GTA / UFRJ
GRUPO DE TELEINFORMÁTICA E AUTOMAÇÃO

Redes de Computadores II
EEL 879

Parte V
Roteamento Multicast na Internet

Luís Henrique M. K. Costa
luish@gta.ufrj.br

Universidade Federal do Rio de Janeiro -PEE/COPPE
P.O. Box 68504 - CEP 21945-970 - Rio de Janeiro - RJ
Brasil - http://www.gta.ufrj.br

Introdução

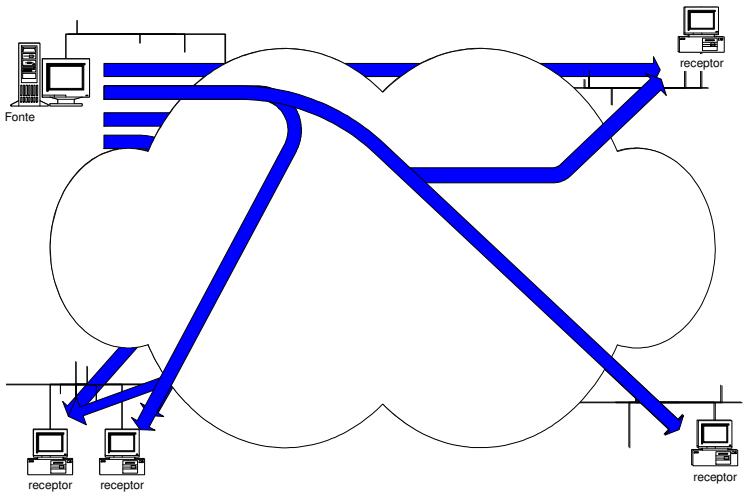
- **Comunicação de grupo (aplicações multi-destinatárias)**
  - Vídeo-conferência
  - Ensino a distância
  - Jogos distribuídos
  - TV na Internet, ...
- **A mesma informação deve ser enviada a múltiplos receptores**

Como enviar a N receptores?

- **Opções: diferentes tipos de transmissão**
- **Unicast**
  - Transmissão ponto-a-ponto
  - 1 emissor, 1 receptor
- **Multicast**
  - Transmissão ponto-a-multiponto
  - 1 emissor, N receptores
- **Broadcast**
  - Envio a todos os nós da rede

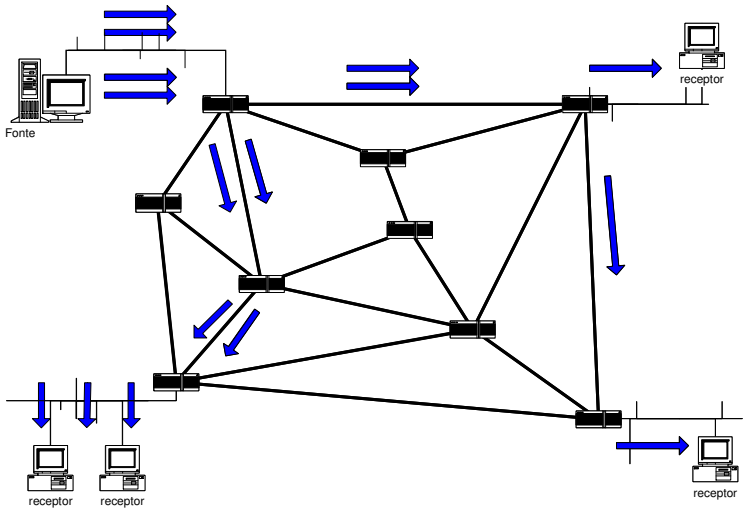
GTA/UFRJ

Unicast x Multicast



GTA/UFRJ

Unicast x Multicast



GTA/UFRJ

Utilização do Multicast

- **Vantagens**
  - Produz menos pacotes
    - Utilização eficiente da banda passante da rede
    - Menor processamento em estações e roteadores

GTA/UFRJ

Utilização do Multicast

- **Problemas**
  - Como identificar o grupo?
    - Lista dos receptores
      - Overhead de cabeçalho limita o tamanho do grupo
    - Endereço de grupo
      - Identidade e número dos receptores desconhecidos
  - Como realizar a distribuição dos pacotes?
    - Endereçamento e roteamento (encaminhamento dos pacotes) são **diretamente** relacionados

GTA/UFRJ

Endereçamento na Internet

- **endereço IP = inteiro de 32 bits**
  - escrito na forma de 4 números decimais separados por pontos: `146.164.69.2`
  - o mapeamento de nomes em endereços IP e vice-versa é feito pelo Sistema de Nome de Domínio (DNS)
  - atribuído a cada interface de rede de uma máquina
    - identifica a conexão de uma estação na rede
- **endereçamento IP**
  - topológico (ou **hierárquico**: utiliza **prefixos**)
    - a posição de uma máquina determina seu endereço
    - torna eficaz as operações de roteamento

GTA/UFRJ

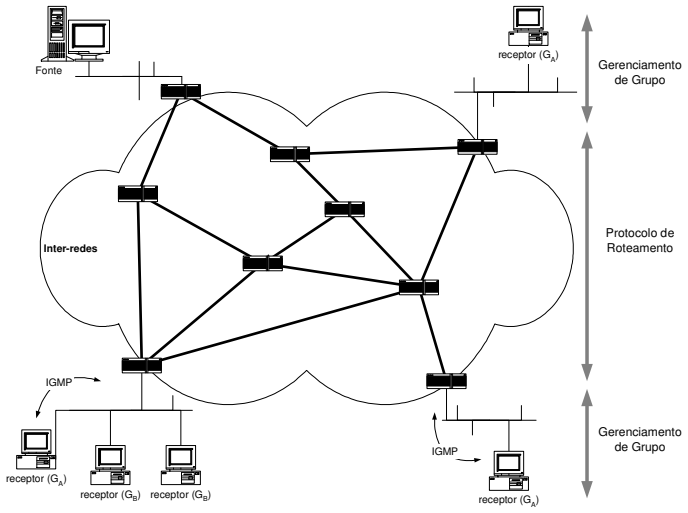
Problema do Multicast

- **Dado o endereçamento, como realizar a distribuição dos pacotes?**
  - Endereço unicast
    - Identifica e localiza uma estação
  - Endereço de grupo
    - Hierarquia impossível, receptores espalhados em toda a rede

Modelo de Serviço IP Multicast

- **Identificação**
  - Endereço de grupo
- **Distribuição dos dados**
  - **Gerenciamento de grupo**
    - Entrada / saída do grupo
      - "quero escutar o grupo" / "quero parar de escutar o grupo"
    - Entre a estação e seu roteador local
  - **Protocolos de roteamento**
    - Distribuição dos dados entre as redes
      - Como fazer os pacotes chegarem ao meu roteador local?

Modelo de Serviço

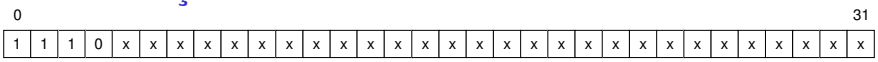


Modelo de Serviço

- **Grupo**
  - **Identificado por um endereço de grupo**
  - **Conversação N x M, aberta**
    - Qualquer estação pode participar
    - Uma estação pode pertencer a vários grupos
    - Uma fonte pode enviar dados ao grupo, tendo se inscrito neste ou não
  - **O grupo é dinâmico**, uma estação pode entrar e sair a qualquer instante
  - O **número e a identidade** dos participantes do grupo são **desconhecidos**

Endereçamento

- **Endereço Multicast = IP Classe D**



- `224.0.0.0` **a** `239.255.255.255 (224.0.0.0/8)`
- **Em geral, o endereço é temporário, *mas*...**
  - `224.0.0.0` a `224.0.0.255` são reservados e de escopo local

224.0.0.1	**All Hosts**
224.0.0.2	**All Multicast Routers**
224.0.0.3	**Não alocado**
224.0.0.4	**All DVMRP Routers**
224.0.0.5	**All OSPF Routers**

Modelo de Serviço

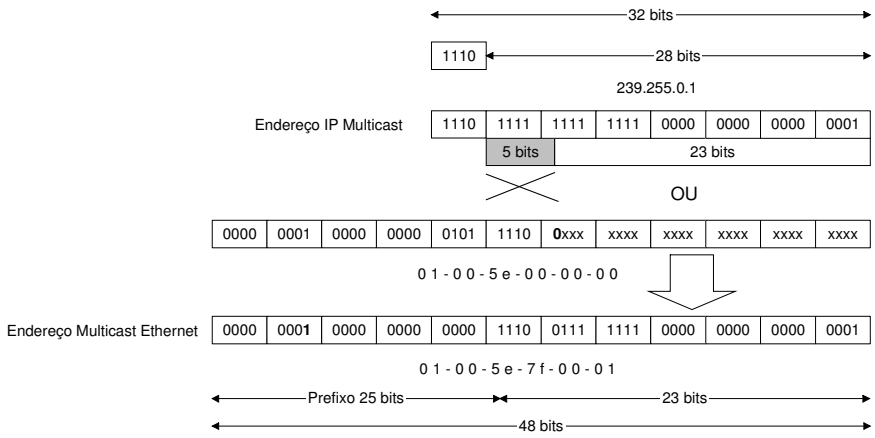
- **O grupo é identificado por um endereço IP Multicast**
  - Endereço IP Classe D
- **Criação do grupo**
  - Escolha de um **endereço multicast** e envio de dados para o grupo
- **Destruição do grupo**
  - Parada do envio de dados

Conexão a um Grupo Multicast

- **A aplicação sinaliza à camada rede interesse no grupo `G`**
  - `socket`
- **Se não havia outra aplicação conectada a `G`**
  - Relatório IGMP é enviado na rede local
  - Camadas inferiores podem ser igualmente programadas
    - Ex. Ethernet

GTA/UFRJ

Multicast Ethernet



- **28 bits IP são mapeados em 23 bits Ethernet**
  - 32 endereço IP multicast = 1 endereço multicast Ethernet

GTA/UFRJ

Por que apenas 23 bits?

- **No início da década de 90, Steve Deering desejava que o IEEE alocasse 16 OUIs (*Organizational Unique Identifier*) para os endereços multicast Ethernet.**
- **Cada OUI equivale a 24 bits de espaço de endereçamento**
  - 16 OUIs consecutivos = 28 bits
- **Na época, 1 OUI = US$ 1.000,00**
- **Jon Postel (chefe de Deering na época) comprou apenas 1 OUI, e liberou apenas a metade do espaço para as pesquisas de Deering...**

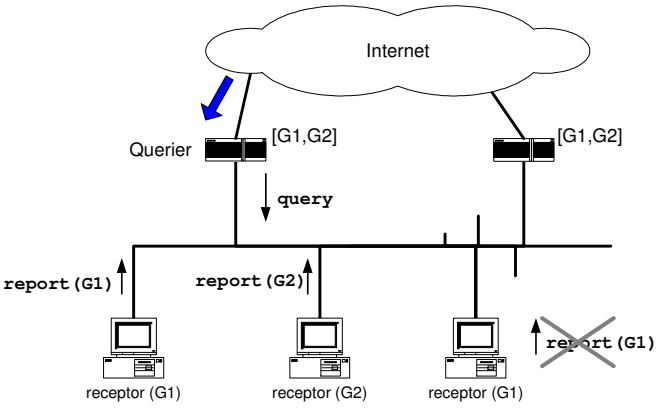
Gerenciamento de Grupo

- **Quem quer ouvir que grupos?**
  - "estação de rádio"
- **IGMP (*Internet Group Management Protocol*)**
  - Detecção de estações interessadas em grupos multicast
  - Existem 4 versões do IGMP
- **Escopo local**
  - diálogo entre a estação e o primeiro roteador
  - criação da árvore de distribuição independente do IGMP

Funcionamento do IGMP

- **Parte estação**
  - Conexão ao grupo (`join(G)`)
    - Receptor envia mensagem `report(G)`
  - Envio de mensagens `report` em resposta às mensagens `query`
    - "Estes são os grupos de interesse desta estação"
- **Parte roteador**
  - Envio periódico de mensagens `query`
    - "Que grupos são escutados na rede?"
- **Parte estação**
  - Mecanismo de supressão de mensagens `report`

Funcionamento do IGMP



IGMPv2

- **Introduz o mecanismo de *fast-leave***
  - Diminuição da latência de desconexão
- **Desconexão**
  - Receptor envia mensagem IGMP **`leave(G)`**
- **Regras de processamento para evitar a desconexão de outras estações**
  - Ex. roteador deve enviar **`query(G)`** para detectar se existem ouvintes remanescentes

IGMPv3

- **Filtragem de fontes**
- **A estação anuncia o interesse no grupo `G` ,**
  - "apenas nos dados enviados por determinadas fontes", ou
  - "nos dados enviados por todas, exceto determinadas fontes"
- **Interface**
  - **`IPMulticastListen (socket, interface,`**
    `mcast-address,`
    `filter mode,`
    `source-list)`
  - **`filter-mode`** **pode ser** **`INCLUDE`** **ou** **`EXCLUDE`**

Exemplo no IGMPv3

- **Recepção do que apenas as fontes S1 e S2 enviam a G**
  - `IPMulticastListen (sock, iface, G, INCLUDE, {S1,S2})`
- **Recepção de tudo que é enviado a G, exceto por S2 e S3**
  - `IPMulticastListen (sock, iface, G, EXCLUDE, {S2,S3})`
- **Estado no roteador**
  - `(G,EXCLUDE{S3})`

Roteamento Multicast

- **Problema de Roteamento Multicast**
- **G= (V, E)**
  - **V** conjunto de vértices
  - **E** conjunto de enlaces
- **M sub-conjunto de V**
  - inclui fontes e receptores do grupo multicast
- **Problema: construir uma, ou várias, topologias de interconexão, árvores, que incluem todos os nós em M**
  - árvore por fonte (*source-based tree*)
  - árvore compartilhada (*shared tree*)

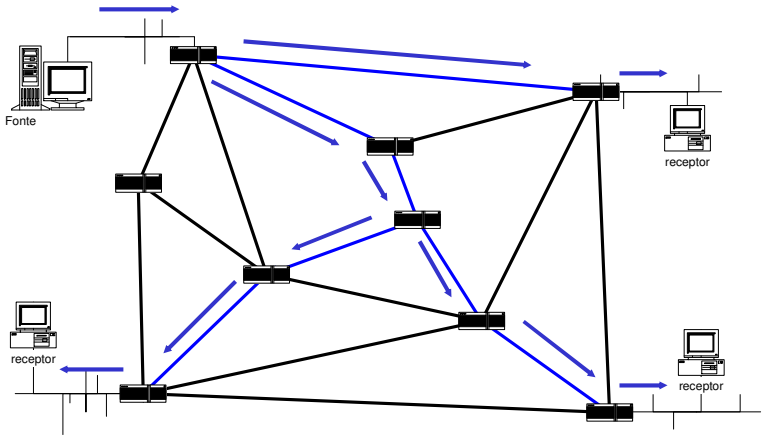
Primeiras Soluções

- **Árvores de cobertura (*spanning trees*)**
- **Algoritmo de inundação**
- **Árvores RPF (*Reverse Path Forwarding*)**
- **Árvores centradas**

Árvores de Cobertura

- **Sub-grafo contendo todos os nós em M, sem ciclos**
- **Pode-se adicionar objetivo de custo mínimo**
  - Associa-se um custo, c_{uv}, a cada enlace (u,v)
- **Se $c_{uv} = 1 \ \forall u,v$, árvore de Steiner**
  - Problema NP-completo

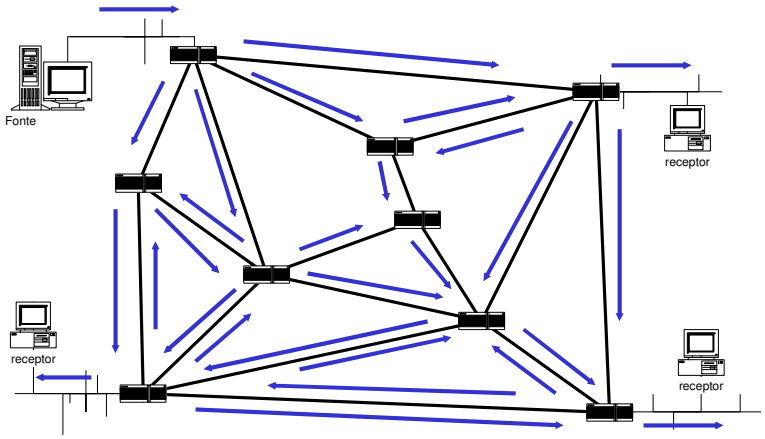
Árvores de Cobertura



Inundação

- **Ao receber o pacote**
  - Esta é a primeira vez que foi recebido?
    - Se sim, re-envio em todas as interfaces de saida
    - Se não, descarte
- **Problema**
  - Como identificar o primeiro envio de um pacote
    - Armazenar identificação
    - Carregar lista dos nós atravessados
  - Consumo de memória e banda passante

Inundação



GTA/UFRJ

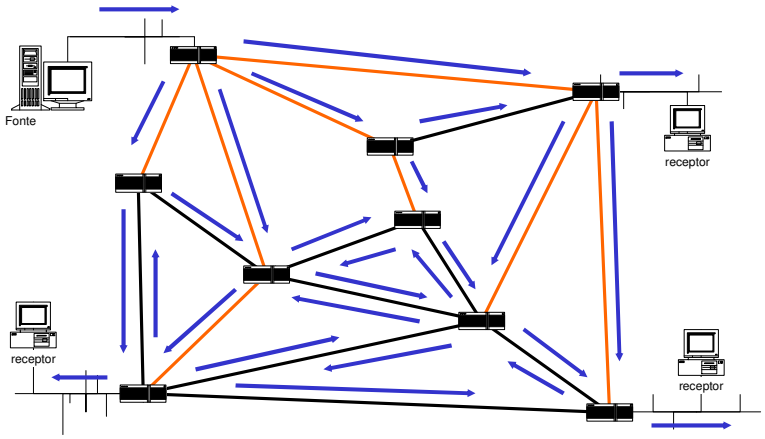
Árvores RPF

- Hipótese: um roteador R conhece o caminho mais curto para ir à fonte, S
- *Reverse Path Forwarding check (RPF check)*

- Reverse Path Broadcasting
  - O roteador R recebe um pacote da fonte S
    - O pacote chegou pela interface utilizada por R para ir à S? (RPF check)
    - Se sim, enviar o pacote por todas as interfaces de saída.
    - Se não, descartar o pacote.

GTA/UFRJ

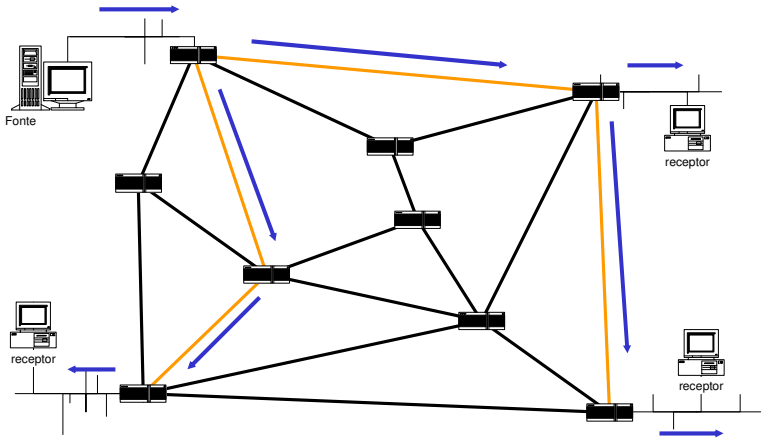
Reverse Path Broadcasting



Reverse Path Forwarding

- **Hipótese**
  - um roteador **R** sabe se seu vizinho o utiliza como caminho para a fonte, **S**
- **Como obter esta informação**
  - trivial, se protocolo de estado do enlace
  - se protocolo de vetor-distância
    - mensagem adicional para alertar o roteador “pai”, ou
    - mensagem de poda para eliminar a rota reversamente
- **Informação por (fonte,grupo)**

Árvore RPF



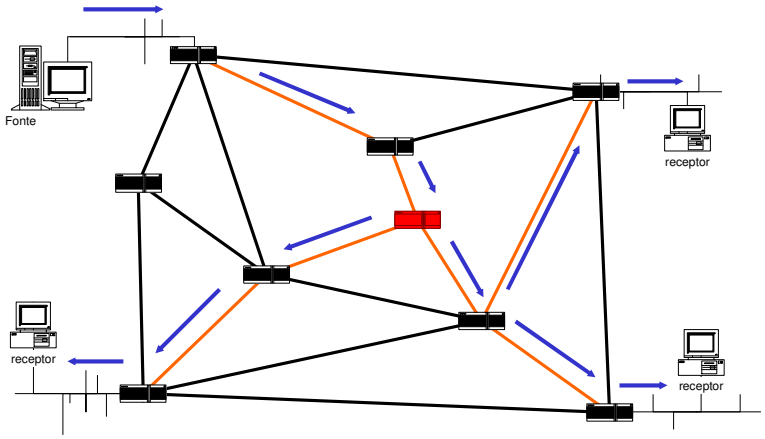
GTA/UFRJ

Árvores Centradas

- **Construída a partir de um nó central (*core*)**
- **Compartilhada por diversas fontes**
  - diversas fontes utilizam o mesmo *core*
  - "pedidos de conexão" são enviados ao *core*

GTA/UFRJ

Árvores Centradas



GTA/UFRJ

Roteamento Multicast Intra-domínio

- **DVMRP (*Distance Vector Multicast Routing Protocol*)**
  - Primeiro protocolo utilizado no MBone
- **MOSPF (*Multicast Open Shortest Path First*)**
- **CBT (*Core Based Trees*)**
- **PIM (*Protocol Independent Multicast*)**
  - PIM-DM (PIM *Dense-Mode*)
  - PIM-SM (PIM *Sparse Mode*)
  - PIM-SSM (PIM Source Specifiic Multicast)

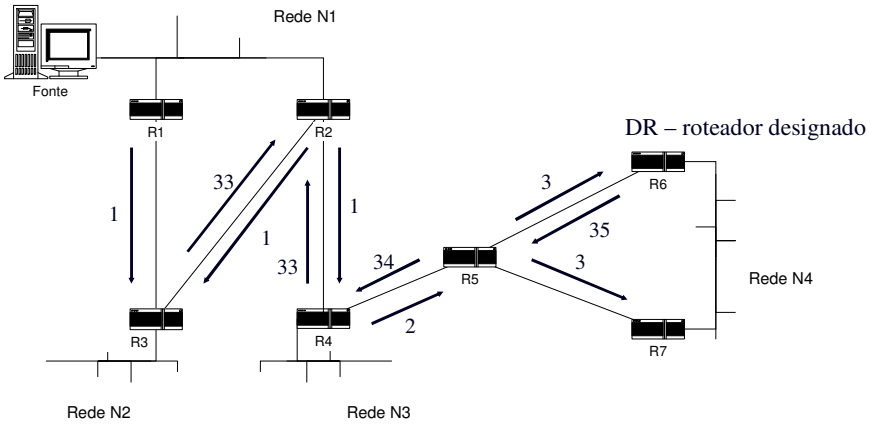
GTA/UFRJ

DVMRP

- **Utiliza vetores de distância**
  - Semelhante ao RIP (*Route Information Protocol*)
  - Constrói rotas ***unicast*** para cada fonte multicast
  - *Poison-reverse* especial utilizado para marcar interfaces filhas
- **Distribuição de dados**
  - Inundação e poda (*flood-and-prune*)
  - Teste RPF baseado em sua tabela de roteamento unicast
- **A inundação é periódica**
  - Descoberta de fontes ativas

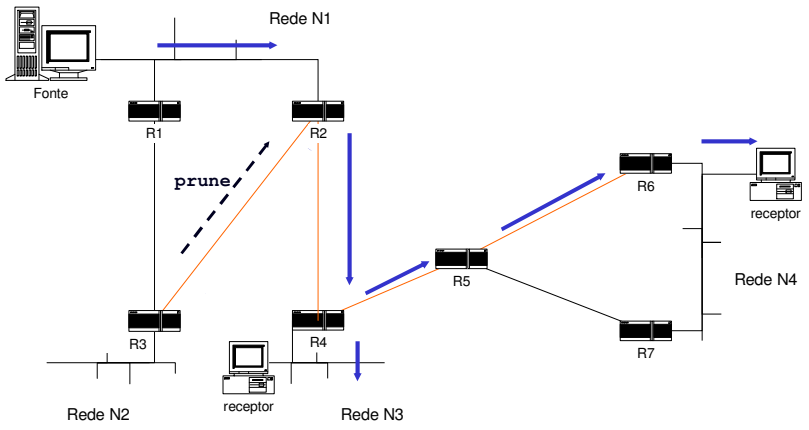
GTA/UFRJ

Funcionamento do DVMRP



GTA/UFRJ

Envio de Dados no DVMRP



GTA/UFRJ

DVMRP

- **Algoritmo simples**
- **Protocolo de roteamento unicast *próprio***
- **Inundação periódica da rede com *dados***
- **Vetores-de-distância**
  - Convergência lenta, como no RIP

GTA/UFRJ

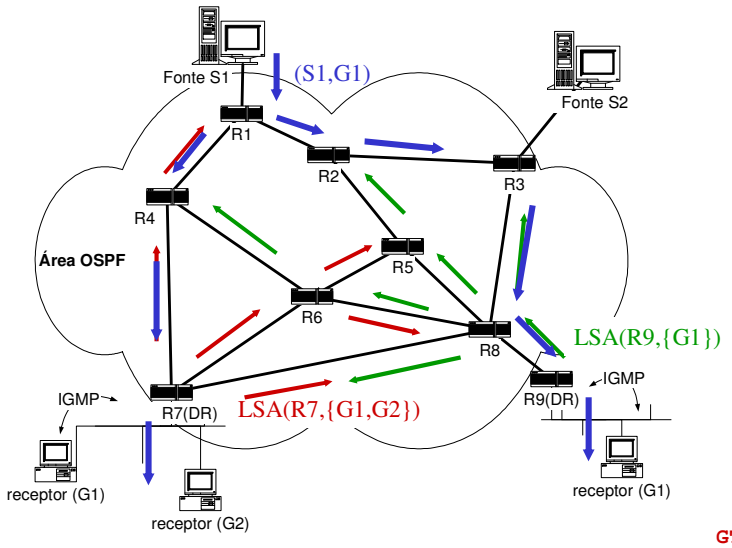
MOSPF

- **Extensão do OSPF (*Open Shortest Path First*)**
  - roteadores trocam mensagens de estado-do-enlace
    - LSA – *Link State Advertisement*
  - Cada nó possui a topologia atualizada da rede
  - Algoritmo de Dijkstra – caminhos mais curtos
- **Novo tipo de LSA anuncia receptores multicast**
- **A árvore de distribuição é uma SPT (*Shortest-Path Tree*)**
  - união dos caminhos mais curtos entre fonte e cada receptor

MOSPF

- **Estrutura hierárquica**
  - Áreas OSPF (roteamento intra-área e inter-área)
- **Intra-área**
  - IGMP – descoberta de receptores
  - *Group Membership* LSAs
    - (roteador, grupo multicast, lista de interfaces)
- **Cálculo da SPT**
  - Disparado apenas após recepção do primeiro pacote de dados
  - Diminui o custo computacional

MOSPF Intra-área



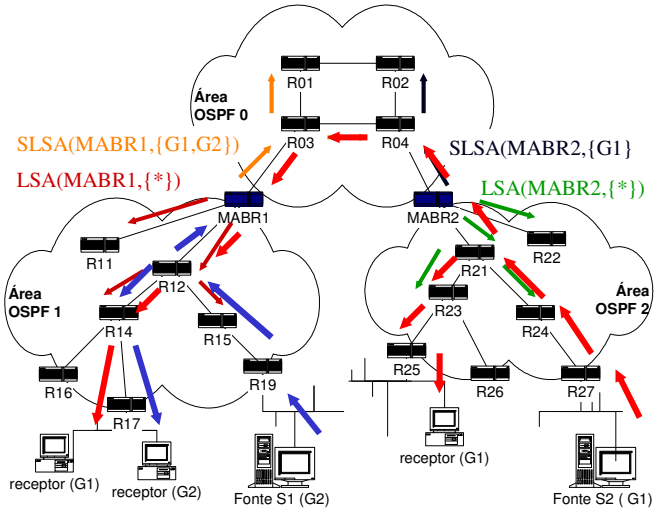
GTA/UFRJ

MOSPF Inter-área

- ***Multicast Area Border Router* (MABR)**
  - Envio de tráfego multicast
  - Informação sobre os grupos multicast
  - Conecta uma área OSPF à área 0 (área *backbone*)
- **Receptor coringa**
  - LSA anuncia que o roteador possui receptores para *todos* os grupos
  - Todos os MABRs em uma área são receptores coringa
    - Injetam LSAs coringa na área OSPF
    - Recebem todo o tráfego e o re-enviam na área 0 se necessário
- **LSA de Resumo de Grupos (*Summary Membership* LSA)**
  - Lista todos os grupos escutados em uma área
  - São injetadas na área 0 pelos MABRs

GTA/UFRJ

MOSPF Inter-área



MOSPF Inter-área

- **Árvore SPT é construída na área 0**
- **A árvore completa (áreas comuns + área 0) não é SPT**
- **Pode haver envio desnecessário de tráfego ao MABR**
  - Receptor coringa

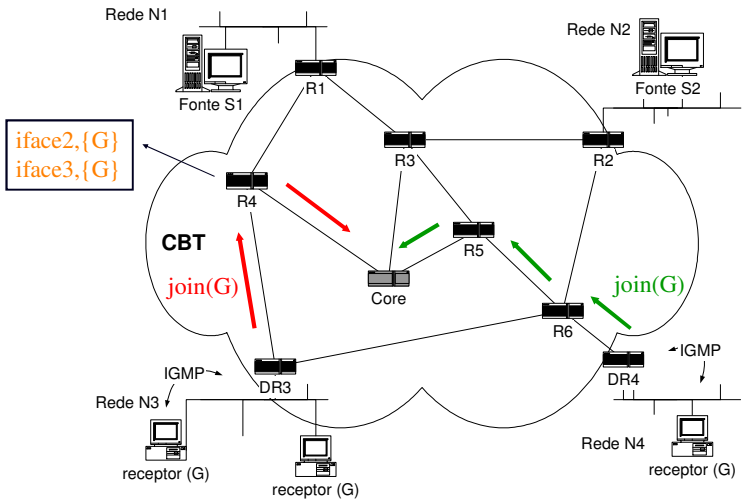
MOSPF

- **Protocolo de roteamento unicast deve ser OSPFv2**
- **Mensagens de estado-do-enlace**
  - evitam a inundação periódica de dados como no DVMRP
  - porém impedem o uso do OSPF em redes muito grandes
    - LSAs inundam toda a rede
- **DVMRP**
  - Dados são uma mensagem **implícita** sobre a localização dos receptores
- **MOSPF**
  - Mensagem **explícita** sobre onde existem receptores

CBT

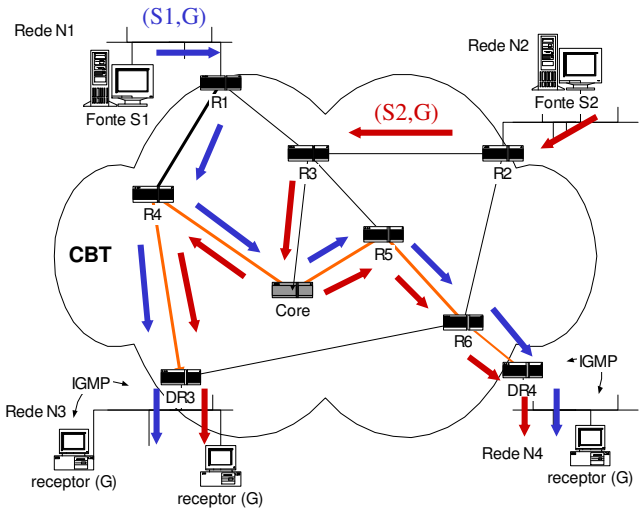
- **Utiliza árvores centradas**
  - Compartilhadas e bi-direcionais
- **Roteador central – *core***
- **Construção da árvore**
  - Mensagens *join*
    - Enviadas pelos receptores na direção do ***core***

Construção da Árvore CBT



GTA/UFRJ

Envio de Dados no CBT



GTA/UFRJ

CBT

- **Escalabilidade**
  - Estado apenas nos roteadores na árvore de distribuição
    - Ao contrário de DVMRP e MOSPF
  - Estado por (grupo), em vez de por (fonte,grupo)
- **Desvantagens**
  - Concentração de tráfego próximo ao *core*
  - Rotas sub-ótimas entre a fonte e o receptor
    - Maiores atrasos
- **Localização do *core* é crítica**

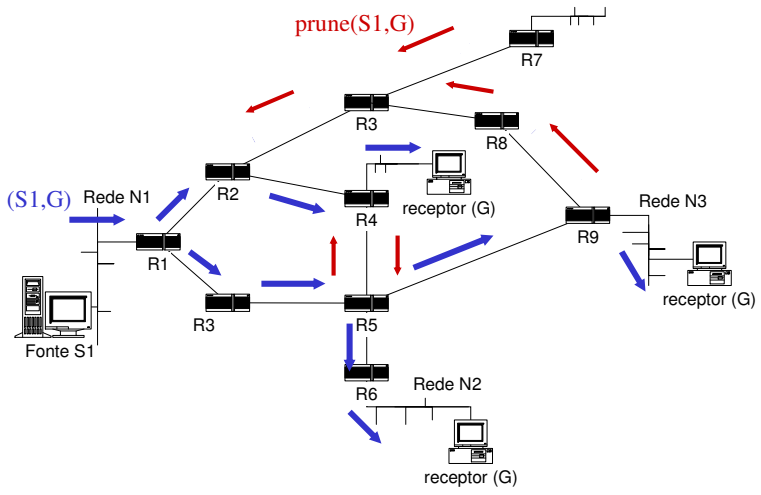
PIM

- ***Protocol Independent Multicast* (PIM)**
  - Independente do protocolo de roteamento unicast
- ***Dense-Mode* (PIM-DM)**
  - Receptores densamente distribuídos
  - Árvores por fonte
  - Inundação-e-poda (semelhante ao DVMRP)
- ***Sparse-Mode* (PIM-SM)**
  - Receptores esparsamente distribuídos na rede
  - Árvores compartilhadas (como o CBT)
    - Uni-direcionais

PIM-DM

- ***Reverse Path Multicast***
  - Utiliza o teste RPF
  - Mas não constrói lista de interfaces filhas como o DVMRP
  - Tráfego enviado em todas as interfaces de saída
  - Duplicação de pacotes, todos os enlaces da rede são utilizados, mas
    - independência do roteamento unicast
    - evita base de dados com pais/filhos
  - Após a inundação inicial, mensagens de poda são enviadas
    - Por roteadores que não possuem receptores do grupo
    - Por roteadores que não possuem vizinhos interessados no grupo
    - Por roteadores que receberam tráfego por uma interface incorreta (RPF)

PIM-DM



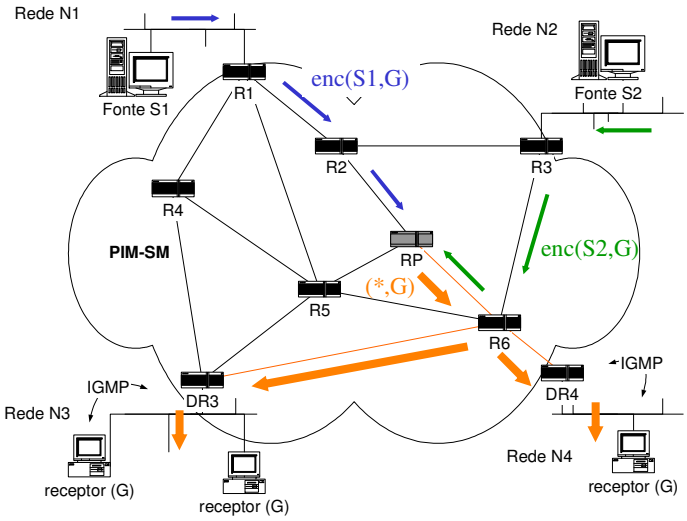
PIM-DM

- **Árvore SPT reversa (RSPT)**
  - União dos caminhos mais curtos dos receptores até a fonte
- **Todos os roteadores da rede armazenam estado (fonte,grupo) para todas as fontes/grupos ativos**
- **Inundação periódica é necessária**
  - Descoberta de novos membros do grupo

PIM-SM

- **Árvores de distribuição centradas (`(*,G)` , como o CBT)**
  - Nó central – roteador RP (*rendez-vous point*)
  - Uni-direcional
- **Construção da árvore**
  - Mensagens *join*
- **Mecanismo de mapeamento entre grupos e RPs**
- **Fontes se "registram" com o RP**
  - Dados são enviados ao RP (encapsulados em mensagens **`PIM-register`**)

Árvore Compartilhada no PIM-SM



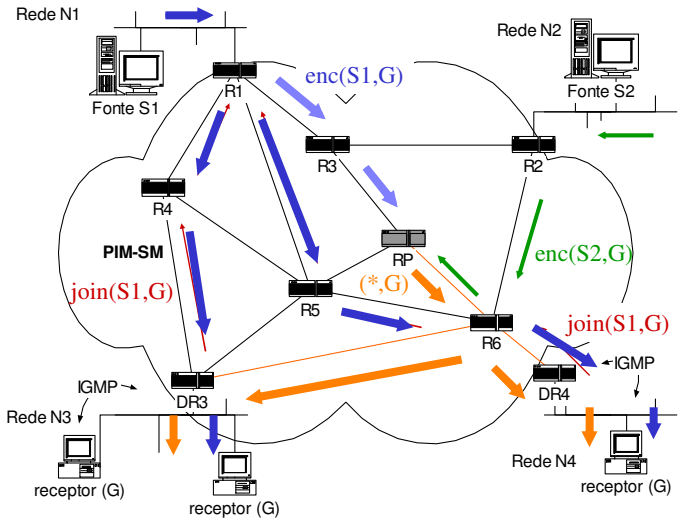
GTA/UFRJ

PIM-SM

- **Árvores por fonte `(S,G)`**
- **Troca realizada por configuração**
  - Taxa de envio de dados
- **Roteador local envia mensagens `join(S,G)`**
  - Mas não pára o envio de join(*,G)
    - Tráfego de outras fontes deve continuar
  - Envia mensagem de poda especial (`RP-bit-prune(S,G)`)
    - Evita a recepção de dados de `S` em duplicata

GTA/UFRJ

Árvore por Fonte no PIM-SM



PIM-SM

- **RP também pode enviar `join(S,G)`**
- **Possibilidade de árvores por fonte**
  - Diminui a importância da localização do RP
  - Reduz o atraso fonte-receptores

Outros Problemas do Modelo de Serviço

- **Como limitar o alcance (ou escopo) do tráfego multicast**
  - Até onde vai o tráfego enviado por uma fonte?
    - (receptores **não** são conhecidos)
- **Como evitar a colisão de endereços**
  - Duas aplicações escolhem o mesmo endereço multicast

Alcance do Tráfego Multicast

- **Definição de Escopos**
- **Por endereço**
- **Utilizando o campo TTL**
- **Administrativos**

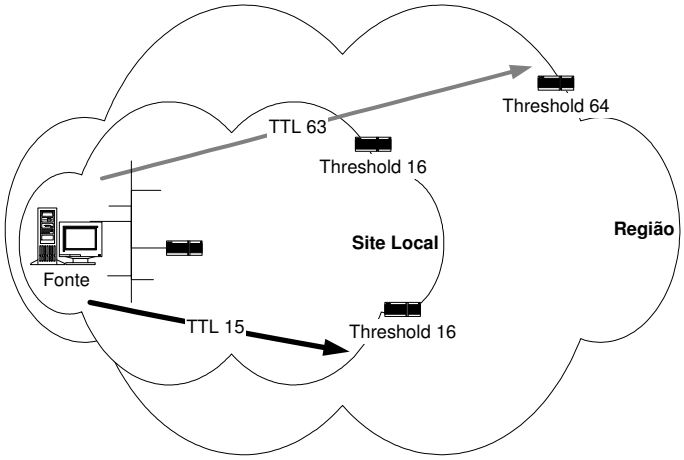
Escopo por Endereço

- **Faixa de endereços dinâmicos**
  - `224.0.1.0` a `239.255.255.255`
  - `224.0.1.0` a `238.255.255.255`
    - aplicações com escopo global
  - `239.0.0.0` a `239.255.255.255`
    - aplicações com escopo limitado
    - `239.253.0.0/16` – local ao site
    - `239.192.0.0/14` – local à organização

Escopo usando o TTL

- **TTL (*Time-to-live*)**
  - Campo decrementado de 1 a cada roteador atravessado
  - Pacote descartado quando TTL=0
- **Escopo usando o TTL**
  - Escolhe-se um valor de TTL inicial para os pacotes multicast
- **Limita-se a distância em número de saltos**
  - Pouca correlação entre numero de saltos e uma região
- **Limiar TTL (TTL *threshold*)**
  - Configurado nos roteadores de borda
  - Pacotes com TTL menor que o limiar de TTL são descartados

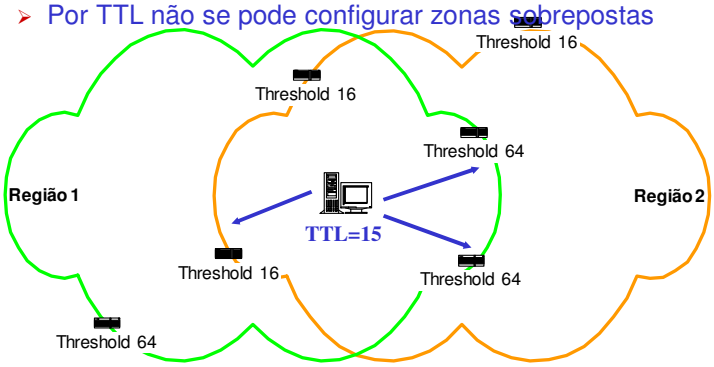
Escopo usando o TTL



GTA/UFRJ

Escopos Administrativos

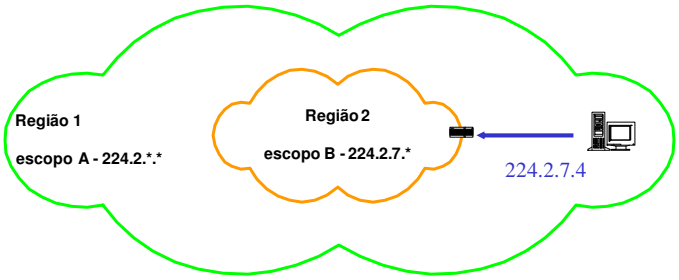
- **Roteadores não encaminham certas faixas de endereços**
  - Maior flexibilidade que por TTL
  - Por TTL não se pode configurar zonas sobrepostas



GTA/UFRJ

Escopos Administrativos

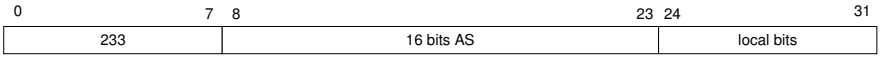
- **Desvantagens**
  - Alcance definido por ***todas*** as zonas às quais a fonte pertence
    - Como descobrir que zonas se aplicam?
  - Zonas sobrepostas devem utilizar faixas de endereços disjuntas



  - Erros de configuração
    - Zonas maiores ou menores que o necessário
    - Com o TTL, pode-se escolher um valor pouco maior que o necessário e garantir o funcionamento da aplicação

Alocação de Endereços

- **Alocação Estática**
  - Endereçamento GLOP [RFC2770]
  - Faixa `233/8` reservada



  - Ex. AS 16007 - faixa `233.64.7.0` à `233.64.7.255`

- **Alocação Dinâmica Hierárquica**
  - Arquitetura MAAA (*Multicast Address Allocation Architecture*)

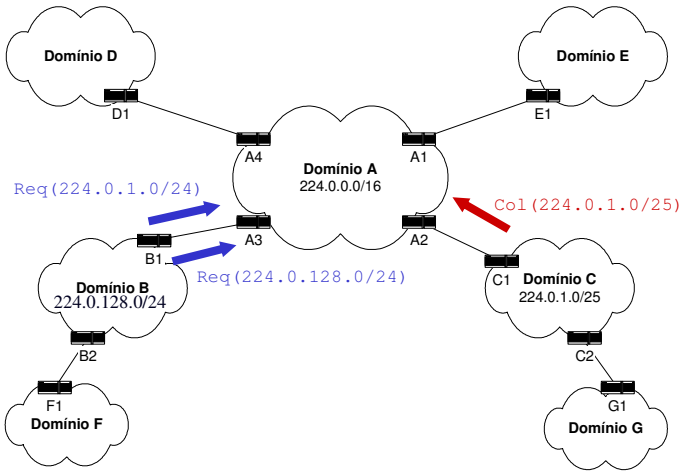
Arquitetura MAAA

- **MADCAP (*Multicast Address Dynamic Allocation Protocol*)**
  - Protocolo cliente-servidor (semelhante ao DHCP)
  - Serviço de alocação de endereços
- **Multicast AAP (*Multicast Address Allocation Protocol*)**
  - Coordena a alocação de endereços dentro de um domínio
  - Executado pelos servidores MADCAP
- **MASC (*Multicast Address Set Claim*)**
  - Coordena a alocação de endereços inter-domínio
  - Trabalha com o BGP

Princípios Básicos do MASC

- **Estrutura hierárquica**
  - Domínios = Sistemas Autônomos (AS)
  - Trabalha em conjunto com o BGP
  - Domínios-"filhos" alocam sub-faixas das faixas alocadas por seus "pais"
- **Mecanismo de escuta e pedido com detecção de colisões**
  - Filho escuta as faixas alocadas por seu pai,
  - escolhe sub-faixas,
  - anuncia as sub-faixas escolhidas aos irmãos.
  - Faixa considerada alocada após um período de detecção de colisões,
  - e comunicada ao servidor MAAS do domínio e a outros domínios
    - Através de rotas de grupo ("group routes") BGP.

Alocação Hierárquica



Rotas de Grupo BGP

- **Rotas de grupo**
  - G-RIB ("Group-Route Information Base")
- **A3 armazena (224.0.128.0/24, B1) em sua G-RIB**
  - B1 é o próximo salto para os grupos dentro da faixa 224.0.128.0/24
- **A1, A2 e A4 armazenam (224.0.128.0/24, A3) em suas G-RIBs**
  - A3 é o próximo salto a partir de A1, A2 e A4

Agregação de Rotas

- **Semelhante às rotas unicast no BGP**
- **Exemplo**
  - Domínio A – 224.0.0.0/16
  - Domínio B – 224.0.128.0/24 (anunciada por B1)
- **A1 anuncia a rota (224.0.0.0/16, A1) ao roteador E1**

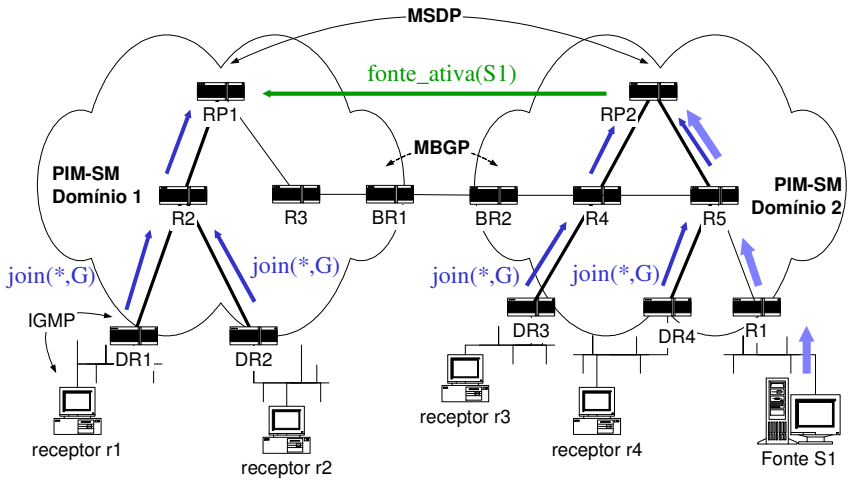
Roteamento Inter-domínio

- **Nem todos os roteadores são multicast**
- **Diferentes protocolos nos diferentes domínios**
- **Problemas com o PIM-SM**
  - Mecanismo escalável de mapeamento entre RPs e grupos
  - Inter-dependência entre provedores de serviço introduzida pelos RPs

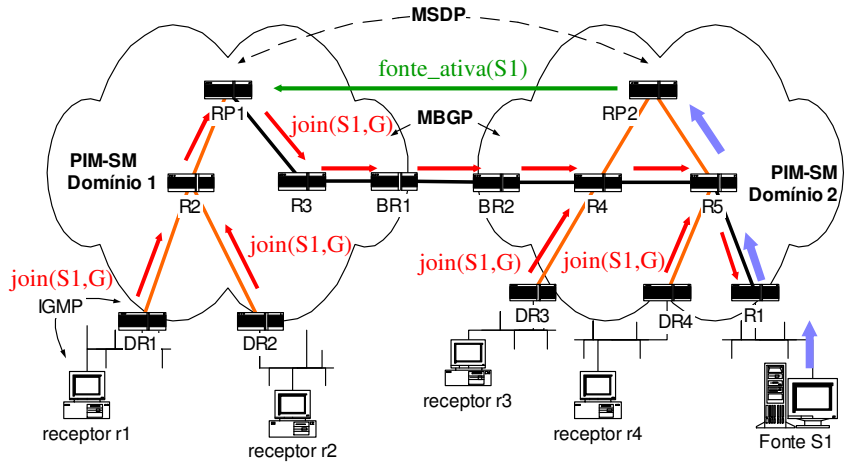
Arquitetura MBGP/MSDP

- **Solução de curto-prazo**
  - Interconexão de domínios PIM-SM
- **MBGP – *Multiprotocol Extensions for BGP-4***
  - Permite múltiplas tabelas de roteamento
    - Pode-se utilizar uma tabela unicast e uma tabela multicast
    - M-RIB (*Multicast – Route Information Base*)
- **MSDP – *Multicast Source Discovery Protocol***
  - Anúncio das fontes ativas, entre todos os RPs

Árvores Intra-domínio no MBGP/MSDP

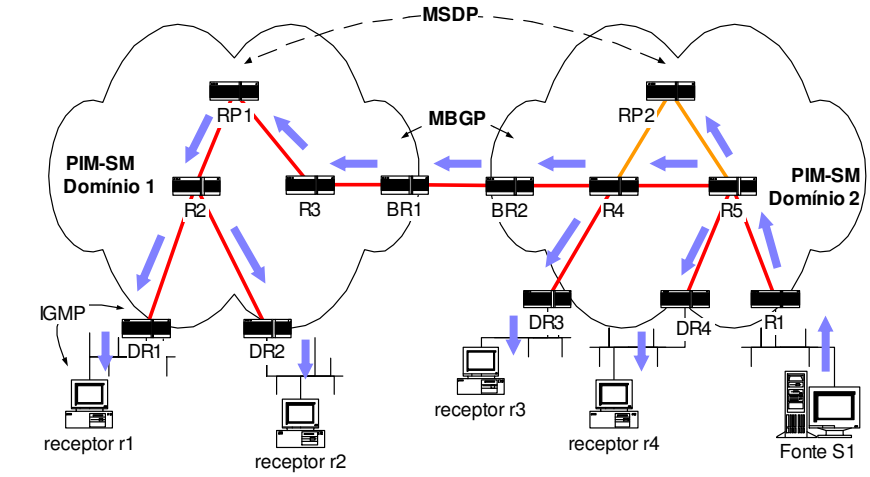


Árvore Inter-domínio no MBGP/MSDP



GTA/UFRJ

Envio de Dados no MBGP/MSDP



GTA/UFRJ

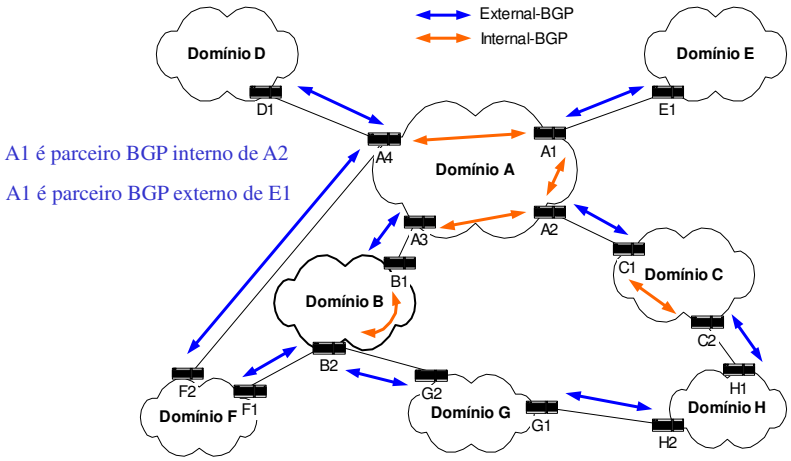
MBGP/MSDP

- **Inter-dependência entre domínios evitada**
- ***Todos* os domínios são notificados de *todas* as fontes ativas**
  - Problema de escalabilidade
- **Tráfego é encapsulado nas mensagens de "fonte-ativa"**
  - Evita perda dos primeiros dados
  - E de fontes em rajadas
  - Problema: dados são enviados a *todos* os RPs

Inter-domínio: Próximo Passo

- ***Border Gateway Multicast Protocol* (BGMP) – (RFC 3913)**
- **Projeto semelhante ao BGP**
  - "Anuncio as rotas que me interessam anunciar"
  - "Sou a raiz dos grupos que me pertencem"

BGP – Visão Geral



GTA/UFRJ

Border Gateway Multicast Protocol

- **Árvores compartilhadas bi-direcionais**
  - Podem ser construídos ramos por fonte
- **A raiz da árvore é um Sistema Autônomo (AS)**
  - Maior estabilidade e tolerância a falhas
  - ASs devem ser associados a endereços de grupo multicast
- **A raiz da árvore do grupo G é o AS ao qual G está associado**
  - Maior probabilidade de este AS possuir receptores de G

GTA/UFRJ

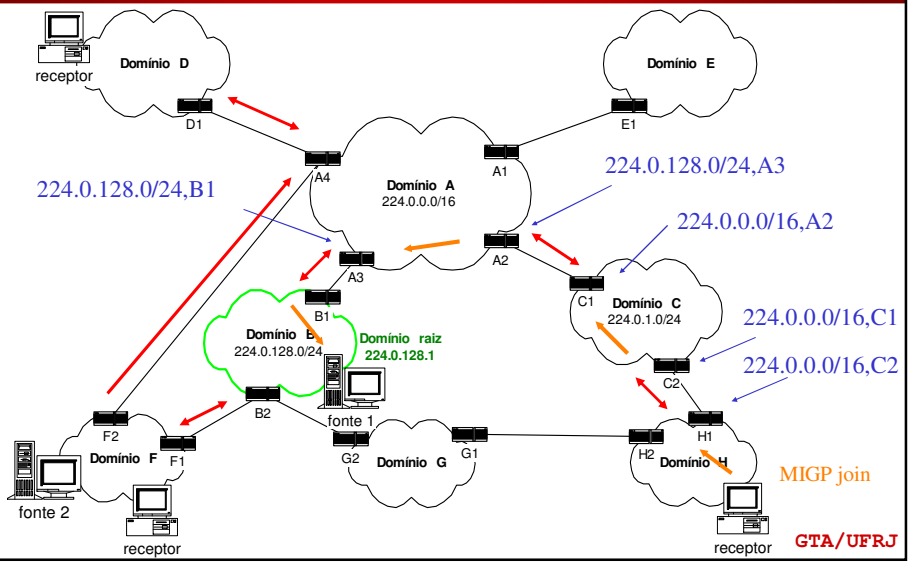
BGMP

- **Supõe mecanismo de associação de endereços**
  - Alocação de faixas pelo MASC
  - Alocação estática GLOP
- **Roteadores de borda executam *dois* protocolos multicast**
  - BGMP
  - MIGP (*Multicast Interior Gateway Protocol*)
    - Ex. PIM-SM, DVMRP

Funcionamento do BGMP

- **Ao receber mensagens `join`, o roteador de borda**
  - Cria um "alvo-pai" – próximo roteador BGMP na direção do AS raiz
  - Cria uma lista de "alvos-filhos" – outro roteador BGMP ou MIGP
  - Propaga o `join` a seu alvo-pai
    - Envia `join` ao MIGP, caso o alvo-pai seja um parceiro BGMP ***interno***

BGMP

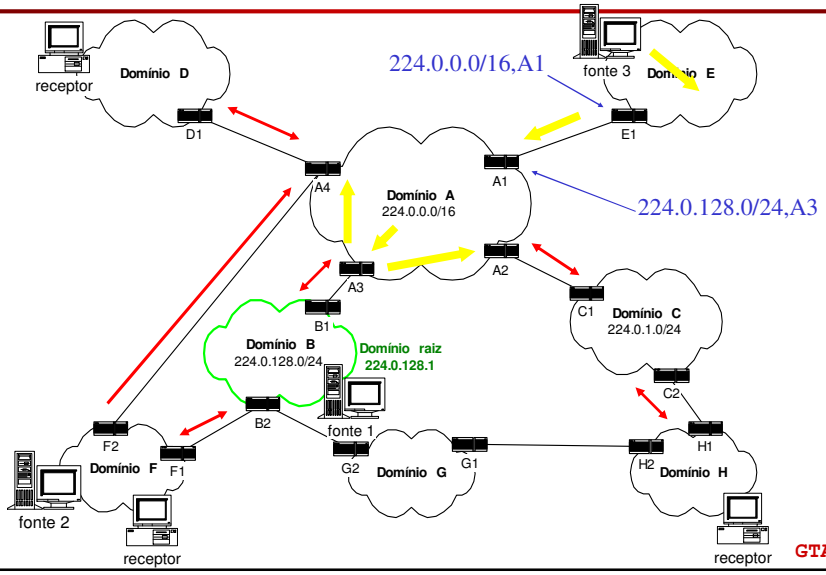


BGMP

- **Modelo de serviço IP Multicast**
  - Fontes que não pertencem ao grupo ***podem enviar ao grupo***
    - Dados encaminhados pelo MIGP até o melhor roteador de saída
      - DVMRP – inundação da rede
      - PIM-SM – envio ao RP (remoto neste caso)
  - Em seguida dados enviados na direção do domínio raiz pelo BGMP

GTA/UFRJ

BGMP



GTA/UFRJ

BGMP

- **Implantação**
  - Na escala da Internet
  - Depende da implantação da arquitetura de alocação de endereços
  - ***Acontece de forma muito lenta...***

GTA/UFRJ

Novas Propostas

- **Modelo de Serviço IP Multicast**
  - Endereço IP class-D = grupo de estações
    - qualquer estação pode se inscrever no grupo
    - e qualquer estação pode enviar dados para o grupo
  - alocação de endereços multicast é problemática
  - protocolos: IGMP + protocolos de roteamento
- **IP Multicast não foi implantado na Internet**
  - Redes de *backbone* superdimensionadas
- **Tentativas de simplificação da arquitetura**
  - Simple Multicast
  - EXPRESS, PIM-SSM
  - REUNITE, HBH

Protocolos Multicast

- **IGMP**
  - Gerenciamento de grupo (estações – roteadores designados)
- **Protocolos de roteamento**
  - Modo denso
    - DVMRP, PIM-DM
      - Inundação-e-poda, árvores por fonte
  - Modo esparso
    - PIM-SM
      - `Join` explícito, árvores compartilhadas, árvores por fonte
- **MBGP (*Multi-protocol BGP*)**
  - Anúncio de rotas unicast e multicast
- **MSDP (*Multicast Source Discovery Protocol*)**
  - Anúncio de fontes ativas entre todos os RPs

Inconvenientes da Arquitetura Atual

- **Modelo de serviço aberto**
- **Alocação de endereços**
- **PIM-SM**
  - é possível comutar da árvore compartilhada para árvore por fonte
  - nos roteadores Cisco
    - limiar de tráfego configurado para 1 pacote
    - RP, MSDP
      - servem apenas para a descoberta de fontes
  - Árvore por fonte é preferível em muitas aplicações
  - Mesmo para fontes conhecidas
    - Construção da árvore compartilhada no início da transmissão

EXPRESS

- **EXPlicitely REquested Single Source multicast**
- ***Canal* multicast**
  - 1 fonte para N receptores
  - ECMP protocol
    - controle do canal
    - coleta de informações sobre o canal
- **Canal**
  - (S,G) - S = endereço IP da fonte, G = endereço multicast classe D

Source Specific Multicast

- **SSM (Source-Specific Multicast)**
  - conversação 1 x N
  - *Subscribe* ***channel*** *<S,G>*

  - Fornece base para o controle de acesso
    - Apenas S pode enviar para (S,G), outras fontes são bloqueadas

  - Alocação de endereços multicast (G)
    - Problema local à fonte

  - Roteadores RP e o protocolo MSDP não são necessários

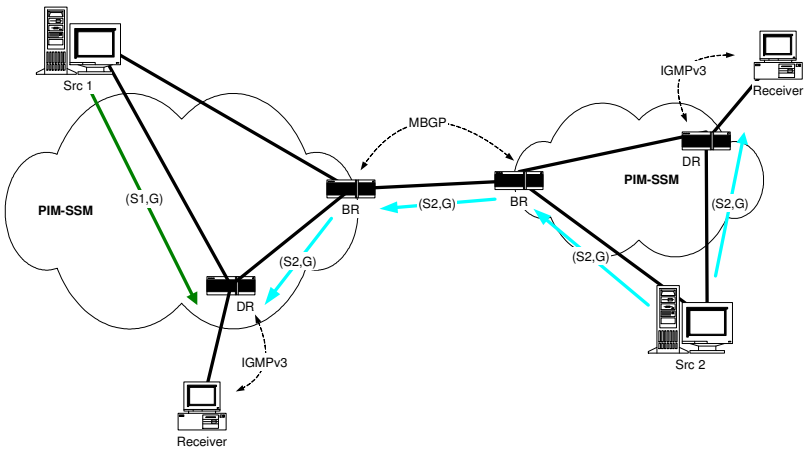
Componentes do Serviço SSM

- **Faixa de endereços exclusiva - `232/8` (IANA)**

- **Roteamento: PIM-SSM**
  - Versão modificada do PIM-SM
  - Pode implementar ambos os serviços (SM & SSM)

- **IGMPv3 (MLDv2 no IPv6)**
  - Suporta a filtragem de fontes
    - `(INCLUDE, EXCLUDE)`

Arquitetura SSM



Funcionamento do PIM-SSM

- **Regras do PIM-SSM**
  - somente *join*(S,G) é permitido na faixa 232/8
  - *join*(*,G) e *join*(S,G) permitidos na faixa restante
  - roteadores de borda (DR no PIM)
    - implementam *join*(S,G) imediato
  - roteadores de núcleo
    - devem evitar as árvores compartilhadas em 232/8

Observações Finais

- **Arquitetura IP Multicast**
  - Continua complexa
  - Ainda possui problemas de escalabilidade
    - Estado armazenado nos roteadores
- **Faltam ferramentas de gerenciamento**
- **Modelo de tarifação em discussão**
- **Conclusão: ainda há muito trabalho a fazer**

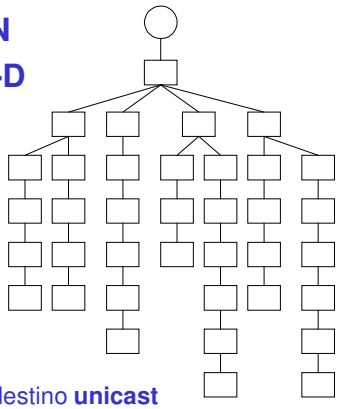
REcursive **UNI**cast **Tr**E**e**s

- **Modelo de distribuição 1 para N**
- **Não utiliza endereço de classe-D**
  - group = <S,P> P – port number
- **Escalabilidade**

  forwarding state (MFT)

  X

  control state (MCT)
- **Distribuição de dados**
  - árvores unicast recursivas
    - os pacotes possuem endereços de destino **unicast**
    - os nós de bifurcação criam cópias modificadas de cada pacote



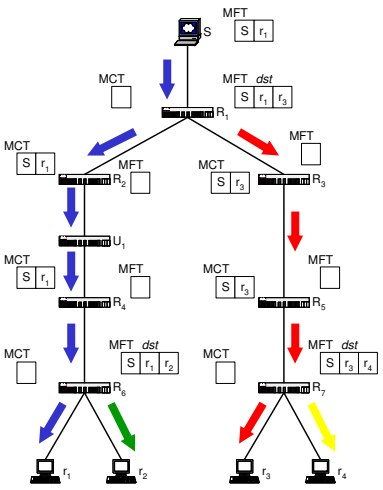
REUNITE

- **Construção da árvore**
  - mensagens ***join(S,G)*** e ***tree(S,G)***
    - ***Joins*** trafegam na direção da fonte
    - ***Trees*** são emitidos em "multicast" pela fonte
  - (potencialmente) árvore SPT (*Shortest-Path Tree*)

- **Problemas se o roteamento unicast é assimétrico**

GTA/UFRJ

Unicast Recursivo



GTA/UFRJ

Construção da árvore REUNITE

Rotas unicast :

$S \leftarrow R_1 \leftarrow R_2 \leftarrow r_1$

$S \rightarrow R_1 \rightarrow R_3 \rightarrow r_1$

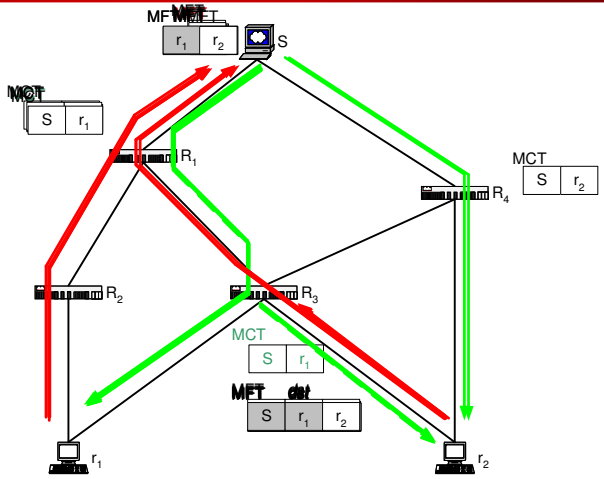
$S \leftarrow R_3 \leftarrow R_1 \leftarrow r_2$

$S \rightarrow R_4 \rightarrow r_2$

r_1 se inscreve;

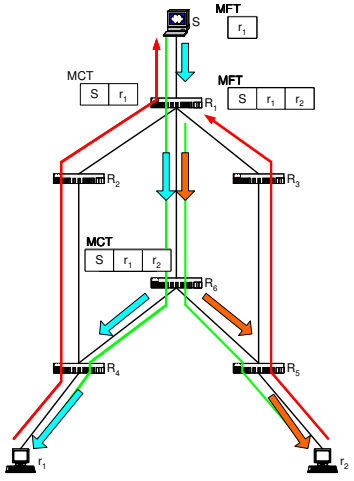
r_2 se inscreve;

r_1 deixa o canal;



GTA/UFRJ

Duplicação de dados



GTA/UFRJ

Problemas do Roteamento Assimétrico

- **Não se garante uma SPT**
  - Atraso
- **Duplicação de dados**
  - Consumo de banda passante
- **Criação de ciclos temporários**
  - Tráfego de controle

XCast

- **Lista explícita de receptores nos dados**
  - Novo cabeçalho no IPv4
  - Extensão de roteamento no IPv6
- **Cada roteador examina o cabeçalho**
  - Se ponto de ramificação
    - Criação de cópias dos pacotes com as respectivas listas de receptores (alcançáveis a partir de cada interface de saída)
- **Não há estado por grupo nos roteadores**
- **Tamanho do grupo é limitado**

Futuro: Multicast no IPv6

- **Todos os nós devem suportar o multicast**
  - Implementações não precisam suportar túneis multicast
- **Modelo de serviço idêntico ao IPv4**
- **Escopo**
  - Definido explicitamente no endereço multicast

GTA/UFRJ