

Redes

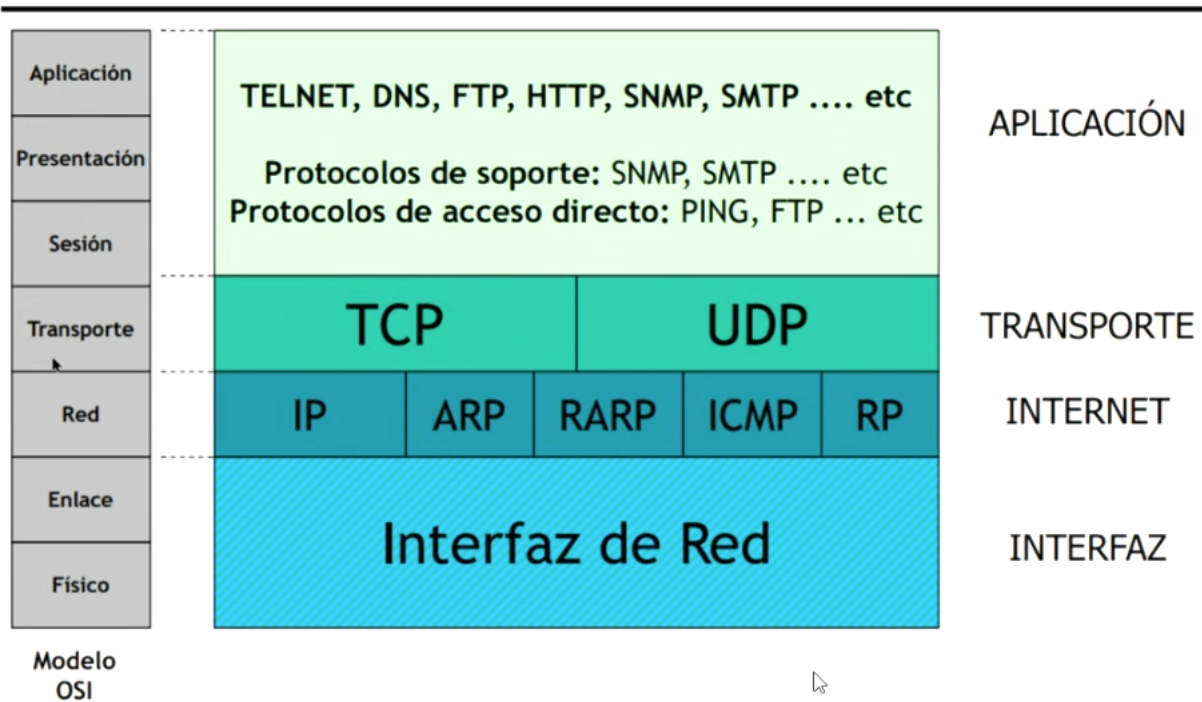
Análisis de tramas

UTN FRBA

2do cuatrimestre 2022

Capas y protocolos

Arquitectura de TCP/IP



Fuente: Presentación de Modelo OSI de Koval, citada por Souza en la última clase de laboratorio.

Encapsulamiento

Esto es una casuística armada en base a las tramas de ejemplo que dieron los profesores junto con los TL 5 y 6.

Protocolos encapsulados						Col der.	Archivo
Ethernet II	802.1Q	IPv4	TCP	(?)			VLAN
Ethernet II	802.1Q	-	UDP	Netbios			VLAN
Ethernet II	802.1Q	LLC	Netbios				VLAN #44
Ethernet II	802.1Q	IPv4	ICMP				VLAN
Ethernet II	802.1Q	LLC	ARP				VLAN
Ethernet II	802.1Q	ARP					VLAN
802.11 (32bytes)	Data						802_11
802.11 (24)	Wireless managem ent 802.11 (beacon)						802_11
Ethernet II	ARP						ARP
802.3	LLC						Eth LLC
Ethernet II	ARP						Eth LLC
Ethernet II	IPv4	ICMP					ICMP 2000
Ethernet II	IPv4	TCP	HTTP			Se ve el request	ICMP 2000
802.3	LLC	Spanning tree (STP)					
Ethernet II	IPv4	UDP	DHCP			-	DHCP
Ethernet II	IPv4	UDP	DNS			La web solicitada	DNS

Ethernet II	IPv4	TCP	FTP			Texto legible	FTP
Ethernet II	IPv4	TCP	HTTP			Generalmente hay texto, pero no necesariamente en la 1a parte	HTTP

Campos por protocolo

Ethernet / 802.3 (c2) <ul style="list-style-type: none"> • MAC destino (6 bytes) • MAC origen (6 bytes) • Ethertype/Length (2 bytes) • [Datos] • Al final: padding (bytes necesarios para alcanzar 46 bytes mínimos de datos) 	Ethernet + 802.1Q (c2) <ul style="list-style-type: none"> • MAC destino (6 bytes) • MAC origen (6 bytes) • Ethertype/Length (2 bytes): 0x8100 • PRI (3 bits) + Token Encapsulation Flag (1 bit) • VLAN ID (12 bits) • Type/Len (2 bytes) • [Datos] • Al final: padding (bytes necesarios para alcanzar 46 bytes mínimos de datos)
IP (c3) <ul style="list-style-type: none"> • Versión (4 bits): casi siempre 4 • HLEN (4 bits, representa cantidad de palabras de 4 bytes): casi siempre 5 • ToS (1 byte) • Long total (2 bytes) - en bytes • ID (2 bytes) • Flags + Offset (en total 2 bytes) <ul style="list-style-type: none"> ◦ Flat reservado (1 bit) ◦ Flag DF (1 bit): No fragmentar ◦ Flag MF (1 bit): Más fragmentos ◦ Offset (5 bits): >0 si estamos en un fragmento que no es el 1 • TTL (1 byte) • Protocolo (1 byte) • CRC (2 bytes) • IP origen (4 bytes) • IP destino (4 bytes) • Opciones y relleno (en total, múltiplos de 4 bytes - sólo si HLEN es > 5) • [Datos] 	TCP (c4) <ul style="list-style-type: none"> • Source port (2 bytes) • Destination port (2 bytes) • Sequence number (4 bytes) • Acknowledgement number (4 bytes) • HLEN (4 bits) (se multiplica x 4 bytes) • Reserved (6 bits) + Flags (6 bits) <ul style="list-style-type: none"> ◦ Flag URG ◦ Flag ACK ◦ Flag PSH ◦ Flag RST ◦ Flag SYN ◦ Flag FIN • Window (2 bytes) • Checksum (2 bytes) • Urgent pointer (2 bytes)
(R)ARP (capa 3? Quizás 2) <ul style="list-style-type: none"> • Hardware type (2 bytes) • Protocolo (2 bytes) 	ICMP <ul style="list-style-type: none"> • Tipo (1 byte) • Código (1 byte) • ICMP Checksum (2 bytes)

<ul style="list-style-type: none"> • Long dirección física en bytes (1 byte) • Long dirección lógica en bytes (1 byte) • Operación (2 bytes) • Dirección física del emisor - MAC (6 bytes) • Dirección lógica del emisor - IP (4 bytes) • Dirección física del destino - MAC (6 bytes) • Dirección lógica del destino - IP (4 bytes) 	<ul style="list-style-type: none"> • Datos (0 a 68 bytes)
UDP	UDP
Ethernet + LLC	HTTPS

General tips

- Una respuesta a otra trama va a tener dirección destino y origen invertidos, en cada uno de los protocolos (ej: Ethernet > IP > UDP)

Ethernet / 802.3 (capa 2)

- Empieza con 2 MAC address (6 pares de letras cada una)
 - Dirección destino.
 - Broadcast: todo F
 - Multicast: ??
 - Unicast
- Type/LEN (2 bytes)
 - LEN: Si el valor es menor a 0x600 (1536 decimal), este campo es LENGTH.
 - Ethertype: Si el valor es mayor a 0x600 (1536 decimal), es un campo type. [Lista completa de tipos en Wikipedia.](#)
 - 0x0800 → [IPv4](#)
 - 0x0806 → [ARP](#)

- 0x8035 → [RARP](#) (NO es respuesta a ARP)
- 0x86DD → IPv6
- 0x8100 → [VLAN TAGGING](#) (ojo, cambia el formato del resto del paquete!

El tag tiene 4 bytes, y después lo sigue el Ethertype de 2 bytes)

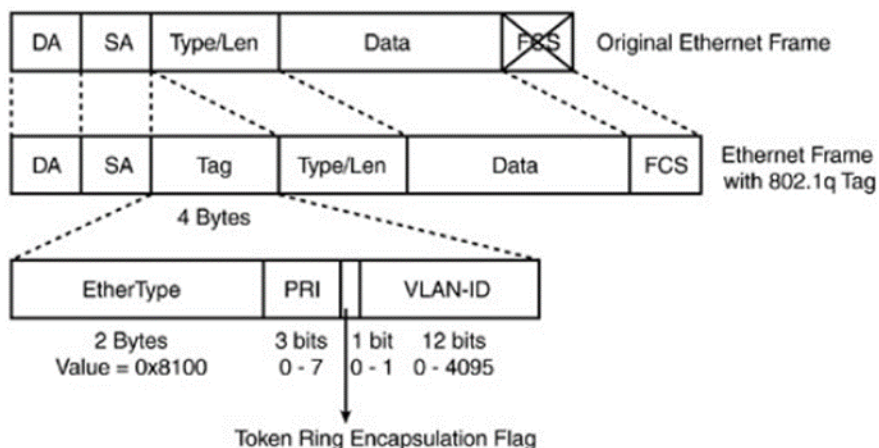
- **Longitud de la trama total:** normalmente falta el CRC en la trama compartida, que es de 4 bytes. Se suman esos 4 bytes a los caracteres visibles. Hay que confirmar que el contenido del datagrama esté completo en la captura (por ej, en [IP](#)).

Trama#11

0000	c4 ea 1d 66 1a d4	e4 f8 9c b4 6c 39 08 00 45 00	...f.... ..l9..E.
0010	00 3e 68 84 00 00 80 11	4e ca c0 a8 01 0f c0 a8	.>h..... N.....
0020	01 01 db 77 00 35 00 2a	95 53 02 1f 01 00 00 01	...w.S.* .S.....
0030	00 00 00 00 00 00 03 77	77 77 08 6d 73 66 74 6ew ww.msftn
0040	63 73 69 03 63 6f 6d 00	00 01 00 01	csi.com.

Según Koval, la MAC destino c4 ea 1d 66 1a d4 es unicast porque termina en 4, que equivale a 0100. Si fuera multicast, no valdría 0.

VLAN tagging (802.1Q)



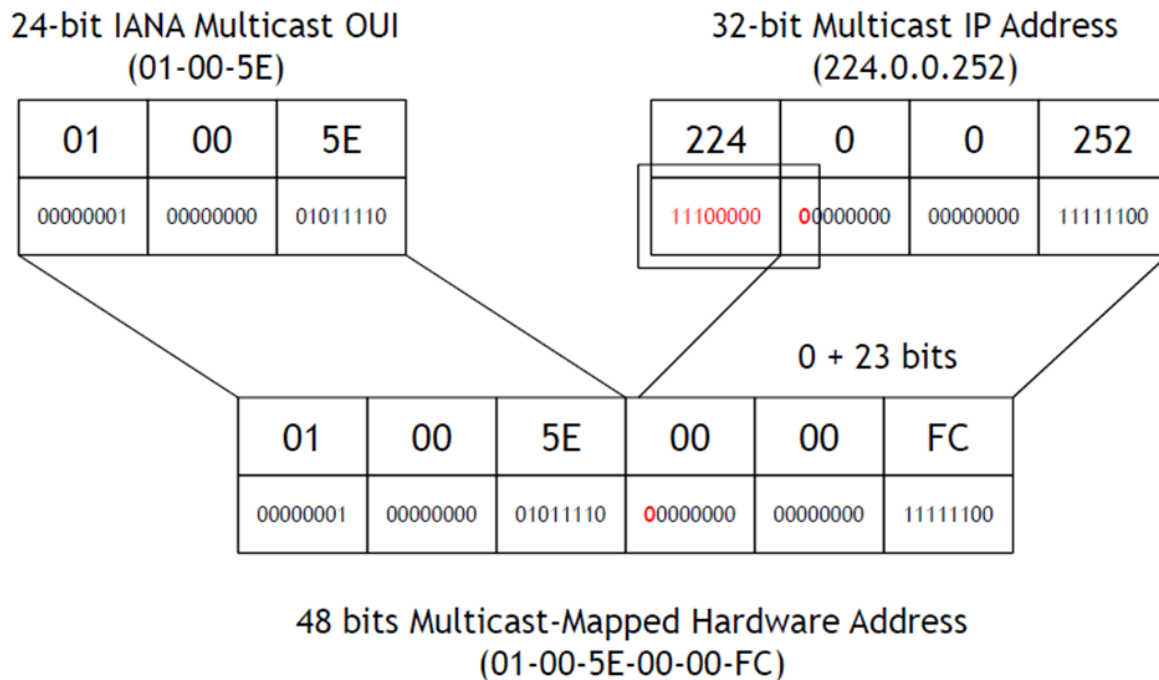
* PRI (3 bits) + Token Encapsulation Flag (1 bit)

* VLAN ID (12 bits)

* Type/Len (2 bytes)

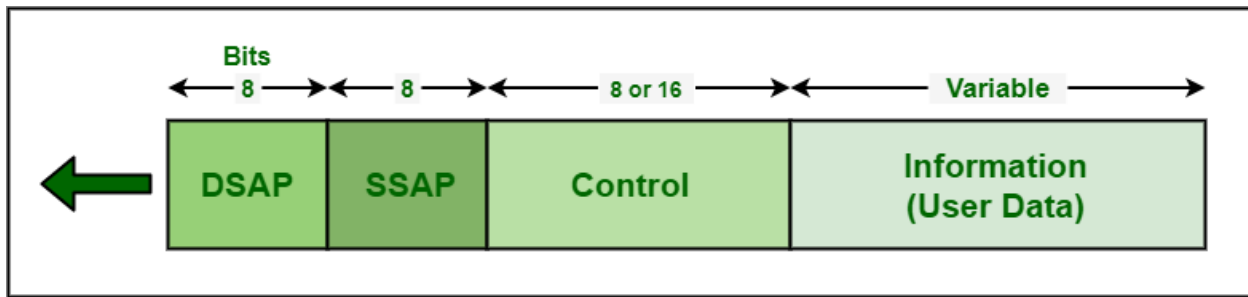
Ejemplo de MAC multicast

Según Souza, todas las multicast empiezan con 01:00:5e y siguiente bit 0. Internet menciona más opciones.



LLC / 802.2 (capa 2)

The LLC sublayer provides [multiplexing](#) mechanisms that make it possible for several network protocols (e.g. [IP](#), [IPX](#) and [DECnet](#)) to coexist within a multipoint network and to be transported over the same network medium. It can also provide [flow control](#) and [automatic repeat request \(ARQ\)](#) error management mechanisms. (Wikipedia)



PDU Format

LLC header [\[edit \]](#)

Any 802.2 LLC PDU has the following format:

802.2 LLC Header			Information
DSAP address	SSAP address	Control	
8 bits	8 bits	8 or 16 bits	multiple of 8 bits

When **Subnetwork Access Protocol (SNAP)** extension is used, it is located at the start of the Information field:

802.2 LLC Header			SNAP extension		Upper layer data
DSAP	SSAP	Control	OUI	Protocol ID	
8 bits	8 bits	8 or 16 bits	24 bits	16 bits	multiple of 8 bits

The 802.2 header includes two eight-bit address fields, called **service access points** (SAP) or collectively LSAP in the OSI terminology:

- SSAP (Source SAP) is an 8-bit long field that represents the logical address of the network layer entity that has created the message.
- DSAP (Destination SAP) is an 8-bit long field that represents the logical addresses of the network layer entity intended to receive the message.

Campos ([fuente](#)):

- DSAP: (1 byte) logical addresses of the network layer entity meant to receive the message. El último bit indica si se trata de una dirección individual (0) o grupal (1).
- SSAP: (1 byte) representa la dirección lógica de la entidad que creó el mensaje. El último bit indica si es comando (0) o respuesta (1)
- Control: (1 o 2 bytes)
 - Según el tipo de mensaje LLC, este campo varía
 - Trama de información: número de secuencia (N(S)) y número de confirmación (N(R)) (para piggyback)
 - De supervisión: número de confirmación (N(R)) y código de supervisión (Receiver Ready, Receiver Not Ready, REJection, Selective REJection)
 - No numerada: indica el tipo de PDU, se usa para control y establecimiento de conexión.
- Datos: (variable)

Ejemplo: Eth2+VLAN+LLC+ARP

> Frame 78: 64 bytes on wire (512 bits), 64 bytes capture	0000	ff ff ff ff ff ff 00 05 02 71 fc db 81 00 00 14
> Ethernet II, Src: ApplePci_71:fc:db (00:05:02:71:fc:db)	0010	00 24 aa aa 03 00 00 00 08 06 00 01 08 00 06 04
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 20	0020	00 01 00 05 02 71 fc db 83 97 14 48 ff ff ff ff
> Logical-Link Control	0030	ff ff 83 97 14 fe 55 55 55 55 55 55 55 55 55
> DSAP: SNAP (0xaa)		
> SSAP: SNAP (0xaa)		
> Control field: U, func=UI (0x03)		
> Organization Code: 00:00:00 (Officially Xerox, but Type: ARP (0x0806))		
> Address Resolution Protocol (request)		

Para notar:

- Inmediatamente antes de LLC: Ethertype/Length en VLAN TAG: 0x0024 = es un length!
- Campos LLC:
 - DSAP, SSAP, Control (comunes siempre en LLC?)
 - Organization code
 - Type: 0806 (mismo Ethertype que en Ethernet)

Ejemplo: Eth2 + VLAN + LLC + Netbios (?)

- Inmediatamente antes de LLC: Ethertype/Length en VLAN TAG: 0x00a6 = es un length!
- Campos LLC:
 - DSAP, SSAP, Control
 - NADA MAS.

> Frame 44: 184 bytes on wire (1472 bits), 184 bytes capture	0000	03 00 00 00 00 01 00 20 18 62 73 a1 81 00 00 05
> Ethernet II, Src: CisTechn_62:73:a1 (00:20:18:62:73:a1)	0010	00 a6 f0 f0 03 2c 00 ff ef 08 00 00 00 00 00
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 5	0020	00 45 52 4c 20 20 20 20 20 20 20 20 20 20 20
> Logical-Link Control	0030	1d 41 4e 41 4e 4e 49 20 20 20 20 20 20 20 20
> DSAP: NetBIOS (0xf0)	0040	20 ff 53 4d 42 25 00 00 00 00 00 00 00 00 00
> SSAP: NetBIOS (0xf0)	0050	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
> Control field: U, func=UI (0x03)	0060	00 11 00 00 21 00 00 00 00 00 00 00 00 e8 03
> NetBIOS	0070	00 00 00 00 00 00 00 00 21 00 56 00 03 00 01 00
> SMB (Server Message Block Protocol)	0080	00 00 02 00 32 00 5c 4d 41 49 4c 53 4c 4f 54 5c
> SMB MailSlot Protocol	0090	42 52 4f 57 53 45 00 01 00 80 fc 0a 00 41 4e 41
> Microsoft Windows Browser Protocol	00a0	4e 4e 49 00 00 00 00 00 00 00 00 00 04 00 03
	00b0	10 03 00 0f 01 55 aa 00

IP (capa 3)

Encapsulado en: Ethernet

Tamaño:

- Total: máximo: 2^{16} bytes (porque el campo longitud total tiene 16 bits).
- Cabecera: Mínimo: 20 bytes. Máximo: 15 bloques de 4 bytes = 60 bytes. Cantidad de palabras (1 pal = 4 bytes) se indica en el campo HLEN.

0	4	8	16	19	24	31
Vers.	HLEN	ToS	Longitud Total			
Identificación			FLAGS	Desplazamiento del Fragmento		
TTL	Protocolo		Checksum del encabezado			
Dirección IP Fuente						
Dirección IP Destino						
Opciones (si las hay)					Relleno	
DATOS						
...						

- Empieza con un par de letras que suele ser 45
 - 4: versión de IP
 - 5: cantidad de palabras que tiene la cabecera IP (1 palabra = 8 dígitos, o sea 4 bytes).
 - Fragmentación se ve en:
 - Flags: 3 bits.
 - 1: No se usa
 - 2: No fragmentar
 - 3: Más fragmentos

- Desplazamiento de fragmento. Si está en 0, o no está fragmentado (flag 3 = 0), o es el 1er fragmento (el flag 3 = 1).
- Las palabras 4 y 5 son IP origen e IP destino (en ese orden, que es al revés que Ethernet)
- Protocolos posibles:
 - 0x01 (1): ICMP
 - 0x06 (6): TCP
 - 0x11 (17): UDP
 - 0x32 (50): ESP

Fragmentación

	Flags			Desplazamiento
	Sin uso	DF: no fragmentar	MF: Más fragmentos	
General				
1er fragmento	0	0	1	0
Penúltimo	0	0	1	No 0
Último	0	0	0	No 0

Cómo saber si el datagrama está completo en la captura de wireshark?

1. La longitud total (2a mitad de la primer palabra) me da la longitud del datagrama en bytes.
2. Cuento los bytes desde el inicio del datagrama IP (normalmente, desde el 45) y confirmo que concuerden.

ARP y RARP (capa 3)

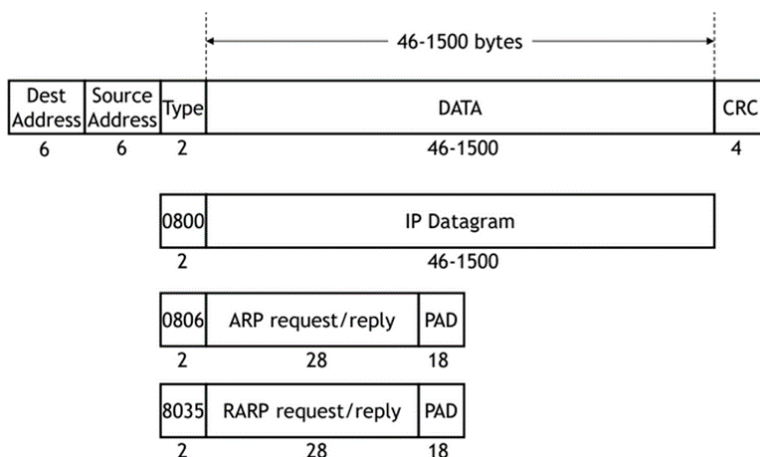
Generalidades

Nivel OSI: 3 según Koval (2 según internet)

Qué hace? Encontrar la dirección de hardware (Ethernet MAC) que corresponde a una determinada dirección IP.

Encapsulado en: Ethernet

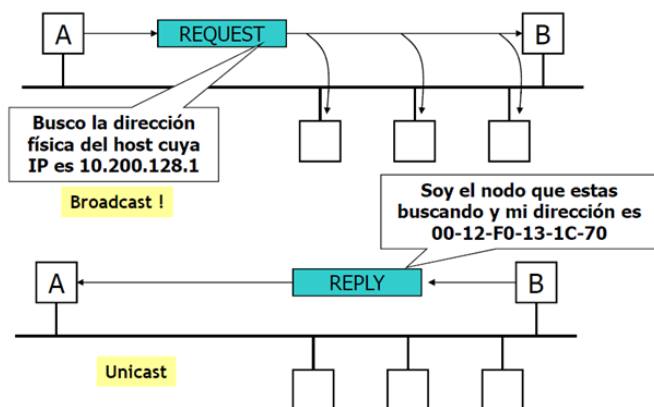
Tamaño: 28 bytes fijos. Dentro de una trama Ethernet, requiere un padding de 18 bytes para alcanzar el tamaño mínimo de PDU.



Cómo funciona? Se envían 2 PDUs.

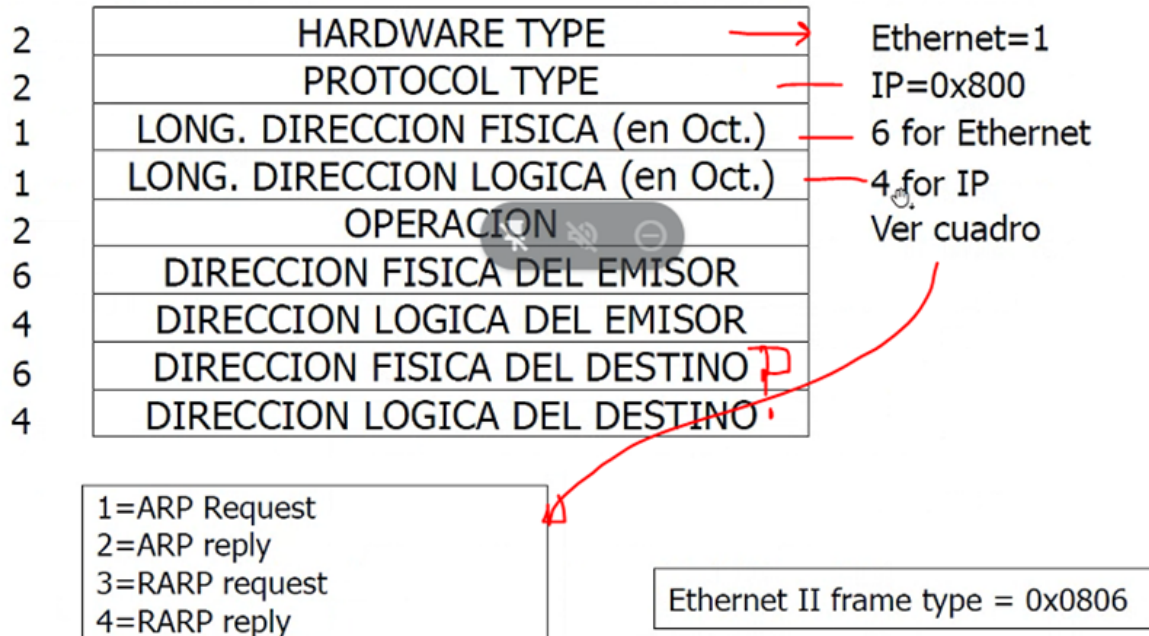
- A envía un request ARP. Envía un broadcast preguntando por el dueño de una IP determinada.
- B, dueño de esa IP, envía una reply ARP, indicando su MAC, que es la correspondiente a la IP.

A va a guardar la relación MAC-IP en un cache ARP.



Formato de trama

Cant. Octetos

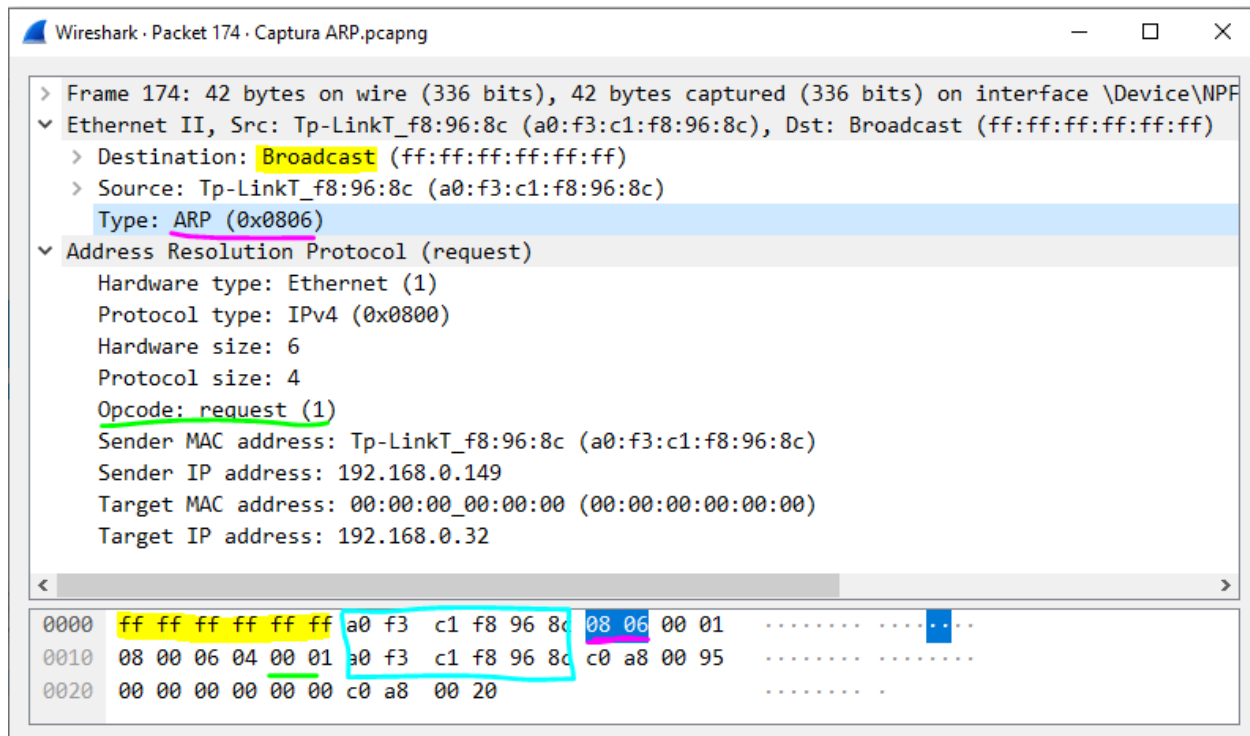


Tips para reconocer tramas ARP y RARP

REQUEST.

- En el header Ethernet
 - Es un broadcast, por lo que empieza con FF FF FF FF FF FF. (MAC de broadcast)
 - Después viene la MAC de origen (6 bytes, o sea, 6 pares de letras)
 - Después viene el type:
 - ARP, que es 0x0806
 - RARP: 0x8035
- En el header ARP
 - Opcode indica que es request o reply, ARP o RARP
 - La MAC de origen de la cabecera Ethernet, coincide con la MAC origen que aparece después del opcode.

- Los últimos 8 dígitos son la dirección IP por la que se está preguntando, precedidos por 12 ceros (MAC desconocida de 6 bytes)



REPLY:

- Encabezado Ethernet
 - MAC destino y MAC de origen (12 pares de dígitos),
 - Type: ARP (0806)
- Encabezado ARP
 - Campos varios.
 - Op code: 00 02 (reply)
 - MAC e IP de origen
 - MAC e IP de destino

Hay coincidencia entre MACs en encabezado Ethernet y ARP.

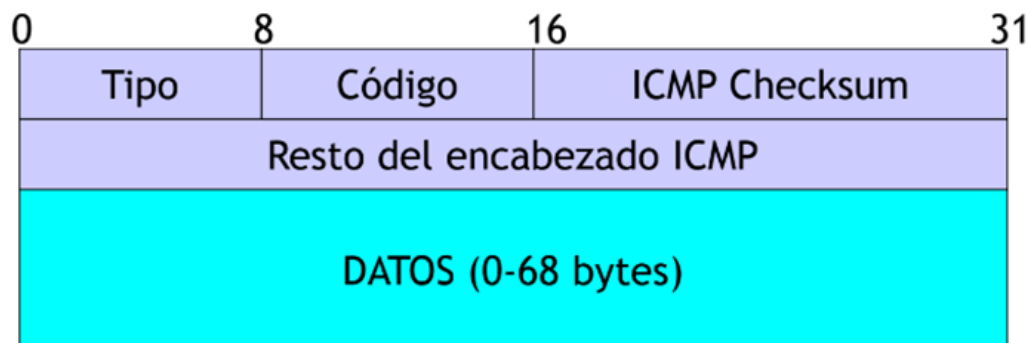
```
> Frame 254: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF
▼ Ethernet II, Src: c2:71:dc:db:f1:d0 (c2:71:dc:db:f1:d0), Dst: Tp-LinkT_f8:96:8c (a0:f3:c1:f8:
  > Destination: Tp-LinkT_f8:96:8c (a0:f3:c1:f8:96:8c)
  > Source: c2:71:dc:db:f1:d0 (c2:71:dc:db:f1:d0)
  Type: ARP (0x0806)
  Trailer: ef5ef15a5010020125e800002b700b64d1c3
▼ Address Resolution Protocol (reply)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: reply (2)
  Sender MAC address: c2:71:dc:db:f1:d0 (c2:71:dc:db:f1:d0)
  Sender IP address: 192.168.0.32
  Target MAC address: Tp-LinkT_f8:96:8c (a0:f3:c1:f8:96:8c)
  Target IP address: 192.168.0.149
```

0000	a0 f3 c1 f8 96 8c c2 71 dc db f1 d0 08 06 00 01q
0010	08 00 06 04 00 02 c2 71 dc db f1 d0 c0 a8 00 20q
0020	a0 f3 c1 f8 96 8c c0 a8 00 95 ef 5e f1 5a 50 10^..ZP..
0030	02 01 25 e8 00 00 2b 70 0b 64 d1 c3	..%...+p .d..

ICMP

Modelo OSI: va encapsulado en IP, pero Koval lo considera nivel 3 también.

Datagrama ICMP



- Tipo (1 byte): tipo de mensaje de error que se está generando. [Valores posibles](#).
- Código (1 byte): Código del mensaje de error. [Valores posibles](#).
- ICMP Checksum (2 bytes): Cubre todo el datagrama, garantiza la exactitud del mensaje.
- Datos (0 a 68 bytes): Generalmente contiene
 - Encabezado IP del datagrama que causó el error.
 - Primeros 8 bytes de datos de este datagrama.
 - Información necesaria para identificar la raíz del error.

Por ejemplo, un mensaje ICMP del tipo 3 indica que no se ha alcanzado el objetivo del paquete de datos, mientras que el código de este dato precisa y ofrece información acerca de si la red de destino (0), el host deseado (1) o el puerto esperado (3) no ha respondido a la solicitud.

Echo Request / Reply

Caso particular de ICMP. Difiere del general en la segunda línea.

0	8	16	31
Tipo		Código	
Identificador		ICMP Checksum	
		Número de secuencia	

Permite conocer si la interfaz destino es alcanzable y está funcionando.

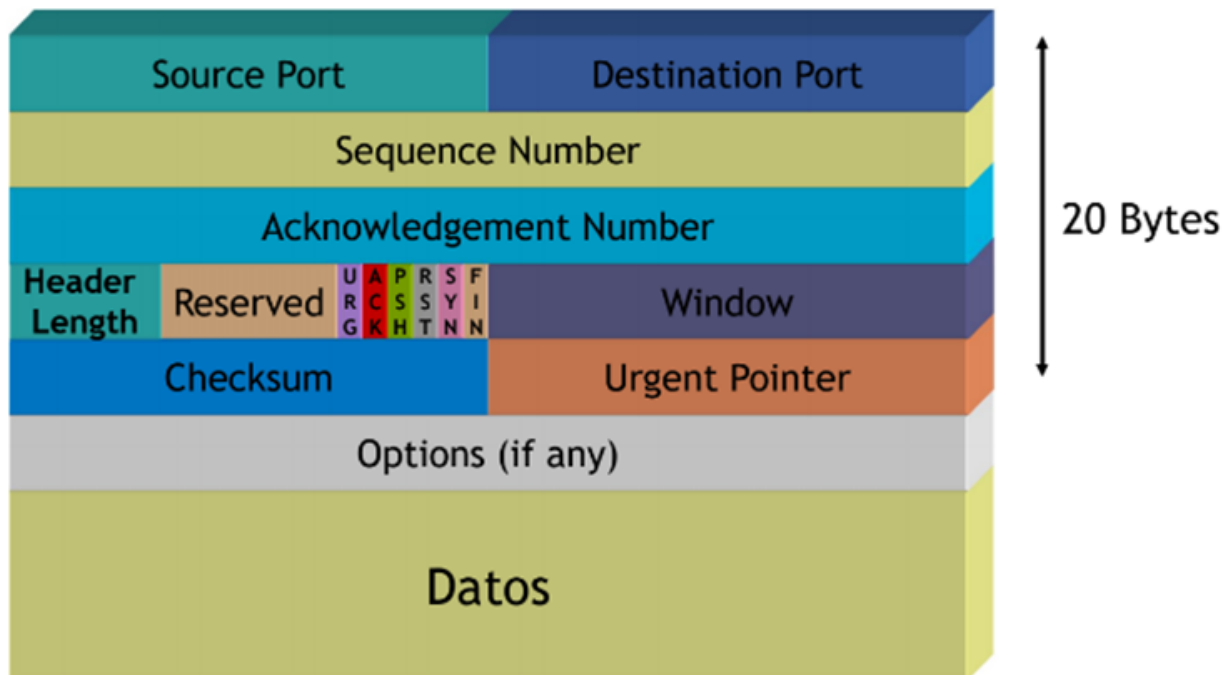
- **Echo request:** Envía un identificador y un número de secuencia para contrastar request y replies.
- **Echo reply:** La respuesta no es obligatoria. Debe responder incluyendo los datos recibidos en el request.

Qué usa estos requests:

- PING: Si se envía un **ping utiliza un echo request y echo reply**.
- Traceroute: me permite identificar si el ruteo es correcto (ver no llegó, o si llego y no me respondió, etc). Se suele utilizar luego de un ping fallido.

TCP

Trama



Flags

- **URG:** El contenido de Urgent pointer es válido y relevante. Existen datos urgentes, y el urgent pointer apunta al último byte de datos urgentes.
- **ACK:** El contenido del campo ACK es válido y relevante. Estoy confirmando la recepción de otro mensaje.
- **PSH:** Push. Procesar tan pronto como pueda. Obliga al receptor a enviar la confirmación.
 - Sin esto, el receptor puede elegir si responder inmediatamente o esperar a tener que mandar otro mensaje y agregar la confirmación ahí (piggy-backing).

- Útil para aplicaciones que necesitan procesamiento inmediato del mensaje, como Telnet.
- 3 flags usados en establecimiento y fin de una conexión.
 - **RST:** Esta conexión debe reiniciarse. Sucede cuando:
 - No se acepta un pedido de conexión.
 - Uno de los dos extremos perdió el estado de la conexión y necesita reiniciar.
 - **SYN:** Sincronizar números de secuencia.
 - **FIN:** El emisor no tiene más datos para enviar. Se solicita cerrar la conexión (lo tiene que enviar el cliente y recibir ACK, y después el servidor y recibir ACK – 4 pasos de cierre).

Well known ports

Permiten identificar lo que hay dentro del mensaje a veces.

Port	Servicio
20	FTP-Data
53	DNS
21	FTP - Command
23	Telnet
80	HTTP
110	POP - Version 3
25	SMTP
1720	H.323

UDP

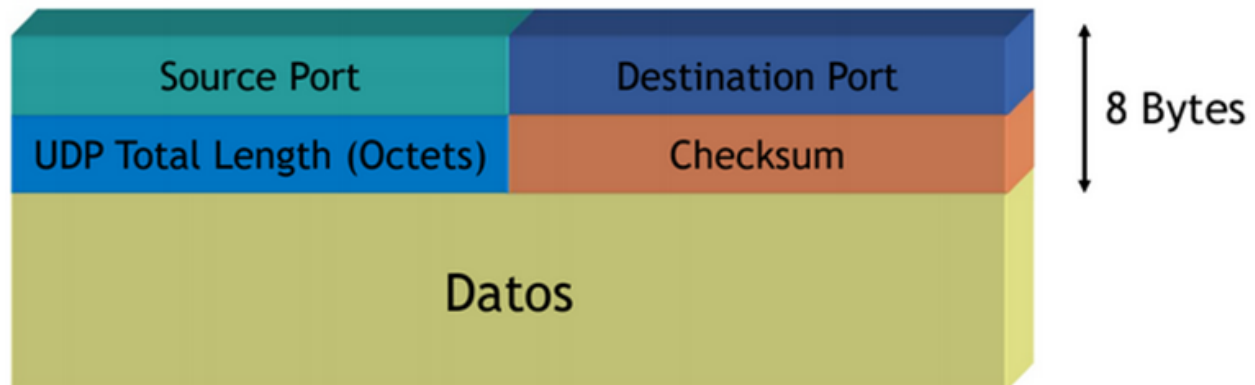
Modelo OSI: capa

Qué hace? Procesos simples de petición/respuesta. Multi- o broadcast.

Qué encapsula? DNS, Echo, TFTP, NTP, SNMP

Cómo reconocer dirección IP multicast? Van entre 224.0.0.0 y 239.255.255.255.

Trama



Well-known Ports

Permiten identificar lo que hay dentro del mensaje a veces.

Port	Servicio
7	ECHO
53	DNS
69	Trivial File Transfer Protocol (TFTP)
123	Network Time Protocol (NTP)
161	Simple Network Management Protocol (SNMP)

DHCP

Centraliza y administra la asignación de direcciones IP

Mantiene un registro de la IP asignada a cada cliente

El cliente inicializa una versión limitada de TCP/IP y envía un pedido de dirección IP a los servidores DHCP



DHCP discover tiene:

dirección origen 0.0.0.0, dirección destino 255.255.255.255.

Dirección física del cliente y el nombre del host.

DHCP Offer tiene:

- Dirección de hardware del cliente
- Dirección IP destino 0.0.0.0
- Una dirección IP ofrecida
- La máscara de subred
- Duración de la asignación (Lease)
- Una identificación del servidor (dirección IP)

FTP

DNS

Encapsulado en: Ethernet > IP > UDP

Cómo reconocer?

- Ethernet header - Type: IPv4 (0800, después de 12 octetos o 24 dígitos)
- Encapsulado en UDP. En IP header: protocol UDP (17 en decimal, o 0x11)
- DNS escucha peticiones en puerto 53. En el UDP header: destination port 53 (well-known port de DNS, en hexa se escribe 0x35!)
- **Cómo ver la query? Aparece textualmente en la columna de la derecha.**

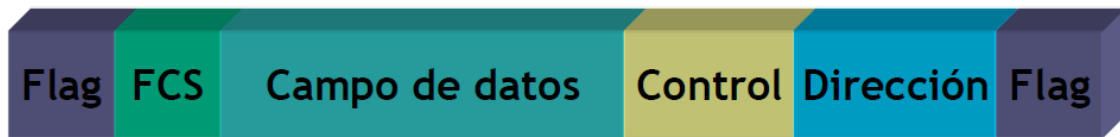
Wireshark · Packet 109 · Wi-Fi

- > Frame 109: 73 bytes on wire (584 bits), 73 bytes captured (584 bits) on interface \Device\N^
- ▼ Ethernet II, Src: Tp-LinkT_f8:96:8c (a0:f3:c1:f8:96:8c), Dst: Sagemcom_cf:61:ac (4c:19:5d:c
 - > Destination: Sagemcom_cf:61:ac (4c:19:5d:cf:61:ac)
 - > Source: Tp-LinkT_f8:96:8c (a0:f3:c1:f8:96:8c)
 - Type: IPv4 (0x0800)
- > Internet Protocol Version 4, Src: 192.168.0.149, Dst: 192.168.0.1
- ▼ User Datagram Protocol, Src Port: 50066, Dst Port: 53
 - Source Port: 50066
 - Destination Port: 53 DNS
 - Length: 39
 - Checksum: 0xd937 [unverified]
 - [Checksum Status: Unverified]
 - [Stream index: 5]
 - > [Timestamps]
 - UDP payload (31 bytes)
- ▼ Domain Name System (query)
 - Transaction ID: 0xb1da
 - > Flags: 0x0100 Standard query
 - Questions: 1
 - Answer RRs: 0
 - Authority RRs: 0
 - Additional RRs: 0
 - ▼ Queries
 - > accenture.com: type A, class IN

0000	4c 19 5d cf 61 ac a0 f3 c1 f8 96 8c 08 00 45 00	L.]·a·····E·
0010	00 3b f7 39 00 00 40 11 01 92 c0 a8 00 95 c0 a8	·;·9··@······
0020	00 01 c3 92 00 35 00 27 d9 37 b1 da 01 00 00 01	·····5·'·7·····
0030	00 00 00 00 00 00 09 61 63 63 65 6e 74 75 72 65	······a ccenture
0040	03 63 6f 6d 00 00 01 00 01	·com·····

Otras tramas

HDLC



→
Sentido de la transmisión

Características:

- Orientado a la conexión (hay establecimiento de conexión, intercambio de info, desconexión).
-

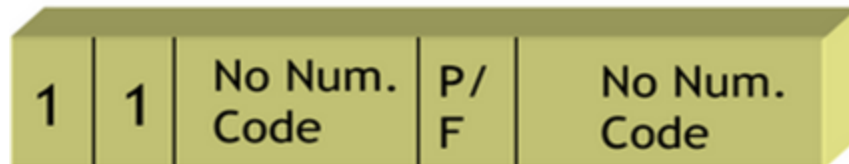
Campos:

- **Flag (1 byte).** Es la secuencia 01111110 (7E) que sirve para indicar el inicio y fin de la trama.
- **Campo de dirección (1 byte).**
 - En link punto-multipunto (config no equilibrada): Identifica a la estación secundaria que ha transmitido o va a recibir la trama.
 - Enlaces punto a punto. Se pone un valor fijo, no relevante, porque en un P2P tengo un único interlocutor posible.
- **Campo de control (1 o 2 bytes).** Sobre él se implementan todos los mecanismos de control de flujo y control de enlace.

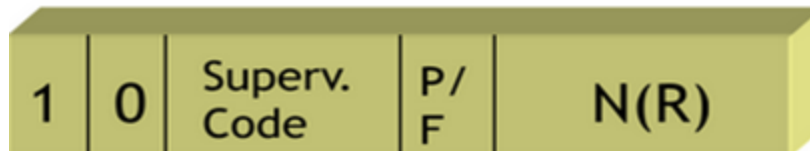


- Trama de info:

- Trama no numerada:



- Trama de supervisión:



- El primer/os bit indica que tipo de trama es.
 - Trama de intercambio de información: empieza con 0.
 - Trama de supervisión: empieza con 10. Activa mecanismos de control de flujo y errores.
 - Trama no numerada: empieza con 11. Se usan para la etapa de establecimiento y cierre de conexión.
- Si se extiende el campo de 8 bits a 16, los subcampos del campo de control que cambian son el N(S) y N(R),
- **Bit P/F (Poll / Final) (1 bit, con doble significado)**. P y F son representados con el mismo bit en 1, su significado depende de quién es el emisor.
- **Campo de datos (variable)**. El campo de datos es de longitud variable, transparente [CM1] e independiente del código.
- **Campo FCS (frame check sequence, 2 o 4 bytes)**. Código para la detección de errores calculado sobre los bits de la trama, excluidos los delimitadores. Existen 2 opciones: CRC-16 o CRC-32.

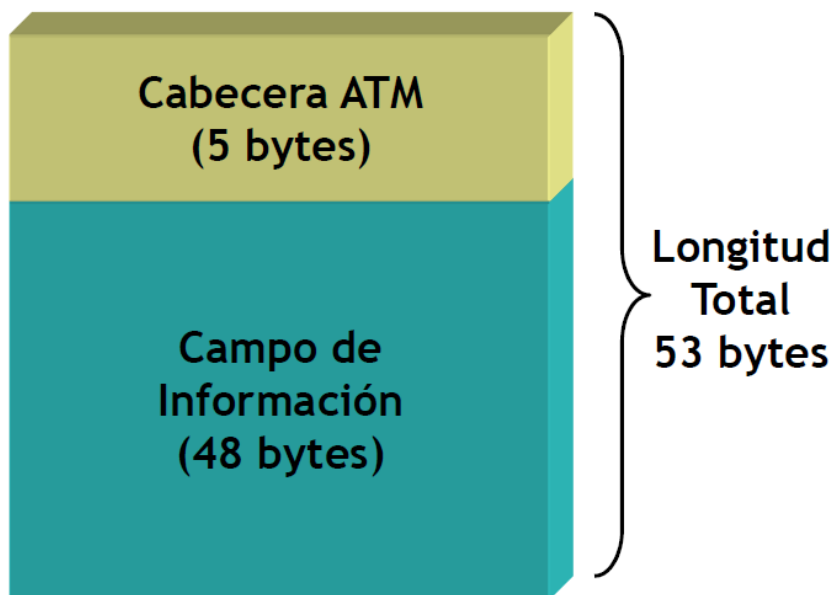
- Flag final (1 byte). El mismo que al inicio: 7E

ATM

Características

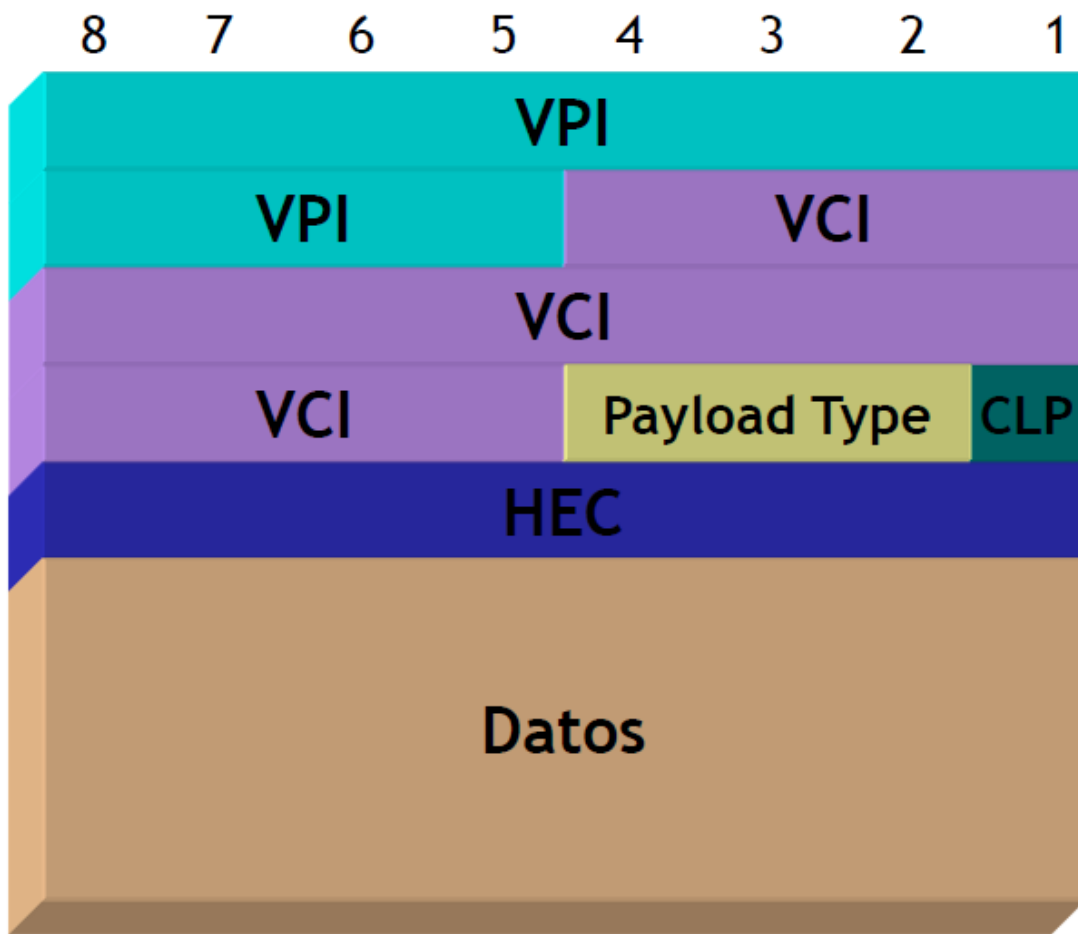
- Orientado a la conexión. Se establece conexión entre ambas estaciones.
- Capa 2 de OSI

Formato:




Tamaño fijo

Formato de celda UNI (ATM también)



Formato de celda NNI (ATM también)



Cada trama va a tener: 1500 (capacidad de datos de Ethernet) - 20 (cabecera IP) = 1480 bytes

El mínimo de datos necesarios en el ping va a ser:

$$1480 * (n-1) + 1 - 8.$$

Notas sobre parcial

- Preguntas 1 y 2 son sobre el TL 5 y 6.
- Acortar pasos para parcial: usar capturas en aula del labo para hacer los TL.

Explicación más completa:

¿Cuál es el número mínimo de Bytes con que debe ejecutarse la aplicación PING extendido para que se produzca una fragmentación de un datagrama IP sobre Ethernet, con 44 paquetes?

Máximo datos permitidos por Ethernet: 1500

Cabecera IP: 20 bytes (una en cada fragmento de PDU ethernet)

Ergo: cada fragmento contendrá 1480 bytes de datos.

Cabecera ICMP (ping): 8 bytes (son parte del paquete de datos completo encapsulado en IP, no se repite).

$$43 \text{ paquetes} * 1480 \text{ bytes} + 1 \text{ byte} - 8 \text{ bytes} = 63633 \text{ bytes}$$

(esto es 1 byte más que 43 paquetes: mínimo para tener 44 paquetes).

¿Cuál es el número de paquetes IP que se generan en una red con una MTU de 1000 bytes si la aplicación de red HTTP encapsula 125.990 bytes en el protocolo de capa 4? Demuestre la fundamentación de su estimación o cálculo (al reverso de la hoja).

Eth	Head	MTU = 1000	
IP		H: 20 bytes	(980)
TCP		H: 20 bytes	(mensaje HTTP de 125990)

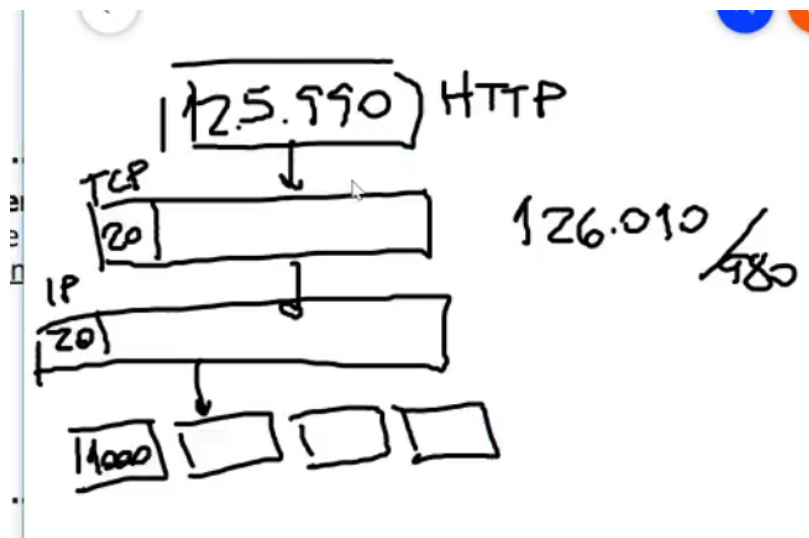
IP va a fragmentar el mensaje TCP = 125.990 bytes (HTTP) + 20 bytes (cabecera TCP)

Con un tamaño de datos de TCP: 980 bytes

Fórmula:

(Datos HTTP + cabecera TCP) / (MTU - cabecera IP) = resultado que se debe redondear hacia el siguiente entero

Resultado = $(125990 + 20) / (1000 - 20) = 126010 / 980 = 128,56 \rightarrow 129$ paquetes



Flags IP:

	Flags			Bits de desplazamiento
	Sin uso	DF: no fragmentar	MF: Más fragmentos	
General				
1er fragmento	0	0	1	0
Penúltimo	0	0	1	No 0
Último	0	0	0	No 0

—