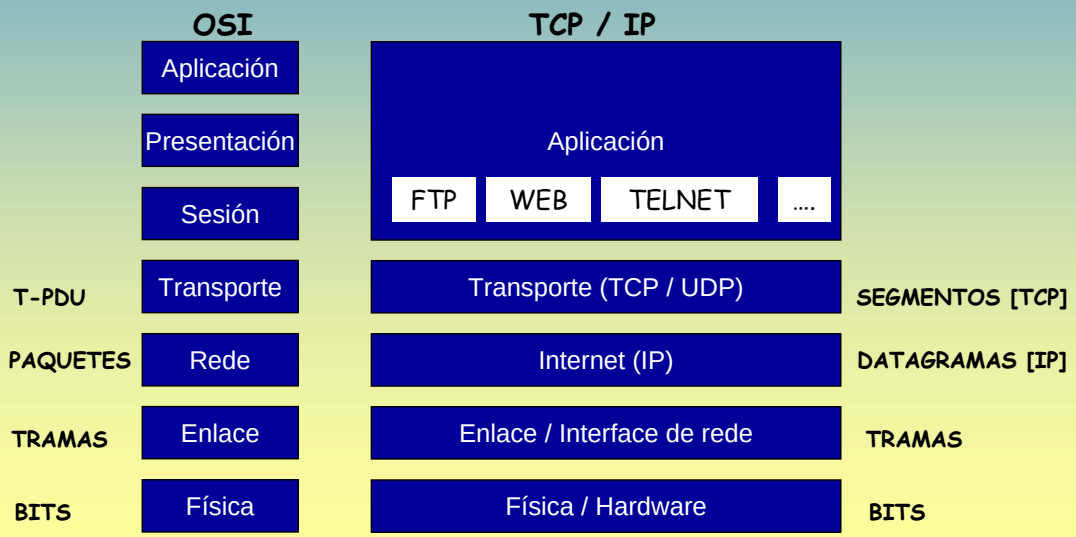
OSI – TCP/IP

IES San Clemente
Ver. 2.6 (27-06-05)

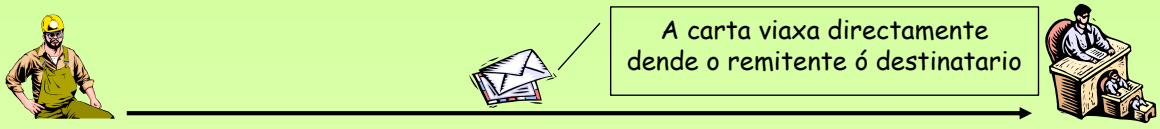


Carlos Carrión Álvarez

1.- Introducción – Arquitectura de Redes

☞ Dous amigos envíanse unha carta.

A imaxe que temos do proceso de envío é o seguinte.



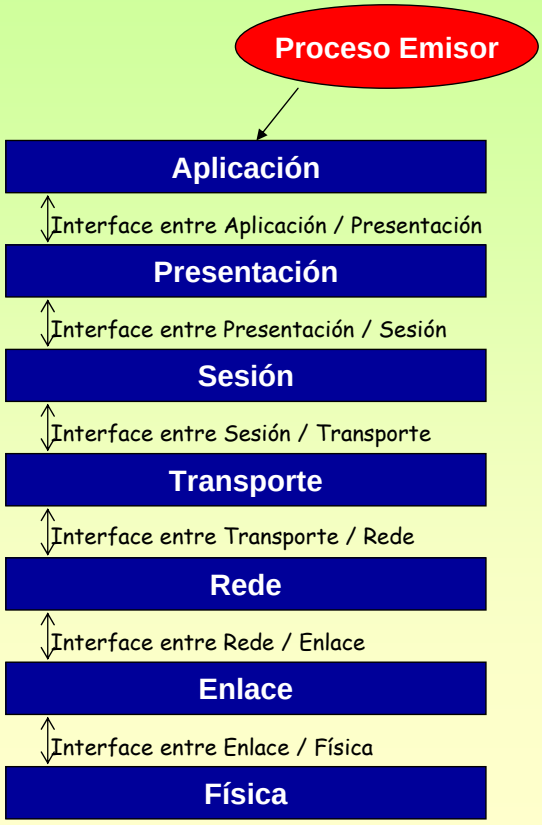
☞ A realidade.

A carta vai a través de diversos medios:
- Oficinas de Correo
- Estacións de tren, etc.

En cada un destes intermediarios engadiráselle información:
- Certificada, (S/N)
- Urxente (S/N), etc.

1.- Introducción

1.- Introducción – Modelo OSI de ISO (1984)

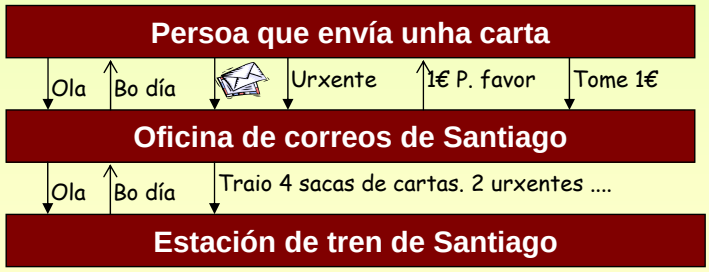


- ISO: International Standard Organization
  (Organismo de estándares internacionais)
- OSI: Open System Interconection.
  (Interconexión de sistemas abertos/heteroxéneos)
- Arquitectura organizada en 7 capas/niveis
  Cada unha con unha función clara e ben definida

- INTERFACES

É o lugar polo que intercambian infomación dúas capas. Unha capa intercambia información coa súa superior/inferior inmediatas.

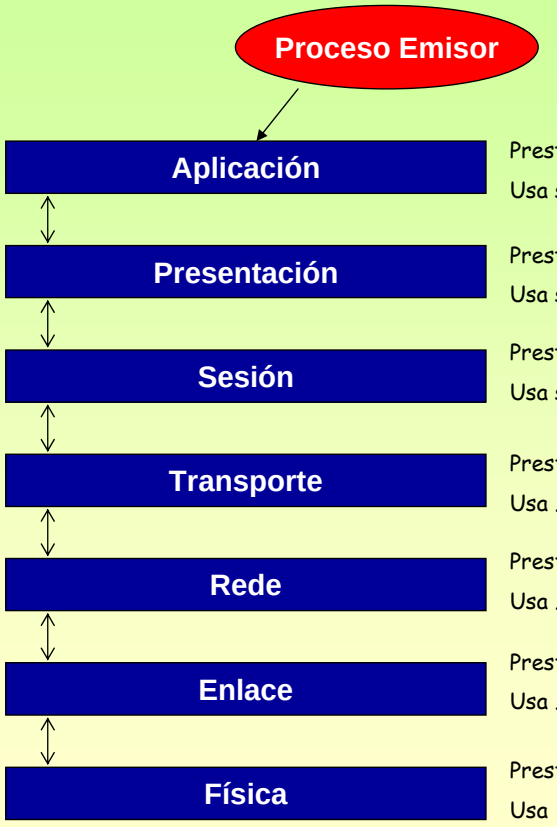
- P. Ex.: Unha persoa en Santiago envía unha carta



- A persoa non interactúa directamente coa estación

1.- Introducción

1.- Introducción – Modelo OSI de ISO (1984)



- SERVIZOS:
  Para que unha capa poida levar a cabo as súa funcións usa os servizos que lle proporciona á capa inferior.

Aplicación
Presta servizos ó proceso emisor
Usa servizos que ofrece presentación

Presentación
Presta servizos á capa Aplicación
Usa servizos que ofrece sesión

Sesión
Presta servizos á capa Presentación
Usa servizos que ofrece Transporte

Transporte
Presta ...
Usa ...

Rede
Presta ...
Usa ...

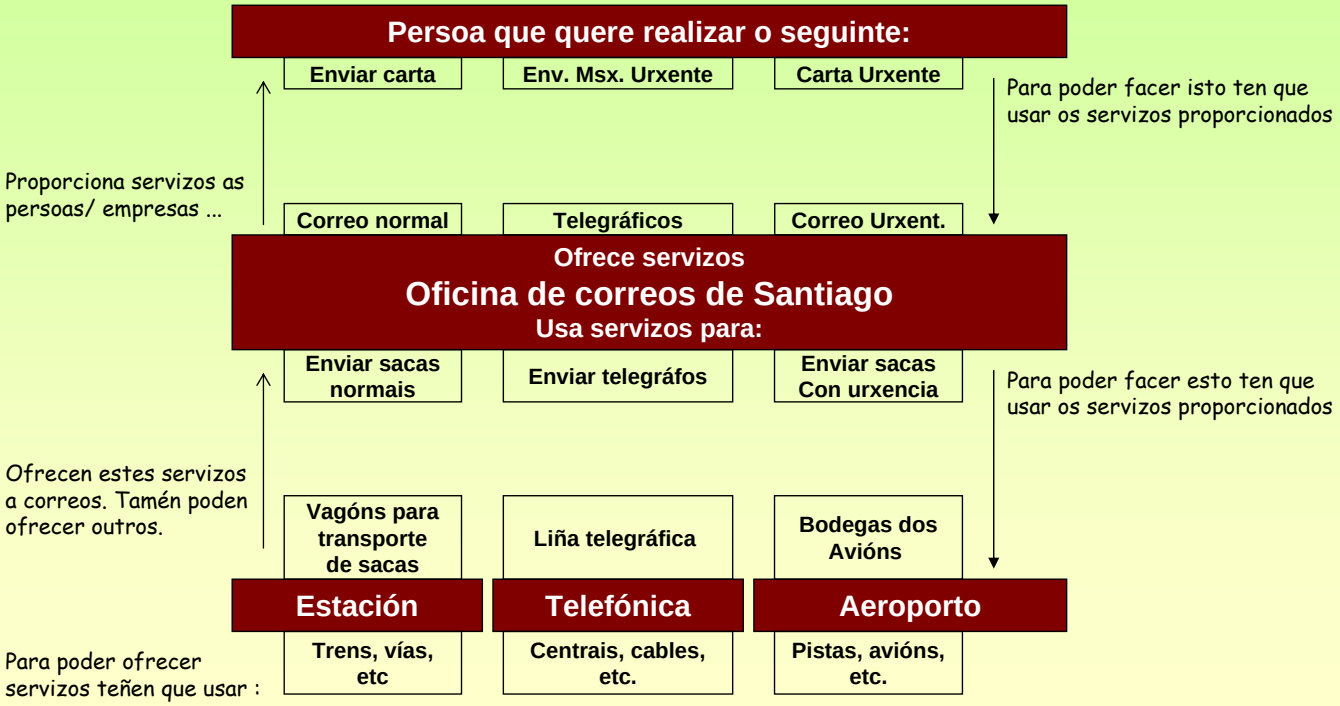
Enlace
Presta ...
Usa ...

Física
Presta ...
Usa os medios físicos...

1.- Introducción – Modelo OSI de ISO (1984)

☞ SERVIZOS - EXEMPLO

Unha persoa desexa enviar unha carta normal outra urxente e unha mensaxe urxente.



1.- Introducción – Modelo OSI de ISO (1984)

☞ ENTIDADES

Os servizos que ofrece unha capa son en realidade ofertados por ENTIDADES desa capa.
Cada capa ten un conxunto de entidades que son as que realizan e ofrecen os distintos servizos.

☞ EXEMPLO

Nunha oficina de correo hai unha/s entidade/s que se encargan de correo normal, outras de xiros, outras de correo urxente...

Na realidade son as ENTIDADES as que ofrecen/usan servizos non toda a capa en si.
En informática imaxinar un ordenador que ten un servidor WEB e un servidor FTP, cada un deles é unha entidade/programa distinto. Non todo o ordenador é o servidor WEB, senón que dentro dese ordenador hai unha entidade/aplicación que realiza esa función.

☞ SAP (Punto de acceso ó servizo)

As entidades ofrecen os seus servizos por un punto concreto, punto ó que se ten que dirixir a entidade da capa superior para poder usar ese servizo. En correos serían as xaneliñas (ventanillas).

☞ Tipos de servizos que se poden ofertar

SERVIZO NON ORIENTADO Á CONEXIÓN:

Equivale ó **sistema postal**. Ó enviar varias cartas a un mesmo destino non se teñen garantías de que chegen todas nin na mesma orde en que saíron.

SERVIZO ORIENTADO Á CONEXIÓN:

Equivale ó **sistema Telefónico**. Para realizar unha comunicación:

1º Realízase unha chamada para establecer unha comunicación.
2º Realízase o intercambio de información. (A información recíbese na mesma orde na que se envía → imaxe TUBO)
3º Unha vez rematada a comunicación, libérase a conexión (cólgase ó teléfono)

1.- Introducción – Modelo OSI de ISO (1984)

☞ EXEMPLO DE ENTIDADES E SAP

Unha entidade da capa superior intercambiará información cunha entidade da capa inferior polo SAP

Persoa que quere usar servizos de correos

SAP 1 | SAP 2 | SAP 3

Entidade Correo Normal	Entidade Telegráficos	Entidade Correo Urxente
Recursos humanos e materiais, que se encargan do correo normal	Recursos humanos e materiais, que se encargan dos telegramas	Recursos humanos e Materiais, que se encargan do C. Urxente

Oficina de correos de Santiago

Puntos polos cales a Entidade usuario accede ós servizos que prestan as entidades de correos.

Un usuario non entra por dentro do mostrador e deposita el a súa carta onde desexe, senón que interactúa por unha xanela (SAP) coa entidade correspondente.

En síntese:

Unha capa ten **ENTIDADES** que realizan **funcións** e estas **ofrecen** os seus **servizos** as entidades da capa **superior** polo **SAP**

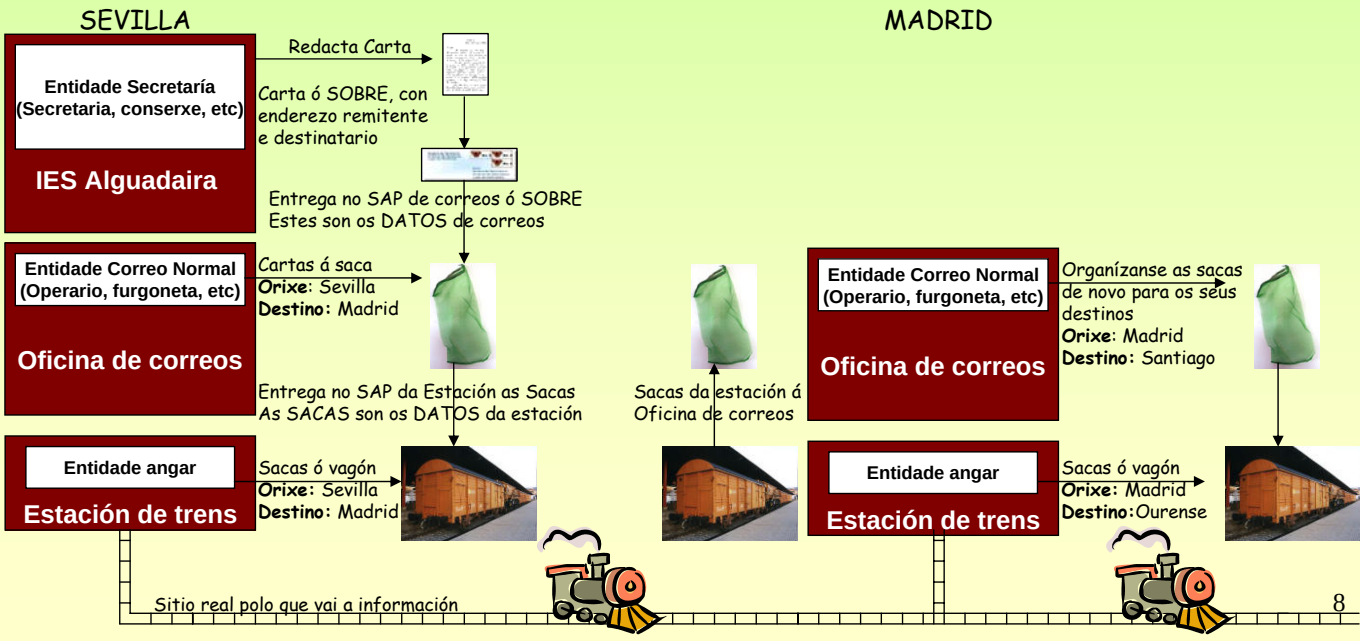
Por outra banda, a oficina de correos intercambiará información coa Estación de Trens, Aeroporto, Telefónica, etc, polos SAPs que estes poñan a disposición da oficina de correos

(No caso da estación e o aeroporto, podería ser a través dos angares, no caso de telefónica polo cable que e telégrafo que lles ten instalado na oficina).

1.- Introducción – Modelo OSI de ISO (1984)

☞ ENCAPSULACIÓN DA INFORMACIÓN

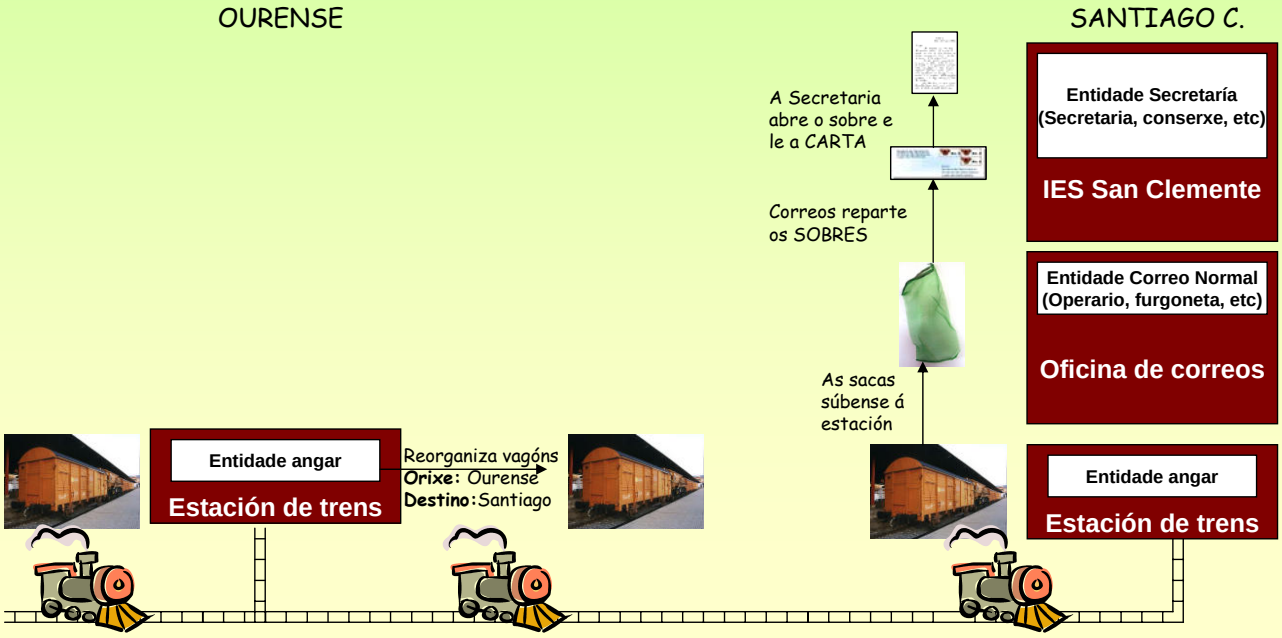
Un **REMITENTE / EMISOR** o único que desexa **transmitir/enviar** ó **DESTINATARIO / RECEPTOR** é unha **CARTA/MENSAXE** (entendida esta sen o sobre)
Pero a CARTA non pode viaxar pola rede de comunicación sen un **ENVOLTORIO/CABECEIRA** que lle permita a esta ser conducida ata o seu destino. Precísase un **SOBRE/CABECEIRA** no que transportar a carta.



1.- Introducción – Modelo OSI de ISO (1984)

☞ ENCAPSULACIÓN DA INFORMACIÓN

Un **REMITENTE/EMISOR** o único que desexa **transmitir/enviar** co **DESTINATARIO/RECEPTOR** é unha **CARTA/MENSAXE** (entendida esta sen o sobre)
Pero a CARTA non pode viaxar pola rede de comunicación sen un **ENVOLTORIO/CABECEIRA** que lle permita a esta ser conducida ata o seu destino. Precísase un **SOBRE/CABECEIRA** no que transportar a carta.



OSI – TCP/IP

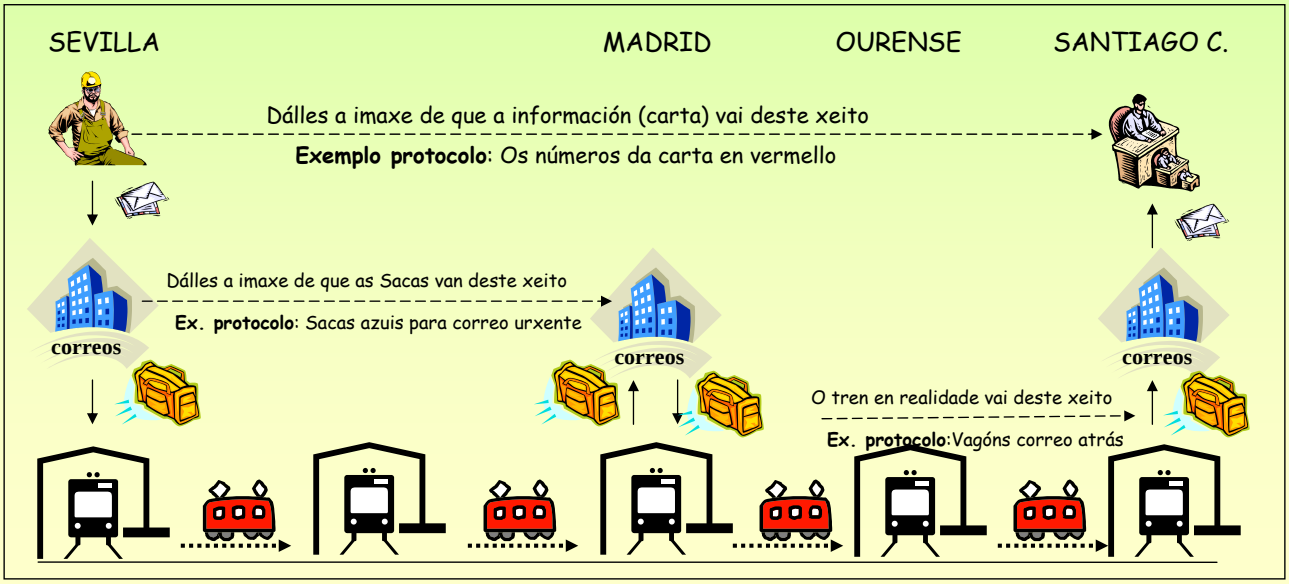
1.- Introducción

1.- Introducción – Modelo OSI de ISO (1984)

☞ SÍNTESE DO PROCESO DE TRANSMISIÓN, ENTIDADES PARES e PROTOCOLOS

Entidades PAR: son dúas entidades na mesma capa e en distinta máquina. (P.ex. Secretaría con Secretaría).

Protocolos: son as **normas/regras** que establece cada **entidade par** para comunicarse entre elas.



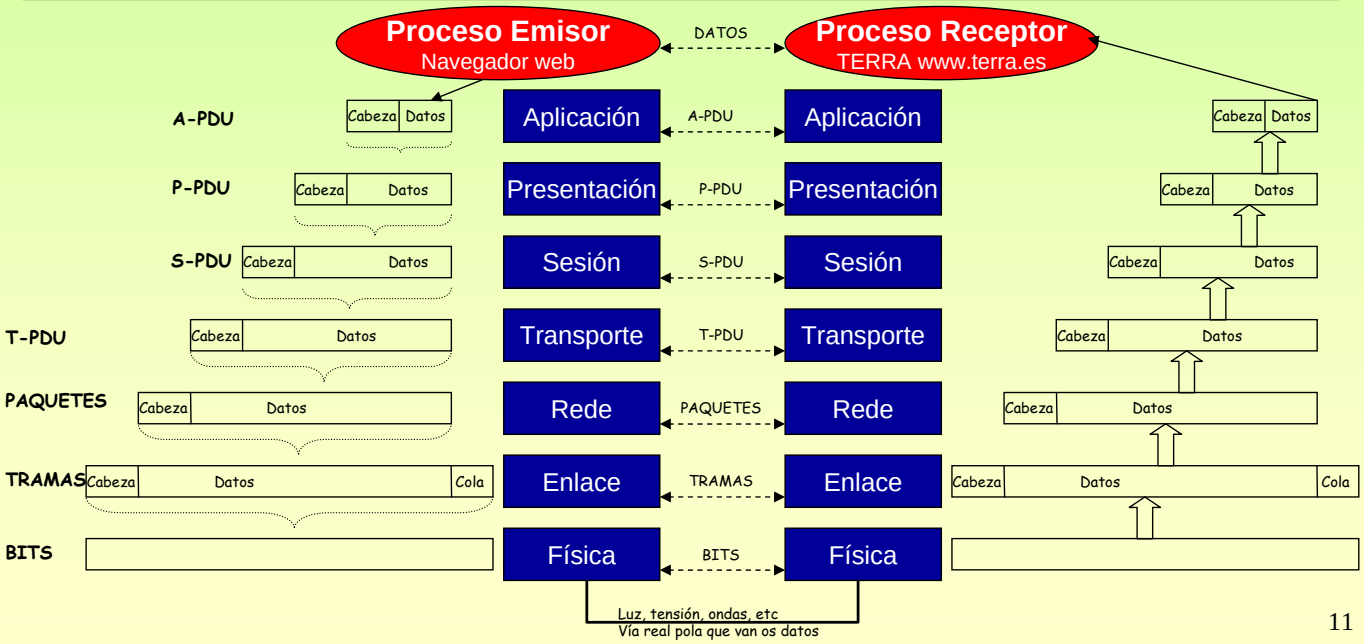
1.- Introducción – Modelo OSI de ISO (1984)

☞ **INTERCAMBIO DE INFORMACIÓN EN OSI**

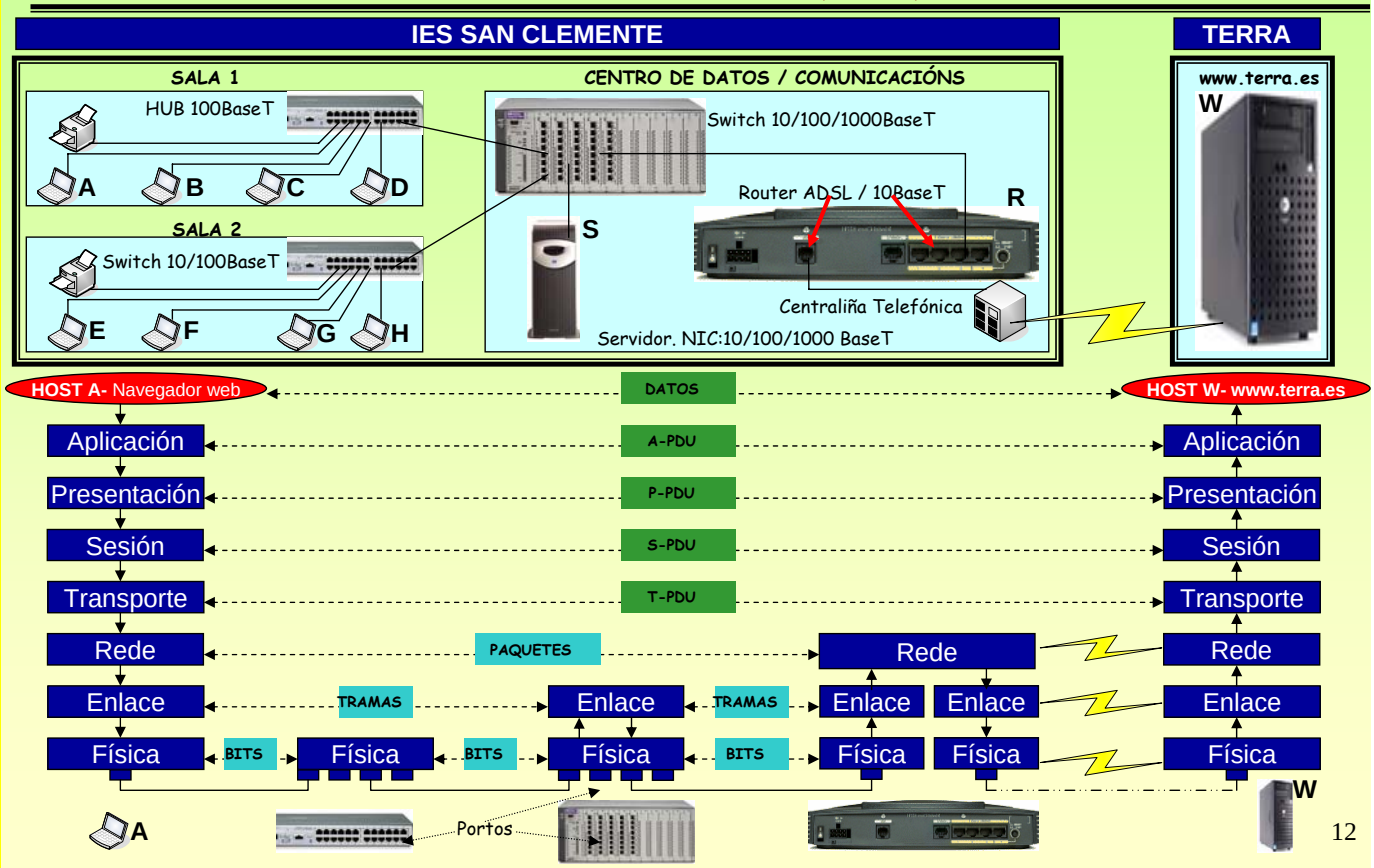
LADO EMISOR: As entidades de cada capa reciben mensaxes das entidades da capa superior, engaden unha cabeceira e baixan a nova mensaxe á capa inferior.

LADO RECEPTOR: As entidades de cada capa reciben das entidades da capa de abaixo as mensaxes, sacan a cabeceira e soben o campo de datos á capa superior.

PDU: (Unidade de datos do protocolo), é a mensaxe que intercambian as entidades pares.



1.- Introducción – Modelo OSI de ISO (1984)



1.- Introducción – Modelo OSI de ISO (1984)

☞ ALGUNHAS FUNCIÓNS DAS CAPAS / NIVEIS (Máis información na unidade de traballo 4)

Aplicación

Constrúe e procesa **A-PDUs**.
Neste nivel están as aplicacións como poderían ser o **FTP**, **DNS**, Servidor **Web**, **Correo electrónico**, etc

Presentación

Constrúe e procesa **P-PDUs**.
Sintaxe e semántica (se unha máquina traballa en Complemento a 1 e outra en complemento a 2, haberá que traducir)
Cifrado de datos (Encriptar/desencriptar a información que sae/chega a un host, P.ex. Chave simétrica, chave privada-pública)
Compresión dos datos (Se se trasmite un "**que**", no emisor podemos sacarlle o "**u**" e volverllo a poñer no receptor)

Sesión

Constrúe e procesa **S-PDUs**.
Encárgase da **xestión do diálogo** entre dúas máquinas finais (Quen transmite primeiro, como nos pasamos a testemuña, etc)

Transporte

Constrúe e procesa **T-PDUs**.
É o primeiro nivel extremo a extremo. (Para este nivel é como se non hai subrede, os protocolos son entre o emisor e receptor reais).
Encárgase do **control de fluxo entre hosts** (Imaxinar un emisor real, que manda libros por correo cada día a un receptor real. O correo, a estación, etc, non son saturados, pero o receptor non ten tempo de ler tódolos libros, o receptor real está saturado)

Rede

Constrúe e procesa **paquetes**.
Encamiña os paquetes. (Equivale a unha rotonda, xa que, ten sinais que indican que dirección coller para ir a un lugar).
Interconexión de redes distintas (Pe: ADSL-Ethernet)(Unha rotonda tamén pode ser o nexo dunha autoestrada cunha estrada)
Controla a conxestión (Unha rotonda conxestiónase se a suma de coches recibidos por tódalas liñas é maior o que pode procesar)

Enlace

Constrúe e procesa **tramas**. Controla o **fluxo** (que un emisor non sature a receptor).
Detección de erros, coa **COLA**. Emisor divide os datos entre un polinomio e o resto ponse na cola. No receptor faise a mesma división e contrástase o resto resultante co que chegou na cola.
Controla o **acceso á canle**: por **loita** (As estacións acceden cando queren), **regulado** (o acceso á canle faise de xeito ordenado)

Física

Encárgase da transmisión dos **bits** (luz, ondas, voltios)
Define aspectos relacionados con aspectos mecánicos, procedimentais, (P.e. conector RJ 45, o seu formato, que cables se usan)

1.- Introducción – CAPA DE ENLACE

☞ Descrición

Trata de asegurar unha conexión libre de erros entre dous ordenadores da mesma rede.

Extremo emisor

Acepta os paquetes do nivel de rede e **trocéaos** en tramas.
Constrúe os campos da trama.
Pasa as **tramas** ó nivel físico.

Extremo receptor

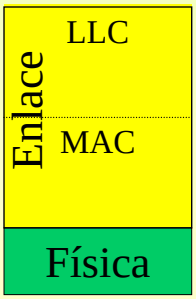
Compón a trama a partir dos bits que van subindo do nivel físico
Comproba os erros
Se a trama é correcta **sobe** a información ó nivel de rede.

☞ Subcapas

O nivel de enlace divídese en dúas subcapas con funcións claramente diferenciadas.

Subcapa LLC (Logic Link Control – Control de Enlace Lóxico)
Confección de tramas
Control de erros, ...

Subcapa MAC (Medium Access Control – Control de Acceso ó Medio)
¿Cando está a canle libre?
Se está libre ¿Podo transmitir?



1.- Introducción – ESTÁNDARES IEEE 802.X

☞ Introducción

A maioría das redes LAN (RAL) seguen os **estándares IEEE* 802**** para acceder ó medio compartido.

Nas LANs a información difúndese entre tódalas estacións,agás se se fai uso de switches, o que implica inseguridade na información

As especificacións 802.x definen tanto subcapa LLC (802.2) como a subcapa MAC e física (802.3, 802.4, 802.5, 802.6, 802.11, 802.12, FDDI, 802.11-inarámicas))

EXEMPLO

LLC	IEEE 802.2			
MAC	IEEE 802.3 Ethernet	Por contenda	IEEE 802.5 Token Ring	Anel con paso de Testemuña
Fisico		Coax: 10 BASE 2 UTP: 10/100 BASE T STP: 100 BASE T Fibra: 10/100 BASE F		STP: 4/16 Mbps UTP: 4 Mbps

*IEEE = Institute for Electrical and Electronics Engeneers

**802.x: Comités dentro do IEEE que desenvolveron os estandares uso de medios compartidos (802.3, 802.4,...)

1.- Introducción – IEEE 802.3 - ETHERNET

☞ FORMATO DAS TRAMAS

☞ Está baseado en CSMA / CD 1-persistente. (Técnica de acceso á canle mediante contenda/loita)

☞ A **trama MAC** está sincronizada polo modo de Principio e Conta. Esta ten o seguinte formato

Preámbulo	Inicio	Dir Destino	Dir Orixe	Lonxitude	Datos	Recheo	CRC
Bytes: 7	1	2 ou 6	2 ou 6	2	0 - 1500	0 – 46	4

Preámbulo: son 7 bytes: 10101010 Para que receptor e transmisor se sincronicen

Inicio: 1 byte: co patrón 10101011 Para indicar que comeza a trama

Dir Destino: é a dirección física (MAC) do destinatario da trama. A dirección física é única no mundo para cada adaptador (tarxeta).

Dir orixe: é a dirección física do transmisor. Hoxe en día nos dous campos de dirección úsanse 6 bytes e non 2. Estes bytes están expresados en Hexadecimal, cada 4 bits

Lonxitude: estes 2 bytes indican cantos bytes van no campo de datos ou de información

Datos: o campo de datos transporta a mensaxe do nivel superior. De 0 a 1500 bytes. Este campo define a **MTU (Maximum Transfer Unit)** da rede, isto é, cal é o tamaño máximo do paquete.

Recheo: Unha trama ethernet debe ter como mínimo 64 bytes, se o campo de datos ten menos de 46 bytes, débese usar o campo de recheo para completar eses 64 bytes.

CRC: Código de redundancia cíclica

1.- Introducción – IEEE 802.3 - ETHERNET

☞ NIVEL FÍSICO

☞ O comité 802.3 foi o que definiu máis configuracións físicas alternativas.

- ❑ Vantaxe: Adaptarse ás innovacións tecnolóxicas
- ❑ Inconveniente: Existencia de grande variedade de opcións
- ❑ Esta flexibilidade non implica que as distintas opcións non poidan estar integradas nun mesmo sistema

☞ O comité 802.3 desenvolveu unha notación concisa para distinguir as diversas opcións:

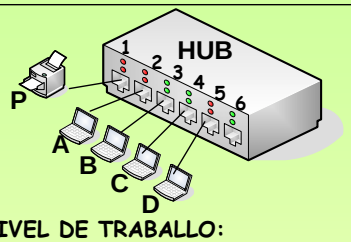
<Mbps> <sinalización> <máxima lonxitude do segmento en hectómetros se é coaxial ou tipo de cable se non é coax>

EXEMPLOS

10BASE 2	Segmentos de 200m de cable coaxial a 10 Mbps. Codificación Banda Base
10BASE 5	Segmentos de 500m de cable coaxial a 10 Mbps. Codificación Banda Base
10BASE T	Cable de pares Telefónico (T), codificación en Banda Base a 10 Mbps
100BASE TX	Cable de pares Telefónico (TX), codificación en Banda Base a 100 Mbps
100/1000BASETX	Cable de pares telefónico (TX), codificación en Banda Base a 100 ou 1000 Mbps
100BASEF	Cable de Fibra óptica (F), codificación en Banda Base a 100 Mbps

1.- Introducción – IEEE 802.3 - ETHERNET

☞ CONCENTRADOR (HUB) vs. CONMUTADOR (SWITCH)



NIVEL DE TRABALLO:
Físico: só entende de electricidade e non do significado do que por el está a pasar. Dito dun xeito non científico é como un **arame**.

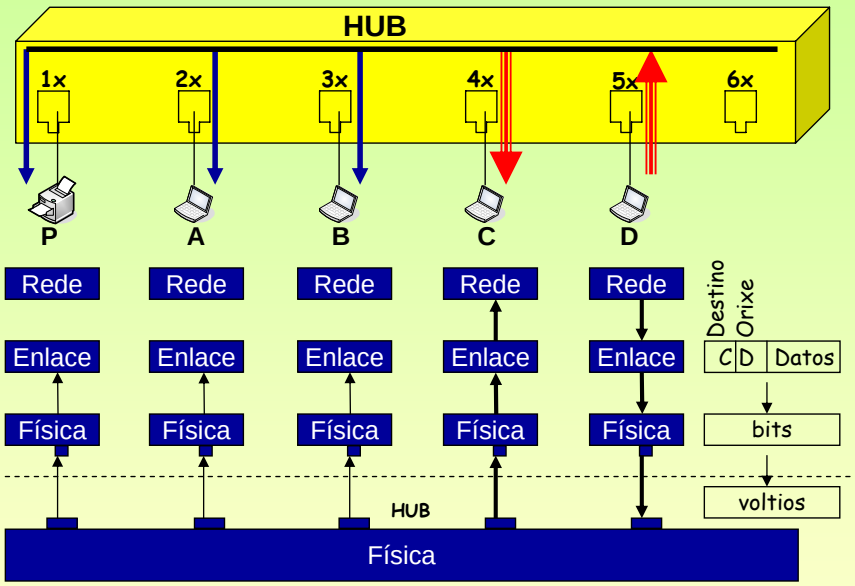
FUNCIONAMENTO:
Todo o que recibe o HUB por un porto é retransmitido polos demais portos

EXEMPLO:
O HOST D desexa enviar unha **trama** ó HOST C. Supoñer que os enderezos FÍSICOS/MAC son as letras A,B,C,D e P

ACTIVIDADE NOS RECEPTORES
Tódolos equipos salvo o transmisor (host D) reciben no nivel de enlace a trama enviada.

C: procesa a trama, pois el é o destinatario

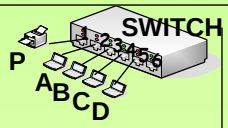
A, B e P: descartan a trama, pois eles non son os destinatarios



CONCLUSIÓNS:
1.- Cando transmite un equipo o hub **inunda** a rede molestando ós demais equipos, salvo ó receptor real.
2.- **Colisións:** cando tx dous ou máis equipos as tramas van chocar, pois por un mesmo porto enviaranse varias tramas simultaneamente.
3.- **Fácil roubo** de información, pois todos están recibindo canto pasa polo hub
4.- Se no proceso de envío se **modificou algún bit** da trama o hub non o pode detectar pois non é capaz de interpretar campos de información

1.- Introducción – IEEE 802.3 - ETHERNET

☞ CONCENTRADOR (HUB) vs. CONMUTADOR (SWITCH)



NIVEL DE TRABALLO:
ENLACE: ó traballar neste nivel entende as tramas, está interesado nas direccións MAC orixe e destino e no CRC.

FUNCIONAMENTO:
Mantén unha **Táboa de MACs** co formato:

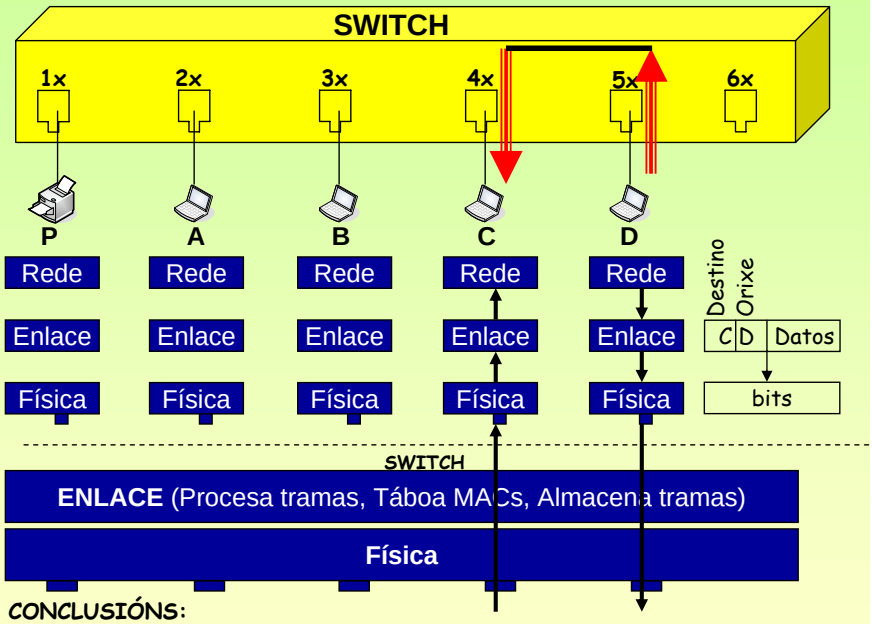
MAC	Porto	Tempo
P	1	10:00:12
B	3	10:00:13
D	5	10:00:27
A	2	10:01:05

Algoritmo de aprendizaxe cara atrás:
1.- Cando chega unha trama, apunta na táboa de MACs: **porto** de **entrada**, dirección **MAC** de quen a **envía** e o **hora** a que chegou.

2.- Mira o campo de **destino** da trama e consulta a táboa para saber porque porto está alcanzable esa dirección MAC.

Se non existe esa MAC (P.e. caso C) entón inunda, se existe envía polo porto axeitado.

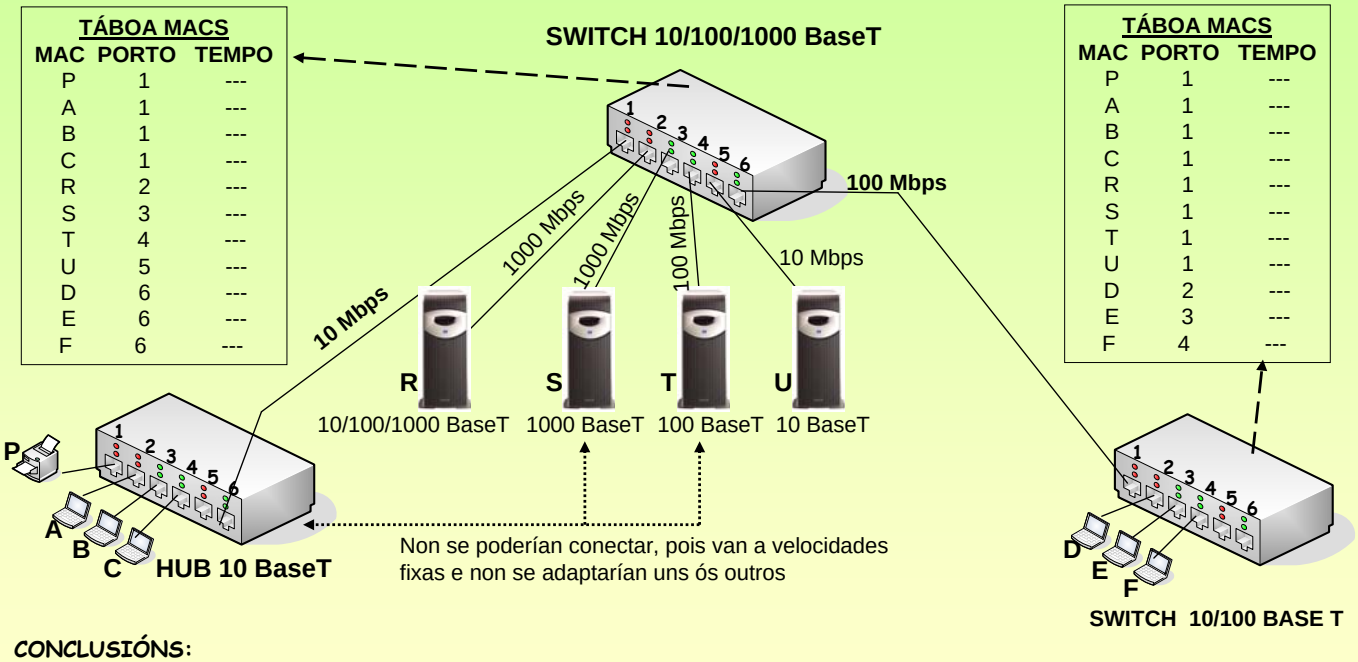
3.- Borra as entradas da táboa cunha antigüidade superior a X segundos



CONCLUSIÓNS:
1.- Cando un equipo tx, o switch recibe a trama e reenvía polo porto axeitado. Salvo que non estea o destino na táboa.
2.- **Colisións:** o switch almacena nunha memoria as tramas que chegan e logo procésaas. Dous hosts poderían estar enviando a outros dous sen molestarse.
3.- O **roubo** de información, precisa usar técnicas de hacker.
4.- O switch pode calcular o CRC da trama e comparalo co que ven na propia trama, se non coinciden descarta a trama

1.- Introducción – IEEE 802.3 - ETHERNET

☞ ETHERNET (10 BASET) – FAST-ETHERNET (100BASET) – GIGABIT (1000BASET)



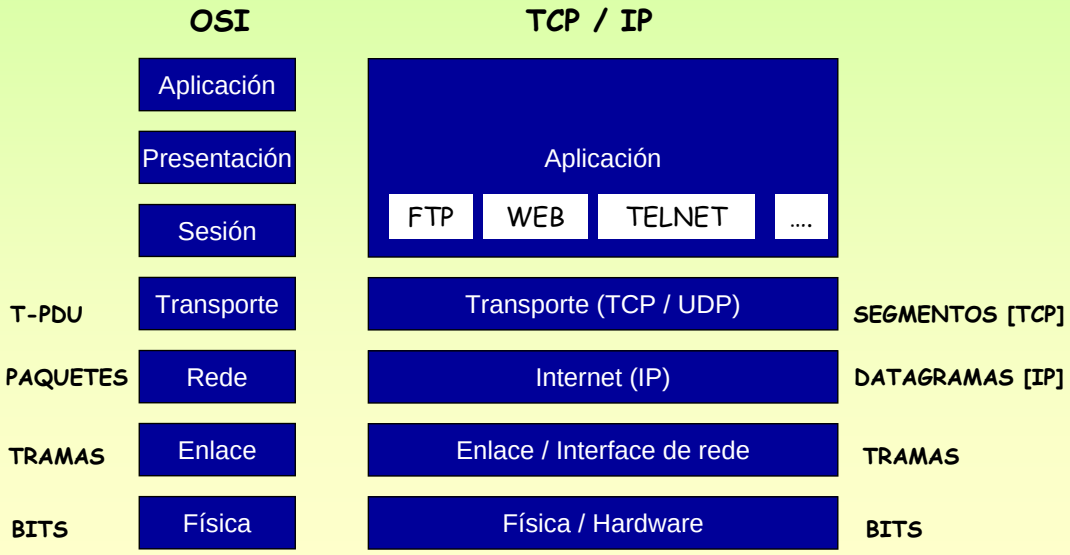
CONCLUSIÓNS:
1.- Un equipo que funcione a 10/100/1000 Mbps pódese conectar con calquera outro elemento.
2.- Un equipo que funcione p.ex. a 10 Mbps pódese conectar a outro que vaia a 10 Mbps ou a 10/100 Mbps ou a 10/100/1000 Mbps
3.- Dous equipos que poidan ir a 2 ou máis velocidades tratarán de ir á velocidade máis alta.

1.- Introducción – TCP / IP

☞ **ORIXES**

O grupo de protocolos TCP/IP foi creado pola ARPA (Axencia de Proxectos de Investigación Avanzada) pertencente ó departamento de defensa de EE.UU.

☞ **OSI vs. TCP/IP**



1.- Introducción – TCP / IP

IETF (The Internet Engineering Task Force) www.ietf.org

É unha grande comunidade e aberta de deseñadores de rede, operadores, vendedores, investigadores, etc involucrados na evolución da Arquitectura e Funcionalidade do Internet. Está organizado en áreas (p.e. Ruteo, transporte, seguridade, etc)

☞ **RFC (Request for comments, Petición de comentarios)**

Son documentos que proporcionan información sobre a Arquitectura e a a Funcionalidade de Internet. Algunhas son documentos oficiais do **IETF**, outros son borradores, propostas, tutoriais de aprendizaxe e finalmente outros son cómicos: RFC 2334 (HTCPCP) ou RFC 2549 (IP sobre pombas mensaxeiras con calidade de servizo)

Ademais do IETF estas pódense atopar en www.cse.ohio-state.edu/hypertext/information/rfc.html, www.rfc-editor.org.
En español está www.rfc-es.org onde se atopan as RFCs máis importantes traducidas.

☞ **Algunhas RFCs**

RFC	Obxectivo
768	UDP
791	IP
792	ICMP
793	TCP
821	SMTP
959	FTP
1034	DNS
1035	DNS
2131	DHCP
2136	DDNS

Etc.

1.- Introducción – TCP / IP

ENDEREZOS IP (Internet Protocol) - TIPOS

Cada equipo da rede que chegue ata o nivel 3 (rede) vai ter un enderezo IP.
Está composto po 32 bits (4 bytes) que se representan con 4 enteiros separados por puntos.
Exemplo: 0000 1010 . 0000 0011 . 0000 0101 . 0000 0110 (binario) → 10.3.5.6 (decimal)
Os 32 bits divídense en **dúas** partes: **Identificador de rede (net id):** indica o número de rede IP.
Identificador de equipo (host id): indica o número de equipo dentro da rede IP.

Valores característicos na parte de identificador de equipo:

- Poñer todo **ceros** na parte de equipo é para referirse á rede en si mesma (úsase par enrutar / encamiñar)
  10.0.0.0 (0000 1010 . 0000 0000 . 0000 0000 . 0000 0000) Fai referencia a toda a rede 10

- Poñer todo **uns** na parte de equipo – Multidifusión (Posto nunha dirección destino é para enviar a todos os da mesma rede IP)
  10.255.255.255 (0000 1010 . 1111 1111 . 1111 1111 . 1111 1111) Para transmitir a todos os da rede 10.0.0.0

DOUS equipos poderanse comunicar directamente se están na mesma rede IP, senón terán que usar intermediarios: **routers**

TIPO A

0	NET ID	IDENTIFICADOR DE EQUIPO (HOST ID)
	_ _ _ _ _ _ _ _	_ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _

1º ÍTEM: 0 – 127 **REDES:** $2^7 = 128$ **EQUIPOS:** $2^{24} - 2 = 16.777.214$
REDE PARA USO PRIVADO: 10.0.0.0 - 10.255.255.255 (1 sóa rede clase A - RFC 1989)

EXEMPLO: 95.3.20.2 **REDE:** 95.0.0.0 **EQUIPO:** 3.20.2 **MULTIDIFUSIÓN:** 95.255.255.255

TIPO B

10	IDENTIFICADOR DE REDE (NET ID)	IDENTIFICADOR DE EQUIPO (HOST ID)
	_ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _	_ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _

1º ÍTEM: 128 – 191 **REDES:** $2^{14} = 16.384$ **EQUIPOS:** $2^{16} - 2 = 65.534$
REDE PARA USO PRIVADO: 172.16.0.0 - 172.31.255.255 (16 redes clase B – RFC 1989)

EXEMPLO: 150.3.20.2 **REDE:** 150.3.0.0 **EQUIPO:** 20.2 **MULTIDIFUSIÓN:** 150.3.255.255

TIPO C

110	IDENTIFICADOR DE REDE (NET ID)	HOST ID
	_ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _	_ _ _ _ _ _ _ _

1º ÍTEM: 192 – 223 **REDES:** $2^{21} = 2.097.152$ **EQUIPOS:** $2^8 - 2 = 254$
REDE PARA USO PRIVADO: 192.168.0.0 - 192.168.255.255 (256 redes clase C – RFC 1989)

EXEMPLO: 192.3.20.2 **REDE:** 192.3.20.0 **EQUIPO:** 2 **MULTIDIFUSIÓN:** 192.3.20.255

1.- Introducción – TCP / IP

TIPOS ESPECIAIS DE IPS

As IPs privadas de cada clase úsanse para fogares, cibers, institucións, etc, que non queiran ter equipos con IPs reais en internet.

A rede 127.0.0.0 non se usa para asignar ós equipos. En concreto a IP 127.0.0.1 úsase para **loopback** (é o propio equipo).
Un equipo aínda que non teña tarxeta de rede sempre ten un IP asignada: 127.0.0.1
Tamén se coñece co nome de "**localhost**" (Explicado máis adiante)

DIFUSIÓN LIMITADA: IP de destino: 255.255.255.255. Úsase para difusión local, cando un equipo desexa enviar a tódolos equipos da súa rede. Úsana os clientes DHCP cando un equipo trata de obter unha dirección IP. (Explicado máis adiante)

DIFUSIÓN: Supoñer esta IP de destino: 10.255.255.255. Se é enviada, por exemplo, por 10.0.3.2 é o mesmo que o caso anterior. Se é enviada, por exemplo, por 11.0.3.4, ese paquete atravesará routers ata alcanzar a rede 10.0.0.0

En www.iana.org (Internet Assigned Numbers Authority) pódense atopar as distintas restricción sobre o uso de IPs.

TIPO D

1110	ENDEREZO DE MULTIDIFUSIÓN
	_ _ _ _ . _ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _

1º ÍTEM: 224 – 239

ÚSASE XERALMENTE PARA A DIFUSIÓN DE VÍDEO (UN **ÚNICO** EMISOR E VARIOS RECEPTORES).

TRÁTASE DE QUE O EMISOR SÓ EMITA UNHA SÓA VEZ E NON TANTAS COMO RECPTORES HAXA.

TIPO E

1111 10	RESERVADO PARA USO FUTURO
	_ _ . _ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _ . _ _ _ _ _ _ _ _

1º ÍTEM: 240 – 247

1.- Introducción – TCP / IP

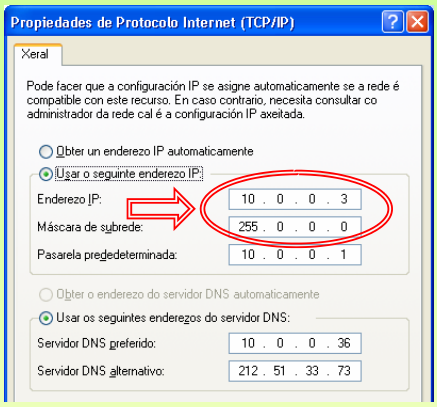
MÁSCARAS

Para determinar nunha dirección IP: ¿que parte é **rede?** e ¿que parte é **equipo?** úsase á máscara.

Está formada por 32 bits, que se organizan en 4 números enteiros

A parte da máscara na que hai **uns (1s)** corresponde coa parte de **rede IP** do enderezo IP.

Unha máscara é como a sombra dun enderezo IP. Se non se ten a máscara que acompaña a unha IP non se poderá determinar a parte de rede e a parte de equipo.



MÁSCARA ____ ____ . ____ ____ . ____ ____ . ____ ____

1º ÍTEM: 0 – 255 2º ÍTEM: 0 – 255 3º ÍTEM: 0 – 255 4º ÍTEM 0-255

MÁSCARA TIPO - A `1111 1111 . 0000 0000 . 0000 0000 . 0000 0000`

255 . 0 . 0 . 0

MÁSCARA TIPO - B `1111 1111 . 1111 1111 . 0000 0000 . 0000 0000`

255 . 255 . 0 . 0

MÁSCARA TIPO - C `1111 1111 . 1111 1111 . 1111 1111 . 0000 0000`

255 . 255 . 255 . 0

1.- Introducción – TCP / IP

MÁIS SOBRE MÁSCARAS

Outra forma de representar as máscaras é indicando o número de **1s** que posúe.

Exemplo: 10.4.5.6 / 8 (Indica que os 8 primeiros bits da máscara son **1s** e os 24 bits restantes **0s**)
A máscara equivalente é 255.0.0.0

O equipo sabe cal é a súa **rede-IP** ó facer un AND BINARIO do enderezo IP coa súa máscara.

Exemplo : 10 . 4.5.6 0000 1010 . 0000 0100 . 0000 0101 . 0000 0110
255.0.0.0 1111 1111 . 0000 0000 . 0000 0000 . 0000 0000 **AND BINARIO**
10 . 0.0.0 0000 1010 . 0000 0000 . 0000 0000 . 0000 0000

Estaríamos a falar da rede-IP 10.0.0.0 e do equipo 4.5.6 dentro desa rede - IP.

IMPORTANCIA DA MÁSCARA

En función da máscara unha dirección IP pode estar nunha rede IP ou noutra.

EXEMPLO:

10.3.2.1 / 8= 10. 3. 2. 1	10.3.2.1 / 16= 10.3.2.1	10.3.2.1 / 24 = 10.3.2.1
255.0.0.0	255.255.0.0	255.255.255.0
REDE: 10.0.0.0	REDE:10.3.0.0	REDE:10.3.2.0
EQUIPO: 3.2.1	EQUIPO: 2.1	EQUIPO:1

SUBREDES

O exemplo anterior é un claro exemplo de subrede, converteuse unha dirección de tipo A noutras de tipo B e tipo C.

Se unha empresa ten 20 departamentos e está interesada en que cada un deles estea nunha rede – IP distinta,
A empresa merca a IANA a rede IP de tipo B: **130.6.0.0.**
Se lle pon a tódolos equipos a máscara **255.255.0.0** tódolos equipos estarían na mesma rede-IP.

A solución pasa por facer subredes, pasar a IP anterior a outra de **tipo C**, iso conséguese coa máscara.
Se poñen a un departamento IPs na subrede **130.6.1.0 / 24** e a outro **130.6.2.0 / 24,** xa estarían en redes - IP distintas.

1.- Introducción – TCP / IP

☞ E REMATAMOS COAS MÁSCARAS

Desafortunadamente, non tódalas máscaras son /8, /16 ou /24 (esto é 255.0.0.0, 255.255.0.0, 255.255.255.0)
O seguinte exemplo mostra que os valores da máscara van dende /0 ata /32 (Estes 2 casos, en concreto, son casos especiais)

Exemplo: Tres equipos coas seguintes IPs: 10.1.4.6 / 23 (Máscara 255.255.254.0)
10.1.5.6 / 23 (Máscara 255.255.254.0)
10.1.6.6 / 23 (Máscara 255.255.254.0)

Faise o paso a binario:

10.1.4.6 /23	0000 1010 . 0000 0001 . 0000 010	0 . 0000 0110	
Máscara	1111 1111 . 1111 1111 . 1111 111	0 . 0000 0000	**AND BINARIO**
	0000 1010 . 0000 0001 . 0000 010	**0 . 0000 0000**	
10.1.5.6 /23	0000 1010 . 0000 0001 . 0000 010	1 . 0000 0110	
Máscara	1111 1111 . 1111 1111 . 1111 111	0 . 0000 0000	**AND BINARIO**
	0000 1010 . 0000 0001 . 0000 010	**0 . 0000 0000**	
10.1.6.6 /23	0000 1010 . 0000 0001 . 0000 011	0 . 0000 0110	
Máscara	1111 1111 . 1111 1111 . 1111 111	0 . 0000 0000	**AND BINARIO**
	0000 1010 . 0000 0001 . 0000 011	**0 . 0000 0000**	
	NET ID : 23 bits	HOST ID : 9 bits	

Os dous primeiros equipos pódense comunicar entre si, pois **están na mesma rede –IP**. Os primeiros 23 bits son iguais.
O terceiro equipo non se pode comunicar cos outros. Está nunha rede-IP distinta. Non coinciden os 23 primeiros bits.

Ollo co seguinte exemplo:

10.1.4.4 /30	0000 1010 . 0000 0001 . 0000 0100 .	0000 01	00 **Esta IP ten 0s na parte de equipo. Refírese á rede-IP**
10.1.4.5 /30	0000 1010 . 0000 0001 . 0000 0100 .	0000 01	01 **Esta IP pódeselle poñer a un equipo.**
10.1.4.6 /30	0000 1010 . 0000 0001 . 0000 0100 .	0000 01	10 **Esta IP pódeselle poñer a un equipo.**
10.1.4.7 /30	0000 1010 . 0000 0001 . 0000 0100 .	0000 01	11 **Esta IP ten 1s na parte de equipo. Multidifusión**
Máscara	1111 1111 . 1111 1111 . 1111 1111 .	1111 11	00
	NET ID : 30 bits		HOST ID : 2 bits

1.- Introducción – TCP / IP

☞ ENRUTAMENTO IP – AS ROTONDAS

As rotondas de tráfico serven para:
- **encamiñar o tráfico**. Grazas ás sinais que indican cara a onde están os destinos.
- **unir estradas de distintos tipos e velocidades**. Por exemplo, unha vía rápida cunha estrada corrente.

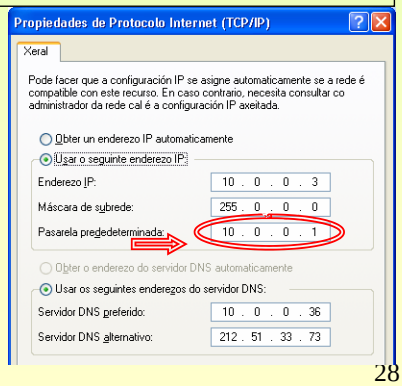
Un conductor ó chegar a unha rotonda encamiña o seu coche en función das sinais de dirección.

☞ ROUTERS / ENCAMIÑADORES / PORTA DE ENLACE / PASARELA

Un **router** actúa coma unha rotonda. A el chegan paquetes que serán encamiñados por unha ou outra liña en función da **táboa de encamiñamento**.

Un conductor para acadar o seu destino pode atravesar moitas rotondas.
Un datagrama / paquete para acadar o seu destino pode atravesar moitos routers.

Un ordenador que desexe enviar un datagrama a outro que non está na mesma rede-IP ca el, debe enviar ese paquete ó router.
Esta é a razón pola que se configura unha porta de enlace no propio equipo. **A porta de enlace estará na mesma rede que o Equipo.**



1.- Introducción – TCP / IP

☞ CONFIGURAR UN ROUTER: IPs

Obsérvese o seguinte exemplo:

- 4 Redes – IP . Dúas delas en subredes (Sala 3 e Sala 4)
- 2 Routers: **Router R:** une dúas redes IP.
  Router S: une tres redes IP.

Cada ordenador debe ter configurada unha porta de enlace á que enviar os paquetes que non vaian para a súa REDE – IP.
Ollar como **C e D** teñen configurada unha porta de enlace distinta, pero correctas. Poderían os dous ter a mesma

ROUTER R

ROUTER S

Eth 0: 10.0.0.1 /8 | Eth 1: 130.2.0.1 /16

SALA 1 – 10.0.0.0 /8

A — IP/MASK: 10.0.0.2 /8 — P. ENLACE: 10.0.0.1

B — IP/MASK: 10.0.0.3 /8 — P. ENLACE: 10.0.0.1

SALA 2 – 130.2.0.0 /16

C — IP/MASK: 130.2.0.2 /16 — P. ENLACE: 130.2.0.1

D — IP/MASK: 130.2.0.3 /16 — P. ENLACE: 130.2.0.4

SALA 3 – 11.1.0.0 /16

E — IP/MASK: 11.1.0.2 /16 — P. ENLACE: 11.1.0.1

F — IP/MASK: 11.1.0.3 /16 — P. ENLACE: 11.1.0.1

Eth 0: 130.2.0.4 /16 | Eth 1: 11.1.0.1 /16

Eth 2: 11.2.0.1 /16

SALA 4 – 11.2.0.0 /16

G — IP/MASK: 11.2.0.2 /16 — P. ENLACE: 11.2.0.1

H — IP/MASK: 11.2.0.3 /16 — P. ENLACE: 11.2.0.1

Débese entender este router como dúas metades, cada unha nunha rede-IP distinta.

Deste xeito o Router pódese comunicar directamente con calquera dos equipos da Sala 1 pois ten unha IP nesa mesma rede – IP e cos da sala 2, polo mesmo.

Ó mesmo tempo pódese comunicar co router S pois os dous están na rede 130.2.0.0 / 16

1.- Introducción – TCP / IP

☞ CONFIGURAR UN ROUTER: O equipo A vaille enviar un paquete ó equipo G

ROUTER R

Eth 0: 10.0.0.1 /8 | Eth 1: 130.2.0.1 /16

ROUTER S

SALA 1 – 10.0.0.0 /8

A — IP/MASK: 10.0.0.2 /8 — P. ENLACE: 10.0.0.1

B — IP/MASK: 10.0.0.3 /8 — P. ENLACE: 10.0.0.1

SALA 2 – 130.2.0.0 /16

C — IP/MASK: 130.2.0.2 /16 — P. ENLACE: 130.2.0.1

D — IP/MASK: 130.2.0.3 /16 — P. ENLACE: 130.2.0.4

SALA 3 – 11.1.0.0 /16

E — IP/MASK: 11.1.0.2 /16 — P. ENLACE: 11.1.0.1

F — IP/MASK: 11.1.0.3 /16 — P. ENLACE: 11.1.0.1

Eth 0: 130.2.0.4 /16 | Eth 1: 11.1.0.1 /16

Eth 2: 11.2.0.1 /16

SALA 4 – 11.2.0.0 /16

G — IP/MASK: 11.2.0.2 /16 — P. ENLACE: 11.2.0.1

H — IP/MASK: 11.2.0.3 /16 — P. ENLACE: 11.2.0.1

Se o **equipo A** se quere comunicar co **equipo G** debe enviar o seguinte datagrama:

11.2.0.2	10.0.0.2	DATOS A ENVIAR
Destino	Orixe	

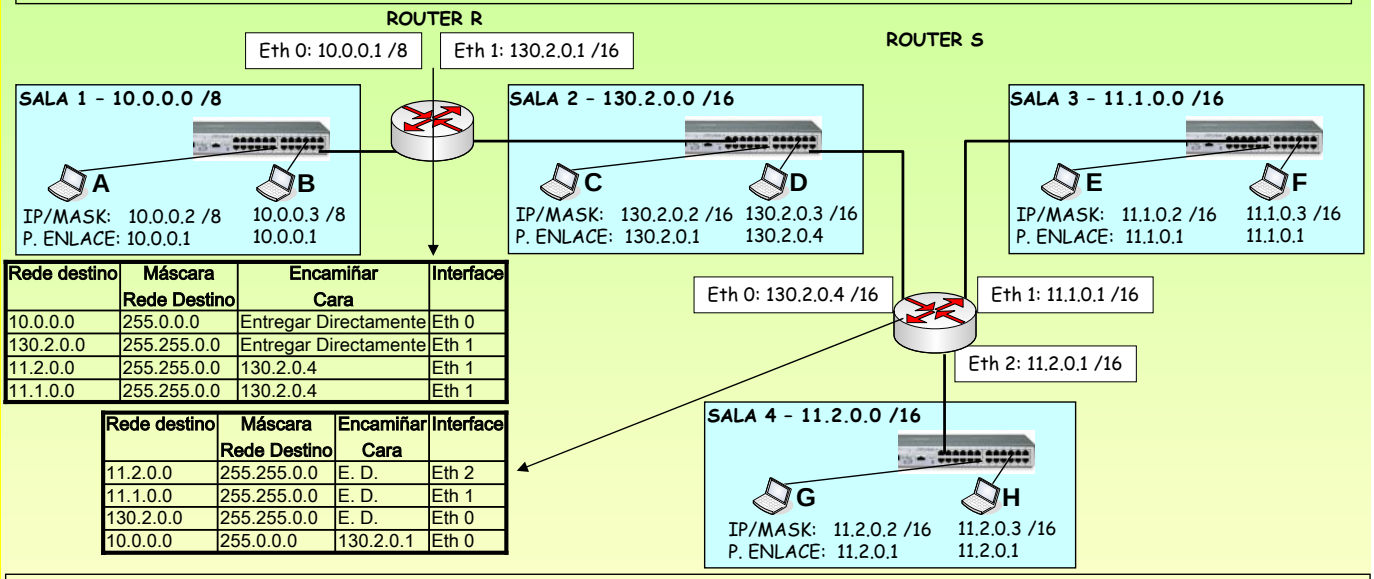
O equipo A precisa saber se o DESTINATARIO está na mesma rede-IP ca el ou en distinta.
- No primeiro caso **A** enviaría o paquete dentro da súa rede.
- No segundo caso enviará o paquete ó router que teña configurado como porta de enlace, para que este o encamiñe.

O equipo **A** fai un AND da **súa** máscara coas IPs ORIXE e DESTINO do paquete, deste xeito **A** saberá se destino e orixe están na mesma rede IP:

	11 .2.0.2	10 .0.0.2
Máscara do orixe (A)	255.0.0.0	255.0.0.0 &
	11 .0.0.0	10 .0.0.0

O **equipo A** chega á conclusión de que o **DESTINATARIO** non está na mesma rede ca el, senón terían que coincidir os resultados.
O **equipo A** decide, entón, enviar o paquete á súa porta de enlace que é 10.0.0.1 (Router R) e que el o **encamiñe**.
O **equipo A** pode comunicarse co **Router R** porque, este por un dos lados está na mesma rede ca el.

☞ CONFIGURAR UN ROUTER: TÁBOAS DE ENCAMIÑAMENTO (I)



Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
10.0.0.0	255.0.0.0	Entregar Directamente	Eth 0
130.2.0.0	255.255.0.0	Entregar Directamente	Eth 1
11.2.0.0	255.255.0.0	130.2.0.4	Eth 1
11.1.0.0	255.255.0.0	130.2.0.4	Eth 1

Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
11.2.0.0	255.255.0.0	E. D.	Eth 2
11.1.0.0	255.255.0.0	E. D.	Eth 1
130.2.0.0	255.255.0.0	E. D.	Eth 0
10.0.0.0	255.0.0.0	130.2.0.1	Eth 0

O **equipo A** decidiu enviar o anterior paquete ó router. Este fará o que fai un carteiro, mirará a dirección de destino. Neste caso: 11.2.0.2
O router realiza una AND da IP DESTINO coa primeira máscara da táboa de ruteo e mira se coincide coa columna **Rede Destino.**

- **SE COINCIDE:** envía o paquete a onde indique a columna **Encamiñar CARA**, polo **interface** indicado.
- **SE NON COINCIDE:** realiza a mesma operación do AND coa segunda entrada da táboa. E así ata coincidir ou rematar.

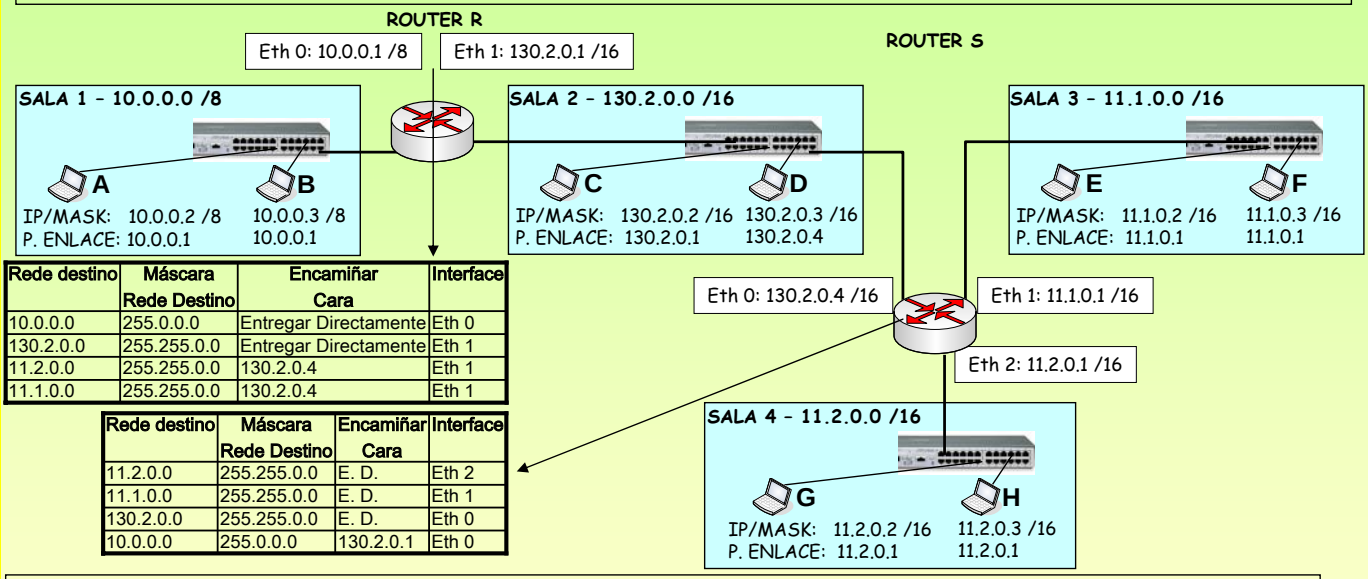
NESTE CASO: (Destino) 11.2.0.2 & (1ª Máscara) 255.0.0.0 = 11.0.0.0 non coincide con 10.0.0.0 (da primeira fila)
11.2.0.2 & 255.255.0.0 = 11.2.0.0 non coincide con 130.2.0.0 (da segunda fila)
11.2.0.2 & 255.255.0.0 = 11.2.0.0 **SI** coincide con 11.2.0.0. Enviar paquete a : 130.2.0.4

OSI – TCP/IP

1.- Introducción – TCP / IP

1. - Introducción

☞ CONFIGURAR UN ROUTER: TÁBOAS DE ENCAMIÑAMENTO (II)



Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
10.0.0.0	255.0.0.0	Entregar Directamente	Eth 0
130.2.0.0	255.255.0.0	Entregar Directamente	Eth 1
11.2.0.0	255.255.0.0	130.2.0.4	Eth 1
11.1.0.0	255.255.0.0	130.2.0.4	Eth 1

Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
11.2.0.0	255.255.0.0	E. D.	Eth 2
11.1.0.0	255.255.0.0	E. D.	Eth 1
130.2.0.0	255.255.0.0	E. D.	Eth 0
10.0.0.0	255.0.0.0	130.2.0.1	Eth 0

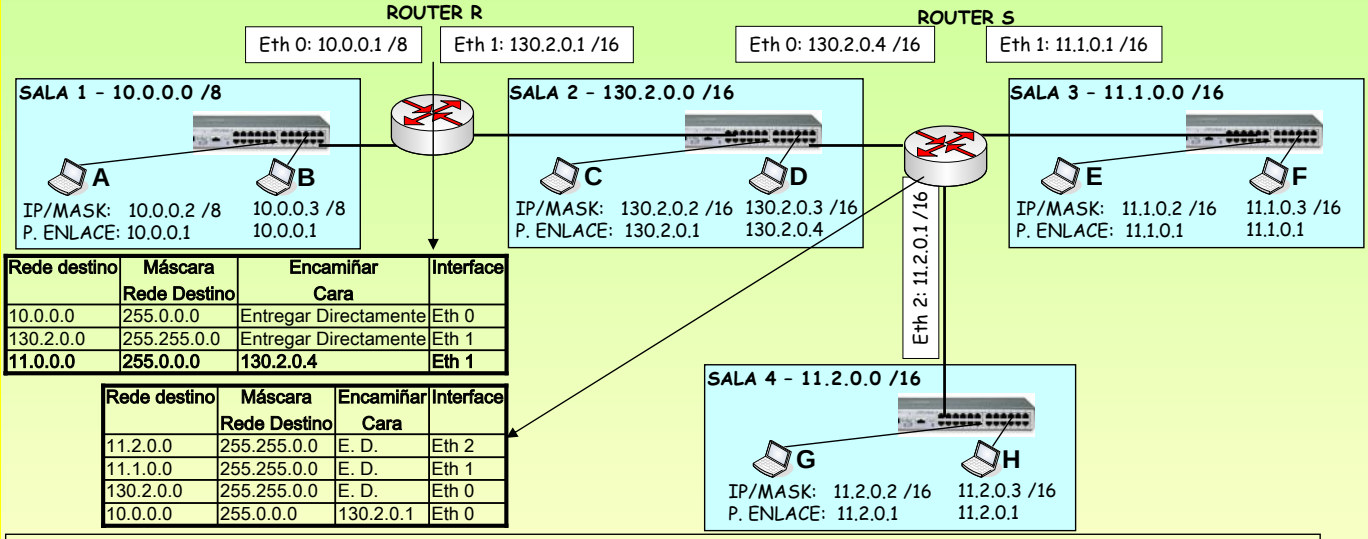
Un router está interesado no DESTINO dos paquetes que lle chegan, ó igual que as oficinas de correos.

Seguindo co exemplo anterior, agora, o paquete teno o Router S. Este realizará o mesmo proceso que o router R.
Neste caso a primeira entrada da táboa xa lle indica que ese paquete teno que **entregar directamente** polo interface Eth 2.

ENTREGAR DIRECTAMENTE: cando unha carta chega á última oficina de correos, só resta que o carteiro colla a Vespa e leve a carta ó seu destinatario real.
Neste caso igual, ó router só lle resta mandarlla ó seu destinatario final.

1.- Introducción – TCP / IP

☞ CONFIGURAR UN ROUTER: TÁBOAS DE ENCAMIÑAMENTO (II)



Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
10.0.0.0	255.0.0.0	Entregar Directamente	Eth 0
130.2.0.0	255.255.0.0	Entregar Directamente	Eth 1
11.0.0.0	**255.0.0.0**	**130.2.0.4**	**Eth 1**

Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
11.2.0.0	255.255.0.0	E. D.	Eth 2
11.1.0.0	255.255.0.0	E. D.	Eth 1
130.2.0.0	255.255.0.0	E. D.	Eth 0
10.0.0.0	255.0.0.0	130.2.0.1	Eth 0

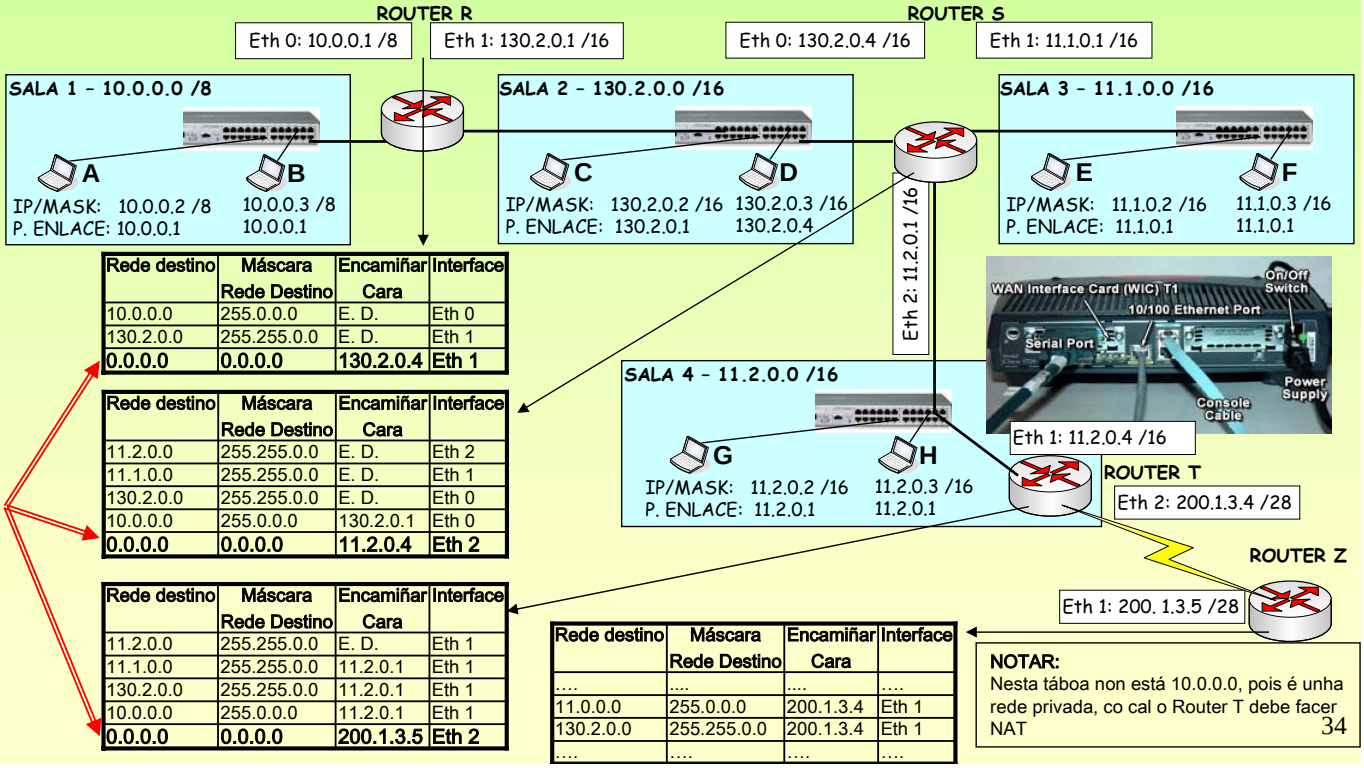
Débese desprender que a dirección IP Destino do paquete non se modifica, ó igual que non se modifica nunha carta, senón non se podería encamiñar ata o seu destino final.

Se a rede 11.0.0.0 é toda da empresa. E se esta é a configuración final da rede, obsérvese como se podería modificar a táboa de encamiñamento do ROUTER R.

Sácanse as dúas entradas 11.2.0.0 /16 e 11.1.0.0 /16 e substitúese por unha soa entrada 11.**0**.0.0**/8**. Pois tanto a subrede **11.1**.0.0 como a **11.2**.0.0 teñen en común rede **11**.0.0.0 na súa totalidade.
Será o router S quen faga as distincións entre unha subrede e a outra.

1.- Introducción – TCP / IP

☞ CONFIGURAR UN ROUTER: TÁBOAS DE ENCAMIÑAMENTO (III)
Conectados a INTERNET



Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
10.0.0.0	255.0.0.0	E. D.	Eth 0
130.2.0.0	255.255.0.0	E. D.	Eth 1
0.0.0.0	**0.0.0.0**	**130.2.0.4**	**Eth 1**

Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
11.2.0.0	255.255.0.0	E. D.	Eth 2
11.1.0.0	255.255.0.0	E. D.	Eth 1
130.2.0.0	255.255.0.0	E. D.	Eth 0
10.0.0.0	255.0.0.0	130.2.0.1	Eth 0
0.0.0.0	**0.0.0.0**	**11.2.0.4**	**Eth 2**

Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
11.2.0.0	255.255.0.0	E. D.	Eth 1
11.1.0.0	255.255.0.0	11.2.0.1	Eth 1
130.2.0.0	255.255.0.0	11.2.0.1	Eth 1
10.0.0.0	255.0.0.0	11.2.0.1	Eth 1
0.0.0.0	**0.0.0.0**	**200.1.3.5**	**Eth 2**

Rede destino	Máscara Rede Destino	Encamiñar Cara	Interface
....	....	....	....
11.0.0.0	255.0.0.0	200.1.3.4	Eth 1
130.2.0.0	255.255.0.0	200.1.3.4	Eth 1
....	....	....	....

NOTAR:
Nesta táboa non está 10.0.0.0, pois é unha rede privada, co cal o Router T debe facer NAT

☞ROUTER R:

O **router R** pode entregar paquetes para a **SALA 1** e a **SALA 2**, se os paquetes van para calquera outro sitio terá que enviarllo ó **router S** e que el se encargue de encamiñalos.

A última entrada da Táboa de Encamiñamento é a que indica que cando chegue un paquete que non vaia para unha desas salas llo envíe ó router S.
Deste xeito non se teñen que contemplar nunha táboa de encamiñamento tódolos posibles destinos (tanto da intranet como de internet, que sería imposible).

EXEMPLO: pénsese que ó **router R** chegaron tres paquetes cos seguintes destinos:

11.1.0.2 (Sala 3)
213.4.130.210 (www.terra.es)

En calquera dos dous casos terá que enviar ese paquete ó router S. Realicemos a proceso do router coa segunda IP.

IP DESTINO		MÁSCARA		RESULTAO		1ª COLUMNA	
213.4.130.210	&	255.0.0.0	=	213.0.0.0	!=	10.0.0.0	→ Seguir co proceso e operar coa 2ª entrada
213.4.130.210	&	255.255.0.0	=	213.4.0.0	!=	130.2.0.0	→ Seguir co proceso e operar coa 3ª entrada
213.4.130.210	**&**	**0.0.0.0**	**=**	**0.0.0.0**	**=**	**0.0.0.0**	**→ Encamiñar cara 130.2.0.4**

CONCLUSIÓN: como calquera IP AND 0.0.0.0 vai dar 0.0.0.0 esa entrada sempre se debe poñer ó final da táboa.
Os demais routers tamén deben ter a entrada 0.0.0.0.

☞ ROUTER T: o router da empresa para saír a internet a través dun ISP

Este router une dúas entidades. Cada unha encárgase de configurar a súa "metade". A empresa non pode condicionar a IP polo lado do Provedor de Servizos de Internet (ISP). Esa función correspóndelle ó ISP para adaptalo á súa rede IP.

☞ ROUTER Z: o router do ISP que encamiña cara á empresa.

Este router configúrao totalmente o ISP, pero nel ten que ter entradas que axuden ós paquetes a chegar ata as dúas redes-IP da empresa.
Dinse dúas redes pois a empresa mercou a 130.2.0.0 /16 e a 11.0.0.0/8 aínda que esta última estea convertida en subredes.
Neste caso as subredes son algo interno da empresa que no exterior non o van saber. No exterior todo é 11.0.0.0 /8

OSI – TCP/IP

1.- Introducción – TCP / IP

1.- Introducción

☞ ALGORITMOS DE ENCAMIÑAMENTO

Indican a forma en que se constrúe a táboa de encamiñamento dun router

☞ NON ADAPTATIVOS (ESTÁTICOS)

Non se adaptan ás situacións cambiantes da rede (unha liña saturada, unha liña que cae, etc). Cando chegen varios paquetes para o mesmo destino sempre os vai encamiñar polo mesmo sitio.

Hai que configuralos manualmente.
Equivalen a unha rotonda na que só hai sinais indicativas e que non sabe en que situación se atopan cada unha das saídas.

☞ ADAPTATIVOS

Os routers que usan algoritmos adaptativos adáptanse ós cambios e situacións da rede. Existen tres tipos:

CENTRALIZADO:

Equivale á sala de control de tráfico dunha cidade onde teñen a información do que está a pasar en cada unha das rotondas, que rúas están saturadas, cales cortadas, etc. Con toda esa información elaboran as accións que deben levar a cabo cada un dos Gardas que están nas rotondas.

Existe un nó central ó que cada router lle envía información (cal é a liña máis solicitada, de onde lle veñen paquetes devoltos, se ten enlace cos demais routers, etc). Con esa información o nó elabora a táboa de cada router e logo envíalla. Existen problemas: uns routers terán as táboas antes que outros, esas táboas son paquetes competindo con outros na rede.

ILLADOS:

Equivale a poñer un GARDA en cada rotonda e que este dirixa o tráfico como lle apeteza sen ter en conta nada de nada, nin se está saturada unha saída, se hai un incidente, etc.

Exemplo: PATACA QUENTE**:** Chega un paquete, desfaise del tan pronto como poida e por calquera liña.

DISTRIBUÍDOS:

Equivale a ter Gardas nas rotondas pero cada un comunícase cos GARDAS das rotondas próximas a el, deste xeito trata de tomar as decisións adaptándose ó que pasa ó seu arredor.

1.- Introducción – TCP / IP

☞ **COMANDOS**
Windows: ROUTE

C:\WINDOWS\System32\CMD.exe

```
L:\>route print
===========================================================================
ILista de interfaces
0x1 ........................... MS TCP Loopback interface
0x4 ...00 0b 6a 2a 74 9a ...... VIA VT6102 Rhine II Fast Ethernet Adapter - Mini
puerto del administrador de paquetes
===========================================================================
===========================================================================
Rutas activas:
Destino de red        Máscara de red   Puerta de acceso   Interfaz  Métrica
          0.0.0.0          0.0.0.0         10.0.0.1        10.0.0.5       20
         10.0.0.0        255.0.0.0         10.0.0.5        10.0.0.5       20
         10.0.0.5  255.255.255.255        127.0.0.1       127.0.0.1       20
   10.255.255.255  255.255.255.255         10.0.0.5        10.0.0.5       20
        127.0.0.0        255.0.0.0        127.0.0.1       127.0.0.1        1
        224.0.0.0        240.0.0.0         10.0.0.5        10.0.0.5       20
  255.255.255.255  255.255.255.255         10.0.0.5        10.0.0.5        1
Puerta de enlace predeterminada:         10.0.0.1
===========================================================================
Rutas persistentes:
  ninguno

L:\>
```

C:\WINDOWS\System32\CMD.exe

```
L:\>ROUTE

Manipula tablas de enrutamiento de red.

ROUTE [-f] [-p] [comando [destino] [MASK
      [METRIC métrica]  [IF interfaz]

  -f                Borra las tablas de
                    de puerta de enlace,
                    comandos, se borrarán las tablas antes de ejecutarse el
                    comando.
  -p                Cuando se usa con el comando ADD, hace una ruta
                    persistente en los inicios del sistema. De manera
                    predeterminada, las rutas no se conservan cuando se
                    reinicia el sistema. Se pasa por alto para todos los
                    demás comandos, que siempre afectan a las rutas
                    persistentes apropiadas. Esta opción no puede utilizarse
                    en Windows 95.
  comando           Uno de los siguientes:
                      PRINT     Imprime una ruta
                      ADD       Agrega una ruta
                      DELETE    Elimina una ruta
                      CHANGE    Modifica una ruta existente
  destino           Especifica el host.
  MASK              Especifica que el siguiente parámetro es el valor de
                    'máscara_red'.
  máscara_red       Especifica un valor de máscara de subred para esta
                    entrada de ruta.
                    Si no se especifica, se usa de forma predeterminada el
                    valor 255.255.255.255.
  puerta_enlace     Especifica la puerta de enlace.
  interfaz          El número de interfaz para la ruta especificada.
  METRIC            Especifica la métrica; por ejemplo, costo para el destino.
```


1.- Introducción – TCP / IP

root@linuxp: /root – Terminal – Konsole

Sesión Editar Vista Marcadores Preferencias Ayuda

```
[root@linuxp root]# route --help
Usage: route [-nNvee] [-FC] [<AF>]           List kernel routing tables
       route [-v] [-FC] {add|del|flush} ...  Modify routing table for AF.

       route {-h|--help} [<AF>]              Detailed usage syntax for specified AF

       route {-V|--version}                  Display version/author and exit.

        -v, --verbose            be verbose
        -n, --numeric            don't resolve names
        -e, --extend             display other/more information
        -F, --fib                display Forwarding Information Base (default)
        -C, --cache              display routing cache instead of FIB

  <AF>=Use '-A <af>' or '--<af>'; default: inet
  List of possible address families (which support routing):
    inet (DARPA Internet) inet6 (IPv6) ax25 (AMPR AX.25)
    netrom (AMPR NET/ROM) ipx (Novell IPX) ddp (Appletalk DDP)
    x25 (CCITT X.25)
[root@linuxp root]#
```

☞ **COMANDOS**
Linux: route

root@linuxp: /root – Terminal – Konsole

Sesión Editar Vista Marcadores Preferencias Ayuda

```
[root@linuxp root]# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.0        *               255.0.0.0       U     0      0        0 eth0
127.0.0.0       *               255.0.0.0       U     0      0        0 lo
default         10.0.0.1        0.0.0.0         UG    0      0        0 eth0
[root@linuxp root]#
```

1.- Introducción

1.- Introducción – TCP / IP

☞ MÁIS TÁBOAS - CACHE ARP (I) (a ligazón do nivel IP co nivel de enlace)

EXEMPLO: O **HOST A** desexa enviar un paquete ó **HOST B.** (No gráfico débense seguir os números. Supoñer que as letras A, B, J son as MACs dos Hosts)

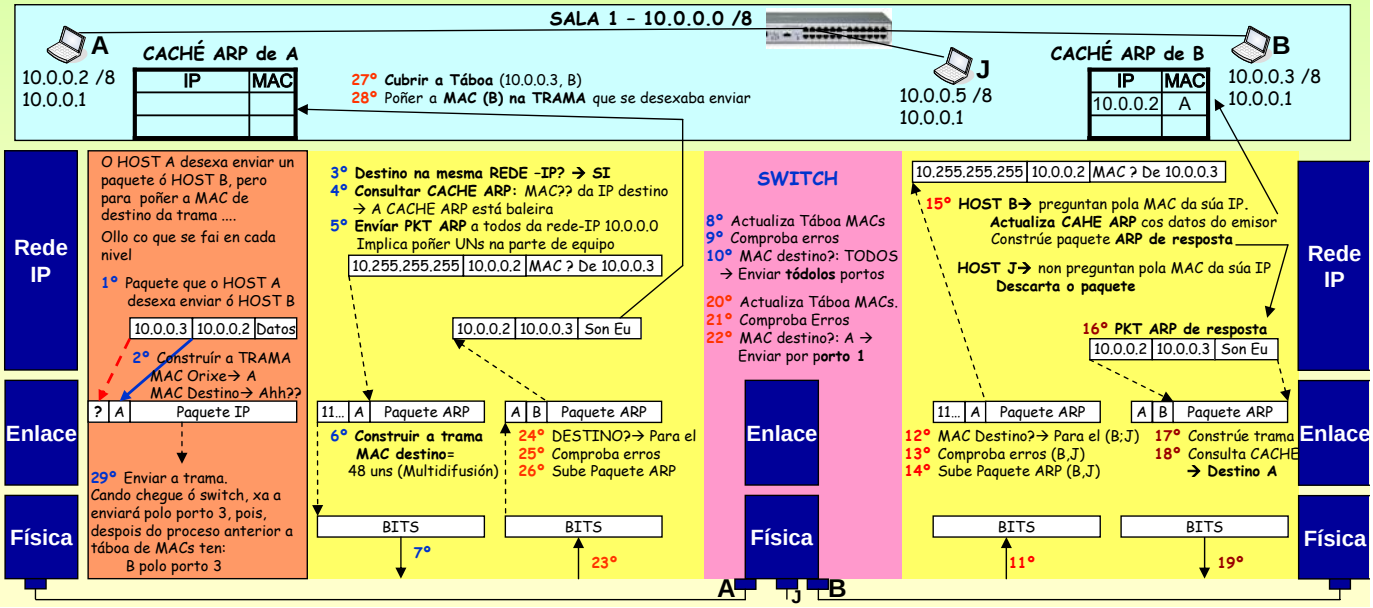
NIVEL IP: constrúe o datagrama cos **enderezos** orixe (10.0.0.2) e destino (10.0.0.3) e o campo de **datos.** Comproba se o destino está na mesma rede IP

NIVEL ENLACE: constrúe á trama, pero ¿Cal é a dirección MAC do destino?. Para achala usa o **Address Resolution Protocol (ARP)**

ARP: Cada equipo almacena en memoria unha táboa (**CACHE ARP**) que asocia IPs con MACs. Para construír esa táboa usa o Protocolo de Resolución de Enderezos (ARP). O protocolo ARP está na capa de REDE, no nivel 3.

Consiste en enviar a tódolos equipos da LAN a seguinte pregunta: **¿Pódeme dicir o ordenador con IP X.Y.Z.T cal é a súa MAC?**

Esta pregunta recibiríana tódolos equipos da LAN e só responderá o afectado, coa resposta imos cubrindo os campos da táboa para futuras ocasións. Ó mesmo tempo o ordenador afectado rexistra na súa CACHE ARP a IP e MAC de que fixo a petición.



1.- Introducción

1.- Introducción – TCP / IP

☞ MÁIS TÁBOAS - CACHE ARP (II)

EXEMPLO: Agora o **HOST A** desexa enviar un paquete ó **HOST D.** Pero para chegar ó HOST D temos que pasar antes polo Router R.

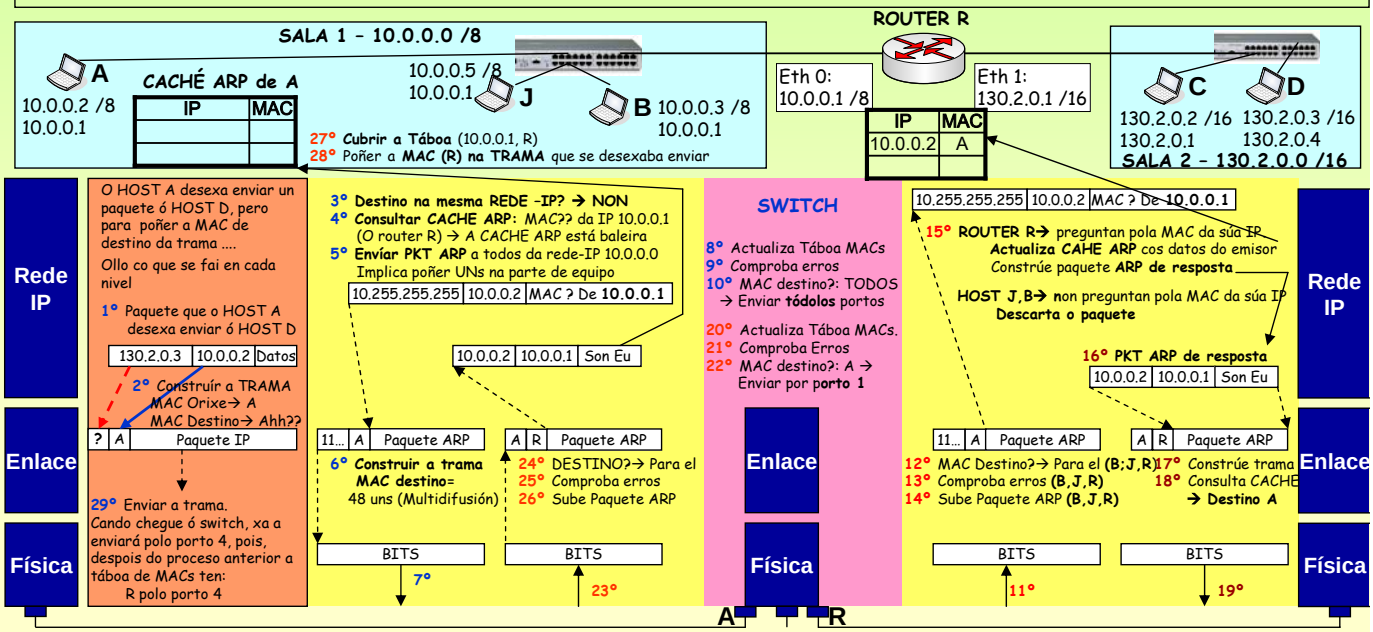
NIVEL IP: constrúe o datagrama cos **enderezos** orixe (10.0.0.2) e destino (130.2.0.3) e o campo de **datos.** Comproba se o destino está na mesma rede IP **É AQUÍ,** onde radica a diferencia co caso anterior. O **host A** tenlle que enviar o paquete ó Router para que el o encamiñe, co cal no nivel 2 a MAC que ten que achar é a do **ROUTER R** e non a do host D. **OBSERVAR OS PASO 1,3,4,5,27 o RESTO E SEMELLANTE.**

NIVEL ENLACE: constrúe a trama, pero ¿Cal é a dirección MAC do ROUTER R (10.0.0.1), NON do DESTINO REAL?.

ARP: Os routers tamén teñen a táboa CACHE ARP, pero neste caso terá IPs e MACs das redes que una por cada interface.

O host A realizará o mesmo proceso que no caso anterior só que a MAC que ten que calcular é a da porta de enlace.

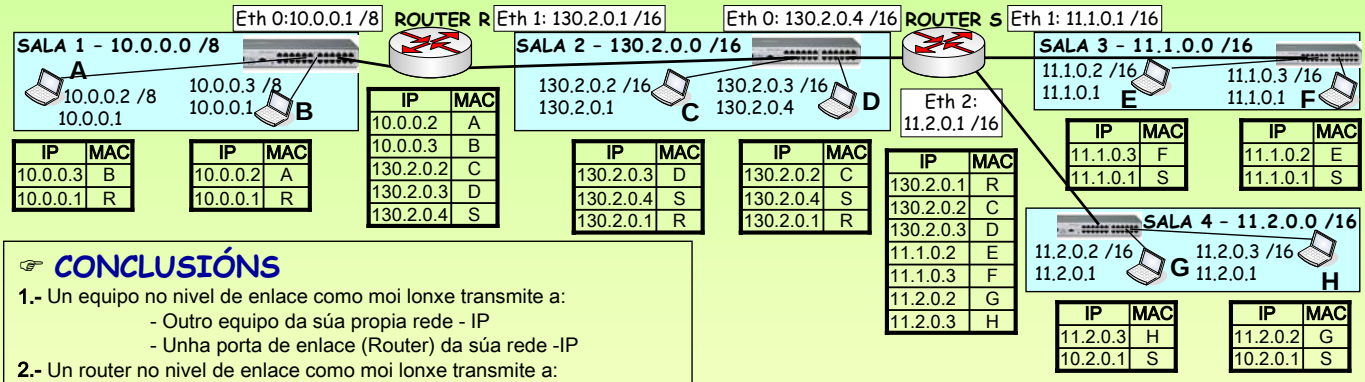
Unha vez que o HOST A averigue a MAC do router R enviaralle a trama a este. Logo, o router terá que facer todo o proceso pero cara á SALA 2.



1.- Introducción – TCP / IP

EXEMPLO - TÁBOAS CACHE ARP (III)

As táboas constrúense dinamicamente. Aquelas entradas na táboa que pasado un tempo non se usen vanse borrando.
No seguinte exemplo suponse que tódolos equipos se comunicaron con todos. As súas táboas serían:



IP	MAC
10.0.0.3	B
10.0.0.1	R

IP	MAC
10.0.0.2	A
10.0.0.1	R

IP	MAC
10.0.0.2	A
10.0.0.3	B
130.2.0.2	C
130.2.0.3	D
130.2.0.4	S

IP	MAC
130.2.0.3	D
130.2.0.4	S
130.2.0.1	R

IP	MAC
130.2.0.2	C
130.2.0.4	S
130.2.0.1	R

IP	MAC
130.2.0.1	R
130.2.0.2	C
130.2.0.3	D
11.1.0.2	E
11.1.0.3	F
11.2.0.2	G
11.2.0.3	H

IP	MAC
11.1.0.3	F
11.1.0.1	S

IP	MAC
11.1.0.2	E
11.1.0.1	S

IP	MAC
11.2.0.3	H
10.2.0.1	S

IP	MAC
11.2.0.2	G
10.2.0.1	S

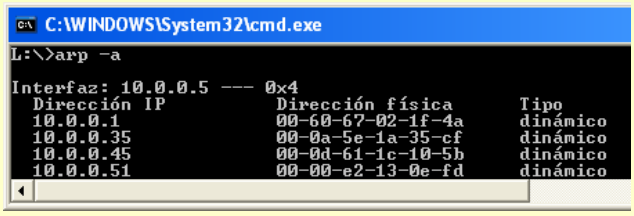
CONCLUSIÓNS

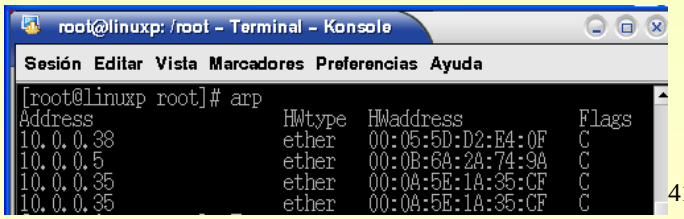
1.- Un equipo no nivel de enlace como moi lonxe transmite a:
- Outro equipo da súa propia rede - IP
- Unha porta de enlace (Router) da súa rede -IP

2.- Un router no nivel de enlace como moi lonxe transmite a:
Outro router da súa mesma rede-IP.
Un equipo de calquera das redes-IP que interconecta.

COMANDOS

COMANDOS: co comando **arp** (Linux / Windows) podemos traballar coa táboa CACHE ARP





1.- Introducción – TCP / IP

IP (Internet Protocol)

DATAGRAMAS: paquetes nos que se divide unha mensaxe e que se envían usando un **servizo non orientado á conexión**.

O nivel IP especifica o formato dos paquetes do nivel de rede, chamados **datagramas**,
Supón unha subrede (elementos de comunicacións entre orixe e destino reais) moi fiable pois fíase de que os paquetes van chegar ó destino.
O datagrama pode fragmentarse noutros máis pequenos se ten que atravesar redes con MTU (Campo de datos da trama) máis pequena.
O tamaño máximo do datagrama é de 64 KBytes. Este divídese en dúas partes CABECEIRA e DATOS

	0 1 2 3 4 5 6 7	8 9 10 11 12 13 14 15	16 17 18 19 20 21 22 23	24 25 26 27 28 29 30 31
Encabezado	Versión	HLEN	Tipo de Servizo (ToS)	Lonxitude Total
	Identificación		Bandeiras	Desprazamento do fragmento
	Tempo de Vida (TTL)		Protocolo	Control de erros da cabeceira (Header CheckSum)
	ENDEREZO IP ORIXE			
	ENDEREZO IP DESTINO			
	OPCIÓNS IP (Se hai algunha)			
	DATOS ...			

VERSION:	Versión do protocolo IP coa que se creou o datagrama. Versións actuais (IPv4 para enderezos de 32 bits)
HLEN:	Lonxitude da cabeceira medida en palabras de 32 bits (1 palabra de 32 bits é igual a unha fila do debuxo) O encabezado común, sen opcións mide 5 (5 filas, 5 palabras de 32 bits). Esto é 5x4= 20 bytes.
LONXITUDE TOTAL:	Medido en Bytes, inclúe os bytes da cabeceira e dos datos. O campo ten 16 bits → 2^{16}=65.535 octetos (64 KB)
TIPO DE SERVIZO:	Para especificar a **prioridade** do datagrama, **fiabilidade**, **retardo**... Os routers non fan moito caso a este campo.
TEMPO DE VIDA:	(Time to live) Especifica o tempo en segundos que o datagrama pode estar na rede. Ó pasar polos routers, estes van decrecendo este valor. Se o seu tempo concluíu e non chegou ó destino os routers elimínano.
PROTOCOLO:	Que protocolo de alto nivel creou o **datagrama**. (TCP ou UDP).
CHECKSUM:	Realiza unha serie de complementos a un coa cabeceira e o resultado pono neste campo, para no receptor comprobar que a cabeceira chegou correctamente.
ENDERZOS IP:	Conteñen as direccións IP orixe do paquete e destino do paquete.
OPCIÓNS:	Úsase para probas de rede e depuración (Rexistrar rutas, etc). Como máximo poden ser 10 palabras de 32 b=40B
DATOS:	Contén bytes que se corresponden a un **segmento** (Unidade de datos que intercambian entidades de transporte)

1.- Introducción – TCP / IP

☞ IP (Internet Protocol) – A fragmentación: (Maximun Transfer Unit) (I)

Un emisor debe pasar un datagrama do nivel 3 ó nivel 2. Isto é debe meter o datagrama no campo de datos dunha TRAMA.

Pero dependendo da especificación que se use no nivel 2 o campo de datos terá un tamaño ou outro, este tamaño coñecese como **MTU**.

Ethernet (IEEE 802.3):	1.500 Bytes	Token Bus (IEEE 802.4):	8.174 Bytes
Token Ring (IEEE 802.5):	ilimitado	FDDI:	ilimitado
ATM (ATM sobre ADSL):	48 bytes	FRAME RELAY:	ilimitado

Co cal se se ten un datagrama de tamaño maior que o campo de datos da trama, terase que fragmentar o datagrama noutros máis pequenos.

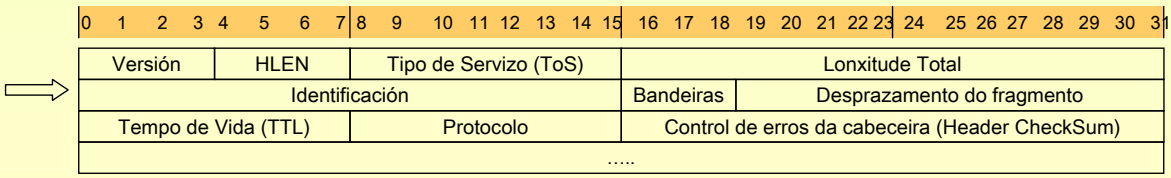
IDENTIFICACIÓN: identifica o número de paquete, se este se fragmenta, cada fragmento levará a mesma IDENTIFICACIÓN. Así o receptor saberá que fragmentos se corresponden a cada paquete orixinal.

BANDEIRAS (FLAGS): indica se o paquete de pode ou non fragmentar. No caso de que se poida, indica se é un fragmento intermedio ou último

DESPRAZAMENTO: Cando se fragmenta un paquete, cada fragmento leva un anaco do datagrama orixinal. Este campo indica que posición ocupan os bytes, que leva un fragmento, no datagrama orixinal. (Enténdase como se fose a numeración de cada fragamento).

ONDE SE FRAGMENTA?: Un datagrama pódese fragmentar no extremo emisor ou en calquera dos routers intermedios, sempre e cando o esixa a MTU da rede a atravesar.

ONDE SE REENSAMBLA?: Só, só, só no **EXTREMO RECEPTOR FINAL**. Pois cada fragmento puido ir por camiños distintos ata chegar ó receptor final, así pois será o que reciba tódolos anacos nos que se dividiron os fragmentos.



1.- Introducción – TCP / IP

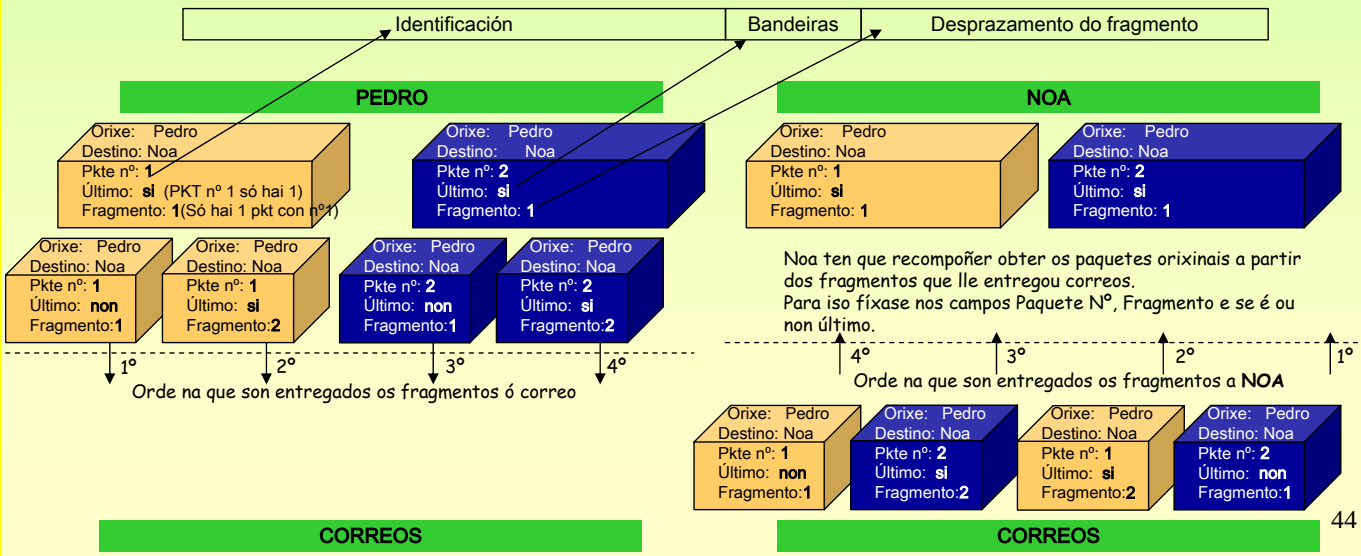
☞ IP (Internet Protocol) – A fragmentación: Exemplo de correos (II)

Obsérvese o seguinte exemplo no que PEDRO desexa enviar dous paquetes a NOA.

Os paquetes a enviar son moi grandes para mandar por correos. Este obrígao a fragmentalos.

Pedro fragmenta cada paquete en 2 anacos, e copia nos anacos a información común do paquete: identificación, destino, orixe, ...
Logo numera cada un dos fragmentos dentro do paquete orixinal para que o receptor ó recibilos poida recompoñer o paquete.

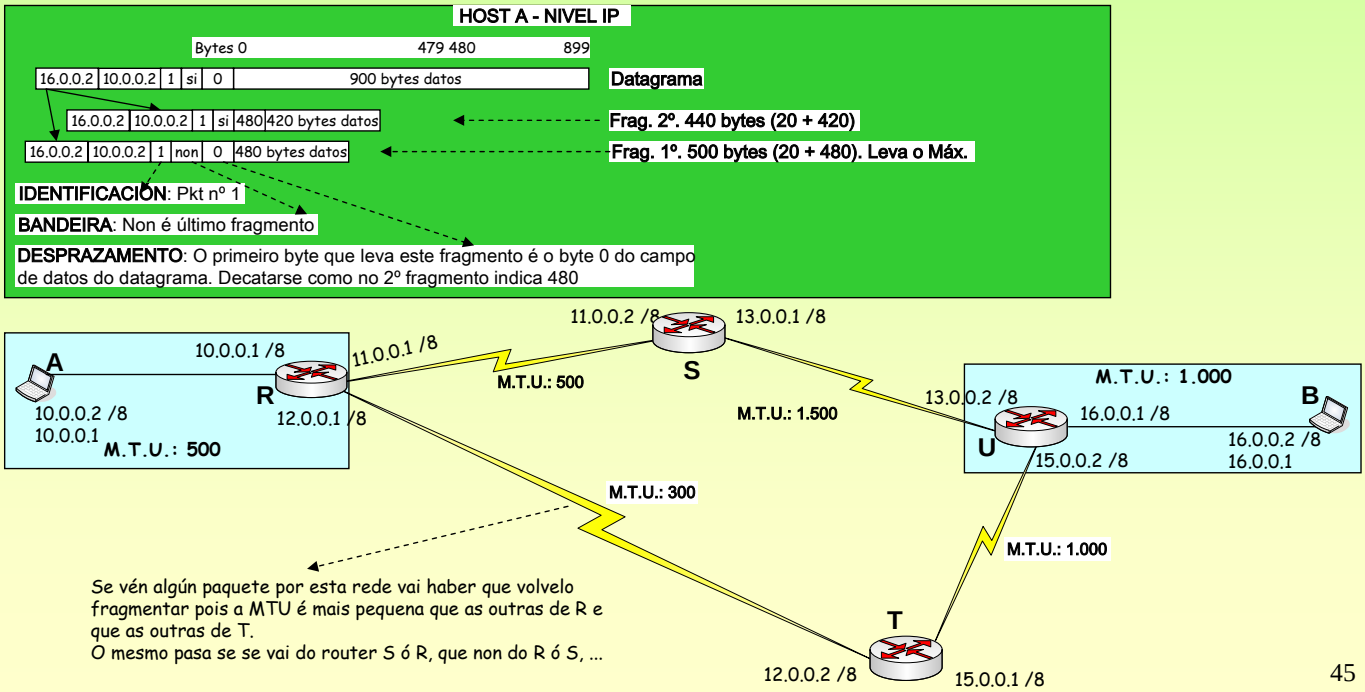
Obsérvese que Pedro envía os fragmentos nunha orde e que correos llos entrega a Noa noutra orde distinta. É Noa quen, coa infomación que ven en cada fragmento ten que recompoñer os paquetes orixinais.



1.- Introducción – TCP / IP

☞ IP (Internet Protocol) – A fragmentación: Exemplo informático (III)

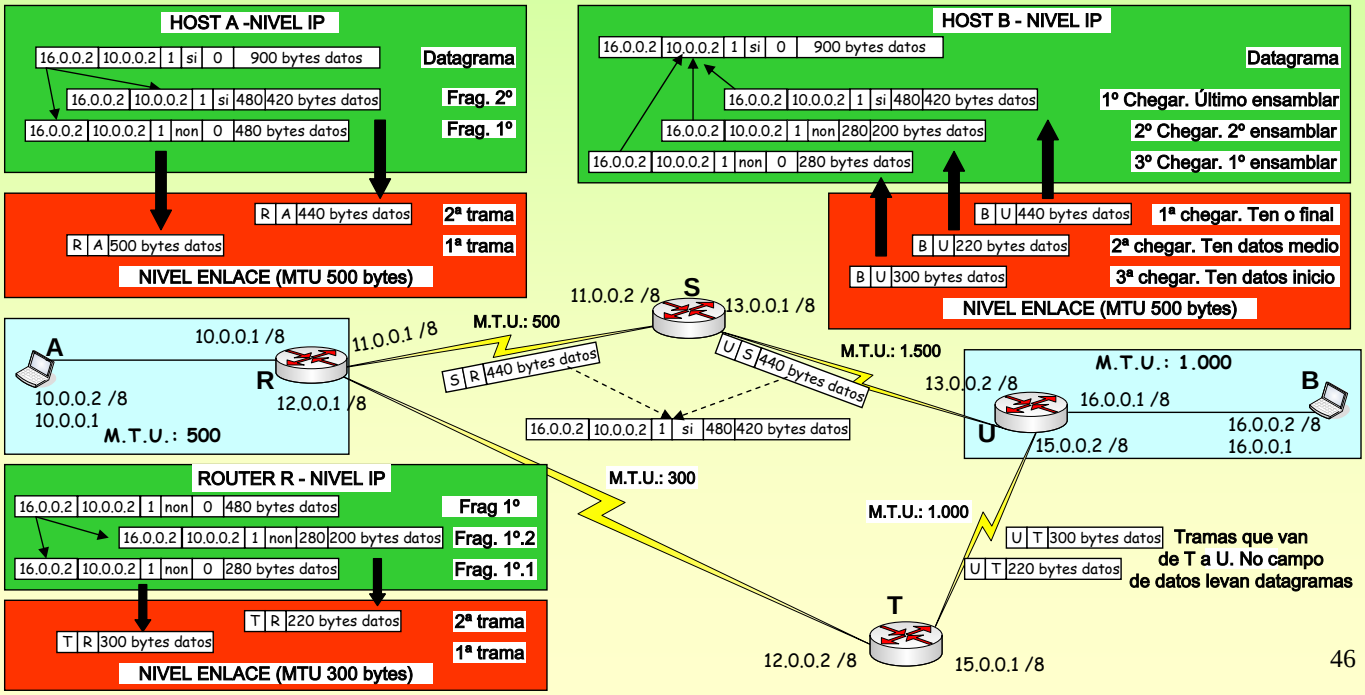
O HOST A desexa enviar un paquete ó HOST B. Existen diferentes MTUs, comprobar no debuxo.
O paquete a enviar mide 920 bytes (900 datos, 20 bytes cabeza sen opcións) e a MTU=500, implica que A vai ter que fragmentar en 2 anacos.
Os routers son dinámicos, isto é, varios paquetes para un mesmo destino, poden ser encamiñados por distintas rutas.
NOTA: O enderezo de máis a esquerda é o destino e o outro é a orixe. Non coincide coa realidade.



1.- Introducción – TCP / IP

☞ IP (Internet Protocol) – A fragmentación: Exemplo informático (IV)

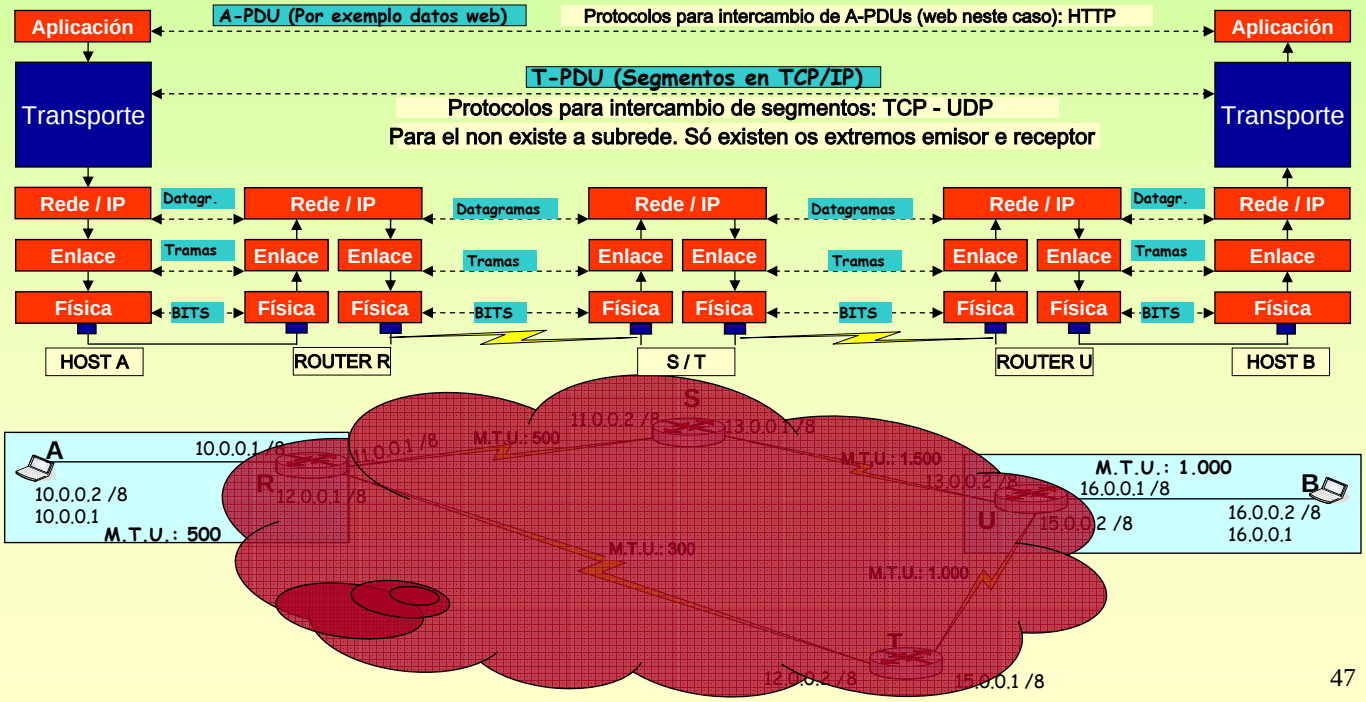
O router R envía o fragmento 2º pola liña superior e o outro pola inferior, que ten MTU=300, co cal ten que volver a fragmentar o fragmento 1º.
No HOST B recíbense os 3 fragmentos desordenados, é responsabilidade do NIVEL IP ordenalos e ensamblalos na orde correcta.
Se non chegou un fragmento, ou a cabeceira chegou con erros (CHEKSUM) descártanse todos os fragmentos coa mesma IDENTIFICACIÓN.
Serán os protocolos da capa de transporte (TCP) os que se encarguen de solucionar eses incidentes.



1.- Introducción – TCP / IP

☞ CAPA DE TRANSPORTE en TCP/IP (TCP - UDP)

É a primeira capa extremo a extremo. Isto é, os protocolos que se establecen nesta capa son entre o extremo EMISOR real e o extremo RECEPTOR real, non entre elementos intermediarios, chamada **Subrede** (routers, switches, hubs, cables, etc.).
O nivel de transporte illa a capa de APLICACIÓN da subrede (nivel IP, enlace, físico).
Para o nivel de transporte é como se só existiran os HOSTS extremos (A e B neste caso), non sabe nada de fragmentación, routers, MTU, hubs...

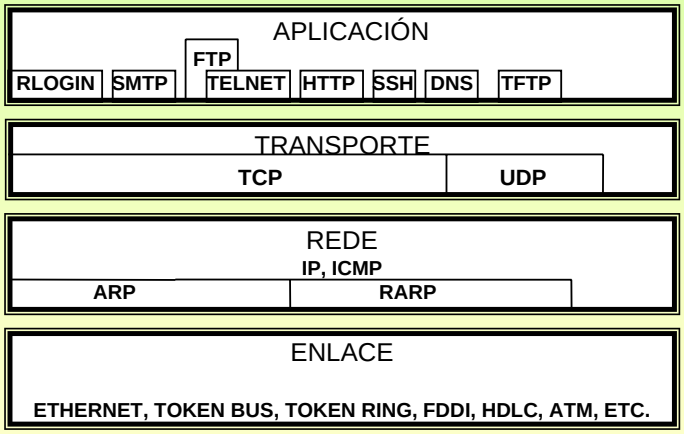


1.- Introducción – TCP / IP

☞ CAPA DE TRANSPORTE en TCP/IP (TCP - UDP)

No seguinte modelo de capas amósase unha síntese dos protocolos que hai en cada nivel.
Obsérvese como hai protocolos de aplicación que só usan TCP, outros UDP e outros os 2.
Pode haber aplicacións que se salten a capa de transporte, por exemplo o comando **Ping.**
A capa de transporte "*transporta*" os datos independentemente das redes subxacentes.

TCP: **Transmision Control Protocol**, é un protocolo orientado á conexión. (Sistema telefónico)
UDP: **User Data Protocol,** é un protocolo non orientado á conexión. (Sistema postal)



1.- Introducción – TCP / IP

☞ TCP (Transmisión Control Protocol) (I)

PORTO: Son os enderezos do nivel de transporte. Son os SAP (Puntos de acceso ó servizo) entre as aplicacións e o TCP/UDP. Cada porto está asociado a unha aplicación. Os portos pódense asignar de dous xeitos:

APLICACIÓN CLIENTE: Cando se abre unha aplicación o SO asínalle un porto dos que teña libres. (Exemplo: navegador web, cliente ftp, etc)

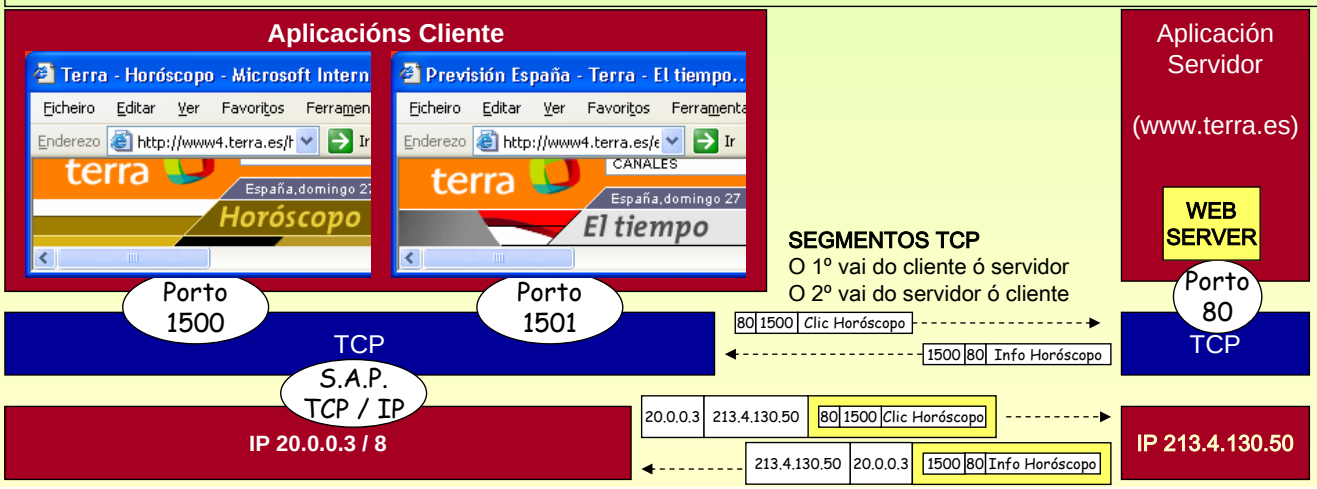
APLICACIÓN SERVIDOR: As aplicacións servidor están sempre escoitando nun porto chamado **BEN COÑECIDO.** Este porto é configurado manualmente. Exemplos **PORTOS BEN COÑECIDOS:**

80 Servidor Web	**21 Servidor FTP**	**23 Telnet**	**22 SSH**
13 Hora / Día	**25 SMTP**	**53 Servidor DNS**	**3389 Terminal Server**

EXEMPLO: Un usuario fai dobre clic sobre o navegador web, nese intre o Sistema Operativo (SO) asínalle un porto a esa aplicación (1500). A aplicación cliente sabe en que porto está escoitando a **Aplicación Servidor** as peticións (neste caso no 80). Se a aplicación servidor está escoitando nun porto distinto ó que lle corresponde, o usuario debe expresar cal é ese porto. (ex. :81)

PUNTO EXTREMO: o par formado por (IP, PORTO), por exemplo: (20.0.0.3, 1500)

CONEXIÓN: circuíto virtual entre dous programas, isto é, un par de puntos extremos. Así podemos abrir varias aplic. nun HOST
Conexión 1: (20.0.0.3, 1500) – (213.4.130.50, 80) **Conexión 2:** (20.0.0.3,1501) – (213.4.130.50, 80)



1.- Introducción – TCP / IP

☞ TCP (Transmisión Control Protocol) (II)

ORIENTADO A CONEXIÓN: Para realizar unha comunicación entre dous puntos extremos, débese:
1º **Establecer** a conexión (O cliente solicita ó servidor que quere comunicarse con el)
2º Unha vez establecida a conexión realízase o **intercambio** de información.
3º Finalizado ó intercambio, **libérase** a conexión.

ASENTIMENTO: **Acuse de recibo**, segmento que envía o receptor ó emisor para informalo de se recibiu correcta (ACK) ou incorrectamente (NACK) o que o emisor enviou.

FULL-DÚPLEX: Permítelle ós dous extremos enviar información nos dous sentidos simultaneamente. Usa para iso o protocolo de ventá deslizante que se verá máis adiante.

PIGGY BACKING: Os segmentos con asentimentos que envía o receptor poden levar ademais datos do receptor cara ó emisor.

FIABLE: Proporciona comunicación extremo a extremo de tal xeito que lle ofrece ás aplicacións unha conexión libre de erros. Para iso úsase o protocolo de ventá deslizante. Lémbrese que o nivel IP non garante que cheguen os datagrama, nin que cheguen ordenados. É o TCP que se encarga de solucionar estes problemas.

CONTROL DE FLUXO: O emisor debe enviar datos adaptándose á velocidade do receptor para procesalos/aceptalos. Unha das funcións do nivel 2 (enlace) do modelo de referencia OSI é o Control de Fluxo, pero nese caso ese control dáse entre os elementos que compoñen a subrede, non entre o emisor e o receptor real.
No nivel de transporte tamén se realiza este control, pero entre o emisor e o receptor real. No caso do TCP úsase o protocolo de ventá deslizante para levar a cabo esta función.

TEMPORIZADORES: O emisor habilita temporizadores para cada segmento que envía se non recibe unha confirmación do receptor antes de que remate o temporizador volve a retransmitir o mesmo segmento.

MSS: **Maximun Segment Size**: (Tamaño do campo de datos do segmento). Cando se establece a conexión entre dous extremos negóciase o tamaño do segmento. O tamaño do segmento deberá ser aquel, que cando se pase este ó nivel de rede, para ir no campo de datos dun datagrama, non provocara a fragmentación do datagrama.

Isto é, debería ir en relación á MTU da rede, co cal, cando se establece a conexión, o nivel TCP trata de averiguar a MTU da rede, e deste xeito calcula o MSS (restar cabeceira segmento e cabeceira datagrama, como mínimo 40 bytes, 20 de cada cabeza). Distínguense dous casos:
- **OS EXTREMOS ESTÁN NUNHA LAN:** a MTU pódese averiguar facilmente pois é a mesma de orixe a destino.
- **OS EXTREMOS ESTÁN EN REDES DISTINTAS**: a MTU é difícil de averiguar, pois no nivel 3 existen routers que poden realizar encamiñamentos dinámicos, o que implica que unhas rotas terán unha MTU e outras terán outra.

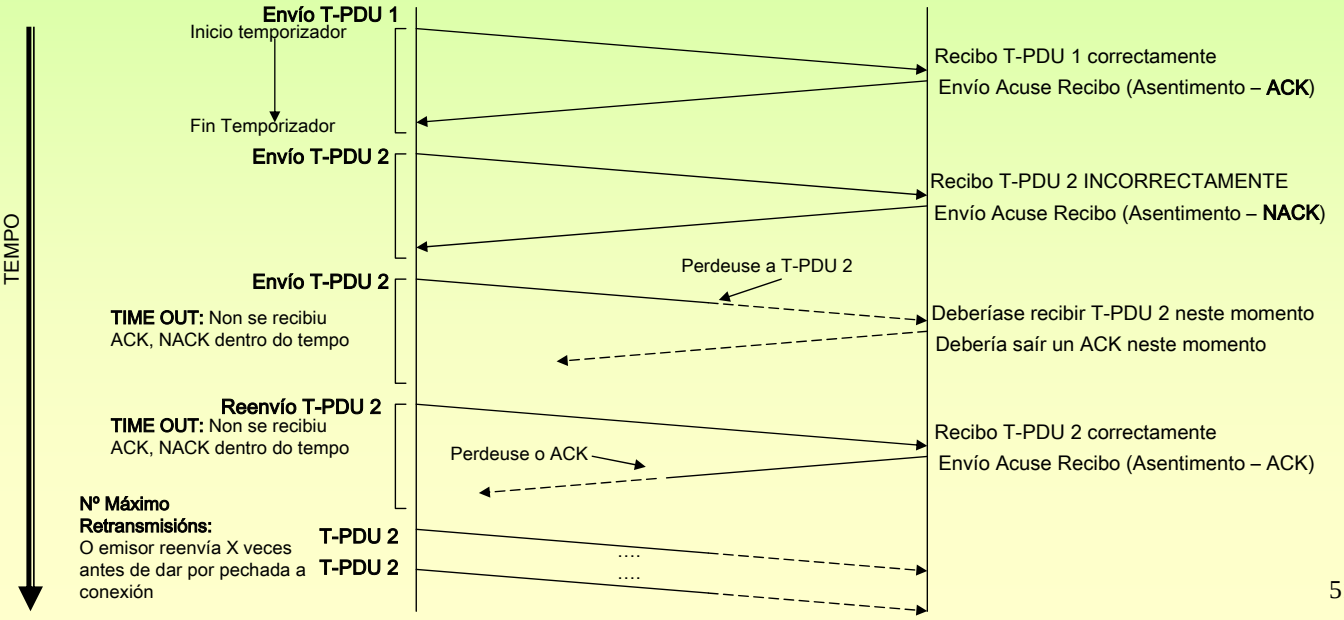
1.- Introducción – TCP / IP

☞ CONTROL DE FLUXO – Técnica: Envío - Espera

Tanto no envío de Tramas (nivel 2) como no envío de segmentos, realízase o control de fluxo. No primeiro caso entre os IPMs que compoñen a subrede e no segundo entre o emisor e o receptor finais.

A técnica de **ENVÍO E ESPERA** consiste en enviar un bloque de información e esperar a que o receptor envíe un acuse de recibo. Mentres non se reciba ese acuse de recibo positivo non se enviará o seguinte bloque de información.

TEMPORIZADOR: o emisor ó enviar un bloque de información abre un temporizador dentro do cal debe recibir un acuse de recibo.
TIME OUT: indica que expirou o temporizador. Cando se trata de conectar a unha páxina e pasado un tempo dá erro.
Nº MAX. RETRANSMIS.: o emisor envía un mesmo bloque de información nun número máximo de X veces. Se se acada péchase a conexión



OSI – TCP/IP

1.- Introducción

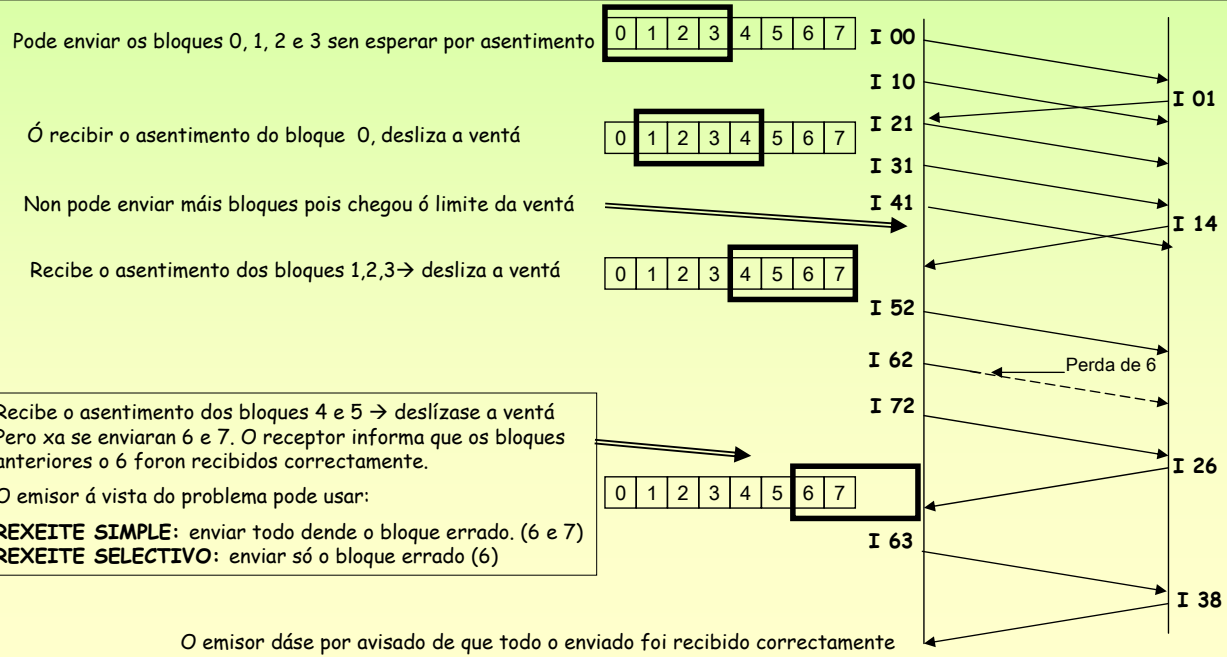
1.- Introducción – TCP / IP

☞ CONTROL DE FLUXO – Técnica: VENTÁ ESVARADÍA (DESLIZANTE)

O protocolo usa a TÉCNICA DE VENTÁ DESLIZANTE CON **REXEITE SELECTIVO**.
O protocolo de ventá deslizante consiste en establecer límite no números de bloques de información que o emisor pode enviar sen recibir acuse de recibo deles.

Cada bloque de información ten o seguinte **formato: I XY**
I= Información **X:** Número de bloque que se envía **Y:** Nº de bloque que se espera, co cal recibiu os **Y-1** bloques **OK**.

EXEMPLO: un emisor ten que enviar 8 bloques de información (0-7) e establécese unha ventá de tamaño 4 bloques.

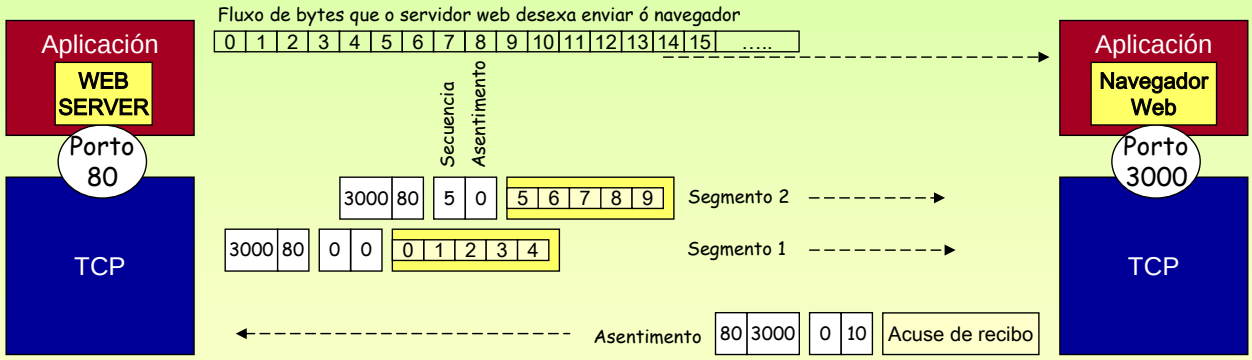


☞ TCP e a Ventá deslizante

O tamaño da ventá deslizante en TCP mídese en bytes, isto é, cantos bytes se van poder enviar sen estar pendente do acuse de recibo.

Cando se envía un segmento o primeiro byte do campo de datos correspóndese cun número de byte do fluxo de bytes que se desexa intercambiar co receptor no nivel de aplicación.

EXEMPLO: Dados: MSS → 5, TAMAÑO DA VENTÁ→10 bytes.
Construír os segmentos necesarios ata o primeiro acuse de recibo.



☞ FIABILIDADE

O software TCP emisor non se desfai dos bytes enviados ata que reciba o asentimento do receptor.

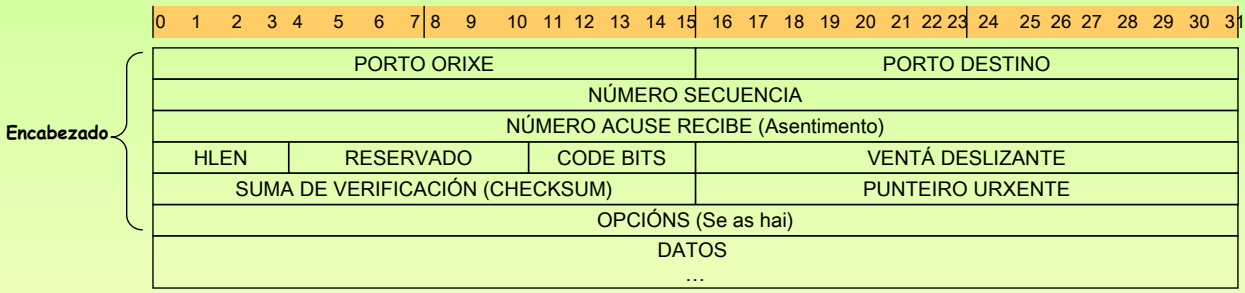
O emisor xestiona temporizadores para cada segmento enviado. No caso de que se perda algún segmento ou se perda un acuse de recibo o temporizador expirará e volverá a retransmitir o segmento errado.

Se o receptor recibe segmentos duplicados vaise decatar, pois cada segmento vai numerado.

Deste xeito o nivel TCP é independente do IP, pois se este perde fragmentos, datagramas enteiros ou estes chegan con erros, ó non subir nada ó nivel TCP, este vaise decatar de que algo anormal está a pasar.

☞ TCP (Transmisión Control Protocol) Formato do segmento (I)

Os segmentos intercámbianse para establecer conexións, transferir datos, enviar acuses de recibo (asentimentos), indicar o tamaño da ventá deslizante e pechar as conexións:
Un acuse de recibo que vaia do HOST A ó B, pode levar datos de A a B.



☞ Algúns campos do segmento.

PORTO: Conteñen os números de porto TCP que identifican as dúas aplicacións dunha conexión.

HLEN: Número enteiro que indica o tamaño da cabeceira medida en palabras de 32 bits (1 liña). Sen opcións: HLEN =5 → 20 bytes.

RESERV.: Reservado para uso futuro

CODE BITS: Pode tomar varios valores, entre eles destacamos:
- **FIN:** indica que é o ultimo segmento dunha restra.
- **URG**: indica que o campo punteiro urxente é válido.
- **RST:** iniciación da conexión.

CHECKSUM.: úsase para o control de erros en TCP, para o seu cálculo inclúese a cabeceira e os datos.

P. URXENTE: Aínda que a información debe ser procesada no receptor na mesma orde na que saíu, ás veces é preciso que o programa dun extremo envíe datos ***fóra de banda*** sen esperar a que o programa do outro lado procese tódolos bytes que aínda están en fluxo. Supóñase que dende un extremo se desexa abortar ou interromper a execución do programa do outro lado. Esa sinal debe saltar todo o fluxo de datos. Exemplo: cando visitamos unha páxina prememos STOP antes de que se remate de cargala.

OPCIÓNS: cando se establece unha conexión entre dous extremos negóciase o MSS (tamaño do segmento). O software TCP usa este campo para realizar esta negociación.

1.- Introducción – TCP / IP

☞ TCP (Transmisión Control Protocol) Formato do segmento (II)

Os segmentos intercámbianse para establecer conexións, transferir datos, enviar acuses de recibo (asentimentos), indicar o tamaño da ventá deslizante e pechar as conexións:
Un acuse de recibo que vaia do HOST A ó B, pode levar datos de A a B.

	0 1 2 3	4 5 6 7	8 9 10 11 12 13 14 15	16 17 18 19 20 21 22 23	24 25 26 27 28 29 30 31
Encabezado	PORTO ORIXE			PORTO DESTINO	
	NÚMERO SECUENCIA				
	NÚMERO ACUSE RECIBE (Asentimento)				
	HLEN	RESERVADO	CODE BITS	VENTÁ DESLIZANTE	
	SUMA DE VERIFICACIÓN (CHECKSUM)			PUNTEIRO URXENTE	
	OPCIÓNS (Se as hai)				
	DATOS …				

☞ Os restantes campos do segmento.

VENTÁ: En cada acuse de recibo que o receptor lle envía ó emisor, infórmao de cantos bytes máis está disposto a recibir, co cal o tamaño da ventá é dinámico e vaise adaptando á dispoñibilidade de memoria do receptor.

Cando o receptor envía este campo cun valor 0, estalle indicando ó emisor que se deteña ata nova orde.

Nº SECUENCIA: O emisor informa ó receptor que byte ocupa o primeiro byte do campo de datos dentro do fluxo de datos que está enviando unha aplicación a outra.

ORDE: ó ir tódolos segmentos numerados, pódese entregar a información á aplicación do HOST receptor na mesma orde en que foron enviados pola aplicación do HOST emisor, ainda que estes foran entregados polo nivel IP do receptor en desorde.

Hai que ter en conta que eses segmentos que chegaron ó TCP receptor puideron ir no nivel IP por rotas distintas, xa que no nivel IP os datagramas son encamiñados dinamicamente.

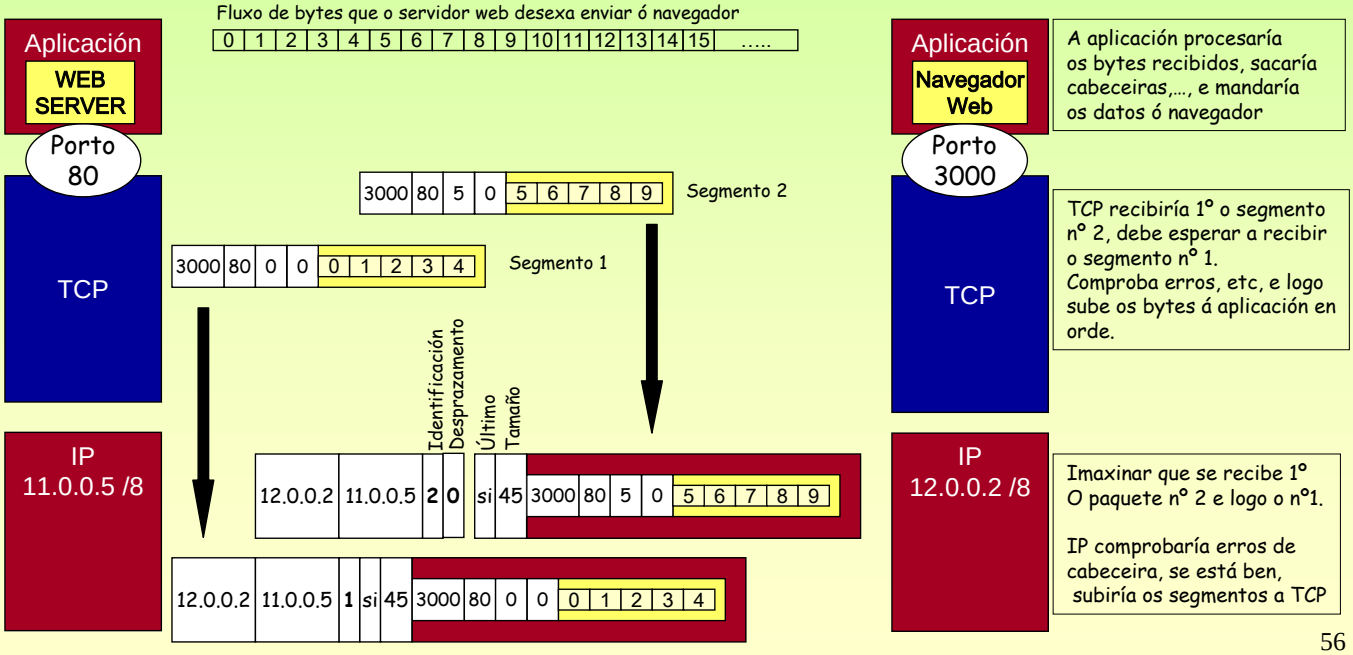
Nº ASENTIMENTO: O receptor informa ó emisor cal é o seguinte byte polo que está a esperar, confirmándolle así o emisor, que todo o enviado ata ese byte - 1 foi recibido correctamente.

1.- Introducción – TCP / IP

☞ A relación entre as tres capas: Aplicación, TCP, IP

EXEMPLO: Dados: MSS → 5, TAMAÑO DA VENTÁ→10 bytes.

Construír os segmentos e datagramas necesarios ata o primeiro acuse de recibo. Fixarse no campo identificación do datagrama.
NOTA: Os enderezos están: 1º o destino e logo a orixe.

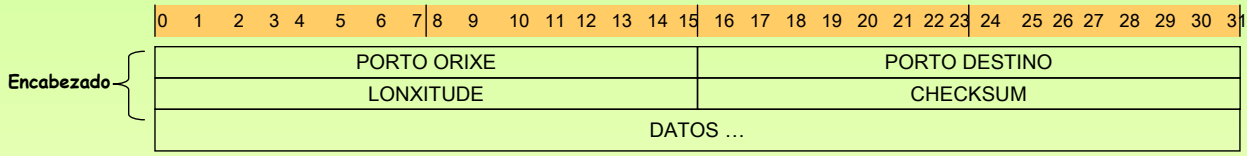


1.- Introducción – TCP / IP

☞ **UDP (Unidade de Datos do Protocolo).**

É o protocolo da capa de transporte NON ORIENTADO Á CONEXIÓN.
A diferencia do TCP non é fiable, non garante que os datos se entreguen en orde nin que se recupere de erros.

En consecuencia, é rápido pero inseguro.



1.- Introducción – TCP / IP

☞ **COMANDOS**
Windows: netstat

C:\WINDOWS\System32\cmd.exe

```
L:\>netstat -?

Muestra estadísticas del protocolo y conexiones TCP/IP actuales.

NETSTAT [-a] [-e] [-n] [-o] [-s] [-p proto] [-r] [intervalo]

  -a            Muestra todas las conexiones y puertos de escucha.
                (Normalmente, el extremo servidor de las conexiones no se
                muestra).
  -e            Muestra estadísticas Ethernet.  Se puede combinar con la
                opción -s.
  -n            Muestra números de puertos y direcciones en formato
                numérico.
  -o            Muestra la Id. de proceso asociado con cada conexión
  -p proto      Muestra conexiones de
                que puede ser TCP, UDP
                -s para mostrar estadí
                TCP, UDP, TCPv6 o UDPv
  -r            Muestra el contenido
  -s            Muestra estadísticas
```

C:\WINDOWS\System32\cmd.exe

```
L:\>netstat

Conexiones activas

  Proto  Dirección local        Dirección remota       Estado
  TCP    BARRIOSESAMO:4446      10.0.0.35:microsoft-ds  ESTABLISHED
  TCP    BARRIOSESAMO:4691      10.0.0.6:microsoft-ds   ESTABLISHED
  TCP    BARRIOSESAMO:4944      www.terra.es:http       TIME_WAIT
  TCP    BARRIOSESAMO:4945      www.terra.es:http       TIME_WAIT
  TCP    BARRIOSESAMO:4955      10.0.0.35:microsoft-ds  TIME_WAIT
```

C:\WINDOWS\System32\cmd.exe

```
L:\>netstat -n

Conexiones activas

  Proto  Dirección local        Dirección remota       Estado
  TCP    10.0.0.5:4446          10.0.0.35:445          ESTABLISHED
  TCP    10.0.0.5:4691          10.0.0.6:445           ESTABLISHED
  TCP    10.0.0.5:4958          213.4.130.50:80        ESTABLISHED
  TCP    10.0.0.5:4959          213.4.130.210:80       TIME_WAIT
  TCP    10.0.0.5:4960          213.4.130.210:80       ESTABLISHED
  TCP    10.0.0.5:4961          195.22.198.32:80       ESTABLISHED
  TCP    10.0.0.5:4962          209.202.249.250:80     ESTABLISHED
  TCP    10.0.0.5:4963          213.4.130.210:80       ESTABLISHED
  TCP    10.0.0.5:4964          213.4.130.210:80       TIME_WAIT
  TCP    10.0.0.5:4965          213.4.130.50:80        ESTABLISHED
  TCP    10.0.0.5:4966          213.4.130.50:80        ESTABLISHED
  TCP    10.0.0.5:4967          213.4.130.50:80        ESTABLISHED
  TCP    10.0.0.5:4968          213.86.246.154:80      ESTABLISHED
  TCP    10.0.0.5:4971          213.86.246.154:80      ESTABLISHED
  TCP    10.0.0.5:4972          64.237.51.161:80       ESTABLISHED
  TCP    10.0.0.5:4973          213.4.130.210:80       ESTABLISHED
  TCP    10.0.0.5:4974          213.4.130.210:80       ESTABLISHED
  TCP    10.0.0.5:4975          200.16.144.230:80      ESTABLISHED
  TCP    10.0.0.5:4976          213.4.130.210:80       ESTABLISHED
  TCP    10.0.0.5:4977          213.4.130.210:80       ESTABLISHED
```

1.- Introducción – TCP / IP

```
Sesión  Editar  Vista  Marcadores  Preferencias  Ayuda
[root@linuxp root]# netstat --help
usage: netstat [-veenNcCF] [<Af>] -r         netstat {-V|--version|-h|--help}
       netstat [-vnNcaeol] [<Socket> ...]
       netstat { [-veenNac] -i | [-cnNe] -M | -s }

        -r, --route              display routing table
        -i, --interfaces         display interface table
        -g, --groups             display multicast group memberships
        -s, --statistics         display networking statistics (like SNMP)
        -M, --masquerade         display masqueraded connections

        -v, --verbose            be verbose
        -n, --numeric            don't resolve names
        --numeric-hosts          don't resolve host names
        --numeric-ports          don't resolve port names
        --numeric-users          don't resolve user names
        -N, --symbolic           resolve hardware names
        -e, --extend             display other/more information
        -p, --programs           display PID/Program name for sockets
        -c, --continuous         continuous listing

        -l, --listening          display listening server sockets
        -a, --all, --listening   display all sockets (default: connected)
        -o, --timers             display timers
        -F, --fib                display Forwarding Information Base (default)
        -C, --cache              display routing cache instead of FIB

  <Socket>={-t|--tcp} {-u|--udp} {-w|--raw} {-x|--unix} --ax25 --ipx --netrom
  <AF>=Use '-A <af>' or '--<af>'; default: inet
  List of possible address families (which support routing):
    inet (DARPA Internet) inet6 (IPv6) ax25 (AMPR AX.25)
    netrom (AMPR NET/ROM) ipx (Novell IPX) ddp (Appletalk DDP)
    x25 (CCITT X.25)
```

COMANDOS

Linux: netstat

Como se pode observar este comando serve para máis cousas que para ver as conexións TCP.

1.- Introducción – TCP / IP

```
Sesión  Editar  Vista  Marcadores  Preferencias  Ayuda
[root@linuxp root]# netstat Socket -t
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 linuxp:postgres        linuxp:34324            ESTABLISHED
tcp        0      0 linuxp:postgres        linuxp:34323            ESTABLISHED
tcp      351      0 linuxp:37063           10.0.0.35:netbios-ssn   ESTABLISHED
tcp        0      0 linuxp:37085           10.0.0.35:microsoft-ds  ESTABLISHED
tcp        0      0 linuxp:33431           10.0.0.35:microsoft-ds  ESTABLISHED
tcp        0      0 linuxp:32769           10.0.0.35:microsoft-ds  ESTABLISHED
tcp        0      0 linuxp:37304           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:37305           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:37297           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:37298           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:37299           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:37300           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:37301           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:37302           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:37303           carpanta.rede.usc.:http TIME_WAIT
tcp        0      0 linuxp:34323           linuxp:postgres         ESTABLISHED
tcp        0      0 linuxp:34324           linuxp:postgres         ESTABLISHED
tcp        0      0 linuxp:37204           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37202           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37200           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37201           10.0.0.5:x11            ESTABLISHED
tcp        0   1204 linuxp:37198           10.0.0.5:x11            ESTABLISHED
tcp       64      0 linuxp:37199           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37196           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37197           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37195           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37192           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37193           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37189           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37187           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:37167           10.0.0.5:x11            ESTABLISHED
tcp        0      0 linuxp:33251           10.0.0.38:netbios-ssn   ESTABLISHED
tcp        0      0 linuxp:37285           prsc12_40.xunta.es:http TIME_WAIT
```

COMANDOS

Linux: netstat

Os estados das conexións tanto en Linux como en Windows, poden ser, entre outros:

- CLOSE_WAIT
- CLOSED
- ESTABLISHED
- FIN_WAIT_1
- FIN_WAIT_2
- LAST_ACK
- LISTEN
- SYN_RECEIVED
- SYN_SEND
- TIME_WAIT

Para coñecer o seu significado recoméndase consultar o:

RFC 793

Onde se especifica o TCP.

www.ietf.org

☞ SISTEMA DE NOMES DE DOMINIOS (DNS).

Pero, ¡¡¡¡Os humanos non traballan directamente con IPs!!!!
DNS deseñouse a comezos dos 80 en 1984 escolleuse como estándar **para asociar IPs a Nomes**.

Antes de que Internet cambiase a DNS existía un único arquivo (Hosts.txt) que se enviaba a través de FTP a quen quixese converter IPs a nomes. Cada cambio implicaba a modificación do arquivo e volvelo a distribuir.

DNS mantén unha base de datos nun servidor ó cal preguntan aqueles que desexen achar a IP asociada a un nome de dominio.

Espacio de nomes.

Describe a estructura en forma de árbore de todos os dominios dende o raís (" . ", punto) ata o nivel inferior da estructura. A estructura é xerárquica e cada nivel sepárase do superior por un punto ". "

Dominios de primeiro nivel.

Son os dominios que se atopan xusto debaixo do dominio raíz " . ". Estes divídense en dous tipos:

Dominios Organizativos: Creados inicialmente para organizar o Internet en EE.UU.

.COM:	inicialmente era para empresas, hoxe está aberto a calquera cousa.
.NET:	inicialmente era para empresas e organismos relacionados coa Rede, hoxe ...
.ORG:	inicialmente era para organismos de EE.UU. sen ánimo de lucro, hoxe ...
.MIL:	inicialmente era para organismos militares de EE.UU. e hoxe sígueo sendo.
.EDU:	inicialmente era para universidades de EE.UU. e hoxe sígueo sendo.
.GOV:	é para organismos relacionados co goberno de EE.UU. e hoxe sígueo sendo.
.INT:	é para organismos internacionais, p.e. www.eu.int (Portal da Unión Europea)

Dominios xeográficos: xurdiron cando o Internet se expandiu alén dos EE.UU.

.ES	España. Non fixo control sobre os dominios secundarios.
.UK	Reino Unido. Fixo control sobre os dominios secundarios. P.e. co.uk, gov.uk, org.uk
.BR	Brasil. Fixo o mesmo que os inglese
.DE	Alemaña
.PT	Portugal

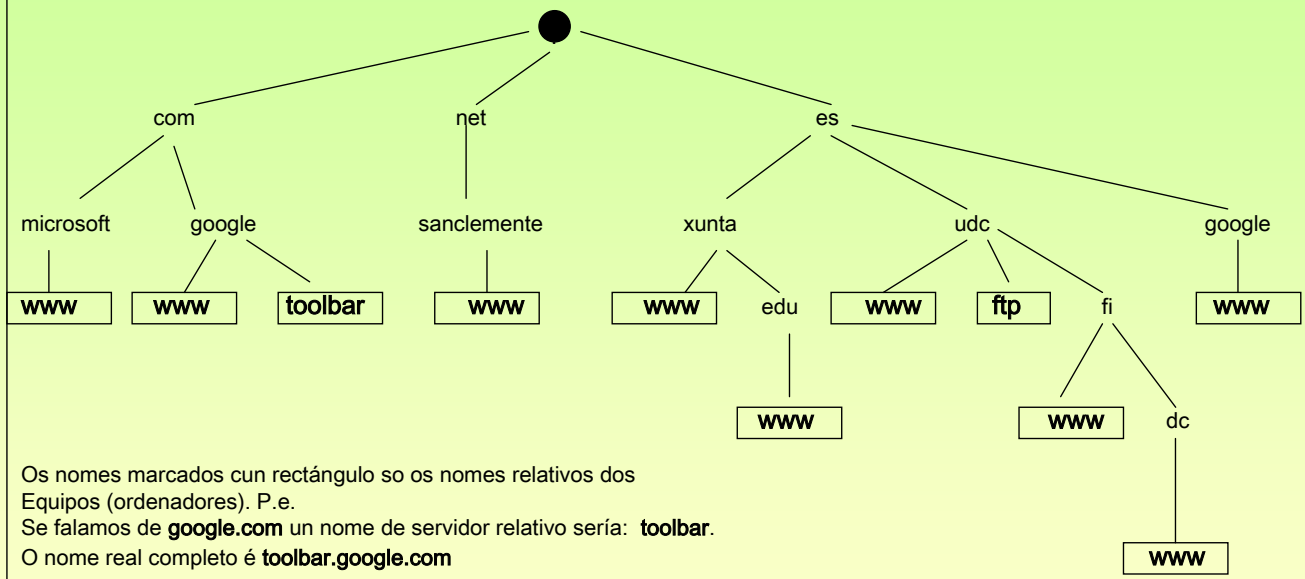
Dominios de recente creación: .tv, .mail, .info, .museum. En www.internic.net ou en www.icann.org están todos.

OSI – TCP/IP

1.- Introducción – TCP / IP

1.- Introducción

☞ SISTEMA DE NOMES DE DOMINIOS (DNS). Estructura.



Os nomes marcados cun rectángulo so os nomes relativos dos Equipos (ordenadores). P.e.
Se falamos de **google.com** un nome de servidor relativo sería: **toolbar**.
O nome real completo é **toolbar.google.com**

☞ Consideracións p.e. do dominio da xunta.

xunta.es	→ é un dominio, e ó mesmo tempo **xunta** é un subdominio de **.es**
edu.xunta.es	→ é un dominio, e ó mesmo tempo **edu** é un subdominio de **xunta.es**
www.xunta.es	→ é o equipo **www** dentro do dominio **xunta.es**
www.edu.xunta.es	→ é o equipo **www** dentro do dominio **edu.xunta.es**
Toolbar.google.com	→ é o equipo **toolbar** dentro do dominio **google.com**

1.- Introducción – TCP / IP

☞ Configuración DNS (Domain Name System)

Pero, ¡¡¡¡Os humanos non traballan directamente con IPs!!!!
Ese problema resólvese con nomes de dominio do estilo www.iessanclemente.nets, www.terra.es, www.edu.xunta.es

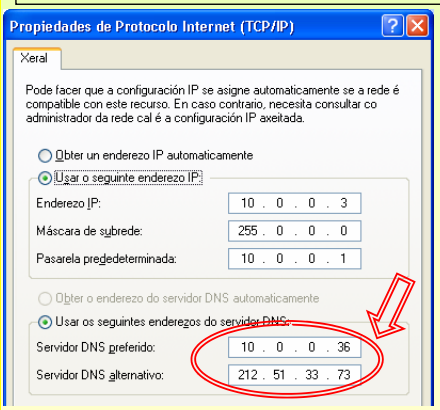
Analoxía con sistema telefónico: Unha persoa pode saber uns cantos números de teléfono, pero se descoñece algún pode chamar ó 11811 para preguntar polo número dun abonado, pero se este número non funciona ou está ocupado podes chamar a outro 11824.

En TCP/IP existe o Servidor de Nomes de Dominio (DNS) que ten unha IP á cal os clientes DNS poden preguntarlle cal é a IP asignada a un nome de dominio.

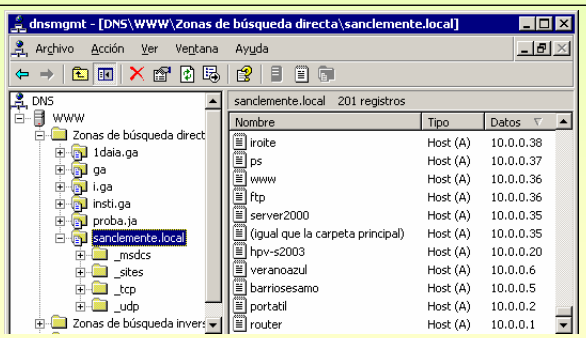
Os clientes configuraranse indicando a IP do servidor de DNS que pode resolver as súas consultas.

Servidor DNS primario, preferido, etc: É ó 1º servidor ó que se lle vai consultar se fallase consultaríase a:
Servidor DNS secundario, alternativo: Este servidor é consultado no caso de que falle o primeiro.

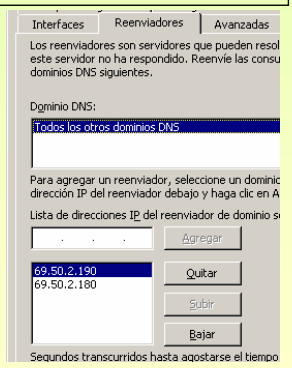
Os servidores DNS non saben tódalas IPs e nomes de dominio existentes. Estes organízanse en forma de árbore, de tal xeito que se un servidor de DNS non é capaz de resolver un nome de dominio este **REENVÍA** a pregunta a outro servidor de DNS ou usa **RECURSIVIDADE** ata atopar o nome de dominio ou obter unha resposta negativa.



Configuración cliente DNS



10.0.0.36. Configuración server DNS. Zonas e equipos

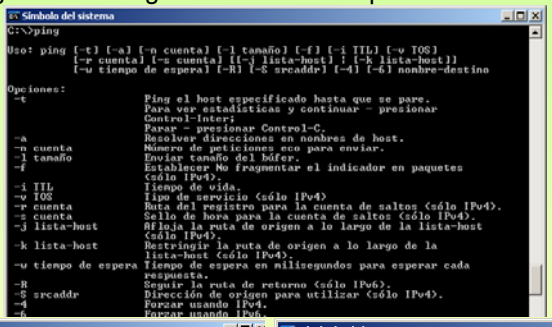


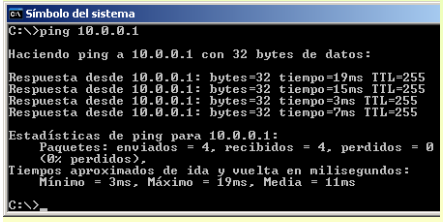
Configuración server DNS. Reenviador

1.- Introducción - TCP/IP

☞ PING (ICMP)

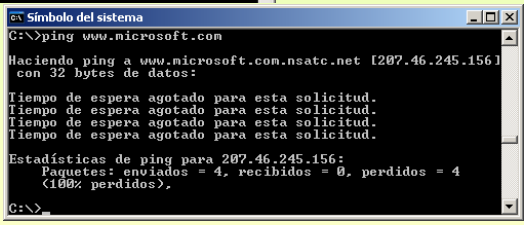
Comando que axuda a comprobar a conectividade no nivel IP, esto é, comprobar que dous HOSTs se poidan conectar.
Para elo precisa coñecer a IP do destinatario.
Se se especifica un nome de dominio o ping encárgase de averiguar a IP usando o proceso de consultas DNS.
Obsérvense os seguintes exemplos:





Ping a unha IP que coñecemos.
O respondernos indícanos canto tempo tarda en chegar un PKT. Deste xeito sabemos que 10.0.0.1 **is alive**

O programa debe averiguar a IP de www.terra.es [está entre corchetes] e logo realiza o "ping".
Terra está acendido, respondendo e polos tempos máis lonxe que 10.0.0.1.

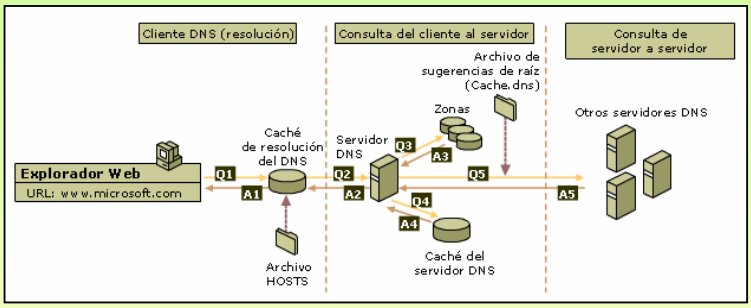


O programa averigua a IP e logo realiza o "ping". O host non responde:
A.- Pode ser que estea apagado, ou non que non se pode chegar a el.
B.- Pode estar acendido pero deshabiltada a resposta a pings.

1.- Introducción - TCP/IP

☞ DNS (Domain Name System)

No seguinte exemplo móstrase como funcionan as consultas DNS. (Tomado da axuda de Windows)
O proceso de averiguar a IP asociada a un nome de dominio coñéceselle co nome: **Resolución DNS**
Un ordenador do IES (cliente) fai un **ping** a **www.microsoft.com**. Para elo débese averiguar a súa IP.
Neste exemplo só nos interesa que se resolva a consulta DNS non que resposte o servidor.



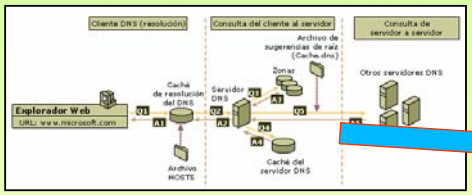
O cliente DNS dispón de:
Caché DNS: onde se almacena resultados de resolucións previas, incluso as de resultado negativo.
Arquivo HOSTS: está (...\system32\drives\etc\). Mantén asociacións estáticas de Nomes con IPs.

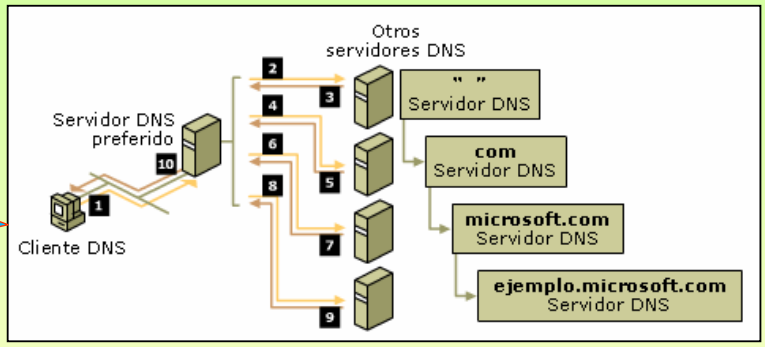
Q1: Cliente DNS consulta a súa cache DNS (xa inclúe os datos do arquivo HOSTS automaticamente) pregunta pola IP de www.microsoft.com.
A1: Se existe entrada devolve a IP senón sigue o proceso:
Q2: Pregunta ó servidor de DNS configurado como preferido:
Q3: O servidor de DNS consulta ás súas zonas (Os dominios que xestiona el) (Arquivos *.dns de windows\dns\ do server)
A3: Se Xestiona ese dominio (microsoft.com) e ten ese host (www) devolve a IP ó cliente, senón segue o proceso.
Q4: O servidor de DNS ten almacenada na Caché do Servidor de DNS as resolucións que resolveu previamente.
A4: Se o servidor ten esa entrada na caché devolve a IP ó cliente, senón segue co proceso.
Q5: Se o server DNS non puido resolver, preguntará a outros servidores DNS.
A5: Eses servidores devolverán ó SERVER DNS anterior a IP ou o fallo DNS. O server DNS anterior almacenará na caché o resultado para as futuras peticións que reciba.
A2: Devolve ó cliente o resultado da busca (IP ou Fallo). O cliente almacenará na súa caché o resultado para futuras consultas.

1.- Introducción - TCP/IP

☞ DNS (Domain Name System) – PROCESO DE RECURSIVIDADE

Cando o **Servidor de DNS preferido** non atopa información nas súas bases de datos locais nin na caché DNS é cando pregunta a outros servidores. Por defecto o servidor de DNS ven configurado cunha lista de servidores raíz (root) os que preguntar para estes casos. Tamén ven, por defecto, activado para usar o **proceso de recursividade**:





1.- O cliente desexa comunicarse con **ftp.ejemplo.microsoft.com**. Tras consultar a súa caché pregunta o servidor DNS preferido.
O servidor DNS preferido consulta as súas zonas e a súa caché e non pode resolver.

PROCESO DE RECURSIVIDADE.
2.- O servidor pregunta a un dos seus servidores raíz(root), ¿Quen é o servidor DNS que xestiona os dominios .COM?
3.- O servidor root dálle unha **referencia (IP)** ó servidor DNS que xestiona os .COM. O servidor preferido almacena na caché esa **referencia** (IP) para futuras consultas a un .COM.
4.- O servidor preferido pregunta ó servidor de DNS que xestiona as .COM ¿Sabes algo de **MICROSOFT.COM**?.
5.- O xestor DNS do dominio .COM devólvelle unha **referencia** (IP) ó servidor que xestiona o dominio **MICROSOFT.COM.**
6,7.- 8,- Semellante ós pasos anteriores.

9.- O servidor DNS **ejemplo.microsoft.com** trata de resolver a IP do host **FTP.** Ben resolva **positivamente** ou **negativamente** informará ó servidor de DNS que fixo a petición do resultado e este almacenarao na súa cache DNS de servidor.
10.- Fin da recursividade. O servidor informa ó cliente do resultado e este almacena na cache e actúa en consecuencia.

1.- Introducción - TCP/IP

☞ DNS (Domain Name System) – REENVÍO – REENVÍO CONDICIONAL

Cando se configura un servidor de DNS pode interesar que este pregunte a outro/s servidor/es de DNS **concreto/s** antes de usar o Proceso de Recursividade.

Observar o seguinte caso:
A Xunta é a Provedora de Servicios de Internet (**ISP**) dos IES. Como tal, ofrécelles dous servidores de DNS os que os clientes dos centros poden facer as súa peticións de Resolución DNS. Estes servidores xestionan o dominio **xunta.es**

Agora ben, o centro ten a súa propia intranet local (p.e. **sanclemente.local**) co seu servidor de DNS local (10.0.0.36). Os clientes do centro preguntan a ese servidor de DNS.

Se o servidor de DNS local está configurado para reenviar as consultas que non poida resolver a eses dous reenviadores.

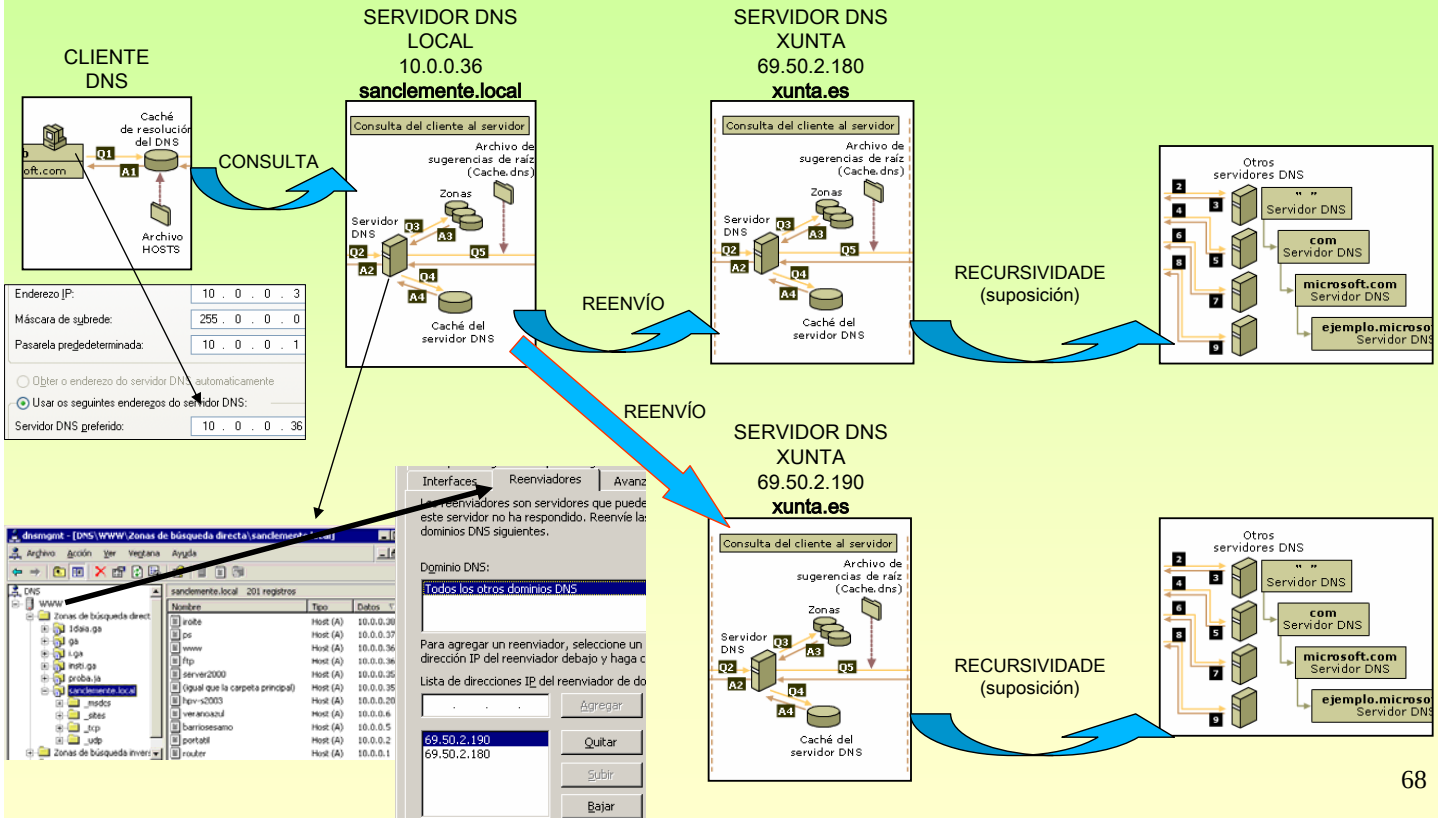
Deste xeito, un cliente desexa conectarse a **ola.xunta.es e a www.microsot.com**. O proceso é como segue.

1.- O cliente consulta a súa cache local, se non atopa nada reenvía a pregunta o server DNS preferido local.
2.- O servidor DNS preferido local, trata de resolver usando as súas bases de datos e a súa cache se non atopa nada preguntará a un servidor DNS da XUNTA.
3.- Se o servidor DNS da XUNTA non resposta no tempo establecido preguntarase ó outro reenviador , neste caso tamén da XUNTA.
4.- Cada un deles consultará a súa base de datos (para **ola.xunta.es**), a caché (para **www.microsoft.com**)
   4.a.- No caso de **ola.xunta.es** o server da XUNTA devolve ó local que non existe ola.xunta.es.
   4.b.- No caso de microsoft se non atopa nada na caché usará reenvío ou recursividade en función de como estea configurado. Unha vez que teña unha resposta almacenaraa na caché e respostaralle ó servidor local
5.- O servidor DNS local almacenará na caché as respostas e enviaraas ó cliente.
6.- Finalmente o cliente almacenará na cache as respostas e actuará en consecuencia.

REENVÍO CONDICIONAL.
Permite que uns dominios sexan consultados por un reenviador e outros por outro.

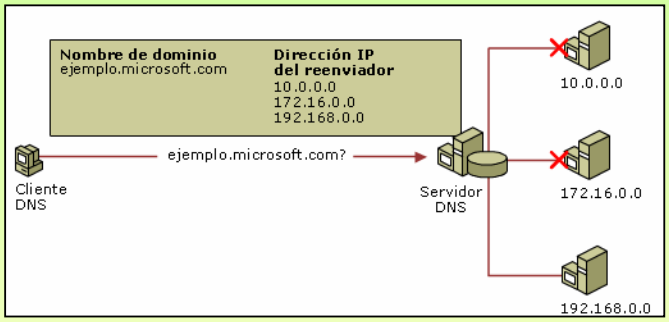
1.- Introducción - TCP/IP

☞ DNS (Domain Name System) – REENVÍO – REENVÍO CONDICIONAL (II) EXEMPLOS

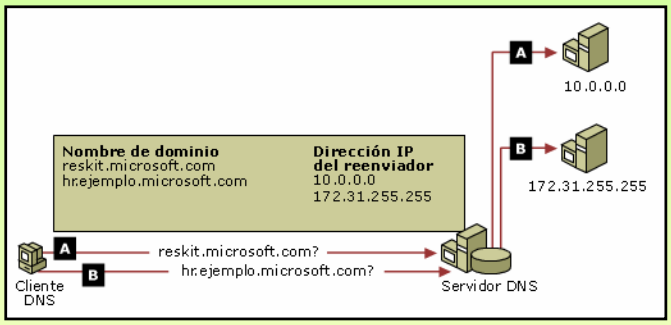


1.- Introducción - TCP/IP

☞ DNS (Domain Name System) – REENVÍO – REENVÍO CONDICIONAL (III) EXEMPLOS



A este servidor DNS non lle responderon no tempo establecido os dous primeiros reenviadores

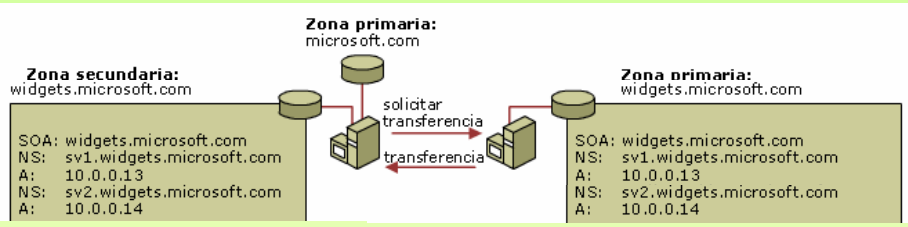


Servidor de reenvío condicional. Un dominio (A) é consultado a un reenviador e o outro dominio (B) a outro reeenviador.

1.- Introducción - TCP/IP

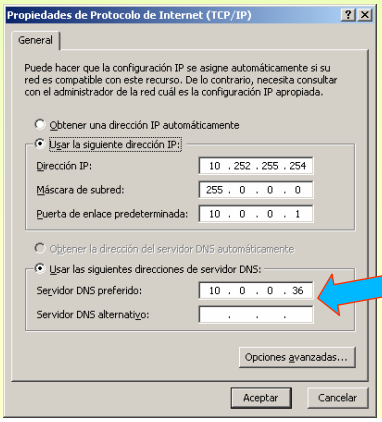
☞ ZONAS SECUNDARIAS

Son copias de respaldo da información que ten unha zona principal. Como no caso anterior da XUNTA que ofertaba dous servidores DNS (primario e secundario)



☞ ACTUALIZACIÓN DUNHA ZONA SECUNDARIA

O servidor secundario envía unha petición principal para pedir permiso par actualizarase, logo pídelle actualización completa (transferir todo de principal a secundario AXFR) ou incremental (IXFR).

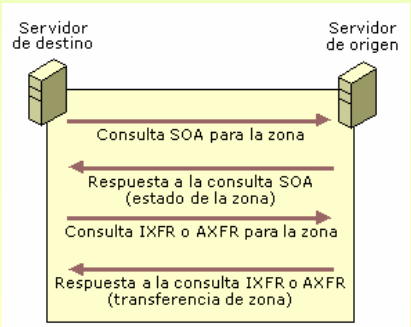


Configuración cliente DNS

Pódense especificar varios DNS ós que preguntar.

Se o 1º non responde Pregúntaselle ó segundo, e así sucesivamente.

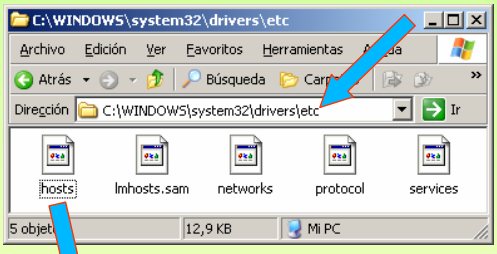
Ata que un deles dea unha resposta ben positiva ben negativa

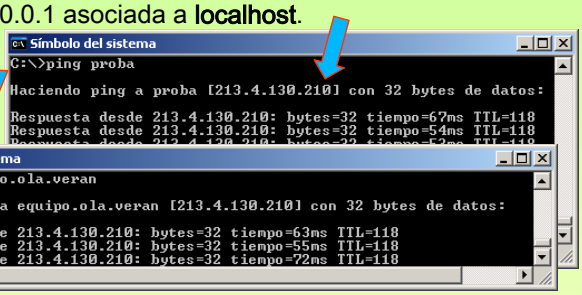


1.- Introducción - TCP/IP

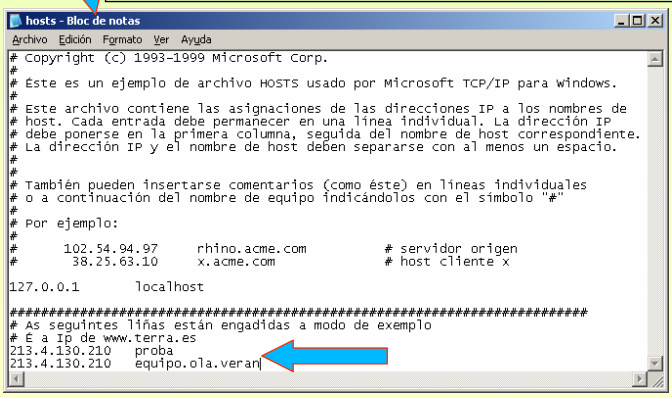
☞ ARQUIVO HOSTS

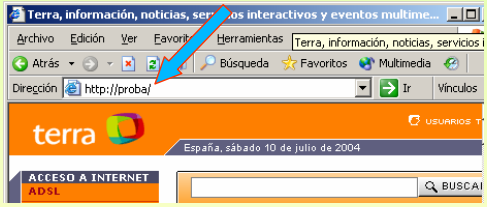
Todo cliente DNS ten un arquivo HOSTS, onde se almacena estaticamente asociacións de de nomes de equipos (con ou sen o dominio) e as súas IPs. Sempre ten a entrada de loopback 127.0.0.1 asociada a **localhost**.



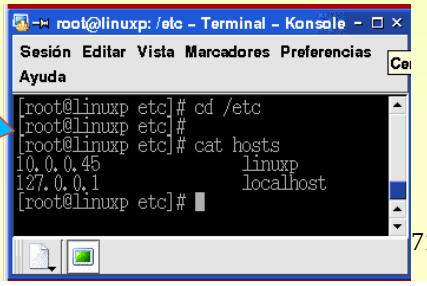


Engadíronselle dúas entradas o final a modo de exemplo. O resultado é o da dereita. Só modificable por administradores





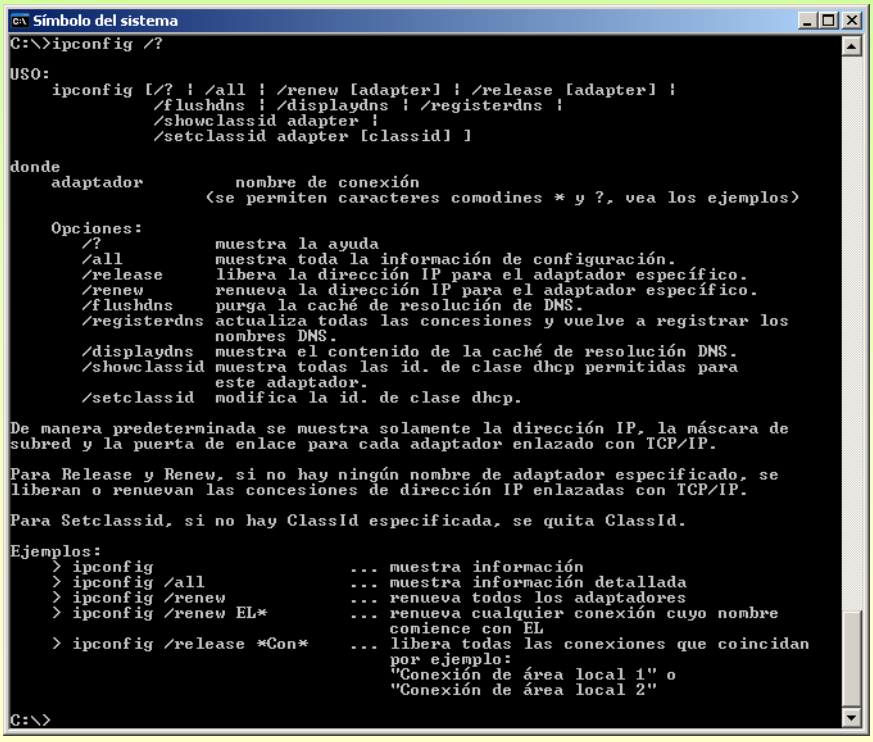
En Linux /etc/hosts



1.- Introducción - TCP/IP

☞ COMANDOS: IPCONFIG (WINDOWS) (I)

Mostra os valores da configuración TCP/IP. E actualiza a configuración de DHCP (Dynamic Host Configuration Protocol, que se verá máis adiante) e de DNS.



1.- Introducción - TCP/IP

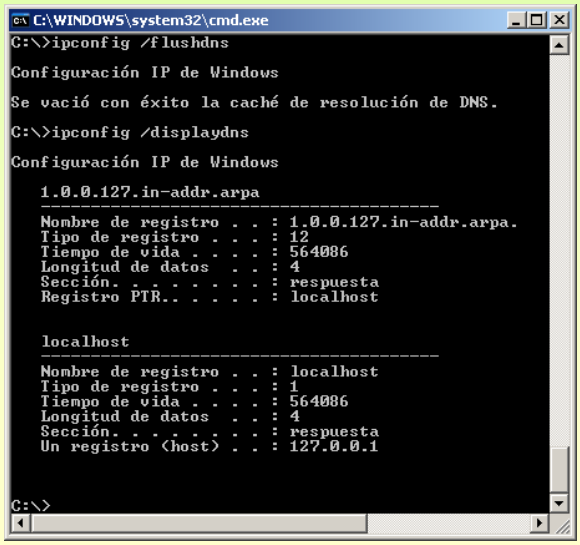
☞ COMANDOS: IPCONFIG (WINDOWS) (II) – BORRADO DA CACHÉ DNS DO CLIENTE

Mostra os valores da configuración TCP/IP. E actualiza a configuración de DHCP (Dyamic Host Configuration Protocol, que se verá máis adiante) e de DNS.

1º Se os datos están no arquivo HOSTS borrando as entradas as dúas entradas anteriores xa non estarán na caché local para a próxima ocasión que se pregunte por elas.

2º As entradas na caché DNS que proceden do Servidor DNS preferido bórranse co comando **ipconfig /flushdns**

Tras o borrado e actualización do arquivo HOSTS, a caché DNS cliente está como segue.



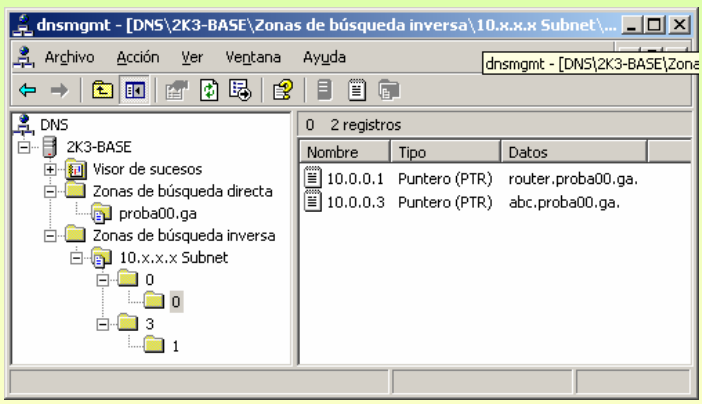
OSI – TCP/IP

1.- Introducción - TCP/IP

☞ DNS (Domain Name System) – Zoas de busca INVERSA

Ás veces é interesante dada unha IP averiguar cal é nome de dominio que ten asignado.
Esto é útil cando se ten un conflicto IP (máis dunha máquina coa mesma IP) e se desexa averiguar quen é o causante.
Pódese desconectar un dos implicados, faise un ping –a <IP en conflicto> e saberase o nome douto dos afectados.

Para elo é preciso dar de alta unha Zoa de Busca Inversa no servidor DNS que teña asociadas IPs a Nomes.



1.- Introducción - TCP/IP

☞ DNS (Domain Name System) – NSLOOKUP

Mostra información sobre a infraestructura dun servidor DNS

Iniciamos a aplicación
Nome e IP do servidor DNS que vai realizar as resolucións .

IP? De quen xestiona a zona **xunta.es**
Observar que amosa o nome e a IP do servidor que resolve
Neste caso **www.sanclemente.local** 10.0.0.36

IP? do equipo **www.xunta.es**.

Observar o alias
Fixarse que servidor DNS e web están na mesma IP

IP? De que xestiona a zona **edu.xunta.es**

IP? Do equipo **www** dentro do dominio **edu.xunta.es**

IP? Do equipo **smtp** (Correo) dentro do dominio **edu.xunta.es**

Saímos

OSI – TCP/IP

1.- Introducción - TCP/IP

☞ DNS (Domain Name System) – Un mesmo nome de dominio con varias IPs

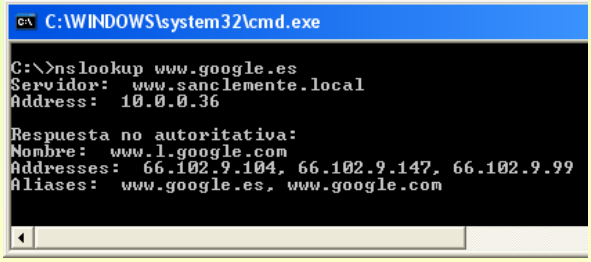
Imaxínese un servidor web (p.e. www.google.es) distribuído en 3 hosts distintos para balancear a carga. Ó mesmo tempo desexase que todos eles respondan ó mesmo nome de dominio (www.google.es).

A solución é simple: so hai que dar de alta na zona google.es 3 hosts de alta co mesmo nome (www) e con distintas IPs.

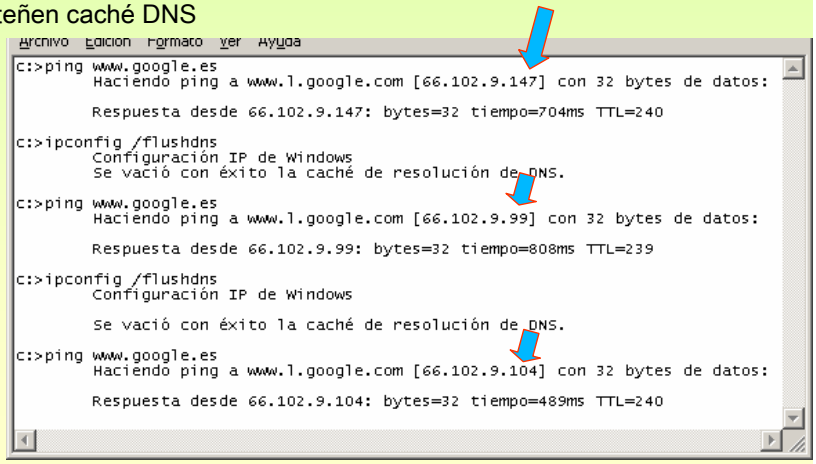
Deste xeito ó servidor DNS ó ser consultado dará unha IP distinta cada vez.

OLLO os SO windows almacenan na caché DNS a IP dunha resolución previa, para comprobar o cambio de IP cada vez que se solicita unha conexión a www.google.es é preciso baleira-la caché.

En linux esto último non é preciso, pois os hosts non teñen caché DNS



Obsérvense as 3 IPs asignadas a www.google.es e os distintos alias



1.- Introducción - TCP/IP

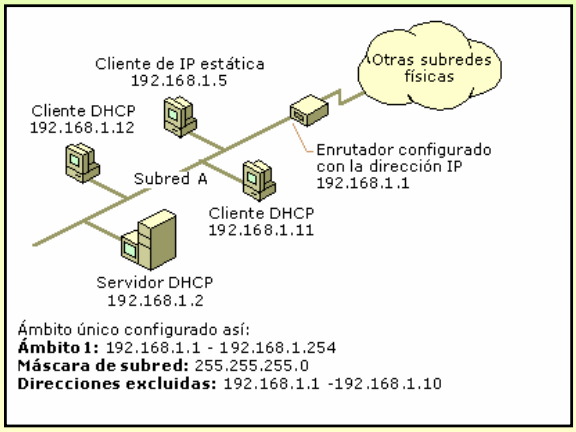
☞ DHCP (Dynamic Host Configuration Protocol).

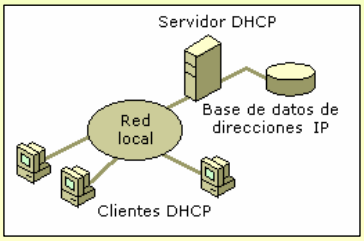
Hai veces nas que é interesante que os usuarios con ordenadores portátiles poidan chegar a un IES (p.e.), conectarse fisicamente á rede (por cable ou por wi-fi) e que o usuario nin o administrador teñan que estar a configurar as propiedades do protocolo de Internet.

Pois ben, débese configurar un servidor de DHCP que ofreza un rango de IPs coa súa máscara, porta de enlace e DNS.

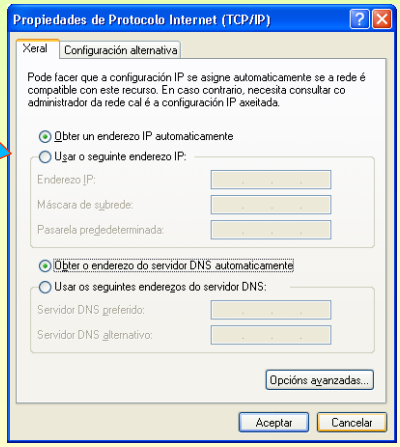
Ó acenderse un equipo que teña configurado **Obter automaticamente unha IP** este preguntará á toda a rede se hai alguén que lle poida dar unha IP, o servidor DCHP escoitará a petición e será el quen lla ofreza. O mesmo co DNS.

O servidor DHCP leva un control das IPs que leva asignadas





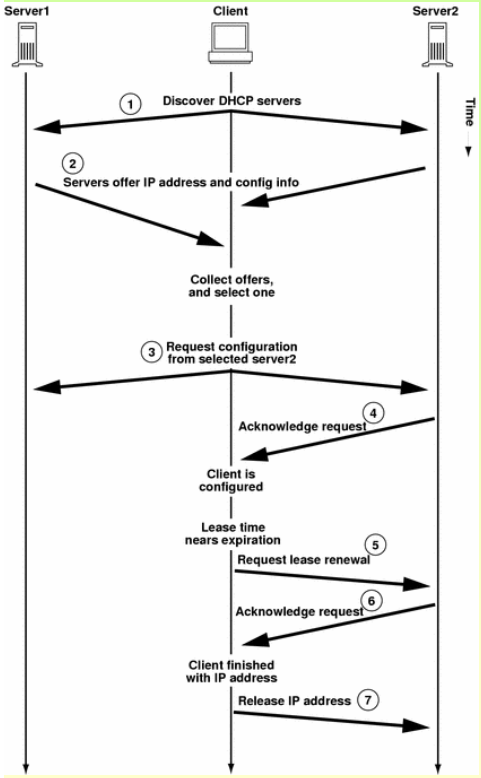
Configuración Cliente

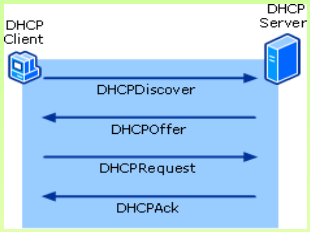


OSI – TCP/IP

1.- Introducción - TCP/IP

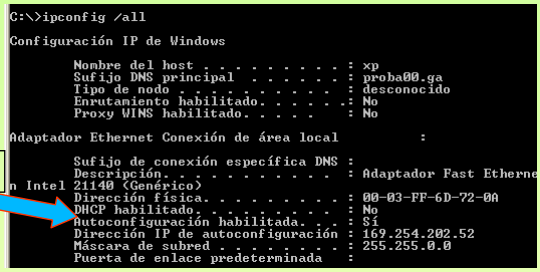
☞ FUNCIONAMENTO do DHCP (Dynamic Host Configuration Protocol).





APIPA

APIPA



1.- O cliente solicita unha IP difundindo unha mensaxe DHCP DISCOVER á subrede local

2.- Os servidores ofrecen unha dirección IP (DHCP OFFER) e demais configuración (DNS, dominio, etc) se esta está configurada para ser entregada. Se ningún servidor DHCP responde ó cliente, este envía DHCP DISCOVER cada 0,4,8,16 e 32 seg e logo un intervalo aleatorio ate un minuto. Se pasado1 minuto e non recibe resposta:

A.- Se o cliente usa APIPA (Automatic Private IP addressing), o cliente autoconfigúrase cunha IP (no caso de Microsoft será un IP da rede 169.254.0.0/24)

B.- O interface do cliente non se inicializa (IP 0.0.0.0 /0)

En ambos casos comeza cun novo ciclo DHCP DISCOVER cada 5 mn.

3.- O cliente ó recibir DHCP OFFER indica a un dos oferentes que acepta a IP recibida (DHCP REQUEST)

4.- O servidor envía unha confirmación DHCP ACK ó cliente indicándolle os termos do arrendamento. A partir de agora o cliente xa pode usas a IP asignada.

5.- O cliente solicita renovación da IP cando pase a metade do tempo da concesión.

6.- O servidor concédelle a renovación.

7.- O Cliente libera a IP

1.- Introducción - TCP/IP

☞ PKI (Public Key Infrastructure, Infraestructura de Chave Pública) (I)

A PKI encargase de procesos relacionados co cifrado de información (Criptografía ven do grego **Krytos** = esconder e **graphos**= grafía, escritura).

☞ PROBLEMAS A RESOLVER (PIANO = CIANO)

Privacidade / Confidencialidade: Un emisor envía unha información cifrada que só o receptor pode entender, ó descifrala. Se a mensaxe é interceptada por un terceiro, este non a entenderá

Integridade: fai referencia a que a información que envía un emisor a un receptor non chegue alterada por un terceiro. Non importa que o terceiro entenda a mensaxe, interesa que non a modifique e se esto ocorre, que o receptor se decate.

Autenticidade: os participantes dunha conversa deben ser quen din ser e non estar suplantados (algo semellante a presentación do DNI por parte dun alumno nun exame, para non suplantar a outra persoa).

Non Repudio: o emisor dunha información nunca pode negar que el foi o remitente.

Lectura recomendada

Para comprender os conceptos asociados a PKI como:

- Chave simétrica,
- Chave pública,
- Resumo,
- Firma dixital,
- Certificados, etc.

Recoméndase a lectura do documento extraído do CERES (Autoridade Pública de Certificación Española). www.cert.fnmt.es

OSI – TCP/IP

1.- Introducción - TCP/IP

☞ PKI (Public Key Infrastructure, Infraestructura de Chave Pública) (II)

Unha vez lido o documento, extráese:

Chave simétrica: serve para intercambiar información cifrada entre interlocutores. Estes deben coñecer a chave de cifrado:
- Ventaxa: é rápido.
- Inconvinte: ¿como intercambiar a chave entre o emisor e o receptor?

Chave pública: cada interlocutor xenera dúas chaves (unha inversa da outra); Privada (quédase o usuario con ela), Pública (distribúea entre os demais usuarios).
- Ventaxa: ainda que alguén intercepte unha mensaxe cifrado coa pública e teña a cha ve pública non poderá descifrar nin a mensaxe nin a chave privada.
- Inconvinte: os algoritmos de cifrados son lentos e xeran mensaxes cifrados moitísimo máis grandes que os orixinais.

- **Resumo**: a través dun algoritmo obtense unha síntese dos datos orixinais. O emisor enviará a mensaxe orixinal e o resumo. O receptor realiza a mesma función sobre a mensaxe orixinal e compara o resume obtido co recibido. Deste xeito comproba se a mensaxe foi modificada polo camiño.
  - Ventaxa: Permite ó receptor asegurarse que a mensaxe non sufriu mudas dende a orixe.

- **Certificado**: é unha garantía emitida por un "notario" asegurando que a chave pública dun usuario é certamente dese usuario.
  - Ventaxa: Un usuario A non poderá pasarse polo usuario B dicindolle a C que lle envía a chave pública B.

Verase máis adiante un estudio máis profundo dos certificados.

1.- Introducción - TCP/IP

☞ PKI (Public Key Infrastructure, Infraestructura de Chave Pública) (IV)

☞Resolución dos problemas:

Problemas	Solucións
Privacidade / Confidencialidade: (Que un terceiro non entenda)	1.- Chave simétrica: (Problema intercambio da chave) 2.- Chave asimétrica: (Problema de lentitude) 3.- Combinación de ambas: Cifrar mensaxe con simétrica e intercambiar a simétrica cifrándoa coa pública do destinatario da mensaxe.
Integridade: (que un terceiro non modifique)	1.- Os tres anteriores. 2.- Obter un resumo da mensaxe e enviar este xunto coa mensaxe. (ten o problema de que un terceiro, sabendo a función de hash, podería modificar a mensaxe e o resumo) 3.- Obter un resumo da mensaxe e cifralo coa chave publica do receptor (non habería firma dixital nin privacidade). 4.- Obter un resumo da mensaxe e cifralo coa chave privada do emisor (a mensaxe estaría firmada pero non habería confidencialidade
Autenticidade: (Emisor sexa quen di ser)	1.- Cifrar a mensaxe coa privada do emisor, só el ten a privada: (Lento). 2.- Realizar un resumo da mensaxe e cifralo co privada do emisor: (rápido)
Non repudio: (Emisor non negue a paternalidade da mensaxe)	1.- Cifrar a mensaxe coa privada do emisor: (Lento). 2.- Realizar un resumo da mensaxe e cifralo co privada do emisor: (rápido) En calquera dos dous casos só o emisor ten a súa chave privada, co cal non pode negar a paternalidade da mensaxe

OSI – TCP/IP

1.- Introducción - TCP/IP

☞ PKI (Public Key Infrastructure, Infraestructura de Chave Pública) (V)

☞Certificados:

Un certificado divídese en tres partes cada unha delas cos seus campos:

- Identidade do solicitante do certificado (persoa, empresa, organismo, etc)
- A chave pública que hai que certificar
- A firma da entidade certificadora.

Os datos dos dous primeiros son proporcionados polo usuario, mentres que o último e xerado pola entidade certificador ((CE) tamén chamada Autoridade Certificadora (CA).

Unha CA non é máis que unha especie de notario que certifica que a chave pública contida no certificado pertence a o usario que identificado, tamén, no certificado. Para elo a CA o que fai e facer un resumo das dúas primeiras partes e logo cifralo coa súa chave privada.

Cada entidade certificadora tamén ten dúas chaves (privada e simétrica). A privada quédase ela con ela e a pública é distribuída mediante un certificado da CA.

Pénsese nun usuario A que recibeu un certificado dun usuario B, para que o usuario A poida comprobar que o certificado é correcto ten que obter o resumo das dúas primeiras partes e logo contrastalo co que ven no certificado (3ª parte). Pero para iso precisa descifralo, e é aquí, cando o usuario A precisa a chave pública (certificado) da CA para poder descifrar esa firma da CA.

1.- Introducción - TCP/IP

☞ PKI (Public Key Infrastructure, Infraestructura de Chave Pública) (VI)

☞ Certificados: X.509 v3

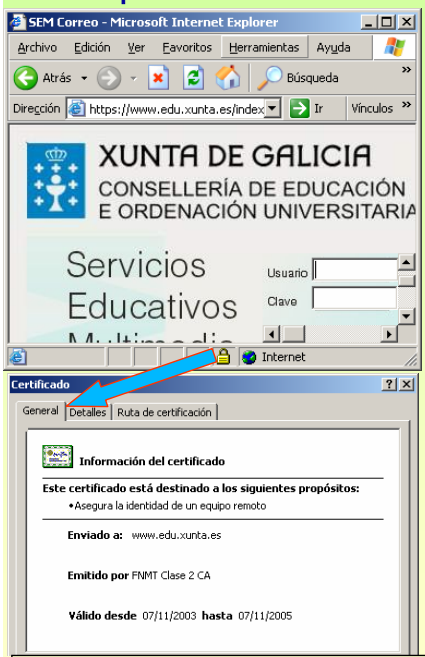
O estándar X.509 define o formato e contido dos campos dun certificado. Actualmente vai na versión 3, esta permite definir campos a parte dos xa establecidos.

Campos	Descrición
Versión	Versión do estándar X.509 (1, 2 ou 3)
Nº Serie	A AC a cada certificado que emite pónlle un nº. Este tamén serve para comprobar se o certificado está na lista dos revocados (CRL).
Emisor Certificado	Quen emite o certificado, esto é quen o firma. Por exemplo, FNMT, Verisign, etc.
Algoritmo de firma	Cal foi o algoritmo usado para obter o resumo (firma)
Período de validez	Dende (data) ate (data)
Usuario	Indentificación do dono do certificado, a quen se lle está certificando a súa chave pública
Chave pública	A chave pública que vai compartir cos demais usuarios. Lonxitude desta, con que algoritmo se xerou, etc.
Datos opcionais	Datos extras que desexe incluir o usuario.
Firma	Resumo do resto dos campos obtido co algoritmo de firma e cifrado coa chave privada da CA

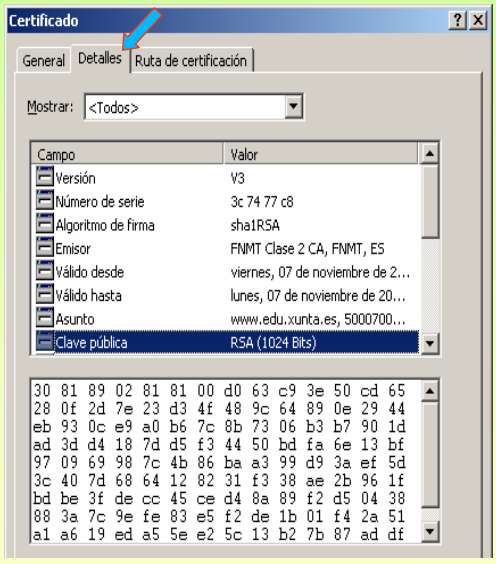
1.- Introducción - TCP/IP

☞ PKI (Public Key Infrastructure, Infraestructura de Chave Pública) (VII)

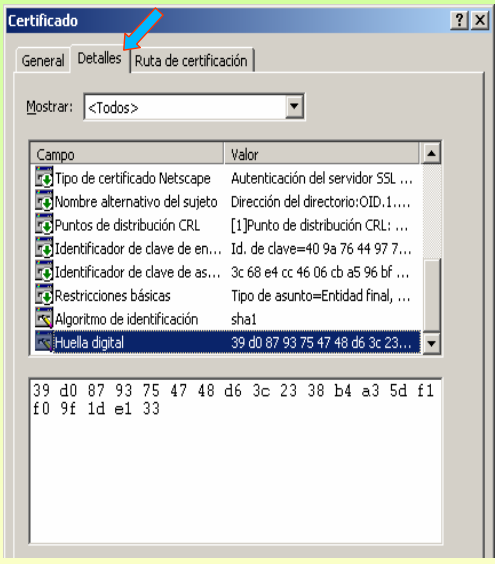
☞ Exemplo de certificado: correo web de www.edu.xunta.es



Nunha páxina https facendo dobre clic sobre o candado inferior vese o certificado SSL. Certificado emitido pola FNMT



Obsérvese:
- Os campos antes indicados.
- Quen o emite.
- Para quen o emite, etc.
- A chave pública do dono do certificado

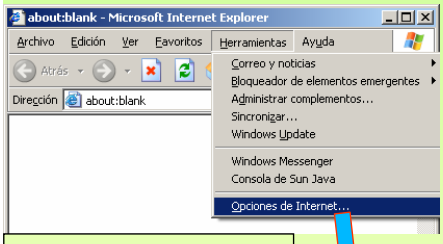


Obsérvese:
A firma dixital, obtida co algoritmo sha1RSA

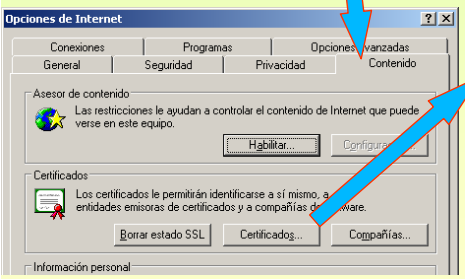
1.- Introducción - TCP/IP

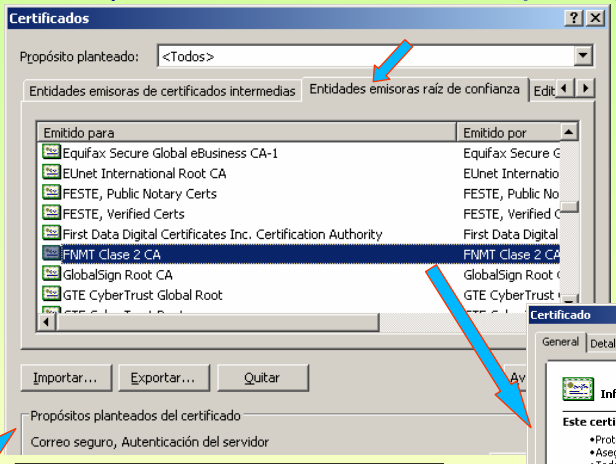
☞ PKI (Public Key Infrastructure, Infraestructura de Chave Pública) (VII)

☞Certificados raíz instalados nos clientes (certificados de emitidos da CA para a propia CA)



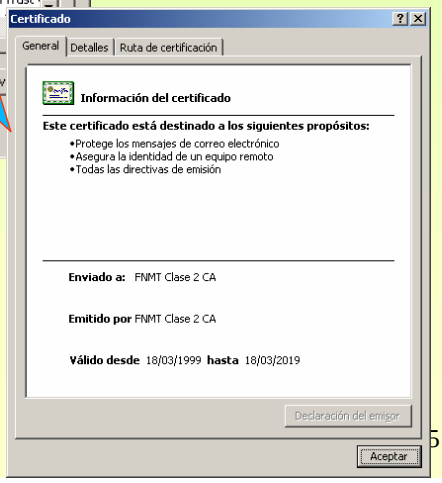
Neste caso trabalarase co Internet Explorer de MS





O certificado anterior foi emitido poal Fábrica Nacional de Moeda e Timbre (FNMT). Para comprobar se está correcto precísase a chave pública da CA, ou sexa o seu certificado. Como se ve está instalado nas CA raíz.

Obsérverse como o emite a FNMT para a FNMT



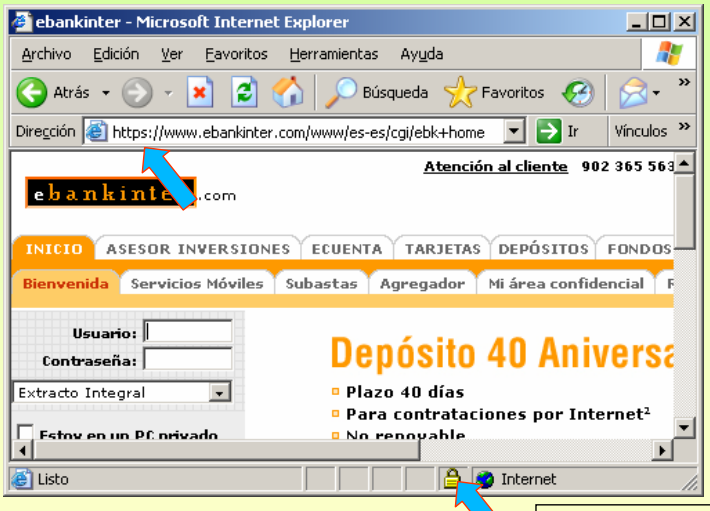
OSI – TCP/IP

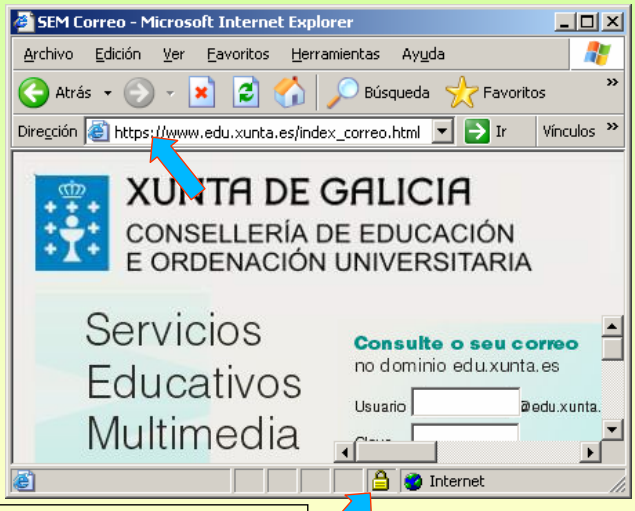
1.- Introducción - TCP/IP

☞ SSL (Secure Socket Layer, Capa de Sockets Seguros) (II)

-O porto ben coñecido dunha conexión que use httpS (SSL) é o 443.

-Nun cliente web (navegador) sábese cando está en modo seguro cando na súa parte inferior aparece un candado e na url Https:





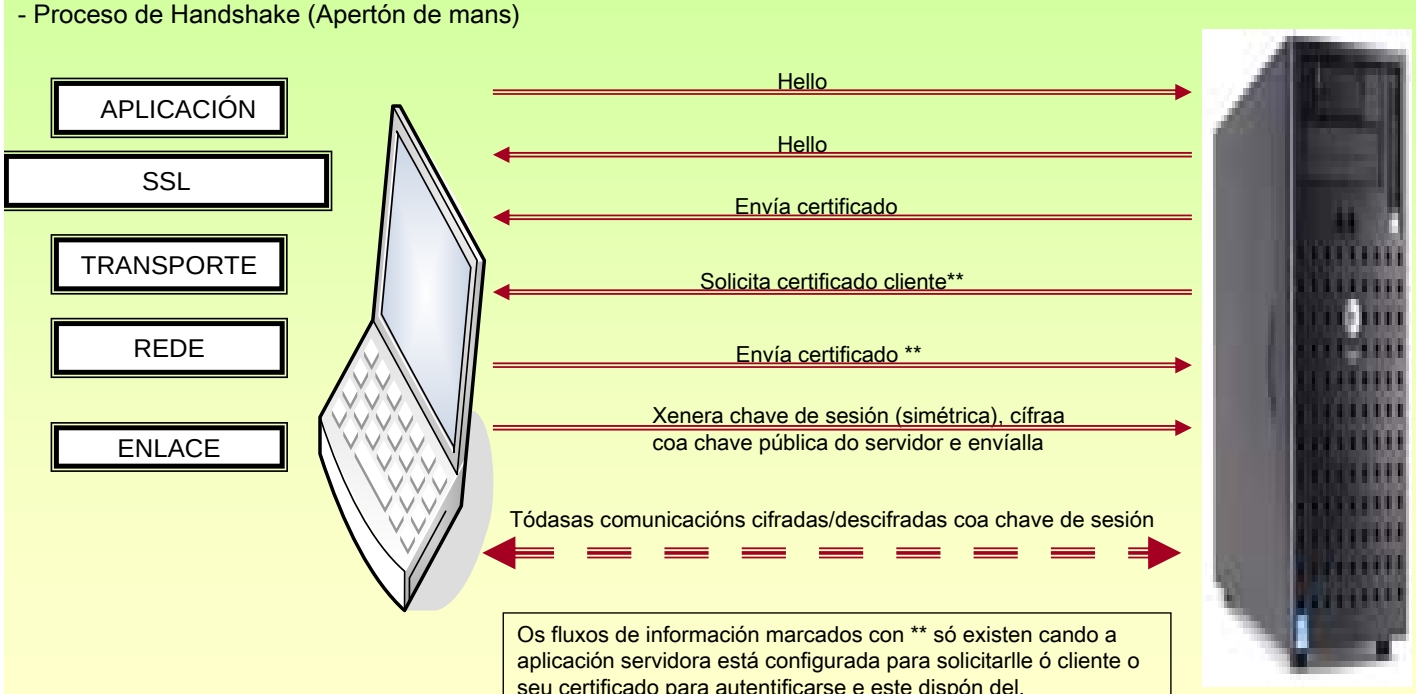
Conexións SSL.
Realizando dobre clic sobre o candado vense as propiedades do certificado.

1.- Introducción - TCP/IP

☞ SSL (Secure Socket Layer, Capa de Sockets Seguros) (I)

-Creado no 1944 por Netscape. Permite crear *túneles* seguros entre unha aplicación cliente e a aplicación servidor

- Proceso de Handshake (Apertón de mans)



Os fluxos de información marcados con ** só existen cando a aplicación servidora está configurada para solicitarlle ó cliente o seu certificado para autentificarse e este dispón del.

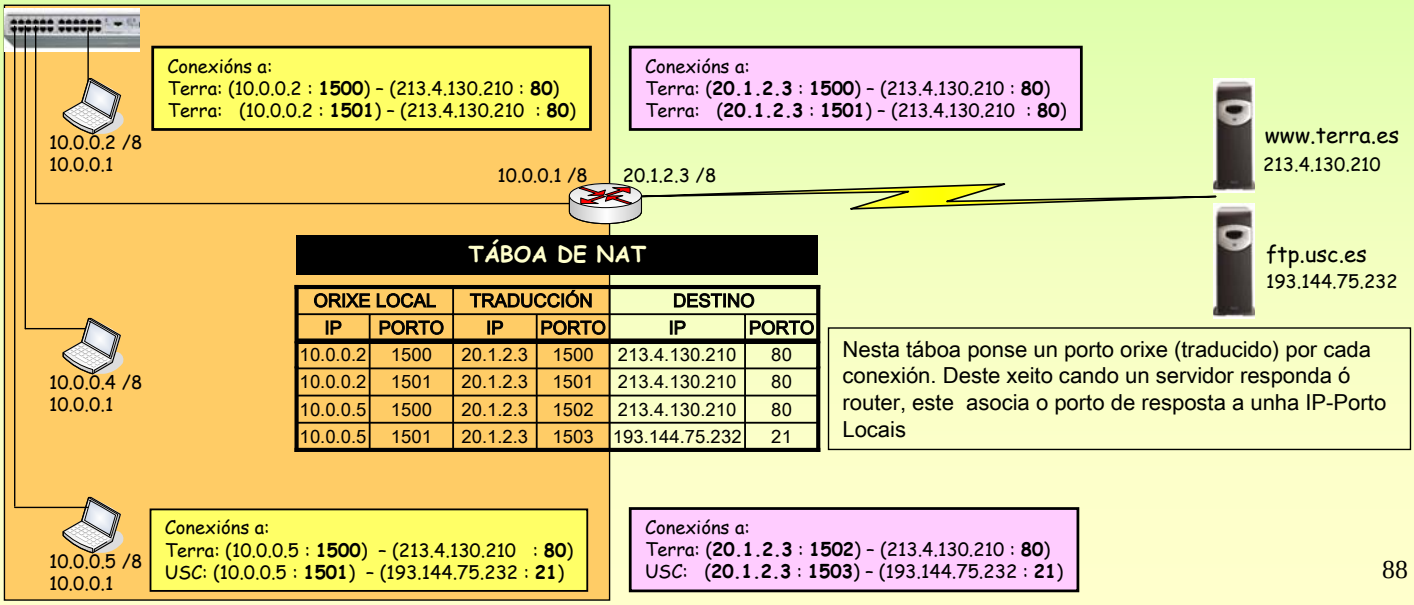
1.- Introducción - TCP/IP

☞ NAT (Network Address Translation – Traducción de enderezos de rede) (I)

Un host cunha IP privada establece unha conexión cun Host cunha IP pública. Pero o host coa IP pública non sabe como chegar o host coa IP privada. Esto é unha conexión ten que ser entre dúas **IPs PÚBLICAS.**

Solución: O router realiza NAT, esto é el pon a súa IP pública como orixe do paquete, e modifica o porto orixe.
Esta táboa constrúese dinamicamente a medida que os hosts locais inician conexións co exterior.

PROCESO DE ENVÍO DE LOCAL A PÚBLICO



TÁBOA DE NAT

ORIXE LOCAL		TRADUCCIÓN		DESTINO	
IP	PORTO	IP	PORTO	IP	PORTO
10.0.0.2	1500	20.1.2.3	1500	213.4.130.210	80
10.0.0.2	1501	20.1.2.3	1501	213.4.130.210	80
10.0.0.5	1500	20.1.2.3	1502	213.4.130.210	80
10.0.0.5	1501	20.1.2.3	1503	193.144.75.232	21

Nesta táboa ponse un porto orixe (traducido) por cada conexión. Deste xeito cando un servidor responda ó router, este asocia o porto de resposta a unha IP-Porto Locais

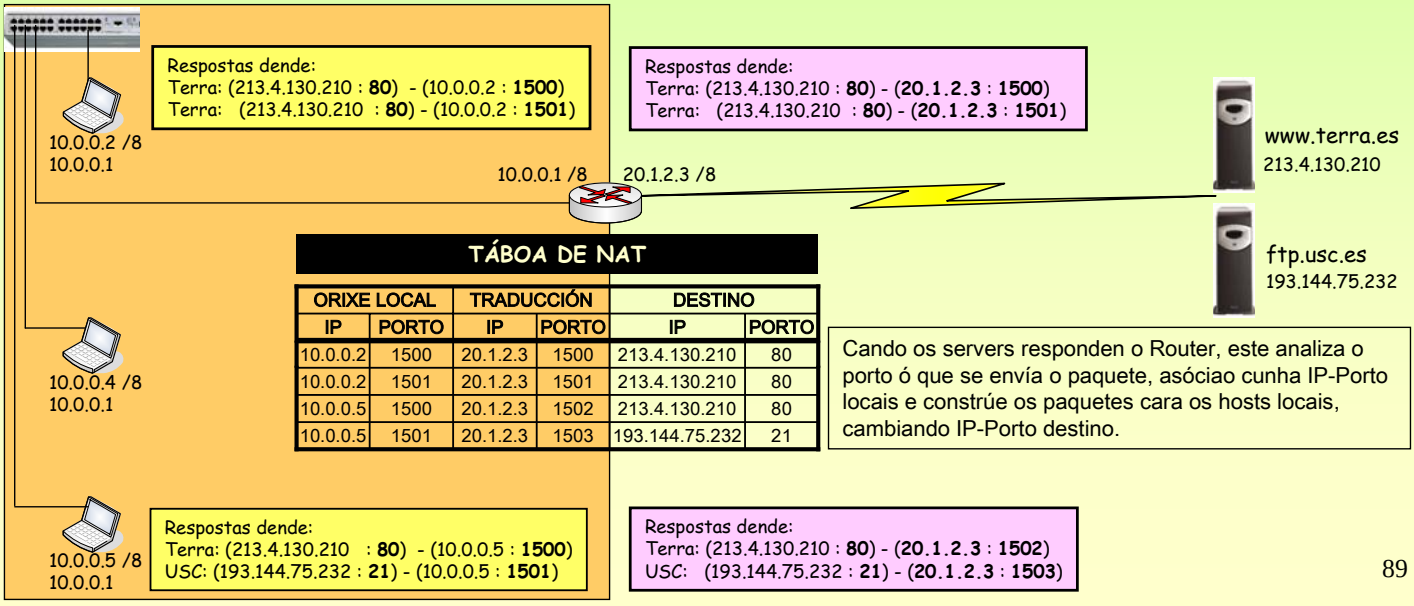
1.- Introducción - TCP/IP

NAT (Network Address Translation – Traducción de enderezos de rede) (II)

Un host cunha IP privada establece unha conexión cun Host cunha IP pública. Pero o host coa IP pública non sabe como chegar o host coa IP privada. Esto é unha conexión ten que ser entre dúas **IPs PÚBLICAS.**

Solución: O router realiza NAT, esto é el pon a súa IP pública como orixe do paquete, e modifica o porto orixe. Esta táboa constrúese dinamicamente a medida que os hosts locais inician conexións co exterior.

PROCESO DE RESPOSTA DE PÚBLICO A LOCAL



TÁBOA DE NAT

ORIXE LOCAL		TRADUCCIÓN		DESTINO	
IP	PORTO	IP	PORTO	IP	PORTO
10.0.0.2	1500	20.1.2.3	1500	213.4.130.210	80
10.0.0.2	1501	20.1.2.3	1501	213.4.130.210	80
10.0.0.5	1500	20.1.2.3	1502	213.4.130.210	80
10.0.0.5	1501	20.1.2.3	1503	193.144.75.232	21

Cando os servers responden o Router, este analiza o porto ó que se envía o paquete, asóciao cunha IP-Porto locais e constrúe os paquetes cara os hosts locais, cambiando IP-Porto destino.