ADDETC – Área Departamental de Engenharia Eletrónica e Telecomunicações e de Computadores

LEIM -Licenciatura Engenharia informática e multimédia

Redes de Internet

Trabalho prático 3

Turma:

LEIM-51D

Trabalho realizado por:

Miguel Távora Nº45102

Carina Fernandes Nº45118

Pedro Henriques Nº45415

Docente:

Vítor Almeida

Data:21/02/2021

Índice

Índice ilustrações

1. Introdução

Esta última parte do trabalho diz respeito à implementação do protocolo BGP na topologia fornecida. O protocolo BGP aplica-se ao encaminhamento entre sistemas autónomos. Este protocolo é do tipo *Path Vector*, isto é, a informação sobre os caminhos ou rotas é atualizada, mediante o envio e receção de mensagens de controlo (com os atributos). O BGP divide-se em eBGP e iBGP. O primeiro destina-se à troca de informação entre *peers* de sistemas autónomos diferentes. Já o segundo diz respeito à troca de informação entre *peers* ou vizinhos de um mesmo AS, e tem como principal objetivo dar a conhecer aos routers "iBGP", as rotas aprendidas pelos *routers* de fronteira (que estabelecem relações eBGP).

Neste protocolo, cada *router* envia aos seus vizinhos (previamente configurados, uma vez que o BGP não os encontra automaticamente), mensagens de controlo com informação acerca de que redes consegue alcançar. Ao receber essas mensagens de controlo, os *routers*, verificam se a informação (rotas) recebida está em concordância com a sua política. Em caso afirmativo, alteram a sua tabela BGP (caso as rotas paras as redes em questão estejam presentes na tabela de *routing*).

O trabalho divide-se em 5 tarefas: Endereçamento; Configuração dos protocolos IGP, rotas estáticas e redistribuição de rotas; Redistribuição de rotas entre o OSPF e BGP; BGPv4 avançado; Realização de 3 exercícios à escolha.

Neste sentido, o principal objetivo deste trabalho, é resolver os exercícios propostos nas tarefas, fazendo uso dos conteúdos abordados no decurso das aulas teóricas e práticas.

2. Desenvolvimento

Fase 1 – Endereçamento

1 - Atribuição de endereços IPv4

a) Sim faz sentido, visto que pela atribuição de diferentes blocos IP para diferentes áreas cada área fica mais isolada das restantes em termos de endereços IP e assim há a garantia que não existem blocos do endereço IP sobrepostos.

b) Como os endereços IPv4 são limitados a distribuição de endereços por área leva ao mau aproveitamento dos endereços IP disponíveis, e ficam muitos endereços por ser utilizados.

c) Para prevenir que o controlo das configurações dos *routers* do ISP não pudesse ser realizado fora do AS, todos os endereços dos equipamentos teriam de ser endereços privados. Desta forma, uma pessoa de fora não poderia aceder diretamente a estes.

A atribuição de endereços na tabela de EXCEL, encontra-se correta sem quaisquer tipo de erros identificados por nós.

Os blocos de endereços utilizados pelo ISP são: 30.0.0.0/14 e 60.0.0.0/19 e 194.14.56.0/22. O segundo e terceiro blocos devem ser atribuídos apenas aos clientes.

Sendo que o bloco 60.0.0.0/19 se destina aos clientes, e já foram utilizados os blocos 60.0.26.0/23 (cliente 3), 60.0.28.0/23 (cliente 2), 60.0.31.128/25 (cliente 1), a tabela representativa de endereços IPv4 utilizados e livres figura abaixo:

Nome	**Rede**	**Nº endereços**	**Nº de dispositivos**	**Tipo de endereço**
Bloco livre 1	60.0.0.0/20	4096	4094	Público
Bloco livre 2	60.0.16.0/21	2048	2046	Público
Bloco livre 3	60.0.24.0/23	512	510	Público
Cliente 3	60.0.26.0/23	512	510	Público
Cliente 2	60.0.28.0/23	512	510	Público
Bloco livre 4	60.0.30.0/24	256	254	Público
Bloco livre 5	60.0.31.0/25	12/8	126	Público

Cliente 1	60.0.31.128/25	128	126	Público
Cliente 4	194.14.56.0/22	1024	1022	Público

Tabela 1 - Endereços atribuídos e ainda não atribuídos pelo ISP aos clientes.

Quanto ao bloco 30.0.0.0/14, este é utilizado nas ligações ponto a ponto e divide-se pelas 3 áreas do ISP:

Nome	Rede	Nº endereços	Nº de dispositivos	Tipo de endereço
Área 0	30.0.0.0/16	65 536	65 534	Público
Área 1	30.1.0.0/16	65 536	65 534	Público
Área 3	30.3.0.0/16	65 536	65 534	Público

Tabela 2 - Endereços públicos utilizados nas LANs do ISP e *loopbacks* dos *routers* PEn.

Finalmente, o bloco de endereços IP privado é utilizado pelo ISP por motivos de segurança, como foi referido anteriormente, e encontra-se distribuído da seguinte forma:

Nome	Rede	Nº endereços	Nº de dispositivos	Tipo de endereço
Área 0	10.0.0.0/16	65 536	65 534	Privado
Área 1	10.1.0.0/16	65 536	65 534	Privado
Área 3	10.3.0.0/16	65 536	65 534	Privado

Tabela 3 - Endereços privados utilizados pelo ISP.

d) Os endereços privados possuem algumas vantagens em relação a endereços públicos, nomeadamente não gastar endereços que poderiam ser utilizados para posteriormente expandir a rede. Além disso, aumentam a segurança da rede contra tráfego ilícito e impedem acessos indevidos.

   Contudo, é necessário que alguns dos dispositivos possuam endereços públicos, nomeadamente, os endereços de *Loopback* nos *routers*, que permitem a comunicação entre AS distintos e para *debug* de tráfego.

Fase 2 - OSPFv2, RIPv2, rotas estáticas e redistribuição de rotas

Parte 1

a) Como todas as ligações existentes dentro do AS 302 entre *routers* são ponto a ponto no OSPF, não existe necessidade da eleição de DRs e BDRs para os segmentos. A partir da configuração descrita abaixo verifica-se que não existe eleição de DR nem BDR nos segmentos de rede.

Perante os *pings* múltiplos, verifica-se que existe conectividade entre todos os *routers*.

```
PE1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.0.255.1        0   FULL/  -        00:00:32    10.1.20.2       GigabitEthernet3/0
10.0.255.3        0   FULL/  -        00:00:33    10.1.19.2       FastEthernet1/0
30.1.255.3        0   FULL/  -        00:00:32    10.1.17.2       GigabitEthernet4/0
```

```
PE1(tcl)#foreach address {
+>10.26.0.1
+>10.1.22.1
+>10.0.241.2
+>10.1.23.2
+>10.1.17.2
+>10.0.35.2
+>10.0.35.1
+>10.1.38.2
+>10.0.255.6
+>} {ping $address repeat 4 size 1500}

Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.26.0.1, timeout is 2 seconds:
....
Success rate is 0 percent (0/4)
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.1.22.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 72/81/84 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.241.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 44/51/60 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.1.23.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 16/43/68 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.1.17.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 16/20/24 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.35.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 52/75/96 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.35.1, timeout is 2 seconds:
!!!!
```

```
Success rate is 100 percent (4/4), round-trip min/avg/max = 52/75/96 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.35.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 16/27/32 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.1.38.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 80/85/88 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.255.6, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 84/92/96 ms
```

```
PE1#ping 10.0.26.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.26.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/10/12 ms
```

Perante os comandos em baixo é possível verificar que ambos os *routers* possuem uma largura de banda de referência igual, caso este valor não esteja igual em todos os *routers* OSPF não existirá troca de pacotes OSPF entre eles.

```
router ospf 1
 log-adjacency-changes
 auto-cost reference-bandwidth 10000
```

Figura 1 - comando show run do router PE2

```
!
router ospf 1
 log-adjacency-changes
 auto-cost reference-bandwidth 10000
```

Figura 2 - comando show run do router P6

Como a área 3 possui comunicação com *routers* fora do domínio OSPF, e para não possuir uma tabela de *routing* muito grande, foi definido que a área 3 é NSSA. Este tipo de área gera LSA tipo 7, que envia para as outras áreas e o ABR que conecta à área 0, converte os LSA tipo 7 em tipo 5 e propaga essa rota para as restantes áreas.

```
router ospf 1
 log-adjacency-changes
 auto-cost reference-bandwidth 10000
 area 3 nssa
```

Figura 3 - comando para área 3 como NSSA

```
PE7#ping
Protocol [ip]:
Target IP address: 10.1.20.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 10.0.49.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.20.1, timeout is 2 seconds:
Packet sent with a source address of 10.0.49.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 108/116/124 ms
PE7#
```

Figura 4 - Ping a partir de interface publica

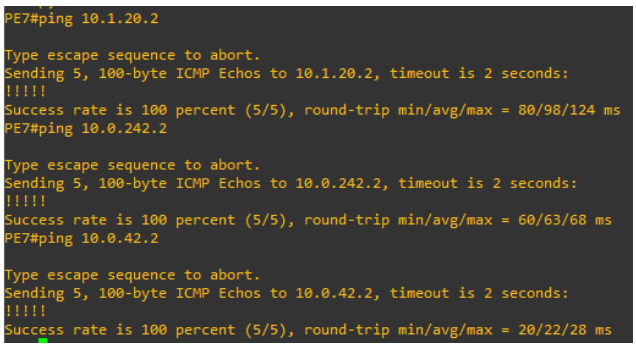

Figura 5 - Ping a todos os PC a partir de PC302_1

b) Como a área 3 possui LSA tipo 1, 3 e também tipo 7 é uma área do tipo NSSA.

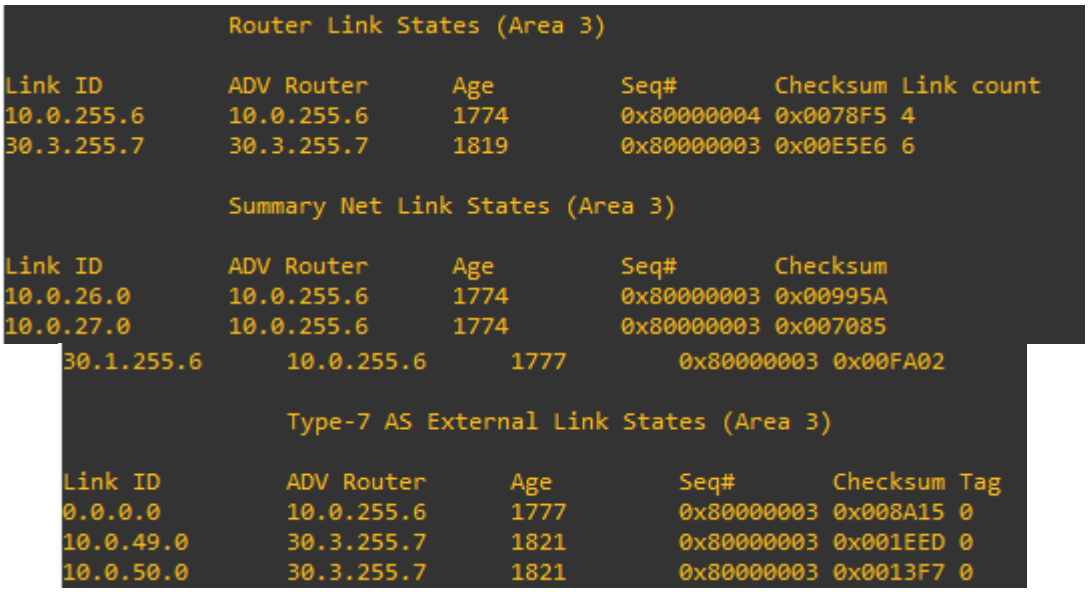

Figura 6 - LSA database Router PE7 (área 3)

c) O processo de eleição de rotas no OSPF para o mesmo destino é 1º rotas que apenas atravessam a própria área, 2º rotas que apenas atravessam o domínio OSPF do router independentemente das áreas que atravessem, 3º rotas externas tipo 1 (somando a métrica original com a métrica recebida), 4º rotas externas tipo 2 (apenas com a métrica recebida).

Assim sendo, tendo em conta que a rota pode ser paralela, assumimos também o débito igual da ligação, impossibilitando a escolha através da métrica. Portanto, a solução encontrada, passaria por criar uma área que somente interligasse o *router* PE1 e também o *router* PE3 através da ligação preferida. Esta solução não é intuitiva, mas consegue garantir o envio de tráfego pela rota pretendida, independentemente da existência de outras rotas.

d) Por causa do comando do custo da interface o custo passa a ser 6 em vez de utilizar o cálculo normal de largura banda referencia /custo da ligação.

```
interface GigabitEthernet4/0
 ip address 10.1.17.2 255.255.255.252
 ip ospf network point-to-point
 ip ospf cost 5
 negotiation auto
!
```

Figura 7 - Configuração da interface com custo alterado

e) O OSPF efetua balanceamento de tráfego de forma automática, se ambos os caminhos possuírem a mesma métrica para a mesma rota. Desta forma seria necessário alterar o custo/métrica do OSPF de uma das rotas, igualando-a á outra.

Parte 2 Configuração de rotas estáticas de interligação entre domínios (ISP e alguns clientes)

O uso de rotas estáticas oferece certas vantagens e desvantagens em relação a rotas dinamicamente partilhadas por protocolos EGP e IGP (tais como BGP e OSPF), através do uso de rotas estáticas possuímos um grande controlo sobre o caminho tomado pelo tráfego, não sendo enviado tráfego por rotas iguais aprendidas por outros protocolos (menor distância administrativa). No entanto rotas estáticas não são flexíveis e são propensas a introdução de erros humanos.

```
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 30.3.48.2
!
```

Figura 8 – Exemplo de utilização de rotas estáticas

A rota estática presente na figura 8 foi utilizada somente no cliente 1. Esta rota estática é a rota por omissão que envia para a interface conectada com o router PE7. Não foram utilizadas mais rotas estáticas pois não permite o encaminhamento dinâmico de tráfego e dificulta o processo de desenvolvimento do trabalho.

Parte 3 - Configuração do protocolo RIPv2

a) Através da configuração de interfaces passivas os protocolos não encaminham, e em alguns casos o *router* não interpreta caso recebam, mensagens com rotas partilhadas através daquele protocolo. Ou seja, a partir da interface passiva, não são enviadas mensagens com atualização de rotas aprendidas dinamicamente através de protocolos de encaminhamento neste caso o RIPv2.

```
RC2_2#config t
Enter configuration commands, one per line.  End with CNTL/Z.
RC2_2(config)#router rip
RC2_2(config-router)#passive-interface fa0/0
RC2_2(config-router)#exit
```

Figura 9-Configuração de interface passiva

b) Não, porque o protocolo RIPv1 é *classful*, isto é, não suporta máscaras. Como o OSPF utilizada máscaras não seria compatível a redistribuição entre estes dois protocolos.

Não é necessário incluir todas as rotas para o exterior, porque é possível injetar uma rota por omissão no protocolo RIP através do comando *default-information originate* no *router* PE7.

Desta forma, sempre que exista tráfego para um endereço que não seja conhecido este tráfego é entregue ao *router* com esta configuração, que assumimos irá saber entregar devidamente, ou reencaminhar para a rede de último recurso que conheça.

```
router rip
 version 2
 passive-interface default
 no passive-interface GigabitEthernet5/0
 network 10.0.0.0
 network 60.0.0.0
 default-information originate
 no auto-summary
!
```

```
Gateway of last resort is 10.0.49.1 to network 0.0.0.0

     10.0.0.0/30 is subnetted, 4 subnets
C       10.0.50.0 is directly connected, GigabitEthernet3/0
C       10.0.49.0 is directly connected, GigabitEthernet5/0
R       10.3.145.0 [120/1] via 10.0.49.1, 00:00:22, GigabitEthernet5/0
R       10.3.245.0 [120/1] via 10.0.49.1, 00:00:22, GigabitEthernet5/0
     60.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
R       60.0.28.0/24 [120/1] via 10.0.50.2, 00:00:15, GigabitEthernet3/0
C       60.0.29.251/32 is directly connected, Loopback0
R       60.0.29.252/32 [120/1] via 10.0.50.2, 00:00:15, GigabitEthernet3/0
R*   0.0.0.0/0 [120/1] via 10.0.49.1, 00:00:22, GigabitEthernet5/0
RC2_1#
```

Figura 10-Tabela de routing com entradas RIP

```
PC302_3> ping 10.3.245.1
84 bytes from 10.3.245.1 icmp_seq=1 ttl=252 time=53.285 ms
84 bytes from 10.3.245.1 icmp_seq=2 ttl=252 time=66.078 ms
84 bytes from 10.3.245.1 icmp_seq=3 ttl=252 time=70.164 ms
84 bytes from 10.3.245.1 icmp_seq=4 ttl=252 time=75.046 ms
84 bytes from 10.3.245.1 icmp_seq=5 ttl=252 time=57.342 ms

PC302_3>
```

Figura 11 - ping PC cliente 2 para o router PC6

Parte 4 - Redistribuição de rotas no AS do ISP entre os protocolos de routing IGP

a) Como mencionado previamente neste relatório, as rotas do tipo E são rotas injetadas no OSPF após aprendizagem por outros protocolos. A diferença entre as rotas tipo 1 e 2 reside no facto de ao custo redistribuído ser ou não somado o custo interno ao AS.

   Neste trabalho optamos por utilizar sobretudo redistribuição do tipo E2 pois desta forma a métrica é apenas aquela recebida de outros protocolos, nomeadamente BGP. Desta forma, somos capazes de encaminhar o tráfego através de políticas aplicadas no BGP e não tendo apenas em conta o melhor caminho possível.

```
O E2    11.101.0.0/21 [110/1] via 10.1.19.1, 04:11:04, FastEthernet1/0
     60.0.0.0/8 is variably subnetted, 5 subnets, 4 masks
O E2    60.0.28.0/24 [110/20] via 10.0.35.2, 04:11:41, GigabitEthernet2/0
O E2    60.0.0.0/19 [110/1] via 10.1.41.2, 04:11:41, FastEthernet0/0
O E2    60.0.31.128/25 [110/20] via 10.0.35.2, 04:11:41, GigabitEthernet2/0
O E2    60.0.29.251/32 [110/20] via 10.0.35.2, 04:11:41, GigabitEthernet2/0
O E2    60.0.29.252/32 [110/20] via 10.0.35.2, 04:11:41, GigabitEthernet2/0
```

Figura 12 - Tabela de routing com rotas do tipo escolhido

b) A redistribuição utilizada foi a redistribuição do RIP no OSPF, mas não no sentido oposto. Desta forma, garantimos que o OSPF conhece as rotas partilhadas por RIP e é capaz de encaminhar tráfego com destino ao cliente, que utiliza RIP. No entanto, não redistribuímos o OSPF, porque o cliente apenas precisa conhecer a rota por defeito para entregar o tráfego a um *router*, que corra OSPF e posteriormente BGP.

```
router ospf 1
 log-adjacency-changes
 auto-cost reference-bandwidth 10000
 area 3 nssa
 redistribute static subnets
 redistribute rip subnets
 passive-interface default
 no passive-interface FastEthernet1/1
 no passive-interface GigabitEthernet3/0
 network 10.3.49.0 0.0.0.3 area 3
 network 10.3.145.0 0.0.0.3 area 3
 network 10.3.245.0 0.0.0.3 area 3
 network 30.3.48.0 0.0.0.3 area 3
 network 30.3.255.7 0.0.0.0 area 3
!
```

```
router rip
 version 2
 passive-interface default
 no passive-interface GigabitEthernet5/0
 network 10.0.0.0
 network 60.0.0.0
 default-information originate
 no auto-summary
!
```

Figura 13 - Configurações OSPF e RIP onde se pode ver as regras de redistribuição

```
RC2_2#tclsh
RC2_2(tcl)#foreach address {
+>20.202.141.2
+>20.201.12.2
+>10.0.26.1
+>10.1.22.1
+>10.0.241.2
+>10.1.23.2
+>10.0.35.1
+>10.0.35.2
+>10.1.36.2
+>10.1.38.2
+>} {ping $address repeat 4 size 1500}

Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 20.202.141.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 176/185/208 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 20.201.12.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 176/195/212 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.26.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 148/150/152 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.1.22.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 176/182/196 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.241.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 144/152/168 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.1.23.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 144/149/156 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.35.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 148/150/152 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.0.35.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 116/122/124 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.1.36.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 148/151/156 ms
Type escape sequence to abort.
Sending 4, 1500-byte ICMP Echos to 10.1.38.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (4/4), round-trip min/avg/max = 116/119/124 ms
RC2_2(tcl)#
```

c) Sim, porque sem a redistribuição o OSPF não conseguiria encaminhar os pacotes com o endereço destino no cliente 1 ou 2. Isto porque possuem protocolos de routing distintos. Alem disso o modificador "subnets" em ambos os comandos deve estar presente para incluir a mascara de rede presente na rota redistribuida.

d) Não, pois, quando uma mensagem do cliente 1 fosse enviada para fora, os restantes *routers* não conseguiriam responder ao pedido pois não tinham forma de encaminhar o pacote de volta á origem, por o endereço ser da interface ser por onde o pacote saiu ser privado.

e) Não, porque criar uma rota estática do cliente 1 até ao PC do cliente 2 não é necessário porque o cliente 1 já possui uma rota estática para o PE7, como o PE7 consegue encaminhar para o cliente 2 não faz sentido existir essa rota estática.

f) Com a remoção do comando "ip ospf network point-to-point" a ligação entre os dois routers deixa de ser P2P para ser do tipo 2-WAY e desta forma passa a existir um *designated router* para o segmento, mesmo que seja uma ligação entre apenas dois routers, levando a envio de tráfego de controlo OSPF desnecessariamente.

g) Não, porque existem muitas redes que seria possível agregar nomeadamente a rede 20.202.0.0 e também a rede 10.0.0.0. Desta forma fica mais simples a tabela de routing, no entanto anunciamos um bloco de IP com mais endereços do que aqueles correspondentes ás redes originais.

h) Se não existir o comando o router de fronteira da área (ABR) não gera uma rota por omissão para a área NSSA. Desta forma não é possível tráfego relativo aos clientes 1 ou 2 serem encaminhados do PE7 para o P6.

i) Não, porque desta maneira a área não poderia possuir ASBR e perder-se-ia a comunicação entre o ISP e os clientes 1 e 2.

Fase 3 – BGPv4, redistribuição de rotas entre o OSPFv2 e o BG

Parte 1 - BGPv4 básico

a) O endereço IP de origem de uma mensagem BGP é o endereço da interface do *router* origem. Se o *router* possuir uma interface de *loopback*, esta será eleita a interface *loopback,* em detrimento do endereço da interface física. Utilizando uma interface *loopback,* previne-se que uma interface quando perde a conexão, a sessão se perca entre os dois routers BGP.

   O endereço IP de origem é o da interface por onde sai a mensagem BGP, a não ser que exista uma indicação explicita por parte do administrador de rede para que o endereço de origem seja outro, normalmente uma interface pública tal como uma de *loopback*.

b) Caso a rota não exista na tabela de *routing* ou a rota recebida possua uma métrica melhor que a está contida, a tabela de *routing* é atualizada. Assim sendo, se a mensagem possuir um next-hop, que não está na sua tabela de *routing*, a segunda é atualizada, ou não, dependendo dos pontos previamente indicados.

c) Caso não seja utilizado este comando, por omissão, o *router* BGP aplica sempre os comandos apenas e somente a redes IPv4 *unicast*.

   Neste trabalho não é necessário utilizar o comando porque apenas utilizamos o protocolo de rede IPv4 *unicast*. Numa rede real, isto pode não ser assim porque pode haver redes que utilizem outros protocolos de rede.

d) Não, porque por para encaminhar do tráfego não precisa ser feito somente por routers BGP. Desta forma o BGP confia nos protocolos de encaminhamento internos (IGP) para encaminhar o tráfego interno dentro da área, através de routers que não correm BGP.

```
R303_1#ping
Protocol [ip]:
Target IP address: 194.14.59.251
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 40.0.3.251
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 194.14.59.251, timeout is 2 seconds:
Packet sent with a source address of 40.0.3.251
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/32/40 ms
R303_1#ping
Protocol [ip]:
Target IP address: 30.1.255.6
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 40.0.3.251
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 30.1.255.6, timeout is 2 seconds:
Packet sent with a source address of 40.0.3.251
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 116/125/136 ms
R303_1#
```

Figura 14 - Ping do router AS303 para AS301 e AS302

e) Não, porque os routers que correm BGP estão em *tiers* diferentes, nomeadamente 1, 2 e 3. Desta forma cada AS anuncia o mínimo de rotas necessário para garantir que anuncia todos os blocos que lhe competem, e não anuncia redes especificas no seu interior.

As redes 4.4.4.4 e 8.8.8.8 são casos especiais pois não são explicitamente anunciados por nenhum AS, sendo entregue sempre ao vizinho de último recurso disponível para cada AS, sendo que o AS 102 e 101 irão depois entregar a um outro router (imaginário) que fará o tráfego chegar ao destino.

f) Os endereços utilizados dentro de cada AS são privados. Assim sendo, mesmo que um *ping* para um endereço privado fosse enviado e recebido com sucesso no seu destino, o destino não conseguiria reencaminhar o tráfego de volta.

g) Nas ligações ponto-a-ponto entre dois routers dentro de um AS não existe problema visto que não há mais nenhum equipamento ligado e desta forma é possível o encaminhamento do tráfego.

   Entre AS não se pode utilizar visto que entre AS não se conhece os endereços dentro do AS e deixaria de haver conexão entre os AS pelo motivo mencionada na pergunta anterior.

h) Visto que não existe nenhuma resolução entre endereços públicos para endereços privados o endereço quando tenta ir para fora do AS302 não consegue porque o BGP não consegue encaminhar tráfego para um destino a partir de um endereço privado. O tráfego segue o caminho: P6 – PE6 – PE5 – PE4 – PE3 – R202_3 – R102_2. Utilizando um endereço público é possível comunicar dentro da área até á internet, caso fosse privado não seria possível.

   Atraves da alteração do endereço de origem é possivel realizar o trace e verificar que não existe nenhuma surpresa significativa, sendo apenas notavel a escolha do router PE3 em prol do PE1 sendo que á partida ambos conseguerian entregar o trafego ao AS 201.

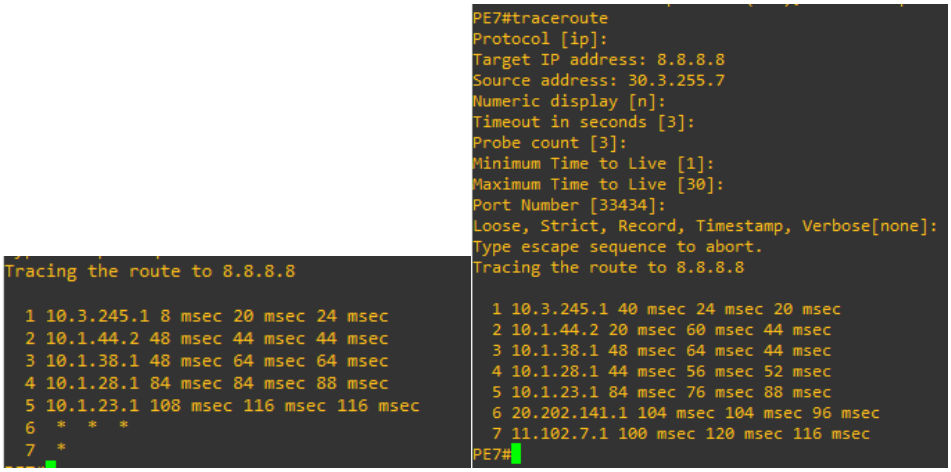

Figura 15 - Traceroute para 8.8.8.8 com endereço privado (esquerda) e publico (direita)

i) Como o endereço do PC dentro do cliente 2 possui um endereço público é possível comunicar para a internet e o caminho percorrido é o mesmo de anteriormente.

```
PC302_3> trace 4.4.4.4
trace to 4.4.4.4, 8 hops max, press Ctrl+C to stop
 1   60.0.28.252   10.842 ms  10.123 ms  9.826 ms
 2   10.0.50.1   31.668 ms  31.797 ms  31.979 ms
 3   10.0.49.1   41.083 ms  54.737 ms  31.054 ms
 4   10.3.245.1   62.926 ms  73.085 ms  72.942 ms
 5   10.1.44.2   85.224 ms  86.735 ms  96.428 ms
 6   10.1.38.1   116.727 ms  116.768 ms  107.329 ms
 7   10.1.28.1   137.730 ms  128.675 ms  139.337 ms
 8   10.1.23.1   160.478 ms  172.744 ms  171.673 ms
```

j) A rota é R101_1 – R102_1 – R102_2

```
PC101_1> trace 8.8.8.8
trace to 8.8.8.8, 8 hops max, press Ctrl+C to stop
 1   11.101.1.251   3.222 ms  9.168 ms  10.294 ms
 2   11.101.2.2   30.334 ms  29.410 ms  31.301 ms
 3   *10.102.3.2   52.123 ms (ICMP type:3, code:3, Destination port unreachable)
```

```
PC101_1> ping 8.8.8.8
84 bytes from 8.8.8.8 icmp_seq=1 ttl=253 time=51.973 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=253 time=52.145 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=253 time=44.033 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=253 time=48.296 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=253 time=47.960 ms
```

k) Os AS do mesmo nível ou de níveis acima não podem servir de trânsito para tráfego. Por isso não é possível o encaminhamento do ping entre o PC101_1 e o router R202_3.

```
PC101_1> trace 20.202.255.253
trace to 20.202.255.253, 8 hops max, press Ctrl+C to stop
 1   11.101.1.251   1.089 ms  10.326 ms  9.266 ms
 2   11.101.4.2   30.296 ms  30.154 ms  31.176 ms
 3   20.201.12.2   52.151 ms  51.993 ms  53.246 ms
 4   20.202.131.1   71.920 ms  74.018 ms  75.362 ms
 5   *20.202.131.1   64.082 ms (ICMP type:3, code:1, Destination host unreachable)
```

l) Como não existe ligação entre o AS202 e o AS201 não existe forma de o router R201_1 comunicar com o router R202_3. Isto devido a não ser possível utilizar *tiers* de outros níveis para fazer de AS de trânsito.

```
R201_1#traceroute 20.202.255.253

Type escape sequence to abort.
Tracing the route to 20.202.255.253

  1 20.201.12.2 16 msec 20 msec 24 msec
  2 20.202.131.1 [AS 202] 28 msec 32 msec 44 msec
  3 20.202.131.1 [AS 202] !H  !H  !H
```

m) Este comando serve para eliminar tráfego que não pode ser entregue com as rotas existentes na tabela de routing de um router. Ou seja, se esta rota em específico não estiver na tabela de routing do router (não contando com a rota 0/0), os pacotes serão eliminados.

Desta forma bloqueamos tráfego a circular que pode não possuir destino no interior do AS, apesar do endereço ser anunciado pelo router. Por exemplo assumindo que o AS XXX possui 3 blocos de IP 30.4.0.0/16, 30.6.0.0/16 e 30.8.0.0/16 e decide anunciar apenas a super-rede 30.0.0.0/12. Um AS vizinho deste AS possui um pacote com destino para 30.0.0.1 e decide entregar ao router do AS XXX. Este pacote, apesar de estar contido no conjunto de endereços anunciado não pode ser entregue dentro do AS pelo que este pacote deve ser eliminado.

n) Atualiza de acordo com a interface *loopback* porque mesmo que a comunicação entre os routers "morra" nunca é retirada a entrada da tabela de routing e não é recalculada nova topologia.

```
R102_2#show ip int brief
Interface                  IP-Address
FastEthernet0/0            unassigned
FastEthernet0/1            unassigned
FastEthernet1/0            unassigned
FastEthernet1/1            unassigned
GigabitEthernet2/0         11.102.7.1
GigabitEthernet3/0         11.102.6.1
GigabitEthernet4/0         10.102.3.2
GigabitEthernet5/0         unassigned
GigabitEthernet6/0         unassigned
Loopback0                  11.102.3.252
Loopback1                  8.8.8.8
```

```
router bgp 102
 no synchronization
 bgp log-neighbor-changes
 network 8.8.8.8 mask 255.255.255.255
 network 11.102.0.0 mask 255.255.252.0
 neighbor 11.102.3.251 remote-as 102
 neighbor 11.102.3.251 update-source Loopback0
 neighbor 11.102.3.251 next-hop-self
 neighbor 11.102.6.2 remote-as 202
 neighbor 11.102.7.2 remote-as 202
 no auto-summary
```

o) O comando "update-source" serve para atualizar a valor da vizinhança, mas em vez de utilizar a interface por onde recebeu a tabela será pelo endereço da interface *loopback*. O comando "next-hop-self" avisa o vizinho para enviar os pacotes para

ele quando não conhece o router anunciante. Perante os comandos ele avisa a vizinhança que para enviar para ele deve enviar para a interface *loopback* e que para quando não conhece o router que enviou deve enviar para ele.

```
!
router bgp 102
 no synchronization
 bgp log-neighbor-changes
 network 11.102.0.0 mask 255.255.252.0
 neighbor 11.101.2.1 remote-as 101
 neighbor 11.102.3.252 remote-as 102
 neighbor 11.102.3.252 update-source Loopback0
 neighbor 11.102.3.252 next-hop-self
 neighbor 11.102.5.2 remote-as 201
 no auto-summary
!
```

p) Não, porque eles não utilizam o protocolo OSPF para comunicar entre eles (pelo menos no contexto do trabalho) e sim o BGP. Desta forma, não será necessário utilizar o OSPF dentro dos AS.

q) O comando "no *synchronization*", permite que o *router* não sincronize o iBGP com outros routers que usam o protocolo de *routing* interno como o OSPF. Quando dois BGP *speakers* tentam comunicar, se existir um router que não fale BGP pode haver *loops*. Para prevenir essa situação, utiliza-se "*synchronization"*.

Desta forma, não se realiza nos *routers* existentes "*synchronization*" e redistrubuição entre o OSPF e o BGP por realizarem as duas a mesma funcionalidade podendo criar *loops*.

```
router ospf 1
 log-adjacency-changes
 auto-cost reference-bandwidth 10000
 redistribute bgp 302 subnets
 passive-interface default
 no passive-interface FastEthernet1/0
 no passive-interface GigabitEthernet3/0
 no passive-interface GigabitEthernet4/0
 network 10.1.17.0 0.0.0.3 area 1
 network 10.1.19.0 0.0.0.3 area 1
 network 10.1.20.0 0.0.0.3 area 1
 network 30.1.255.1 0.0.0.0 area 1
 default-information originate
!
```

```
PC65005_1> ping 11.101.7.251
84 bytes from 11.101.7.251 icmp_seq=1 ttl=250 time=118.571 ms
84 bytes from 11.101.7.251 icmp_seq=2 ttl=250 time=108.212 ms
84 bytes from 11.101.7.251 icmp_seq=3 ttl=250 time=105.202 ms
84 bytes from 11.101.7.251 icmp_seq=4 ttl=250 time=112.431 ms
84 bytes from 11.101.7.251 icmp_seq=5 ttl=250 time=99.241 ms
```

Figura 16 - ping AS 101

```
PC65005_1> ping 20.201.15.251
84 bytes from 20.201.15.251 icmp_seq=1 ttl=251 time=79.851 ms
84 bytes from 20.201.15.251 icmp_seq=2 ttl=251 time=67.348 ms
84 bytes from 20.201.15.251 icmp_seq=3 ttl=251 time=103.004 ms
84 bytes from 20.201.15.251 icmp_seq=4 ttl=251 time=74.910 ms
84 bytes from 20.201.15.251 icmp_seq=5 ttl=251 time=65.198 ms
```

Figura 17 - ping AS 201

```
PC65005_1> ping 194.14.59.251
84 bytes from 194.14.59.251 icmp_seq=1 ttl=252 time=83.221 ms
84 bytes from 194.14.59.251 icmp_seq=2 ttl=252 time=85.410 ms
84 bytes from 194.14.59.251 icmp_seq=3 ttl=252 time=85.819 ms
84 bytes from 194.14.59.251 icmp_seq=4 ttl=252 time=81.179 ms
84 bytes from 194.14.59.251 icmp_seq=5 ttl=252 time=80.058 ms
```

Figura 18 - ping AS 301

```
PC65005_1> ping 30.1.255.1
84 bytes from 30.1.255.1 icmp_seq=1 ttl=252 time=82.057 ms
84 bytes from 30.1.255.1 icmp_seq=2 ttl=252 time=71.840 ms
84 bytes from 30.1.255.1 icmp_seq=3 ttl=252 time=85.280 ms
84 bytes from 30.1.255.1 icmp_seq=4 ttl=252 time=83.222 ms
84 bytes from 30.1.255.1 icmp_seq=5 ttl=252 time=79.885 ms
```

Figura 19 - ping AS 302

```
R301_1#show ip bgp
BGP table version is 12, local router ID is 194.14.59.251
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  4.4.4.4/32       30.1.254.132                           0 303 302 201 101 i
*                   30.1.212.2                             0 302 201 101 i
*>                  30.1.254.132                           0 302 201 101 i
*  8.8.8.8/32       30.1.254.132                           0 302 202 102 i
*                   30.1.254.133                           0 303 202 102 i
*>                  30.1.212.2                             0 302 202 102 i
*  11.101.0.0/21    30.1.254.132                           0 303 302 201 101 i
*                   30.1.212.2                             0 302 201 101 i
*>                  30.1.254.132                           0 302 201 101 i
*  11.102.0.0/22    30.1.254.132                           0 302 202 102 i
*                   30.1.254.133                           0 303 202 102 i
*>                  30.1.212.2                             0 302 202 102 i
*  20.201.0.0/20    30.1.254.132                           0 303 302 201 i
*                   30.1.212.2                             0 302 201 i
*>                  30.1.254.132                           0 302 201 i
*  20.202.0.0/16    30.1.254.132                           0 303 302 202 i
*                   30.1.254.132                           0 302 202 i
*>                  30.1.212.2                             0 302 202 i
*  30.0.0.0/14      30.1.254.132                           0 303 302 i
*>                  30.1.212.2               0             0 302 i
*                   30.1.254.132             1             0 302 i
*  40.0.0.0/22      30.1.212.2                             0 302 303 i
*                   30.1.254.133                           0 302 303 i
*>                  30.1.254.133             1             0 303 i
*  60.0.0.0/19      30.1.254.132                           0 303 302 i
*>                  30.1.212.2               0             0 302 i
*                   30.1.254.132             1             0 302 i
*  60.0.26.0/23     30.1.254.132                           0 303 302 i
*                   30.1.254.132                           0 302 i
*>                  30.1.212.2                             0 302 i
*> 194.14.56.0/22   0.0.0.0                  0         32768 i
R301_1#
```

Figura 20 - tabela BGP

```
Gateway of last resort is not set

     4.0.0.0/32 is subnetted, 1 subnets
B       4.4.4.4 [20/0] via 30.1.254.132, 02:52:12
     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
B       20.201.0.0/20 [20/0] via 30.1.254.132, 02:52:36
B       20.202.0.0/16 [20/0] via 30.1.212.2, 02:52:36
     194.14.59.0/32 is subnetted, 1 subnets
C       194.14.59.251 is directly connected, Loopback0
     8.0.0.0/32 is subnetted, 1 subnets
B       8.8.8.8 [20/0] via 30.1.212.2, 02:52:36
     40.0.0.0/22 is subnetted, 1 subnets
B       40.0.0.0 [20/1] via 30.1.254.133, 02:52:35
     11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
B       11.102.0.0/22 [20/0] via 30.1.212.2, 02:52:36
B       11.101.0.0/21 [20/0] via 30.1.254.132, 02:52:13
     60.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
B       60.0.26.0/23 [20/0] via 30.1.212.2, 02:52:38
B       60.0.0.0/19 [20/0] via 30.1.212.2, 02:52:38
     30.0.0.0/8 is variably subnetted, 3 subnets, 3 masks
B       30.0.0.0/14 [20/0] via 30.1.212.2, 02:52:38
C       30.1.254.128/25 is directly connected, FastEthernet1/1
C       30.1.212.0/30 is directly connected, GigabitEthernet2/0
C    194.14.56.0/23 is directly connected, FastEthernet0/0
S    194.14.56.0/22 is directly connected, Null0
```

Figura 21 - tabela routing

Parte 2 Implementação de políticas no iBGP no ISP

Os *routers* PE1 e PE3 possuem full-mesh visto que nenhum dos *routers* possui um *route reflector* ou *confederation*.

a) Por omissão, um *router* BGP sem filtros, receberá todas as rotas BGP do mundo. Tanto no anúncio como na receção são incluídas todas as entradas recebidas das tabelas BGP. Este comportamento pode originar problemas, tais como a sobrecarga de endereços e consequentemente, a sobrecarga de memória e o maior tempo de processamento para encaminhar tráfego.

b) O timer hold é 180 segundos o keepalive é 60 segundos. O timer keepalive é o intervalo de tempo entre mensagens Keepalive e o timer hold é a quantidade de tempo que o *router* "aguenta" sem receber uma mensagem Keepalive. Caso não receba nenhuma mensagem hold ele termina a sessão BGP entre os *routers*. Estes *timers* são tão longos para não sobrecarregar a rede com mensagens de controlo. No entanto, não são maiores para poder reagir com alguma rapidez a mudanças na rede.

c) Existiria mais tráfego de controlo BGP a circular na rede, sobrecarregando a rede sem grandes benefícios.

d) Isto seria conseguido através da utilização do endereço *loopback (*comando update-source*)*, para quando uma interface falhasse existiria um caminho alternativo para o mesmo endereço conseguindo assim redundância das ligações.

Parte 3 - Políticas de eBGP, entre o ISP e os seus clientes

```
R301_1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R301_1(config)#router bgp 301
R301_1(config-router)#neighbor 30.1.254.132 maximum-prefix 50
R301_1(config-router)#neighbor 30.1.254.133 maximum-prefix 50
```

Figura 22 – Configuração do máximo de prefixos permitidos

```
ip prefix-list MATCH seq 5 permit 40.0.0.0/22
```

a) O router RC2_1 teria de ser configurado com BGP, anunciar as redes que possui e anunciar quais os vizinhos a que está ligado. Além disso, seria necessário configurar uma interface com um endereço público para receber tráfego de fora do AS.

b) O encaminhamento entre os AS, é feito através do AS 302. Como o AS 302 possui endereços privados no seu interior não é possível o *traceroute* indicar todo o percurso até ao destino. Isto deve-se ao facto da transição entre endereços privados para públicos, o traceroute deixa de conseguir indicar o caminho. Contudo existe comunicação entre eles visto que o *ping* funciona.

```
PC301_1> trace 60.0.26.1
trace to 60.0.26.1, 8 hops max, press Ctrl+C to stop
 1   194.14.57.251   8.194 ms  8.109 ms  10.902 ms
 2   30.1.254.132   31.113 ms  31.743 ms  30.924 ms
 3   10.1.22.2   41.970 ms  42.028 ms  42.003 ms
 4   10.1.25.2   64.068 ms  52.016 ms  54.092 ms
 5   10.1.29.5   64.029 ms  72.058 ms  73.955 ms
 6   *10.1.29.5   73.602 ms (ICMP type:3, code:1, Destination host unreachable)
```

Figura 23 - Traceroute PC cliente 4 para cliente 3

```
PC301_1> ping 60.0.26.1
84 bytes from 60.0.26.1 icmp_seq=1 ttl=53 time=141.801 ms
84 bytes from 60.0.26.1 icmp_seq=2 ttl=53 time=157.127 ms
84 bytes from 60.0.26.1 icmp_seq=3 ttl=53 time=151.480 ms
84 bytes from 60.0.26.1 icmp_seq=4 ttl=53 time=145.040 ms
84 bytes from 60.0.26.1 icmp_seq=5 ttl=53 time=159.922 ms
```

Figura 24 - Ping PC cliente 4 para o cliente 3

c) Sim, porque não é possível encaminhar tráfego para dentro de um AS privado e o AS302 e os seus clientes não servem de tráfego para outros AS.

d) Como o número de entradas na tabela BGP é proporcional ao número de AS a que está ligado, então possuir muitas ligações a muitos AS desencadearia uma sobrecarga na memória dos *routers* do ISP e para encaminhar demoraria muito tempo a encontrar qual a porta a partir da qual deve encaminhar.

```
router bgp 65005
 no synchronization
 bgp log-neighbor-changes
 network 60.0.26.0 mask 255.255.254.0
 neighbor 10.1.29.1 remote-as 302
 neighbor 10.1.29.3 remote-as 302
 no auto-summary
!
```

Parte 4 - Route Refletor (RR)

O comando “neighbor [address] route-reflector-client” serve para configurar um *router reflector* e configurar o vizinho especificado como o seu cliente. Para indicar que o vizinho não é um cliente utiliza-se o “no” antes do comando.

Para configurar o router P4 como route reflector, primeiro ativa-se o BGP no próprio router, utilizou-se uma interface *loopback*, depois faz-se *peer* com todos os outros routers iBGP presentes no AS e o OSPF também necessita de anunciar a interface *loopback*. Mesmo que não sejam seus clientes é estabelecida uma relação de *peer* iBGP para manter o *full-mesh*.

```
 neighbor 30.1.255.1 remote-as 302
 neighbor 30.1.255.2 remote-as 302
 neighbor 30.1.255.3 remote-as 302
 neighbor 30.1.255.4 remote-as 302
 neighbor 30.1.255.5 remote-as 302
 neighbor 30.1.255.6 remote-as 302
 !
```

Figura 25 - Vizinhança full mesh, router P4, antes de ser aplicado o RR

De seguida, definimos os *routers* pretendidos como clientes de R4

```
P4(config-router)#neighbor 30.1.255.2 route-reflector-client
P4(config-router)#neighbor 30.1.255.4 route-reflector-client
P4(config-router)#neighbor 30.1.255.5 route-reflector-client
P4(config-router)#neighbor 30.1.255.6 route-reflector-client
```

Figura 26 - Definição dos clientes

E eliminamos rotas noutros routers que não são mais necessárias. De notar que, apesar de apenas termos colocado o print do processo em 1 dos routers, este procedimento foi realizado em todos os routers excepto o P4.

```
PE1(config-router)#no neighbor 30.1.255.2
PE1(config-router)#no neighbor 30.1.255.4
PE1(config-router)#no neighbor 30.1.255.5
```

Figura 27-Remover o full mesh

```
P4#show ip bgp
BGP table version is 11, local router ID is 30.0.255.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - inte
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
r i30.0.0.0/14      10.1.23.2                1    100      0 i
r>i                 30.1.255.6               0    100      0 i
r i                 30.1.255.5               1    100      0 i
r i                 10.1.28.2                1    100      0 i
r i                 10.1.22.2                1    100      0 i
r i60.0.0.0/19      10.1.23.2                1    100      0 i
r>i                 30.1.255.6               0    100      0 i
r i                 30.1.255.5               1    100      0 i
r i                 10.1.28.2                1    100      0 i
r i                 10.1.22.2                1    100      0 i
P4#
```

Figura 28 - Tabela encaminhamento BGP após route-reflector

Fase 4 – BGPv4 avançado

Parte 1

Existe comunicação entre o AS 202 e 302.

```
router bgp 302
 bgp log-neighbor-changes
 neighbor 20.202.132.1 remote-as 202
 neighbor 20.202.141.1 remote-as 202
```

```
router bgp 302
 bgp log-neighbor-changes
 neighbor 20.201.12.1 remote-as 201
 neighbor 20.202.131.1 remote-as 202
```

```
router bgp 202
 bgp log-neighbor-changes
 neighbor 20.202.131.2 remote-as 302
 neighbor 20.202.132.2 remote-as 302
```

```
router bgp 303
 no synchronization
 bgp log-neighbor-changes
 network 40.0.0.0 mask 255.255.252.0
```

Não existe *routing* estático entre a área 202 e 302, as rotas existentes como *routing* estático são para o null0, que servem para ser adicionadas à tabela de *routing* quando a entrada existe na tabela BGP e não na tabela de *routing*.

```
ip forward-protocol nd
ip route 20.202.0.0 255.255.0.0 Null0
!
```

Figura 29 - Rota null0 do R202_1

```
R301_1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R301_1(config)#router bgp 301
R301_1(config-router)#neighbor 30.1.254.133 route-map NO-EXPORT in
R301_1(config-router)#exit
```

Figura 30 - Filtro de communities no-export

Parte 2

Para garantir que não são propagadas as rotas dentro do ISP e dos seus clientes, foram utilizados os route-maps onde se faz a atribuição com o valor no-export para o neighbor não pertencente ao ISP. Desta forma é garantido que não são propagadas rotas para o exterior internas ao ISP e os seus clientes.

```
address-family ipv4
 neighbor 20.201.12.1 activate
 neighbor 20.201.12.1 remove-private-as
 neighbor 20.201.12.1 route-map NO-EXPORT in
```

```
route-map NO-EXPORT permit 10
 set community no-export
!
```

Para garantir os filtros *anti-spoffing* entre os clientes e o ISP é introduzido uma *password* em ambos, de maneira a que mensagens que não possuam o mesmo valor sejam descartadas.

```
router bgp 65005
 no synchronization
 bgp log-neighbor-changes
 network 60.0.26.0 mask 255.255.254.0
 neighbor 10.1.29.1 password CISCO
 neighbor 10.1.29.3 password CISCO
 no auto-summary
!
```

O comando indica que são removidos os AS privados do atributo AS_PATH nos pacotes BGP que são enviados aos seus *neighbors* de *tiers* superiores.

```
address-family ipv4
 neighbor 20.201.12.1 activate
 neighbor 20.201.12.1 remove-private-as
 neighbor 20.201.12.1 route-map NO-EXPORT in
 neighbor 20.202.131.1 activate
 neighbor 20.202.131.1 remove-private-as
 neighbor 20.202.131.1 route-map NO-EXPORT in
```

Para garantir que não entram pacotes no seu AS com o mesmo endereço IP de origem, foi utilizado um *prefix-list* que permite negar o tráfego para dentro do AS com o mesmo endereço que o AS possui.

```
ip prefix-list deny-self seq 5 deny 30.0.0.0/14
ip prefix-list deny-self seq 10 deny 60.0.0.0/19
ip prefix-list deny-self seq 15 permit 0.0.0.0/0 le 24
no cdp log mismatch duplex
!
```

```
!
address-family ipv4
 neighbor upstream-group prefix-list deny-self in
```

Parte 3 - Políticas de tráfego de saída do ISP

Para ser mais favorável a utilização do AS 201 em detrimento do AS 202, foi utilizado o atributo LOCAL_PREF. Colocou-se um valor maior na saída do PE1 para o AS201, que fosse maior que o *default*. Em relação ao PE3 foi reduzido o LOCAL_PREF menor para o AS202.

```
address-family ipv4
 neighbor 20.201.12.1 activate
 neighbor 20.201.12.1 route-map local-pref in
```

Figura 31 - Configuração do PE1 para alterar o LOCAL_PREF

```
route-map local-pref permit 10
 match ip address prefix-list MATCH-ALL
 set local-preference 800
!
```

```
PE1#traceroute 8.8.8.8

Type escape sequence to abort.
Tracing the route to 8.8.8.8

  1 20.201.12.1 [AS 201] 12 msec 20 msec 24 msec
  2 11.102.5.1 20 msec 72 msec 24 msec
  3 10.102.3.2 68 msec 52 msec 56 msec
```

Figura 32 - Trace para 8.8.8.8 com LOCAL_PREF PE1

```
address-family ipv4
 neighbor 20.202.132.1 activate
 neighbor 20.202.132.1 remove-private-as
 neighbor 20.202.132.1 route-map LOCAL_PREF_50 in
 neighbor 20.202.141.1 activate
 neighbor 20.202.141.1 remove-private-as
 neighbor 20.202.141.1 route-map LOCAL_PREF_70 in
```

Figura 33 - Configuração LOCAL_PREF no router PE3

```
route-map LOCAL_PREF_50 permit 10
 set local-preference 50
!
route-map LOCAL_PREF_70 permit 10
 set local-preference 70
```

```
PE3#trace
Protocol [ip]:
Target IP address: 8.8.8.8
Source address: 30.1.255.3
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to 8.8.8.8

  1 10.1.17.1 32 msec 20 msec 24 msec
  2 20.201.12.1 48 msec 16 msec 44 msec
  3 11.102.5.1 76 msec 64 msec 56 msec
  4 10.102.3.2 64 msec 68 msec 64 msec
```

Figura 34 - Ping quando é possível enviar pelo AS201

```
PE3#trace
Protocol [ip]:
Target IP address: 8.8.8.8
Source address: 30.1.255.3
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to 8.8.8.8

  1 20.202.141.1 [AS 202] 44 msec 20 msec 20 msec
  2 11.102.7.1 36 msec 48 msec 24 msec
```

Figura 35 - Ping quando não é possível enviar para o AS201

a) Poder-se-ia efetuar a sumarização de rotas. No entanto, esta solução não é conveniente, pois perde-se precisão aquando do reenvio do tráfego, além da possibilidade da criação de tráfego para endereços sem redes atribuídas (necessário utilizar endereços com null0)

Tarefa 5

Parte 2- Nova saída de tráfego internacional

Inicialmente, foi criada uma ligação entre o *router* PE2 e o *router* R101_1. De seguida, foram atribuídos endereços das interfaces.

```
!
interface GigabitEthernet4/0
 ip address 11.101.5.2 255.255.255.252
 negotiation auto
!
```

```
interface GigabitEthernet4/0
 ip address 11.101.5.1 255.255.255.252
 negotiation auto
!
```

De seguida, foi feito AS_PATH *prepending* em ambos os *routers*, de maneira que a rota não seja escolhida em detrimento das restantes, garantindo a simetria do tráfego. Além disso poderíamos ter utilizado route-maps com local pref para influenciar o tráfego que sai do AS de

modo a não utilizar a ligação direta.

```
route-map PREPEND permit 10
 set as-path prepend 10
!
```

```
address-family ipv4
 neighbor 11.101.2.2 activate
 neighbor 11.101.4.2 activate
 neighbor 11.101.5.2 activate
 neighbor 11.101.5.2 route-map PREPEND out
```

Parte 6

Para garantir que o tráfego passa preferencialmente pela ligação internacional do ISP, é utilizado um local-preference no *router* para o *router* PE2.

```
route-map local-pref permit 10
 set local-preference 200
!
```

```
neighbor 30.1.254.132 remote-as 302
neighbor 30.1.254.132 route-map local-pref in
neighbor 30.1.254.132 maximum-prefix 50
```

Para distinguir a saída para uma determinada rede que possui uma prioridade menor foi criado um route-map que aplica um *weight* maior para essa saída.

```
!
ip prefix-list MATCH101 seq 5 permit 4.4.4.4/32
ip prefix-list MATCH101 seq 10 permit 8.8.8.8/32
!
```

Figura 36 - Configuração no router PE2

```
route-map ROUTEAS101 permit 10
 match ip address prefix-list MATCH101
 set weight 65535
!
```

```
PC301_1> trace 4.4.4.4 -m 16 -P 6
trace to 4.4.4.4, 16 hops max (TCP), press Ctrl+C to stop
 1   194.14.57.251   9.138 ms  9.728 ms  10.916 ms
 2   30.1.254.133   31.643 ms  31.828 ms  30.097 ms
 3   40.0.0.251   53.957 ms  51.005 ms  42.011 ms
 4   20.202.142.1   53.723 ms  73.871 ms  63.969 ms
 5   11.102.7.1   95.725 ms  94.443 ms  96.201 ms
 6   10.102.3.1   115.981 ms  83.864 ms  118.547 ms
 7   4.4.4.4   97.651 ms  108.918 ms  107.948 ms
```

Parte 1

a) Sim, porque o tráfego entre o router P3 e o PE6 é feita através do P4 – P6 e PE6.

b) Continua a existir comunicação mesmo sem o comando default-information originate nos *routers* PEn, uma vez que, o AS101 não utiliza este tipo de informação para o encaminhamento.

```
R101_1#trace 30.1.255.6

Type escape sequence to abort.
Tracing the route to 30.1.255.6

  1 11.101.2.2 8 msec 24 msec 20 msec
  2 10.102.3.2 32 msec 60 msec 44 msec
  3 11.102.7.2 44 msec 56 msec 56 msec
  4 20.202.141.2 [AS 202] 68 msec 84 msec 76 msec
  5 10.1.23.2 80 msec 100 msec 92 msec
  6 10.1.28.2 136 msec 128 msec 120 msec
  7 10.1.38.2 132 msec 160 msec 140 msec
```

c) Sim, porque possui uma interface *loopback* pública, e possui o BGP que encaminha para fora.

```
P4#trace
Protocol [ip]:
Target IP address: 4.4.4.4
Source address: 30.0.255.4
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to 4.4.4.4

  1 10.1.36.2 36 msec 20 msec 20 msec
  2 10.1.28.1 36 msec 28 msec 76 msec
  3 10.1.23.1 92 msec 72 msec 68 msec
  4 20.202.141.1 88 msec 56 msec 64 msec
  5 11.102.7.1 96 msec 84 msec 88 msec
  6 10.102.3.1 116 msec 108 msec 100 msec
  7 11.101.2.1 116 msec 128 msec 124 msec
```

d) Outra forma de não utilizar a redistribuição das rotas, seria a utilização de injeção, onde todos os *routers* internos devem correr iBGP, além do protocolo IGP em execução.

e) O comando deve existir em pelo menos um *router* PE1 ou PE3, de maneira que os *routers* da área 65005 consigam encaminhar para um determinado *router* no AS 302 (que consegue encaminhar posteriormente o tráfego).

```
PC65005_1> ping 4.4.4.4
4.4.4.4 icmp_seq=1 timeout
4.4.4.4 icmp_seq=2 timeout
4.4.4.4 icmp_seq=3 timeout
4.4.4.4 icmp_seq=4 timeout
4.4.4.4 icmp_seq=5 timeout

PC65005_1> ping 8.8.8.8
8.8.8.8 icmp_seq=1 timeout
8.8.8.8 icmp_seq=2 timeout
8.8.8.8 icmp_seq=3 timeout
8.8.8.8 icmp_seq=4 timeout
8.8.8.8 icmp_seq=5 timeout
```

f) Os *routers* PEn, à exceção dos routers PE1 e PE3 não têm informação acerca das redes de interligação do PE1 e PE3 para os outros AS. Desta forma, os *routers* PE1 e PE3 anunciam aos restantes *peers*, que caso queiram enviar para fora devem enviar para eles, através do comando next-hop-self. A partir desta informação os *routers* interiores de iBGP sabem que devem enviar para os *routers* PE1 ou PE3, sempre que pretendam

enviar pacotes para fora.

3. Conclusões

Neste trabalho foi colocado em prática o protocolo BGP, bem como outros protocolos abordados anteriormente: OPSFv2, RIP e rotas estáticas.

No decurso da realização do trabalho, verificou-.se que o BGP é um protocolo bastante útil na troca de informação com outros domínios ou sistemas autónomos. No entanto, este protocolo exige que a maior parte das configurações, seja realizada com o auxílio do gestor de rede, por exemplo, a configuração dos vizinhos ou *peers* BGP.

Ao realizar o trabalho, as maiores dificuldades surgiram de:

- Tentar compreender quais os objetivos da implementação fornecida pelo docente;
- Aplicar as políticas corretamente, de forma a obter os resultados pretendidos;
- Garantir que todos os requisitos presentes nas tarefas foram cumpridos com sucesso.

Contudo, uma das principais preocupações, foi garantir que independentemente dos filtros utilizados, fosse sempre possível a comunicação entre os clientes dentro e fora do ISP.

Em suma, pensamos ter cumprido o objetivo deste trabalho, que se centra na realização dos exercícios propostos em cada uma das cinco tarefas evidenciadas no enunciado.