

IMPLEMENTAÇÃO E APLICAÇÕES DE UM BACKBONE IP MULTICASTING

Alunos

André Luiz Bacellar de Miranda 99/16986
Yamar Aires da Silva 98/20312

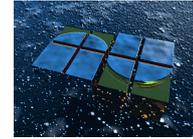
Orientadores

Ricardo Staciarinni Puttini
Rafael Timóteo de Sousa Junior

- 1 Semestre, 2000 -

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Agradecimentos

Agradeço à minha família que me deu o suporte e as condições para a produção e conclusão deste projeto, ao Enio e o Eduardo que me salvaram nas horas de dúvida, ao Álvaro e o Tuti que foram companheiros de madrugada, ao Yamar que foi meu companheiro de luta e aos meus orientadores, Ricardo e Rafael que deram o empurrão final indispensável para a conclusão deste projeto.

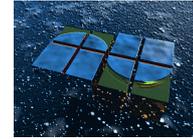
André Luiz Bacellar de Miranda

Agradeço aos meus pais e a toda minha família que sempre acreditou e me apoiou nessa jornada para a minha formatura. Agradeço aos meus orientadores Ricardo e Rafael pela disponibilização de recursos de infraestrutura e de conhecimento fundamentais para a realização e conclusão deste trabalho. Aos bolsistas e funcionários do NTI, em especial Enio e Eduardo, que foram fundamentais pela boa vontade e conhecimento de ferramentas essenciais para o desenvolvimento do projeto. Aos companheiros Álvaro e Tuti pela solidariedade nas horas difíceis e ao André, parceiro neste projeto, cuja participação foi fundamental para a qualidade do projeto e pelo cumprimento dos prazos, valeu André. E, por fim, gostaria de agradecer a Deus por me permitir concluir minha jornada até hoje. Obrigado Senhor!

Yamar Aires da Silva

Backbone multicast

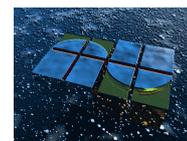
Universidade de Brasília – Engenharia Elétrica



Resumo

Este projeto tem como objetivo a montagem de um *backbone IP multicasting* no laboratório do curso de Redes de Comunicação da UNB. Nele estão contidos os dados e resultados relacionados a sua construção. Em um primeiro momento, a técnica de *multicasting* é descrita teoricamente ressaltando-se os seus processos e características. Depois, são descritos os procedimentos, problemas ocorridos e soluções encontradas durante as etapas de projeto e implantação. Após a descrição da montagem, são apresentadas experiências ministradas durante a mesma e descritas algumas aplicações utilizadas.

O resultado deste projeto foi a montagem do backbone proposto além de um estudo comparativo entre unicast e multicast e um documento guia de instalação e configuração de um roteador multicast sobre um computador com o sistema operacional Linux.

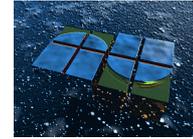


Índice Analítico

Resumo	2
1 - Introdução	8
2 - Fundamentos da tecnologia de multicasting:	13
2.1 – Conceitos Básicos	13
2.2 - Interfaces Multicast	14
2.3 - IP Multicasting	16
2.4 - Mapeamento de Endereços IP Em Endereços MAC	18
2.5 - Datagramas IP Multicast	19
2.6 - Níveis de Conformância Para Multicasting	20
2.7 - IGMP (Internet Group Management Protocol)	22
2.8 - Algoritmos de Roteamento	26
2.8.1 - Flooding	26
2.8.2 - Spanning tree	28
2.8.3 – RPF	29
2.8.4 - TRPF	30
2.9 -Protocolos de Roteamento Multicast	35
2.9.1 - DVMRP (Distance Vector Multicast Routing Protocol)	35
2.9.2 - MOSPF (Multicast <i>Open Shortest Path First</i>)	38
2.9.3 -PIM (Protocol-Independent multicast)	40
2.10 - MBONE (Multicast <i>Backbone</i>)	43
3 – Implementação Prática	44
3.1 – Experiência 1 - Ethernet	44
3.2 – Experiência 2 - Roteador	46
3.3 - Experiência 3 - DVMRP	49
3.4 - Experiência 4 – Backbone multicast	50
3.5 – Experiência 5 - Estudo de Roteamento e Tráfego	53
3.5.1 - Confirmação de transmissão em multicast	53
3.5.2 - Estudo de Roteamento	56
3.5.3 - Estudo de tráfego.	58
3.5.4 - Economia em termos de processamento no servidor	59

Backbone multicast

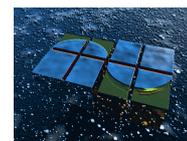
Universidade de Brasília – Engenharia Elétrica



4 - Aplicações	62
4.1 Sniffer Pro	62
4.2 Real Networks	63
Real Server	63
Real Player	65
Real Producer	66
4.3 Mrouted	67
4.4 Mbone Tools	69
Sdr – Multicast Session Directory	70
4.5 Live Networks	71
Live Caster	71
Live Gate	71
5 – Conclusão	72

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Índice de Figuras

Figura 1: unicast em rede ethernet	8
Figura 2: multicast em rede ethernet	9
Figura 3:– unicast em rede IP	10
Figura 4: multicast em rede IP	11
Figura 5 : Classes de endereços IP	16
Figura 6 : Formato de mensagens IGMP	24
Figura 7 : Início do algoritmo <i>Flooding</i>	27
Figura 8 : Algoritmo <i>Flooding</i> - passo intermediário	27
Figura 9 : Algoritmo <i>Flooding</i> - passo intermediário	28
Figura 10 : Algoritmo <i>Flooding</i> - passo final	28
Figura 11 : Exemplo de algoritmo do tipo <i>Spanning Trees</i>	29
Figura 12 : Árvores RPF	30
Figura 13 : Exemplo de início mensagens <i>Prune</i> no protocolo TRPF	32
Figura 14 : Propagação de mensagem <i>Prune</i>	32
Figura 15 : Ramos podados por mensagens <i>Prune</i>	32
Figura 16 : Resultado da situação dos roteadores depois do processo de <i>prunning</i>	33
Figura 17 : Árvore baseada em CBT	34
Figura 18: Classes e subdivisões de protocolos de roteamento multicast	35
Figura 19 : Início de mensagem de adesão a grupos em redes podadas	36
Figura 20 : Mensagem <i>Graft</i> reportando inclusão em grupos multicast	37
Figura 21 : Propagação de mensagem <i>graft</i> reinserindo roteadores antes podados	37
Figura 22 : Última mensagem de liberação de rotas multicast	37
Figura 23 : <i>Host</i> reinserido nas rotas multicast.	38
Figura 24 : Mensagem <i>Join</i> do protocolo PIM-SM	41
Figura 25 : <i>Rendezvous point</i> assume o roteamento multicast	41
Figura 26 : Início do tráfego de datagramas multicast	41
Figura 27 : Tráfego de dados utilizando o protocolo PIM-SM	42
Figura 28 : Mensagens <i>Prune</i> em roteadores PIM-SM	42
Figura 29 : Tráfego de dados utilizando o protocolo PIM-SM	42
Figura 30 : Barramento Ethernet	45
Figura 31 : Roteador multicast (Linux)	46

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

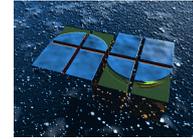


Figura 32 : Roteadores multicast _____	49
Figura 33 – configuração Lógica_____	50
Figura 34 – configuração Física_____	51
Figura 35 : Transmissão de dados em unicast _____	54
Figura 36 : Transmissão em IP multicast_____	55
Figura 37 : Transmissão de dados em Ethernet multicast_____	56
Figura 38 : Tráfego em unicast _____	58
Figura 39 : Tráfego em multicast _____	59
Figura 40 : Processamento de Servidor para 6 usuários em multicast _____	59
Figura 41 : Processamento do servidor para 1 usuário em unicast _____	60
Figura 42 : Processamento de servidor para 6 usuários em multicast _____	60
Figura 43 : Exemplo de imagem transmitida em multicast _____	61
Figura 44 : Informações conseguidas a respeito da transmissão no Real Áudio Player _	62
Figura 45 : Interface do Real Server – Configuração multicast escalável _____	63
Figura 46 : Interface do Real Server – Configuração multicast escalável (continuação)_____	64
Figura 47 : Interface do Real Player G2 _____	65
Figura 48 : Interface do Real Producer _____	66
Figura 49 : Encapsulamento de datagramas IP multicast em datagramas IP_____	68
Figura 50 : Figura do SDR_____	70

Índice de Tabelas

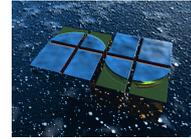
Tabela 1 : Relação valor de TTL e alcance do datagrama_____	20
Tabela 2 : Tabela de Roteamento - rotas estáticas_____	52
Tabela 3 : Ferramentas do Mbone _____	69

Anexos

- Anexo 1 : Configuração de roteadores multicast em ambiente Linux
- Anexo 2 : Mini How-to mrouterd
- Anexo 3 : E-Mail

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



LISTA DE ACRÔNIMOS

IGMP – Internet Group Management Protocol

ICMP – Internet Control Management Protocol

DVMRP – Distance Vector multicast Routing Protocol

PIM – Protocol-Independent multicast

MOSPF - multicast Open Shortest Path First

IP – Internet Protocol

TCP – Transmission Control Protocol

RIP – Routing Internet Protocol

OSPF - Open Shortest Path First

ATM – Asynchronous Transfer Mode

RPF – Reverse Path Forward

IANA - Internet Assigned Number Authority

UDP – User Datagram Protocol

SDR – Session Directory Protocol

RTSP – Real Time Streaming Protocol

TTL – Time to Live

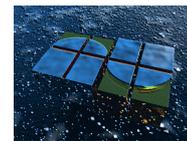
TRPB – Truncated Reverse Path *broadcasting*

MBone - *Backbone* de multicast

RPM – Reverse Path *multicasting*

RPM (LINUX) – Formato de Arquivo De Instalação de Programas

CBT – Core - Based Trees



1 - Introdução

O objetivo deste projeto é a implementação de um backbone de IP multicasting.

O multicast é um modo de transmissão em que os dados são transmitidos uma única vez e recebidos por um número determinado grupo de hosts. Ele difere do *unicast*, que é a transmissão de um computador para outro e do *broadcast*, que é a transmissão de um computador para todos de uma determinada rede.

A motivação do estudo da tecnologia de multicast se deve ao fato de este tipo de transmissão estar se tornando cada vez mais frequente. Ela se faz necessária em aplicações de vídeo conferência, atualizações de software e muitas outras. Grande parte destas transmissões são atualmente feitas em unicast, o que acarreta uma grande geração de tráfego e processamento no servidor devido ao fato de ser feita uma conexão exclusiva para cada um dos participantes.

A rede ethernet é muito apropriada para a transmissão em multicast porque ela parte do princípio que o meio de transmissão é compartilhado, ou seja, todas as transmissões são teoricamente recebidas por todas as máquinas.

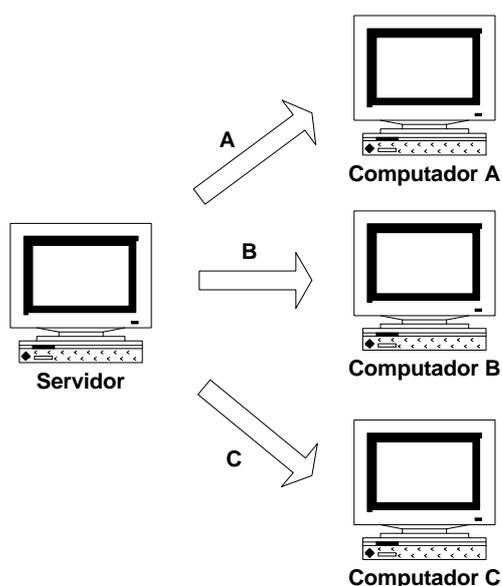
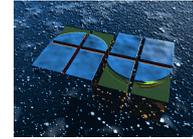


Figura 1: unicast em rede ethernet

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Como visto na figura 1, devido as características da transmissão em unicast, o servidor se torna obrigado a transmitir uma mensagem para cada um dos computadores.

Usando o multicast, ele só tem de transmitir a mensagem uma vez que todos os computadores que estiverem no grupo vão receber a informação, isto está demonstrado na figura 2. Para receber a mensagens as máquinas tem de criar uma interface virtual que possibilita que ele tenha um endereço por um tempo determinado e compartilhado por outras máquinas.

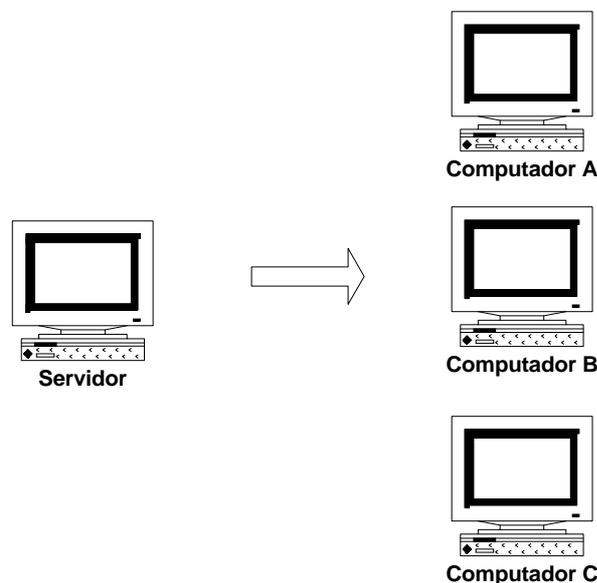
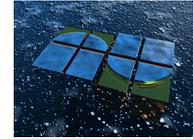


Figura 2: multicast em rede ethernet

O ATM (Asynchronous transfer mode) é uma tecnologia baseada em circuitos virtuais, nas transmissões multicast se faz necessária a criação de um circuito para cada um dos hosts. Também é possível se aproveitar as técnicas de roteamento multicast desenvolvidas para redes IP(Internet Protocol) e utilizá-las em ATM. Para isso é preciso simular o o endereçamento IP. Existem várias formas de se realizar isto sendo uma delas o IP Classico.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Nas redes IP, pelo fato de o meio não ser compartilhado e sim composto por várias redes ligadas através de roteadores, se faz necessário que eles sejam capazes de entender e transmitir estas mensagens com endereços multicast, endereços estes que são diferentes dos das redes que ele está interligado.

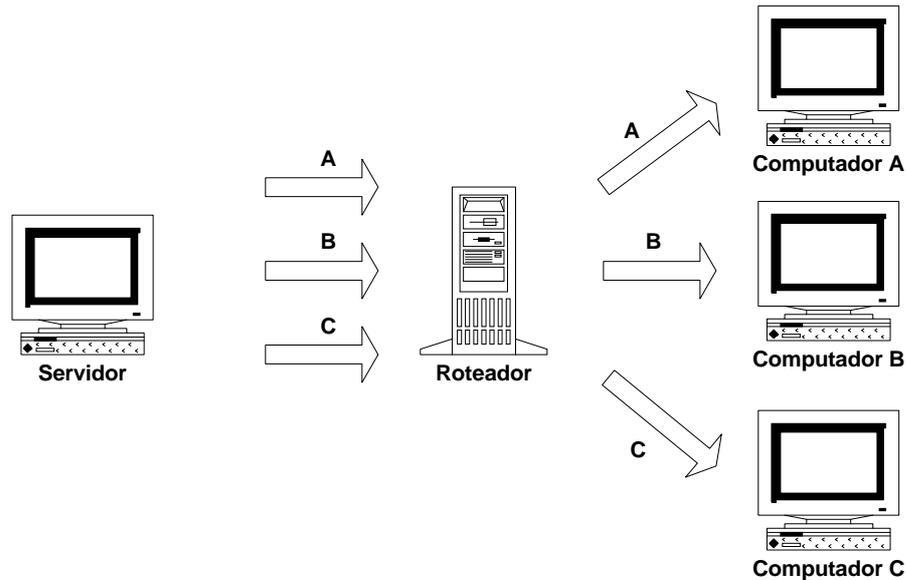


Figura 3: – unicast em rede IP

Pode-se ver na figura 3 como ocorre uma transmissão unicast em um ambiente IP.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

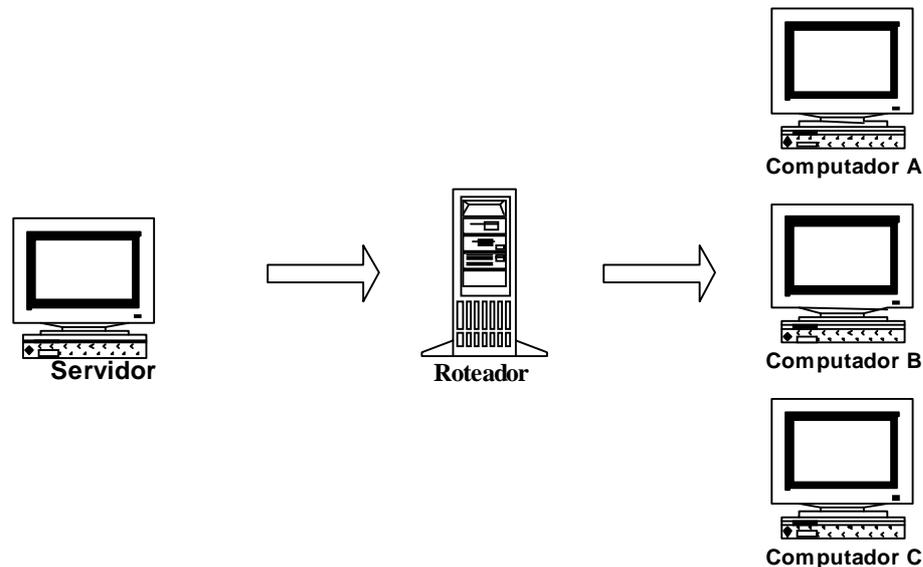
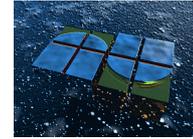


Figura 4: multicast em rede IP

Na figura 4 é mostrado como ocorre uma transmissão multicast em uma rede IP. No caso acima mostrado só aparece uma rede ligada ao roteador, mas caso houvessem outras redes, este roteador teria de transmitir as mensagens multicast para cada uma das redes. Isto ocorre porque elas não estão compartilhando o mesmo meio como nas redes ethernet. Muito mais que isso, ele deve ser capaz de saber se uma rede está ou não interessada em receber este tipo de dado porque, senão, ele estaria criando tráfego desnecessário em uma rede que não havia requisitado tal informação.

A montagem do backbone experimental de IP *multicasting* se inicia com um estudo teórico do multicast em diversos tipos de rede e dos métodos de roteamento deste tipo de tráfego em redes IP.

E sua montagem constitui de pequenos experimentos que compunham os passos necessários para a montagem do *backbone*.

Experimento 1 – Ethernet

- Execução e comprovação de uma transmissão multicast em uma rede ethernet

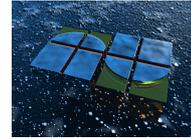
Experimento 2 – Roteador

- Instalação e configuração de uma máquina Linux para funcionar com um roteador de tráfego multicast e comprovação experimental do roteamento.

Experimento 3 – DVMRP

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



- Comprovação do funcionamento do mprotocolo DVMRP no daemon mrouted.

Experimento 4 - Backbone Multicast

- Montagem de um backbone e estudo do funcionamento de aplicações multicast.

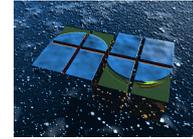
Após os experimentos executaram-se estudos de comparação entre o *unicast* e o *multicast* em termos de:

- tráfego
- processamento do servidor
- estudo de rotas de tráfego

Por fim analisa-se as possibilidades e as perspectivas dos multicast nas redes atuais e nas redes do futuro.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



2 - Fundamentos da tecnologia de multicasting:

2.1 – Conceitos Básicos

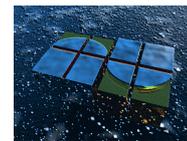
Alguns conceitos básicos são essenciais para a compreensão da distribuição multiponto de datagramas. Uma análise do ponto de vista de tecnologias e arquiteturas de redes é importante para entender o que ocorre ao ser implementada tal distribuição. Também se faz necessária a compreensão, do ponto de vista da abstração em Internet, da transmissão de um único datagrama IP para um conjunto de *hosts* que formam um único grupo multicast. Tal abstração é conhecida como IP *multicasting*. Outros conceitos mais avançados também serão abordados para completar a compreensão de *multicasting*.

Muitas tecnologias de redes possuem mecanismos para enviar, simultaneamente ou quase, pacotes para múltiplos destinos. A forma mais comum de distribuição multiponto é conhecida como *broadcasting*. O mecanismo utilizado para tal distribuição faz com que a rede entregue uma cópia do datagrama para cada destino da rede. Redes com tecnologia podem cumprir tal tarefa com a transmissão de um único frame em seu barramento. Já em redes com tecnologias orientadas a conexão é necessário enviar cópias do pacote através de todas as conexões individuais.

O que torna isso possível é que a maioria das interfaces reconhecem endereços de destino especialmente reservados para tal distribuição. No caso de tecnologia *ethernet*, cujos endereços físicos são formados por 48 bits de identificação, o endereçamento com todos os bits iguais a 1 denota *broadcasting*. Sendo assim, as interfaces reconhecem tanto seu próprio endereço MAC (*Media Access Control*) como o endereço de *broadcasting* e aceita pacotes com ambos os endereços de destino.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



2.2 - Interfaces Multicast

Diferentemente de broadcast, o multicast permite que cada máquina seja configurada para participar de um grupo multicast. Estes grupos são identificados por endereços reservados para este fim.

Quando um grupo de estações deseja se comunicar, um endereço multicast é escolhido. Após a configuração de todas as interfaces para reconhecer determinado endereço de *multicasting*, todas as estações no grupo receberão uma cópia de cada pacote enviado para o endereço de *multicasting*.

Em tecnologia *Ethernet* usa-se o bit menos significativo do octeto mais significativo do endereço MAC, igual a 1, para distinguir endereços multicast de endereços convencionais *unicast*, onde o mesmo bit possui valor 0. Em notação hexadecimal temos:

x0.xx.xx.xx.xx.xx	endereços MAC <i>unicast</i>
11.11.11.11.11.11	endereços MAC <i>broadcast</i>
01.xx.xx.xx.xx.xx	endereços MAC multicast

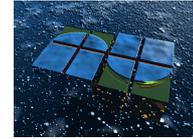
Porém, a IANA (*Internet Assigned Number Authority*) reservou uma gama de endereços MAC para designar multicast. Estes endereços começam em 01-00-5e-00-00-00 e vão até 01-00-5e-ff-ff-ff . O prefixo 01-00-5e identifica o pacote como um datagrama multicast.

A interface é, inicialmente, configurada para aceitar pacotes para o endereço de *broadcast ethernet* e para seu próprio endereço MAC. Entretanto, a interface pode ser reconfigurada para aceitar pacotes destinados à endereços multicast. As interfaces capazes de serem reconfiguradas dependem de um nível de conformância e do *software* de controle da interface. Tais assuntos serão abordados nas seções seguintes.

Atualmente, a maioria das interfaces é habilitada a reconhecer endereços multicast. Porém, apesar de habilitadas, as interfaces devem ser instruídas de quais endereços multicast, relativos aos grupos multicast de interesse do usuário, devem ser compreendidos como endereço de destino referidos à esta interface. Endereços multicast são divulgados

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



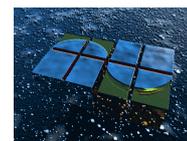
através de endereços IP classe D. A relação entre endereços IP e endereços MAC é feita através do mapeamento de endereços IP em endereços MAC. A abstração em Internet da tecnologia multicast é abordada na seção seguinte.

É importante ressaltar que apesar das interfaces habilitadas para *multicast* distinguirem endereços MAC de destino como *unicast* ou multicast, não cabe à interface decidir se aceita ou não tais pacotes. Tal tarefa é desempenhada nas camadas de enlace e inter-rede referentes ao modelo OSI (*Open Systems Interconnection*). O protocolo IP é o responsável pela decisão final referente à captura ou descarte do datagrama. Os detalhes deste processo serão melhor descritos a seguir.

Endereçamento em multicast pode ser visto como uma generalização de todas as outras formas de endereçamento. Pode-se pensar o convencional *unicast*, que se utiliza de mecanismos como o ARP (*Address Resolution Protocol*) para resolver endereços MAC de destino, como uma única máquina em um único grupo multicast, onde os pacotes são endereçados somente à esta máquina, porém com endereço de destino de multicast, o qual foi escolhido pelo usuário e informado ao *kernel*, seu interesse em receber pacotes com tais endereços de destino. Endereçamento em *broadcast* também pode ser visto como uma forma de multicast no qual todas as estações são membros de um único grupo multicast.

A notória diversidade de tecnologias e arquiteturas de redes na Internet demanda a introdução do conceito de grupos multicast. Um grupo multicast é a abstração em Internet da comunicação efetiva de vários *hosts* separados por redes heterogêneas, com a transmissão de um único pacote com um endereço IP de destino classe D e a recepção deste pacote por todos os *hosts* interessados neste conteúdo. O conjunto destes *hosts* comunicando-se através de um único endereço IP de destino com a transmissão de um único pacote para todo o grupo constitui um grupo multicast.

Backbone multicast



2.3 - IP Multicasting

Para estações membros de grupos comuns de multicast separadas por diferentes redes físicas, a abstração dos conceitos de *hardware* em tecnologias IP se fazem necessárias. Esta abstração é conhecida como IP multicast. Permite a transmissão de datagramas IP para todas as estações de um grupo multicast, usando o mesmo princípio de melhor esforço do IP como qualquer outra comutação de pacotes onde os datagramas podem ser perdidos, duplicados ou atrasados e descartados.

A participação em grupos é feita de forma dinâmica, uma estação pode se juntar a um grupo de multicast ou sair, sempre que desejar. Ainda mais, pode participar de vários grupos diferentes na mesma interface. A participação indica que determinada interface é capaz de receber datagramas destinados a um grupo particular de multicast. Uma estação pode inclusive enviar datagramas para um grupo mesmo sem ser membro deste grupo.

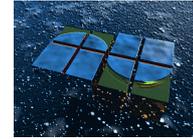
O endereçamento para IP multicast é feito utilizando-se endereços IP classe D. Esta classe de endereços IP é exclusivamente reservada para *multicasting* como pode-se observar na figura 5:

0	31	Gama de endereços
0		Endereços classe A 0.0.0.0 – 127.255.255.255
1 0		Endereços classe B 128.0.0.0 – 191.255.255.255
1 1 0		Endereços classe C 192.0.0.0 – 223.255.255.255
1 1 1 0		Endereços classe D 224.0.0.0 – 239.255.255.255
1 1 1 1 0		Endereços classe E 240.0.0.0 – 247.255.255.255

Figura 5 : Classes de endereços IP

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Todo datagrama IP cujo endereço de destino inicia com os bits “1110” é um datagrama IP multicast. Os 28 bits restantes são utilizados para designar um grupo multicast.

É importante observar que endereços IP classe D são utilizados sempre como endereços de destino e jamais são utilizados para designar endereços de origem em datagramas IP. Quando um grupo multicast é iniciado, os hosts enviam datagramas endereçados ao grupo, enquanto os hosts interessadas instruem as interfaces, através do *Kernel*, para que recebam os datagramas destinados ao grupo de interesse.

Alguns destes endereços são reservados para designar grupos que sempre existem, independente do número de membros, tais endereços são tidos como grupos multicast notórios (“*well-known*”). São designados por razões especiais tais como:

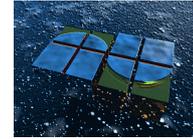
- 224.0.0.1 : Grupo de todos os hosts. Ao iniciar, todos as estações capazes de realizar *multicasting* se inscrevem nestes grupos, em todas as suas interfaces.
- 224.0.0.2 : Grupo de todos os roteadores *multicasting*. Todos os roteadores *multicasting* devem-se inscrever neste grupo em todas as interfaces.
- 224.0.0.4 : Grupo de todos os roteadores DVMRP.
- 224.0.0.5 : Grupo de todos os roteadores OSPF.
- 224.0.0.13 : Grupo de todos os roteadores PIM.

De modo geral, os endereços IP de 224.0.0.0 até 224.0.0.255 são reservados para propósitos administrativos e de manutenção, bem como de 239.0.0.0 até 239.255.255.255. Datagramas enviados com tais endereços de destino jamais são encaminhados por roteadores. Os demais endereços multicast estão disponíveis para uso temporário, são designados quando necessários e descartados quando o número de membros chega à zero. O protocolo que gerência a adesão aos grupos é o IGMP (*Internet Group Management Protocol*) e será explicado e descrito posteriormente.

IP *multicasting* pode ser implementado em uma rede local bem como através da Internet. No caso da implementação através da Internet, são necessários roteadores especiais que suportem *multicasting* para o encaminhamento de datagramas multicast.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Entretanto, as estações não necessitam de qualquer informação sobre roteadores. Uma estação transmite datagramas IP multicast usando a tecnologia da rede local e, se um roteador multicast estiver presente, este recebe o datagrama e o encaminha para as redes adequadas. Datagramas IP de multicast serão detalhados posteriormente.

O padrão TCP/IP para *multicasting* define o endereçamento de IP multicast, especifica como estações recebem e enviam datagramas IP multicast e descreve os protocolos de roteamento para alcançar a comunicação em multicast.

2.4 - Mapeamento de Endereços IP Em Endereços MAC

Para uma comunicação efetiva, em nível de enlace, endereços IP devem também ser mapeados em endereços MAC. Este mapeamento é feito pegando-se os 23 bits menos significativos do endereço IP de multicast e colocando-os nos 23 bits menos significativos do endereço especial MAC de multicast. Por exemplo, Um endereço IP de multicast como 224.0.0.4, cuja notação binária seria:

11000000.00000000.00000000.00000100

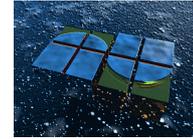
Portanto pegando-se os 23 bits menos significativos, teríamos 0000000.0000000.00000100, encapsulando nos 23 bits menos significativos do endereço MAC para *multicasting*, teremos :

01.00.5E.00.00.04 , em notação hexadecimal.

Pode-se notar que o mapeamento de um endereço IP de multicast em endereço MAC não gera uma identificação única de grupos, pois como já explicado, o endereçamento IP para multicast utiliza-se de 28 bits para a identificação de grupo, ou seja, mais de um grupo pode ser mapeado com o mesmo endereço MAC. Porém, acredita-se que o número de endereços IP de multicast disponíveis seja tão grande que a probabilidade de acontecerem 2 grupos com os mesmos 23 bits de ordem inferior seja insignificante. Caso

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



aconteça um mapeamento de endereço IP em endereço MAC idêntico para grupos diferentes, os datagramas recebidos serão descartados pelo aplicativo. Sendo assim, esta solução de utilizar tal encapsulamento facilita a implementação de *multicasting*, do ponto de vista de *debugging* e de interferência com outros protocolos que utilizem *ethernet*, sem prejuízo para a comunicação.

2.5 - Datagramas IP Multicast

Algumas considerações devem ser feitas em relação à utilização de datagramas IP para alcançar comunicações em multicast. Porém, inicialmente, deve-se notar que o tráfego em multicast é gerenciado pela camada de transporte usando-se UDP (*User Datagram Protocol*). É óbvio que seria desperdício de recursos implementar multicast através de TCP, além de ser muito bem efetuado através de UDP.

Basicamente, um aplicativo abre um *socket* UDP e preenche com um endereço IP classe D o campo de endereço de destino desejado. Entretanto, algumas outras operações são necessárias para o controle de um processo de envio. Uma operação relevante é relativa ao campo de TTL no cabeçalho do datagrama IP. Como já se sabe, tal campo controla o tempo de vida de um datagrama para evitar que este se perpetue na rede devido a erros de roteamento. O valor neste campo é decrementado ao passo que o datagrama atravessa de uma rede para outra, sendo descartado ao atingir valor zero. Para evitar que o tráfego multicast colapse a rede, existe a necessidade de limitar quão longe um datagrama multicast pode alcançar. Para tal, multicast utiliza-se do valor de TTL como um limiar para decidir se um pacote deve ser encaminhado ou não por um roteador. Estes limiares são designados para cada interface e o datagrama IP multicast é encaminhado caso seu valor de TTL seja maior que o da interface designada. Uma lista de limiares TTL's é apresentada na tabela 1 a seguir junto com o alcance desejado:

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

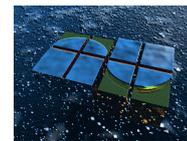


Tabela 1 : Relação valor de TTL e alcance do datagrama

Valor de TTL	Alcance
0	Restrito à própria estação. Não será transmitido pela interface
1	Restrito à mesma sub-rede. Não serão encaminhados por roteadores
<32	Restrito à mesma organização ou departamento.
<64	Restrito à mesma região
<128	Restrito ao mesmo continente.
<255	Irrestrito ou Global.

Este tipo de controle pode causar inconvenientes quando deseja-se comunicações em níveis intermediários. A solução mais recente não utiliza TTL para isso, utiliza-se dos endereços classe D reservados para tal função.

2.6 - Níveis de Conformância Para Multicasting

As estações podem apresentar-se em três diferentes níveis de conformância com as especificações de multicast:

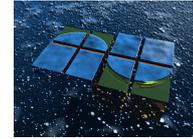
Nível 0 : Sem suporte para IP multicast. A maioria das estações e roteadores da internet ainda estão neste nível, haja visto que suporte a multicast não é obrigatório em IPv.4, apesar de ser obrigatório no IPv.6 . Basicamente, estações com este nível de conformância não são capazes de enviar nem receber pacotes multicast. Estas estações ignoraram pacotes enviados com endereço de destino de um grupo de multicast.

Nível 1 : Suporte para enviar datagramas IP multicast sem suporte para recebê-los. Poucas modificações são necessárias no software de IP para implementar este nível de conformância.

Nível 2 : Suporte pleno para IP multicast. Com este nível, As estações são capazes tanto de enviar como receber datagramas IP multicast. As estações são capazes de juntar-se

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



ou deixar grupos multicast e propagar esta informação aos roteadores. Para isso, é necessária a implementação do protocolo IGMP na pilha TCP/IP.

As modificações necessárias para que uma estação envie datagramas IP multicast são elementares. O *software* de IP deve permitir que um aplicativo especifique um endereço IP multicast como endereço de destino, e ainda, o *software* de interface de rede deve ser capaz de encapsular o endereço IP de *multicasting* em um endereço MAC de *multicasting* correspondente.

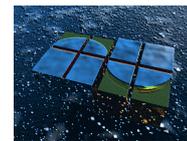
Porém, implementar uma estação que também receba datagramas IP multicast é um pouco mais complexo. É necessário que o *software* de IP detenha uma interface que permita um aplicativo declarar se deseja participar ou deixar um grupo de multicast. Caso vários aplicativos desejem participar de determinado grupo multicast, o *software* de IP deve passar uma cópia dos datagramas destinados à este grupo para cada um dos aplicativos. Caso todos os aplicativos deixem o grupo, a interface deve ser informada que não participa mais de determinado grupo. Através do protocolo IGMP, as estações atualizam sua posição em relação aos grupos passando informações aos roteadores.

A complexidade agregada à esta implementação decorre de um conceito básico em *multicasting*: Estações participam de específicos grupos de multicast em redes específicas. Ou seja, estações com múltiplas interfaces de rede podem participar de um determinado grupo multicast em uma interface e não participar em outra, por exemplo. Isto torna flexível a implementação de grupos multicast limitadas á redes específicas. Esta associação de grupos e interfaces de redes faz com que listas de endereços de grupos *multicat* sejam distintas para cada interface da estação. Além disso, um aplicativo deve especificar uma interface quando deseja juntar-se ou deixar um grupo multicast. Caso não seja especificado, o *Kernel* designa uma interface padrão.

É importante observar que, no sistema operacional Linux, o *Kernel* pode ser reconfigurado e recompilado para habilitar *multicasting* tanto em *hosts* como em roteadores. Obviamente Linux possui nível de conformância 2. O sistema operacional Windows nas versões 95/98/2000 também já vem habilitado para *multicasting* apesar de não ser possível qualquer acesso à configuração de seu *kernel*.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Portanto, podemos observar que dependendo da interface, os datagramas multicast podem ser filtrados tanto pela interface quanto pela camada IP ou por ambas, ou seja, apenas datagramas requisitados pelo aplicativo ao *Kernel* serão aceitos.

Outro aspecto relevante diz respeito a uma estação que deseja tanto transmitir quanto receber datagramas referentes ao mesmo grupo. Pelo padrão, este datagrama deve ser retornado à estação que transmitiu. Este retorno é efetuado na própria camada IP, que copia o datagrama e o retransmite ao aplicativo designado.

Essencialmente, quando deseja-se participar de um determinado grupo multicast, um aplicativo está dizendo ao *Kernel* que datagramas com determinado endereço de destino IP classe D, devem ser também endereçados ao aplicativo que o solicitou. Por exemplo, ao iniciar, todas as estações que suportam multicast se inscrevem no grupo 224.0.0.1, o grupo de todas as estações que suportam multicast.

Resumindo, ao ingressar em um grupo, é apenas instruído para a camada IP e para camada de enlace para que aceitem datagramas destinados à este grupo. É um processo de decisão da estação se determinado grupo deve ser anexado ou não.

Analogamente, pode-se pensar em um grupo multicast como uma estação de rádio, a qual pode-se sintonizar o *kernel* de um host para receber as informações referentes à este grupo. Porém, com a vantagem de ser possível receber várias estações simultaneamente.

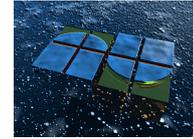
2.7 - IGMP (Internet Group Management Protocol)

Ao participar de um grupo multicast em uma rede local, uma estação deve ter um *software* que permita enviar e receber datagramas multicast, entretanto, para participar de grupos multicast além da rede local, a estação deve informar aos roteadores multicast locais seu interesse de participar em determinados grupos multicast. Estes roteadores comunicam-se com outros roteadores, encaminhando informações de adesão aos grupos e estabelecendo as rotas. Idéia muito similar ao roteamento convencional entre roteadores.

Antes, porém, de um roteador multicast poder propagar informações de adesão aos grupos, deve-se determinar se existem estações interessadas em determinados grupos. Para tal, roteadores e estações que suportam multicast devem implementar um protocolo

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



conhecido como IGMP, *Internet Group Management Protocol*, para comunicar a adesão aos grupos.

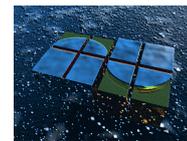
IGMP é um protocolo parecido com o ICMP, no sentido que também utiliza-se de datagramas IP para carregar informações relativas à gerência de grupos, o qual pode ser considerado parte integrante do protocolo IP. IGMP é um protocolo cujo tipo é indentificado pelo código binário referente ao número 2 no campo tipo do datagrama IP, que para a sua implementação, necessita-se que todas as máquinas envolvidas estejam configuradas para o nível 2 de conformância.

A gerência de grupos pelo IGMP é alcançada em duas etapas: A primeira ocorre quando uma estação deseja se juntar a um determinado grupo. Esta estação envia ao grupo de todas estações a informação que deseja receber datagramas destinados a determinado grupo. Os roteadores locais encaminham estas informações para outros roteadores na rede. A segunda etapa é a verificação frequente por parte dos roteadores da manutenção da participação de estações em determinados grupos. Caso após diversas verificações, nenhuma estação responder, o roteador deixa de encaminhar datagramas referentes à estes grupos e avisa aos roteadores adjacentes para que deixem também de encaminhar datagramas relativos à este grupo.

O protocolo IGMP é concebido para evitar congestionamento nas redes. Para isso, entre outras coisas, todas as comunicações entre estações e roteadores são feitas utilizando-se datagramas IP multicast. Como mensagens IGMP são encapsuladas em datagramas IP para a transmissão, o campo de endereço de destino é preenchido com o endereço IP multicast reservado para todas as estações com multicast habilitado. Sendo assim, mensagens IGMP não são processadas por estações que não são habilitadas para *multicasting*. Ainda, roteadores multicast não enviam mensagens individuais para inscrição em cada grupo, ao contrário, enviam frequentemente mensagens idênticas destinadas a todos as estações habilitadas para *multicasting* requerendo os grupos, os quais as estações estão interessadas. Estas mensagens são enviadas em uma taxa de no máximo uma mensagem por minuto. Por fim, estações membros de vários grupos simultâneos não respondem ao mesmo tempo para todos os grupos. Ao contrário, depois que uma mensagem IGMP para todas as estações habilitadas para multicast é enviada por um

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



roteador, as estações iniciam um temporizador, variando de 0 a 10 segundos aleatoriamente, para cada grupo em que desejam se inscrever, enviando as mensagens assim que o temporizador chegar ao fim. As mensagens de resposta são intercaladas por intervalos de 10 segundos. Mais de uma estação, na mesma rede, interessada no mesmo grupo, não envia resposta, caso outra estação responda, anteriormente, interessada pelo mesmo grupo.

Torna-se claro que roteadores multicast não necessitam controlar quantas ou quais estações estão interessadas em determinado grupo. Basta que controle se existe alguma estação em determinada rede interessada nas informações pertinentes à este grupo. sendo assim, basta que a primeira estação transmita uma mensagem IGMP de participação em um grupo, para que as outras estações interessadas da mesma rede também estejam inscritas.

IGMP versão 0 está especificado na RFC-998 e já está obsoleto, acredita-se que ninguém mais use esta versão. A versão 1 do IGMP está descrita na RFC-1112 e já se encontra atualizada pela RFC-2236, cujas modificações são suficientes para chamá-la de IGMP versão 2. A seguir, pretende-se dar uma descrição informal do protocolo IGMP.

As mensagens IGMP tem o seguinte formato observado na figura 6:

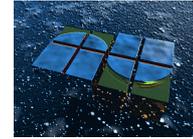
0	4	8	16	31
Versão	Tipo	Máx. tempo de resposta	Checksum	
Endereço de grupo multicast				

Figura 6 : Formato de mensagens IGMP

Aqui serão tratadas as diferenças entre as versões 1 e 2 do IGMP, referidas agora como IGMPv1 e IGMPv2 respectivamente, haja visto que a versão 0 não é mais usada. Utilizando-se IGMPv1, o campo de Máximo tempo de resposta não é utilizado, zerando-se todos os bits relativos a este campo ao enviar a mensagem e ignorando este campo na recepção da mensagem. Entretanto com IGMPv2, este campo leva o valor de tempo máximo para que um host responda à inscrição em grupos, variando de zero a dez como mencionado anteriormente. Os campos Versão e Tipo são formados por 4 bits cada, onde

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



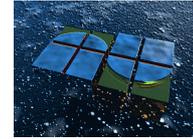
versões 1 e 2 são representados em binário por 1 e 2 respectivamente. Em IGMPv1, o campo Tipo pode denotar 2 mensagens diferentes, Tipo 1, que denota a mensagem enviada pelo roteador requerendo inscrições em grupos, e Tipo 2, que denota mensagens de inscrição em grupos enviadas pelos *hosts*. Quanto ao IGMPv2, outros dois tipos são adicionados, tipo 6, que denota uma mensagem IGMPv2 enviada por um *host* caso este detecte a presença de um roteador com IGMPv2 também presente, e tipo 7, a qual representa um importante incremento na versão 2, pois denota uma mensagem para deixar determinado grupo enviada pelos *hosts*. Esta modificação é importante pois economiza largura de banda pois, ao passo que o último *host* deixa determinado grupo, o roteador deixa, quase que de imediato, de encaminhar datagramas referentes à este grupo.

Quando uma mensagem tipo 2 é enviada, o campo de endereço de grupo é preenchido com o endereço classe D respectivo, porém quando se trata de uma mensagem tipo 1 este campo é zero e ignorado pelos *hosts*.

Periodicamente os roteadores enviam uma mensagem IGMP para inscrição em grupos endereçada ao grupo de todos os *hosts* que suportam multicast (224.0.0.1) com o valor de TTL igual a 1. Todos os *hosts* com multicast habilitado recebem esta mensagem. Caso estejam interessados em alguns grupos, os *hosts*, após o fim do temporizador, respondem com uma mensagem IGMP, com TTL igual a 1, informando o interesse por datagramas enviados ao respectivo endereço de grupo. O roteador por sua vez informa aos outros roteadores o interesse por datagramas referentes àqueles solicitados pelos *hosts*. O protocolo IGMP não trata da propagação de informações relativas a grupos entre roteadores, para tal, utiliza-se um protocolo, o DVMRP (*Distance Vector multicasting Routing Protocol*), ainda experimental, que será discutido a seguir.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



2.8 - Algoritmos de Roteamento

Vários algoritmos têm sido propostos para formar árvores de roteamento multicast. O roteamento baseado em árvores de rota é implementado com o objetivo de se determinar uma rota única e ótima para alcançar determinadas redes. Utilizando-se o algoritmo de Dijkstra, que se baseia nas informações de estado de *link*, ou seja as métricas em termos de saltos para cada roteador que atravessa e as respectivas situações de tráfego de cada link, pode-se construir árvores de rotas.

Estes algoritmos podem ser utilizados para implementar protocolos de roteamento multicast. Inicialmente serão tratados dois algoritmos básicos chamados *Flooding* e *spanning tree*. Serão discutidos também algoritmos mais sofisticados a exemplo do RPF (*Reverse Path Forwarding*), TRPF (*Truncated Reverse Path Forwarding*) e CBT (*Core Based Trees*). A intenção é mostrar como alguns desses algoritmos podem ser implementados para desenvolver protocolos de roteamento multicast.

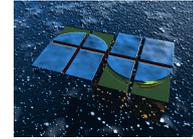
2.8.1 - Flooding

Um algoritmo do tipo *Flooding* é a técnica mais simples de encaminhar datagramas multicast entre roteadores na Internet. Com este algoritmo, quando um roteador recebe um datagrama multicast é feita uma verificação se este datagrama já foi encaminhado antes. Caso seja a primeira vez que o datagrama chega ao roteador, ele é encaminhado para todas as interfaces exceto àquela por onde veio tal datagrama. Caso contrário, o roteador simplesmente descarta o datagrama. Desse modo é garantido que todos os roteadores de uma rede recebam uma cópia do datagrama.

Apesar de ser um algoritmo simples, ele acarreta algumas desvantagens. Um algoritmo *flooding* produz um grande número de datagramas duplicados causando um grande desperdício de banda. Além disso, a verificação de todo pacote recém chegado produz grande demanda de recursos de memória e de processamento, tornando a tarefa de roteamento ineficiente.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



As figuras de 7 a 10 a seguir ilustram a implementação de um algoritmo do tipo *Flooding*:

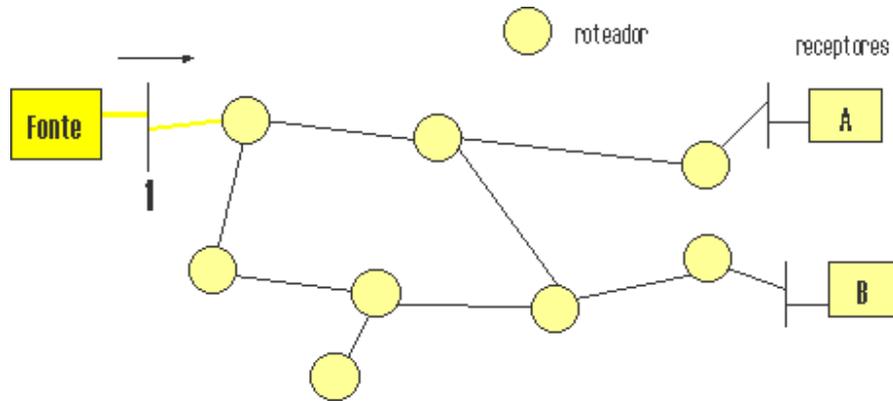


Figura 7 : Início do algoritmo *Flooding*

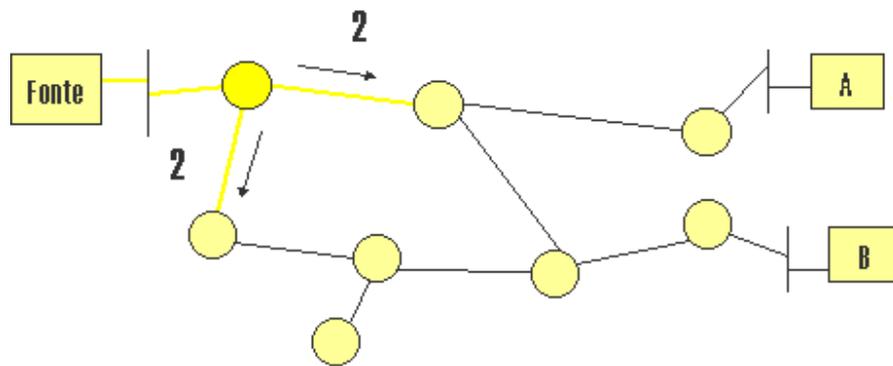
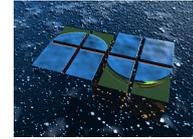


Figura 8 : Algoritmo *Flooding* - passo intermediário

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



datagrama chegou. A única informação necessária ao roteador é uma variável booleana para cada interface afim de sinalizar se determinado *link* pertence à árvore ou não.

É apresentado a seguir a figura 11 que ilustra a implementação de um algoritmo do tipo *Spanning Trees*:

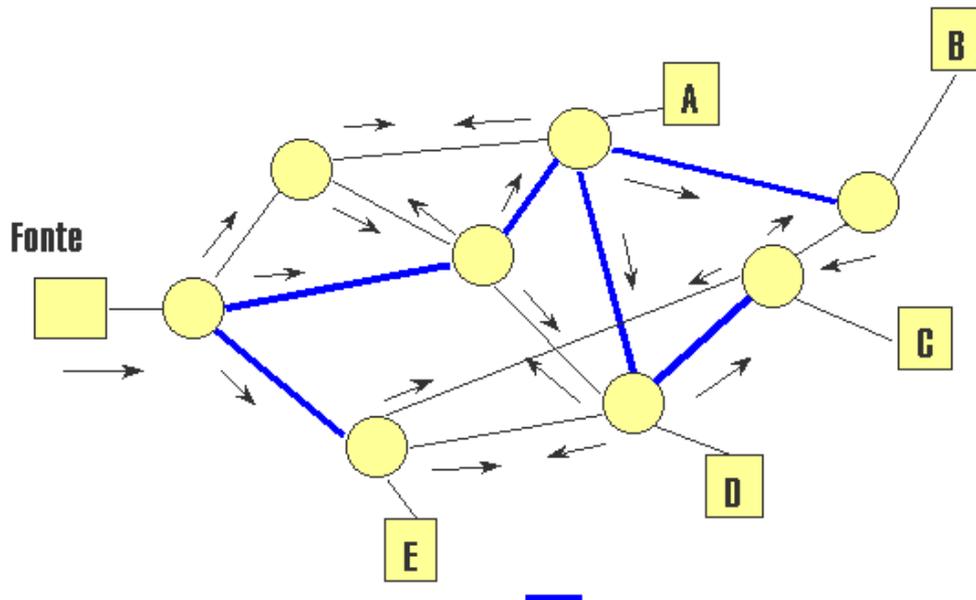


Figura 11 : Exemplo de algoritmo do tipo *Spanning Trees*

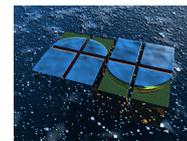
2.8.3 – RPF

Outro algoritmo bastante utilizado é o RPF (*Reverse Path Forwarding*). É basicamente uma modificação do algoritmo *Spanning Tree*. Uma *spanning Tree* implícita é construída para cada fonte. Neste caso, toda vez que roteador recebe um datagrama multicast em uma interface “X” de uma fonte “Y”, o roteador verifica se a interface “X” pertence ao menor caminho a partir de “Y”. Neste caso, o datagrama é encaminhado para todas as interfaces, à exceção de “X”, caso contrário, o datagrama é descartado.

Este algoritmo pode ser implementado facilmente considerando o fato de que roteadores locais não pertencerem ao caminho mais curto entre a fonte e um vizinho. Neste caso, o datagrama é descartado nos roteadores vizinhos. Este tipo de informação pode ser

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



facilmente obtido no caso de um protocolo de estado de *link* está sendo utilizado, caso do OSPF. Se um protocolo de roteamento do tipo vetor distância é utilizado, um roteador vizinho pode informar sua distância em saltos da fonte.

Uma vez que os datagramas são enviados através do menor caminho entre fonte e destino, pode-se dizer que é um algoritmo inteligente e de alto desempenho, haja visto que os roteadores não necessitam conhecer toda a árvore de rotas, desde que os datagramas encaminhados através de diferentes árvores de rotas, o tráfego é distribuído através de diversas árvores de rotas, fazendo melhor uso da rede. Uma grande deficiência a ser apontada é o fato do algoritmo RPF não tratar informações relativas a inscrição em grupos multicast para a construção das árvores de rotas, como pode-se observar na figura 12 :

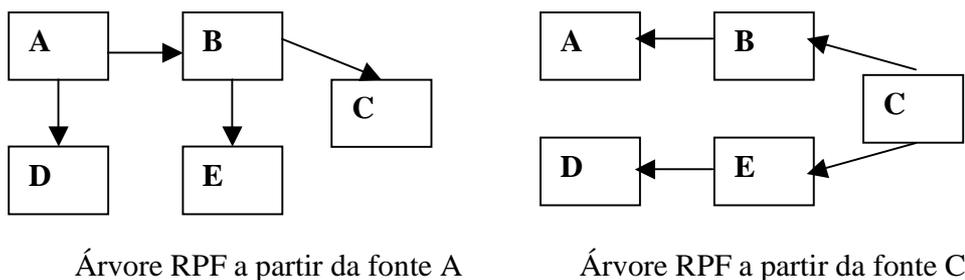


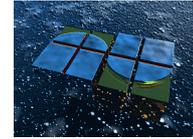
Figura 12 : Árvores RPF

2.8.4 - TRPF

Um algoritmo eficiente criado para resolver algumas limitações do algoritmo RPF é conhecido como TRPF (*Truncated Reverse Path Forwarding*). Como explicado anteriormente , através do IGMP, roteadores podem determinar se existem membros de determinados grupos presentes em subredes. Utilizando-se se tal informação, TRPF identifica se uma subrede possui outro roteador conectado à ela, caso não possua, o roteador trava a expansão da árvore de roteamento naquela interface.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Tal como RPF, o TRPF não encaminha mensagens para roteadores vizinhos, pois entendem que roteadores locais não são o menor caminho entre roteadores vizinhos e a fonte.

Apesar de eliminar tráfego redundante dentro de redes com um único *gateway*, o TRPF não elimina tráfego desnecessário em redes com mais *gateways* mesmo sem membros de grupos.

O algoritmo conhecido como RPM (*Reverse Path multicasting*) é versão aprimorada dos algoritmos RPF e TRPF . É também conhecido como RPF com *Prunes*. *Prune* pode ser traduzido como poda, no sentido figurado de cortar ramos na árvore de rotas. A árvore de rotas expande apenas em duas circunstâncias: quando encontra membros de grupos em subredes ou quando roteadores e subredes ao longo do caminho mais curto possuem membros de grupos multicast. Isto faz com que as árvores de rotas possam ser podadas conforme datagramas multicast são encaminhados para os membros dos grupos.

Inicialmente, para um dado par (fonte , grupo), o primeiro datagrama multicast é enviado utilizando-se o algoritmo TRPF. Os roteadores que não possuem uma rota direta na árvore TRPF são chamados roteadores “folha”, pois caso um roteador “folha” receba uma mensagem com um par (fonte , grupo) e não possua nenhum membro deste grupo na sua subrede, este roteador envia uma mensagem de poda (*prune*) para o roteador de quem recebeu tal pacote multicast. Esta mensagem de poda indica que datagramas multicast relativos à este par (fonte , grupo) não deve ser encaminhado ao link o qual enviou a mensagem de poda. O roteador que recebe uma mensagem de poda deve gravar a informação de poda em sua memória.

Por outro lado, se este roteador não tiver memória para isso, ele deve encaminhar mensagens aos seus vizinhos informando que não encaminhem mais datagramas referentes ao determinado par (fonte, grupo) . A “poda” em cascata fará com que a árvore original TRPF seja truncada a ponto que datagramas multicast só serão encaminhados apenas através dos *links* que levem a um nó de destino, ou seja um membro de um grupo. Basicamente procura-se eliminar ramos desnecessários como pode-se ver nas figuras de 13 a 16 a seguir:

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

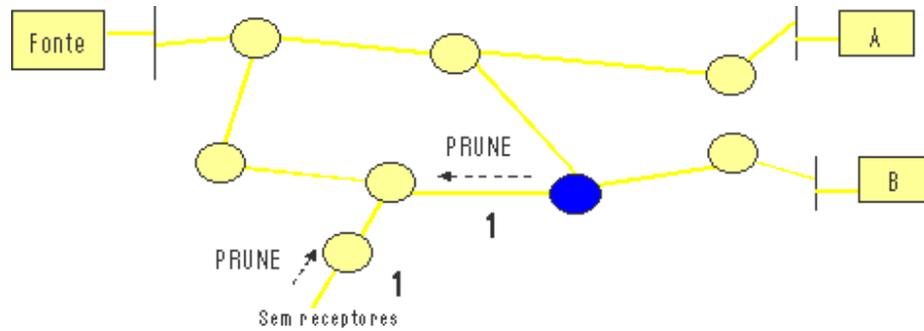
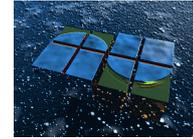


Figura 13 : Exemplo de início mensagens *Prune* no protocolo TRPF

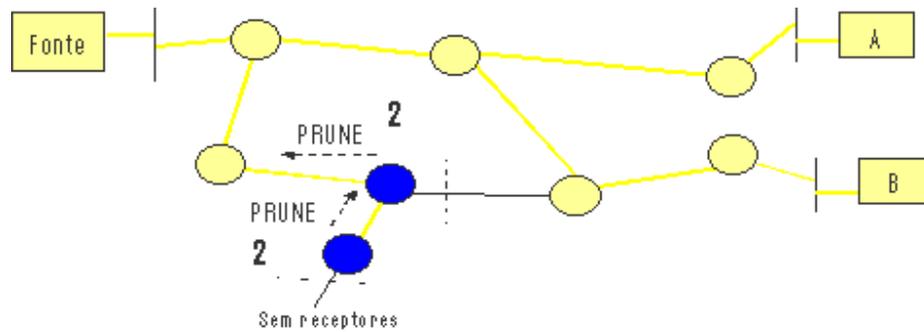


Figura 14 : Propagação de mensagem *Prune*.

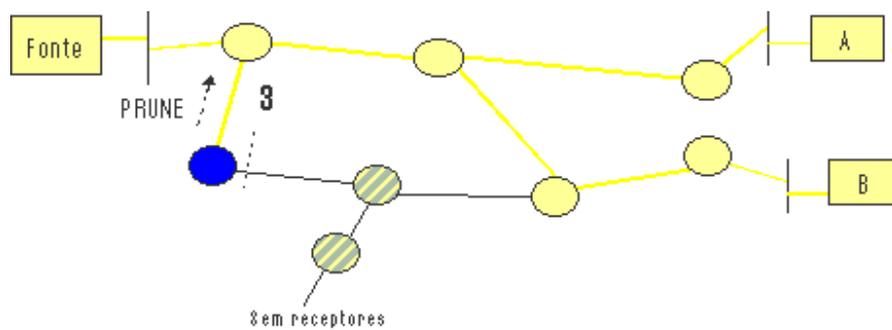


Figura 15 : Ramos podados por mensagens *Prune*.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

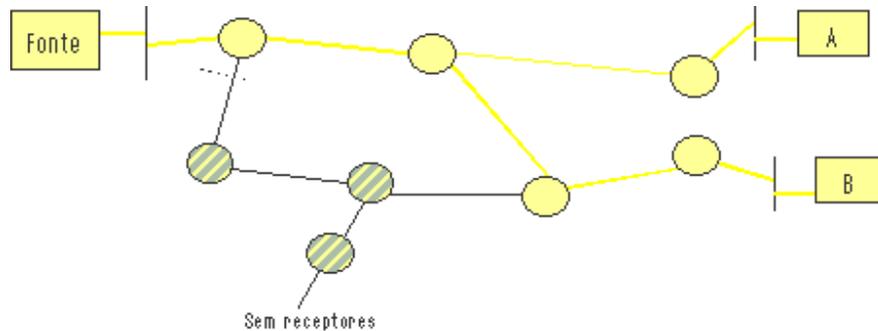
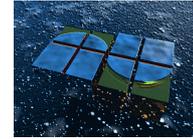


Figura 16 : Resultado da situação dos roteadores depois do processo de *pruning*.

Os círculos rajados simbolizam roteadores “podados” na árvore de rotas por não possuírem receptores.

Por fim, um dos últimos algoritmos propostos para a construção de árvores de rotas para datagramas *multicast*, CBT (*Core-Based-Tree*), diferente dos algoritmos anteriores , cria uma única árvore de rota para cada grupo. Em outras palavras, a árvore utilizada para encaminhar datagramas multicast de um grupo em particular , é única, independente da localização do nó da fonte. Um único roteador ou um conjunto de roteadores são escolhidos para implementar um roteador Core. Todas as mensagens de um determinado grupo são encaminhadas como mensagens *unicast* através deste roteador core até encontrarem um roteador o qual pertença à correspondente árvore de entrega. A partir daí as mensagens são encaminhadas através das interfaces que compõe a árvore de entrega , excluindo a interface por onde chegou tal mensagem. A figura 17 a seguir ilustra o funcionamento do algoritmo CBT :

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

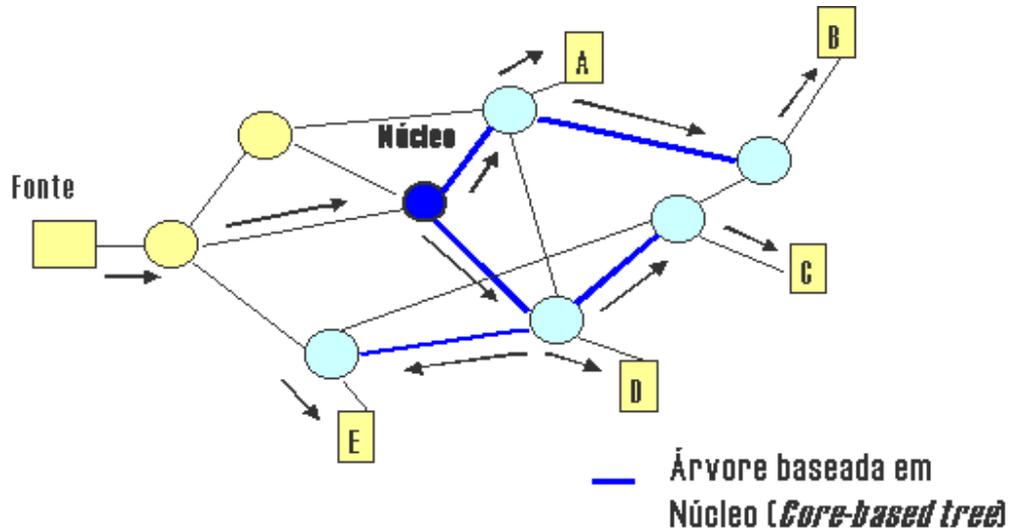
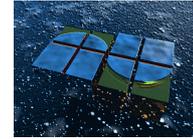
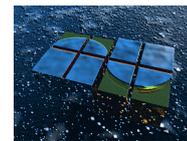


Figura 17 : Árvore baseada em CBT

Uma vez que o CBT constrói uma única árvore de rotas para cada grupo multicast, pode-se observar uma diminuição na quantidade de informação a ser mantida pelos roteadores. CBT também economiza largura de banda já que não necessita implementar algoritmos do tipo *Flooding* . Por outro lado, uma única árvore de rota para cada grupo pode levar a uma grande concentração de tráfego ao redor de roteadores *Core*, causando gargalos no fluxo de dados.

Estes algoritmos aqui tratados podem ser usados para implementar protocolos de roteamento multicast. Cada um destes algoritmos apresentam vantagens e desvantagens, sendo alguns mais eficientes em algumas situações e outros algoritmos em outras situações. Os protocolos de roteamento serão discutidos a seguir.



2.9 - Protocolos de Roteamento Multicast

Os protocolos de roteamento multicast são protocolos exclusivos de roteadores. Buscam um roteamento ótimo, procurando sempre que possível reduzir largura de banda e demandar pouca memória e processamento, tentando atingir um roteamento ótimo. Tais protocolos são divididos em duas classes básicas e suas subdivisões como pode ser observado na figura 18 a seguir:

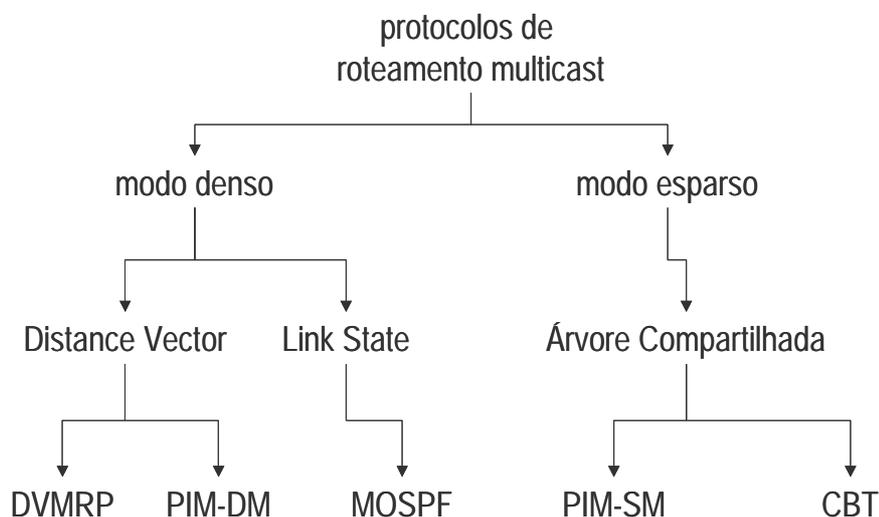


Figura 18: Classes e subdivisões de protocolos de roteamento multicast.

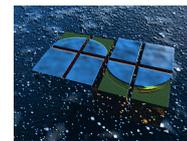
2.9.1 - DVMRP (Distance Vector Multicast Routing Protocol)

Roteadores multicast utilizam DVMRP para trocar informações a respeito de inscrições em grupos entre eles. Estas informações são úteis para que os roteadores estabeleçam rotas para que possam entregar cópias de datagramas de multicast para cada membro dos grupos.

O protocolo DVMRP apresenta semelhanças com o protocolo RIP, no sentido que permite a troca de informações entre roteadores sobre os grupos multicast existentes e os

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



custos respectivos entre as rotas. Para cada possível grupo multicast, os roteadores mantêm uma árvore de roteamento sobre um gráfico de interconexões físicas. Quando um roteador recebe um datagrama destinado a um endereço IP de multicast, uma cópia é enviada através das redes conectadas correspondentes aos galhos da árvore de roteamento.

Inicialmente, o protocolo DVMRP construiu sua árvore de rotas baseado em algoritmos do tipo TRPF, porém, atualmente é implementado através de algoritmos do tipo RPM. O primeiro datagrama multicast é enviado a partir de uma determinada fonte para um grupo multicast na forma de *Flood*. Então, mensagens de poda são usadas para truncar rotas onde não existem membros. Além disso, um novo tipo de mensagens, chamadas *Graft*, são enviadas para roteadores anteriormente podados, para que no caso de um host na subrede deste roteador se tornar um novo membro de um grupo multicast, este possa participar. As mensagens *Graft* são encaminhadas entre roteadores, salto por salto, assim como as mensagens *Prune*, até que encontrem um nó que implemente o encaminhamento. As figuras de 19 a 23 a seguir ilustram a participação de um novo host em grupo multicast através de mensagens *Graft*, enviadas a destinos já podados anteriormente.

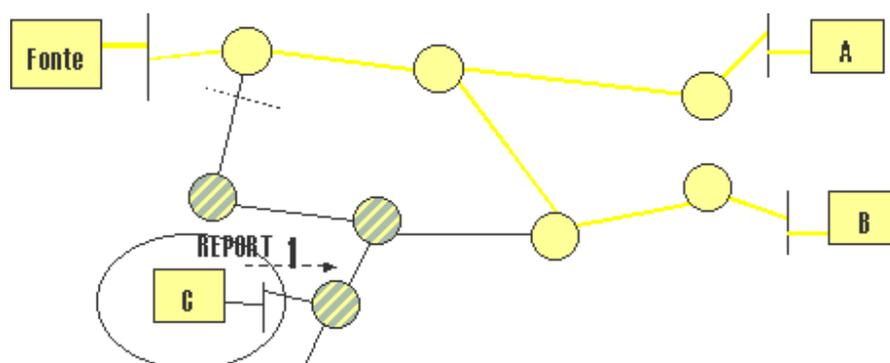


Figura 19 : Início de mensagem de adesão a grupos em redes podadas.

Backbone multicast

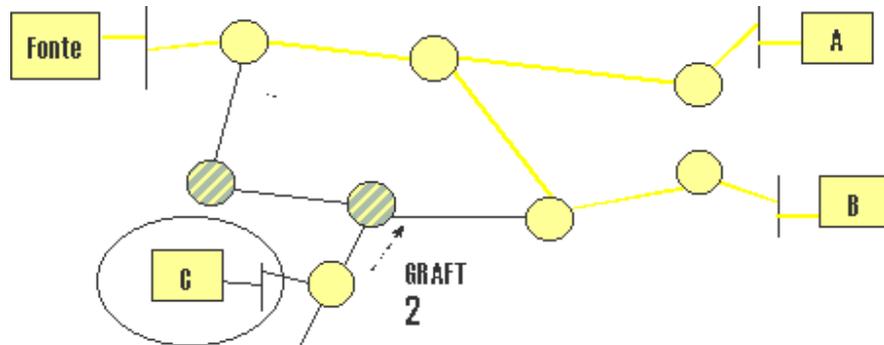
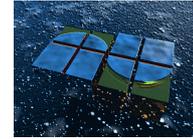


Figura 20 : Mensagem *Graft* reportando inclusão em grupos multicast

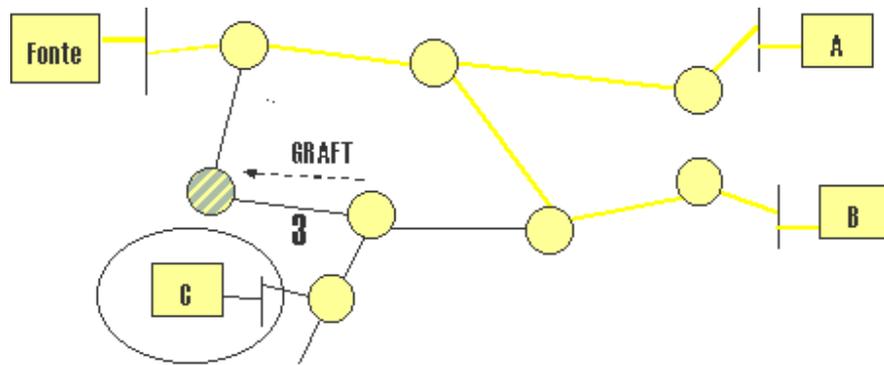


Figura 21 : Propagação de mensagem *graft* reinserindo roteadores antes podados.

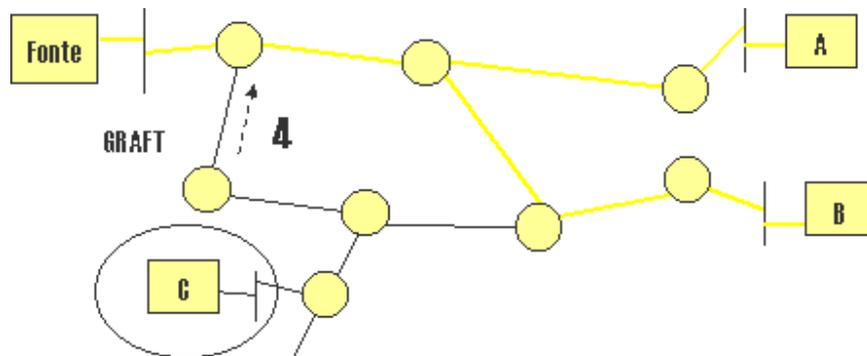


Figura 22 : Última mensagem de liberação de rotas multicast.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

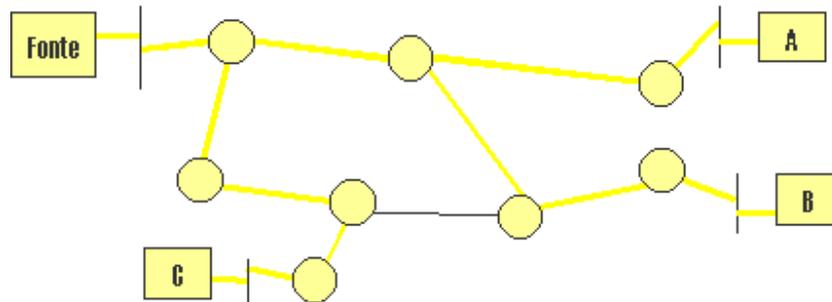
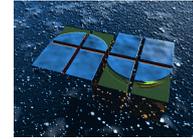


Figura 23 : *Host* reinserido nas rotas multicast.

O protocolo DVMRP utiliza mensagens IGMP para propagar as informações. Define tipos de mensagens IGMP que permitem roteadores declararem-se membros de grupos multicast, deixarem de ser membros e ainda solicitar informações de outros roteadores. Estas mensagens carregam informações relativas ao roteamento, incluindo métricas e custos.

2.9.2 - MOSPF (Multicast Open Shortest Path First)

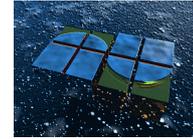
MOSPF é uma extensão do protocolo OSPF e implementado sobre ele. Também utiliza-se de mensagens IGMP e combinado com dados SPF implementa árvores de rotas multicast. Estas rotas são implementadas por árvores com caminhos mais curtos para cada par (Fonte , Grupo). Apesar de não suportar tunelamento, MOSPF opera com roteadores onde tal protocolo não está implementado.

Uma característica importante do MOSPF é que este suporta roteamento hierárquico, onde todos os *hosts* na internet são particionados em Sistemas Autônomos, chamados aqui de AS (*Autonomous Systems*). Cada AS é dividida em subgrupos chamados áreas.

A base de dados OSPF fornece um mapa completo de cada área em cada roteador. Ao adicionar um novo estado de *link* de divulgação chamado “*Group-Membership-LSA*” , o

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



qual permite que as informações referentes à localização de membros de grupos multicast sejam armazenadas em sua base de dados. Utiliza também mensagens *Prune* a fim de eliminar *links* desnecessários para o roteamento..

Roteamento Intra-Area

Desde que todos os roteadores possuem informações completas sobre a área e membros de grupos, todos os roteadores terão a mesma árvore de rotas para determinados grupos contanto que todos os membros estejam na mesma área. Pode-se notar que as árvores são construídas conforme solicitadas, fazendo com que um roteador que receba o primeiro datagrama referente a um par (fonte , grupo) possa construir sua tabela de rotas. Baseado nestas tabelas de rotas, o roteador sabe de qual interface deve esperar receber datagramas referentes a específicos grupos multicast e para qual interface encaminhá-los..

Roteamento Inter-Area

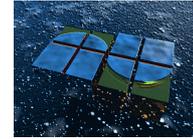
Este é o caso em que a fonte e ou membros de grupos que estejam em diferentes áreas de um AS. Neste caso uma sub-rede contendo roteadores de borda de determinada área são implementados para funcionar como roteadores Inter-Area. São responsáveis pelo encaminhamento de um resumo das informações a respeito de membros dos grupos em sua área à outros roteadores de borda de áreas, utilizando um a nova LSA.

Roteamento Inter-AS

Este é o caso em que a fonte e ou membros de grupos que estejam em diferentes AS. A abordagem para a implementação deste roteamento é muito similar ao roteamento Inter-Area. As informações pertinentes aos grupos são trocadas por roteadores na borda dos sistemas autônomos, permitindo o roteamento entre estes sistemas.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



2.9.3 -PIM (Protocol-Independent multicast)

Este é um protocolo que tem sido desenvolvido no sentido de implementar roteamento multicast independente de qualquer protocolo de roteamento *unicast*. O principal propósito deste protocolo visa grupos multicast densamente localizados e sem preocupação com largura de banda, haja visto que DVMRP periodicamente inunda a rede com algoritmos *Flood* e que MOSPF trata informações respectivas a membros de grupos através dos *links*, não resultando em protocolos muito eficientes quando membros de grupos estão muito distribuídos e largura de banda é uma limitação.

Para contornar tais problemas, PIM utiliza dois diferentes protocolos : PIM-DM (PIM-Dense Mode) e PIM-SM (PIM-Sparse Mode). A diferença básica entre os dois modos é que o modo denso parece muito com o DVMRP, porém não necessita de nenhum protocolo de roteamento *unicast*. Ao passo que no modo esparso, os roteadores devem explicitar o interesse por datagramas multicast de determinados grupos multicast. Conceitos como *Rendezvous Points* (RP) e *Core* também diferenciam o modo esparso do denso.

Em uma Recepção via *rendezvous point*, o *Host* informa ao roteador designado via IGMP. Por outro lado o roteador designado periodicamente envia PIM-*Join* ao RP. Roteadores no caminho estabelecem uma entrada de encaminhamento até o RP. Quando a fonte transmite primeiro pacote, o roteador designado encapsula pacote em PIM-*Register* e envia via *unicast* para o RP.

Os pacotes vão ao RP e de lá são distribuídos pela árvore centrada no RP.

O tráfego pode trocar da árvore compartilhada para a árvore de menor caminho com o fonte. Quando membros saem do grupo, o roteador envia mensagem PIM-*Prune* para o RP. Por fim, todos os possíveis transmissores enviam para o RP.

As figuras de 24 a 29 a seguir ilustram o protocolo PIM-SM :

Backbone multicast

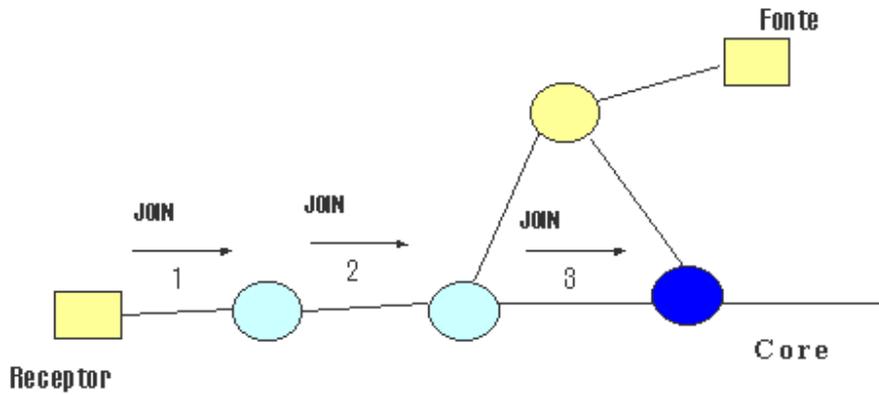
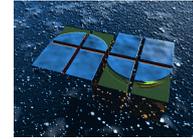


Figura 24 : Mensagem Join do protocolo PIM-SM.

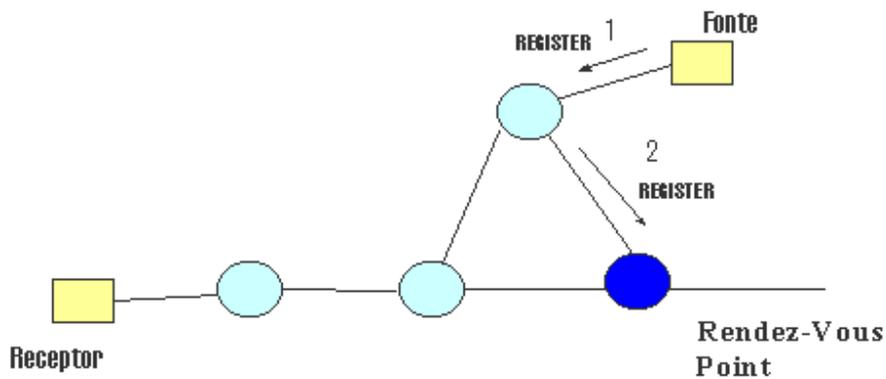


Figura 25 : Rendezvous point assume o roteamento multicast

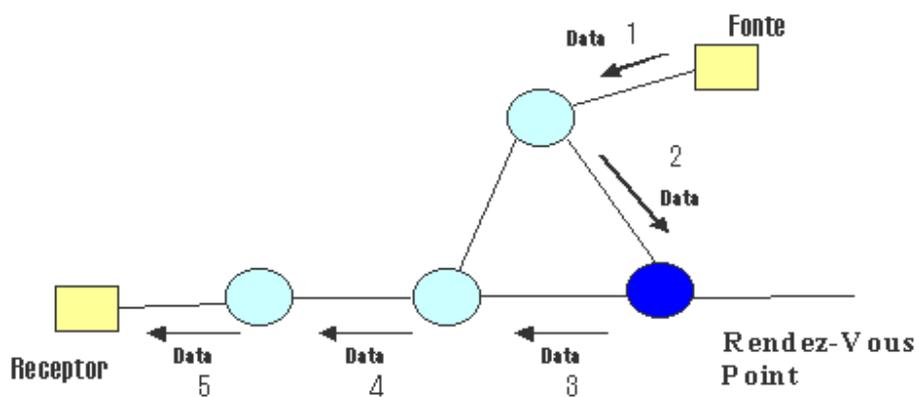


Figura 26 : Início do tráfego de datagramas multicast.

Backbone multicast

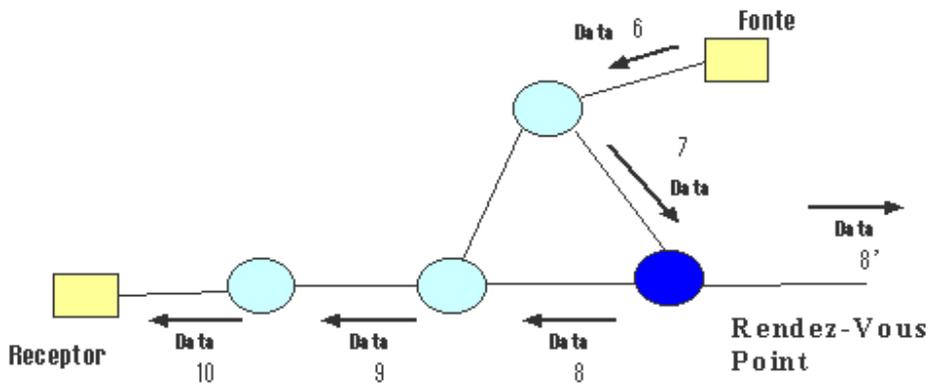
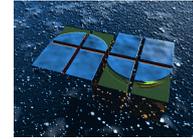


Figura 27 : Tráfego de dados utilizando o protocolo PIM-SM

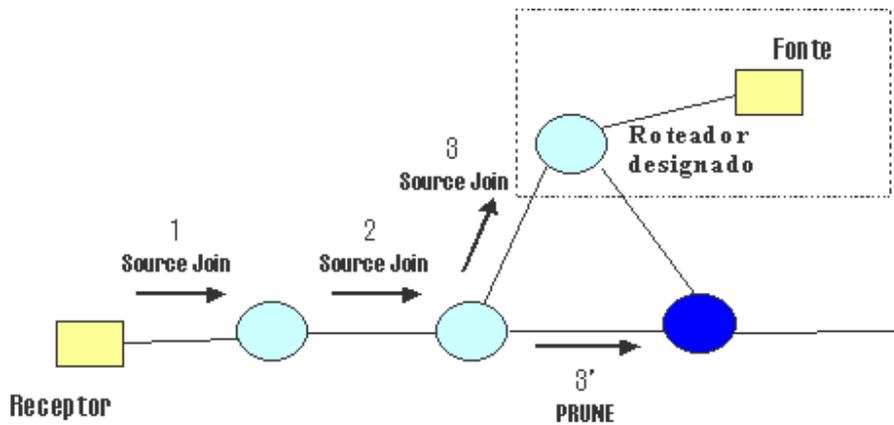


Figura 28 : Mensagens *Prune* em roteadores PIM-SM

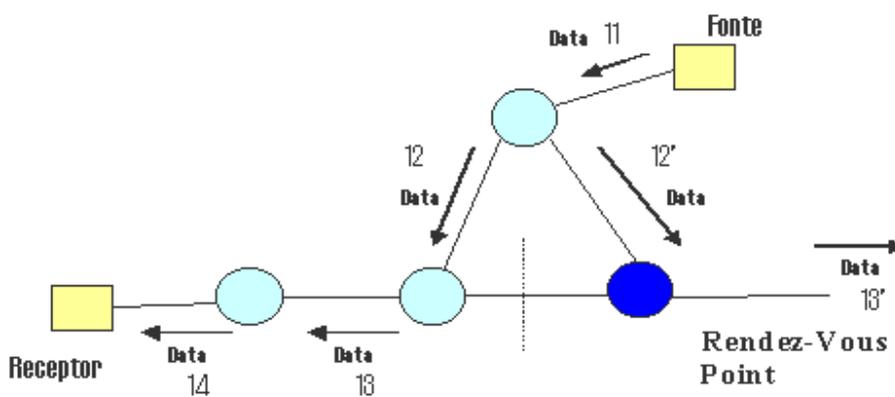
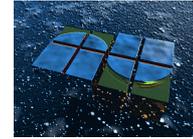


Figura 29 : Tráfego de dados utilizando o protocolo PIM-SM com rotas isoladas por *Prune*.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



2.10 - MBONE (Multicast *Backbone*)

Um MBone é essencialmente uma rede virtual implementada no topo da Internet. O MBone é formado por ilhas de redes multicast interligadas por túneis *unicast*. Através destes túneis, os datagramas multicast são encaminhados através de partes não habilitadas para multicast na Internet. Para encaminhar estes datagramas através destes túneis, é necessário que os datagramas IP multicast sejam encapsulados em datagramas IP convencionais *unicast*. O conjunto dos roteadores multicast com os túneis constituem essencialmente o MBone.

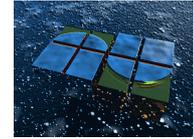
No início, o único protocolo de roteamento usado no MBone era o DVMRP, entretanto, protocolos como o MOSPF e PIM já tem sido utilizados atualmente no MBone, sendo o DVMRP ainda a maioria.

Conforme a habilitação de multicast vem se tornando nativa e mais softwares começam a utilizar tal tecnologia, o MBone tende a reduzir a quantidade de túneis até extinguí-los. É importante ressaltar que no IPv6 *multicasting* é obrigatório, o que denota uma tendência na adesão desta modalidade de comunicação em redes.

As aplicações mais utilizadas no MBone são as relativas ao tráfego de multimídia para muitos usuários, o caso das radios e televisões na Internet.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



3 – Implementação Prática

Para a execução deste projeto foi necessário inicialmente um estudo de como a tecnologia multicast funcionava e de como ela poderia ser implementada em um *backbone* experimental.

Nesta fase foi decidido que os roteadores seriam computadores da plataforma Intel com o sistema operacional Linux e munidos de um daemon chamado de mrouterd que os habilitaria para o roteamento de tráfego multicast.

Foi também decidido que seriam usadas algumas máquinas com sniffers instalados para comprovar as transmissões de multicast em uma rede Ethernet e em uma rede IP. Esta comprovação ocorreria através de pacotes IGMP, DVMRP e de mensagens UDP com um endereço multicast como destino.

3.1 – Experiência 1 - Ethernet

O primeiro passo foi o de montar uma rede bem simples que consistia de quatro máquinas ligadas em um mesmo barramento através de um Hub, como mostra a figura 30 a seguir. Esta configuração tinha como propósito o estudo da transmissão multicast em uma rede Ethernet.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

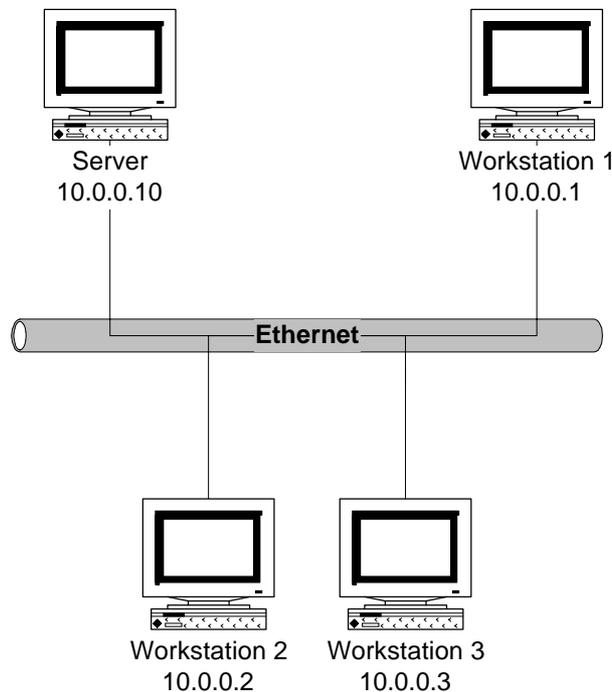
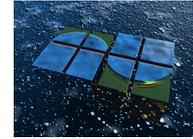


Figura 30 : Barramento Ethernet

Nesta primeira tentativa foram utilizadas máquinas com o sistema operacional Windows, com o programa chamado SDR e aplicativos de recepção de Vídeo e de White Board do Mbone Tools. A configuração das máquinas utilizadas foi de Pentiums II 300 MHz, sendo um deles munido de uma câmera de vídeo da Creative Labs chamada Web Cam 2.

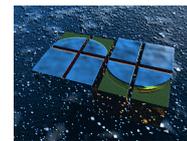
Foi inicializada uma sessão do SDR no computador com a Web Cam 2 e ela foi transmitida com sucesso para o resto da rede.

Iniciou-se então a aplicação vinculada e esta Session e os dados foram transmitidos para o mesmo endereço MAC de multicast, endereço este que foi criado a partir do seu endereço IP multicast de destino.

Para uma análise mais aprofundada, ligou-se uma quarta máquina a este barramento, máquina esta, que possuía um sniffer que seria utilizado para uma melhor visualização dos pacotes que estavam sendo transmitidos naquele barramento.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Com seu uso foi constatada e comprovada a passagem dos pacotes de IGMP e de RTP (Real Time Protocol), sendo o protocolo RTP, transmitido dentro do UDP que continha como endereço de destino o endereço multicast.

Já nesta parte foi notada uma demora de 19s entre o tempo real e o tempo em que a imagem era mostrada. A demora se deve principalmente à capacidade dos processadores das máquinas e da velocidade da interface de entrada da câmera vídeo.

3.2 – Experiência 2 - Roteador

Em um segundo estágio do projeto foi decidido que seria efetuado uma transmissão simples entre duas redes com um roteador multicast (Linux com 2 placas de rede) entre elas.

Sua configuração era como a mostrado abaixo na figura 31:

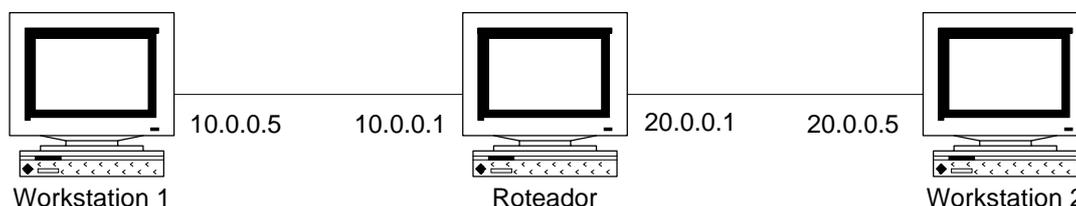


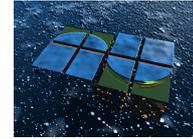
Figura 31 : Roteador multicast (Linux)

O grande problema da montagem deste roteador se devia ao fato do programa mrouterd ser uma versão beta e também de que grande parte dos experimentos que haviam sido reportados sobre o seu uso relatavam diversos problemas em sua instalação.

A primeira tentativa foi com o Linux Conectiva 5.0, esta versão foi logo abolida devido ao fato de que para facilitar sua instalação, grande parte dos aplicativos de desenvolvimento que eram instalados em versões anteriores não eram instalados e nem podiam ser selecionados para instalação. Na tentativa de instalá-los um a um após a sua instalação básica, notamos que vários componentes importantes estavam faltando ou estavam com defeito em seu arquivo de instalação. Decidiu-se pelo Linux Conectiva 3.0

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



que ainda usava o kernel da Red Hat 2.2.3. Ela também foi descartada como a anterior porque se apresentou muito instável e gerou vários problemas durante a sua instalação.

Reinstalou-se o computador com o Linux Conectiva 4.0, pôr ser uma versão mais difundida e testada. A instalação ocorreu tranqüilamente e foi possível se passar para o passo seguinte que consistia de instalar o mrouterd e configurá-lo.

Esta foi talvez a parte mais difícil do projeto devido ao fato deste *daemon* ser uma versão beta e possuir vários bugs. Haviam várias versões do mrouterd na Internet, optou-se pela versão 3.9-beta-3 pôr ser considerada pela maioria das referências visitadas como a versão mais estável.

Por se constituir de um daemon era necessário que ele fosse compilado, para que então pudesse ser executado. Já nesta parte não conseguimos compilá-lo porque ele não respondia ao comando make, apesar de possuir um arquivo Makefile (arquivo que possibilita sua compilação pôr esse comando) dentro de seu diretório.

Novamente foi pesquisado na Internet e então foi encontrada uma versão deste mrouterd que se encontrava no padrão RPM, que é um formato de instalação bem conhecido do Linux da Red Hat. Este arquivo além de compilar e instalar o mrouterd também configurava a máquina para que o serviço de roteamento fosse acionado a partir de sua instalação. Optou-se por desligar o serviço e iniciá-lo novamente em seu modo de debug, que nos possibilitaria um maior entendimento dos processos que ocorreriam ao ele efetuar o roteamento de uma rede para a outra.

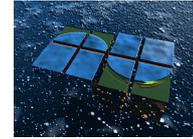
Quando o programa foi iniciado atestou-se que está versão de mrouterd não conseguia rotear devido a problemas no Kernel. Chegamos a conclusão que devido ao fato de o Kernel ser traduzido pela Conectiva ele poderia possuir algumas mudanças em relação às versões da Red Hat originais, que eram as citadas nos documentos pesquisados na Internet.(estes documentos serão apresentados no anexo 2 no final deste relatório).

Decidiu-se então pela utilização do Kernel 2.2.12-20 da Red Hat para a instalação. Nesta versão notamos que logo que o mrouterd era iniciado ocorria uma mensagem de Kernel panice impossibilitando também o seu uso.

Finalmente instalou-se a versão de Kernel 2.2.12-20 e atualizou-se para a versão 2.2.14.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



A versão 2.2.14 foi a mais tranqüila em instalação e o seu Kernel se mostrou mais estável que os anteriores.

Depois de recompilado e do computador ter sido reiniciado usando este novo Kernel iniciou-se o processo de instalação do mouted.

Novamente a versão em RPM foi usada. Em um primeiro momento parecia que a instalação havia sido bem sucedida pois na sua instalação tudo transcorreu normalmente e o daemon parecia estar funcionando bem. Passou-se para a fase seguinte que consistia de mandar um datagrama de uma rede para a outra. Neste momento foi observado que este datagrama era recebido pelo roteador, mas que ele não conseguia escrevê-lo em sua memória cache para então ser roteado, esta informação foi conseguida através da execução do mouted em modo de debug.

O processo de instalação foi repetido várias vezes até que se conclui que o mouted estava com problemas.

Optou-se então por usar uma versão do mouted que não era não estava em arquivo do tipo RPM e que segundo o dono do site (este site será mencionado no anexo 2) encontrava-se em perfeita ordem.

Todo o processo de instalação foi refeito e ainda assim era obtida uma mensagem de erro na hora do roteamento. Através deste erro foi encontrado na Internet um documento (anexo 3) que apesar e não tratar deste problema específico, mencionava a tal aparição quando na exclusão de uma certa linha do programa. Verificou-se que está linha estava faltando e então ela foi adicionada.

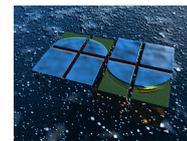
Todos os problemas mencionados acima e alguns outros estão comentados e solucionados em um documento que se encontra no anexo 1 deste projeto. Ele é um guia de instalação e configuração de um *mrouter*.

Seguidos todos estes passos foi novamente testado o roteamento e finalmente foi obtido êxito. O roteamento foi comprovado através do uso de um sniffer já mencionado anteriormente, aonde foi captado um pacote IGMP na rede 10.0.0.0, que tinha como endereço de envio o endereço 20.0.0.5 e como endereço de destino um endereço multicast.

Para efetuar esta operação nos dois sentidos da rede foi necessário criar um arquivo mouted.conf que criava e configurava o roteamento entre estas duas redes. Vale mencionar

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



que o programa SDR nos *Hosts* era iniciado na interface X e que era preciso adicionar a rota multicast nas suas duas interfaces.

3.3 - Experiência 3 - DVMRP

A tarefa que foi feita a seguir era de testar a comunicação DVMRP entre os dois roteadores multicast.

A configuração da rede para esta parte foi através de dois roteadores e três redes que possuíam a configuração abaixo mostrada pela figura 32:

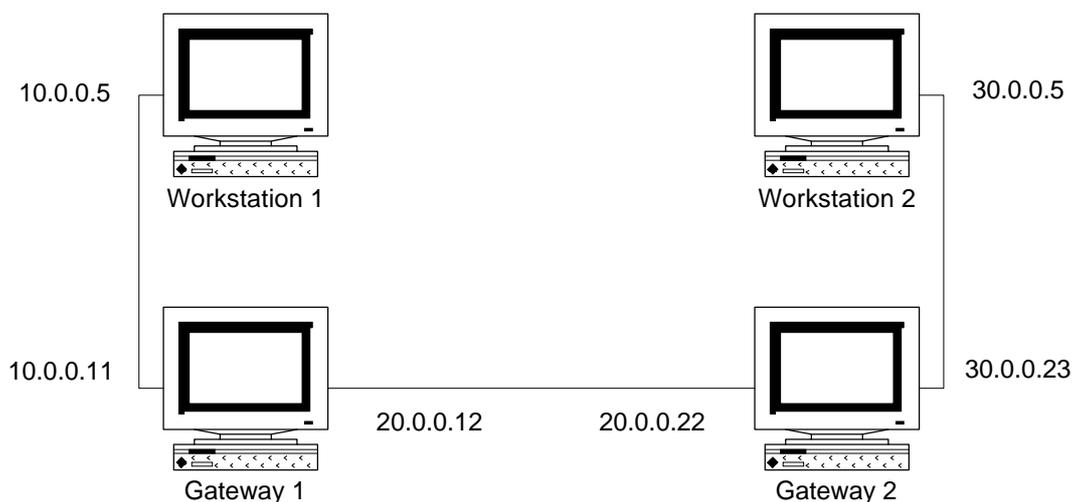
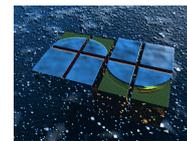


Figura 32 : Roteadores multicast

Nesta parte foi comprovado o roteamento através de um sniffer registrando a passagem dos pacotes DVMRP entre os roteadores e do recebimento de um pacote da rede 10.0.0.0 na rede 30.0.0.0 com um endereço multicast como destino. Vale lembrar que estes pacotes foram criados usando-se o programa SDR já citado anteriormente.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



3.4 - Experiência 4 – Backbone multicast

A idéia era de construir um *backbone* multicast e comparar o aspecto de tráfego, perda e do comportamento dos seus protocolos de roteamento em comparação com uma transmissão de unicast. Para esta parte foram utilizadas 12 máquinas na configuração abaixo descrita pelas figura 33 e 34:

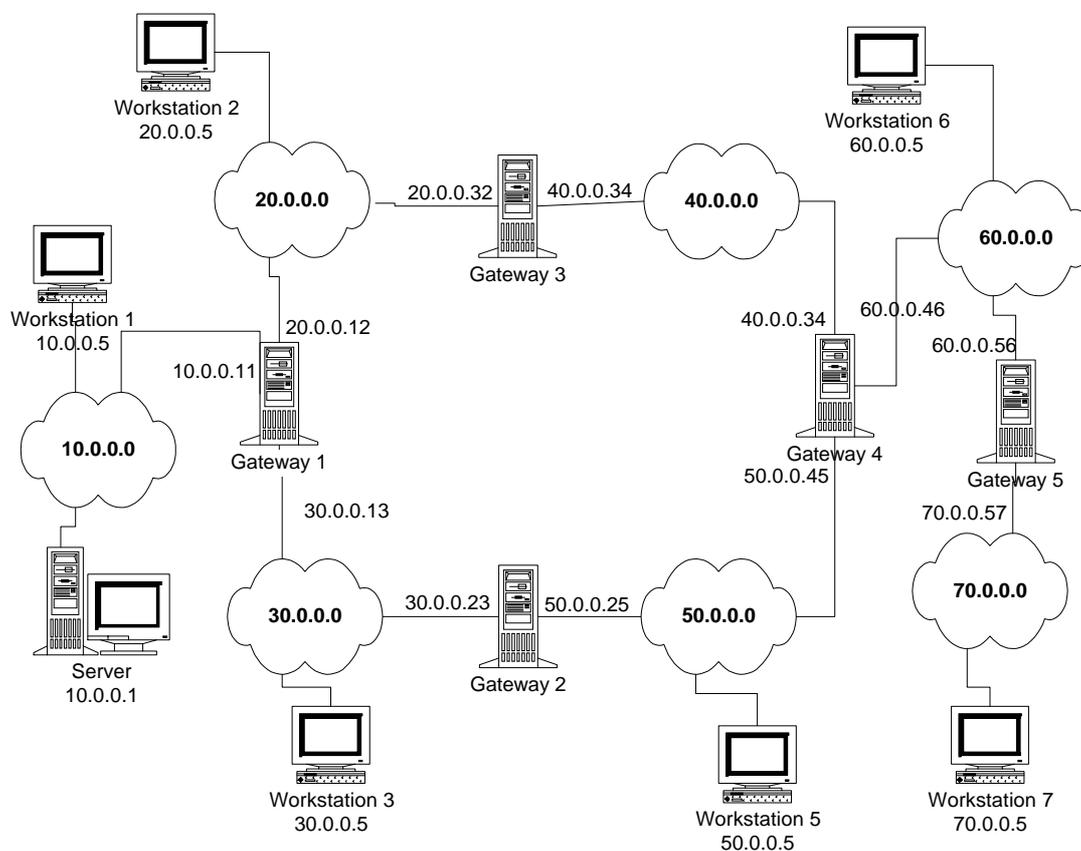


Figura 33 – configuração Lógica

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

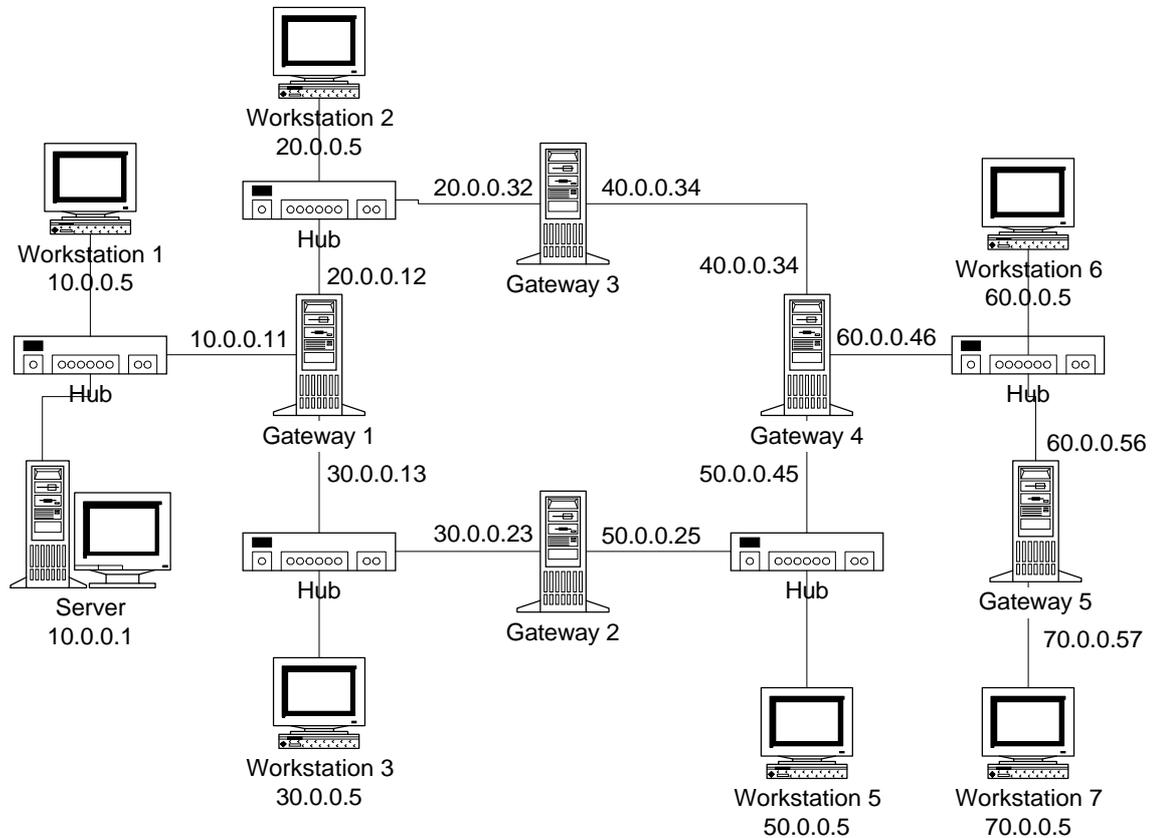
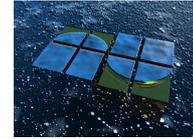
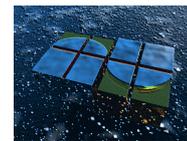


Figura 34 – configuração Física

Para testar se todo o *backbone* estava funcionando normalmente foi iniciado seu roteamento em unicast inserindo rotas estáticas na sua tabela de roteamento. Depois foram apagadas as rotas estáticas e acionado o protocolo RIP através do daemon denominado routed, ele construiu a tabela de roteamento dinamicamente sem problemas. Testado o roteamento unicast da rede foi desabilitado o protocolo RIP e reiniciado o roteamento estático.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



As tabelas inseridas são as mostradas na tabela 2:

Tabela 2 : Tabela de Roteamento- rotas estáticas

	Gateway1	Gateway2	Gateway3	Gateway4	Gateway5
10.0.0.0	eth0	30.0.0.13	20.0.0.12	40.0.0.34	60.0.0.46
20.0.0.0	eth1	30.0.0.13	eth0	40.0.0.34	60.0.0.46
30.0.0.0	eth2	eth0	20.0.0.12	50.0.0.25	60.0.0.46
40.0.0.0	20.0.0.32	50.0.0.45	eth1	Eth0	60.0.0.46
50.0.0.0	30.0.0.23	eth1	40.0.0.44	Eth1	60.0.0.46
60.0.0.0	20.0.0.32	50.0.0.45	40.0.0.44	Eth2	eth0
70.0.0.0	20.0.0.32	50.0.0.45	40.0.0.44	60.0.0.65	eth1

Neste momento foram configurados os roteadores que neste caso podem também ser chamados de gateways para rotear em multicast e foi testada sua funcionalidade instalando o programa SDR em todos os *hosts*. Foi enviada uma mensagem pela rede 10.0.0.5 e todos os *Hosts* em todas as redes a receberam sem maiores problemas. Foi repetido o mesmo teste em todos os *hosts* para avaliar se algum dos gateways não estava funcionando normalmente. Neste momento descobriu-se que era necessário adicionar as rotas de multicast para todas as interfaces porque senão o gateway só roteava em uma direção.

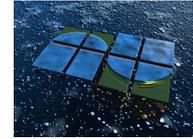
A idéia inicial era de se construir uma rádio e transmiti-la através do *Backbone* se utilizando a funcionalidade do multicast.

Para esta análise foi necessário instalar uma máquina com o Windows NT Workstation e nela foi instalado o programa chamado de Real Server. Este programa é um servidor de streaming de áudio para os programas Real Áudio Player. Para a conversão de áudio ao vivo foi utilizado um outro programa que trabalha em conjunto com o Real Áudio Server e que se chama Real Áudio Producer. Este programa pegava o som recém digitalizado pela placa de som e o convertia para o formato .rm que é o formato *streaming* da Real Networks e o direcionava para o Real Áudio Server transmiti-lo em *streaming*.

Novamente foi usado o sniffer para comprovar a transmissão de pacotes que possuíssem um endereço multicast como endereço de destino.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Vale ressaltar o processo que ocorria ao se iniciar uma transmissão multicast com o uso do Real Server.

Uma vez iniciada a transmissão digitava-se no *host* com o programa Real Player o endereço desta sessão, que no nosso caso era:

<http://10.0.0.1:8080/scalable/live.rm>

O processo que se seguia após a este foi captado por um sniffer e é descrito mais detalhadamente abaixo.

Os pacotes foram recebidos na seguinte ordem:

TCP para o endereço unicast 10.0.0.1

Procedimento de *three way handshake*

HTTP para o endereço unicast 10.0.0.1

Passava as informações que configuravam o Real Player

UDP para o endereço multicast 228.10.10.10

Dentro do pacote UDP era transportado a protocolo RTSP que continha as informações de áudio e vídeo no formato streaming.

Após a primeira mensagem que foi mandada e recebida foram mensagens unicast entre o servidor e o *host*, o resto da comunicação foi toda recebida através do endereço multiponto.

Configurado o *backbone* e o Real Áudio Server iniciou-se a bateria de testes descritos abaixo.

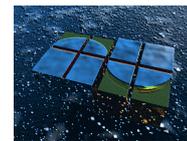
3.5 – Experiência 5 - Estudo de Roteamento e Tráfego

3.5.1 - Confirmação de transmissão em multicast

Para a confirmação da transmissão em multicast foi utilizado um programa chamado de Sniffer Pro, que possibilitou uma melhor visualização desses pacotes que estavam sendo transmitidos pela rede.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Este sniffer foi instalado na máquina cujo IP era 10.0.0.5. Esta escolha se deu devido a vários fatores:

- Prevenção de sobrecarregar o servidor com mais esta aplicação (o servidor já estava encarregado da codificação de vídeo pelo Real Producer e da transmissão em streaming pelo Real Server).
- Encontrava-se no mesmo barramento do servidor e por isso sendo capaz de captar todo o tipo de informação que era transmitido por ele.
- Possuir o Real Áudio Player e com isso se tornando capaz de atestar o recebimento de tráfego pelo endereço multicast

Outro modo comprovar a transmissão de pacotes se tornou possível através de gráficos de rota de tráfego que este sniffer nos fornecia. Estes gráficos são mostrados abaixo nas figuras 35 e 36 a fim de enfatizar as diferenças entre unicast e multicast.

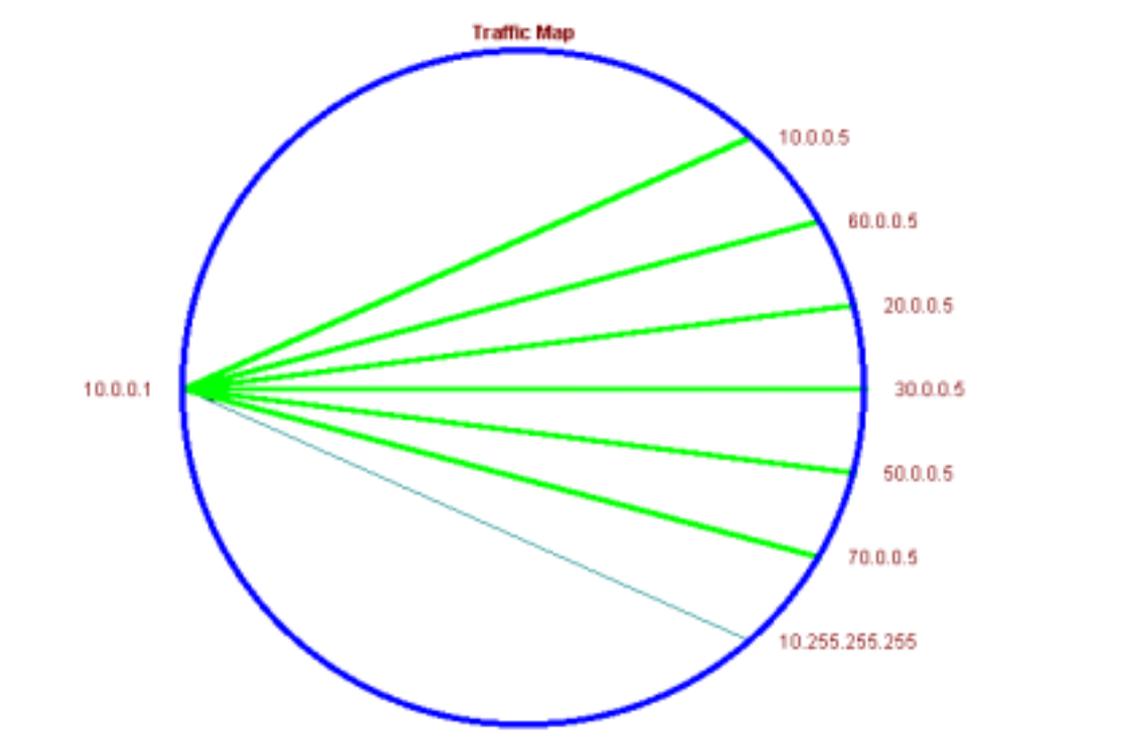
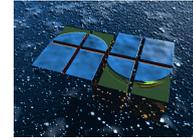


Figura 35 : Transmissão de dados em unicast

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



O gráfico de rotas apresentado acima representa bem como ocorre uma transmissão em unicast. A transmissão ocorre do servidor para cada um dos *hosts* do *backbone*.

Em seguida todos os Real Players nos *hosts* foram configurados para recepção dos dados em multicast.

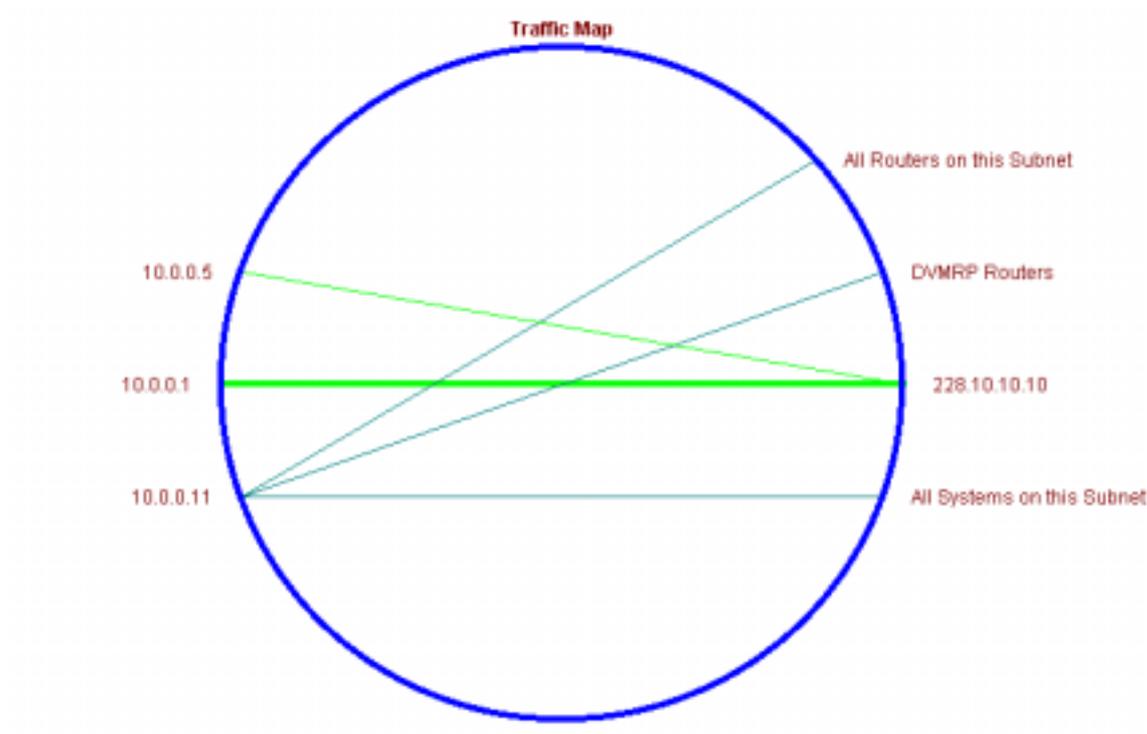


Figura 36 : Transmissão em IP multicast

Detalhando melhor as rotas acima apresentadas comprovamos a presença de três interfaces na rede.

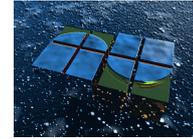
10.0.0.1 – Computador com o Real Áudio Server

10.0.0.5 – Computador com o Sniffer e com o Real Áudio Player

10.0.0.11 – Interface do Gateway 1(Roteador multicast)

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Começando pela análise do 10.0.0.11(rooteador) ele transmite dados para todos os roteadores na rede e para todos os roteadores com DVMRP (protocolo de roteamento multicast).

O endereço 10.0.0.1(Servidor) possui como único tráfego IP o endereço 228.10.10.10.

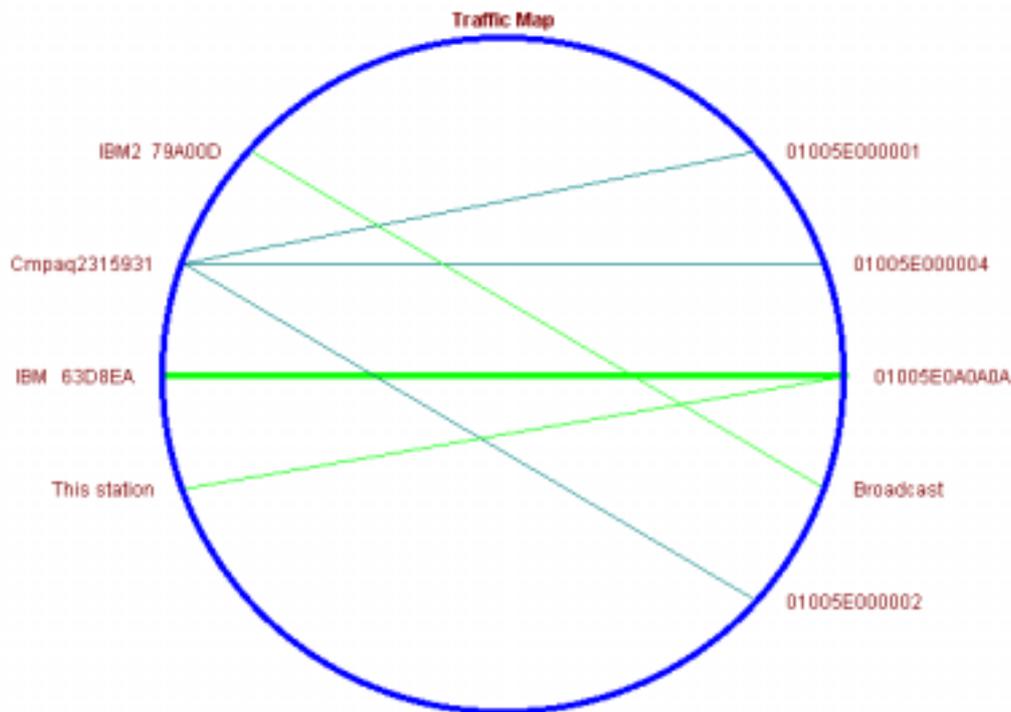


Figura 37 : Transmissão de dados em Ethernet multicast

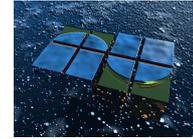
Na figura 37 pode atestar-se a transmissão em Ethernet multicast através dos endereços MAC que começam com 01.00.5E.

3.5.2 - Estudo de Roteamento

Um primeiro teste consistia de verificar o mapeamento da rede pelo roteador. Para esta verificação foi usado um comando do mrouterd
map-mbone

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Este comando mostrava todas as redes que compunham o *backbone* além de mapear todas as interfaces dos outros roteadores como visto abaixo

```
[root@localhost mrouterd-3.9-beta3]# ./map-mbone
```

```
multicast Router Connectivity:
```

```
10.0.0.11: alias for 20.0.0.12
```

```
20.0.0.12: <v3.255>
```

```
30.0.0.13: 30.0.0.23 [1/1/querier]
```

```
20.0.0.12: 20.0.0.32 [1/1/querier]
```

```
10.0.0.11: 10.0.0.11 [1/1/querier]
```

```
30.0.0.13: alias for 20.0.0.12
```

```
30.0.0.23: alias for 50.0.0.25
```

```
50.0.0.25: <v3.255>
```

```
50.0.0.25: 50.0.0.45 [1/1/querier]
```

```
30.0.0.23: 30.0.0.13 [1/1]
```

```
20.0.0.32: alias for 40.0.0.34
```

```
40.0.0.34: <v3.255>
```

```
40.0.0.34: 40.0.0.44 [1/1/querier]
```

```
20.0.0.32: 20.0.0.12 [1/1]
```

```
40.0.0.44: alias for 60.0.0.46
```

```
50.0.0.45: alias for 60.0.0.46
```

```
60.0.0.46: <v3.255>
```

```
60.0.0.46: 60.0.0.56 [1/1/querier]
```

```
50.0.0.45: 50.0.0.25 [1/1]
```

```
40.0.0.44: 40.0.0.34 [1/1]
```

```
60.0.0.56: <v3.255>
```

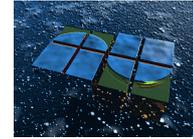
```
70.0.0.57: 70.0.0.57 [1/1/querier]
```

```
60.0.0.56: 60.0.0.46 [1/1]
```

```
70.0.0.57: alias for 60.0.0.56
```

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



O processo seguinte consistia de estudar a eficácia do método de roteamento em multicast.

Para isso foi inicializado o programa Real Player em todos os *hosts* com o objetivo de gerar tráfego multicast em todas as redes. Feito isso desligou-se o programa de todas as redes menos da rede 70.0.0.0. Foi atestado que o tráfego de dados multicast que estava sendo requisitado por 70.0.0.5 não passava pelos dois caminhos possíveis e somente por um deles que era o que passava pelas redes 20.0.0.0 e 40.0.0.0. Este teste confirmou a eficácia do protocolo DVMRP em traçar rotas.

3.5.3 - Estudo de tráfego.

O objetivo do estudo de tráfego era de medir quanto de tráfego era economizado fazendo-se o uso da tecnologia de multicast em relação a uma transmissão em unicast.

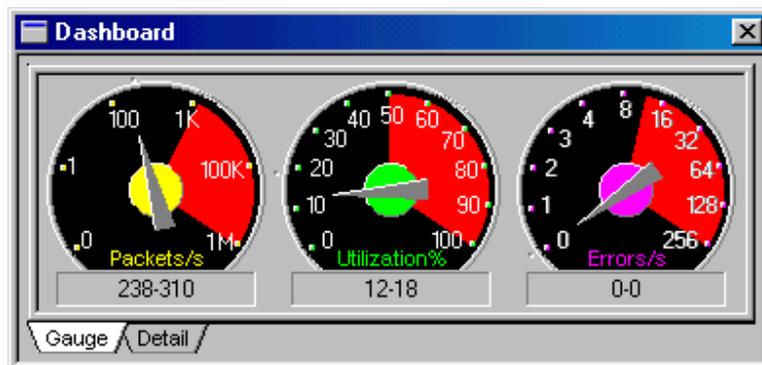


Figura 38 : Tráfego em unicast

A figura 38 corresponde ao tráfego de pacotes ao se utilizar unicast para a transmissão de dados de vídeo e áudio estéreo.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

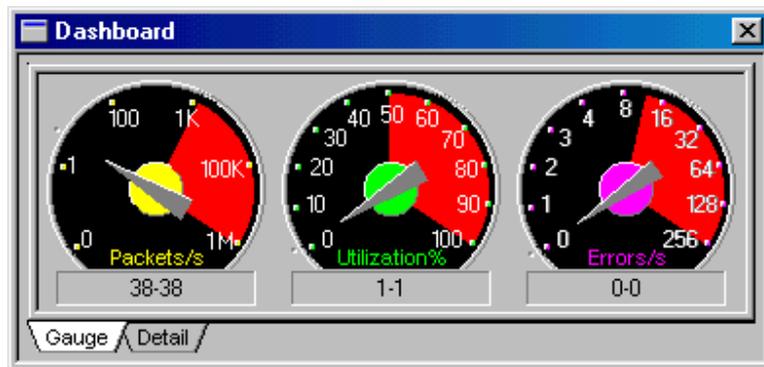
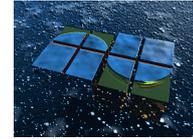


Figura 39 : Tráfego em multicast

Analisando-se o mesmo tipo de transmissão mas utilizando-se o multicast, foi comprovado o ganho em termos de tráfego. Este fato fica bem claro se compararmos o primeiro mostrador da figura 39 acima (multicast) com o mesmo mostrador da figura da transmissão em unicast.

3.5.4 - Economia em termos de processamento no servidor

O objetivo deste estudo era o de comprovar a economia no servidor quando em uso da tecnologia multicast mostrados nas figura 40 e 41e 42 a seguir:

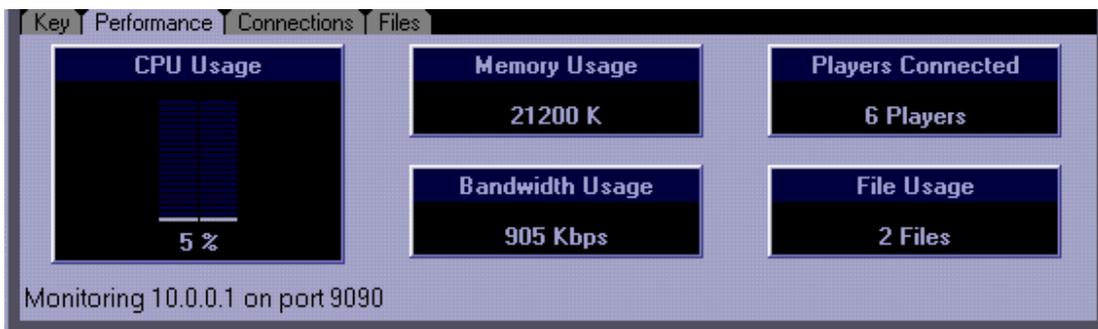
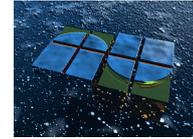


Figura 40 : Processamento de Servidor para 6 usuários em multicast

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Primeiro foram conectadas seis máquinas requisitando transmissões em unicast. Atestou-se um uso do processador da ordem de 5%.

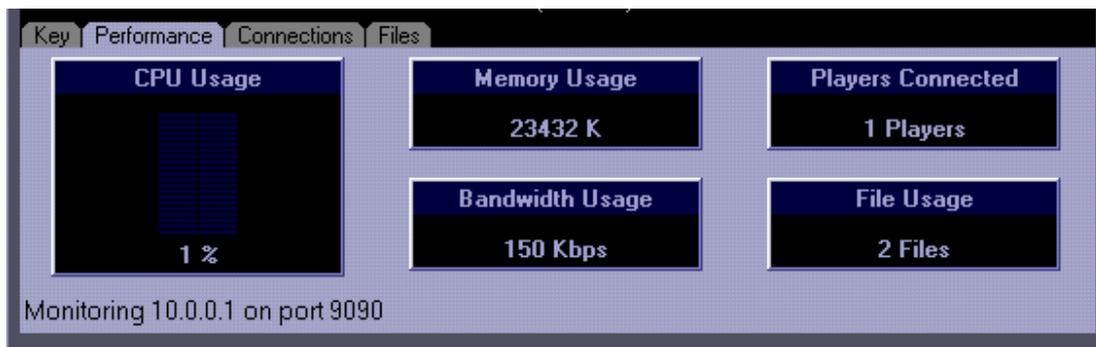


Figura 41 : Processamento do servidor para 1 usuário em unicast

Em seguida registrou-se o nível de processamento necessário para atender apenas uma máquina.

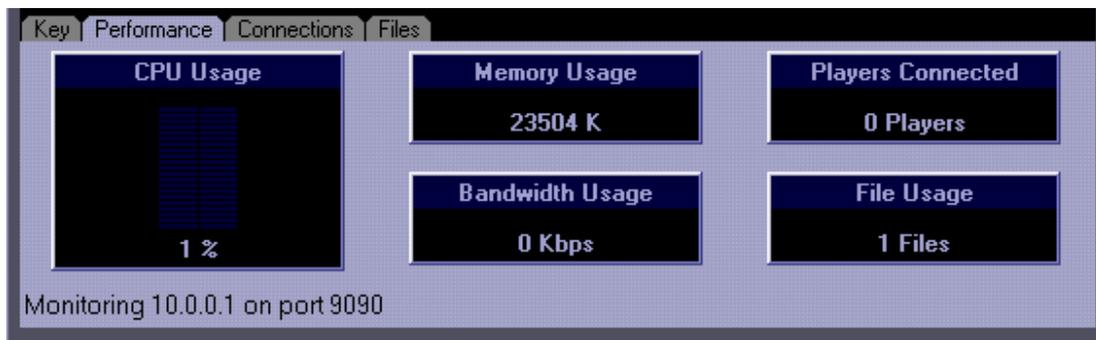
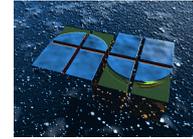


Figura 42 : Processamento de servidor para 6 usuários em multicast

Nos gráficos e tabelas acima apresentados fica claro que o processamento no servidor em uma transmissão para seis usuários em multicast se assemelha em muito com o da transmissão para um usuário em unicast . Isto ocorre porque como já explicado anteriormente o servidor só precisa mandar os dados uma vez para endereço comum a todos eles. Também são comprovadas as vantagens em termos de processamento, visto que ele ficou bem menor que os da transmissão em unicast.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Vale ressaltar que os exemplos ficaram um pouco prejudicados na análise de processamento devido ao pequeno número de usuários utilizados, que acarretou um tímido uso da capacidade de processamento do Servidor.

Nas experiências acima citadas foi feito o uso do software Real Áudio Player. Abaixo está um exemplo da qualidade de sua recepção e algumas das informações disponíveis em seu software a respeito da transmissão.



Figura 43 : Exemplo de imagem transmitida em multicast

Na figura 43 acima fica comprovada a qualidade da transmissão e que o áudio foi transmitido em estéreo. Também é visível na parte inferior esquerda que a taxa de recebimento de dados foi de 150,9 Kbps.

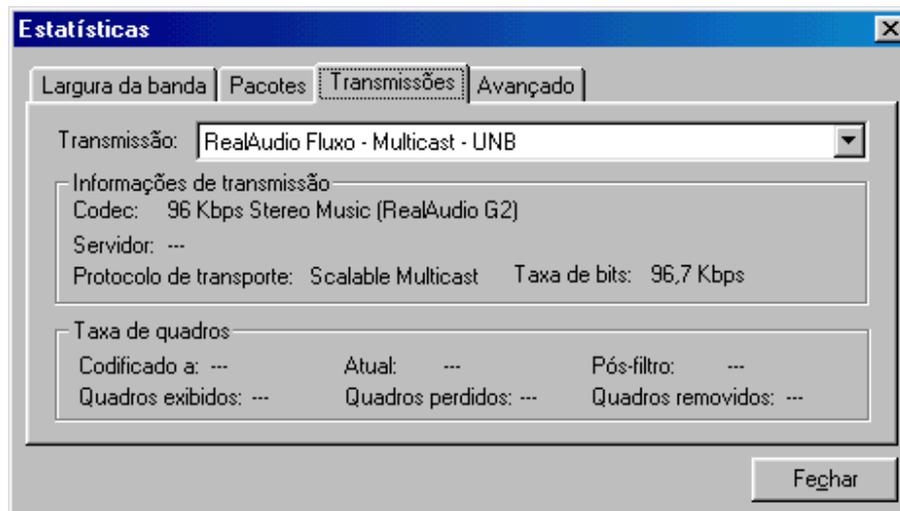
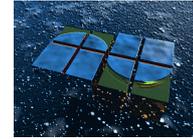


Figura 44 : Informações conseguidas a respeito da transmissão no Real Áudio Player

Na figura 44 fica bem claro que o player reconheceu a transmissão como de multicast e que a taxa de bits usada para codificar a transmissão no Real Áudio Producer foi de 96,7 Kbps.

4 - Aplicações

4.1 Sniffer Pro

Fabricante: Network Associates

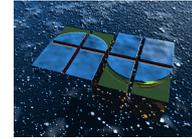
Versão: 3.0.99

Faz a placa de rede trabalhar em modo promíscuo habilitando-a a receber e processar todo o tipo de pacote que trafega pela rede mesmo que não seja endereçada a máquina onde ele está instalado.

Este programa foi largamente utilizado para a comprovação das transmissões em multicast e para se fazer uma análise do ganho em termos de tráfego com a utilização de um endereço IP de grupo.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



4.2 Real Networks

Real Server

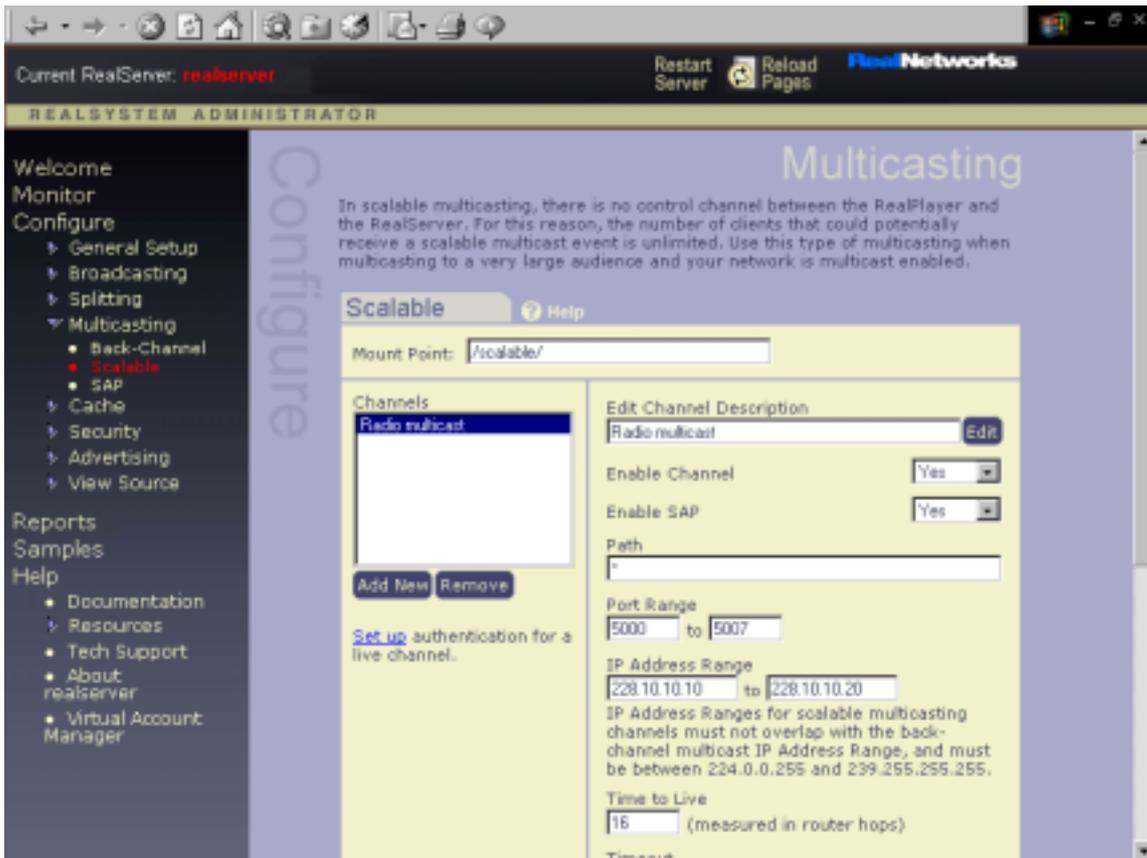


Figura 45 : Interface do Real Server – Configuração multicast escalável

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

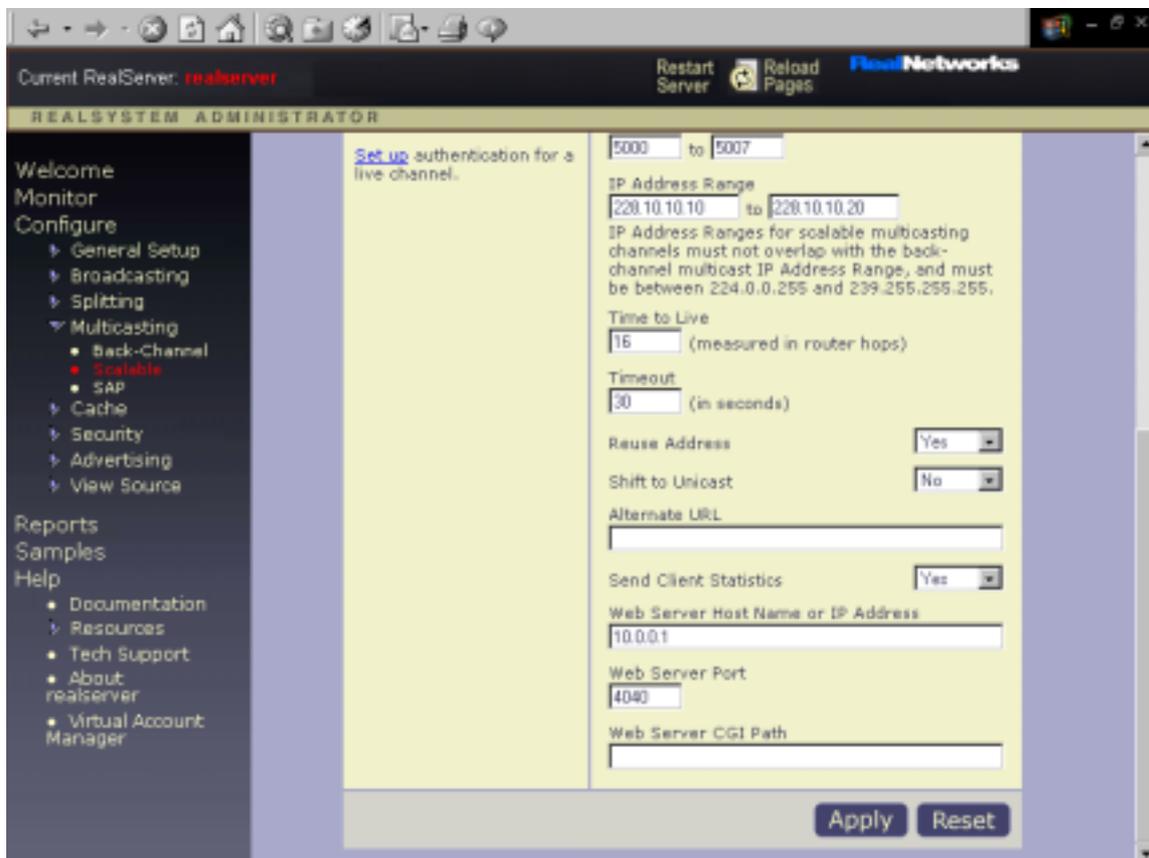
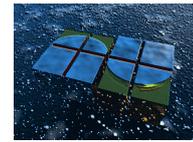


Figura 46 : Interface do Real Server – Configuração multicast escalável (continuação)

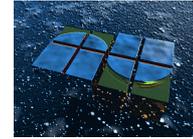
Fabricante: Real Networks

Versão: 6.1.3.871

Estudando o programa foi descoberto que ele possui duas formas de transmissão em multicast. Uma chamada de Back Channel e outra chamada de Scalable. Com a opção de back channel a configuração do servidor era feita automaticamente sendo apenas necessário se escolher um intervalo de endereços multicast que o programa se encarregava do resto do processo. Apesar de parecer a melhor escolha, este método ainda fazia uso de TCP/IP em sua transmissão a fim de construir um canal de controle entre o transmissor e o receptor para um maior controle transmissão, vale lembrar que em multicast não existe controle de erro como no TCP, porque ele é transmitido no UDP. Apesar de apresentar uma ótima forma de melhorar a transmissão de multicast essa forma também limita o número de

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



participantes porque ainda se utiliza da transmissão em unicast para esta transmissão do canal de controle.

A outra forma de transmissão chamada de scalable precisa de uma configuração mais apurada porque e não possui o canal de controle citado anteriormente, mas oferece uma capacidade ilimitada no âmbito do número de usuários.

Nesta configuração a única mensagem que não é endereçada a um endereço multicast é a primeira que possui o endereço do Servidor em unicast, que é quando ele informa ao *host* qual de ser o canal em multicast que ele deve utilizar. Após esta pequena transmissão ele só se utiliza do endereço de multicast para transmissão. O protocolo utilizado por este programa é o RTSP que significa Real Time Streaming Protocol.

As figuras 45 e 46 acima mostram quais os parâmetros foram adotados para a transmissão de áudio e vídeo em multicast.

Real Player

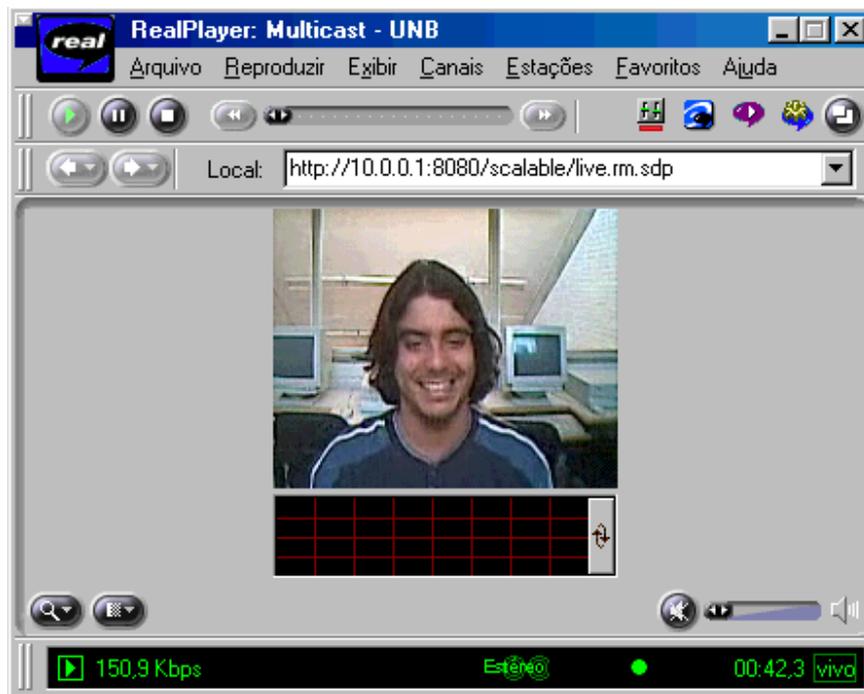
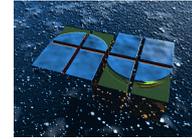


Figura 47 : Interface do Real Player G2

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Fabricante: Real Networks

Versão : 7.0

Programa utilizado nos *hosts* para a recepção dos dados de vídeo e som. Através dele também é possível analisar a taxa de transmissão de dados e a taxa de amostragem dos dados recebidos.

Real Producer

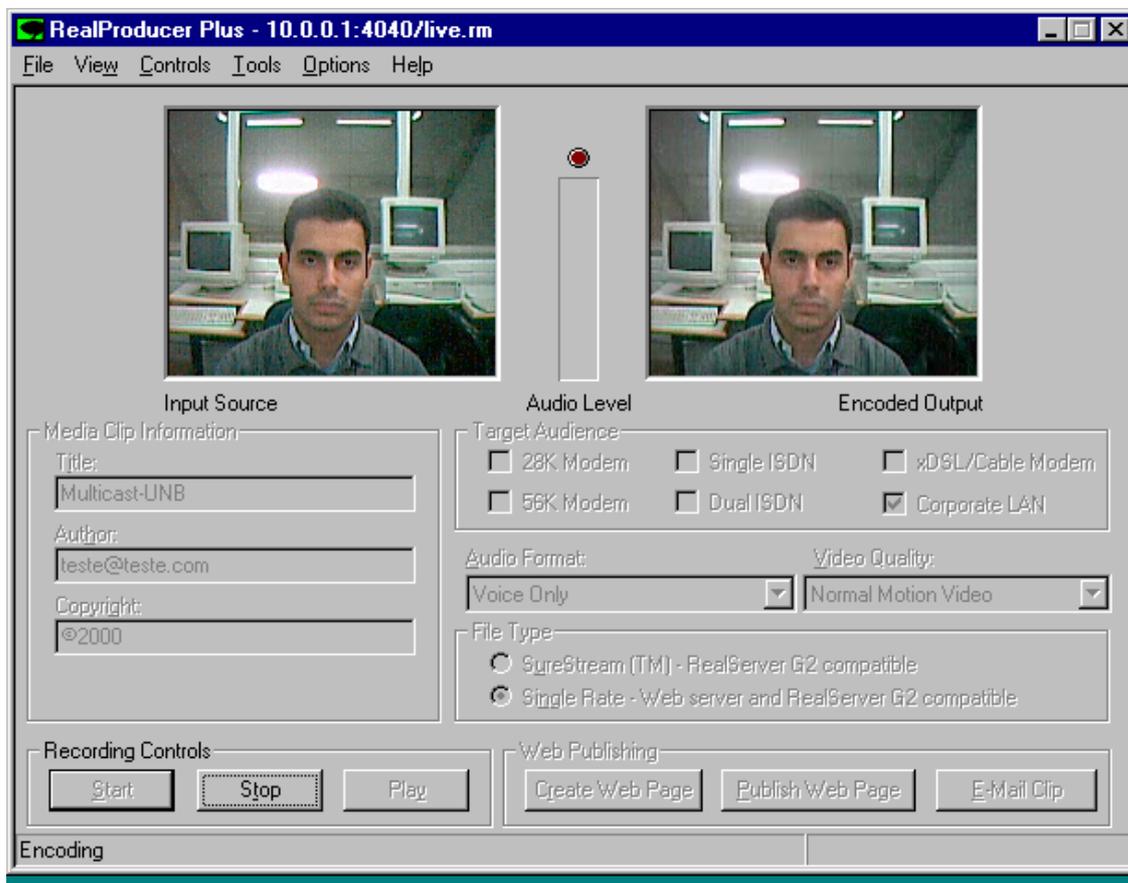


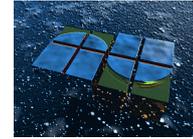
Figura 48 : Interface do Real Producer

Fabricante: Real Networks

Versão : 6.0.3.271

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Este programa do pacote da Real Networks era o responsável pela codificação e compressão dos dados de áudio e vídeo que eram recebidos e de enviá-los para o Real Server que era encarregado da transmissão em streaming.

O método de codificação de voz utilizado é o ACELP-NET.

4.3 Mouted

Fabricante : Stanford University

Versão : 3.9-beta3 - alterada

Mudanças efetuadas por

André Luiz Bacellar de Miranda - andremiranda@ig.com.br

Yamar Aires da Silva – yamar@mymail.com.br

Ricardo Staciarini Puttini - orientador

Eduardo Porto da Silva – eduardo@nti.com.br

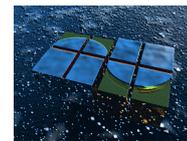
O programa Mouted implementa dois serviços fundamentais: a propagação de rotas e o tunelamento. Mouted utiliza DVMRP para propagar informações de roteamento multicast entre roteadores. Isto é feito através da interpretação das informações de roteamento para a construção da tabela de roteamento multicast, utilizando-se um algoritmo conhecido como Truncated Reverse Path broadcast (TRPB) . Porém, Mouted não substitui protocolos convencionais de propagação de rotas, ao contrário, trabalha em conjunto com estes protocolos.

Tunelamento é uma implementação essencial para resolver um grande problema em multicast, o fato de a maior parte dos roteadores não suportar multicast. O programa Mouted possibilita um datagrama multicast tunelar através de redes unicast entre roteadores multicast. Tunelamento será discutido propriamente a seguir.

Antes, porém, é importante ressaltar que o programa mouted possui um arquivo de configuração, o qual pode ser configurado para implementar ambos os serviços de propagação de rotas e de tunelamento. Este arquivo contém opções específicas a serem implementadas, as quais devem ser selecionadas de acordo com a operação desejada. Estas

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



opções especificam os grupos multicast que podem ser anunciados pelo mrouter em cada interface bem como informações relativas ao encaminhamento dos datagramas. Ainda, associam informações relativas à métrica e limiares para cada rota. A métrica permite designar um custo para cada rota, enquanto o limiar trata o TTL mínimo no respectivo campo do cabeçalho IP.

Tunelamento consiste de um acordo entre programas Mrouter implementados em dois roteadores. Cada roteador encaminha datagramas multicast enviados pelos hosts para grupos multicast, os quais possuem túneis configurados. Quando um datagrama multicast chega ao endereço de destino correspondente ao túnel, O programa Mrouter envia o datagrama para o Mrouter no outro roteador utilizando um endereço convencional unicast. Quando o datagrama chega de um túnel, o Mrouter extrai o datagrama multicast e o entrega à rede local de destino. Isto é feito encapsulando-se datagramas IP multicast em datagramas IP convencionais como se pode observar na figura 49:

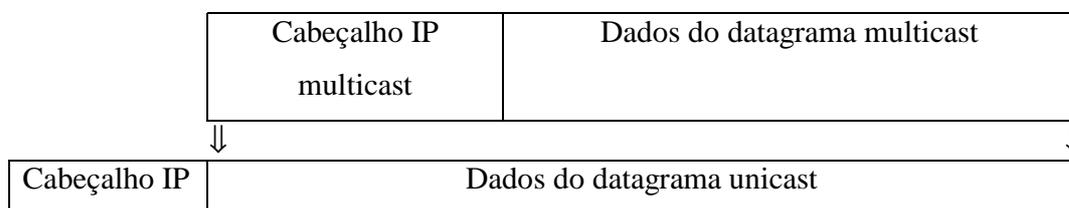
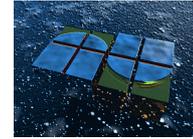


Figura 49 : Encapsulamento de datagramas IP multicast em datagramas IP

Quando um túnel é criado, o programa Mrouter enxerga as redes que interconectam dois roteadores multicast como uma única rede física. O campo de TTL no datagrama IP é decrementado de uma unidade ao atravessar o túnel, enquanto o datagrama no qual vai encapsulado tem seu próprio TTL independente, o que torna possível também limitar o número de saltos permitido em um túnel.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



4.4 Mbone Tools

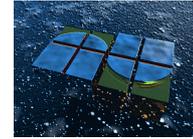
Tabela 3 : Ferramentas do Mbone

Application web-page	Bug report address	Solaris	SunOS4	Irix 6.2	Linux	FreeBSD	Windows
<u>Robust Audio Tool: RAT</u>	rat-users@cs.ucl.ac.uk	<u>3.0.35</u> <u>4.2.x</u>	<u>3.0.34</u> -	<u>3.0.35</u> -	<u>3.0.33</u> <u>4.2.x</u>	<u>3.0.34</u> <u>4.2.x</u>	<u>3.0.35</u> <u>4.2.x</u>
<u>Video Conference: VIC</u>	vic@cs.ucl.ac.uk	<u>2.8ucl-1.1.3</u> <u>2.8ucl-1.1.3</u> (XIL)	<u>2.8</u>	<u>2.8ucl-1.1.3</u>	<u>2.8ucl-1.1.3</u>	<u>2.8ucl-1.1.3</u>	<u>2.8ucl-1.1.3</u>
<u>WhiteBoard: WB</u>	n/a	<u>1.60</u>	<u>1.60</u>	<u>1.60</u>	<u>1.59</u>	<u>1.59</u>	n/a
<u>WhiteBoard: WBD</u>	wbd@cs.ucl.ac.uk	<u>1.0ucl4</u>	-	<u>1.0ucl4</u>	-	-	<u>1.0ucl4</u>
<u>Network Text Editor: NTE</u>	nte@cs.ucl.ac.uk	<u>1.7.0</u>	-	<u>1.7.0</u>	-	-	<u>1.7.0</u>
<u>Session Directory: SDR</u>	sdr@cs.ucl.ac.uk	<u>2.9</u>	-	<u>2.9</u>	-	<u>2.9</u>	<u>2.9</u>
<u>SDP Parser: SPAR</u>	K.Hasler@cs.ucl.ac.uk	-	-	-	-	-	-
<u>ReLaTe Integrated Interface</u>	relate-ui@cs.ucl.ac.uk	<u>2.1</u>	-	<u>2.1</u>	-	-	<u>2.1</u>
<u>MMCR (Java Client)</u>	L.Lambrinos@cs.ucl.ac.uk	<u>1.1</u>					<u>1.1</u>
<u>MMCR (Server)</u>	L.Lambrinos@cs.ucl.ac.uk	<u>1.1.1</u>	-	-	-	-	-
<u>UTG (Java Client)</u>	utg-support@cs.ucl.ac.uk	<u>1.3</u>					<u>1.3</u>
<u>UTG (Server)</u>	utg-support@cs.ucl.ac.uk	<u>1.2a</u>	-	-	-	-	-
<u>RTP Quality Matrix</u>	c.perkins@cs.ucl.ac.uk	<u>1.0.0</u>	-	-	-	-	<u>1.0.0</u>

Fabricante : UCL Networked Multimedia Research Group

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Sdr – Multicast Session Directory



Figura 50 : Figura do SDR

Fabricante: Projeto MICE

Versão: 2.6.1

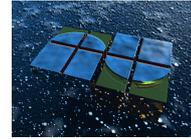
Este programa merece uma maior atenção porque ele foi escolhido para testar todas as redes montadas. Isto se deu pelo fato de sua fácil instalação e de não requerer muito do processador.

Ele é um programa chave do conjunto de programas chamado de Mbone Tools e que são largamente utilizados no Mbone, já descrito anteriormente. Seu trabalho é o de fazer uma propaganda das *Sessions*, que são as transmissões multicast que estão ocorrendo, ou vão ser transmitidas no Mbone. Seu funcionamento é bem simples, ele utiliza um endereço de multicast reservado e comum a todos os programas SDR para transmitir informações sobre as *Sessions* que irão ocorrer, quais serão os endereços multicast utilizados, e qual aplicativo do Mbone Tools será utilizado. Entre estes programas estão programas de Vídeo Conferência, de Transmissão de Áudio, White Board e outros.

Um usuário que possua uma máquina com o SDR instalado cria uma *Session*, nesta *Session* ele coloca informações sobre o que será a transmissão, que horas irá ocorrer, o aplicativo que será usado, o responsável pela transmissão e qual o TTL. No fim, quando se

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



termina a configuração, o próprio SDR designa um endereço multicast que não esteja sendo usado naquele momento e o reserva para esta transmissão.

Nos *hosts* aparecerá o nome da *Session* e as informações para recebê-la. Caso seja de interesse do usuário ele poderá receber esta transmissão bastando somente clicar nesta propaganda que o SDR se encarregará de iniciar o programa apropriado e configurá-lo para o endereço multicast no qual está ocorrendo a transmissão.

Após se configurar todos os parâmetros e pressionar-se o botão de *Accept*, estas informações serão transmitidas para todas as redes que possuem SDR acionados pois elas estarão requisitando informações deste endereço multicast.

4.5 Live Networks

Live Caster

Fabricante : Live Networks

Aplicação para a transmissão de mp3 em *streamming*.

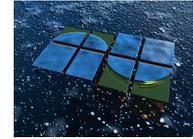
Live Gate

Fabricante : Live Networks

Cria um túnel IP entre um computador dentro do Mbone e o seu computador para que se torne possível a recepção de multicast que ele esteja recebendo.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



5 – Conclusão

Este projeto resultou na montagem de um backbone de IP multicast totalmente funcional além de ter gerado um estudo comparativo prático entre o unicast e o multicast nos parâmetros de tráfego e processamento de servidor. Também foi gerado uma nova versão do program mouted, devido ao fato das versões encontradas na Internet não estarem funcionais e um tutorial da instalação e configuração de um roteador multicast.

Os estudos realizados nesse projeto vêm comprovar as vantagens agregadas ao endereço IP de grupo chamado de multicast. Todos os experimentos realizados comprovaram que para o caso de transmissão de dados para vários usuários, o multicast foi bem superior as transmissões em unicast. As informações apresentadas neste projeto que comparam o tráfego e processamento não deixam dúvidas sobre os benefícios trazidos pelo uso deste tipo de transmissão.

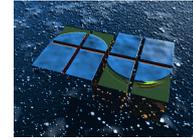
Um ponto importante a ser ressaltado é que ainda se faz necessário uma melhora da qualidade da Internet para a tecnologia ser mais difundida. Os *hosts* necessitam de uma grande capacidade de processamento porque grande parte dos atrasos nas transmissões se deram pelo fato de que as máquinas demoravam um bom tempo codificando e decodificando os dados de voz ou de vídeo. Ocorria uma demora de em média 14s entre a transmissão e a primeira recepção nas transmissões em multicast e uns 21s para as transmissões em unicast.

Analisando-se este atraso de recepção entre as máquinas que não ocorreu de modo lógico, ou seja, as redes mais próximas receberem os dados primeiro, concluiu-se que processamento foi fator determinante para este tipo de resposta.

Por parte dos roteadores é praticamente mandatório que eles sejam capazes de rotear tráfego multicast para um maior aproveitamento da tecnologia, apesar, da possibilidade do datagrama multicast ser tunelado em um datagrama IP normal. Os grandes fabricantes de roteadores estão agregando esta capacidade a todos os seus novos equipamentos e será apenas uma questão de tempo para que grande parte da rede mundial seja composta por roteadores com esta opção.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



O futuro das transmissões ponto multiponto podem ser consideradas como uma certeza, visto que, o desenvolvimento de novos protocolos como por exemplo, o IPv6, que já possui capacidade de transmissões em multicast intrínseca ao protocolo.

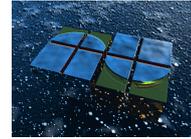
Seguindo esta linha de pensamento pode-se dizer que a primeira grande rede a fazer proveito das facilidades do multicast será a Internet 2 visto que ela possui todos os requisitos acima citados e pelo fato de se uma rede bem nova e ainda em construção. Apesar de não ser abordado com profundidade neste projeto a parte de gerenciamento de grandes redes será muito beneficiada pelas características das transmissões em multicast. Um bom exemplo que poderia ser citado e que já vem sendo implantada em redes Corporativas é o de atualizações e instalações de software. Com o uso de multicast economiza-se tempo e dinheiro pois todas as máquinas podem ser instaladas ao mesmo tempo e através de uma mesma transmissão.

Do ponto de vista do aprendizado este projeto possibilitou uma ótima oportunidade para o estudo mais aprofundado do endereçamento e da estruturação de redes IP e Ethernet, do estudo de protocolos de transporte como o UDP, TCP, da configuração de redes e roteadores, do manuseio de sistema operacional Linux, do estudo de protocolos streaming, sem contar com a própria oportunidade do estudo de uma forma de transmissão que será amplamente usada em um futuro próximo.

O multicast é uma peça importante no desenvolvimento das redes no futuro, diversos avanços têm sido feitos com relação aos protocolos de roteamento como por exemplo o MOSPF, o PIM e o MBGP(*multicast Border Gateway Protocol*) que não foram abordados neste projeto, sua utilização em massa se torna apenas uma questão de tempo para as redes se adequem a está nova tecnologia.

Backbone multicast

Universidade de Brasília – Engenharia Elétrica

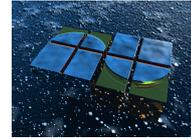


Bibliografia

- [1] Douglas E. Comer. “Internetworking with TCP/IP – Vol 1 – Principles, Protocols and Architecture” Quarta edição. Publicado por Prentice Hall, 1999
- [2] Marcus Goncalves & Kitty Niles. “IP Multicasting Concepts and Applications” primeira edição. Publicado por McGraw-Hill, 1999
- [3] C. Kenneth Miller. “Multicast Networking and Applications” primeira edição. Publicado por Addison Wesley, 1998

Backbone multicast

Universidade de Brasília – Engenharia Elétrica



Referências Eletrônicas

[1] <http://www.ipmulticast.com>

[2] <http://www.rnp.br>

[3] <http://http://www.mice.cs.ucl.ac.uk/>

[4] <http://www.live.com>

[5] <http://www.real.com>

[6] <http://http://www.linuxdoc.org/HOWTO/multicast-HOWTO.html>

[7] <http://metalab.unc.edu/pub/Linux/docs/HOWTO/multicast-HOWTO>