



UNIVERSIDADE ESTADUAL PAULISTA
“JULIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA ELÉTRICA
CAMPUS DE ILHA SOLTEIRA

GILVANI ALVES

**O EMPREGO DE TÉCNICAS DE QUALIDADE DE SERVIÇO EM
REDES SEM FIO APLICADO À AUTOMAÇÃO PREDIAL**

Iha Solteira
2016

GILVANI ALVES

**O EMPREGO DE TÉCNICAS DE QUALIDADE DE SERVIÇO EM
REDES SEM FIO APLICADO À AUTOMAÇÃO PREDIAL**

Dissertação apresentada à Faculdade de Engenharia - UNESP - Campus de Ilha Solteira, como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica.

Área de Conhecimento: Qualidade de Serviço.
Automação Predial. Redes sem Fio.

Prof. Dr. AÍLTON AKIRA SHINODA
Orientador

Ilha Solteira
2016

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

Alves, Gilvani.

A474e O emprego de técnicas de qualidade de serviço em redes sem fio aplicado à automação predial / Gilvani Alves. -- Ilha Solteira: [s.n.], 2016
102 f. : il.

Dissertação (mestrado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2016

Orientador: Ailton Akira Shinoda
Inclui bibliografia

1. Qualidade de serviço. 2. Automação predial. 3. Redes sem fio.

CERTIFICADO DE APROVAÇÃO

TÍTULO DA DISSERTAÇÃO: O emprego de técnicas de qualidade de serviço em redes sem fio aplicada à automação predial.

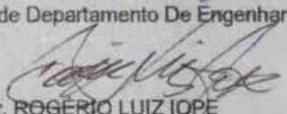
AUTOR: GILVANI ALVES

ORIENTADOR: AILTON AKIRA SHINODA

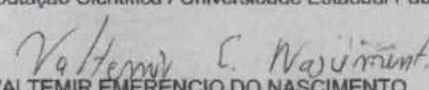
Aprovado como parte das exigências para obtenção do Título de Mestre em ENGENHARIA ELÉTRICA, área: AUTOMAÇÃO, pela Comissão Examinadora:


Prof. Dr. AILTON AKIRA SHINODA

Deplo. de Departamento De Engenharia Elétrica / Faculdade De Engenharia De Ilha Solteira


Prof. Dr. ROGÉRIO LUIZ IOPE

Núcleo de Computação Científica / Universidade Estadual Paulista Júlio de Mesquita Filho


PROFESSOR VALTEMIR EMERENCIO DO NASCIMENTO

Departamento de Área de Informática / INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MATO GROSSO

Ilha Solteira, 23 de fevereiro de 2016

Dedico aos meus queridos pais, Josias Alves (in-memoriam) e Dalva Alves de Carvalho Alves, que me educaram e me deram estrutura para tal feito. À minha esposa e filho, que tantas vezes abdicaram da minha presença, e mesmo assim me incentivaram a alcançar este objetivo acadêmico e profissional. Aos meus amigos, Douglas e Cássia, que sempre foram minha âncora nos momentos mais difíceis. À Prof. Dra. Suzana minha mentora. Ao Professor Dr. Ed'Wilson T. Ferreira que sempre acreditou no meu potencial e me coorientou nesta pesquisa e a todos aqueles que direta ou indiretamente contribuíram para o sucesso desta jornada.

RESUMO

O presente trabalho procede da pesquisa que investigou sobre a aplicação de técnicas de qualidade de serviços - QoS em redes sem fio, à rede principal do sistema de automação predial do Instituto Federal de Educação, Ciência e Tecnológica – Campus Cuiabá Bela Vista. A finalidade proposta foi verificar a eficácia de uma rede sem fio no transporte de pacotes entre a central de controle e o sistema controlado, quando esta rede é compartilhada por fluxos de dados provenientes da comunidade acadêmica. Os serviços consumidos pelos usuários conectados à rede através de dispositivos móveis como *tablets*, *smartphones* e computadores portáteis, geram gargalos capazes de comprometer o sistema de automação predial, considerado um sistema crítico sensível ao tempo de entrega de pacotes. Os procedimentos metodológicos realizados buscaram mitigar a latência dos pacotes provenientes dos processos gerenciados. Os resultados obtidos demonstram que é possível ampliar o número de unidades de controle atendidas, por intermédio de controles de admissão e definição de níveis de prioridades aos pacotes trafegados em um domínio de redes sem fio IEEE 802.11. A escolha do método DiffServ integrado ao algoritmo de escalonamento *Hierarchical Token Bucket* (HTB), permitiu classificar, medir, marcar e policiar os pacotes, o que possibilitou o tratamento e transmissão de pacote no núcleo da rede de modo fácil e eficiente. A taxa de erro de transmissão em situação de estresse na rede sem aplicação de políticas de QoS, quando comparada aos demais cenários são elevados, gerando travamento ou inoperabilidade. O uso dos métodos de QoS proporciona ao sistema maior confiabilidade mesmo em situações de anormalidade na rede. Esta pesquisa demonstra claramente que é possível utilizar um ambiente sem fio compartilhado com aplicações consumidoras de banda como rede principal de um sistema de automação predial, fator que possibilita aos proprietários de imóveis que não possuem infraestrutura de cabeamento estruturado implantar processos prediais automatizados.

Palavras- Chave: Qualidade de serviço. Automação predial. Redes sem fio.

ABSTRACT

This work comes from a research that investigated the implementation of quality service techniques - QoS on wireless networks, applied to the main network of the building automation system of the Federal Institute of Education, Science and Technology - Câmpus Cuiabá Bela Vista. The purpose was to verify wireless network effectiveness in the transport of packets between the control center and the controlled system, when this network is shared by data flows from the academic community. The services consumed by users connected to the network via devices such as tablets, smartphones and mobile computers, generate bottlenecks that can compromise the building automation system, considered a critical system sensitive to package delivery time. The methodological procedures performed seek to mitigate the latency of packets from the managed processes. The results obtained demonstrate that is possible to increase the number of control units served, by using intake controls and setting priority levels to the packets trafficked in a wireless networks domain IEEE 802.11. The choice of DiffServ method integrated to *Hierarchical Token Bucket* (HTB) scheduling algorithm, allowed to classify, measure, mark and police the packets, allowing an easily and efficiently packet processing and transmission in the core of the network. The transmission error rate under stress on the network without the application of QoS policies, compared to other scenarios, is high, generating crash or inoperability. The use of QoS methods provides more reliability to the system even under abnormal conditions in the network. This research clearly demonstrates that it is possible to use a shared wireless environment with band consumer applications as the core network of a building automation system, a factor that enables homeowners who do not have structured cabling infrastructure to deploy automated building processes.

Keywords: Service quality. Building automation. Wireless networks.

LISTA DE FIGURAS

Figura 1 - Sistemas de controle	19
Figura 2 - Sistema supervisorio (nível gerencial).....	20
Figura 3 - Diagrama de blocos da CPU do CLP	21
Figura 4 - Sensores e atuadores em redes ad-hoc.....	23
Figura 5 - Modelo de aplicação de redes sem fio em um edifício.....	24
Figura 6 - Arduíno	25
Figura 7 - Efeito do jitter (Aplicações).....	27
Figura 8 - Marcação de pacotes.....	29
Figura 9 - Campo connection mark, registrando os pacotes marcados	29
Figura 10 - Fluxo dos tokens através dos buckets.....	31
Figura 11 - Síntese de operação do RSVP.....	32
Figura 12 - Funções dos roteadores de borda e de núcleo.....	33
Figura 13 - Controle de admissão na arquitetura DiffServ	34
Figura 14 - Arquitetura de escalonamento	37
Figura 15 - Estrutura do campo DiffServ	37
Figura 16 - Domínio DiffServ	38
Figura 17 - Mecanismos básicos para implantarDiffServ	39
Figura 18 - Escalonador WFQ.....	41
Figura 19 - Escalonador PQ	42
Figura 20 - Escalonador CBQ	43
Figura 21 - Classes root, inner, leaf.....	44
Figura 22 - Exemplo prático de uma árvore HTB.....	44
Figura 23 - Estrutura MPLS	46
Figura 24 - Exemplo de operação do MPLS	48
Figura 25 - Rede MPLS sem TE	49
Figura 26 - Rede MPLS com TE.....	49
Figura 27 -Exemplos de redes mesh.....	51
Figura 28 -Visão Panorâmica do IFMT.....	52
Figura 29 -Visão de alto nível do modelo de RSSF	54
Figura 30 - Topologia da rede primária e secundária.....	57
Figura 31 -Controle de banda por HTB.....	60
Figura 32 -Cenário utilizado para instalação do controle de banda	61
Figura 33 -IPERF - Lado servidor - tráfego UDP	61
Figura 34 -Números de processos simultâneos	63
Figura 35 -Análises diárias com 45 processos simultâneos	63
Figura 36 -Interpolação polinomial em fluxo normal de pacotes.....	64
Figura 37 -Interpolação polinomial - fluxos com QoS.....	64
Figura 38 -Fila DiffServ e HTB	67
Figura 39 -Forwarding table MPLS	67
Figura 40 -Topologia de implantação da política de QoS.....	68
Figura 41 -Estrutura física e lógica	69
Figura 42 -Coleta Wireshark pacotes arduíno e Câmera.....	70
Figura 43 -Comparação entre a vazão (<i>throughput</i>) total e os erros na transmissão em função do tempo (em segundos) para cada um dos 04 cenários.....	70
Figura 44 -Média amostras durante sete dias Com e Sem aplicação de políticas de QoS	72

LISTA DE SIGLAS

AODV	<i>Ad hoc On-Demand Distance Vector</i>
BA	<i>Behavior Aggregated</i>
BRR	<i>Bit-by-bit-Round Robin</i>
CBQ	<i>Class Based Queueing</i>
CBS	<i>Committed Burst Size</i>
CIR	<i>Committed Information Rate</i>
CLP	<i>Controlador Lógico-Programável</i>
CoS	<i>Class of Service</i>
CQ	<i>Custom Queueing</i>
CU	<i>Currently Unused</i>
DCS	<i>Distributed Control System</i>
DIFFSERV	<i>Differentiated services</i>
DSCP	<i>Differentiated Service Code Point</i>
DSR	<i>Dynamic Source Routing</i>
FEC	<i>Forwarding Equivalence Class</i>
FIFO	<i>First In First Out</i>
FTP	<i>File Transfer Protocol</i>
GSC	<i>Group Sequential Communication</i>
HCCA	<i>Hybrid Coordination Function Controlled Channel Access</i>
HTB	<i>Hierarchical Token Bucket</i>
HWMP	<i>Hybrid Wireless Mesh Protocol</i>
IBI	<i>Intelligent Buildings Institute</i>
IEEE	<i>Institute of Electrical and Electronic Engineers</i>
IDE	<i>Interface de Desenvolvimento</i>
IEEE	<i>Institute of Electrical and Electronic Engineers</i>
IETF	<i>Internet Engineering Task Force</i>
IFMT	<i>Instituto Federal de Educação, Ciência e Tecnologia de Mato Grosso</i>
INTSERV	<i>Integrated Services</i>
IOS	<i>Internetwork Operating System</i>
IP	<i>Internet Protocol</i>
ITU-T	<i>International Telecommunications Union – Telecommunication Standardization Sector</i>
LER	<i>Label Edge Router</i>
LIB	<i>Label Information Base</i>
LSH	<i>Label Switched Hops</i>
LSP	<i>Label Switched Path</i>
LSR	<i>Label Switched Router</i>
MANET	<i>Mobile Ad-Hoc Network</i>
MF	<i>Multi Field</i>
MPLS	<i>Multiprotocol Label Switching</i>
OSI	<i>Open Systems Interconnection</i>
PDB	<i>Per Domain Behavior</i>
PHB	<i>Per Hop Behavior</i>
PIR	<i>Peak Information Rate</i>
PQ	<i>Priority Queueing</i>

QoS	<i>Quality of Service</i>
RERR	<i>Route Error</i>
RREP	<i>Route Reply</i>
RREQ	<i>Route Request</i>
RSVP	<i>Resource Reservation Protocol</i>
SCADA	<i>Supervisory Control and Data Acquisition</i>
SDN	<i>Rede Definida por Software</i>
SLA	<i>Service Level Agreement</i>
TC	<i>Traffic Class</i>
TCP	<i>Transmission Control Protocol</i>
TE	<i>Traffic Engineering</i>
ToS	<i>Type of Service</i>
UDP	<i>User Datagram Protocol</i>
UPS	<i>Uninterruptible Power Supply</i>
VLAN	<i>Virtual LAN</i>
VoIP	<i>Voice over IP</i>
WDS	<i>Wireless Distribution System</i>
WFQ	<i>Weighted Fair Queueing</i>
WLAN	<i>Wireless Local Area Network</i>
WPA	<i>Wifi Protected Access</i>

SUMÁRIO

1	INTRODUÇÃO	12
2	AUTOMAÇÃO PREDIAL	16
2.1	CONCEITOS GERAIS	17
2.2	O CRESCIMENTO DO USO DE REDES SEM FIO NA ÁREA DE AUTOMAÇÃO	20
2.2.1	Sensores e atuadores sem fio	20
3	QOS - QUALIDADE DE SERVIÇO - CONCEITOS E DEFINIÇÃO	24
3.1	IMPLANTAÇÃO DE QUALIDADE DE SERVIÇO EM REDE	26
3.2	METODOLOGIA PARA QUALIDADE DE SERVIÇO EM REDE	28
3.3	ORIENTAÇÃO AO PACOTE	29
3.4	TÉCNICAS DE PRIORIZAÇÃO	33
3.4.1	Categorização do tráfego	34
3.5	ESCALONAMENTO	38
4	MULTI PROTOCOL LABEL SWITCHING-MPLS	44
4.1	CONCEITOS FUNDAMENTAIS SOBRE MPLS	44
5	O EMPREGO DE TÉCNICAS DE QUALIDADE DE SERVIÇO EM REDES SEM FIO APLICADA À AUTOMAÇÃO DE PROCESSOS AMBIENTAIS	49
6	O EMPREGO DE TÉCNICAS DE QUALIDADE DE SERVIÇO EM REDES SEM FIO APLICADO À AUTOMAÇÃO PREDIAL	55
6.1	PRIMEIRA FASE – IDENTIFICAÇÃO DO NÚMERO DE DISPOSITIVOS DE AUTOMAÇÃO SUPORTADOS POR UMA ZONA SEM FIO COMPARTILHADA	59
6.2	SEGUNDA FASE – AVALIAÇÃO DO COMPORTAMENTO DA REDE DE AUTOMAÇÃO PREDIAL SOBRE ARQUITETURA DIFFSERV INTEGRADA AO ALGORITMO HTB EM CENÁRIOS REAIS	65
6.3	ANÁLISE DA REDE SEM POLÍTICA DE QOS	72
6.4	ANÁLISE DA REDE COM POLÍTICA DE QOS	72
6.5	ANÁLISE DA MÉDIA DE AMOSTRA COLETADAS DURANTE SETE DIAS COM E SEM APLICAÇÃO DE POLÍTICAS DE QOS	73
7	CONSIDERAÇÕES FINAIS	75
	REFERÊNCIAS	77

APÊNDICE A – ARTIGOS PUBLICADOS EM ANAIS DE EVENTOS.....	83
APÊNDICE B – ARTIGO SUBMETIDO.....	84
APÊNDICE C – COMPARAÇÃO ENTRE A VAZÃO (<i>THROUGHPUT</i>) TOTAL E OS ERROS NA TRANSMISSÃO EM FUNÇÃO DO TEMPO (EM SEGUNDOS) PARA CADA UM DOS 04 CENÁRIOS – SETE AMOSTRAS.....	85
APÊNDICE D – MÉDIA DAS AMOSTRAS DURANTE SETE DIAS COM E SEM APLICAÇÃO DE POLÍTICAS DE QOS POR SERVIÇOS DE AUTOMAÇÃO PREDIAL.....	92
APÊNDICE E – CÓDIGO CARREGADO NO ARDUÍNO – PROJETO SENSOR DE TEMPERATURA USANDO MODBUS/TCP.....	93
APÊNDICE F – CONFIGURAÇÃO DO RÁDIO MIKROTIK NETIFMTAP01.....	94

1 INTRODUÇÃO

A cada dia novos produtos e serviços estão sendo exigidos nos escritórios e residências. Observa-se mudança tanto na arquitetura, na organização e utilização do espaço, no projeto das instalações e nos ambientes das edificações (NEVES, 2002), devido à necessidade de transmissão de voz, dados, textos e imagens, consequências da generalização e extensão da informática na vida cotidiana.

À medida que o conceito de rede amadurece, muitos serviços anteriormente isolados (*standalone*) estão mudando para redes *Internet Protocol* (IP). A convergência de dados, voz, segurança, sistemas de automação predial e muitos outros serviços para uma só rede, podem simplificar a instalação e reduzir significativamente os custos. É recomendável instalar redes de fibra óptica ou de cabeamento metálico de última geração, por prover capacidade escalável em largura de banda e desempenho, porém, nem sempre essa abordagem é possível. Considerando os custos de implantação e o fato de alguns prédios serem desprovidos de infraestrutura para passagens de cabos, o que torna inviável a utilização destas redes.

Para estes cenários, as redes sem fio destacam-se por integrar ambientes sem a necessidade de cabos, provendo serviços de acesso com menores taxas de entrega de dados por tempo, quando comparadas às redes de fibra óptica ou metálicas. Dentro desta perspectiva a utilização de dispositivos sem fio vem se tornando cada vez mais viáveis pelas vantagens da integração e menores custos de aquisição e implantação (MONTEBELLER, 2006).

Porém, trazem consigo algumas desvantagens, entre elas a baixa capacidade de largura de banda e velocidade de transmissão, além de uma alta taxa de erro de comunicação. Estas variáveis quando aplicadas em sistemas de transmissão de “tempo real”, como o caso de aplicações de sistema predial inteligente, geram atrasos na ativação de uma zona sensoreada, causando impactos visíveis aos usuários e ao próprio sistema de controle (MELO, 2005). Contudo, se forem aplicadas técnicas corretas de qualidade de serviços (QoS), torna-se uma solução viável para automação predial.

As redes sem fio padronizadas pela norma *Institute of Electrical and Electronic Engineers* (IEEE) 802.11 não concorrem com os sistemas de cabeamento estruturado, pelo contrário, elas têm a função de complementar a abrangência da rede de dados quando o cabeamento não for viável. Elas são base de comunicação de inúmeros dispositivos encontrados nos estabelecimentos corporativos, como universidades, museus, *shopping centers*, dentre outros. Com a popularização da arquitetura sem fio, grandes partes dos

produtos integram conexão a esta tecnologia, tornando-se útil em aplicações de automação predial (SILVA, 2014a), como nos sensores de medição, monitoramento e controle, tais como: temperatura e umidade, pressão volumétrica, energia, sons e imagens. A grande vantagem dos sensores sem fio é a versatilidade, os sensores são instalados em qualquer ponto do ambiente sem a necessidade de se colocar fios ou cabos.

O desenvolvimento de um estudo de viabilidade no uso de redes sem fio foi necessário para avaliar a transmissão de dados no monitoramento de sensores. Com isto é possível garantir QoS exigida pelas aplicações de gerenciamento e controle, de acordo com suas características temporais de entrega dos dados (PEREZ; HELENA, 2011). Parâmetros como atraso (*delay*), variação do atraso (*jitter*), erros de transmissão e perda de pacotes (*packet loss*) passam a ser importantes para o correto funcionamento se tais aplicações. Manter a QoS tornou-se um desafio, devido as diversas variáveis físicas e/ou lógicas que impactam negativamente na entrega de novos serviços, que requerem mecanismos de transporte de dados mais ágeis e são mais sensíveis ao tempo.

O foco principal deste estudo é combinar técnicas de QoS, como investigações de mecanismos de priorização, controle e engenharia de tráfego, e aplicar no ambiente de redes sem fio utilizado pela comunidade do Instituto Federal de Educação, Ciência e Tecnologia de Mato Grosso (IFMT) – Campus Cuiabá Bela Vista, que compartilha os fluxos de dados com a rede de sensores da automação predial. Nesse contexto, percebe-se que a banda de ocupação do canal foi sobrecarregada pelo tráfego em rajadas gerado pelos alunos e professores. Busca-se garantir que os serviços de comunicação entre os sensores e a aplicação de controle possam ser executados com o mínimo de erros possíveis.

Existem diversas pesquisas sobre o tema em Montebeller (2006), foi apresentado um estudo de tecnologias de comunicação sem fio e suas aplicações na automação predial. Nesse trabalho foram estudados, de forma comparativa, essas tecnologias e os principais sistemas que podem ser automatizados dentro de um edifício inteligente. Demonstrou-se o uso de dispositivos sem fio a partir de um protótipo que foi elaborado com o objetivo de substituir sensores de temperatura de um sistema de ar condicionado. O estudo demonstra também que dispositivos sem fio podem ser mais uma opção na automação de edifícios e residências, podendo servir como solução, em vários casos, para problemas de infraestrutura e de integração dos sistemas de automação.

Em Murazzo et al. (2013), investigam-se o comportamento das redes *Mobile Ad-hoc Network* (MANET) frente à necessidade de requisitos de QoS. Os autores propõem trabalhar o melhoramento dos protocolos de roteamento reativos, *Dynamic Source Routing* (DSR) e *Ad*

hoc On-demand Distance Vector (AODV), para MANET usando o padrão 802.11e. Como segundo objetivo, os autores também trataram da gestão e monitoramento de banda nas redes com fio e sem fio, dividindo e gerenciando tráfego de dados, permitindo o tráfego priorizado em níveis restritos de recursos.

Em Jaramillo (2013), foi proposta uma solução baseada em dispositivos Mikrotik (SIA, 1999) para garantir QoS à rede interna da Escola Politécnica do Exército com sede em Sangolqui. Analisa o sistema operacional Mikrotik, investiga os fatores estabelecidos pela instituição que afetam o rendimento dos serviços de rede, realiza catalogações de aplicações e aloca largura de banda para melhorar o rendimento da rede.

Em Gennaoui et al. (2013), foi comparado o desempenho de aplicativos, usando vários mecanismos de enfileiramento no contexto da arquitetura de serviços diferenciados, *Differentiated Services* (DiffServ) contra o desempenho alcançado na rede com *Multiprotocol Label Switching* (MPLS). O estudo de simulação foi realizado utilizando OPNET TI Guru versão 17, onde mostrou que a arquitetura de DiffServ pode proporcionar certo nível de garantia de QoS, para os casos onde os enlaces do caminho mais curto tomado pelos fluxos de tráfego não estão devidamente provisionados ou com vazão limitada. O MPLS é uma alternativa e mecanismo complementar que permite que o tráfego seja encaminhado ao longo dos caminhos não mais curtos, utilizando os recursos sobre os enlace que em redes tradicionais permanecem sem uso, o que pode levar a níveis mais altos de satisfação nos requisitos de QoS. A configuração de rede pode ser mais refinada através da combinação de MPLS e DiffServ, que podem conduzir a um desempenho ainda melhor da aplicação.

Em Viégas et al. (2012), foi proposta uma comunicação em tempo real para rede local sem fio. Esta estratégia chamada de *Group Sequential Communication* (GSC) tem como objetivo principal reduzir a sobrecarga de rede. A fim de melhorar a confiabilidade do novo esquema, os autores apresentaram um mecanismo tolerante a falhas baseado na estratégia *block-ack* para evitar perder mensagens em tempo real. O GSC foi implementado no *Network Simulation version 2* e o resultado da simulação provou o desempenho superior em relação ao *Hybrid Coordination Function Controlled Channel Access* (HCCA) para comunicações industriais em “tempo real”.

Existem diversas pesquisas sendo desenvolvidas para prover suporte de “tempo real” para redes sem fio, porém poucas aplicadas à comunicação entre os controladores e o sistema supervisor do serviço de automação. Este trabalho visa contribuir aos proprietários de prédios que foram construídos sem perspectiva de ampliação tecnológica implementar soluções de “tempo real”, como os sistemas de automação sobre redes sem fio, pela aplicação

de técnicas de QoS na rede principal que interliga os dispositivos finais de automação ao sistema gerencial supervisorio.

Este texto está organizado da seguinte forma: o Capítulo 2 apresenta as principais características da automação predial, conceituando os sistemas e mostrando os principais elementos a serem automatizados em um edifício inteligente. Este capítulo também apresenta a arquitetura dos sistemas em rede utilizados na automação predial.

O Capítulo 3 apresenta os conceitos e definições de QoS. O objetivo deste capítulo é apresentar as arquiteturas e modelos para provisão de QoS e os mecanismos para qualidade e diferenciação de serviços.

O Capítulo 4 apresenta o multiprotocolo MPLS, seus fundamentos, protocolos e componentes o capítulo mostra as vantagens, desvantagens e sua extensão para engenharia de tráfego.

O Capítulo 5 apresenta a proposta inicial do emprego de técnicas de QoS em redes sem fio aplicado a automação de processos ambientais.

O Capítulo 6 apresenta o uso de QoS em uma rede sem fio em um ambiente acadêmico, que compartilha o fluxo de dados com o sistema de automação predial. É analisada a quantidade de dispositivos de automação suportados por uma zona sem fio compartilhada e usado o método de injeção de pacotes para inundar a rede. Foi avaliado, também, o comportamento da rede de automação predial sobre arquitetura DiffServ integrada ao algoritmo HTB em cenários reais e demonstrados os resultados obtidos.

Adicionalmente, no apêndice A, apresentam-se os artigos publicados em anais de eventos.

No apêndice B apresentam-se os artigos submetidos.

No apêndice C mostram-se os gráficos de comparação entre a vazão (*Throughput*) total e os erros na transmissão em função do tempo (em segundos) para cada um dos 04 cenários – sete amostras.

No apêndice D apresentam-se a média das amostras durante sete dias com e sem aplicação de políticas de QoS por serviços de automação predial.

No apêndice E apresentam-se os códigos carregados no Arduíno para projeto de sensor de temperatura utilizando ModBus/TCP.

No apêndice F apresenta a configuração do rádio Mikrotik NetIFMTAP01.

2 AUTOMAÇÃO PREDIAL

O conceito de automação nasce na indústria objetivando substituir a força de trabalho humana por máquinas e sistemas de controle. Estes equipamentos fazem a gestão e melhoram o controle dos processos, aumentando o rendimento e a qualidade das mercadorias. A partir dos anos 80, a automação proporcionou diversos ganhos aos prédios e residências, tais como bem-estar e comodidade dos usuários e uso inteligente de energia.

Com a automação predial nasce o conceito de edifício inteligente. De acordo com o *Intelligent Buildings Institute* (IBI), o termo refere-se à infraestrutura que fornece um ambiente eficaz e que é economicamente racional, através do aprimoramento dos seus quatro conjuntos básicos: estrutura, sistemas, serviços e gestão, e das relações entre estes. Os edifícios inteligentes auxiliam seus proprietários, gestores e ocupantes a alcançar suas metas sobre as dimensões de custo, comodidade, conformidade, segurança, mobilidade e ganho comercial (MARTE, 1995).

Um edifício inteligente é aquele planejado e construído de modo a disponibilizar uma grande adaptabilidade de uso, provendo características evolutivas, de se adaptar aos objetivos das organizações e de fornecer, a cada instante, um suporte mais eficiente à sua atividade. Portanto, deve conter sistemas de automação, de computação e de comunicações que permitam, de forma integrada e coesa, gerenciar de modo eficaz os recursos existentes, aumentando assim sua produtividade, minimizando o consumo e fornecendo alto grau de conforto e segurança aos usuários que neles circulam (MONTEBELLER, 2006).

Automação de sistemas como controle de iluminação, alarmes de incêndio, sistemas de ar condicionado, dentre outros, foram instalados nos edifícios inteligentes com a finalidade de melhor utilizar tais recursos e assim, diminuir os custos de implantação e utilização (PINTO, 2000).

Os prédios inteligentes são construídos sobre concepção de automação e integração no projeto. Prédios já constituídos possuem barreiras que podem dificultar a implantação de dispositivos de automação e das benfeitorias que estes trazem. Geralmente nestes edifícios não existe infraestrutura prevista para ampliação da acomodação de fiação, conforme normas de cabeamento. É diante dessa perspectiva que o uso de equipamento sem fio na área de automação predial apresenta-se cada vez mais interessante pelos ganhos de integração e facilidade de implantação.

É certo que a automação é fundamental e será um requisito no futuro, permitindo com que os prédios que não possuem sistemas automatizados possam tê-los. Porém seus

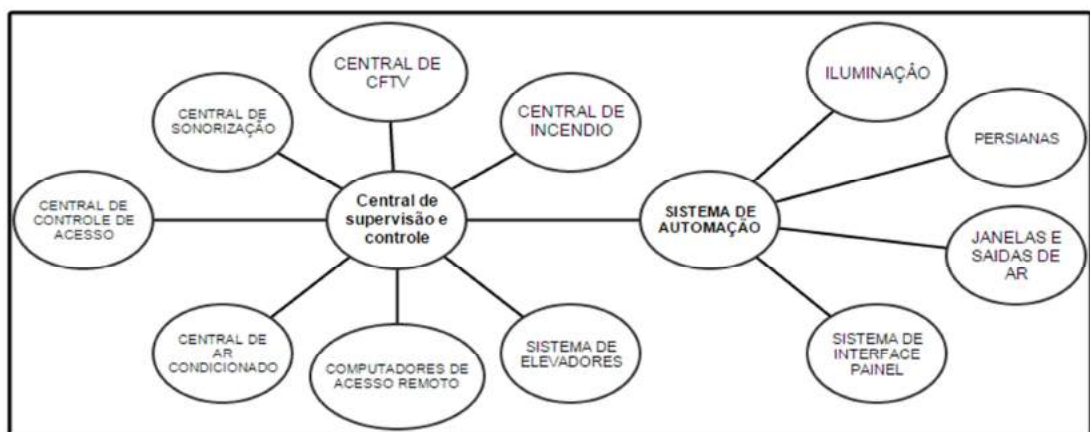
proprietários irão deparar com problemas de falta de caminhos de passagens de cabos e deverão tomar a decisão sobre alterar a estrutura do prédio ou arquitetar um novo que atenda os requisitos.

O uso da tecnologia sem fio é uma opção, pois está presente em dispositivos de sensoriamento e controle, além de computadores. Permite que mesmo prédios antigos sem estrutura capaz de suportar novos avanços, ou que suportam parcialmente, possuam interligação sem necessidade de modificação física no imóvel, agregando valor aos serviços já instalados e garantindo qualidade aos novos serviços a serem implantados.

2.1 CONCEITOS GERAIS

O conceito de integração dos sistemas é princípio indispensável nos edifícios inteligentes. Esta integração proporciona ao Sistema de Supervisão e Controle de um edifício monitorizar sensores, gerenciar atuadores e armazenar eventos oriundos das inúmeras áreas da construção. Estes componentes são interconectados por uma rede de comunicação de dados, permitindo aos sistemas autônomos comunicar-se com a central de controle e supervisão (PINTO, 2000), conforme mostrado na Figura 1.

Figura 1 - Sistemas de controle.



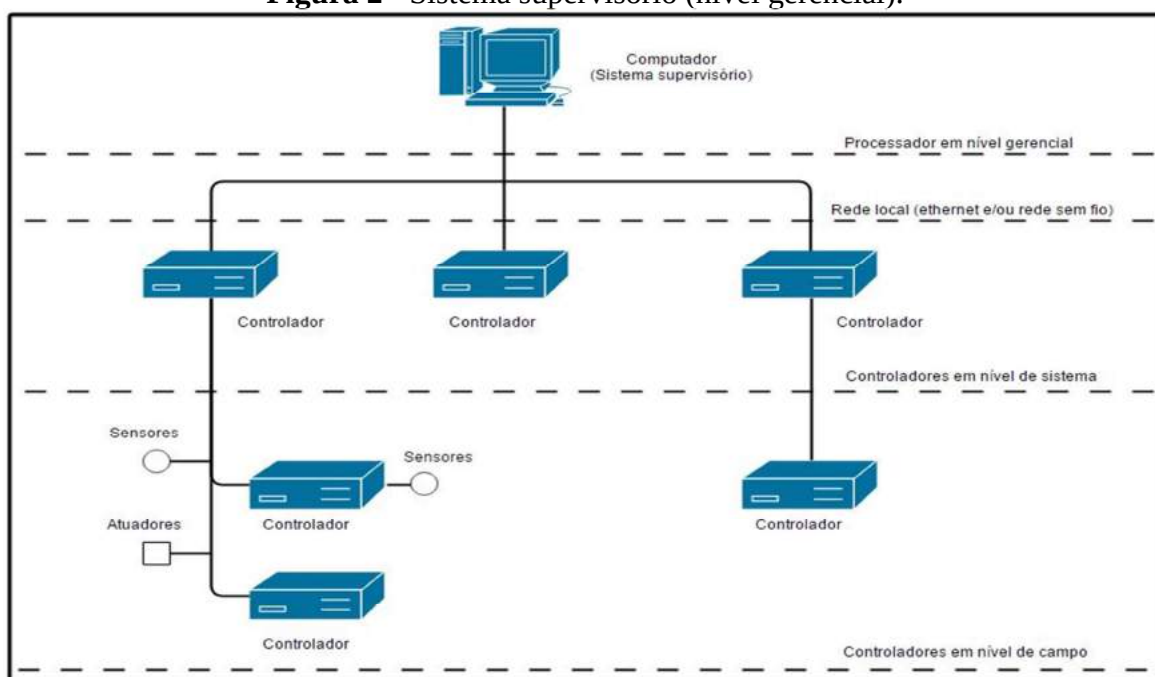
Fonte: Adaptado de (RAMOS, 2013)

A arquitetura de um sistema de supervisão e controle de um edifício é baseada em uma rede de equipamentos (processadores e controladores). Esta rede, chamada de sistema de controle distribuído, *Distributed Control System* (DCS), é definida por topologia hierárquica, com três níveis de controle: gerencial, sistema e campo, conforme mostrado na Figura 2.

Observando esta figura de cima para baixo, os sensores-SE, os atuadores-AT e os controladores-CR em nível de campo encaminham as informações coletadas do processo automatizado ao CR em nível de sistema. No nível de sistema, os pacotes oriundos do nível

imediatamente inferior são encaminhados pela rede principal de automação (sem fio e/ou ethernet), para serem armazenados e visualizados no sistema supervisório em nível gerencial. O sistema supervisório atua nos controladores, encaminhando variáveis de controle, que serão processadas nos níveis de sistema e/ou campo.

Figura 2 - Sistema supervisório (nível gerencial).



Fonte: Adaptado de (RAMOS, 2013).

Os computadores conectados à rede principal do edifício são responsáveis pelo gerenciamento do sistema. Estes computadores fazem parte da central de operação e utilizam um software de supervisão, com recursos geralmente gráficos e de interface amigável. Com este software, os operadores podem visualizar remotamente os eventos, gerenciar sistemas de ar condicionado, iluminação, consumo de energia elétrica, dentre outras funções (PINTO, 2000).

Os sistemas supervisórios, ou sistemas do tipo *supervisory control and data acquisition* (SCADA), proporcionam o monitoramento e rastreamento de informações em um processo de produção. Estas informações são coletadas através de equipamentos de aquisição de dados e os dados são contabilizados, analisados, armazenados e apresentados ao usuário.

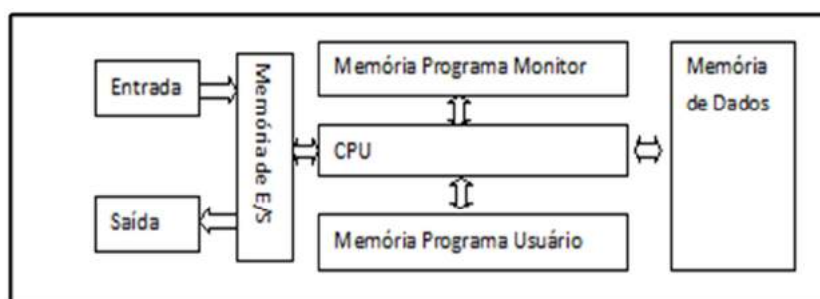
Os controladores de nível de sistema normalmente são representados por controladores lógico-programáveis (CLP) ou por equipamentos dedicados que estão interligados entre si e também com a central de operação por meio da rede principal do edifício (SOUSA, 2004).

Estes controladores recebem informações originadas nos sensores ou outros dispositivos e podem atuar em motores, máquinas e equipamentos através de entradas e saídas digitais

e/ou analógicas de forma autônoma. São instalados em locais nomeados como zonas, centralizando todo monitoramento naquela região. Entretanto, a central de operação também pode encaminhar comandos para os controladores, definindo a operação ou alteração de variáveis de controle daquela área.

Esta capacidade de autonomia dos controladores é chamada de inteligência distribuída, e permite ao sistema estar disponível mesmo diante de uma falha na rede principal ou na central de operações, sem consequências graves às operações vitais do edifício. Alguns sistemas não possuem controladores em nível de sistema e comunicam-se diretamente com o sistema em nível gerencial. A Figura 3 mostra o diagrama de um controlador.

Figura 3 - Diagrama de blocos da CPU do CLP.



Fonte: (SEMERIA, 2000).

Os controladores de nível de campo são sistemas de porte inferior, possuem reduzido número de entradas e saídas digitais e/ou analógicas, e podem ser utilizados para o controle da iluminação, dos ventiladores e *dampers* do ar condicionado, das bombas de água, etc. Também provêm extensões de entradas e saídas remotas dos controladores de nível de sistema. Podem ser exercidos por CLP de pequeno porte ou controladores dedicados com função específica (PINTO, 2000), como por exemplo, dispositivos arduínos e zigbee. A manipulação dos controladores de campo é feita na central de operação, que envia os comandos via controladores de nível de sistema e/ou diretamente.

O uso de redes de comunicação de dados permite a supervisão e o controle de todos os setores do edifício, através de sensores distribuídos geograficamente. Tipicamente apresentam, em automação predial, dois tipos de redes (PINTO, 2000):

- rede principal – segmento que interconecta todos os controladores de nível de sistema à central de operação. Essa rede transporta informações de supervisão, monitoramento e controle do edifício, entre os controladores de campo e sistema até a central de operação. O requisito confiabilidade é de fundamental importância, devido ao quesito entrega de pacotes; e

- rede secundária – segmento que interliga todos os controladores de nível de sistema com os controladores de campo. Por intermédio dessa rede os controladores de campo podem encaminhar informações de sensores e receber comandos de controle de dispositivos originados, indiretamente, da central de operação.

2.2 O CRESCIMENTO DO USO DE REDES SEM FIO NA ÁREA DE AUTOMAÇÃO

As redes sem fio estão transformando-se em uma alternativa muito interessante para automação predial. Os custos do sistema de cabeamento estruturado são bastante elevados, fator este que contribuiu para o aumento de prédios não preparados para automação, aumentando-se ainda mais a aderência das redes sem fio, devido ao menor custo de implantação.

Além da aplicação na comunicação de dados, as redes sem fio podem ser ampliadas e compartilhadas no seu uso aos atuadores, sensores e dispositivos autônomos, dispostos em locais onde o sistema de cabeamento não alcança. Em edifícios inteligentes a aplicação é direta, visto que o conceito de inteligência está ligado ao poder de adaptar-se às tecnologias emergentes (NEVES, 2002).

Na maioria dos edifícios inteligentes, as redes sem fio foram planejadas e compõem a infraestrutura. Porém, em outras situações, principalmente em construções de prédios antigos, existe a necessidade de uma análise de viabilidade das instalações. Em redes convencionais, torna-se mais complicada a instalação por deficiência da infraestrutura, pela falta de rota de passagem de fios e da topologia implantada. Estas instalações geralmente exigem obras civis, como montagem de pisos elevados, dutos e eletrocalhas.

Os custos de aquisição de equipamentos de redes sem fio vêm diminuindo gradativamente, permitindo que um número maior de adeptos da tecnologia usufrua dos seus benefícios.

2.2.1 Sensores e Atuadores Sem Fio

Os atuadores e sensores sem fio são pequenos dispositivos que combinam sensores especializados em determinada função, com reduzida capacidade de processamento, gravação e pouca potência na comunicação sem fio, visando economizar o consumo de energia. Estes equipamentos têm a capacidade de criar uma malha de conexão, podendo chegar a milhares, todos com foco em monitoração e controle de variáveis físicas (PEREIRA; AMORIM; CASTRO, 2005). Os atuadores realizam as funções necessárias no processo controlado,

quando informados pelo controlador. Os sensores são aqueles que enviam informações ao controlador sobre o processo controlado.

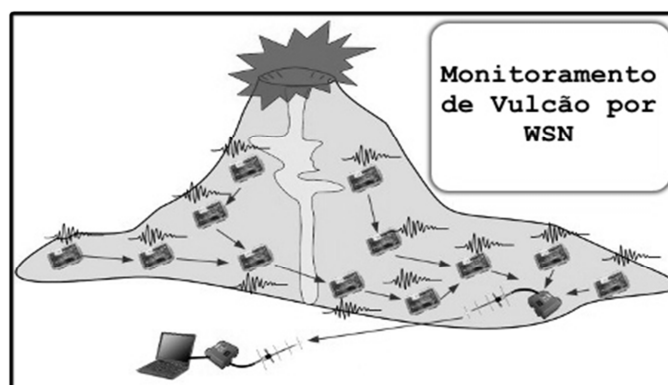
Esta malha de dispositivos interconectados funciona como rede ponto-a-ponto (*ad-hoc*), cujo objetivo é detectar um dado, transportar e controlar características físicas do ambiente. Os dados transmitidos são armazenados em uma base central de dados. Estes elementos da rede podem trabalhar de forma isolada, porém também podem coletar as informações de modo individual (PEREIRA; AMORIM; CASTRO, 2005).

A conexão entre os controladores pode ser sem fio, porém deve ser observada a relação custo-benefício. As redes principais e secundárias de um prédio são projetadas para buscar o máximo desempenho e a simples troca dessas redes por uma infraestrutura sem fio pode não ser uma boa opção, se porventura não aplicar métodos de qualidade de serviço e engenharia de tráfego, visto que as redes sem fio sofrem interferência do meio ambiente, além de ter limitações da arquitetura em relação à largura de banda. Porém estas limitações tendem a diminuir à medida que novas soluções surgem no cenário de redes sem fio.

Em automação predial quase sempre são utilizados grandes quantidades de elementos e as soluções existentes proporcionam a criação de redes sem fio capaz de suportar estes números de dispositivos.

Os controladores e sensores sem fio podem formar uma grande rede de comunicação, mesmo em ambiente inóspito, como é o caso da natureza. Estes equipamentos conectam entre si formando uma malha de conexão e na borda da rede encaminham estes pacotes por rede IP até o computador de destino, conforme mostrado na Figura 4.

Figura 4 - Sensores e atuadores em redes *ad-hoc*.

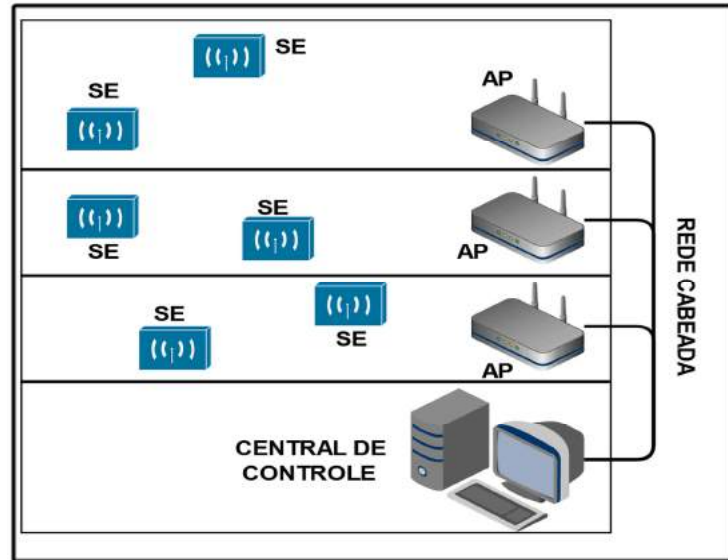


Fonte: (SANTOS, 2012).

Um modelo de automação predial com tecnologia sem fio utilizando a infraestrutura de cabos é mostrado na Figura 5. Neste modelo híbrido a central de controle recebe os dados provenientes dos sensores que estão conectados aos pontos de acesso, *Acess Point* (AP), e no

sentido inverso a central de controle envia os dados através de redes cabeadas até os sensores instalados na borda da rede.

Figura 5 - Modelo de aplicação de redes sem fio em um edifício.



Fonte: Adaptado (MONTEBELLER, 2006).

Em um projeto de rede sem fio, os sensores e dispositivos sem fio contribuem para o aumento dos custos, que são maiores quando comparados a equipamentos com fio. Entretanto, o cabeamento e a instalação da infraestrutura são itens que mais apresentam custos na automação utilizando cabos. A escolha da tecnologia a ser implantada na automação depende de muitas variáveis, incluindo a relação custo-benefício.

A mobilidade dos pontos da *Wireless Local Area Network* (WLAN), que permite movimentar-se em um escritório ou sala dentro de um prédio sem desconexão com a rede, é um dos pontos fortes para o uso de redes sem fio na automação predial. Apesar dos edifícios inteligentes serem concebidos com cabeamento estruturado, caso não seja planejada a expansão, o acréscimo de mais pontos de rede ou mesmo alteração de *layout* podem causar muitas complicações.

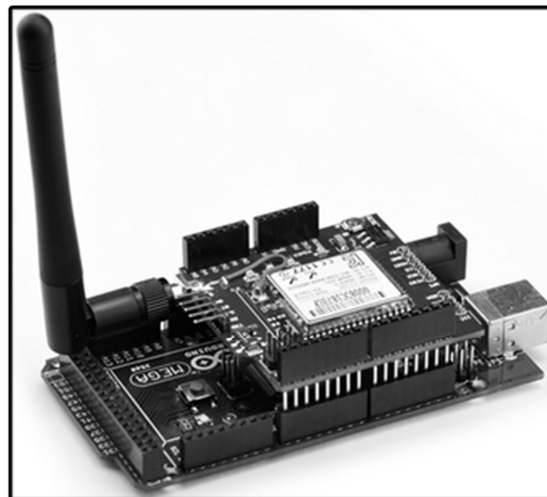
Em automação predial, o uso de redes de computadores permite que uma central conectada à rede tenha acesso às informações geradas pelos sensores e atuadores dentro de um edifício. Com estes dados, a central pode monitorar e gerenciar todas as variáveis do edifício.

Dentro de um edifício, além das redes empregadas para o monitoramento e controle do edifício, os usuários dispõem de uma rede de dados que permite o acesso à Internet ou outros tipos de redes privadas. O seu uso pode ser ampliado aos sensores, atuadores, em locais onde

o uso de cabos não é possível. A aplicação em edifícios inteligentes é direta visto que o conceito de inteligência de um edifício está vinculado à sua capacidade de se adaptar às novas tecnologias (NEVES, 2002). Um microcontrolador será responsável pela comunicação com os dispositivos de medição, de atuação e de envio de dados ao servidor.

Na Figura 6 é apresentado um exemplo desse microcontrolador com módulos (*Shields*) para comunicação em rede sem fio, denominado *Arduíno*. O conceito surgiu na Itália, em 2005, com objetivo de criar um dispositivo de controle para protótipos construídos de forma menos dispendiosa do que outras soluções disponíveis. É uma plataforma de computação física, de código aberto, baseada em placa microcontroladora e um ambiente para escrever a programação a ser embutido para esta placa. Nela pode-se desenvolver objetos interativos, admitindo entrada de uma série de sensores ou chaves, o que permite controlar luzes, motores ou outras saídas físicas.

Figura 6 - Arduíno.



Fonte: (REIS, 2006).

Há uma grande comunidade de pessoas utilizando *Arduíno*, compartilhando seus códigos e diagramas de circuito para que outros os copiem e modifiquem (MCROBERTS, 2011).

Para a comunicação, utiliza-se um *shield* sem fio, o qual foi desenvolvido exclusivamente para se encaixar ao *Arduíno*, possibilitando o envio de dados. Este componente, além de ser pronto para uso direto com o microcontrolador, também possui bibliotecas de comunicação disponibilizadas gratuitamente (TAKAHASHI; VIZCAINO, 2013).

3 QOS - QUALIDADE DE SERVIÇO - CONCEITOS E DEFINIÇÃO

Pela ascensão vertiginosa da Internet nos últimos anos e por conseguinte, sua maturidade, apareceram novas aplicações distribuídas sensíveis ao atraso, que precisavam de grande largura de banda. Muitas classes de serviços, com carência de recursos e diferentes níveis de prioridade, tornaram-se necessárias, criando necessidades de aprimoramento nos serviços entregues. Dessa necessidade surgiu o conceito do termo designado como qualidade de serviço ou QoS.

Qualidade de serviço é determinada na recomendação E.800 do *International Telecommunications Union – Telecommunication Standardization Sector* (ITU-T) como sendo a ação agregada do desempenho do serviço que mede o grau de satisfação de um usuário por um serviço (ITU-T, 2008).

Em um meio de rede compartilhada, QoS está ligada à reserva de recursos ou priorização de tráfego, e essa reserva está agregada a um grupo de fluxos ou para fluxo individual. Também pode ser entendida como um método para disponibilizar alguma técnica de tratamento priorizado para determinado volume de tráfego de rede. Noutra interpretação, QoS é entendida como a aptidão de um elemento de rede ter certo nível de garantia para que as solicitações de serviços e tráfego possam ser satisfeitas.

A habilidade da rede em oferecer um melhor serviço para um tráfego específico em relação aos outros fluxos concorrentes, é provida por técnicas de QoS. Isto é alcançado através da garantia de banda, redução de perdas de pacotes, prevenção e gerenciamento de congestionamento da rede, limitação de banda para determinadas classes de tráfego e na priorização de tráfego através da rede (FORD et al., 1997). Este é o conceito mais claro e objetivo, que mostra os parâmetros a serem trabalhados na obtenção de QoS.

O *jitter* é um parâmetro de suma importância para prover qualidade de serviço, e quanto menor seu valor, maior será a eficiência do serviço. Este é importante para as aplicações executadas em redes, cuja operação adequada dependa de algum modo de garantia para que os pacotes possam ser executados no espaço de tempo requerido. Este é o caso dos serviços de *Voiceover IP* (VoIP), aplicações de tempo real e sensores sem fio para automação predial.

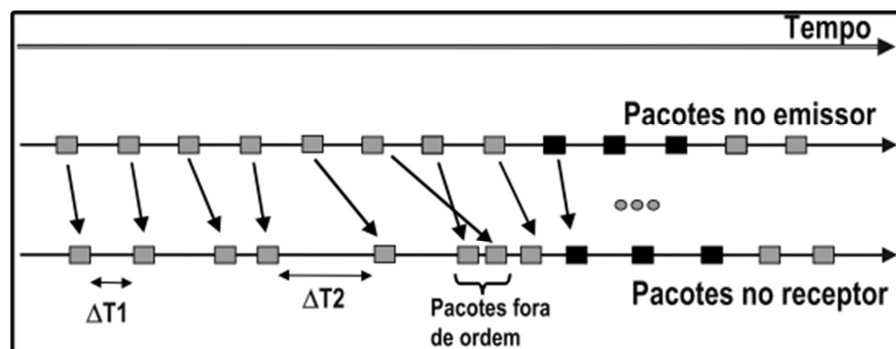
Jitter é a variação do atraso entre pacotes consecutivos que fazem parte de um mesmo fluxo (SEMERIA, 2000). A medida do *jitter* está definida na diferença entre tempo de chegada previsto para um pacote e seu tempo realizado de chegada.

Os nós da rede impõem certo atraso às informações que são variáveis devido a um conjunto de fatores: tempos de processamento desigual nos roteadores intermediários; tempos

de retenções diferentes impostos pelas redes públicas; alterações dinâmicas do tráfego de rede; utilização, por uma conexão, de rotas de custo equivalente, porém com distâncias físicas ou elétricas diferentes (SEMERIA, 2000).

A Figura 7 apresenta o efeito do *jitter* entre a entrega de pacotes na origem e o seu processamento no destino. Com o efeito do *jitter* os pacotes são entregues em periodicidade variável e fora de ordem. Quando um conjunto de pacotes é roteado através da Internet, diferentes pacotes podem tomar diferentes rotas, cada uma provocando um diferente atraso. O resultado é que os pacotes chegam ao destino em ordens diferentes daqueles que foram enviados e o *buffer* do receptor deve ter capacidade de armazenamento para aguardar o pacote fora de ordem e reorganizar na interface de saída. Este problema gera atraso significativo no encaminhamento dos pacotes, sendo prejudicial aos sistemas sensíveis ao tempo, como é o caso dos sistemas de automação predial.

Figura 7- Efeito do *jitter* (aplicações).



Fonte: (VIVA SEM FIO, 2011).

Largura de banda (*bandwidth*) é um conceito genérico usado para explicar a capacidade de um sistema para encaminhamento de dados.

Em um enlace de transmissão de dados, um dos principais atributos é sua capacidade nominal de transportar bits, ou banda nominal. A vazão (*throughput*) ou a banda de rede ocupada pelo meio corresponde à taxa de bits que é realmente transportada, tendo seu valor máximo a banda nominal (FONSECA; STANTON, 2012). É descrito como sendo o número de bytes ou bits por segundo recebidos pelo destinatário na comunicação. É um valor finito, geralmente limitado pelas propriedades físicas do meio de comunicação, velocidade dos enlaces, protocolos, etc (REIS, 2006).

Um limite inadequado de vazão para uma aplicação específica pode gerar atraso e perda de pacotes, caso não haja algum tipo de priorização.

3.1 IMPLANTAÇÃO DE QUALIDADE DE SERVIÇO EM REDE

O estabelecimento de QoS em redes é alcançado por intermédio das funções de enfileirar, escalonar e formatar tráfego; identificar e marcar pacotes; supervisionar e gerenciar QoS. Estas funções permitem o controle dinâmico dos pacotes dentro do domínio de rede.

Gerenciar congestionamento e fila, eficiência de enlace, policiar e formatar o tráfego são algumas das ações quando a pretensão é implantar QoS. O gerenciamento de congestionamento é fundamental para evitar o congestionamento gerado pelos fluxos das aplicações. A gestão do congestionamento define um mecanismo de analisar o fluxo, posicionando os pacotes dessemelhantes em filas de prioridade ou em uma única, onde todos os fluxos adquirem o mesmo tratamento. O gerenciamento de fila é necessário para equalizar o uso deste recurso e sem possibilitar que filas sobrecarreguem e transbordem, ou que os pacotes ultrapassem o espaço de armazenamento e sejam descartados (*tail drop*) ou tratem o descarte com políticas de prioridades preestabelecidas.

A identificação e marcação são alcançadas através do método de classificação. Este método fundamenta-se na provisão de um serviço prioritário a um modelo definido de tráfego, que deve ser caracterizado principalmente por um ou muitos campos do cabeçalho IP, pela porta de entrada, rota característica ou por lista de acesso e posteriormente pode ou não ser marcado. Se o pacote é somente identificado e não é marcado, é considerado pertencer a uma *per-hop basis*, pois esta classificação está vinculada a somente este nó, não passando para o seguinte. Quando o pacote é classificado e marcado, o bit do campo IP precedence é atualizado e a marcação segue o pacote até o destino ou até que seja remarcado.

Na Figura 8 a interface do software Winbox (gerenciador dos rádios sem fio Mikrotik), é demonstrada a marcação de pacotes de origem na rede interna, sentido à rede externa (*prerouting*), bem como os fluxos direcionados à interface interna (*postrouting*) para os pacotes entrando na rede local, utilizando a tabela Mangle. São marcados os pacotes da aplicação de webconferência “Webconf_RNP”, “*File Transfer Protocol (FTP)*” e “Facebook” saindo pela interface externa.

Figura 8- Marcação de pacotes.

Firewall											
Filter Rules NAT Mangle Service Ports Connections Address Lists Layer7 Protocols											
+ - ✓ ✗ 📄 🔍 00 Reset Counters 00 Reset All Counters											
#	Action	Chain	Src. Address	Dst. Address	Prot...	Src. Port	Dst. Port	In. Int...	Out. I...	Bytes	Packets
;;; Webconf_RNP											
4	ma...	prerouting			6 (tcp)					5.6 MiB	4 325
5	ma...	prerouting								28.6 MiB	53 843
6	ma...	postrouting			6 (tcp)				MeshI...	3880.6 KiB	2 720
7	ma...	postrouting								182.1 MiB	224 821
;;; FTP - File Transfer Protocol											
8	ma...	prerouting								310.7 KiB	522
9	ma...	prerouting								310.7 KiB	522
10	ma...	postrouting							MeshI...	285.2 KiB	210
11	ma...	postrouting								285.2 KiB	210
;;; MeshIFMTBVAcad-----Facebook											
12	ma...	prerouting			6 (tcp)				MeshI...	360.7 KiB	1 028
13	ma...	prerouting								9.5 MiB	24 917
14	ma...	postrouting			6 (tcp)				MeshI...	1975.2 KiB	1 377
15	ma...	postrouting								19.6 MiB	47 076

Fonte: Autoria própria.

Na figura 9, percebe-se a correspondência entre as regras implantadas dos pacotes que passam pela interface da origem “src address” para o destino “dst address”, marcados conforme os serviços em “connection mark” e a situação atual da conexão em *Transmission Control Protocol* (TCP) “TCP state”. Os mesmos estão marcados conforme a descrição de seus serviços, no exemplo dado na Figura 8.

Figura 9 - Campo connection mark, registrando os pacotes marcados.

Filter Rules NAT Mangle Service Ports Connections Address Lists Layer7 Protocols									
🔍 Tracking									
	Src. Address	Dst. Address	Prot...	Connecti...	Connection Mark	P2P	Timeout	TCP State	
U	172.16.51.145:57208	54.251.158.63:80	6 (tcp)		con_http_in		15:42:22	established	
A	172.16.51.149:56623	173.223.53.8:80	6 (tcp)		con_http_in		00:05:00	established	
U	172.16.51.245:43641	216.52.203.13:80	6 (tcp)		con_http_in		22:50:49	established	
U	172.16.51.245:52866	216.52.203.13:80	6 (tcp)		con_http_in		22:50:49	established	
U	179.225.137.200:5...	211.151.165.108:80	6 (tcp)		con_http_in		21:53:54	established	
A	172.16.51.67:49222	200.123.198.162:80	6 (tcp)		con_http_out		17:22:03	established	
A	172.16.51.140:51417	107.23.21.160:80	6 (tcp)		con_http_out		00:00:05	close	
A	172.16.51.140:51418	107.23.21.160:80	6 (tcp)		con_http_out		00:00:05	close	
A	172.16.51.140:51419	107.23.21.160:80	6 (tcp)		con_http_out		00:00:05	close	
A	172.16.51.140:51451	23.44.176.159:80	6 (tcp)		con_http_out		00:00:09	close	
U	172.16.3.31:600	224.0.0.2:646	17 (...)		con_skypeout_in		00:02:14		
U	172.16.3.41:600	224.0.0.2:646	17 (...)		con_skypeout_in		00:02:10		
U	172.16.3.51:646	224.0.0.2:646	17 (...)		con_skypeout_in		00:02:12		
U	172.16.48.11:646	224.0.0.2:646	17 (...)		con_skypeout_out		00:02:12		
A	172.16.51.4:49349	200.130.35.147:1935	6 (tcp)		con_webconf_out		23:57:40	established	
A	172.16.3.31:35879	173.194.73.188:5228	6 (tcp)		con_youtube_in		23:57:33	established	
A	172.16.3.31:51510	74.125.26.188:5228	6 (tcp)		con_youtube_in		23:09:26	established	
A	172.16.3.41:46878	74.125.21.188:5228	6 (tcp)		con_youtube in		23:39:26	established	

Fonte: Autoria própria.

A formatação do tráfego (*shaping*) é utilizada para prender o tráfego no limite de uma largura de banda fixada. Esta formatação é utilizada para proteger contra transbordo de filas (*overflow*). No caso de termos uma fonte com barramento de banda larga maior do que o destino é possível inserir *shaping* para que o tráfego fique dentro da largura de banda do destino e evitar momentos de *overflow*. O policiamento (*policing*) fornece um serviço singular ao *shaping* diferenciando a forma de tratamento do excesso, que geralmente é descartado.

3.2 METODOLOGIA PARA QUALIDADE DE SERVIÇO EM REDE

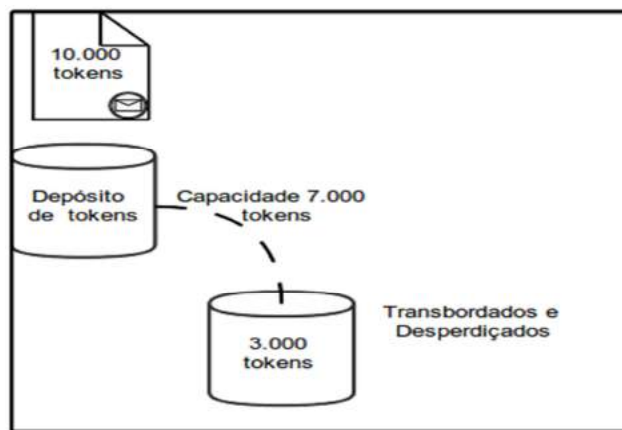
Diversas técnicas foram desenvolvidas para se alcançar os requisitos de QoS, que podem serem usadas combinadas. Algumas predominantes são apresentadas a seguir (TANENBAUM, 2011):

- superdimensionamento: a solução consiste entregar uma largura de banda sobressalente, aumento do processamento e espaço de *buffer* dos roteadores, para que o tráfego flua com extrema facilidade. Neste caso, sempre haverá recursos disponíveis. Porém, é uma solução de custo elevado;
- armazenamento em *buffers*: baseia-se em providenciar *buffers* para o depositar o fluxo na recepção. O mecanismo amplia o atraso, conseqüentemente ameniza a flutuação. Parâmetros como largura de banda e confiabilidade não são atingidos. Para aplicações que usam áudio e vídeo, onde o requisito de flutuação é primordial, este método é muito útil. Muitos sites de vídeo por demanda ou tempo real, “*bufferizam*” alguns segundos antes de transmitir seu conteúdo;
- moldagem de tráfego: atua no lado servidor, o inverso do armazenamento em *buffers*. Assim como pacotes podem ser recebidos desordenadamente, eles também podem ser encaminhados fora de ordem. Isto pode causar congestionamento na rede. A moldagem de tráfego regula a taxa média da transmissão, conforme política predefinida, sobre determinado padrão de tráfego, na configuração da conexão. Esta aplicação desta técnica é de fundamental importância para aplicações de dados em tempo real;
- reserva de recursos: objetiva reter recursos ao longo de uma rota especificada para um fluxo, para assegurar que a demanda de recursos seja satisfeita;
- controle de admissão: consiste na capacidade de roteador, baseado no seu poder de processamento e no número de compromissos já aceito em relação a outros fluxos, se pode aceitar ou rejeitar o fluxo que entra. Este tipo de decisão é mais complicado do que simplesmente comparar os recursos solicitados com os disponíveis, fatores como

buffers e ciclos de CPU devem ser especificados, pois as aplicações não reconhecem estes elementos, apesar de conhecer seus requisitos; e

- algoritmo do balde furado: assim como a analogia de um balde furado de água apresentado na Figura 10, que independente da velocidade de entrada do fluxo de água, o fluxo de saída será semelhante, pode se aplicar o paradigma de um balde de pacotes. Se o balde lotar e transbordar a capacidade, mesmo com a vazão existente, a água transbordará pelas laterais, e, assumindo a forma de um balde com pacotes de dados, os pacotes que entrarem após o uso completo dos recursos serão descartados. Consiste em um sistema de enfileiramento com período de serviço constante.

Figura 10 - Fluxo dos tokens através dos *buckets*.



Fonte: Autoria própria.

3.3 ORIENTAÇÃO AO PACOTE

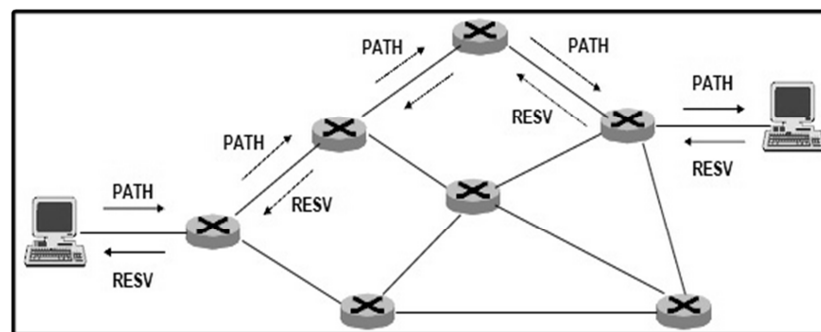
A RFC 791 (POSTELS, 1981), publicada em 1981, apresenta um modelo de tratamento de pacotes através da definição de indicadores em cada unidade de pacotes. Em cada pacote existe um octeto designado de *Type of Service* (ToS), onde a relevância do pacote (campo *precedence*) e as necessidades do serviço (campo ToS) são marcadas. O campo ToS fornece uma indicação dos valores teóricos da qualidade de serviço esperada.

No começo da década de 90, foi proposto o modelo chamado de Serviços Integrados, posteriormente chamado *Integrated Services* (IntServ), RFC 1633 (BRADEN; CLARK; SHENKER, 1994). Esta arquitetura descreveu o conceito de qualidade de serviço relativa ao fluxo, mecanismos de controle de admissão e reserva de recursos. E propôs que recursos, como largura de banda tivesse explicitamente gerenciado de modo a corresponder com os objetivos das aplicações.

O processo de reserva de recursos e controle de admissão é praticado pelo protocolo de sinalização *Resource Reservation Protocol* (RSVP) (BRADEN et al., 1999), através do encaminhamento de mensagens de sinalização em cada roteador, envolvidos no transporte do fluxo fim a fim. Este protocolo tem como objetivo observar a disponibilidade de recursos em cada equipamento e, caso haja recurso, processar a reserva de recursos de acordo com a necessidade do fluxo. Não havendo disponibilidade de recursos em quaisquer uns dos roteadores, a reserva não é criada.

Contudo, as características deste protocolo geram consigo sérias dificuldades de escalabilidade, pelo fato que os roteadores precisam manter o estado, mesmo sem o uso da reserva por uma aplicação, causando desperdício de recursos. Na Figura 11 é mostrado o protocolo RSVP, que prove a reserva de caminho entre dois computadores, encaminhando mensagem de solicitação de reserva e recebendo na origem a resposta para efetivamente processar a reserva de recurso.

Figura 11- Síntese de operação do RSVP.



Fonte: (REIS, 2006).

Em 1998 a RFC 2474 tornou público a arquitetura DiffServ (NICHOLS et al., 1998) como um padrão escalável, possível de ser instalado em fases, acompanhando a evolução e dentro de domínios de redes administrativas. Na arquitetura DiffServ a unidade essencial é o tráfego agregado, que baseia em fluxos individuais com requisitos semelhantes de QoS, portanto cada tráfego agregado é vinculado a uma classe de serviço, *Class of Service* (CoS). Deste modo, fluxos individuais que fazem parte do mesmo tráfego agregado obterão tratamentos iguais pela rede, sendo que este tratamento é denominado de comportamento agregado, *Behavior Aggregated* (BA).

A identificação de cada CoS é executada de forma explícita através de marcação de bits no cabeçalho do pacote IP no campo *Differentiated Service Code Point* (DSCP). Este campo é uma redefinição do campo ToS, só que agora apenas 6 bits do conjunto de 8 são precisos. Deste modo a marcação do campo DSCP define o tratamento que será executado no

transporte dos pacotes em cada roteador. Esta marcação caracteriza o que chamamos de comportamento por salto *Per Hop Behavior* (PHB).

Enquanto que na arquitetura IntServ os processos de reserva de recursos e controle de admissão são processados localmente pelos roteadores, na arquitetura DiffServ estas tarefas são movidas para o nível do domínio da rede. Deste modo, são as regras do domínio que designam quais pacotes serão aceitos junto à rede. Essas políticas são armazenadas geralmente em uma entidade central e são processadas nas bordas da rede.

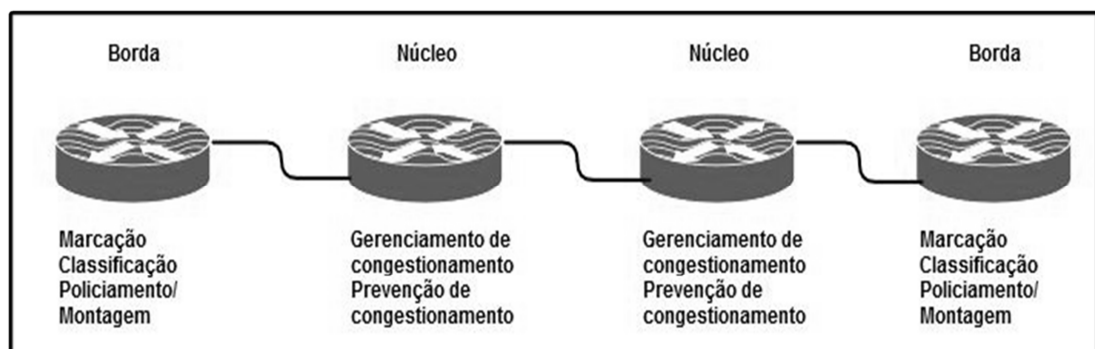
O *Service Level Agreement* (SLA) descreve o comportamento que o tráfego do cliente deve receber dentro do domínio. Utilizam variáveis, tais como: taxa de pico de informação *Peak Information Rate* (PIR), taxa de informação comprometida *Committed Information Rate* (CIR), tamanho da rajada comprometida *Committed Burst Size* (CBS), atraso fim a fim, *jitter*, entre outros.

Na arquitetura DiffServ a complexidade da rede é atribuída às bordas, possibilitando o tratamento e a transmissão de pacotes no núcleo da rede de modo fácil e eficiente. Os roteadores de borda, denominado de *edge router*, executa as seguintes funções:

- classificação: vincula o tráfego ao seu correspondente SLA;
- medição: calcula as taxas de geração do tráfego (PIR, CIR, CBS);
- marcação: atribui o valor de DSCP relacionado ao PHB aguardado pelos pacotes junto à rede; e
- policiamento: executa ações para os pacotes que apresentam fora de um perfil SLA, tomando decisão sobre o descarte os pacotes, ou designação um valor de DSCP com qualidade de serviço inferior, ou retardar a transmissão dos pacotes.

Na Figura 12 ilustram-se as funções dos roteadores de borda e de núcleo. Os roteadores de núcleo ou interior da rede introduz o PHB conforme a marcação no campo DSCP, apoiando-se em métodos de gerenciamento de filas, escalonamento e descarte de pacotes.

Figura 12 - Funções dos roteadores de borda e de núcleo.



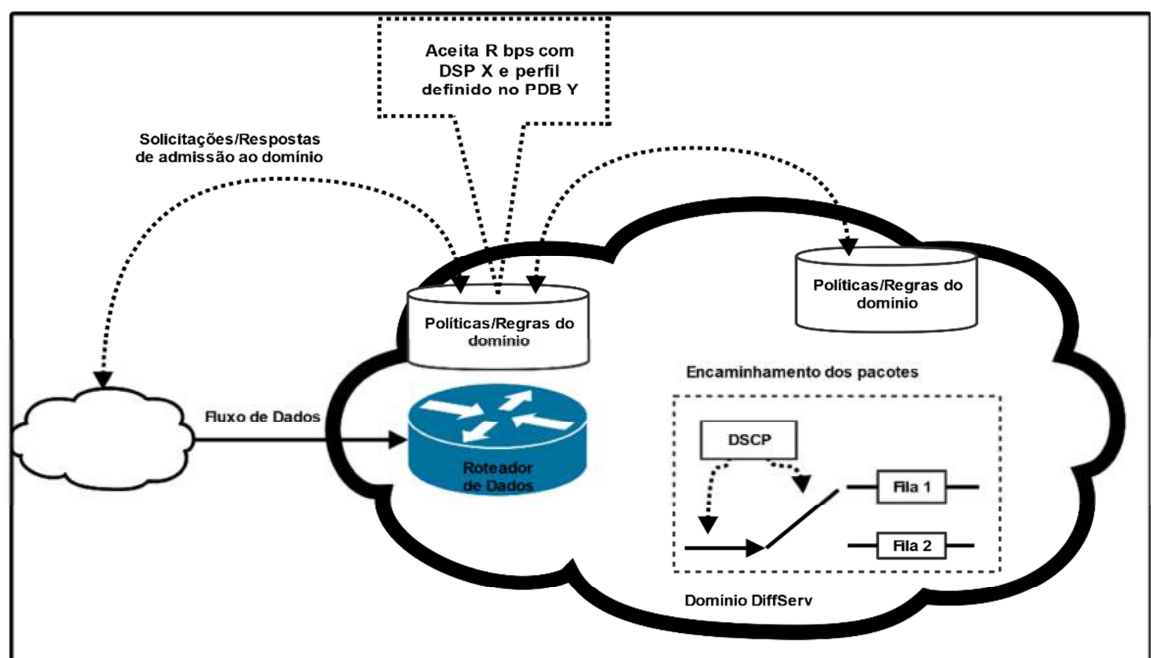
Fonte: (MORTOL; MOREIRA, 2012).

O tratamento uniforme atribuído a uma determinada classe de serviço, durante todo o domínio, chamado de comportamento por domínio *Per Domain Behavior* (PDB). A exclusão do processo de reserva de recurso e controle de admissão, do nível dos roteadores para o nível do domínio da rede, é vista como o desmembramento entre o plano de dados e o plano de controle.

- plano de controle: suporta o roteamento e muda informações de rótulos entre os dispositivos adjacentes; e
- plano de dados: também denominado como plano de encaminhamento, encaminha as informações com base nos endereços ou rótulos de destino.

Na Figura 13 demonstra-se o processo de controle de admissão na arquitetura DiffServ. O Fluxo de dados é admitido junto ao tráfego agregado, logo após corresponder à condição definida por políticas/regras do domínio no roteador de borda. Este comportamento é válido em todo domínio, pois a política é conhecida por todos os roteadores. A fila é escolhida de acordo com o DSCP especificado.

Figura 13- Controle de admissão na arquitetura DiffServ.



Fonte: (REIS, 2006).

No processo de controle de admissão um fluxo é admitido junto a um tráfego agregado, e vinculado a um definido PDB. Esse processo pode ser executado de modo estático ou dinâmico, sendo que para a forma dinâmica é exigido sinalização por fluxo proporcionado pela arquitetura DiffServ. Sua escalabilidade é provida pelas características informadas abaixo e são de suma importância:

- alteração no controle de admissão, como por exemplo, nas regras do domínio, ocasionam reconfigurações na borda da rede, contudo, não altera as configurações do interior da rede; e
- a decisão em relação à aceitação ou não do fluxo, é armazenada em uma base de dados, definindo as políticas do domínio, e a mesma é encaminhada aos roteadores de borda.

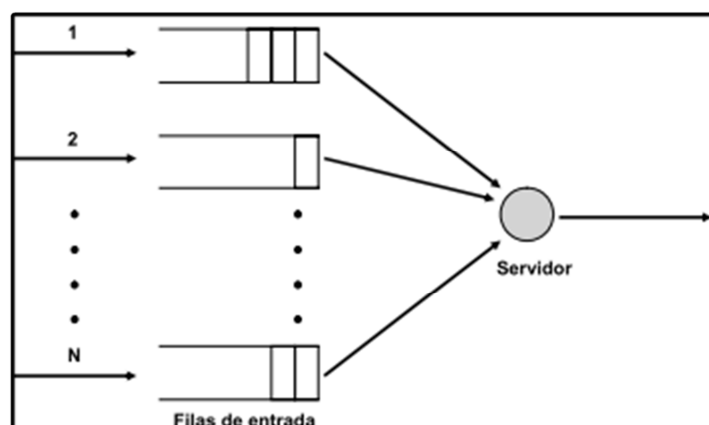
Apesar de a arquitetura DiffServ ter demonstrado como uma solução escalável desde a sua concepção, existem ainda alguns desafios a serem alcançados.

3.4 TÉCNICAS DE PRIORIZAÇÃO

A atribuição básica de um algoritmo de priorização é escolher, entre os pacotes prontos a serem encaminhados, para cada segmento de saída de um roteador, o pacote a ser encaminhado no próximo ciclo. Uma simples arquitetura é mostrada na Figura 14, o *buffer* de entrada no servidor, escalona o encaminhamento do pacote de acordo com a prioridade do serviço.

Por causa dos diferentes atributos e necessidades de QoS dos tráfegos em uma rede, principalmente com a coexistência de voz, vídeo e dados críticos na rede, reforça a essencialidade dos métodos de priorização e compartilhamento de banda. Para que se possa executar a priorização de tráfegos específicos, faz-se necessário ter métodos de classificação do tráfego.

Figura 14 - Arquitetura de escalonamento.



Fonte: (OYAMA; LUCENA, 2002).

Os tráfegos de dados, voz e vídeo possuem formas distintas. O tráfego de dados, normalmente, é injetado na rede em rajadas, tendo pouca sensibilidade a atraso, porém sensível a perdas. Os dados de dispositivos gerenciáveis remotamente de sistemas de automação são sensíveis a atraso e pouco sensíveis a perda. Os tráfegos de vídeo e voz são

sensíveis a atraso, mas são menos sensíveis a perdas. Essa diversidade nas características dos tráfegos e consequentemente suas diferenças na necessidade de QoS demonstra que os tráfegos precisam ser tratados de forma diferenciada e separados em classes que reflitam estes atributos (NYIRENDA-JERE, 1996).

3.4.1 Categorização do Tráfego

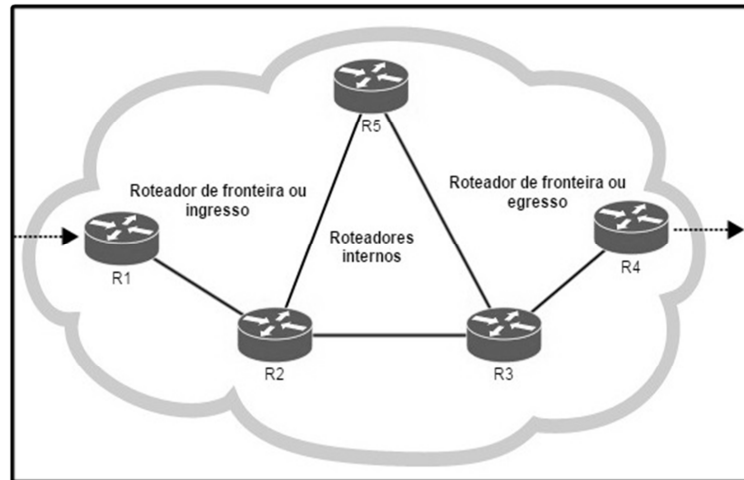
Para a aplicação de prioridade aos tráfegos, estes precisam ser antes identificados e marcados. Aplicando-se classificação de pacotes, é possível particionar os tráfegos de uma rede em oito níveis de prioridade ou classes de serviço. Para executar a classificação do tráfego, cada pacote que entra no roteador de classificação é roteado para uma classe específica. Os pacotes que estiverem classificados em uma única classe obterão o mesmo tratamento, pelo algoritmo de priorização.

Há diversas alternativas de classificação de um determinado tráfego. A vinculação pelo classificador da classe para o tráfego pode estar ligada a diversos aspectos, no entanto algumas formas de identificação e classificação do tráfego são mais comuns, a saber: (FERGUSON; HUSTON, 1998).

- protocolos da camada de rede ou transporte, tais como IP, TCP, *User Datagram Protocol* (UDP);
- número da porta de origem;
- número da porta de destino;
- endereço IP de origem;
- endereço IP de destino;
- interface de chegada do tráfego; e
- fluxo, uma combinação do endereço IP de origem e destino com o número da porta de origem e destino.

No DiffServ, os pacotes são marcados diferentemente para criar muitas classes de pacotes, que serão tratadas de forma diferente através dos serviços desocupados na rede. Os serviços são entregues conforme o *Differentiated Services CodePoint* (DSCP).

O DSCP está implementado no campo DiffServ descrito na RFC-2474 (NICHOLS et al., 1998). Esta definição modifica a definição do IPv4, onde o campo é nominado de ToS, e altera a definição do IPv6, onde o campo é chamado *Traffic Class* (TC). Na Figura 15 apresenta-se o campo DiffServ, o mesmo é estruturado por seis bits (DSCP) utilizado para designar o comportamento por salto, ou PHB, que o pacote passará em cada nó. Este campo é tratado como um ponteiro de uma tabela que é utilizada para escolher um mecanismo de

Figura 16 - Domínio DiffServ.

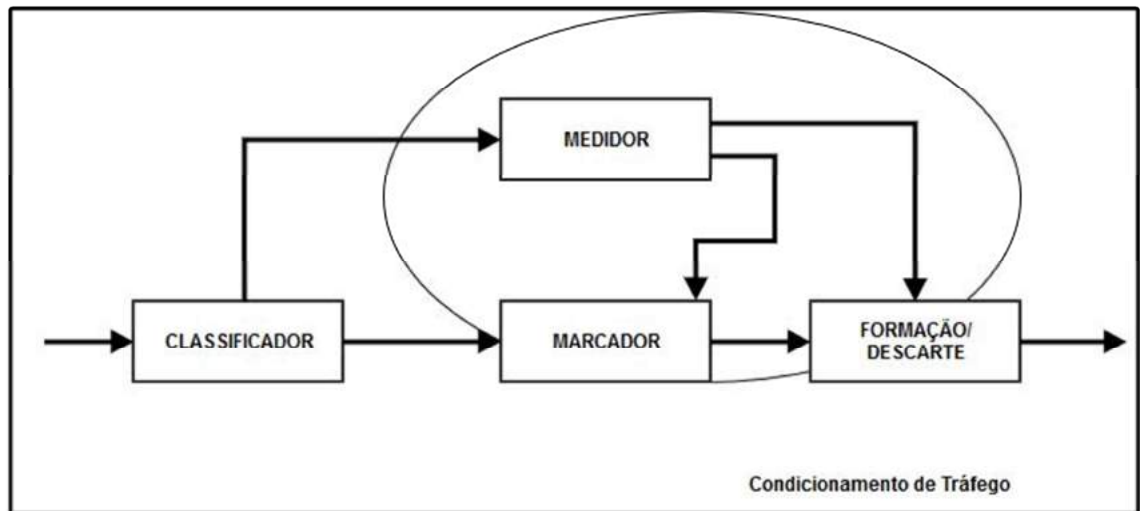
Fonte: Adaptado de (SILVA, 2005).

Nos roteadores de borda, que efetuam a conexão da rede do usuário com o domínio DiffServ, existe uma maior complexidade do modelo DiffServ. Estes realizam funções como classificação e condicionamento dos pacotes de cada fluxo de entrada no domínio.

Para implantação de QoS nas redes IP é fundamental o uso de muitos mecanismos nos roteadores. Estes são usados para fornecer a simplicidade no tratamento aos pacotes conforme modelo DiffServ. Entretanto podem ser utilizados para finalidades mais complexas.

Na Figura 17 são apresentados os mecanismos básicos que devem estar inclusos em um roteador de borda utilizando DiffServ. Pode-se observar que primeiramente os pacotes são classificados conforme tipo de serviço. O bloco medidor mede o fluxo para verificar se um pacote está dentro dos limites do perfil daquele tráfego. O bloco marcador tem a função de marcação ou remarcação do campo DiffServ para um valor específico de DSCP. O conformador, ou suavizador de tráfego, possui a função de atrasar um ou mais pacotes, de um fora do perfil, até que estes estejam dentro da conformidade do perfil, descritos na política de cada fluxo, e possam ser transmitidos pela rede. O último, o bloco descartador, ou policiador, tem a atribuição de descartar pacotes fora do perfil, tomando por base regras do perfil.

Figura 17 - Mecanismos básicos para implementar DiffServ.



Fonte: (ARDUÍNO, 2015).

Para realizar o tratamento de QoS, é necessário primeiramente classificar os pacotes e logo após condicioná-los. O classificador de pacotes coleta os pacotes de um fluxo de tráfego baseado no conteúdo de campos do cabeçalho destes pacotes. O DiffServ utiliza duas maneiras de classificar os pacotes: seleciona os pacotes através do valor do DSCP por comportamento agregado ou seleciona pacotes com base na combinação de valores de um ou mais campos do cabeçalho, chama-se esta forma de multi-campo, *Multi Field* (MF).

O condicionamento de tráfego, normalmente é atribuído aos nós de borda do domínio DiffServ, porém podem estar atribuídos aos nós interiores. Este condicionador pode conter todos os blocos ou apenas alguns. Os pacotes que não fazem parte do perfil podem ser gerenciados de formas distintas, podendo ser descartados ou remarcados para uma classe inferior através da função de policiamento do tráfego, ou conformado-suavizados até que fiquem dentro do perfil. Na prática, as funções de medição e conformação ou descarte são implantadas, principalmente, através do método balde de fichas (*token bucket*), que será explicado em momento oportuno.

Em sua totalidade, os mecanismos apresentados têm grande relevância na implantação de DiffServ, e são utilizados nos roteadores do seguinte modo:

- roteadores de ingresso – fazem classificação de pacotes tipo MF, em geral, e condicionamento de tráfego nos fluxos de entrada. Também aplicam o PHB devido, com base no processo de classificação, através de mecanismos de diferenciação de serviços;

- roteadores internos – utilizam a classificação tipo BA e vinculam ao PHB apropriado através dos mecanismos de diferenciação de serviços. Podendo fazer também prevenção de congestionamentos; e
- roteadores de egresso – geralmente efetuam conformação do tráfego nos pacotes de saída. E podem fazer a classificação MF ou BA e remarcação de pacotes, caso exista um perfil no domínio de saída.

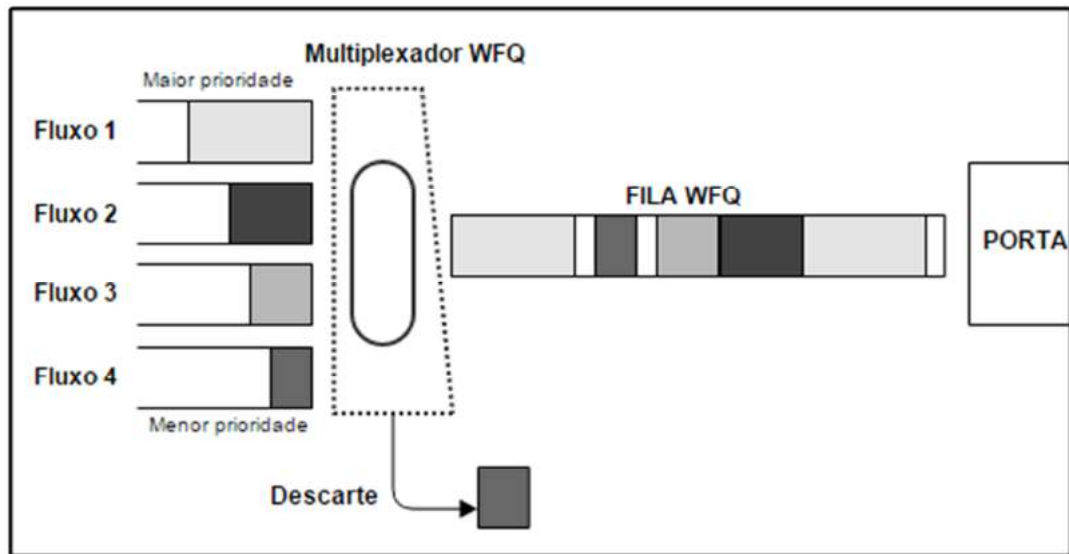
3.5 ESCALONAMENTO

O papel do escalonador é estabelecer regras de enfileiramento e prevenção de congestionamento nas portas dos roteadores (ERDTMANN; NETO, 2003). O algoritmo de escalonamento escolhe qual o seguinte pacote a ser encaminhado na fila de espera, distribuindo a largura de banda da ligação pelos fluxos diferentes, atribuindo a cada um destes, a largura solicitada e aceita.

A distribuição dos recursos em redes e, em especial, a adesão às estratégias de multiplexação estatística, criam momentos de contenção por motivo de competição pelo uso dos recursos. Em redes que provem integração de serviços torna-se imprescindível proporcionar QoS diferenciada por categorias de serviço ou classes de tráfego, oferecendo garantias de eficácia às aplicações críticas e no mesmo momento permitindo partilhar os recursos de forma igualitária.

O algoritmo mais popular é o *First In First Out* (FIFO), que tratam os pacotes conforme ordem de chegada à fila. Nesta fila o tráfego é repassado, sem qualquer forma de classificação. Isto acontece quando não se está utilizando qualidade de serviço, portanto este mecanismo não é indicado.

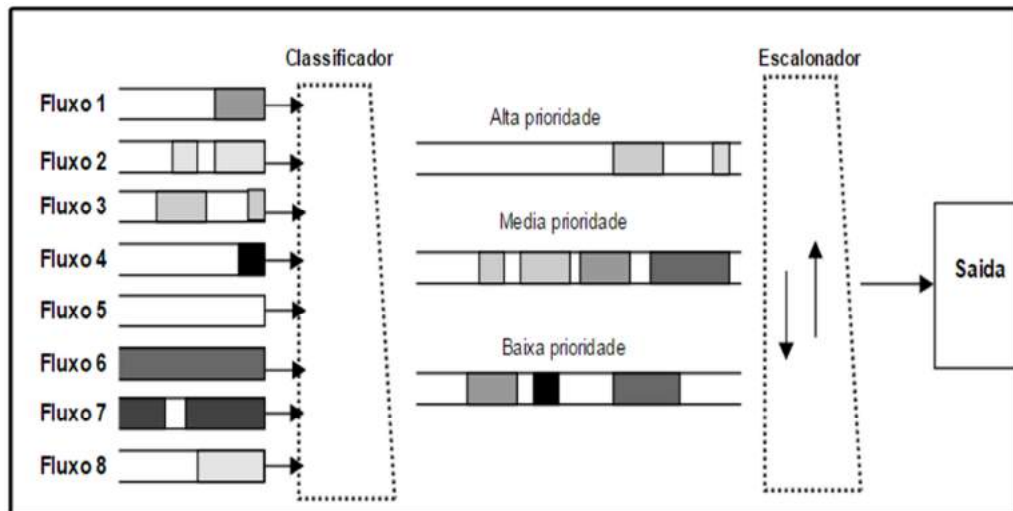
Na figura 18 é demonstrado o escalonamento *Weighted Fair Queueing* (WFQ), este algoritmo denominado WFQ baseia-se no algoritmo chamado *Bit-by-bit-Round Robin* (BRR), onde o fluxo individual é preso a uma fila de saída exclusiva e um bit de cada fluxo é encaminhado pela porta de saída. O algoritmo escalona o tráfego prioritário para o início da fila, reduzindo o tempo de resposta. Enquanto ao mesmo tempo, compartilha o restante da banda com os outros tipos de fluxo de uma forma justa (KENDY; NABAS, 2009).

Figura 18 - Escalonador WFQ.

Fonte: Adaptado de (REIS, 2006).

O algoritmo *Priority Queueing* (PQ) é fundamentado na visão de que alguns tipos de tráfego podem ser identificados e qualificados como tráfego prioritário e deste modo encaminhados antes de outros tráfegos. Este mecanismo de escalonamento apresenta, no entanto, falha de atraso excessivo aos tráfegos de baixa prioridade, contribuindo para o aumento da latência (FERGUSON; HUSTON, 1998).

O roteador verifica minuciosamente cada pacote para identificar como este deve ser escalonado, sobrecarregando deste modo o processamento. Na Figura 19 é demonstrado como os pacotes são recebidos na interface de entrada e reordenados, fundamentado em políticas definidas pelo administrador, até serem adicionados na fila de saída. Os pacotes de alto grau de prioridade são inseridos na fila de saída primeiramente que os pacotes de menor prioridade. É salutar destacar que uma fila só é alimentada se todas as filas com maiores prioridades estiverem sem volume de tráfego, podendo levar a situação de negação de serviço, a não ser que sejam impostos limites à largura de banda utilizada por cada classe.

Figura 19 - Escalonador PQ.

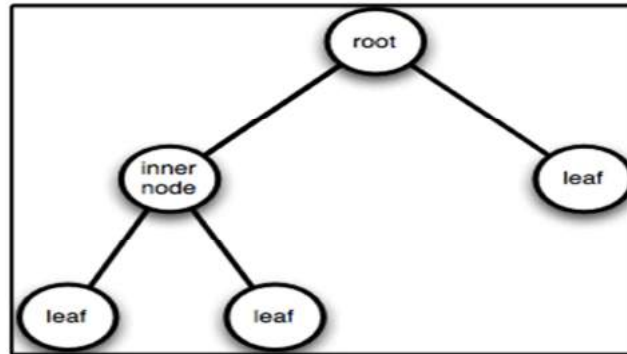
Fonte: Adaptado de (JOSÉ, 2012b).

O algoritmo de escalonamento *Class Based Queueing* (CBQ), também designado *Custom Queueing* (CQ) foi descrito para proporcionar a que muitas aplicações, com necessidades de largura de banda pequenas ou exigências de latência controlada, partilhem o mesmo recurso de rede. Este mecanismo é uma variante do escalonamento por prioridades, onde muitas filas de saída podem ser implantadas. Pode-se também escolher a preferência com que cada fila será servida e o número de tráfego escalonado que deve ser encaminhado de cada fila em cada ciclo na rotação do serviço (FERGUSON; HUSTON, 1998).

No exemplo da Figura 20 três *buffers* com prioridades diferentes foram criados: alta, média e baixa prioridade. Logo após o tráfego em cada fila ser concluído, os pacotes continuam a ser encaminhados até que o contador supere o limite definido para essa fila ou a fila encontre-se vazia. Deste modo, o tráfego que foi categorizado e classificado para ser escalonado em múltiplas filas encontra boas condições de ser encaminhado sem que um atraso considerável seja apresentado, proporcionando ao sistema diminuir a escassez de armazenamento em *buffer*.

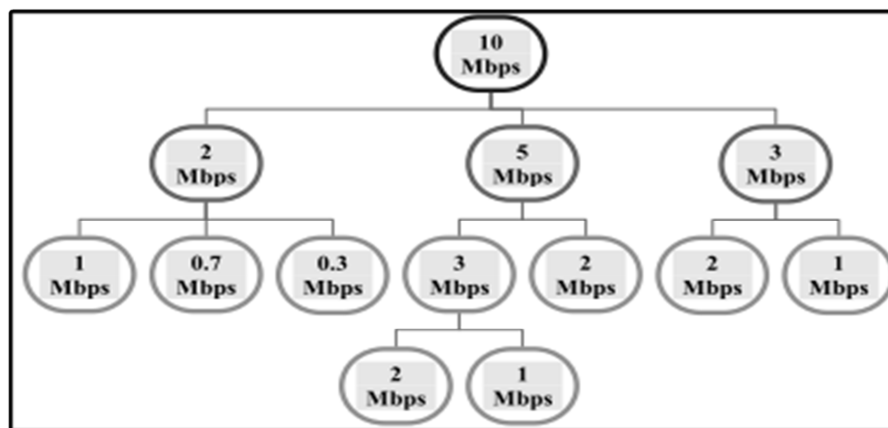
vazão das classes *inner*. O empréstimo de banda pode ser feito às classes superiores, caso haja disponibilidade.

Figura 21 - Classes *root*, *inner*, *leaf*.



Fonte Autoria própria.

Figura 22 - Exemplo prático de uma árvore HTB.



Fonte: (SEMERIA, 2000).

Após a marcação do tráfego é realizado gerenciamento em filas. O HTB usa o conceito de fichas (*tokens*) e baldes (*buckets*) para controlar a utilização da largura de banda em um *link* e ajustar a taxa de transferência de acordo a regra definida para o fluxo.

Uma das vantagens do HTB como modelador de tráfego é a capacidade de partilhar a largura de banda entre os serviços. Este escalonador garante certa porção do enlace para cada classe, permitindo ainda especificar quanto pode ser emprestado entre as classes.

Os mecanismos de descarte controlam quais os pacotes, em uma fila de espera de um roteador, que precisam ser descartados em determinado momento. Esse descarte pode ser executado numa situação de congestionamento ou com finalidade de prevenir essa situação (KESHAV, 1997). Conquanto, no caso mais básico, os pacotes podem ser descartados sem sequer serem inseridos na fila de espera do roteador.

Os mecanismos de descarte diferenciam-se pela ordem na fila de espera do pacote a descartar e pela política que define o descarte dos pacotes. A situação mais simples de

implantar é o descarte do último pacote recebido. Desta forma, assim que um pacote entra no sistema, é verificado as seguintes condições:

- Se a fila de espera tem espaço suficiente para o recebimento e se ela pode ser aceita conforme estratégia definida. Caso não corresponda em ambas às condições o pacote é descartado.

4 MULTI PROTOCOL LABEL SWITCHING-MPLS

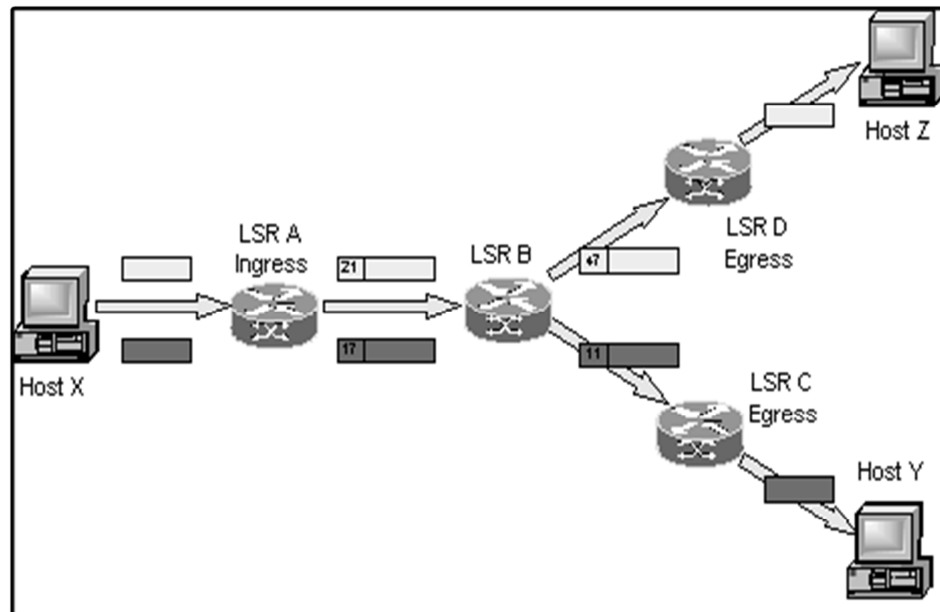
4.1 CONCEITOS FUNDAMENTAIS SOBRE MPLS

Com a popularização da internet e consequente ampliação da procura por maiores bandas e por novos serviços que requerem garantias como confiança e qualidade surgem à necessidade de uma implantação tecnológica que suporte tal demanda. Neste capítulo serão conceituados os fundamentos de uma tecnologia, o MPLS, e seus elementares componentes, tais como o *Forwarding Equivalence Class* (FEC), *Label Swithed Path* (LSP), *Label Edge Routers* (LER), *Label Switching Routers* (LSR), rótulos, bem como suas formas de execução.

O MPLS é um padrão que foi desenvolvido pela *Internet Engineering Task Force* (IETF) baseando-se em alguns experimentos semelhantes, porém proprietárias, em soluções de comutação. Esta tecnologia torna praticável a mudança das redes IP atuais, alicerçadas no paradigma de "melhor esforço", para redes de múltiplos serviços IP, formando então o alicerce de novas redes públicas em um futuro próximo.

Uma FEC é uma classe equivalente de encaminhamento, sendo um dos principais fundamentos do MPLS (ROSEN; VISWANATHAN; CALLON, 2001) que determina o caminho dos pacotes pelo uso de rótulos. Para uma FEC específica, a conexão de caminhos comutados, forma um caminho unidirecional através da rede, denominado como caminho comutado por rótulo LSP, pacotes que fazem parte de uma FEC comum, seguirá sempre a mesma rota através do domínio MPLS. O LSP baseia-se em saltos comutados por rótulo *Label Switched Hops* (LSH) entre pares de LSRs.

A arquitetura MPLS utiliza de dois tipos principais de roteadores, LSR que apresenta somente portas unicamente MPLS, que permite encaminhar o fluxo fundamentado exclusivamente em rótulos, executa somente na camada 2. Para lidar com os pacotes originados nas redes IP existe o LER, podendo ser de ingresso ou egresso na rede MPLS, dependendo se este está acessando ou saindo do domínio MPLS. Cada LSR e LER guardam um banco de informação de rótulos denominado como *Label Information Base* (LIB). Esta base é utilizada para proporcionar encaminhamento aos pacotes. Na figura 23 é apresentada uma visão em bloco da estrutura MPLS. O fluxo do host de origem x é ingressado no domínio MPLS e atribuído rótulos de acordo com LSP do destino. Estes rótulos são alterados na saída de cada roteador intermediário até alcançar o LSR de egresso, definido pelo LSP, que encaminhará o pacote ao host de destino Z ou Y.

Figura 23 - Estrutura MPLS.

Fonte: (MONTEBELLER, 2006).

A colaboração do MPLS é a viabilidade de introduzir a regra de conexão orientada nas redes IP. Os pacotes devem ser categorizados em fluxos ou categorias, de forma agrupada estes pacotes devem ter o mesmo modo de tratamento. Um grupo deste tipo é denominado FEC em MPLS. Uma FEC pode ser descrita como um atributo de pacotes que serão transmitidos da mesma forma, isto é, sobre a mesma rota com tratamento de envio de forma equivalente. Todos os pacotes que fazem parte da mesma FEC passarão pela mesma rota no domínio MPLS (SANTOS, 2003).

A partir do momento em que um pacote seja atribuído para uma FEC, ele é rotulado com uma identificação única da FEC à qual o pacote faz parte. O mapeamento de um rótulo para uma FEC é conceituado como vinculação de rótulo. Cada LSR vincula um rótulo a todas as FECs que ele aprendeu.

As FECs são vinculadas de modo direto para os LSPs. Uma FEC é elucidada como um conjunto de pacotes de nível três, que podem ser transportados da mesma forma podendo abranger o tráfego para um destino particular com requisitos de tráfegos distintos. No MPLS uma FEC é associada a um pacote IP uma única vez, no roteador de entrada da rede MPLS, o LER de ingresso.

O mapeamento de pacotes IP para a FEC requer identificar o usuário que está encaminhando o pacote. Deste modo é possível implantar os filtros com base no endereço de origem e de destino, porta de entrada, entre outras formas. Sendo assim o MPLS pode

fornecer um método eficaz de vinculação de rótulo com QoS, relacionado a um pacote específico.

Um percurso comutado por rótulo LSP são atributos dos LSRs pelo qual os pacotes que fazem parte de uma mesma FEC transitam com a finalidade de chegar a seus destinos. Ele também pode ser definido como a união de um ou mais saltos de comutação de rótulos, proporcionando a um pacote ser encaminhado pela troca de rótulo de um LSR para o outro.

Um pacote quando rotulado é transmitido dentro da rede baseado na informação deste rótulo, com isto o cabeçalho da camada três do modelo *Open Systems Interconnection* (OSI) não é aberto, analisado e utilizado, isto melhora a rapidez do encaminhamento do pacote, o que não é possível utilizando somente o cabeçalho da rede IP. Especialistas classificam o MPLS como um protocolo de camada dois e meio, por estar entre a camada de enlace e redes do modelo.

Os roteadores que adicionam rótulos na borda da rede, são chamados LER ou *edge LSR* (DAVIE et al., 2001), estão instalados nas bordas das redes onde manuseiam funções convencionais de roteamento e proporcionam conectividade aos usuários da rede. Os LERs observam e classificam os pacotes IP que acessam a rede e inserem rótulos que orientam sobre qual rota o pacote deve seguir. Quando um pacote não marcado entra no domínio MPLS e acessa em uma interface não MPLS, o LER o vincula a uma FEC e anexa o rótulo correspondente ao pacote. Na outra borda, quando um pacote rotulado está para sair do domínio MPLS o LER exclui o rótulo do pacote e continua a transmissão por um processo normal da camada três.

Os LERs são denominados de LSRs de ingresso (*ingress LSRs*) e LSRs de egresso (*egress LSRs*). O *ingress LSR* é o LSR que verifica os pacotes que estão chegando e insere rótulos, que serão usados dentro do domínio MPLS. Já o *egress LSR* retira os rótulos dos pacotes que estão saindo do domínio MPLS, entregando ao destino da conexão.

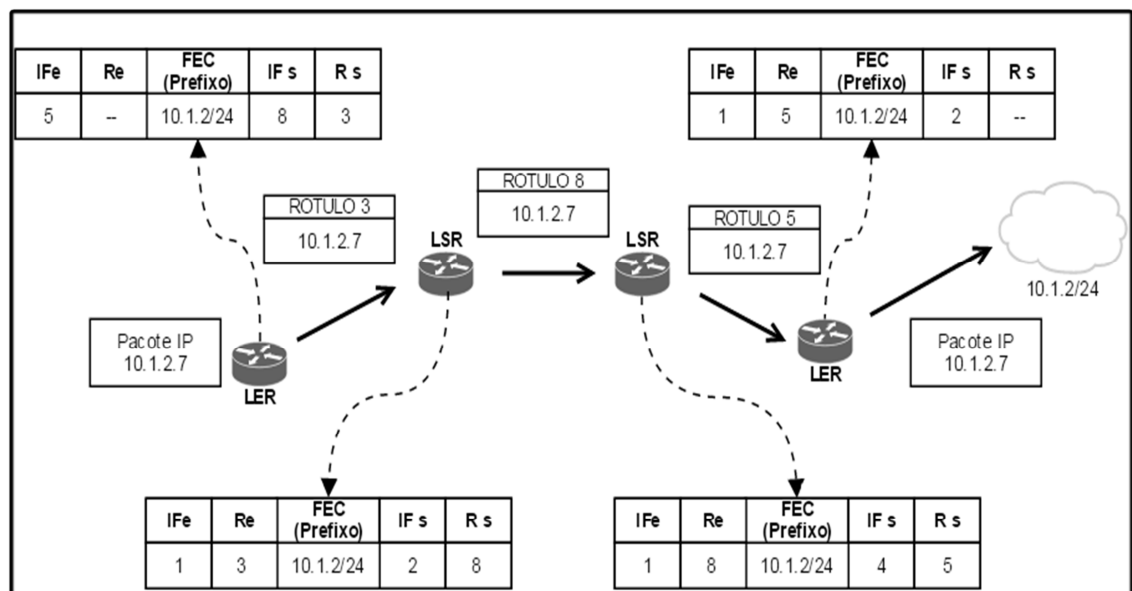
Os roteadores de comutação LSR estão instalados no núcleo das redes comutação por rótulos. Os LSRs transportam pacotes rotulados, durante a rota compreendida entre o LER de ingresso e o LER de egresso.

A forma de operação do MPLS difere da forma de operação do IP, pois os pacotes são verificados na camada três somente quando acessam ou quando são retirados do domínio MPLS, esta função é executada pelo LER que além de verificar o cabeçalho IP faz a associação do rótulo com a FEC de destino; daí em diante o pacote é encaminhado através da rede pelos LSR que manejam somente rótulos, como é apresentado na Figura 24. Conforme exemplo, o pacote IP ao entrar na interface de entrada-IFe do LER com destino ao IP 10.1.2.7,

analisa a tabela FEC e vincula o rótulo específico de acordo com a rede de destino, que no exemplo é 5. O LER altera na interface de saída-IFs o rótulo para 3, que é encaminhado ao próximo salto, chegando aos LSRs intermediários. Este altera o rótulo na IFs e assim sucessivamente até chegar a IFe do LER de egresso, responsável por remover o rótulo e encaminhar através de informações do cabeçalho IPv4 até o *host* de destino.

O MPLS aprimora os serviços que podem ser suportados pelas redes IPs, adiciona propósitos de engenharia de tráfego e simplifica o fornecimento de qualidade de serviço QoS. Como o MPLS provê elementos de gerenciamento de tráfego e mecanismos eficazes de

Figura 24 - Exemplo de operação do MPLS.



Fonte: (RUELA, 2006).

roteamento explícito, pode oferecer uma base de sustentação eficaz aos novos serviços.

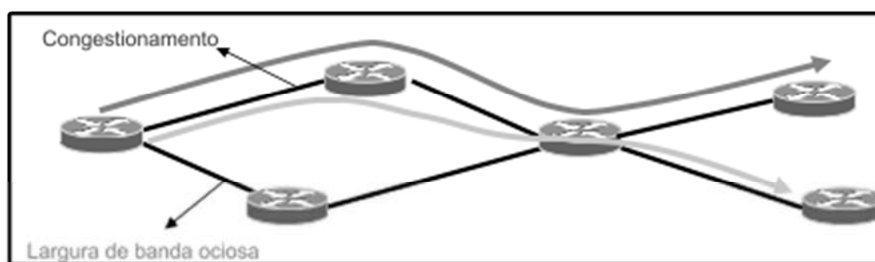
Os protocolos de roteamento usados nas redes IP são em sua maioria fundamentada em algoritmos que escolhem o caminho mais curto entre dois nós de rede distintos. Entretanto, as redes fundamentadas em MPLS oferecem roteamento explícito, que proporciona a um gerente de rede definir explicitamente a rota por onde os pacotes serão transmitidos na rede. Como por exemplo, é possível determinar uma rota baseado na disponibilidade de largura de banda, ou ainda, determinar caminhos redundantes para uso em momentos de falhas de enlace. Esta aptidão de oferecer roteamento explícito transforma o MPLS em um excelente instrumento para oferecer engenharia de tráfego *Traffic Engineering* (TE).

A possibilidade de um gerente de rede em fornecer engenharia de tráfego permite um melhor controle sobre os recursos da rede, por conseguinte potencializa a sua função. Como outro exemplo, pode-se fornecer engenharia de tráfego com finalidade de distribuir o tráfego

na rede de forma semelhante entre os caminhos possíveis, proporcionando utilização uniforme dos recursos da rede e minimizando situações de congestionamento.

Na Figura 25, ilustra-se uma rede MPLS sem TE, onde todos os tipos de fluxo originado são encaminhados através de uma única rota, congestionando o tráfego nos LSRs intermediários, sem avaliar o nível de congestionamento de outras rotas de encaminhamento que pode ter largura de banda ociosa, ao contrário do cenário encontrado na Figura 26.

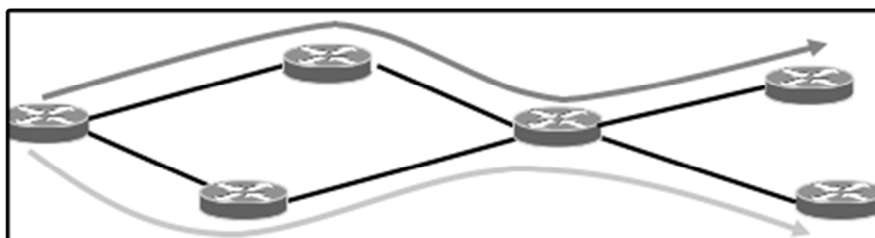
Figura 25 - Rede MPLS sem TE.



Fonte: (JOSÉ, 2012a).

Na Figura 26, é apresentada uma rede MPLS com TE, onde o administrador do domínio MPLS aplicou roteamento explícito, balanceando o tráfego em rotas diferentes, o que evita o congestionamento em apenas um enlace.

Figura 26 - Rede MPLS com TE.



Fonte: (SANTOS, 2003).

5 O EMPREGO DE TÉCNICAS DE QUALIDADE DE SERVIÇO EM REDES SEM FIO APLICADO A AUTOMAÇÃO DE PROCESSOS AMBIENTAIS

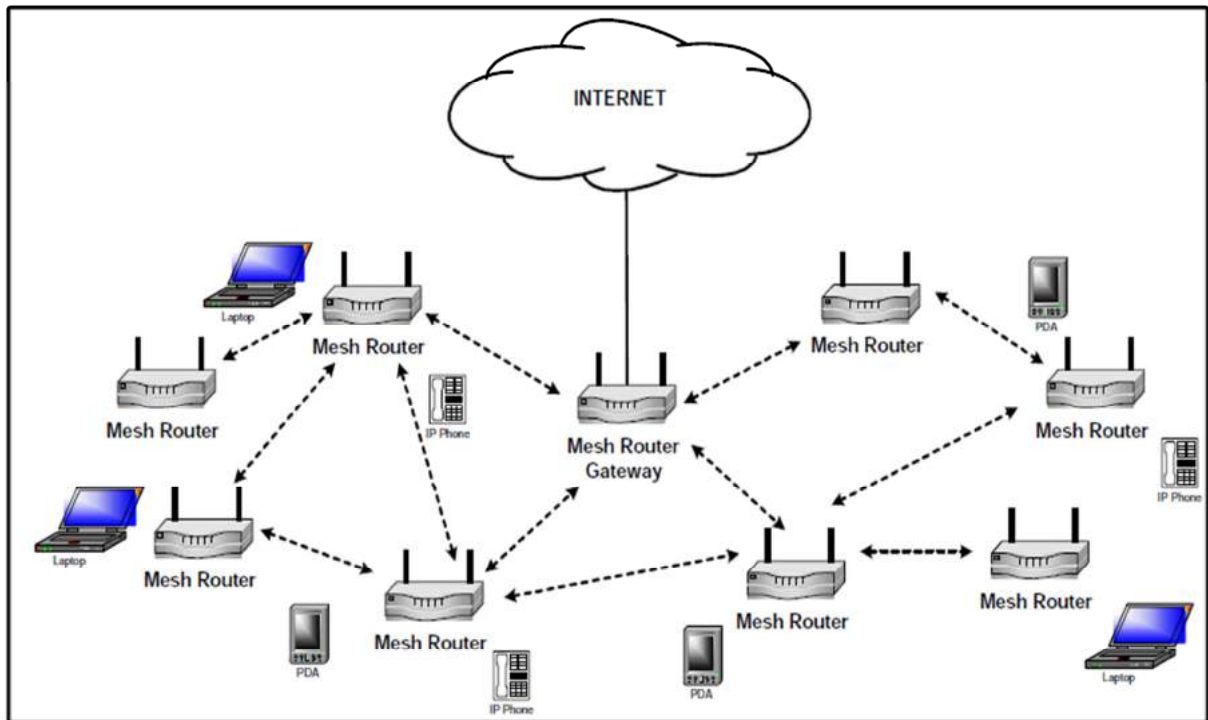
A praticidade, mobilidade e controle são termos que estão se tornando presentes nas vidas das pessoas, exigindo que processos sejam mais eficientes e eficazes. A Tecnologia da Informação deixa de ser vista como um custo ao processo, e passa a gerar valores aos negócios, interligando não só computador como também coisas. Na atualidade não é admissível uma empresa que almeje ser competitiva sem se utilizar de ferramentas tecnológicas, abrindo lacunas para que os novos gestores visionários busquem o aperfeiçoamento contínuo para sua empresa. (BAZZOTI et al., 2007).

Tecnologias emergentes são criadas, impulsionadas pelo crescimento contínuo de usuários e dispositivos instalados nos mais diversos ambientes. Destacam-se os sensores inteligentes que se comunicam entre si, com objetivo de solucionar problemas em áreas específicas, como em indústrias, monitoramento ambiental de vulcões, monitoramento de saúde pessoal, rastreamento de objetos e pessoas e medição de energia (LEITÃO, 2012).

Uma multiplicidade de Sensores, geograficamente distribuídos, encaminham dados de controle e informação, provenientes dos meios que estão atuando, até concentradores. Estes convertem os pacotes que entram nas interfaces, para o protocolo padrão de comunicação da interface de saída. Uma ampla variedade de meios físicos pode ser utilizada para reencaminhar os pacotes até os Sistemas Supervisórios.

Esta proposta utiliza redes sem fio em malha (*Mesh*), que utiliza roteamento dinâmico, topologia variável e de crescimento orgânico, constituídas por nós cuja comunicação, através de variantes dos padrões IEEE 802.11 e 802.16 no nível físico, permitem mais eficiência no transporte dos pacotes (ABELÉM, 2007). Este sistema encaminha os dados entre o coordenador da rede de sensores até o supervisor. Uma topologia usual de rede é apresentada na Figura 27, destacando-se a possibilidade de utilização de diversos tipos de clientes, desde computadores a telefones inteligentes.

Figura 27 - Exemplos de Redes Mesh.



Fonte: Adaptado (ABELÉM, 2007).

Os sistemas supervisórios permitem que sejam monitoradas e rastreadas informações de um processo produtivo ou instalação física. Tais informações são coletadas através de equipamentos de aquisição de dados e, em seguida, manipuladas, analisadas, armazenadas e posteriormente, apresentadas ao usuário. Estes sistemas também são chamados de SCADA.(SILVA; SALVADOR, 2005, p. 1).

É necessário garantir o transporte dos dados pela rede de comunicação. A falha na entrega pode causar impacto significativo à função do SCADA. Por exemplo, dispositivos gerenciando um sistema de caldeira, que gera energia térmica para um ambiente industrial. Qualquer atraso de resposta em qualquer nó pode gerar aumento ou redução da temperatura, e gerar impacto na potência das turbinas de geração de energia. Os Sensores espalhados neste ambiente devem coletar e encaminhar os dados com menor atraso (*delay*) possível.

O grande volume de dados gerado por estes dispositivos deve ser trafegado em meios confiáveis, porém, devido à falta de meios de acesso dedicado a este serviço, alguns optam por usar a infraestrutura existente, que enviam e recebem dados de diversos protocolos, alguns destes consomem muito espaço de banda, gerando atrasos e/ou perdas nas entregas de pacotes de serviços sensíveis. Níveis de prioridade devem ser considerados para cada um dos pedidos em trânsito, a ser enviado a partir de uma rede de sensores. O foco principal deste trabalho é

combinar técnicas de qualidade de serviço (QoS), investigações de mecanismos de priorização e controle de tráfegos e aplicar o método proposto em ambiente de Redes *Mesh*, que transportará os dados provenientes de redes inteligentes.

O ambiente apresentado na figura 28 é composto por quatro pontos de acessos sem fio espalhados dentro de um ambiente, interligados via *Wireless Distribution System* (WDS), formando uma malha *mesh*. Os pontos de acesso do fabricante Mikrotik, estão configurados e gerenciado pelo protocolo *Mesh, Hybrid Wireless Mesh Protocol* (HWMP)/IEEE 802.11s, que usa um conjunto comum de primitivas de protocolo, geração e regras de processamento retirados do *Ad Hoc On Demand* (AODV), (PERKINS; BELDING-ROYER; DAS, 2003), adaptado para camada 2. Executada sobre hardware modelo RB433, com *Internetwork Operating System* (IOS) na versão 6.15. Estes equipamentos foram adquiridos previamente pela instituição para serem empregados na distribuição de acesso à Internet aos docentes e discentes. Por ser um ambiente que provê acesso móvel, a segurança *Wifi Protected Access Enterprise* (WPA2) está integrada. O cenário foi escolhido devido à proximidade de uma reserva ambiental, que o tornou propício à instalação de sensores, facilidade de integração a rede *mesh* e de ter uma infraestrutura para instalação do supervisor.

Figura 28 - Visão Panorâmica do IFMT.



Fonte: Autoria própria.

Alguns trabalhos foram propostos, em Yaghmaee (2012), os autores propuseram um mecanismo de recuperação de falha em um link ativo e explica detalhes de um esquema baseado em prioridade para superar a parada neste LSP. Consideram uma rede IP/MPLS como

uma infraestrutura de uma rede inteligente, dá níveis de prioridade e um QoS adequado a todos solicitações de entrada em um domínio MPLS.

Em Zolhavarieh e Barati (2013), discute-se sobre o conceito de QoS em redes de sensores sem fio e diferentes métodos para melhorar a segurança da rede de dados. Comenta sobre os métodos de controle de tráfego, DiffServ, IntServ, MPLS, RSVP e TE. Enfatiza que largura de banda não é um grande problema para um nó sensor, mas pode ser problemático para um grupo de sensores e define alguns desafios de Qualidade de Serviços em Redes de Sensores.

Em Pozzuoli (2003), discute sobre a necessidade de perda mínima de pacotes dentro de uma rede de subestação, descreve o problema da falta de padronização aberta nos dispositivos eletrônicos inteligentes e a convergência entre fornecedores através de conversores, que dificultam e deixam mais caros os dispositivos e aumentam o atraso na conversão.

Em Bertocchi (2004), é feita uma investigação de desempenho do (UDP) e (TCP) para eficiência no consumo de energia dos nós uma rede, demonstra as características do protocolo de roteamento (DSR), demonstra os algoritmos utilizados para escolha de melhor caminho da origem até o destino, os envios *Route Request* (RREQ) e *Route Reply* (RREP), *Route Error* (RERR). Comenta sobre os problemas em relação aos dispositivos móveis, que alteram a cada momento sua localização.

Um grande conjunto de protocolos trafega por uma rede pública, neste caso, a rede acadêmica. A quantidade de pacotes inseridos nestes segmentos, se não forem gerenciados, podem gerar gargalo no enlace, pela utilização de 100% da largura de banda, visto que não reserva banda para serviços essenciais, como é o caso dos sensores. A necessidade de sincronismo da comunicação entre os sensores de campo e o nó central, é crucial. Qualquer falha neste processo, dependendo da criticidade da função do dispositivo de sensoriamento e da janela de indisponibilidade aceitável, pode causar a ineficiência e inviabilizar aquele monitoramento dinâmico.

Percebe-se a importância do tratamento da informação, principalmente quando trafega em meios não tão confiáveis, no sentido do transporte. A utilização de redes sem fio implica em sérios riscos de segurança, que podem ser explorados e podem causar indisponibilidade de serviços.

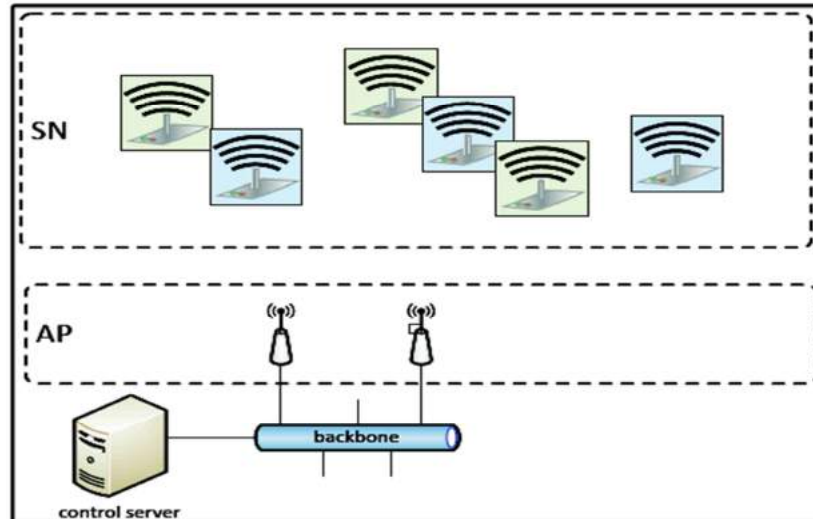
Segundo Vilela (2012), “as falhas existentes nas ementas de segurança do padrão IEEE 802.11 são evidentes. Isso mostra a necessidade em agregar outro mecanismo de segurança nas WLAN para minimizar as ameaças que exploram estas vulnerabilidades”. Outros fatores podem ser citados, como interferência de sinais, causado por canais sobrepostos e

sobrecarregados, números excessivos de usuários conectados a um mesmo AP, baixa qualidade dos enlaces WDS e nas associações entre os clientes e pontos concentradores.

É um desafio, considerando as inseguranças do IEEE 802.11, o uso de redes de sensores sobre redes WLAN, pois os dados *sensíveis* ao atraso devem ser entregues ao destinatário na sequência correta, pois se isto não ocorrer, eles podem ser descartados, que exigirá a retransmissão dos mesmos, aumentando desnecessariamente o consumo da banda disponível. Porém, é interessante investigar a utilização de uma rede em malha sem fio, afinal, existem mecanismos que podem ser empregados neste cenário, por exemplo, o roteamento dinâmico, que na ocorrência de problemas com alguns enlaces, podem-se utilizar outros nós para o encaminhamento dos pacotes, da rede de sensores até o supervisor.

A infraestrutura proposta inicialmente é mostrada na Figura 29. Nesta o tráfego gerado na rede de sensores-SN seria transportado na rede sem fio. Na última milha, existiriam sensores *Zigbee*, coletando informações de temperatura e fumaça, na reserva ambiental do Parque Massaro Okamura, onde está localizado o (IFMT) – Campus Cuiabá Bela Vista, que tem dentro dos seus Cursos, o tecnólogo ambiental.

Figura 29 - Visão de alto nível do modelo de RSSF.



Fonte: Adaptado (SVEN JÄGER et al., 2012).

Os equipamentos foram configurados para permitir o encaminhamento de pacotes oriundos da rede de sensores para os softwares supervisórios. Os pontos de acesso Mikrotik, distribuídos no Campus, interconectam com os controladores das redes de sensores pela interface sem fio. Após a entrada no domínio *WLAN Mesh*, os dados são etiquetados pelo MPLS encaminhados até o roteador *Mesh*, chamado de Portal, que está ligado via interface metálica (*ethernet*) a um *Switch* com *Virtual LAN (VLAN)*, segmentando a rede e isolando os

dados que seriam entregues ao *Software* SCADA, responsável pela coleta, armazenamento e retroalimentação de dados para os sensores.

6 O EMPREGO DE TÉCNICAS DE QUALIDADE DE SERVIÇO EM REDES SEM FIO APLICADO A AUTOMAÇÃO PREDIAL

As redes sem fio são bases de comunicação de inúmeros dispositivos encontrados nos estabelecimentos corporativos, como universidades, museus, *shoppings*, dentre outros. Um dos principais fatores que levaram a este crescimento é o aumento do número de usuários conectados, utilizando uma gama de dispositivos, como os *tablets*, *smartphones* e computadores, pela mobilidade e facilidade de conexão das redes sem fio.

Entretanto, estes múltiplos dispositivos são utilizados por diversos serviços heterogêneos que inundam estas redes de pacotes com fluxos tomadores de banda. Portanto, quanto maior for a capacidade efetiva de encaminhamento de pacotes e controle de admissão, maior será a qualidade do serviço entregue pela rede sem fio às aplicações ou serviços demandantes. Isto proporciona com eficácia o alcance do objetivo esperado pelo usuário ou pelo sistema.

A evolução dos meios de comunicação, em especial às redes sem fio, vem contribuir no desenvolvimento de novas tecnologias de integração que chegam às empresas e residências e levam conforto e comodidade a seus usuários. Além do vasto uso na comunicação de dados entre os usuários, podem ser ampliadas e compartilhadas em automação predial. Pois permitem que equipamentos controladores possam ser instalados e dispostos em locais onde o sistema de cabeamento não alcança.

As redes principais de um edifício são planejadas para terem o máximo de desempenho, isto garante a capacidade de escalonamento de serviços instalados nos cliente destas conexões. Porém, nem sempre, prover soluções de cabeamento de alta velocidade é possível, pelas limitações impostas pela infraestrutura dos prédios e dos custos de instalação de tecnologia com fio.

Muitos imóveis foram construídos sem sequer um planejamento de instalação de infraestrutura de passagens de cabos lógicos ou mesmo projetado de forma subdimensionada, reduzindo a capacidade de adição de novos fios, o que dificulta e onera a instalação de um sistema de cabeamento estruturado. Outro caso, é que em prédios históricos esta instalação não é possível, em virtude da necessidade de garantir a originalidade do imóvel.

A ocorrência de falhas em uma rede, muitas vezes significa o fracasso de um sistema de automação, principalmente se este estiver sobre uma rede sem fio, pois este tipo de conectividade é mais vulnerável a interferência do meio, quando comparadas a outras tecnologias. Uma falha de comunicação ou perda de sinal poderia causar sérios danos em

equipamentos de natureza crítica, como em sensores que atuam sobre processos críticos de negócio.

Porém, estas possibilidades de falhas dos dispositivos de automação podem ser mitigadas, ao aplicar técnicas de QoS ou engenharia de tráfego sobre a rede principal do sistema. Esta ação proporciona ao sistema diminuição na latência e perda de pacotes, e aumenta a disponibilidade do sistema de comunicação entre os controlares de campo/sistema até a central de operações.

O objetivo central deste trabalho é avaliar o emprego de redes sem fio IEEE 802.11, como rede principal em um sistema de automação predial. Principalmente quando esta rede está sobrecarregada com dados de outras aplicações, a exemplo de fluxos multimídia, que consomem banda em rajadas, ocupando todo o canal durante a transmissão. Este cenário é encontrado na maioria dos prédios que não foram planejados sobre o conceito de edifícios inteligentes, e que compartilham a rede sem fio com o tráfego internos das organizações e os fluxos do sistema de automação predial. Torna-se necessário avaliar qual a capacidade da rede sem fio em proporcionar um meio de transportar dados compartilhados, sem degradar o sistema de automação predial.

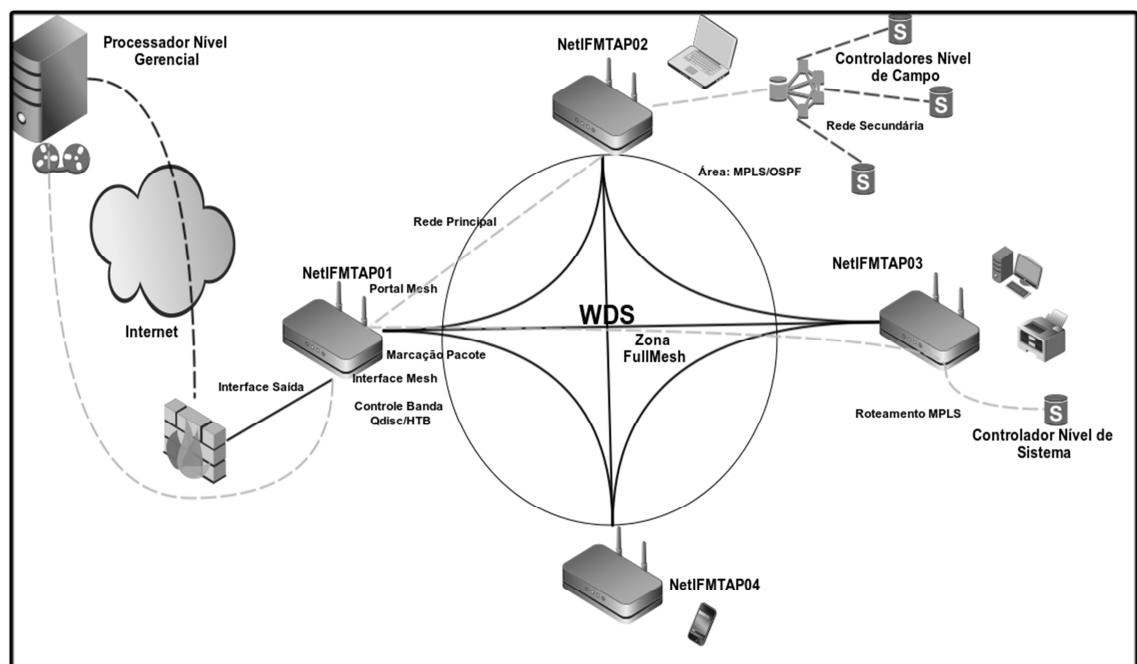
Em edifício inteligente devem ser previstos equipamentos e espaços físicos utilizados para a automação de seus recursos. Os projetistas devem ter em mente quais tecnologias serão utilizadas e qual o espaço físico disponível para o uso dessas tecnologias. Dessa forma, os edifícios inteligentes devem ser capazes de se adaptarem às futuras tecnologias sem que haja uma modificação profunda em sua estrutura.

Porém, alguns prédios não foram planejados para prover espaços físicos capazes de suportarem crescimentos tecnológicos de forma estruturada, devido a diversos fatores, nas quais, pode-se destacar a falta de orçamento, deficiência na fiscalização de obras e projetos replicados de outras unidades com finalidade operacional divergente. Problemáticas encontradas no ambiente acadêmico do (IFMT) – Campus Cuiabá Bela Vista, que está localizado na região central da cidade de Cuiabá - MT e atende a 1200 alunos de ensino médio integrado e superior conforme demonstrado na Figura 30, cuja única solução de conexão instalada é uma rede sem fio, com finalidade de prover acesso à internet à comunidade acadêmica e o uso como rede principal do sistema de automação, que visa a sustentabilidade.

Os usuários conectam-se através de dispositivos heterogêneos, tais como: *tablets*, *smartphones* e computadores pessoais móveis. Estes tem acesso ilimitado à banda de conexão com a internet, sem regras de bloqueio sobre quantidade de fluxos por protocolos e serviços.

Os rádios sem fio, ou roteadores de borda MPLS, foram configurados para permitir o encaminhamento de pacotes oriundos da rede de sensores, através dos controladores em nível de sistema ou diretamente pelos controladores de campo, com destino ao software supervisor em nível gerencial. Portanto, é essencial garantir a entrega dos pacotes com maior confiabilidade possível, principalmente quando se trata de um domínio sem fio compartilhado por diversos serviços, a destacar aqueles que usam tráfego em rajadas, como FTP e fluxos multimídia.

Figura 30 - Topologia da rede primária e secundária.



Fonte: Autoria própria.

Para um bom funcionamento de qualquer sistema de controle é necessário que os sensores e controladores sejam escolhidos e instalados adequadamente. Todo o mapeamento do processo de produção pode ficar comprometido caso esses elementos da automação sejam relegados á segundo plano (BOARETTO, 2005).

Para função de controladores de sistemas ou campo, foi utilizada a plataforma Arduino, um microcontrolador embarcado com software livre. A escolha pelo uso da plataforma Arduino está baseada em diversos fatores: facilidade na utilização, custo benefício, multiplataforma, interface de programação, não utilização das portas seriais, hardware e software que são de fontes abertas, bibliotecas de desenvolvimento, *bootloader* e *shields* (MOREIRA; PORTELA; SILVA, 2013), além de ser compatível com protocolo Modbus/TCP (Modbus, 2004).

Visando mitigar as interferência entre rádios localizados na periferia do domínio DiffServ e equilibrar cargas de processamentos nos APs, foi atribuído uso de canais de conexão com menor quantidade de dispositivos ocupantes e menor ruído de comunicação. Os APs Mikrotik formaram uma malha de conexão ponto-multiponto entre si, cuja finalidade é permitir múltiplos caminhos entre a origem e o destino, garantindo failover ao sistema em caso de inoperabilidade de um nó específico. Os clientes finais não conectam diretamente ao domínio, pois foi instalado na interface com fio de cada nó de borda, rádios APs para servir a conexão nas zonas sem fio a estes clientes, contribuindo para o balanceamento da carga de processamento dos rádios Mikrotik.

O foco deste trabalho foi a área de automação predial. O conceito de QoS está relacionado ao uso otimizado dos recursos de rede, envolvendo uma classificação criteriosa para priorização de determinados fluxos, a racionalização do uso da banda e o uso de restrições que garantam que os serviços prioritários estejam dentro dos limites aceitáveis de operação. Assim, qualidade de serviço é uma conjunção de equipamentos e tecnologias que possibilitam às redes garantirem desempenho às aplicações.

A pretensão não foi caracterizar todo o tráfego que entra ou sai de um determinado nó, o objetivo é permitir que determinados fluxos tivessem garantia de fluência nos momentos de congestionamentos. Desta maneira o trabalho de caracterização de tráfego está resumido à identificação das características dos fluxos responsáveis por manterem os serviços de automação predial e manter certo nível de QoS para estes. Como são conhecidas as características dos fluxos prioritários, o trabalho de caracterização da carga de trabalho está resumido à observação do comportamento dos fluxos originados e recebidos por determinados *hosts* identificados por seus endereços, tipo de protocolo, porta de conexão e marcação DSCP.

Além da caracterização, será aplicado controle de fluxo com o uso do algoritmo de controle de carga HTB. Uma das vantagens do HTB como modelador de tráfego é a capacidade de partilhar a largura de banda entre os serviços. Este escalonador garante certa porção do enlace para cada classe, permitindo ainda especificar quanto pode ser emprestado entre as classes.

A implantação do HTB foi realizada na interface do *gateway* por onde os pacotes são encaminhados, após marcação do campo de DSCP, que permitiu identificar os fluxos distintos dos sistemas críticos no domínio DiffServ. Esta ação permite definir critérios de prioridade desde a origem dos pacotes na rede interna até a interface de entrada ou saída ao nó de destino.

A grande vantagem dos sensores sem fio é a versatilidade, o que possibilita aos sensores serem instalados em qualquer ponto do ambiente sem a necessidade de se colocar fios ou cabos. As redes principais e secundárias de um edifício são planejadas para terem máximo desempenho e a substituição dessas redes por redes sem fio pode não ser vantajosa, caso não sejam aplicadas técnicas de QoS e/ou engenharia de tráfego. Neste ambiente, a quantidade de controladores em automação instalados por zona sem fio deve ser planejada, devido a momentos de gargalo inserido por outras aplicações consumidoras de banda, que mesmo ao aplicar técnicas de controle de fluxo, podem ainda ocasionar muitas perdas de pacotes (*packetloss*) e *jitters* maiores do que os suportados pelas aplicações de controle e supervisão do sistema de automação, além de sobrecarregar o processador do dispositivo sem fio concentrador, causando travamento e/ou paralisação do equipamento.

6.1 PRIMEIRA FASE – IDENTIFICAÇÃO DO NÚMERO DE DISPOSITIVOS DE AUTOMAÇÃO SUPOSTOS POR UMA ZONA SEM FIO COMPARTILHADA

Na primeira fase deste trabalho é apresentado o resultado de uma pesquisa sobre a quantidade de dispositivos de automação que podem ser instalados em uma zona sem fio compartilhada, sem comprometer a entrega de pacotes dos dispositivos finais às aplicações de controle e supervisão. Para isto foram injetados pacotes de tamanho fixo de 1040 bytes pelo software IPERF, estes pacotes são similares ao fluxo de dados de sensores, sobre uma rede 802.11, com vazão limitada, por conseguinte, comparado às métricas de *jitters* e taxa de pacotes perdidos de cada interação, aplicado a cada momento, quantidades variáveis de processos até atingir o número máximo de dispositivos suportados pelo equipamento concentrador de rede sem fio, quando utilizadas políticas de controle de banda.

Existem diversas pesquisas sendo desenvolvidas para prover suporte para as redes sem fio transportarem dados sensíveis, porém poucas aplicadas à comunicação entre os controladores e o sistema supervisor. O foco principal desta parte foi combinar técnicas de qualidade de serviço, investigações de mecanismos de priorização e controle, e aplicar ao ambiente de redes sem fio, utilizado pelos alunos do (IFMT) – Campus Cuiabá Bela Vista, que compartilha os fluxos de dados com a rede de sensores da automação predial.

Neste cenário, a banda de ocupação do canal, com vazão foi limitada de 2,6Mbps, padrão de velocidade de acesso à Internet encontrada em muitas organizações brasileiras, segundo o ranking da akamai (THOMPSON et al., 2014), que estava sobrecarregada pelo tráfego em rajada gerado pelos alunos. Buscava-se garantir que os serviços de comunicação entre os sensores e a aplicação do SCADA, fosse executado com variação de atraso entre 1 à 100ms,

características intrínsecas das aplicações críticas, com ênfase em baixo consumo, baixo custo, baixa latência, pacotes pequenos e baixa razão de dados (SANTOS, 2007).

A tabela Mangle foi utilizada para marcar ou alterar o campo no cabeçalho do protocolo IP, através de filtros com regras diferentes, por tipo de serviços, endereços IP de *hosts* ou até mesmos endereços de rede. Este procedimento é conhecido como processo de classificação. Sendo assim, foi possível diferenciar os tipos de tráfego e prioridade especificada pelo administrador do sistema firewall.

Foi empregado o agendador HTB que está embutido nos principais rádios sem fio, em especial ao rádio Mikrotik que compõe o ambiente analisado. Este agendador foi capaz de maximizar a largura de banda utilizada, aprimorando a qualidade de serviço oferecida. A configuração do HTB foi realizada na interface do gateway para Internet, em que os pacotes são encaminhados e na interface de acesso em cada rádio sem fio da célula, sendo responsável pela organização dos fluxos das filas de entrada e saída. A vantagem do HTB como modelador de tráfego é a capacidade de partilhar a largura de banda entre os serviços. Este escalonador garante certa porção do enlace para cada classe, e especifica quanto pode ser emprestado entre as classes.

Na Figura 31, é demonstrada a configuração de controle de banda, nos pacotes de *downloads* marcados pela tabela Mangle e suas respectivas filas controladas pelo HTB, bem como a definição das prioridades de entrega para cada fila, em que 1 é prioridade maior e 8 é prioridade menor.

Figura 31- Controle de Banda por HTB.

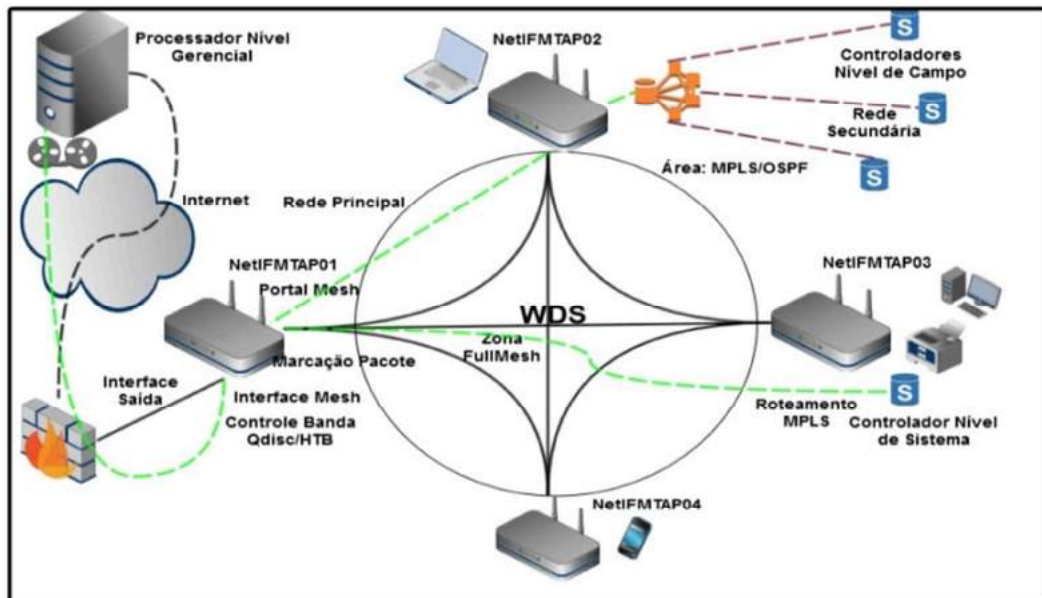
Name	Parent	Priority	Limit A...	Max Limit...	Avg. Rate	Queued Bytes	Bytes	Packets
Download	global	8	2500k	3M	634.1 kbps	0 B	721.7 ...	530 573
http_out	Download	2	1024k	1500k	651.8 kbps	35.2 KiB	226.0 ...	163 505
ssl_out	Download	2	512k	1024k	38.5 kbps	0 B	9.3 MiB	14 384
gov.b...	Download	2	512k	1024k	688 bps	0 B	631.3 KiB	5 830
ftp_out	Download	2	64k	128k	0 bps	0 B	483.3 ...	337 928
faceb...	Download	8	64k	128k	4.9 kbps	0 B	351.7 KiB	1 212
skype...	Download	5	64k	128k	0 bps	0 B	144.1 KiB	1 259
skype...	Download	5	64k	128k	120 bps	0 B	65.6 KiB	970
vnc_out	Download	5	64k	128k	0 bps	0 B	6.1 KiB	109

Fonte: Autoria própria.

O cenário empregado nesta fase da pesquisa trata-se de uma rede real e sua topologia é apresentada na Figura 32, é composta por rádios sem fio espalhados dentro de um campus universitário que estão configurados sobre topologia em malha. O rádio NetIFMTBVAP01 é o portal *mesh* com acesso ao ambiente externo, com enlace limitado à 2,6Mbps, que permite

acesso ao servidor IPERF, que é responsável pelas coletas das métricas da qualidade de serviço. Na zona sem fio do rádio NetIFMTBVAP03 está o cliente IPERF, que inunda a rede com datagramas definido pelo administrador, nos ambientes com e sem controle de tráfego por HTB. O software IPERF (GATES, 2007) foi empregado para simular o uso da rede para transporte de dados oriundos da automação predial. A comunidade acadêmica utiliza o domínio sem fio com tráfego heterogêneo, muitos apresentam grande consumo de banda. As métricas sobre *jitters* e porcentagem de pacotes perdidos (*lost*) são coletadas e comparadas a cada iteração quando aplicadas quantidades diferentes de processos, com objetivo de definir o número de controladores que podem ser inseridos na rede.

Figura 32 - Cenário utilizado para instalação do controle de Banda.



Fonte: Autoria própria.

O pacote de software livre IPERF, modelo cliente/servidor, foi utilizado para testar o controle de banda, realizando injeção de pacotes UDP com finalidade de medir o desempenho de redes de sem fio. A Figura 33 demonstra os parâmetros adotados pelo servidor para geração e análise dos fluxos de dados.

Figura 33 - IPERF - Lado Servidor - Tráfego UDP.

```
root@pequi:~# iperf -s -u -l 1024 -f K -x cms -i 120
[ ID] Interval      Transfer    Bandwidth    Jitter  Lost/Total Datagrams
[ 3] 0.0-120.0 sec  783 KBytes  6.53 KBytes/sec  42.518 ms  232/ 1015 (23%)
[ 4] 0.0-120.0 sec  862 KBytes  7.18 KBytes/sec  28.613 ms  154/ 1016 (15%)
[ 7] 0.0-120.0 sec  881 KBytes  7.34 KBytes/sec  37.640 ms  135/ 1016 (13%)
[ 6] 0.0-120.0 sec  907 KBytes  7.56 KBytes/sec  24.590 ms  109/ 1016 (11%)
[ 5] 0.0-120.0 sec  893 KBytes  7.44 KBytes/sec  28.035 ms  122/ 1015 (12%)
```

Fonte: Autoria própria.

Nos casos de testes, foram analisados cenários com números de processos variáveis, com objetivo de identificar a quantidade de controladores de automação que são suportados por uma zona sem fio. A metodologia utilizada foi à inundação de pacotes UDP com tamanho fixo de 1040 bytes, cujo tamanho do datagrama é similar ao fluxo de um controlador de automação. Na tentativa de garantir *jitters* médios variando entre 1 à 10ms e perda máxima de pacotes (*packetlost*) menor ou igual a 5% (VIEGAS et al., 2012). Para isso, foi simulado o encaminhamento dos datagramas de um controlador até a aplicação de controle e supervisão, hospedada em um ambiente remoto. As coletas de dados foram feitas em momentos de congestionamento da rede por fluxos multimídia gerados de forma proposital, e aos pacotes injetados pelo cliente IPERF na zona NetIFMTAP03 foi definido prioridade de tráfego sobre o fluxo concorrente. O uso desta técnica garantiu o mesmo cenário de fluxo em momentos diferentes de coleta, o que proporcionou maior confiança destes dados.

Os valores coletados conforme Equações 1 e 2, em amostras individuais a cada 120s, durante o período total de 3600s, permitem identificar o *Jitter* Médio e Porcentagem Média de Tempos Perdidos, e compará-los à política definida como requisito de qualidade do serviço da rede principal do sistema de automação.

Equação 1: Considere o sistema que demonstra o *Jitter* Médio, em que $J=Jitter$ da Amostra e $TA=Tempo$ Amostra.

$$\frac{\sum J_n * TA_n}{\sum TA_n} \quad (1)$$

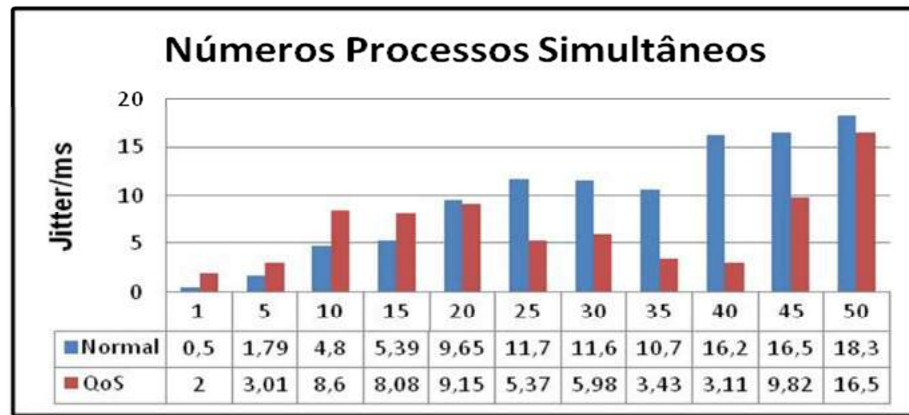
Equação 2: Considere o sistema que calcula a *Porcentagem Média de Pacotes Perdidos*, em que $PP=Pacotes$ Perdidos e $PT=Pacotes$ Transmitidos.

$$\left(\frac{\sum PP_n}{\sum PT_n} \right) * 100 \quad (2)$$

A Figura 34 apresenta os resultados obtidos, ao aplicar a cada iteração uma progressão aritmética na quantidade de processos inseridos, visando identificar o número de processos suportados no segmento de rede que atendam os parâmetros definidos de níveis de qualidade de serviço. Pode-se observar que sem o uso de QoS, foi possível utilizar apenas 20 processos simultâneos com dados semelhantes aos gerados por dispositivos finais de um sistema de automação, pois com 25 processos as métricas de *jitter* foram superiores á 10ms, métrica que extrapola a taxa de atraso permitida ao sistema. Enquanto que, ao aplicar QoS, foi possível empregar 45 processos e permanecer dentro dos critérios estabelecidos de qualidade. Portanto, tornou-se possível mensurar a quantidade de equipamentos de automação predial que

poderiam ser instalados com garantia de comunicação até o sistema em nível gerencial. Nas Figuras 34 e 35 não foram apresentadas informações em relação aos pacotes perdidos, devido ao fato de que em todos os casos apresentaram valores inferiores a 1%.

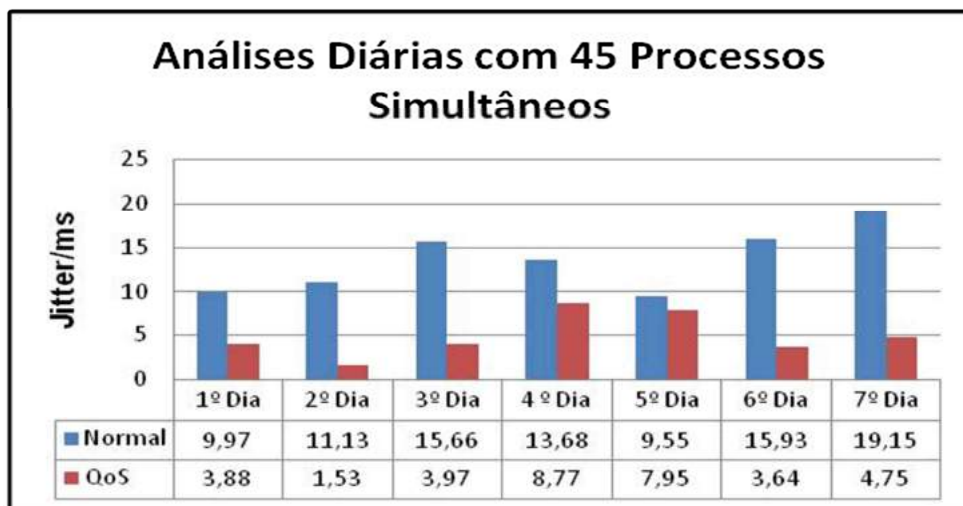
Figura 34 - Números de Processos Simultâneos.



Fonte: Autoria própria.

Na Figura 35 é apresentado os parâmetros encontrados durante o ciclo de 07 dias, aplicado a quantidade de 45 processos simultâneos identificados como possíveis de utilização na zona sem fio do rádio NetIFMTAP03 ao aplicar QoS. Com fluxo normal percebe-se que somente no primeiro e quinto dia os valores dos *jitters* ficaram inferiores à 10ms, enquanto nos ciclos com QoS em 100% dos casos os parâmetros ficaram dentro da limiar de qualidade, ou seja até 10ms.

Figura 35 - Análises Diárias com 45 Processos Simultâneos.



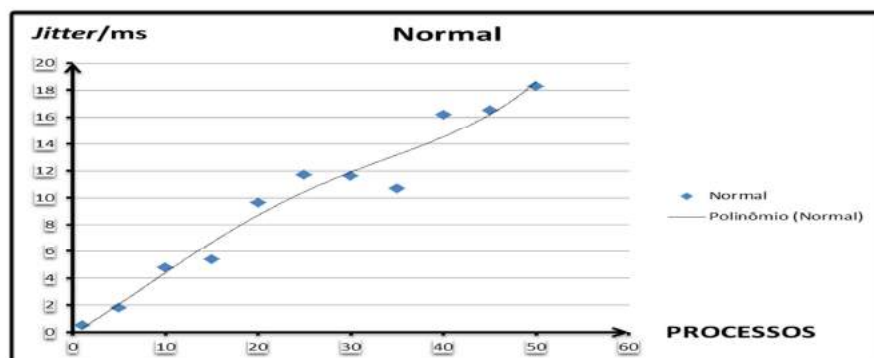
Fonte: Autoria própria.

Com base nos resultados encontrados, foi possível estimar a quantidade de dispositivos que podem ser instalado na zona de sem fio do mesmo domínio, pela aproximação dos dados obtidos da matriz *jitter* por número de processos para obtenção das equações 3, 4. As

Equações 3 e 4 são resultados da estimativa da curva através do método aproximação polinomial. Este método não linear é baseado em modelos não lineares de regressão em conjunto ao ferramental de mínimos quadrados, método de Newton e método de Lagrange, que busca uma curva mais suave na tentativa de interpolar um número maior de pontos, como apresentado por (LEAL, 2008).

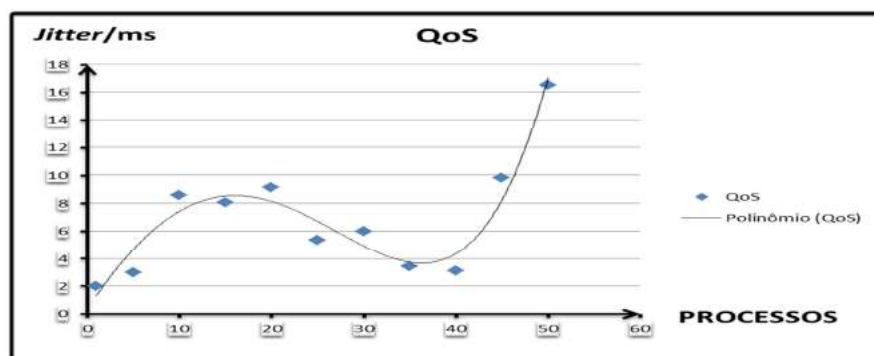
Os métodos de interpolação polinomiais são de grande utilidade na modelagem matemática de problemas reais, pois permitem um aumento do grau do polinômio com o propósito de atender um número maior de ponto da curva e um aumento significativo do grau de confiança do modelo obtido. Optou-se pelo uso de tal modelo em comparação a outros modelos não lineares, pois foi encontrado um coeficiente de correlação para os dois casos, com valores acima de 92%, atestando que o modelo é confiável e parte-se da hipótese que ao menos um trecho superior da curva chave obedece a equação, conforme apresentado nas Figuras 36 e 37.

Figura 36 - Interpolação Polinomial em Fluxo Normal de Pacotes.



Fonte: Autoria própria.

Figura 37 - Interpolação Polinomial - Fluxos com QoS.



Fonte: Autoria própria.

O experimento realizado produziu um conjunto de dados, e o ajuste de curva permitiu obter as equações 3 e 4 que têm um papel importante na estimativa quantitativa de dispositivos relacionados aos dados a serem inseridos na rede sem fio. Destaca-se que o uso

de um modelo matemático permite uma melhor exploração dos pontos não analisados no experimento, criando assim a possibilidade de uma previsão futura de resultados de *jitter* para um número de processos não explorados.

Equação 3: A Equação Polinomial que descreve o tráfego em Fluxo Normal, na qual x representa o número de processos testados nos experimentos da figura 34 e y os resultados encontrados dos *jitters* para a simulação normal com R-quadrado representando o grau de confiança da equação em 96,21%. O formato polinomial de quarto grau foi muito eficiente na interpolação dos pontos com uma ótima aproximação dos pontos acima da curva. Destaca-se que o ajuste de curva de quarto grau foi o que melhor representou os resultados obtidos no experimento.

$$y = 6 \times 10^{-6} x^4 - 0,0005 x^3 + 0,0103 x^2 + 0,39 x - 0,0836 \quad (3)$$

$$R^2 = 0,9621$$

Equação 4: A Equação Polinomial que descreve o tráfego em Fluxo QoS, na qual x representa o número de processos testados no experimento da figura 34 e y os resultados encontrados dos *jitters* para a simulação QoS com R-quadrado representando o grau de confiança da equação em 92,59%. O formato polinomial de quarto grau foi muito eficiente na interpolação dos pontos com uma ótima aproximação dos pontos acima da curva. Destacamos que o ajuste de curva de quarto grau foi o que melhor representou os resultados obtidos no experimento.

$$y = 2 \times 10^{-5} x^4 - 0,0006 x^3 - 0,0255 x^2 + 1,0081 x + 0,3272 \quad (4)$$

$$R^2 = 0,9259$$

6.2 SEGUNDA FASE – AVALIAÇÃO DO COMPORTAMENTO DA REDE DE AUTOMAÇÃO PREDIAL SOBRE ARQUITETURA DIFFSERV INTREGRADA AO ALGORITMO HTB EM CENÁRIOS REAIS

A Qualidade de Serviços proporcionada por uma rede de computadores deve estar diretamente relacionada às necessidades das aplicações. Estas primícias possuem maior relevância quando se utiliza serviços críticos como é o caso de sistemas de automação predial e residencial instalados em ambientes de rede sem fio, compartilhado por inúmeras aplicações consumidoras de banda de acesso. Esta arquitetura é encontrada em muitos prédios, principalmente devido às limitações estruturais ou pelo maior custo das soluções cabeadas, é um meio de comunicação eficaz na entrega de pacotes de dados originados nos atuadores ou controladores dos sistemas de automação, quando aplicadas técnicas de QoS. Porém, pode

apresentar problemas quando existe concorrência de aplicações para uso do meio compartilhado da rede.

Nesta fase é apresentado o resultado da pesquisa sobre o desempenho do sistema de automação frente às situações sofrida na rede principal que interliga a aplicação de supervisão central aos dispositivos de controle instalados na rede sistemas finais.

Alguns trabalhos foram apresentados, em Medeiros et al. (2014), que fazem uso das atuais estruturas de redes existentes nos vários hospitais e permite que aplicações de Telemedicina e Telessaúde funcionem de forma satisfatória nestes ambientes. Os autores apresentam uma abordagem baseada em políticas de QoS para assegurar as condições necessárias, a previsibilidade e diminuição de *jitter* (variação do atraso das mensagens) e melhor a eficiência dos serviços prestados.

Em Blanes et al. (2015), o potencial das redes sem fio é analisado em um cenário real e seu desempenho é comparado com um cenário ideal. Portanto, em ambos os casos, as prioridades comportaram-se como esperado, no entanto, em situações reais, algoritmos de controle de banda não foram eficientes em ambientes industriais, em que a mobilidade das fontes e da interferência, produzida por frequentes mudanças de taxa, ocasionou redução do desempenho da rede.

Estes trabalhos visam medir a eficiência de técnicas de QoS na rede principal ou secundária em ambientes de automação de processos, no entanto, o primeiro tem foco no uso de Virtual Lan (VLAN) e o segundo nas interferências sofridas pelas redes sem fio em ambientes industriais. O intuito da presente pesquisa foi realizar um estudo direcionado ao uso do DiffServ e HTB no controle de prioridades de tráfego em um domínio de rede sem fio compartilhado entre aplicações consumidoras de banda e sistemas críticos de automação predial.

O foco desta fase é combinar técnicas de qualidade de serviço, investigações de mecanismos de prioridade e controle, aplicado a um ambiente de redes sem fio em operação. Estas técnicas proporcionadas por um conjunto de protocolos, dentre os quais se destacam o DiffServ, que permite a diferenciação e tratamento dos pacotes críticos, que trafegam no domínio da rede através da marcação do campo DSCP em cada pacote desde o controlador sem fio de origem até o de destino.

Esta marcação oferece ao protocolo HTB a possibilidade de classificar o fluxo de entrada agrupando em fila única os pacotes com características idênticas, dando a estes níveis de prioridade de processamento e vazão na interface de saída do segmento de rede e consequente diminuição do número de pacotes e bytes perdidos na comunicação. Conforme apresenta a

Figura 38, os pacotes marcados com o código DSCP 31 e 36 ao chegarem à interface recebem tratamento diferenciado por ter nível de prioridade maior em relação aos pacotes padrões marcados com DSCP 0. É possível perceber que em momentos de sobrecarga de dados, um número maior de pacotes é descartado na fila padrão.

Figura 38 - Fila DiffServ e HTB.

Name	Parent	Packet ...	Priority	Limit At (p...	Max Limit ...	Avg. Rate	Avg. Packet Rate	Queued Bytes	Queued Packets	Bytes	Packets	Dropped
qos_global	global		8	10M	11M	11.0 Mbps	994	0 B	0	1017.0 MB	767 233	0
critical_global	qos_global		3			92.8 kbps	12	0 B	0	8.2 MB	9 167	0
critical_global_1	critical_global	dscp_46	2	3M	4M	92.8 kbps	12	0 B	0	8.2 MB	9 167	0
flash_global	qos_global		5			600 bps	2	0 B	0	48.1 KB	1 103	0
flash_global_0	flash_global	dscp_31	1	3M	4M	600 bps	2	0 B	0	48.1 KB	1 103	0
flash_override_global	qos_global		4			1788.0 kbps	198	0 B	0	160.2 MB	149 258	0
flash_override_glo...	flash_override...	dscp_36	4	3M	4M	1788.0 kbps	198	0 B	0	160.2 MB	149 260	0
intercon_global	qos_global		2			960 bps	1	0 B	0	90.7 KB	1 036	0
intercon_global_7	intercon_global	dscp_48	8			960 bps	1	0 B	0	90.7 KB	1 036	0
routine_global	qos_global		8			9.1 Mbps	781	0 B	0	848.5 MB	606 677	0
routine_global_7	routine_global	dscp_0	8			9.1 Mbps	778	71.8 KB	49	848.5 MB	606 686	74 245

Fonte: Autoria própria.

Para garantir *failover* no sistema diante das falhas nos concentradores sem fio e proporcionar maior disponibilidade ao ambiente, os rádios sem fio foram configurados sob a topologia de ponto multiponto formando uma malha e por fim o MPLS foi habilitado para simplificar o roteamento no domínio. Na Figura 39 é apresentada a *Forwarding Table MPLS* do rádio *mesh* portal, que vincula um *label* às rotas possíveis de tráfego de pacotes.

Figura 39 - Forwarding Table MPLS.

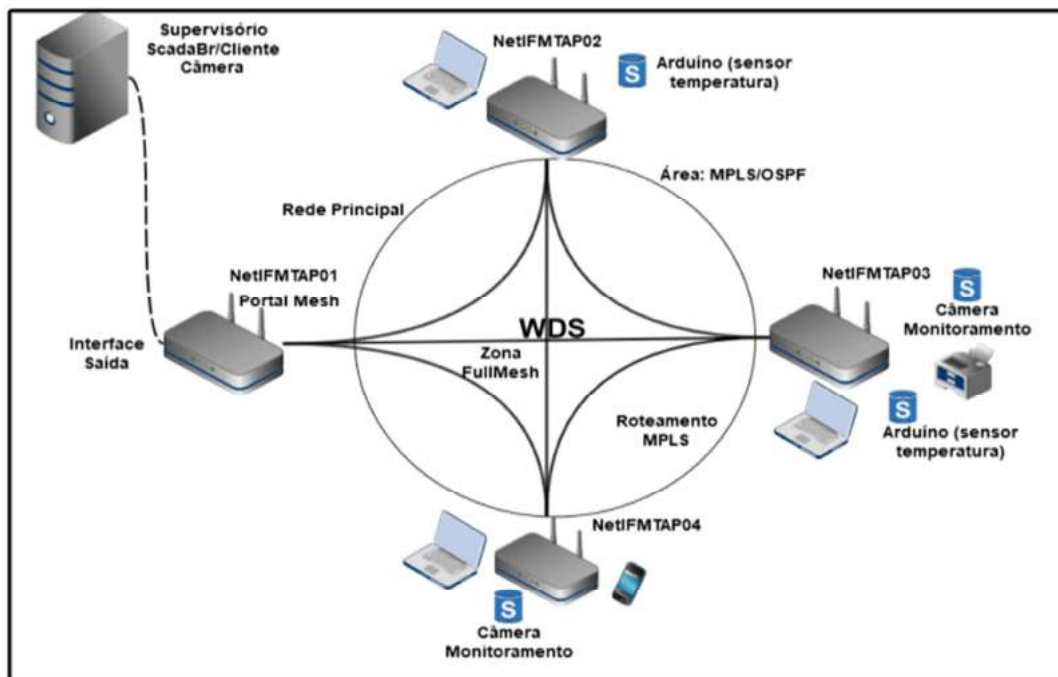
In Label	Out Labels	Interface	Nexthop	Destination
expl-null				
1620		Bridge_WDS	172.16.246.15	10.255.255.3
1621		Bridge_WDS	172.16.246.17	10.255.255.5
1622		Bridge_WDS	172.16.246.16	10.255.255.4
1623		Bridge_WDS	172.16.246.18	10.255.255.6
1624		Bridge_WDS	172.16.246.14	10.255.255.2

Fonte: Autoria própria.

Este trabalho foi desenvolvido nas dependências do IFMT – Campus Cuiabá Bela Vista, onde foi implantado o sistema de acesso á rede sem fio para acesso dos alunos e tendo como grande desafio o uso da topologia apresentada na Figura 40 para tráfego do sistema de automação composto por câmera de monitoramento e dispositivos de controle de temperatura.

Neste cenário, o protocolo HTTP é utilizado para tráfego de imagens. O protocolo nativo do setor industrial ModBus/TCP (MODBUS, 2004) foi empregado para encaminhar registros dos controladores em nível de sistema para o supervisor em nível gerencial. Para cada tipo de aplicação foi marcado o campo DSCP e definido prioridade distintas quando aplicado técnica de QoS, o contrário do tráfego no modelo *BestEffort*, assim sem QoS ambas recebem o mesmo tratamento no domínio da rede.

Figura 40 -Topologia de implantação da Política de QoS.



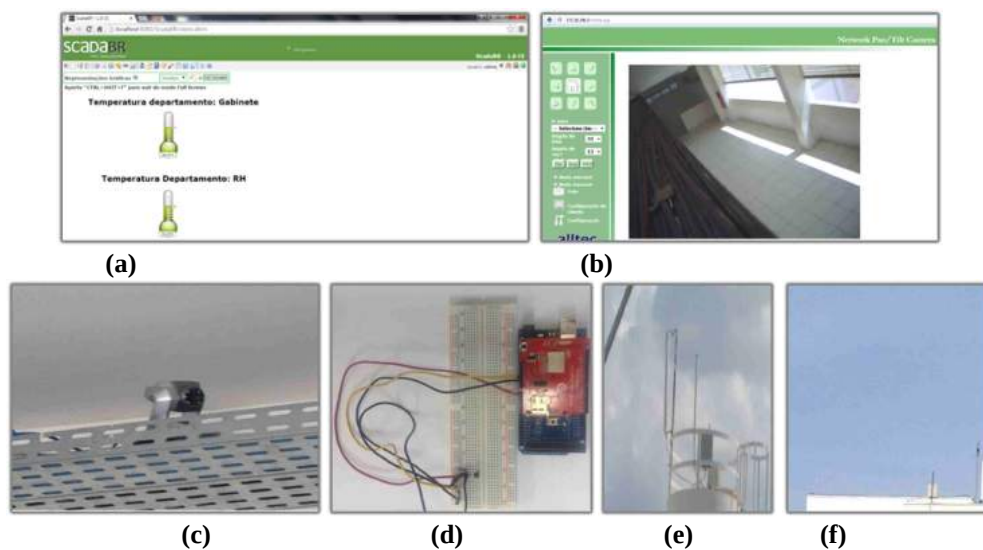
Fonte: Autoria própria.

A aplicação de automação de processos ScadaBr (ScadaBr, 2014) foi empregada; Trata-se de um sistema supervisor *open-source* desenvolvido para monitorar e controlar processos industriais, facilmente estendido para ambientes de automação predial. É usado para coletar os dados de temperatura do sensor LM-35 prototipado na porta serial da plataforma Arduíno (Mellis et al, 2007), de forma temporal cíclica no modelo cliente-servidor modbus/TCP que encaminha e registra alerta dos eventos. Esta comunicação em Modbus/TCP encaminha um frame que contém o endereço do escravo, o comando a ser executado, variável de dados e uma verificação de consistência de dados-CRC. Para garantir a disponibilidade do sistema de controle central, foi instalado o sistema SCADA, na sala de equipamentos, protegida

eletricamente por *uninterruptible power supply* (UPS) e provida de controle de acesso físico à área. Este sistema permite integrar equipamentos de vários fabricantes, incluindo sensores, atuadores, motores controlados e CLPs.

A Figura 41 apresenta a arquitetura física e lógica instalada. No nível lógico está o sistema supervisor ScadaBr (41a) e o software administrativo da câmera (41b). No nível físico os dispositivos finais, câmera e arduíno (41c e 41d) e os equipamentos de meio, rádios sem fio e as antenas direcionais/omnidirecionais (41e e 41f).

Figura41 - Estrutura física e lógica.



Fonte: Autoria própria.

Abaixo estão apresentados os resultados obtidos que demonstram o comportamento da rede de automação diante dos cenários criados e seus respectivos desempenhos, quando aplicadas políticas de QoS. Os dados foram coletados pela ferramenta de análise de desempenho Wireshark (LAMPING; SHARPE; WARNICKE, 2014), em sete amostras de tempo 600s.

Os filtros foram aplicados ao campo DSCP, agrupando em fluxos distintos as aplicações ScadaBR e o software administrativo da câmera. O número de pacotes e bytes encaminhados pelas câmeras é diferente a cada amostra, pois os frames são enviados ao cliente somente após uma detecção de movimento, ao contrário do sistema ScadaBR que foi configurado para solicitar registros a cada 10s.

Na figura 42, apresenta o filtro dos pacotes ModBus/TCP originados no arduíno e marcado pelo campo DSCP com o valor 31 (42a), é possível verificar a baixa carga de dados dos pacotes com 63 bytes, diferente da carga de *payload* de 1514 bytes da câmera, marcada com o valor do campo DSCP 36 (42b).

Figura 42: Coleta Wireshark pacotes arduino e Câmera.

No.	Time	Time Delta	Time since first frame	DSCP	LSR ID	Source	Protocol	Length
200197	2015-12-02 14:43:44.735	0.007985000	0.020621000	31		172.16.247.215	Modbus/TCP	63
200181	2015-12-02 14:43:44.727	0.006518000	0.012636000	31		172.16.247.215	TCP	60
200178	2015-12-02 14:43:44.720	0.005591000	0.005591000	31		172.16.247.215	TCP	60
198677	2015-12-02 14:43:40.271	0.000857000	0.556277000	31		172.16.247.215	TCP	60
198676	2015-12-02 14:43:40.270	0.004944000	0.555420000	31		172.16.247.215	TCP	60
198662	2015-12-02 14:43:40.227	0.007890000	0.512424000	31		172.16.247.215	Modbus/TCP	63
198658	2015-12-02 14:43:40.219	0.005916000	0.504534000	31		172.16.247.215	TCP	60
198480	2015-12-02 14:43:39.746	0.017040000	0.031045000	31		172.16.247.215	Modbus/TCP	63
198474	2015-12-02 14:43:39.729	0.006043000	0.014005000	31		172.16.247.215	TCP	60

(a)

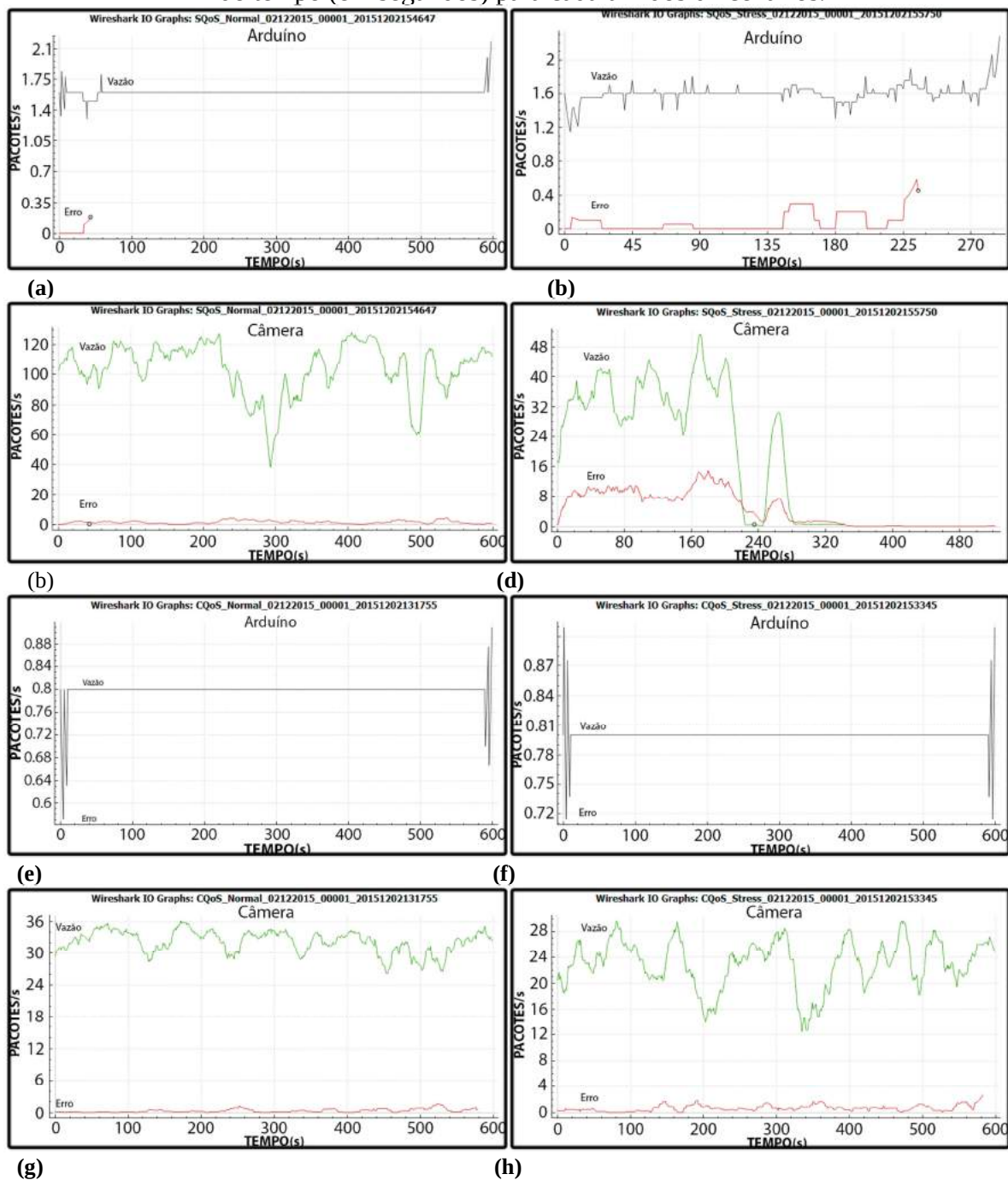
No.	Time	Time Delta	Time since first frame	DSCP	LSR ID	Source	Protocol	Length
199967	2015-12-02 14:43:44.165	0.087270000	0.523726000	36		172.16.246.3	TCP	1514
199960	2015-12-02 14:43:44.078	0.004171000	0.436450000	36		172.16.246.3	TCP	1514
199958	2015-12-02 14:43:44.073	0.004416000	0.432214000	36		172.16.246.3	TCP	1514
199957	2015-12-02 14:43:44.069	0.041831000	0.427798000	36		172.16.246.3	TCP	1514
199955	2015-12-02 14:43:44.068	0.001609000	0.521009000	36		172.16.246.3	TCP	1514
199953	2015-12-02 14:43:44.067	0.000333000	0.519400000	36		172.16.246.3	TCP	1514
199951	2015-12-02 14:43:44.066	0.041676000	0.518996000	36		172.16.246.3	TCP	1514
199948	2015-12-02 14:43:44.027	0.001236000	0.385889000	36		172.16.246.3	TCP	1514
199947	2015-12-02 14:43:44.026	0.186219000	0.384653000	36		172.16.246.3	TCP	103

(b)

Fonte: Autoria própria.

A figura 43 mostra os resultados obtidos na comparação entre a vazão (*throughput*) total e os erros na transmissão em função do tempo (em segundos) para cada um dos 04 cenários, quando aplicados filtros nos serviços de automação, arduino e câmera: sem políticas de QoS em situação normal (figura 43a e 43c), em situação crítica (estresse) (43b e 43d); com políticas de QoS em situação normal (figura 43e e 43g); em situação crítica (estresse) (43f e 43h).

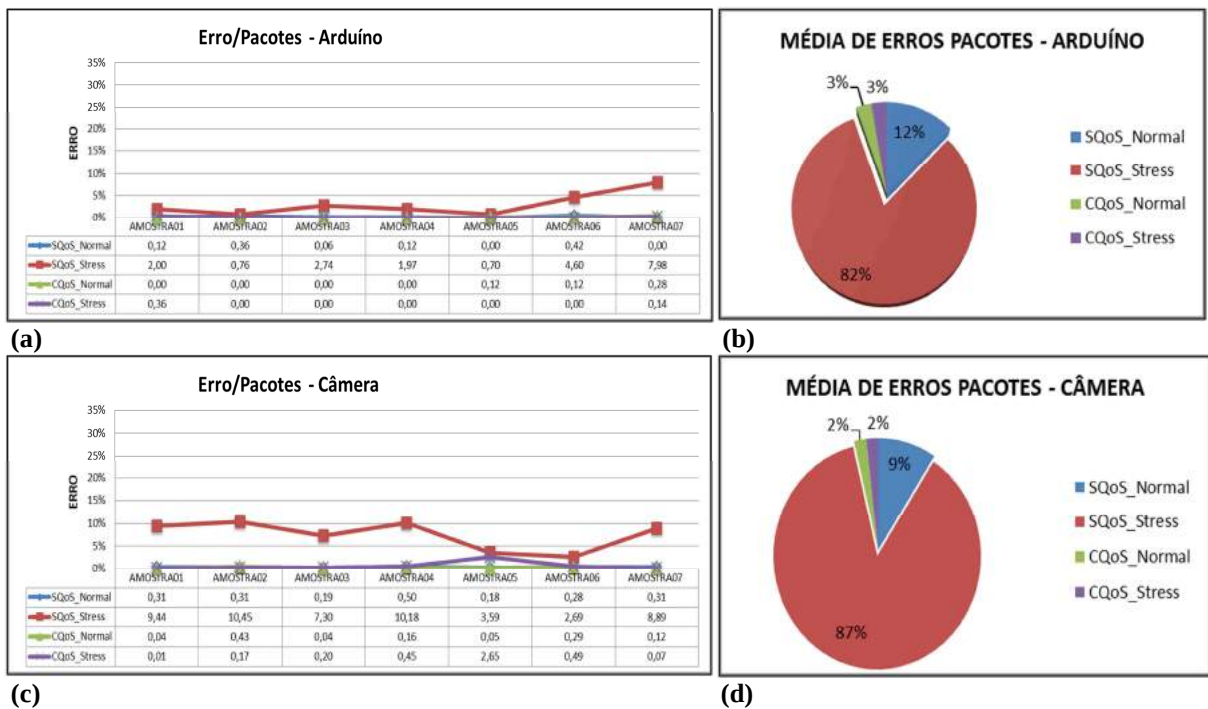
Figura 43 -Comparação entre a vazão (*throughput*) total e os erros na transmissão em função do tempo (em segundos) para cada um dos 04 cenários.



Fonte: Autoria própria.

A Figura 44 apresenta os resultados obtidos na comparação entre as médias das amostras dos erros na transmissão e a da vazão (*throughput*) total, em porcentagem, para cada um dos 04 cenários, quando aplicados filtros nos serviços de automação, arduino e câmera.

Figura 44 - Média das amostras durante sete dias Com e Sem aplicação de políticas de QoS



Fonte: Autoria própria.

6.3 ANÁLISE DA REDE SEM POLÍTICA DE QOS

Nas Figuras 43a e 43c, observa-se em situação normal, erro de transmissão praticamente nulo e vazão normal de pacotes provenientes dos serviços de automação. Isto foi possível devido ao fato da rede estar ociosa, pois o fluxo gerado pelos alunos foi inferior à capacidade de transmissão dos dados pela rede. O contrário acontece no cenário demonstrado na figura 43b e 43d, onde a rede em situação de estresse limita a vazão e aumenta o número de erros, o que torna inoperável o serviço, depois de decorrido um pequeno intervalo de tempo. Cabe também destacar que o arduíno em momentos de estresse chega a travar, sendo necessário reiniciar a placa microcontroladora para retornar a operação normal.

6.4 ANÁLISE DA REDE COM POLÍTICA DE QOS

Nas Figuras 43e e 43g, observa-se em situação normal, erro de transmissão nula no arduíno, baixo índice de erro com dados oriundos da câmera e vazão normal de pacotes provenientes destes serviços. Isto foi possível devido ao fato da rede estar ociosa. Nas Figuras 43f e 43h, nota-se que com a utilização da técnica de QoS os erros e vazão apresentam características semelhantes as observadas em situação normal, o que mantém os serviços operacionais, mesmo a rede em situação crítica. Foi possível também demonstrar o comportamento diferente dos protocolos na comunicação, no caso do pacote do arduíno de

baixo payload (63 bytes) e fluxo contínuo, a vazão segue uniforme, enquanto para a câmera com carga de dados maiores (1514 bytes) e fluxo variável conforme movimentação dos objetos frente a lente, a vazão comporta-se proporcional a estes movimentos.

6.5 ANÁLISE DA MÉDIA DE AMOSTRA COLETADAS DURANTE SETE DIAS COM E SEM APLICAÇÃO DE POLÍTICAS DE QOS

Nas Figuras 44b e 44d, demonstram-se claramente a eficiência da aplicação de técnicas de QoS, pois apresentam taxa de erros de 82% ao arduíno e 87% para câmera em situação crítica sem aplicação da mesma, o que não acontece nos períodos de emprego da solução, onde ficou com uma taxa de 3% ao arduíno e 2% para câmera. As Figuras 44a e 44c, apresentam os cenários, nota-se que a técnica é mais impactante nos casos em que o tráfego da rede está em situação de estresse, pois a taxa de variação entre os cenários em situação normal é equivalente, devido à ociosidade da rede. Nos casos de sobrecarga na rede, situação problema, observa-se que o erro é controlado, o que proporciona a eficácia da rede principal no sistema de automação.

Os resultados apresentados permitem ampliar as possibilidades de utilização de tecnologias sem fio, ao expandir seu uso como rede principal de um sistema de automação. Ao aplicar políticas de QoS, foi possível mesmo diante de situação de carga não controlada, os serviços definidos como prioritários ficarem dentro do limiar de qualidade exigida, ao reduzir o número de erros de transmissão por aplicar na comunicação entre o sistema de controle e os dispositivos gerenciados, técnicas de marcação, formatação e policiamento dos pacotes. Com o DiffServ, foi possível reduzir a complexidade de processamento de pacotes no núcleo da rede, deixando as funções complexas de processamento aos roteadores de borda do domínio.

Embora ainda não seja utilizado em grande escala, o DiffServ integrado ao HTB é uma tendência para propiciar QoS às aplicações de automação de processos, independente do meio de transporte, com fio ou sem fio. Com aplicações conhecendo a qualidade necessária e de como melhor otimizar os recursos disponíveis tomando decisões em “tempo real”, diminui a complexidade da rede e tem-se um ganho de qualidade na utilização da mesma.

Os múltiplos fluxos concorrentes melhoram a utilização da banda, porém comprometem a previsibilidade, pois, diante da presença de vários fluxos prioritários em uma classe, a ordem de entrada infere sobre seu desempenho, isto é, vários fluxos idênticos no momento de congestionamento têm desempenhos diferentes. Quando temos um número excessivo de

fluxos concorrendo dentro de uma classe é oportuno conhecer bem o mecanismo que irá tratar estes fluxos, para tornar suas respostas previsíveis.

Atentar à largura de banda para comportar os fluxos simultâneos é um ponto no projeto importante a ser observado e dimensionado. É verificado que, sobre concorrência pesada, o tráfego TCP não classificado fica enfileirado e gera erro de transmissão ou tem seu desempenho inferior ao esperado.

Esta pesquisa demonstrou claramente que é possível utilizar um ambiente sem fio compartilhado com aplicações consumidoras de banda como rede principal de um sistema de automação predial, mesmo em situações com fluxos excessivos de pacotes. Nas situações com superdimensionamento ou baixa carga de dados, o sistema comportou-se de forma satisfatória, independente do uso dos métodos de QoS. Porém em situações de fluxos superiores a vazão máxima da rede, a utilização de políticas de QoS manteve a operação normal do sistema, fatores estes que possibilitam aos proprietários de imóveis, que não possuem infraestrutura de cabeamento estruturado, possam implementar processos prediais automatizados.

7 CONSIDERAÇÕES FINAIS

Diante da necessidade de sincronismo de comunicação e baixa taxa de erro entre os dispositivos de campo e o nó central de um sistema de automação predial, é crucial que o sistema de comunicação seja eficaz, pois qualquer falha neste processo, dependendo da criticidade da função do dispositivo e da janela de indisponibilidade aceitável, pode causar a ineficiência e inviabilizar aquele monitoramento dinâmico. A utilização de redes sem fio, pela limitação natural da capacidade de vazão dos dados e por sofrer fortes interferências do meio, é inviável caso não aplique ao sistema ações que melhorem o requisito de disponibilidade. Neste sentido foi possível verificar a importância da utilização de políticas de QoS para prover eficácia no transporte de dados críticos, principalmente em ambientes de redes sem fio compartilhadas.

Aplicações de sistemas críticos exigem um atraso médio mínimo e com poucos erros de transmissão, situação que diminui os números de unidades remotas atendidas por concentrador de acesso. A ampliação deste número é possível controlando a prioridade dos pacotes que entram no domínio sem fio, principalmente quando este está sobrecarregado com fluxos consumidores de banda e/ou quando a vazão é limitada. Esta expansão provê uma maior granularidade de dispositivos espalhados na área de controle por automação, permitindo fazer mais com menos recursos técnicos e financeiros.

O experimento realizado na primeira fase produziu um conjunto de dados, e o ajuste de curva permitiu obter equações que podem ser utilizadas para estimar a quantidade de dispositivos a serem inseridos na rede sem fio. O uso da forma polinomial de quarto grau permitiu interpolar os pontos com uma excelente aproximação dos pontos sobre a curva experimental. Foi possível também estimar, através dos métodos logarítmicos e exponenciais, as equações que representam o funcionamento da rede, quando são inseridos tráfegos oriundos dos dados dos sensores e atuadores. Ficou evidente que estudos mais aprofundados e baseados em outros métodos devem ser realizados, porém, foi possível identificar o padrão de comportamento da rede para o cenário estudado.

O intuito da pesquisa foi realizar um estudo direcionado ao uso do DiffServ em conjunto com o protocolo HTB na definição de marcação e controle da prioridade de tráfego em um domínio de rede sem fio compartilhado entre aplicações consumidoras de banda e sistemas críticos de automação predial. Outros trabalhos já foram desenvolvidos sobre QoS, porém poucos focados à rede sem fio, principalmente como meio de transporte no núcleo da rede para sistemas críticos sensíveis a variação na entrega de pacotes, em especial aos sistemas de

automação predial na comunicação entre os dispositivos finais e o sistema supervisor. Uma gama de estratégias foi utilizada para alcançar a eficácia em um sistema em produção, o que tornou mais complexo se comparados a ambientes simulados, como encontrado em diversos artigos. Através da marcação do bit no campo DSCP, foi possível controlar o tráfego desde a origem até o destino do pacote, alocando largura de banda a pacotes definidos como prioritários a rede. Isto contribuiu para o balanceamento das cargas por classes, melhorando a qualidade e confiabilidade dos serviços.

A escolha do método DiffServ integrado ao algoritmo de escalonamento HTB, permitiu classificar, medir, marcar e policiar os pacotes, o que possibilitou o tratamento e transmissão de pacote no núcleo da rede de modo fácil e eficiente. A taxa de erro de transmissão em situação de estresse na rede sem aplicação de políticas de QoS, quando comparada ao demais cenários é elevada, gerando travamento ou inoperabilidade dos nós gerenciados. O uso desta política proporciona ao sistema maior confiabilidade mesmo em situações de anormalidade na rede. Esta pesquisa demonstrou claramente que é possível utilizar um ambiente sem fio compartilhado com aplicações consumidoras de banda como rede principal de um sistema de automação predial, fator que possibilita aos proprietários de imóveis, que não possuem infraestrutura de cabeamento estruturado, implementar processos prediais automatizados.

Pelos dados coletados foi possível concluir que pela implantação de QoS na estrutura de rede sem fio do IFMT – Campus Cuiabá Bela Vista existente a entrega dos fluxos classificados é aumentada e a garantia de sua operação é conseguida. Todas as políticas implantadas deram estas certezas, os cenários experimentados mostraram com clareza os diversos métodos que podem ser aproveitados para conformar os fluxos de maneira estratificada por prioridades, diminuição da carga de processamento nos roteadores inseridos no núcleo da rede pela marcação do campo DSCP. Estes resultados demonstram a eficácia da arquitetura DiffServ ao permitir a coexistência de fluxos de várias naturezas, mesmo em condições de congestionamento extremo. Esta característica torna esta solução bastante viável para os serviços de automação de processos prediais.

Para trabalhos futuros é pretendido avaliar os benefícios do MPLS como mecanismo TE no núcleo de uma Rede Definida por Software - SDN, como ferramenta para determinar a rota baseada na disponibilidade da largura de banda, assim como analisar a variação no atraso dos pacotes provenientes dos serviços prioritários no domínio MPLS quando comparados ao modelo de roteamento padrão das redes atuais. Tolerância a falhas em algumas rotas pode ser possível pela utilização do MPLS, pois permite ao sistema readaptar o caminho em caso de inoperância de um concentrador.

REFERÊNCIAS

- ABELÉM, A. J. G.; ALBUQUERQUE, C. V. N.; SAADE, D. C. M.; AGUIAR, E. S.; DUARTE, J. L.; FONSECA, J. E. M. da; MAGALHÃES, L. C. S. **Redes mesh: mobilidade, qualidade de serviço e comunicação em grupo**. Niterói: Universidade Federal Fluminense, 2007.
- ARDUÍNO. **Arduíno**. [S. l.: s. n.], 2015. Disponível em: <<http://arduino.cc/en/Main/Products>>. Acesso em: 11 jan. 2015.
- ARIFIN, Y. Implentasi quality of service dengan metode HTB (Hierarchical Token Bucket) PadaPT. Komunika Lima Duabelas. **J. Elektron. IlmuKomput.**, v. 1, n. 2, p. 1–7, 2012.
- BAZZOTTI, C.; GARCIA, E. **A importância do sistema de informação gerencial para tomada de decisões**. Belo horizonte: Tupi, 2007. Disponível em http://tupi.fisica.ufmg.br/michel/docs/Artigos_e_textos/Ciencia_da_informacao/SIG_importancia_na_tomada_de_decisao.pdf>. Acesso em: 11 jan. 2015.
- BERTOCCHI, F.; GIOVANARDI, A.; MAZZINI, G.; ZORZI, M. UDP and TCP performance investigation in energy efficient ad hoc networks. In: PIMRC 2004. IEEE INTERNATIONAL SYMPOSIUM ON., 15., 2004, [S. l.]. **Proceedings...** New York: IEEE, 2004. p. 1822-1826.
- BOARETTO, N. **Tecnologia de comunicação em sistema SCADA: enfoque em comunicação wireless com espalhamento espectral**. Ponta Grossa: Cefet-PR, 2005.
- BRADEN, R.; CLARK, D.; SHENKER, S. **RFC 1633: Integrated services in the internet architecture: an overview**. [S. l.]: Status: Informational, 1994. 28 p.
- DAVIE, B.; LAWRENCE, J.; MCCLOGHRIE, K.; ROSEN, E.; SWALLOW, G.; REKHTER, Y.; DOOLAN, P. **RFC 3035: MPLS using LDP and ATM VC Switching**. [S. l.]: Network Working Group, 2001. 20 p.
- ERDTMANN, Â. M.; NETO, W. F. C. **Uma proposta para prover qualidade de serviço em redes IP**. Florianópolis: Universidade Federal de Santa Catarina, 2003.
- FERGUSON, P.; HUSTON, G. **Quality of service: delivering QoS on the internet and in corporate networks**. New York: John Wiley & Sons, 1998.
- FLOYD, S.; JACOBSON, V. Link-sharing and resource management models for packet networks. **IEEE Transactions on ACM**, New York, v. 4, p. 365–386, 1995.
- FONSECA, J. L. A. da; STANTON, M. A. **Estudo experimental de vídeo-conferência pessoal em inter-redes IP com QoS**. Rio de Janeiro: Petrobrás/TI-RJ, 2012. 8 p.
- FORD, M.; STEVENSON, T.; LEW, H. K.; SPANIER, S. **Internetworking technologies handbook**. San Jose: Macmillan Publishing, 1997.
- GATES, M.; WARSHAVSKY, A. **IPerf version 2.0.3**. [S. l.: s. n.], 2007.

GENNAOUI, S.; YIN, J.; SWINTON, S.; HNATYSHIN, V. **Investigation and comparison of MPLS QoS solution and differentiated services QoS solutions**. Glassboro: Rowan University, 2013. v. 15. 200 p.

ITU. **Definition of terms related to quality of service**. [S. l.], 2014. Disponível em: <http://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-E.800-200809-I!!PDF-E&type=items>. Acesso em: 18 nov. 2014.

JAGER, S.; JUNGEBLOD, T.; MASCHOTTA, R.; ZIMMERMANN, A. Model-based QoS evaluation for embedded wireless sensor networks. In: SYSTEMS CONFERENCE (SYSCON), 2013 IEEE INTERNATIONAL, 2013, Orlando. **Proceedings...** New York: IEEE, 2013. p. 195-199.

JARAMILLO, X. A. **Estudio de catalogación de las aplicaciones y estructuración del ancho de banda en la red interna institucional de la escuela politécnica del ejército sede sangolquí basada en mikrotik para garantizar los servicios de red**. Sangolquí: Escuela Politécnica Del Ejército, 2013.

JOSÉ, I. S. **Estrutura MPLS**. [S. l.: s. n.], 2014a. Disponível em: <<http://wiki.sj.ifsc.edu.br/wiki/images/0/03/Mplsouters.gif>>. Acesso em: 14 jul. 2014.

JOSÉ, I. S. **Estrutura campo DS (dscp)**. [S. l.: s. n.], 2014b. Disponível em: <<http://wiki.sj.ifsc.edu.br/wiki/images/thumb/4/4e/Dscp.png/400px-Dscp.png>>. Acesso em: 13 jul. 2014.

KENDY, K.; NABAS, H. **Proposta de um modelo para análise de desempenho do escalonador WFQ alimentado com tráfego LRD**. Curitiba: Universidade Federal do Paraná, 2009.

KESHAV, S. **An engineering approach to computer networking**: ATM networks, the internet, and the telephone network. Derlhi: Addison-Wesley Professional, 1997. 688 p.

LAMPING, U.; SHARPE, R.; WARNICKE, E. **Wireshark user's guide**: for wireshark 1.99: 1.99. Mountain View: Wireshark Foundation, 2014.

LEAL, A. L. D. C. **Método analítico de aproximação polinomial para problemas de ordenadas discretas em geometria cartesiana unidimensional**. Rio de Janeiro: Instituto de Engenharia Nuclear da Comissão Nacional de Energia Nuclear, 2008.

MARTE, L. C. **Automação predial**: a inteligência distribuída nas edificações. São Paulo: Carthago & Forte, 1995.

MEDEIROS, R. M. de; MORAIS, A. H. F. de; VALENTIM, R. A. de. M; HÉKIS, H. R.; DANTAS, M. D. C. R.; BARROS, D. M. da. S; BRANDÃO, G. B. Uma análise de desempenho da rede metropolitana de telemedicina dos hospitais universitários da cidade de Natal-RN/Brasil. **Holos**, Natal, v. 4, p. 153-174, 2014.

MELLIS, D.; BANZI, M.; CUARTIELLES, D.; IGOE, T. **Arduíno**: an open electronic prototyping platform. [S. l.]: Proc. CHI., 2007.

MELO, J. C. DE. **Estudo da utilização de mecanismos de QoS em redes com enlaces de banda estreita**. São Luiz: Universidade Federal do Maranhão, 2005.

MCROBERTS, M. **Arduíno básico**. São Paulo: Editora Novatec, 2011.

MODBUS, I. D. A. Modbus messaging on TCP. **IP implementation guide v1. 0a**. North Grafton: Modbus, 2004.

MONTEBELLER, S. J. **Estudo sobre o emprego de dispositivos sem fios wireless na automação do ar condicionado e de outros sistemas prediais**. São Paulo: Escola Politécnica da Universidade de São Paulo, 2006.

MOREIRA, A. S.; PORTELA, A. M.; SILVA, R. Uso da plataforma Arduíno no desenvolvimento de soluções tecnológicas para pesquisas de dados atmosféricos na Amazônia. **Revista Perspectiva AMAZÔNICA**, Santarém, p. 119–126, 2013.

MORTOL, M.; MOREIRA, J. QoS em um cenário VoIP: utilizando o agendador HTB. **Revista TIS**, São Carlos, v. 1, p. 102–110, 2012.

MURAZZO, M.; RODRÍGUEZ, N.; VERGARA, R.; CARRIZO, F.; GONZALEZ, F.; GROSSO, E. Administración de QoS en ambientes de redes de servicios convergentes. In: WORKSHOP DE INVESTIGADORES EN CIENCIAS DE LA COMPUTACION, 15., Entre Rios. **Anais...** [S. l.: s. n.], 2013. p. 53–57.

NEVES, R. P. A. DE A. **Espaços arquitetônicos de alta tecnologia: os edifícios inteligentes**. São Carlos: Universidade de São Paulo, 2002.

NICHOLS, K.; BLACK, D. L.; BLAKE, S.; BAKER, F. **RFC 2474**: definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers. [S. l.]: The Internet Society, 1998.

NYIRENDA-JERE, T. P. R. **Impact of traffic aggregation on network capacity and quality of service**. Kansas City: University of Kansas, 1996.

OYAMA, C. S. O.; LUCENA, S. C. DE. **Funções dos roteadores de borda e de núcleo**. [S. l.: s. n.], 2014. Disponível em: <<http://www.rnp.br/newsgen/0205/imagens/qos2.gif>>. Acesso em: 12 jul. 2014.

PEREIRA, M. R.; AMORIM, C. L.; CASTRO, M. C. S. **Tutorial sobre redes de sensores**. [S. l.: s. n.], 2013. Disponível em: <<http://magnum.ime.uerj.br/cadernos/cadinf/vol14/3-clicia.pdf>>. Acesso em: 12 out. 2013.

PEREZ, F.; HELENA, E. S. **Estudo de viabilidade de uso de redes sem fio no monitoramento de parques eólicos 1**. Canoas: Ulbra, 2011.

PERKINS, C.; BELDING-ROYER, E.; DAS, S. **RFC 3561**: ad hoc on-demand distance vector (AODV) routing. [S. l.: s. n.], 2003.

PINTO, J. R. **Utilização racional de energia elétrica em instalações comerciais empregando sistemas de automação**. São Paulo: Escola Politécnica da Universidade de São Paulo, 2000.

POSTELS, J. **RFC 791 internet protocol DARPA internet program protocol specification**. [S. l.]: Defense Advanced Research Projects Agency, 1981.

POZZUOLI, M. P. “Zero-Packet-Loss” in the substation. **Electric Energy T and D**, Ontoro, v. 7, n. 1, p. 72-74, 2003.

RAMOS, D. F. DA S. **Algoritmo de Escalonamento para Diferenciação de Qualidade de Serviço em Redes DiffServ/MPLS**. Porto: Faculdade de Engenharia da Universidade do Porto, 2013.

REIS, L. G. dos. **Proposta de um mecanismo para alocação otimizada de recursos em domínios DIFFSERV**. Uberlândia: Universidade Federal de Uberlândia, 2006.

ROSEN, E.; VISWANATHAN, A.; CALLON, R. RFC 3031: multiprotocol label switching architecture. [S. l.]: Network Working Group, 2001.

RUELA, J. C. **Algumas Análises Sobre Mecanismos para Prover Qualidade de Serviço em Redes Multimídia**. Santa Rita do Sapucaí: Instituto Nacional de Telecomunicações, 2006.

SANTOS, R. C. dos. **Um estudo do uso da tecnologia MPLS em backbones no Brasil**. Florianópolis: Universidade Federal de Santa Catarina, 2003.

SANTOS, R. R. **Implementação de rede mesh com servidor centralizado de autenticação**. Curitiba: Universidade Federal do Paraná, 2012.

SANTOS, S. T. dos. **Redes de sensores sem fio em monitoramento e controle**. 2007. 87 f. Tese (doutorado) - Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2007.

SCADABR, **Manual**. [S. l.: s. n.], 2014. Disponível em: <[http://ufpr.dl.sourceforge.net/project/scadabr/Software/Docs/Manual% 20ScadaBR. pdf](http://ufpr.dl.sourceforge.net/project/scadabr/Software/Docs/Manual%20ScadaBR.pdf)>. Acesso em: 12 dez. 2014.

SEMERIA, C. **Supporting differentiated service classes: queue scheduling juniper networks: 200020-001**. Sunnyvale: Juniper Networks, 2000.

SIA, M. **MikroTik™ V2.0 router software technical reference manual: 2000-05-05**. Berkeley: Mikrotikls SIA, 1999.

SIEWERT, V. C. **Firewall suas características e vulnerabilidades**. [S. l.: s. n.], 2012. Disponível em: <http://www.researchgate.net/publication/228788099_Firewall_Suas_Caractersticas_e_Vulne_rabilidades>. Acessado em: 18 dez. 2014.

- SILVA, A. P. G. da; SALVADOR, M. **O que são sistemas supervisórios?**. [S. l.: s. n.], 2005. Disponível em: http://scholar.googleusercontent.com/scholar?q=cache:8DgYDPvsC9kJ:scholar.google.com/+O+que+s%C3%A3o+sistemas+supervis%C3%B3rios%3F&hl=pt-BR&as_sdt=0,5, 2013. Acesso em: 08 jan. 2016.
- SILVA, B. R. S. da. **Sistema de automação residencial de baixo custo para redes sem fio**. Porto Alegre: Universidade Federal do Rio Grande do Sul, 2014a.
- SILVA, M. A. da. **Um SLA para VoIP e seu mapeamento em uma rede DiffServ/MPLS**. Florianópolis: Universidade Federal de Santa Catarina, 2005.
- SILVA, R. R. da. **Diagrama de blocos da CPU do CLP**. [S. l.: s. n.], 2014b. Disponível em: http://profrfaelrs.files.wordpress.com/2012/11/arquitetura_clp02.png. Acesso em: 7 jul. 2014.
- SILVESTRE BLANES, J.; BERENGUER-SEBASTIÁ, J.; SEMPERE-PAYA, V.; TODOLÍ FERRANDIS, D. 802.11 n Performance analysis for a real multimedia industrial application. **Computers in Industry**, Alcoy, v. 66, p. 31-40, 2015.
- SOUSA, M. A. B. DE. **Estudo comparativo entre a aplicação de sistemas dedicados e a utilização de controladores lógico-programáveis na automação de sistemas prediais**. São Paulo: Escola Politécnica da Universidade de São Paulo, 2004.
- TANEMBAUM, A. **Rede de computadores**. 5. ed. São Paulo: Pearson Education, 2011.
- TAKAHASHI, B. K.; VIZCAINO, F. M. **Sistema de informação residencial para consumo energético**. São Paulo: Escola Politécnica da Universidade de São Paulo, 2013.
- THOMPSON, J.; MCKEAY, M.; BRENNER, B.; MOLLER, R.; SINTORN, M.; HUSTON, G. **Akamai's State Of The Internet, Q1 2014 Rep.**, v. 7, n. 1, p. 1-44, 2014.
- TORRES, G. **Redes de computadores: curso completo**. Rio de Janeiro: Novaterra, 2009.
- VIEGAS JR, R.; AFFONSO, L. A.; VASQUES, F.; PORTUGAL, P.; MORAES, R. Real-time industrial communication over IEEE802.11e wireless local area networks. **IEEE Latin America Transactions**, Amsterdam, v. 10, n. 3, p. 1844–1849, 2012.
- VILELA, D. W. F. L.; SHINODA, A. A.; TAVARES, W.; FERREIRA, R. D. O.; DO NASCIMENTO, V. E.; ARAÚJO, N. Construção de bases de dados para auxiliar a avaliação de sistemas de detecção de intrusos em uma rede IEEE 802.11 com Criptografia WEP, WPA e WPA2. In: ENCONTRO ANUAL DE COMPUTAÇÃO - ENACOMP 2013, 10., 2013, Catalão. **Proceedings...** [S. l.: s. n.], 2013. p. 145–151.
- VIVA SEM FIO. **Sensores e atuadores em redes ad-hoc**. [S. l.: s. n.], 2014. Disponível em: http://www.vivasemfio.com/blog_images/WSN_vulcao.jpg. Acesso em: 7 jul. 2014.
- YAGHMAEE, M. H.; MOBASHERI, B. **A priority-based IP/MPLS management scheme**. Case-study: Smart Grids, 2012.

ZANETTI, M.; MICHELON, G. A. **Avaliação de serviços diferenciados no Linux**. Uberlândia: UFMG, 2007.

ZHANG, L.; BERSON, S.; HERZOG, S.; & JAMIN, S. **RFC 2205**: Resource reservation protocol (RSVP) - version 1 functional Specification. [S. l.]: Network Working Group, 1999.

ZOLHAVARIEH, S.; BARATI, M. **Quality of service in wireless sensor networks (QoS in WSN)**. Belo Horizonte: UFMG, 2012.

APÊNDICE A – ARTIGOS PUBLICADOS EM ANAIS DE EVENTOS

ALVES, G., FERREIRA, E. T., & SHINODA, A. A. Uma proposta de avaliação de dados de redes de sensores utilizando redes em malha sem fio. **Sistema de Publicações. IFMT-Campus Cuiabá**, v. 1, n. 1, 2013.

ALVES, G., SHINODA, A., FERREIRA, E. T., OLIVEIRA, R., & NASCIMENTO, V. E. **Uma proposta de emprego de técnicas de qualidade de serviço em redes sem fio aplicada à automação predial.** 6º Workshop de Negócios e Inovação do IFSP, Caraguatatuba, 2015.

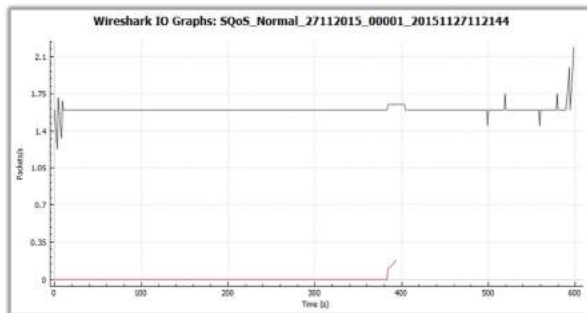
APÊNDICE B – ARTIGOS SUBMETIDOS

ALVES, G., SHINODA, A., FERREIRA, E. T., OLIVEIRA, R., & NASCIMENTO, V. E. O emprego de técnicas de qualidade de serviços em redes sem fio aplicadas a automação predial. **Revista RETEC**, ISSN: 2236-739X, 2016.

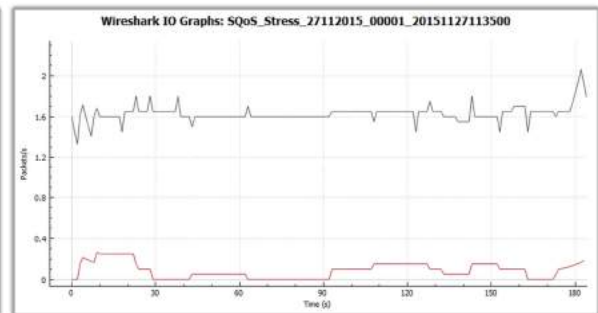
Aceito e aguardando publicação

ALVES, G., SHINODA, A., FERREIRA, E. T., OLIVEIRA, R., & NASCIMENTO, V. E. Uma proposta de emprego de técnicas de qualidade de serviço em redes sem fio aplicada à automação predial. **Revista Sinergia**, ISSN: 2177-451X, 2016.

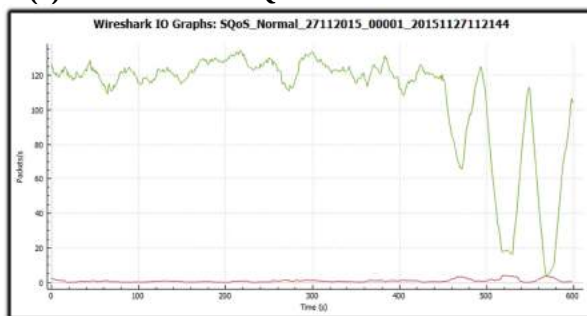
APÊNDICE C – COMPARAÇÃO ENTRE A VAZÃO (*THROUGHPUT*) TOTAL E OS ERROS NA TRANSMISSÃO EM FUNÇÃO DO TEMPO (EM SEGUNDOS) PARA CADA UM DOS 04 CENÁRIOS – SETE AMOSTRAS



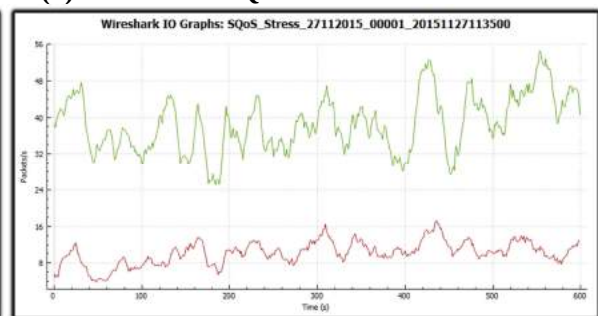
(a) Arduíno Sem QoS Normal



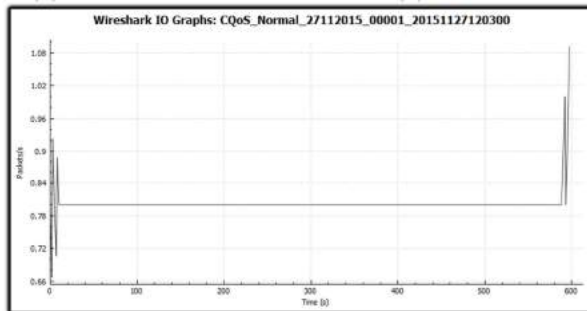
(b) Arduíno Sem QoS Stress



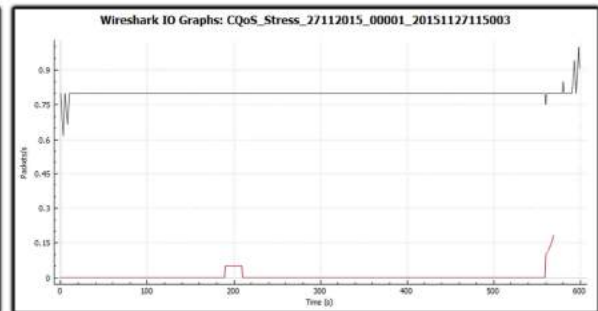
(c) Câmera Sem QoS Normal



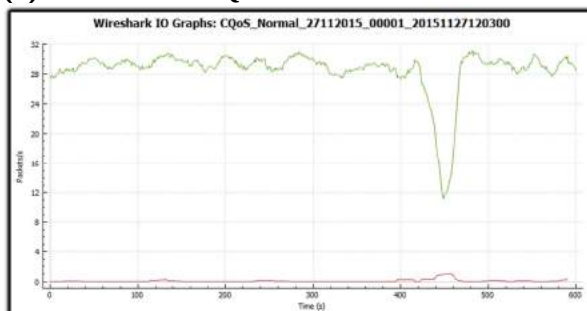
(d) Câmera Sem QoS Stress



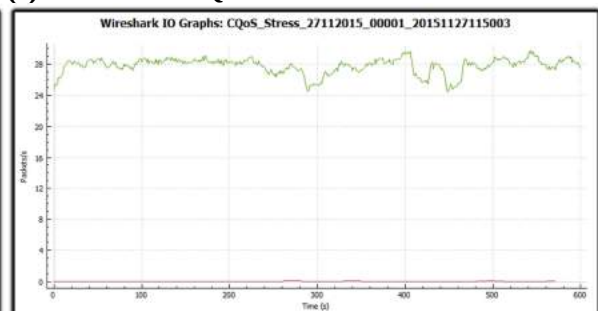
(e) Arduíno Com QoS Normal



(f) Arduíno Com QoS Stress

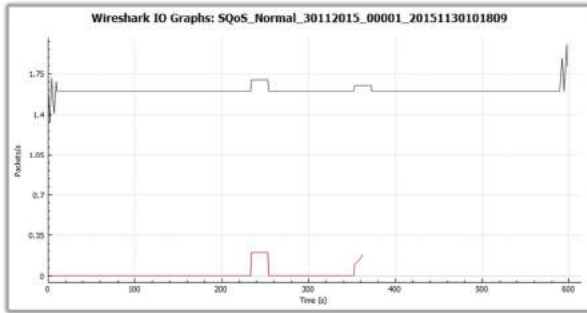
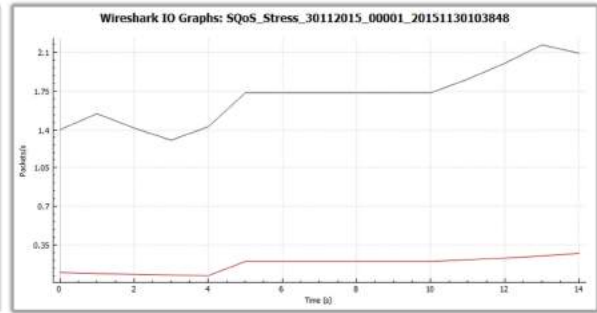
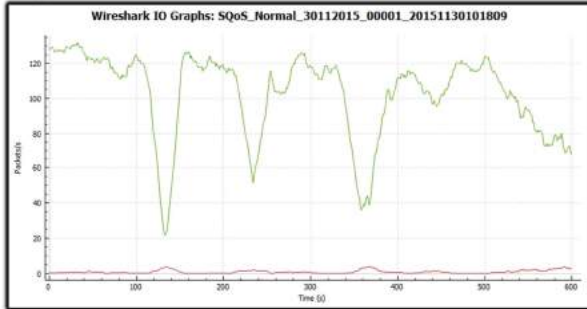
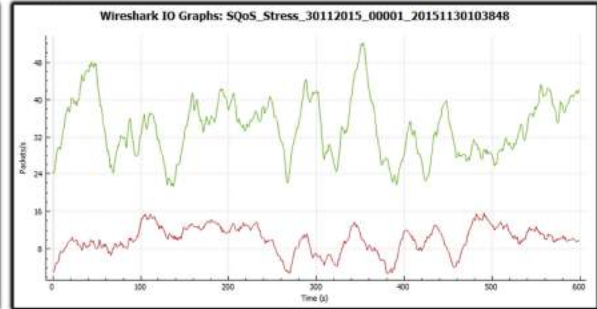
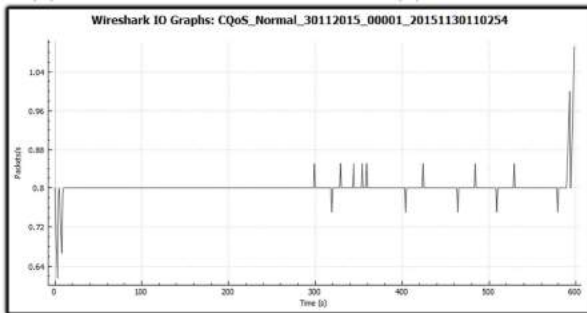
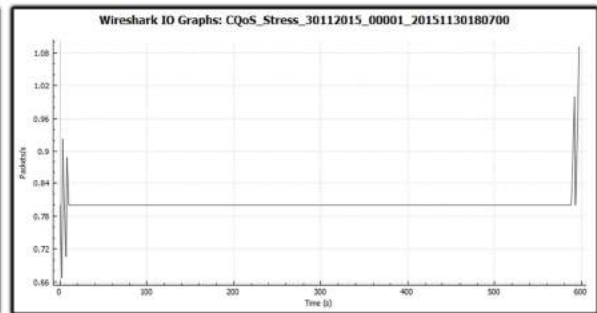
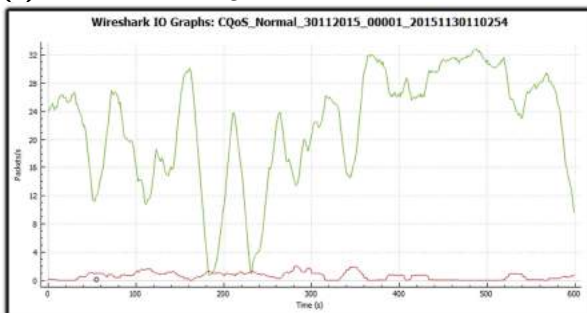
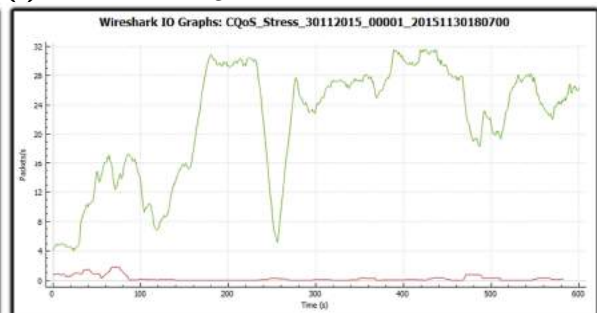


(g) Câmera Com QoS Normal

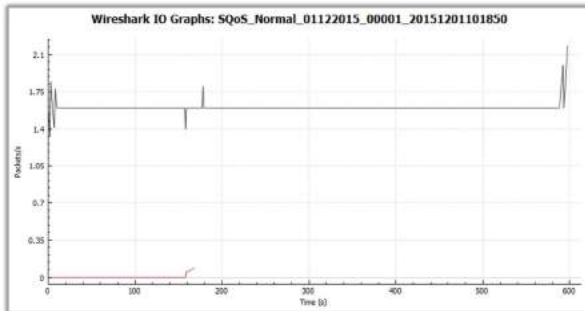
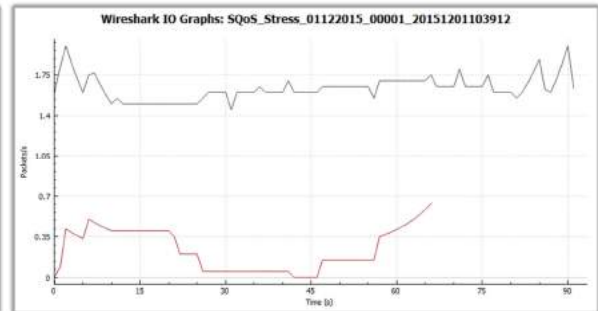
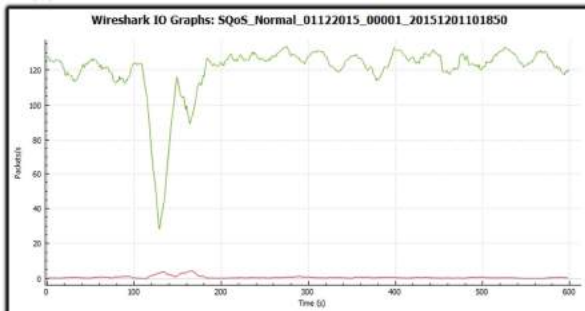
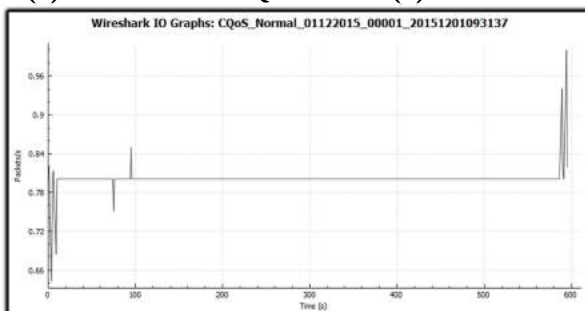
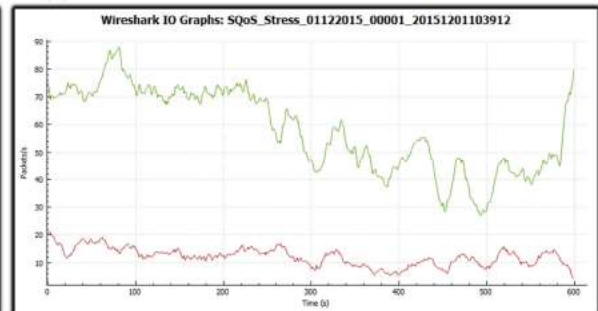
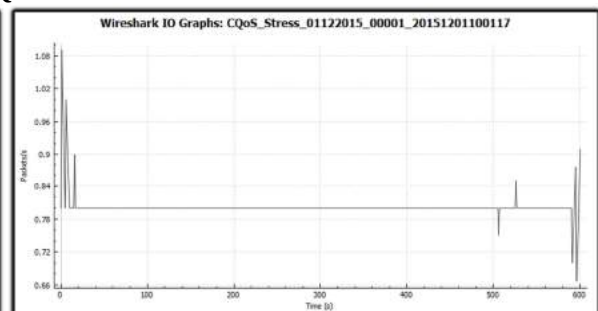
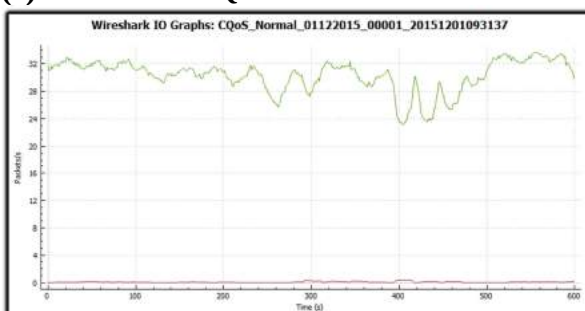
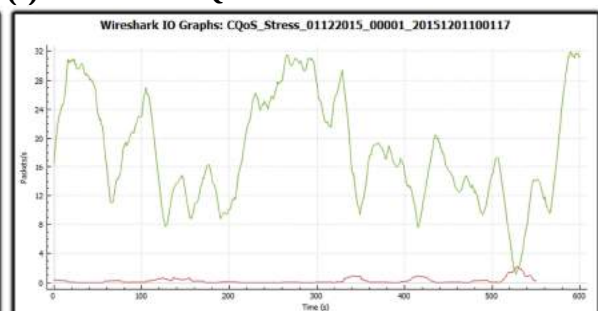


(h) Câmera Com QoS Stress

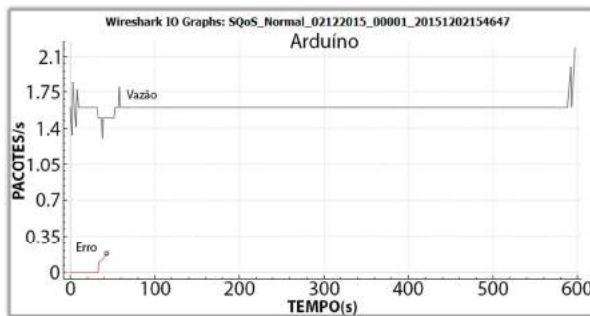
PRIMEIRA AMOSTRA

(c) **Arduíno Sem QoS Normal**(b) **Arduíno Sem QoS Stress**(d) **Câmera Sem QoS Normal**(d) **Câmera Sem QoS Stress**(e) **Arduíno Com QoS Normal**(f) **Arduíno Com QoS Stress**(g) **Câmera Com QoS Normal**(h) **Câmera Com QoS Stress**

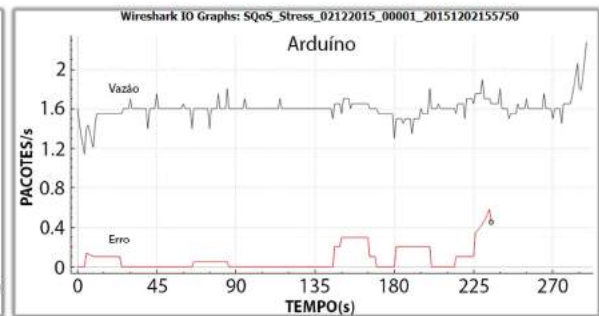
SEGUNDA AMOSTRA

(a) **Arduíno Sem QoS Normal**(b) **Arduíno Sem QoS Stress**(b) **Câmera Sem QoS Normal(d) Câmera Sem QoS Stress**(e) **Arduíno Com QoS Normal**(f) **Arduíno Com QoS Stress**(g) **Câmera Com QoS Normal**(h) **Câmera Com QoS Stress**

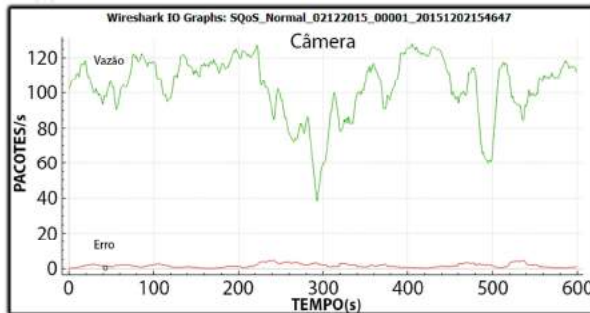
TERCEIRA AMOSTRA



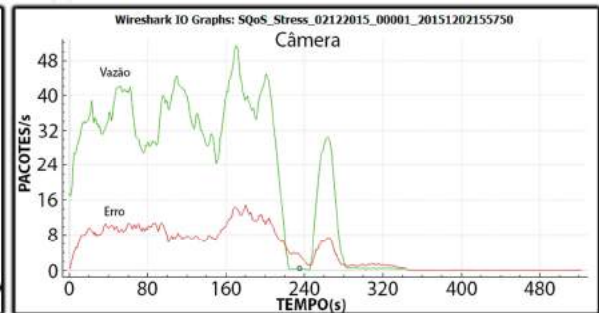
(a) Arduino Sem QoS Normal



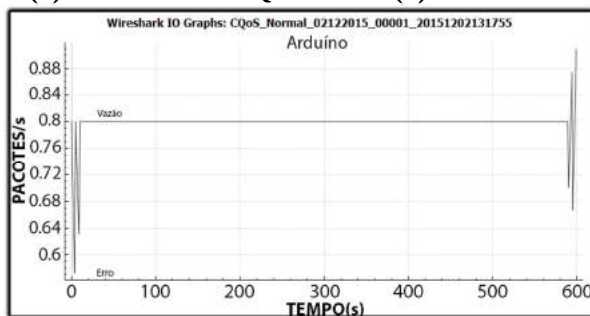
(b) Arduino Sem QoS Stress



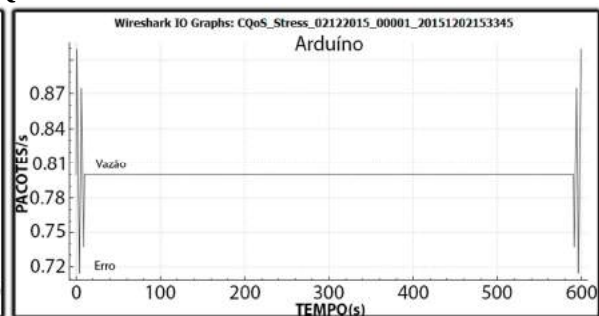
(b) Câmera Sem QoS Normal



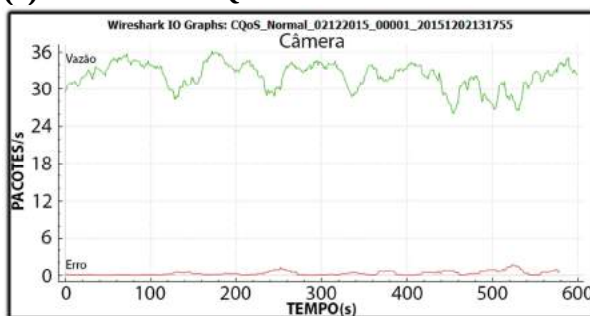
(d) Câmera Sem QoS Stress



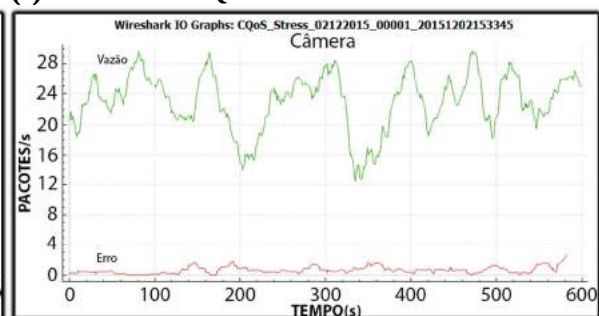
(e) Arduino Com QoS Normal



(f) Arduino Com QoS Stress

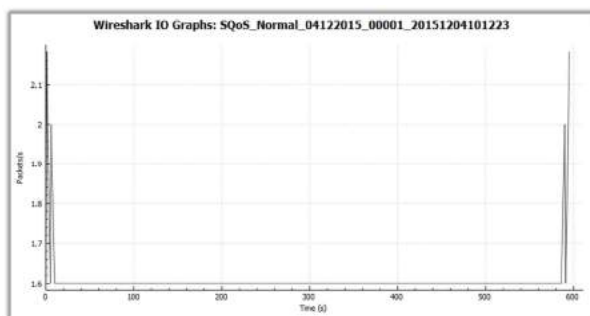
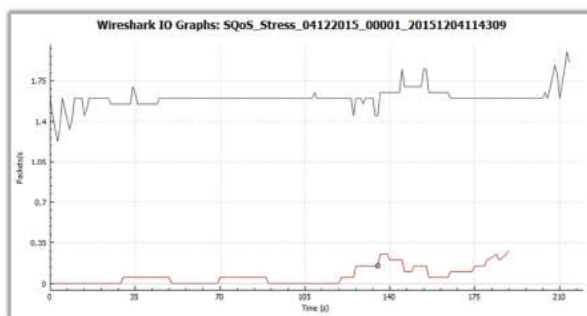
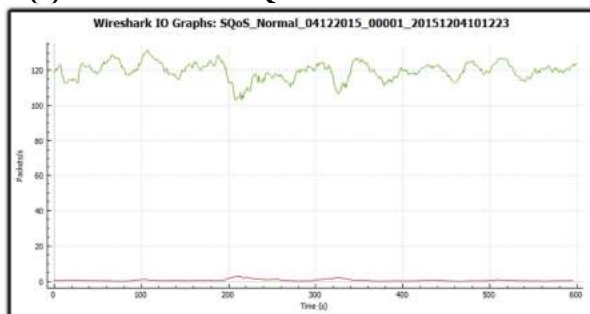
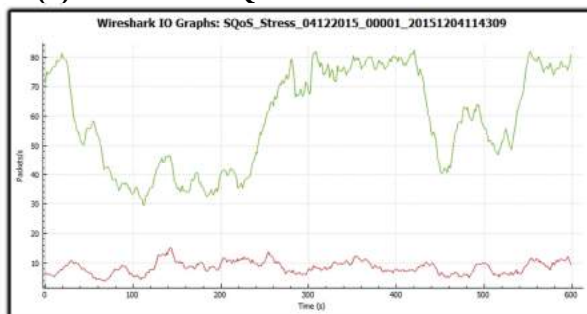
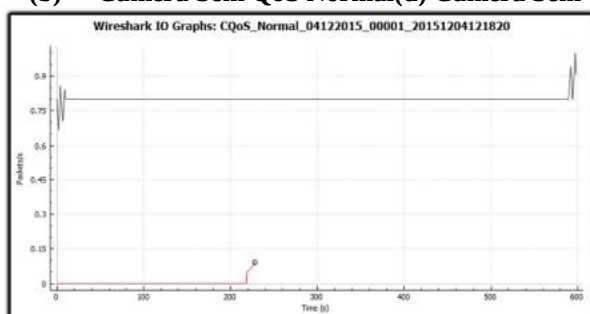
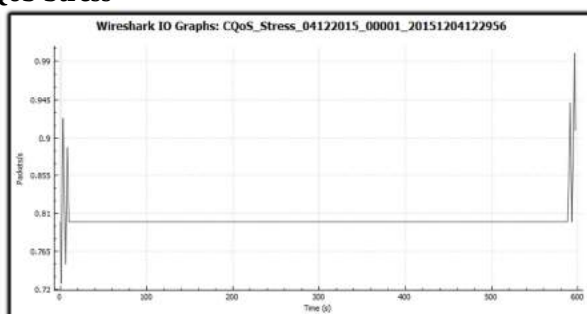
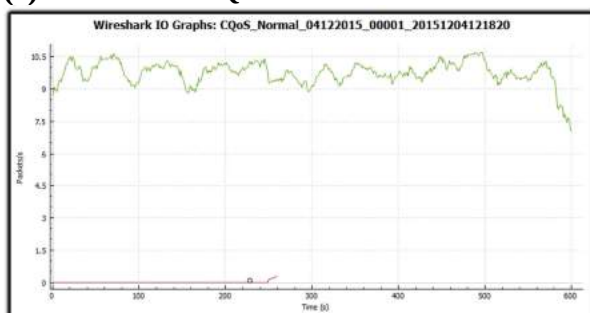
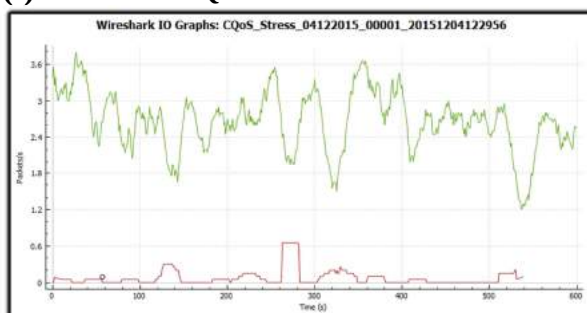


(g) Câmera Com QoS Normal

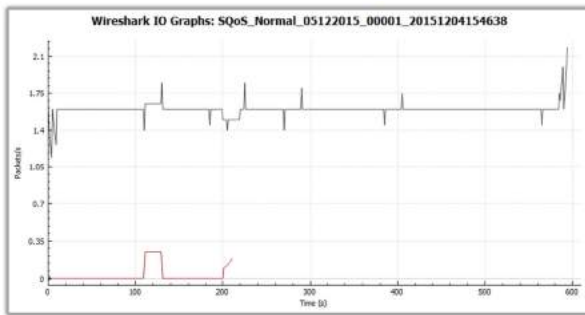
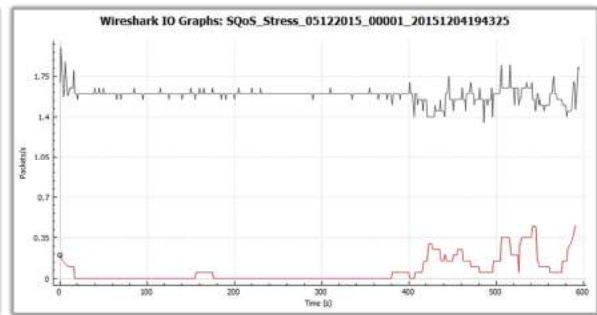
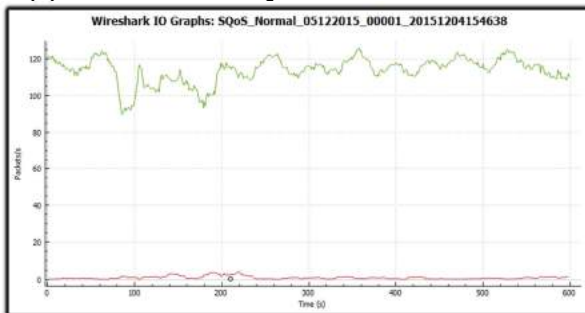
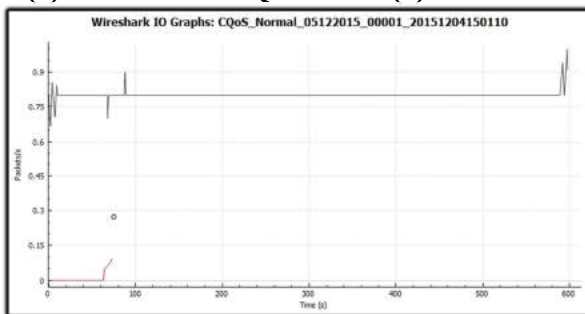
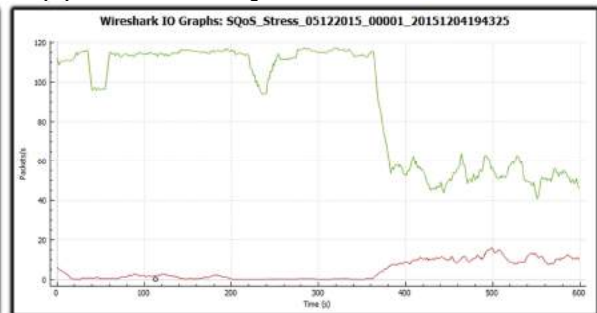
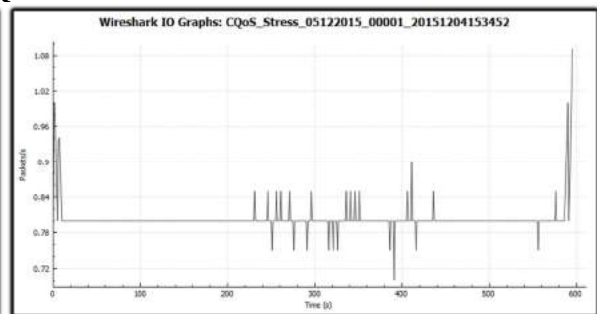
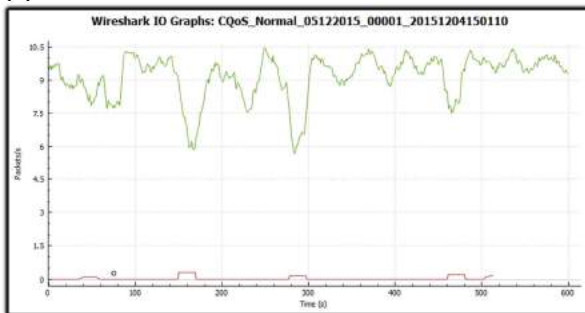
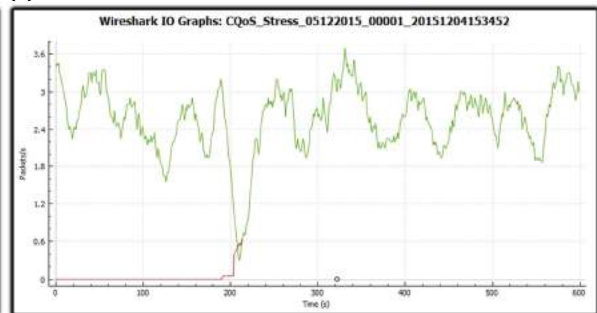


(h) Câmera Com QoS Stress

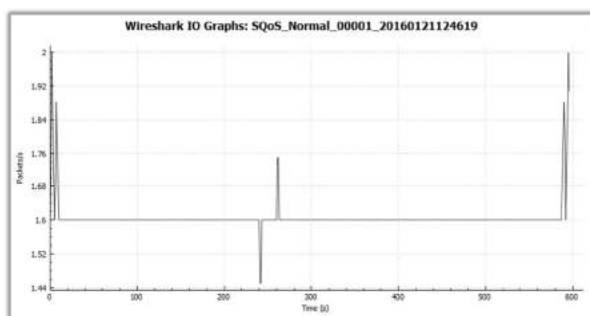
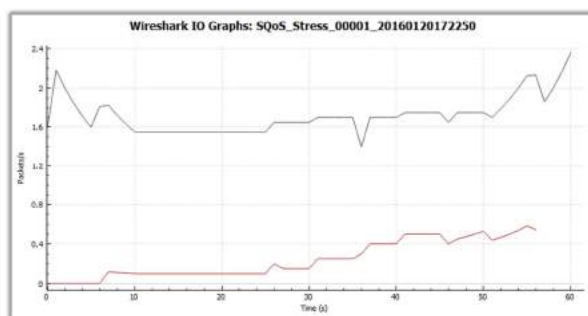
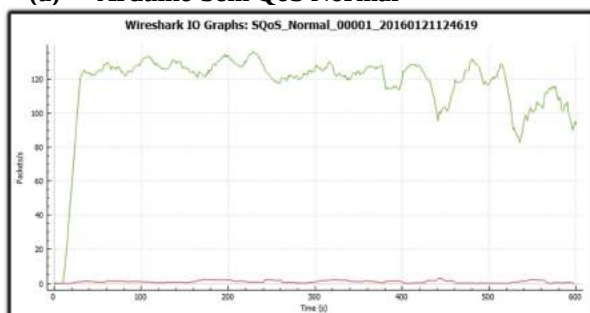
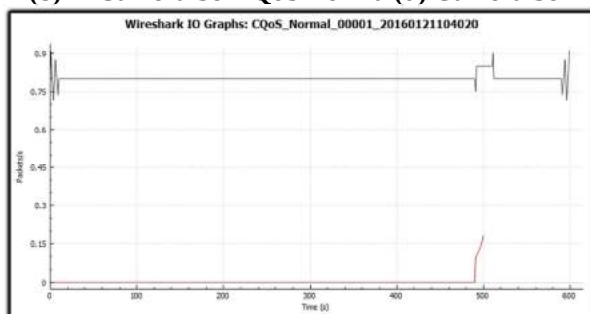
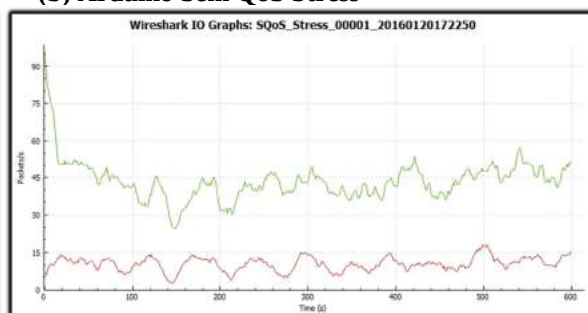
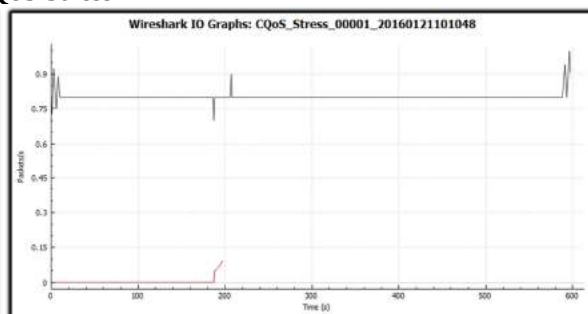
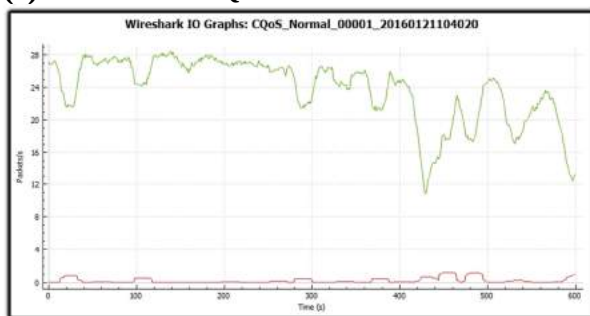
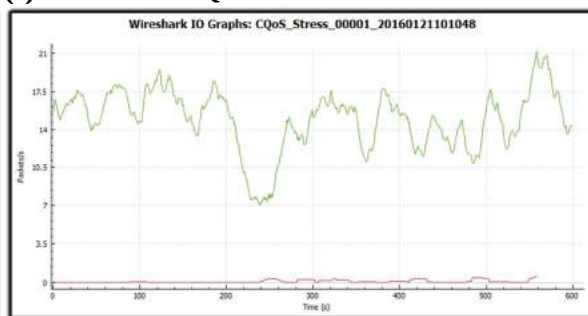
QUARTA AMOSTRA

(a) **Arduíno Sem QoS Normal**(b) **Arduíno Sem QoS Stress**(b) **Câmera Sem QoS Normal**(d) **Câmera Sem QoS Stress**(e) **Arduíno Com QoS Normal**(f) **Arduíno Com QoS Stress**(g) **Câmera Com QoS Normal**(h) **Câmera Com QoS Stress**

QUINTA AMOSTRA

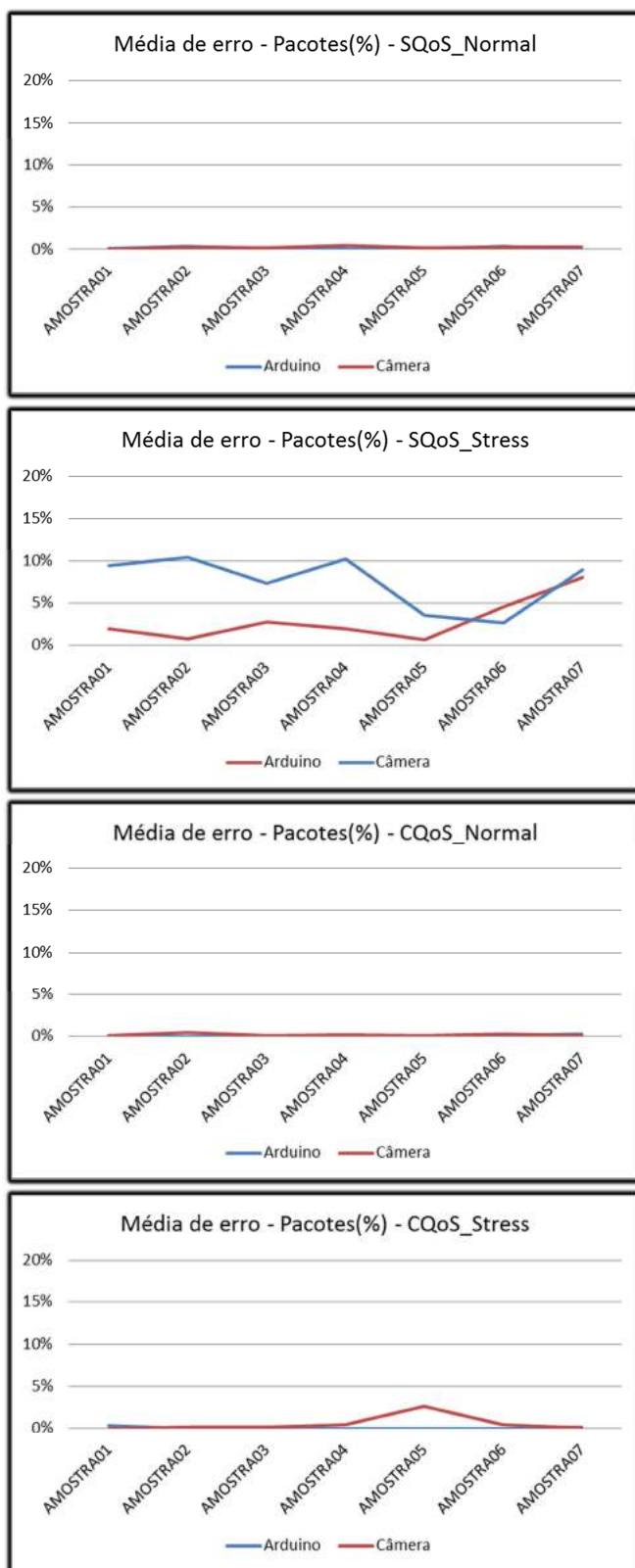
(a) **Arduíno Sem QoS Normal**(b) **Arduíno Sem QoS Stress**(b) **Câmera Sem QoS Normal(d) Câmera Sem QoS Stress**(e) **Arduíno Com QoS Normal**(f) **Arduíno Com QoS Stress**(g) **Câmera Com QoS Normal**(h) **Câmera Com QoS Stress**

SEXTA AMOSTRA

(a) **Arduíno Sem QoS Normal**(b) **Arduíno Sem QoS Stress**(b) **Câmera Sem QoS Normal(d) Câmera Sem QoS Stress**(e) **Arduíno Com QoS Normal**(f) **Arduíno Com QoS Stress**(g) **Câmera Com QoS Normal**(h) **Câmera Com QoS Stress**

SÉTIMA AMOSTRA

APÊNDICE D – MÉDIA DAS AMOSTRAS DURANTE SETE DIAS COM E SEM APLICAÇÃO DE POLÍTICAS DE QOS POR SERVIÇOS DE AUTOMAÇÃO PREDIAL



APÊNDICE E – CÓDIGO CARREGADO NO ARDUÍNO – PROJETO SENSOR DE TEMPERATURA USANDO MODBUS/TCP

```
/*
```

```
Simple Test
```

```
*/
```

```
// Note: The Arduino IDE does not respect conditional included
// header files in the main sketch so you have to select your
// hardware here, and in ModbusTCP.h.
// If you want to use both wired and wireless in different
// projects copy ModbusTCP.h and ModbusTCP.cpp to the individual
// sketch folders instead of installing the library.
```

```
#include <Ethernet.h>
#include <Adafruit_CC3000.h>
#include <SPI.h>
#include "ModbusTCP.h"
```

```
// Register map for this example
#define S1_FloatConstant 40001 // modpoll -m tcp -t 4:float -r 40001 [ip_addr]
#define S1_CurrentMillis 40003 // modpoll -m tcp -t 4:int -r 40003 [ip_addr]
#define S1_FeetPerMile 40005 // modpoll -m tcp -t 4 -r 40005 [ip_addr]
```

```
ModbusTCP m;
```

```
void setup(void)
```

```
{
  Serial.begin(115200);

  // set some initial values
  m.setFloat(S1_FloatConstant, PI);
  m.setU32(S1_CurrentMillis, millis());
  m.setU16(S1_FeetPerMile, 5280);

  //uint8_t mac[] = { 0x70, 0xFF, 0x76, 0x00, 0xDF, 0x25 };
  //m.begin(mac);
  m.begin("ccbv-cti", "BelaVista", WLAN_SEC_WPA2);
}
```

```
void loop(void)
```

```
{
  // Process MODBUS requests on every scan
  m.run();

  // Update the MODBUS registers
  m.setU32(S1_CurrentMillis, millis());
}
```

APÊNDICE F – CONFIGURAÇÃO DO RÁDIO MIKROTIK NETIFMTAP01

```

/interface bridge
add name=Bridge_WDS
add name=loopback protocol-mode=none
/interface ethernet
set [ find default-name=ether1 ] advertise=10M-half,10M-full,100M-half,100M-
full
set [ find default-name=ether2 ] advertise=10M-half,100M-half
/interface wireless security-profiles
set [ find default=yes ] supplicant-identity=MikroTik
add authentication-types=wpa-psk,wpa2-psk eap-methods="" group-ciphers=\
tkip,aes-ccm mode=dynamic-keys name=NETFMTBVPSPK supplicant-
identity="" \
unicast-ciphers=tkip,aes-ccm wpa-pre-shared-key=XXXXXXXX \
wpa2-pre-shared-key=xxxxxxx
add authentication-types=wpa-psk,wpa2-psk eap-methods="" group-ciphers=\
tkip,aes-ccm mode=dynamic-keys name=CCBV-AUTOMACAO supplicant-
identity="" \
unicast-ciphers=tkip,aes-ccm wpa-pre-shared-key=arduinos \
wpa2-pre-shared-key=arduinos
add authentication-types=wpa-psk,wpa2-psk eap-methods="" group-ciphers=\
tkip,aes-ccm mode=dynamic-keys name=PSK_WDS supplicant-identity="" \
unicast-ciphers=tkip,aes-ccm wpa-pre-shared-key=xxxxxxx \
wpa2-pre-shared-key=XXXXXX
/interface wireless
set [ find default-name=wlan1 ] band=2ghz-b/g disabled=no frequency=2412 \
hide-ssid=yes l2mtu=1600 mode=ap-bridge radio-name=NetIFMTBVAP01 \
security-profile=PSK_WDS ssid=NetIFMTBVAcad wds-default-
bridge=Bridge_WDS \
wds-ignore-ssid=yes wds-mode=dynamic
/interface wireless wds
add disabled=no l2mtu=1600 master-interface=wlan1
name=WDS_NetIFMTBVAP02 wds-address=00:02:6F:75:CF:50
add disabled=no l2mtu=1600 master-interface=wlan1
name=WDS_NetIFMTBVAP03 wds-address=00:02:6F:75:CF:40
add disabled=no l2mtu=1600 master-interface=wlan1
name=WDS_NetIFMTBVAP04 wds-address=00:02:6F:75:CF:30
add disabled=no l2mtu=1600 master-interface=wlan1
name=WDS_NetIFMTBVAP05 wds-address=00:02:6F:75:CF:20
/interface wireless
add disabled=no l2mtu=1600 mac-address=02:02:6F:75:CF:20 master-
interface=\
wlan1 max-station-count=30 name=VAP01 security-
profile=NETFMTBVPSPK ssid=\
CCBV-AP01-ACADEMICA wds-cost-range=0 wds-default-cost=0
/queue tree
add limit-at=10M max-limit=11M name=qos_global parent=global
add name=netcon_global parent=qos_global priority=1 queue=ethernet-default
add name=netcon_global_0 packet-mark=dscp_63 parent=netcon_global
priority=1 queue=ethernet-default
add name=netcon_global_1 packet-mark=dscp_62 parent=netcon_global
priority=2 queue=ethernet-default
add name=netcon_global_2 packet-mark=dscp_61 parent=netcon_global
priority=3 queue=ethernet-default
add name=netcon_global_3 packet-mark=dscp_60 parent=netcon_global
priority=4 queue=ethernet-default
add name=netcon_global_4 packet-mark=dscp_59 parent=netcon_global
priority=5 queue=ethernet-default
add name=netcon_global_5 packet-mark=dscp_58 parent=netcon_global
priority=6 queue=ethernet-default
add name=netcon_global_6 packet-mark=dscp_57 parent=netcon_global
priority=7 queue=ethernet-default
add name=netcon_global_7 packet-mark=dscp_56 parent=netcon_global
queue=ethernet-default
add name=intercon_global parent=qos_global priority=2 queue=ethernet-default
add name=intercon_global_0 packet-mark=dscp_55 parent=intercon_global
priority=1 queue=ethernet-default
add name=intercon_global_1 packet-mark=dscp_54 parent=intercon_global
priority=2 queue=ethernet-default
add name=intercon_global_2 packet-mark=dscp_53 parent=intercon_global
priority=3 queue=ethernet-default
add name=intercon_global_3 packet-mark=dscp_52 parent=intercon_global
priority=4 queue=ethernet-default
add name=intercon_global_4 packet-mark=dscp_51 parent=intercon_global
priority=5 queue=ethernet-default
add name=intercon_global_5 packet-mark=dscp_50 parent=intercon_global
priority=6 queue=ethernet-default
add name=intercon_global_6 packet-mark=dscp_49 parent=intercon_global
priority=7 queue=ethernet-default
add name=intercon_global_7 packet-mark=dscp_48 parent=intercon_global
queue= ethernet-default
add name=critical_global parent=qos_global priority=3 queue=ethernet-default
add limit-at=1M max-limit=2M name=critical_global_0 packet-mark=dscp_47
parent=critical_global priority=1 queue=ethernet-default
add limit-at=3M max-limit=4M name=critical_global_1 packet-mark=dscp_46
parent=critical_global priority=2 queue=ethernet-default
add name=critical_global_2 packet-mark=dscp_45 parent=critical_global
priority=3 queue=ethernet-default
add name=critical_global_3 packet-mark=dscp_44 parent=critical_global
priority=4 queue=ethernet-default
add name=critical_global_4 packet-mark=dscp_43 parent=critical_global
priority=5 queue=ethernet-default
add name=critical_global_5 packet-mark=dscp_42 parent=critical_global
priority=6 queue=ethernet-default
add name=critical_global_6 packet-mark=dscp_41 parent=critical_global
priority=7 queue=ethernet-default
add name=critical_global_7 packet-mark=dscp_40 parent=critical_global
queue= ethernet-default
add name=flash_override_global parent=qos_global priority=4 queue=ethernet-
default
add name=flash_override_global_0 packet-mark=dscp_39 parent=
flash_override_global priority=1 queue=ethernet-default
add name=flash_override_global_1 packet-mark=dscp_38 parent=
flash_override_global priority=2 queue=ethernet-default
add name=flash_override_global_2 packet-mark=dscp_37 parent=
flash_override_global priority=3 queue=ethernet-default
add limit-at=3M max-limit=4M name=flash_override_global_3 packet-
mark=dscp_36 parent=flash_override_global priority=4 queue=ethernet-default
add name=flash_override_global_4 packet-mark=dscp_35 parent=
flash_override_global priority=5 queue=ethernet-default
add name=flash_override_global_5 packet-mark=dscp_34
parent=flash_override_global priority=6 queue=ethernet-default
add name=flash_override_global_6 packet-mark=dscp_33 parent=
flash_override_global priority=7 queue=ethernet-default
add name=flash_override_global_7 packet-mark=dscp_32 parent=
flash_override_global queue=ethernet-default
add name=flash_global parent=qos_global priority=5 queue=ethernet-default
add limit-at=3M max-limit=4M name=flash_global_0 packet-mark=dscp_31
parent= flash_global priority=1 queue=ethernet-default
add name=flash_global_1 packet-mark=dscp_30 parent=flash_global priority=2
queue=ethernet-default
add name=flash_global_2 packet-mark=dscp_29 parent=flash_global priority=3
queue=ethernet-default
add name=flash_global_3 packet-mark=dscp_28 parent=flash_global priority=4
queue=ethernet-default
add name=flash_global_4 packet-mark=dscp_27 parent=flash_global priority=5
queue=ethernet-default
add name=flash_global_5 packet-mark=dscp_26 parent=flash_global priority=6
queue=ethernet-default
add name=flash_global_6 packet-mark=dscp_25 parent=flash_global priority=7
queue=ethernet-default
add name=flash_global_7 packet-mark=dscp_24 parent=flash_global
queue=ethernet-default
add name=immedate_global parent=qos_global priority=6 queue=ethernet-
default
add name=immedate_global_0 packet-mark=dscp_23 parent=immedate_global
priority=1 queue=ethernet-default
add name=immedate_global_1 packet-mark=dscp_22 parent=immedate_global
priority=2 queue=ethernet-default
add name=immedate_global_2 packet-mark=dscp_21 parent=immedate_global
priority=3 queue=ethernet-default
add name=immedate_global_3 packet-mark=dscp_20 parent=immedate_global
priority=4 queue=ethernet-default
add name=immedate_global_4 packet-mark=dscp_19 parent=immedate_global
priority=5 queue=ethernet-default
add name=immedate_global_5 packet-mark=dscp_18 parent=immedate_global
priority=6 queue=ethernet-default
add name=immedate_global_6 packet-mark=dscp_17 parent=immedate_global
priority=7 queue=ethernet-default
add name=immedate_global_7 packet-mark=dscp_16 parent=immedate_global
queue= ethernet-default
add name=priority_global parent=qos_global priority=7 queue=ethernet-default
add name=priority_global_0 packet-mark=dscp_15 parent=priority_global
priority=1 queue=ethernet-default
add name=priority_global_1 packet-mark=dscp_14 parent=priority_global
priority=2 queue=ethernet-default
add name=priority_global_2 packet-mark=dscp_13 parent=priority_global
priority=3 queue=ethernet-default
add name=priority_global_3 packet-mark=dscp_12 parent=priority_global
priority=4 queue=ethernet-default
add name=priority_global_4 packet-mark=dscp_11 parent=priority_global
priority=5 queue=ethernet-default
add name=priority_global_5 packet-mark=dscp_10 parent=priority_global
priority=6 queue=ethernet-default
add name=priority_global_6 packet-mark=dscp_9 parent=priority_global
priority=7 queue=ethernet-default
add name=priority_global_7 packet-mark=dscp_8 parent=priority_global
queue= ethernet-default
add name=routine_global parent=qos_global queue=ethernet-default
add name=routine_global_0 packet-mark=dscp_7 parent=routine_global
priority=1 queue=ethernet-default
add name=routine_global_1 packet-mark=dscp_6 parent=routine_global
priority=2 queue=ethernet-default
add name=routine_global_2 packet-mark=dscp_5 parent=routine_global
priority=3 queue=ethernet-default
add name=routine_global_3 packet-mark=dscp_4 parent=routine_global
priority=4 queue=ethernet-default
add name=routine_global_4 packet-mark=dscp_3 parent=routine_global
priority=5 queue=ethernet-default
add name=routine_global_5 packet-mark=dscp_2 parent=routine_global queue=
ethernet-default
add name=routine_global_6 packet-mark=dscp_1 parent=routine_global
priority=7 queue=ethernet-default
add name=routine_global_7 packet-mark=dscp_0 parent=routine_global queue=
ethernet-default
/routing ospf area
add area-id=1.1.1.1 name=LAN
/routing ospf instance
set [ find default=yes ] mpls-te-area=backbone mpls-te-router-id=loopback \
redistribute-connected=as-type-1 router-id=10.255.255.1
/routing ospf-v3 area
set [ find default=yes ] disabled=yes
/routing ospf-v3 instance
set [ find default=yes ] disabled=yes
/interface bridge nat

```

```

add chain=srcnat out-bridge=Bridge_WDS
add chain=dstnat in-bridge=Bridge_WDS
# Bridge_ACAD not ready
add chain=srcnat out-bridge=*1B
# Bridge_ACAD not ready
add chain=dstnat in-bridge=*1B
# Bridge_VPLS not ready
add chain=srcnat out-bridge=*19
# Bridge_VPLS not ready
add chain=dstnat in-bridge=*19
/interface bridge port
add bridge=Bridge_WDS interface=ether2
add bridge=Bridge_WDS interface=VAP01
add bridge=Bridge_WDS interface=ether1
/interface bridge settings
set use-ip-firewall=yes
/ip settings
set accept-redirects=yes
/interface wireless access-list
add interface=wlan1 mac-address=00:02:6F:75:CF:12
add interface=wlan1 mac-address=00:02:6F:75:CF:00
add interface=wlan1 mac-address=00:02:6F:75:CF:26
add interface=wlan1 mac-address=00:02:6F:75:CF:18
/ip accounting
set account-local-traffic=yes enabled=yes
/ip address
add address=10.255.255.1 interface=loopback network=10.255.255.1
add address=10.255.254.12/24 interface=WDS_NetIFMTBVAP02
network=10.255.254.0
add address=10.255.254.13/24 interface=WDS_NetIFMTBVAP03
network=10.255.254.0
add address=10.255.254.14/24 interface=WDS_NetIFMTBVAP04
network=10.255.254.0
add address=10.255.254.15/24 interface=WDS_NetIFMTBVAP05
network=10.255.254.0
add address=172.16.246.13/23 interface=Bridge_WDS network=172.16.246.0
add address=192.168.100.1/24 network=192.168.100.0
add address=192.168.10.1/24 network=192.168.10.0
add address=10.255.253.13/24 disabled=yes network=10.255.253.0
/ip dns
set servers=172.16.246.1
/ip firewall address-list
add address=31.13.64.51 list=whatsapp
add address=172.16.247.207 list=Arduino
add address=172.16.247.121 list=ScadaBr
add address=172.16.246.3 list=Camera
add address=172.16.247.58 list=ScadaBr
/ip firewall mangle
add action=mark-connection chain=prerouting comment="Camera In" new-
connection-mark=con_camera_in src-address-list=Camera
add action=mark-packet chain=prerouting connection-mark=con_camera_in
new-packet-mark=pac_camera_in
add action=mark-routing chain=prerouting connection-mark=con_camera_in
disabled=yes new-routing-mark=route_camera_in packet-mark=pac_camera_in
add action=change-dscp chain=prerouting connection-mark=con_camera_in
disabled=yes new-dscp=36 packet-mark=pac_camera_in passthrough=no
add action=mark-connection chain=postrouting comment=Camera_Out_Camera
dst-address-list=Camera new-connection-mark=con_camera_out src-address-
list=ScadaBr
add action=mark-packet chain=postrouting connection-mark=con_camera_out
new-packet-mark=pac_camera_out
add action=mark-routing chain=prerouting connection-mark=con_camera_out
new-routing-mark=route_camera_out packet-mark=pac_camera_out
add action=change-dscp chain=postrouting connection-mark=con_camera_out
disabled=yes new-dscp=36 packet-mark=pac_camera_out passthrough=no
add action=mark-connection chain=prerouting comment=Arduino_In new-
connection-mark=con_arduino src-address-list=Arduino
add action=mark-packet chain=prerouting connection-mark=con_arduino
new-packet-mark=pac_arduino
add action=mark-routing chain=prerouting connection-mark=con_arduino
log=yes new-routing-mark=Route_Arduino
add action=change-dscp chain=prerouting connection-mark=con_arduino
disabled=yes new-dscp=31 packet-mark=pac_arduino passthrough=no
add action=set-priority chain=postrouting connection-mark=con_arduino
disabled=yes log=yes new-priority=1 packet-mark=pac_arduino passthrough=\
no
add action=mark-connection chain=postrouting
comment=ScadaBR_Out_Arduino \
dst-address-list=Arduino new-connection-mark=con_ScadaBR \
src-address-list=ScadaBr
add action=mark-routing chain=prerouting connection-mark=con_ScadaBR \
disabled=yes log=yes new-routing-mark=Route_ScadaBr
add action=mark-packet chain=postrouting connection-mark=con_ScadaBR \
new-packet-mark=pac_ScadaBR
add action=change-dscp chain=postrouting connection-mark=con_ScadaBR \
disabled=yes new-dscp=31 packet-mark=pac_ScadaBR passthrough=no
add action=set-priority chain=postrouting connection-mark=con_ScadaBR \
disabled=yes log=yes new-priority=1 packet-mark=pac_ScadaBR
passthrough=\
no
add action=mark-routing chain=output comment="Route VPLS" disabled=yes
log=\
yes new-routing-mark=TE_R01_03 out-interface=*21
add action=change-dscp chain=postrouting disabled=yes log=yes new-dscp=47 \
passthrough=no routing-mark=route_VPLS
add action=mark-connection chain=prerouting comment=whatsapp \
dst-address-list=whatsapp log=yes new-connection-mark=con_whatsapp
add action=mark-packet chain=postrouting connection-mark=con_whatsapp
log=yes \
new-packet-mark=pac_whatsapp
add action=change-dscp chain=postrouting connection-mark=con_whatsapp \

```

```

disabled=yes log=yes new-dscp=47 packet-mark=pac_whatsapp
passthrough=no
add action=set-priority chain=postrouting connection-mark=con_whatsapp \
disabled=yes log=yes new-priority=1 packet-mark=pac_whatsapp
passthrough=\
no
add action=mark-packet chain=prerouting comment="Marcado de paquetes
FTP" \
new-packet-mark=ftp_in passthrough=no protocol=tcp src-port=20
add action=change-dscp chain=postrouting disabled=yes packet-mark=ftp_in \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=ftp_in passthrough=no
add action=mark-packet chain=postrouting dst-port=20 new-packet-
mark=ftp_out \
passthrough=no protocol=tcp
add action=change-dscp chain=postrouting disabled=yes packet-mark=ftp_out \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=ftp_out passthrough=no
add action=mark-packet chain=prerouting new-packet-mark=ftp_21_in \
passthrough=no protocol=tcp src-port=21
add action=change-dscp chain=postrouting disabled=yes packet-mark=ftp_21_in
\
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=ftp_21_in passthrough=no
add action=mark-packet chain=postrouting dst-port=21 new-packet-mark=\
ftp_21_out passthrough=no protocol=tcp
add action=change-dscp chain=postrouting disabled=yes packet-
mark=ftp_21_out \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=ftp_21_out passthrough=no
add action=mark-packet chain=prerouting comment="Marcado de paquetes
SSH" \
new-packet-mark=ssh_in passthrough=no protocol=tcp src-port=22
add action=change-dscp chain=postrouting disabled=yes packet-mark=ssh_in \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=ssh_in passthrough=no
add action=mark-packet chain=postrouting dst-port=22 new-packet-
mark=ssh_out \
passthrough=no protocol=tcp
add action=change-dscp chain=postrouting disabled=yes packet-mark=ssh_out \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=ssh_out passthrough=no
add action=mark-packet chain=prerouting comment="Marcado de paquetes
TELNET" \
new-packet-mark=telnet_in passthrough=no protocol=tcp src-port=23
add action=change-dscp chain=postrouting disabled=yes packet-mark=telnet_in
\
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=telnet_in passthrough=no
add action=mark-packet chain=postrouting dst-port=23 new-packet-mark=\
telnet_out passthrough=no protocol=tcp
add action=change-dscp chain=postrouting disabled=yes packet-
mark=telnet_out \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=telnet_out passthrough=no
add action=mark-packet chain=prerouting comment="Marcado de paquetes
SMTP" \
new-packet-mark=smtp_in passthrough=no protocol=tcp src-port=25
add action=change-dscp chain=postrouting disabled=yes packet-mark=smtp_in \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=smtp_in passthrough=no
add action=mark-packet chain=postrouting dst-port=25 new-packet-
mark=smtp_out passthrough=no protocol=tcp
add action=change-dscp chain=postrouting disabled=yes packet-mark=smtp_out
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=smtp_out passthrough=no
add action=mark-packet chain=prerouting comment=\
"Marcado de paquetes SMTP sobre SSL" new-packet-mark=smtp_ssl_in \
passthrough=no protocol=tcp src-port=465
add action=change-dscp chain=postrouting disabled=yes packet-
mark=smtp_ssl_in \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=smtp_ssl_in passthrough=no
add action=mark-packet chain=postrouting dst-port=465 new-packet-mark=\
smtp_ssl_out passthrough=no protocol=tcp
add action=change-dscp chain=postrouting disabled=yes packet-mark=\
smtp_ssl_out passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=smtp_ssl_out passthrough=no
add action=mark-packet chain=prerouting comment="Marcado de paquetes
DNS" \
new-packet-mark=dns_in passthrough=no protocol=tcp src-port=53
add action=change-dscp chain=postrouting disabled=yes packet-mark=dns_in \
passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
packet-mark=dns_in passthrough=no
add action=mark-packet chain=postrouting dst-port=53 new-packet-
mark=dns_out \
passthrough=no protocol=tcp
add action=change-dscp chain=postrouting disabled=yes packet-mark=dns_out \

```



```

add action=change-dscp chain=postrouting disabled=yes packet-
mark=winbox_in \
    passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=winbox_in passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=winbox_out
protocol=
    tcp src-port=8291
add action=change-dscp chain=postrouting new-dscp=46 packet-
mark=winbox_out
add action=mark-packet chain=prerouting comment="Marcado de paquetes
P2P" \
    new-packet-mark=p2p_in p2p=all-p2p passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=p2p_out p2p=all-
p2p \
    passthrough=no
add action=change-dscp chain=postrouting disabled=yes packet-mark=p2p_out \
    passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=p2p_out passthrough=no
add action=mark-packet chain=prerouting new-packet-mark=emule_in
passthrough=
    no protocol=tcp src-port=4662
add action=change-dscp chain=postrouting disabled=yes packet-mark=emule_in
\
    passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=emule_in passthrough=no
add action=mark-packet chain=postrouting dst-port=4672 new-packet-mark=
    emule_out passthrough=no protocol=udp
add action=change-dscp chain=postrouting disabled=yes packet-
mark=emule_out \
    passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=emule_out passthrough=no
add action=mark-packet chain=prerouting new-packet-mark=gnutella_6346_in \
    passthrough=no protocol=tcp src-port=6346
add action=change-dscp chain=postrouting disabled=yes packet-mark=
    gnutella_6346_in passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=gnutella_6246_in passthrough=no
add action=mark-packet chain=postrouting dst-port=6346 new-packet-mark=
    gnutella_6346_out passthrough=no protocol=tcp
add action=change-dscp chain=postrouting disabled=yes packet-mark=
    gnutella_6346_out passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=gnutella_6346_out passthrough=no
add action=mark-packet chain=prerouting new-packet-mark=gnutella_6347_in \
    passthrough=no protocol=udp src-port=6347
add action=change-dscp chain=postrouting disabled=yes packet-mark=
    gnutella_6347_in passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=gnutella_6347_in passthrough=no
add action=mark-packet chain=postrouting dst-port=6347 new-packet-mark=
    gnutella_6347_out passthrough=no protocol=udp
add action=change-dscp chain=postrouting disabled=yes packet-mark=
    gnutella_6347_out passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=gnutella_6347_out passthrough=no
add action=mark-packet chain=prerouting new-packet-mark=gnutella_6348_in \
    passthrough=no protocol=udp src-port=6348
add action=change-dscp chain=postrouting disabled=yes packet-mark=
    gnutella_6348_in passthrough=no
add action=set-priority chain=postrouting disabled=yes new-priority=8 \
    packet-mark=gnutella_6348_in passthrough=no
add action=mark-packet chain=postrouting comment=dscp_0 dscp=0 \
    new-packet-mark=dscp_0 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_1 dscp=1 \
    new-packet-mark=dscp_1 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_2 dscp=2 \
    new-packet-mark=dscp_2 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_3 dscp=3 \
    new-packet-mark=dscp_3 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_4 dscp=4 \
    new-packet-mark=dscp_4 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_5 dscp=5 \
    new-packet-mark=dscp_5 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_6 dscp=6 \
    new-packet-mark=dscp_6 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_7 dscp=7 \
    new-packet-mark=dscp_7 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_8 dscp=8 \
    new-packet-mark=dscp_8 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_9 dscp=9 \
    new-packet-mark=dscp_9 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_10 dscp=10 \
    new-packet-mark=dscp_10 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_11 dscp=11 \
    new-packet-mark=dscp_11 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_12 dscp=12 \
    new-packet-mark=dscp_12 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_13 dscp=13 \
    new-packet-mark=dscp_13 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_14 dscp=14 \
    new-packet-mark=dscp_14 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_15 dscp=15 \
    new-packet-mark=dscp_15 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_16 dscp=16 \
    new-packet-mark=dscp_16 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_17 dscp=17 \
    new-packet-mark=dscp_17 passthrough=no

```

[illegible]

```

new-packet-mark=dscp_59 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_58 dscp=58 \
new-packet-mark=dscp_58 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_57 dscp=57 \
new-packet-mark=dscp_57 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_56 dscp=56 \
new-packet-mark=dscp_56 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_55 dscp=55 \
new-packet-mark=dscp_55 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_54 dscp=54 \
new-packet-mark=dscp_54 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_53 dscp=53 \
new-packet-mark=dscp_53 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_52 dscp=52 \
new-packet-mark=dscp_52 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_51 dscp=51 \
new-packet-mark=dscp_51 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_50 dscp=50 \
new-packet-mark=dscp_50 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_49 dscp=49 \
new-packet-mark=dscp_49 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_48 dscp=48 \
new-packet-mark=dscp_48 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_47 dscp=47 \
new-packet-mark=dscp_47 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_46 dscp=46 \
new-packet-mark=dscp_46 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_45 dscp=45 \
new-packet-mark=dscp_45 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_44 dscp=44 \
new-packet-mark=dscp_44 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_43 dscp=43 \
new-packet-mark=dscp_43 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_42 dscp=42 \
new-packet-mark=dscp_42 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_41 dscp=41 \
new-packet-mark=dscp_41 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_40 dscp=40 \
new-packet-mark=dscp_40 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_39 dscp=39 \
new-packet-mark=dscp_39 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_38 dscp=38 \
new-packet-mark=dscp_38 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_37 dscp=37 \
new-packet-mark=dscp_37 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_36 dscp=36 \
new-packet-mark=dscp_36 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_35 dscp=35 \
new-packet-mark=dscp_35 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_34 dscp=34 \
new-packet-mark=dscp_34 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_33 dscp=33 \
new-packet-mark=dscp_33 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_32 dscp=32 \
new-packet-mark=dscp_32 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_31 dscp=31 \
new-packet-mark=dscp_31 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_30 dscp=30 \
new-packet-mark=dscp_30 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_29 dscp=29 \
new-packet-mark=dscp_29 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_28 dscp=28 \
new-packet-mark=dscp_28 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_27 dscp=27 \
new-packet-mark=dscp_27 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_26 dscp=26 \
new-packet-mark=dscp_26 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_25 dscp=25 \
new-packet-mark=dscp_25 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_24 dscp=24 \
new-packet-mark=dscp_24 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_23 dscp=23 \
new-packet-mark=dscp_23 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_22 dscp=22 \
new-packet-mark=dscp_22 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_21 dscp=21 \
new-packet-mark=dscp_21 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_20 dscp=20 \
new-packet-mark=dscp_20 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_19 dscp=19 \
new-packet-mark=dscp_19 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_18 dscp=18 \
new-packet-mark=dscp_18 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_17 dscp=17 \
new-packet-mark=dscp_17 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_16 dscp=16 \
new-packet-mark=dscp_16 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_15 dscp=15 \
new-packet-mark=dscp_15 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_14 dscp=14 \
new-packet-mark=dscp_14 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_13 dscp=13 \
new-packet-mark=dscp_13 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_12 dscp=12 \
new-packet-mark=dscp_12 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_11 dscp=11 \
new-packet-mark=dscp_11 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_10 dscp=10 \
new-packet-mark=dscp_10 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_9 dscp=9 \
new-packet-mark=dscp_9 passthrough=no

```

```

add action=mark-packet chain=postrouting comment=dscp_8 dscp=8 \
new-packet-mark=dscp_8 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_7 dscp=7 \
new-packet-mark=dscp_7 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_6 dscp=6 \
new-packet-mark=dscp_6 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_5 dscp=5 \
new-packet-mark=dscp_5 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_4 dscp=4 \
new-packet-mark=dscp_4 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_3 dscp=3 \
new-packet-mark=dscp_3 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_2 dscp=2 \
new-packet-mark=dscp_2 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_1 dscp=1 \
new-packet-mark=dscp_1 passthrough=no
add action=mark-packet chain=postrouting comment=dscp_0 dscp=0 \
new-packet-mark=dscp_0 passthrough=no
/ip firewall nat
add action=masquerade chain=srcnat log=yes out-interface=Bridge_WDS \
src-address=192.168.0.0/16
/ip route
add distance=1 gateway=172.16.246.1
add distance=1 dst-address=10.255.255.2/32 gateway=\
172.16.246.14,192.168.100.2
add distance=1 dst-address=10.255.255.3/32 gateway=\
172.16.246.15,192.168.100.3
add distance=1 dst-address=10.255.255.4/32 gateway=\
172.16.246.16,192.168.100.4
add distance=1 dst-address=10.255.255.5/32 gateway=\
192.168.100.5,172.16.246.17
/mps ldp
set enabled=yes loop-detect=yes lsr-id=10.255.255.1 transport-address=\
10.255.255.1
/mps ldp interface
add interface=loopback
add interface=Bridge_WDS
/routing ospf network
add area=backbone network=10.255.255.0/24
add area=backbone network=10.255.254.0/24
add area=backbone network=172.16.246.0/23
/system script
add name=MarcarPacotesDSCP owner=admin policy=\
ftp,reboot,read,write,policy,test,password,sniff,sensitive source="\
for x \
from 0 to 63 do={/ip firewall mangle add action=mark-packet chain=postrout\
ing \\\r\
\ncoment=("${dscp}_\" . \${x} ) disabled=no dscp=${x} new-packet-
mark=("${dscp}_\" . \${x} passthrough=no)"
add name=QueueBridge_WDS owner=admin policy=\
ftp,reboot,read,write,policy,test,password,sniff,sensitive source="\
tree\r\
\nadd burst-limit=0 burst-threshold=0 burst-time=0s disabled=no limit-at=0\
\_max-limit=5000000 name=Bridge_WDS \\\r\
\nparent=Bridge_WDS queue=default\r\
\n\r\
\n#prio8\r\
\n:for z from 0 to 7 do={/queue tree add burst-limit=0 burst-threshold=0 b\
urst-time=0s disabled=no limit-at=0 max-limit=0 \\\r\
\nname=("${routine}_\" . \${z} . \"_Bridge_WDS\") packet-mark=("${dscp}_\" . \${z}\
\_\"_Bridge_WDS\") parent=Bridge_WDS priority=8 queue=ethernet-
default)\
\r\
\n\r\
\n#prio7\r\
\n:for z from 8 to 15 do={/queue tree add burst-limit=0 burst-threshold=0 \
burst-time=0s disabled=no limit-at=0 max-limit=0 \\\r\
\nname=("${priority}_\" . \${z} . \"_Bridge_WDS\") packet-mark=("${dscp}_\" . \${z}\
\_\"_Bridge_WDS\") parent=Bridge_WDS priority=7 queue=ethernet-
default)\
\r\
\n\r\
\n#prio6\r\
\n:for z from 16 to 23 do={/queue tree add burst-limit=0 burst-threshold=0\
\_burst-time=0s disabled=no limit-at=0 max-limit=0 \\\r\
\nname=("${immediate}_\" . \${z} . \"_Bridge_WDS\") packet-mark=("${dscp}_\" . \
\${z} . \"_Bridge_WDS\") parent=Bridge_WDS priority=6 queue=ethernet-
default)\
\r\
\n\r\
\n#prio5\r\
\n:for z from 24 to 31 do={/queue tree add burst-limit=0 burst-threshold=0\
\_burst-time=0s disabled=no limit-at=0 max-limit=0 \\\r\
\nname=("${flash}_\" . \${z} . \"_Bridge_WDS\") packet-mark=("${dscp}_\" . \${z}\
\_\"_Bridge_WDS\") parent=Bridge_WDS priority=5 queue=ethernet-
default)\
\r\
\n\r\
\n#prio4\r\
\n:for z from 32 to 39 do={/queue tree add burst-limit=0 burst-threshold=0\
\_burst-time=0s disabled=no limit-at=0 max-limit=0 \\\r\
\nname=("${flash_override}_\" . \${z} . \"_Bridge_WDS\") packet-mark=("${dscp}_\" . \
\${z} . \"_Bridge_WDS\") parent=Bridge_WDS priority=4 queue=ethernet-
default)\
\r\
\n\r\
\n#prio3\r\
\n:for z from 40 to 47 do={/queue tree add burst-limit=0 burst-threshold=0\
\_burst-time=0s disabled=no limit-at=0 max-limit=0 \\\r\
\nname=("${critical}_\" . \${z} . \"_Bridge_WDS\") packet-mark=("${dscp}_\" . \${z}\
\_\"_Bridge_WDS\") parent=Bridge_WDS priority=3 queue=ethernet-
default)\
\r\
\n\r\

```



```

\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
HT)
TPS\ " disabled=no new-packet-mark=https_in passthrough=no \\\r\
\n protocol=tcp src-port=443\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=443 new-
pa)
cket-mark=https_out passthrough=no protocol=tcp\r\
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
MS\
N\ " disabled=no new-packet-mark=msn_in passthrough=no \\\r\
\n protocol=tcp src-port=1863\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=1863 new-
p)
cket-mark=msn_out passthrough=no protocol=tcp\r\
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
Vo IP\ " disabled=no new-packet-mark=voip_in passthrough=no \\\r\
\n protocol=udp src-port=5060\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=5060 new-
p acket-mark=voip_out passthrough=no protocol=udp\r\
\nadd action=mark-packet chain=prerouting disabled=no new-packet-
mark=voip_5061_in passthrough=no protocol=tcp src-port=5061\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=5061 new-
p acket-mark=voip_5061_out passthrough=no protocol=tcp\r\
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
MS\
N archivos\ " disabled=no new-packet-mark=msn_files_in \\\r\
\n passthrough=no protocol=tcp src-port=6891-6900\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=6891-6900
\
new-packet-mark=msn_files_out passthrough=no protocol=tcp\r\
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
MS\
N voz\ " disabled=no new-packet-mark=msn_voz_in passthrough=no \\\r\
\n protocol=tcp src-port=6901\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=6901 new-
p acket-mark=msn_voz_out passthrough=no protocol=tcp\r\
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
WI NBOX\ " disabled=no dst-port=8291 new-packet-mark=winbox_in \\\r\
\n passthrough=no protocol=tcp\r\
\nadd action=mark-packet chain=postrouting disabled=no new-packet-
mark=win box_out passthrough=no protocol=tcp src-port=8291\r\
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
P2 P\ " disabled=no new-packet-mark=p2p_in p2p=all-p2p \\\r\
\n passthrough=no\r\
\nadd action=mark-packet chain=postrouting disabled=no new-packet-
mark=p2p)
_out p2p=all-p2p passthrough=no\r\
\nadd action=mark-packet chain=prerouting disabled=no new-packet-
mark=emul e_in passthrough=no protocol=tcp src-port=4662\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=4672 new-
p acket-mark=emule_out passthrough=no protocol=udp\r\
\nadd action=mark-packet chain=prerouting disabled=no new-packet-
mark=gnut ella_6346_in passthrough=no protocol=tcp src-port=6346\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=6346 new-
p acket-mark=gnutella_6346_out passthrough=no protocol=tcp\r\
\nadd action=mark-packet chain=prerouting disabled=no new-packet-
mark=gnut ella_6347_in passthrough=no protocol=udp src-port=6347\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=6347 new-
packet-mark=gnutella_6347_out passthrough=no protocol=udp\r\
\nadd action=mark-packet chain=prerouting disabled=no new-packet-
mark=gnut ella_6348_in passthrough=no protocol=udp src-port=6348\r\
\nadd action=mark-packet chain=postrouting disabled=no dst-port=6348 new-
p acket-mark=gnutella_6348_out passthrough=no protocol=udp"
add name=01-MarcarPacote owner=admin policy=\
ftp,reboot,read,write,policy,test,password,sniff,sensitive source="/ip fir\
ewall address-list add address="31.13.64.51/32" list="\whatsapp"\r\
\n/ip firewall address-list add address="31.13.65.49/32" list="\whatsapp\
"\r\
\n/ip firewall address-list add list=Arduino address=172.16.247.221/32\r\
\n\r\
\n/ip firewall address-list add list=ScadaBr address=172.16.247.24/32\r\
\n\r\
\n/ip firewall mangle\r\
\nadd action=mark-connection chain=prerouting comment=Arduino dst-
address-\
list=Arduino\\\r\
\n log=yes new-connection-mark=con_arduino\r\
\nadd action=mark-packet chain=postrouting connection-mark=con_arduino
log\
=\\\r\
\n yes new-packet-mark=pac_arduino\r\
\nadd action=change-dscp chain=postrouting connection-mark=con_arduino
\\\
\r\
\n disabled=yes log=yes new-dscp=47 packet-mark=pac_arduino
passthrough\
=no\r\
\nadd action=set-priority chain=postrouting connection-mark=con_arduino \\\
\r\
\n disabled=yes log=yes new-priority=1 packet-mark=pac_arduino \\\r\
\n passthrough=no\r\
\n\r\
\nadd action=mark-connection chain=prerouting comment=ScadaBR dst-
address=\
\\\r\
\n 172.16.247.24 log=yes new-connection-mark=con_ScadaBR\r\
\nadd action=mark-packet chain=postrouting connection-mark=con_ScadaBR
log\
=yes \\\r\
\n new-packet-mark=pac_ScadaBR\r\
\nadd action=change-dscp chain=postrouting connection-mark=con_ScadaBR
log\
=yes \\\r\
\n new-dscp=47 packet-mark=pac_ScadaBR passthrough=no\r\
\nadd action=set-priority chain=postrouting connection-mark=con_ScadaBR
\\\
\r\
\n disabled=yes log=yes new-priority=1 packet-mark=pac_ScadaBR
passthro\
ugh=\\\r\
\n no\r\
\n\r\
\nadd action=mark-routing chain=output comment="\Route VPLS\ " log=yes
\\\r\
\n new-routing-mark=route_VPLS out-interface=Bridge_WDS\r\
\nadd action=change-dscp chain=postrouting log=yes new-dscp=1 routing-
mark\
=\\\r\
\n route_VPLS\r\
\n\r\
\nadd action=mark-connection chain=prerouting comment=whatsapp dst-
address\
-list=whatsapp\\\r\
\n log=yes new-connection-mark=con_whatsapp\r\
\nadd action=mark-packet chain=postrouting connection-mark=con_whatsapp
lo\
g=\\\r\
\n yes new-packet-mark=pac_whatsapp\r\
\nadd action=change-dscp chain=postrouting connection-mark=con_whatsapp
\\\
\r\
\n disabled=yes log=yes new-dscp=47 packet-mark=pac_whatsapp
passthrough\
h=no\r\
\nadd action=set-priority chain=postrouting connection-mark=con_whatsapp \
\\\r\
\n disabled=yes log=yes new-priority=1 packet-mark=pac_whatsapp \\\r\
\n passthrough=no\r\
\n\r\
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
FTP\ " \\\r\
\n new-packet-mark=ftp_in passthrough=no protocol=tcp src-port=20\r\
\nadd action=change-dscp chain=postrouting \\\r\
\n disabled=yes log=no new-dscp=00 packet-mark=ftp_in passthrough=no\r\
\nadd action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=ftp_in \\\r\
\n passthrough=no\r\
\nadd action=mark-packet chain=postrouting dst-port=20 new-packet-
mark=ftp_out \\\r\
\n passthrough=no protocol=tcp\r\
\nadd action=change-dscp chain=postrouting \\\r\
\n disabled=yes log=no new-dscp=00 packet-mark=ftp_out
passthrough=no\r\
\nadd action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=ftp_out \\\r\
\n passthrough=no\r\
\n\r\
\nadd action=mark-packet chain=prerouting new-packet-mark=ftp_21_in \\\r\
\n passthrough=no protocol=tcp src-port=21\r\
\nadd action=change-dscp chain=postrouting \\\r\
\n disabled=yes log=no new-dscp=00 packet-mark=ftp_21_in
passthrough=no\
\r\
\nadd action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=ftp_21_in \\\r\
\n passthrough=no\r\
\nadd action=mark-packet chain=postrouting dst-port=21 new-packet-
mark=\\\
\r\
\n ftp_21_out passthrough=no protocol=tcp\r\
\nadd action=change-dscp chain=postrouting \\\r\
\n disabled=yes log=no new-dscp=00 packet-mark=ftp_21_out
passthrough=n\
o\r\
\nadd action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=ftp_21_out \\\r\
\n passthrough=no\r\
\n\r\
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
SS\
H\ " \\\r\
\n new-packet-mark=ssh_in passthrough=no protocol=tcp src-port=22\r\
\nadd action=change-dscp chain=postrouting \\\r\
\n disabled=yes log=no new-dscp=00 packet-mark=ssh_in
passthrough=no\r\
\nadd action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=ssh_in \\\r\
\n passthrough=no\r\
\nadd action=mark-packet chain=postrouting dst-port=22 new-packet-
mark=ssh\
_out \\\r\
\n passthrough=no protocol=tcp\r\
\nadd action=change-dscp chain=postrouting \\\r\
\n disabled=yes log=no new-dscp=00 packet-mark=ssh_out
passthrough=no\r\
\nadd action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=ssh_out \\\r\
\n passthrough=no\r\
\n\r\
\n\r\

```

```

\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
TE\
LNET\ "\r\r
\n new-packet-mark=telnet_in passthrough=no protocol=tcp src-port=23\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=telnet_in
passthrough=no\
\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=telnet_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=23 new-packet-
mark=\\
\r\r
\n telnet_out passthrough=no protocol=tcp\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=telnet_out
passthrough=n\
o\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=telnet_out \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
SM\
TP\ "\r\r
\n new-packet-mark=smtp_in passthrough=no protocol=tcp src-port=25\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=smtp_in
passthrough=no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=smtp_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=25 new-packet-
mark=smtp\
p_out \r\r
\n passthrough=no protocol=tcp\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=smtp_out
passthrough=no\
\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=smtp_out \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=prerouting comment=\\r\r
\n "\Marcado de paquetes SMTP sobre SSL\ " new-packet-
mark=smtp_ssl_in \
\r\r
\n passthrough=no protocol=tcp src-port=465\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=smtp_ssl_in
passthrough=\
no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=smtp_ssl_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=465 new-packet-
mark=\\
\r\r
\n smtp_ssl_out passthrough=no protocol=tcp\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=smtp_ssl_out
passthrough\
=no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=smtp_ssl_out \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
DN\
S\ "\r\r
\n new-packet-mark=dns_in passthrough=no protocol=tcp src-port=53\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=dns_in
passthrough=no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=dns_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=53 new-packet-
mark=dns\
_out \r\r
\n passthrough=no protocol=tcp\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=dns_out
passthrough=no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=dns_out \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=prerouting new-packet-mark=dns_udp_in
\r\r
\n passthrough=no protocol=udp src-port=53\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=dns_udp_in
passthrough=n\
o\r\r

```

```

\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=dns_udp_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=53 new-packet-
mark=\\
\r\r
\n dns_udp_out passthrough=no protocol=udp\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=dns_udp_out
passthrough=\
no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=dns_udp_out \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
WW\
W\ "\r\r
\n new-packet-mark=www_in passthrough=no protocol=tcp src-port=80\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=www_in
passthrough=no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=www_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=80 new-packet-
mark=www\
_out \r\r
\n passthrough=no protocol=tcp\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=www_out
passthrough=no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=www_out \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
PO\
P3\ "\r\r
\n new-packet-mark=pop3_in passthrough=no protocol=tcp src-port=110\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=pop3_in
passthrough=no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=pop3_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=110 new-packet-
mark=\\
\r\r
\n pop3_out passthrough=no protocol=tcp\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=pop3_out
passthrough=no\
\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=pop3_out \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=prerouting new-packet-mark=pop3_995_in
\r\r
\n passthrough=no protocol=tcp src-port=995\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=pop3_995_in
passthrough=\
no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=pop3_995_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=995 new-packet-
mark=\\
\r\r
\n pop3_995_up passthrough=no protocol=tcp\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=pop3_995_up
passthrough=\
no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=pop3_995_up \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=prerouting comment="\Marcado de paquetes
SQ\
L\ "\r\r
\n new-packet-mark=sql_in passthrough=no protocol=tcp src-port=118\r\r
\nadd action=change-dscp chain=postrouting \r\r
\n disabled=yes log=no new-dscp=00 packet-mark=sql_in
passthrough=no\r\r
\nadd action=set-priority chain=postrouting \r\r
\n disabled=yes log=no new-priority=8 packet-mark=sql_in \r\r
\n passthrough=no\r\r
\r\r
\nadd action=mark-packet chain=postrouting dst-port=118 new-packet-
mark=sql\
_up \r\r
\n passthrough=no protocol=tcp\r\r
\nadd action=change-dscp chain=postrouting \r\r

```



```
\n voip_5061_out passthrough=no protocol=tcp\r\n\nadd action=change-dscp chain=postrouting \\\r\n\n disabled=yes log=no new-dscp=00 packet-mark=voip_5061_out  
passthrough\  
h=no\r\n\n\nadd action=set-priority chain=postrouting \\\r\n\n disabled=yes log=no new-priority=8 packet-mark=voip_5061_out \\\r\n\n passthrough=no\r\n\n\r\n\nadd action=mark-packet chain=prerouting comment=\"Marcado de paquetes  
WT\  
NBOX\" \\\r\n\n dst-port=8291 new-packet-mark=winbox_in protocol=tcp\r\n\nadd action=change-dscp chain=postrouting \\\r\n\n disabled=yes log=no new-dscp=00 packet-mark=winbox_in  
passthrough=no\  
\r\n\n\nadd action=set-priority chain=postrouting \\\r\n\n disabled=yes log=no new-priority=8 packet-mark=winbox_in \\\r\n\n passthrough=no\r\n\n\r\n\r\n\nadd action=mark-packet chain=postrouting new-packet-mark=winbox_out  
prot  
ocol=\\\r\n\n tcp src-port=8291\r\n\nadd action=change-dscp chain=postrouting new-dscp=46 packet-  
mark=winbox_  
out\r\n\n\nadd action=mark-packet chain=prerouting comment=\"Marcado de paquetes  
P2\  
P\" \\\r\n\n new-packet-mark=p2p_in p2p=all-p2p passthrough=no\r\n\n\r\n\nadd action=mark-packet chain=postrouting new-packet-mark=p2p_out  
p2p=all  
-p2p \\\r\n\n passthrough=no\r\n\n\nadd action=change-dscp chain=postrouting \\\r\n\n disabled=yes log=no new-dscp=00 packet-mark=p2p_out  
passthrough=no\r\n\n\nadd action=set-priority chain=postrouting \\\r\n\n disabled=yes log=no new-priority=8 packet-mark=p2p_out \\\r\n\n passthrough=no\r\n\n\r\n\r\n\nadd action=mark-packet chain=prerouting new-packet-mark=emule_in  
passth\r\nough=\\\r\n\n no protocol=tcp src-port=4662\r\n\n\nadd action=change-dscp chain=postrouting \\\r\n\n disabled=yes log=no new-dscp=00 packet-mark=emule_in  
passthrough=no\  
\r\n\n\nadd action=set-priority chain=postrouting \\\r\n\n disabled=yes log=no new-priority=8 packet-mark=emule_in \\\r\n\n passthrough=no\r\n\n\nadd action=mark-packet chain=postrouting dst-port=4672 new-packet-  
mark=  
\\\r\n\n emule_out passthrough=no protocol=udp\r\n\n\nadd action=change-dscp chain=postrouting \\\r\n\n disabled=yes log=no new-dscp=00 packet-mark=emule_out  
passthrough=no\  
\r\n\n\nadd action=set-priority chain=postrouting \\\r\n\n disabled=yes log=no new-priority=8 packet-mark=emule_out \\\r\n\n passthrough=no\r\n\n\r\n\r\n\nadd action=mark-packet chain=prerouting new-packet-  
mark=gnutella_6346_in\  
_\r\n\n\n passthrough=no protocol=tcp src-port=6346\r\n\n\nadd action=change-dscp chain=postrouting \\\r\n\n disabled=yes log=no new-dscp=00 packet-mark=gnutella_6346_in  
passth\r\nough=no\r\n\n\nadd action=set-priority chain=postrouting \\\r\n\n disabled=yes log=no new-priority=8 packet-mark=gnutella_6246_in \\\r\n\n passthrough=no\r\n\n\nadd action=mark-packet chain=postrouting dst-port=6346 new-packet-  
mark=  
\\\r\n\n gnutella_6346_out passthrough=no protocol=tcp\r\n\n\nadd action=change-dscp chain=postrouting \\\r\n\n disabled=yes log=no new-dscp=00 packet-mark=gnutella_6346_out  
passth\r\nough=no\r\n\n\nadd action=set-priority chain=postrouting \\\r\n\n disabled=yes log=no new-priority=8 packet-mark=gnutella_6346_out \\\r\n\n passthrough=no\r\n\n\r\n\r\n\nadd action=mark-packet chain=prerouting new-packet-  
mark=gnutella_6347_in\  
_\r\n\n\n passthrough=no protocol=udp src-port=6347\r\n\n\nadd action=change-dscp chain=postrouting \\\r\n\n disabled=yes log=no new-dscp=00 packet-mark=gnutella_6347_in  
passth\r\nough=no\r\n\n
```

```

\add action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=gnutella_6347_in \\\r\
\n passthrough=no\r\
\add action=mark-packet chain=postrouting dst-port=6347 new-packet-
mark=\
\\\r\
\n gnutella_6347_out passthrough=no protocol=udp\r\
\add action=change-dscp chain=postrouting \\\r\
\n disabled=yes log=no new-dscp=00 packet-mark=gnutella_6347_out
passth\
rough=no\r\
\add action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=gnutella_6347_out \\\
\r\
\n passthrough=no\r\
\n\r\
\add action=mark-packet chain=prerouting new-packet-
mark=gnutella_6348_in\
_\|\r\
\n passthrough=no protocol=udp src-port=6348\r\
\add action=change-dscp chain=postrouting \\\r\
\n disabled=yes log=no new-dscp=00 packet-mark=gnutella_6348_in
passth\
ough=no\r\
\add action=set-priority chain=postrouting \\\r\
\n disabled=yes log=no new-priority=8 packet-mark=gnutella_6348_in \\\r\
\n passthrough=no\r\
\n"

add name=02-DSCPHTB owner=admin policy=\
ftp,reboot,read,write,policy,test,password,sniff,sensitive source=":for x \
from 0 to 63 do={/ip firewall mangle add action=mark-packet chain=postrout\
ing \\\r\
\comment=(\'dscp_\' . \$x ) disabled=no dscp=\$x new-packet-
mark=(\'dscp_\
\' . \$x) passthrough=no}\r\
\n\r\
\n#Set interface here\r\
\n:global outboundInterface \'\'global\'\'r\
\n#Set bandwidth of the interface (remember, this is for OUTGOING)\r\
\n:global interfaceBandwidth 0\r\
\n#Set where in the chain the packets should be mangled\r\
\n:global mangleChain postrouting\r\
\n\r\
\n#Don't mess with these. They set the parameters for what is to follow\r\
\n:global queueName (\'\'qos_\' . \$outboundInterface)\r\
\n:global qosClasses {toarray \'\'netcon,intercon,critical,flash_override,f\
lash,immedate,priority,routine\'\'}\r\
\n:global qosIndex 64\r\
\n\r\
\n#Set up mangle rules for all 64 DSCP marks\r\
\n#This is different in that the highest priority packets are mangled first\
\r\
\n:for indexA from 63 to 0 do={/r\
\n/ip firewall mangle add \\\r\
\n\taction=mark-packet \\\r\
\n\tchain=\$mangleChain \\\r\
\n\tcomment=(\'\'dscp_\' . \$indexA) \\\r\
\n\t\tdisabled=no \\\r\
\n\t\tdscp=\$indexA \\\r\
\n\t\tnew-packet-mark=(\'\'dscp_\' . \$indexA) \\\r\
\n\t\tpassthrough=no\r\
\n}\r\
\n\r\
\n\r\
\n#Add a base queue to the queue tree for the outbound interface\r\
\n:queue tree add \\\r\
\n\tmax-limit=\$interfaceBandwidth \\\r\
\n\tname=\$queueName \\\r\
\n\t\tparent=\$outboundInterface \\\r\
\n\t\tpriority=1\r\
\n\r\
\n#Set up queues in queue tree for all 64 classes, subdivided by 8.\r\
\n:for indexA from=0 to=7 do={/r\
\n\t:local subClass [{pick \$qosClasses \$indexA} . \'_\' . \$outboundInt\
erface]\r\
\n\t:queue tree add \\\r\
\n\t\tname=\$subClass \\\r\
\n\t\t\tparent=\$queueName \\\r\
\n\t\t\tpriority=(\$indexA+1) \\\r\
\n\t\t\tqueue=ethernet-default\r\
\n\t:for indexB from=0 to=7 do={/r\
\n\t\t:set qosIndex (\$qosIndex-1)\r\
\n\t\t:queue tree add \\\r\
\n\t\t\tname=(\$subClass . \'_\' . \$indexB) \\\r\
\n\t\t\t\tparent=\$subClass \\\r\
\n\t\t\t\tpriority=(\$indexB+1) \\\r\
\n\t\t\t\t(packet-mark=(\'\'dscp_\' . \$qosIndex) \\\r\
\n\t\t\t\t\tqueue=ethernet-default\r\
\n\t\t\t)\r\
\n\t}"

/tool bandwidth-server
set authenticate=no
/tool romon port
add
/tool sniffer
set filter-ip-address=172.16.246.195/32,172.16.247.24/32,172.16.247.90/32 \
filter-stream=yes streaming-enabled=yes streaming-server=172.16.247.58

```