Transmissão multicast versão resumida

Sumário

1 Multicast em redes de computadores

Existem três tipos de endereços diferentes no IPv4: unicast, multicast e broadcast. No unicast, a comunicação é 1:1, ou seja, um endereço origem e um destino. No multicast, é 1:n, e no broadcast é 1:todos. O endereçamento multicast permite enviar pacotes IP para um determinado grupo de usuários que previamente se cadastraram neste grupo (identificado por ser um IP classe D, ou seja, entre 224.0.0.0 até 239.255.255.255). Para entrar, sair e se manter em grupos multicast, as estações devem utilizar o protocolo IGMP (*Internet Group Management Protocol*), a ser analisado posteriormente.

Em termos de qualidade de serviço (QoS), uma transmissão multicast é tratada da mesma forma que unicast, ou seja, possui as mesmas características de "melhor esforço" do IP unicast, sofrendo as mesmas políticas de controle de acesso e conformação de tráfego.Como a transmissão multicast possui o mesmo cabeçalho IP do unicast, os métodos de garantia de qualidade de serviço são os mesmos, e ambos podem usar diffserv, RSVP, MPLS, e assim por diante.

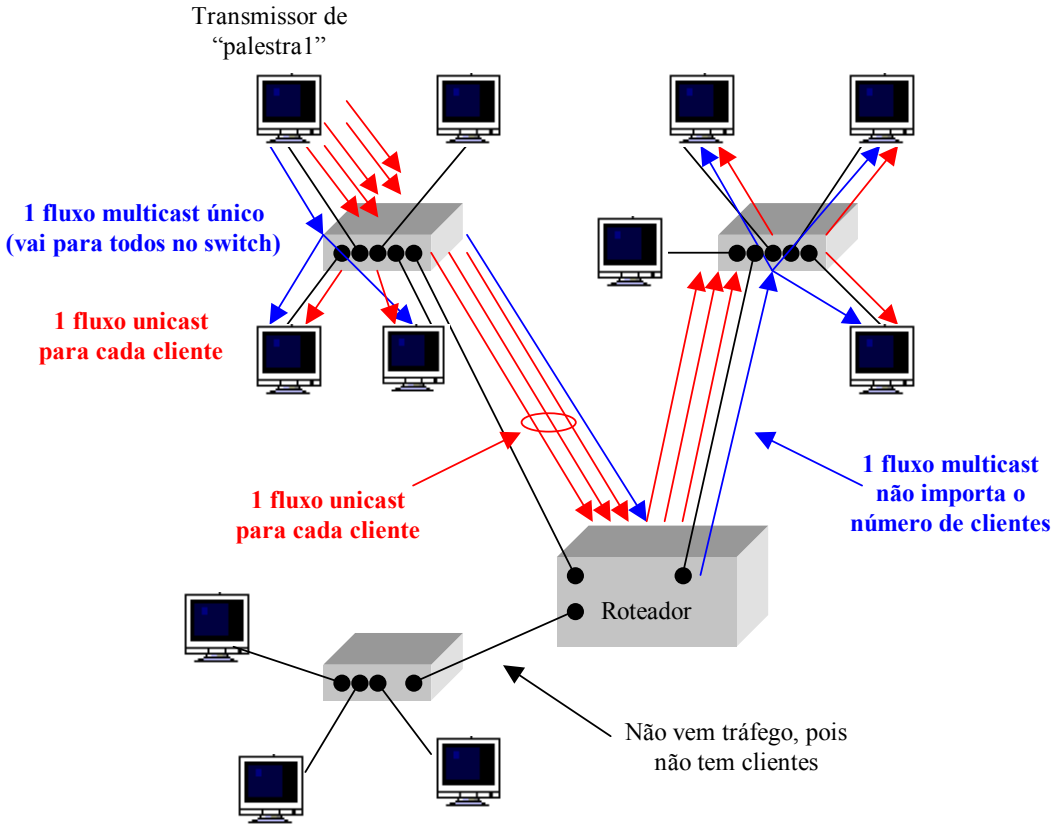
A diferença do multicast está no roteamento dos pacotes, pois vai exigir uma certa inteligência do roteador para saber por quais portas existem usuários cadastrados em grupos, fazendo com que ele replique os pacotes multicast por **todos os caminhos que possuem receptores**. Existem alguns protocolos de roteamento para fazer isso, como, por

exemplo, o DVMRP (*Distance Vector Multicast Routing Protocol*), o MOSPF (*Multicast extensions to Open Shortest Path First*) e o PIM (*Protocol-Independent Multicast*), usados para os roteadores se conversarem entre si e descobrirem por quais rotas devem ser encaminhados os pacotes. Esses protocolos serão analisados adiante.

O grande ponto a favor do multicast é que ele permite uma distribuição simultânea para um grande número de clientes, **sem exigir muito dos recursos do servidor e sem gerar muito tráfego na rede**. Assim, essa tecnologia facilita a existência de uma nova geração de aplicações "um para muitos" em rede, como **tráfego de vídeo, áudio, trabalho colaborativo, tecnologia "push", transmissão de arquivos simultaneamente para muitos usuários, e assim por diante**. As novas aplicações podem ou não exigir confiabilidade (transmissão livre de erros), entretanto, para transmissões confiáveis (como transmissão de arquivos), existe uma complexidade maior para controlar fluxo e o *feedback* dos receptores.

A figura a seguir mostra uma comparação entre multicast e unicast em um ambiente com três *switches* ligados através de um roteador que suporta multicast. No caso, a aplicação que roda no transmissor (vídeo, por exemplo) está habilitada para gerar tráfego multicast na rede, e também tráfego unicast para cada cliente que solicita. As duas situações são mostradas a seguir:

- **Tráfego multicast**: o transmissor gera **um fluxo** de pacotes IP multicast no *switch1*, que distribui para todas suas portas (inclusive a do roteador). Os clientes cadastrados no grupo multicast pegam a informação e a apresentam na tela. O roteador, por sua vez, verifica se tem alguma de suas portas com clientes cadastrados no grupo multicast. Caso tenha (caso do *switch2*), envia **um fluxo** multicast para essa porta. O *switch* respectivo o distribui em todas suas portas via IP multicast. Como não existem clientes cadastrados no *switch3*, não é gerado tráfego para essa porta.
- **Tráfego unicast**: para cada cliente, o transmissor deve gerar um fluxo de pacotes IP unicast. Assim, na figura a seguir, pode-se ver que o pobre do transmissor deve gerar seis vezes o mesmo tráfego, uma vez para cada cliente que o solicitou, e a pobre da rede deve suportar seis vezes mais largura de banda.



Os clientes que desejam receber determinado tráfego de IP multicast, como a palestra1, devem se inscrever no grupo multicast em que está sendo gerada a palestra1. Evidentemente o transmissor de "palestra1" deve transmitir neste grupo.

Um problema do multicast é para atingir pontos onde não existem roteadores multicast configurados (a maioria da Internet atual, por exemplo). Os pacotes multicast simplesmente são descartados nos roteadores que não rodam o protocolo de roteamento multicast adequado. Para conviver com esse problema enquanto a estrutura não está montada, foi criada uma estrutura de túneis, que passam multicast através de conexões unicast. Essa estrutura é conhecida como MBONE (*Multicast backbone*), e será analisada com maiores detalhes posteriormente.

1.1 Endereçamento multicast

1.1.1 Endereço multicast nível 3

O multicast utiliza a classe D de endereçamento da Internet, ou seja, de 224.0.0.0 a 239.255.255.255. A diferença no cabeçalho IP de um pacote unicast e um multicast é apenas o endereço. A classe D permite até 28 bits (268 milhões de endereços), como mostra a figura a seguir.

1	1	1	0	Grupo multicast (28 bits)

Existem alguns endereços multicast reservados pelo IANA (*Internet Assigned Numbers Authority*), como o 224.0.0.0, que é a base, e não deve ser usado. Já os endereços

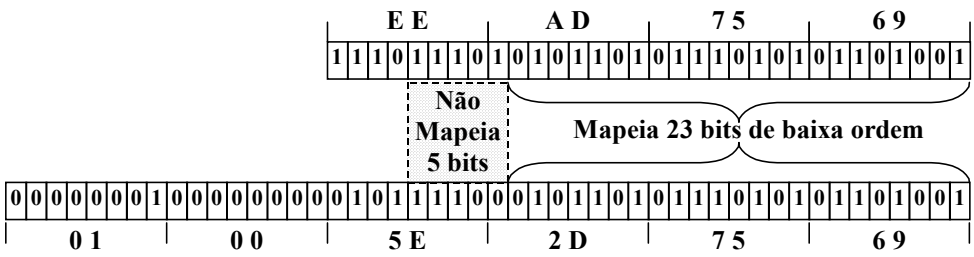
de 224.0.0.1 a 224.0.0.255 são reservados para protocolos de roteamento, como, por exemplo, "todos sistemas na subrede" (224.0.0.1), "todos roteadores nesta subrede" (224.0.0.2), "todos roteadores DVMRP" (224.0.0.4), "todos roteadores MOSPF" (224.0.0.6).

Outros são reservados para determinadas aplicações, como o "IETF1-áudio" (224.0.1.11) e o IETF1-vídeo (224.0.1.12). **O conjunto de endereços de 239.0.0.0 a 239.255.255.255 é reservado para uso local, podendo ser usado em intranets**. Maiores detalhes na RFC 1700.

1.1.2 Endereço multicast nível 2

Dentro de uma rede local, existe um conjunto de endereços MAC (*Medium Access Control*) especial destinado ao multicast. Assim, o protocolo de nível 2 e, conseqüentemente, o *host* do cliente, sabe se a mensagem é destinada a ele ou não.

Esse conjunto de números foi reservado pelo IANA, e compreende todos os endereços de 01-00-5E-00-00-00 a 01-00-5E-7F-FF-FF (somente 23 bits, comparado aos 28 do endereço multicast). Existe um processo de mapeamento entre o IP multicast e o Ethernet multicast, que é simplesmente substituir os 23 bits menos significativos do Ethernet pelos 23 bits menos significativos do IP. Assim, quando uma estação cliente se cadastra em determinado grupo multicast, como, por exemplo, o 238.173.117.105 (EE.AD.75.69), automaticamente o driver da placa de rede passa a receber mensagens MAC que cheguem no endereço 01-00-5E-2D-75-69, como mostra a figura a seguir. Já o transmissor, que criou o grupo 238.173.117.105, quando enviar a mensagem para a rede, vai enviar no endereço nível 2 01-00-5E-2D-75-69, como explicado anteriormente.



Endereço multicast (nível 2): 01.00.94.45.117.105

Observa-se que o mapeamento visto na figura anterior provoca a existência de 32 números multicast iguais para o mesmo número Ethernet (observe que os bytes "117" e "105" permaneceram iguais, enquanto que o byte "173" manteve somente 7 bits). Isso acontece, pois os 5 bits mais significativos do endereço multicast são ignorados no mapeamento. Quando se faz o mapeamento para o nível 2, **utiliza-se somente os 23 bits menos significativos**, ou seja, **de (224 a 239).0.0.0 até (224 a 239).127.255.255**. Isso faz com que, para 32 endereços, a placa de rede receba o grupo pensando que é o mesmo, e a **filtragem para saber se a máquina está cadastrada ou não se dá no nível 3** [PAU 98, pg 12].

A maioria dos *switches* Ethernet atuais, quando recebem um número MAC multicast, tratam como se fosse endereço *broadcast*, ou seja, enviam para todas suas portas. Assim, qualquer cliente cadastrado no grupo vai receber o pacote e jogar para o nível de cima, e qualquer cliente não cadastrado vai receber o pacote e ignorá-lo. Existem alguns switches que já entendem IGMP, e são úteis para minimizar o tráfego desnecessário em redes nível 2, compostas principalmente de muitos switches e poucos roteadores [CRO 00, pg 179].

1.2 IGMP (Internet Group Management Protocol)

As RFCs relativas ao IGMP são a RFC 1112 (agosto de 1989), que define o IGMPv1 no apêndice. Sua atualização é a RFC 2236 (novembro de 1997). A especificação do IGMP mais nova é a versão 3, que está em draft, porém, já existem algumas implementações disponíveis. Mais detalhes na página do IETF http://www.ietf.org/html.charters/idmr-charter.html.

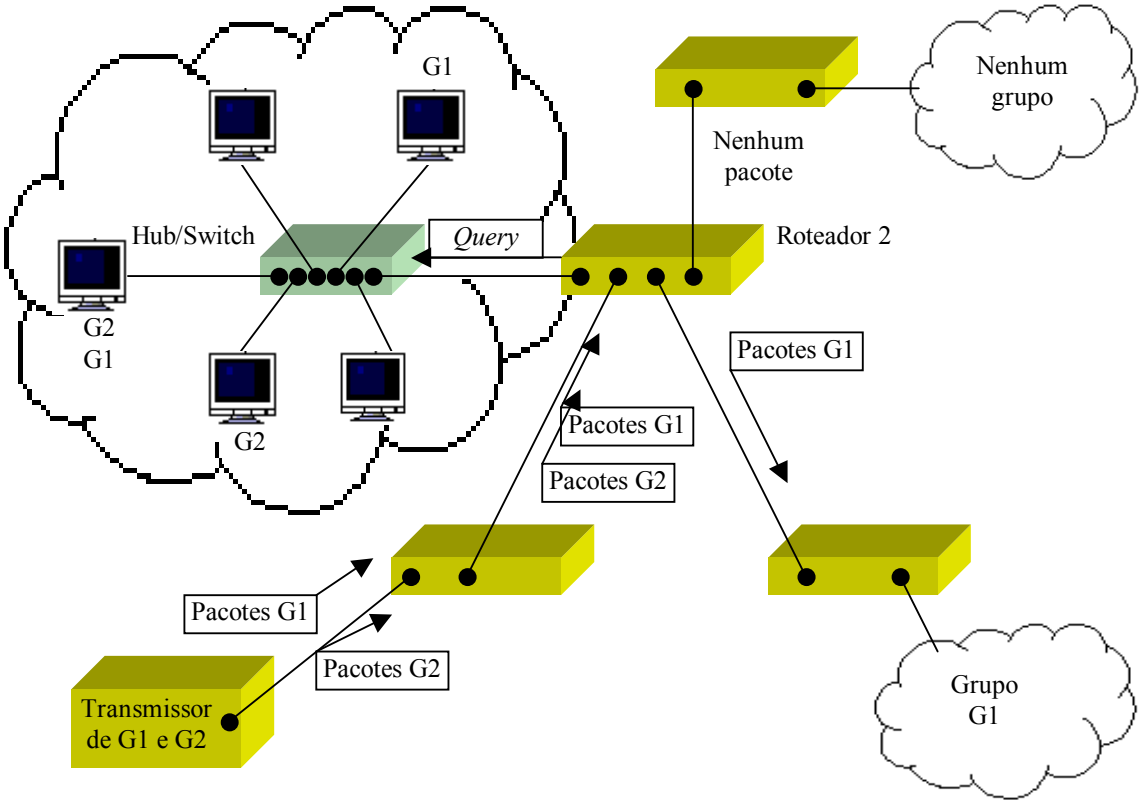
Os roteadores multicast aprendem quais grupos possuem membros em cada uma de suas interfaces através do IGMP, armazenando para cada grupo multicast se tem "presença" ou "ausência" de clientes, bem como um temporizador associado (para efetuar as *queries*).

Como pode existir mais de um roteador ligado na mesma subrede, é necessário um processo de eleição a fim de que somente um deles efetue as consultas. A definição do IGMP é simples, e o roteador eleito é o que tem o menor número IP (todo roteador inicia como "*querier*" e, caso ouça uma consulta vindo de um roteador com número IP mais baixo, se configura como "*non-querier*") [RFC 2236, pg 4].

Outra utilidade associada ao IGMP é para conversas entre roteadores, conforme será analisado adiante.

A figura a seguir resume o funcionamento do IGMP. Como pode-se ver na nuvem maior, existem estações pertencentes ao grupo G1 e ao grupo G2.

Escopo do IGMP



A *query* efetuada na subrede a partir do roteador 2 tem por objetivo descobrir TODOS os grupos que devem ser redirecionados para aquela subrede, ou seja, em quais grupos existem estações querendo receber pacotes. Assim, o roteador envia uma mensagem do tipo 0x11 (*membership query*) para o endereço 224.0.0.1 com o campo "endereço de grupo" em zero.

A figura a seguir mostra uma *query* geral do roteador (10.16.169.1) e a resposta da estação cliente (10.16.169.52) dizendo que está assistindo o grupo 238.173.117.105.

No.	Time	Source	Destination	Protocol	Info
78	78.458522	10.16.169.1	224.0.0.1	IGMP	Router query
79	79.377006	10.16.169.52	238.173.117.105	IGMP	Host response (v2)

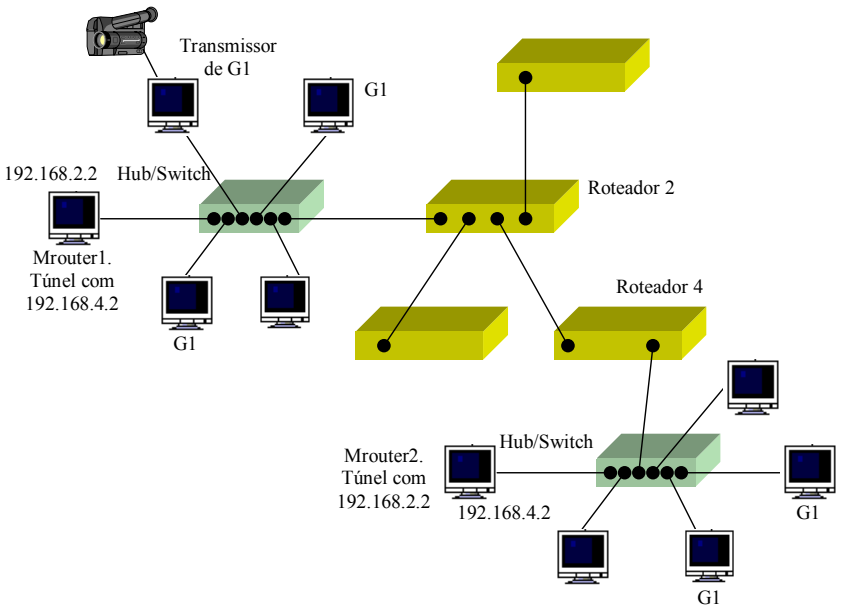
2 MBONE (Multicast Backbone)

O *backbone* multicast da Internet, conhecido como MBONE, é um conjunto de roteadores interligados a fim de distribuir tráfego multicast. Seu uso é provisório, pois devem ser eliminados quando todos os roteadores da Internet suportarem protocolos de roteamento multicast.

A figura a seguir mostra o funcionamento básico do MBONE numa forma bem simplificada para uma intranet. Como pode-se ver, o transmissor do grupo multicast G1 (que pode ser de número 238.1.1.2, por exemplo) envia pacotes multicast dentro da subrede local 192.168.2.0, ou seja, vai até o roteador 2. Para que os pacotes cheguem à subrede 192.168.4.0 deve ser feito um túnel unicast, conseguido através dos mrouters, que nada mais são do que máquinas configuradas para encapsular IP multicast dentro de IP unicast.

O mrouter2 recebe os pacotes em unicast, desencapsula o multicast, e o redistribui na sua subrede (no caso a 192.168.4.0). As máquinas dessa subrede receberão os pacotes em multicast, como se o transmissor estivesse na própria rede.

Caso os roteadores suportassem roteamento multicast, os túneis seriam inúteis, otimizando a rede, pois o roteador faria o papel do mrouter. Além disso, a manutenção dos túneis é um processo bem trabalhoso, ficando bem mais fácil o uso no roteador. Entretanto, o roteador vai ser afetado no desempenho de entrega de pacotes, pois vai ter mais uma tarefa, e isso também deve ser levado em consideração.



2.1 Configuração dos túneis multicast (mrouted)

O mrouted é uma implementação do DVMRP (*Distance-Vector Multicast Routing Protocol*), e mantém a topologia baseado num protocolo de vetor de distância semelhante ao RIP, sobre o qual implementa o esquema de RPF (*Reverse Path Forwarding*), descrito anteriormente.

Os pacotes só são encaminhados para aqueles caminhos que possuem clientes cadastrados em algum grupo multicast, evitando tráfego desnecessário. Para limitar o alcance geográfico do multicast, o campo TTL é utilizado, através de uma configuração de *threshold* a ser vista adiante.

Os túneis multicast encapsulam os pacotes em túneis IP unicast através do protocolo IP-IP. A configuração dos túneis (base no sistema FreeBSD) é feita através de um arquivo chamado "/etc/mrouted.conf".

Um túnel necessita ser configurado em ambos mrouters a fim de que possa ser usado. Um exemplo de túnel está mostrado a seguir, extraído do arquivo /etc/mrouted.conf, e explicado em seguida [FRE 01]:

```
tunnel 10.16.168.3 10.16.169.6 threshold 1 metric 1 rate_limit 0
```

- **Threshold**: define o valor mínimo de TTL (Time To Live) que um pacote multicast deve ter para que seja encaminhado por determinada interface ou túnel. Este valor é usado para controlar a abrangência do multicast. Cada roteador compara o TTL do pacote com o threshold; se o threshold for maior ou igual ao TTL, decrementa o TTL no valor de "1" e o redireciona pela interface adequada. Se o threshold for menor que o TTL, descarta o pacote. O valor default para o threshold é 1;
- **Metric**: define o custo associado a esse túnel, e pode ser usado para que o roteador decida qual interface utilizar para alcançar determinado destino (a de custo menor). Este parâmetro deve permanecer tão pequeno quanto possível, visto que o DVMRP não consegue rotear através de caminhos cuja soma de métricas seja superior a 31. A métrica default é 1;
- **Rate_limit**: esta opção permite ao administrador limitar a largura de banda (Kbit/s) do fluxo multicast que passa pelo mrouter. O default é 0 (banda ilimitada).

Em geral, todos os roteadores multicast conectados a um determinado túnel devem usar a mesma métrica e threshold para este túnel.

2.2 Limitação de escopo no TTL

O TTL (*Time to Live*) do pacote IP limita a abrangência da árvore multicast, evitando que determinadas transmissões passem da área destinada. Isso é necessário devido a razões de desempenho e também por determinadas transmissões não serem relevantes fora dum contexto. Por exemplo, uma estação de notícias do Rio Grande do Sul teria pouco interesse fora da região sul, e deveria ser filtrado para evitar que saísse dos roteadores do RS e ocupasse espaço na área de memória de roteadores do mundo inteiro. Da mesma forma, uma estação de notícias do Brasil teria pontos isolados de interesse no mundo, e deveria ser filtrado dentro do Brasil, evitando que gerasse tráfego em roteadores do exterior.

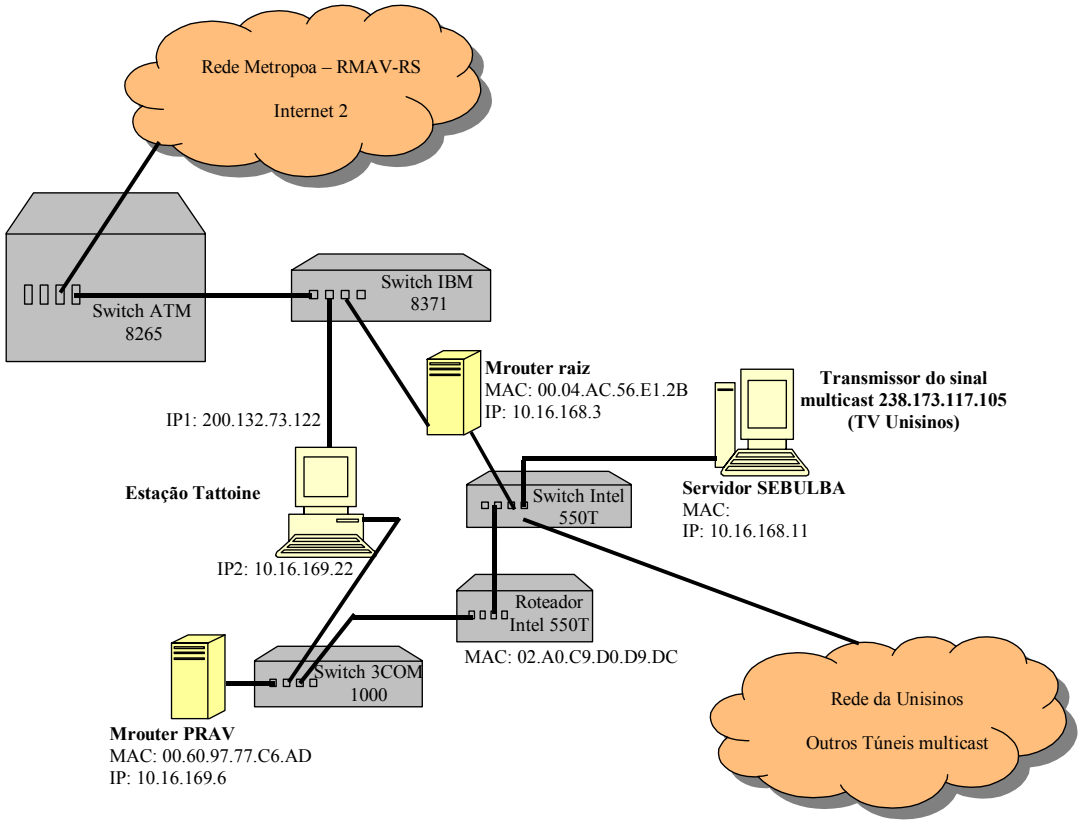
Uma forma de filtrar esses pacotes é colocar um limite de abrangência através do TTL, e é largamente utilizada. Um exemplo de configuração poderia ser o seguinte:

- 0: restrito ao mesmo *host*;
- 1: restrito à mesma subrede;
- 15: restrito ao mesmo site;
- 48: restrito à mesma região;

- 63: restrito ao mesmo país;
- 127: mundial com largura de banda limitada;
- 255: sem limites.

2.3 Estudo de caso

A figura a seguir mostra a topologia utilizada para efetuar os testes.



Existe um transmissor de sinal multicast (SEBULBA), transmitindo a TV Unisinos no IP 238.173.117.105. Este sinal é transmitido na subrede 10.16.168.x, na qual existem apenas duas estações, o transmissor e o mrouter raiz (isso para evitar enviar tráfego multicast para estações em subredes que não tenha ninguém assistindo).

O mrouter raiz recebe o sinal multicast da estação SEBULBA e o **retransmite** para a rede do Metropoa (através do switch 8371), a rede do PRAV (via túnel com o mrouter PRAV) e outras redes da Unisinos (através de túneis para o centro 3 e centro 6). Além disso, o mrouter raiz possui um túnel com a UFRGS, **recebendo** o sinal multicast do MBONE através de sua placa de rede ligada ao Metropoa. Pode-se ver seu arquivo mrouted.conf a seguir.

```
tunnel 200.132.73.102 200.132.0.66 threshold 16 metric 1 rate_limit 512
tunnel 10.16.168.3 10.16.169.6 threshold 1 metric 1 rate_limit 0
tunnel 10.16.168.3 10.13.131.101 threshold 1 metric 1 rate_limit 0
tunnel 10.16.168.3 10.16.165.243 threshold 1 metric 1 rate_limit 0
```

O arquivo mrouted.conf localizado no mrouter do PRAV deve fechar o túnel com o raiz, e seu conteúdo é mostrado a seguir.

```
tunnel 10.16.169.6 10.16.168.3 threshold 1 metric 1 rate_limit 0
```

Qualquer usuário acessando a página web da TV Unisinos ou do PRAV pode escolher o link para assistir a TV, recebendo o IP multicast que deve fazer o "join". A partir daí, recebe o sinal numa taxa de aproximadamente 700Kbps, conforme mostra a figura a seguir.

A figura a seguir é um arquivo de "sniffer" (no caso, o Ethereal), que mostra uma transmissão multicast sendo recebida pela estação "Tattoine", através de sua interface com o Metropoa. Observe que o transmissor envia um pacote com mais de 8.000 bytes, e o mesmo necessitou ser fragmentado. Observe também que, simultaneamente, estava sendo transmitida uma outra sessão multicast vinda da George Mason University, identificada pelo pacote de número 25, que vem da máquina X0C0V7 (IP 129.174.216.85) e transmitida ao IP multicast de número 224.2.197.11 (reservado para transmissões temporárias [RFC 1700]).

```
 19 0.113138 SEBULBA          238.173.117.105     UDP    Source port: 1030  Destination port: 17764
 20 0.113271 SEBULBA          238.173.117.105     IP     Fragmented IP protocol (proto=UDP 0x11, off=1480)
 21 0.113462 SEBULBA          238.173.117.105     IP     Fragmented IP protocol (proto=UDP 0x11, off=2960)
 22 0.113689 SEBULBA          238.173.117.105     IP     Fragmented IP protocol (proto=UDP 0x11, off=4440)
 23 0.113714 SEBULBA          238.173.117.105     IP     Fragmented IP protocol (proto=UDP 0x11, off=5920)
 24 0.113738 SEBULBA          238.173.117.105     IP     Fragmented IP protocol (proto=UDP 0x11, off=7400)
 25 0.156044 X0C0V7           224.2.197.11        UDP    Source port: 1031  Destination port: 26200
```

A figura a seguir detalha o pacote de número 19. Observe que o endereço Ethernet destino começa por 01-00-5E, que é o número reservado para multicast, conforme foi descrito em detalhes anteriormente. Veja que o mapeamento do IP somente pega os últimos 23 bits, pois 2D-75-69 equivale, em decimal, a 45.117.105. Veja que os últimos dois bytes são iguais ao IP. O terceiro byte não é igual pois somente porque o bit 7 não é mapeado, assim, o número 173 que, em binário equivale a 10101101 possui o bit 7 em "1". Como ele

não é mapeado, fica em “0”, resultando em “00101101”, que é igual a 45 em decimal ou 2D em hexadecimal.

```
⊟ Ethernet II
    Destination: 01:00:5e:2d:75:69 (01:00:5e:2d:75:69)
    Source: 00:04:ac:56:e1:2b (00:04:ac:56:e1:2b)
    Type: IP (0x0800)
⊟ Internet Protocol
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default)
    Total Length: 1500
    Identification: 0x6fe5
  ⊞ Flags: 0x02
    Fragment offset: 0
    Time to live: 4
    Protocol: UDP (0x11)
    Header checksum: 0x0afa (correct)
    Source: SEBULBA (10.16.168.11)
    Destination: 238.173.117.105 (238.173.117.105)
⊞ User Datagram Protocol
  Data (1472 bytes)
```

Ainda na figura acima, pode-se ver que o Ethernet origem é 00:04:AC:56:E1:2B, que é o MAC da placa de rede do mrouted raiz da topologia, pois é ele quem está enviando o sinal. Para comprovar que o número equivale ao MAC do mrouter raiz, foi efetuado o seguinte comando, cujo resultado enfatizado mostra o MAC em questão.

```
$ifconfig fxp0
fxp0: flags=8a43<UP,BROADCAST,RUNNING,ALLMULTI,SIMPLEX,MULTICAST> mtu 1500
 inet 200.132.73.102 netmask 0xffffff00 broadcast 200.132.73.255
 inet6 fe80::204:acff:fe56:e12b%fxp0 prefixlen 64 scopeid 0x2
 ether 00:04:ac:56:e1:2b
 media: autoselect (100baseTX <full-duplex>) status: active
 supported media: autoselect 100baseTX <full-duplex> 100baseTX 10baseT/UTP
<full-duplex> 10baseT/UTP
```

O mrouter mantém o IP origem como sendo a máquina transmissora (SEBULBA – 10.16.169.11), e ele somente redireciona o pacote (como um roteador deve fazer).

A figura a seguir detalha um pacote da transmissão multicast visto por uma máquina na rede do PRAV. Para alcançar essa máquina, o pacote teve que vir através de um túnel, que foi configurado entre os dois mrouters na topologia do teste (comandos vistos anteriormente).

Observe na figura a seguir que os endereços de nível 2 tem sua origem no roteador e destino no mrouter do PRAV. Já o protocolo de nível 3 é o IP encapsulado (unicast) entre os dois mrouters (origem em 10.16.168.3 e destino em 10.16.169.6). O encapsulamento é feito através de um protocolo chamado IP-IP, e pode ser visto um segundo nível de cabeçalho IP antes de chegarmos no nível de transporte.

O segundo nível IP é o que contém a informação do IP origem (transmissor do grupo multicast) e do IP destino (grupo multicast – no caso é o 238.173.117.105).

O nível 4 é UDP, pois uma transmissão multicast de multimídia normalmente possui limites de tempo, e um pacote perdido não vai impactar grandemente na hora de mostrar a

transmissão na tela. Além disso, a transmissão multicast via TCP é bem mais complicada, visto que existem "n" receptores que estariam mandando ACKs dos pacotes recebidos, e isso causaria uma implosão de ACKs. Existem certos protocolos específicos para resolver a questão do multicast confiável, e alguns podem ser vistos em [PAU 98].

```
⊟ Ethernet II
    Destination: 00:60:97:77:c6:ad (00:60:97:77:c6:ad)
    Source: 02:a0:c9:d0:d9:dc (02:a0:c9:d0:d9:dc)
    Type: IP (0x0800)
⊟ Internet Protocol
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default)
    Total Length: 1500
    Identification: 0x3a1e
  ⊞ Flags: 0x02
    Fragment offset: 0
    Time to live: 63
    Protocol: IPIP (0x04)
    Header checksum: 0xb6d6 (correct)
    Source: 10.16.168.3 (10.16.168.3)
    Destination: 10.16.169.6 (10.16.169.6)
⊟ Internet Protocol
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default)
    Total Length: 1500
    Identification: 0x0524
  ⊞ Flags: 0x02
    Fragment offset: 0
    Time to live: 4
    Protocol: UDP (0x11)
    Header checksum: 0x75bb (correct)
    Source: SEBULBA (10.16.168.11)
    Destination: 238.173.117.105 (238.173.117.105)
⊞ User Datagram Protocol
```

3 Multicast confiável

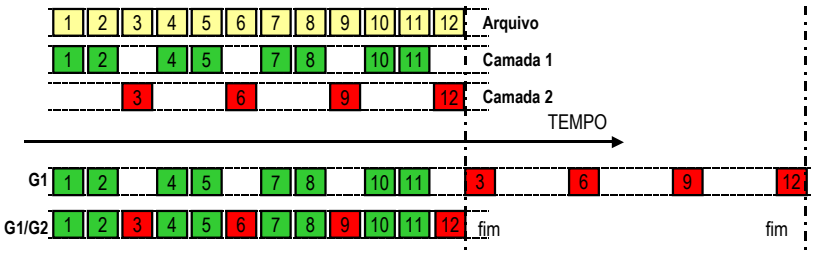
Em determinadas aplicações a confiabilidade é essencial, como transmissões de arquivo para muitos usuários, entretanto, para conseguir escalabilidade, a complexidade necessária para a gerência do fluxo e monitoração dos erros (ACKS) aumenta bastante, fazendo com que o tempo de entrega seja maior em relação ao multicast não confiável. Os seguintes aspectos devem ser abordados em protocolos de multicast confiável [BAR 00]:

- **Implosão de *feedback***: o protocolo TCP é baseado em *feedback* através de ACKs e NACKs, que indicam se o pacote chegou corretamente ao destino ou não. O problema dessa abordagem é que limita a escalabilidade na transmissão em multicast confiável (visto que podem existir milhares de receptores), impossibilitando a simples extensão do protocolo TCP/IP para multicast. Além disso, a recuperação de pacotes NACK é bastante ineficiente em multicast, pois o pacote normalmente é perdido por poucas estações.
- **Escalabilidade**: são considerados bons protocolos de multicast confiável aqueles onde o aumento no número de receptores não afeta a velocidade de transmissão dos dados. Alguns métodos para conseguir escalabilidade são o de tirar do transmissor qualquer informação de estado a respeito dos receptores e substituir o ACK dos receptores por NACKs, visto que a probabilidade de erro é menor que a de que o pacote chegue corretamente (entretanto, em certos casos, pode acontecer implosão de NACKs).
- **Controle de erro**: consiste basicamente na problemática de como detectar pacotes perdidos e, uma vez detectados, como retransmitir esses pacotes. **Para detectar pacotes perdidos no receptor**, existem pelo menos dois métodos: a) ***gap***, onde o receptor detecta falha no número de seqüência; b) ***heartbeat***: o remetente envia pacotes numa periodicidade fixa, e o receptor supõe falha quando acontece um tempo muito grande na recepção de dois pacotes. **O transmissor detecta pacotes perdidos** através de **timeout** ou **NACK** vindo do receptor. Para **retransmitir o pacote perdido**, existem duas formas: a) transmissão unicast para cada receptor que perdeu o pacote; b) transmissão multicast para o grupo. Outra forma de controle de erro é através de FEC (*Forward Error Correction*), que consiste no envio de informações redundantes que permitem ao receptor reconstruir a informação corretamente mesmo com perda de alguns pacotes, diminuindo a necessidade de *feedback*. Esse tipo de método utiliza mais largura de banda, mas ganha em latência, visto que não é necessário esperar um ou mais RTTs para recuperação de dados.
- **Controle de fluxo**: o controle de fluxo é necessário em transmissões confiáveis pois nem todos receptores possuem a mesma capacidade de recepção de dados, e se o transmissor não se adaptar adequadamente, pode causar excesso de carga em alguns receptores. Existem pelo menos três métodos de controle de fluxo: a) **baseado em janela**: os receptores limitam a janela de transmissão de acordo com sua capacidade (basicamente o mesmo método do TCP unicast); b) **baseado em taxa**: o transmissor inicia com um valor pré-determinado, e altera esse valor de acordo com feedback ou perdas; c) **baseado em camadas**: o transmissor envia o mesmo arquivo em diferentes grupos multicast, sendo que cada grupo possui uma velocidade de

transmissão diferente. O receptor assina o grupo que lhe convier. Outra forma de transmissão em camadas é detalhada no item a seguir.

3.1 Transmissão em camadas

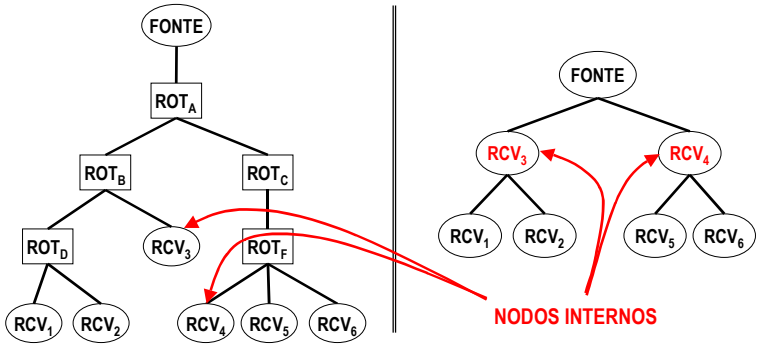
Outro método de transmissão em camadas é ilustrado na figura a seguir, onde o transmissor envia o arquivo em diferentes camadas (duas, no caso), e os receptores assinam tantas camadas quanto conseguirem para obterem a melhor taxa de recepção. Os pacotes errados (NACKs), podem ser enviados ao final da transmissão [BAR 00].



Os receptores lentos assinam somente o grupo 1, que envia numa velocidade mais lenta e demora mais para transmitir o arquivo. Já os receptores mais rápidos assinam os dois grupos, recebendo todos os pacotes de uma só vez e remontando a informação original no destino.

3.2 Transmissão hierárquica

Uma forma de aumentar a escalabilidade do sistema é organizar os receptores logicamente em forma de árvore, como mostra a figura a seguir, onde o desenho da esquerda representa a árvore de distribuição multicast (*donwstream*), e o desenho da direita representa a árvore de recuperação (*upstream*) [BAR 00].



Cada receptor tem um pai e se comunica somente com ele, limitando a abrangência do feedback, facilitando o controle de erro, de fluxo e de congestionamento. Tipicamente, cada pai é responsável pelo controle de perdas dos seus filhos, bem como a retransmissão dos dados.

Bibliografia

[FRE 01] FreeBSD. Sistema Operacional versão 4.3. **Manual do mrouted** ($man mrouted). Abr, 2001.

[FEN 01] FENNER, B. HANDLEY, M. HOLBROOK, H. KOUVELAS, I. *Protocol Independent Multicast – Sparse Mode (PIM-SM: Protocol Specification (Revised)*. **Draft-ietf-pim-sm-v2-new-02.txt**. California: IETF. Mar, 2001.

[HUI 00] HUITEMA, Christian. ***Routing in the Internet***. Second edition. Ed. Prentice Hall, New Jersey, 1999. 384 p.

[BAR 00] BARCELLOS, Marinho. ROESLER, Valter. **M&M – Multicasting & Multimídia**. In: Jornada de Atualização em Informática, JAI, XIX, 2000. Anais... Curitiba: PUC-PR, jul. 2000.

[CRO 00] CROLL, Alistair; PACKMAN, Eric. ***Managing Bandwidth – Deploying QoS in Enterprise Networks***. New Jersey: Prentice Hall. 2000.

[PAU 98] PAUL, Sanjoy. ***Multicasting on the Internet and its applications***. Kluwer Academic Publishers: Massachussets. 1998. 421p.

[RFC 1075] WAITZMAN, D. PARTRIDGE, C. DEERING, S. *Distance Vector Multicast Routing Protocol*. **RFC 1075**. California: IETF. Nov, 1988.

[RFC 1584] MOY, J. ***Multicast Extensions to OSPF***. RFC 1584 (Standards Track). California: IETF. 1994.

[RFC 1700] REYNOLDS, J. POSTEL, J. ***Assigned Numbers***. RFC 1700 (Standards Track). California: IETF. 1994.

[RFC 2236] FENNER, W. *Internet Group Management Protocol, Version 2*. **RFC 2236 (*Standards Track*)**. California: IETF. 1997.

[SEM 97] SEMERIA, Chuck. MAUFER, Tom. ***Introduction to IP multicast routing***. 1997. Em http://www.3com.com/nsc/501303.html.

[TAN 97] TANENBAUM, Andrew C. **Redes de Computadores** - 3a edição. Ed. Campus, Rio de Janeiro, 1997.