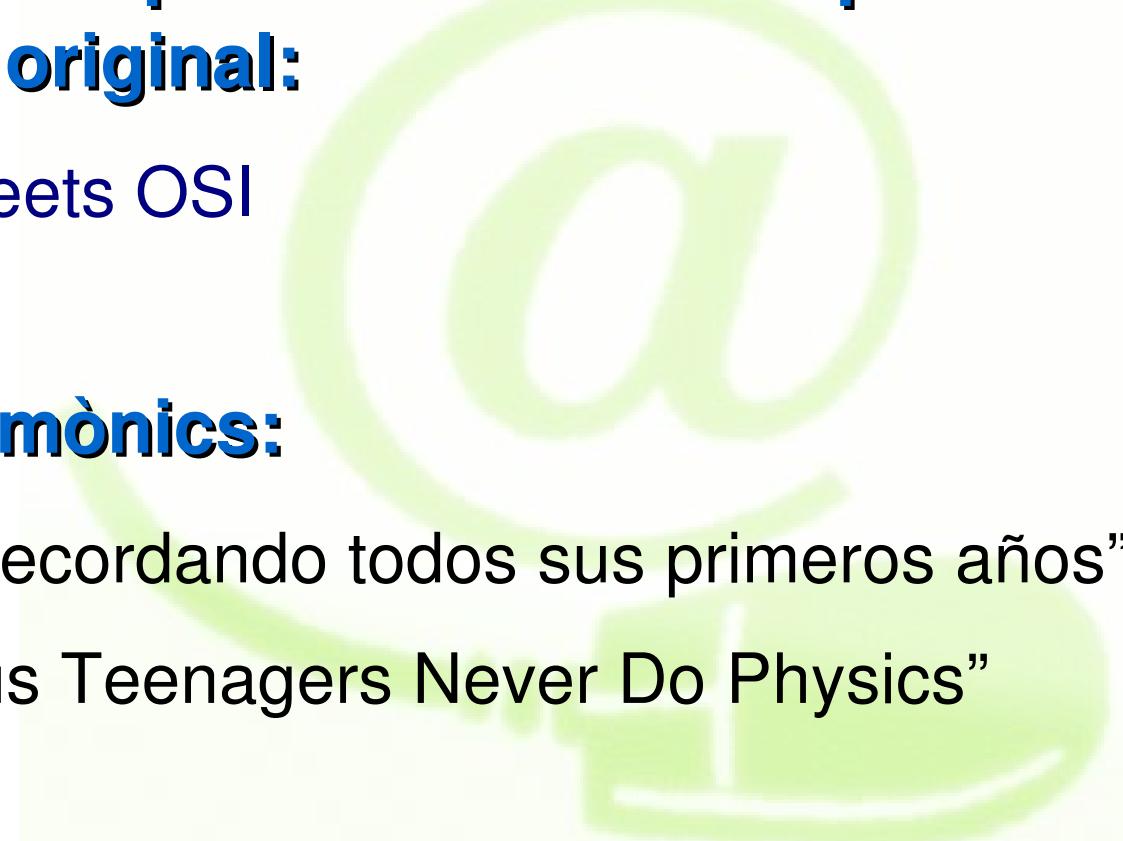
- ◆ Les xarxes van néixer sense una forma de fer comuna.
- ◆ A principis dels anys 80 es va disparar l'existència de xarxes i van començar a sorgir grans problemes de compatibilitats.

## • Organització Internacional per l'Estandardització (ISO)

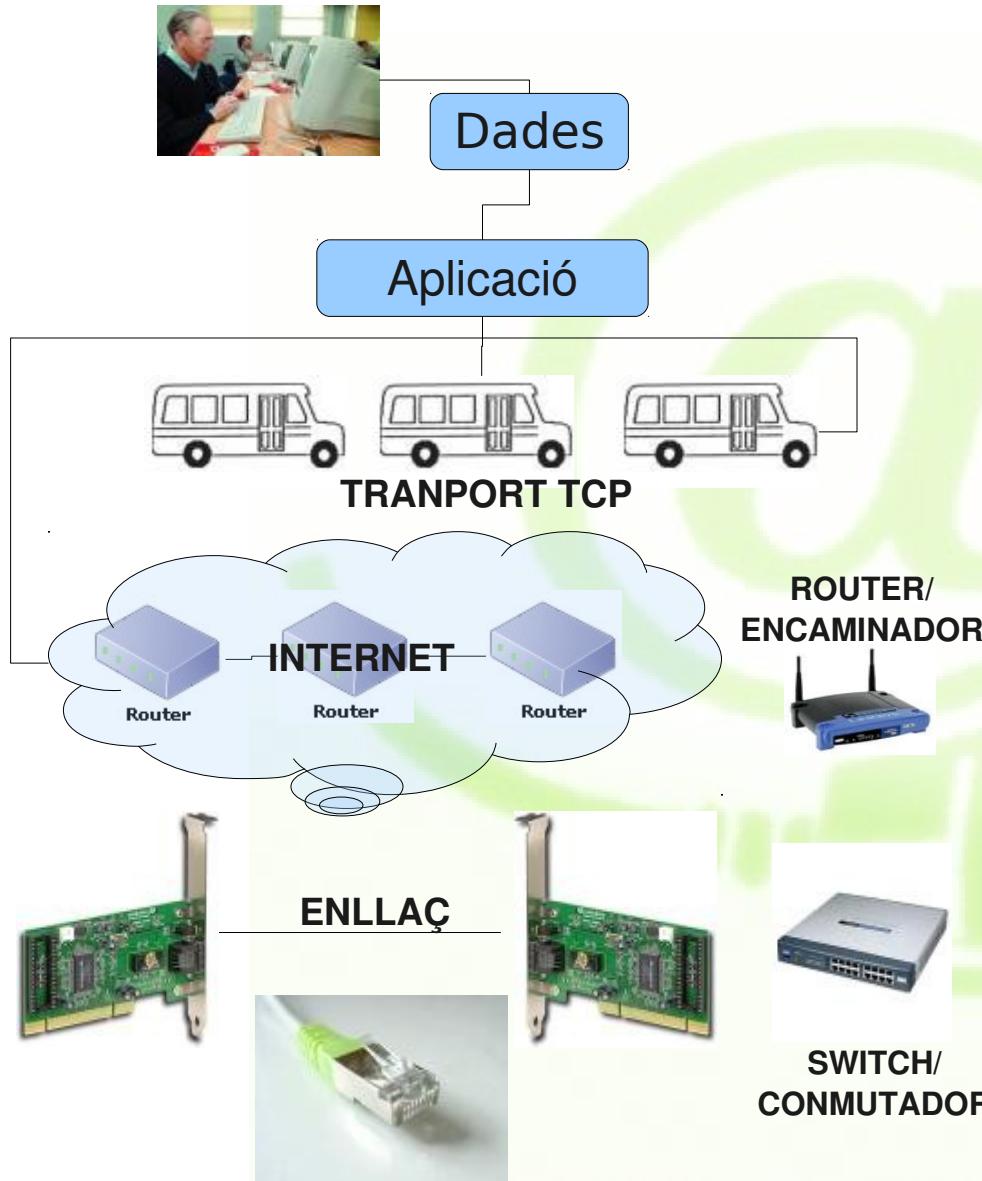
- ◆ Aquesta organització va decidir posar ordre establint-hi un **model estàndard de referència** --> **Model OSI**
  - **Open Systems Interconnection Basic Reference Model**
- ◆ El model TCP/IP es va desplegar més ràpid.
- ◆ Tot i això, OSI s'utilitza com a model de referència per ensenyar xarxes.

# James Bond utilitza el model OSI

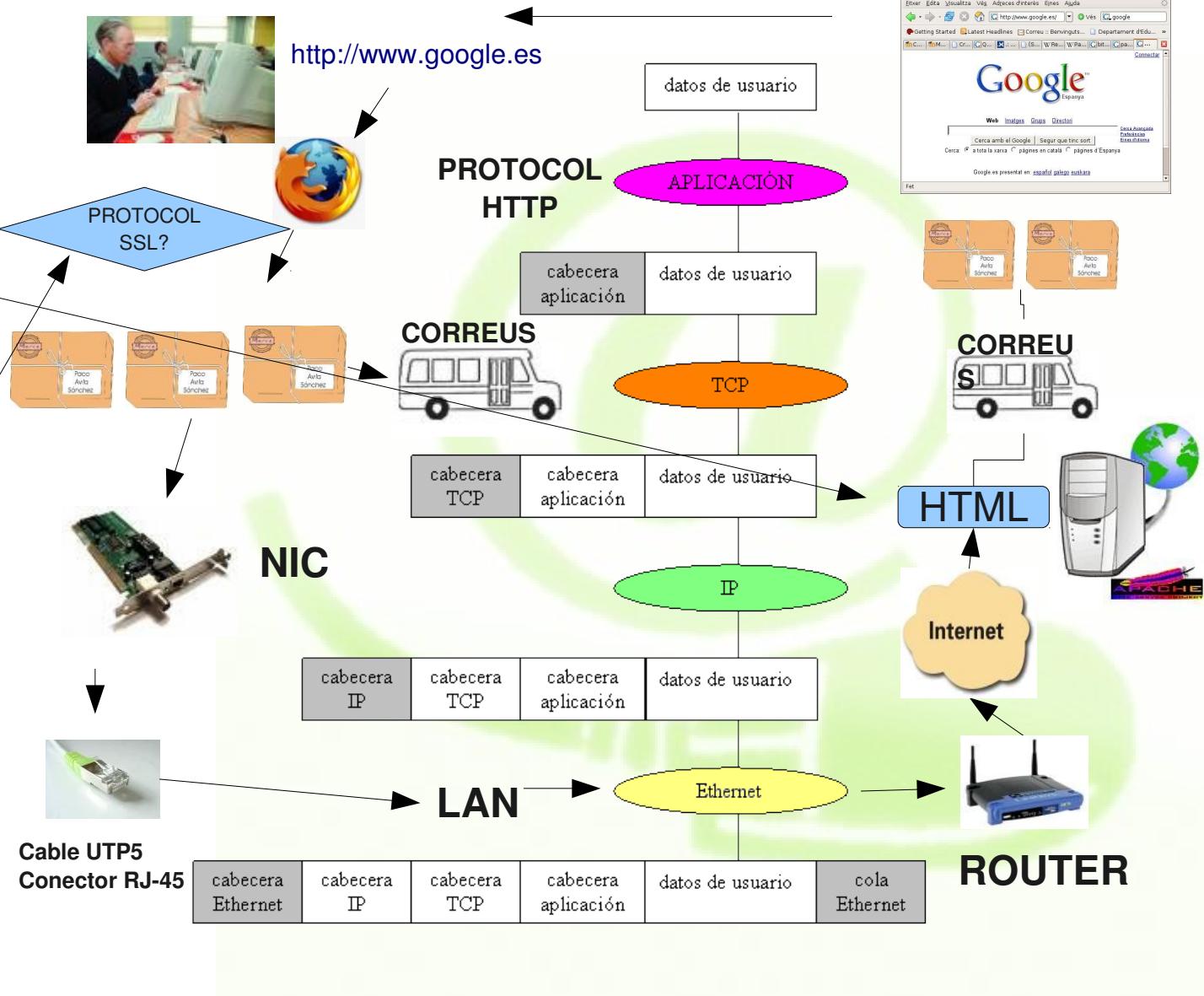
- **Hi ha mil formes d'explicar el model OSI però aquesta és força original:**
  - ◆ James Bonds Meets OSI
- **Hi han força Mnemònics:**
  - ◆ “Fernando esta recordando todos sus primeros años”
  - ◆ “All Pretty Serious Teenagers Never Do Physics”
- **Vídeos:**
  - ◆ Warriors of the Net



# Model OSI



# Model OSI. Bits, Paquets, Trames ...



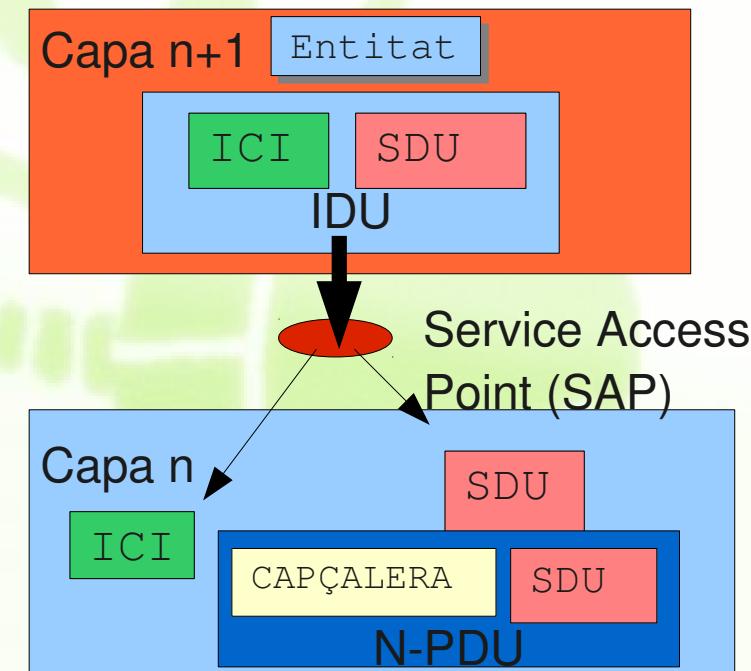
# Comunicació entre capes (serveis)

## • Serveis del telèfon

- ◆ Transmissió de veu i/o Transmissió de dades
- ◆ Trucada en espera, trucada a tres, accés al contestador...
- ◆ Cada servei té un protocol diferent de funcionament

## • Conceptes

- ◆ SAP: Punt d'accés a serveis
- ◆ IDU: Interface Data Unit
  - SDU: Service Data Unit
  - ICI: Interface Control Information
- ◆ PDU: Protocol Data Unit
  - Capçalera: Informació de control del protocol



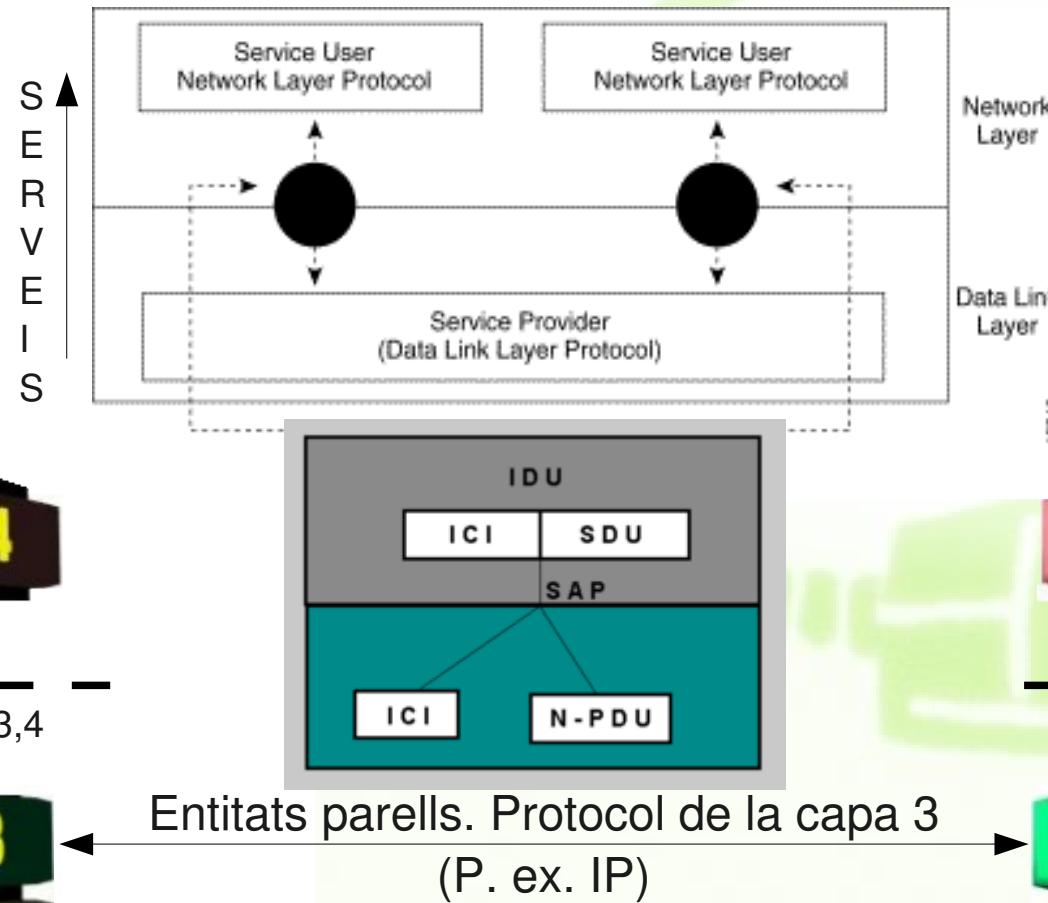
# Serveis, Primitives, Interfícies i Entitats

Els nivells inferiors ofereixen serveis als superiors a través dels anomenats Service Acces Points **SAPS**

Cada servei es detalla en una sèrie de **primitives** de servei, les quals permeten establir un diàleg entre les dues capes.

**PRIMITIVES**

REQUEST  
INDICATION  
RESPONSE  
CONFIRM





# Definicions

## ◆ **Entitats.**

- ◆ Són els elements actius que hi ha a cadascuna de les capes. Hi ha entitats de programari (com els processos) o entitats de maquinari (com els xips encarregats de fer la E/S de dades). Les entitats d'una mateixa capa però localitzades en màquines diferents són anomenades entitats parells.

## ◆ **Punt d'accés al servei (SAP).**

- ◆ Els SAP són els punts d'entrada (“portes”) en els que una capa pot trobar disponibles els serveis de la capa immediatament inferior. Cada SAP té una adreça que l'identifica.



# Definicions

- ❖ **Unitat de dades de la interfície IDU.**
  - ❖ És el bloc informatiu que l'entitat de capa N passa a l'entitat de capa N -1 a través de la interfície N/N-1.
- ❖ **Unitat de dades del servei SDU.**
  - ❖ Cada IDU està compost per un camp amb informació per al control de la interfície (camp ICI) i de un segon camp anomenat SDU, que és la informació que es passa a través de la xarxa a l'entitat parella, és a dir, al seu equivalent en el host destinatari.
- ❖ **Unitat de dades del protocol PDU.**
  - ❖ La informació del SDU no sempre es pot transmetre en directe. De vegades cal fraccionar-la ja que la seva mida és massa gran i sempre s'ha d'afegir informació de control.

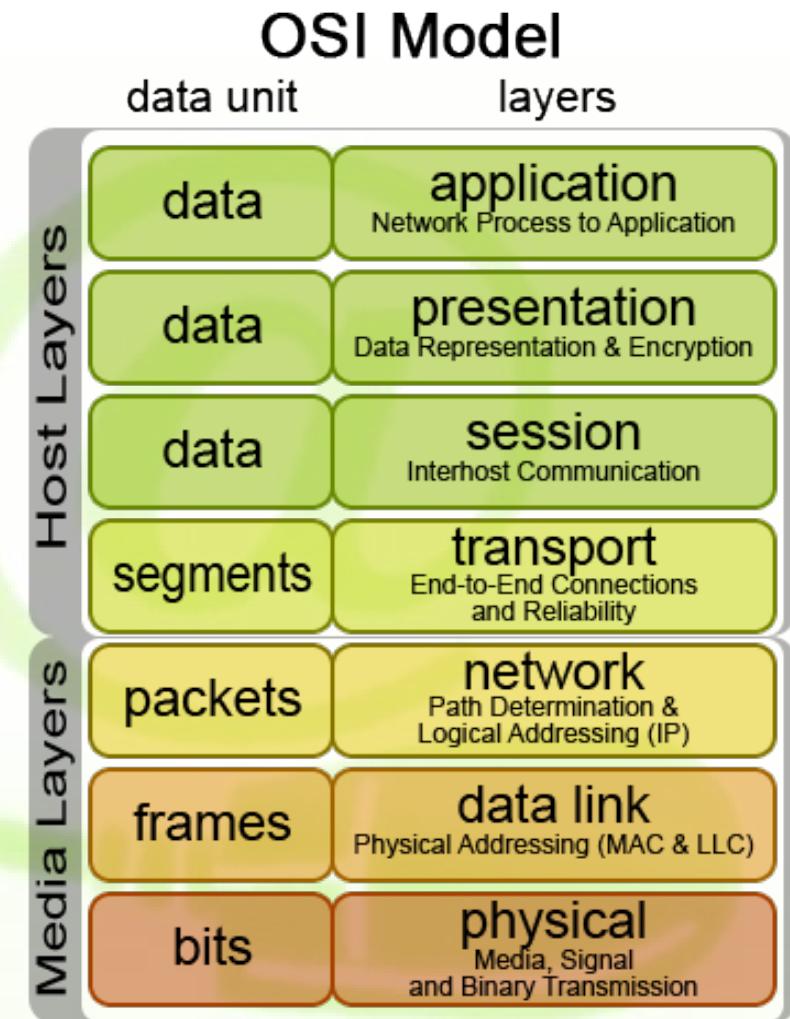
# Unitats de dades

- **Cada capa té un tipus de dades**

- ◆ Unitats de dades
  - Dades pures
  - Segments
  - Packets (Paquets)
  - Frames (Trames)
  - Bits

- **Overhead**

- ◆ Les comunicacions són “més lentes” (**overhead**) del que haurien de ser.



# Sistemes oberts

Son aquells sistemes capaços d'interconnectar-se amb altres sistemes d'acord amb unes normes preestablertes i obertes

- ◆ El concepte de sistema obert va ser proposat inicialment per la **ISO** (*International Organization for Standardization*)
- ◆ La ISO ha proposat el model arquitectònic OSI (*Open Systems Interconnection*) que s'encarrega de definir un model genèric per a la interconnexió de Sistemes Oberts
- ◆ El contrari a un sistema obert seria un **sistema patentat i tancat**.



# Importància dels estàndards oberts

- **Què succeiria si no haguessin models i estàndards en les comunicacions?**



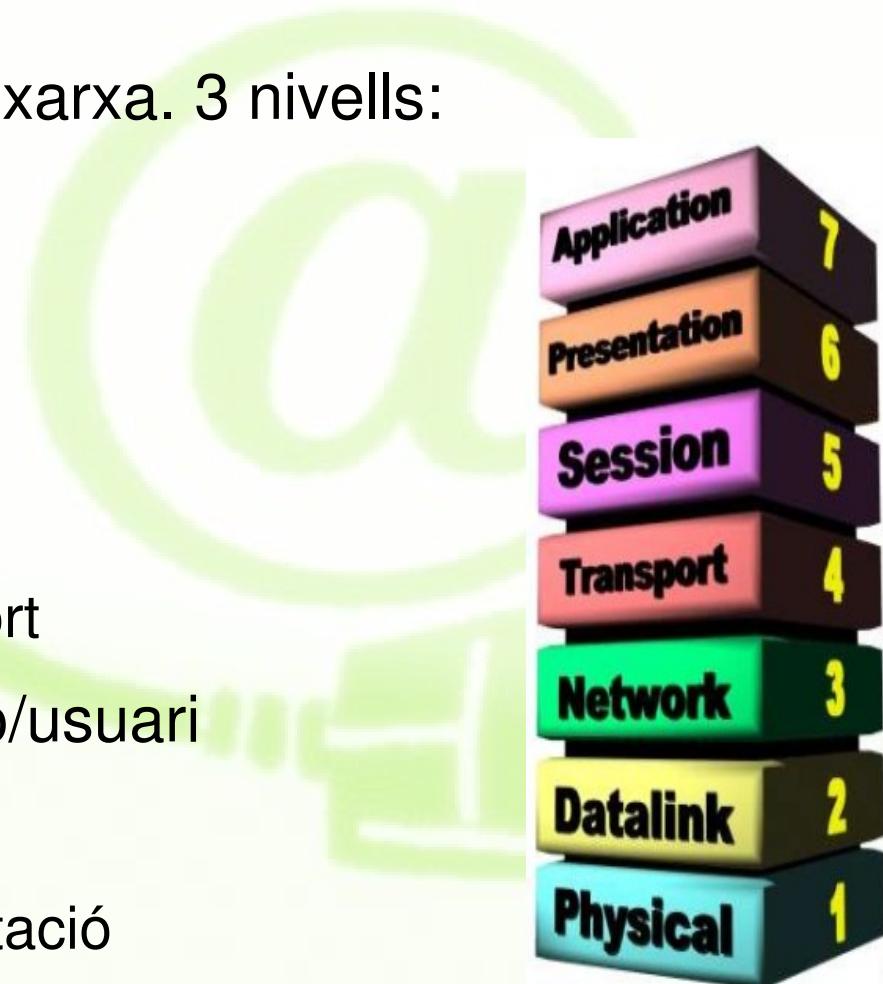
EL CAOS! simbolitzat per la història bíblica de la **TORRE DE BABEL**

La ISO s'encarrega de definir coses com la mida dels papers, noms de llengües i codis de països, codis de divises, codificació de caràcters, ISOS de qualitat, llenguatge de programació C i BASIC, MPEG, ISO 26300 (*Open Document*)

# L'estructura de capes OSI

- **El model de referència també defineix l'arquitectura de capes. Hi ha 7 nivells**

- ◆ Nivells o capes orientats a xarxa. 3 nivells:
  - **Nivell 1.** Nivell Físic
  - **Nivell 2.** Nivell d'enllaç
  - **Nivell 3.** Nivell de xarxa
- ◆ Nivell de transport
  - **Nivell 4.** Nivell de transport
- ◆ Nivells orientats a aplicació/usuari
  - **Nivell 5.** Nivell de sessió
  - **Nivell 6.** Nivell de presentació
  - **Nivell 7.** Nivell d'aplicació



# L'estructura de capes OSI

## ◆ Propietats

- ◆ Una capa és un nivell d'abstracció lògic que defineix un conjunt d'operacions funcions o serveis. Existeixen tantes capes com serveis
- ◆ Cada capa té una funció o servei perfectament definit
- ◆ Les funcions es defineixen de forma que siguin fàcil de crear estàndards
- ◆ Cal disminuir al màxim el flux d'informació entre capes
- ◆ Hi ha tantes capes com sigui necessari de forma que no hi hagin funcions molt diferents en una mateixa capa.



# Nivells o capes orientats a xarxa

Nivells o capes orientats a xarxa



# Nivell 1. Nivell Físic

- ❖ **És la capa de més baix nivell i s'ocupa de la transmissió dels bits.**
- ❖ **S'ocupa de definir:**
  - ❖ Característiques mecàniques (connectors, nombre de pins, tipus de cables...)
  - ❖ Característiques elèctriques/electromàgnetiques (senyals elèctriques a utilitzar, duració i voltatge de les senyals...)
  - ❖ Característiques funcionals (interfícies de connexió al medi i funcions) i de procediment (fases i estats)
  - ❖ ...per a poder establir i destruir connexions entre dos equips de la xarxa.
  - ❖ **Estàndards i protocols:** RS-232, xarxes Ethernet, etc.

# Nivell 2. Nivell d'enllaç

## → **Controla l'intercanvi de dades entre dues màquines connectades directament per un medi físic (enllaç)**

- ◆ Encarregat d'establir una línia de comunicació lliure d'errors que pugui ser utilitzada per la capa immediatament superior (capa de xarxa).
- ◆ No treballa amb bits directament. Treballa amb blocs de dades de nivell 2 (2-PDU) anomenats **trames**. Els missatges es formen per una o més trames.
- ◆ Les trames s'envien seqüèncialment per la línia de transmissió utilitzant els serveis de la capa física.
- ◆ La gestió de les trames (tractament d'errors, eliminar errors, retransmetre, descartar trames duplicades, etc.) és l'objectiu d'aquest nivell.



# Nivell 2. Nivell d'enllaç

## • Funcions

- ◆ **Sincronització** de les trames (identificar inici i final de trames)
- ◆ **Control de flux**: estableix el ritme de transmissió de dades per evitar sobrecàrregues.
- ◆ **Control d'errors**: permet comprovar que les dades que arriben al receptor són idèntiques a les que ha enviat l'emissor.
- ◆ **Direcccionament**: En medis de difusió (que no són punt a punt) és necessari identificar els components (MAC)
- ◆ **Gestió de l'enllaç**: per controlar la transmissió cal enviar dades “pures” i dades de control de l'enllaç.



# Nivell 3. Nivell de xarxa

## ◆ **Control de la xarxa/subxarxa**

- ◆ Treballa amb blocs de dades de xarxa (3-PDU) anomenats paquets.

## ◆ **Funcions**

- ◆ **Encaminament:** Determinar la ruta (nodes de xarxa pels quals circular) més adequada per als paquets
- ◆ **Identificació:** Els nodes han de tenir una identificació única que els permeti distingir dels altres nodes i localitzar-los a la xarxa.
- ◆ **Control de la congestió:** determina quins són els camins menys congestionats (similar al trànsit rodat)
- ◆ **Interconnexió de xarxes**
- ◆ **Protocol:** IP (Internet Protocol)



# Nivell de transport

Nivell de transport



# Nivell 4. Nivell de transport

## • Capa de transició que connecta les aplicacions i/o usuaris amb la xarxa

- ◆ És una capa de transició entre els nivells orientats a la xarxa i els orientats a les aplicacions
- ◆ Treballa amb unitats de dades 4-PDU també anomenades TPDU o **segments**.
- ◆ Té funcions similars al nivell d'enllaç (salt a salt) però entre dues màquines que no estan connectades directament (extrem a extrem)
- ◆ S'encarrega de preparar les dades de les aplicacions per a la xarxa i assegurar-se que arribaran correctament al nivell de transport del destinatari.
- ◆ **Protocols:** TCP (Transport Control Protocol) i UDP (User Datagram Protocol)



# Nivell 4. Nivell de transport

## ◆ Funcions

- ◆ Ofereix els serveis que no ofereix el nivell de xarxa
  - **Establiment de connexió:** en serveis orientats a connexió.
  - **Reordenació de paquets:** serveis no orientats a connexió.
  - **Control d'errors:** recuperació de caigudes de xarxa, reenviament de paquets, etc.
  - **Control de flux:** Implementació de buffers.
  - **QoS (Quality of Service):** Garanteix la fiabilitat i la qualitat del servei.
- ◆ **Multiplexació de connexions:** Permet tenir més d'una connexió oberta a través d'un mateix medi físic.  
S'utilitzen **ports** i el concepte de **sockets**.



# Nivells orientats a aplicació/usuari

Nivells orientats a usuari/aplicació



# Nivell 5. Nivell de sessió

- ◆ **Permet el diàleg entre emissor i receptor establint una sessió.**
  - ◆ A través d'una sessió es pot portar a terme un transport de dades ordinari (nivell de transport).
  - ◆ Millora els serveis de la capa de transport:
    - Transmetre un fitxer gran per una línia telefònica que té caigudes cada 15 minuts. La capa de sessió es pot encarregar de la resincronització de la transferència, de manera que a la següent connexió es transmetin dades a partir de l'últim bloc tramés sense error.
    - Restauració d'un estat anterior (sessió).
  - ◆ Aquest nivell sovint es dilueix dins del nivell d'aplicació

# Nivell 6. Nivell de presentació

- ❖ **S'ocupa de la sintaxi i de la semàntica de la informació que es desitja transmetre.**
  - ◆ Coordina la representació de dades entre emissor i receptor
    - Per exemple, un emissor que utilitza el codi ASCII i un receptor que utilitza UNICODE. Necessitem un servei de conversió i interpretació de dades que normalment proveeix la cap de presentació.
  - ◆ Sovint es tracta més de llenguatges (XML, HTML) o protocols d'interpretació de dades o formats (MIME)
  - ◆ Altres serveis:
    - **Compressió de dades**
    - **Xifratge de dades** (Protocol SSL)
  - ◆ També és una capa que normalment es dilueix dins la capa d'aplicació

# Nivell 7. Nivell d'aplicació

- **És la capa superior de la jerarquia OSI i on es defineixen els protocols que utilitzaran les aplicacions i processos dels usuaris.**
  - ◆ Quan dos processos es volen comunicar i resideixen en el mateix ordinador, utilitzen el Sistema Operatiu per comunicar-se (Linux utilitza dispositius especials de xarxa – loopback – per comunicar processos)
  - ◆ Si resideixen en ordinadors diferents, la capa d'aplicació dispararà els mecanismes adequats per produir la connexió entre els processos, servint-se dels serveis de les capes inferiors.
  - ◆ **Exemples:** DNS, FTP, HTTP, IMAP, IRC, NFS, NNTP, NTP, POP3, SMB/CIFS, SMTP, SNMP, SSH, Telnet, SIP...

# Nivell 7. Nivell d'aplicació

## • **5 grups de protocols definits per la ISO:**

- ◆ **Grup 1.** Protocols de gestió del sistema. Orientats a la gestió del propi sistema de interconnexió dels ordinadors a la xarxa.
- ◆ **Grup 2.** Protocols de gestió de l'aplicació. Porten el control de la gestió d'execució de processos: bloquejos, accessos indeguts, assignació i càlcul de recursos, etc.
- ◆ **Grup 3.** Protocols de sistema. Gestionen les tasques del sistema operatiu com el accés a fitxers, la comunicació entre tasques o processos, l'execució de tasques remotes, etc...
- ◆ **Grups 4 i 5.** Protocols específics per aplicacions. Depenen absolutament de les aplicacions que els utilitzen.



# Protocols





# Protocols

## NIVELL DE XARXA

NetBEUI, OSPF, RIP, EIGRP, IP, IPX, ...

## NIVELL D'ENLLAÇ

Ethernet, Token Ring, LocalTalk, FDDI, X.21, X.25, Frame Relay, BitNet, CAN, ATM, Wi-Fi, HDLC, SDLC, CSMA/CD, CSMA/CA, ...

## NIVELL FÍSIC

Codis NRZ, Codificació Manchester, Cable coaxial, Par trenat, 10Base2, 10BASE5, 10BASE-T, 100BASE-TX, PDH, SDH, T-carrier, E-carrier, SONET, DSSS, FHSS, ...

# La família de protocols TCP-IP

El model de referència OSI és un MODEL TEÒRIC.  
No hi ha ninguna arquitectura de xarxa que sigui 100% OSI

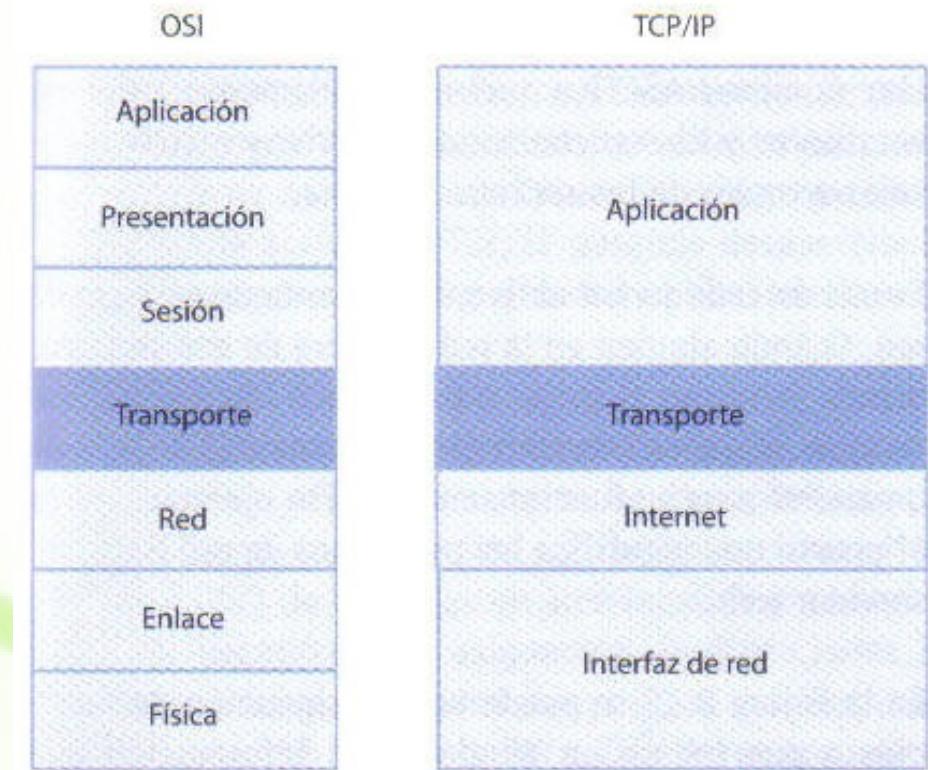
## • Quina arquitectura és doncs la més utilitzada?

L'arquitectura més utilitzada és **TCP/IP** o la **família de protocols d'Internet**, que és el conjunt de protocols de xarxa en que esta basat Internet.

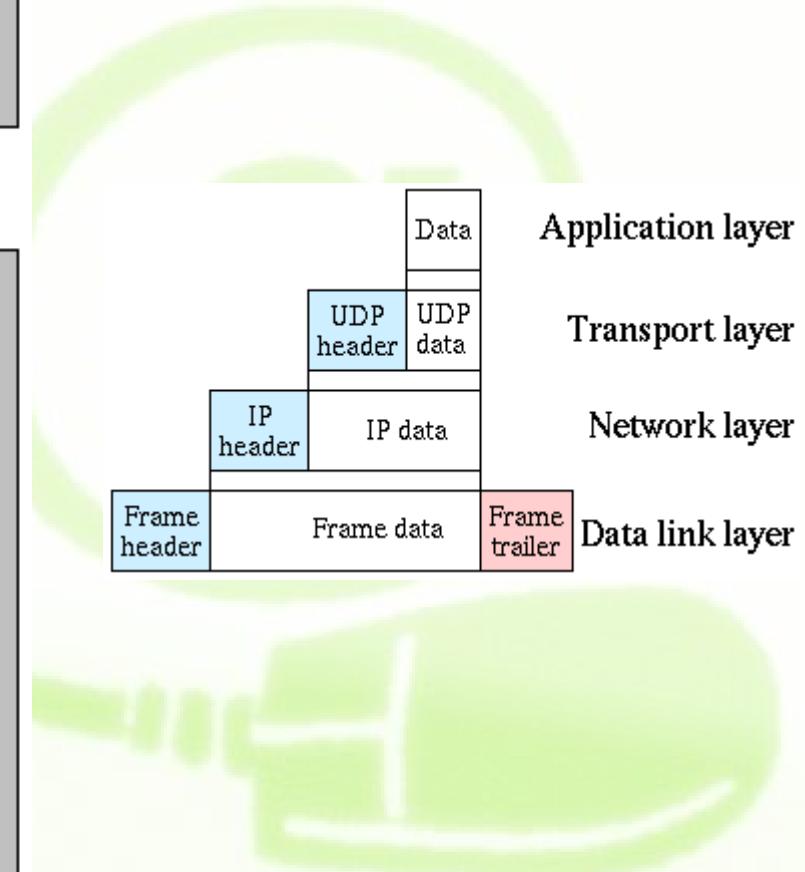
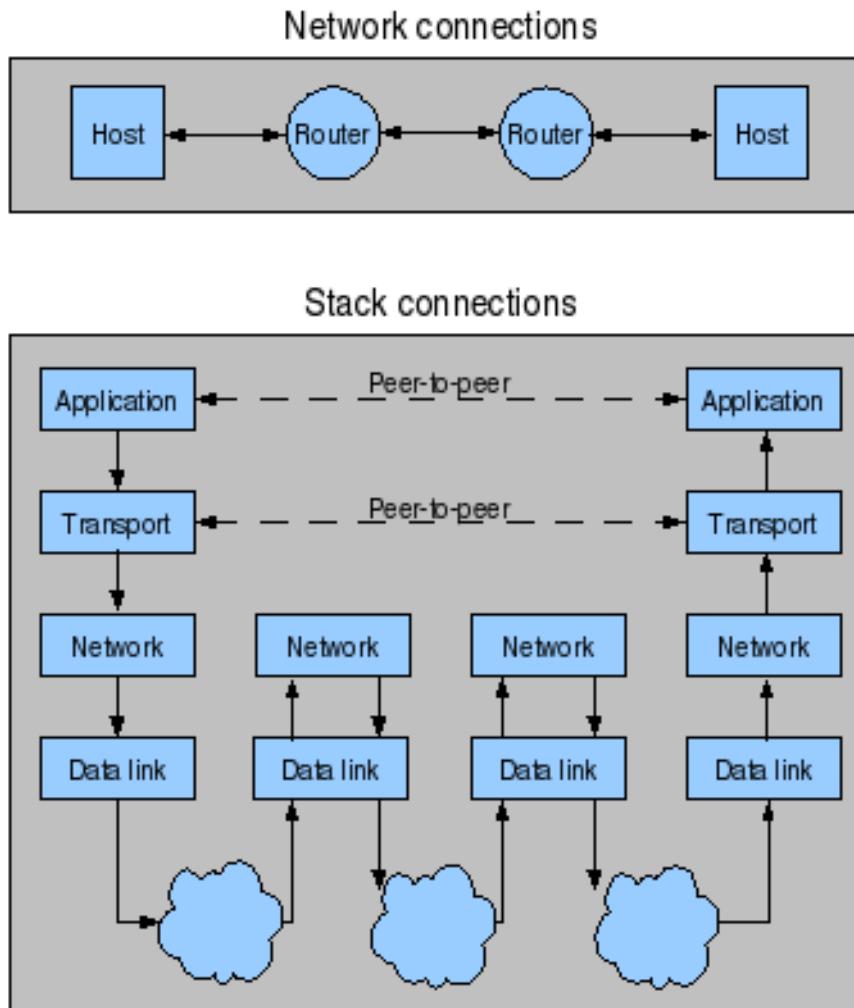
- ◆ També anomenada la família de protocols d'Internet
- ◆ Anomenada així pels dos protocols més importants (i els que es van definir primer): **Protocol de Control de Transmissió (TCP)** y **Protocol d'Internet (IP)**
- ◆ Hi han més de 100 protocols diferents en aquesta família (**HTTP, ARP, FTP, SMTP, POP, IMAP, TELNET, SSH, etc.**)
- ◆ El seu origen és la xarxa militar ARPANET.

# TCP/IP vs OSI

- ◆ TCP/IP redueix la complexitat a 4 capes i canvia el nom a algunes capes.
- ◆ La unió de les capes superiors en una sola capa d'aplicació té sentit des de la perspectiva de xarxa.
- ◆ La separació en subcapes de la capa d'aplicació que fa OSI té més sentit per a desenvolupadors de programari.
  - ◆ Les capes de Transport i Internet amb els seus respectius protocols (TCP i IP) donen nom a la família.



# Capes TCP/IP i encapsulament



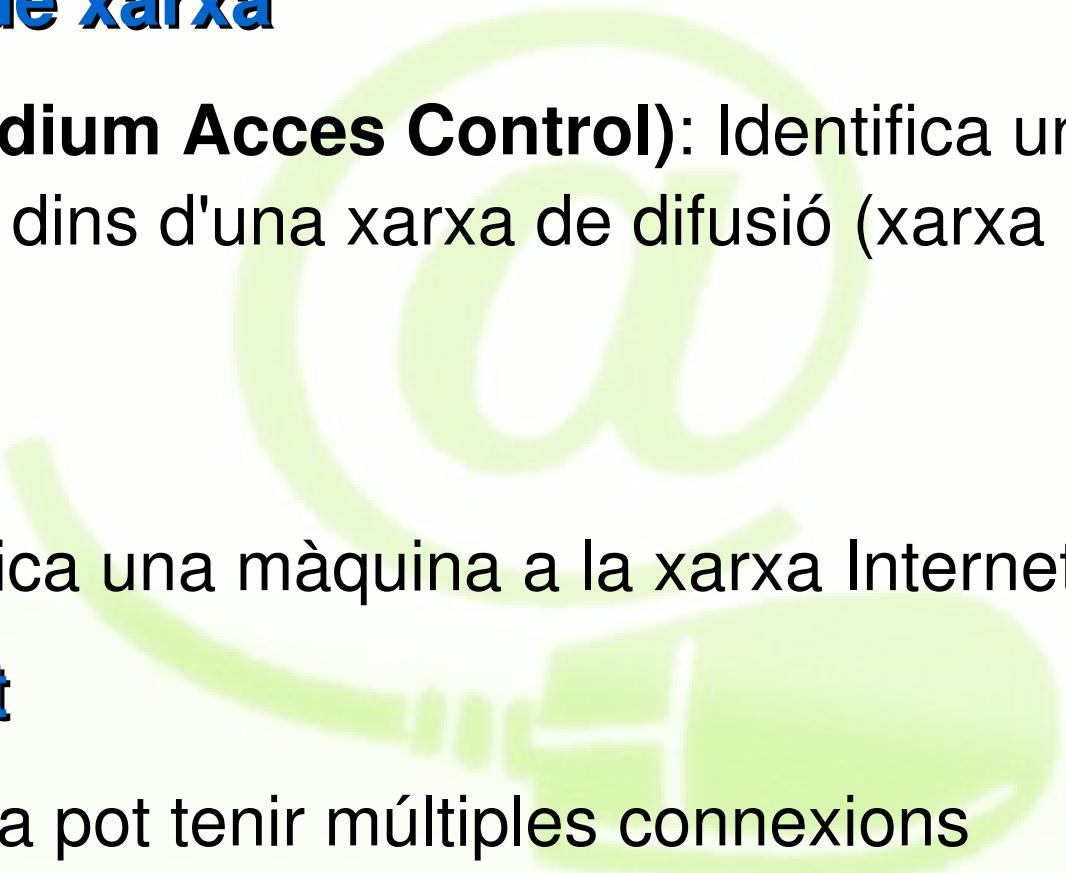


# TCP/IP Layers

<b>4. Application</b>	DNS, TFTP, TLS/SSL, FTP, HTTP, IMAP, IRC, NNTP, POP3, SIP, SMTP, SNMP, SSH, TELNET, ECHO, BitTorrent, RTP, PNRP, rlogin, ENRP, ...  Routing protocols like BGP and RIP, which for a variety of reasons run over TCP and UDP respectively, may also be considered part of the application or network layer.
<b>3. Transport</b>	TCP, UDP, DCCP, SCTP, IL, RUDP, ...  Routing protocols like OSPF, which run over IP, may also be considered part of the transport or network layer. ICMP and IGMP run over IP may be considered part of the network layer.
<b>2. Internet</b>	IP (IPv4, IPv6)  ARP and RARP operate underneath IP but above the link layer so they belong somewhere in between.
<b>1. Network access</b>	Ethernet, Wi-Fi, token ring, PPP, SLIP, FDDI, ATM, Frame Relay, SMDS, ...

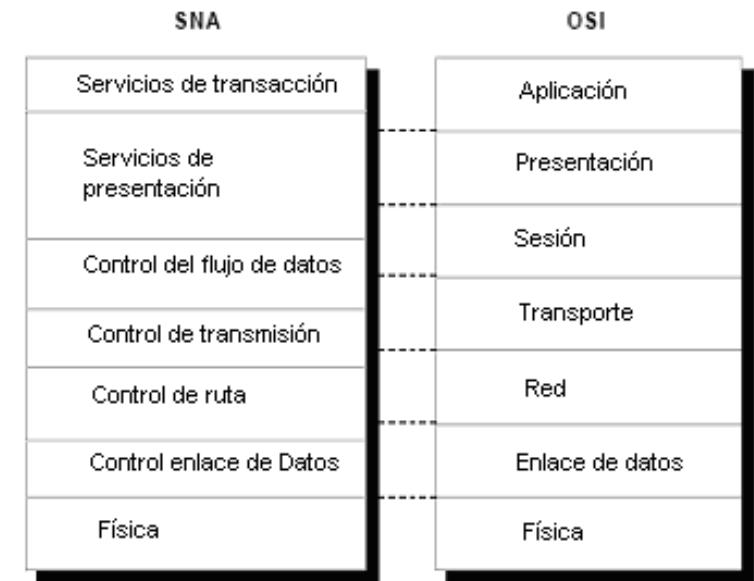
# Paràmetres més importants capes TCP/IP

- ❖ **Nivell d'interfície de xarxa**
  - ❖ **Adreça MAC (Medium Acces Control)**: Identifica una interfície de xarxa dins d'una xarxa de difusió (xarxa local Ethernet)
- ❖ **Nivell d'Internet**
  - ❖ **Adreça IP**: Identifica una màquina a la xarxa Internet
- ❖ **Nivell de transport**
  - ❖ **Port**: Una màquina pot tenir múltiples connexions establertes al mateix temps. Cada connexió esta controlada per un port



# Altres famílies de protocols

- ❖ **Família SNA IBM**
  - ❖ Antecessor i inspiració del model OSI
  - ❖ Apareix al 1974
- ❖ **Família Novell Netware**
  - ❖ Protocol IPX
- ❖ **Família AppleTalk**
  - ❖ Empresa Apple
- ❖ **Família de protocols Windows**
  - ❖ Protocol NetBeui



# Vídeo Warriors of The Net

- <http://www.warriorsofthe.net/>





# TCPDUMP

- ◆ Eina de línia de comandes que permet visualitzar el tràfic de xarxa (Packet Sniffer)
- ◆ Hi ha un “port” per a Windows (WinDump) basat en Wincap (port de libcap)
- ◆ Cal ser superusuari (root) per utilitzar tcpdump (sudo). Activa automàticament el mode promiscu
  - [Tcpdump a la wiki del curs](#)
  - [Pàgina oficial](#)
  - **man tcpdump**

**Desenvolupador:** The Tcpdump team  
**OS:** gairebé tots  
**Llicència:** lliure (BSD)



# TCPDUMP

## ◆ Característiques:

- ◆ Disposa de filtres.
- ◆ És necessari tenir privilegis de superusuari (root) per utilitzar tcpdump.
- ◆ Ethernet és un medi compartit. Si es volen capturar tots els paquets de la xarxa encara que no estiguin destinats al nostre host hem d'activar el mode promiscu.
- ◆ Com gairebé el 100% d'analitzadors de xarxa utilitza la llibreria libcap



# TCPDUMP

## ◆ Filtrant i expressions:

```
$ sudo tcpdump -i eth0
```

```
$ sudo tcpdump -n -i eth0
```

```
$ sudo tcpdump -n -i eth0 > captura_eth0
```

```
$ sudo tcpdump -X -i eth1 dst www.iescopernic.com
```

```
$ sudo tcpdump -X -i eth1 dst 192.168.1.1 and src 192.168.1.2
```

## ◆ Recursos:

- ◆ Depurant amb tcpdump



# TCPDUMP

- **Instal·lació**

```
$ sudo apt-get install tcpdump
```

- **Filtres**

- Podem aplicar filters segons l'origen o destinació del paquet, segons els protocol, per màquines, per xarxes, per ports...

```
$ sudo tcpdump tcp and \ (port 22 or port 23\)
```

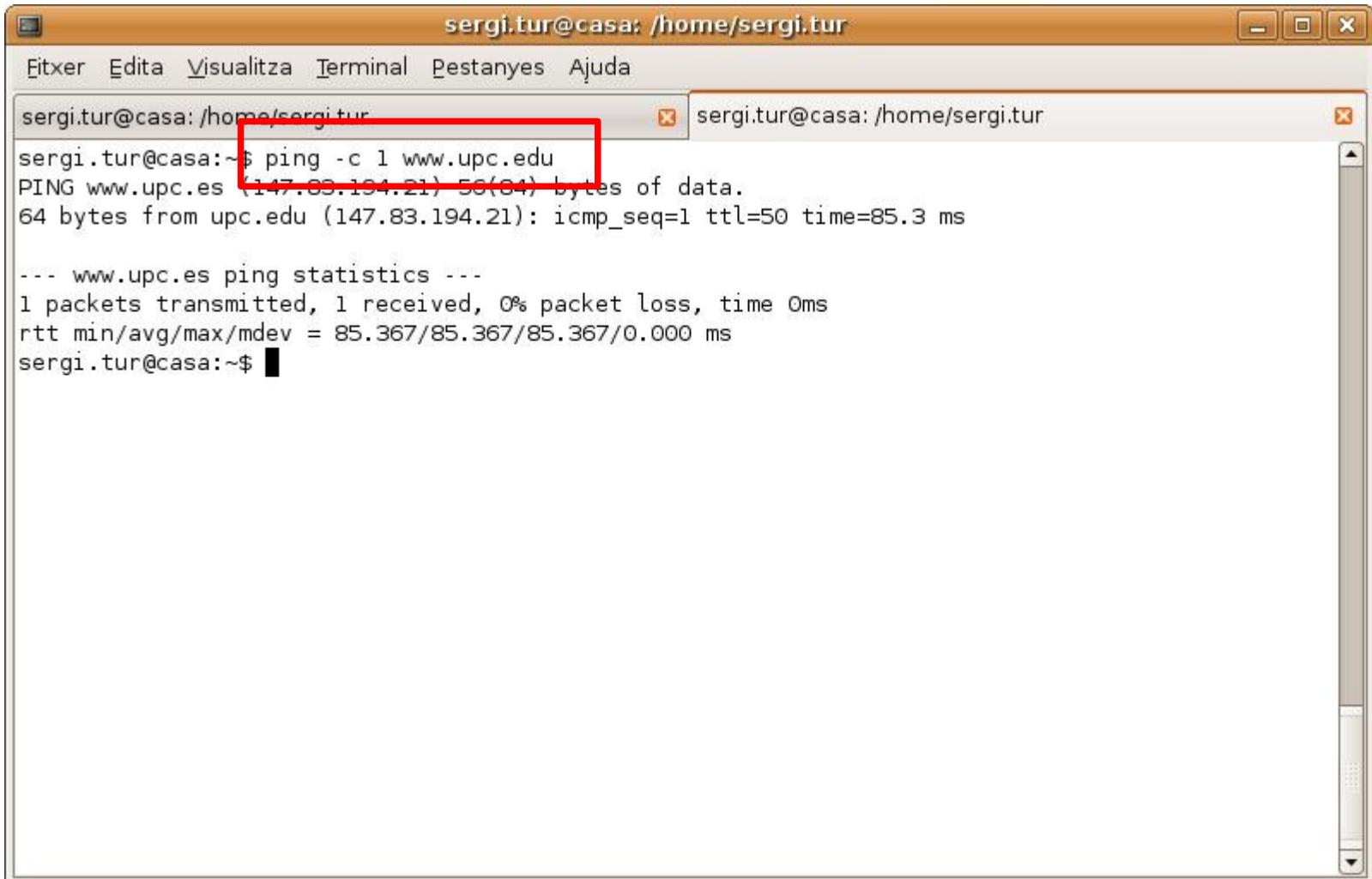
```
$ sudo tcpdump -i lo
```

```
$ sudo tcpdump icmp
```

- Activitat per parelles: Provem de **capturar pings**

# TCPDUMP

## Exemple. Captura d'un ping



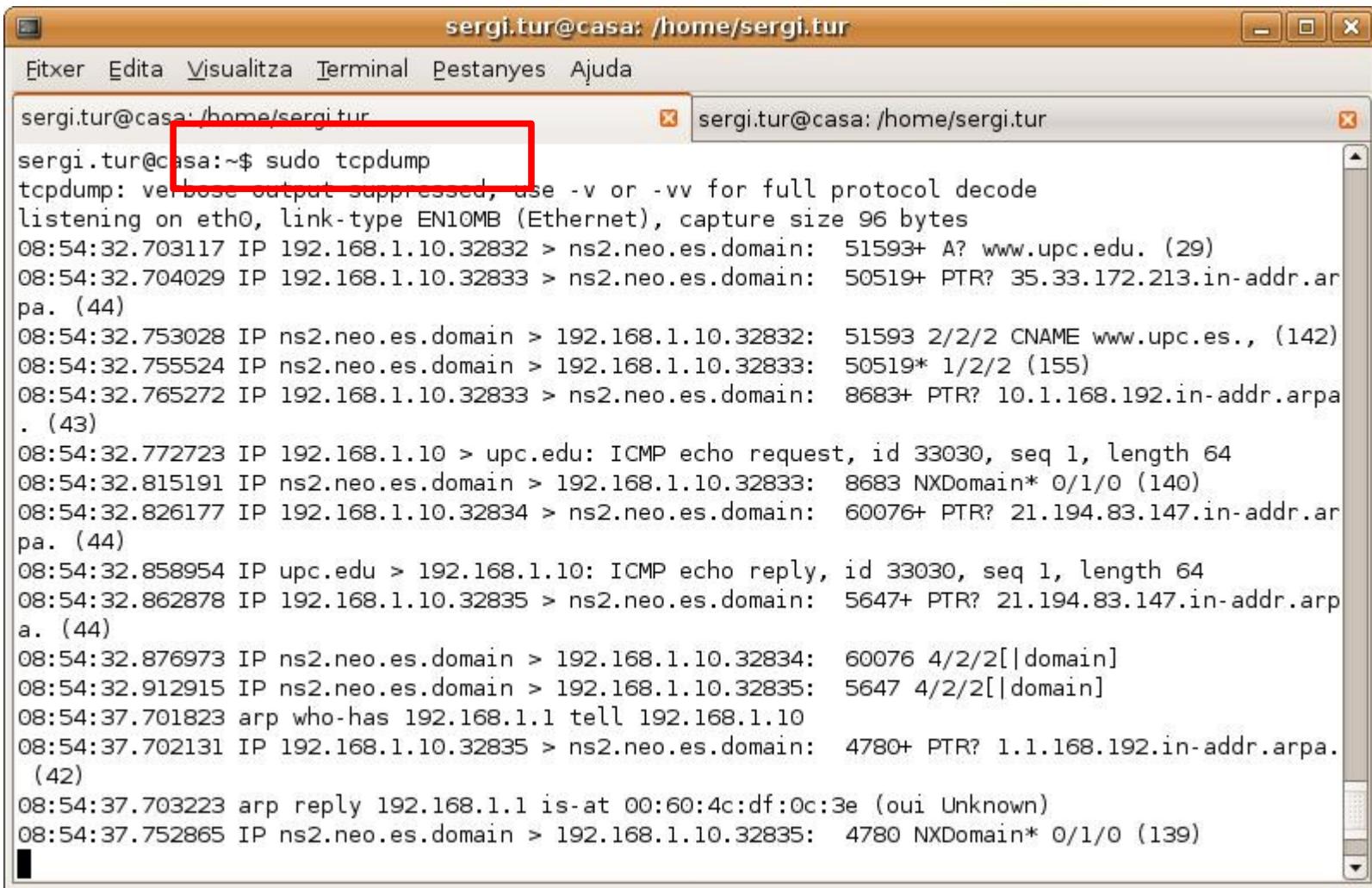
The screenshot shows a terminal window titled "sergi.tur@casa: /home-sergi.tur". The window has two tabs, both labeled "sergi.tur@casa: /home-sergi.tur". The left tab contains the command "sergi.tur@casa:~\$ ping -c 1 www.upc.edu" and its output: "PING www.upc.es (147.83.194.21) 56(84) bytes of data.", "64 bytes from upc.edu (147.83.194.21): icmp\_seq=1 ttl=50 time=85.3 ms", and "--- www.upc.es ping statistics ---". The right tab is empty. A red box highlights the command and its output in the left tab.

```
sergi.tur@casa: /home-sergi.tur
sergi.tur@casa:~$ ping -c 1 www.upc.edu
PING www.upc.es (147.83.194.21) 56(84) bytes of data.
64 bytes from upc.edu (147.83.194.21): icmp_seq=1 ttl=50 time=85.3 ms

--- www.upc.es ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 85.367/85.367/85.367/0.000 ms
sergi.tur@casa:~$
```

# TCPDUMP

## Exemple. Captura d'un ping



```
sergi.tur@casa: /home/sergi.tur
Fitxer Edita Visualitza Terminal Pestanyes Ajuda
sergi.tur@casa: /home/sergi.tur
sergi.tur@casa:~$ sudo tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
08:54:32.703117 IP 192.168.1.10.32832 > ns2.neo.es.domain: 51593+ A? www.upc.edu. (29)
08:54:32.704029 IP 192.168.1.10.32833 > ns2.neo.es.domain: 50519+ PTR? 35.33.172.213.in-addr.arpa. (44)
08:54:32.753028 IP ns2.neo.es.domain > 192.168.1.10.32832: 51593 2/2/2 CNAME www.upc.es., (142)
08:54:32.755524 IP ns2.neo.es.domain > 192.168.1.10.32833: 50519* 1/2/2 (155)
08:54:32.765272 IP 192.168.1.10.32833 > ns2.neo.es.domain: 8683+ PTR? 10.1.168.192.in-addr.arpa. (43)
08:54:32.772723 IP 192.168.1.10 > upc.edu: ICMP echo request, id 33030, seq 1, length 64
08:54:32.815191 IP ns2.neo.es.domain > 192.168.1.10.32833: 8683 NXDomain* 0/1/0 (140)
08:54:32.826177 IP 192.168.1.10.32834 > ns2.neo.es.domain: 60076+ PTR? 21.194.83.147.in-addr.arpa. (44)
08:54:32.858954 IP upc.edu > 192.168.1.10: ICMP echo reply, id 33030, seq 1, length 64
08:54:32.862878 IP 192.168.1.10.32835 > ns2.neo.es.domain: 5647+ PTR? 21.194.83.147.in-addr.arpa. (44)
08:54:32.876973 IP ns2.neo.es.domain > 192.168.1.10.32834: 60076 4/2/2[|domain]
08:54:32.912915 IP ns2.neo.es.domain > 192.168.1.10.32835: 5647 4/2/2[|domain]
08:54:37.701823 arp who-has 192.168.1.1 tell 192.168.1.10
08:54:37.702131 IP 192.168.1.10.32835 > ns2.neo.es.domain: 4780+ PTR? 1.1.168.192.in-addr.arpa. (42)
08:54:37.703223 arp reply 192.168.1.1 is-at 00:60:4c:df:0c:3e (oui Unknown)
08:54:37.752865 IP ns2.neo.es.domain > 192.168.1.10.32835: 4780 NXDomain* 0/1/0 (139)
```



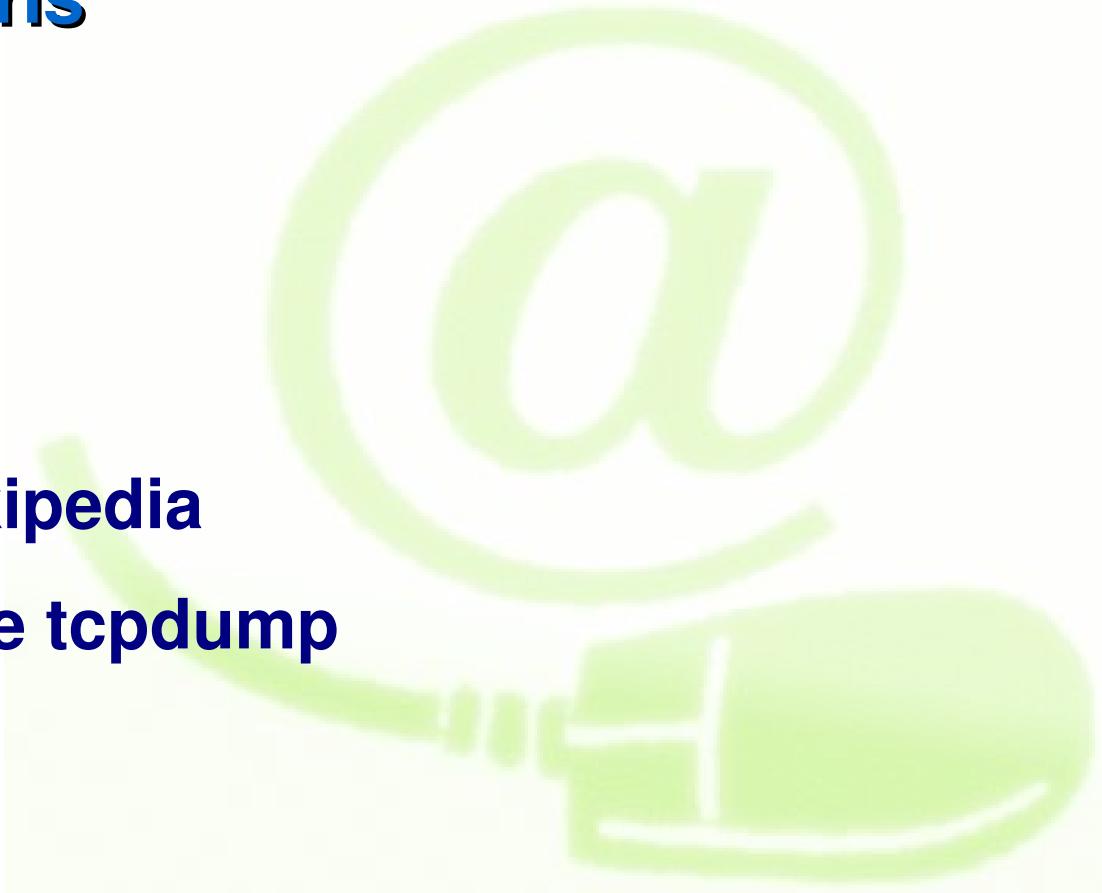
# TCPDUMP

## ◆ Utilitats:

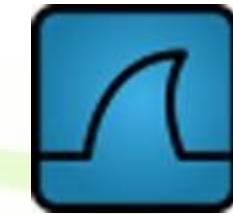
- ◆ Per depurar aplicacions que utilitzen la xarxa per comunicar-se. Per exemple es pot utilitzar per comprovar el funcionament d'un tallafocs.
- ◆ Per depurar la xarxa mateixa.
- ◆ Per comprovar quan la NIC està transmetent o reben dades.
- ◆ Per capturar i llegir dades enviades per altres usuaris o ordinadors. Un usuari que té el control d'un enrutador pel qual circula tràfic pot obtenir la informació que no viatgi xifrada.

# TCPDUMP

- ❖ **Paquets necessaris**
  - ❖ tcpdump
- ❖ **Referències**
  - ❖ man tcpdump
  - ❖ Article de la wikipedia
  - ❖ Pàgina oficial de tcpdump
- ❖ **Altres enllaços**
  - ❖ WinDump
  - ❖ Article de la wikipedia sobre Paquet Sniffers



# Ethereal (WireShark)

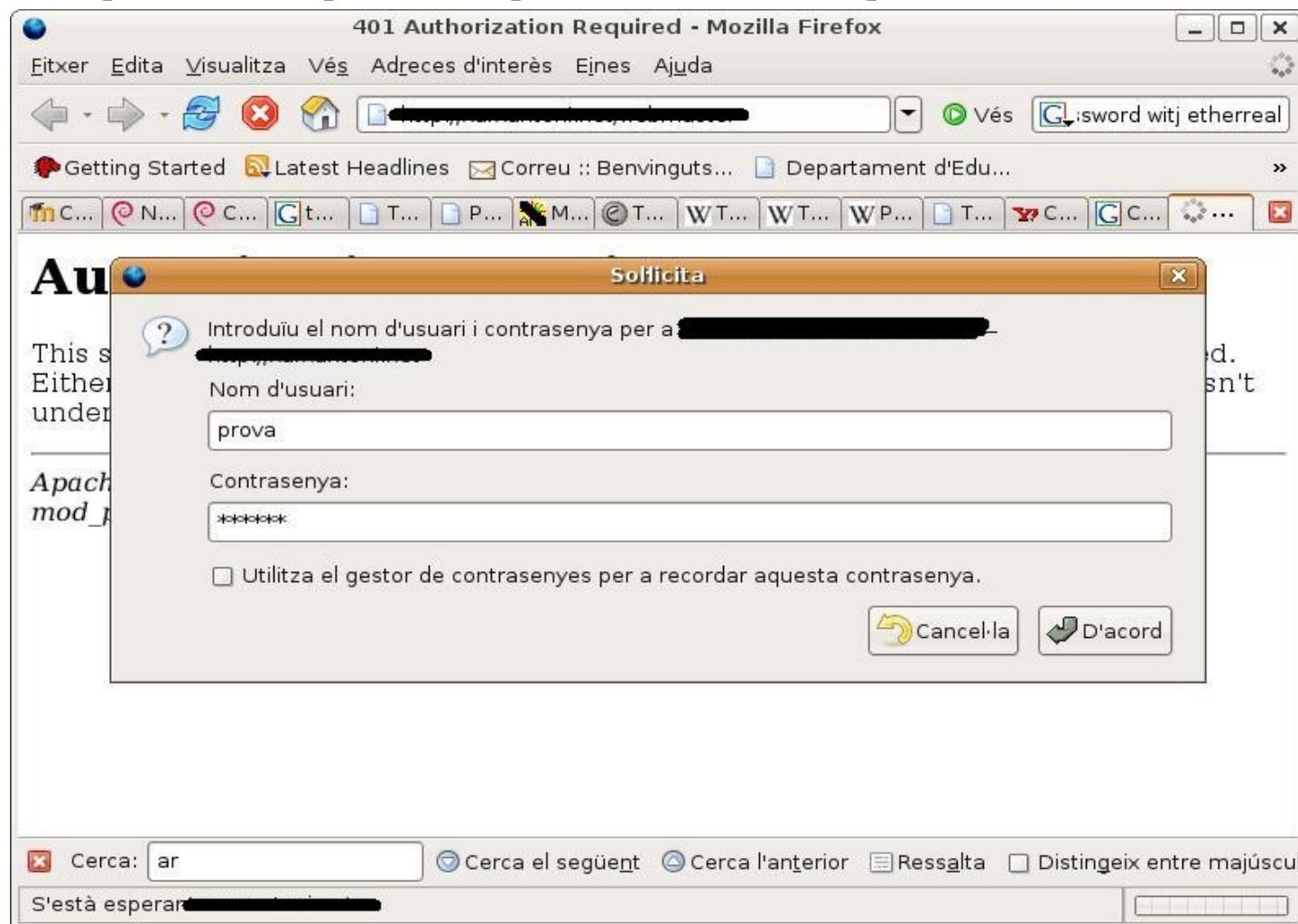


## • Característiques:

- ◆ Ethereal és un analitzador de protocols utilitzat per analitzar i solucionar problemes de xarxes de comunicacions.
- ◆ És similar a tcpdump però amb una interfície gràfica i moltes opcions extres d'organització i filtratge de la informació.
- ◆ Com tcpdump és codi obert està disponible per gairebé totes les plataformes (UNIX/LINUX, MAC OS i Windows).

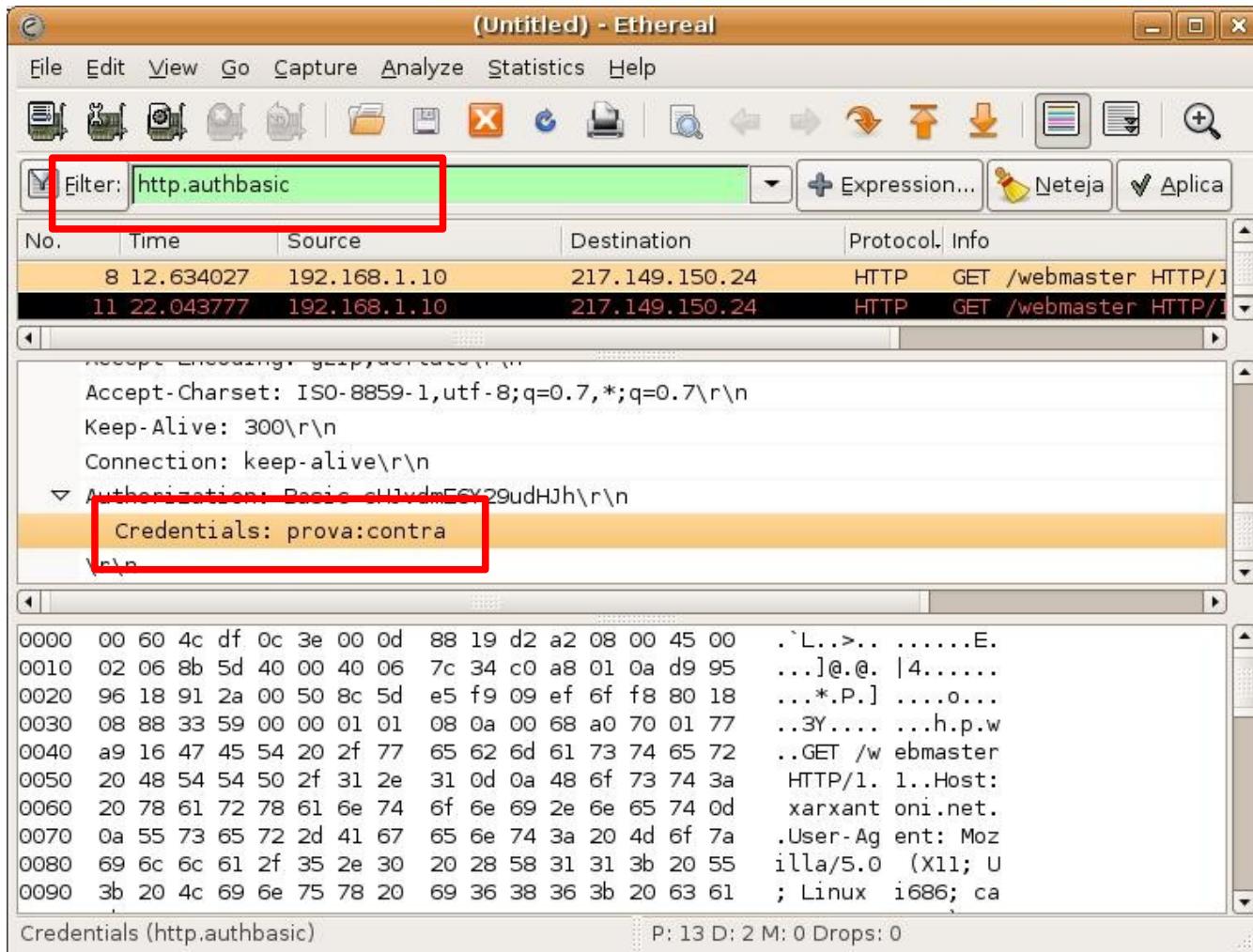
# Ethereal. Captura contrasenyes HTTP

## Exemple. Captura paraula de pas web.



# Ethereal

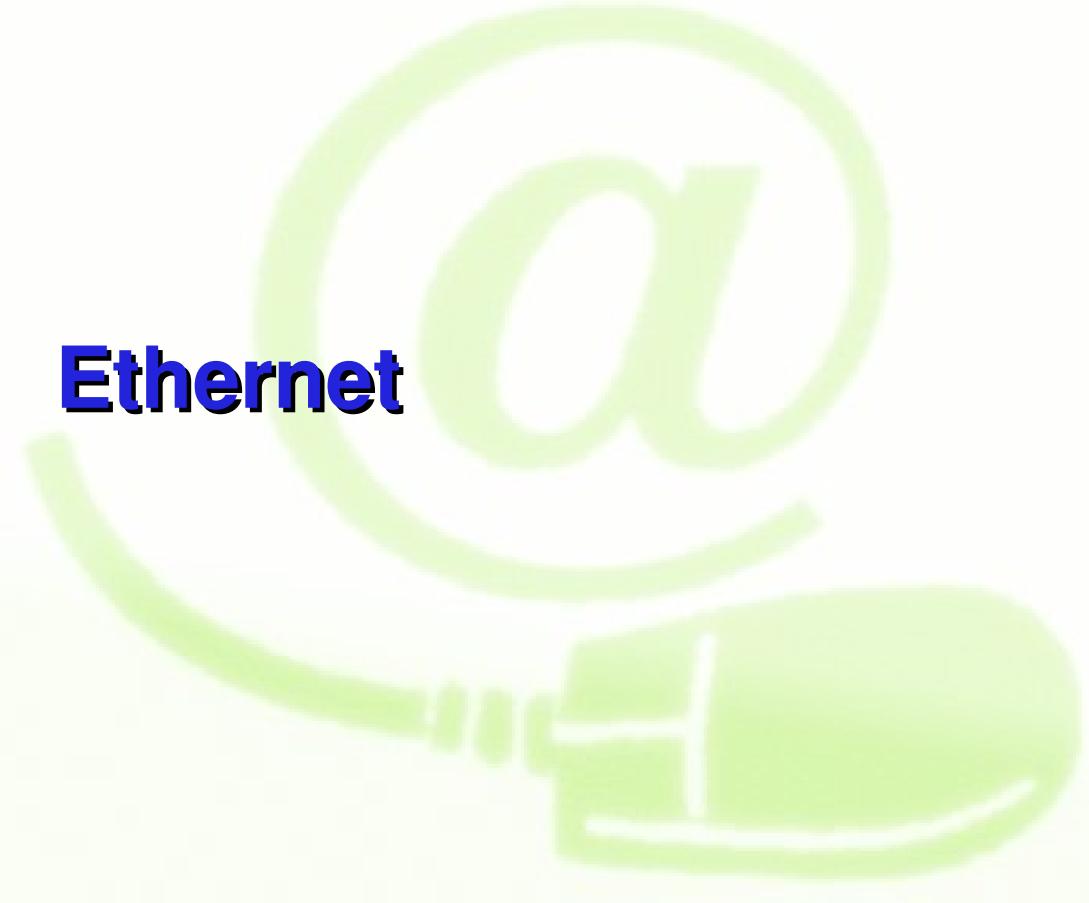
## Exemple. Captura paraula de pas web.





# Arquitectures de comunicacions

Ethernet



# Ethernet

- **Nivell 1 TCP/IP (Nivells físics i d'enllaç OSI).**
- **Família d'estàndards IEEE 802:**
  - ◆ 802.2: **Capa LLC** (Logical Link Control). Interfície comuna entre el nivell de xarxa i la família de protocols.
  - ◆ La resta de protocols defineixen el nivell físic i el subnivell **MAC**.
    - **802.3 Ethernet**
    - 802.4 Token Ring
    - **802.11 Wi-Fi**
    - 802.15 Bluetooth



# Ethernet

- ❖ **Nivell LLC (Logical Link control). Compartit per tots els protocols de la família.**
  - ❖ Lògica de reenviaments
  - ❖ Control de flux
  - ❖ Comprovació d'errors
- ❖ **Nivell MAC (Medium Acces Control).**
  - ❖ Control d'accés a medi compartits (cables en bus, ràdio, etc.)
  - ❖ No utilitzat en protocols punt a punt com ppp o xarxes commutades (no hi ha medi compartit)
  - ❖ **Adreça MAC:** Sistema adreçament de nivell 2 **equivalent a les adreces IP** al nivell 3

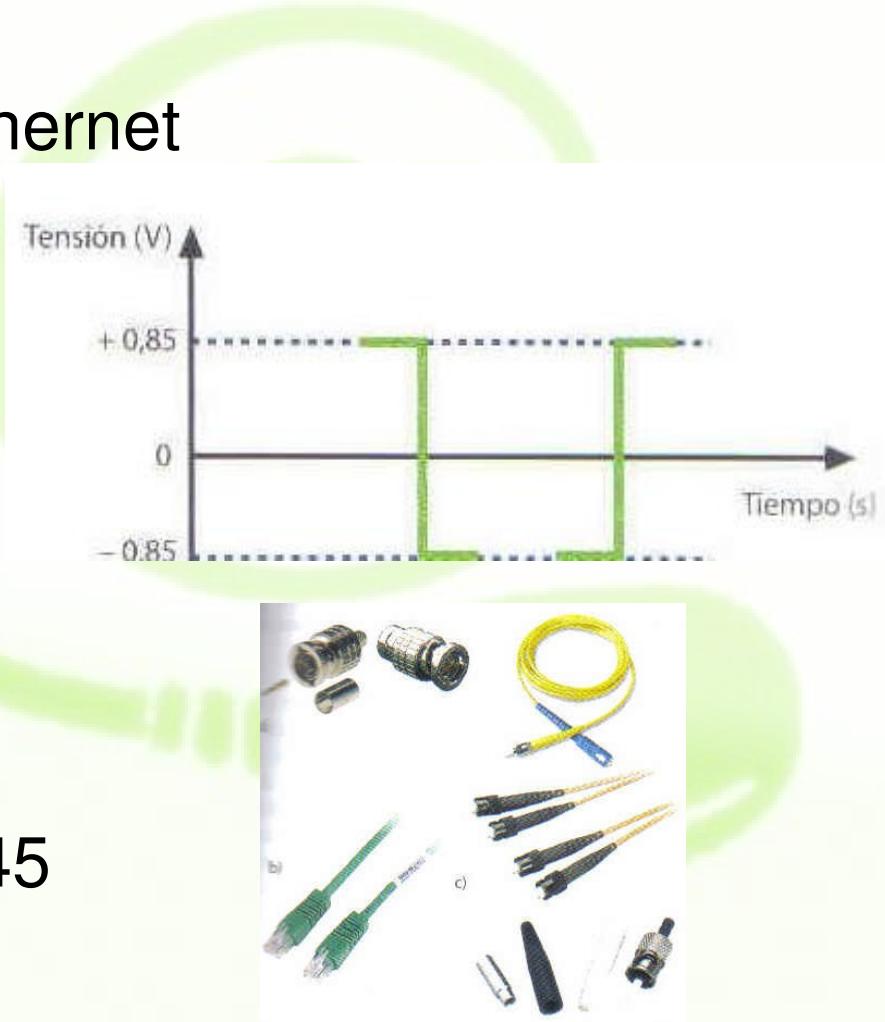
# Ethernet

## ◆ Protocols MAC

- ◆ CSMA/CD: Utilitzat per Ethernet
- ◆ Aloha i Aloha ranurat
- ◆ Token Ring|Token Bus

## ◆ Nivell físic Ethernet

- ◆ Codificació manchester
- ◆ Connectors coaxials i RJ-45



# Ethernet. Nivell MAC

## • **Algorismes MAC**

- ◆ **Aloha i Aloha Ranurat** (desenvolupats per la Universitat de Hawai). S'envia un paquet i si hi ha col·lisió es torna a enviar.
- ◆ **CSMA/CD (*Carrier sense multiple access with collision detection*)**. Detecta si hi ha senyals utilitzant el medi i té un procediment en cas de col·lisió.
- ◆ Antics sistemes Ethernet funcionaven amb coaxials en bus físic i lògic.
- ◆ Actualment el problema de les col·lisions està més limitat gràcies als switches.
- ◆ Torna a ser un tema candent en xarxes wireless (l'aire és un medi compartit).



# La targeta de xarxa. Network Interface Card

La targeta de xarxa és el pont d'enllaç entre el sistema operatiu i el accés al medi de transmissió (ja sigui aquest un cable o un sistema sense fils)

- ◆ També anomenada **NIC** (Network Interface Card) o **adaptador de xarxa**.
- ◆ Dispositiu que treballa als nivells baixos d'OSI (capa 1 física i capa 2 d'enllaç)
- ◆ Cada interfície de xarxa té una adreça MAC única
  - La MAC permet adreçar i identificar de forma unívoca les targetes de xarxa
  - La MAC és un identificador de 48 bits amb dos parts
    - **Id del venedor:** 00:0D:88
    - **Id de la targeta de xarxa:** 19:D2:A2
  - Cal tenir en compte que hi han targetes de xarxa amb més d'una interfície de xarxa (i per tant amb més d'una MAC)

00:0D:88:19:D2:A2

# La targeta de xarxa. Network Interface Card

```
$ /sbin/ifconfig
```

**Primera NIC**

```
eth0      Link encap:Ethernet HWaddr 00:0D:88:19:D2:42  
          inet addr: 192.168.1.10  Bcast: 192.168.1.255  Mask: 255.0.0.0  
          inet6 addr: fe80::20d:88ff:fe19:d2a2/64 Scope:Link  
             UP BROADCAST RUNNING MULTICAST  MTU:1500 Metric:1  
             RX packets:957270 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:1254234 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:1000  
             RX bytes:581767799 (554.8 MiB)  TX bytes:228519990 (217.9 MiB)  
             Interrupt:11 Base address:0x3000  Interfície de loopback
```

```
lo        Link encap:Local Loopback  
          inet addr:127.0.0.1  Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
             UP LOOPBACK RUNNING  MTU:16436 Metric:1  
             RX packets:799643 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:799643 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:164196675 (156.5 MiB)  TX bytes:164196675 (156.5 MiB)
```



# La targeta de xarxa. Network Interface Card

- **La ranura d'expansió (slot en anglès) és el connector físic a on es connecta la targeta**
- **Tipus de ranures de connexió**
  - ◆ ISA (Industry Standard Architecture)
    - Creat al 1981. Ja no s'utilitza en sistemes nous
  - ◆ PCI (Peripheral Component Interconnect)
    - El més utilitzat actualment
  - ◆ PCI-E: Nou bus PCI
  - ◆ PCMCIA: Utilitzat en ordinadors portàtils
  - ◆ USB: Només recomanat si no hi ha un altre opció
  - ◆ Actualment moltes targetes estan integrades a la placa mare (però continuen utilitzant un BUS PCI)



Targeta ISA (1990) amb connectors coaxial i RJ-45

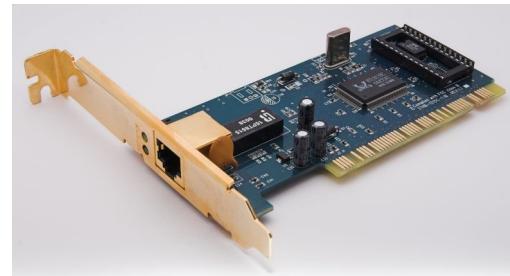


PCMCIA

# La targeta de xarxa. Network Interface Card

## ♦ Fabricants

- ◆ Novell
- ◆ Intel
- ◆ Realtek



PCI NIC



USB NIC



WIRELESS PCI NIC

## ♦ BOOT-ROM

- ◆ La ROM de BOOT porta el programa per arrencar des de la targeta de xarxa un sistema
- ◆ Moltes targetes o plaques mare la porten integrada a la BIOS
- ◆ Algunes targetes de xarxa permeten incorporar una targeta **CompactFlash**



# La targeta de xarxa. Network Interface Card

## ◆ Paràmetres habituals de la targeta de xarxa

- ◆ **IRQ (Interrupt Request)**: Sol·licitud d'interrupció. Número de la línia d'interrupció que utilitza la NIC per avisar a la CPU que han arribat dades.
- ◆ **Adreça d'E/S**: Espai de memòria que utilitzen la CPU i la targeta de xarxa per comunicar-se
- ◆ **DMA (Direct Memory Access)**: S'utilitza poc en targetes modernes
- ◆ La configuració dels paràmetres actualment es fa per programari
- ◆ Abans algunes configuracions es feien utilitzant jumpers

## ◆ Identificació de la targeta de xarxa

- ◆ **Venedor: pci.vendor\_id (0x11ab)**
- ◆ **Producte: pci.product\_id (0x4320)**



# La targeta de xarxa. Network Interface Card

```
$ ifconfig
eth0      Link encap:Ethernet HWaddr 00:0D:88:19:D2:A2
          inet addr:192.168.1.10 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: fe80::20d:88ff:fe19:d2a2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:957270 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1254234 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:581767799 (554.8 MiB) TX bytes:228519990 (217.9 MiB)
IRQ → Interrupt:11 Base address:0x4000 ← Memòria d'entrada i sortida
```

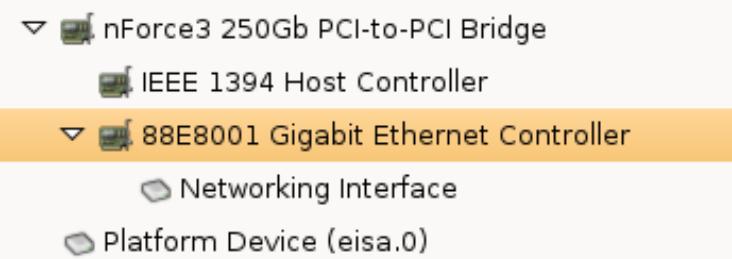
```
$ watch ifconfig
eth0      Link encap:Ethernet HWaddr 00:0D:88:19:D2:A2
          inet addr:192.168.1.10 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: fe80::20d:88ff:fe19:d2a2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:957270 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1254234 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:581767799 (554.8 MiB) TX bytes:228519990 (217.9 MiB)
          Interrupt:11 Base address:0x4000
```



# La targeta de xarxa. Network Interface Card

## Ubuntu

### Sistema/Prefèries/Informació de maquinari



Dispositiu	PCI	Avançat
<b>Venedor:</b>	Marvell Technology Group Ltd.	
<b>Dispositiu:</b>	88E8001 Gigabit Ethernet Controller	
<b>Estat:</b>	Estat	
<b>Tipus de bus:</b>	PCI	
<b>Tipus de dispositiu:</b>	Unknown	
<b>Capacitats:</b>	Unknown	

```
$ lspci | grep Ethernet
02:08.0 Ethernet controller: Marvell Technology Group Ltd. 88E8001 Gigabit
Ethernet Controller (rev 13)
```

- Busqueu els paràmetres comentats anteriorment utilitzant la comanda:

```
$ sudo lspci -vvvxxxx
```



# La targeta de xarxa. Network Interface Card

```
$ lspci -vvvxxxx
```

```
02:08.0 Ethernet controller: Marvell Technology Group Ltd. 88E8001 Gigabit
Ethernet Controller (rev 13)
```

```
Subsystem: Holco Enterprise Co, Ltd/Shuttle Computer Unknown device c231
```

```
Control: I/O+ Mem+ BusMaster+ SpecCycle- MemWINV- VGASnoop- ParErr-
Stepping- SERR- FastB2B-
```

```
Status: Cap+ 66MHz+ UDF- FastB2B+ ParErr- DEVSEL=medium >TAbsrt- <TAbsrt-
<MAbsrt- >SERR- <PERR-
```

```
Latency: 32 (5750ns min, 7750ns max), Cache Line Size: 32 bytes
```

```
Interrupt: pin A routed to IRQ 20
```

```
Region 0: Memory at e5020000 (32-bit, non-prefetchable) [size=16K]
```

```
Region 1: I/O ports at d100 [size=256]
```

```
[virtual] Expansion ROM at 50000000 [disabled] [size=128K]
```

```
Capabilities: [48] Power Management version 2
```

```
Flags: PMEClk- DSI- D1+ D2+ AuxCurrent=0mA PME(D0+,D1+,D2+,D3hot
+,D3cold+)
```

```
Status: D0 PME-Enable- DSel=0 DScale=1 PME-
```

```
Capabilities: [50] Vital Product Data
```

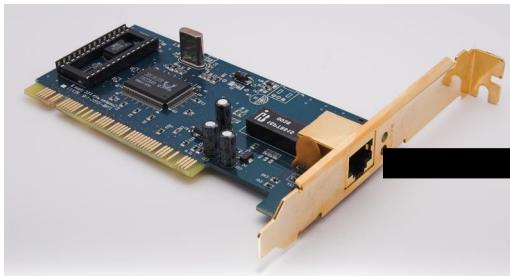


# La targeta de xarxa. Network Interface Card

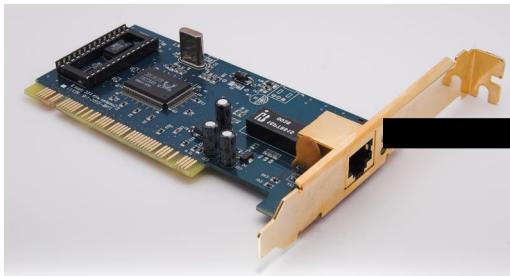
- ◆ Les targetes de xarxa modernes suporten diferents modes i velocitats
- ◆ **Auto-negociació:** permet a targetes amb diferents velocitats i modes poder establir una comunicació
- ◆ Algunes targetes fins i tot detecten quan un cable esta creuat i permeten treballar amb aquests tipus de cables
- ◆ **Link:** Índica que la targeta de xarxa esta connectada a un altre dispositiu (switch o targeta de xarxa)

```
$ sudo ethtool eth0
Settings for eth0:
  Supported ports: [ TP (Twisted Pair) ]
  Supported link modes: 10baseT/Half 10baseT/Full
                         100baseT/Half 100baseT/Full
                         1000baseT/Half
  1000baseT/Full
    Supports auto-negotiation: Yes
    Advertised link modes: 10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Half
  1000baseT/Full
    Advertised auto-negotiation: Yes
    Speed: 100Mb/s
    Duplex: Full
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    Auto-negotiation: on
    Supports Wake-on: pg
    Wake-on: g
    Current message level: 0x00000037 (55)
    Link detected: yes
```

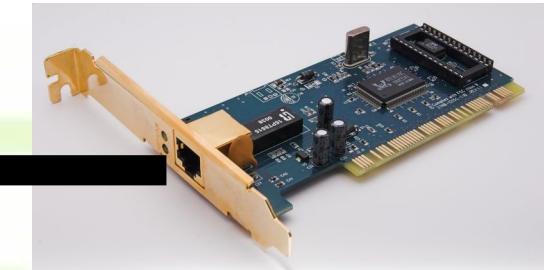
# Targetes de xarxa, connexions i cables



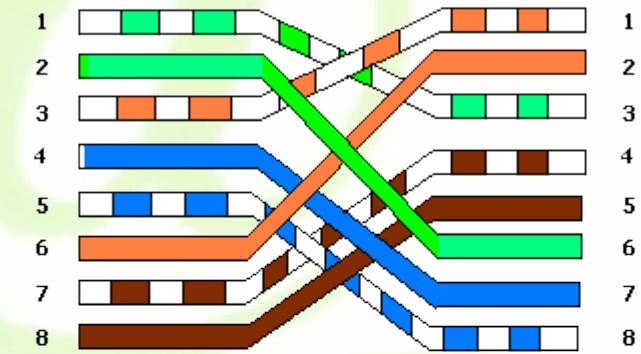
Cable UTP creuat  
Connexió entre dispositius  
iguals



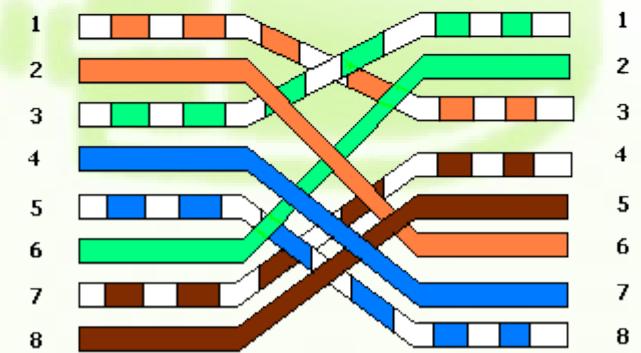
Cable UTP normal  
Connexió entre dispositius  
diferents



TIA/EIA 568A Crossed Wiring



TIA/EIA 568B Crossed Wiring



# Wake On LAN

- ❖ La majoria de targetes permeten encendre un PC enviant una senyal a la targeta de xarxa
  - ❖ Feu una pràctica per parelles
  - ❖ Primer cal utilitzar la comanda **ethtool**, per configurar la màquina que volem que s'encengui des de la targeta de xarxa
  - ❖ Amb les comandes **wakeonlan** o **etherwake** enviem la senyal per encendre la màquina remota
    - Necessitarem saber la MAC de la targeta de xarxa del PC que volem engegar
  - ❖ Wake ON LAN

## ♦ **Segments de xarxa**

- ◆ És una porció de xarxa separada de la resta per un dispositiu de xarxa com:
  - Repetidor
  - Bridge o Switch
  - Router

## ♦ **Domini de col·lisió**

- ◆ És un segment lògic de xarxa on els paquets poden col·lisionar al ser enviats a un medi compartit.



# Trama Ethernet

Preàmbulo	SOF	Destino	Origen	Típo	Datos	FCS
7 bytes	1 byte	6 bytes	6bytes	2 bytes	46 a 1500 bytes	4 bytes

- ◆ **Preàmbul:** alterna zeros i uns, estabilitza el canal físic
- ◆ **SOF (Start Of Frame):** identifica l'inici d'una trama
- ◆ **Origen:** Adreça MAC origen de la trama
- ◆ **Destí:** Adreça MAC destinació de la trama
- ◆ **Tipus:** Identifica el protocol de nivell superior (IP)
- ◆ **Dades:** són les dades reals que s'estan enviant (inclou capçaleres de protocols superiors)
  - Falta el **camp de farciment** al dibuix-> Trama mínima de 64 bytes
- ◆ **FCS (Frame Check Sequence):** Codi CRC per comprovar la trama

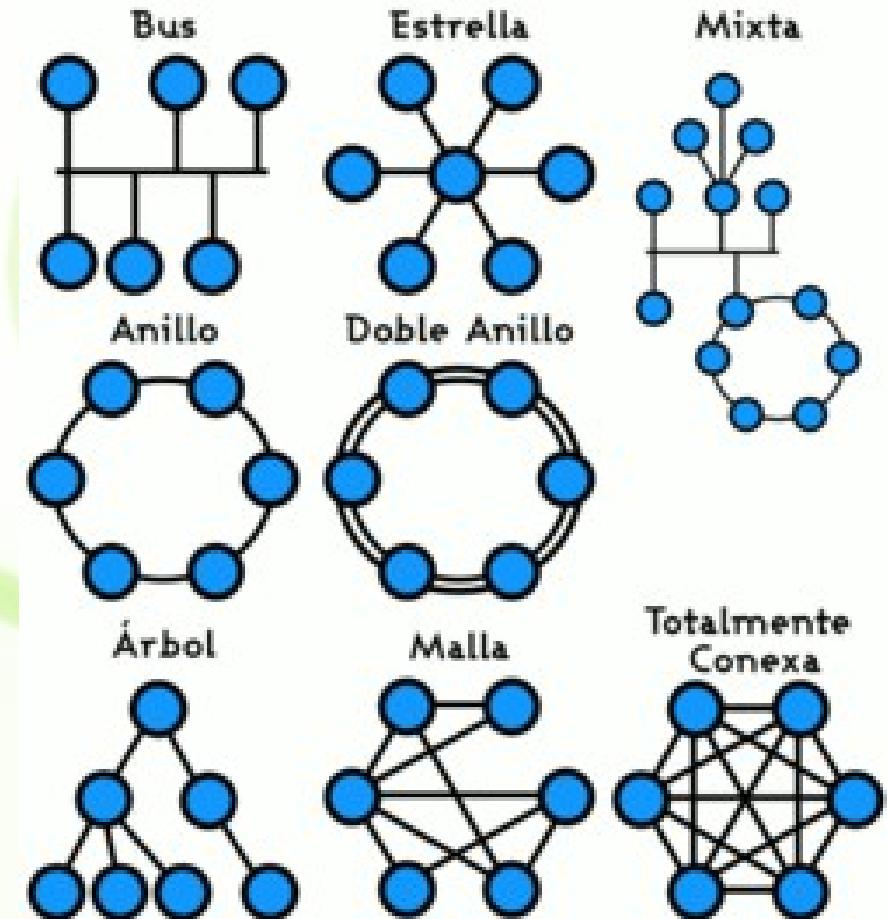
# Switched LAN. Hubs i Switchs

- ❖ **Les LANs connectades a switchs o HUBS tenen una topologia física d'estrella.**
- ❖ **Topologia lògica:**
  - ❖ **HUB:** mateix segment de xarxa (bus compartit). Treballa a nivell físic (mecànic). Dispositiu “tonto” (dumb)
  - ❖ **Switch:** s'utilitza una base de dades per recordar les MACs (Ips) de cada port i es connecta de forma directa els ports d'origen i destinació d'una comunicació. Treballa a nivell d'enllaç (taula de MACS). Dispositiu intel·ligent.
    - LAN Commutada. Cada PC té el seu propi segment de xarxa no compartit.
    - Els switches són més segurs.



# Topologies de xarxa

- ◆ **Classificació de les xarxes segons la seva forma**
  - ◆ Estrella
  - ◆ Bus
  - ◆ Anell
  - ◆ Topologies de xarxa





# Topologia en estrella

- ◆ **Tots els nodes es connecten a un node concentrador**

- ◆ **Exemple:** Ethernet amb RJ-45

- ◆ **Avantatges**

- Fàcil d'implementar i d'ampliar
- Instal·lació ràpida
- Un error en un node o segment de xarxa no influeix a la resta de la xarxa
- No hi ha problemes de col·lisió de dades
- Més seguretat

- ◆ **Inconvenients**

- Major longitud de cable i nombre de nodes limitat pel concentrador
- Un error en el concentrador és un error a tota la xarxa





# Topologia en anell

- ◆ **Cada node esta connectat al següent i l'últim al primer**

- ◆ Cada node té un receptor i un transmissor i fa la funció de repetidor

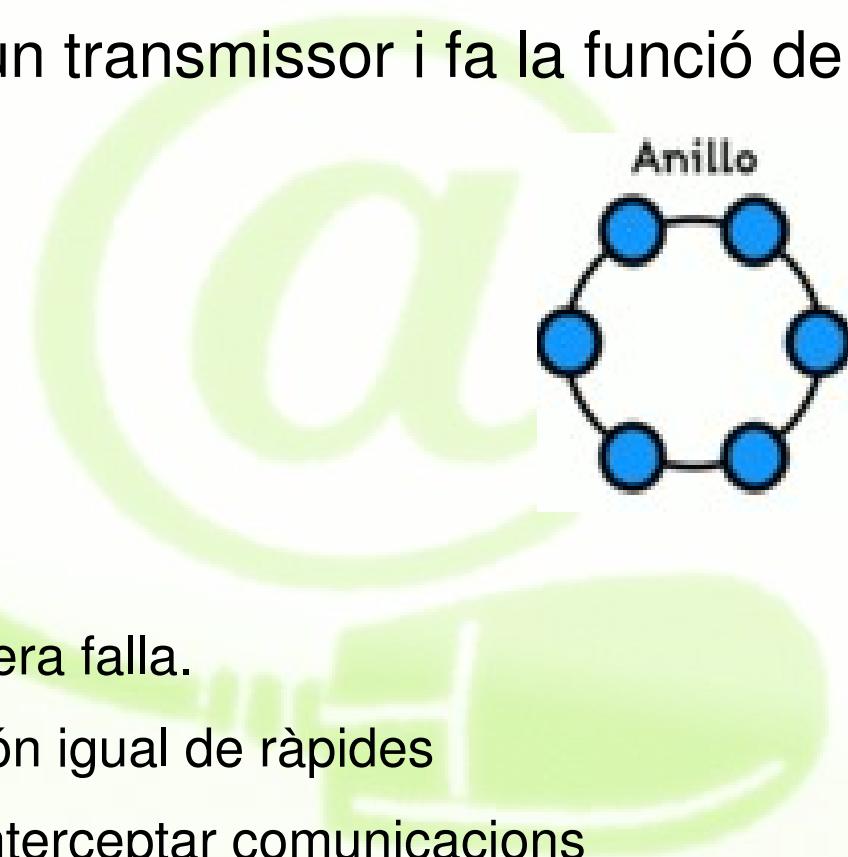
- **Exemple:** Token Ring

- ◆ **Avantatges**

- Poca longitud del cable

- ◆ **Inconvenients**

- Si un node falla la xarxa sencera falla.
- No totes les comunicacions són igual de ràpides
- Seguretat. Els nodes poden interceptar comunicacions





# Topologia en bus

- **Tots els nodes es connecten a la mateixa línia (bus)**

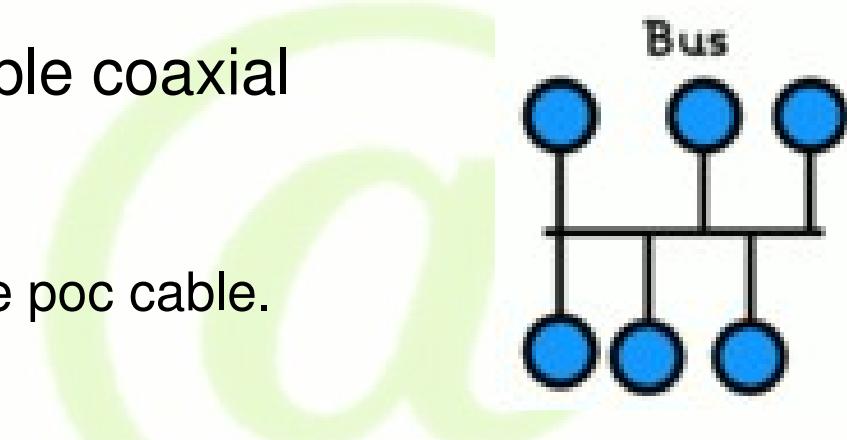
- ◆ **Exemple:** Ethernet amb cable coaxial

- Avantatges**

- Fàcil d'instal·lar i requereix de poc cable.

- Inconvenients**

- El medi de transmissió es compartit. Poden succeir col·lisions i lluites per la utilització del canal de comunicacions
    - Seguretat. Tots els nodes poden accedir a la informació de la resta.





# Altres topologies

## ♦ **Malla**

- ◆ Entre els nodes s'estableixen enllaços punt a punt.
  - **Exemple:** Xarxes WAN

## ♦ **Interconnexió total**

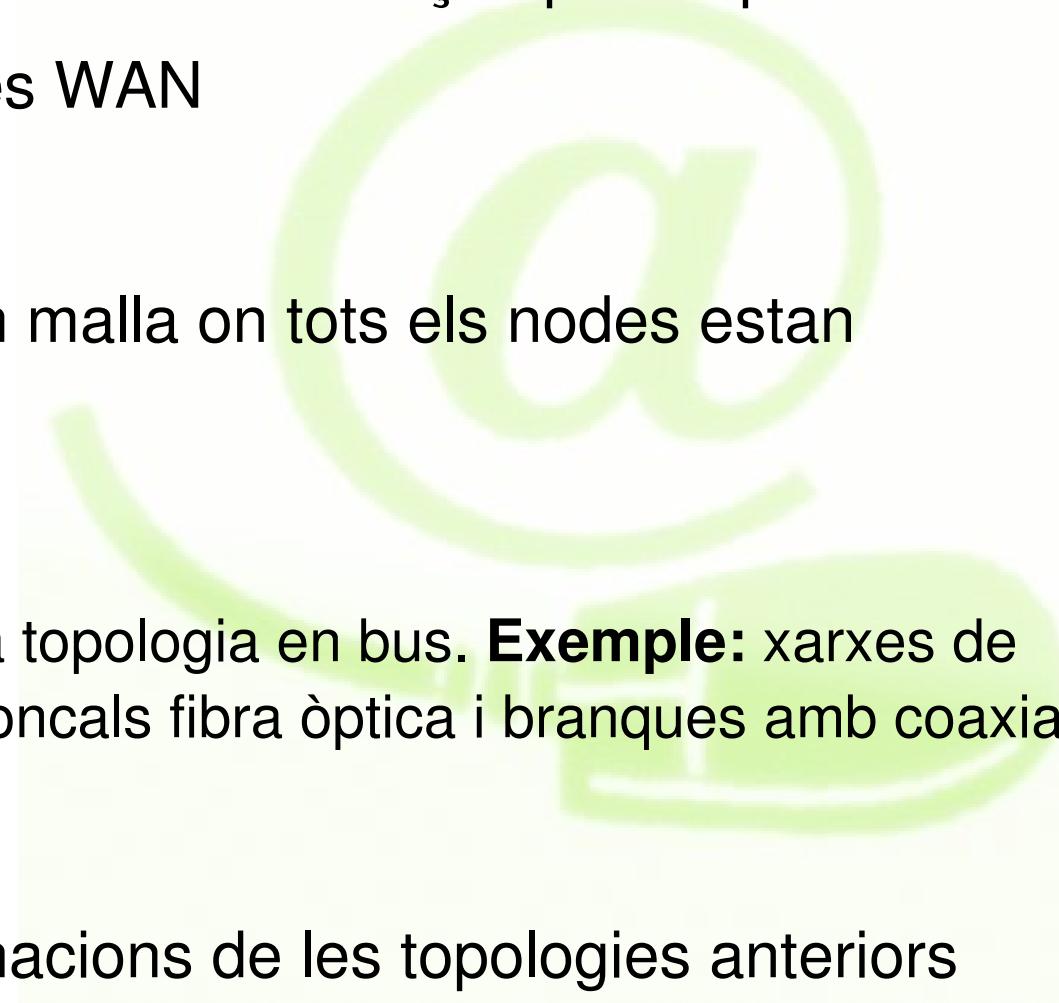
- ◆ És una topologia en malla on tots els nodes estan connectats entre si.

## ♦ **Arbre**

- ◆ És una extensió de la topologia en bus. **Exemple:** xarxes de televisió per cable, troncals fibra òptica i branques amb coaxial

## ♦ **Mixta**

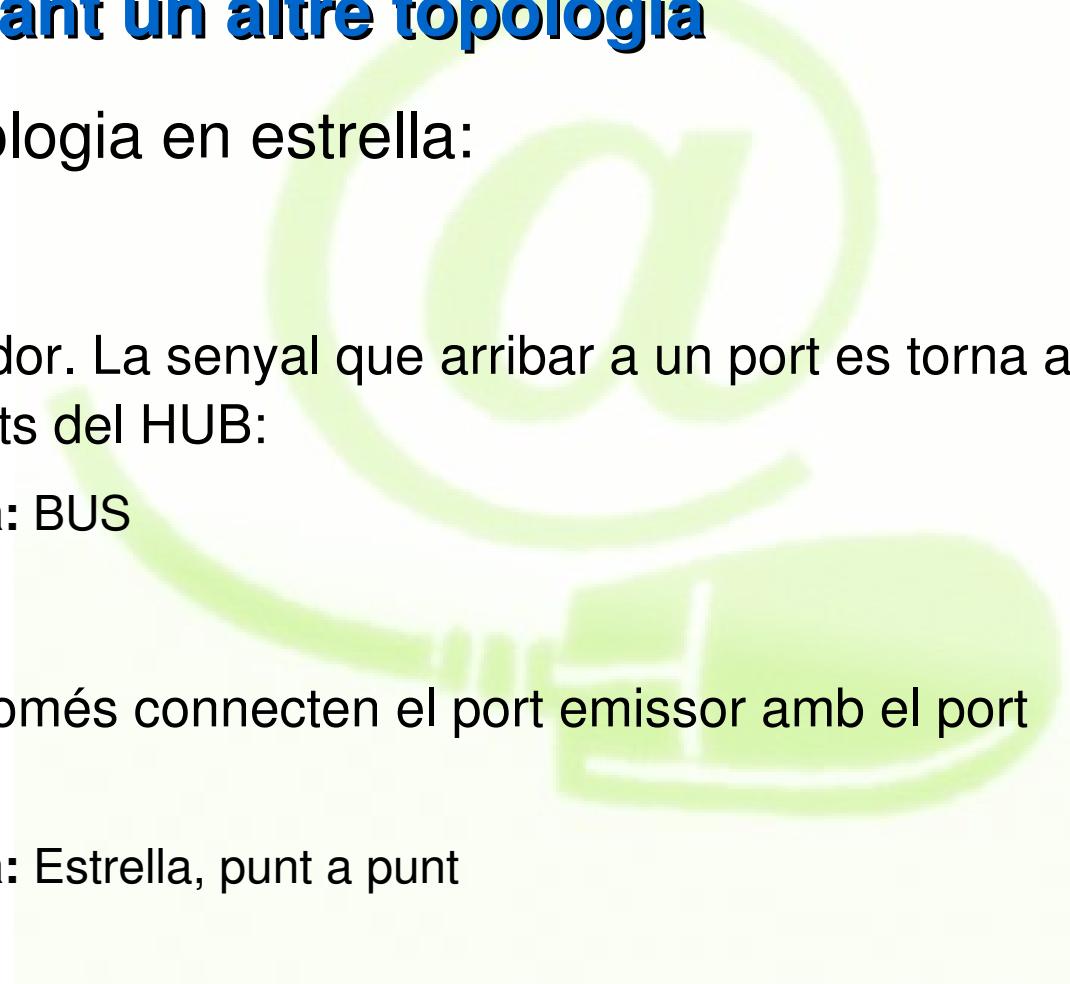
- ◆ Xarxes amb combinacions de les topologies anteriors





# Topologia física vs Topologia lògica

- **Físicament podem tenir una topologia concreta però realment esta utilitzant un altre topologia**
  - ◆ Per exemple la topologia en estrella:
  - ◆ **HUB**
    - És un simple repetidor. La senyal que arribar a un port es torna a enviar a tots els ports del HUB:
      - **Topologia lògica: BUS**
  - ◆ **SWITCH**
    - Els commutadors només connecten el port emissor amb el port receptor
      - **Topologia lògica: Estrella, punt a punt**



# Switches

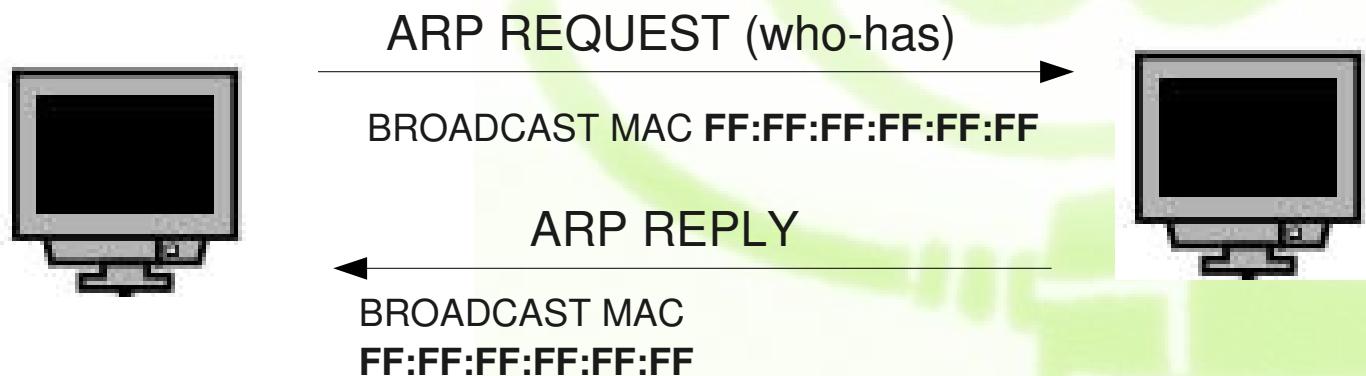
## Tipus:

- ◆ **Home Switches** (no gestionats). Típics en entorns SoHo (Small Office/ Home Office).
- ◆ **Switches gestionats** (Managed Switches).
  - Controlar el port de forma individual (on/off)
  - Control de la velocitat del link
  - Prioritats de ports
  - Filtratge MAC
  - Port Mirroring per tal de monitoritzar ports.
  - Altres: Suport per SNMP, VLAN, Link Aggregation.
- ◆ **Switches intel·ligents**. Realitzen tasques de forma automàtica: establir velocitats d'enllaç, permetre connexions directes i creuades, etc.

# Protocol ARP

- ARP és un protocol a cavall entre el nivell de xarxa i el nivell d'enllaç (MAC)

- Permet resoldre adreces MAC a partir d'adreces IP.
- S'utilitza en xarxes LAN (nivell 2) per poder treballar amb adreces IP (nivell 3)
- Es transportat per trames Ethernet (type=ARP)



```
$ sudo tcpdump -i eth0
```

```
17:51:38.740533 arp who-has 192.168.1.2 tell mygateway1.ar7
17:51:38.740550 arp reply 192.168.1.2 is-at 00:30:1b:b7:cd:b6 (oui Unknown)
```



# Protocol arp

## • Taula arp (cache arp)

```
$ arp -n
Address      HWtype  HWaddress          Flags Mask Iface
192.168.1.1  ether   00:15:E9:CA:34:A5 C      eth0
```

- ◆ Eliminar una entrada: `$ sudo arp -d 192.168.1.1`
- ◆ Afegir una entrada estàtica:

```
$ sudo arp -s 87.111.152.1 00:14:1c:32:af:1a
```

- ◆ Ordre ip

```
$ sudo ip neigh show
192.168.111.100 dev eth0 lladdr 00:15:6d:ae:70:ee REACHABLE
192.168.111.1 dev eth0 lladdr 00:25:69:3f:d5:ae REACHABLE
192.168.111.10 dev eth0 lladdr 00:19:5b:ef:8e:9c STALE
```



# Protocol ARP

## Exercici

- Consultem la taula ARP

```
$ arp
Address      HWtype  HWaddress          Flags Mask   Iface
mygateway1.ar7  ether   00:15:E9:CA:34:A5  C      eth0
```

- Executem alguna comanda que obligui a fer un broadcast de la xarxa (utilitzar totes les IPs)

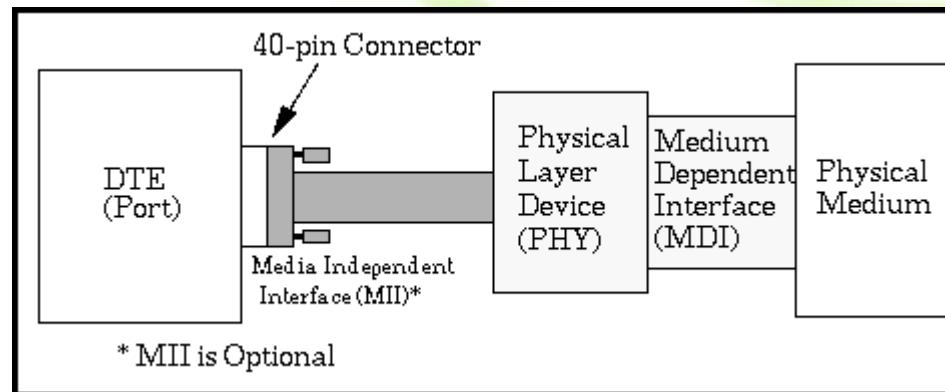
```
$ ping 192.168.1.255 -b
# echo 0 >/proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
$ sudo nmap 192.168.1.1-255
```

- Tornem a consultar la taula ARP i podrem comprovar com ja tenim assignades les adreces MAC a IPs de tots els PCs de la xarxa

# Medis físics i interfícies

## ◆ Diferents components

- ◆ **DTE:** Data Terminal Equipment
- ◆ **MII:** Media Independent Interface
- ◆ **PHY:** PHYSical Layer
- ◆ Enllaç extern





# Comandes nivell Físic/Enllaç

## • Mii-tool:

```
$ dpkg -S /sbin/mii-tool  
net-tools: /sbin/mii-tool  
$ sudo apt-get install net-tools
```

```
$ sudo mii-tool  
eth0: negotiated 100baseTx-FD flow-control,  
link ok
```

## • Mii-diag

```
$ sudo apt-get install mii-diag  
$ sudo mii-diag eth0  
Basic registers of MII PHY #0: 1000 796d 0141 0c25 0de1 45e1 0005 2001.  
The autonegotiated capability is 01e0.  
The autonegotiated media type is 100baseTx-FD.  
Basic mode control register 0x1000: Auto-negotiation enabled.  
You have link beat, and everything is working OK.  
Your link partner advertised 45e1: Flow-control 100baseTx-FD 100baseTx 10baseT-FD  
10baseT, w/ 802.3X flow control.  
End of basic transceiver information.
```

# Comandes nivell Físic/Enllaç

## ◆ Comanda ethtool

```
$ sudo ethtool eth0
Settings for eth0:
  Supported ports: [ TP ]
  Supported link modes:      10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Half 1000baseT/Full
  Supports auto-negotiation: Yes
  Advertised link modes:      10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Half 1000baseT/Full
  Advertised auto-negotiation: Yes
  Speed: 100Mb/s
  Duplex: Full
  Port: Twisted Pair
  PHYAD: 0
  Transceiver: internal
  Auto-negotiation: on
  Supports Wake-on: pg
  Wake-on: g
  Current message level: 0x00000037 (55)
  Link detected: yes
```

## ◆ Ehtool a la wiki del curs

# Comandes nivell Físic/Enllaç

- ◆ **Permet modificar la configuració a nivell d'enllaç**

- ◆ Per exemple:

```
$ sudo ethtool -s eth0 speed 10 duplex
full autoneg off
```

- ◆ **Activar Wake ON LAN:**

```
$ sudo ethtool -s eth0 wol g
```

- ◆ Wake on LAN a la wiki del curs
  - ◆ No confondre amb **arranca des de xarxa amb PXE**



# Comandes nivell Físic/Enllaç

## • Ethtool exemples

- ◆ Comprovar link:
- ◆ Obtenir el driver Linux:

```
$ sudo ethtool eth0 | grep Link  
Link detected: yes
```

```
$ sudo ethtool -i eth0  
driver: skge  
version: 1.13  
firmware-version: N/A  
bus-info: 0000:02:08.0
```

- ◆ Identificar la targeta (fer parralejar la llum de link)

```
$ sudo ethtool -p eth0 10
```

- ◆ Renegociar l'enllaç:
- ◆ Realitzar un test:

```
$ sudo ethtool -r eth0
```

```
$ sudo ethtool --test eth0
```



# Arquitectures de comunicacions

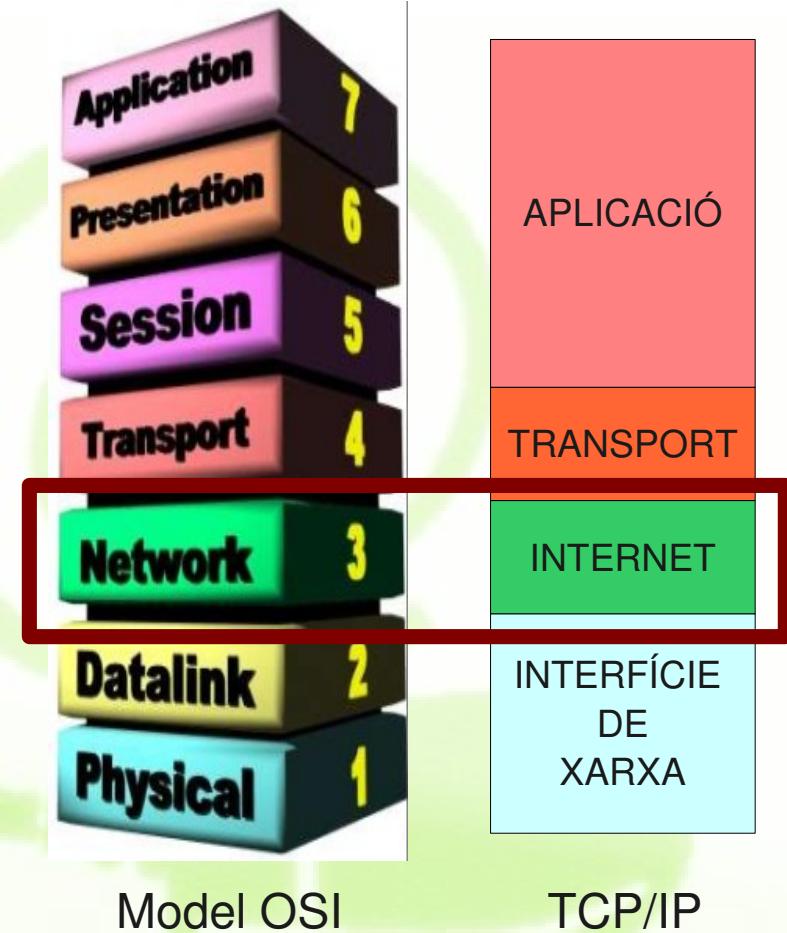
Protocol IP





# Nivell d'Internet (Nivell 2 TCP/IP) - Nivell de xarxa (Nivell 3 OSI)

El nivell de xarxa és l'encarregat de realitzar les tasques bàsiques per transportar les dades des d'un origen fins a una destinació a través d'una xarxa

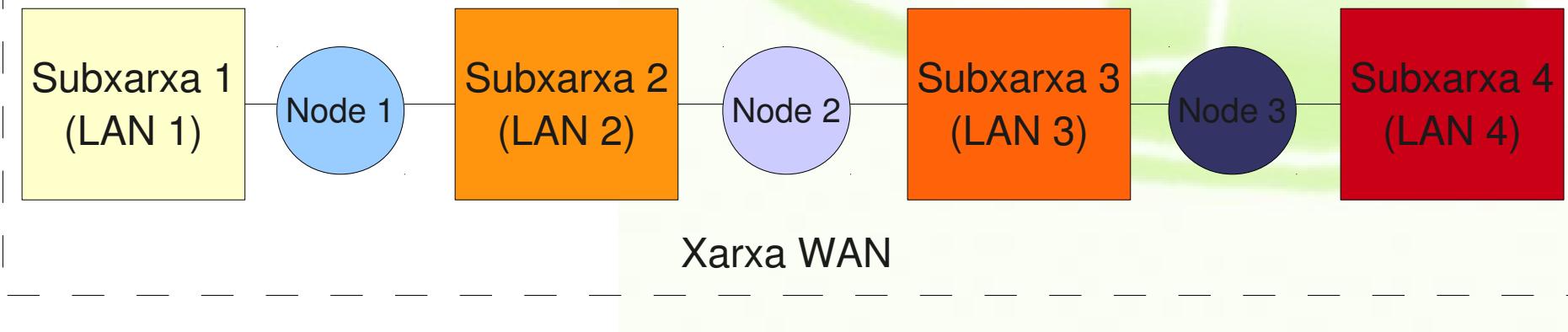


- ❖ **Model de referència OSI**
  - ❖ Nivell 3. Nivell de xarxa
- ❖ **Pila de protocols TCP/IP**
  - ❖ Nivell 2. Nivell d'Internet

# Xarxes WAN

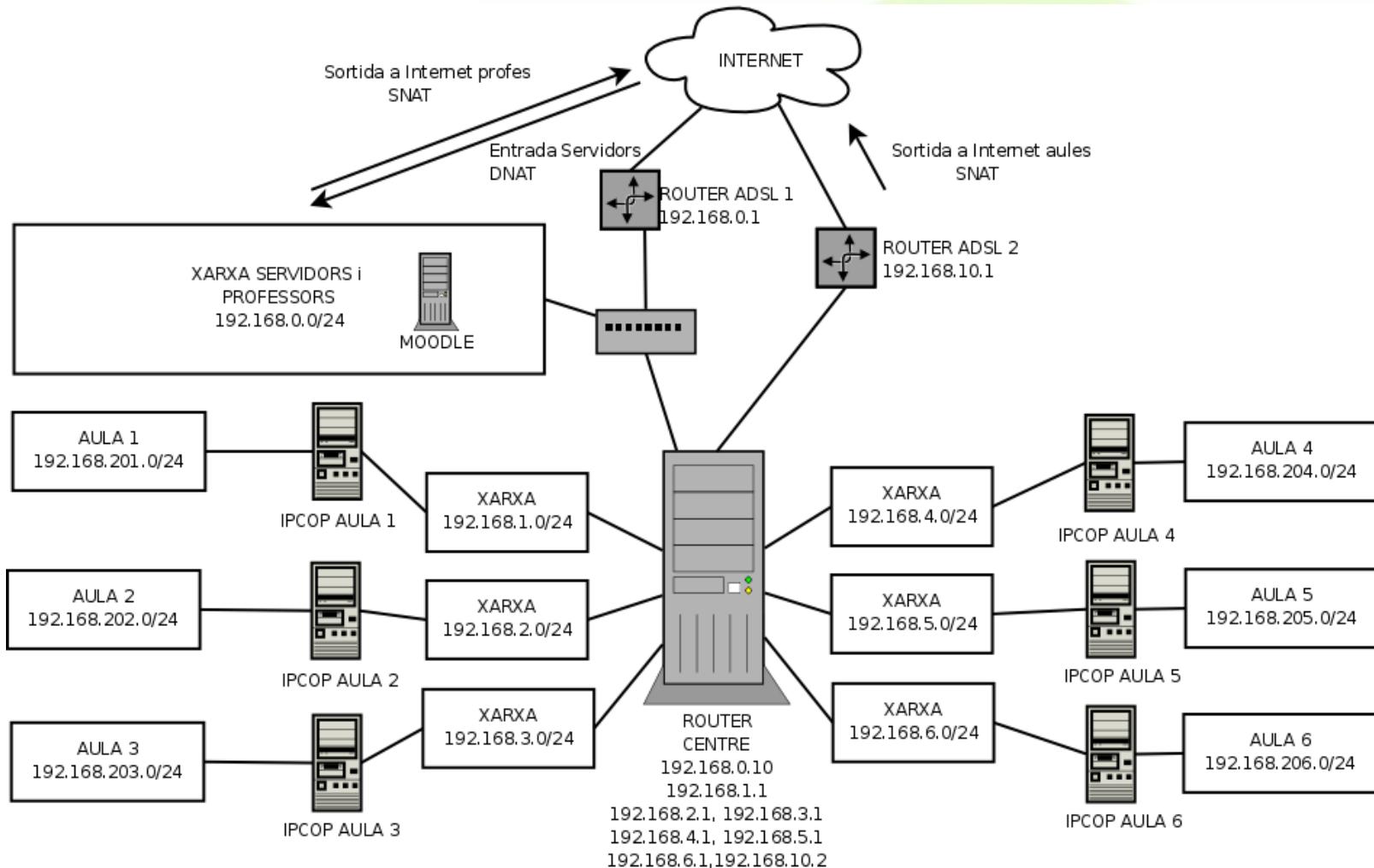
## ♦ **Wide Area Network**

- ◆ El nivell de xarxa treballa amb tot tipus de xarxes però adquireix la seva raó de ser quan treballem amb múltiples xarxes.
- ◆ A la xarxa formada per aquest subconjunt de xarxes o subxarxes de l'anomena **WAN (Wide Area Network)**



# Xarxes WAN

## Exemple Xarxa “WAN” de centre



## ◆ Control de la xarxa/subxarxa

- ◆ Treballa amb blocs de dades de xarxa (3-PDU) anomenats **paquets**.

## ◆ Funcions

- ◆ **Encaminament:** Determinar la ruta (nodes de xarxa pels quals circular) més adequada per als paquets
- ◆ **Identificació:** Els nodes han de tenir una identificació única que els permeti distingir dels altres nodes i localitzar-los a la xarxa. **ADREÇES IP**
- ◆ **Control de la congestió:** determina quins són els camins menys congestionats (similar al trànsit rodat)
- ◆ **Interconnexió de xarxes**
- ◆ **Protocol:** IP (Internet Protocol)

# Encaminament

## ♦ **Encaminament**

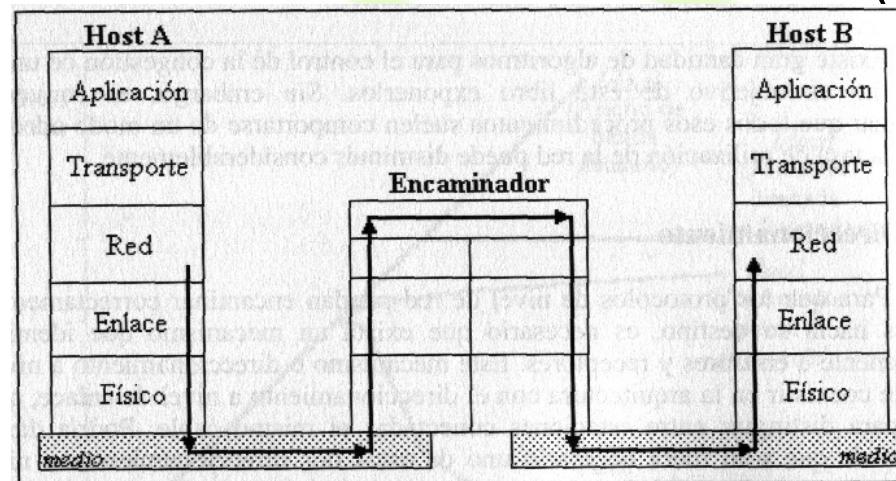
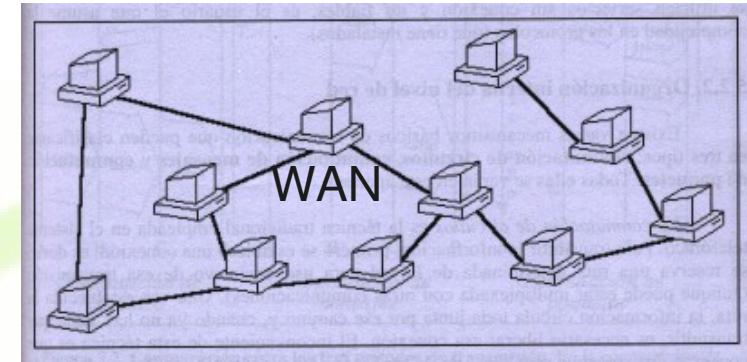
- ◆ És el mecanisme pel qual en una xarxa els paquets es fan arribar d'un origen a una destinació seguint un camí o ruta concreta.
- ◆ Cada node de la xarxa, quan rep un paquet a de prendre una decisió de que fer amb aquest paquet:
  - Quedar-se el paquet quan ell és el destinatari
  - Enviar al paquet cap a un altre node veí
  - O potser eliminar el paquet per què és incorrecta.

## ♦ **Routers/Encaminadors**

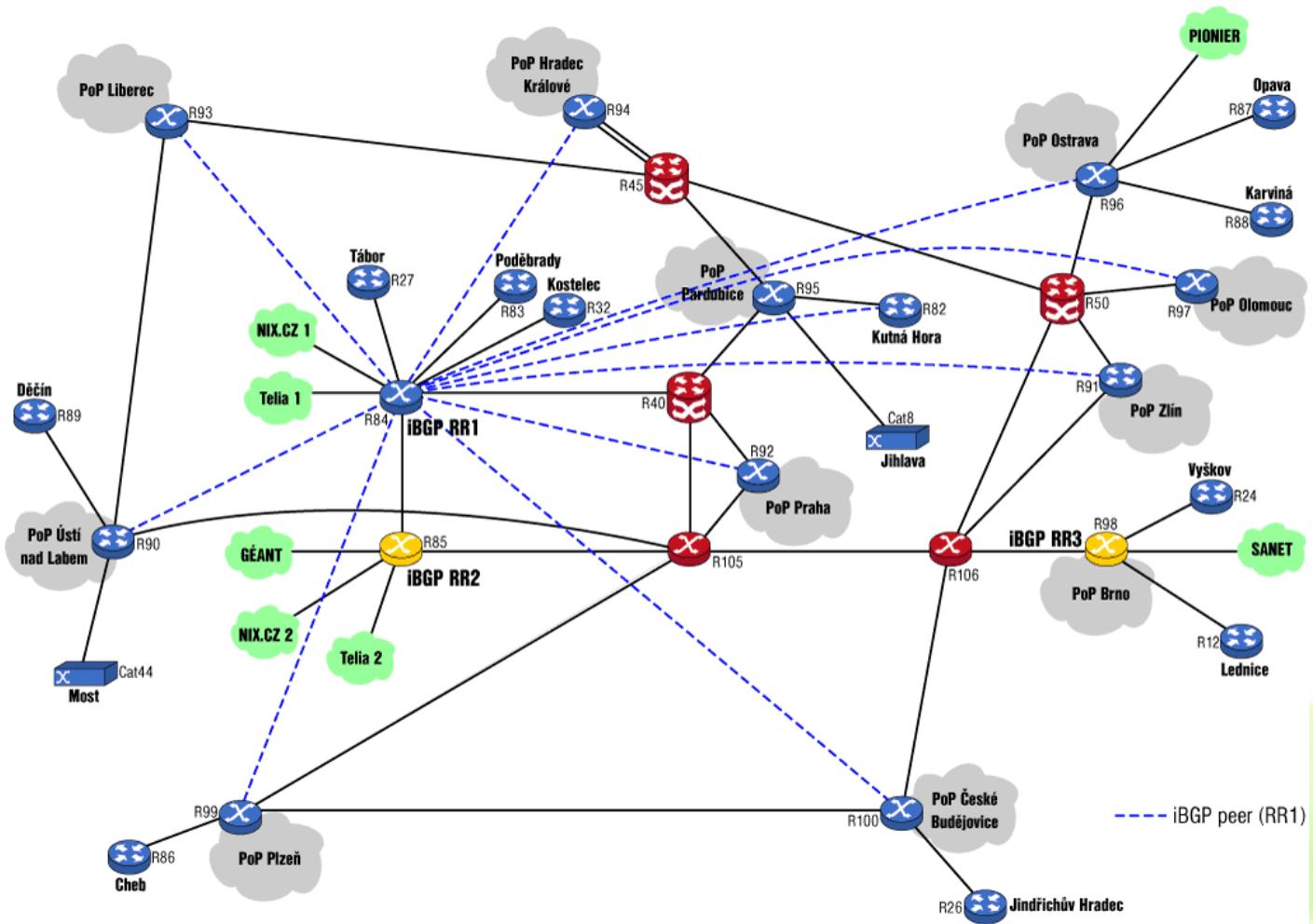
- ◆ Els routers o encaminadors són els dispositius/nodes de xarxa que s'encarreguen de l'encaminament a nivell de xarxa.

# Encaminament

- ◆ **Típic de les xarxes WAN**
  - ◆ A diferència de les xarxes
    - ◆ LAN, el medi no és compartit
- ◆ **Enllaços punt a punt (PPP)**
  - ◆ Cada node de la xarxa és un router (encaminador)



# Control de congestió



- ◆ **Control de la congestió:** determina quins són els camins menys congestionats (similar al trànsit rodat)



# Protocol IP

## • Internet Protocol

- ◆ IP és el protocol més utilitzat a nivell de xarxa
- ◆ La versió actual del protocol és la versió 4 (Ipv4) i data del 1980
- ◆ IP és un protocol **Best Effort** (El millor esforç possible)
  - Intenta transmetre els paquets el millor possible per la xarxa però no pot assegurar:
    - Que els paquets arribin
    - Que els paquets arribin correctament (sense errors)
    - Que els paquets arribin en ordre
  - El nivell superior (transport) és qui fa el control d'errors
- ◆ La **Internet Engineering Task Force (IETF)** és qui s'encarrega de definir el protocol IP.



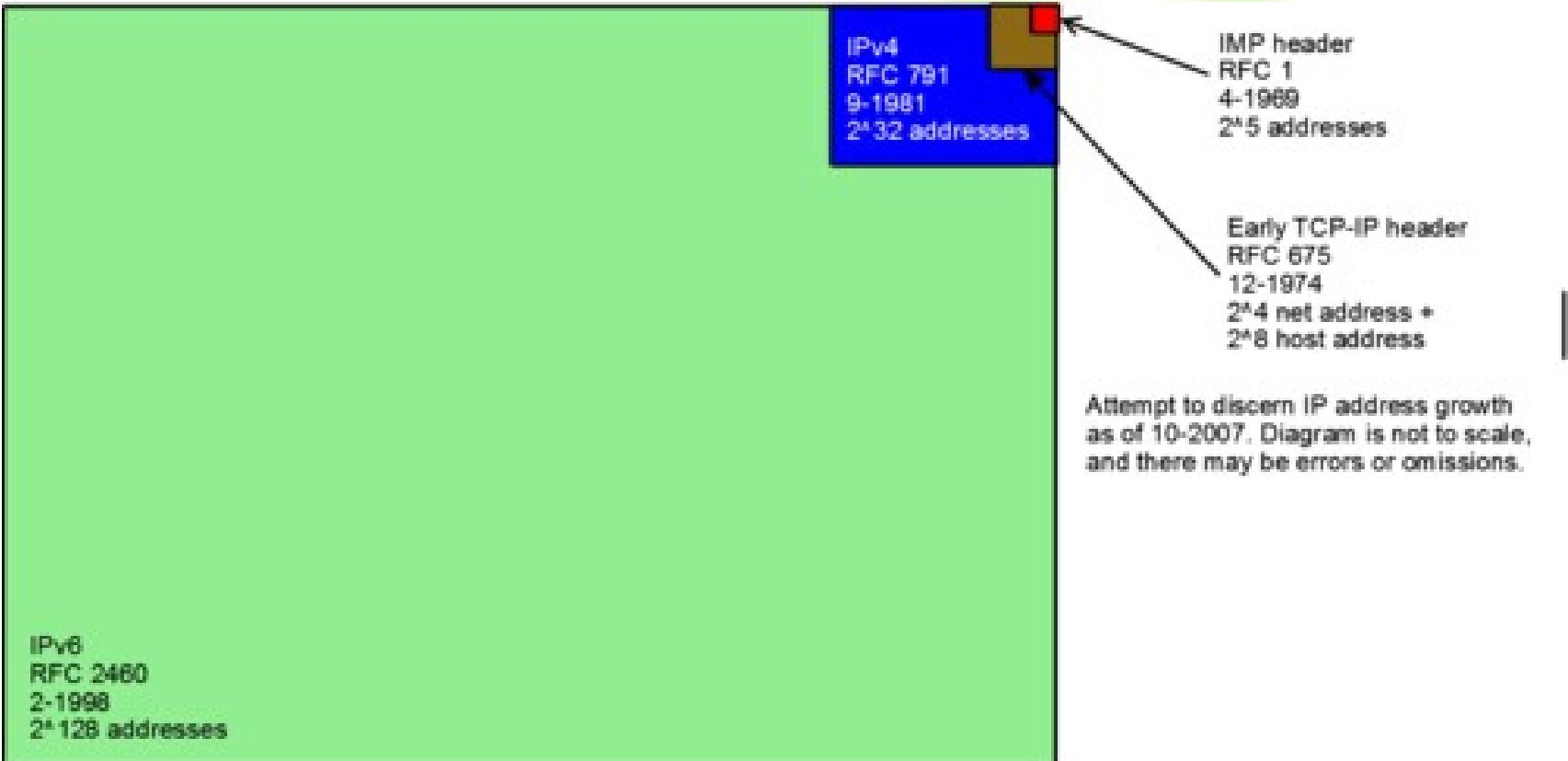
# Protocol IP

## ◆ Història

- ◆ TCP/IP va ser creat pel DoD (Departament of Defense) dels Estats Units amb l'objectiu de crear una xarxa que sobrevisques a qualsevol circumstància (per exemple un atac Nuclear).
- ◆ La idea és que les comunicacions funcionin encara que un moment concret un o més nodes de xarxa no funcionin
- ◆ IP ha anat creixent a mesura que Internet anava creixent
  - Les primeres xarxes tenien molts pocs nodes
  - La primera versió d'IP era per a xarxes de com a màxim  $2^5$  màquines (32 màquines)
  - La següent versió era per a  $2^4$ (16) xarxes i  $2^8$  (256) màquines per xarxa.
  - La versió actual suporta  $2^{32}$  (4.294.967.296 màquines)
  - Actualment uns 4 billions d'adreces no són suficients adreces. S'està implantant poc a poc el **protocol IPv6** amb  $2^{128}$  ( $3,4 \times 10^{38}$  màquines)

# Protocol IP

## Creixement d'Internet i de les adreces IP





# Adreces IP

## Protocol Ipv4

- ◆ Estan formades per 32 Bits i permeten adreçar una mica menys de 4300 milions de màquines.
- ◆ El format més comú és el decimal amb punts.

207.142.131.235 correspon als 32 bits:  
11001111.10001110.10000011.11101011

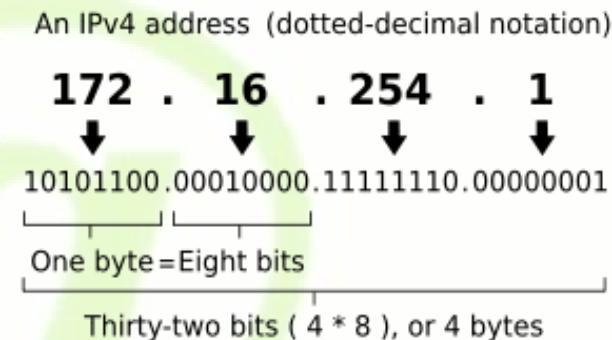
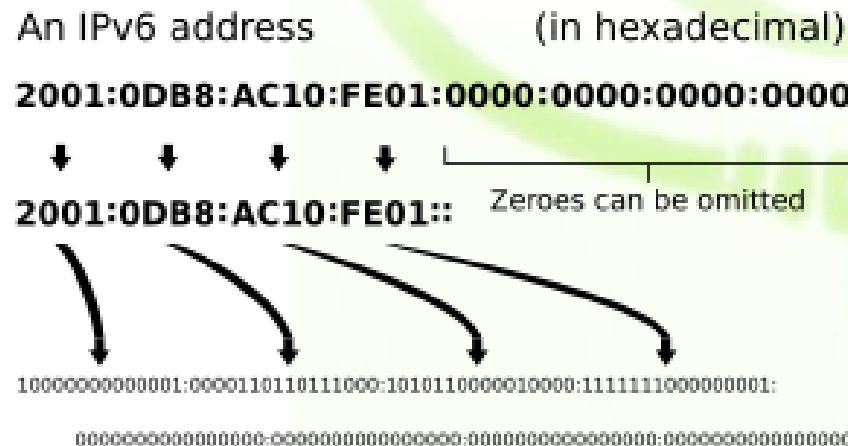
Les adreces IP identifiquen de forma unívoca un node dins d'una xarxa concreta

## Altres notacions

Notation	Value
Dot-decimal notation	207.142.131.235
Dotted Hexadecimal	0xCF.0x8E.0x83.0xEB
Dotted Octal	0317.0216.0203.0353
Hexadecimal	0xCF8E83EB
Decimal	3482223595
Octal	031743501753

# Adreces IP

- ◆ **Conversió d'adreces IPv4 a binari**
  - ◆ Hi ha una eina anomenada **ipcalc** que ens permet fer operacions amb adreces IP.
- ◆ **Conversió d'adreces IPv6 a binari**





# Adreces IP

## • **Ipv6**

- ◆ És la versió 6 del Protocol d'Internet (IP)
- ◆ Ha estat dissenyat per a substituir l'actual IPv4, ja que el seu límit en el nombre d'adreces de xarxa disponibles està començant a restringir el creixement d'Internet (l'elevat increment d'assignacions a zones d'Àsia n'està accelerant l'exhauriment).
- ◆ La nova IPv6, permetrà proporcionar als futurs telèfons cel·lulars i mòbils, una direcció fixa i pròpia per a cada un d'ells.
- ◆ IPv6 és la segona versió del Protocol d'Internet que s'ha adoptat per l'ús general. Va ser definit l'any 1996 per l'IETF a partir del document RFC 2460. També va existir un IPv5, tot i que no va ser un successor d'IPv4, sinó que va ser un protocol experimental orientat al flux d'streaming que intentava suportar veu, vídeo i so.



# Adreces IP

- ◆ Es preveu que caldrà seguir suportant el protocol IPv4 junt amb l'IPv6 durant força temps. A més, els nodes exclusivament IPv4 no són capaços de comunicar-se directament amb els nodes IPv6 i necessitaran, per tant, utilitzar algun intermediari.
- ◆ El canvi més dràstic d'IPv4 respecte d'IPv6 és la longitud de les adreces de xarxa. Les adreces IPv6 són de **128 bits**.
  - IPv4 suporta  $4.294.967.296$  ( $2$  elevat a  $32$ ) adreces de xarxa diferents, un número inadequat si a la llarga volem donar una adreça a cada persona del planeta, i menys encara per a cada cotxe, telèfon, PDA, etc.
  - **IPv6 suporta**  $340.282.366.920.938.463.463.374.607.431.768.211.456$  (2128 o 340 sextilions) adreces —prop de  $3,4 \times 10^{20}$  (340 trillions) adreces per cada polzada quadrada ( $6,7 \times 10^{17}$  o 670 mil bilions d'adreces/mm<sup>2</sup>) de la superfície de La Terra. Aquest nombre pot representar-se també com 1632, amb 32 díigits hexadecimals, cadascun dels quals pot prendre 16 valors (vegeu combinatòria).
- ◆ **Wiki del curs**



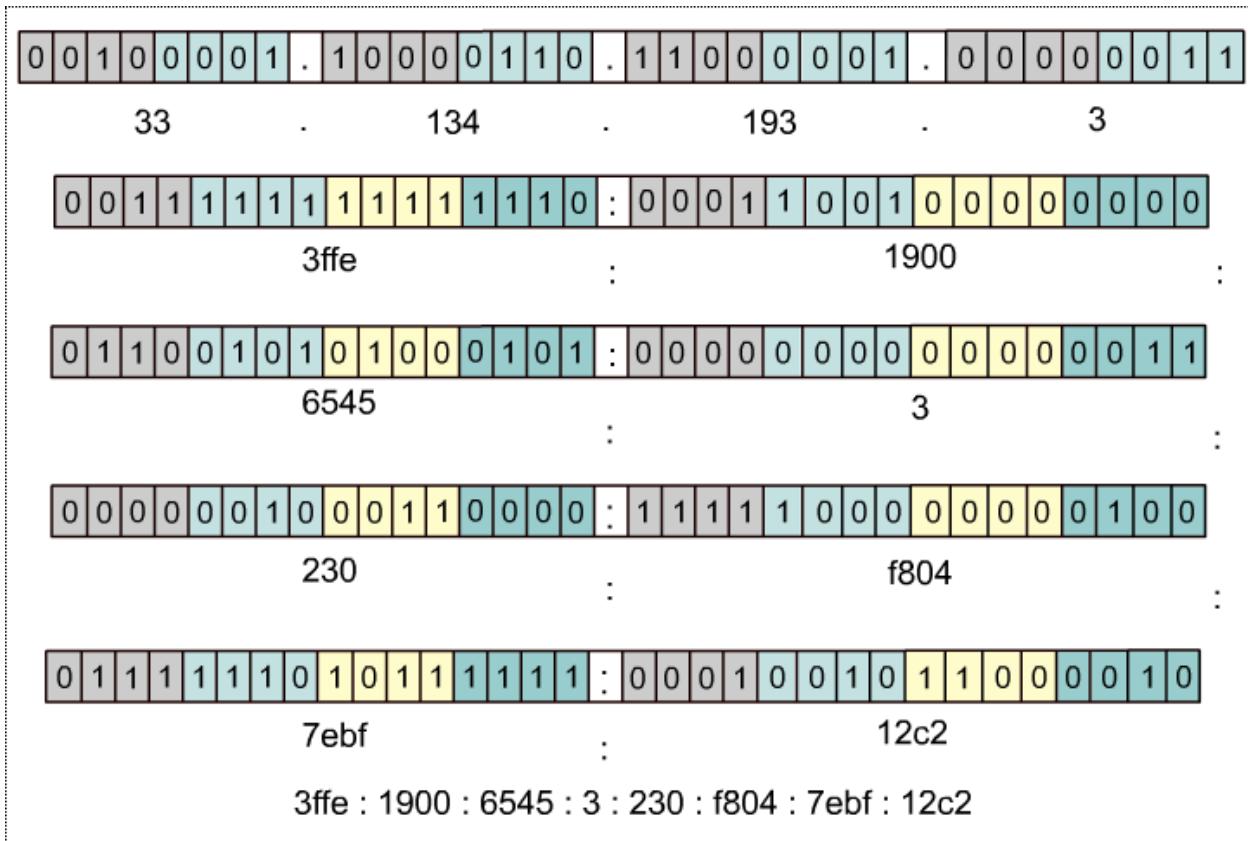
# Adreces IP

- ◆ **Les adreces IPv6, de 128 bits de longitud, s'escriuen com vuit grups de quatre díigits hexadecimals.**
  - ◆ Adreces més habituals (ping6)
    - **::/128** : l'adreça amb tot zeros s'utilitza per a indicar l'absència d'adreça, i no s'assigna cap node.
    - **::1/128** : l'adreça de loopback és una adreça que pot usar un node per a enviar-se paquets a si mateix (correspon amb 127.0.0.1 de IPv4). No pot assignar-se a cap interfície física.
    - **::ffff:0:0/96** – L'adreça IPv4 mapejada és utilitzada com un mecanisme de transició en terminals duals.
    - **fe80::/10** – El prefix d' enllaç local (< anglès link local) especifica que l'adreça només és vàlida a l'enllaç físic local.
    - **ff00::/8** : El prefix de multicast.



# Adreces IP

## Ipv6



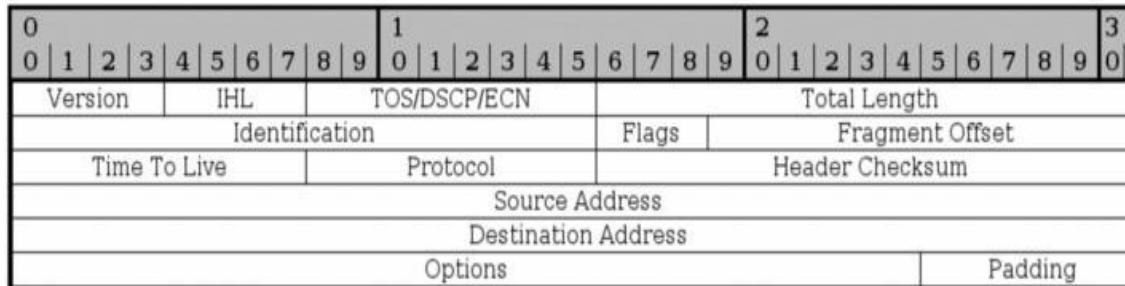
2001:0db8:85a3:08d3:1319:8a2e:0370:7334

# Paquets IP

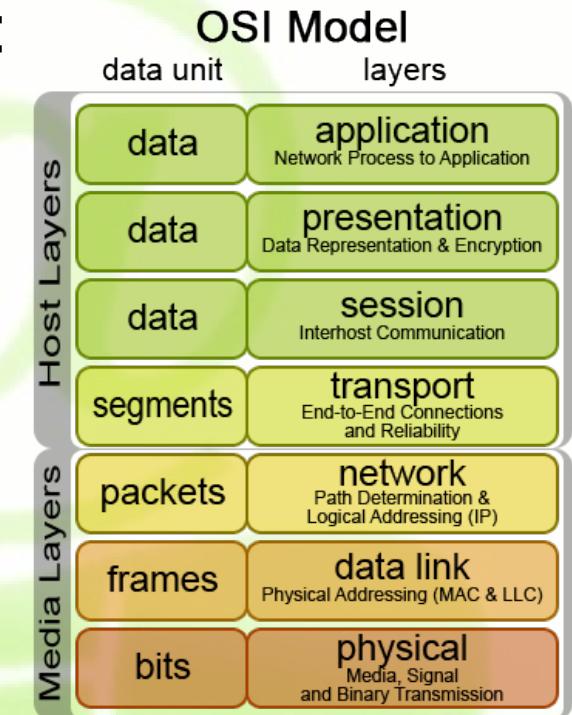
- **La unitat de dades del nivell 4 és el paquet/packet**

- Un paquet està format per dos parts:

- **Capçalera:**



- **Dades:** si les dades a transportar són moltes, les dades s'hauran de fragmentar/repartir en diferents paquets
- Consulte la [wiki del curs](#)



# Configuració IP de nodes de xarxa

- ❖ **Paràmetres necessaris per configurar un paràmetre de xarxa**
  - ❖ Paràmetres imprescindibles
    - Adreça IP
    - Màscara de xarxa
  - ❖ Paràmetres “opcionals”
    - Porta d'enllaç
    - Servidors de DNS
  - ❖ Altres paràmetres
    - Adreça de difusió, adreça de xarxa, adreça MAC



Aquests paràmetres no són imprescindibles per tal que una màquina és pugui comunicar amb altres màquines de la mateixa xarxa!

# Màscara de xarxa

- La màscara determina quins bits estan reservats a la xarxa i quins bits a les màquines.

- ◆ La màscara més utilitzada és la màscara:

**255.255.255.0**

11111111.11111111.11111111.00000000

- ◆ Tenen el format de les adreces IP però no tots els valors són possibles
  - ◆ En format binari, la màscara ha de tenir tots els uns junts i al principi, seguit d'un sèrie de ceros.
    - Només són vàlides les màscares que tenen els valors:

255, 254, 252, 248, 240, 224, 192, 128



# Màscara de xarxa

- ◆ **Per que tenen aquest format les màscares de xarxa?:**

- ◆ Els bits de l'esquerre, marcats amb uns (1) s'utilitzen per indicar la xarxa
- ◆ Els bits de la dreta, marcats amb ceros (0) s'utilitzen per identifica una màquina dins d'una xarxa concreta

- ◆ **Exemples**

- ◆ **Aula 1:** Totes les adreces IP comencen per 192.168.201 (màscara 255.255.255.0)
- ◆ **Aula 4:** Totes les adreces IP comencen per 192.168.204 (màscara 255.255.255.0)

Decimal	Binari
128	10000000
192	11000000
224	11100000
240	11110000
248	11111000
252	11111100
254	11111110
255	11111111



# Màscara de xarxa

## • La màscara 255.255.255.0:

- ◆ Ens indica que estem a una xarxa de 254 màquines
- ◆ Ens indica quines adreces IP són de la nostra xarxa
- ◆ Hi ha una adreça màxima i una adreça mínima dins de la xarxa

```
$ ipcalc 192.168.201.0/255.255.255.255.0
Address: 192.168.201.0          11000000.10101000.11001100. 00000000
Netmask: 255.255.255.0 = 24  11111111.11111111.11111111. 00000000
Wildcard: 0.0.0.255            00000000.00000000.00000000. 11111111
=>
Network: 192.168.201.0/24      11000000.10101000.11001100. 00000000
HostMin: 192.168.201.1      11000000.10101000.11001100. 00000001
HostMax: 192.168.201.254     11000000.10101000.11001100. 11111110
Broadcast: 192.168.201.255     11000000.10101000.11001100. 11111111
Hosts/Net: 254              Class C, Private Internet
```



# Màscara de xarxa

## • AULA 1

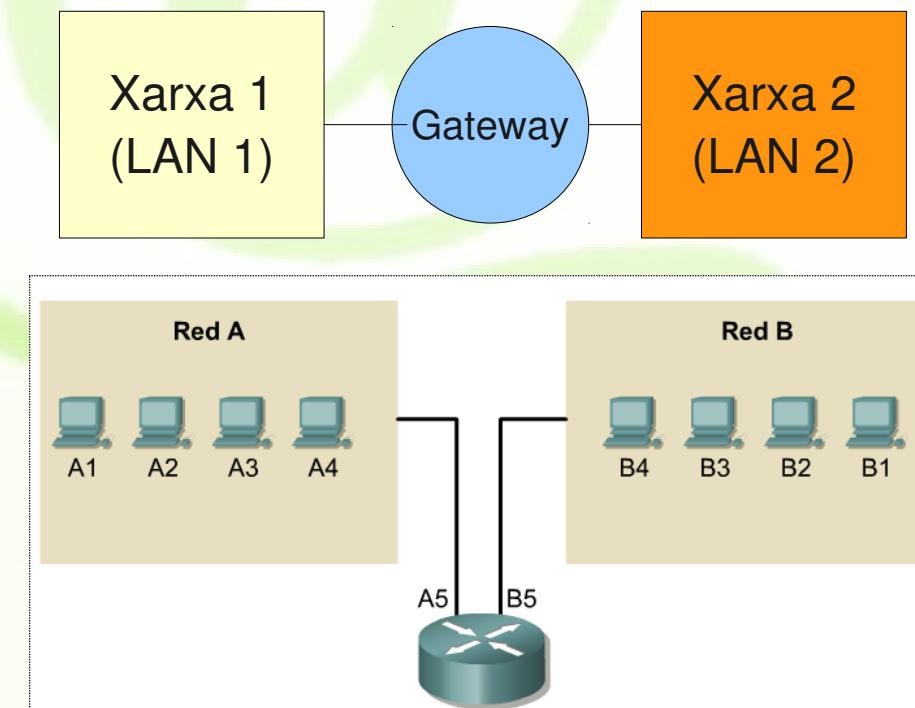
Network (N) / Hosts (H)	NNNNNNNN.NNNNNNNN.NNNNNNNN.HHHHHHHH
MÀSCARA : <b>255.255.255.0</b> /	<b>11111111.11111111.11111111.00000000</b>
IP xarxa: <b>192.168.201.x</b> /	<b>11000000.10101000.11001001.00000000</b>
Màquina1: 192.168.201.1 /	<b>11000000.10101000.11001001.00000001</b>
Màquina2: 192.168.201.2 /	<b>11000000.10101000.11001001.00000010</b>
Màquina3: 192.168.201.3 /	<b>11000000.10101000.11001001.00000011</b>
Màquina4: 192.168.201.4 /	<b>11000000.10101000.11001001.00000100</b>
Màquina5: 192.168.201.5 /	<b>11000000.10101000.11001001.00000101</b>
Màquina6: 192.168.201.6 /	<b>11000000.10101000.11001001.00000110</b>
Màquina7: 192.168.201.7 /	<b>11000000.10101000.11001001.00000111</b>
Màquina8: 192.168.201.8 /	<b>11000000.10101000.11001001.00001000</b>
Màquina9: 192.168.201.9 /	<b>11000000.10101000.11001001.00001001</b>
.....	.....
Màqui252: 192.168.201.250/	<b>11000000.10101000.11001001.1111010</b>
Màqui252: 192.168.201.251/	<b>11000000.10101000.11001001.1111011</b>
Màqui252: 192.168.201.252/	<b>11000000.10101000.11001001.1111100</b>
Màqui252: 192.168.201.253/	<b>11000000.10101000.11001001.1111101</b>
Màqui252: 192.168.201.254/	<b>11000000.10101000.11001001.1111110</b>
Màqui255: 192.168.201.255/	<b>11000000.10101000.11001001.1111111</b>

# Porta d'enllaç

- ◆ **Gateway/Pasarel·la/Router/Encaminador**
  - ◆ Té múltiples noms però la seva definició és:

La porta d'enllaç és el node de la xarxa local que ens permet connectar-nos a una altra xarxa

- ◆ Pot ser el router ADSL que ens dona accés a Internet, un router que ens connecta a la xarxa d'una institució, etc.





# Servidors de DNS

- ◆ **Són una eina per facilitar l'ús de la xarxa**
  - ◆ Ens permeten treballar a un nivell més humà (o menys de màquina) i treballar amb noms en comptes d'adreses IP
  - ◆ Per tal de treballar amb noms de màquina, hem d'utilitzar un servidor de DNS que s'encarregarà de traduir els noms en adreces IP.
  - ◆ No és imprescindible. Per exemple si s'espatlla el DNS del vostre proveïdor d'accés a Internet sempre podeu accedir a una web de a través de la seva IP

```
$ ping www.iescopernic.com
PING www.iescopernic.com (80.34.23.149) 56(84) bytes of data.
64 bytes from 149.Red-80-34-23.staticIP.rima-tde.net (80.34.23.149): icmp_seq=1
ttl=252 time=106 ms
64 bytes from 149.Red-80-34-23.staticIP.rima-tde.net (80.34.23.149): icmp_seq=2
ttl=252 time=108 ms
```

<http://80.34.23.149/moodle>



# Configuració per línia de comandes de nodes de xarxa

Quan no es disposa d'entorn gràfic (servidors), normalment els passos que cal seguir per configurar la xarxa són una combinació de comandes i d'edicions de fitxers de configuració.

- ◆ En aquest apartat veurem les comandes i els fitxers de configuració bàsics per a configurar nodes de xarxa.
  - **NOTA:** és molt important conèixer la configuració per línia de comandes per què sovint és l'única opció que tenim. També cal dir que totes les eines gràfiques acaben utilitzant aquestes comandes i/o fitxers de configuració.
- ◆ **Pàgines de consulta/referència**
  - ◆ Comandes de xarxa
  - ◆ Fitxers de configuració
  - ◆ Paquets de xarxa

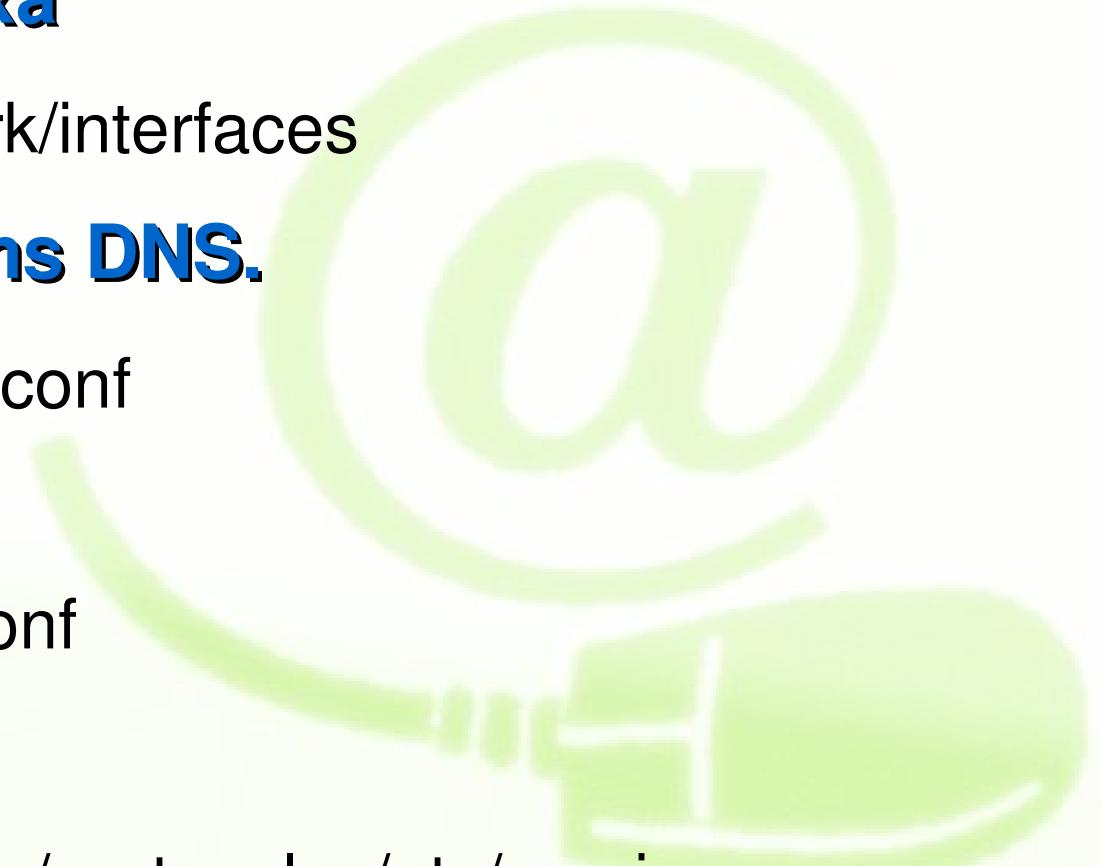


# Comandes de xarxa

- ❖ **Consulta i configuració de les interfícies**
  - ◆ ifconfig, iwconfig (obsoletes !) -> ip
- ❖ **Consulta de la taula de rutes i de la passarel·la**
  - ◆ Route (obsoletes !) -> ip
- ❖ **Consulta dels paràmetres de xarxa**
  - ◆ ipcalc
- ❖ **Comandes per apagar i encendre interfícies**
  - ◆ ifup, ifdown, dhclient
- ❖ **Comprovació de xarxes i altres comandes**
  - ◆ ping, traceroute, host, arp, netstat, ip, tracepath, nslookup, dnstracer, nmap, tcpdump, whois, iptables, etc.

# Fitxers de configuració

- ◆ **Interfícies de xarxa**
  - ◆ Fitxer /etc/network/interfaces
- ◆ **Resolució de noms DNS.**
  - ◆ Fitxer /etc/resolv.conf
  - ◆ Fitxer /etc/hosts
  - ◆ Fitxer /etc/host.conf
- ◆ **Altres fitxers**
  - ◆ /etc/networks, /etc/protocols, /etc/services...
  - ◆ Veieu l'apartat **Fitxers de configuració** a la wiki





# Paquets de xarxa

- **La majoria del programari necessari per a xarxes ja es troba instal·lat:**
  - ◆ net-tools (netbase), ifupdown, gnome-nettool, dnsutils, iproute, etc
- **Altres paquets es poden instal·lar des de els repositoris amb apt-get install:**
  - ◆ nmap, tcpdump, traceroute, dnstracer, wireshark, ipcalc, arpwatch, whois, etc
  - ◆ La imatge de l'aula ja porta instal·lats molts d'aquests paquets.
- **Exemple d'instal·lació:**

```
$ sudo apt-get install whois
```

# Configuració per línia de comandes

## ◆ Configuració estàtica

- ◆ La configuració de cada node de xarxa es realitza de forma manual pels administradors de les màquines i els paràmetres de xarxa “són fixos”.
- ◆ Cal conèixer els paràmetres de la xarxa

## ◆ Configuració dinàmica (DHCP)

- ◆ La configuració dels nodes de xarxa la realitza de forma automàtica utilitzant un servidor de DHCP. Els paràmetres dels nodes de xarxa poden ser fixos o dinàmics (varien amb el temps)
- ◆ No cal conèixer els paràmetres de la xarxa (el servidor DHCP de la xarxa ens els proporciona)

# Configuració per línia de comandes

## → Passos a seguir per configurar la xarxa

- ◆ Escollir quin tipus de configuració volem: estàtica o dinàmica
- ◆ Si la configuració és estàtica decidir (o preguntar a l'administrador de la xarxa) els 4 paràmetres bàsics de xarxa (IP, Màscara, Passarel·la/Gateway i servidors de DNS)
- ◆ Modificar la configuració de la màquina modificant els **Fitxers de Configuració** adients. **NOTA:** Cal utilitzar un editor de text per línia de comandes
- ◆ Un cop fetes les modificacions:
  - **IMPORTANT:** Cal **executar les comandes** necessàries per que la màquina apliqui la NOVA configuració.



# Paràmetres de xarxa

## • Paràmetres de xarxa

### ◆ ifconfig

- **Adreça IP.** Adreça lògica del protocol IP. Nivell de xarxa (Nivell 3 OSI). Protocol IP.
- **Adreça MAC.** Adreça física. Assignada a la NIC. Nivell MAC (Nivell 2 OSI). Protocol ARP.
- **Màscara de xarxa.** Determina quina part de l'adreça correspon a la xarxa i quina a les màquines de la xarxa.
- **Adreça de xarxa.** Ve determinada per la màscara i és l'adreça que té els bits corresponents a adreces de màquines a 0.
- **Adreça de difusió (broadcast).** Ve determinada per la màscara i és l'adreça que té l'últim octet establert a 255.



# ifconfig

- ◆ **Paquet necessari**

- ◆ **net-tools**

- ◆ **Referències**

- ◆ **man ifconfig**
  - ◆ **Article de la wikipedia**

- ◆ **Altres enllaços**

- ◆ **Exemples ifconfig**
  - ◆ [http://xarxantoni.net:8080/mediawiki/index.php/Xarxes\\_Linux#ifconfig](http://xarxantoni.net:8080/mediawiki/index.php/Xarxes_Linux#ifconfig)

```
eth0      Link encap:Ethernet HWaddr 00:00:E2:1E:4C:A3
          inet addr:10.0.0.2  Bcast:10.0.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:59108 errors:0 dropped:0 overruns:0 frame:0
          TX packets:59011 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6415496 (6.1 MiB)  TX bytes:30712998 (29.2 MiB)
          Interrupt:19 Base address:0xb400 Memory:1b500000-1b500038
```



GNU-LINUX



# ifconfig

## • **Característiques:**

- ◆ És la comanda utilitzada per configurar les interfícies de xarxa (NICs) per TCP/IP i actualment és l'estàndard dels sistemes Unix i derivats.
- ◆ Amb ifconfig es poden establir i consultar els paràmetres generals de les NICs d'una màquina.
- ◆ Amb aquesta comanda també es pot aturar o engegar la interfície de xarxa.
- ◆ Un mateix host pot tenir més d'una interfície de xarxa (p. ex. els encaminadors, connexions híbrides cable i wireless, sistemes virtuals, etc.).
- ◆ En Windows la comanda anàloga és **ipconfig**.



# ifconfig

## → **ifconfig només configura els paràmetres:**

- **IP:** Adreça lògica del protocol IP. Nivell de xarxa (Nivell 3 OSI).
- **MAC:** Adreça física. Assignada a la NIC. Nivell MAC (Nivell 2 OSI).
- **Màscara de xarxa:** determina quina part de l'adreça correspon a la xarxa i quina a les màquines de la xarxa.

## → **Tipus d'interfícies:**

- **Loopback:** lo. Encara que la màquina estigui sola (stand-alone) és necessària l'adreça de loopback.
- **Ethernet:** eth0, eth1, ...
- **Wi-Fi:** wlan0, wlan1. Tot i això sovint també s'utilitza la sintaxi.  
**d'ethernet:** ethX
- **Token Ring:** tr0, tr1, ...
- **PPP:** ppp0, ppp1, ... **vmnet0, vmnet1, vmnet8:** Interfícies virtuals vmware.

# ifconfig

## Exemple:

Primera NIC

```
$ /sbin/ifconfig
eth0      Adreça MAC          IP de la NIC
          Adreça de difusió
          HWaddr 00:0D:88:10:D2:A2
          Brd:192.168.1.1          Mask:255.255.255.0
                                      inet6 addr: fe80::20d:88ff:fe19:d2a2/64 Scope:Link
                                      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                                      RX packets:957270 errors:0 dropped:0 overruns:0 frame:0
                                      TX packets:1254234 errors:0 dropped:0 overruns:0 carrier:0
                                      collisions:0 txqueuelen:1000
                                      RX bytes:581767799 (554.8 MiB)   Interfície de loopback
                                      Interrupt:11 Base address:0x4000
lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
                                      UP LOOPBACK RUNNING MTU:16436 Metric:1
                                      RX packets:799643 errors:0 dropped:0 overruns:0 frame:0
                                      TX packets:799643 errors:0 dropped:0 overruns:0 carrier:0
                                      collisions:0 txqueuelen:0
                                      RX bytes:164196675 (156.5 MiB) TX bytes:164196675 (156.5 MiB)
```

# Configuració amb ifconfig

## • Configuració manual:

- ◆ Configuració xarxa estàtica
  - **\$ sudo ifconfig eth0 192.168.99.35 netmask 255.255.255.0 up**
- ◆ Configuració loopback
  - **\$ sudo ifconfig lo inet 127.0.0.1 up**
- ◆ Gateway
  - **\$ sudo route add default gw 192.168.1.1**
- ◆ La configuració manual no és persistent

## • IP Aliasing

- ◆ Es pot configurar una targeta amb més d'una IP
  - **\$ sudo ifconfig eth0:1 10.10.10.2 netmask 255.255.255.248**



# /etc/network/interfaces

## Exemples:

```
$ sudo gunzip /usr/share/doc/ifupdown/examples/network-interfaces.gz
$ cat /usr/share/doc/ifupdown/examples/network-interfaces
```

```
# auto eth0
# iface eth0 inet static
#     address 192.168.1.42
#     network 192.168.1.0
#     netmask 255.255.255.128
#     broadcast 192.168.1.0
#     up route add -net 192.168.1.128 netmask
#         255.255.255.128 gw 192.168.1.2
#     up route add default gw 192.168.1.200
#     down route del default gw 192.168.1.200
#     down route del -net 192.168.1.128 netmask
#         255.255.255.128 gw 192.168.1.2
```

# /etc/network/interfaces

## Fitxer de configuració principal

```
$ cat /etc/network/interfaces
# The loopback interface
auto lo
iface lo inet loopback
```

Configuració del loopback

```
auto eth0
iface eth0 inet static
    address 10.0.2.2
    netmask 255.255.254.0
    broadcast 10.0.3.255
    dns-nameserver 127.0.0.1
    dns-search intern
    gateway 10.0.2.1
```

Configuració ethernet:  
Es configuren tots els paràmetres de  
xarxa (configuració estàtica)

```
auto eth1
iface eth1 inet dhcp
```

Configuració ethernet del segon  
dispositiu de xarxa (configuració DHCP)

- Els fitxers de configuració també tenen manual:

```
$ man interfaces
```



# ifup/ifdown

- **Encarregades d'activar/desactivar les interfícies**
  - ◆ Sempre segons els paràmetres dels fitxers de configuració!
  - ◆ El sistema operatiu s'encarrega de cridar aquestes comandes a l'iniciar l'ordinador.

```
$ sudo ifup eth0
.....
.....
Listening on LPF/eth0/00:0d:88:19:d2:a2
Sending on  LPF/eth0/00:0d:88:19:d2:a2
Sending on  Socket/fallback
DHCPREQUEST on eth0 to 255.255.255.255 port 67
DHCPACK from 192.168.1.1
bound to 192.168.1.14 -- renewal in 244026 seconds.
$ sudo ifdown eth0
.....
.....
Listening on LPF/eth0/00:0d:88:19:d2:a2
Sending on  LPF/eth0/00:0d:88:19:d2:a2
Sending on  Socket/fallback
DHCPRELEASE on eth0 to 192.168.1.1 port 67
```

- ◆ Ifup a la wiki del curs



# Inicialització de la xarxa a Debian/Ubuntu

## • S'utilitzen scripts d'inicialització SystemV

- ◆ La xarxa s'executa al nivell S (Single User Mode) (primer de tots, fins i tot abans del nivell 1)
- ◆ Els Links:
  - /etc/rcS.d/S08loopback
  - /etc/rcS.d/S40networking
- ◆ Executen els scripts d'inicialització
  - /etc/init.d/loopback start
  - /etc/init.d/networking start
- ◆ UPSTART
  - /etc/init/networking.conf



# Inicialització de la xarxa a Debian/Ubuntu

## • **Carpetes .d**

```
$ ls /etc/network |  
grep "\.d"  
if-down.d  
if-post-down.d  
if-pre-up.d  
if-up.d
```

- ◆ Scripts a executar durant l'arrancada/aturada de xarxa
- ◆ Per exemple, assegurar-se que les targetes de xarxa es queden a l'espera de rebre paquets wake on lan després d'aturar la xarxa.

```
$ sudo joe /etc/network/if-post-down.d/wol-enable
```

Amb el contingut:

```
#!/bin/bash  
/usr/bin/sudo /usr/sbin/ethtool -s eth0 wol g
```

Cal que sigui executable:

```
$ sudo chmod +x /etc/network/if-post-down.d/wol-enable
```



# Xarxes Wireless

- Protocol IEEE 802.11 (WIFI). Protocols b i g



Protocol	Release Date	Op. Frequency	Data Rate (Typ)	Data Rate (Max)	Range (Indoor)	Range (Outdoor)
Legacy	1997	2.4-2.5 GHz	1 Mbit/s	2 Mbit/s	?	?
802.11a	1999	5.15-5.35/5.47-5.725/5.725-5.875 GHz	25 Mbit/s	54 Mbit/s	~30 meters (~100 feet)	?
802.11b	1999	2.4-2.5 GHz	6.5 Mbit/s	11 Mbit/s	~100 meters	?
802.11g	2003	2.4-2.5 GHz	25 Mbit/s	54 Mbit/s	~30 meters	?
802.11n	2006 (draft)	2.4 GHz or 5 GHz bands	200 Mbit/s	540 Mbit/s	~50 meters	?

- Sistemes d'encriptació:

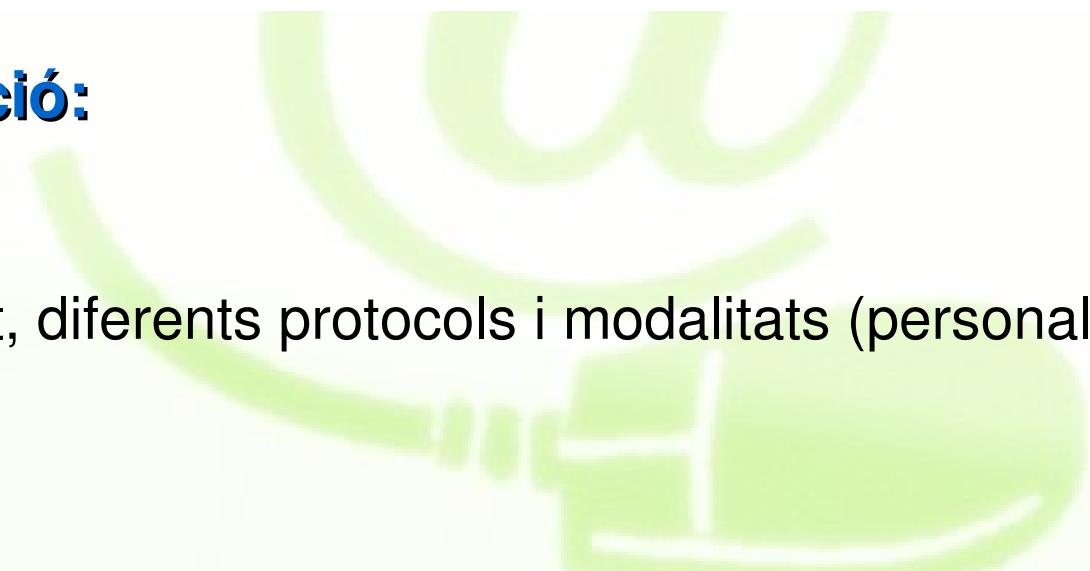
- WEP (64 i 128 bits)

- WPA. Més seguretat, diferents protocols i modalitats (personal i enterprise).

- Claus d'accés

- Hexadecimals (4 bits) | ASCII (1 Byte) | WEP

- WIFI a la wiki del curs





# Xarxes Wireless

## • Comanda iwconfig

```
$ iwconfig
lo    no wireless extensions.
eth0   no wireless extensions.
eth1   IEEE 802.11g ESSID:"WLAN_8A"
        Mode:Managed Frequency:2.427 GHz Access Point:
        00:16:38:89:F6:57
        Bit Rate:54 Mb/s Tx-Power=20 dBm Sensitivity=8/0
        Retry limit:7 RTS thr:off Fragment thr:off
        Power Management:off
        Link Quality=94/100 Signal level=-38 dBm Noise level=-91 dBm
        Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
        Tx excessive retries:0 Invalid misc:0 Missed beacon:0
```

## • /etc/network/interfaces

```
iface eth1 inet dhcp
    wireless-essid WLAN_8A
    wireless-key
        s:C0030GR89FC8A
        b:C0030GR89FC8A
```

## • [http://acacha.dyndns.org/mediawiki/index.php/Xarxes\\_Linux\\_Wireless](http://acacha.dyndns.org/mediawiki/index.php/Xarxes_Linux_Wireless)



# Xarxes Wireless

- ❖ **Paràmetres de xarxa wireless**
  - ❖ Paràmetres de xarxa amb cable+
  - ❖ wireless-essid: Identificador de la xarxa
  - ❖ wireless-key: Clau WEB o WAP
    - Si la clau està en format ASCII hem de posar **s:** davant
  - ❖ wireless\_mode managed | ad-hoc
- ❖ **Comandes iw** (iwlist, iwgetssid, iwspy...)
- ❖ **Paquet:** wireless-tools
- ❖ **man wireless**



# Fitxers de configuració de DNS

- ◆ **/etc/resolv.conf**

```
$ cat /etc/resolv.conf
nameserver 192.168.1.3
nameserver 192.168.1.1
```

- ◆ **/etc/hosts**

```
$ cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      ubuntu-sala
192.168.1.3    acacha.dyndns.org
```

- ◆ **/etc/host.conf (obsolet. Ara NSSWITCH)**

```
$ cat /etc/host.conf
# The "order" line is only used by old versions of
the C library.
order hosts,bind
multi on
```

- ◆ **/etc/hostname**

```
$ cat /etc/hostname
ubuntu-sala
```



# Fitxers de configuració de DNS

## • Client DNS a la wiki del curs

## • /etc/hosts

```
$ cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      ubuntu-sala
192.168.1.3    acacha.dyndns.org
```

## • /etc/host.conf i /etc/nsswitch.conf

order Linux standard library, libc (libc4 | libc5)

New Linux standard library, glibc (libc6)

```
$ cat /etc/host.conf
# The "order" line is only used by old
# versions of the C library.
order hosts,bind
multi on
```

```
$ cat /etc/nsswitch.conf
...
hosts:            files mdns4_minimal
[NOTFOUND=return] dns mdns4
...
```

## • /etc/hostname

```
$ cat /etc/hostname
ubuntu-sala
```



# Client DNS

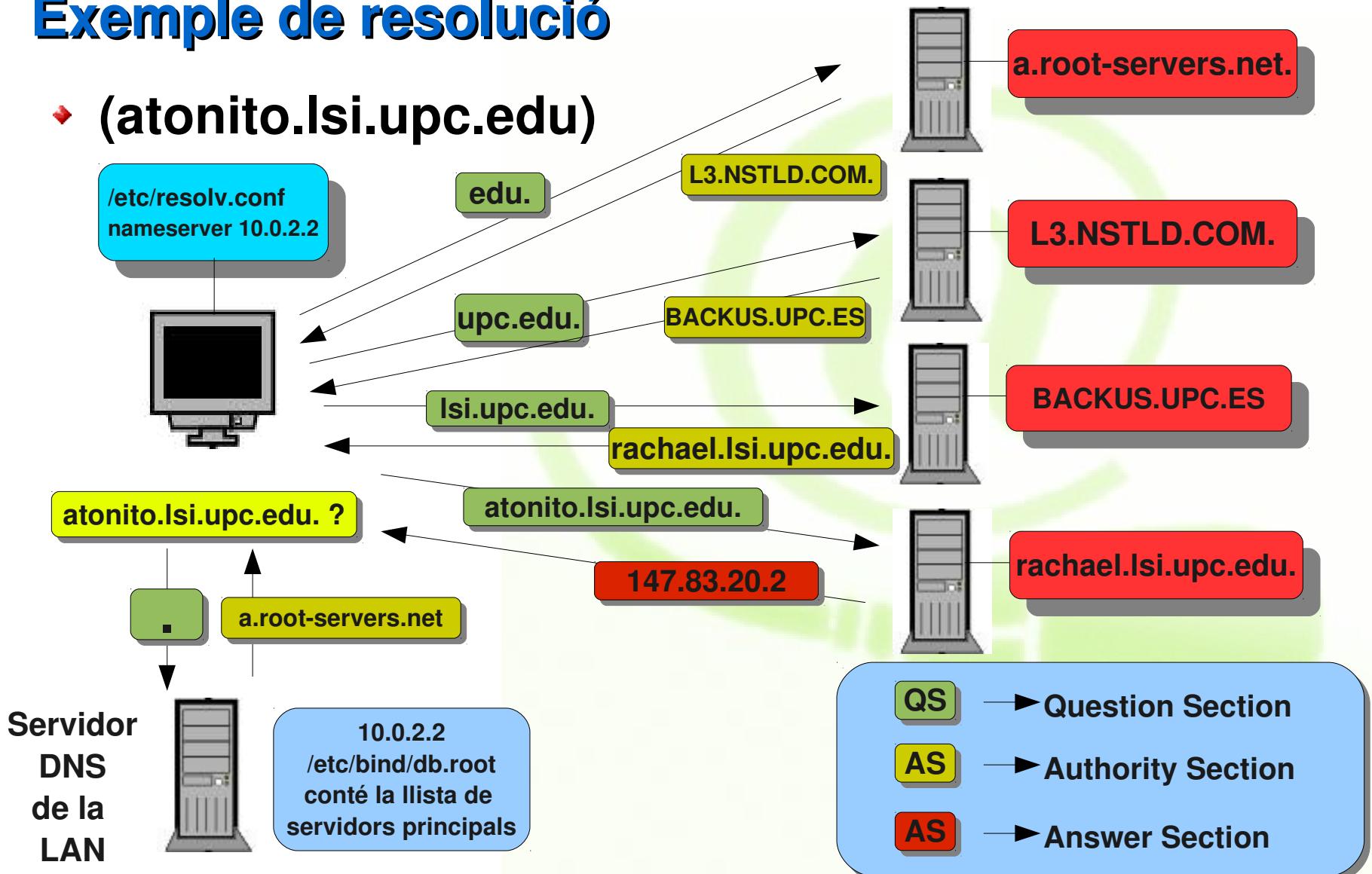
## • **Llibreria resolver (client DNS)**

- ◆ El terme resolver no fa referència a cap aplicació especial sinó a una biblioteca de C. La biblioteca resolver és una col·lecció de funcions que poden ser trobades a la llibreria estàndard de C. Les rutines principals són:
  - **gethostbyname(2)**: Obté totes les IP associades a un nom de màquina
  - **gethostbyaddr (2)**: Rutina inversa a l'anterior.
- ◆ Aquestes llibreries es poden configurar en temps d'execució amb els fitxers de configuració de resolver:
  - /etc/host.conf
  - /etc/nsswitch.conf
  - /etc/resolv.conf
- ◆ Consulteu la [wiki del curs](#)

# Resolució de noms (DNS)

- Exemple de resolució

- (atonito.lsi.upc.edu)





# Comandes DNS

## ◆ Comandes

### ◆ dig

- És una utilitat de resolució de noms amb DNS.

### ◆ dnstrace

- Rastreja la cadena de servidors DNS utilitzats per a resoldre una adreça.

## ◆ Exemple de resolució. Comanda dig

```
$ dig .  
.....  
;; QUESTION SECTION:  
;.  
;; AUTHORITY SECTION:  
.  
NSTLD.VERISIGN-GRS.COM. 143 IN SOA A.ROOT-SERVERS.NET.  
.....
```

## ◆ Paquet : dnsutils



# Resolució de noms (DNS)

```
$ dig edu.  
.....  
;; QUESTION SECTION:  
;edu.                      IN      A  
;; AUTHORITY SECTION:  
edu.           86400    IN      SOA     L3.NSTLD.COM.  
NSTLD.VERISIGN-  
GRS.COM. ....
```

```
$ dig upc.edu.  
.....  
;; QUESTION SECTION:  
;upc.edu.                  IN      A  
;; AUTHORITY SECTION:  
upc.edu.        149289    IN      NS      EULER.UPC.ES.  
upc.edu.        149289    IN      NS      BACKUS.UPC.ES.  
.....
```

```
$ dig upc.edu.  
.....  
;; QUESTION SECTION:  
;atonito.lsi.upc.edu.       IN      A  
;; ANSWER SECTION:  
atonito.lsi.upc.edu.   172800    IN      A      147.83.20.2  
.....
```

# Resolució de noms (DNS)

## Exemple dnstracer

```
$ dnstracer -s B.ROOT-SERVERS.NET www.upc.edu
Tracing to www.upc.edu[a] via B.ROOT-SERVERS.NET, maximum of 3 retries
B.ROOT-SERVERS.NET (192.228.79.201)
  | \__ H3.NSTLD.COM [edu] (192.54.112.32)
  |   | \__ BACKUS.UPC.ES [upc.edu] (147.83.2.3) Got authoritative answer
  |   [received type is cname]
  |     \__ EULER.UPC.ES [upc.edu] (147.83.2.10) Got authoritative answer
  |   [received type is cname]
  | \__ M3.NSTLD.COM [edu] (192.55.83.32)
  |   | \__ BACKUS.UPC.ES [upc.edu] (147.83.2.3) (cached)
  |   \__ EULER.UPC.ES [upc.edu] (147.83.2.10) (cached)
  | \__ G3.NSTLD.COM [edu] (192.42.93.32)
  |   | \__ BACKUS.UPC.ES [upc.edu] (147.83.2.3) (cached)
  |   \__ EULER.UPC.ES [upc.edu] (147.83.2.10) (cached)
  | \__ C3.NSTLD.COM [edu] (192.26.92.32)
  |   | \__ EULER.UPC.ES [upc.edu] (147.83.2.10) (cached)
  |   \__ BACKUS.UPC.ES [upc.edu] (147.83.2.3) (cached)
  | \__ E3.NSTLD.COM [edu] (192.12.94.32)
  |   | \__ EULER.UPC.ES [upc.edu] (147.83.2.10) (cached)
  |   \__ BACKUS.UPC.ES [upc.edu] (147.83.2.3) (cached)
  ....
```



# Resolució de noms (DNS)

## • Resolució inversa. Comanda host

```
$ host 147.83.194.21
21.194.83.147.in-addr.arpa domain name pointer upc.edu.
21.194.83.147.in-addr.arpa domain name pointer www.upc.es.
21.194.83.147.in-addr.arpa domain name pointer raiden.upc.es.
21.194.83.147.in-addr.arpa domain name pointer upc.es.
```

## • Resolució directa. Comanda ping

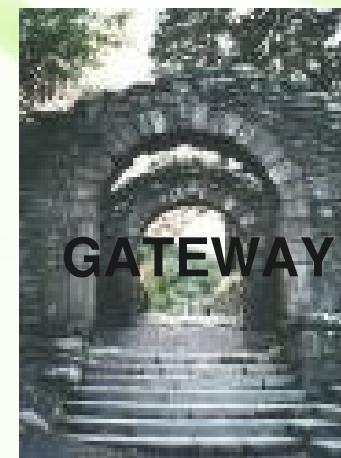
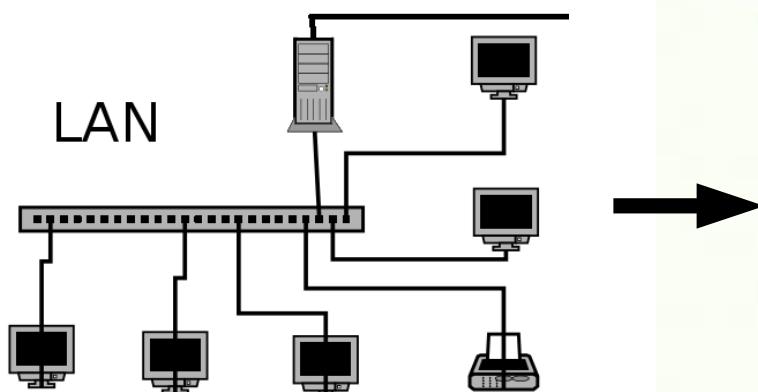
```
$ ping www.upc.edu
PING www.upc.es (147.83.194.21) 56(84) bytes of data.
64 bytes from upc.edu (147.83.194.21): icmp_seq=1 ttl=50 time=86.2 ms
64 bytes from upc.edu (147.83.194.21): icmp_seq=2 ttl=50 time=86.1 ms
64 bytes from upc.edu (147.83.194.21): icmp_seq=3 ttl=50 time=86.1 ms
64 bytes from upc.edu (147.83.194.21): icmp_seq=4 ttl=50 time=86.4 ms
```

# Gateway. Comanda route

- La passarel·la o gateway es configura mitjançant la comanda route

```
$ route add default gw 192.168.1.1
```

- És comú que la primera IP de la xarxa sigui el gateway (.1) però no obligatori.



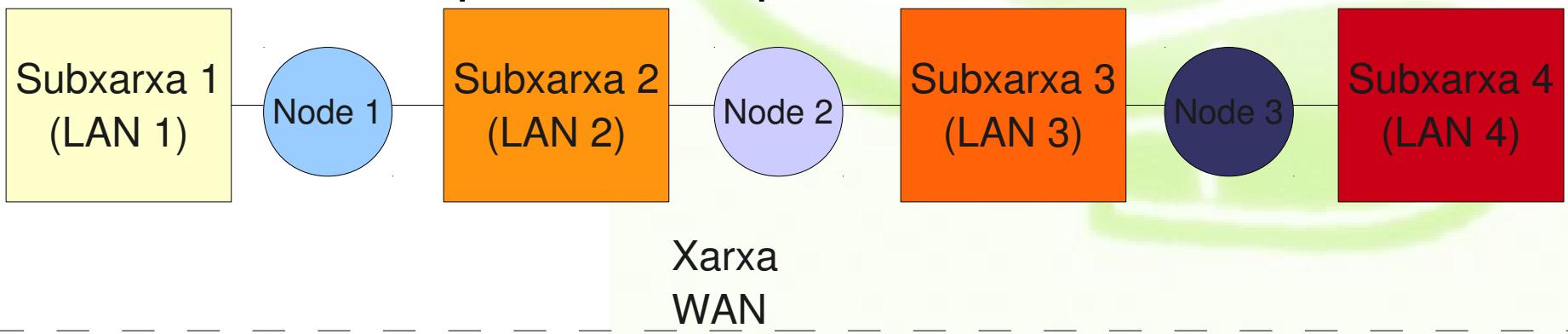
# Xarxes WAN o Múltiples LAN

## ♦ **Wide Area Network**

- ◆ El nivell de xarxa treballa amb tot tipus de xarxes però adquireix la seva raó de ser quan treballem amb múltiples xarxes.

- ◆ Xarxes WAN (**Wide Area Network**) o

- ◆ **Xarxa Complexa: Múltiples LAN**





# route

## ◆ Comanda route

```
$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.196.0   *              255.255.255.0   U        0      0        0 vmnet8
192.168.1.0     *              255.255.255.0   U        0      0        0 eth0
192.168.252.0   *              255.255.255.0   U        0      0        0 vmnet1
default         192.168.1.1   0.0.0.0        UG       0      0        0 eth0
```

## ◆ Característiques:

- La comanda route permet manipular i visualitzar les taules d'enrutament del kernel dels sistemes GNU/Linux.
- El tema d'enrutament i interconnexió de xarxes d'àrea local el veurem a la unitat didàctica 6: Interconnexió de xarxes d'àrea local.

## ◆ Proporcionat pel paquet netbase

◆ [http://acacha.dyndns-org/mediawiki/index.php/Xarxes\\_Linux#route](http://acacha.dyndns-org/mediawiki/index.php/Xarxes_Linux#route)

# traceroute

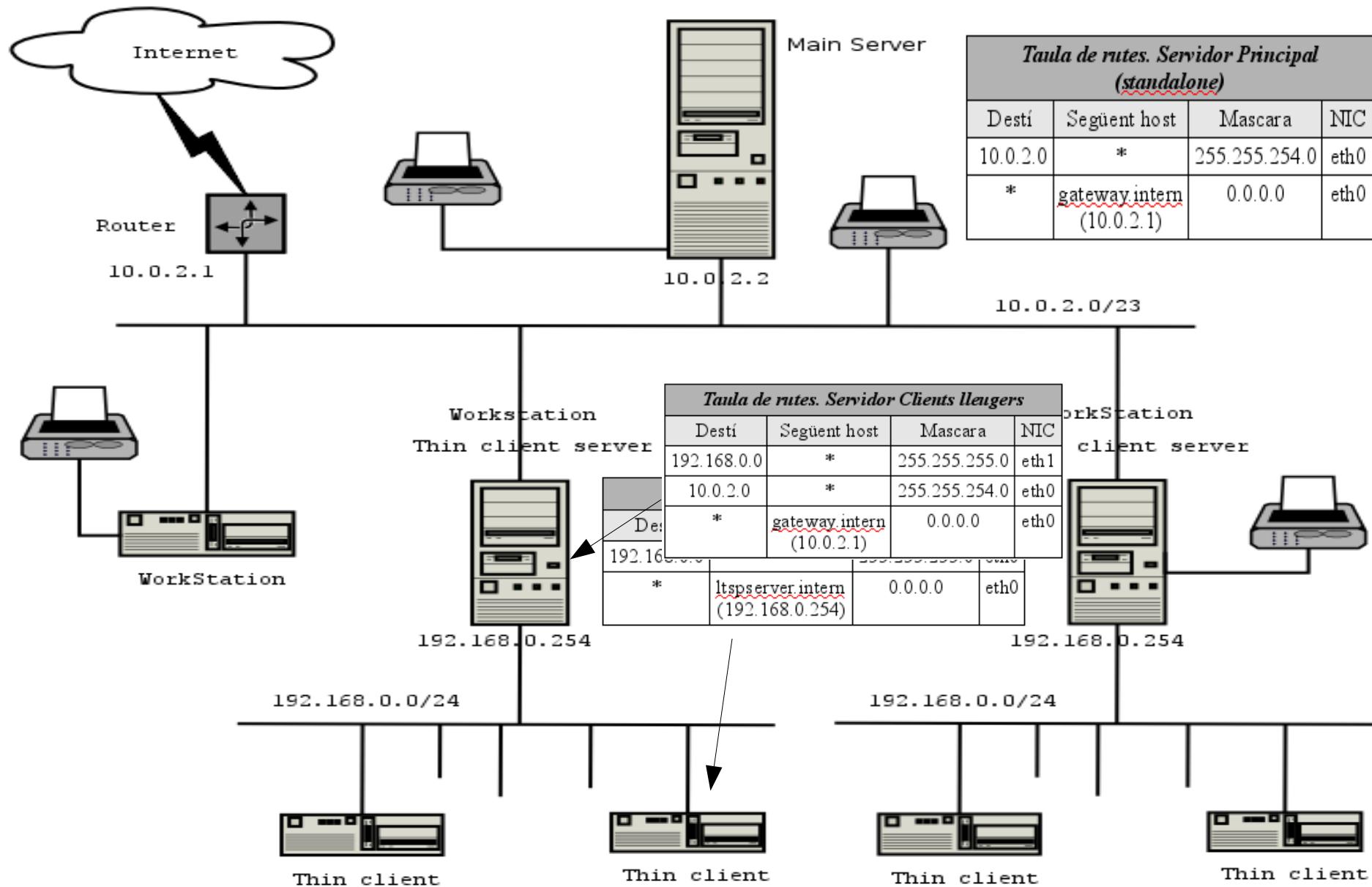
## Exemple

```
$ sudo traceroute www.jazztel.es
traceroute to www.jazztel.es (212.106.192.74), 64 hops max, 40 byte
packets
 1  192.168.1.1 (192.168.1.1)  1 ms  1 ms  1 ms
 2  inversas.2g.jazztel.es (87.219.198.1)  39 ms  38 ms  39 ms
 3  10.255.136.254 (10.255.136.254)  54 ms  49 ms  50 ms
 4  inversas.2g.jazztel.es (87.216.0.2)  38 ms  38 ms  38 ms
 5  inversas.2g.jazztel.es (87.216.0.1)  243 ms  177 ms  222 ms
 6  208.175.154.177 (208.175.154.177)  42 ms  37 ms  38 ms
 7  ge-7-1-0-zcrl.bap.cw.net (208.175.154.38)  37 ms so-1-0-0-
      ycr1.bap.cw.net (208.175.154.42)
  .....
 11  * * *
 12  * * *
```

## Utilitats

- Utilitzat conjuntament amb la comanda ping es pot utilitzar per detectar els punts conflictius de l'enllaç entre dues màquines.
- Per comprovar la configuració de les taules de rutes.

# Taules d'enrutament



# Altres distribucions Linux

- ❖ **Altres distribucions Linux**
  - ❖ /etc/sysconfig/network/ifcfg-\* (SUSE) amb DHCP

```
$ cat /etc/sysconfig/network/ifcfg-eth0
BOOTPROTO='dhcp'
BROADCAST=''
ETHTOOL_OPTIONS=''
IPADDR=''
MTU=''
NAME='AMD PCnet - Fast 79C971'
NETMASK='255.255.255.0'
NETWORK=''
REMOTE_IPADDR=''
STARTMODE='auto'
UNIQUE='rBUF.weGuQ9ywYPF'
USERCONTROL='no'
_nm_name='bus-pci-0000:00:10.0'
```



# Comandes

- **Per tal que la màquina apliqui els canvis del fitxer de configuració cal executar:**

```
$ sudo /etc/init.d/networking restart
```

- ◆ Aquesta comanda s'executa sola cada cop que inicieu la màquina. Si després es fa algun canvi l'heu d'executar vosaltres mateixos.

```
$ sudo ifdown ethx  
$ sudo ifup ethx
```

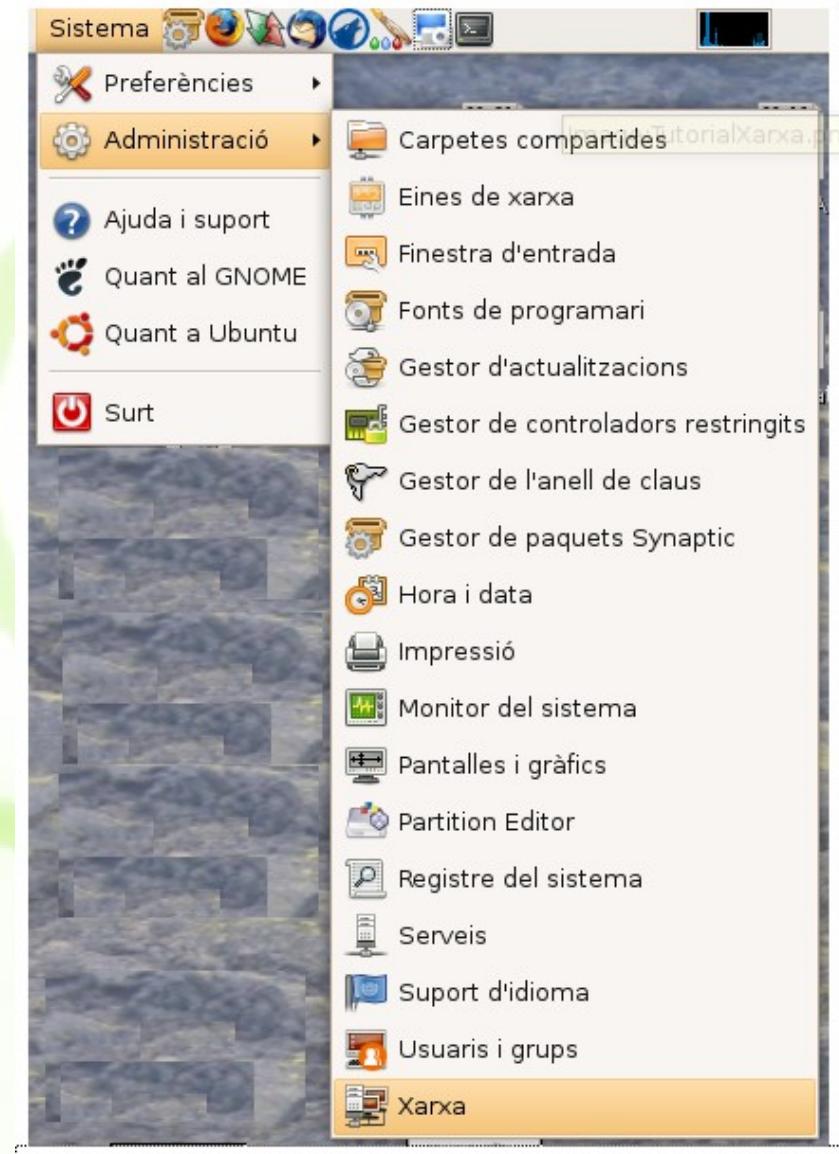
- **També podeu utilitzar les comandes:**

- ◆ On x és el número de la vostra interfície de xarxa
- ◆ La comanda down apaga la interfície de xarxa i la comanda up l'encén
- ◆ DHCP

```
$ sudo dhclient ethx
```

# Configuració gràfica de nodes de xarxa

- La configuració des de el menú **Administració** i l'opció **Xarxa**
- NM-> Network Manager**





# Configuració gràfica de nodes de xarxa

## Ubuntu

- ◆ La configuració gràfica sempre acabarà modificant els fitxers de configuració
- ◆ Podeu comprovar les modificacions als fitxers
- ◆ Podeu utilitzar l'entorn gràfic per aprendre com funcionen els fitxers de configuració
- ◆ web del professor



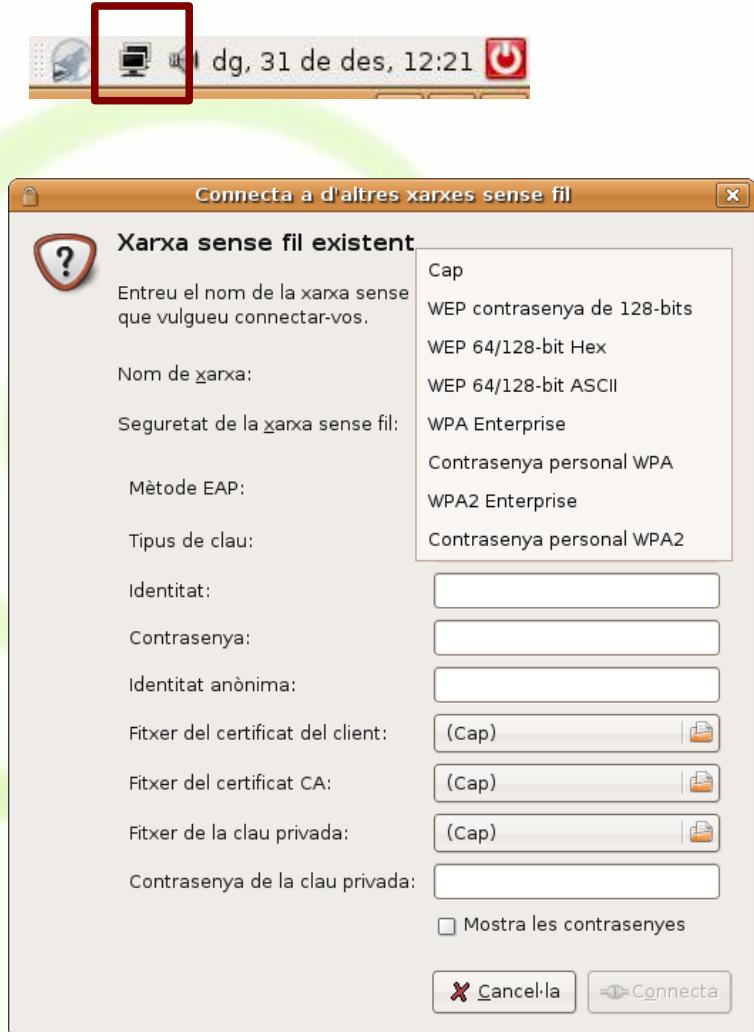
# NetworkManager

## • Aplicació gràfica Gnome



## • Ús avançat i simple de la xarxa:

- ◆ Canvi de dispositius en calent
- ◆ Roaming entre xarxes wireless
- ◆ Suport WEP i WPA



# Eines gràfiques de xarxa

**Eines gràfiques de xarxa**



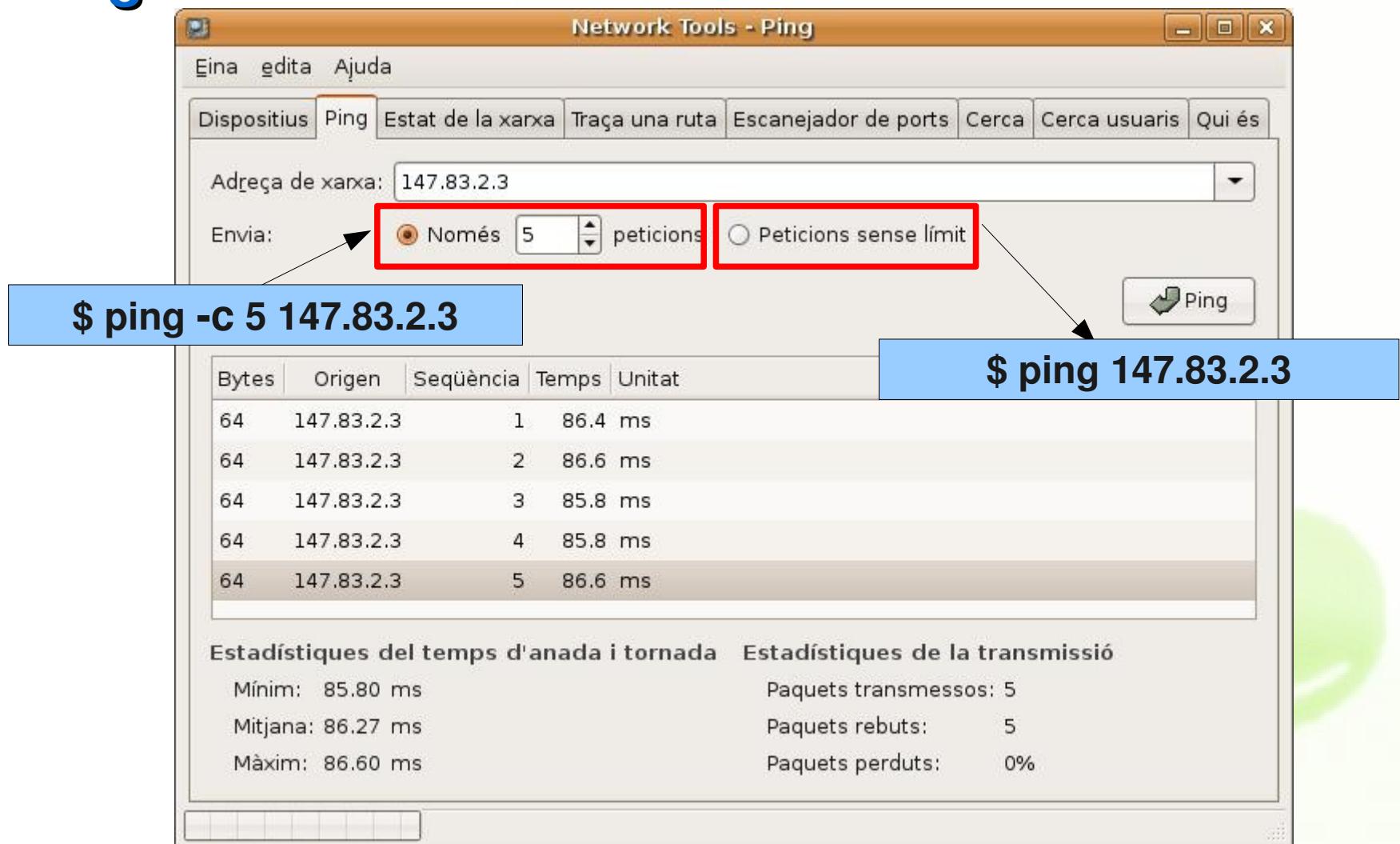
# Gnome-Neetool

## → Configuració de dispositius:



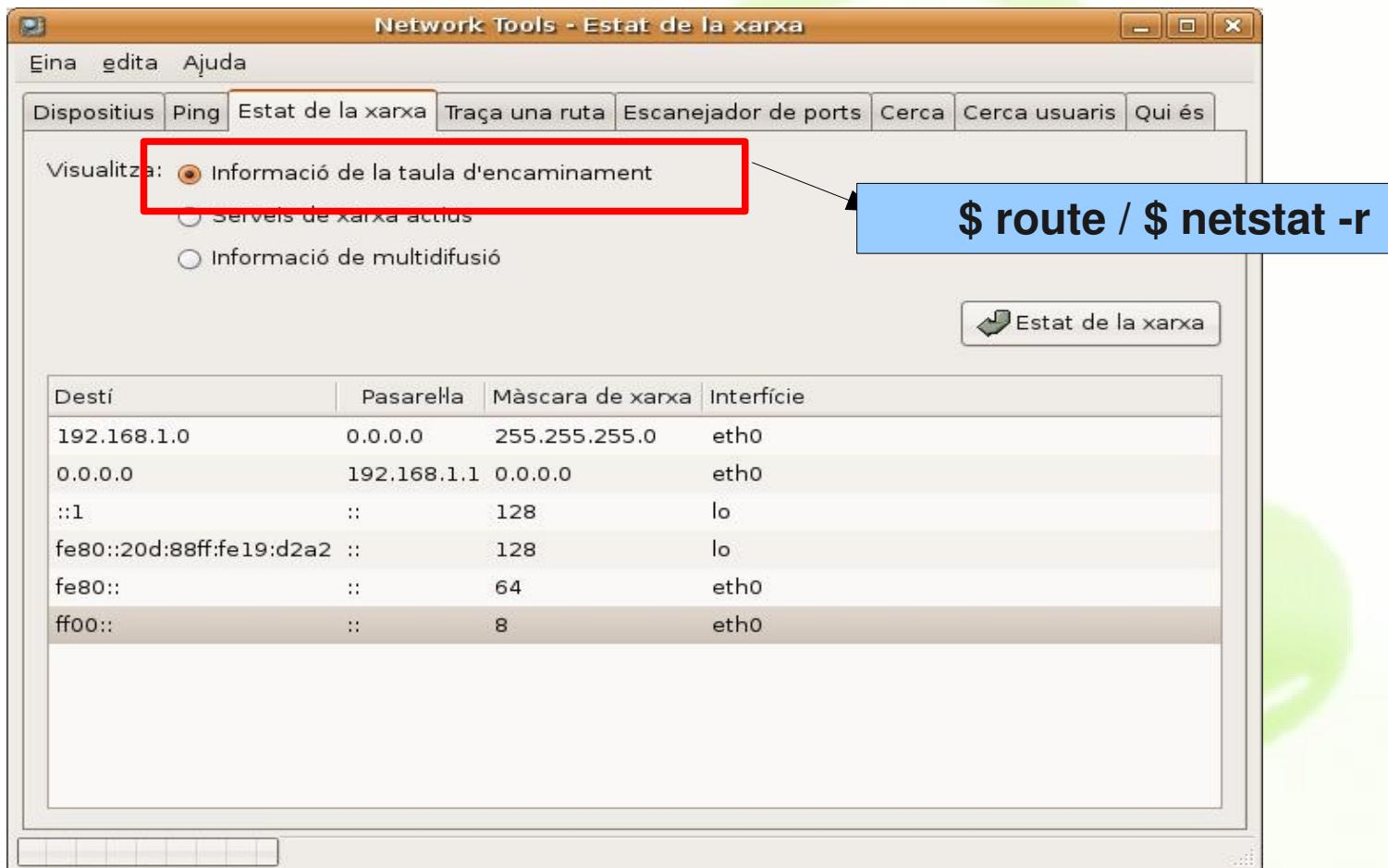
# Gnome-Nettool

## • Ping:



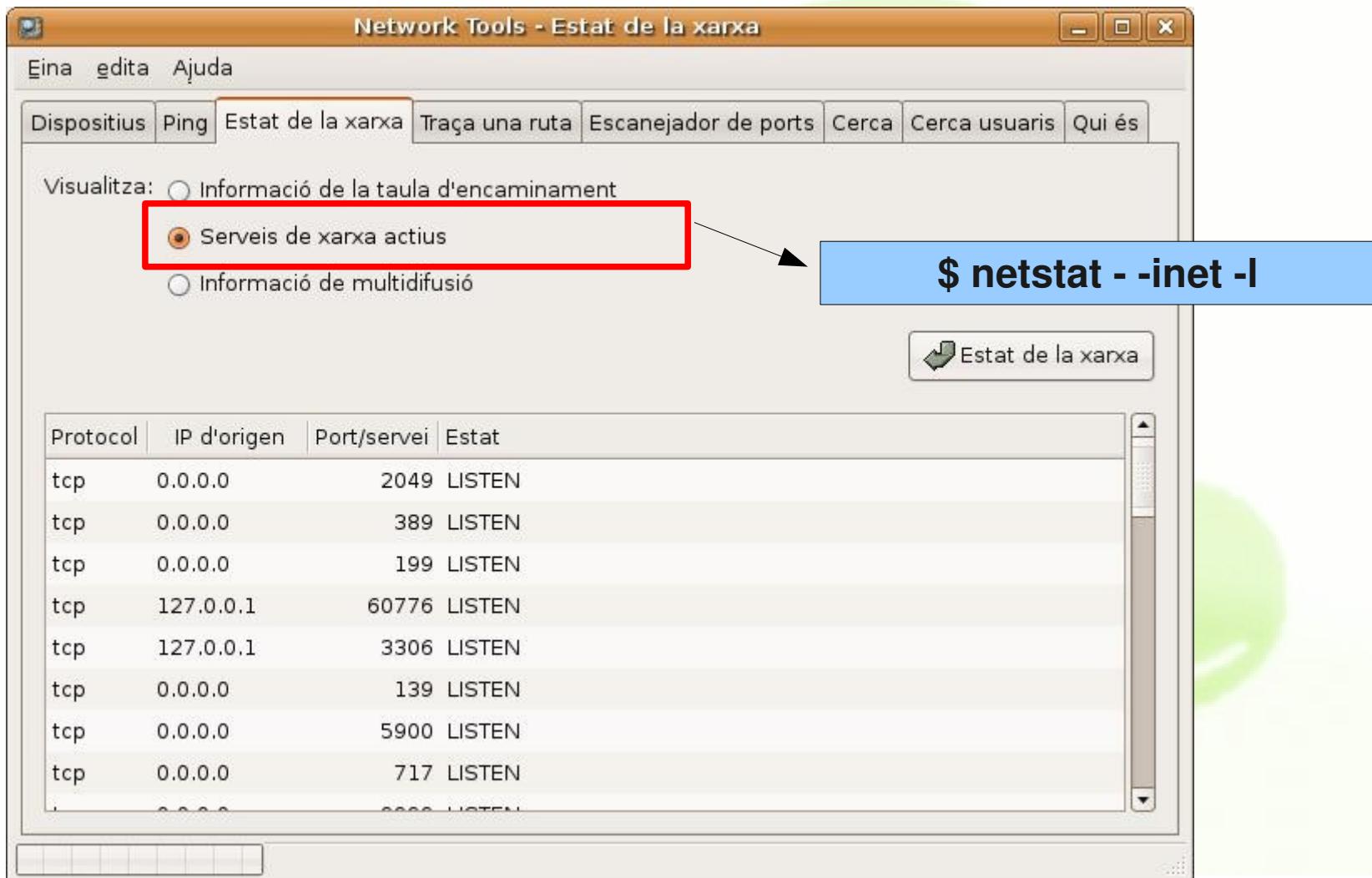
# Gnome-Neetool

## ◆ Encaminament:



# Gnome-Neetool

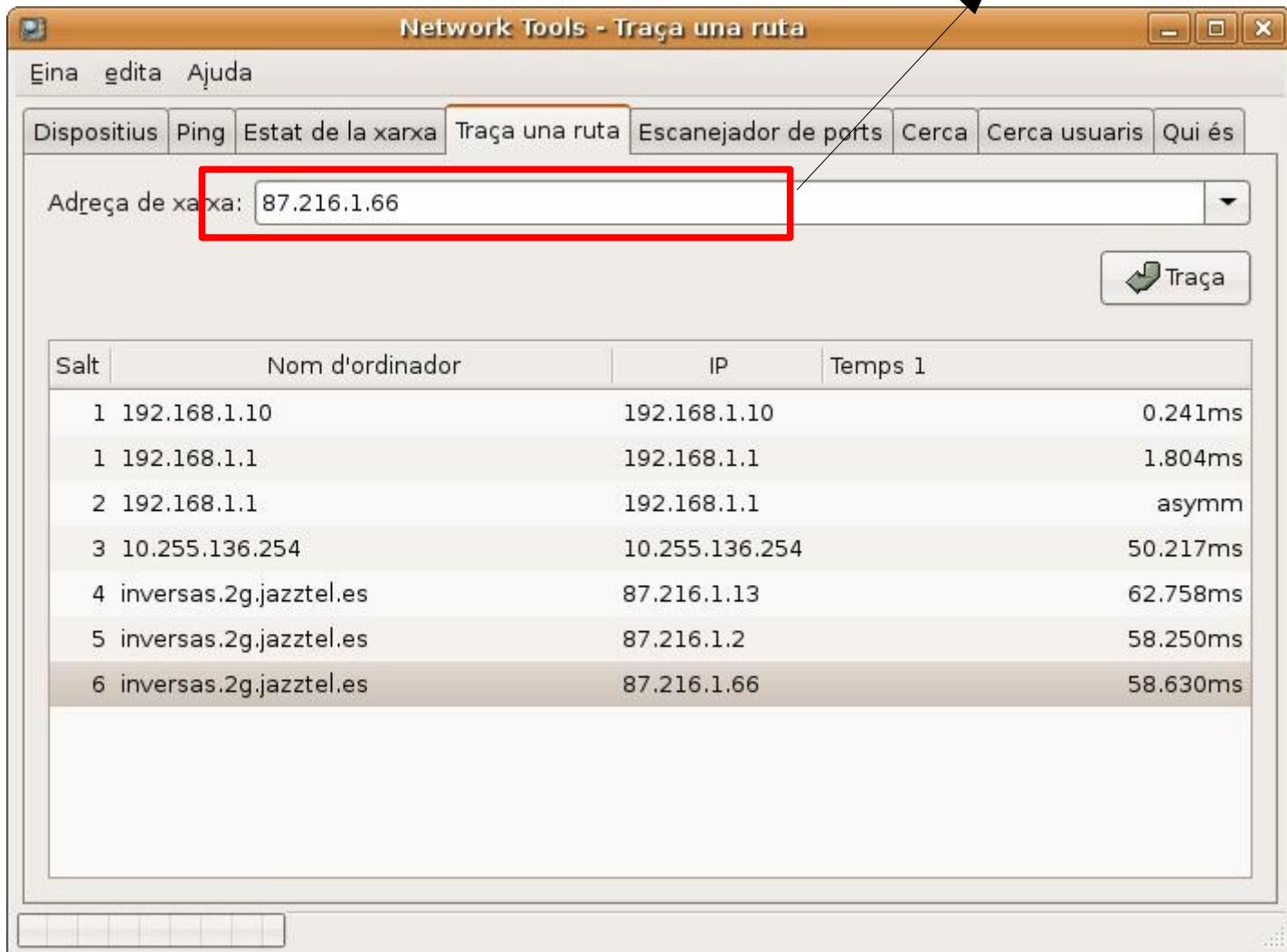
## • Estat de les connexions de xarxa:



# Gnome-Neetool

- Traçar una ruta:

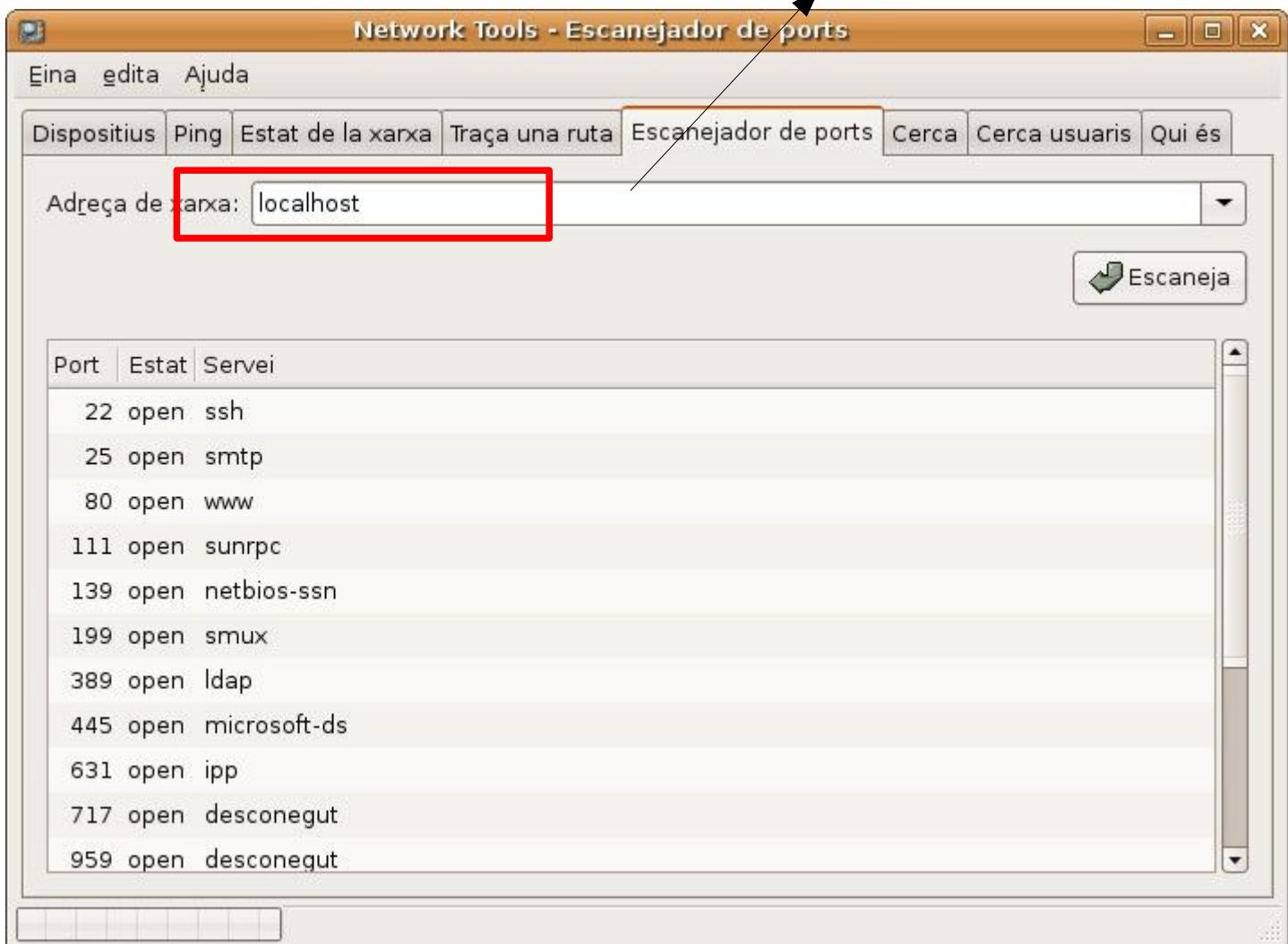
\$ traceroute 87.216.1.66



# Gnome-Neetool

## • Escànnner de ports:

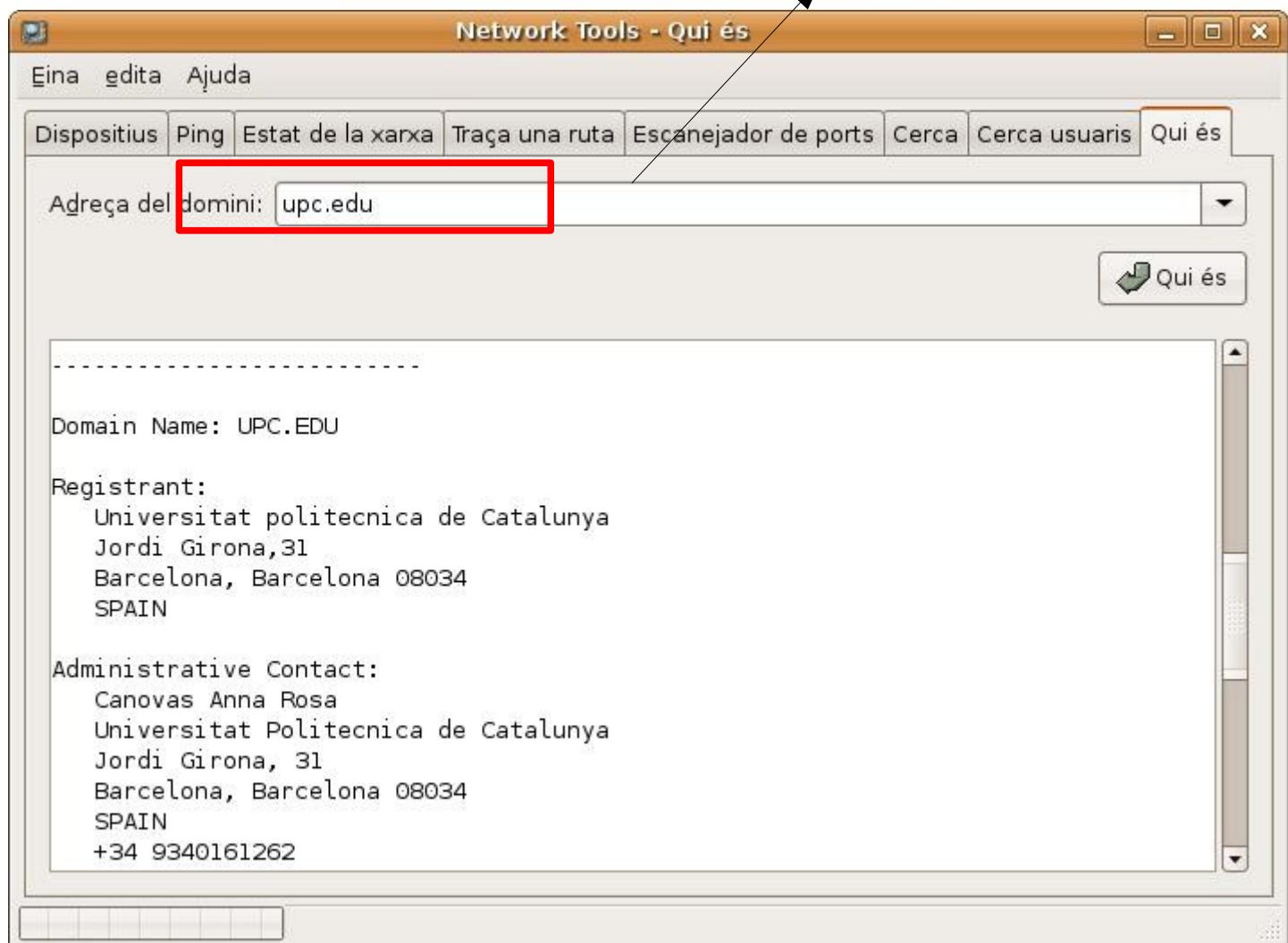
\$ nmap localhost



# Gnome-Neetool

## → Servei Whois:

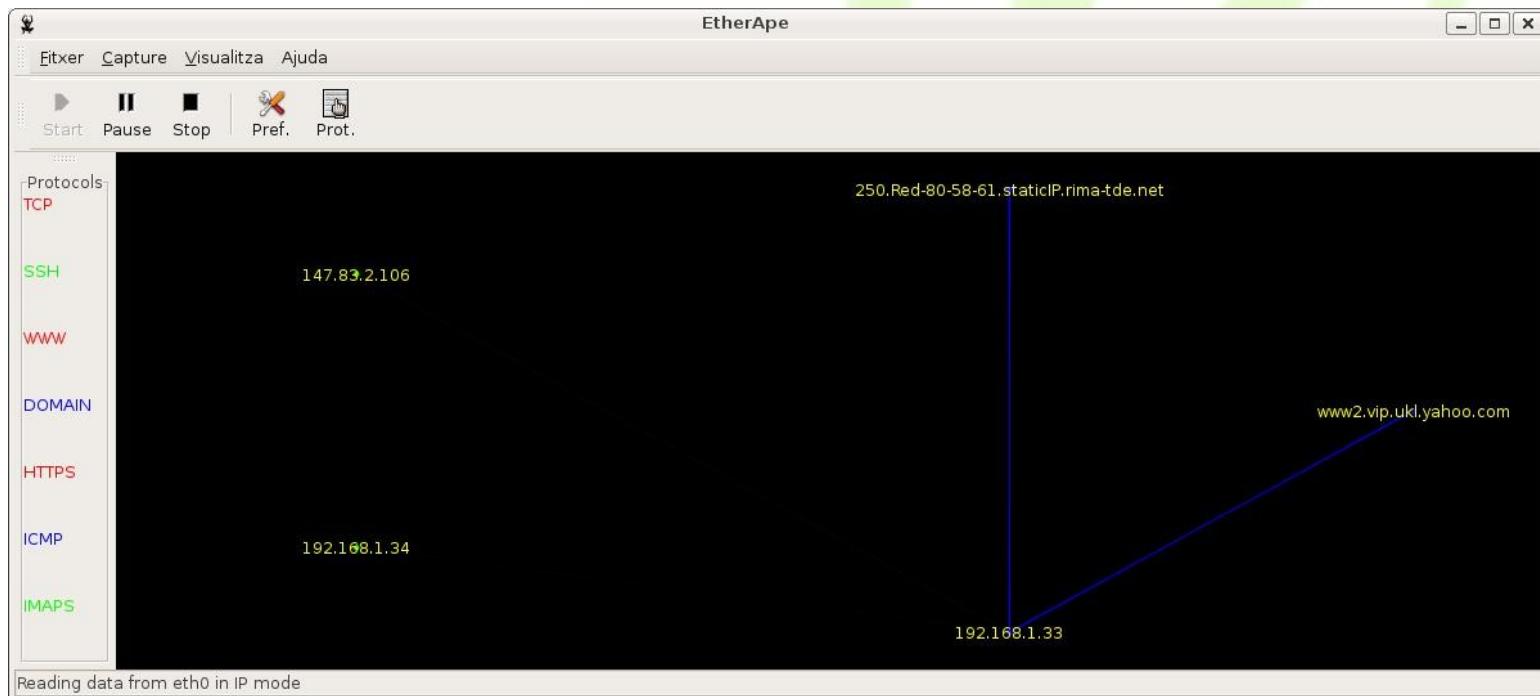
\$ whois upc.edu



# Etherape

## → Mostra gràficament les connexions de xarxa

- ◆ Diferència protocols per colors i amples de banda per amplituds de línia.
- ◆ Ús didàctic i de monitorització (netstat gràfic)



# EtherApe

- **Instal·lació:**

```
$ sudo apt-get install etherape
```

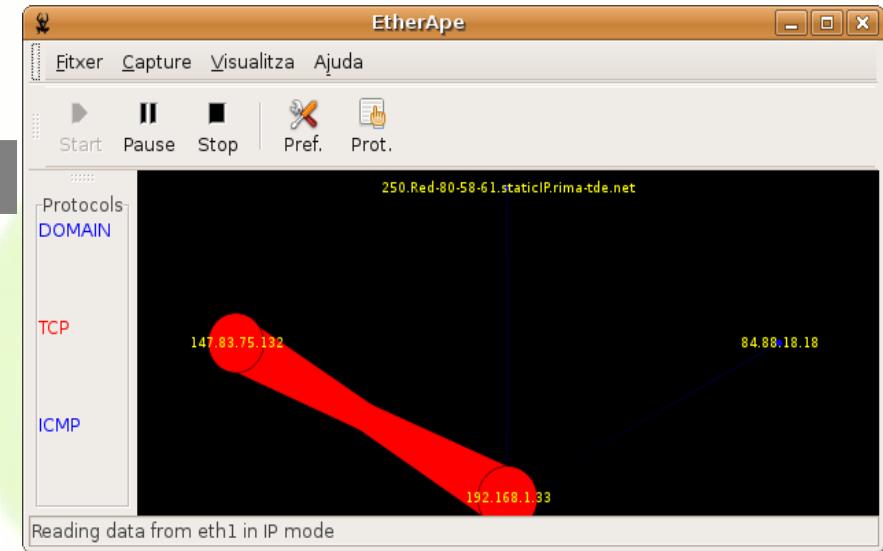
- **Com veure tota la xarxa**

```
$ sudo nmap 147.83.75.129-255
```

- **Com crear un raig potent:**

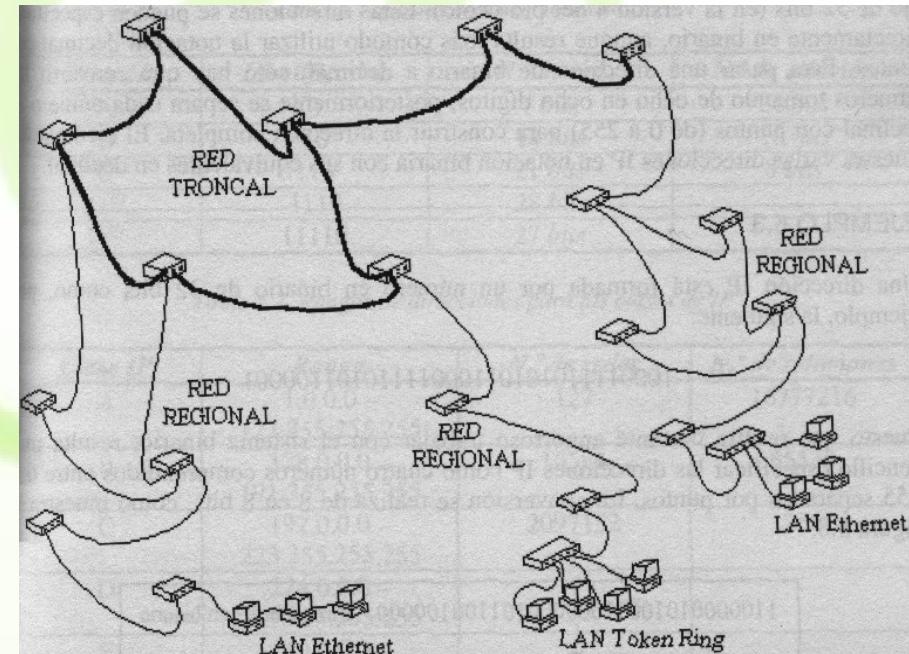
```
$ sudo nmap 147.83.75.132 -p 1-65535
```

```
$ vncviewer 147.83.75.132
```



# Subxarxes

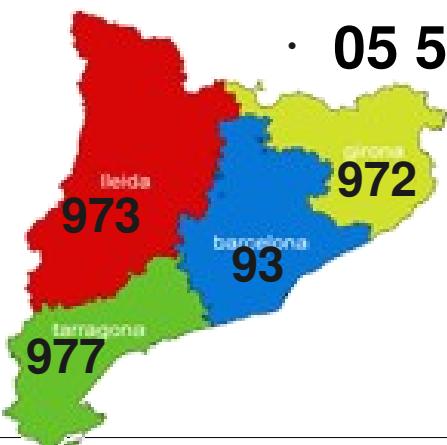
- ❖ **La xarxa (Internet) està formada per subxarxes.**
  - ❖ L'adreça de xarxa conjuntament amb la màscara de xarxa configuren les subxarxes.
- ❖ **Les subxarxes permeten aprofitar millor les IPS**
  - ❖ Recurs limitat
  - ❖ Millor organització jeràrquica.
- ❖ **Subxarxes a la wiki del curs**



# Subxarxes

## • PSTN (Public Switched Telephone Network)

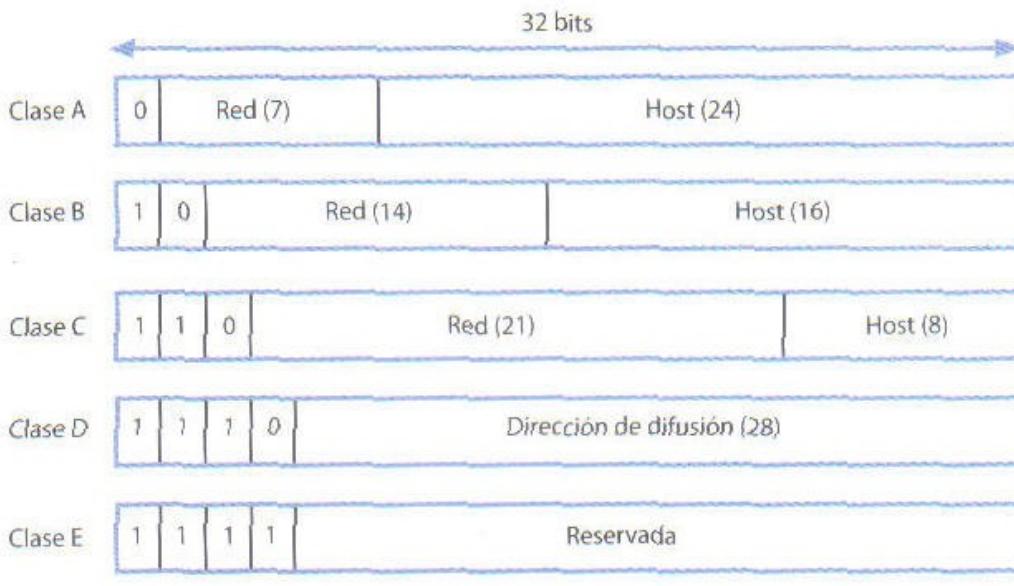
- La xarxa telefònica commutada (xarxa telefònica) també utilitza subxarxes
  - Nº Telèfon: **+34 93 894 05 50**
    - +34**: Codi de país (Espanya)
    - 93**: Codi de província (Barcelona)
    - 894**: Codi de ciutat/zona (Sitges)
    - 05 50**: Número de l'abonat



+34 93 894 05 50



# Subxarxes



N: Network

H: Host

0NNNNNNN.HHHHHHHH.HHHHHHHH.HHHHHHHH  
0NNNNNNN.HHHHHHHH.HHHHHHHH.HHHHHHHH

10NNNNNN.NNNNNNNN.HHHHHHHH.HHHHHHHH  
10NNNNNN.NNNNNNNN.HHHHHHHH.HHHHHHHH

110NNNNN.NNNNNNNN.NNNNNNNN.HHHHHHHH  
110NNNNN.NNNNNNNN.NNNNNNNN.HHHHHHHH

- La màscara determina quins bits estan reservats a la xarxa i quins bits a les màquines.
  - Depenent de les necessitats de xarxa (nombre de subxarxes i nombre de màquines per xarxa) s'escull la classe més adequada.



# Classes

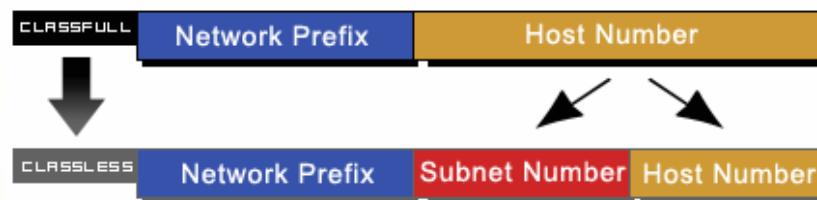
## ◆ **Classful Networks**

- ◆ Va aparèixer als anys 80 per poder classificar les xarxes en tres mides (classe A, B i C)
- ◆ **Classe A:** Poques xarxes grans
- ◆ **Classe B:** Força xarxes mitjanes
- ◆ **Classe C:** Moltes xarxes de 254 màquines
- ◆ **Classes especials (D i E):** Reservades per a usos especials

# Subxarxes. CIDR

## Classless Inter-Domain Routing. CIDR

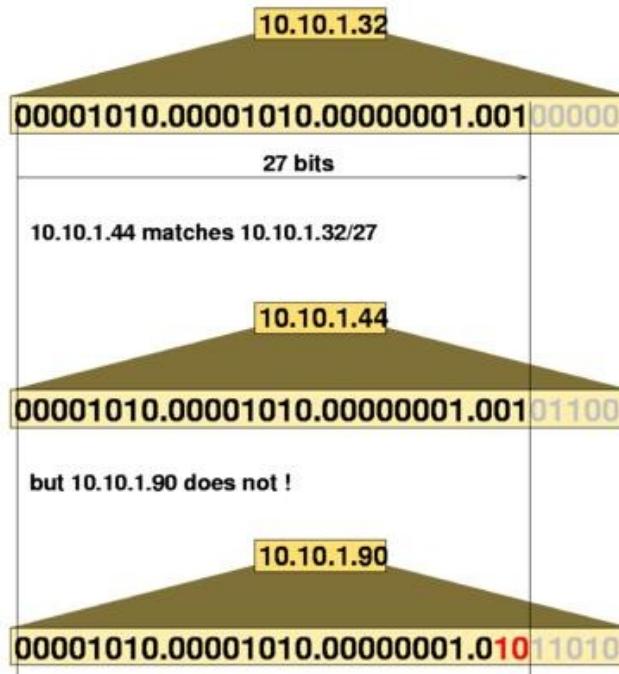
- ◆ Apareix als anys 90 per substituir el sistema de classes.
- ◆ Permet utilitzar bits d'host per a crear subxarxes:



- ◆ Ens permet obtenir més tipus de subxarxes que el sistema de classes
- ◆ La notació de les màscares amb barra (/24) també s'anomena notació CIDR.

# Subxarxes

- Per què s'utilitza aquest sistema?
  - Per que per a les màquines és molt fàcil fer càlculs de subxarxes
  - Càlcul molt fàcil (AND binari) per saber si



\$ ipcalc 192.168.5.130/26  
 Address: 192.168.5.130 11000000.10101000.00000101.10 000010  
 Netmask: 255.255.255.192=26 11111111.11111111.11111111.11 000000  
 Network: 192.168.5.128/26 11000000.10101000.00000101.10 000000

x	y	x AND y
0	0	0
0	1	0
1	0	0
1	1	1

	Dot-decimal Address	Binary
Full Network Address	192.168.5.130	11000000.10101000.00000101.10000010
Subnet Mask	255.255.255.192	11111111.11111111.11111111.11000000
Network Portion	192.168.5.128	11000000.10101000.00000101.10000000



# Subxarxes

## • Són de la mateixa xarxa (màscara 27) les IPs?:

- ◆ 10.10.1.44
- ◆ 10.10.1.90

```
$ ipcalc 10.10.1.44/27
Address: 10.10.1.44          00001010.00001010.00000001.001 01100
Netmask: 255.255.255.224=27 11111111.11111111.11111111.111 00000
Network: 10.10.1.32/27      00001010.00001010.00000001.001 00000
```

```
$ ipcalc 10.10.1.44/27
Address: 10.10.1.90          00001010.00001010.00000001.010 11010
Netmask: 255.255.255.224=27 11111111.11111111.11111111.111 00000
Network: 10.10.1.64/27      00001010.00001010.00000001.010 00000
```

```
$ ipcalc 192.168.201.50/27
Address: 192.168.201.50      11000000.10101000.11001001.00110010
Netmask: 255.255.255.224=27 11111111.11111111.11111111.11100000
Network: 192.168.201.32/27  11000000.10101000.11001001.00100000
```

## • Són de la mateixa xarxa (màscara 25) les Ips?:

- ◆ 192.168.201.50
- ◆ 192.168.201.220

```
$ ipcalc 192.168.201.220/27
Address: 192.168.201.220    11000000.10101000.11001001.11011100
Netmask: 255.255.255.224=27 11111111.11111111.11111111.11100000
Network: 192.168.201.192/27 11000000.10101000.11001001.11000000
```

# Subxarxes. Classes IP

## ♦ **RESUM: Classful Networks**

- ♦ Va aparèixer als anys 80 per poder classificar les xarxes en tres mides (classe A, B i C).

Class	Leading bits	Start	End	Default Subnet Mask In dotted decimal	CIDR notation
A	0	0.0.0.1	126.255.255.255	255.0.0.0	/8
B	10	128.0.0.0	191.255.255.255	255.255.0.0	/16
C	110	192.0.0.0	223.255.255.255	255.255.255.0	/24
D	1110	224.0.0.0	239.255.255.255		
E	1111	240.0.0.0	255.255.255.0		

## ♦ **3 màscares possibles, 3 possibilitats**

Class	Leading Value	Network Numbers	Addresses Per Network
Class A	0	126	16,777,216
Class B	10	16,384	65,534
Class C	110	2,097,152	254



# Exemple. 4 subxarxes classe C.

- **Xarxa classe C 192.168.0.1/24**
  - ◆ Cada bit d'host que agafem com a subxarxa ens permet multiplicar per dos les anteriors subxarxes que teníem.
- **Nova màscara 255.255.255.192/26**

```
sergi@casa-linux:~$ ipcalc 192.168.1.1/26
Address: 192.168.1.1          11000000.10101000.00000001.00 0000001
Netmask: 255.255.255.192 = 26 11111111.11111111.11111111.11 0000000
Wildcard: 0.0.0.63            00000000.00000000.00000000.00 1111111
=>
Network: 192.168.1.0/26      11000000.10101000.00000001.00 0000000
HostMin: 192.168.1.1          11000000.10101000.00000001.00 0000001
HostMax: 192.168.1.62         11000000.10101000.00000001.00 111110
Broadcast: 192.168.1.63        11000000.10101000.00000001.00 111111
Hosts/Net: 62                  Class C, Private Internet

sergi@casa-linux:~$ ipcalc 192.168.1.65/26
Address: 192.168.1.65          11000000.10101000.00000001.01 0000001
Netmask: 255.255.255.192 = 26 11111111.11111111.11111111.11 0000000
Wildcard: 0.0.0.63            00000000.00000000.00000000.00 1111111
=>
Network: 192.168.1.64/26      11000000.10101000.00000001.01 0000000
HostMin: 192.168.1.65          11000000.10101000.00000001.01 0000001
HostMax: 192.168.1.126         11000000.10101000.00000001.01 111110
Broadcast: 192.168.1.127        11000000.10101000.00000001.01 111111
Hosts/Net: 62                  Class C, Private Internet
```



# Exemple. 4 subxarxes classe C.

```
sergi@casa-linux:~$ ipcalc 192.168.1.129/26
Address: 192.168.1.129      11000000.10101000.00000001.10 000001
Netmask: 255.255.255.192 = 26 11111111.11111111.11111111.11 000000
Wildcard: 0.0.0.63          00000000.00000000.00000000.00 111111
=>
Network: 192.168.1.128/26   11000000.10101000.00000001.10 000000
HostMin: 192.168.1.129      11000000.10101000.00000001.10 000001
HostMax: 192.168.1.190      11000000.10101000.00000001.10 111110
Broadcast: 192.168.1.191    11000000.10101000.00000001.10 111111
Hosts/Net: 62                Class C, Private Internet

sergi@casa-linux:~$ ipcalc 192.168.1.200/26
Address: 192.168.1.200      11000000.10101000.00000001.11 001000
Netmask: 255.255.255.192 = 26 11111111.11111111.11111111.11 000000
Wildcard: 0.0.0.63          00000000.00000000.00000000.00 111111
=>
Network: 192.168.1.192/26   11000000.10101000.00000001.11 000000
HostMin: 192.168.1.193      11000000.10101000.00000001.11 000001
HostMax: 192.168.1.254      11000000.10101000.00000001.11 111110
Broadcast: 192.168.1.255    11000000.10101000.00000001.11 111111
Hosts/Net: 62                Class C, Private Internet
```

- ◆ **Algunes adreces no es poden utilitzar**
  - ◆ **Xarxa:** 192.168.1.0 | 192.168.1.64 | 192.168.1.128 | 192.168.1.192
  - ◆ **Broadcast:** 192.168.1.63 | 192.168.1.127 | 192.168.1.191 | 192.168.1.255

# Adreces IP reservades

## ♦ Adreça de xarxa

### ◆ Quina adreça té una xarxa?

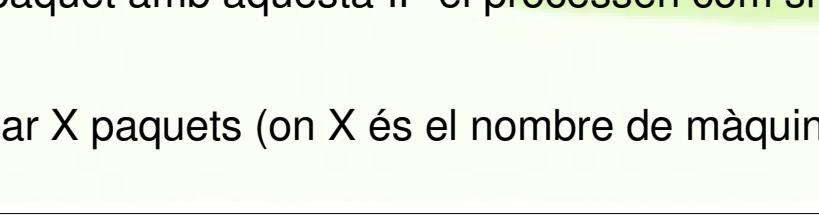
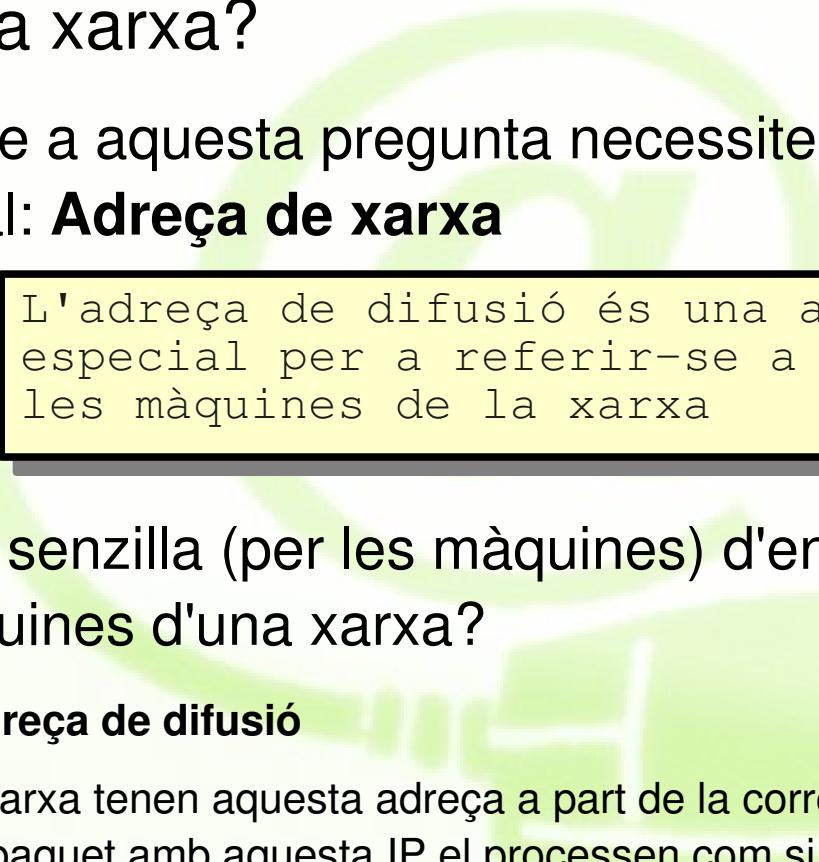
- Per poder respondre a aquesta pregunta necessitem reservar una adreça especial: **Adreça de xarxa**

La primera adreça del rang d'adreces determinat per la màscara de xarxa és l'adreça de la xarxa

## ♦ Adreça de difusió

### ◆ Quina és la forma més senzilla (per les màquines) d'enviar un paquet a totes les màquines d'una xarxa?

- La resposta és utilitzar **l'adreça de difusió**
- Totes les màquines de la xarxa tenen aquesta adreça a part de la corresponent adreça IP (quan reben un paquet amb aquesta IP el processen com si tingues la seva IP)
- Molt més còmode que enviar X paquets (on X és el nombre de màquines de la xarxa)



# IPs reservades

- Hi ha altres adreces IP reservades

Addresses	CIDR Equivalent	Purpose	RFC	Class	Total # of addresses
0.0.0.0 - 0.255.255.255	0.0.0.0/8	Zero Addresses	RFC 1700	A	16,777,216
10.0.0.0 - 10.255.255.255	10.0.0.0/8	Private IP addresses	RFC 1918	A	16,777,216
127.0.0.0 - 127.255.255.255	127.0.0.0/8	Localhost Loopback Address	RFC 1700	A	16,777,216
169.254.0.0 - 169.254.255.255	169.254.0.0/16	Zeroconf / APIPA	RFC 3330	B	65,536
172.16.0.0 - 172.31.255.255	172.16.0.0/12	Private IP addresses	RFC 1918	B	1,048,576
192.0.2.0 - 192.0.2.255	192.0.2.0/24	Documentation and Examples	RFC 3330	C	256
192.88.99.0 - 192.88.99.255	192.88.99.0/24	IPv6 to IPv4 relay Anycast	RFC 3068	C	256
192.168.0.0 - 192.168.255.255	192.168.0.0/16	Private IP addresses	RFC 1918	C	65,536
198.18.0.0 - 198.19.255.255	198.18.0.0/15	Network Device Benchmark	RFC 2544	C	131,072
224.0.0.0 - 239.255.255.255	224.0.0.0/4	Multicast	RFC 3171	D	268,435,456
240.0.0.0 - 255.255.255.255	240.0.0.0/4	Reserved	RFC 1700	E	268,435,456

Network address range	CIDR notation
10.0.0.0 - 10.255.255.255	/8
172.16.0.0 - 172.31.255.255	/12
192.168.0.0 - 192.168.255.255	/16

- Xarxes privades



# Comanda ifconfig

- **Consulta i configuració de paràmetres de xarxa**

- ◆ **ifconfig**

- **Adreça IP.** Adreça lògica del protocol IP. Nivell de xarxa (Nivell 3 OSI). Protocol IP.
    - **Adreça MAC.** Adreça física. Assignada a la NIC. Nivell MAC (Nivell 2 OSI). Protocol ARP.
    - **Màscara de xarxa.** Determina quina part de l'adreça correspon a la xarxa i quina a les màquines de la xarxa.
    - **Adreça de xarxa.** Ve determinada per la màscara i és l'adreça que té els bits corresponents a adreces de màquines a 0.
    - **Adreça de difusió (broadcast).** Ve determinada per la màscara i és l'adreça que té l'últim octet establert a 255.



# ifup/ifdown

- **Són les encarregades d'activar/desactivar les interfícies de xarxa segons els paràmetres dels fitxers de configuració.**
  - ◆ El sistema operatiu s'encarrega de cridar aquestes comandes a l'iniciar l'ordinador.

```
$ sudo ifup eth0
...
Listening on LPF/eth0/00:0d:88:19:d2:a2
Sending on  LPF/eth0/00:0d:88:19:d2:a2
Sending on  Socket/fallback
DHCPREQUEST on eth0 to 255.255.255.255 port 67
DHCPACK from 192.168.1.1
bound to 192.168.1.14 -- renewal in 244026
seconds.

$ sudo ifdown eth0
...
Listening on LPF/eth0/00:0d:88:19:d2:a2
Sending on  LPF/eth0/00:0d:88:19:d2:a2
Sending on  Socket/fallback
DHCPRELEASE on eth0 to 192.168.1.1 port 67
```

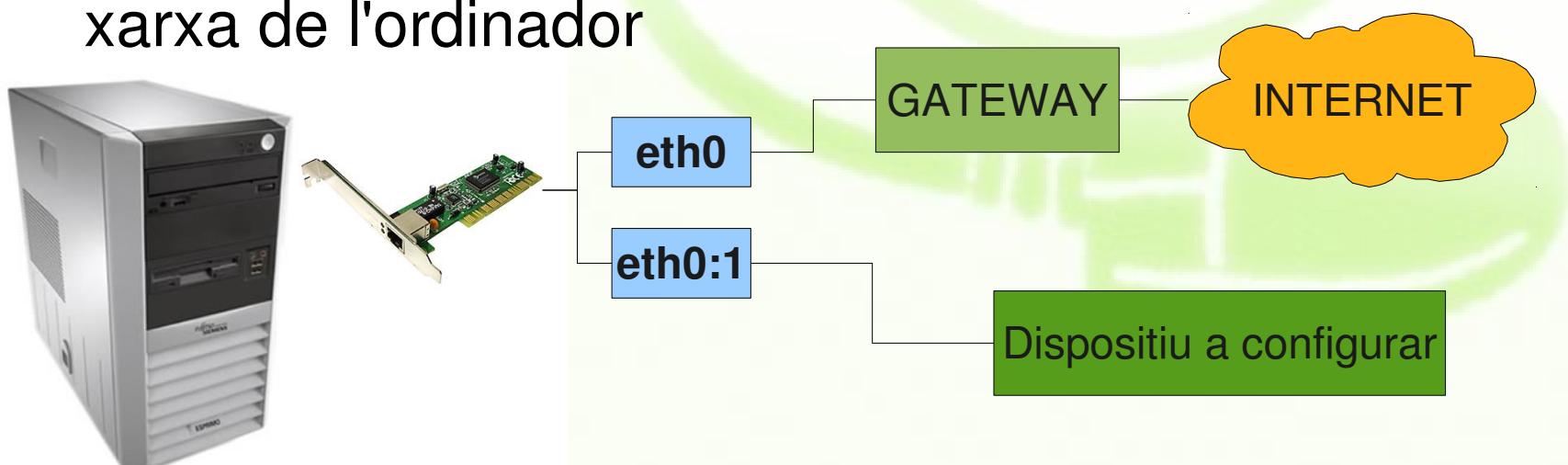
# Inicialització de la xarxa a Debian

## • S'utilitzen scripts d'inicialització SystemV

- ◆ La xarxa s'executa al nivell S (Single User Mode) (primer de tots, fins i tot abans del nivell 1)
- ◆ Els Links:
  - /etc/rcS.d/S08loopback
  - /etc/rcS.d/S40networking
- ◆ Executen els scripts d'inicialització
  - /etc/init.d/loopback start
  - /etc/init.d/networking start
- ◆ Upstart:
  - /etc/init/networking.conf

# IP ALIASING

- Amb una sola NIC (targeta de xarxa) podem tenir més d'una interfície de xarxa
  - ◆ Permet tenir configurada una targeta de xarxa per a múltiples IP i xarxes.
  - ◆ Ideal per a configurar dispositius de xarxa. Permet accedir al dispositiu, sense perdre la configuració de xarxa de l'ordinador



# IPALIASING

- Per configurar els dispositius cal estar a la mateixa xarxa
  - El dispositiu no pot canviar de xarxa! L'ordinador ha de canviar
  - INCORRECTE:**



IP: 192.168.0.1  
Màscara: 24

XARXA: 192.168.0.0/24

No hi ha  
comunicació

IP: 192.168.201.5  
Màscara: 24

XARXA AULA 1:  
192.168.201.0/24



- CORRECTE:**



IP: 192.168.0.1  
Màscara: 24

XARXA: 192.168.0.0/24

OK

IP: 192.168.0.5  
Màscara: 24

XARXA DISPOSITIU:  
192.168.0.0/24





# IP ALIASING

- La interfície “virtual” s’ha de configurar amb
  - ◆ IP diferent a la del dispositiu però que estigui a la mateixa xarxa (podeu utilitzar ipcalc per obtenir un rang d’adreses IP vàlides)
  - ◆ Mateixa xarxa que el dispositiu (mateixa màscara)

Dispositiu a configurar  
IP: 192.168.0.1  
Màscara: 255.255.255.0

```
$ ipcalc  
192.168.0.1/24  
.....  
Network:  
192.168.0.0/24  
HostMin:  
192.168.0.1  
HostMax:  
192.168.0.254  
.....
```

```
sudo ifconfig eth0:1 192.168.0.5 netmask  
255.255.255.0 up
```

```
$ ifconfig  
eth0 Link encap:Ethernet HWaddr 00:30:1B:B7:CD:B6  
inet addr:192.168.1.2 Bcast:192.168.1.255 Mask:255.255.255.0  
inet6 addr: fe80::230:1bff:feb7:cdb6/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:71667 errors:0 dropped:0 overruns:0 frame:0  
TX packets:52003 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:73735124 (70.3 MB) TX bytes:6732597 (6.4 MB)  
  
eth0:1 Link encap:Ethernet HWaddr 00:30:1B:B7:CD:B6  
inet addr:192.168.0.5 Bcast:192.168.0.255 Mask:255.255.255.0  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
Interrupt:20
```



# IP ALIASING

- ◆ **Es configura una nova ruta local:**

```
$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.1.0     *              255.255.255.0   U      0      0        0 eth0
192.168.0.0     *              255.255.255.0   U      0      0        0 eth0
link-local       *              255.255.0.0     U      1000   0        0 eth0
default          gateway.casa  0.0.0.0         UG     100    0        0 eth0
```

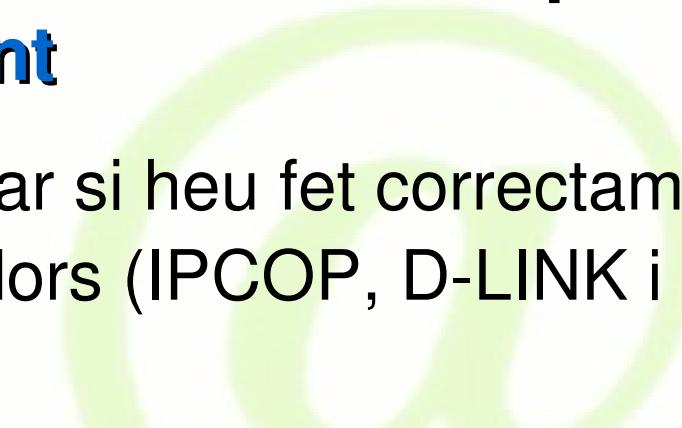
- ◆ Les rutes i la passarel·la antigues es mantenen
- ◆ Ja podeu comunicar-vos amb el dispositiu (proveu de fer un ping)
- ◆ **Desfer canvis IP ALIASING**

```
sudo ifconfig eth0:1 192.168.0.5 netmask
255.255.255.0 down
```



# Comanda traceroute

- ♦ Aquesta comanda us permetrà saber quina Pasarel·la esteu utilitzant
  - ♦ Us servirà per comprovar si heu fet correctament les pràctiques d'encaminadors (IPCOP, D-LINK i router Cisco)

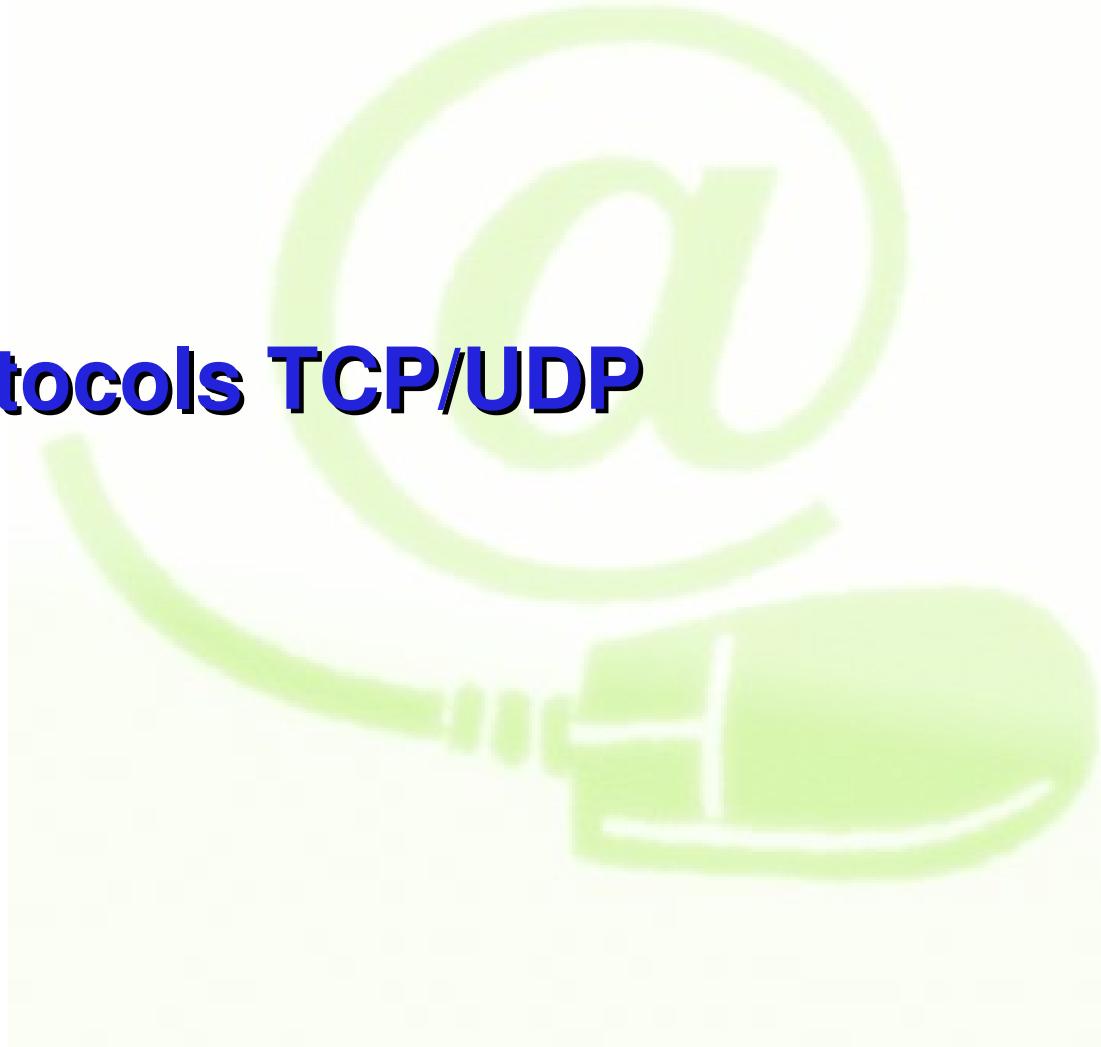


```
$ sudo traceroute www.upc.edu
traceroute to www.upc.edu (147.83.194.21), 30 hops max, 40 byte packets
 1 gateway.casa (192.168.1.1)  0.752 ms  1.028 ms  1.319 ms
 2 1.198.219.87.dynamic.jazztel.es (87.219.198.1)  34.337 ms  34.417 ms  35.291 ms
 3 * * *
 4 162.216.106.212.static.jazztel.es (212.106.216.162)  40.810 ms  178.216.106.212.static.jazztel.es
(212.106.216.178)  42.508 ms  162.216.106.212.static.jazztel.es (212.106.216.162)  44.324 ms
 5 * * *
 6 * * *
 7 rediris-2.espanix.net (193.149.1.154)  59.089 ms  41.301 ms  42.646 ms
 8 EXP.SO2-1-0.EB-IRIS2.red.rediris.es (130.206.250.13)  120.591 ms  119.406 ms  121.715 ms
 9 NAC.XE0-1-0.EB-Barcelona0.red.rediris.es (130.206.250.26)  140.802 ms  142.380 ms  133.971 ms
10 cesca-ge.red.rediris.es (130.206.202.2)  57.517 ms  57.871 ms  57.067 ms
11 upc-anella.cesca.es (84.88.18.18)  132.862 ms  132.530 ms  133.845 ms
12 * * *
13 * * *
14 * * *
15 cerberuso2.upc.es (147.83.172.181)  137.137 ms  137.418 ms  139.317 ms
16 raiden.upc.es (147.83.194.21)  140.729 ms  140.562 ms  143.700 ms
```



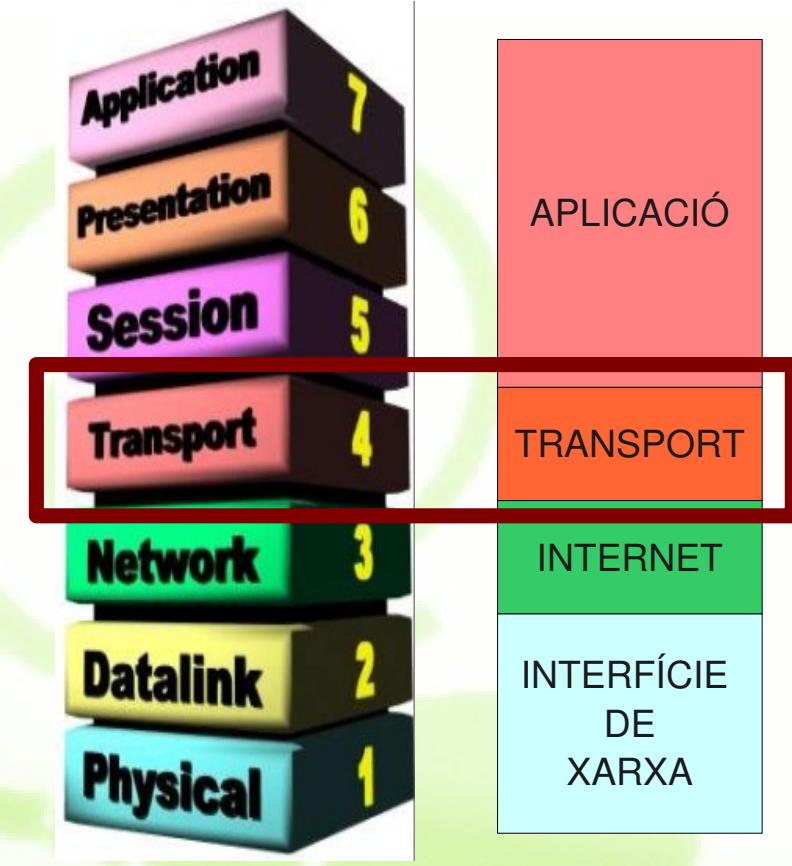
# Arquitectures de comunicacions

## Protocols TCP/UDP



# Nivell de transport (nivell 3 TCP/IP - nivell 4 OSI)

El nivell de transport és l'encarregat de que les dades transferides entre emissor i receptor estiguin lliures d'errors. També és l'encarregat de controlar el flux de dades.



- ❖ **Model de referència OSI**
  - ❖ Nivell 4. Nivell de transport
- ❖ **Pila de protocols TCP/IP**
  - ❖ Nivell 3. Nivell de transport
  - ❖ Els objectius són els mateixos que el nivell d'enllaç però aquest com **la comunicació és entre màquines que no estan directament connectades**.

# Nivell 4. Nivell de transport

## ◆ Capa de transició que connecta les aplicacions i/o usuaris amb la xarxa

- ◆ És una capa de transició entre els nivells orientats a la xarxa i els orientats a les aplicacions
- ◆ Treballa amb unitats de dades 4-PDU també anomenades TPDU o **segments**.
- ◆ Té funcions similars al nivell d'enllaç (salt a salt) però entre dues màquines que no estan connectades directament (extrem a extrem)
- ◆ S'encarrega de preparar les dades de les aplicacions per a la xarxa i assegurar-se que arribaran correctament al nivell de transport del destinatari.
- ◆ **Protocols:** TCP (Transport Control Protocol) i UDP (User Datagram Protocol)



# Funcions del nivell de xarxa

## • **Establiment de connexió**

- ◆ **Opcional.** Només s'aplica al serveis orientats a connexió
- ◆ S'estableix un camí virtual a través de qual es durà a terme la comunicació. Les dades s'envien de forma ordenada per aquest camí

## • **Reordenació de paquets**

- ◆ **Opcional.** Només s'aplica al serveis **no** orientats a connexió. No s'estableix cap camí i els paquets poden arribar desordenats

## • **Control d'errors**

- ◆ Les capçaleres (*headers*) del nivell 4 contenen dades redundants que permeten detectar errors en la transmissió
- ◆ Recuperació de caigudes de xarxa, reenviament de paquets, etc.

# Funcions del nivell de xarxa

- ❖ **Control de flux: Implementació de buffers.**
  - ❖ S'utilitzen memòries intermediàries (buffers o cache) que permeten controlar el flux.
  - ❖ Si un servidor esta massa ocupat els paquets que arriben es guarden a una memòria o cua a l'espera que el servidor pugui processar-los.
- ❖ **Qualitat de servei. QoS (Quality of Service)**
  - ❖ Garanteix la fiabilitat i la qualitat del servei. Per exemple es pot reservar un ampla de banda mínim per a una connexió concreta
- ❖ **Multiplexació de connexions**
  - ❖ Permet tenir més d'una connexió oberta a través d'un mateix medi físic. S'utilitzen **ports** i el concepte de **sockets**.



# Serveis OSI

## • **Servei orientat a connexió**

- ◆ Abans d'intercanviar dades es necessita establir una connexió (Exemples: telèfon, accés a una pàgina web, protocol TCP)

## • **Servei no orientat a connexió**

- ◆ Les dades s'envien directament sense establir cap connexió prèvia (enviar una carta, enviar un email, protocol UDP)

## • **Servei confirmat o no confirmat**

- ◆ **Servei confirmat:** Trucada telefònica (pot ser confirmada (et despengen el telèfon) o no confirmada (no et despengen, et denegen la trucada o comunica))
- ◆ **Servei no confirmat:** Per exemple al parlar per telèfon. És una comunicació full duplex i ni emissor ni receptor necessiten de confirmació per començar a parlar.

# Serveis orientats a connexió

## ◆ Propietats

- ◆ Requereixen el establiment inicial de una connexió i la ruptura o alliberament final de la mateixa.
- ◆ Entre la connexió i l'alliberament es produeix l'intercanvi de dades d'usuari.
- ◆ Els blocs de dades es reben en el destí en el mateix ordre en que s'emeten a l'origen.
- ◆ Tots els paquets segueixen la mateixa ruta, aconseguida en l'establiment de la connexió
- ◆ Com que la ruta es coneguda, els paquets de dades no precisen indicar l'adreça de destinació.
- ◆ **Exemple:** Trucada telefònica

# Protocols orientats a connexió. TCP

- ❖ **Orientades a connexió (Connection-oriented)**
  - ❖ Els dispositius de cada banda de la connexió (emissor i receptor) utilitzen un protocol preliminar a l'enviament de dades per establir una connexió punta a punta.
  - ❖ Sovint també s'anomenen serveis de xarxa fiables (**reliable**) per què es garanteix que les dades arribaren en l'ordre adequat.
  - ❖ La comunicació pot estar en diferents **estats**
- ❖ **La comunicació es duu a terme en tres fases:**
  - ❖ **Fase 1:** Establiment de la connexió (**handshake**)
  - ❖ **Fase 2:** Transmissió de dades
  - ❖ **Fase 3:** Tancament de la connexió

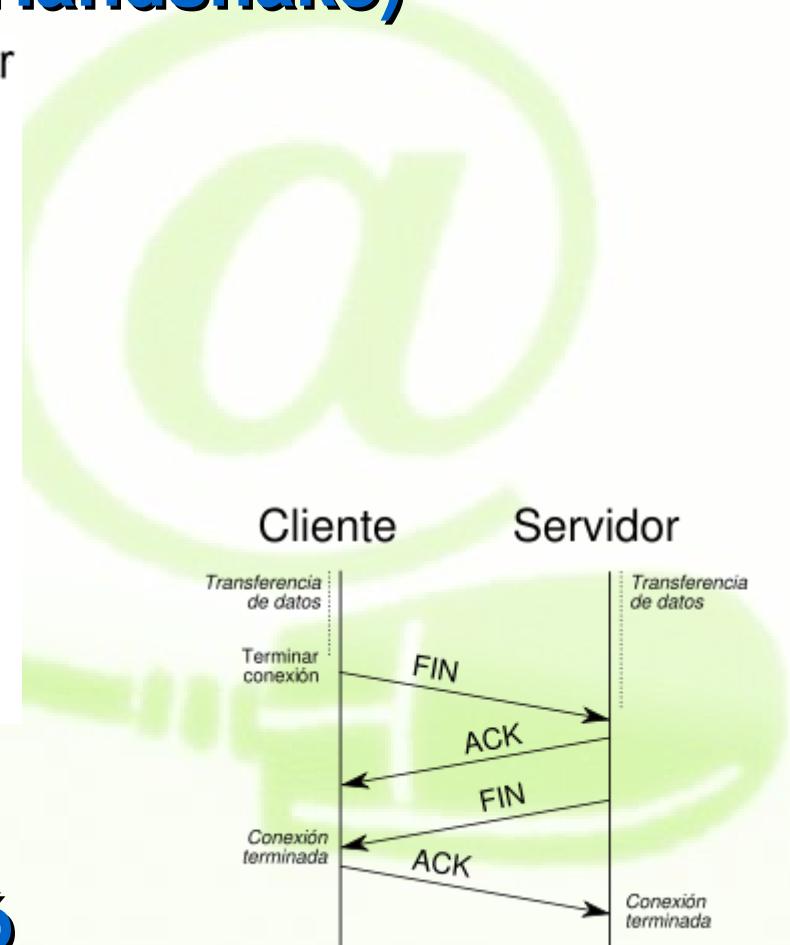
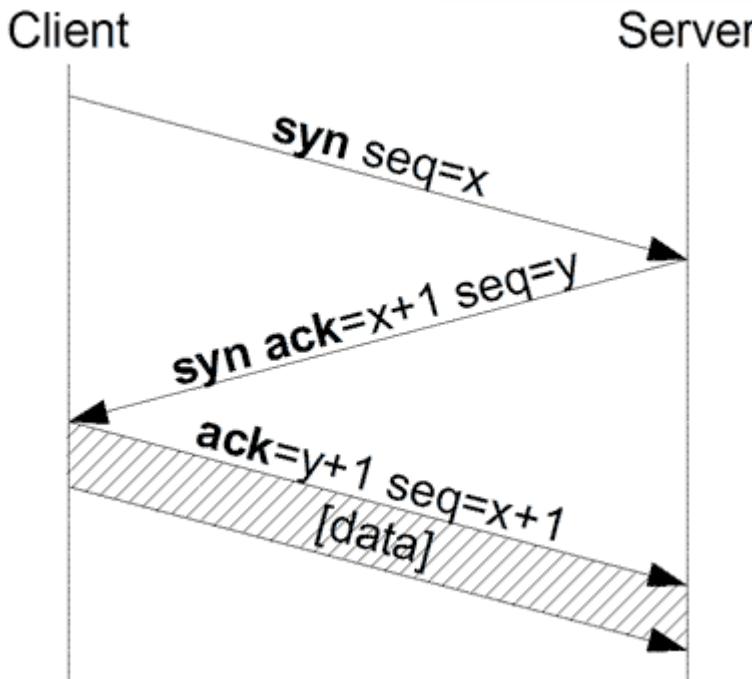
# Serveis orientats a connexió

- ◆ **Tenen dues variants:**

- ◆ **Seqüència de missatges.** S'estableixen fronteres que defineixen i determinen cada missatge (és equivalent a la sincronització).
- ◆ **Seqüència de bytes.** En aquests serveis no hi ha contorns entre els missatges. Cada missatge és una seqüència de caràcters, deixant al receptor la responsabilitat de la seva interpretació.

# Protocols orientats a connexió. TCP

- Encaixada de mans TCP (Handshake)



- Acabament de la connexió



# Estats de la connexió

## ◆ Estats

- ◆ En els protocols orientats a connexió, la connexió passa per diferents estats.

## ◆ Pràctica: Comanda netstat.

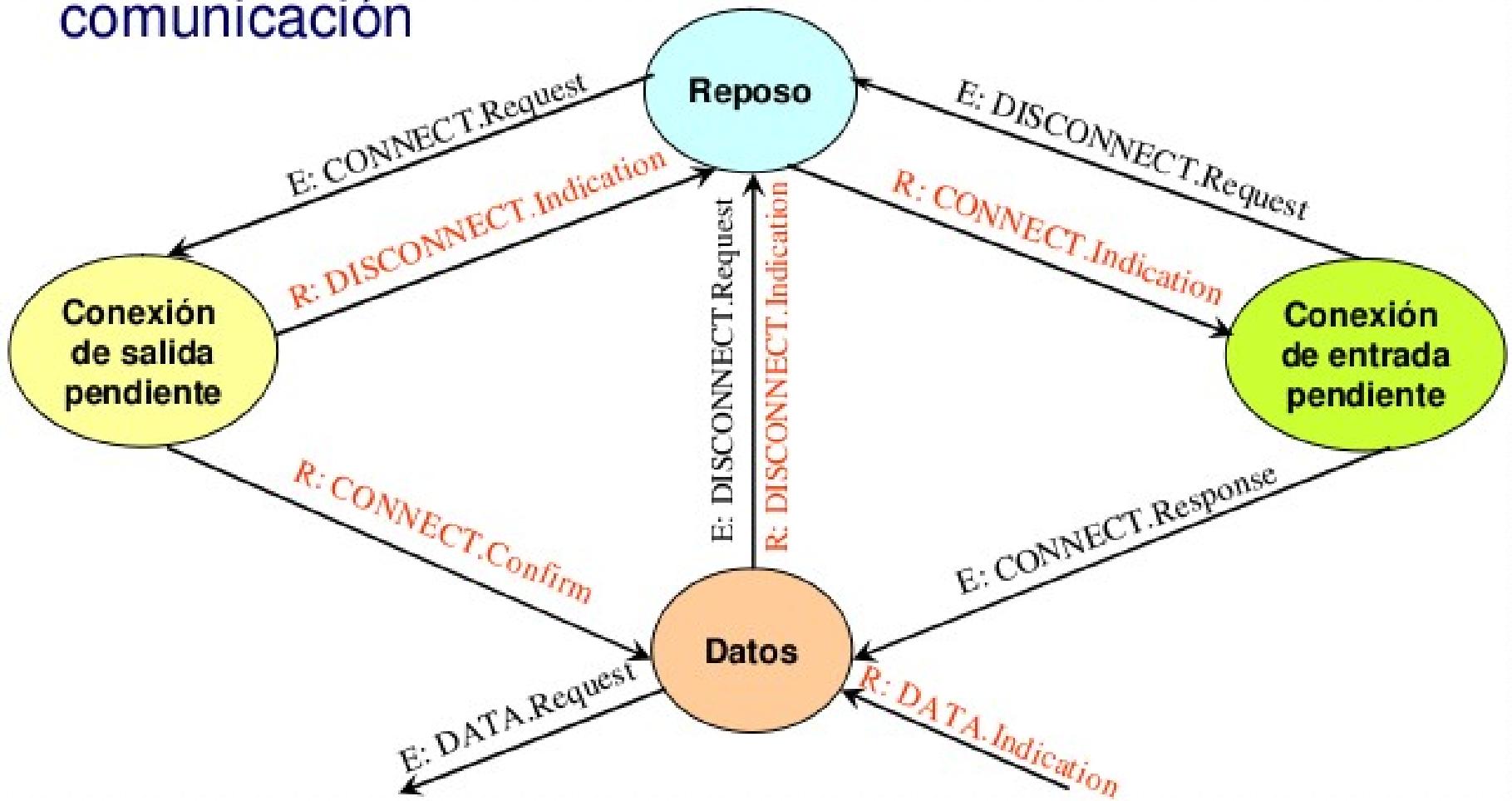
- ◆ Executeu:

```
$ netstat --inet -t -a -c | grep upc.es:www
tcp 1 0 casa-linux.local:48520 raiden.upc.es:www CLOSE_WAIT
tcp 0 687 casa-linux.local:48529 raiden.upc.es:www ESTABLISHED
tcp 0 1 casa-linux.local:48522 raiden.upc.es:www SYN_SENT
tcp 0 709 casa-linux.local:48537 raiden.upc.es:www ESTABLISHED
tcp 0 1 casa-linux.local:48522 raiden.upc.es:www SYN_SENT
tcp 1 1 casa-linux.local:48560 raiden.upc.es:www LAST_ACK
tcp 0 0 casa-linux.local:48556 raiden.upc.es:www TIME_WAIT
tcp 0 0 casa-linux.local:48556 raiden.upc.es:www TIME_WAIT
.....
```

- ◆ I seguidament connecteu-vos a la web de la UPC
- ◆ Comanda netstat a la wiki del curs

# Estats de la connexió

comunicación





# Estats de la connexió

## ◆ Estats

- ◆ En els protocols orientats a connexió, la connexió passa per diferents estats.

## ◆ Pràctica: Comanda netstat.

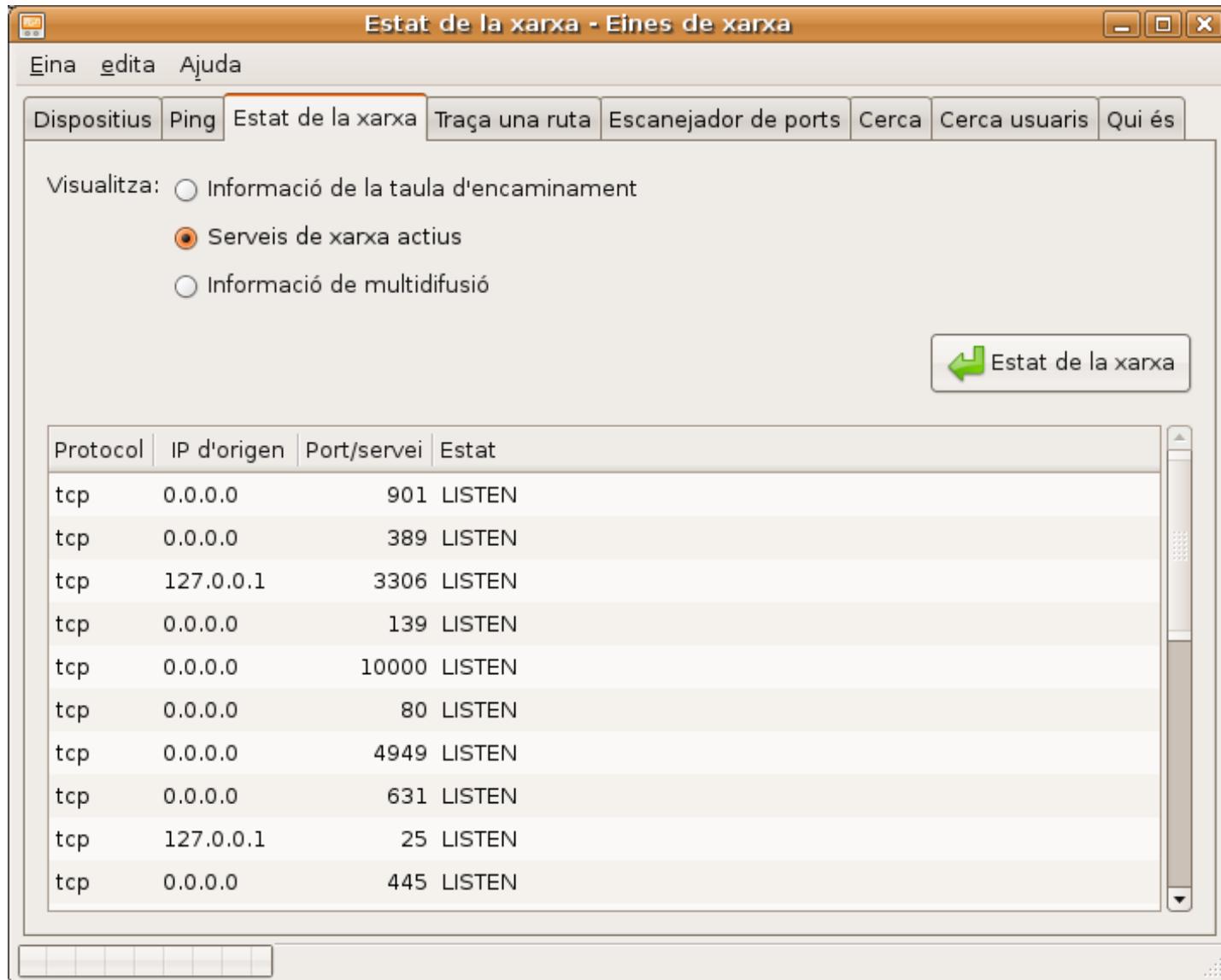
- ◆ Executeu:

```
$ netstat --inet -t -a -c | grep upc.es:www
tcp 1 0 casa-linux.local:48520 raiden.upc.es:www CLOSE_WAIT
tcp 0 687 casa-linux.local:48529 raiden.upc.es:www ESTABLISHED
tcp 0 1 casa-linux.local:48522 raiden.upc.es:www SYN_SENT
tcp 0 709 casa-linux.local:48537 raiden.upc.es:www ESTABLISHED
tcp 0 1 casa-linux.local:48522 raiden.upc.es:www SYN_SENT
tcp 1 1 casa-linux.local:48560 raiden.upc.es:www LAST_ACK
tcp 0 0 casa-linux.local:48556 raiden.upc.es:www TIME_WAIT
tcp 0 0 casa-linux.local:48556 raiden.upc.es:www TIME_WAIT
.....
```

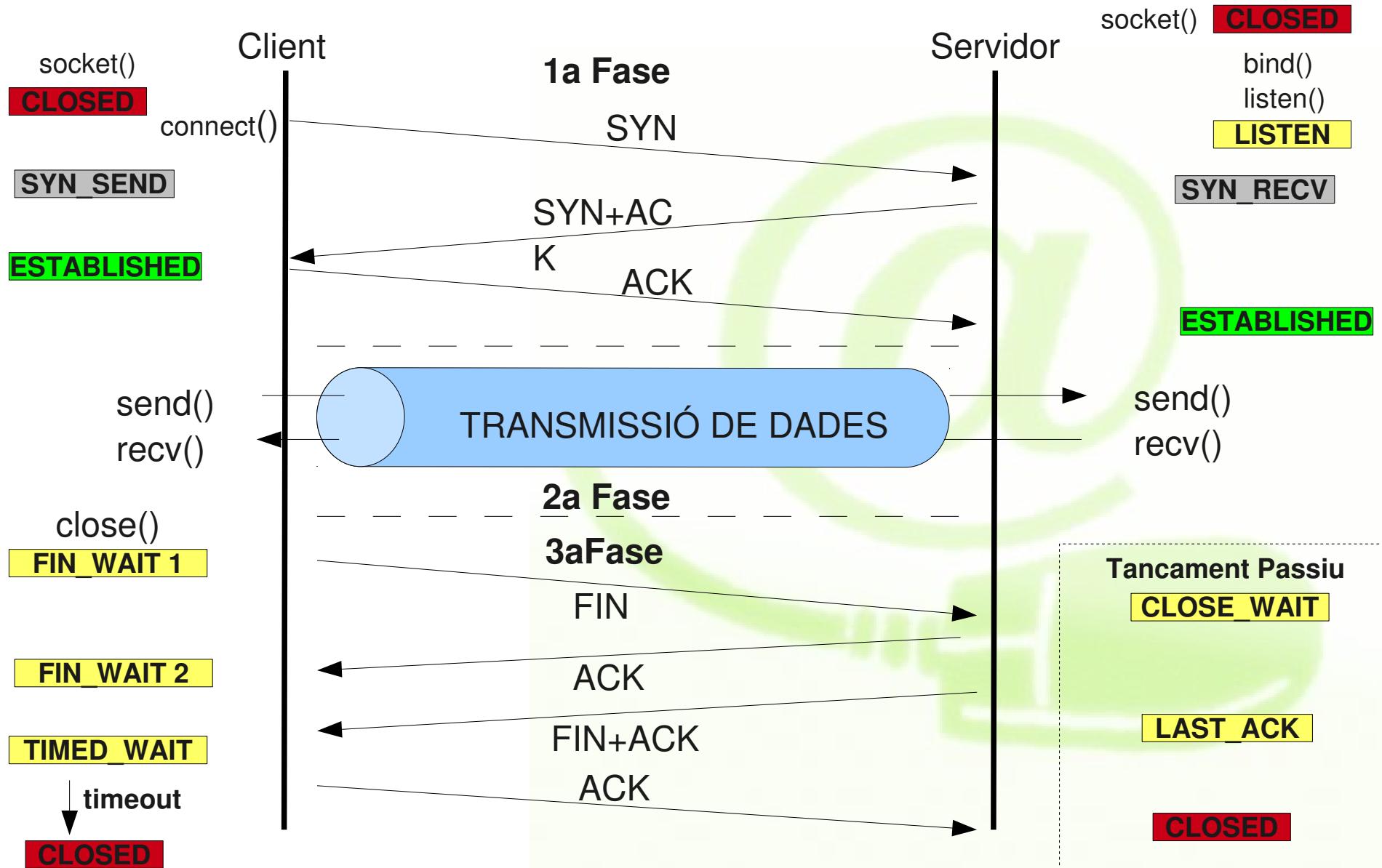
- ◆ I seguidament connecteu-vos a la web de la UPC
- ◆ Ara feu el mateix amb una connexió SSH
- ◆ Comanda netstat a la [wiki del curs](#)



# Netstat

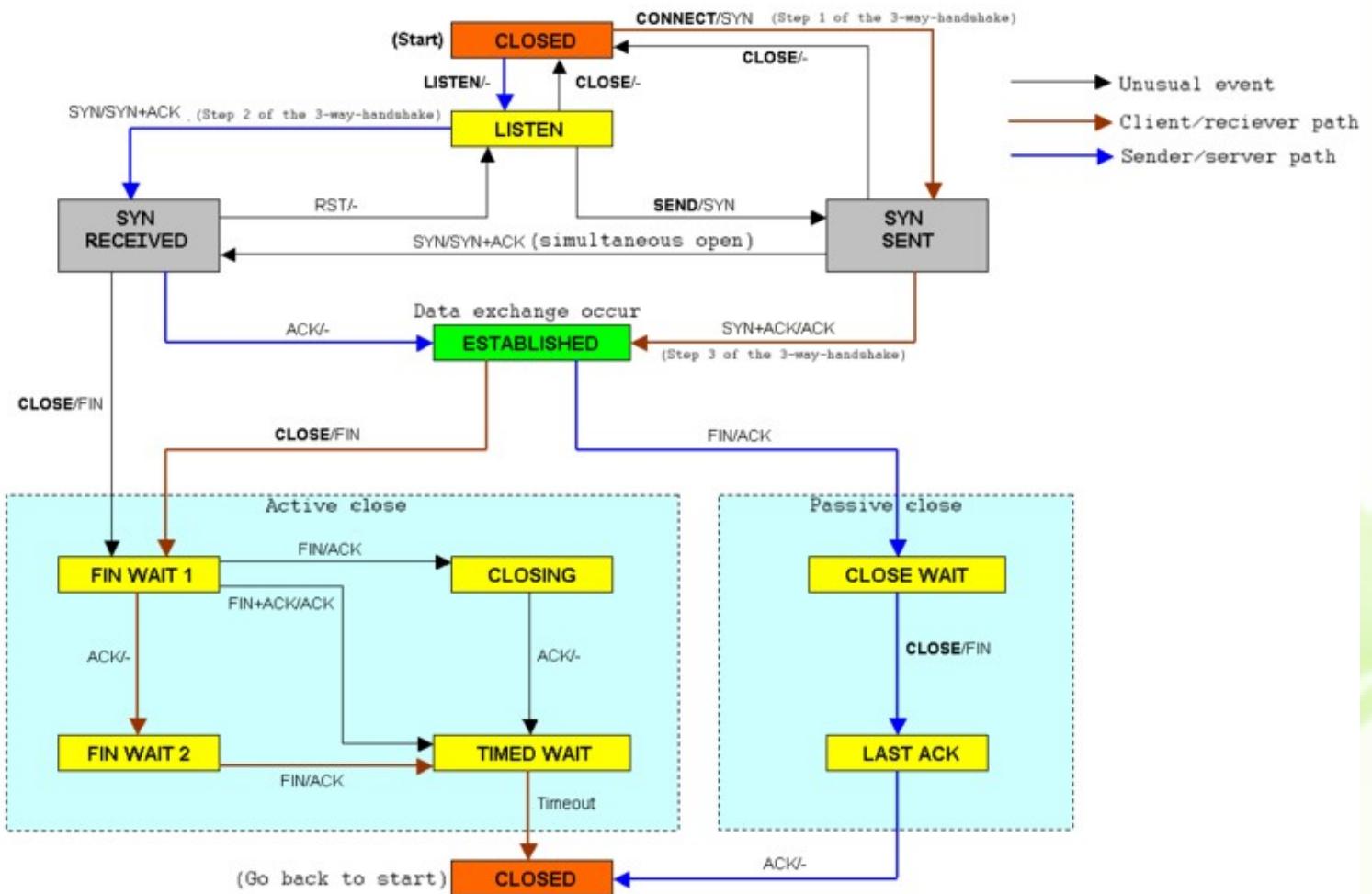


# Estats d'un socket. Establiment de connexió



# Estats d'un socket

- El diagrama d'estats complert és el següent





# Estats d'un socket

## • **Timeouts**

- En les comunicacions sempre hi han errors (de programació, dispositius que es pengen, caigudes de xarxa, etc).
- Per evitar que els sockets es quedin penjats en una estat fins indeterminadament s'estableixen uns **timeouts**.

ESTAT	TIMEOUT
NONE	30 minuts
ESTABLISHED	5 dies
SYN_SENT	2 minuts
SYN_RECV	60 segons
FIN_WAIT	2 minuts
CLOSE	10 segons
CLOSE_WAIT	12 hores
LAST_ACK	30 segons
LISTEN	2 minuts

# Serveis sense connexió

## ◆ Propietats

- ◆ Ofereixen la capacitat de comunicació sense necessitat de realitzar una connexió amb el destinatari.
- ◆ L'emissor envia paquets de dades al receptor confiant en que la xarxa tindrà prou intel·ligència com per a conduir les dades per rutes adequades.
- ◆ Els paquets poden seguir rutes diferents durant la comunicació.
- ◆ Els blocs de dades es poden rebrà desordenats.
- ◆ Cada paquet ha de portar l'adreça de destinació i, en alguns casos, el receptor ha d'enviar un acusament de rebuda per confirmar l'èxit de la comunicació.
- ◆ **Exemple:** Correu postal

# Protocols no orientats a connexió. UDP

- ◆ **Datagrames**

- ◆ S'envien directament paquets de l'emissor al receptor sense establir prèviament una connexió.

- ◆ **User Datagram Protocol (UDP)**

- ◆ No és una connexió fiable, ja que els paquets poden arribar en qualsevol ordre, duplicats, es poden perdre...

- ◆ **Protocols:**

- DNS, DHCP, VoIP, Videoconferència, jocs en xarxa.



# Serveis sense connexió

## ◆ Tipologies

- ◆ **Servei de datagrama sense confirmació.**
  - L'emissor no necessita confirmació per part del receptor de que els paquets de dades li arriben correctament (protocol IP)
- ◆ **Servei de datagrama amb confirmació.** El receptor envia confirmacions a l'emissor. (correu electrònic)
- ◆ **Servei de petició i resposta.**
  - És un servei propi de gestió interactiva basat en que a cada petició li segueix una resposta. (peticions a bases de dades).

**Datagrama:** L'origen prové de la paraula telegrama



# Ports

## Ports

- ◆ Un port és una connexió virtual que pot ser utilitzada per les aplicacions per intercanviar dades.
- ◆ Els ports més comuns són els dels protocols TCP i UDP
- ◆ **Notació:** Decimal (22, 80) o Hexadecimal
- ◆ El fitxer **/etc/services** manté una llista de ports i els seus serveis associats.
- ◆ Cada port està associat a un servei per la IANA
  - Els ports per defecte dels serveis es poden canviar

```
$ cat /etc/services | more
...
tcpmux          1/tcp
# TCP port service multiplexer
echo            7/tcp
echo            7/udp
...
```

# Ports

- **Els ports (de destinació) més habituals són:**
  - ◆ **20 i 21:** Port de dades i control de FTP respectivament. TCP
  - ◆ **22:** Servidor SSH (OpenSSH). TCP
  - ◆ **23:** Servidor Telnet. TCP
  - ◆ **25:** SMTP. Enviament de correu electrònic no segur
  - ◆ **53:** Servidor de DNS (Bind). UDP per a consultes (TCP per a transferències de zona)
  - ◆ **80:** Servidor web (Apache). TCP



# Ports

- ◆ **110:** POP3. Recepció de correu electrònic. TCP
- ◆ **119:** Servidor de notícies (Usenet News Transfer P)
- ◆ **137, 138 i 139:** Netbios protocols, Samba. WINS (UDP)
- ◆ **143:** IMAP. Recepció de correu electrònic. TCP
- ◆ **161:** SNMP. Monitorització. UDP
- ◆ **443:** Protocol HTTPS. Servidor Web Apache TCP.
- ◆ **465:** SMTP segur
- ◆ **993:** IMAP segur
- ◆ **995:** POP3 segur

# Sockets



- ◆ **Sockets**
  - ◆ Dispositius virtuals de comunicacions bidireccionals.
- ◆ **Hi han tantes famílies de sockets com protocols**
  - ◆ Unix Domain Sockets
  - ◆ Internet Sockets (TCP, UDP i RAW)



# Unix domain socket

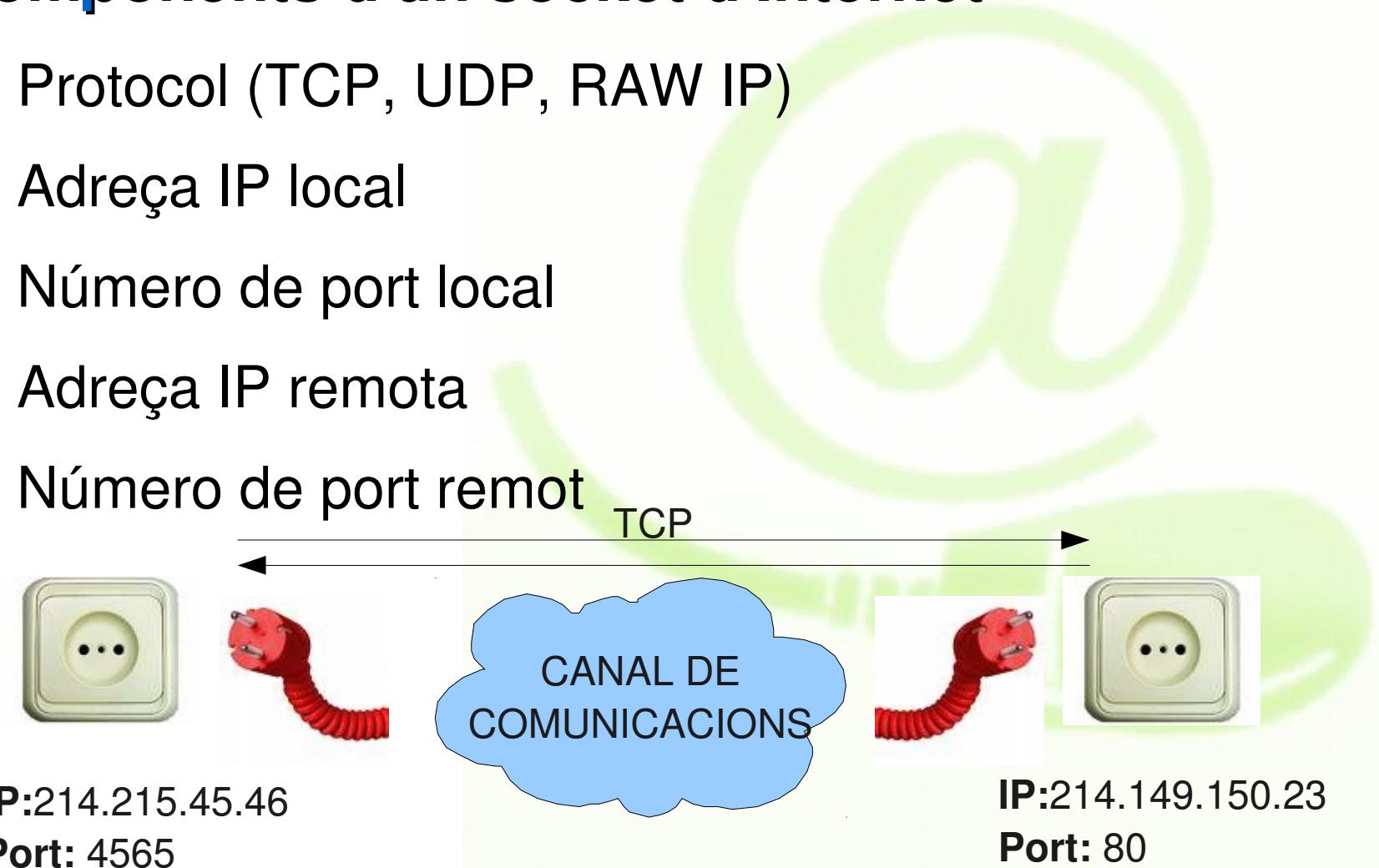
- ❖ **Unix domain socket (UDS o IPC socket)**
  - ❖ Són sockets virtuals, similars als sockets d'Internet que s'utilitzen en sistemes operatius POSIX per a la comunicació entre processos (IPC)
  - ❖ També anomenats POSIX Local IPC Sockets.
- ❖ **Components**
  - ❖ Tipus: Datagrama o Stream
  - ❖ Camí absolut del fitxer

```
$ ls -la /var/run/mysql/mysqld.sock
srwxrwxrwx 1 mysql mysql 0 2007-05-10 07:42 /var/run/mysql/mysqld.sock
```

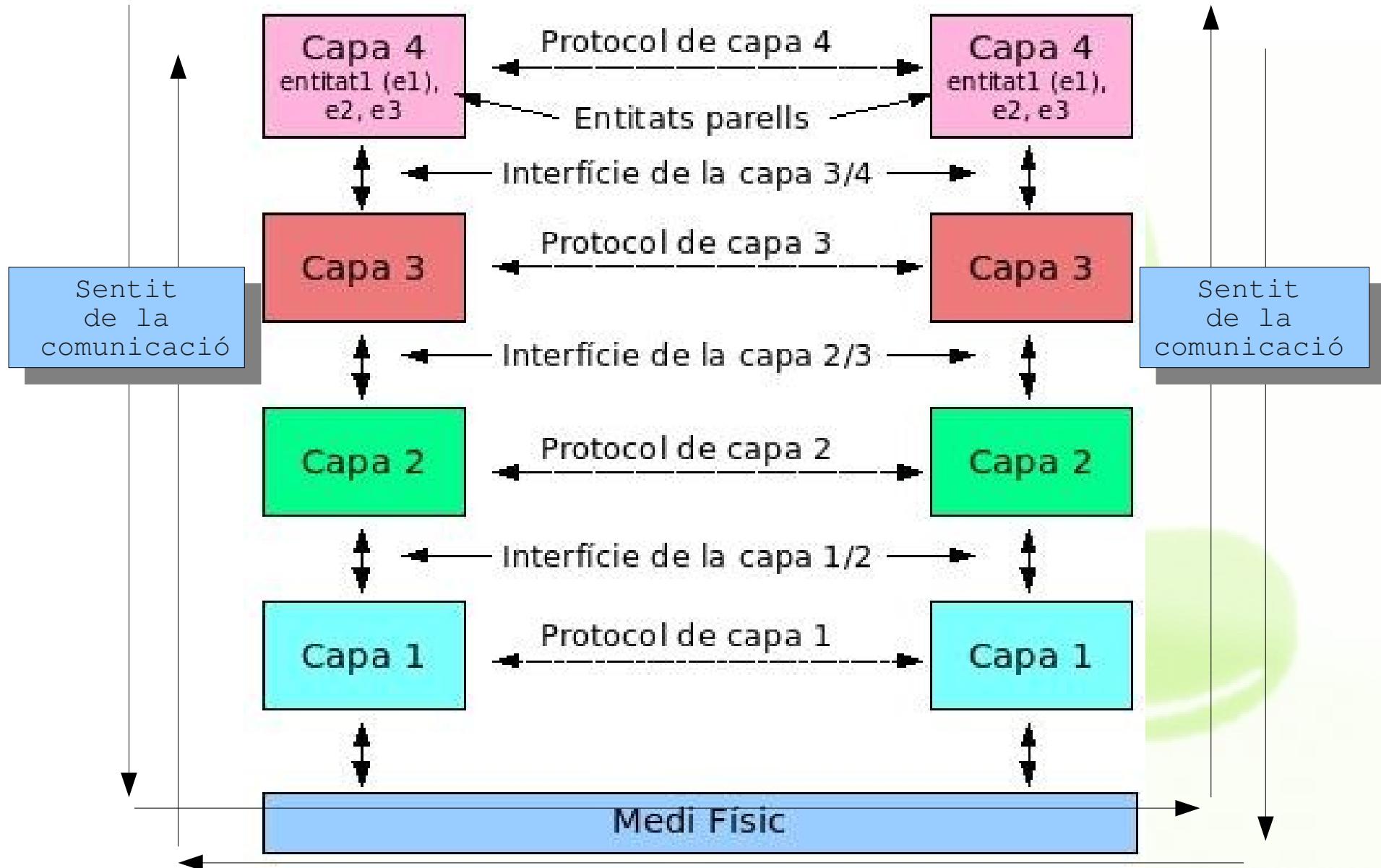
# Sockets d'Internet

## ◆ Components d'un socket d'Internet

- ◆ Protocol (TCP, UDP, RAW IP)
- ◆ Adreça IP local
- ◆ Número de port local
- ◆ Adreça IP remota
- ◆ Número de port remot



# Ports, serveis i sockets



# Ports, serveis i sockets

Client SSH

```
$ ssh alumne@192.168.201.5
```



IP: 192.168.201.4

Port: 4565

Servidor SSH

Adreça IP: 192.168.201.5



IP: 192.168.201.5

Port: 80

## CAIXA NEGRA

(la resta de nivell, Internet, cables  
de xarxa...)



# Protocol TCP

## Transmission Control Protocol (TCP)

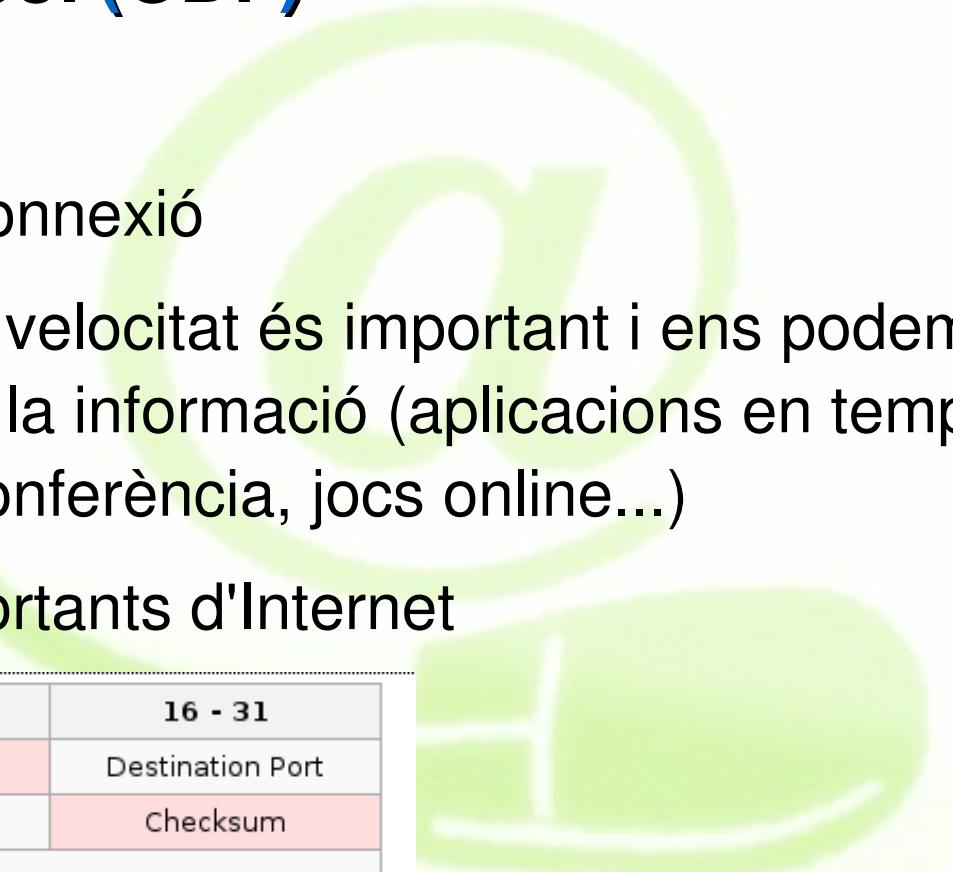
- ◆ Protocol de nivell 4
- ◆ Serveis orientats a connexió
- ◆ Utilitzat en serveis on cada bit és important (transmissió de dades, fitxers, pàgines web, correu electrònic...)
- ◆ És un dels protocol importants d'Internet

TCP Header											
Bit offset	Bits 0-3	4-7	8-15				16-31				
0	Source port				Destination port						
32	Sequence number										
64	Acknowledgment number										
96	Data offset	Reserved	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN	Window
128	Checksum								Urgent pointer		
160	Options (optional)										
160/192+	Data										

# Protocol UDP

## ‣ User Datagram Protocol (UDP)

- Protocol de nivell 4
- Serveis no orientats a connexió
- Utilitzat en serveis on la velocitat és important i ens podem permetre perdre part de la informació (aplicacions en temps real com veu IP, videoconferència, jocs online...)
- És un dels protocol importants d'Internet



+	Bits 0 - 15	16 - 31
0	Source Port	Destination Port
32	Length	Checksum
64	Data	



# TCP vs UDP

- **Els dos protocols de nivell de transport més utilitzats són TCP i UDP.**
  - ◆ TCP és més fiable però més lent. S'utilitza en comunicacions on la integritat de les dades és vital (per exemple la **transferència de fitxers**)
  - UDP és menys fiable però més ràpid (aprox. 40% ). S'utilitza en aplicacions on la velocitat és important i ens podem permetrà la pèrdua d'algunes dades (P. ex. **serveis en temps real** com la **telefonia IP** o **videoconferència**)

	TCP	UDP
Mida de la capçalera	20 bytes	8 Bytes
Unitat de dades	Segment	Paquet
Control d'errors	Si	Si
Ports	Si	Si
Orientat a connexió	Si	No
Control de flux	Si	No
Control de congestió	Si	No
Numeració de segments	Si	No
Automatic Repeat Request	Si	No



# Eines

## • Telnet

Protocol HTTP

- ◆ Com a eina de connexió remota ha estat substituïda per SSH. Servei però per obrir connexions TCP

```
$ telnet www.google.com 80
Trying 209.85.229.104...
Connected to www.l.google.com.
Escape character is '^]'.
GET /index.html
HTTP/1.0 302 Found
Location: http://www.google.es/index.html
Cache-Control: private
Content-Type: text/html; charset=UTF-8
.
<HTML><HEAD><meta http-equiv="content-type" content="text/html; charset=utf-8">
<TITLE>302 Moved</TITLE></HEAD><BODY>
<H1>302 Moved</H1> The document has moved
<A HREF="http://www.google.es/index.html">here</A>.
</BODY></HTML>
```

Connexió TCP al port 80

GET REQUEST

HEADER RESPONSE

Codi HTML

- ◆ Consulteu la [wiki del curs](#)



# NAT (Traducció d'adreça de xarxa)

## ◆ **Network Address Translation**

- ◆ És un estàndard creat de la Internet Engineering Task Force (IETF). Creat per lluitar contra la falta d'IPs.

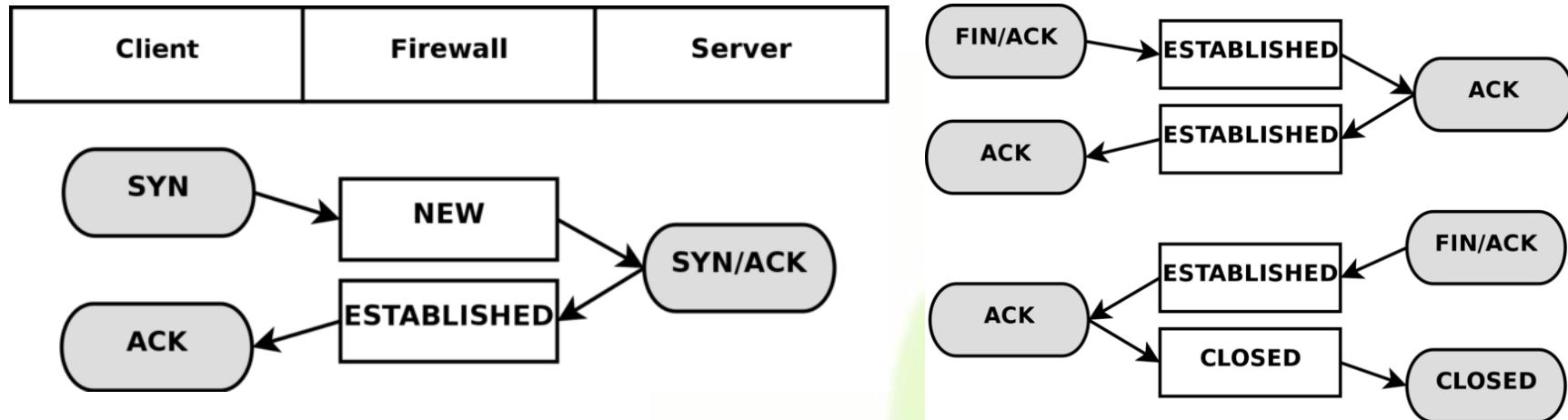
## ◆ **Dos usos, dos tipus de NAT**

- ◆ **SNAT (Source NAT)**: Compartir una connexió a Internet. Permet compartir una adreça vàlida d'Internet entre diverses adreces de xarxa privades.
- ◆ **DNAT (Destination NAT)**: Permet accedir als serveis d'una màquina local.

## ◆ **Funcionament**

- ◆ Canvia les adreces d'Internet (SNAT adreces origen i DNAT adreces destinació) de les capçaleres IP.

# Connection Tracking



- ◆ iptables pot controlar l'estat de les connexions dels protocols TCP, UDP i ICMP.

## ◆ **SNAT. Compartició de la connexió.**

- ◆ Que passa amb els paquets de retorn (P. ex. retorn d'una pàgina web consultada per un PC local)?
  - Com podem recordar, les connexions, un cop s'estableix una connexió, ja es recorda el seu origen i no es denega el paquet de tornada.

# Exemple de SNAT

- **Flash Cisco sobre NAT**
- **Utilitzat en les màquines que fan de passarel·les**
  - ◆ Els requisits que explicàvem abans per a les passarel·les també s'apliquen ara.
- **SNAT també és conegut com Masquerade**
  - ◆ De fet, masquerade és millor ja que permet que el gateway tingui una IP dinàmica.
- **Documentació**
  - ◆ <file:///usr/share/doc/iptables/html/NAT-HOWTO.html>

# Exemple de DNAT

- Configuració per interfície gràfica

**DrayTek Router Web Configurator**

> Advanced Setup > NAT Setup > Port Redirection [<< Main Menu](#)

**Port Redirection Table** [<< Back](#)

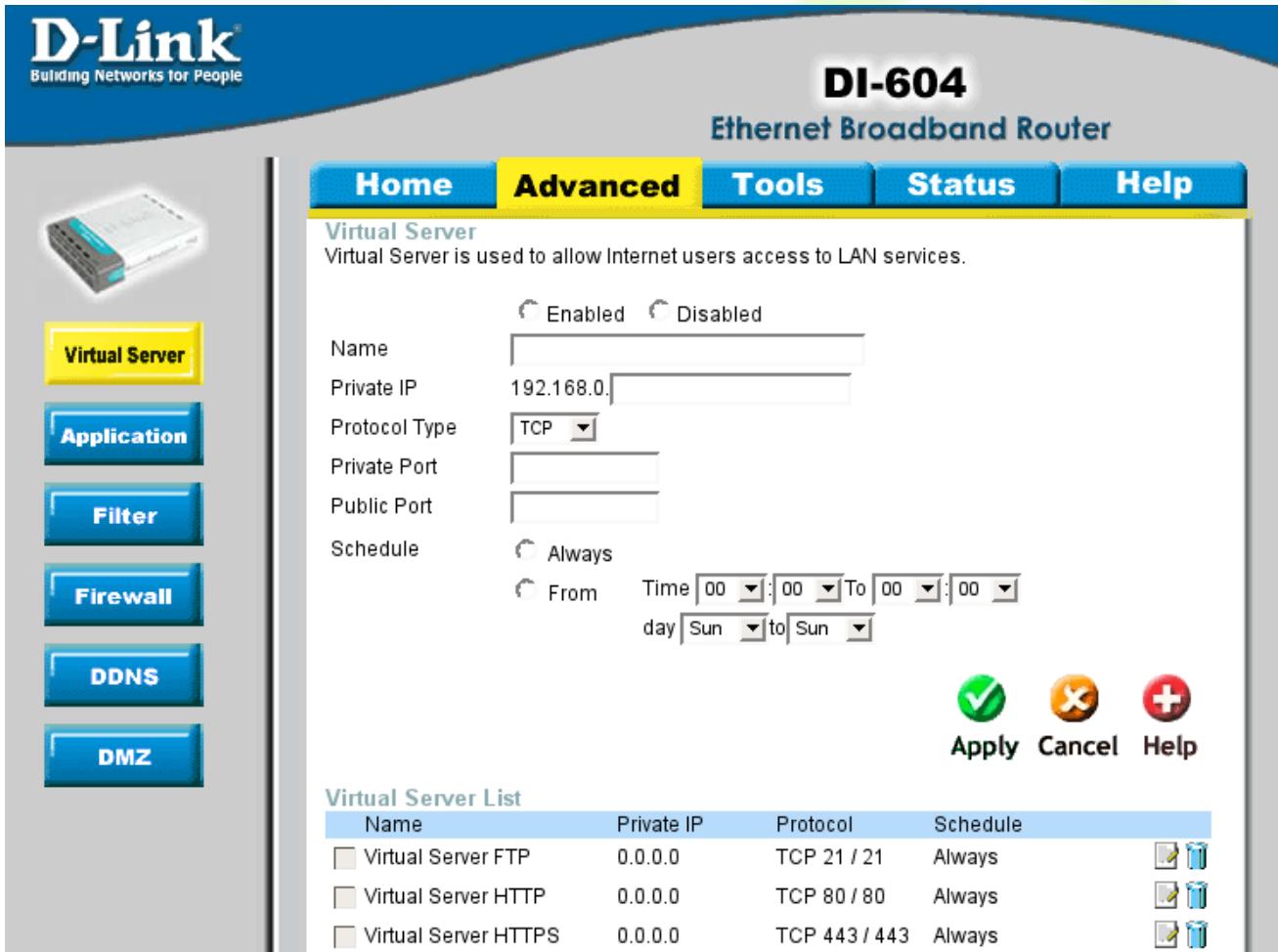
Index	Service Name	Protocol	Public Port	Private IP	Private Port	Active
1	ssh	TCP	22	10.0.3.234	22	<input checked="" type="checkbox"/>
2	smtp	TCP	110	10.0.3.234	110	<input checked="" type="checkbox"/>
3	pop3	TCP	25	10.0.3.234	25	<input checked="" type="checkbox"/>
4	www	TCP	80	10.0.3.234	80	<input checked="" type="checkbox"/>
5	ssh2	TCP	24	10.0.2.2	22	<input checked="" type="checkbox"/>
6	www2	TCP	8080	10.0.2.2	80	<input checked="" type="checkbox"/>
7	webmin2	TCP	10000	10.0.2.2	10000	<input checked="" type="checkbox"/>
8		---	0		0	<input type="checkbox"/>
9		---	0		0	<input type="checkbox"/>
10		---	0		0	<input type="checkbox"/>

[OK](#)

Copyright (c) 2002, DrayTek Corp. All Rights Reserved.

# Exemple de DNAT

- DNAT = Obrir ports del router

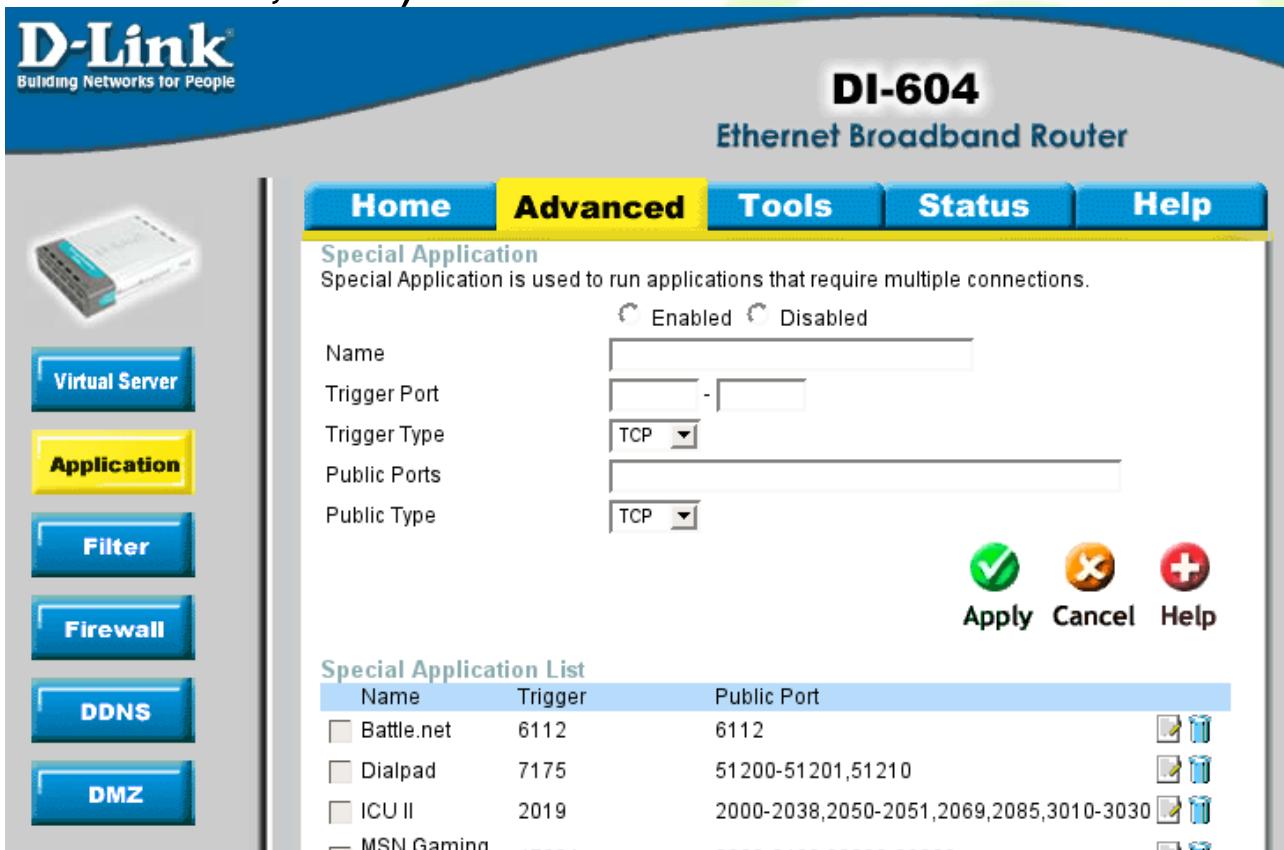


The screenshot shows the D-Link DI-604 Ethernet Broadband Router's configuration interface. The left sidebar has buttons for Virtual Server, Application, Filter, Firewall, DDNS, and DMZ. The main area has tabs for Home, Advanced (selected), Tools, Status, and Help. The Advanced tab has a sub-section for Virtual Server. It says "Virtual Server is used to allow Internet users access to LAN services." There are two radio buttons: Enabled (unchecked) and Disabled (checked). Below that is a table with fields for Name (empty), Private IP (192.168.0.), Protocol Type (TCP), Private Port (empty), Public Port (empty), Schedule (Always selected), From (Time: 00:00 To 00:00, day: Sun to Sun), and a checkbox for Always. At the bottom are Apply, Cancel, and Help buttons. A "Virtual Server List" table shows three entries: Virtual Server FTP (Private IP 0.0.0, Protocol TCP 21/21, Schedule Always), Virtual Server HTTP (Private IP 0.0.0, Protocol TCP 80/80, Schedule Always), and Virtual Server HTTPS (Private IP 0.0.0, Protocol TCP 443/443, Schedule Always).

Name	Private IP	Protocol	Schedule
Virtual Server FTP	0.0.0	TCP 21 / 21	Always
Virtual Server HTTP	0.0.0	TCP 80 / 80	Always
Virtual Server HTTPS	0.0.0	TCP 443 / 443	Always

# Exemple de DNAT

- Utilitzat per poder accedir directament a algunes aplicacions des de l'exterior de la xarxa (eMule, jocs de xarxa, servidors, etc)





# Netstat

- Ens mostrat l'estat de les connexions d'una màquina
  - Comanda disponible en Windows i Linux

```
$ netstat --inet puta
```

Active Internet connections (w/o servers)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	ubuntu-sala.local:36009	acacha.dyndns.org:www	ESTABLISHED
tcp	0	0	ubuntu-sala.local:36010	acacha.dyndns.org:www	ESTABLISHED
tcp	0	0	ubuntu-sala.local:36008	acacha.dyndns.org:www	TIME_WAIT
tcp	0	0	ubuntu-sala.local:36710	nobel.upc.es:imaps	ESTABLISHED

- Netstat a la wiki



# Nmap

## ◆ Escànners de ports

- ◆ Comanda que utilitzem sovint per comprovar quin serveis estan oberts en una màquina

```
$ sudo nmap localhost
Starting Nmap 4.20 ( http://insecure.org ) at 2008-02-
27 11:23 CET
Interesting ports on localhost (127.0.0.1):
Not shown: 1686 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    open  smtp
80/tcp    open  http
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
631/tcp   open  ipp
901/tcp   open  samba-swat
3000/tcp  open  ppp
3306/tcp  open  mysql
```

- ◆ Nmap a la wiki del professor



# FTP

## • **FTP File Transfer Protocol**

- ◆ Dissenyat per que els usuaris puguin copiar fitxers entre dos computadors via internet. Té en compte que cada computador:
  - Pot tenir un sistema operatiu propi: No tenen per que ser iguals.
  - Els fitxers poden estar en sistemes d'emmagatzemament diferents.
  - Utilitzar diferents codis de caràcters.
- ◆ Probablement un dels protocols més antics d'Internet per transferir fitxers. Primera descripció en el RFC 114 de 197.
- ◆ Inicialment usat dins del M.I.T. Després adaptat a Arpanet i finalment en 1980 (RFC 765) es defineix sobre TCP/IP.
- ◆ Obsolet per falta de seguretat. Podeu utilitzar SSH amb scp i sftp (tots dos pel port 22)
- ◆ Consulteu la [wiki del curs](#)

 creative commons

Reconeixement 3.0 Unported

**Sou lliure de:**

-  copiar, distribuir i comunicar públicament l'obra
-  fer-ne obres derivades

**Amb les condicions següents:**

-  **Reconeixement.** Heu de reconèixer els crèdits de l'obra de la manera especificada per l'autor o el llicenciat (però no d'una manera que suggereixi que us donen suport o rebeu suport per l'ús que feu l'obra).
- Quan reutilitzeu o distribuïu l'obra, heu de deixar ben clar els termes de la llicència de l'obra.
- Alguna d'aquestes condicions pot no aplicar-se si obtenuï el permís del titular dels drets d'autor.
- No hi ha res en aquesta llicència que menyscabi o restringeixi els drets morals de l'autor.

Els drets derivats d'usos legítims o altres limitacions reconegudes per llei no queden afectats per l'anterior  
Això és un resum fàcilment llegible del text legal (la llicència completa).

Advertiment 

<http://creativecommons.org/licenses/by/3.0/deed.ca>

LPIC-1. Examen 102. Objectiu 107.3

ICE-UPC



Autor: Sergi Tur Badenas