

Compartilhamento Igualitário de Banda de Internet em Redes Domésticas

José Gilmar Alves Santos Jr¹, Aluizio Ferreira da Rocha Neto²

¹Petróleo Brasileiro S.A. – Petrobras

Av. Euzébio Rocha, 1000 – 59070-900 – Natal, RN – Brasil

²Universidade Federal do Rio Grande do Norte, Superintendência de Informática
Campus Universitário – 59078-970 – Natal, RN – Brasil

jgasjr@gmail.com, aluizio@farn.br

Abstract. *Residential routers currently available at the market do not apply QoS policies by default, leading to problems like degradation of interactive applications that runs at the same time as bulk ones and unequal distribution of bandwidth among devices. QoS options provided by such routers make it possible to prioritize traffic, minimizing the first pointed issue, but it's ineffective to the other. This paper presents a QoS configuration that achieves equal distribution of Internet bandwidth among active devices of the residential network and also make interactive and bulk applications do not harm each other.*

Resumo. *Os roteadores de uso doméstico atualmente disponíveis no mercado não aplicam políticas de QoS na sua configuração padrão, levando a problemas como degradação de aplicações interativas, quando ocorrem simultaneamente a aplicações em rajada, e distribuição desigual de banda entre os dispositivos. As opções de configuração de QoS dos roteadores permitem realizar priorização de tráfego, minimizando o primeiro problema citado, mas sem efeito sobre o segundo. Este artigo propõe uma configuração de QoS na qual a distribuição da banda de Internet disponível é realizada de forma dinâmica e igualitária entre os dispositivos ativos da rede doméstica, mantendo também o efeito de que aplicações de tráfego em rajada não prejudicam as interativas.*

1. Introdução

A Internet está presente em um número cada vez maior de lares brasileiros, bem como a disponibilidade de bandas maiores a custos acessíveis. Segundo levantamento feito pelo Comitê Gestor de Internet no Brasil em 2010, a forma mais comum de acesso é a banda larga fixa e a banda mais comum entre 1 Mbps e 2 Mbps [Barbosa 2011]. Tem se tornado frequente também a presença de mais de um computador por lar, assim como de outros dispositivos capazes de usufruir de recursos em rede, como televisores, consoles de jogos, smartphones e tablets. Neste contexto surge a necessidade da criação de redes domésticas, visando o compartilhamento de recursos como impressora, armazenamento e a própria conexão à Internet.

As atividades realizadas pelos usuários da Internet são diversas: leitura e envio de e-mails, navegação por páginas web, transferência de arquivos, chamadas de voz e vídeo, uso de rádios online, redes sociais, vídeos sob demanda, jogos em rede e muitas outras.

Um dos desafios dos roteadores domésticos é justamente distribuir a banda de Internet entre dos dispositivos conectados, de forma que as atividades possam ser realizadas adequadamente. Aplicações interativas, como jogos e chamadas de voz e vídeo, têm sua qualidade diretamente relacionada à latência observada na rede bem como à variação desta (*jitter*), além de tolerar um certo nível de falhas. Já aplicações em rajada, como transferências de arquivos, serão melhores na medida em que mais banda estiver disponível. Elas são pouco sensíveis à latência e jitter, mas não toleram falhas: um byte errado pode comprometer toda a transferência.

A maioria dos roteadores fornecem mecanismos de configuração de QoS, que possibilitam, dentre outros, priorizar pacotes ACK e aplicações interativas em detrimento das demais. Dessa forma os uploads não degradam os downloads, dado que os pacotes ACK são transmitidos primeiro mesmo que haja fila, bem como a qualidade das aplicações interativas fica pouco comprometida, ao custo de que as aplicações em rajada sofrem mais latência e recebem um pouco menos de banda, fatores que não prejudicam, dado que o requisito mais importante para elas é a transmissão correta dos dados.

A aplicação da solução descrita acima é comumente associada à divisão igualitária de banda entre tráfegos similares, que possui como efeito colateral a distribuição de banda entre os dispositivos da rede de maneira proporcional a quantidade de tráfegos simultâneos realizados, ou seja, se temos um usuário assistindo a um vídeo sob demanda em um dispositivo e outro usuário transferindo um arquivo via *peer-to-peer* em outro, o segundo tende a receber mais banda, dado que a aplicação do primeiro consiste de uma transferência de arquivo e o segundo realiza a tarefa através de várias conexões simultâneas.

Neste artigo propomos uma política de QoS que faz com que a banda de Internet seja distribuída de forma igualitária entre os dispositivos ativos, mantendo o efeito de que uploads não degradam downloads e aplicações interativas não são prejudicadas pelas aplicações em rajada, no limite da banda disponível para cada dispositivo ativo.

Foi desenvolvido um protótipo, baseado na distribuição Linux OpenWRT, a fim de realizar experimentos e comparações com algumas soluções domésticas de QoS existentes. Os testes confirmaram que sem uso de QoS as aplicações interativas são degradadas pelas aplicações em rajada e que o tráfego de upload prejudica o de download. Foi confirmado também que a aplicação de configurações de QoS baseadas em priorização de tráfego viabilizam a ocorrência simultânea de aplicações diferentes, porém levam à distribuição desigual de banda entre os dispositivos. Por fim, a configuração proposta e implementada pelo protótipo leva à distribuição igualitária entre os dispositivos ativos, mantendo viável a ocorrência simultânea de aplicações em rajada e interativas.

O restante deste artigo está organizado da seguinte maneira: na próxima seção são apresentadas as principais características de alguns trabalhos relacionados. Depois a arquitetura de QoS proposta para aplicação em roteadores domésticos é detalhada, seguida da descrição dos experimentos realizados e resultados. Por fim são feitas as considerações finais e propostas de trabalhos futuros.

2. Trabalhos Relacionados

O Linux possui funcionalidades avançadas para roteamento de pacotes e controle de tráfego. O portal *Linux Advanced Routing & Traffic Control*¹ disponibiliza documentação bastante detalhada a respeito e também um script, o *Wondershaper*, usado para otimizar o uso de banda de Internet residencial. Dentre os objetivos do *Wondershaper* estão: manter baixa latência para tráfego interativo, melhorar a experiência do usuário ao navegar por páginas web ao mesmo tempo em que se faz um download ou upload e que os tráfegos de download e upload não prejudiquem um ao outro.

O *Wondershaper* utiliza a infra-estrutura de escalonamento de pacotes do Linux para atingir seus objetivos. Existem duas versões²: a primeira é baseada no escalonador CBQ [Floyd and Jacobson 1995] e a segunda no HTB [Devera and Cohen 2002]. Ambas utilizam o controlador de filas SFQ [McKenney 1990], com objetivo de distribuir a banda igualmente entre os tráfegos na mesma categoria. Independente do escalonador de pacotes, a estrutura de funcionamento é a mesma: são definidas duas classes de tráfego, interativo e não interativo, organizadas hierarquicamente no mesmo nível e filhas de uma classe raiz cujo objetivo é estabelecer um limite global de banda a utilizar. Filtros associados ao escalonador são utilizados para direcionar os pacotes para a classe adequada: os pacotes IP que sinalizam importância no envio³ ou pacotes ACK do TCP ou pacotes ICMP são tratados como tráfego interativo e os demais como não interativos.

É importante ressaltar que o escalonamento de pacotes tem seu efeito anulado se for realizado em um dispositivo no qual não há filas de envio. Por exemplo, se o roteador doméstico for conectado a um modem através de uma rede Ethernet cuja taxa de transmissão seja de 100Mbps e o link do modem com o provedor seja de 1Mbps de download e 256kbps de upload, então o gargalo da comunicação será o modem e raramente serão formadas filas no roteador, tornando inefetiva qualquer configuração de escalonamento de pacotes no mesmo. Daí a necessidade de criar uma classe raiz que estabeleça um limite global de banda a utilizar: dessa forma o roteador manda pacotes para o modem a uma taxa que este último possa dar vazão sem a formação de filas.

Projetado para ser executado em um roteador de borda do lado do cliente⁴, o *Wondershaper* atua no escalonamento de pacotes apenas no sentido do upload. Já para os pacotes no sentido de download, a fila de envio é formada no roteador de borda do provedor de acesso forçando o *Wondershaper* a utilizar um meio indireto de controle: uma política de descarte de pacotes. A existência de filas aumenta a latência na comunicação, efeito prejudicial para aplicações interativas. Como um pacote perdido é interpretado pelo TCP como sinal de congestionamento na rede [Allman et al. 2009], sua reação é reduzir a taxa de transmissão, o que leva a uma redução no tamanho da fila no roteador de borda do provedor, mantendo assim a latência relativamente baixa.

Vale destacar também um trabalho baseado no *Wondershaper*, disponível no por-

¹Disponível em <http://lartc.org/>. Acesso em 3 dez. 2012.

²Disponíveis em: <http://lartc.org/wondershaper/>, último acesso em 3 dez 2012.

³O filtro do *Wondershaper* é baseado no antigo campo *Type of Service* do IPv4, que foi modificado pela RFC 2474.

⁴Equipamento com capacidade de roteamento sob jurisdição do contratante do link de Internet.

tal The Linux Documentation Project⁵, o *ADSL Bandwidth Management HOWTO*⁶. Neste documento é apresentado o *Myshaper*: um script similar ao *Wondershaper*, porém com algumas diferenças relevantes: ao invés de separar apenas tráfego interativo e não-interativo, o *Myshaper* define sete classes de tráfego com diferentes prioridades, aumentando a granularidade no controle. Além disto, ao invés de utilizar apenas filtros associados ao escalonador de pacotes, o *Myshaper* aproveita as facilidades de classificação do *iptables*⁷. Por fim, no lugar de uma política de descarte de pacotes de download, é utilizada uma segunda estrutura de escalonamento. Na implementação do *Myshaper* é utilizado o escalonador HTB e o controlador de filas SFQ.

3. Solução Proposta

3.1. Arquitetura

Podem ser destacadas algumas similaridades entre o *Wondershaper* e o *Myshaper*, no que diz respeito às suas arquiteturas: ambos definem estruturas de escalonamento de tráfego de upload, regras para alocar o tráfego nas mesmas e mecanismos para reduzir a formação de filas no roteador de borda do provedor de acesso, no sentido de download. As estruturas de escalonamento em ambos os casos são hierárquicas em dois níveis: o primeiro limita a banda total usada no sentido de upload e o segundo trata cada classe de tráfego. Cada um possui um mecanismo para identificar e alocar os tráfegos nas classes de escalonamento. A solução aqui proposta conta com quatro componentes: estrutura hierárquica de escalonamento, política de descarte de pacotes, um classificador de pacotes e um monitor de dispositivos ativos.

A estrutura de escalonamento possui três níveis hierárquicos, organizados em árvore. O primeiro nível conta apenas com um nó raiz e é responsável por limitar a banda total usada, forçando com que a fila de pacotes fique no roteador doméstico ao invés de no modem ou no roteador de borda do provedor de acesso à Internet, assim o escalonamento de fato tem efeito. O segundo nível é responsável por dividir o tráfego entre os dispositivos ativos: um nó para cada dispositivo, configurados para usar apenas a fatia da banda que lhe cabe. E, finalmente, o terceiro nível distribui a banda entre os tráfegos concorrentes de cada dispositivo. Existe uma estrutura para o tráfego de download e outra para o tráfego de upload, como no *Myshaper*.

A política de descarte é aplicada a pacotes trafegando no sentido de download, do mesmo modo que é feito pelo *Wondershaper*, com objetivo de minimizar o tamanho da fila no roteador de borda do provedor de acesso e, conseqüentemente, reduzindo a latência na transmissão de pacotes.

O escalonamento de pacotes de download viabiliza a distribuição igualitária de banda entre os dispositivos, enquanto a política de descarte mantém pequena a fila de transmissão no roteador de borda do provedor de acesso. Eventuais atrasos na entrega de pacotes causados pelo escalonador aumentam o RTT e, conseqüentemente, a taxa de envio de pacotes é reduzida. Tal redução também tem efeito sobre o tamanho da fila no roteador

⁵Disponível em <http://tldp.org/>. Acesso em 3 dez. 2012.

⁶Disponível em <http://ldp.linux.no/HOWTO/ADSL-Bandwidth-Management-HOWTO/implementation.html>. Acesso em 3 dez. 2012.

⁷Mecanismo de filtragem de pacotes do Linux. Mais informações em <http://www.netfilter.org/projects/iptables/>. Acesso em 3 dez. 2012.

de borda do provedor, porém a reação do TCP a um pacote perdido é mais drástica, daí a necessidade de usar ambas estratégias para obter baixa latência e distribuição igualitária de banda.

O classificador de pacotes atua em duas etapas: primeiro identifica o dispositivo ao qual cada pacote está relacionado e, em seguida, o tipo de tráfego. De posse das duas informações, a alocação é feita no nó folha adequado da estrutura de escalonamento.

O último componente, o monitor de dispositivos ativos, completa a arquitetura. Ele é responsável por detectar quais são os dispositivos ativos da rede e manter a estrutura de escalonamento e classificador de pacotes atualizados: quando um novo dispositivo é detectado, um nó de segundo nível é criado no escalonador, juntamente com os nós de terceiro nível responsáveis pelos tipos de tráfego e também são criadas as regras no classificador referentes ao novo dispositivo. Os demais nós do escalonador são atualizados a fim de igualar a distribuição de banda. Medidas similares são tomadas quando é detectado que um dispositivo se torna inativo: os nós da estrutura de escalonamento correspondentes são eliminados, bem como as regras do classificador.

Esta solução proposta, assim com as similares, conta com os mecanismos de controle de fluxo e congestionamento do TCP principalmente no que diz respeito à manter baixa a latência nas comunicações. Na presença de tráfego UDP não há garantias de baixa latência, já que o mesmo não conta com os mecanismos do TCP.

3.2. Implementação

Foi desenvolvido um protótipo como prova de conceito da proposta deste trabalho. Assim como o Wondershaper e Myshaper, optou-se pela utilização de Linux no desenvolvimento por uma série de motivos: é um sistema de código aberto, portado para diversas arquiteturas e que conta com boas ferramentas de QoS e classificação de pacotes. Vale destacar também a disponibilidade de firmwares baseados em Linux⁸ para uso em roteadores domésticos, facilitando a aplicação prática da solução proposta neste artigo.

Para o escalonamento de pacotes foi utilizado o *Hierarchical Fair Service Curve* [Stoica et al. 2000], HFSC, presente no Linux. A classificação de pacotes é feita com o iptables em conjunto com seu alvo *CLASSIFY*, também presentes no Linux. Para o monitoramento de dispositivos ativos foi desenvolvida uma ferramenta baseada na tabela ARP do sistema operacional.

O HFSC é um escalonador de pacotes hierárquico, assim como o HTB, mas permite a definição de regras de escalonamento baseadas não somente em compartilhamento de banda, como também em requisitos de tempo real. São criadas duas estruturas de escalonamento com HFSC, uma para o tráfego de download e outra para o de upload. A primeira fica associada à interface do roteador ligada à rede doméstica e a outra à interface ligada ao provedor de acesso⁹.

Abaixo do nó raiz, tanto na estrutura de download quanto na de upload, são criados os nós correspondentes a cada dispositivo, limitando a banda total de cada um à fração

⁸OpenWRT e DD-WRT, por exemplo. Disponíveis em <https://openwrt.org/> e <http://www.dd-wrt.com> respectivamente. Acesso em 3 dez. 2012

⁹Ligação direta, caso o roteador receba diretamente uma conexão a cabo ou ADSL, ou ligação indireta se o roteador estiver ligado a um modem. Em ambos casos o roteador é configurado de modo a ser o gargalo da rede e, portanto, o efeito da solução é o mesmo.

da banda de Internet de modo que a divisão seja igualitária. Finalmente, abaixo do nó de cada dispositivo, são criados três nós: o primeiro com baixíssima latência e baixa taxa de transmissão, o segundo com latência e taxa de transmissão relativamente baixas porém maiores do que no primeiro nó e o terceiro sem especificação de latência e com a maior taxa de transmissão. A ideia é utilizar o primeiro nó para ICMP, pacotes de controle do TCP e qualquer outro que seja muito pequeno. O segundo para aplicações interativas e o terceiro para aplicações em rajada. Por fim, a cada nó folha é associada uma fila SFQ, assim a banda disponível é distribuída igualmente entre os tráfegos concorrentes na mesma categoria de prioridade. Esta estratégia também está presente no Wondershaper e Myshaper.

Para o classificador de pacotes propõe-se utilizar o iptables e seu alvo CLASSIFY. O Wondershaper utiliza apenas filtros associados ao escalonador de pacotes. O Myshaper utiliza o iptables e seu alvo MARK para rotular os pacotes e depois filtros sobre estes rótulos, associados ao escalonador de pacotes. O alvo CLASSIFY do iptables é capaz de alocar os pacotes diretamente aos nós do escalonador, por isso foi preferido em detrimento do MARK nesta solução. Já o iptables foi escolhido pela flexibilidade e facilidade na classificação, comparado aos filtros diretamente associados ao escalonador de pacotes.

A classificação é feita na tabela *mangle* do iptables. Duas *chains* são criadas para cada dispositivo ativo na rede: uma para tratar o tráfego originado no mesmo e outra para o tráfego destinado a ele. São criadas também duas regras na *chain* FORWARD para cada dispositivo ativo, que redirecionam o tratamento do pacote para a *chain* adequada, de entrada ou saída.

São alocados para o nó de escalonamento mais prioritário correspondente a cada dispositivo: os pacotes ICMP, os pacotes TCP com flag SYN, FIN ou RST ativados e demais pacotes cujo tamanho¹⁰ seja de até 64 bytes, como pacotes TCP com flag ACK ativado, enviados somente para sinalizar o recebimento de outro pacote. Para o segundo nó de escalonamento, criado para atender aplicações interativas, são alocados os pacotes de tráfego DNS e qualquer pacote cujo tamanho seja entre 65 e 478 bytes. Os demais pacotes são tratados como pertencentes a aplicações em rajada e, portanto, alocados para o terceiro nó de escalonamento.

O critério de tamanho de pacote foi escolhido para classificação como uma aproximação e por simplicidade: se fosse tomado como critério o protocolo de aplicação então o número de regras seria consideravelmente maior, como pode ser observado comparando o Myshaper e Wondershaper: o primeiro utiliza como critério o protocolo de aplicação e possui 14 regras de classificação, enquanto o segundo utiliza informações no cabeçalho IPv4 e possui apenas 4 regras. Uma quantidade menor de regras é desejável, pois reduz o processamento necessário no roteador. A classificação por protocolo de aplicação tem ainda outro inconveniente: a diversidade de aplicações que usam o mesmo protocolo, como acontece com o HTTP. Ele é usado tanto para aplicações em rajada quanto para interativas: é possível obter grandes arquivos via HTTP, como imagens de CD e instaladores de aplicativos, ao mesmo tempo em que há aplicações como bate-papo web ou mesmo as páginas interativas da Web 2.0. No caso da transferência de grandes arquivos, pacotes com o maior tamanho possível são utilizados pelo TCP, com

¹⁰O tamanho referido aqui e ao longo do artigo é de um pacote da camada 3, ou seja, o tamanho do pacote transmitido fisicamente é maior, pois é adicionado o cabeçalho das camadas mais baixas.

intuito de otimizar a utilização de banda. Já as aplicações interativas tipicamente usam pacotes pequenos de cada vez.

O último componente da solução, o monitor de dispositivos ativos, teve de ser desenvolvido, dado que não há ferramenta nativa do Linux que faça esse papel. Sua função é manter as estruturas de escalonamento e as regras de classificação sempre atualizadas, de acordo com os dispositivos ativos na rede doméstica. A base de funcionamento do monitor é a tabela ARP do sistema operacional: através dela é possível determinar endereços IP e MAC dos dispositivos na rede local doméstica que trocaram mensagens com o roteador recentemente [Benvenuti 2005]. Uma entrada na tabela é criada tão logo haja troca de pacotes entre o roteador e um dispositivo e a entrada é removida após um certo tempo de inatividade, consistindo num bom critério para definir dispositivos ativos na rede.

4. Avaliação Experimental

Foram realizados experimentos com finalidade de comparar o funcionamento da solução proposta com o do Wondershaper, Myshaper e com a situação padrão, ou seja, sem configuração de QoS. Um ambiente composto por cinco máquinas virtuais foi montado, reproduzindo os elementos mínimos necessários ao teste:

- 3 máquinas representando dispositivos da rede doméstica
- 1 máquina fazendo papel de roteador doméstico
- 1 máquina fazendo papel de roteador de borda do provedor de acesso

Optou-se pela utilização da distribuição Linux *OpenWRT* na máquina virtual que faz papel de roteador doméstico. Esta distribuição foi desenvolvida especificamente para dispositivos embutidos, especialmente roteadores. Há diversas imagens binárias disponíveis para uma grande gama de roteadores disponíveis no mercado, bem como para utilização como máquina virtual.

A Figura 1 abaixo ilustra a topologia do ambiente criado. A máquina física que hospeda o ambiente atua no teste representando um servidor na Internet, além de coletar os dados.

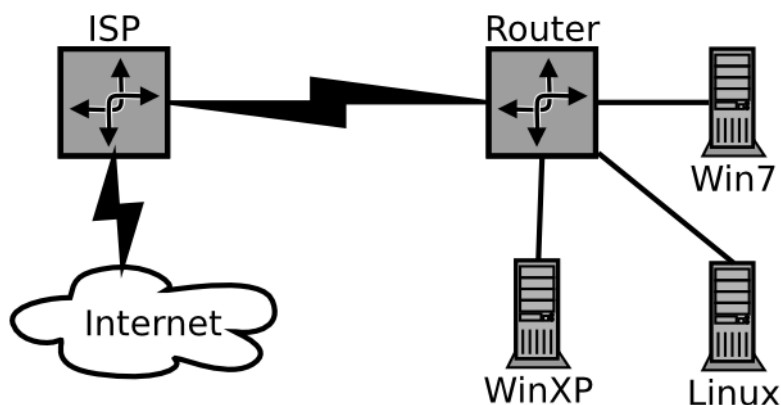


Figura 1. Ambiente de testes

O link entre o Router e os dispositivos da rede doméstica possui capacidade de transmissão de 1Gbps. Entre o Router e o ISP o link físico é de 1Gbps, provido pelo VirtualBox, porém a máquina ISP possui um filtro, baseado no algoritmo *Token Bucket Filter*

[Hubert et al. 2002], limitando as taxas de transmissão a 1Mbps no sentido de download e 256kbps no sentido de upload. A Tabela 1 contém as informações de configuração das máquinas virtuais.

Máquina	Descrição	Memória	Sistema Operacional
Linux	Dispositivo da rede doméstica	128 MB	Debian 6.0 (Linux 2.6.32)
WinXP	Dispositivo da rede doméstica	512 MB	Windows XP SP3
Win7	Dispositivo da rede doméstica	512 MB	Windows 7 Ultimate SP1
Router	Roteador da rede doméstica	32 MB	OpenWRT 10.03.1 r29592 (Linux 2.6.32)
ISP	Roteador de borda do provedor de acesso	64 MB	Debian 6.0 (Linux 2.6.32)

Tabela 1. Configuração das Máquinas Virtuais

Nenhuma das máquinas criadas possui mais de um processador. A máquina ISP possui duas interfaces de rede e cada dispositivo da rede doméstica possui uma. A máquina Router possui duas interfaces: uma ligada à rede doméstica e a outra à máquina ISP.

A máquina que hospeda o ambiente possui a seguinte configuração:

- Processador Intel Core i7 3612QM
4 núcleos de processamento, 2 threads por núcleos
- 8 GB de memória RAM, DDR3 1600 MHz
- Sistema Operacional: Debian GNU/Linux Wheezy
Linux 3.2.32
- Software de Virtualização: VirtualBox 4.1.18

O experimento consistiu em realizar uma sequência de tarefas nos dispositivos da rede doméstica, variando a configuração do Router. A sequência de tarefas está listada a seguir¹¹:

1. Ativar a máquina WinXP
2. Iniciar um ping contínuo com a máquina física
3. Iniciar um download de arquivo da máquina física por HTTP
4. Ativar a máquina Linux
5. Iniciar um ping contínuo com a máquina física
6. Iniciar primeiro download com SCP
7. Iniciar segundo download com SCP
8. Ativar a máquina Win7
9. Iniciar um ping contínuo com a máquina física

¹¹Houve um intervalo entre um e dois minutos do momento em que se inicia uma tarefa até iniciar a seguinte. Os arquivos usados nas transferências foram gerados com tamanho suficiente para que as mesmas não fossem concluídas antes do término do teste.

10. Iniciar primeiro download por FTP da máquina física
11. Iniciar primeiro download da máquina física por HTTP
12. Iniciar segundo download da máquina física por HTTP
13. Iniciar upload de arquivo por FTP para a máquina física a partir da WinXP
14. Iniciar upload de arquivo por SCP para a máquina física a partir da Linux
15. Desconectar a WinXP da rede
16. Desconectar a Win7 da rede
17. Parar o upload de arquivo por SCP na Linux
18. Parar o download de arquivo por SCP na Linux
19. Terminar o teste

Durante os testes havia um capturador de pacotes, o *tcpdump*¹², atuando na máquina Router, salvando o tráfego de ambas interfaces. Após cada rodada de testes os arquivos gerados foram copiados para a máquina física para posterior análise.

Foram realizadas 4 rodadas de testes, variando apenas a configuração de QoS no Router:

1. Configuração padrão, sem QoS
2. Wondershaper ativo
3. Myshaper ativo
4. Solução proposta neste trabalho

4.1. Resultados

Os arquivos de captura gerados durante os testes foram usados para análise através do aplicativo Wireshark¹³. Foi utilizada sua função de gerar gráficos de Entrada/Saída detalhando o tráfego por dispositivo e a exportação de dissecação de pacotes em texto para análise do tempo de resposta dos pacotes ICMP ping.

A Figura 2 ilustra a distribuição do tráfego de download entre os dispositivos da rede doméstica ao longo da duração dos testes.

Pode-se observar que a banda total de entrada ficou aproximadamente em 1Mbps, no cenário sem uso de QoS, com algumas quedas entre 1000 e 1200 segundos. Nos demais casos a banda permaneceu praticamente constante, um pouco abaixo de 900kbps, consequência do limite imposto para evitar a formação de filas no roteador de borda do provedor de acesso.

Quanto à distribuição de banda entre os dispositivos, no cenário sem QoS há muita variabilidade, com clara desigualdade quando apenas as máquinas WinXP e Linux estavam ativas. Após a ativação da Win7 a variabilidade aumentou ainda mais, de modo que não fica explícito se há um dispositivo que recebe mais banda do que os demais.

No caso do Wondershaper, a variabilidade foi menor e a distribuição de banda desigual ao longo da duração dos testes. A WinXP faz apenas um download, enquanto a Linux dois e a Win7 três. O Wondershaper distribui a banda igualmente entre tráfegos da mesma categoria, ou seja, a banda total destinada a um dispositivo é proporcional à quantidade de tráfegos que os mesmos mantêm ativos, efeito que pode ser constatado no gráfico.

¹²Disponível em <http://www.tcpdump.org/>. Acesso em 3 dez. 2012.

¹³Disponível em <http://www.wireshark.org/>. Acesso em 3 dez. 2012.

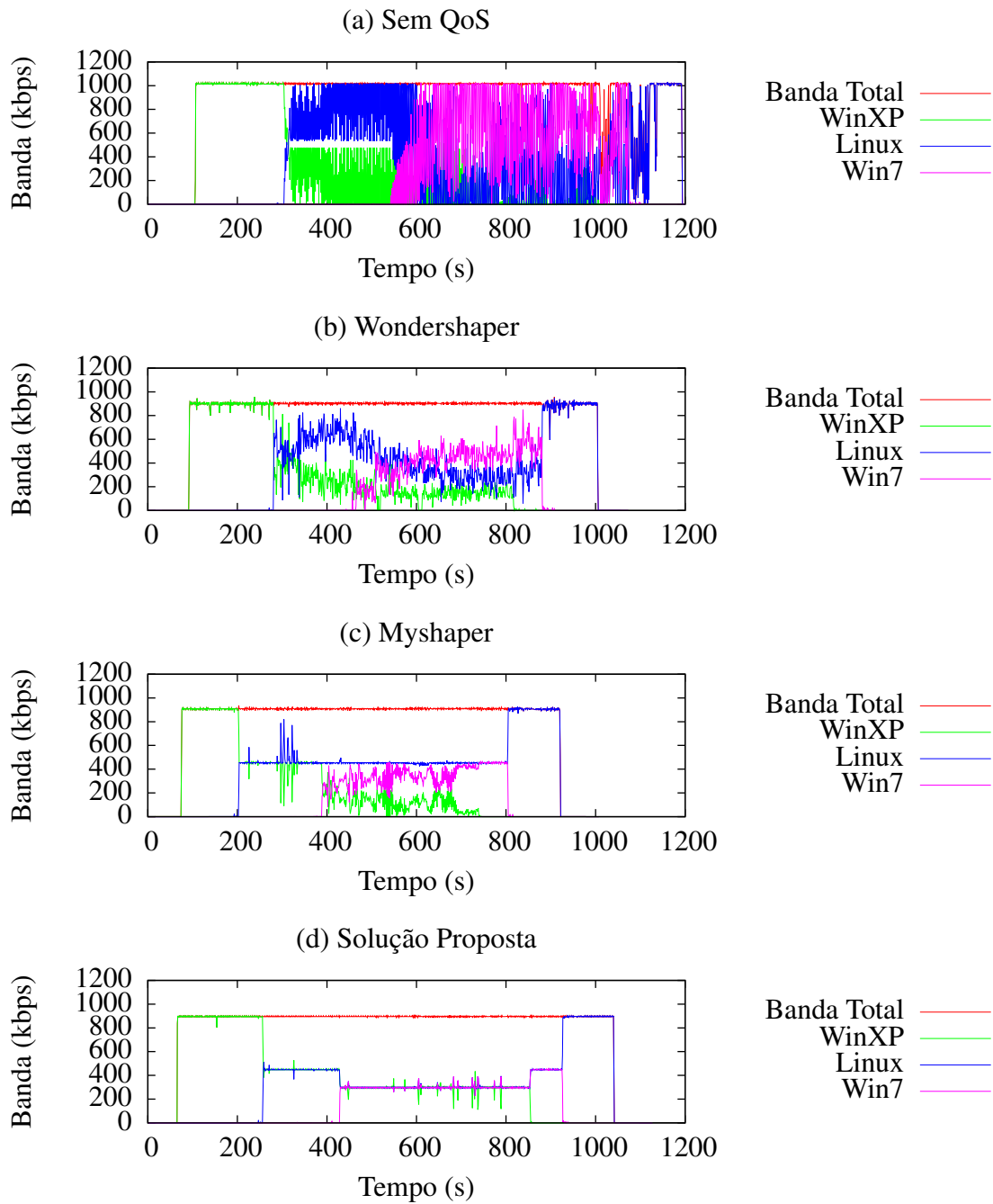


Figura 2. Distribuição de tráfego de Download

Já o Myshaper opera com mais categorias de tráfego, mas mantém a política de distribuição de banda igualitária entre os tráfegos da mesma categoria. Os downloads do Linux são por SSH, considerado pelo Myshaper como prioritário, daí a explicação para a divisão de banda da forma que se vê no gráfico: metade para a Linux e a outra metade distribuída entre Win7 e WinXP com clara vantagem para a Win7.

Finalmente, com a solução aqui proposta, observa-se pouquíssima variabilidade na distribuição de tráfego entre os dispositivos ativos. Inicialmente toda a a banda é

alocada para a WinXP. Após a ativação da Linux cada uma recebe exatamente metade da banda, independente da quantidade de transferências em andamento. Com a ativação da Win7 a distribuição é rebalanceada, de modo que cada uma fica com um terço da banda. Após a desativação da WinXP houve novo rebalanceamento, com distribuição igualitária entre a Linux e Win7. Finalmente, com a desativação da Win7, a Linux recebeu toda a banda.

A Figura 3 ilustra a distribuição de banda para o tráfego de upload. Os dados foram extraídos do conjunto de pacotes capturados na interface do Router conectada à máquina ISP. Neste experimento, refletindo a realidade da grande maioria dos casos reais, o Router realiza NAT para poder compartilhar o link de Internet entre os dispositivos da rede doméstica. Uma das consequências do mascaramento dos endereços de IP originais é que a separação do tráfego por dispositivo de origem fica mais complexa, daí adotou-se uma simplificação. O tráfego foi separado em três categorias: FTP, SCP e Outros. A primeira, de acordo com o roteiro dos testes, foi originado na máquina WinXP, enquanto a segunda na Linux. A terceira categoria agrega os fluxos restantes, compostos majoritariamente por pacotes TCP com flag ACK ativo, pertencentes aos fluxos de download. Dado que o objetivo é verificar a distribuição de tráfego entre os dispositivos e que os uploads por FTP e SCP foram os únicos que saturaram a conexão, então esta aproximação não prejudica a análise.

Um fato que chama a atenção é a ocorrência de picos no cenário sem QoS, onde a taxa de transferência passa dos 1200kbps, valor muito acima do limite de 256kbps configurado na ISP. Isto acontece porque o TCP faz uso de toda a banda disponível, enviando os pacotes a taxa mais alta possível. Então uma fila se forma na ISP, de onde os pacotes são transmitidos na taxa configurada. Eventuais descartes de pacotes pela ISP por ter sua fila cheia e o tempo necessário para que as confirmações de entrega cheguem fazem com que o TCP diminua a taxa de envio, o que explica a formação dos picos no gráfico. Nos cenários reais a formação de fila acontece no modem ligado ao roteador doméstico ou no próprio roteador, se este acumular o papel de modem.

Ainda sobre o cenário sem QoS, vale destacar que no momento do teste em que há uploads concorrentes, um por FTP da WinXP e outro por SCP da Linux, não há distribuição clara da distribuição de banda entre elas, devido à altíssima variabilidade observada e ao fato de que a fila de transmissão foi formada na ISP.

Nos demais cenários, é possível notar um fluxo relativamente constante de tráfego na categoria *Outros*: são os pacotes TCP ACK relacionados aos tráfegos de download. A pouca variabilidade, mesmo no momento em que as transferências de upload estavam ativas, é o que garante que toda a banda disponível de download seja usada, por isso não há reduções na banda de download nos cenários em que alguma solução de QoS foi aplicada, em comparação com o cenário sem QoS.

No caso do Wondershaper nota-se no momento de concorrência entre FTP e SCP a distribuição foi aproximadamente igual, porém com certa variabilidade. Já no cenário com Myshaper, tão logo a transferência por SCP foi iniciada, a banda alocada para a por FTP foi drasticamente reduzida. Além disto, existem alguns momentos, como por volta de 650 segundos, em que as transferências estavam ativas mas nem toda a banda foi usada. Por último, no cenário com a solução proposta neste trabalho, o comportamento

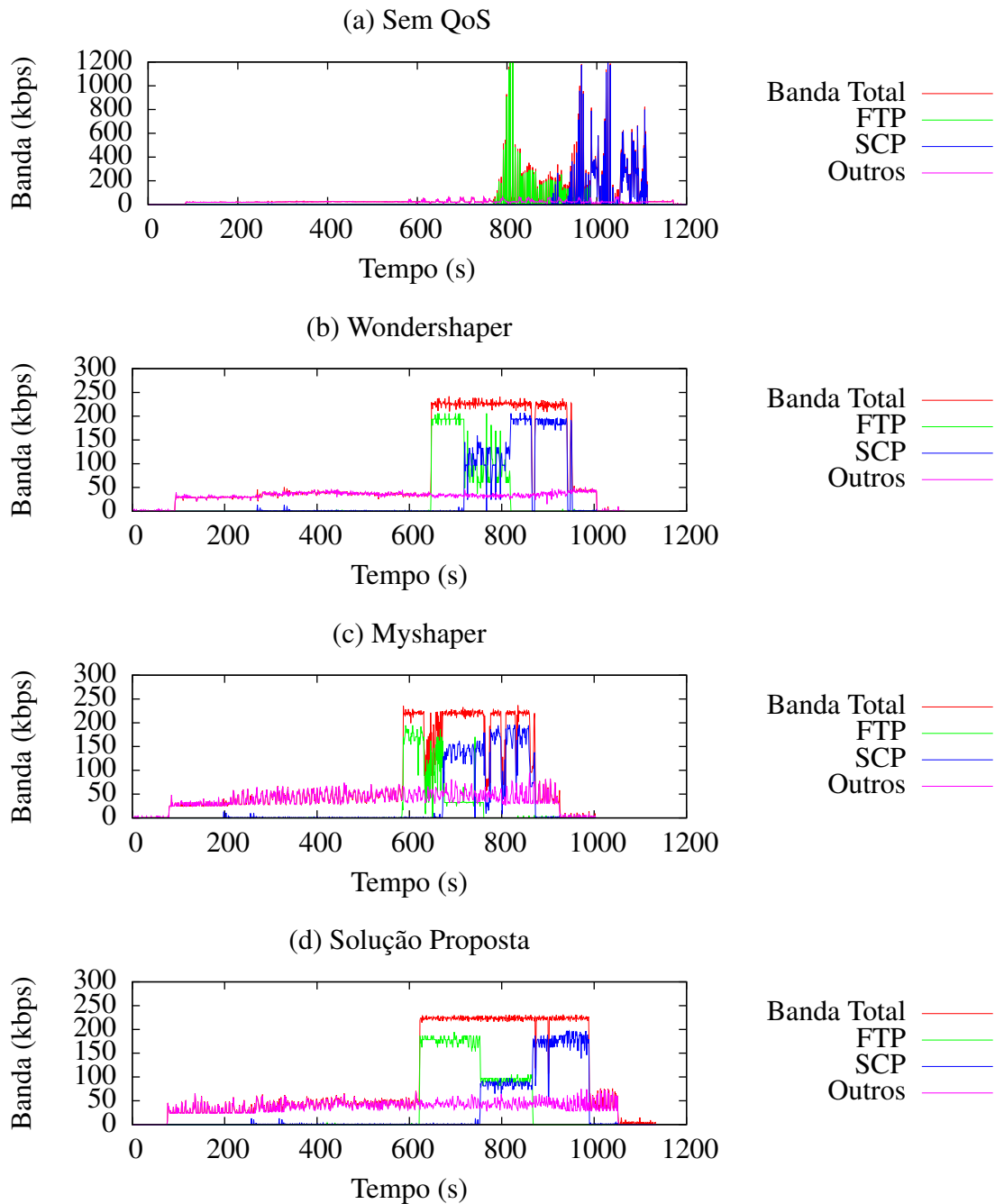


Figura 3. Distribuição de tráfego de Upload

foi bastante parecido com o Wondershaper, porém com menos variabilidade.

Encerrando a análise do cenário sem configuração de QoS, na Figura 4 são apresentados os tempos de resposta calculados através do ICMP ping.

Foi necessário usar escalas diferentes nos tempos de resposta, para viabilizar a percepção de detalhes. No cenário sem QoS e no Myshaper o eixo de tempo varia entre 0 e 7000ms, enquanto no Wondershaper e na solução proposta entre 0 e 400ms.

Após cerca de 300 segundos, no cenário sem QoS, o tempo de resposta aumenta

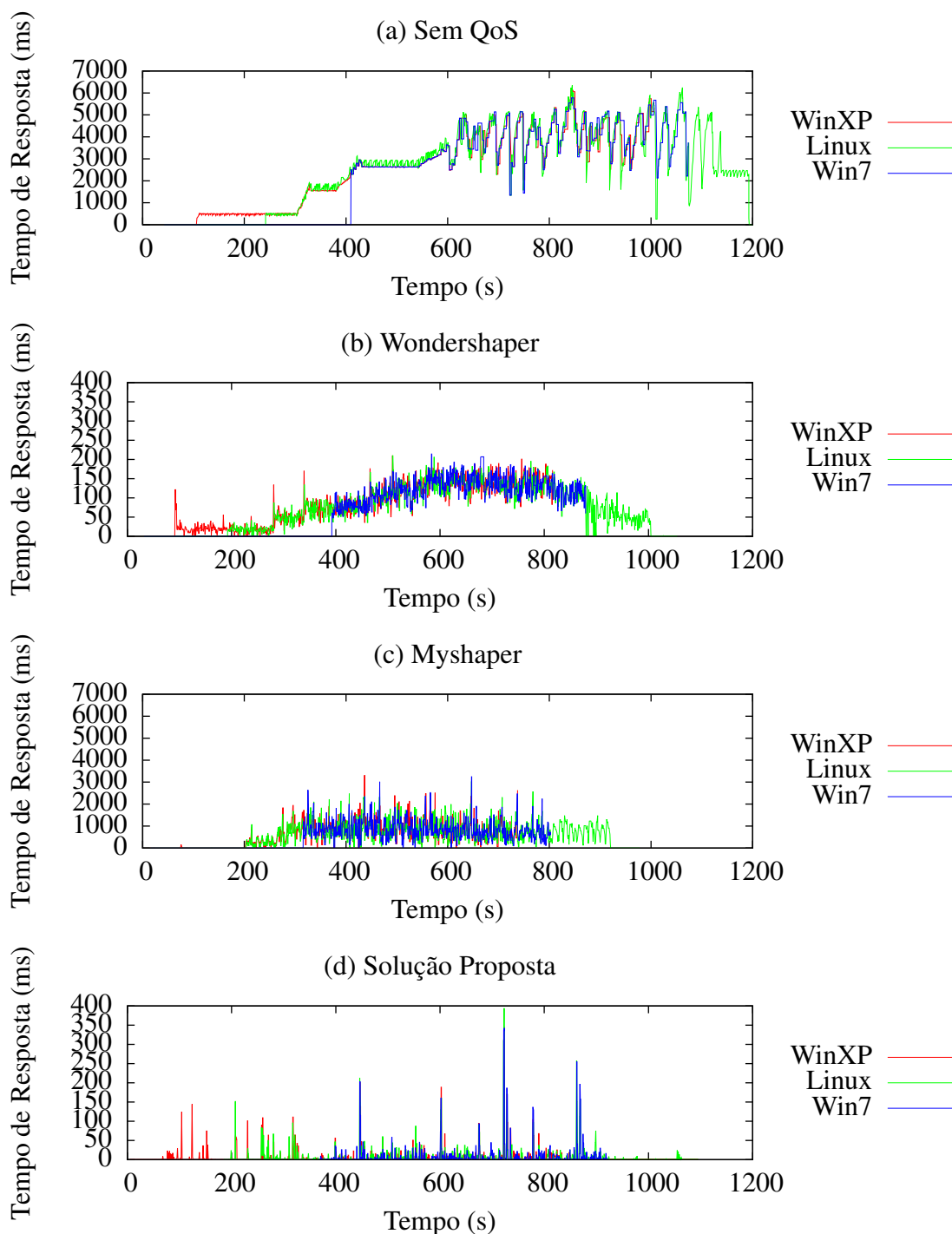


Figura 4. Tempo de Resposta

de aproximadamente 500ms para pouco menos de 2000ms, no momento em que a Linux inicia seu primeiro download, concorrendo com o da WinXP que já estava em andamento. A maioria das aplicações interativas não funcionam com tempos de respostas tão altos. Daí em diante o tempo aumenta mais ainda, chegando aos 6000ms nos momento de mais carga do teste. No caso do Myshaper os tempos de resposta foram claramente menores que no primeiro caso, mas ainda altos para o bom desempenho de aplicações interativas.

Os tempos de resposta observados no cenário do Wondershaper são bastante reduzidos, normalmente abaixo de 200ms, e com relativamente pouca variabilidade, condições bastante favoráveis para aplicações interativas. Já no cenário da solução proposta, a grande maioria dos valores está bem abaixo dos valores do Wondershaper, porém alguns foram acima dos 350ms.

5. Conclusão

Neste trabalho foi proposta uma solução de QoS para uso em roteadores domésticos, com objetivo de distribuir a banda de Internet de forma igualitária e dinâmica entre os dispositivos ativos, além de outros efeitos observados em soluções similares: tráfegos de download e upload não interferem no desempenho um do outro e aplicações interativas não são degradadas por aplicações em rajada.

A solução é composta por quatro componentes: estruturas hierárquicas de escalonamento de pacotes, política de descarte de pacotes, regras de classificação de tráfego e um monitor de dispositivos ativos. A avaliação experimental realizada indicou que a solução proposta de fato atinge o objetivo de distribuir a banda disponível de maneira igualitária entre os dispositivos ativos, comportamento que as soluções similares avaliadas não demonstraram.

Trabalhos futuros incluem a aplicação da solução em roteadores físicos e avaliação de seu comportamento em uma gama maior de situações de tráfego, incluindo transferências de arquivos via peer-to-peer e aplicações de chamada de voz e vídeo.

Referências

- Allman, M., Paxson, V., and Blanton, E. (2009). TCP Congestion Control. RFC 5681 (Draft Standard).
- Barbosa, A. F., editor (2011). *Pesquisa sobre o Uso das Tecnologias da Informação e da Comunicação no Brasil: TIC domicílios e TIC empresas 2010*. Comitê Gestor da Internet no Brasil, São Paulo.
- Benvenuti, C. (2005). *Understanding Linux network internals*. O'Reilly Media, Incorporated.
- Devera, M. and Cohen, D. (2002). Htb linux queuing discipline manual-user guide. *M. Devera web site, Tech. Rep.*
- Floyd, S. and Jacobson, V. (1995). Link-sharing and resource management models for packet networks. *Networking, IEEE/ACM Transactions on*, 3(4):365–386.
- Hubert, B., Graf, T., Maxwell, G., van Mook, R., van Oosterhout, M., Schroeder, P., Spaans, J., and Larroy, P. (2002). Linux advanced routing & traffic control. In *Ottawa Linux Symposium*, page 213.
- McKenney, P. (1990). Stochastic fairness queueing. In *INFOCOM'90. Ninth Annual Joint Conference of the IEEE Computer and Communication Societies. 'The Multiple Facets of Integration'*. *Proceedings., IEEE*, pages 733–740. IEEE.
- Stoica, I., Zhang, H., and Ng, T. (2000). A hierarchical fair service curve algorithm for link-sharing, real-time, and priority services. *IEEE/ACM Transactions on Networking (TON)*, 8(2):185–199.