Experiências Laboratoriais sobre Encaminhamento na Internet

Carlos Lopes, Paulo Ferreira, Rui Valadas

Resumo **- Este artigo descreve um conjunto de experiências laboratoriais que foram concebidas para ilustrar as características mais importantes dos protocolos de encaminhamento utilizados nas redes IP.**

Abstract **- This paper describes a set of laboratory experiments designed to illustrate the main characteristics of the routing protocols used in IP networks.**

I. INTRODUÇÃO

Este artigo descreve um conjunto de experiências laboratoriais que foram concebidas para ilustrar as características mais importantes dos protocolos de encaminhamento utilizados nas redes IP. Os protocolos de encaminhamento estudados foram o RIP (*Routing Information Protocol*), o OSPF (*Open Shortest Path First*), o IGRP (*Inter Gateway Routing Protocol*), o EIGRP (*Enhanced Inter Gateway Routing Protocol*), o PIM (*Protocol Independent Multicast*), o EGP (*Exterior Gateway Protocol*) e o BGP (*Border Gateway protocol*).

As experiências foram realizadas no Laboratório de Comunicações do Departamento de Electrónica e Telecomunicações da Universidade de Aveiro. Este laboratório, ainda em fase de instalação, tem como missão constituir uma infraestrutura de suporte ao ensino das Redes e Sistemas de Telecomunicações, no âmbito da formação oferecida pelo DET-UA. Espera-se que o carácter experimental de que o ensino destas matérias poderá vir a ser dotado venha enriquecer significativamente a qualidade do mesmo.

As experiências utilizaram quatro routers Cisco 2514, cada um com duas interfaces Ethernet. Cada experiência incluiu os seguintes passos: definição da topologia da rede, programação dos *routers*, estudo do comportamento do protocolo usando os meios de monitorização proporcionados pelo *software* dos *routers.* Algumas experiências recorreram a um analisador de protocolos instalado num computador pessoal, tornando possível a visualização do conteúdo dos pacotes enviados para a rede pelos vários protocolos. Algumas experiências utilizaram também um computador pessoal com sistema operativo Windows NT configurado como router.

No Apêndice A é feita uma listagem completa das experiências efectuadas. A descrição de todas as experiências pode ser encontrada em [1].

II. ENCAMINHAMENTO NA INTERNET

A Internet está estruturada em *sistemas autónomos*. Um sistema autónomo é constituído por um conjunto de redes IP debaixo de uma mesma administração. O sistema autónomo define um domínio de encaminhamento. O encaminhamento da informação dentro dos sistemas autónomos é assegurado por protocolos de encaminhamento ditos *internos* (ex. RIP e OSPF). Para o encaminhamento entre sistemas autónomos são utilizados protocolos ditos *externos* (ex. EGP e BGP). Podem distinguir-se duas categorias de protocolos de encaminhamento:

– *Distance vector protocols* (ex.: RIP, IGRP) – Este tipo de protocolos de encaminhamento caracteriza-se por utilizar o algoritmo de percursos mínimos (*shortest path*) distribuído e assíncrono de Bellman-Ford. Neste caso, cada nó tem apenas conhecimento do nó vizinho que deverá ser utilizado para atingir um determinado destino.

– *Link state routing protocols* (ex.: OSPF) – Este tipo de protocolos de encaminhamento é caracterizado por ser mantido um mapa com a topologia da rede em cada nó, que é idêntico para todos os nós de uma mesma rede. É com base neste mapa que são calculados os percursos mínimos para todos os destinos, utilizando o algoritmo de Dijkstra.

III. DESCRIÇÃO DOS PROTOCOLOS DE ENCAMINHAMENTO

Protocolo RIP

O protocolo RIP (*Routing Information Protocol*), é o mais antigo dos protocolos estudados, mas ainda o mais usado hoje em dia. Para este facto contribuíram a sua simplicidade e facilidade de manutenção, para além dos custos associados a instalar numa rede um novo protocolo de encaminhamento (quer custos financeiros, quer de recursos humanos, temporais ou de possível inactividade da rede aquando da instalação de um novo protocolo de encaminhamento).

Cada nó ao ser inicializado contém apenas uma entrada na sua tabela de encaminhamento: a distância a ele próprio, que por convenção é 0. O nó ignora totalmente a topologia da rede onde está inserido. A sua

primeira acção é enviar através de todas as suas ligações a sua tabela de encaminhamento (neste caso muito simples).

Os seus nós vizinhos, ao receberem esta informação, irão acrescentar mais uma entrada nas suas tabelas com distância igual à distância recebida mais 1 (a distância ao nó vizinho), e a ligação por onde atingir o nó em questão. Seguidamente fazem o mesmo, isto é, enviam a sua tabela para os nós vizinhos (nesta altura a tabela tem duas entradas: a distância a eles próprios, zero, e a distância ao nó inicial, um). Por sua vez, os nós vizinhos farão o mesmo e assim trocarão a informação das distâncias indefinidamente.

Como o comportamento é idêntico para todos os nós, após várias trocas de mensagens, ter-se-á atingido um estado em que todos os nós possuem informação sobre as distâncias e ligações por onde enviar a informação para qualquer dos nós restantes. De notar que os nós, ao receberem uma distância para um outro nó, apenas actualizam a sua tabela se o valor da distância recebida for menor que a já existente. Caso contrário, não seria possível obter os caminhos mais curtos para todos os nós, como é o objectivo.

Por outro lado, existe um mecanismo que invalida ligações previamente existentes caso não sejam recebidas mensagens sobre estas durante um certo período de tempo. Este intervalo de tempo é normalmente considerado como sendo 6 vezes o período entre a troca de mensagens (6 × 30 segundos por convenção).

Uma vez que este protocolo de encaminhamento é relativamente simples, oferece poucas técnicas de prevenção de ocorrência de *loops* (os quais podem acontecer por várias razões, nomeadamente erros de transmissão).

Um dos efeitos verificados aquando da ocorrência de *loops* é a contagem para infinito, que ocorre quando vários *routers* trocam entre si mensagens com uma distância finita para uma rede inacessível, sem o terem detectado. Como os *routers* não detectaram a falha no acesso à rede, aceitam distâncias desactualizadas dos *routers* vizinhos que foram enviadas por eles próprios, incrementam-nas de 1 e enviam-nas. Os *routers* vizinhos procedem do mesmo modo e a métrica da rede inacessível nas tabelas de encaminhamento aumenta indefinidamente. Para contornar este problema definiu-se que no RIP uma métrica infinita corresponde ao valor 16. Isto limita o número de *hops* numa rede a 15 e como tal, uma rede a correr o RIP tem as suas dimensões limitadas por este valor.

A solução descrita atrás tem o inconveniente de deixar a rede convergir para infinito durante um intervalo de tempo bastante longo (os pacotes de *update* RIP são, por *default*, enviados de 30 em 30 segundos, logo são precisos 16 × 30 = 8 minutos para a rede convergir). Durante este período a rede está instável e ocorrem retransmissões dos pacotes, podendo mesmo ocorrer situações de congestionamento, o que pode levar à perda dos pacotes de *update* do protocolo. Para solucionar este problema foram propostas as seguintes soluções:

- *Triggered updates* – Esta solução consiste no despoletar de uma mensagem de *update* sempre que é detectada uma alteração na tabela de encaminhamento. Isto reduz significativamente o período de convergência da rede que passa a ser quase instantâneo.

- *Split horizon* – Neste caso, um *router* não anuncia numa *interface* os caminhos recebidos pela mesma. Na forma de *split horizon* com *poisonous reverse*, as distâncias que foram recebidas por uma ligação são transmitidas nesta com valor infinito.

As soluções apresentadas reduzem bastante a probabilidade de ocorrência de *loops*, mas são ainda vulneráveis a erros de transmissão e estão longe de serem perfeitas. Estes factos conjugados com outras limitações do RIP, levaram ao desenvolvimento de outros protocolos de encaminhamento.

Protocolo OSPF

O protocolo OSPF (*Open Shortest Path First*) foi o protocolo proposto pelo IETF (*Internet Engineering Task Force*) para suceder ao RIP e é o recomendado actualmente pelo IAB (*Internet Architectural Board*). A sua versatilidade e fiabilidade são bastante superiores e como tal também a sua complexidade.

O OSPF introduz entre outras as seguintes facilidades adicionais: possibilidade de escolha de múltiplos caminhos de igual custo para o mesmo destino; convergência mais rápida e isenta de *loops*; divisão de redes grandes em áreas; representação separada de caminhos externos (para outros sistemas autónomos).

Uma vez que o OSPF é um protocolo do tipo *link state routing*, cada *router* possui uma base de dados com toda a informação sobre a topologia da área da rede onde se encontra inserido. Essa base de dados é igual para todos os *routers* dentro da mesma área. Outro dos conceitos introduzidos pelo OSPF é a noção de *backbone area*, que é a área que faz a ligação entre o seu sistema autónomo e os sistemas autónomos externos. Todas as áreas do sistema autónomo devem estar ligadas à área *backbone*.

Uma das opções do OSPF para diminuir o volume de tráfego de pacotes de encaminhamento na rede é a escolha de um *designated router*. Este *router* é o responsável por anunciar o *Link State Advertisement* da sua rede local (o *LSA* é um pacote com o conteúdo da base de dados sobre a topologia da rede local) aos *routers* adjacentes (*routers* que trocam entre si informação sobre a rede dizem-se adjacentes). Um *designated router* é adjacente de todos os *routers* da sua rede local e é adjacente dos *designated routers* das redes vizinhas. No entanto, um outro *router* só é adjacente do *designated router* da sua rede local. Isto reduz o número de trocas de pacotes de sincronismo das bases de dados, uma vez que dentro de uma rede local só existe troca de informação entre cada router e o *designated router* ou o *backup designated router*.

A base de dados do OSPF é composta por registos (também denominados de *Link State Advertisements*) de cinco tipos distintos:

- *Router Links* - são usados para caracterizar completamente cada *router*. Contêm informação sobre o estado e o custo de todas as interfaces de um *router*.

- *Network Links* - são gerados pelos *designated routers* e contém informação sobre todos os *routers* da rede local incluindo o *designated router*. Não são gerados para redes com apenas um *router* (*stub networks*).

- *Summary Links* (para uma rede IP) - são anunciados pelos *area-border routers* (os *routers* nos limites das áreas OSPF) e contêm informação sobre as redes a que estes *routers* têm acesso noutras áreas.

- *Summary Links* (para um *border router*) - tal como os anteriores só que para *border routers* (*routers* nos limites de sistemas autónomos).

- *External Links* - são anunciados pelos *border routers* e anunciam redes adquiridas por protocolos de encaminhamento externos (como o EGP ou BGP).

Esta representação separada dos diferentes tipos de registos trás vantagens do ponto de vista de organização da topologia da rede.

Por outro lado, o OSPF é constituído por vários sub-protocolos:

- *Hello* – É o procedimento usado para se eleger o *designated router* e o *backup designated* router, aquando da inicialização da rede. Durante esta fase os *routers* difundem para toda a rede local os seus pacotes de *Hello*. Cada um destes pacotes contem a prioridade do *router* que o envia – o que tiver maior prioridade será o *designated router*. Posteriormente são trocados periodicamente pacotes de *Hello* para testar a operacionalidade das ligações.

- *Exchange* - Este protocolo serve para estabelecer comunicação entre dois *routers* adjacentes. Consiste em determinar um *router* como *master* (aquele que inicia a comunicação é o escolhido para ser o *master*) e outro como *slave* (cujo papel é emitir *acknowledges* ao receber as mensagens). A seguir são trocados pacotes de descrição dos registos das bases de dados respectivas.

- *Flooding* – Após a troca das descrições das bases de dados, são trocados os conteúdos dos registos destas através da difusão por todos os routers adjacentes de pacotes de *Link State Update*. Segundo este protocolo, quando um *router* recebe um destes *updates*, envia-o para todos os *routers* adjacentes excepto para o que enviou esse pacote.

Protocolo IGRP

O IGRP (*Inter Gateway Routing Protocol*) é um protocolo de encaminhamento do tipo *distance vector*. Foi desenvolvido pela Cisco Systems para superar algumas das limitações do RIP. Como alterações, incorpora entre outras, suporte a diferentes métricas, *multipath routing* (o *router* guarda os vários caminhos possíveis para um destino), maior protecção face a *loops* e intervalos maiores entre *updates* (90 segundos).

Sendo um melhoramento ao RIP, o IGRP já incorpora alguns mecanismos mais eficazes de prevenção a *loops* e suporte de métricas compostas.

As métricas usadas no IGRP para a selecção dos caminhos são quatro: atraso (*Delay*), largura de banda (*Bandwidth*), fiabilidade (*Reliability*), carga (*Load*). O *Delay* (D) é expresso em unidades de 10 microsegundos e representa a soma dos atrasos de transmissão de todas as ligações entre o *router* e o destino. A *Bandwidth* (B) corresponde à divisão de 10 milhões pela menor largura de banda (em kbps) encontrada em todos os *links* para o destino. A *Reliability* (R) mede a probabilidade de ocorrência de erros no caminho e é expressa como uma grandeza de 8 bits (100% corresponde a 255). A *Load* (L) mede a ocupação da ligação mais "estreita" no caminho e é medida como uma percentagem de ocupação da ligação, do mesmo modo que a fiabilidade (100% de ocupação corresponde a 255). Tanto a fiabilidade como a carga são grandezas dinâmicas (medidas constantemente pelos *routers*) ao passo que a largura de banda e o atraso são estáticos (são parâmetros definidos pelo gestor da rede).

Para calcular os melhores caminhos é utilizada a seguinte fórmula:

$$M = \left(K1 \times B + \frac{K2 \times B}{256 - L} + K3 \times D \right) \frac{K5}{R + K4}$$

Uma das vantagens do IGRP é a capacidade de guardar na memória uma lista de todos os caminhos possíveis para um destino. Isto permite, caso haja uma falha numa ligação, haver uma transição quase imediata para o seguinte melhor *link*, permitindo assim um melhor funcionamento da rede.

Por outro lado, se existirem caminhos com métricas iguais, é possível ao *router* utilizar simultaneamente mais do que um para o mesmo destino, permitindo assim uma distribuição da carga pelas ligações. É possível também modificar o parâmetro variância (V) que por *default* é igual a 1, ou seja, apenas são seleccionados caminhos com métricas iguais às do melhor caminho. Modificando este coeficiente, são utilizados também os caminhos cuja métrica não seja maior que V vezes a métrica do melhor caminho.

No que se refere à protecção contra *loops*, tal como o RIP, o IGRP incorpora *triggered updates* e *split horizon*, mas em vez de *poisonous reverse*, utiliza técnicas como *path holddown* ou *route poisoning* (que substitui *path holddown* nas últimas versões) para prevenir a ocorrência de *loops*.

Path holddown consiste em manter um *link* em "quarentena" quando é detectada uma falha neste. Na prática, a ligação fica *on hold*, o que significa que durante este período (180 segundos), não são aceites quaisquer *updates*. Este período serve para garantir que a informação sobre a falha da ligação já atingiu toda a rede. Quando acaba o período de quarentena, a selecção de caminhos volta a processar-se como anteriormente.

Route poisoning consiste em não usar caminhos cujo *hop count* (número de saltos percorridos para o destino) aumenta - o que pode acontecer devido à formação de um *loop*. Caso isto se verifique, o caminho deixa de ser usado até que um *update* posterior confirme o novo *hop count*.

Protocolo EIGRP

O EIGRP (*Enhanced Inter Gateway Routing Protocol*) é uma extensão do protocolo IGRP também desenvolvido pela Cisco Systems para superar algumas das limitações do IGRP, nomeadamente o grande intervalo de tempo de inoperacionalidade parcial da rede imposto pelas técnicas de prevenção de *loops*.

Para cálculo da métrica, o EIGRP usa um processo semelhante ao IGRP e utiliza as mesmas constantes e parâmetros (*Delay*, *Bandwidth*, *Reliability* e *Load*).

A maior diferença do EIGRP para o IGRP consiste na técnica utilizada para a prevenção de ocorrência de *loops*.

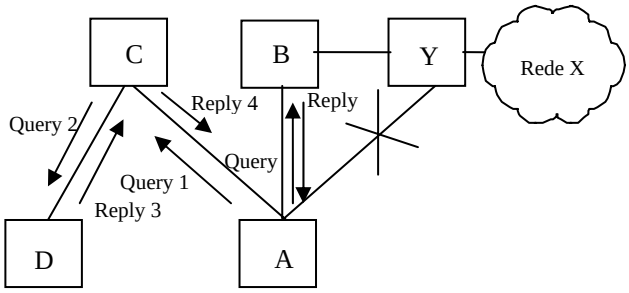

Fig. 1 - Exemplo algoritmo DUAL

A técnica usada pelo EIGRP para melhorar a tecnologia dos *distance vector protocols* no que respeita à prevenção de *loops* é conhecida por *Diffusing Update Algorithm* (DUAL). No exemplo da Fig.1 o *router* A para calcular o custo para um destino X, ao receber de outro *router* Y o seu custo para X, soma a este valor o custo da ligação entre si e Y. Este processo é repetido para todos os routers Y. O caminho escolhido para X é pelo *router* Y que minimiza o custo. Um *router* com uma versão estável da sua tabela de encaminhamento diz-se um *router* passivo.

Quando um *router* recebe um *update* com um custo inferior ao da sua tabela, ele aceita a nova ligação e anuncia o *update* para os seus vizinhos. Caso o *update* provenha do *router* Y (que tinha enviado o custo antigo) com um custo maior ao da sua tabela, então o *router* A procura de novo um *router* vizinho Y com o custo inferior, por onde atingir o destino X. Se houver um vizinho aceite, o *router* A anuncia o *update* aos seus vizinhos. Caso contrário, o *router* inicia o processo de *diffusion* (DUAL).

Enquanto o processo de *diffusion* não se completar, o *router* A irá congelar a sua entrada da tabela com o destino X. Durante este processo, o *router* A envia uma mensagem de *Query* a todos os *routers* excepto para aquele de onde recebeu o *update* (Y). Este *Query* possui a distância entre o *router* A e X e a nova distância entre A e X.

O *router* que recebe esta mensagem, trata-a como um *update* normal. Se este *router* verificar que (i) não utilizava o *router* que difundiu o *Query* para atingir X ou (ii) encontrou um caminho alternativo para X, então este *router* deve ficar no estado passivo e responder com a nova versão da sua tabela de encaminhamento - *Reply*. Caso contrário, deve passar ao estado activo e propagar *Queries* a todos os seus vizinhos, que por sua vez devem proceder do mesmo modo. Quando um *router* recebeu *replies* de todos os seus *routers* vizinhos, passa do estado activo para o estado passivo e deve responder ao *Query* que lhe foi dirigido no início. Eventualmente o *router* A que enviou o primeiro *Query* receberá *Replies* de todos os seus vizinhos e regressará ao estado passivo - a convergência será atingida.

Na figura anterior o router A tinha um caminho para a rede X por Y e ao perder esse caminho emitiu *Queries* para todos os seus vizinhos (excepto Y). O router B já conhecia um caminho para X por Y, logo responde com um *Reply* (com o seu caminho para X). O router C como utilizava A para atingir X, emite *Queries* para todos os seus vizinhos (D). D como não tem nenhum vizinho, emite um *Reply* (3) para C com distância infinita para X. C ao receber este *Reply*, responde a A (*Reply* 4) e A escolhe o melhor caminho (por B – único caminho). A seguir passa ao estado passivo e difunde a sua nova tabela.

Protocolo PIM

O interesse demonstrado nos últimos anos em tecnologias emergentes de vídeo a pedido, videoconferência, etc., têm em grande parte contribuído para o desenvolvimento de técnicas de *multicasting*. Isto provém nomeadamente da diminuição da carga na rede resultante da aplicação destas técnicas. Enquanto em transmissões ponto-a-ponto quase toda a rede precisa de ser inundada com informação (as mesmas mensagens são consecutivamente transmitidas para cada destino), em *multicasting* a informação é transmitida apenas uma vez (cada elemento recebe a informação se pertencer ao grupo de *multicast*).

Uma das soluções propostas pelo IDMR (*Inter-Domain Multicast Routing working group*) é o chamado PIM (*Protocol Independent Multicast*) que é dirigido à área do *multicast routing* na Internet. Como o nome indica, o PIM pode ser implementado em cima de qualquer protocolo de encaminhamento (que tem obrigatoriamente de existir).

Esta solução engloba dois modos distintos: *dense* e *sparse*. O modo *dense*, tal como o nome indica, é apropriado para redes com grande incidência de membros do grupo de *multicast*, ao passo que o modo *sparse* é indicado quando o número de membros de um grupo é pequeno relativamente ao número de membros das redes onde estão inseridos.

O modo *dense* caracteriza-se da seguinte maneira. Quando um *router* recebe um pacote *multicast* (endereço do tipo 224.x.x.x) de uma fonte X para um grupo destino Y, ele verifica na tabela de encaminhamento (do protocolo usado: RIP, OSPF, etc.) se a interface por onde este pacote foi recebido é usada para enviar pacotes para a fonte X. Se não for este o caso, o *router* ignora o pacote e envia uma mensagem de *prune*(X,Y) pela mesma interface. Esta mensagem indica ao *router* que enviou o pacote que não deve enviar mais pacotes *multicast* para a interface do *router* que enviou a mensagem de *prune*. Caso contrário, o *router* envia a mensagem por todas as interfaces que não receberam uma mensagem de *prune*(X,Y). Se não houver interfaces nestas condições, é enviada uma mensagem de *prune*(X,Y) pela interface onde recebeu o pacote. Este procedimento é seguido por todos os *routers*.

Como se pode verificar, no início toda a rede é inundada com as mensagens *multicast*, e à medida que são recebidas mensagens de *prune*, vai-se formando a "árvore" de encaminhamento. Quando um utilizador se pretende juntar ao grupo de *multicast*, o router envia uma mensagem *multicast* de *join* a todos os elementos do grupo respectivo.

No caso do modo *sparse*, para toda a rede não ser inundada de cada vez que há uma transmissão *multicast*, os membros que querem pertencer ao grupo *multicast* devem inicialmente avisar um *router* (o *rendezvous point RP*) que se pretendem juntar ao respectivo grupo através do envio de uma mensagem de *join*. O *rendezvous point* tem de ser previamente conhecido pela fonte *multicast*. A cada grupo podem estar associados um ou mais *rendezvous points*.

Os *routers* intermédios devem guardar as interfaces por onde recebem as mensagens de *join* para o *rendezvous point* como pertencentes ao grupo. Assim é possível formar uma "árvore" de encaminhamento inicial à volta do *rendezvous point*. Seguindo um processo idêntico ao modo *dense*, o percurso seguido pelos pacotes *multicast* vai sendo optimizado com o envio de mensagens de *prune*.

Protocolo EGP

O EGP (*Exterior Gateway Protocol*) é um protocolo de encaminhamento externo concebido para a troca de informação entre sistemas autónomos. Apesar de ainda se encontrar em uso, está hoje em dia a ser substituído pelo BGP devido às suas limitações.

Alguns dos defeitos do EGP devem-se a ser orientado para uma topologia à volta de um núcleo central (era assim a estrutura da Internet nos primeiros anos). Como tal, supõe uma topologia hierarquizada e não é adequado à complexidade que tem vindo a manifestar-se no desenvolvimento da Internet actual.

Outros problemas do EGP são a falta de segurança das mensagens (é bastante vulnerável a erros induzidos por *routers* ou outras causas externas), o não permitir *policy routing* e o envio de toda a informação sobre as redes num único pacote. O *Policy routing* provém de diferentes organizações terem diferentes preferências relativamente aos caminhos por onde passa a sua informação - diferentes redes têm diferentes graus de segurança e fiabilidade e são adequadas a políticas de encaminhamento distintas. O problema de enviar toda a informação sobre as redes num único pacote advém da grande variedade de redes existente hoje em dia em alguns sistemas autónomos. Isto provoca uma necessidade de fragmentação do pacote pois algumas redes impõem um comprimento máximo para estes. Como tal, basta que um fragmento do pacote se perca, para que se perca toda a mensagem.

Quanto ao modo de funcionamento, o EGP troca informação entre sistemas autónomos através de três procedimentos:

- *Neighbor acquisition* - É o processo segundo o qual dois *routers* concordam em tornar-se vizinhos. Este processo implica uma mensagem de confirmação por parte do *router* que recebe a mensagem.

- *Neighbor reachability* - É usado para testar a operacionalidade das ligações. É repetido em intervalos periódicos definidos no processo anterior (tipicamente 30 segundos).

- *Network reachability* - O seu propósito é trocar informação sobre as redes acessíveis por cada *router*. Para tal, cada *router* envia periodicamente aos seus vizinhos uma mensagem de *polling* (normalmente de 3 em 3 minutos), ao que estes devem responder com uma mensagem de *update* (com a informação sobre todas as redes do seu sistema autónomo).

Protocolo BGP

O BGP (*Border Gateway Protocol*) é um protocolo de encaminhamento externo concebido pelo IETF para substituição do EGP uma vez que este impunha algumas limitações ao desenvolvimento da Internet. Foram desenvolvidas várias versões deste e neste momento já existem as versões BGP-1, BGP-2, BGP-3 e BGP-4, sendo esta última a que está a ser empregue hoje em dia.

Este protocolo tem como particularidade correr em cima do TCP, o que liberta o BGP de algumas das complexas funcionalidades de recuperação de erros e de definição do tamanho das mensagens IP.

O BGP utiliza um conceito semelhante a *distance vectors* para prevenir a ocorrência de *loops*, conceito esse que se denomina *path vectors*. Um *path vector*, tal como o nome indica, descreve todo o caminho percorrido pelas mensagens de *update* e não apenas a distância como no caso *distance vector*.

Neste caso, o caminho (*path vector*) corresponde à lista de todos os sistemas autónomos atravessados pela mensagem de *update*. Esta informação é usada pelo BGP para evitar a ocorrência de *loops*. Como tal, se um sistema autónomo detectar que um caminho já passou por ele,

então esse caminho é recusado, o que elimina a formação de qualquer *loop*.

Cada *path* é descrito por um conjunto de atributos que o caracterizam e que podem ser usados pelos *routers* para a escolha entre os vários caminhos possíveis.

Outra das novidades introduzidas pelo BGP é a noção de vizinhos internos, ou seja, cada *router* externo deve, além de trocar informação com os outros sistemas autónomos, trocar informação com os outros *routers* externos dentro do mesmo sistema autónomo. Este procedimento ocorre independentemente do IGP, e permite que os *routers* cheguem a acordo entre si sobre quais os melhores caminhos para outros sistemas autónomos.

IV. EXPERIÊNCIAS COM OS PROTOCOLOS

Apesar se um grande número de experiências terem sido efectuadas, e que estão descritas no anexo A. Neste artigo apenas estão descritas aquelas que na nossa opinião melhor descrevem o espirito que esteve inerente na concepção de todas elas.

Aspectos de configuração das experiências

Nas experiências foram usados quatro *routers* 2514 com o Cisco IOS 11.2(5) como sistema operativo. Foram ainda usados vários PC's onde foi instalado o analisador de protocolos por *software* NetXRay 3.0.3 da *Network General.*

O encaminhamento de pacotes IP, é feito através de *routers* os quais possuem uma tabela de encaminhamento. Na tabela de encaminhamento está descrito o modo como o *router* deve proceder quando recebe um pacote IP num dos seus interfaces. Um exemplo de uma tabela está representada a seguir.

```
R   2.0.0.0/8 [120/1] via 4.1.1.3, 00:00:26, Ethernet1
                  [120/1] via 3.1.1.1, 00:00:02, Ethernet0
C   3.0.0.0/8 is directly connected, Ethernet0
C   4.0.0.0/8 is directly connected, Ethernet1
R   5.0.0.0/8 [120/1] via 4.1.1.4, 00:00:23, Ethernet1
```

A letra no início de cada linha representa o modo como essa entrada da tabela foi obtida, C para redes directamente ligadas ao *router*, R para entradas obtidas com o protocolo RIP, O para entradas obtidas com o protocolo OSPF, I para IGRP, etc.

De seguida vem o endereço da rede destino, acrescido da informação sobre a máscara associada ao endereço IP dessa rede. A máscara do endereço permite a divisão de uma rede em sub-redes.

Entre parêntesis rectos estão as métricas associadas a esse caminho, primeiro a distância associada ao respectivo protocolo, no exemplo 120 para o protocolo RIP, este valor apenas serve para estabelecer uma hierarquia de protocolos sendo sempre escolhidos os caminhos obtidos pelo protocolo de menos distância. O segundo campo é custo desse caminho, e o *router* escolhe o caminho que menor custo tiver.

A próxima informação da tabela é o endereço no nó para onde o *router* deve enviar o pacote. Depois vem a idade da entrada da tabela seguida da indicação do interface por onde o *router* vai enviar o pacote IP.

Reacção do protocolo RIP a uma simples alteração na rede

A rede utilizada na experiência foi a da Fig.2. A rede consistia em quatro redes interligadas por quatro *routers* a correr o protocolo RIP. Esta experiência tinha como objectivo observar a reacção do processo de

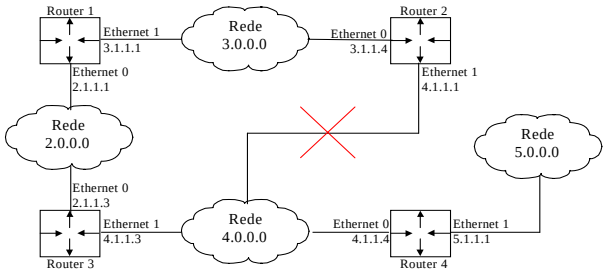

Fig.2 - Rede 1

encaminhamento a uma simples alteração na estrutura da rede. Para tal foi desligado o interface Eth.1(4.1.1.1) do *router* 2. Antes de desligar o interface a tabela de encaminhamento do *router* 2 era a seguinte:

```
R   2.0.0.0/8 [120/1] via 4.1.1.3, 00:00:26, Ethernet1
                  [120/1] via 3.1.1.1, 00:00:02, Ethernet0
C   3.0.0.0/8 is directly connected, Ethernet0
C   4.0.0.0/8 is directly connected, Ethernet1
R   5.0.0.0/8 [120/1] via 4.1.1.4, 00:00:23, Ethernet1
```

Após desligar o interface a tabela reconfigurou-se do seguinte modo:

```
R   2.0.0.0/8 [120/1] via 3.1.1.1, 00:00:02, Ethernet0
C   3.0.0.0/8 is directly connected, Ethernet0
R   4.0.0.0/8 [120/2] via 3.1.1.1, 00:00:08, Ethernet0
R   5.0.0.0/8 [120/3] via 3.1.1.1, 00:00:23, Ethernet0
```

Verifica-se que os caminhos que passavam pelo interface Eth.1(4.1.1.1) do *router* 2 desapareceram da tabela. Por exemplo o caminho do *router* 2 para a rede 5.0.0.0 que antes de desligar o interface tinha apenas um custo de 1, pois passava directamente do *router* 2 para o 4, passou a ter um custo de 3 pois agora o encaminhamento é feito através dos *routers* 1,3 e 4.

Análise da troca de pacotes durante a inicialização do protocolo RIP usando o analisador de protocolos

A rede utilizada na experiência foi a da Fig.2. A rede consistia em quatro redes interligadas por quatro *routers* a correr o protocolo RIP. Com o objectivo de observar a troca de mensagens de inicialização do protocolo RIP. Montou-se a rede da Fig. 3. No analisador de protocolos escolheu-se a opção de filtrar pacotes RIP. A seguir, antes de ligar os *routers*, activou-se a captura de pacotes. Após isto, ligou-se o *router* 2 e dois minutos depois o *router* 1. Passados mais alguns minutos observaram-se os pacotes filtrados.

Na rede 200.10.10.0 os 12 primeiros pacotes capturados foram os seguintes:

Frame	Source Address	Dest. Address	Rel. Time	Summary
1	200.10.10.2	Broadcast	000:00:37.075	(Request)
2	200.10.10.2	Broadcast	000:00:47.087	(Response)
3	200.10.10.2	Broadcast	000:00:47.119	(Response)
4	200.10.10.2	Broadcast	000:00:47.147	(Response)
5	200.10.10.2	Broadcast	000:01:15.831	(Response)
6	200.10.10.2	Broadcast	000:01:43.321	(Response)
7	200.10.10.1	Broadcast	000:02:07.356	(Request)
8	200.10.10.1	Broadcast	000:02:13.041	(Response)
9	200.10.10.2	Broadcast	000:02:13.109	(Response)
10	200.10.10.1	Broadcast	000:02:17.376	(Response)
11	200.10.10.2	Broadcast	000:02:40.571	(Response)
12	200.10.10.1	Broadcast	000:02:45.559	(Response)

Observando o pacote n.º 9 mais em pormenor, este continha a seguinte informação relativa ao protocolo RIP:

Packet 9
IP Routing Information Protocol
Command: 2 (Response)
Version: 1
Address Family ID: 2
IP Address: 200.10.10.0
Metric: 1
Address Family ID: 2
IP Address: 200.10.11.0
Metric: 1
Address Family ID: 2
IP Address: 200.10.12.0
Metric: 2
Address Family ID: 2
IP Address: 200.10.13.0
Metric: 2

Os pacotes provenientes do *router* 1 são aqueles cujo *Source Address* é 200.10.10.1 e os provenientes do *router 2* são aqueles cujo *Source Address* é 200.10.10.2.

Dos pacotes capturados pode-se verificar que o primeiro pacote que um *router* envia é um pacote de *request*, o qual serve para notificar os restantes *routers* da rede da sua activação e pedir que lhe façam a difusão das suas tabelas. Toda a informação relativa à rede é passada em pacotes do tipo *response*. Depois da estabilização das tabelas de encaminhamento, cada *router* apenas envia um pacote de *response* de 30 em 30 segundos. Como se pode ver no caso do pacote n.º 9 (*response*), este transporta os endereços das redes conhecidas e o custo do caminho entre cada destino e o *router* que envia o pacote de *res*ponse.

Reflexo da alteração do custo dos interfaces nas tabelas de encaminhamento do OSPF

A rede utilizada na experiência foi a da Fig.3. A rede consistia em quatro redes interligadas por quatro *routers* a correr o protocolo OSPF.

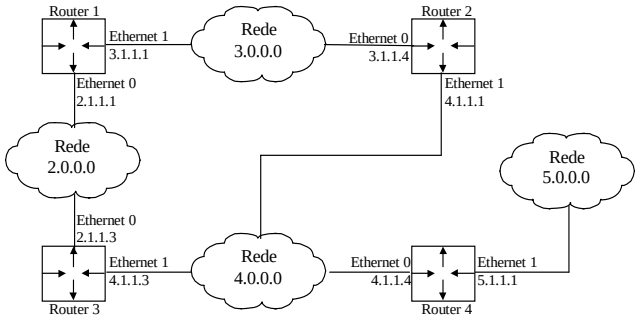

Fig. 3 - Rede 2

O custo dos interfaces, ou seja o valor que é acrescido à métrica de um determinado caminho que passe por esse interface, era inicialmente de 10 em todos eles. A tabela de encaminhamento do *router* 1era então a seguinte:

C 2.0.0.0/8 is directly connected, Ethernet0
C 3.0.0.0/8 is directly connected, Ethernet1
O 4.0.0.0/8 [110/20] via 3.1.1.4, 00:01:13, Ethernet1
[110/20] via 2.1.1.3, 00:01:31, Ethernet0
O 5.0.0.0/8 [110/30] via 3.1.1.4, 00:01:10, Ethernet1
[110/30] via 2.1.1.3, 00:01:10, Ethernet0

Sendo o objectivo verificar como o processo de encaminhamento reage a alterações do custo dos interfaces, foi mudando custo associado ao interface Eth.1(4.1.1.1) do *router* 2 para 5. A tabela de encaminhamento do *router* 1 reconfigurou-se do seguinte modo:

C 2.0.0.0/8 is directly connected, Ethernet0
C 3.0.0.0/8 is directly connected, Ethernet1
O 4.0.0.0/8 [110/15] via 3.1.1.4, 00:00:13, Ethernet1
O 5.0.0.0/8 [110/25] via 3.1.1.4, 00:00:13, Ethernet1

De seguida foi alterado o custo associado ao interface Eth.1(4.1.1.1) do *router* 3 de 10 para 2. A tabela do *router* 1 ficou assim:

C 2.0.0.0/8 is directly connected, Ethernet0
C 3.0.0.0/8 is directly connected, Ethernet1

O 4.0.0.0/8 [110/12] via 2.1.1.3, 00:01:17, Ethernet0
O 5.0.0.0/8 [110/22] via 2.1.1.3, 00:01:17, Ethernet0

Depois da primeira alteração de custos, o encaminhamento passou a fazer-se pelo *router* 2 devido à diminuição do custo do interface deste. Quando se coloca o custo do interface do *router* 3 ainda mais baixo do que o do *router 2* então o encaminhamento passa a ser feito pelo *router 3.*

Reflexo da alteração da carga da rede nas tabelas de encaminhamento do IGRP

A rede utilizada na experiência foi a da Fig.4. A rede consistia em quatro redes interligadas por quatro *routers* a correr o protocolo IGRP.

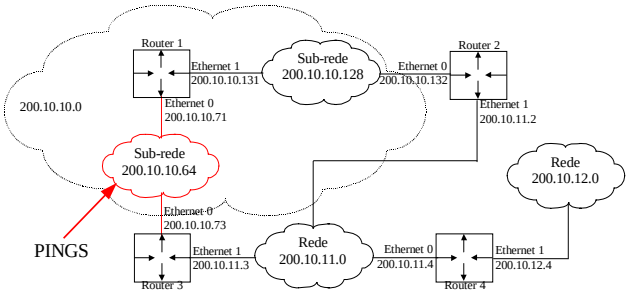

Fig.4 - Rede 3

O objectivo desta experiência foi verificar a alteração do encaminhamento quando há variação da carga de uma rede.

Um dos parâmetros que o *software* do *router* mede é a carga da rede dando a esta um valor de 1 a 255, 1 para a rede quase sem tráfego e 255 para uma rede completamente congestionada. Visto a métrica do protocolo IGRP ser composta, onde a carga da rede é uma das suas componentes, é possível que o encaminhamento seja alterado apenas devido à sobrecarga de uma rede com tráfego.

Com tráfego mínimo em todas as redes a tabela de encaminhamento do *router* 1 era a seguinte:

200.10.10.0/26 is subnetted, 2 subnets
C 200.10.10.128 is directly connected, Ethernet1
C 200.10.10.64 is directly connected, Ethernet0
I 200.10.11.0/24 [100/39215] via 200.10.10.73, 00:00:30, Ethernet0
[100/39215] via 200.10.10.132, 00:00:51, Ethernet1
I 200.10.12.0/24 [100/39215] via 200.10.10.73, 00:00:30, Ethernet0
[100/39215] via 200.10.10.132, 00:00:13, Ethernet1

De modo a sobrecarregar a sub-rede 200.10.10.64 com tráfego, injectaram-se pacotes PING na rede. Com o *router* 1 a considerar a sub-rede 200.10.10.64 com uma carga de 255, a tabela de encaminhamento era a seguinte:

200.10.10.0/26 is subnetted, 2 subnets
C 200.10.10.128 is directly connected, Ethernet1
C 200.10.10.64 is directly connected, Ethernet0
I 200.10.11.0/24 [100/39215] via 200.10.10.132, 00:00:51, Ethernet1
I 200.10.12.0/24 [100/39215] via 200.10.10.132, 00:00:13, Ethernet1

Verifica-se que os caminhos que passam pela sub-rede 200.10.10.64 foram retirados da tabela, sendo o encaminhamento feito pela rede não sobrecarregada.

Reflexo da alteração da fiabilidade da rede nas tabelas de encaminhamento do IGRP

A rede utilizada na experiência foi a da Fig.5. A rede consistia em quatro redes interligadas por quatro *routers* a correr o protocolo IGRP.

Outro dos parâmetros que o *software* do *router* mede é a fiabilidade da rede, dando a esta um valor de 1 a 255, 255 para a rede sem falhas e 1 para uma rede com um muito elevado número de falhas.

Com todas as redes com fiabilidade máxima a tabela de encaminhamento do *router* 1 era a seguinte:

200.10.10.0/26 is subnetted, 2 subnets
C 200.10.10.128 is directly connected, Ethernet1
C 200.10.10.64 is directly connected, Ethernet0
I 200.10.11.0/24 [100/3] via 200.10.10.73, 00:00:30, Ethernet0
[100/3] via 200.10.10.132, 00:00:51, Ethernet1
I 200.10.12.0/24 [100/3] via 200.10.10.73, 00:00:30, Ethernet0
[100/3] via 200.10.10.132, 00:00:13, Ethernet1

De modo a diminuir a fiabilidade desligou-se e ligou-se algumas vezes as sub-redes directamente ligadas ao *router* 1, o valor da fiabilidade da rede do interface Eth0 passou a ter um valor de 240 e a tabela do *router* 1 ficou a seguinte:

200.10.10.0/26 is subnetted, 2 subnets
C 200.10.10.128 is directly connected, Ethernet1
C 200.10.10.64 is directly connected, Ethernet0
I 200.10.11.0/24 [100/3] via 200.10.10.132, 00:00:51, Ethernet1
I 200.10.12.0/24 [100/3] via 200.10.10.132, 00:00:13, Ethernet1

Após a diminuição da fiabilidade da rede 200.10.10.64, o encaminhamento passou a fazer-se pelo *router* 2.

AGRADECIMENTOS

À CONVEX e à CISCO Portugal, pelas facilidades concedidas na aquisição dos routers 2514.

V. BIBLIOGRAFIA

[1]- C. Borges, P. Ferreira,., “Encaminhamento e qualidade de serviço na Internet”, Relatório do projecto de fim-de-curso da Licenciatura em Eng. Electrónica e Telecomunicações, DET-UA, 1998
[2]- Huitema, Christian, “Routing in the Internet”, Prentice Hall. 1995
[3]- Tanenbaum, Andrew S., “Computer Networks”, Prentice Hall, 1996

[4]- RFC 1058 Routing Information Protocol. C.L. Hedrick. Jun-01-1988.
[5]- RFC 1388 RIP Version 2 Carrying Additional Information. G. Malkin. January 1993.
[6]- Halabi, Sam, Cisco Systems OSPF Design Guide, Apr. 1996
[7]- RFC 2328 OSPF Version 2. J. Moy. April 1998.
[8]- RFC 0904 Exterior Gateway Protocol formal specification. D.L. Mills. Apr-01-1984.
[9]- RFC 1267 Border Gateway Protocol 3 (BGP-3). K. Lougheed, Y. Rekhter. Oct-01-1991
[10]- RFC 1654 A Border Gateway Protocol 4 (BGP-4). Y. Rekhter & T. Li, Editors. July 1994.
[11]- Documentação do Cisco IOS 11.2(5).
[12]- Página da Cisco na Internet: http://www.cisco.com/ .

ANEXO A - LISTA DE EXPERIÊNCIAS

Protocolo RIP

- Reacção do processo de encaminhamento a uma simples alteração na rede
- Verificação da existência de *triggered updates*
- Análise do processo de escolha entre caminhos com métricas idênticas
- Estudo da difusão das máscaras das subredes
- Tentativa de reproduzir uma contagem para infinito
- Experiências com o analisador de protocolos
- Inicialização
  - *Triggered updates*
  - Sem *split horizon*
  - Uso do *debugger* do IOS
- Uso do *debugger* do IOS

Protocolo OSPF

- Reacção do processo de encaminhamento a uma simples alteração na rede
- Reacção do processo de encaminhamento a alterações dos custos dos interfaces
- Verificação da rapidez de convergência das tabelas de encaminhamento face ao RIP
- Estudo da difusão das máscaras das subredes
- Implementação de diferentes áreas numa rede
- Estudo do processo de escolha do *designated router* e *backup designated router*
- Experiências com o analisador de protocolos
  - Inicialização
  - Reacção a alterações na rede
- Experiências com RIP e OSPF simultaneamente

Protocolo IGRP

- Reacção do processo de encaminhamento a uma simples alteração na rede
- Estudo da difusão das máscaras das subredes
- Alteração da constante da métrica associada ao atraso por interface
- Reflexo da alteração da largura de banda de um interface nas tabelas de encaminhamento
- Reflexo do atraso de um interface nas tabelas de encaminhamento
- Reflexo da alteração da carga da rede nas tabelas de encaminhamento
- Reflexo da alteração da fiabilidade da rede nas tabelas de encaminhamento
- Experiências com o analisador de protocolos
  - Inicialização
  - Reacção a alterações na rede

Protocolo EIGRP

- Reacção do processo de encaminhamento a uma simples alteração na rede
- Estudo da difusão das máscaras das subredes
- Alteração da constante da métrica associada ao atraso por interface
- Reflexo da alteração da largura de banda de um interface nas tabelas de encaminhamento
- Reflexo do atraso de um interface nas tabelas de encaminhamento
- Reflexo da alteração da carga da rede nas tabelas de encaminhamento
- Reflexo da alteração da fiabilidade da rede nas tabelas de encaminhamento

PIM

- *Dense mode*
  - Experiências com diferentes fontes para vários grupos
- *Sparse mode*
  - Experiências com diferentes fontes para vários grupos

Protocolo EGP

- Configuração do EGP com apenas um *border router*
- Configuração do EGP com mais de um *border router*
- Experiências com o analisador de protocolos

Protocolo BGP

- Configuração do BGP com apenas um *border router*
- Configuração do BGP com mais de um *border router*
- Reacção das tabelas à alteração dos pesos associados aos *routers* vizinhos
- Experiências com o analisador de protocolos

Configuração de um PC como router

- Protocolo RIP
- Protocolo OSPF