



## Roteamento em Redes de Computadores CPE 825

### Parte VI Roteamento Multicast na Internet

Lúis 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**

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

---

---

---

---

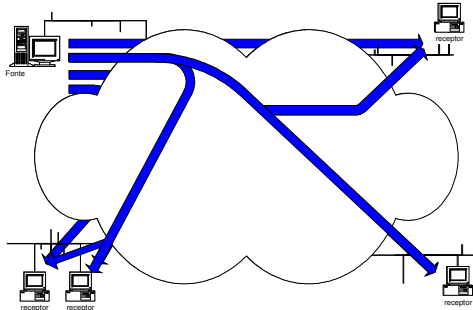
---

---

---

---

## Unicast x Multicast



GTA/UFRJ

---

---

---

---

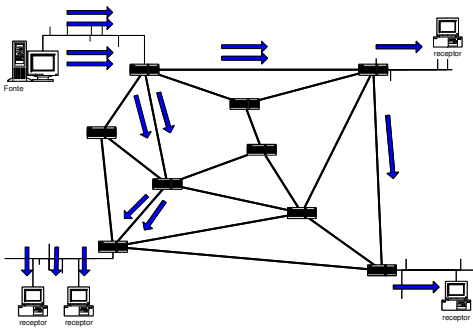
---

---

---

---

## Unicast x Multicast



GTA/UFRJ

---

---

---

---

---

---

---

---

## Utilização do Multicast

### o Vantagens

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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Utilização do Multicast

### o 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

---

---

---

---

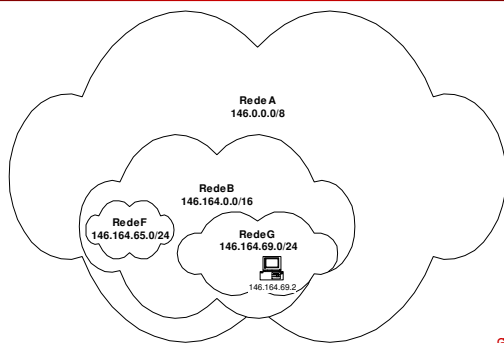
---

---

---

---

## Endereçamento Hierárquico



GTA/UFRJ

---

---

---

---

---

---

---

---

## Endereçamento Hierárquico

### o Hoje em dia

- > CIDR (*Classless Inter-domain routing*)
  - Prefixos podem ter comprimento arbitrário
  - Redes podem ter tamanho arbitrário

### o Rota:

- > Dado um destino, qual o próximo salto? (qual a "porta de saída"?)

### o Roteamento:

Escolher a melhor rota

### o Roteamento Hierárquico:

Agregação de rotas

GTA/UFRJ

---

---

---

---

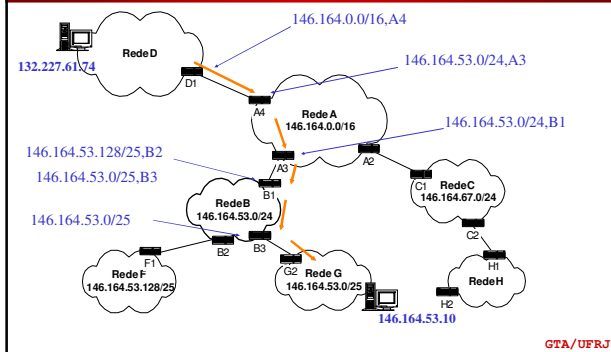
---

---

---

---

## Agregação de Rotas



---

---

---

---

---

---

---

---

## Problema do Multicast

- o **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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Modelo de Serviço IP Multicast

- o **Identificação**
  - > Endereço de grupo
- o **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?

GTA/UFRJ

---

---

---

---

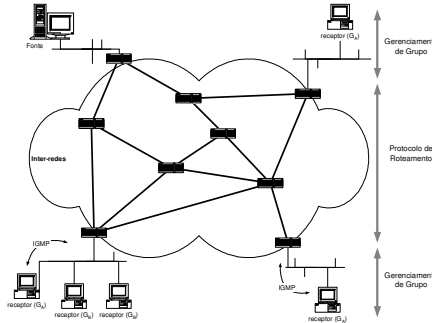
---

---

---

---

## Modelo de Serviço



GTA/UFRJ

---

---

---

---

---

---

---

---

## Modelo de Serviço

### o 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Endereçamento

o **Endereço Multicast = IP Classe D**



o 224.0.0.0 a 239.255.255.255 (224.0.0.0/4)

o **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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

---

---

---

---

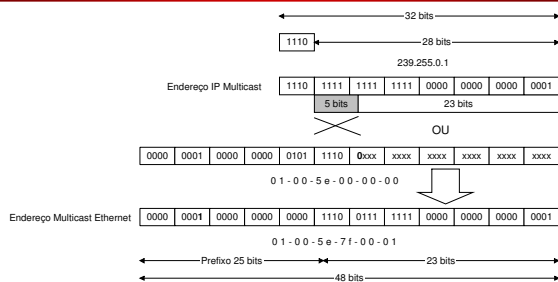
---

---

---

---

## 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...

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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*

GTA/UFRJ

---

---

---

---

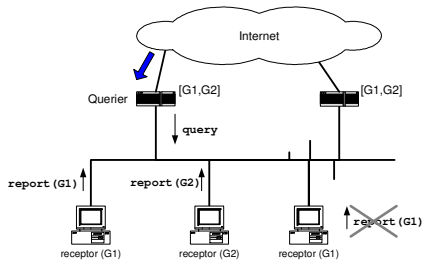
---

---

---

---

## Funcionamento do IGMP



GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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**

GTA/UFRJ

---

---

---

---

---

---

---

---



## 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})`

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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*)

GTA/UFRJ

---

---

---

---

---

---

---

---

## Primeiras Soluções

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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

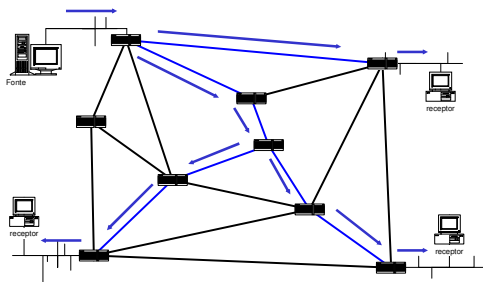
---

---

---

---

## Árvores de Cobertura



GTA/UFRJ

---

---

---

---

---

---

---

---

## Inundação

- Ao receber o pacote
  - Esta é a primeira vez que foi recebido?
    - Se sim, re-envio em todas as interfaces de saída
    - 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

GTA/UFRJ

---

---

---

---

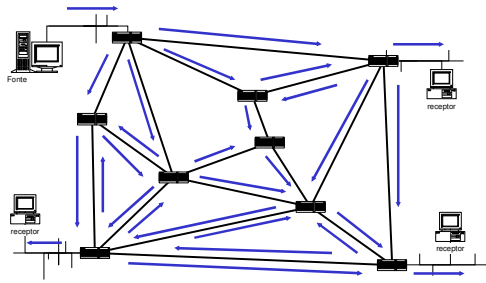
---

---

---

---

## 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**
    - 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

---

---

---

---

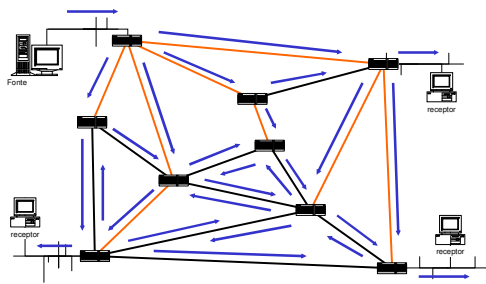
---

---

---

---

## Reverse Path Broadcasting



GTA/UFRJ

---

---

---

---

---

---

---

---

## 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)**

GTA/UFRJ

---

---

---

---

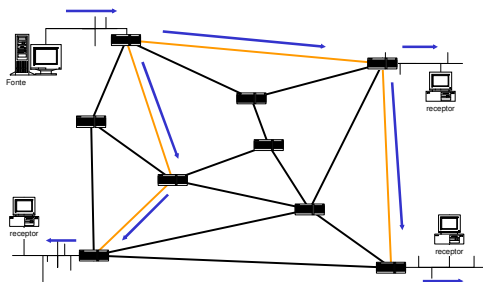
---

---

---

---

## 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

---

---

---

---

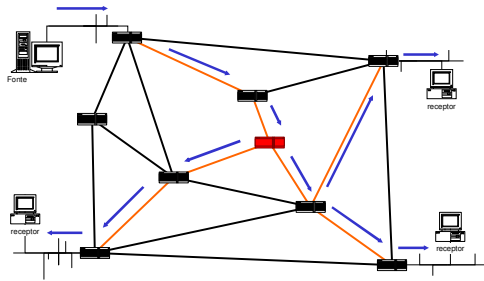
---

---

---

---

## Árvores Centradas



GTA/UFRJ

---

---

---

---

---

---

---

---

## Roteamento Multicast Intra-domínio

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

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

---

---

---

---

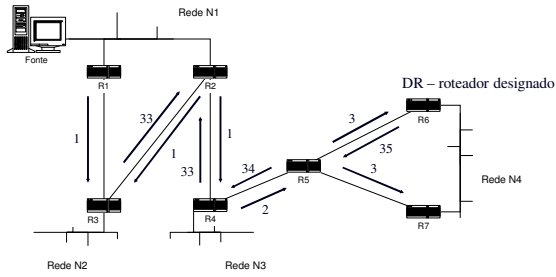
---

---

---

---

## Funcionamento do DVMRP



GTA/UFRJ

---

---

---

---

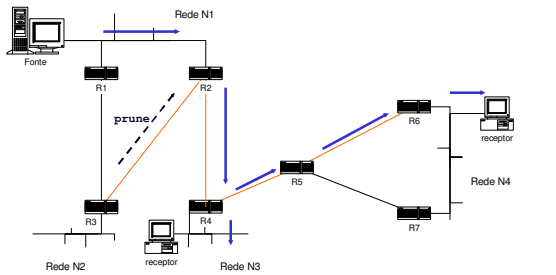
---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

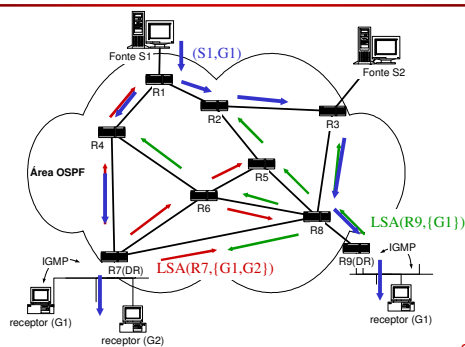
---

---

---

---

## 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-enciam 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

---

---

---

---

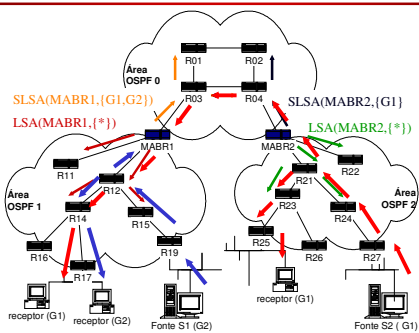
---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---



## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## CBT

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

GTA/UFRJ

---

---

---

---

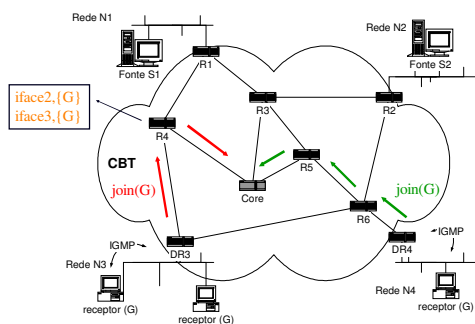
---

---

---

---

## Construção da Árvore CBT



GTA/UFRJ

---

---

---

---

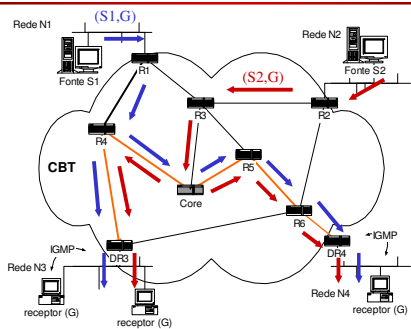
---

---

---

---

## 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**

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## PIM-DM

### o **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)

GTA/UFRJ

---

---

---

---

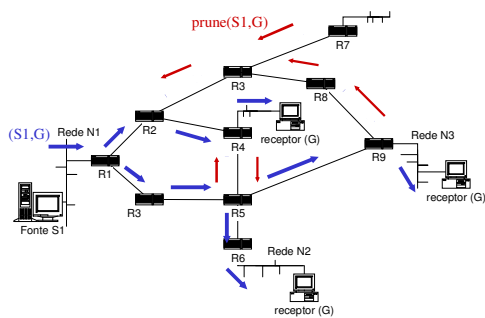
---

---

---

---

## PIM-DM



GTA/UFRJ

---

---

---

---

---

---

---

---

## PIM-DM

### o **Árvore SPT reversa (RSPT)**

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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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**)

GTA/UFRJ

---

---

---

---

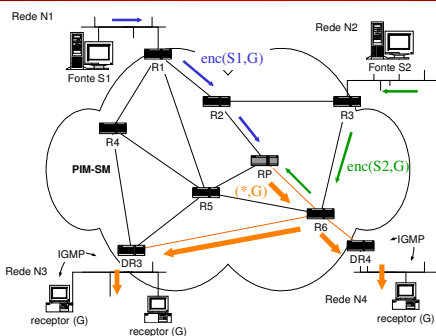
---

---

---

---

## 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

---

---

---

---

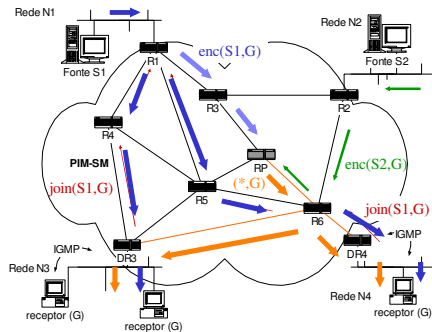
---

---

---

---

## Árvore por Fonte no PIM-SM



GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Alcance do Tráfego Multicast

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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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 número 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

GTA/UFRJ

---

---

---

---

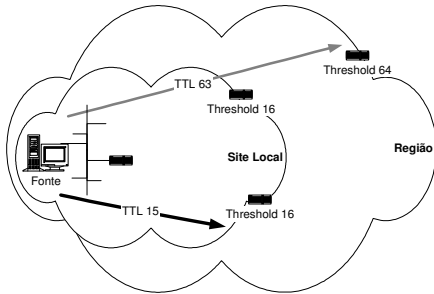
---

---

---

---

## Escopo usando o TTL



GTA/UFRJ

---

---

---

---

---

---

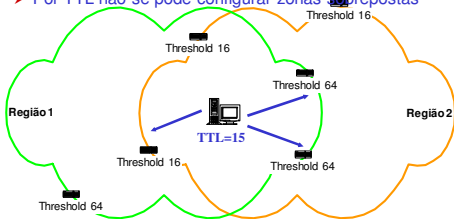
---

---

## Escopos Administrativos

### o Roteadores não encaminham certas faixas de endereços

- > Maior flexibilidade que por TTL
- > Por TTL não se pode configurar zonas sobrepostas



GTA/UFRJ

---

---

---

---

---

---

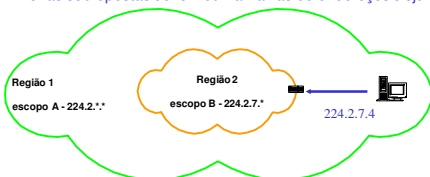
---

---

## Escopos Administrativos

### o 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Escopos Administrativos

- **MZAP (Multicast Zone Announcement Protocol)**
  - Descoberta de zonas de escopo e detecção das inconsistências de configuração mais comuns
- **Idéia básica**
  - O escopo local é a menor zona visível de qualquer ponto da rede
  - Nenhuma fronteira pode cruzar a zona de escopo local
    - (ou a dividiria em zonas locais menores)
  - Receptores escutam um grupo bem-conhecido na zona local e recebem anúncios das zonas de interesse maiores

GTA/UFRJ

---

---

---

---

---

---

---

---

## Alocação de Endereços

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



- Ex. AS 16007 - faixa 233.64.7.0 a 233.64.7.255

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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Arquitetura MAAA (RFC 2908)

- **MADCAP (Multicast Address Dynamic Allocation Protocol) (RFC 2730)**
  - 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) (RFC 2909)**
  - Coordena a alocação de endereços inter-domínio
  - Trabalha com o BGP

GTA/UFRJ

---

---

---

---

---

---

---

---



## 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.

GTA/UFRJ

---

---

---

---

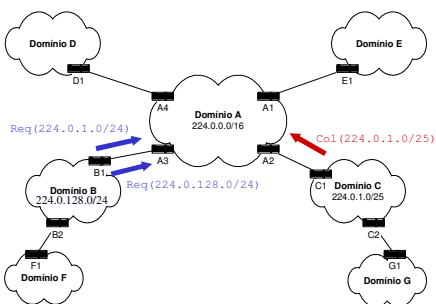
---

---

---

---

## Alocação Hierárquica



GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

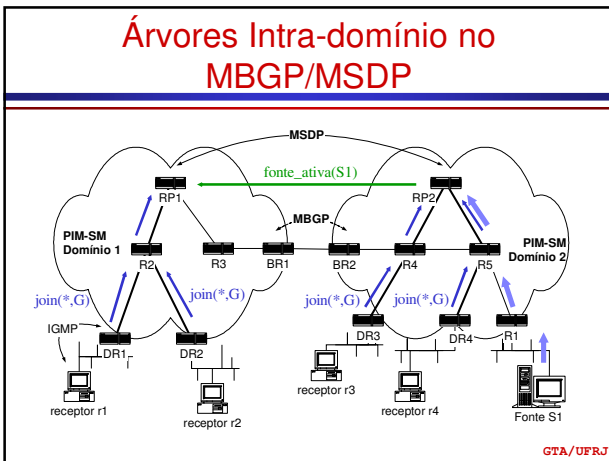
---

---

---

---

## Árvores Intra-domínio no MBGP/MSDP




---

---

---

---

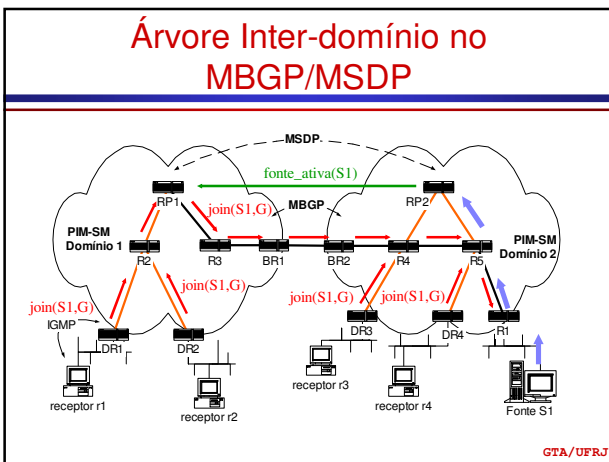
---

---

---

---

## Árvore Inter-domínio no MBGP/MSDP




---

---

---

---

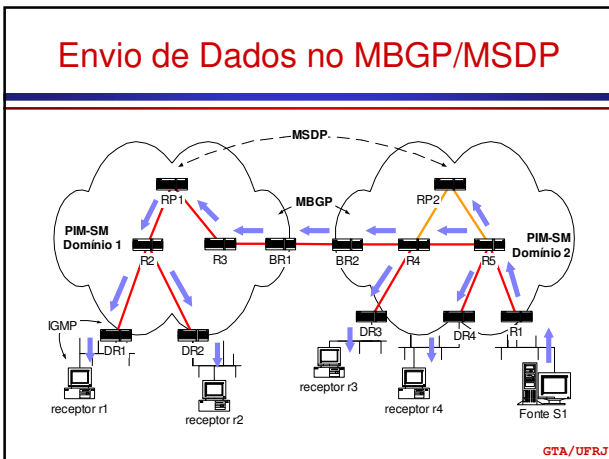
---

---

---

---

## Envio de Dados no MBGP/MSDP




---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Inter-domínio: Próximo Passo

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

GTA/UFRJ

---

---

---

---

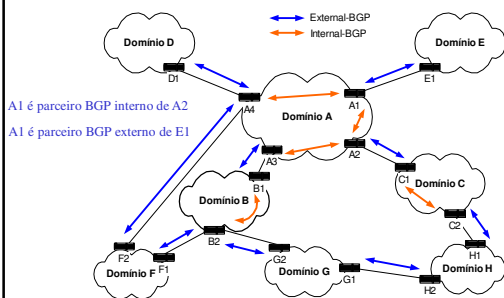
---

---

---

---

## 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**
  - 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

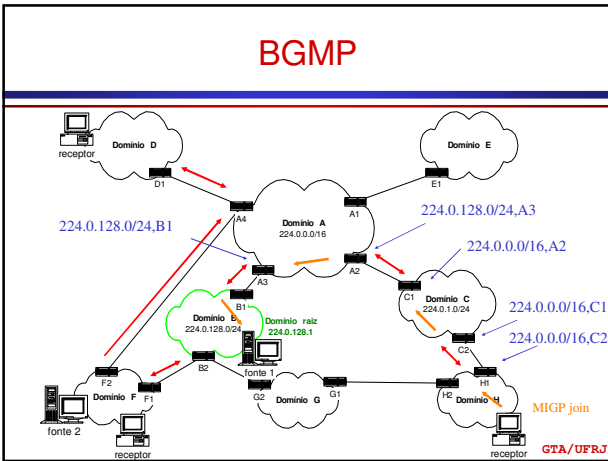
---

---

---

---

# BGMP



---

---

---

---

---

---

---

---

# BGMP

## o 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

---

---

---

---

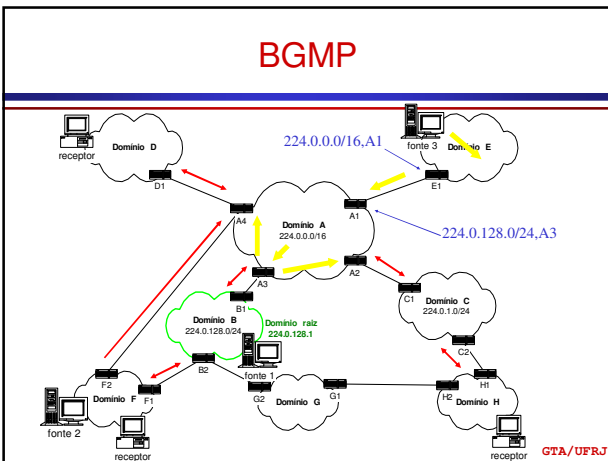
---

---

---

---

# BGMP



---

---

---

---

---

---

---

---

## BGMP

- **Em padronização no IETF**
- **Implantação**
  - > Na escala da Internet
  - > Depende da implantação da arquitetura de alocação de endereços
  - > *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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Protocolos Multicast

- **IGMP**
  - > Gerenciamento de grupo (estações – roteadores designados)
- **Protocolos de roteamento**
  - > Modo denso
    - DVMRP, PIM-DM
      - Inundação-e-poda, árvores por fonte
  - > Modo esparsos
    - 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

GTA/UFRJ

---

---

---

---

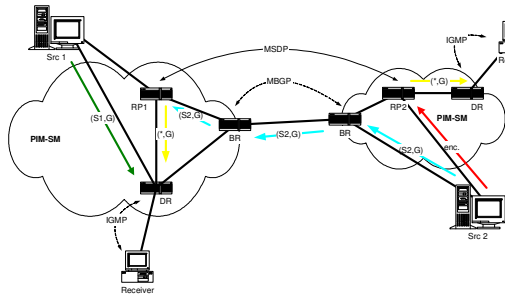
---

---

---

---

## Arquitetura Atual



GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---



## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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)

GTA/UFRJ

---

---

---

---

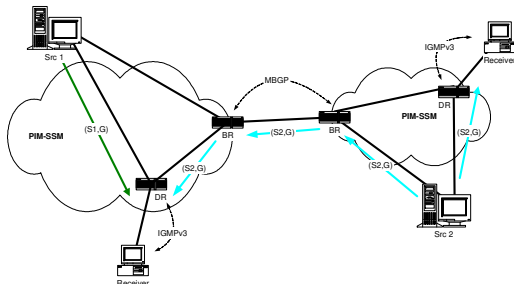
---

---

---

---

## Arquitetura SSM



GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Modificações no IGMPv3

### ○ Using IGMPv3 and MLDv2 For Source-Specific Multicast

<draft-holbrook-idmr-igmpv3-ssm-00.txt>

### ○ Estações

- Módulo IGMP não precisa ser modificado
- Aplicações devem conhecer a faixa de endereços SSM, e utilizar apenas uma API específica à fonte nesta faixa

### ○ Roteadores

- Na faixa de endereços SSM, apenas modo INCLUDE
  - IGMP reports (queries) são processados (produzidos) de acordo

GTA/UFRJ

---

---

---

---

---

---

---

---

## Padronização

### ○ IETF

- > ASM (Any Source Multicast)
  - conversação M x N
  - *Join group G*
- > SSM (Source-Specific Multicast)
  - conversação 1 x N
  - *Subscribe channel <S,G>*

### ○ Protocolo PIM-SSM

- > Incluído na nova especificação do PIM-SM
  - draft-ietf-pim-sm-v2-new-07.txt

GTA/UFRJ

---

---

---

---

---

---

---

---

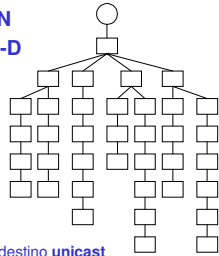
## REcursive UNICAST TrEes

- 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



GTA/UFRJ

---

---

---

---

---

---

---

---

## 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

---

---

---

---

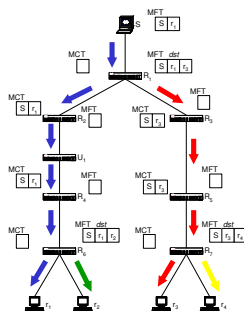
---

---

---

---

## Unicast Recursivo



GTA/UFRJ

---

---

---

---

---

---

---

---

## Construção da árvore REUNITE

Rotas unicast :

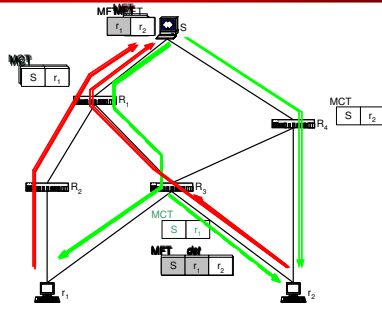
$S \leftarrow R_1 \leftarrow R_2 \leftarrow r_1$   
 $S \rightarrow R_1 \rightarrow R_3 \rightarrow r_1$

$S \leftarrow R_1 \leftarrow R_3 \leftarrow r_2$   
 $S \rightarrow R_4 \rightarrow r_2$

$r_1$  se inscreve;

$r_2$  se inscreve;

$r_1$  deixa o canal;



GTA/UFRJ

---

---

---

---

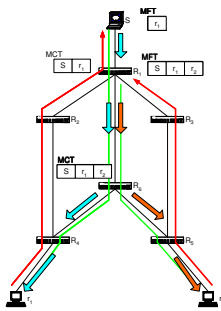
---

---

---

---

## 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

GTA/UFRJ

---

---

---

---

---

---

---

---

## Hop-By-Hop Multicast

- Modelo de distribuição 1 para N
- Abstração de serviço: canal - EXPRESS
  - canal = <S,G> S - @ unicast da fonte  
G - @ IP classe D
- Distribuição de dados - REUNITE
  - > árvores unicast recursivas
    - os pacotes possuem endereços de destino unicast
    - os nós de bifurcação criam cópias modificadas de cada pacote

GTA/UFRJ

---

---

---

---

---

---

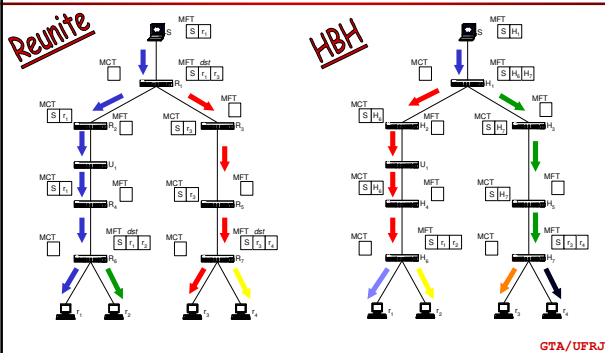
---

---

---

---

## Unicast Recursivo



GTA/UFRJ

---

---

---

---

---

---

---

---

---

---

## Funcionamento do HBH

- Primeiro join sempre atinge S
- mensagens tree instalam entradas na MFT
- mensagens fusion refinam a estrutura da árvore
- Soft-states
  - > joins atualizam as entradas MFT
  - > trees atualizam as entradas MCT
  - > fusions marcam e/ou atualizam as entradas MFT em certos casos especiais
- Data Forwarding
  - entradas *marcada*
  - > envio de controle, não de dados
  - entradas *stale*
  - > envio de dados, não de controle

GTA/UFRJ

---

---

---

---

---

---

---

---

---

---

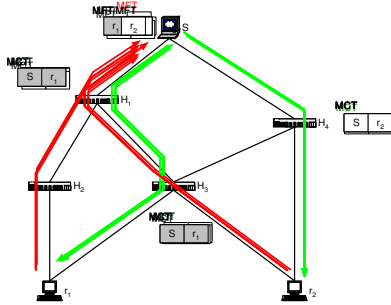
## Construção da árvore HBH

Rotas unicast :

$S \leftarrow H_1 \leftarrow H_2 \leftarrow r_1$   
 $S \rightarrow H_1 \rightarrow H_3 \rightarrow r_1$

$S \leftarrow H_1 \leftarrow H_3 \leftarrow r_2$   
 $S \rightarrow H_4 \rightarrow r_2$

$r_1$  se inscreve;  
 $r_2$  se inscreve;  
 $r_1$  deixa o canal;



GTA/UFRJ

---

---

---

---

---

---

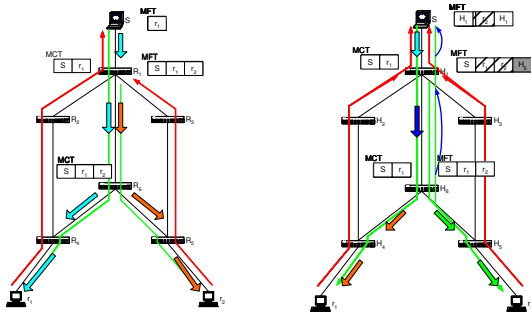
---

---

REUNITE

## Duplicação de dados

HBH



GTA/UFRJ

---

---

---

---

---

---

---

---

## REUNITE x HBH

### ○ A árvore HBH é

- sempre uma SPT
- de menor custo
- mais estável: o caminho de dados fonte-receptor não muda durante a comunicação

### ➤ porém...

- a convergência é mais lenta – o protocolo é mais complexo
- em cada nó de bifurcação uma cópia *modificada* a mais é produzida

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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**

GTA/UFRJ

---

---

---

---

---

---

---

---

## 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
- **Implementação**
  - > IPv4: endereço unicast é utilizado na identificação da interface
  - > Inadequado no IPv6, uma interface pode ter vários endereços

GTA/UFRJ

---

---

---

---

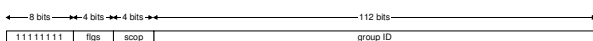
---

---

---

---

## Endereço Multicast IPv6



- **Primeiros 8 bits**
  - > Identificam um endereço multicast
- **Flags**
  - > 3 primeiros bits reservados
  - > bit 4 - flag **T**
    - **T=0** – endereço permanente
    - **T=1** – endereço alocado temporariamente
- **group id**
  - > identificação do grupo (112 bits)
- **Scop**
  - ▣ escopo do tráfego multicast
  - 0 reserved
  - 1 node-local scope
  - 2 link-local scope
  - 5 site-local scope
  - 8 organization-local scope
  - demais não alocados

GTA/UFRJ

---

---

---

---

---

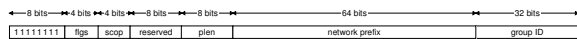
---

---

---

## Alocação de Endereços Multicast

### ○ Endereços multicast baseados no prefixo unicast



- **flgs**
  - bit 2 - flag **P**
    - **P=0** – endereço não baseado no prefixo
    - **P=1** – endereço baseado no prefixo unicast
- **scop**
  - escopo do tráfego multicast
- **plen**
  - comprimento do prefixo de rede
- **network prefix**
  - prefixo da rede (64 bits)
- **group id**
  - identificação do grupo (32 bits)

GTA/UFRJ

---

---

---

---

---

---

---

---

---

---

## Protocolos Multicast no IPv6

- **Roteamento**
  - PIM-SM, PIM-SSM
    - Poucas modificações, já existem implementações
- **Gerenciamento de grupo**
  - MLD (Multicast Listener Discovery)
    - MLDv1 implementa IGMPv2 [RFC2710]
    - MLDv2 implementa IGMPv3 – *internet-draft*
- **MSDP não deve existir**
- **BGMP deve ser implantado**

GTA/UFRJ

---

---

---

---

---

---

---

---

---

---

## 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**

GTA/UFRJ

---

---

---

---

---

---

---

---

---

---