INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIAS E TECNOLOGIA DE RONDÔNIA

CAMPUS VILHENA

ALLISON SOUSA BAHLS

DEYVID DALBEM DE SOUZA

**CAMADA DE REDES**

Vilhena

2019

ALLISON SOUSA BAHLS

DEYVID DALBEM DE SOUZA

**CAMADA DE REDES**

Trabalho de conclusão de curso apresentada como requisito de avaliação do curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação Ciências e Tecnologia de Rondônia Campus Vilhena para obtenção do título de tecnólogo em Analise e Desenvolvimento de Sistemas.

Vilhena

2019

**SUMÁRIO**

[4.1 Introdução](#_tyjcwt) **4**

[**4.2 Redes de circuitos virtuais e de datagramas**](#_yd3bs4b3ojc5) **6**

[**4.3 O que há dentro de um roteador?**](#_phx3l5u3cqd) **9**

[**4.4 O Protocolo da Internet (IP): repasse e endereçamento na Internet**](#_ffxkfxkapynk) **10**

[**4.4.4 IPv6**](#_q0edlj2ltd0x) **19**

[**4.5 Algoritmos de roteamento**](#_jlkkf0d7tnmf) **20**

[**4.6 Roteamento na Internet**](#_1sug5dvlt6jr) **20**

[**4.7 Roteamento por difusão e para um grupo**](#_pzhd7e98o06e) **23**

1 INTRODUÇÃO

Neste Trabalho estudaremos exatamente como a camada de rede executa o serviço de comunicação hospedeiro a hospedeiro. Veremos que há um pedaço da camada de rede em cada hospedeiro e roteador na rede, o que não acontece com as camadas de transporte e de aplicação. Por causa disso, os protocolos de camada de rede estão entre os mais desafiadores (e, portanto, os mais interessantes!) da pilha de protocolos.

A camada de rede é, também, uma das mais complexas da pilha de protocolos e, assim, temos um longo caminho a percorrer. Iniciaremos nosso estudo com uma visão geral da camada de rede e dos serviços que ela pode prover. Em seguida, examinaremos novamente as duas abordagens da estruturação da entrega de pacotes de camada de rede — o modelo de datagramas e o modelo de circuitos virtuais que vimos pela primeira vez.

Veremos o papel fundamental que o endereçamento desempenha na entrega de um pacote a seu hospedeiro de destino. Faremos, uma distinção importante entre as funções de repasse e roteamento da camada de rede. Repasse envolve a transferência de um pacote de um enlace de entrada para um enlace de saída dentro de um único roteador. Roteamento envolve todos os roteadores de uma rede, cujas interações coletivas por meio do protocolo de roteamento determinam os caminhos que os pacotes percorrem em suas viagens do nó de origem ao de destino.

Para aprofundar nosso conhecimento do repasse de pacotes, examinaremos o “interior” de um roteador a organização e a arquitetura de seu hardware. Então, estudaremos o repasse de pacotes na Internet, junto com o famoso Protocolo da Internet (IP). Investigaremos o endereçamento na camada de rede e o formato de datagrama IPv4. Em seguida, estudaremos a tradução de endereço de rede (Network Address Translation, NAT), a fragmentação de datagrama, o Protocolo de Mensagem de Controle da Internet (Internet Control Message Protocol — ICMP) e IPv6.

* 1. **REPASSE E ROTEAMENTO**

Assim, o papel da camada de rede é aparentemente simples — transportar pacotes de um hospedeiro remetente a um hospedeiro destinatário. Para fazê-lo, duas importantes funções da camada de rede podem ser identificadas:

• **Repasse.** Quando um pacote chega ao enlace de entrada de um roteador, este deve conduzi-lo até o enlace de saída apropriado.

**• Roteamento.** A camada de rede deve determinar a rota ou o caminho tomado pelos pacotes ao fluírem de um remetente a um destinatário. Os algoritmos que calculam esses caminhos são denominados algoritmos de roteamento.

Os termos repasse e roteamento são usados indistintamente por autores que estudam a ­camada de rede. Neste Trabalho, usaremos tais termos com maior exatidão. Repasse refere-se à ação local realizada por um roteador para transferir um pacote da interface de um enlace de ­entrada para a interface de enlace de saída apropriada. Roteamento refere-se ao processo de âmbito ­geral da rede que determina os caminhos fim a fim que os pacotes percorrem desde a origem até o destino.

Enquanto estamos no tópico da terminologia, é interessante mencionar dois outros termos que também são utilizados indistintamente, mas que usaremos com maior cuidado. Reservaremos o termo comutador de pacotes para designar um dispositivo geral de comutação de pacotes que transfere um pacote de interface de enlace de entrada para interface de enlace de saída conforme o valor que está em um campo no cabeçalho do pacote. Alguns comutadores de pacotes, denominados comutadores de camada de enlace baseiam a decisão de repasse no valor que está no campo da camada de enlace. Outros denominados roteadores, baseiam sua decisão de repasse no valor que está no campo de camada de rede. Os roteadores, portanto, são dispositivos da camada de rede (camada 3), mas também devem utilizar protocolos da camada 2, pois os dispositivos da camada 3 exigem os serviços da camada 2 para implementar sua funcionalidade (camada 3).

**ESTABELECIMENTO DE CONEXÃO**

Acabamos de dizer que a camada de rede tem duas funções importantes, repasse e roteamento. Mas logo veremos que em algumas redes de computadores há uma terceira função importante, a saber, o estabelecimento de conexão. Lembre-se de que, quando estudamos o TCP, verificamos que é necessária uma apresentação de três vias antes de os dados realmente poderem fluir do remetente ao destinatário. Isso permite que o remetente e o destinatário estabeleçam a informação de estado necessária (por exemplo, número de sequência e tamanho inicial da janela de controle de fluxo). De modo semelhante, algumas arquiteturas de camada de rede — por exemplo, ATM, frame-relay e MPLS exigem que roteadores ao longo do caminho escolhido desde a origem até o destino troquem mensagens entre si com a finalidade de estabelecer estado antes que pacotes de dados de camada de rede dentro de uma dada conexão origem-destino possam começar a fluir. Na camada de rede, esse processo é denominado estabelecimento de conexão.

# 2. APLICAÇÕES DE SERVIÇOS DA CAMADA DE REDE

A camada de rede pode oferecer aplicações de serviços não orientado à conexão ou orientado a conexão, da mesma forma que a camada de transporte que oferece o TCP um serviço orientado a conexão e o UDP não orientado à conexão. Ambas as aplicações das duas camadas tem semelhanças, como o fato de que a o serviço orientado executa uma apresentação antes da troca de informações e o não orientado a conexão não realiza nenhuma apresentação. Mesmo dadas as semelhança eles ainda possuem enormes diferenças em outros aspectos.

Os serviços da camada de rede são de hospedeiro a hospedeiro providos pela própria camada de rede à camada de transporte, enquanto a camada de transporte são serviços de processo a processo fornecidos pela camada de transporte à camada de aplicação.

Enquanto na camada de transporte o serviço orientado a conexão é o TCP e o não orientado a conexão, a camada de rede oferece serviços entre hospedeiros orientado ou não a conexão. Os serviços orientado a conexão entre hospedeiros são denominados de circuitos virtuais (CV) enquanto redes de computadores que oferecem apenas um serviço não orientado para conexão na camada de rede são denominadas redes de Datagramas.

Diferente dos serviços orientados a conexão que eram executados nas bordas da rede nos sistemas finais, esses serviços na camada de rede é executada diretamente no núcleo do roteador.

**2.1. REDES DE CIRCUITOS VIRTUAIS**

Em serviços orientado a conexão é necessário uma rede de circuitos virtuais, seu objetivo é evitar a necessidade de escolher uma nova rota para cada pacote enviado. Em vez disso quando uma conexão é estabelecida, escolhe-se uma rota desde a máquina de origem até a máquina de destino, como parte da configuração de conexão e essa rota é armazenada em tabelas internas do roteador. Uma vez armazenada a rota ela é utilizada por todo o tráfego que flui pela conexão. Quando a conexão é liberada, o circuito virtual é encerrado.

Um circuito virtual consiste em:

(1) Um caminho entre hospedeiros de origem e de destino.

(2) Números de CVs (Os identificadores), um número para cada enlace ao longo do caminho.

(3) Registros na tabela de repasse em cada roteador ao longo do caminho.

Um circuito virtual é dividido em três fases.

1. Estabelecimento de CV – nesta fase é realizado o início da chamada. A camada de transporte remetente contata a camada de rede, especifica o endereço do receptor e espera até a rede estabelecer o CV. A camada de rede determina o caminho entre remetente e destinatário, ou seja, a série de enlaces e roteadores pelos quais todos os pacotes do CV trafegarão.

2. Transferência de dados – os pacotes começam a fluir ao longo da CV.

3. Encerramento do CV – isto ocorre quando remetente ou destinatário informa à camada de rede seu desejo de desativar a CV. Desse modo a camada de rede informará o sistema final do outro lado e atualizará as tabelas de repasse de cada roteador informando que aquele CV não existe mais.

Quando tratamos de uma rede de circuitos virtuais, os roteadores da rede devem manter as informações do estado da conexão para as conexões em cursos. Toda vez que uma nova conexão for estabelecida através de um roteador, um novo registro de conexão deve ser adicionado a tabela de repasse do roteador. O mesmo ocorre quando uma conexão for desativada, um registro deve ser removido da tabela de repasses. As mensagens utilizadas para iniciar finalizar um CV entre os roteadores para estabelecer o CV são conhecidas como mensagens de sinalização e os protocolos usados para trocá-las costumam ser denominados protocolos de sinalização.

**2.2. REDE DE DATAGRAMAS**

A rede datagramas é um serviço sem conexão, quando o sistema quer enviar um pacote ele marca o pacote com endereço do sistema final de destino e o envia para dentro da rede. Todo o processo é feito sem o estabelecimento de nenhum circuito virtual.

Nesta rede os pacotes atravessam vários roteadores e cada um desses roteadores usa o endereço de destino do pacote para repassá-lo. Especificamente, cada roteador tem uma tabela de repasse que mapeia endereços de destino interfaces de enlaces; quando um pacote chega ao roteador, este usa o endereço de destino do pacote para procurar a interface de enlace de saída apropriada na tabela de repasse. Os Roteadores em uma rede de datagramas não mantêm nenhuma informação de estado sobre CVs, até porque não a nenhuma informação da conexão para ser guardada porém mantém informações do estado de repasse em suas tabelas de repasse.

**2.3. REDE DE CIRCUITOS VIRTUAIS X DATAGRAMAS**

Tanto os circuitos virtuais quanto datagramas tem seus prós e contras. O uso de circuitos virtuais exige uma fase de configuração, que leva tempo e consome recursos, mas depois de feito torna-se fácil descobrir o que fazer com um pacote de dados em uma rede de CV. Já em uma rede de datagramas, nenhuma configuração é necessária, porém é preciso um procedimento de pesquisa mais complicado para mapear o endereço de destino.

Uma rede de datagramas precisa ter uma entrada para cada destino possível, enquanto uma rede de circuitos virtuais só precisa de uma entrada para cada circuito virtual. No entanto, essa vantagem é um pouco ilusória, pois pacotes de configuração de conexões também têm de ser roteados e, da mesma forma que os datagramas, eles usam endereços de destino.

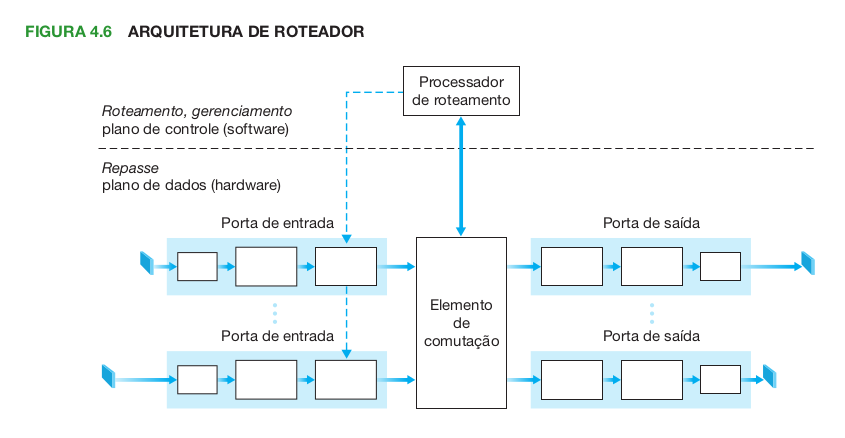
|  |  |  |
| --- | --- | --- |
| **Questão** | **Rede de datagramas** | **Rede de circuitos virtuais** |
| Configuração de circuitos | Desnecessária | Obrigatória |
| Endereçamento | Cada pacote contém os endereços completos de origem e de destino | Cada pacote contém um pequeno número do circuito virtual |
| Informações sobre o estado | Os roteadores não armazenam informações sobre o estado das conexões | Cada circuito virtual requer espaço em tabelas de roteadores por conexão |
| Roteamento | Cada pacote é roteado independentemente | A rota é escolhida quando o circuito virtual é estabelecido; todos os pacotes seguem essa rota |
| Efeito de falhas no roteador | Nenhum, com exceção dos pacotes perdidos durante a falha | Todos os circuitos virtuais que tiverem passado pelo roteador que apresentou o defeito serão encerrados |
| Qualidade de serviço | Difícil | Fácil, se for possível alocar recursos suficientes com antecedência para cada circuito virtual |
| Controle de congestionamento | Difícil | Fácil, se for possível alocar recursos suficientes com antecedência para cada circuito virtual |

O QUE HÁ DENTRO DE UM ROTEADOR?

Agora que já tivemos uma visão geral das funções e serviços da camada de rede, voltaremos nossa atenção para a função de repasse, a transferência, propriamente dita, de pacotes dos enlaces de entrada até os enlaces de saída adequados de um roteador. A saber, endereçamento e correspondência com o prefixo mais longo. Mencionamos, de passagem, que os pesquisadores e profissionais de redes de computadores usam as palavras repasse e comutação indistintamente; nós usaremos ambos os termos neste Trabalho.

Uma visão de alto nível da arquitetura de um roteador genérico é mostrado quatro componentes de um roteador podem ser identificados:

* **Portas de entrada.** A porta de entrada tem diversas funções. Ela realiza as funções de camada física de terminar um enlace físico de entrada em um roteador. Executa também as de camada de enlace necessárias para interoperar com as funções da camada de enlace do outro lado do enlace de entrada. Talvez mais importante, a função de exame também é realizada na porta de entrada; isso ocorrerá na caixa mais à direita da porta de entrada. É aqui que a tabela de repasse é consultada para determinar a porta de saída do roteador à qual um pacote que chega será repassado por meio do elemento de comutação. Pacotes de controle (por exemplo, pacotes carregando informações de protocolo de roteamento) são repassados de uma porta de entrada até o processador de roteamento. Note que o termo porta aqui refere-se às interfaces físicas de entrada e saída do roteador.
* **Elemento de comutação.** O elemento de comutação conecta as portas de entrada do roteador às suas portas de saída. Ele está integralmente contido no interior do roteador, uma rede dentro de um roteador da rede!
* **Portas de saída.** Uma porta de saída armazena os pacotes que foram repassados a ela através do elemento de comutação e, então, os transmite até o enlace de saída, realizando as funções necessárias da camada de enlace e da camada física. Quando um enlace é bidirecional (isto é, carrega um tráfego em ambas as direções), uma porta de saída para o enlace será emparelhada com a porta de entrada para esse enlace na mesma placa de linha (uma placa de circuito impresso contendo uma ou mais portas de entrada, e que está conectada ao elemento de comutação).
* **Processador de roteamento.** O processador de roteamento executa os protocolos de roteamento, mantém as tabelas de roteamento e as informações de estado do enlace, e calcula a tabela de repasse para o roteador. Ele também realiza funções de gerenciamento de rede.



**O PROTOCOLO DA INTERNET (IP)**

A camada de rede é formada pelos protocolos IP, Protocolos de roteamento e Protocolo de comunicação de erro e de informações da internet (ICMP). O IP é responsável por endereçar e encaminhar os pacotes que trafegam pela rede. Atualmente existem duas versões do IP, o IPv4 e IPv6. Na camada de rede esse pacote é chamado de datagrama. O datagrama é dividido em duas partes, o cabeçalho e os dados. No cabeçalho existem todas as informações que são usadas para o roteador saber para quem mandar os pacotes como que tipo de informações contém o pacote. Ex: UDP ou TCP.

**4.4.1 Formato de datagrama**

Os componentes que compõem o cabeçalho do IPv4 são:

·**Número da versão**: Indicam qual versão do IP está sendo usada. IPv4 ou IPv6.

·**Comprimento do cabeçalho**: Indicam onde no IPv4 os dados que são transportados de fato começam. Ele existe pelo fato de ser possível colocar opções que aumentam o tamanho do datagrama. No IPv6 o campo opções foi retirado.

· **Tipo de Serviço**: Serve para diferenciar os diferentes tipos de datagrama IP. Ex: diferenciar datagramas de tempo real dos que não são de tempo real.

· **Comprimento do datagrama**: Diz respeito ao tamanho total do datagrama, medido em bytes, o tamanho máximo do datagrama IP é de 65.535 bytes.

· **Identificador, flags, deslocamento de fragmentação**: São usados quando é preciso fazer a fragmentação dos pacotes. No IPv6 não existe esses campos.

· **Tempo de vida** (TTL): Serve para eliminar um datagrama que não chegou ao seu destino. Toda vez que ele passa por um roteador seu valor é diminuído.

· **Protocolo**: Usado quando o datagrama chega ao destino final. Seu valor indica o protocolo de camada de transporte específico ao qual a porção do datagrama IP deve ser passada. Ex: UDP, TCP.

· **Soma da verificação do cabeçalho**: Tem a função de auxiliar o roteador a descobrir erros em um datagrama IP. O roteador calcula o valor da soma de verificação e se o valor carregado no cabeçalho não for igual então existe um erro e ele descarta o datagrama.

· **Endereços IP origem e destino**: Armazena os IPs da máquina que mandou e vai receber o pacote.

· **Opções**: Permite que o cabeçalho seja estendido. Não é um campo muito usado. Também foi retirado na versão IPv6 por aumentar o processamento por parte dos roteadores e ficar difícil determinar onde de fato começa os dados no datagrama.

· **Dados**: contém o segmento da camada de transporte (TCP ou UDP) a ser entregue ao destino. Ele também pode carregar outros tipos de dados como ICMP.

**Fragmentação do datagrama IP**

A fragmentação do datagrama ocorre porque alguns protocolos podem transportar datagramas grandes, mas outros não podem. A quantidade máximo que um quadro (**quadro** é a informação que é traduzida e transmitida entre dois ou mais pontos de **rede** como uma unidade de dados) pode carregar é chamada de Unidade Máxima de Transmissão (MTU). O datagrama é encapsulado dentro do quadro de camada de enlace para ser transportado de um roteador até o roteador seguinte. Se um pacote tem tamanho maior do que o suportado pela MTU então o roteador faz a fragmentação do datagrama para tamanhos menores e os envia. Para que os dados chequem o roteador usa os campos identificação, flag e deslocamento de fragmentação do cabeçalho do datagrama IP.

**4.4.2 Endereçamento IPv4**

Cada os computadores ou máquinas (hospedeiros) e roteadores que estão interconectados entre si possuem uma interface, essa interface é o limite entre o computador e o enlace físico. Os roteadores também tem uma interface entre eles e os demais enlaces físicos que recebem ou passam dados. Para que não haja confusão na hora de enviar e receber dados o IP exige que todos os hospedeiros e roteadores possuam um endereço IP único. Esse endereço é formado por 32 bits. Existem aproximadamente cerca de 4 bilhões de endereços IP possíveis. Esse endereço é escrito na forma decimal e separado dos outros bytes do endereço por um ponto.

Apesar de ter um endereço IP ser exclusivo para cada hospedeiro ou roteador, eles não podem ser escolhidos aleatoriamente. Um roteador pode ter mais de uma interface de hospedeiro conectada a sua interface, quando isso acontece é formada uma sub – rede. Mais de uma sub – rede pode estar conectada a um mesmo roteador. Desse modo uma instituição acadêmica por exemplo pode ter várias sub – redes e todos os equipamentos conectados nela terão o mesmo endereço de sub – rede.

Para ter um bloco de endereços IP para usar em uma subrede de uma empresa, o administrador da rede pode pedir ao seu Provedor que então forneceria endereços IP a partir de um bloco maior. Para o Provedor conseguir seus endereços para distribuir ele deve obter os endereços de Entidades responsáveis por gerenciar o espaço de endereços IP e alocar ISPs.

**IPv6**

No começo da década de 1990, a IETF (Internet Engineering Task Force) iniciou um esforço para desenvolver o sucessor do protocolo IPv4. Uma motivação importante para isso foi o entendimento de que o espaço de endereços IP de 32 bits estava começando a escassear, com novas sub-redes e nós IP sendo anexados à Internet (e ainda recebendo endereços IP exclusivos) a uma velocidade estonteante. Para atender a essa necessidade de maior espaço para endereços IP, foi desenvolvido um novo protocolo IP, o IPv6. Os projetistas do IPv6 também aproveitaram essa oportunidade para ajustar e ampliar outros aspectos do IPv4, com base na experiência operacional acumulada sobre esse protocolo.

O momento em que todos os endereços IPv4 estariam alocados (e, por conseguinte, mais nenhuma sub-rede poderia ser ligada à Internet) foi objeto de considerável debate. Os dois líderes do grupo de trabalho de Expectativa de Tempo de Vida dos Endereços (Address ­Lifetime Expectations) da IETF estimaram que os endereços se esgotariam em 2008 e 2018, respectivamente [Solensky, 1996]. Em fevereiro de 2011, a IANA alocou o último conjunto restante de endereços IPv4 a um registrador regional. Embora esses registradores ainda tenham endereços IPv4 disponíveis dentro de seus conjuntos, quando esses endereços se esgotarem, não haverá mais blocos de endereços disponíveis para serem alocados a partir de um conjunto central [Houston, 2011a]. Embora as estimativas de esgotamento de endereço IPv4 de meados de 1990 sugerissem que poderia se passar um longo tempo até que o espaço de endereços do IPv4 fosse esgotado, ficou claro que seria necessário um tempo expressivo para disponibilizar uma nova tecnologia em escala tão gigantesca. Assim, foi dado início ao esforço denominado Próxima Geração do IP (Next Generation IP — IPng) [Bradner, 1996; RFC 1752]. O resultado foi a ­especificação IP versão 6 (IPv6) [RFC 2460]. (Uma pergunta recorrente é o que aconteceu com o IPv5. Foi proposto de início que o protocolo ST-2 se tornasse o IPv5, porém, mais tarde, esse protocolo foi descartado.) Excelentes fontes de informação sobre o IPv6 podem ser encontradas em Huitema [1998] e IPv6 [2012].

**Formato do datagrama IPv6**

O formato do datagrama IPv6 é mostrado na Figura 4.24. As mudanças mais importantes introduzidas no IPv6 ficam evidentes no formato do datagrama:

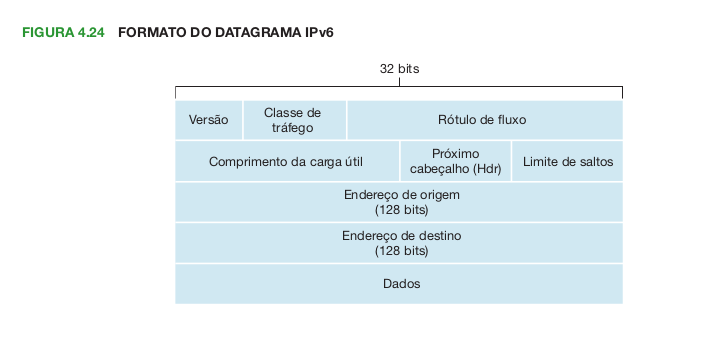
* **Capacidade de endereçamento expandida.** O IPv6 aumenta o tamanho do endereço IP de 32 bits para 128 bits. Isso garante que o mundo não ficará sem endereços IP. Agora, cada grão de areia do planeta pode ter um endereço IP. Além dos endereços para um grupo de individuais, o IPv6 introduziu um novo tipo de endereço, denominado endereço para qualquer membro do grupo (anycast), que permite que um

datagrama seja entregue a qualquer hospedeiro de um grupo. (Essa característica poderia ser usada, por exemplo, para enviar uma mensagem HTTP GET ao site mais próximo de um conjunto de sites espelhados que contenham um dado documento.)

* **Cabeçalho aprimorado de 40 bytes.** Como discutiremos adiante, vários campos IPv4 foram descartados ou tornaram-se opcionais. O cabeçalho de comprimento fixo de 40 bytes resultante permite processamento mais veloz do datagrama IP. Uma nova codificação de opções permite um processamento de

opções mais flexível.

* **Rotulação de fluxo e prioridade.** O IPv6 tem uma definição dúbia de fluxo. O RFC 1752 e o RFC 2460 declaram que isso permite “rotular pacotes que pertencem a fluxos particulares para os quais o remetente requisita tratamento especial, tal como um serviço de qualidade não padrão ou um serviço de tempo real”. Por exemplo, a transmissão de áudio e vídeo seria tratada como um fluxo. Por outro lado, aplicações mais tradicionais, como transferência de arquivos e e-mail, poderiam não ser tratadas assim. É possível que o tráfego carregado por um usuário de alta prioridade (digamos, alguém que paga por um serviço melhor de tráfego) seja também tratado como um fluxo. O que fica claro, contudo, é que os projetistas do IPv6 preveem a possível necessidade de conseguir diferenciá-los, mesmo que o exato significado de fluxo ainda não tenha sido determinado. O cabeçalho IPv6 também tem um campo de 8 bits para classe de tráfego. Assim, como o campo TOS do IPv4, ele pode ser usado para dar prioridade a certos datagramas em um fluxo ou a datagramas de certas aplicações (por exemplo, pacotes ICMP) em relação aos de outras (por exemplo, notícias pela rede).



Como foi observado na figura anterior, Os seguintes campos são definidos no IPv6:

• **Versão.** Esse campo de 4 bits identifica o número da versão do IP. Não é surpresa que o IPv6 tenha o valor 6. Note que colocar 4 nesse campo não cria um datagrama IPv4 válido. (Se criasse, a vida seria bem mais simples — veja a discussão mais adiante, referente à transição do IPv4 para o IPv6.)

• **Classe de tráfego.** Esse campo de 8 bits tem função semelhante à do campo TOS que vimos no IPv4.

• **Rótulo de fluxo.** Como já discutimos, esse campo de 20 bits é usado para identificar um fluxo de datagramas.

• **Comprimento da carga útil.** Esse valor de 16 bits é tratado como um número inteiro sem sinal que dá o número de bytes no datagrama IPv6 que se segue ao pacote do cabeçalho, que tem tamanho fixo de 40 bytes.

• **Próximo cabeçalho.** Esse campo identifica o protocolo ao qual o conteúdo (campo de dados) desse datagrama será entregue (por exemplo, TCP ou UDP). Usa os mesmos valores do campo de protocolo no cabeçalho IPv4.

• **Limite de saltos.** O conteúdo desse campo é decrementado em um para cada roteador que repassa o datagrama. Se a contagem do limite de saltos chegar a zero, o datagrama será descartado.

• **Endereços de origem e de destino.** Os vários formatos do endereço de 128 bits do IPv6 são descritos no RFC 4291.

• **Dados.** Esta é a parte da carga útil do datagrama IPv6. Quando este alcança seu destino, a carga útil pode ser extraída do datagrama IP e passada adiante para o protocolo especificado no campo de próximo cabeçalho.

**Transição do IPv4 para o IPv6**

Agora que vimos os detalhes técnicos do IPv6, vamos tratar de um assunto muito prático: como a Internet pública, que é baseada no IPv4, fará a transição para o IPv6? O problema é que, enquanto os novos sistemas habilitados para IPv6 podem ser compatíveis, isto é, podem enviar, rotear e receber datagramas IPv4, os sistemas habilitados para IPv4 não podem manusear datagramas IPv6. Há várias opções possíveis [Huston 2011b].

Uma opção seria determinar um “dia da conversão” — uma data e um horário definidos em que todas as máquinas da Internet seriam desligadas e atualizadas, passando do IPv4 para o IPv6. A última transição importante de tecnologia (do uso do NCP para o uso do TCP para serviço confiável de transporte) ocorreu há quase 25 anos. E, mesmo naquela época [RFC 801], quando a Internet era pequenina e ainda gerenciada por um número reduzido de “sabichões”, ficou claro que esse “dia da conversão” não era possível. Um dia assim, envolvendo centenas de milhões de máquinas e milhões de administradores e usuários de rede, é ainda mais impensável hoje. O RFC 4213 descreve duas abordagens (que podem ser usadas independentemente ou em conjunto) para a integração gradual dos hospedeiros e roteadores IPv4 ao mundo IPv6 (com a meta de longo prazo de fazer a transição de todos os nós IPv4 para IPv6).

Provavelmente, a maneira mais direta de introduzir nós habilitados ao IPv6 seja uma abordagem de pilha dupla, em que nós IPv6 também tenham uma implementação IPv4 completa. Esse nó, denominado nó IPv6/IPv4 no RFC 4213, estaria habilitado a enviar e receber datagramas tanto IPv4 quanto IPv6. Ao interagir com um nó IPv4, um nó IPv6/IPv4 poderá usar datagramas IPv4; ao interagir com um nó IPv6, poderá utilizar IPv6. Nós IPv6/IPv4 devem ter endereços IPv6 e IPv4. Além disso, devem ser capazes de determinar se outro nó é habilitado para IPv6 ou apenas para IPv4. Esse problema pode ser resolvido usando o DNS, que poderá retornar um endereço IPv6 se o nome do nó a ser resolvido for capacitado para IPv6.

**Uma breve investida em segurança IP**

Ao ler este trabalho, você pode ter percebido que nenhum serviço de segurança foi mencionado. De fato, o IPv4 foi projetado em uma era (anos 1970) em que a Internet era utilizada, principalmente, entre pesquisadores de rede mutuamente confiáveis. Criar uma rede de computadores que integrava uma grande quantidade de tecnologias da camada de enlace já era desafiador, sem ter de se preocupar com a segurança.

Contudo, com a segurança sendo a principal preocupação hoje, pesquisadores da Internet começaram a criar novos protocolos da camada de rede que oferecem variados serviços de segurança. Um desses protocolos é o IPsec, um dos protocolos da camada de rede mais conhecidos e empregados em Redes Virtuais Privadas (VPNs). Embora o IPsec e seus suportes sejam abordados no Capítulo 8, apresentamos uma breve introdução sobre seus serviços nesta seção.

O IPsec foi desenvolvido para ser compatível com o IPv4 e o IPv6. Em particular, para obter os benefícios do IPv6, não precisamos substituir as pilhas dos protocolos em todos os roteadores e hospedeiros na Internet. Por exemplo, usando o modo transporte (um dos “modos” do IPsec), se dois hospedeiros querem se comunicar em segurança, o IPsec precisa estar disponível apenas nesses dois. Todos os outros roteadores e hospedeiros podem continuar a rodar o IPv4 simples.

Para uma abordagem concreta, a partir daqui focaremos no modo de transporte do IPsec. Nesse modo, dois hospedeiros estabelecem, primeiro, uma sessão IPsec entre si mesmos. (Assim, o IPsec é orientado a conexão!) Com a sessão pronta, todos os segmentos TCP e UDP enviados entre os dois hospedeiros aproveitam os serviços de segurança fornecidos pelo IPsec. No lado remetente, a camada de transporte passa um segmento para o IPsec. Este, então, codifica o segmento, acrescenta campos de segurança adicionais a ele e envolve a carga útil resultante em um datagrama IP comum. Depois, o hospedeiro remetente envia o datagrama para a Internet, a qual o transporta ao destinatário. Em seguida, o IPsec decodifica o segmento e o encaminha à camada de transporte.

Os serviços oferecidos por uma sessão IPsec incluem:

* **Acordo criptográfico.** Mecanismos que permitem que dois hospedeiros de comunicação concordem nos algoritmos criptográficos e chaves.
* **Codificação das cargas úteis do datagrama IP.** Quando o hospedeiro destinatário recebe um segmento da camada de transporte, o IPsec codifica a carga útil, que pode ser somente decodificada pelo IPsec no hospedeiro destinatário.
* **Integridade dos dados.** O IPsec permite que o hospedeiro destinatário verifique se os campos do cabeçalho do datagrama e a carga útil codificada não foram modificados enquanto o datagrama estava no caminho da origem ao destino.
* **Autenticação de origem.** Quando um hospedeiro recebe um datagrama IPsec de uma origem confiável (com uma chave confiável, o hospedeiro está certo de que o endereço IP remetente no datagrama é a verdadeira origem do datagrama.

Quando dois hospedeiros estabelecem uma sessão IPsec, todos os segmentos TCP e UDP enviados entre eles serão codificados e autenticados. O IPsec, portanto, oferece uma cobertura geral, protegendo toda a comunicação entre os dois hospedeiros para todas as aplicações de rede.

**PROTOCOLO DE CONFIGURAÇÃO DINÂMICA DE HOSPEDEIROS (DHCP)**

O protocolo DHCP tem a função de atribuir um endereço IP a um hospedeiro de maneira automática. Ele possui um grupo de endereços em sua tabela e todo vez que um dispositivo se conecta na rede é atribuído um endereço temporário a ele. Quando é feita a desconexão o endereço volta a tabela de endereços disponíveis. Os passos que um hospedeiro recém – chegado faz para obter um endereço IP são:

· Descoberta de servidor DHCP: o dispositivo manda uma mensagem de descoberta DHCP dentro de um pacote UDP para a porta 67 com endereço IP de destino de difusão 255.255.255.255 e endereço IP destinatário 0.0.0.0.

· Oferta do servidor DHCP: quando o servidor DHCP recebe a mensagem de descoberta ele responde com uma mensagem de oferta DHCP, para todos os nós da sub – rede utilizando o IP 255.255.255.255. Na oferta é enviado ao cliente ID de transação da mensagem de descoberta recebida, endereço IP proposto, máscara de rede e tempo de concessão do endereço IP.

· Solicitação DHCP: o cliente escolhe a oferta com uma mensagem de solicitação DHCP.

· DHCP ACK: O servidor responde à mensagem de requisição com a mensagem DHCP ACK confirmando os parâmetros.

**TRADUÇÃO DE ENDEREÇOS NA REDE (NAT)**

Com o aumento das sub – redes particulares, aumenta também o número de endereços IP que um ISP deve fornecer, acontece que pode ocorrer que não tenha endereços IP disponíveis para todos os que se conectam já que um ISP possui um número limitado de endereços para fornecer aos clientes.

Para que seja possível se conectar e ao mesmo tempo preservar endereços foi criada a NAT, ela tem a função de permitir que essas sub – redes particulares usam endereços IP que não são registrados na tabela de endereços do ISP. Para que isso seja possível existe um roteador que conecta duas redes entre si e converte os endereços privados e não exclusivos em endereços legais. Assim o roteador passa a atuar entre a internet e a rede local com apenas um único endereço IP.

PROTOCOLO DE MENSAGENS DE CONTROLE DA INTERNET (ICMP)

Esse protocolo é usado por hospedeiros e roteadores para comunicar, entre si, informações de camada de rede, em sua maioria para comunicar sobre erros. Ele contém um campo de tipo e um de com o código da mensagem. Ex: tipo 3, código 0, descrição: rede de destino inalcançável.

# 4.4.4 IPv6

# 4.5 Algoritmos de roteamento

O algoritmo de roteamento é a parte do software da camada de rede responsável pela decisão sobre a linha de saída a ser usada na transmissão do pacote de entrada. É usado para encontrar o caminho de menor custo possível entre dois roteadores.

**Classificação dos algoritmos de roteamento**

Há algumas formas de classificar/agrupar os algoritmos de roteamento, sendo elas:

I – Por lugar de roteamento:

a) Algoritmo de roteamento global: calcula o caminho de menor custo entre uma fonte e um destino usando conhecimento completo e global sobre a rede. O cálculo pode ser rodado em um local ou duplicado em vários locais.

b) Algoritmo de roteamento descentralizado: calcula o caminho de menor custo de modo interativo e distribuído. Cada nó começa sabendo apenas os custos dos enlaces diretamente ligados a ele.

II – Por tipo roteamento:

a) Não adaptativos: também conhecidos por estáticos. Só mudam por intervenção humana direta. Normalmente está em redes locais.

b) Adaptativos: mudam suas decisões de roteamento para refletir mudanças na topologia e no tráfego. Estes diferem em termos de lugar em que obtêm suas informações para definir as métricas para a otimização do encaminhamento dos pacotes.

**Algoritmo de roteamento *Link State (LS)***

No algoritmo de roteamento *Link State* (estado de enlace) a topologia da rede e todos os custos de enlace são conhecidos e ficam disponíveis para o algoritmo, sendo que todos os nós transmitem pacotes de estado de enlace à todos os outros nós da rede.

A transmissão acontece por difusão de estado de enlace, todos os nós tem uma visão idêntica e completa da rede, além disso, todos os roteadores rodam o algoritmo de estado de enlace e calculam o mesmo conjunto de caminhos de menor custo. Os updates são enviados para toda a rede(broadcast).

É conhecido como algoritmo de *Dijkstra*. Ele calcula o caminho de menor custo entre um nó (origem determinada em **u**) e todos os demais nós da rede de roteadores.

O algoritmo de *Dijkstra* pode ser definido da seguinte forma: um algoritmo interativo (após várias exceções determinadas possui uma saída) possui a propriedade de, após a \*k-\*ésima interação conhece os caminhos de menor custo para k nós de destino.

**Notação Matemática:**

· D(v): custo do caminho de menor custo entre o nó de origem e o destino “**v**” até essa interação (resolução) do algoritmo;

· P(v): nó anterior (vizinho do destinatário “v”) de acordo com o caminho de menor custo de a origem até **v**;

· N’: subconjunto de nós. **V** pertence a **N’** se o caminho de menor custo entre a origem e o destino for realmente conhecido.

· D(v) = min( D(v), D(w) + c(w,v) )

Os passos para o cálculo do custo de menor custo de rota são os seguintes:

Passo 1: Os caminhos não são conhecidos:

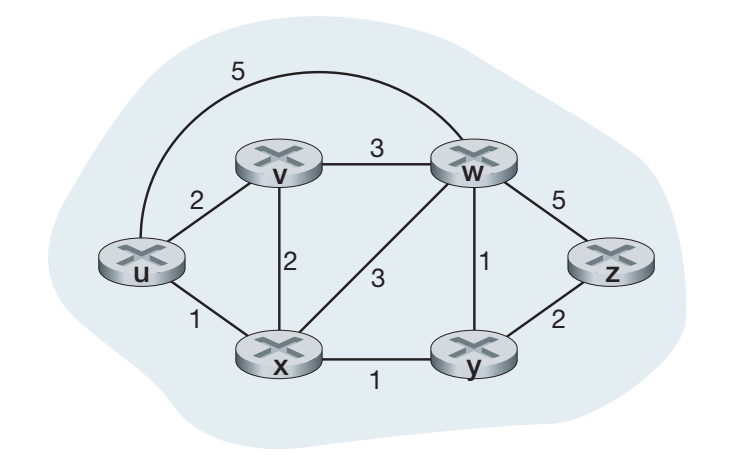
- Atribuir o valor infinito a todos os nós;

Passo 2: À medida que os caminhos são encontrados:

- Mudam-se os rótulos, refletindo os melhores caminhos, porém ainda são provisórios;

- Se for o melhor caminho o rótulo se torna permanente.

**Figura 1 - Modelo abstrato de grafo de uma rede de computadores**

****

Fonte: KUROSE, JAMES F. Redes de computadores e a Internet: uma abordagem top-down, 6ª edição.

O algoritmo, em questão, consiste em uma etapa de inicialização seguida de um loop, sendo que o número de vezes que o loop é rodado é igual ao número de nós na rede. Após a realização dos loops é possível obter os seguintes resultados:

**Tabela 1 - Resultado por loop do algoritmo Dijkstra**



**Algoritmo de roteamento *Distance-vector (DV)***

Enquanto o algoritmo LS usa informação global, o algoritmo de vetor de distâncias é:

a) Distribuído: cada nó recebe alguma informação de um ou mais vizinhos diretamente ligado a ele (largura de banda, congestionamento etc);

b) Iterativo: porque esse processo continua até que mais nenhuma informação seja trocada entre vizinhos;

c) Assíncrono: não requer que todos os nós rodem simultaneamente.

A equação de *Bellman-Ford* fornece a solução para os registros das tabelas de repasse do nó ao seu vizinho, na seguinte sequência em forma de loop: **espera** por mudança no custo do enlace local na mensagem do vizinho, **recalcula** estimativas e, se o DV para qualquer destino mudou, **notifica** os vizinhos.

Dx(y) = minv{c(x,v) + Dv(y)} para cada nó Y ∊ N.

Analisando o mesmo modelo abstrato de grafo obtemos os seguintes resultados:

Du(z) = min{c(u,v)+dv(z),c(u,x)+dx(z),c(u,w)+dw(z)}

Du(z) = min{ 2 + 5,1 + 3, 5 + 3 }

Du(z) = min{ 7 , 4 , 8 } = 4

No *distance vector* os updates são periódicos a cada 30 a 90 segundos. Enviando apenas para os seus vizinhos o custo do enlace baseado no # de hops, com isso pode carregar informações erradas, já que confiam nos seus vizinhos.

**Roteamento Hierárquico**

Nos algoritmos LS e DC, consideramos a rede simplesmente como uma coleção de roteadores interconectados, todos rodando o mesmo algoritmo. Na prática, temos:

a) Escalabilidade: aumento no número de roteadores, sobrecarga relativa ao cálculo, ao armazenamento e à comunicação da tabela de roteamento;

b) Autonomia administrativa: empresas desejam controlar seus roteadores como bem entendem;

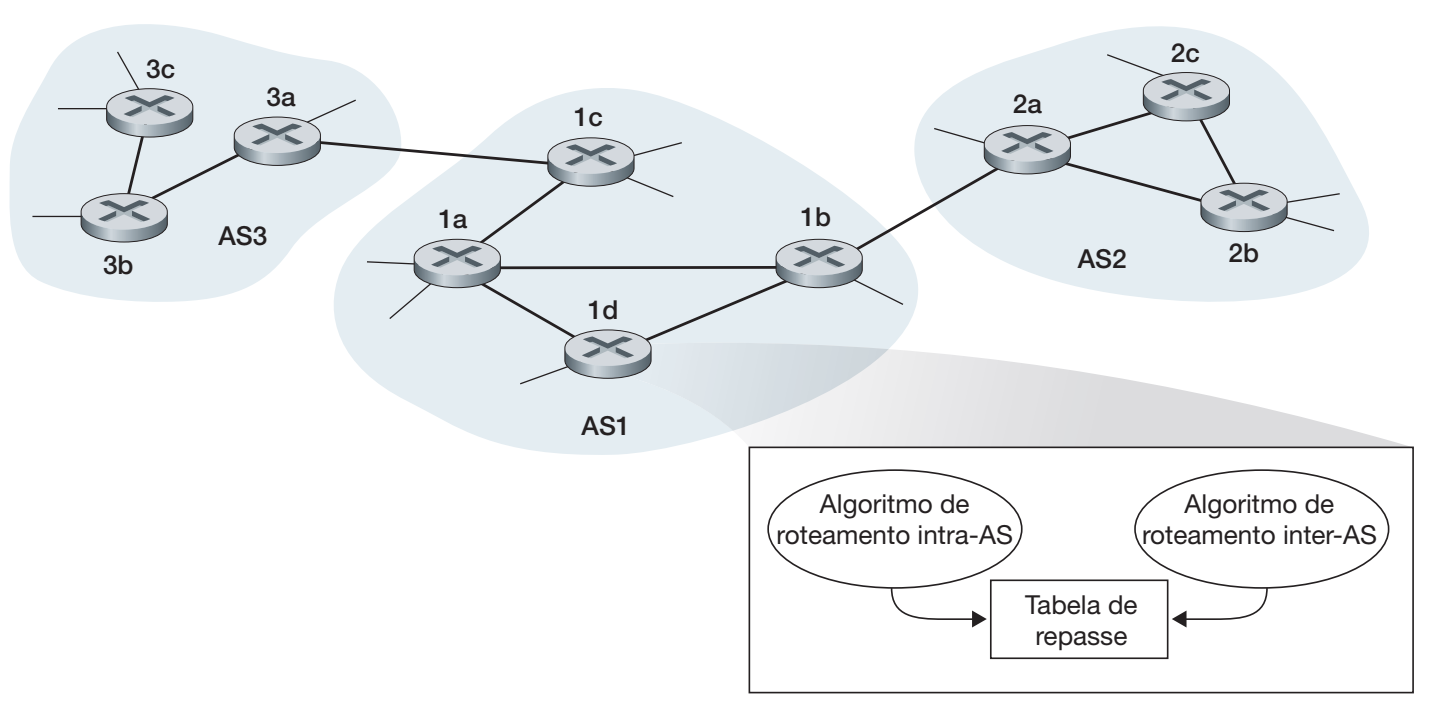
Nesse sentido surge o roteamento hierárquico, na qual agrupamos, portanto, os roteadores em:

a) Sistemas autônomos (*autonomous systems* – AS): roteadores sob o mesmo controle administrativo rodando o mesmo algoritmo de roteamento (roteamento intrasistema autônomo);

b) Roteadores de borda (*gateway routers*): roteadores responsáveis em conectar os AS entre si.

O algoritmo de roteamento que roda dentro de um AS é denominado de protocolo de roteamento intrassistema. Os roteadores são autônomos dentro da AS, definido pelo administrador da rede, e dispõem das informações sobre cada um dos outros.

**Figura 2 - Exemplo de sistemas autônomos interconectados**

****

Fonte: KUROSE, JAMES F. Redes de computadores e a Internet: uma abordagem top-down, 6ª edição.

# 4.6 Roteamento na Internet

**Roteamento intra-AS na Internet:**

A internet é composta de um grande número de redes independentes, ou sistemas autônomos, que são operados por diferentes organizações, normalmente um empresa, universidade ou ISP. Um protocolo de roteamento insta-AS é usado para determinar como é rodado o roteamento dentro de um sistema autônomo (AS). Esses protocolos são também conhecidos como protocolos de roteadores internet, os IGP - interior gateway protocols,

Historicamente, dois protocolos de roteamento têm sido usados para roteamento dentro de um sistema autônomo na Internet: o protocolo de informações de roteamento, RIP (Routing Information Protocol) e o OSPF (Open Shortest Path First). Cada um desses protocolos utilizam algoritmos de roteamento. O algoritmo de vetor de distâncias é utilizado pelo RIP. O estado de Enlace é usado pelo OSPF e o roteamento hierárquico é usado pelo BGP.

**RIP Routing Information Protocol**

O RIP foi um dos primeiros protocolos de roteamento intra-AS da Internet, e seu uso é ainda mais disseminado. Sua origem vem da arquitetura XNS ((Xerox Network Systems). O RIP é um protocolo de vetor de distâncias que funciona de modo parecido com o protocolo DV. Ele se baseia na contagem de saltos como métrica de custo, isto é, cada enlace tem um custo, porém ele executa um número limitado de saltos que no máximo são 15 saltos. Cada roteador mantém uma tabela RIP, denominada tabela de roteamento. A tabela de roteamento de um roteador inclui o vetor de distâncias e a tabela de repasse desse roteador.

Por ser um vetor de distância as tabelas são intercambiadas tipicamente a cada 30 segundos, ou seja, são atualizadas as informações de roteamento entre os vizinhos por meio de mensagem de respostas, estas mensagens são ICMP que trata dos erros, controles e anúncios da própria redes. Caso algum desses vizinhos fique por mais de 180 segundos sem nenhum anúncio, então ele considera um vizinho morto.

As tabelas de roteamento RIP são controlados por processo do nível de aplicação chamado de routed, ficando responsável pelas atualizações da tabela de repasse. Porém a tabela de repasse se encontra na camada de rede enquanto a aplicação routed está na camada de aplicação. Para que haja esse controle é preciso da camada de transporte utilizando o protocolo UDP para fazer o encaminhamento o encaminhamento das tabelas de roteamento do RIP.

**OSPF Open Shortest Path First**

O OSPF é um protocolo de roteamento de estado de enlace. Durante seu projeto, os desenvolvedores deveriam alcançar a listas de objetivos a ser alcançados.

Primeiro, o algoritmo tinha de ser publicado na literatura aberta, daí o ‘O’ (Open) em OSPF. Uma solução patenteada, pertencente a uma empresa, não serviria.

Em segundo lugar, o novo protocolo deveria dar suporte a uma série de métricas de distância, incluindo distância física, atraso e assim por diante.

Em terceiro lugar, ele tinha de ser um algoritmo dinâmico, que se adaptasse às mudanças na topologia de maneira automática e rápida.

Em quarto lugar, e novo para o OSPF, ele tinha de dar suporte ao roteamento com base no tipo de serviço. O novo protocolo deveria ser capaz de rotear o tráfego em tempo real de uma maneira e o restante do tráfego de outra. Na época, o IP tinha um campo Tipo de serviço, mas nenhum protocolo de roteamento existente o utilizava. Esse campo foi incluído no OSPF, mas ninguém o usava ainda, e, por fim, foi removido. Talvez esse requisito tenha estado à frente de seu tempo, e precedeu o trabalho da IETF sobre serviços diferenciados, que rejuvenesceu as classes de serviço.

Em quinto lugar, e relacionado ao anterior, o OSPF tinha de realizar balanceamento de carga, dividindo-a por várias conexões. A maioria dos protocolos anteriores enviava todos os pacotes por uma única melhor rota, mesmo que houvesse duas rotas que fossem igualmente boas. A outra rota nem sequer era usada. Em muitos casos, a divisão da carga por várias rotas oferece melhor desempenho.

Em sexto lugar, o suporte para sistemas hierárquicos era necessário. Em 1988, algumas redes tinham se tornado tão grandes que não se poderia esperar que algum roteador conhecesse a topologia inteira. O OSPF tinha de ser projetado de modo que nenhum roteador precisasse disso.

Em sétimo lugar, algum modo de segurança era exigido para impedir que estudantes procurando diversão bisbilhotarem os roteadores enviando-lhes informações de roteamento falsas. Finalmente, era preciso algum meio de lidar com os roteadores que estavam conectados à Internet por meio de um túnel. Os protocolos anteriores não cuidavam disso muito bem.

**BGP Border Gateway Protocol**

O BGP, sobretudo, permite que cada sub-rede anuncie sua existência ao restante da Internet. Uma sub-rede anuncia sua localização e o BGP garante que todos os ASs da Internet saibam de sua existência e como chegar até ela. Não fosse o BGP, cada sub-rede ficaria isolada sozinha e desconhecida pelo restante da Internet.

A BGP oferece a cada sistema autônomo um meio de prover informações de acessibilidades para as outras redes de ASs vizinhas denominada inter-AS ou Inter domínio. Ela também propaga informações de acessibilidade de todos roteadores internas ao AS ou seja a intra-AS ou intradomínio. E determina as todas boas para sub-redes com base na informação e política de acessibilidade. BGP oferece a cada AS meios de:

1. Obter de ASs vizinhos informações de alcançabilidade de sub-redes.

2. Propagar a informação de alcançabilidade a todos os roteadores internos ao AS

3. Determinar rotas “boas” para sub-redes com base na informação de alcançabilidade e na política do AS.

No BGP, um sistema autônomo é identificado por seu número de sistema autônomo (ASN) globalmente exclusivo.

Quando um roteador anuncia um prefixo para uma sessão BGP, inclui vários atributos BGP juntamente com o prefixo. Na terminologia do BGP, um prefixo, junto com seus atributos, é denominado uma rota. Assim, pares BGP anunciam rotas uns aos outros. Dois dos atributos mais importantes são AS-PATH e NEXT-HOP:

AS-PATH. Esse atributo contém os ASs pelos quais passou o anúncio para o prefixo. Quando um prefixo é passado para dentro de um AS, este adiciona seu ASN ao atributo AS-PATH. O NEXT-HOP é a interface do roteador que inicia o AS-PATH.

O BGP trabalha usando sessões eBGP e iBGP para distribuir rotas dentro de ASs., sendo o eBGP que possui as melhores rotas para acesso externos e a iBGP possui melhoras rotas para um acesso interno.

# 4.7 Roteamento por difusão e para um grupo

O roteamento por difusão a camada de rede provê um serviço de entrega de pacote enviado de um nó de origem a todos os outros nós da rede. o roteamento para um grupo habilita um único nó de origem a enviar a cópia de um pacote a um subconjunto de nós das outras redes.

**Roteamentos por Difusão (Broadcast)**

No roteamento por difusão a camada de rede provê um serviço de entrega de pacote enviado de um nó de origem a todos os outros nós da rede. No modo mais direto de consegui comunicação por difusão é o nó remetente enviar uma cópia separada do pacote para cada destino. Mesmo essa abordagem sendo difusão é simples não é preciso nenhum novo protocolo de roteamento de camada de rede, nem duplicação de pacotes, nem funcionalidade de repasse, ela trás muitas desvantagens como Ineficiência - Se o nó de origem estiver conectado ao resto da rede por um único enlace, então N cópias separadas do (mesmo) pacote transitarão por esse único enlace.

A técnica mais óbvia para conseguir difusão é uma abordagem de inundação na qual o nó de origem envia uma cópia do pacote a todos os seus vizinhos. Quando um nó recebe um pacote de difusão, ele o duplica e repassa a todos os seus vizinhos (exceto ao vizinho do qual recebeu o pacote). Porém Se o grafo tiver ciclos, então uma ou mais cópias de cada pacote de difusão permanecerão em ciclo indefinidamente gerando uma tempestade de difusões.

A chave para evitar uma tempestade de difusão é um nó escolher de modo sensato quando repassa um pacote quando não repassa um pacote sendo por Inundação controlada por número de sequência, onde um nó de origem coloca seu endereço bem como um número de sequência de difusão em um pacote de difusão e então envia o pacote a todos os seus vizinhos. Outra abordagem é o repasse pelo caminho inverso que ocorre quando o roteador recebe um pacote de difusão com determinado endereço de origem, ele transmite o pacote para todos os seus enlaces de saída somente se o pacote chegou pelo enlace que está em seu próprio caminho individual mais curto de volta ao remetente. Caso contrário, o roteador apenas descarta o pacote que está entrando sem repassá-lo para nenhum de seus enlaces de saída.

De início o ideal é a utilização da inundação controlada por números de sequência e o RPF impeçam tempestades de difusão porém eles não evitam completamente a transmissão de pacotes por difusão redundantes. O ideal seria que cada nó deveria receber apenas uma cópia do pacote de difusão recebendo exatamente uma cópia. Para isso existe o spanning tree.

**Serviço para um grupo (Multicast)**

Um Multicast é quando o pacote é enviado para um grupo de usuários dentro da própria rede. Na comunicação para um grupo, enfrentamos imediatamente dois problemas — como identificar os destinatários de um pacote desse tipo e como endereçar um pacote enviado a um desses destinatários. No caso da comunicação individual, o endereço IP do destinatário (receptor) é levado em cada datagrama IP individual e identifica o único destinatário; no caso do serviço para um grupo, temos vários destinatários

Por essas razões, na arquitetura da Internet, um pacote para um grupo é endereçado usando endereço indireto, isto é, um único identificador é utilizado para o grupo de destinatários e uma cópia do pacote que é endereçada ao grupo usando esse único identificador é entregue a todos os destinatários associados ao grupo.

# 

**APÊNDICE A –** **TÍTULO DO APÊNDICE**

"Elemento opcional. O(s) apêndice(s) são identificados por letras maiúsculas consecutivas, travessão e pelos respectivos títulos. Excepcionalmente utilizam-se letras maiúsculas dobradas, na identificação, quando esgotadas as 23 letras do alfabeto" (ABNT, 2005).

**ANEXO A – TÍTULO DO ANEXO**

"Elemento opcional. O(s) anexo(s) são identificados por letras maiúsculas consecutivas, travessão e pelos respectivos títulos. Excepcionalmente utilizam-se letras maiúsculas dobradas, na identificação dos anexos, quando esgotadas as 23 letras do alfabeto" (ABNT, 2005).

**ÍNDICE**

"Elemento opcional, elaborado conforme a ABNT NBR 6034" (ABNT, 2005).