

1. Guión RIP

Alumnos: Juan José López Gómez y Sergio Sánchez García

Fecha: 22 de marzo de 2022

Duración estimada de la práctica: 1 sesión de 2h.

1.1. Cómo realizar un buen informe

- Al ir realizando todas las actividades que se proponen se han de documentar todos los pasos, adjuntando las órdenes (comandos o actividades) realizadas junto con las capturas de pantalla de su ejecución, explicando siempre la salida obtenida. No es suficiente con una captura de pantalla sin texto que acompañe a la imagen y lo contrario tampoco; es decir, adjuntar la orden (o comando) sin aportar prueba alguna de que se ejecutó realmente y sin una interpretación de la salida o resultado obtenido.
- En el informe se ha de mostrar que se han realizado y entendido todas las actividades propuestas.
- En la entrega de la práctica se han de adjuntar todos los ficheros auxiliares que hayan sido utilizados y que no se incluyan en el informe. Por ejemplo: el escenario final obtenido, los ficheros de capturas del tráfico de red, scripts realizados para automatizar tareas, etc.
- **Contestar en color verde para diferenciar claramente las respuestas del enunciado.**

1.2. Comprobando la configuración

En el fichero *RIPOSPF-02.zip* está definida una red como la que se muestra en la Figura 1. Descomprime el fichero en la carpeta correspondiente de GNS3.

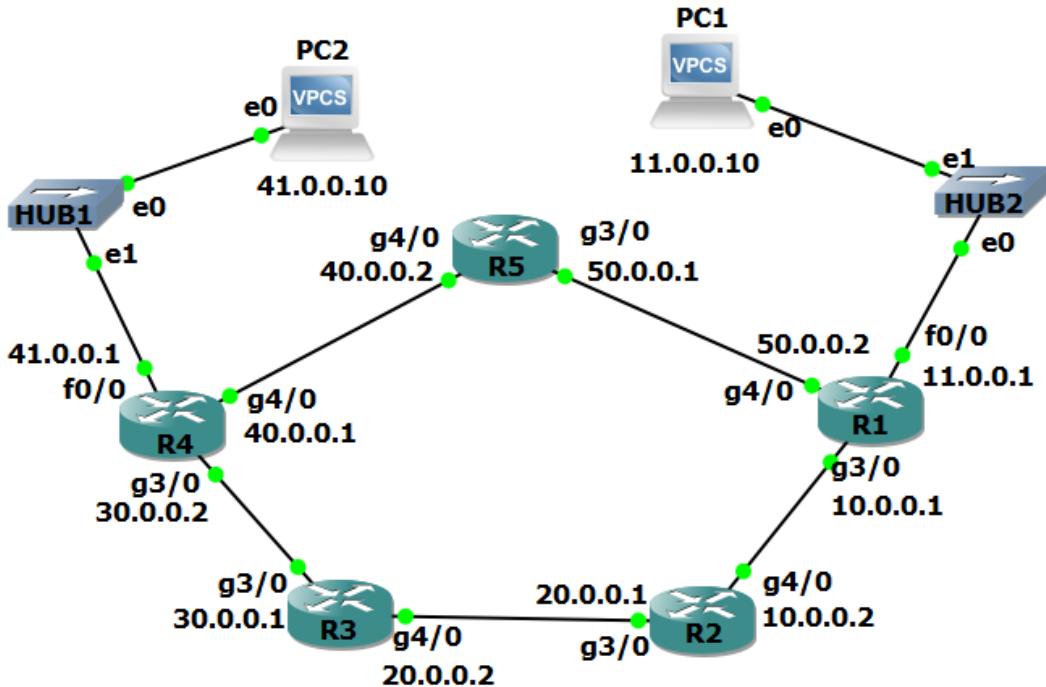


Figura 1: Escenario RIP y OSPF.

Arranca todas las máquinas y abre una consola con cada una de ellas. Los equipos PC2 y PC1 tienen rutas por defecto a R4 y R1 respectivamente. Compruébalo con la orden `show ip route` (Incluye aquí esa información).

```

PC1> show ip route
NAME      IP/MASK          GATEWAY          MAC          LPORT    RHOST:PORT
PC1      11.0.0.10/8        11.0.0.1        00:50:79:66:68:01 10082  127.0.0.1:10083
PC1>

```

En la foto podemos ver como la puerta por defecto de PC1 es la dirección IP 11.0.0.1 que es la interfaz f0/0 de R1.

```

PC2> show ip route
NAME      IP/MASK          GATEWAY          MAC          LPORT    RHOST:PORT
PC2      41.0.0.10/8        41.0.0.1        00:50:79:66:68:00 10084  127.0.0.1:10085
PC2>

```

En la foto podemos ver como la puerta por defecto de PC2 es la dirección IP 41.0.0.1 que es la interfaz f0/0 de R4.

Los routers no tienen configurada ninguna ruta, salvo la de las subredes a las que están directamente conectados. Compruébalo con la orden `show ip route` (Incluye aquí esa información).

Para consultar las tablas de rutas de los routers, tenemos que lanzar las CLI de cada router y una vez que se ha lanzado si en el prompt no saliese "#" habría que usar la orden `enable` pero en este caso no hace falta por lo que simplemente con el uso de `show ip route` se mostraría la tabla de rutas de cada uno de ellos.

```

R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
C   50.0.0.0/8 is directly connected, GigabitEthernet4/0
C   10.0.0.0/8 is directly connected, GigabitEthernet3/0
C   11.0.0.0/8 is directly connected, FastEthernet0/0
R1# 

R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
C   20.0.0.0/8 is directly connected, GigabitEthernet3/0
C   10.0.0.0/8 is directly connected, GigabitEthernet4/0
R2# 

R3#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
C   20.0.0.0/8 is directly connected, GigabitEthernet4/0
C   30.0.0.0/8 is directly connected, GigabitEthernet3/0
R3# 

R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
C   40.0.0.0/8 is directly connected, GigabitEthernet4/0
C   41.0.0.0/8 is directly connected, FastEthernet0/0
C   30.0.0.0/8 is directly connected, GigabitEthernet3/0
R4# 

R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
C   50.0.0.0/8 is directly connected, GigabitEthernet4/0
C   40.0.0.0/8 is directly connected, GigabitEthernet4/0
R5# 

```

Las tablas muestran las entradas de las subredes que tiene en entrega directa ya que en este punto no está configurado ningún protocolo para que pueda conocer a sus vecinos

R1 → En este caso las rutas son 30.0.0.0, 10.0.0.0 y 11.0.0.0

R2 → 20.0.0.0 y 10.0.0.0

R3 → 20.0.0.0 y 30.0.0.0

R4 → 40.0.0.0, 41.0.0.0 y 10.0.0.0

R5 → 40.0.0.0 y 50.0.0.0

En los siguientes apartados se configurará RIP en los routers para que las tablas de encaminamiento permitan alcanzar cualquier punto de la red.

1.3. Arrancando RIP en el primer router

Para observar los mensajes que envía R4 cuando se activa RIP, arranca *wireshark* en todos los enlaces de R4. A continuación configura RIP en R4 para que exporte las rutas de las tres redes a las que está conectado.

```

config t
router rip
version 2
network 30.0.0.0
network 40.0.0.0
network 41.0.0.0
no auto-summary
exit
exit
wr

```

Activa la depuración de los mensajes rip: `debug ip rip`.

```

R4#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#router rip
R4(config-router)#version 2
R4(config-router)#network 30.0.0.0
R4(config-router)#network 40.0.0.0
R4(config-router)#no auto-summary
R4(config-router)#exit
R4(config)#
R4#
Building configuration...
*Mar 22 18:32:06.203: %SYS-5-CONFIG_I: Configured from console[OK]
R4#debug ip rip
RIP protocol debugging is on
R4#
*Mar 22 18:32:29.591: RIP: sending v2 update to 224.0.0.9 via FastEthernet0/0 (41.0.0.1)
*Mar 22 18:32:29.591: RIP: build update entries
*Mar 22 18:32:29.591: 30.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:32:29.591: 40.0.0.0 via 0.0.0.0, metric 1, tag 0
R4#
*Mar 22 18:32:32.515: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet4/0 (40.0.0.1)
*Mar 22 18:32:32.515: RIP: build update entries
*Mar 22 18:32:32.515: 30.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:32:32.515: 41.0.0.0 via 0.0.0.0, metric 1, tag 0
R4#
*Mar 22 18:32:33.787: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet3/0 (30.0.0.2)
*Mar 22 18:32:33.787: RIP: build update entries
*Mar 22 18:32:33.787: 40.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:32:33.791: 41.0.0.0 via 0.0.0.0, metric 1, tag 0
R4#

```

Captura de la inicialización del modo debug en R4

Espera un minuto aproximadamente e interrumpe las capturas. Fíltralas por rip y guárdalas (*File/Sport Specified Packets...*) con un nombre significativo. Estas capturas y todas las que realices hasta completar el guión hay que adjuntarlas al informe y quedan para vuestro estudio.

En nuestro caso hemos decidido dejarlas en la carpeta que se almacenan por defecto (`/project-files/captures`).

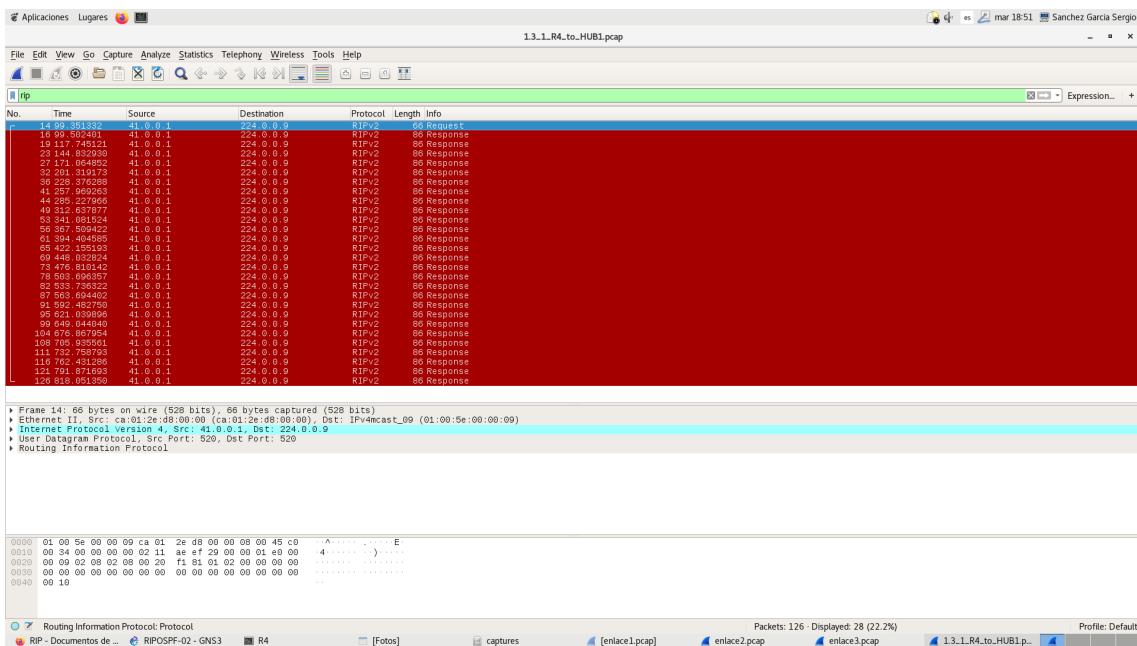
Interrumpe también los mensajes de depuración: no debug ip rip.

```

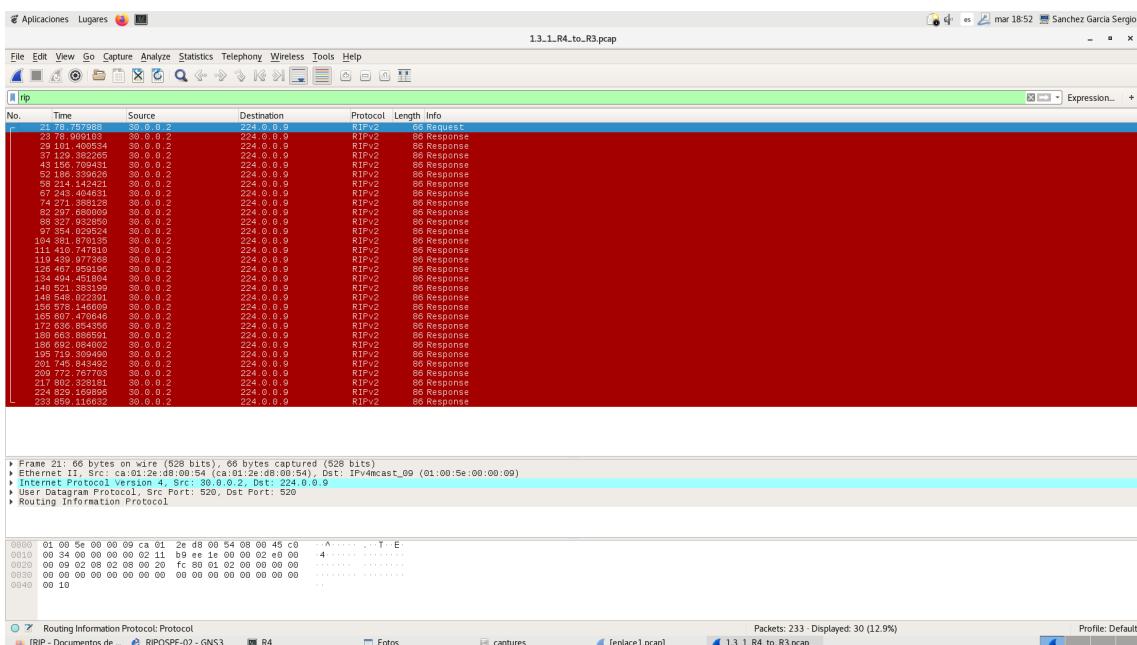
R4#
*Mar 22 18:33:22.259: RIP: sending v2 update to 224.0.0.9 via FastEthernet0/0 (41.0.0.1)
*Mar 22 18:33:22.259: RIP: build update entries
*Mar 22 18:33:22.259: 30.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:33:22.259: 40.0.0.0 via 0.0.0.0, metric 1, tag 0
R4#
*Mar 22 18:33:28.427: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet3/0 (30.0.0.2)
*Mar 22 18:33:28.427: RIP: build update entries
*Mar 22 18:33:28.427: 40.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:33:28.431: 41.0.0.0 via 0.0.0.0, metric 1, tag 0
R4#
*Mar 22 18:33:29.459: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet4/0 (40.0.0.1)
*Mar 22 18:33:29.459: RIP: build update entries
*Mar 22 18:33:29.459: 30.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:33:29.463: 41.0.0.0 via 0.0.0.0, metric 1, tag 0
R4#
*Mar 22 18:33:52.115: RIP: sending v2 update to 224.0.0.9 via FastEthernet0/0 (41.0.0.1)
*Mar 22 18:33:52.115: RIP: build update entries
*Mar 22 18:33:52.115: 30.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:33:52.115: 40.0.0.0 via 0.0.0.0, metric 1, tag 0
R4#
*Mar 22 18:33:57.783: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet3/0 (30.0.0.2)
*Mar 22 18:33:57.783: RIP: build update entries
*Mar 22 18:33:57.783: 40.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:33:57.787: 41.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:33:58.067: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet4/0 (40.0.0.1)
*Mar 22 18:33:58.067: RIP: build update entries
*Mar 22 18:33:58.067: 30.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:33:58.067: 40.0.0.0 via 0.0.0.0, metric 1, tag 0
*Mar 22 18:33:58.067: 41.0.0.0 via 0.0.0.0, metric 1, tag 0
R4#no debug ip rip
RIP protocol debugging is off
R4#

```

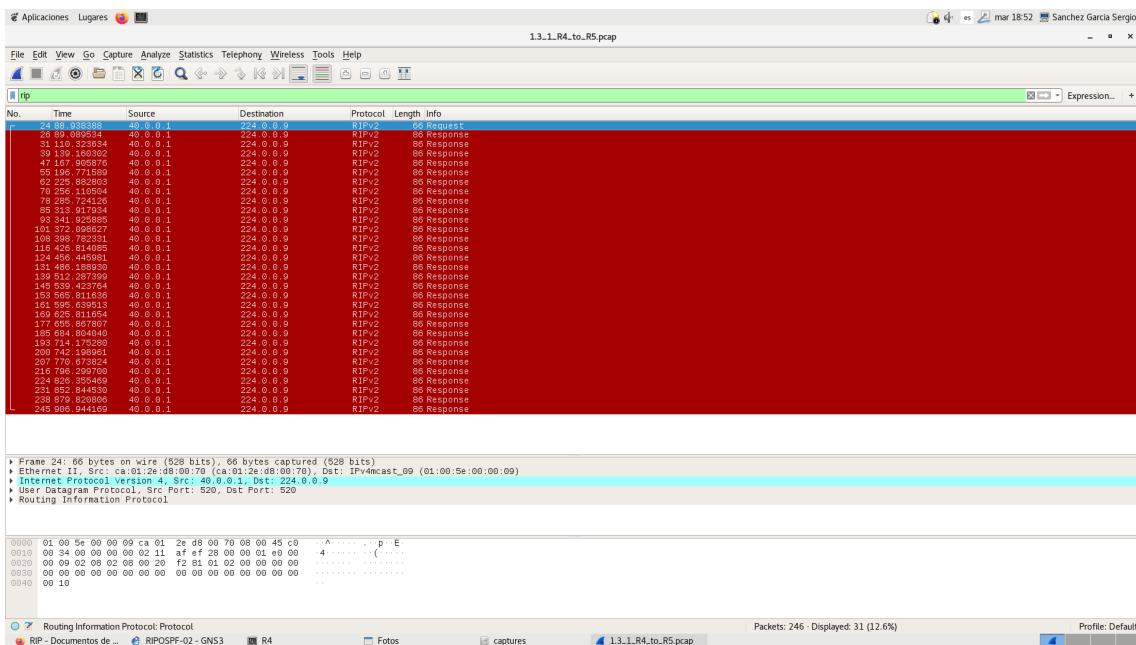
Aquí finalizamos la depuración de RIP



Captura del analizador de red en el enlace entre R4 y el HUB 1 por la subred 41.0.0.0 y la interfaz f0/0 del mismo

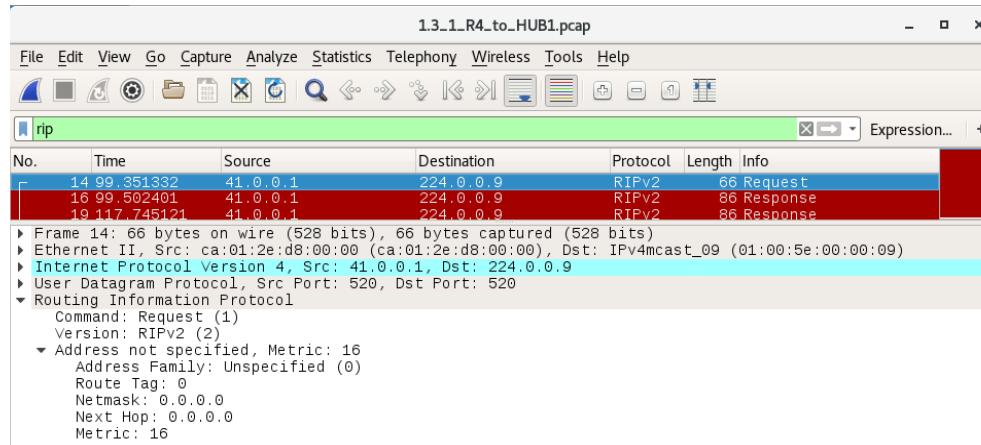


Captura del analizador de red en el enlace entre R4 y el R3 por la subred 30.0.0.0 y la interfaz g3/0 del mismo



Captura del analizador de red en el enlace entre R4 y R5 por la subred 40.0.0.0 y la interfaz g4/0 del mismo

1. Analiza el comportamiento de R4 estudiando las capturas del tráfico y los mensajes de depuración para responder a las siguientes preguntas:
 - a) Observa los mensajes REQUEST que se envían al arrancar RIP en R4 y analiza su contenido. ¿Son iguales en todas las interfaces? ¿Para qué se utilizan? ¿Qué rutas viajan en estos mensajes? ¿Quién responde a estos mensajes?



Mensaje RIP Request que envía R4 al HUB1 al arrancar.

1.3_1_R4_to_R3.pcap						
No.	Time	Source	Destination	Protocol	Length	Info
21	78.757988	30.0.0.2	224.0.0.9	RIPv2	66	Request
23	78.909103	30.0.0.2	224.0.0.9	RIPv2	86	Response
29	101.400534	30.0.0.2	224.0.0.9	RIPv2	86	Response

Frame 21: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
Ethernet II, Src: ca:01:2e:d8:00:54 (ca:01:2e:d8:00:54), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
Internet Protocol Version 4, Src: 30.0.0.2, Dst: 224.0.0.9
User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
Command: Request (1)
Version: RIPv2 (2)
Address not specified, Metric: 16
Address Family: Unspecified (0)
Route Tag: 0
Netmask: 0.0.0.0
Next Hop: 0.0.0.0
Metric: 16

Mensaje RIP Request que envía R4 a R3 al arrancar.

1.3_1_R4_to_R5.pcap						
No.	Time	Source	Destination	Protocol	Length	Info
24	88.938388	40.0.0.1	224.0.0.9	RIPv2	66	Request
26	89.089534	40.0.0.1	224.0.0.9	RIPv2	86	Response
31	110.323634	40.0.0.1	224.0.0.9	RIPv2	86	Response

Frame 24: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
Ethernet II, Src: ca:01:2e:d8:00:70 (ca:01:2e:d8:00:70), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
Internet Protocol Version 4, Src: 40.0.0.1, Dst: 224.0.0.9
User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
Command: Request (1)
Version: RIPv2 (2)
Address not specified, Metric: 16
Address Family: Unspecified (0)
Route Tag: 0
Netmask: 0.0.0.0
Next Hop: 0.0.0.0
Metric: 16

Mensaje RIP Request que envía R4 a R5 al arrancar.

Hay un único mensaje request (command: 1) de RIP por cada enlace, estos son iguales en todas las interfaces en cuanto a dirección IP de destino, ya que cuando un terminal activa el protocolo RIP automáticamente se subscribe a la dirección multicast 224.0.0.9 y en cuánto a contenido también ya que está propagando su suscripción a este protocolo para que sus vecinos le envíen sus tablas de rutas, que en este caso por el momento no recibirá nada ya que no hay ningún router más activo.

La familia de direcciones y la métrica de este protocolo (que es la distancia en este caso), al ser un mensaje de request tienen valores por defecto.

La familia de direcciones a 0 significa que está asignada pero no especificada y se encuentra asociada a un intercambio de tabla de rutas completo.

La métrica es 16 y este valor está asociado a un destino inalcanzable, que en esta ocasión no importa pero más adelante en el guión tendrá su relevancia.

que rutas viajan

quien responde

- b) Observa los mensajes RESPONSE que envía R4 periódicamente a través de cada una de sus interfaces. ¿Son iguales en todas las interfaces? ¿Qué rutas viajan en estos mensajes?

En cuanto a los mensajes de RESPONSE la IP de destino en el nivel de red si que es la misma ya que el mensaje de RESPONSE que se está generando es el de actualización periódica, esto se puede saber porque el intervalo en que se mandan estos mensajes son 30 s + 15s.

No.	Time	Source	Destination	Protocol	Length	Info
5	43.011363	30.0.0.2	224.0.0.9	RIPv2	86	Response
13	75.088794	30.0.0.2	224.0.0.9	RIPv2	86	Response
18	104.559922	30.0.0.2	224.0.0.9	RIPv2	86	Response

```

Frame 18: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface 0
Ethernet II, Src: ca:01:2e:d8:00:54 (ca:01:2e:d8:00:54), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
Internet Protocol Version 4, Src: 30.0.0.2, Dst: 224.0.0.9
User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
  Command: Response (2)
  Version: RIPv2 (2)
  IP Address: 40.0.0.0, Metric: 1
  IP Address: 41.0.0.0, Metric: 1

```

Mientras que el contenido de cada una es diferente, si observamos el mensaje response que se envía por la subred 30.0.0.0 (a través de 30.0.0.2) el contenido del mismo son las subredes a las que están conectadas las demás interfaces del router 40.0.0.0 y 41.0.0.0

No.	Time	Source	Destination	Protocol	Length	Info
3	0.317937	40.0.0.1	224.0.0.9	RIPv2	86	Response
9	31.349904	40.0.0.1	224.0.0.9	RIPv2	86	Response
14	58.795658	40.0.0.1	224.0.0.9	RIPv2	86	Response
20	89.249407	40.0.0.1	224.0.0.9	RIPv2	86	Response
25	119.706583	40.0.0.1	224.0.0.9	RIPv2	86	Response

```

Ethernet II, Src: ca:01:2e:d8:00:70 (ca:01:2e:d8:00:70), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
Internet Protocol Version 4, Src: 40.0.0.1, Dst: 224.0.0.9
User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
  Command: Response (2)
  Version: RIPv2 (2)
  IP Address: 30.0.0.0, Metric: 1
  IP Address: 41.0.0.0, Metric: 1

```

Por la subred 40.0.0.0 (a través de 40.0.0.1) se envía 30.0.0.0 y 41.0.0.0, que coinciden con las subredes de f0/0 y g3/0

No.	Time	Source	Destination	Protocol	Length	Info
5	26.424561	41.0.0.1	224.0.0.9	RIPv2	86	Response
9	53.906358	41.0.0.1	224.0.0.9	RIPv2	86	Response
13	83.720702	41.0.0.1	224.0.0.9	RIPv2	86	Response
17	113.018742	41.0.0.1	224.0.0.9	RIPv2	86	Response

```

Ethernet II, Src: ca:01:2e:d8:00:00 (ca:01:2e:d8:00:00), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
Internet Protocol Version 4, Src: 41.0.0.1, Dst: 224.0.0.9
User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
  Command: Response (2)
  Version: RIPv2 (2)
  IP Address: 30.0.0.0, Metric: 1
  IP Address: 40.0.0.0, Metric: 1

```

Por la subred 41.0.0.0 (a través de 41.0.0.1) se envía 30.0.0.0 y 40.0.0.0, que coinciden con las subredes de f0/0 y g4/0

- c) ¿Debería haber aprendido alguna ruta R4? Compruébalo consultando la tabla de encaminamiento mediante la orden show ip route.

No se ha aprendido ninguna ruta ya que el router R4 es el único que está suscrito a RIP, por lo que es el único que envía los mensajes de Response con las tablas predeterminadas que tenía configuradas por defecto

```

R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    40.0.0.0/8 is directly connected, GigabitEthernet4/0
C    41.0.0.0/8 is directly connected, FastEthernet0/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R4#

```

Y como se observa no se ha aprendido nada ya que las únicas entradas que tiene son las que hemos configurado al iniciar RIP

- d) Ejecuta en R4 la orden show ip protocols. ¿Qué información nos facilita? Adjunta y comenta la salida obtenida.

```

R4#show ip protocols
Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 19 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface          Send  Recv Triggered RIP Key-chain
      FastEthernet0/0   2      2
      GigabitEthernet3/0 2      2
      GigabitEthernet4/0 2      2
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    30.0.0.0
    40.0.0.0
    41.0.0.0
  Routing Information Sources:
    Gateway          Distance      Last Update
    Distance: (default is 120)
R4#

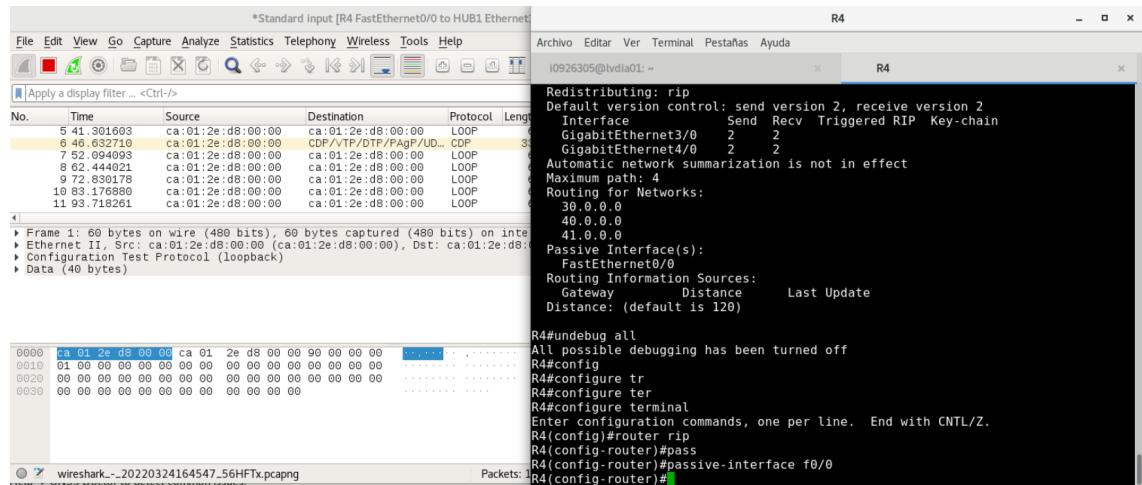
```

Al utilizar la orden *show ip protocols* se muestra la captura anterior. En ella nos muestra que el protocolo de encaminamiento que sigue este router es RIP, que por definición el “sondeo” de actualizaciones son 30 y en el momento de hacer la captura el próximo mensaje va a ser en 19 s, además del tiempo que las entradas van a permanecer en la tabla si no recibe actualizaciones de ellas que en este caso son 240s, muestra las interfaces activas del router y los mensajes que ha recibido y enviado que en este caso son dos para todas.

Seguidamente de en el apartado “Routing for Networks” muestra el contenido de su tabla de rutas, y en el siguiente apartado estarían las entradas que le llegarían de routers vecinos

2. ¿Para qué sirve la orden *passive-interface <interface>*? ¿Sería útil configurar en R4 alguna de sus interfaces de este modo? Incluye aquí la orden.

La orden *passive-interface* sirve para deshabilitar el envío de mensajes de configuración por la interfaz dada por parámetro. En nuestra situación sería útil hacerlo por la interfaz f0/0 (subred 41.0.0.0) ya que en esa subred no tenemos ningún router por lo que el envío de mensajes de protocolo de encaminamiento dinámico no tendría sentido y solo serviría para sobrecargar la subred.



Para poder configurar la interfaz de este modo tenemos que establecer el router en modo configuración con *configure terminal*, después acceder a la configuración del protocolo RIP con *router rip* una vez aquí podremos configurar la interfaz con la orden *passive-interface f0/0*. Cómo se observa en la captura del wireshark que aunque no esté filtrado por RIP deberían salir los mensajes de Response por esa interfaz, aún pasando más del tiempo no se observa ningún mensaje de este tipo

¿En qué otros routers del escenario se podría utilizar esta orden? Úsala cuando configures RIP en ellos.

Se podría utilizar en el router R1 por la interfaz f0/0 también ya que estamos en el mismo caso que en el router R4 y es en la única interfaz en la que no va a haber ningún otro router por lo que no es necesario que los mensajes del protocolo RIP sean enviados por ella. En todos los demás habría que dejar tal y como están las interfaces ya que todas están unidas a otros routers y si se bloquea una no sería correcto

1.4. Arrancado RIP en los routers de la parte inferior

- Ve configurando y arrancando a continuación RIP en los encaminadores R3, R2 y R1, de uno en uno: primero en R3, luego en R2 y finalmente en R1. Cada vez que configures uno de esos encaminadores realiza las capturas de tráfico que estimes conveniente, estudiando junto con los mensajes de depuración y responde a las siguientes cuestiones:

a) Arranque de R3

- Incluye las órdenes de configuración.

```

config t
router rip
version 2
network 30.0.0.0
network 20.0.0.0
no auto-summary
exit
exit
wr

```

```

R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#router rip
R3(config-router)#ver
R3(config-router)#version 2
R3(config-router)#network 30.0.0.0
R3(config-router)#network 20.0.0.0
R3(config-router)#no auto-summary
R3(config-router)#exit
R3(config)#exit
R3#
*Mar 24 16:33:31.591: %SYS-5-CONFIG_I: Configured from console by console
R3#wr
Building configuration...
[OK]
R3#debug ip rip

```

2. Comprueba el envío de mensajes REQUEST. ¿Existe algún mensaje de RESPONSE a esos REQUEST? ¿Por qué?

No.	Time	Source	Destination	Protocol	Length	Info
6	25.293855	20.0.0.2	224.0.0.9	RIPv2	66	Request
11	47.479350	20.0.0.2	224.0.0.9	RIPv2	86	Response
13	48.487052	20.0.0.2	224.0.0.9	RIPv2	106	Response
19	78.358280	20.0.0.2	224.0.0.9	RIPv2	106	Response


```

Ethernet II, Src: ca:02:03:d8:00:70 (ca:02:03:d8:00:70), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
Internet Protocol Version 4, Src: 20.0.0.2, Dst: 224.0.0.9
User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
    Command: Response (2)
    Version: RIPv2 (2)
    IP Address: 30.0.0.0, Metric: 1
    IP Address: 40.0.0.0, Metric: 2
    IP Address: 41.0.0.0, Metric: 2

```

Primero vamos a analizar los mensajes de la interfaz por la que no está configurado el siguiente router, por lo que pasa lo mismo que cuando el router R4 estaba solo configurado su mensaje de Request para los demás routers no va a ser respondido y cuando se expire el contador periódico va a enviar su tabla de rutas predeterminada.

No.	Time	Source	Destination	Protocol	Length	Info
3	5.059815	30.0.0.2	224.0.0.9	RIPv2	86	Response
11	33.352622	30.0.0.2	224.0.0.9	RIPv2	86	Response
18	61.284709	30.0.0.2	224.0.0.9	RIPv2	86	Response
20	62.873819	30.0.0.1	224.0.0.9	RIPv2	66	Request
21	62.883847	30.0.0.2	30.0.0.1	RIPv2	86	Response
26	81.577892	30.0.0.1	224.0.0.9	RIPv2	66	Response
28	88.302258	30.0.0.2	224.0.0.9	RIPv2	86	Response
31	93.120568	30.0.0.1	224.0.0.9	RIPv2	66	Response


```

Frame 21: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface 0
Ethernet II, Src: ca:01:2e:d8:00:54 (ca:01:2e:d8:00:54), Dst: ca:02:03:d8:00:54 (ca:02:03:d8:00:54)
Internet Protocol Version 4, Src: 30.0.0.2, Dst: 30.0.0.1
User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
    Command: Response (2)
    Version: RIPv2 (2)
    IP Address: 40.0.0.0, Metric: 1
    IP Address: 41.0.0.0, Metric: 1

```

En cuanto a la otra interfaz al mensaje de request si que el mensaje de Request tiene una respuesta asociada como se puede observar, que el mensaje de Response inmediatamente siguiente al de Request es un mensaje unicast dirigido a R3 en el que se envía desde R4 su tabla de rutas a enviar por esta interfaz que en este caso es 40.0.0.0 y 41.0.0.0, con métrica 1, lo que significa que estas subredes son de entrega directa a partir de R4.

3. Comprueba la tabla de encaminamiento del encaminador recién arrancado, así como las tablas de encaminamiento del resto de los encaminadores para ver cómo se van propagando las rutas. Presta especial atención a la métrica de las rutas aprendidas y explica su valor. Explica el proceso de aprendizaje de rutas apoyándote en las capturas realizadas y en los mensajes de depuración.

```

R3#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    20.0.0.0/8 is directly connected, GigabitEthernet4/0
R    40.0.0.0/8 [120/1] via 30.0.0.2, 00:00:01, GigabitEthernet3/0
R    41.0.0.0/8 [120/1] via 30.0.0.2, 00:00:01, GigabitEthernet3/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R3#

```

Al realizar el comando *show ip route* en R3, se observa la tabla de rutas del router, en ella podemos observar como a las dos que tenía anteriormente (identificadas por la C) se han añadido dos más, 40.0.0.0/8 y 41.0.0.0/8 ambas por la interfaz g3/0 que corresponde con la dirección ip 30.0.0.2, el penúltimo parámetro es un contador que llegará como máximo a 180s que en ese momento se eliminará la entrada de la tabla, pero que en el momento de la captura se acababa de actualizar la entrada (por los mensajes de actualización periódica del protocolo). Y el último parámetro a comentar el es valor entre corchetes, de este importa el valor tras la "/" ya que identifica el valor de la métrica que tiene esa entrada en este caso es 1 por lo que las subredes son de entrega directa desde el router que está conectado a través de la interfaz g3/0

```

R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:05, GigabitEthernet3/0
C    40.0.0.0/8 is directly connected, GigabitEthernet4/0
C    41.0.0.0/8 is directly connected, FastEthernet0/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R4#

```

Si se realiza el mismo procedimiento pero en R4, se observa que éste ha aprendido la ruta 20.0.0.0/8 por la interfaz g3/0 a la que le corresponde la IP 30.0.0.1, está entrada lleva almacenada 5s en el momento de la captura, y la métrica de la misma es 1 ya que la subred 20.0.0.0 es de entrega directa a través de R3

Comentando los mensajes de depuración generados por el router R3 para observar cómo aprende y envía las rutas:

Primeramente por la interfaz g3/0 envía la ruta 20.0.0.0/8 con métrica uno que es la que va a recibir el router R4, como hemos visto anteriormente. No envía en las siguientes iteraciones el valor aprendido por esta ruta porque se da por supuesto que el emisor de las mismas las conoce

```

Mar 24 16:52:14.611: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet3/0 (30.0.0.1)
*Mar 24 16:52:14.611: RIP: build update entries
*Mar 24 16:52:14.611:    20.0.0.0/8 via 0.0.0.0, metric 1, tag 0
R3#

```

Seguidamente por la interfaz g4/0 envía la tabla de rutas con las entradas que no están en ese lado del ejemplo, con métrica 1 se manda la subred 30.0.0.0/8 ya que está en entrega directa con R3 mientras que las otras dos entradas tienen métrica 2 ya que son las que ha recibido desde R4

```
*Mar 24 16:52:27.095: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet4/0 (20.0.0.2)
*Mar 24 16:52:27.095: RIP: build update entries
*Mar 24 16:52:27.095:   30.0.0.0/8 via 0.0.0.0, metric 1, tag 0
*Mar 24 16:52:27.099:   40.0.0.0/8 via 0.0.0.0, metric 2, tag 0
*Mar 24 16:52:27.099:   41.0.0.0/8 via 0.0.0.0, metric 2, tag 0
R3#
```

Y por último el mensaje de actualización periódico que envía R4 hacia la subred 30.0.0.0, en el que envía las entradas de las otras dos subredes que tiene en entrega directa, en este caso, pero si tuviese más entradas aprendidas desde otros lados también se enviarán cambiando el número de saltos, que en este caso es 1 por lo que hemos mencionado anteriormente que estas subredes están en entrega directa con R4

```
R3#
*Mar 24 16:52:36.627: RIP: received v2 update from 30.0.0.2 on GigabitEthernet3/0
*Mar 24 16:52:36.627:   40.0.0.0/8 via 0.0.0.0 in 1 hops
*Mar 24 16:52:36.631:   41.0.0.0/8 via 0.0.0.0 in 1 hops
R3#
```

b) Arranque de R2

1. Incluye las órdenes de configuración.

```
config t
router rip
version 2
network 20.0.0.0
network 10.0.0.0
no auto-summary
exit
exit
wr
debug ip rip
```

2. Comprueba el envío de mensajes REQUEST. ¿Existe algún mensaje de RESPONSE a esos REQUEST? ¿Por qué?

The screenshot shows a Wireshark capture window titled 'rip'. The main pane displays a list of network packets. The selected packet is number 11, timestamped at 46.726319, source 10.0.0.2, destination 224.0.0.9, protocol RIPV2, length 106, and info 'Response'. Below the list, the packet details pane shows the raw hex and ASCII data for this response. The bytes pane shows the raw binary data. A tooltip for the selected packet indicates: 'Internet Protocol Version 4, Src: 10.0.0.2, Dst: 224.0.0.9' and 'User Datagram Protocol, Src Port: 520, Dst Port: 520'. The expanded details pane shows the RIP message structure: Command: Response (2), Version: RIPV2 (2), IP Address: 30.0.0.0, Metric: 2, IP Address: 40.0.0.0, Metric: 3, and IP Address: 41.0.0.0, Metric: 3.

Primero vamos a analizar los mensajes de la interfaz por la que no está configurado el siguiente router, por lo que pasa lo mismo que cuando el router R4 estaba solo configurado su mensaje de Request para los demás routers no va a ser respondido y cuando se expire el contador periódico va a enviar su tabla de rutas predeterminada.

En este caso en los Response le envía las rutas que ha aprendido por la otra interfaz de los demás routers.

No.	Time	Source	Destination	Protocol	Length	Info
6 26 408588		20.0.0.1	224.0.0.9	RIPv2	66	Request
7 26 417534		20.0.0.2	20.0.0.1	RIPv2	106	Response
10 38 848966		20.0.0.2	224.0.0.9	RIPv2	106	Response
11 39 678715		20.0.0.1	224.0.0.9	RIPv2	66	Response
14 49 277271		20.0.0.1	224.0.0.9	RIPv2	66	Response
19 77 309169		20.0.0.1	224.0.0.9	RIPv2	66	Response

Frame 7: 106 bytes on wire (848 bits), 106 bytes captured (848 bits) on interface 0
 ▶ Ethernet II, Src: ca:02:03:d8:00:70 (ca:02:03:d8:00:70), Dst: ca:03:04:f4:00:54 (ca:03:04:f4:00:54)
 ▶ Internet Protocol Version 4, Src: 20.0.0.2, Dst: 20.0.0.1
 ▶ User Datagram Protocol, Src Port: 520, Dst Port: 520
 ▶ Routing Information Protocol
 Command: Response (2)
 Version: RIPv2 (2)
 IP Address: 30.0.0.0, Metric: 1
 IP Address: 40.0.0.0, Metric: 2
 IP Address: 41.0.0.0, Metric: 2

En cuanto a la otra interfaz al mensaje de Request sí que tiene una respuesta asociada como se puede observar, que el mensaje de Response inmediatamente siguiente al de Request es un mensaje unicast dirigido a R2 en el que se envía desde R3 su tabla de rutas a enviar por esta interfaz que en este caso es 30.0.0.0 con métrica 1 (entrega directa desde R3), y 40.0.0.0, 41.0.0.0 con métrica 2 ya que son de entrega directa pero desde R4 que es el router siguiente a R3

3. Comprueba la tabla de encaminamiento del encaminador recién arrancado, así como las tablas de encaminamiento del resto de los encaminadores para ver cómo se van propagando las rutas. Presta especial atención a la métrica de las rutas aprendidas y explica su valor. Explica el proceso de aprendizaje de rutas apoyándote en las capturas realizadas y en los mensajes de depuración.

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    20.0.0.0/8 is directly connected, GigabitEthernet3/0
R    40.0.0.0/8 [120/2] via 20.0.0.2, 00:00:32, GigabitEthernet3/0
C    10.0.0.0/8 is directly connected, GigabitEthernet4/0
R    41.0.0.0/8 [120/2] via 20.0.0.2, 00:00:32, GigabitEthernet3/0
R    30.0.0.0/8 [120/1] via 20.0.0.2, 00:00:33, GigabitEthernet3/0
R2#
```

Al realizar el comando *show ip route* en R2, se observa la tabla de rutas del router, en ella podemos observar como a las dos que tenía anteriormente (identificadas por la C) se han añadido tres más, 40.0.0.0/8, 41.0.0.0.0/8, 30.0.0.0/8 todas por la interfaz g3/0 que corresponde con la dirección ip 20.0.0.2, el penúltimo parámetro es un contador que llegará como máximo a 180s que en ese momento se eliminará la entrada de la tabla, pero que en el momento de la captura llevaban en la caché apenas medio minuto. Y el último parámetro a comentar el es valor entre corchetes, de este importa el valor tras la "/" ya que identifica el valor de la métrica que tiene esa entrada, en este caso podemos observar como 40.0.0.0 y 41.0.0.0 tiene una métrica 2 ya que se necesitan dos saltos para alcanzarlas ya que son subredes de entrega directa de R4, mientras que 30.0.0.0 tiene métrica 1 porque es de entrega directa pero de R3

```

R3#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    20.0.0.0/8 is directly connected, GigabitEthernet4/0
R    40.0.0.0/8 [120/1] via 30.0.0.2, 00:00:07, GigabitEthernet3/0
R    10.0.0.0/8 [120/1] via 20.0.0.1, 00:00:08, GigabitEthernet4/0
R    41.0.0.0/8 [120/1] via 30.0.0.2, 00:00:07, GigabitEthernet3/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R3#

```

Si se realiza el mismo procedimiento pero en R3, se observa que éste ha aprendido la ruta 10.0.0.0/8 por la interfaz g3/0 a la que le corresponde la IP 20.0.0.1, está entrada lleva almacenada 8s en el momento de la captura, y la métrica de la misma es 1 ya que la subred 10.0.0.0 es de entrega directa a través de R2

```

R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:01, GigabitEthernet3/0
C    40.0.0.0/8 is directly connected, GigabitEthernet4/0
R    10.0.0.0/8 [120/2] via 30.0.0.1, 00:00:01, GigabitEthernet3/0
C    41.0.0.0/8 is directly connected, FastEthernet0/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R4#

```

Si se realiza el mismo procedimiento pero en R4, se observa que éste ha aprendido la ruta 10.0.0.0/8 por la interfaz g3/0 a la que le corresponde la IP 30.0.0.1, está entrada lleva almacenada 1s en el momento de la captura, y la métrica de la misma es 2 debido a que para llegar a ella tiene que ser a través de R3 y R2

Comentando los mensajes de depuración generados por el router R2 para observar cómo aprende y envía las rutas:

Primeramente por la interfaz g3/0 envía la ruta 10.0.0.0/8 con métrica uno que es la que va a recibir el router R3, como hemos visto anteriormente. No envía en las siguientes iteraciones el valor aprendido por esta ruta porque se da por supuesto que el emisor de las mismas las conoce

```

R2#
*Mar 25 12:07:00.679: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet3/0 (20.0.0.1)
*Mar 25 12:07:00.679: RIP: build update entries
*Mar 25 12:07:00.679:    10.0.0.0/8 via 0.0.0.0, metric 1, tag 0
R2#

```

Seguidamente por la interfaz g4/0 envía la tabla de rutas con las entradas que no están en ese lado del ejemplo, con métrica 1 se manda la subred 20.0.0.0/8 ya que está en entrega directa con R2, la subred 30.0.0.0 se envía con métrica 2 porque ha sido aprendida mediante el envío desde R3,

mientras que las otras dos entradas tienen métrica 3 ya que son las que ha recibido desde R4

```
R2#  
*Mar 25 12:07:10.611: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet4/0 (10.0.0.2)  
*Mar 25 12:07:10.611: RIP: build update entries  
*Mar 25 12:07:10.611:   20.0.0.0/8 via 0.0.0.0, metric 1, tag 0  
*Mar 25 12:07:10.615:   30.0.0.0/8 via 0.0.0.0, metric 2, tag 0  
*Mar 25 12:07:10.615:   40.0.0.0/8 via 0.0.0.0, metric 3, tag 0  
*Mar 25 12:07:10.615:   41.0.0.0/8 via 0.0.0.0, metric 3, tag 0  
R2#
```

Y por último el mensaje de actualización periódico que envía R3 hacia la subred 20.0.0.0, en el que envía las entradas de las otras dos subredes que tiene en entrega directa, en este caso, pero si tuviese más entradas aprendidas desde otros lados también se enviarán cambiando el número de saltos, que en este caso es 2 por lo que hemos mencionado anteriormente que estas subredes están en entrega directa con R4

```
R2#  
*Mar 25 12:07:27.063: RIP: received v2 update from 20.0.0.2 on GigabitEthernet3/0  
*Mar 25 12:07:27.063:   30.0.0.0/8 via 0.0.0.0 in 1 hops  
*Mar 25 12:07:27.067:   40.0.0.0/8 via 0.0.0.0 in 2 hops  
*Mar 25 12:07:27.067:   41.0.0.0/8 via 0.0.0.0 in 2 hops  
R2#
```

c) Arranque de R1

1. Incluye las órdenes de configuración.

```
config t
router rip
version 2
network 10.0.0.0
network 11.0.0.0
network 50.0.0.0
no auto-summary
exit
exit
wr
debug ip rip
```

2. Comprueba el envío de mensajes REQUEST. ¿Existe algún mensaje de RESPONSE a esos REQUEST? ¿Por qué?

rip

No.	Time	Source	Destination	Protocol	Length	Info
13	81.988144	50.0.0.2	224.0.0.9	RIPv2	66	Request
17	103.548106	50.0.0.2	224.0.0.9	RIPv2	86	Response
20	111.912574	50.0.0.2	224.0.0.9	RIPv2	126	Response
25	130.253613	50.0.0.2	224.0.0.9	RIPv2	166	Response
31	159.133012	50.0.0.2	224.0.0.9	RIPv2	166	Response
36	187.678145	50.0.0.2	224.0.0.9	RIPv2	166	Response

> Ethernet II, Src: ca:04:1a:18:00:70 (ca:04:1a:18:00:70), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
> Internet Protocol Version 4, Src: 50.0.0.2, Dst: 224.0.0.9
> User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
Command: Response (2)
Version: RIPv2 (2)
IP Address: 10.0.0.0, Metric: 1
IP Address: 11.0.0.0, Metric: 1

0000 01 00 50 00 00 09 ca 04 1a 18 00 70 08 00 45 c0 ..^.....p-E.
0010 00 48 00 00 00 00 02 11 a5 da 32 00 00 02 e0 00 .H.....2.....
0020 00 09 02 08 02 08 00 34 d4 60 02 02 00 00 00 024`.....
0030 00 00 0a 00 00 00 ff 00 00 00 00 00 00 00 00 00
0040 00 01 00 02 00 00 0b 00 00 00 ff 00 00 00 00 00 00
0050 00 00 00 00 00 00 01

En la interfaz g4/0 al hacer la captura de tráfico filtrando por el protocolo RIP no encontramos lo que cabría esperar ya que no hay ningún otro router suscrito a la dirección Multicast 224.0.0.9 en esa parte del modelo. Por lo que los mensajes de Response son únicamente de esta misma interfaz, en la primera captura solo se enviaron las rutas que hemos configurado desde el primer momento

rip

No.	Time	Source	Destination	Protocol	Length	Info
13	81.988144	50.0.0.2	224.0.0.9	RIPv2	66	Request
17	103.548106	50.0.0.2	224.0.0.9	RIPv2	86	Response
20	111.912574	50.0.0.2	224.0.0.9	RIPv2	126	Response
25	130.253613	50.0.0.2	224.0.0.9	RIPv2	166	Response
31	159.133012	50.0.0.2	224.0.0.9	RIPv2	166	Response
36	187.678145	50.0.0.2	224.0.0.9	RIPv2	166	Response
41	213.962574	50.0.0.2	224.0.0.9	RIPv2	166	Response

> Frame 25: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits)
> Ethernet II, Src: ca:04:1a:18:00:70 (ca:04:1a:18:00:70), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
> Internet Protocol Version 4, Src: 50.0.0.2, Dst: 224.0.0.9
> User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
Command: Response (2)
Version: RIPv2 (2)
IP Address: 10.0.0.0, Metric: 1
IP Address: 11.0.0.0, Metric: 1
IP Address: 20.0.0.0, Metric: 2
IP Address: 30.0.0.0, Metric: 3
IP Address: 40.0.0.0, Metric: 4
IP Address: 41.0.0.0, Metric: 4

Si dejamos que pase el tiempo y volvemos a mostrar el contenido de los mensajes podemos observar que al haber recibido ya la información por las otras interfaces puede enviar correctamente todas las rutas que tienen en su caché

No.	Time	Source	Destination	Protocol	Length	Info
9	75.569874	11.0.0.1	224.0.0.9	RIPv2	66	Request
11	84.804884	11.0.0.1	224.0.0.9	RIPv2	66	Response
13	95.872493	11.0.0.1	224.0.0.9	RIPv2	86	Response
16	112.708289	11.0.0.1	224.0.0.9	RIPv2	126	Response
19	123.956745	11.0.0.1	224.0.0.9	RIPv2	166	Response
23	150.895116	11.0.0.1	224.0.0.9	RIPv2	166	Response

```

> Frame 13: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface -, id 0
> Ethernet II, Src: ca:04:1a:18:00:00 (ca:04:1a:18:00:00), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
> Internet Protocol Version 4, Src: 11.0.0.1, Dst: 224.0.0.9
> User Datagram Protocol, Src Port: 520, Dst Port: 520
< Routing Information Protocol
  Command: Response (2)
  Version: RIPv2 (2)
  > IP Address: 10.0.0.0, Metric: 1
  > IP Address: 50.0.0.0, Metric: 1

```

En la interfaz f0/0 al hacer la captura de tráfico filtrando por el protocolo RIP no encontramos lo que cabría esperar ya que no hay ningún otro router suscrito a la dirección Multicast 224.0.0.9 en esa parte del modelo. Por lo que los mensajes de Response son únicamente de esta misma interfaz, en la primera captura solo se enviaron las rutas que hemos configurado desde el primer momento

No.	Time	Source	Destination	Protocol	Length	Info
9	75.569874	11.0.0.1	224.0.0.9	RIPv2	66	Request
11	84.804884	11.0.0.1	224.0.0.9	RIPv2	66	Response
13	95.872493	11.0.0.1	224.0.0.9	RIPv2	86	Response
16	112.708289	11.0.0.1	224.0.0.9	RIPv2	126	Response
19	123.956745	11.0.0.1	224.0.0.9	RIPv2	166	Response
23	150.895116	11.0.0.1	224.0.0.9	RIPv2	166	Response
26	177.761426	11.0.0.1	224.0.0.9	RIPv2	166	Response

```

> Frame 19: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits)
> Ethernet II, Src: ca:04:1a:18:00:00 (ca:04:1a:18:00:00), Dst: IPv4mcast_09 (01:00:5e:00:00:09)
> Internet Protocol Version 4, Src: 11.0.0.1, Dst: 224.0.0.9
> User Datagram Protocol, Src Port: 520, Dst Port: 520
< Routing Information Protocol
  Command: Response (2)
  Version: RIPv2 (2)
  > IP Address: 10.0.0.0, Metric: 1
  > IP Address: 20.0.0.0, Metric: 2
  > IP Address: 30.0.0.0, Metric: 3
  > IP Address: 40.0.0.0, Metric: 4
  > IP Address: 41.0.0.0, Metric: 4
  > IP Address: 50.0.0.0, Metric: 1

```

Si dejamos que pase el tiempo y volvemos a mostrar el contenido de los mensajes podemos observar que al haber recibido ya la información por las otras interfaces puede enviar correctamente todas las rutas que tienen en su caché

No.	Time	Source	Destination	Protocol	Length	Info
11	54.001474	10.0.0.1	224.0.0.9	RIPv2	66	Request
12	54.009476	10.0.0.2	10.0.0.1	RIPv2	126	Response
14	55.846611	10.0.0.1	224.0.0.9	RIPv2	86	Response
16	64.859561	10.0.0.2	224.0.0.9	RIPv2	126	Response
20	79.881943	10.0.0.1	224.0.0.9	RIPv2	86	Response
27	106.144632	10.0.0.1	224.0.0.9	RIPv2	86	Response

> Frame 12: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits) on interface -, id 0
> Ethernet II, Src: ca:03:04:f4:00:70 (ca:03:04:f4:00:70), Dst: ca:04:1a:18:00:54 (ca:04:1a:18:00:54)
> Internet Protocol Version 4, Src: 10.0.0.2, Dst: 10.0.0.1
> User Datagram Protocol, Src Port: 520, Dst Port: 520
Routing Information Protocol
Command: Response (2)
Version: RIPv2 (2)
IP Address: 20.0.0.0, Metric: 1
IP Address: 30.0.0.0, Metric: 2
IP Address: 40.0.0.0, Metric: 3
IP Address: 41.0.0.0, Metric: 3

En cuanto a la otra interfaz (g3/0) al mensaje de Request sí que tiene una respuesta asociada como se puede observar, que el mensaje de Response inmediatamente siguiente al de Request es un mensaje unicast dirigido a R1 en el que se envía desde R2 su tabla de rutas a enviar por esta interfaz que en este caso es 20.0.0.0 con métrica 1 (entrega directa desde R2), 30.0.0.0 con métricas 2 (entrega directa desde R3) y 40.0.0.0, 41.0.0.0 con métrica 3 ya que son de entrega directa pero desde R4 que es el router siguiente a R3

- Comprueba la tabla de encaminamiento del encaminador recién arrancado, así como las tablas de encaminamiento del resto de los encaminadores para ver cómo se van propagando las rutas. Presta especial atención a la métrica de las rutas aprendidas y explica su valor. Explica el proceso de aprendizaje de rutas apoyándote en las capturas realizadas y en los mensajes de depuración.

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    50.0.0.0/8 is directly connected, GigabitEthernet4/0
R    20.0.0.0/8 [120/1] via 10.0.0.2, 00:00:43, GigabitEthernet3/0
R    40.0.0.0/8 [120/3] via 10.0.0.2, 00:00:43, GigabitEthernet3/0
C    10.0.0.0/8 is directly connected, GigabitEthernet3/0
R    41.0.0.0/8 [120/3] via 10.0.0.2, 00:00:43, GigabitEthernet3/0
C    11.0.0.0/8 is directly connected, FastEthernet0/0
R    30.0.0.0/8 [120/2] via 10.0.0.2, 00:00:43, GigabitEthernet3/0
R1#
```

EL FORMATO DE LAS CAPTURAS CAMBIA PORQUE PUDIMOS REALIZAR ESTAS EN UNA MÁQUINA NATIVA SIN VIRTUALIZAR UN DISPOSITIVO A TRAVÉS DE NOGAL.USAL.ES

Al realizar el comando *show ip route* en R1, se observa la tabla de rutas del router, en ella podemos observar como a las dos que tenía anteriormente (identificadas por la C) se han añadido cuatro más, 40.0.0.0/8, 41.0.0.0/8, 30.0.0.0/8 y 20.0.0.0/8 todas

por la interfaz g3/0 que corresponde con la dirección ip 10.0.0.2, el penúltimo parámetro es un contador que llegará como máximo a 180s que en ese momento se eliminará la entrada de la tabla, pero que en el momento de la captura llevaban en la caché poco más de cuarenta segundos. Y el último parámetro a comentar es valor entre corchetes, de este importa el valor tras la "/" ya que identifica el valor de la métrica que tiene esa entrada, en este caso podemos observar como 40.0.0.0 y 41.0.0.0 tiene una métrica 3 ya que se necesitan tres saltos para alcanzarlas ya que son subredes de entrega directa de R4, 30.0.0.0 tiene métrica 2 porque es de entrega directa pero de R3 y por último 20.0.0.0 con métrica 1 ya que es la subred de entrega directa de R2

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    50.0.0.0/8 [120/1] via 10.0.0.1, 00:00:06, GigabitEthernet4/0
C    20.0.0.0/8 is directly connected, GigabitEthernet3/0
R    40.0.0.0/8 [120/2] via 20.0.0.2, 00:00:12, GigabitEthernet3/0
C    10.0.0.0/8 is directly connected, GigabitEthernet4/0
R    41.0.0.0/8 [120/2] via 20.0.0.2, 00:00:12, GigabitEthernet3/0
R    11.0.0.0/8 [120/1] via 10.0.0.1, 00:00:06, GigabitEthernet4/0
R    30.0.0.0/8 [120/1] via 20.0.0.2, 00:00:12, GigabitEthernet3/0
R2#
```

Si se realiza el mismo procedimiento pero en R2, se observa que éste ha aprendido la ruta 11.0.0.0/8, 50.0.0.0/8 por la interfaz g4/0 a la que le corresponde la IP 10.0.0.1, esta entrada lleva almacenada 6s en el momento de la captura, y la métrica de la misma es 1 ya que las subredes ambas subredes son de entrega directa a través de R1

```
R3#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    50.0.0.0/8 [120/2] via 20.0.0.1, 00:00:06, GigabitEthernet4/0
C    20.0.0.0/8 is directly connected, GigabitEthernet4/0
R    40.0.0.0/8 [120/1] via 30.0.0.2, 00:00:18, GigabitEthernet3/0
R    10.0.0.0/8 [120/1] via 20.0.0.1, 00:00:06, GigabitEthernet4/0
R    41.0.0.0/8 [120/1] via 30.0.0.2, 00:00:18, GigabitEthernet3/0
R    11.0.0.0/8 [120/2] via 20.0.0.1, 00:00:06, GigabitEthernet4/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R3#
```

Si se realiza el mismo procedimiento pero en R3, se observa que éste ha aprendido la ruta 11.0.0.0/8, 50.0.0.0/8 por la interfaz g4/0 a la que le corresponde la IP 20.0.0.1, esta entrada lleva almacenada 6s en el momento de la captura, y la métrica

de la misma es 2 ya que la subred ambas subredes son de entrega directa a través de R1

```
R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    50.0.0.0/8 [120/3] via 30.0.0.1, 00:00:08, GigabitEthernet3/0
R    20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:08, GigabitEthernet3/0
C    40.0.0.0/8 is directly connected, GigabitEthernet4/0
R    10.0.0.0/8 [120/2] via 30.0.0.1, 00:00:08, GigabitEthernet3/0
C    41.0.0.0/8 is directly connected, FastEthernet0/0
R    11.0.0.0/8 [120/3] via 30.0.0.1, 00:00:08, GigabitEthernet3/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R4#
```

Si se realiza el mismo procedimiento pero en R4, se observa que éste ha aprendido la ruta 11.0.0.0/8, 50.0.0.0/8 por la interfaz g3/0 a la que le corresponde la IP 30.0.0.1, está entrada lleva almacenada 8s en el momento de la captura, y la métrica de la misma es 3 ya que la subred ambas subredes son de entrega directa a través de R1

Analizando los mensajes de depuración podemos observar cómo se envían y reciben los mensajes del protocolo:

Primeramente R1 envía su tabla de rutas por sus interfaces, en este caso por la interfaz g4/0, envía la tabla de rutas al completo en ella observamos como las entradas con métrica 1 son las que tiene R1 en entrega directa, la de métrica 2 es la entrada que tiene en entrega directa por R2, la de métrica 3 es la de R3 y las que quedan con métrica 4 son las que tiene R4 asociadas directamente a sus interfaces

```
RIP protocol debugging is on
R1#
*Mar 27 11:18:34.871: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet4/0 (50.0.0.2)
*Mar 27 11:18:34.871: RIP: build update entries
*Mar 27 11:18:34.871:   10.0.0.0/8 via 0.0.0.0, metric 1, tag 0
*Mar 27 11:18:34.871:   11.0.0.0/8 via 0.0.0.0, metric 1, tag 0
*Mar 27 11:18:34.875:   20.0.0.0/8 via 0.0.0.0, metric 2, tag 0
*Mar 27 11:18:34.875:   30.0.0.0/8 via 0.0.0.0, metric 3, tag 0
*Mar 27 11:18:34.875:   40.0.0.0/8 via 0.0.0.0, metric 4, tag 0
*Mar 27 11:18:34.879:   41.0.0.0/8 via 0.0.0.0, metric 4, tag 0
R1#
```

En f0/0 envía la tabla de rutas al completo quitando la entrada propia de la subred porque no tendría sentido, en ella observamos como las entradas con métrica 1 son las que tiene R1 en entrega directa, la de métrica 2 es la entrada que tiene en entrega directa por R2, la de métrica 3 es la de R3 y las que quedan con métrica 4 son las que tiene R4 asociadas directamente a sus interfaces

```
R1#
*Mar 27 11:18:36.367: RIP: sending v2 update to 224.0.0.9 via FastEthernet0/0 (11.0.0.1)
*Mar 27 11:18:36.367: RIP: build update entries
*Mar 27 11:18:36.367:   10.0.0.0/8 via 0.0.0.0, metric 1, tag 0
*Mar 27 11:18:36.371:   20.0.0.0/8 via 0.0.0.0, metric 2, tag 0
*Mar 27 11:18:36.371:   30.0.0.0/8 via 0.0.0.0, metric 3, tag 0
*Mar 27 11:18:36.371:   40.0.0.0/8 via 0.0.0.0, metric 4, tag 0
*Mar 27 11:18:36.371:   41.0.0.0/8 via 0.0.0.0, metric 4, tag 0
*Mar 27 11:18:36.375:   50.0.0.0/8 via 0.0.0.0, metric 1, tag 0
R1#
```

```
R1#
*Mar 27 11:18:37.519: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet3/0 (10.0.0.1)
*Mar 27 11:18:37.519: RIP: build update entries
*Mar 27 11:18:37.519:   11.0.0.0/8 via 0.0.0.0, metric 1, tag 0
*Mar 27 11:18:37.523:   50.0.0.0/8 via 0.0.0.0, metric 1, tag 0
R1#
```

Por último en g3/0 al ser esta interfaz por la que ha recibido otras rutas solo va a enviar las rutas que no haya aprendido por esta interfaz porque se utiliza el protocolo “Split Horizon” de RIP

```
R1#
*Mar 27 11:18:43.271: RIP: received v2 update from 10.0.0.2 on GigabitEthernet3/0
*Mar 27 11:18:43.271:   20.0.0.0/8 via 0.0.0.0 in 1 hops
*Mar 27 11:18:43.275:   30.0.0.0/8 via 0.0.0.0 in 2 hops
*Mar 27 11:18:43.275:   40.0.0.0/8 via 0.0.0.0 in 3 hops
*Mar 27 11:18:43.275:   41.0.0.0/8 via 0.0.0.0 in 3 hops
R1#no debug ip rip
```

Y en el momento de recibir, R1 recibe las rutas por la interfaz g3/0 y en ella R2 le envía su tabla de rutas que ha actualizado por medio de los otros routers (R3,R4) con sus tablas. En la imagen superior se observa como con salto 1 está la subred de 20.0.0.0 (R2), con salto 2 30.0.0.0 (R3) y 40.0.0.0, 41.0.0.0 a 3 saltos (R4)

- d) ¿La implementación de RIP que utilizan los routers de Cisco está empleando el mecanismo *Split Horizon* o el mecanismo *Split Horizon + Poison Reverse*? ¿Cómo lo sabes?

Los routers de Cisco que se simulan en el programa GNS3 utilizan el mecanismo del protocolo RIP “Split Horizon”, esto lo sabemos porque en este protocolo cuando envía sus mensajes de Response, no envía las entradas de las rutas que ha aprendido por ella, esto mismo es lo que hemos ido diciendo en la explicación de los puntos anteriores.

Además si se utilizase “Split Horizon + Poison Reverse” si que se enviarían esas entradas pero estarían marcadas con métrica 16 y como podemos observar en las imágenes esto no pasa.

2. Tras haber arrancado RIP en los encaminadores R4, R3, R2 y R1, PC2 y PC1 deberían tener conectividad IP. Compruébalo con las órdenes *ping* y *trace* (incluye aquí su salida).

```

Welcome to Virtual PC Simulator, version 0.6.2
Dedicated to Daling.
Build time: Apr 10 2019 02:42:20
Copyright (c) 2007-2014, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file

Checking for duplicate address...
PC1 : 11.0.0.10 255.0.0.0 gateway 11.0.0.1

PC1> ping 41.0.0.10
41.0.0.10 icmp_seq=1 timeout
84 bytes from 41.0.0.10 icmp_seq=2 ttl=60 time=74.432 ms
84 bytes from 41.0.0.10 icmp_seq=3 ttl=60 time=79.328 ms
84 bytes from 41.0.0.10 icmp_seq=4 ttl=60 time=74.308 ms
84 bytes from 41.0.0.10 icmp_seq=5 ttl=60 time=73.287 ms

PC1> trace 41.0.0.10
trace to 41.0.0.10, 8 hops max, press Ctrl+C to stop
 1  11.0.0.1   8.191 ms  9.184 ms  9.237 ms
 2  10.0.0.2   29.212 ms  29.219 ms  29.289 ms
 3  20.0.0.2   49.224 ms  49.190 ms  49.239 ms
 4  30.0.0.2   68.228 ms  69.243 ms  69.319 ms
 5  *41.0.0.10  79.226 ms (ICMP type:3, code:3, Destination port unreachable)

PC1>

```

solarwinds | Solar-PuTTY *free tool*

En la captura superior se muestra que efectivamente la conectividad entre PC1 → PC2 es correcta, en el momento de realizar el *ping*, se pierde un paquete pero los cuatro restantes se envían y responden correctamente. Con *trace* que muestra los saltos que tiene que hacer el paquete hasta llegar a la IP pasada como parámetro, en la ejecución del comando se observa efectivamente las IPs de los routers por los que pasa y el orden, hasta el destino, que responde con un ICMP de código 3, porque el puerto en el que se realiza la petición con *trace* no hay nada escuchando, pero nos da igual porque solo queremos comprobar que existe el terminal asociado a la IP

```

Welcome to Virtual PC Simulator, version 0.6.2
Dedicated to Daling.
Build time: Apr 10 2019 02:42:20
Copyright (c) 2007-2014, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file

Checking for duplicate address...
PC1 : 41.0.0.10 255.0.0.0 gateway 41.0.0.1

PC2> ping 11.0.0.10
84 bytes from 11.0.0.10 icmp_seq=1 ttl=60 time=76.270 ms
84 bytes from 11.0.0.10 icmp_seq=2 ttl=60 time=71.237 ms
84 bytes from 11.0.0.10 icmp_seq=3 ttl=60 time=71.521 ms
84 bytes from 11.0.0.10 icmp_seq=4 ttl=60 time=80.241 ms
84 bytes from 11.0.0.10 icmp_seq=5 ttl=60 time=72.296 ms

PC2> trace 11.0.0.10
trace to 11.0.0.10, 8 hops max, press Ctrl+C to stop
 1  41.0.0.1   9.182 ms  9.226 ms  9.221 ms
 2  30.0.0.1   29.187 ms  29.185 ms  29.252 ms
 3  20.0.0.1   49.244 ms  48.246 ms  49.250 ms
 4  10.0.0.1   68.226 ms  69.264 ms  69.240 ms
 5  *11.0.0.10  69.402 ms (ICMP type:3, code:3, Destination port unreachable)

PC2>

```

solarwinds | Solar-PuTTY *free tool*

Y en el caso de PC2 exactamente lo mismo, pero en este caso con *ping* no pierde ningún paquete.

Deja lanzado el *ping* de PC2 a PC1 (*ping 11.0.0.10 -t*), y borra la tabla de rutas de R4 (*clear ip route **). ¿Se ha producido pérdida de paquetes? ¿Por qué? Comprueba lo que ha sucedido con las capturas de tráfico necesarias.

Para realizar este apartado utilizamos la orden *ping* con el modificador “-t” que lo que provoca es que se estén enviando secuencias de mensajes ICMP Request y Reply hasta el usuario lo detenga o se llegue a 256 (supuestamente viendo la captura de Wireshark).

Mientras que con *clear ip route ** se eliminan todas las entradas de la tabla de rutas

```

PC2> ping 11.0.0.10 -t
11.0.0.10 icmp_seq=1 timeout
11.0.0.10 icmp_seq=2 timeout
84 bytes from 11.0.0.10 icmp_seq=3 ttl=60 time=46.181 ms
84 bytes from 11.0.0.10 icmp_seq=4 ttl=60 time=49.212 ms
84 bytes from 11.0.0.10 icmp_seq=5 ttl=60 time=50.362 ms
84 bytes from 11.0.0.10 icmp_seq=6 ttl=60 time=49.881 ms
84 bytes from 11.0.0.10 icmp_seq=7 ttl=60 time=50.258 ms
84 bytes from 11.0.0.10 icmp_seq=8 ttl=60 time=44.190 ms
84 bytes from 11.0.0.10 icmp_seq=9 ttl=60 time=95.459 ms
84 bytes from 11.0.0.10 icmp_seq=10 ttl=60 time=61.257 ms
84 bytes from 11.0.0.10 icmp_seq=11 ttl=60 time=63.601 ms
84 bytes from 11.0.0.10 icmp_seq=12 ttl=60 time=71.855 ms
84 bytes from 11.0.0.10 icmp_seq=13 ttl=60 time=73.262 ms
84 bytes from 11.0.0.10 icmp_seq=14 ttl=60 time=47.287 ms
84 bytes from 11.0.0.10 icmp_seq=15 ttl=60 time=50.568 ms
84 bytes from 11.0.0.10 icmp_seq=16 ttl=60 time=68.798 ms
84 bytes from 11.0.0.10 icmp_seq=17 ttl=60 time=72.762 ms
84 bytes from 11.0.0.10 icmp_seq=18 ttl=60 time=79.441 ms
84 bytes from 11.0.0.10 icmp_seq=19 ttl=60 time=80.324 ms
84 bytes from 11.0.0.10 icmp_seq=20 ttl=60 time=72.206 ms
84 bytes from 11.0.0.10 icmp_seq=21 ttl=60 time=74.290 ms
84 bytes from 11.0.0.10 icmp_seq=22 ttl=60 time=71.291 ms
84 bytes from 11.0.0.10 icmp_seq=23 ttl=60 time=73.308 ms
84 bytes from 11.0.0.10 icmp_seq=24 ttl=60 time=78.850 ms
84 bytes from 11.0.0.10 icmp_seq=25 ttl=60 time=78.297 ms
84 bytes from 11.0.0.10 icmp_seq=26 ttl=60 time=75.276 ms
84 bytes from 11.0.0.10 icmp_seq=27 ttl=60 time=81.244 ms
84 bytes from 11.0.0.10 icmp_seq=28 ttl=60 time=80.936 ms
84 bytes from 11.0.0.10 icmp_seq=29 ttl=60 time=78.992 ms
84 bytes from 11.0.0.10 icmp_seq=30 ttl=60 time=73.258 ms

[...]
R4#clear ip route *
R4#clear ip route *
R4#

```

Capturas de la ejecución de los dos comandos, hemos realizado dos limpiezas de las tablas para comprobar el funcionamiento mejor.

Tx/Rx	Source	Destination	Type	Length	Time
50	56.363588	30.0.0.2	224.0.0.9	RIPv2	66 Request
51	56.363588	30.0.0.2	224.0.0.9	RIPv2	66 Request
52	56.363588	30.0.0.2	224.0.0.9	RIPv2	66 Request
53	56.374144	30.0.0.1	30.0.0.2	RIPv2	126 Response
54	56.374144	30.0.0.1	30.0.0.2	RIPv2	126 Response
55	56.374144	30.0.0.1	30.0.0.2	RIPv2	126 Response
56	57.071687	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0x7176, seq=19/4864, ttl=63 (reply in 57)
57	57.132277	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0x7176, seq=19/4864, ttl=61 (request in 56)
58	57.114116	30.0.0.2	224.0.0.9	RIPv2	86 Response
59	58.144575	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0x7276, seq=20/5120, ttl=63 (reply in 60)
60	58.205384	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0x7276, seq=20/5120, ttl=61 (request in 59)
61	59.220922	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0x7376, seq=21/5376, ttl=63 (reply in 62)
62	59.282028	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0x7376, seq=21/5376, ttl=61 (request in 61)
63	60.294911	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0x7476, seq=22/5632, ttl=63 (reply in 64)
64	60.354999	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0x7476, seq=22/5632, ttl=61 (request in 63)
65	61.369997	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0x7576, seq=23/5888, ttl=63 (reply in 66)
66	61.430821	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0x7576, seq=23/5888, ttl=61 (request in 65)
67	62.448477	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0x7676, seq=24/6144, ttl=63 (reply in 68)

> Frame 54: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits)
> Ethernet II, Src: ca:02:03:d8:00:54 (ca:02:03:d8:00:54), Dst: ca:01:2e:d8:00:54 (ca:01:2e:d8:00:54)
> Internet Protocol Version 4, Src: 30.0.0.1, Dst: 30.0.0.2
> User Datagram Protocol, Src Port: 520, Dst Port: 520
> Routing Information Protocol
 Command: Response (2)
 Version: RIPv2 (2)
 > IP Address: 10.0.0.0, Metric: 2
 > IP Address: 11.0.0.0, Metric: 3
 > IP Address: 20.0.0.0, Metric: 1
 > IP Address: 50.0.0.0, Metric: 3

Como se observa tanto en la captura del *ping* no se pierde ningún mensaje porque al limpiar las tablas de rutas, este mismo router inmediatamente manda tres mensajes de Request a la dirección multicast del protocolo RIP (224.0.0.9), y estos mensajes son contestados automáticamente por el router de siguiente salto que en este caso es R3 (30.0.0.1) con la tabla de rutas (IPs y métricas de cada entrada). Gracias a la rapidez de esto se completa antes de que el siguiente mensaje de Echo Request llegue al router.

109	82.021071	30.0.0.2	224.0.0.9	RIPv2	66 Request
118	82.021071	30.0.0.2	224.0.0.9	RIPv2	66 Request
111	82.022071	30.0.0.2	224.0.0.9	RIPv2	66 Request
112	82.031482	30.0.0.1	30.0.0.2	RIPv2	126 Response
113	82.031482	30.0.0.1	30.0.0.2	RIPv2	126 Response
114	82.031482	30.0.0.1	30.0.0.2	RIPv2	126 Response
115	82.711853	ca:01:2e:d8:00:54	ca:01:2e:d8:00:54	LOOP	60 Reply
116	82.924971	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0xb76, seq=43/11008, ttl=63 (reply in 117)
117	82.973978	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0xb76, seq=43/11008, ttl=61 (request in 116)
118	83.671869	30.0.0.2	224.0.0.9	RIPv2	86 Response
119	83.986920	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0x8c76, seq=44/11264, ttl=63 (reply in 120)
120	84.035941	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0x8c76, seq=44/11264, ttl=61 (request in 119)
121	85.049024	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0xd76, seq=45/11520, ttl=63 (reply in 122)
122	85.098932	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0xd76, seq=45/11520, ttl=61 (request in 121)
123	86.108843	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0xe6, seq=46/11776, ttl=63 (reply in 124)
124	86.157855	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0xe6, seq=46/11776, ttl=61 (request in 123)
125	86.614240	30.0.0.1	224.0.0.9	RIPv2	126 Response
126	87.169215	41.0.0.10	11.0.0.10	ICMP	98 Echo (ping) request id=0x8f6, seq=47/12032, ttl=63 (reply in 127)
127	87.218401	11.0.0.10	41.0.0.10	ICMP	98 Echo (ping) reply id=0x8f6, seq=47/12032, ttl=61 (request in 126)

Y en la captura de la siguiente limpieza completamente lo mismo sin ningún problema, realiza el Request, R3 le envía su tabla en el Response y se sigue completando el ping.

1.5. Actualizaciones de las tablas de rutas

- A continuación, realiza los cambios necesarios para que la ruta seguida por los datagramas IP que envía PC2 a PC1 sea PC2 => R4 => R5 => R1 => PC1, y para que los que envía PC1 a PC2 sigan la ruta PC1 => R1 => R5 => R4 => PC2. Para realizar este apartado no podrás añadir o eliminar manualmente rutas en las tablas de encaminamiento. Describe las acciones realizadas.

```

R5#config t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#router rip
R5(config-router)#network 50.0.0.0
R5(config-router)#network 40.0.0.0
R5(config-router)#no auto-summary
R5(config-router)#exit
R5(config)#exit
R5#
*Mar 29 18:08:28.487: %SYS-5-CONFIG_I: Configured from console by console
R5#

```

Para ello configuramos R5 como lo hemos realizado previamente con otros routers:

```

config t
router rip
version 2
network 40.0.0.0
network 50.0.0.0
no auto-summary
exit
exit

```

```

R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    50.0.0.0/8 is directly connected, GigabitEthernet3/0
R    20.0.0.0/8 [120/2] via 50.0.0.2, 00:00:09, GigabitEthernet3/0
                  [120/2] via 40.0.0.1, 00:00:29, GigabitEthernet4/0
C    40.0.0.0/8 is directly connected, GigabitEthernet4/0
R    10.0.0.0/8 [120/1] via 50.0.0.2, 00:00:09, GigabitEthernet3/0
R    41.0.0.0/8 [120/1] via 40.0.0.1, 00:00:29, GigabitEthernet4/0
R    11.0.0.0/8 [120/1] via 50.0.0.2, 00:00:09, GigabitEthernet3/0
R    30.0.0.0/8 [120/1] via 40.0.0.1, 00:00:29, GigabitEthernet4/0
R5#

```

En la imagen vemos como R5 se ha configurado correctamente.

Mirando la tabla de encaminamiento de R4, observa y apunta el número de segundos que aproximadamente tarda en aprenderse la nueva ruta. ¿Por qué?

El número de segundos que tarda se aproxima a 180, esto se debe a que es el tiempo en el que R4 no recibe los seis períodos de actualización periódica sobre la ruta por lo que finalmente la elimina y deja de anunciar. Posteriormente, R4 introducirá en su tabla la nueva ruta gracias a la información transmitida en la actualización periódica.

Comprueba que se está utilizando dicha ruta a través de la orden *trace*.

```

PC1> trace 41.0.0.10
trace to 41.0.0.10, 8 hops max, press Ctrl+C to stop
 1  11.0.0.1   6.290 ms  10.089 ms  9.537 ms
 2  50.0.0.1   20.069 ms  19.911 ms  19.760 ms
 3  40.0.0.1   29.680 ms  30.130 ms  29.340 ms
 4    **41.0.0.10   35.637 ms (ICMP type:3, code:3, Destination port unreachable)
PC1>

```

Podemos observar como ahora el trace de PC1 a PC2 sigue el camino superior de routers: (11.0.0.0) → R1 → (50.0.0.0) → R5 → (40.0.0.0) → R4 → (41.0.0.0)

```

PC2> trace 11.0.0.10
trace to 11.0.0.10, 8 hops max, press Ctrl+C to stop
 1  41.0.0.1   7.907 ms  9.500 ms  9.974 ms
 2  40.0.0.2   19.459 ms  19.378 ms  19.254 ms
 3  50.0.0.2   29.139 ms  29.449 ms  29.776 ms
 4    *11.0.0.10   40.179 ms (ICMP type:3, code:3, Destination port unreachable)
PC2>

```

Y viceversa: (41.0.0.0) → R4 → (40.0.0.0) → R5 → (50.0.0.0) → R1 → (11.0.0.0)

Comprueba las rutas y sus métricas en las tablas de encaminamiento de cada encaminador. Inclúyelas aquí. Comenta la salida.

R1

Archivo Editar Ver Terminal Pestañas Ayuda

◀ i0961594@... × R1 × R2 × R3 × R4 × ▶

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    50.0.0.0/8 is directly connected, GigabitEthernet4/0
R    20.0.0.0/8 [120/1] via 10.0.0.2, 00:00:20, GigabitEthernet3/0
R    40.0.0.0/8 [120/1] via 50.0.0.1, 00:00:17, GigabitEthernet4/0
C    10.0.0.0/8 is directly connected, GigabitEthernet3/0
R    41.0.0.0/8 [120/2] via 50.0.0.1, 00:00:17, GigabitEthernet4/0
C    11.0.0.0/8 is directly connected, FastEthernet0/0
R    30.0.0.0/8 [120/2] via 50.0.0.1, 00:00:17, GigabitEthernet4/0
                           [120/2] via 10.0.0.2, 00:00:21, GigabitEthernet3/0
R1#
```

Como se puede observar R1 tiene las siguientes entradas registradas:

Entrega directa → a las redes 50.0.0.0, 10.0.0.0 y 11.0.0.0.

Métrica 1 → a 20.0.0.0 mediante 10.0.0.2 y a 40.0.0.0 mediante 50.0.0.1.

Métrica 2 → a 30.0.0.0 mediante 50.0.0.1 y 10.0.0.2 y a 41.0.0.0 mediante 50.0.0.1.

R2

Archivo Editar Ver Terminal Pestañas Ayuda

◀ i0961594@... × R1 × R2 × R3 × R4 × ▶

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    50.0.0.0/8 [120/1] via 10.0.0.1, 00:00:11, GigabitEthernet4/0
C    20.0.0.0/8 is directly connected, GigabitEthernet3/0
R    40.0.0.0/8 [120/2] via 20.0.0.2, 00:00:16, GigabitEthernet3/0
                           [120/2] via 10.0.0.1, 00:00:12, GigabitEthernet4/0
C    10.0.0.0/8 is directly connected, GigabitEthernet4/0
R    41.0.0.0/8 [120/2] via 20.0.0.2, 00:00:16, GigabitEthernet3/0
R    11.0.0.0/8 [120/1] via 10.0.0.1, 00:00:12, GigabitEthernet4/0
R    30.0.0.0/8 [120/1] via 20.0.0.2, 00:00:16, GigabitEthernet3/0
R2#
```

Como se puede observar R2 tiene las siguientes entradas registradas:

Entrega directa → a las redes 20.0.0.0 y 10.0.0.0.

Métrica 1 → a 50.0.0.0 y 11.0.0.0 mediante 10.0.0.1 y a 30.0.0.0 mediante 20.0.0.2.

Métrica 2 → a 40.0.0.0 mediante 20.0.0.2 y 10.0.0.1 y a 41.0.0.0 mediante 20.0.0.2.

R3

Archivo Editar Ver Terminal Pestañas Ayuda

◀ i0961594@... × R1 × R2 × R3 × R4 × ▶

```
R3#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    50.0.0.0/8 [120/2] via 30.0.0.2, 00:00:25, GigabitEthernet3/0
      [120/2] via 20.0.0.1, 00:00:08, GigabitEthernet4/0
C    20.0.0.0/8 is directly connected, GigabitEthernet4/0
R    40.0.0.0/8 [120/1] via 30.0.0.2, 00:00:25, GigabitEthernet3/0
R    10.0.0.0/8 [120/1] via 20.0.0.1, 00:00:08, GigabitEthernet4/0
R    41.0.0.0/8 [120/1] via 30.0.0.2, 00:00:25, GigabitEthernet3/0
R    11.0.0.0/8 [120/2] via 20.0.0.1, 00:00:08, GigabitEthernet4/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R3#
```

Como se puede observar R3 tiene las siguientes entradas registradas:

Entrega directa → a las redes 20.0.0.0 y 30.0.0.0.

Métrica 1 → a 40.0.0.0 y 41.0.0.0 mediante 30.0.0.2 y a 10.0.0.0 mediante 20.0.0.1.

Métrica 2 → a 50.0.0.0 mediante 30.0.0.2 y a 11.0.0.0 mediante 20.0.0.1.

R4

Archivo Editar Ver Terminal Pestañas Ayuda

◀ i0961594@... × R1 × R2 × R3 × R4 × ▶

```
R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    50.0.0.0/8 [120/1] via 40.0.0.2, 00:00:10, GigabitEthernet4/0
R    20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:12, GigabitEthernet3/0
C    40.0.0.0/8 is directly connected, GigabitEthernet4/0
R    10.0.0.0/8 [120/2] via 40.0.0.2, 00:00:10, GigabitEthernet4/0
      [120/2] via 30.0.0.1, 00:00:12, GigabitEthernet3/0
C    41.0.0.0/8 is directly connected, FastEthernet0/0
R    11.0.0.0/8 [120/2] via 40.0.0.2, 00:00:10, GigabitEthernet4/0
C    30.0.0.0/8 is directly connected, GigabitEthernet3/0
R4#
```

Como se puede observar R4 tiene las siguientes entradas registradas:

Entrega directa → a las redes 30.0.0.0, 40.0.0.0 y 41.0.0.0.

Métrica 1 → a 20.0.0.0 mediante 30.0.0.1 y a 50.0.0.0 mediante 40.0.0.2.

Métrica 2 → a 10.0.0.0 mediante 40.0.0.2 y 30.0.0.1 y a 11.0.0.0 mediante 40.0.0.2.

```

R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    50.0.0.0/8 is directly connected, GigabitEthernet3/0
R    20.0.0.0/8 [120/2] via 50.0.0.2, 00:00:09, GigabitEthernet3/0
      [120/2] via 40.0.0.1, 00:00:05, GigabitEthernet4/0
C    40.0.0.0/8 is directly connected, GigabitEthernet4/0
R    10.0.0.0/8 [120/1] via 50.0.0.2, 00:00:09, GigabitEthernet3/0
R    41.0.0.0/8 [120/1] via 40.0.0.1, 00:00:05, GigabitEthernet4/0
R    11.0.0.0/8 [120/1] via 50.0.0.2, 00:00:09, GigabitEthernet3/0
R    30.0.0.0/8 [120/1] via 40.0.0.1, 00:00:05, GigabitEthernet4/0
R5#

```

Como se puede observar R5 tiene las siguientes entradas registradas:

Entrega directa → a las redes 40.0.0.0 y 50.0.0.0.

Métrica 1 → a 10.0.0.0 y 11.0.0.0 mediante 50.0.0.2 y a 41.0.0.0 y 30.0.0.0 mediante 40.0.0.1.

Métrica 2 → a 20.0.0.0 mediante 50.0.0.2 y 40.0.0.1.

2. ¿Por qué ruta deberían ir los datagramas IP que envíe PC2 a la dirección 10.0.0.2? Justifica la respuesta. Comprueba tu respuesta utilizando `trace`. Con la misma red del escenario, y con los 5 encaminadores con RIP activado, ¿podrían haber seguido otra ruta los datagramas IP PC2 a la dirección 10.0.0.2? ¿Cómo actúan los routers CISCO cuando reciben una ruta con igual métrica que la que ya tienen en su tabla de rutas?

```

PC2
Archivo Editar Ver Terminal Pestañas Ayuda
i0961594@sundia05: ~
PC2> trace 10.0.0.2
trace to 10.0.0.2, 8 hops max, press Ctrl+C to stop
 1  41.0.0.1   3.106 ms  10.020 ms  9.554 ms
 2  30.0.0.1   20.038 ms  20.071 ms  20.044 ms
 3  *20.0.0.1   30.139 ms (ICMP type:3, code:3, Destination port unreachable)
PC2>

```

La ruta que deberían seguir los datagramas IP de PC2 a 10.0.0.2 como se ve en la imagen superior de R4 pueden seguir dos caminos, ir por 40.0.0.2 o 30.0.0.1 ya que tienen el mismo 'coste'. Las rutas serían las siguientes:

(41.0.0.0) → R4 → (40.0.0.0) → R5 → (50.0.0.0) → R1

(41.0.0.0) → R4 → (30.0.0.0) → R3 → (20.0.0.0) → R2

Como se ve en la imagen, en este caso se ha seguido la ruta: **(41.0.0.0) → R4 → (30.0.0.0) → R3 → 20.0.0.0 → R2**

Los routers CISCO seleccionan la ruta con menor métrica pero si existen múltiples rutas al mismo destino con métricas iguales la selecciona según el equilibrio de carga de estos trayectos.

1.6. Eliminación de rutas

El objetivo de este apartado es observar lo que ocurre cuando se interrumpe *RIP* en R5. Se estudiará, en particular, el comportamiento de los encaminadores R4 y R1.

1. Asegúrate de que los 5 encaminadores tienen arrancado *RIP*. Comprueba la ruta que están siguiendo los mensajes intercambiados entre PC2 y PC1 con *trace*. Incluye aquí la salida.

```
i0961594@sundia05: ~          PC1          - □ ×
Archivo Editar Ver Terminal Pestañas Ayuda
PC1                                     PC2
i0961594@sundia05: ~      x      PC1      x      PC2      x
PC1> trace 41.0.0.10
trace to 41.0.0.10, 8 hops max, press Ctrl+C to stop
 1  11.0.0.1    2.554 ms  9.456 ms  9.930 ms
 2  50.0.0.1    19.520 ms  19.660 ms  19.718 ms
 3  40.0.0.1    29.866 ms  29.233 ms  29.909 ms
 4  *41.0.0.10   39.630 ms (ICMP type:3, code:3, Destination port unreachable)

PC1>
```

```
i0961594@sundia05: ~          PC2          - □ ×
Archivo Editar Ver Terminal Pestañas Ayuda
PC2                                     PC1          x      PC2
i0961594@sundia05: ~      x      PC1      x      PC2      x
PC2> trace 11.0.0.10
trace to 11.0.0.10, 8 hops max, press Ctrl+C to stop
 1  41.0.0.1    0.567 ms  9.553 ms  10.071 ms
 2  40.0.0.2    19.713 ms  19.820 ms  19.703 ms
 3  50.0.0.2    29.500 ms  30.062 ms  29.355 ms
 4  *11.0.0.10   39.887 ms (ICMP type:3, code:3, Destination port unreachable)

PC2>
```

Como se puede observar mediante las ordenes *trace* ambos equipos siguen direccionándose por los routers superiores, es decir, R4, R5 y R1.

2. Deja en ejecución en PC2 un *ping* hacia PC1 (*ping 11.0.0.10 -t*).

```

PC2> ping 11.0.0.10 -t
11.0.0.10 icmp_seq=1 timeout
11.0.0.10 icmp_seq=2 timeout
84 bytes from 11.0.0.10 icmp_seq=3 ttl=61 time=32.128 ms
84 bytes from 11.0.0.10 icmp_seq=4 ttl=61 time=35.531 ms
84 bytes from 11.0.0.10 icmp_seq=5 ttl=61 time=35.929 ms
84 bytes from 11.0.0.10 icmp_seq=6 ttl=61 time=36.082 ms
84 bytes from 11.0.0.10 icmp_seq=7 ttl=61 time=35.636 ms
84 bytes from 11.0.0.10 icmp_seq=8 ttl=61 time=35.997 ms
84 bytes from 11.0.0.10 icmp_seq=9 ttl=61 time=36.062 ms
84 bytes from 11.0.0.10 icmp_seq=10 ttl=61 time=35.550 ms
84 bytes from 11.0.0.10 icmp_seq=11 ttl=61 time=36.029 ms
84 bytes from 11.0.0.10 icmp_seq=12 ttl=61 time=45.830 ms
84 bytes from 11.0.0.10 icmp_seq=13 ttl=61 time=46.189 ms
84 bytes from 11.0.0.10 icmp_seq=14 ttl=61 time=35.640 ms
84 bytes from 11.0.0.10 icmp_seq=15 ttl=61 time=36.005 ms
84 bytes from 11.0.0.10 icmp_seq=16 ttl=61 time=36.279 ms
84 bytes from 11.0.0.10 icmp_seq=17 ttl=61 time=35.680 ms
84 bytes from 11.0.0.10 icmp_seq=18 ttl=61 time=35.652 ms
84 bytes from 11.0.0.10 icmp_seq=19 ttl=61 time=45.965 ms
84 bytes from 11.0.0.10 icmp_seq=20 ttl=61 time=35.297 ms
84 bytes from 11.0.0.10 icmp_seq=21 ttl=61 time=45.739 ms
84 bytes from 11.0.0.10 icmp_seq=22 ttl=61 time=45.481 ms
84 bytes from 11.0.0.10 icmp_seq=23 ttl=61 time=45.435 ms
84 bytes from 11.0.0.10 icmp_seq=24 ttl=61 time=45.888 ms
84 bytes from 11.0.0.10 icmp_seq=25 ttl=61 time=46.128 ms
84 bytes from 11.0.0.10 icmp_seq=26 ttl=61 time=35.954 ms
84 bytes from 11.0.0.10 icmp_seq=27 ttl=61 time=36.779 ms
84 bytes from 11.0.0.10 icmp_seq=28 ttl=61 time=36.139 ms
84 bytes from 11.0.0.10 icmp_seq=29 ttl=61 time=35.474 ms
84 bytes from 11.0.0.10 icmp_seq=30 ttl=61 time=35.999 ms
84 bytes from 11.0.0.10 icmp_seq=31 ttl=61 time=35.109 ms
84 bytes from 11.0.0.10 icmp_seq=32 ttl=61 time=36.613 ms

```

3. Para ver los mensajes RIP que envían R4 y R1, arranca *wireshark* en sus interfaces g3/0 y activa los mensajes de depuración (debug ip rip).
4. A continuación, interrumpe la ejecución de *RIP* en el encaminador R5 utilizando las ordenes

```

config t
no router rip

```

Podrás observar con la orden show ip route que ahora R5 no conoce rutas aprendidas por RIP. Tampoco exporta rutas hacia otros encaminadores.

```

R5
Archivo Editar Ver Terminal Pestañas Ayuda
◀ PC1 × PC2 × R4 × R1 × R5 × ▶
R5#show ip route
*Mar 29 18:47:13.415: %SYS-5-CONFIG_I: Configured from console by console
R5#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#no router rip

```

R5

Archivo Editar Ver Terminal Pestañas Ayuda

PC1 R4 R1 R5

```
R5#show ip
*Mar 29 18:47:52.995: %SYS-5-CONFIG_I: Configured from console by console
R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C      50.0.0.0/8 is directly connected, GigabitEthernet3/0
C      40.0.0.0/8 is directly connected, GigabitEthernet4/0
R5#
```

5. Observarás que el *ping* de PC2 a PC1 deja de funcionar durante un buen rato. Observa durante este periodo en el que no está funcionando RIP en R5, las entradas de las tablas de encaminamiento de R4 y de R1. Observa la evolución de la columna de tiempo de las distintas entradas. ¿Qué entradas no reinician la cuenta cada 30 segundos? ¿Por qué? Observa el valor de tiempo de esas entradas. Presta especial atención cuanto el valor de la columna de tiempo de algunas entradas de R4 y R1 se acerquen a 3 minutos ¿qué ocurrirá después?

R4

Archivo Editar Ver Terminal Pestañas Ayuda

```
R4#show ip route rip
R    50.0.0.0/8 [120/1] via 40.0.0.2, 00:01:02, GigabitEthernet4/0
R    20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:59, GigabitEthernet3/0
R    10.0.0.0/8 [120/2] via 40.0.0.2, 00:01:02, GigabitEthernet4/0
                  [120/2] via 30.0.0.1, 00:00:59, GigabitEthernet3/0
R    11.0.0.0/8 [120/2] via 40.0.0.2, 00:01:02, GigabitEthernet4/0
R4#show ip route rip
R    50.0.0.0/8 [120/1] via 40.0.0.2, 00:01:10, GigabitEthernet4/0
R    20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:05, GigabitEthernet3/0
R    10.0.0.0/8 [120/2] via 40.0.0.2, 00:01:10, GigabitEthernet4/0
                  [120/2] via 30.0.0.1, 00:00:05, GigabitEthernet3/0
R    11.0.0.0/8 [120/2] via 40.0.0.2, 00:01:10, GigabitEthernet4/0
```

En R4 las entradas que no se reinician cada minuto son aquellas cuyos destinos son 50.0.0.0, 10.0.0.0 (mediante 30.0.0.1) y 11.0.0.0 que corresponden con las rutas que conoce gracias a R5.

Sin embargo, si reinicia la opción de 10.0.0.0 mediante 30.0.0.1 y la de 20.0.0.0 que son las que conoce mediante R3.

R1

Archivo Editar Ver Terminal Pestañas Ayuda

```
R1#show ip route rip
R    20.0.0.0/8 [120/1] via 10.0.0.2, 00:00:21, GigabitEthernet3/0
R    40.0.0.0/8 [120/1] via 50.0.0.1, 00:01:21, GigabitEthernet4/0
R    41.0.0.0/8 [120/2] via 50.0.0.1, 00:01:21, GigabitEthernet4/0
R    30.0.0.0/8 [120/2] via 50.0.0.1, 00:01:21, GigabitEthernet4/0
                  [120/2] via 10.0.0.2, 00:00:21, GigabitEthernet3/0
R1#show ip route rip
R    20.0.0.0/8 [120/1] via 10.0.0.2, 00:00:09, GigabitEthernet3/0
R    40.0.0.0/8 [120/1] via 50.0.0.1, 00:02:05, GigabitEthernet4/0
R    41.0.0.0/8 [120/2] via 50.0.0.1, 00:02:05, GigabitEthernet4/0
R    30.0.0.0/8 [120/2] via 50.0.0.1, 00:02:05, GigabitEthernet4/0
                  [120/2] via 10.0.0.2, 00:00:09, GigabitEthernet3/0
```

En R1 las entradas que no se reinician cada minuto son aquellas cuyos destinos son 40.0.0.0, 41.0.0.0 y 30.0.0.0 (mediante 50.0.0.1) que corresponden con las rutas que conoce gracias a R5.

Sin embargo, si reinicia la opción de 30.0.0.0 mediante 10.0.0.2 y la de 20.0.0.0 que son las que conoce mediante R2.

```
R4#show ip route rip
R  50.0.0.0/8 [120/1] via 40.0.0.2, 00:03:02, GigabitEthernet4/0
R  20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:54, GigabitEthernet3/0
R  10.0.0.0/8 [120/2] via 40.0.0.2, 00:03:02, GigabitEthernet4/0
          [120/2] via 30.0.0.1, 00:00:54, GigabitEthernet3/0
R  11.0.0.0/8 [120/2] via 40.0.0.2, 00:03:02, GigabitEthernet4/0
R4#show ip route rip
R  50.0.0.0/8 is possibly down, routing via 40.0.0.2, GigabitEthernet4/0
R  20.0.0.0/8 [120/1] via 30.0.0.1, 00:01:03, GigabitEthernet3/0
R  10.0.0.0/8 [120/2] via 30.0.0.1, 00:01:03, GigabitEthernet3/0
R  11.0.0.0/8 is possibly down, routing via 40.0.0.2, GigabitEthernet4/0

R1#show ip route rip
R  20.0.0.0/8 [120/1] via 10.0.0.2, 00:00:09, GigabitEthernet3/0
R  40.0.0.0/8 [120/1] via 50.0.0.1, 00:02:05, GigabitEthernet4/0
R  41.0.0.0/8 [120/2] via 50.0.0.1, 00:02:05, GigabitEthernet4/0
R  30.0.0.0/8 [120/2] via 50.0.0.1, 00:02:05, GigabitEthernet4/0
          [120/2] via 10.0.0.2, 00:00:09, GigabitEthernet3/0
R1#show ip route rip
R  20.0.0.0/8 [120/1] via 10.0.0.2, 00:00:16, GigabitEthernet3/0
R  40.0.0.0/8 is possibly down, routing via 50.0.0.1, GigabitEthernet4/0
R  41.0.0.0/8 is possibly down, routing via 50.0.0.1, GigabitEthernet4/0
R  30.0.0.0/8 [120/2] via 10.0.0.2, 00:00:17, GigabitEthernet3/0
```

Se observa como cuando pasan los 3 minutos elimina de la tabla aquellas rutas para las que existe otra con el mismo destino y también avisa de que posiblemente estén caídas aquellas que dependen del direccionamiento de R5.

6. Poco después el *ping* entre PC2 y PC1 habrá vuelto a funcionar. ¿Por qué? Interrúmpelo y, mirando los valores del *icmp_seq* apunta el número de segundos¹ que aproximadamente ha estado sin funcionar el *ping* debido a que aún no se había olvidado la ruta a través de R5. Comprueba la ruta que están siguiendo los mensajes intercambiados entre PC2 y PC1 con *trace*. Incluye aquí la salida.

¹ El timeout de ping es de 2s (FUENTE: <https://www.cisco.com/c/en/us/support/docs/ip/routing-information-protocol-rip/13730-ext-ping-trace.html>)

```
PC2
Archivo Editar Ver Terminal Pestañas Ayuda
i0961594@sun... x PC1 x PC2 x R4 x R1 x R5 x
84 bytes from 11.0.0.10 icmp_seq=147 ttl=61 time=35.656 ms
84 bytes from 11.0.0.10 icmp_seq=148 ttl=61 time=66.118 ms
84 bytes from 11.0.0.10 icmp_seq=149 ttl=61 time=35.917 ms
84 bytes from 11.0.0.10 icmp_seq=150 ttl=61 time=36.136 ms
84 bytes from 11.0.0.10 icmp_seq=151 ttl=61 time=65.454 ms
84 bytes from 11.0.0.10 icmp_seq=152 ttl=61 time=66.464 ms
84 bytes from 11.0.0.10 icmp_seq=153 ttl=61 time=35.560 ms
84 bytes from 11.0.0.10 icmp_seq=154 ttl=61 time=35.847 ms
84 bytes from 11.0.0.10 icmp_seq=155 ttl=61 time=66.139 ms
84 bytes from 11.0.0.10 icmp_seq=156 ttl=61 time=65.715 ms
84 bytes from 11.0.0.10 icmp_seq=157 ttl=61 time=35.276 ms
84 bytes from 11.0.0.10 icmp_seq=158 ttl=61 time=65.861 ms
84 bytes from 11.0.0.10 icmp_seq=159 ttl=61 time=65.708 ms
84 bytes from 11.0.0.10 icmp_seq=160 ttl=61 time=66.826 ms
84 bytes from 11.0.0.10 icmp_seq=161 ttl=61 time=65.493 ms
84 bytes from 11.0.0.10 icmp_seq=162 ttl=61 time=65.406 ms
84 bytes from 11.0.0.10 icmp_seq=163 ttl=61 time=35.624 ms
84 bytes from 11.0.0.10 icmp_seq=164 ttl=61 time=36.018 ms
84 bytes from 11.0.0.10 icmp_seq=165 ttl=61 time=35.777 ms
84 bytes from 11.0.0.10 icmp_seq=166 ttl=61 time=36.049 ms
84 bytes from 11.0.0.10 icmp_seq=167 ttl=61 time=65.678 ms
11.0.0.10 icmp_seq=168 timeout
11.0.0.10 icmp_seq=169 timeout
11.0.0.10 icmp_seq=170 timeout
11.0.0.10 icmp_seq=171 timeout
11.0.0.10 icmp_seq=172 timeout
11.0.0.10 icmp_seq=173 timeout
11.0.0.10 icmp_seq=174 timeout
11.0.0.10 icmp_seq=175 timeout
11.0.0.10 icmp_seq=176 timeout
11.0.0.10 icmp_seq=177 timeout
11.0.0.10 icmp_seq=178 timeout
```

```
PC2
Archivo Editar Ver Terminal Pestañas Ayuda
i0961594@sun... x PC1 x PC2 x R4 x R1 x R5 x
11.0.0.10 icmp_seq=255 timeout
11.0.0.10 icmp_seq=256 timeout
11.0.0.10 icmp_seq=257 timeout
11.0.0.10 icmp_seq=258 timeout
11.0.0.10 icmp_seq=259 timeout
11.0.0.10 icmp_seq=260 timeout
11.0.0.10 icmp_seq=261 timeout
11.0.0.10 icmp_seq=262 timeout
11.0.0.10 icmp_seq=263 timeout
11.0.0.10 icmp_seq=264 timeout
11.0.0.10 icmp_seq=265 timeout
11.0.0.10 icmp_seq=266 timeout
11.0.0.10 icmp_seq=267 timeout
11.0.0.10 icmp_seq=268 timeout
11.0.0.10 icmp_seq=269 timeout
84 bytes from 11.0.0.10 icmp_seq=270 ttl=60 time=59.347 ms
84 bytes from 11.0.0.10 icmp_seq=271 ttl=60 time=45.929 ms
84 bytes from 11.0.0.10 icmp_seq=272 ttl=60 time=45.937 ms
84 bytes from 11.0.0.10 icmp_seq=273 ttl=60 time=48.207 ms
84 bytes from 11.0.0.10 icmp_seq=274 ttl=60 time=45.569 ms
84 bytes from 11.0.0.10 icmp_seq=275 ttl=60 time=45.440 ms
84 bytes from 11.0.0.10 icmp_seq=276 ttl=60 time=45.427 ms
84 bytes from 11.0.0.10 icmp_seq=277 ttl=60 time=45.962 ms
84 bytes from 11.0.0.10 icmp_seq=278 ttl=60 time=45.885 ms
84 bytes from 11.0.0.10 icmp_seq=279 ttl=60 time=45.844 ms
84 bytes from 11.0.0.10 icmp_seq=280 ttl=60 time=45.537 ms
84 bytes from 11.0.0.10 icmp_seq=281 ttl=60 time=45.985 ms
84 bytes from 11.0.0.10 icmp_seq=282 ttl=60 time=45.661 ms
84 bytes from 11.0.0.10 icmp_seq=283 ttl=60 time=45.574 ms
84 bytes from 11.0.0.10 icmp_seq=284 ttl=60 time=46.321 ms
84 bytes from 11.0.0.10 icmp_seq=285 ttl=60 time=46.352 ms
84 bytes from 11.0.0.10 icmp_seq=286 ttl=60 time=46.039 ms
```

En las imágenes vemos como tras interrumpir RIP en R5 el ping deja de funcionar y pasados aproximadamente 3 minutos vuelve a funcionar correctamente.

Esto se debe a que finalmente se elimina la ruta que pasa por R5, ya que lleva tiempo sin estar disponible y se comienza a usar la ruta inferior que pasa por R3 y R2 que tiene un mayor coste.

PC2

Archivo Editar Ver Terminal Pestañas Ayuda

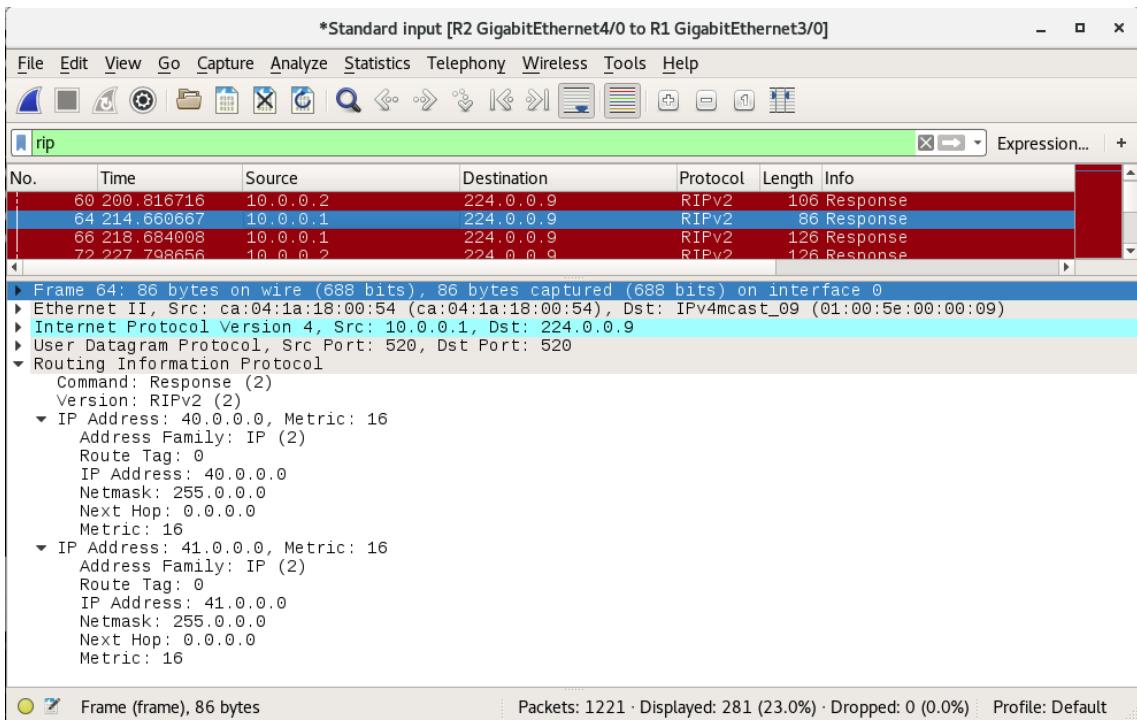
PC1 PC2 R4 R1

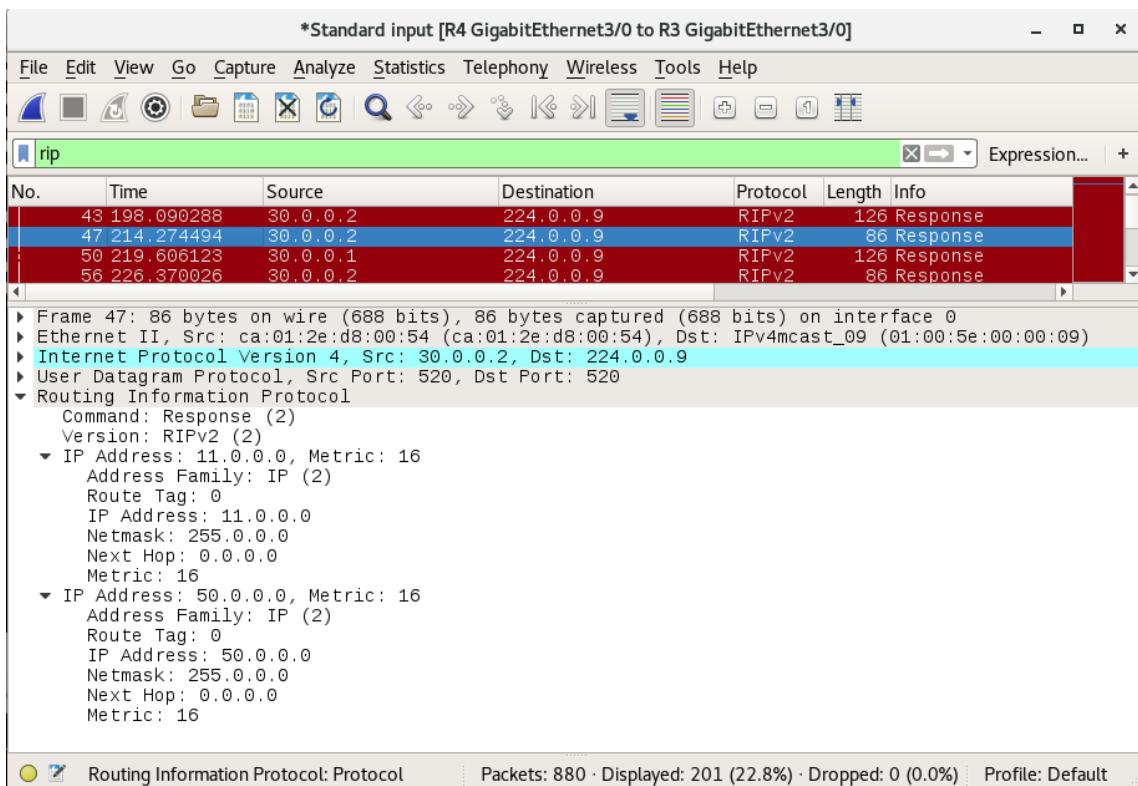
```
PC2> trace 11.0.0.10
trace to 11.0.0.10, 8 hops max, press Ctrl+C to stop
 1  41.0.0.1    9.971 ms   9.392 ms   9.859 ms
 2  30.0.0.1    19.324 ms   19.393 ms   19.223 ms
 3  20.0.0.1    29.342 ms   29.988 ms   29.491 ms
 4  10.0.0.1    40.003 ms   40.100 ms   40.029 ms
 5  * * *
 6  *11.0.0.10   45.974 ms (ICMP type:3, code:3, Destination port unreachable)

PC2>
```

En la imagen podemos observar la nueva ruta.

- Interrumpe las capturas. Analiza el tráfico capturado junto a los mensajes de depuración de R4 y R1. ¿Envían algún mensaje estos encaminadores en el momento en que el *tiempo* de algunas entradas llega a 3 minutos? ¿Por qué? ¿Qué mensajes de las capturas explican que estos encaminadores descubran las nuevas rutas? ¿Utiliza la implementación RIP de IOS *triggered update*?





Como se puede ver pasados los 3 minutos ambos routers envían a 224.0.0.9 un mensaje tipo RIPv2 de tipo Response en el que las métricas para llegar a 11.0.0.0 y 50.0.0.0 son 16 indicando que ha de ser borrada y dejar de anunciarse.

Posteriormente se buscan nuevas rutas alternativas.

Si que se usa la implementación RIP de IOS Triggered Update debido a la existencia de mensajes tipo Response enviados a 224.0.0.9 (multicast) para indicar que se ha aprendido una nueva ruta o ha cambiado la métrica de una existente.

- Vuelve a configurar de nuevo RIP en R5. Observa cómo cambian las tablas de encaminamiento en R4 y R1. ¿Cuánto tiempo han tardado en aprender la nueva ruta? ¿Por qué? Comprueba de nuevo cuál es ahora la ruta que están siguiendo los mensajes intercambiados entre PC2 y PC1 con trace. Incluye aquí la salida.

```

R1
Archivo Editar Ver Terminal Pestañas Ayuda
i0961594@lvdia01: ~          R1          R4
R1#show ip route rip
R  20.0.0.0/8 [120/1] via 10.0.0.2, 00:00:15, GigabitEthernet3/0
R  40.0.0.0/8 [120/3] via 10.0.0.2, 00:00:15, GigabitEthernet3/0
R  41.0.0.0/8 [120/3] via 10.0.0.2, 00:00:15, GigabitEthernet3/0
R  30.0.0.0/8 [120/2] via 10.0.0.2, 00:00:15, GigabitEthernet3/0
R1#

```

```

R4#show ip route rip
R    50.0.0.0/8 [120/3] via 30.0.0.1, 00:00:26, GigabitEthernet3/0
R    20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:26, GigabitEthernet3/0
R    10.0.0.0/8 [120/2] via 30.0.0.1, 00:00:26, GigabitEthernet3/0
R    11.0.0.0/8 [120/3] via 30.0.0.1, 00:00:26, GigabitEthernet3/0
R4#

```

Tablas de encaminamiento cuando R5 tiene desconfigurado RIP.

Como se ve, para llegar a las redes que no son vecinas el siguiente salto es siempre el mismo, el único que tiene disponible si no contamos con R5.

Volvemos a configurar R5 como habíamos hecho previamente.

```

R1#show ip route rip
R    20.0.0.0/8 [120/1] via 10.0.0.2, 00:00:17, GigabitEthernet3/0
R    40.0.0.0/8 [120/1] via 50.0.0.1, 00:00:18, GigabitEthernet4/0
R    41.0.0.0/8 [120/2] via 50.0.0.1, 00:00:18, GigabitEthernet4/0
R    30.0.0.0/8 [120/2] via 50.0.0.1, 00:00:18, GigabitEthernet4/0
                  [120/2] via 10.0.0.2, 00:00:17, GigabitEthernet3/0
R1#

```



```

R4#show ip route rip
R    50.0.0.0/8 [120/1] via 40.0.0.2, 00:00:24, GigabitEthernet4/0
R    20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:41, GigabitEthernet3/0
R    10.0.0.0/8 [120/2] via 40.0.0.2, 00:00:24, GigabitEthernet4/0
                  [120/2] via 30.0.0.1, 00:00:41, GigabitEthernet3/0
R    11.0.0.0/8 [120/2] via 40.0.0.2, 00:00:24, GigabitEthernet4/0
R4#

```

Tablas de encaminamiento cuando en R5 se ha vuelto a configurar RIP.

Como se ve, de forma casi instantánea se añaden las nuevas opciones de rutas en las que interviene R5. Esto se debe a que R5 envía mensajes de solicitud a R1 y R4 para que le envíen su tabla de rutas y estos contestan con un mensaje de tipo respuesta (unicast) cuyo contenido es el vector de distancias.

9. ¿Qué podemos concluir respecto a los tiempos de aprendizaje y eliminación de rutas?

Se puede concluir que el aprendizaje es muy rápido ya que en cuanto algún router se registra mediante los distintos mensajes se transmite el conocimiento y se actualizan las tablas de rutas, este proceso es casi inmediato.

Sin embargo, la eliminación de rutas es muy lenta (180 segundos), lo que supone una gran desventaja y problema ya que durante ese período de tiempo se envían mensajes por una ruta que ya no está disponible.

2. Órdenes IOS

RIP

Permite configurar el protocolo de encaminamiento dinámico RIP. Es necesario indicar la versión en la que trabajará RIP, las subredes por las que se propagarán las rutas, deshabilitar auto-summary y opcionalmente los temporizadores propios de RIP.

router rip	Activa RIP en nuestro router.
no router rip	Desactiva RIP (Si se desea volver a activar RIP hay que establecer de nuevo todos los parámetros).
versión <versión>	Indica la versión de RIP. 2.
timers basic <update> <timeout> <holddown> <flush>	Establece los temporizadores de RIP. Los valores predeterminados son 30, 180, 180, 240 respectivamente.
network <subred>	Indica la/s subred/es por las que se distribuirán las direcciones aprendidas.
passive-interface <ifaz>	Indica las interfaces por las que NO se distribuirán rutas.
no auto-summary	Deshabilita la opción de auto-summary que está habilitada por defecto.

Monitorizando el funcionamiento de RIP

debug ip rip	Habilita los mensajes de depuración.
no debug ip rip	Deshabilita los mensajes de depuración.
show ip protocols	Muestra detalles de los protocolos.

Tabla de rutas	
show ip route	Muestra la tabla de rutas
show ip route rip	Muestra las entradas de la tabla de rutas aprendidas con RIP.
clear ip route *	Borra la tabla de rutas