

# Trabalho 2 - Encaminhamento Dinâmico

Miguel Ferreira  
miguelferreira108@gmail.com  
Vanessa Silva  
up201305731@fc.up.pt

*Administração de Redes,  
Departamento de Ciências de Computadores,  
Faculdade de Ciências da Universidade do Porto*

18 de Abril de 2016

## Introdução

No âmbito da unidade curricular de Administração de Redes, implementamos a rede emulada descrita na figura 1.

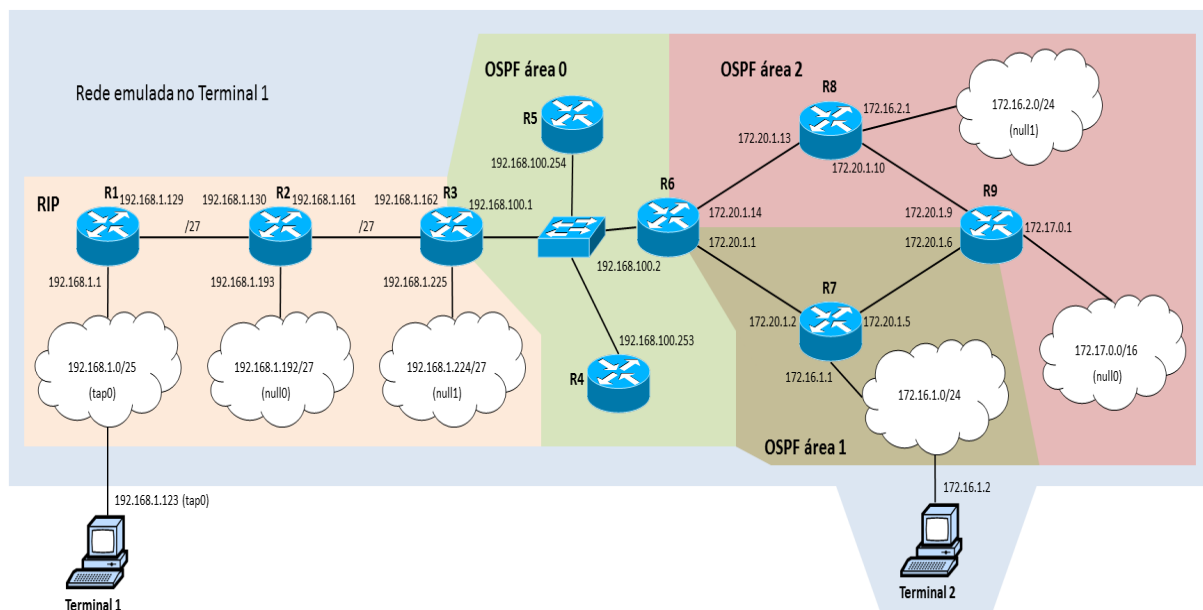


Figura 1: Rede emulada implementada na aula.

Na rede emulada os *routers* utilizados na simulação são da série 7200, o terminal 2 é um *router* que funciona como se fosse um terminal. Os terminais 1 e 2, usam como *default gateway* o *router* da rede à qual estão ligados, *router* R1 e R7, respetivamente. O *router* R3 corre RIP em duas interfaces, (192.168.1.225 e 192.168.1.162), e OSPF numa outra, (192.168.100.1). Este *router* redistribui no OSPF as rotas aprendidas pelo RIP e vice-versa.

## Questões

1. traceroute antes do corte da ligação entre R6 e R7:

```
R10#traceroute 172.16.1.2
```

```
Type escape sequence to abort.  
Tracing the route to 172.16.1.2
```

```
 1 192.168.1.1 64 msec 60 msec 60 msec  
 2 192.168.1.130 64 msec 60 msec 60 msec  
 3 192.168.1.162 88 msec 96 msec 100 msec  
 4 192.168.100.2 92 msec 88 msec 92 msec  
 5 172.20.1.2 120 msec 120 msec 128 msec  
 6 172.16.1.2 120 msec * 148 msec
```

traceroute após o corte da ligação entre R6 e R7:

```
R10#traceroute 172.16.1.2
```

```
Type escape sequence to abort.  
Tracing the route to 172.16.1.2
```

```
 1 192.168.1.1 36 msec 32 msec 36 msec  
 2 192.168.1.130 72 msec 60 msec 64 msec  
 3 192.168.1.162 88 msec 92 msec 64 msec  
 4 192.168.100.2 92 msec 88 msec 100 msec  
 5 * * *  
 6 * * *  
 7 *  
   192.168.1.162 !H *
```

a.

```
Router#show ip ospf database
```

```
OSPF Router with ID (0.0.0.6) (Process ID 6)
```

```
Router Link States (Area 0)
```

Link ID	ADV Router	Age	Seq#	Checksum	Link count
0.0.0.3	0.0.0.3	1670	0x80000003	0x00DDA3	1
0.0.0.4	0.0.0.4	1653	0x80000003	0x009DE6	1
0.0.0.5	0.0.0.5	1643	0x80000003	0x009BE5	1
0.0.0.6	0.0.0.6	1665	0x80000003	0x00B8C2	1

```
Net Link States (Area 0)
```

Link ID	ADV Router	Age	Seq#	Checksum
192.168.100.2	0.0.0.6	1665	0x80000003	0x008EB9

```
Summary Net Link States (Area 0)
```

Link ID	ADV Router	Age	Seq#	Checksum
172.16.2.0	0.0.0.6	1665	0x80000002	0x0090D2
172.17.0.0	0.0.0.6	1665	0x80000002	0x00FE5B
172.20.1.0	0.0.0.6	485	0x8000000A	0x00E480
172.20.1.8	0.0.0.6	1665	0x80000002	0x000952
172.20.1.12	0.0.0.6	1665	0x80000004	0x0078E6

#### Router Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
0.0.0.6	0.0.0.6	496	0x80000006	0x006EE9	1
0.0.0.7	0.0.0.7	800	0x80000006	0x007F90	3
0.0.0.9	0.0.0.9	1599	0x80000003	0x005630	1

#### Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
172.20.1.2	0.0.0.7	796	0x80000004	0x00233F
172.20.1.6	0.0.0.9	1599	0x80000002	0x001546

#### Summary Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
172.16.2.0	0.0.0.6	1666	0x80000002	0x0090D2
172.17.0.0	0.0.0.6	1666	0x80000002	0x00FE5B
172.20.1.8	0.0.0.6	1666	0x80000002	0x000952
172.20.1.12	0.0.0.6	1666	0x80000004	0x0078E6
192.168.100.0	0.0.0.6	1666	0x80000004	0x00C19A

#### Summary ASB Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
0.0.0.3	0.0.0.6	1666	0x80000002	0x009B8C

#### Router Link States (Area 2)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
0.0.0.6	0.0.0.6	1668	0x80000003	0x00403C	1
0.0.0.8	0.0.0.8	1603	0x80000004	0x000BE5	3
0.0.0.9	0.0.0.9	1600	0x80000003	0x006147	2

#### Net Link States (Area 2)

Link ID	ADV Router	Age	Seq#	Checksum
172.20.1.9	0.0.0.9	1600	0x80000002	0x000552
172.20.1.13	0.0.0.8	1603	0x80000002	0x00BC9A

#### Summary Net Link States (Area 2)

Link ID	ADV Router	Age	Seq#	Checksum
172.20.1.0	0.0.0.6	487	0x8000000A	0x00E480

```
192.168.100.0    0.0.0.6          1668          0x80000004 0x00C19A
```

#### Summary ASB Link States (Area 2)

Link ID	ADV Router	Age	Seq#	Checksum
0.0.0.3	0.0.0.6	1668	0x80000002	0x009B8C

#### Type-5 AS External Link States

Link ID	ADV Router	Age	Seq#	Checksum	Tag
192.168.1.0	0.0.0.3	1673	0x80000002	0x003D24	0
192.168.1.128	0.0.0.3	1673	0x80000002	0x007A06	0
192.168.1.160	0.0.0.3	1673	0x80000002	0x003927	0
192.168.1.192	0.0.0.3	1673	0x80000002	0x00F748	0
192.168.1.224	0.0.0.3	1673	0x80000002	0x00B669	0

b. O corte da ligação entre R6 e R7 não fez com que o percurso dos pacotes fosse desviado de modo a atravessar R8 e R9, uma vez que, como podemos verificar no `show ip ospf database` obtido na alínea anterior, o *router* R6 "conhece" todos os *routers* envolvidos e em especial o R7, (Router Link States (Area 1)), para a qual tem uma rota intra-área que é sempre preferida pelo OSPF.

## 2.

a. Como podemos verificar a baixo, no *router* R8, o estado do vizinho R9 fica DOWN, (state DOWN). O evento que ocorreu nesse instante e despoletou essa mudança foi o IE2 (ver lista dos eventos de transição na tabela 8-1 do livro Routing TCP/IP, vol.1, 2ª ed).

```
Router#debug ip ospf adj
OSPF adjacency events debugging is on
Router#
*Apr 14 15:58:38.411: OSPF: 0.0.0.9 address 172.20.1.9 on FastEthernet1/1 is dead
*Apr 14 15:58:38.411: OSPF: 0.0.0.9 address 172.20.1.9 on FastEthernet1/1 is dead,
state DOWN
*Apr 14 15:58:38.411: %OSPF-5-ADJCHG: Process 8, Nbr 0.0.0.9 on FastEthernet1/1
from FULL to DOWN, Neighbor Down: Dead timer expired
*Apr 14 15:58:38.415: OSPF: Neighbor change Event on interface FastEthernet1/1
*Apr 14 15:58:38.415: OSPF: DR/BDR election on FastEthernet1/1
*Apr 14 15:58:38.415: OSPF: Elect BDR 0.0.0.8
*Apr 14 15:58:38.415: OSPF: Elect DR 0.0.0.8
*Apr 14 15:58:38.415: OSPF: Elect BDR 0.0.0.0
*Apr 14 15:58:38.415: OSPF: Elect DR 0.0.0.8
*Apr 14 15:58:38.419: DR: 0.0.0.8 (Id) BDR: none
*Apr 14 15:58:38.419: OSPF: Remember old DR 0.0.0.9 (id)
*Apr 14 15:58:38.915: OSPF: Build router LSA for area 2, router ID 0.0.0.8,
seq 0x80000005
*Apr 14 15:58:38.919: OSPF: No full nbrs to build Net Lsa for interface
FastEthernet1/1
```

b. R8 envia um *LS Update* para a rede *multicast* 224.0.0.5 no instante em que detecta que R9 "morreu", nesta mensagem é enviado um LSA (*Router-LSA*). O objetivo dos LAS's é permitir construir um grafo dirigido da rede para posteriormente se calcular o caminho mais curto para cada destino.

O *Router LSA*, enviado por R8, serve para este se anunciar e identificar as suas ligações na área a que pertence juntamente com as respetivas métricas, de forma a se poder cumprir com o objetivos dos LSA's. Uma vez que R8 detecta que um vizinho, R9, "morreu" este envia um *Router LSA* para se poder "construir" um novo grafo dirigido com a atualização das suas ligações. R8 informa então que tem 3 ligações disponíveis, **Number of Links=3**, em que uma é de trânsito, **Type: Transit**, (informando que o endereço IP do *Designated Router* é 172.20.1.13, o próprio R8, e a métrica é 10), e as duas ligações restantes são *stub* (ponta), **Type: stub**, em que uma tem número de IP da rede/subrede 172.20.1.8 com métrica 10 e a outra tem número de IP da rede/subrede 172.16.2.0 com métrica 10.

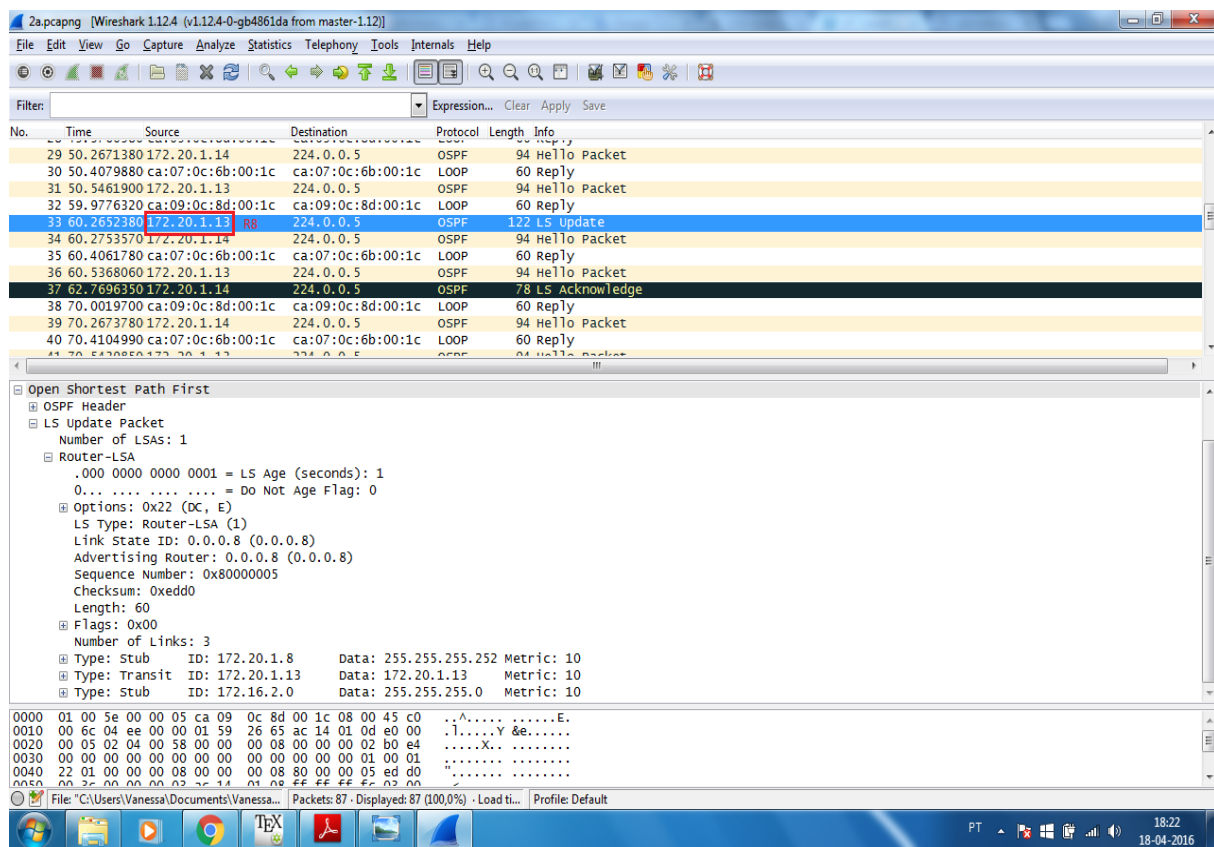


Figura 2: Captura de pacotes na ligação entre R6 e R8.

c. A mensagem que permite ao *router* R9 saber que tem comunicação bidirecional (*two-way*) com R8 apresenta-se na figura 3, devidamente sinalizada. Escolhemos esta mensagem, *Hello Packet*, uma vez que é o sub-protocolo *Hello* que verifica conexões bidirecionais entre os vizinhos. Quando um *router*, R8, detecta o seu *Router ID* num *Hello* recebido, sabe que existe comunicação bidirecional com o vizinho que o enviou, R9, como se pode analisar na figura.

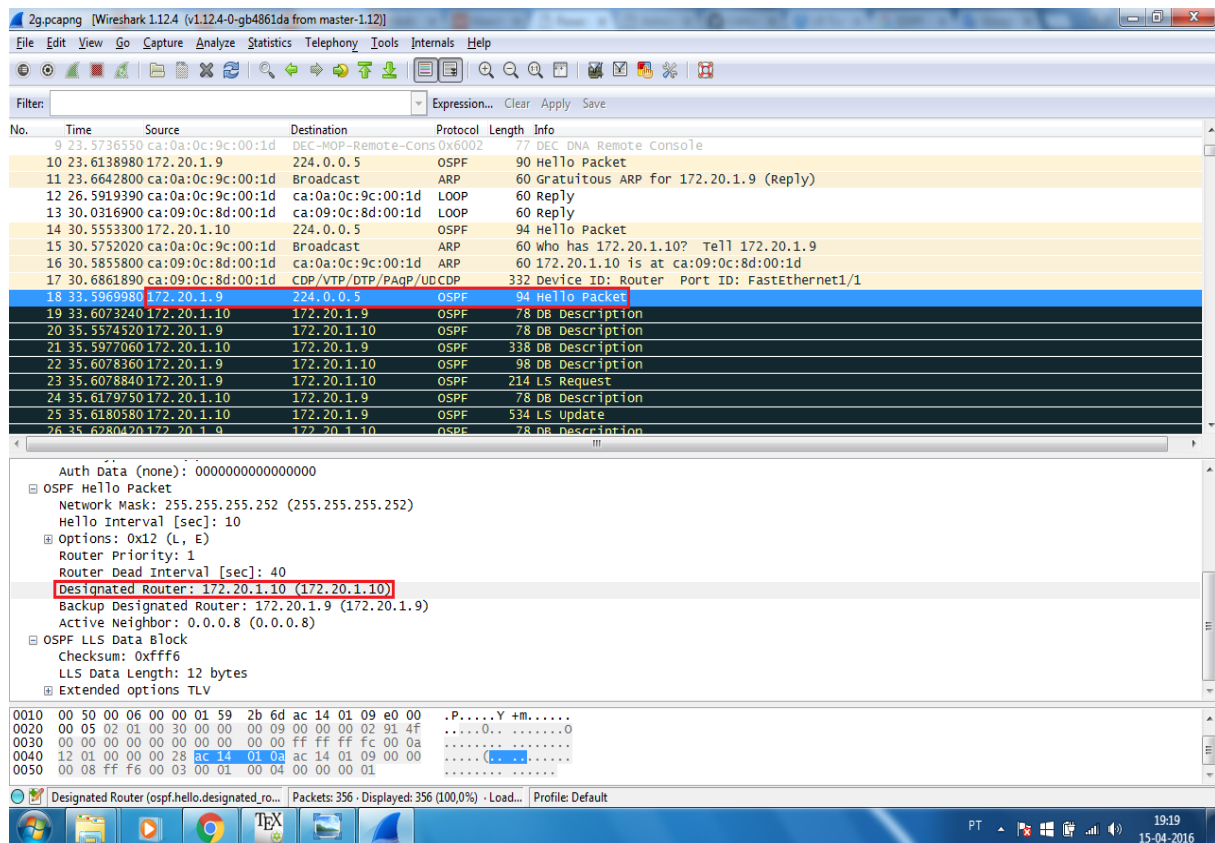


Figura 3: Captura de pacotes na ligação entre R8 e R9.

d. No ficheiro que contem a captura de R8-R9 (Capturas Wireshark/2g.pcapng) encontramos vários tipos de anúncios em cada mensagem *LS Update*:

- Tipo 1 - Router LSA - no pacote 25, servindo para o *router* R8 identificar e partilhar as suas ligações.
- Tipo 2 - Network LSA - servindo para o *router* designado informar sobre os *routers* ligado ao nó virtual correspondente à rede do DR.
- Tipo 3 - Network Summary - informa sobre os destinos sumarizados com custos a partir do Area Border Router, neste caso o R3.
- Tipo 4 - ASBR Summary - Rota para o *router Autonomous System Boundary Router* que gerou os LSAs de tipo 5.
- Tipo 5 - AS External LSA - servindo para o *router* informar sobre os destinos externos ao OSPF. Neste caso em particular, as rotas externas são todas do Tipo 2 pois são importadas de uma métrica incompatível com OSPF.

e. Campos Link State ID:

- Tipo 1 - ID do router que gerou o LSA.
- Tipo 2 - Endereço IP da interface do DR nessa rede.
- Tipo 3 - Endereço IP da rede de destino.

- Tipo 4 - Router ID do ASBR que gerou o AS External LSA.
- Tipo 5 - Endereço IP da rede de destino do External LSA.

Advertising Router - ID do *router* que gera o LSA. Nos LSA do tipo 1, dentro de cada entrada LSA (LinkID, LinkData, LinkType, Metric), podemos construir um grafo com adjacência entre destinos e o *router* que emite o LSA. Exemplo (pacote 25, *advertising router* 0.0.0.8):

```
Type: Stub      ID: 172.20.1.8      Data: 255.255.255.252 Metric: 10
  Link ID: 172.20.1.8 - IP network/subnet number
  Link Data: 255.255.255.252
  Link Type: 3 - Connection to a stub network
  Number of Metrics: 0 - TOS
  0 Metric: 10
```

Nos LSA do tipo 2, a informação em “Attached router” permite separar o grafo nas suas componentes fortemente conexas, separando-o em “áreas”:

```
LSA-type 2 (Network-LSA), len 32
  .000 0000 1101 1101 = LS Age (seconds): 221
  0... .... .... .... = Do Not Age Flag: 0
  Options: 0x22 ((DC) Demand Circuits, (E) External Routing)
  LS Type: Network-LSA (2)
  Link State ID: 172.20.1.13
  Advertising Router: 0.0.0.8
  Sequence Number: 0x80000003
  Checksum: 0xba9b
  Length: 32
  Netmask: 255.255.255.252
  Attached Router: 0.0.0.8
  Attached Router: 0.0.0.6
```

Nos LSA do tipo 5, podemos acrescentar ao grafo informação sobre os destinos externos ao OSPF e a sua adjacência ao ASBR:

```
LSA-type 5 (AS-External-LSA (ASBR)), len 36
  .000 0000 1111 0101 = LS Age (seconds): 245
  0... .... .... .... = Do Not Age Flag: 0
  Options: 0x20 ((DC) Demand Circuits)
  LS Type: AS-External-LSA (ASBR) (5)
  Link State ID: 192.168.1.224
  Advertising Router: 0.0.0.3
  Sequence Number: 0x80000003
  Checksum: 0xb46a
  Length: 36
  Netmask: 255.255.255.224
  1... .... = External Type: Type 2 (metric is larger than any other link state path)
  .000 0000 = TOS: 0
  Metric: 100
  Forwarding Address: 0.0.0.0
  External Route Tag: 0
```

f. O último anúncio do tipo *Router LSA* originado pelo *router* R8 é o pacote 1761, do ficheiro Capturas Wireshark/2g.pcapng, que se apresenta a seguir e onde podemos ver, para cada uma das ligações, o respetivo tipo, campos ID e Data e os seus significados:

```

LSA-type 1 (Router-LSA), len 60
.000 0000 0000 0101 = LS Age (seconds): 5
0... .... .... = Do Not Age Flag: 0
Options: 0x22 ((DC) Demand Circuits, (E) External Routing)
LS Type: Router-LSA (1)
Link State ID: 0.0.0.8
Advertising Router: 0.0.0.8
Sequence Number: 0x80000003
Checksum: 0x0de4
Length: 60
Flags: 0x00
Number of Links: 3
Type: Transit ID: 172.20.1.9      Data: 172.20.1.10      Metric: 10
  Link ID: 172.20.1.9 - IP address of Designated Router
  Link Data: 172.20.1.10 - endereço IP do router nessa rede
  Link Type: 2 - Connection to a transit network
  Number of Metrics: 0 - TOS
  0 Metric: 10
Type: Transit ID: 172.20.1.13     Data: 172.20.1.13     Metric: 10
  Link ID: 172.20.1.13 - IP address of Designated Router
  Link Data: 172.20.1.13 - endereço IP do router nessa rede
  Link Type: 2 - Connection to a transit network
  Number of Metrics: 0 - TOS
  0 Metric: 10
Type: Stub ID: 172.16.2.0         Data: 255.255.255.0   Metric: 10
  Link ID: 172.16.2.0 - IP network/subnet number (prefixo)
  Link Data: 255.255.255.0 - máscara da rede
  Link Type: 3 - Connection to a stub network
  Number of Metrics: 0 - TOS
  0 Metric: 10

```

g. Pacote 1737: R8 a R9.

```

LSA-type 1 (Router-LSA), len 60
.000 0000 0010 1011 = LS Age (seconds): 43
0... .... .... = Do Not Age Flag: 0
Options: 0x22 ((DC) Demand Circuits, (E) External Routing)
LS Type: Router-LSA (1)
Link State ID: 0.0.0.8
Advertising Router: 0.0.0.8
Sequence Number: 0x80000001
Checksum: 0xdab9
Length: 60
Flags: 0x00
Number of Links: 3
Type: Stub ID: 172.20.1.8        Data: 255.255.255.252 Metric: 10
  Link ID: 172.20.1.8 - IP network/subnet number
  Link Data: 255.255.255.252
  Link Type: 3 - Connection to a stub network
  Number of Metrics: 0 - TOS
  0 Metric: 10
Type: Stub ID: 172.20.1.12       Data: 255.255.255.252 Metric: 10

```



```

Link ID: 172.20.1.12 - IP network/subnet number
Link Data: 255.255.255.252
Link Type: 3 - Connection to a stub network
Number of Metrics: 0 - TOS
0 Metric: 10
Type: Stub      ID: 172.16.2.0      Data: 255.255.255.0  Metric: 10
Link ID: 172.16.2.0 - IP network/subnet number
Link Data: 255.255.255.0
Link Type: 3 - Connection to a stub network
Number of Metrics: 0 - TOS
0 Metric: 10

```

Após um *Hello Packet* (cujo propósito é estabelecer conexões bidirecionais) de R9, o *router* R8 responde com uma *BD Description* cujo propósito é descrever as ligações entre cada *router* com os seus parceiros adjacentes. Seguidamente há um *LS Request* de R9 para R10, **terminal** 1, pedindo informação desta forma. Como resposta a este pedido o R10 envia um *LS Update*. Cada LSA é finalmente confirmado com um *LS Acknowledge* por cada recetor.

### 3. Router R3:

```

Router#show ip ospf interface FastEthernet 1/1
%OSPF: OSPF not enabled on FastEthernet1/1
Router#show ip ospf interface FastEthernet 1/0
%OSPF: OSPF not enabled on FastEthernet1/0
Router#show ip ospf interface FastEthernet 2/0
FastEthernet2/0 is up, line protocol is up
  Internet Address 192.168.100.1/24, Area 0
  Process ID 3, Router ID 0.0.0.3, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DROTHER, Priority 1
  Designated Router (ID) 0.0.0.6, Interface address 192.168.100.2
  Backup Designated router (ID) 0.0.0.5, Interface address 192.168.100.254
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:00
  Supports Link-local Signaling (LLS)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 5
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 3, Adjacent neighbor count is 2
    Adjacent with neighbor 0.0.0.5  (Backup Designated Router)
    Adjacent with neighbor 0.0.0.6  (Designated Router)
  Suppress hello for 0 neighbor(s)

```

### Router R4:

```

Router#show ip ospf interface FastEthernet 1/0
FastEthernet1/0 is up, line protocol is up
  Internet Address 192.168.100.253/24, Area 0
  Process ID 4, Router ID 0.0.0.4, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DROTHER, Priority 1
  Designated Router (ID) 0.0.0.6, Interface address 192.168.100.2
  Backup Designated router (ID) 0.0.0.5, Interface address 192.168.100.254

```

```
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  oob-resync timeout 40
  Hello due in 00:00:08
Supports Link-local Signaling (LLS)
Index 1/1, flood queue length 0
Next 0x0(0)/0x0(0)
Last flood scan length is 0, maximum is 1
Last flood scan time is 0 msec, maximum is 4 msec
Neighbor Count is 3, Adjacent neighbor count is 2
  Adjacent with neighbor 0.0.0.5 (Backup Designated Router)
  Adjacent with neighbor 0.0.0.6 (Designated Router)
Suppress hello for 0 neighbor(s)
```

*Router R5:*

```
Router#show ip ospf interface FastEthernet 1/0
FastEthernet1/0 is up, line protocol is up
  Internet Address 192.168.100.254/24, Area 0
  Process ID 5, Router ID 0.0.0.5, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 0.0.0.6, Interface address 192.168.100.2
  Backup Designated router (ID) 0.0.0.5, Interface address 192.168.100.254
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 3, Adjacent neighbor count is 3
    Adjacent with neighbor 0.0.0.3
    Adjacent with neighbor 0.0.0.4
    Adjacent with neighbor 0.0.0.6 (Designated Router)
  Suppress hello for 0 neighbor(s)
```

*Router R6:*

```
Router#show ip ospf interface FastEthernet 1/0
FastEthernet1/0 is up, line protocol is up
  Internet Address 172.20.1.14/30, Area 2
  Process ID 6, Router ID 0.0.0.6, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 0.0.0.8, Interface address 172.20.1.13
  Backup Designated router (ID) 0.0.0.6, Interface address 172.20.1.14
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Index 1/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 6
  Last flood scan time is 0 msec, maximum is 4 msec
```

```

Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 0.0.0.8 (Designated Router)
  Suppress hello for 0 neighbor(s)
Router#show ip ospf interface FastEthernet 1/1
FastEthernet1/1 is up, line protocol is up
  Internet Address 172.20.1.1/30, Area 1
  Process ID 6, Router ID 0.0.0.6, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 0.0.0.6, Interface address 172.20.1.1
  Backup Designated router (ID) 0.0.0.7, Interface address 172.20.1.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 6
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 0.0.0.7 (Backup Designated Router)
    Suppress hello for 0 neighbor(s)
Router#show ip ospf interface FastEthernet 2/0
FastEthernet2/0 is up, line protocol is up
  Internet Address 192.168.100.2/24, Area 0
  Process ID 6, Router ID 0.0.0.6, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 0.0.0.6, Interface address 192.168.100.2
  Backup Designated router (ID) 0.0.0.5, Interface address 192.168.100.254
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:00
  Supports Link-local Signaling (LLS)
  Index 1/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 8
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 3, Adjacent neighbor count is 3
    Adjacent with neighbor 0.0.0.3
    Adjacent with neighbor 0.0.0.4
    Adjacent with neighbor 0.0.0.5 (Backup Designated Router)
  Suppress hello for 0 neighbor(s)

```

a. O *router* R3 contém 2 adjacências, na interface FastEthernet 2/0.  
 O *router* R4 contém 2 adjacências, na interface FastEthernet 1/0.  
 O *router* R5 contém 3 adjacências, na interface FastEthernet 1/0.  
 O *router* R6 contém 1 adjacências, na interface FastEthernet 1/0, 1 adjacências, na interface FastEthernet 1/1 e 3 adjacências, na interface FastEthernet 2/0, tendo assim um total de 5 adjacências.  
 R6 é o *Designated Router* e por isso, não é de estranhar que seja o *router* com maior número de adjacências, isto implicaria um grande volume de troca de informações de encaminhamento.  
 R5 é o segundo *router* com maior número de adjacências e por isso é o *Backup Designated Router*,

que tem o papel de toma o lugar do *Designated Router* caso ele falhe.

Já R3 e R4 são *DROthers* que estabelecem adjacências apenas com o *Designated Router*, R6, e com o *Backup Designated Router*, R5.

b. No OSPF o *Designated Router* serve para gerir o processo de inundação na rede de acesso múltiplo e gerar informação topológica sobre um nó virtual que representa a rede de acesso múltiplo.

4.

a. O caminho seguido pelos pacotes ICMP *echo request* é do IP 172.16.1.2, **Terminal 2**, passando por **R7**, de seguida por **R6** e por **R8** chegando ao destino, interface 172.17.0.1 de R9. O caminho seguido pelos pacotes ICMP *echo reply* é da interface 172.17.0.1 de **R9** passando por **R7** e por último chega ao destino, **Terminal 2**.

Os pacotes ICMP seguem caminhos diferentes por causa das rotas especificadas pelo protocolo OSPF para cada *router*. O *router* R7 tem rota definida para R9 através de R6, (0 IA 172.17.0.0/16 [110/40] via 172.20.1.1, 00:16:38, FastEthernet1/0), enquanto que o *router* R9 tem a rota para Terminal 2 através de R7, (0 172.16.1.0 [110/20] via 172.20.1.5, 00:17:29, FastEthernet1/0).

No.	Time	Source	Destination	Protocol	Length	Info
26	56.31993300	172.20.1.2	224.0.0.5	OSPF	94	Hello
27	56.36117800	ca:08:0c:7a:00:1c	ca:08:0c:7a:00:1c	LOOP	60	Reply
28	59.74067000	172.16.1.2	172.17.0.1	ICMP	114	Echo
29	60.01370100	172.20.1.1	224.0.0.5	OSPF	94	Hello
30	60.25650600	ca:07:0c:6b:00:1d	ca:07:0c:6b:00:1d	LOOP	60	Reply
31	61.04546300	172.16.1.2	172.17.0.1	ICMP	114	Echo
32	61.10883000	172.16.1.2	172.17.0.1	ICMP	114	Echo
33	61.17180700	172.16.1.2	172.17.0.1	ICMP	114	Echo
34	61.22367500	172.16.1.2	172.17.0.1	ICMP	114	Echo
35	66.33100700	172.20.1.2	224.0.0.5	OSPF	94	Hello

Figura 4: Captura de pacotes na ligação entre as interfaces 172.20.1.2 e 172.20.1.5 de R7.

b. Nós tentamos alterar o custo da interface 172.20.1.6 para um valor elevado, (`ip ospf cost 65000`), para tentar que a ligação se corrompa, de modo a que o percurso do *echo reply* seja o inverso do do *echo request*. Tentamos ainda para o valor máxima, (`ip ospf cost 65535`), mas não obtemos sucesso.

O percurso não foi alterado pois os *routers* internos de cada área têm penas uma base de dados topológica dessa área. O *router* R6 é um *Area Border Router* logo, conhece a topologia das áreas às quais está ligado, área 1 e 2. Deste modo, uma vez que o **Terminal 2** só pertence à área 1, o melhor caminho continua a ser por essa área, e não passando por outra área, área 2.

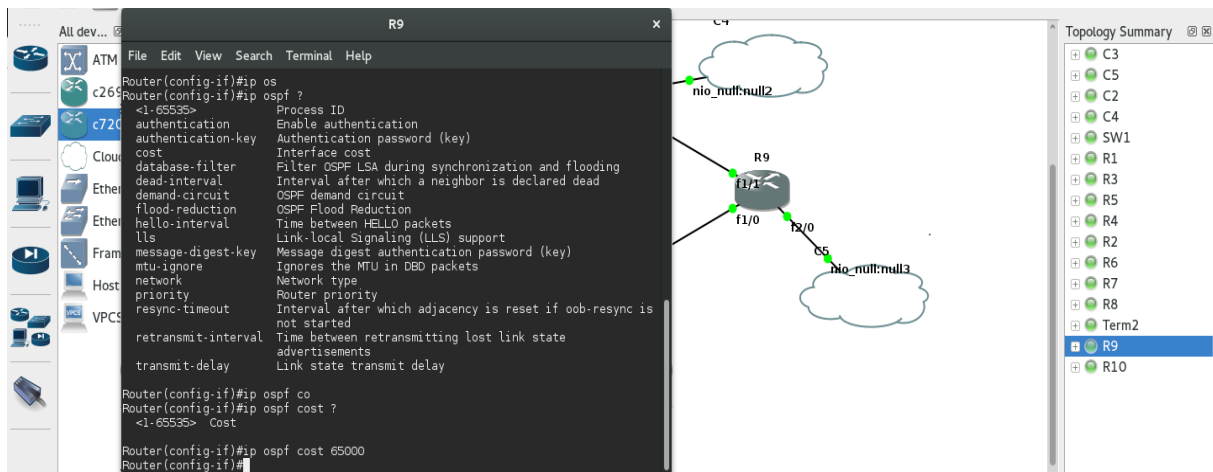


Figura 5: Alteração do custo da interface 172.20.1.6 de R9.

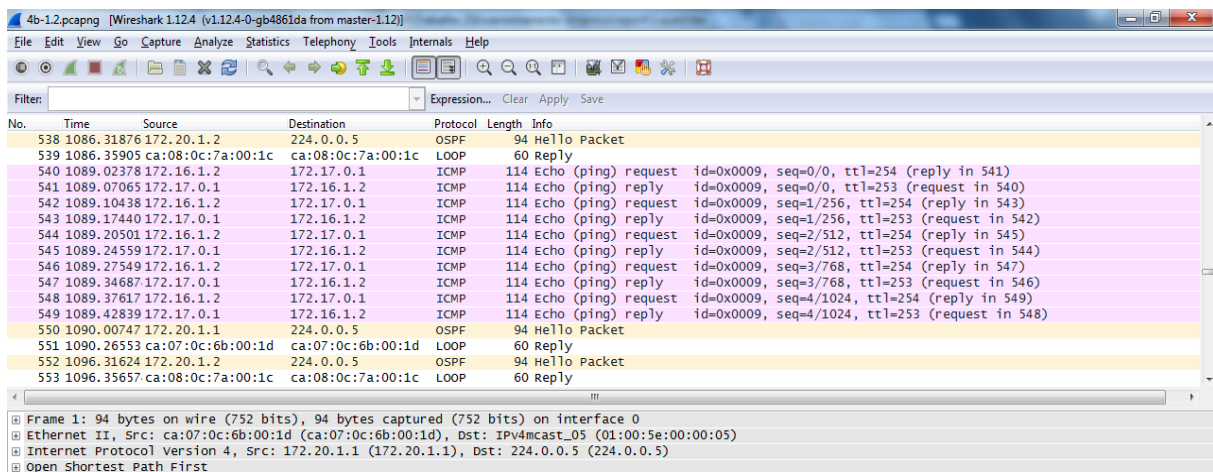


Figura 6: Captura de pacotes na ligação entre as interfaces 172.20.1.2 e 172.20.1.5 de R7.

5. Como podemos verificar na figura 7, as mensagens RIP trocadas entre R1 e R2 são inicialmente enviadas para o IP 255.255.255.255, uma vez que o *broadcast* estava ativado. Uma vez desativado as mensagens RIP são enviadas para o IP 224.0.0.9.

Os *routers* RIP foram configurados de modo a correrem a versão 2 do protocolo, e por isso as mensagens são enviadas para o IP 224.0.0.9, que é um endereço *multicast* "todos os *routers* IPv2". Isto evita que os terminais recebam os anúncios, que não lhes interessam.

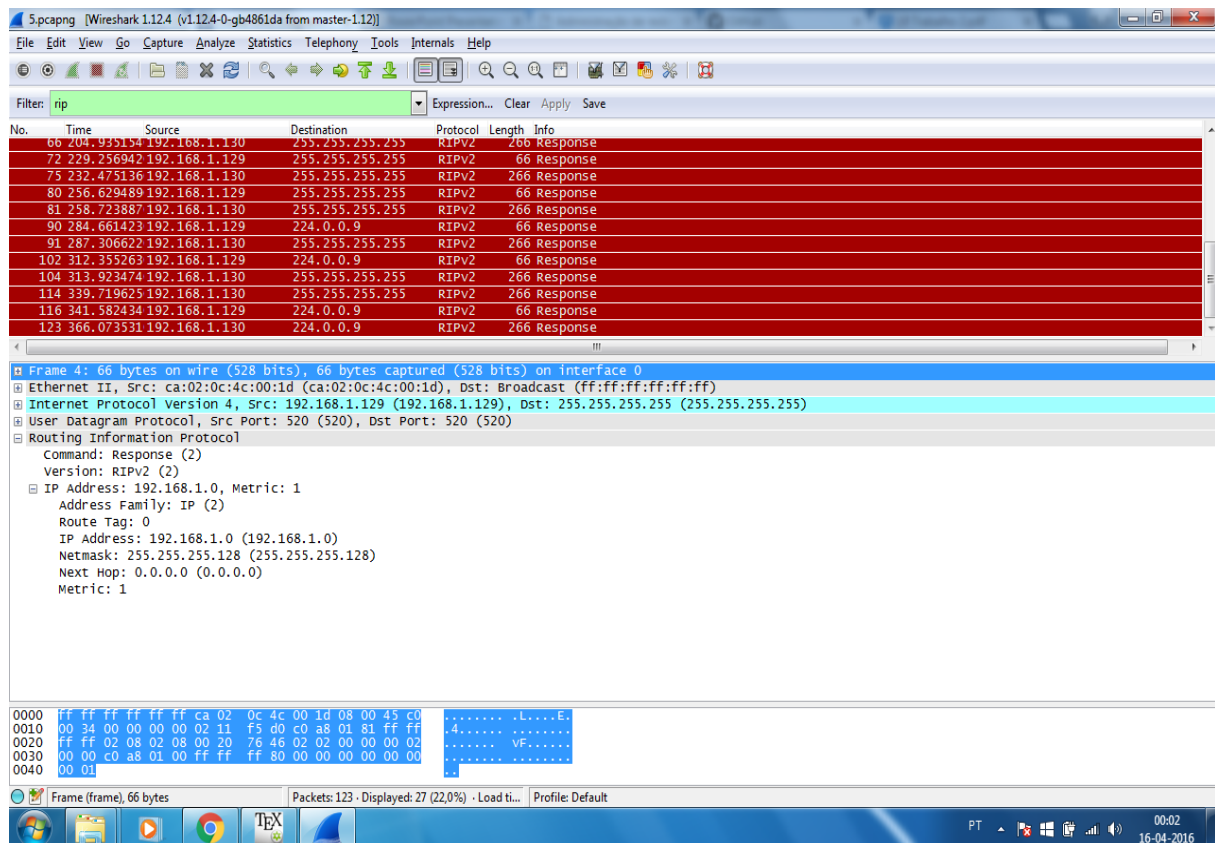


Figura 7: Capture de mensagens RIP trocadas entre R1 e R2.

## 6.

```
*Apr 14 17:00:31.055: RIP: sending v2 update to 224.0.0.9 via
FastEthernet1/1 (192.168.1.130)
*Apr 14 17:00:31.055: RIP: build update entries
*Apr 14 17:00:31.055: 172.16.1.0/24 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:31.059: 172.16.2.0/24 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:31.059: 172.17.0.0/16 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:31.059: 172.20.1.0/30 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:31.059: 172.20.1.4/30 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:31.063: 172.20.1.8/30 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:31.063: 172.20.1.12/30 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:31.063: 192.168.1.160/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:00:31.067: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:00:31.067: 192.168.1.224/27 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:00:31.067: 192.168.100.0/24 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:35.871: RIP: received v2 update from 192.168.1.129 on
FastEthernet1/1
*Apr 14 17:00:35.875: 192.168.1.0/25 via 0.0.0.0 in 1 hops
*Apr 14 17:00:43.175: RIP: sending v2 update to 255.255.255.255 via
FastEthernet2/0 (192.168.1.161)
*Apr 14 17:00:43.175: RIP: build update entries
*Apr 14 17:00:43.175: 192.168.1.0/25 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:00:43.179: 192.168.1.128/27 via 0.0.0.0, metric 1, tag 0
```

```

*Apr 14 17:00:43.179: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:00:48.691: RIP: received v2 update from 192.168.1.162 on
FastEthernet2/0
*Apr 14 17:00:48.695: 172.16.1.0/24 via 0.0.0.0 in 5 hops
*Apr 14 17:00:48.695: 172.16.2.0/24 via 0.0.0.0 in 5 hops
*Apr 14 17:00:48.695: 172.17.0.0/16 via 0.0.0.0 in 5 hops
*Apr 14 17:00:48.699: 172.20.1.0/30 via 0.0.0.0 in 5 hops
*Apr 14 17:00:48.699: 172.20.1.4/30 via 0.0.0.0 in 5 hops
*Apr 14 17:00:48.699: 172.20.1.8/30 via 0.0.0.0 in 5 hops
*Apr 14 17:00:48.699: 172.20.1.12/30 via 0.0.0.0 in 5 hops
*Apr 14 17:00:48.703: 192.168.1.224/27 via 0.0.0.0 in 1 hops
*Apr 14 17:00:48.703: 192.168.100.0/24 via 0.0.0.0 in 5 hops
*Apr 14 17:00:57.695: RIP: sending v2 update to 224.0.0.9 via
FastEthernet1/1 (192.168.1.130)
*Apr 14 17:00:57.695: RIP: build update entries
*Apr 14 17:00:57.695: 172.16.1.0/24 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:57.699: 172.16.2.0/24 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:57.699: 172.17.0.0/16 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:57.699: 172.20.1.0/30 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:57.699: 172.20.1.4/30 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:57.703: 172.20.1.8/30 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:57.703: 172.20.1.12/30 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:00:57.707: 192.168.1.160/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:00:57.707: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:00:57.711: 192.168.1.224/27 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:00:57.711: 192.168.100.0/24 via 0.0.0.0, metric 6, tag 0
*Apr 14 17:01:06.039: RIP: received v2 update from 192.168.1.129 on
FastEthernet1/1
*Apr 14 17:01:06.039: 192.168.1.0/25 via 0.0.0.0 in 1 hops

```

a. As mensagens enviadas na interface **FastEthernet1/1** dizem respeito às redes do lado direito de R2, enquanto que as mensagens enviadas na interface **FastEthernet2/0** dizem respeito às redes do lado esquerdo de R2.

Podemos ainda verificar que as mensagens que dizem respeito aos *routers* que correm RIP têm métrica=1 e os que correm OSPF têm métrica=6.

Cada interface apenas envia informação sobre os destinos e redes atingíveis por outras interfaces. A **FastEthernet 2/0** envia informações sobre a rede 192.168.1.0/25, com métrica 2 (pois é calculada pelo RIP). A **FastEthernet1/1** envia informações sobre os destinos alcançáveis pela interface **FastEthernet2/0**, por exemplo 172.16.2.0 com métrica 6 (importada por OSPF).

b. Este mecanismo chama-se "*split horizon*", e a sua função é não anunciar numa interface as rotas que aprendeu por ela.

## 7.

```

Router#no ip split-horizon
Router#no redistribute rip

```

a.

```

Router#debug ip rip
*Apr 14 17:09:14.219: RIP: received v2 update from 192.168.1.129 on

```

```

FastEthernet1/1
*Apr 14 17:09:14.219:      192.168.1.0/25 via 0.0.0.0 in 1 hops
*Apr 14 17:09:14.223:      192.168.100.0/24 via 0.0.0.0 in 16 hops
(inaccessible)
*Apr 14 17:09:19.119: RIP: sending v2 update to 255.255.255.255 via
FastEthernet2/0 (192.168.1.161)
*Apr 14 17:09:19.119: RIP: build update entries
*Apr 14 17:09:19.119: 192.168.1.0/25 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:09:19.119: 192.168.1.128/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:09:19.123: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:09:19.123: 192.168.100.0/24 via 0.0.0.0, metric 16, tag 0
*Apr 14 17:09:24.187: RIP: received v2 update from 192.168.1.162 on
FastEthernet2/0
*Apr 14 17:09:24.187:      192.168.1.224/27 via 0.0.0.0 in 1 hops
*Apr 14 17:09:24.191:      192.168.100.0/24 via 0.0.0.0 in 16 hops
(inaccessible)
*Apr 14 17:09:25.543: RIP: sending v2 update to 255.255.255.255 via
FastEthernet1/1 (192.168.1.130)
*Apr 14 17:09:25.543: RIP: build update entries
*Apr 14 17:09:25.547: 192.168.1.160/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:09:25.547: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:09:25.547: 192.168.1.224/27 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:09:25.547: 192.168.100.0/24 via 0.0.0.0, metric 16, tag 0
*Apr 14 17:09:40.339: RIP: received v2 update from 192.168.1.129 on
FastEthernet1/1
*Apr 14 17:09:40.339:      192.168.1.0/25 via 0.0.0.0 in 1 hops
*Apr 14 17:09:46.243: RIP: sending v2 update to 255.255.255.255 via
FastEthernet2/0 (192.168.1.161)
*Apr 14 17:09:46.243: RIP: build update entries
*Apr 14 17:09:46.243: 192.168.1.0/25 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:09:46.247: 192.168.1.128/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:09:46.247: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:09:53.971: RIP: sending v2 update to 255.255.255.255 via
FastEthernet1/1 (192.168.1.130)
*Apr 14 17:09:53.971: RIP: build update entries
*Apr 14 17:09:53.971: 192.168.1.160/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:09:53.975: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:09:53.975: 192.168.1.224/27 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:10:07.383: RIP: received v2 update from 192.168.1.129 on
FastEthernet1/1
*Apr 14 17:10:07.383:      192.168.1.0/25 via 0.0.0.0 in 1 hops
*Apr 14 17:10:13.087: RIP: sending v2 update to 255.255.255.255 via
FastEthernet2/0 (192.168.1.161)
*Apr 14 17:10:13.087: RIP: build update entries
*Apr 14 17:10:13.087: 192.168.1.0/25 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:10:13.087: 192.168.1.128/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:10:13.091: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:10:22.991: RIP: sending v2 update to 255.255.255.255 via
FastEthernet1/1 (192.168.1.130)
*Apr 14 17:10:22.991: RIP: build update entries
*Apr 14 17:10:22.991: 192.168.1.160/27 via 0.0.0.0, metric 1, tag 0

```



```

*Apr 14 17:10:22.995: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:10:22.995: 192.168.1.224/27 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:10:32.907: RIP: received v2 update from 192.168.1.129 on
FastEthernet1/1
*Apr 14 17:10:32.907: 192.168.1.0/25 via 0.0.0.0 in 1 hops
*Apr 14 17:10:42.323: RIP: sending v2 update to 255.255.255.255 via
FastEthernet2/0 (192.168.1.161)
*Apr 14 17:10:42.323: RIP: build update entries
*Apr 14 17:10:42.323: 192.168.1.0/25 via 0.0.0.0, metric 2, tag 0
*Apr 14 17:10:42.327: 192.168.1.128/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:10:42.327: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:10:48.643: RIP: sending v2 update to 255.255.255.255 via
FastEthernet1/1 (192.168.1.130)
*Apr 14 17:10:48.643: RIP: build update entries
*Apr 14 17:10:48.643: 192.168.1.160/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:10:48.647: 192.168.1.192/27 via 0.0.0.0, metric 1, tag 0
*Apr 14 17:10:48.647: 192.168.1.224/27 via 0.0.0.0, metric 2, tag 0

```

Antes do *router* R3 ser suspenso, o *router* R2 enviou informação pela *FastEthernet1/1* (ao R0) das rotas à esquerda de R2 inclusive a rota da interface que está situada na OSPF área 0 de R3. Depois deste ser suspenso essa rota deixou de ser informada pela interface *FastEthernet1/1* como podemos ver no *debug* do *router* R2.

b. O mecanismo de *route poisoning* serve para prevenir *loops* de encaminhamento causados por atualizações inconsistentes. Este consiste em não usar caminhos cujo *hop count* (contagem de saltos até ao destino) aumenta, o que pode acontecer quando se forma um *loop*. Se isto acontecer, a rota deixa de ser usada até que uma posterior atualização confirme um novo *hop count*. Basicamente, se uma rede falha, o *router* mais próximo insere na sua tabela o valor de *hop count* de 16 (valor máximo permitido), ficando a rota inatingível. Assim, o protocolo de encaminhamento informa todos os *routers* ligados na rede que uma determinada rota é inválida. Este mecanismo acelera, assim, o processo de convergências.

c.

- *invalid timer* - especifica quanto tempo uma rota pode estar na tabela de encaminhamento sem ser atualizada. O valor típico é de 180 segundos.
- *holddown timer* - quando um destino é dado como inatingível, o *router* deixa de "acreditar" em anúncios desse destino durante um certo tempo. Ou seja, controla o tempo entre que uma rota é considerada invalidada ou marcado como inacessível e a sua remoção da tabela de encaminhamento. O valor típico é de 180 segundos.
- *flush timer* - conta o tempo desde que foi recebida o último anúncio válido para uma rota até ela ser removida da tabela de encaminhamento. O valor típico é de 240 segundos.

8.

```
R10#ping 192.168.1.162
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 192.168.1.162, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/68/92 ms
```

Em cada *router*, R1, R2 e R3, altera-se a configuração do RIP para usar a versão 1 do protocolo:

```
version 1
```

```
R10#ping 192.168.1.162
```

```
PING 192.168.1.162 (192.168.1.162) 56(84) bytes of data.
```

```
^C
```

```
--- 192.168.1.162 ping statistics ---
```

```
29 packets transmitted, 0 received, 100% packet loss, time
```

Das principais diferenças entre a versão 1 e 2 do protocolo RIP é que a primeira versão é *classfull*, ou seja, suporta apenas classes A, B ou C ou subredes com a mesma máscara e troca atualizações de encaminhamento via *broadcast*. Já a segunda versão suporta *classless* e VLSM (divisão de subredes com várias máscaras de subrede) e troca informações através de *multicast*. Vamos focar-nos na segunda característica de cada versão, o primeiro ping, RIP com versão 2, é bem sucedido pois suporta divisão de subredes com máscaras diferentes (na rota que queremos percorrer temos duas máscaras diferentes 255.255.255.128 e 255.255.255.224) o que não é suportado pela versão 1 do protocolo e por essa razão o segundo ping, RIP com versão 1, não é bem sucedido.

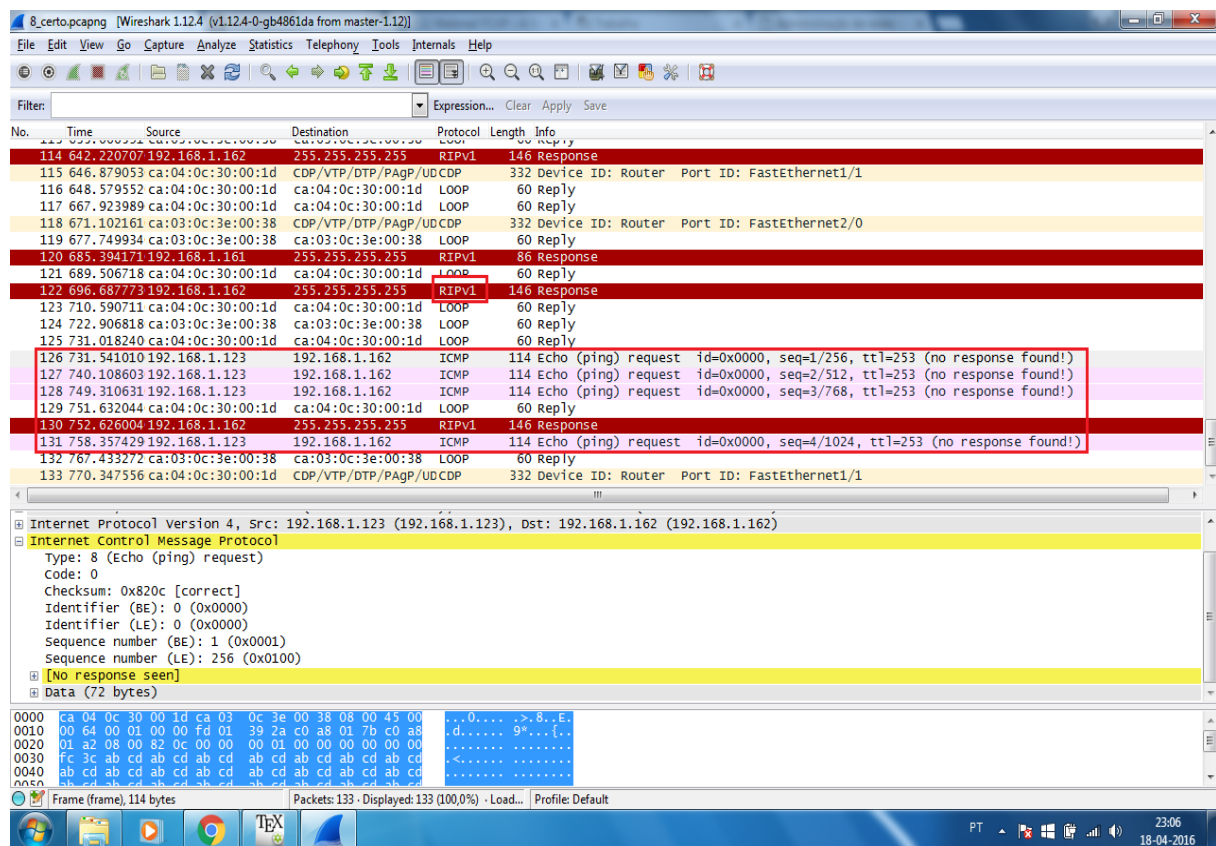


Figura 8: Captura na interface F1/1 de R3.