

# Dispositivos de Interconexión en Redes Locales

## Sistemas Telemáticos

Departamento de Teoría de la Señal y Comunicaciones y  
Sistemas Telemáticos y Computación (GSyC)

Enero de 2018



©2018 Grupo de Sistemas y Comunicaciones.  
Algunos derechos reservados.  
Este trabajo se distribuye bajo la licencia  
Creative Commons Attribution Share-Alike  
disponible en <http://creativecommons.org/licenses/by-sa/3.0/es>

# Contenidos

- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

# Contenidos

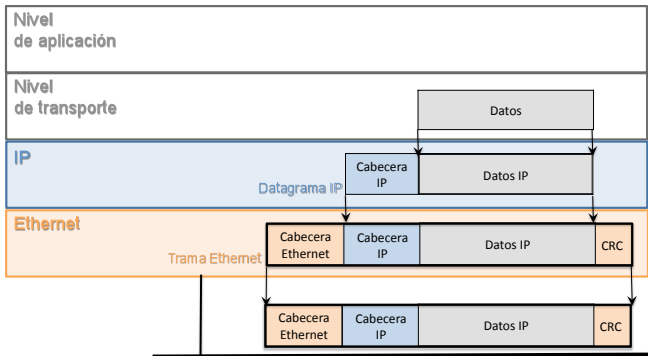
- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

# Contenidos

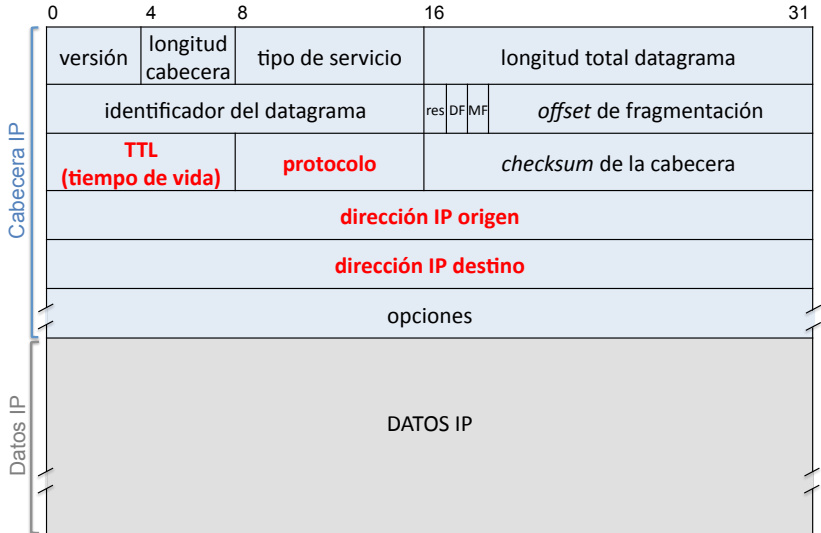
- 1 Repaso
  - Datagramas IP
  - Direcciones IP
  - Trayecto de un paquete IP
  - Protocolo ICMP
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
  - Máquinas de la misma subred IP a través de un *switch*
  - Máquinas de diferentes subredes IP a través de un *router*
  - Interconexión: *Switch versus router*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs
- 8 Referencias

# Encapsulación

- Un datagrama IP se encapsula dentro de la parte de datos de la trama del nivel de enlace.
- Si el nivel de enlace es Ethernet, el datagrama IP viaja en la parte de datos de la trama Ethernet.



# Formato del datagrama IP



# Contenidos

- 1 Repaso
  - Datagramas IP
  - Direcciones IP
  - Trayecto de un paquete IP
  - Protocolo ICMP
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
  - Máquinas de la misma subred IP a través de un *switch*
  - Máquinas de diferentes subredes IP a través de un *router*
  - Interconexión: *Switch versus router*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs
- 8 Referencias



# Direcciones IP y máscaras de subred

- Cada interfaz físico conectado a una red IP tiene asignada una **dirección IP**. Ejemplo: 212.128.4.4
- La **máscara de subred** especifica qué parte de la dirección IP debe ser igual entre máquinas **adyacentes** (o vecinas).  
Ejemplo: 255.255.255.0
- La parte de la dirección IP que se mantiene fija entre máquinas vecinas se llama **identificador de red** o **identificador de subred**.
- La parte de la dirección IP que puede variar entre máquinas vecinas se llama **identificador de máquina**.
- De los 32 bits de la máscara:
  - están a uno los bits del identificador de red o subred
  - están a cero los bits del identificador de máquina

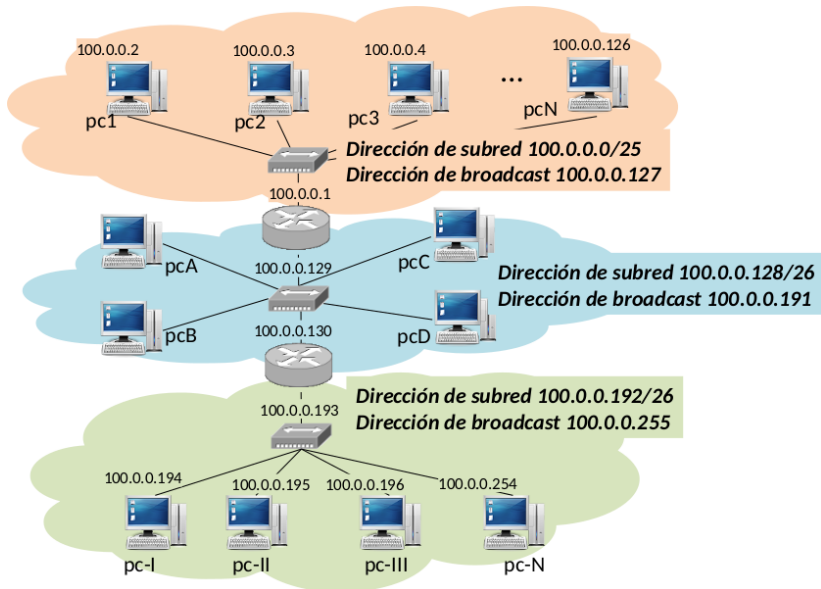
# Direcciones de red y de broadcast

- Por cada identificador de red hay dos direcciones especiales que no pueden ser asignadas a máquinas:
  - **Dirección de red:** Con los bits del identificador de máquina **todos a cero**. Identifica a toda la red. Se utiliza en las tablas de encaminamiento.  
Ejemplo: **212.128.4.0** (máscara: 255.255.255.0):
    - identificador de red: 212.128.4
    - identificador de máquina: 0
  - **Dirección de broadcast:** Con los bits del identificador de máquina **todos a uno**. Identifica a todas las máquinas de la red. Se utiliza como dirección de destino para enviar un datagrama IP a todas las máquinas de una red.  
Ejemplo: **212.128.4.255** (máscara: 255.255.255.0):
    - identificador de red: 212.128.4
    - identificador de máquina: 255

# Prefijos de red o subred

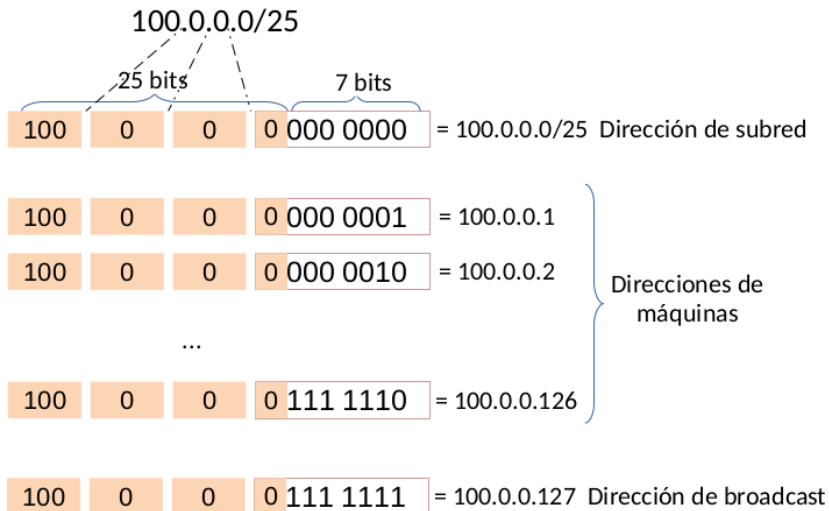
- Muchas veces se escribe la máscara de subred como un **prefijo** detrás de una dirección de red, que indica el número de bits que están a 1 en la máscara. Ejemplo: **/24**
- Ejemplos:
  - Subred 193.147.1.0/24:
    - dirección de red: 193.147.1.0
    - dirección de broadcast: 193.147.1.255
  - Subred 12.0.0.0/16:
    - dirección de red: 12.0.0.0
    - dirección de broadcast: 12.0.255.255
  - Subred 193.147.71.0/25:
    - dirección de red: 193.147.71.0
    - dirección de broadcast: 193.147.71.127

# Ejemplo (I)



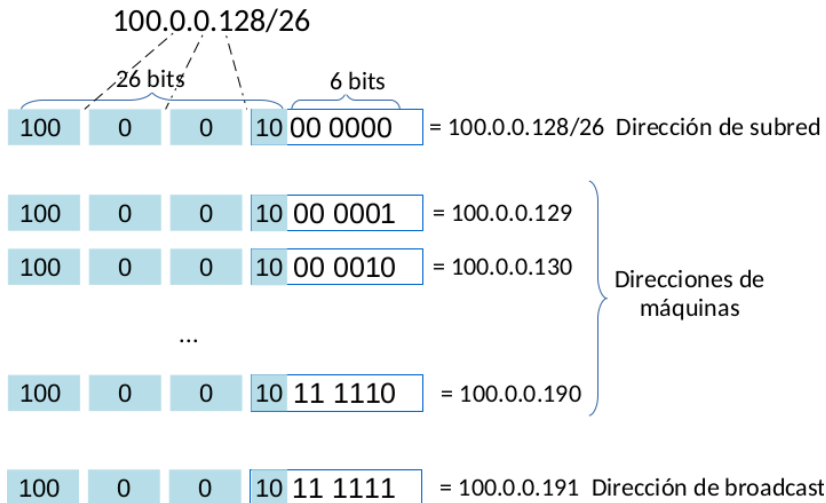
## Ejemplo (II)

Las direcciones IP de la subred 100.0.0.0/25 son las siguientes:



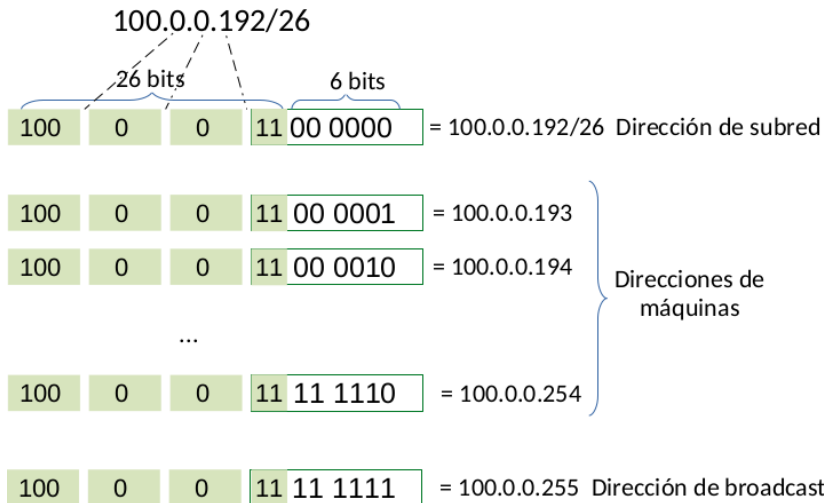
# Ejemplo (III)

Las direcciones IP de la subred 100.0.0.128/26 son las siguientes:



## Ejemplo (IV)

Las direcciones IP de la subred 100.0.0.192/26 son las siguientes:



# Tablas de Encaminamiento IP

Las **tablas de encaminamiento** tienen el siguiente aspecto (ejemplo tomado de una máquina Linux):

```
% route
Kernel IP routing table
Destination      Gateway          Genmask         Iface
193.147.71.0     0.0.0.0         255.255.255.0   eth0
212.128.4.0      0.0.0.0         255.255.255.0   eth1
145.154.12.0     193.147.71.2    255.255.255.0   eth0
145.154.12.14    212.128.4.2     255.255.255.255 eth1
0.0.0.0          193.147.71.1    0.0.0.0         eth0
```

- **Destination/Genmask**: estos campos van unidos y determinan si la dirección destino de un paquete IP pertenece a la subred dada por la pareja de valores (Destination/Genmask).
- **Gateway**: **encaminador vecino** a la máquina en la que se está consultando la tabla de encaminamiento.



# Contenidos

- 1 Repaso
  - Datagramas IP
  - Direcciones IP
  - Trayecto de un paquete IP
  - Protocolo ICMP
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
  - Máquinas de la misma subred IP a través de un *switch*
  - Máquinas de diferentes subredes IP a través de un *router*
  - Interconexión: *Switch versus router*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs
- 8 Referencias

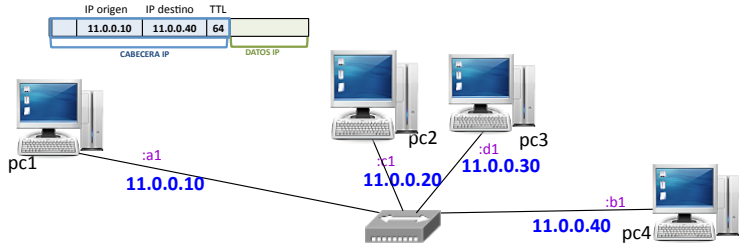
# Procesando un paquete IP en envío

- Se consulta la tabla de encaminamiento, desde las entradas más específicas a las más generales.
- Cuando se encuentra una entrada de la tabla en la que se cumple que la dirección IP destino del paquete pertenece a la subred definida por la pareja (Destination, Genmask) se usa su valor en la columna Gateway como siguiente salto:
  - **Si el campo Gateway tiene un valor distinto de 0.0.0.0:** la dirección IP destino no es una máquina vecina y hay que enviar a través del nivel de enlace el paquete IP en una trama Ethernet dirigida a Gateway. Se requiere conocer su dirección Ethernet, si ésta no se encuentra en la caché de ARP será necesario realizar una solicitud de ARP.
  - **Si el campo Gateway = 0.0.0.0:** la dirección IP destino es una máquina vecina y hay que enviar a través del nivel de enlace el paquete IP en una trama Ethernet dirigida a la máquina destino final del paquete. Se requiere conocer su dirección Ethernet, si ésta no se encuentra en la caché de ARP será necesario realizar una solicitud de ARP.

# Procesando un paquete IP en recepción

- El nivel Ethernet procesa una trama si la dirección Ethernet destino es Broadcast o dicha dirección Ethernet pertenece a una de sus interfaces. En cuyo caso comprueba el CRC y se lo pasa al protocolo que se indica en la cabecera Ethernet.
  - Si el protocolo es ARP, la solicitud será atendida si se está preguntando por una dirección IP configurada en la máquina. La respuesta será atendida porque la dirección Ethernet destino será la de la máquina que ha procesado la trama Ethernet.
  - Si el protocolo es IP:
    - Si la dirección IP destino del paquete está configurada en la máquina, el paquete ha llegado a su destino final.
    - Si la dirección IP destino del paquete no está configurada en la máquina, el paquete no ha llegado a su destino final y podrá ser encaminado (comenzando el proceso del paquete en emisión).

# Máquinas vecinas: Envío de pc1 a pc4



## Envío pc1 -> pc4

- pc1 dispone de un datagrama IP para enviar a pc4

# Máquinas vecinas: Envío de pc1 a pc4

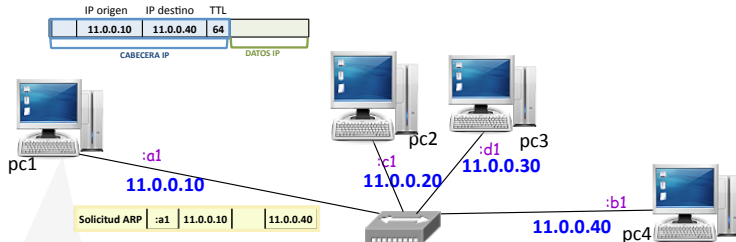


Tabla de encaminamiento en pc1

Destino	Gateway	Máscara
11.0.0.0	0.0.0.0	255.255.255.0

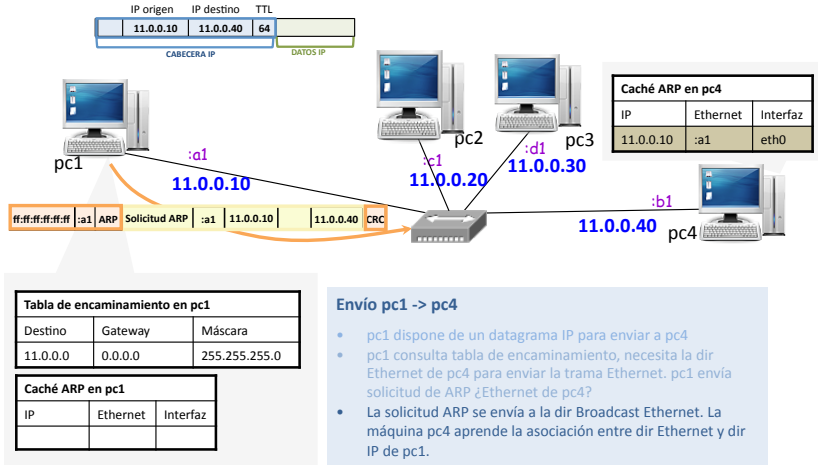
Caché ARP en pc1

IP	Ethernet	Interfaz

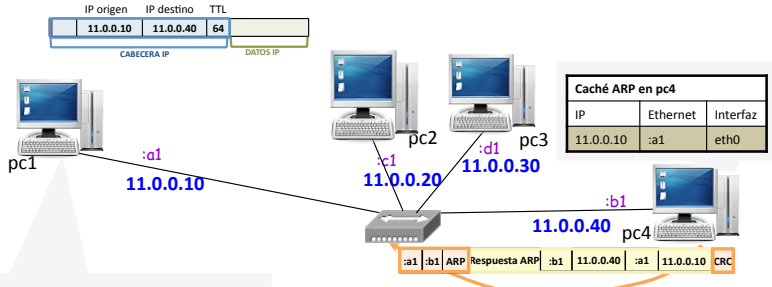
## Envío pc1 -> pc4

- pc1 dispone de un datagrama IP para enviar a pc4
- pc1 consulta tabla de encaminamiento, necesita la dir Ethernet de pc4 para enviar la trama Ethernet. pc1 envía solicitud de ARP ¿Ethernet de pc4?

# Máquinas vecinas: Envío de pc1 a pc4



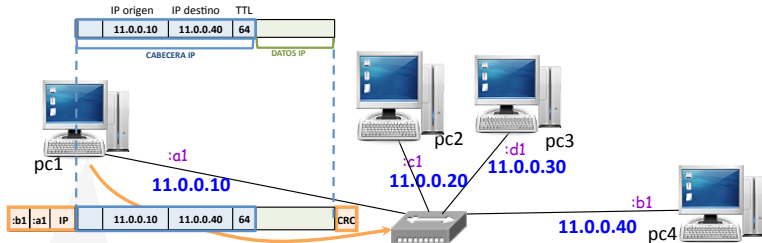
# Máquinas vecinas: Envío de pc1 a pc4



## Envío pc1 -> pc4

- pc1 dispone de un datagrama IP para enviar a pc4
- pc1 consulta tabla de encaminamiento, necesita la dir Ethernet de pc4 para enviar la trama Ethernet. pc1 envía solicitud de ARP ¿Ethernet de pc4?
- La solicitud ARP se envía a la dir Broadcast Ethernet. La máquina pc4 aprende la asociación entre dir Ethernet y dir IP de pc1.
- pc4 responde a pc1 con mensaje ARP su dir Ethernet y pc1 la apunta en su caché ARP

# Máquinas vecinas: Envío de pc1 a pc4



**Tabla de encaminamiento en pc1**

Destino	Gateway	Máscara
11.0.0.0	0.0.0.0	255.255.255.0

**Caché ARP en pc1**

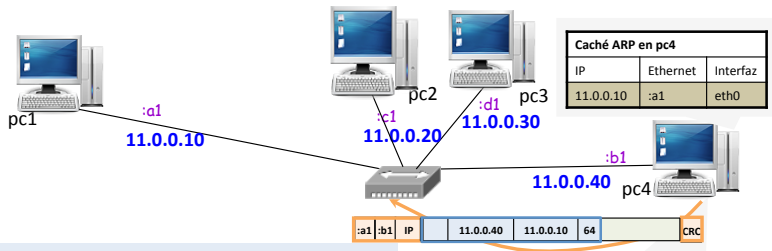
IP	Ethernet	Interfaz
11.0.0.40	:b1	eth0

## Envío pc1 -> pc4

- pc1 dispone de un datagrama IP para enviar a pc4
- pc1 consulta tabla de encaminamiento, necesita la dir Ethernet de pc4 para enviar la trama Ethernet. pc1 envía solicitud de ARP ¿Ethernet de pc4?
- La solicitud ARP se envía a la dir Broadcast Ethernet. La máquina pc4 aprende la asociación entre dir Ethernet y dir IP de pc1.
- pc4 responde a pc1 con mensaje ARP su dir Ethernet y pc1 la apunta en su caché ARP
- pc1 envía trama Ethernet a pc4



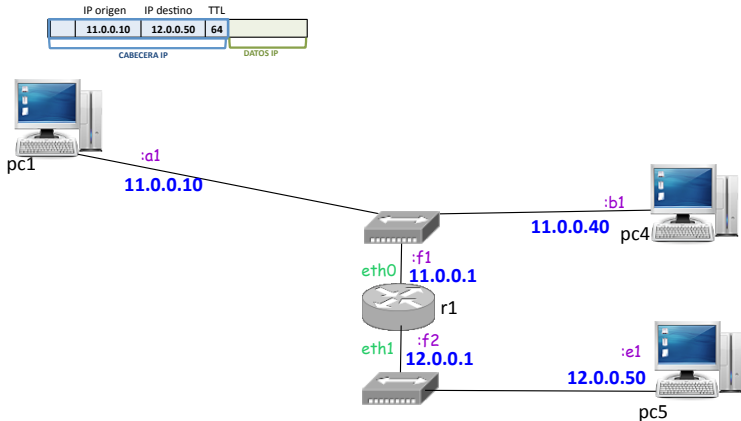
# Máquinas vecinas: Envío de pc4 a pc1



## Envío pc4 -> pc1

- Si ahora pc4 quiere "responder" a pc1 con un datagrama IP, lo prepara, mira su tabla de encaminamiento, y al tener ya en la caché de ARP la dirección Ethernet de pc1, construye la trama con el datagrama IP directamente: no se necesita ejecutar el protocolo ARP.

# Máquinas NO vecinas: Envío de pc1 a pc5



# Máquinas NO vecinas: Envío de pc1 a pc5

IP origen	IP destino	TTL
11.0.0.10	12.0.0.50	64

CABECERA IP      DATOS IP



pc1

:a1  
11.0.0.10

Solicitud ARP	:a1	11.0.0.10	11.0.0.1
---------------	-----	-----------	----------

**Tabla de encaminamiento en pc1**

Destino	Gateway	Máscara
11.0.0.0	0.0.0.0	255.255.255.0
0.0.0.0	11.0.0.1	0.0.0.0

**Caché ARP en pc1**

IP	Ethernet	Interfaz
11.0.0.40	:b1	eth0



eth0

:f1

11.0.0.1

eth1

:f2

12.0.0.1

r1

eth0

:f1

11.0.0.1

eth1

:f2

12.0.0.1

r1

:b1

11.0.0.40

pc4



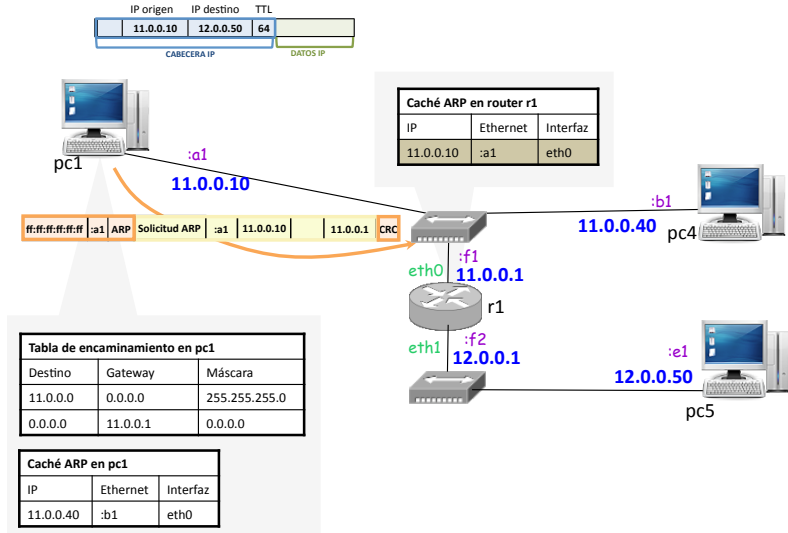
:e1

12.0.0.50

pc5



# Máquinas NO vecinas: Envío de pc1 a pc5



# Máquinas NO vecinas: Envío de pc1 a pc5

IP origen	IP destino	TTL
11.0.0.10	12.0.0.50	64

CABECERA IP      DATOS IP



pc1

:a1  
11.0.0.10

:a1	:f1	ARP	Respuesta ARP	:f1	11.0.0.1	:a1	11.0.0.10	CRC
-----	-----	-----	---------------	-----	----------	-----	-----------	-----

Caché ARP en router r1		
IP	Ethernet	Interfaz
11.0.0.10	:a1	eth0

:b1  
11.0.0.40



pc4

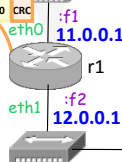


Tabla de encaminamiento en pc1

Destino	Gateway	Máscara
11.0.0.0	0.0.0.0	255.255.255.0
0.0.0.0	11.0.0.1	0.0.0.0

Caché ARP en pc1

IP	Ethernet	Interfaz
11.0.0.40	:b1	eth0
11.0.0.1	:f1	eth0

:e1  
12.0.0.50



pc5

# Máquinas NO vecinas: Envío de pc1 a pc5

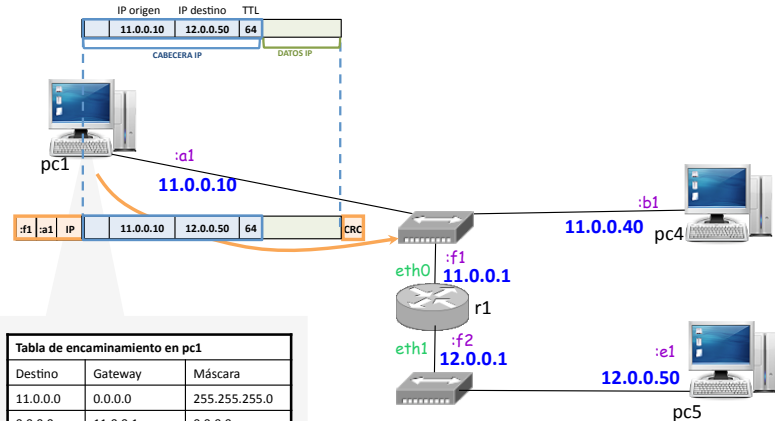


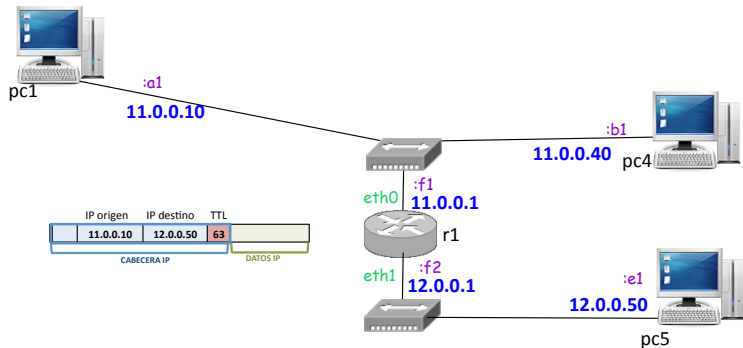
Tabla de encaminamiento en pc1

Destino	Gateway	Máscara
11.0.0.0	0.0.0.0	255.255.255.0
0.0.0.0	11.0.0.1	0.0.0.0

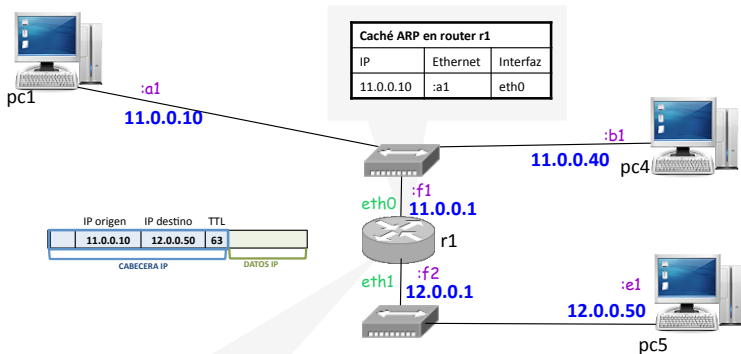
Caché ARP en pc1

IP	Ethernet	Interfaz
11.0.0.40	:b1	eth0
11.0.0.1	:f1	eth0

# Máquinas NO vecinas: Envío de pc1 a pc5

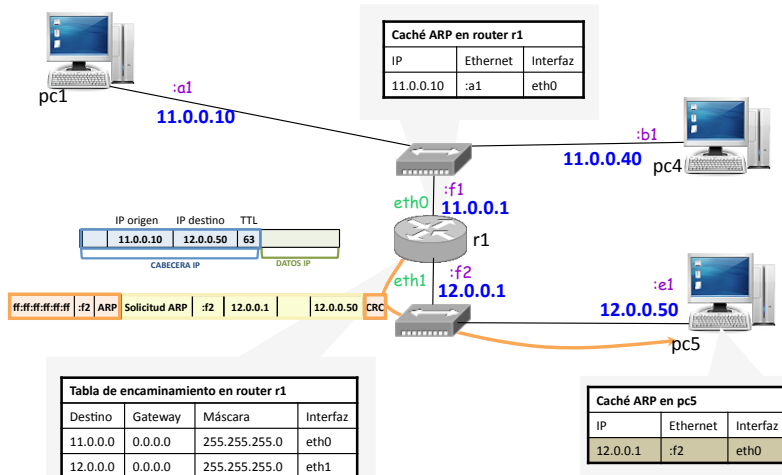


# Máquinas NO vecinas: Envío de pc1 a pc5

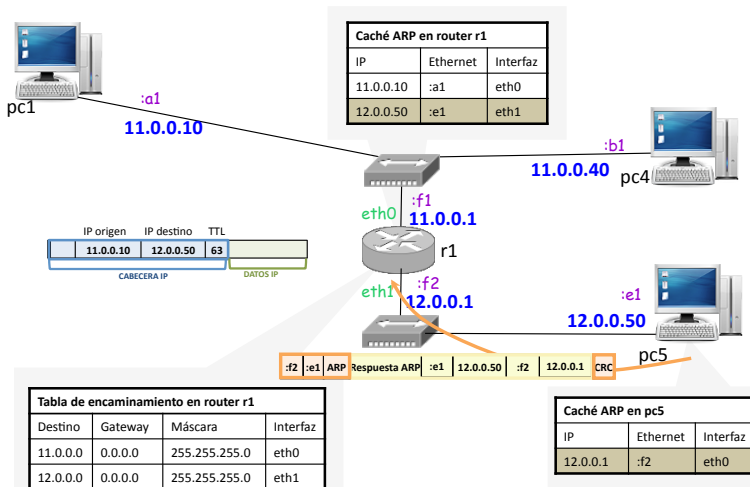




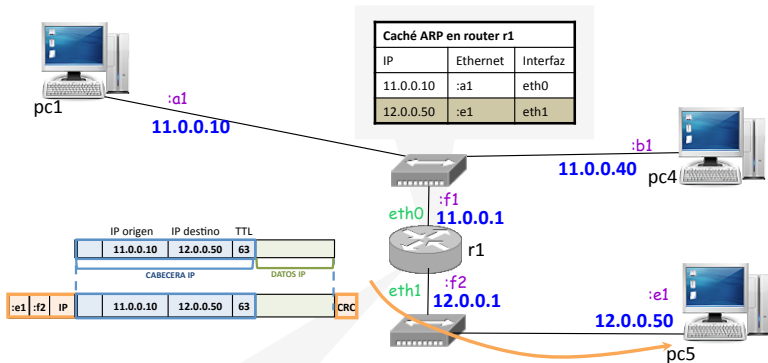
# Máquinas NO vecinas: Envío de pc1 a pc5



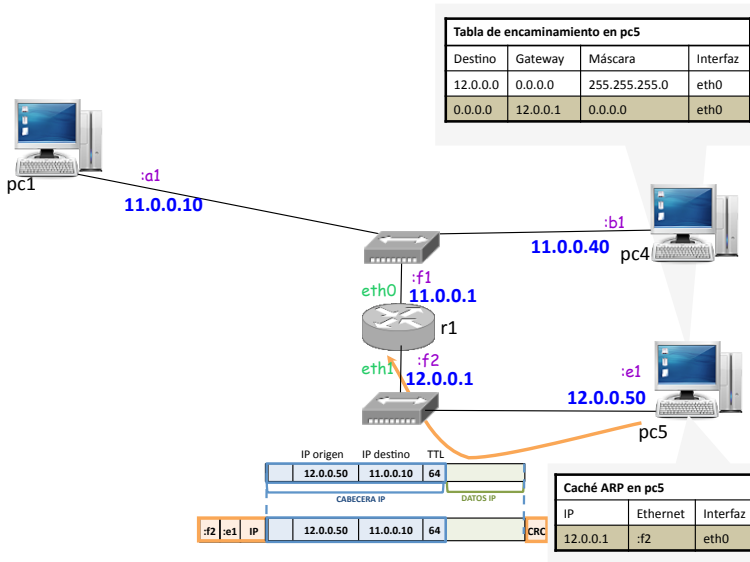
# Envío de pc1 a pc5



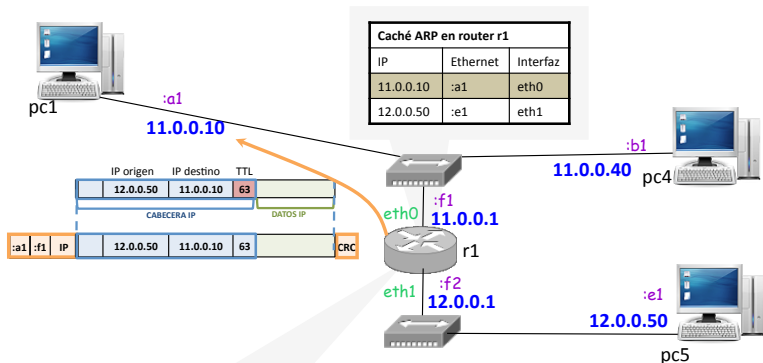
# Envío de pc1 a pc5



# Envío de pc5 a pc1



# Envío de pc5 a pc1



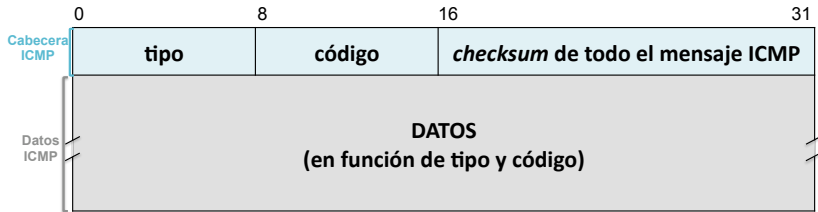
Destino	Gateway	Máscara	Interfaz
11.0.0.0	0.0.0.0	255.255.255.0	eth0
12.0.0.0	0.0.0.0	255.255.255.0	eth1

# Contenidos

- 1 Repaso
  - Datagramas IP
  - Direcciones IP
  - Trayecto de un paquete IP
  - **Protocolo ICMP**
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
  - Máquinas de la misma subred IP a través de un *switch*
  - Máquinas de diferentes subredes IP a través de un *router*
  - Interconexión: *Switch versus router*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs
- 8 Referencias

# ICMP (*Internet Control Message Protocol*)

- Este protocolo se utiliza para comunicar condiciones de error entre máquinas y para realizar algunas funciones de diagnóstico.
- Los mensajes ICMP se transmiten encapsulados dentro de datagramas IP.
- Formato de los mensajes ICMP:



# Mensajes ICMP

Algunos mensajes ICMP:

tipo	código	descripción
0	0	respuesta de eco
3	0	destino inalcanzable: red inalcanzable
3	1	destino inalcanzable: máquina inalcanzable
3	3	destino inalcanzable: puerto inalcanzable
8	0	solicitud de eco
11	0	tiempo excedido: $TTL = 0$
12	1	cabecera IP incorrecta: falta una opción
13	0	solicitud de marca de tiempo
14	0	respuesta de marca de tiempo

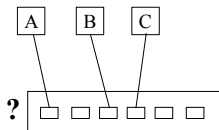


# Contenidos

- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

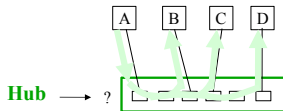
# Introducción

- El protocolo Ethernet (802.3) se concibió como una red en bus
- El cable es un único par al que se conectan todas la máquinas:
  - Problemas para cablear un edificio
  - Propenso a errores: un corte en el cable deja sin red a todas
- Topología en estrella
  - Se adapta bien a cablear edificios, sobre todo cuando empezó a utilizarse cable de pares trenzados (TP: *twisted pair*).



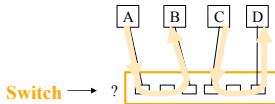
- Dispositivos:
  - Concentrador (*hub*)
  - Conmutador (*switch*)

# Concentrador (*hub*)



- Un *hub* se comporta como un **repetidor multipuerto**:
  - La señal que recibe por uno de sus puertos la copia en todos los demás: si A envía una trama a B, el *hub* (al recibirla) la copia en los puertos que le comunican con B, C y D.
  - Copia cualquier tipo de señal, incluso si es ruido o el producto de una colisión.
  - Va copiando “bit a bit”, según va recibiendo bits. Eso implica que el *hub* no puede gestionar dos o más tramas que le lleguen simultáneamente.
  - Por ello:
    - Hay colisión si dos estaciones transmiten a la vez
    - Todos los puertos deben ser de la misma velocidad
  - A veces al *hub* se le llama “switch de nivel 1” o “switch no inteligente”.
- Una Ethernet con *Hub* es EQUIVALENTE a una Ethernet de bus.
- En las figuras seguiremos utilizando a veces la representación del bus como equivalente a la interconexión de máquinas mediante un *hub*.

# Conmutador (*switch*)



- Un *switch* reconoce el formato de las tramas Ethernet, y mira sus direcciones de origen y destino:
  - El reenvío de tramas por parte del *switch* se hace mediante **almacenamiento y reenvío** (*store-and-forward*): las tramas que se van recibiendo se almacenan en un *buffer* de tramas del que se van sacando para su reenvío.
  - A la hora de reenviar una trama, el *switch* mira la dirección de destino de la misma:
    - si conoce en qué puerto está conectado el destino, sólo reenvía la trama por ese puerto
    - si no lo conoce, reenvía la trama por todos los puertos
  - El *switch* aprende las máquinas que hay en cada puerto mirando las direcciones de origen y apuntándolas en la **Tabla de direcciones Ethernet aprendidas**.
  - **No hay colisiones**: A puede enviar a B y simultáneamente C a D. Incluso A y B pueden transmitir simultáneamente a C: el límite lo impone el tamaño del *buffer* de tramas del *switch*.
  - Los puertos pueden tener diferentes velocidades: 10/100/1000 Mbps.
  - A veces al *switch* se le llama **bridge**, aunque éste término suele reservarse para un *switch* de sólo 2 puertos.
- Una Ethernet con *Switch* es **BASTANTE DIFERENTE** a una Ethernet de bus.

# Switch: Aprendizaje de direcciones (1)

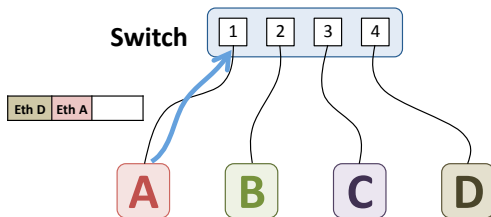


Tabla de direcciones aprendidas por el switch

Ethernet	Puerto

- El *switch* está recién arrancado
- A envía una trama a D

## Switch: Aprendizaje de direcciones (2)

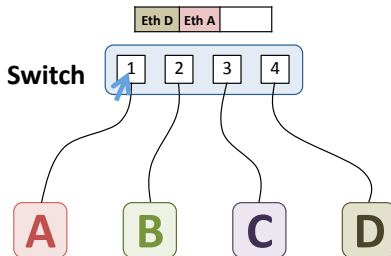


Tabla de direcciones aprendidas por el switch

Ethernet	Puerto
Eth A	1

- El *switch* aprende que A está en el puerto 1
- Aún no sabe dónde está D

# Switch: Aprendizaje de direcciones (3)

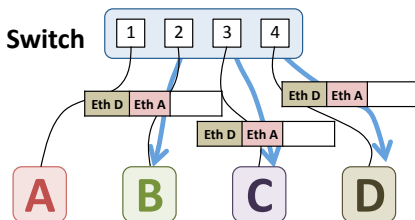
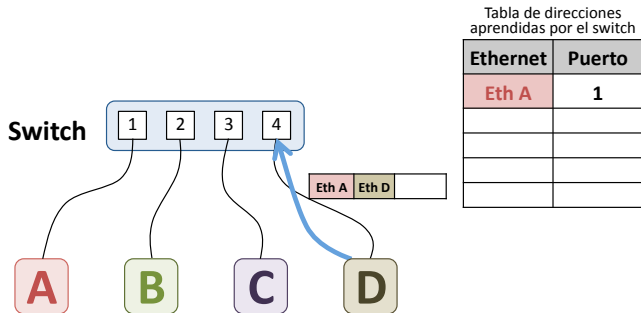


Tabla de direcciones aprendidas por el switch

Ethernet	Puerto
Eth A	1

- El *switch* envía la trama por los puertos 2, 3 y 4
- B y C descartarán la trama por no ser para ellos.

# Switch: Aprendizaje de direcciones (4)



- D envía ahora una trama a A



# Switch: Aprendizaje de direcciones (5)

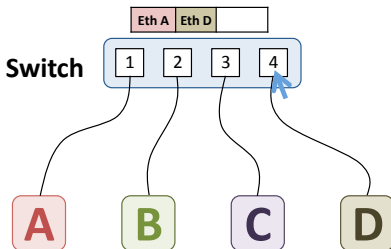
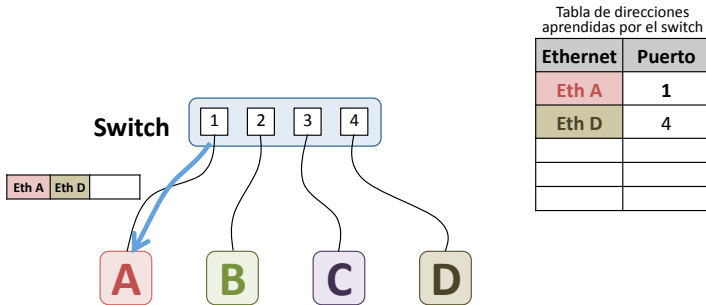


Tabla de direcciones aprendidas por el switch

Ethernet	Puerto
Eth A	1
Eth D	4

- El *switch* aprende que D está en el puerto 4

## Switch: Aprendizaje de direcciones (6)



- Como el *switch* ya sabe dónde está A, reenvía la trama sólo por el puerto 1.

# Aprendizaje y reenvío del switch

- Las entradas en **la tabla de direcciones aprendidas** caducan si no se utilizan durante un tiempo. El tiempo que una dirección Ethernet lleva almacenado en la tabla se guarda en un contador *ageing timer* (como máximo 5 minutos, valor por defecto).
- Funciones principales del switch:
  - **Aprendizaje:** el switch aprende las direcciones Ethernet origen en las tramas que recibe asociándolas a una de sus interfaces y guardando esta información en su tabla de direcciones aprendidas. Si esa dirección ya la tiene aprendida, reinicia el contador *ageing timer*.
  - **Reenvío:** el switch reenvía en función de la tabla de direcciones aprendidas. Si la dirección Ethernet destino de la trama es Broadcast o no la tiene en su tabla, reenvía la trama por todas las interfaces salvo por donde le ha llegado.

# Contenidos

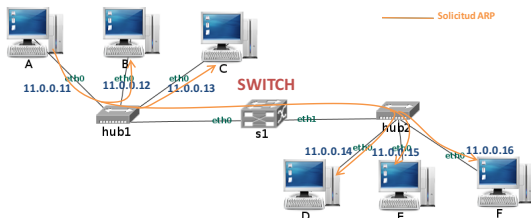
- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches***
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

# Contenidos

- 1 Repaso
  - Datagramas IP
  - Direcciones IP
  - Trayecto de un paquete IP
  - Protocolo ICMP
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
  - Máquinas de la misma subred IP a través de un *switch*
  - Máquinas de diferentes subredes IP a través de un *router*
  - Interconexión: *Switch versus router*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs
- 8 Referencias

# Máquinas de la misma subred IP a través de un *switch*

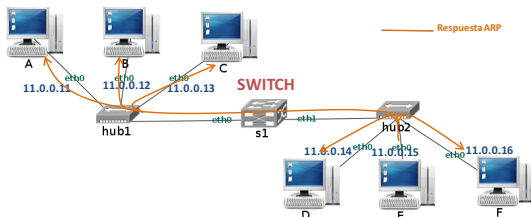
## Caso 1a: A envía a E



- El *switch*, aunque tiene dos tarjetas Ethernet, no tendrá asignada ninguna dirección IP (al menos para su trabajo como *switch*).
- Si A quiere enviar un datagrama IP a E:
  - La tabla de encaminamiento de A le dice que E es vecino suyo
  - A envía una **solicitud de ARP preguntando con la IP de E**. s1 la propaga por ser un *broadcast*.

# Máquinas de la misma subred IP a través de un *switch*

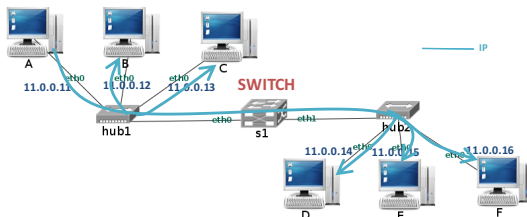
## Caso 1a: A envía a E



- El *switch*, aunque tiene dos tarjetas Ethernet, no tendrá asignada ninguna dirección IP (al menos para su trabajo como *switch*).
- Si A quiere enviar un datagrama IP a E:
  - La tabla de encaminamiento de A le dice que E es vecino suyo
  - A envía una solicitud de ARP preguntando con la IP de E. s1 la propaga por ser un *broadcast*.
  - E responde a A con su dirección Ethernet. s1 la propaga porque ya sabe que A está en su eth0.

# Máquinas de la misma subred IP a través de un *switch*

## Caso 1a: A envía a E



- El *switch*, aunque tiene dos tarjetas Ethernet, no tendrá asignada ninguna dirección IP (al menos para su trabajo como *switch*).
- Si A quiere enviar un datagrama IP a E:
  - La tabla de encaminamiento de A le dice que E es vecino suyo
  - A envía una solicitud de ARP preguntando con la IP de E. s1 la propaga por ser un *broadcast*.
  - E responde a A con su dirección Ethernet. s1 la propaga porque ya sabe que A está en su eth0.
  - A envía el datagrama IP dentro de una trama Ethernet con destino la dirección Ethernet de E:

Eth.Dst	Eth.Or	IP.Or	IP.Dst	
E(eth0)	A(eth0)	IP	11.0.0.11	11.0.0.15 ...

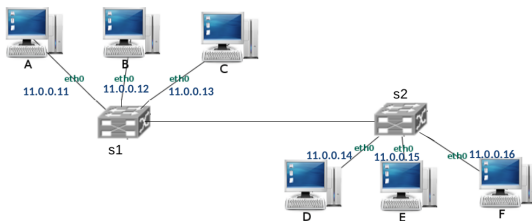
- s1 propaga la trama que contiene el datagrama IP porque ya sabe que E está en su eth1.
- La trama que contiene el datagrama IP es la misma en el hub1 y en el hub2.



# Máquinas de la misma subred IP a través de un *switch*

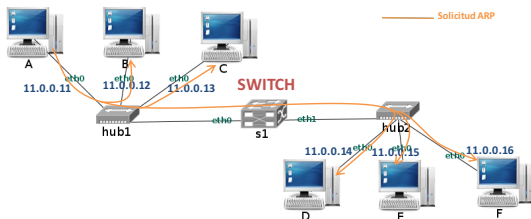
## Caso 1b: A envía a E

- ¿Qué crees que hubiera ocurrido si A envía a E en un escenario como éste?
- ¿Cuántos mensajes recibirían las máquinas que no son ni A ni E?



# Máquinas de la misma subred IP a través de un *switch*

## Caso 2a: A envía a C

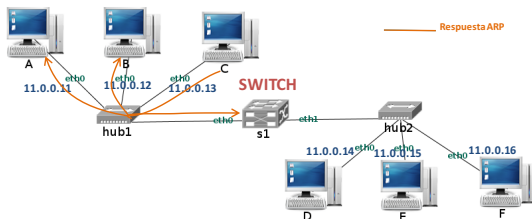


- Si A quiere enviar un datagrama IP a C:

- La tabla de encaminamiento de A le dice que C es vecino suyo
- A envía una **solicitud de ARP preguntando con la IP de C**. s1 la propaga por ser un *broadcast* y aprende que la dirección Ethernet A(eth0) se encuentra en la interfaz s1(eth0).

# Máquinas de la misma subred IP a través de un *switch*

## Caso 2a: A envía a C

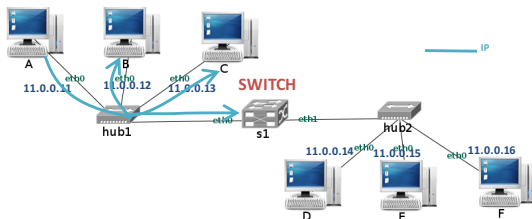


- Si A quiere enviar un datagrama IP a C:

- La tabla de encaminamiento de A le dice que C es vecino suyo
- A envía una solicitud de ARP preguntando con la IP de C. s1 la propaga por ser un *broadcast* y aprende que la dirección Ethernet A(eth0) se encuentra en la interfaz s1(eth0).
- **C responde a A con su dirección Ethernet.** s1 NO la propaga porque ya sabe que A está en su eth0, y además, s1 aprende que la dirección Ethernet C(eth0) se encuentra en la interfaz s1(eth0).

# Máquinas de la misma subred IP a través de un *switch*

## Caso 2a: A envía a C



### • Si A quiere enviar un datagrama IP a C:

- La tabla de encaminamiento de A le dice que C es vecino suyo
- A envía una solicitud de ARP preguntando con la IP de C. s1 la propaga por ser un *broadcast* y aprende que la dirección Ethernet A(eth0) se encuentra en la interfaz s1(eth0).
- C responde a A con su dirección Ethernet. s1 NO la propaga porque ya sabe que A está en su eth0, y además, s1 aprende que la dirección Ethernet C(eth0) se encuentra en la interfaz s1(eth0).
- A envía el datagrama IP dentro de una trama Ethernet con destino la dirección Ethernet de C:

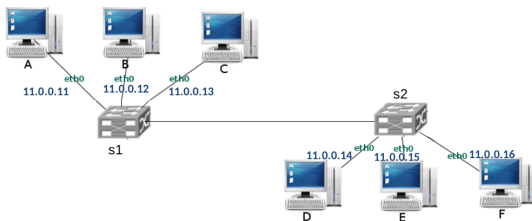
Eth.Dst	Eth.Or	IP.Or	IP.Dst	
C(eth0)	A(eth0)	IP	11.0.0.11	11.0.0.15 ...

- s1 NO propaga la trama que contiene el datagrama IP porque ya sabe que C está en s1(eth0).

# Máquinas de la misma subred IP a través de un *switch*

## Caso 2b: A envía a C

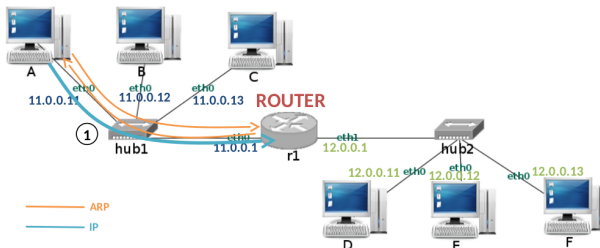
- ¿Qué crees que hubiera ocurrido si A envía a C en un escenario como éste?
- ¿Cuántos mensajes recibirían las máquinas que no son ni A ni C?



# Contenidos

- 1 Repaso
  - Datagramas IP
  - Direcciones IP
  - Trayecto de un paquete IP
  - Protocolo ICMP
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
  - Máquinas de la misma subred IP a través de un *switch*
  - Máquinas de diferentes subredes IP a través de un *router*
  - Interconexión: *Switch versus router*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs
- 8 Referencias

# Máquinas de diferentes subredes IP mediante un *router* (I)



- La situación más habitual es que a cada lado del *router* las máquinas tengan direcciones IP de diferentes subredes IP.
- Cada interfaz del *router* tendrá asignada una dirección IP de la subred correspondiente.
- **Si A quiere enviar un datagrama IP a E:**

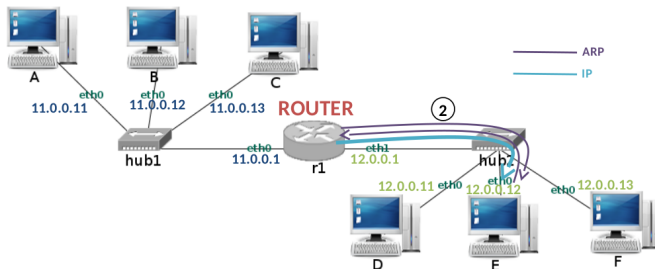
① A envía trama Ethernet dirigida a r1(eth0):

- La tabla de encaminamiento de A le dice que E no es vecino suyo y tiene que pasar por el *router*.
- A envía una solicitud de ARP preguntando con la IP 11.0.0.1 de r1.
- r1 responde a A con la dirección Ethernet de r1(eth0).
- A envía el datagrama IP dentro de una trama Ethernet con destino la dirección Ethernet de r1(eth0):

Eth.Dst	Eth.Or	IP.Or	IP.Dst	
r1(eth0)	A(eth0)	IP	11.0.0.11	12.0.0.12 ...

- Nótese que el hub1 copia por todas sus interfaces, aunque no se muestra en la figura.

# Máquinas de diferentes subredes IP mediante un *router* (II)



- Si A quiere enviar un datagrama IP a E:

- ② r1(eth1) envía trama Ethernet dirigida a E :

- El *router* consulta su tabla de encaminamiento y comprueba que E es vecino suyo. Tras el **ARP correspondiente**, reenvía el datagrama dentro de una trama con Ethernet de destino la de E, desde su dirección Ethernet r1(eth1):

Eth.Dst	Eth.Or	IP.Or	IP.Dst	
E(eth0)	r1(eth1)	IP	11.0.0.11	12.0.0.12 ...

- Nótese que el hub2 copia por todas sus interfaces, aunque no se muestra en la figura.
    - **Un router no copia una trama Ethernet de una interfaz a otra.**



# Contenidos

- 1 Repaso
  - Datagramas IP
  - Direcciones IP
  - Trayecto de un paquete IP
  - Protocolo ICMP
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 **Interconexión de máquinas a través de *hubs* y *switches***
  - Máquinas de la misma subred IP a través de un *switch*
  - Máquinas de diferentes subredes IP a través de un *router*
  - **Interconexión: *Switch versus router***
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs
- 8 Referencias

# Switch versus router

- No hay que confundir un *switch* con un *router*: ambos reenvían datos mirando una tabla, pero su comportamiento es muy diferente:
  - Un *switch* reenvía tramas Ethernet que NO van dirigidas a él, las coge “sin permiso”, las almacena, e intactas las reenvía por el puerto adecuado.
  - Un *router* reenvía datagramas IP contenidos en tramas Ethernet que SÍ van dirigidas a él (esas tramas tienen como dirección destino Ethernet la dirección de una de las interfaces del *router*) por eso recibe esas tramas, elimina sus cabeceras, y obtiene los datagramas IP que contienen. A continuación reenvía por la interfaz adecuada esos datagramas IP, contenidos en nuevas tramas Ethernet (con diferentes direcciones Ethernet de las de las tramas en que llegaron).
- Por esto:
  - la presencia de un *switch* es “transparente” para las máquinas, éstas no son conscientes de si existe, o de si están interconectadas por un *hub* o por un bus.
  - la presencia de un *router* es “manifiesta” para las máquinas, éstas le envían tramas expresamente dirigidas al *router*.

# Dominios de colisión, dominios de *broadcast*

- Un *hub* crea:
  - un solo dominio de colisión:
    - si dos máquinas transmiten a la vez se produce una colisión y el resultado de esa colisión lo reciben todas las máquinas conectadas al *hub*.
  - un solo dominio de *broadcast*:
    - las tramas de *broadcast* Ethernet se copian a todos los puertos, salvo en el puerto por el que se han recibido. Todas las máquinas conectadas al *hub* (excepto la emisora) reciben una trama de *broadcast*.
- Un *switch* crea:
  - diferentes dominios de colisión:
    - si dos máquinas transmiten a la vez NO se produce colisión, sus tramas se almacenan en el *switch* esperando a ser procesadas (aunque si hay demasiado tráfico pueden perderse tramas).
  - un solo dominio de *broadcast*:
    - las tramas de *broadcast* Ethernet se copian a todos los puertos salvo en el puerto por el que se han recibido.
- Nótese que las interfaces de un *router* aíslan tanto dominios de colisión como dominios de *broadcast*, ya que un *router* nunca reenvía tramas Ethernet, ni siquiera las tramas de *broadcast*.

# Contenidos

- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
- 4 IP Aliasing**
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

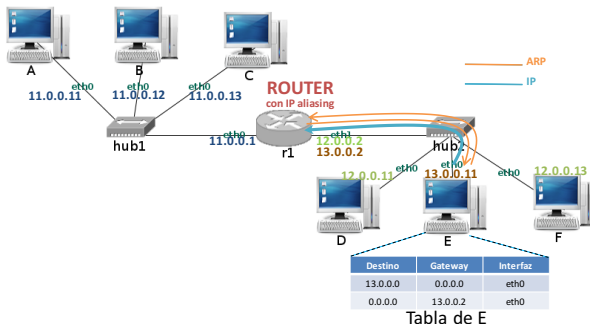
# IP Aliasing

- IP *Aliasing* consiste en asignar más de una dirección IP a una misma interfaz de red. De esta forma dicha interfaz pertenecerá a varias subredes IPs.
- Permite tener máquinas de distinta subred IP en LANs diferentes comunicadas a través de un *router*, y que una o varias interfaces de ese router sirvan de *gateway* para las máquinas de las distintas subredes conectadas al mismo nivel de enlace, sin necesidad de contar con un *switch*.

# IP Aliasing

Comunicación E → F:

(1) E envía trama Ethernet a r1



## • Si E quiere enviar un datagrama IP a F:

### ④ E envía trama Ethernet a r1(eth1):

- Ahora E tendrá a la 13.0.0.2 como *gateway* para alcanzar a F.
- La solicitud de ARP llega a r1(eth1), y r1 contesta.
- E envía el datagrama IP dentro de una trama Ethernet con destino la dirección Ethernet de r1(eth1):

Eth.Dst	Eth.Or	IP.Or	IP.Dst	
r1(eth1)	E(eth0)	IP	13.0.0.11	12.0.0.13 ...

# IP Aliasing

Comunicación E → F:

(2) r1 envía trama Ethernet a F

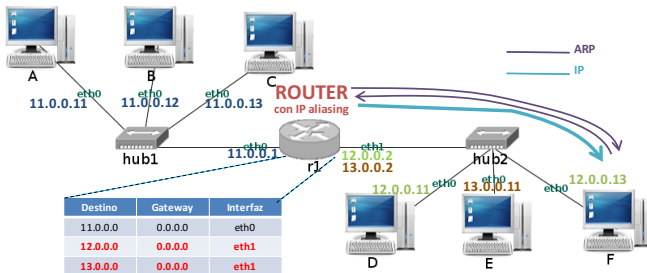


Tabla de r1

- Si E quiere enviar un datagrama IP a F:

- 2 r1(eth1) envía trama Ethernet a F:

- Tras el ARP correspondiente, r1 reenvía el datagrama dentro de una trama con Ethernet de destino la de F, desde su dirección Ethernet r1(eth1):

Eth.Dst	Eth.Or	IP.Or	IP.Dst	
F(eth0)	r1(eth1)	IP	13.0.0.11	12.0.0.13 ...

# Contenidos

- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
- 4 IP Aliasing
- 5 Proxy ARP**
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

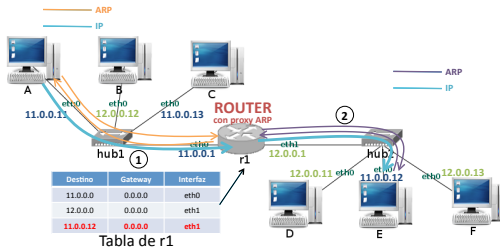


# Proxy ARP

- *Proxy ARP* permite a una máquina (normalmente un *router*) responder a solicitudes de ARP dirigidas a otras máquinas.
- La máquina que realiza la solicitud de ARP no distingue si la respuesta proviene de la máquina a la que iba dirigida la solicitud o de otra máquina que está configurada con *proxy ARP*.
- Permite tener máquinas de la misma subred IP en LANs diferentes intercomunicadas a través de un *router* que tenga activado *proxy ARP*, y **sin necesidad de contar con un switch**.

# Proxy ARP

Comunicación A → E



- Para que A pueda enviar un datagrama IP a E:

- Proxy ARP:

- r1 se configura para que responda a las solicitudes de ARP que vea en el hub1 preguntando por E.
    - r1 responderá "haciéndose pasar por E", es decir, con la dirección Ethernet de r1-eth0.

- Así, A enviará el datagrama dentro de una trama Ethernet con Ethernet destino r1-eth0:

Eth.Dst	Eth.Or	IP.Or	IP.Dst	
r1-eth0	A-eth0	IP	11.0.0.11	12.0.0.12 ...

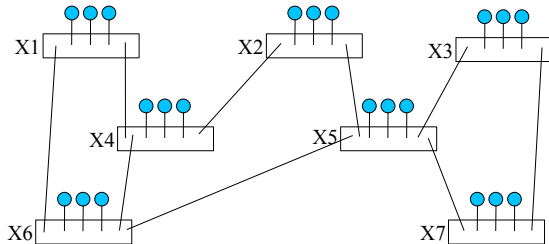
- Cuando r1 reciba esa trama, su tabla de encaminamiento debe tener una entrada adecuada que le permita reenviar el datagrama IP usando su interfaz r1-eth1:

Eth.Dst	Eth.Or	IP.Or	IP.Dst	
E-eth0	r1-eth1	IP	11.0.0.11	12.0.0.12 ...

# Contenidos

- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

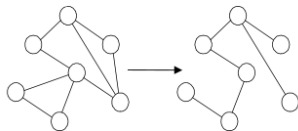
# Problema



- Las conexiones de *switch-a-switch* (en cascada) pueden causar la generación de excesivo tráfico si hay **bucles**:
  - Una trama *broadcast* se envía por todos los puertos de un *switch*. Si hay un bucle, una sola trama *broadcast* no sólo se quedaría viajando infinitamente por el bucle (no hay TTL) sino que se multiplicaría.
  - Una trama cuya dirección destino es desconocida para un *switch* se propaga por todos sus puertos. Si hay un bucle, una trama de estas características quedaría viajando por el bucle y se multiplicaría hasta que los *switches* aprendieran dónde se encuentra el destino.

# Protocolo del árbol de expansión (STP: *Spanning Tree Protocol*)

- OBJETIVO: convertir el grafo de interconexión de *switches* en un árbol, de forma que se eviten los bucles
  - Desarrollado originalmente por Radia Perlman para *bridges* (que presentan el mismo problema si se conectan en cascada), IEEE 802.1D



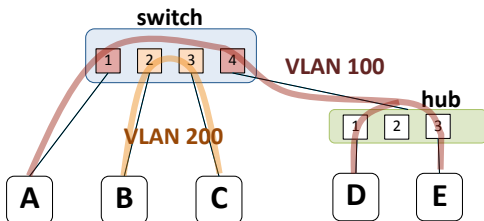
- Los *switches* intercambian mensajes de configuración para calcular el árbol de expansión:
  - BPDU: Bridge Protocol Data Unit.
- Con este algoritmo, los *switches* dejan inactivos algunos puertos de cascada para evitar los bucles
- Si algún *switch* falla, el algoritmo lo detecta y se recalcula el árbol reactivándose los puertos que sean necesarios.

# Contenidos

- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

# VLANs: Virtual LANs (802.1q)

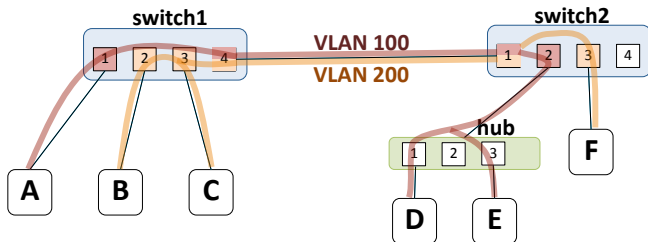
- Todos los dispositivos conectados a un mismo conjunto de hubs o switches pertenecen a la misma LAN Ethernet, tienen el **mismo dominio de broadcast**.
- Las VLANs definen una división en LANs virtuales sobre una LAN física.
  - Físicamente todas las máquinas se encuentran conectadas a la misma LAN Ethernet pero funcionan como si estuvieran en diferentes LAN Ethernet.
- Un *switch* con capacidad para definir VLANs aísla el tráfico de dispositivos que pertenecen a diferentes VLANs y que están conectados directamente a dicho *switch*.



- Hay 2 mecanismos básicos para definir VLANs en un *switch*:
  - Se especifica por cada puerto del *switch* a que VLAN pertenece dicho puerto. Todas las máquinas conectadas a un determinado puerto del *switch* pertenecerán a la VLAN de ese puerto.
  - Se define una tabla asociando máquinas a VLANs.
- Un *switch* sólo reenvía una trama (incluyendo las tramas de *broadcast*) a través de un puerto si dicho puerto pertenece a la misma VLAN que la trama.
- Dos máquinas conectadas al mismo *switch* pero configuradas en VLANs diferentes necesitan un *router* para comunicarse.

## VLANs: Virtual LANs (II)

- *Trunk* es una conexión física que puede transportar diferentes VLANs. Para distinguir el tráfico entre diferentes VLANs, es necesario que dicho tráfico contenga el identificador de VLAN a la que pertenece. Típicamente un *trunk* se define en una conexión entre 2 *switches*.
- Un *switch* sólo reenvía una trama (incluyendo las tramas de *broadcast*) a través de un *trunk* si dicho puerto pertenece a la misma VLAN que la trama.



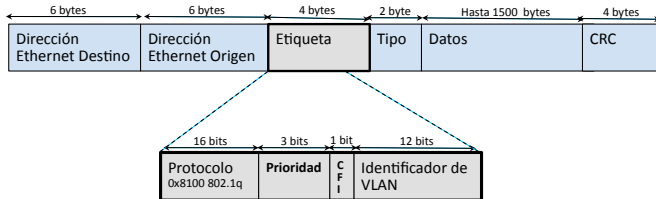


# VLANs: Virtual LANs (III)

- Las VLANs permiten reconfigurar la red sin cambiar físicamente cables o *switches* ya que la configuración se realiza vía software.
- Principalmente hay 2 estándares para definir VLANs:
  - el estándar 802.1q para Ethernet de 100Mbps (10 Mbps no está soportado).
  - el protocolo ISL (InterSwitch Link) de Cisco (previo a la estandarización de 802.1q).

# Etiqueta VLAN

- La etiqueta VLAN son 4 bytes que identifican la VLAN a la que pertenece una trama. La etiqueta VLAN contiene los siguientes campos:
  - Protocolo: Estándar VLAN que se está utilizando, en el caso de 802.1q el campo protocolo lleva el campo 0x8100.
  - Prioridad: reservados para proporcionar QoS según el estándar 802.1p (7 más prioritario)
  - CFI: Canonical Format Indicator. Tiene valor 0 en los *switches* Ethernet. Se utiliza por compatibilidad con redes Token Ring, si lleva CFI=1 no debe ser reenviado.
  - Identificador de VLAN: está formado por 12 bits (de 1 a 4094, los identificadores 0 y 4095 están reservados)



# Gestión de etiquetas VLAN

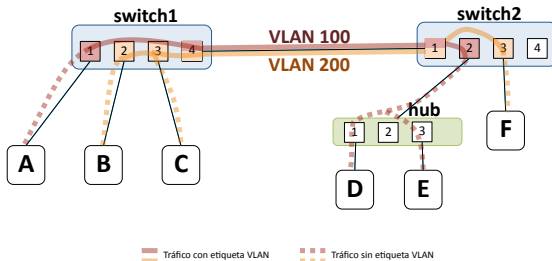
- Habitualmente la gestión de etiquetas VLAN la realizan los *switches* de forma transparente a las máquinas finales que intercambian el tráfico. Por tanto, son los *switches* los que se encargan de introducir/eliminar las etiquetas VLAN.

# Inserción/Eliminación de etiqueta VLAN

- Cuando un *switch* recibe una trama Ethernet a través de un puerto que tiene definida una VLAN:
  - si la trama no tiene la etiqueta VLAN, el *switch* le añade la etiqueta con el identificador de VLAN definido en ese puerto.
  - si la trama tiene la etiqueta VLAN, el *switch* no la modifica.
- Cuando un *switch* tiene que reenviar una trama Ethernet a través de un puerto:
  - si la **interfaz** por la que va a reenviar está conectada a las máquinas finales (no tienen configuración VLAN), el *switch* elimina la etiqueta VLAN.
  - si la **interfaz** por la que va a reenviar está conectada a dispositivos que están utilizando VLAN (por ejemplo conexión *trunk* con otro *switch*), el *switch* no modifica la etiqueta VLAN.

# Ejemplo

- Ejemplo tráfico de A  $\rightarrow$  D:
  - El *switch1* recibe una trama de A sin etiqueta VLAN a través del puerto 1. El *switch1* añade la etiqueta VLAN con identificador 100 a dicho tráfico.
  - El *switch1* debe reenviar ese tráfico a través de una interfaz del puerto 4 que está utilizando VLAN, por tanto, el *switch1* no modifica la etiqueta.
  - El *switch2* recibe una trama del *switch1* con identificador VLAN 100 y no modifica dicha etiqueta.
  - El *switch2* debe reenviar la trama con identificador VLAN 100 a través de una interfaz del puerto 2 que no está utilizando VLAN y deberá eliminar la etiqueta VLAN.



# Contenidos

- 1 Repaso
- 2 Ethernet en estrella: el *hub* y el *switch*
- 3 Interconexión de máquinas a través de *hubs* y *switches*
- 4 IP Aliasing
- 5 Proxy ARP
- 6 Conexión en cascada de *switches*: STP
- 7 VLANs: Virtual LANs

# Referencias

- A. Tanembaum, **Computer Networks (4th ed)**: apartados 4.2 y 4.7.
- L. Peterson, **Computer Networks: A Systems Approach (3rd ed)**: apartado 3.2.
- R. Perlman, **Interconnections: Bridges, Routers, Switches, and Internetworking Protocols (2nd ed)**: capítulos 5 y 3.