

Estabelecimento de Sessões SIP com Garantias de QoS e sua Aplicação em um Domínio DiffServ

Roberto Willrich^{1,2}, Luiz H. Vicente^{1,2}, Achilles C. Prudêncio^{1,2},
Victor S.N. Alves¹, Rafael B. Uriarte^{1,2}, Felipe B. Teixeira¹

¹Dpto de Informática e Estatística – Universidade Federal de Santa Catarina (UFSC)

²Programa de Pós-Graduação em Ciência da Computação (PPGCC - UFSC)

Caixa Postal 476 – 88040-900 – Florianópolis – SC – Brasil

{willrich, lhvicente, achilles, victors,
Rafael.uriarte, felipecomp19}@inf.ufsc.br

Resumo. *As soluções atuais de Qualidade de Serviço (QoS) asseguram que o tráfego de voz receba um tratamento preferencial sem a intervenção direta dos usuários realizando a chamada. Já existem algumas propostas de extensão do protocolo SIP (Session Initiation Protocol) permitindo a negociação do nível de QoS das sessões. Mas elas usam parâmetros de QoS fixos e/ou são dependentes de tecnologias. Este artigo propõe uma solução baseada em ontologia para o estabelecimento de sessões SIP com garantias de QoS, oferecendo como principal característica a flexibilidade em termos de parâmetros de QoS. A solução proposta foi prototipada e testada em um domínio de rede Linux DiffServ e com um servidor VoIP Asterisk.*

Abstract. *Current Quality of Service (QoS) solutions ensure that the voice traffic receives a preferential treatment without a direct intervention of the callers. Conversely, there are some SIP (Session Initiation Protocol) extensions to let the users specify the QoS level of the sessions. However, they use a fixed list of QoS parameters and often coupled to particular technologies. This paper proposes an ontology-based solution to establish SIP sessions with QoS guarantees providing flexibility in terms of QoS parameters. The proposed solution was prototyped and tested in a Linux DiffServ domain and an Asterisk VoIP server.*

1. Introdução

Um dos principais requisitos dos serviços de Voz sobre IP (VoIP), e para vários outros serviços multimídia, é a garantia da Qualidade de Serviço (QoS). Provedores de Serviço de Rede (NSPs – *Network Service Providers*) que implantam soluções de QoS podem oferecer a seus clientes serviços de comunicação com garantias de desempenho de rede (expresso em termos de limites de vazão, atraso, taxa de perda de pacotes, e outros). Neste cenário, cliente e NSP firmam um Acordo de Nível de Serviço (SLA – *Service Level Agreement*) que contém a especificação dos limites de desempenho de rede para cada tipo de serviço ou tráfego importante para o cliente. Estas especificações são chamadas de Especificações do Nível de Serviço (SLSs - *Services Level Specification*).

Uma arquitetura de QoS muito conhecida é a Serviços Diferenciados (DiffServ) [Black 1998], que se baseia na classificação do tráfego em diferentes Classes de Serviço (CoS) oferecendo diferentes níveis de qualidade [Santi 2007]. Esta classificação é realizada nos roteadores de borda da rede e é feita com base nos SLSs negociados. Além

da classificação de pacotes, o roteador de borda realiza a marcação e o condicionamento do tráfego. Os roteadores de núcleo preocupam-se em oferecer o encaminhamento adequado aos pacotes dependendo da CoS em que eles foram classificados.

Em geral, os SLSs são estáticos, que são praticamente imutáveis durante o ciclo de vida de um SLA. Este tipo de SLS resulta em uma configuração dos roteadores de borda. Por exemplo, para um serviço de VoIP em uma rede DiffServ, um SLS estático resulta na configuração do(s) roteador(es) de borda(s) para classificar o tráfego VoIP em uma determinada CoS. Nesta situação, o nível de qualidade a ser oferecido para uma chamada VoIP independe da vontade dos usuários do serviço VoIP. Existem atualmente propostas permitindo a negociação dinâmica de SLSs, aonde os clientes podem criar, modificar e remover SLSs durante o ciclo de vida de um SLA. A negociação dinâmica de QoS oferece um maior dinamismo nas conexões, dando ao usuário maior poder de escolha no momento do estabelecimento da sessão: usuários de serviços de VoIP, TV sobre IP ou vídeo sob-demanda, poderiam negociar dinamicamente o nível de qualidade para suas sessões de comunicação.

Já existem alguns trabalhos ([Camarillo 2002], [Polk 2009], [Park 2007] e [Alexander 2006]) que utilizam os protocolos SIP (*Session Initiation Protocol*) [Rosenberg 2002] e SDP (*Session Description Protocol*) [Handley 2006] para a sinalização de sessões com qualidade explicitamente definida pelos usuários/aplicações. Grande parte destas propostas são orientadas a uma determinada solução de QoS (por exemplo, DiffServ). Além disso, nenhuma delas prevê flexibilidade em termos de parâmetros de QoS. Ou seja, elas geralmente definem um número fixo de parâmetros. Em [Vicente 2010], nós propomos um novo atributo SDP aonde a qualidade da chamada é descrita referenciando conceitos especificados na ontologia NetQoSOnt [Prudêncio 2009]. Esta ontologia fornece uma base para criar especificações de QoS em vários níveis da rede, relacionar e comparar níveis de qualidade entre si. Graças a NetQoSOnt, o usuário/cliente do serviço pode expressar suas necessidades em termos de parâmetros de Qualidade de Experiência (QoE), mais simples do que usar parâmetros de desempenho de rede. A base formal da NetQoSOnt permite ao NSP comparar automaticamente os parâmetros usados pelos clientes/usuários com os parâmetros de redes adotados pelo NSP utilizando a inferência em ontologias.

Este artigo apresenta uma solução para sinalização e instalação de sessões SIP com garantias de QoS utilizando a extensão SIP/SDP proposta em [Vicente 2010]. Ela permite que usuários de aplicações utilizando o protocolo SIP possam sinalizar a QoS para suas sessões. Esta solução se baseia no uso do protocolo Diameter [Calhoun 2003][Sun 2010] para a autenticação, autorização e contabilidade, bem como a instalação da QoS, e do protocolo NETCONF [Enns 2006] para a configuração dinâmica dos equipamentos de rede. A solução proposta foi prototipada e testada em uma estrutura de rede formada por roteadores Linux DiffServ e um serviço VoIP.

O restante deste artigo está organizado na forma que segue. A seção 2 apresenta a ontologia NetQoSOnt. A seção 3 revisa conceitos associados ao protocolo SIP e apresenta as propostas de sinalização de sessão SIP com QoS. A seção 4 introduz os protocolos Diameter e NETCONF. A seção 5 detalha a solução proposta para sinalização e instalação de sessões SIP com QoS. A seção 6 apresenta o protótipo implementado e os testes realizados. Conclusões e trabalhos futuros são apresentados na seção 7.

2. NetQoSOnt

Em diversas operações relacionadas ao gerenciamento de QoS é necessário um meio eficiente de especificar a qualidade. A grande diversidade de soluções de QoS utilizadas pelos NSPs, cada um adotando uma terminologia própria, torna difícil o desenvolvimento de soluções de QoS válidas em todos os cenários. Para resolver um problema análogo, vários trabalhos na área de *Web Services* adotam ontologias para realizar a especificação da QoS. Inspirado em *Web Services* semânticos, em [Prudêncio 2009] propomos uma ontologia para a especificação de QoS para serviços de rede, chamada NetQoSOnt. Esta seção revisa os principais conceitos desta ontologia.

Uma ontologia, na área da computação, compreende uma especificação explícita e formal da conceitualização de um domínio de interesse [Davies, 2006]. Nesta definição, o termo “formal” significa que a especificação pode ser processada automaticamente por computador. Basicamente, uma ontologia constitui-se de um conjunto de classes (representações concretas de conceitos), relações (ou propriedades), indivíduos (ou instâncias) e axiomas. O uso de ontologias está associado ao uso de motores de inferência, que podem inferir novas relações entre classes que antes não estavam codificadas na ontologia.

A NetQoSOnt é uma ontologia codificada em OWL 2.0 (a versão mais atual do padrão de codificação de ontologias da W3C), e especifica vários conceitos de base para a especificação de QoS em redes na forma de classes OWL. Dentre as principais classes destaca-se a *QoS*Spec, que permite descrever especificações de QoS. Esta classe tem como propriedade uma lista de parâmetros de qualidade, cada parâmetro é representado por uma subclasse de *Parameter*. Um parâmetro, por sua vez, pode ter uma ou mais medidas distintas (expressas em diferentes unidades), representadas por subclasses de *Measure*. Para refletir a existência de diferentes tipos de parâmetros nas diferentes camadas da rede, foi especificado o conceito de camada com a classe *Layer*. Inicialmente, NetQoSOnt segue o modelo TCP/IP, fornecendo subclasses de *Parameter* para quatro das camadas de rede (Enlace, Internet e Transporte), mais a camada Usuário. Esta última contendo especificações de QoS ao nível de usuário, os parâmetros de QoE (Qualidade de Experiência).

A Fig. 1 exemplifica o uso da NetQoSOnt na criação de uma especificação de QoS em nível de usuário utilizando o parâmetro MOS (*Mean Opinion Score*). O MOS é uma medida clássica de qualidade de voz, uma indicação numérica da qualidade subjetiva de voz, variando de 1 (qualidade baixa) a 5 (qualidade excelente). No exemplo da Fig. 1, a especificação da qualidade foi descrita na forma $MOS \geq 4.5$, expressando a qualidade mínima de voz que o sistema deveria garantir. A qualidade $MOS \geq 4.5$ é especificada através de uma subclasse de *QoS*Spec, chamada *MOS4.5-Spec*. Esta classe tem como propriedade *MOS4.5*, uma subclasse de *UserPerformanceParameter* utilizada para definir o parâmetro MOS. *UserPerformanceParameter* por sua vez é subclasse de *Parameter*. *MOS4.5* tem como relação uma subclasse de *Measure*, *MOS4.5Measure* que está relacionada ao intervalo “ ≥ 4.5 ”.

Em geral, uma determinada QoS em uma camada de rede implica, ou é equivalente, a outra QoS nas camadas inferiores ou superiores. Em NetQoSOnt, isso é representado usando as relações equivalência de classes em ontologias. Continuando o exemplo da Fig. 1, a classe *MOS4.5-Spec* tem uma relação de equivalência com a classe *QoS*Spec *MOS4.5IP*. Esta última especifica a qualidade equivalente na camada de rede

para $MOS \geq 4.5$ usando os parâmetros atraso (OWD – *One Way Delay*) e taxa de perdas de pacotes (PLR – *Packet Loss Rate*). Para simplificar a explicação sobre *MOS4.5IP*, a Fig. 1 apresenta apenas o limite de atraso que *MOS4.5* exige, que é $OWD \leq 150ms$. Para representar este atraso é criada uma subclasse de *OneWayDelay* *MOS4.5IP-OWD*, e *MOS4.5IP* é relacionada a *MOS4.5IP-OWD* através de *hasParameter*. O valor 150ms é representado por uma subclasse de *Measure*, *MOS4.5IP-OWD-Measure*, que está relacionada ao intervalo ≤ 150 e à unidade *millisecond*.

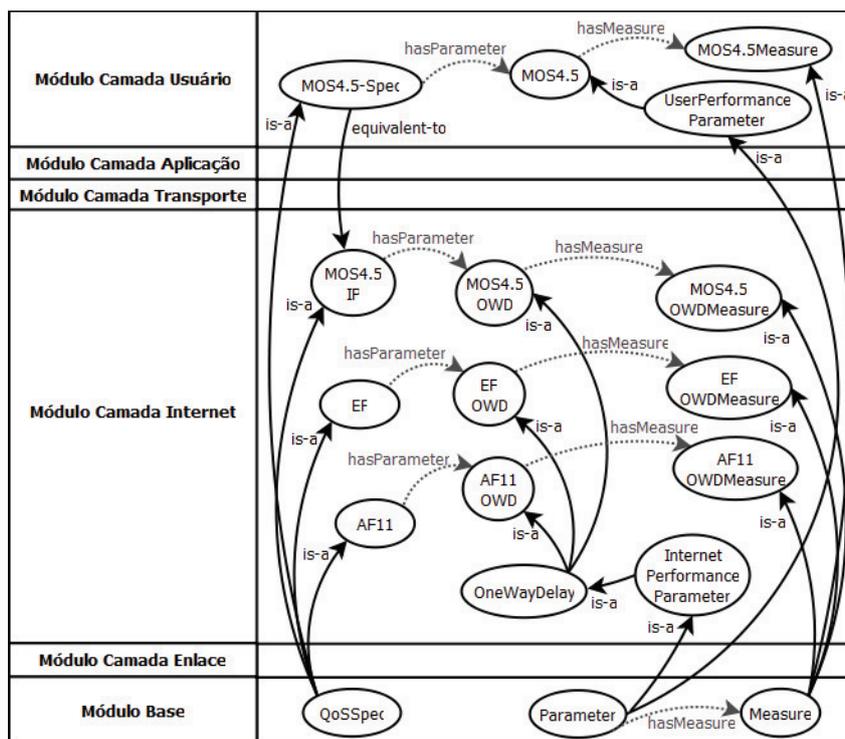


Figura 1. Exemplo de uso da NetQoSOnt.

A Fig. 1 apresenta também a especificação da QoS garantida por duas CoSs de um NSP hipotético, nomeadas com EF e AF11. No domínio do NSP é garantido que o tráfego na classe AF tenha um atraso máximo de 100ms. Essa CoS é representada em NetQoSOnt através de uma subclasse *QoSSpec EF*, e o parâmetro de atraso é representado de maneira análoga a realizada para a classe *MOS4.5IP*.

NetQoSOnt oferece meios para realização de comparações automatizadas entre especificações de QoS. Graças a modelagem das especificações na forma de classes OWL, a comparação entre especificações de QoS se resume a um problema de herança de classes. Através do uso de um motor de inferência, uma nova hierarquia das subclasses *QoSSpec* (que representam especificações de QoS) pode ser calculada, e a partir desta hierarquia é possível determinar se duas ou mais especificações de QoS estão relacionadas, e qual oferece melhor qualidade. Retomando o exemplo da Fig. 1, NetQoSOnt oferece os meios de comparar *MOS4.5* (uma especificação da camada de usuário) com *EF* (especificação da camada de rede). Isso é possível, pois a equivalência entre *MOS4.5* e *MOS4.5IP* faz com que *MOS4.5* herde o parâmetro de atraso de *MOS4.5IP*. Como os atrasos de *MOS4.5* e *EF* são definidos de maneira análoga, os intervalos de valores são comparados diretamente, e como $100 < 150$, a conclusão final do motor de inferência é que *EF* é subclasse de *MOS4.5*, e por isso possui uma qualidade melhor. Um sistema de gerência de QoS do NSP poderia utilizar esta

conclusão para determinar quais de suas CoSs atendem um pedido do usuário. No exemplo dado, um pedido de qualidade $MOS \geq 4.5$ poderia ser atendido com a CoS *EF*.

3. Protocolo SIP e a QoS

O SIP (*Session Initiation Protocol*) [Rosenberg 2002] é o protocolo mais utilizado na sinalização de serviços VoIP, sendo responsável pela negociação entre as partes envolvidas no estabelecimento de uma chamada VoIP.

3.1. Estabelecimento de sessão SIP

Esta seção apresenta o procedimento de estabelecimento e encerramento de uma sessão SIP entre dois agentes usuários. Para ilustrar este procedimento, considere os usuários Alice e Bob (nomes fictícios tipicamente utilizados em trabalhos da área) realizando uma chamada VoIP, cuja sinalização é ilustrada na Fig. 2a. Neste cenário, Alice e Bob utilizam o mesmo servidor VoIP. O processo de negociação de uma sessão SIP inicia pelo envio da mensagem *INVITE* por parte do agente usuário SIP de Alice para seu servidor SIP. A Fig. 2b apresenta a mensagem *INVITE* enviada por Alice, onde pode ser visto que o conteúdo desta mensagem é uma descrição dos parâmetros da sessão, usando o protocolo SDP. Os parâmetros SDP usados na Fig. 2b são: *v* indica a versão do protocolo SDP; *o* identifica a origem e a sessão; *s* define o assunto (nome da sessão); *t* define o tempo de partida e parada (zero significa permanente); *m* define uma mídia da sessão, onde é indicando a mídia, o protocolo de transporte, e a lista de codecs usados; e *a* define um atributo da mídia declarada anteriormente.

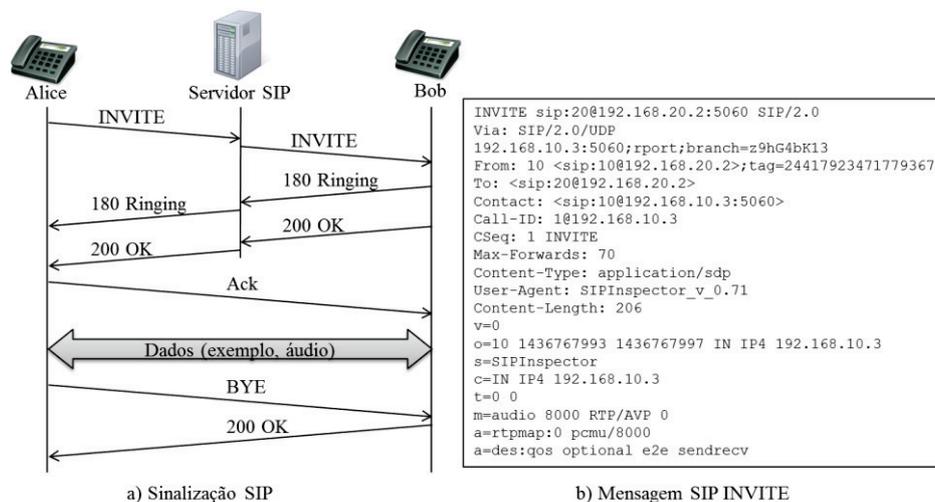


Figura 2. Estabelecimento de uma sessão SIP.

Recebendo a mensagem *INVITE*, o servidor VoIP usa o serviço de localização para determinar o endereço IP atual de Bob, reencaminhando em seguida o *INVITE* ao mesmo. Após receber o *INVITE*, o agente usuário de Bob irá chamar a atenção de Bob e enviar uma mensagem *180 Ringing*. No momento em que Bob atende a chamada, é enviada uma mensagem *200 OK* que é respondida por um *ACK*, estabelecendo finalmente a sessão SIP. Neste cenário não se considera a renegociação de parâmetros da sessão. Depois de estabelecida a sessão ocorre a transferência de pacotes de voz entre os agentes usuários de Alice e Bob. Quando Alice encerra a chamada, uma mensagem *BYE* encaminha o pedido de desconexão entre as partes, que é confirmada com a mensagem *200 OK*.

3.2. Negociação de Sessões SIP com QoS

Existem algumas RFCs que tratam de assuntos relacionados à QoS no estabelecimento de sessões SIP. A RFC3312 [Camarillo 2002] define como a QoS pode ser definida como uma das precondições da sessão SIP. A título de exemplo, na Fig. 2b o atributo do áudio *a=des:qos optional e2e sendrecv*, define que QoS é uma precondição opcional para o áudio. Mas a RFC 3312 não define que nível de qualidade deve ser garantido. Em seguida, a RFC 4566 [Handley 2006] definiu um atributo *quality* permitindo especificar um número inteiro para expressar a qualidade da codificação de áudio e vídeo. Mas não existe uma recomendação que defina claramente o significado destes valores inteiros. Posteriormente, a RFC5432 [Polk 2009] definiu novos atributos ao SDP permitindo expressar os mecanismos de QoS (como RSVP) desejados para a sessão, mas novamente não permite expressar o nível da qualidade para a sessão.

Além das RFCs citadas previamente, existem propostas de sinalização de QoS em sessões SIP. Em [Park 2007], os autores alteraram a semântica do atributo *quality* definido na RFC4566 para que seus valores correspondam as classes de QoS definidas na ITU-T Y.1541 [ITU-T 2002]. Mas como cada classe Y.1541 é destinada a um tipo de aplicação, no caso de serviços VoIP as chamadas seriam atendidas sempre com a mesma qualidade, impedindo que o usuário selecione a QoS para sua chamada. Outras duas propostas, [Alexander 2006] e [Polk 2008], não oferecem transparência de parâmetros de QoS, pois elas permitem expressar apenas o nível de QoS usando parâmetros de desempenho de redes WiMax e classes Diffserv, respectivamente.

Em [Vicente 2010] nós propomos uma extensão do SDP oferecendo flexibilidade em termos de parâmetros de QoS negociados nas sessões SIP. Para tal, foi proposta uma alteração na RFC3312 para que a precondição de QoS explicita a qualidade desejada para a sessão SIP. Mais especificamente, o valor da qualidade é definido como uma *URI-reference* que é o identificador de uma subclasse de *QoSSpec* da ontologia NetQoSOnt que especifica a QoS desejada para a sessão SIP. Aplicado ao exemplo da Fig. 2b, o atributo do áudio seria modificado para *a=des:qos http://voiporg.org/voipqos.owl#MOS4.5-Spec optional e2e sendrecv*. Este último define que a precondição de qualidade é *http://voiporg.org/voipqos.owl#MOS4.5-Spec*, opcional, fim-a-fim, para o tráfego de voz nas duas direções. O valor deste atributo é uma URI referenciando a qualidade $MOS \geq 4.5$ que foi publicada por uma suposta organização de recomendações para voip chamada de *voiporg*.

4. Protocolos Diameter e NETCONF

As soluções que suportam invocação explícita de QoS exigem mecanismos de AAA (*Authentication, Authorization and Accounting*) para verificação da identidade do usuário requerendo o serviço, autorização de uso do recurso e contabilidade deste uso [Royer 2009]. Além deste, é necessário um protocolo para configuração dos equipamentos de rede para que os tráfegos tenham o tratamento adequado. Esta seção apresenta os protocolos AAA Diameter [Calhoun 2003] e NETCONF [Enns 2006] utilizados na proposta deste artigo.

4.1. Protocolo Diameter

O protocolo de AAA Diameter foi concebido como um protocolo extensível e adaptável às necessidades das aplicações. As chamadas “aplicações Diameter” estendem o

protocolo de base Diameter adicionando novos comandos e/ou atributos adequados às necessidades das aplicações. Uma aplicação Diameter importante para este trabalho é a Diameter QoS definida na RFC5866 [Sun 2010], que descreve um arcabouço, mensagens e procedimentos para que elementos de rede interajam com servidores Diameter em procedimentos de alocação de recursos em redes. Nela são definidas quatro novas mensagens: *QoS-Authorization-Request* (QAR) e *QoS-Authorization-Answer* (QAA) para solicitar a autorização de serviços com QoS; e *QoS-Install-Request* (QIR) e *QoS-Install-Answer* (QIA) para a solicitação da instalação da QoS. A mensagem QAR inclui, dentre outros parâmetros: (i) o ID da sessão, (ii) o ID de autorização da aplicação, (iii) o host de origem, (iv) o domínio de origem, (v) o domínio de destino, (vi) o tipo de requisição de autorização (somente autorização, somente autenticação ou ambos), (vii) o host de destino, (viii) o nome do usuário e outros dados do usuário, e (ix) a especificação de QoS desejado. As mensagens do protocolo base Diameter também são usadas, em particular, as mensagens *Abort-Session-Request* (ASR) e *Abort-Session-Answer* (ASA) são usadas para abortar a sessão autorizada.

4.2. Protocolo NETCONF

Para a configuração dinâmica de equipamentos de rede para provimento de QoS é necessário a utilização de protocolos de gerenciamento e configuração. Um dos protocolos que foi muito adotado em diversos projetos de pesquisa na área de QoS é o COPS-PR [Chan 2001], que é usado para controlar as políticas aplicadas nos dispositivos de rede. Nos últimos anos, a IETF padronizou o protocolo NETCONF [Enns 2006], uma alternativa mais simples para instalar, manipular e remover a configuração de dispositivos de rede. NETCONF adota um modelo de comunicação baseado em chamada de procedimento remoto (RPC – *Remote Procedure Call*) e utiliza XML para codificação. No NETCONF, os elementos XML que simulam o paradigma RPC são:

- `<rpc>`, usado pelo gerente para encaminhar um pedido ao agente. Ele contém um identificador da mensagem `<message-id>` e a operação solicitada;
- `<rpc-reply>`, que é usado pelo agente para responder o pedido do gerente. Ele contém o mesmo `<message-id>` do pedido e o resultado da operação.

As operações possíveis são: `<get>`, `<get-config>`, `<edit-config>`, `<copy-config>`, `<delete-config>`, `<lock>`, `<unlock>`, `<close-session>`, `<kill-session>`. Por exemplo, `<edit-config>` permite encaminhar uma nova configuração no equipamento de rede, usando um arquivo local, arquivo remoto ou em linha. `<delete-config>` permite eliminar uma configuração. Para o transporte das mensagens são considerados os protocolos SSH (*Secure Shell*), SOAP (*Simple Object Access Protocol*), BEEP (*Blocks Extensible Exchange Protocol*) e TLS (*Transport Security Layer*).

5. Suportando a Sinalização e Instalação de QoS em Sessões SIP

Esta seção apresenta a solução proposta para a sinalização e instalação de QoS em Sessões SIP sobre domínios de rede suportando a arquitetura Serviços Diferenciados (DiffServ), como solução de QoS. A Fig. 3 apresenta os principais componentes da solução proposta em um cenário simples. Neste cenário, dois usuários, chamados Alice e Bob, estabelecem uma sessão SIP, sendo que os dois usuários são autorizados por clientes de um único NSP que gerencia o domínio DiffServ. Além disso, os usuários

estão registrados no mesmo servidor VoIP. As seções que seguem apresentam os componentes da solução proposta.

5.1. Agentes Usuários SIP

Os agentes usuários SIP são utilizados para realizar a chamada VoIP. Para realizar a sinalização da QoS da sessão SIP, estes agentes devem ser suportar a extensão do protocolo SDP proposta. Além disso, o agente usuário deve oferecer uma interface para que o usuário indique a qualidade desejada para a chamada (que pode ser feita logo antes de realizar uma chamada, ou então na agenda pessoal, onde é registrada a qualidade desejada para as chamadas a cada destino cadastrado). Para especificar a QoS, o usuário poderá utilizar qualquer parâmetro de QoS, subjetivo ou objetivo, desde que publicada em uma ontologia que reuses a NetQoSOnt. Como ilustrado na seção 2, uma organização poderia publicar o parâmetro de QoS MOS que poderia ser referenciado pelo usuário para especificar a qualidade desejada. Estes parâmetros de QoS poderiam ser estaticamente definidos no software do agente usuário SIP ou importados via a indicação da URI da ontologia que define estes conceitos.

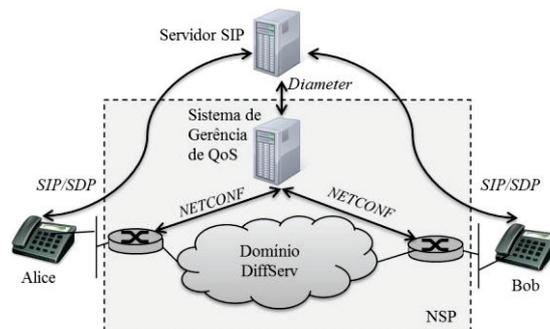


Figura 3. Componentes do Sistema VoIP com QoS

5.2. Servidor SIP

O servidor SIP, ou PBX-IP, é responsável pelo gerenciamento das chamadas VoIP, incluindo serviços para registro de usuários, de AAA, e de encaminhamento de chamadas. Para atender solicitações de sessões SIP com QoS, o servidor SIP deve ser capaz de interpretar a extensão do SDP proposta e interagir com o sistema de gerência de QoS.

Em termos de serviços de AAA, nesta proposta foi adotada a aplicação do protocolo AAA Diameter para QoS definida na RFC 5866. A Fig. 4 apresenta a sinalização envolvida no estabelecimento e encerramento de uma sessão SIP com QoS, novamente considerando o cenário apresentado na Fig. 3. As principais operações relacionadas são as seguintes: autorização do serviço com QoS, instalação da QoS e encerramento da chamada. O processo de autorização do serviço com QoS inicia no recebimento pelo servidor SIP da mensagem INVITE contendo a especificação da QoS desejada. Neste momento, o servidor SIP encaminha uma mensagem QAR com a identificação do usuário, a qualidade desejada e outras informações extraídas do INVITE necessárias no processo de autorização da chamada. Na recepção de uma resposta de autorização QAA positiva, o servidor SIP deve verificar se a QoS é obrigatória (*mandatory*) ou desejada (*optional*). Caso seja obrigatória, a instalação da QoS deve ser solicitada imediatamente (com a mensagem QIR), e a continuação da chamada dependerá de um QIA com sucesso. A Fig. 4 apresenta um cenário onde a

garantia de qualidade é desejada (não obrigatória), opção mais adequada para VoIP. Neste último caso, a chamada prossegue normalmente após a autorização do pedido e a instalação da QoS ocorre apenas na chegada no servidor SIP da mensagem *200 OK*, quando o usuário Bob atende o fone SIP. Neste momento, é encaminhada uma mensagem Diameter QIR ao sistema de gerenciamento de QoS para a instalação da QoS (termo usado pela RFC5866 que corresponde a realizar as ações necessárias para garantir a QoS). A QIR contém as informações extraídas do SDP que descrevem a sessão (como endereços e portas de envio e recepção das mídias). O encerramento da sessão ocorre quando o servidor SIP recebe a mensagem SIP BYE, quando o servidor SIP encaminha uma mensagem Diameter ASR para abortar a sessão com QoS.

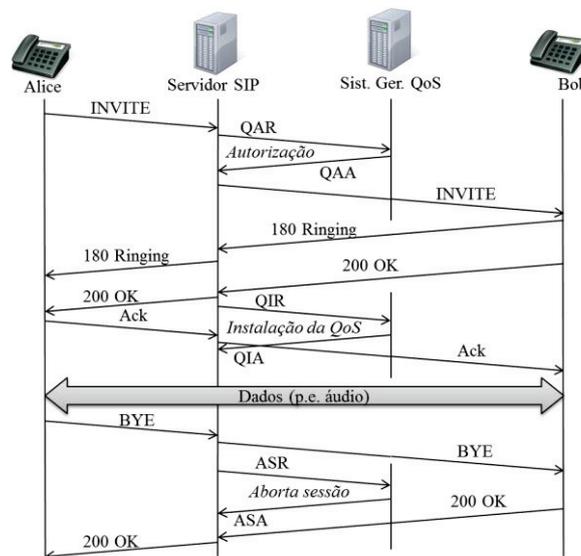


Figura 4. Sinalização sessão SIP com QoS

Em um cenário mais complexo, Alice e Bob poderiam ser usuários de diferentes servidores VoIP. Neste caso, os agentes usuário de Alice e Bob deveriam interagir com os seus respectivos Servidores SIP, que por sua vez devem interagir com seus sistemas de gerência de QoS.

5.3. Sistema de Gerência de QoS

O sistema de gerência de QoS (podendo aqui ser chamado também de *Bandwidth Broker* ou Gerenciador de Recursos) realiza o gerenciamento de uso dos serviços com QoS em um domínio DiffServ. Como visto na Fig. 5, este sistema é composto de quatro módulos que interagem com os atores envolvidos no gerenciamento de recursos de rede.

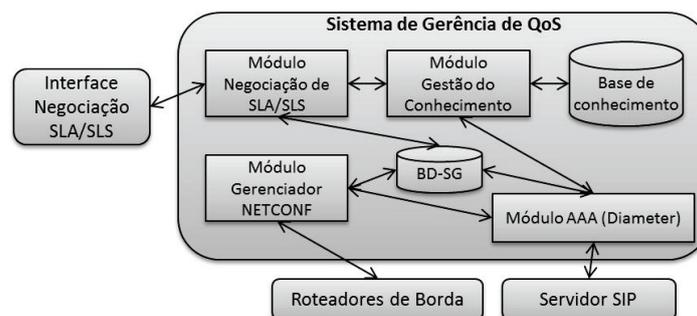


Figura 5. Sistema de Gerência de QoS

Módulo de Negociação de SLA/SLS

O módulo de negociação de SLA/SLS é o lado servidor de um cliente http permitindo ao cliente do NSP negociar SLA/SLS via um navegador Web. Esta interface web na realidade permite ao cliente, de maneira implícita, especificar um SLA, composto de diversos SLSs, usando uma ontologia de SLA que reusa a NetQoSOnt. Esta interface também deve permitir a definição de perfis de usuários e o cadastramento de usuários habilitados a utilizar os serviços. O modelo de perfil de usuário adotado é baseado na ontologia de perfil de usuário proposto em [Royer 2008]. Este artigo tem por objetivo apresentar o processo de sinalização de QoS durante o estabelecimento de sessões SIP, não são detalhados aqui aspectos do uso da ontologia NetQoSOnt na negociação de SLA/SLSs, nem no processo de autorização do serviço com base nos perfis dos usuários.

O banco de dados BD-SG mantém informações necessárias ao gerenciamento de alguns serviços do sistema. Em particular, ele mantém informações dos SLAs/SLSs acordados, dos usuários autorizados a utilizar o serviço, informações das sessões SIP ativas, entre outros.

Módulo de Gestão do Conhecimento

Este módulo é responsável para manutenção e acesso as informações mantidas na base de conhecimento. Esta base de conhecimento mantém as seguintes informações:

- Especificações de QoS do Provedor: ela mantém a definição das CoSs do provedor e as especificações de QoS dessas CoSs que definem as garantias de atraso, taxa de perda de pacotes, variação de atraso de cada CoS. Todas estas são modeladas como classes *QoSSpec* da ontologia NetQoSOnt. Como por exemplo, as classes EF e AF11 apresentadas na Fig. 1.
- Especificações de SLAs/SLSs: produzidas pelo Módulo de Negociação de SLA/SLS com base na ontologia de SLA/SLS.
- Especificações de QoS usadas nas chamadas SIP: são especializações de *QoSSpec* geradas pela sinalização de sessões SIP com QoS.
- Conceitos representando grupos de usuários e usuários autorizados a utilizar o serviço, conforme a definição Perfis de usuários proposto em [Royer 2008].

Este módulo também é responsável por todo processo de inferência acerca das informações contidas na base de conhecimento. Com isto, é possível realizar o mapeamento da especificação de QoS de alto nível em uma das classes de serviço do NSP, conforme detalhado em [Prudêncio 2009]. O processo de inferência é invocado em dois momentos:

- Na negociação de um SLS: o módulo de Gestão do Conhecimento oferece a função de verificação de quais CoS atendem a solicitação do usuário (como exemplificado na seção 2).
- No estabelecimento da sessão SIP com QoS: como usuário pode especificar a qualidade usando parâmetros de alto nível, o Módulo de Gestão do Conhecimento também permite verificar se existe um SLS negociado que cubra a solicitação do usuário.

O módulo de Gestão do Conhecimento mantém na base de conhecimento todas as conclusões acerca de novas especificações de QoS que sejam negociadas pelo sistema de gerência ou utilizadas na sinalização de sessões SIP com QoS. Toda vez que

é realizada uma nova negociação ou sinalização, primeiramente a base é consultada para verificar se a especificação de QoS já é conhecida (se já foi verificado qual CoS atende a solicitação). Caso não for conhecido, um processo de inferência é realizado com os novos conceitos, como descrito na seção 2.

Módulo AAA

Nesta proposta, considera-se o uso da aplicação QoS do protocolo de AAA Diameter proposta por [Sun 2010] e o modelo de perfil de usuário proposto em [Royer 2008]. Portanto, o módulo AAA é o lado servidor do protocolo Diameter usado pelo servidor SIP para os procedimentos de autenticação, autorização, instalação e liberação da QoS vistos na seção 5.2.

No momento da autorização da chamada SIP com QoS, o módulo de AAA deve verificar se existe um CoS que atenda a solicitação do usuário, se esta chamada se enquadra em uma das SLSs negociadas pelo cliente, se o usuário é autorizado a utilizar o serviço especificado pelo SLS, e finalmente verificar a disponibilidade de recursos de rede para atender a solicitação do cliente. Para isto, este módulo utiliza informações mantidas no BD-SG e interage com o módulo Gestão do Conhecimento para determinar relações de equivalência entre especificações de QoS e CoS.

Módulo Gerenciador NETCONF

Este módulo atua como um gerente NETCONF, responsável pelo gerenciamento dos roteadores de borda da rede DiffServ. Em particular, ele é responsável pela alteração das regras de configuração dos roteadores de borda quando da instalação da QoS para uma chamada SIP com QoS. Também é responsável pela remoção da regra de classificação ao final da sessão. Estas duas operações são solicitadas pelo módulo AAA.

No momento da instalação da QoS, o módulo AAA informa ao gerenciador de NETCONF a QoS desejada pela sessão SIP. Esta especificação de QoS pode ser feita usando parâmetros de QoE ou outros.

6. Protótipo Desenvolvido e Experimentações

A fim de testar a solução proposta, em termos funcionais e de desempenho, parte das funcionalidades da solução proposta foi implementada em um protótipo. Note que o objetivo não foi o de implementar todas as funcionalidades dos protocolos NETCONF e Diameter utilizados na solução, mas de avaliar a eficiência do uso de uma abordagem semântica para especificar a QoS e validar o mapeamento desta especificação em parâmetros de configuração de rede. Além disso, este estudo compara a solução proposta com um serviço SIP tradicional sem QoS. Não foi possível comparar o desempenho da presente proposta com outras soluções de sinalização SIP com QoS (como, [Camarillo 2002], [Polk 2009], [Park 2007] e [Alexander 2006]), pois os autores deste trabalho não entraram implementações utilizando estas outras extensões do SIP/SDP.

A Fig. 8 apresenta a estrutura de teste implantada. Ela é composta dois roteadores DiffServ Linux, dois agentes usuários SIP, um sistema de gerência de QoS e um servidor SIP. Em relação aos agentes usuários SIP, foram utilizados a ferramenta SIP Inspector (2010) e o softphone X-Lite (2010). O SIP Inspector foi utilizado, pois ele

permite a edição dos campos da mensagem INVITE, permitindo a inclusão do atributo necessário à solicitação de um serviço com QoS.

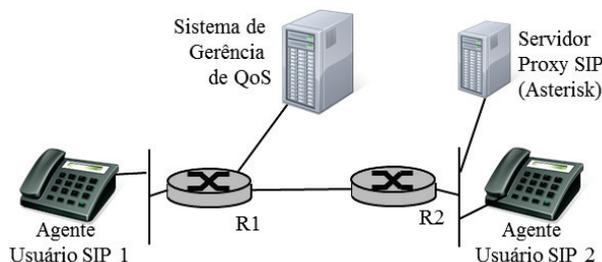


Figura 8. Estrutura de Teste

O servidor VoIP instalado foi o PBX IP Asterisk (2010), que foi estendido para interpretar o novo atributo SDP proposto e o uso dos serviços de AAA via um cliente Diameter. Este cliente foi desenvolvido na linguagem C, e é responsável pelas solicitações de autenticação, autorização, instalação e encerramento de sessões com QoS. Ele implementa as funções mínimas necessárias à realização do teste.

Algumas das funcionalidades básicas do sistema de gerência foram prototipadas:

- O módulo de Gestão do Conhecimento foi desenvolvido em Java utilizando a API OWL 2.0, sendo que o motor de inferência utilizado foi o Pellet (2010). A especificação de QoS das CoSs do provedor é a mesma da seção 2.
- Para simular o funcionamento do gerente NETCONF foi implementado um script Python usando a biblioteca XML-RPC, com recursos mínimos para suportar RPC codificadas em XML. Este módulo simula o uso do protocolo NETCONF com suporte a operações de edição de regras de classificação nos roteadores de borda DiffServ, bem como operação para eliminação das regras.
- Em vez de serem instalados agentes NETCONF nos roteadores de borda foi executado um script Python também usando a biblioteca XML-RPC e o banco Sqlite. Este sistema simula o funcionamento de um agente NETCONF sobre o sistema Linux.

Implementado protótipo, foram realizados alguns testes para avaliar a solução em termos funcionais e de desempenho. Foram estabelecidas 20 sessões SIP iniciadas pelo agente usuário 1 (SIP Inspector), sendo que 10 chamadas foram realizadas sem a pré-condição de QoS e 10 chamadas com negociação da QoS. A QoS solicitada foi a de $MOS \geq 4.5$ (<http://voiporg.org/voipqos.owl#MOS4.5-Spec>), e conforme visto na seção 2, a CoS inferida para atender a chamada será EF. O codec G.711 com tamanho de pacote de voz de 20ms foi adotado em todas as chamadas.

Durante a realização das chamadas, foram realizadas capturas de tráfego no agente usuário SIP 1, usando o analisador de protocolos Wireshark (2010). A partir dele foi possível medir os atrasos médios nas sinalizações das chamadas sem e com sinalização da QoS. Os atrasos médios entre a solicitação da chamada SIP até a recepção da mensagem SIP *180 Ringing* pelo agente usuário SIP 1 foram os seguintes: 7ms para chamadas sem QoS; 578ms para chamadas com QoS usando especificações de QoS não contidas na base de conhecimento do NSP; 132ms para chamadas com QoS usando especificações de QoS contidas na base de conhecimento do NSP.

Nos experimentos observou-se um acréscimo de 571ms de atraso na sinalização SIP para chamadas com QoS usando especificações de QoS não contidas na base de

conhecimento do NSP. Neste cenário, o usuário utilizaria um conceito não conhecido previamente pelo NSP. Portanto, o NSP deve importar *http://voiporg.org/voipqos.owl* e realizar o processo de inferência para avaliar qual CoS atende a solicitação do usuário. Apesar de 571ms ser um atraso computacional não desprezível, considera-se aqui que no contexto de estabelecimento de chamada este atraso é quase imperceptível pelos usuários do serviço. Note que esta situação ocorre apenas no primeiro momento em que um usuário do sistema solicitar uma qualidade que não tenha ainda sido solicitada. Quando ele ou outro usuário utilizar a mesma especificação de qualidade, não será mais necessária a importação dos conceitos, nem a realização da inferência. Para este cenário, observou um acréscimo de apenas 125ms em relação a chamadas sem QoS.

Para avaliar o atraso da instalação da QoS após o atendimento do telefone, foi medido no agente usuário SIP 1 o número de pacotes de voz que chegaram no destino antes da inclusão da nova regra de classificação dos pacotes nos roteadores R1 e R2. Foi verificado o valor de um pacote de voz em todas as medidas, que corresponde a um tempo total de voz de 20ms. Isto quer dizer que após 20ms de comunicação, o tráfego será marcado com a classe negociada. Esta medida demonstra que o usuário não deverá perceber esta não exigência momentânea da qualidade solicitada.

Para medir o impacto na qualidade da voz na troca da classificação dos pacotes de voz, foi medido o tempo entre o último pacote marcado com BE e o primeiro marcado com EF. Em condições ideais, este atraso seria na ordem de 20ms, e o valor médio medido foi de 24ms e esta variação de atraso está dentro da média da chamada.

Considerando a aplicação da solução proposta em um domínio real de rede, deve-se também considerar os atrasos de rede do domínio e os atrasos da importação das ontologias de QoS.

7. Conclusões

Neste artigo propusemos uma solução de gerenciamento de QoS em sistemas SIP utilizando o protocolo Diameter, para autenticação, autorização e contabilidade, e o protocolo NETCONF para configuração de equipamentos de rede. A negociação de sessões SIP com garantias de QoS permite ao usuário selecionar a qualidade desejada para as suas chamadas. A solução proposta foi prototipada e testada em uma rede Linux DiffServ com um serviço VoIP. As medidas realizadas demonstraram a aplicabilidade da solução em ambientes reais.

Como trabalho futuro, pretende-se continuar o desenvolvimento de todas as funcionalidades da solução proposta, e a aplicação desta solução em um ambiente real. Outro aspecto a ser investigado é o uso da proposta em redes multidomínio heterogêneos.

Referências

- Alexander, M. and Suppan, P. (2006) "An Architecture for SDP-based Bandwidth Resource Allocation with QoS for SIP in IEEE 802.16 Networks". 2nd Int. Workshop on QoS & Security for Wireless and Mobile Networks, p. 75-82.
- Asterisk. (2010) "The Open Source PBX & Telephony Platform". URL: <http://www.asterisk.org/>. Acessado em Dezembro, 2010
- Black, D. et al. (1998) "An Architecture for Differentiated Services", RFC 2475.

- Calhoun, P. et al. (2003) "Diameter Base Protocol", RFC 3588.
- Camarillo, G., Marshall, W., and Rosenberg, J. (2002) "Integration of Resource Management and Session Initiation Protocol (SIP)", RFC 3312.
- Pellet (2010). Pellet: OWL 2 Reasoner for Java. URL: <http://clarkparsia.com/pellet>.
- Chan, K. et al. (2001) "COPS Usage for Policy Provisioning (COPS-PR)", RFC 3084.
- Davies, J., Studer, R. e Warren, P. (2006). Semantic Web Technologies Trends and Research in Ontology-based Systems, John Wiley & Sons Ltd.
- Enns, R. (2006) "NETCONF Configuration Protocol", RFC 4741.
- Handley, M. et al. (2006) "SDP: Session Description Protocol", RFC 4566.
- ITU-T. (2002) "ITU-T Recommendation Y.1541, Network Performance objectives for IP-based services", Telecommunication Standardization Sector of ITU.
- Park, H.J. et al. (2007) "QoS Negotiation for IPTV Service using SIP. 9th Int. Conf. on Advanced Communication Technology", p. 945-948.
- Polk, J. (2008) "Configuring the Differentiated Services Codepoint of Session Description Protocol Established Media Streams", IETF Draft, draft-polk-mmusic-dscp-attribute-02.
- Polk, J., Dhesikan, S., and Camarillo, G. (2009) "Quality of Service (QoS) Mechanism Selection in the Session Description Protocol (SDP)", RFC 5432.
- Prudêncio, A.C., Willrich, R., Tazi, S. and Diaz, M. (2009) "Quality of Service Specifications: A Semantic Approach". 8th IEEE International Symposium on Network Computing and Application, p. 219-226.
- Rigney, C. et al. (2000) "Remote Authentication Dial In User Service", RFC 2865.
- Rosenberg, J. et al. (2002) "SIP: Session Initiation Protocol", RFC 3261, Internet Engineering Task Force.
- Royer, J., Willrich, R. and Diaz, M. (2008) "User Profile-Based Authorization Policies for Network QoS Services". 7th IEEE Int. Symp. on Network Computing and Applications (NCA), p. 68-75.
- Santi, J., Fonseca, N. L. S. and Lima, M. M. A. E. (2007) "Projeto de Controladores Ótimos para Redes com Produto Banda-Atraso Elevado". XXV Simpósio Brasileiro de Redes de Computadores (SBRC 2007).
- SIP Inspector. (2010) URL: <http://sourceforge.net/projects/sipinspector/>. Acessado em Dezembro, 2010
- Sun, D. et al. (2010) "Diameter Quality of Service Application", RFC 5866, Internet Engineering Task Force.
- X-Lite. (2010) URL: <http://www.counterpath.com>. Acessado em Dezembro, 2010.
- Vicente, L.H. (2010) "Estabelecimento de Sessões SIP com garantias de QoS". Dissertação de Mestrado do Programa de Pós-Graduação em Ciência da Computação da UFSC.
- Wireshark. (2010) URL: <http://www.wireshark.org/>. Acessado em Dezembro, 2010.