Os Mecanismos de Qualidade de Serviço em Redes IP

Maria Joana Urbano

O papel da Internet no mundo é hoje inquestionável. De dia para dia, cresce exponencialmente o número de utilizadores que em todo o mundo acede a esta rede global, no emprego, na escola, em cibercafés ou em casa, com finalidades tão distintas como a investigação, o tele-trabalho, a pesquisa e/ou compra de produtos, a consulta e/ou pagamento de serviços, a socialização ou, simplesmente, o lazer. O presente trabalho pretende apontar algumas das limitações técnicas que a Internet enfrenta actualmente – limitações estas resultantes do crescente e diversificado número de utilizadores, aplicações e, consequentemente, tecnologias que lhe são inerentes – ao mesmo tempo que apresenta algumas das tendências tecnológicas mais recentes que têm vindo a ser desenvolvidas no sentido de tentar colmatar tais limitações. Este trabalho mostra ainda a importância crescente da utilização de mecanismos de controlo de tráfego e de qualidade de serviço em equipamentos de comunicações de fabricantes comerciais e faz uma descrição sumária dos módulos de controlo de tráfego existentes nas versões actuais dos sistemas abertos FreeBSD e Linux.

A Internet foi pensada de início para o transporte de tráfego de aplicações tais como a transferência de ficheiros ou a troca de mensagens electrónicas (email), apresentando um modelo de transferência de dados relativamente limitado e insensível ao tipo de tráfego que o utiliza. Neste modelo, baseado no protocolo IP (Internet Protocol)[1],

Interacções número 4. pp. 124-138. © do Autor 2003

todas as aplicações recebem o mesmo tratamento da rede, independentemente das suas características e das suas necessidades. Contudo, o tráfego que atravessa actualmente a Internet está longe de se limitar à simples transferência de dados. Aplicações telemáticas multimédia, tais como o vídeo a pedido (VOD – *Video On Demand*), a videoconferência, a telemedicina ou mesmo aplicações de supercomputação distribuída, são cada vez mais uma realidade nos dias de hoje. Estas aplicações caracterizam-se por terem grandes requisitos de largura de banda, de sincronização entre os diversos meios transmitidos (por exemplo, áudio, vídeo e texto) e podem ser extremamente sensíveis a atrasos de propagação na rede.

Para suportar tal diversidade e heterogeneidade de informação, é necessário fazer alterações na estrutura original da Internet, mais particularmente em camadas específicas da arquitectura TCP/IP. Neste sentido, têm surgido, nos últimos tempos, diferentes estratégias ou tendências, não obrigatoriamente dissociadas, para a evolução da Internet, todas elas com um objectivo comum: a integração no modelo TCP/IP de mecanismos de qualidade de serviço (QoS). Uma destas tendências passa pelo desenvolvimento de aplicações distribuídas adaptativas, capazes de ajustar os parâmetros de tráfego que geram ao estado de carga instantâneo da rede. Como exemplo, uma aplicação pode gerar e transmitir imagens com maior ou menor resolução, de acordo com informação que vai recolhendo ou estatisticamente determinando sobre o estado de carga da rede. Uma outra tendência passa pela diferenciação do tráfego que circula na rede e posterior fornecimento de serviços específicos a cada um dos tipos de tráfego identificados. No fundo, trata se de fornecer um serviço com maior ou menor qualidade de serviço, de acordo com as necessidades e os requisitos de cada tipo de tráfego. A esta estratégia está normalmente associado uma taxa ou custo monetário que os utilizadores têm que suportar quando requerem um determinado nível de QoS para as suas aplicações.

Existem actualmente algumas estratégias que permitem a diferenciação de tráfego e o fornecimento de diferentes serviços ao tráfego diferenciado. O grupo INTSERV do IETF (*Internet Engineering Task Force*) [1] tem trabalhado activamente no desenvolvimento do modelo IntServ (*Integrated Services*) [2], modelo este que se baseia na reserva de recursos do sistema de comunicação (por exemplo, espaço de *buffer*) para fluxos individuais de dados. Por seu lado, o grupo DIFFSERV, também do IETF, apresenta o modelo DiffServ

(*Differentiated Services*) [3], um esquema de diferenciação baseado em agregados de tráfego e classes de serviço, aos quais são fornecidos diferentes níveis de QoS. Mais recentemente, surgiu o MPLS (*Multiprotocol Label Switching*) [4], do IETF, uma arquitectura que integra tecnologias de camada 2 e de camada 3 e que pretende melhorar o método tradicional de encaminhamento de pacotes numa rede de comutação de pacotes.

Nestas tendências, tem-se assistido a uma clara prevalência dos modelos DiffServ e MPLS, em detrimento do modelo IntServ, sendo que existem já operadores de telecomunicações, nacionais e internacionais, a oferecer serviços baseados em *Differentiated Services* e na arquitectura MPLS.

QUALIDADE DE SERVIÇO (QoS)

Foi já referido que as aplicações telemáticas que atravessam a Internet ou qualquer outra rede de comunicações têm diferentes características e diferentes necessidades em relação ao tratamento ou serviço que podem obter do sistema de comunicação. Estas necessidades podem ser traduzidas em parâmetros de qualidade de serviço (QoS), tais como o débito de transmissão, o atraso experimentado pelos fluxos de tráfego da aplicação – ou a variação desse atraso (*jitter*) – e a taxa de erros ou de perdas de pacotes de informação. Aplicações tradicionais, tais como a transferência de ficheiros, a troca de mensagens electrónicas entre utilizadores ou o acesso remoto a sistemas, são aplicações relativamente tolerantes a atrasos e não requerem a disponibilização de grandes quantidades de largura de banda. Já aplicações de videotelefonia ou de videoconferência, que implicam a transmissão de diferentes fluxos de tráfego entre o emissor e o receptor, tais como fluxos de áudio e de vídeo, são muito sensíveis ao atraso sofrido pelos pacotes de informação no trajecto entre as duas entidades comunicantes. Estas aplicações requerem um serviço do sistema de comunicação que lhe garanta um baixo nível de atrasos e, sobretudo, valores quase residuais da variação de atraso ou *jitter*.

Idealmente, um sistema de comunicação conseguirá garantir as necessidades das aplicações que nele transitam, independentemente da sua natureza, se o tráfego de cada uma dessas aplicações puder circular sem quaisquer restrições entre o emissor e o destinatário. Tal implica, por um lado, que cada troço do sistema de comunica-

ções entre o emissor e o receptor deva ter disponível a largura de banda ou capacidade de transmissão que essa aplicação necessita e que, por outro lado, em todos os elementos intermédios da rede (tais como encaminhadores ou *routers*) existentes, entre o emissor e o destinatário, o tráfego da aplicação que chega a esses elementos seja imediatamente processado e imediatamente transmitido para o elemento seguinte, até alcançar o destino. Tal significa que, em todos os subsistemas do sistema de comunicação, incluindo os meios de transmissão e os dispositivos de comutação e encaminhamento, os recursos de transmissão (largura de banda, espaço de *buffer*, capacidade de processamento, etc.) deverão ser ilimitados. Tal, obviamente, não constitui uma verdade nos sistemas actuais e muito dificilmente poderá acontecer em sistemas futuros.

A introdução de mecanismos de QoS surge, então, como uma medida de gestão dos recursos de transmissão existentes nos sistemas de comunicação que, como vimos já, são limitados, na tentativa de o sistema dar a resposta mais adequada a cada tipo de tráfego que o atravessa. Este artigo aborda a gestão dos recursos na óptica dos dispositivos de acesso à rede, quer em termos de gestão de largura de banda no acesso, quer invocando os conceitos de QoS extremo-a-extremo.

Gestão de Largura de Banda no Acesso à Rede

A gestão de largura de banda no acesso a um sistema de comunicação faz se num dispositivo de fronteira (por exemplo, um encaminhador), pressupondo uma série de mecanismos e procedimentos, que são sumariamente listados de seguida:

– A classificação ou diferenciação de todo o tráfego que atravessa o dispositivo e sua associação a classes de tráfego.

– A atribuição de diferentes políticas de QoS ao tráfego diferenciado que podem passar, por exemplo, pela definição de limites mínimos e máximos do débito de transmissão, de prioridades relativas ou absolutas e de condições de empréstimo de largura de banda disponível. A atribuição de políticas de QoS pode ser feita por classe ou agregado de tráfego, fluxo ou até mesmo por ligação individual.

– A execução de acções de condicionamento do tráfego que passa pelo dispositivo, de acordo com a sua classificação. Normalmente, estas acções são divididas em acções de 'policiamento' e de 'modelação' e podem consistir, por exemplo, na eliminação selectiva ou arbitrária de pacotes na interface de entrada, no encaminhamento

directo para a interface de saída ou a colocação dos pacotes em filas de prioridades[2], assim como na limitação do débito máximo de um dado tráfego.

– O escalonamento, associado aos diversos disciplinadores[3] que podem ser implementados no dispositivo de rede e que indicam a ordem e o ritmo com que os pacotes de dados são enviados para o sistema de comunicação, de forma a se poderem cumprir os parâmetros definidos nas políticas de QoS.

– A configuração de classes de tráfego, de políticas de QoS, de filas, de algoritmos de escalonamento e de todos os outros parâmetros associados à gestão de QoS que se encontram normalmente num dispositivo de rede de fronteira com suporte de controlo de tráfego.

– A monitorização do tráfego que entra e que sai do dispositivo físico e a apresentação de um conjunto de relatórios e estatísticas acerca da utilização dos recursos desse dispositivo.

QoS Extremo-a-Extremo

Existem alguns modelos e tecnologias em desenvolvimento para o fornecimento de qualidade de serviço extremo-a-extremo em redes IP. Para cada um dos modelos que vamos abordar, é definido o conceito de 'domínio' como o conjunto de elementos de rede que implementam as funcionalidades do modelo em causa e, por outro lado, o conceito de 'qualidade de serviço extremo-a-extremo' como a QoS oferecida a determinada aplicação entre quaisquer dois elementos de rede pertencentes a esse domínio, normalmente, os elementos de acesso à rede no lado do emissor e no lado do receptor.

Como foi já referido, o IETF (*Internet Engineering Task Force*) tem vindo a trabalhar em duas infra-estruturas distintas para a inclusão de mecanismos de QoS em redes IP: o modelo IntServ (Integrated Services [2]) e o modelo DiffServ (*Differentiated Services* [3]). Paralelamente, tem vindo a ganhar relevo no cenário das comunicações globais a arquitectura MPLS (*Multiprotocol Label Switching* [4]), também do IETF. Estes três modelos vão ser brevemente apresentados de seguida.

IntServ

O modelo de serviço da Internet espelha a filosofia de 'melhor esforço' (best effort) do protocolo IP. Isto é, cada pacote de informação enviado para a Internet recebe da rede, indistintamente, a mesma

qualidade de serviço, tendo como única garantia o facto de a rede tentar o seu melhor para entregar o pacote no destino correcto, de uma forma correcta. De modo a prover a rede com mecanismos de QoS que lhe permitam a diferenciação dos fluxos (i.e., a identificação de sequências de pacotes relacionados gerados por determinada aplicação) que a atravessam e efectuar o posterior fornecimento de um dado serviço a esses fluxos, o grupo INTSERV estendeu o seu modelo de serviços clássico a um modelo de Serviços Integrados ou IntServ [2]. Este modelo assume que as aplicações com necessidades de QoS têm que requerer um serviço específico da rede antes de para lá poderem enviar dados. Tal pedido é feito através de sinalização específica (por exemplo, através do protocolo *Resource ReSerVation setup Protocol* (RSVP) [5]), onde a aplicação dá informação à rede relativamente ao seu perfil de tráfego e pede um serviço que vá de encontro às suas necessidades de largura de banda, de tolerância a atrasos e de níveis de perdas de pacotes. A rede, por seu lado, compromete-se a respeitar as necessidades do serviço desde que o tráfego respeite o perfil especificado.

Actualmente, estão disponíveis os serviços Garantido e de Carga Controlada (controlled-load). Embora existam alguns sistemas que disponibilizam estes serviços, na prática, o modelo IntServ revela-se pouco flexível e pouco escalável, pelo tratamento fluxo a fluxo que realiza e pelo facto de implicar alterações significativas ao modo de funcionamento actual da Internet. Como iremos ver de seguida, o modelo DiffServ mostra-se significantemente mais simples, tanto em termos conceptuais como em termos de implementação, pelo que se tem assistido a uma clara preferência deste último modelo em relação ao modelo IntServ.

DiffServ

O modelo DiffServ ([3]) permite fornecer diferentes tipos ou níveis de serviço ao tráfego de rede. Ao contrário do modelo IntServ, vocacionado para o tratamento de fluxos individuais, a arquitectura DiffServ faz a agregação de fluxos de tráfego em classes, atribuindo posteriormente um determinado comportamento de QoS a cada agregado. Tal permite que os elementos de rede que constituem um dado domínio DiffServ só tenham que distinguir e processar um número relativamente pequeno de agregados de tráfego, independentemente do número de fluxos individuais que cada um dos agregados possa conter.

A arquitectura DiffServ é composta por um número de elementos implementados em nós de rede, incluindo:

– *Per-Hop Forwarding Behaviours* (PHB). Um PHB pode ser definido como o comportamento observável da acção de encaminhamento (*forwarding*) praticada por um elemento de rede Diffserv, aplicada a um determinado agregado de tráfego. Actualmente, são utilizados os PHBs EF ([6]) e AF ([7]) (ver mais adiante).

– Funcionalidades de classificação de pacotes.

– Funcionalidades de condicionamento de tráfego, incluindo medição, marcação de pacotes, modelação e policiamento de tráfego.

As funcionalidades de classificação e de condicionamento de tráfego ocorrem apenas em elementos de fronteira da rede. De um modo bastante resumido, os PHBs são aplicados a agregados de tráfego previamente marcados, utilizando o campo DS do cabeçalho IP. Os pacotes são, então, encaminhados em cada elemento da rede de acordo com o PHB associado ao código DS de cada pacote. O PHB EF (*Expedited Forwarding*, [6]), quando aplicado ao tráfego de uma dada aplicação, num determinado domínio DiffServ, permite fornecer à aplicação garantias de baixos níveis de perdas, latência e *jitter*, assim como um nível de largura de banda mínima garantida. O serviço assim prestado assemelha-se a uma linha alugada virtual. Por seu lado, um grupo PHB AF (*Assured Forwarding*, [7]) permite diferentes níveis de garantias de encaminhamento (*forwarding*) dos pacotes IP de um dado cliente. Existem, actualmente, quatro classes AF, cada qual reservando um determinado espaço de *buffer* e uma determinada largura de banda.

Muito resumidamente, um pacote IP pode ser atribuído a uma ou mais classes AF, de acordo com os serviços subscritos pelos cliente. Dentro de cada uma das classes AF, os pacotes são marcados com um de três possíveis valores de probabilidade de eliminação (*drop precedence*). Em caso de congestão no elemento da rede, este valor indica a importância relativa do pacote na classe AF. Pacotes com valores mais altos são preferencialmente eliminados, em caso de congestão. É ainda de referir que os pacotes pertencentes a uma classe AF devem ser encaminhados de forma independente dos pacotes pertencentes a outra classe AF. Uma classe AF deverá ser configurada de forma a poder receber recursos de encaminhamento extra, quando existe um excesso de recursos disponíveis de outras classes AF ou mesmo de outros grupos PHB.

Por fim, registe-se, e relativamente ao modelo DiffServ, que existem já diversos operadores de telecomunicações, internacionais e nacionais, a oferecer aos seus clientes serviços DiffServ.

MPLS

Nas redes IP actuais, quando um pacote é transmitido ao longo de uma rede constituída por um ou mais encaminhadores (routers), cada um desses encaminhadores analisa o cabeçalho do pacote de dados e executa um dado algoritmo de encaminhamento, de modo a poder determinar o próximo dispositivo de rede para onde deve transmitir o pacote. Na realidade, o encaminhador executa duas funções: primeiro, divide todos os possíveis pacotes numa série de classes de encaminhamento (*forwarding equivalence classes* ou FECs). Depois, mapeia cada FEC num próximo elemento de rede. No entanto, este algoritmo de encaminhamento segue o paradigma do caminho mais curto de Dijkstra, sem atender ao estado de carga de cada troço da rede, o que, por vezes, leva a que sejam escolhidos caminhos congestionados em detrimento de caminhos mais longos, mas nos quais existem os recursos de rede que as aplicações necessitam. O conceito subjacente à arquitectura MPLS ([4]) vem, de alguma forma, colmatar esta deficiência, permitindo encontrar caminhos mais curtos por entre os caminhos que têm efectivamente recursos suficientes para suportar as necessidades dos fluxos de tráfego das aplicações.

Numa arquitectura MPLS, a atribuição de um determinado FEC a um pacote é feita apenas uma vez, à entrada da rede. O valor do FEC é, desta forma, gravado no pacote como uma etiqueta, sendo que os pacotes são etiquetados antes de serem encaminhados. Tal permite que, em *routers* subsequentes, o cabeçalho do pacote IP não necessite de ser novamente processado, bastando usar a etiqueta que o pacote carrega como um índice para uma tabela que especifica o próximo dispositivo de rede e também uma nova etiqueta. Este conceito traz uma série de vantagens relativamente às redes IP convencionais. Para além das já mencionadas, destaca-se a possibilidade de o pacote poder indicar o *router* através do qual ingressou na rede e, consequentemente, as decisões de encaminhamento poderem ser baseadas nessa informação. Uma outra característica de relevo da arquitectura MPLS tem a ver com o facto de as etiquetas MPLS, para além de carregarem informação útil ao encaminhamento, permitem inferir informação relacionada com precedências ou classes de serviço.

Tal como acontece com o DiffServ, alguns operadores de telecomunicações possuem já redes MPLS, oferecendo aos seus clientes este tipo de serviços.

IMPLEMENTAÇÃO DOS MECANISMOS DE QoS

A importância e a necessidade de mecanismos de QoS para superar as limitações da Internet torna-se ainda mais evidente se analisarmos os produtos lançados no mercado pelos grandes fabricantes de dispositivos de rede.

A Packeteer [8] e a Allot Communications [9] são dois exemplos de empresas comerciais que lançaram produtos dedicados ao controlo de tráfego e à gestão de largura de banda no acesso à Internet. A Packeteer desenvolveu o PacketShaper, um dispositivo localizado entre a rede local e o *router* de acesso à Internet, cujos trunfos são a diferenciação e posterior tratamento de uma grande diversidade de tráfego[4], o suporte do modelo DiffServ (PHBs AF e EF) e a inclusão de funcionalidades MPLS. Entre os clientes deste produto encontram se a Sony, o Ministério da Defesa do Reino Unido, a Hewlett Packard e a Walter Thompson. O Packeteer é ainda muito popular em ambientes universitários para a gestão e o controlo dos acessos de todo o campus universitário.

A Allot Communications lançou o NetEnforcer, um dispositivo com características semelhantes ao PacketShaper, com uma versão Enterprise e uma versão ISP. Entre os clientes deste produtos encontram-se a Marinha Norte-Americana, a Wolkswagen e a Ana Aeroporto de Portugal.

O controlo de tráfego é também um ponto estratégico em equipamentos de encaminhamento (*routers*) com integração de serviços IP[5], dispositivos cada vez mais procurados por pequenas e médias empresas para o acesso à Internet. Neste sector de equipamento, encontra-se, por exemplo, o 6WIND GATE, da 6WIND [10], um *router* multiserviço de acesso à Internet, cujo modelo de QoS é baseado na diferenciação do modelo DiffServ.

Finalmente, modelos extremamente avançados de controlo de tráfego e gestão de largura de banda podem ser encontrados em arquitecturas para comutadores (*switches*) e outros dispositivos a operarem ao nível da aplicação. Por uma questão de eficiência e desempenho, estes modelos são normalmente implementados recorrendo a hardware específico, tais como processadores dedicados e memórias ultra-rápidas. Dois exemplos de arquitecturas que incluem sofisticados modelos de QoS são a SynApps, da Radware [11], e a IOS QoS, da Cisco [12].

Normalmente, as empresas comerciais que implementam módulos de controlo de tráfego e de QoS, como as que mencionámos acima,

utilizam versões proprietárias dos algoritmos de QoS que utilizam. O PacketShaper da Packeteer, por exemplo, utiliza tecnologia proprietária e patenteada, tendo, actualmente, 9 patentes e 29 pedidos de patente.

Uma visão oposta é-nos dada pelos sistemas abertos, i.e., sistemas que disponibilizam gratuitamente o seu código para utilização e mesmo para alteração. Destes sistemas, são aqui destacados dois: o FreeBSD, pela sua relevância a nível científico e académico e com uma pilha protocolar bem conhecida e bem comentada em documentação diversa (ver, por exemplo, [13]); e o Linux, um sistema aberto que tem conhecido uma rápida aceitação, não só pela comunidade científica, como também pelos gestores de redes de organizações públicas e privadas. De seguida, são sumariamente apresentados os módulos de controlo de tráfego destes dois sistemas.

FreeBSD

O módulo de controlo de tráfego e de QoS dos sistemas BSD – onde se inclui o FreeBSD – é o ALTQ (*Alternate Queueing*) REF _Ref35542318 \h [14]. A última versão estável do ALTQ é a 3.1, e apresenta as seguintes funcionalidades:

– Implementação dos disciplinadores CBQ, WFQ, FIFOQ, HFSC, JoBS, RED e RIO, fornecendo ainda uma interface para a implementação de novos disciplinadores. O suporte destes disciplinadores é feito apenas na interface de saída.

– Suporte do modelo IntServ e do protocolo RSVP.

– Suporte do modelo DiffServ (nomeadamente, dos PHBs AF e EF), através do condicionamento de tráfego nas interfaces de entrada e do escalonamento preferencial na interface de saída.

– Suporte de mecanismos de gestão de congestão.

– Suporte de IPv6.

– Apresentação de ferramentas de gestão (altq daemon) e de monitorização de tráfego (altqstat monitoring tool).

O ALTQ pode ser implementado no FreeBSD como uma simples extensão do kernel[6].

Linux

O Linux é um 'clone' do sistema Unix que está particularmente bem preparado para a tarefa de controlo de tráfego. De facto, o Linux apresenta, a partir da versão 2.2 do seu *kernel*, um subsistema de rede completamente redesenhado, com código de encaminhamento, filtragem e classificação tão ou mais poderoso do que aquele forneci-

do por muitos routers dedicados e produtos de *firewall* e de modelação de tráfego. Para além do subsistema de controlo de tráfego, o Linux apresenta ainda o Netfilter [15], uma versão particularmente interessante da pilha protocolar que apresenta ganchos (*hooks*) ou pontos de interrupção no percurso dos pacotes de tráfego dentro do dispositivo onde o sistema está instalado para o processamento mais ágil e eficaz desses mesmos pacotes.

O subsistema de controlo de tráfego (TC) do Linux foi escrito por Alexey N. Kuznetsov e adicionado ao *kernel* 2.2 do Linux[7]. O seu funcionamento baseia-se na combinação de três elementos básicos – disciplinadores, classes e filtros de tráfego – que, em conjunto, permitem a classificação, a prioritização, a partilha de largura de banda e a limitação de tráfego em interfaces de entrada e de saída.

As características mais relevantes deste subsistema enumeram-se de seguida:

– Implementação estável (a partir do *kernel* 2.4) de vários disciplinadores: FIFO, TBF, SFQ, PRIO, CBQ, CSZ, TEQL, WRR, HTB, RED, GRED e DS_MARK.

– Oferta de um conjunto rico e diversificado de filtros (*matching rules*), dentro de cada disciplinador, incluindo filtros u32[8], fw, route, rsvp, rsvp6, tcindex, protocol, parent, prio e handle. A implementação destes filtros é optimizada através da utilização de algoritmos *hashing*.

– Implementação de um dispositivo virtual, IMQ (*Intermediate Queueing Device*) que estende até à interface de entrada (ingress) as capacidades de gestão de tráfego existentes na Interface de saída (outgress) do dispositivo e permite o estabelecimento de limites globais de largura de banda, tratando cada interface como uma classe.

– Capacidade para a configuração hierárquica de classes de tráfego.

– Suporte do modelo IntServ.

– Suporte do modelo DiffServ, nomeadamente dos PHBs AF e EF[9].

A interface entre o utilizador e o código do controlo de tráfego do *kernel* é feita a partir da '*package*' iproute2, sendo normalmente necessário acrescentar este código à distribuição do Linux. O iproute2 é constituído pela aplicação tc que permite a configuração dos disciplinadores, classes e filtros; pela aplicação ip, que permite a configuração estática ou baseada em políticas (*policy-†based*) de endereçamento e *routing* (i.e., a configuração de ligações, endereços e rotas); e a aplicação rtmon, usada para a captura de estatísticas sobre o tráfego que atravessa o dispositivo de rede num determinado instante.

O Netfilter, como já foi referido, é uma versão da pilha protocolar do Linux (IPv4, IPv6 e DECnet) que apresenta uma série de ganchos (*hooks*) onde podem ser acrescentados módulos do *kernel* para o processamento de pacotes de rede. Por exemplo, o Netfilter permite que, em certos pontos do caminho de um pacote pela pilha protocolar, exista um módulo que encaminhe o pacote para a interface de saída, elimine ou altere esse pacote ou o coloque numa fila para ser tratado em espaço de utilizador. Um destes módulos é o iptables[9] [15], que é parte integrante da versão Netfilter do *kernel* 2.4 do Linux.

Um pacote que atravesse o IPv4 pode ser interceptado em cinco ganchos diferentes: logo quando entra no dispositivo de rede, após a execução de testes de validação do pacote, e antes do código de encaminhamento (NF_IP_PRE_ROUTING); quando o pacote é destinado à máquina local e antes de ser passado a qualquer processo (NF_IP_LOCAL_IN); quando o pacote é encaminhado para outro dispositivo de rede (NF_IP_FORWARD); depois do código de encaminhamento e imediatamente antes de ser novamente enviado para a rede, no caso de o pacote ser destinado a outra máquina (NF_IP_POST_ROUTING); e no caminho dos pacotes que são criados localmente (NF_IP_LOCAL_OUT).

Os módulos do kernel podem-se registar para ouvir em cada um dos ganchos. Um módulo que esteja a manipular um determinado pacote de dados num dado gancho pode dizer ao Netfilter para continuar a travessia normalmente (NF_ACCEPT), eliminar o pacote, finalizando a respectiva travessia (NF_DROP), tomar conta do pacote, não continuando a travessia (NF_STOLEN), colocar o pacote numa fila, normalmente em espaço de utilizador (NF_QUEUE), ou chamar de novo o gancho (NF_REPEAT).

O iptables é uma aplicação para a selecção de pacotes construída sobre a arquitectura Netfilter. Este método de selecção de pacotes é usado para a filtragem de pacotes (*stateless* e *stateful*), para a tradução de endereços de rede (NAT, ou *Network Address Translation*) e para executar funcionalidades genéricas de manipulação de pacotes, tal como a alteração do campo DS dos pacotes IP.

O iptables pode ser utilizado conjuntamente com o iproute2, tornando o módulo de controlo de tráfego mais ágil e robusto.

NOTAS

1 Este facto leva a que, por vezes, os conceitos de redes IP e de Internet se confundam.

2 Num dispositivo de rede com mecanismos de QoS, os pacotes de dados das aplicações são colocados em filas de espera antes de serem transmitidos para o sistema de comunicação. De um modo simplista, um disciplinador é um mecanismo de gestão de filas de espera que, entre outras coisas, determina qual é o próximo pacote a ser enviado para a rede. Na realidade, os disciplinadores assumem um papel de primordial importância na implementação de algoritmos de qualidade de serviço.

3 Incluindo diferenciação ao nível da aplicação, detectando, por exemplo, aplicações Napster, Oracle e Citrix e tráfego resultante de aplicações de porto dinâmico.

4 Entre estes serviços, incluem-se serviços de segurança, VPN e *firewall*, mobilidade, migração IPv4/IPv6 e, claro, os serviços de QoS e controlo de tráfego.

5 Estava prevista a integração do ALTQ no *kernel* 5.0, mas prevê-se que tal venha a ser adiado para versões 5.X posteriores.

6 A primeira versão estável do Linux a incluir mecanismos de controlo de tráfego foi a versão 2.0.36, através de um mecanismo chamado *Traffic Shaper*. Este mecanismo oferecia, contudo, diversas limitações a uma gestão eficaz de QoS e largura de banda e destinava-se apenas a tráfegos de saída.

7 Permitem a classificação do tráfego, tendo em conta os campos dos pacotes IP, nomeadamente, endereços e portos de origem e de destino, protocolo (tcp, udp, icmp, gre, ipsec), TOS e fwmark (útil, por exemplo, para pacotes previamente marcados com ipchains ou iptables).

8 O código para a implementação do DiffServ no Linux foi escrito por Werner Almesberger e é suportado pelas seguintes funcionalidades: o disciplinador sch_dsmark que extrai e marca o campo DS dos pacotes; o classificador cls_tcindex que utiliza a informação do campo DS para colocar o tráfego em classes de tráfego; e o disciplinador sch_gred que suporta múltiplas prioridades de eliminação de pacotes e partilha de espaço de *buffer*.

9 O módulo iptables substitui o módulo ipchains de versões anteriores do kernel do Linux.

REFERÊNCIAS

[1] Internet Engineering Task Force, http://www.ietf.org.

[2] R. Braden , D. Clark, S. Shenker , 'Integrated Services in the Internet Architecture: An Overview', RFC 1633, http://www.faqs.org/rfcs/rfc1633.html.

[3] S. Blake et al., 'An Architecture for Differentiated Services Framework', RFC 2475 http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2475.html.

[4] E. Rosen, A. Viswanathan, R. Callon, RFC 3031, 'Multiprotocol Label Switching Architecture, http://www.faqs.org/rfcs/rfc3031.html.

[5] Resource ReSerVation setup Protocol (RSVP), http://www.ietf.org/html.charters/rsvp-charter.html.http://www.ietf.org/html.charters/rsvp-charter.html.

[6] V. Jacobson et al., 'An Expedited Forwarding PHB', RFC 2598, http://www.faqs.org/rfcs/rfc2598.html.

[7] J. Heinanen, 'Assured Forwarding PHB', RFC 2597, http://www.faqs.org/rfcs/rfc2597.html.

[8] Packeteer, http://www.packeteer.com.

[9] Allot Communications, http://www.allot.com.

[10] 6WIND, http://www.6wind.com.

[11] Arquitectura SynApps da Radware, http://www.radware.com/content/products/synapps.asp.

[12] Arquitectura IOS QoS da Cisco, http://www.cisco.com/en/US/tech/tk543/tk545/tech_protocol_family_home.html.

[13] G. Wright, R. Stevens, 'TCP/IP Illustrated', volumes I and II, Addison Wesley, 1995.

[14] Kenjiro Cho, 'A Framework for Alternate Queueing: Towards Traffic Manangement by PC-UNIX Based Routers'.

[15] Netfilter, http://www.netfilter.org.

Os Mecanismos de Qualidade de Serviço em Redes IP

Quality of Service Mechanisms in IP Networks

Sumário

O modelo actual da Internet apresenta algumas limitações principais, quando confrontado com a cada vez maior quantidade, diversidade e exigência das aplicações que atravessam esta rede universal. Este artigo faz uma apresentação sumária dos modelos desenvolvidos pelo IETF no sentido de colmatar essas limitações, nomeadamente, os modelos IntServ e DiffServ e a arquitectura MPLS. É ainda referida a importância crescente da utilização de mecanismos de controlo de tráfego e de QoS, em equipamentos de comunicações de fabricantes comerciais, e efectuada uma descrição sumária dos módulos de controlo de tráfego existentes nas versões actuais dos sistemas abertos FreeBSD e Linux.

Summary

This paper points out the main limitations of the current model of the Internet, when confronted with the growing number and diversity of the applications that traverses this global network, and briefly presents three IETF models intended to couple with these limitations: the IntServ and DiffServ models and the MPLS architecture. The article also makes reference to the growing importance of the traffic control and the QoS mechanisms on commercial network equipments, and briefly presents the traffic control modules in current versions of the FreeBSD and Linux open systems.