

# ***Uma Arquitetura para Tráfegos "Multicast" no Modelo de Serviços Diferenciados***

**Jefferson Manhães de Azevedo, Luci Pirmez, Luiz Fernando Rust da Costa Carmo,  
Oswaldo Vernet de Souza Pires.**

Núcleo de Computação Eletrônica – UFRJ Caixa Postal 2324, Rio de Janeiro, RJ, Brasil  
E-mail: jazevedo@ucam-campos.br / {luci, rust,oswaldo}@nce.ufrj.br

## **ABSTRACT**

This paper describes an architecture to use multicast with QoS in differentiated services model. That architecture aims at creating an Internet infrastructure for distance learning projects where multimedia interactive applications, like multimedia teleconferences and retrieving multimedia documents, are used. To perform the proposed multicast traffic, the architecture allows an advanced network resource verification and reserve in distribution tree branches of multicast session. This architecture is composed by a reservation system of network resources and differentiated service routers. These routers implement the "multicast premium" service through Expedited Forwarding PHB. Three queuing disciplines were developed in the router schedule module: Priority Queuing, Deficit Round-Robin and to Start-Time Fair Queuing (a variant of WFQ). Two multicast routing protocols types are used such as MASC/BGMP interdomain protocol and a DVMRP intradomain protocol.

## **1- INTRODUÇÃO**

A arquitetura apresentada neste artigo é parte de um projeto de pesquisa sobre qualidade de serviço na Internet que vem se realizando no Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro (NCE/UFRJ). Este projeto tem por objetivo criar uma infra-estrutura de Internet para o projeto de educação à distância do Centro Federal de Ensino Tecnológico de Campos (CEFET-Campos) [01], que faz uso intensivo de aplicações multimídia, como a tele-conferência de vídeo/áudio e a recuperação de documentos multimídia. Tais aplicações necessitam de níveis de QoS ("Quality of Service" - qualidade de serviço) não oferecidos pelo modelo de serviço de entrega de pacotes pelo melhor esforço ("best-effort") do protocolo IP.

A arquitetura está baseada no Modelo de Serviços Diferenciados (Diffserv) [02], que permite diferentes níveis de QoS ao protocolo IP, sem a necessidade de dar um tratamento específico a cada fluxo de dados. Neste modelo, os fluxos de dados são agregados em algumas categorias de serviços pré-estabelecidas. Somente as informações de estado destas categorias de serviço são armazenadas nos roteadores, permitindo que cada agregado de fluxos receba um tratamento diferenciado. Portanto, o modelo Diffserv é bastante adequado às redes de longo alcance (WAN), como a Internet.

Uma porção da Internet, formada por uma ou mais redes, cujos nós (roteadores e "hosts") implementam uma mesma política de tratamento diferenciado aos agregados de fluxos, é chamada de domínio DS ("Differentiated Service"). Cada fluxo que entra em um domínio deve ter seus pacotes identificados por um código ("codepoint") em seu cabeçalho IP [03], que está associado ao tratamento dado a cada pacote no interior do domínio. Além disso, todo fluxo que entra em um domínio deverá estar de acordo com um perfil pré-estabelecido através de um contrato de fornecimento de serviços (SLA - "Service Level Agreement").

Geralmente, um domínio DS possui uma rede provedora de serviços diferenciados e várias redes clientes.

O tráfego "multicast" possui duas características que dificultam a sua implementação no Diffserv: a replicação de pacotes e o dinamismo dos grupos "multicast". Estas duas características podem comprometer os recursos reservados para o tráfego "unicast", pois quando um pacote "multicast" chega a um nó de ingresso de um domínio, ele pode se replicar e sair do domínio por vários nós. Este comportamento do tráfego "multicast" pode ocasionar sobrecarga em alguns enlaces do domínio e violações de alguns SLAs.

O processo de admissão de um novo receptor é complexo, pois os grupos "multicast" são dinâmicos. A qualquer momento um receptor pode solicitar a sua participação em uma sessão "multicast", passando a receber os fluxos de dados daquela sessão. Assim, torna-se impossível saber antecipadamente quais serão os caminhos por onde os fluxos "multicast" irão passar e a quantidade de recursos necessária a uma sessão "multicast". Uma outra dificuldade ocorre quando há uma divisão do fluxo "multicast" no interior do domínio. Como um fluxo "multicast" pode entrar por um nó de extremidade do domínio e sair por vários outros, em algum roteador do interior do domínio haverá uma divisão do fluxo. Neste caso, o fluxo "multicast" se replicará e seguirá por pelo menos dois caminhos diferentes. É possível que um dos caminhos não tenha recursos suficientes para o fluxo "multicast" daquela sessão, causando sobrecarga dos enlaces e violações de SLAs.

## **2- ARQUITETURA PARA TRÁFEGOS "MULTICAST" NO DIFFSERV**

A arquitetura proposta neste trabalho tem por objetivo possibilitar o tráfego "multicast" em redes que implementam os serviços diferenciados fazendo uma verificação antecipada da disponibilidade de recursos e a sua reserva nos enlaces por onde o tráfego "multicast" irá passar. Cada participante de uma sessão "multicast", que desejar receber tráfego "multicast" com QoS, deve fazer uma solicitação de reserva de recursos a um Servidor de Reserva de Recursos (SRR). Esta reserva garante que o tráfego "multicast" que chega ao "host", a partir do transmissor com QoS da sessão, receberá, ao longo de todo o caminho percorrido, um tratamento segundo o tipo de serviço estabelecido no perfil da sessão "multicast". Caso haja falha na reserva, os pacotes dos fluxos "multicast", passando pelo ramo da árvore "multicast" do receptor que não obteve a reserva, são remarcados para receberem um tratamento "best-effort". As reservas podem ser para sessões "multicast" futuras (reservas antecipadas) ou para sessões "multicast" em andamento (reservas imediatas).

O SRR é parte de um sistema de reserva, composto por um conjunto de SRRs, responsável pela verificação e a reserva dos recursos nos enlaces que formam a árvore de distribuição "multicast" de uma sessão. Estes SRRs são similares aos "Bandwith Brokers" [04] do modelo de serviços diferenciados (Diffserv). Cada provedor DS ("Differentiated Service") e cada rede de seus clientes possuem pelo menos um SRR. Os SRRs conhecem a topologia da rede a qual pertencem e executam o mesmo algoritmo de roteamento "multicast" usado nelas (protocolo "multicast" intradomínio), o que possibilita a descoberta dos enlaces por onde o tráfego "multicast" vai passar naquele domínio. Dessa forma, os SRRs podem fazer uma verificação de sobrecarga nos enlaces, bem como fazer a verificação de violações dos SLAs com suas redes clientes (SLAs intradômios). Além disso, é possível localizar os pontos de divisão do fluxo "multicast" no interior do domínio, visto que um dos ramos desta divisão pode não ter recursos suficientes para o tráfego "multicast" de uma sessão. A falta de

recursos em um dos ramos torna necessária uma remarcação dos pacotes para um tratamento “best-effort”. O protocolo intradomínio escolhido foi o DVMRP [05], devido ao seu amplo uso na Internet. Outros protocolos intradomínios podem ser utilizados na arquitetura.

Quando a árvore de distribuição “multicast” de uma sessão se espalha por diversos domínios (uma região DS), os SRRs destes domínios devem se comunicar a fim de fazer as verificações dos enlaces e SLAs interdomínios. Para isso, os SRRs devem participar do roteamento “multicast” interdomínio, implementando o mesmo protocolo de roteamento “multicast” utilizado na região DS. Esta participação permitirá que um SRR de um domínio descubra o endereço do SRR do domínio adjacente, por onde a árvore de distribuição “multicast” se estende. Apesar da arquitetura proposta poder utilizar outros protocolos interdomínio, neste trabalho iremos utilizar o protocolo interdomínio MASC/BGMP ("Multicast Address-Set Claim/ Border Gateway Multicast Protocol") [06].

Além disso, os SRRs também são os responsáveis pela configuração dos roteadores dos domínios. Os roteadores implementam o serviço “multicast Premium” através do PHB “Expedited Forwarding” [07]. O serviço premium [04] é um serviço fim-a-fim, de baixo atraso e baixo jitter, onde a alocação de recursos é feita pela taxa de pico. Apesar das recomendações propostas na RFC 2475 [02], neste trabalho é usado o mesmo “codepoint” e SLA (“Service Level Agreement”) para os fluxos “unicast” e “multicast”.

### **3- IMPLEMENTAÇÃO**

Um protótipo da arquitetura apresentada neste trabalho foi desenvolvido para possibilitar o tráfego “multicast” intradomínio. Os SRR (Servidores de Reserva de Recursos) foram desenvolvidos utilizando o ambiente de desenvolvimento Delphi 4.0 com tabelas em Paradox e sistema operacional Windows 98. Ele é composto por três aplicativos que implementam as suas funcionalidades.

O primeiro aplicativo que compõe o SRR é o cadastro de configurações que permite o cadastramento das configurações necessárias ao funcionamento do SRR. O segundo aplicativo é o responsável por implementar o módulo de autenticação, o módulo de controle de admissão e o módulo de descoberta de rotas. O módulo de autenticação é responsável por verificar se um “host” possui permissão para fazer um reserva. O módulo de controle de admissão é encarregado pela avaliação da disponibilidade de recursos de rede para a admissão de um novo receptor de fluxo “multicast” com QoS. Já o módulo de descoberta de rotas é o encarregado pela descoberta dos enlaces e os roteadores que compõem o ramo que interliga o receptor à árvore de distribuição da sessão “multicast”. Este aplicativo está sempre ativo esperando uma solicitação de perfil ou reserva. O terceiro aplicativo implementa o módulo configurador, que está constantemente verificando a tabela de reservas, verificando se há reservas para configurar. Caso haja, este módulo faz a configuração dos roteadores de acordo com as informações armazenadas na tabela de reservas.

Para permitir que os receptores possam solicitar o perfil e a reserva de uma sessão “multicast” aos SRRs, foi necessário o desenvolvimento de uma outra aplicação hospedada nos receptores, denominada Agente para Reserva.

Os roteadores utilizam o sistema operacional TROPIX (padrão UNIX) [08]. Eles possuem os seguintes módulos: classificador, condicionador, escalonador, configurador e comunicador. O módulo classificador permite a classificação dos pacotes utilizando o “codepoint” (para os agregados de fluxos) ou um conjunto de quatro campos do cabeçalho IP

do pacote (para fluxos individuais): IP origem e destino, porta do aplicativo origem e destino. O módulo condicionador assegura que os fluxos entrando no domínio atendam as características acordadas no SLA. Ele faz o condicionamento tanto dos fluxos individuais quanto dos agregados de fluxos, armazenando estados descritores de reserva com as seguintes informações: identificação do fluxo ou agregado, taxa de transmissão, tipo de serviço e informações de controle. As informações de controle são utilizadas pelo algoritmo de condicionamento "Token Bucket" (balde de permissões). O módulo escalonador é o encarregado pelo tratamento diferenciado dado aos pacotes, sendo implementado por quatro políticas de enfileiramento: FIFO, Priority Queuing, Deficit Round-Robin e Start-Time Fair Queuing (uma variante da "Weight Fair Queuing"). Uma importante característica da implementação deste escalonador é a possibilidade da troca dinâmica das disciplinas de filas (com o roteador em funcionamento). A implementação também possibilita o uso de diferentes disciplinas de filas simultaneamente (cada placa de rede do roteador pode ter uma disciplina de fila diferente das demais placas). O módulo configurador é o responsável pela troca de informações com os SRRs, a fim de configurar o roteador. Por fim, o módulo comunicador implementa o protocolo de comunicação, possibilitando a troca de informações com os SRRs.

O módulo classificador, condicionador e escalonador foram implementados no núcleo do sistema operacional. A decisão de implementar estes módulos no núcleo do sistema está relacionada ao aumento da eficiência do roteamento dos pacotes, pois um evento de chegada de pacotes de dados ao roteador implica necessariamente na ativação imediata dos procedimentos implementados por estes módulos. Tanto o módulo configurador quanto o módulo comunicador foram desenvolvidos como aplicativos de usuário, visto que os procedimentos que eles realizam são mais esporádicos. Todos os módulos foram desenvolvidos utilizando a linguagem ANSI-C.

#### 4- REFERÊNCIAS BIBLIOGRÁFICAS

- [01] E. Reis, J. Azevedo, L. Moreira. "Desenvolvimento e Avaliação de um Ambiente Construtivista de Aprendizagem a Distância para a Formação Continuada de Professores. Os Primeiros Passos da Academia Virtual". III Jornadas de Educação a Distância, MERCOSUL 99, Chile.
- [02] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. "An Architecture for Differentiated Services". RFC 2475, dezembro 1998.
- [03] K. Nichols, S. Blake, F. Baker and D. Black. "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers". RFC 2474, dezembro de 1998.
- [04] K. Nichols, V. Jacobson, and L. Zhang. "A Two-bit Differentiated Services Architecture for the Internet". Internet Draft <draft-nichols-diff-svc-arch-00.txt>, novembro 1997.
- [05] D. Waitzman, S. Deering, C. Partridge. "Distance Vector Multicast Routing Protocol". RFC 1075, novembro 1988.
- [06] S. Kumar, P. Radoslavov, D. Thaler, C. Alaettinoglu. "The MASC/BGMP Architecture for Inter-domain Multicast Routing". SIGCOMM'98.
- [07] V. Jacobson, K. Nichols, K. Poduri, "An Expedited Forwarding PHB", RFC 2598, Junho 1999.
- [08] <http://www.tropix.nce.ufrj.br/>