

---

# Capítulo 2. IP

---

Redes de Ordenadores  
2º Grado en Ingeniería en Tecnologías de Telecomunicación



# Índice

*Hora 1*

1 Características de IP

2 Cabecera IPv4

2.1 Opciones de cabecera IPv4

*Hora 2*

3 Direccionamiento IP

3.1 Direccionamiento classful

3.2 Subnetting

3.2.1 Variable-Length Subnet Mask (FLSM)

3.2.2 Variable-Length Subnet Mask (VLSM)

*Hora 3*

3.3 Supernetting

3.4 Direccionamiento classless

3.4.1 Classless Interdomain Routing (CIDR)

3.5 Direcciones IP especiales

3.6 Direccionando una red

*Hora 4*

4 Router

5 Reenvío

5.1 Reenvío en redes classful

5.2 Reenvío en redes con subnetting VLSM y FLSM

5.3 Reenvío en redes classless – CIDR

6 Fragmentación y reensamblado

# Índice hora 1

## *Hora 1*

### 1 Características de IP

### 2 Cabecera IPv4

#### 2.1 Opciones de cabecera IPv4

## *Hora 2*

### 3 Direccionamiento IP

#### 3.1 Direccionamiento classful

#### 3.2 Subnetting

##### 3.2.1 Variable-Length Subnet Mask (FLSM)

##### 3.2.2 Variable-Length Subnet Mask (VLSM)

## *Hora 3*

#### 3.3 Supernetting

#### 3.4 Direccionamiento classless

##### 3.4.1 Classless Interdomain Routing (CIDR)

#### 3.5 Direcciones IP especiales

#### 3.6 Direccionando una red

## *Hora 4*

### 4 Router

### 5 Reenvío

#### 5.1 Reenvío en redes classful

#### 5.2 Reenvío en redes con subnetting VLSM y FLSM

#### 5.3 Reenvío en redes classless – CIDR

### 6 Fragmentación y reensamblado

# Objetivos

- Conocer las funcionalidades del protocolo de nivel de red de Internet: IP
- Entender la función de cada campo de la cabecera IP junto con sus campos opcionales

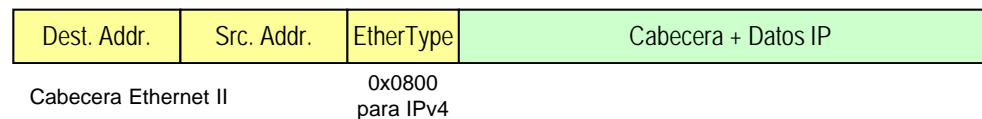
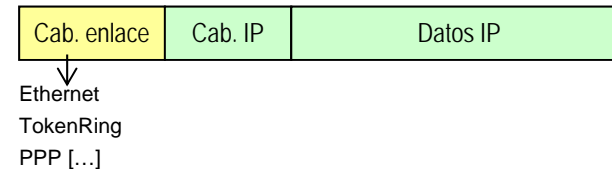
# 1 Características IP

*RFC791(STD5), RFC1349*

- Internet Protocol (IP) es un protocolo de nivel de red.
- Servicio de datagramas, no orientado a conexión: cada datagrama se maneja independientemente de los demás.
  - Desorden (distintas rutas para cada datagrama).
- Servicio no fiable: no hay garantías (Best Effort). Los datagramas:
  - Se pueden corromper.
  - Pueden llegar duplicados.
  - Pueden no llegar.
- IP versiones:
  - IPv4, usada desde los años 70.
  - IPv6, en fase de implantación.

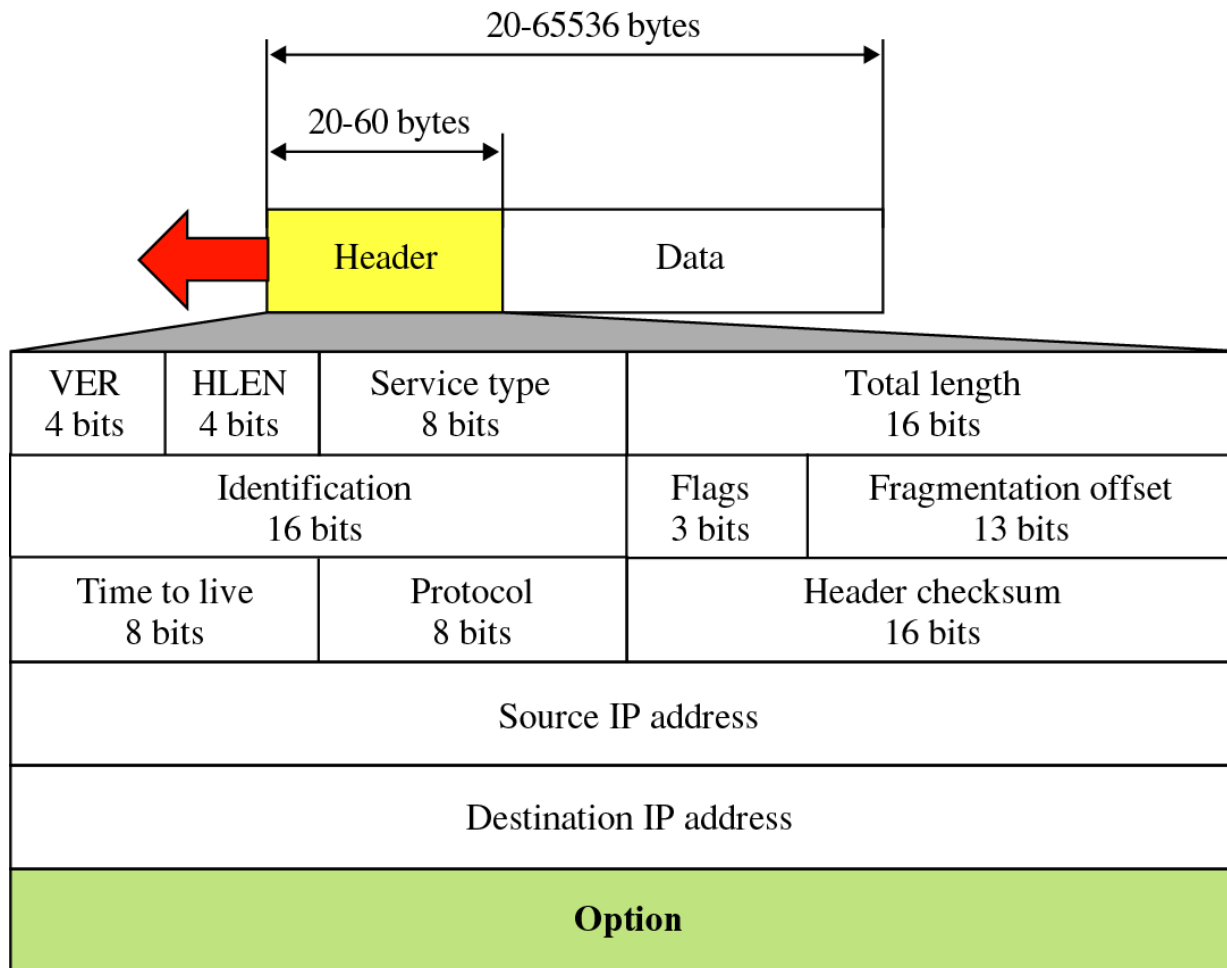
# Características IP

- Diferentes tecnologías de nivel de enlace.
- Diferentes cabeceras de nivel de enlace.



- Protocolo IP trabaja sobre cualquier nivel de enlace (encapsulación)
  - Permite conectividad entre sistemas en diferentes redes con distinta tecnología.
- IP provee:
  - Direccionamiento lógico: permite identificar unívocamente un sistema a través de su dirección IP.
  - Encaminamiento: provee conectividad extremo a extremo.
    - Dispositivo de interconexión: router
  - Fragmentación y reensamblado: atravesar cualquier red independientemente de su MTU.
  - Posibilidad de QoS (Quality of Service): priorización del tráfico. No se usa en Internet pero si dentro de redes empresariales u operadoras.

## 2 Cabecera IPv4



## Campos de la cabecera IPv4

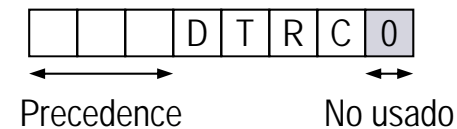
- Cabecera IP, tamaño:
  - mínimo de 20 bytes: sin opciones.
  - máximo de 60 bytes: 40 bytes de opciones. Limitado por HLEN.
- VER (4bits)
  - Versión del protocolo IP. IPv4  $\Rightarrow$  0x4.
  - Determina el formato de los campos que siguen a continuación.
- HLEN (4bits)
  - Header Length, longitud de la cabecera IP en palabras de 4 bytes.
  - Máximo valor: 1111=0xF=15 palabras  $\Rightarrow$  60 bytes
- TOS o DS (8bits)
  - Type of Service (TOS), denominación anterior. RFC1349 (1992).
  - Differentiated Services (DS), denominación actual. RFC2474 (1998).



# Campos de la cabecera IPv4

## □ Type of Service

- Precedence (3 bits), prioridad: 0=baja, 7=alta
- Sólo un bit puede estar a 1 para indicar el tipo de servicio deseado
  - D: minimizar retardo
  - T: maximizar throughput
  - R: maximizar fiabilidad
  - C: minimizar coste
- Este campo ha sido tradicionalmente ignorado en implementaciones IP de hosts o routers, en particular los bits de precedencia.



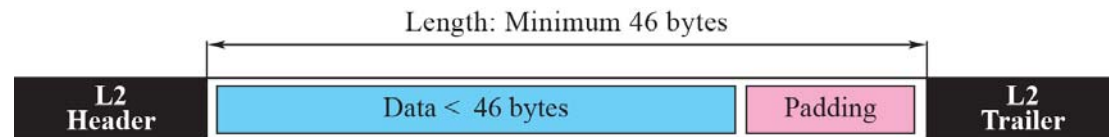
## □ Differentiated Services

- Si DEF=000  $\Rightarrow$  ABC se interpretan como los bits de prioridad compatible con la interpretación TOS.
- Si DEF $\neq$ 000  $\Rightarrow$  ABCDEF definen 56 ( $2^6-2^3$ ) tipos de servicio asignados por diferentes autoridades según los últimos bits del *codepoint*.
  - XXXXX0: Internet
  - XXXX11: Local
  - XXXX01: Temporal o experimental



## Campos de la cabecera IPv4

- Total Length (16bits)
  - Longitud total cabecera+datos del paquete/fragmento IP en bytes.
  - Longitud datos (bytes) = TotalLength – (HLEN\*4)
  - Datagrama IP:
    - Tamaño máximo:  $2^{16}-1 = 65.535$  bytes
    - Tamaño mínimo que ha de procesar toda máquina en Internet: 576 bytes [RFC1032,STD43]
  - Necesario conocerlo para identificar los datos que no sean padding:



- Identification (16bits)
  - Identificador común a todos los fragmentos que proceden del mismo datagrama IP.
    - En el destino, todos los fragmentos con la misma dirección IP origen y mismo identificador, se tomarán para su reensamblado.
  - Dependiendo de la implementación IP, el identificador se genera para cada datagrama por ejemplo a base de un contador secuencial.

## Campos de la cabecera IPv4

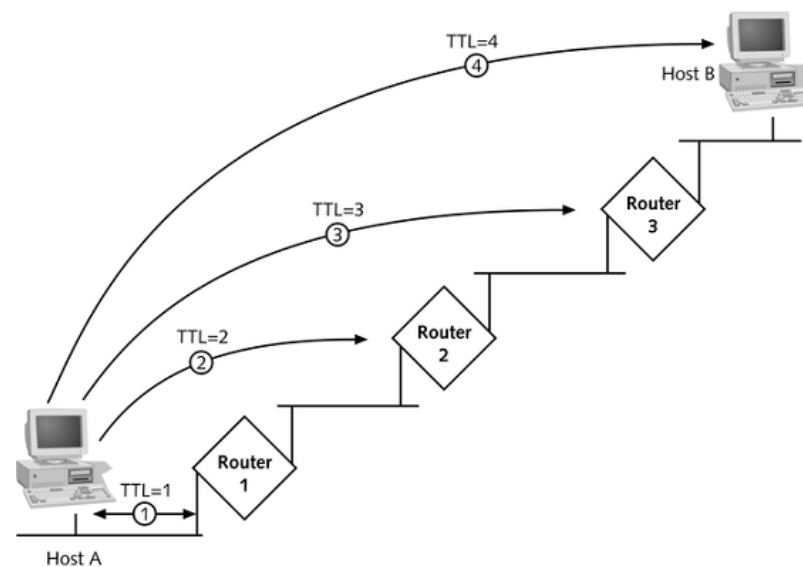
- Flags (3bits) 

0	DF	MF
---	----	----

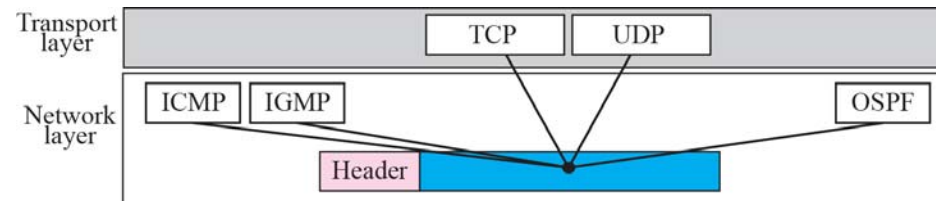
  - Usados en la fragmentación.
  - DF (Don't Fragment), solicitud para que este datagrama no se fragmente en los diversos saltos.
  - MF (More Fragments), vale 0 en el último fragmento o en un datagrama no fragmentado.
- Fragmentation Offset (13bits)
  - Determina la posición del presente fragmento dentro del datagrama original, en palabras de 8 bytes.
    - Vale 0 en el primer fragmento o datagrama no fragmentado.
    - Obligará a que los fragmentos se realicen siempre en múltiplos de 8 bytes de datos IP.
- Time to Live, TTL (8bits)
  - Controla el máximo número de saltos (enlaces entre routers) que un datagrama IP puede realizar.
  - El nodo origen coloca un valor de aproximadamente el doble del  $n^{\circ}$  de saltos con cualquier destino. Cada router que lo procesa decrementa el TTL en una unidad, y si tras ello el valor es 0 el paquete se descarta.

## Campos de la cabecera IPv4

- Valores típicos de TTL en el origen: 64, 128
- TTL se utiliza para evitar que haya paquetes circulando continuamente por la red, por ejemplo debido a tablas de rutas corruptas que creen ciclos.
- En teoría, en su concepción original se medía en segundos, restando 1sg al menos cada router, por lo que en la práctica ha equivalido tradicionalmente a la cuenta de saltos.
- También se utiliza para limitar el alcance del paquete (p.e. traceroute).



## Campos de la cabecera IPv4



### ■ Protocol (8bits)

- Determina el protocolo que se encapsula dentro de la parte de datos del datagrama.
- Valores asignados por el ICANN [RFC3232, actualmente base de datos online]. Típicos:

Valor	Protocolo
0	IP
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

- En Linux, fichero /etc/protocols contiene el listado de protocolos.

## Campos de la cabecera IPv4

- Header Checksum (16bits)
  - Se aplica sólo sobre la cabecera IP.
  - Se calcula en el emisor:
    - Poniendo a 0's el CRC.
    - Suma de la cabecera en palabras de 16 bits (X).
    - El resultado se complementa a 1 y ese será el nuevo CRC (-X) (suma en complemento a 1).
  - En destino, la suma en complemento a 1 de la cabecera en palabras de 16 bits deberá dar 1's (es decir,  $X+(-X)=0$  en complemento a 1).
  - En un punto intermedio, los routers
    - comprueban el CRC en cada salto y tiran el paquete si es erróneo.
    - modifican siempre el TTL por lo que necesitan recalcular el CRC. Para ello usan una manera rápida de calcular el CRC de manera incremental:
      - Localizar la palabra de 16bits modificada (B) y restarla de la original (A), el valor resultante se sumará al CRC original para obtener el nuevo CRC =  $-X+A-B$ .

## Campos de la cabecera IPv4

- Ejemplo de cálculo del CRC:

4	5	0	28
1		0	0
4	17	0	
10.12.14.5			
12.6.7.9			

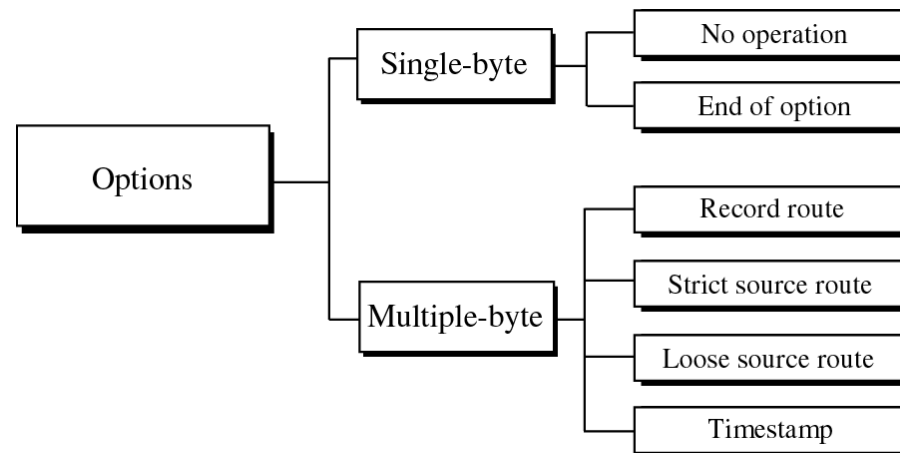
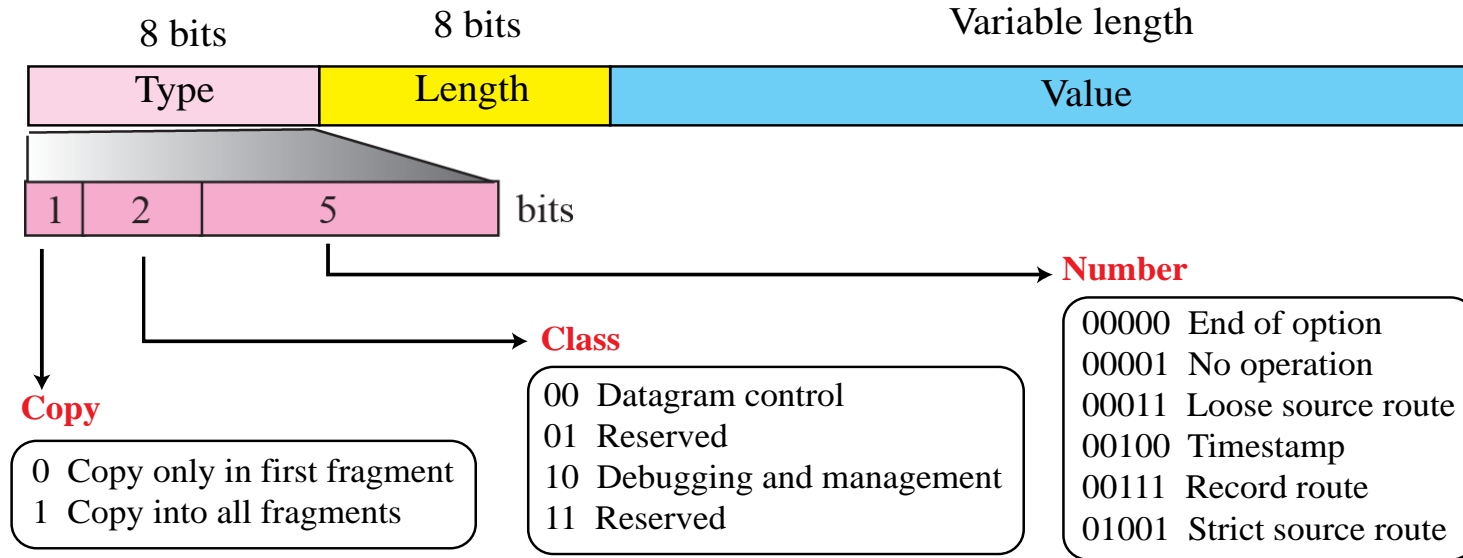
4, 5, and 0	→	01000101	00000000
28	→	00000000	00011100
1	→	00000000	00000001
0 and 0	→	00000000	00000000
4 and 17	→	00000100	00010001
0	→	00000000	00000000
10.12	→	00001010	00001100
14.5	→	00001110	00000101
12.6	→	00001100	00000110
7.9	→	00000111	00001001
Sum	→	01110100	01001110
Checksum	→	10001011	10110001

## Campos de la cabecera IPv4

- Source/Destination IP address (32bits cada uno)
  - Direcciones IP origen y destino, identifican unívocamente al origen y destino del paquete.
  - Permanecen intactas por todo el recorrido del paquete en la red.
  - Típicamente se representan como 4 números decimales 0-255 separados por puntos que hacen referencia a cada byte. Ej:  
130.206.160.215
- Options (0-40bytes)
  - Proveen funcionalidades de testeo de la red.
  - Se organizan en una estructura *Code+Length+Data*. Los datos son de longitud variable.
  - Siempre se realiza padding (relleno) de las opciones a múltiplos de 32 bits para que pueda representarse su tamaño en palabras de 4 bytes que necesita el campo HLEN.



## 2.1 Opciones en cabecera IPv4

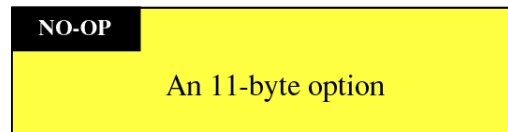


## Opción: No operación

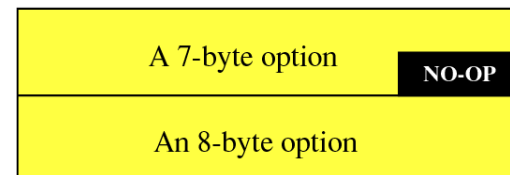
- Se usa como relleno entre opciones para alinear la siguiente opción a fronteras de 16 o 32 bits (tamaño de palabras de procesador típico)

Code: 1  
 00000001

a. No operation option



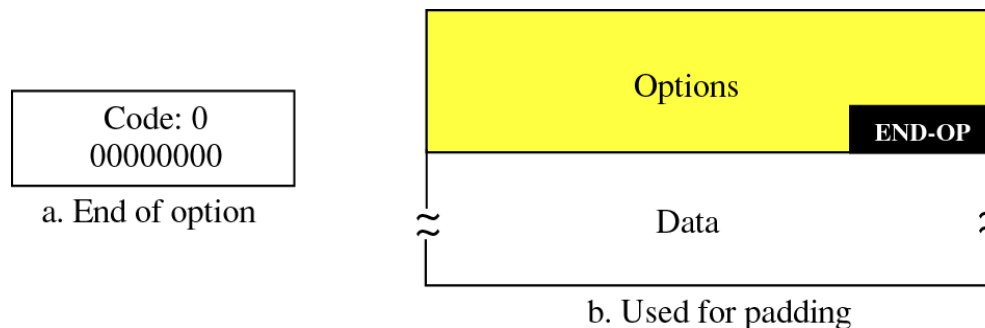
b. Used to align beginning of an option



c. Used to align the next option

## Opción: Fin de opciones

- Se usa como relleno al final de la parte de opciones.
- Si existe, sólo se puede usar como última opción una sola vez, ya que lo que viene a continuación se suponen datos.
- Si se necesitan varios bytes de relleno al final de las opciones, tendrán que usarse varios de NO\_OP y sólo uno final de END-OP.



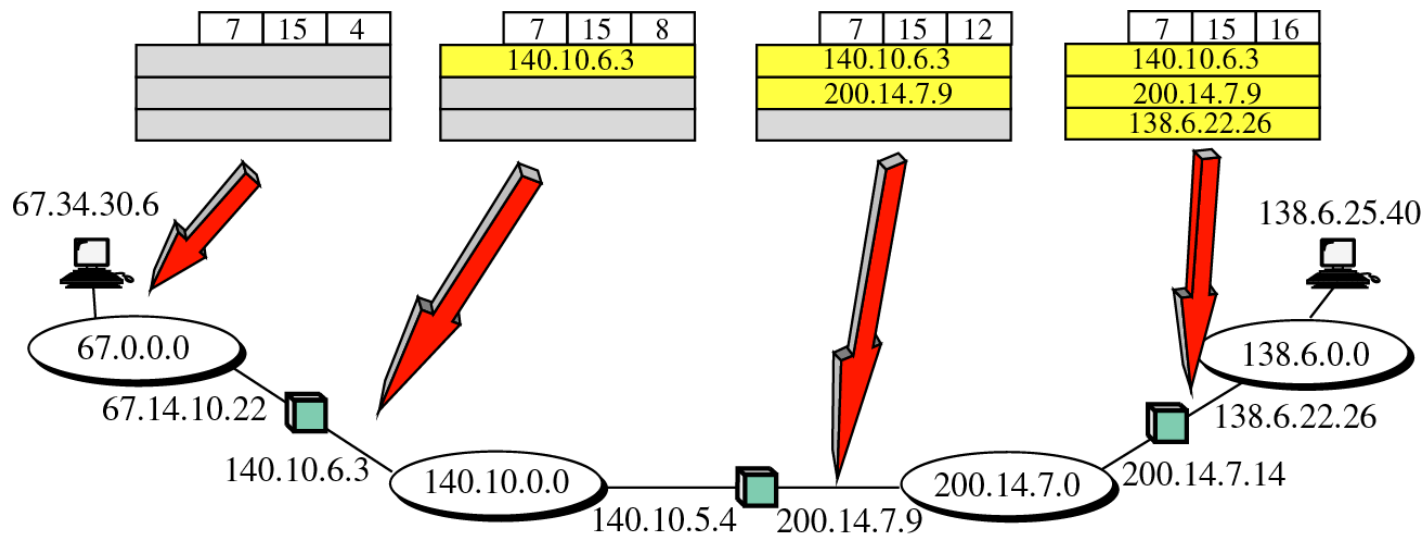
## Opción: Registro de ruta

- Permite que hasta 9 routers intermedios almacenen su dirección IP en la opción para llegar al destino (limitado por los 40 bytes de opciones).
- El campo *pointer* es un número entero que contiene la posición del primer byte libre con respecto a la primer byte de la opción.
- Se almacena la dirección IP del interfaz por el que sale el paquete.

Code: 7 00000111	Length (Total length)	Pointer
First IP address (Empty when started)		
Second IP address (Empty when started)		
• • •		
Last IP address (Empty when started)		

# Opción: Registro de ruta

- Ejemplo de opción registro de ruta (record route)



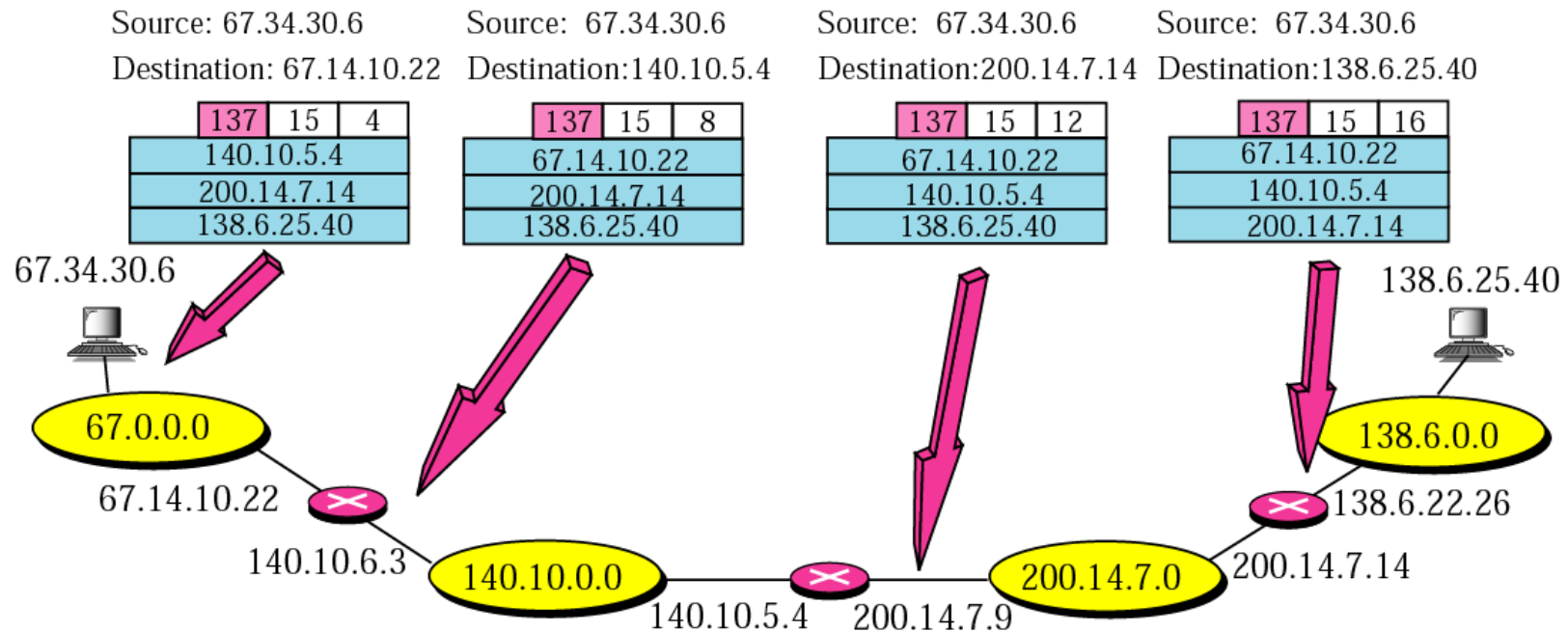
## Opción: Enrutamiento fuente estricto

- El origen especifica el camino que ha de seguir el paquete con las IPs de los routers a atravesar. Máximo 9 saltos.
- Si no se puede seguir el camino fijado se tira el paquete.
- El primer salto (router) se coloca como IP destino del paquete
- El segundo salto está en pointer=4.
- Cuando el paquete llega a un router, comprueba que el puntero sea menor que la longitud y reemplaza la IP destino del paquete con la apuntada por el pointer y reemplaza la dirección apuntada por el pointer con la dirección IP de su interfaz de entrada.

Code: 137 10001001	Length (Total length)	Pointer
First IP address (Filled when started)		
Second IP address (Filled when started)		
• • •		
Last IP address (Filled when started)		

## Opción: Enrutamiento fuente estricto

- Ejemplo de opción de enrutamiento fuente estricto (strict-source-route)



## Opción: Enrutamiento fuente débil

- Fija algunas direcciones IPs de routers a visitar, pero se pueden atravesar otros routers también.
  - Si la dirección apuntada por el pointer se puede usar como siguiente salto se usa. En caso contrario se usa como siguiente salto la que marque la tabla de rutas para llegar a esa dirección.
- Se sigue el mismo procedimiento de reemplazo de direcciones que con el enrutamiento fuente estricto.

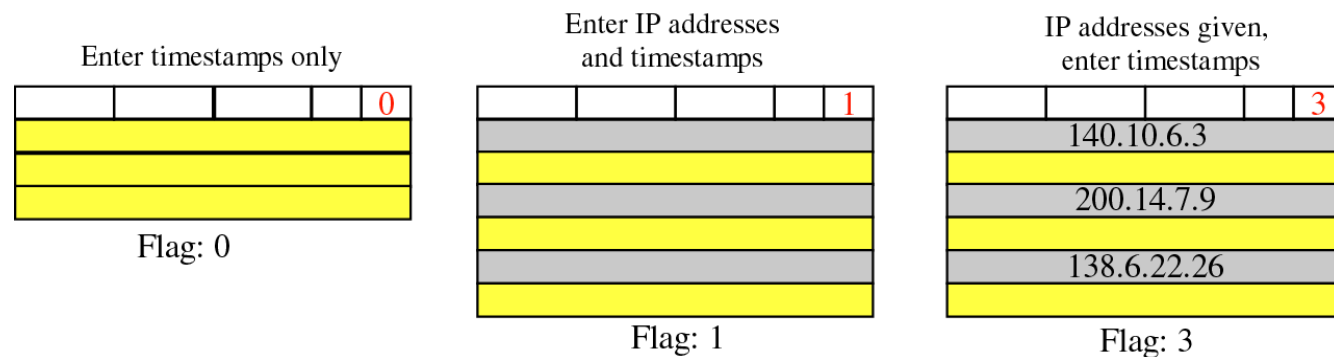
Code: 131 10000011	Length (Total length)	Pointer
First IP address (Filled when started)		
Second IP address (Filled when started)		
• • •		
Last IP address (Filled when started)		



# Opción: Timestamp

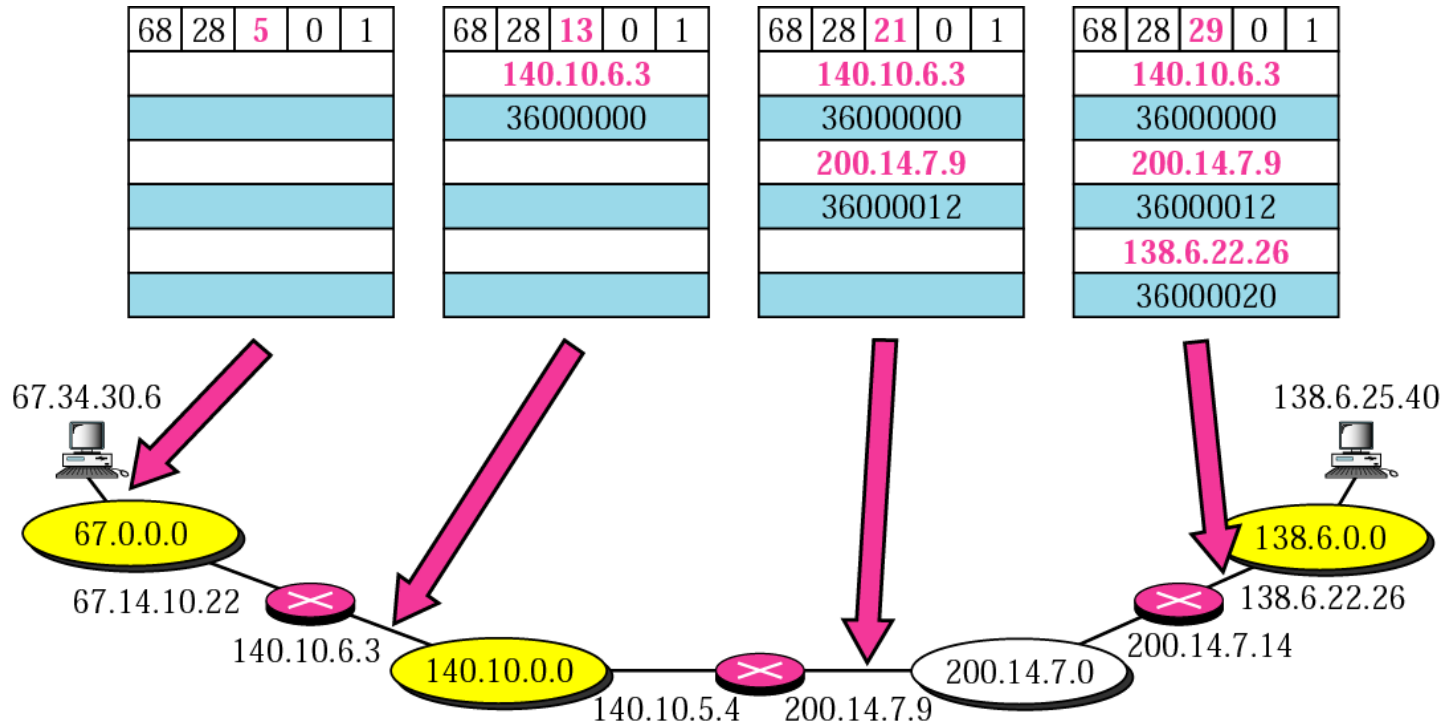
- Almacena los timestamps de cada router junto con sus direcciones IP.
- Los timestamps son en milisegundos desde medianoche UTC.
- Campo overflow: lo incrementa cada router que por falta de espacio no ha podido meter su información.
- 3 posibilidades según los flags:

Code: 68 01000100	Length (Total length)	Pointer	O-Flow 4 bits	Flags 4 bits
First IP address				
Second IP address				
⋮				
Last IP address				



# Opción: Timestamp

- Ejemplo de opción timestamp



# Resumen

- Protocolo IP
  - Nivel de red, servicio de datagramas no fiable.
  - Provee:
    - Direccionamiento.
    - Encaminamiento.
    - Fragmentación y reensamblado.
- Cabecera IP
  - Multiplexación de protocolos de nivel de transporte.
  - Incluye campos específicos para el tratamiento de la fragmentación.
  - TTL, limitación en el tiempo de vida.
  - Campos opcionales para enrutamiento fuente y registro de ruta.

## Referencias

- [Forouzan]
  - Capítulo 7, secciones 7.1-7.2 “Introduction”, “Datagrams”, secciones 7.4-7.5 “Options”, “Checksum”
- [Stevens]
  - Capítulo 3, secciones 3.1-3.2 “Introduction”, “IP header”