

# Redes

## Tema 2: Capa de aplicación

Oscar García Lorenzo

Escola Politécnica Superior de Enxeñaría

# Índice

- 1 Introducción
- 2 Protocolo de transferencia de hipertexto
- 3 Protocolo de transferencia de archivos
- 4 Protocolos de correo electrónico
- 5 DNS: Servicio de nombres de dominio
- 6 Distribución de contenidos

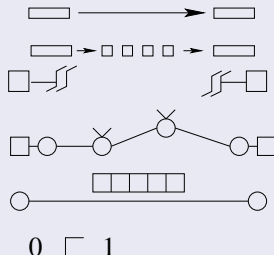
# Índice

- 1 **Introducción**
- 2 Protocolo de transferencia de hipertexto
- 3 Protocolo de transferencia de archivos
- 4 Protocolos de correo electrónico
- 5 DNS: Servicio de nombres de dominio
- 6 Distribución de contenidos

# Arquitectura en capas

## A arquitectura TCP/IP

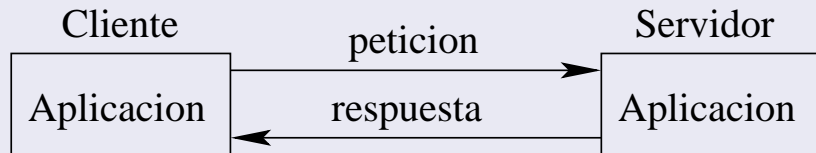
APLICACION	genera mensajes
TRANSPORTE	prepara los mensajes para transmitir
RED	busca rutas
ENLACE	transmite bloques de datos a través de un enlace
FISICA	detalles a nivel de bit



# Introducción

## Capa de aplicación

- Ocupase da comunicación entre procesos



# Capa de aplicación

## Protocolos de comunicación

- Necesarios para la comprensión de los mensajes entre los procesos
- Facilitan la programación de las funciones de envío y recepción

## Deben especificar

- Tipo de mensajes que se intercambian: petición, respuesta, etc.
- Reglas que especifican cuándo y cómo se envían los mensajes
- Sintaxis de los mensajes: campos
- Semántica de cada campo

# Capa de aplicación

## Servizos que necesita a aplicación de rede

- Os protocolos da capa de aplicación usan protocolos básicos da capa de transporte
- Transferencia fiábel de datos ou non fiábel
  - TCP proporciona un servizo fiábel e orientado a conexión
  - UDP un servizo non fiábel máis sinxelo pero rápido e non orientado a conexión
- Ancho de banda
- Temporización

# Servizo orientado a conexión

## Fases

- Establecemento da conexión
  - O cliente solicita unha conexión
  - Fixanse parámetros
  - Ambos os extremos prepáranse para a transmisión
- Transmisión de datos
- Desconexión: terminan a transmisión e libéranse os recursos



# Servizo orientado a conexión

## Características

- *Segmentación*: TCP recolle datos que a aplicación escribe no socket e forma paquetes (Maximum Segment Size - MSS)
- *Transferencia fiable*: o receptor envía confirmacións (ACK). Se o emisor non recibe ACK dun paquete, retransmítese
- *Control de fluxo*: permite que o receptor controle a taxa de envío do emisor.
  - TCP ten un mecanismo para que o receptor lle indique ao emisor a taxa de datos que pode aceptar
- *Control de conxestión*: permite que a taxa de envío do emisor se axuste ás capacidades da rede

# Servizo non orientado a conexión

## Servizo sen conexión

- Non hai fase de establecemento de conexión
- Non hai confirmacións: o emisor descoñece se o paquete chegou ó destino
- Non hai control de fluxo nen control de conxestión

⇒ transmisión máis rápida, aínda que menos fiábel

## Na Internet: UDP

- TCP: cando se necesite fiabilidade, telnet, FTP, SMTP, HTTP, etc.
- UDP: velocidade e non importa a perda de paquetes, telefonía IP, videoconferencia

# Capa de aplicación

## Protocolos a tratar

TCP	HTTP (web) SMTP, POP3 e IMAP (correo) FTP (ficheiros)
UDP	DNS (traduccions)

## Axente de usuario

- Interface entre o usuario e a aplicación (navegador, xestor de correo, etc.)

# Índice

- 1 Introducción
- 2 Protocolo de transferencia de hipertexto
- 3 Protocolo de transferencia de archivos
- 4 Protocolos de correo electrónico
- 5 DNS: Servicio de nombres de dominio
- 6 Distribución de contenidos

# HTTP

## Protocolo de transferencia de hipertexto

- Protocolo que define a comunicación entre un servidor web e un cliente web
- Usa TCP
- É un protocolo sen estado (ás veces mecanismos como cookies)
- Usa por defecto o porto 80

# HTTP

## Conceptos

- Páxina web (ou documento): consta de obxectos (arquivos direccionables por un URL)
- As páxinas web: arquivo HTML base (conten a distribución e, en xeral, o texto) e varios obxectos
- Navegador: axente do usuario para a web
- Servidor web: alberga obxectos
- Solicitud dunha páxina web  $\implies$  o servidor devolve o documento base

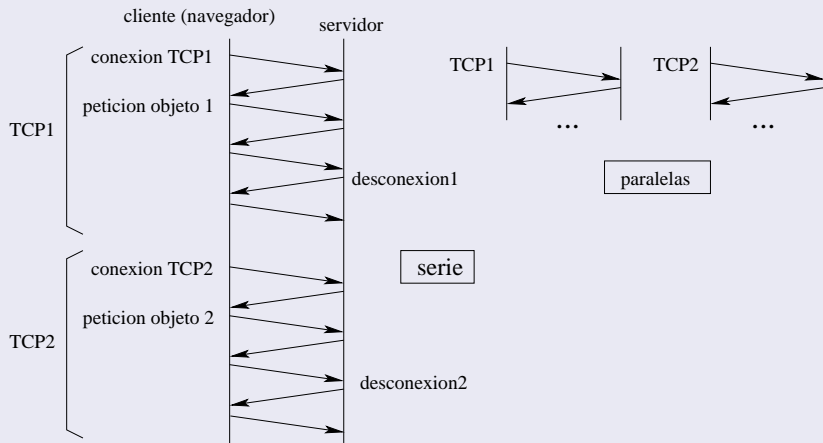
# HTTP

## Conexións HTTP

- Non persistentes: usase unha conexión TCP distinta para transferir cada obxeto (HTTP/1.0)
  - Serie: esperase a que acabe a conexión TCP previa
  - Paralelo: inicianse varias conexións TCP á vez
- Persistentes: pódense transferir varios obxetos (e incluso varias páxinas) coa mesma conexión TCP
  - Sen entubamento: o cliente pide un novo obxeto cando se recibiu o previo
  - Con entubamiento: o cliente pode facer peticións de varios obxetos antes de recibir os anteriores (modo por defecto en HTTP/1.1)

# HTTP

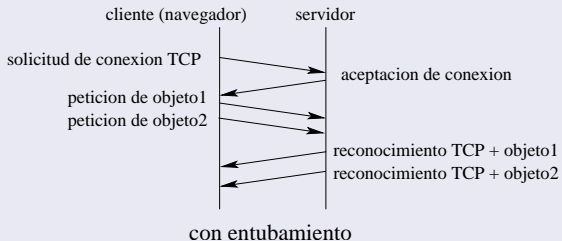
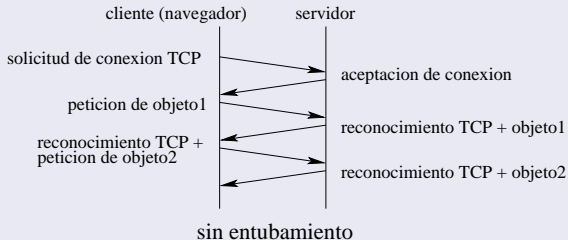
## HTTP non persistente



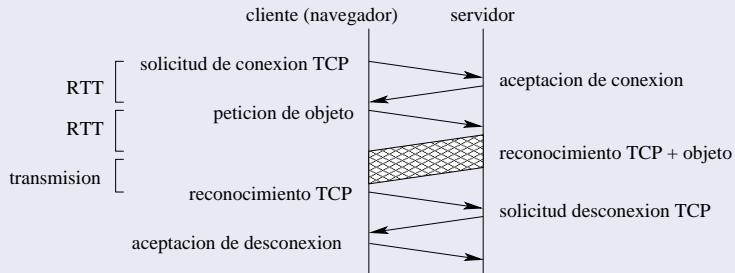


# HTTP

## HTTP persistente



## Conexións non persistentes e serie



# Tempo de transferencia dunha páxina web

## Parámetros

- RTT (tempo de ida e volta): tempo necesario para que un paquete pequeno vaia do cliente ao servidor e volva ao cliente
- Tempo de transmisión do arquivo: depende do tamaño do arquivo

## Todos os tipos de conexións

- O primeiro obxeto transfírese en  $2RTT + t_{transmission}$
- Os seguintes obxectos dependen do tipo de conexión

# HTTP

## Mensaxes HTTP

Dous tipos:

- Petición: para pedir obxetos
- Resposta: conteñen os obxetos

Ambos mensaxes teñen:

- Cabeceira: información de control, en ASCII de 7 bits
- Corpo: os datos (un obxeto, contido dun formulario, etc.) e son en binario

# Mensajes HTTP

## Mensajes de petición

cabecera	{	línea de petición	metodo	sp	URL	sp	version	cr	lf	
		líneas de cabecera	nombre campo cabecera				sp	valor	cr	lf
			nombre campo cabecera				sp	valor	cr	lf
cuerpo	{	línea en blanco	cr	lf						
		cuerpo								

# Mensajxes HTTP

## Mensaxes de petición

- Liña de petición: obxeto solicitado
  - Método: GET (páxina normal) POST (formulario)
  - Menos comúns: HEAD (sen devolución, debug) DELETE (borrar)

- Liñas de cabeceira: opcións

Host:	nome servidor web (pode haber varios para unha IP)
Connection:	close (se queremos conexións non persistentes)
User-agent:	mozilla/4.0 (o navegador)
Accept-languaje:	es (o idioma preferido da páxina, se o hai)
IF-modified-since	Wed, 15 Sep 2021 09:23:24 (so se se modificou)

- Corpo
  - GET: baleiro
  - POST: os datos do formulario

# Mensaxes HTTP

## Mensaxes de resposta

cabecera	línea de estado	version	sp	cod. estado	sp	frase	cr	lf
		nombre campo cabecera		sp	valor	cr	lf	
	líneas de cabecera							
		nombre campo cabecera		sp	valor	cr	lf	
cuerpo	línea en blanco	cr	lf					
		objeto						

# Mensaxes HTTP

## Mensaxes de resposta

- Liña de estado

200	OK (o obxecto ou páxina servese sen problemas)
404	Not found (a páxina non existe)
400	Bad Request (no se entendi� el formato de la petici�n)
304	Not Modified (pediuse con GET condicional If-modified-since)

- Li as de cabeceira: opci ns

Connection:	close (indica que se usan conexi�ns non persistentes)
Date:	data de env�o da p�xina
Server:	Apache/1.3.0 (o servidor web)
Last-Modified:	cando se creou ou modificou por �ltima vez a p�xina
Content-Length:	tama�o en bytes do obxecto, p�xina ou imaxe
Content-Type:	text/html, image/gif, image/jpeg

- Corpo: os datos



# Mensaxes HTTP

## Cookies

- Liñas de cabeceira
  - set-cookie
  - cookie
- Arquivo común no cliente (navegador)
- Base de datos no servidor
- Número identificativo xerado no servidor
- Mensaxe petición
  - no primeiro nada
  - despois devolve a cookie: cookie id
- Mensaxe de resposta
  - crea a cookie : set-cookie id
- Garda estado

# HTTP/2

## Nova versión do protocolo HTTP

- Baseado no protocolo SPDY, desenvolto por Google (64 % máis rápido)
- Especificación publicada en maio de 2015
- Soportado polas últimas versións de moitos navegadores

## Melloras

- Multiplexación de peticións HTTP sobre unha mesma conexión TCP
- Compresión de cabeceiras
- Soporte para *server push*
- Pipelining de solicitude–resposta
- Cifrado da información: uso obrigatorio de SSL

# HTTP/3

## ALERTA: Tomado da wikipedia

- Outra vez desenvolto por Google (problemas multiplexado)
- Especificación todavía en borrador
- Soportado polo 73 % dos navegadores
- Chrome e Edge, abril 2020, Firefox, maio 2021
- 21 % das páxinas web top 10 millóns

## Melloras

- Capa de transporte cambiada a QUIC (desde TCP)
- QUIC: Quick UDP Internet Connections, pero agora considerada so un nome
- UDP con cambios para control de conxestión
- Mesmos mensaxes e códigos
- Require retoques para adaptarse aos paquetes perdidos

# Índice

- 1 Introducción
- 2 Protocolo de transferencia de hipertexto
- 3 Protocolo de transferencia de archivos**
- 4 Protocolos de correo electrónico
- 5 DNS: Servicio de nombres de dominio
- 6 Distribución de contenidos

# FTP

## Protocolo de transferencia de archivos

- Define a comunicación cun servidor de arquivos
- Con estado: manteña información durante toda a sesión
- Usa dúas conexións TCP paralelas (información de control fora de banda)
  - Conexión de control
    - Usa o porto 21
    - Para enviar os comandos: nome e chave de usuario, *dir*, *put*, *get* e recibir as respostas
    - É persistente: dura todo o tempo da conexión FTP
    - Usa ASCII de 7 bits
  - Conexión de datos
    - Usa o porto 20
    - Para transmitir datos en resposta aos comandos
    - É non persistente: abrese e péchase unha conexión nova por cada arquivo que se transmite
    - Usa ASCII de 7 bits

# FTP

## Comandos e respostas FTP

- Pola conexión de control
- Comandos: constan de 4 caracteres en maiúsculas e campos adicionais acabados en cr+lf

USER	nome de usuario
PASS	chave
LIST	lista de arquivos
RETR	nombre de ficheiro (traer ficheiro)
STOR	nombre de ficheiro (almacenar ficheiro)

- Respostas: un código de 3 díxitos e unha frase explicativa

331	Username OK, password required
125	Data connection already open; transfer starting
425	Can't open data connection
452	Error writing file

## Transmisión cifrada

- FTP, telnet y HTTP sin cifrar. SFTP, SSH y HTTPS no

# Índice

- 1 Introducción
- 2 Protocolo de transferencia de hipertexto
- 3 Protocolo de transferencia de archivos
- 4 Protocolos de correo electrónico**
- 5 DNS: Servicio de nombres de dominio
- 6 Distribución de contenidos

# Correo electrónico

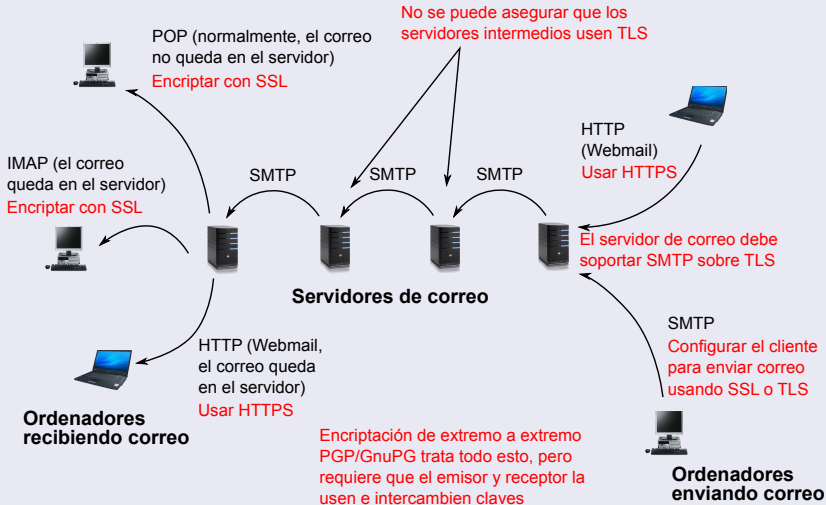
## Protocolos de correo electrónico

- Para enviar correo ao servidor de correo ou entre servidores: SMTP, HTTP (se se usa un navegador)
- Acceso ao correo polo axente de usuario: POP3, IMAP, HTTP (se se usa un navegador)



# Correo electrónico

## Protocolos de correo electrónico

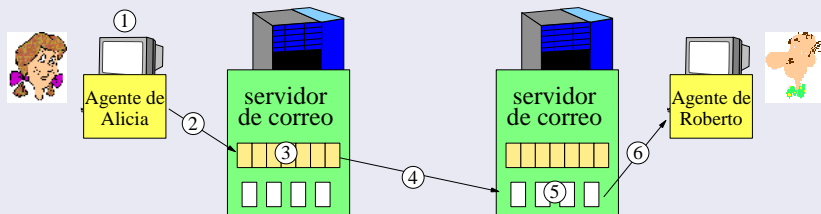


# SMTP

## Características do funcionamento do correo electrónico

- Caixa de correo no servidor
- O axente de usuario (AU) comunícase co servidor
- Se o servidor destino está fora de servizo, reintentase (30 minutos)
- SMTP usa o porto 25
- Usa conexións TCP persistentes: mantén aberta a conexión durante toda a sesión (varios mensaxes)
- Emprega ASCII de 7 bits tanto para as cabeceiras como para os corpos. Unha imaxe, o AU a transforma en ASCII (protocolo de 1982)

## Funcionamiento do correo electrónico



- 1 Alicia usa sue AU para escribir unha mensaxe
- 2 O AU de Alicia envía a mensaxe ao sue servidor de correo: almacenase nunha cola de mensaxes
- 3 O lado cliente de SMTP (servidor de Alicia) abre unha conexión TCP co servidor de Roberto
- 4 Tras unha sincronización inicial (envío dirección destino e remite), o cliente envía a mensaxe sobre a conexión TCP
- 5 O servidor de Roberto o deposita na súa caixa de correo
- 6 Roberto usa seu AU para ler a mensaxe

# SMTP

## Mensaxes SMTP

Tres tipos de mensaxes:

- Comandos: palabra en maiúscula seguida dos parámetros

HELO	nome servidor
MAIL FROM:	dirección de correo do remitente
RCPT TO:	dirección de correo do destinatario
DATA	(o contido do correo)
QUIT	

- Respostas: código numérico seguido dunha frase aclarativa

220	nome do servidor
250	comando que se executou satisfactoriamente
354	envíe o correo, terminando en .
221	peche de conexión

- Datos: o contido dos correos: todos os obxectos encapsulados nun único arquivo codificado en ASCII de 7 bits (coas chamadas extensiones MIME)

# SMTP: Exemplo de transacción entre cliente e servidor

Cliente (C) e servidor (S): `telnet smtp.usc.es 25`

```
S: 220 vm075043.usc.es ESMTP Postfix (ATIC-USC)
C: HELO smtp.usc.es
S: 250 vm075043.usc.es
C: MAIL FROM: pepito@gmail.com (que non sexa @usc)
S: 250 2.1.0 Ok
C: RCPT TO: juanito@usc.es
S: 250 2.1.0 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: Ola
C: Isto é unha proba, adeus
C: .
S: 250 2.0.0 Ok: queued as 6A7084E9
C: quit
S: 221 2.0.0 Bye
```

# SMTP

## Protocolo inseguro

- Non se pide nome e chave de usuario: permite enviar correos a calquera

## Comparación con HTTP

- Ambos transmiten arquivos e usan conexións TCP persistentes
- HTTP é un protocolo de demanda (o cliente demanda o arquivo) e SMTP de oferta (o cliente oferta o arquivo)
- SMTP require que a cabeceira e o corpo sexan de ASCII de 7. En HTTP o corpo pode conter datos binarios
- HTTP envía obxectos en arquivos diferentes, SMTP van encapsulados no mesmo arquivo que contén o texto
- HTTP non tene estado e SMTP debe recordar a fase da sesión na que se atopa

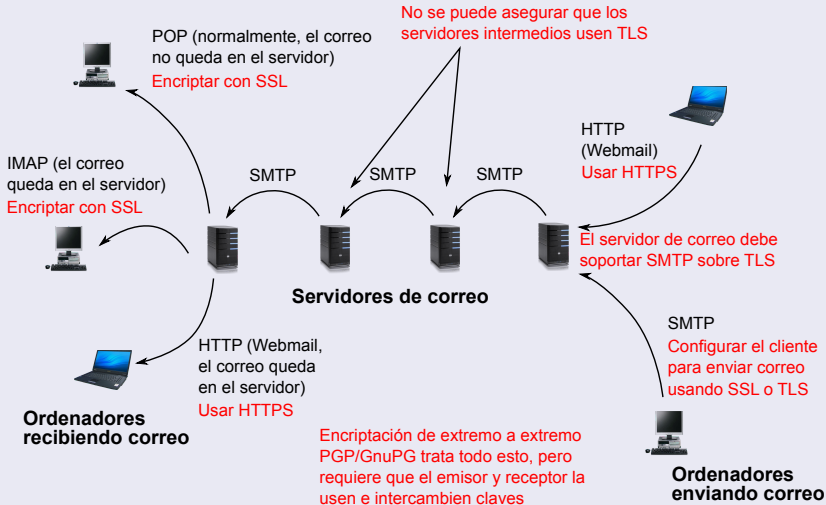
# Correo electrónico

## Protocolos de acceso ao correo

- Os axentes de usuarios para transferir o correo da caixa destino ao computador local
- Non se usa SMTP porque é un protocolo de oferta  $\Rightarrow$  computador local sempre acendido esperando correos
- Usanse protocolos que permiten descargar os correos: POP3, IMAP y HTTP

# Correo electrónico

## Protocolos de correo electrónico





# POP3: Protocolo de oficina postal, versión 3

## POP3

- Protocolo sinxelo para o acceso ao correo
  - Descargar e borrar do servidor
  - Descargar e manter no servidor
- Conexión TCP co porto 110 do servidor

## Protocolo simple: tres fases

- Autorización: autenticación do usuario
- Transacción
  - Recuperar as mensaxes
  - Marcar/desmarcar para borrado
  - Estadísticas de correo
- Actualización: borrado dos mensaxes marcados
  - Termina a sesión POP3

# POP3: Protocolo de oficina postal, versión 3

## Mensajes POP3

Tres tipos de mensajes:

- Comandos: palabra de 4 caracteres seguida dos parámetros

user	nome do usuario
pass	palabra chave
list	
retr	número de correo (traer correo)
dele	número de correo (borrar correo)
quit	

- Respuestas: básicamente de dous tipos

+OK frase explicativa -ERR frase explicativa
---

- Datos: datos da resposta
  - A lista de mensaxes, os contidos dos correos, etc.
  - O correo completo nunha única mensaxe (SMTP xeraba unha mensaxe por cada liña de cabeceira)

# POP3: Protocolo de oficina postal, versión 3

## Outras características

- Usa conexións persistentes: usase a mesma conexión TCP mentras dure a sesión na que se poden descargar varios correos
- POP3 podese usar para descargar e borrar (retr+dele) os correos do servidor ou para descargar e manter no servidor (retr)
- Necesita conta con contrasinal
- Non manteñen información de estado entre sesións POP3
- Definido no RFC 1939

# IMAP: Protocolo de acceso a mensaxes de Internet

## Inconvintes de POP3

- Acceso ao correo en múltiples dispositivos
- Non permite cartafols no servidor

## IMAP

- Máis funcionalidades que POP3  $\implies$  máis complexo
- O servidor asocia cada mensaxe cun cartafol
  - Cando chega unha mensaxe vai ao cartafol INBOX
  - Permite ler, borrar, mover a outro cartafol...
- Comandos para crear cartafols, realizar buscas, descargar partes da mensaxe...
- Mantén información de estado entre sesións
- Definido no RFC 3501

# Índice

- 1 Introducción
- 2 Protocolo de transferencia de hipertexto
- 3 Protocolo de transferencia de archivos
- 4 Protocolos de correo electrónico
- 5 DNS: Servicio de nombres de dominio**
- 6 Distribución de contenidos

# DNS

## Servizo de nomes de dominio

- Para traducir nomes de hosts  $\Leftrightarrow$  direccións IP

## Compoñentes

- Numerosos servidores de nomes distribuídos por Internet
- A base de datos do DNS está distribuída de forma xerárquica
- Un protocolo que permite aos hosts pedir traducións aos servidores e que os servidores se intercambien datos entre eles

## Protocolo sen conexión

- DNS usa UDP, no porto 53 e non ten estado

# DNS

## Servicios proporcionados

- Traducir nomes de hosts  $\Leftrightarrow$  direcciones IP e obtención de alias
- Informa dos servidores autorizados para un dominio
- Alias de servidores de correo: permite simplificar as direccións
- Distribución da carga
  - Asígnanse varias IPs a un nome de host
  - Servidores espello: co mesmo contido
  - Devolve de forma cíclica unha IP do conxunto das asignadas

# DNS

## Tipos de servidores

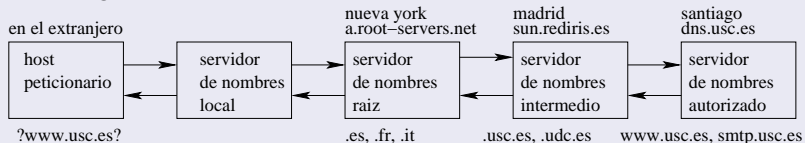
- Servidores locais: atenden as consultas dos hosts
- Servidores autorizados o autoritativos
  - Para que un host sexa accesibel en Internet, debe estar rexistrado nun servidor autorizado
  - Normalmente pertencen ao ISP
  - Cada host debe estar en dous servidores autorizados por fiabilidade
  - Moitos servidores autorizados comportanse como locais
- Servidores raíz: sobre 400 na Internet, xestionados por 13 organizacións
  - Información dos dominios de primer nivel (TLD)
- Servidores intermedios ou TLD: información sobre os niveis intermedios



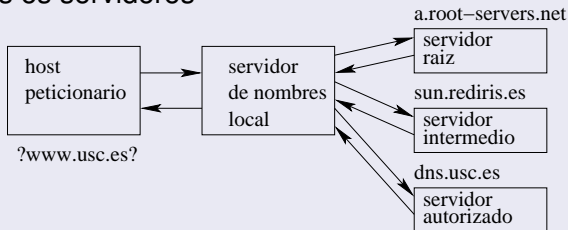
# DNS

## Tipos de consultas

- Consultas recursivas: cada servidor DNS interroga al siguiente



- Consultas iterativas: el servidor DNS local contacta con todos los servidores



# DNS

## Caché de DNS

- Os servidores almacenan copias locais das correspondencias que obteñen
- Borranse despois dun certo tempo sen usar (dous días)
- En todos os niveis da xerarquía, incluso nos hosts locais

# DNS

## Mensaxes DNS

### Consultas e respostas:

- Cabeceira con información de control
  - Identificación: 16 bits que identifican a consulta (e a sua correspondente resposta)
  - Sinais: 4 bits que indican se consulta ou resposta, tipo de consulta, etc.
  - Tamaño dos campos do corpo
- Corpo: 4 campos
  - Cuestións: unha ou varias preguntas. Nome ou dirección a traducir, etc
  - Respostas: unha ou varias respostas. Parella host/IP
  - Servidores autorizados. Permite facer unha cadea de consultas
  - Información adicional

# DNS

## Mensajes DNS

cabecera	identificación	senales
	num. cuestiones	num. respuestas
	num. s. autorizados	num. inf. adicional
cuerpo	cuestiones	
	respuestas	
	servidores autorizados	
	información adicional	

Ejemplo:

1	1
2	2

www.usc.es?

www.usc.es → 193.144.74.224

usc.es → dns.usc.es, dns2.usc.es

direcciones IP de los servidores autorizados

# DNS

## Mensaxes DNS

Esos campos conteñen rexistros de recursos: 4-tupla

Tipo	Nome	Valor	TTL
A	nome de host	dirección IP	
NS	dominio	servidor autorizado para o dominio	
CNAME	alias	nome de host	
MX	alias de correo	servidor de correo	
AAAA	nome de host	dirección IPv6	

TTL: tempo durante o cal é válida a resposta

# Índice

- 1 Introducción
- 2 Protocolo de transferencia de hipertexto
- 3 Protocolo de transferencia de archivos
- 4 Protocolos de correo electrónico
- 5 DNS: Servicio de nombres de dominio
- 6 Distribución de contenidos

# Distribución de contidos

## Proporcionar contidos

O acceso a servidores centralizados pode ser lento:

- O camiño das mensaxes é lento ou está conxestionado
- O servidor está sobrecargado

⇒ distribúense (e duplican) os contidos en distintas zonas e diríxense as peticións ao servidor de menor tempo de resposta

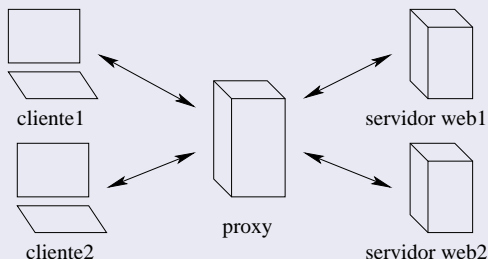
## Métodos

Método de distribución	Quen proporciona a infraestrutura
Caché web (ou servidor proxy)	ISP
Redes de distribución de contidos	distribuidor de contidos
Redes P2P (de igual a igual)	usuario

# Distribución de contidos

## Caché web ou servidor proxy

- Un servidor intermedio (proxy) polo que pasan todas as peticións web dos hosts dunha rede
- Mantén copias dos contidos durante certo tempo
- O usuario debe configurar o navegador
- Permite un esquema xerárquico
- Proporcionado polo ISP





# Distribución de contidos

## Redes de distribución de contidos (CDN)

- Empresas que posúen *centros de host de Internet* (compañía CDN) alquilan a súa infraestrutura
- A CDN replica os contidos dos seus clientes nos servidores CDN e os mantén actualizados
- A CDN proporciona un mecanismo para que o contido sexa entregado polo servidor CDN que poida facelo máis rapidamente
- Acceso aos contidos mediante dúas técnicas:
  - Redirección de obxectos
  - Balanceo das peticións usando o DNS

# Redes de distribución de contenidos (CDN)

## CDN

- Akamai
  - Introducción profunda, perto usuarios, miles servidores
- Limelight, outras
  - ISP, menos servidores, puntos críticos
- DNS para resolver a dirección do video
- DNS devolve a páxina do CDN
- Estatexias de selección de clústeres propietarias

# Distribución de contidos

## Redes P2P

- Todos os usuarios son á vez servidores e clientes usando unha aplicación
- Necesitase un nodo de arranque
- Necesitase construír un directorio

Tipo de directorio	Exemplo
centralizado	napster
non centralizado	kaaza
inundación de consultas	gnutella

- BitTorrent
  - Cada arquivo compartese nunha rede P2P propia
  - Necesita un *tracker* para cada rede
  - Non necesita construír un directorio. A información de cada arquivo vai nos .torrent

# Distribución de contidos

## Vídeo por Internet

- HTTP unha soa mensaxe
  - Mesma versión para todos
  - Mensaxe grande, pode fallar
- DASH, (Dynamic Adaptive Streaming over HTTP, Fluxos dinámicos adaptativos sobre HTTP)
  - Tasa variabel
  - HTTP pequenas mensaxes
  - Diferentes versións

# Distribución de contenidos

## Aplicacións P2P

- Escalabilidade das arquitecturas P2P
- BitTorrent
- Tablas hash distribuidas (DHT)
- Telefonía sobre Internet: Skype

## Fluxos de vídeo e redes de distribución de contido

- Vídeo por Internet
- Fluxos de vídeo HTTP e tecnoloxía de fluxos dinámicos adaptativos sobre HTTP (DASH)
- Redes de distribución de contenidos (CDN)
- Casos de estudo: Netflix, YouTube y Kankan