



Módulo 04

La Capa de Red

(Pt. 2)



Redes de Computadoras
Depto. de Cs. e Ing. de la Comp.
Universidad Nacional del Sur



Copyright

- Copyright © **2010-2024** A. G. Stankevicius
- Se asegura la libertad para copiar, distribuir y modificar este documento de acuerdo a los términos de la **GNU Free Documentation License**, versión 1.2 o cualquiera posterior publicada por la Free Software Foundation, sin secciones invariantes ni textos de cubierta delantera o trasera
- Una copia de esta licencia está siempre disponible en la página <http://www.gnu.org/copyleft/fdl.html>
- La versión transparente de este documento puede ser obtenida de la siguiente dirección:

<http://cs.uns.edu.ar/~ags/teaching>

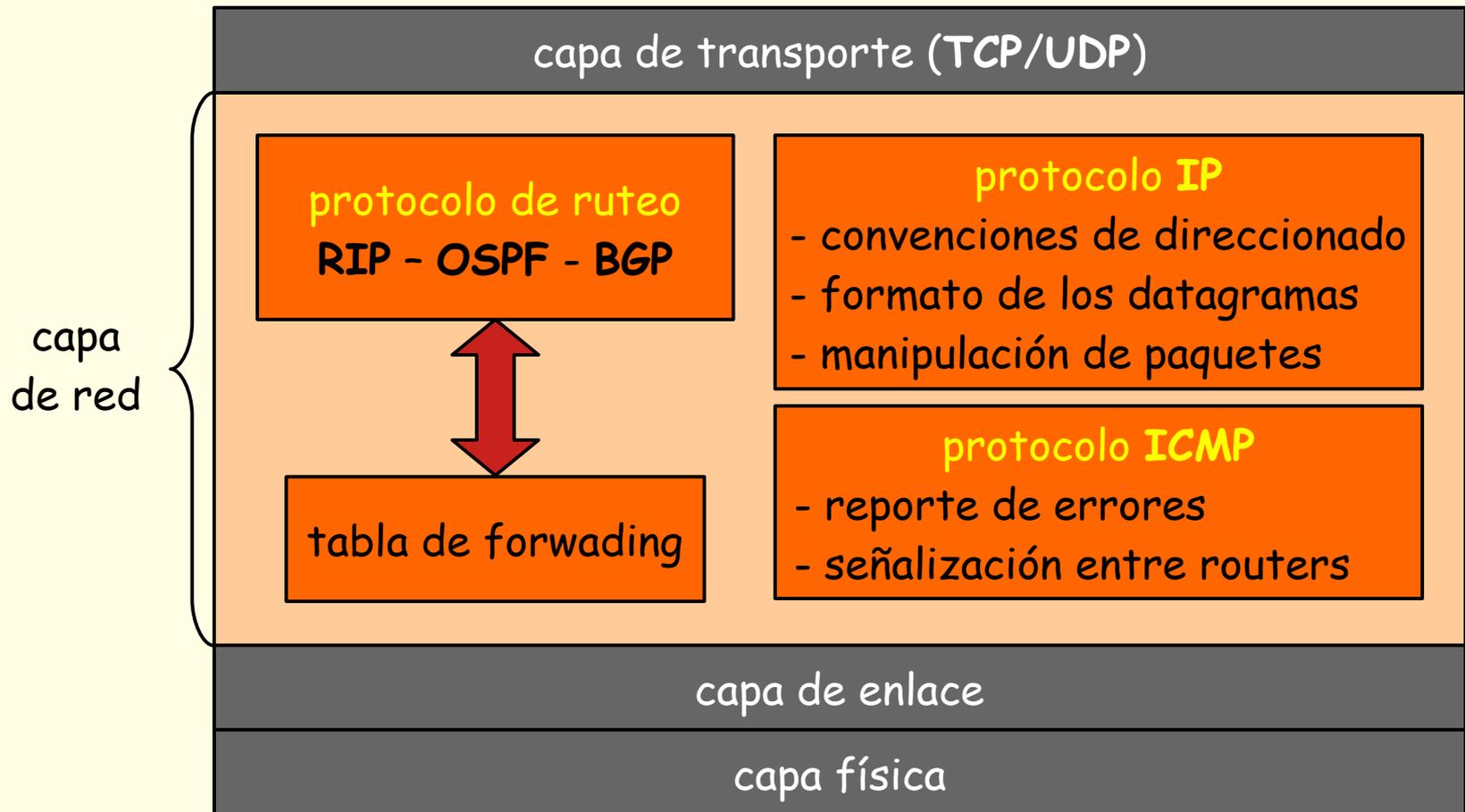


Contenidos

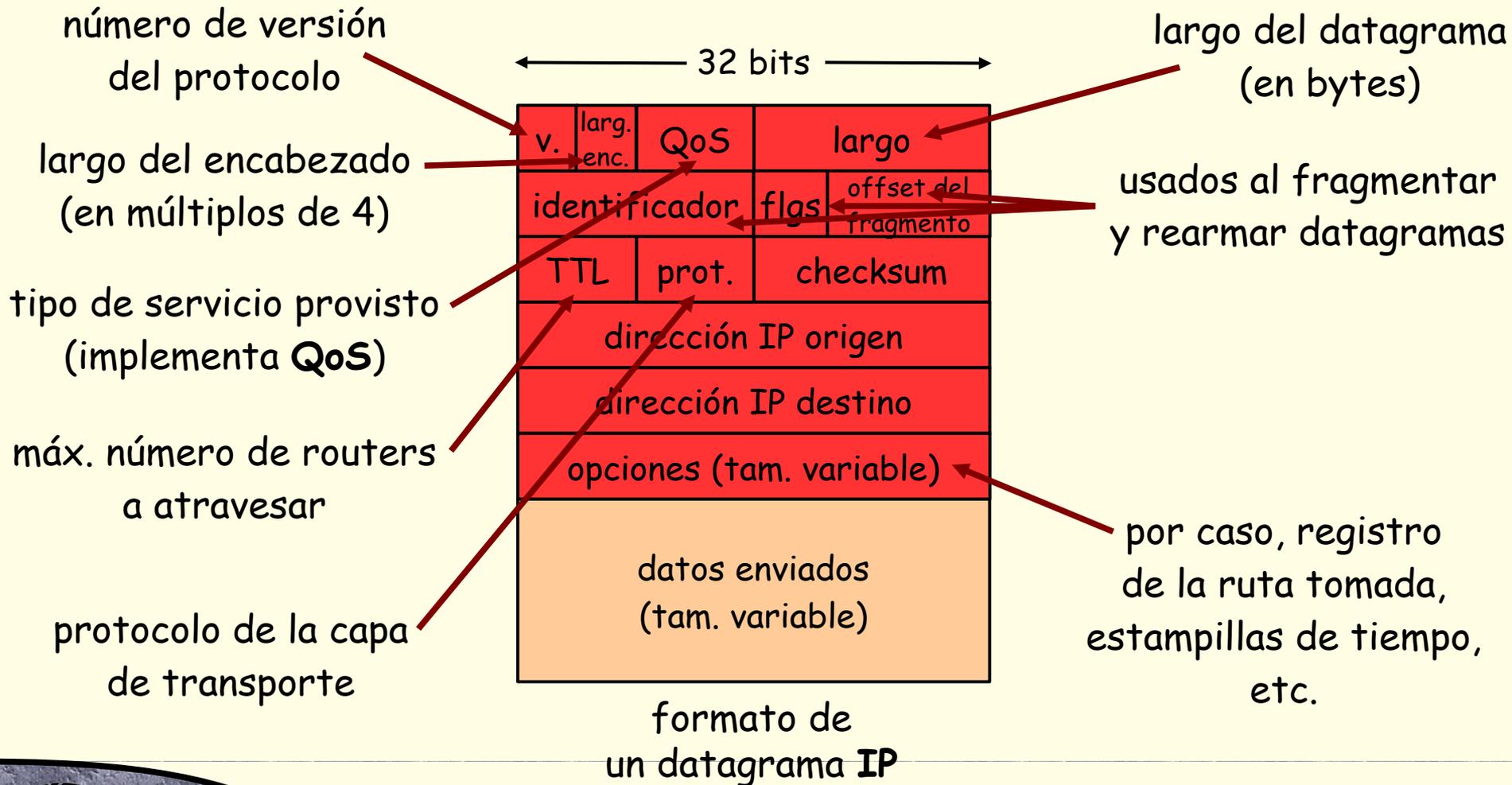
- Modelos de servicios de la capa de red
- Estructura interna de un router
- El protocolo **IP**
- **IPv4** vs. **IPv6**
- Protocolos de ruteo
- Ruteo jerárquico
- Ruteo en internet
- Multicast



Capa de red de internet



Formato de los datagramas



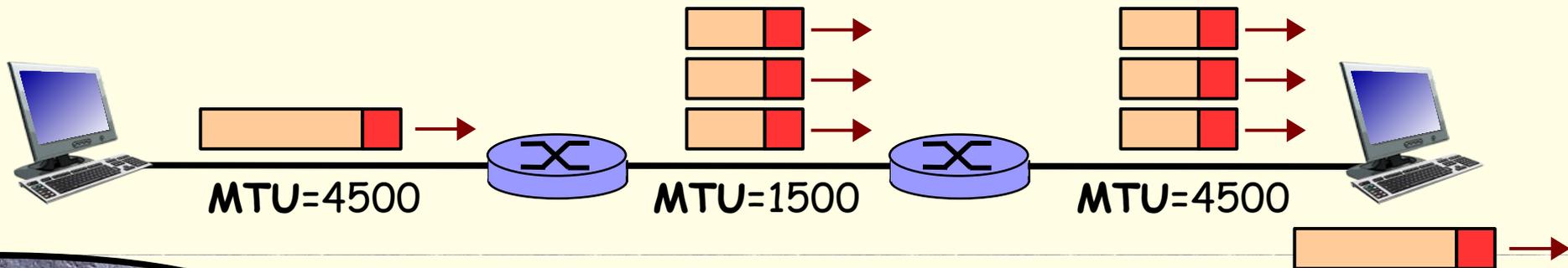
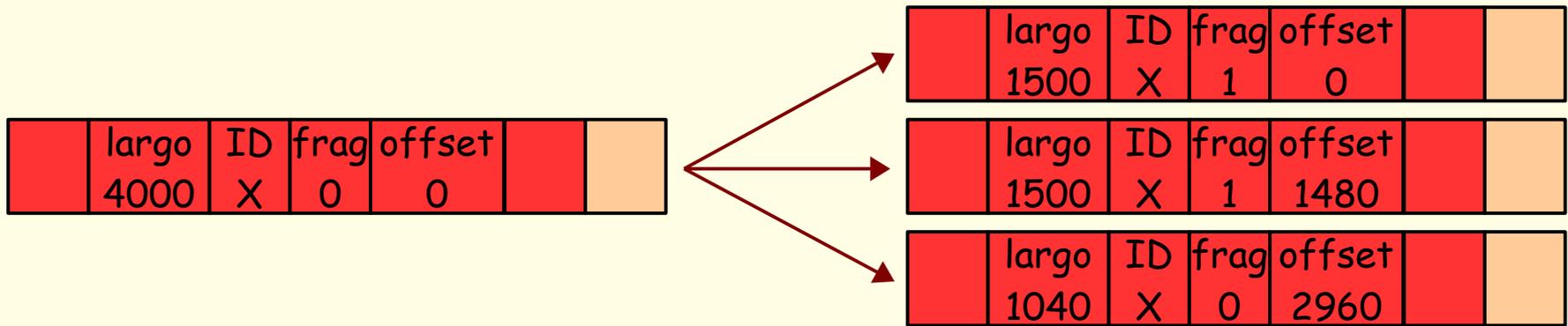
Fragmentación de datagramas

- La capa de enlace determina el máximo tamaño de datagrama que cabe en una trama
- Cada enlace tiene asociado un dado valor de **MTU** (Maximum Transmission Unit)
- Para transmitir un datagrama de gran tamaño **se lo fragmenta en múltiples datagramas** de tamaño menor
 - ➔ Cada fragmento **viaja a través de la red por separado**
 - ➔ El datagrama original **se rearma recién en el destino**



Fragmentación de datagramas

- Supongamos que se desean enviar 4000 bytes sobre un enlace con un **MTU** de 1500 bytes

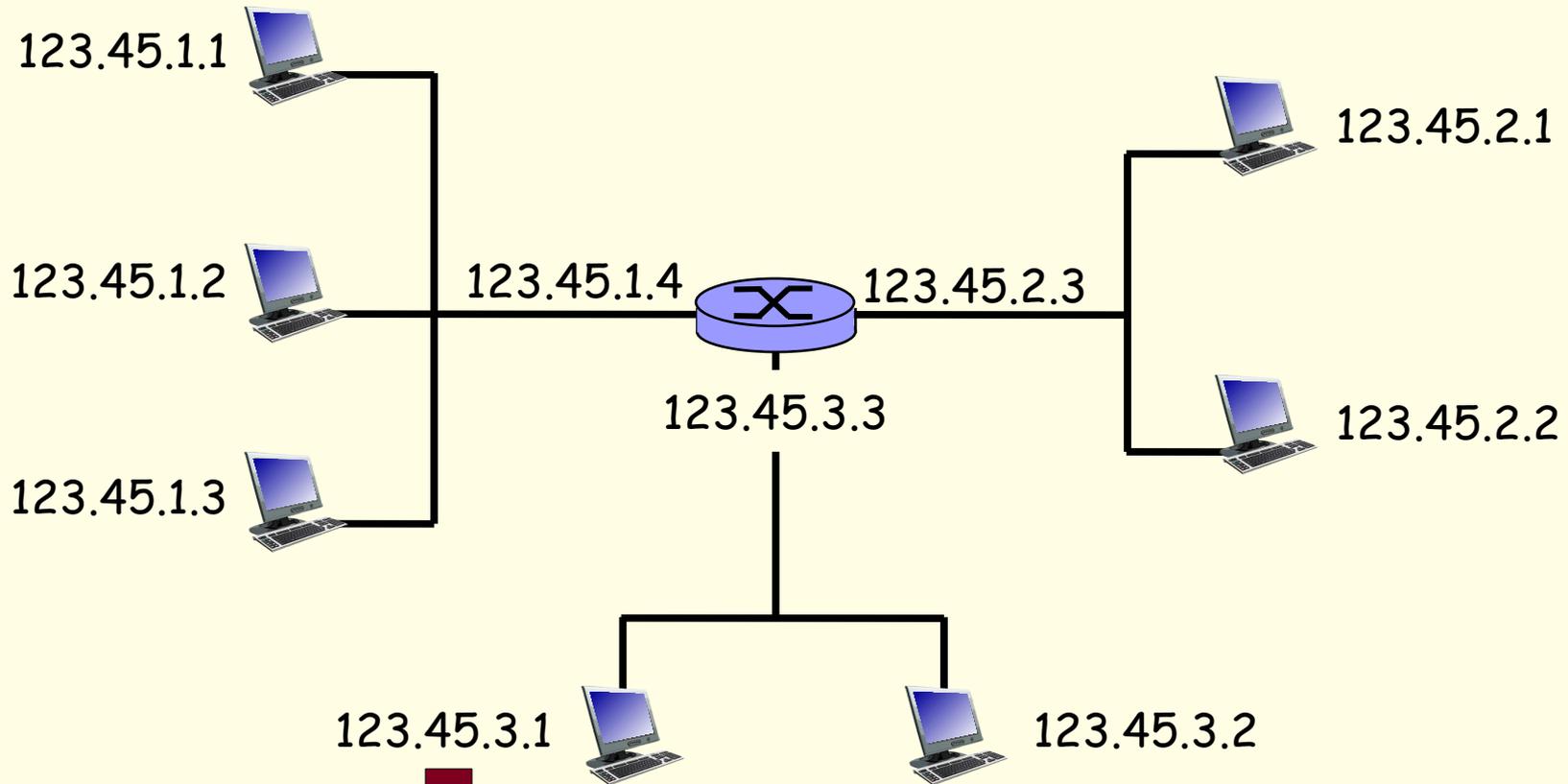


Direccionado IP

- Denominaremos **dirección IP** a la secuencia de 32 bits que identifica unívocamente una interfaz de una computadora o de un router
- Denominaremos **interfaz** a la conexión entre una computadora o router con el enlace físico
 - ➔ Los routers típicamente tienen múltiples interfaces
 - ➔ Las computadoras usualmente tienen una única interfaz pero también pueden tener más de una
 - ➔ Cada interfaz tiene una dirección **IP** propia



Direccionado IP



123.45.3.1

↓

$$123.45.3.1 = \underbrace{01111011}_{123} . \underbrace{00101101}_{45} . \underbrace{00000011}_{3} . \underbrace{00000001}_{1}$$

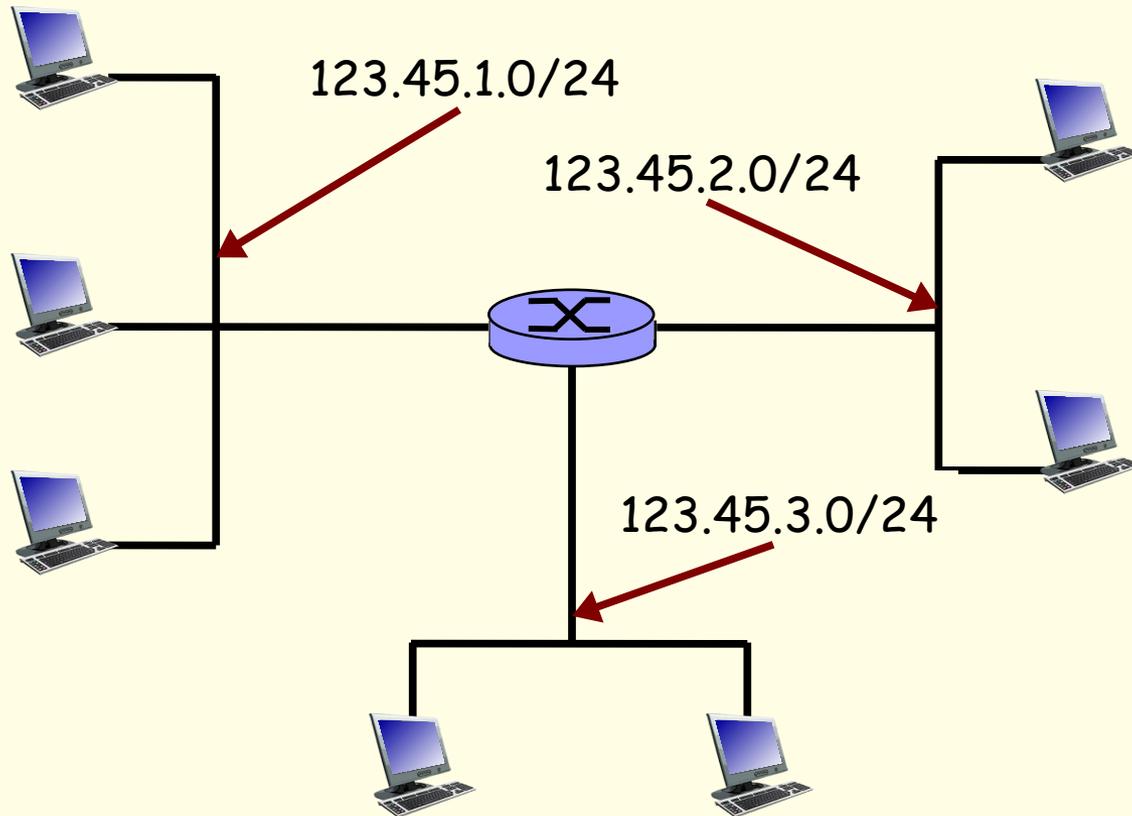


Redes y computadoras

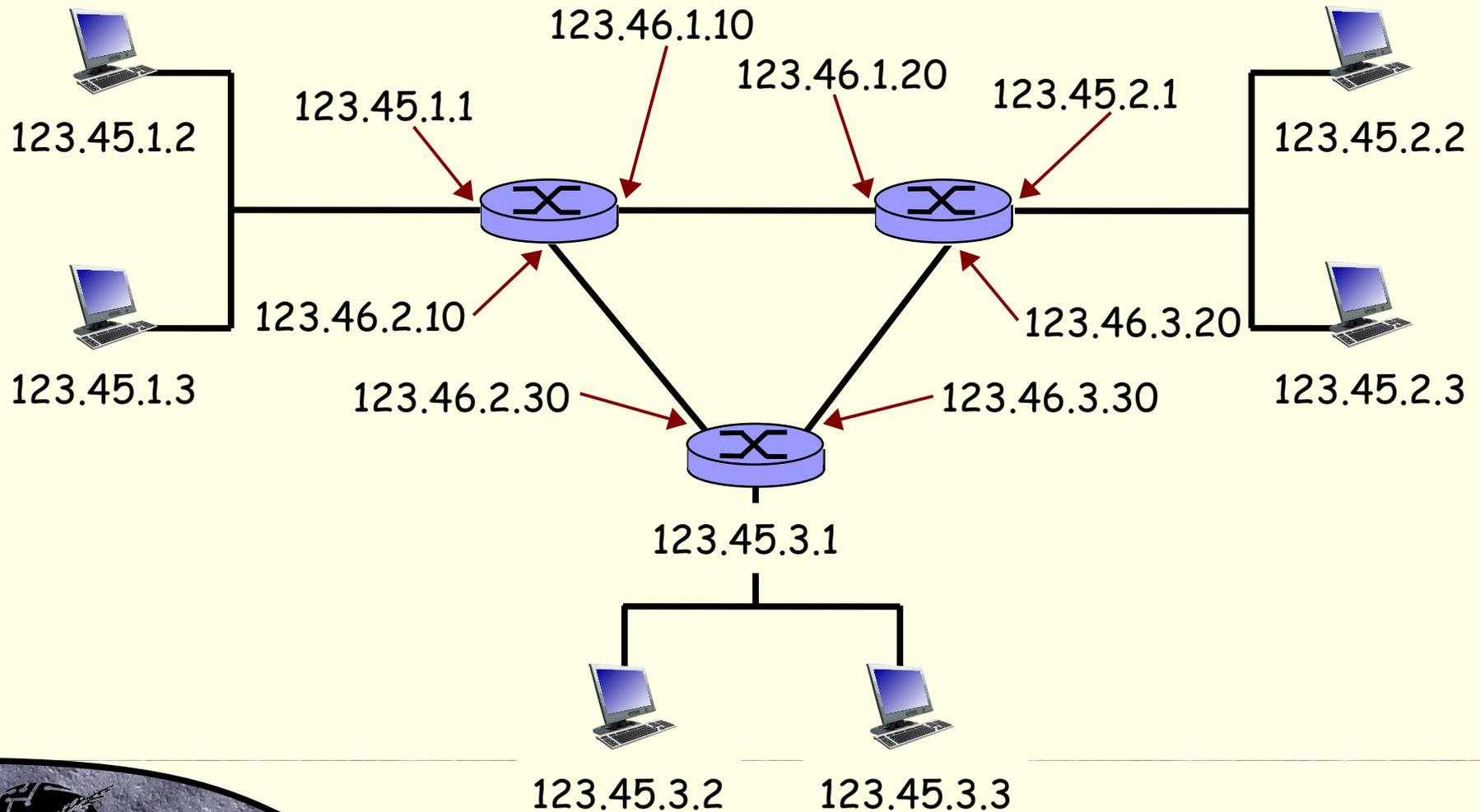
- Toda dirección **IP** se compone de dos partes:
 - Los bits más significativos identifican la red
 - Los bits menos significativos identifican a la computadora
- Bajo este enfoque, ¿en qué consiste una red?
 - Un conjunto de interfaces que **comparten el mismo identificador de red**
 - Las computadoras dentro de una misma red tienen que **poder accederse una a otra de forma directa**



Redes y computadoras

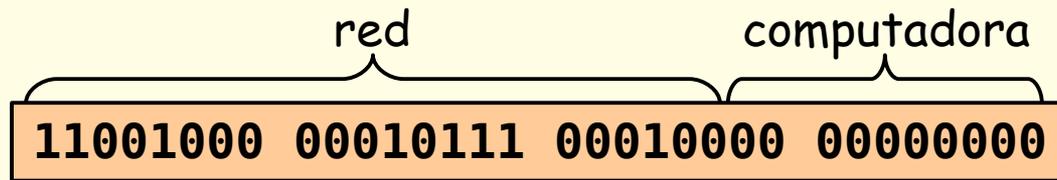


Redes y computadoras



Direccionamiento sin clases

- La clave del direccionamiento sin clases (**CIDR**) consiste en permitir una cantidad arbitraria de bits para identificar la red.



200.23.16.0/22

- La notación adoptada es **a.b.c.d/x**, donde **x** indica la cantidad de bits a ser usados para identificar la red.



IP especiales

- Algunas direcciones **IP** tienen preasignado un significado en particular:

000...000	la computadora actual
111...111	dirección de broadcast en la red local
000...	...000	computadora		una computadora de la red local
red		111...	...111	dirección de broadcast en una red remota
127	dispositivo de loopback



IP privados

- La **RFC 1918** establece tres rangos de direcciones **IP** que **no son ruteables**
 - ➔ De **10.0.0.0** a **10.255.255.255**
 - ➔ De **172.16.0.0** a **172.31.255.255**
 - ➔ De **192.168.0.0** a **192.168.255.255**
- En otras palabras, el tráfico de estas redes jamás atravesará router alguno
 - ➔ Resulta altamente convenientes hacer uso de las mismas dentro de las redes locales



Obtención de una dirección IP

- ¿Cómo hace una computadora para asignar una dirección **IP** a cada una de sus interfaces?
- Pueden estar **determinadas previamente** por el administrador del sistema
 - ➔ Por caso, en Windows accediendo al panel de control o bajo **UNIX** dentro del archivo **/etc/rc.conf**
- También pueden **determinarse dinámicamente** cada vez que se enciende la computadora
 - ➔ Por caso, usando el protocolo **DHCP**



Protocolo DHCP

- El protocolo **DHCP** (Dynamic Host Configuration Protocol) fue introducido para permitir que las computadoras **reciban dinámicamente una dirección de IP** al unirse a una cierta red
 - ➔ Se define formalmente en el **RFC 2131**
 - ➔ Un servidor centralizado lleva el registro de las direcciones **IP** “prestadas” a las computadoras
 - ➔ Permite reutilizar direcciones, ya que quedan reservadas sólo mientras estén en uso
 - ➔ También puede ser usado en computadoras móviles

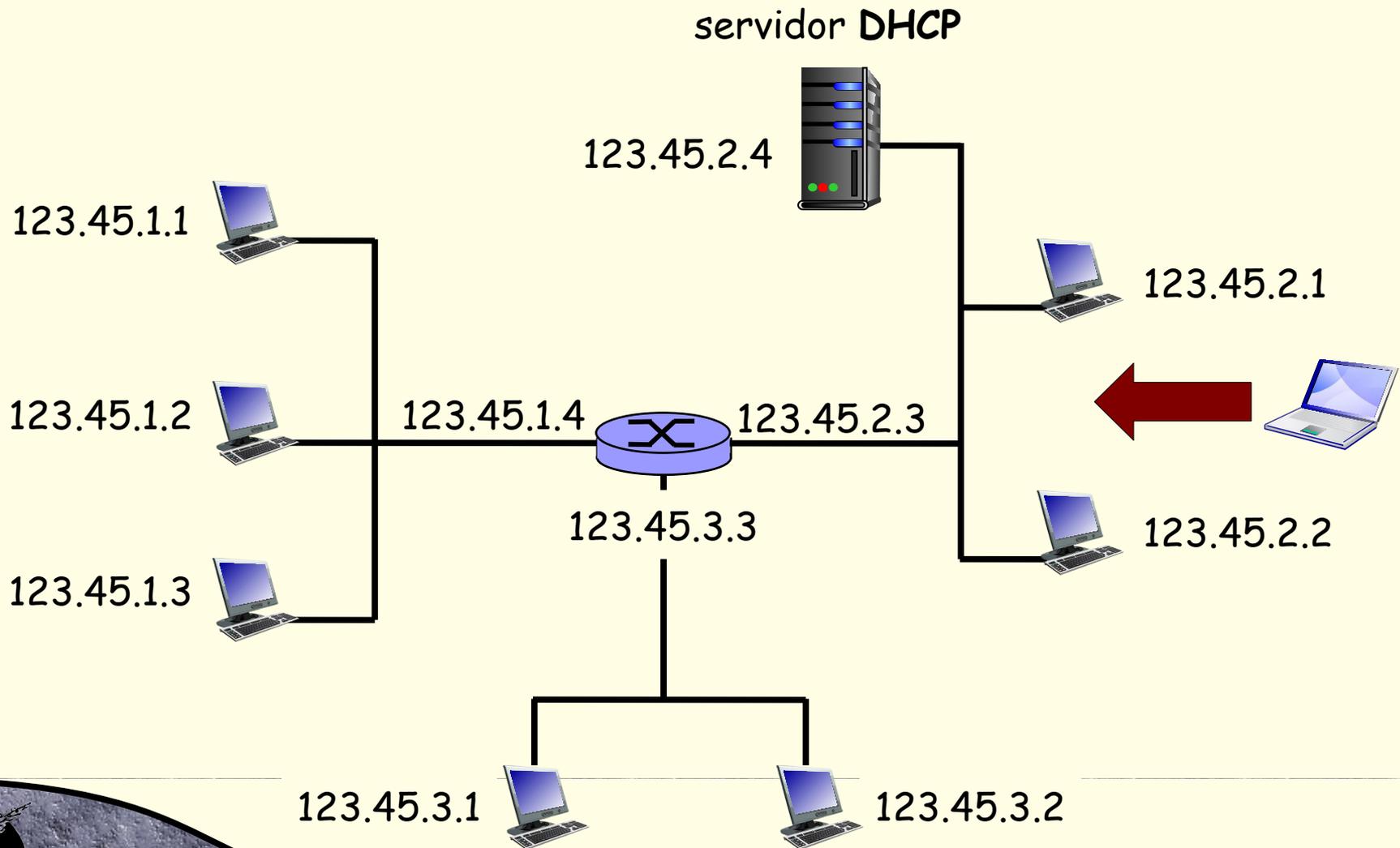


Intercambio de mensajes

- Para obtener una dirección **IP** el cliente **DHCP** debe intercambiar los siguientes mensajes con el servidor:
 - El cliente emite un mensaje “**DHCP discovery**” con destino toda la red (broadcast)
 - El servidor **DHCP** responde al cliente con una oferta dentro de un mensaje “**DHCP offer**”
 - El cliente solicita una dirección **IP** en particular al servidor a través de un mensaje “**DHCP request**”
 - El servidor confirma la delegación de esa dirección **IP** a través de un mensaje “**DHCP ack**”



Intercambio de mensajes

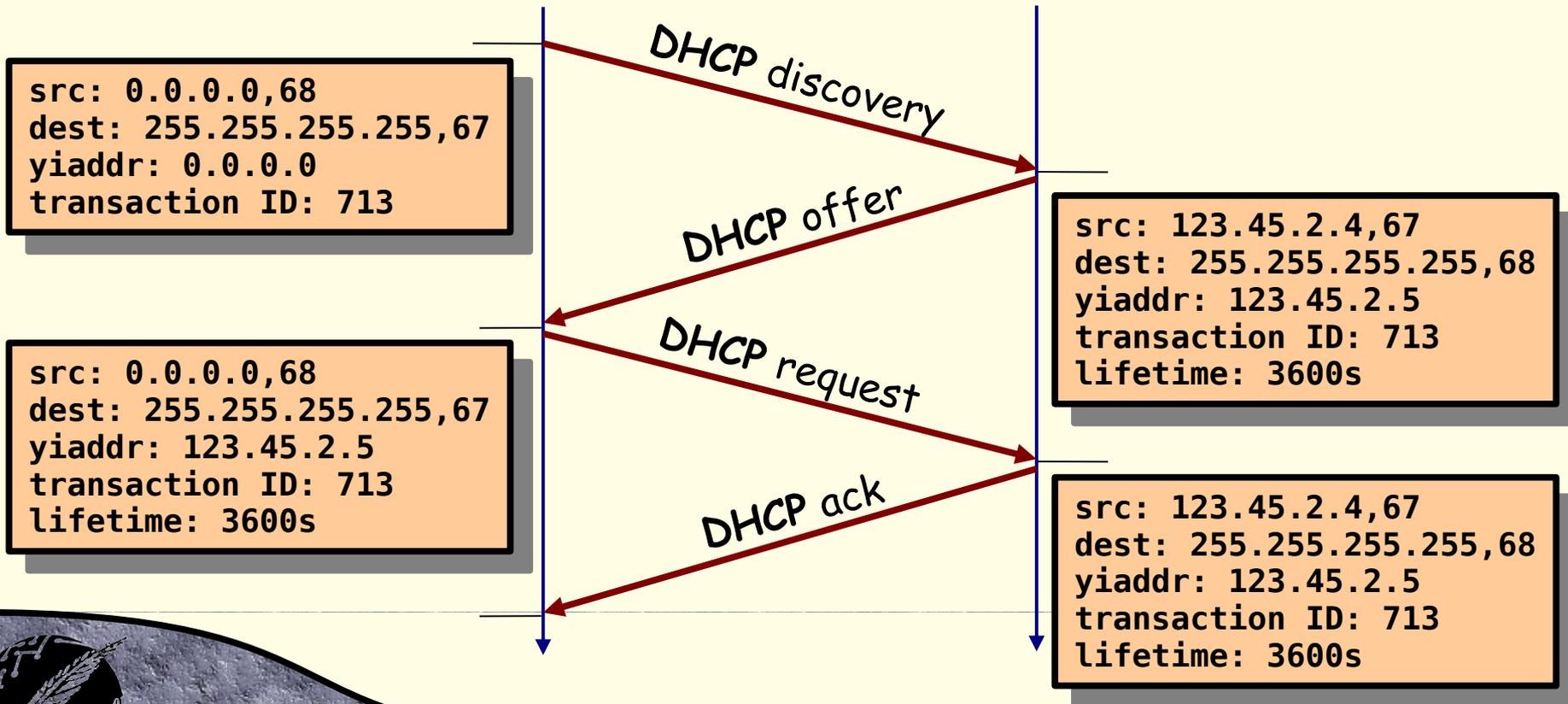


Intercambio de mensajes

cliente DHCP



servidor DHCP



Funcionalidades adicionales

- El protocolo **DHCP** puede automatizar la configuración de otros parámetros de la red, tales como:
 - Asignar **la máscara IP** actualmente en uso en la red local, la cual permite distinguir qué porción de los bits de una dirección representan la red local
 - Configurar **el nombre y la dirección IP del servidor DNS por defecto** en uso en la red local
 - Registrar **la dirección IP del router local** a cargo de encaminar el tráfico hacia otras redes (gateway)



Direcciones de red

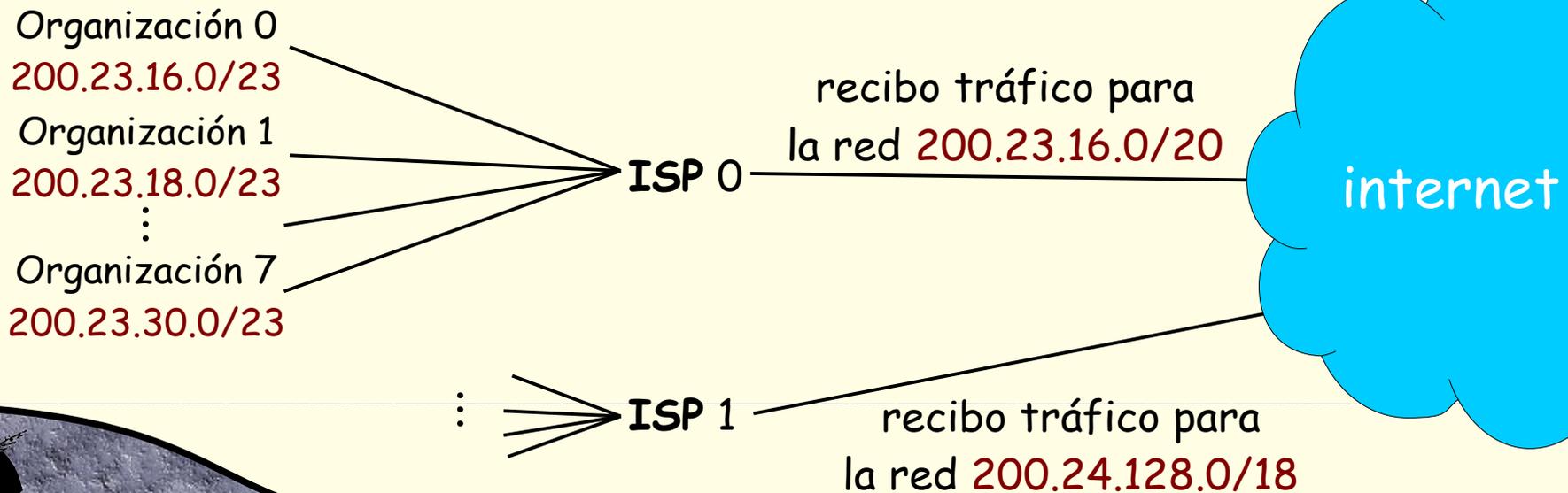
- Determinada la dirección de una computadora dentro de la red local, sólo resta determinar **qué dirección de red usar** para completar la misma
 - La respuesta es simple: haremos uso de una de las direcciones de red asignadas a nuestro proveedor de internet

ISP	11001000	00010111	00010000	00000000	200.23.16.0/20
Organización 0	11001000	00010111	00010000	00000000	200.23.16.0/23
Organización 1	11001000	00010111	00010010	00000000	200.23.18.0/23
...					
Organización 7	11001000	00010111	00011110	00000000	200.23.30.0/23



Agrupado de rutas

- El direccionado jerárquico permite publicitar de manera eficiente a las rutas
- Distintas redes se pueden **agrupar y publicitar como una única entidad**, reduciendo el tamaño de las tablas de forwarding involucradas



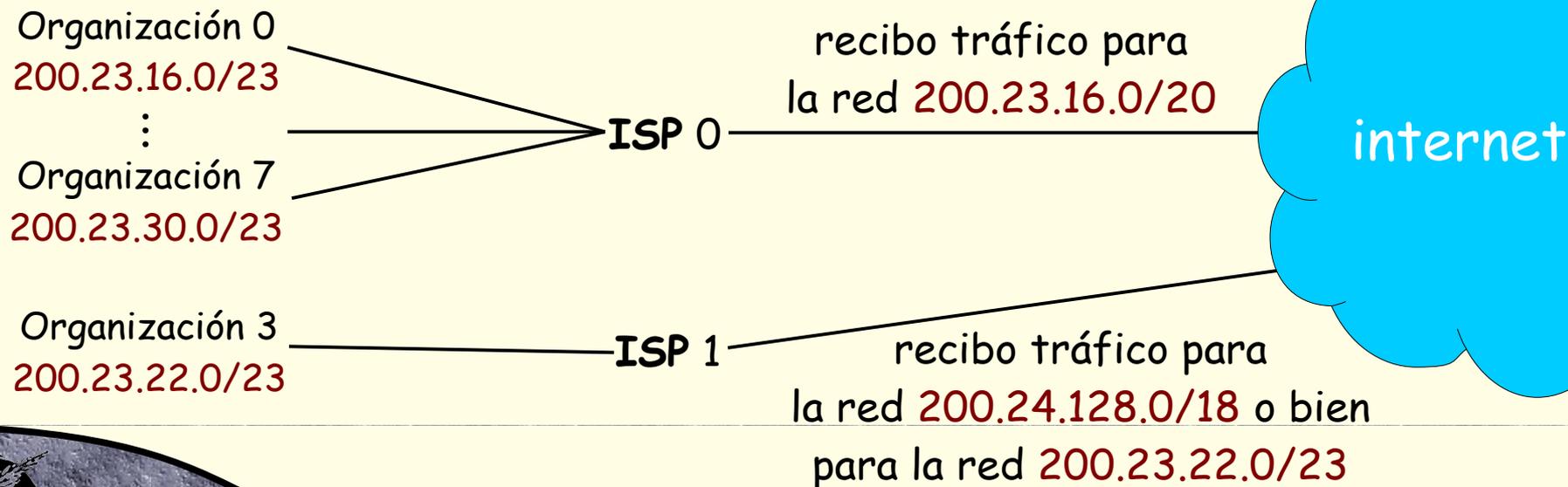
Si, otra "car analogy"

- Como no podía ser de otra forma, es posible que otra analogía usando autos nos clarifique este último concepto
- El agregado de rutas es análogo a lo que hace un **GPS** al resolver el camino que debemos usar por ejemplo para ir una dirección concreta en la Ciudad Autónoma de Buenos Aires
 - ➔ Sin importar la calle y altura que indiquemos, el **GPS** siempre nos va a recomendar tomar por caso la ruta nacional nro. 3 para llegar a destino



Preferencia entre rutas

- ¿Qué sucede si múltiples **ISP** publicitan bloques de direcciones **IP** que se superponen?
- ➔ Se adopta como criterio elegir la ruta que presente **mayor nivel de concordancia a nivel de red**



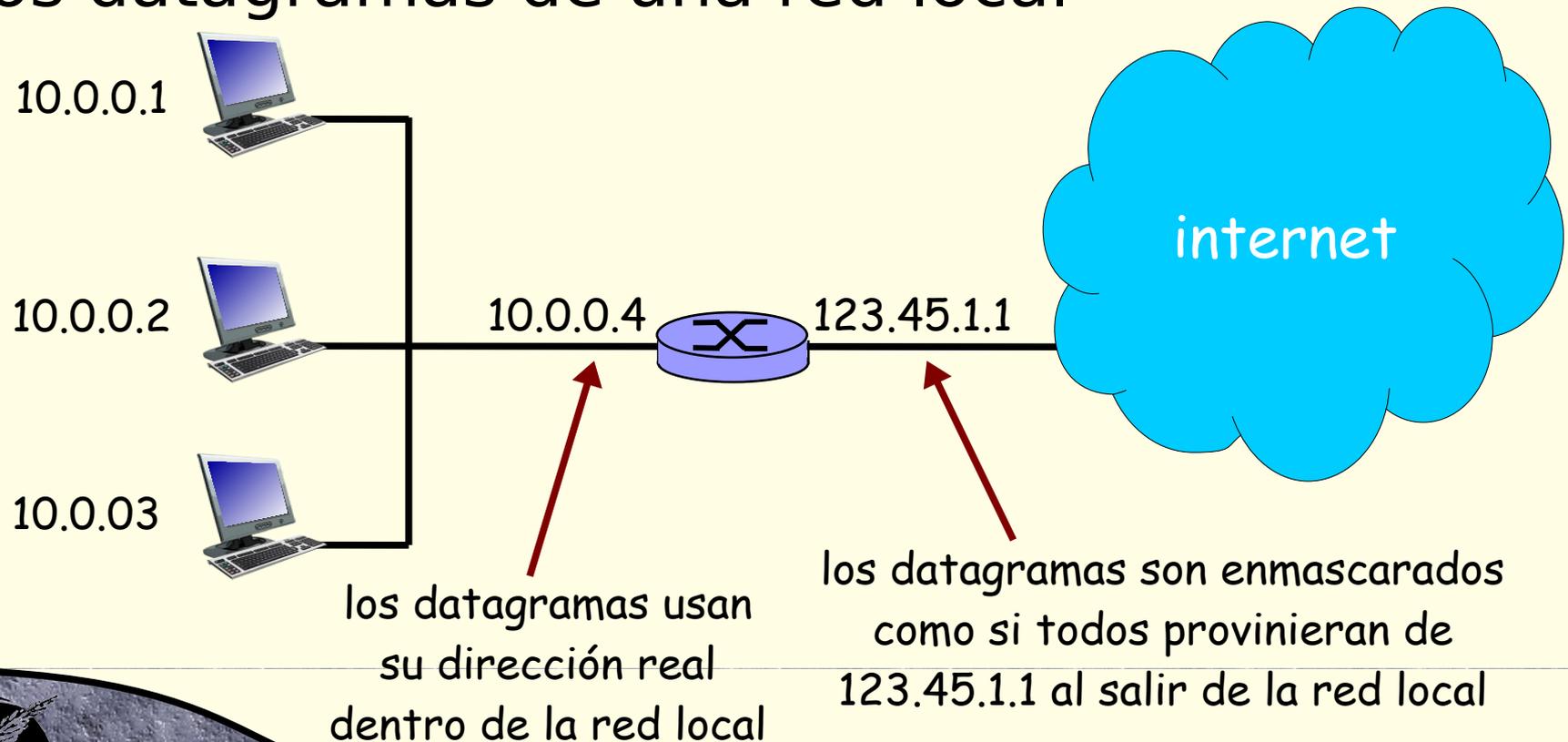
Obtención de un bloque IP

- ¿Cómo hace un proveedor de internet para que le asignen un bloque de direcciones **IP**?
 - ➔ Se tramita ante el **ICANN** (Internet Corporation for Assigned Names and Numbers)
- El **ICANN** tiene a cargo varias tareas:
 - ➔ Asigna los bloques de direcciones
 - ➔ Asigna los nombre de dominios
 - ➔ Administra los servidores raíz **DNS**
 - ➔ Resuelve las disputas relativas a los dominios



Network Address Translation

- La técnica **NAT** (Network Address Translation) consiste en enmascarar el origen real de los datagramas de una red local



Motivación

- Todos los usuarios locales comparten la misma dirección **IP** al ser vistos desde afuera
 - ➔ No es necesario comprar un rango de dirección de nuestro proveedor, con una sola dirección basta
 - ➔ Podemos modificar las direcciones locales sin tener que notificar a nadie por fuera de la organización
 - ➔ Podemos cambiar de proveedor de internet sin tener que modificar la configuración de los dispositivos
 - ➔ Los dispositivos en la red local no son direccionables desde afuera, mejorando la seguridad de la red



Implementación

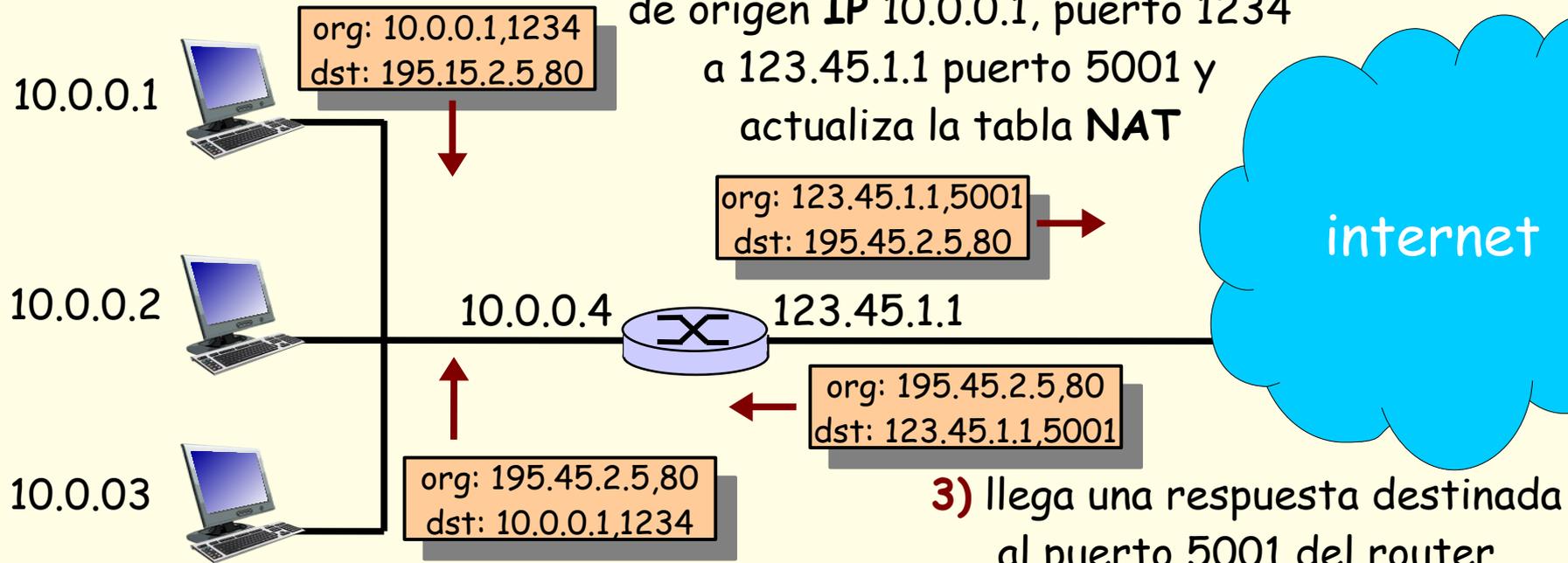
- Para implementar la técnica **NAT**, se debe:
 - ➔ En los datagramas salientes: reemplazar el **IP** de origen y el puerto de origen por el **IP** del router y un número de puerto no en uso
 - ➔ Los clientes o servidores remotos van a contestar usando el **IP** del router y al nuevo puerto usado (se debe recordar cada mapeo entre direcciones **IP** y puertos locales con el puerto usado en el router)
 - ➔ En los datagramas entrantes: reemplazar el par (**IP** del router, puerto del router) por el correspondiente (**IP** local, puerto original), consultando la tabla **NAT**



Implementación

1) la computadora 10.0.0.1
manda un datagrama
destinado a 195.15.2.5

2) el router cambia la dirección
de origen **IP** 10.0.0.1, puerto 1234
a 123.45.1.1 puerto 5001 y
actualiza la tabla **NAT**



3) llega una respuesta destinada
al puerto 5001 del router

4) el router consulta su tabla **NAT**
y cambia la dirección de destino
123.45.1.1, puerto 5001
a los valores originales

Análisis

- Recordemos que en **TCP** y **UDP** se dispone de 16 bits para codificar números de puertos
 - Esto implica que **NAT** puede soportar más de **60000** conexiones en simultáneo por cada **IP** público
- No obstante, **NAT** resulta controversial:
 - Los routers **no deberían involucrarse con la capa de transporte**
 - **Las aplicaciones deben modificarse** para tener en cuenta a esta técnica
 - La falta de direcciones **IP** se debe resolver con **IPv6**



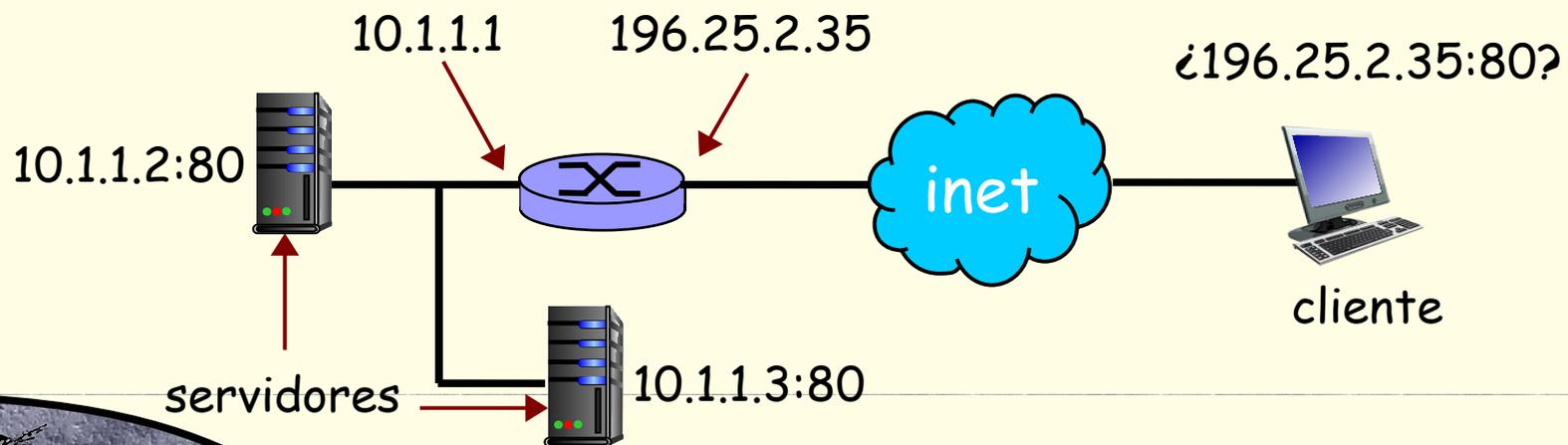
Atravesando el NAT

- El principal inconveniente al hacer uso de la técnica de **NAT** se torna evidente al intentar acceder desde afuera de la organización a alguna de las computadoras de la red local
 - ➔ Por caso, ¿cómo hacen los routers para resolver el enrutado hacia el **IP** privado de la red local?
 - ➔ Una posibilidad es usar el **IP** público, configurando en el router **NAT** que los paquetes recibidos en un determinado puerto sean acercados a ese **IP** privado
 - ➔ Esta técnica se denomina **port forwarding**



Solución estática

- La alternativa más evidente consiste en **configurar a mano** cada uno de los forwardings de puertos que sean requeridos
- ➔ Esta tarea no es trivial, requiere un conocimiento de bajo nivel y una adecuada organización en la preasignación de los puertos disponibles



Solución dinámica

- Otra alternativa consiste en que el router **NAT** implemente el protocolo **IGD** (Internet Gateway Device), para permitir la **configuración por demanda** de los distintos port forwardings
 - ➔ El protocolo **IGD** forma parte del parte del estándar **UPnP** (Univesal Plug and Play)
 - ➔ Permite que una aplicación tome conocimiento de cuál es el **IP** público del router **NAT**
 - ➔ Si una aplicación necesita ser accedida desde fuera de la red local, puede solicitar al router **NAT** que configure dinámicamente un cierto port forwarding



Solución tercerizada

- Existe una tercer alternativa a la hora de atravesar un router **NAT**, la cual consiste en **hacer uso de un intermediario**
 - ➔ El intermediario debe tener acceso directo a la red (es decir, debe no estar detrás de otro **NAT**)
 - ➔ En primer lugar, el servidor detrás del **NAT** se conecta al intermediario (recordemos que la técnica de **NAT** funciona correctamente con las conexiones salientes)
 - ➔ Luego, cuando el cliente desee contactar al servidor se conecta directamente al intermediario y éste se encarga de que ambos se puedan comunicar entre sí



Enrutado de datagramas

- A continuación ensayaremos un primer intento de integrar lo hasta aquí abordado
- Recordemos que los datagramas **IP** han de atravesar la red sin sufrir grandes cambios
 - Esto obedece a que los routers sólo necesitan inspeccionar la dirección **IP** de origen y de destino



Ejemplo de enrutado

- Se desea enrutar un datagrama **IP** desde la computadora **A** hacia la computadora **B**

campos adicionales	123.45.1.1	123.45.1.3	datos
-----------------------	------------	------------	-------

- En primer lugar se busca la entrada asociada a la red del destino **B** en la tabla de forwarding de **A**
- Mirando la tabla de forwarding se determina que **B** está en la misma red que **A**
- Finalmente, la capa de enlace envía el datagrama directamente a **B** dentro de una o varias tramas



Ejemplo de enrutado

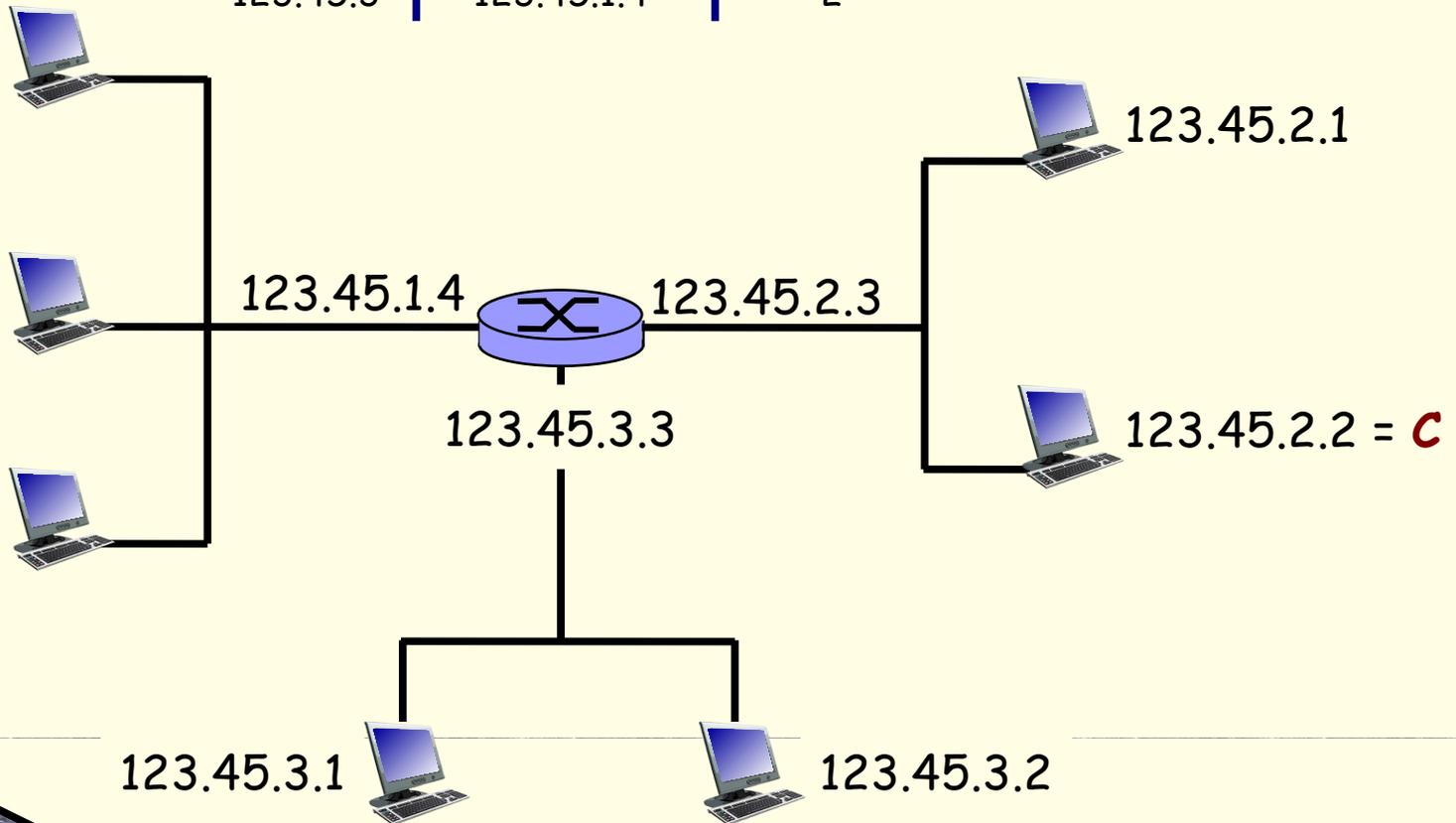
tabla de forwarding
de la computadora **A**

red dest.	próx. router	distancia
123.45.1	-	1
123.45.2	123.45.1.4	2
123.45.3	123.45.1.4	2

A = 123.45.1.1

123.45.1.2

B = 123.45.1.3



Ejemplo de enrutado

- Ahora se desea enrutar un datagrama **IP** desde la computadora **A** hacia la computadora **C**

campos adicionales	123.45.1.1	123.45.2.2	datos
-----------------------	------------	------------	-------

- En primer lugar se busca la entrada asociada a la red del destino **C** en la tabla de forwarding de **A**
- Mirando la tabla de forwarding se determina que **C** no está en la misma red que **A**
- La tabla de forwarding señala que el tráfico para **C** tiene que ser enviado a través del router **123.45.1.4**



Ejemplo de enrutado

- La capa de enlace envía el datagrama al router **123.45.1.4** dentro de una o más tramas
- El datagrama llega al router por una de sus interfaces
- Se consulta la entrada asociada a la red del destino **C** en la tabla de forwarding del router
- Mirando la tabla de forwarding se determina que **C** está en la misma red que la interfaz **123.45.2.3**
- La capa de enlace envía el datagrama directamente a **C** dentro de una o más tramas
- Finalmente, el datagrama llega a la computadora **C**



Ejemplo de enrutado

tabla de forwarding
del router

red dest.	próx. router	distancia	interfaz
123.45.1	-	1	123.45.1.4
123.45.2	-	1	123.45.2.3
123.45.3	-	1	123.45.3.3

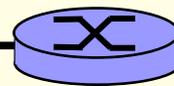
A = 123.45.1.1

123.45.1.2

B = 123.45.1.3



123.45.1.4



123.45.2.3

123.45.3.3



123.45.2.1



123.45.2.2 = **C**

123.45.3.1



123.45.3.2



¿Preguntas?

