

Universidade Tecnológica Federal do Paraná - UTFPR

Campus:
Campo Mourão

Professor:
Dr. Luiz Arthur Feitosa dos Santos

E-mail:
luizsantos@utfpr.edu.br

Sumário:

- Protocolos de roteamento
- RIP
- Implementando/configurando RIP-1 e RIP-2
- OSPF (Áreas, anúncio de estado de link, Roter-ID, Wildcard mask e custos)
- Implementando/configurando OSPF

Atenção - este material/slides são em grande parte um apanhado dos conteúdos dos livros apresentados na Bibliografia (último slide) e que estão disponíveis na biblioteca da universidade. Tais slides não dispensam o uso dos referentes livros e de outros materiais de apoio.

Protocolos de roteamento

Agora que sabemos a teoria a respeito de roteamento dinâmico (algoritmos e roteamento). Iremos aprender um pouco mais a respeito dos protocolos que realmente implementam tais algoritmos de roteamento.

RIP (*Routing Information Protocol*)

O RIP é o protocolo de vetor de distância mais utilizado em rede intradomínios. O RIP foi a primeira geração de protocolo de roteamento para o IPv4. O RIP é considerado fácil de configurar e uma boa escolha para redes pequenas. Embora hoje seja considerado um protocolo legado, o RIP ainda está disponível em muitos roteadores e é ótimo para o aprendizado de roteamento.

Apesar do RIP ser implementado com o algoritmo de vetor de distância, na prática são necessárias algumas alterações para que tudo funcione corretamente:

- Os roteadores anunciam o custo para alcançar redes e não nós.
- O custo é definido apenas pelo número de saltos (*hops*) e não por métricas mais complexas (*delay*, etc).
- O custo máximo para uma rede é 15, sendo que 16 é considerado infinito, o que ameniza o problema de contagem para o infinito. Assim, uma rede não pode ter mais do 15 saltos.

Tabelas de roteamento no RIP

A tabela de roteamento no RIP é uma tabela de três colunas: (i) a primeira coluna representa o endereço da rede de destino; (ii) a segunda é endereço do próximo roteador, para o qual o pacote deve ser encaminhado; (iii) o custo (em saltos) para chegar ao destino.

A figura a seguir apresenta um exemplo de tabela, sendo que o “-”, na coluna próximo roteador, significa que é uma rede diretamente conectada ao roteador em questão (não precisa ir para outro roteador).

Rede Destino	Próximo Roteador	Custo (em saltos)
N1	-	1
N2	-	1
N3	R2	2
N4	R2	3

Implementação do RIP

O RIP é implementado como um processo que usa o serviço UDP na porta 520. Isso significa que embora o RIP seja um protocolo de roteamento, ele trabalha na camada de aplicação. E funciona no esquema cliente/servidor.

Há duas versões do RIP: RIP-1 e RIP-2, sendo que a segunda é compatível com a primeira. As principais diferenças são que RIP-1 é *classful* e RIP-2 é *classless*. Também o RIP-2 suporta autenticação, o que melhora sua segurança.

Na sequência serão apresentados apenas informações a respeito do RIP-2. A figura a seguir apresenta o formato da mensagem RIP-2:

Com	Ver	Reserved
Family		Tag
Network address		
Subnet mask		
Next-hop address		
Distance		

Quanto aos campos da mensagem RIP-2:

- *Com* – comando: que pode ser uma pergunta (*request* - 1) RIP ou uma resposta (*response* - 2):
 - A pergunta é enviada por roteadores que foram ligados ou que tiveram rotas que sofreram *time-out*.
 - A resposta pode ser: solicitada ou não solicitada:
 - Uma resposta solicitada é enviada somente em quando há uma mensagem de pergunta e pode conter informações a cerca de um destino específico.
 - Uma resposta não solicitada, é enviada periodicamente, a cada 30 segundos ou se há alterações na tabela de roteamento;
- *Ver* – versão: atualmente é a versão 2;
- *Family* – família: família de protocolo utilizada, será o valor 2 para TCP/IP;
- *Network address* – endereço de rede: endereço de destino;
- *Subnet mask* – máscara de subrede: tamanho do prefixo;
- *Next-hop address* – endereço do próximo salto: endereço do próximo roteador;
- *Distance* – distancia: número de saltos até o destino.

Na prática, para a implementação do RIP, também há algumas diferenças no algoritmo/informações vetor de distancia, mas no geral a implementação é bem similar ao que já foi estudado.

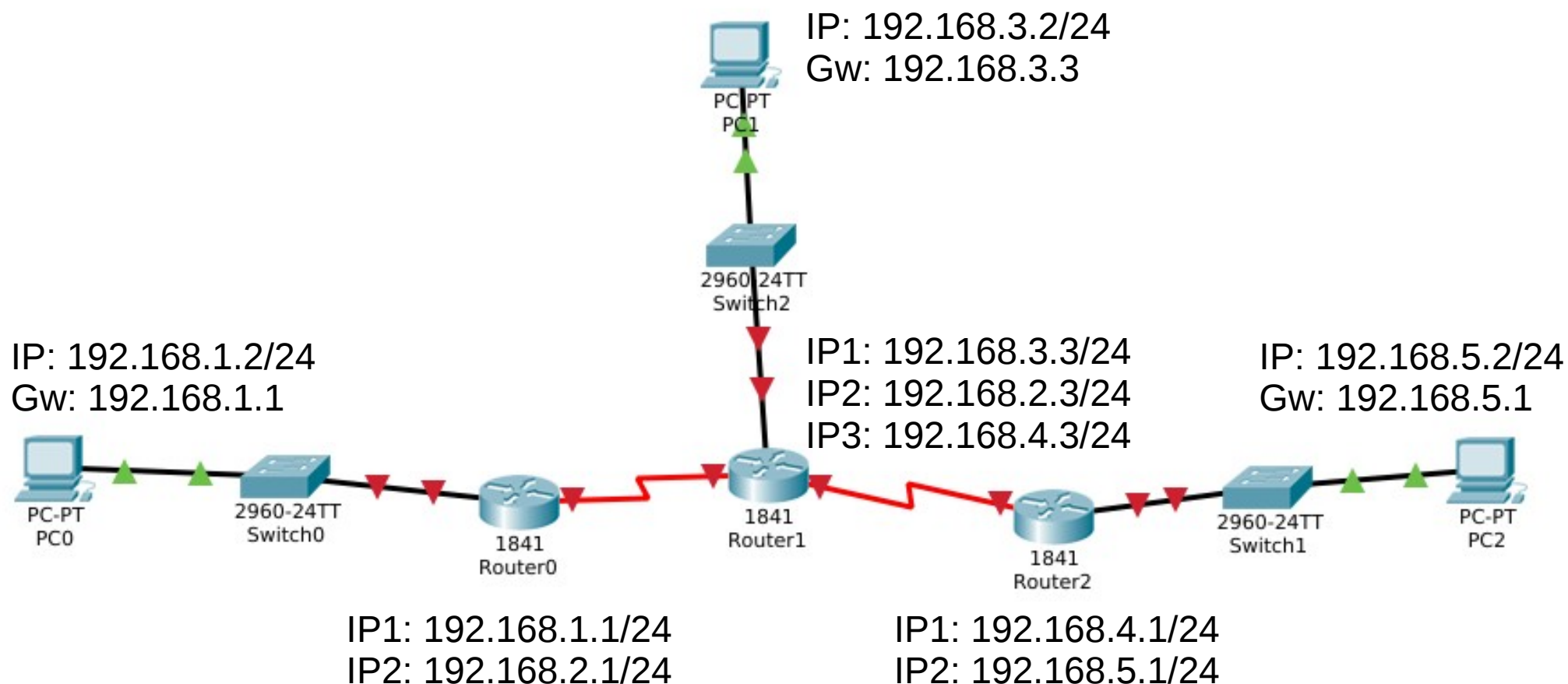
O RIP ainda utiliza três *timers* para suas operações:

- 1) *Periodic timer* (*timer* periódico): controla os avisos periódicos para a rede. Esse tem um valor randômico entre 25 e 53, para evitar excesso de tráfego entre os roteadores;
- 2) *Expiration timer* (*timer* de expiração): controla a validade de uma rota. Quando um pacote atualiza uma rota essa terá uma vida de 180 segundos, toda vez que a rota for atualizada o *timer* é reiniciado, caso contrário a rota será marcada com infinito (custo 16);
- 3) *Garbage collection timer* (*timer* coletor de lixo): é usado para excluir uma rota da tabela de roteamento. Quanto uma rota se torna inválida, ela não é removida imediatamente da tabela de roteamento. A rota em questão continua sendo anunciada com o custo 16, ao mesmo tempo o *timer* é marcado para 120 segundos, para essa rota. Quando o *timer* de 120 segundos chegar em zero, essa rota é realmente excluída da tabela de roteamento. Isso permite que vizinhos tomem ciência de rotas inválidas.

Implementando e configurando RIP em roteadores CISCO

Exemplo 1

Vamos configurar o seguinte cenário de rede:



Bem, devemos iniciar configurando as interfaces de rede e seus endereços, tal como foi feito no slide Roteamento. A configuração do Router 2 ficaria assim:

```
Would you like to enter the initial configuration dialog? [yes/no]: no
Router>enable
Router#configure terminal
Router(config)#interface se0/1/0
Router(config-if)#ip address 192.168.4.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#interface fa0/0
Router(config-if)#ip address 192.168.5.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
```

Em ordem, entramos no modo de configuração do roteador (enable, configure terminal). Depois configuramos os IPs das interfaces (interface ..., ip address...) e dizemos para as interfaces ficarem ativas (no shutdown) e saímos da configuração (exit).

É necessário fazer isso para cada roteador e para cada PC (já vimos isso no slide Roteamento). **Atenção**, os nomes de interface podem mudar conforme sua configuração, então fique atento.

Após as configurações de endereçamento, podemos iniciar a configuração do RIP. Primeiramente, caso o terminal de configuração do roteador não esteja no modo de configuração, é preciso ativá-lo (comandos `enable` e `configure terminal`). No modo configuração apenas execute os seguintes comandos em cada roteador:

- **No Router0:**

```
Router(config)#router rip
Router(config-router)#network 192.168.1.0
Router(config-router)#network 192.168.2.0
```

- **No Router1:**

```
Router(config)#router rip
Router(config-router)#network 192.168.2.0
Router(config-router)#network 192.168.3.0
Router(config-router)#network 192.168.4.0
Router(config-router)#exit
```

- **No Router2:**

```
Router(config)#router rip
Router(config-router)#network 192.168.4.0
Router(config-router)#network 192.168.5.0
Router(config-router)#exit
```

Resumidamente:

- O comando `router rip` ativa a configuração do roteador para o RIP.
- O comando `network <ip_rede>` informa ao RIP quais redes conectadas ao roteador serão anunciadas pelo RIP.

Após tais comandos o roteador em questão começará a anunciar as rotas definidas pelo comando `network`.

Caso seja necessário desligar o RIP no roteador, basta utilizar o comando `no router rip`. Isso desliga o processo do RIP e também elimina todas as configurações do RIP.

O RIP nos roteadores CISCO não irão anunciar, por padrão, as rotas existentes em tais roteadores. Assim, é necessário utilizar o comando `network <ip_rede>`, para que o RIP anuncie as rotas. Neste comando `<ip_rede>` deve ser substituído pelo IP da rede a ser anunciada pelo protocolo RIP.

O comando `network` também irá iniciar o processo RIP e por padrão irá anunciar a rota por todas as interfaces de rede do roteador. Tais anúncios são feitos a cada 30 segundos.

O comando a seguir mostra as configurações do RIP, no Router2:

```
Router>show ip protocols
```

```
Routing Protocol is "rip"
```

```
Sending updates every 30 seconds, next due in 16 seconds
```

```
Invalid after 180 seconds, hold down 180, flushed after 240
```

```
Outgoing update filter list for all interfaces is not set
```

```
Incoming update filter list for all interfaces is not set
```

```
Redistributing: rip
```

```
Default version control: send version 1, receive any version
```

Interface	Send	Recv	Triggered	RIP	Key-chain
FastEthernet0/0	1	2	1		
Serial0/1/0	1	2	1		

```
Automatic network summarization is in effect
```

```
Maximum path: 4
```

```
Routing for Networks:
```

```
192.168.4.0
```

```
192.168.5.0
```

```
Passive Interface(s):
```

```
Routing Information Sources:
```

Gateway	Distance	Last Update
192.168.4.3	120	00:00:11

```
Distance: (default is 120)
```

```
Router>
```

O Rip está ativo!

Valores do RIP,
timers, etc

Versão do
RIP

Redes classful que estão
sendo anunciadas

Vizinho utilizado para atualização
do RIP

Já o comando a seguir mostra as rotas instaladas pelo RIP:

Router>**show ip route**

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

Gateway of last resort is not set

**Indica rota usando
RIP**

```
R 192.168.1.0/24 [120/2] via 192.168.4.3, 00:00:09, Serial0/1/0
R 192.168.2.0/24 [120/1] via 192.168.4.3, 00:00:09, Serial0/1/0
R 192.168.3.0/24 [120/1] via 192.168.4.3, 00:00:09, Serial0/1/0
C 192.168.4.0/24 is directly connected, Serial0/1/0
C 192.168.5.0/24 is directly connected, FastEthernet0/0
```

Router>

Configurando o roteador para utilizar o RIP-2

Como pode ser visto anteriormente o padrão do roteador é usar o RIP-1, mas é possível utilizar a versão 2 do RIP, através do comando `version 2`, no *prompt* de configuração do RIP. Exemplo:

```
Router(config)#router rip
Router(config-router)#version 2
Router(config-router)# ^Z
Router# show ip protocols
```

Com o último comando é possível verificar a alteração de RIP-1 para RIP-2.

Ativando o RIP-2, com o comando anterior, torna-se possível utilizar endereços de rede *classless* e as mensagens RIP agora são enviadas com a máscara de rede.

Da mesma forma que foi ativado o RIP-2 é possível desativar a auto sumarização, com o comando `no auto-summary`. A auto sumarização só tem efeito usando RIP-1 (*classful*). Desativando a auto sumarização torna-se possível ver todas as subredes, ou seja o RIP-2 coloca as subredes em suas tabelas. Atenção o RIP-2 deve ser ativado antes de desabilitar a auto sumarização.

Configurando interfaces RIP para o modo passivo

Por padrão o RIP envia pacotes RIP, de atualização para todas as interfaces de rede conectadas ao roteador. Todavia tais pacotes só devem ser enviados para interfaces de rede que estão conectadas à outros roteadores, ou seja, interfaces de redes conectadas a LANs, não devem receber tais pacotes por três motivos:

- Consumo da largura de banda: apesar dos pacotes RIP não consumirem muita largura de banda, o envio sucessivo (a cada 30 segundos) pode consumir um montante considerável da rede depois de um tempo. Além disso o RIP utiliza *broadcast* e *multicast* para atualizações e isso pode também afetar o desempenho de *switches*;
- Gasto de recursos: todos os dispositivos conectados em uma LAN devem processar os pacotes de atualização RIP até a camada de transporte, o que acaba consumindo recursos dos *hosts*.
- Risco a segurança: as mensagens RIP enviadas via *broadcast* podem representar um risco a segurança da rede, pois tais mensagens podem ser interceptadas e alteradas com o intuito de comprometer o bom funcionamento da rede.

Para evitar os problemas citados anteriormente é possível colocar as interfaces de rede de roteadores CISCO no modo passivo, com o comando `passive-interface <interface_rede>`. Isso previne que pacotes de atualização sejam enviados pela interface, contudo tal interface ainda poderá receber mensagens RIP vindas de outros roteadores por essa interface.

A seguir é apresentado um exemplo da sequência de comandos necessária para colocar uma interface no modo passivo:

```
Router(config)#router rip
Router(config-router)#passive-interface g0/0
Router(config-router)# end
Router# show ip protocols
```

Caso o comando `passive-interface` tenha sucesso aparecerá, na saída do comando `show ip protocols`, na seção “Passive Interface(s)”, a interface de rede em questão.

Também é possível deixar todas interfaces passíveis, por padrão, isso é feito com o comando `passive-interface default`. Desta forma, as interfaces que não devem ficar no modo passivo deverão ser ativadas utilizando-se o comando `no passive-interface`.

Propagando rota padrão via RIP

É possível fazer com que o RIP propague rota padrão. Por exemplo, se um roteador da rede estiver conectado à Internet, é possível configurar tal roteador como o roteador padrão para outros roteadores da rede.

Para propagar a rota padrão via RIP, o roteador de borda (*router edge*) deve ser configurado da seguinte maneira:

- Configure a rota padrão estática com o comando `ip route 0.0.0.0 0.0.0.0 exit-interface next-hop-ip`.
- Execute o comando `default-information originate` no roteador de borda. É esse comando que diz para o RIP propagar a rota padrão.

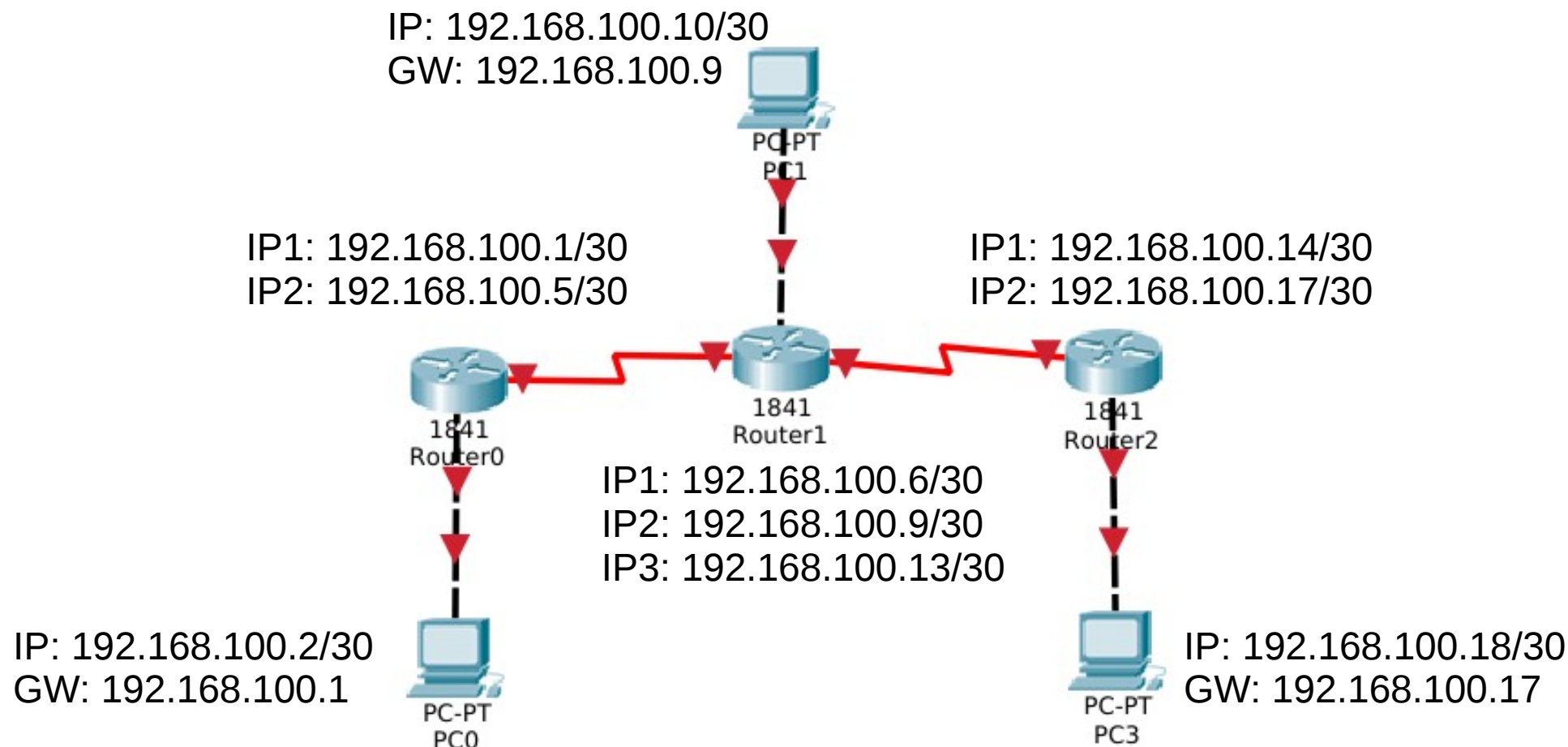
Exemplo:

```
Router>enable
Router#configure terminal
Router(config)#ip route 0.0.0.0 0.0.0.0 200.0.0.1
Router(config)#router rip
Router(config-router)#default-information originate
Router(config-router)#^Z
Router#
```

No exemplo o IP 200.0.0.1 faz referencia ao IP do roteador padrão, mas poderia ser a interface de rede de saída.

Atividade 1

Configure a rede a seguir da seguinte forma:



Ainda são requisitos do cenário: (i) As rotas entre os roteadores devem ser distribuídas via RIP (ii) Router1 deve fornecer rota padrão via RIP através do PC1; (iii) As interfaces dos roteadores, conectadas à PC1 e PC3, não devem enviar mensagens RIP, por motivos de segurança

OSPF (*Open Shortest Path First*)

O OSPF é um protocolo de roteamento intradomínio, tal como o RIP. Todavia o OSPF é do tipo Link-State. Tal como o nome sugere o OSPF é um protocolo aberto, o que significa que as especificações do protocolo são públicas.

Custos e métricas do OSPF

No OSPF, tal como no RIP o custo é calculado da origem para o destino. Entretanto, o custo de cada link (rede) pode ser dado por diferentes pesos: saltos (*hops*), vazão (*throughput*), tempo de viagem (*round-trip time*), confiança (*reability*), etc - o administrador é quem decide. Diferentes tipos de serviços (ToS - *Type of Service*) podem ter diferentes pesos.

Tabelas de roteamento do OSPF

Cada OSPF router pode criar uma tabela de roteamento após calcular a árvore de menor custo usando o algoritmo de Dijkstra. As tabelas de roteamento do RIP e do OSPF, basicamente são idênticas, a única diferença são os valores dos custos, já que o OSPF pode utilizar outras métricas. Contudo, se o OSPF utilizar apenas a medida de saltos, como custo, a árvore de menor custo, bem como a tabela de roteamento, serão idênticas ao RIP.

Áreas no OSPF

Ao contrário do RIP, que é indicado apenas para pequenas ASs, o OSPF foi projetado para ser executado em ASs de pequeno ou grande porte.

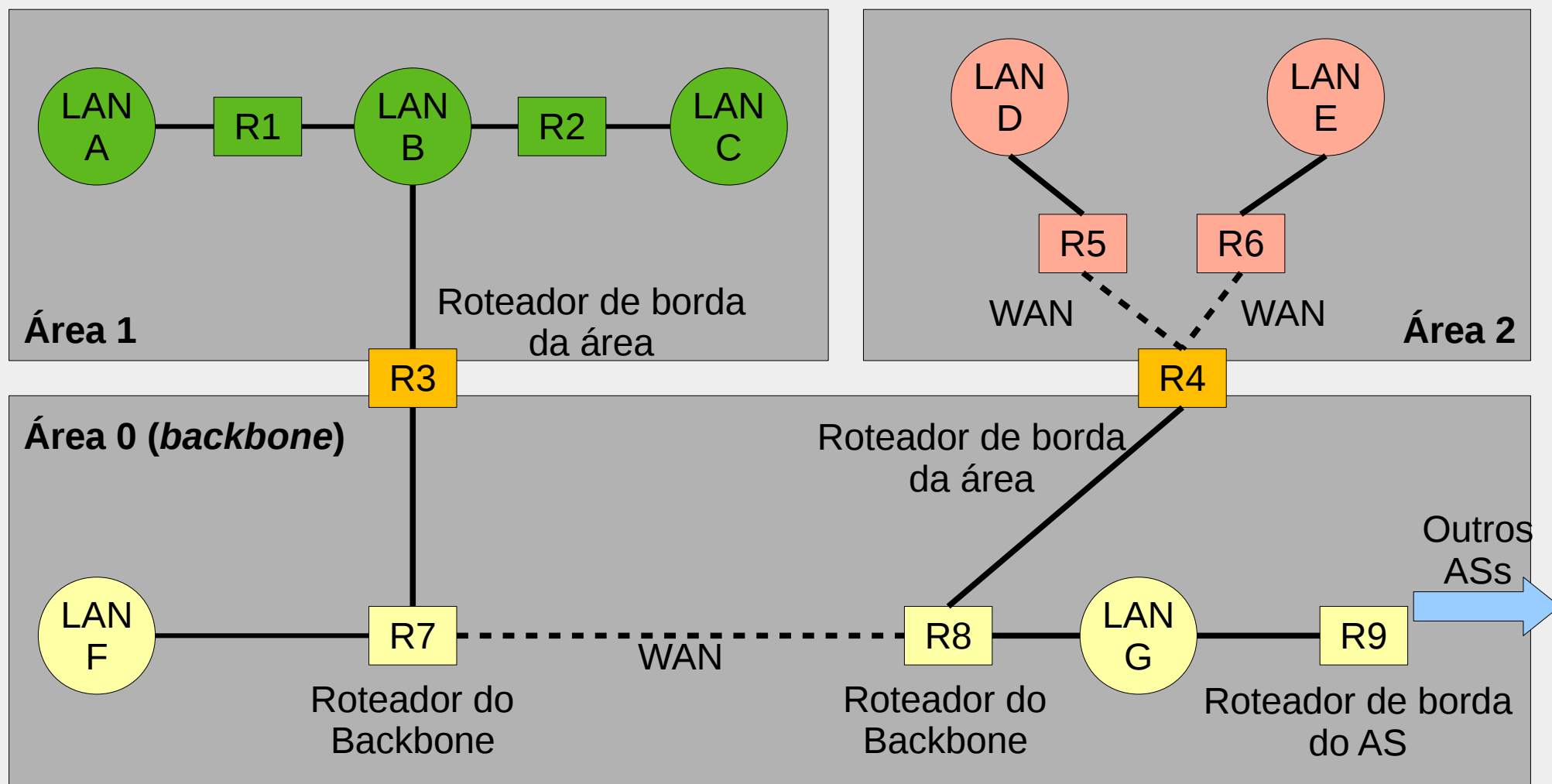
Entretanto, a árvore de menor custo, no OSPF requer que todos os roteadores inundem (*flooding*) todo AS com LSPs, para criar o LSDB global. Assim, embora isso não cause problemas em ASs pequenos, essa inundação de LSPs pode causar problemas em grandes ASs – devido a quantidade de fluxo de pacotes deste tipo. Para evitar tal problema, um AS pode ser dividido em pequenas seções chamadas de áreas (*areas*).

Cada área atua como um pequeno e independente domínio de inundação (*flooding*) LSPs. Em outras palavras, o OSPF usa outro nível de hierarquia no roteamento, sendo que o primeiro nível é o AS e o segundo é a área.

Então no OSPF um AS é dividido em áreas, contudo cada roteador precisa de informações a respeito dos estados dos links não somente de sua área, mas também de outras áreas. Por isso, uma área do AS deve ser designada como uma área de *backbone*, que é responsável por unir as demais áreas do AS.

Os roteadores na área do *backbone* são responsáveis por passar informações coletadas nas outras áreas. Para uma boa comunicação, cada área recebe uma identificação de área. A identificação do *backbone* é 0 (zero).

AS – Autonomous System



Descrevendo um pouco mais as áreas do OSPF

Uma área é uma generalização de uma rede individual. Fora da área, é possível até ver um destino, mas não a topologia desse destino. Essa característica ajuda a escalar o roteamento (ajuda em ambientes grandes).

É possível ir de uma área para outra, dentro do AS através do *backbone* de área. Todas as áreas são conectadas por *backbones*, possivelmente por túneis. Tal como ocorre com as outras áreas, a topologia da área do *backbone* não é visível fora do *backbone* (área 0).

Cada roteador conectado a duas ou mais áreas são chamados de *backbone* de áreas, e fazem parte do *backbone* (área 0). O principal objetivo dos roteadores de área é resumir os destinos de uma área para enviar tais destinos para outras áreas, nas quais ele está conectado. Tais informações incluem destinos, custos, mas não topologia. Passar o custo permite que roteadores em outras áreas saibam qual é o melhor roteador de borda para um destino fora da sua área. Não enviar os detalhes de uma topologia reduz o custo de rede e processamento do OSPF. Rotas para destinos fora da área sempre inicia com a instrução “vá para o roteador de borda”.

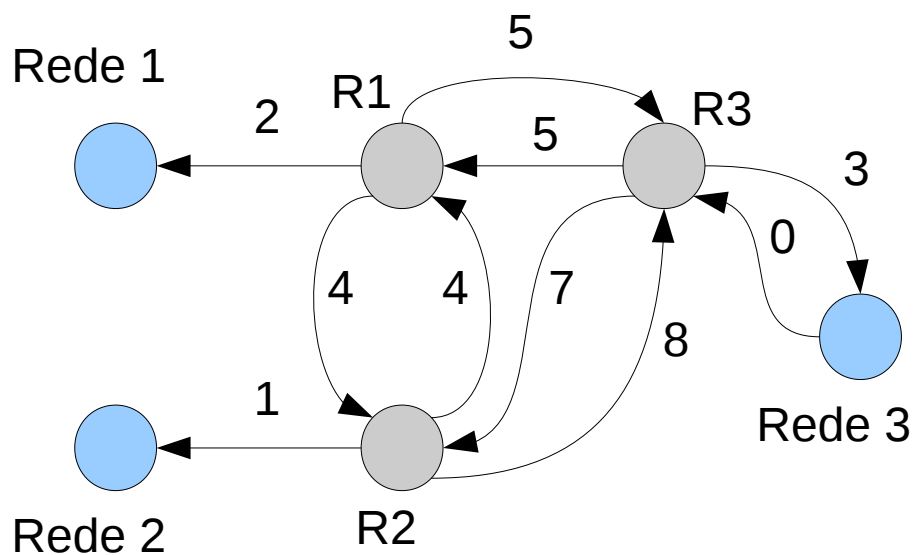
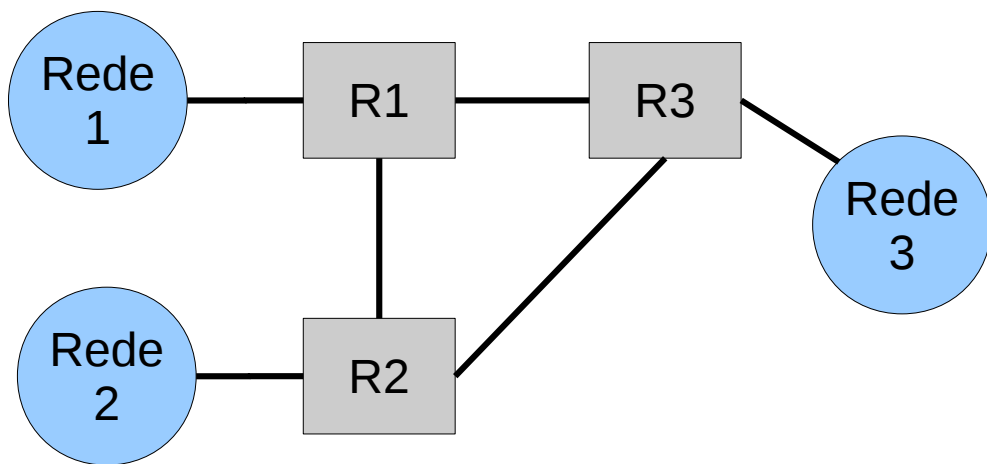
Os roteadores de borda do AS (*AS boundary router*), injetam dentro da área roteadores para destinos externos (fora da AS em questão). Então, rotas externas aparecem como destinos que podem ser alcançados através dos roteadores de borda do AS, com algum custo.

Mudanças de topologia são distribuídas para outras áreas no formato de vetor de distância. Em outras palavras esses roteadores apenas atualizam duas tabelas de roteamento e não precisam executar o algoritmos SPF.

Anúncio de estado de link

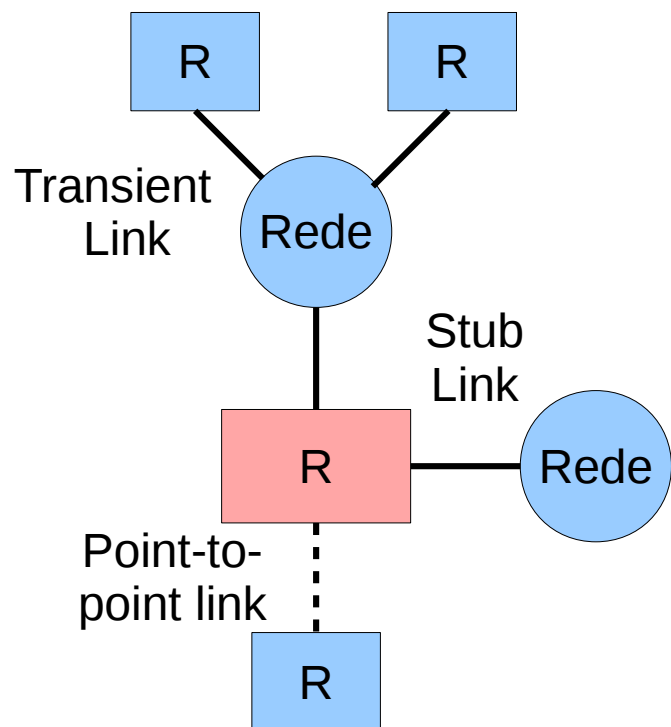
Como o OSPF é do tipo Link-State, tal protocolo requer que os roteadores avisem outros roteadores a respeito dos estados dos seus links, para a formação do LSDB.

No algoritmo Link-State, visto anteriormente, nós assumimos que cada nó do grafo era um roteador e cada aresta entre dois roteadores era uma rede. Todavia, na prática isso não é bem assim, pois é necessário anunciar a existência de várias entidades, os diferentes tipos de links que conectam cada nó aos seus vizinho e uma variedade de tipos de custos associados a cada link.



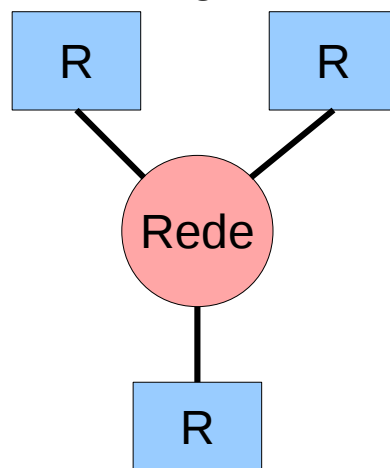
Então são necessários diferentes tipos de anúncios, para diferentes tipos e situações. É possível ter cinco tipos de anúncios de estados de link:

- *Router link* – anuncia um roteador como nó. O anúncio pode definir o endereço da rede e o custo.
- *Network link* – anuncia uma rede como um nó. Contudo uma rede não pode fazer anúncios de si mesma (é uma entidade passiva), assim um roteador é designado para fazer esse anúncio;
- *Summary link to network* – esse é um anúncio feito por um roteador de borda de área. Tal anúncio é a respeito do resumo dos links coletados pelo *backbone* e é utilizado para unir as áreas;
- *Summary link to AS* – Isso é feito por um roteador AS que anuncia o resumo dos link (*summary links*) de outros ASs para o backbone do AS em questão. Isso é útil para se ter informações de outras redes dentro do AS, e fará mais sentido quando discutirmos roteamento Inter-AS;
- *External link* – Também é realizado por um roteador AS para anunciar a existência de uma única rede fora do AS para o backbone, para ser disseminada entre as áreas.

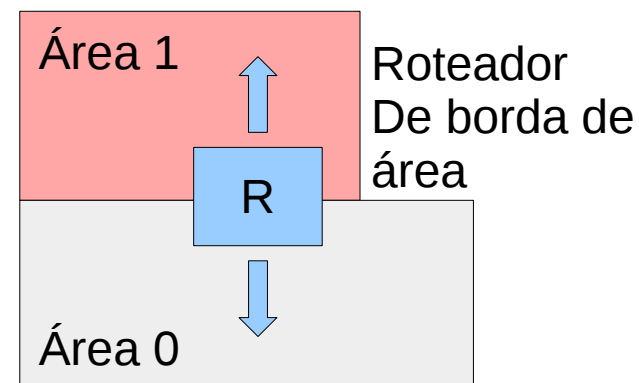


(1) Router link to AS

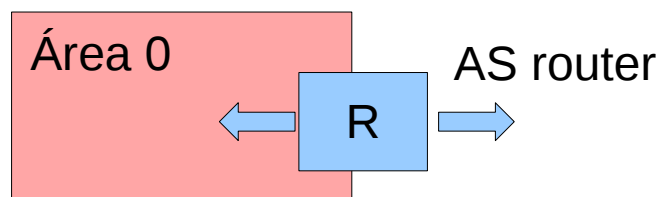
A rede é anunciada
Por um roteador
designado



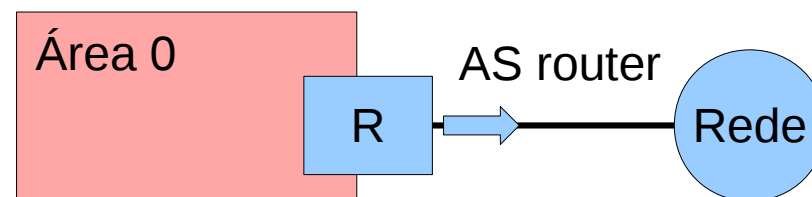
(2) Network Link



(3) Summary link to network



(4) Summary link to AS



(5) External link

Implementação do OSPF

O OSPF é um programa implementado na camada de rede, usando o IP para propagação.

Um datagrama IP levando uma mensagem OSPF terá o valor 89 no campo *protocol*.

Há duas versões do OSPF - versão 1 e 2. Contudo a maioria das implementações utilizam a versão 2. Há uma terceira versão para IPv6.

O OSPF é um protocolo bem complexo, por isso utiliza cinco diferentes tipos de mensagens e essas utilizam um cabeçalho comum (usado em todas mensagens).

Common
header

Version	Type	Message length
Source router IP address		
Area identification		
Checksum		Authentication Type
Authentication		

Também há um cabeçalho geral de link-state (usado em algumas mensagens).

LS age		E	T	LS type
LS ID				
Advertising router				
LS sequence number				
LS checksum		Length		

Os cinco tipos de mensagens são:

- *Hello* (tipo 1)- usada por um roteador para descobrir quem são os vizinhos e anunciar que ele está pronto.
- *Database description* (tipo 2) - é normalmente enviada em resposta a uma mensagem *Hello* e usada pelo novo roteador para adquirir o LSDB.
- *Link-state request* (tipo 3) - enviada por um roteador que deseja alguma informação de LS específica.
- *Link-state update* (tipo 4) - é a principal mensagem do OSPF, usada para construir o LSDB. Tal mensagem fornece o custo dos vizinhos do roteador. Há cinco mensagens (*router link*, *network link*, *summary link to network*, *summary link to AS border router* e *external link*).

- *Link-state acknowledgment* (tipo 5) – é usado para dar confiança no OSPF, cada roteador que recebe uma mensagem *LS update*, precisa confirmá-la.

O OSPF também fornecer mecanismos de autenticação, que evita que pacotes falsificados (*spoofing*) enviem informações falsas de roteamento. O que pode comprometer a segurança da rede.

Em resumo:

No OSPF, usando *flooding*, cada roteador informa todos outros roteadores na sua área a respeito de seus links para outros roteadores, redes e o custo desses links. Essa informação permite que cada roteador construa o grafo para sua área (ou áreas) e calcule a árvore de menor custo.

O *backbone* de área faz isso também, mas os roteadores de backbone, adicionalmente, aceitam informações vindas de roteadores de borda de área para calcular a melhor rota para cada roteador *backbone* para todos os outros roteadores. Tal informação é propagada de volta para os roteadores de borda de área, que anunciam isso em suas áreas. Usando essas informações, roteadores internos pode selecionar a melhor rota para o destino fora de suas áreas, incluindo o melhor roteador de saída para o *backbone*.

Implementando e configurando OSPF em roteadores CISCO

Protocolos de roteamento Link-State possuem a reputação de serem muito mais complexos de se configurar do que os de vetor de distância. Contudo, configurar as funcionalidades básicas dos protocolos de roteamento LS, não é muito difícil.

O OSPF é um protocolo de roteamento *classless* que usa o conceito de áreas para conseguir escalabilidade.

Tal como o RIP, as operações básicas do OSPF podem ser configuradas utilizando-se os comandos:

- `router ospf <process-id>` - que é o comando de configuração global;
- `network` - utilizado para anunciar as redes.

Então o OSPF versão 2 é habilitado utilizando-se o comando `router ospf <process-id>`, no qual `process-id` é um valor entre 1 e 65535, que é selecionado pelo administrador da rede. Tal valor é localmente significativo e não devem haver números iguais em uma rede (roteadores vizinhos).

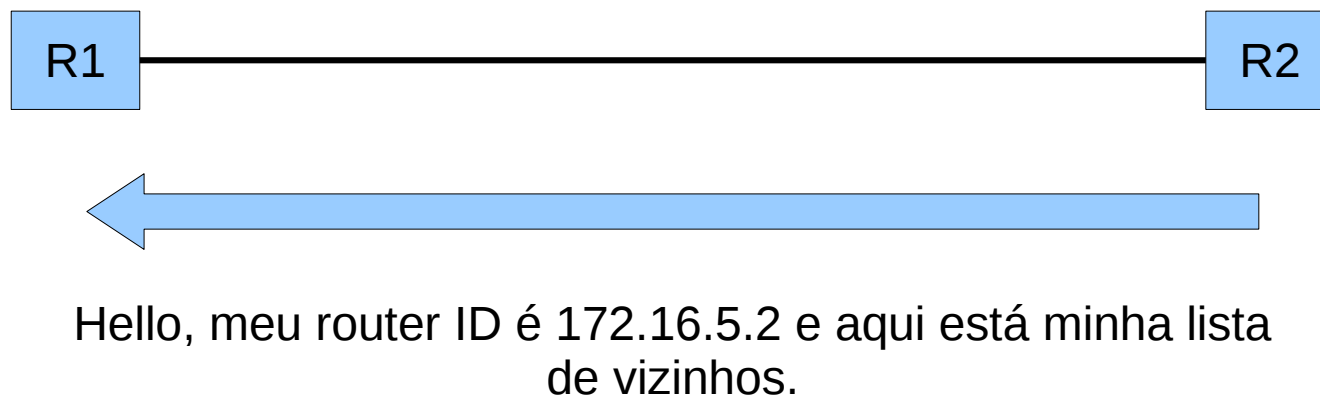
Identificação dos roteadores OSPF

Cada roteador precisa de um número de identificação, chamado de router ID, para participar de um domínio OSPF. O router ID pode ser definido pelo administrador ou ser gerado automaticamente pelo roteador.

Quando o OSPF é habilitado em uma interface do roteador, o roteador deve determinar se há outro OSPF vizinho no link, para tanto o roteador envia o pacote Hello contendo o router ID para todas as interfaces OSPF.

Um router ID é um endereço IP utilizado para identificar o roteador.

Quando um vizinho OSPF recebe um Hello com um router ID que ele não conhece, ele deve tentar estabelecer adjacência com esse roteador que deve estar inicializando.



No exemplo da figura anterior, R1 recebe o Hello e adiciona R2 na lista de vizinhos.

Quando um roteador recebe um Hello com uma lista de router IDs, o roteador sai do estado Init para o estado Two-Way.

A ação executada no estado Two-Way, depende do tipo de interconexão entre os roteadores adjacentes:

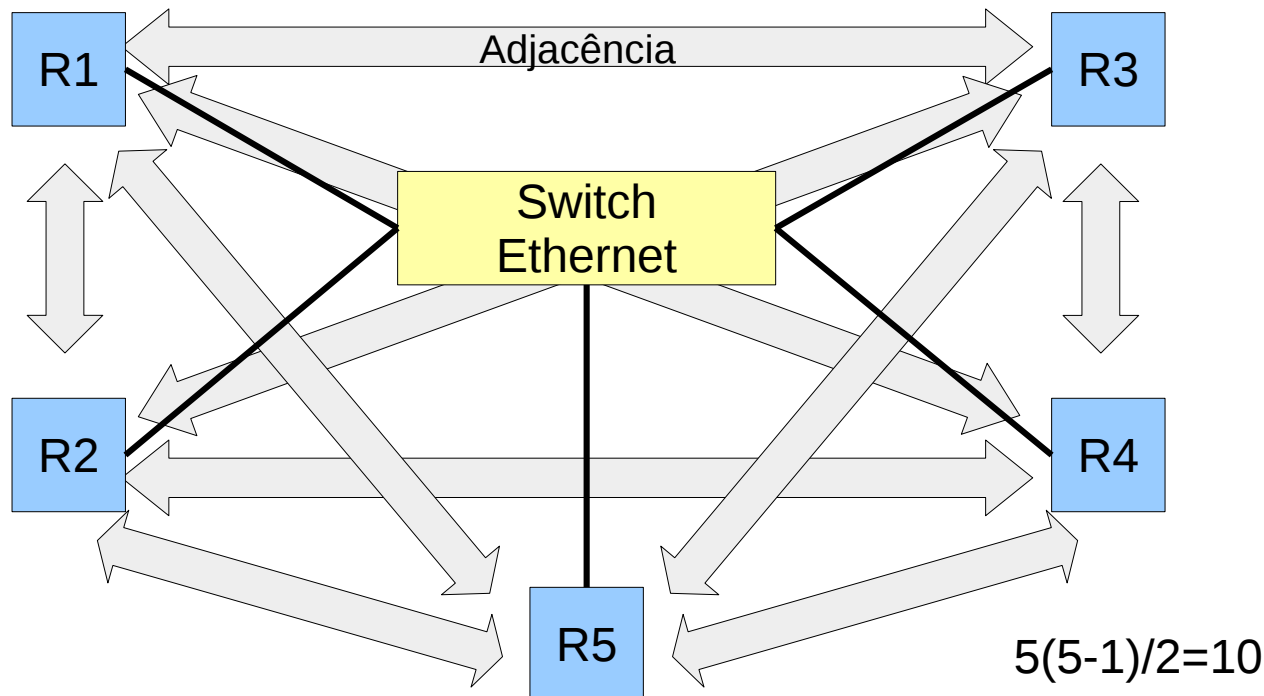
- Se os vizinhos adjacentes são interconectados por um link ponto-a-ponto, então eles podem imediatamente ir do estado Two-Way para a fase de sincronização de banco de dados.
- Se os vizinhos são conectados em uma rede multiponto/multiacesso (ex. Ethernet), antes de prosseguir é obrigatória a eleição de um roteador DR (*Designated Router*) e outro BDR (*Backup Designated Router*).

Pois, o uso de OSPF em links multiponto pode criar dois desafios devido ao *flooding* de pacotes de anúncios (LSA – *Link-State advertisements*):

- Criação de múltiplas adjacências: Redes multiponto podem potencialmente interconectar muitos roteadores OSPF em um mesmo link. Todavia não é necessário nem desejável criar adjacência com todos esses roteadores. Isso leva a criação/disseminação de um número excessivo de LSAs na mesma rede.

- Flooding excessivo de LSAs: roteadores Link-State inundam a rede com pacotes LSAs durante a inicialização do OSPF ou quando há mudanças na topologia da rede. Em redes multiponto essa inundação pode ser excessiva.

Para entender o problema com múltiplas adjacências, podemos observar a seguinte formula: Para cada numero de roteadores (determinado como n) em uma rede multiponto, há $n(n-1)/2$ adjacências.



Segundo a formula anterior, com 10 roteadores em uma rede multiponto seriam criadas 45 adjacências, 20 seria 190 e 100 roteadores teriam 4950 adjacências.

A solução para gerenciar o número de adjacências e a inundação de pacotes LSAs em rede multiponto é elegendo um DR (roteador designado).

O roteador eleito como DR será o ponto de coleta e distribuição para LSAs enviados e recebidos. Assim, todos os roteadores devem enviar pacotes para o DR, que da mesma forma deve responder e manter os outros roteadores atualizados a respeito de mudanças que ocorrem na rede. Note que sem o DR, todos os roteadores deveriam enviar mensagens (uma mesma) para todos os outros roteadores da rede, o que causaria *overhead*.

Um BDR (DR backup) também é eleito para caso de falhas. O BDR não executa nenhuma função enquanto o DR estiver ativo. Mas sempre recebe as mensagens OSPF enviadas.

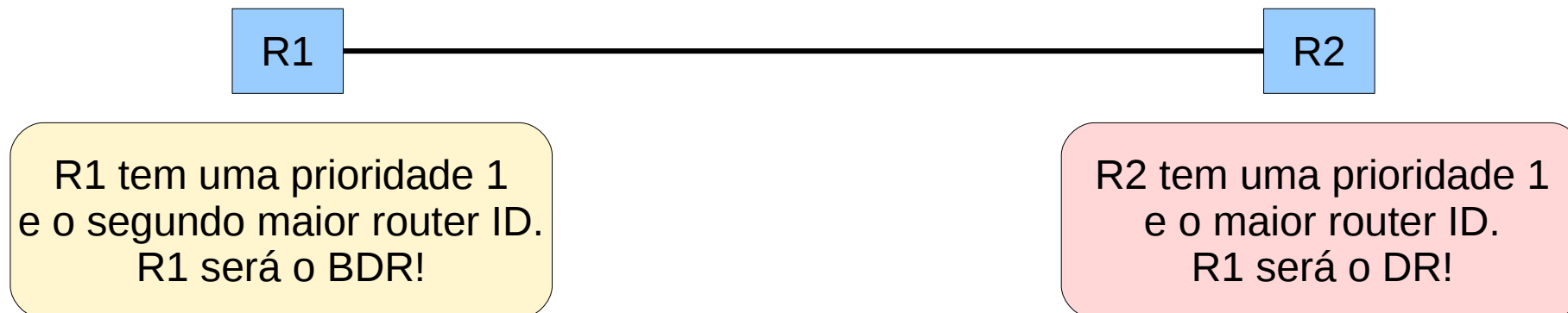
Todos os outros roteadores se tornam DROUTHERs, sendo que um DROUTHER é um roteador que não é nem DR e nem BDR.

Após o estado Two-Way os roteadores sincronizam o banco de dados e continuam o processo que envolve o OSPF (visto anteriormente, na teoria).

Assim o router ID é usado para:

- Identificar unicamente o roteador: tal número é usado para identificar o roteador em questão em relação aos outros roteadores, bem como é usado para identificar os pacotes desse roteador;
- Para a participação do DR: em uma rede multiponto (ex. Ethernet). Durante a eleição do DR o roteador com maior prioridade ganha a eleição. Se o número de prioridade não estiver configurado, ganhará a eleição o roteador com o maior router ID. O roteador com o segundo maior router ID será o BDR.

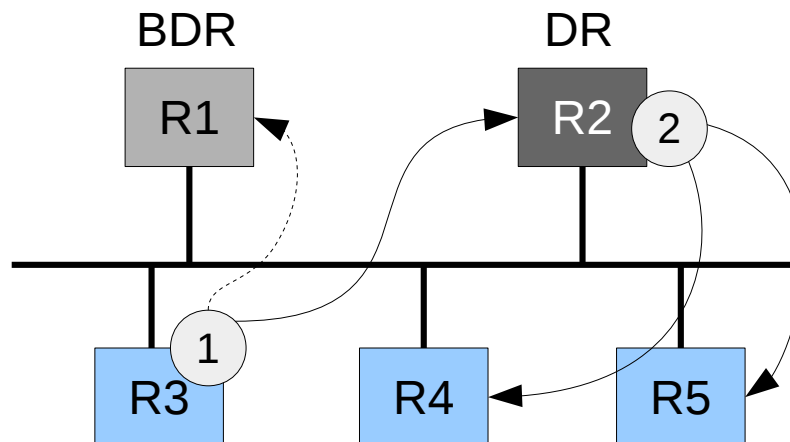
Eleição do DR e BDR:



Exemplo de como as mensagens são trocadas em redes multiponto com DR e BDR:

1 - Atualização para o DR.

2 – Atualização encaminhada para os vizinhos.



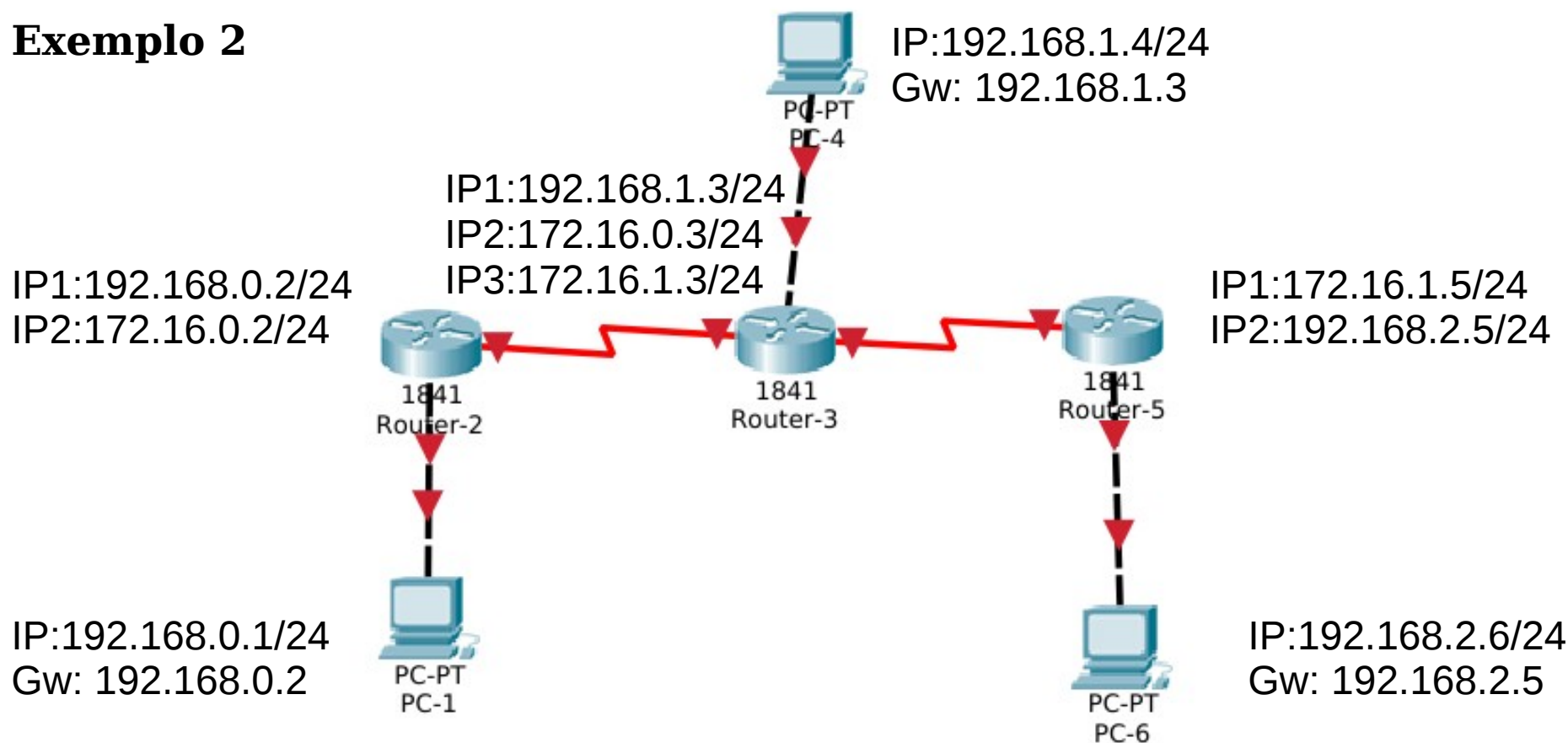
Os roteadores CISCO utilizam três critérios para determinar o Router ID, em ordem:

- 1) Foi passado algum Router ID, via linha de comando (`router-id <rid>`)? Se sim, utilize esse, caso contrário vá para o próximo critério. `<rid>` é um valor de 32 bits, tal como um IP. Esse é o método mais recomendado.
- 2) Se o roteador não não teve um Router ID configurado manualmente, o roteador escolhe o maior endereço IPv4 de qualquer interface de *loopback*. Essa é a segunda melhor alternativa.
- 3) Caso nenhuma das alternativas/critérios anteriores sejam utilizados, o roteador deve escolher o maior endereço IPv4 ativo de qualquer interface física. Essa é a menos recomendada.

No caso do IPv4 da interface física, esse não precisa um IP habilitado via OSPF, basta ser um IP de uma interface ativa.

O Router ID, apesar de ser um endereço IP (ter esse formato) não é incluído na tabela de roteamento, a menos que o roteamento OSPF escolha uma interface física ou de *loopback* declarada com o comando `network`.

Exemplo 2



No exemplo 2, tal como nos outros, devemos iniciar configurando todos os *hosts* do cenário. Aqui vamos apenas lembrar configurando Router-5:

```
Router>enable
Router#configure terminal
Router(config)#interface fa0/0
Router(config-if)#ip address 192.168.2.5 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#interface s0/1/0
Router(config-if)#ip address 172.16.1.5 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#end
Router#
```

Lembrando que os nomes das interfaces de rede podem mudar, conforme você criou o seu cenário no Packet Tracer CISCO.

Nos comandos anteriores basicamente entramos no modo de configuração do Router-5 e atribuímos IPs as duas interfaces de rede e pedimos para que essas fiquem ligadas.

Agora o próximo passo é configurar efetivamente o OSPF para anunciar as rotas na rede.

A seguir são apresentados os comandos para os três roteadores do cenário do Exemplo 2:

Router-2:

```
Router(config)#router ospf 20  
Router(config-router)#router-id 2.2.2.2  
Router(config-router)#network 192.168.0.0 0.0.0.255 area 0  
Router(config-router)#network 172.16.0.0 0.0.0.255 area 0  
Router(config-router)#end
```

Router-3:

```
Router(config)#router ospf 30  
Router(config-router)#network 192.168.1.0 0.0.0.255 area 0  
Router(config-router)#network 172.16.0.0 0.0.0.255 area 0  
Router(config-router)#network 172.16.1.0 0.0.0.255 area 0  
Router(config-router)#end
```

Router-5:

```
Router(config)#router ospf 50  
Router(config-router)#network 172.16.1.0 0.0.0.255 area 0  
Router(config-router)#network 192.168.2.0 0.0.0.255 area 0  
Router(config-router)#end
```

Bem, vamos iniciar a explicação tomando por base o Router-2.

O primeiro comando habilita o OSPF (`router ospf 20`), neste caso o roteador foi configurado com o *process-id* 20, como esse número deve ser único os roteadores Router-3 e Router-5 também ganharam números diferentes.

O segundo comando faz a configuração do Router-ID, que também deve ser único. No caso do Router-2 esse foi identificado com o ID 2.2.2.2 através do comando `router-id 2.2.2.2`. Tal como *process-id*, o Router-ID deve ser único. No caso de Router-3 e Router-5, não foi utilizado o comando Router-id, então tais roteadores receberam como Router-ID, um endereço da interface física. Lembrando que é aconselhável atribuir o Router-ID manualmente para evitar confusões, por exemplo, não sabemos quais IDs foram atribuídos a Router-3 e Router-5. Se dois roteadores forem configurados com o mesmo Router-ID, o roteador mostrará uma mensagem de erro com algo como “foi detectado *router-id* duplicado”.

O próximo passo é executar os comandos que adicionam ao OSPF as redes que esse deve anunciar, isso é feito através do comando `network`, no exemplo do Router-2 foram executados dois comandos, ou seja foram adicionadas duas redes:

```
network 192.168.0.0 0.0.0.255 area 0
network 172.16.0.0 0.0.0.255 area 0
```

Analisando em específico o primeiro comando `network` utilizado no OSPF (ver a seguir) e comparando com o mesmo comando só que no RIP, percebemos que há duas grandes diferenças:

1
2
`network 192.168.0.0 0.0.0.255 area 0`

- 1) Uma diferença bem sutil no comando é a máscara que está com o valor 0.0.0.255, sendo que no RIP era 255.255.255.0. Isso acontece, pois o OSPF utiliza não a máscara de rede convencional, mas sim uma máscara coringa (*wildcard mask*), que veremos melhor daqui a pouco, mas de forma geral dá um poder maior na hora de se configurar a rede.
- 2) No final do comando percebemos o uso da opção `area` com o valor 0. Isso é necessário devido ao fato do OSPF dividir o AS em áreas, tal como explicado anteriormente. Como neste cenário de rede do exemplo, foi configurado apenas uma única área (não há várias áreas), recomenda-se utilizar o valor 0 (valor da área de backbone), mas isso não é obrigatório.

Com todo o cenário do Exemplo 2, configurado é possível testar sua conectividade (ex. ping) e verificar mais a fundo o status da rede/rotas.

Por exemplo, executando o comando `show ip protocols` em Router-2:

Router>`show ip protocols`

Routing Protocol is "ospf 20"

Outgoing update filter list for all interfaces is not set

Incoming update filter list for all interfaces is not set

Router ID 2.2.2.2

Number of areas in this router is 1. 1 normal 0 stub 0 nssa

Maximum path: 4

Routing for Networks:

192.168.0.0 0.0.0.255 area 0

172.16.0.0 0.0.0.255 area 0

Routing Information Sources:

Gateway	Distance	Last Update
2.2.2.2	110	00:28:35
192.168.1.3	110	00:28:35
192.168.2.5	110	00:28:35

Distance: (default is 110)

Router>

Identificação do processo

Identificação do roteador

Redes anunciadas

Vizinhos

(o próprio roteador
Aparece na lista de vizinhos).

Também é possível utilizar o comando `show ip ospf`, para verificar algumas informações do OSPF:

```
Router>show ip ospf
```

```
Routing Process "ospf 20" with ID 2.2.2.2
```

```
Supports only single TOS(TOS0) routes
```

```
Supports opaque LSA
```

```
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
```

```
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
```

```
Number of external LSA 0. Checksum Sum 0x000000
```

```
Number of opaque AS LSA 0. Checksum Sum 0x000000
```

```
Number of DCbitless external and opaque AS LSA 0
```

```
Number of DoNotAge external and opaque AS LSA 0
```

```
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
```

```
External flood list length 0
```

```
Area BACKBONE(0)
```

```
Number of interfaces in this area is 2
```

```
Area has no authentication
```

```
SPF algorithm executed 12 times
```

```
Area ranges are
```

```
Number of LSA 3. Checksum Sum 0x021aeb
```

```
Number of opaque link LSA 0. Checksum Sum 0x000000
```

```
Number of DCbitless LSA 0
```

```
Number of indication LSA 0
```

```
Number of DoNotAge LSA 0
```

```
Flood list length 0
```

```
Router>
```

O comando `show ip route`, apresenta as rotas conectadas e obtidas via OSPF:

Router#`show ip route`

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

Gateway of last resort is not set

```

172.16.0.0/24 is subnetted, 2 subnets
C      172.16.0.0 is directly connected, Serial0/1/0
O      172.16.1.0 [110/128] via 172.16.0.4, 00:46:12, Serial0/1/0
C      192.168.0.0/24 is directly connected, FastEthernet0/0
O      192.168.1.0/24 [110/65] via 172.16.0.4, 00:46:12, Serial0/1/0
O      192.168.2.0/24 [110/129] via 172.16.0.4, 00:46:12, Serial0/1/0
  
```

Router#



Rotas OSPF

O comando `show ip ospf interface`, ainda mais informações:

```
Router#show ip ospf interface
```

```
FastEthernet0/0 is up, line protocol is up
```

```
Internet address is 192.168.0.2/24, Area 0
```

```
Process ID 20, Router ID 2.2.2.2, Network Type BROADCAST, Cost: 1
```

```
Transmit Delay is 1 sec, State DR, Priority 1
```

```
Designated Router (ID) 2.2.2.2, Interface address 192.168.0.2
```

```
...
```

```
Serial0/1/0 is up, line protocol is up
```

```
Internet address is 172.16.0.2/24, Area 0
```

```
Process ID 20, Router ID 2.2.2.2, Network Type POINT-TO-POINT, Cost: 64
```

```
Transmit Delay is 1 sec, State POINT-TO-POINT, Priority 0
```

```
No designated router on this network
```

```
No backup designated router on this network
```

```
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
```

```
Hello due in 00:00:04
```

```
Index 2/2, flood queue length 0
```

```
Next 0x0(0)/0x0(0)
```

```
Last flood scan length is 1, maximum is 1
```

```
Last flood scan time is 0 msec, maximum is 0 msec
```

```
Neighbor Count is 1 , Adjacent neighbor count is 1
```

```
Adjacent with neighbor 192.168.1.3
```

```
Suppress hello for 0 neighbor(s)
```

```
Router#
```

Máscara Coringa (*Wildcard Mask*)

Como visto anteriormente o OSPF utiliza a combinação de endereço de rede com a máscara curinga para habilitar o OSPF nas interfaces.

O OSPF é *classless*, então o OSPF deve utilizar máscara para saber qual parte do IP representa rede e qual parte representa host (*net-id/host-id*). Contudo, ao invés de utilizar a máscara tradicional, o OSPF utiliza a máscara curinga que é mais flexível se comparada com a máscara padrão (máscara de rede/subrede), o que dá mais possibilidades durante a configuração das rotas.

De forma geral a *wildcard mask* é o inverso da máscara de subrede configurada na interface de rede. Contudo, é importante saber que a máscara de rede e a máscara curinga não estão diretamente relacionadas uma com a outra.

A máscara de rede deve identificar o prefixo da rede e o sufixo do host no IP de forma contígua. Por exemplo, na máscara de rede tradicional, não é possível intercalar zeros (000...) e uns (111...) na máscara. Sendo que um (1) representa rede e zero (0) representa host.

Já a máscara coringa permite esse tipo de prática, intercalar na máscara zeros e uns, para representar rede ou host e qualquer bit da máscara/IP.

Assim, tal como uma máscara tradicional, uma máscara coringa é um número de 32bits, separados em quatro octetos por pontos, tal como um IPv4.

A máscara coringa é utilizada para determinar quais bits do IP, que está sendo combinado com a máscara, são relevantes e quais são irrelevantes. A regra é:

- Bit 0 na máscara coringa: O bit equivalente do IP deve corresponder com o bit que está sendo analisado. Ou seja, durante a análise de uma rota, por exemplo, o bit do IP de destino deve combinar com o bit do IP da rede/rota.
- Bit 1 na máscara coringa: Ignore o valor do bit correspondente no endereço. Pois, o bit equivalente é irrelevante e pode assumir qualquer valor.

IP host:	11000000.10000000.00000001.00000001
IP rede:	11000000.10000000.00000001.00000000
Máscara coringa:	00000000.00000000.00000000.11111111

Observe, que normalmente parece que a máscara coringa é o contrário da máscara padrão, isso acontece pois na máscara padrão é como se fosse ao contrário, o que são bits relevantes e não relevantes, pois:

- Bit 1 na máscara padrão representa rede, que normalmente é o que importa.
- Bit 0 na representa host, ou seja, a parte do IP que normalmente é “ignorada”.

A relação entre as máscaras é tão íntima que é comum usar a máscara padrão para determinar a máscara coringa. Vamos ver um exemplo: imagine que temos a rede 192.168.1.0 com a máscara de rede 255.255.255.0. Agora queremos anunciar essa rota no OSPF que utiliza a máscara coringa! Bem, o que podemos fazer é o seguinte:

	255.255.255.255
Máscara de rede:	-255.255.255.000
	<hr/>
Máscara coringa:	000.000.000.255

No exemplo, a máscara de rede 255.255.255.0 é subtraída de 255.255.255.255, o que dá o resultado 0.0.0.255. Desta forma, 192.168.10.0/24 é 192.168.10.0 com a máscara coringa 0.0.0.255.

Outro exemplo, dada a rede 192.168.10.64/26, qual seria a máscara coringa equivalente? É só subtrair 255.255.255.255 de 255.255.255.192 (que é a máscara de rede /26).

	255.255.255.255
Máscara de rede:	-255.255.255.192
	<hr/>
Máscara coringa:	000.000.000.063

Fazendo o calculo a máscara coringa 0.0.0.63 é equivalente à máscara de rede 255.255.255.192 (/26 na notação CIDR).

Entretanto, como já foi descrito anteriormente a máscara coringa não é somente o contrário da máscara de rede. Ela é mais poderosa, por exemplo: dados os seguintes valores:

IP host:	11000000.10000000.00000001.00000001
IP rede:	11000000.10000000.00000001.00000000
Máscara coringa:	00000000.00000000.00000000.11111110

Note na máscara coringa utilizada anteriormente, que há um 0 (zero) no último bit da máscara coringa, o que significa que esse bit deve ser equivalente no IP do host, sendo analisado, com o IP da rede (os bits que deve ser equivalentes estão em vermelho). Assim, tal IP não combinaria com a rede.

Ainda analisando o exemplo anterior, note também que só combinaria com o IP da rede IPs de *hosts* que iniciam com os octetos 192.128.1.X, no qual o X deve obrigatoriamente ser um número par. Por exemplo, o IP 192.128.1.2, combina com o IP da rede, enquanto o IP 192.128.1.3, não combina:

IP host:	11000000.10000000.00000001.00000010
IP rede:	11000000.10000000.00000001.00000000
Máscara coringa:	00000000.00000000.00000000.11111110
IP host:	11000000.10000000.00000001.00000011
IP rede:	11000000.10000000.00000001.00000000
Máscara coringa:	00000000.00000000.00000000.11111110

É muito importante observar que jamais seria possível fazer esse controle (*host* par ou ímpar) mostrado anteriormente utilizando uma máscara de rede padrão, pois não seria possível intercalar zeros e uns na mesma.

Assim, a máscara coringa dá mais flexibilidade durante a configuração de rede/rota no OSPF e é utilizada em alguns casos, por questão de melhor gerência ou segurança.

Modificando o Router-ID

Para modificar o ID de um roteador basta executar o comando `clear ip ospf process`, depois de alterar o ID do roteador. Exemplo, vamos alterar o router-ID do Router-5 do exemplo anterior:

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 50
Router(config-router)#router-id 5.5.5.5
Router(config-router)#Reload or use "clear ip ospf process" command, for this to take effect
Router(config-router)#end
```

```
Router#clear ip ospf process
Reset ALL OSPF processes? [no]: yes
14:09:46: %OSPF-5-ADJCHG: Process 50, Nbr 192.168.1.3 on Serial0/1/0 from FULL to DOWN, Neighbor Down: Adjacency forced to reset
14:09:46: %OSPF-5-ADJCHG: Process 50, Nbr 192.168.1.3 on Serial0/1/0 from FULL to DOWN, Neighbor Down: Interface down or detached

14:10:04: %OSPF-5-ADJCHG: Process 50, Nbr 192.168.1.3 on Serial0/1/0 from LOADING to FULL, Load
Router#show ip ospf
```

Usando a interface de loopback como Router-ID

Para determinar o Router-ID utilizando a interface de *loopback*, primeiramente não se deve dar o nome manualmente, via comando `router-id`. Lembrando que sem usar o comando `router-id`, a segunda opção de nome, por padrão, é o nome de *loopback*.

Então, configurar a interface de *loopback* com um IP e máscara de rede.
Exemplo:

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface loopback 0
%LINK-5-CHANGED: Interface Loopback0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed
state to up
Router(config-if)#ip address 3.3.3.3 255.255.255.255
Router(config-if)#end
Router#
```

Lembrando que se for uma alteração de nome tem que executar o comando `clear ip ospf process`, para que o Router-ID passe a valer.

Configurar a interface OSPF no modo passivo

Tal como foi feito no RIP, também é possível configurar interfaces de rede no modo passivo no OSPF. Inclusive o comando (`passive-interface`) e os objetivos (segurança, evitar overhead, etc) são os mesmos.

Exemplo:

```
Router(config)#router rip
Router(config-router)#passive-interface g0/0
Router(config-router)# end
Router# show ip protocols
```

Custo dos links no OSPF

Originalmente o OSPF tem custos que devem ser configurados manualmente pelos administradores das redes. Contudo muitas empresas utilizam um custo automático padrão, tal como 1, o que torna o OSPF muito similar ao RIP, nesta questão de custo.

Desta forma, para se diferenciar do RIP e proporcionar custos mais coerentes, a CISCO decidiu utilizar uma métrica de custo baseada na largura de banda de cada interface de rede.

Para roteadores CISCO, o custo de uma interface é inversamente a largura de banda da interface. Assim, uma grande largura de banda significa um baixo custo. Uma pequena largura de banda irá gerar um link com custo alto. Nessa linha de pensamento, um link Ethernet de 10Mbps tem um custo maior do que um link de 100Mbps.

A fórmula para calcular o custo OSPF é:

$$\text{Custo} = \text{largura de banda de referencia} / \text{largura de banda da interface}$$

Sendo que a largura de banda de referência é 100.000.000, assim a fórmula é:

$$\text{Custo} = 100.000.000 / \text{largura de banda da interface}$$

Valor padrão para interfaces de rede em roteadores CISCO:

Tipo de interface	Referencia em bps	Largura de banda bps	Custo
10 Gigabit Ethernet (10Gbps)	100.000.000	10.000.000.000	1
1 Gigabit Ethernet (1Gbps)	100.000.000	1.000.000.000	1
Fast Ethernet (100Mbps)	100.000.000	100.000.000	1
Ethernet (10Mbps)	100.000.000	10.000.000	10
Serial (1.544Mbps)	100.000.000	1.544.000	64
Serial (128Kbps)	100.000.000	128.000	781
Serial (64Kbps)	100.000.000	64.000	1562

Note que todos links Ethernet da tabela possuem a mesmo valor, pois o custo OSPF é inteiro. Isso ocorre pois, a referencia de largura de banda foi criada com para links Fast Ethernet, mas hoje temos links mais velozes.

É possível verificar o custo de um link com o comando `show ip route`: (comando executado no Router-5 do exemplo anterior):

```
Router>show ip route
```

```
...
```

```
O    172.16.0.0 [110/128] via 172.16.1.4, 02:06:50, Serial0/1/0
O    192.168.0.0/24 [110/129] via 172.16.1.4, 02:06:40, Serial0/1/0
O    192.168.1.0/24 [110/65] via 172.16.1.4, 02:06:50, Serial0/1/0...
```

Ajustando referencia de largura de banda

Como dito anteriormente o OSPF utiliza como valor de referencia 100Mbps (100.000.000 bps), o que dá custo 1 para interfaces Fast Ethernet, mas o mesmo valor é obtidos para interfaces com largura de banda maior que isso, tal como 1 Gigabit Ethernet ou superior, e isso é um problema, pois pode gerar injustiças na escolha do link.

Para resolver o problema citado anteriormente os roteadores CISCO permitem alterar tal valor de referencia para o calculo do custo, com o comando `auto-cost reference-bandwidth <novo_custo_referencia>`. Sendo que o novo custo de referencia é dado em Mbps.

Por exemplo para colocar um valor de referencia compatível com link com largura de banda de 10Gbps podemos usar o seguinte comando:

```
Router(config)#router ospf 50
Router(config-router)#auto-cost reference-bandwidth 10000
```

% OSPF: Reference bandwidth is changed.

Please ensure reference bandwidth is consistent across all routers.

```
Router(config-router)# end
```

Agora, podemos ver o novo custo com o comando:

```
Router#show ip route
```

```
...
```

```
O    172.16.0.0 [110/6540] via 172.16.1.4, 00:00:16, Serial0/1/0
O   192.168.0.0/24 [110/6541] via 172.16.1.4, 00:00:16, Serial0/1/0
O   192.168.1.0/24 [110/6477] via 172.16.1.4, 00:00:16, Serial0/1/0
```

Essa alteração não afeta a largura de banda da interface, é apenas para o calculo de custos do OSPF. Os novos valores de custo para os links são:

Tipo de interface	Referencia em bps	Largura de banda bps	Custo
10 Gigabit Ethernet (10Gbps)	10.000.000.000	10.000.000.000	1
1 Gigabit Ethernet (1Gbps)	10.000.000.000	1.000.000.000	10
Fast Ethernet (100Mbps)	10.000.000.000	100.000.000	100
Ethernet (10Mbps)	10.000.000.000	10.000.000	1000
Serial (1.544Mbps)	10.000.000.000	1.544.000	6477
Serial (128Kbps)	10.000.000.000	128.000	78125
Serial (64Kbps)	10.000.000.000	64.000	156250

Também, dá para alterar a largura de banda da interface, com o comando: `bandwidth <valor_da_largura_banda_kbps>`, neste caso é necessário estar no modo de configuração da interface de rede. Tal comando não altera a velocidade da interface é só para o calculo do OSPF.

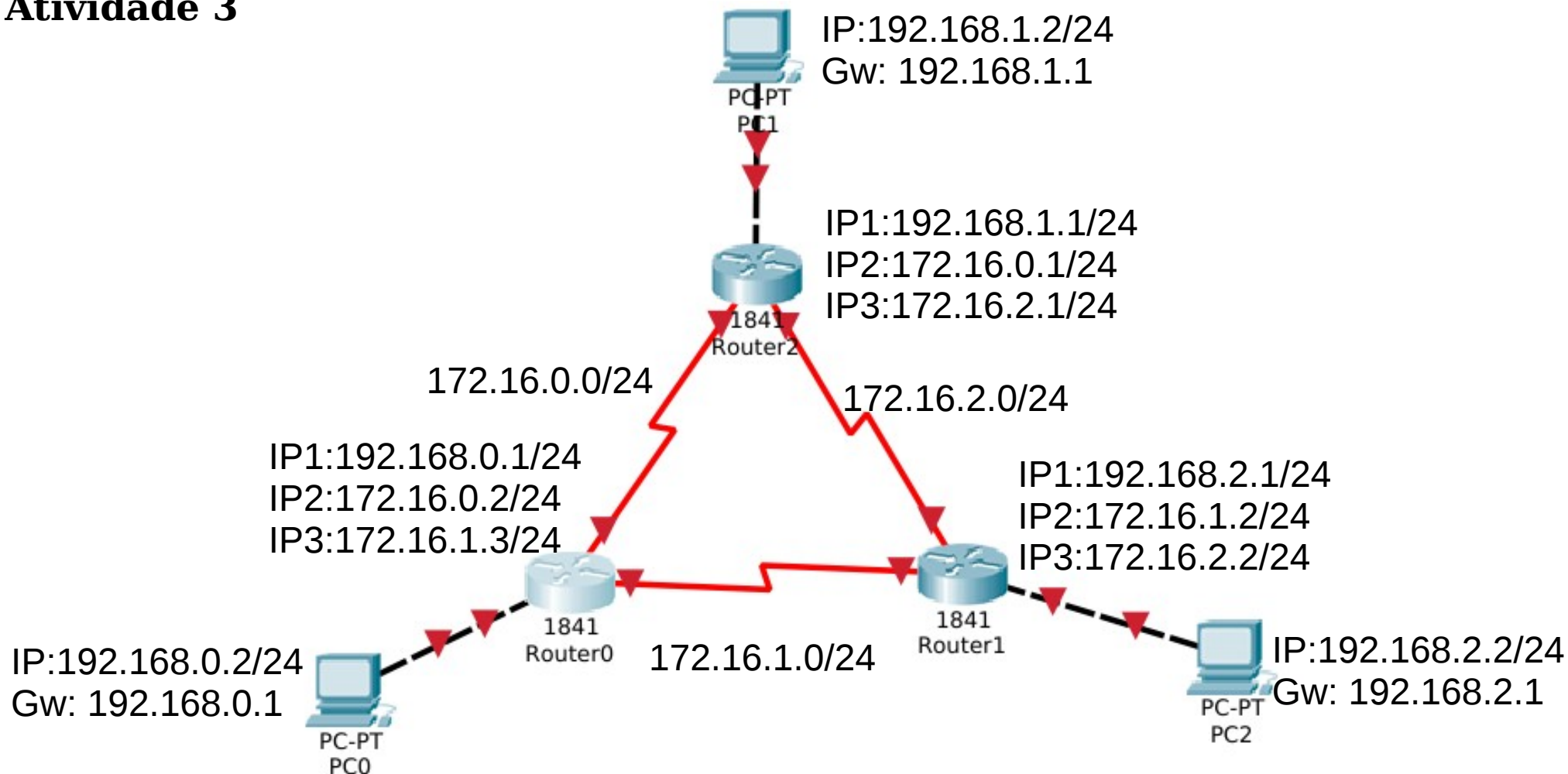
Por fim, dá para passar o custo manualmente, desconsiderando a largura de banda da interface. Isso é feito através do comando `ip ospf cost <valor>`. Uma vantagem desse método é que o roteador não tem que realizar cálculos. Contudo a coerência do custo fica todo a cargo do administrador, que pode cometer erros e causar problemas na rede. Esse método é indicado quando o roteador CISCO está conectado com vizinhos OSPF, de outros fabricantes e as métricas de custos entre esses são distintas. Desta forma, o administrador pode configurar métricas manualmente para equalizar tais custos e criar uma rede mais justa.

Um exemplo de ajuste manual seria o comando:

```
Router(config)#int s0/1/0
Router(config-if)#no bandwidth
Router(config-if)#ip ospf cost 15000
Router(config-if)#end
Router#show ip ospf interface
```

O último comando apresenta o custo nas interfaces.

Atividade 3



Configure o ambiente de rede de forma que esse utilize roteamento dinâmico via OSPF. Você também deve setar o custo dos links manualmente de forma que os custos sejam crescentes, na seguinte ordem de link/rede: 172.16.0.0, 172.16.2.0 e 172.16.1.0.

Referência:

FOROUZAN, Behrouz A. Data Communications and Networking. 5ª Edição. 2013.

TANENBAUM, Andrew S. Computer Networks. 5ª Edição. 2011.

CISCO. Routing and Switching Essentials Companion Guide. 3ª Edição. 2014.

COMER, Douglas E. Interligação de Redes em TCP/IP Volume 1. 5ª Edição. 2006.

FILIPPETTI, Marco A. CCNA 4.0 Guia Completo de Estudo. 2006.

ODOM, Wendell. CCENT/CCNA ICND1 Guia Oficial de Certificação do Exame. 2ª Edição. 2008.

(https://www.youtube.com/watch?v=aJ_2c9NVC1c)

<http://labcisco.blogspot.com/2015/01/como-funcionam-as-wildcard-masks.html>

Fim!