UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO DE ESPECIALIZAÇÃO EM CONFIGURAÇÃO E GERENCIAMENTO DE SERVIDORES E EQUIPAMENTOS DE REDE

GUSTAVO DE OLIVEIRA MATTAR NAVES

ESTUDO E IMPLEMENTAÇÃO DE QoS EM REDES IPv6

MONOGRAFIA

CURITIBA
2014

GUSTAVO DE OLIVEIRA MATTAR NAVES

ESTUDO E IMPLEMENTAÇÃO DE QoS EM REDES IPv6

Monografia apresentada como requisito parcial para a obtenção do grau de Especialista em Configuração e Gerenciamento de Servidores e Equipamentos de Redes, do Departamento Acadêmico de Eletrônica da Universidade Tecnológica Federal do Paraná – UTFPR
Orientador: Prof. MSC. Juliano de Mello

CURITIBA
2014

RESUMO

NAVES, Gustavo O. M. **ESTUDO E IMPLEMENTAÇÃO DE QoS EM REDES IPv6.** 2014. 56 PÁGINAS. Monografia (Especialização em Configuração e Gerenciamento de Servidores e Equipamentos de Redes). Universidade Tecnológica Federal do Paraná. Curitiba, 2014.

A presente monografia aborda o estudo para a implementação de técnicas de QoS (qualidade de serviço) em redes puramente IPv6. Apresentam-se aqui as principais premissas para utilização do QoS, convergência da rede IPv6 e definição de fluxos prioritários, para que deste modo, a implementação e aplicação de QoS não sejam em vão e possibilite realmente uma melhora exponencial no tráfego priorizado. O projeto inicia utilizando método bibliográfico, seguido de um estudo em campo, emulação da rede com máquinas virtuais, saturação e posteriormente análise dos resultados obtidos. A conclusão mostrará a eficácia de uma rede com QoS aplicada e funcionando de acordo com a necessidade de cada administrador da rede.

Palavras-chave: Redes. Qualidade de Serviço. Priorização de Tráfego. IPV6. Desempenho

ABSTRACT

NAVES, Gustavo O. M. **IMPLEMENTATION AND STUDY OF APLIED QoS IN IPv6 NETWORKS.** 2014. 56 PAGES. Monograph (Specialization in Configuration and Management of Servers and Network Equipments) – Federal Techonological University of Paraná. Curitiba, 2014.

This monograph deals with the study for the implementation of QoS (quality of service) techniques in pure IPv6 network topologies. Presents the main advantages for using QoS, how to define priorities on a QoS table and how to converge an IPv6 network. For thus QoS implementation and application is not in vain and really allows an exponential improvement in prioritized traffic. The project starts up with bibliographic method, followed by laboratory study, than the GNS3 lab will be configured with virtual machine, network saturation and subsequent analysis of the results. The results show the effectiveness of a network with QoS implemented and operates in accordance with the needs of each network administrator.

Keywords: Network, Quality of Service, Traffic Priority, IPv6, Performance.

LISTA DE SIGLAS

ARPA - Advanced Research Projects Agency

BGP - Border Gateway Protocol

BSS - Basic Service Set

CIR - Commited Information Rate

CSMA/CA - Carrier Sense Multiple Access with Collision Avoidance

DNS - Domain Name System

DHCP - Dynamic Host Configuration Protocol

DSCP - Differentiated Services Code Point

DSSS - Direct Sequence Spread Spectrum

ESS - Extended Service Set

FHSS - Frequency Hopping Spread Spectrum

FTP – File Transfer Protocol

GHz – Giga Hertz

GIF - Graphics Interchange Format

GLP - General Public Licence

HTTP - Hypertext Transfer Protocol

ICMP - Internet Control Message Protocol

IEEE - Institute of Electrical and Eletronics Engineers

IP – Internet Protocol

JPEG - Joint Photographic Experts Group

LAN – Local Area Network

LLC - Logical Link Control

MAC - Media Access Control

Mbps - Megabits por Segundo

MIMO - Multiple-Input Multiple-Output

MPEG - Motion Picture Experts Group

MPLS - Multi-Layer Protocol Label Switching

MPR - Multipoint Relays

NAT - Network Address Translation

OFDM - Orthogonal Frequency Division Multiplexing

OLSR - Optimized Link State Routing

OSI - Open Systems Interconnection

OSPF - Open Shortest Path First

PCI - Protocol Control Information

PDU - Protocol Data Unit

PHB - Per-Hop Behavior

QoS – Quality of Service

RFC - Request for Comments

RIP - Routing Information Protocol

RSVP - Resource Reservation Protocol

SDU - Service Data Unit

SIP - Session Initiation Protocol

SLA - Service Level Agreement

SNMP - Simple Network Management Protocol

TCP - Transmission Control Protocol

TCP/IP - Transmission Control Protocol over Internet Protocol

ToS – Type of Service

TTL – Time to Live

UDP - User Datagram Protocol

VoIP – Voice over Internet Protocol

LISTA DE ILUSTRAÇÕES

SUMÁRIO

1. INTRODUÇÃO

Neste capítulo serão tratados os elementos introdutórios relacionados ao estudo e implementação de técnicas de qualidade de serviço em redes IPv6.

1.1. TEMA

Com o crescimento e popularização da internet, a utilização do protocolo de IPv4 foi se tornando insuficiente para a enorme quantidade de elementos da nova geração de Redes, visto esse problema o IETF (*Internet Engineering Task Force*) desenvolveu o *Internet Protocol* (IP) *version* 6 (IPv6) com o objetivo de substituir a versão atual IPv4. O IPv6 além de resolver o problema da quantidade de endereçamentos IP disponíveis na Internet, adicionou também melhorias como auto configuração, mobilidade e buscou resolver o problema existente da qualidade de serviço (QoS). Este trabalho tem como objetivo estudar como a garantia da qualidade de serviço beneficia a implementação de QoS nas redes IPv6 e se realmente existe um ganho considerável em comparação com o IPv4. Em comunicação de dados, as informações necessitam de um meio para chegar ao seu destino, seja fibra ótica, cabos de cobre ou ar. Uma sequência de pacotes desde uma origem até um destino é chamada fluxo. Em uma rede orientada a conexão, todos os pacotes que pertencem a um fluxo seguem mesma rota, já em uma rede sem esta especificação, eles provavelmente seguiriam caminhos diferentes. As necessidades podem ser caracterizadas por quatro parâmetros principais: confiabilidade, retardo, flutuação e largura de banda. Juntos, esses parâmetros definem a *Quality of Service* – Qualidade de Serviço (QoS) que o fluxo exige (TANENBAUM, ANDREW S., 2003, p.307).

1.2. OBJETIVOS

Nesta sessão serão trabalhados objetivos gerais e objetivos específicos.

1.2.1. Objetivo Geral

O principal objetivo deste projeto é analisar a implementação de QoS em redes IPv6.

1.2.2. Objetivos Específicos

- Identificar a necessidade da utilização de QoS em redes;
- Descrever as principais situações que dependem da aplicação da QoS para melhor funcionamento;
- Exemplificar como deverá ser feito a aplicação de QoS em redes IPv6;
- Efetuar testes em topologia IPv6;
- Analisar a rede após a inserção de configurações de QoS;
- Comparar a eficácia das redes antes e depois da configuração de QoS;

1.3. JUSTIFICATIVA

Atualmente existem poucas literaturas abordando um estudo comparativo de melhoria do QoS entre redes IPv6 e IPv4. Esse estudo é importante visto o caminho que a “Internet das Coisas” vem tomando e como, cada vez mais, o mercado exige maior velocidade e confiabilidade nas transmissões de dados.

Com base nos resultados dos testes que serão realizados, este trabalho apresentará um parecer se realmente o IPv6 consegue melhorar de forma significativa a implantação de QoS em comparação ao IPv4, como devemos implementar corretamente QoS em redes IPv6 e alguns motivos para a utilização da QoS nas redes atuais.

1.4. PROCEDIMENTOS METODOLÓGICOS

Seguindo a linha de raciocínio de Gil (2002) sobre a classificação das pesquisas e levando em consideração os objetivos de cada uma, este trabalho de monografia estará seguindo os procedimentos técnicos de pesquisa bibliográfica e estudo de campo. Pesquisa bibliográfica, pois é desenvolvida com base em material já elaborado, constituído principalmente de livros e artigos científicos. A principal vantagem da pesquisa bibliográfica reside no fato de permitir ao investigador a cobertura de um gama de fenômenos muito mais ampla do que aquela que poderia pesquisar diretamente (GIL, Antônio Carlos, 2002, p. 44-45). Já o estudo de campo é definido, pois procura mais o aprofundamento das questões propostas do que a distribuição das características da população segundo determinadas variáveis. Como consequência, o planejamento do estudo de campo apresenta muito maior flexibilidade, podendo ocorrer mesmo que seus objetivos sejam reformulados ao longo da pesquisa. Outra distinção é que no levantamento das informações procura-se identificar as características dos componentes do universo pesquisado, possibilitando a caracterização precisa de seus segmentos (GIL, Antônio Carlos, 2002, p. 53).

1.5. ESTRUTURA

A monografia é composta por 4 capítulos. Primeiramente, o capítulo 1, tratará da parte introdutória, sendo apresentados o tema, os objetivos a serem atingidos, a justificativa da escolha e os problemas a serem resolvidos. Também nesta primeira parte, apresenta-se o embasamento teórico, procedimento metodológico e a estrutura da monografia.

O capítulo 2 trata do referencial teórico do projeto. Teoria sobre redes, modelos de referência em camadas *Open System Connection* (OSI) e *Transmission Control Protocol over Internet Protocol* (TCP/IP), protocolo IPv6 seus métodos de implementação, regras de endereçamento e por fim a apresentação da QoS. Este capítulo trará de forma clara e objetiva os conceitos de rede que qualquer administrador deve conhecer antes de aplicar QoS em sua estrutura ou

até mesmo antes de promover qualquer mudança na arquitetura de seu rede. Tratá também uma explicação sobre o funcionamento de QoS, como por exemplo, marcação dos pacotes por portas, por endereçamento IP e por aplicação utilizada.

Partindo para a parte prática do estudo, o capítulo 3 mostrará os passos seguidos para a configuração do emulador GNS3, das máquinas virtuais no *Virtual Box* da Oracle e também a aplicação das ferramentas de QoS disponíveis nos roteadores CISCO. Com isso, associa-se a parte teórica (marcação dos pacotes) com a parte prática (como marcar os pacotes). O estudo de campo será visto neste mesmo capitulo, onde a emulação de uma rede puramente IPv6 fará uma análise do comportamento com e sem QoS aplicado. A partir dos resultados obtidos, poderá se afirmar que a configuração de QoS é necessária e em quais situações torna-se praticamente obrigatória. Finalizando a monografia, o capítulo 4 traz as conclusões sobre o estudo como um todo e também quesitos comumente vistos após esta sessão, como as referências.

2. REFERENCIAIS TEÓRICOS

2.1. REDES DE COMPUTADORES

As redes atuais de computadores foram criadas nos anos 70, essas redes ainda eram extremamente limitadas e somente computadores dos mesmos fabricantes conseguiam comunicar entre si.

Apenas em 1974 a IBM (International Business Machines) publicou o seu modelo de redes, Arquitetura de Redes de Sistemas (Systems Network Architecture, ou SNA), após essa divulgação, os fabricantes das mais variadas marcas passaram a utilizar o SNA para que fosse possível a interconexão entre seus produtos e os da IBM. Aos poucos essa solução mostrou-se negativa visto que com o passar do tempo os maiores fabricantes de computadores poderiam dominar o mercado de redes. (ODOM, 2008).

A fim de resolver este problema na década de 1980, criou-se um grupo de trabalho que iria criar um modelo de redes padronizado e aberto. E em 1983 foi apresentado pela ISO (International Standards Organization) o modelo de referência OSI (Open Systems Interconnection), que seria o primeiro passo em direção à padronização internacional de interconexão de sistemas abertos. (TANENBAUM, 2011)

2.2. MODELO DE REFERÊNCIA OSI

O modelo de referência OSI foi definido em camadas, sete exatamente, onde cada uma possui sua função na pilha de protocolos. Cada camada interage com sua correspondente no equipamento remoto, ou seja, camada 3 de uma estação local só troca informações com camada 3 da estação remota. Contudo, não se deve confundir comunicação entre camadas correspondentes, com encapsulamento de dados, pois dados oriundos de aplicativos da camada sete são encapsulados dentro do formato oferecido pela camada imediatamente inferior, camada 6. As sete camadas mencionadas são nomeadas, da mais superior (camada 7) para a mais inferior (camada 1), da seguinte maneira: Aplicação, Apresentação, Sessão, Transporte, Rede, Enlace e Física.

(MEGGER, Chrystian L., 2011). A seguir, na figura 2 é ilustrado o Modelo de Referência OSI:

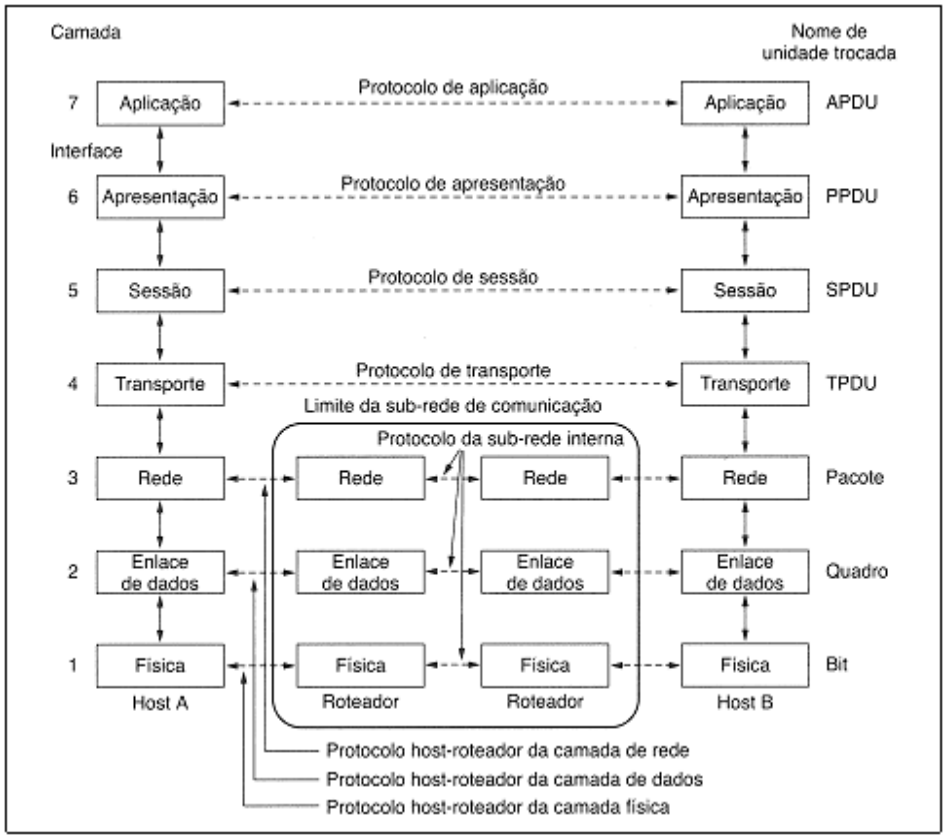

Figura 1 - O modelo de referência OSI - Fonte: TANENBAUM, 2011.

2.2.1. Camada de Aplicação

A camada de aplicação é a que mais se interage com o usuário. Nesta camada encontram-se as principais aplicações (softwares / protocolos) utilizadas atualmente, como servidores de e-mail, navegador web, banco de dados, DNS, DHCP etc (MEGGER, Chrystian L., 2011).

A transferência de um arquivo entre dois sistemas requer uma forma de trabalhar com as incompatibilidades existentes. A camada de aplicação tem grande importância na resolução deste problema. O dado entregue pelo usuário a camada de aplicação recebe a denominação *Service Data Unit* (SDU), e então, junta a ela (dados do usuário) um cabeçalho chamado *Protocol Control Information* (PCI). O objetivo resultante desta junção é chamado de *Protocol*

Data Unit (PDU), que corresponde à unidade de dados especificada de um certo protocolo da camada em questão (PINHEIRO, JOSÉ MAURICIO S., 2004).

2.2.2. Camada de Apresentação

A camada de apresentação responde às solicitações de serviço da camada de aplicação e envia solicitações de serviço para a camada imediatamente inferior (sessão). Diferentemente das camadas mais inferiores, preocupadas em mover bits de forma confiável de um ponto a outro, essa camada preocupa-se com a sintaxe e a semântica dos dados transmitidos. Por exemplo, após receber dados da camada de aplicação, pode ser necessário converter esses dados de seu formato original para um formato compreendido e aceitável por outras camadas do modelo, garantindo assim uma transmissão mais eficiente. Exemplos de formatações incluem PostScript, ASCII, EBCDIC e ASN.1 (FILIPPETTI, MARCO AURÉLIO, 2008, p. 43).

2.2.3. Camada de Sessão

Segundo (FELIPPETTI, 2009): “Ela é responsável pelo estabelecimento, gerenciamento e finalização de sessões entre a entidade transmissora e a entidade receptora. Ela basicamente mantém os dados de diferentes aplicações separados uns dos outros.”.

Um dos diversos serviços oferecidos é o controle de diálogo (quem deve transmitir ao longo do tempo), gerenciamento e sincronização das transmissões caso ocorra uma falha. (TANENBAUM, 2011).

2.2.4. Camada de Transporte

A camada de transporte, segundo (MEGGER, Chrystian L.,2011) é responsável pela segmentação e controle de fluxo, trabalhando com dois protocolos de comunicação mais comumente encontrados, o TCP e o UDP. Esta camada recebe os dados da camada superior (Sessão), divide-os em unidades menores e repassa esses segmentos para a camada de rede, assegurando que todas as informações chegarão ao destino na ordem correta e sem erros.

Também conforme descrito por (MEGGER, Chrystian L.,2011): “...o controle de fluxo proporcionado pela camada de transporte garante uma conexão lógica ponto a ponto e gerencia o fluxo de dados fim a fim, onde o

destino envia a confirmação dos dados recebidos e aguarda a chegada dos demais segmentos para fazer a reconstrução da informação". Caso essa confirmação não seja recebida os dados são retransmitidos.

O controle de fluxo também tem a função de evitar o congestionamento e/ou sobrecarga da rede, para essa atividade existe um recurso chamado *buffer*, que atua armazenando dados e informações para que eles possam ser processados no destino na sequência correta ou descartados caso seus dados antecessores e/ou predecessores não cheguem ao destino.

Observa-se este tipo de situação de controle de fluxo quando utiliza-se o protocolo TCP, pois este é orientado a conexão e garante a confiabilidade e integridade na entrega dos dados. A contrapartida da utilização do TCP é a perda na velocidade de envio e processamento das informações. Exemplos clássicos de utilização do TCP navegadores web e e-mails.

Ao contrário do TCP e também definido na camada de transporte, encontra-se o protocolo UDP. Este protocolo possui *overhead* baixo, não é orientado a conexão e não oferece dispositivos de controle de fluxo sofisticados. Apesar de ser um protocolo simples e possuir apenas as funções básicas da camada de transporte, o UDP é mais rápido em relação ao TCP, o que o torna eficaz em transmissões de voz sobre IP (Voip) e gerenciamento de equipamentos (protocolo SNMP). Contudo, as transmissões estarão sujeitas a perdas, o que pode ser prejudicial em uma conversa Voip, por exemplo. Neste quesito de definirem-se prioridades de tráfego na rede (engenharia de tráfego) é que se encaixa a aplicação de QoS, o qual será visto posteriormente (MEGGER, Chrystian L.,2011).

2.2.5. Camada de Rede

A camada de rede é a responsável pelo roteamento dos pacotes entre a fonte e seu destino. Nessa camada estão alocados e tem papel fundamental na estrutura de rede atual, os roteadores.

Segundo Felippetti (2009, p.47): "Roteadores ou "routers" - também chamados de dispositivos de camada 3 (layer 3 devices) - são definidos nessa camada e provêm todos os serviços relacionados ao processo de roteamento.".

Existem dois tipos de pacotes definidos nesta camada: pacotes de dados (data packets) e pacotes de atualização (router update packets). (FELIPPETTI, 2009).

É nessa camada que encontramos o protocolo IP, que é estudado nessa monografia, essencialmente a evolução do IPv4 para IPv6 e o possível incremento na eficiência do QoS dessa evolução.

2.2.6. Camada de Enlace

A camada de enlace é responsável pela adaptação dos dados recebidos da camada superior (Rede) para que possam ser transmitidos através do meio físico. Essa camada também atua no controle de fluxo, permitindo que as diferenças de capacidade de transmissão entre meios diferentes não impeça a transmissão de acontecer.

A camada de enlace formata a mensagem em frames e adiciona um cabeçalho customizado contendo o endereço de hardware (MAC *Address*) das máquinas transmissora e destinatária. É importante também entender que para a camada de rede (onde os roteadores são definidos) não importa a localização física das máquinas, mas a localização lógica das redes. A camada de enlace (onde switches e *bridges* são definidos), sim, é responsável pela identificação de cada máquina (MAC *address*) em uma rede local (FILIPPETTI, MARCO AURÉLIO, 2008, p. 49).

Outras funções exercidas pela camada de enlace são detecção e correção de erros. Essa função é exercida através de alguns algoritmos matemáticos, como: Checksum, paridade e CRC (código de redundância cíclica).

2.2.7. Camada de Física

A primeira camada do modelo de referência OSI é camada física e esta tem por seu principal objetivo a transmissão dos bits (sinal elétrico, óptico ou microondas) que formam os quadros (*frames*) da camada de enlace através dos meios cabos, fibras ópticas ou ar. Assim como a transmissão, é também de responsabilidade da camada física a recepção e organização dos sinais, de modo que estes, ao serem enviados para a camada superior, formem um *frame* completo.

2.3. O MODELO DE REFERÊNCIA TCP/IP

Segundo Tanembaum (2011, p.28): “A ARPANET era uma rede de pesquisa patrocinada pelo Departamento de Defesa dos Estados Unidos (DoD).”. Com a evolução das redes de comunicação, divergindo da interconexão exclusiva de linhas telefônicas, houve uma procura por outra arquitetura de referência que resolvesse o problema de interligação das novas tecnologias com os protocolos existentes. Pouco tempo depois a resolução de heterogeneidade dessas diferentes redes foi alcançada pela criação do Modelo de Referência TCP/IP. (TANENBAUM, 2011).

O Modelo de Referência TCP/IP, ficou organizado em quatro camadas: Aplicação, Transporte, Inter-rede e Host/Rede. Abaixo, a figura 3 expõe lado a lado os dois modelos discutidos até este ponto.

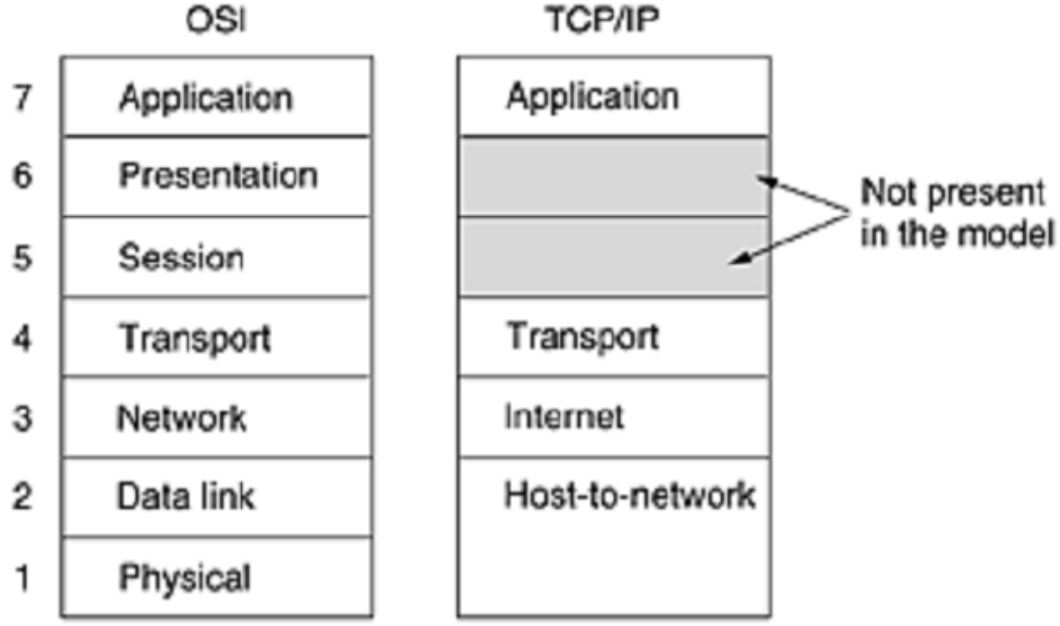

Figura 2 - Comparação da Arqiutetura OSI e TCP/IP

Fonte: TANENBAUM, 2011.

2.4. O PROTOCOLO IPV6

O protocolo IPv6 começou a ser pensando devido a necessidade de melhorias no protocolo antecessor o IPv4 e também ao iminente esgotamento deste protocolo. Além da melhoria e do esgotamento, a criação de um novo protocolo de comunicação IP tornou-se cada vez mais necessário devido ao o que chamamos de “Internet das Coisas” pois o protocolo atual foi criado pensando em redes usadas de forma mais ampla, por universidades, industrias de tecnologia e órgão do governo americano. Conforme descrito por Tanenbaum,2011: “com a inevitável convergência das indústrias de informática, comunicação e entretenimento, talvez não demore para que cada telefone e cada televisor do mundo seja um nó da Internet, resultando no uso de áudio e vídeo por demanda em um bilhão de máquinas. Sob essas circunstâncias, ficou claro que o IP precisava evoluir para se tornar mais flexível.

Abaixo temos o que foi listado pela IETF como os principais objetivos do desenvolvimento do IPv6, conforme Tanenbaum,2011:

Aceitar bilhões de hosts, mesmo com alocação de espaço de endereços ineficiente.

Reduzir o tamanho das tabelas de roteamento.

Simplificar o protocolo, de modo a permitir que os roteadores processem os pacotes com mais rapidez.

Oferecer mais segurança (autenticação e privacidade) do que o IP atual.

Dar mais importância ao tipo de serviço, particularmente no caso de dados em tempo real.

Permitir multidifusão, possibilitando a especificação de escopos.

Permitir que um host mude de lugar sem precisar mudar o endereço.

Permitir que o protocolo evolua no futuro.

Permitir a coexistência entre protocolos novos e antigos

Com o intuito de cumprir esse objetivo o IEFT convocou a RFC 1550, para que fossem apresentadas propostas de novos protocolos e modelos de melhorias. Foram recebidas 21 propostas de consideradas válidas e estudadas destas sete propostas, três foram consideradas as melhores, a de Deering; Francis e de Katz e Ford. Porém dessas 03 duas foram consideradas de forma

combinadas dando origem a SIPP (Simple Internet Protocol Plus), que ficou designada como IPv6.

Enfim, o novo protocolo, IPv6, atende, na teoria todas as necessidades divulgadas pelo IETF sendo que ele, de forma geral, preserva as “boas” características do IPv4 e descarta, ou diminui a importância, quando necessário, as características que ruins e cria outras.

Os principais recursos do IPv6 são, descritos de forma resumida, os abaixo listados:

IPv6 tem endereços mais longos que o IPv4. Ele tem 16 bytes o que resolve o problema de endereços limitados do IPv4

O cabeçalho deste novo protocolo foi simplificado, tendo somente 07 campos, contra 13 do IPv4. Isso reduz o tempo de processamento dos roteadores que processam assim os pacotes com maior velocidade. Isso ajuda a melhorar o *troughput* e reduzir o *delay.*

Os campos que até então eram obrigatórios agora são opcionais. Além disso, é diferente a forma como as opções são representadas, o que torna mais simples para os roteadores ignorar as opções a que eles não se propõem. Esse recurso diminui o tempo de processamento de pacotes (TANENBAUM, 2011)

A segurança seria a quarta melhoria no IPv6 em comparação ao IPv4, mas recentemente essas melhorias também foram introduzidas no protocolo atual.

O novo protocolo, também foi estruturado com melhoria na qualidade de serviço, possuindo campos específicos para determinação de fluxo e prioridade nas conexões.

2.4.1. Estrutura do Pacote IPv6

Sabendo a estrutura do cabeçalho padrão do IPv6 e também dos cabeçalhos de extensão, temos agora a estrutura completa de uma pacote IPv6.

A estrutura final de um pacote IPv6 conforme Oliveira, Clécio é o seguinte:

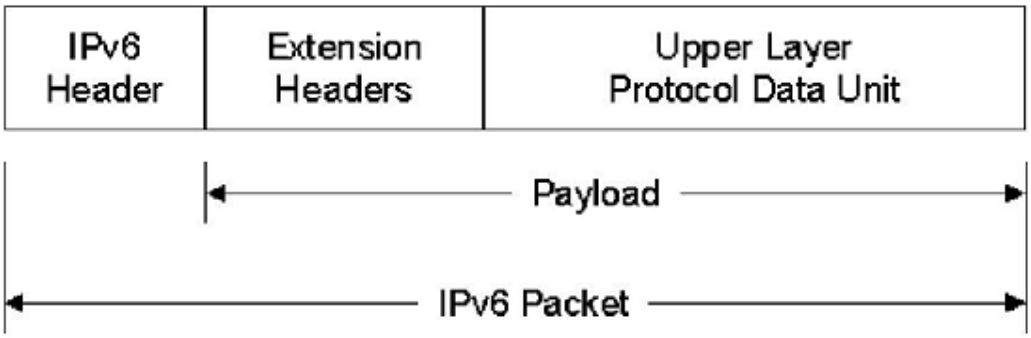

Figura 3 - Estrutura do Pacote IPv6

Fonte: OLIVEIRA, CLÉCIO 2011.

2.4.2. O cabeçalho IPv6

O cabeçalho IPv6 foi elaborado com alterações em relação ao IPv4 com o objetivo de reduzir o processamento dos roteadores que são responsáveis de propagar esses datagramas. Isso foi possível ao se reduzir a quantidade de informações no cabeçalho, retirando alguns campos como o Header Length, Identification, Fragment Offset, Flags e o Header Checksum. Em especial o checksum além da redução do datagrama, contribui com a redução de processamento dos roteadores pois estes agora não precisam mais realizar a verificação em cada quadro que é transportado.

O tamanho do cabeçalho agora é fixo, diferente do cabeçalho IPv4 que tinha, no mínimo 20 bytes e no máximo 60 bytes. O cabeçalho IPv6 tem 40 bytes de tamanho. Outra alteração é que campos não necessários e opcionais foram transformados em cabeçalhos de extensão.

Abaixo temos o cabeçalho IPv6:

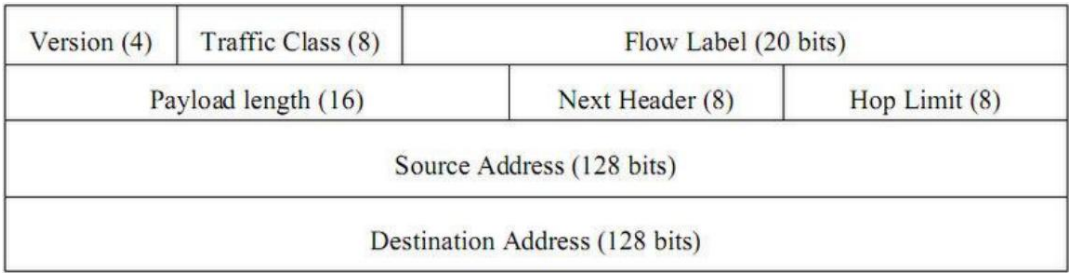

Figura 4 - Formato do cabeçalho IPv6

Fonte: PEARCE et. al., 2010.

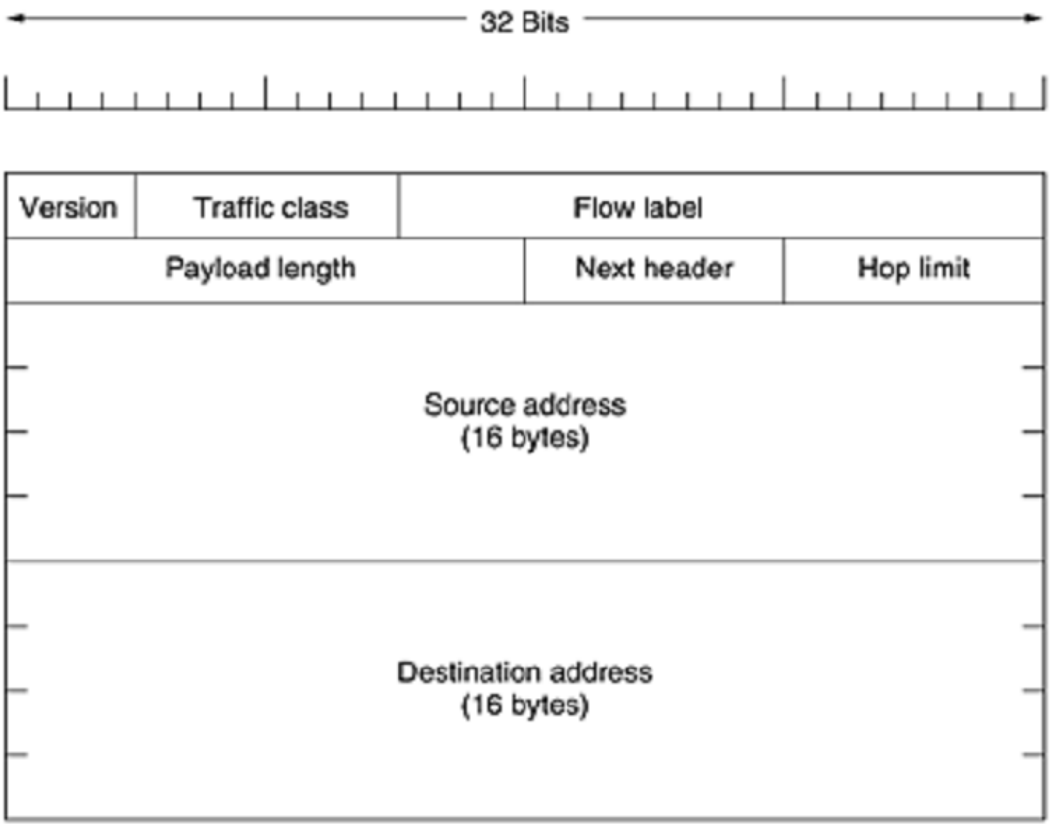

Figura 5 - Formato do cabeçalho IPv6

Fonte: TANENBAUM, 2011.

O campo version é responsável por informar a versão no protocolo IP, nestes caso sempre será 6.

O campo *Traffic class* é responsável pela distinção de pacotes devido a seus requisitos de entrega como, por exemplo, aconteceria para um pacote de voz que precisa de entrega em tempo real.

De acordo com Filipetti (2008, p. 174): "*Traffic class* é usado para assinalar a classe de serviço a que o pacote pertence, permintindo assim dar diferentes tratamentos a pacotes provenientes de aplicações com exigências

distintas. Este campo serve de base para o funcionamento do mecanismo de QoS na rede”

O campo *Flow label*, é utilizado para estabelecer prioridade na transmissão dos pacotes através da verificação, pelos roteadores, para um tratamento especial. Segundo Tanenbaum (2011, p. 359): “Quando um pacote com o campo *Flow label* com valor diferente de zero aparece, todos os roteadores podem verificar nas tabelas internas que tipo de tratamento especial ele exige.”

Esses são dois campos importantes na utilização de QoS em redes IPv6, que serão avaliados na parte prática desse trabalho.

Os outros campos do IPv6 são o *Payload Length* que é utilizado para indicar a quantidade de dados que serão enviados sequencialmente ao cabeçalho, o *Next Header* indica qual ou quais cabeçalhos de extensão podem vir a seguir esse cabeçalho inicial ou, caso não exista nenhum cabeçalho de extensão ligado ele indicará o tipo de informação a ser tratado na camada de transporte. O próximo campo é o *Hop Limit* que substitiu o *Time To Live* do IPv4 que tem em sua importância evitar que pacotes fiquem eternamente sendo propagados.

Por último vem os campos de *Source Address* e *Destination Address* que conforme descrito pelos nomes indicam o endereço de onde o pacote IP está saindo e qual será seu destino.

2.4.3. Fragmentação de dados no IPv6

Em redes IPv6, como descrito no capítulo anterior, não teremos mais fragmentação dos pacotes, feita pelos roteadores. Isso ocorre por que no IPv6 os roteadores determinam, através de um *feature* qual é o *Maximum Transmit Unit* (MTU) da rede na qual ele está inserido, caso o host envie um pacote que não seja compatível com essa MTU, o roteador devolve ao host uma mensagem de erro. Assim o próprio Host fica responsável em dividir os pacotes em tamanho compatíveis com a MTU da rede.

2.4.4. Cabeçalhos de Extensão

São cabeçalhos complementares e não obrigatórios que podem ser adicionado ao IPv6, eles vem depois do cabeçalho padrão e dão uma indicação específica sobre o roteamento dos pacotes IPv6.

Em resumo os cabeçalhos de extensão são os definidos pela RFC 2460 e estão resumidos na imagem abaixo:

Cabeçalho de extensão	Descrição
Hop-by-hop options	Informações diversas para os roteadores
Destination options	Informações adicionais para o destino
Routing	Lista parcial de roteadores a visitar
Fragmentation	Gerenciamento de fragmentos de datagramas
Authentication	Verificação da identidade do transmissor
Encrypted security payload	Informações sobre o conteúdo criptografado

Figura 6 - Cabeçalhos de Extensão IPv6

Fonte: TANENBAUM, 2011.

Outro ponto importante é que os cabeçalhos de extensão podem ser atualizados e que podem ser criadas novas funções, atendendo assim um dos objetivos na criação do IPv6 que é a escalabilidade e melhor eficiência.

2.4.5. Endereçamento IPv6

A maior mudança do IPv4 para o IPv6 é a mudança do tipo de endereçamento aonde no IPv4 tínhamos 32 bits e no IPv6 temos 128 bits ou 16 bytes, essa alteração tem como objetivo garantir que o IPv6 não termine tal como aconteceu com o IPv4.

Segundo (TANENBAUM, 2011): “existem muitos endereços de 16 bytes. Especificamente, existem 2^{128} endereços desse tipo, o que significa cerca de $3x10^{38}$. Se colocássemos um computador em cada pedaço de terra e água do nosso planeta, o IPv6 permitiria $7x10^{23}$ endereços IP por metro quadrado. Os estudantes de química perceberão que esse número é maior que o número de Avogadro. Embora não exista a intenção de dar a cada molécula na superfície da Terra seu próprio endereço IP, não estamos longe de chegar a essa marca.”

Ou seja, temos agora endereços suficientes para muitos anos e também para atender a latente demanda da “internet das coisas”.

Como definido na RFC 4291, o endereço IPv6 é representado por oito grupos de um a quatro dígitos hexadecimais, separados por “:”. Três abreviações são permitidas, zeros a esquerda no mesmo bloco podem ser omitidos, longas

sequencias de zeros pode ser substituídos por um par de dois-pontos e endereços IPv4 podem ser escrito por um par de dois-pontos e sequencialmente o próprio número tradicional.

Alguns exemplos de abreviações:

8000:0000:0000:0000:0123:4567:89AB:CDEF

8000::123:4567:89AB:CDEF – endereço hexadecimal reduzindo, seguindo a primeira e segunda regra descritas acimas.

::192.31.20.46 – Endereço IPv4 clássico escrito em formato IPv6.

Fonte: TANENBAUM, 2011

Além da nova estrutura o IPv6 tem seus endereços especiais, que são:
Unicast Address – que identifica uma interface única de um host, assim sendo um envio *unicast* será entrege a um único destinatário
Multicast Address – que identifica várias interfaces IPv6, aonde um pacote *multicast* será entregue a várias interfaces em um nó específico.
Anycast Address – que identifica várias interfaces IPv6, mas o pacote entregue é entregue somente para a interface “mais próxima” dentro do nó.

Visto isso precisamos agora identificar e definir como são atribuídos os endereços aos hosts de rede e subrede no IPv6.

2.4.6. Atribuindo o endereço IPv6

Quando uma empresa adiquiri um range de endereços IPv6 de um ISP ela recebe, normalmente, um endereço /48. Para a atribuição interna de endereço de hosts da empresa, o engenheiro responsável quebra essa rede em um /64 e mapeia quantas sub-redes serão necessárias.

Neste cenário o /48 compõe com mais 16 bits o prefixo da sub-rede e em conjunto com cada Interface ID, geram o endereço IPv6 de 128 bits de cada host.

Por exemplo, uma empresa que precisa de 04 sub-redes, conforme descrito por (ODOM, 2008), teríamos a seguinte estrutura da empresa e também de cada endereço de host:

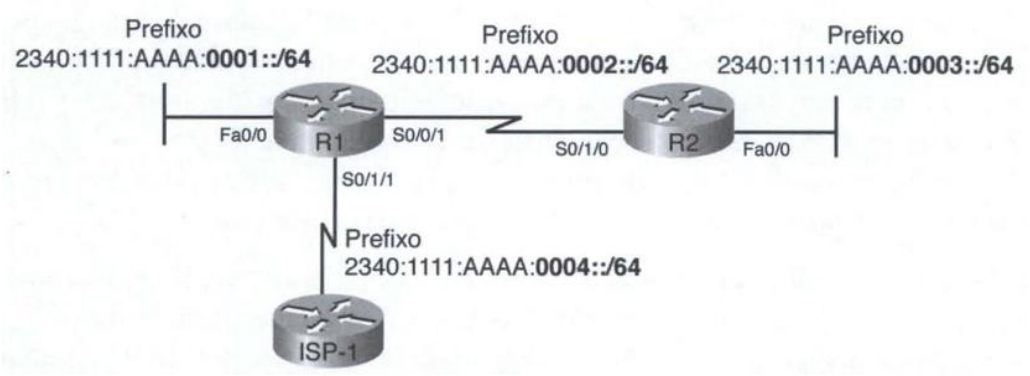

Figura 7 - Endereçamento IPv6 de Sub-Rede de uma empresa

Fonte: (ODOM, 2008)

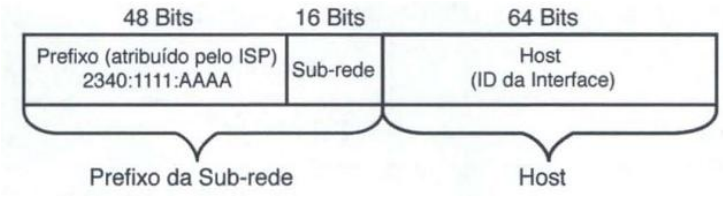

Figura 8 - Estrutura do endereço de Host IPv6

Fonte: (ODOM, 2008)

Especificamente o endereço de Host, composto por 64 bits, seria formado por uma estrutura que combina do endereço MAC do Hardware, dividido em duas partes e separado “ao meio” pelo hexa FFFE. Esse formato é chamado de EUI-64. Outro ponto de atenção nesse meio de estruturação é que o 7º bit da esquerda para a direita do primeiro byte precisa ser alterado para 1.

Segundo (ODOM, 2008): "A razão por detrás disso é que os endereços MAC Ethemet são apresentados com os bits de mais baixa ordem de cada byte na esquerda, e os bits de mais alta ordem na direita. Portanto, o oitavo bit em um byte (lendo da esquerda para a direita) é o bit de mais alta ordem do endereço, e o sétimo bit (lendo da esquerda para a direita) é o segundo bit de mais alta ordem. Este segundo bit de mais alta ordem no primeiro byte - o sétimo bit lendo da esquerda para a direita - é chamado de bit universa1/local (VIL). Quando configurado com o binário 0, significa que o endereço MAC é um endereço MAC gravado. Configurado em 1, significa que o endereço MAC foi configurado localmente. O EUI-64 diz que o bit VIL deve ser configurado com 1, o que significa local.”

Em resumo temos 04 meios de atribuição de endereço IPv6, duas de aprendizado dinâmico e duas de aprendizado estático. O método que

detalhamos neste capítulo foi o de atribuição estática usando o EUI-64, em resumo além do método descrito aqui, temos também a configuração estática sem utilização do EUI-64, configuração dinâmica usando DHCPv6 *stateful* que usam o endereço de 128 bits completos para ser configurado e a autoconfiguração stateless que volta a utilização somente do prefixo /64.

2.5. QUALIDADE DE SERVIÇO (QOS)

Visto a unificação cada vez maior dos meios de transmissões e a utilização da internet cada vez mais popularizada, a qualidade de serviço (QoS), vem se tornando uma ferramenta fundamental nas redes atuais. O QoS (*Quality of Service*) é um mundo muito específico e que demanda dos engenheiros de rede cada vez mais estudo e constante atualização para cada vez melhor ser aplicado nas redes atuais. A sua utilização é feita através de regras específicas para que as redes funcionem da melhor maneira possível e cada implementação de QoS deve ser estudada em cima da característica da rede a ser trabalhada. Uma analogia que explica essa questão é a feita por (BARREIROS & Lundqvist, 2011): “Permitir uma competição justa e igual, sem ter a diferenciação de trafego, não funciona, pois diferentes tráfegos demandam diferentes necessidades, tal como uma ambulância e um caminhão em uma estrada”.

Uma das primeiras tentativas para tentar resolver essa problema foi aumentar a largura de banda, mas isso vai contra toda tendência de mercado atual que é a redução de custos. Portanto o QoS é aplicado não com o intuito de deixar a “estrada” mais larga e sim de dividir os recursos dessa de maneira não igual, favorecendo alguns e preterindo outros.

Visto isso, na implementação de QoS o primeiro passo sempre é mapear e identificar quem deve ser favorecido e quem pode ser preterido, isso é feito analisando a característica de cada tipo de trafego que temos na rede tal como voz, vídeo, trafego de internet, e-mail, etc. E para cada tipo de rede podemos ter a combinação de várias técnicas de QoS para conseguirmos um melhor resultado.

Os principais fatores que influenciam na aplicação de QoS são Latência, *Jitter*, perda de pacotes. A a largura de banda (*bandwith*) deve ser tratada mais como recurso da rede, visto que normalmente os operadores de rede tem essa informação.

Latência é o tempo que o pacote demora para atravessar a rede, caso exista necessidade de confirmação de entrega esse tempo também é contabilizado.

O *Jitter* é a variação dos tempos de latência quando verificamos vários pacotes, ou seja, a latência do pacote 1 menos a latência do pacote 2, que seriam o delta1 e o respectivo delta2 da figura abaixo:

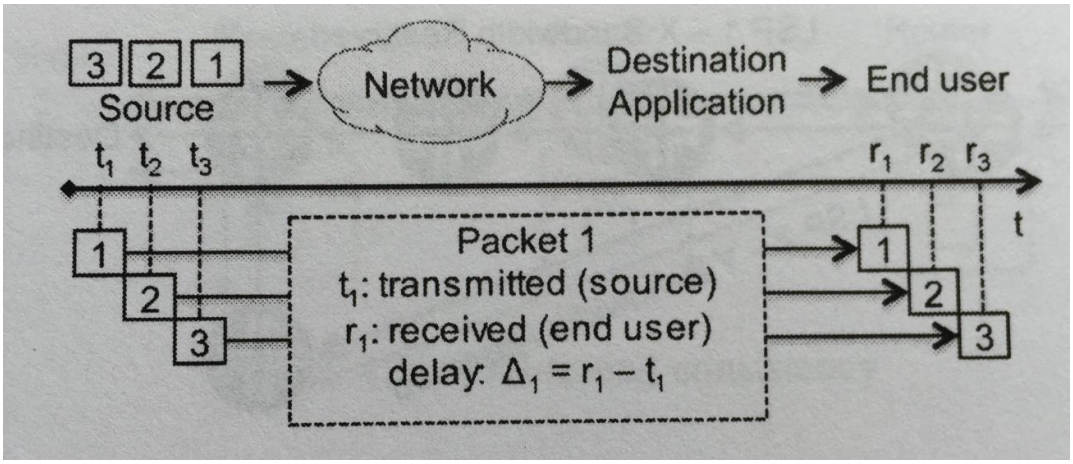

Figura 9 - Exemplo de Latência e Jitter – Fonte: (BARREIROS & Lundqvist, 2011)

A perda de pacotes representa a quantidade de pacotes que é perdida ao longo do tempo, ou seja, o volume de pacotes que foram transmitidos mas não foram recebidos pelo destino.

Conhecidos os três parâmetros, o próximo passo é identificar a sensibilidade do tráfico baseados nesses parâmetros e, neste caso, precisamos dividir em trafego em tempo real e trafego em tempo “não” real.

Para trafego em tempo-real, latência e *jitter* tem um impacto muito grande e precisam ser verificados, pois a ordem na qual os pacotes transmitidos chegam tem influência direta no percepção de qualidade do usuário. Por exemplo um grande delay em uma ligação VoIP. Uma porção de uma frase que já foi 90% recebida pelo destinatário, chegando com grande atraso passa a ser um pacote descartável, pois entraria na ligação totalmente foram de contexto e poderia até atrapalhar outras frases da áudio em curso. Também atrasos no envio de pacotes de vídeo sendo transmitidos em *streaming* causaria um impacto significativo na percepção do cliente, com travamentos e perda de quadros.

Referente a um valor de *jitter* muito alto o impacto não só é percebido na qualidade da áudio como também na aplicação utilizada para a comunicação VoIP em questão, forçando essa aplicação a ter que se adaptar constantemente

a novos valores de delay e impactando em possíveis Buffer de uma transmissão *streaming* de vídeo.

Conforme exemplificado acima, apesar de termos dividido tempo-real em um só grupo, é preciso que seja analisado cada característica específica dos dados, pois casos como VoIP que não são transmissões unidirecionais exigem um tipo de cuidado diferente de uma execução de vídeo por um usuário o qual temos como recurso o *buffer* para mitigar possíveis problemas de latência e *jitter*.

A perda de pacote influencia em dados transmitidos em tempo-real quando é muito grande e constante, o que não entra como uma solução a ser resolvida por QoS, visto que neste caso uma maior largura de banda precisa ser contratada pelo cliente, em casos menores somente algumas perdas de pixels na transmissão de vídeo e ruídos nas comunicações de voz são percebidas, visto isso a perda de pacote seria o item menos importante para transmissões em tempo real.

No caso de transmissão em tempo “não” real, de aplicações como e-mail, a latência e *jitter* tem um impacto muito pequeno, pois n] ao existe uma correspondência direta entre quando o pacote será entregue e sim quando ele será utilizado pelo destinatário. No entanto neste caso a perda de pacotes seria a maior preocupação, não pela percepção do cliente, mas pela integridade dos dados transmitidos. Esse problema é resolvido por protocolos de outra camada que seria o TCP fazendo o controle de pacotes perdidos e solicitando a retransmissão e no caso de conexões UDP a camada de sessão deverá resolver o problema de perda de pacotes.

Visto quais são as informações e necessidades que os engenheiros de rede precisam estar atentos ao aplicar QoS em suas redes, vamos discutir nos próximos tópicos os tipos de serviços são usados para aplicação prática de QoS em roteadores.

2.5.1. Padrões de QoS e *Per-Hop Behavior* (PHB)

Os dois principais padrões de QoS definidos pelo IETF são IntServ (*Integrated Services* ou Serviços Integrados) e o DiffServ (*Differentiated Services* ou Serviços Diferenciados). Estes foram definidos pela RFC1633 e RFC2475 respectivamente.

Além desses temos o Tipo de Serviço ou ToS definido na RFC791 que utiliza o campo *Type of Service* presente no cabeçalho IPv4. Esse campo possui 08 bits aonde os três primeiros bits são responsáveis por identificar o tipo de prioridade que o pacote deve ter na rede e os bits 3,4 e 5 representam necessidades de Latência (*Delay*), Rendimento (*Troughput*) e Confiabilidade

(*Reliability*). Os últimos bits 6 e 7 não são utilizados. Abaixo temos o significado das combinações dos 3 primeiros bits do campo ToS:

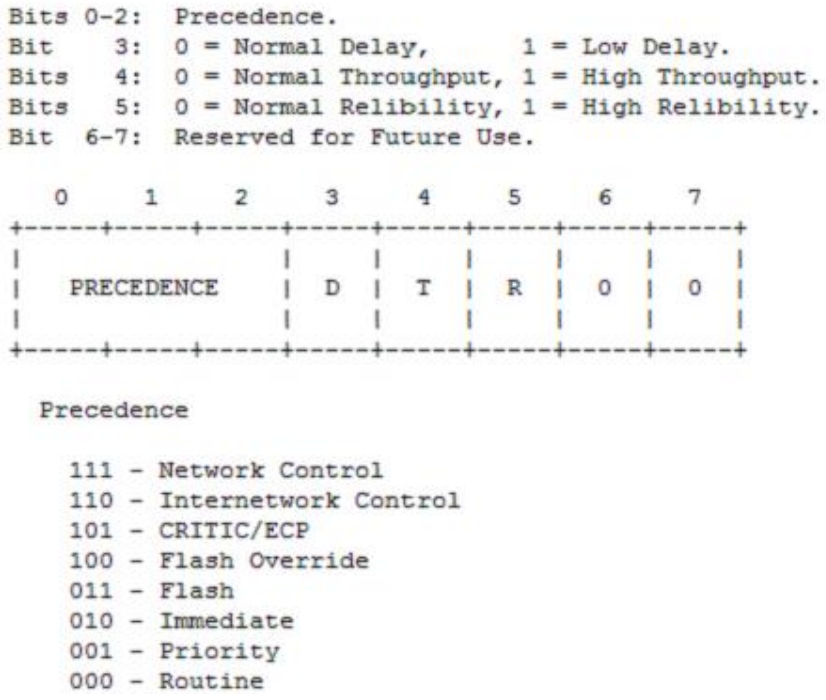

Figura 10 - Detalhamento do campo Type of service (ToS) – Fonte: IETF – RFC791

2.5.2. Serviços Integrados (IntServ)

IntServ foi definido entre 1995 e 1997, através das RFC's 2205 a 2210, esse modelo de arquitetura de fluxo. Ele funciona estabelecendo uma canal dedicado entre o Transmissor e o Receptor, seja em uma transmissão direta ou uma multitransmissão, aonde vários hosts se conectam ao transmissor. Dessa forma, através do IntServ essas transmissões teriam largura de banda, delay e jittter controlados e garantidos. No entanto essa modalidade de QoS não seria viável em redes maiores pois não tem a escalabilidade necessária para atender à crescente demanda atual e devido a sua complexidade e consumo de processamento.

Para que fosse possível estabelecer essa comunicação proposta pelo IntServ, foi necessário também definir um protocolo de reserva de banda ou *Resource Reservation Protocol (RSVP).*

O RSVP é responsável pela sinalização entre o transmissor e o receptor para estabelecimento do canal dedicado de comunicação, ele somente faz

a reserva de canal, a comunicação em si é feita por outros protocolos. Além disso o RSVP permite que sejam conectados vários transmissores a vários receptores e estes podem ser alterados, além disso otimiza a banda ao mesmo tempo que elimina o congestionamento.

Conforme descrito por (TANENBAUM, 2011): “Em sua forma mais simples, o protocolo utiliza roteamento por multidifusão com árvores de amplitude, como discutimos anteriormente. Cada grupo recebe um endereço de grupo. Para transmitir dados a um grupo, um transmissor coloca o endereço desse grupo em seus pacotes. Em seguida, o algoritmo de roteamento por multidifusão padrão constrói uma árvore de amplitude que cobre todos os membros”. Então com esse canal construído a rota entre origem e destino fica registrada nos roteadores, caso o host decida mudar de é possível, desde que ele tenha solicitado mais de uma origem na mensagem inicial de RSVP.

Mesmo o IntServ garantindo uma banda dedicada e dando condições de atender várias aplicações em redes menores esse método não foi muito utilizado, isso por causa da sua complexidade e também pela sua limitação de escalabilidade e de consumo dinâmico de recursos.

2.5.3. Serviços Diferenciados

O modelo DiffServ é baseado no comportamento “nó-a-nó” da rede, aonde o QoS é aplicado através de classes de serviços identificados nos pacotes. Essas classificações ficam inseridas no campo ToS do IPv4 e no campo Traffic Class do IPv6. Esse campo possui 08 bits mas somente os 06 primeiros são utilizados para identificação do tipo de trafego, esses seis bits são chamados de DSCP (*Differentiated Services Code Point*).

Além da classificação dos pacotes, no *DiffServ* os roteadores devem ser separados e mapeados em Domínios DiffsServ (DS). Isso significa que eles possuem as mesmas regras de encaminhamento de pacotes e as mesmas políticas de serviços. Caso um roteador não faça parte desse DS e não siga as regras dos outros roteadores os pacotes não terão o devido tratamento de QoS. Portanto para o correto e eficiente funcionamento do DiffServ a rede deve ser mapeada pelo Gerente de Rede e todos os roteadores devem estar alinhados com as políticas e prioridades necessárias para o bom funcionamento da rede com vários serviços, priorizando os que quais impactam na atividade dos clientes.

Esse modo de trabalho dos rotedores é chamado de *Peer-Hop-Behavior* (PHB) e as classificações do campo DSCP são cruzadas por cada roteador as suas regras de QoS para que o pacote tenha o correto tratamento, outra vantagem do PHB é que não existe mais sinalização entre vizinhos para políticas de QoS, o que reduz o processamento dos roteadores em relação ao IntServ.

O PHB pode ser divididos em 03 categorias PHB padrão, PHB-EF e PHB-AF.

O PHB padrão é comparável ao melhor esforço, pois pacotes nessa categoria serão transmitidos sempre que possível, mas em caso de falta de banda para transmissão, ou filas de encaminhamento de pacotes, estes serão os primeiros pacotes a serem descartados.

O PHB-EF aonde EF é *Expedited Fowarding*, é definido pela RFC2598 como uma ferramenta de QoS fim-a-fim para os pacotes marcados nessa categoria, pois estes terão baixa perda, baixo *jitter*, baixa latência e uma banda especificada. Caso essa banda seja excedida, o pacote pode ser descartado ou atrasado para posterior encaminhamento pelo roteador. Além do descarte, a alocação de banda em excesso pode prejudicar o restante da rede, por isso para aplicação do PHB-EF o mapeamento da banda necessária a cada serviço incluído nele deve ser cuidadosamente avaliado. Segundo definição feita pelo IETF esse serviço é chamado de uma serviço “*Premium”* tem o DSCP de 101110 o que corresponde a 46.

O PHB-AF diferente do PHB-EF o *Assured Forwarding* (AF) tem maior granularidade para a seleção de facilidades que o pacote precisa, fornecendo uma expectativa quanto ao serviço e não uma garantia restritiva, para tal funcionamento no PHB-AF, utiliza-se os campos do DSCP para criar classes de pacotes e também para segmentar essas classes em probabilidade de descarte. Os campos DS5, DS4 e DS3 são os bits mais significativos e são eles quem definem a classe do pacote, os campos DS2 e DS1 especificam a probabilidade de descarte do pacote. O campo DS0 neste caso sempre será setado em “0”.

DiffServ Field

DS5	DS4	DS3	DS2	DS1	DS0	ECN	ECN

- DSCP—six bits (DS5-DS0)
- ECN—two bits

Figura 11 - Detalhamento do Campo DifServ - Fonte: CISCO - QoS Packet Marking - Implementing Quality of Service Policies with DSCP, 2008

Abaixo temos uma tabela aonde são demonstrada as 4 classes que existem no PHB-AF e também os níveis de probabilidade de descarte:

Drop	Class 1	Class 2	Class 3	Class 4
Low	001010 AF11 DSCP 10	010010 AF21 DSCP 18	011010 AF31 DSCP 26	100010 AF41 DSCP 34
Medium	001100 AF12 DSCP 12	010100 AF 22 DSCP 20	011100 AF32 DSCP 28	100100 AF42 DSCP 36
High	001110 AF13 DSCP 14	010110 AF23 DSCP 22	011110 AF33 DSCP 30	100110 AF43 DSCP 38

Figura 12 - Tabela de Classes PHB-AF

Fonte: CISCO - QoS Packet Marking - Implementing Quality of Service Policies with DSCP, 2008

Nos valores DSCP apresentados acima, um pacote da classe AF11 será um pacote com maior prioridade e menor probabilidade de descarte e o pacote da classe AF43 é o pacote de menor prioridade e maior probabilidade de descarte.

Portanto o PHB-AF assegura somente uma grande probabilidade de entrega do pacote, assumindo que pequenos congestionamentos ou poucos pacotes de classes prioritárias podem ser descartados. Neste caso, um cliente que queira enviar um pacote acima da sua banda contratada poderá fazê-lo, mas tem consciência que este pacote pode ser descartado.

Em resumo, conforme descrito por (LIMA, 2001): “O PHB-EF realiza uma alocação explícita de recursos para a sua agregação de fluxos e, por isso, deve ter um maior custo final para o cliente. O PHB-AF oferece garantias estatísticas de banda passante e seus mecanismos de gerenciamento ativo de filas visam controlar fluxos adaptativos. Este serviço pode ser oferecido com um custo menor aos clientes”. Levando em consideração essas especificidades as técnicas de aplicação de QoS podem ser utilizadas de forma separada e também de forma conjunta e a determinação das técnicas, garantias e tipos de serviços

a serem priorizados, devem ser definidas entre a operadora e o cliente e ser registrada como um SLA (Service Level Agreement).

3. ESTUDO DE CAMPO

Esse capítulo apresentará uma simulação de trafego em uma rede contendo pacotes diversos para análise e comparação do desempenho entre a rede IPv6 pura configurada com QoS e sem QoS. Com isso poderemos comprovar a eficiência das técnicas de QoS aplicadas em um cenário atual que é o de uma rede IPv6.

Para simulação deste projeto iremos utilizar o GNS3 com que é um software de emulação de roteadores CISCO gratuito. Neste software as configurações dos roteadores é semelhante a uma configuração real, porém os resultados tem uma pequena diferença com a realidade, pois todos os componentes e cabos utilizados no simuladores são considerados como “ideais”, o que sabemos não acontecer na prática.

Também será apresentado nesse capítulo a configuração dos roteadores no GNS3, para estabelecermos a rede IPv6 e também para a aplicação das políticas de marcação das redes e Vlan’s que vão separar os tráfegos.

Com a rede configurada em IPv6 vamos limitar o link de transmissão e sequencialmente saturá-lo, com eles saturado vamos medir o tempo de resposta de uma conexão Telnet aos roteadores da rede e também de pacotes ICMP.

Após essa primeira análise, será aplicado o QoS, e novamente vamos avaliar os pacotes ICMP, Telnet e também o comportamento da comunicação entre as redes de VoIP (simulação de uma comunicação UDP pela porta 5000).

Para saturação do link iremos utilizar o software livre JPerf e virtualizaremos 04 máquinas, sendo 02 com Windows XP e 02 com Linux Ubuntu.A virtualização será feita através do Virtual Box da Oracle, que também pode ser instalado de forma gratuita. Esses dois conjuntos de sistemas operacionais irão trabalhar como servidor/cliente pois é esse o modelo de funcionamento do Jperf que será nossa ferramenta para gerar o tráfego desse estudo e também para acompanhamento gráfico dos resultados.

A topologia a ser montada e estudada é apresentada na figura abaixo:

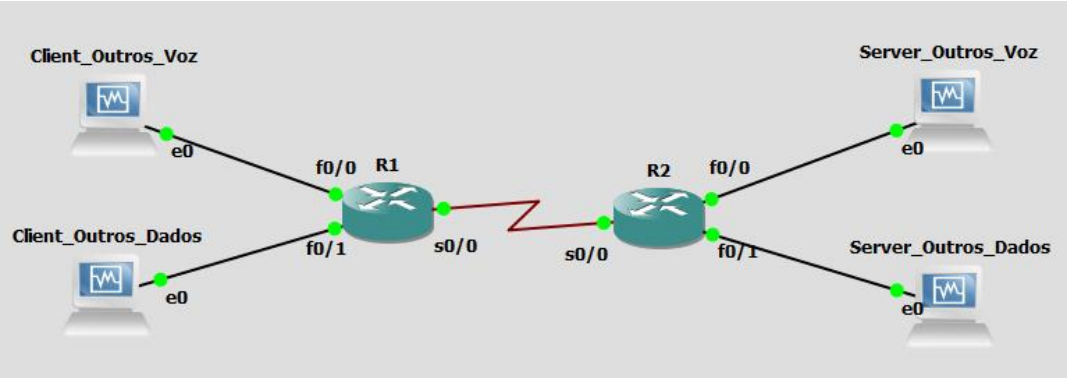

Figura 13 - Topologia de Estudo

Fonte – Autoria própria

3.1. CONFIGURANDO AS MÁQUINAS VIRTUAIS

Inicialmente foram configuradas 02 máquinas virtuais, uma com o sistema operacional Ubuntu e uma com o sistema operacional Windows XP.

Após a configuração das máquinas, cada uma foi clonada no próprio Virtual Box da Oracle para que conseguíssemos estabelecer dois conjuntos cliente/servidor. Finalizando a criação e clonagem o Virtual Box apresentava as quatro máquinas conforme a figura abaixo:

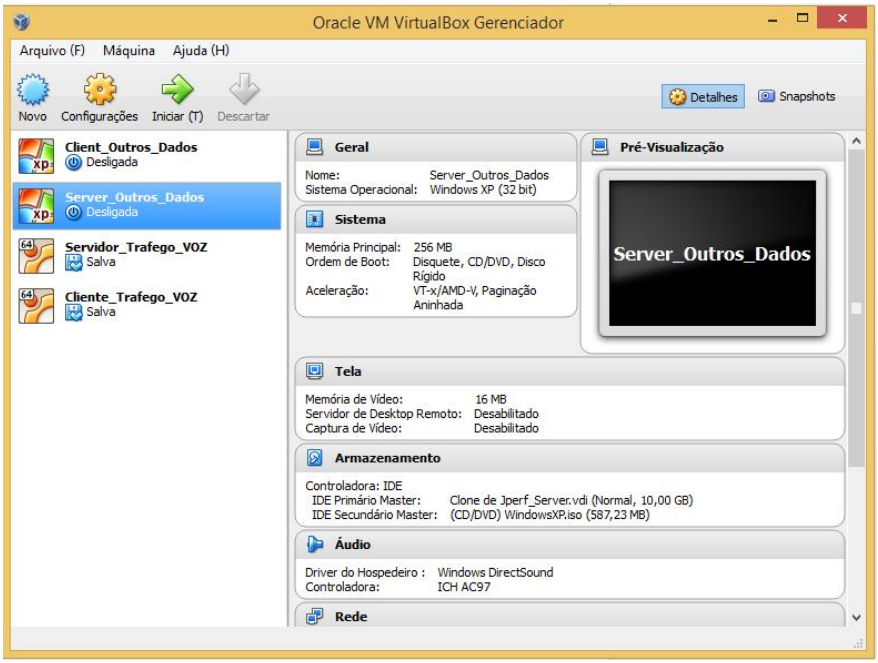

Figura 14 - Virtual Box

Fonte: Autoria Própria

Após a criação das máquinas iremos configurar o Windows XP e o Ubuntu para integração com o GNS3 e também com a Rede IPv6.

No Windows XP devemos primeiramente instalar o pacote IPv6, que não é nativo a esse sistema operacional. Para isso devemos abrir o Prompt de Comando e executar o comando **ipv6 install**, após esse comando teremos a seguinte confirmação:

Figura 15 - Comando ipv6 install - Windows XP

Fonte: Autoria própria

Após a confirmação de sucesso na instalação do pacote ipv6, devemos identificar qual interface está nossa conexão local, isso deve ser feito através do comando **ipv6 if**. Sabendo qual é a interface, devemos utilizar o comando **ipv6 adu [InterfaceIndex]/[Address]** para definir o endereço IPv6 fixo para máquina, caso não seja necessário estabelecer um endereço fixo, somente será necessário habilitar o IPv6 que será atribuído o endereço IP automaticamente.

Abaixo temos a confirmação do endereço IPv6 configurado na máquina virtual deste trabalho:

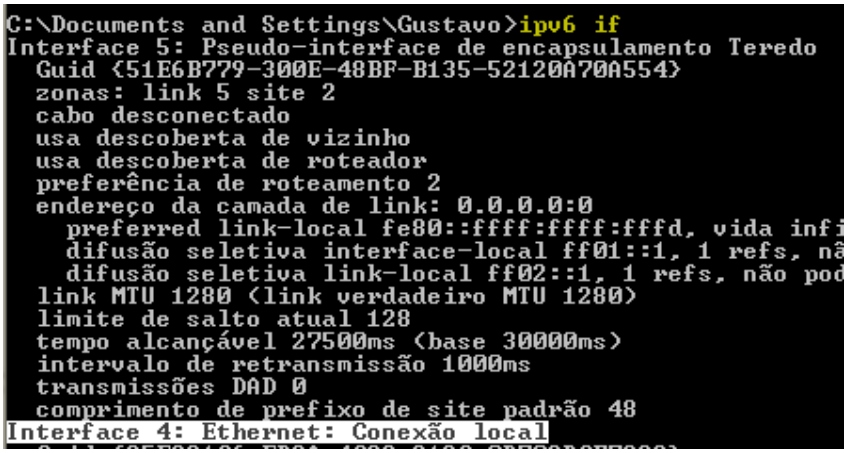

Figura 16 - Identificação de Interface - Windows XP

Fonte: Autoria própria

```
C:\Documents and Settings\Gustavo>ipv6 adu 4/2001:db8:cafe:2::44

C:\Documents and Settings\Gustavo>ipconfig

Configuração de IP do Windows


Adaptador Ethernet Conexão local:

        Sufixo DNS específico de conexão  . :
        Endereço IP de config. automática . : 169.254.56.218
        Máscara de sub-rede . . . . . . . . : 255.255.0.0
        Endereço IP . . . . . . . . . . . . : 2001:db8:cafe:2::44
        Endereço IP . . . . . . . . . . . . : fe80::a00:27ff:fe4d:d3bb%4
        Gateway padrão. . . . . . . . . . . :

Adaptador de túnel Teredo Tunneling Pseudo-Interface:

        Sufixo DNS específico de conexão  . :
        Endereço IP . . . . . . . . . . . . : fe80::ffff:ffff:fffd%5
        Gateway padrão. . . . . . . . . . . :

Adaptador de túnel Automatic Tunneling Pseudo-Interface:

        Sufixo DNS específico de conexão  . :
        Endereço IP . . . . . . . . . . . . : fe80::5efe:169.254.56.218%2
        Gateway padrão. . . . . . . . . . . :
```

Figura 17- Configuração de IPv6 Fixo - Windows XP

Fonte: Autoria própria

No Ubuntu a configuração do endereçamento IPv6 pode ser feito através de sua interface gráfica, para tal devemos clicar no símbolo de conexão e posteriormente em “Editar Conexões...”. Assim será aberto as conexões de rede existe e neste ponto deveremos clicar para realizar a configuração da nova conexão. Abaixo temos algumas imagens das configurações realizadas:

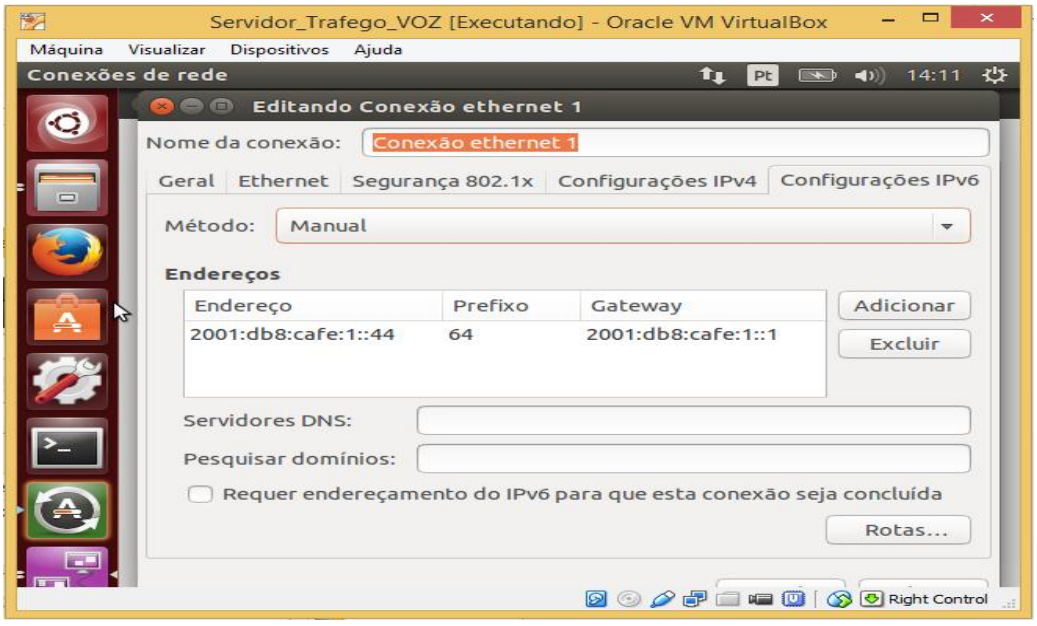

Figura 18 - Configuração de Interface IPv6 – Ubuntu

Fonte: Autoria própria

O mesmo processo foi realizado nas outras duas máquinas virtuais.

Com as máquinas configuradas com os endereços IP abaixo, realizamos a instalação do Jperf.

MAPEAMENTO DAS INTERFACES DOS ELEMENTOS			
ELEMENTO	***Rede***	***Interface***	***Endereço IP***
Dados Cliente	2001:db8:café:3:: /64	fa0/1 - RT_0	2001:db8:cafe:3::44
Dados Servidor	2001:db8:café:4:: /64	fa0/1 - RT_1	2001:db8:cafe:4::44
VOZ Servidor	2001:db8:café:2:: /64	fa0/0 - RT_1	2001:db8:café:2::44
VOZ Cliente	2001:db8:café:1:: /64	fa0/1 - RT_0	2001:db8:café:1::44

Figura 19 - Tabela de endereço de Host - Máquinas Virtuais

Fonte: Autoria própria

No WindowsXP conseguimos rodar direto do executável do Jperf, mas no Ubuntu foi necessário instalar além do Jperf e o Iperf, que é o mesmo software sem interface gráfica, executar o comando "***sudo chmod u+x jperf.sh"*** para criar a permissão de execução do arquivo. Criada essa permissão executamos o Jperf no Ubuntu com o comando "***./jperf.sh"***.

3.2. CONFIGURAÇÃO DO GNS3

Para configuração do GNS3, primeiro foi inserido o IOS de um Router CISCO 3725 para configuração da rede IPv6 e posteriormente das políticas de QoS. Também será necessário configurar as máquinas virtuais para integração com a rede emulada.

A configuração das máquinas virtuais, na versão 1.1 do GNS3 já é nativa e muito simples de ser realizada, basta entrar em *Edit>Preferences>VirtualBox>VirtualBox VMs*. Após isso é só clicar em New, que as máquinas virtuais criadas estarão disponíveis.

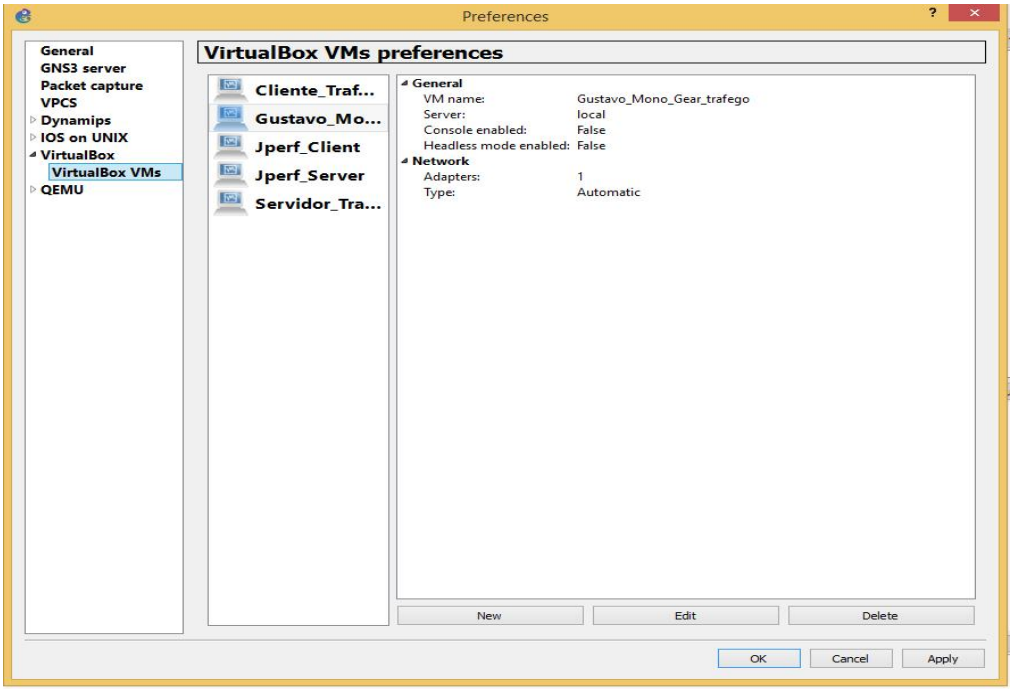

Figura 20 - Configuração de Máquinas Virtuais - GNS3 1.1

Fonte: Autoria própria

Após configuradas as máquinas, estas estarão disponíveis para serem inseridas na topologia. Toda configuração de rede nestes elementos será feita na própria máquina virtual.

A configuração do roteador, como temos no GNS3 uma simulação fidedigna a realidade é feita através do console. Inicialmente foi configurado somente uma rede IPv6 simples, com DHCPv6, acesso Telnet ao console de comandos e também senhas de segurança para configuração dos roteadores.

Os comandos utilizados no Roteador RT_0 e RT_1 são demonstrados nas imagens abaixo:

Description	Command Line
Router>	enable
Router	conf term
Router(config)	hostname g_RT_0
g_RT_0(config)	banner motd 'Somente acesso autorizado!!!'
g_RT_0(config)	enable secret class
g_RT_0(config)	service password-encryption
g_RT_0(config)	line console 0
g_RT_0(config-line)	password cisco
g_RT_0(config-line)	login
g_RT_0(config-line)	line vty 0 4
g_RT_0(config-line)	password cisco
g_RT_0(config-line)	login
g_RT_0(config-line)	exit
g_RT_0(config)	int s0/0
g_RT_0(config-if)	ipv6 address 2001:db8:cafe:ffff::0/127
g_RT_0(config-if)	no shutdown
g_RT_0(config-if)	exit
g_RT_0(config)	int fa0/0
g_RT_0(config-if)	ipv6 address 2001:db8:cafe:1::1/64
g_RT_0(config-if)	no shutdown
g_RT_0(config-if)	exit

Figura 21 - Script de Configuração Incial RT_0

Fonte: Autoria Própria

Description	Command Line
Router>	enable
Router	conf term
Router(config)	hostname g_RT_1
g_RT_1(config)	banner motd 'Somente acesso autorizado!!!'
g_RT_1(config)	enable secret class
g_RT_1(config)	service password-encryption
g_RT_1(config)	line console 0
g_RT_1(config-line)	password cisco
g_RT_1(config-line)	login
g_RT_1(config-line)	line vty 0 4
g_RT_1(config-line)	password cisco
g_RT_1(config-line)	login
g_RT_1(config-line)	exit
g_RT_1(config)	int s0/0
g_RT_1(config-if)	ipv6 address 2001:db8:cafe:ffff::1/127
g_RT_1(config-if)	no shutdown
g_RT_1(config-if)	exit
g_RT_1(config)	int fa0/0
g_RT_1(config-if)	ipv6 address 2001:db8:cafe:2::1/64
g_RT_1(config-if)	no shutdown
g_RT_1(config-if)	exit

Figura 22 - Script de Configuração Inicial RT_1

Fonte: Autoria própria

Após rodarmos os scripts nos roteadores, estes foram configurados para anunciarem suas rotas através da regra *Open Short Path First* ou simplesmente OSPFv3 que é especificamente este protocolo para IPv6. Como neste trabalho temos somente dois roteadores, utilizamos somente uma área de OSPF, a área 0, aonde estes dois roteadores trocam informações de estados dos seus links e também de suas rotas.

Os comandos utilizados nos roteadores RT_0 e RT_1 são os abaixo descritos:

Description	Command Line	Description	Command Line
g_RT_0(config)	ipv6 unicast-routing	g_RT_1(config)	ipv6 unicast-routing
g_RT_0(config)	ipv6 router ospf 1	g_RT_1(config)	ipv6 router ospf 1
g_RT_0(config-rtr)	router-id 1.1.1.1	g_RT_1(config-rtr)	router-id 2.2.2.2
g_RT_0(config-rtr)	int s0/0	g_RT_1(config-rtr)	int s0/0
g_RT_0(config-if)	ipv6 ospf 1 area 0	g_RT_1(config-if)	ipv6 ospf 1 area 0
g_RT_0(config-rtr)	int fa0/0	g_RT_1(config-if)	int fa0/0
g_RT_0(config-if)	ipv6 ospf 1 area 0	g_RT_1(config-if)	ipv6 ospf 1 area 0
g_RT_0(config-rtr)	int fa0/1	g_RT_1(config-if)	int fa0/1
g_RT_0(config-if)	ipv6 ospf 1 area 0	g_RT_1(config-if)	ipv6 ospf 1 area 0
g_RT_0(config-if)	exit	g_RT_1(config-if)	exit

Figura 23 - Script de Configuração OSPFv3 - RT_0 e RT_1

Fonte: Autoria própria

Seguindo os passos acima descritos já é possível realizar teste de comunicação entre os hosts da rede, máquinas virtuais e também entre os roteadores.

Para aplicação e configuração de QoS o primeiro passo foi dividir as redes de Dados e Voz, que são representadas pelas máquinas virtuais Cliente/Servidor do Virtual Box que foram integradas com o GNS3, em Vlan's. Com isso podemos classificar e aplicar o QoS específico para cada rede. Também foi criado uma regra para acesso Telnet através do QoS aplicado por uma *Access Control List* (ACL) para porta TCP 23.

Devido a uma limitação do GNS3 a restrição de banda do link entre os roteadores não pode ser feitas pelos comando Bandwith e Clock Rate direto nos roteadores RT_0 e RT_1, com isso uma técnica de QoS chamada Policy foi aplicada nas interfaces de entrada e saída dos roteadores, os comandos utilizados foram:

Description	Command Line	Description	Command Line
g_RT_0(config)	policy-map 500Kbps	g_RT_1(config)	policy-map 500Kbps
g_RT_0(config-pmap)	class class-default	g_RT_1(config)	class class-default
g_RT_0(config-pmap-c)	police cir 512000 16000	g_RT_1(config-rtr)	police cir 512000 16000
g_RT_0(config-pmap-c-police)	conform-action transmit	g_RT_1(config-rtr)	conform-action transmit
g_RT_0(config-pmap-c-police)	exceed-action drop	g_RT_1(config-if)	exceed-action drop
g_RT_0(config-pmap-c-police)	exit	g_RT_1(config-if)	exit
Router	conf term	g_RT_1(config-if)	conf term
g_RT_0(config)	int s0/0	g_RT_1(config-if)	int s0/0
g_RT_0(config-if)	service-policy input 500Kbps	g_RT_1(config-if)	service-policy input 500Kbps
g_RT_0(config-if)	end	g_RT_1(config-if)	end

Figura 24 - Script de configuração de Policy para limitação de banda em 500Kbps

Fonte: Autoria própria

A confirmação de funcionamento dessa política de QoS como restrição de banda, foi feita utilizando o Jperf para congestionar a rede usando 500Kbits de transferência TCP e confirmando o congestionamento com ICMP de 1500bytes em paralelo a um envio de 5Mbits pelo Jperf em uma conexão UDP.

Abaixo temos os prints do cliente e servidor, confirmando a limitação em 500Kbps:

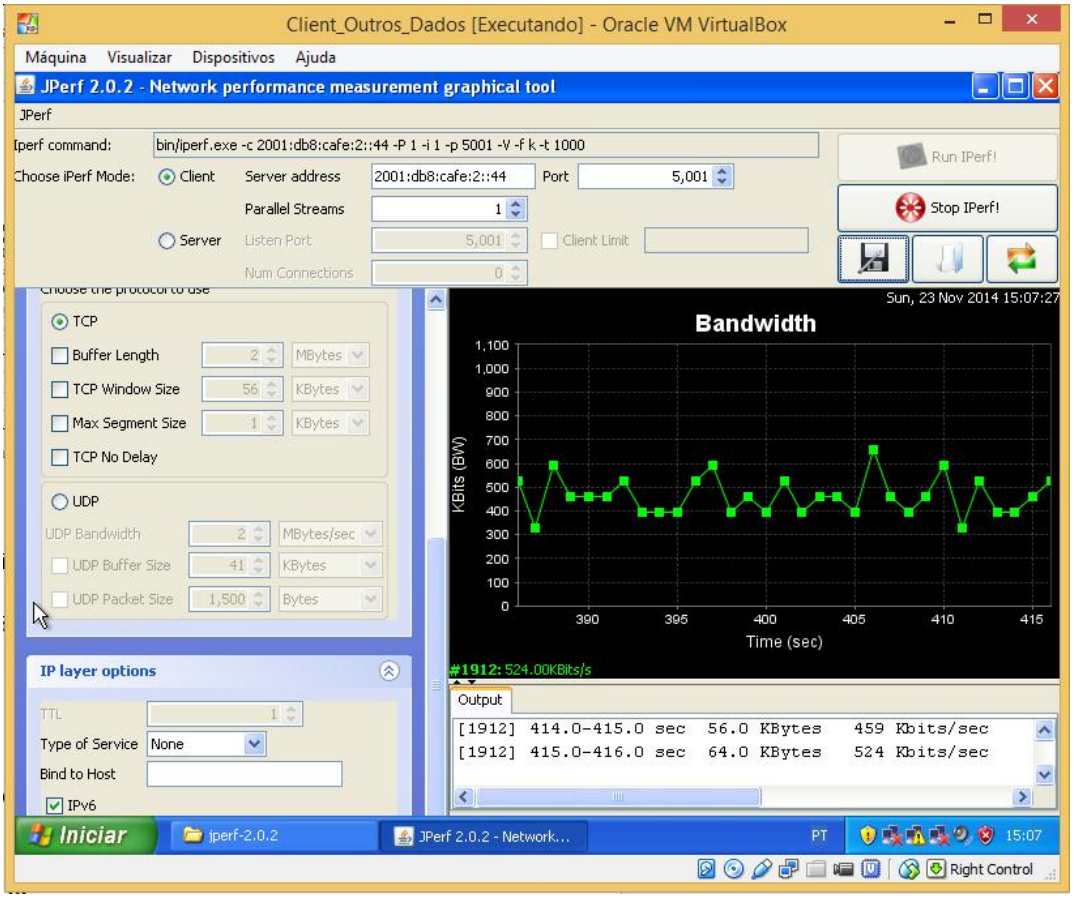

Figura 25- Jperf cliente - Limitação de Banda a 500Kbits

Fonte: Autoria própria

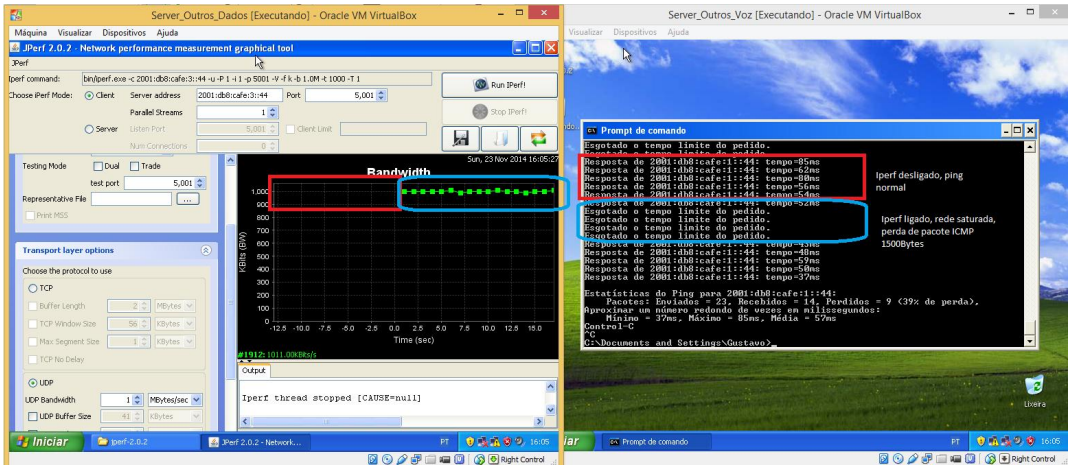

Figura 26 - Teste de Saturação UDP Jperf e ICMP

Fonte: Autoria própria

Confirmada a saturação do link, o próximo passo é demonstrar a aplicação de QoS para garantir serviços essenciais para administração de rede e quaisquer outros serviços que o administrador entenda como prioritário.

Para tal iremos aplicar uma tabela de QoS conforme a imagem abaixo:

PHB	DSCP	Serviço	Policy Map
CS6	56	FTP	ftp
AF43	38	5001	Dados

Figura 27- Tabela de Referência de QoS para Testes

Fonte: Autoria própria

Essa tabela foi elaborada visando garantir a conexão e transferência de dados FTP, mesmo com a rede recebendo um trafego UDP, gerado pela Jperf de 30Mbits. Caso fosse necessário priorizar outros tipos de conexões, teríamos que mapeá-las com o intuito de balancear o tráfego, isso ocorre por que a banda de redes comerciais normalmente são limitadas e sua expansão tem um impacto alto no custo da operação.

3.3. CONFIGURAÇÃO DE QOS – IOS CISCO

Para a configuração de QoS nos roteadores cisco, precisamos inicialmente definir em quais interfaces iremos aplicar cada política de marcação

de pacotes e também de controle de trafego. Para os testes deste trabalho as políticas de marcação foram inseridas nas interfaces de entrada S0/0 de cada roteador. A política de controle de trafego foi aplicada nas interfaces do link entre os roteadores como "*output" nas Seriais 0/0.*

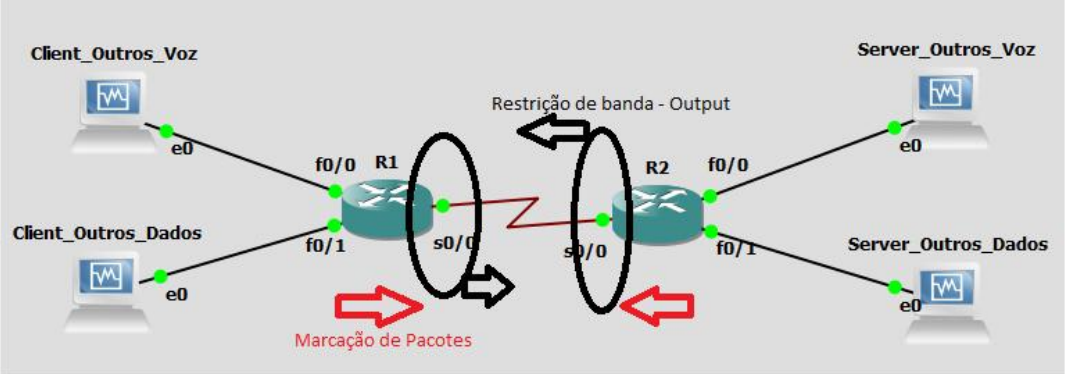

Figura 28 - Locais de Marcação e Policiamento de QoS

Fonte: Autoria própria

Iniciando com a marcação dos pacotes por tipo de trafego, fizemos uma ACL de FTP e uma de UDP, poderíamos ter feito diretamente a marcação através do *match protocol ftp*, mas não seria possível fazer assim para UDP. Outro fato identificado foi que quando marcamos os pacotes através do match protocol ftp, a marcação não estava ocorrendo, por isso partimos para uma alternativa que foi a marcação através de ACL's. Primeiramente criamos a ACL para pacotes ftp:

```
ipv6 access-list FTP
permit tcp any eq ftp any
permit tcp any eq ftp-data any
exit
```

Depois criamos uma ACL para pacotes UDP (utilizado no Jperf para saturar a rede):

```
ipv6 access-list UDP
permit udp any any
exit
```

Com essas duas ACL's criadas, realizamos agora a criação das *class-maps* através dos comandos abaixo:

```
class-map match-all FTP
  match protocol ipv6
  match access-group name FTP
exit
class-map match-all UDP
  match protocol ipv6
  match access-group name UDP
exit
```

Notem que foi criado uma class-map para FTP e uma para UDP, pois cada uma receberá um tipo de marcação diferente nas interfaces de entrada da nossa rede. Essa marcação será feita através de uma Policy-map que nomeamos neste projeto como QOS:

```
policy-map QOS
  class FTP
   set dscp cs6

  class UDP
   set dscp af13

 class class-default
   fair-queue
   set dscp default
```

Também foi acrescentado na política que usamos para restringir a banda em 500Kbps as classes FTP e UDP:

```
policy-map 500Kbps
class UDP
set dscp af13
exit
class FTP
set dscp cs6
exit
```

Para a class FTP o pacote receberá marcação CS6 que é de alta prioridade, pois neste trabalho queremos garantir o trafego de arquivos FTP, mesmo com a rede congestionada e para contra prova não iremos mais colocar a prioridade nos pacotes ICMP, pois estes servirão de confirmação do congestionamento da rede.

Os pacotes UDP receberam o dscp AF13 que significa que eles terão um tratamento de alta probabilidade de descarte de pacotes, garantindo assim a passagem de pacotes marcados na regra FTP. Todos os outros tipos de pacotes

que passarem por essa interface receberão a marcação default de DSCP, sendo sempre descartados quando houver um congestionamento.

Após a criação das classes, políticas e definição de prioridades, aplicamos nas interfaces de RT_0 e RT_1 as políticas de QoS. Para tal, foi utilizado os comandos abaixo nos roteadores:

```
int s0/0
  service-policy input 500Kbps
  service-policy output QOS
```

Com todas essas políticas aplicadas as interfaces dos roteadores iniciamos os testes de QoS para verificar se estava existindo marcação dos pacotes e se tínhamos garantia de banda para o FTP.

3.4. TESTES E RESULTADOS

A primeira etapa de testes foi com a rede sem aplicação de QoS, para tal abrimos 02 conexões entre máquinas virtuais. A primeira para comunicação pelo Jperf com um trafego UDP de 30Mbits, após essa conexão estabelecida, podemos verificar que na segunda conexão, aonde temos o servidor e cliente FTP não é possível realizar o download de arquivos e temos dificuldade de conexão ao servidor.

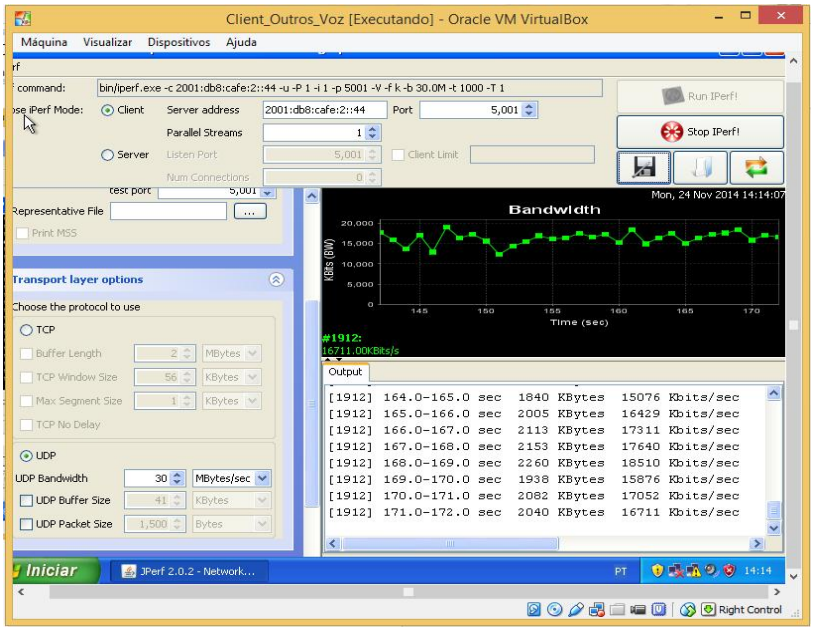

Figura 29 - Jperf Trafego Gerado para congestionamento da Rede

Fonte: Autoria própria

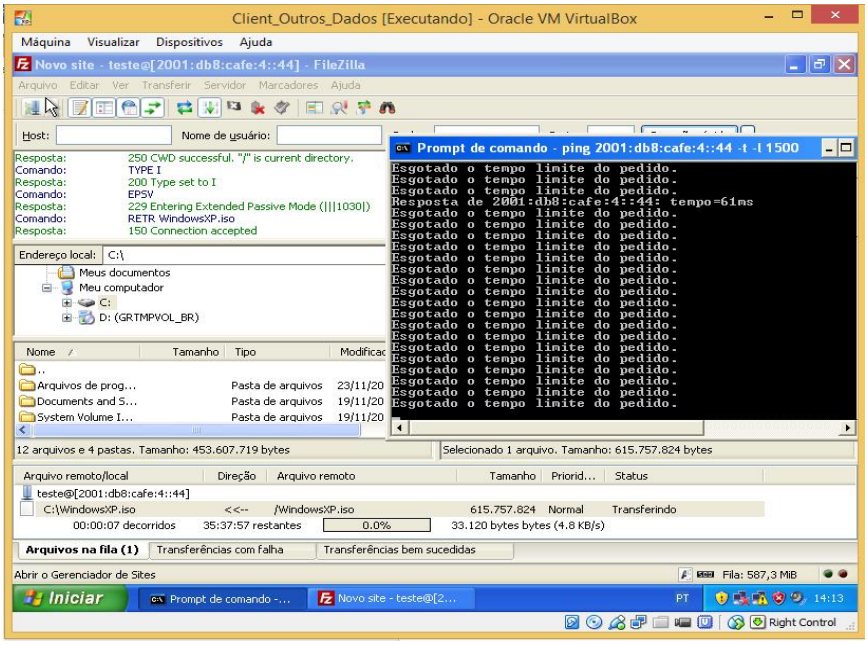

Figura 30 - Download via FTP com baixa velocidade e ICMP com grande perdas

Fonte: Autoria própria

Também é possível identificar nos roteadores o incremento na política de restrição de Banda, aplicada a Serial 0/0:

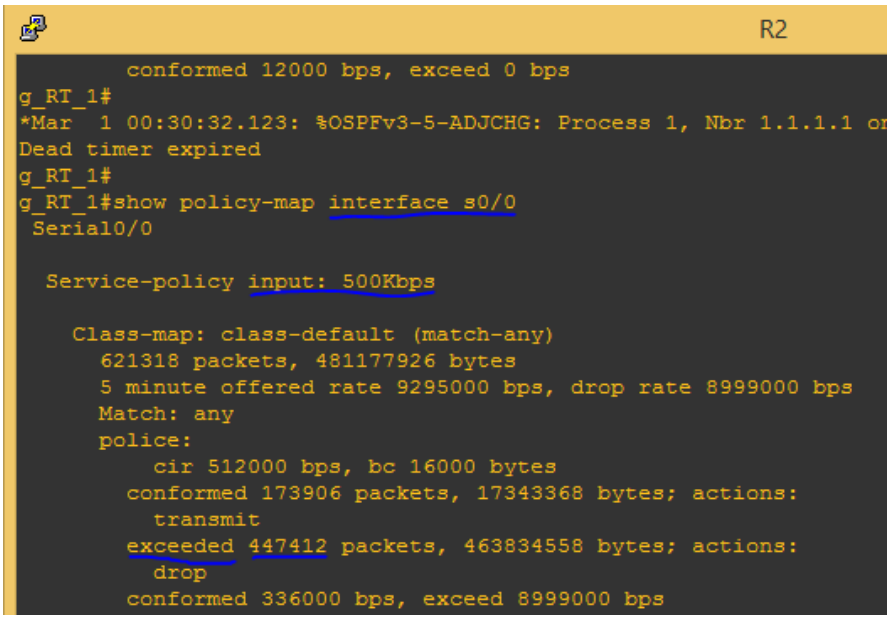

Figura 31- Confirmação de aplicação da política de restrição de Banda RT_1

Fonte: Autoria própria

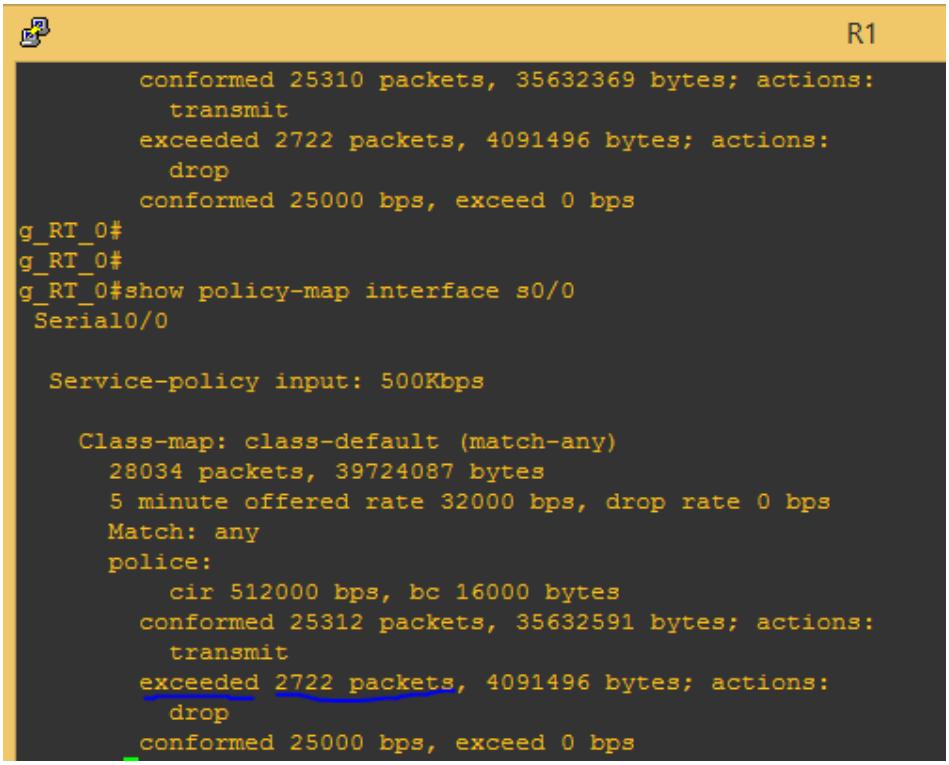

Figura 32 - Confirmação de aplicação da política de restrição de Banda em RT_0

Fonte: Autoria própria

O próximo passo de testes foi a aplicação das políticas de QoS nos roteadores, para isso usamos os scripts abaixo, que é um consolidado do que foi explicado na sessão anterior desse documento:

Description	Command Line	Description	Command Line
g_RT_0(config)	ipv6 access-list FTP	g_RT_1(config)	ipv6 access-list FTP
g_RT_0(config-ipv6-acl)	permit tcp any eq ftp any	g_RT_1(config-ipv6-acl)	permit tcp any eq ftp any
g_RT_0(config-ipv6-acl)	permit tcp any eq ftp-data any	g_RT_1(config-ipv6-acl)	permit tcp any eq ftp-data any
g_RT_0(config-ipv6-acl)	exit	g_RT_1(config-ipv6-acl)	exit
g_RT_0(config)	ipv6 access-list UDP	g_RT_1(config)	ipv6 access-list UDP
g_RT_0(config-ipv6-acl)	permit udp any any	g_RT_1(config-ipv6-acl)	permit udp any any
g_RT_0(config-ipv6-acl)	exit	g_RT_1(config-ipv6-acl)	exit
g_RT_0(config)	class-map match-all FTP	g_RT_1(config)	class-map match-all FTP
g_RT_0(config-cmap)	match protocol ipv6	g_RT_1(config-cmap)	match protocol ipv6
g_RT_0(config-cmap)	match access-group name FTP	g_RT_1(config-cmap)	match access-group name FTP
g_RT_0(config-cmap)	class-map match-all UDP	g_RT_1(config-cmap)	class-map match-all UDP
g_RT_0(config-cmap)	match protocol ipv6	g_RT_1(config-cmap)	match protocol ipv6
g_RT_0(config-cmap)	match access-group name UDP	g_RT_1(config-cmap)	match access-group name UDP
g_RT_0(config-cmap)	exit	g_RT_1(config-cmap)	exit
g_RT_0(config)	policy-map 500Kbps	g_RT_1(config)	policy-map 500Kbps
g_RT_0(config-pmap)	class UDP	g_RT_1(config-pmap)	class UDP
g_RT_0(config-pmap-c)	set dscp af13	g_RT_1(config-pmap-c)	set dscp af13
g_RT_0(config-pmap-c)	exit	g_RT_1(config-pmap-c)	exit
g_RT_0(config-pmap)	class FTP	g_RT_1(config-pmap)	class FTP
g_RT_0(config-pmap-c)	set dscp cs6	g_RT_1(config-pmap-c)	set dscp cs6
g_RT_0(config-pmap-c)	exit	g_RT_1(config-pmap-c)	exit
g_RT_0(config)	policy-map QOS	g_RT_1(config)	policy-map QOS
g_RT_0(config-pmap)	class FTP	g_RT_1(config-pmap)	class FTP
g_RT_0(config-pmap-c)	set dscp cs6	g_RT_1(config-pmap-c)	set dscp cs6
g_RT_0(config-pmap-c)	class UDP	g_RT_1(config-pmap-c)	class UDP
g_RT_0(config-pmap-c)	set dscp af13	g_RT_1(config-pmap-c)	set dscp af13
g_RT_0(config-pmap-c)	class class-default	g_RT_1(config-pmap-c)	class class-default
g_RT_0(config-pmap-c)	fair-queue	g_RT_1(config-pmap-c)	fair-queue
g_RT_0(config-pmap-c)	set dscp default	g_RT_1(config-pmap-c)	set dscp default
g_RT_0(config-pmap-c)	end	g_RT_1(config-pmap-c)	end
Router	conf term	Router	conf term
g_RT_0(config)	int s0/0	g_RT_1(config)	int s0/0
g_RT_0(config-if)	service-policy output QOS	g_RT_1(config-if)	service-policy output QOS
g_RT_0(config-if)	end	g_RT_1(config-if)	end

Figura 33 - Script de Aplicação QoS para FTP e UDP

Fonte: Autoria própria

Após executarmos o script nos roteadores RT_0 e RT_1 as configurações foram confirmadas com os comandos "***show policy-map interface s0/0***":

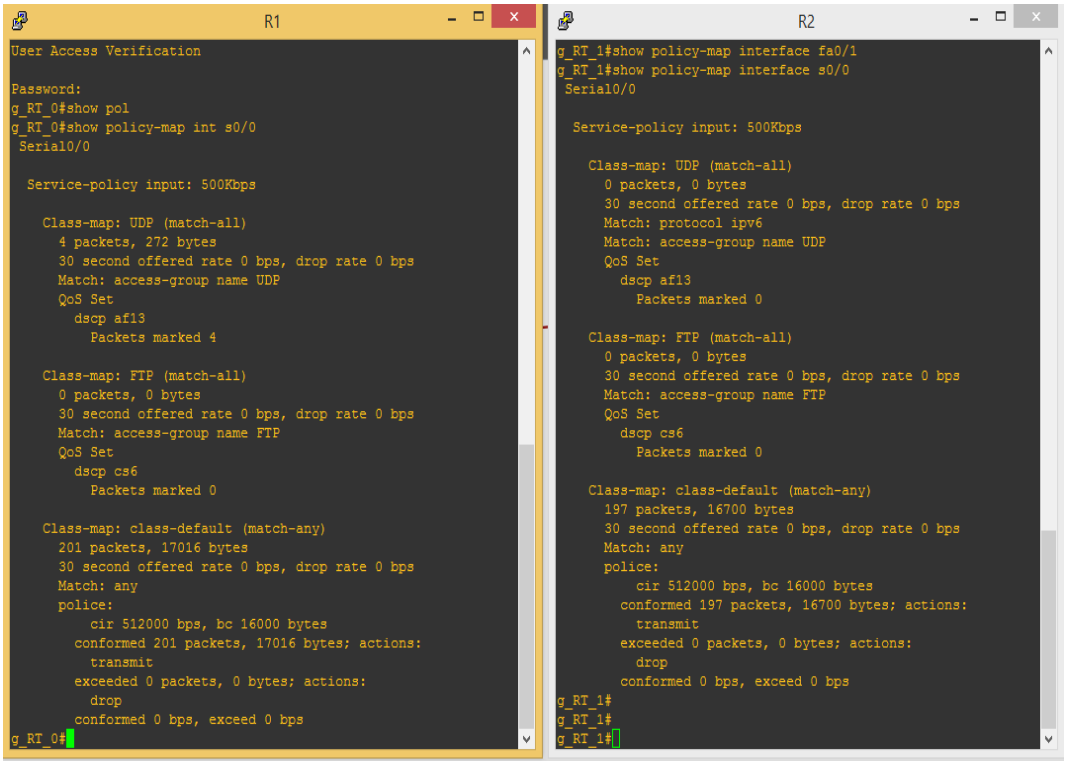

Figura 34 - Resultado do comando show policy-map s0/0

Fonte: Autoria própria

Também podemos ver que agora temos poucas marcações nas novas políticas de QoS, para verificar e validar a efetividade da solução de QoS escolhida, iniciamos novamente as máquinas virtuais e fizemos dois fluxos de dados. Novamente o primeiro com o Jperf, trafego UDP de 30Mbits/s e posteriormente com o servidor e cliente de FTP.

Nessa nova situação da rede, podemos identificar que os pacotes UDP e FTP estavam sendo marcados conforme figura abaixo:

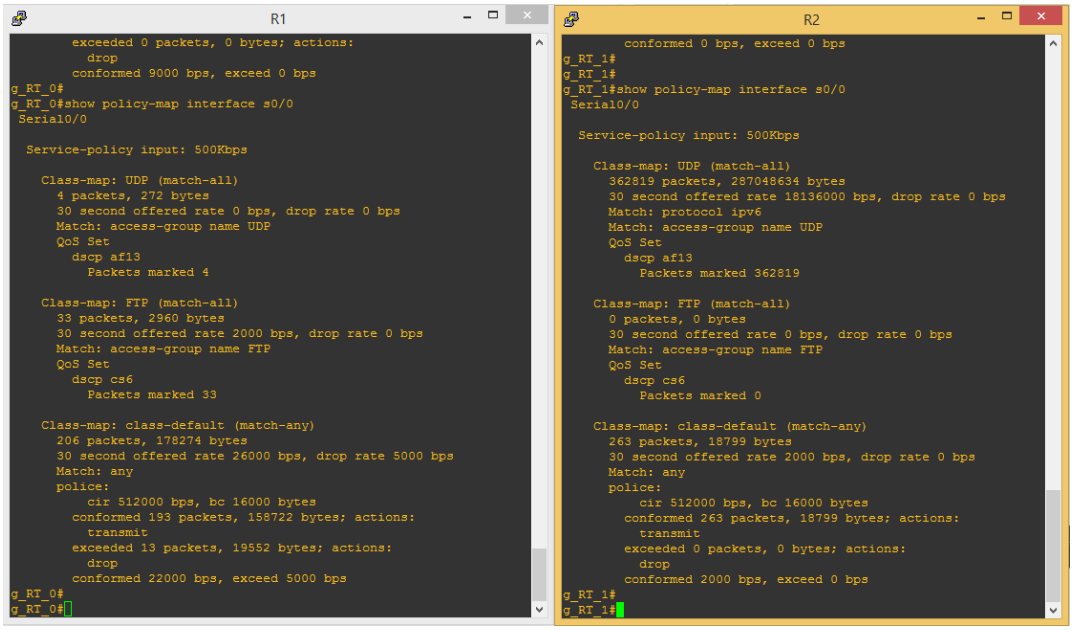

Figura 35 - Marcação de Pacotes na nova política de limitação incrementada das marcações de QoS

Fonte: Autoria própria

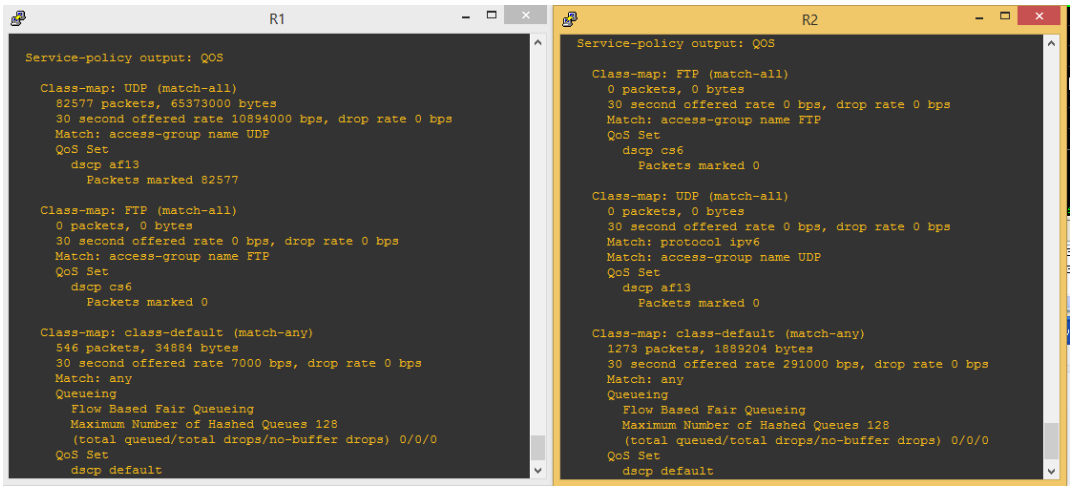

Figura 36 - Marcação de Pacotes pela nova política de QoS nas saídas das Seriais dos roteadores RT_0 e RT_1

Fonte: Autoria própria

É importante observar que a marcação dos pacotes FTP aparecem no RT_0 e a de UDP no RT_1 mas em ambos continuamos tendo a marcação de pacotes na política de restrição de banda.

Também foi possível confirmar o sucesso na aplicação do QoS nas máquinas virtuais, aonde agora, com a nova política de QoS, podemos identificar que mesmo com a saturação de trafego através do Jperf, o download de arquivos via FTP mantem-se normalmente na taxa de 39KBits/s:

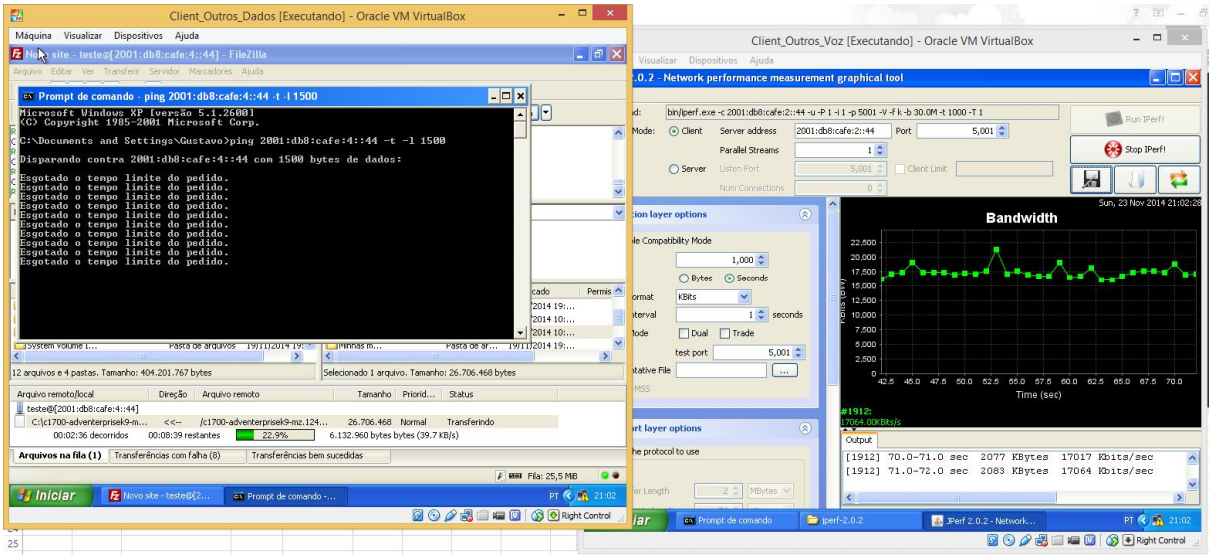

Figura 37 - Comparação das máquinas virtuais e tráfegos FTP, Jperf_UDP e ICMP

Fonte: Autoria própria

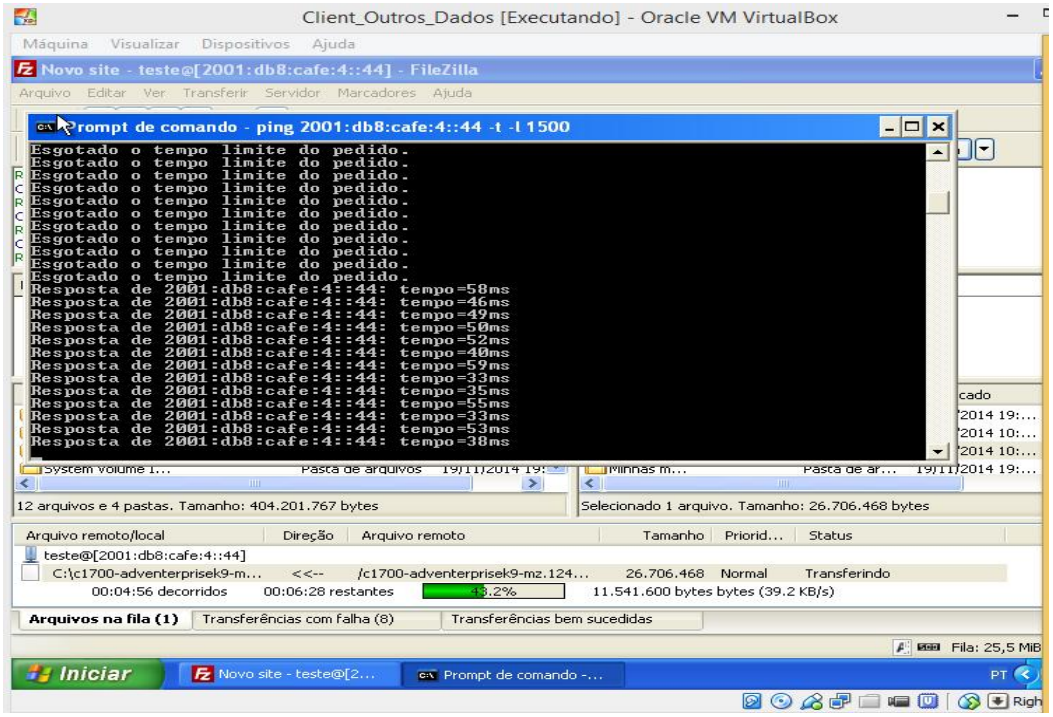

Figura 38 - Transferência FTP com velocidade constante em ambiente congestionado

Fonte: Autoria própria

4. CONSIDERAÇÕES FINAIS

Devido as mudança que o mundo de redes vem passando, com a migração do IPv4 para o IPv6, com o aumento do consumo de dados por toda a população mundial e a crescente necessidade de qualidade nas transmissões, o QoS tem seu papel como fundamental para o bom funcionamento deste "emaranhado" chamado de Internet das Coisas. Ficou claro com esse estudo que sem QoS uma aplicação fundamental pode ser descartada e prejudicada por um simples download de Jogos, filmes dentre outros, com a aplicação de QoS pelos engenheiros de Rede problemas como esses, que hoje podem causar danos e altíssimos custos a grandes empresas e até a pessoas podem ser sanados.

Através da pesquisa de campo e também de vivencia no mercado corporativo, outra conclusão importante desse trabalho é como deve ser bem detalhado e mapeado o plano de QoS de uma empresa pelo seu Engenheiro ou Administrador de Redes, todas as necessidades da empresa e níveis de tolerância a falhas devem ser elencados mapeados e depois implementados. Também é importante uma revisão constante das políticas de QoS conforme as necessidades da empresas vão mudando. Um exemplo real disso são as empresas de Telecomuicações, que hoje vêem uma migração massiva de seus clientes do Perfil de Voz para uma perfil de Dados, isso tem um impacto direto nos seus elementos de Redes e de transmissão, os quais precisam passar por uma revisão massiva da tabelas de balanceamento de QoS e de prioridade de controle, sinalização e transmissão.

Portanto é recomendado as grandes empresas um bom investimento na qualificação de seus gestores de Redes principalmente na área de QoS, e que estes profíssionais mantenham-se atualizados a todos tempo, pois o "mundo" do QoS, tal como é mencionado por (BARREIROS & Lundqvist, 2011) é vasto e tem uma gama enorme de combinações em busca do melhor resultado para os clientes.

A questão incial desse projeto quanto a diferença de aplicação de QoS em redes puramente IPv6 e seu desempenho foi respondida e hoje entedemos que não existe praticamente diferença nenhuma entre a aplicação de QoS em redes IPv6 e IPv4, sendo o desempenho destes dois protocolos de endereçamento semelhante. Um ponto que foi chave nesta conclusão é a indefinição de utilização e aplicação do campo Flow Label do IPv6 para políticas de QoS, pois este pode ser um fator de melhoria nas técnicas de QoS em redes IPv6, porém essa conclusão deverá ser realizada em trabalhos futuros.

5. REFERÊNCIAS

CISCO, Networking Academy. **CCNA Exploration – Fundamentos de Rede.** Cisco Systems, Inc., 2007-2009.

FILIPPETTI, Marco Aurélio. **CCNA 4.1 – Guia Completo de Estudos.** Florianópolis: Editora Visual Books, 2008.

GIL, Antônio Carlos. Como elaborar projetos de pesquisa. 4ª. ed. São Paulo: Atlas, 2002.

LIMA, Carlos Eduardo Parag; HOLLICK Matthias; STEINMETZ Ralf. Diferenciação de Serviços na Internet - DiffServ. Universidade Federal do Rio de Janeiro – Rio de Janero, 2001. Disponível em < http://www.gta.ufrj.br/grad/01_2/diffserv/index.html> Acesso em 13/10/14, 19:20.

OLIVEIRA, Clécio – Segurança e Integração em Redes de Computadores para Ambientes Corporativos. Faculdade Tecnológica SENAC Goiás, 2011. Disponível em <http://clecioliveira.com/blog/2011/06/08/ipv6-hoje-e-seu-dia/> Acesso em 12/10/2014, 12:23

TANENBAUM, Andrew. S. Redes de Computadores. 4ª ed. Rio de Janeiro: Editora Campus (Elsevier), 2011.

BARREIROS, M., & Lundqvist, P. QOS - Enabled Networks. Southern Gate, Chicester: John Wiley & Sons Ltd, 2011

ODOM,W. CCNA ICND2 – Guia Oficial de Certificação do Exame. Rio de Janeiro: Alta Books (2008)

MEGGER, Chrystian L. Estudo e implementação de QoS em redes 802.11g sob topologia malha. 2011. 64 f. Monografia (Especialização em Configuração e Gerenciamento de Servidores e Equipamentos de Redes). Universidade Tecnológica Federal do Paraná. Curitiba, 2011.