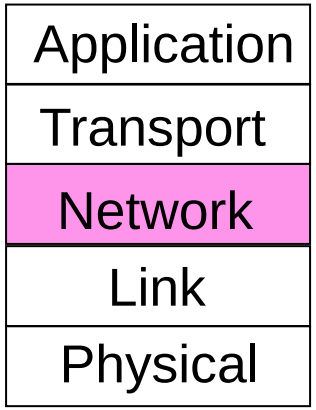
Capítulo 4

La Capa de Red
Complementos de IP y lecturas complementarias



El protocolo IP

- El campo **tiempo de vida** es un contador usado para limitar el tiempo de vida de los paquetes.
  - Debe ser decrementado en cada salto y se supone que va a ser decrementado varias veces cuando se encola por un largo tiempo en el enrutador.
  - En la práctica solo cuenta saltos. Cuando llega a 0, el paquete es descartado y un paquete de advertencia es enviado de regreso al host de origen.
  - Esta facilidad previene que un datagrama esté dando vueltas para siempre, cosa que puede pasar si las tablas de enrutamiento alguna vez están dañadas.

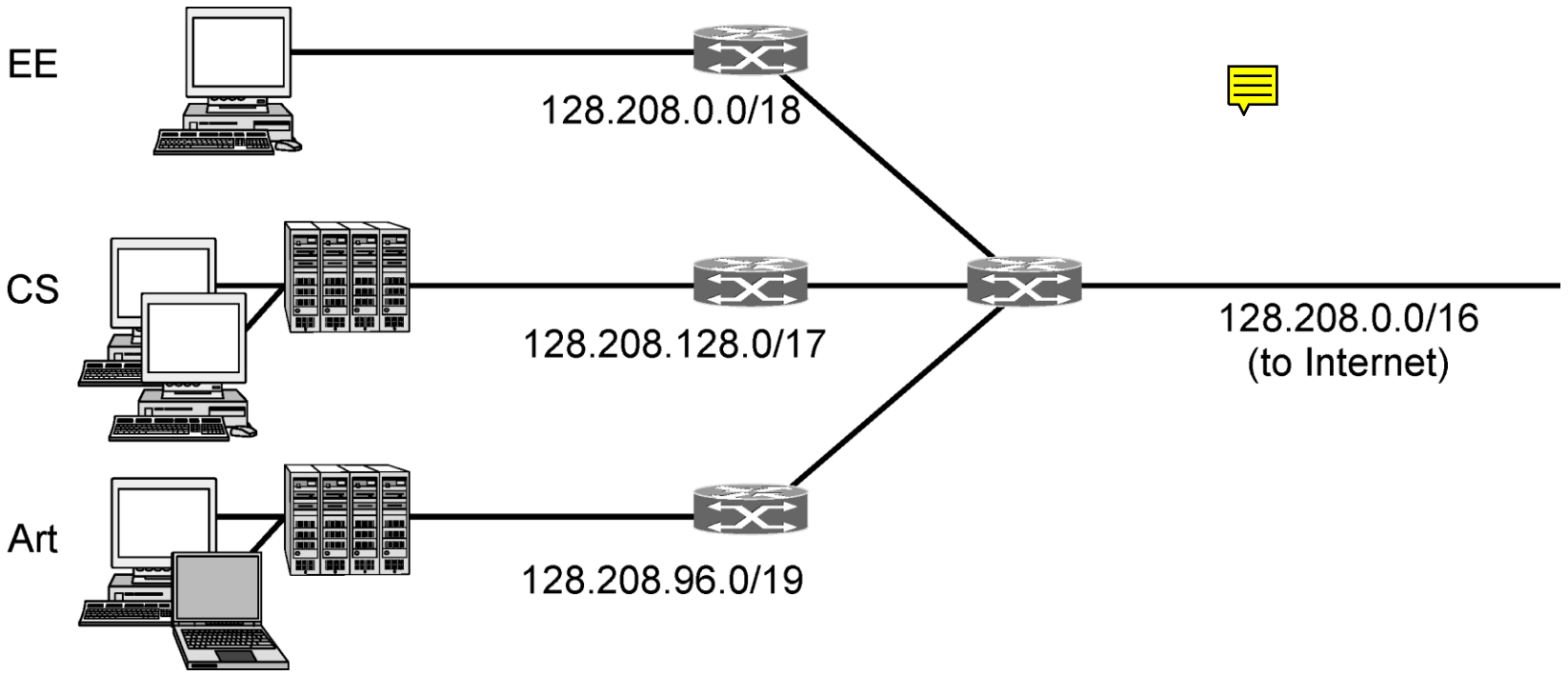
Subredes

- **Uso de subredes**: permitir una red que sea dividida en varias partes para uso interno pero que todavía actúe como una red simple para el mundo externo.
  - Cada subred puede ser una LAN que tiene un **enrutador**.
  - Los enrutadores de una subred conectados a un **enrutador principal**.
  - Fuera de la red, una subred no es visible.

Subredes

- Una subred típica de un campus universitario podría lucir como en la Fig. 57 con un enrutador principal conectado a un ISP o a una red regional, y numerosas Ethernet dispersas en diferentes departamentos.
  - Cada una de las Ethernet tiene su propio enrutador conectado al enrutador principal, posiblemente mediante una LAN de red dorsal.
- En la literatura sobre internet a estas partes de la red (en el ejemplo Ethernets) se les llama **subredes**.

Subredes



Splitting an IP prefix into separate networks with subnetting.

Subredes

- **Problema**: Cuando un paquete entra en el enrutador principal, ¿cómo sabe a cuál subred pasarlo?
- **Solución 1**: tener una tabla en el enrutador principal (con tantas entradas como el tamaño de su red) que indique cuál enrutador usar para cada host.
  - **Evaluación**: se requeriría una tabla muy grande en el enrutador principal y mucho mantenimiento manual conforme se agregan, movieran o eliminaran hosts.
- **Solución 2**: algunos bits se eliminan del N° de host para crear un número de subred
  - P.ej. si la universidad tiene 35 departamentos, se usa 6 bits para el número de subred y 10 bits para el número de host; lo que permite hasta 64 Ethernets, cada una con a lo más 1022 hosts.

Subredes

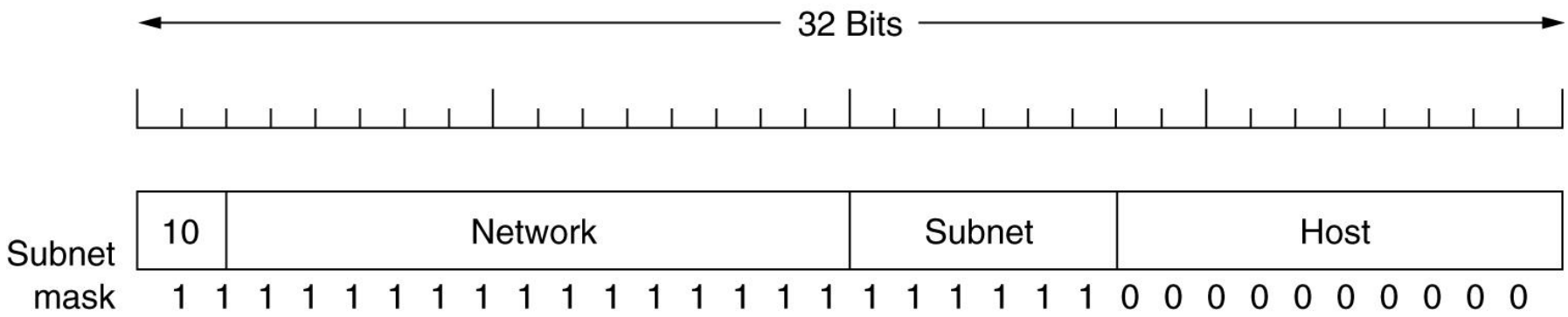


Fig. 58: Una red sub-dividida en 64 subredes.

Subredes

- **Problema**: ¿Cómo expresar subredes?
- **Solución**: el enrutador principal usa una **máscara de subred** que indique la división entre el número de red + número de subred y el host, como se ve en la Fig. 58.
  - Las máscaras de subred también se pueden escribir en notación decimal con puntos, o agregando a la dirección IP una diagonal seguida del número de bits usado para los números de red y subred.
- Para el ejemplo de la Fig. 58 la máscara de subred puede escribirse como:
  1111 1111. 1111 1111. 1111 1100. 0000 0000

  255. 255 . 252 . 0.
  - Esta máscara permite 1024 hosts
  - Una notación alternativa es /22 para indicar que la máscara de subred tiene una longitud de 22 bits.

Subredes

- Fuera de la red, la subred no es visible, por lo que la asignación de una subred nueva no requiere comunicación con el ICANN (Corporación de Internet para la Asignación de Nombres y Números) ni la modificación de bases de datos externas.

Subredes

- **Ejemplo**: Red de universidad
  - Computer Science: 10000000 11010000 1|xxxxxxx xxxxxxxx
  - Electrical Eng.: 10000000 11010000 00|xxxxxx xxxxxxxx
  - Art: 10000000 11010000 011|xxxxx xxxxxxxx
- La barra vertical (|) muestra el límite entre el N° de subred y el N° de host. A su izquierda se encuentra el número de subred, a su derecha el número de host.
  - Computer Science: Comenzando en 128.208.128.0
  - Electrical Eng. : Comenzando en 128.208.0.0
  - Art: Comenzando en 128.208.96.0

Subredes

- **¿Cómo serían las tablas de enrutamiento para el enrutador principal (i. e. cuando hay subredes)?**
- Se tienen entradas con forma de
  - (dirección IP inicio subred, máscara).
  - Cuando un paquete llega al enrutador principal, el enrutador hace un AND booleano de la dirección de destino con la máscara de subred para deshacerse del número de host y buscar la dirección resultante en sus tablas (hay que ver si coincide con la dirección de inicio de subred o prefijo).

Subredes

- **Ejemplo**: un paquete dirigido a 128.208.2.151 que llega al enrutador principal.
  - Para ver si es para el departamento de Computer Science se hace AND de 128.208.2.151 con la máscara de subred 255.255.128.0, para dar la dirección 128.208.0.0.
  - Hay que ver si esto concuerda con la dirección de inicio de subred (prefijo) que es 128.208.128.0. No concuerdan.
  - Para ver si es para el departamento de Electrical Engineering se hace AND de 128.208.2.151 con la máscara de subred 255.255.192.0, para dar la dirección 128.208.0.0.
  - Esto concuerda con el prefijo. Así que el paquete se envía a la línea que va a la red de Electrical Engineering.

Subredes

- **Observación**: El origen de una subred denota el tamaño máximo de hosts que puede albergar.
- Por ejemplo, supongamos que inicia en 130.50.8.0:

  1000 0010 . 0011 0010 . 0000 1000 . 0000 0000

  – Esta red puede crecer hasta 2^11 hosts = 2048. Dicha mascara es:
  – 255 . 255 . 248 . 0

Subredes

- Tenemos la subred que inicia en 130.50.8.0:

  1000 0010 . 0011 0010 . 0000 1000 . 0000 0000
- **Problema**: queremos hacer la red mas grande,
- **Idea**: deberíamos elegir otra máscara,
  - por ejemplo: 255. 255. 128. 0 que albergaría 2^15= 32 K hosts.
  - Esto haría que la red llegue hasta la 130.50.135.255.
  - Pero escribamos la mascara en binario:

    1111 1111. 1111 1111. 1000 0000 . 0000 0000
  - ¡Ningún paquete que se haga AND con esta máscara me da la IP de origen de la subred! (130.50.8.0)
  - **Moraleja**: la cantidad máxima de hosts se da por la cantidad de 0 a la derecha del ultimo 1 en la dirección de origen:

    1000 0010 . 0011 0010 . 0000 1000 . 0000 0000

IPv6

- **Problema:** el espacio de direcciones de 32-bit ya ha sido agotado en varias regiones del mundo (incluyendo Latinoamérica, Europa, Norteamérica).
- **Solución**: Considerar un espacio de direcciones mucho más grande.
- **Problema**: con IPv4 algunos campos del encabezado hacen que el procesamiento de datagramas en los enrutadores lleve tiempo:
  - p.ej: campos para fragmentación, procesamiento de suma de verificación, etc.

IPv6

- **Requisitos:**
  - Que el formato de encabezado ayude a aumentar la velocidad de procesamiento y reenvío
  - Cambios en el encabezado para facilitar la calidad de servicio.
- **¿Por qué hace falta que el procesamiento de encabezados sea más rápido?**
  - Porque las redes cada vez son más rápidas, en cambio la velocidad de los procesadores se está estabilizando.
  - Entonces para compensar hay que agilizar el procesamiento de los datagramas.

IPv6

- **Formato de datagrama IPv6:**
  - **Encabezado de longitud fija** de 40 bytes para procesamiento más rápido de datagramas
  - **Capacidad de direccionamiento expandida**: direcciones de 128 bits.
  - **Etiquetado de flujos**: se etiquetan paquetes que pertenecen a un mismo flujo para los cuales el emisor requiere manejo especial.

IPv6

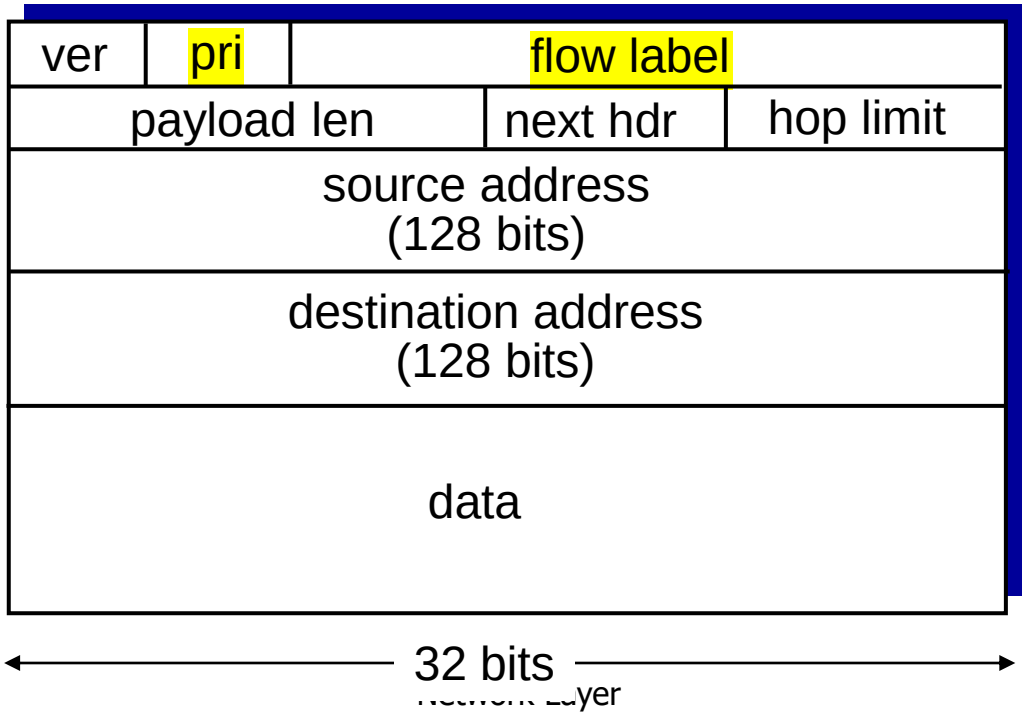
- Ejemplos de flujo o no flujo:
  - P.ej. Transmisión de audio y video pueden ser tratados como un flujo.
  - P.ej. Transferencia de archivos y e-mail pueden no ser tratados como flujos.
  - P.ej. El tráfico de un usuario de alta prioridad puede ser tratado como un flujo.
- **Consecuencia del etiquetado de flujos:**
  - Cuando un paquete con una etiqueta de flujo distinta de cero aparece, los enrutadores pueden ver en tablas internas para ver qué tipo de tratamiento especial requiere.

IPv6

Etiqueta de flujo: (20 b) para identificar datagramas en el mismo "flujo" (El concepto de"flujo"no está bien definido).

Prioridad tiene dos usos:

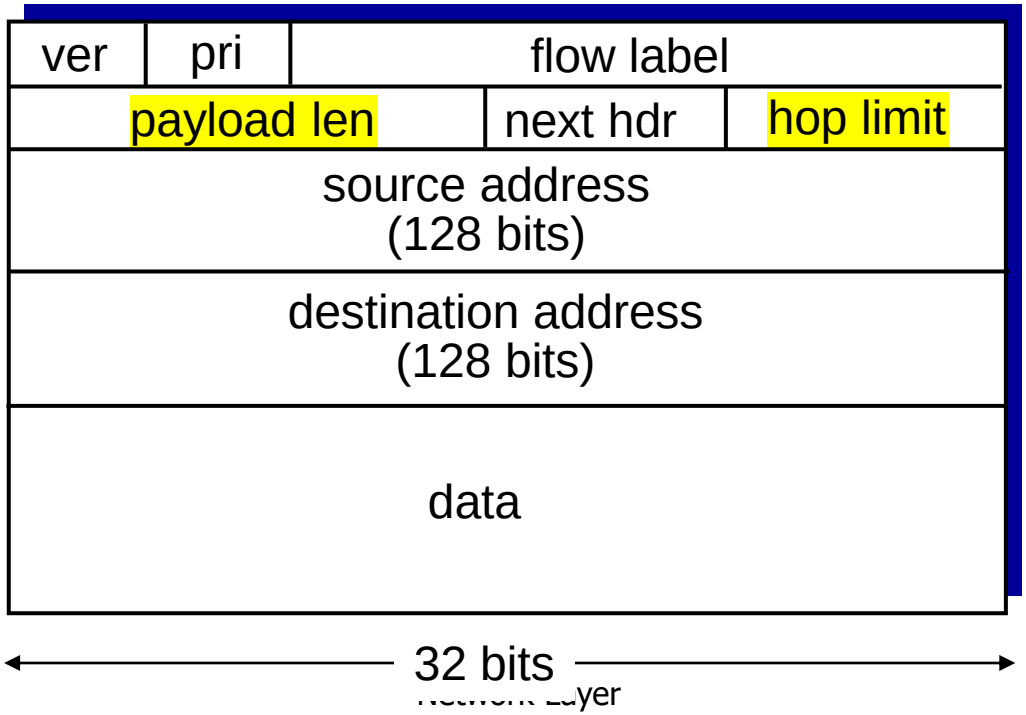
- para dar prioridad a ciertos datagramas dentro de un flujo.
- para dar prioridad a datagramas de ciertas aplicaciones sobre datagramas de otras aplicaciones.



IPv6

Longitud de carga útil: (16 b) número de bytes en el datagrama IPv6 luego del encabezado (de 40 B).

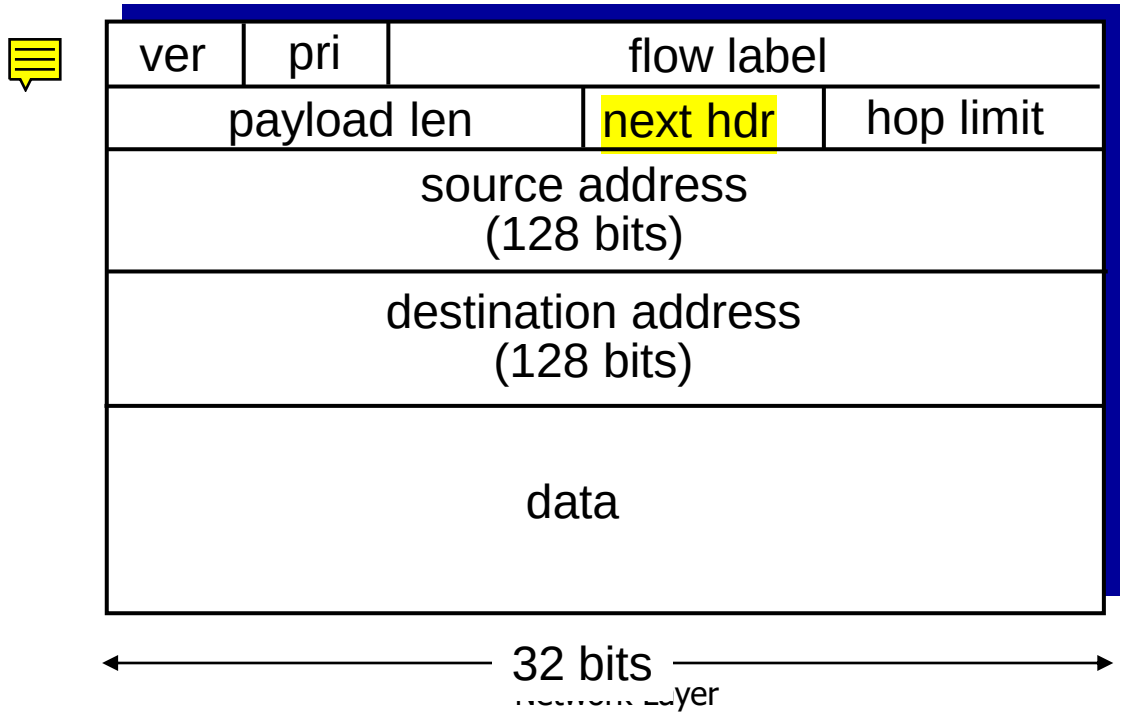
Límite de saltos: (8 bits) el contenido de este campo se decrementa en 1 por cada enrutador que entrega el datagrama. Si el contador alcanza 0, el datagrama se descarta.



IPv6

Próximo encabezado: (8 bits) significa:

- Cuál de los 6 encabezados extensión de opciones actuales le sigue al encabezado.
- Si este encabezado es el último encabezado IP, el campo dice a cuál protocolo de transporte entregar el datagrama.
- Los encabezados de opciones también tienen este campo.



IPv6

- **Direcciones IPv6:**
  - Son escritas como 8 grupos de 4 dígitos hexadecimales.
  - Para separar los grupos se usa “:”.
  - P.ej: 8000:0000:0000:0000:0123:4567:89AB:CDEF
  - **Optimización**:
    - Ceros a la izquierda de grupos pueden ser omitidos
    - Grupos con 16 bits iguales a 0 pueden reemplazarse con dos “:”.
  - P.ej. la dirección anterior: 8000::123:4567:89AB:CDEF

IPv6

- ***Otros cambios en relación a IPv4***
  - No se permite fragmentación ni re-ensamblado en enrutadores intermedios.
    - Esto solo puede hacerse por el origen y el destino.
  - ***Suma de verificación***: removido para reducir el tiempo de procesamiento en cada salto (ya la capa de transporte y de enlace de datos usan suma de verificación).
    - Trabajar con este campo era costoso en IPv4.
  - ***Opciones:*** están permitidas, pero fuera del encabezado, indicado por el campo de próximo encabezado.

IPv6

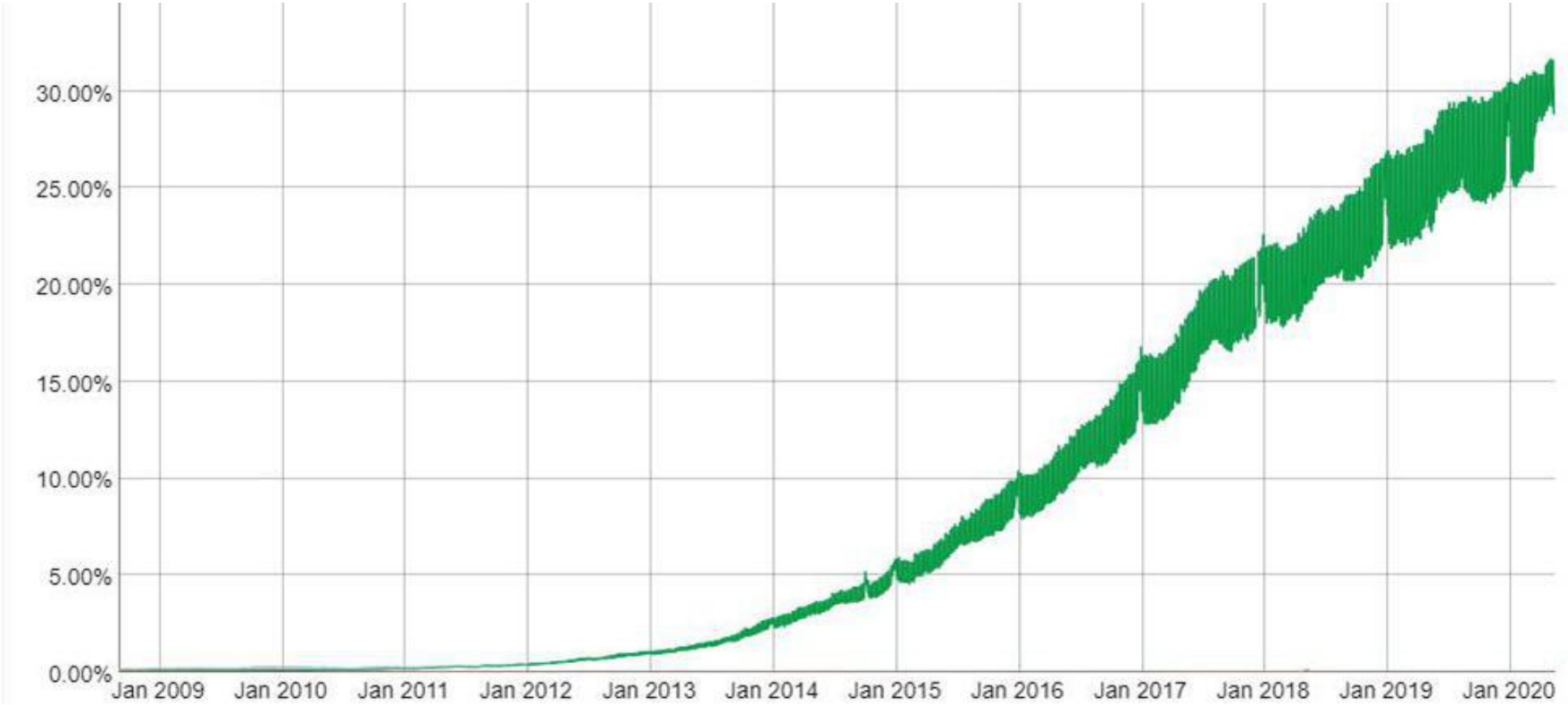
- **Problema: ¿Qué se puede hacer si un datagrama es demasiado grande para pasar por una línea de salida de un enrutador?**
  - Un enrutador descarta paquetes que son demasiado grandes para la línea de salida;
  - y manda al emisor un ***mensaje de paquete demasiado grande***.
  - Luego el emisor puede reenviar los datos usando datagramas IP más chicos.

IPv6

- Los conceptos de prefijo y agregación de prefijos se usan también en IPv6.
  - Las direcciones IPv6 se siguen asignando a interfaces.
- **Adopción de IPv6:**
  - En enero del 2020 el 30 % de los usuarios acceden a Google usando IPv6.
  - En Argentina la adopción de IPv6 es del 11%, en Brasil del 34%, en USA es del 43%.

IPv6

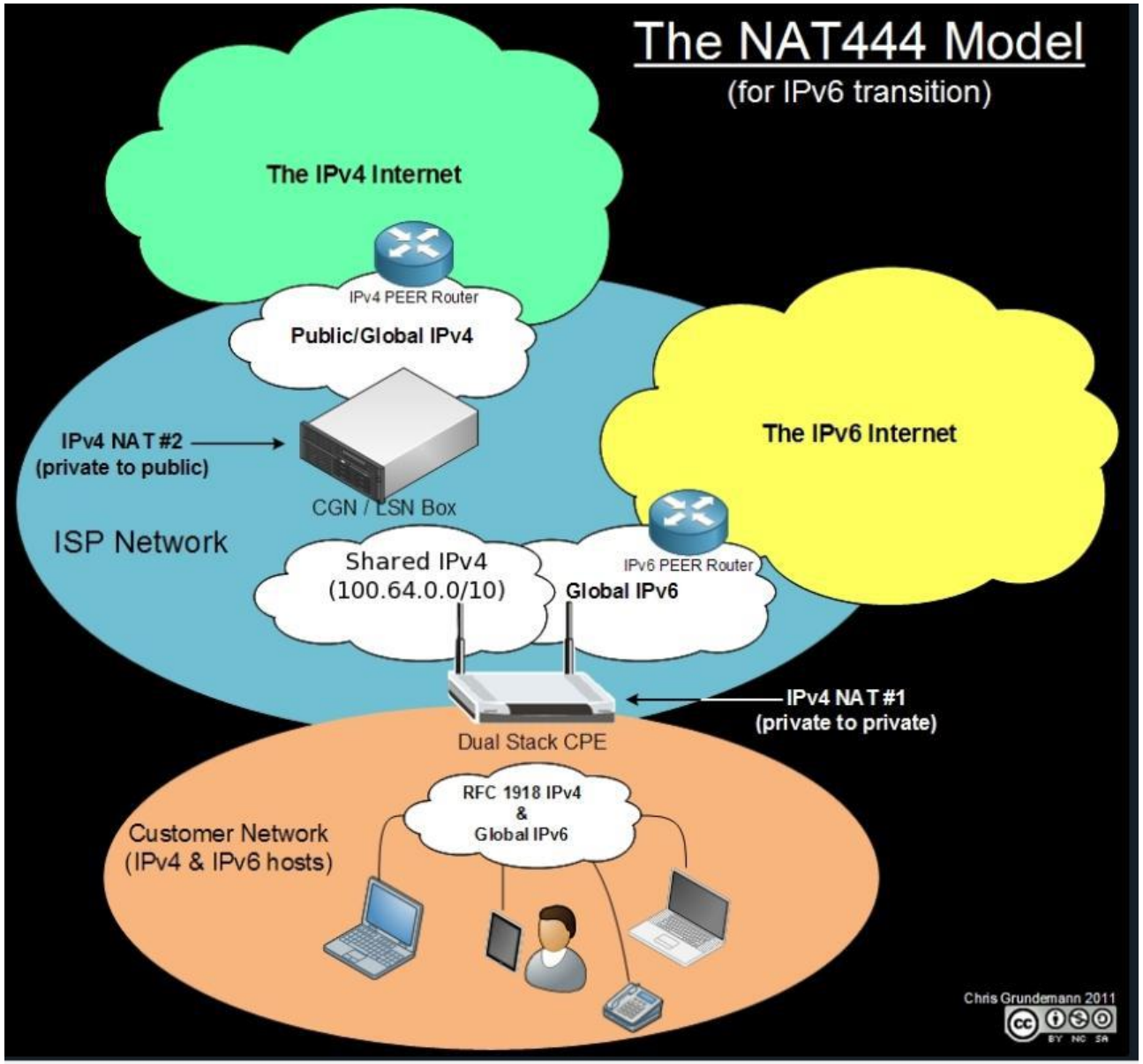
- Según la figura de abajo se está acelerando la adopción de IPv6.



NAT

- **NAT 444**:
  - Los proveedores de servicio de internet (PSI) también pueden tener NAT.
    - Esto hace que las direcciones IPv4 puedan racionarse más aun y durar aun más tiempo.
  - Se llama NAT 444.
  - El espacio de direcciones IP reservado para NAT 444 es 100.64.0.0/10 (o sea alrededor de 4000.000 de IP para ser usadas por la red del PSI)

NAT 444


The NAT444 Model
(for IPv6 transition)
The IPv4 Internet
IPv4 PEER Router
Public/Global IPv4
IPv4 NAT #2
(private to public)
CGN / LSN Box
ISP Network
Shared IPv4
(100.64.0.0/10)
IPv6 PEER Router
Global IPv6
The IPv6 Internet
IPv4 NAT #1
(private to private)
Dual Stack CPE
RFC 1918 IPv4
&
Global IPv6
Customer Network
(IPv4 & IPv6 hosts)
Chris Grundemann 2011

DHCP: Protocolo de Configuración Dinámica de Host

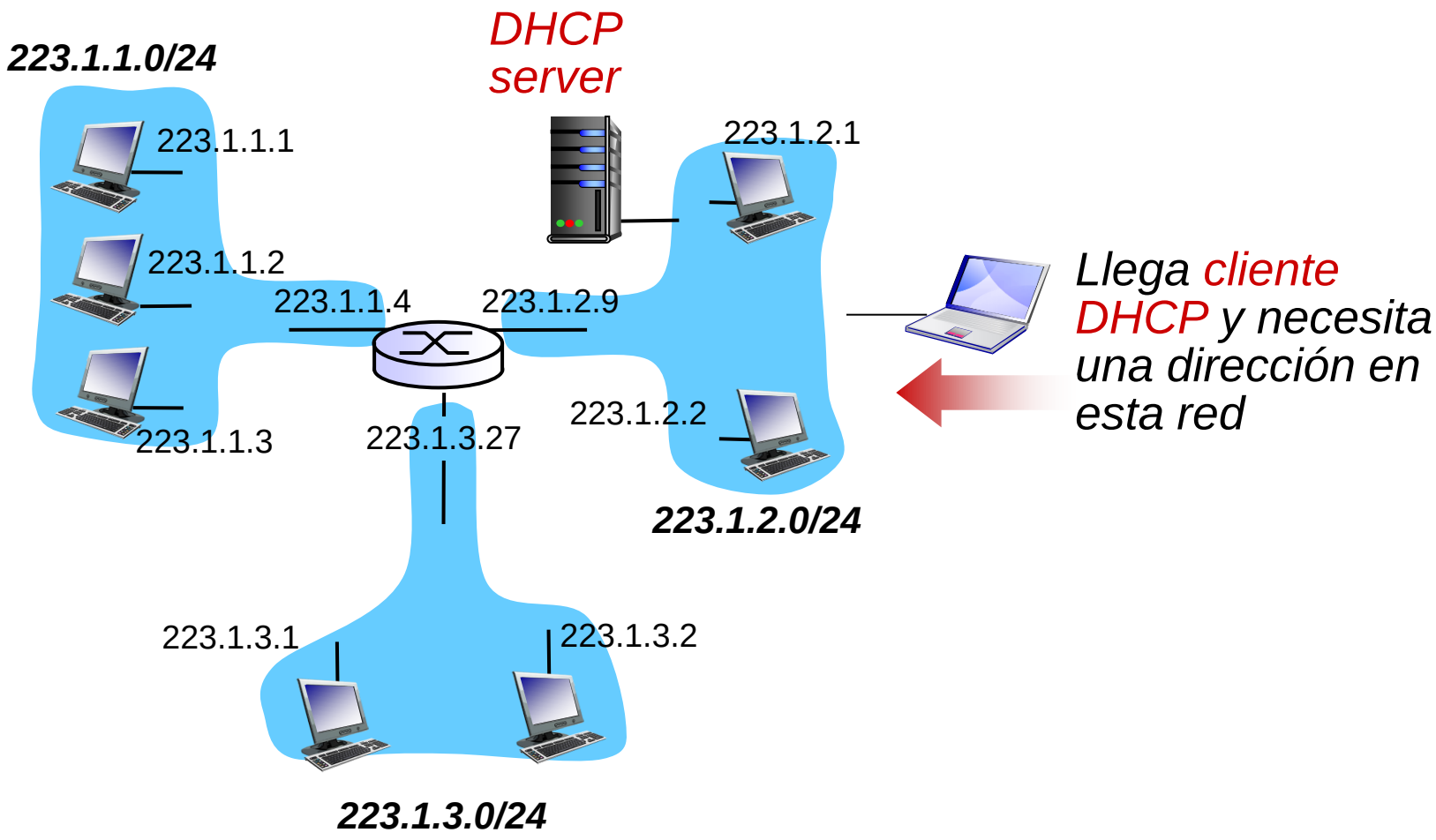
Meta: permitir a los hosts cuando se unen a la red obtener *dinámicamente* su dirección IP a partir de servidor de red

- Un host podrá renovar la dirección IP que usa.
- Permitirá reutilizar las direcciones (solo se sostendrá direcciones mientras está conectado/"prendido")
- Soporte a usuarios móviles que quieren unirse a la red.

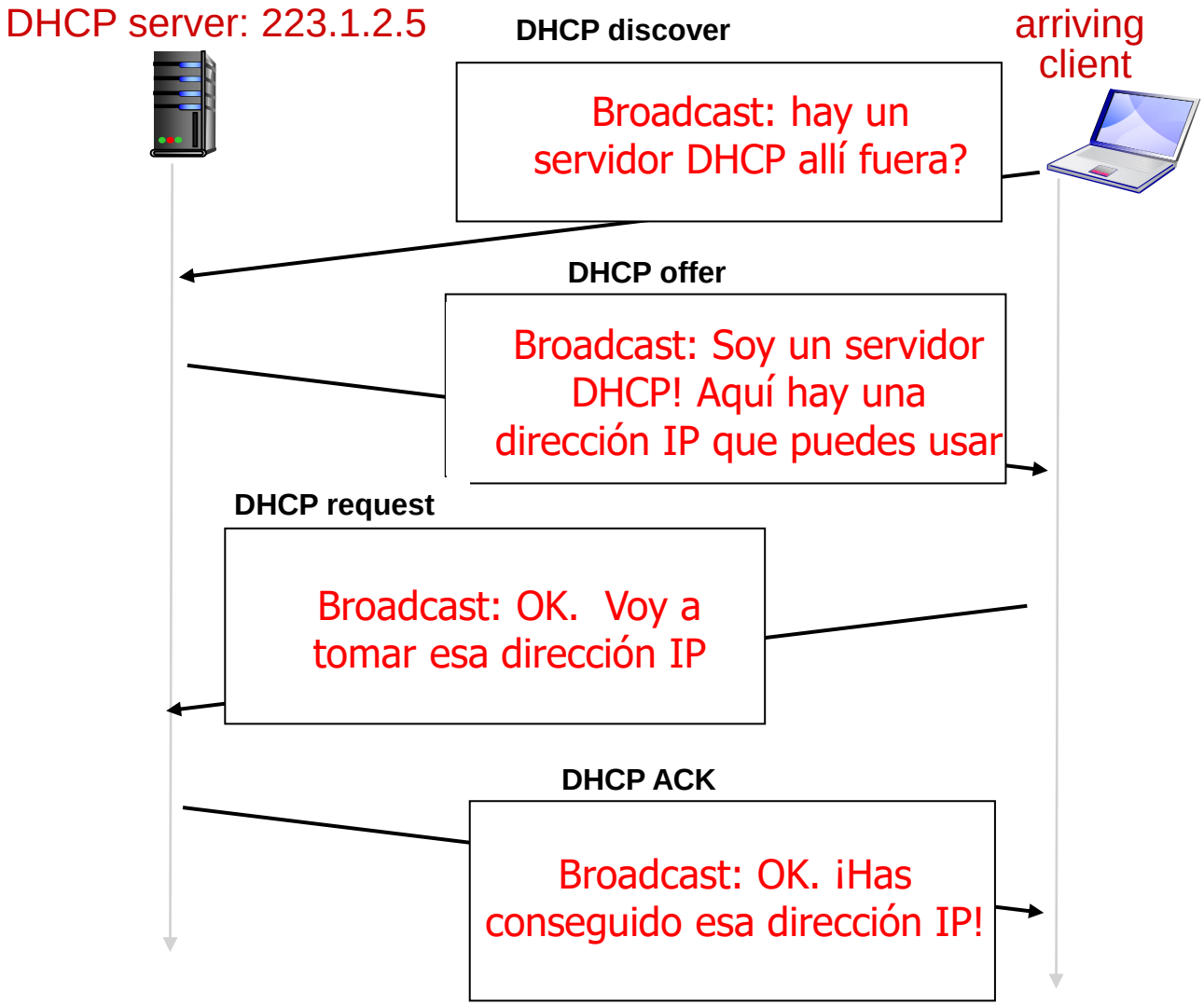
Resumen de DHCP:

1. host transmite "DHCP discover" msg [opcional]
2. Servidor DHCP responde con "DHCP offer" msg [opcional]
3. host pide dirección IP : "DHCP request" msg
4. Servidor DHCP envía dirección: "DHCP ack" msg

DHCP: Protocolo de Configuración Dinámica de Host



Escenario Cliente-Servidor de DHCP



DHCP: es más que direcciones IP

- DHCP puede retornar más que la dirección IP alojada en una subred:
  - Dirección del enrutador del primer salto para el cliente
  - Nombre y dirección IP del servidor DNS
  - Máscara de red
- DHCP es ampliamente usado en redes de acceso a internet residenciales y en redes LAN inalámbricas.

UPnP

- **Problema: ¿Cómo puede un host detrás de NAT permitir pedidos de conexiones entrantes?**
- **Solución:** Usar protocolo **Universal Plug and Play (UPnP)**
  - Una aplicación ejecutada en un host puede pedir un mapeo NAT entre su (IP privado, Port privado) y su (IP público, Port público).
  - **¿Si se acepta el pedido y se crea el mapeo entonces qué consecuencias tiene?**
  - Nodos de afuera pueden iniciar conexiones TCP con el (IP público, Port público) asignado.
  - **¿Cómo se pueden enterar máquinas de afuera de un servicio disponible por detrás de una NAT?**
  - UPnP permite a la aplicación conocer el valor de (IP público, Port público) de modo que la aplicación lo puede avisar al mundo externo.

UPnP

- **Ejercicio**: tu host detrás de UPnP y NAT tiene dirección privada 10.0.0.1 y ejecuta BitTorrent en el puerto 3345. La dirección pública del NAT es 138.76.29.7. BitTorrent pide crear mapeo en NAT y se obtiene mapeo de (10.0.0.1, 3345) a (138.76.29.7, 5001). **¿Cómo se entera otro host ejecutando BitTorrent de la aplicación en (138.76.29.7, 5001)?**
  - La aplicación avisa al tracker que está disponible en (138.76.29.7, 5001).
  - El host externo que ejecuta BitTorrent contacta el tracker y aprende que tu aplicación BitTorrent ejecuta en (138.76.29.7, 5001).
  - El host externo puede enviar paquete SYN a (138.76.29.7, 5001).
  - La caja NAT traduce esa dirección a (10.0.0.1, 3345) y luego se envía el paquete.
  - Luego tu host contesta con otro SYN al host externo y se establece la conexión.

UPnP

- **Conclusión**: Hemos visto se puede establecer conexión con un servidor o un nodo que usa P2P y que está por detrás de NAT y que para eso se puede usar UPnP
  - El único requisito es que el proceso que quiere iniciar conexión va a necesitar saber IP y puerto públicos del destino y dicha información va a ***tener que consultarla*** porque no la tiene.
  - Pero ese problema ya existía antes de NAT y se crearon mecanismos para resolverlo (P.ej. DNS para Web, registro para RMI, etc.);
  - Solo que con NAT el problema se agrava un poquito porque además de IP del servicio (que era lo que antes se necesitaba), ahora se va a necesitar un número de puerto.

ARP

- **Situación**: En internet una máquina tiene una o más direcciones IP; estas no pueden usarse para enviar paquetes
  - porque *el hardware de la capa de enlace de datos no entiende las direcciones de internet*.
  - La mayoría de los hosts se une a una LAN por una tarjeta de red que solo entiende direcciones LAN.
    - Cada tarjeta Ethernet viene con una dirección Ethernet de 48 b.
    - Las tarjetas envían y reciben tramas basadas en direcciones Ethernet de 48 b.
    - No saben nada de direcciones IP.
- **¿Por qué necesitamos preocuparnos por esto?**
  - Si queremos mandar un paquete a una computadora de destino dada por un IP, si no conseguimos la dirección LAN de la máquina con ese IP, entonces no podremos enviar el paquete.
  - Simplemente la capa de enlace de datos no lo va a poder procesar.

ARP

- **Problema**: ¿cómo se convierten direcciones IP en direcciones de Ethernet?
- **Solución: protocolo de resolución de direcciones** (**ARP**): el host de origen da salida a un paquete de difusión hacia Ethernet preguntando: ¿quien posee una dirección IP w.x.y.z ?
  - **Nota**: para hacer una difusión la dirección de destino consiste solo de 1s.
  1. La difusión llegará a cada máquina en Ethernet y cada una verificará su dirección IP.
  2. Al host de destino le bastará con responder con su dirección de Ethernet *E*.
  3. Así el host de origen aprende que la dirección IP de w.x.y.z está en el host con la dirección de Ethernet *E*.

ARP

- Casi cada máquina en Internet ejecuta ARP.
- El gerente de sistemas solo tiene que asignar a cada máquina una dirección IP y decidir respecto de las máscaras de subred. ARP hace el resto.

ARP

- Se pueden hacer optimizaciones para que ARP funcione con más eficiencia.
- **Optimización 1:** Una vez que una máquina ha ejecutado ARP, guarda el resultado en caso de que en poco tiempo tenga que ponerse de nuevo en contacto con la misma máquina.
  - La próxima vez encontrará la correspondencia en su propia caché, eliminando así la necesidad de una segunda difusión.
- **Optimización 2:** En muchos casos el host de destino necesitará devolver una respuesta, forzando también a que se ejecute el ARP para determinar la dirección Ethernet del emisor.
  - Esta difusión de ARP puede evitarse teniendo el host de origen que incluir su correspondencia IP a Ethernet en el paquete ARP.
  - Cuando la difusión de ARP llega al host de destino, se introduce la dirección IP y de Ethernet del origen en el caché del host 2 para su uso futuro.

ARP

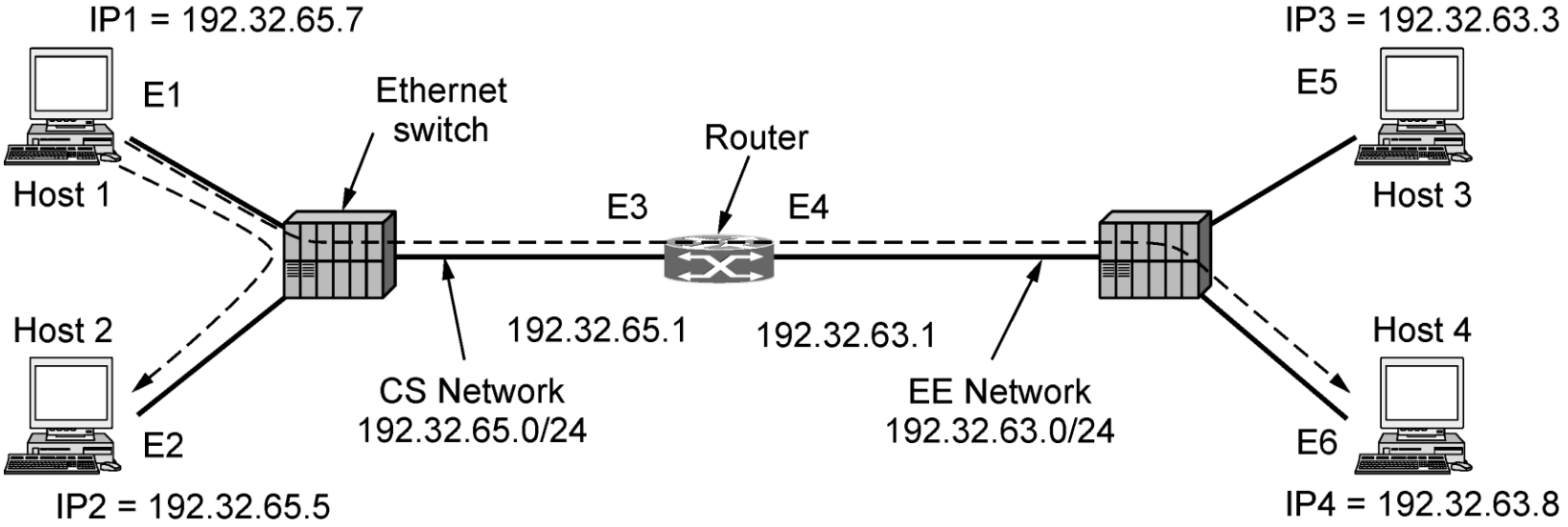
- **Optimización 3**: cada máquina difunde su correspondencia cuando arranca.
  - Esto se hace mediante un ARP que busca su propia dirección IP.
  - No debe haber una respuesta, pero un efecto lateral de la difusión es hacer una entrada en el **caché ARP** de todas las máquinas.
  - Si llega inesperadamente una respuesta, es que la misma dirección IP se ha asignado a dos máquinas.
    - La más reciente debe avisar al gerente de sistemas y no arrancar.

ARP

- **Problema**: **¿Cómo permitir que las correspondencias cambien?**
  - P. ej., cuando una tarjeta de Ethernet falla y se la reemplaza por una nueva
- **Solución**: las entradas en el **caché ARP** deben expirar en unos cuantos minutos.

ARP

- **Situación**: Cuando el host de origen y el host de destino están en distintas Ethernet LAN 1 y LAN 2 respectivamente separadas por enrutadores.
  - Si se usa ARP fallará porque el host de destino no verá la difusión (la difusión es algo interno a una LAN y no puede atravesar subredes y distintas LANs).
- **Problema:** ¿Cómo extender ARP al caso de que los hosts de origen y de destino estén en distintas LAN separadas por enrutadores?
- **¿Por qué necesitamos preocuparnos por esto?**
- Porque sino el paquete no puede recorrer ningún salto en el camino hacia el destino.



Frame	Source IP	Source Eth.	Destination IP	Destination Eth.
Host 1 to 2, on CS net	IP1	E1	IP2	E2
Host 1 to 4, on CS net	IP1	E1	IP4	E3
Host 1 to 4, on EE net	IP1	E4	IP4	E6

Two switched Ethernet LANs joined by a router.

ARP

- **Solución:**
  - El host 1 quiere enviar un paquete al host 4 en una red EE.
  - Host 1 va a ver que la dirección IP de destino no está en la red CS.
  - El host 1 sabe enviar todo ese tráfico al enrutador, el cual también se conoce como **default gateway** - esta es la menor dirección de la red (198.32.65.1).
    - Para enviar una trama al enrutador, el host 1 debe conocer la dirección Ethernet de la interfaz del enrutador en la red CS.
      - Lo descubre enviando una difusión ARP para 198.32.65.1, del cual aprende *E3*. Luego envía la trama.
  - El mismo mecanismo de búsqueda es usado para enviar un paquete de un enrutador al siguiente sobre una secuencia de enrutadores en un camino de internet.

ARP

- Cuando la tarjeta de red de Ethernet del enrutador consigue esta trama, da el paquete al software IP.
  - Conoce de las máscaras de red que el paquete debería ser enviado en la red EE donde va a alcanzar el host 4.
  - Si el enrutador no conoce la dirección Ethernet para el host 4, entonces va a usar ARP nuevamente.
- Las direcciones Ethernet cambian con la trama en cada red mientras que las direcciones IP permanecen constantes (porque indican los puntos finales a lo largo de todas las redes interconectadas).