

# Redes y comunicaciones

## Resumen teorías - Capa de red

<b>Clase 1 - Introducción a la capa de red</b>	<b>4</b>
Estructura de Internet	4
Modelo de reloj de arena	5
¿Reloj de arena actual?	6
¿Cómo trabaja IP?	6
Protocolos IP actuales	7
Características de IPv4	7
Funcionalidad	7
No orientado a conexión	7
Esquema de IP en TCP/IP	8
Direccionamiento IP	9
Direcciones IP	9
Tipos de Direcciones IP	10
Direcciones IP especiales	11
Direcciones Privadas	11
Problemas con direccionamiento IP fijo	12
Subnetting	12
Ejemplos	13
VLSM subnetting	13
CIDR supernetting	14
Datagrama IP	16
Campos de un datagrama IP	17
Version	17
Header length	17
DS/ECN field	17
Differentiated Service (DS):	17
Explicit Congestion Notification (ECN)	17
Total length	17
Identification	18
Flags	18
Fragment offset	18
Time To Live (TTL)	18
Protocol	18
Header checksum	19
Source IP address	19
Destination IP address	19
Options	19
Padding	20
Data/payload	20

Ruteo	20
Tabla de ruteo	21
Tareas del ruteo	21
Fragmentación	22
<b>Clase 2 - ICMP</b>	<b>23</b>
Formato de Mensaje	23
Mensajes	23
Ping (Echo)	24
Destino inalcanzable	25
TTL expirado	26
Traceroute	27
<b>Clase 3 - Protocolos de ruteo dinámico</b>	<b>29</b>
Routing	31
Tipos de Routing	31
Ruteo estático	32
Ruteo dinámico	32
Routing domain	32
Autonomous System (AS)	32
Protocolos de ruteo dinámico	33
Estructura de Internet	35
Otra clasificación	35
<b>Clase 4 - DHCP</b>	<b>37</b>
DHCP	37
Mensajes	38
Ejemplo	38
Proceso	40
DHCP relay	41
<b>Clase 5 - NAT</b>	<b>42</b>
Problemas con IPv4	42
NAT (Network Address Translation)	42
Problemas con IP Privadas	42
Procesos de Traducción	42
NAT básico	43
Ejemplo estático	44
Ejemplo dinámico	44
NAPT	45
Ejemplo con pool	45
Ejemplo con overload/masquerade	46
Port forwarding	46
Ejemplo	47
Conclusiones	47
<b>Clase 6 - IPv6 - Parte 1</b>	<b>48</b>
Comutación de paquetes	48
Problemas con NAT	48

Problemas en IPv4	49
Otras cuestiones no contempladas desde el inicio	49
Beneficios de IPv6	49
Cambios en IPv6	50
Funcionalidad de IPv6	51
Servicios nuevos	52
Cabeceras de extensión	52
Orden de las cabeceras de extensión	53
Más direcciones disponibles	53
Tipos de direcciones:	53
Alcance (Scope) de las direcciones Unicast	54
Formato de las direcciones	55
Notación Direcciones IPv6	55
Direcciones IPv6 locales	56
Generación de IID	56
Direcciones IPv6 Site-Local	57
Direcciones IPv6 Unique-Local	57
Direcciones IPv4-compat IPv6	58
Direcciones IPv4-mapped IPv6	58
Direcciones IPv6 globales	58
Generación de IID	59
Direcciones IPv6 Multicast	59
Ejemplos	60
Direcciones IPv6 Multicast SD	60
IPv6 Multicast mapeada en IEEE EUI-48	61
Direcciones IPv6 - Casos Especiales	61
<b>Clase 7 - IPv6 - Parte 2</b>	<b>62</b>
ICMPv6	62
IPv6 Stateless Autoconfiguration	62
Router discovery	63
Mensaje RA	64
IPv6 autoconfiguration DHCPv6	64
Neighbor Discovery	65
Neighbor advertisement	68
PMTU (Path MTU) Discovery	68
Transición de IPv4 a IPv6	69
Dual stack IPv4/IPv6	69
Túneles IPv4/IPv6	70
Manuales	70
GRE	70
SIT	71
6to4	72
Teredo	72

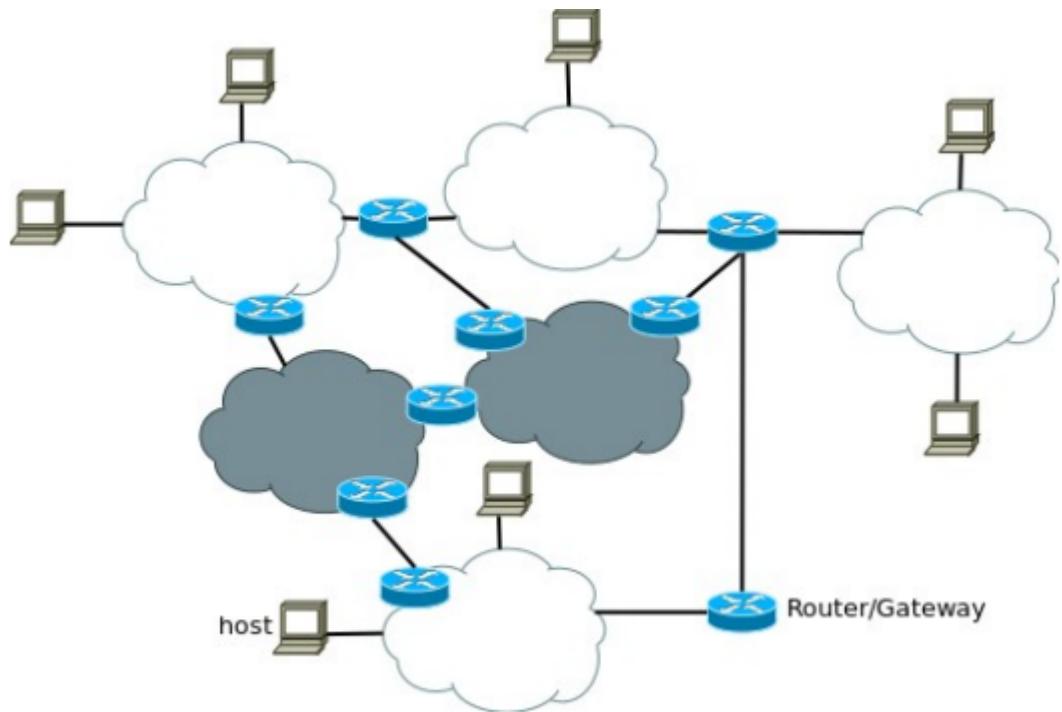
# Clase 1 - Introducción a la capa de red

## Estructura de Internet

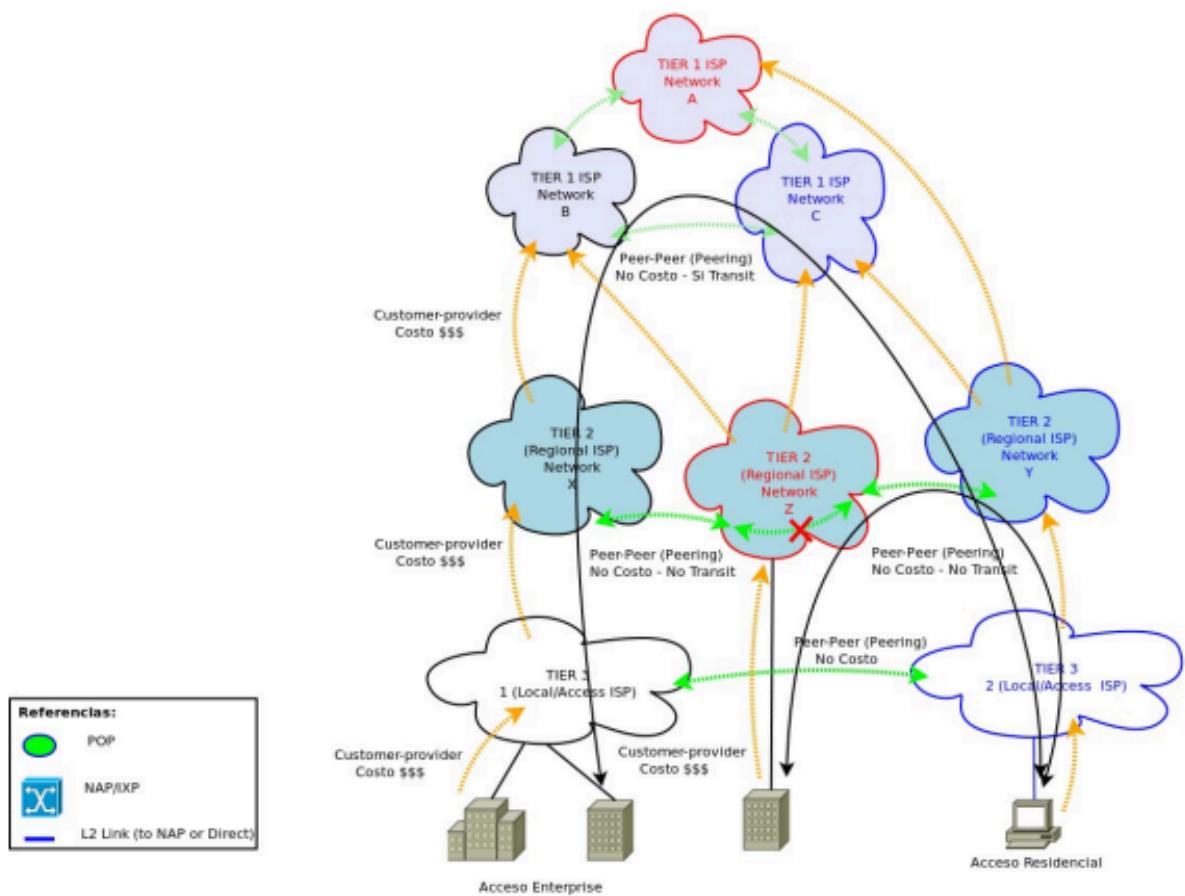
Es un conjunto de redes interconectadas y agregadas

Es una red de redes

Protocolo común: IP

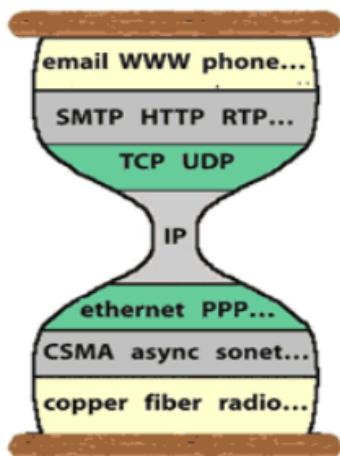


La organización de este conjunto de redes se da de forma jerárquica

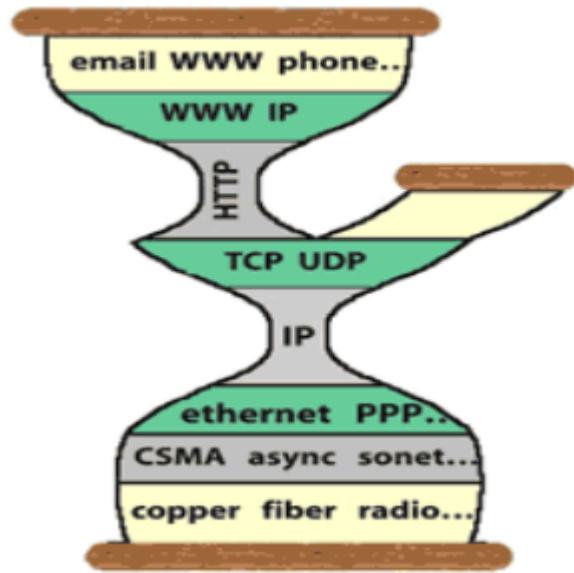


Internet tiene: Net Neutrality (Neutralidad de la Red), es decir, se trata a todo el tráfico de forma equivalente por redes de ISP y carriers

## Modelo de reloj de arena



¿Reloj de arena actual?

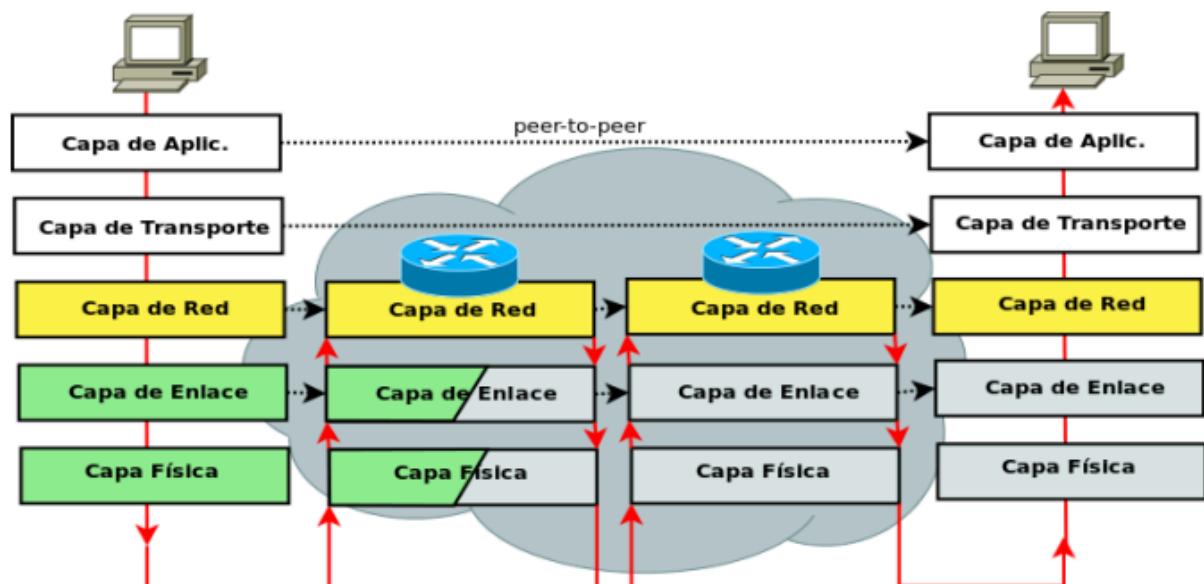


¿Cómo trabaja IP?

Es end-to-end, de extremo a extremo

El ruteo se produce hop-by-hop, pasando por los diferentes routers

Cada nodo debe implementar IP



## Protocolos IP actuales

Brinda servicios a Transporte  
Usa servicios de Enlace  
IPv4, comúnmente llamado IP  
IPv6 llamado antiguamente IP-ng  
IPv4 e IPv6 no son versiones de uno mismo, no son compatibles

## Características de IPv4

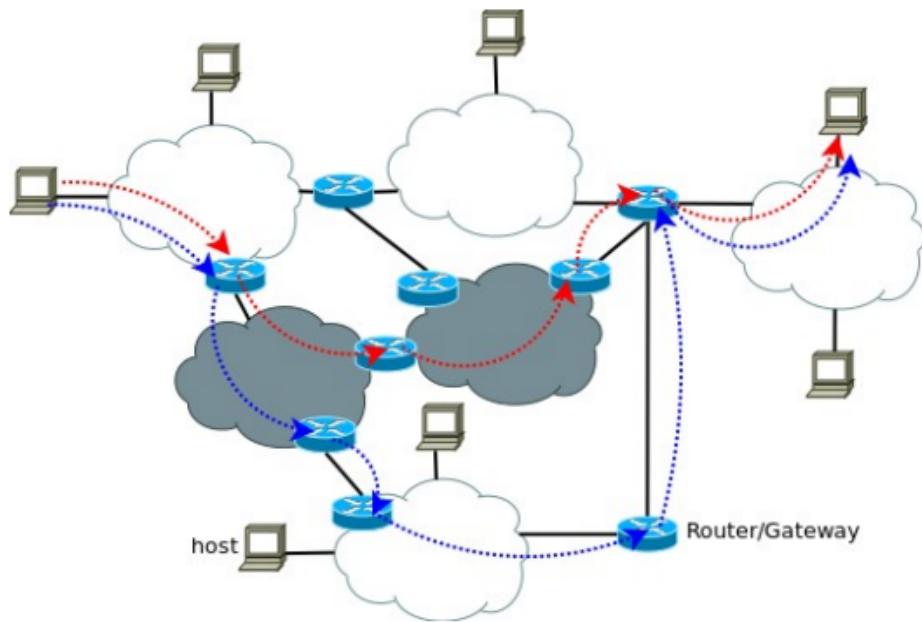
Protocolo de Red no orientado a conexión  
Protocolo de mejor esfuerzo: best-effort. No confiable (no asegura el arribo de los mensajes).  
PDU: datagrama o paquete

## Funcionalidad

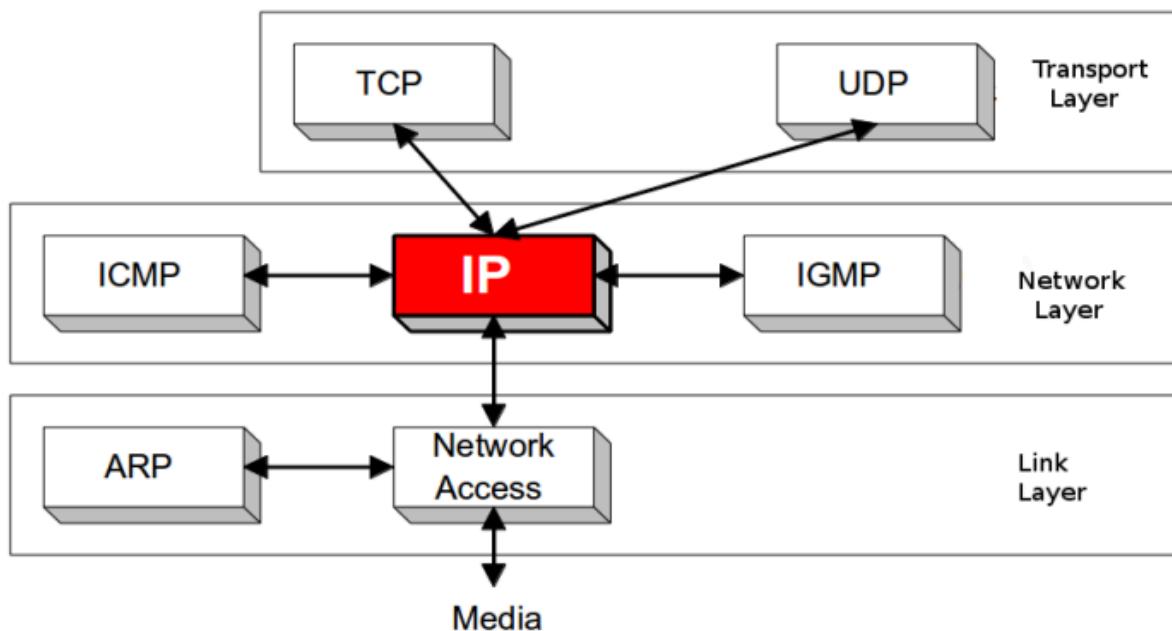
- Direccionamiento
- Ruteo/Forwarding/Switching L3
- Mux/Demux de protocolos superiores
- Accesorias (solucionar deficiencias del protocolo):
  - Fragmentación
  - Otras: como evitar loops (TTL), detección de errores

## No orientado a conexión

Protocolo de Red no orientado a conexión a diferencia de TCP u otros protocolos considerados de red X.25, ATM



## Esquema de IP en TCP/IP



Es el núcleo de Internet

Requiere protocolos Helpers (ICMP, IGMP)

## Direccionamiento IP

Dirección IP: identifica únicamente un punto de acceso (interfaz) a la red

Un router o un host multi-homed tienen varias IPs. Cada interfaz tiene un valor único

Una interfaz puede tener varias direcciones IP

Tienen un significado global en la Internet o privado (local)

Globales: asignadas por autoridad central:

- En un principio: John Postel, InterNIC (Internet Network Information Center)
- Hoy: el IANA (Internet Assigned Numbers Authority), responsable, el ICANN, delegando la asignación a los RIRs (Regional Internet Registry), siendo para América Latina y parte del Caribe: LACNIC

## Direcciones IP

Son números de 32 bits, expresados en notación decimal delimitada por puntos byte a byte (e.g. 163.10.45.77)

Son 4G de direcciones ( $2^{32}$ ) puras, que organizadas en forma jerárquica se reducen

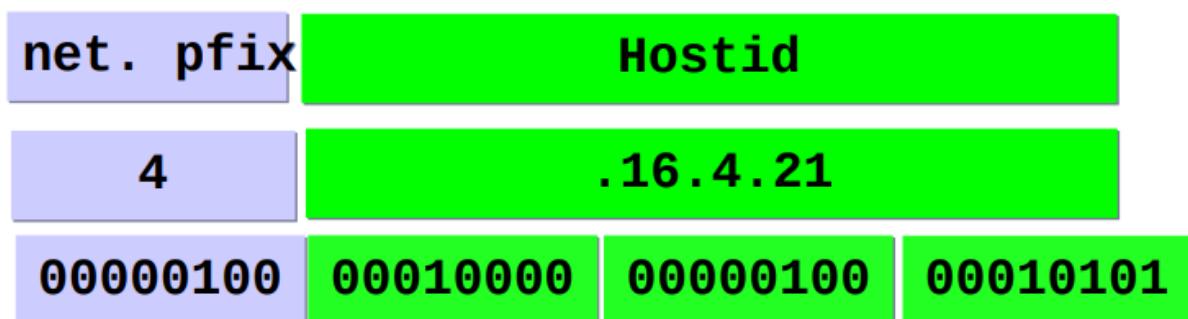
Para facilidad de los usuarios se usa mapping con nombres de dominio (DNS)

Son necesarias para rutear la información por Internet

Son direcciones lógicas

Codificadas en dos partes:

- Red (net)
- Anfitrión (host)



Hasta 1981, sólo había pocas redes con muchos hosts disponibles. Sin clases.

Redes 8 bits

En 1981: RFC-790 define clases

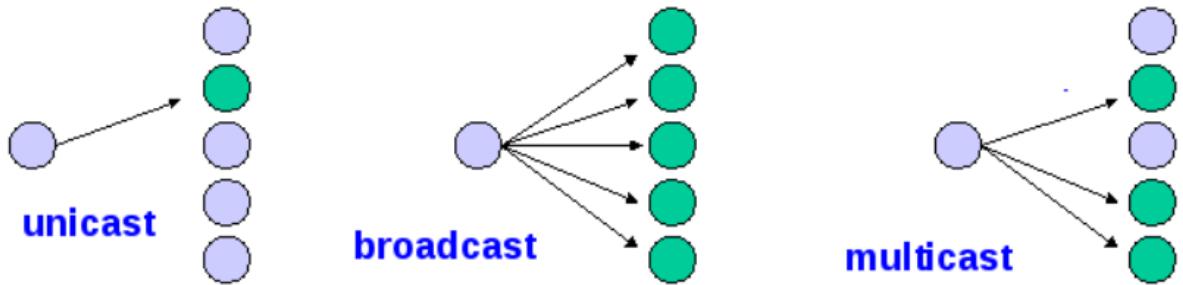
Cada clase usada para diferentes tipos de redes:

- Clases A, pocas redes grandes
  - Clases B, más redes medianas
  - Clases C, muchas redes chicas

En 1984 se agrega una tercer parte, subred y se requiere un máscara: RFC-917

## Tipos de Direcciones IP

- Unicast: destino a un host/interfaz en particular, son las más comunes e.g: 172.16.4.21
  - Broadcast: destino a todos los hosts en una red
  - Multicast: destinada a un grupo de hosts en una red o varias redes (Clase D)
  - Anycast: destinada al primero que resuelva. IPv4 no hay casos especiales



## Direcciones IP especiales

- Loopback: unicast, red clase A. 127.0.0.1
  - La más utilizada: 127.0.0.1, localhost
  - Aunque podría ser cualquier otra: 127.10.0.1 127.34.34.1, etc.
  - No pueden salir de nuestra red
- Dirección de red: la primera (zero)
  - e.g. 172.16.0.0, 192.168.1.0
  - Es la dirección que identifica a una red (la cual tiene varios hosts)
- Dirección de broadcast:
  - Directed Broadcast: la última (ones)
    - e.g. 172.16.255.255, 192.168.1.255
    - Es la dirección de broadcast de cada red
  - Limited Broadcast: (all ones)
    - 255.255.255.255
- “Este host”, cuando aún no tiene asignada una dirección: 0.0.0.0 (Utilizada en BOOTP/DHCP)

## Direcciones Privadas

No tienen significado global, no son únicas

Definidas en RFC-1918

Se utilizan en Intranets

Redes autónomas sin conexión a Internet

Para conectarse a Internet requieren un proceso de transformación: NAT, RFC-1631

No deberían pasar a Internet

Filtradas por routers de borde

Son:

- 10.0.0.0 - 10.255.255.255, (1 clase A)
- 172.16.0.0 - 172.31.255.255, (16 clases B)
- 192.168.0.0 – 192.168.255.255, (256 clases C)

## Problemas con direccionamiento IP fijo

Prefijos de longitud fija por clase, provoca un uso ineficiente en el espacio de direcciones

Muchos equipos producen escasez de direcciones

El crecimiento acelerado de Internet evidencia la falta de escalabilidad del esquema

Crecimiento de tablas de ruteo en el núcleo de la red

Codificar la red en la dirección IP implica que si un host cambia de red, cambiará su dirección (IP Mobility)

Problema atacado en IPv4, mejor resuelto en IPv6

Soluciones IPv4: subnetting, CIDR, NAT, DHCP

Definitivamente solucionados en IPv6

## Subnetting

Se toma una parte del hostid

Se utiliza para generar subredes dentro de la red

Se agrega una “máscara” de bits

Para saber la subred se aplica un “AND” lógico

Agrega un nivel más en la estructura: Red, Subred, Host

Ejemplo usar un bloque clase B como 256 clases C

Las máscaras se escriben en notación decimal o hex: 255.255.255.0 o 0xff ff ff 00

También pueden escribirse como longitud de prefijo: /24

Ejemplos:

- 255.255.255.192 /26
- 255.224.0.0 /11
- 255.255.255.252 /30

Las máscaras defaults:

- Clase A: 255.0.0.0
- Clase B: 255.255.0.0
- Clase C: 255.255.255.0

## Ejemplos

Valen los mismos conceptos para redes completas

Ejemplo para 172.16.4.21:

- Dirección de broadcast: 172.16.4.255
- Dirección de red: 172.16.4.0
- Redes y hosts:  $(2^n)$ ,  $(2^{(32-(m+n))})$

Ejemplo Clase B con /24: n=8 (bits subnet), m=16 (bits net) (n+m = bits mask):

- Cantidad de hosts:  $2^{(32-(8+16))} = (2^8) = 256$
- Cantidad de hosts útiles:  $256-2 = 254$
- Cantidad de subredes:  $(2^8)$
- Cantidad de subredes útiles:  $(2^8)-2$  (se aclara más abajo que actualmente se usan todas)
- Las 2 que se restan a las subredes se pueden utilizar: dando:  $2^8$  redes útiles

En un principio, por cuestiones de compatibilidad con viejos sistemas no se permitía utilizar la primera ni la última subred:

Red A: 193.168.4.0 Dir. de sub-red = Dirección de red

Red D: 193.168.4.192 Dir. de “sub-bcast” = Dirección de broadcast de red

Se genera mucho desperdicio, en este caso el 50% de las direcciones

RFCs subsiguientes lo permiten, hoy completamente difundido

## VLSM subnetting

Variable Length Subnet Mask. RFC-1009, RFC1878

La longitud de la máscara no tiene necesidad de ser para todas las subredes igual

¿Qué sucede si se tienen diferentes cantidades de hosts en las subredes?

Por ejemplo la Red A: tiene 70 hosts, la Red B tiene 40 host y la C,D tienen 25 hosts

El siguiente esquema no sirve:

□ 193.168.4.0 /26

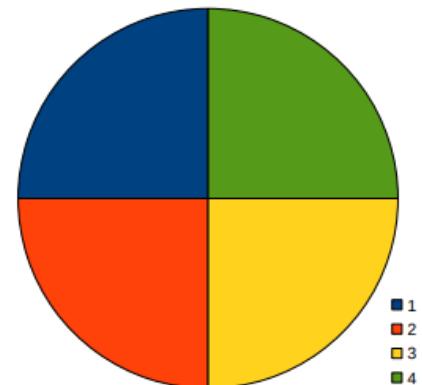
/26 00 193.168.4.0,  
/26 01 193.168.4.64,  
/26 10 193.168.4.128,  
/26 11 193.168.4.192. 62 host c/u.

Se deben agrupar o dividir redes del esquema fijo:

/25 000 193.168.4.0,  
/26 100 193.168.4.128,  
/27 110 193.168.4.192,  
/27 111 193.168.4.224. 126, 62, 30, 30 hosts respectivamente.

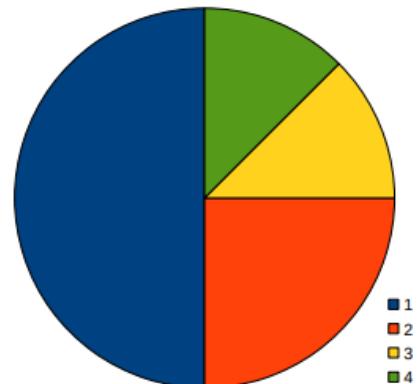
Subredes iguales: /26

255.255.255.192  
255.255.255.192  
255.255.255.192  
255.255.255.192



VLSM: /25, /26, /27, /27:

255.255.255.128  
255.255.255.192  
255.255.255.224  
255.255.255.224



CIDR supernetting

Classless Inter Domain Routing

Hasta 1993, se asumía, de acuerdo a la clase de la dirección IP la máscara default

Los bits de la red definida por la clase eran fijos

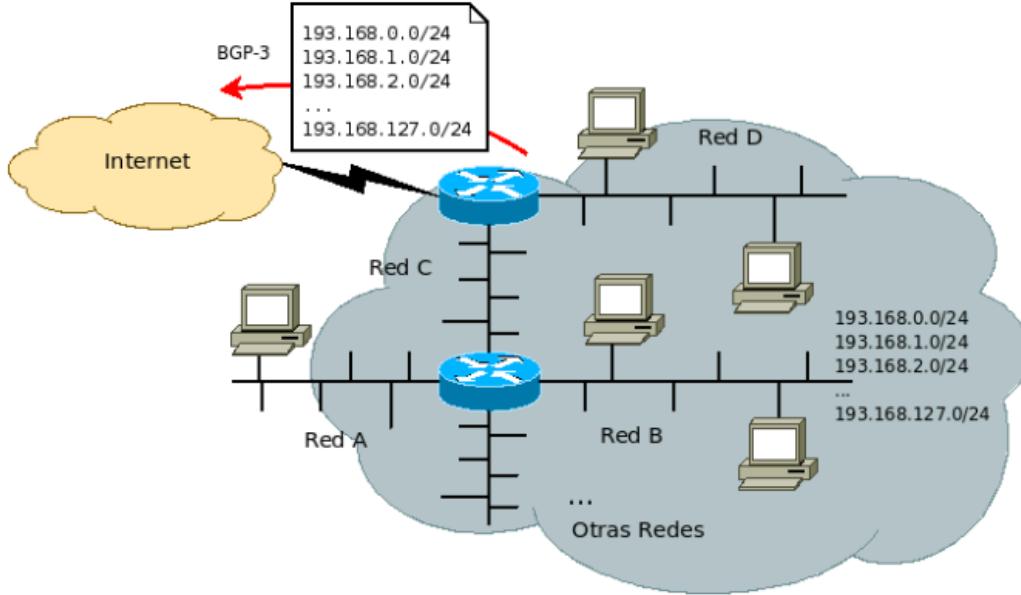
El direccionamiento era classful

Con CIDR, se sacan las clases: Classless y siempre debe haber una máscara o longitud de prefijo

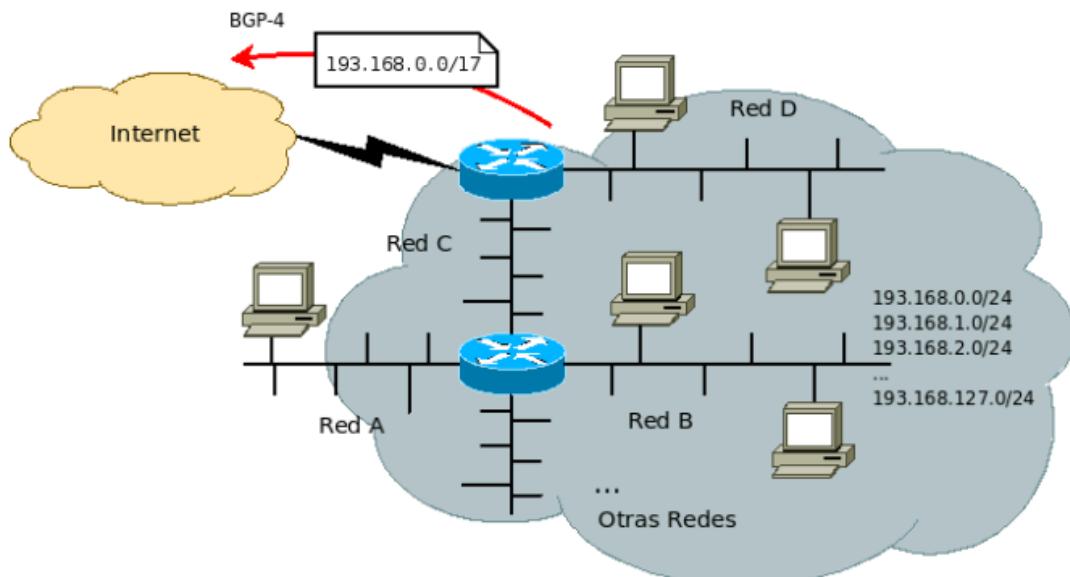
Permite agrupar, reducen longitud de tablas de ruteo:

- 193.168.0.0, 193.168.1.0, ... 193.168.3.0 /24
- En 193.168.0.0/22 se agrupan 4 redes

## ■ BGP update classful:



## ■ BGP update classless:

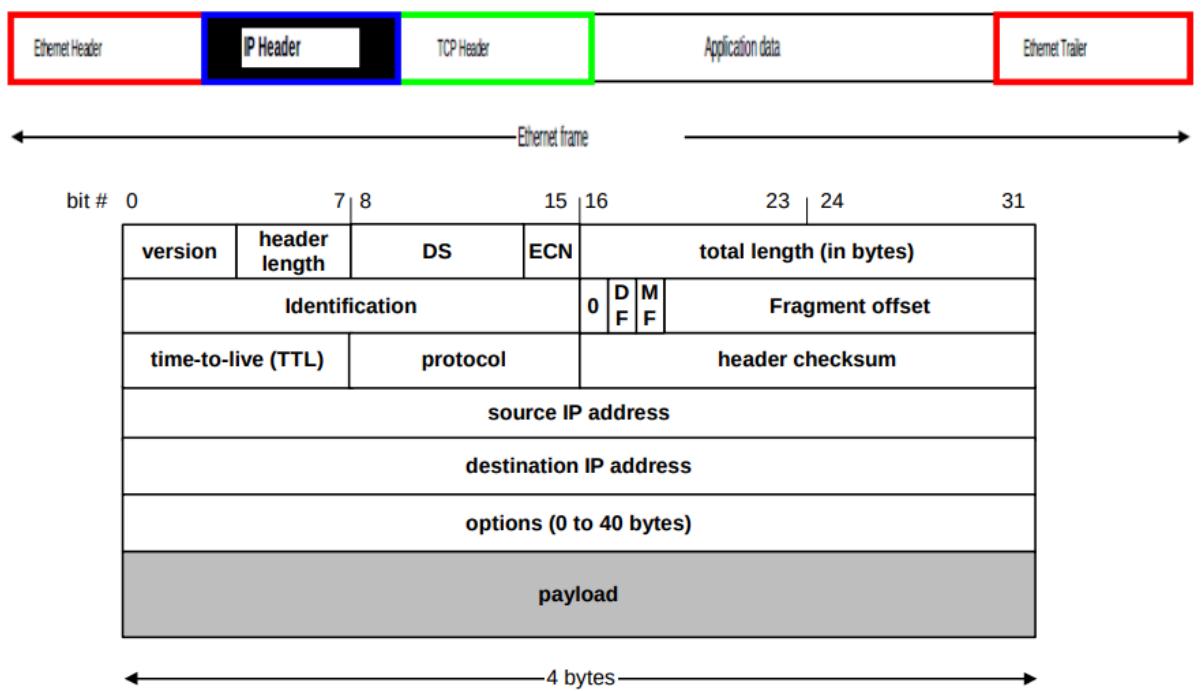
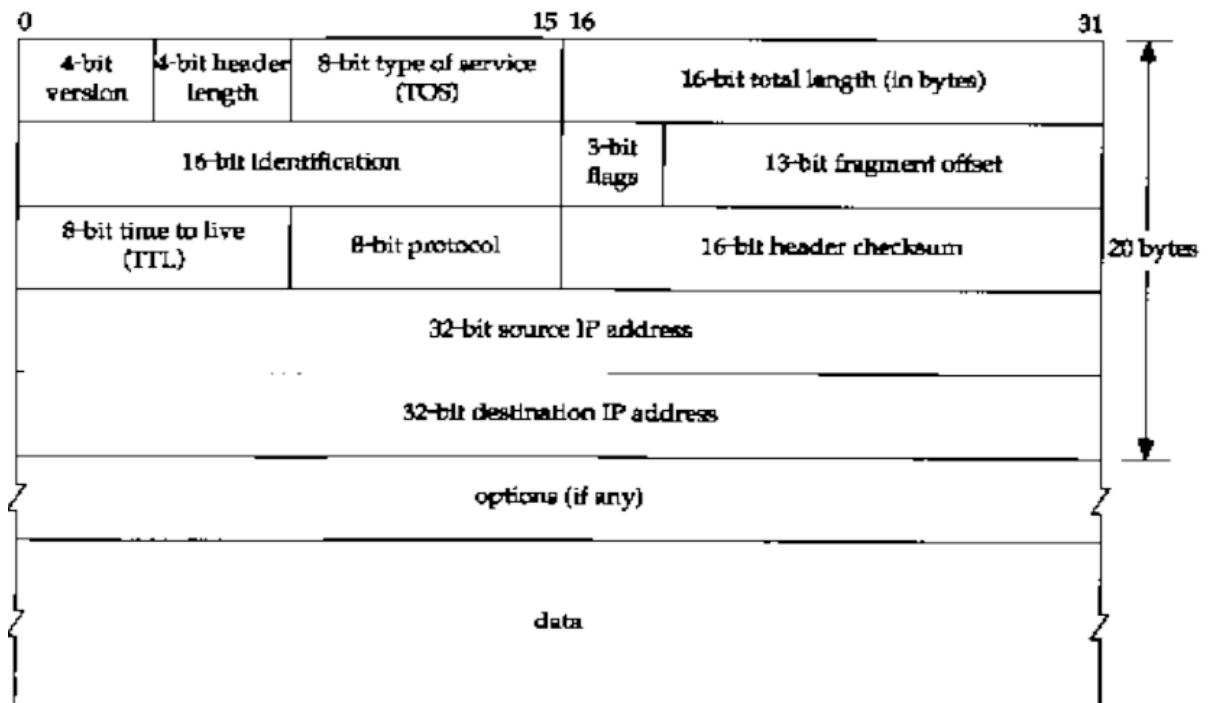


Agrupación de bloques de forma contigua por ISP

Asignación por regiones geográficas

El IANA crea los RIRs (Regional Internet Registry)

## Datagrama IP



## Campos de un datagrama IP

### Version

4 bits

Versión actual 4, la nueva 6

### Header length

4 bits

Longitud en múltiplos de 4B

Valor mínimo: 5 (20B)

Valor máximo: 15 (60B)

### DS/ECN field

8 bits

TOS (Type of Service), DSCP Differentiated Service CodePoint

Differentiated Service (DS):

6 bits

Usado para marcar QoS

Explicit Congestion Notification (ECN)

2 bits

Usado en control de congestión con TCP

### Total length

16 bits

Longitud total del datagrama IP, incluyendo la cabecera y los datos

Puede ser de hasta 65,535 bytes

### Identification

16 bits

Identificador único que permite identificar y reensamblar fragmentos en el destino

Todos los fragmentos del mismo datagrama comparten este valor

Utilizado para la fragmentación

### Flags

3 bits:

1. Es 0
2. DF bit (Do not fragment)
3. MF bit (More fragments)

Utilizados para la fragmentación

### Fragment offset

13 bits

Indica la posición del fragmento dentro del datagrama original

Cada fragmento se desplaza en bloques de 8 bytes

### Time To Live (TTL)

8 bits

Cuantos saltos puede dar el datagrama

Evita loops

Emisor lo pone a un valor, e.g. 128 o 64

Cada router por el que pasa lo decrementa en 1

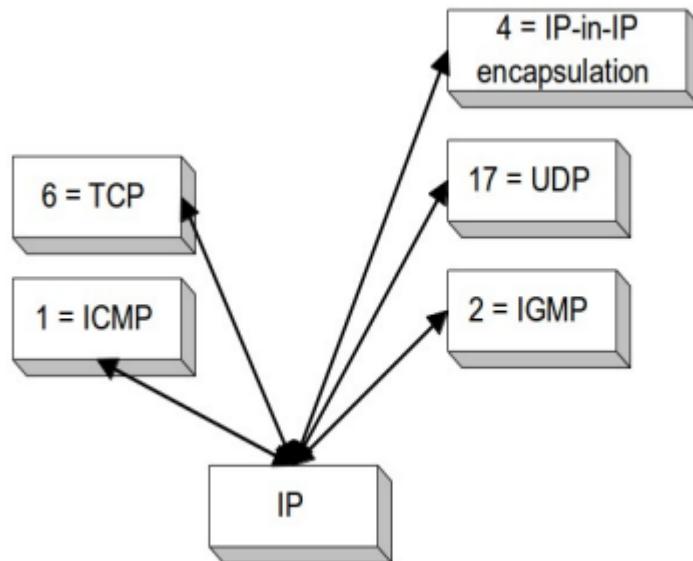
Si está más de un segundo también

Si llega a un router que no está en la red destino y TTL=0, se descarta

### Protocol

8 bits

Para mux/demux



Header checksum

16 bits

Checksum del header solamente

Source IP address

32 bits

IP del dispositivo que envía el datagrama

Destination IP address

32 bits

IP del dispositivo que recibe el datagrama

Options

Tamaño variable entre 0 y 40 bytes

Si las opciones están presentes, la longitud de la cabecera aumentará en consecuencia

Posibilidades:

- Security restrictions
- Record Route

- Timestamp
- (loose) Source Routing
- (strict) Source Routing

Padding

Agregado para ser múltiplo de 4B

Data/payload

Datos (tcp, udp o lo que sea)

## Ruteo

Tabla de ruteo:

- Estructura en hosts y routers (gateways) que indica cómo despachar un mensaje
- Perspectiva del vecino, siguiente salto

Host:

- No despacha mensajes que recibe que no son para él
- Despacha solo sus mensajes mirando su tabla de ruteo

Router: nodos intermedios, más de una interfaz, despacha mensajes mirando tabla de ruteo, desde cualquier interfaz

Host multihome: tiene varias interfaces, no rutea

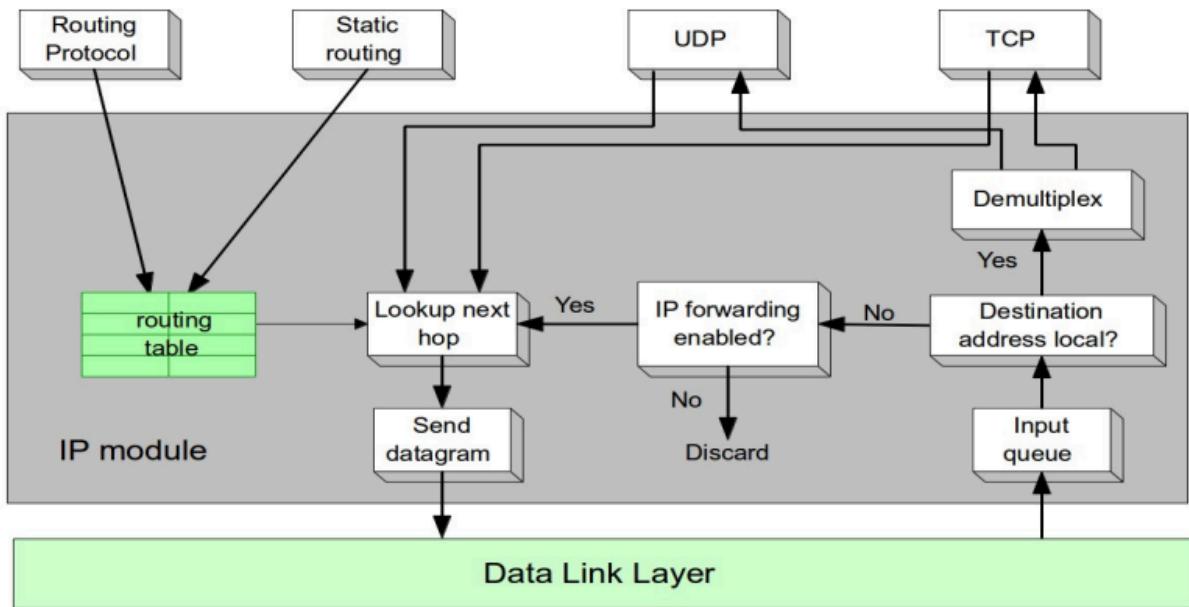
Ruteo: seleccionar la interfaz de salida y el próximo salto. Lo hacen routers y hosts

Forwarding/Despacho: pasar el paquete desde una interfaz de entrada hacia una de salida. Solo lo hacen routers

El forwarding es más intensivo

El ruteo es de control, alimentado por protocolos de enrutamiento (routing)

El forwarding es de datos, envía protocolos enrutados (routed)



## Tabla de ruteo

Estructura de tabla de ruteo:

- Red Destino (Net/Mask)
- Next Hop (Próximo salto)
- Interfaz de salida

En un Host es más simple:

```

andres@h1:~$ netstat -nr
Destination      Gateway          Genmask        Metric  Iface
193.168.4.224   0.0.0.0          255.255.255.224  0        e0
193.168.4.128   193.168.4.225  255.255.255.192  2        e0
0.0.0.0          193.168.4.225  0.0.0.0          -        e0
  
```

## Tareas del ruteo

- Validación de datagrama: IP header
- Calcula el checksum (solo header)
- Leer IP destino
- Buscar en tabla de ruteo, seleccionar prefijo más largo (“best match”)
- Decrementar TTL
- Fragmentar (alternativo)
- Transmitir o Descartar

- Generar ICMP (alternativo)

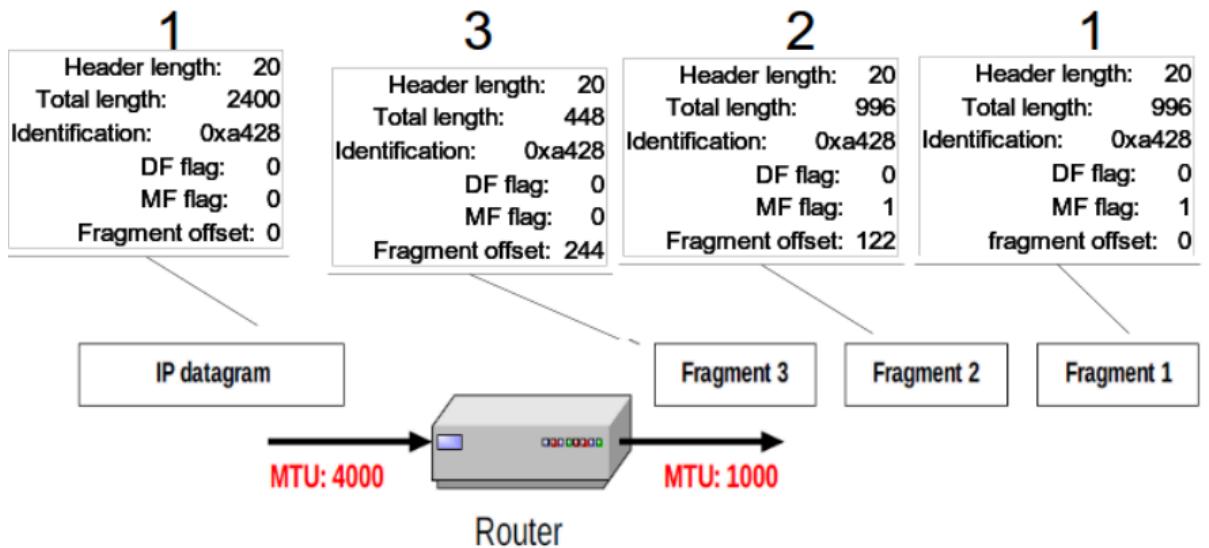
## Fragmentación

Debido a que hay diferentes capas de enlaces con diferentes MTUs

Fragmentos múltiplos de 8 bytes (offset en unidades de 8 bytes)

Se deben agregar los headers

version	header length	DS	ECN	total length (in bytes)				
Identification				0	D	M		Fragment offset
time-to-live (TTL)		protocol		header checksum				



# Clase 2 - ICMP

Protocolo de L3

Protocolo “Helper” de IP

IP carece de control, el mismo es dado por un protocolo auxiliar

ICMP no le agrega confiabilidad a IP, solo brinda un “feedback” para poder resolver problemas en la red

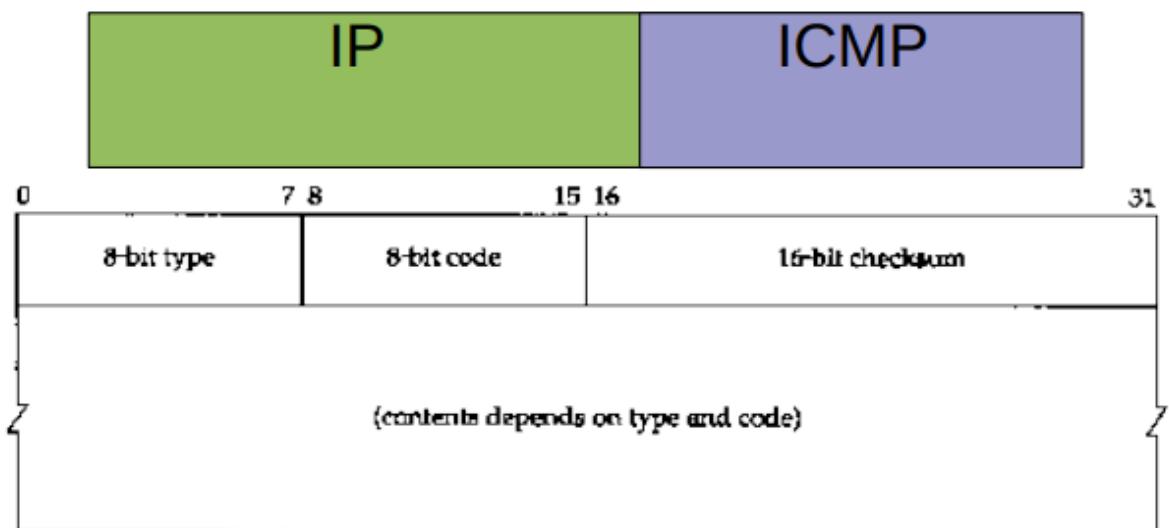
ICMP podría ser prescindible en IPv4, aunque en la RFC se indica que se debe implementar en cada módulo IP

Definido en RFC-792

ICMP se encapsula en IP

ICMP no es un protocolo de transporte, ya que no fue concebido para llevar datos de usuario

## Formato de Mensaje



## Mensajes

- Echo Request/Echo Reply (PING)
- Destino Inalcanzable
- TTL expirado
- Source Quench (Control de Congestión)

- Redirección de Ruta
- Address Mask y Timestamp

## Ping (Echo)

Pensado para probar conectividad IP entre dos hosts

Sirve para medir el RTT min/avg/max/desviación estándar/loss, y de esta forma poder diagnosticar problemas

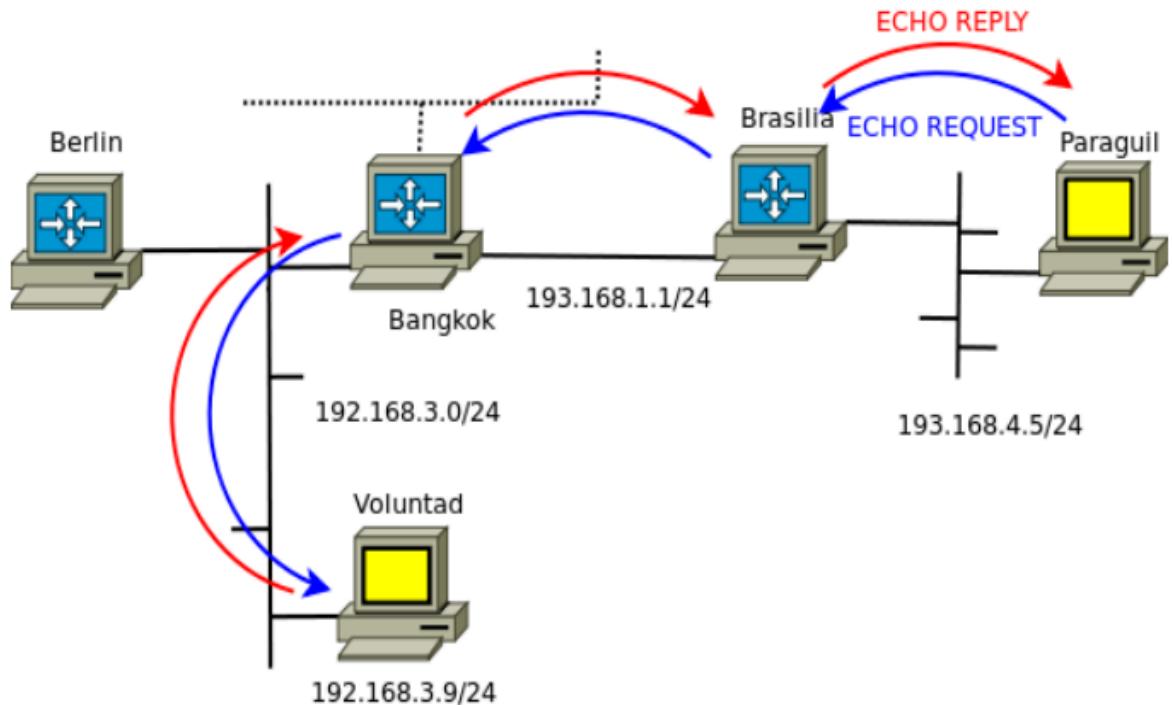
Packet Internet Gopher, el nombre basado en el sonido de un sonar de un submarino al escanear

Si un nodo recibe un ICMP Echo Request, debe responder copiando el contenido con un Echo Reply (PONG). RFC-1122

Actualmente, muy desprestigiado ;-) Es filtrado

```
andres@h1(paraguil):~$ ping -c 3 193.168.3.9
PING 193.168.3.9 (193.168.3.9) 56(84) bytes of data.
64 bytes from 193.168.3.9: icmp_seq=1 ttl=53 time=149 ms
64 bytes from 193.168.3.9: icmp_seq=2 ttl=53 time=150 ms
64 bytes from 193.168.3.9: icmp_seq=3 ttl=53 time=147 ms

--- 193.168.3.9 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time
2002ms
rtt min/avg/max/mdev = 147.498/149.014/150.128/1.197 ms
```



## Destino inalcanzable

Para indicar que una red, un host, un puerto es inalcanzable, existen diferentes causas:

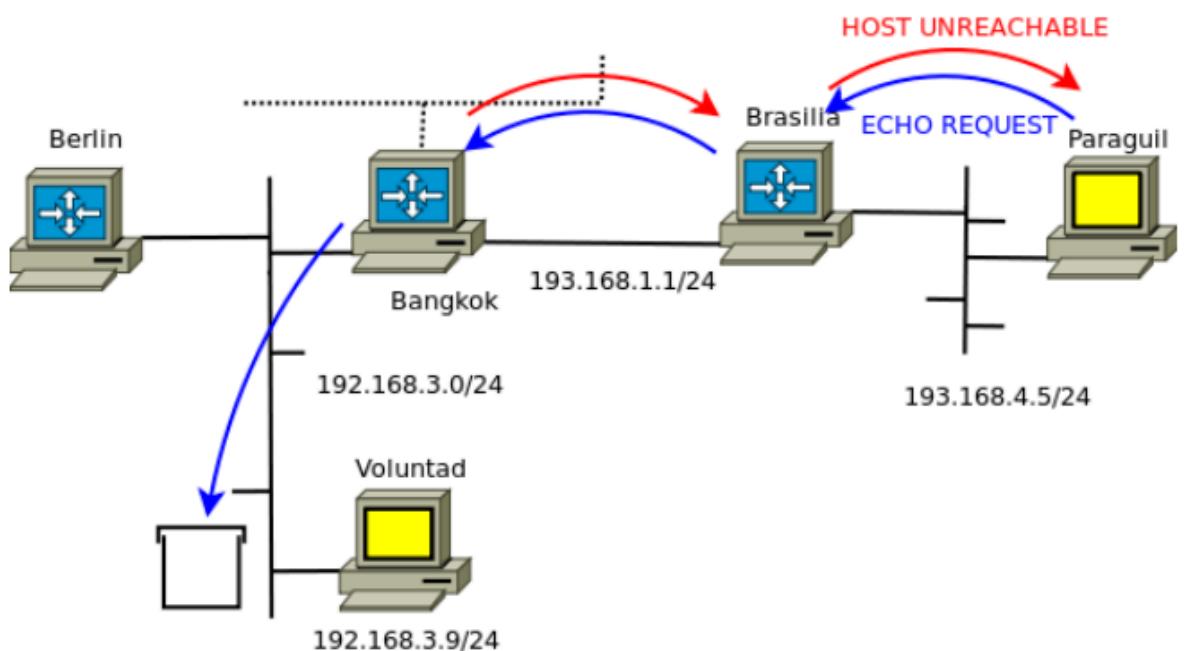
- Host Inalcanzable (Host Unreachable). Posibles causas:
  - No está encendido el host
  - No responde ARP
- Red Inalcanzable (Network Unreachable). No tiene el router una ruta en la tabla de ruteo a esta red
- Puerto Inalcanzable (Port Unreachable). No hay un proceso UDP en el puerto
- Los mensajes requieren fragmentación
- El mensaje fue filtrado (admin)
- Otros

```

andres@h1(paraguil):~$ ping -c 3 193.168.3.10
PING 193.168.3.10 (193.168.3.10) 56(84) bytes of data.
From 193.168.1.2 icmp_seq=1 Destination Host Unreachable
From 193.168.1.2 icmp_seq=2 Destination Host Unreachable
From 193.168.1.2 icmp_seq=3 Destination Host Unreachable

--- 193.168.3.10 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet
loss, time 2007ms
, pipe 3

```



## TTL expirado

El tiempo de vida ha expirado. En realidad es el hop count con el cual salió el mensaje ha expirado

Time Exceeded:

- Tiempo Excedido en viaje
- Tiempo Excedido en re-ensamblado

TTL en IP, no solo ICMP

Valor máximo de TTL=255

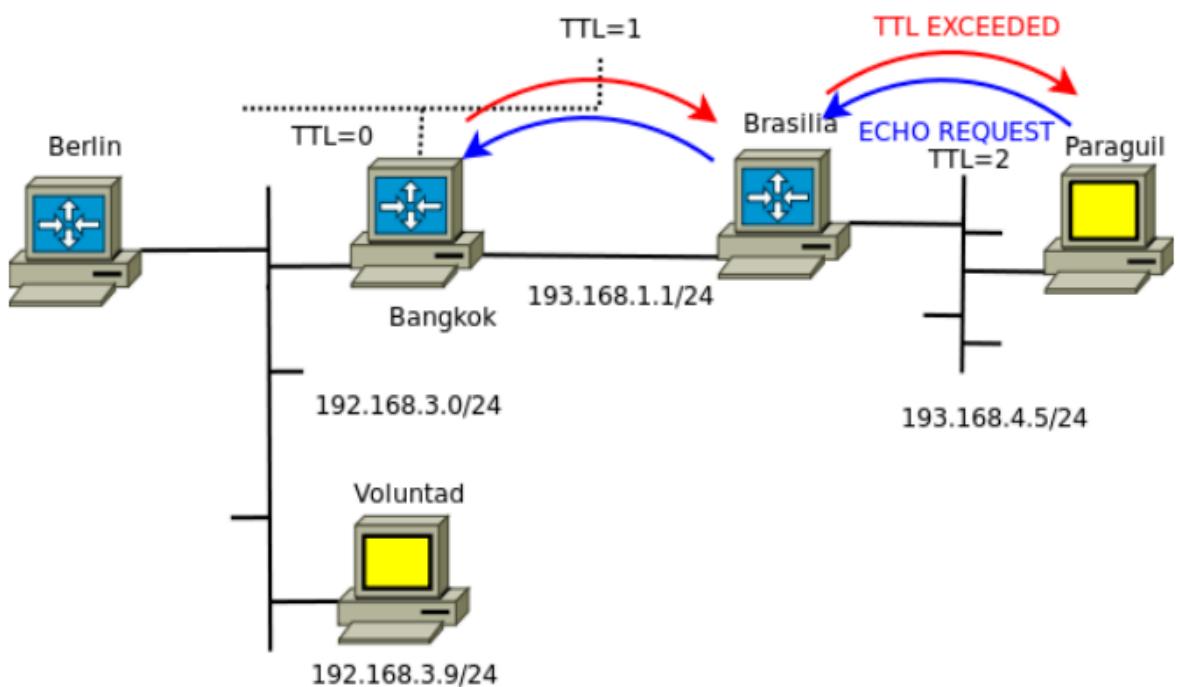
Puede salir con otro valor

Si TTL == 0, pero ya llegó a la red destino debería enviarse

Utilizado por traceroute(8) con UDP o ICMP

```
andres@h1(paraguil):~$ ping -c 2 -t 2 193.168.50.50
PING 193.168.50.50 (193.168.50.50) 56(84) bytes of
data.
From 193.168.1.2 icmp_seq=1 Time to live exceeded
From 193.168.1.2 icmp_seq=2 Time to live exceeded
From 193.168.1.2 icmp_seq=3 Time to live exceeded

--- 193.168.50.50 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100%
packet loss, time 2003ms
```

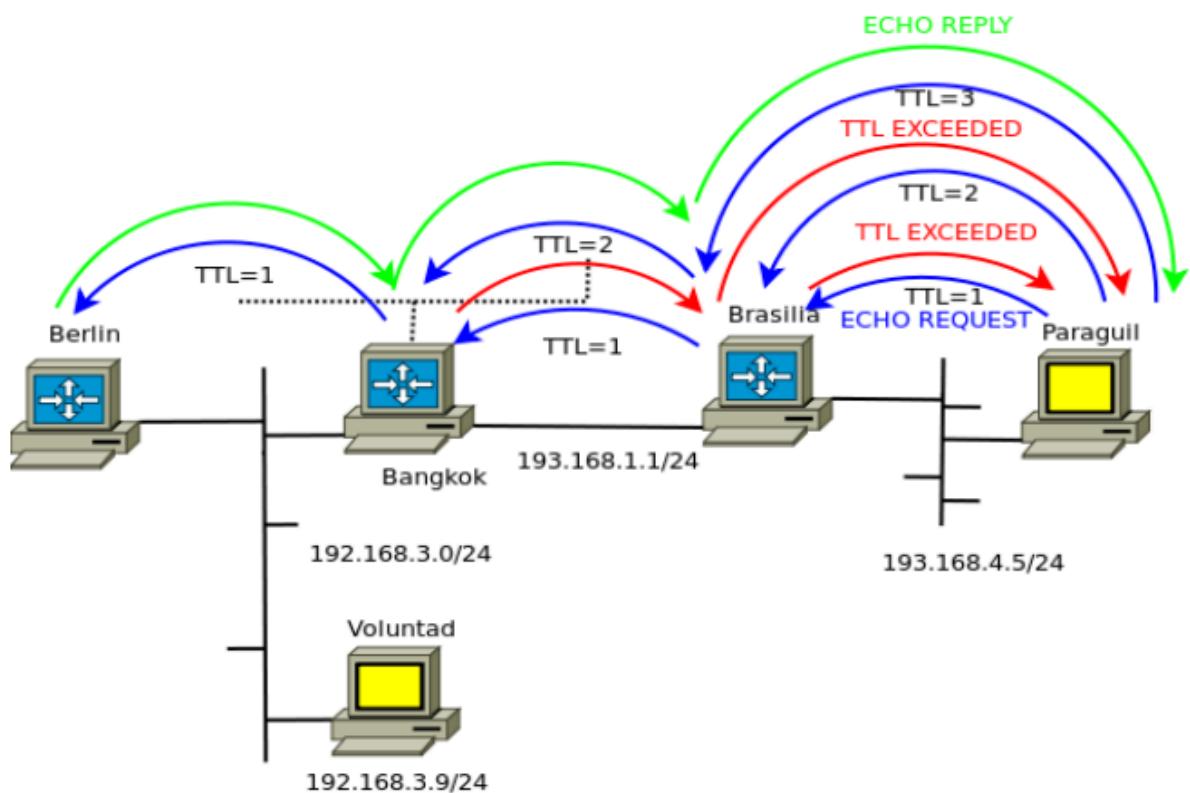


## Traceroute

En primer lugar envía un mensaje con TTL 1, y lo va aumentando cada vez que le llega un mensaje TTL exceeded. De esta manera, puede saber qué hosts están en el camino hacia el host destino (hace un trace de la ruta)

```
root@h1(paraguil)~# traceroute -n -I 193.168.3.254
traceroute to 193.168.3.254 (193.168.3.254), 30 hops max,
60 byte packets
```

```
1 193.168.4.65 3.698 ms 3.740 ms 4.405 ms
2 193.168.1.2 9.185 ms 9.809 ms 14.130 ms
3 193.168.3.254 14.973 ms 15.073 ms 17.312 ms
```



# Clase 3 - Protocolos de ruteo dinámico

Protocolo de enrutado (routed protocol):

- Define cómo enviar paquetes de datos desde un origen hasta un destino final en una red
- Requiere los servicios del protocolo de ENRUTAMIENTO/RUTEO (Routing) para construir las tablas de ruteo en cada router (gateway)
- IP es un ejemplo (no es responsable de construir las tablas de ruteo, solo usa la información disponible para enviar los paquetes)

Forwarding/Switching:

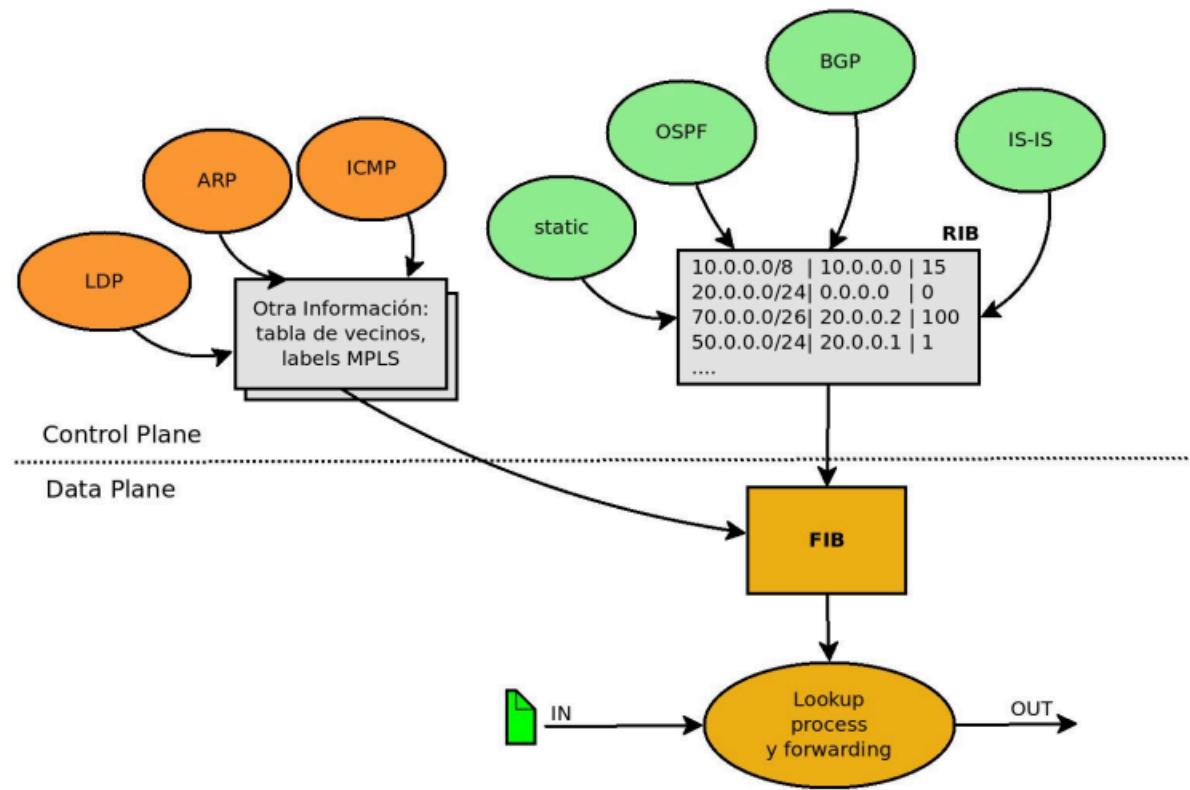
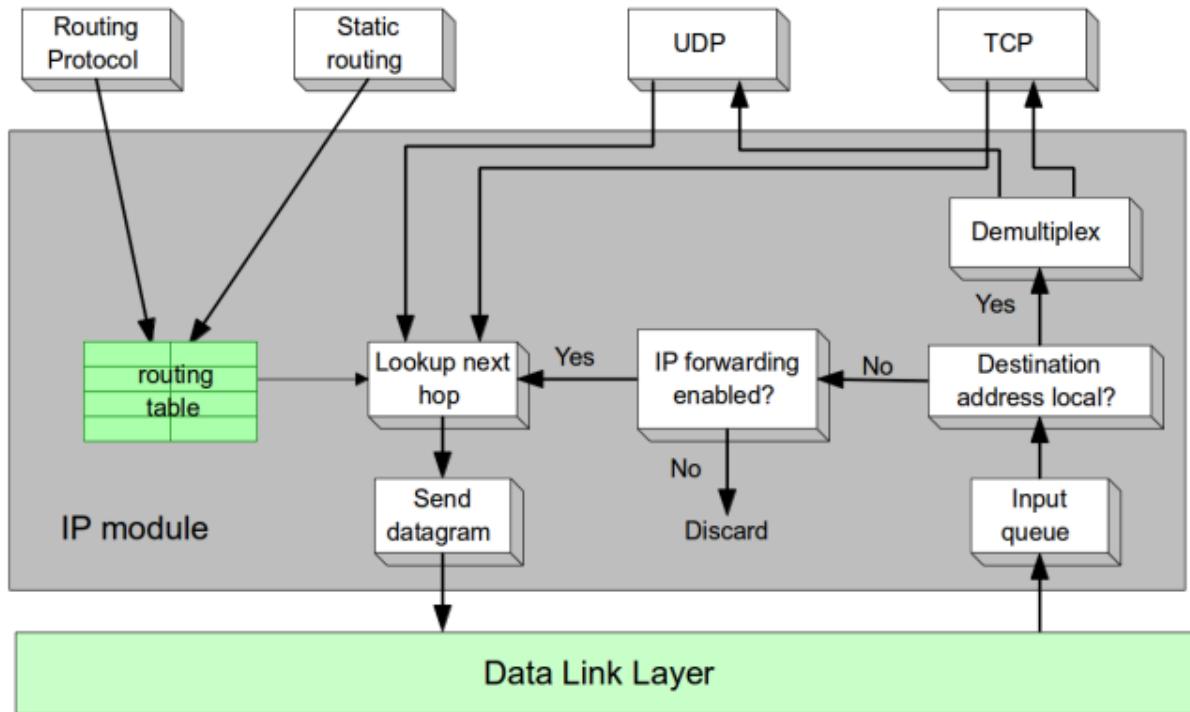
- Seleccionar un port de salida en función de la dirección de destino y tabla de ruteo
- Usado por el protocolo ENRUTADO
- Plano de Datos
- Todos los router lo hacen

Ruteo:

- Proceso mediante el cual se construye la tabla de ruteo o RIB (Routing Information Base)
- Protocolo de ENRUTAMIENTO
- Plano de Control
- Algunos routers lo hacen

FIB:

- Forwarding Information Base/Forwarding Table
- El proceso de forwarding que se hacer a partir de la RIB se optimiza generando una tabla más eficiente la FIB
- Está enfocada en acelerar el proceso de forwarding



## Routing

Decisiones de “forwarding” en IP se llevan a cabo localmente (haciendo uso de la tabla de ruteo)

Deriva en conectividad entre los diferentes puntos de la red

Se requieren recolectar y procesar un estado global

Se mantiene un estado global localmente en cada router

Los estados locales deben ser consistentes, si son inconsistentes la red no habrá convergido a un estado estable, se generan loops

Para un routing eficiente se requiere:

- Consistencia
- Completitud
- Escalabilidad

Se desea:

- Camino óptimo
- Balanceo
- Adaptabilidad

## Tipos de Routing

Todos los equipos en la red corren el protocolo ENRUTADO, por ejemplo IP (IPv4 o IPv6)

Los host no requieren correr protocolos de ENRUTAMIENTO/RUTEO

Los router requieren hacer el ENRUTAMIENTO podrían trabajar de dos formas/tipos de Routing:

- Ruteo Estático
- Ruteo Dinámico

Una red compleja: muchos routers y enlaces requiere un protocolo de ruteo dinámico

Los routers pueden participar de forma activa en el routing: reciben, generan y propagan información, los hosts lo hacen de forma pasiva (no envían ni propagan información)

## Ruteo estático

Las rutas son establecidas por el administrador manualmente  
Propenso a errores  
Si se cambia la topología requiere cambios manuales en los routers  
Sirve cuando se tiene una red sencilla  
No tiene problemas de seguridad ni de incompatibilidad  
No implica costo de procesamiento extra  
Mayor control  
Esquema NO escalable y NO tolerante a fallos

## Ruteo dinámico

Requiere una configuración inicial por el administrador  
Si se cambia la topología se adapta de forma automática  
Facilita mantenimiento cuando se tiene una red compleja  
Implica costo de procesamiento extra  
Esquema escalable y tolerante a fallos  
Resolución de Problemas/Debugging, más complejo  
Caminos “óptimos” de acuerdo a la información manejada por el protocolo (métrica, costo)

## Routing domain

Seleccionamos el/los protocolo/s de Ruteo en un Routing Domain, conjunto de routers con Routing Protocols comunes

Cada routing domain tiene uno o varios protocolos de ruteo internos (IGP)

Los routers que comparten el mismo protocolo de ruteo forman una "familia" de enrutamiento, donde las rutas son distribuidas y calculadas según las reglas del protocolo común

Uno o más de estos incluidos en un AS

## Autonomous System (AS)

Conjunto de redes bajo la misma administración (podría ser gestionada por más de un operador de red), y utilizando un protocolo de ruteo o combinaciones para rutear internamente, independientemente de la red de su proveedor

Clara y única política de ruteo

Para comunicarse con otros AS se usan protocolos de ruteo externos (EGP)

Cada AS (Sistema Autónomo) en Internet (necesidad de intercambiar tráfico con otros Dominios) debe tener un número identificador: ASN (AS Number). Relacionado con BGP, otorgado por los RIRs, LACNIC, RIPE, ARIN, AFRINIC, ARIN

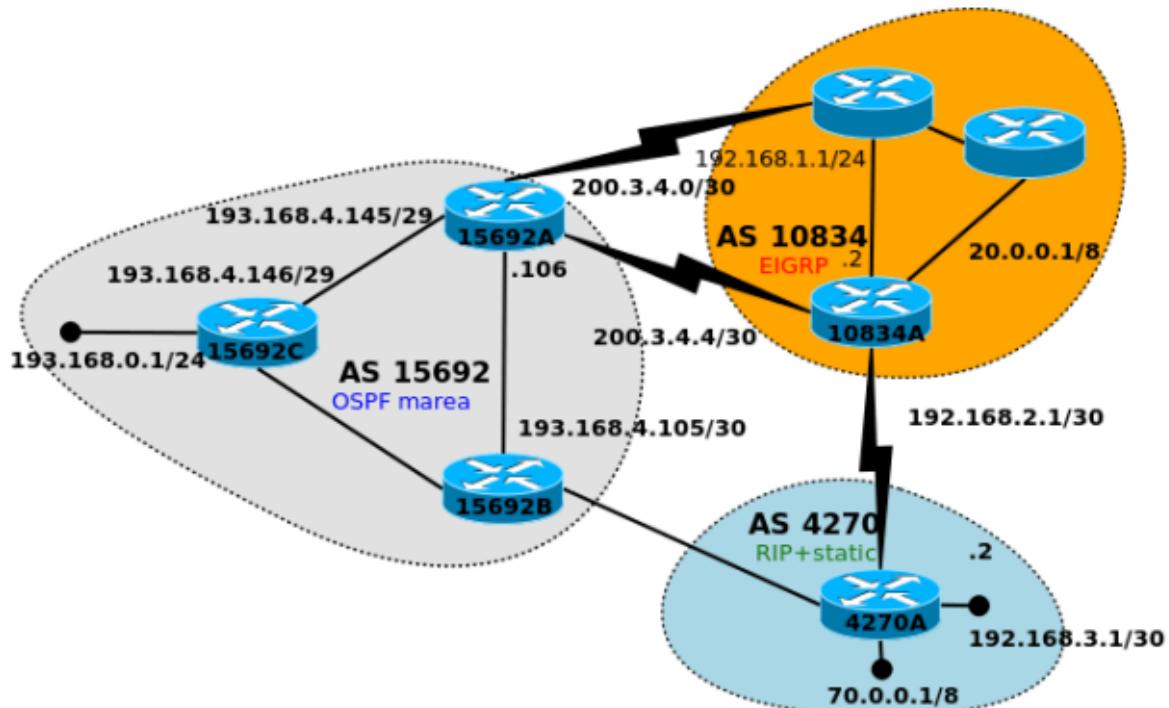
## Protocolos de ruteo dinámico

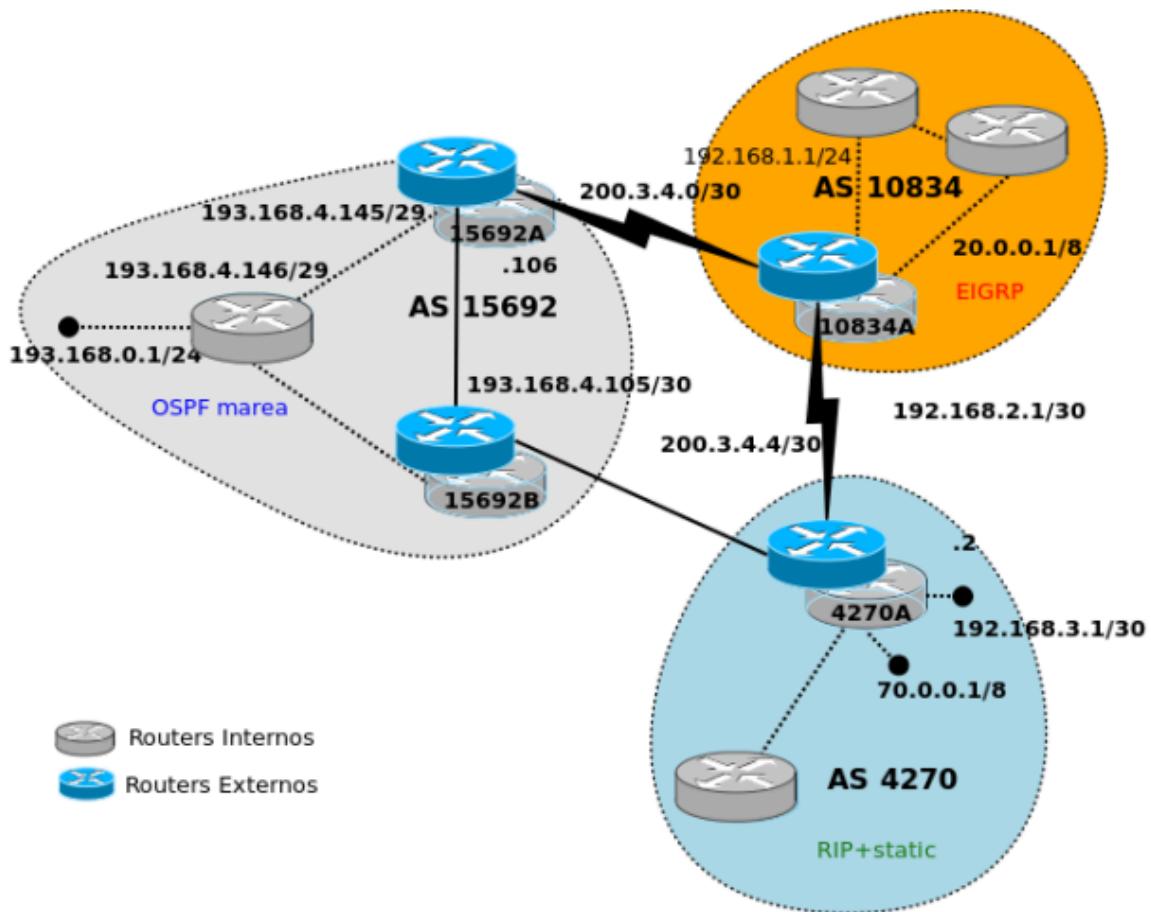
IGP (Interior Gateway Protocols), trabajan dentro del mismo AS:

- RIP (Routing Internet Protocol) v1, v2 (estándar IETF)
- IGRP e EIGRP ([Enhanced] Interior Gateway Routing Protocol) (propietarios de cisco)
- OSPF (Open Shortest Path First) v2, v3 (estándar IETF)
- IS-IS (Intermediate System to Intermediate System) (estándar de la ISO)

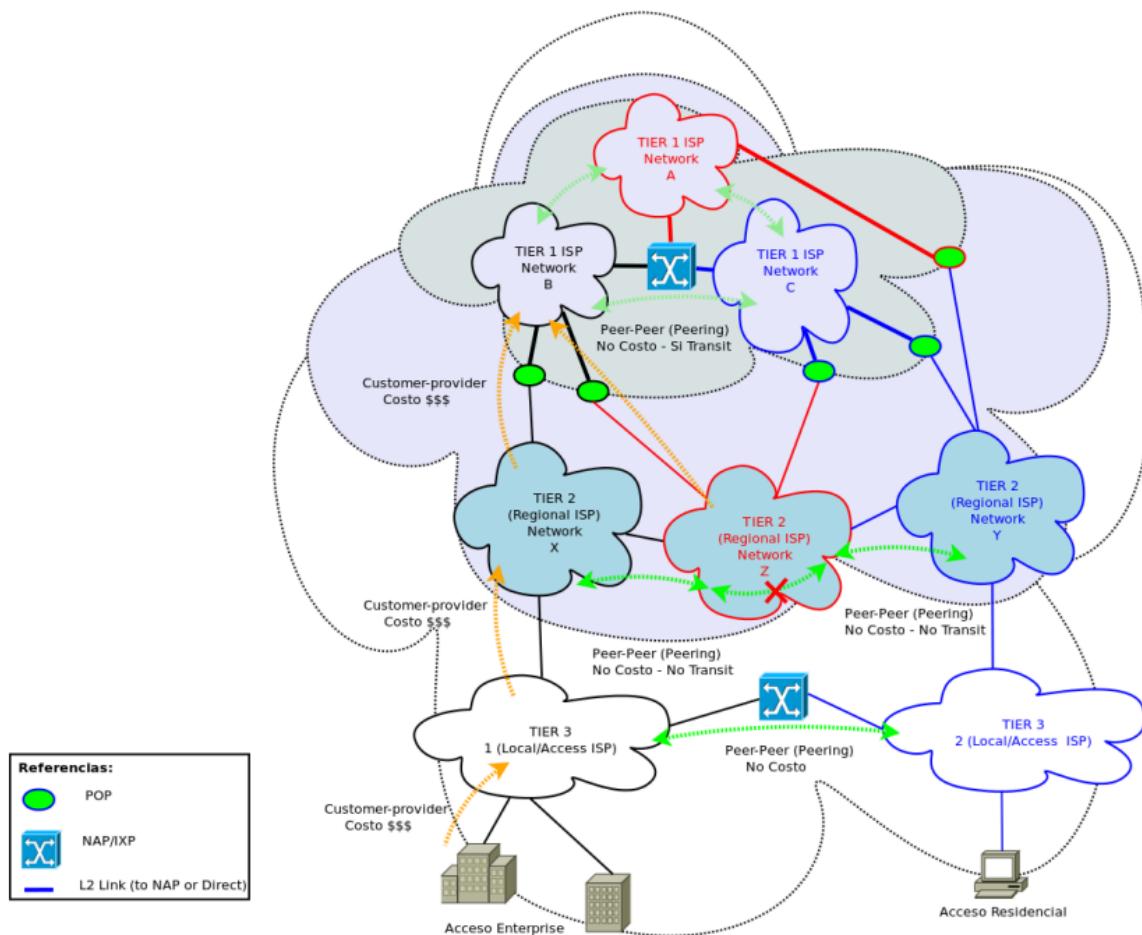
EGP (Exterior Gateway Protocols), trabajan entre diferentes AS:

- GGP (Gateway to Gateway Protocol) (antecesor)
- EGP (Exterior Gateway Protocol) (estándar IETF, en desuso)
- BGP (Border Gateway Protocol) (estándar IETF)





# Estructura de Internet



## Otra clasificación

- Protocolos de DV (Vector de Distancia)
  - RIP, v1, v2
  - IGRP
  - GGP
- Protocolos de PV (Vector de Camino)
  - BGP
  - EGP
- Link State (Estado de Enlace)
  - OSPF
  - IS-IS
- Vector de Distancia Avanzado (Advanced VD) (considerado Híbrido)

- EIGRP

# Clase 4 - DHCP

Un host para conectarse a una red IP requiere 3 parámetros + 1:

- Para conectarse a la red local:
  - Dirección IP
  - Máscara de red
- Para conectarse a otras redes: Router por default (Default Gateway)
- Para usar servicios: Servidor(es) de DNS

Forma de obtener los parámetros:

- Configuración manual/estática:
  - Difícil de mantener
  - No escalable (recolección, re-assign.)
  - No sirve para movilidad
- Configuración dinámica:
  - RARP
  - ICMP
  - BOOTP
  - DHCP

## DHCP

Dynamic Host Configuration Protocol

Protocolo de L3

Protocolo “Helper” de IP

Al estar montado sobre UDP se lo suele considerar protocolo de nivel de aplicación

Tanto para IPv4 o IPv6

Permite la configuración dinámica de los parámetros de red de los hosts

Definido en RFC-2131

Los host al arrancar solo tienen acceso a su red local de forma broadcast

En la red local existe uno o más servidores de auto-configuración: DHCP servers

Los host sin parámetros de red envían requerimiento

Los servidores los atienden asignando los valores que brindan conectividad

El parámetro se reserva por un tiempo

## Mensajes

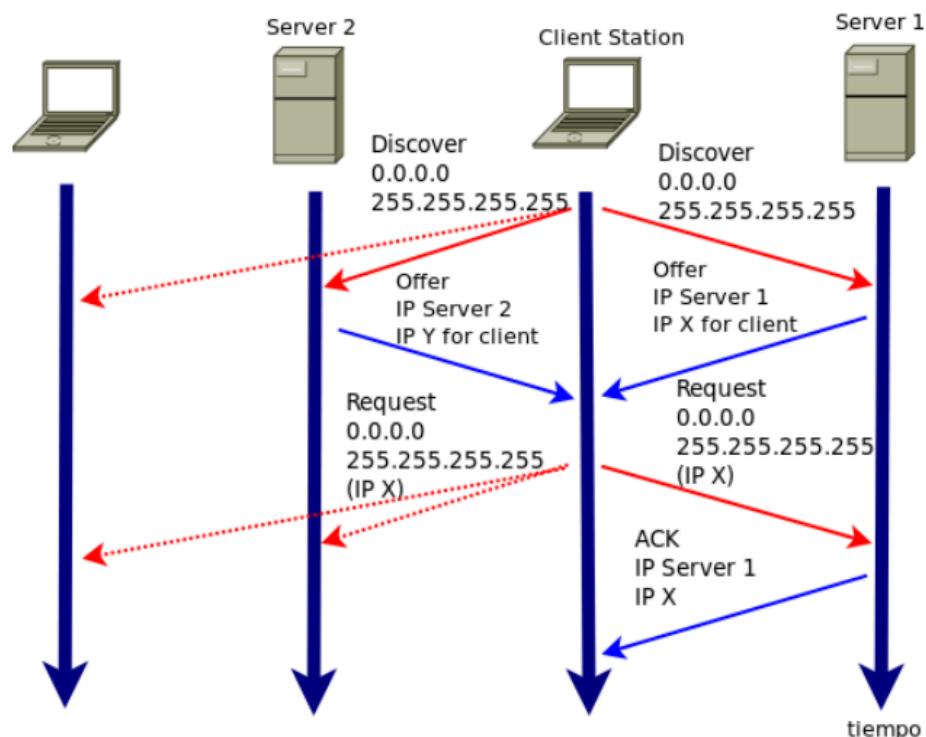
Algunos Mensajes DHCP:

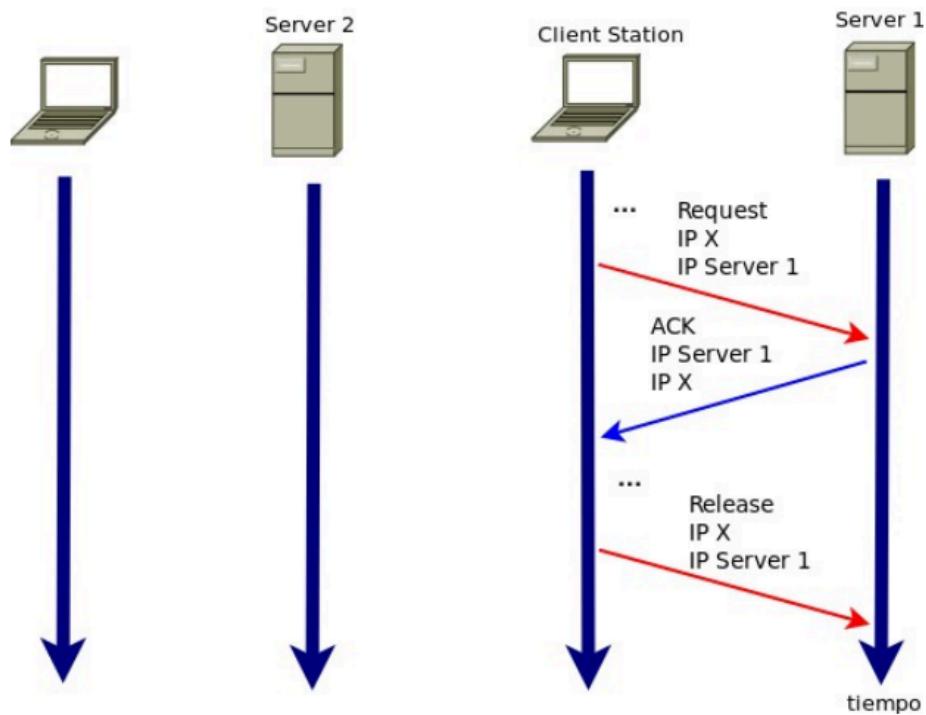
- Discover (broadcast)
- Offer (unicast/broadcast en general se envía unicast, pero debido a que pueden existir equipos que no procesan mensajes unicast antes de tener configurada la dirección IP completa, se podrían enviar en forma broadcast)
- Request (broadcast)
- ACK
- Release
- NAK

Montado sobre UDP:

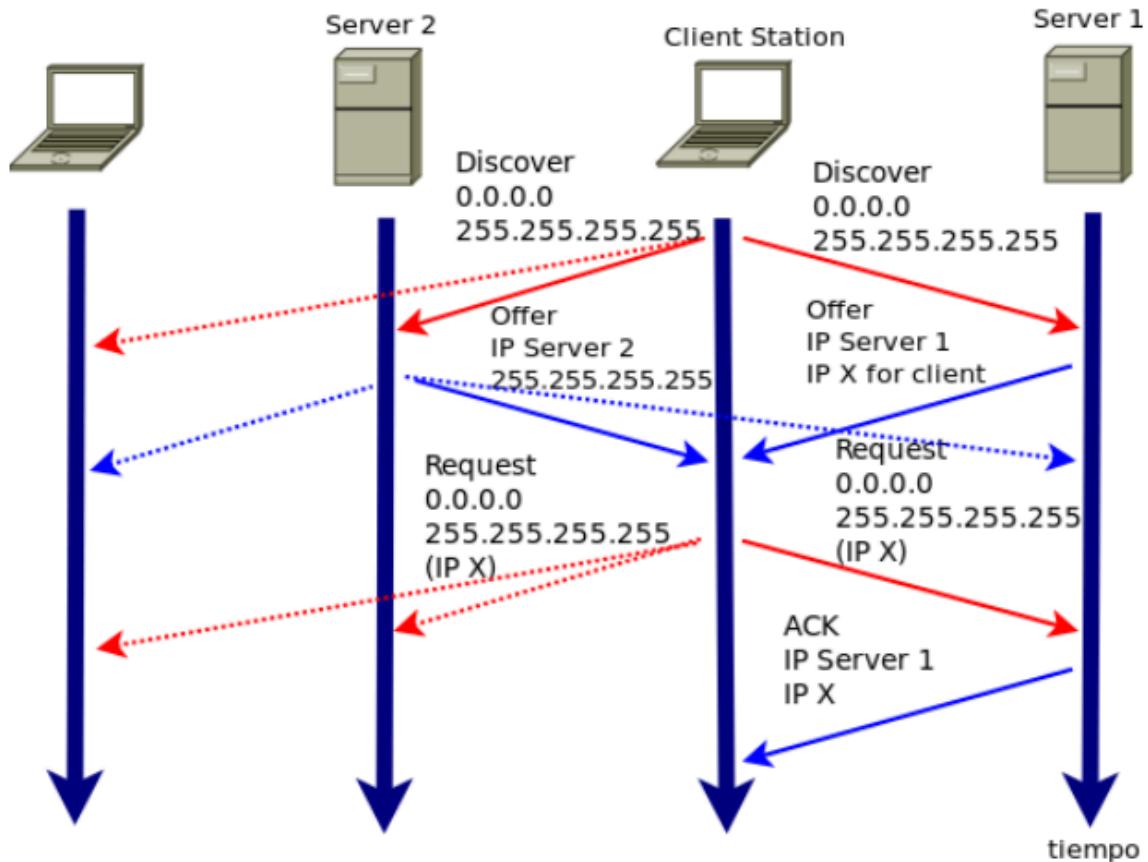
- Bootpc (client) 68
- Bootps (server) 67

## Ejemplo





Source	Destination	Port	Protocol	Time	Comment
0.0.0.0 255.255.255.255	10.0.2.2	10.0.2.15	CadmusCo_21:2c:e0	RealtekU_12:35:0	
DHCP Discover - Tra					DHCP: DHCP Discover - Transaction ID 0xb947b252
DHCP Request - Tra					DHCP: DHCP Offer - Transaction ID 0xb947b252
					DHCP: DHCP Request - Transaction ID 0xb947b252
					DHCP: DHCP ACK - Transaction ID 0xb947b252
					ARP: Who has 10.0.2.2? Tell 10.0.2.15
					ARP: 10.0.2.2 is at 52:54:00:12:35:02
					DNS: Standard query A www.google.com
					DNS: Standard query response A 74.125.234.116 A 74.125.234.
					ICMP: Echo (ping) request id=0xfe0d, seq=1/256, ttl=64
					ICMP: Echo (ping) reply id=0xfe0d, seq=1/256, ttl=63
					DNS: Standard query PTR 116.234.125.74.in-addr.arpa
					DNS: Standard query response PTR gru03s08-in-f20.1e100.net
					ICMP: Echo (ping) request id=0xfe0d, seq=2/512, ttl=64
					ICMP: Echo (ping) reply id=0xfe0d, seq=2/512, ttl=63
					DNS: Standard query PTR 116.234.125.74.in-addr.arpa
					DNS: Standard query response PTR gru03s08-in-f20.1e100.net
					ICMP: Echo (ping) request id=0xfe0d, seq=3/768, ttl=64
					ICMP: Echo (ping) reply id=0xfe0d, seq=3/768, ttl=63
					DNS: Standard query PTR 116.234.125.74.in-addr.arpa
					DNS: Standard query response PTR gru03s08-in-f20.1e100.net
					DHCP: DHCP Request - Transaction ID 0x1368c207
					DHCP: DHCP ACK - Transaction ID 0x1368c207
					ARP: Who has 10.0.2.2? Tell 10.0.2.15
					ARP: 10.0.2.2 is at 52:54:00:12:35:02
					DHCP: DHCP Release - Transaction ID 0xd089ee55



## Proceso

Cuando un host se conecta a una red, como no tiene dirección IP ni configuración de red, envía un mensaje broadcast discovery solicitando configuración

Un servidor DHCP de la red recibe la solicitud y asigna una configuración válida al host (mediante el mensaje offer donde le propone la dirección IP y otros parámetros)

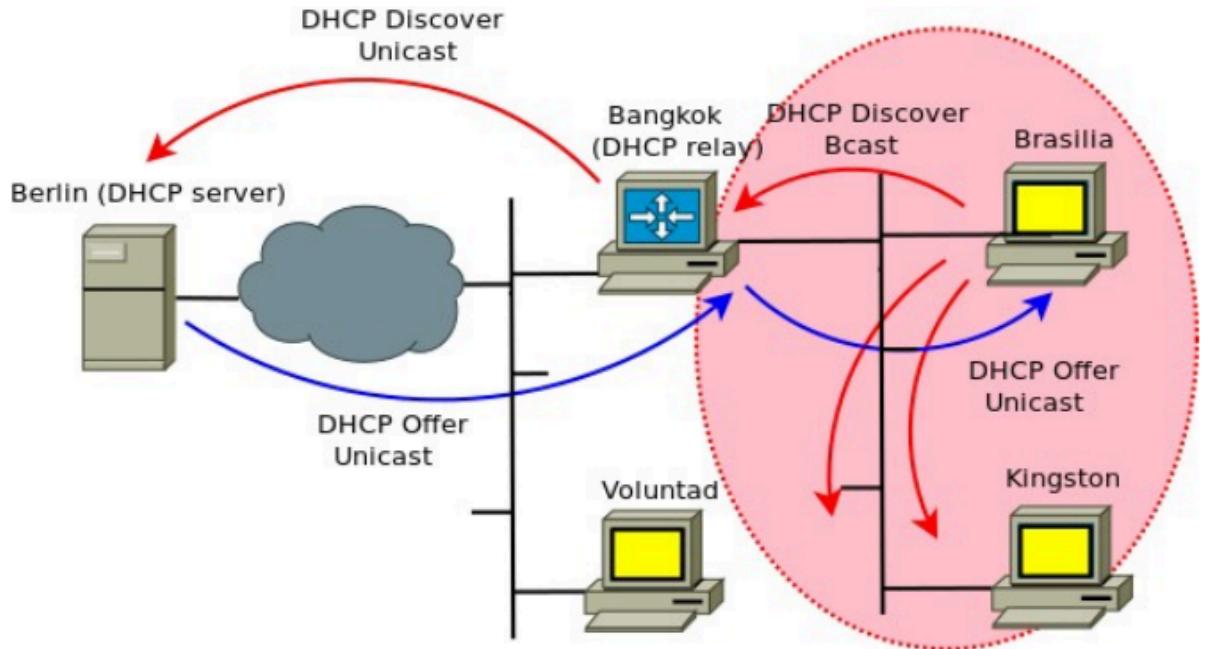
El host selecciona una de las ofertas y envía un mensaje request para confirmar la elección

El servidor DHCP envía un ACK confirmando la asignación y proporcionando la configuración definitiva

La asignación se hace por un tiempo limitado después del cual el host debe renovar su configuración

## DHCP relay

Los routers pueden funcionar como agentes DHCP Relay y enviar los mensajes de DHCP broadcast de forma unicast a helper (DHCP server)



# Clase 5 - NAT

## Problemas con IPv4

IPv4 tiene el espacio de direcciones “casi” agotado

Soluciones temporales:

- CIDR: Tablas de ruteo
- DHCP: direcciones escasas, facilidad de administración
- NAT: direcciones escasas

Solución definitiva: IPv6

## NAT (Network Address Translation)

Traducción de direcciones de un espacio privado (no “enrutable” en Internet) a un espacio público

Direcciones Privadas:

- Clase A: 10.0.0.0/8
- 16 Clases B: 172.16.0.0/12
- 256 Clases C: 192.168.0.0/16

## Problemas con IP Privadas

No son únicas, por lo tanto:

- Las rutas pueden ser confundidas
- Habitualmente son filtradas por routers de borde
- Algunos protocolos no funcionan adecuadamente, FTP, VoIP, etc.

## Procesos de Traducción

Se realizan sobre redes stubs (solo una salida)

Se deben mantener tablas de translaciones

Varias formas de realizarlo:

- NAT (Network Address Translation):

- Estático
- Dinámico
- NAPT (Network Address Port Translation):
  - Dinámico sobre pool
  - Dinámico sobre dirección overload/masquerade

Modificación de direcciones, ports, checksums

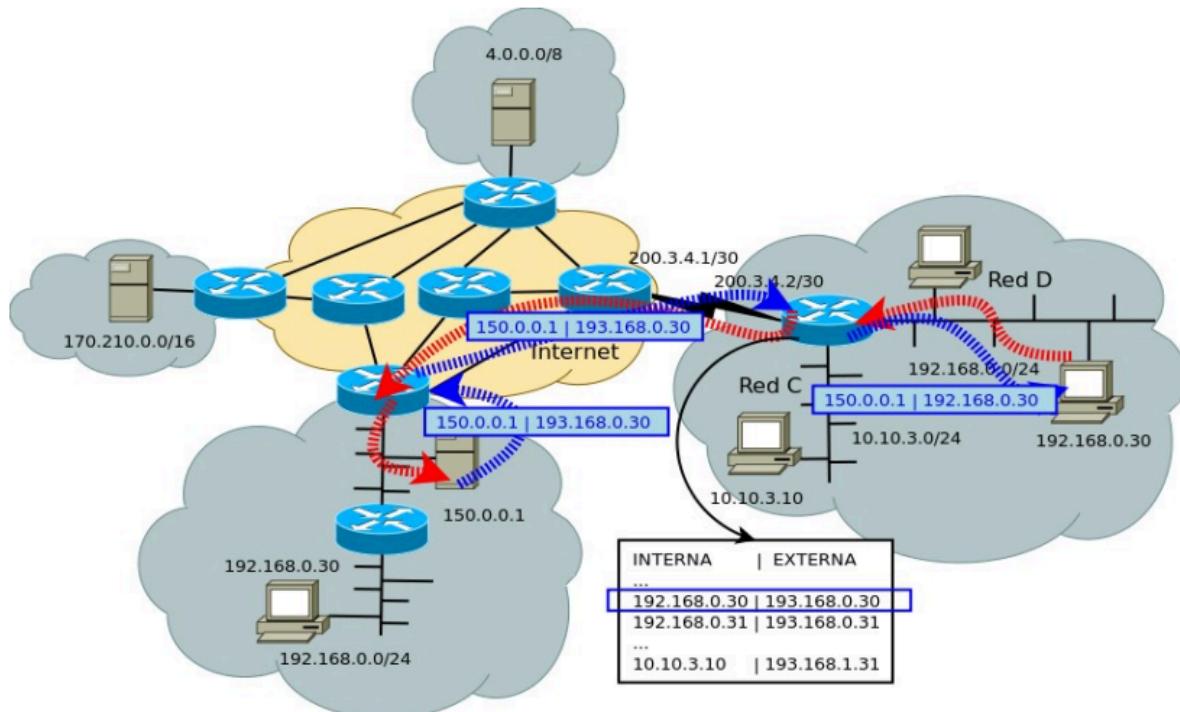
## NAT básico

Una forma de realizarlo es: “one-to-one” (uno a uno)

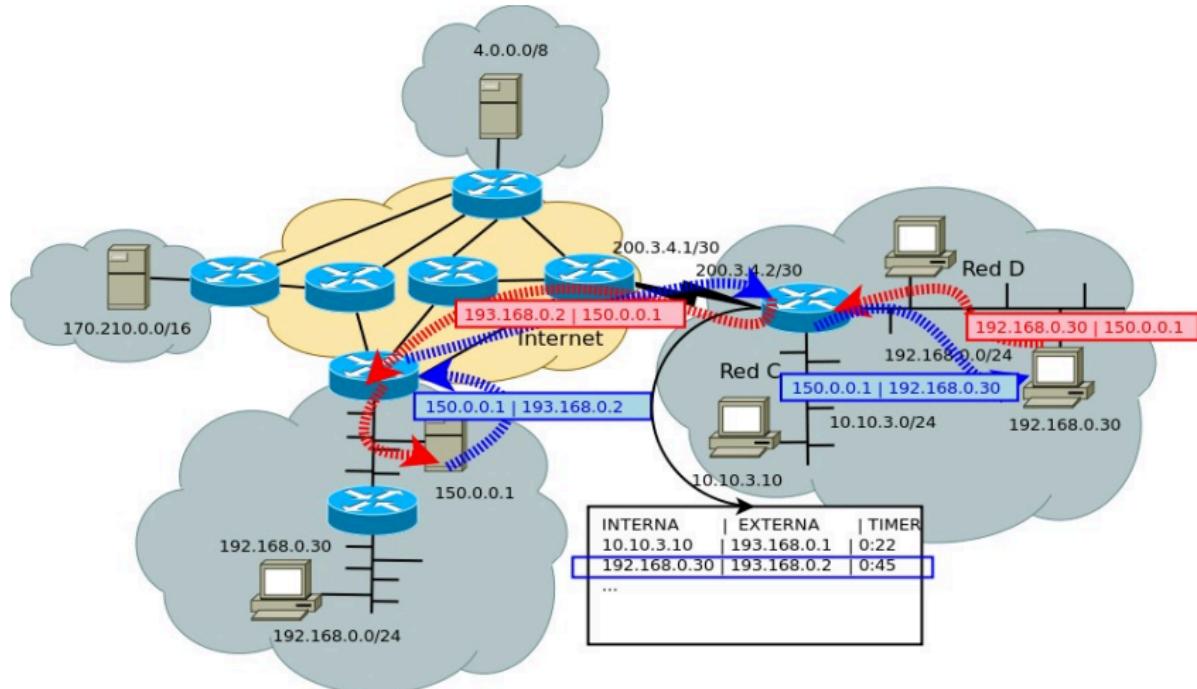
NAT básico:

- Se mapea una dirección IPv4 privada a una dirección IPv4 pública
- Si se hace de forma estática requiere tantas direcciones públicas como privadas
- Permite acceso en ambas direcciones
- Si se hace de forma dinámica no es necesario, pero sí se requiere un timer por cada entrada
- Limita acceso simultáneo de acuerdo al pool pub

## Ejemplo estático



## Ejemplo dinámico



## NAPT

NAT no es implementable cuando se tiene un pool chico de direcciones o no se posee direcciones públicas asignadas

En ese caso se debe trabajar con campos de la capa de transporte o del payload.

NAPT es conocido como PAT (Port Address Translation): "one-to-many"

Se utilizan los puertos de los protocolos u otros valores como ICMP Identifier para resolver el mapeo

Se pueden usar timers y sesión del protocolo

En la tabla de traslaciones se mantienen el protocolo y los puertos origen y destino

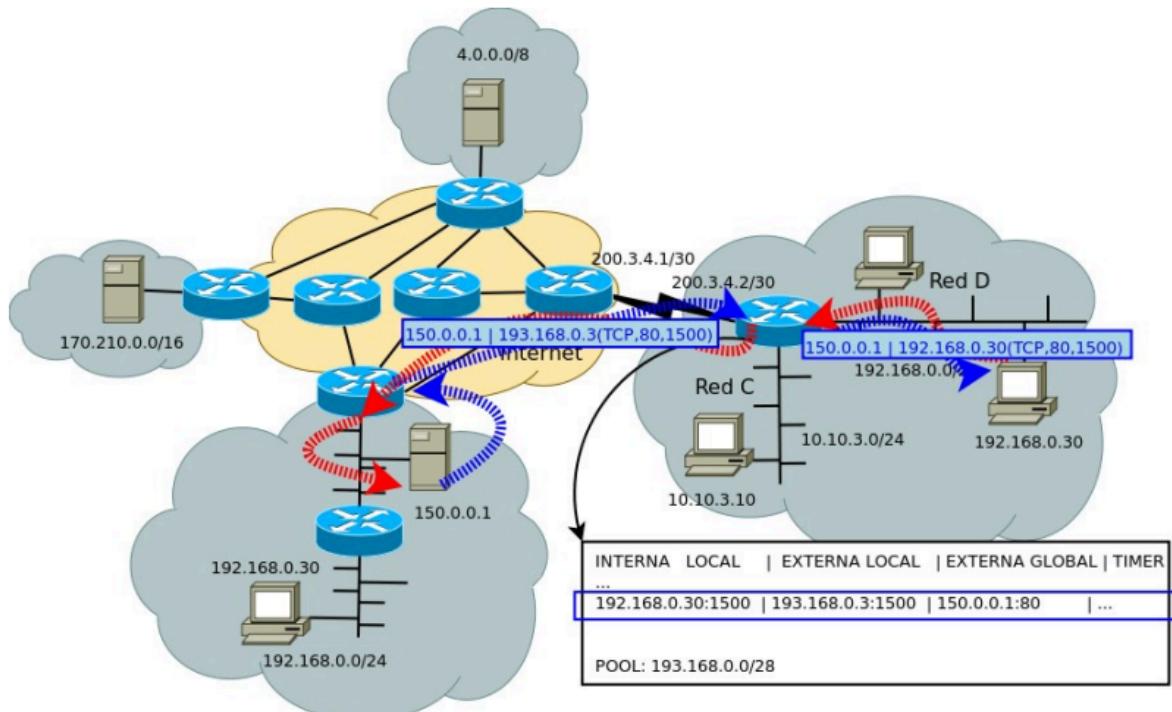
Se intenta conservar el puerto origen, pero si está "ocupado" se debe reemplazar por otro

El dispositivo debe "violar" los límites impuestos por la división en capas

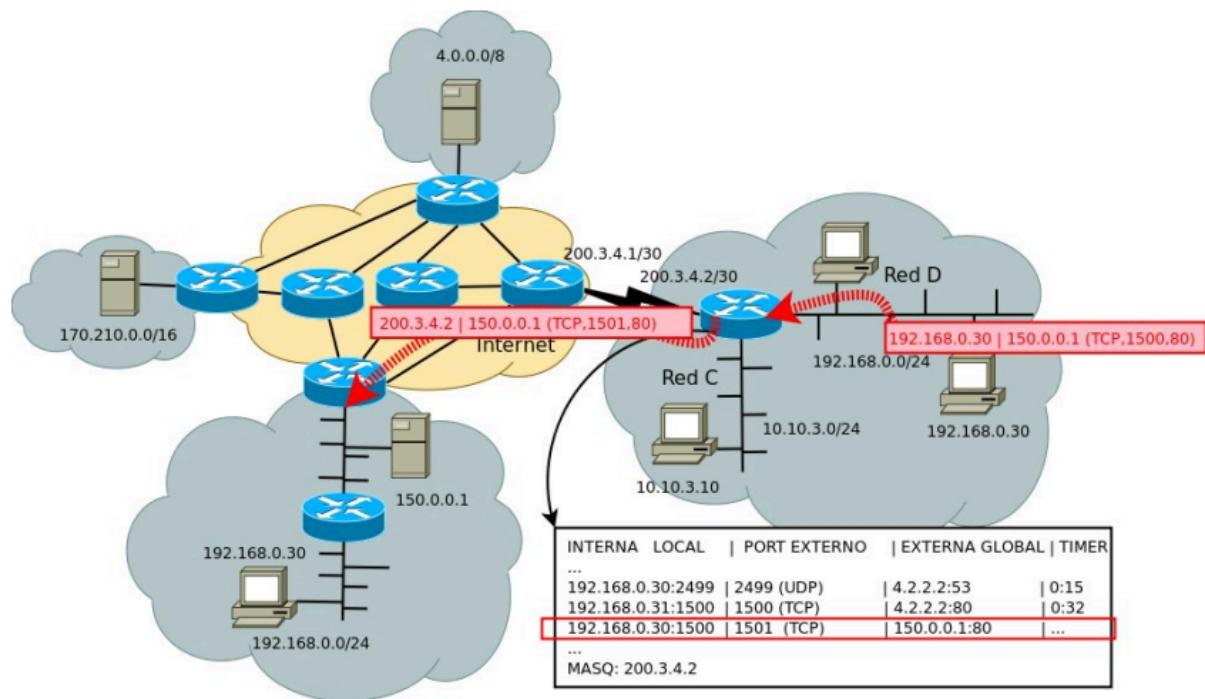
Dos alternativas:

- Utilizando un pool y haciendo PAT sobre este
- Utilizando la dir. IP externa y haciendo overloading/masquerading sobre esta

## Ejemplo con pool



## Ejemplo con overload/masquerade



## Port forwarding

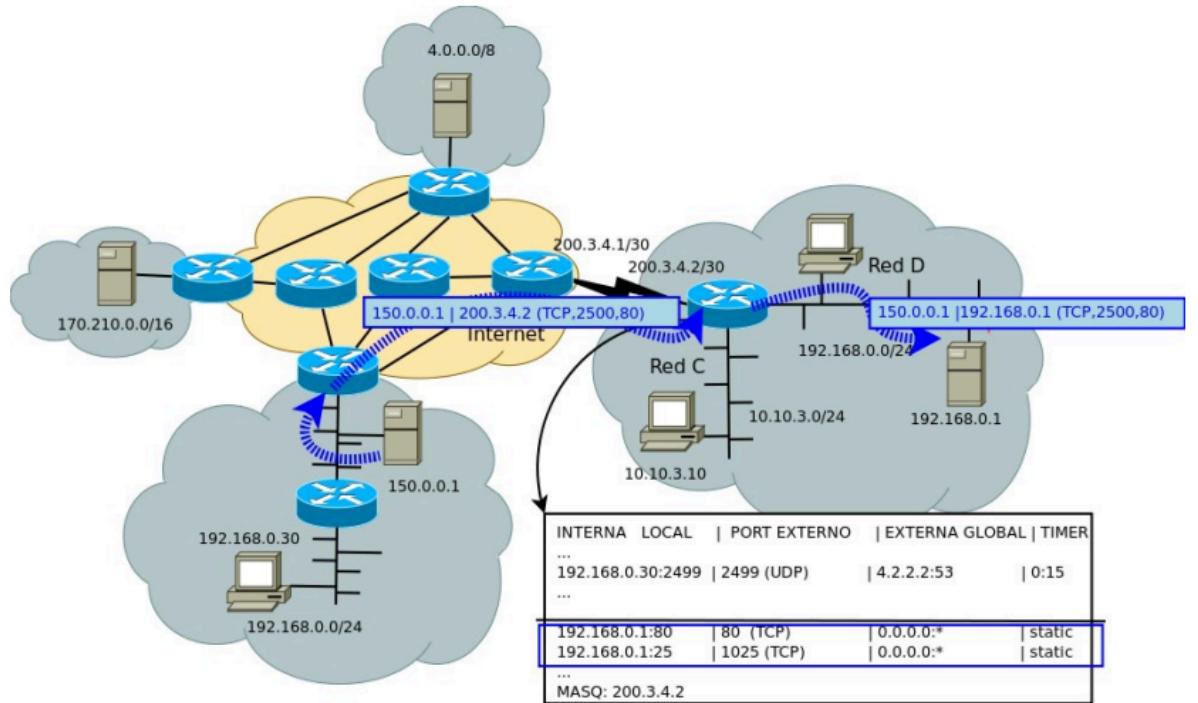
Overloading/Masq no permiten acceso desde “afuera” hacia “adentro”

Solo se permiten entrar tráfico de conexiones generadas internamente

Mediante Port Forwarding (Reenvío de puerto) se permite poder tener servicios en una red privada accesibles desde “afuera”

No se requiere NAT estático, se implementa con NAPT y mapeo reverso estático de puertos

## Ejemplo



## Conclusiones

NAT/NAPT resuelve temporalmente la escasez de direcciones IPv4

Algunos servicios no funcionan

Da una “sensación” de seguridad, aunque no siempre es verdad

Se pierde la idea de IP end-to-end

Firewalls más complejos

# Clase 6 - IPv6 - Parte 1

## Comutación de paquetes

Es el modelo que usa el protocolo IP

Modelo de Red, L3: Comutación de Paquetes

Cada PDU: Unidad de datos, datagrama/paquete puede ser “transportado” por la red de forma independiente

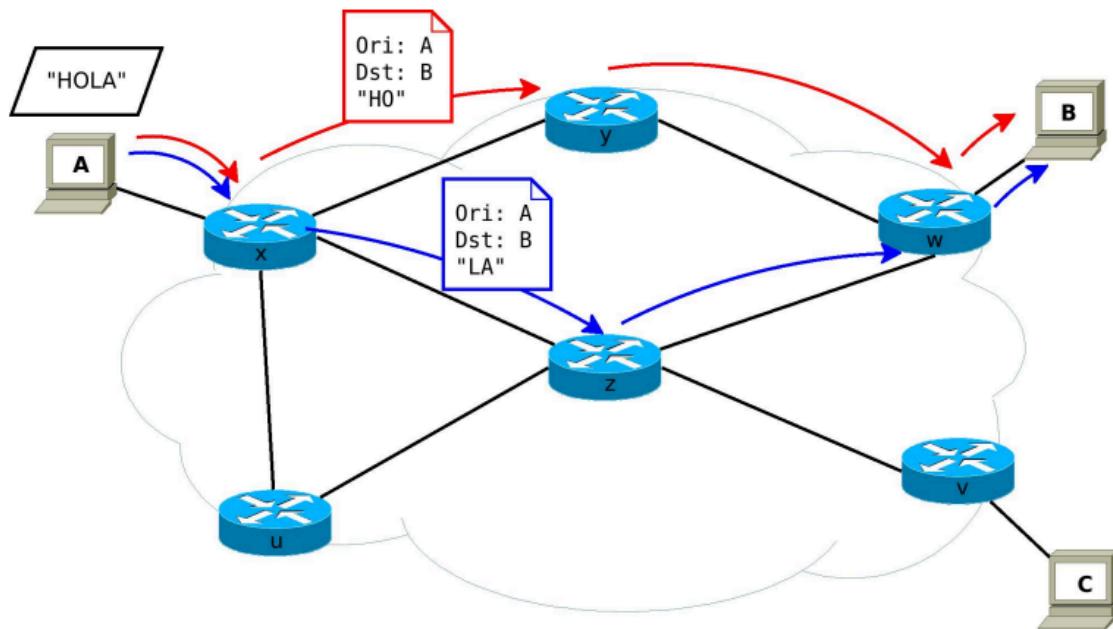
Los datagramas tienen información en su encabezado para ser “manejados” por los dispositivos intermedios (la red)

Componentes/dispositivos intermedios de la red: comutadores/routers/gateways

Routers trabajan básicamente en Store & Forward y se interconectan entre sí físicamente (a nivel de enlace)

El ruteo se produce, hop-by-hop (salto a salto)

Modelo best-effort, más flexible y eficiente



## Problemas con NAT

Se vuelve compleja la red, dispositivos intermedios

Dificultades:

- Acceso directo a red privada
- Protocolos Peer-to-Peer
- Problemas protocolos particulares: FTP, VoIP (SIP, RTP), VPN (IPSec), on-line gaming

Agregados (Parches):

- Port-Forwarding, UPnP (Universal Plug & Play)
- ISP requieren CGN (Carrier Grade NAT), NAT444, LSN: 100.64.0.0/10 (RFC-6598)
- STUN (Session Traversal Utilities for NAT), NAT Traversal

## Problemas en IPv4

- Direcciones IPv4 no disponibles, uso de NAT
- Tablas de ruteo muy grandes en el backbone de Internet
- Congestión en los routers, demasiado procesamiento

Otras cuestiones no contempladas desde el inicio

- Seguridad a nivel L3, IP
- Extensiones al modelo de Calidad de Servicio (QoS)
- Fácil auto-configuración y re-numeración de direcciones
- Movilidad a nivel de red no contemplada en el diseño del protocolo.

## Beneficios de IPv6

- IPv4 e IPv6 NO son versiones del mismo protocolo
- Mayor espacio de direcciones - 128 bits:
  - 340.282.366.920.938.463.463.374.607.431.768.211.456 direcciones
- Formato de cabecera simplificado
- Menor overhead de procesamiento
- Ordenar las tablas de enrutamiento
- Conectar todo, usar autoconfiguración de direcciones (plug and play)
- Arquitectura de red jerárquica para un ruteo eficiente
- Seguridad a nivel IP (IPSec obligatorio)

- Jumbogramas, size(datagrama) > 64KB
- Movilidad y más direcciones de multicast

## Cambios en IPv6

Definido RFC-8200 (STD-86) 2017 (hace obsoleta RFC-2460)

Direcciones más largas

Datagramas de 40 bytes (contra 20 bytes + opt, max 60B)

Simplifica cabecera:

- Se saca la fragmentación, se deja solo de extremo a extremo como opción
- Se saca checksum de cabecera
- Header de tamaño fijo. No existen más las Opciones
- Flow Label: identificador de flujo (20 bits)
- Se renombran los campos:
  - Traffic Class (ex ToS)
  - Hop Limit (ex TTL)
  - Next Header (ex Protocol)
- Cabeceras de extensión

Ver.	TrafficClass	Flow Label		
	Payload Length	Next Header	Hop Limit	
<b>128 bit</b> <b>Source Address</b>				
<b>128 bit</b> <b>Destination Address</b>				

Ver.	header	TOS	total length	
identification		flag	fragment offset	
TTL		Protocol	Checksum	
32 bit Source Address				
32 bit Destination Address				

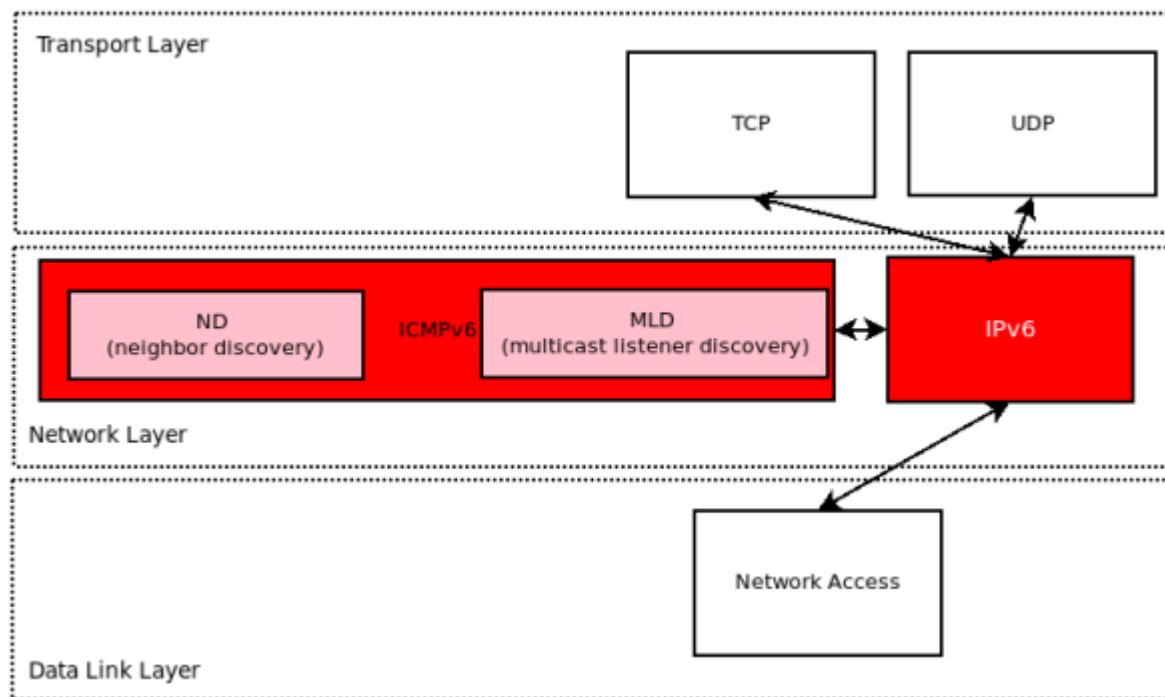


removed  
changed



## Funcionalidad de IPv6

- Direccionamiento
- Ruteo/Forwarding
- Mux/Demux de protocolos superiores
- Fragmentación (ya no)
- Evitar loops
- Detección de errores en header (ya no)



## Servicios nuevos

IGMP pasa a ser MLD

ARP pasa a ser ND

Descubrimiento de Vecinos (NDP):

- ND propiamente
- Router discovery y autoconfiguración

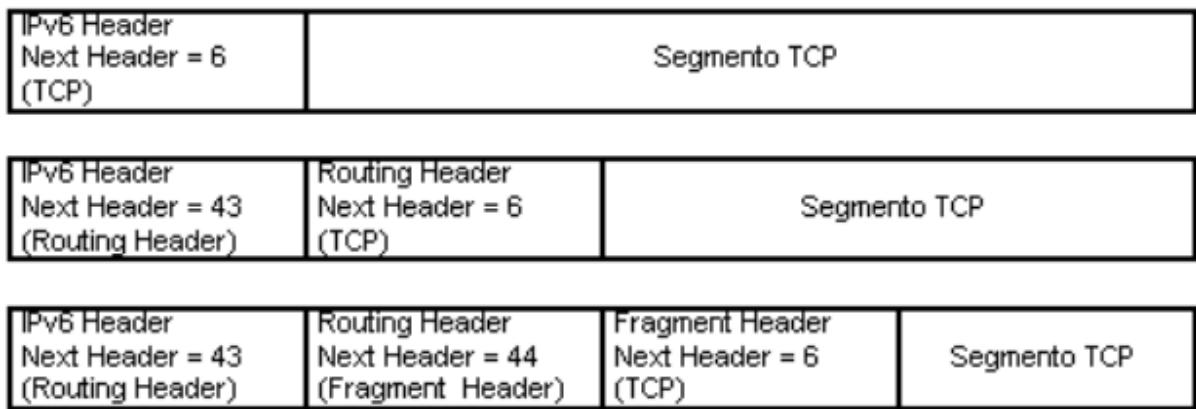
Manejo de Grupos de Multicast

## Cabeceras de extensión

Permite la extensibilidad del protocolo

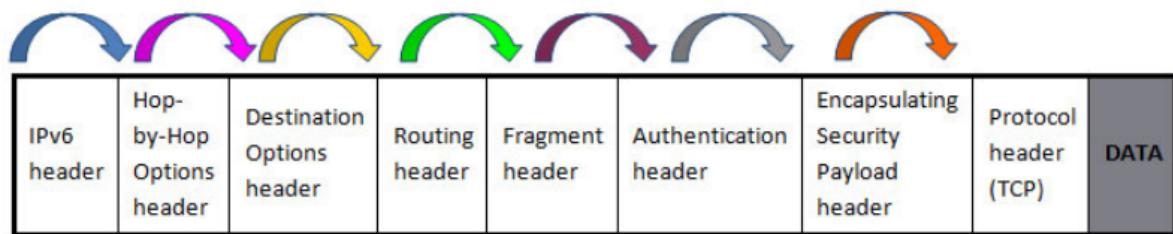
Se encuentran a continuación del header

En general, son procesadas por los extremos



## Orden de las cabeceras de extensión

- Hop-by-hop: procesado por cada router
- Dest Opt: procesado por routers incluidos
- Routing: procesado por routers, RH0 desaconsejado
- Frag, Auth, Sec, Dest. procesado por extremos



## Más direcciones disponibles

Más direcciones para más gente, más dispositivos, nuevas tecnologías, por ejemplo para IoT

Acuerdo:

- Direcciones de longitud fija de 128 bits
- Varias direcciones por interfaz
- Direcciones con diferentes alcances y tiempos de vida

## Tipos de direcciones:

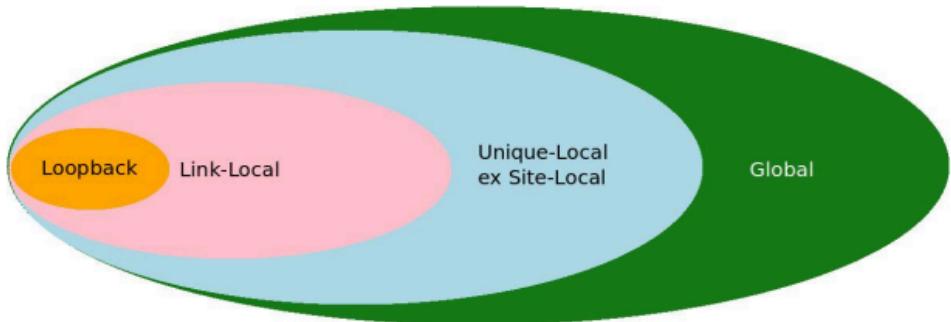
- Unicast

- Anycast (tomadas del rango Unicast)
- Multicast (no hay direcciones broadcast): FF00::/8

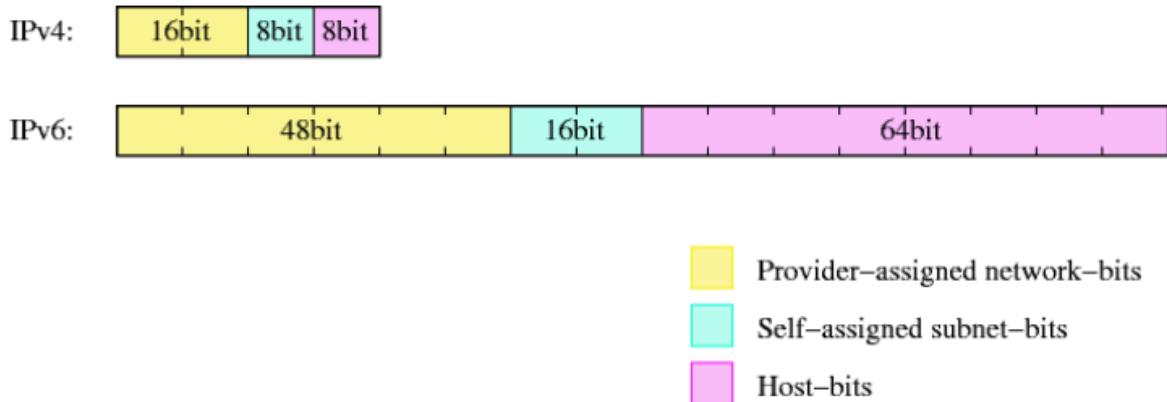
## Alcance (Scope) de las direcciones Unicast

- Locales (Link-local): FE80::/64
- De sitio site-local (desaconsejadas), unique-local
- Compatibilidad ipv4-compat (desaconsejadas), ipv4-mapped
- Globales: 2000::/3

- Link-local.
- Site-local.
- Unique-Local.
- IPv4-Compatible.
- IPv4-Mapped.
- Global.



## Formato de las direcciones



128 bits

Unicast: separadas en red, [subred] y host

No hay clases de direcciones

## Notación Direcciones IPv6

Hexadecimal en grupos de 16 bits, separadas por ":"

nnnn:nnnn:nnnn:ssss:hhhh:hhhh:hhhh:hhhh/prefix length

Ceros al inicio de cada grupo se pueden obviar:

2001:0db8:1011:0001:36ed:04ff:fe32:0076 == 2001:db8:1011:1:36ed:4ff:fe32:76

Ceros contiguos se puede eliminar con "::" (sólo se puede utilizar una vez):

- 2001:db8:1011:1:0:0:0:1 == 2001:db8:1011:1::1
- 2001:db8:0:0:1011:0:0:1 == 2001:db8:0:0:1011::1
- 2001:db8:0:1011:0:0:0:1 == 2001:db8:0:1011::1

Se utilizan "[" "]" para indicar port en URL: [http://\[2001:db8:1011:1:0:0:0:1\]:8080](http://[2001:db8:1011:1:0:0:0:1]:8080)

No se usa máscara, solo prefix length

## Direcciones IPv6 locales



Prefijo Asignado: FE80::/10

Prefijo Utilizado: FE80::/64 (len. en LAN /64)

Alcance: sólo la red directamente conectada

Mayormente auto-generadas stateless a partir de IID

IID: Interface Identifier, 64 LSb

Obligatoria en todas las interfaces multiacceso

## Generación de IID

Usando IEEE MAC-64

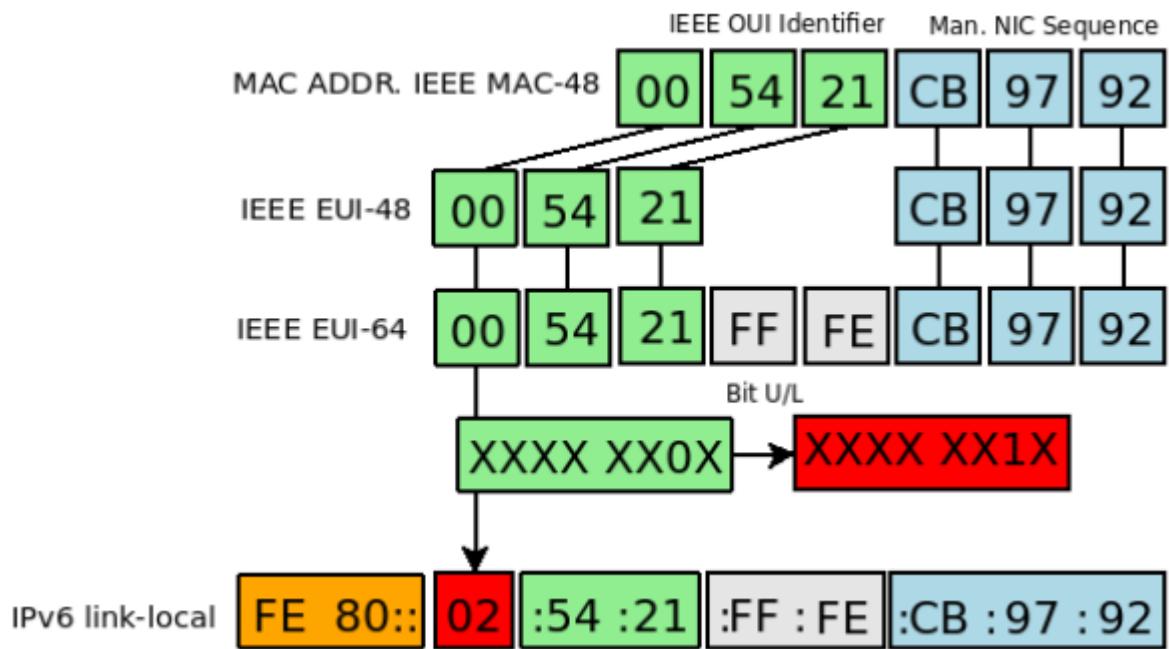
Extended Unique Identifier 64, EUI-64 derivado de IEEE MAC-48, EUI-48

De forma manual

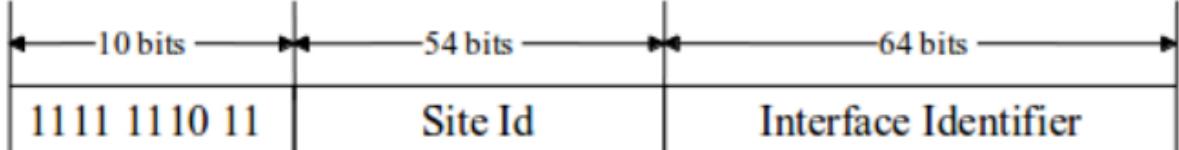
RFC-4941 Privacy Extensions for Stateless Address Autoconfiguration (dir. temporales)

RFC-7217 A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)

Finalmente se generan con el prefijo link-local y realiza DAD (Duplicate Address Detection)



## Direcciones IPv6 Site-Local



Prefijo: FEC0::/10

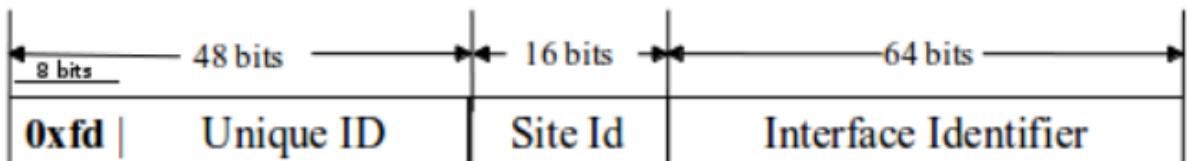
Alcance: sitio u organización

Similar a las redes privadas de IPv4

Dificultad de establecer los límites

Desaconsejado su uso en la RFC 3879, Deprecating Site Local Addresses, de 2004

## Direcciones IPv6 Unique-Local



Prefijo: FC00::/7, dividido en FC00::/8 y FD00::/8

Prefijo Utilizado: FD00::/8, [xxxxxxxxL] L bit = 1 (def. local)

Alcance: sitio u organización

Reemplazan las direcciones de Site Local

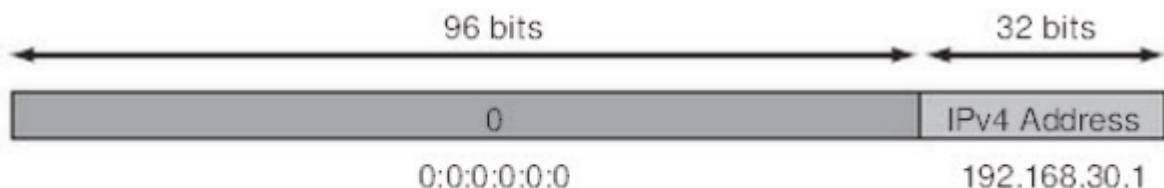
Unique ID debe ser generado de forma (pseudo)-aleatoria

## Direcciones IPv4-compat IPv6

Usadas para la transición

Definidas en RFC-4291 y desaconsejadas

Asigna a un IPv4 global única una IPv6



IPv4-Compatible Address = 0:0:0:0:0:192.168.30.1  
= ::192.168.30.1  
= ::C0A8:1E01

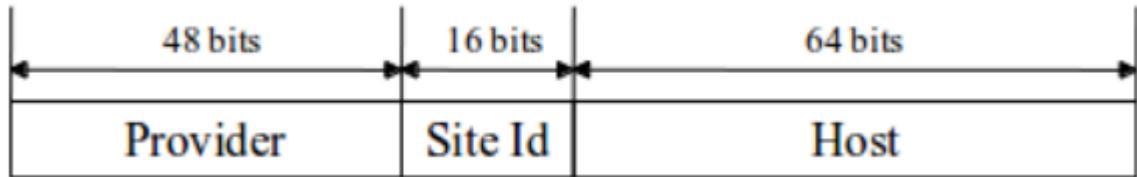
## Direcciones IPv4-mapped IPv6



IPv4-Mapped Address = 0:0:0:0:0:FFFF:192.168.30.1

## Direcciones IPv6 globales

Aggregatable global unicast address



Prefijo: cedidos por un provider

Alcance: Internet. Similar a las direcciones públicas de IPv4

## Generación de IID

Usando IEEE MAC-64 (desaconsejado en RFC-8064)

Extended Unique Identifier 64, EUI-64 derivado de IEEE MAC-48, EUI-48  
(desaconsejado en RFC-8064 para dir. estables)

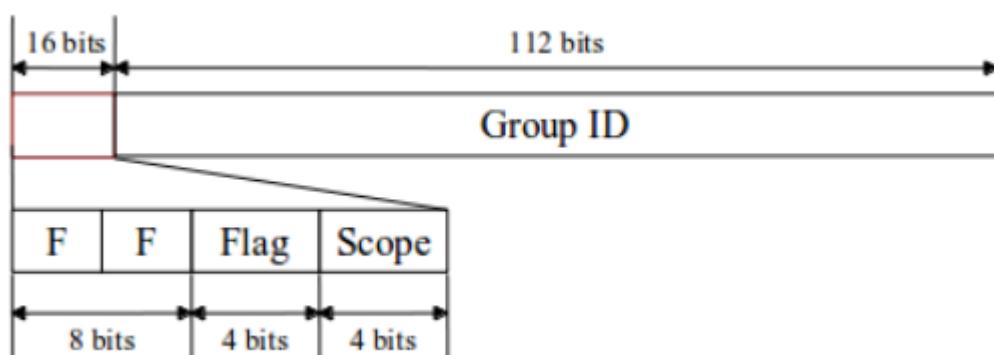
De forma manual

RFC-4941 Privacy Extensions for Stateless Address Autoconfiguration (dir. temporales)

RFC-7217 A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)

Finalmente se generan con el prefijo link-local y realiza DAD (Duplicate Address Detection)

## Direcciones IPv6 Multicast



Prefijo: FF00::/8

Flags: permanente, temporaria. Otros reservados

Alcance: 1: nodo local, 2: link local, 5: site local, 8: org. local, E: global  
GID: grupo de multicast

## Ejemplos

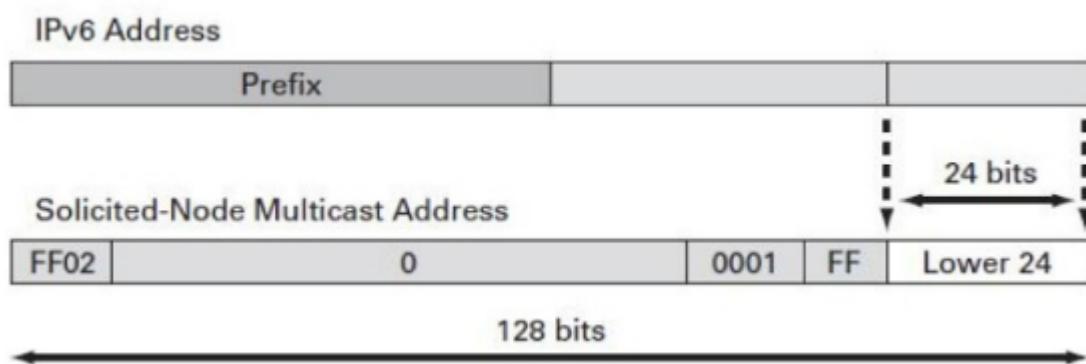
- Node-Local/Interface-Local: FF01::1, FF01::2 (no salen a la red)
- Link-Local (quedan en la LAN):
  - FF02::1 (todos los nodos en la LAN) equivale a 224.0.0.1. Posible reemplazo de broadcast: 255.255.255.255
  - FF02::2 (todos los routers en la LAN) 224.0.0.2
  - FF02::5 OSPFv3 All SPF routers (224.0.0.5)
  - FF02::6 OSPFv3 All DR routers (224.0.0.6)
  - FF02::8 IS-IS for IPv6 routers
  - FF02::9 RIP routers (224.0.0.9)
  - FF02::1:2 All DHCP-Agents (255.255.255.255)
- Site-Local: (quedan en el site)
  - FF05::2 All routers
  - FF05::1:3 All-dhcp-servers RFC-3315
- Generales:
  - FF0X::FB mDNSv6 (DNS multicast)
  - FF0X::102 NTP

## Direcciones IPv6 Multicast SD

Usada para ND (Neighbor Discovery) en lugar de flooding en la LAN

Generada a partir de unicast/anycast

Por cada unicast/anycast debe hacer join de la multicast



Los últimos 24 bits de la dirección de multicast de solicitud van a ser los últimos 24 bits de la dirección IPv6 por la cual estoy consultando

## IPv6 Multicast mapeada en IEEE EUI-48

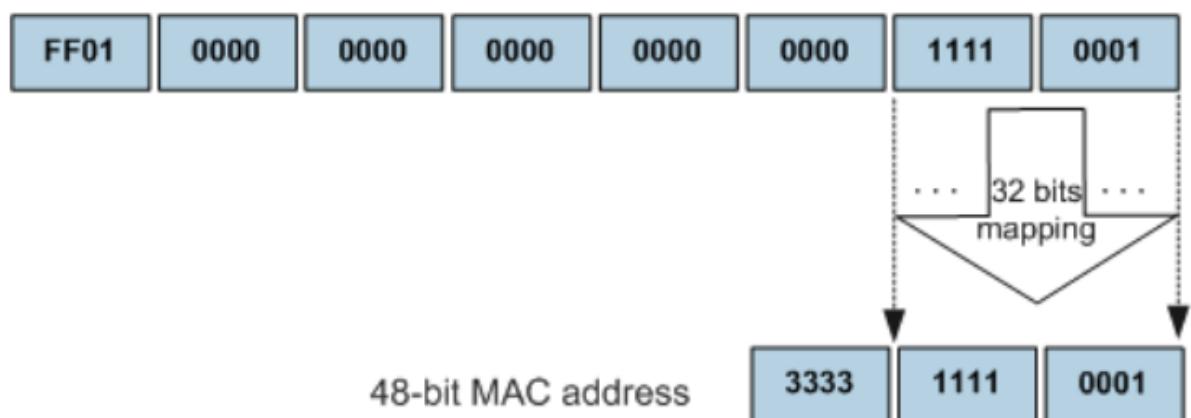
IPv4 multicast mapped to Ethernet:

- 01:00:5E:00:00:00 - 01:00:5E:7F:FF:FF
- 23 LS bits de mcast IPv4 en MAC

IPv6 multicast mapped to Ethernet:

- 33:33:00:00:00:00 - 33:33:FF:FF:FF:FF
- 32 LS bits de mcast IPv6 en MAC

128-bit IPv6 address



## Direcciones IPv6 - Casos Especiales

Any (sin especificar): ::0/0

Loopback/localhost: ::1/128

Documentación: 2001:db8::/32

# Clase 7 - IPv6 - Parte 2

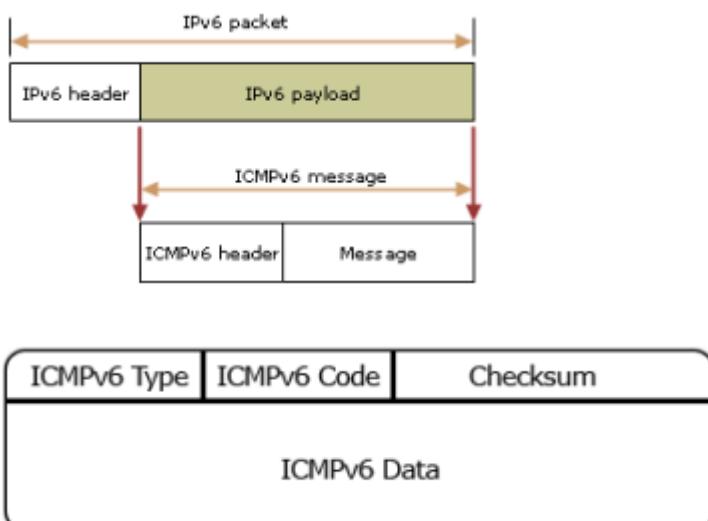
## ICMPv6

ICMPv6 definido en RFC-4443

Parte fundamental del stack IPv6

Resuelve:

- Multicast Listener Discovery (MLD), reemplazo de IGMP
- Neighbor Discovery Protocol (NDP), reemplazo de ARP y mensajes Router Discovery (Auto-configuración), Redirect
- Mensajes de control de ICMP: informativos (ping), errores



## IPv6 Stateless Autoconfiguration

Parte del NDP

Reemplaza la configuración manual de direcciones IPv6 en IPv4

Alternativa básica a DHCPv6, pero sin estados, SLAAC

El router anuncia uno o más prefijos de red mediante mensajes Router Advertisement RA (134)

Se pueden solicitar bajo demanda Router Solicitation RS (133)

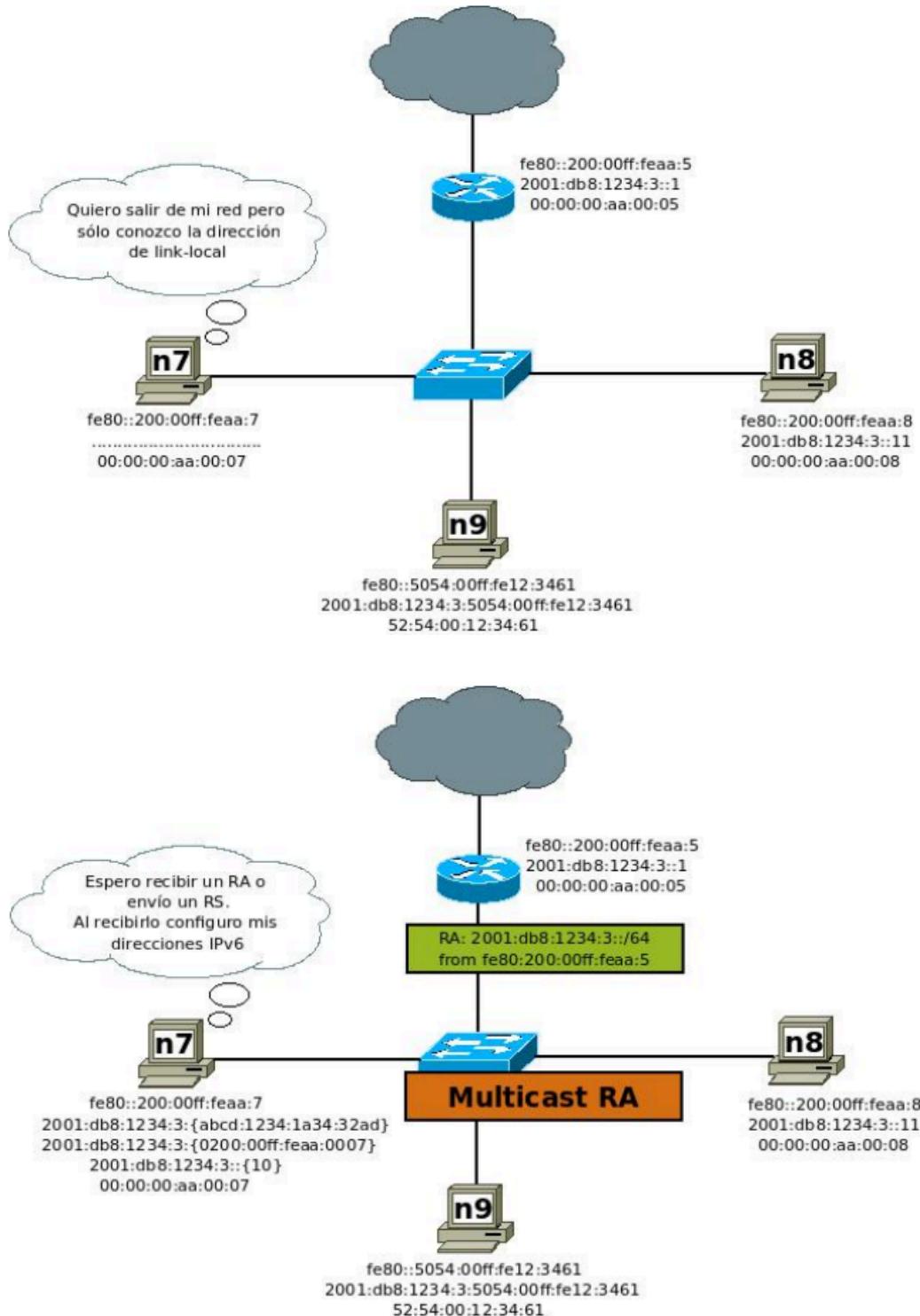
Los hosts auto-configuran su dirección de link-local y solicitan el prefijo a algún router de la red

Una vez obtenido se auto-configuran generando su propias direcciones, previo realizar DAD (Duplicate Address Detection)

Determinan y configuran el default gateway a partir de los Router Advertisement recibidos. Router Advertisement puede llevar opciones de configuración del DNS, RFC-6106

## Router discovery

Aprendizaje de su propia configuración

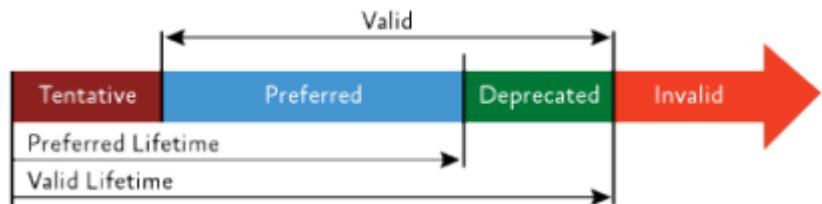


## Mensaje RA

Flags: L(on-link), A(Autonomous), R(Router)

- L, indica que está asignado a una interfaz (si no está no se asume off-link)
- A, sirve para autoconfiguración global
- R, indica que es router, sirve para NUD (Neighbor Unreach. Detection)

Otros parámetros: Tiempo de vida válido y preferido (valid and preferred lifetime)



## IPv6 autoconfiguration DHCPv6

Configuración manual es posible (routers, servers)

Configuración automática, hay variantes: SLAAC, DHCPv6

Combinaciones:

- SLAAC solo, hoy puede obtener conf. básica para Internet, Prefijo, Router y DNS (otros MTU)
- SLAAC solo, algunos equipos no soportan la opción de DNS, requieren DHCPv6
- SLAAC + DHCPv6, configuración básica más parámetros extras por DHCPv6
- DHCPv6 solo, requiere RA para Router/Gateway de la red

Flags en RA:

- O bit - Other Configuration Flag, RFC 4861, indica que puede usar DHCPv6 para obtener otros parámetros, por ejemplo DNS info
- M bit - Managed Address Configuration Flag, RFC4861, indica que usa DHCP6

Combinaciones:

- O = 0, M = 0, configuración vía SLAAC, stateless; si hay DHCPv6 no lo usaría

- O = 1, M = 0, configuración vía SLAAC, stateless; por DHCPv6 obtiene parámetros adicionales. AdvOtherConfigFlag
- O = \*, M = 1, configuración vía DHCPv6 stateful; salvo router. AdvManagedFlag

## Neighbor Discovery

Reemplaza básicamente al protocolo ARP de IPv4

Mapea direcciones lógicas (IPv6) a direcciones de Hardware (MAC, EUI-48, EUI-64)

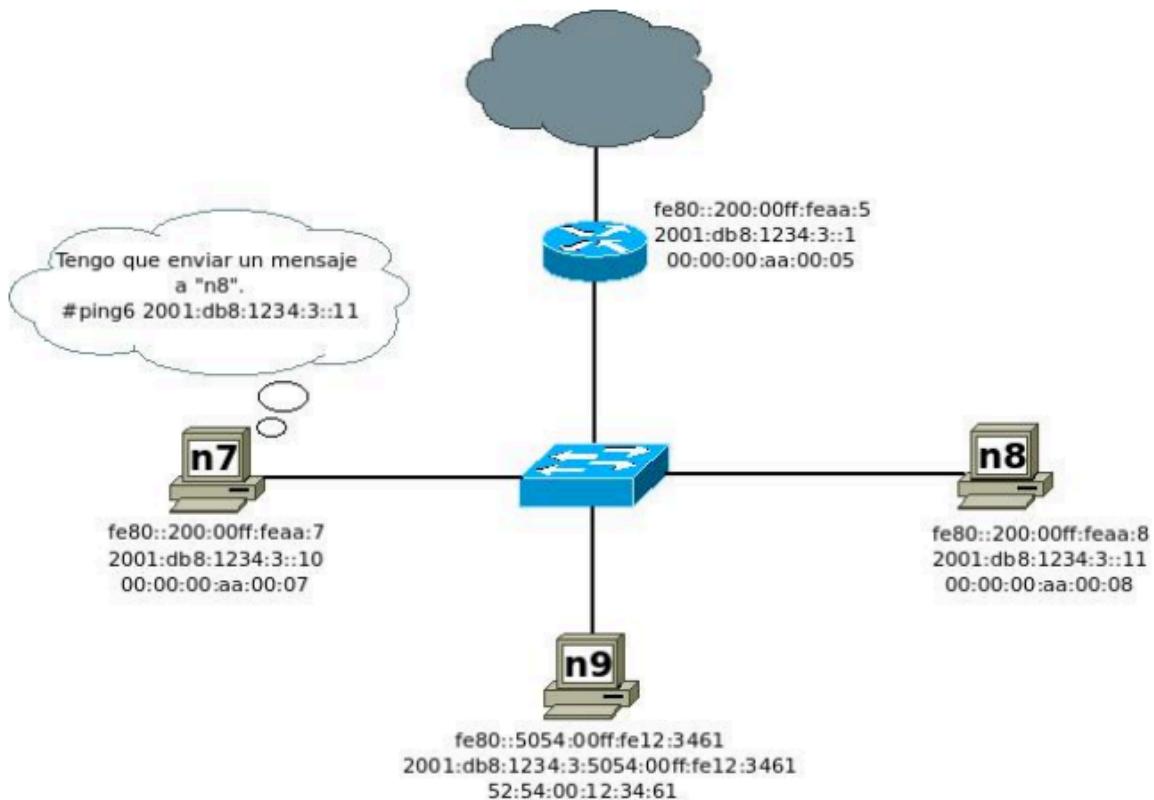
Trabaja conjuntamente con Ethernet (u otros protocolos de L2 multiacceso con broadcast: Bluetooth, 802.11, (Token-Ring, FDDI)

Trabaja de forma dinámica, auto-aprendizaje, sin configuración

Puede configurarse de forma estática

Definido en RFC-4861

2 tipos de mensajes: Neighbor Solicitation NS(135) y Neighbor Adv. NA(136)



- “n7” debe recurrir a un Neighbor Solicitation (NS).
- Como no sabe la MAC debe enviar un multicast L2 y L3.

```
root@n7:/# ip -6 addr show dev eth0
27: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc 1000
    ...
    inet6 2001:db8:1234:3::10/64 scope global
        valid_lft forever preferred_lft forever
    inet6 fe80::200:ff:feaa:7/64 scope link
        valid_lft forever preferred_lft forever

root@n7:/# ip -6 neigh show
fe80::200:ff:feaa:5 dev eth0 lladdr 00:00:00:aa:00:05 router STALE

root@n7:/# ping6 -c 5 2001:db8:1234:3::11 -I 2001:db8:1234:3::10
```

### ● Mensajes NS (Neighbor Solicitation):

No.	Time	Source	Destination	Length	Info
1 0.000000		2001:db8:1234:3::10	ff02::1:ff00:11	86	Neighbor Solicitation for 2001:db8:1234:3::11 from 00:00:00:aa:00:07
2 0.000151		2001:db8:1234:3::11	2001:db8:1234:3::10	86	Neighbor Advertisement 2001:db8:1234:3::11 (sol, ovr) is at 00:00:00:aa:00:07
3 0.000161		2001:db8:1234:3::10	2001:db8:1234:3::11	118	Echo (ping) request id=0x004e, seq=1, hop limit=64 (reply in 4)
4 0.000186		2001:db8:1234:3::11	2001:db8:1234:3::10	118	Echo (ping) reply id=0x004e, seq=1, hop limit=64 (request in 3)
► Frame 1: 86 bytes on wire (688 bits), 86 bytes captured (688 bits)					
► Ethernet II, Src: 00:00:00:aa:00:07 (00:00:00:aa:00:07), Dst: 33:33:ff:00:00:11 (33:33:ff:00:00:11)					
▼ Internet Protocol Version 6, Src: 2001:db8:1234:3::10 (2001:db8:1234:3::10), Dst: ff02::1:ff00:11 (ff02::1:ff00:11)					
► 0110 .... = Version: 6					
► .... 0000 0000 .... .... .... = Traffic class: 0x00000000					
.... .... 0000 0000 0000 0000 = Flowlabel: 0x00000000					
Payload length: 32					
Next header: ICMPv6 (58)					
Hop limit: 255					
Source: 2001:db8:1234:3::10 (2001:db8:1234:3::10)					
Destination: ff02::1:ff00:11 (ff02::1:ff00:11)					
▼ Internet Control Message Protocol Version 6					
Type: Neighbor Solicitation (135)					
Code: 0					
Checksum: 0xf8db [correct]					
Reserved: 00000000					
Target Address: 2001:db8:1234:3::11 (2001:db8:1234:3::11)					
▼ ICMPv6 Option (Source link-layer address : 00:00:00:aa:00:07)					
Type: Source link-layer address (1)					
Length: 1 (8 bytes)					
Link-layer address: 00:00:00:aa:00:07 (00:00:00:aa:00:07)					

En el ejemplo, yo quiero saber la MAC del dispositivo con IP 2001:db8:1234:3::11

Para eso, primero obtengo la dirección IP multicast a la cual debo enviar un mensaje para hacer la solicitud

Eso se obtiene con los últimos 24 bits de la IP target, en este caso son **0000011** (en hexadecimal), lo cual, con el [prefijo de las direcciones de multicast para neighbor discovery](#) nos queda: **FF02::1:FF00:0011**

Por último, hay que obtener la dirección MAC destino a la que hacer la solicitud, y eso se obtiene [mapeandola](#). Para eso hay que obtener los últimos 32 bits de la dirección IP que acabamos de generar (**FF0000011**) y luego mapear, entonces nos queda: 33:33:**FF:00:00:11**

- “n8” procesa el requerimiento y responde con un **Link Layer Advertisement (NA)**:

No.	Time	Source	Destination	Length	Info
1 0.000000	2001:db8:1234:3::10	ff02::1:ff00:11		86	Neighbor Solicitation for 2001:db8:1234:3::10
2 0.000151	2001:db8:1234:3::11	2001:db8:1234:3::10		86	Neighbor Advertisement 2001:db8:1234:3::11
3 0.000161	2001:db8:1234:3::10	2001:db8:1234:3::11		118	Echo (ping) request id=0x004e, seq=1, hop limit=1
4 0.000186	2001:db8:1234:3::11	2001:db8:1234:3::10		118	Echo (ping) reply id=0x004e, seq=1, hop limit=1

```

►Frame 2: 86 bytes on wire (688 bits), 86 bytes captured (688 bits)
►Ethernet II, Src: 00:00:00:aa:00:08 (00:00:00:aa:00:08), Dst: 00:00:00:aa:00:07 (00:00:00:aa:00:07)
▼Internet Protocol Version 6, Src: 2001:db8:1234:3::11 (2001:db8:1234:3::11), Dst: 2001:db8:1234:3::10 (2001:db8:1234:3::10)
  ►0110 .... = Version: 6
  ►.... 0000 0000 .... .... .... .... = Traffic class: 0x00000000
  .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
  Payload length: 32
  Next header: ICMPv6 (58)
  Hop limit: 255
  Source: 2001:db8:1234:3::11 (2001:db8:1234:3::11)
  Destination: 2001:db8:1234:3::10 (2001:db8:1234:3::10)
▼Internet Control Message Protocol v6
  Type: Neighbor Advertisement (136)
  Code: 0
  Checksum: 0x54ef [correct]
  ►Flags: 0x60000000
  Target Address: 2001:db8:1234:3::11 (2001:db8:1234:3::11)
▼ICMPv6 Option (Target link-layer address : 00:00:00:aa:00:08)
  Type: Target link-layer address (2)
  Length: 1 (8 bytes)
  Link-layer address: 00:00:00:aa:00:08 (00:00:00:aa:00:08)

** Request: Neighbor Solicitation (NS)

From: MAC:<my-MAC-address> , IPv6:<my-Link-Local-Address> / IPv6: <my-global-Address>
To: MAC:<MAC-multicast-address> , IPv6:<solicited-node_multicast_address>

<solicited-node-multicast-address> (SNMA): ff02:0000:0000:0000:0000:0001:ff00:0000/104
  + last 24bits IPv6
<MAC-multicast-address> (MMA) : 33:33: + last 32bits IPv6 mcast address

n7: mac address: 00:00:00:aa:00:07
  ipv6 address: 2001:db8:1234:3::10/64
  ipv6 link-local address: fe80::0200:00ff:fe00:0007
  "n7" Try to discover: 2001:db8:1234:3::11/64 MAC address ("n8")

SNMA: ff02:0000:0000:0000:0000:0001:ff00:0000/104 + 00:00:11
  = ff02:0000:0000:0000:0001:ff00:0011
  https://tools.ietf.org/html/rfc4291
MMA: 33:33: + ff:00:00:01 = 33:33:ff:00:00:11
  https://tools.ietf.org/html/rfc2464#section-7
  https://tools.ietf.org/html/rfc6085
  https://tools.ietf.org/html/rfc7042
From: MAC: 00:00:00:aa:00:07 , IPv6: fe80::0200:00ff:fe00:0007 / 2001:db8:1234:3::10
To: MAC: 33:33:ff:00:00:11 IPv6: ff02:0000:0000:0000:0001:ff00:0011

** Reply: Neighbor Advertisement (NA)

From: MAC:<my-MAC-address> , IPv6:<my-IPv6-Requested-Address>
To: MAC:<MAC-who-request-address> , IPv6:<IPv6-who-request-address>

From: MAC: 00:00:00:aa:00:08 , IPv6: fe80::0200:00ff:fe00:0008 / 2001:db8:1234:3::11
To: MAC: 00:00:00:aa:00:07 IPv6: fe80::0200:00ff:fe00:0007 / 2001:db8:1234:3::10

```

## Neighbor advertisement

Solicitado o no solicitado

Solicitado, respuesta a RS Flag: (S)Solicited=1

NO-solicitado, para actualizar caches Flag: (O)Override=1

Flag (R)Router=1, lo envía el router.

## PMTU (Path MTU) Discovery

MTU (Maximum Transmission Unit) depende de L2

Cada Link L2 puede tener su MTU Link-MTU

A lo largo de un camino se puede establecer Path MTU (PMTU)

Para IPv6 min MTU=1280B (RFC-2460, RFC-8200), Para IPv4 68B(RFC-791)

Recomendado, (estandarizado) 1500B

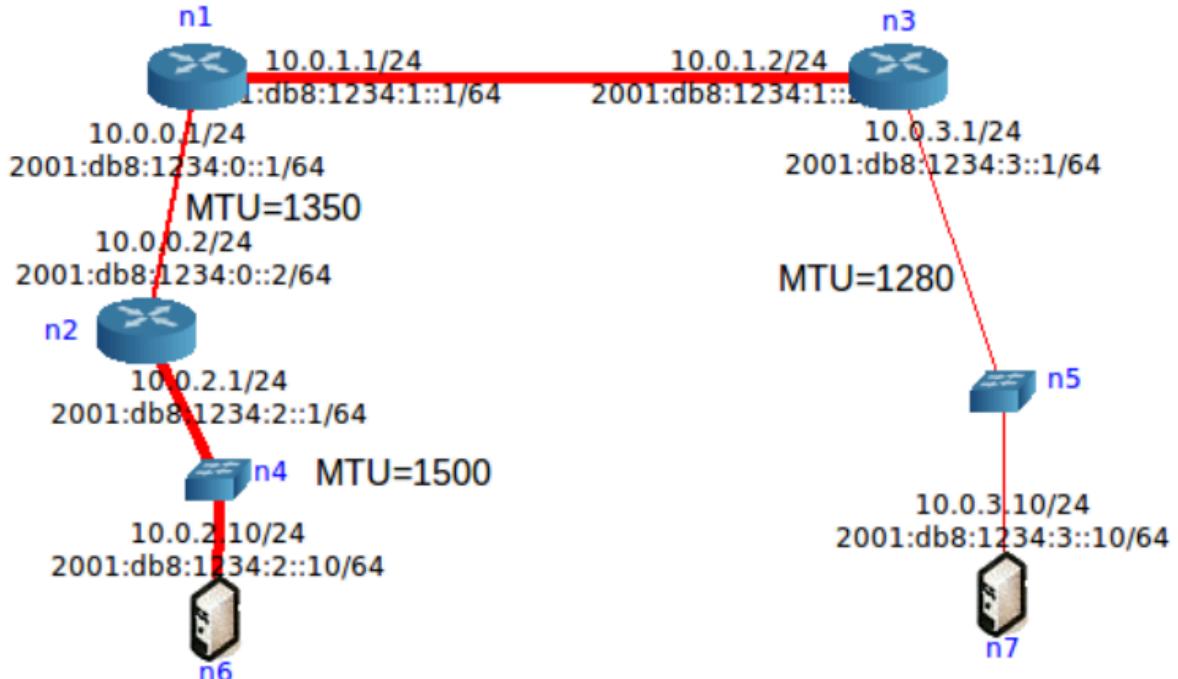
Las implementaciones hacen el PMTU discovery (RFC-1981)

Una vez que un paquete llega a un router, si el camino que sigue tiene una MTU menor al tamaño del paquete, el router devuelve un error de ICMPv6 Error: "Packet Too Big"

Una vez determinado el PMTU se fragmenta de extremo a extremo

Se puede saltar y usar directamente 1280, no es óptimo

TCP trata de usar MSS (Maximum Segment Size)



Time	01:db8:1234:2::10	2001:db8:1234:2::10	2001:db8:1234:3::10	2001:db8:1234:1::2	Comment
0.000000					
0.000539					ICMPv6: Echo (ping) request id=0x0032, seq=1, hop limit=64 (no response found!)
1.999869					ICMPv6: Packet Too Big
2.000198					IPv6: IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0xbff2f2768)
2.000231					ICMPv6: Packet Too Big
4.002568					ICMPv6: Echo (ping) request id=0x0032, seq=2, hop limit=64 (no response found!)
4.002912					IPv6: IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0xbff2f2769)
4.003248					ICMPv6: Echo (ping) request id=0x0032, seq=3, hop limit=64 (reply in 9)
4.003259					IPv6: IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0x55c4a2e5)
6.005221					ICMPv6: Echo (ping) reply id=0x0032, seq=3, hop limit=61 (request in 7)
6.005583					IPv6: IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0xbff2f276a)
6.005893					ICMPv6: Echo (ping) request id=0x0032, seq=4, hop limit=64 (reply in 13)
6.005904					IPv6: IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0x55c4a2e6)
8.007685					ICMPv6: Echo (ping) reply id=0x0032, seq=4, hop limit=61 (request in 11)
8.007895					IPv6: IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0xbff2f276b)
8.008197					ICMPv6: Echo (ping) request id=0x0032, seq=5, hop limit=64 (reply in 17)
8.008208					IPv6: IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0x55c4a2e7)
					ICMPv6: Echo (ping) reply id=0x0032, seq=5, hop limit=61 (request in 15)

## Transición de IPv4 a IPv6

No va a existir un día “D” para cambiar de IPv4 a IPv6

Amplio abanico de técnicas disponibles:

- Doble pila (Dual-Stack)
- Técnicas de tunneling
- Técnicas de traducción IPv6 ↔ IPv4
- Técnicas combinadas

## Dual stack IPv4/IPv6

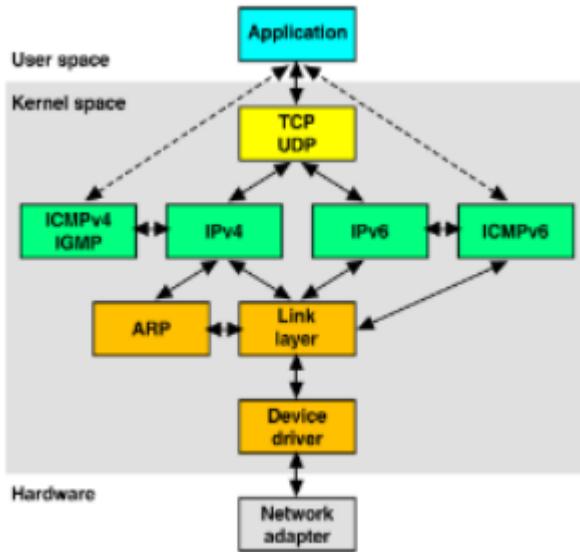
No se elimina la pila IPv4

Técnica multi-protocolo como con: NetBIOS, Appletalk, IPX

Actualmente, la mayoría de los OSs soportan IPv6

La aplicación, biblioteca de código elige cual usar (registros de DNS AAAA y A)

Van a coexistir por “mucho” tiempo



## Túneles IPv4/IPv6

Encapsular IPv6 en IPv4 donde no hay cobertura

Se requiere doble pila IPv4/IPv6 en los routers

Pueden ser:

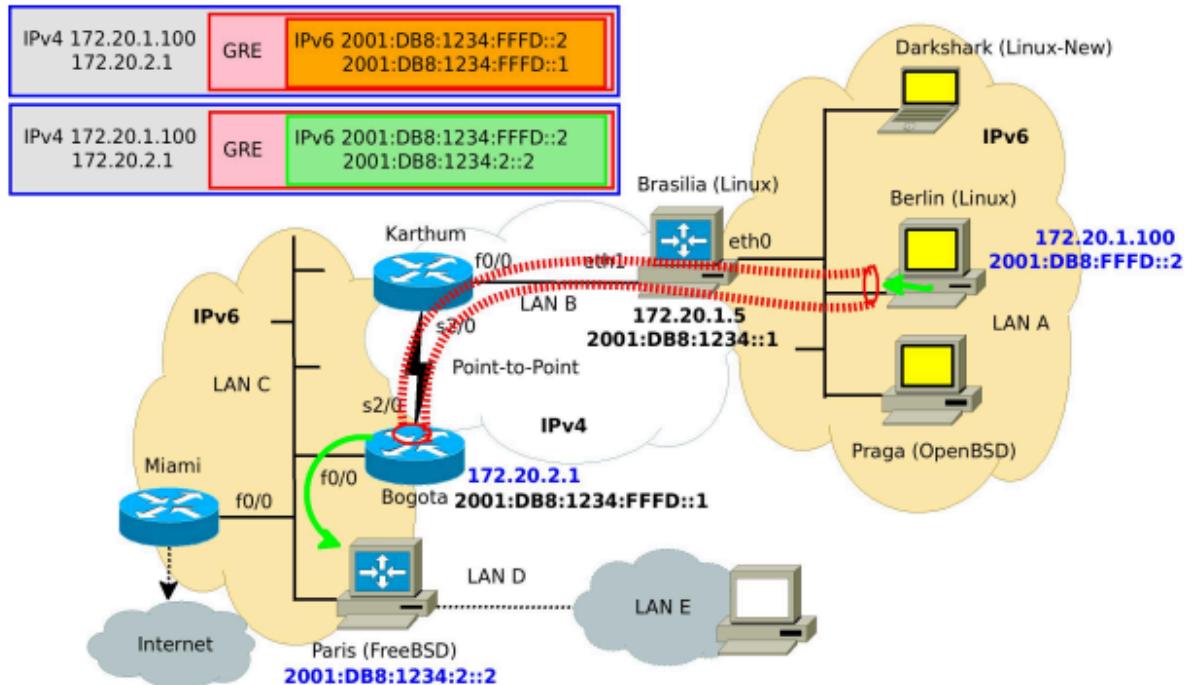
- Manuales:
  - GRE (point-to-point)
  - SIT (Simple Internet Tunnel) IP-IP - IPv6-IPv4 (point-to-point)
  - 6in4
- Tunnels Brokers, interfaces web de ayuda:
  - 6in4+TB (<https://tunnelbroker.net/>)
- Automáticos:
  - 6to4 (No funciona sobre NAT)
  - ISATAP (no funciona sobre NAT)
  - Teredo (sobre NAT)

### Manuales

#### GRE

Encapsula en GRE (47) General Routing Encapsulation

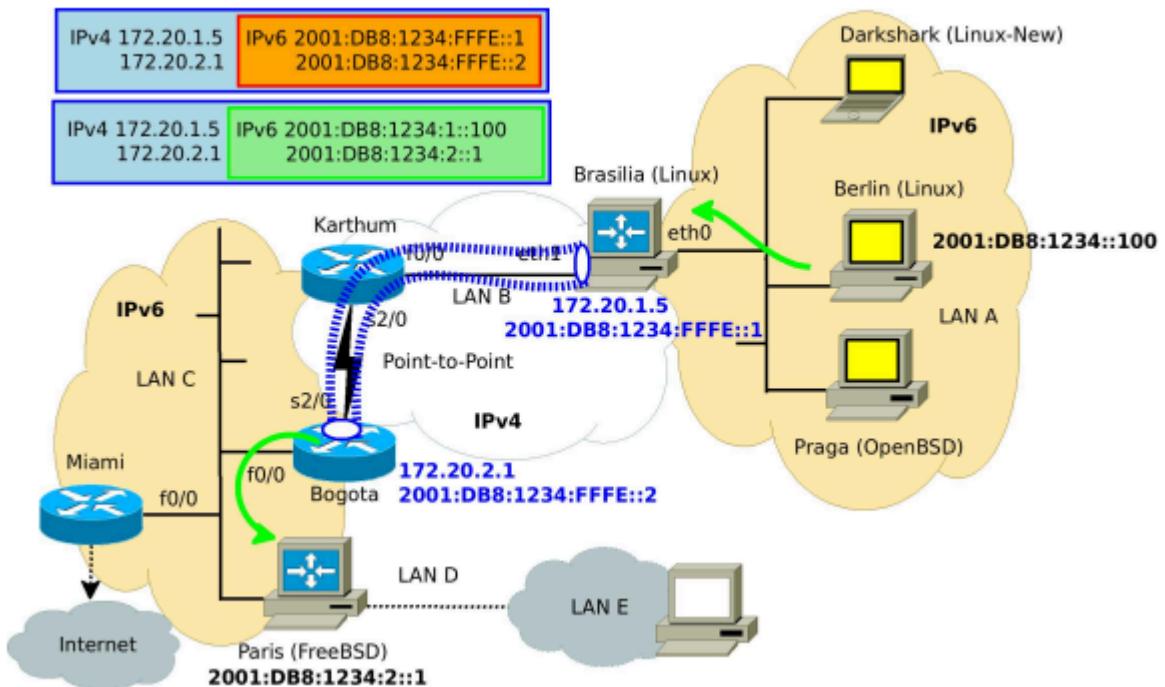
Punto a punto entre host y router o entre routers



## SIT

Encapsula directamente en IPv4, proto 41(IPv6)

Punto a punto entre host y router o entre routers



## 6to4

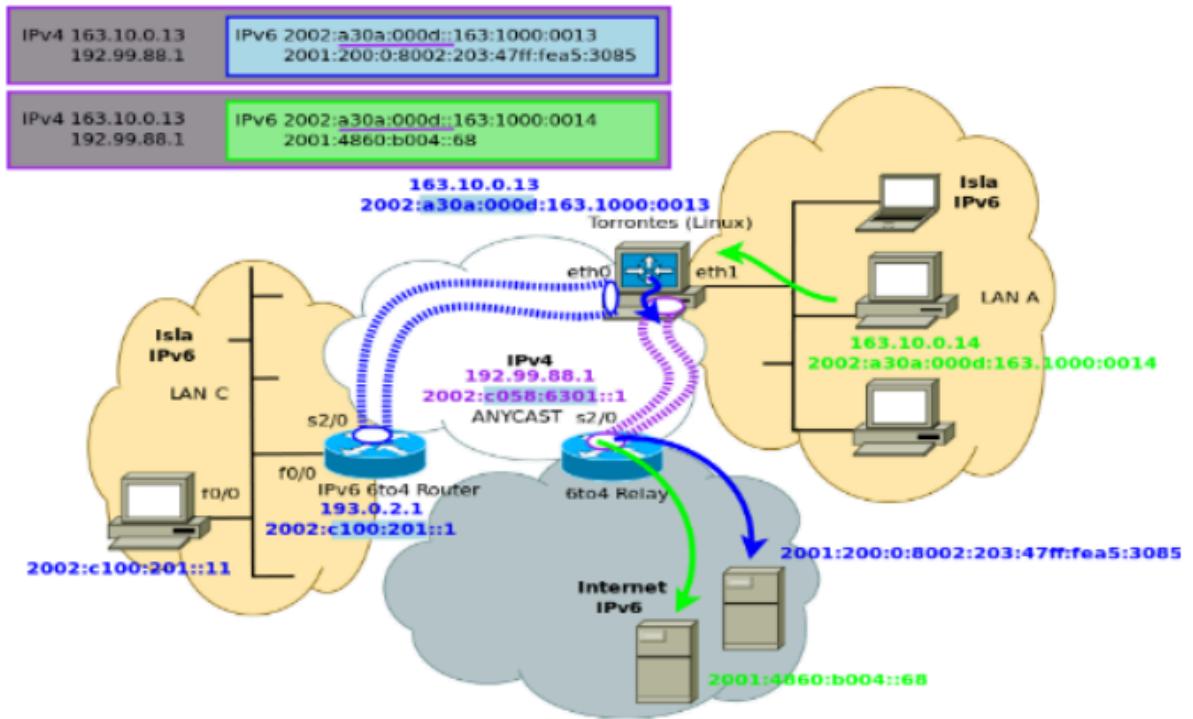
Sobre direcciones IPv4 globales

Se usa el bloque: 2002::/16

Paquetes de salida siempre al 6to4 relay

Paquetes de vuelta, pueden usar otro origen

Prefijo 6to4 relays anycast: 192.88.99.1/24



## Teredo

Funciona sobre direcciones IPv4 privadas, con NAT y sin Proto 41

Encapsula en datagramas UDP

Se configura el cliente contra un Server Teredo

El Server Teredo proporciona acceso a los Teredo Relay

