Graphical user interface

Description automatically generated with low confidence

2021/2022

Daniel Graça, n.º 20948

Guilherme Lourenço, n.º 23053

Grupo 9

|  |
| --- |
|  |
| Redes de Dados II |
|  |
| Trabalho Prático 4 |

(Parte 1)

Índice

[Objetivos 2](#_Toc105774444)

[Cenário único 3](#_Toc105774445)

[Topologia da rede (enunciado) 3](#_Toc105774446)

[Topologia da rede 4](#_Toc105774447)

[Tabela de endereçamento 5](#_Toc105774448)

[Tarefa 1: Montagem da rede 6](#_Toc105774449)

[Passo 1: Apagar as configurações dos routers 6](#_Toc105774450)

[Passo 2: Ligação dos cabos aos equipamentos de acordo com a topologia 7](#_Toc105774451)

[Tarefa 2: Configurações Básicas 8](#_Toc105774452)

[Passo 1: Configurações básicas em cada router 8](#_Toc105774453)

[Tarefa 3: Configuração das *interfaces* dos routers 10](#_Toc105774454)

[Passo 1: Esquema do endereçamento do *AS* atríbuido 10](#_Toc105774455)

[Passo 2: Configurar as *interfaces* de acordo com a topologia e a tabela de endereçamento 10](#_Toc105774456)

[Passo 3: Verificação das *interfaces* e respetivos endereços 11](#_Toc105774457)

[Tarefa 4: Configuração do encaminhamento *IGP* 13](#_Toc105774458)

[Passo 1: Utilização do *OSPF* como *IGP* 13](#_Toc105774459)

[Passo 2: Consulta e verificação da tabela de encaminhamento 13](#_Toc105774460)

[Tarefa 5: Configuração do encaminhamento *EGP* 20](#_Toc105774461)

[Passo 1: Configuração do *BGP* 20](#_Toc105774462)

[Passo 2: Interpretação de alguns comandos 20](#_Toc105774463)

[Passo 3: Configurações necessárias para que exista conectividade entre todas as redes do nosso *AS* e as redes dos outros *AS’s* 26](#_Toc105774464)

[Passo 4: Explicação do comando *neighbor <endereço IP> next-hop-self* 27](#_Toc105774465)

[Passo 5: Configuração dos routers de modo a que a rede *110.110.0.0/0* não seja importada para o *IGP* 27](#_Toc105774466)

[Passo 6: Configuração do R1 para que este seja o ponto de saída preferencial do tráfego gerado dentro do *AS* do grupo 28](#_Toc105774467)

[Passo 7: Configuração do R1 para que este seja o ponto de entrada preferencial no *AS* do grupo 29](#_Toc105774468)

[Passo 8: Explicação do processo utilizado pelo *BGP* para seleção do melhor caminho 30](#_Toc105774469)

[Passo 9: Explicação da função do comando do *BGP* *neighbor <endereço IP do peer> update source <endereço IP>* e em que situações deve ser utilizado 30](#_Toc105774470)

[Passo 10: Configurações para que seja suportado *BGP equal cost multipath* 30](#_Toc105774471)

[*Troubleshooting* 33](#_Toc105774472)

[Conclusão 34](#_Toc105774473)

# Objetivos

Os objetivos deste trabalho prático são os seguintes:

* Montagem da componente física de uma rede;
* Configuração de equipamento ativo;
* Configuração de protocolos de encaminhamento *Intra* e *Inter* domínio;
* Troubleshooting;

# Cenário único

## Topologia da rede (enunciado)

Diagram

Description automatically generated

Figura 1 Topologia da rede (enunciado)

## Topologia da rede

Diagram, radar chart

Description automatically generated

Figura 2 Topologia da rede

## Tabela de endereçamento

|  |  |  |
| --- | --- | --- |
| Dispositivo | Interface | Endereço IPv4 |
| R1 | S0/0 | 172.100.100.9 |
| F0/0 | 200.9.0.1 |
| Lo0 | 1.1.1.1 |
| R2 | S0/0 | 172.200.200.9 |
| F0/1 | 200.9.0.6 |
| Lo0 | 200.9.1.1 |
| R3 | F0/0 | 200.9.0.2 |
| F0/1 | 200.9.0.5 |
| Lo0 | 3.3.3.3 |
| AS100 | S0/0 | 172.100.100.100 |
| F0/0 | 192.168.2.1 |
| Lo0 | 100.100.0.1 |
| AS200 | S0/0 | 172.200.200.100 |
| F0/0 | 192.168.2.2 |
| Lo0 | 200.200.0.1 |

Figura 3 Tabela de endereçamento

## Tarefa 1: Montagem da rede

### Passo 1: Apagar as configurações dos routers

De modo a certificar que os routers estão com as configurações “limpas”, procedemos à remoção das configurações dos routers. Para isto, corre-se o comando “reload” nos routers.

**NOTA 1:** No programa *GNS3* aquando a execução do comando “reload”, este deixa de responder (no terminal do router no qual foi executado o comando). Para isso, removeram-se as configurações através da *GUI*, como se pode ver pela seguinte figura.

Graphical user interface, application

Description automatically generated

Figura 4 Remoção das configurações de um router

### Passo 2: Ligação dos cabos aos equipamentos de acordo com a topologia

Neste passo procedeu-se à ligação dos cabos aos equipamentos. Para os routers R1, R2 e R3 (pertencentes ao nosso *AS*), bem como entre os AS100 e AS200 escolheram-se cabos do tipo *crossover*.

Para as liagações entre os routers que interligam a nossa *AS* com as *AS’s* externas utilizaram-se cabos *serial*, sendo que os *DCE’s* correspondem às *interfaces* externas dos routers da nossa área.

Diagram

Description automatically generated

Figura 5 Montagem da rede de acordo com a topologia

## Tarefa 2: Configurações Básicas

### Passo 1: Configurações básicas em cada router

Neste passo, efetuaram-se as configurações básicas necessárias em cada router.

Nos seguintes pontos, para efeitos de exemplificação, apenas são listados comandos para um router só.

#### Atribuir um nome ao router

Para atribuir um nome ao router, efetuou-se o seguinte comando:



Figura 6 Atribuição de nome a um router

#### Desativar o *DNS Lookup*

Para desativar o *DNS Lookup* num router, executou-se o seguinte comando:



Figura 7 Desativação da *DNS Lookup*

#### Configurar uma password para o modo *Exec Privileged Mode*

Para configurar uma *password* para o modo *Exec Privileged Mode*, executou-se o seguinte comando:



Figura 8 Configurar uma *password* para o exec *privileged mode*

#### Configurar a *MOTD Banner*

Ao efetuar o seguinte comando, configurou-se a *MOTD (Message of The Day) Banner* num router:



Figura 9 Configuração da *MOTD Banner*

#### Configurar uma password para ligações do tipo *console*

Para configurar uma *password* para ligações do tipo *console*, executaram-se os seguintes comandos:



Figura 10 Configuração de uma *password* para ligações do tipo *console*

#### Configurar uma password para ligações do tipo *line vty*

Para configurar uma *password* para ligações do tipo *vty*, executaram-se os seguintes comandos:

#### 

Figura 11 Configuração de uma *password* para ligações do tipo *vty*

## Tarefa 3: Configuração das *interfaces* dos routers

### Passo 1: Esquema do endereçamento do *AS* atríbuido

**NOTA 2:** O esquema de endereçamento do *AS* atríbuido está listado na tabela de endereçamento.

A subdivisão da rede é constituída por uma subrede /17 e duas /30. Escolheu-se fazer a subdivisão da rede desta forma de modo a minimizar o desperdício de endereços. É de notar que esta subdivisão não é a melhor prática numa rede com dispositivos reais, visto que nesta simulação apenas existem 3 routers existentes dentro da *AS* que nos foi atribuída.

Para complementar, a *interface* de *loopback* do R3 é uma forma de simular a continuação da rede.

### Passo 2: Configurar as *interfaces* de acordo com a topologia e a tabela de endereçamento

Neste passo, efetuaram-se as configurações das interfaces necessárias (de acordo com a topologia e a tabela de endereçamento) em cada router.

Nas seguintes alíneas, para efeitos de exemplificação, apenas são listados comandos para um (ou mais, caso assim o seja necessário) router(s).

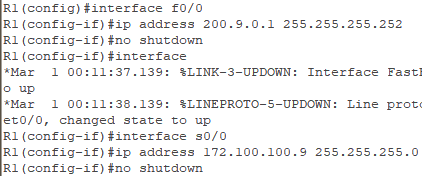


Figura 12 Configuração das *interfaces* de um router

#### i) Configuração das *interfaces* *DCE*

Como optámos por utilizar ligações do tipo *serial* entre os routers que interligam a nossa *AS* com as *AS’s* externas, definimos os nossos routers como tendo as interfaces *DCE*.



Figura 13 Configuração do clockrate - *interface DCE*



Figura 14 Configuração do clockrate - interface DTE

#### ii) Configuração das *interfaces loopback*

Criámos uma *interface* de *loopback* no R3, como forma de simular a continuação da rede, como já fora referido no **passo 1**, da **tarefa 3**.

Para configurar uma *interface* de *loopback*, efetuou-se o seguinte comando:

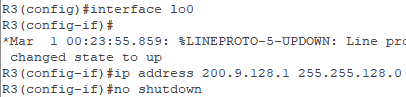


Figura 15 Configuração de uma *interface* de *lookback*

### Passo 3: Verificação das *interfaces* e respetivos endereços

Neste passo, verificamos o estado das interfaces e respetivos endereços atribuídos, através do uso do comando *show ip interface brief*.

Podemos observar, com este comando, cada *interface* de um router bem como:

* O endereço atribuído à mesma (se for o caso);
* O estado “OK?” da mesma (representado por *“*YES*”* se o endereço *IP* for válido ou “NO” em caso contrário);
* O método – que representa o modo como a *interface* foi configurada – que se encontra em “manual” porque foi configurada através da *CLI* (*Command Line*);
* Um outro estado, que representa o estado da própria *interface* física (“up – encontra-se ativa; “down” – encontra-se inativa; “administratively down” – encontra-se administrativamente inativa, ou seja, esta não foi configurada);



Figura 16 Verificação das *interfaces* e respetivos *endereços* - R1

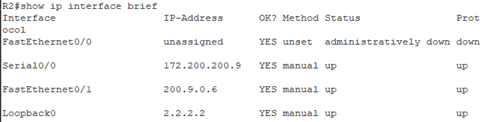


Figura 17 Verificação das interfaces e respetivos endereços - R2

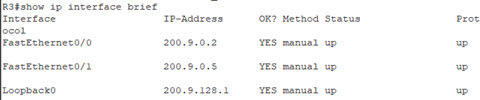


Figura 18 Verificação das interfaces e respetivos endereços - R3

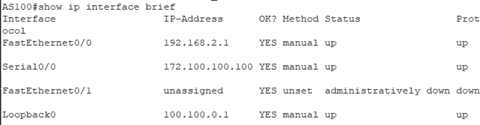


Figura 19 Verificação das interfaces e respetivos endereços - AS100

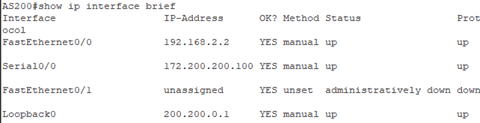


Figura 20 Verificação das interfaces e respetivos endereços - AS200

Como podemos verificar pelas figuras apresentadas anteriormente, as *interfaces* ficaram corretamente configuradas, com os endereços corretos atribuídos.

## Tarefa 4: Configuração do encaminhamento *IGP*

### Passo 1: Utilização do *OSPF* como *IGP*

Para fazer comunicação na nossa rede, utilizou-se o *OSPF* como *IGP* (*Interior Gateway Protocol*). Configuraram-se todos os routers com *OSPF*, como se assemelha a seguinte configuração do R1:

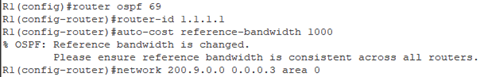


Figura 21Configuração do *OSPF* como *IGP* – R1

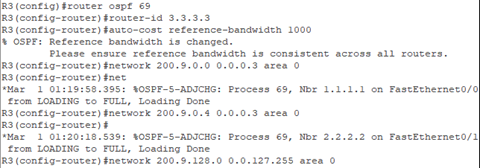


Figura 22 Configuraçao do *OSPF* como *IGP* - R3

### Passo 2: Consulta e verificação da tabela de encaminhamento

Neste passo, consultamos e verificamos a tabela de encaminhamento dos routers, através do uso do comando *show ip route*.

Podemos observar, com este comando, cada ligação que um determinado router conhece, bem como:

* O protocolo de encaminhamento (“O” – *OSPF*; “C” – diretamente conectado, entre outras);
* O tipo de rota, apesar de neste caso específico não ser listado;
* O endereço que indica a rede remota que este conhece;
* A distância administrativa (x) e a métrica da rota (y), representada por [x/y];
* Especifica o endereço do próximo router para a rede remota (via x.x.x.x);
* Especifica a última vez que a rota foi atualizada ( com o formato *horas:minutos:segundos*);
* Especifica qual a interface pela qual a rede consegue ser acedida/alcançada.

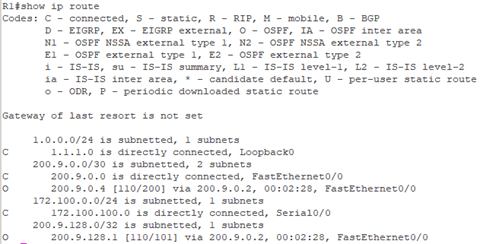


Figura 23 Consulta e verificação da tabela de encaminhamento

Como podemos observar, são listadas todas as rotas que um router aprende (neste router em específico, aprendidas por *OSPF* – representadas por “O”) e as ligações diretamente conectadas (representadas por “C”), como já explicado anteriormente.

Verificamos, também, a informação geral sobre os processos de encaminhamento *OSPF*, com uso do comando *show ip ospf*.

Este comando dispõe de muita informação útil, pelo que não iremos detalhar cada uma dessa informação. Podemos, no entanto, concluir – segundo a figura que se segue – que o R1 está a correr *OSPF* com o *ID* 1.1.1.1 e que se encontra na área OSPF 0 (área *backbone*).

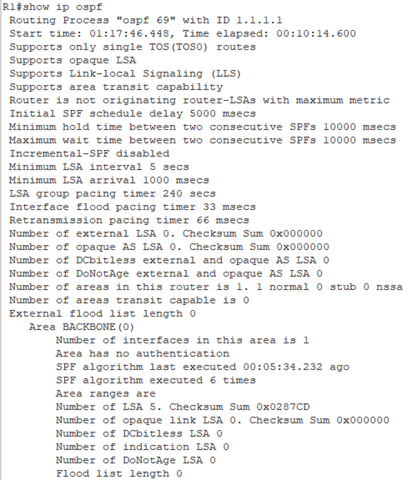


Figura 24 Consulta dos processos de encaminhamento *OSPF*

**NOTA 3:** *"Number of interfaces in this area is 1"* -> esta linha significa que existe um total de 1 *interface* diretamente conectada a este router, das que pertencem à área 0 da rede *OSPF* em funcionamento. No router 3, por exemplo, já mostraria 3 *interfaces* (200.9.0.2/30, 200.9.0.5/30 e 200.9.128.1/17 (lo0)).

O comando *show ip ospf interface* revela informações mais detalhadas acerca das *interfaces* numa determinada área *OSPF*.

Mais uma vez, este comando dispõe de muita informação útil, pelo que não iremos detalhar cada uma dessa informação. Podemos, no entanto, concluir – segundo a figura que se segue – que existe uma *interface fastEthernet0/0* com o endereço 200.9.0.1/30 que se encontra na área 0. O router *ID* representa o *ID* de um router na área OSPF, sendo que deve ser um *ID* único. Pode se observar que existe uma informação “*State DR”*. Com esta informação sabemos que este foi o *Designated Router* (*DR*) escolhido pelo *OSPF*.

**NOTA 4:** Apesar do router com o *ID* se identificar como *DR*, as prioridades entre os routers da área 0 são iguais, pelo que o verdadeiro *DR* é o router com o *ID* 3.3.3.3 (que é o maior endereço na área).

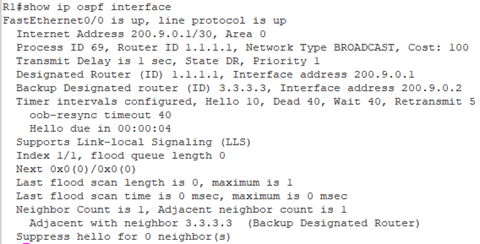


Figura 25 Consulta da informação sobre as *interfaces* numa área *OSPF*

O comando *show ip ospf neighbor*, tal como o comando sugere, apresenta informação sobre os vizinhos *OSPF* de um determinado router.

Podemos observar, com este comando, as informações sobre um vizinho *OSPF* :

* *ID* do vizinho (em formato x.x.x.x);
* Prioridade: Indica a prioridade de cada router, sendo que o router com a maior prioridade torna-se o *DR*. Caso as prioridades sejam iguais, o router com o maior *ID* torna-se o *DR*;
* O estado indica se o router é *DR* ou *BDR*, e o “FULL” significa que o router está adjacente com o seu vizinho;
* “Dead Time” indica quanto tempo um router espera para receber um pacote “hello” de um vizinho antes de declarar esse vizinho como “down”;
* O endereço representa a *interface* do vizinho a qual está diretamente conectado;
* A *interface* física a qual está ligado;



Figura 26 Consulta dos vizinhos OSPF de um determinado router

Nesta **figura 26** podemos concluir que o R1 tem um vizinho *OSPF* com o *ID* 3.3.3.3, com priodidade 1, estado “FULL”, sendo um *BDR* (*Backup Designated Router*), com um “Dead Time” de 38 segundos, endereço 200.9.0.2 e interface *FastEthernet0/1*. (Apesar de não se conseguir ver pela figura, é possível saber pela tabela de endereçamento).

As próximas figuras são os mesmos comandos já listados anteriormente, mas em dispositivos diferentes com o objetivo de mostrar resultados diferentes. (Não se mostra o R2 neste passo porque os resultados são semelhantes aos do R1).

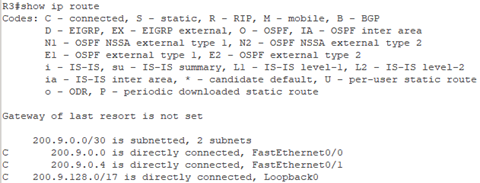


Figura 27 Consulta e verificação da tabela de encaminhamento

Este router R3 apenas apresenta rotas diretamente conectadas porque ele não aprende mais rotas nenhumas por *OSPF*. O R1 e o R2 aprendem rotas *OSPF* através do R3, porque é o *DR*.

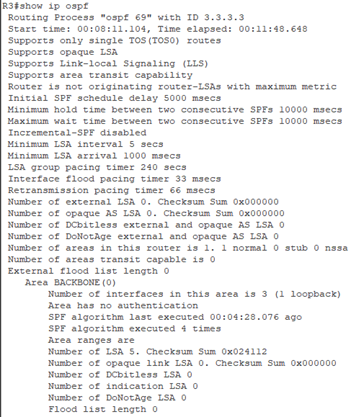


Figura 28 Consulta dos processos de encaminhamento *OSPF*

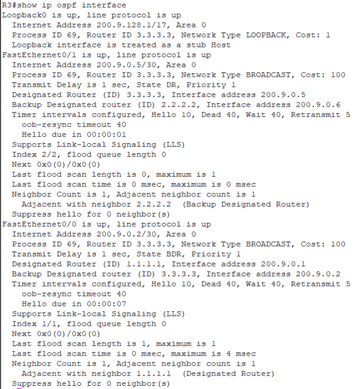


Figura 29 Consulta da informação sobre as *interfaces* numa área *OSPF*

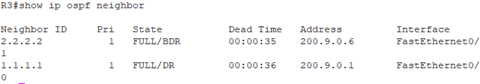


Figura 30 Consulta dos vizinhos *OSPF* de um determinado router

## Tarefa 5: Configuração do encaminhamento *EGP*

### Passo 1: Configuração do *BGP*

Neste passo, procedeu-se à configuração do *BGP* nos routers fronteira, como de acordo com a topologia da rede apresentada anteriormente.

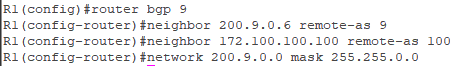


Figura 31 Configuração do BGP num determinado router



Figura 32 Configuração de um rota estática onde o tráfego irá parar se não existir uma certa rota nesta rede

O comando *ip route 200.9.0.0 255.255.0.0 null 0* tem como objetivo criar uma rota estática para si mesmo, de forma a assegurar que o tráfego vá parar ao router se não existir uma rota mais específica.

**NOTA 5:** configuraram-se os routers de modo a que se estabelecesse uma relação de peering entre os R1 e R2, por *iBGP*.

### Passo 2: Interpretação de alguns comandos

Neste passo, iremos interpretar alguns comandos sobre *BGP*.

Este primeiro comando não é sobre *BGP* em específico (e já foi mencionado na **tarefa 4 – passo 2**), mas contém rotas *BGP* das quais vamos falar. Uma diferença no *output* deste comando para o anterior é a rota estática, que está a apontar para si mesmo (router R1).

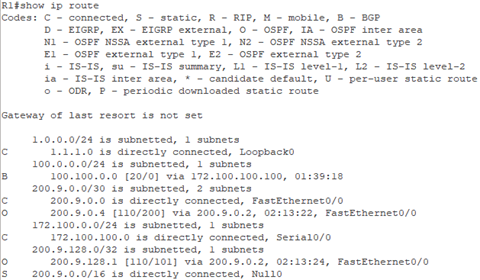


Figura 33 Consulta e verificação da tabela de encaminhamento

Após a configuração de BGP nos routers, podemos observar que o R1 aprendeu como chegar ao destino 100.100.0.0 através da *interface* 172.100.100.100. Este é um *peering* de *eBGP* entre o R1 e o router do *AS100*.

Verificamos, também, a informação geral sobre as entradas *BGP* na tabela de encaminhamento, com uso do comando *show ip bgp*.

Podemos observar, com este comando:

* O estado da entrada, em que:
  + \* (asterisco): representa que a entrada é válida;
  + > (símbolo “maior que”): representa que esta entrada é a melhor a usar para essa rede;
* O endereço da rede que se conhece pela entrada;
* O “Next Hop” (próximo salto), que representa qual a próxima interface por onde se vai passar para chegar ao destino;
* A métrica, que é um atributo que permite influenciar *AS’s* a usar uma determinada rota de entrada para outro *AS* quando existem múltiplas rotas para chegar a esse mesmo *AS* de destino. A métrica preferencial é o valor mais baixo;
* A “Local Preference”, que é outro atributo, é semelhante à metrica. No entanto, esta permite influencia o ponto de de saída de um *AS*. A “Local Preference” preferencial é a maior valor;
* O peso (“Weight”), mais um atributo de escolha do melhor caminho. O peso preferencial é o maior valor;
* O “Path”, que representa o caminho para chegar ao destino (*AS’s* por onde passa);

**NOTA 6:** O *BGP* utiliza os atributos como forma de escolher o melhor caminho. Quando duas rotas encontram o mesmo valor num atributo, o BGP vai comparando outros atributos até encontrar um atributo com melhor (maior ou menor valor, dependendo do atributo). O primeiro atributo a ser visto é o peso, depois a “Local Preference”, seguido de outros quantos atributos até encontrar um valor diferente ou chegar ao fim da lista de atributos.

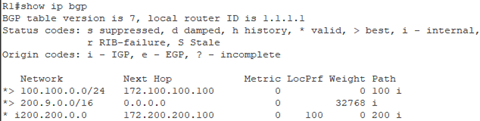


Figura 34 Consulta das entradas na tabela de encaminhamento *BGP*

Nesta figura anterior, podemos concluir que, por exemplo, a rede 100.100.0.0/24 tem uma forma válida e é a melhor forma de se chegar a ela através do salto 172.100.100.100, com uma métrica de 0 e sem “local preference”, um peso de 0 e através o “path” 100, que representa passar pelo *AS* 100. Por fim, depois do “path” é apresentado um “i”. Este caractere representa que a rede foi anunciada para o *BGP* através do comando *network*.

Para consultar o estado de toda as ligações do *BGP*, executou-se o comando *show ip bgp summary*.

Podemos observar, com este comando:

* O endereço da *interface* do próximo vizinho *BGP* de um determinado router;
* “V” – representa a versão do *BGP* comunicada para um determinado vizinho;
* “*AS*” – representa o número do *AS* ao qual o vizinho pertence;
* “MsgRcvd” – representa o número de mensagens recebidas, de um determinado vizinho;
* “MsgSent” – representa o número de mensagens enviadas para um determinado vizinho;
* “TblVer” – versão da tabela *BGP* de um determinado vizinho. Este número da versão é incrementado cada vez que existem alterações;
* “InQ” – número de mensagens recebidas através da fila de “input”;
* “OutQ” – número de mensagens prontas a enviar pela fila de “output”;
* “Up/Down” – tempo que um determinado vizinho esteve “up” ou “down”, em formato dias:horas:minutos. (não é dada informação de qual o estado – se “up”, se “down”);
* “State/PfxRcd” – Estado do vizinho/números de rotas recebidas. Se nenhum estado for indicado, então o estado é “up”.

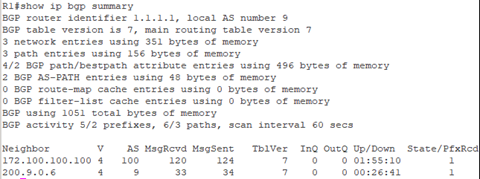


Figura 35 Consulta do estado das conexões *BGP*

Nesta figura anterior podemos observar que existe, por exemplo, um vizinho com uma *interface* com o endereço 172.100.100.100, com versão BGP 4, pertencente ao *AS* 100. Este dispositivo R1 recebeu 33 mensagens e enviou 32 a este vizinho do *AS* 100. A versão da tabela *BGP* é a quarta, pelo que se atualizou a tabela 4 vezes. Verificamos, também, que nao houve envio e recebimento de mensagens através das filas “input” e “output”, como também vemos o tempo de “Up/Down” (apesar de não sabermos se se refere a “up” ou “down”) e o número de rotas recebidas é 1. O estado não é indicado, pelo que significa que o vizinho está “Up”.

Para consultar a informação acerca dos vizinhos *BGP* de um determinado router, executou-se o comando *show ip bgp neighbors*.

Este comando dispõe de muita informação útil, pelo que não iremos detalhar cada uma dessa informação. A informação que consideramos mais relevante durante a execução das configurações foram o “*BGP State = Active”* e o *“BGP State = Established”*. Este foi, possivelmente, os detalhes aos quais tivemos mais atenção, porque um estado a “*Active*” significa que um router está atentar estabelecer um *peering* com outro router, enquanto que “*Established*” significa que esse router conseguiu estabelecer *peering* com outro router.

BGP neighbor is 172.100.100.100, remote AS 100, external link

BGP version 4, remote router ID 100.100.0.1

BGP state = Established, up for 01:56:29

Last read 00:00:29, last write 00:00:29, hold time is 180, keepalive interval is 60 seconds

Neighbor capabilities:

Route refresh: advertised and received(old & new)

Address family IPv4 Unicast: advertised and received

Message statistics:

InQ depth is 0

OutQ depth is 0

Sent Rcvd

Opens: 1 1

Notifications: 0 0

Updates: 5 1

Keepalives: 119 119

Route Refresh: 0 0

Total: 125 121

Default minimum time between advertisement runs is 30 seconds

For address family: IPv4 Unicast

BGP table version 7, neighbor version 7/0

Output queue size : 0

Index 1, Offset 0, Mask 0x2

1 update-group member

Sent Rcvd

Prefix activity: ---- ----

Prefixes Current: 1 1 (Consumes 52 bytes)

Prefixes Total: 3 1

Implicit Withdraw: 0 0

Explicit Withdraw: 2 0

Used as bestpath: n/a 1

Used as multipath: n/a 0

Outbound Inbound

Local Policy Denied Prefixes: -------- -------

Bestpath from this peer: 1 n/a

Total: 1 0

Number of NLRIs in the update sent: max 1, min 0

Connections established 1; dropped 0

Last reset never

Connection state is ESTAB, I/O status: 1, unread input bytes: 0

Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 1

Local host: 172.100.100.9, Local port: 28750

Foreign host: 172.100.100.100, Foreign port: 179

Enqueued packets for retransmit: 0, input: 0 mis-ordered: 0 (0 bytes)

Event Timers (current time is 0xD49C58):

Timer Starts Wakeups Next

Retrans 124 0 0x0

TimeWait 0 0 0x0

AckHold 119 117 0x0

SendWnd 0 0 0x0

KeepAlive 0 0 0x0

GiveUp 0 0 0x0

PmtuAger 0 0 0x0

DeadWait 0 0 0x0

iss: 429990350 snduna: 429992864 sndnxt: 429992864 sndwnd: 15335

irs: 807567781 rcvnxt: 807570140 rcvwnd: 15491 delrcvwnd: 893

SRTT: 300 ms, RTTO: 303 ms, RTV: 3 ms, KRTT: 0 ms

minRTT: 0 ms, maxRTT: 300 ms, ACK hold: 200 ms

Flags: active open, nagle

IP Precedence value : 6

Datagrams (max data segment is 1460 bytes):

Rcvd: 126 (out of order: 0), with data: 119, total data bytes: 2358

Sent: 244 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 124, total data bytes: 2513

BGP neighbor is 200.9.0.6, remote AS 9, internal link

BGP version 4, remote router ID 2.2.2.2

BGP state = Established, up for 00:28:08

Last read 00:00:08, last write 00:00:08, hold time is 180, keepalive interval is 60 seconds

Neighbor capabilities:

Route refresh: advertised and received(old & new)

Address family IPv4 Unicast: advertised and received

Message statistics:

InQ depth is 0

OutQ depth is 0

Sent Rcvd

Opens: 1 1

Notifications: 0 0

Updates: 4 3

Keepalives: 31 31

Route Refresh: 0 0

Total: 36 35

Default minimum time between advertisement runs is 0 seconds

For address family: IPv4 Unicast

BGP table version 7, neighbor version 7/0

Output queue size : 0

Index 2, Offset 0, Mask 0x4

2 update-group member

Sent Rcvd

Prefix activity: ---- ----

Prefixes Current: 2 1 (Consumes 52 bytes)

Prefixes Total: 3 2

Implicit Withdraw: 0 0

Explicit Withdraw: 1 1

Used as bestpath: n/a 0

Used as multipath: n/a 0

Outbound Inbound

Local Policy Denied Prefixes: -------- -------

Total: 0 0

Number of NLRIs in the update sent: max 1, min 0

Connections established 1; dropped 0

Last reset never

Connection state is ESTAB, I/O status: 1, unread input bytes: 0

Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255

Local host: 200.9.0.1, Local port: 23284

Foreign host: 200.9.0.6, Foreign port: 179

Enqueued packets for retransmit: 0, input: 0 mis-ordered: 0 (0 bytes)

Event Timers (current time is 0xD4BCD4):

Timer Starts Wakeups Next

Retrans 35 0 0x0

TimeWait 0 0 0x0

AckHold 32 29 0x0

SendWnd 0 0 0x0

KeepAlive 0 0 0x0

GiveUp 0 0 0x0

PmtuAger 0 0 0x0

DeadWait 0 0 0x0

iss: 14358324 snduna: 14359152 sndnxt: 14359152 sndwnd: 16099

irs: 3109346374 rcvnxt: 3109347148 rcvwnd: 16156 delrcvwnd: 228

SRTT: 297 ms, RTTO: 321 ms, RTV: 24 ms, KRTT: 0 ms

minRTT: 24 ms, maxRTT: 300 ms, ACK hold: 200 ms

Flags: active open, nagle

IP Precedence value : 6

Datagrams (max data segment is 536 bytes):

Rcvd: 37 (out of order: 0), with data: 32, total data bytes: 773

Sent: 66 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 34, total data bytes: 827

Figura 36 Consulta das conexões dos vizinhos *BGP*

Por fim, o comando *show ip bgp rib-failure* permite verificar todas as entradas que deram erro ao serem inseridas na tabela de encaminhamento *BGP*. Por exemplo, se um endereço já estiver na tabela de encaminhamento do R3 aprendida pelo R1 e este R3 tentar injetar o mesmo endereço aprendido pelo R2 (por BGP), esse mesmo endereço irá aparecer na tabela de *rib-failure*.



Figura 37 Consulta de falhas a "instalar" rotas na tabela de encaminhamento

Como podemos averiguar pela **figura 37**, não houve qualquer problema com as nossas configurações.

### Passo 3: Configurações necessárias para que exista conectividade entre todas as redes do nosso *AS* e as redes dos outros *AS’s*

Neste passo, procedeu-se às configurações necessárias para que exista conectividade entre todas as redes do nosso *AS* e as redes dos outros *AS’s*.

Para isso, executaram-se os seguintes comandos:



Figura 38 Atualização do *next-hop­* - R1



Figura 39 Atualização do *next-hop­* - R2



Figura 40 Anúncio da rede 172.100.100.0 a partir do router AS100



Figura 41 Anúncio da rede 192.168.2.0 a partir do router AS100



Figura 42 Anúncio da rede 172.200.200.0 a partir do router AS200



Figura 43 Anúncio da rede 192.168.2.0 a partir do router AS200

Parte da configuração do *iBGP* envolvia a utilização do comando *neighbor <endereço ip do peer> update source <interface>*. No entanto, os routers não estavam a comunicar corretamente após a configuração executada. Por isso, foram executados os seguintes comandos, ao invés do que era suposto, de maneira a chegar ao mesmo resultado:



Figura Configurações dos comandos explicados acima

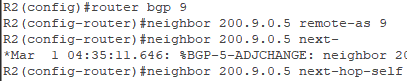


Figura Configurações dos comandos explicados acima



Figura Configurações dos comandos explicados acima

### Passo 4: Explicação do comando *neighbor <endereço IP> next-hop-self*

Quando um router de um *AS* anuncia uma rota a outro router de outro *AS* – ou seja, por *eBGP* – e esse router anuncia essa rota por iBGP para outros routers do mesmo *AS*, o caminho para chegar ao destino não é atualizado. Isto porque, como a rota é anunciada por routers pertencentes ao mesmo *AS*, para evitar loops o *iBGP* descarta esses anúncios. O comando *neighbor <endereço IP> next-hop-self* permite informar um router a como chegar a um destino fora do seu *AS* obrigando a passar por um outro router dentro desse *AS*.

### Passo 5: Configuração dos routers de modo a que a rede *110.110.0.0/0* não seja importada para o *IGP*

Para não permitir uma rede ser importada para o *IGP*, configuraram-se, nos routers de entrada no *AS* os seguintes comandos:

Criou-se uma *prefix-list* para conter o endereço da rede que queremos ignorar nos routers de entrada do nosso *AS*.



Figura 47 Criação de uma *prefix-list*

Verificamos que a *prefix-list* continha o endereço de rede correto, com o seguinte comando:



Figura 48 Verificação da *prefix-list* criada

Para realmente não permitir a rede ser anunciada para o *IGP*, fez-se um filtro com o comando ­*route-map* para que todos os endereços que fossem provenientes da rede 110.110.0.0/24 não fossem permitidos. No entanto, foi preciso fazer outro *route-map* com uma *prefix-list* não existente (ou seja, vazia) para que todos os outros endereços fossem permitidos. Se este último passo não fosse feito, não só estava-se a ignorar os endereços da rede 110.110.0.0/24 como qualquer outro endereço.

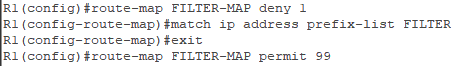


Figura 49 Criação de *route-maps* para filtrar os endereços na(s) *prefix-list*(s)

Por fim, verificamos se os *route-map’s* foram bem configurados, pelo que podemos ver que sim na figura seguinte.

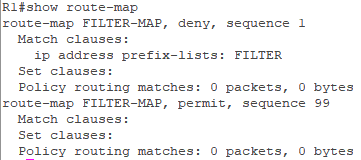


Figura 50 Verificação das *route-maps*

### Passo 6: Configuração do R1 para que este seja o ponto de saída preferencial do tráfego gerado dentro do *AS* do grupo

Para configurar o router R1 como sendo o ponto de saída preferencial de todo o tráfego gerado dentro do nosso *AS*, apenas foi preciso garantir que a “*Local Preference*” do R1 é maior que a dos outros routers dentro do *AS*.

Dessa forma, executaram-se os seguintes comandos:



Figura 51 Alteração da "*Local Preference*" do R1

### Passo 7: Configuração do R1 para que este seja o ponto de entrada preferencial no *AS* do grupo

Para configurar o router R1 como sendo ponto de entrada preferencial do tráfego do nosso *AS*, foram precisos ser executados alguns comandos.

Primeiramente criou-se uma *access-list* (*ACL*) com o endereço da rede *IGP* para utilizar como filtragem no *route-map*.

Com esta lista, criou-se uma regra que, caso o endereço destino pertença à rede que está na *access-list* (neste caso, a rede *IGP*) irá ter um valor de métrica **1**.

Procedeu-se às mesmas configurações no router R2. No entando, neste caso o valor de métrica tem de ser superior, pelo que foi definido o valor **2**.

**NOTA** 7**:** Como já referido anteriormente, o valor de métrica mais baixo é o escolhido pelo *BGP*. Dessa forma, podia-se ter escolhido qualquer valor inteiro 32 *bit* maior que o escolhido no R1.

Por fim, o definiu-se que, dos routers dos *AS’s* vizinhos que fazem borda com os routers de entrada no nosso *AS*,serão aplicadas as regras definidas no *route-map* definido anteriormente (todos os endereços com destino à rede 200.9.0.0/16 terão uma métrica de **1** ou **2**, consoante por onde entrarem).

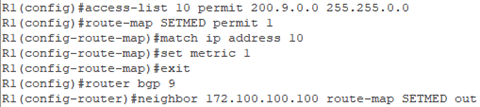


Figura Configuração da *access-list* e do *route-map*

O comando *bgp deterministic-med* tem como função ativar a comparação das variáveis de métrica (*MED*) quando for para escolher rotas anunciadas por diferentes *peers* dentro do mesmo *AS*. No entanto, neste exemplo não é necessário executar este comando.



Figura 53 Configuração do *bgp demterministc-med*

### Passo 8: Explicação do processo utilizado pelo *BGP* para seleção do melhor caminho

A seleção do melhor caminho envolve um conjunto de dados (chamados atributos) que cada router tem no seu software. Esses valores têm uma ordem hierárquica que depois decide qual o caminho preferencial para se enviar um pacote. O atributo que o *BGP* tem maior consideração para o processo de seleção do melhor caminho é o *weight*, em que quanto maior o seu valor, mais preferência tem.

### Passo 9: Explicação da função do comando do *BGP* *neighbor <endereço IP do peer> update source <interface>* e em que situações deve ser utilizado

O comando *neighbor <peer's ip address> update source <interface>* serve para forçar o *BGP* a usar uma certa *interface* de um router quando quiser comunicar com um *neighbor* desse router. O campo da *interface* é geralmente preenchido com uma *interface* *Loopback*, porque como se trata de uma *interface* virtual significa que está sempre ativa, o que garante uma conexão mais segura com um *neighbor*.

### Passo 10: Configurações para que seja suportado *BGP equal cost multipath*

Para configurar *equal cost multipath* em *BGP* deve-se certificar que existem dois *AS* *paths* para o mesmo destino, em que ambos são considerados ter a mesma distância e sem existir configurações de preferência nos routers. É importante denotar que o objetivo neste projeto é implementar um *equal cost multipath* no router R3 para a sub-rede 192.168.2.0/30.

Como tal, executou-se os seguintes comandos nos routers R1 e R2 para anular essas preferências, previamente configuradas nos passos 6 e 7:

* *no neighbor <neighbor interface address> route-map <map name> out -* está a fazer com que o *route-map* que trata da métrica (MED) não seja propagado para o router AS100;
* *bgp default local-preference 100 -* faz com que a atributo *local-preference* volte a ficar com o valor 100, que é o valor predefinido (e equivalente ao *local-preference do router R2*).



Figura 54 Configuração dos comando préviamente explicados no router R1



Figura 55 Configuração dos comandos préviamente explicados no router R2

Em seguida, deve-se configurar as definições que permitem a utilização de *equal cost multipath.* Estes são os comandos a ser executados:



Figura 56 Configuração do *equal cost multipath* para *eBGP*



Figura 57 Configuração do *equal cost multipath* para *iBGP*

O comando *maximum-paths <#>* serve para permitir que possa utilizar vários caminhos para o mesmo destino. A utilização da palavra adicional *iBGP* na configuração do router R3 é necessária porque as suas *interfaces* *BGP* pertencem ao *iBGP* da AS9 e não a *eBGP*.

O comando *bgp bestpath as-path multipath-relax* “diz”ao router para tratar dois caminhos (*AS paths*) que tenham o mesmo comprimento (em termos de número de *AS’s*) e em que certos atributos tenham o mesmo valor como caminhos com custo igual.

Depois destas configurações, aqui estão as tabelas *BGP* distintas resultantes (as tabelas do R2 e AS200 são muito parecidas às tabelas do R1 e AS100, respetivamente):

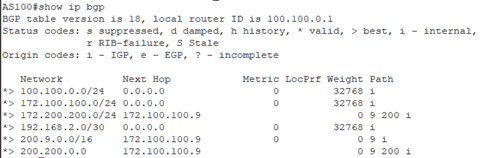


Figura 58 Tabelas de encaminhamento *BGP* - router AS100

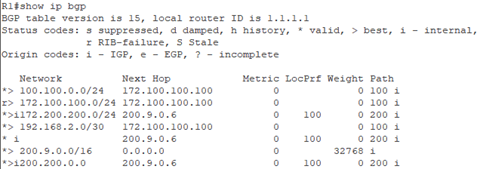


Figura 59 Tabelas de encaminhamento *BGP* - Router R1

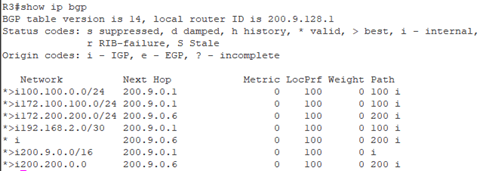


Figura 60 Tabelas de encaminhamento *BGP* - Router R3

Como se pode verificar, o *equal cost multipath* está implementado. No entanto, tecnicamente não está correto. Olhando para a tabela, apenas um dos dois caminhos válidos para a sub-rede 192.168.2.0/30 está operacional.

Isto acontece porque a *AS* de destino tem de ser a mesma, isto é, ter o mesmo número de identificação. Como, neste exemplo, por um caminho a última *AS* é a AS100 e pelo o outro é a AS200, o router R3 acaba por não considerar ambos os caminhos como igualmente utilizáveis, e escolhe um deles.

Para resolver esta situação existem duas possíveis soluções:

* Conectar diretamente os routers R2 e AS100 ou R1 e AS200, criando assim um caminho adicional que acaba com a mesma *AS*;
* Fazer “*AS manipulation*”, que consiste em “enganar” os routers modificando o número de identificação de uma *AS*. No entanto, não é recomendado fazer isto porque pode criar loops.

## *Troubleshooting*

Num trabalho como este, é certo que iriam aparecer problemas nas configurações. E apareceram.

Para colocar este trabalho a funcionar, foi preciso fazer algum *troubleshooting* para tentar perceber o que estava a correr mal, ou o porquê de algo não estar 100% funcional – ou não estar funcional de todo.

Para isso, listamos alguns dos comandos usados para *troubleshooting*:

*show ip route –* mostra na consola o estado atual da tabela de *routing* de um determinado router.

*show ip route bgp* – mostra na consola o estado atual da tabela de *routing* de um determinado router, apenas as rotas *BGP*.

*show ip bgp* – mostra na consola as redes (ou prefixo de rede) que um determinado router conhece, e como lá chegar (*next hop*).

*show run* | *section bgp* – mostra na consola a configuração bgp a correr num determinado router.

**NOTA 8:** Alguns destes comandos foram explicados mais detalhadamente ao longo deste documento.

# Conclusão

Com este trabalho pretendeu-se demonstrar conhecimentos sobre implementação e configuração de protocolos de encaminhamento *Intra* e *Inter* domínio, com os protocolos OSPF e BGP em multihomed (respetivamente), bem como demonstrar a capacidade de resolver problemas (*troubleshooting*).