

Enrutamiento IP

Álvaro González Sotillo

8 de septiembre de 2024

Índice

1. Introducción	1
2. Cómo enruta IP	2
3. Ejemplo de enrutamiento	3
4. Tablas de enrutamiento	4
5. Referencias	7

1. Introducción

- IP se encarga de llevar cada paquete hasta su red
- Una vez llegado a la red, IP supone que:
 - La red es de difusión (bus), con lo cual llega a su destino
 - La entrada a la red es un router que conoce las subredes dentro de la red, con lo que se sigue enrutando el paquete

1.1. pathping en Windows

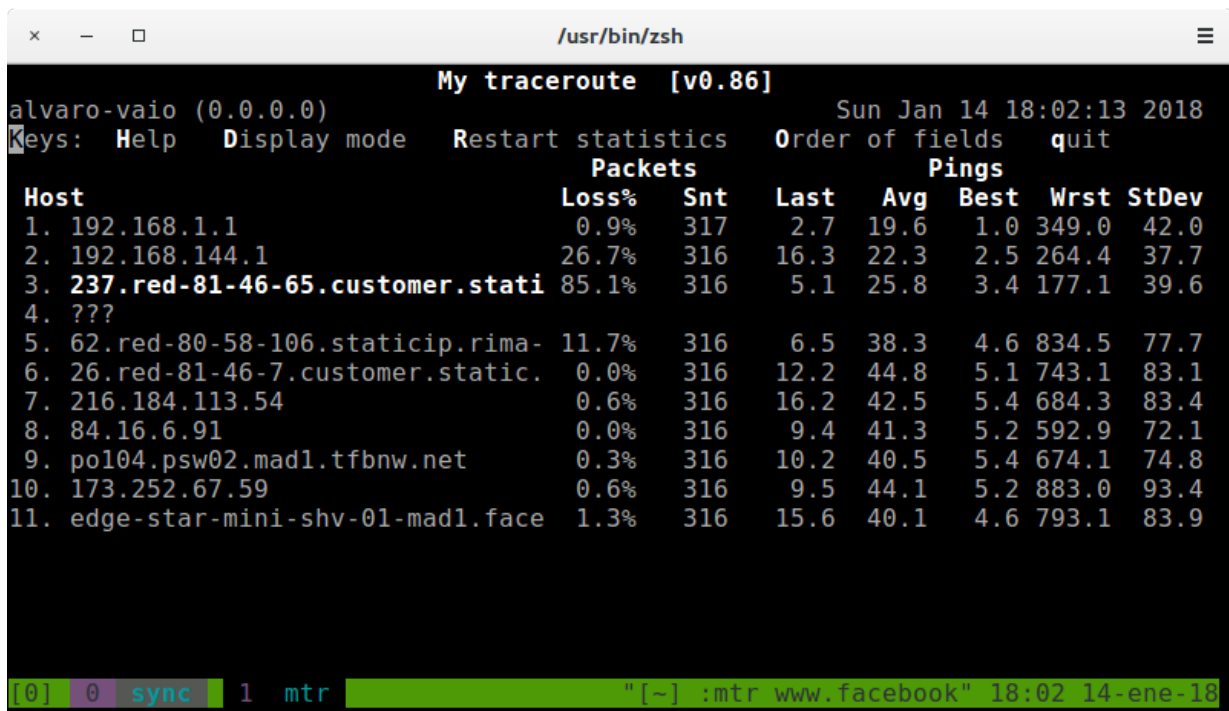


```
C:\WINDOWS\system32\cmd.exe - pathping www.facebook.com

C:\Documents and Settings\Alvaro>pathping www.facebook.com

Traza a www.facebook.com[69.63.184.29] sobre caminos de 30 saltos como máximo:
 0 ALVAROGONZALEZ [192.168.20.232]
 1 192.168.20.1
 2 10.7.239.1
 3 81.Red-81-46-40.staticIP.rima-tde.net [81.46.40.81]
 4 So6-0-0-0-grtmadpe3.red.telefonica.wholesale.net [84.16.8.121]
 5 So2-2-0-0-grtmadde2.red.telefonica-wholesale.net [84.16.12.205]
 6 p16-7-1-1.r21.mdrdsp01.es.bb.gin.ntt.net [129.250.8.197]
 7 p64-2-1-0.r22.amstnl02.nl.bb.gin.ntt.net [129.250.5.45]
 8 ae-1.r23.amstnl02.nl.bb.gin.ntt.net [129.250.4.222]
 9 as-0.r21.asbnva01.us.bb.gin.ntt.net [129.250.2.158]
10 * * *
Procesamiento de estadísticas durante 250 segundos...
```

1.2. mtr en Linux



```
alvaro-vaio (0.0.0.0) Sun Jan 14 18:02:13 2018
My traceroute [v0.86]
Keys: Help Display mode Restart statistics Order of fields quit

Host                Loss%  Snt   Last   Avg    Best  Wrst  StDev
1. 192.168.1.1        0.9%   317   2.7    19.6   1.0   349.0  42.0
2. 192.168.144.1      26.7%  316   16.3   22.3   2.5   264.4  37.7
3. 237.red-81-46-65.customer.static. 85.1%  316   5.1    25.8   3.4   177.1  39.6
4. ???
5. 62.red-80-58-106.staticip.rima- 11.7%  316   6.5    38.3   4.6   834.5  77.7
6. 26.red-81-46-7.customer.static. 0.0%   316   12.2   44.8   5.1   743.1  83.1
7. 216.184.113.54    0.6%   316   16.2   42.5   5.4   684.3  83.4
8. 84.16.6.91        0.0%   316   9.4    41.3   5.2   592.9  72.1
9. pol04.psw02.mad1.tfbnw.net      0.3%   316   10.2   40.5   5.4   674.1  74.8
10. 173.252.67.59    0.6%   316   9.5    44.1   5.2   883.0  93.4
11. edge-star-mini-shv-01-mad1.face 1.3%   316   15.6   40.1   4.6   793.1  83.9

[0] 0 sync 1 mtr [~] :mtr www.facebook" 18:02 14-ene-18
```

2. Cómo enruta IP

- Hacen falta los siguientes datos para enrutar un paquete IP
 - La dirección de destino del paquete
 - La red a la que pertenece la dirección de destino
 - La interfaz de red por donde se alcanza la red de destino
 - La dirección del router que nos ayuda a alcanzar el destino

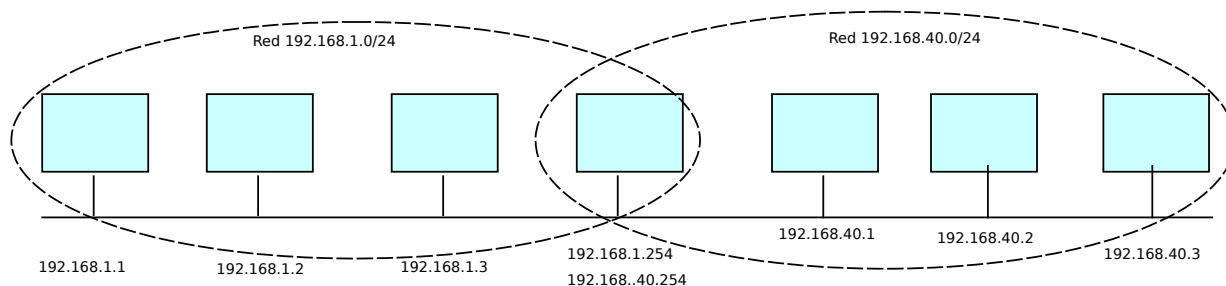
2.1. IP vs MAC

- Un host con IP recibe y procesa paquetes tanto con su propia dirección IP como la de otros:
 - Ethernet (o el protocolo usado en la capa de enlace) pasa a IP todos los paquetes que llegan con su dirección MAC
 - IP pasa hacia arriba los paquetes con la dirección IP propia
 - IP enruta (reenvía) los paquetes no dirigidos a su propia dirección IP
- Por tanto, para que funcione IP tiene que ser posible enviar paquetes IP a otro host aunque no sea el indicado en la dirección IP destino
 - Se hace utilizando las direcciones MAC o físicas del otro host
 - La unión entre direcciones MAC y físicas es el protocolo ARP

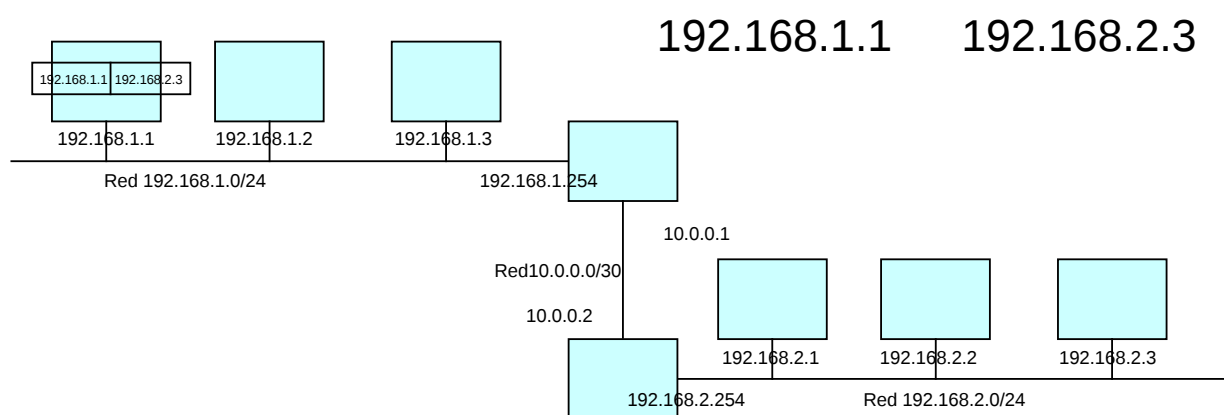
2.2. Red IP vs Red Ethernet

- IP trabaja sobre la capa de enlace sin conocer sus detalles
 - Ni siquiera conoce la topología física
 - Un enlace físico podría tener más de una dirección IP
- Como consecuencia, puede haber más de una red IP encima de la misma red ethernet

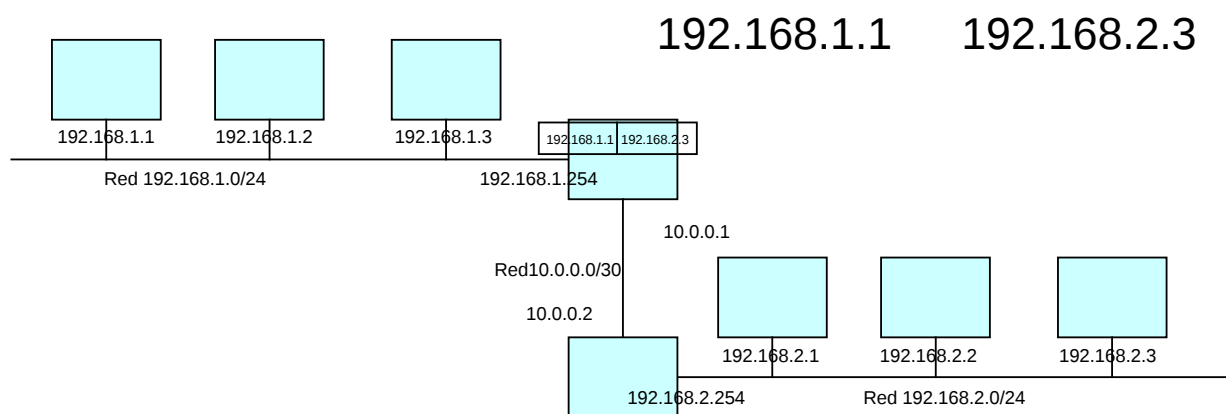
- Por interés administrativo o de seguridad



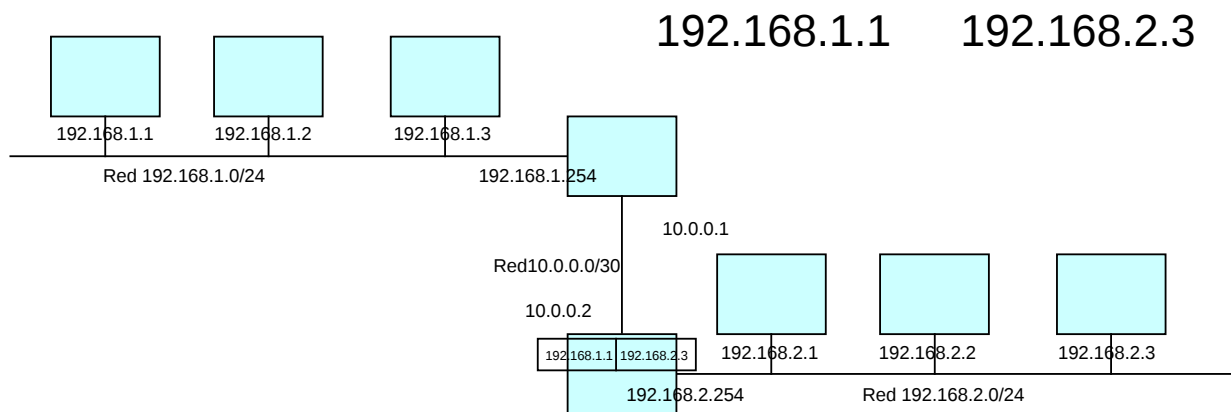
3. Ejemplo de enrutamiento



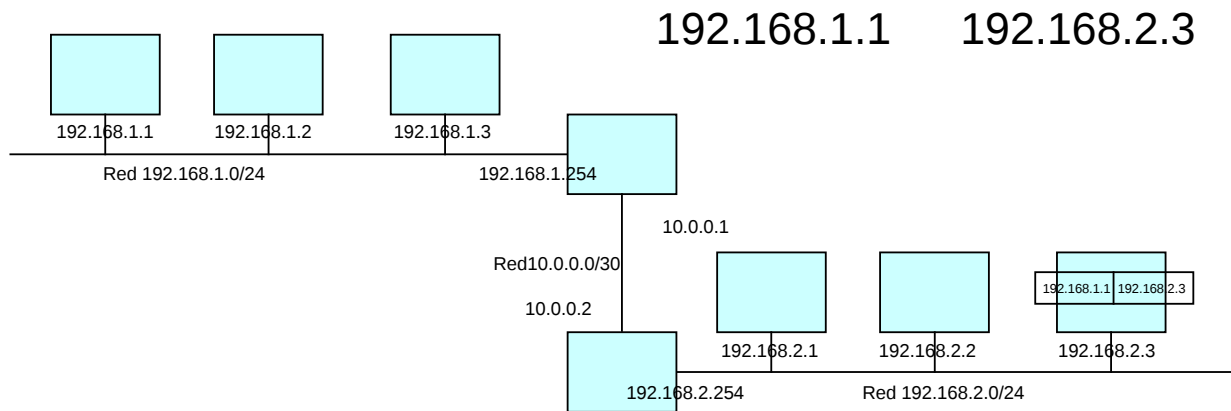
- 192.168.1.1 debe conocer su máscara de red (/24)
- Con ella decide si 192.168.2.3 está en su red
 - Está en su red si la dirección de destino está en la misma red, usando la máscara
- Puesto que el destino no está en la misma red, 192.168.1.254 debe utilizar una puerta de enlace



- La puerta de enlace de 192.168.1.1 debe estar configurada a 192.168.1.254
- Se envía una trama por el bus con la MAC del router, pero con dirección IP origen 192.168.1.1 y dirección destino 192.168.2.3
- La capa ethernet del router la recibe, y la pasa al protocolo superior, IP
- IP no la pasa al protocolo superior, ya que no va dirigida al propio router. Sin embargo, enrutará de nuevo el paquete para que llegue a su destino



- El router consulta sus tablas, y sabe que para llegar a la red 192.168.2.0/24 tiene que utilizar el router 10.0.0.2
- Envía el paquete a 10.0.0.2
 - Pero con dirección IP origen 192.168.1.1 y dirección destino 192.168.2.3.
 - La trama ethernet donde va el paquete tiene la dirección MAC asociada a 10.0.0.2 (si no, no le llegaría)



- El segundo router repite los pasos que realizó el primero, reenviando la trama por su enlace 192.168.2.254
- El equipo 192.168.2.3 recibe la trama
- IP detecta que va dirigida a este mismo equipo, por lo que no la enruta y la pasa al protocolo superior

4. Tablas de enrutamiento

- Cada host con IP debe tener configuradas sus tablas de enrutamiento
- Estudiaremos las tablas de enrutamiento en
 - Windows
 - Linux/Unix
- Se manejan con el comando `route` o `ip route`

4.1. Filas y columnas

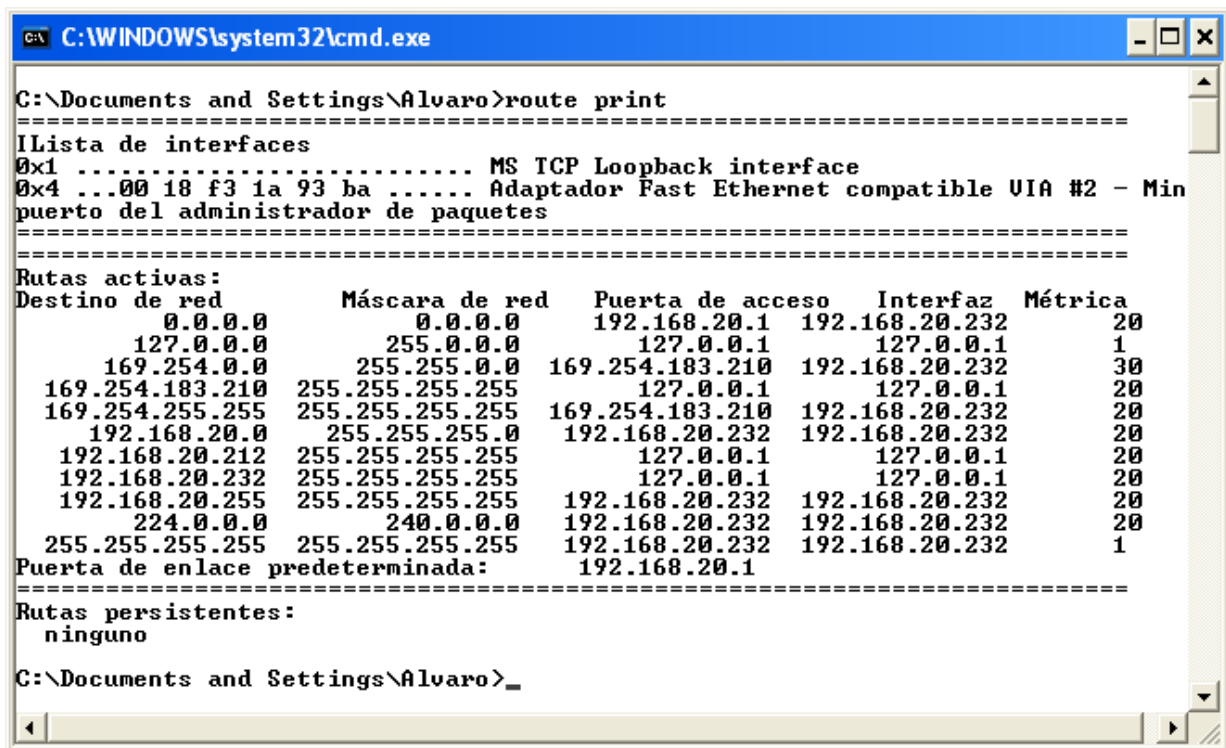
- Cada **fila** de la tabla es una regla
 - Tiene información de cómo llegar a una red o host
- Cada **columna** es el nombre de la información
 - Destino, máscara...

4.2. Descripción de columnas

- Destino: IP (host o red) con el que se comparará la dirección de destino de los paquetes a enrutar
- Máscara: Máscara que se aplica a la dirección de destino del paquete IP
 - Para saber si una entrada es utilizable por una IP destino, se aplica la máscara sobre la IP destino del paquete y se comprueba si coincide con la IP destino de la entrada
- Puerta de acceso: Dirección IP del siguiente enrutador, si se utiliza esta entrada
- Interfaz: enlace del host local por que el saldrá el paquete si se utiliza esta entrada
- Métrica: si es posible usar más de una entrada, se utilizará preferiblemente la entrada con menor métrica

4.3. Windows

- Comando `route print`



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Alvaro>route print
=====
ILista de interfaces
0x1 ..... MS TCP Loopback interface
0x4 ...00 18 f3 1a 93 ba ..... Adaptador Fast Ethernet compatible UIA #2 - Min
puerto del administrador de paquetes
=====
Rutas activas:
Destino de red      Máscara de red    Puerta de acceso  Interfaz  Métrica
0.0.0.0            0.0.0.0          192.168.20.1     192.168.20.232  20
127.0.0.0         255.0.0.0        127.0.0.1       127.0.0.1      1
169.254.0.0       255.255.0.0     169.254.183.210 192.168.20.232  30
169.254.183.210   255.255.255.255 127.0.0.1       127.0.0.1      20
169.254.255.255   255.255.255.255 169.254.183.210 192.168.20.232  20
192.168.20.0      255.255.255.0   192.168.20.232  192.168.20.232  20
192.168.20.212    255.255.255.255 127.0.0.1       127.0.0.1      20
192.168.20.232    255.255.255.255 127.0.0.1       127.0.0.1      20
192.168.20.255    255.255.255.255 192.168.20.232  192.168.20.232  20
224.0.0.0         240.0.0.0        192.168.20.232  192.168.20.232  20
255.255.255.255   255.255.255.255 192.168.20.232  192.168.20.232  1
Puerta de enlace predeterminada: 192.168.20.1
=====
Rutas persistentes:
ninguno
C:\Documents and Settings\Alvaro>
```

4.3.1. Modificar las tablas en Windows

- Añadir una ruta para la red 157.0.0.0/8: se va por la interfaz 2, con métrica 3. Se envía al router 157.55.80.1

```
ROUTE ADD 157.0.0.0 MASK 255.0.0.0 157.55.80.1 METRIC 3 IF 2
```

- La métrica y la interfaz son opcionales
- Internet se puede especificar como 0.0.0.0

4.3.2. Activar el enrutamiento

- Hay varios métodos, pero todos acaban **modificando el registro**
 - HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\IpEnableRouter debe estar a 1
- Después, hay que reiniciar

4.4. Linux

- Comando route

```
alvaro@alvaro-vaio$ route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default gateway 0.0.0.0 UG 600 0 0 wlan0
link-local * 255.255.0.0 U 1000 0 0 wlan0
192.168.1.0 * 255.255.255.0 U 600 0 0 wlan0
192.168.56.0 * 255.255.255.0 U 0 0 0 vboxnet0
```

4.5. Linux

- Comando ip route

```
alvaro@alvaro-vaio~$ ip route
default via 192.168.1.1 dev wlp0s20f3 proto dhcp metric 600
169.254.0.0/16 dev docker0 scope link metric 1000
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1
192.168.1.0/24 dev wlp0s20f3 proto kernel scope link src 192.168.1.13 metric 600
```

4.5.1. Modificar las tablas en Linux (**route**)

- Añadir una ruta para la red 157.0.0.0/8: se va por la interfaz eth0, con métrica 3. Se envía al router 157.55.80.1

```
sudo route add -net 157.0.0.0 netmask 255.0.0.0 gw 157.55.80.1 metric 3 dev eth0
```

- La métrica y la interfaz son opcionales
- Internet se puede especificar como 0.0.0.0 o como default

4.5.2. Modificar las tablas en Linux (**ip**)

- Añadir una ruta para la red 157.0.0.0/8: se va por la interfaz eth0, con métrica 3. Se envía al router 157.55.80.1

```
sudo ip route add 157.0.0.0/8 via 157.55.80.1 dev eth0 weight 3
```

- La métrica y la interfaz son opcionales
- Internet se puede especificar como 0.0.0.0/0 o como default

4.5.3. Activar el enrutamiento

- El *ip forwarding* es un parámetro de kernel.
- Se puede comprobar si se está enrutando con
 - `sysctl net.ipv4.ip_forward`
- Se puede activar con
 - `sysctl -w net.ipv4.ip_forward=1`
- Para que se active **en el siguiente inicio** se debe cambiar el fichero `/etc/sysctl.conf`:
 - `net.ipv4.ip_forward = 1`

5. Referencias

- Formatos:
 - [Transparencias](#)
 - [PDF](#)
 - [Página web](#)
 - [EPUB](#)
- Creado con:
 - [Emacs](#)
 - [org-re-reveal](#)
 - [Latex](#)
- Alojado en [Github](#)