UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

JÚLIO CÉSAR ROYER

AUTORIZAÇÃO DE SERVIÇOS COM GARANTIAS DE QOS BASEADA EM PERFIS DE USUÁRIO

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação.

Prof. Dr. Roberto Willrich

Florianópolis
2008

Autorização de Serviços com Garantias de QoS baseada em Perfis de Usuário

Júlio César Royer

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, Área de Concentração Sistemas de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

_____

Prof. Dr. Frank Augusto Siqueira
Coordenador do Curso de Pós-Graduação em Ciências da Computação

Banca Examinadora:

_____

Prof. Roberto Willrich, Dr.
Presidente

_____

Prof. Renato Fileto, Dr.

_____

Prof. Luiz Fernando Rust da Costa Carmo, Ph.D.

_____

Prof. Jean Marie Farines, Dr.

EPÍGRAFE

“*Um homem pode, de fato, ser considerado alguém que gosta de aprender se é consciente, ao longo de um dia, sobre aquilo que ele não sabe e se nunca esquece, ao longo de um mês, aquilo que já dominou*.”
Os Analectos, Confúcio.

DEDICATÓRIA

À comunidade docente,
pelo abnegado esforço na construção de um mundo melhor
através da instrução, convívio e auxílio na formação
de caráter dos discentes-cidadãos.

AGRADECIMENTOS

À minha esposa Valdirene e aos meus pais Avelino e Romana,
pela paciência e apoio durante a realização deste trabalho;

Ao meu orientador Dr. Roberto Willrich,
pela confiança em mim depositada, pelo suporte e orientações constantes e precisas, e por sua amizade.

Aos meus sócios Arthur, Flávio e Otávio,
pela compreensão e pelo esforço redobrado que minha ausência lhes impôs;

Ao meu coordenador Ms. Carlos Kaneko,
pelo incentivo e apoio para a liberação das atividades docentes durante a realização das disciplinas.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

GLOSSÁRIO

AAA *Authentication, Authorization and Accounting*, um elemento que realiza os serviços de autenticação, autorização e contabilidade.

AAAC *Authentication, Authorization, Accounting and Charging*, uma extensão do AAA que inclui também o processamento do faturamento.

ABNF *Augmented Backus-Naur Form*, uma metalinguagem usada para definir formalmente protocolos de comunicação bidirecionais.

AM *Access Mediator*, elemento da arquitetura CADENUS que faz a mediação do acesso de um usuário a uma rede que oferece serviços com QoS.

AS *Autonomous System*, ou sistema autônomo, é um domínio de uma rede controlada por uma organização, usualmente um provedor de serviços de rede sobre IP.

ASIG *Application Signalling*, é um módulo do EuQoS responsável pelo controle de sinalização a nível de aplicação, localizado no terminal do usuário.

ASM *Application Specific Module*, ou módulo específico de aplicação, é um elemento da arquitetura genérica de AAA que gerencia os recursos de uma rede, e programa os equipamentos de rede.

A-SSN *Application-level Signaling and Service Negotiation*, ou Sinalização e

Negociação de Serviço a nível de Aplicação, é o módulo do sistema EuQoS responsável por fazer o controle de sinalização e negociação de serviço fim-a-fim a nível de aplicação. Está localizado no servidor EuQoS do NSP, e se comunica com o ASIG do terminal do usuário local e com o A-SSN do AS do usuário remoto.

AVP *Attribute-Value Pair,* ou par atributo-valor, usado pelos protocolos Radius e Diameter para transportar a informação entre o cliente e o servidor.

BB *Bandwidth Broker*, um elemento central de uma arquitetura de QoS que faz o gerenciamento de largura de banda em um domínio de uma rede IP com DiffServ.

BQCM *Basic QoS Control Module*, um sub-módulo do QCM do EuQoS, obrigatório e responsável pelo serviço de gerenciamento de sesssão.

CAC *Connection Admission Control*, o serviço de controle de admissão de conexões em uma rede, para evitar a sobrecarga.

CADENUS *Creation And Deployment of ENd-User Services in IP Premium Networks,* um projeto suportado pela União Européia para definir uma arquitetura de QoS focada na definição de serviços com QoS sobre redes IP.

CHAR *Charging*, é um módulo do EuQoS responsável pelo faturamento.

COPS *Common Open Policy Service*, é um protocolo simples de pergunta e

resposta usado para a troca de informações de políticas entre PDP e PEP.

CoS *Class of Service*, ou Classe de Serviço, em redes com DiffServ.

DIAMETER Protocolo de AAA, sucessor do RADIUS.

DiffServ *Differentiated Services*, ou Serviços Diferenciados, é uma arquitetura de QoS sobre IP proposta pelo IETF, que se baseia no tratamento diferenciado para cada Classe de Serviço, ou CoS.

E2E CAC *End-to-End Connection Admission Control*, módulo do gerenciador de recursos do EuQoS responsável pelo controle das admissões de novas conexões para os EQ-paths fim-a-fim.

EQ-BGP *End-to-end Qos Path*, ou caminho fim-a-fim com QoS. Trata-se de um caminho fim-a-fim com suporte a uma determinada CoS, no EuQoS.

EQ-path *End-to-end Qos Path*, ou caminho fim-a-fim com QoS. Trata-se de um caminho fim-a-fim com suporte a uma determinada CoS, no EuQoS.

EQ-PIB *E2E QoS Policy Information Base*, é a base de informações de políticas de gerenciamento independentes de tecnologia que especificam as regras de reserva de recursos para as conexões fim-a-fim com QoS no EuQoS.

EQ-SIP Protocolo derivado do SIP (*Session Initiation Protocol*), usado no estabelecimento de sessões, estendido para suportar requisições de QoS, usado na arquitetura EuQoS.

EuQoS Acrônimo para "*End-to-end Quality of Service support over*

heterogeneous networks", ou suporte de qualidade de serviço fim-a-fim sobre redes heterogêneas, é um projeto suportado pela União Européia para definir uma arquitetura de QoS sobre redes heterogêneas, em larga escala, sobre a Internet, considerando múltiplos ASs.

FO *Foreign Organization*, ou Organização Estrangeira, é usado para referenciar uma organização diferente daquela que o usuário possui um contrato de serviço (a UHO, ou *User Home Organizarion*).

IANA *Internet Authority Number Assignment*, organização responsável pelas atribuições de números na Internet, tais como números de portas e endereços IP.

IETF *Internet Engineering Task Force*, organização responsável pelas padronizações de protocolos para a Internet.

IntServ *Integrated Services*, ou Serviços Integrados, é uma arquitetura de QoS sobre IP proposta pelo IETF, baseada na reserva de recursos de rede para cada fluxo, individualmente.

IPDR *Internet Protocol Detail Record*, um padrão mantido pelo TM Fórum, que define o formato para a especificação de detalhes de registros de uso de serviços baseados em IP.

MMS *Monitor and Mesurement System*, é o módulo do gerenciador de recursos do EuQoS responsável pelas medições e monitoração do uso dos recursos de rede de um AS.

NAS *Network Access Server*, ou servidor de acesso à rede.

NSP *Network Service Provider*, ou provedor de serviços de rede, é a empresa que oferece o serviço de conectividade.

PDP *Policy Decidion Point*, ou ponto de decisão de política, é o elemento da rede que processa as regras e decide se uma requisição será aceita ou não.

PEP *Policy Enforcement Point*, ou ponto de aplicação de política, é o elemento da rede que aplica as decisões tomadas pelo PDP.

PR *Policy Repository*, ou Repositório de Políticas, é a entidade de um sistema gerenciamento de redes baseado em políticas que armazena as regras sobre as quais o sistema tomará as decisões de gerenciamento.

QAA *QoS Authorization Answer*, é uma mensagem do RADIUS ou do DIAMETER, que contém uma resposta a uma requisição de QoS (QAA).

QAR *QoS Authorization Request*, é uma mensagem do RADIUS ou do DIAMETER que contém uma requisição de QoS.

QCM *QoS Control Module*, é um módulo do EuQoS localizado no equipamento do usuário, responsável pelo controle de QoS.

QoS *Quality of Service*, Qualidade de Serviço.

QoSR *Quality of Service Routing*, ou Roteamento com Qualidade de Serviço.

RA *Resource Allocator*, ou Alocador de Recursos, é o módulo do EuQoS

responsável por alocar recursos para as requisições de QoS aceitas, no nível dependente de tecnologia (TD).

RA-COPS *Resource Allocator – Common Open Policy Service*, é o módulo do Alocador de Recursos do EuQoS responsável pela comunicação com o RM, através do protocolo COPS.

RADIUS *Remote Authentication Dial-In User Services*, protocolo de AAA criado para autenticação de acesso discado.

RDF *Resource Description Framework*, é uma família de especificações de modelos de meta-dados, sobre XML, mantido pelo W3C, usado na especificação de ontologias.

RFC *Request for Comment*, documento usado na definição de um padrão para a Internet, pelo IETF.

RM *Resource Mediator* (no CADENUS), ou *Resource Manager* (no EuQoS), é um elemento da arquitetura de QoS que é responsável pelo gerenciamento dos recursos de transmissão da rede, fazendo o controle para evitar a sobrecarga.

RM-COPS *Resource Manager Common Open Policy Service*, é o módulo do gerenciador de recursos (camada independente de tecnologia) do EuQoS responsável pela comunicação com o alocador de recursos (camada dependente de tecnologia), para a reserva de recursos para uma sessão com QoS aceita.

RM-DB *Resource Manager Data Base*, é a base de dados do gerenciador de recursos do EuQoS.

RM-SSN *Resource Manager Signalling and Service Negotiation*, é o módulo responsável pela sinalização e negociação de serviços do gerenciador de recursos do EuQoS.

SAAA *Security, Authentication, Authorization and Accounting*, é o elemento da arquitetura do EuQoS responsável pelo gerenciamento de segurança, autenticação de usuários, autorização e contabilidade de requisições de serviços com QoS.

SE *Service Equipment*, ou Equipamento de Serviço, em um sistema gerenciamento de redes baseado em políticas, referencia qualquer equipamento de rede onde são aplicadas as regras de políticas de gerenciamento.

SLA *Service Level Agreement*, ou acordo de nível de serviço. Documento que especifica detalhes administrativos de um contrato de prestação de serviço.

SLS *Service Level Specification*, ou especificação de nível de serviço. Documento que especifica tecnicamente os parâmetros de desempenho de um serviço contratado.

SM *Service Mediator, ou Mediador de Serviços, é o elemento da arquitetura do CADENUS que oferece os serviços ao AM e interage com o RM para*

obter os recursos de rede necessários.

SPARQL *Simple Protocol and RDF Query Language*, ou Protocolo e Linguagem de Consulta RDF Simples, é uma linguagem mantida pelo W3C, que permite realizar consultas sobre bases RDF de maneira similar a consultas SQL.

TD *Technology Dependent*, ou Dependente de Tecnologia, é o nível da arquitetura EuQoS que trata do controle dos recursos de QoS de acordo com a tecnologia utilizada.

TERO *Traffic Engeneering and Resource Optimization*, é o módulo do gerenciador de recursos do EuQoS responsável pela engenharia de tráfego e otimização de recursos, através da configuração das rotas com QoS interdomínios e alocação de recursos ao longo do caminho.

TI *Technology Independent*, ou Independente de Tecnologia, é o nível da arquitetura EuQoS que trata do controle dos recursos de QoS independente do tipo de tecnologia utilizada.

TI Tecnologia da Informação, no exemplo ilustrativo, refere-se ao nome de um departamento da organização.

UHO *User Home Organization*, ou Organização-Sede do Usuário, é usado para referenciar uma organização com a qual o usuário possui um contrato de serviço.

VoIP *Voice over IP,* ou voz sobre IP, faz referência a serviços de transporte de

voz sobre a Internet.

VPN *Virtual Private Network,* ou rede virtual privada, é um serviço oferecido por operadoras que garante segurança e desempenho na conexão entre dois pontos remotos pertencentes à mesma organização, de forma transparente.

W3C *World Wide Web Consortium,* organização de padronização para as aplicações, ferramentas, formatos e linguagens para a Web.

XML *Extensible Markup Language,* ou linguagem de marcação extensível, é uma meta-linguagem, padronizada pelo W3C, baseada em texto, que permite a definição de sintaxes de outras linguagens.

XQCM *eXtended Qos Control Module*, um sub-módulo opcional do QCM do EuQoS, responsável pelo gerenciamento de perfil de usuário, mostrar dados da sessão do usuário e informações de cobrança.

RESUMO

Vários trabalhos recentes propõem soluções de QoS permitindo que os usuários especifiquem explicitamente a qualidade de serviço requisitada durante a chamada invocação explícita de serviço com QoS. Esta flexibilidade requer uma negociação dinâmica de QoS, incluindo novos mecanismos para autenticação, autorização e contabilidade (AAA). Em especial, uma organização deveria ser capaz de controlar o uso dos serviços de rede, de modo a atender aos seus objetivos de negócios, autorizando somente serviços de rede com QoS baseados em regras de destino de serviços, de aplicações e de usuários. Esta dissertação trata do modelo do conjunto de dados e conhecimentos para suportar o processo de concessão de autorização para requisições de acesso a serviços com requisitos de QoS, e propõe um modelo de perfil de usuário para autorização de serviços com QoS, baseado em uma descrição semântica do SLA contratado com a operadora, utilizando uma ontologia para a definição de conceitos empregados. Nesta proposta, os usuários podem ser classificados de acordo com as necessidades das funções que exercem em perfis específicos. Cada perfil de usuário indica os serviços que o usuário está autorizado a requisitar, seus limites de consumo e escopo. Políticas de autorização baseadas em perfil de usuário podem ser definidas e executadas durante a invocação de serviços com QoS explícita.

.

ABSTRACT

Recent works propose QoS solutions allowing the users to explicitly specify the quality level they are requesting, during a so-called explicit QoS service invocation. This flexibility requires dynamic QoS service negociation including new mechanisms for authentication, authorization and accounting (AAA). In particular, an organization should be able to control their network services usage so as to support their main business objectives, by authorizing only network QoS services based on service destinations, applications and users' roles. This dissertation deals with the knowledge and data set model to support the authorization concession of explicit QoS services requests and proposes a user profile model for QoS service authorization, based on a semantic description of the SLA signed with the NSP, using an ontology for the definition of applied concepts. In this proposal, the users may be classified according to their job activities into specific profiles. Each user profile indicates the services that the user is authorized to request, their consumption limits and scope. User profile-based authorization polices may be defined and executed during explicit QoS service invocations.

1 INTRODUÇÃO

A crescente demanda por serviços multimídia sobre a Internet tem estimulado esforços de pesquisa e desenvolvimento de arquiteturas de Qualidade de Serviço (QoS) sobre redes IP. Muitas aplicações, entre elas as aplicações multimídia apresentam requisitos de desempenho específicos, que não são garantidos pelo serviço padrão de melhor esforço das redes IP. Daí a necessidade de mecanismos de controle (as chamadas arquiteturas de QoS) para que as redes ofereçam desempenho previsível para suportar tais serviços.

Há pelo menos três atores em uma prestação de serviços de rede com QoS: (1) o *Cliente*, uma entidade com permissão legal para assinar o serviço; (2) o *Usuário*, uma entidade autorizada pelo Cliente para invocar/usar o serviço; e (3) o *Prestador de Serviço de Rede* (NSP – *Network Service Provider*), uma entidade que oferece os serviços de rede aos Clientes/Usuários.

Durante a fase de assinatura do serviço, o Cliente e o NSP negociam um contrato de serviço, o Acordo de Nível de Serviço (SLA – *Service Level Agreement*) [Westerinen 2001]. O SLA especifica os termos e condições para provisionamento e requisição/acesso aos serviços de rede. Um SLA é composto por uma lista de Especificações de Nível de Serviço (SLS – *Service Level Specification*), que contêm os parâmetros de desempenho esperados, usualmente em termos de vazão, atraso, variação de atraso (jitter) e taxa de perda de pacotes.

Algumas soluções de QoS não necessitam de serviços de AAA (*Authentication, Authorization and Accounting*, – Autenticação, Autorização e Contabilidade), que realizam

a verificação da identidade dos usuários, a autorização para verificar se o usuário tem permissão para invocar o serviço, e a contabilidade do uso dos serviços. Este é o caso, por exemplo, quando é realizada uma simples diferenciação de tráfego na conexão de Internet do cliente, definindo diferentes prioridades para diferentes classes de tráfego. Neste caso, normalmente os campos dos protocolos são utilizados para classificar os pacotes (por exemplo, endereços IP e portas), não oferecendo a possibilidade de autorizar o uso dos serviços apenas para determinados usuários.

As invocações de serviços com QoS podem ser *explícitas* ou *implícitas* [Damilatis 2002]. Em uma invocação implícita, a fonte de tráfego simplesmente submete o tráfego para a rede, sem qualquer especificação de QoS. Esse tráfego receberá o tratamento determinado pela configuração dos equipamentos de rede a partir do SLA negociado previamente entre o cliente e o NSP. Em uma invocação de serviço explícita, a fonte escolhe o nível de QoS no momento da invocação, e há a necessidade de um módulo de autorização para verificar se o usuário tem a permissão para requisitá-la, antes de permitir que a fonte submeta o tráfego.

Soluções de QoS que suportam invocação explícita de QoS usualmente utilizam controle de QoS fim-a-fim orientados a sessão (usando protocolos de negociação e de sinalização de QoS) ([Mykoniati 2003], [Chakravorti 2003], [Masip-Bruin 2007]). Tais soluções normalmente propõem o uso de servidores de AAA (*Authentication, Authorization and Accounting*, – Autenticação, Autorização e Contabilidade) que realizam a verificação da identidade dos usuários, a autorização para verificar se o usuário tem permissão para invocar o serviço de QoS, e a contabilidade do uso dos serviços. Uma destas soluções de

QoS é a arquitetura EuQoS (*End-to-End Quality of Service over Heterogeneous Networks*) [Masip-Bruin 2007], que provê garantias de QoS fim-a-fim sobre redes heterogêneas em larga escala, considerando a natureza de múltiplos Sistemas Autônomos (AS – *Autonomous Systems*) da Internet.

Em ambientes corporativos, cada vez mais dinâmicos e competitivos, uma das metas constantemente buscadas é a redução de custos. Assim, muitos clientes corporativos têm interesse em poder controlar o uso dos serviços de rede (e conseqüentemente os custos). Observe que a redução de custo depende do modelo de tarifação utilizado. Os modelos de tarifação mais comumente utilizados são (i) valor fixo mensal, independente do consumo, com limitação de utilização de recursos, e (ii) valor variável, baseado no consumo, tanto em quantidade de tráfego mensal (por exemplo, em serviços de VPN), quanto em quantidade de minutos de utilização do serviço (por exemplo, em serviços de VoIP). A vantagem econômica oferecida por sistemas que ofereçam esse tipo de controle é óbvia, e útil mesmo se o modelo de tarifação for por preço fixo, uma vez que o uso dos recursos limitados pode ser otimizado para as atividades estratégicas e/ou mais lucrativas para a empresa.

Para a redução dos custos operacionais de uma instituição, é desejável que o sistema permita aos clientes definir perfis de usuários com diferentes direitos de uso e diferentes limites de consumo dos serviços de rede. Para atender a demanda de clientes corporativos em termos de autorização de serviços de QoS, o cliente ou o NSP precisa adotar arquiteturas de AAA que implementem políticas de autorização baseadas em perfil de usuário. Trata-se de um paradigma baseado em regras que torna simples configurar dinamicamente o comportamento do sistema, uma vez que as regras são armazenadas

separadas da implementação. Os perfis de usuário criados pelo cliente definem quais serviços de rede com QoS o usuário é autorizado a requisitar. Para simplificar este gerenciamento, podem ser definidos grupos (ou classes) de usuários com os mesmos direitos de uso dos serviços. Por exemplo, o cliente pode definir que os usuários do grupo do departamento de *TI* (Tecnologia da Informação) podem usar VoIP com qualidade *Gold* para alguns destinos específicos, limitando o consumo individual a 2.000 minutos por mês para cada usuário, e o consumo agregado para todos os usuários desse departamento em 7.000 minutos por mês, enquanto o departamento de vendas pode usar VoIP com qualidade *Premium* para qualquer destino, com 6.000 minutos de limite individual mensal e 20.000 minutos para todos os usuários do departamento.

1.1 Objetivos

1.1.1 Objetivo Geral

Esta dissertação trata da estrutura do conjunto de dados e conhecimentos para suportar o processo de concessão de autorização para requisições de serviços com requisitos de QoS baseado em um modelo de perfil de usuário destinado a arquiteturas de QoS que suportam a invocação explícita de serviços com QoS. O perfil do usuário proposto é especificado usando RDF (*Resource Description Framework*) [W3C 2004], oferecendo a característica da extensibilidade do perfil de usuário via a incorporação de diferentes conceitos adotados para compor os SLA/SLS de um NSP específico. Os conceitos usados no perfil de usuário proposto e seus relacionamentos são formalmente especificados por

uma ontologia, usando o editor de ontologias Protégé [Noy 2003]. O trabalho também exemplifica a criação de perfis dos usuários de uma empresa hipotética, e apresenta um possível processo de autorização de serviços de QoS usando o perfil proposto considerando a arquitetura EuQoS. A validação do modelo é realizada através de um protótipo que permite validar a proposta.

1.1.2 Objetivos específicos

Os objetivos específicos desta dissertação são os seguintes:

a) Apresentar os conceitos envolvidos com a autorização de invocação explícita de serviços de rede com QoS;
b) Especificar formalmente um modelo de perfil de usuário para ser utilizado em autorização de invocação de serviços de rede com QoS para clientes corporativos;
c) Definir o processo de autorização de serviços com QoS baseado no perfil do usuário.
d) Validar o modelo proposto usando um protótipo.

1.2 Justificativa

Uma tendência de evolução da Internet é a oferta de serviços com QoS suportada por arquiteturas de QoS orientadas a sessão. Isso deve levar a um novo modelo de negócios na oferta de serviços de rede, com cobrança baseada no consumo. Em um contexto corporativo, clientes e usuários são entidades diferentes, e é relevante para o cliente

controlar o uso dos serviços de rede pelos seus usuários. O presente trabalho se justifica pela inexistência de soluções de AAA que permitam a separação clara dos conceitos de usuário e cliente, necessária para o agrupamento de usuários e definições de direitos de uso e limites de consumo para cada grupo de usuários de um mesmo cliente corporativo.

1.3 Organização do Trabalho

O restante desta dissertação está organizado da forma que segue. O capítulo 2 apresenta os sistemas de AAA. O capítulo 3 destaca algumas aplicações de AAA em projetos de arquiteturas de QoS, e como isso afeta as negociações de SLA e as invocações de serviço. O capítulo 4 trata dos requisitos mínimos para a implementação de políticas de autorização baseadas em perfis, para arquiteturas de QoS que permitem invocação de serviços com QoS explícita. O modelo de perfil de usuário proposto é apresentado no capítulo 5, e um protótipo para a validação do modelo é apresentado no capítulo 6. Finalmente, o capítulo 7 apresenta as conclusões e os trabalhos futuros.

2. AAA – AUTENTICAÇÃO, AUTORIZAÇÃO E CONTABILIDADE

Os sistemas de AAA oferecem as funções de autenticação, autorização e contabilidade. A autenticação é a verificação da identidade do usuário ou do provedor de serviço, que pode ser baseada em algo que o usuário sabe (uma senha), em algo que o usuário tem (um token, ou certificado digital), ou em alguma característica física do usuário (biometria). A autorização verifica se o usuário tem o direito de requisitar o serviço. Finalmente a contabilidade é o mecanismo responsável pela coleta de dados dos sistemas de medição e por agregar os dados e armazená-los sob a forma de registros contábeis [Rensing 2002]. Algumas vezes tais servidores também incluem o faturamento, sendo então chamados AAAC (*Authentication, Authorization, Accounting and Charging*), que reúne os registros de contabilidade de uso, calculando os valores a serem faturados.

Este capítulo apresenta os principais protocolos de AAA e as principais arquiteturas genéricas de AAA encontradas na literatura, com foco nos aspectos relevantes para a autorização de serviços para clientes corporativos.

2.1. Protocolos de AAA

Os protocolos de AAA estão relacionados aos procedimentos de autenticação e autorização e ao registro do uso dos recursos pelos usuários. Os protocolos mais usados para AAA são o RADIUS (*Remote Authentication Dial In User Services*) [Rigney 2000-1] e o DIAMETER [Calhoun 2003].

O protocolo RADIUS foi projetado originalmente para prover AAA no processo de

acesso a redes discadas. Ele foi o primeiro protocolo a combinar os serviços de AAA em suas mensagens. Atualmente, além do serviço de AAA em acessos discados, este protocolo é usado em várias outras situações e foram definidas extensões que permitem vários métodos de autenticação e tarefas básicas de contabilidade [Rigney 2000-2]. O RADIUS segue uma arquitetura cliente-servidor, conforme ilustrado na Fig. 1. O usuário se conecta ao NAS (*Network Access Server*), que é o cliente RADIUS. O NAS faz as requisições ao servidor AAA (servidor RADIUS), informando as credenciais do usuário e as informações do serviço requisitado. O RADIUS não separa a autenticação da autorização, e as realiza em uma mesma transação. O servidor então valida as credenciais contra uma base de dados. Se não for encontrado um registro válido, a requisição é rejeitada ou aceita com uma configuração padrão (visitante). Caso um registro válido seja encontrado, a requisição é aceita, e é retornada também uma lista de atributos para serem aplicados a essa conexão, como tipo de serviço, IP, etc [Carrol 2004]. De acordo com a resposta do servidor, o NAS permite ou não o acesso do usuário, e aciona a transação de requisição de início de contabilidade. O final da contabilização de uso pode ser acionado pelo usuário, pelo NAS ou por uma interrupção no serviço. O servidor RADIUS pode autenticar e autorizar as requisições localmente, ou repassar para outro servidor RADIUS, atuando como um proxy [Hewlett-Packard 2003]. O RADIUS não pode ser considerado um protocolo genérico para AAA, devido às suas limitações no que tange à contabilidade, segurança fim-a-fim, e mobilidade de usuários entre domínios [Rensing 2002].

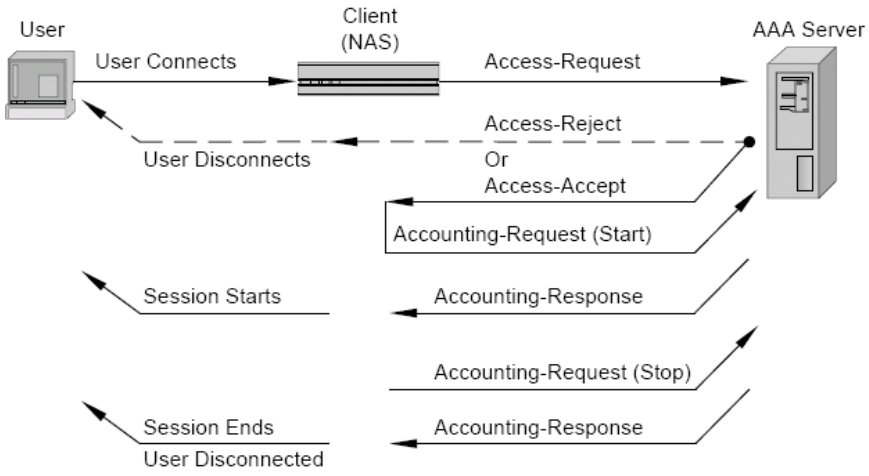


FIGURA 1: Transação RADIUS para acesso à rede [Hewlett-Packard 2003]

O protocolo DIAMETER foi projetado para ser o sucessor do RADIUS, permitindo uma fácil migração, e superando as limitações do RADIUS, através da inclusão do suporte (i) a várias redes de acesso, (ii) a modelos de segurança distribuídos, e (iii) à mobilidade de usuários entre domínios. O DIAMETER possui um protocolo base [Calhoun 2003], que suporta funções genéricas de AAA, e permite o desenvolvimento de extensões específicas para fazer a autorização em cada tipo de aplicação [Fajardo 2007].

Tanto o DIAMETER quanto o RADIUS trabalham com o conceito de AVP (*Attribute-Value Pair* – Par Atributo-Valor), para transportar os parâmetros entre cliente e servidor. No RADIUS os AVPs são identificados por um código de 8 bits, o que permite 256 AVPs diferentes. O DIAMETER expandiu esse código para até 32 bits, ampliando a possibilidade de criação de novos AVPs de acordo com a necessidade. A atribuição dos códigos a cada novo AVP criado para aplicações específicas ficou a cargo do IANA. Os AVPs já utilizados pelo RADIUS também são suportados pelo DIAMETER, para facilitar a migração. Um exemplo de AVPs usados para QoS pode ser encontrado em [Korhonen

2007-2]. A Fig. 2 mostra a estrutura de um AVP para o DIAMETER.

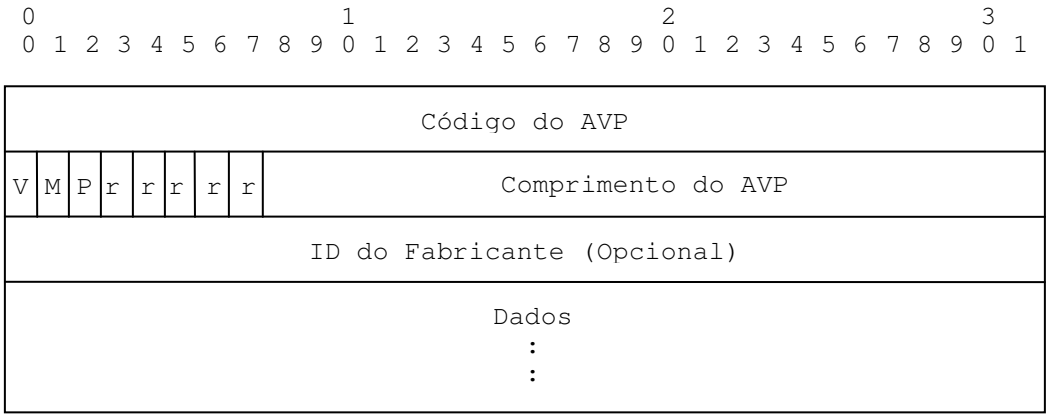


FIGURA 2: Formato de AVP do DIAMETER [Calhoun 2003]

Onde:

- Código do AVP (32 bits): É o código que identifica o atributo de maneira inequívoca. Valores até 255 são reservados para compatibilidade com o RADIUS, e não usam o campo *ID do Fabricante*. Valores a partir de 256 são atribuídos pelo IANA.
- Flags AVP (8 bits): Os flags AVP indicam ao destinatário como a estrutura deve ser manipulada. Os bits marcados com "r" são reservados. Se o bit "P" estiver marcado, o AVP conteúdo do AVP deve ser cifrado por questões de segurança fim a fim. O bit "M" indica que o AVP é obrigatório, e toda a mensagem deverá ser recusada se o AVP não for reconhecido ou estiver com um valor inválido. Se o bit "M" estiver zerado, o AVP é só informacional, e se não for reconhecido pode ser ignorado. O bit "V" indica que o AVP é específico de um fabricante, e o campo *ID do Fabricante* está presente e especifica o código do fabricante.

- Comprimento do AVP (24 bits): É o tamanho total da AVP em bytes, incluindo o cabeçalho e os dados.
- ID do Fabricante (32 bits): Usado em conjunto com o flag “V”, para indicar que o AVP é associado a um fabricante. O código do fabricante é atribuído pelo IANA.

O DIAMETER define um AVP para conter as credenciais do usuário (AVP User-Name), mas não está definido nenhum AVP para a identificação do cliente. O formato da área de dados do AVP User-Name segue o padrão de e-mails (usuário@domínio), com até 63 caracteres, onde o domínio é a identificação do NSP com o qual foi firmado o contrato, usado para validações de usuários de outros domínios. Para suportar a identificação do usuário e do cliente, basta criar um domínio para cada cliente, administrado pelo NSP, o que permite usar o AVP User-Name para conter também a identificação do cliente (usuario@cliente.nsp). Outra alternativa é criar outro AVP para conter a identificação do cliente. Para o controle de custos, há mensagens e AVPs para controle de crédito pré-pago, mas os usuários são tratados individualmente.

Em termos de QoS, o AVP de autorização QoSFilterRule [Calhoun 2005] provê regras de filtragem de QoS para serem configuradas no servidor de rede de acesso para o usuário. [Korhonen 2008] estende as funcionalidades do QoSFilterRule e define novos AVPs relacionados com QoS. [Sun 2008] descreve uma aplicação DIAMETER que realiza os serviços AAA para QoS, usando os AVPs de QoS. Nesta aplicação, um usuário final solicita um serviço com QoS enviando uma mensagem de reserva de QoS para o Cliente DIAMETER (um elemento de rede, no caso do EuQoS é o módulo A-SSN), que completa

os parâmetros necessários à autorização e em seguida encaminha uma mensagem QAR (QoS-Authorization Request) para o Servidor DIAMETER (entidade autorizadora, no caso do EuQoS, o módulo SAAA). O cabeçalho das mensagens do DIAMETER é ilustrado na Fig. 3.

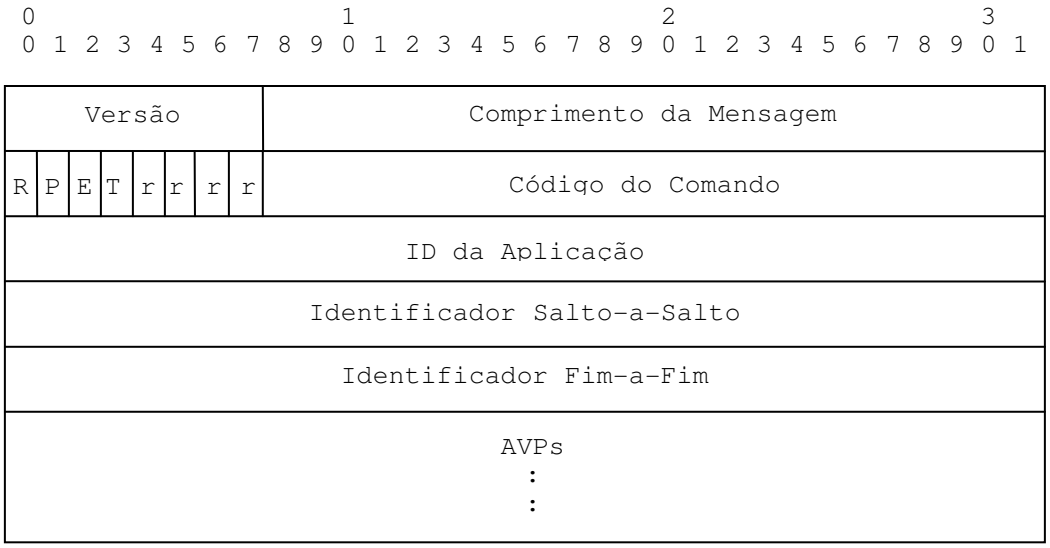


FIGURA 3: Cabeçalho das Mensagens DIAMETER [Calhoun 2003]

Onde:

- *Versão* (8 bits): Indica a versão do DIAMETER. Atualmente, versão 1.
- *Comprimento da Mensagem* (24 bits): Indica o tamanho da mensagem, incluindo os headers.
- *Flags de Comando* (8 bits):
  - "R" (*Request*) – Indica que é uma requisição. Se estiver com valor zero, é uma resposta. Em notação ABNF (*Augmented Backus-Naur Form*) é representado por "REQ".
  - "P" (*Proxiable*) – Indica que a mensagem pode ser tratada por um

proxy. Caso contrário, somente pode ser processada localmente. Em notação ABNF é representado por “PXY”.

- “E” (*Error*) – Indica que a mensagem contém uma mensagem de erro.
- “T” – Indica que a mensagem é uma retransmissão.
- r (reservado) – Reservados para uso futuro.

- *Código do Comando* (24 bits): Indica o comando associado à mensagem.
- *ID da Aplicação* (32 bits): Indica a qual aplicação a mensagem se relaciona.
- *Identificador Salto-a-Salto* (32 bits): Serve para associar a mensagem de requisição com a correspondente mensagem de resposta.
- *Identificador Fim-a-Fim* (32 bits): Não é alterado no caminho da origem ao destino, e é usado para identificar eventuais duplicações de mensagens, em conjunto com a identificação do host de origem.

A estrutura da mensagem QAR é ilustrada em notação ABNF na Fig. 4 (campos entre “[“ e “]” são opcionais).

```
<QoS-Request> ::= < Diameter Header: XXX, REQ, PXY >
                        < Session-Id >
                        { Auth-Application-Id }
                        { Origin-Host }
                        { Origin-Realm }
                        { Destination-Realm }
                        { Auth-Request-Type }
                        [ Destination-Host ]
                        [ User-Name ]
                    *   [ QoS-Resources ]
                        [ QoS-Authz-Data ]
                        [ Bound-Auth-Session-Id ]
                    *   [ AVP ]
```

FIGURA 4: Estrutura da QAR [SUN 2008]

Onde:

- <*Diameter Header*>: O Command-Code (XXX) ainda está para ser definido, é uma requisição (REQ), e pode ser tratada por proxy (PXY);
- <*Session-ID*>: O Identificador da sessão, criado pelo cliente DIAMETER, composto pela identificação do remetente, um número de 32 bits correspondente ao relógio da máquina, um número seqüencial de 32 bits, e opcionalmente um valor adicional, todos separados por ";";
- *{Auth-Application-ID}*: o ID da aplicação responsável pela autorização;
- *{Origin-Host}*: A identificação do host que originou a mensagem;
- *{Origin*-Realm}: A identificação do domínio de origem;
- *{Destination-Realm}*: A identificação do domínio de destino;
- *{Auth-Request-Type}*: O tipo de requisição de autorização (somente autorização, somente autenticação ou autenticação e autorização);
- *[Destination-Host]*: A identificação do host de destino;
- *[User-Name]*: O nome do usuário, no formato de endereço de correio eletrônico – usuário@domínio;
- *[QoS-Resources]*: Os recursos de QoS desejados. Podem ser especificados vários fluxos com QoS, através do AVP *Extended-QoS-Filter-Rule* [Korhonen 2008], que pode incluir, para cada fluxo [Korhonen 2007-1]: (i) a banda mínima, e de pico em bits por segundo, (ii) o tamanho máximo das rajadas admitidas, em bytes, (iii) a unidade mínima de controle, em bytes, (iv) as taxas máximas de erros e de perdas de pacotes admitidas, em

unidades de 10^{-11}, (vi) o atraso máximo, em microssegundos e (vii) a flutuação máxima, em microssegundos;

- *[QoS-Authz-Data]*: Credenciais do usuário, necessário em caso de requisição de autenticação;
- *[Bound-Auth-Session-Id]*: Contém a identificação da sessão de autenticação do usuário, previamente autenticado no sistema.

Tanto o DIAMETER quanto o RADIUS especificam os elementos envolvidos e como é realizada a troca de informações entre eles, mas não especificam como a autorização é efetuada, e nem como o SLA ou os perfis e permissões de usuários são tratados.

2.2. Arquiteturas Genéricas de AAA Baseadas em Políticas

O gerenciamento de redes baseadas em políticas é um paradigma baseado em regras que oferece facilidade de configuração dinâmica do comportamento do sistema, uma vez que as regras são separadas da implementação. A RFC 2903 [Laat 2000] definiu uma arquitetura genérica de AAA baseada em políticas, considerando a topologia da Internet baseada em múltiplos ASs, onde todos os AS tem pelo menos um servidor AAA. O AS onde o usuário assinou o serviço é chamado UHO (*User Home Organization*, ou Organização-sede do Usuário). O usuário pode requisitar um serviço com QoS na sua própria UHO ou em uma FO (*Foreing Organization*, ou Organização Estrangeira). Neste caso, a FO deve enviar as credenciais do usuário e o serviço requisitado para a UHO, para autenticar o usuário e autorizar o uso do serviço.

A arquitetura proposta pela RFC 2903 tem os seguintes componentes [Laat 2000]: (1) um servidor AAA genérico, que processa as regras de políticas armazenadas em um (2) Repositório de Políticas (PR – *Policy Repository*); (3) Módulo Específico de Aplicação (ASM – *Application Specific Module*), que faz o gerenciamento de recursos de configura os equipamentos de serviço (SE – *Service Equipment*), que provêem o serviço autorizado, e (4) uma base de dados com os registros de autorização, para efeitos de contabilidade e auditoria.

As RFCs 2904 [Vollbrecht 2000-1], 2905 [Farrel 2000] e 2906 [Vollbrecht 2000-2] formam um grupo de documentos que especificam, respectivamente, o esquema de autorização para arquiteturas de AAA, seus requisitos e exemplos de aplicações. O esquema proposto pela RFC 2904 define as formas de interação entre os elementos envolvidos na autorização (usuário, SE, e servidor AAA), para a operação na UHO, e as operações em uma FO (autorizando através de trocas de mensagens com o servidor AAA da UHO). Além disso, a RFC 2904 também trabalha com o gerenciamento de recursos, troca de mensagens AAA, e questões de segurança, desempenho e escalabilidade.

Em [Rensing 2002], os autores propõe outra arquitetura genérica de AAA baseada em políticas, chamada de A^x. Ela expande a arquitetura proposta na RFC 2903, definindo uma clara separação entre os serviços e um particionamento entre os níveis de serviço. O particionamento define horizontalmente 4 camadas de serviço (1-conectividade Internet, 2-transporte, 3-aplicações e 4-conteúdo) e verticalmente dois tipos (controle e dados), conforme ilustrado na Tabela 1. A divisão horizontal ajuda a definir os requisitos de cada serviço. Por exemplo, na camada 1 (conectividade Internet) a autenticação baseada em

hardware pode ser feita, enquanto na camada 4 (conteúdo) uma autorização baseada em credenciais do usuário geralmente é necessária. A divisão vertical ajuda a identificar em que ponto os serviços são necessários. A autenticação e sinalização são efetuadas durante a fase de controle, enquanto a contabilidade é realizada durante a fase de troca de dados.

Tabela 1: A estrutura de particionamento da arquitetura A^x [Rensing 2002]

Nível	Controle	Dados
Conteúdo	RTSP	News, Fluxo de Vídeo
Aplicação	http, H.245, SIP	Videoconferência, Telefonia IP, Applets Java
Transporte	RSVP, RTCP, ICMP	TCP, UDP, RTP
Conectividade	DHCP, ICMP	SONET/SDH, DWDM

Os serviços são separados em serviços prestados ao usuário (*user services*), cuja tarefa fica a cargo dos equipamentos de serviço usuário (*user SE*) e serviços de AAA (*A^x services*), sendo esses últimos prestados aos *user SE* ou a outros servidores A^x, no caso de um acesso de um usuário fora do seu domínio. Os principais componentes da arquitetura A^x são descritos abaixo e ilustrados na Fig. 5 [Rensing 2002]:

- **A^x PDP**: (*Policy Decision Point*), uma parte importante do servidor A^x, responsável pelas avaliações das políticas, resultando em decisões de aceitação ou negação das solicitações de serviços com QoS;
- **A^x PR**: (*Policy Repository*), onde são armazenadas as políticas de aceitação de novos fluxos com QoS;

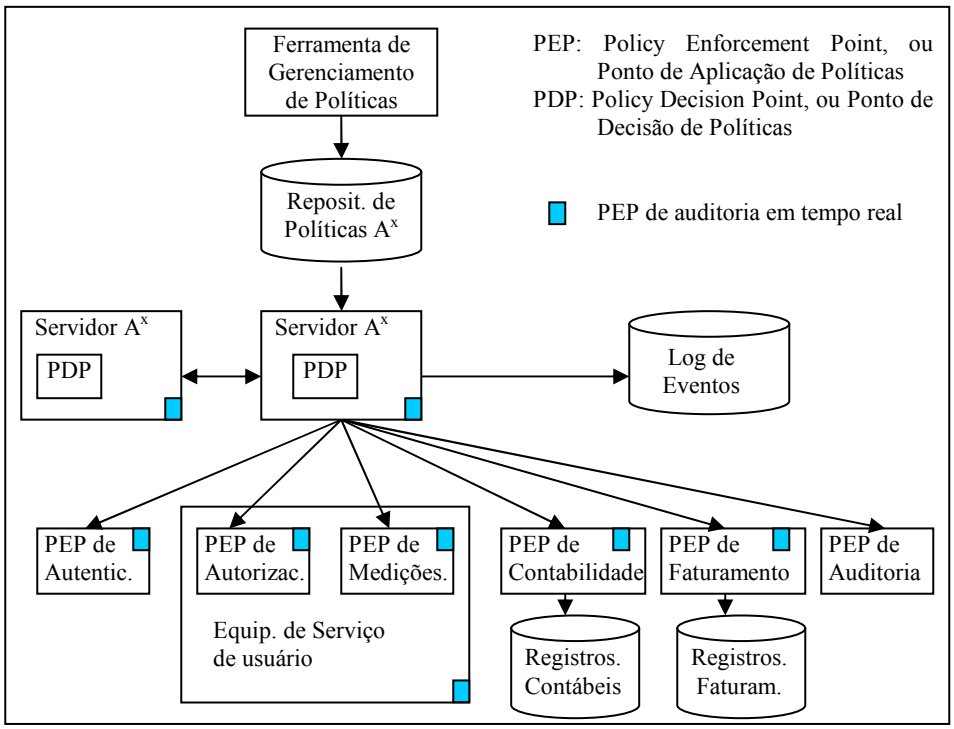


FIGURA 5: A arquitetura genérica A^x [Rensing 2002]

- **PEP de Autenticação**: (*Policy Enforcement Point*), módulo responsável pela aplicação das políticas relativas à autenticação;
- **PEP de Autorização**: módulo interno ao equipamento de serviço do usuário (*user SE*), responsável pela aplicação das políticas de autorização;
- **PEP de Medições**: módulo interno ao *user SE*, responsável pela coleta de dados dos serviços prestados ao usuário;
- **PEP de Contabilidade e PEP de Faturamento**: módulos com bases de dados, responsáveis pela contabilidade e pela geração das faturas de consumo;

- **PEPs de Auditoria**: módulos que implementam as políticas de auditoria, e podem estar embutidos em cada um dos outros módulos ou como um módulo independente.

Ambas as arquiteturas genéricas de AAA propostas por [Laat 2000] e [Rensing 2002], assim como outros trabalhos na área ([Altmann 2002], [Briscoe 2005], [Chaochi 2004], [Hasan 2002], [Hwang 2002], [Kurtansky 2004], [Oromont 2006], [Papaioannou 2004] e [Roberts 2004]) não diferenciam claramente os conceitos de usuário e cliente. Dessa forma, eles não consideram os clientes corporativos que gostariam de agrupar os limites de consumo entre diferentes perfis de usuário e também querem receber uma única conta com o consumo de todos os seus usuários (incluindo ou não os consumos individuais detalhados). Para cobrir essa lacuna, este trabalho propõe um modelo de autorização de usuário, descrito no capítulo 5.

2.3. Considerações Finais

Este capítulo apresentou os principais conceitos envolvidos em sistemas de AAA, bem como suas limitações para aplicação em um contexto de um provedor de serviços que tem clientes corporativos, incluindo os principais protocolos utilizados para AAA, o RADIUS e o DIAMETER, e as principais arquiteturas genéricas de AAA, a AAAArch, proposta por [Laat 2000] e a A^x, proposta por [Rensing 2002].

O próximo capítulo explora algumas das principais arquiteturas de QoS encontradas e sua relação com os sistemas de AAA.

3. ARQUITETURAS DE QOS E AAA

As primeiras arquiteturas de QoS propostas pelo IETF foram a IntServ [Braden 1994], que controla as reservas de recursos individualmente, por fluxo, e por isso tem problemas de escalabilidade, e a DiffServ [Blake 1998], que apenas diferencia os tipos de aplicações em classes de serviço (CoS – *Class of Service*) com um tratamento diferenciado para cada classe (e não individualmente por fluxo), nos roteadores, o que é escalável mas não oferece garantia plena de atendimento a cada fluxo. Estas duas arquiteturas originalmente não apresentavam um serviço de autorização de tráfego baseada em permissões de usuário.

Trabalhos recentes propõem soluções de QoS envolvendo serviços de AAA, de modo a oferecer uma garantia de desempenho aceitável aos fluxos autorizados, e de forma escalável. Este capítulo apresenta algumas destas soluções.

3.1 DiffServ com *Bandwidth Brokers* (BB) [Nichols 1999].

A arquitetura DiffServ foi incrementada com a adição dos *Bandwidth Brokers* (BB), na RFC 2638 [Nichols 1999]. Um BB é um elemento que é configurado com as políticas da organização e administra de modo centralizado a disponibilidade de largura de banda em um domínio de rede com suporte a DiffServ, realizando as tarefas de Autenticação, Autorização e Controle de Admissão de Conexões (CAC – *Connection Admission Control*). Um BB pode ser um host, um roteador ou um processo em um roteador. Uma rede pode ser conter BB redundantes, por questão de escalabilidade ou tolerância a falhas.

Quando uma máquina precisa submeter um tráfego com QoS para a rede, ela solicita ao BB (enviando uma solicitação ao roteador de borda) qual a classe de serviço (CoS) que deve marcar em seus pacotes. A largura de banda disponível é controlada somente nos roteadores de borda da rede do AS. Os roteadores internos implementam um domínio DiffServ, com as classes suportadas pelo AS. Quando um roteador de borda recebe uma requisição de QoS, ela é encaminhada para o BB, que verifica: (i) as credenciais do usuário (autenticação), (ii) se o fluxo total para esse usuário, incluindo a nova requisição não é maior do que a largura de banda contratada (autorização) e (iii) se a rede tem recursos suficientes para suportar a nova requisição (CAC). Se todas as três condições forem satisfeitas, uma mensagem é enviada para os roteadores envolvidos para reservar a largura de banda para o fluxo, e a quantidade de largura de banda requisitada é decrementada da quantidade disponível, e registrada como usada pelo usuário.

A arquitetura BB suporta dois tipos de serviço, além do melhor esforço: serviço *premium*, e serviço garantido (*assured*). O serviço premium recebe um tratamento de encaminhamento expresso, prioritário em relação a todos os outros, independente do tamanho das filas dos demais serviços. A reserva de banda para esse tipo de serviço é efetuada pela taxa de pico, e não são admitidas rajadas, dispensando ou minimizando a ocorrência de filas nos roteadores. Caso a taxa de tráfego submetida seja maior do que a taxa prevista, os pacotes excedentes são descartados. Dessa forma, a variação de atraso para esse tipo de pacote é minimizada, tornando-o adequado a serviços como VoIP, e VPN. Em contrapartida, por ter recursos alocados pela taxa de pico, muitas vezes está subutilizado, e usualmente é mais caro. Já os serviços garantidos possuem diferentes níveis de prioridade, e

apresentam a mesma característica de variação do atraso do serviço de melhor esforço, com a diferença de que seus pacotes têm mais prioridade. A intensidade dessa variação e a garantia de desempenho dependem do ajuste do provisionamento de banda para cada classe, dos tamanhos das rajadas aceitas e dos tamanhos das filas alocadas. Caso a taxa de submissão de uma rajada de tráfego garantido seja maior do que a taxa prevista, o excedente é remarcado como melhor esforço. Em casos eventuais de redes congestionadas, alguns pacotes podem ser descartados, o que pode ser tolerado por muitos usuários.

Nesta solução de QoS, cada usuário pode submeter tráfego até a quantidade contratada, e isso é controlado pelo módulo AAA do BB. Evitar essa sobrecarga geral da rede é função do controle de admissão de conexões (CAC).

Alguns outros trabalhos seguem essa linha, como [Terzis 1999], [Zhang 2000], [Zhang 2001] e [Kim 2004].

3.2 Projeto CADENUS

O projeto europeu CADENUS (*Creation And Deployment of ENd-User Services in premium IP networks*) [D'Arienzo 2001] [Chakravorty 2003] [Cortese 2003] também propõe uma arquitetura de QoS incluindo o AAA e o controle de admissão. O foco desse projeto é a especificação e negociação de serviços, entre o usuário, um provedor de serviços e um provedor de rede.

Para auxiliar essa negociação, são utilizados três mediadores: Mediador de Acesso (AM – *Access Mediator*), Mediador de Serviço (SM – *Service Mediator*) e Mediador de Recursos (RM – *Resource Mediator*). O AM serve para regular o acesso do usuário, tem um

servidor AAA e oferece uma lista de serviços para o usuário, com as características de cada serviço, incluindo os custos, através de uma interface pré-definida. A relação de serviços é buscada pelo AM em um diretório de serviços. Sobre essa base de dados o AM pode selecionar, por exemplo, a opção mais econômica, se o mesmo serviço for oferecido por mais de um prestador de serviço. A principal função do AM é tornar mais simples para o usuário selecionar um serviço. Quando o usuário seleciona um serviço, o AM acessa um ou mais Mediadores de Serviço (SM – *Service Mediator*) para buscar as opções de serviço disponíveis que atendem à seleção do usuário e negocia a assinatura do SLA com o usuário.

Uma vez que o serviço é selecionado, o SLA é considerado assinado e o AM encaminha a requisição para o SM correspondente, para ser armazenada. No momento da invocação, o usuário contata o AM, informando suas credenciais e o serviço contratado. Após a autenticação e autorização do serviço, o AM submete a requisição ao SM que localiza os recursos e serviços necessários para suportar o SLA (que especifica o serviço em termos de qualidade percebida pelo usuário), compõe o SLS (que especifica o serviço em termos técnicos de parâmetros de desempenho de rede) e o encaminha ao Mediador de Recursos (RM – *Resource Mediator*).

O RM verifica a disponibilidade de recursos da rede e aplica as regras de reserva. Em caso de sucesso, uma resposta positiva é retornada ao usuário, e o tráfego com QoS pode iniciar.

A arquitetura do CADENUS separa claramente os serviços das funções de gerenciamento e controle de recursos, bem como do processo de criação de novos serviços. Separa também os provedores de serviço e os provedores de rede, permitindo que os

primeiros se concentrem na elaboração dos serviços, deixando o controle de QoS a cargo dos provedores de rede.

O CADENUS também não separa os conceitos de usuário e cliente.

3.3 Projeto EuQoS

O projeto EuQoS visa a definição e implementação da arquitetura EuQoS (*End to end Quality of Service over heterogeneous networks*) [Masip-Bruin 2007] para um suporte de QoS fim-a-fim sobre redes heterogêneas, considerando múltiplos sistemas autônomos (AS) e em larga escala. Também é suportado pela União Européia, reunindo os resultados das pesquisas realizadas por outros projetos europeus na área de QoS (Áquila, Tequila e Cadenus).

O EuQoS é orientado a sessões, oferecendo QoS somente às aplicações que necessitam, e quando elas necessitam. Durante o processo de negociação destas sessões com QoS, as aplicações origem e destino definem parâmetros de QoS necessários e, caso haja recurso disponível, a rede realiza a reserva destes recursos necessários para garantir os parâmetros de QoS estabelecidos. A arquitetura EuQoS é dividida em três planos: plano de serviço, ou camada de aplicação; plano de controle ou camada de rede virtual; e plano de transferência. A camada de rede virtual é ainda subdividida em uma camada independente de tecnologia (TI – *Technology Independent*) e uma camada dependente de tecnologia (TD – *Technology Dependent*).

3.3.1 Camada de Aplicação

O sistema EuQoS suporta tanto aplicações prontas para o EuQoS (*EuQoS aware*), que já foram projetadas para suportar a abertura de sessões EuQoS, quanto outras aplicações. Estas últimas podem fazer uso das funcionalidades do sistema EuQoS via um módulo de adaptação.

Antes de iniciar uma sessão com QoS, a aplicação precisa se autenticar junto ao sistema EuQoS e requisitar a QoS. Isso é feito com o auxílio dos módulos de controle de QoS (QCM - *QoS Control Module*), e de sinalização de Aplicações (ASIG - *Application Signaling*), implementados no terminal do usuário. Já no servidor EuQoS de cada AS, há um módulo responsável pela interface com os terminais de usuário para o estabelecimento das sessões (A-SSN - *Application-level Signaling and Service Negotiation*). Esse módulo também interage com o módulo SAAA (*Security, Authentication, Authorization and Accounting*) responsável pela autenticação dos usuários, autorização das requisições e registro de consumo. A requisição de QoS só é passada para a camada de Rede Virtual, para a verificação da disponibilidade de recursos após a autenticação do usuário e autorização da requisição.

3.3.2 Camada de Rede Virtual

A camada de rede virtual é dividida em uma camada independente de tecnologia (TI), controlada por um gerenciador de recursos (RM - *Resource Manager*) para cada AS, e uma camada dependente de tecnologia (TD), controlada por um alocador de recursos (RA - *Resource Allocator*), também para cada AS.

O RM de cada AS controla a disponibilidade de recursos do próprio AS com o controle de admissão de novas conexões (CAC), o gerenciamento dos contratos de QoS e encaminhamento de solicitações de QoS para os ASs vizinhos e o controle do roteamento com QoS inter-domínio. Sempre que necessário, as decisões do RM são aplicadas na forma de configurações de dispositivos específicos por meio do RA, que está localizado na camada TD.

O RA é voltado para uma tecnologia específica, e faz as reservas e liberações de recursos (tais como banda e buffer) nos roteadores com base nas solicitações do RM. Também possui um controle de admissões, com base nos recursos efetivamente disponíveis nos equipamentos dessa tecnologia específica. Pode haver um ou mais RAs por AS, dependendo das tecnologias utilizadas na rede do AS.

A principal vantagem da existência dessas duas subcamadas é a dissociação entre as decisões da rede e as tecnologias de rede. Para maiores detalhes sobre o EuQoS, vide o anexo.

3.3.3 Limitação do modelo de usuário

Da mesma forma que os projetos anteriores, o EuQoS também não diferencia os conceitos de usuário e cliente. O EuQoS também não permite a mobilidade do usuário entre ASs.

3.4 Considerações Finais

Este capítulo apresentou uma visão geral de algumas das arquiteturas de QoS fim-a-

fim que envolvem mecanismos de AAA, incluindo o Bandwidth Broker, e os projetos europeus CADENUS e EuQoS. Também é possível observar que nenhuma das arquiteturas apresentadas suporta a distinção entre os conceitos de usuário e cliente. O próximo capítulo apresenta os requisitos para a implantação de um modelo que suporte essa distinção entre usuário e cliente para suportar uma maior flexibilidade de controle de direitos de uso dos serviços com QoS em ambiente corporativo.

4. IMPLEMENTANDO POLÍTICAS DE AUTORIZAÇÃO BASEADAS EM PERFIL PARA SERVIÇOS COM QOS

Considerando que clientes corporativos podem desejar controlar o consumo de serviços com QoS em função dos objetivos de negócio, novos modelos de autorização são necessários. As abordagens correntes para autorização de serviços nas arquiteturas de QoS mencionadas não permitem uma definição de limitações de serviços suficientemente flexível para atender aos requisitos dos clientes corporativos.

Neste capítulo são expostos os requisitos mínimos para a implementação de políticas de autorização baseadas em perfis em arquiteturas de QoS que permitem invocação explícita de serviços com QoS. As seguintes questões são analisadas: (i) o que deve ser negociado entre o cliente e o NSP durante a fase de negociação de serviço; (ii) que informações devem ser incluídas na requisição de autorização, no momento da invocação de serviço; e (iii) a quais informações o serviço de autorização deve ter acesso para autorizar ou negar a requisição.

A fim de ilustrar os conceitos apresentados, este trabalho considera uma empresa hipotética, chamada ABC, que além da matriz possui uma fábrica remota. Esta empresa pode definir que os usuários do departamento de *TI* (Tecnologia da Informação) podem usar VoIP com qualidade *Gold* até um determinado limite de tráfego total mensal, apenas em ligações para a fábrica remota, enquanto o departamento de Vendas pode usar VoIP com qualidade *Premium* para qualquer destino, com outro limite de tráfego total mensal. Por outro lado, para a limitação global, o sistema precisa permitir a especificação de um limite

de consumo para cada classe de serviço contratada. Essa limitação será compartilhada por todos os usuários autorizados pelo cliente.

4.1. Requisitos de Negociação de Serviço

Para alcançar a flexibilidade desejada ao cliente para limitar o consumo dos seus usuários, encontramos pelo menos cinco requisitos:

1) Uma clara distinção entre o usuário (a pessoa que usa os serviços) e o cliente (a entidade legal que contrata e paga pelos serviços);
2) A possibilidade de associar cada usuário a um perfil que mantém o conjunto de permissões em termos de utilização dos serviços;
3) A possibilidade de agrupar os usuários em grupos com permissões diferentes;
4) A possibilidade de definir limites de consumo globais, individuais e grupais;
5) Suportar diferentes conceitos envolvidos no processo de autorização.

A distinção entre usuário e cliente (1) é necessária para permitir a associação de vários usuários a um mesmo cliente corporativo. A associação dos usuários a perfis de usuário (2) e o seu agrupamento (3) são necessários para facilitar o gerenciamento de permissões de uso dos serviços. A definição de limites de consumo (4) serve como ferramenta de limitação de custos e de otimização de uso dos recursos contratados. Já a necessidade de suportar conceitos (5) se deve ao fato da inexistência de padrões que normalizem os conceitos envolvidos no processo de autorização, como por exemplo níveis/classes de serviço, formatos de SLA/SLS, formas de definição de técnicas de medição de consumo, etc.

Um ponto importante a ser discutido é onde os serviços de autenticação e autorização são implantados. Considerando o caso do cliente corporativo que autoriza vários usuários ou grupos de usuários com diferentes direitos, estes serviços podem ser implantados no domínio do cliente ou no domínio do NSP:

- **Implantação no domínio do cliente**: oferece uma maior facilidade ao cliente para configurar suas políticas internas e os perfis de usuário. Entretanto, o cliente precisa adquirir os equipamentos para realizar estes serviços e precisa investir em treinamento. Neste caso, a relação de usuários e seus perfis não precisam ser informados ao NSP. Assim, é responsabilidade do cliente autorizar o serviço, e o tráfego agregado é controlado pelo NSP com base no SLA firmado com o cliente (independente do conceito de usuário).
- **Implantação no domínio do NSP**: a lista de usuários e os respectivos perfis e direitos de uso devem ser informados ao NSP (por exemplo, através de uma aplicação Web). Escolher a melhor opção para transferir essa informação do cliente para o NSP está fora do escopo dessa dissertação. Essas atualizações podem ser freqüentes, e podem implicar ou não em renegociações do SLA. É necessária a renegociação caso a mudança no perfil de um grupo de usuários aumentar ou diminuir os níveis de serviços contratados. Assim, estas operações devem ser realizadas por um usuário com direitos administrativos. Neste caso, o cliente não precisa adquirir equipamentos adicionais. Esses equipamentos são mantidos pelo provedor, e podem ser usados para autenticar e autorizar usuários de vários clientes.

Note que em ambos os casos, o CAC deve ser feito no domínio do NSP para evitar a sobrecarga da rede. Esta dissertação considera a opção da implantação dos controles de AAA no domínio do provedor, por se tratar de uma implementação mais genérica. Caso se queira implantá-la no domínio do cliente, a única mudança é a replicação do mesmo modelo para o domínio do cliente. Em se tratando de um cliente com várias unidades geograficamente distribuídas, interconectadas através da rede do NSP, se for implementado do lado do cliente deverá ser replicado em cada uma das unidades, ao passo que se for implementado no lado do NSP, o controle pode ser único, agregando o controle para todos os usuários do mesmo cliente.

4.2. Parâmetros de Autorização de Serviço

Durante as invocações explícitas de serviços com QoS, o serviço de autorização deve receber pelo menos: (i) a identificação do usuário, (ii) a identificação do serviço, (iii) os endereços de origem e destino e (iv) o nível de QoS requisitado. As identificações de usuário e de serviço (i) e (ii) são usadas para verificar se o usuário tem o direito de invocar o serviço requisitado. Os endereços de origem e destino (iii) são usados para verificar se o escopo do serviço requisitado se encaixa em algum escopo contratado para esse serviço. Finalmente, o nível de QoS requisitado (iv) é usado para verificar se ele não é mais alto do que o nível de QoS que o usuário está autorizado a solicitar. Por exemplo, se um usuário do cliente ABC solicitar uma chamada VoIP para a fábrica remota da mesma empresa, o servidor AAA deve saber qual o usuário está invocando o serviço (para identificar seu grupo e como conseqüência o seu perfil), saber qual a qualidade desejada (por exemplo,

QoS *Silver*), e o destino da chamada (o endereço IP da rede local da fábrica remota).

4.3. Informações Necessárias pelo Processo de Autorização de Serviço

O serviço de autorização deve também ter acesso: (i) ao repositório de políticas de autorização do NSP; (ii) aos perfis de usuários, que mostram quais serviços os usuários podem solicitar, seus escopos e limites de consumo; e (iii) à contabilidade do consumo dos serviços, em bases individuais, grupais e globais.

O acesso ao repositório de políticas (i) é necessário para considerar as regras de políticas que o próprio provedor pode estabelecer, que podem indicar, por exemplo, a prioridade de alguns serviços quando a carga da rede estiver acima de um determinado limiar. O acesso aos perfis de usuários (ii) autorizados pelos clientes é necessário para verificar, no momento da autorização de uma requisição de serviço, se a requisição está de acordo com o perfil do usuário, não violando nenhuma de suas restrições. O acesso à contabilidade de consumo dos serviços (iii) é necessário para implementar as validações quanto aos limites de consumo do serviço solicitado impostos para o usuário que está requisitando o serviço, para o grupo ao qual ele faz parte, e para o consumo agregado de todos os usuários do mesmo cliente.

4.4. Formalização dos Conceitos Envolvidos

A formalização dos conceitos envolvidos no processo de autorização deve suportar facilmente a alteração dos termos nos quais os SLSs são especificados. É também desejável que essa formalização (i) permita essa alteração sem a necessidade de alteração da

codificação do sistema de autorização, e (ii) permita uma futura expansão do modelo para suportar a mobilidade de usuários entre ASs diferentes.

A referência atualmente mais utilizada para a especificação formal dos conceitos de um domínio é ontologia. Em ciência da computação, uma ontologia, segundo Gruber (1993), é uma "especificação explícita de uma conceitualização". Uma ontologia permite definir formalmente os termos usados em um domínio do conhecimento, incluindo as propriedades desses termos e as relações entre eles.

No contexto desse trabalho, uma característica interessante é a possibilidade de criação de novos conceitos a partir da especificação de novas propriedades sobre um conceito já existente, criando uma subclasse que herda as propriedades da classe original. Essa é uma associação do tipo "isa" ou "é um". Isso dá mais flexibilidade para o NSP definir novos parâmetros de desempenho, criando uma subclasse a partir de uma classe padrão de SLS. O módulo de autorização pode identificar esses novos parâmetros e validá-los no momento da requisição.

Para atender a um usuário móvel com contrato com um NSP (de origem), mas visitando outro NSP (corrente), o sistema do NSP corrente precisará fazer a correspondência entre a especificação do serviço que está sendo requisitado e a especificação o serviço que ele contratou no NSP de origem, uma vez que os dois NSPs podem oferecer serviços especificados em termos diferentes. O uso de ontologias permite que se faça o casamento entre esses dois serviços.

4.5. Considerações Finais

Este capítulo apresentou os requisitos mínimos necessários para a implantação de autorização de serviços de rede com QoS baseada perfis de usuário. Foram discutidos os requisitos mínimos para a negociação de serviço, os parâmetros necessários para a autorização de uma requisição, as informações às quais o processo de autorização deve ter acesso e o modelo escolhido para formalizar os conceitos utilizados.

O próximo capítulo apresenta um modelo de perfil de usuário para o uso em sistemas de autorização de requisições de serviços com QoS, voltado para arquiteturas de QoS que suportam a invocação explícita de serviços com QoS. Este trabalho propõe o uso de ontologia, permitindo a extensão do modelo de acordo com a necessidade e/ou políticas do provedor, e prevendo também a futura expansão para o suporte a usuários móveis, fornecendo uma forma de descobrir qual o serviço local corresponde ao que o usuário de outro domínio está autorizado a usar, pois os provedores diferentes podem oferecer serviços especificados em termos diferentes.

5. UM MODELO DE PERFIL DE USUÁRIO PARA A AUTORIZAÇÃO DE SERVIÇOS COM QOS

Conforme discutido na sessão 2.2, tanto a arquitetura genérica de AAA proposta por [Laat 2000] quanto a arquitetura A^x [Rensing 2002] são inapropriadas para suportar o agrupamento de usuários e as limitações de consumo. Neste capítulo é proposto um modelo de perfil de usuário para ser usado em políticas de autorização baseadas em perfis para invocação explícita de serviços com QoS, com suporte ao agrupamento de usuários e limitações de consumo, expandindo as arquiteturas genéricas de AAA mencionadas acima.

O modelo proposto está mantido no Repositório de Políticas (PR), em ambas as arquiteturas propostas por [Laat 2000] e [Rensing 2002]. Em ambos os casos, o PR é alterado para conter as associações entre o cliente e seus usuários, os serviços contratados, e as limitações de consumo definidas pelo cliente e os correspondentes registros de consumo. Baseado nessas informações, o servidor AAA pode verificar se o usuário pode requisitar o serviço. Em caso afirmativo, o ASM pode proceder à verificação da disponibilidade de recursos (CAC), e à reserva dos recursos necessários.

Para fins de autorização, existe um relacionamento entre o perfil do usuário, que indica quais serviços o usuário pode requisitar, e o SLA/SLS, que especifica esses serviços. Assim, para definir um perfil de usuário a ser aplicado durante a autorização de serviço deve-se também definir os conceitos do SLA/SLS usados durante a especificação do perfil do usuário.

A definição de SLA/SLS oferecidos varia de um provedor para outro, cada um com

a liberdade de oferecer os serviços em termos de parâmetros de desempenho diferentes. No caso de clientes móveis, que assinaram um serviço em um provedor, mas momentaneamente o acessam através de outro provedor, é preciso fazer o mapeamento dos serviços assinados no provedor de origem com os serviços oferecidos pelo provedor onde está sendo feito o acesso. Para permitir essa adequação futura do modelo proposto para um ambiente que permita a mobilidade entre domínios, a solução adotada é o uso de ontologias. Tanto o modelo de perfil de usuário proposto quanto os conceitos gerais de SLA/SLS são formalmente especificados por uma ontologia, usando o editor de ontologias Protégé [Noy 2003].

5.1. SLA e a Autorização

Um SLA pode ser genericamente definido como um acordo contendo um conjunto de obrigações entre duas partes, o provedor de serviços e o cliente. Alguns trabalhos consideram o SLA como um sinônimo de contrato de serviço, enquanto outros consideram que um contrato de serviço contém um ou mais SLAs. Este trabalho adota a segunda abordagem por ser mais genérica e incluir a primeira (um contrato de serviço com somente um SLA).

A Fig. 12 apresenta uma ontologia dos conceitos básicos ao contrato de serviço (incluindo o SLA) que são usados para a definição do nosso modelo de perfil de usuário. Foram usados os termos definidos na RFC 3198 [Westerinen 2001], na RFC 3260 [Grossman 2002], no projeto Tequila [Goderis 2003] e no Tele Management Fórum [Borioni 2001] como referência para compor a ontologia de SLA. Segue uma descrição dos

conceitos especificados na Fig. 6:

- ***Service Contract***: É o contrato de serviço, com um ou mais SLAs especificando os níveis de serviços contratados (*hasSLA* *), e é associado ao cliente (*hasCustomerParty*).
- **SLA**: É o acordo de nível de serviço (*Service Level Agreement*), incluindo um ou mais objetivos de serviço (*hasObjective* *), especificado por um SLS.
- ***Customer***: É o cliente do serviço, a entidade que autoriza (*authorizes* *) um ou mais usuários (*User*).
- ***User***: É alguém autorizado a usar os serviços especificados no contrato.
- **SLS**: O SLS especifica as condições sob as quais um serviço deve ser prestado. Um SLS especifica as conseqüências para não atender aos parâmetros especificados (*hasConsequence* *), os períodos em que o serviço estará disponível ao cliente (*hasSchedule*), o escopo (*hasScope*) indicando para quais endereços de origem e destino o serviço estará disponível, e opcionalmente a limitação de consumo para todos os usuários desse cliente (*hasGlobalConsumptionLimit*). O SLS também deve conter um identificador do serviço (*hasServiceID*), que contém o tipo do serviço (*hasServiceType*), um atributo enumerado que indica um dos tipos de serviço suportados pela NSP, e o identificador de fluxo (*hasFlowIdentifier*), com os parâmetros de identificação dos pacotes IP desse fluxo.

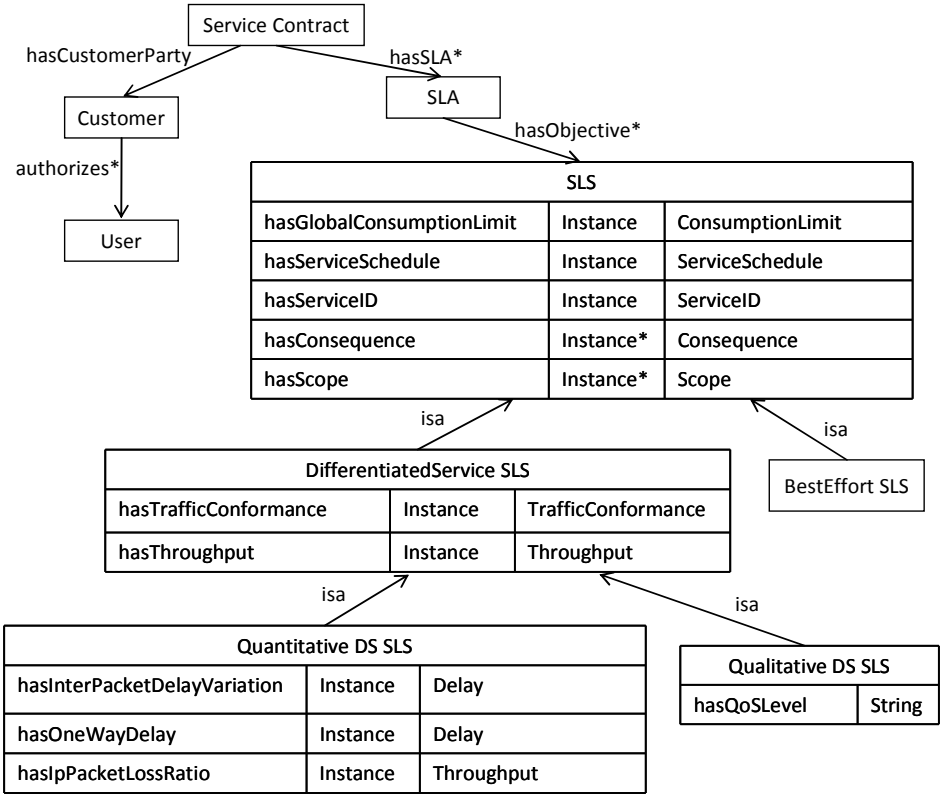


FIGURA 6. Ontologia de Contratos de Serviços

Note que a ontologia permite definir conceitos específicos do provedor, pela associação "*isa*" ("é um"), criando outra subclasse que herda os atributos da classe acima e permite criar novos atributos. Assim cada provedor pode especificar quais parâmetros serão usados para definir concretamente os conceitos do SLS, (conseqüência por não atender ao SLS, escopo, agendamento da disponibilidade, limitação de consumo global e qualquer outro que considere necessário). Por exemplo, a Fig. 6 apresenta a complementação do conceito de SLS (classe abstrata) por uma NSP que adota dois tipos de SLS:

- ***Best Effort* SLS**: Especifica um serviço de melhor esforço, sem parâmetros de garantias de desempenho adicionais. É uma classe concreta;

- ***Differentiated Service* SLS**: Especifica um serviço diferenciado, com a vazão permitida e a conformação de tráfego que define o tamanho do buffer, o tratamento dado ao excesso de tráfego e a taxa máxima permitida para rajadas. A classe *DifferentiatedServiceSLS* também é abstrata e os parâmetros adicionais de desempenho podem ser especificados de forma quantitativa ou qualitativa, em um desses dois tipos de DS SLS:
    - ***Quantitative* DS SLS**: Define o atraso em cada sentido (*hasOneWayDelay*), a variação de atraso entre os pacotes (*hasInterPacketDelayVariation*) e a taxa máxima admitida de perda de pacotes (*hasIpPacketLossRatio*).
    - ***Qualitative* DS SLS**: Usa um qualificador para escolher um dos tratamentos predefinidos a ser aplicado ao fluxo. Por exemplo, um serviço pode ser contratado com qualidade "*Gold*", "*Silver*" ou "*Bronze*", cada um com tratamentos diferentes.

O modelo ainda permite a definição de novas proprieades sem a necessidade de alteração do módulo de autorização, por exemplo através da criação de outra subclasse da classe *Quantitative DS SLS*, com novos parâmetros de desempenho quantitativos. Para isso o modelo dispõe de uma classe de parâmetros de SLS, com uma subclasse para faixas de valores admitidos (*AdmittedRangeValue*). Essa classe contém dois parâmetros opcionais do tipo real, chamados *minValue* e *maxValue*. O trabalho não considera a unidade de medida (segundo, milissegundo, microssegundo, etc), sendo o valor expresso em um valor real. Os valores inteiros são convertidos para reais.

Caso o cliente precise incluir um novo de desempenho "abcd" no SLS, com garantia

de limite superior de desempenho (por exemplo, atraso) de até 100ms, por exemplo, a rede do NSP será programada para atender a esse requisito em até 100ms. Isso significa que o usuário pode solicitar esse mesmo parâmetro de desempenho “abcd” com um valor de 100 ou superior, pois se solicitar um valor menor que esse, a rede não estará apta a atender. Nesse caso, o cliente deve definir uma nova propriedade com nome “abcd”, e valor mínimo (*minValue*) 100.0. No momento da requisição, se o usuário solicitar um valor de desempenho “abcd” menor que 100, o pedido não será autorizado.

Por outro lado, se for necessário incluir um outro parâmetro “xyz” de desempenho, que a rede deve fornecer no mínimo um determinado valor (por exemplo, vazão) de 30000 bits por segundo, esse valor deve ser associado ao *maxValue* da propriedade “xyz” do SLS, pois será o valor máximo que o usuário pode solicitar no momento de requisitar uma sessão com QoS através desse SLS.

5.2. Perfil de Usuário

Os usuários autorizados por um cliente podem ser agrupados para a definição de limites de consumo grupais. Cada grupo de usuários é associado a um perfil, que define quais os direitos que esses usuários têm para requisitar serviços contratados pelo cliente. Cada um dos direitos de uso especifica em que condições um serviço contratado pode ser requisitado. A Fig. 7 apresenta a ontologia de Perfil de Usuário orientada a autorização de serviços de rede com QoS. Os seguintes conceitos são especificados:

- ***User*:** um usuário do serviço de rede que é associado a um grupo de usuários (*hasUserGroup*) e a um registro de consumo (*hasAccounting*). Esse processo de contabilidade mantém um registro para cada direito associado ao usuário, e a forma como isso é definido é particular a cada NSP.
- ***User Group***: um cliente pode definir vários grupos de usuários, onde os membros de um grupo têm o mesmo Perfil de Usuário (*hasUserProfile*). Por exemplo, a empresa ABC poderia definir os grupos *TI e Vendas*, com diferentes permissões sobre os serviços de rede. Cada grupo está também associado a um processo de contabilidade de uso (*hasAccounting*), para o consumo agregado de todos os seus usuários, também particular a cada NSP.
- ***User Profile***: descreve os serviços que os usuários estão autorizados a requisitar, seus limites de consumo e escopo. Todos os usuários de um grupo de usuários têm o mesmo perfil de usuário, que especifica um conjunto de direitos de usuário (*hasUserRights* *). Note que vários grupos de usuários podem ser associados ao mesmo perfil. Cada perfil de usuário caracteriza um tipo de usuário diferente, quanto às necessidades de uso de diferentes tipos de serviços contratados pelo cliente. Mantendo separados os conceitos de grupo (equipe) e perfil de usuário (tipo de usuário), o modelo permite um pequeno número de perfis de usuário (por questões de escalabilidade), enquanto possibilita uma granularidade fina na limitação de consumos de grupos;

- ***User Right***: especifica o direito do usuário para um serviço específico (*hasServiceID*), escopo definindo os endereços origem e destino admitidos (*hasScope*), e os períodos onde essa permissão é válida (*hasServiceSchedule*). Além disso, o direito do usuário pode especificar limites de consumo individuais (*hasIndividualConsumptionLimit*) ou grupais (*hasGroupConsumptionLimit*). Esses limites são específicos do NSP, e podem especificar, por exemplo, a quantidade total de tráfego mensal (em bytes), tempo total de utilização (em minutos), ou valores monetários. O limite global de uso (incluindo todos os grupos de usuários) pode ser definido no objetivo associado (*hasAssociatedObjective*). Observe que todas as três formas de limitação de consumo são opcionais. Há dois tipos de direitos de usuário:
    - ***Unconstrained***: é um direito irrestrito, e o usuário com esse direito pode requisitar qualquer nível de QoS (para o tipo de serviço, agendamento e escopo especificados). Suas requisições são somente limitadas pelas políticas do provedor e pela disponibilidade dos serviços da rede. A invocação deste tipo de serviço pode implicar em uma renegociação de SLA, que está fora do escopo desta dissertação.
    - ***Constrained***: é um direito limitado. As requisições de QoS feitas por usuários com direito limitado somente serão aceitas se estiverem em conformidade com o SLA contratado. Um direito restrito é associado com um SLS (*hasAssociatedObjective*) e define os limites mais restritos para um

ou mais parâmetros dos serviços especificados neste SLS para este tipo de usuário.

- ***Service* ID**: Identificador do serviço, que contém o tipo do serviço (*hasServiceType*), um atributo enumerado que indica um dos tipos de serviço suportados pelo NSP, e o identificador de fluxo (*hasFlowIdentifier*), com os parâmetros de identificação dos pacotes IP desse fluxo.

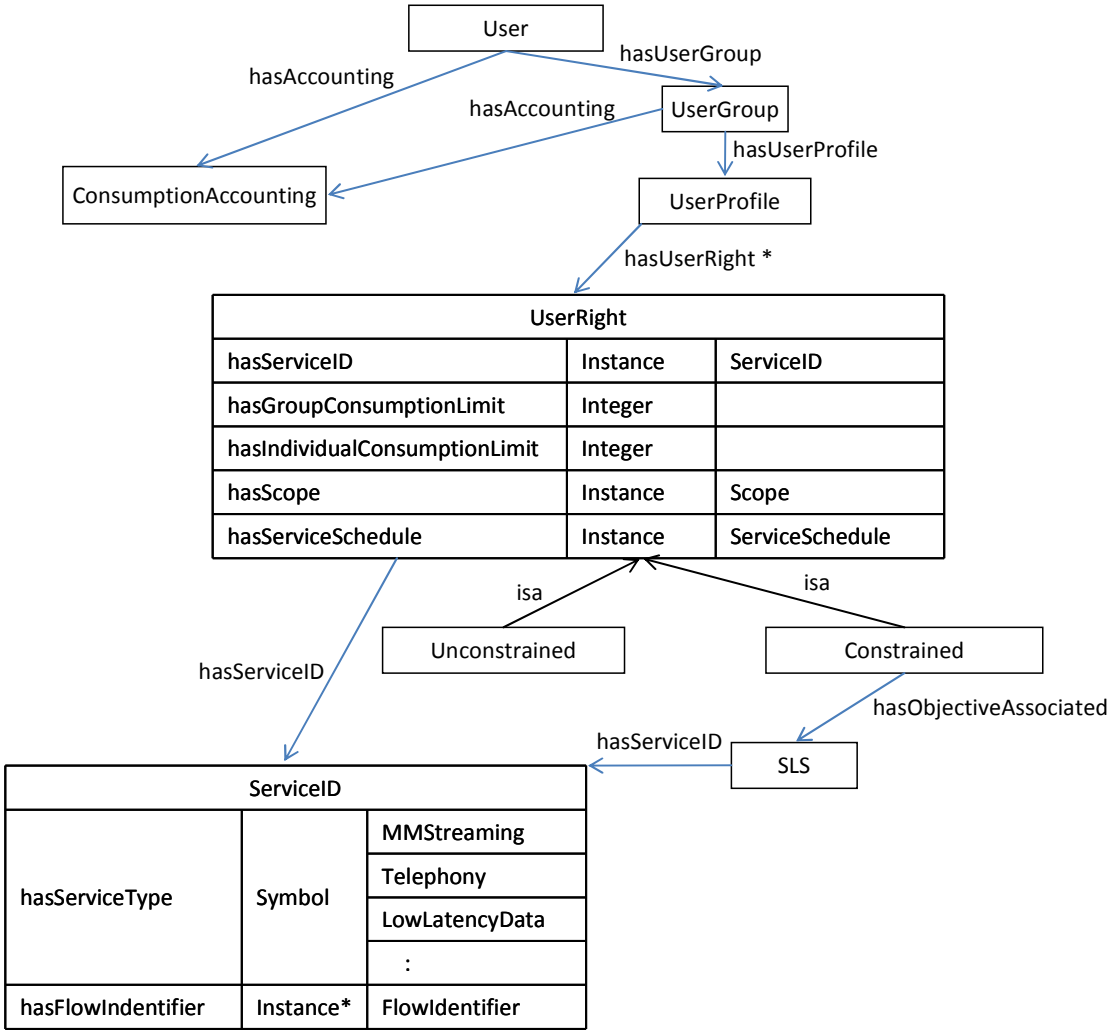
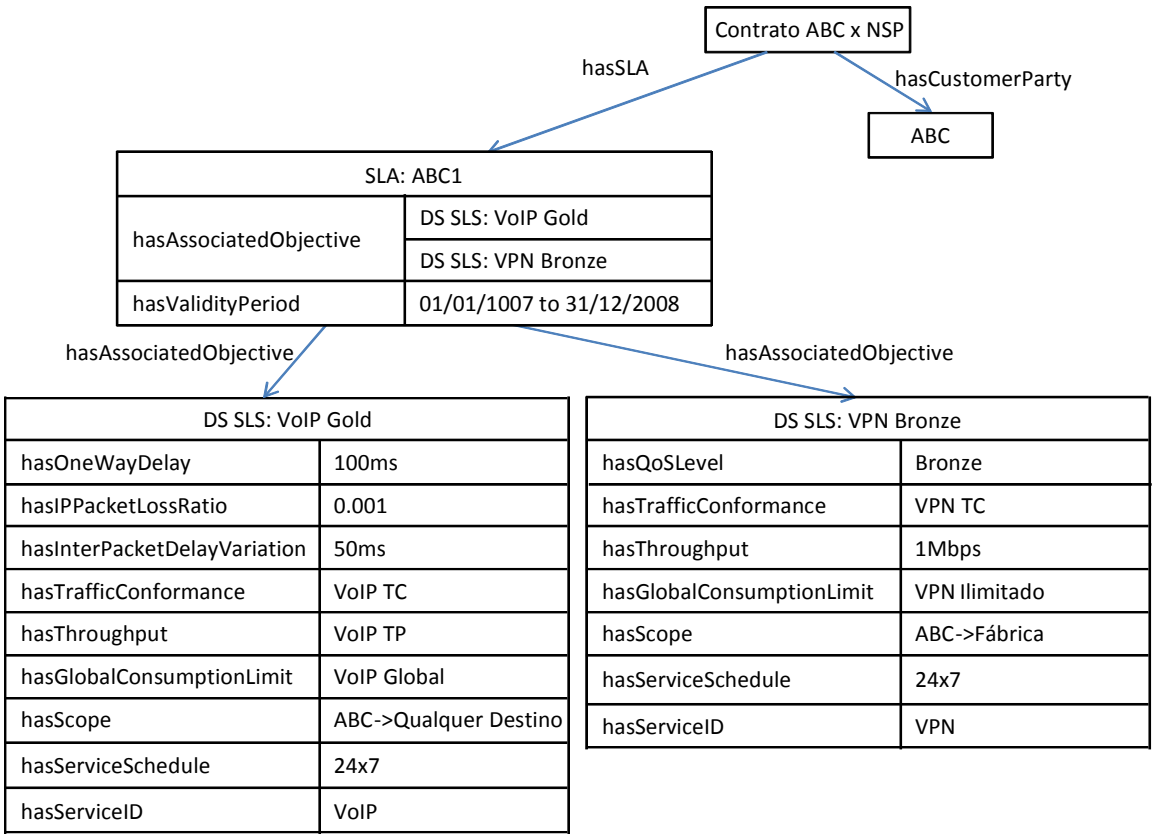


FIGURA 7. Ontologia de Perfil de Usuário

5.3. Exemplo Ilustrativo

Essa sessão apresenta um exemplo ilustrativo que mostra a aplicabilidade do modelo de autorização orientado a perfis de usuário proposto. Este exemplo considera novamente o cliente ABC que possui além da matriz uma fábrica remota. Neste exemplo, é considerado que ABC contrata um NSP que adota os conceitos especificados na Fig. 6 para estabelecer seus contratos de serviço. A Fig. 8 apresenta dois dos SLSs negociados:

- **VPN Bronze**: Um SLS de serviços diferenciados que define o nível de qualidade para a VPN entre a ABC e a fábrica remota. Estes níveis de qualidade são especificados em termos de limites qualitativos para determinados parâmetros de desempenho de rede. Também são especificados os TC (*Traffic Conformance*), escopo e período de validade;
- **VoIP Gold**: Um SLS de serviço diferenciado que define o nível de qualidade para o tráfego de VoIP partindo do roteador da ABC para qualquer destino. O atraso máximo admitido em um sentido é de 100ms, a taxa de perdas de pacotes é 0,001 (0,1%), e a variação de atraso máxima é de 50ms.



DS SLS: VoIP Gold	
hasOneWayDelay	100ms
hasIPPacketLossRatio	0.001
hasInterPacketDelayVariation	50ms
hasTrafficConformance	VoIP TC
hasThroughput	VoIP TP
hasGlobalConsumptionLimit	VoIP Global
hasScope	ABC->Qualquer Destino
hasServiceSchedule	24x7
hasServiceID	VoIP

DS SLS: VPN Bronze	
hasQoSLevel	Bronze
hasTrafficConformance	VPN TC
hasThroughput	1Mbps
hasGlobalConsumptionLimit	VPN Ilimitado
hasScope	ABC->Fábrica
hasServiceSchedule	24x7
hasServiceID	VPN

FIGURA 8. Parte do contrato de serviço entre ABC e seu NSP

Cada um dos SLS apresentados na Fig. 8 especifica os consumos globais, independentes de qual usuário está gerando o tráfego. Para gerenciar o uso destes serviços, o primeiro passo é classificar os usuários em grupos. Os empregados da ABC foram divididos nos grupos *TI* e *Vendas*. Para atender seus objetivos comerciais, o grupo *TI* precisa se comunicar com a fábrica e realizar as tarefas de manutenção remota, e o grupo *Vendas* precisa falar muito com clientes, e eventualmente com a fábrica, mesmo fora do horário de expediente.

A Fig. 9 mostra a especificação de dois Perfis de Usuário: *Perfil de Usuário de Vendas* e *Perfil de Usuário de TI*. Nesse exemplo, o cliente ABC autoriza 4 usuários,

agrupados nos departamentos de Vendas (*Maria, e Marcos*) e no departamento de TI (*Júlio e Roberto*). Cada usuário tem sua própria contabilidade (*hasAccounting*) e é associado ao grupo de usuários (*hasUserGroup*) *Departamento de TI* ou *Departamento de Vendas*. Cada grupo de usuários tem seu próprio perfil de usuário (*hasUserProfile*) e contabilidade de consumo (*hasAccounting*) para controlar o consumo agregado de todos os usuários desse grupo. O departamento de vendas é associado ao *Perfil de Usuário de Vendas* (*hasUserProfile*) que indica quais serviços podem ser requisitados pelos usuários do departamento (*hasUserRight*). A mesma idéia vale para o grupo de usuários do departamento de TI.

O *Perfil de Usuário de Vendas* contém dois direitos de usuário:

- **Direito de Usuário VPN Vendas**: direito de usuário restrito que especifica os limites de consumo individual e grupal definidos pelo SLS *VPN Bronze*;
- **Direito de Usuário VoIP Vendas**: direito de usuário irrestrito, que permite aos membros do grupo *Vendas* negociar qualquer nível de qualidade para o tráfego VoIP em qualquer horário, e para qualquer destino. Quando um membro do grupo *Vendas* invoca um serviço de VoIP, se essa invocação não pode ser atendida sob o SLS *VoIP Gold*, pode ser iniciada uma negociação dinâmica, que está fora do escopo desta dissertação.

O Perfil de Usuário de TI contém dois direitos de usuário:

- **Direito de Usuário VPN TI**: direito de usuário restrito que especifica os limites de consumo individual e grupal para o serviço definido pelo SLS *VPN Bronze*;

- **Direito de Usuário VoIP TI**: direito de usuário restrito que especifica os limites de consumo individual e grupal e limita o escopo (*ABC => Fábrica*) do serviço definido pelo SLS *VoIP Gold*.

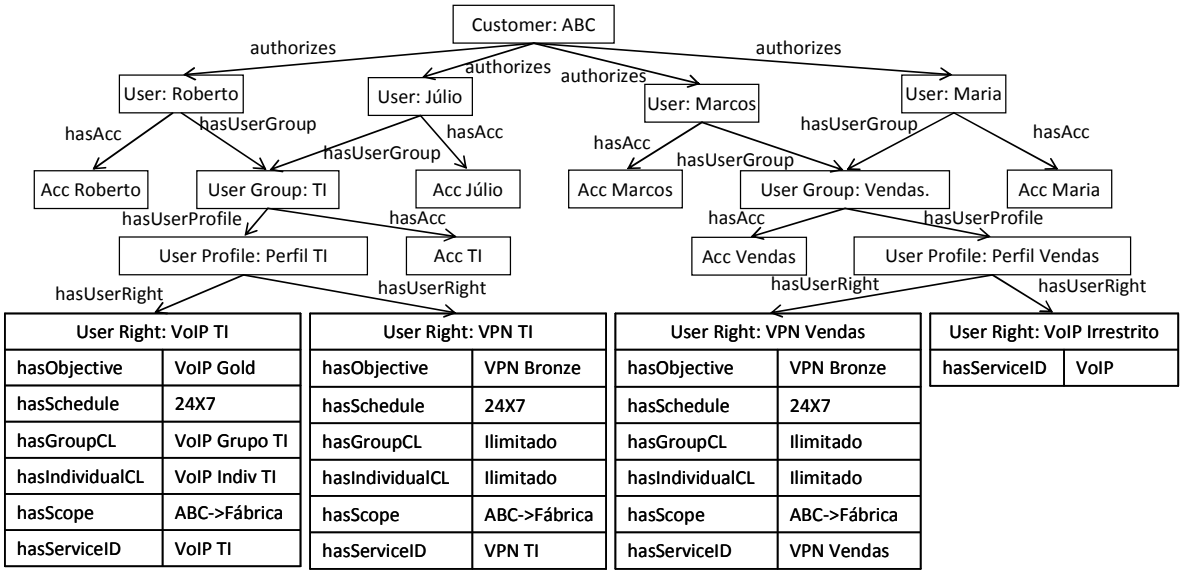


FIGURA 9. Exemplos de Perfil de Usuário

5.4. Linguagem de Especificação de Perfil de Usuário

Nessa proposta, os perfis de usuário são especificados usando RDF. Assim, os perfis de usuário, incluindo os seus direitos podem ser armazenados no Repositório de Políticas como arquivos RDF, ou como um mapeamento do grafo RDF em bancos de dados relacional, aos moldes do exemplo de persistência encontrado em [Jena 2008]. Quando o usuário se registra no sistema, recebe o seu perfil, que contém os seus direitos de uso. O modo como ele será expresso dependerá da arquitetura de QoS em uso. No caso do EuQoS, o usuário receberá o seu perfil em formato XML. No momento da requisição de serviços

com QoS, o próprio terminal do usuário pode fazer uma verificação prévia, para selecionar um serviço entre os que o usuário tem direito de acessar. A Fig. 10 mostra um exemplo de um direito de uso para o perfil TI, envolvendo os direitos “VoIP TI” e “VPN TI”, descritos na Fig. 9, e a Fig. 11 traz um exemplo de definição para o SLS VoIP Gold.

```
<?xml version = "1.0" ?>
<UserRight name = "VoIP TI">
   <hasAssociatedObjective>VoIP Gold</hasAssociatedObjective>
   <hasScope>ABC->Fabrica</hasScope>
   <hasGroupCL>10000</hasGroupCL>
   <hasIndividualCL>7000</hasIndividualCL>
   <hasServiceSchedule>24X7</hasServiceSchedule>
</UserRight>
<UserRight name = "VPN TI">
   <hasAssociatedObjective>VPN Bronze</hasAssociatedObjective>
   <hasScope>ABC->Fabrica</hasScope>
   <hasGroupConsumptionLimit>Ilimitado</hasGroupConsumptionLimit>
   <hasIndividualConsumptionLimit>Ilimitado
                                 </hasIndividualConsumptionLimit>
   <hasServiceSchedule>24X7</hasServiceSchedule>
</UserRight>
```

FIGURA 10. Arquivo XML com os direitos de uso do perfil TI

```
<?xml version = "1.0" ?>
<SLS name = "VoIP Gold">
   <hasServiceID>Telephony</hasServiceID>
   <hasOneWayDelay>150</hasOneWayDelay>
   <hasIpPacketLossRatio>0.001</hasIpPacketLossRatio>
   <hasInterPacketDelayVariation>50</hasInterPacketDelayVariation>
   <hasTrafficConformance>VoIP TC</hasTrafficConformance>
   <hasThroughput>30000</hasThroughput>
   <hasGlobalConsumptionLimit>
      <Value>30000</Value>
      <Unit>Minutes</Unit>
   </hasGlobalConsumptionLimit>
   <hasScope>ABC->Fabrica</hasScope>
   <hasServiceSchedule>24x7</hasServiceSchedule>
</SLS>
```

FIGURA 11. Arquivo XML com a definição do SLS VoIP Gold

Como já apresentado, cada direito de usuário restrito é associado a um SLS. Este direito pode definir limites ainda mais restritos para um ou mais parâmetros dos serviços especificados neste SLS para este tipo de usuário. Assim, o Repositório de Políticas pode

manter um conjunto de SLSs específicos para cada perfil de usuário. Estes SLSs específicos têm os valores dos parâmetros obtidos pela aplicação das restrições definidas no *User Right* aos valores contidos nos SLSs globais. Por exemplo, a Fig. 12 apresenta o SLS específico para o serviço VoIP para o grupo TI. Para a obtenção deste SLS específico foi combinado o direito de usuário TI VoIP (Fig. 9) ao SLS VoIP Gold (Fig. 8). Essa especificação de serviço combinada é construída no momento da contratação do serviço, e atualizada a cada atualização de definição de direitos de usuários, mas não no momento da autorização de serviços.

DSSLS: VoIP Gold (Para grupo TI)	
hasOneWayDelay (ms)	100ms
hasIpPacketLossRatio	0.001
hasInterPacketDelayVariation (ms)	50ms
hasTrafficConformance	VoIP TC
hasThroughput (bps)	30.000
hasGlobalConsumptionLimit (min)	20.000
hasGroupConsumptionLimit (min)	10.000
hasIndividualConsumptionLimit (min)	7.000
hasScope	ABC->Fabrica
hasServiceSchedule	24x7
hasServiceID	VoIP

FIGURA 12. SLS Específico TI VoIP

5.5. Escalabilidade

A escalabilidade do modelo proposto depende do número de usuários, grupos, perfis de usuário e SLSs. O número de direitos de usuário, no pior caso, é igual ao número de perfis de usuário multiplicado pelo número de SLS. Como cada perfil de usuário representa

um tipo diferente de usuário, de acordo com a sua necessidade de usar diferentes tipos de serviços de rede, o número de perfis de usuário é normalmente baixo, mesmo para empresas com grande número de usuários. Por exemplo, uma empresa com 10.000 usuários pode ter menos de 10 perfis de usuários diferentes, cada um associado com um conjunto diferente de direito de usuários. Em termos de escalabilidade, essa característica o torna semelhante ao DiffServ, onde um grande número de fluxos (usuários) é agrupado em um pequeno número de classes de serviço (perfis de usuário).

Outro ponto importante do ponto de vista da escalabilidade é a manutenção dos registros de log dos serviços utilizados. Para a limitação do consumo individual, é necessário manter um registro contábil para cada serviço que cada usuário pode requisitar. O pior caso ocorre quando todos os usuários podem requisitar todos os serviços contratados. Mesmo em uma grande empresa, por exemplo, com 100.000 usuários, e 100 SLSs diferentes, o número total de registros é 10.000.000, que é perfeitamente suportado por um banco de dados em uma única máquina. As atualizações desses registros contábeis devem ocorrer a cada vez que um usuário inicia ou encerra uma sessão com QoS. Como os usuários tipicamente usam uma única sessão (ou poucas sessões) com QoS a cada instante, e cada sessão dura, em média, mais de um minuto, o processo de contabilidade também não é um problema, pois para o caso mencionado acima, o pior caso seria de 200.000 atualizações por minuto. As mesmas regras valem para a limitação de consumo por grupo, considerando ainda que o número de grupos é menor que o número de usuários.

5.6 Considerações Finais

Este capítulo apresentou um modelo de perfil de usuário que suporta a flexibilidade desejada para clientes corporativos. O modelo é formalizado como uma ontologia, e permite a visualização das definições de serviços, usuários, agrupamentos de usuários, e suas associações com perfis de usuários com diferentes permissões de uso dos serviços contratados. São apresentados exemplos ilustrativos de implementação de definições de serviços, direitos de uso e agrupamentos de usuários, além de requisições de serviço. Também são discutidos aspectos referentes à escalabilidade do modelo.

O capítulo seguinte apresenta uma forma de validação do modelo proposto através de um protótipo de testes.

6. EXEMPLO DE USO E VALIDAÇÃO DO MODELO PROPOSTO

O capítulo anterior apresenta o modelo do perfil de usuário que suporta a distinção entre os conceitos de usuário e cliente, bem como oferece flexibilidade na definição de restrições de uso dos serviços contratados. Este capítulo tem por objetivo descrever uma validação do modelo proposto, via a implementação e teste de um protótipo fazendo uso do modelo de perfil proposto.

Primeiramente é apresentada uma seqüência de passos que pode ser seguida para processar uma autorização de requisição de serviço, usando como referência a arquitetura EuQoS. Em seguida o protótipo de teste implementado é descrito, incluindo sua arquitetura e o mapeamento da ontologia para banco de dados relacional. Na seqüência é apresentado o cenário de testes utilizado, e as alterações necessárias para expandir o EuQoS para suportar o modelo de perfil de usuários proposto.

6.1. Processo de Autorização

Para explicar o uso do perfil de usuário proposto, esta seção apresenta uma possível seqüência de passos para processar uma autorização de requisição de serviço na arquitetura EuQoS estendida para implementar o perfil do usuário proposto. Conforme apresentado no capítulo 4, a arquitetura EuQoS é orientada a sessões. Antes de iniciar uma transmissão de dados com QoS, a aplicação precisa estabelecer uma sessão com QoS, que dispara a reserva de recursos e a contabilidade.

Para este exemplo, considere o usuário Júlio (na companhia ABC) requisitando um

serviço VoIP para a fábrica remota, com codec de áudio G.729, segunda-feira às 10:10 da manhã. Esses parâmetros são traduzidos para uma descrição de sessão detalhada (para efeitos de autorização e CAC), pelo módulo cliente do EuQoS, com a ajuda do proxy EQ-SIP. Para autorizar a sessão com QoS, o módulo A-SSN (cliente DIAMETER) envia através de uma mensagem DIAMETER (QAR – *QoS Authorization Request*), a requisição de estabelecimento de sessão ao módulo SAAA (servidor DIAMETER), incluindo as informações apresentadas na Fig. 13 [Baresse 2005]. O Formato da mensagem QAR do DIAMETER é apresentado na Fig. 14.

Service Request VoIP Julio	
UserName =	julio@abc
SourceIP =	A.B.C.D
DestinationIP =	E.F.G.H
SourcePort =	2300
DestinationPort =	5060
TransportProtocol =	RTP
CodecName =	G.729
ApplicationType =	Telephony
MaxDelay =	100
MaxJitter =	50
MaxLossRatio =	0.0010
MaxBitRate =	30000

FIGURA 13. Características do Serviço Solicitado

Quando o SAAA recebe a requisição (QAR), ele precisa identificar o Grupo de Usuário e Perfil de Usuário associados ao usuário *Júlio* (no exemplo, *TI* e *Perfil de Usuário de TI*). O SAAA precisa confrontar as características do serviço (Fig. 8) com o SLS específico para estes usuário e serviço (Fig. 7). Este serviço será autorizado se: o escopo permitido inclui os endereços de origem e destino (*A.B.C.D* e *E.F.G.H*); os parâmetros de

desempenho de QoS requisitados não são maiores do que os especificados; a data e hora da requisição (*segunda-feira, 10:10 da manhã*) é permitida pelo agendamento; o consumo contabilizado para o usuário *Júlio* ainda não chegou ao seu limite individual de consumo para esse serviço; e consumo de VoIP contabilizado para o grupo *TI* é menor que o limite definido para o grupo *TI* para o serviço de VoIP; e o consumo global de VoIP contabilizado para todos os usuários da ABC é menor que o limite de consumo para o serviço VoIP. Ao final do processo, o SAAA envia a resposta (QAA – *QoS Authorization Answer*) ao A-SSN com o resultado.

QoS-Request		
Diameter Header		(Cod. QAR), REQ, PXY
Session-Id		Note4.abc.com;1874123412;12
Auth-Application-Id		(Cod. SAAA)
Origin-Host		A.B.C.D (IP Address)
Origin Realm		ABC
Destination-Realm		Factory
Auth-Request-Type		Authorize_only
Destination-Host		E.F.G.H (IP Address)
User-Name		julio@abc
QoS-Profile		G.729
QoS-Semantics		Minimum-QoS
QoS-Parameters	Rate	30000 (30Kbps)
	Bucket-size	1024 (1KB)
	Peak rate	30000 (30Kbps)
	Minimum policed unit	1024 (1KB)
	Path Latency	100000us (100ms)
	Path Jitter	50000us (50ms)
	Path Packet Loss Ratio	0,001 (10^{-3})

FIGURA 14: Formato da mensagem DIAMETER QAR para uma chamada VoIP no EuQoS

6.2. Protótipo de teste

Para testar o modelo proposto, foi implementado um protótipo de um sistema de

autorização, em Java, usando a API Jena [Jena 2008] para fazer o carregamento das ontologias específicas do provedor para uma base de dados relacional, utilizando o banco de dados MySQL.

As informações de contabilidade, que estão associadas a um direito de uso e ao usuário/grupo/serviço, conforme suas limitações de consumo sejam individuais, grupais ou globais, são armazenadas diretamente em tabelas do bando de dados MySQL, e acessadas com auxílio da biblioteca java.sql. O protótipo permite incluir novos SLSs e direitos de usuários a partir de arquivos RDF e também efetua a autorização de uma requisição de serviço com QoS, a partir da especificação do usuário e do cliente (no formato usuário@cliente), da identificação do serviço requisitado, do endereço de origem e destino da solicitação e dos parâmetros de desempenho requisitados. Caso os parâmetros do serviço requisitado estejam dentro dos limites contratados, e o consumo (individual, grupal e global) não tenha sido ultrapassado, a autorização é requisitada. Em caso contrário é rejeitada.

O protótipo permite também a inclusão de novos SLSs, com parâmetros novos de desempenho, que são validados sem necessidade de alteração do código.

6.2.1 Arquitetura do protótipo

A arquitetura para a implementação do modelo segue a estrutura ilustrada na figura 15, que contém um módulo para a inclusão de novos SLA/SLSs, uma base de dados para o armazenamento dos SLA/SLSs, uma base de dados para as informações de uso (*Accounting*), e os módulos cliente e servidor de AAA:

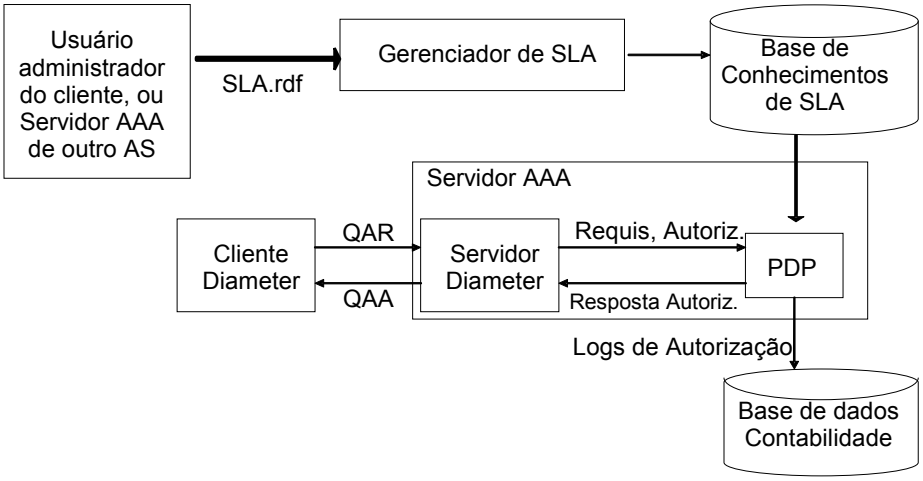


FIGURA 15: Arquitetura para a implementação do modelo

- Gerenciador de SLA: Permite a inclusão de novos SLA/SLS descritos em arquivos RDF. O módulo Gerenciador de SLA interpreta o arquivo RDF e atualiza a base de dados. Esse arquivo RDF pode ser gerado por uma interface de uso do cliente ou por um servidor AAA de outro AS, no caso de usuário visitante. No protótipo desenvolvido, essa carga de arquivos RDF é realizada com auxílio da API Jena [Jena 2008].
- Base de Conhecimentos de SLA: A base de conhecimentos contendo os SLAs assinados pelos clientes, incluindo as definições de serviços, suas especificações técnicas (SLS), clientes, usuários, grupos de usuários, perfis de usuários e as respectivas permissões de uso dos serviços contratados. No protótipo implementado, a base de dados é suportada por um banco de dados MySQL, com auxílio da API Jena.
- Base de Dados de Contabilidade: É a base de dados que contém o registro das requisições de serviço autorizadas, informadas pelo PDP (*Policy*

Decision Point). Essas informações são utilizadas para a limitação de consumo, usadas como base para futuras autorizações de serviço, e para fins de faturamento.

- Servidor AAA: O servidor AAA contém dois sub-módulos: O Servidor DIAMETER e o PDP.
    - Servidor DIAMETER: É o módulo que recebe as requisições de QoS no formato do DIAMETER, como QAR (*Qos Authorization Request*), identifica os parâmetros da requisição, e compõe um objeto java com os parâmetros da requisição, e o submete ao PDP. Recebendo a resposta do PDP, o DIAMETER Server compõe a resposta QAA (*Qos Authorization Answer*) e a envia ao Cliente DIAMETER. Esse módulo não foi implementado no protótipo de teste.
    - PDP (*Policy Decision Point*): É o módulo que decide se a requisição pode ser aceita ou não, a partir de um objeto Java com os parâmetros da requisição. Utiliza a base de conhecimentos de SLA e a base de dados de contabilidade. Caso a requisição seja aceita, é gerado um registro de log de autorização, gravado na base de dados de contabilidade. O protótipo permite a definição de novas subclasses de SLS, com qualquer número de parâmetros de desempenho quantitativos adicionais, sem a necessidade de alteração do código do PDP.

- Cliente DIAMETER: É o cliente DIAMETER, no equipamento do usuário, ou em algum proxy, dependendo da arquitetura de QoS utilizada. É o responsável pela formatação da requisição de QoS em mensagens DIAMETER, e pela comunicação com o DIAMETER Server. Este módulo também não foi implementado no protótipo de teste.

6.2.2 Armazenamento do SLA em base de dados

A definição de SLA/SLS do capítulo 5 foi mapeada para o armazenamento em base de dados MySQL, utilizando como referência para popular a base de dados o caso do provedor hipotético ilustrado no exemplo do item 5.3. A ontologia foi simplificada e adaptada para incluir os dados necessários para a autorização de uma requisição em um único provedor, conforme descrito no item 6.1.

O mapeamento foi realizado seguindo a aplicação-exemplo de persistência disponível na documentação da API Jena [Jena 2008]. A API Jena permite o acesso aos dados da ontologia de forma padronizada, independente de eles estarem armazenados em memória após a carga de um arquivo texto em formato RDF, ou então a partir de uma base de dados.

O protótipo utiliza *queries* compostas em SPARQL [W3C 2008] nos métodos criados para acesso aos dados. A Figura 16 ilustra a título de exemplo o código do método que carrega os dados de um escopo.

```
public Scope getScope(String id){
  Scope sc = new Scope();
  try {
    String queryStr="PREFIX refURI:<http://protege.stanford.edu/rdf>"+
      " SELECT ?name ?srcip ?srcnetmask ?dstip ?dstnetmask " +
      "  WHERE { " +
      "        ?x refURI:hasTitle \"" + id + "\" . " +
      "        ?x refURI:hasTitle ?name . " +
      "        ?x refURI:hasSourceIp ?srcip . " +
      "        ?x refURI:hasSourceNetmask ?srcnetmask . " +
      "        ?x refURI:hasDestinationIp ?dstip . " +
      "        ?x refURI:hasDestinationNetmask ?dstnetmask ." +
      "      } ";
    Query query = QueryFactory.create(queryString);
    QueryExecution qe = QueryExecutionFactory.create(query, model);
    ResultSet rs = qe.execSelect();
    if (rs.hasNext()) {
      QuerySolution qs = rs.nextSolution();
      sc.setName (qs.get("name").toString());
      sc.setSourceIp (qs.get("srcip").toString());
      sc.setSourceNetmask (qs.get("srcnetmask").toString());
      sc.setDestinationIp (qs.get("dstip").toString());
      sc.setDestinationNetmask (qs.get("dstnetmask").toString());
    }
    return sc;

  } catch (Exception e) {
    e.printStackTrace();
  }
  return sc;
}
```

FIGURA 16: Exemplo de código para acesso aos dados

A contabilidade de consumo é armazenada na base de dados, em três tabelas, sendo a primeira para o registro de consumo individual, com os campos de identificação do direito de uso, do usuário e do consumo registrado, a segunda para o registro do consumo por grupo, identificando o grupo, o direito de uso e o consumo, e a terceira para o consumo global do serviço, identificando o sls e o consumo global. A limitação de consumo é implementada comparando a limitação definida na ontologia contra o consumo registrado na base de dados.

A limitação de escopo é realizada por um método que valida se os endereços de origem e destino especificados na requisição se encaixam em algum escopo definido para o

direito de uso do usuário e para o SLS, no caso de direito de uso restrito.

De forma semelhante, a limitação de agendamento é definida no *sls* e no *userright*, indicando uma instância de *schedule*, que define intervalos de validade em termos de horas do dia, mês do ano e dia da semana. A validação é realizada com base na data e hora em que a requisição é recebida.

6.2.3 Critérios de Validação

Para ser considerado válido o modelo precisa suportar a flexibilidade desejada descrita ao longo deste trabalho, permitindo a clara separação entre os conceitos de usuário e cliente, bem como as limitações de direitos de uso e de consumo por serviço contratado, por usuário, por grupo ou por qualquer combinação destas. Além disso, deve oferecer a possibilidade de inclusão de novos parâmetros de desempenho a um SLS, sem a necessidade de alteração de código do protótipo.

6.3. Cenário de testes

Para o teste do protótipo foi utilizado um cenário populando a base de conhecimentos com um SLS com as mesmas características ilustradas nos itens 5.3 e 5.4, conforme ilustra a Fig. 17.

Perfis de usuário: Foram incluídos os usuários Julio e Roberto, pertencentes ao grupo TI, que está associado ao perfil TI. O perfil de TI possui o direito de uso VoIP TI, de acordo com a Fig. 18.

hasName	VoIP Gold
hasServiceId	VoIP
hasOneWayDelay (ms)	100
hasInterPacketDelayVariation (ms)	50
hasIpPacketLossRatio	0,001
hasThroughput (bps)	30000
hasScope	ABC -> Qualquer Destino (200.100.50.0/24 -> 0.0.0.0/0)
hasSchedule	24X7
hasGlobalConsumptionLimit	20
hasLimitationUnit	Minute

FIGURA 17: SLS usado no cenário de testes

hasName	VoIP TI
hasAssociatedObjective	VoIP Gold
hasScope	ABC -> Fabrica (200.100.50.0/24 -> 200.150.50.0/24)
hasSchedule	Comercial (segunda a sexta, 08:00–18:00)
hasIndividualConsumptionLimit	7
hasGroupConsumptionLimit	10

FIGURA 18: Direito de uso definido para o cenário de testes

Desta forma, as restrições impostas às solicitações de serviço por parte de um usuário do grupo TI são resultado da combinação das restrições do SLS com as restrições do direto de uso, de acordo com a Fig. 19.

hasOneWayDelay (ms)	100
hasInterPacketDelayVariation (ms)	50
hasIpPacketLossRatio	0,001
hasThroughput (bps)	30000
hasScope	ABC -> Fabrica (200.100.50.0/24 -> 200.150.50.0/24)
hasSchedule	Comercial (segunda a sexta, 08:00–18:00)
hasGlobalConsumptionLimit	20
hasIndividualConsumptionLimit	7
hasGroupConsumptionLimit	10
hasLimitationUnit	Minute

FIGURA 19: Restrições combinadas entre o SLS VoIP Gold e o Direito de uso VoIP TI

Para testar a funcionalidade do protótipo foram realizados testes para validar cada um dos parâmetros que restringem o direito do usuário em requisitar um serviço. O protótipo não inclui os testes de autenticação, que valida a identidade do usuário. O primeiro teste realizado foi requisitando um serviço que o usuário não possui direito de requisitar.

O segundo teste avaliou os parâmetros de QoS da requisição (vazão, atraso, variação do atraso e taxa de perda de pacotes). A vazão solicitada não pode ser maior do que o definido pelo SLS, enquanto o atraso, variação de atraso e taxa de perdas solicitados não podem ser menores do que o definido pelo SLS.

Em seguida foram feitos os testes de escopo, chamando destinatários dentro e fora do escopo permitido, e a partir de endereços de origem dentro e fora do escopo permitido.

Passado o teste do escopo, passou-se a testar o enquadramento ao agendamento de disponibilidade do serviço, usando como referência o instante da solicitação da requisição.

O teste seguinte buscou identificar a capacidade de tratamento de novos parâmetros de desempenho no SLS, sem a alteração do protótipo. Para isso foram incluídos os AdmittedRangeValues: (i) AtrasoVoIPPremium, com minValue=130.0, (ii) JitterVoIPPremium, com minValue=60.0, (iii) BandaVoIPPremium, com maxValue=90000.0 e (iv) ValoresPorFaixa, com minValue=1000.0 e maxValue=5000.0. (Fig. 20) Em seguida foi criada uma nova classe de SLS, chamada *Real Time Quantitative DS SLS,* uma subclasse de *Quantitative DS SLS*. Nessa classe foram adicionadas as propriedades bandaMinima, atrasoGarantido, variacaoAtraso e faixaValores, todas do tipo instâncias de AdmittedRangeValues (Fig. 21). Em seguida foi adicionada uma instância

para essa classe, para o serviço VoIP Premium, associando essas quatro propriedades aos valores correspondentes adicionados de AdmittedRangeValues. Em seguida a requisição foi alterada para incluir os novos parâmetros, conforme ilustrado na Fig. 22, com os nomes das propriedades do SLS, e com valores dentro e fora do admitido.

AdmittedRangeValue		
hasParameterName	minValue	maxValue
AtrasoVoIPPremium	130.0	
JitterVoIPPremium	60.0	
BandaVoIPPremium		90000.0
ValoresPorFaixa	1000.0	5000.0

FIGURA 20: Instâncias da classe AdmittedRangeValues, com parâmetros adicionais de desempenho

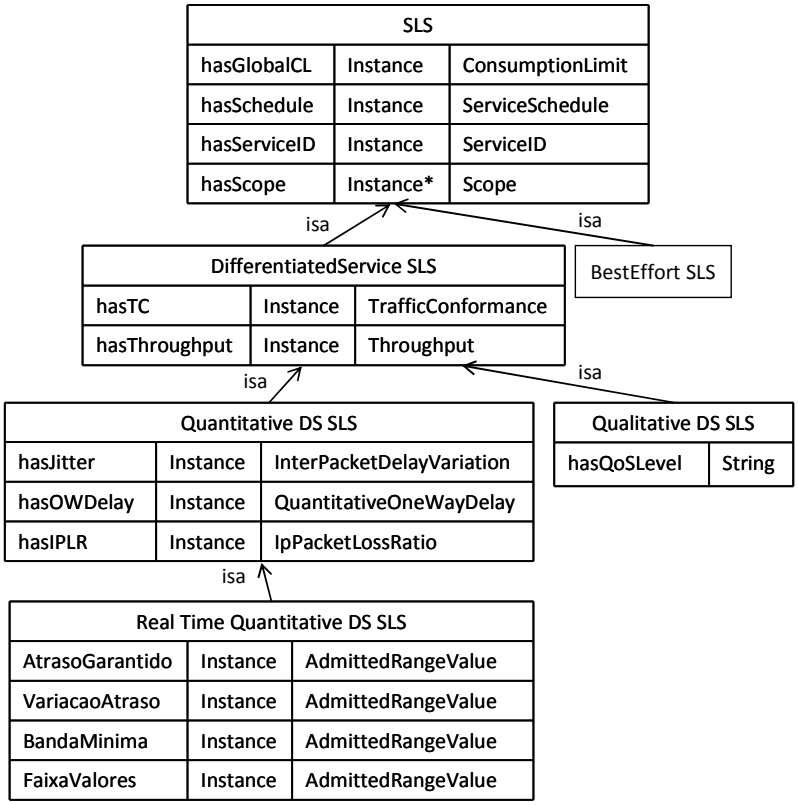


FIGURA 21: Ontologia de SLS incluindo uma nova subclasse *Real Time Quantitative DS SLS* com quatro parâmetros de desempenho adicionais

Finalmente foram realizados os testes de limitação de consumo individual e grupal. Para a realização dos testes, foi considerado que cada requisição equivaleria ao consumo de uma unidade de tempo (minuto). Em caso de uma aplicação de produção real, as duas abordagens suportadas pelo DIAMETER são (i) a autorização de um serviço por um tempo informado, que precisa ser renovada antes que esse prazo expire, ou (ii) o simples registro do momento em que a autorização foi concedida, (considerado início de sessão) atualizando periodicamente o consumo durante a sessão e no momento do encerramento da sessão.

Os resultados obtidos em todos os testes de autorização de requisições de serviços com QoS estavam de acordo com o esperado, considerando-se os parâmetros e a seqüência de testes.

Real Time Quantitative DS SLS: VoIP Premium	
hasOneWayDelay (ms)	100
hasIpPacketLossRatio	0.001
hasInterPacketDelayVariation (ms)	50
hasTrafficConformance	VoIP TC
hasThroughput (bps)	30000
hasGlobalConsumptionLimit	20
hasLimitationUnit	Minute
hasConsequence	Marcar como VoIP Gold
hasScope	ABC->Fabrica
hasServiceSchedule	24x7
hasServiceID	VoIP
bandaMinima	BandaVoIPPremium
atrasoGarantido	AtrasoVoIPPremium
variacaoAtraso	JitterVoIPPremium
faixaValores	ValoresPorFaixa

FIGURA 22: Exemplo de SLS com parâmetros de desempenho adicionais, usado nos testes

6.4. Expansão do EuQoS para adequação ao modelo proposto

No EuQoS a autorização das sessões é realizada pelo módulo SAAA, atendendo a uma requisição de serviço formatada pelo módulo A-SSN. O módulo A-SSN possui um cliente do SAAA, que se comunica com o servidor SAAA através do protocolo DIAMETER. O servidor SAAA realiza a autorização das requisições com base em um repositório de políticas, que inclui os SLS contratados. Esses SLSs são fornecidos pelo módulo CHAR.

O modelo de usuário do EuQoS não separa os conceitos de usuário e cliente, o que não permite o compartilhamento dos recursos de um SLS contratado por um cliente entre os diversos usuários desse cliente. Para superar essa limitação torna-se necessário implementar o conceito de cliente, separando-o de usuário, tanto na base de dados do CHAR, como no SAAA. No CHAR, essa separação entre usuário e cliente também permite que um usuário do cliente (com perfil de administrador) possa gerenciar todas as contas de usuários associadas à conta do cliente, adicionando, removendo ou alterando perfis de usuários. Deixar esse gerenciamento com o cliente torna a atualização de cadastro mais rápida e barata, dando agilidade ao negócio. As alterações na base de dados do SAAA podem ser baseadas nos conceitos e relações apresentadas nesta dissertação. O SAAA também terá que manter um status de consumo para os usuários e serviços de um cliente, para verificar, a cada autorização de início de sessão, se os limites previstos no SLS do cliente ainda não foram alcançados (se esses limites existirem).

O processo de decisão sobre autorizar ou não uma requisição, obviamente, também deve ser alterado para considerar as novas restrições que o cliente pode estabelecer a seus

usuários, em termos de limites de consumo, agendamento de disponibilidade e escopo, além das verificações já realizadas quando aos parâmetros de desempenho de QoS contratados.

6.5. Considerações Finais

Este capítulo apresentou um protótipo para testes do modelo proposto, expondo sua arquitetura e base de dados, assim como um cenário de testes, fornecendo assim um meio de validação da proposta. Apresentou também indicativos de como expandir a arquitetura do EuQoS para suportar o modelo proposto, tornando-o adequado ao suporte de clientes corporativos. O próximo capítulo apresenta as conclusões do trabalho e indicações de trabalhos futuros.

7 CONCLUSÃO

O modelo de perfil de usuário proposto suporta a separação entre os conceitos de usuário e cliente, permitindo uma maior flexibilidade para configurar os SLAs para clientes corporativos. Ele também permite agrupar os usuários, atribuindo-lhes direitos de usuário específicos, para requisitar alguns dos serviços contratados. Esses direitos de usuário podem ainda ser restringidos em termos de escopo, agendamento de disponibilidade e limitação de consumo. Os limites de consumo podem ser definidos em bases individuais, grupais, em um único limite para todos os usuários de um cliente, ou em qualquer combinação dessas opções. O modelo também permite a contabilização do consumo, para a verificação dos limites de consumo permitidos. A flexibilidade alcançada pelo modelo permite novas opções de configuração de controle de custos ao cliente de serviços de rede com invocação explícita de QoS. Dessa forma, esse modelo atende a todos os requisitos apresentados no capítulo 5.

Outro aspecto importante do modelo é a possibilidade de extensão e adaptação às características dos serviços oferecidos pela provedora, adequando a diferentes formatos de SLA/SLS, uma vez que o modelo é especificado como uma ontologia. É possível acrescentar novas definições de SLS, a partir das definições originais, incluindo novos parâmetros de autorização quantitativos, que serão validados sem a necessidade de alteração do código do protótipo de validação, atendendo aos critérios expostos no item 6.2.3.

A implementação da autorização pode ser feita sem grande complexidade, e é

escalável, mesmo com limitações de consumo individuais, grupais e globais, conforme discutido no item 5.5, e ilustrado a partir do protótipo de testes apresentado nos itens 6.2 e 6.3.

7.1 Trabalhos Futuros

- Realizar a expansão do sistema EuQoS para a implementação do modelo proposto;
- Implememementar o suporte a usuários móveis, aplicando a interpretação de conteúdo semântico, a fim de identificar a equivalência de serviços prestados por diferentes provedores, descritos de maneira diferente.

REFERÊNCIAS

[1] ALTMANN, J., et al. “How to market-manage a QoS network”. INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, Junho, 2002.

[2] ANGORI, E., MARFUTI, G. “Annex to D1.1.2: System Design: Functions, Interfaces Specification”, Maio, 2005. Disponível em http://www.euqos.eu/documents/ md_817_d112_annex_v2_pdf.zip; Capturado em agosto de 2006.

[3] BARESSE, L. et al. “Extended QoS API and Middleware layer for phase 1 application use-cases”. EuQoS Deliverable D3.1.1, WP3, Agosto, 2005.

[4] Beben, A. “EQ-BGP: an efficient inter-domain QoS routing protocol”. 20th International Conference on Advanced Information Networking and Applications (AINA 2006), Abril, 2006.

[5] BLAKE, S., et al. “An Architecture for Differenciated Services”. RFC 2475, Internet Engineering Task Force, Dezembro, 1998.

[6] BORIONI, S. et al. “SLA Management Handbook”. Tele Management Forum.

[7] BRADEN, R., CLARK, D., SHENKER, S. “Integrated Services in the Internet Architecture: An Overview”. RFC 1633, Internet Engineering Task Force, Junho, 1994.

[8] BRISCOE, B., RUDKIN, S. “Commercial Models for IP Quality of Service Interconnect”. BT Technology Journal, Volume 23, N. 2, Abril, 2005.

[9] BURAKOWSKI, W. et al. “On multi-domain Connection Admission Control in EuQoS System”. 15th IST Mobile & Wireless Communication Summit, Myconos (Grécia), Junho, 2006.

[10] CALHOUN, P. et al. “Diameter Base Protocol”. RFC 3588, Internet Engineering Task Force, Setembro, 2003.

[11] CALHOUN, P. et al. “Diameter Network Access Server Application”. RFC 4005, Internet Engineering Task Force, Agosto, 2005.

[12] CARROL B. (2004) “Cisco Access Control Security: AAA Administrative Service”. Cisco Press, Maio, 2004.

[13] CHAKRAVORTY, R. et al. “Dynamic SLA-based QoS control for third generation wireless networks: the CADENUS extension”, IEEE International Conference on Communications, Maio, 2003.

[14] CORTESE, G. et al. “CADENUS: Creation and Deployment of End-User Services in IP Premium Networks”, IEEE Communication Magazine, Janeiro, 2003.

[15] DAMILATIS, T., et al. “Final System Evaluation”. IST-1999-11253-TEQUILA Deliverable D3.4 Part B – WP1, Outubro, 2002.

[16] D’ARIENZO, M. et al. “Dynamic SLA-Based Management of Virtual Private Networks”. Thyrrhenian International Workshop on Digital Communications, Setembro, 2001.

[17] DUGEON, O., et al. “End to End Quality of Service over Heterogeneous Networks”. Network Control and Engineering for QoS, Security and Mobility (NetCon’05), Lannion (França), Novembro, 2005.

[18] ENRÍQUEZ, J. “D1.1.3: Business model and system design specification”. Agosto, 2005-1. Disponível em http://www.euqos.eu/documents/md_670_d113_v2_pdf.zip; Capturado em agosto de 2006.

[19] ENRÍQUEZ, J., “Annex to D1.1.3: Business model and system design specification”. Agosto, 2005-2. Disponível em http://www.euqos.eu/documents/md_819_d113-annex_v2_pdf.zip; Capturado em Agosto de 2006.

[20] FAJARDO, V. et al. “Diameter Application Design Guidelines: draft-ietf-dime-app-design-guide-02.txt”. Internet Engineering Task Force, Julho, 2007. Trabalho em andamento.

[21] FARREL, S. et al. “AAA Authorization Requirements”. RFC 2906, Internet Engineering Task Force, Agosto, 2000.

[22] GROSSMAN, D. “New Terminology and Clarifications for DiffServ”, RFC 3260, Internet Engineering Task Force, Abril, 2002.

[23] GRUBER, T. “A Translation Approach to Portable Ontology Specifications”, Knowledge Acquisition, 5(2), 199-220, 1993.

[24] HANCOCK, R. et al, 2005. “Next Steps in Signaling: Framework”. Request For Comments RFC 4080, Internet Engineering Task Force. Junho, 2005.

[25] HASAN, et al. “The Design of an Extended AAAC Architecture”. IST Mobile & Wireless Telecommunications Summit, Junho, 2002.

[26] HEWLETT-PACKARD. “HP-UX AAA Server A.06.00: Getting Started Guide”. Hewlett-Packard Company, 2003. Disponível em http://www.docs.hp.com/en/T1428-90026/T1428-90026.pdf, acessado em Agosto, 2007.

[27] HWANG, J. et al. “Enabling dynamic market-managed QoS interconnection in the next generation internet by a modified BGP mechanism”, IEEE International Conference on Communications, Agosto, 2002.

[28] JENA. “A Semantic Web Framework for Java”. Disponível em: http://jena.sourceforge.net. Acessado em Julho, 2008.

[29] KIM, B., CHAE, H., HAN, T. e JEONG, Y. “Bandwidth Broker Signaling for Service Level Negotiation over Heterogeneous IPv4/IPv6 DiffServ Networks”. Em: 6th International Conference on Advanced Communication Technology. Volume: 2, Págs. 1097 a 1102, Outubro, 2004.

[30] KORHONEN, J., et al. “Quality of Service Parameters for Usage with the AAA Framework”, Internet Engineering Task Force, Setembro, 2007-1. Trabalho em andamento.

[31] KORHONEN, J., et al. “Diameter Proxy Mobile IPv6: Support For Mobility Access Gateway and Local Mobility Anchor to Diameter Server Interaction”, Internet Engineering Task Force, Novembro, 2007-2. Trabalho em andamento.

[32] KORHONEN, J., et al. “Quality of Service Attributes for Diameter”, Internet Engineering Task Force, Janeiro, 2008. Trabalho em andamento.

[33] KURTANSKY, P., et al. “Extensions of AAA for Future IP Networks”. Em: IEEE WCNC 2004 - Wireless Communications and Networking Conference, Março, 2004.

[34] MASIP-BRUIN, X. et al. “The EuQoS System: A Solution for QoS Routing in Heterogeneous Networks”. IEEE Communications Magazine, Fevereiro, 2007.

[35] MYKONIATI, E. et al. “Admission Control for Providing QoS in DiffServ IP Networks: The TEQUILA Approach”. IEEE Communications Magazine, Janeiro, 2003.

[36] NICHOLS, K., Jacobson, V. e Zhang, L. “A two-bit differentiated services architecture for the Internet”, RFC 2638, Internet Engineering Task Force, Julho, 1999.

[37] PAPAIOANNOU, T., Stamoulis, G. “Design of a Charging and Accounting for QoS-differentiated VPN Services to Mobile Users”. Computer Communications Journal, Março, 2004.

[38] RENSING, C., et al. “AAA: A Survey and a Policy-Based Architecture and Framework”, IEEE Network, Novembro/Dezembro, 2002.

[39] RIGNEY, C., et al. “Remote Authentication Dial In User Service (RADIUS)”, RFC 2865, Internet Engineering Task Force, Junho, 2000-1.

[40] RIGNEY, C. “RADIUS Accounting”, RFC 2866, Internet Engineering Task Force, Junho, 2000-2.

[41] ROBERTS, J. “Internet Traffic, QoS and Pricing”. Proceedings of IEEE, Setembro, 2004.

[42] SUN, D. et al. “Diameter Quality of Service Application”, Internet Engineering Task Force, Janeiro, 2008. Trabalho em andamento.

[43] TERZIS, A. et al. “A two-tier resource management model for the Internet”, IEEE Global Internet, Dezembro, 1999.

[44] VOLLBRECHT, J. et al. “AAA Authorization Framework”. RFC 2904, Internet Engineering Task Force, Agosto, 2000-1.

[45] VOLLBRECHT, J. et al. “AAA Authorization Application Examples”. RFC 2905, Internet Engineering Task Force, Agosto, 2000-2.

[46] ZHANG, Z. et al. “Decoupling QoS control from core routers: A novel bandwidth broker architecture for scalable support of guaranteed services”. On the proceedings of ACM SIGCOMM, Sweden, Agosto, 2000.

[47] ZHANG, Z., DUAN, Z. e HOU, T. “On Scalable Design of Bandwidth Brokers”, IEICE Trans. Commun, Agosto, 2001.

[48] W3C. “Resource Description Framework (RDF)”, W3C Recommendation. Disponível em http://www.w3.org/RDF/. 2004. Acessado em Julho, 2008.

[49] W3C. “SPARQL Query Language for RDF”, W3C Recommendation. Disponível em: http://www.w3.org/TR/rdf-sparql-query/. Janeiro, 2008. Acessado em Julho, 2008.

[50] WESTERINEN, A. et al. “Terminology for Policy-Based Management”, RFC 3198, Internet Engineering Task Force, Novembro, 2001.

ANEXO A

O Projeto EuQoS

A arquitetura EuQoS (*End to end Quality of Service over heterogeneous networks*) [Masip-Bruin 2007] oferece um suporte de QoS fim-a-fim sobre redes heterogêneas, considerando múltiplos sistemas autônomos (AS) e em larga escala. É o resultado do projeto EuQoS, suportado pela União Européia, reunindo os resultados das pesquisas realizadas pelos projetos europeus Áquila, Tequila e Cadenus, todos na área de QoS.

O projeto EuQoS reúne vários centros de pesquisa, universidades, fabricantes e operadoras de telecomunicação trabalhando nos pontos descritos acima. O principal objetivo deste projeto é definir e implementar um modelo arquitetural de rede, o Sistema EuQoS, capaz de garantir QoS fim-a-fim em redes heterogêneas, entre múltiplos ASs, em larga escala.

A arquitetura EuQoS foi definida de acordo com as seguintes regras [Masip-Bruin 2007]: (i) as aplicações dos usuários devem ser capazes de negociar o conteúdo e a qualidade de cada comunicação; (ii) os administradores de rede devem ter a liberdade para usar qualquer das tecnologias de rede e implementar o EuQoS sobre elas; e (iii) os mecanismos propostos devem ser incrementais, e coexistir com a base já instalada, o que permite a continuidade dos serviços já contratados e aplicações em uso.

Para suportar a negociação de QoS a cada comunicação, atendendo a regra (i), o sistema EuQoS é orientado a sessões, oferecendo QoS somente às aplicações que

necessitam, e quando elas necessitam. Assim, antes de iniciar uma transmissão é necessário que a aplicação (possivelmente em resposta a uma solicitação do usuário) inicie o processo de estabelecimento de uma sessão com os requisitos de QoS desejados. Durante o processo de negociação desta sessão com QoS, as aplicações origem e destino definem parâmetros de QoS necessários e, caso haja recurso disponível, a rede realiza a reserva destes recursos necessários para garantir os parâmetros de QoS estabelecidos. Este processo de estabelecimento de sessões é detalhado na seção 4 deste anexo.

Como apresentado na Fig. 23, a arquitetura EuQoS é dividida em três planos: plano de serviço, ou camada de aplicação; plano de controle ou camada de rede virtual ; e plano de transferência. O plano de controle é ainda subdividido em uma camada independente de tecnologia e uma camada dependente de tecnologia.

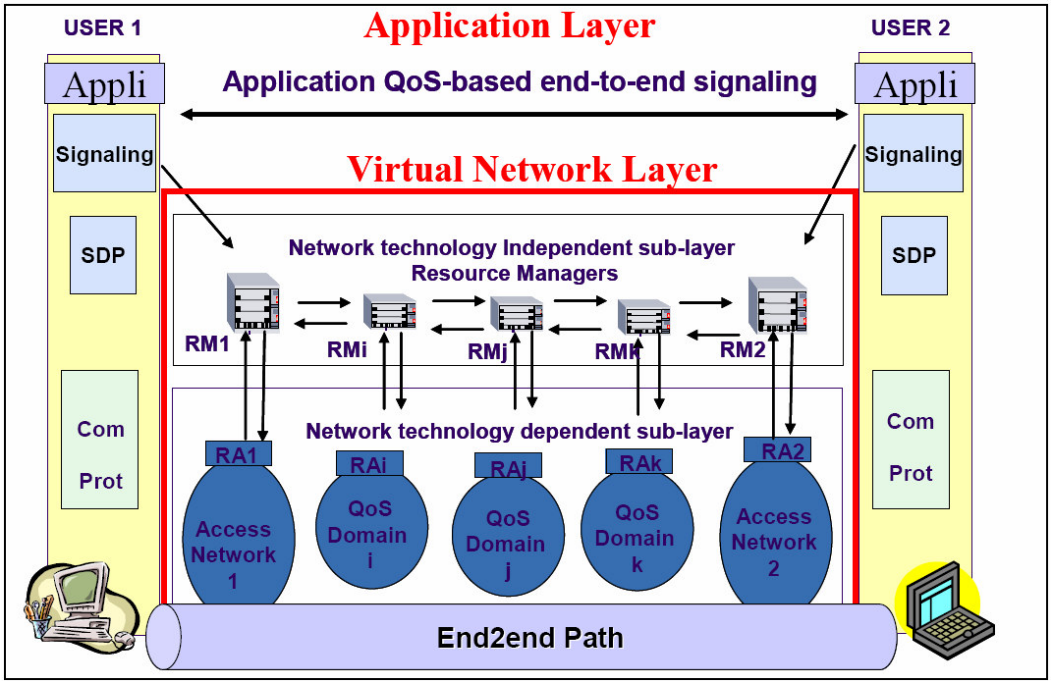


FIGURA 23: Arquitetura de rede fim-a-fim do EuQoS [Enríquez 2005-2]

1 Camada de Aplicação

O sistema EuQoS suporta tanto aplicações prontas para o EuQoS (*EuQoS aware*), que já foram projetadas para suportar a abertura de sessões EuQoS, quanto outras aplicações. Estas últimas podem fazer uso das funcionalidades do sistema EuQoS via um módulo de adaptação.

A Fig. 24 apresenta os componentes da camada de aplicação do EuQoS. Para o suporte a essas aplicações, são implementados no terminal do usuário os módulos de controle de QoS (QCM - *QoS Control Module*), e de sinalização de Aplicações (ASIG - *Application Signaling*). Já no servidor EuQoS de cada AS, há um módulo responsável pela interface com os terminais de usuário para o estabelecimento das sessões (A-SSN - *Application-level Signaling and Service Negotiation*).

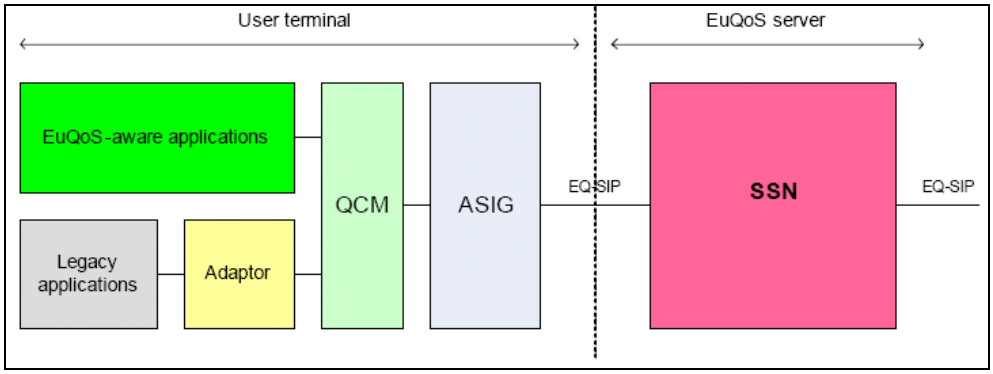


FIGURA 24. Camada de Aplicação da Arquitetura EuQoS [Enríquez 2005-2]

2 Camada de Rede Virtual

A camada de rede virtual é dividida em uma camada independente de tecnologia (TI), controlada por um gerenciador de recursos (RM - *Resource Manager*) para cada AS, e

uma camada dependente de tecnologia (TD), controlada por um alocador de recursos (RA - *Resource Allocator*), também para cada AS.

Para gerenciar os recursos de um AS, o RM deve implementar uma série de funcionalidades, dentre as principais estão a tomada de decisões sobre o controle de admissão de novas conexões (CAC) para o domínio local, o gerenciamento dos contratos de QoS com os ASs vizinhos, o controle do roteamento com QoS inter-domínio, e o encaminhamento de solicitações de QoS para os RMs vizinhos. Sempre que necessário, as decisões do RM são aplicadas na forma de configurações de dispositivos específicos por meio do RA, que está localizado na camada TD. O controle sobre as disponibilidades de recursos da rede, para tomar as decisões é baseado nos recursos nominais da rede, nos contratos assinados e nas conexões aceitas.

O RA é voltado para uma tecnologia específica, e faz as reservas e liberações de recursos (tais como banda e buffer) nos roteadores com base nas solicitações do RM. Também possui um controle de admissões, com base nos recursos efetivamente disponíveis nos equipamentos dessa tecnologia específica. Pode haver um ou mais RAs por AS, dependendo das tecnologias utilizadas na rede do AS.

A principal vantagem da existência dessas duas subcamadas é a dissociação entre as decisões da rede e as tecnologias de rede.

3 Principais módulos

Segue uma descrição dos principais módulos da arquitetura EuQoS, agrupados de

acordo com a camada em que estão localizados, conforme apresentado na Fig. 25.

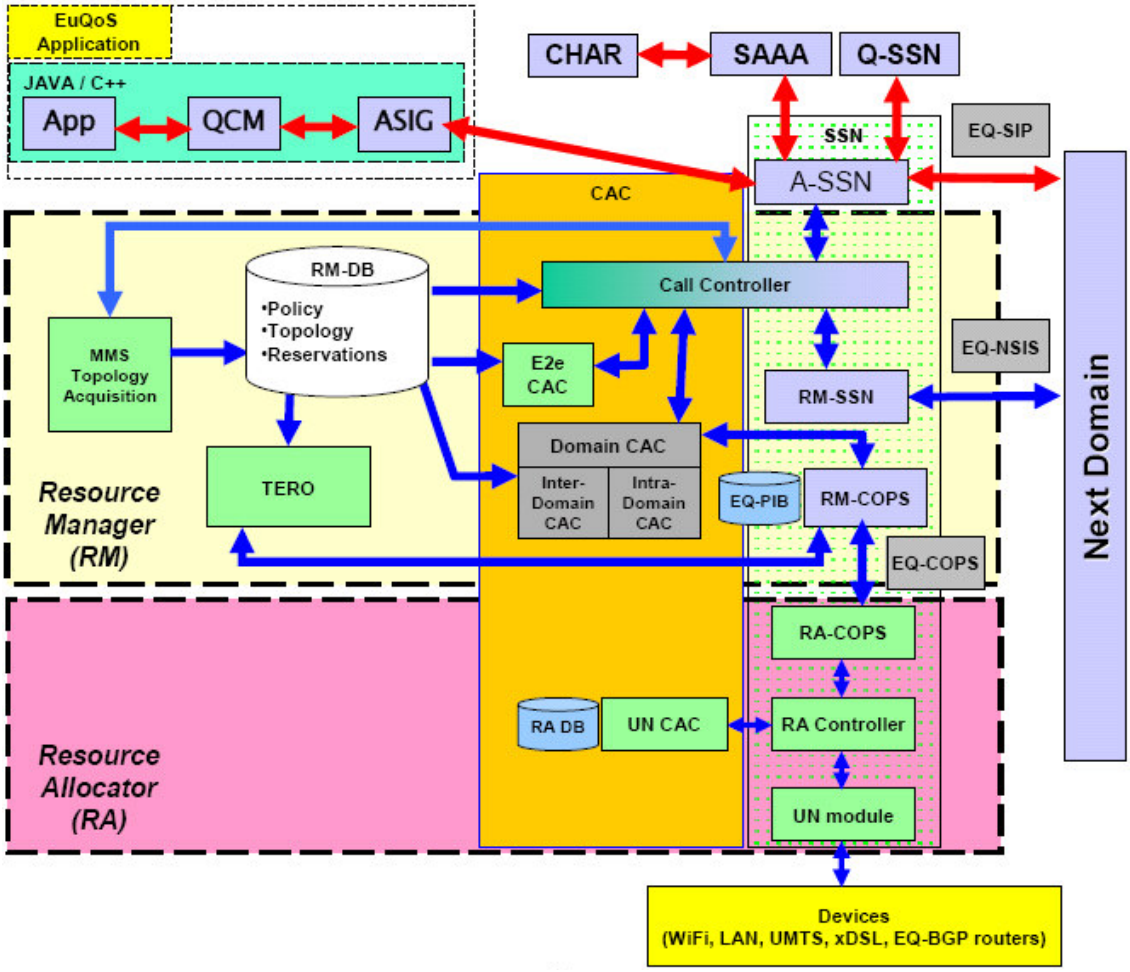


FIGURA 25. Principais módulos do EuQoS dentro de um AS [Masip-Bruim 2007].

Camada de Aplicação

A Camada de aplicação possui diversos módulos localizados nos terminais do usuário e nos AS no lado do NSP. No lado do terminal do usuário, estão localizados o **QCM (*Quality Control Module*)** e o **ASIG (*Application Signaling*)**. E no lado do NSP estão o **CHAR (*Charging*)**, o **SAAA (*Security, Authentication, Authorization and***

***Accounting*)**, e o **A-SSN (*Application-level SSN*)**. A seguir será dada um visão geral destes módulos.

O Módulo de Controle de Qualidade **(QCM - *Quality Control Module*)** é instalado no terminal do usuário, e tem por objetivo fornecer uma interface comum entre as aplicações e o sistema EuQoS. Ele oferece meios para as aplicações gerenciarem sessões com QoS, acessar informações da conta do usuário e tratar eventos que chegam do EuQoS. O QCM faz o mapeamento entre os requisitos de QoS expressos pela aplicação (por exemplo, codec utilizado) e os parâmetros de QoS padronizados pelo EuQoS (vazão, atraso, variação de atraso e taxa de perda de pacotes). O QCM é composto por dois sub-módulos:

- BQCM: é obrigatório, e fornece os serviços de gerenciamento de sessão (registro de usuário, estabelecimento, alteração e finalização de sessões com QoS e tratamento de eventos vindos do EuQoS);
- XQCM: é opcional, e provê outros serviços, como o gerenciamento do perfil do usuário, mostrar dados da sessão do usuário, bem como informações de cobrança.

O módulo **ASIG (*Application Signaling*)** também é localizado no terminal do usuário, provê uma interface que permite às aplicações finais (via o módulo QCM) estabelecer, gerenciar e terminar sessões com QoS, negociando características de sessão e de QoS com o sistema EuQoS e com as aplicações remotas. Para tal, este módulo implementa as funções de um agente de usuário SIP (*Session Initiation Protocol*), estendido para suportar o protocolo EQ-SIP. EQ-SIP é um protocolo de sinalização de aplicações, baseado no SIP, expandido para suportar negociação de características de QoS. As

principais funções do ASIG são:

- Realizar os procedimentos de autenticação do usuário no sistema EuQoS, através do A-SSN;
- Realizar os procedimentos de estabelecimento, alteração e finalização de sessões com QoS usando o protocolo EQ-SIP para negociar com o A-SSN os parâmetros de QoS solicitados pelo QCM. É através do A-SSN, que o ASIG local negocia o estabelecimento de sessões com QoS com o ASIG remoto.
- Sinalizar ao QCM a chegada de mensagens de notificação de eventos do EuQoS, via JMS (por exemplo, um usuário remoto solicitando uma sessão com o usuário local).

O módulo **A-SSN (*Application-level SSN*)** suporta a sinalização fim-a-fim no nível de aplicação, para estabelecer, alterar e finalizar sessões de usuário com QoS. Ele implementa um servidor proxy baseado no protocolo EQ-SIP. No estabelecimento de sessão, após verificar junto ao SAAA se o usuário tem direito de estabelecer a sessão com a CoS solicitada, o A-SSN extrai as informações de QoS do campo SDP da mensagem EQ-SIP e repassa ao RM local, para a efetuar a reserva de recursos da rede. O A-SSN tem interações com o ASIG, com o SAAA, com o RM local e com o A-SSN remoto (onde está registrado o usuário destino da sessão). Para se comunicar com o SAAA, o SSN deve ser acoplado a um ***SAAA Client***, que permite a troca de mensagens com o servidor para autenticação, autorização e contabilidade.

O **SAAA (*Security, Authentication, Authorization and Accounting*)** é o módulo responsável pelo gerenciamento do acesso dos usuários ao sistema EuQoS (verificando suas

credenciais – Autenticação), por permitir que um usuário utilize certos serviços e níveis de QoS (verificando os direitos de uso – Autorização), e pela coleta de dados de consumo (verificando os recursos utilizados – Contabilidade), enviando essas informações para o sistema de cobrança (módulo CHAR). O SAAA é composto pelos seguintes módulos: ***SAAA Server***, que realiza as tarefas de autenticação, autorização e contabilidade; ***SAAA Client***, que é acoplado ao *SSN*, e faz a interface entre este e o SAAA Server. A comunicação entre o *SAAA Client* e o *SAAA Server* é feita via o protocolo *DIAMETER* [Calhoun 2003]; **Gerenciador do repositório de usuários**, que mantém as informações dos usuários; e o **Sistema de Contabilidade**, que coleta informações de uso, armazena-as em uma base de dados, formata os dados IPDR (*IP Detail Record*) e os envia para o módulo CHAR usando JMS.

É no módulo **CHAR (*Charging*)** que são cadastrados os assinantes do EuQoS, juntamente com as aplicações e os perfis de QoS contratados. É o módulo responsável pelo faturamento, com base nas informações de uso coletadas pelo SAAA e nas políticas de faturamento definidas pelo NSP. Ele também informa ao SAAA as credenciais e os perfis dos usuários autorizados a usar o sistema. O CHAR é organizado nos seguintes sub-módulos:

- ***Charging Server***: O software servidor, que oferece uma interface web para inclusão, alteração e exclusão de assinantes e verificação de status de conta. Além disso, ele oferece uma interface para o SAAA permitindo o recebimento de notificações de eventos do SAAA (*Charging Daemon*).

- ***Charging DB Server***: O servidor de base de dados do faturamento, que contém as seguintes bases de dados: *Charging Profiles DB,* que mantém os perfis de usuário, que especificam os diferentes níveis de serviços oferecidos (ouro, prata, bronze), e contém também as políticas de faturamento para cada perfil; e a *Charging Customers DB* que mantém a relação de usuários assinantes, os serviços contratados e seus registros de consumo.

Camada de Rede Virtual – Independente de Tecnologia

Envolve diversos módulos componentes do gerenciador de recursos (RM - *Resource Manager*):

- ***Call Controller*:** é o módulo central do controle de admissão de conexões (CAC), recebe as requisições do A-SSN e as gerencia, dividindo e repassando para os módulos RM-SSN, E2e CAC, *Domain* CAC, MMS. Para a sua operação, ele acessa a base de dados RM-DB.
- **RM-DB (*Resource Manager Data Base*):** é a base de dados do RM, que inclui informações sobre topologia da rede, políticas de gerenciamento de recursos, capacidade de recursos da rede, os contratos de SLS inter-domínios firmados e o status de reserva de recursos da rede e dos enlaces inter-domínios.
- **RM-SSN (*RM Signalling and Service Negotiation*)**: gerencia a sinalização entre os RMs de cada domínio, para reserva de recursos ao longo do EQ-path, desde o AS origem até o AS destino, durante o estabelecimento de cada sessão. Para isso, ele

utiliza o protocolo EQ-NSIS, uma extensão do protocolo NSIS (*Next Step in Signalling*) [Hancock 2005] para suportar QoS.

- **E2E CAC (*End to End CAC*):** durante o estabelecimento de uma sessão com QoS, este módulo verifica se existe na RM-DB um EQ-path para a Classe de Serviço (CoS) solicitada até o destino. Um EQ-path é um caminho fim-a-fim com suporte a uma determinada CoS, e é abordado na sessão 3.3.4. Caso existam EQ-paths até o destino, é retornado ao *Call Controler* o EQ-path com as melhores características de QoS.
- ***Domain* CAC**: verifica se há recursos para a sessão no domínio do AS, e nos enlaces com os ASs vizinhos, com base nas informações de topologia da rede do AS e nas reservas de recursos armazenadas na RM-DB. Ele é composto por um ***Inter-Domain* CAC,** que verifica se os enlaces com os ASs vizinhos no EQ-path suportam a nova requisição de QoS; e um ***Intra-Domain* CAC**, que verifica se a rede interna do AS suporta a nova requisição. Em caso de aceitação da nova requisição, a reserva de recursos é registrada na RM-DB e é solicitada ao RM-COPS a alocação dos recursos nos roteadores.
- **RM-COPS (*RM Common Open Policy Service*)**: responsável pela comunicação com os RA-COPS (localizados nos *Resource Alocators*), através do protocolo EQ-COPS, uma extensão do protocolo COPS-PR para suportar QoS. Pode haver mais de um RA por AS, dependendo das tecnologias usadas na rede do AS. Este módulo é acionado pelo *Domain* CAC para a alocação de recursos dentro da rede do AS, e nos enlaces inter-domínios nos roteadores de borda, quando uma nova sessão com

QoS é aceita. Também é acionado pelo TERO, durante o estabelecimento dos EQ-paths, para aplicação de comandos aos roteadores de borda.

- **EQ-PIB (*E2E QoS Policy Information Base*)**: é uma base de dados com as políticas independentes de tecnologia enviadas pelo RM para o RA. Essas políticas especificam as regras para a reserva de recursos para os fluxos das sessões aceitas. É gerenciada pelo RM-COPS.
- **MMS (*Monitoring and Measurement System*)**: faz as monitorações sobre a utilização real dos recursos da rede, e sobre o atendimento aos SLSs contratados, informando ao *call controller* em caso de desrespeito a algum SLS contratado. Ele também coleta informações sobre a topologia da rede, atualizando a RM-DB.
- **TERO (*Traffic Engeneering and Resource Optimization*):** é o responsável pela configuração das rotas inter-domínio com QoS e pela reserva de recursos ao longo do caminho. Ele recebe as informações de roteamento dos ASs vizinhos, com as respectivas informações de QoS, e computa os parâmetros totais de QoS do início ao fim do caminho. Acessa na RM-DB as informações dos contratos inter-domínios firmados para cada CoS, os links inter-domínios, o tráfego dentro do AS e os valores de parâmetros de QoS para o trânsito dos pacotes dentro do AS. Se comunica com os roteadores de borda para aplicar as regras de reserva de recursos para cada CoS e propagar informações de rotas com QoS, usando o RM-COPS para abstrair da tecnologia dos roteadores.

Camada de Rede Virtual – Dependente de Tecnologia

Envolve os módulos que compõem o RA:

- **RA-COPS:** responsável pelo encapsulamento e tratamento das mensagens com o RM usando o protocolo EQ-COPS.
- **RA *Controller*:** gerencia as funções de controle de alocação de recursos
- **UN CAC:** faz o controle de admissão de conexões para uma tecnologia específica, baseado nos recursos realmente alocados.
- **UN *Module*:** aplica os comandos de alocação e liberação de recursos nos dispositivos da rede, utilizando o protocolo COPS.

4. Principais processos do EuQoS

Para garantir QoS fim-a-fim, é necessário garantir certos parâmetros de desempenho em todos os saltos desde a origem até o destino. Isso é alcançado usando o roteamento com QoS (QoSR - *QoS Routing*). Para isso, no EuQoS é usado o conceito chamado de EQ-path (*End-to-end QoS path*), representado na parte inferior da Fig. 6. Além de QoSR, também é necessário um controle de admissão de conexões (CAC) para evitar a sobrecarga da rede, que no caso do EuQoS é realizado durante o estabelecimento das sessões com QoS. O processo de sinalização e negociação de serviços (SSN) coordena o estabelecimento das sessões com QoS, incluindo as negociações entre aplicações origem e destino, a verificação de credenciais e direitos de usuário e a solicitação de abertura de conexão com QoS junto à camada virtual de rede do EuQoS.

Essa sessão tem por objetivo apresentar esses quatro elementos do EuQoS: QoSR, SSN, CAC e o modelo de usuários do EuQoS, destacando algumas das limitações impostas por esse modelo.

Roteamento com QoS (QoSR) e EQ-paths

Um EQ-path (*End-to-end QoS path*) é um caminho fim-a-fim configurado para uma determinada CoS (Classe de Serviço). O conceito de EQ-path foi criado de maneira a conseguir a escalabilidade do DiffServ e a garantia oferecida pelo IntServ. Esta escalabilidade é obtida pela definição de uma pequena quantidade de CoS a serem controladas, e a garantia é alcançada com o controle da alocação dos recursos dessa classe ao longo do EQ-path, não permitindo a aceitação de novos fluxos que venham a comprometer a qualidade dos fluxos já aceitos.

Há três processos envolvendo os EQ-paths: QoSR, que elabora a **construção do EQ-path**, computado pelo TERO em retaguarda, em uma fase anterior ao estabelecimento da sessão de troca de dados, a **Invocação de tráfego**, onde é feito o controle de admissão para evitar a sobrecarga, realizada durante o estabelecimento da sessão, e o **Monitoramento**, que ocorre durante todo o andamento da sessão, e aciona mecanismos de reparo em caso de falhas.

Os EQ-path são fim-a-fim, cruzando vários ASs. A arquitetura EuQoS é baseada na construção de EQ-paths segmentados por AS. Assim, cada AS intermediário pode agregar vários EQ-paths vindos dos ASs vizinhos em um único EQ-path, oferecendo a escalabilidade no núcleo da rede. Esse modelo também permite maior flexibilidade de

administração para cada AS, que pode definir caminhos internos entre um roteador de borda e outro independentemente. Outra conseqüência da adoção desse modelo é que o CAC precisa ser administrado em cada AS, e as reservas de uso tem que ser solicitadas aos RMs de todos os ASs da origem ao destino.

Há três elementos envolvidos no estabelecimento dos EQ-paths [Masip-Bruim 2007]:

- **p-SLS** são contratos entre dois ASs vizinhos que regulam o tráfego em um enlace inter-domínio, para uma determinada CoS, em um sentido. Pacotes de uma CoS podem atravessar um enlace inter-domínio somente se existir um p-SLS para essa classe. Assim o QoSR inter-domínio é limitado aos p-SLS, cujos recursos são controlados pelo módulo TERO. Os p-SLS dependem de condições comerciais e são negociados manualmente.
- **EQ-BGP** é uma extensão do protocolo BGP-4 para suportar a seleção e a divulgação de caminhos com QoS para diferentes CoS [Beben 2006]. Além das informações de alcançabilidade, o protocolo também informa as características de QoS do caminho. Um roteador de borda só propaga as informações de QoS de uma CoS através de um enlace inter-domínio se houver um p-SLS para essa CoS sobre esse enlace. É com base nessas informações de QoS que o TERO computa os EQ-paths.
- **TERO** é o módulo que faz o cômputo final dos valores de QoS ao longo do percurso, durante o processo de estabelecimento do EQ-path, calculando os limites máximos de carga que a rede pode suportar para a CoS em questão ao longo de

todos os ASs e dos enlaces inter-domínio. Outros parâmetros de QoS, como atraso total, variação de atraso e taxa de erros também são computados. O TERO é executado em retaguarda, em uma escala de tempo da ordem de horas ou dias, sendo, portanto maior do que a escala de tempo de estabelecimento de sessões.

Há duas opções possíveis previstas para o estabelecimento de EQ-paths:

- **Modo relaxado**: onde o QoSR estabelece o caminho a ser seguido da origem até o destino, com suporte a QoS para a CoS em questão, mas não aloca os recursos nessa fase. Os recursos terão que ser reservados dinamicamente no momento do estabelecimento da sessão, e associados ao EQ-path. Para isso são necessários alguns pré-requisitos: o caminho fim-a-fim deve ser conhecido; cada AS no caminho deve ter recursos previstos para a CoS em questão em "seu trecho do caminho" (reserva dentro da rede do AS); devem existir contratos p-SLS pré-estabelecidos entre os AS, reservando recursos para a CoS, nos enlaces entre os AS consecutivos, ao longo de todo o percurso (reserva nos enlaces entre os ASs). Não há como saber a priori se há recursos disponíveis para a CoS solicitada ao longo de todo o percurso, razão pela qual o RM de cada AS terá que implementar o CAC para evitar a sobrecarga e conseqüente degradação das conexões aceitas.
- **Modo rígido**: Nesse modo o provisionamento de recursos é feito antecipadamente, usando engenharia de tráfego (túneis MPLS), com configuração estática. Esse modo oferece maiores garantias, mas apresenta um

custo mais elevado, uma configuração mais trabalhosa, apresenta limitações para ser implementado em larga escala e deve servir a aplicações de alto valor agregado, tais como aplicativos de missão crítica de empresas geograficamente distribuídas, com uso intensivo. Está previsto para ser implementado apenas na segunda fase do projeto EuQoS.

Estabelecimento de sessão no EuQoS

A Fig. 26 ilustra a interação entre os módulos envolvidos no estabelecimento de uma sessão, no nível da aplicação (setas azuis), a invocação do CAC (seta vermelha), uma visão geral sobre o CAC (setas verdes, descrito em maiores detalhes na próxima sessão) e a solicitação da alocação de recursos (seta preta).

Quando o usuário realiza o seu registro no sistema EuQoS, o QCM passa as credenciais do usuário ao ASIG (1), para que este efetue o registro no sistema. O ASIG monta uma mensagem EQ-SIP e envia ao A-SSN (2). O A-SSN verifica as credenciais do usuário junto ao SAAA - Autenticação (3), elabora um desafio baseado em um string fornecido pelo SAAA, a ser respondido em um prazo informado, e submete o desafio ao ASIG (4). Sendo este respondido corretamente o A-SSN então retorna ao ASIG o perfil de QoS que o usuário assinou (5), que é então repassado ao QCM (6) e na seqüência para a aplicação (7). Esse registro é válido durante um período especificado pelo SAAA. Ao final desse período, o registro precisa ser renovado [Enríquez 2005-1].

Antes de submeter um tráfego com QoS é necessário estabelecer uma sessão

solicitando a CoS desejada. Quando uma aplicação precisa estabelecer uma sessão com QoS, ela submete seus requisitos de QoS ao QCM (8), que os converte para uma forma padronizada do EuQoS e repassa o perfil da sessão solicitada ao ASIG (9), que o repassa através de mensagens EQ-SIP ao A-SSN (10). O A-SSN recebe o perfil de sessão e verifica junto ao SAAA (11) se o usuário está autorizado a fazer a requisição de QoS especificada para esse tipo de aplicação (Autorização). Em caso afirmativo, o A-SSN deve negociar o perfil da sessão (por exemplo, codecs possíveis e seus parâmetros de QoS) com o usuário destino, também usando EQ-SIP (12), e depois verificar se a rede tem condições de suportar a sessão requisitada com o usuário destino. Para isso solicita ao RM (13) a verificação da disponibilidade e conseqüente alocação dos recursos, realizada pelo CAC. Essa solicitação contém as especificações de QoS solicitadas para a sessão (chamada no projeto EuQoS de SLS), e é enviada pela camada de aplicação (A-SSN) para o RM do primeiro AS. Esse primeiro RM é o responsável por verificar se existe um EQ-path para a CoS especificada. Uma vez verificada a existência do EQ-path (14), o RM precisa verificar em todo o EQ-path se há recursos disponíveis para atender à requisição, solicitando recursivamente aos outros RMs que fazem parte do percurso [Enríquez 2005-1].

Cada RM no EQ-path verifica se há disponibilidade de recursos no seu trecho do EQ-path para atender à requisição (15). Em caso afirmativo, as regras de QoS são aplicadas aos roteadores desse trecho do EQ-path, através do RA (16, 17), e a requisição é passada ao próximo RM do EQ-path (18). Quando o último RM finaliza a alocação dos recursos, é devolvido um ACK pelo mesmo caminho. Caso algum RM no caminho não disponha de recursos para atender à solicitação, é devolvido um NACK, e todos os recursos já alocados

no caminho, por todos os RMs são liberados.

Em caso de resposta positiva (o CAC aceitou a conexão), o RM retorna a confirmação positiva para o A-SSN (19), que por sua vez confirma o estabelecimento da sessão com o ASIG do terminal do usuário (20). O ASIG devolve a resposta ao QCM (21), que repassa à aplicação (22), e a transmissão de dados pode iniciar.

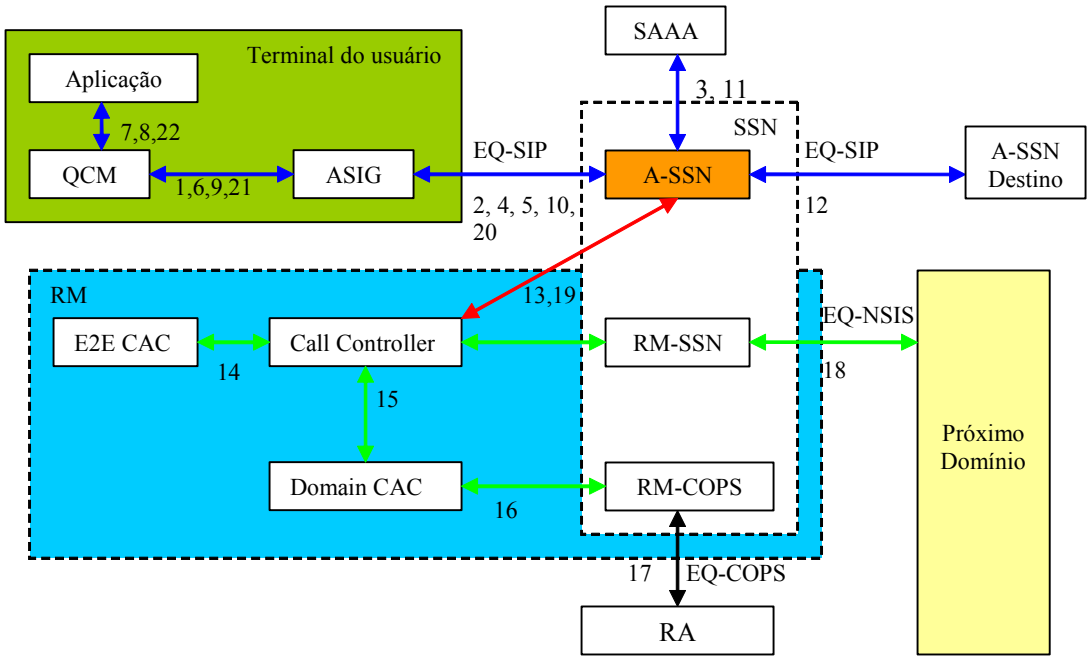


FIGURA 26: Módulos envolvidos no estabelecimento de uma conexão

Processo de Controle de Admissão de Conexões (CAC)

O CAC é a função do RM responsável por verificar se existem recursos suficientes para atender a uma nova requisição de QoS. Essa nova requisição especifica os parâmetros de desempenho necessários para atender à aplicação.

Para controlar os parâmetros de QoS em diversos contextos, alguns padrões de SLSs

foram definidos [Burakowski 2006]:

- **a-SLS**: Contém os requisitos de QoS passados pelo A-SSN para o RM;
- **e-SLS**: Contém os requisitos de QoS com as informações de origem e destino;
- **i-SLS**: Contém os parâmetros de QoS do enlace inter-domínio;
- **d-SLS**: Contém os parâmetros de QoS do domínio do AS (inclui o enlace para o próximo domínio);
- **r-SLS**: Contém os parâmetros de QoS passados ao próximo RM.

O processo de CAC no EuQoS pode ser observado com maiores detalhes na Fig. 27, e ocorre da seguinte forma [Enríquez 2005-1]:

1. O *Call Controller* do primeiro RM recebe do A-SSN duas informações: o a-SLS, que contém os requisitos de desempenho (vazão, atraso, variação de atraso e taxa de erros), e a identificação do fluxo (ID, IP de origem e de destino do fluxo, protocolo de transporte, CoS, etc), analisa os parâmetros e compõe o e-SLS, para invocar o *End-to-end CAC*;
2. O *End-to-end CAC* verifica se há EQ-paths disponíveis para a solicitação, escolhe o melhor e o envia ao *Call Controller*. O EQ-path contém os parâmetros de QoS suportados por todos os AS no caminho, incluindo o endereço de todos os RMs;
3. O *Call Controller* computa os endereços de ingresso e egresso do domínio atual, com base no EQ-path informado, compõe o d-SLS e solicita sua validação ao Domain CAC;

4. O *Domain CAC* faz a verificação na rede do AS (*IntraDomain CAC*) e no enlace com o próximo AS (*InterDomain CAC*), e solicita (através do RA-SSN) aos UN CACs relevantes do seu AS a reserva dos recursos nos dispositivos de rede;
5. Os UN CACs confirmam a reserva ao *Domain CAC*;
6. O *Domain CAC* confirma ao *Call Controller* a admissão no domínio corrente;
7. O *Call Controler* então analisa novamente os parâmetros de QoS e compõe o r-SLS, para ser passado ao próximo RM no EQ-path, descontando o atraso, variação de atraso e taxa de erros impostos pelo domínio local, e o envia através do módulo RM-SSN;
8. O próximo RM retira seu endereço do EQ-path, processa o r-SLS para o seu domínio, desconta os parâmetros de QoS relativos ao seu AS, recompõe outro r-SLS e o repassa para o próximo RM do caminho, até o final do EQ-path;
9. O último RM inicia a devolução das confirmações que efetivam as alocações de recursos em todos os RM-DBs, até o RM de origem;
10. O *Call Controler* do AS de origem confirma a admissão da conexão ao A-SSN;

Caso uma resposta de um domínio seja negativa (NACK), o NACK é retornado e todos os recursos alocados em todos os ASs são liberados.

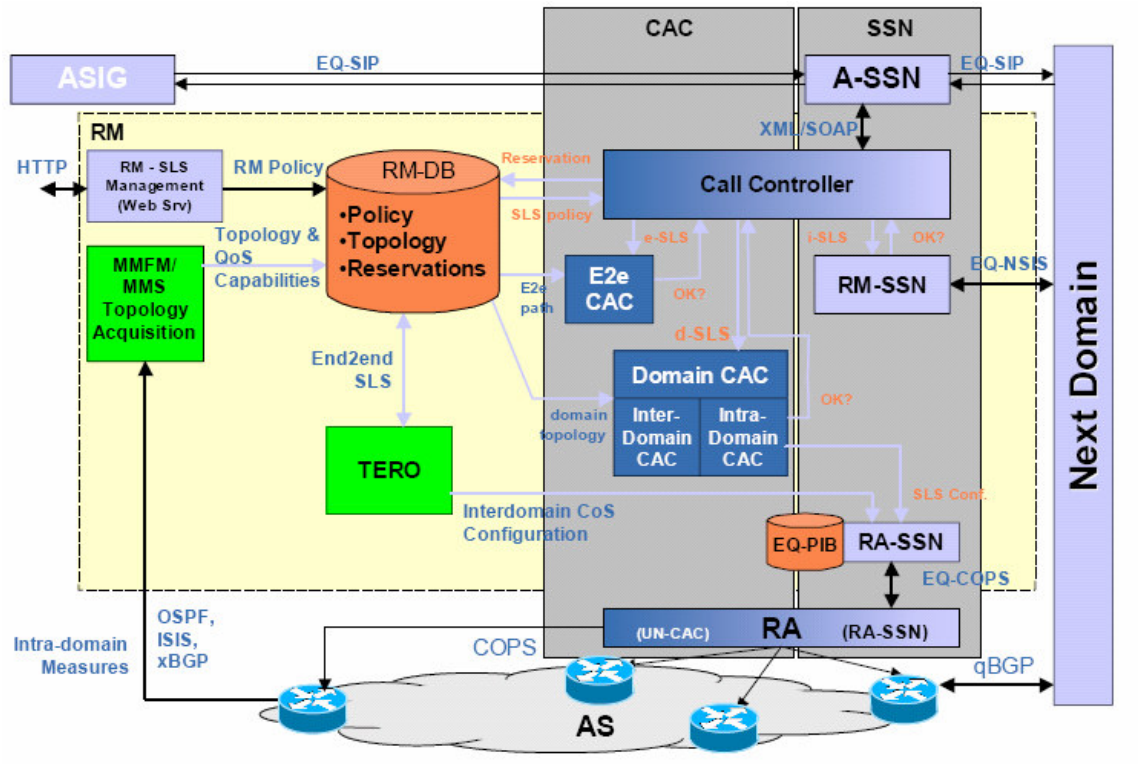


FIGURA 27: Arquitetura do Gerenciador de Recursos (RM) [Enríquez 2005-1]

O modelo de usuário do EuQoS

Uma das motivações ao desenvolvimento do EuQoS é a possibilidade de exploração de novos modelos de negócio que os serviços com QoS fim a fim suportam. Isso obviamente inclui a cobrança diferenciada pelo uso desses serviços. Assim, o sistema precisa identificar os usuários dos serviços e os seus consumos, para fins de faturamento.

Os procedimentos de inclusão, alteração e exclusão dos usuários do EuQoS são feitos através de uma interface WEB do módulo CHAR, responsável pelo faturamento. Na primeira versão do EuQoS, quem faz essa manutenção é um usuário do NSP com perfil de administrador, que inclui novos usuários com base em um contrato negociado manualmente com o cliente. Esses novos usuários são associados a perfis definidos pelo NSP,

contemplando também os tipos de aplicações e o nível de QoS contratados.

Concluído o cadastramento de um novo usuário, o CHAR informa ao SAAA os dados do novo usuário, que a partir de então pode se registrar no sistema e iniciar sessões com a QoS contratada.

Autenticação do usuário e autorização de uso dos serviços

O SAAA é o módulo responsável pela verificação das credenciais do usuário (autenticação) e dos seus direitos de uso (autorização) e pelo registro do seu consumo (contabilidade). Quando o A-SSN solicita um registro de usuário ou o início de uma sessão ao SAAA, este responde informando [Baresse 2005]:

- O resultado da autenticação/autorização (booleano);
- A razão com o código de uma eventual negação;
- O identificador do usuário (no registro) ou da sessão (no estabelecimento de sessão), que é único no sistema EuQoS;
- O tempo de validade do registro, após o qual deverá ser refeito (apenas para o registro);
- Uma lista com os perfis de QoS que o usuário pode utilizar, na fase de início de sessão, começando pelo perfil mais alto contratado, até o de menor qualidade permitida, para a aplicação específica. Por exemplo, se o usuário assinou um perfil de QoS Silver, ele poderá usar os perfis Silver, Bronze ou mais baixos.

O SAAA mantém as informações dos usuários assinantes do EuQoS no AS local em um arquivo XML (users.xml), a partir das informações passadas pelo CHAR. Na segunda

fase do projeto EuQoS, está prevista a extensão para a Autenticação e Autorização de usuários usando um SAAA remoto, para suportar a mobilidade de usuários.

O arquivo users.xml contém (conforme ilustrado na Fig. 28):

- A relação de usuários, cada usuário com login, senha e lista de aplicações;
- Para cada aplicação pode haver uma lista de conexões possíveis, cada uma contendo a relação de mídias, a classe de QoS (Gold, Silver, Bronze), e o tipo de aplicação;
- Para cada mídia há uma descrição da mídia (porta, protocolo, tipo de mídia) e uma relação de codecs;
- Cada codec por sua vez especifica os seus parâmetros de QoS, em termos de vazão, atraso, variação de atraso e taxa de erros.

Contabilidade e Faturamento

O A-SSN envia os dados da sessão para o SAAA, no início da sessão, durante o andamento da sessão, e no final da sessão, para fins de contabilidade de uso. O SAAA mantém uma base de dados com esses registros para repassar de maneira consolidada ao CHAR, para o respectivo faturamento (*IP Detail Record*, ou IPDR).

Essas sessões são contabilizadas pelo SAAA, que notifica o CHAR a cada evento de início, alteração ou finalização de sessão. Ao receber essas notificações, o CHAR solicita os registros da contabilização de uso do usuário e processa esses registros de acordo com a política de faturamento do NSP e o contrato do usuário, gerando as informações de faturamento. O usuário pode então acessar as informações sobre a sua conta através da interface WEB do CHAR, usando para isso suas credenciais [Angori 2005].

```
<?xml version="1.0" encoding="UTF-8"?>
<users>
   <user>
      <username>app2@euqos.org</username>
       <password>app2</password>
       <applicationList>
          <connectionCharacteristics>
             <mediaList>
                <mediaDescription transportProtocol="RTP/AVP"
                                  mediaType="audio"
                                  sourcePort="8080"
                                  destinationPort="1050"
                                  reservationDirection="send"
                                  sendStrength="optional"
                                  recvStrength="mandatory">
                  <codecList>
                       <codec codecName="GSM">
                          <qosCharacList>
                            <qosCharacteristics peakBitRate="50"
                                                maxJitter="1000"
                                                maxDelay="2000"
                                                maxLoss="100.0"/>
                          </qosCharacList>
                      </codec>
                  </codecList>
               </mediaDescription>
            </mediaList>
            <sessionUserQoS requestedUserClass="Gold"
                            applicationType="MMConferencing"/>
         </connectionCharacteristics>
      </applicationList>
   </user>
   <user>
      <username>app@euqos.org</username>
      <password>app</password>
      <applicationList>
         <connectionCharacteristics>
            <mediaList>
               <mediaDescription transportProtocol="RTP/AVP"
                                 mediaType="audio"
                                 sourcePort="4502"
                                 destinationPort="1050"
                                 reservationDirection="send"
                                 sendStrength="optional"
                                 recvStrength="mandatory">
                  <codecList>
                     <codec codecName="GSM">
                        <qosCharacList>
                           <qosCharacteristics peakBitRate="50"
                                               maxJitter="1000"
                                               maxDelay="2000"
                                               maxLoss="100.0"/>
                        </qosCharacList>
                     </codec>
                  </codecList>
               </mediaDescription>
            </mediaList>
            <sessionUserQoS requestedUserClass="Gold"
                            applicationType="MMConferencing"/>
         </connectionCharacteristics>
      </applicationList>
   </user>
</users>
```

FIGURA 28: Exemplo de Arquivo de configuração de usuários do SAAA (users.xml)

Limitação do modelo de usuário

Uma limitação do EuQoS é o fato de o seu modelo de usuário não diferenciar usuários de clientes. Em um ambiente corporativo real, onde vários usuários trabalham em uma mesma empresa, não há como agrupar o consumo e o faturamento para a empresa, pois os registros são feitos por usuário.