UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE TECNOLOGIA E GEOCIÊNCIAS

PROGRAMA DE PÓS – GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Análise e Otimização de Roteamento em Backbones OSPF Utilizando MPLS-TE.

Elaborado por:

José Mário Alexandre Melo de Oliveira

RECIFE, 28 de Fevereiro 2011

UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE TECNOLOGIA E GEOCIÊNCIAS

PROGRAMA DE PÓS – GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Análise e Otimização de Roteamento em Backbones OSPF Utilizando MPLS-TE.

por

José Mário Alexandre Melo de Oliveira

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco como parte dos requisitos para a obtenção do grau de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. RAFAEL DUEIRE LINS

RECIFE, 28 de Fevereiro 2011.

Catalogação na fonte
Bibliotecária Rosineide Mesquita Gonçalves Luz / CRB4-1361 (BCTG)

O48a **Oliveira, José Mário Alexandre Melo de.**
Análise e otimização de roteamento em Backbones OSPF utilizando MPLS-TE / José Mário Alexandre Melo de Oliveira. - Recife: O Autor, 2011
xiii, 177f., il., figs., gráfs., tabs.

Orientador : Prof. Dr. Rafael Dueire Lins.

Dissertação (Mestrado) - Universidade Federal de Pernambuco. CTG. Programa de Pós Graduação em Engenharia Elétrica, 2011.
Inclui Referências Bibliográficas e Anexos.

1.Engenharia Elétrica 2. Telecomunicações. 3.Redes. 4. IP. 5. MPLS. 5. OSPF. I. Lins, Rafael Dueire. II. Título.

621.3 CDD (22.ed) UFPE/BCTG-100/2011

Universidade Federal de Pernambuco
Pós-Graduação em Engenharia Elétrica

PARECER DA COMISSÃO EXAMINADORA DE DEFESA DE DISSERTAÇÃO DO MESTRADO ACADÊMICO DE

JOSÉ MÁRIO ALEXANDRE MELO DE OLIVEIRA

TÍTULO

"ANÁLISE E OTIMIZAÇÃO DE ROTEAMENTO EM *BACKBONES* OSPF UTILIZANDO MPLS-TE"

A comissão examinadora composta pelos professores: RAFAEL DUEIRE LINS, DES/UFPE, JOAQUIM FERREIRA MARTINS FILHO, DES/UFPE e CARMELO JOSÉ ALBANEZ BASTOS FILHO, DSC/UPE sob a presidência do primeiro, consideram o candidato **JOSÉ MÁRIO ALEXANDRE MELO DE OLIVEIRA APROVADO.**

Recife, 28 de fevereiro de 2011

CECÍLIO JOSÉ LINS PIMENTEL
Vice-Coordenador do PPGEE

RAFAEL DUEIRE LINS
Orientador e Membro Titular Interno

CARMELO JOSÉ ALBANEZ BASTOS FILHO
Membro Titular Externo

JOAQUIM FERREIRA MARTINS FILHO
Membro Titular Interno

Agradecimentos

Antes de iniciar agradecendo as pessoas que me ajudaram diretamente e indiretamente no desenvolvimento deste trabalho agradeço a Deus, pois durante vários momentos ele me ajudou a não desistir mesmo quando tinha que trabalhar durante toda madrugada e no amanhecer do dia ir assistir aula ou até fazer prova.

Agradeço a minha esposa Irene e meu filho Dante, pois os privei em muitos momentos da minha presença, pois estava estudando ou compensando horário do trabalho que tinha faltado para ir para as aulas do mestrado.

A minha esposa Irene agradeço por ter lido e relido diversas vezes esta dissertação buscando o entendimento de uma área completamente distinta de sua formação.

Agradeço ao meu orientador professor Rafael Dueire por ter me aceitado como aluno e ter me orientado durante o desenvolvimento desta dissertação.

Gostaria de agradecer aos professores Valdemar Rocha e Cecicilo Pimentel pelas cartas de recomendação, dando voto de confiança no meu ingresso no mestrado.

A meu amigo Roberto Mendonça agradeço, pois em vários domingos tirou muitas das minhas duvidas e ficou em casa discutindo comigo as soluções implementadas nesta dissertação.

A meu coordenador na Claro, empresa onde trabalho, Lindberg Tertuliano, por ter ajudado na liberação dos equipamentos para montagem do laboratório de teste.

A minha mãe Osilda, por ter feito todo o esforço necessário de mostrar o caminho do estudo, fazendo de mim um vencedor, pois consegui chegar até aqui.

Agradeço aos professores Carmelo José Albanez e Joaquim Ferreira Martins, membros da banca, por terem lido esta dissertação e contribuído com seus comentários.

Resumo

O aumento da demanda pela utilização de novos serviços de valor agregado sobre as infraestruturas de rede baseadas no protocolo IP (*Internet Protocol*), tais como voz e vídeo, contribui para que as operadoras de telecomunicações enfrentem um grande desafio para prover um *backbone* estável, escalável e otimizado, para atender satisfatoriamente essas novas expectativas de tráfego dos seus clientes. As empresas de telecomunicações buscam uma rede com uma maior disponibilidade e menor custo. A tecnologia MPLS (*Multi-Protocol Label Switching*) é indicada para prover evolução, otimização e flexibilidade aos *backbones* atuais, mostrando-se uma tecnologia emergente a ser empregada nessas redes.

O novo paradigma de transporte baseado no MPLS permite as operadoras de telecomunicações construírem, sobre as infraestruturas IP, ATM (*Asynchronous Transfer Mode*) e *Frame Relay*, um conjunto de soluções criativas e flexíveis para suprir essa nova demanda de serviços. Dentre esses serviços, destacam-se a oferta de Internet e VPN (*Virtual Private Network*) na rede da operadora apoiada pela implementação de uma arquitetura com QoS (*Quality of Service*) e satisfazendo os requisitos de TE (*Traffic Engineering*) sobre MPLS.

Esta dissertação apresenta a engenharia de tráfego aplicada a *backbone* IP MPLS de uma operadora de telecomunicações, com a aplicação prática do MPLS-TE (*Multi-Protocol Label Switching Traffic Engineering*), analisando e otimizando os recursos de transmissão com caminhos redundantes não utilizados pelo protocolo OSPF (*Open Shortest Path First*).

Palavra-Chave: Telecomunicações, Redes, IP, MPLS, OSPF.

Abstract

Telecommunication companies face a growing demand for new added value services on network infrastructures based on IP (Internet Protocol), such as voice and video. This demand brings a major challenge to provide a stable, scalable and optimized backbone, to satisfactorily meet the expectations of their customers. Telecom operatars are looking for a network with greater availability and lower cost. The MPLS (Multi-Protocol Label Switching) technology is indicated to provide the needed development, optimization and flexibility to current backbones, standing as viable technology to be employed in such networks.

The new paradigm of MPLS-based transport enables telecom operators to build on top of the IP, ATM (Asynchronous Transfer Mode) and Frame Relay infrastructures a set of creative and flexible solutions for such services. Among those services there is the provision of Internet and VPN (Virtual Private Network) in the carrier network supported by the implementation of an architecture for QoS (Quality of Service) and TE (Traffic Engineering) over MPLS.

This M.Sc. dissertation presents how traffic engineering can be applied to a MPLS IP backbone of a telecommunication carrier. The MPLS-TE (Multi-Protocol Label Switching Traffic Engineering) is used to analyse and optimise the transmission resources in a "real" network taking advantage of the redundant paths, which are not used by the OSPF (Open Shortest Path First) protocol.

Keywords: *Telecommunications, Network, IP, MPLS, OSPF.*

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

AF	*Assured Forwarding*
AS	*Automonous System*
ASIC	*Application Specific Integrated Circuit*
ATM	*Asynchrounous Transfer Mode*
AToM	*Any Transport over MPLS*
BGP	*Border Gateway Protocol*
BoS	*Bottom of Stack*
CBR	*Constant Bit Rate*
CBWFQ	*Class-Based Weighted Fair Queueing*
CE	*Customer Edge*
CEF	*Cisco Express Forwarding*
CIDR	*Classless interdomain routing*
CLNP	*Connectionless Network Protocol*
CQ	*Custom Queueing*
CSPF	*Constrained Shortest Path First*
CU	*Currently Unused*
DiffServ	*Differentiated Service*
DLCI	*Data Link Connection Identifier*
DNS	*Domain Name System*
DS	*Differentiated Services*
DSCP	*DiffServ Codepoint*
DWDM	*Dense Wavelength Division Multiplexing*
EBGP	*External BGP*
ECMP	*Equal Cost Multipath*
ECN	*Explicit Congestion Notification*
EF	*Expedited Forwarding*
EGP	*Exterior Gateway Protocol*
EIGRP	*Enhanced Interior Gateway Protocol*
ES	*End Systems*
EXP	*Experimental bits*
FEC	*Forwarding Equivalent Class*
FIB	*Forwarding Information Base*
FIFO	*First in First out*
FQ	*Fair Queueing*
FR	*Frame-Relay*
FRR	*Fast Rerouting*
GMPLS	*Generalized Multiprotocol Label Switching*
GRE	*Generic Routing Encapsulation*
IANA	*Internet Assigned Numbers Authority*
IBGP	*Internal BGP*
ICMP	*Internet Control Message Protocol*
IETF	*Internet Engineering Task Force*
IGP	*Interior Gateway Protocol*
IGRP	*Interior Gateway Routing Protocol*
IOS	*Internetwork Operation System*
IP	*Internet Protocol*

IS	*Intermediate Systems*
IS-IS	*Intermediate System to Intermediate System*
IS-IS-TE	*Intermediate System to Intermediate System Traffic Engineering*
ISO	*International Organization for Standardization*
ITU-T	*International Telecommunication Union - Telecommunication*
ISP	*Internet Service Provider*
ISUP	*ISDN User Part*
L2TP	*Layer 2 Tunneling Protocol*
LAN	*Local Area Network*
LDP	*Label Distribution Protocol*
LER	*Label Edge Router*
LFIB	*Label Forwarding Information Base*
LIB	*Label Information Base*
LLQ	*Low Latency Queueing*
LP	*Private Line*
LSA	*Link-State Advertisement*
LSP	*Label Switched Path*
LSR	*Label Switch Router*
L2TP	*Layer 2 Tunneling Protocol*
MAC	*Media Access Control*
MP-BGP	*Multi Protocol BGP*
MOS	*Mean Opinion Score*
MPLS	*Multi-Protocol Label Switching*
MPLS-TE	*Multi-Protocol Label Switching Traffic Engineering*
NSF	*National Science Foundation*
OSI	*Open Systems Interconnection*
OSPF	*Open Shortest Path First*
OSPF-TE	*Open Shortest Path First Traffic Engineering*
P	*Provider*
PBT	*Provider Backbone Transport*
PDH	*Plesiochronous Digital Hierarchy*
PDU	*Protocol Data Unit*
PE	*Provider Edge*
PHB	*Per-Hop Behavior*
PHP	*Penultimate Hop Popping*
POP	*Penultimate hop pop*
POS	*Packet Over Sonet*
PPP	*Point to Point*
PQ	*Priority Queueing*
PSN	*Packet Switched Network*
PVC	*Private Virtual Circuit*
QoS	*Quality of Service*
RD	*Router Distinguisher*
RED	*Random Early Detection*
RFC	*Request for Comments*
RIB	*Routing Information Base*
RIP	*Routing Information Protocol*
RIPng	*Routing Information Protocol next generation*
ROAD	*Routing and Addressing*
RR	*Router Reflection*

RSVP	*Resource Reservation Protocol*
RSVP-TE	*Resource Reservation Protocol –Traffic Engineering*
RT	*Router Targets*
SCTP	*Stream Control Transmission Protocol*
SDH	*Synchronous Digital Hierarchy*
SLA	*Service Level Agreement*
SNMP	*Simple Network Management Protocol*
SONET	*Synchronous optical networking*
STM	*Synchronous Transport Module*
SIGTRAN	*Signaling Transport*
SIP	*Session Initiation Protocol*
TCP	*Transmission Control Protocol*
TDM	*Time Division Multiplexing*
TE	*Traffic Engineering*
TLV	*Type-Length-Variable*
T – MPLS	*Transport MPLS*
TOS	*Type Of Service*
TTL	*Time to Live*
UDP	*User Datagram Protocol*
VCI	*Virtual Channel Identifier*
VoIP	*Voice over IP*
VPLS	*Virtual Private Lan Service*
VPN	*Virtual Private Network*
VPWS	*Virtual Private Wire Service*
VRF	*Virtual Routing and Forwarding*
WAN	*Wide Area Network*
WFQ	*Weighted Fair Queueing*
WRED	*Weighted Random Early Detection*

Capítulo 1 - Introdução

As redes de computadores mudaram a maneira pela qual são feitos os negócios e o modo como as pessoas vivem. Hoje, o acesso à Internet por meio de plataformas cabeadas já não atende ao mercado consumidor, pois há uma grande necessidade de mobilidade. Não há mais modelos de escritórios fixos, com ramais e estações de trabalho presas a mesas e cadeiras. A tecnologia da mobilidade está presente nas residências, empresas, escolas, universidades, aeroportos, etc. Para atender a essa grande demanda, novas tecnologias estão em constante desenvolvimento, visando fornecer um serviço com segurança e qualidade.

Estar conectado a uma rede não é mais suficiente, é necessário largura de banda, segurança e disponibilidade do serviço, já que os usuários necessitam de ligações de videoconferência, transferência de arquivos em tempo real, assistir a programas de televisão via Internet, provimento de escritórios virtuais, etc. Com essa demanda, os fornecedores de serviços de telecomunicações têm investido bastante em redes de multiserviços, com o objetivo de atender a serviços para clientes finais como voz, vídeo, acesso à Internet e ainda usar essa rede de dados para enviar serviços legados como sinalização de telecomunicações, bilhetes de tarifação, serviços de recargas para plataforma prepago entre outros. Com essa integração, as redes se tornam complexas em topologia e em gerenciamento, tendo cada vez mais a necessidade de utilizar técnicas de comutação associada a técnicas de roteamento, tirando dos roteadores do núcleo da rede tarefas complexas, e deixando apenas as tarefas mais rápidas associadas à comutação. Com isso, os roteadores de *backbones* não devem fazer o trabalho complexo das redes. Pode-se utilizar uma tecnologia mais apropriada para WAN (*Wide Area Network*), rede espacialmente distribuída que abrange uma grande área geográfica [1], assim como da adoção de uma metodologia de consenso em relação a aspectos de projeto por parte dos desenvolvedores.

A arquitetura de muitos *backbones* hoje, já está baseada em IP (*Internet Protocol*) [2] e muitas operadoras de telecomunicações possuem equipamentos de Multiplexação baseados em SONET/SDH (*Synchronous Opitcal Network/ Synchronous Digital Hierarchy*) [3], com tecnologia ATM *(Asynchronous Transfer Mode)* [1] e a tecnologia *Ethernet* [4], que está bastante presente nesse mercado. O MPLS (*Multi-*

Protocol Label Switching) [4] surge como uma tecnologia que associa as características rápidas de comutação da camada 2 e a inteligência da camada 3 do modelo OSI (*Open Systems Interconnection*) [1], além de oferecer maior possibilidade de gerenciamento e engenharia de tráfego, reduzindo o processamento necessário para realizar o roteamento de datagramas de rede.

A princípio, o MPLS foi desenvolvido para melhorar o desempenho das redes IP no transporte de pacotes, através de pequenos rótulos com tamanho fixo. Isso é feito combinando o processo de roteamento da camada 3 com a comutação da camada 2.

Na arquitetura IP sobre MPLS para encaminhar um pacote é necessário obter as informações no cabeçalho MPLS (32 *bits*), que é menos complexo que o cabeçalho IP (20 *bytes*), permitindo assim melhorar o desempenho dos equipamentos com menor poder de processamento e armazenamento.

Objetivando a implementação de novas tecnologias em *backbones* e sem perder o investimento feito em equipamentos, a tecnologia MPLS (*Multi-Protocol Label Switching*) possibilita a integração de equipamentos que não possuem o IP nativo com equipamentos que possuem IP nativo.

Com a possibilidade de utilização de uma única infraestrutura de dados e, partindo da premissa de uma rede segura e de menor custo, a utilização de um *backbone* IP MPLS (*Internet Protocol Multi-Protocol Label Switching*) provê aos assinantes os serviços de voz, vídeo e dados, sendo ainda possível a integração de serviços legados como, por exemplo, sinalização ISUP (*ISDN User Part*) sobre uma rede IP com o protocolo SIGTRAN (*Sinaling Transport*).

A principal contribuição desta dissertação é apresentar e analisar a engenharia de tráfego sendo aplicada em um *backbone* IP MPLS de uma operadora comercial, com a aplicação prática do MPLS-TE (*Multi-Protocol Label Switching Traffic Engineering)* [5], mostrando a análise e a otimização dos recursos de transmissão e utilização de caminhos redundantes não utilizados pelo protocolo OSPF (*Open Shortest Path First*) [6]. Foram feitos experimentos do uso da engenharia de tráfego aplicada ao MPLS, através de um cenário representativo de um *backbone* de operadora de telecomunicações criado para este trabalho.

1.1 Objetivos da Dissertação

Esta dissertação analisa um *backbone* IP e seus protocolos de roteamento internos e apresenta uma solução para otimização do uso de caminhos redundantes com a engenharia de tráfego aplicada pelo protocolo MPLS-TE.

É tratado especificamente o caso de um *backbone* que utiliza como IGP (*Interior Gateway Protocol)* [5] o protocolo de roteamento OSPF (*Open Shortest Path First*), onde é aplicada a solução do MPLS-TE com objetivo de otimizar os recursos de transmissão.

São exibidos gráficos de utilização de enlaces de um *backbone* representativo de operadora de telecomunicações comercial, onde vários caminhos são possíveis para chegar a um destino final.

Os objetivos desta dissertação são:

- estudar as redes IPs e protocolos de roteamento interno e externo;
- apresentar a tecnologia MPLS, suas funções e serviços;
- mostrar a vantagem do encaminhamento do tráfego em condições onde se tem circuitos baseados na tecnologia TDM (*Time Division Multiplexing*).
- montar uma topologia de rede que simule o mais próximo possível um ambiente real de uma de rede de dados de uma operadora de telecomunicações;
- fazer um comparativo entre uso do protocolo de roteamento interno e a aplicação da engenharia de tráfego;

1.2 Estrutura da Dissertação

Esta dissertação é dividida em oito capítulos, incluindo este de Introdução. A seguir, é exibido um breve resumo dos demais capítulos:

- **Capítulo 2 – A Internet e o protocolo IP**

O segundo capítulo apresenta o protocolo IP. Neste capítulo é feito um comparativo entre as versões do protocolo IPv4 e IPv6.

- **Capítulo 3 –Protocolos de Roteamento**

O terceiro capítulo apresenta os protocolos de roteamento IGP (*Interior Gateway Protocol*) [5] e EGP (*Exterior Gateway Protocol*) [5]. Neste capítulo é feita uma revisão dos conceitos de roteamento e comutação, base necessária para esta dissertação.

- **Capítulo 4 – A Tecnologia MPLS-IP**

No capítulo quatro são tratadas sucintamente as definições do protocolo MPLS e seu uso com o protocolo IP. Esse capítulo é responsável por apresentar o embasamento teórico das vantagens do protocolo MPLS. É descrito como é feito o encaminhamento de pacotes IP sobre MPLS e como são comutados os rótulos.

- **Capítulo 5 – Serviços do MPLS**

No quinto capítulo será abordado os serviços da tecnologia MPLS tais como QoS (*Quality of Service*) [8], VPN (*Virtual Private Network*) [9], *pseudowire* ou AToM (*Any Transport over MPLS)* [5] e MPLS-TE (*Multi-Protocol Label Switching Traffic Engineering*) [5].

- **Capítulo 6 – Engenharia de Tráfego com MPLS**

O sexto capítulo discorre sobre engenharia de tráfego e é dado completo foco na aplicação do MPLS-TE, onde são tratados os tipos de túneis como, por exemplo, por afinidade, por especificação de caminho (dinâmico ou explícito) e a prioridade dada a cada túnel. Nesse capítulo ainda é tratada a recuperação rápida, uma aplicação do MPLS-TE chamada FRR (*Fast Rerouting)* [9].

- **Capítulo 7 – Desempenho do MPLS-TE em um Sistema Comercial**

No sétimo capítulo, faz-se a aplicação prática no *backbone* da operadora em estudo, onde a solução está aplicada para estabelecimento de túneis explícitos. Nesse capítulo, é descrita a topologia do laboratório, utilizado para simulação da rede. São apresentados os detalhes dos roteadores utilizados nos experimentos [7].

- **Capítulo 8 – Conclusões e Trabalhos Futuros**

No oitavo capítulo, são apresentadas as conclusões desta dissertação sobre a eficácia da utilização do MPLS-TE em uma rede de telecomunicações. São também apresentadas novas linhas de pesquisas que podem dar continuidade a este trabalho.

- **Anexos**

No apêndice podem ser encontradas as listagem com as configurações dos roteadores de rede utilizados nos testes aqui descritos.

Capítulo 2 – A Internet e o protocolo IP

Com a disseminação da ARPANET (*Advanced Research Projects Agency Network*), a NSF *(National Science Foundation)* resolveu criar uma sucessora, que seria aberta a todos os grupos de pesquisa universitários e, com essa premissa, tomou a decisão de interligar seus centros de pesquisas, usando a mesma tecnologia de *hardware* da ARPANET, mas com a tecnologia de *software* diferente, pois seus equipamentos utilizavam TCP/IP (*Transmission Control Protocol Internet Protocol)*.

Em primeiro de janeiro de 1983, o TCP/IP se tornou o protocolo oficial na ARPANET, em seguida, essa foi interconectada à NSF, e a partir disso, o crescimento dessas redes se tornou exponencial [1]. Essas redes de redes [2] se tornou a Internet, não mais com fins acadêmicos, e sim com fins comercias, crescendo cada vez mais. Na tabela 1, pode-se visualizar o crescimento do número de usuários de Internet por região geográfica [10].

Tabela 1: Usuários de Internet por Região Geográfica [10].

Regiões	População (em 2010)	Usuários de Internet (2000)	Usuários de Internet (atualmente)	% por Região	% no mundo	Crescimento 2000-2010
África	1.013.779.050	4.514.400	110.931.700	10,9 %	5,6 %	2.357,3 %
Ásia	3.834.792.852	114.304.000	825.094.396	21,5 %	42,0 %	621,8 %
Europa	813.319.511	105.096.093	475.069.448	58,4 %	24,2 %	352,0 %
Oriente Médio	212.336.924	3.284.800	63.240.946	29,8 %	3,2 %	1.825,3 %
América Norte	344.124.450	108.096.800	266.224.500	77,4 %	13,5 %	146,3 %
América Latina /Caribe	592.556.972	18.068.919	204.689.836	34,5 %	10,4 %	1.032,8 %
Oceania	34.700.201	7.620.480	21.263.990	61,3 %	1,1 %	179,0 %
Total	6.845.609.960	360.985.492	1.966.514.816	28,7 %	100,00%	444,8 %

No modelo de Internet, o principal protocolo de rede é o IP *(Internet Protocol)* [2]. O protocolo IP foi criado com objetivo simples de tornar possível a comunicação entre máquinas independente do meio de transmissão, não possuindo mecanismos de notificação ou correção de erro. O protocolo IP não tem mecanismos que permitem realizar consultas de gerenciamento, é sem controle de fluxo, não orientado à conexão e não era prevista qualidade de serviço, ou seja, um protocolo desenvolvido para trabalhar em uma rede *best-effort* (este termo no IP significa que não provê mecanismos de controle de erros ou fluxo) [3]. Desde o seu desenvolvimento, esse protocolo sofreu várias modificações em seu projeto inicial com objetivo de se adequar às necessidades atuais de serviços de voz sobre uma rede de dados e serviço de vídeo sobre demanda.

Para isso, foi necessário a criação de campos no cabeçalho do IPv4 e o desenvolvimento de um protocolo que adicionasse o controle de fluxo como, por exemplo, o protocolo ICMP (*Internet Control Message Protocol)* [3].

Além desses pontos, existem dois problemas importantes que precisam ser resolvidos quando se conecta redes: heterogeneidade e escalabilidade [4]. O desafio da heterogeneidade é oferecer um serviço *host*-a-*host* útil e bastante previsível através desse emaranhado de redes diferentes. Para entender o problema de escalabilidade, é necessário considerar o crescimento da Internet, que praticamente dobrou de tamanho a cada ano durante os últimos 20 anos. Esse tipo de crescimento traz inúmeros desafios, um deles é o roteamento [4]. A Internet se tornou um bem pervasivo para a população, como também para as empresas e centros de pesquisas, hoje sem Internet pouco se faz.

Neste capítulo, é apresentado o protocolo IP (*Internet Protocol*) e como ele pode ser usado para a montagem de uma inter-rede escalável e heterogênea. São mostrados vários problemas que a Internet apresentou durante seu crescimento e as técnicas que foram empregadas para resolver esses problemas e o projeto que culminou na nova versão do protocolo IP, que é o IP versão 6 (IPv6), também conhecida como IP da próxima geração.

2.1 Protocolo IP

O *Internet Protocol* é a ferramenta chave usada para montar inter-redes escaláveis e heterogêneas [4]. O protocolo IP foi projetado para permitir a interconexão de redes de computadores que utilizam a tecnologia de comutação de pacotes. O ponto inicial, quando se deseja montar uma inter-rede, é definir um modelo de serviço, ou seja, os serviços *host*-a-*host* que devem ser oferecidos. O protocolo IP possui uma filosofia onde ele não exige praticamente nada de qualquer tecnologia utilizada para montar uma inter-rede. O IP é um protocolo sem conexão, onde o datagrama é um tipo de pacote que é enviado de uma maneira sem conexão por uma rede. O datagrama é conhecido como um pacote, que trafega de forma independente na rede. Nesse datagrama ou pacote são transportadas informações que permitem o encaminhamento até o seu destino e cada roteador pertencente a essa rede toma a decisão por qual interface deve enviar esse pacote através de tabelas de roteamento, que são atualizadas periodicamente. Nesse modelo de serviço, os datagramas são enviados baseados no

melhor esforço, muito embora o protocolo IP se esforce ao máximo para entregá-los, ele não faz garantias de entrega e será visto que foi criado um campo adicional no cabeçalho IP com objetivo de fazer marcação de pacotes.

O projeto inicial do protocolo IP foi criar um protocolo baseado em melhor esforço (*best-effort*) e, com isso, os roteadores seriam o mais simples possível. O IP pressupõe a falta de confiabilidade das camadas inferiores e faz o máximo para levar a transmissão até o seu destino, mas sem garantias [3]. O protocolo IP é limitado apenas à criação e encaminhamento de datagramas e, se a confiabilidade for importante, o IP deve ser utilizado com outros protocolos confiáveis como, por exemplo, os protocolos da camada de transporte e da camada de aplicação. Nessas camadas superiores (transporte ou aplicação) cabe a função de organização dos datagramas recebidos, bem como fazer um pedido de retransmissão de um datagrama perdido, caso seja necessário. A parte principal do modelo de serviço IP é o tipo de pacote que pode ser transportado. O datagama IP, como a maioria dos pacotes, consiste em um cabeçalho seguindo por uma série de *bytes* de dados [4].

2.1.1 Protocolo IPv4

O formato do datagrama IPv4 é exibido na Figura 1. Ele tem um comprimento variável e é dividido em duas partes: cabeçalho e dados. O cabeçalho tem comprimento de 20 a 60 *bytes* e contém informações essenciais para o roteamento e a entrega [3]. O cabeçalho contém as informações administrativas do datagrama, já o campo de dados contém as informações das aplicações. Os principais campos desse datagrama são os seguintes [3]:

Versão (VER – 4 *bits*): Este campo de 4 *bits* define a versão do protocolo IP. Atualmente, a versão é 4. Entretanto, a versão 6 (IPng) poderá substituir completamente a versão 4 no futuro. Observe-se que a colocação deste campo diretamente no início do datagrama facilita para que tudo o mais no formato do pacote seja redefinido em versões posteriores. Este campo informa ao *software* do IPv4, que roda na máquina em processamento, que o datagrama tem formato da versão 4. Todos os campos devem ser interpretados conforme especificado na quarta versão do protocolo. Se a máquina estiver usando alguma outra versão do IP, o datagrama é descartado em vez de ser interpretado incorretamente.

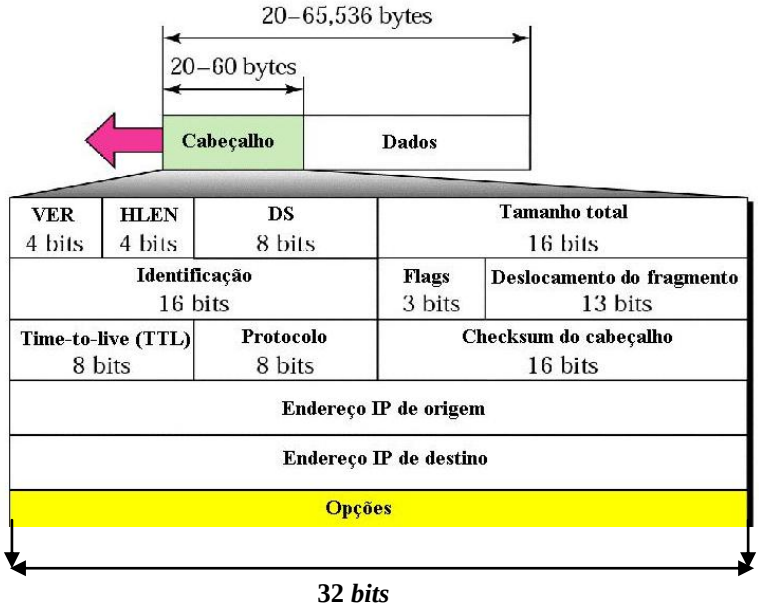


Figura 1: Formato do Datagrama no IPv4 [3].

Tamanho do Cabeçalho (HLEN – 4 *bits*): Este campo define o comprimento total do cabeçalho do datagrama em palavras de 32 bits e é necessário porque o comprimento do cabeçalho é variável. Quando não existem opções, o que quase sempre acontece [4], o comprimento do cabeçalho é de 20 *bytes* e o valor desse campo é 5 (5 x 4 = 20 *bytes*) de extensão. Quando o campo de opções estiver em seu tamanho máximo, seu valor é 15 (15 x 4 = 60 *bytes*) de extensão [3].

Tipo de Serviço (DS - 8 *bits*): O campo ToS (*Type of Service*) teve diversas definições diferentes no decorrer dos anos, mas a função básica é permitir que os pacotes sejam tratados de modo diferente, com base nas necessidades da aplicação [4]. Este campo é utilizado pelos roteadores para determinar como o datagrama deve ser tratado, podendo diferenciar os diferentes tipos de datagramas IP. Por exemplo, pode ser útil distinguir os datagramas de tempo real (Ex.: Telefonia IP) dos tráfego que não são de tempo real (Ex.: FTP). O IETF (*Internet Engiennering Task Force*) mudou a interpretação e o nome deste campo de 8 *bits*. Este campo, anteriormente denominado tipo de serviço, agora se chama serviços diferenciados [3] e na figura 2 é possível visualizar as duas interpretações.

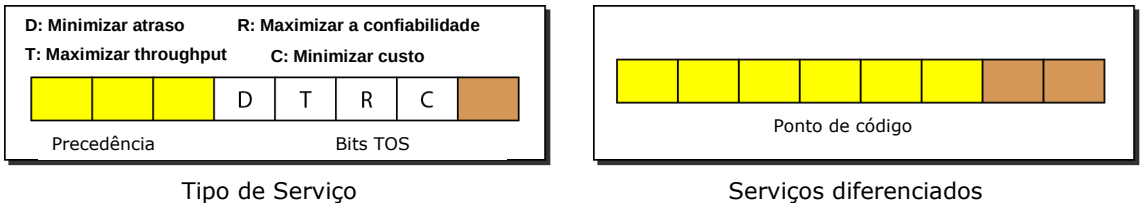


Figura 2: Tipo de serviço ou serviço diferenciados [3].

No tipo de serviço, os três primeiros *bits* são denominados *bits* de precedência. Os 4 *bits* seguintes são chamados *bits* ToS (*Type of Service*) e o último *bit* não é usado. A precedência é um subconjunto de três *bits* no intervalo que vai de 0 (000 em binário) a 7 (111 em binário). A precedência define a prioridade do datagrama em questões como congestionamento. Se um roteador estiver congestionando e precisar descartar alguns pacotes, aqueles com menor precedência serão descartados primeiro. O ToS é um subcampo de 4 *bits*, onde cada *bit* tem um significado especial. Embora um *bit* possa ser 0 ou 1, um e somente um dos *bits* do subcampo pode ter valor 1 em cada datagrama. Os padrões de *bits* e suas interpretações são apresentados na tabela 2.

Tabela 2: Tipos de Serviço [3].

Bits ToS	Descrição
0000	Normal (padrão)
0001	Minimizar custo
0010	Maximizar confiabilidade
0100	Minimizar *throughput*
1000	Minimizar atraso

Na interpretação com os serviços diferenciados os seis primeiros *bits* formam o subcampo ponto de código e os últimos 2 *bits* não são usados.

Esse campo será mais detalhado no capítulo 5, sessão 5.3.1.2, desta dissertação.

Tamanho total do datagrama (16 *bits*): Trata-se de um campo que define o comprimento total do datagrama IPv4, incluindo o cabeçalho. O tamanho máximo de um datagrama IP é de 65.535 *bytes*. Porém, a rede física em cima da qual o IP está sendo executado não pode admitir pacotes tão grandes. Por esse motivo, o IP admite um processo de fragmentação e remontagem. Para descobrir o comprimento dos dados provenientes da camada superior, subtrai-se o comprimento do cabeçalho do comprimento total. O comprimento do cabeçalho pode ser encontrado multiplicando-se o valor do campo tamanho do cabeçalho por 4.

Alguns padrões físicos não são capazes de encapsular um datagrama de 65.535 *bytes* em seus quadros. O datagrama tem que ser fragmentado para conseguir ser transmitido por essas redes, com exemplo o protocolo *Ethernet* apresenta uma restrição mínima e máxima no tamanho dos dados que podem ser encapsulados em um quadro

(46 a 1.500 *bytes*). Se o tamanho de um datagrama IPv4 for menor que 46 *bytes*, serão acrescidos alguns *bits* de preenchimento para atender a essa exigência.

Identificação (16 *bits*): Utilizado para identificação do datagrama IP, no processo de fragmentação do pacote, para que o mesmo seja remontado na mesma ordem em que foi fragmentado. Quando um datagrama é maior do que o MTU (*Maximum Transfer Unit*) de uma determinada tecnologia, ele precisa ser dividido em fragmentos para que possa ser transmitido na rede. Assim, o campo identificador é utilizado para que seja possível saber a qual datagrama cada fragmento pertence.

Um dos problemas de oferecer um modelo de serviço uniforme ponta-a-ponta por uma coleção heterogênea de redes é que cada tecnologia de rede costuma ter sua própria idéia quanto ao tamanho que um pacote pode ter.

Um datagrama pode trafegar por várias redes diferentes. Cada roteador desencapsula o datagrama IP, a partir do quadro que ele recebe, o processa e então o encapsula em outro quadro. O formato e o tamanho do quadro recebido dependem do protocolo usado pela camada física por meio do qual o quadro acaba de passar. Se, por exemplo, um roteador interliga uma LAN (*Local Area Network)* a uma WAN (*Wide Area Network*), ele recebe um quadro no formato da LAN e transmite um quadro no formato da WAN.

A idéia central é que cada tipo de rede tenha uma MTU (*Maximum Transfer Unit*), que é o maior datagrama IP que ele pode transportar em um quadro.

Para tornar o protocolo IP independente da rede física, os projetistas decidiram fazer o comprimento máximo de um datragrama IP igual a 65.535 *bytes*. Isso torna a transmissão mais eficiente quando se utiliza um protocolo com MTU desse tamanho. Entretanto, para outras redes físicas, é necessário dividir o datagrama para tornar possível sua passagem por essas redes. Isso é denominado fragmentação. O interessante é que a nova versão do protocolo IP, o IPv6, não permite fragmentação em roteadores [2].

Flags (3 *bits*): Trata-se de um campo de 3 *bits*. O primeiro é reservado. O segundo é denominado *bit* DF (*Don't Fragment*), não fragmentado e é utilizado para indicar aos roteadores que não fragmentem o pacote, porque o destino não os saberá reconstruir. Se seu valor for 1, a máquina não poderá fragmentar o datagrama. Se não puder passar o datagrama por meio de qualquer rede física disponível, ele descarta o datagrama e envia uma mensagem de erro ICMP (*Internet Control Message Protocol)* ao *host* de origem. Se seu valor for 0, o datagrama pode ser fragmentado se necessário.

O terceiro *bit* é o chamado de *bit* MF (*More Fragments*) mais fragmentos. Se seu valor for 1, significa que esse datagrama não é o último fragmento; existem mais fragmentos após este. Se seu valor for 0, significa que esse é o último ou único fragmento [3]. A figura 3, mostra os *flags* usados na fragmentação.

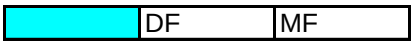


Figura 3: *Flags* usados na fragmentação [3].

Deslocamento do Fragmento (*Fragmentation Offset* - 13 *bits*): Esse campo de 13 bits mostra a posição relativa desse fragmento em relação ao datagrama inteiro. É o *offset* dos dados no datagrama original medido em unidades de 8 *bytes*. A figura 4 mostra um datagrama cujo tamanho dos dados é igual a 4000 *bytes*, fragmentados em três partes.

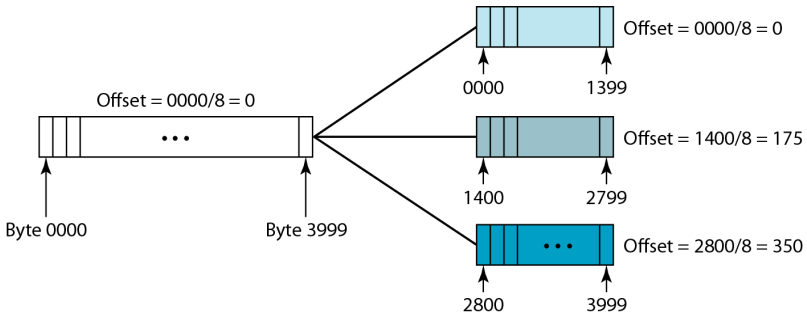


Figura 4: Exemplo de fragmentação [3].

Tempo de Vida (*Time-to-live* - 8 *bits*): É utilizado para garantir que datagramas não fiquem circulando para sempre na rede. Ao receber um datagrama, todo roteador deve ler esse campo, se seu valor for maior que zero, ele deverá decrementá-lo em uma unidade e, se seu valor for igual a zero esse datagrama deverá ser descartado, evitando assim um laço de roteamento de longa duração. Um datagrama tem um tempo de vida útil limitado em sua transmissão por uma rede de computadores. Esse campo foi projetado originalmente para armazenar um registro de horas, que era reduzido pelos roteadores visitados. O datagrama era descartado quando o valor se tornava zero. Entretanto, para implementar esse método, todas as máquinas devem ter relógios sincronizados e devem saber quanto tempo leva para um datagrama ir de uma máquina a outra.

Protocolo (8 *bits*): Esse campo define o protocolo de nível superior que está utilizando os serviços da camada de rede. Um datagrama IP pode encapsular dados de vários protocolos superiores como TCP (*Transmission Control Protocol)*, UDP (*User*

Datagram Protocol), ICMP (*Internet Control Message Protocol*) e OSPF. Esse campo especifica o protocolo de destino final para o datragrama IP que será entregue. A figura 5 mostra os detalhes do campo.

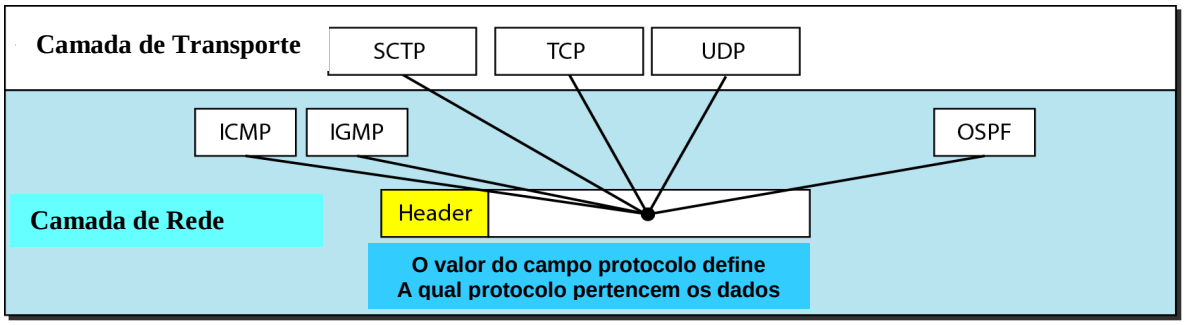


Figura 5: Campo de protocolo e dados encapsulados [3].

Checksum (16 *bits*): A paridade (*checksum*) no datagrama IP cobre apenas o cabeçalho, e não todos os dados. Há duas boas razões para isso. Em primeiro lugar, todos os protocolos de nível superior que encapsulam dados em um datagrama IP têm um campo de paridade que cobre o pacote inteiro. Portanto, a paridade para um datagrama IP não precisa verificar os dados encapsulados. Em segundo, o cabeçalho de um datagrama IP muda a cada roteador visitado, mas não os dados. Portanto, a paridade inclui apenas a parte alterada. Se os dados forem inclusos, cada roteador terá que recalcular a paridade para o pacote inteiro, significando um aumento no tempo de processamento.

Endereços IP de origem e destino (32 *bits*): Representam os endereços IP do *host* que envia o datagrama (fonte) e do *host* que receberá o datagrama (destino).

Opções (0 a 320 *bits*): Permite que o cabeçalho IP seja ampliado. Existem opções para segurança, armazenamento de rota, roteamento mandatório, *timestamp*, etc. Uma vez que alguns datagramas podem requerer processamento de opções e outros não, a quantidade de tempo necessária para processar um datagrama IP em um roteador pode variar bastante. Por essas razões, este campo foi descartado no cabeçalho da versão IPv6.

A entidade que controla os números IP é o IANA (*Internet Assigned Numbers Authority*), que hoje é parte da ICANN (*Internet Corporation for Assigned Names and Numbers*). A autoridade sobre os números IP é delegada regionalmente para outras entidades. Na América Latina e Caribe, a entidade responsável é o LACNIC, e no Brasil, o responsável é o NIC.br (Núcleo de Informação e Coordenação no Brasil) [12].

As especificações do IPv4 reservam 32 *bits*, ou seja, o espaço corresponde a 2^{32} (4.294.967.296) para endereçamento, possibilitando gerar mais de 4 bilhões de endereços distintos. Na Internet, os dispositivos devem ser identificados por números de 32 *bits* que definem de forma única e universal a conexão de um dispositivo. Os endereços IPs devem ser únicos, não podendo haver dois ou mais equipamentos utilizando o mesmo endereço IP. Assim, um endereço IP é um número que identifica unicamente no mundo cada conexão de dispositivo à Internet.

Basicamente, um endereço IP consiste em duas partes: a primeira conhecida como *netid*, que representa o endereço da rede à qual pertence o dispositivo, e a segunda, conhecida como *hostid*, que representa a identificação do dispositivo nessa rede. No esquema original de endereçamento IP, existem cinco classes de endereços, conforme a figura 6. Nessa tabela, pode-se verificar quais *bits* representam o *hostid* e quais correspondem ao *netid*, bastando analisar os 3 primeiros *bits*, pois os mesmos definem a classe a qual o endereço IP pertence.

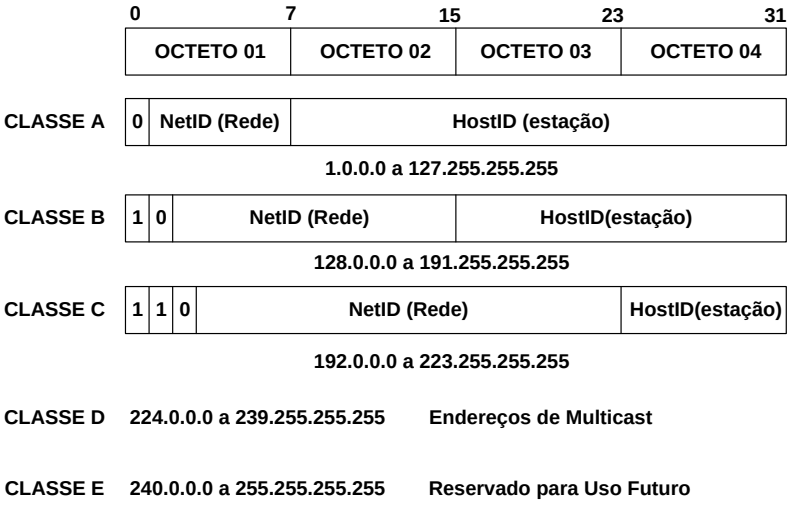


Figura 6: Classess dos endereços IPs.

O endereçamento IPv4, em seu início, usava o conceito de classes. Essa arquitetura é chamada: endereçamento por classe. Esse método, embora esteja se tornando obsoleto, será explicado rapidamente para mostrar o conceito por trás do endereçamento sem classe.

Na classe “A”, o primeiro *bit* (mais significativo) é zero, e os próximos 7 *bits* representam a identificação da rede. Os 24 *bits* restantes representam o *host*, ou o dispositivo na rede. Essa classe é recomendável para redes com um número muito grande de dispositivos, pois pode representar mais de 16 milhões de endereços IPs.

Na classe “B”, o primeiro *bit* é 1 e o segundo é 0. Os próximos 14 *bits* representam a identificação da rede e os 16 *bits* restantes representam o dispositivo na rede. O número de dispositivos que podem ser endereçados nessa classe é relativamente grande, mais de 65 mil.

Na classe “C”, os dois primeiros *bits* têm valor 1, e o terceiro *bit* é 0. A identificação da rede é representada pelos próximos 21 *bits*, o que possibilita a representação de um grande número de redes. Para representação de dispositivos são utilizados os últimos 8 *bits*, o que limita em 254 o número máximo de endereços IPs.

Na classe “D”, os três primeiros *bits* assumem valor 1 e o quarto vale 0. Esta classe é utilizada para identificar grupos de endereços de *multicast*. Diferentemente das classes anteriores, essa classe não possui uma divisão entre *netid* e *hostid*. Os 4 primeiros *bits* são fixos e os 28 restantes representam um grupo.

Na classe “E”, os quatro primeiros *bits* valem 1. Esta é uma classe de endereços reservada para uso futuro.

Um problema com o endereçamento com classes é que cada classe é dividida em um número fixo de blocos tendo um tamanho fixo. É possível notar uma falha nesse esquema, um bloco em um endereçamento classe “A” é muito grande para praticamente qualquer organização. No início essa classe foi atribuída integralmente às grandes instituições como IBM, AT&T, Xerox, HP, Apple, MIT, Ford, Departamento de Defesa Americano, entre muitas outras, e foram disponibilizados, para cada uma, 16.777.216 milhões de endereços [11]. Isso significa que a maioria dos endereços na classe “A” era desperdiçada e não era usada. Um bloco classe “B” também é muito grande, provavelmente muito grande para muitas organizações que recebiam um bloco classe “B”. Um bloco classe “C”, certamente, era muito pequeno para muitas organizações. Os endereços classe “D” foram projetados para *multicast*. Cada endereçamento nessa classe é utilizado para definir um grupo de hosts na Internet. Os provedores de Internet previam erroneamente a necessidade 268.435.456 grupos [3]. Nesse caso, isso jamais aconteceu e muitos endereços também foram desperdiçados. E finalmente, os endereços da classe “E” eram reservados para o futuro, mas apenas um pequeno número foi utilizado, resultando, mais uma vez, em desperdício de endereços.

Devido ao ritmo de crescimento da Internet e a política de distribuição de endereços, em maio de 1992, 38% das faixas de endereços da classe “A”, 43% da classe “B” e 2% da classe “C”, já estavam alocados. Nessa época, a rede possuía 1.136.000 *hosts* conectados [11].

Esse problema motivou a IETF (*Internet Engineering Task Force*) a criar soluções para o esgotamento dos endereços IP e o aumento da tabela de roteamento. Em novembro de 1991, é formado o grupo ROAD (*Routing and Addressing*), que apresenta como solução a utilização do CIDR (*Classless Interdomain Routing*) [11]. O roteamento interdomínio sem classes foi definido na RFC 4632 (*Request for Comments 4632*). A sua idéia básica é o fim do uso de classe de endereços, permitindo a alocação de blocos de tamanho apropriado a real necessidade de cada rede. O CIDR é uma técnica que resolve os dois aspectos de expansão na Internet: crescimento das tabelas de roteamento do *backbone*, à medida que mais números de rede precisam ser armazenadas nelas, e potencial para o espaço de endereços IP de 32 *bits*. O CIDR tenta equilibrar o desejo de reduzir o número de rotas que um roteador precisa conhecer, com a necessidade de oferecer endereços de modo eficiente [4].

Outra solução para o esgotamento de endereços IPv4 foi o NAT (*Network Addres Translation*), definido na RFC 3022. Esse protocolo permite ao usuário ter internamente um grande conjunto de endereços e externamente um outro endereço, ou então um pequeno conjunto de endereços, ou seja, a idéia básica é permitir que um único endereço IPv4, ou um pequeno número deles, possa ser utilizado por vários *hosts* para trafegar na Internet. Para separar os endereços usados internamente na residência, ou na empresa, daqueles utilizados para a Internet, os provedores reservam três conjuntos de endereços, denominados privados conforme RFC 1918 e apresentados na tabela 3 abaixo:

Tabela 3: Endereços IPv4 para redes privadas.

Intervalo	Total
10.0.0.0 a 10.255.255.255	2^{24} (16.777.216 *hosts*)
172.16.0.0 a 172.31.255.255	2^{20} (1.048.576 *hosts*)
192.168.0.0 a 192.168.255.255	2^{16} (65.536 *hosts*)

Essa solução se mostrou eficiente em relação à economia de endereços IPv4, pois apresenta alguns aspectos positivos como: facilitar a numeração interna das redes, ocultar a topologia das redes e só permitir a entrada de pacotes gerados em resposta a um pedido da rede. No entanto, o uso do NAT apresenta inconvenientes que não compensam as vantagens como, por exemplo, a quebra do modelo fim-a-fim da Internet, não permitindo a conexão direta entre dois *hosts*, o que dificulta o

funcionamento de aplicações como VoIP (*Voice over IP*) e VPN (*Virtual Private Network*). Além disso, exige que os equipamentos responsáveis pelo serviço do NAT possuam grande poder de processamento.

As soluções apresentadas diminuíram a demanda por IPv4, porém elas não solucionaram os problemas do crescimento da Internet. A adoção dessas técnicas reduziu em apenas 14% a quantidade de blocos solicitados a IANA (*Internet Assigned Numbers Authority*) e a curva de crescimento da Internet continuava apresentado um aumento exponencial. Na figura 7 são apresentadas as soluções paliativas do NAT e do CIDR na distribuição de blocos de endereços pela IANA [11]:

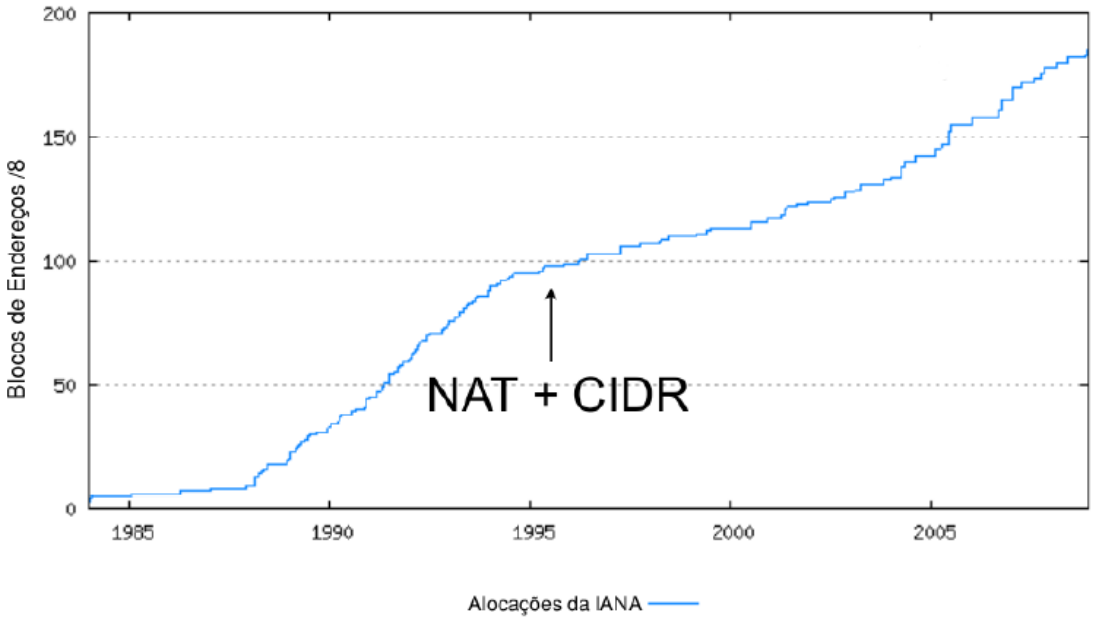


Figura 7: Curva de crescimento de distribuição de blocos de endereços com aplicação de NAT e CIDR [11].

Hoje, porém, com o crescimento acelerado do volume de máquinas e redes na Internet, assim como o surgimento de novas tecnologias e dispositivos de redes (*Smartphones*, computadores portáteis, etc.), segundo dados da ABI Reaserch, a quantidade de equipamentos móveis capazes de acessarem a Internet deve chegar a 2,25 bilhões [11]. Isso indica um rápido esgotamento dos endereços IPs disponíveis e a solução em definitivo para escassez do IPv4 é o IPv6.

2.1.2 Protocolo IPv6

O novo protocolo IPv6 deverá possibilitar a resolução dos problemas atuais de esgotamento do IPv4 e prover as funcionalidades necessárias para acompanhar as novas

tecnologias de redes que surgirem. Estima-se que, no Brasil, o esgotamento IPv4 ocorra entre 2012 e 2014 [12].

No projeto original do protocolo IPv4 não foram previstos aspectos de segurança, priorização de pacote, aumento da tabela de roteamento e esgotamento dos endereços IP. Em 1993 com objetivo de superar essas e outras deficiências, a IETF (*Internet Engineering Task Force*) formalizou, através da RFC 1560, as pesquisas a respeito da nova versão do protocolo IP. As principais questões que deveriam ser abordadas seriam: escalabilidade, segurança, configuração, administração de redes, suporte a QoS, mobilidade, políticas de roteamento e transição. O IPv6, também conhecido como IPng (*Intenet Protocol Next Generation*), foi proposto e agora é um padrão. No IPv6, o IP foi extensivamente modificado para acomodar o crescimento não previsto da Internet. O formato e o comprimento do endereço IP foram modificados juntamente com o formato do datagrama. Os protocolos relacionados, como ICMP, também foram modificados. Outros protocolos da camada de rede, como ARP (*Adress Resolution Protocol*), RARP (*Reverse Adress Resolution Protocol*) e IGMP (*Internet Group Management Protocol*), foram eliminados ou inclusos no protocolo ICMPv6. Os protocolos de roteamento, como o RIP (*Routing Information Protocol*) e o OSPF, também foram ligeiramente modificados para acomodar essas alterações. As especificações do IPv6 foram apresentadas na RFC 1883 de dezembro de 1995, no entanto, em dezembro de 1998, esta RFC foi substituída pela RFC 2460 [11].

É feito um breve comentário sobre um pacote IPv6 que é mostrado na figura 8. Foram feitas algumas mudanças no formato do cabeçalho-base do IPv6, de modo a torná-lo mais simples, com apenas 8 campos. Cada pacote é composto por um cabeçalho-base obrigatório seguido do *payload*. O *payload* consiste de duas partes: cabeçalhos de extensão opcionais e dados da camada superior. O cabeçalho-base possui tamanho fixo de 40 *bytes*, ao passo que os cabeçalhos de extensão e os dados da camada superior contêm até 65.535 *bytes* de informação. Uma das mudanças foi a remoção de seis campos do cabeçalho do IPv4, já que as funções não são mais necessárias ou são implementadas pelos cabeçalho de extensão. Com essas modificações o protocolo se tornou mais simples, mais flexível e mais eficiente, pois foi minimizado o *overhead* nos cabeçalhos reduzindo o processamento dos pacotes.

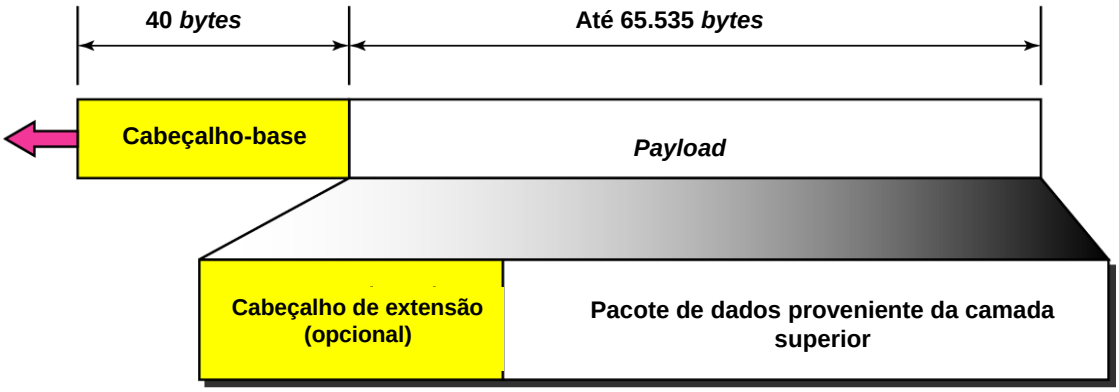


Figura 8**:** Formato do Datagrama no IPv6: cabeçalho e *payload* [3].

O cabeçalho-base é mostrado na figura 9. Comparando o cabeçalho do IPv4 com o cabeçalho IPv6 é possível verificar que seis campos do cabeçalho do IPv4 foram removidos e quatro campos tiveram seus nomes alterados e seus posicionamentos modificados.

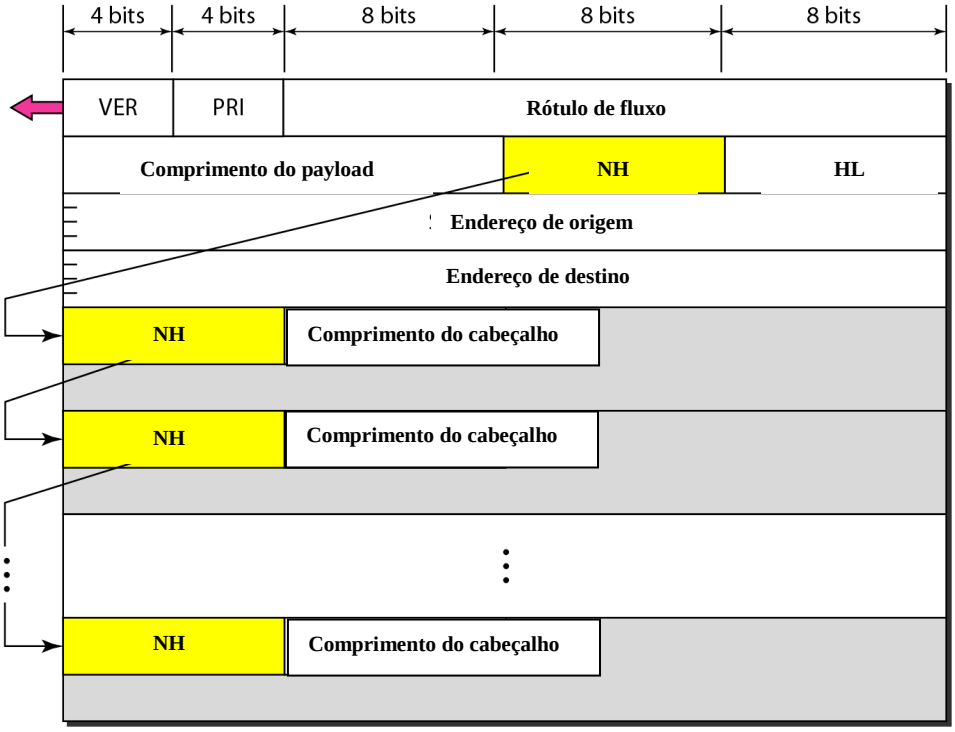


Figura 9: Formato de um Datagrama IPv6 [3].

Estes campos são os seguintes [3]:

Versão (VER – 4 *bits*): Um campo de 4 *bits* que define o número da verão do IP. Para o IPv6, o valor é 6.

Prioridade (PRI – 4 *bits*): O campo de 4 *bits* define a prioridade do pacote em situações de congestionamento de tráfego. O campo de prioridade em um pacote IPv6

define a prioridade de cada pacote em relação a outros pacotes de uma mesma origem. O IPv6 divide o tráfego em duas amplas categorias: controlado por congestionamento e não controlado por congestionamento.

Rótulo de Fluxo (*Flow Label* - 3 *bytes*): É um campo de 24 *bits* que se destina a oferecer tratamento especial para um determinado fluxo de dados. Uma sequência de pacotes, enviada de determinada origem a certo destino, que precisa de tratamento especial por parte dos roteadores, é denominado de fluxo de pacotes. A combinação do endereço de origem com o valor do rótulo de fluxo define de forma exclusiva um fluxo de pacotes. Esse campo pode ser utilizado na transmissão de áudio e vídeo em tempo real e em conjunto com os outros protocolos que fazem reserva de recurso como, por exemplo, RSVP (*Resource Reservation Protocol).*

Comprimento do *payload* (*Payload Length* – 2 *bytes)*: Este campo define o comprimento do datragrama IP excluindo o cabeçalho-base.

Próximo Cabeçalho (NH - *Next Header* – 8 *bits*): É o campo que aponta para o próximo cabeçalho após o cabeçalho-base em um datagrama IP. O *next header* é um dos cabeçalhos de extensão opcional usado pelo IP ou cabeçalho de um pacote encapsulado como o UDP ou TCP. Cada cabeçalho de extensão também contém esse campo. A tabela 4 é apresentada para o cabeçalho *Next Header*. Esse campo no IPv4 é denominado *protocol*.

Tabela 4: Códigos Próximo Cabeçalho no IPv6 [3].

Código	Próximo Cabeçalho
0	Opção Nó a Nó
2	ICMP
6	TCP
17	UDP
43	Roteamento de origem
44	Fragmentação
50	*Encrypted Security Playload*
51	Autenticação
59	Nulo (Próximo Cabeçalho inexistente)
60	Opção de destino

Limite de Saltos (HL - *Hop limit* – 8 *bits*): Esse campo atende ao mesmo objetivo de campo TTL no IPv4.

Endereço de origem (*Source address* – 128 *bits*): É um endereço Internet de 16 *bytes* que identifica a fonte de origem do datagrama.

Endereço de destino (*Destination address* – 128 *bits*): É um endereço de 16 *bytes* que normalmente identifica o endereço de destino final do datagrama. Entretanto, se for usado o roteamento de origem, esse campo contém o endereço do próximo roteador.

O IPv6, assim como IPv4, teve uma grande aceitação na área acadêmica, mas para que seu uso se torne exponencial é necessário que os provedores de acesso a Internet sejam capazes de transmitir tráfego IPv6 de forma nativa em seus *backbones*. Em contrapartida, sabe-se que sua implementação envolve custos em equipamentos de rede, além de treinamento para toda a equipe.

Estudos realizados mediram a quantidade de informação que trafega na Internet sobre o protocolo IPv6. Essas análises foram realizadas baseadas na quantidade de páginas na Internet que usam o IPv6 e na consulta a servidores DNS (*Domain Name System*), tentado medir a evolução do uso do IPv6 na Internet [14].

O Google tem realizado [13] uma avaliação do estado atual do uso do IPv6 por pessoas comuns, coletando informações fornecidas pelos navegadores de uma parcela de usuários de seus serviços. Com isso, foi possível determinar que aproximadamente 0,2% de seus clientes tem IPv6 ativado, e a quantidade de acesso que utilizam o IPv6 subiu de 0,189% em agosto de 2008, para 0,261% em janeiro de 2009.

Blocos de endereços IPv6 vêm sendo alocados pelos RIRs (*Regional Internet Registries*) há aproximadamente dez anos. No entanto, o fato dos RIRs alocarem endereços aos Registros Nacionais, ou aos ISPs, não significa que esses endereços estejam sendo utilizados. Fazendo o cruzamento de dados sobre a quantidade de blocos /32 IPv6 já alocados com números de rotas anunciadas na tabela de roteamento, nota-se que apenas 3% desses recursos estão sendo efetivamente utilizados, isto é, dos 73.000 blocos já alocados, apenas pouco mais 2.500 estão presentes na tabela global de roteamento [11].

2.1.3 Protocolo IPv4 x IPv6

O IPv6 apresenta algumas vantagens em relação ao IPv4 que podem ser resumidas a seguir:

- Maior espaço de endereçamento: um endereço IPv6 tem 128 *bits* de comprimento, permitindo níveis mais específicos de agregação de endereços, identificação de uma quantidade muito maior de dispositivos na rede e implementação de mecanismos de autoconfiguração. A escalabilidade de roteamento *multicast* também foi melhorada através da adição do campo "escopo" no endereço *multicast*. E um novo tipo de endereço, *anycast*, foi definido [11];
- Formato mais adequado do cabeçalho. O IPv6 usa um novo formato, no qual as opções são separadas do cabeçalho-base e inseridas, quando necessário, entre o cabeçalho-base e os dados da camada superior. Isso simplifica e acelera o processo de roteamento, pois grande parte das opções não precisa ser processada pelos roteadores [3];
- Espaço para expansão. O IPv6 foi desenvolvido para permitir a extensão do protocolo, caso seja preciso suportar novas tecnologias ou aplicações [3];
- Suporte para alocação de recursos. No IPv6 foi adicionado um novo recurso que permite identificação de pacotes que pertençam a determinados tráfegos de fluxos, o campo tipo de serviço do IPv4 foi eliminado, mas um mecanismo, denominado *flow label* – rótulo de fluxo foi acrescentado para permitir que a origem solicite tratamento especial de pacote. Esse mecanismo pode ser usado para suportar tráfego como áudio e vídeo em tempo real [3];
- Melhor suporte à segurança. O IPv6 oferece confidencialidade e integridade para os pacotes, foram especificados cabeçalhos de extensão capazes de fornecer mecanismos de autenticação e garantir a integridade e a confidencialidade dos dados transmitidos.

A tabela 5 apresenta de forma resumida a comparação do cabeçalho do IPv4 em relação à nova versão do protocolo IPv6.

Tabela 5: Comparação entre os cabeçalhos de IPv4 e IPv6.

1 – O campo de comprimento do cabeçalho é eliminado no IPv6, pois o comprimento do cabeçalho é fixo nessa versão.
2 – O campo de tipo de serviço é eliminado no IPv6. Os campos de prioridade e de rótulo de fluxo, juntos, assumem a função do campo de tipo de serviço.
3 – O campo comprimento total é eliminado no IPv6 e substituído pelo campo de comprimento do *payload*.
4 – Os campos de identificação, *flag* e *offset* são eliminados do cabeçalho-base no IPv6. Eles são inclusos no cabeçalho de extensão.
5 – O campo TTL chama-se limite de saltos IPv6.
6 – O Campo de protocolo é substituído pelo campo *next header*.
7 – A paridade (*checksum*) do cabeçalho foi eliminada, pois a paridade já é calculada pelos protocolos de camada superior; portanto, ele não é necessário neste nível.
8 – Os campos de opções do IPv4 são implementados como cabeçalho de extensão no IPv6.

2.2 Considerações Finais

Neste capítulo foi realizada uma revisão geral dos protocolos IPv4 e IPv6, para apresentar um comparativo entre as duas versões. Essa revisão do protocolo IPv4 foi necessária, pois ele é utilizado nesta dissertação para identificar logicamente as interfaces dos roteadores do *backbone* de teste.

Depois que foram tratados os aspectos do protocolo IP, serão introduzidos no capítulo 3 os princípios de roteamento e comutação. Serão explorados os problemas de expansão dos protocolos de roteamento.

Capítulo 3 - Protocolos de Roteamento

O roteamento é um processo de escolha do caminho a ser seguido pelos dados a serem transmitidos. Ele pode levar vários aspectos em consideração desde a velocidade dos enlaces, ao número de saltos (*hops*) envolvidos, passando pelo custo de transmissão e confiabilidade dos canais. No roteamento há a retransmissão, no qual os pacotes são encaminhados de um canal para outro. O roteamento diz respeito à maneira pela qual as tabelas de roteamento são criadas para auxiliar no encaminhamento dos pacotes. Nas redes de datagrama, incluindo as redes IP, o roteamento é tratado pacote a pacote [4]. De uma forma simples, um roteador precisa apenas ser capaz de examinar o endereço de destino do pacote e depois determinar qual rota é a melhor opção para encaminhar o pacote até seu destino. O protocolo IP com sua simplicidade e flexibilidade o torna um grande sucesso na função do roteamento, sendo esse protocolo responsável pela entrega das informações geradas pelas aplicações aos seus destinos de forma correta e eficiente.

Roteamento IP é um termo utilizado para descrever as ações efetuadas pela rede para enviar um pacote de um dispositivo a outro em uma rede diferente, ou seja, os endereços IPs de origem e destino devem pertencer à endereços de redes diferentes. Cada dispositivo da rede tem um endereço lógico para que ele possa ser alcançado individualmente. Em alguns casos, os dispositivos também podem ser alcançados como parte de um grupo maior de dispositivos como, por exemplo, o *multicast.*

Se o *host* não estiver conectado a uma mesma rede física do *host* de destino, então ele precisa enviar o pacote a um *gateway*. O elemento *gateway*, que é a antiga nomenclatura do roteador [6], é o principal componente das redes IPs, e a sua principal tarefa é o encaminhamento de pacotes. Os roteadores podem efetuar o roteamento direto usando tabelas de pesquisas como aquela produzida pelo ARP para mapear os endereços IP aos endereços MAC (*Media Access Control*) e assim empacotar os datagramas IP em quadros endereçados aos seus destinos finais. Esse roteamento é útil em uma única rede física, como uma *Ethernet,* mas não é prático quando muitas redes diferentes são reunidas.

O roteamento indireto permite aos roteadores encaminharem datagamas IP com base em seus endereços IP de destino. Para qualquer datagrama, um roteador determina o roteador do próximo salto (*next hop*), ao longo do caminho até o destino, e empacota o datagrama em um quadro no nível do enlace. Um roteador só precisa responder a uma

pergunta muito simples: dado um datagrama IP que transporta um endereço de *host* de destino específico, por qual interface o datagrama deve ser enviado e para qual salto seguinte? Portanto, ao receber um pacote o roteador precisa ler o cabeçalho IP, determinar para qual rede de destino o pacote pertence e, através da leitura de uma tabela de roteamento, encaminhá-lo para o próximo salto. Essa é uma forma de algoritmo de pesquisa que apanha um endereço IP e deriva em um identificador de interface e um endereço IP do salto seguinte. A implementação das tabelas de roteamento varia muito de um fabricante de roteador para outro, e os requisitos em conflito de desempenho e ocupação de dados são responsáveis pelas vantagens competitivas que eles afirmam [6].

Os roteadores obtêm seu conhecimento da topologia das redes remotas através dos pares vizinhos ou das informações configuradas manualmente por um administrador. Assim, esses equipamentos constroem uma tabela de rotas que descreve como encontrar as redes remotas. Estando uma rede diretamente conectada ao roteador, o mesmo saberá como alcançá-la, não sendo necessário nenhum mecanismo de criação de rotas. Caso a rede não esteja diretamente conectada, será necessário o uso de um processo de roteamento estático, o que significa dizer que um administrador inseriu manualmente todas as localizações das redes na tabela de roteamento, ou de um processo de roteamento dinâmico, que nesse caso o administrador pode fazer uso de algum protocolo de roteamento onde as rotas são divulgadas automaticamente.

Quando na construção das tabelas de roteamento, um protocolo em um roteador se comunica com o seu vizinho usando o mesmo protocolo, informando quais são as redes que podem ser alcançadas por ele e vice-versa. Em seguida, os roteadores se atualizam sobre todas as redes que sabem e que aprenderam, colocando essas informações na tabela.

Na prática, os detalhes internos de um roteador são ligeiramente mais complexos do que uma única tabela de roteamento. Como é possível observar na figura 10, o roteador pode apanhar suas entradas em diversas origens, incluindo a configuração do administrador, a descoberta por meio de protocolos como ICMP, o compartilhamento de informações de rota com redes IGP (*Interior Gateway Protocol*) e a distribuição de rota de roteadores parceiros, utilizando um EGP (*Exterior Gateway Protocol*). As rotas descobertas por meio de protocolos de roteamento normalmente estão sujeitas a alguma filtragem de importação de rota, de acordo com as preferências configuradas no roteador local. Todas as rotas aceitáveis a partir do protocolo de roteamento são

combinadas com as rotas estáticas configuradas e as rotas diretas descobertas, para formar que é conhecido como RIB (*Routing Information Base*).

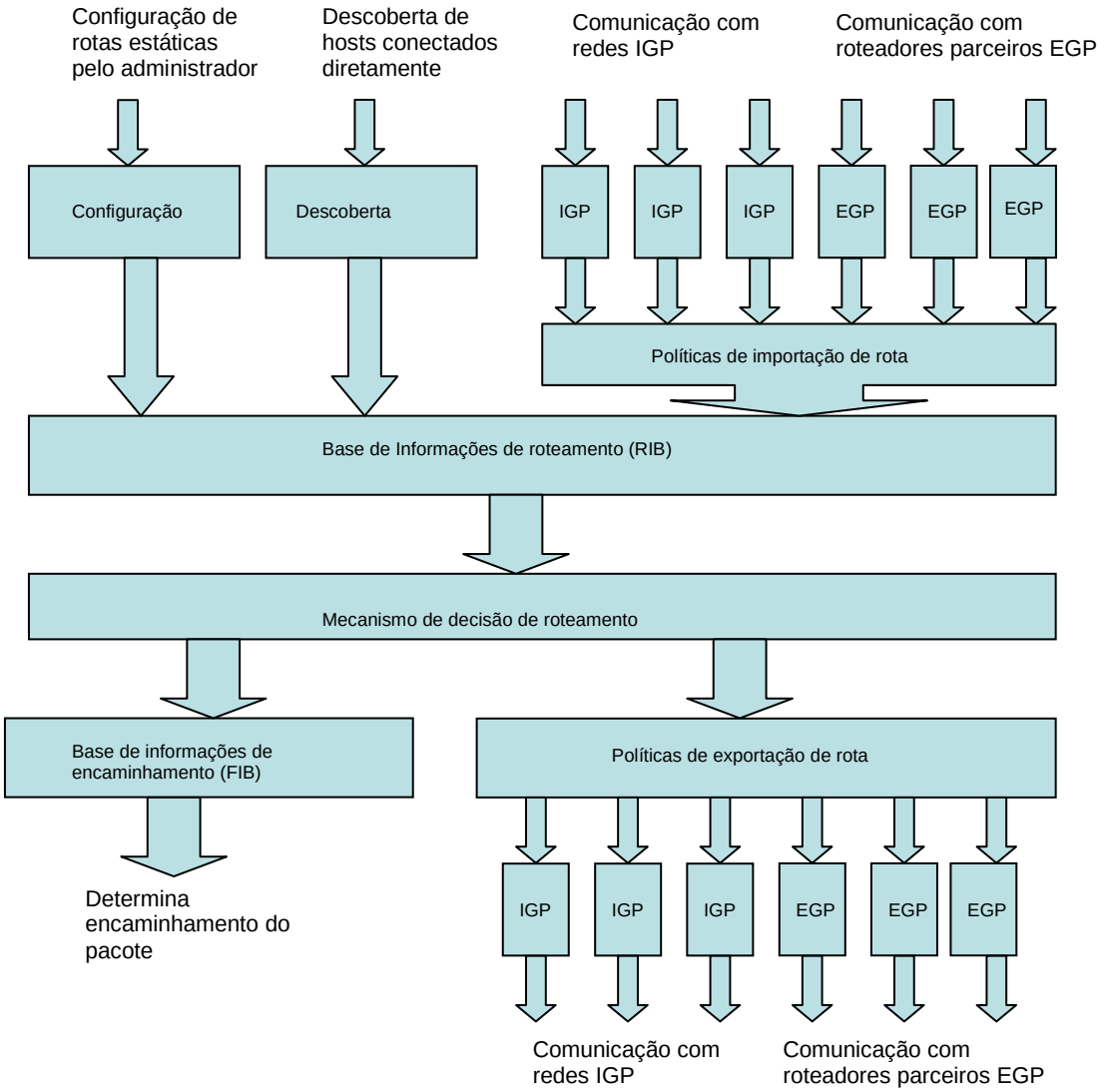


Figura 10: Diagrama interno de um roteador [6].

A FIB (*Forwarding Information Base*) oferece instruções não ambíguas ao componente do roteador que encaminha os pacotes de dados. Dado um endereço de destino, a FIB dirá ao roteador por qual interface ele deve encaminhar o pacote e o endereço do salto seguinte. Em muitas aplicações, a FIB também divulgará outras informações úteis, como por exemplo, o endereço MAC do salto seguinte, para que o componente de encaminhamento só precise realizar uma operação de pesquisa por pacote.

A FIB contém as melhores rotas definitivas, de acordo com as políticas de decisão de roteamento locais, e o roteador precisa compartilhar essas informações com os outros roteadores nas redes às quais pertence. Contudo, os protocolos de roteamento

se preocupam não apenas com as melhores rotas, mas com todas as rotas disponíveis [6].

A grande vantagem do uso de protocolos de roteamento dinâmico sobre o roteamento estático é a possibilidade de adaptação dinâmica das rotas em situações de falhas ou congestionamentos detectados. Se alguma mudança ocorrer na rede, os protocolos de roteamento dinâmico informam automaticamente todos os roteadores sobre o evento. Em se tratando de um roteamento estático, o administrador é responsável por atualizar todas as mudanças manualmente em todos os roteadores. Tipicamente, nas grandes redes, uma combinação do roteamento dinâmico e estático é usada [15].

3.1 Comutação e Roteamento

O mecanismo de aprendizado e manutenção do conhecimento da topologia de rede é considerado como a função de roteamento. O movimento real do tráfego transiente por meio do roteador, da interface de entrada para uma interface de saída, é uma função separada considerada como a função de comutação [16]. Comutação é o processo de apanhar um quadro de uma entrada e enviá-lo por uma saída apropriada, baseado na informação da camada de enlace. Os métodos para a comutação de quadros baseiam-se nas informações de endereço da camada de enlace. Já os recursos e as funcionalidades dos *switches* de camada 3 e dos roteadores têm diversas semelhanças.

Os *switches* de camada 3 são uma mistura de tecnologias de *bridge* (ponte) e roteadores. Ou seja, um *switch* de camada 3 é um roteador, embora mais rápido e mais sofisticado. Eles operam montando uma tabela de endereços IP mapeados para interfaces de saída e pesquisam o endereço de destino de um pacote para determinar a interface por meio da qual o pacote deve ser encaminhado. Eles passam o pacote de uma interface para a seguinte no nível de rede, da mesma forma que uma ponte passa um quadro em um nível de enlace [6]. Nos roteadores, de modo geral, a comutação de pacotes acontece em *software*, usando mecanismos baseados em microprocessadores, enquanto em um *switch* de camada 2 ou 3, o encaminhamento dos quadros ou pacotes é realizado usando *hardware* ASIC (*application specific integrated circuit*) [17].

Durante um tempo, os *switches* de camada 3 eram a esperança para a Internet. Eles não exigiam o cálculo continuo de rotas toda vez que os protocolos de roteamento distribuíam novas informações e poderiam colocar sua tabela de comutação em circuitos

integrados (ASIC) para encaminhar muito mais rápido. Contudo, logo ficou aparente que uma nova geração de roteadores poderia ser criada, que também usava ASICs para o encaminhamento, enquanto mantinha a flexibilidade e a reatividade dos protocolos de roteamento [6].

A comutação de pacotes oferece um novo modelo para o encaminhamento de dados na rede. Em vez de encaminhar cada pacote com base no endereço da camada de rede e nas informações distribuídas por protocolos de roteamento, os nós na rede podem usar rótulos transportados nos pacotes e informações de comutação de rótulos distribuídas por novos protocolos ou extensões dos protocolos existentes.

A comutação de pacotes IP é o processo de encaminhar pacotes de dados dentro de uma rede, baseado em algum rótulo associado com cada pacote. Em alguns sentidos, o roteamento IP tradicional é uma forma de comutação de pacotes – cada pacote transporta um endereço IP de destino que pode ser usado para determinar o próximo salto no caminho em direção ao destino realizando, uma consulta na tabela de roteamento. Há, no entanto, muitas limitações para o roteamento, e a comutação de rótulos foi desenvolvida para revolver algumas delas. Esse assunto será mais detalhado no capítulo 4 desta dissertação onde serão tratados os aspectos do protocolo MPLS.

É possível concluir a discussão sobre comutação e roteamento IP descrevendo uma idéia que foi vista originalmente como um meio de melhorar o desempenho da Internet. A chamada comutação de rótulos multiprotocolo, tenta combinar algumas das características da comutação com a flexibilidade e a robustez do roteamento. O casamento dessas duas tecnologias, aparentemente opostas, tem feito com que o MPLS seja bem aceito nas comunidades da engenharia e da Internet, como a solução perfeita para encaminhamento de pacotes em redes legadas e não legadas, com a possibilidade de implementar diversos serviços como, por exemplo, a engenharia de tráfego que é o foco principal desta dissertação.

3.2 Protocolos de Roteamento

O roteamento é um problema de teoria de grafos. A figura 11 mostra um grafo representando uma rede. Os nós do grafo são rotulados de A até F, esses nós podem ser *hosts*, *switches*, roteadores ou redes [4].

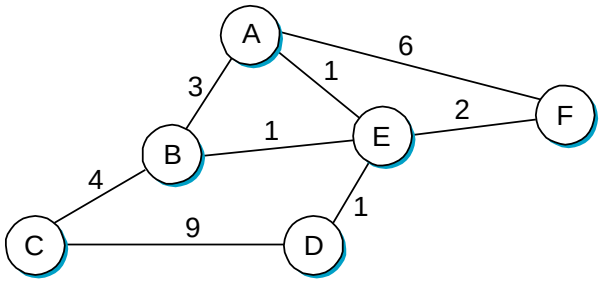


Figura 11: Rede representada como um grafo [4].

O roteamento apresenta a proposta de solucionar o problema encontrando o caminho de menor custo entre dois nós quaisquer da rede. No grafo da rede apresentada na figura 11 seria possível simplesmente calcular todos os caminhos mais curtos e definir uma métrica, porém essa técnica estática possui limitações tais como:

- ela não trata de falhas de nó ou de enlace;
- ela não considera o acréscimo de novos nós ou enlaces;
- ela implica que os custos da borda não podem mudar, embora seja possível razoavelmente querer atribuir temporariamente um custo alto a cada enlace que esteja congestionado.

Por isso, diante dos pontos apresentados, o roteamento na maioria dos *backbones* utiliza os protocolos de roteamento dinâmicos. Os protocolos de roteamento foram criados em resposta à demanda por tabelas roteamento dinâmicas, onde essas deveriam ser as melhores e menores possíveis. Um protocolo de roteamento é uma combinação de regras e procedimentos que possibilita aos roteadores na rede informarem as mudanças entre os pares. Eles permitem que os roteadores compartilhem tudo o que souberem em relação à rede diretamente conectadas ou em relação aos vizinhos dos outros roteadores. Assim como informações de falha em linhas (enlaces) de transmissão como, por exemplo, um circuito de STM (*Synchronous Transport Module*) que esteja mudando seu estado repentinamente, e caminhos que estão com alto nível de congestionamento. Por esse motivo, as tabelas de roteamento normalmente são recalculadas em tempos definidos ou quando uma quantidade mínima de mudança na rede estiver sendo observada, pois em casos de várias mudanças isso acarretaria em uma alta necessidade do processamento dos roteadores.

Cada agrupamento organizacional de computadores é definido como um AS (*Autonomous System*), ou seja, um sistema que pode operar isoladamente de todos os outros agrupamentos. Dentro de um AS, as informações de roteamento em geral são bastante distribuídas, e um roteador pode claramente ver o caminho pela rede AS até

outro roteador dentro do mesmo AS. Os protocolos de roteamento dentro de um AS e entre ASs são diferentes. Roteadores dentro de um AS trocam informações de roteamento através de um protocolo comum conhecido como IGP (*Interior Gateway Protocol*), já os roteadores que fazem comunicação entre ASs, o fazem através de um protocolo de roteamento chamado de EGP (*Exterior Gateway Protocol*) [15].

Para reduzir o tráfego de roteamento, as redes IP (assim como as redes MPLS) podem ser divididas em diferentes sistemas autônomos (*autonomous sytems*), também conhecidos como domínios de roteamento.

A natureza distribuída dos algoritmos de roteamento é um dos principais motivos pelos quais esse tem sido um campo de pesquisa e desenvolvimento tão rico. Existem muitos desafios para fazer com que os algoritmos distribuídos funcionem bem, por exemplo, os algoritmos distribuídos levam a possibilidade de dois roteadores, em determinado instante, terem idéias diferentes a respeito do caminho mais curto até algum destino [4].

Os protocolos de roteamento possuem um grande conjunto de informações e características diferenciadas que é possivel categorizar – *classful* versus *classless* e *distance vetor* versus *link-state*.

O nome roteamento por vetor de distância é encontrado também nas literaturas como algoritmo de *Bellman-Ford* [16], a idéia por trás do algoritmo com vetor distância é sugerida por seu nome: cada nó constrói um *array* unidimensional (um vetor) contendo as "distâncias" (custos) até todos os outros nós e distribui esse vetor aos seus vizinhos imediatos. A suposição inicial para o roteamento com vetor distância é que cada nó conhece o custo do enlace para cada um de seus vizinhos conectados diretamente. O melhor caminho pode estar relacionado com várias medidas, sendo que o número de roteadores na rota (*hop count*) é a mais utilizada. Um enlace que esteja inativo recebe o custo infinito.

As rotinas periódicas de atualizações de roteamento geradas pela maioria dos protocolos de roteamentos vetor distância vão apenas para os dispositivos de roteamento conectados diretamente. Num ambiente de vetor distância puro, as atualizações de roteamento incluem uma tabela completa de roteamento, a qual pode ser chamada de atualização integral, ou seja, a troca de toda a tabela de roteamento. Ao receber uma tabela completa de um vizinho, um roteador pode verificar todas as rotas conhecidas e, em seguida, fazer as alterações na tabela local, com base nas informações atualizadas recebidas. Esse processo de roteamento é muito simples, e na prática pode ser lento,

provocando um alto tempo de convergência na rede (tempo que os roteadores levam para estabilizar as tabelas de roteamento de acordo com uma mudança ocorrida na topologia) e possíveis *loops* de roteamento. Como exemplos de protocolos vetor distância existem o RIP (*Routing Information Protocol*) definido na RFC 1058, o RIPv2 definido na RFC 1723 e o RIPng (*Routing Information Protocol next generation*) para IPv6 definido na RFC 2080. Já o IGRP (*Interior Gateway Routing Protocol*), de propriedade da *Cisco*, embora também seja protocolo vetor distância, é pouco usado no mercado, pois foi substituído por um protocolo de roteamento mais avançado, o EIGRP (*Enhanced Interior Gateway Protocol*) e exibe algumas características de *link-state*. Destaca-se que todos os protocolos citados acima são do tipo IGP [16].

O roteamento por vetor distância aplica o enfoque incremental para a construção e distribuição de informações de caminho. Cada roteador distribui rotas inteiras, e seus vizinhos as selecionam a partir dessas rotas, que devem ser acrescentadas na tabela de roteamento, antes de passar adiante um conjunto de rotas até seus próprios pares.

O roteamento por estado de enlace (*link-state*) não distribui rota alguma, mas troca informações de topologia que descreve a rede. Cada nó é responsável por anunciar os detalhes dos enlaces que aceita e por passar adiante informações semelhantes que recebem de outros roteadores. Desse modo, cada roteador na rede monta um banco de dados completo dos enlaces disponíveis e quais nós eles interconectam. Com efeito, cada roteador possui o mapa completo e idêntico da rede.

No roteamento por vetor distância, cada roteador envia informações de roteamento por seus enlaces – não importa se existe um roteador no enlace para receber as informações ou não. No roteamento por estado de enlace, existe uma ligação mais próxima entre os roteadores vizinhos; eles precisam se tornar parceiros, para estabelecer um relacionamento de parceria, a fim de haver a troca de informações de estado de enlace. Essa primeira etapa é obtida por meio de um protocolo de *Hello*, que em cada roteador envia uma mensagem "*Hello*" a cada enlace para se apresentar aos vizinhos. O formato e o conteúdo exato da mensagem de "*Hello*" dependem do protocolo de roteamento por estado de enlace em uso, mas ele precisa identificar com exclusividade o enlace em que a mensagem foi enviada (usando o endereço IP) e o roteador que enviou a mensagem. O receptor de mensagem *Hello* responde com seu próprio *Hello*, de modo que ambos os roteadores conheçam um ao outro.

Após a troca inicial da mensagem *Hello*, os roteadores trocam e negociam os parâmetros que usarão para controlar sua associação e depois se declaram como

parceiros. A primeira coisa que os parceiros fazem é sincronizar o banco de dados de estado de enlace (*Link State Database*) trocando mensagens que relatam cada enlace que conhecem. Para um roteador novo, isso começará apenas com enlaces locais que sabem que estão ativos – os enlaces às subredes conectadas e os enlaces recentemente abertos até o parceiro – mas, se o roteador já estiver recebido informações de outros roteadores, o sincronismo incluirá informações sobre outros enlaces dentro da rede. As informações sobre cada enlace são enviadas com LSA (*Link State Advertisement*) ou *Link State Packet* que é formatado e embutido em uma mensagem de acordo com regras do protocolo de roteamento específico[6].

Dessa forma, dois roteadores que se tornam parceiros rapidamente alcançam uma posição de ter banco de dados de estado de enlace idênticos, ou seja, ambos conhecem a mesma lista de enlaces dentro da rede. Daí por diante, como funcionalidade desse algoritmo, sempre que um enlace muda de estado, o dispositivo que detecta a alteração cria um LSA, que diz respeito àquele enlace (rota) e, em seguida, o LSA é propagado para todos os dispositivos vizinhos que usam um endereço especial de *multicast* (endereço de um roteador para um grupo de roteadores). Cada dispositivo de roteamento recebe uma cópia do LSA, encaminha-o para todos os dispositivos vizinhos e, em seguida, atualiza a sua base de dados topológica. Esse encaminhamento de LSA é conhecido como *flooding*, e é necessário para garantir que todos os dispositivos de roteamento aprendam sobre as alterações, para que eles possam atualizar seus bancos de dados e criar uma tabela de roteamento atualizada, que irá refletir a nova topologia [16].

O processo de inundação (*flooding*) poderia ocupar uma grande quantidade de largura de banda da rede e resultar em LSAs enviados aos roteadores que já possuem a informação. Por isso, a maioria dos protocolos de roteamento que usam o algoritmo de estado de enlace requer um projeto hierárquico e assim, é possível reduzir a necessidade de *flooding* de LSA para todos os dispositivos do domínio de roteamento, porque o uso de áreas (segmentação lógica formada por alguns roteadores) restringe *o flooding* ao limite lógico da área e não a todos os dispositivos do domínio. Em resumo, quaisquer mudanças que ocorram em uma área devem causar o recálculo da tabela de roteamento apenas naquela área, e não em todo o domínio.

O envelhecimento do LSA cria seus próprios problemas, pois os roteadores agora precisam realizar alguma ação para evitar que os banco de dados de estado de enlace se esvaziem. Isso é obtido renovando-se periodicamente (*flooding*) o conteúdo do banco de dados de estado de enlace de um roteador nos parceiros. Como uma renovação

em lote provavelmente congestionaria a rede, temporizadores individuais são mantidos para cada LSA.

Como último ponto operacional no roteamento por estado do enlace, destaca-se que os roteadores precisam ser capazes de distinguir entre os LSAs que se referem ao mesmo enlace. Como os LSAs podem chegar por diferentes caminhos na rede, é importante poder sequenciá-los para determinar se o enlace ficou interrompido e depois voltou ou vice-versa. Os temporizadores não ajudaram, pois o roteador de origem poderia reiniciar seu relógio a qualquer momento, de modo que são usados números de seqüência simples. Diversos esquemas de contagem são usados nos protocolos de roteamento por estado de enlace para gerar números de seqüência de LSA e tratar o fato dos LSAs antigos poderem persistir na rede por muito tempo. Outra questão que precisa ser resolvida com números de seqüência é o fato de que quando um roteador reinicia, ele começará a contar novamente na origem dando a impressão de que os LSAs antigos, repetidos na rede, são mais precisos do que aqueles anunciados no reinício. Além disso, a contagem linear simples tem um problema inerente porque o valor inteiro usado para manter e trocar o número de seqüência, em algum ponto, será preenchido e precisará retornar ao ponto inicial.

Esses problemas de contagem são resolvidos em alguns protocolos de roteamento por estado de enlace usando um espaço de número de seqüência em forma de "pirulito" (*lollipop*). Na figura 12, é apresentado esse método, quando o roteador reinicia ele começa a contar em um valor de origem bem conhecido (n). Os números de seqüência são incrementados linearmente até que alcançam o início de um *loop* de contagem (m). Uma vez no *loop*, a contagem continua de forma incremental, mas retorna ao valor de partida do *loop* quando o máximo é alcançado (x).

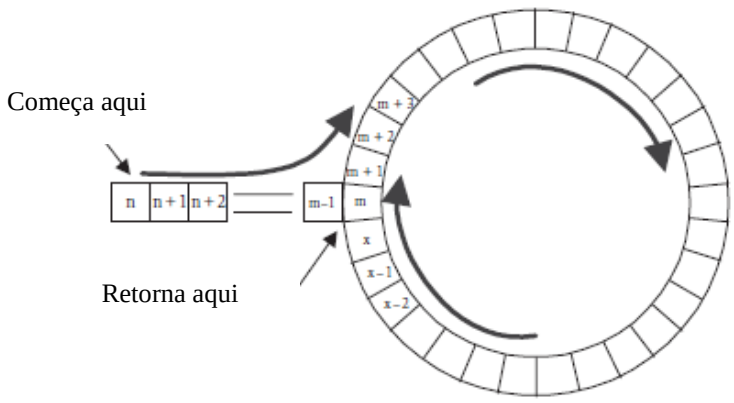


Figura 12: Contagem do número de seqüência em forma de *lollipop* (pirulito) [6].

Um exemplo de um protocolo de roteamento por estado de enlace é o OSPFv2 definido na RFC 1247, que é o protocolo IGP de maior utilização no momento, inclusive no MPLS [9], e na RFC 5340 foi definido o OSPFv3 para IPv6 [11].

O IS-IS (*Intermediante System to Intermediate System*) é um protocolo IGP com métrica de roteamento por estado de enlace desenvolvido pela ISO e especificado nos mesmos moldes que o OSPF. O IS-IS tem menos utilização que o OSPF [9]. Como já foi apresentado o EIGRP com um protocolo hibrido sendo este tecnicamente um vetor distância avançado, ele apresenta alguns recursos de estado de enlace.

O roteamento vetor distância e o de estado de enlace são protocolos de roteamento intradomínios. Quando as redes se tornam tão grandes que o número de áreas se torna difícil de administrar, elas são divididas em sistemas autônomos distintos. Cada sistema autônomo executa um IGP e pode ser desmembrado em áreas por conta própria. Os sistemas autônomos trocam informações de roteamento usando alguns outros meios, como um protocolo de roteamento por vetor caminho (*path vector*). Para esse tipo roteamento, entre sistemas autônomos, é utilizado o EGP, como por exemplo, o protocolo BGP (*Border Gateway Protocol*), que foi definido na RFC 4271.

O roteamento por vetor caminho, de várias maneiras, é semelhante ao roteamento por vetor de distância, mas foi melhorado bastante pela inclusão do caminho inteiro nos anúncios de rota. Isso permite que os roteadores identifiquem facilmente os *loops* de roteamento e, por isso, removam quaisquer problemas com a contagem para o infinito. A desvantagem é que anúncios de rota são muito maiores, por que cada rota pode incluir vários saltos [6].

O roteamento vetor caminho tem uma vantagem significativa, uma vez que ele permite que um roteador escolha uma rota com base não apenas na distância ou custo associados à rota, mas também examinando os roteadores e enlaces que compreendem o caminho, ou seja, as decisões são tomadas em políticas de roteamento. Essas políticas podem ser elaboradas por meio de regras locais, baseadas no conhecimento de enlaces que são passíveis de erro, vulneráveis a ataque de segurança ou dispendiosos financeiramente. Desse modo, é possível determinar os melhores caminhos para encaminhamento de pacotes.

Um problema sério com o roteamento baseado em vetor caminho é que cada roteador na rede pode aplicar diferentes políticas. O roteamento por vetor de distância e estado do enlace são técnicas de roteamento, mas utilizam uma política de menor custo

(métrica), e todas as rotas na rede se utilizam da mesma política – isso significa que é possível prever o comportamento da rede e saber a rota que o diagrama seguirá em uma rede estável uma vez iniciada. Não é difícil ver como essa previsibilidade poderia falhar catastroficamente, assim que roteadores aplicassem políticas diferentes um do outro ou diferentes da política de menor custo.

Um exemplo deste tipo de ocorrência foi o sequestro do prefixo do YouTube. Por determinação do Governo Paquistanês, o tráfego do YouTube deveria ser bloqueado para evitar o acesso ao trailer de um filme anti-Islâmico. Para cumprir essa ordem, a operadora PakistanTelecom gerou o anúncio de um prefixo mais específico do que o utilizado pelo YouTube, com o intuito de direcionar todos os acessos a ele para uma página que dizia "YouTube was blocked". No entanto, a operadora anunciou essa nova rota a seu *upstream provider* (primeiro erro), que, além de não verificar a nova rota (segundo erro) a propagou por toda a Internet (terceiro erro). Com isso, todo o tráfego do YouTube passou a ser direcionado para o Paquistão e foi descartado [18] [11]. No sítio do YouTube pode ser visto um vídeo gerado para esse caso [19].

Dentro da Internet, existe a exigência de conectar redes e sistemas autônomos divergentes que compõem a Internet em geral, e isso é feito usando EGPs. Esses protocolos utilizam uma propriedade de resumo de rota dos protocolos de roteamento por vetor caminho para permitir que os sistemas autônomos sejam caracterizados dentro das rotas anunciadas, tornando-os muito mais escaláveis e flexíveis.

São apresentadas na tabela 6 as características dos protocolos de roteamento que serão utilizados no decorrer desta dissertação.

Tabela 6: Comparação dos protocolos de roteamento escaláveis.

Protocolo	**Interior ou Exterior**	**Vetor Caminho ou Estado de Enlace**	**Hierarquia Requerida**
OSPF	Interior	Estado de Enlace	Sim
IS-IS	Interior	Estado de Enlace	Sim
BGP	Exterior	Vetor Caminho	Não

3.2.1 IS-IS – *Intermediate System to Intermediate-System*

O IS-IS é um protocolo de roteamento de estado de enlace utilizado entre roteadores nos protocolos de rede OSI (*Open Systems Interconnection*) projetado pela

ISO (*International Standards Organization*). De acordo com [20], o protocolo IS-IS tem maior popularidade na Internet que na arquitetura OSI. No OSI, o equipamento terminal e os *hosts* são chamados de ES - sistemas finais (*End Sytems*), os roteadores são conhecidos como IS - sistemas intermediários (*Intermediate Systems*). Dessa forma, o protocolo de roteamento que trabalha entre roteadores é o IS-IS. Esse protocolo foi especificado no ISO 10589, e é voltado para CLNP (*Connectionless Network Protocol*) da ISO, com base no protocolo de roteamento desenvolvido pela DEC para incorporação no *DECnet Phase 5*. Ele é um protocolo de estado de enlace, com muitas das capacidades do OSPF, o que não é surpresa, pois o desenvolvimento do OSPF tinha informações do trabalho feito pela ISO [6]. Embora o mesmo tenha sido projetado para roteamento ISO, ele foi adaptado para dar suporte ao TCP/IP e CNLP simultaneamente, daí a origem do termo *Integrated* IS-IS, muito utilizado atualmente para se referir ao uso desse protocolo nas redes IPs. A RFC 1142 foi recentemente copiada do ISO 10589, mas a RFC 1195 focaliza o uso do *Integrated* IS-IS unicamente em um ambiente IP.

Mensagens IS-IS não são transportadas em datagramas IP, diferente de todos os outros protocolos de roteamento IP. As mensagens são chamadas PDU (*Protocol Data Unit*), são encapsuladas diretamente no quadro da camada de enlace e, portanto, o IS-IS trabalha ao lado do IP na camada de rede. O endereçamento OSI utiliza um formato genérico e um modelo hierárquico que serve para descrever áreas, roteadores, interfaces e serviços de protocolos em um nó. O IS-IS admite dois níveis de hierarquia, como no OSPF: a área do *backbone* é conhecida como nível 2 (L2) e outras áreas são nível 1 (L1) [6].

Há algumas diferenças entre o IS-IS e o OSPF, uma delas é a forma como ambos manipulam os pacotes de "*Hello*", que são pacotes utilizados para formação das adjacências entre os vizinhos. O IS-IS utiliza uma PDU de *Hello* para descobrir e manter adjacências. Existem três tipos de PDU *Hello* para uso em circunstâncias diferentes, e cada um transporta vários campos adicionais do cabeçalho [6].

No IS-IS é usado o NLPID (*Network Layer Protocol Identifiers*), para informar qual o protocolo de rede está sendo utilizado. Não foi desenvolvida uma nova versão do IS-IS para trabalhar com IPv6 foram adicionados novas funcionalidade na versão em uso, com por exemplo dois novos TLVs (*Type-Length-Variable*), IPv6 *reachability*, IPv6 *Interface Address*. Foi criado um novo identificador da camada de rede – IPv6 NLPID conforme RFC 1195 – *Use of OSI IS-IS for Rooting in TCP/IP and Dual Environments*, RFC 5308 [21] – *Routing IPv6 with IS-IS*.

A RFC 3784 [22] descreve extensões para o protocolo IS-IS para suporte a engenharia de tráfego, que se realiza pela adição de novas informações relativas ao estado da rede que são úteis às métricas TE (*Traffic Engineering*), nos roteadores da rede [8]. A RFC 4124 define extensões a RFC 3784, no sentido de possibilitar a aplicação do protocolo IS-IS para MPLS DS-TE (*Multi-Protocol Label Switching Differentiated Services Traffic Engineering*), ou seja, para funcionalidade que conjuga o uso de DS (*Differentiated Services*) e de TE (*Traffic Engineering*) em redes MPLS.

3.2.2 OSPF – *Open Shortest Path First*

O protocolo de roteamento por estado de enlace desenvolvido pelo IETF, que utiliza o algoritmo SPF (*Shortest Path First*) e tem um olhar voltado para o IS-IS criado pelo ISO, é um IGP de código aberto e amplamente divulgado na literatura. Foi projetado com o objetivo de substituir o protocolo RIP (*Routing Information Protocol*), resolvendo diversas limitações que o mesmo apresenta, e abordar as necessidades das redes grandes e escaláveis, as quais não eram abordadas pelo RIP. OSPF envia avisos sobre o estado da conexão a todos os outros roteadores em uma mesma área hierárquica, e usa o algoritmo SPF para calcular o caminho mais curto para cada nó. O cálculo do OSPF seleciona o caminho de menor custo para uma rede, da origem ao destino, usando apenas os enlaces ativos. O que poderia parecer um problema simples não funciona bem com o aumento do número de roteadores e de enlaces em uma rede. Esse problema pode ser resolvido facilmente pelo cérebro humano com uma relativa precisão, examinando um mapa em escala, mas é difícil converter para um problema sintetizado para um computador.

O roteamento por estado de enlace apresenta um problema totalmente diferente dos protocolos de roteamento por vetor distância e vetor caminho. No protocolo de roteamento por estado de enlace, cada roteador possui uma visão completa da rede, fornecida pelas informações de todos os roteadores na rede, mas os roteadores precisam construir uma tabela de roteamento do zero, usando apenas a informação do caminho mais curto. Não é uma exigência que todos os roteadores usem o mesmo mecanismo para calcular os caminhos mais curtos, porque todos eles chegam em resultados coerentes. Apesar disso, essa coerência é tão importante que os dois principais protocolos de roteamento por estado de enlace o OSPF e o IS-IS exigem o uso do algoritmo *Shortest Path First*. O algoritmo de Dijkstra é um meio relativamente eficaz e

simples de criar uma tabela de roteamento a partir de um conjunto de informações de estado de enlace. Um ponto importante é que o algoritmo deve preencher totalmente a tabela de roteamento de uma só vez calculando os caminhos mais curtos para todos os destinos. A partir de um nó específico, cada roteador vizinho é acrescentado a uma lista de candidatos, que é ordenada por custos (métricas) dos enlaces até os vizinhos, com o enlace de menor custo em primeiro lugar. O algoritmo constrói uma tabela de roteamento, onde o roteador local é o primeiro na tabela, selecionando o roteador vizinho com o enlace de menor custo para ser o próximo nó na árvore.

O algoritmo, então, prossegue para examinar os vizinhos na ponta do ramo da árvore, criando uma lista de candidatos e seleciona o vizinho de menor custo que ainda não esteja na árvore de único ramo, dando o caminho mais curto para um grupo de *hosts*. O algoritmo então descarta o início da lista de candidatos na ponta do ramo e processa o próximo membro da lista. Isso bifurca a árvore e visita os vizinhos até que as rotas para um novo conjunto de *hosts* tenham sido completadas.

Essa interação se repete até que a lista de candidatos no ramo esteja vazia. Nesse ponto, o algoritmo recua um nó na arvore e continua a trabalhar na lista de candidatos a partir deste ponto. O algoritmo termina quando a lista de candidatos no nó de base estiver vazia. Uma árvore foi montada, com cada ramo representando o caminho mais curto para o *host* no extremo da árvore. Assim, não se tem um roteador duas vezes na árvore, e existe uma rota para cada *host*.

Uma observação importante a respeito do algoritmo de Dijkstra é que ele examina cada enlace na rede exatamente uma vez enquanto monta a árvore de roteamento. Toda vez que um enlace é examinado, um novo vizinho é encontrado, e esse vizinho precisa ser comparado com as entradas na lista de candidatos para verificar se já está lá, e para inserir o vizinho no ponto correto da lista. Se houver *l* enlaces de um roteador e *n* vizinhos, o processo de classificação será uma função da ordem *log(n)*, e o algoritmo possui uma eficiência na ordem de *l*log(n)* para cada nó. Uma vez que cada enlace é visitado apenas uma vez durante o todo o algoritmo, pode-se somar essa eficiência por todos os nós para alcançar uma eficiência geral da ordem de:

$$\Sigma(l*\log(n)) = L*\log(N), \quad (1)$$

onde L é o número total de enlaces na rede, e N é o número total de nós. Nitidamente, em uma rede com malha totalmente conectada, L = N(N-1)/2 e a eficiência é mais próxima de N^2 [6].

Entender o SPF é fundamental para entender o CSPF (*Constrained Shortest Path First*) da engenharia de tráfego do MPLS, que é baseado no algoritmo Dijkstra. A seguir será analisada uma rede como a apresentada na figura 13.

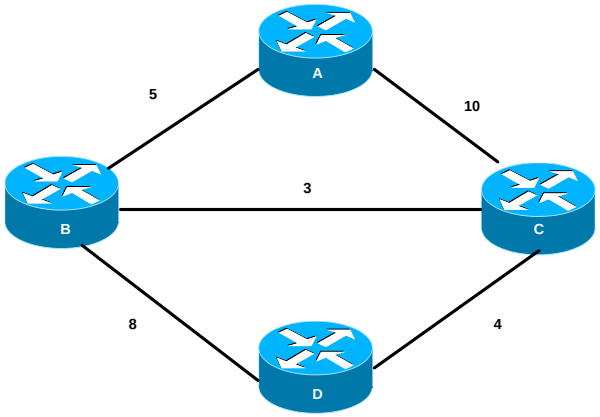


Figura 13: Topologia de rede simples, demonstrado o algoritmo SPF [5].

Para o do roteador “A” na figura 13, que está rodando o algoritmo do SPF, este gera a tabela de roteamento e, depois que cada roteador tiver feito o processo de *flooding* para toda a rede, todos os roteadores saberão a respeito de todos os outros roteadores e os enlaces entre eles. Dessa forma, o banco de estado de enlace em cada roteador se parece com a tabela 7.

Tabela 7: Combinação do mapa de Classes.

Roteador	Pares de {vizinho, custo}
A	{B, 5} {C,10}
B	{A, 5} {C,3} {D,8}
C	{A, 10} {B, 3} {D,4}
D	{B, 8} {C,4}

No cálculo do SPF, cada roteador mantém duas listas:

- uma lista de nós que é conhecida como estando no caminho mais curto até o destino, que é chamada lista PATH;
- uma lista de *hops* seguintes, que podem ou não estar no caminho mais curto até esse destino, que é chamada lista de tentativas, ou lista de TENT.

Para o roteador que está calculando, cada lista é uma tabela de trios {roteador, distância, *hop* seguinte}.

Em seguinda será analisado como o roteador A da figura 12 monta a tabela de roteamento:

Passo 1 – Coloque em "*self*" na lista PATH com uma distância de 0 e um *hop* seguinte igual a *self*. Então, seu banco de dados se parece com a tabela 8.

Tabela 8: Lista PATH e TENT para o roteador A com SPF.

Lista PATH	**Lista TENT**
{A, 0, 0}	(vazia)

Passo 2 – Apanhe o nó recém-colocado na lista PATH e chame-o de nó PATH. Examine a lista de vizinhos desse nó. Acrescente cada vizinho dessa lista à lista TENT com um *hop* seguinte do nó PATH, a menos que o vizinho já esteja na lista TENT ou PATH com custo inferior. Se o nó recém-incluído em lista TENT já existir na lista, mas com um custo maior, substitua o nó de custo maior pelo nó atualmente em consideração. A lista TENT é apresentada na tabela 9.

Tabela 9: Lista PATH e TENT para o roteador A após o passo2 com SPF.

Lista PATH	**Lista TENT**
{A, 0, 0}	{B, 5, B}
	{C, 10, C}

Passo 3 – Encontre o vizinho da lista TENT com menor custo, acrescente este vizinho na lista PATH e repita o passo 2. Se a lista TENT estiver vazia, pare. A tabela 10 apresenta a lista PATH e TENT neste passo.

Tabela 10: Lista PATH e TENT para o roteador A após o passo3 com SPF.

Lista PATH	**Lista TENT**
{A, 0, 0}	{C, 10, C}
{B, 5, B}	

Passo 4 – Repita o passo 2. Apanhe o nó recém-colocado na lista PATH e chame-o de nó PATH. Examine a lista de vizinhos desse nó. Acrescente cada vizinho dessa lista à lista TENT com *hop* seguinte do nó PATH, a menos que o vizinho já esteja na lista TENT ou PATH com custo inferior. Se o nó recém incluído em uma lista TENT já existir na lista, mas com um custo maior substitua o nó de custo maior pelo nó atualmente em consideração.

Neste caso foram examinados os vizinhos do roteador B. O roteador B da figura 13 tem um enlace para C com custo 3 e um enlace para D com custo 8. O roteador C, com um custo 5 (para ir de "*self*" até B) + 3 (para ir de B até C) = 8 (o custo total de A até C via B) e o *hop* seguinte de B, é acrescentado à lista TENT, assim como o roteador D, com custo de 5 (o custo para ir do nó raiz até B) + 8 (o custo para ir de B até D) = 13 e um *hop* seguinte de B. Como o caminho até C com custo 8 passando por B é inferior ao caminho até C com um custo 10 passando por C, o caminho até C com custo 10 é removido da lista TENT. Neste passo a tabela 11 apresenta a lista PATH e TENT.

Tabela 11: Lista PATH e TENT para o roteador A após o passo 4 com SPF.

Lista PATH	**Lista TENT**
{A, 0, 0}	~~{C, 10, C}~~
{B, 5, B}	{C, 8, B}
	{D, 13, B}

Passo 5 – Encontre o caminho a lista TENT com menor custo, acrescente esse caminho na lista PATH e repita o passo 2. Se a lista TENT estiver vazia, pare.

Tabela 12: Lista PATH e TENT para o roteador A após o passo 5 com SPF.

Lista PATH	**Lista TENT**
{A, 0, 0}	{D, 13, B}
{B, 5, B}	
{C, 8, B}	

Os passos são repetidos para nó colocado PATH até que a lista TENT esteja vazia. Quando a lista TENT estiver vazia então a lista se torna a tabela de roteamento do roteador A que se parece com a tabela 13.

Tabela 13: Tabela de roteamento do roteador A com SPF.

Nó	**Custo**	***Hop* seguinte**
A	0	*Self*
B	5	B (conectado diretamente)
C	8	B
D	12	B

Esse é o funcionamento básico do algoritmo SPF. Depois que isso é feito, a topologia apresentada na figura 14 representa a tabela de roteamento do roteador A.

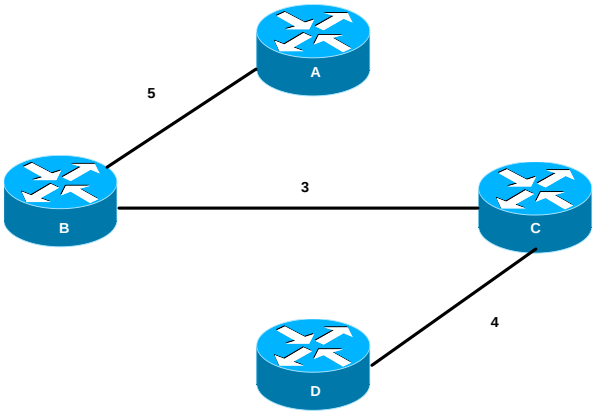


Figura 14: A visão do roteador A para rede depois de rodar o algoritmo SPF [5].

O algoritmo de Dijkstra aplica uma única restrição à escolha de uma rota: o caminho resultante precisa ser mais curto. Isso permite que os enlaces sejam configurados com valores métricos diferentes de 1, alterando o efeito do cálculo do SPF, mas, com relação ao algoritmo, ele ainda está fazendo a escolha do SPF.

Em muitos *backbones*, há a necessidade de fazer cálculos de SPF que também consideram outros atributos do tráfego nos enlaces disponíveis. Essas outras considerações oferecem restrições ao algoritmo SPF, transformando-o em cálculo de CSPF (*Constrained Shortest Path First*).

No roteamento SPF normal, não há motivo para distinguir as rotas, e é comum usar a primeira descoberta. O roteamento CSPF pode distinguir entre caminhos de mesmo custo usando restrições diferentes da métrica do custo, mas que, apesar disso, podem derivar vários caminhos que satisfaçam a todas as restrições da mesma forma. O CSPF, diferentemente do SPF, baseia-se em uma série de elementos como, por exemplo, o prefixo do endereço IP de destino de pacotes MPLS, que englobam, além do prefixo IP de destino, a topologia da rede, os seus recursos e a configuração dos atributos de tráfego.

No SPF é valido usar diferentes caminhos para o destino contendo o mesmo custo. Esse roteamento é chamado de ECMP (*Equal Cost Mulipath*), que é uma boa solução quando se trata com protocolos de roteamento por estado de enlace. Como exemplo, em um *backbone* de uma operadora de telecomunicações, pode-se decidir oferecer um balanceamento de carga de tráfego entre diferentes enlaces de diferentes velocidades usando um recurso de roteamento ECMP, onde esse serviço tem como objetivo fazer balanceamento de carga do tráfego entre diferentes caminhos de forma a distribuir melhor o tráfego pela rede. No entanto, deve ficar claro que o CSPF não está

tentando calcular todos os melhores caminhos a todos os destinos possíveis, ele está na verdade procurando um caminho para o destino.

O CSPF é particularmente relevante para o cálculo do caminho da engenharia de tráfego. O caminho é calculado para colocar um fluxo de dados bem qualificado pela rede, de modo que características do fluxo sejam conhecidas. Isso permite que o algoritmo CSPF selecione apenas os enlaces que atendam aos requisitos do fluxo.

Para que os cálculos de engenharia de tráfego do CSPF sejam feitos em um único ponto, os protocolos OSPF ou IS-IS em uso na Internet têm sido estendidos (modificados) para transportar a informação básica de disponibilidade de recursos, descrevendo a largura de banda total disponível e atualmente em uso em cada enlace. Para aplicações mais complexas, especialmente para engenharia de tráfego usando túneis IP ou MPLS, os cálculos de CSPF são muito comuns e contam com informações adicionais, normalmente compartilhadas apenas pelos protocolos de como OSPF ou IS-IS modificados para a engenharia de tráfego. Se fossem usados os protocolos RIP, RIPv2 e BGP para distribuir detalhes completos para o cálculo de engenharia de tráfego esses ficariam rapidamente sobrecarregados.

O processo que gera um caminho para um túnel TE (*Traffic Engineering*) seguir é diferente do processo normal do SPF [5], mas não muito. Existem duas diferenças principais entre o SPF e o CSPF, executadas pela engenharia de tráfego do MPLS.

Por um lado, o processo de determinação do caminho não é preparado para encontrar a melhor rota para todos os roteadores – apenas para extremidade do túnel. Isso torna o SPF ligeiramente diferente: o algoritmo pára assim que o nó ao qual está tentado chegar estiver na lista PATH, em vez de tentar calcular o caminho mais curto para todos os nós.

Além disso, agora há mais de uma métrica em cada nó. Em vez de apenas um único custo para um enlace entre dois vizinhos, há também:

- largura de banda;
- atributos do enlace;
- peso administrativo.

O funcionamento do CSPF segue os seguintes passos:

- primeiro, o trio usado no SPF normal precisa manter largura de banda, atributos do enlace e peso administrativo, o que o torna um sexteto;

- segundo, no CSPF como está sendo procurado um único caminho até um nó final, não existe compartilhamento de carga, há algumas formas de desempate quando os dois caminhos possuem os mesmos atributos: largura de banda mínima no caminho, medição IGP mais baixa até um caminho e contagem de *hops* mais baixo do caminho, o que torna o sexteto na realidade um noneto.

Será analisado o funcionamento do CSPF de uma rede como a da figura 15:

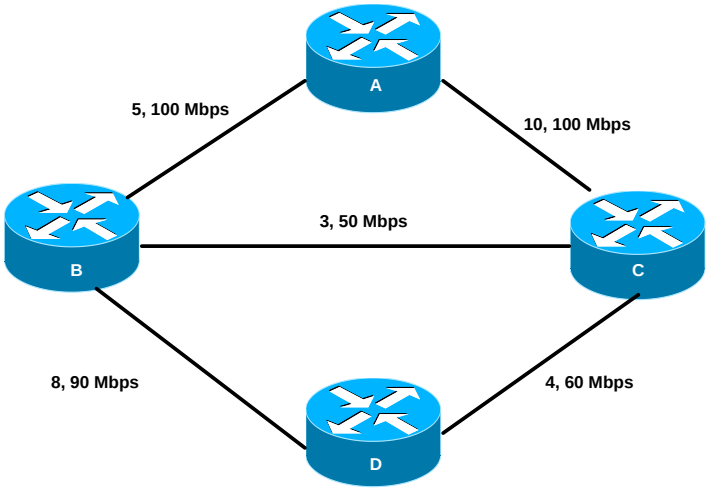


Figura 15: Topologia de rede simples demonstrando o algoritmo CSPF [5].

A figura 15 possui a mesma topologia da figura 13, mas cada enlace está anunciando sua largura de banda disponível. Com o objetivo de tornar a análise mais simples, serão apresentadas quatro propriedades do enlace nas listas PATH, TENT {enlace, custo, *hop* seguinte, largura de banda}.

Na topologia da figura 15, o roteador "A" deseja montar um túnel TE até o roteador "D", com uma largura de banda de 60 Mbps. Cada enlace lista sua métrica e sua largura de banda disponível.

Sem levar em consideração a largura de banda, o melhor caminho do roteador "A" para o roteador "D" é A➔B➔C➔D, com custo total de 12. Mas o caminho A➔B➔C➔D não tem 60 Mbps disponíveis. O CSPF precisa calcular o caminho mais curto que possui 60 Mbps disponíveis [5].

Como foi feito no algoritmo SPF, os passos do CSPF são:

Passo 1 – Coloque "*self*" na lista PATH com uma distância de 0 e um *hop* seguinte de *self*. Defina a largura de banda como N/A. Neste passo obtem-se a lista PATH e TENT conforme tabela 14.

Tabela 14: Lista inicial de PATH e TENT após o passo 1 com CSPF.

Lista PATH	**Lista TENT**
{A, 0, *self*,N/A}	(vazia)

Passo 2 – Apanhe o nó recém-colocado na lista PATH e chame-o de nó PATH. Examine a lista de vizinhos do nó. Acrescente cada vizinho nesta lista à lista TENT, com um *hop* seguinte do nó PATH, a menos que o vizinho já esteja na lista TENT ou PATH com um custo menor. Na tabela 15 são apresentadas as lista PATH e TENT.

Tabela 15: Lista PATH e TENT para o roteador A após passo 2 com CSPF.

Lista PATH	**Lista TENT**
{A, 0, *self*, N/A}	{B, 5, B, 100}
	{C, 10, C, 100}

Passo 3 – Encontre o vizinho na lista TENT com o menor custo, acrescentar esse vizinho na lista PATH e repetir o passo 2. Se a lista TENT estiver vazia ou o nó que é a cauda do túnel estiver na lista PATH, pare. Mova B da lista TENT para lista PATH e coloque os vizinhos de B na lista TENT. Pode-se verificar o resultado na tabela 16.

Tabela 16: Lista PATH e TENT para o roteador A após passo 3 com CSPF.

Lista PATH	**Lista TENT**
{A, 0, self, N/A}	{C, 10, C, 100}
{B, 5, B, 100}	{D, 13, D, 90}

Passo 4 – Coloque os vizinhos de B na lista TENT e apanhe C de TENT, colocando-o em PATH. As listas PATH e TENT são apresentadas na tabela 17.

Tabela 17: Lista PATH e TENT para o roteador A após passo 4 com CSPF.

Lista PATH	**Lista TENT**
{A, 0, self, N/A}	{D, 13, D, 90}
{B, 5, B, 100}	
{C, 10, C, 100}	

Passo 5 – Retire D da lista TENT. Neste ponto, o melhor caminho possível para D está na lista PATH, de modo que o processo termina. O fato da lista TENT está vazia

é um artefato da topologia da rede; D foi o último nó encontrado no SPF. As listas PATH e TENT são apresentadas na tabela 18.

Tabela 18: Lista PATH e TENT para o roteador A após passo 5 com CSPF.

Lista PATH	**Lista TENT**
{A, 0, self, N/A}	
{B, 5, B, 100}	
{C, 10, C, 100}	
{D, 13, D, 90}	

Foi apresentado de forma simples o cálculo de caminho no CSPF, mas na realidade esse cálculo é mais complexo do que o apresentado aqui nesta dissertação. O CSPF precisa acompanhar todos os nós no caminho, e não apenas o *hop* seguinte. Além disso, há mais coisas a considerar do que apenas a largura de banda – há também os atributos do enlace e os critérios de desempates.

Uma questão importante no critério do desempate é a seguinte: o que deve ser feito se um nó já está na lista TENT e vai ser colocado nesta lista com o mesmo custo? Será necessário encontrar um meio de diferenciar esses caminhos.

Podem-se listar as formas de desempate entre caminhos na seguinte ordem:

1. utilize o caminho com maior largura de banda mínima disponível;
2. se ainda houver um empate, use o caminho com a menor contagem de *hops*;
3. se ainda houver um empate, escolha um caminho aleatoriamente.

Tais critérios de desempate são aplicados à medida que o nó é colocado na lista TENT. A qualquer momento, determinado nó deverá estar listado apenas uma vez na lista TENT. Isso é diferente de um SPF do IGP, em que se podem ter várias maneiras de se chegar a determinado nó e podem compartilhar a carga entre eles.

Será analisada a rede da figura 16, onde se deseja estabelecer um túnel de RtrA até RtrZ com largura de banda de 10 Mbps. Cada caminho na rede se encaixa a essa descrição. Dessa forma, qual deve o caminho a ser tomado?

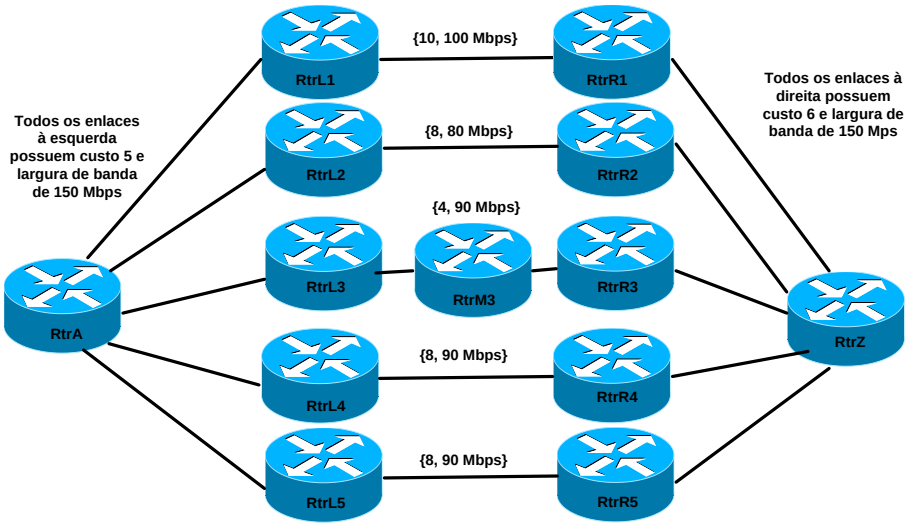


Figura 16: Rede em que os critérios de desempate do CSPF entra em ação [5].

Na figura 16 são apresentas cinco possibilidades de caminhos para estabelecer o túnel. Na tabela 19 estão listados os atributos de cada caminho.

Tabela 19: Atributos dos cinco caminhos possíveis de RtrA até RtrZ com CSPF.

Nome do caminho	**Roteadores no caminho**	**Custo do caminho**	**Largura de banda mínima no caminho**
C1	RtrA➔RtrL1➔RtrR1➔RtrZ	21	100 Mbps
C2	RtrA➔RtrL2➔RtrR2➔RtrZ	19	80 Mbps
C3	RtrA➔RtrL3➔RtrM3➔RtrR3➔RtrZ	19	90 Mbps
C4	RtrA➔RtrL4➔RtrR4➔RtrZ	19	90 Mbps
C5	RtrA➔RtrL5➔RtrR5➔RtrZ	19	90 Mbps

Será analisado como o CSPF escolhe qual caminho deve ser seguido. Para isso, será verificado como o processo de decisão ocorre a partir de RtrA com os seguintes passos:

1. C1 não é usado porque é o caminho com maior custo entre todos os caminhos.
2. C2 não é usado porque sua largura de banda mínima é de 80 Mbps, que é inferior às larguras de banda mínimas dos outros caminhos.
3. C3 não é usado porque tem uma contagem de *hops* de 5, enquanto os outros caminhos considerados possuem uma contagem de apenas 4 *hops* .
4. RtrA escolhe C4 ou C5, o que estiver no topo da lista TENT.

3.2.3 Diferenças entre IS-IS e OSPF

Durante a escolha do protocolo de roteamento um fator que influenciará na escolha é técnica de cálculo de caminho e, dessa forma, limita a escolha do protocolo de roteamento utilizado para distribuição de rotas na rede. Além disso, devem ser consideradas as diferenças entre os protocolos. Na tabela 20 são apresentados as diferenças entre o IS-IS e o OSPF.

Tabela 20: Um resumo das diferenças entre OSPF e IS-IS [6].

Recurso	**OSPF**	**IS-IS**
Protocolo de distribuição	Opera sobre IP.	Executa como um protocolo de rede junto ao IP.
Roteia qual protocolo?	Projetado para IPv4. Agora admite o IPv6 na versão do OSPFv3.	Não importa. Pode facilmente ser feitas mudanças para dar suporte a qualquer protocolo, incluindo IPv4 e IPv6.
Dados nos enlaces	Tudo é alinhado em limites de 32 bits, tornando os pacotes um pouco maiores do que precisam ser. LSAs geralmente são muito pequenos, permitindo melhor granularidade nos enlaces e menos dados para refletir uma mudança na rede.	O alinhamento de *bytes* mantém os pacotes pequenos, mas *Link State Packet*s são grandes e disparados em sua totalidade até por mudanças pequenas na topologia.
Atualizações do banco de dados de escalabilidade	LSA são limitados pela idade a uma hora, gerando um alto nível de tráfego de segundo plano em rede grande.	As informações do banco de dados podem durar mais de 18 horas, produzindo uma melhoria significativa.
Tamanho da rede de escalabilidade	Não escala bem em redes grandes. A única opção é dividir a rede em áreas ou sistemas autônomos.	Não escala bem em redes grandes, mas o uso de grupos mesclados pode evitar a necessidade de usar áreas ou sistemas autônomos.
Suporte para áreas	Entende com facilidade a hierarquia de dois níveis, com roteadores da borda da área que se encontram em múltiplas áreas.	Os roteadores são residentes exatamente em uma área com enlaces entre áreas. Isso faz com que as áreas se encaixem melhor ao conceito de sistemas autônomos, mas, na prática, leva ao requisito para roteadores virtuais, de modo que um único roteador físico possa estabelecer uma presença em duas áreas como ABR. Experiência de implantação muito limitada de sistemas IP em múltiplas áreas.
Recursos avançados	Inclui uma grande quantidade de recursos avançados para tratar de requisitos específicos.	O projeto é limitado aos principais requisitos, sem recursos avançados do OSPF.
Simplicidade da implementação	Um protocolo muito complexo quando todas as opções e recursos são considerados. Essa complexidade pode ser refletida nas implementações.	Um protocolo relativamente simples, em parte porque está faltando muitos dos recursos avançados do OSPF. Isso deve levar a implementações mais robustas.

Caso já se tenha decido pelo protocolo de roteamento IGP por estado de enlace, onde, neste caso, têm-se ainda duas opções o OSPF e o IS-IS, esses protocolos possuem seguidores que os defendem a qualquer custo.

Durante o processo de decisão, no caso de um *backbone* que já está funcionando com o IGP por estado de enlace, é bem mais fácil continuar seguindo este nos novos roteadores, mesmo sabendo que a sua troca é possível. Mas para uma rede nova, ou uma rede pequena o suficiente para ser migrada, deverá ser decido qual o protocolo de roteamento de estado de enlace utilizar. Em nível alto, existem apenas algumas diferenças entre esses protocolos, conceitualmente com a mesma informação de estado de enlace sendo distribuída por cada um e as mesmas tabelas de roteamento sendo geradas como resultado da execução do algoritmo de Dijkstra. Os dois protocolos admitem áreas e hierarquias, cada um usa um mecanismo de *Hello* para descobrir e manter um relacionamento com seus vizinhos; tanto o OSPF quanto o IS-IS aceitam o conceito de roteador designado em um enlace de acesso múltiplo; e os dois protocolos são extensíveis para a engenharia de tráfego, pois neste caso existem o OSPF-TE (*Open Shortest Path First Traffic Engineering*) e o IS-IS-TE (*Intermediate System to Intermediante System Traffic Engineering*) [22]. Esses fatos não surpreendem, pois o OSPF foi criado tendo como base o IS-IS, e os dois evoluíram com a necessidade de novos recursos.

De forma expressiva, tanto o OSPF quanto o IS-IS são largamente implantados em grandes redes da Internet de modo bem sucedido. O IS-IS foi desenvolvido primeiro que o OSPF, porém o OSPF teve maior influência nas redes IP, parcialmente porque foi entregue pela *Cisco* um ano antes que o IS-IS aceitasse o IP. Embora a popularidade do OSPF tenha crescido desde então, o IS-IS possui uma maior base implantada e um forte clube de defensores, a ponto de nenhum fabricante de roteador ser considerado sério se não oferecer os dois protocolos em suas linhas de produtos.

O padrão de implementação para os dois protocolos mostra que o IS-IS geralmente é usado pela “camada um” de muitos ISPs, em que o AS inteiro é gerenciado como uma única área. O OSPF é usado na maioria das outras redes que normalmente utilizam múltiplas áreas. Embora seja incomum para um operadora de rede migrar de um protocolo de roteamento para outro, ocorreram alguns casos recentes dos principais ISPs passando do OSPF para o IS-IS.

A escalabilidade continua é um problema para os dois protocolos, particularmente com relação ao grande número de atualizações de banco de dados

geradas quando um enlace ou roteador fica indisponível na rede. Por exemplo, uma rede OSPF totalmente mesclada com *n* roteadores geraria algo na ordem de n^2 mensagens na falha de um enlace e n^3 mensagens quando um roteador entrar em falha. O IS-IS sofre porque os *Link State Packet*s são grandes e precisam ser totalmente atualizados quando existe uma falha. Esse problema aumenta em redes que possuem uma detecção rápida de falha do enlace e é um problema particular em ambientes como IP sobre ATM. Para os dois protocolos a solução é dividir a rede em áreas. Isso é normalmente é feito no OSPF, mas menos comum no IS-IS.

3.2.4 BGP – *Border Gateway Protocol*

Já foi tratado dos protocolos IGP, onde esses fazem o roteamento dentro de um mesmo sistema autônomo e foi descrito sobre a Internet, que é organizada como sistemas autônomos, cada qual sob controle de uma única entidade administrativa. Um Sistema Autônomo (AS) é uma coleção de redes sob uma administração comum.

Se à Internet tivesse que operar como uma única rede e executar uma instância de IGP em toda rede, ela apresentaria problemas. O problema mais significativo seria o tamanho dos bancos de dados de estado de enlace que os roteadores precisariam manter para o cálculo da rota, além da taxa de mudança das informações que ocorressem na rede. Fora isso, os ISPs que cooperam para formar a Internet teriam que compartilhar as informações de roteamento, o que colocaria em exposição toda a topologia de rede aos concorrentes.

Para tanto é desejável tentar segmentar a Internet em domínios separados sob o gerenciamento dos ISPs e limitar as informações passadas entre esses domínios.

As redes gerenciadas pelos ISPs são designadas como sistemas autônomos, e o roteamento é obtido dentro de todos os AS pela execução de um protocolo de roteamento interno. Os IGPs não conhecem a topologia da Internet fora do AS, mas sabem como encaminhar o tráfego para qualquer nó em um sistema autônomo e para os roteadores que se encontram na borda da rede os ASBRs (*Autonomous System Border Routers*). Esses roteadores oferecem conectividade aos sistemas autônomos sob gerenciamento separado.

Dessa forma, como será feito o encaminhamento do tráfego entre os sistemas autônomos diferentes? Será viável configurar essa informação manualmente e injetá-la no IGP executado no AS? Mas, o número de ASs na Internet cresceu muito, as

interconexões entre os ASs são inúmeras, e essa configuração manual seria muito difícil de manter a precisão. A resposta para todas essas perguntas é executar um protocolo de roteamento entre os sistemas autônomos, como, por exemplo, um protocolo EGP. O protocolo EGP, que tem a mesma denominação que o tipo de protocolo a que pertence, foi o primeiro protocolo interdomínios, tendo sido posteriormente suplantando pelo BGP (*Border Gateway Protocol*).

O protocolo BGP-4, definido pela RFC 4271, é um protocolo com métrica vetor caminho que se tornou predominante no IP e no MPLS. Como protocolo BGP-4 destina-se especificamente ao IPv4, o IETF emitiu a RFC 4760 com objetivo de estender a aplicabilidade do BGP-4 para os demais protocolos de camada de rede, extensão essa que passou a denominar MP-BGP (*Multiprotocol BGP*).

Essa extensão é utilizada para a implementação de VPN-MPLS. O MP-BGP possui as mesmas características de funcionamento do BGP tradicional, mas é capaz de anunciar não só as rotas (prefixos) IPv4, mas também prefixos VPNv4. Prefixos VPNv4 são prefixos IP que pertencem a uma VPN específica.

3.3 Considerações Finais

Neste capítulo foram abordadas as diferenças entre roteamento e comutação, já que esta dissertação trata do protocolo MPLS o qual está inserido entre a camada 2 e a camada 3 do modelo OSI e esse protocolo utiliza características de comutação e roteamento. Foram feitas revisão e análise comparativa dos protocolos de roteamento dinâmico, além de ressaltar as principais características e modificações dos protocolos de roteamento visando o atendimento do protocolo IP e aplicabilidade para o MPLS-TE que será o foco principal desta dissertação. Este capítulo é finalizado com os protocolos de roteamento mais utilizados na Internet.

Essa revisão foi necessária, pois serão utilizados os protocolos OSPF e o BGP para simulações e comprovações dos testes executados nesta dissertação.

O próximo capítulo descreve o protocolo MPLS (*Multiprotocol Label Switching*), que modifica o mecanismo de encaminhamento dos pacotes IP, e como este se utiliza desse protocolo para transporte de pacotes.

Capítulo 4- A Tecnologia MPLS-IP

Antes de iniciar o estudo sobre o MPLS, deve-se estudar o seu funcionamento e para isso é necessário iniciar respondendo a uma questão simples "Para que serve o MPLS?". Pode-se elencar respostas a essa pergunta, mas existem três principais pontos para os quais é possível usar o MPLS, que são:

- para permitir capacidade IP em dispositivos que não têm a capacidade de encaminhar datagramas IP pelo modo normal;
- para encaminhar pacotes IP ao longo de "rotas explícitas" – rotas pré-calculadas que não necessariamente combinam com aquelas que o os protocolos de roteamento IP normais selecionariam;
- para admitir certos tipos de serviços de rede privada virtual.

Uma observação que deve ser feita é que nestes principais pontos não consta o principal objetivo original de melhorar o desempenho. Essa exclusão está relacionada com os avanços que foram feitos nos protocolos de roteamento nos últimos anos, e com o conjunto complexo de fatores, além do processamento do cabeçalho, que determina o desempenho.

É necessário algum processo para preencher as tabelas de consulta em cada nó MPLS. Além da óbvia possibilidade de configuração manual, vários protocolos foram desenvolvidos, ou adaptados a partir de seus usos iniciais, para distribuir rótulos e construir tabelas de consulta. Alguns desses protocolos foram adaptados de seus usos iniciais, como por exemplo, o MP-BGP (*Multi Protocol Border Gateway Protocol)*, e o RSVP-TE (*Resource Reservation Protocol – Traffic Engineering*), conforme a RFC 3209, enquanto que novos protocolos foram desenvolvidos. Esses protocolos foram desenvolvidos pelo IETF para fornecer a família completa de protocolo de roteamento e sinalização para o MPLS. Em todos os casos, os protocolos utilizam o IP, mas isso não tem relação com o fato de os dados comutados por meio de rede MPLS serem ou não tráfego IP.

O MPLS é uma tecnologia aberta, que foi apresentada inicialmente como uma solução que possibilitava melhorar o desempenho das redes IPs na função de encaminhamento de pacotes IPs, combinando o processo de roteamento de nível 3 com a comutação de nível 2 para realizar o encaminhamento de datagramas através de

pequenos rótulos de tamanho fixo. Tais rótulos são números utilizados no protocolo MPLS e, baseados nesses, é tomada a decisão por qual interface deve ser encaminhado o datagrama [32]. A comutação de rótulos combina a funcionalidade dos protocolos de roteamento da camada de rede com a velocidade de comutação dos equipamentos da camada de enlace, além de fornecer significativos benefícios às redes com IP, ou uma combinação de outras tecnologias no nível da camada de rede. Portanto, em uma arquitetura IP sobre MPLS as informações necessárias para o encaminhamento são obtidas do cabeçalho MPLS (32 *bits*), que é bem menor e menos complexo que o cabeçalho IP (20 *bytes*), contribuindo para que os equipamentos de menor poder de processamento e armazenamento tenham desempenho melhor nesse tipo de arquitetura em relação a outras.

Outra vantagem significativa da arquitetura IP sobre MPLS que pode ser destacada, diz respeito ao encaminhamento de datagramas ao longo de um caminho. Em redes IPs convencionais, todos os roteadores da topologia precisam saber a melhor rota em sua tabela de roteamento para encaminhar o pacote ao seu destino pelo melhor caminho possível. Já o protocolo MPLS trabalha com encaminhamento dos pacotes baseado em rótulos, pois os roteadores de núcleo, conhecidos como P (*Provider*), não têm acesso ao endereço IP de destino do pacote, assim não há inteligência de roteamento nesses roteadores de núcleo, e sim o encaminhamento local, de uma interface para outra, tomando como base os valores dos rótulos dos pacotes, ou seja, fazendo um processo apenas de comutação de rótulos.

Uma das primeiras publicações a introduzir o conceito de anexar rótulos aos pacotes IP foi um artigo de Chandranmenon e Varghese [4][63], que descrevia o conceito "índices em *thread*". Essa solução é semelhante a implementada nos roteadores que utilizam MPLS.

4.1 Surgimento da Tecnologia – Histórico e Evolução

O MPLS (*Multiprotocol Label Swiching*) tem suas raízes em várias tecnologias de comutação de pacotes IP que estavam em desenvolvimento no início e meados da década de 1990. Algumas empresas começaram a desenvolver tecnologias baseadas na utilização de rótulos, das quais se destacam:

- *Ipsilon* – criou o *IP Switching*, em que o *hardware* ATM era programado com instruções de comutação de acordo com os caminhos de encaminhamento necessários distribuídos usando o IFMP (*Ipsilon Flow Management Protocol*) [24];
- *Toshiba* – desenvolveu uma das primeiras tecnologias de comutação controladas por IP quando introduziu o *Cell Switched Router;*
- *Cisco* – a comutação de Tag, proposta não era limitada aos comutadores ATM, mas era um método mais generalizado de comutação de rótulos no qual as informações de rótulo eram distribuídas usando o *Tag Switching Architecture* [25];
- *ARIS-Aggregate Route-based IP Switching (IBM), SITA-Switching IP Through ATM (Telecom Finland*) e *IP Navigator (Ascend)* [26].

Destaca-se que devido à incapacidade de interação entre essas tecnologias desenvolvidas em 1996 a IETF começou a organizar as tendências e, em 1997, o MPLS *Working Group* foi constituído para padronização e abordagens para o MPLS. O MPLS é uma tecnologia desenvolvida no âmbito do IETF [26], inicialmente como uma tentativa de padronizar a comutação de pacotes baseada na troca de rótulos e, com isso, melhorar a eficiência de fluxos de tráfegos através da rede, modificando um paradigma fundamental até então existente nas redes IPs, com a inserção de um rótulo ao datagrama, propiciando assim, a comutação IP.

Deve ser visto que o MPLS é de interesse do IETF, não por que o tráfego IP pode ser comutado por meio de uma rede MPLS, mas pelo fato de que os protocolos de controle são baseados na tecnologia IP. Para o IETF, os meios de transportar o tráfego IP são de interesse superficial, devendo esses ser tratados por outros organismos.

Mais recentemente, a idéia de executar protocolos de controle IP em dispositivos que são incapazes de encaminhar pacotes IP em forma nativa foi estendida para *switches* ópticos e dispositivos TDM (*Time Division Multiplexing*), como os multiplexadores SDH/SONET, DWDM (*Dense Wavelength Division Multiplexing*). Isso é conhecido como GMPLS (*Generalized Multiprotocol Label Swithing*) [9], que se encontra já especificado pelo IETF, e abre uma nova perspectiva em rede ao

proporcionar a possibilidade da aplicação de processos de sinalização para o provimento dinâmico de conexões em redes constituídas por esses multiplexadores.

Existe uma grande expectativa quanto ao desenvolvimento de novas tecnologias de rede de transporte operando diretamente no modo de pacotes. A tecnologia MPLS foi considerada apropriada para esse fim, mas ainda têm carências de operação, de administração e de manutenção adequada para tanto. Ocorreram então duas iniciativas paralelas com o propósito de eliminar essas carências, respectivamente denominadas T-MPLS (*Transport MPLS*), segundo recomendação Y.1710 do ITU-T (*International Telecommunication Union – Telecommunication*) [28] que complementa a RFC 4377, e o PBT (*Provider Backbone Transport*), conforme recomendação do IEEE 802.1 ah [29].

Essas iniciativas apresentam problemas, porém proporcionam o entendimento entre o ITU-T e o IETF no sentido de concentrar esforços sob uma concepção global referida como MPLS-TP (*MPLS Transport Profile*) [9].

O IP MPLS Fórum vem concentrando esforços no sentido de definir o modo de utilização do MPLS no segmento *backhaul* das redes de acesso via rádio que suportam a comunicação para serviços móvel celular de terceira geração.

4.2 Roteamento Convencional x Baseado em Rótulos

O MPLS define formas de comutar dados por meio de uma rede consultando um pequeno rótulo ou etiqueta do pacote de dados. Cada nó extrai o rótulo do pacote, consultando uma tabela para determinar o próximo salto para o qual enviará o pacote e colocará um novo rótulo no mesmo. Esse processo não precisa estar ciente do protocolo contido nos pacotes e também não se preocupa com o mecanismo de transporte básico em cada salto. Pode-se fazer uma lista de alguns dos maiores problemas do roteamento tradicional e como eles podem ser solucionados com o MPLS.

- Escalabilidade administrativa: um problema frequentemente agravado com o roteamento IP é o grande tamanho das tabelas de roteamento que precisam ser suportadas, em especial nas redes de núcleo. Ao mesmo tempo, mudanças na topologia de rede causam atualizações de roteamento que exigem recálculo das rotas em todos os roteadores participantes. O MPLS provê algum auxílio como relação a isso, como as redes de borda não estão preocupadas com a rota tomada por meio da rede de núcleo, que por sua vez, usa o MPLS para prover conexões borda-a-borda por meio de VPWSs (*Virtual Private Wire Services*).

- Flexibilidade de roteamento: nem sempre é desejável que todos os dados para um mesmo destino sejam roteados pelo mesmo caminho. Por exemplo, dados de baixa prioridade podem ser encaminhados por um caminho mais longo para manter o caminho mais curto livre para tráfego de prioridade mais alta. A maioria das técnicas de roteamento costuma convergir o tráfego para o caminho mais curto ou usar tabelas para separar pacotes através de marcação dos pacotes IP usando o campo ToS ou a técnica de DiffServ (*Differentiated Services*) para classificar os pacotes. O MPLS possui um campo em seu cabeçalho preparado para atender requisitos de QoS de forma a absorver demandas de tráfego com prioridade. Ainda se pode implementar o MPLS-TE de modo que sejam criados túneis por afinidade de tráfego.

- Integração com tecnologias estabelecidas: a ATM *(Asynchronous Transfer Mode)* e a *Frame Relay* são tecnologias de comutação de rótulos. Nenhuma é baseada em IP, mas ambas são extensivamente empregadas. Um método formalizado para a comutação de rótulos permite que o tráfego IP seja transportado por redes ATM e *Frame Relay* sem as funções de descoberta ou de mapeamento complexo. A teoria aqui é que realizar engenharia de tráfego com MPLS pode ser tão eficaz quanto com o ATM, mas sem todas as desvantagens do IP sobre ATM.

- Escalabilidade: quando todos os nós na rede são comutadores de dados, eles também podem receber capacidades de roteamento IP. Isso não significa que roteiam pacotes de dados, pois esses são comutadores. Entretanto, os comutadores podem usar roteamento para trocar informações de controle e para estabelecer circuitos comutados ao longo dos quais os dados são encaminhados. Tornar todos os nós na rede capazes de realizar roteamento IP reduz os requisitos de conectividade das bordas: em vez de cada nó de borda precisar manter a conectividade com cada um dos nós de borda, ele simplesmente manterá a conectividade com seu roteador fisicamente adjacente.

- Serviços adicionais: alguns dos problemas tradicionais associados com VPNs (*Virtual Private Networks*) são resolvidos usando a comutação de rótulos. Onde os endereços IP dos espaços de endereços privados precisam ser mapeados ou ocultados da rede IP de núcleo, os rótulos podem ser superpostos e usados por meio de toda a rede MPLS.

- Extensibilidade para novas tecnologias: embora não tenha sido parte da motivação inicial para o MPLS, o processo de rotular fluxos de dados se tornou um conceito útil nas novas tecnologias, oferecendo a possibilidade de provisão dinâmica em redes ópticas e TDM (*Time Division Multiplexing*).

Algumas das motivações para a comutação de rótulos foram apresentadas. Normalmente nenhum único problema é considerado uma razão absoluta para escolher uma tecnologia de comutação, mas a combinação da engenharia de tráfego com a facilidade de provisão de serviços de VPN está incentivando as operadoras de telecomunicações e as demais empresas a migrarem suas redes para usar comutação de rótulos.

Dentre um dos fatores que motivou a aplicação dessa tecnologia está relacionado o crescimento da Internet no mundo [33]. A demanda de tráfego requerida pelos provedores de acesso a Internet aumentou bastante e para suportar esse crescimento as operadoras de telecomunicações precisam de roteadores de alto desempenho, pois além da crescente demanda por banda, os mesmos precisam lidar com o crescente número de nós na rede e, consequentemente, um aumento nas tabelas de roteamento.

Na tabela 21 é possível visualizar a comparação entre a arquitetura IP convencional e a arquitetura IP baseada em rótulos:

Tabela 21: Arquitetura IP Convencional e Arquitetura IP baseada em rótulos.

	Roteamento IP Convencional	**Roteamento baseado em rótulos**
Análise de todo cabeçalho IP	Verificação dos pacotes a cada salto em todo caminho na rede	Verificação dos pacotes apenas uma vez no ingresso do caminho virtual.
Suporte para dados *Unicast* e *Multicast*	Necessita de roteamento especial para *Multicast* e algoritmos de encaminhamento	Necessita de somente um algoritmo de encaminhamento
Decisão de roteamento	Baseado no endereço de destino no cabeçalho do pacote IP.	Baseado em vários parâmetros: endereço de destino no cabeçalho IP, QoS, tipo de dados, etc.

4.3 O Cabeçalho MPLS

Um pacote IP ao entrar em rede MPLS está sem rótulo. Depois que o pacote entra no primeiro roteador de ingresso na rede MPLS, esse faz o mapeamento acrescentando um rótulo necessário para comutação no *backbone*.

A comutação de rótulos consiste em associar um rótulo curto de tamanho fixo de 4 *bytes* em cada pacote de dados, de modo que ele possa ser encaminhado na rede. Esse

é o item mais importante para o MPLS [34]. Isso significa que cada pacote, quadro ou célula precisa transportar algum identificador que informe aos nós de rede como encaminhá-lo. Em cada salto por meio da rede, o pacote é encaminhado com base no valor do rótulo que chega.

Em uma rede MPLS, os pacotes são rotulados pela inserção de uma informação adicional chamada *shim header*. Esse cabeçalho é conhecido como *shim* ou cabeçalho de calço. Na figura 17 é apresentado o cabeçalho de calço inserido entre o cabeçalho nível 2 e o cabeçalho IP.

Cabeçalho Nível 2	Cabeçalho de calço	Cabeçalho IP	Dados

Figura 17: O cabeçalho de calço inserido entre o cabeçalho nível 2 e o cabeçalho IP.

Quanto às formas de codificação de rótulos, têm-se:

1. rótulos codificados no interior de um cabeçalho de calço, que são adicionados aos pacotes sem rótulos para construir os pacotes MPLS, essa forma é conhecida com rótulos genéricos;
2. rótulos codificados em campos já existentes nos protocolos de camada de rede que suportam o MPLS, que são os rótulos ATM e os rótulos *Frame Relay*.

O cabeçalho de calço é utilizado no caso do MPLS sobre redes sem conexão, principalmente rede *Ethernet*, ou de uso do protocolo PPP em redes MPLS com suporte em roteadores IP.

Dois métodos comuns de transportar rótulos em pacotes aparecem na figura 18. Quando os pacotes IP são transportados como quadros completos, como na maioria dos tipos de enlace, incluindo o *Ethernet* e PPP, o rótulo é inserido como um "calço" entre o cabeçalho da camada 2 e o cabeçalho IP conforme figura 18 (b).

Isso poderia ser usado, por exemplo, onde os LSRs (*Label Switching Routers*) são conectados pelos enlaces *Ethernet.* Se a conectividade de rede é fornecida por uma tecnologia de comutação, como ATM ou *Frame Relay*, o rótulo MPLS é transportado nos campos VPI/VCI (*Vitual Patch Identifer*)/(*Virtual Channel Identifier*) do ATM conforme apresentado na figura 18 (a) ou no DLCI (*Data Link Connection Identifier*) do protocolo *Frame Relay*. Nesse caso existem situações em que são utilizados cabeçalho de calço que complementam a existência de rótulos codificados nesses campos.

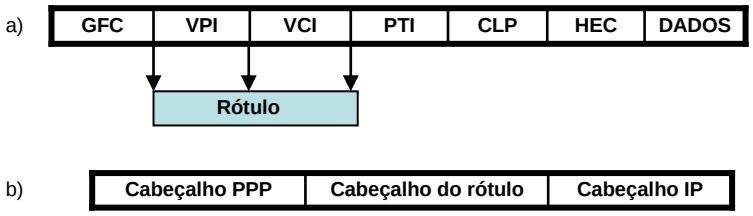


Figura 18: (a) Rótulo em um pacote encapsulado por ATM
(b) Rótulo em um pacote encapsulado no quadro.

O cabeçalho de calço transporta um rótulo de 20 *bits*, conforme apresentado na figura 19.

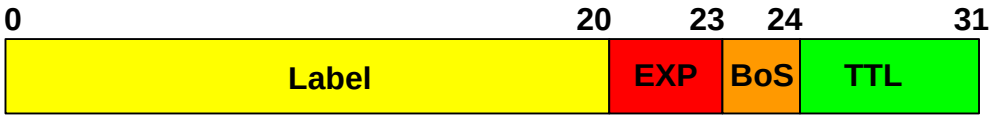


Figura 19: O cabeçalho do MPLS [34].

O cabeçalho do MPLS e a função de cada campo, vistos na figura 19, são descritos a seguir:

***Label* (Rótulo):** Contém o valor do rótulo MPLS. Como o tamanho é de 20 *bits*, esse valor pode variar de 0 a 2^{20} -1, ou 1.048,575. Existem alguns valores que são reservados ao protocolo, e têm significados especiais [35]:

0 – IPv4 *Explicit NULL Label*: indica que o rótulo deve ser retirado, e desse ponto em diante, o roteamento será feito com base no endereço de rede.

1 – *Router Alert Label:* indica que o datagrama deve ser analisado pelo *software* local. O encaminhamento seguinte é definido pelo próximo rótulo da pilha MPLS.

2 – *IPv6 Explicit NULL Label*: mesma funcionalidade do valor 0, mas aplicada ao protocolo IPv6.

3 – *Implicit* NULL *Label*: valor utilizado pelos LSRs (*Label Switch Routers*) para a distribuição de rótulos (LDP- *Label Distribution Protocol*).

4 a 15 – Reservados para definições futuras.

EXP (*Experimental bits*): este campo é composto por três *bits* e são utilizados para alterar os algoritmos de enfileiramento (*queuing*) e descarte. Dessa forma, é possível dar prioridade a determinados pacotes. Esses *bits* permitem que oito classes de serviço sejam definidas dentro de um LSP (*Label Switched Path*), que é o caminho definido pelo roteador de entrada no *backbone* MPLS. O valor desses *bits* pode ser usado para indicar a prioridade relativa no processamento e no uso de recursos. Usado atualmente por classes de serviços (CoS).

BoS (*Bottom of Stack*): Formado por apenas um *bit*, esse campo permite a criação de uma pilha hierárquica de rótulos. Indica se o cabeçalho, ao qual o pacote pertence, é o último da pilha MPLS. Todos os cabeçalhos MPLS devem ter esse *bit* em 0, e através desse campo um roteador de saída tem condições de decidir se o próximo encaminhamento será baseado em MPLS ou IP.

TTL (*Time to live*): Esse campo é formado por 8 *bits* e funciona de maneira semelhante ao TTL do protocolo IP já explicado no capítulo 2 desta dissertação. O TTL é copiado do cabeçalho IP quando um pacote é rotulado e é tratado da mesma maneira, à medida que o pacote rotulado é encaminhado pela rede.

Esse rótulo tem significado local no roteador que é usado para identificar uma FEC (*Forwarding Equivalent Class*), isto é, um grupo de pacotes IPs que são enviados na mesma maneira, sobre o mesmo trajeto e com o mesmo tratamento de transmissão. Uma FEC pode corresponder a uma *subnet* do endereço IP de destino, mas igualmente pode corresponder a qualquer classe de tráfego que o roteador de borda considera significativo, como exemplo, todo o tráfego com o mesmo valor de "*IP Precedence*" pode constituir uma FEC. Dessa forma, para que um datagrama IP destinado a uma FEC, que ingressa em uma rede MPLS, possa trafegar corretamente por um LSP (*Label Switched Path*) até seu roteador de egresso e então ser transmitido, é necessário que a rede MPLS passe por uma fase preparatória de controle.

Como o mapeamento em cada roteador é constante, o caminho é determinado de maneira única pelo valor do rótulo no primeiro roteador. Esse caminho é chamado LSP (*Label Switched Path*). Cada roteador na rede MPLS mantém uma tabela de consulta que permite determinar o próximo saldo no LSP. Esta tabela é conhecida como LFIB (*Label Forwarding Information Base*) ela faz o mapeamento da interface de entrada, rótulo de entrada, para interface de saída, rótulo de saída, respectivamente. Os roteadores em *backbone* MPLS só precisam dessa informação para encaminhar os pacotes rotulados.

4.4 Plano de Controle e Dados do MPLS

Como visto anteriormente, o cabeçalho MPLS pode ser encapsulado em diversos protocolos de nível 2 e encapsular qualquer protocolo de nível 3. Existem muitos debates sobre o local onde o MPLS pertence nas arquiteturas de protocolos em camadas.

Como o cabeçalho MPLS é normalmente encontrado entre os cabeçalhos da camada 3 e da camada 2 em um pacote, ele às vezes é considerado como protocolo da camada 2,5. Alguns autores [33] argumentam que, como os pacotes IP são encapsulados dentro do cabeçalho MPLS, esse deve estar "abaixo" do IP, tornando-o um protocolo da camada 2. Segundo Harnedy [33], o MPLS pode ser considerado um protocolo de camada 2.5. Outros [36] argumentam que os protocolos de controle para o MPLS são, em grande parte, os mesmo protocolos do IP, pois o MPLS utiliza os protocolos de roteamento e deve estar na mesma camada do IP. Assim, devido a camada 2,5 ser uma camada de integração, é necessário que a mesma seja compatível com diversos protocolos da camada 3, assim como, com as tecnologias de camada 2, o que justifica o nome de "*Multi-Protocol*" do MPLS.

As arquiteturas em camadas são ferramentas úteis, mas nem sempre podem descrever exatamente o mundo real, e o MPLS é bom exemplo de que as visões estritamente em camadas podem ser difíceis de conciliar com a realidade.

No MPLS é possível separar o plano de controle (*Control Plane*) do plano de dados (*Data Plane*), conforme apresentado na figura 20:

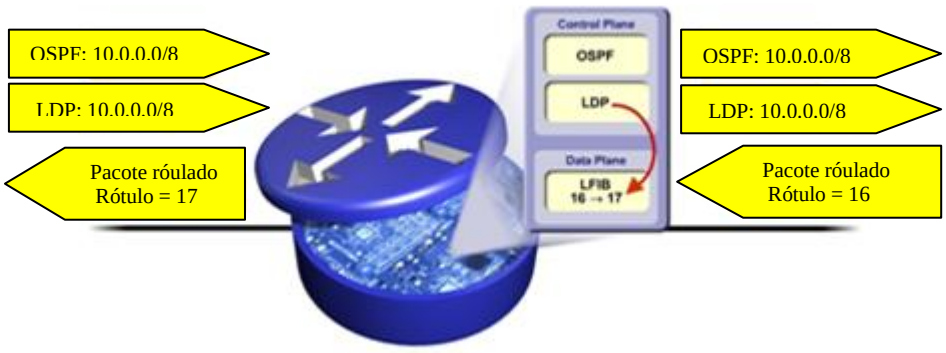


Figura 20: Plano de Controle e Plano de Dados [7].

Essa é uma característica forte do MPLS, pois caso se deseje mudar a estratégia de roteamento na rede, não será necessário mudar os dispositivos de encaminhamento. No plano de controle são trocadas as informações de roteamento e rótulos. Neste contém os protocolos de roteamento e o protocolo de distribuição de rótulos LDP (*Label Distribution Protocol*). Ele contém os mecanismos complexos para troca de informações de roteamento e troca de rótulos. Esse plano precisa reagir quando houver mudanças na rede, mas não é envolvido no processamento individual dos pacotes. No

plano de controle é construída e mantida a tabela de encaminhamento do nó em uso. É nesse plano que estão localizadas as funções de controle, tais como sinalização, roteamento, policiamento de tráfego, conversão de endereços, dentre outras. Os protocolos de roteamento: RIP, OSPF, IS-IS e BGP, atuam nesse plano, e são usados para trocar informações de roteamento entre os componentes de controle, conforme pode ser observado na figura 21:

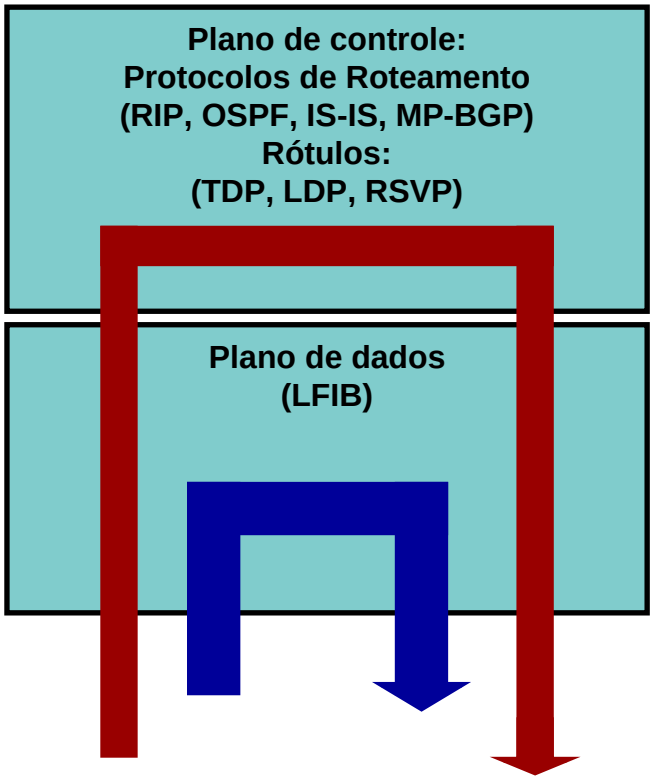


Figura 21: Plano de Controle e Plano de Dados [36].

O plano de dados, que tem sua operação ditada pelo plano de controle, é responsável pelo encaminhamento do tráfego baseado apenas em rótulos. Neste plano é mantido o conteúdo da tabela de comutação de rótulos, que é conhecida como LFIB (*Label Forwading Information Base*). Observe que, uma vez que a LFIB tenha sido preenchida, não há liberdade de escolha dos valores de rótulos. Em geral, o que os roteadores da rede MPLS fazem é utilizar o plano de controle para, através de um protocolo de roteamento, descobrir e escolher os melhores caminhos até o destino e acrescentar na tabela do plano de dados.

4.5 Elementos da Arquitetura MPLS

Uma visão geral dos equipamentos de uma rede MPLS é apresentada na figura 22:

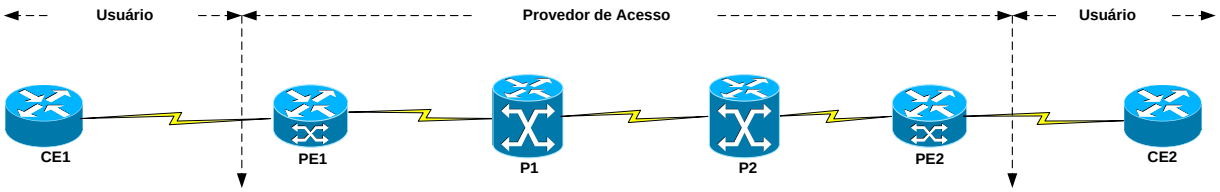


Figura 22: Componentes de uma rede MPLS [9].

Na figura 22, tem-se os roteadores CEs (*Customer Edges*), os roteadores PEs (*Provider Edges*) e os roteadores Ps (*Providers*).

Os roteadores Ps fazem apenas a comutação de rótulos, esses estão internos no *backbone* MPLS.

Os roteadores PEs estão na borda do backbone fazendo a fronteira entre a rede do cliente e da operadora de telecomunicações, esses fazem roteamento e comutação.

Os roteadores CE não fazem parte do domínio MPLS, mas tem sua nomenclatura na rede MPLS, pois estão integrando a rede do cliente à rede da operadora de telecomunicações. Tais elementos não têm conhecimento da tecnologia MPLS, e o tráfego gerado pelos mesmos é baseado apenas nos protocolos roteáveis, dos quais se destaca o protocolo IP.

A seguir serão detalhados a função de cada equipamento na rede MPLS e como são estabelecidos os LSPs, a função das tabelas LIB, FIB, LFIB e FEC, para isso será usada a figura 23:

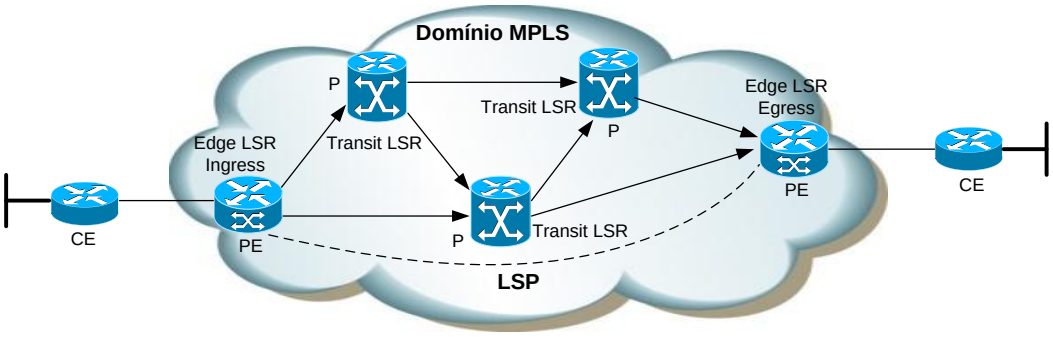


Figura 23: Componentes da Arquitetura MPLS em Detalhes [34].

• LSR (*Label Switch Router*): é um equipamento capaz de realizar encaminhamento de datagramas de rede através de rótulos MPLS. Uma das principais

funções do primeiro LSR em uma rede MPLS é determinar com qual LSP (*Label Switched Path*) associar os pacotes de dados. Sua participação é ativa no estabelecimento de um LSP, usando protocolo de sinalização de rótulo, tais como: LDP, RSVP-TE (*Resource Reservation Protocol – Traffic Engineering*) e BGP, e no encaminhamento de tráfego baseado nos caminhos estabelecidos. Dois LSRs que usam LDP para anunciar rótulos uns aos outros são chamados de LDP *peers*. Um LSR se apresenta aos parceiros difundindo uma mensagem *Hello* LDP usando UDP. Ele envia periodicamente ao endereço de grupo "todos os roteadores nesta sub-rede" e visa a uma porta UDP bem conhecida reservada para descoberta LDP. Existem basicamente 03 tipos de LSRs:

- *Ingress* LSR: É responsável por impor um rótulo de modo que sigam o caminho correto por meio da rede. Ele pode ser um roteador ou *switch* com funções de roteamento de entrada de uma rede MPLS. O mesmo realiza o processamento e classificação inicial do pacote e aplica o primeiro rótulo na entrada (*ingress*) do pacote no domínio MPLS. Os *ingress* LSRs analisam as informações do cabeçalho de rede e associam cada datagrama a uma FEC (*Forwarding Equivalence Class*). Toda FEC tem um rótulo associado que será utilizado no encaminhamento para o próximo nó.

- *Transit* LSR: São LSRs intermediários que têm a função de apenas fazer a comutação, ou seja, a troca de rótulos, e encaminhar o datagrama para o próximo nó. Os mesmos oferecem comutação em alta velocidade, sendo este um processo que mais contribui para o ganho de desempenho na utilização do protocolo MPLS, já que os mesmos não precisam mais analisar cabeçalhos da camada de rede (IP).

- *Egress* LSR: É um LSR responsável pela retirada do rótulo do pacote e encaminhamento ao seu destino final.

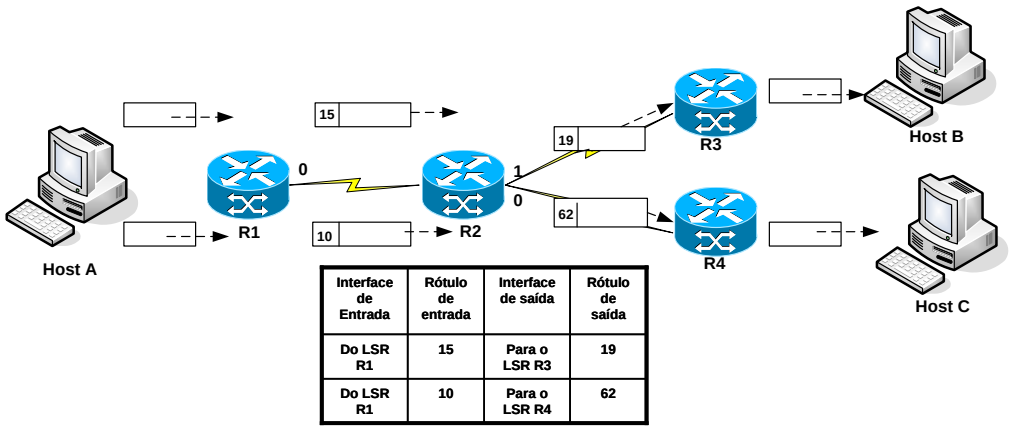
• LSP (*Label Switched Path*): Conforme visto na figura 23, é uma sequência ordenada de LSRs, sendo o primeiro um LSR de ingresso e o último um LSR de egresso, ou seja, é o caminho entre o nó de entrada, possíveis nós intermediários, e o nó de saída de uma rede MPLS. É similar a um circuito virtual ATM ou *Frame-Relay*, sendo unidirecional no sentido da origem para o destino, podendo ser estático ou dinâmico.

• LIB (*Label* Information *Base*): É uma tabela que contém os diversos vínculos de rótulos, que um LSR (*Label Switch Router*) recebe sobre o protocolo LDP, ou seja, uma tabela que apresenta informações correlacionando os rótulos às interfaces do roteador. É através desta tabela que o LSR determina para qual interface deverá

encaminhar o pacote recebido. Essas bases de informações nos LSRs contém os *bindings* distantes entre rótulos e FECs. As informações contidas na LIB são usadas para propagar informação utilizada pelo plano de dados para fornecer a funcionalidade ao MPLS. No plano de dados, os rótulos são atribuídos pelos roteadores *next-hop* e adicionada à tabela FIB (*Forwarding Information Base*). A tabela LIB pode ser visualizada nas figuras 27 e 28.

- FIB (*Forwarding Information Base*): É uma base de informação contida em um LSR, que resulta normalmente dos processos de roteamento IP, que associam as FECs aos endereços IP dos *next-hops* e às respectivas interfaces de saída. Para todo possível endereço IP de destino, uma pesquisa de prefixo longo é executada pela FIB. Se um endereço é localizado na tabela, o roteador saberá para qual interface de saída deverá enviar o pacote e, se nenhum endereço é localizado, o pacote é descartado. Os conteúdos da FIB refletem o estado atual da topologia IP que cerca o roteador, como determinado pelos protocolos de roteamento, por exemplo, OSPF (*Open Shortest Path First*) ou BGP4 (*Border Gateway Protocol version 4*). O processo de roteamento pode ser também do tipo *constraint-based routing*. Foi feita uma pequena introdução na sessão 3.2 (roteamento e comutação) sobre essa tabela.

- LFIB (*Label Forwarding Information Base*): É uma tabela que indica onde e como encaminhar os pacotes. É criada por equipamentos pertencentes a um domínio MPLS. A LFIB contém uma lista de entradas que consistem em uma subentrada e ingresso e uma ou mais subentradas de egresso, de rótulo de saída, de interface de saída, de componentes de saída de nível de enlace. É baseada nas informações obtidas pelo LSR através da interação com os protocolos de roteamento. Em geral, a escolha dos valores de rótulos a serem colocados na LFIB é controlada apenas pela consideração de que os rótulos já estão em uso e as capacidades do *hardware/software* estarão comutando pacotes com base nos valores de rótulo. Será examinado o funcionamento da figura 24, para isso foi ilustrado dois LSPs transportando dados do *Host* A para o *Host* B e C. A rede MPLS é composta por quatro LSRs (*Label Switching Routers*) que encaminham os pacotes. O *Host* A envia pacotes IP para o LSR R1 usando a rota padrão. O LSR R1 é um LSR de ingresso e classifica os pacotes com base no destino final, os atribui a um LSP e os rotula. Os pacotes com destino ao *Host* B são atribuídos ao LSP superior e são rotulados com 15; os pacotes para *Host* C são atribuídos ao LSP inferior e são rotulados com 10. Uma vez rotulados, os pacotes são encaminhados da interface apropriada em direção ao LSR R2.



Interface de Entrada	Rótulo de entrada	Interface de saída	Rótulo de saída
Do LSR R1	15	Para o LSR R3	19
Do LSR R1	10	Para o LSR R4	62

Figura 24: Caminhos comutados por rótulos.

No LSR R2, cada pacote rotulado é examinado para determinar a interface de entrada e o rótulo de saída. Esses são consultados na LFIB conforme apresentado na figura 24. O LSR R3 e o LSR R4 são LSRs de egresso. Eles também realizam uma consulta em suas LFIBs, mas as entradas indicam que devem remover o cabeçalho de calço.

- FEC (*Forwarding Equivalence Class*): É um conjunto de todos os pacotes que são encaminhados da mesma maneira. Uma FEC é descrita pelos parâmetros usados para identificar os pacotes que a constituem. Como exemplo, um grupo de pacotes cujos endereços de origem e destino são os mesmos. Em aplicações mais avançadas, a identidade de uma FEC pode incluir a classe de serviços dos pacotes, ou seja, um mesmo fluxo de dados geralmente pertence à mesma FEC. Uma FEC é representada por um rótulo e cada LSP (*Label Switch Path*) é associado a uma FEC. Com essa interpretação, a função do LSR de ingresso é identificar a FEC a qual cada pacote pertence e usar isso para determinar o LSP e o primeiro rótulo, de modo que ele possa encaminhar o pacote por meio da rede MPLS. Portanto, existe uma associação pacote-rótulo-FEC-LSP, e essa associação pacote-FEC ocorre apenas quando o pacote entra na rede MPLS, proporcionando grande flexibilidade e escalabilidade a este tipo de rede. Da mesma forma como as rotas IP convergem para o seu destino, os caminhos tomados pelos pacotes MPLS podem convergir. É possível manter LSPs diferentes de ingresso para egresso, mas isso não é necessário e pode ser um desperdício de recursos de rede e capacidade de processamento. Em vez disso, os LSPs podem se fundir nos nós de trânsito para que os pacotes do mesmo FEC sejam agrupados no mesmo LSP. Na figura 25 é apresentada essa idéia, onde os *host* A e *host* B estão enviando pacotes para o *host*

C. Os LSRs de ingresso R1 e R2 rotulam os pacotes e os encaminham para o LSR R3, então, os dois são mapeados para o mesmo rótulo.

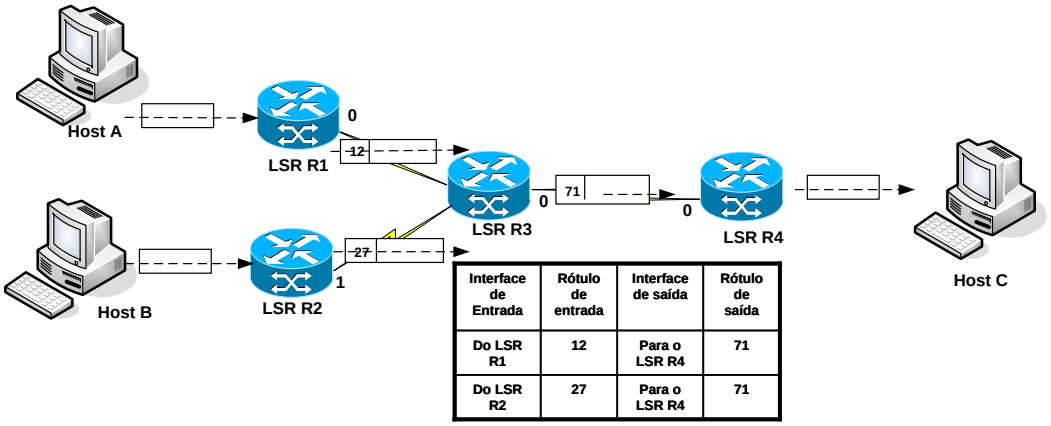


Figura 25: Fusão de rótulos em um mesmo LSP.

O primeiro passo para operacionalização de redes MPLS é a determinação de FECs pela supervisão da rede, por meio da adoção de critérios para tanto. A cada uma das FECs definidas é atribuído um valor de identificação próprio.

Para entender a função dos elementos de uma rede MPLS, inicialmente se deve entender como são distribuídos os rótulos nesta rede pelo protocolo LDP (*Label Distribution Protocol*). Esse protocolo foi desenvolvido pelo MPLS *Working Group* do IETF no final da década de 1990, esboçando idéias do TDP (*Tag Distribution Protocol*) da *Cisco*. Os rótulos são gerados localmente por cada dispositivo MPLS e atribuidos a redes IP de destino. Uma vez que um novo rótulo é criado ou removido, esse dispositivo MPLS deve anunciar esta informação para seus vizinhos diretamente conectados através do protocolo LDP ou TDP. Esse protocolo é o responsável pela distribuição de rótulos para os prefixos IPs em uma rede MPLS, podendo ser utilizado para tráfego correspondente a uma FEC. Os rótulos são assinalados para cada prefixo IGP aprendido na tabela de rotas global de um roteador. O anúncio é transportado no LDP conforme ilustrado na figura 26, que apresenta uma rede com quatro roteadores R1, R2, R3 e R4. Os roteadores R3 e R4 são conectados na LAN com os prefixos 10.1.1/24 e 10.3.3./24.

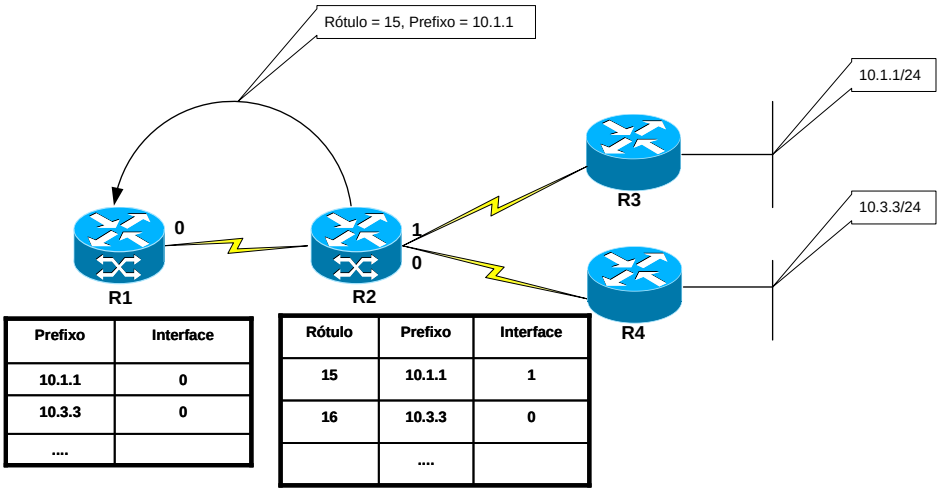


Figura 26: R2 aloca rótulos e anuncia ligações com R1.

Os roteadores R1 e R2 possuem tabelas de roteamento que indicam qual a interface de saída que cada roteador usaria ao encaminhar pacotes para uma dessas duas redes. O roteador R2 alocou o valor de rótulo 15 para o prefixo 10.1.1 e valor de rótulo 16 para o prefixo 10.3.3. Esses rótulos podem ser escolhidos por conveniência do roteador que os aloca e podem ser considerados como índices para a tabela de roteamento. Depois de alocar os rótulos, R2 anuncia as ligações de rótulos com seus vizinhos; neste caso, o R2 anuncia uma ligação entre rótulo 15 e o prefixo 10.1.1 para R1. O significado de tal anúncio é que R2 informa que anexe o rótulo 15 a todos os pacotes enviados ao seu encontro com destino 10.1.1. Dessa forma, R1 armazena o rótulo em uma tabela junto com o prefixo que ele representa como rótulo "remoto" para quaisquer pacotes que ele envia para esse prefixo. Na figura 27 é visualizado como são alocados os rótulos:

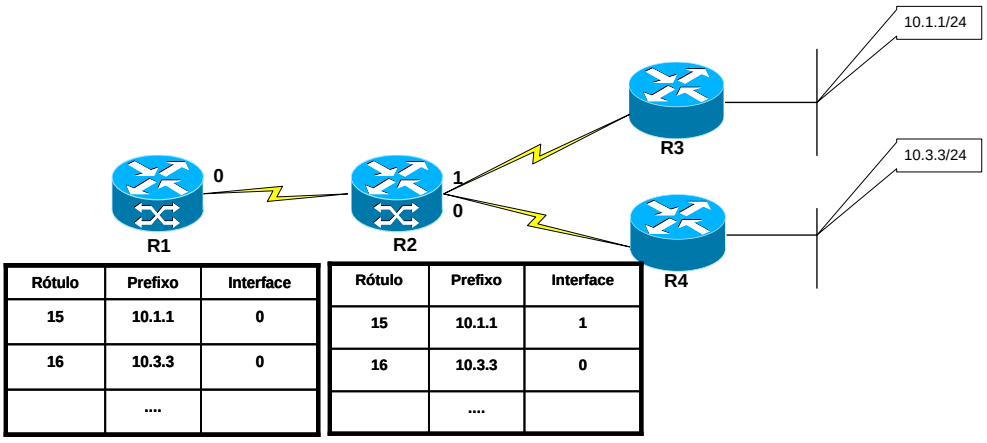


Figura 27: R1 armazena os rótulos recebidos em uma tabela.

Na figura 28, observa-se o anuncio de outro rótulo do roteador R3 para o roteador R2 para o prefixo 10.1.1, e R2 coloca o rótulo “remoto” que descobriu de R3 [36].

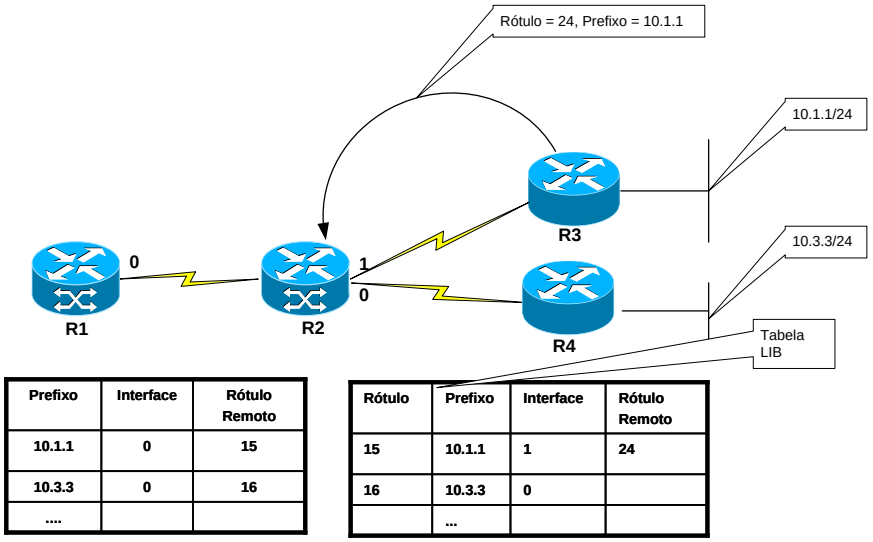


Figura 28: R3 anuncia outra ligação e R2 armazena o rótulo recente em uma tabela.

Todos os prefixos anunciados por um mesmo equipamento vizinho recebem o mesmo rótulo. Com isso, os elementos intermediários em uma rede MPLS não precisam conhecer a tabela de roteamento completa da rede. Um pacote IP com rótulo é encaminhado para o “*next-hop*” baseado somente no rótulo externo, isto é, aquele alocado pelo LDP com base na tabela de roteamento, e isso ocorre até a chegada na rede de destino. Por padrão, o LDP é configurado para formar adjacências somente com seus vizinhos diretamente conectados. Isso é realizado através de um pacote com o endereço de *multicast* 224.0.0.2 (“*all routers*”) e TTL igual a um. Assim que os vizinhos são descobertos inicia-se a troca de rótulos, e as sessões são mantidas através de “*hellos*” periódicos. A perda de três “*hellos*” faz com que a sessão seja desfeita. Uma falha direta na interface faz com que a sessão seja excluída imediatamente. O LDP troca mensagens entre os LSRs (*Label Switch Routers*) com capacidade de LDP empacotados nas PDU (*Protocol Data Unit*). O conteúdo dessas mensagens é construído a partir de objetos componentes do TLV (*Type-Length-Variable*), de modo que, quando a mensagem é lida *byte* a *byte*, há primeiro um identificador do tipo do objeto, depois, uma indicação do tamanho do objeto e, então, o conteúdo do objeto. Tal construção de mensagem tornou o LDP altamente extensível, permitindo que novos valores e funções fossem suportados de uma maneira fácil e compatível com versões anteriores, possibilitando o desenvolvimento do protocolo CR-LDP (*Constraint-Based LSP Setup Using LDP*), que

é um conjunto de TLVs adicionais que tornam o LDP adequado para o uso em uma rede com engenharia de tráfego, conforme descrito na RFC 3212 [38].

CR-LDP (*Constrained-Based LSP Setup Using LDP*) é desenvolvido sobre LDP para fornecer um mecanismo para estabelecer LSP fim-a-fim por meio de uma rede MPLS sob um conjunto de restrições. Ele aceita duas classes de restrições: rotas explícitas e parâmetros de tráfego, e é definido na RFC 3212.

4.6 Funcionamento

Quando o MPLS está habilitado em um roteador, este aloca um rótulo para cada prefixo em sua tabela de roteamento e anuncia o rótulo e o prefixo que ele representa aos seus roteadores vizinhos. A decisão de encaminhamento é tomada com base em tais rótulos, evitando assim o esquema de intenso processo de pesquisa de dados, utilizado no roteamento convencional. Dessa forma, substitui-se a pesquisa normal do endereço IP de um destino por uma pesquisa de rótulo. Para entender o ganho significativo é importante lembrar que, embora os endereços IP sempre tenham o mesmo tamanho, os prefixos IP têm um tamanho variável, e o algoritmo de pesquisa do endereço IP de destino precisa encontrar a combinação mais longa, ao contrário, do mecanismo de encaminhamento de rótulo que utiliza a combinação exata. A distribuição de rótulos dinâmica confere inteligência para preencher as LFIBs na rede em si. Os protocolos de sinalização chamados protocolos de distribuição de rótulos são usados para trocar informações de rótulo a fim de configurar LSPs. Essa distribuição de rótulos é baseada na tabela de roteamento, onde os rótulos são alocados e anunciados de modo a corresponder às rotas na tabela de roteamento local. Esse modo de operação normalmente é chamado de distribuição de rótulos *Downstream Not Requested*. A distribuição de rótulos *Downstream Not Requested* é uma boa solução para garantir que todos os dados possam ser encaminhados imediatamente usando um LSP. Entretanto, ele usa muitos recursos de rede, já que os rótulos são distribuídos para todas as rotas, ainda que nenhum deseje enviar quaisquer dados.

Já a distribuição de rótulos por demanda resolve o problema da distribuição de rótulos não requisitada, ao custo de que os LSPs não são necessariamente estabelecidos. Neste caso, o nó *upstrem* faz uma requisição explícita ao nó de *downstream*, para que um rótulo possa usar uma FEC especifica.

O LDP pode ser usado para distribuir rótulos que podem ser utilizados para tráfego correspondente a uma FEC, conforme requisições específicas para esse rótulo ou, de uma maneira não requisitada, quando novas rotas se tornam disponíveis.

O LDP tem dois métodos de distribuição de rótulos: *Downstream on Demand* e *Downstream Not Requested*.

Uma sessão LDP é uma conversação de protocolo entre LDP *peers* usada para gerenciar e trocar informações sobre um único par de rótulos, um em cada *peers*. As sessões LDP são estabelecidas de forma segura utilizando o protocolo TCP, ou seja, as conexões TCP são configuradas entre LSRs adjacentes usando a porta bem conhecida 646. Uma única conexão TCP não pode transportar mais que uma sessão LDP entre um par LDP parceiros.

A função LDP baseia-se em conexões TCP entre pares LDP. Se uma conexão de rede em um LSR falha, a conexão TCP será derrubada, resultando em uma interrupção na sessão LDP. Mesmo que essa interrupção na sessão LDP seja muito curta, e a conexão TCP possa ser estabelecida imediatamente, os LSRs precisam seguir procedimentos descritos na RFC 3036 e considerar que todos os rótulos anteriormente anunciados por meio do enlace sejam liberados. Isso é necessário porque não é possível saber se alguma mensagem relacionada a esses rótulos foi perdida e porque a conexão pode realmente permanecer inativa por um longo tempo.

Será analisado passo a passo o funcionamento de uma rede MPLS com a que segue na figura 29:

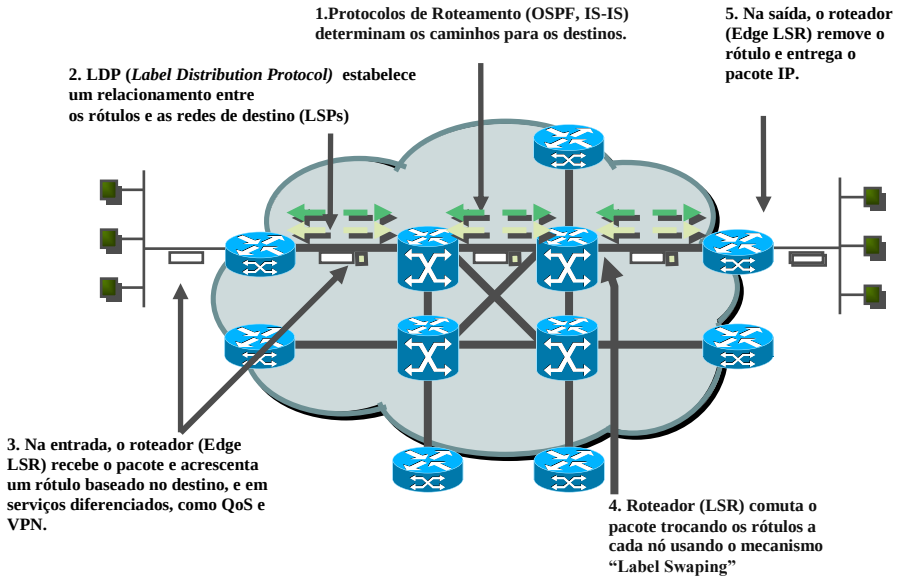


Figura 29: Diagrama de operação do MPLS.

1. No primeiro passo existe a construção da tabela de roteamento.

Através dos protocolos de roteamento OSPF e IS-IS, são construídas as tabelas de roteamento, as quais irão determinar os melhores caminhos para atingir às redes de destino por toda a rede. Deve ser observado que embora o algoritmo de encaminhamento tenha sido mudado da combinação mais longa para a combinação exata, o algoritmo de roteamento pode ser qualquer algoritmo de roteamento IP. O caminho que o pacote seguirá neste ambiente é exatamente o mesmo caminho que ele segue se o MPLS não estivesse envolvido.

2. No segundo passo é feito o mapeamento entre os rótulos e destino IP.

Neste passo há a atuação do protocolo LDP, o qual irá fazer o mapeamento entre rótulos e IPs de destino. Esse processo já foi explicado no item 4.5 desta dissertação.

3. No terceiro passo ocorre o ingresso dos pacotes na rede.

O *Edge* LSR de ingresso recebe os pacotes que irão entrar na rede, executando serviços de nível 3 e valor agregado (QoS e VPN). O requisitante de um fluxo do LSP precisa indicar as características do fluxo, tais como a largura de banda e os requisitos de qualidade de serviço. Em seguida é acrescentado o rótulo aos pacotes. É possível também que, ao invés de um único rótulo, o MPLS permita que os pacotes de dados carreguem uma pilha de rótulos, segundo a ordem de que o último rótulo a ser colocado no pacote deverá ser o primeiro a ser retirado [30].

Uma pilha de rótulos é obtida simplesmente incluindo cabeçalho de calço adicional no pacote de dados, conforme a figura 30:

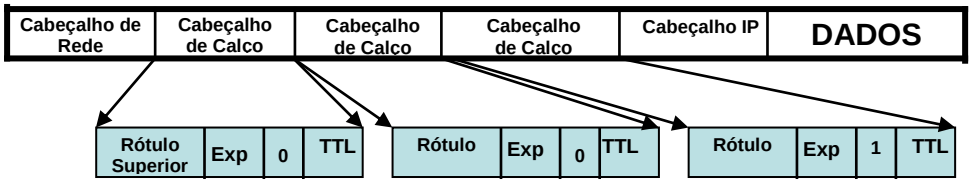


Figura 30: Pilha de rótulos [6].

Na figura 31 é apresentado um exemplo de como são empilhados os rótulos durante o encaminhamento dos pacotes na rede MPLS. Como pode ser visto, os roteadores R1 e R2 acrescentam os rótulos 5 e 8 respecitivamente nos pacotes, em seguida o roteador R3 faz a comutação do rótulo de entrada para os rótulos 3 e 7, onde esses ainda são tunelados com o rótulo 9. Os roteadores R4, R5 fazem comutação no rótulo externo.

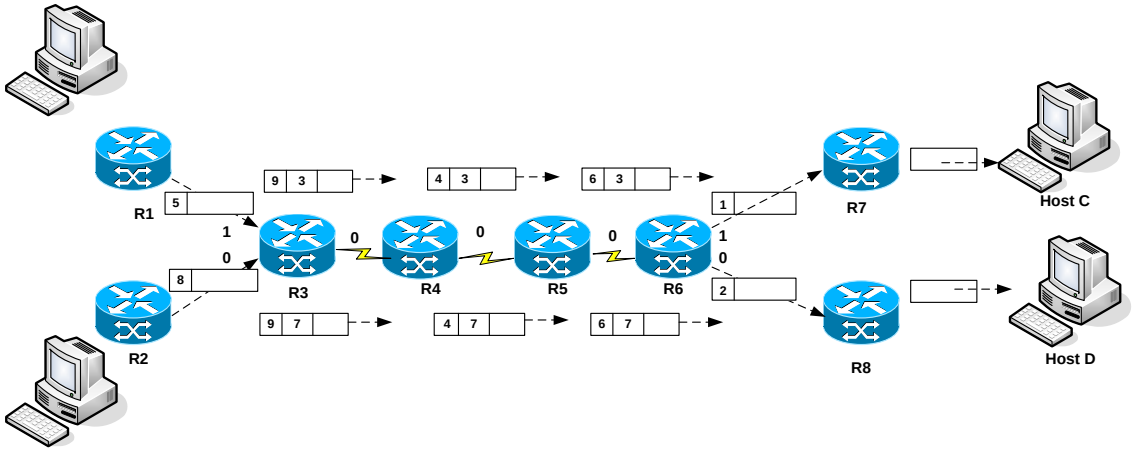


Figura 31: Exemplo ilustrativo de rede MPLS com pilha de rótulos [6].

4. No quarto passo é apresentado o encaminhamento dos pacotes na rede.

O LSR encaminha os pacotes usando mecanismo de *swapping* de rótulos. Na figura 32, é possível visualizar como são encaminhados e acrescentados os rótulos e feito o *swapping* de rótulos.

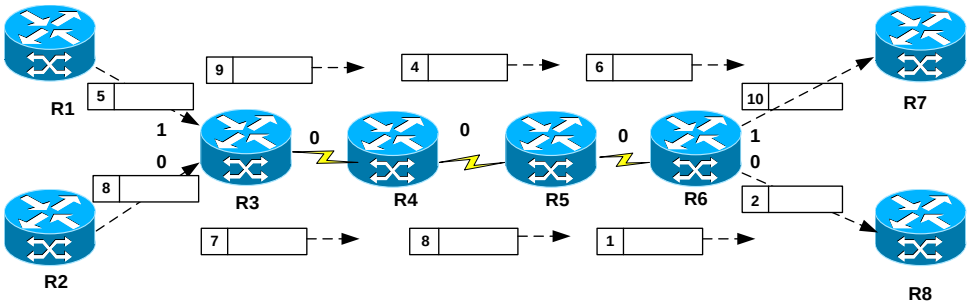


Figura 32: Exemplo de pilha de rótulos em uso [6].

O LSR R1 recebe o tráfego dos *hosts* conectados em sua subrede e o direciona para o destino acrescentando um rótulo. Ao receber os pacotes com rótulos, o LSR R3 lê os rótulos dos dois LSRs anteriores, substitui os mesmos por um novo rótulo de acordo com a tabela LFIB e os encaminha, sendo essa ação repetida por todos os roteadores no núcleo do *backbone*.

5. No quinto passo é realizado o processo de retirada do rótulo na rede.

O *Edge* LSR de egresso remove o rótulo e entrega o pacotes IPs. Esse processo de retirada de rótulo é conhecido como uma técnica PHP (*Penultimate Hop Popping*) [32], que consiste na configuração da retirada do cabeçalho MPLS no penúltimo LSR, e não no *egress* LSR. Essa técnica não compromete o funcionamento de um LSP e

propicia um ganho de desempenho nos roteadores de borda. Na figura 33 pode vista a retirada do rótulo pelo LSR C.

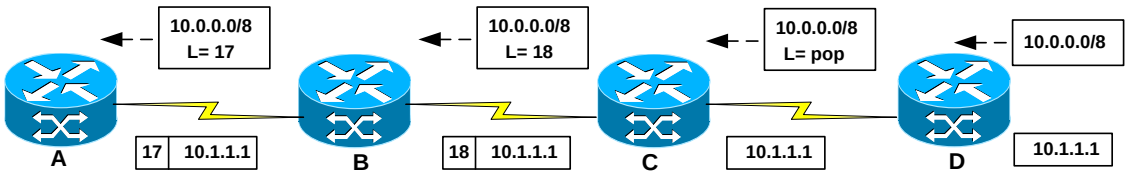


Figura 33: *Penultimate Hop Popping* em rede MPLS.

4.7 Roteamento Explícito

O protocolo MPLS oferece a capacidade de roteamento baseado na origem, que é conhecido na rede MPLS como roteamento explícito. Apesar de o protocolo IP possuir tal característica de roteamento baseado na origem, ele não é muito utilizado por diversos motivos, incluindo o fato de que apenas um número limitado de saltos pode ser especificado, e porque normalmente ele é processado fora do "caminho rápido" na maioria dos roteadores.

A figura 34 apresenta como a capacidade de roteamento explícito do MPLS pode ser aplicado para resolver um problema bastante conhecido na engenharia de tráfego. Esse problema é conhecido como "problema peixe", devido á sua forma.

Será analisado a rede da figura 34 supondo que um administrador tenha determinado que qualquer tráfego fluindo de R1 para R7 deve seguir o caminho R1➔R3➔R6➔R7, e que qualquer tráfego indo de R2 para R7 deve seguir o caminho R2➔R3➔R4➔R7. Um motivo para tal escolha seria fazer o melhor uso da capacidade disponível ao longo dos dois caminhos distintos de R3 a R7. Essa requisição não pode ser feita facilmente com roteamento IP normal, pois R3 não examina de onde o tráfego veio ao tomar suas decisões de encaminhamento. Já com o protocolo MPLS, que utiliza a comutação de rótulos para encaminhar pacotes, é muito fácil conseguir o roteamento desejado.

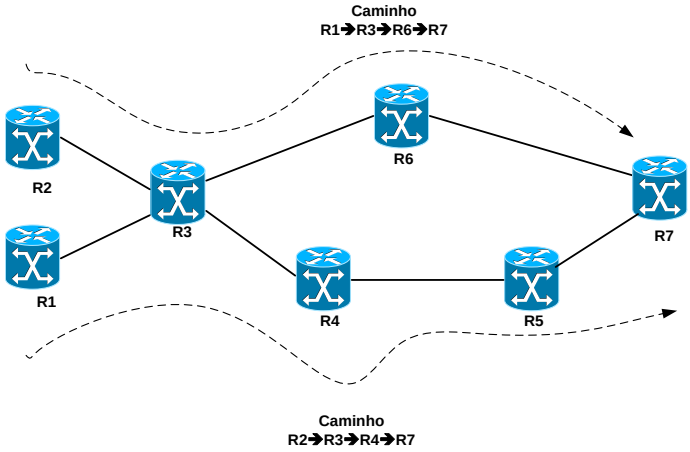


Figura 34: Uma rede exigindo roteamento explícito.

Uma das aplicações do roteamento explícito é na engenharia de tráfego, que é o foco principal desta dissertação, onde o objetivo é apresentar como é possível garantir que os diversos caminhos possam ser utilizados para o envio do tráfego sem sobrecarregar um determinado caminho, deixando o outro subutilizado. No capítulo 6 será detalhado o assunto engenharia de tráfego.

As redes das empresas podem se beneficiar com o MPLS para melhorar a convergência em caso de falhas de enlaces usando MPLS-TE e FRR (*Fast Reroute*).

4.8 Vantagens do MPLS

O MPLS possibilita a melhoria do desempenho no encaminhamento dos pacotes, já que existe uma separação do plano de controle do plano de dados. Existe um ganho na diminuição da latência, já que não há roteamento e sim a comutação dos rótulos.

No MPLS há possibilidade de criar túneis. Tal mecanismo é útil para permitir que muitos LSPs sejam tratados da mesma forma no núcleo da rede sem perder sua individualidade nas bordas. Dessa forma, tem-se um ganho na escalabilidade dos LSRs do núcleo da rede.

Um dos usos mais importantes do MPLS é facilitar a engenharia de tráfego nas redes IPs de provedores de serviços de telecomunicações. A principal capacidade que o MPLS traz às redes com engenharia de tráfego é a possibilidade de configurar um circuito virtual *overlay* comutado para o modelo de roteamento da Internet.

Outro beneficio no MPLS é a possibilidade de transformar um *switch* ATM em um LSR.

As redes corporativas baseadas em WANs tradicionais podem se beneficiar com MPLS na utilização dos enlaces e otimizar o rendimento bruto da rede utilizando MPLS-TE.

A motivação real que deve ser considerada na implementação do MPLS em sua rede são as aplicações que ele permite como, por exemplo:

1. Transporte de qualquer serviço sobre o MPLS;
2. Integração da tecnologia IP e ATM;
3. Provisão de VPN de camada 2 e camada 3;
4. Qualidade de Serviço;
5. Engenharia de Tráfego.

As aplicações citadas são difíceis de implementar ou operacionalmente quase impossíveis de realizar com redes IP tradicionais [5].

4.9 Considerações Finais

Neste capítulo foi justificado o uso do protocolo MPLS como solução única de transporte de pacotes IP. Tal solução foi proposta por grandes empresas na área de redes. O IETF tratou de definir o padrão a ser seguido por essas, com o objetivo de uma melhor integração entre esses vendores. Para justificar o uso do MPLS foi feito um comparativo entre o roteamento convencional e o roteamento baseado em rótulos.

Já que será utilizado o MPLS nesta dissertação, foi necessário explicar onde ele está inserido, bem como o seu funcionamento em uma rede.

Foi feita uma breve explicação sobre o roteamento explícito remetendo para solução do problema a ser demonstrado na aplicação prática desta dissertação, onde será aplicada a engenharia de tráfego para solucioná-lo.

Pode ser comprovado que o MPLS tem diversas aplicações voltadas as necessidades do mercado sendo assim muito utilizado pelos provedores de telecomunicações.

Capítulo 5- Serviços do MPLS

As operadoras de telecomunicações já possuíam vários *backbones* baseados na tecnologia TDM (*Time Division Multiplexing*), para atender diversos serviços separadamente, tais como voz, sinalização, gerência, envio de bilhetes de tarifação, circuitos dedicados para clientes finais, acesso a Internet, entre outros, gerando altos custos de manutenção. Para determinados serviços tinham que manter redes baseadas em tecnologia *Frame Relay* ou ATM, sendo estas utilizadas apenas para transporte de dados dos clientes.

Ao mesmo tempo, as redes de comutação de pacotes estão provendo cada vez mais serviços, porém essas em sua maioria se utilizam do protocolo IP, que em sua própria definição não garante a entrega e não é orientado a conexão. Então a proposta do MPLS é agregar valor aos *backbones* das operadoras sem descartar os investimentos anteriores e ainda possibilitar a integração dos diversos serviços, com a possibilidade de servir com a mesma qualidade das redes de telecomunicações. O protocolo MPLS se mostra uma solução única para um *backbone* e tem agregado diversos serviços.

Embora haja sucesso do MPLS como solução para transporte do IP a maiores distâncias, e há a possibilidade da implementação alternativa de WANs totalmente IP, persiste a oferta de redes legadas de camada 2, com, por exemplo, ATM e *Frame Relay*. Por outro lado, a proliferação de soluções metro *Ethernet* gerou a necessidade da extensão das respectivas redes *switched Ethernet* para transporte de *Ethernet* PDUs a longa distância.

Foram depois desenvolvidas formas de utilização de PSN (*Packet Switching Networks*) [9] como *backbone* universais, capacitados para transportar, de diferentes formas, as diferentes tecnologias de redes camada 2, tais como ATM, *Frame Relay*, PPP, *Ethernet* e SDH/SONET. As PSN utilizadas como *backbone* foram as redes IP e as redes MPLS.

5.1 *Pseudowire*

As operadoras de telecomunicações oferecem serviços de transporte determinístico e estatístico baseados na tecnologia TDM. Essas possuem em seus *backbones switchs Frame Relay* ou ATM e separadamente roteadores para o *backbone*

IP. Isso requer gerenciamento separado aumentando os custos de administração da rede. A tendência natural é que *backbones* de camadas dois e três sejam integrados em um *backbone* MPLS, facilitando principalmente o gerenciamento e administração da rede. As operadoras de telecomunicações passaram a chamar esse *backbone* de Rede de Multiserviços ou RMS, onde neste estão agregados diversos serviços como, por exemplo, aplicação do MPLS, denominada VPWS (*Virtual Private Wire Service*) ou MPLS PW3, que foi também intitulada AToM (*Any Transport over MPLS*) pela *Cisco Sytems*, título esse amplamente utilizado.

Durante a última década, a tecnologia *pseudowire* tem sido estabelecida como o facilitador de fato para migração de serviços legados de comunicação para as redes comutadas por pacotes que possuem maior largura de banda e são mais econômicas. Atualmente, *pseudowire* TDM tem sido implementado em muitos serviços, transportando transparentemente tráfego de voz, vídeo e dados fim-a-fim sobre *backbones* MPLS, *Ethernet* ou IP.

O AToM é a solução para transportar qualquer tráfego de camadas dois e três sobre *backbone* MPLS. O uso de AToM estende o uso do *backbone* podendo este passar a oferecer serviços como *Frame Relay*, ATM, *Ethernet*, TDM sobre a rede MPLS. Na figura 35 é aplicada a funcionalidade do AToM para interligar redes remotas aos *switchs Frame Relay* e esses interligados ao *backbone* MPLS como se estivessem diretamente conectados por um cabo.

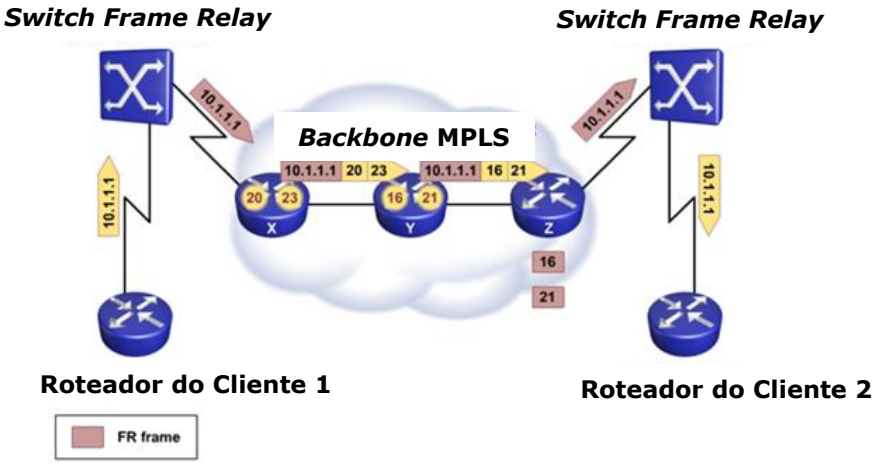


Figura 35: *Backbone* MPLS fornecendo serviço *Pseudowire* -AToM [34].

Na figura 35, a operadora de serviços de telecomunicações está fornecendo um *backbone* MPLS para transmitir os quadros *Frame Relay*. Na entrada do *backbone* MPLS o LSR da operadora de telecomunicações recebe os quadros *Frame Relay* em

uma interface serial de um cliente. Os quadros são encapsulados no MPLS e são atribuidos dois rótulos. O rótulo mais alto indica como os quadros devem ser tunelados e enviados através do *backbone* MPLS da operadora. O segundo rótulo indica de que forma os quadros devem ser transmitidos para o cliente final. Os roteadores do cliente visualizam o encaminhamento em toda rede da operadora de telecomunicações como um enlace *Frame Relay*. Isso significa que os dois *switchs Frame Relay* podem ser configurados para fornecer PVC entre os dois roteadores do cliente final.

Com essa característica o AToM pode fornecer aplicações que incluem:

1. Fornecimento de transporte de circuito ATM e *Frame Relay* para serviços de telecomunicações legados, como por exemplo, tráfego GPRS (*General Packet Radio Service*) sendo transportado sobre *Frame Relay* em um *backbone* MPLS;
2. Garantias de largura de banda e atraso ponto a ponto, quando combinados com outras técnicas com DS-TE (*Differentiated Services Traffic Engineering*) e MPLS QoS;
3. Extensão do domínio de *broadcast* da camada 2;
4. Conectividade de ponto de presença (POP) remoto, especialmente operadoras de telecomunicações.

Em redes de telecomunicações o serviço do *pseudowire* possibilita a integração e a redução do custo quando provêm túneis virtuais que levam através do *backbone* MPLS a conexão para transpor aplicações legadas, como pode ser visualizado na figura 36. Nesta é apresentada uma MGW (*Mídia Gateway*) do fabricante *Ericsson* que apresenta interface em ATM trocando informações com uma RNC (*Radio Node Controller*) também da *Ericsson* com interface em ATM, sendo a comunicação entre os dois elementos ocorre via um *backbone* MPLS IP com o serviço do *pseudowire*. Para os os dois elementos da operadora de telecomunicações a comunicação ocorre como se eles estivessem conectados diretamente via um circuito em TDM.

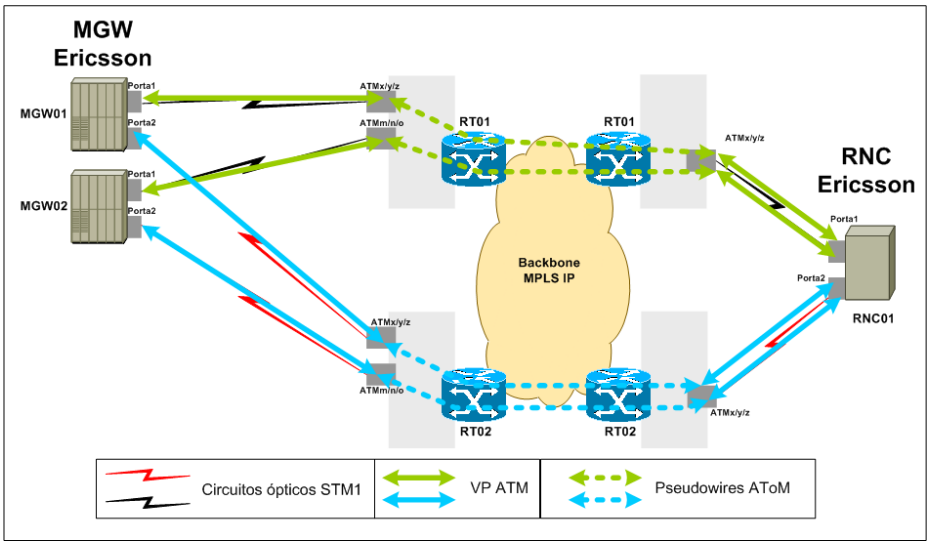


Figura 36: Aplicação do *pseudowire* em rede de serviços de telecomunicações móvel.

Os *pseudowire* podem ser estabelecidos manualmente ou através de usos de protocolos de sinalização, tais como o BGP e o LDP estendidos.

5.1.1 Arquitetura de Referência *Pseudowire*

A RFC 3985 descreve, com base no conceito a *pseudowire*, a emulação de serviços como ATM, *Frame Relay*, *Ethernet*, TDM e SONET/SDH sobre PSNs (*Packet Switched Networks*), utilizando o *backbone* MPLS como transporte. O *pseudowire* permite a conexão perfeita entre dois elementos de rede criando enlaces lógicos, ou túneis virtuais, através da rede de pacotes. No *pseudowire* TDM, os fluxos de transmissão para E1, E3 são encapsulados em pacotes ao entrarem na rede e, então, reconstruídos na saída do *pseudowire*, onde a informação de relógio também é regenerada. Como resultado, o tráfego em tempo real é transportado transparentemente sem distorção, evitando as complexidades de traduzir dados de sinalização e, ao mesmo tempo assegurando que os critérios de sincronização foram atendidos.

O AToM usa o modelo de referência da arquitetura *pseudowire*, conforme figura 37, para transportar tráfego de nível 2 através de um *backbone* MPLS. Esse modelo é baseado no trabalho do grupo IETF PWE (*PseudoWire Emulation Edge to Edge*), definido pela RFC 3985 [52], que provê o *framework* para emulação *edge to edge* sobre uma rede baseada em pacotes de uma operadora de telecomunicações.

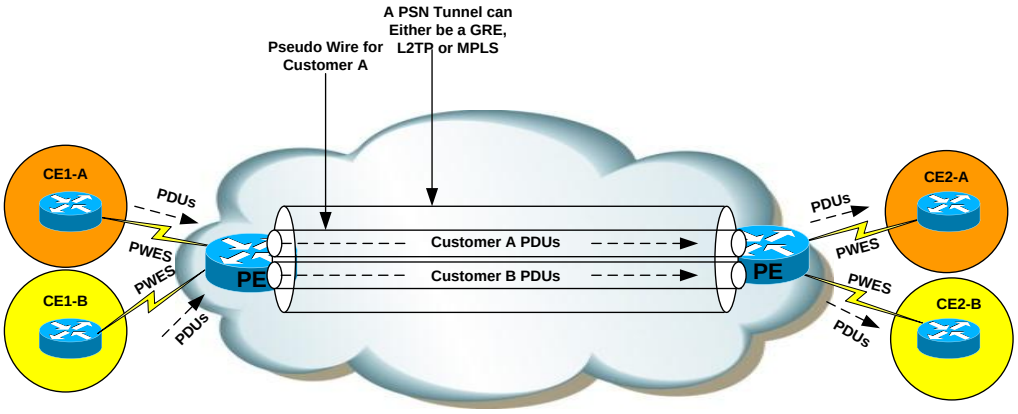


Figura 37: Modelo de Referência *Pseudowire* [30]

Os dois roteadores PEs (*provider edges)* provêm um ou mais *pseudowires* para possibilitar a comunicação entre dois CEs (*Customer Edges)* através do *backbone* MPLS, de forma tal que o tráfego nos *pseudowires*, que emulam um tipo de serviço de telecomunicações, seja invisível para o interior do *backbone* MPLS.

O *pseudowire* é uma conexão lógica entre dois dispositivos PE, que conectam dois *pseudowire end-services* (PWES) do mesmo tipo.

Os roteadores PEs são configurados como os pontos finais ou extremos de uma conexão *pseudowire*. Após a formação do *pseudowire*, PDU (*Protocol Data Unit*) de camada 1 ou de camada 2 são encapsulados no dispositivo PW (*PseudoWire*) de ingresso. O PDU encapsulado é enviado sobre o PW para o dispositivo PW de egresso, onde cabeçalhos de camada 2 ou de camada 3 são reconstruídos, e os quadros são enviados no formato original para o dispositivo CE [30].

A RFC 4447 [53] considera, no entanto, a utilização de túneis MPLS para a condução e a agregação de PWs, constituídos nos moldes da facilidade *hierarchy os routing Knowledge*, de forma a isolar logicamente o *backbone* MPLS do processo de controle que ocorre no exterior dessa rede.

Os túneis MPLS podem ser constituídos pelos processos do MPLS básico, utilizando protocolos IGP convencionais e LDP, ou podem consistir em túneis MPLS-TE, incorporando assim as facilidades do TE.

5.2 VPN (*Virtual Private Network*)

VPN (*Virtual Private Network*) [62] é uma área de crescimento importante na Internet. Usando protocolos padronizados, as empresas são capazes de conectar suas redes privadas usando os recursos econômicos e altamente disponíveis da Internet. O

objetivo da Internet é proporcionar comunicação entre os nós das redes de modo irrestrito. Porém, existem muitas situações nas quais se exige uma conectividade controlada e uma maneira de obter tal resultado é a utilização de VPNs.

O crescimento das VPNs tem sido acompanhado por uma explosão nas técnicas disponíveis para prover essa função. A maior parte dessas técnicas utiliza abordagens padronizadas, mas cada uma utiliza protocolos diferentes e possui suas próprias vantagens e desvantagens [62].

O MPLS vem se tornando um suporte fundamental para disseminação de VPNs, devido a sua aceitação pelas operadoras de telecomunicações em separar os tipos de tráfegos de usuários comuns e tráfego de serviços legados.

Uma solução de VPN híbrida que utiliza BGP e MPLS IP VPN é descrita na RFC 4364 [55]. Essa denominação deriva do fato de que o roteamento inter-AS e a distribuição de rótulos se efetivam com base na RFC 4760 [56] (*Multiprotocol Extensons for* BGP-4) e na RFC 3107 [57] (*Carrying Label Information in* BGP-4).

O MPLS pode ser utilizado para estabelecer VPNs de camada 2 e 3, pois esse protocolo possibilita a criação de túneis. O MPLS fornece um mecanismo eficaz para suporte a VPNs. VPNs MPLS utilizam um rótulo adicional para determinar a rede de destino.

5.2.1 Tipos de VPN

O MPLS pode ser considerado um meio de montar túneis e isso o torna adequado para montagem de VPNs de vários tipos.

Tais túneis podem ser usados para estabelecer as VPNs de camada 2 para redes *Ethernet*, ATM, *Frame Relay* e outros capazes de usar o MPLS como transporte. A forma mais simples de VPN MPLS é uma VPN de camada 2. Neste tipo de VPN, o MPLS é utilizado para criar um túnel, através de LSRs, com o objetivo de conectar redes remotas, como pode ser visualizado na figura 38. Esse tipo de VPN assegura a separação de tráfego em um *backbone* MPLS. Na figura 38 o túnel é estabelecido do roteador PE (*provider edges*) de entrada passado pelo roteador P (*provider*) até o roteador PE (*provider edges*) de saída. Cada túnel provê um fio virtual (*pseudowire*) entre a origem e o destino, para conectar diferentes partes da VPN. O roteador do CE (*Customer Edges)* na rede do cliente o *backbone* MPLS apresenta uma conexão ponto-a-ponto.

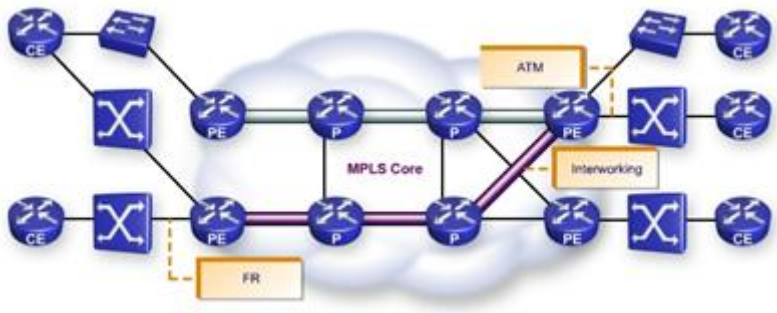


Figura 38: VPN MPLS de Camada 2 [7].

O MPLS pode ser utilizado para estabelecer VPN de camada 3, e essas utilizam pilhas de rótulos MPLS para enviar pacotes por túnel através de uma rede IP. Como alternativa, os pacotes MPLS podem ser encapsulados em algum outro mecanismo de tunelamento, para permitir que sejam transportados pelo núcleo do *backbone*. Essa segunda opção pode ser particularmente útil quando o MPLS for usado dentro da VPN, ou quando muitos pontos da borda oferecem, cada um, acesso a várias VPNs e for desejável reduzir o número de túneis pela rede. Onde as VPNs serão separadas pelo rótulo acrescentados nos pacotes. Como o exemplo apresentado na figura 39, o roteador PE de entrada acrescenta os rótulos 65 e 13 para diferenciar as VPNs.

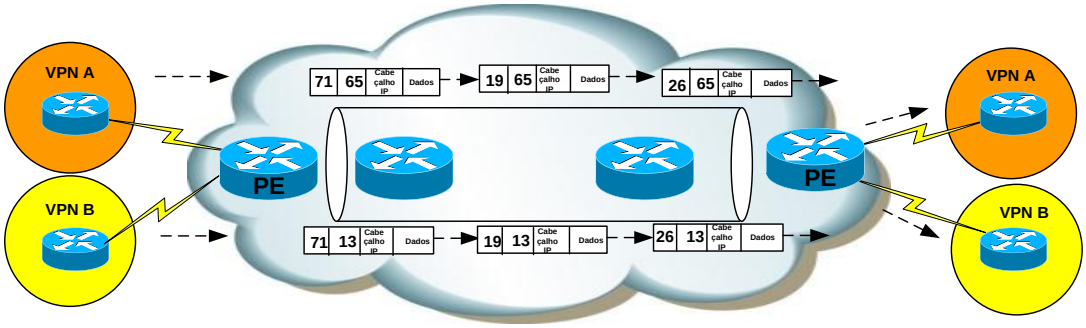


Figura 39: MPLS conectando várias VPNs [30].

As VPNs ópticas são um conceito novo, que tratam da provisão automática de serviços de dados por meio de núcleo óptico. As redes que estão sendo conectadas, nesse caso, provavelmente são fragmentos da rede inteira de uma operadora de telecomunicações. Um exemplo desse tipo de VPN é o GMPLS (*Generalized* MPLS), que é uma extensão do serviço MPLS-TE e está descrito na RFC 4397.

5.2.2 Modelos de VPN

O termo VPN é bastante utilizado e as definições variam, mas intuitivamente é possível definir uma VPN considerando primeiro a idéia de rede privada. As empresas com muitas filiais distribuídas ao longo do país normalmente criam redes privadas contratando circuitos dedicados de operadoras de telecomunicações como, por exemplo, E1, E3, STM-1 e STM-4. Nesse tipo de rede a comunicação é restrita apenas aos pontos remotos e a matriz da empresa, o que normalmente é desejável por motivos de segurança. Para montar uma rede privada virtual, os circuitos dedicados devem ser substituídos por uma rede compartilhada com outras empresas – compartilhamento de multiplexadores ou *switchs Frame Relay* e ATM [59]. As tecnologias *Frame-Relay* e ATM foram as primeiras tecnologias a serem adotadas largamente como VPN. Esse modelo é conhecido como modelo *overlay* e nela a operadora de telecomunicações não participa do roteamento. Essas redes consistem de vários dispositivos, pertencentes aos clientes e as operadoras de telecomunicações, que são componentes da solução VPN [30]. Um segundo modelo de VPN é o *peer to peer* onde a operadora de telecomunicações participa do roteamento dos pacotes na rede. Um exemplo de VPNs no modelo *peer to peer* são as VPNs BGP/MPLS IP. Uma VPN MPLS combina as melhores características do modelo de VPN *overlay* e *peer to peer*.

5.2.2.1 - O Modelo *Overlay*

No modelo *overlay* para VPNs, toda a lógica funcional das VPNs ocorre nos equipamentos dos usuários, limitando a operadora de telecomunicações a fornecer circuitos físicos como, por exemplo, E1, E3 e STM ou circuitos virtuais de redes como *Frame Relay* ou ATM, mas também o modelo *overlay* pode prover protocolos de camada 3, nesse caso, são utilizados túneis para construir as redes *overlay*, como exemplo, túneis GRE (*Generic Routing Encapsulation*), L2TP e IPSec para interconectar sites dos clientes. Na figura 40 é possível visualizar um *backbone* ATM e *Frame Relay*, essa apresenta *switchs* FR (*Frame Relay*) e *switchs* ATM interligados por vários PVCs formando uma rede *full mesch*, onde cada PVC tem custo adcional para ser estabelecido.

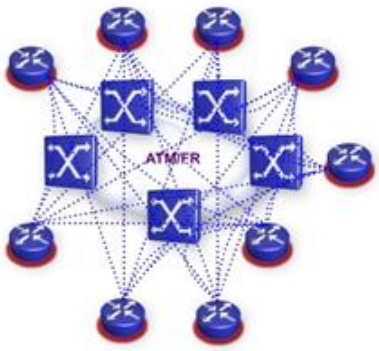


Figura 40: Modelo de VPN *Overlay* com ATM e *Frame-Relay* [7].

Neste modelo de VPN o planejamento e administração dos endereços IPs não são feitos pela operadora de telecomunicações e fica sob responsabilidade do cliente final administrar o roteamento, gerando a necessidade de possuir uma equipe especializada para esse fim. Para o cliente final a rede funciona como se os roteadores estivessem conectados por circuitos físicos, como pode ser visualizado na figura 41:

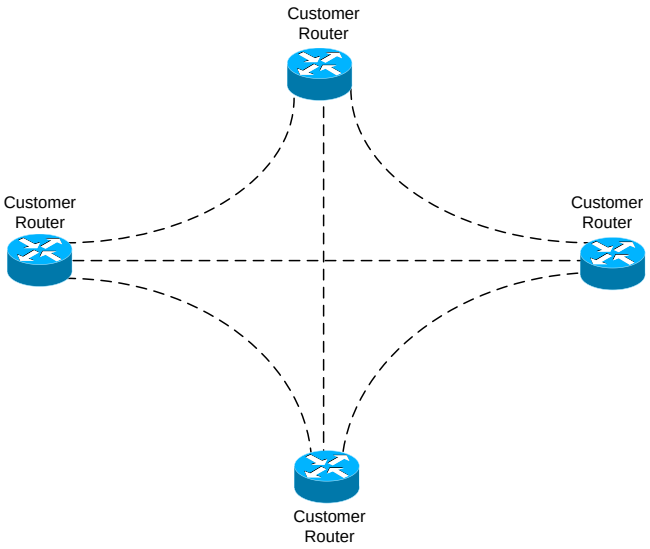


Figura 41: Modelo de VPN *Overlay* – Ponto de vista do cliente [34].

Para grandes redes o modelo *overlay* não é adequado, pois para cada novo ponto que entra na rede existe a necessidade de criação de um novo PVC pela operadora de telecomunicações com todos os pontos da rede, gerando um custo para o cliente final. Ou seja, esse modelo não é escalável.

Comparando o modelo tradicional do *overlay* das tecnologias ATM ou *Frame Relay* as VPNs de camada 3 MPLS fornecem mais escalabilidade em termos operacionais e de gerenciamento, pois elas fazem seus encaminhamento através dos protocolos TDP/LDP que gera uma malha completa ou seja uma rede *full mesh* com

vários LSPs que no modelo *overlay* ATM ou *Frame Relay* isso só é possível na criação de vários PVCs.

5.2.2.2 – O Modelo *Peer-to-Peer*

No modelo *peer-to-peer* para VPNs, as operadoras de telecomunicações participam diretamente dos mecanismos funcionais das VPNs, proporcionando o roteamento entre os roteadores dos clientes e da operadora de telecomunicações. Os CE e PE das tecnologias das VPNs constituem-se em pares no que concerne o processo de roteamento das VPNs.

Um exemplo desse tipo de VPN pode ser visualizado na figura 42, onde há os roteadores CE (*Customer Edge*), PE (*Provider Edge*) e P (*Provider*), que tiveram as funções definidas nesta dissertação na sessão 4.5. Nesse serviço, o *backbone* da operadora de telecomunicações provê uma a conexão *full-mesh* de forma a otimizar o encaminhamento dos pacotes entre os pontos.

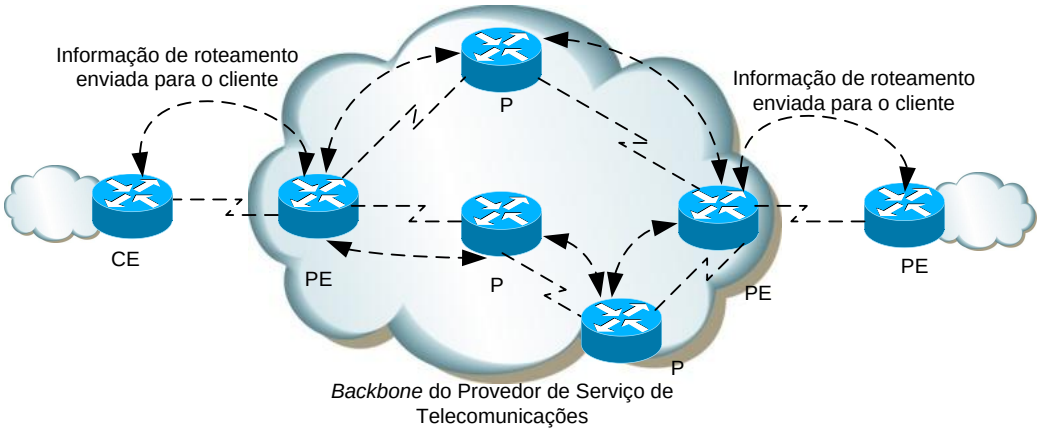


Figura 42: Modelo *Peer-to-Peer* [34].

Essa forma de solução evita as limitações fundamentais das VPNs no modelo *overlay*, particularmente no diz respeito à escalabilidade.

Já no modelo *peer-to- peer* não existe a necessidade de criação de PVCs. Nesse modelo de VPN a privacidade é garantida através de lista de acesso, que tem o objetivo de separar o tráfego entre os diferentes clientes através de configurações de filtros de pacotes, controlando assim os dados dos clientes. Outra maneira de proporcionar o isolamento e privacidade entre os clientes é através de filtros de rotas, anunciando ou parando rotas para determinados clientes, esses dois métodos podem também trabalhar em conjunto [34].

5.2.3 Conceitos Específicos da VPN MPLS

O MPLS é utilizado para estabelecer VPN *peer-to-peer*. Uma VPN MPLS é constituída pelos seguintes equipamentos:

CE – *Customer Edge*;

PE – *Provider Edge*;

P – *Provider*.

As funções desses equipamentos já estão definidas nesta dissertação no item 4.5.

Para os clientes que utilizam essa modalidade de VPN é necessário apenas configurar um protocolo de roteamento dinâmico ou acrescentar rotas estáticas em direção ao roteador PE da operadora de telecomunicações, que em muitas vezes se encarrega de efetuar as configurações necessárias no roteador PE e CE. Para os clientes a VPN MPLS torna-se transparente de modo que toda a configuração e manutenção no *backbone* são realizadas pela operadora de telecomunicações reduzindo o custo de investimento das empresas contratantes do serviço. Os roteadores CEs podem estar conectados a um ou mais PEs, e cada PE pode fornecer serviço de VPN para mais de uma VPN.

No roteador PE são executadas múltiplas funções, pois o mesmo deve ser capaz de isolar o tráfego se mais de um cliente estiver conectado ao mesmo e, para cada cliente, é designada uma tabela de roteamento independente. O PE mantém múltiplas *forwarding tables*. Uma dessas tabelas é a *default forwarding table*, e as demais são denominadas VPN *routing and forwarding tables*. O roteamento através do *backbone* é desempenhado usando um processo de roteamento na tabela de roteamento global. Esses roteadores de *backbone*, conhecidos como P (*Provider*) fazem a comutação de rótulos entre os PEs e não requerem quaisquer configurações de VPN. Os roteadores CEs não conhecem os roteadores Ps, e a topologia interna da rede da operadora de telecomunicações é transparente para o cliente, conforme se observa na figura 43:

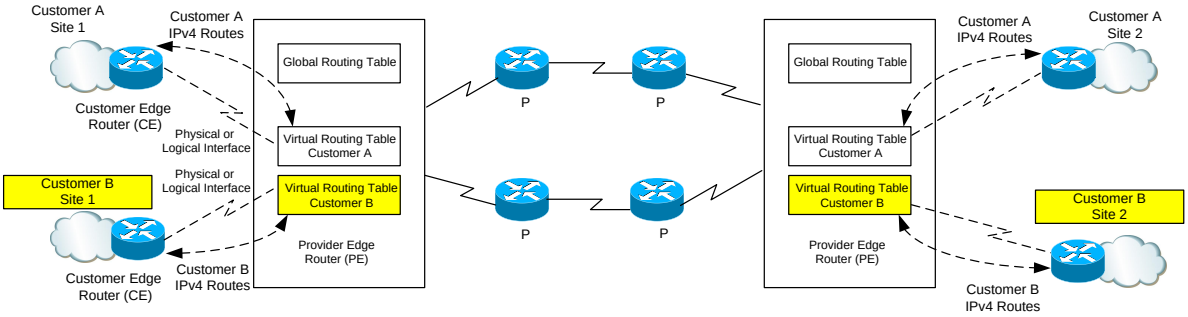


Figura 43: Topologia de VPN MPLS [34].

O protocolo MPLS proporciona a separação das VPNs dos clientes. Tal distinção é obtida devido a separação de tabelas de roteamento, plano de endereçamento e tráfego, de modo que dois clientes de empresas diferentes podem possuir os mesmos endereços IPs, como cada cliente possui sua própria tabela de roteamento, os seus pacotes não serão misturados. Essa separação é possível devido a VRF (*virtual routing and forwarding table*) do roteador PE.

O conceito de VRF é criar um roteador virtual para cada cliente que se conecta a operadora de telecomunicações, neste caso há o compartilhamento de CPU, de largura de banda e de recursos de memória com outros roteadores virtuais pertencentes ao mesmo roteador PE. A função de uma VRF é similar a uma tabela de roteamento global, exceto que ela contém todas as rotas pertencentes para uma VPN específica. Uma VRF pode ser associada a um roteador CE ou VLAN criada em um *switch* de camada dois conectado a um roteador PE.

Observa-se que as operadoras de telecomunicações têm um ganho com o uso de roteadores virtuais, já que não é necessário um novo investimento em equipamentos.

Na arquitetura de uma VPN MPLS é necessário o conhecimento de alguns atributos que são utilizados nos roteadores PEs, que são: RD (*Route Distinguisher*), RT (*Route Targets)* e MP-BGP (*Multi Protocol Border Gateway Protocol*).

O RD é um identificador único de 64 *bits* que é inserido na frente do IPv4, portanto sendo único dentro do *backbone* MPLS. Pode ser visualizado na figura 44 como é inserido o valor do RD. Nessa é possível observar que o mesmo endereço, 172.16.10.0/24, recebido de dois diferentes clientes, tornam-se únicos no *backbone* ao receber os valores de RD no roteador PE do *backbone* MPLS, no caso, 1:100 e 1:101.

O valor do RD é acrescentado no endereço IPv4 de forma que eles serão distintos dentro do *backbone* MPLS. No roteador PE são criadas duas entradas na tabela

de roteamento global, onde são acrescentadas as informações de roteamento dos clientes A e B respectivamente.

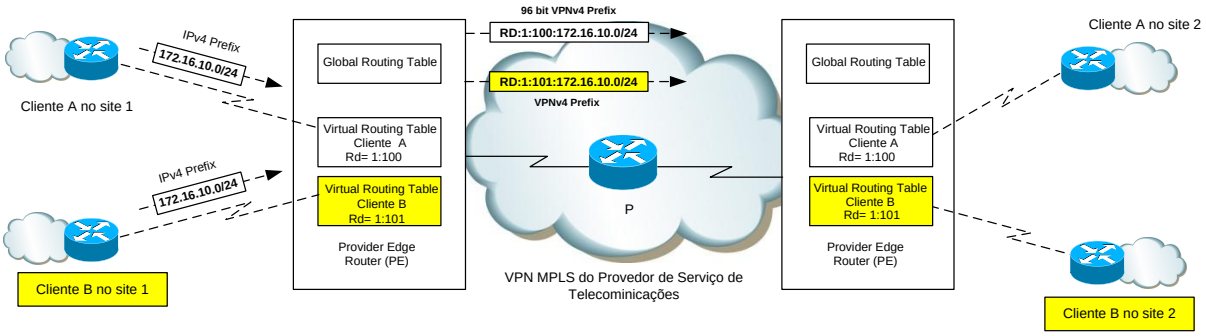


Figura 44: Funcionamento *Route Distinguisher* [34].

Na figura 44 são apresentados dois clientes A e B que estão conectados a uma operadora de telecomunicações que fornece o serviço de VPN MPLS. Neste caso, os clientes estão usando o endereço de rede 172.16.10.0/24, o que resulta em um espaço de endereçamento que se sobrepõem. O roteador CE, que representa o cliente B, envia uma atualização de roteamento IPv4 para o roteador PE. Para lidar com a sobreposição de espaço de endereço, o roteador PE deverá acrescentar um RD de 64 bits no prefixo IPv4, resultando em um prefixo global exclusivo de 96 bits VPNv4. A combinação dos endereçamentos IPv4 e esses identificadores de rotas fazem com que as rotas IPv4 sejam únicas através da rede VPN MPLS. Dessa forma, é possível aos clientes de diferentes VPNs o uso dos mesmos endereços privados [30]. Neste caso, PE possui prefixos RD associados aos enlaces de acesso aos clientes A e B, resultando em duas vias originais e distinguíveis via VPNv4. O prefixo VPNv4 é propagado através de uma sessão MP-BGP (*Multiprotocolo BGP*), conforme definido nas RFCs 2858 [60] e 4760 [56], interno aos outros roteadores PE.

O RD não pode indicar que um cliente participa em mais de uma VPN. Um método diferente, onde é necessário um conjunto de identificadores VPN, poderia ser anexado a uma rota para indicar a sua participação em várias VPNs. Para solucionar esse caso foi introduzido o RT na arquitetura VPN MPLS para suportar a este requisito. A natureza genérica da arquitetura do MPLS e VPN requer a rota de destino que deve ser usada para indicar a VPN de participação em todas as topologias. Também topologias simples requerem o uso de RD.

O RT (*Route Target*) é um atributo que indica uma coleção de VRFs pelo qual um roteador PE irá distribuir as rotas, ou seja, ele indica quais rotas devem ser importadas e exportadas pelo MP-BGP, permitindo assim que possa haver conversação entre diferentes VRFs, e que também possa ser feito restrições de importação e exportação de rotas.

5.3 Qualidade de Serviço (QoS)

Durante muitos anos, as redes por comutação de pacotes ofereceram a promessa de admitir aplicações de multimídia, ou seja, aquelas que combinam áudio, vídeo e dados. Afinal uma vez digitalizadas, as informações de áudio e vídeo tornam-se como qualquer outra forma de dados – um fluxo de *bits* a serem transmitidos.

Todavia, existem diversos parâmetros na transmissão de áudio e vídeo por uma rede do que apenas oferecer largura de banda suficiente. Neste caso, uma referência muito importante é o atraso e a variação do atraso que para as aplicações em tempo real é de fundamental importância.

As características distintas das aplicações de tempo real precisam de algum tipo de garantia da rede de que os dados provavelmente chegarão a tempo.

A chegada a tempo precisa ser fornecida pela própria rede (os roteadores), e não apenas nas bordas das redes (*hosts*).

Qualidade de serviço é um conceito familiar no setor de telecomunicações. Desenvolvida principalmente para transporte de tráfego de voz, a rede de telecomunicações moderna é sensível aos aspectos de ruído, distorção, perda, atraso e variação de atraso (*jitter*), que torna a voz humana ininteligível ou inaceitavelmente difícil de decifrar. Apesar disso, o setor está dominado por protocolos patenteados, apesar da existência de soluções padronizadas e de requisitos reguladores, para convergir em técnicas interoperáveis.

Além disso, existem algumas diferenças importantes entre a estrutura das redes IP e das redes de telecomunicações. Talvez a diferença mais óbvia seja o modo como as redes de telecomunicações são orientadas a conexão, ou baseadas em circuitos virtuais, de modo que o tráfego para determinar o fluxo segue, de forma confiável, o mesmo caminho pela rede. Naturalmente, o tráfego IP é roteado com base em cada pacote.

Para construir uma rede usando qualidade de serviço, são necessárias políticas e arquiteturas de QoS [41]. Uma das primeiras tentativas na aplicação de políticas de

qualidade de serviço foi focalizar na priorização do tráfego. Quando há um congestionamento, pacotes são descartados sem distinção, não havendo garantia que o serviço será realizado com sucesso, nem que haverá bom desempenho.

Uma Arquitetura de QoS é essencial para orientar a organização do tráfego, permitindo o controle dos resultados obtidos com a priorização do mesmo, além de garantir qualidade à determinados tipos de tráfegos críticos, como é o caso da sinalização em rede de telecomunicações. O IETF define dois modelos para implementação de QoS numa rede IP [42]: o de serviços integrados (*IntServ*) e o de serviços diferenciados (*DiffServ*).

Qualidade de serviço e MPLS são em um nível político, semelhantes. Ambas são tecnologias que ganharam destaque nos últimos anos. [5]. No entanto, em nível técnico, o QoS e o MPLS são muito diferentes.

5.3.1 Arquiteturas de QoS

Como já foi dito no capítulo dois, o modelo de operação tradicional das redes IP foi baseado no serviço de melhor esforço. Nenhuma garantia foi gerada sobre a qualidade de serviço fornecido às aplicações que utilizam a Internet. A pesar disso, com todas as facilidades de serviços sendo disponibilizadas nessa rede, os provedores de serviço despertaram para venda de acessos diferenciados.

As operadoras de telecomunicações tem a necessidade de se manterem competitivas no mercado, visto que cada vez mais as redes comutadas por pacotes só utilizavam os meios de transmissão como transporte para os provedores de acesso a Internet ou como interligação entre LANs coorporativa. Foi nesse momento que entraram em franca expansão as redes de multiserviço das operadoras de telecomunicações, onde inicialmente esses *backbones* nasceram para prover serviços de VPN de camada 2 e 3, acesso a Internet para grandes clientes e sobre tudo disponibilidade baseados em SLAs (*Service Level Aggrements*). Tais redes necessitam de implementações de QoS, já que possuem serviços integrados baseados em fluxos ou até serviços diferenciados por classes.

5.3.1.1 Serviços Integrados (IntServ)

Serviços Integrados, algumas vezes chamado *IntServ* (*Integrated Services*), é um modelo de QoS baseado em fluxo, significando que o usuário precisa criar um fluxo, uma espécie de circuito virtual, da origem até o destino e informar a todos os roteadores sobre os recursos necessários. O objetivo desse modelo é obter largura de banda e latência necessárias para uma determinada aplicação [43].

IntServ oferece uma série de maneiras padronizadas para classificar os fluxos de tráfego e recursos da rede, focalizando as capacidades e a estrutura comum dos roteadores de pacotes IP. A finalidade dessa função é permitir que as aplicações escolham entre vários níveis de remessa, bem caracterizados, de modo que possam quantificar e prever o nível de serviço que o tráfego receberá.

A especificação do fluxo é separada em duas partes:

Rspec – define a especificação de recurso que o fluxo precisa reservar (*buffer*, largura de banda); e

Tspec – define a especificação de tráfego do fluxo (como a largura que varia com tempo).

O modelo de serviços integrados permite que os fluxos de aplicações individuais especifiquem suas necessidades aos roteadores, usando um mecanismo de sinalização explícito e, para isso, utilizam o protocolo RSVP (*Resource Resevation Protocol*) [44]. O RSVP é basicamente um protocolo de requisição/resposta com mensagens *Path* sendo usado para definir um caminho e requisitar recursos para fluxos de tráfego, e mensagens *Resv* retornando pelo caminho para indicar que recursos devem ser reservados.

Esse fluxo de mensagens atende aos requisitos para distribuição de rótulos *downstream* por demanda e pode ser extendido facilmente acrescentando informações às mensagens. Devido a esse motivo, o protocolo RSVP foi extendido para a engenharia de tráfego.

Embora o RSVP contenha bons mecanismos que descrevem o tráfego e especificam os requisitos de reserva, ele não possui recursos para outros aspectos necessários para protocolo MPLS com engenharia de tráfego. Neste caso foi necessário criar a extensão do protocolo RSVP, RSVP-TE.

O modelo de serviço integrado baseia-se em quatro componentes conforme a figura 45, e serão explicados a seguir**:**

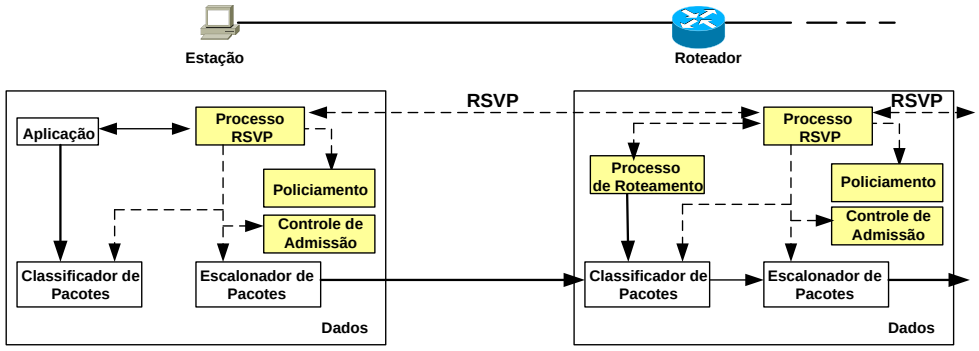


Figura 45: Componentes da Arquitetura *IntServ* [45].

- Controle de admissão: quando um novo fluxo deseja receber determinado nível de serviço, o controle de admissão examina o *Tspec* e o *Rspec* do fluxo e tenta decidir se o nível de serviço desejado pode ser fornecido. Ou seja, é implementado um algoritmo no roteador para determinar se a solicitação de QoS de um novo fluxo pode ser atendida sem interferir nas garantias que outros fluxos receberam.
- Policiamento: verifica se o fluxo está de acordo com as especificações negociadas na fase de estabelecimento da conexão. Fluxos fora do acordo podem ter seus pacotes descartados para evitar congestionamentos.
- Escalonador de pacotes: gerencia os pacotes nas filas dos roteadores de modo que recebam o serviço que foi solicitado. Os detalhes da classificação são bastante relacionados aos detalhes do gerenciamento da fila.
- Classificador: associa cada pacote à reserva apropriada, para que ele possa ser tratado corretamente. É feito o reconhecimento dos fluxos segundo sua identificação, mapeando os pacotes desses fluxos nas diferentes categorias de serviço, que por sua vez notifica a função de policiamento e, caso os pacotes estejam em conformidade com o controle imposto pelo policiamento, os coloca nos *buffers* das filas de saída apropriada.

Por vários motivos, o *IntServ* nunca foi escalável até o ponto necessário para chegar a redes do tamanho da Internet, devido a alta troca de sinalização na rede. O *IntServ* é bom para redes pequenas a médias, mas sua necessidade de criar microfluxos *host* a *host*, por aplicação, através de uma rede, significa que ele não pode crescer até o nível que as grandes redes de provedores de serviço precisam [5].

5.3.1.2 Serviços Diferenciados (*DiffServ*)

O *DiffServ* (*Differentiated Services*) é uma técnica de classificar os pacotes dentro de uma rede para que possam ser tratados de forma diferente, priorizando aqueles que pertencem a uma mesma FEC.

Os serviços diferenciados (DS ou *Diffserv*) foram introduzidos pelo IETF para tratar das limitações dos serviços integrados, tais como escalabilidade e complexidade. A RFC 2475 define uma arquitetura para serviços diferenciados. O DSCP (*Differentiated Services Code Point*) é utilizado por diversos mecanismos de QoS para oferecer diferentes qualidade de serviços na sua rede.

O *DiffServ* possui dois componentes principais:

- Condicionamento de tráfego – Define como é feito o policiamento, a classificação, a marcação e a moldagem, ocorre apenas na borda da rede; e

- Comportamentos por salto – Consiste essencialmente em mecanismos de enfileiramento, escalonamento e descarte de pacotes, ocorre a cada *hop* da rede;

O condicionamento de tráfego geralmente envolve classificação, policiamento, marcação e moldagem, conforme pode ser visto na figura 46:

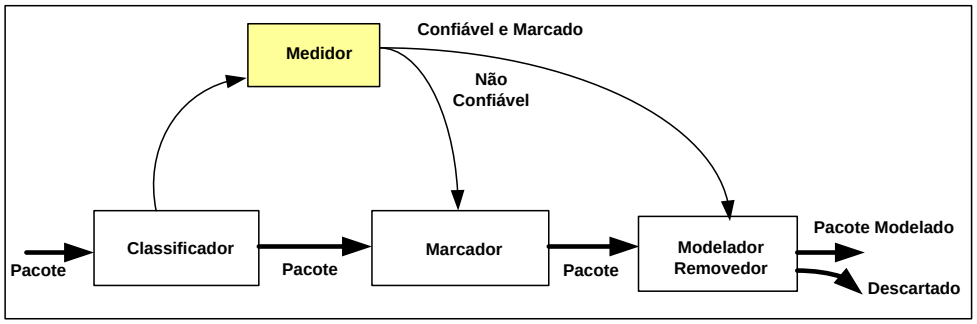


Figura 46: Condicionador de Tráfego de *DiffServ*.

A seguir, têm-se os passos do funcionamento do condicionamento de tráfego:

1. No primeiro passo é feita a classificação dos pacotes. Neste passo são examinados os pacotes para decidir qual regra deve ser executada, e na seqüência é definido o valor de DSCP ou EXP para MPLS. Ao classificar pacotes MPLS deve ser comparado com o valor EXP mais externo na pilha de rótulos. Não há como examinar além do cabeçalho MPLS sobre o IP e realizar qualquer comparação ou modificação desse pacote. No MPLS não se pode fazer a comparação com o valor do rótulo no topo da pilha, e nem com o TTL.

2. No segundo passo é feita medição ou policiamento. Este envolve a medição do tráfego onde é verificada a conformidade com parâmetros de tráfego e passa o resultado para o marcador e condicionador de pacotes, para disparar uma ação específica para pacotes que estão fora ou dentro do perfil definido. O policiamento é realizado na borda da rede. Assim, os pacotes que chegam na rede normalmente são pacotes IP. Contudo, sob alguns cenários, é possível receber pacotes com rótulos MPLS na borda da rede, como por exemplo, *Carrier Supporting Carrier* – que é um provedor recebendo pacotes com rótulos MPLS de um cliente.

3. No terceiro passo é feita a marcação dos pacotes pelo campo DSCP escrevendo e sobreescrendo o valor desse campo. A configuração da marcação está ligada ao policiamento aplicado ao tráfego, pode-se, por exemplo, fazer um mapeamento entre o DSCP do pacote IP e os bits EXP do MPLS, conforme pode ser visto nas figuras 47 e 48:

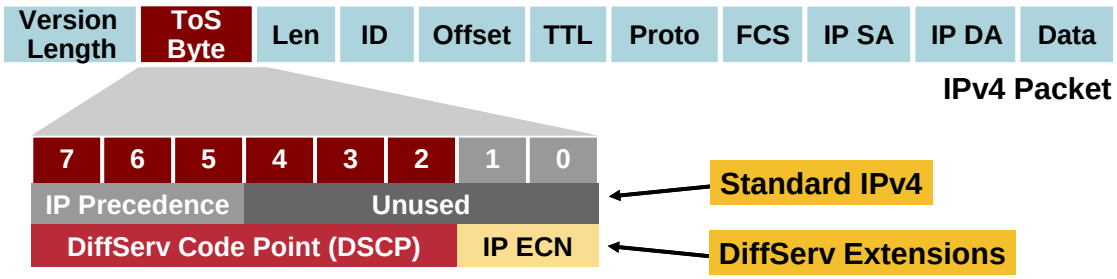


Figura 47: Marcação no Cabeçalho IP – Campo ToS [34].

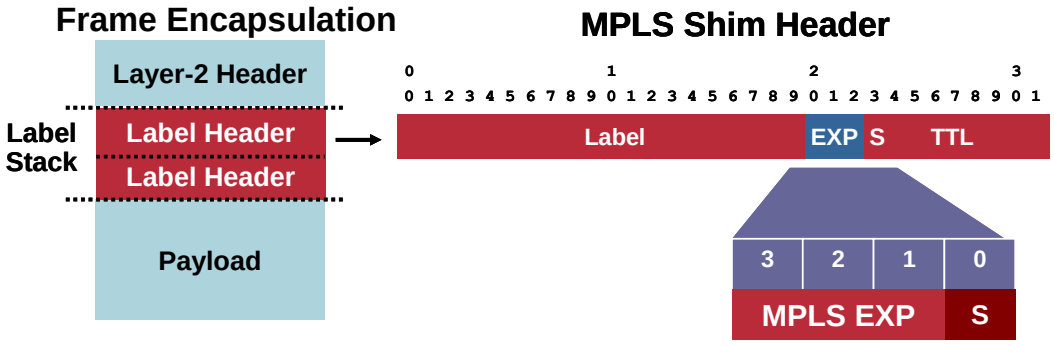


Figura 48: Marcação no Cabeçalho do MPLS – Campo EXP [34].

4. No quarto passo, condicionamento de pacotes ou moldagem é feita a aplicação de algoritmos de enfileiramento com objetivo de atrasar alguns pacotes para que os mesmos permaneçam em conformidade com o perfil definido, ou descartar

pacotes que excederam o perfil definido. O congestionamento é controlado pelo algoritmo de filas de pacotes onde esses são classificados no recebimento do roteador.

A interpretação do campo ToS do cabeçalho do pacote IP, tornou-se obsoleta conforme explicado na sessão 2.1 desta dissertação. Esse campo agora é chamado de DSCP, mas continua ocupando o mesmo espaço dentro do cabeçalho IP e ainda é frequentemente referenciado como ToS [41].

As RFCs 2474 e 2475 [46][47] do *DiffServ* redefiniram o *byte* de ToS inteiro. O *byte* de ToS agora contém 6 *bits* de informação, que declaram o tratamento desejado do pacote definido conforme DSCP. Os dois últimos *bits* CU (*Currently Unused*) atualmente inutilizados no campo *DiffServ* não foram definidos dentro da arquitetura do *DiffServ*; eles tem sido utilizados como *bits* ECN (*Explicit Congestion Notification*) [48] para notificação de congestionamento, conforme descrito na RFC 3168 [49]. O campo DS contém dois subcampos, como pode ser visualizado na figura 49.

Figura 49: O campo DS do IP [3].

O DSCP é um subcampo de 6 bits que define o PHB (*Per-Hop Behavior*). O PHB contém as informações do tipo de tratamento que o pacote deve obter em um determinado roteador. No campo DS são codificadas as classes para os serviços diferenciados. A IANA (*Internet Assigned Numbers Authority*) é responsável por controlar a alocação dos valores de DSCP. Além do valor para o melhor esforço e dos sete valores que correspondem à precedência de enfileiramento do ToS, outros treze valores são definidos. Doze dos valores são usados para representar os PHBs do AF (*Assured Forwarding*) conforme a RFC 2597[50]. A classe AF consiste de um grupo PHB com serviços especificados em termos de largura de banda relativa disponível e políticas de descarte de pacotes. O serviço AF é composto de várias sub-classes de serviço, que possuem diferentes níveis de precedência em relação ao descarte de pacotes. É indicado para as aplicações que não são sensíveis ao atraso e requerem garantia de banda (DSCP = “001,” “010,” “011”, ou “100”).

O décimo terceiro valor do DSCP padronizado é definido na RFC 3246 [51] para representar o PHB EF (*Expedited Forwarding*). A classe EF oferece um serviço de redes com baixa perda, baixo atraso, baixo *jitter* e banda garantida. É indicado para aplicações de tempo real. Para indicar PHB EF é utilizado apenas um DSCP:101110. É importante observar que, apesar deste PHB ser fácil de definir, um serviço baseado em

PHB EF necessita de uma cuidadosa coordenação entre policiamento, conformação e escalonamento nos caminhos que os pacotes EF poderão seguir. Pode-se observar na tabela 22 as classes padronizadas pela IANA.

Tabela 22: Classes Padrão do DSCP

Default PHB Codepoint (melhor esforço)	**0 (000000)**			
EF PHB	**EF/DSCP46 (101110)**			
AF PHB	**Classes**	**Precedência de descarte baixa**	**Precedência de descarte média**	**Precedência de descarte alta**
	AF 1	**AF11=DSCP10 (001010)**	**AF12=DSCP12 (001100)**	**AF13=DSCP14 (001110)**
	AF 2	**AF21=DSCP18 (010010)**	**AF22=DSCP20 (010100)**	**AF23=DSCP22 (010110)**
	AF 3	**AF31=DSCP26 (011010)**	**AF32=DSCP28 (011100)**	**AF33=DSCP30 (011110)**
	AF 4	**AF41=DSCP34 (100010)**	**AF42=DSCP36 (100100)**	**AF43=DSCP38 (100110)**

Os antigos roteadores que usavam o campo do ToS (IP *Precedence*) não podem se relacionar com sucesso com os roteadores que utilizam o DSCP, pois o significado dos *bits* podem entrar em conflito ou causar confusão. Em particular, os *bits* no campo ToS tinham significados muito específicos, enquanto aqueles DSCP permitem a definição de 64 classes diferentes, que podem ser aplicadas aos pacotes. Para fornecer compatibilidade com a antiga definição de precedência do campo ToS, são utilizados *Class Selector Codepoints*, que são definidos de maneira que seus PHBs se aproximem dos tratamentos de pacotes definidos pelos níveis de precedência IPv4. Na tabela 23 é possível visualizar os níveis de precedência e os *Class Selector Codepoints* equivalentes.

Tabela 23: *Class Selector Codepoints* [5].

Class Selector Codepoint	Precedência IPv4	Função	DSCP (decimal)	IP Precedência (decimal)
000000	000	*Routine*	0	0
001000	001	*Priority*	8	1
010000	010	*Immediate*	16	2
011000	011	*Flash*	24	3
100000	100	*Flash Override*	32	4
101000	101	*Critical*	40	5
110000	110	*Internet Control*	48	6
111000	111	*Network Control*	56	7

Com a introdução do DiffServ na rede esse ajuda ao MPLS-TE a garantir banda, com o tráfego mapeado no DSCP específico ou precedência do IP que podem ser mapeados para túneis MPLS-TE específicos. Os túneis podem ser feitos, opcionalmente, como túneis que garantem banda

5.3.2 *DiffServ* MPLS

A extensão da aplicação da arquitetura *DiffServ* em IP QoS, especificada na RFC 2475 para o MPLS QoS, foi definida na RFC 3270 [9]. A RFC 3270 [54] define procedimentos e extensões de protocolo para suporte completo dos serviços diferenciados sobre o MPLS descrevendo como os LSPs podem ser configurados com prioridades especificas e como se pode atribuir prioridade aos pacotes dentro de um determinado LSP. Um grupo PHB passou a ser denominado PSC (PHB *Scheduling Class*). Um grupo *DiffServ* BAs (*DiffServ Behavior Aggregates*) que corresponde a uma PSC recebeu a denominação OA (*Ordered Aggregate).*

Se cada LSP que está configurado estiver associado a uma *DiffServ Ordered Aggregate*, então todo o tráfego de um mesmo LSP pode ser atribuído a um PSC com as mesmas preferências de descarte em um LSR. Isso permite que tráfego de LSPs diferentes sejam diferenciados em LSR de trânsito. Se o LSR de ingresso colocar

tráfego nos LSPs de acordo com os *DiffServ* BAs, então os recursos dentro da rede MPLS podem ser alcançados [6].

Os LSPs são estabelecidos para transportar tráfego associado a OAs específicos associando a classe *DiffServ* a um rótulo, que são chamados L-LSPs.

Para que esse processo funcione, é necessário que a relação entre L-LSP, a PSC e a preferência de descarte seja conhecida em cada LSR de trânsito.

A solução definida pela RFC 3270 possibilita flexibilidade aos provedores de acesso em dois aspectos operacionais.

Em primeiro lugar, os provedores de acesso podem selecionar, a seu critério, diferentes formas de associar diferentes classes de serviço a LSPs de seus domínios *DiffServ*. Podem, por exemplo, associar cada classe de serviço a um LSP especifico, ou pode, alternativamente, associar o conjunto ou subconjunto de classes de serviço a um único LSP.

Outra forma de permitir flexibilidade aos provedores de serviços é mediante a liberdade de escolha da forma de prover proteção dos LSPs com aplicação do FRR (*Fast Rerounting*).

Usando MPLS, é possível estabelecer esquemas de proteção de tráfego que protejam LSPs individuais ou grupos de LSPs. Um modelo para isso seria proteger apenas o tráfego com certas características de PSC e descarte.

Duas informações precisam ser sinalizadas pelo protocolo de engenharia de tráfego do MPLS para que esse processo funcione. Primeiro, todos os LSRs no LSP precisam estar cientes de que podem interpretar os *bits* EXP ao indicarem uma PSC e, segundo, os LSRs precisam ter um entendimento comum dos significados dos *bits* EXP e o "*ranking*" das PSCs que codificam. Entretanto, deve ser observado que, assim como *DiffServ*, não há obrigação de um LSR priorizar os pacotes da mesma maneira ou mesmo de manipular os *bits* EXP.

5.4 MPLS-TE

O MPLS-TE - acrônimo de *Multiprotocol Label Switching Traffic Engineering* - é uma tecnologia que implementa engenharia de tráfego em redes IPs ao permitir o estabelecimento de caminhos alternativos nestas redes - diferentes dos caminhos definidos pelo protocolo IGP - baseado em critérios de recursos disponíveis, métricas

sensíveis ao atraso ou, por exemplo, características físicas do enlace como velocidades diferentes.

Para exemplificar o funcionamento da engenharia de tráfego considere a topologia mostrada na figura 50. Suponha que as conexões entre os roteadores tenham a mesma métrica IGP, portanto o tráfego originado no roteador R1 em direção ao roteador R6 utilizará o caminho R1➔R2➔R4➔R6 definido pelo IGP mesmo que este esteja congestionado, enquanto o caminho alternativo R1➔R3➔R5➔R4➔R6 pode estar subutilizado. Para aproveitar o caminho alternativo T2, poderiam ser criados dois túneis (T1 e T2) TE entre os roteadores R1 e R6, para que houvesse balanceamento de carga entre os mesmos. Assim o tráfego será encaminhado para caminhos diferentes ou alternativos aos caminhos definidos pelo IGP.

A ativação desses caminhos ocorre pela configuração de túneis TE unidirecionais, sendo que a origem do túnel é denominada de *Headend* e o destino é denominado de *Tailend*, conforme figura 50:

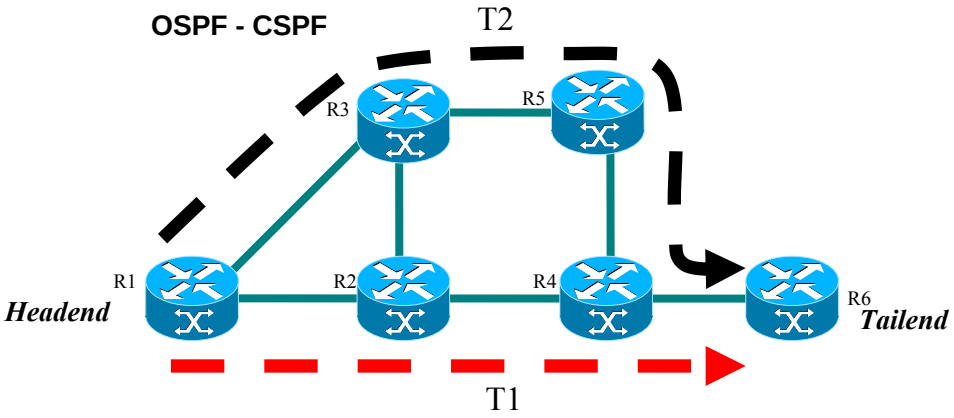


Figura 50: Caminhos com MPLS -TE.

O encaminhamento do tráfego nos roteadores será baseado na comutação de rótulos que, por sua vez, foram mapeados utilizando o protocolo RSVP (*Resource Reservation Protocol*). Esse protocolo foi originalmente concebido para ser utilizado como um mecanismo de sinalização para a arquitetura de QoS, denominada de Serviços Integrados (*IntServ*), na qual a aplicação do cliente sinaliza na rede a reserva de banda necessária para a mesma. O desenvolvimento desse protocolo permitiu que ele fosse utilizado como um mecanismo de divulgação de rótulos para a aplicação de MPLS-TE.

Antes do estabelecimento do túnel, o roteador *Headend* precisa determinar o caminho que será utilizado. Esse processo, denominado *Constrained* SPF (CSPF), é similar ao SPF (*Shortest Path First*) do OSPF e determina o caminho do túnel se

baseando na informação de banda disponível e reservável em cada enlace do *backbone*, na métrica do TE, e na característica de cada enlace denominada de afinidade ou *affinity*. Essas informações são divulgadas pelo protocolo OSPF ou IS-IS que, por sua vez, foram adaptados para divulgá-las. No OSPF, tais informações são divulgadas utilizando o LSA tipo 10, enquanto que, no IS-IS são divulgadas utilizando o TLV tipo 22.

De posse dessas informações, o roteador *Headend* executa o CSPF, explicado no capítulo 3 desta dissertação, e determina o caminho do túnel MPLS-TE de acordo com os requisitos especificados no mesmo. O caminho utilizado pelo túnel pode ser definido automaticamente através do CSPF ou pode ser definido um caminho explícito indicado e os nós que serão incluídos ou excluídos entre o *Headend* e o *Tailend*. Uma vez determinado o caminho, o túnel é estabelecido através da sinalização RSVP mediante as mensagens *Path* e *Resv*.

O roteador *Headend,* conforme figura 51, envia uma mensagem *Path* ao longo do caminho determinado pelo CSPF em direção ao roteador *Tailend*. Quando um roteador recebe uma mensagem *Path*, ele verifica se há banda suficiente para estabelecer o túnel. Este processo é denominado de controle de admissão. Quando a mensagem *Path* chega ao roteador *Tailend,* ele gera uma mensagem *Resv* em direção ao *Headend* com o intuito de indicar que a reserva de banda foi estabelecida e divulgar os rótulos a serem utilizados pelos roteadores *Upstream*. Quando a mensagem *Resv* chega ao roteador *Headend,* o túnel é ativado e está pronto para encaminhar o tráfego. A reserva de banda estabelecida pelo túnel TE atua no plano de controle e, consequentemente, não há garantia de banda no plano de encaminhamento. Isto significa que a reserva é utilizada como sinalização e controle de admissão para o tráfego que será encaminhado pelo túnel.

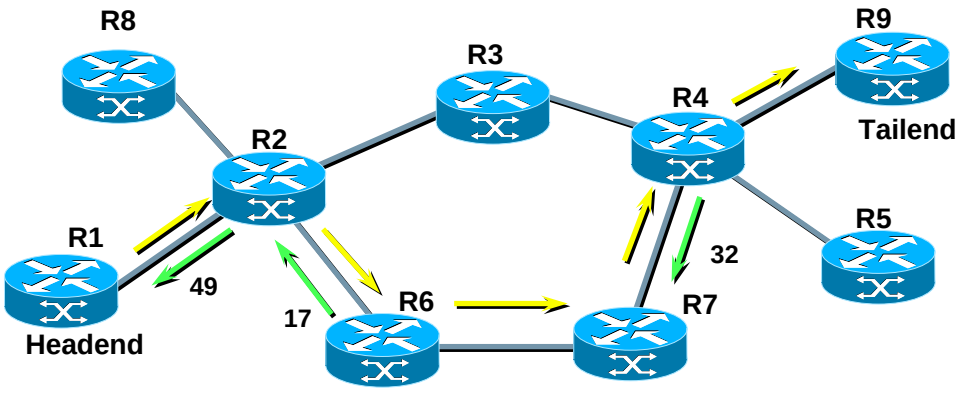


Figura 51: Estabelecendo o túnel MPLS -TE.

5.5 GMPLS

O *Generalized* MPLS é conjunto de extensões aos protocolos de sinalização de engenharia de tráfego MPLS e aos protocolos de roteamento de engenharia de tráfego, com objetivo de promover um conjunto padronizado e comum para controlar as redes de núcleo. GMPLS é desenvolvido sobre MPLS porque as noções de comutação são muito semelhantes e devido à vantagem de potencializar a reconhecida tecnologia MPLS.

A sinalização GMPLS é desenvolvida sobre a sinalização MPLS-TE. Isso não é apenas uma conveniência que reduz a quantidade de novo desenvolvimento de protocolo necessário, mas reflete o fato de o GMPLS ser um conceito advindo do MPLS-TE e usar muitos dos mesmos termos e conceitos.

O MPLS possui dois protocolos de sinalização, ambos tendo sido estendidos para o uso no GMPLS. O GMPLS é uma tecnologia de engenharia de tráfego.

Com base no fato de que no MPLS o plano de controle ser totalmente dissociado do plano de dados, surgiu no IETF a idéia de se estender o uso do plano de controle do MPLS, com as devidas adequações, para generalidade de tecnologias de transmissão de dados, incluindo-se aquelas que se baseiam em comutação por circuitos [9].

O GMPLS se utiliza do mesmo conceito de comutação de rótulos do MPLS, sendo que o seu objetivo é integrar as tecnologias TDM, WDM com a comutação de pacotes do MPLS. Deve ser observado que no GMPLS a comutação de pacotes é apenas um tipo de GMPLS. Ele encontra-se em fase de implantação, havendo uma grande expectativa quanto ao seu uso futuro.

As extensões do MPLS para GMPLS definidas nas RFCs 4203 e 5063 relativas as extensões dos protocolos respectivamente OSPF-TE e RSVP-TE não são, contudo, suficientes para suportar certas capacitações requeridas pelo plano de controle para pleno funcionamento de uma ASON (*Automatically Switched Optical Network*). O plano de controle das ASONs foi especificado segundo a recomendação G.8080 do ITU-T. Para a extensão da aplicação dos processos de roteamento do GMPLS no sentido de atender essa recomendação do ITU-T, o IETF emitiu a RFC 4258 que trata esse assunto.

5.6 Análise critica ao MPLS

O MPLS com todos esses serviços descritos até aqui possui algumas desvantagens como as que seguem:

- o MPLS introduz um *overhead* adicional (um cabeçalho de calço de 4 *bytes*) em cada pacote;
- os usuários compartilham a responsabilidade de roteamento com a operadora de telecomunicações;
- ainda existe a questão de segurança, pois como há uma conexão de camada 3 entre o cliente e a operadora de telecomunicações, essa tem acesso a tabela de roteamento dos clientes.
- cada pacote recebido precisa ser examinado para determinar como ele deve ser comutado, isso pode limitar significativamente o *throughput* dos comutadores de pacotes.

5.7 Considerações Finais

Neste capítulo foram estudados os serviços que o MPLS podem proporcionar para os *backbones* das operadoras de telecomunicações. Alguns dos serviços já existiam em outros tipos de tecnologia como, por exemplo, ATM, *Frame Relay*. O MPLS proporciona ainda aplicação da engenharia de tráfego. O MPLS está em constante desenvolvimento e tem grande aplicação não só em *backbones*, como também no acesso a rede das operadoras de telecomunicações, através do *pseudowire*. Bem como já aponta para um futuro de uma tecnologia do GMPLS.

Foi apresentado o conceito do MPLS-TE, que será tratado em detalhes no capítulo 6 desta dissertação, pois esse é o foco principal.

Capítulo 6 - Engenharia de Tráfego com MPLS

Este capítulo introduz a engenharia de tráfego, que é utilizada nesta dissertação, onde serão tratados os pontos referentes ao encaminhamento do tráfego de dados por diversos percursos, tornando possível a utilização dos diversos recursos de rede que estejam subutilizados pelo protocolo de roteamento dinâmico.

Um dos maiores problemas numa rede é que as rotas preferidas tendem a convergir pelos caminhos de maior banda. Tal decisão causa o desperdício de recursos de forma a haver muito tráfego em poucos enlaces, enquanto outros enlaces permanecem ociosos. Uma premissa importante da engenharia de tráfego é distribuir o tráfego por meio dos enlaces disponíveis para garantir que a carga seja dividida segundo critério e esse é um dos objetivos desta dissertação, comprovar a eficácia do MPLS-TE. Um fato importante é que a engenharia de tráfego está utilizando cada vez mais o MPLS [39].

A engenharia de tráfego também pode ser realizada em uma base de fluxo-a-fluxo por intervenção direta, permitindo que o operador monitore a rede e redirecione o tráfego. Esse monitoramento pode ser realizado por gráfico gerado pelo protocolo de gerenciamento como, por exemplo, o SNMP (*Simple Network Management Protocol*).

Nesta dissertação utilizou-se gráficos gerados a partir de um *software* de gerência de rede com objetivo de comprovar a eficiência da implementação do MPLS-TE. Com isso, a engenharia de tráfego otimiza os *backbones* das operadoras fazendo um melhor uso dos recursos disponíveis nos enlaces. Utiliza-se cada vez mais o MPLS para atender às necessidades de tunelamento.

A boa engenharia de tráfego é essencial para a eficiência dos *backbones* das operadoras de telecomunicações. Assim como os *backbones* devem suportar alta utilização da capacidade de transmissão, as redes devem ser bastante robustas, de forma que possam suportar falhas de enlaces ou de um roteador.

O MPLS-TE pode trazer benefício para todas as tecnologias e ou aplicações que requerem banda, atraso ou variação dos níveis (*jitter*) de eficiência no mapeamento de fluxos de recursos. A aplicação do MPLS-TE não cria banda e não reduz os problemas com insuficiência de recursos na rede, mas ajuda no mapeamento mais eficiente desses

recursos, como por exemplo, o mapeamento de caminhos redundantes, como é apresentado nesta dissertação.

6.1 Engenharia de Tráfego

A engenharia de tráfego há muito tempo é um conceito familiar para os urbanistas de cidades e engenheiros de segurança de estradas – eles se preocupam em como conseguir os melhores fluxos de veículos, por ruas congestionadas, com o menor número de acidentes. Um planejador de estradas se preocupa com os efeitos de junções de estradas, bifurcações no fluxo de tráfego e retornos. Quando uma nova estrada é construída, uma única pista será suficiente, ou será necessária uma rodovia de oito pistas? Quais são os limites de velocidade ideais para definir e sincronizar os semáforos? É possível priorizar certos tipos de tráfegos sem causar mais interrupções numa área de menor prioridade?

A maioria desses conceitos se aplica as redes por comutação de pacotes, pois os dados trafegam livremente encapsulados em pacotes. Um pacote de dados é análogo a um veículo, os enlaces podem ser comparados com as estradas, e os entroncamentos são comutadores e roteadores dentro de uma rede.

Ao lidar com crescimento e a expansão da rede, existem dois tipos de engenharia – engenharia de rede e engenharia de tráfego.

A engenharia de rede é a manipulação da rede para ajustar seu tráfego. Essa engenharia normalmente é feita em uma escala bastante longa, pois o tempo inicial para a instalação de novos circuitos ou equipamentos pode ser grande.

A engenharia de tráfego é a manipulação do tráfego para se ajustar à rede. Não importa o quanto se tente, o tráfego da rede nunca corresponderá a 100% da previsão. Às vezes a taxa de crescimento do tráfego excede todas as previsões, e não é possível fazer um *upgrade* da rede com rapidez suficiente. Muitas vezes, um evento repentino aumenta o tráfego de maneira que nunca poderia ter sido planejado. Em alguns casos a interrupção de um circuito como, por exemplo, um enlace óptico do tipo DWDM de 10 Gbps transportando dados de acesso à Internet de uma operadora de telecomunicações causa o aumento do tráfego nos outros enlaces.

De uma forma geral, embora ocorra um crescimento rápido do tráfego, eventos repentinos e a interrupção de enlaces da rede podem causar grandes demandas de

largura de banda em um local, mas ao mesmo tempo existem enlaces na rede que são pouco utilizados.

A engenharia de tráfego é amplamente utilizada pelas operadoras de telecomunicações para analisar a ocupação e o fornecimento de circuitos de voz no modelo TDM. Elas criam um modelo estatístico e então é aplicado ao padrão de tráfego para fazer prognóstico e estimativas. Para se adequar ao mercado de voz em pacotes e fornecimento de acesso a Internet, elas já estão implementado protocolos que possam fornecer o serviço de TE para *backbones*.

A Engenharia de tráfego é um importante serviço na operação de grandes *backbones*, pois permite direcionar o tráfego da rede para caminhos diferentes dos que foram estabelecidos por um roteamento IP convencional, distribuindo melhor o tráfego, evitando pontos de congestionamento e otimizando a utilização de recursos de rede. A engenharia de tráfego, em seu núcleo, é a arte de movimentar o tráfego de modo que o tráfego de um enlace congestionando seja movido para a capacidade não usada de outro enlace [5].

A engenharia de tráfego de forma alguma é algo especifico do MPLS; ela é uma prática geral e pode ser implementada por algo tão simples quanto mexer nos custos das interfaces de um roteador IP, ou algo tão complexo quanto rodar uma malha completa de PVC (*Private Virtual Circuit*) ATM e reotimizar caminhos do PVC com base nas demandas do tráfego que atravessa. A engenharia de tráfego com MPLS é uma tentativa de se obter o melhor das técnicas de engenharia de tráfego orientadas à conexão e mescladas com roteadores IP.

TE não resolve o congestionamento da rede temporariamente que é causada por rajadas de tráfego. Este tipo de problema pode ser tratado por uma expansão da capacidade ou através de técnicas clássicas, como por exemplo, algoritmos de filas.

O congestionamento de uma rede pode ser ocasionado por insuficiência total de recursos, ou pela má distribuição do tráfego na rede [9]. O tratamento dessa última situação quando, por exemplo, alguns enlaces da rede que estão congestionando enquanto outros enalces estão sendo subutilizados. Esta dissertação trata da otimização dos recursos de transmissão estabelecendo túneis com MPLS. Como foi apresentado no capítulo 3 desta dissertação os protocolos convencionais de roteamento impõem limitações para o uso de TE em redes, por sua inadequação para essa funcionalidade.

6.2 Engenharia de Tráfego sobre MPLS

O roteamento baseado no IP de destino não provê qualquer mecanismo para balanceamento de carga por caminhos redundantes. Nesse encaminhamento tem-se uma superutilização dos enlaces principais e de maior banda, já os enlaces redundantes são subutilizados. Analisando a rede da figura 52, onde o enlace principal possui uma banda de 10 Gbps que é superior em relação ao enlace redundante de banda de 2,5 Gbps, é verificado que o roteamento IP tradicional, apenas baseado no destino, terá o encaminhamento por apenas o enlace de maior banda, como por exemplo, o tráfego entre as redes dos pontos de presença A e B.

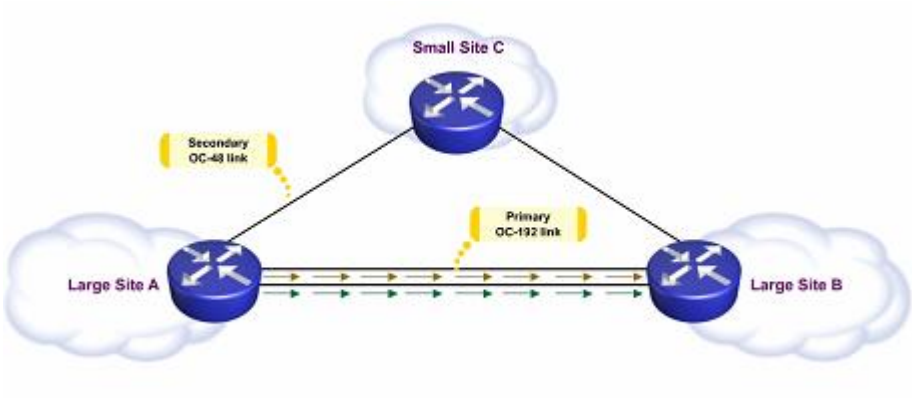


Figura 52: Encaminhamento IP tradicional [7].

As políticas de roteamento e balanceamento de carga baseado no encaminhamento dos pacotes podem ser empregadas ou é possivel se utilizar de outros parâmetros, mas isso não ocorre em redes com alto volume de tráfego devido às limitações de desempenho.

Há algumas limitações distintas do modo como a engenharia de tráfego pode ser realizada usando basicamente o protocolo IP. Se as regras de encaminhamento forem modificadas, então os roteadores na rede terão de ser mantidos sincronizados, e todos deverão operar com o mesmo nível de função, ou então poderão acontecer *loops* e maior congestionamento. Como já foi explicado no item 4.7, o roteamento baseado na origem do IP é limitado tanto pela quantidade de nove *hops*, quanto pelo fato de que nem todos os roteadores admitem o roteamento baseado na origem de uma maneira consistente. Como solução para esses problemas, o MPLS apresenta o roteamento explícito [5]. O MPLS pode ser utilizado para criar túneis de engenharia de tráfego com base na análise do tráfego e com objetivo de fornecer balanceamento de carga entre caminhos de

diferentes taxas de transmissão como apresentado na figura 53. Dessa forma, a engenharia de tráfego está utilizando cada vez mais o MPLS para atender as suas necessidades de tunelamento.

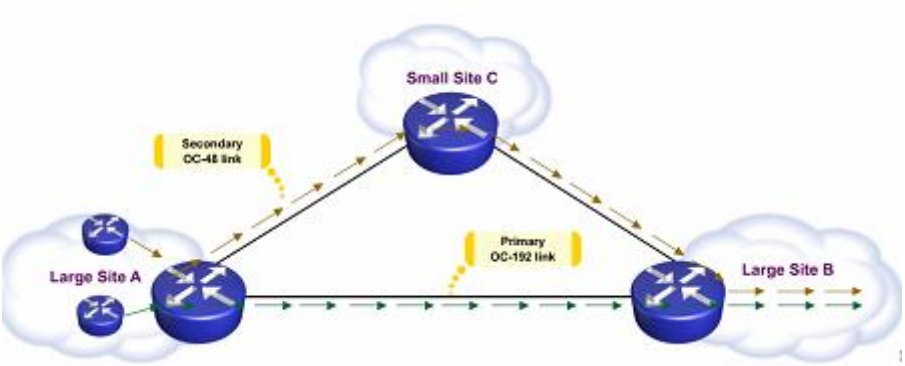


Figura 53: Balanceamento de carga com MPLS-TE.

O MPLS possui muitos componentes que o tornam atraente para uso em uma rede com engenharia de tráfego. Assim, é possível enumerar os seguintes aspectos:

- O MPLS tem a capacidade de estabelecer um LSP que segue um caminho diferente do oferecido como “preferido” pelo protocolo de roteamento.
- Os recursos dentro da rede podem ser reservados dinamicamente, conforme os LSPs são estabelecidos, e podem ser atualizados dinamicamente, conforme mudam as necessidades dos LSPs, para que esses fluxos de tráfego possam ter garantia de nível e qualidade de serviço.
- O tráfego pode ser ordenado para LSPs “paralelos”, ou seja, vários LSPs podem ser estabelecidos de origem e destino, e o tráfego pode ser distribuído entre os LSPs de acordo com qualquer número de algoritmos. Os LSPs “paralelos” podem tomar caminhos significativamente diferentes por meio da rede.
- Os recursos de rede podem ser automaticamente gerenciados com novos LSPs configurados para atender as exigências imediatas da rede e com recursos liberados novamente quando os LSPs antigos não são mais necessários e liberados.
- Procedimentos de recuperação podem ser definidos descrevendo como o tráfego pode ser transferido para LSPs alternativos no

caso de uma falha e indicando como e quando LSPs de *backup* e espera devem ser configurados e roteados.

Na engenharia de tráfego é determinado o caminho por meio da rede que diversos fluxos de dados seguirão.

Com TE as operadoras de telecomunicações podem oferecer um *backbone* para aos seus clientes com mais eficiência.

6.2.1 Extensões do OSPF para TE

A RFC 3630[40] define as extensões do protocolo OSPF versão 2 para engenharia de tráfego aplicáveis a redes IP, sendo também totalmente válidos para o MPLS-TE. O OSPF-TE (*Open Shortest Path First Traffic Engineering*) inclui o conceito de LSAs (*Link State Advertisements*) opacas. Eles permitem que os roteadores compartilhem informações privadas ou proprietárias pela rede de uma maneira interoperável. Foram definidos três tipos de LSA opacos:

1. LSA opaco tipo 8 – abrange apenas um link;
2. LSA opaco tipo 10 – abrange uma área;
3. LSA opaco tipo 11 – abrange um sistema autônomo OSPF.

Os roteadores que não "entendem" os LSAs opacos não os precisa examinar nem os usar em seus cálculos de caminho, mas precisam armazená-las e encaminhá-las. Isso provê um modo elegante de acrescentar função a uma rede de maneira compatível e é usado para acrescentar informações de engenharia de tráfego ao OSPF, de modo que os roteadores cientes da engenharia de tráfego possam descobrir a atuar sobre as informação de TE em cooperação com roteadores mais antigos, que continuam a implementar o OSPF padrão [6]. A RFC 3630 [40] engloba apenas LSA opaca tipo 10, ou seja, restringe a sua aplicabilidade ao interior de uma área OSPF [9].

Dentro do LSA opaco tipo 10 foi definido *Traffic Engineering LSA*, identificada pelo código 1. Esse LSA descreve roteadores, enlaces ponto-a-ponto e conexões para redes multi-acesso, de forma similar a um router LSA do OSPF convencional. Essas extensões, partindo da descrição da topologia e dos parâmetros de QoS da rede que suporta o TE, provêem os mecanismos de distribuição dessas informações dentro de uma área de roteamento OSPF.

O OSPF-TE descreve e define uma forma de distribuição de *extended link attributes*, que se baseia em *costraint-based routing*.

Ao contrário do OSPF convencional, onde cada *hop* realiza o roteamento de pacotes com base em tabelas de roteamento local, o OSPF-TE centraliza as informações em uma base de dados localizada no *headend* LSR, no caso do MPLS-TE, denominada TE *database*.

O formato do corpo do LSA opaco é criado por meio de construções TLV (*Type-length-Variable*), que formam uma sequência, mas também podem ser sobrepostas de modo que o tamanho de TLV de nível superior cubra toda uma série de sub-TLVs. Todos os sub-TLVs são encontrados na variável do TLV de nível superior.

Existe a possibilidade da utilização conjunta da métrica do OSPF convencional e de uma métrica TE no MPLS-TE para diferentes classes de tráfego. Uma aplicação de voz pode utilizar a métrica OSPF TE relativa a retardos e a *jitters* na rede, enquanto uma transferência de grandes arquivos pode utilizar uma métrica OSPF convencional.

6.2.2 Extensões do IS-IS para TE

Para que o MPLS-TE funcione sobre uma rede com protocolo IS-IS pré-existente, é preciso transitá-lo para uma nova versão de IS-IS, já com OSPF não é necessário transição.

As extensões de engenharia de tráfego para o IS-IS (*Intermediate System to Intermediante System Traffic Engineering*) estão descritas na RFC 3784 [22] conforme mencionado no item 3.2.1 desta dissertação.

Os procedimentos no IS-IS-TE são semelhantes aos do OSPF-TE, com algumas pequenas diferenças. Como por exemplo, o período padrão para *reflooding* periódico de informações de roteamento, que é de 30 minutos para o OSPF-TE, assume o valor de 15 minutos para IS-IS-TE.

Para o IS-IS-TE foram criados novos TLVs definidos como:

- *Extended IS Reachabulity TLV*, com o valor de tipo 22;
- *Traffic Engineering Router ID TLV*, com o valor de tipo igual a 134.
- *Extended IP reachability TLV*, com o valor de tipo igual a 135.

6.2.3 Protocolo RSVP-TE

O RSVP-TE (*Resource Reservation Protocol – Traffic Engineering*) é o protocolo apropriado para distribuição de rótulos em redes MPLS com engenharia de tráfego, que é baseado no RSVP (*Resource Resevation Protocol*). O RSVP é adequado para extensão ao mundo MPLS porque lida com reservas de recursos fim-a-fim para fluxos de tráfego, de forma muito semelhante com o MPLS com engenharia de tráfego. Por outro lado, ele não atende a todas as exigências necessárias para o MPLS – principalmente a distribuição de rótulo e controle de caminhos por meio de rotas explícitas [6].

O RSVP foi inicialmente estendido para esse tipo de aplicação pela *Cisco Systems* [7] quando ele estava desenvolvendo a comutação por TAG. Desde então, o IETF tem publicado o RSVP-TE [37].

O RSVP-TE consegue reutilizar o RSVP de maneira bastante completa. Todas as sete mensagens RSVP encontram um uso no RSVP-TE, embora o *ResvConf*, que é a mensagem para reservar os recursos ao longo do caminho seja menos significativa do que quando usado para o RSVP.

6.3 Operação do MPLS-TE

O MPLS-TE é usado para otimizar a utilização dos recursos de redes com enlaces redundantes e com velocidades diferentes. Além disso, o MPLS-TE fornece mecanismos para a recuperação rápida (*FastReRoute*) em caso de enlace ou um roteador entrar em falha. O MPLS-TE também pode ser combinado com outros mecanismos de QoS para fornecer garantias para VPNs ou classes de tráfego.

MPLS-TE combina a capacidade de engenharia de tráfego do ATM com a flexibilidade do IP. Basicamente, a ativação dos caminhos no MPLS-TE ocorre pela configuração de túneis de TE unidirecionais, sendo a origem do túnel denominada de *headend,* e o destino do túnel denominado *tailend*. O encaminhamento do tráfego pelos roteadores será baseado na comutação de rótulos, que são mapeados pelo RSVP.

O MPLS- TE permite um esquema de engenharia de tráfego, onde o roteador conhecido como *headend* do LSP pode calcular a rota de forma mais eficiente através da rede em direção ao roteador conhecido como *tailend.* O *headend* pode fazer isso se

ele conhece a topologia da rede. Ele também precisa saber a banda disponível de todos os enlaces na rede e é necessário habilitar MPLS nos roteadores para estabelecimento do LSP fim a fim. O fato da comutação de rótulos ser utilizada, e não o encaminhamento baseado em IP, permite o roteamento baseado em origem ao invés do roteamento baseado em IP de destino.

6.3.1 Atributos de Túneis MPLS-TE

Alguns dos atributos de túneis MPLS-TE são de mesma natureza que os atributos dos enlaces distribuídos pelos protocolos IGP estendidos para TE. Alguns outros atributos dos túneis MPLS-TE são configurados pelos administradores da rede, transparentemente ao protocolo de roteamento utilizado.

Podem-se listar os atributos conforme descrito abaixo:

- endereço do *tailend LSR*;
- largura de banda desejada;
- atributo de preempção;
- atributo de reotimização;
- *fast rerouting*.

6.3.2 Proteção e Restauração – FRR (*Fast Reroute*)

As redes são projetadas com um alto nível de redundância para garantir a oferta dos serviços oferecidos aos clientes. No entanto, muitas vezes, a falha em um circuito de comunicação faz com que a recuperação de um serviço gaste um tempo na ordem de dezenas de segundos, dada a quantidade de protocolos envolvidos na convergência. Este tempo de convergência é resultante, principalmente, do tempo de propagação do protocolo IGP, responsável por fazer o reroteamento rápido para contornar as falhas, efetuando a convergência na rede. Dependendo do tamanho da rede, este tempo pode levar de 5 a 10 segundos. Durante esta convergência há perda de pacotes e, conseqüentemente, uma indisponibilidade do serviço oferecido ao usuário e isto pode afetar o SLA (*Service Level Aggrement*) acordado entre o usuário e o provedor.

Fast Reroute é uma ferramenta integrante do MPLS-TE. Ela permite que circuitos e roteadores sejam protegidos pelos túneis do MPLS-TE com rápido tempo de

convergência. A proteção dos circuitos é denominada *Link Protection* e dos roteadores *Node Protection.*

6.4 Transmissão de pacotes no MPLS-TE

Com o MPLS-TE é possível forçar o encaminhamento do tráfego por caminhos redundantes que não são utilizados pelo protocolo de roteamento IGP. O MPLS-TE tem suporte para determinar o caminho na origem com base nos parâmetros adicionais, tais como QoS. Então, os túneis do MPLS-TE podem ser criados levando ao mesmo destino, mas usando caminhos diferentes. O compartilhamento de carga ao longo de caminhos de largura de banda diferentes pode ser obtido, o que resulta em uma ótima utilização de enlaces de rede.

6.5 Considerações Finais

Neste capítulo foi dado foco na engenharia de tráfego aplicada no MPLS, para tal foi necessário apresentar as extensões dos protocolos de roteamento OSPF-TE, IS-IS-TE bem como o protocolo de sinalização RSVP-TE. Foi apresentado de forma sucinta o FRR.

Não deve ser esquecido que existem dois objetivos principais na engenharia de tráfego, que são: as minimizações de retardo e *jitter* e a otimização dos recursos de transmissão.

Capítulo 7 – Desempenho do MPLS-TE em um Sistema Comercial

Este capítulo analisa o funcionamento do MPLS-TE em um *backbone* de uma operadora de telecomunicações. São feitos inicialmente testes com o MPLS e os protocolos de roteamento dinâmico sem aplicação da engenharia de tráfego com o objetivo de obter parâmetros de comparação entre uso ou não do MPLS-TE. Para isso, foi necessária a montagem de uma topologia de rede utilizando roteadores, *software* de gerência de redes, gerador de tráfego, estações de trabalho e analisador de protocolo.

Há no mercado diversos fabricantes de roteadores que disponibilizam o MPLS para uso em *backbones.* Dentre eles, pode-se citar *Cisco Systems* [7], *Huawei Techonologies, Juniper Networks e Alcatel-Lucent.* Os experimentos descritos nesta dissertação fizeram uso dos roteadores *Cisco Systems*, já que são os utilizados pela operadora de telecomunicações em análise.

7.1 *Backbone* utilizado para testes

Para os testes foi montado fisicamente o *backbone* apresentado na figura 54, que é apenas um fragmento de um ponto de presença de uma operadora de telecomunicações, suficiente para a comprovação dos itens desta dissertação. Essa topologia de rede é dividida em três camadas:

- A de acesso, representada pelos os roteadores CEs (*Customer Edges)*, os roteadores do na rede do cliente que fazem conexão com o porvedor de telecomunicações.
- A de distribuição, pelos dois roteadores PEs (*Provider Edges*), os roteadores que fazem a borda da rede puramente IP com o *backbone* MPLS da operadora de telecomunicações.
- A de núcleo, pelos dois roteadores Ps (*Providers)*, os roteadores que estão no núcleo do *backbone* MPLS da operadora de telecomunicações.

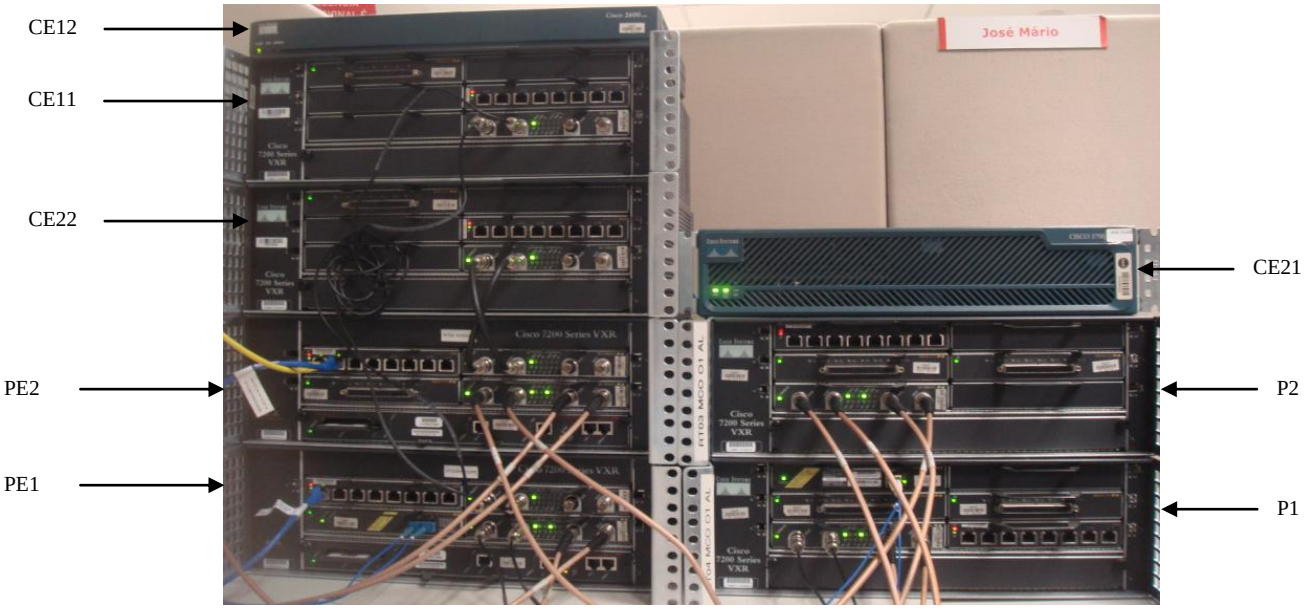


Figura 54: *Backbone* físico montado para testes.

O cenário montado na figura 54 pode ser representado logicamente pela topologia mostrada na figura 55, onde há a separação da função dos roteadores CE, PE e P utilizados para os testes.

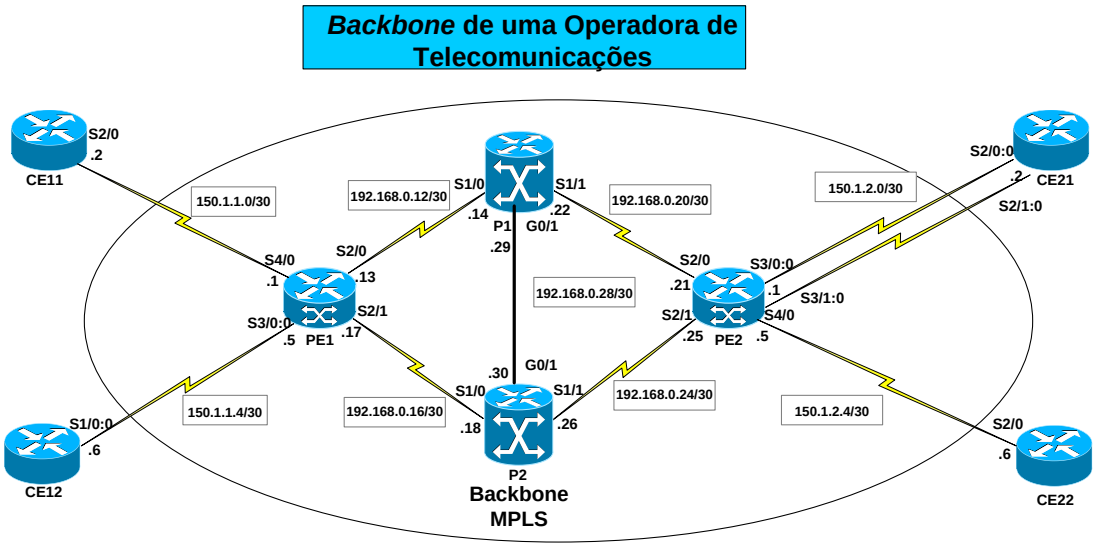


Figura 55: Fragmento de um *Backbone* de uma Operadora Telecomunicações Comercial.

Para montagem dessa topologia de rede são utilizados os seguintes roteadores listados na tabela 24.

Tabela 24: Roteadores do *backbone*.

Função	Modelo	IOS
CE11 e CE 22	*Cisco* 7206	c7200-jk9s-mz.123-17a.bin
CE12	*Cisco* 2611	c2600-is-mz.122-21b.bin
CE 21	*Cisco* 3725	c3725-js-mz.123-5d.bin
PE	*Cisco* 7204	c7200-jk9s-mz.123-17a.bin
P	*Cisco* 7206	c7200-jk9s-mz.123-17a.bin

Os *hardwares* utilizados na topologia possuem as seguintes características:

- quatro roteadores modelo *Cisco* 7206VXR (NPE-G1), com 512MB de memória RAM, 128 MB de memória PCMCIA e 16MB de memória *flash*;
- dois roteadores modelo *Cisco* 7204VXR (NPE225), com 160MB de memória RAM, 64MB de memória PCMCIA e 8MB de memória *flash*;
- um roteador modelo *Cisco* 3725 (R7000), com 256 MB de memória RAM, 32MB de memória flash;
- um roteador modelo *Cisco* 2611 (MPC860), com 50MB de memória RAM, 16MB de memória *flash*.

O sistema operacional utilizado nos roteadores da *Cisco* é chamado IOS (*Internetwork Operation System*). Nos roteadores dessa topologia foram utilizadas as versões desse sistema operacional conforme tabela 25. Além do *software* dos roteadores, é utilizado um gerador de tráfego TfGEn (*Traffic Generation*) que foi usado com o objetivo de gerar tráfego para o *backbone*. Ainda foi utilizado um *software* de gerência de redes, para obter gráficos e tabelas que comprovem os testes efetuados pelo serviço MPLS-TE. Também é utilizado um analisador de protocolo DA-340 do fabricante *Acterna*.

Nos roteadores foram instaladas placas que possuem as seguintes características:

- Nos roteadores P1, P2, PE1, PE2, CE11, CE22 foi usada a placa PA-2E3 que possui duas interfaces de E3 com taxa de transmissão de 34 Mbps no padrão Europeu. Essa placa pode ser vista na figura 56;

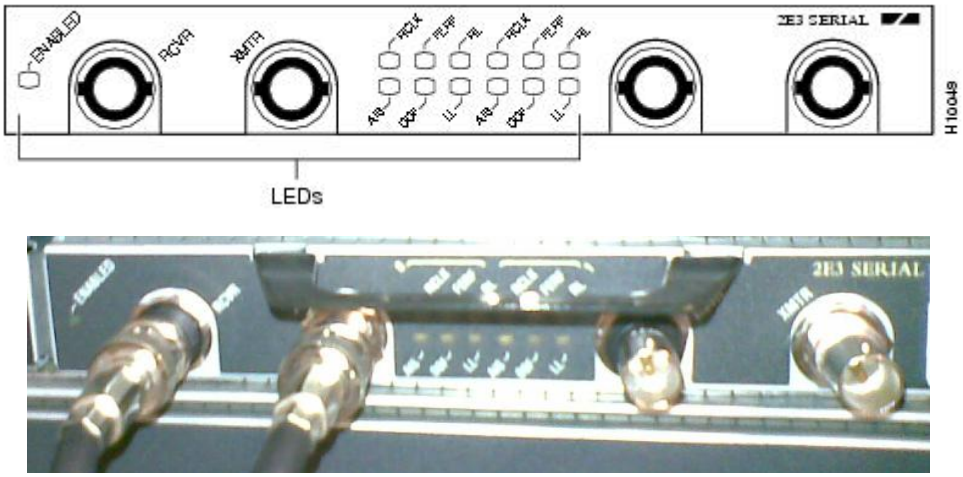


Figura 56: Placa PA-2E3.

- Na conexão entre o roteador P1 com os roteadores PE1 e PE2 utiliza-se a placa PA-2E3 (figura 56). Da mesma forma, na conexão de P2 com PE1 e PE2 foi usado esse mesmo modelo de placa. A conexão entre os roteadores P1 e P2 foi realizada via interface G*igabitEthernet* da placa NPE-G1, conforme figura 57;

Figura 57: Placa NPE-G1.

- Na conexão do PE1 com CE11 e PE2 com CE22 também foi utilizada a placa PA-2E3 (figura 56) com uma taxa de transmissão de 34 Mbps. Já na conexão dos roteadores CE12 com PE1 foi utilizada a placa PA-MC-8TE1, essa placa possui oito interfaces de E1 no padrão Europeu de 2 Mbps, foi utilizado apenas um E1 para essa conexão. Entre os roteadores CE21 e PE2 também foi utilizada a placa PA-MC-8TE1 conforme figura 58, inserida nos roteadores PE1 e PE2;

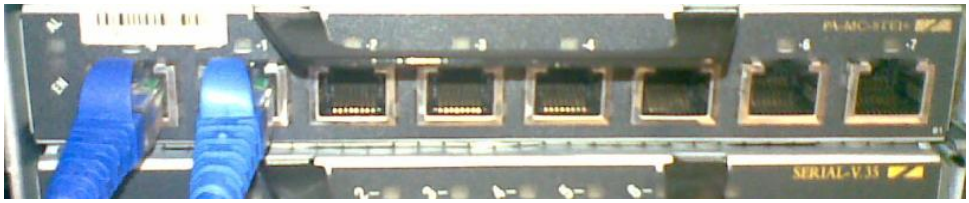


Figura 58: PA-MC-8TE1.

- No roteador CE21 foi utilizada a placa NM-2CET1-PRI, conforme a figura 59, neste caso os roteadores PE2 e CE21 estão conectados com dois E1s totalizando uma taxa de transmissão de 4 Mbps, para tal conexão foi necessário criar uma interface *multilink* PPP;

Figura 59: NM-2CET1-PRI.

- No roteador CE12 foi utilizado o modulo NM-HDV com placa VWIC-1MFT-E1, conforme figura 60, neste caso a conexão PE1 com CE1 será com apenas um E1 com taxa de transmissão de 2 Mbps.

Figura 60: NM-HDV com placa VWIC-1MFT-E1.

7.2 Escolhendo o protocolo IGP e o esquema de endereçamento IP para o *Backbone* de testes.

Nos testes do *backbone* foi utilizado o protocolo de roteamento dinâmico OSPF (*Open Shortest Path First*), mas nada impediria a utilização do IS-IS. Esses dois protocolos desempenham a função de IGP (*Interior Gateway Protocol*), sendo o OSPF o mais difundido em *backbones* de operadoras de telecomunicações. Como esta dissertação trata da análise e otimização de roteamento em *backbones* OSPF utilizando MPLS-TE, foi usado o OSPF como protocolo IGP principal.

O OSPF é utilizado para fazer o anúncio dos endereços de *loopback* (/32) e dos enlaces internos ao *backbone.* O OSPF tem convergência rápida e garante que falhas em circuitos ou dispositivos tenham rápida convergência, minimizando a interrupção no *backbone*.

7.2.1 Protocolo OSPF - Processo do *Backbone*

O OSPF é utilizado como protocolo de roteamento dessa rede. O objetivo principal do OSPF é garantir que o anúncio de todas as *loopback*s dos roteadores seja feito e que a rede se atualize rapidamente em caso de falha de circuitos ou dispositivos. O OSPF é utilizado como protocolo de infraestrutura e é através do OSPF que se garante a conectividade entre as *loopback*s, para que o MP-BGP consiga chegar aos *next-hop* necessários. O processo OSPF usado no *backbone* é o processo 1 e contém todas as rotas existentes na rede. Com o perfeito funcionamento do OSPF, o LDP será utilizado para propagar os rótulos e construir os LSPs pelos melhores caminhos possíveis com segurança e estabilidade.

7.2.2 Protocolo OSPF - Configuração em Áreas

Preservar todos os roteadores em uma única área OSPF traz grandes vantagens para o MPLS-TE [30]. Estudos desenvolvidos pela *Cisco Systems* relatam que não há grande penalização de desempenho em se manter 500 ou 1000 roteadores em uma única área, desde que se tenha estabilidade.

7.2.3 Protocolo OSPF - *Router-ID*

Para simplificar a identificação de cada roteador, o *router-id* é manualmente configurado no processo OSPF de cada dispositivo. Por padrão, o roteador seleciona, como seu *router-id*, o maior endereço IP de suas interfaces *loopback* ou, na ausência de qualquer *loopback,* o maior endereço das interfaces físicas. Para o *backbone* montado foram utilizados os IPs para as *loopbacks* dos roteadores, conforme a tabela 25.

Tabela 25: IP das loopbacks dos roteadores.

Função	IP da Loopback	Configuração da Loopback	OSPF 1
CE11	200.1.0.1	*ip address 200.1.0.1 255.255.255.255*	*NA*
CE12	200.1.0.2	*ip address 200.1.0.2 255.255.255.255*	*NA*
CE21	200.2.0.1	*ip address 200.2.0.1 255.255.255.255*	*NA*
CE22	200.2.0.2	*ip address 200.2.0.2 255.255.255.255*	*NA*
PE1	172.1.0.1	*ip address 172.1.0.1 255.255.255.255*	`router-id 172.1.0.1`
PE2	172.2.0.1	*ip address 172.2.0.1 255.255.255.255*	`router-id 172.2.0.1`
P1	172.10.0.1	*ip address 172.10.0.1 255.255.255.255*	`router-id 172.10.0.1`
P2	172.20.0.1	*ip address 172.20.0.1 255.255.255.255*	`router-id 172.20.0.1`

7.2.4 Protocolo OSPF - Cálculo de Métrica

O OSPF calcula a métrica através de uma fórmula matemática que leva em consideração a velocidade dos circuitos utilizados para atingir cada destino. O cálculo da métrica do OSPF de cada interface é realizado através da divisão de um valor de referência pelo valor da largura de banda (*bandwidth*) da interface física. O valor de referência de 100 Mbps é utilizado por *default* no OPSF versão 2. A fórmula utilizada pelo OSPF, 10^8/BW (*Bandwidth*), é simples e gera como resultado um número inteiro. O valor BW é uma medida em *bits* por segundo (bps). Utilizando-se deste valor como padrão, o OSPF utilizaria métrica 1 para todos os circuitos com 100 Mbps ou mais. Assim sendo, não há distinção entre circuitos 100 Mbps e 1 Gbps, ambos utilizados nessa rede.

Para garantir que o OSPF seja capaz de diferenciar circuitos de diferentes velocidades, o valor utilizado pela fórmula (10^8) deve ser alterado para 10^{10}, garantindo distinção de velocidade até 10 Gbps. Na configuração do OSPF, o comando "*auto-cost reference-bandwidth 10.000*" faz esta adequação. Para assegurar a coerência dos cálculos do OSPF, todos os roteadores da rede deverão receber esse novo comando.

7.2.5 Protocolo OSPF - Anúncio das rotas

Cada roteador do *backbone* é responsável por gerar anúncios de todos os *loopbacks* participantes no roteamento. Os *loopbacks* dos roteadores PE1, PE2, P1, P2,

são usados como IP de origem da sessão BGP e LDP. Todas as *loopbacks,* que são usadas como *next-hop* nos anúncios de BGP, também devem ser anunciadas via OSPF.

7.2.6 Protocolo OSPF - *OSPF Flooding reduction*

As operadoras de telecomunicações e os clientes com redes grandes sabem que o comportamento do protocolo OSPF gera uma sobrecarga de tráfego, já que a cada trinta minutos, aproximadamente, mesmo quando a topologia da rede é estável, é feito o *flooding* na rede, conforme explicado nesta dissertação no item 3.2.

Por projeto, o OSPF requer que os anúncios dos estados dos enlaces (*link-state advertisements*) sejam atualizados a cada 3.600 segundos no máximo. Por segurança, o OSPF reenvia seus LSAs a cada intervalo pouco superior a 1.800 segundos.

O OSPF Flooding reduction funciona reduzindo o *flooding* desnecessário dos LSAs já conhecidos e inalterados. Para executar essa redução, os LSAs são enviados sem um período de expiração, tornando-os assim DNA (*DoNotAge*) LSAs. Essa recomendação pode ser encontrada na RFC 4136 [61]. Foi implementada essa característica com o seguinte comando aplicado nas interfaces dos roteadores "*ip ospf flood-reduction*", pois esse tipo de tráfego é substancial e trará economia de recursos aos roteadores.

7.3 Protocolo BGP (*Border Gateway Protocol*)

O BGP (*Border Gateway Protocol*) é um protocolo robusto e escalável, suportando atualmente algo em torno de 200.000 rotas, por isso é o protocolo utilizado na Internet. Para atingir essa escalabilidade, o BGP utiliza diversos parâmetros, chamados de atributos, que definem políticas de roteamento e mantém a estabilidade do ambiente. Além desses atributos, o BGP se utiliza do CIDR (*Classless Interdomain Routing*) para reduzir o tamanho das tabelas de roteamento.

7.3.1 Sistema Autônomo para BGP

O sistema autônomo utilizado no *backbone* MPLS não terá vizinhança com a Internet, sendo, portanto, de âmbito privado. De acordo com a recomendação da

RFC1930, o intervalo a ser usado para sistemas autônomos privados deve estar entre 64512 e 65535. Será adotado, para esta dissertação, o sistema autônomo 65500.

7.3.2 Configuração do BGP

É necessário ensinar aos vizinhos do BGP, pois não existe uma descoberta automática de adjacências como no OSPF. Devido a esse motivo, os vizinhos BGP não precisam ser diretamente conectados. Para que seja estabelecida a sessão BGP é necessário configurar os paramentos nos roteadores do *backbone* da seguinte forma:

- configuração do processo BGP, como é definido o valor do AS igual a 65500, é aplicado o seguinte comando no roteador – ***router bgp 65500***. Essa configuração será aplicada apenas nos roteadores PE1 e PE2;
- endereço IP do vizinho e o número de AS remoto. O endereço IP utilizado também será o da *loopback0*, conforme listado na tabela 25. O número do AS será o 65500. O comando deverá ser aplicado nos roteadores PE1 e PE2 da seguinte forma:

No roteador PE1

router bgp 65500

neighbor 172.2.0.1 remote-as 65500

No roteador PE2

router bgp 65500

neighbor 172.1.0.1 remote-as 65500

- o sistema operacional da *Cisco* utiliza o endereço IP da interface de saída em direção ao vizinho, portanto para obtermos o IP de origem será necessário explicitar a configuração para estabelecimento da sessão BGP entre os roteadores PE1 e PE2. Dessa forma, serão aplicados os seguintes comandos:

No roteador PE1

router bgp 65500

neighbor 172.2.0.1 remote-as 65500

neighbor 172.2.0.1 update-source Loopback0

No roteador PE2

router bgp 65500

neighbor 172.1.0.1 remote-as 65500

neighbor 172.1.0.1 update-source Loopback0

7.4 Protocolo LDP (*Label Distribuition Protocol*)

O LDP é o protocolo utilizado para troca de rótulos entre roteadores MPLS, conforme explicado no capítulo 4 desta dissertação. O LDP envia mensagens de *hello* a cada 5 segundos por padrão (podendo ser alterado) utilizando *multicast* (224.0.0.2 porta 646 UDP). O vizinho MPLS que recebe a mensagem automaticamente responde abrindo uma sessão TCP com a porta de destino 646. A partir daí, os dois roteadores estabelecem uma sessão LDP através de *unicast* TCP e trocam os rótulos existentes.

Para o estabelecimento de vizinhança LDP, cada roteador utiliza o seu LDP *router-id* que funciona de forma semelhante ao *router-id* do BGP e do OSPF. No processo de vizinhança do LDP é imprescindível que o LDP *router-id* do vizinho esteja na tabela de roteamento. Caso contrário, a vizinhança não será estabelecida.

Quem garante o roteamento de todos os *router-ids* é o IGP presente em todos os roteadores do MPLS *domain*. No caso do *backbone* de testes, o processo OSPF 1 é o responsável por essa tarefa. Consequentemente, o IGP será responsável por prover a melhor rota para estabelecimento das adjacências LDP. Portanto, qualquer otimização no processo OSPF 1 é também considerada otimização no *backbone*.

7.5 Comportamento básico do MPLS no *backbone* de testes e configurações aplicadas.

Esta sessão busca explicar o cenário do laboratório implementado, conforme a topologia lógica do *backbone* apresentada na figura 54, que será utilizado para testes. No primeiro momento não será aplicado o MPLS-TE, serão apenas implemnetadas as funcionalidades do MPLS rodando com os protocolos OSPF e BGP.

O MPLS pode operar em dois modos: o *Frame-Mode* e o *Cell-Mode.* Nesta dissertação é aplicado o modo *Frame-Mode* que consiste na inserção de um rótulo entre a camada dois e a camada três de um pacote IP. Assim, os roteadores são diretamente conectados através de interfaces *frame-mode*, utilizando o protocolo PPP (*Point to Point Protocol*). Os cabeçalhos PPP e *Ethernet* exibem o rótulo sendo inserido entre os cabeçalhos da camada dois e da camada três, conforme a figura 61. Esse cabeçalho é conhecido como *shim* ou cabeçalho de calço conforme explicado no item 4.3 desta dissertação.

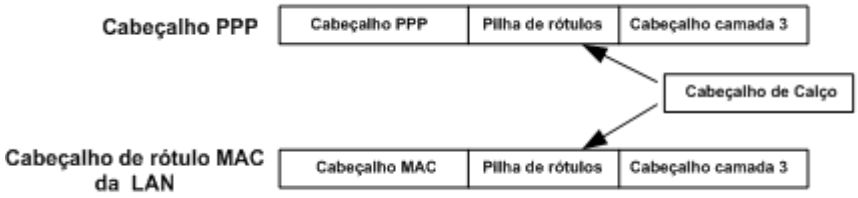


Figura 61: MPLS *Frame-Mode*.

No caso de utilizar o modo *Cell-Mode* o cabeçalho de calço é sempre visto, mesmo quando se está simplesmente conectando roteadores por PVC ATM e realizando MPLS em um ambiente clássico IP sobre ATM [5].

Para os testes desta dissertação foram configurados os roteadores PE1, PE2, P1, P2, CE11, CE12, CE22 e CE21 com as respectivas funções de *Provider Edge* (PE), *Provider* (P) e *Customer Edge* (CE), sendo que será apenas configurado o MPLS nos roteadores PE1, PE2, P1 e P2. Os testes serão feitos a partir de estações conectadas nos roteadores CEs, onde sua comunicação é estabelecida através do *backbone* MPLS, como apresentado na figura 54.

7.5.1 Testes de Conectividade e encaminhamento de Tráfego

Para os testes de conectividade do *backbone* foram configurados quatro roteadores CEs com estações conectadas em suas interfaces *Ethernet*, *Fastethernet* e *Gigabitethernet*. Foi verificada a conectividade entre as estações de trabalho com os comandos de *ping* e *trace* entre as estações.

Os testes iniciais de tráfego no *backbon*e foram feitos apenas com MPLS habilitado nos roteadores e protocolo de roteamento dinâmico OSPF.

Na primeira parte do teste na estação conectada na interface na *Gigabitethernet* do CE11 (ip 10.81.20.60) foi instalado o gerador de tráfego TfGen, conforme a figura 62, foi gerado um tráfego em direção a estação conectada no CE22 (ip 10.80.20.30) e observou-se que esse tráfego segue apenas um único caminho, conforme *trace* da figura 63. O gerador gera tráfego apenas UDP.

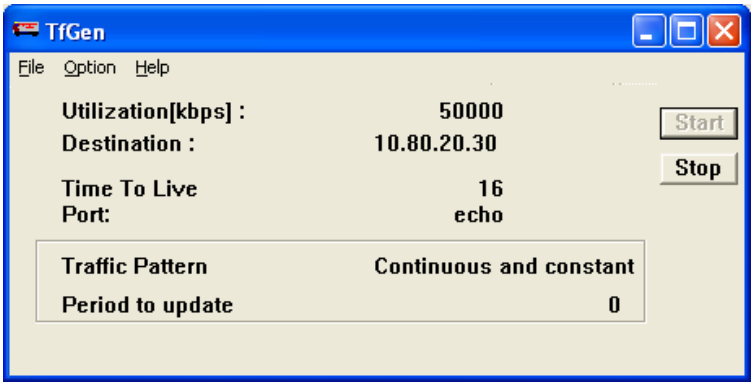


Figura 62: Tela do TfGen gerando um tráfego de 50000 kbps.

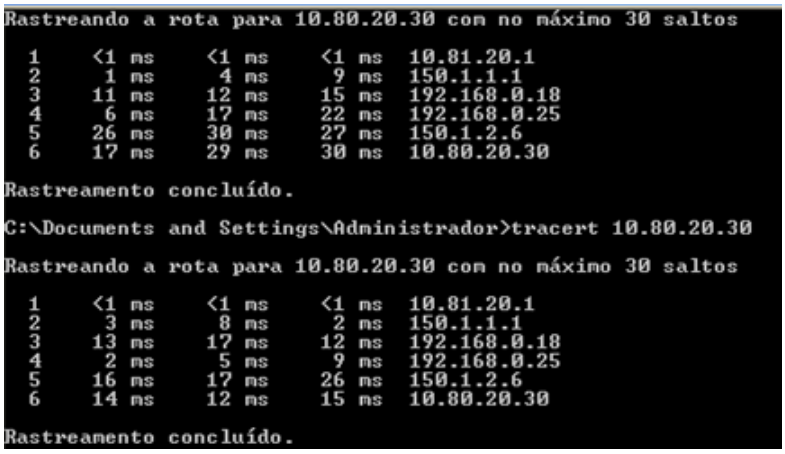


Figura 63: Trace na estação conectada no roteador CE11.

É possível observar que o caminho seguido é CE11 (interface *Gigabitethernet*0/1) ➔ PE1(interface S4/0) ➔ P2 (interface S1/0) ➔ PE2 (interface S2/1) ➔ CE22 (interface S4/0) ➔ Estação de destino. Durante este teste, os roteadores estão configurados apenas o protocolo de roteamento ativo e o MPLS.

Na estação (ip 10.80.20.30) conectada na *Gigabitethernet* do roteador CE22 foi instalado o *software* PRTG *Traffic Grapher*, conforme apresentado na figura 64, com o objetivo de gerar gráficos de utilização dos enlaces de comunicação entre os roteadores PEs e Ps. Para isso, foi necessário habilitar o protocolo SNMP no roteador PE1. Como resultado do teste, o tráfego foi encaminhado apenas pela interface de saída do roteador de P2 ➔ PE1 e o caminho P1 ➔ PE1 ficou subutilizado. Dessa é possível constar a má destribuição de tráfego pelo protoloco OSPF.

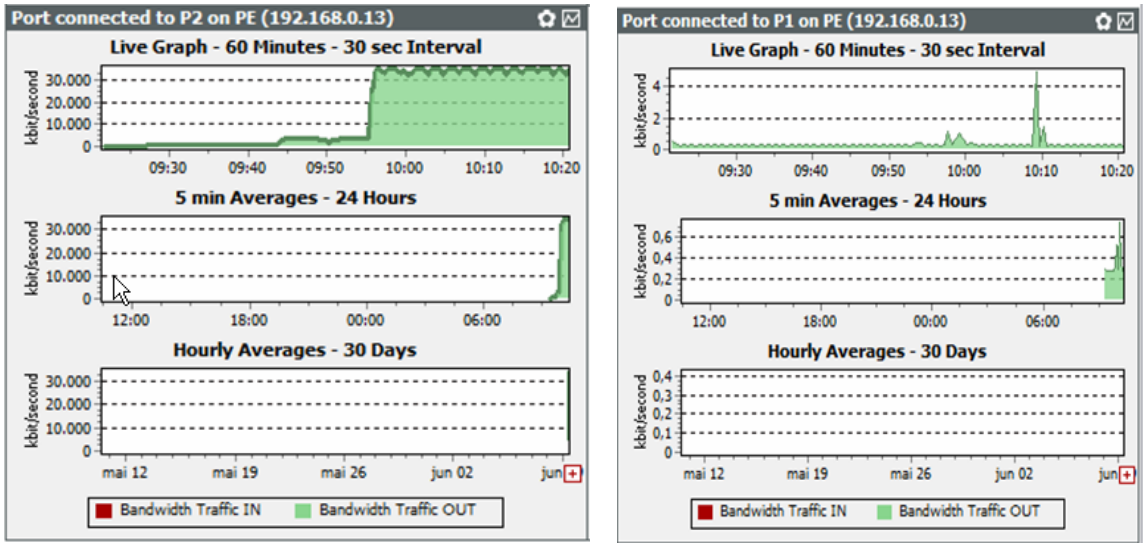


Figura 64: PRTG *Traffic Grapher* gerando gráficos das interfaces entre PE1, P1 e P2.

Antes de iniciar o segundo teste foi parada a geração de tráfego do primeiro teste com objetivo de estabelecer uma nova sessão. No primeiro teste ainda é verificado o tráfego apenas de saída na interface do roteador P2 isso é explicado, pois o *TfGen* gera tráfego apenas UDP.

Foi feito um segundo teste de geração de tráfego com o *TfGen* instalado na estação (ip 10.80.20.30) que está conectada no CE22 e gerando um tráfego para a *loopback* (200.1.0.2) do CE12, conforme a figura 65. Nesse mesmo instante é gerando um tráfego via *TfGen* da estação (10.81.20.60) conectada no CE11 para a estação conectada no CE22 (10.80.20.30). A partir da estação conectada no CE22 (ip 10.80.20.30) foi feito um *trace* para estação conectada no CE11 (ip 10.81.20.60) e obtem-se a resposta da figura 66, que segue o seguinte caminho CE22 (interface *gigabitethernet*) ➔ PE2 (interface S4/0) ➔ P1 (interface S1/1) ➔ PE1 (interface S2/0) ➔ CE11 (interface S2/0).

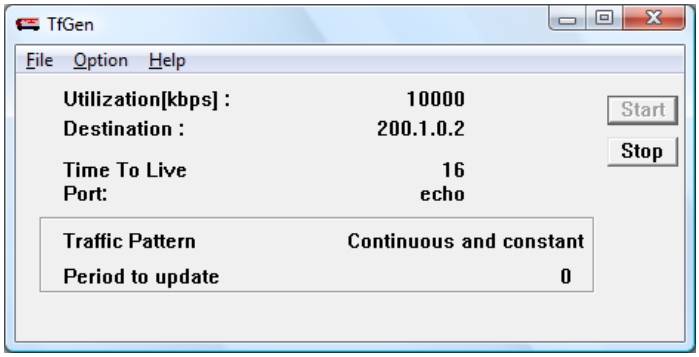


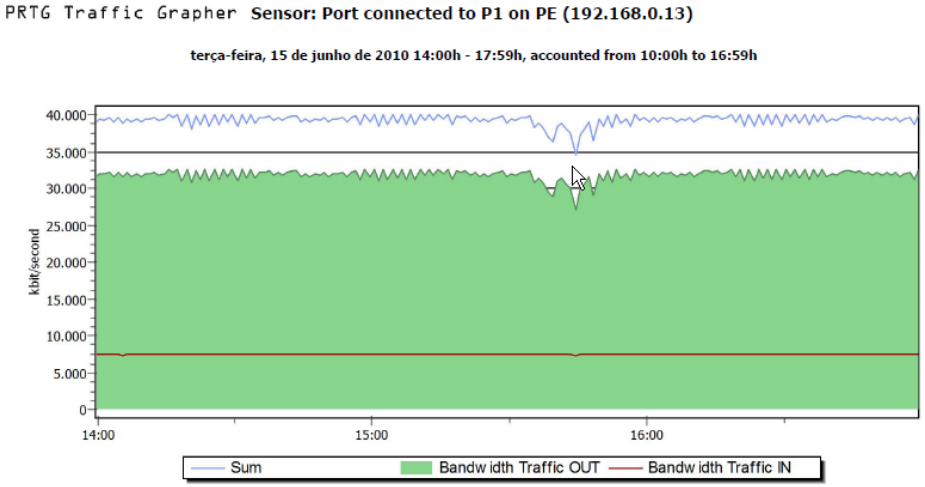
Figura 65: Tela do *TfGen* gerando um tráfego de 10000 kbps.

```
C:\Users\José Mario>tracert 10.81.20.60
Tracing route to BSE81N00716 [10.81.20.60]
over a maximum of 30 hops:

  1    1 ms   <1 ms   <1 ms  10.80.20.1
  2    1 ms    1 ms    1 ms  150.1.2.5
  3    2 ms    2 ms    2 ms  192.168.0.22
  4    2 ms    2 ms    4 ms  192.168.0.13
  5   11 ms    5 ms    8 ms  150.1.1.2
  6   15 ms   15 ms   15 ms  BSE81N00716 [10.81.20.60]
```

Figura 66: *Trace* na estação conectada no roteador CE22.

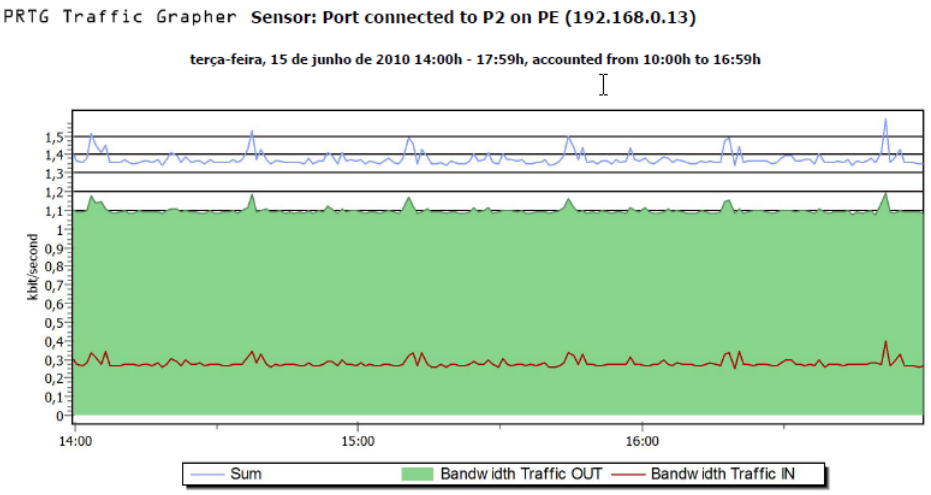
Com o *software* de gerência instalado na estação (10.80.20.30) do CE22 foram obtidos os resultados gerados nos gráficos nas figuras 67 e 68, onde é verificado que toda sessão de tráfego seguiu pela interface do roteador P1 ➔ PE1 deixando o enlace de transmissão do roteador P2 ➔ PE1 subutilizado.



Summary

	Bandwidth Traffic IN		Bandwidth Traffic OUT		Sum		Coverage
	kbyte	kbit/second	kbyte	kbit/second	kbyte	kbit/second	%
Total	9.740.512,198		41.620.366,804		51.360.879,002		
Average	54.526,193	7.444,725	232.971,334	31.812,192	143.748,764	39.256,916	75
95th Perc. (5 min)		7.454,094		32.160,327		39.609,887	

Figura 67: PRTG *Traffic Grapher* monitorando o tráfego que foi gerado na interface entre o roteador P1 e PE1.



Summary

	Bandwidth Traffic IN		Bandwidth Traffic OUT		Sum		Coverage
	kbyte	kbit/second	kbyte	kbit/second	kbyte	kbit/second	%
Total	363,908		1.437,438		1.801,347		
Average	2,034	0,278	8,036	1,097	5,035	1,375	75
95th Perc. (5 min)		0,304		1,120		1,421	

Figura 68: PRTG *Traffic Grapher* monitorando o tráfego que foi gerado na interface entre o roteador P2 e PE1.

O segundo teste foi necessário, pois no primeiro foi observada a existência apenas tráfego de saída. Já no segundo foi gerado tráfego nas duas direções pelo *TfGen*, como pode ser visto nas figuras 67 e 68, mas ainda não correspondente a um tráfego de aplicação real numa rede de pacotes IP, pois essa possui características de tráfego em rajada.

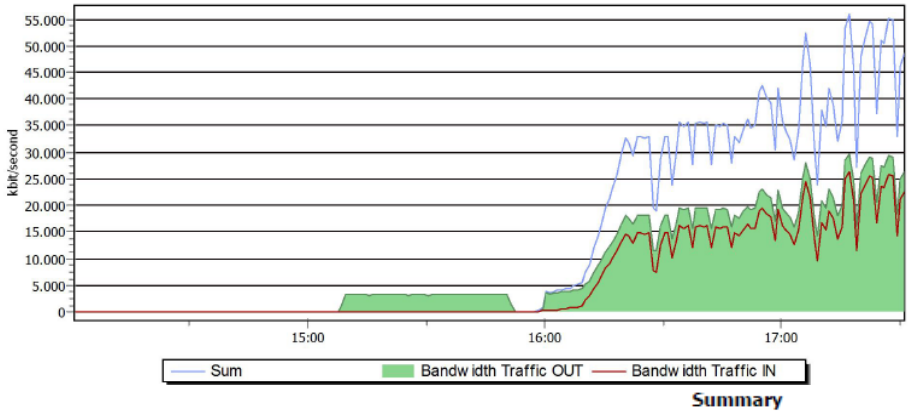
Para comprovar que o OSPF estabele uma sessão de tráfego apenas por um único caminho foi feito um teceiro teste. Nesse teste foi gerado tráfego utilizando o protocolo ICMP, gerando tráfego com aplicativo *ping*. Em uma estação conectada no CE11 foi utilizado o seguinte comando "*ping* 10.80.20.1 –t –l 17010", para que se tenha volume de tráfego foram abertas 187 janelas de testes enviando pacotes na direção ao roteador CE22 (ip 10.80.20.1 interface *Gigabitethernet* 0/1). Conectado no roteador CE22, uma estação com o *software* PRTG *Traffic Grapher* monitorando as interfaces dos roteadores PE1, foi obtido os seguintes resultados apresentados nas figuras 69, 70 e 71, onde é observado que toda a sessão de tráfego segue apenas por um único caminho decido pelo protocolo de roteamento OSPF. Na figura 69 são constatadas a super utilização da interface do roteador P1 ➔ PE1 e a subutilização da interface P2 ➔ PE1.

Figura 69: PRTG *Traffic Grapher* gerando gráficos das interfaces com ICMP.

Na figura 70 é observado o resultado da má distribuição do tráfego, pois a interface P1 ➔ PE1 está sobrecarregada enquando que a interface P2 ➔ PE2 está subutilizada conforme apresentado na figura 71, onde tráfego nesta interface é apenas de sinalização do protocolo OSPF.

PRTG Traffic Grapher **Sensor: Port connected to P1 on PE (192.168.0.13)**

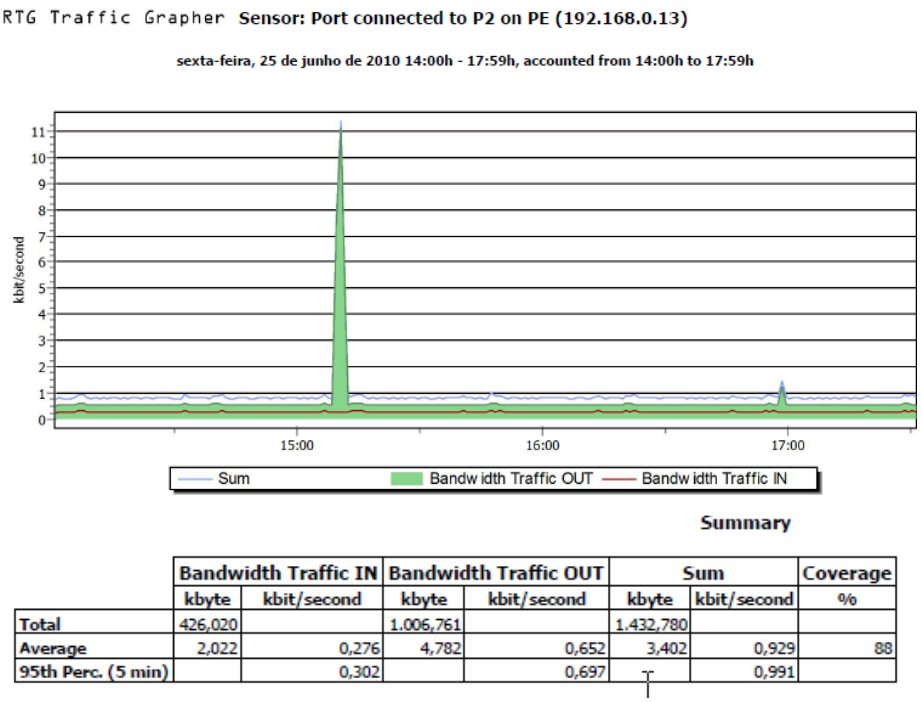
sexta-feira, 25 de junho de 2010 14:00h - 17:59h, accounted from 14:00h to 17:59h



Summary

	Bandwidth Traffic IN		Bandwidth Traffic OUT		Sum		Coverage
	kbyte	kbit/second	kbyte	kbit/second	kbyte	kbit/second	%
Total	9.367.964,885		12.742.103,643		22.110.068,527		
Average	44.208,904	6.113,477	60.207,216	8.305,797	52.208,060	14.419,274	88
95th Perc. (5 min)		19.800,656		23.695,355		43.495,381	

Figura 70: Detalhando a interface do P1 com PE1 do tráfego ICMP gerado.



Summary

	Bandwidth Traffic IN		Bandwidth Traffic OUT		Sum		Coverage
	kbyte	kbit/second	kbyte	kbit/second	kbyte	kbit/second	%
Total	426,020		1.006,761		1.432,780		
Average	2,022	0,276	4,782	0,652	3,402	0,929	88
95th Perc. (5 min)		0,302		0,697		0,991	

Figura 71: Detalhando a interface do P2 com PE1 do tráfego ICMP gerado.

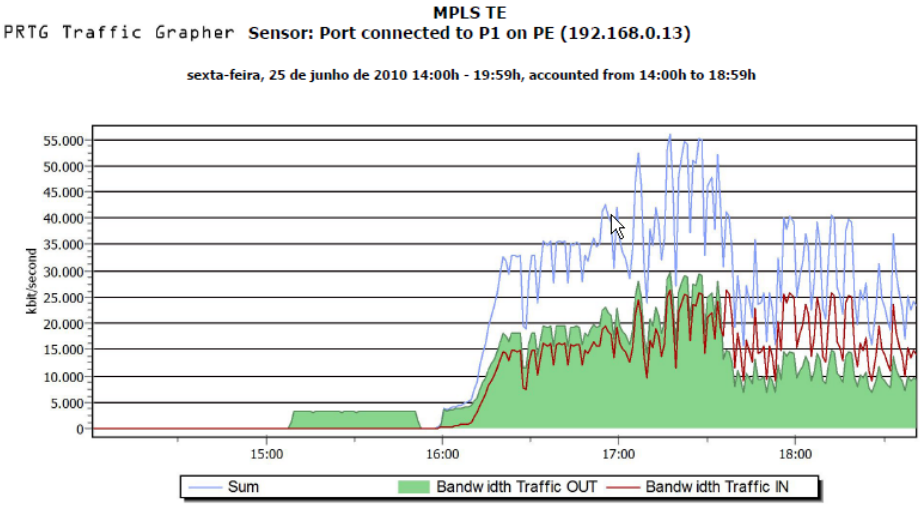
Foi observado que tráfego está seguindo apenas um único caminho. Ou seja, todo o tráfego segue o seguinte caminho estação de trabalho de origem ➔ CE11 (interface *Gigabitethernet* - entrada) - CE11 (interface Serial 2/0 - saída) ➔ PE1 (interface Serial 4/0 - entrada) – PE1 (interface Serial 2/0 - saída) ➔ P1 (interface Serial 1/0 - entrada) – P1 (interface Serial 1/1 – saída) ➔ PE2 (interface Serial 2/0 - entrada) – PE2 (interface Serial 4/0 - saída) ➔ CE22 (interface Serial 2/0 - entrada) – CE22 (interface *Gigabitethernet* - saída) ➔ estação de trabalho de destino.

Durante os testes foi utilizado apenas o protocolo de roteamento dinâmico com MPLS no *backbone* de testes.

Para uma comparação inicial da aplicação do MPLS-TE no *backbone* de testes foram criados dois túneis explícitos. O túnel T1 que segue o caminho PE1➔P2➔PE2 e túnel T2 que segue o caminho PE1➔P1➔PE2, os túneis serão representados pelas interfaces lógicas T1 e T2 respectivamente. O primeiro passo na configuração é habilitar o MPLS-TE, através do comando global "***mpls traffic-eng tunnels***", em todos os roteadores que participarão da arquitetura MPLS TE. Além disso, é necessário ativar MPLS-TE em todas as interfaces que participarão da arquitetura MPLS-TE com o seguinte comando "***mpls traffic-eng tunnels***". Os dois comandos foram aplicados nos roteadores PE1, P1, P2 e PE2. Ainda deve ser habilitado o protocolo RSVP através do comando de reserva de banda "***ip rsvp bandwidth***" que é inserido em cada interface dos

roteadores que participam do MPLS TE. No roteador PE1 que é *headend* do MPLS-TE, foram criadas as interfaces túneis T1 e T2 com o comando "***interface Tunnel***".

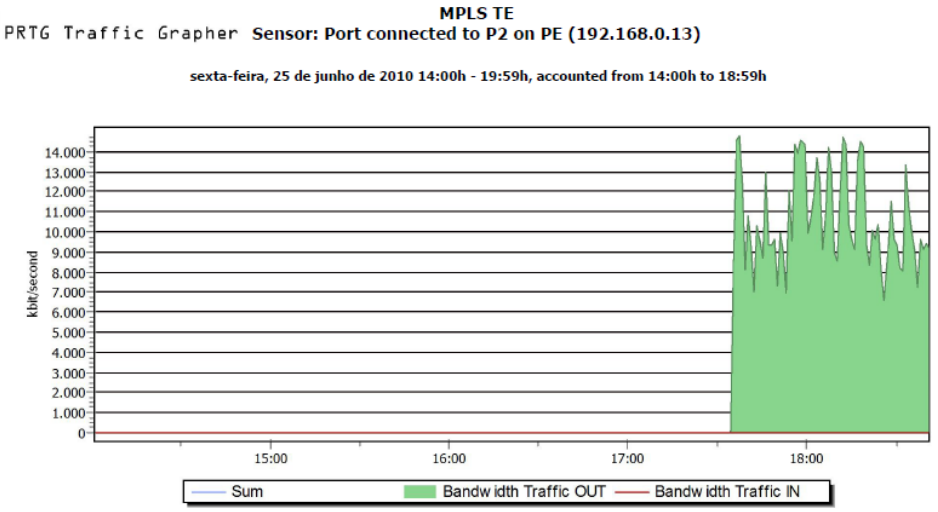
Com o tráfego sendo gerado, foram ativadas as interfaces túneis e com isso foram obtidos os seguintes gráficos das figuras 72, 73, 74 e 75. Na figura 72 é possível verificar que quando foi aplicado o MPLS-TE o tráfego na interface do roteador P1 ➔ PE1 foi reduzido de picos de 30 Mbps para picos de 15 Mbps. Já no gráfico da figura 73 é verificado que com a aplicação do MPLS-TE passa a existir volume de tráfego, pois antes essa interface estava subutilizada. A figura 74 apresenta um compativo da divisão do tráfego pelas interfaces P1 ➔ PE1 e P2 ➔ PE1 e dessa forma, tem-se um balanceamento do tráfego pelos dois caminhos o que pode constatar a otimização dos recusos de transmissão. Na figura 74 é apresentado o tráfego dividido pelos dois túneis T1 e T2 unidirecionais criados pelo MPLS-TE.



Summary

	Bandwidth Traffic IN		Bandwidth Traffic OUT		Sum		Coverage
	kbyte	kbit/second	kbyte	kbit/second	kbyte	kbit/second	%
Total	18.313.807,200		18.596.209,559		36.910.016,759		
Average	65.184,701	8.868,326	66.189,283	9.004,998	65.686,992	17.873,324	78
95th Perc. (5 min)		21.159,911		23.985,011		44.071,062	

Figura 72: Detalhe da interface P1→ PE1 com aplicação do MPLS-TE.



Summary

	Bandwidth Traffic IN		Bandwidth Traffic OUT		Sum		Coverage
	kbyte	kbit/second	kbyte	kbit/second	kbyte	kbit/second	%
Total	585,913		5.201.806,673		5.202.392,586		
Average	2,083	0,284	18.514,734	2.518,916	9.258,409	2.519,200	78
95th Perc. (5 min)		0,318		11.826,882		11.827,202	

Figura 73: Detalhe da interface P2 → PE1 com aplicação do MPLS-TE.

Figura 74: Tela do PRTG *Traffic Grapher* para comparação da aplicação do MPLS-TE.

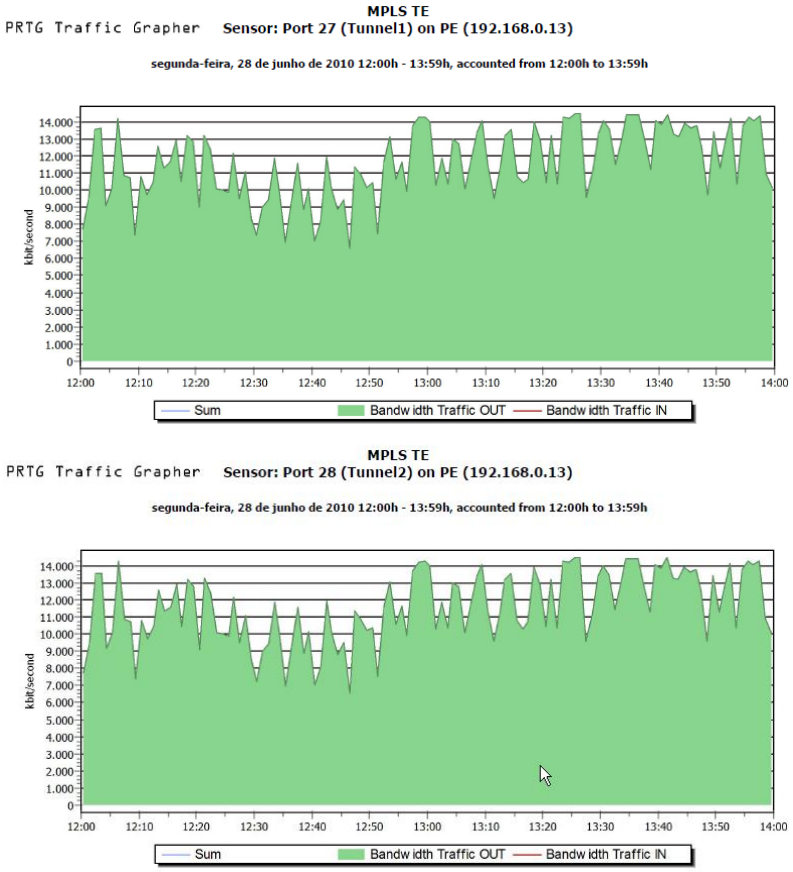


Figura 75: Gráfico para comprovação do tráfego pelo túneis T1 e T2 do MPLS-TE.

Na figura 76 é possível observar um comparativo da utilização do OSPF no *backbone* e aplicação do serviço do MPLS-TE e a partir daí é possível constatar a divisão do tráfego de forma balacenda sem deixar caminhos subtulizados.

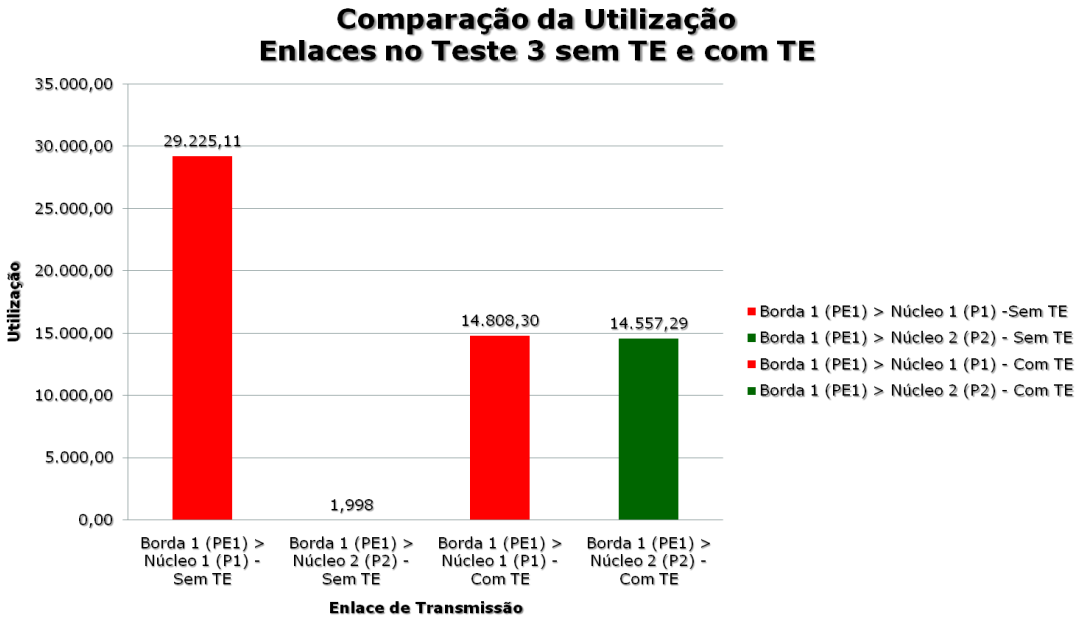


Figura 76: Gráfico para comparação da utilização do OSPF no *backbone* no cenário do teste 3 e a aplicação do MPLS-TE.

Com os gráficos apresentados nas figuras 72, 73, 74 e 75, pode-se visualizar o balanceamento do tráfego por dois caminhos distintos. E com o gráfico da figura 76 é constatado que a utilização do MPLS-TE apresenta um ganho na utilização de caminhos não utilizados pelos protocolos de roteamento dinâmico, já que para as operadoras de telecomunicações um dos maiores custo está associado a transmissão de longa distância.

Aplicando o MPLS-TE no cenário do teste 2, e gerando um gráfico de comparação conforme apresentado na figura 77, é possível constatar também o balaceamento do tráfego contínuo e constante, de forma a otimizar os recursos de trasmissão.

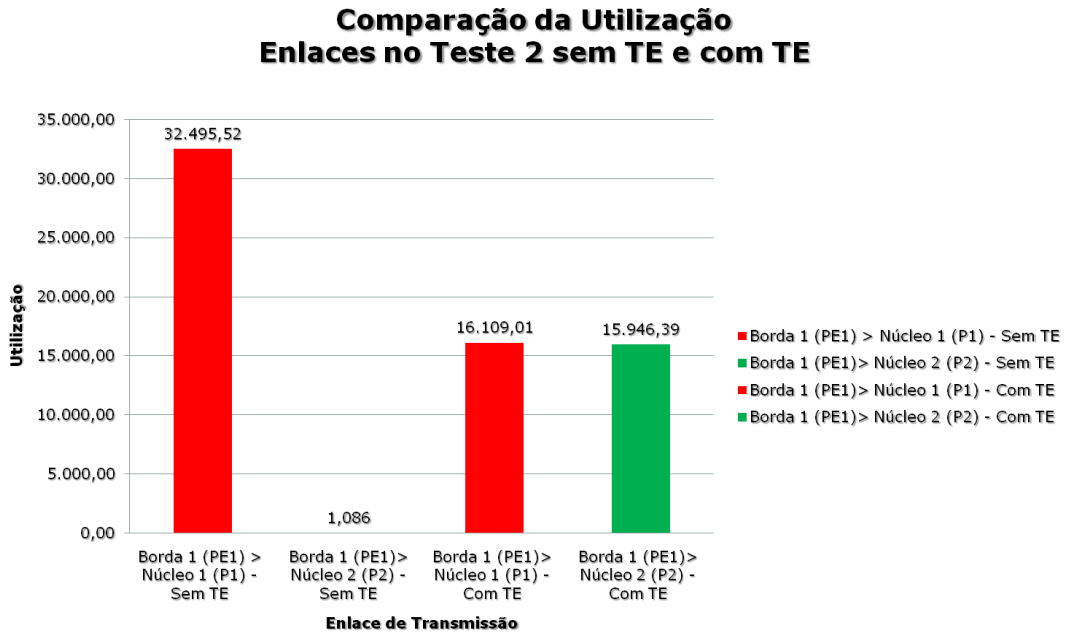


Figura 77: Gráfico para comparação da utilização do OSPF no *backbone* no cenário do teste 2 e a aplicação do MPLS-TE.

Foi feito um quarto teste apenas com MPLS e protocolo de roteamento dinâmico habilitado, mas agora gerando um tráfego contínuo e randômico a partir da estação (10.81.20.60) conectada no CE11 em direção à estação (10.80.20.60) conectada no CE22 com o gerador de tráfego *TfGen*, como apresentado na figura 78:

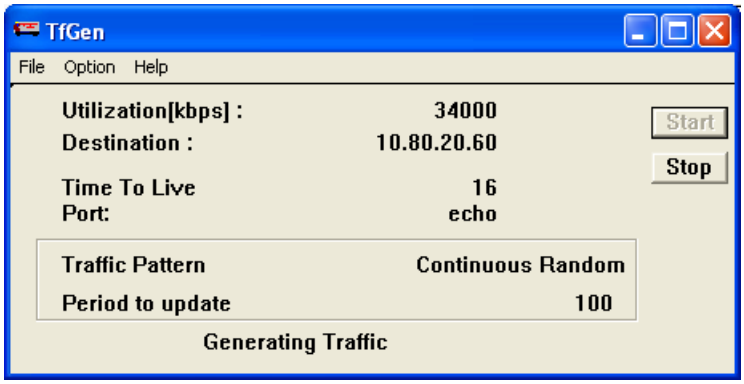


Figura 78: Tela do *TfGen* gerando um Contínuo e Randômico de 34000 kbps.

O objetivo desse teste é mostrar que é possível estabelecer sessões de tráfego do CE11➔CE22 pelo caminho PE1➔P2➔PE2 e também constatar que as sessões de tráfego contínuo e randômico também seguem um único caminho, quando é utilizado o OSPF. Ressalte-se que foi possível verificar que mesmo com um tráfego contínuo e randômico há o desperdício dos recursos de transmissão.

Foi verificada na figura 79 que foi estabelecida uma sessão de tráfego que segue apenas um único caminho (CE11➔PE1➔P2➔PE2➔CE22) determinado pelo protocolo de roteamento dinâmico OSPF como sendo o melhor caminho para essa sessão. É possível observar que todas as sessões estão sendo estabelecidas pela interface P2 ➔ PE1 e que a interface P1 ➔ PE1 está subutilizada, comprovando a má distribuição de tráfego pelo OSPF.

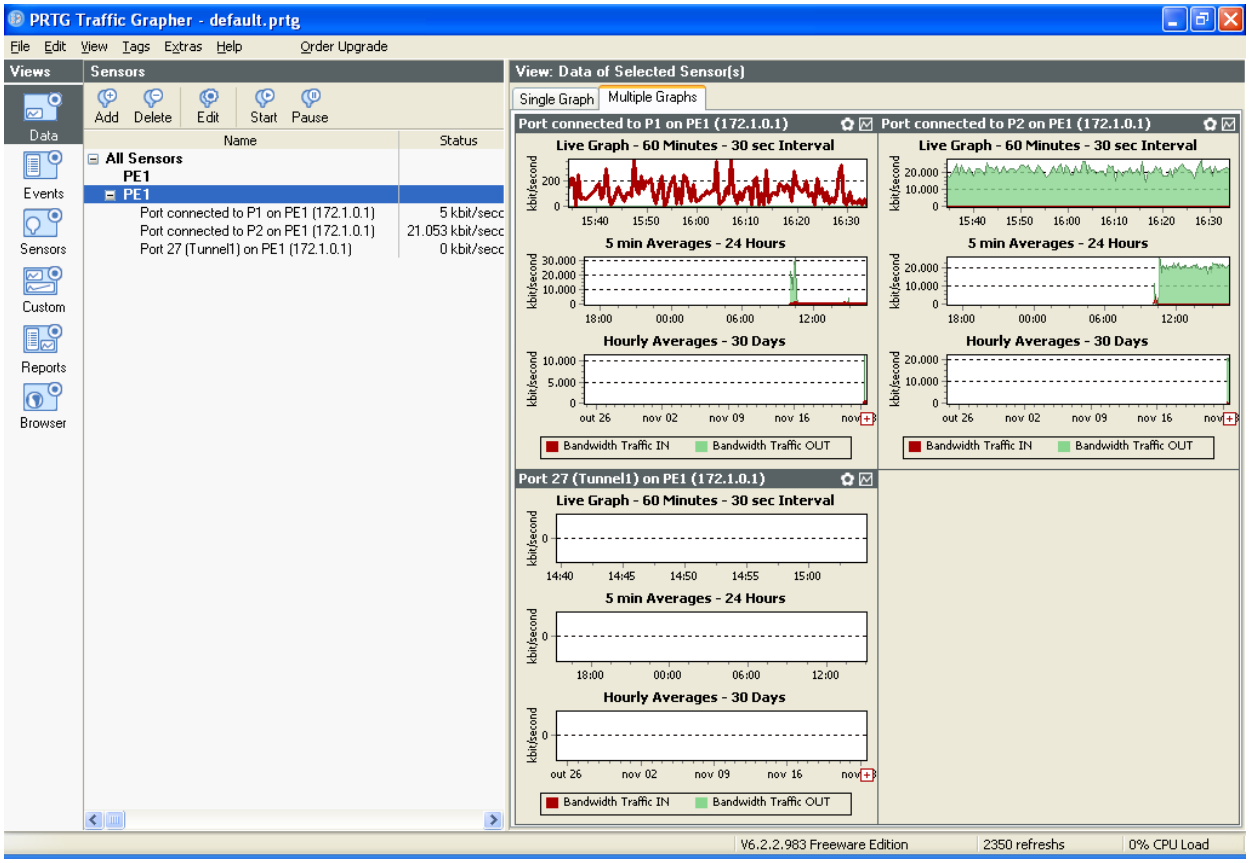


Figura 79: PRTG *Traffic Grapher* Apresentado o Tráfego Contínuo e Randômico seguindo apenas um caminho.

Com objetivo de comprovar a passagem do tráfego pela interface do P2 (serial 1/0) com PE1 (serial 2/1) foi instalado nesta interface um analisador de protocolo do modelo DA-340 do fabricante *Acterna*, que tem sua tela principal apresentada na figura 80. Nessa é possível comprovar a super utilização da interface P2 ➔ PE1 e ainda verificar a distribuição dos tipos de pacotes que estão passando por essa interface de E3.

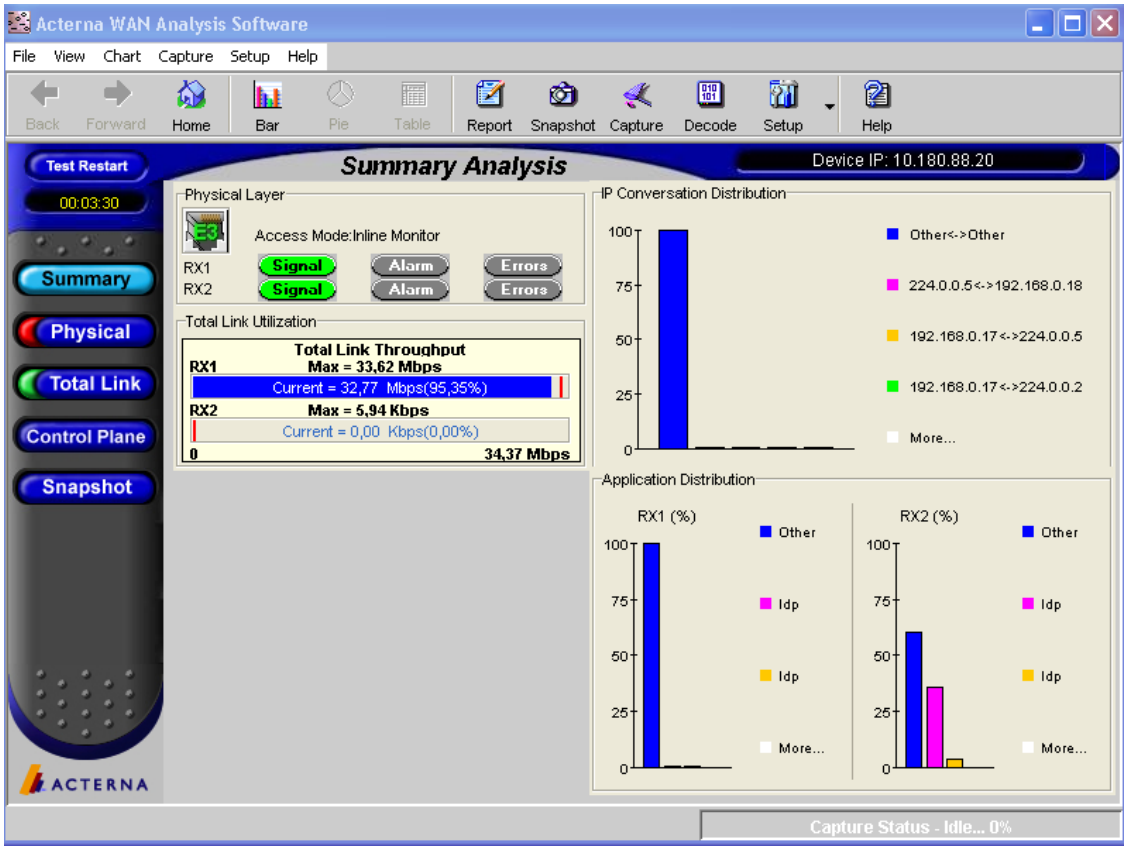


Figura 80: Analisador de protocolo *Acterna DA-340*.

O analisador de protocolo foi instalado na interface E3 conforme apresentado na figura 81, em linha, e dessa forma é possível monitorar o tráfego passando pela conexão entre os roteadores P2 e PE1.

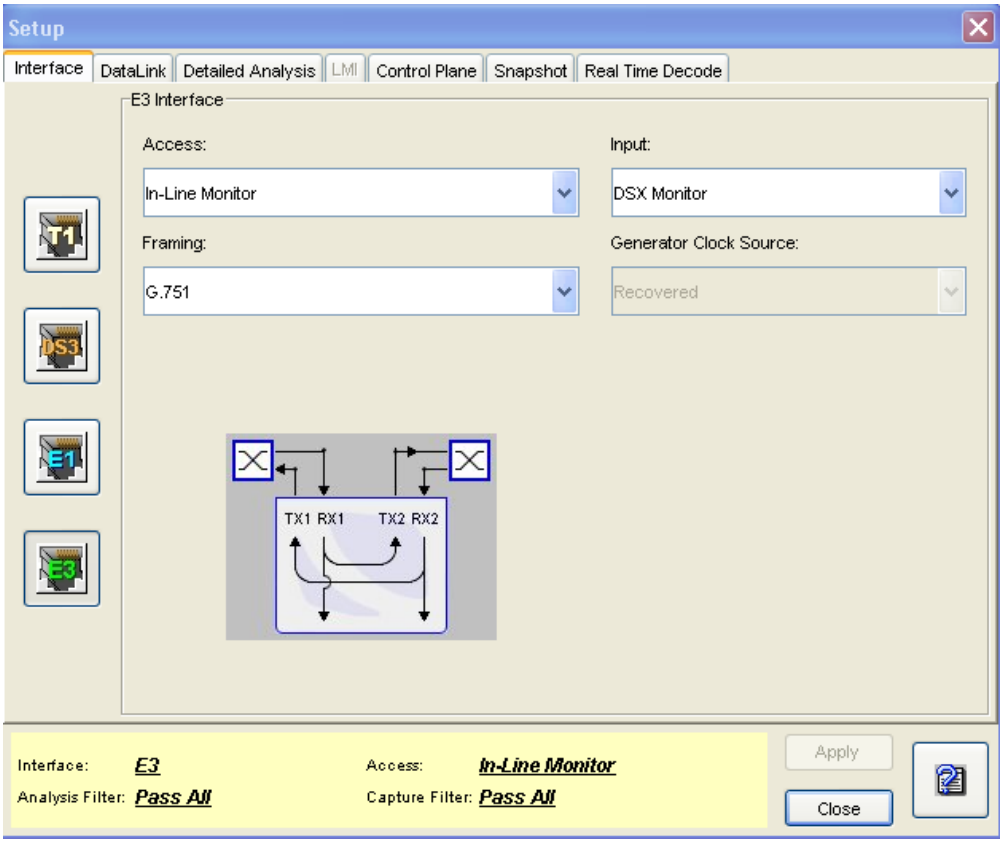


Figura 81: Analisador de protocolo *Acterna DA-340* detalhamento da instalação na WAN entre o P2 com PE11.

Com esse tráfego sendo gerado contínuo e randômico são ativadas as interfaces túneis do MPLS-TE e pode ser visto a divisão dos tráfegos pelos dois caminhos como apresentado na figura 82, nos gráficos das interfaces P1 ➔ PE1 e P2 ➔ PE.

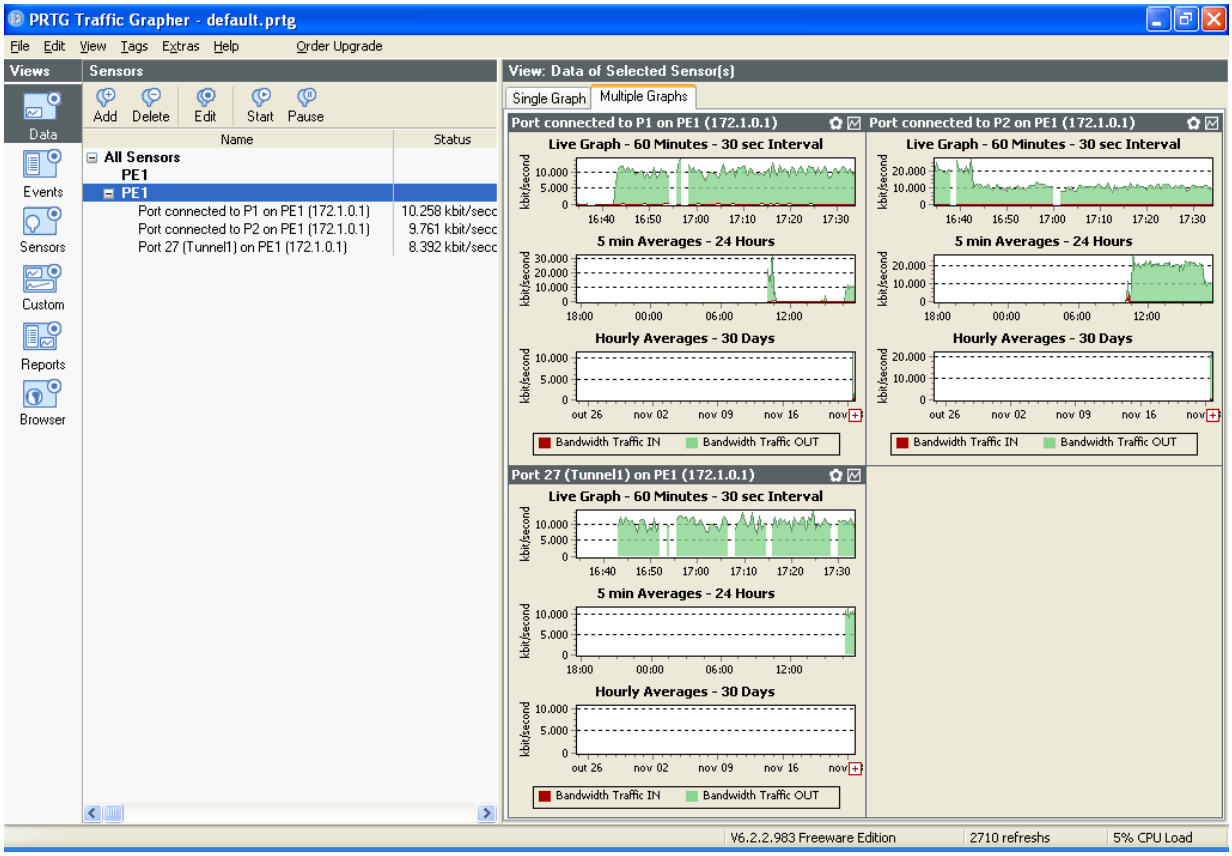


Figura 82: PRTG *Traffic Grapher* Apresentado o Tráfego Contínuo e Randômico seguindo com caminhos redundantes.

Esse mesmo comportamento pode ser observado no analisador de protocolo da *Acterna* onde pode ser visto que a utilização da interface agora é 48,76%, conforme apresentado na figura 83, e na figura 84 pode ser visto quando ocorreu a transição da aplicação do MPLS-TE. Ainda com o analisador de protocolo é gerado um relatório que comprova a aplicação do TE e desse relatório é extraída a tabela 26.

Tabela 26: Estatísticas de mensagens do plano de controle

Protocol	Monitor	Last Packet Time	Analyzed Count
BGP	Yes	11/23/2010 16:46:17.881616	109
LDP/TE	Yes	11/23/2010 16:46:36.980993	3297
OSPF	Yes	11/23/2010 16:46:29.880014	1603
RSVP	Yes	11/23/2010 16:46:28.132630	20

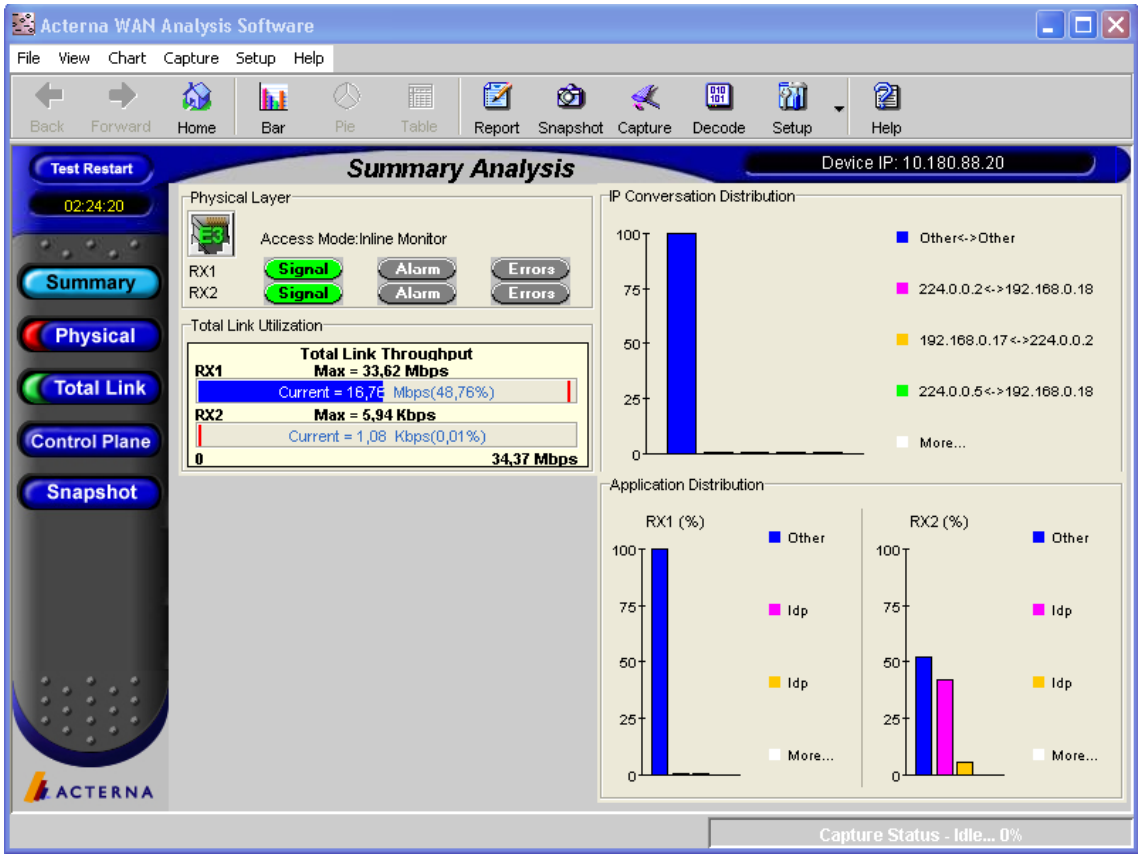


Figura 83: Analisador de protocolo Acterna com 48,76%.

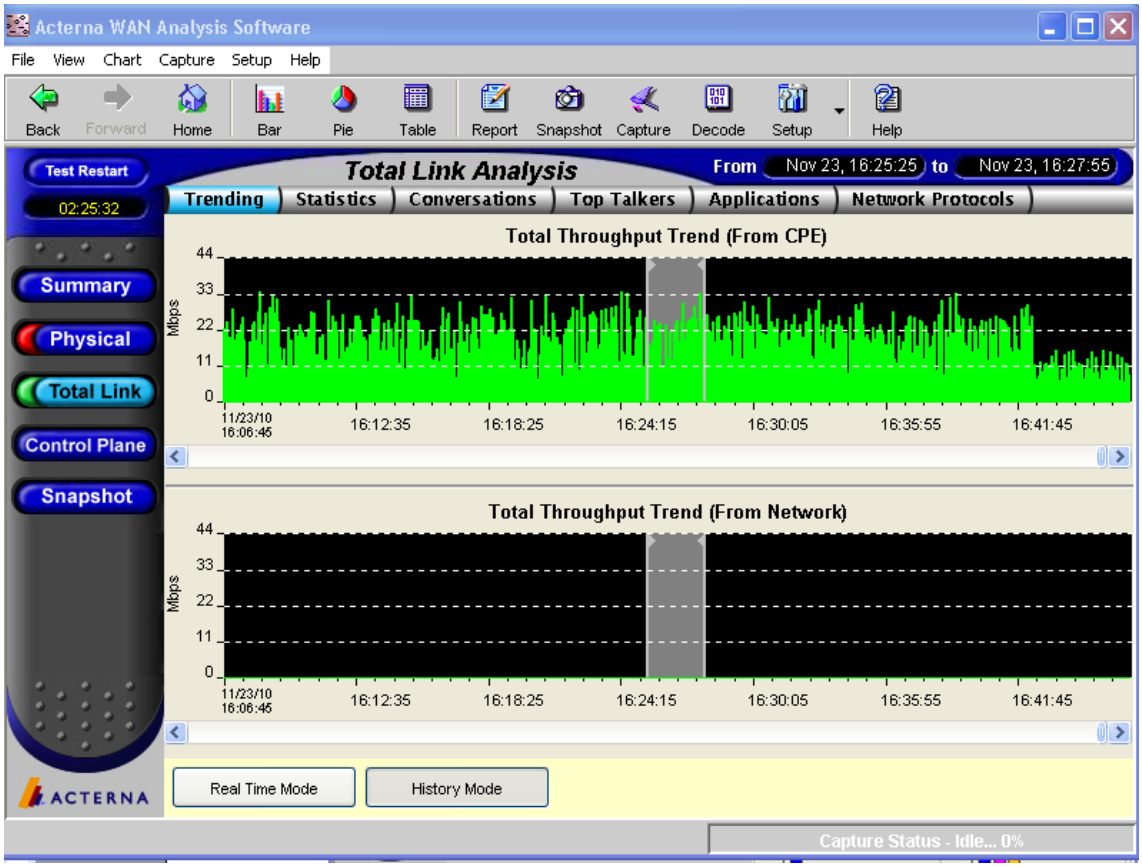


Figura 84: Analisador de protocolo Acterna transição do MPLS-TE.

Foi feito o quinto cenário de testes onde foi montado um servidor de VoIP com sistema operacional Linux e ativado o serviço do Asterisk. Esse servidor (10.81.31.60) foi conectado no roteador CE11 na interface *Gigabitethernet* 0/2 que foi configurada na rede 10.81.31.60. No roteador CE11 na interface *Gigabitethernet* 0/1 foi conectada uma estação (10.81.20.60) com cliente VoIP que utiliza o *softphone X-Lite* e foi configurado o ramal 401, e entre essa estação (10.81.20.60) e o roteador CE11 foi instalado o analisador de protocolo da *Acterna* conforme apresentado na figura 85:

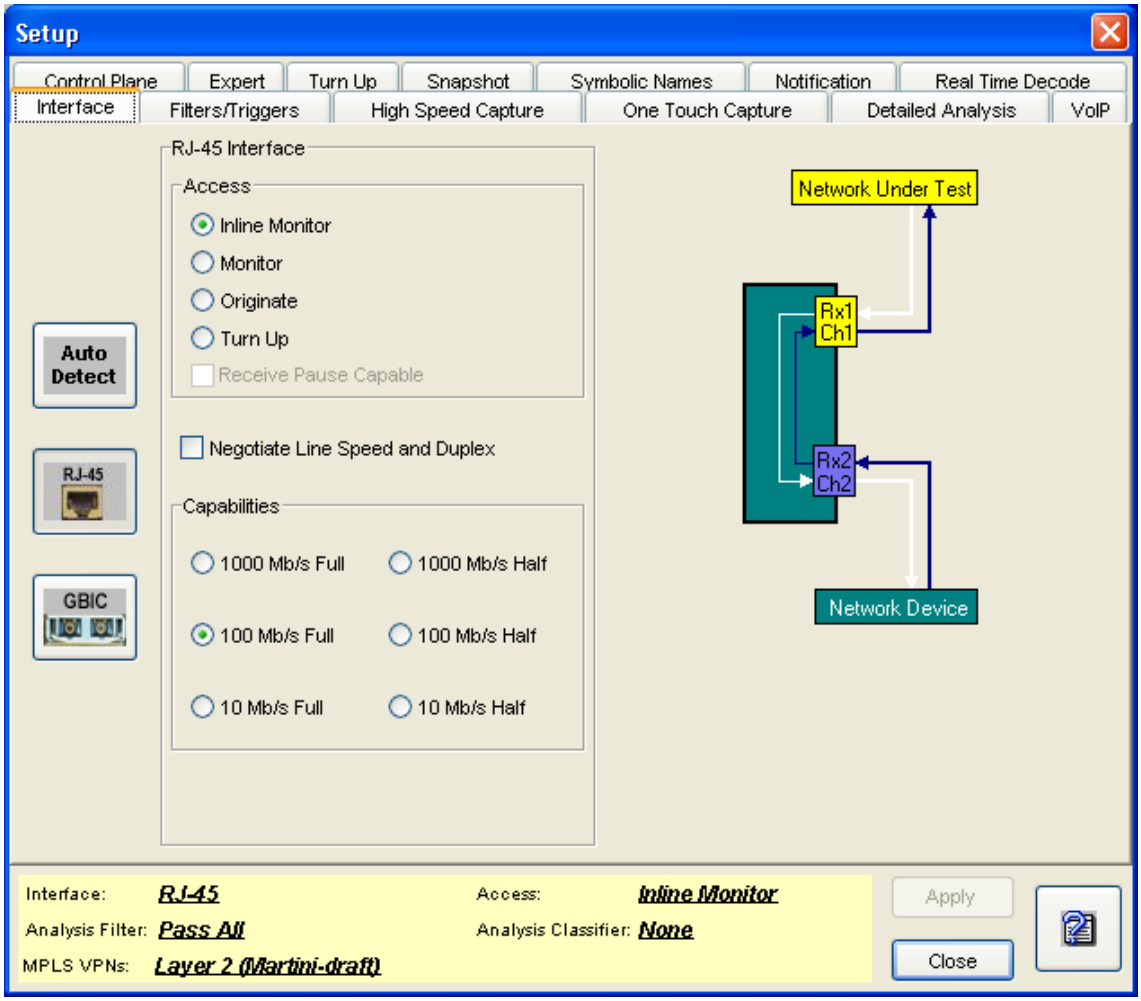


Figura 85: Analisador de protocolo Acterna, detalhamento da instalação na LAN do CE11.

Foi conectada uma estação (10.80.20.60) no CE22 na interface *Gigabitethernet* 0/1 e nessa foi instalado o cliente com VoIP X-*Lite* e configurado o ramal 501.

Na estação conecta no CE11 (10.81.20.60) foi gerado um tráfego contínuo e randômico de 34 Mbps com gerador de tráfego *TfGen* em direção a estação (10.80.20.60) conectada no CE22 esse tráfego pode ser visualizado no analisador de

protocolo conectado entre o CE11 e a estação 10.81.20.60 conforme apresentado na figura 86.

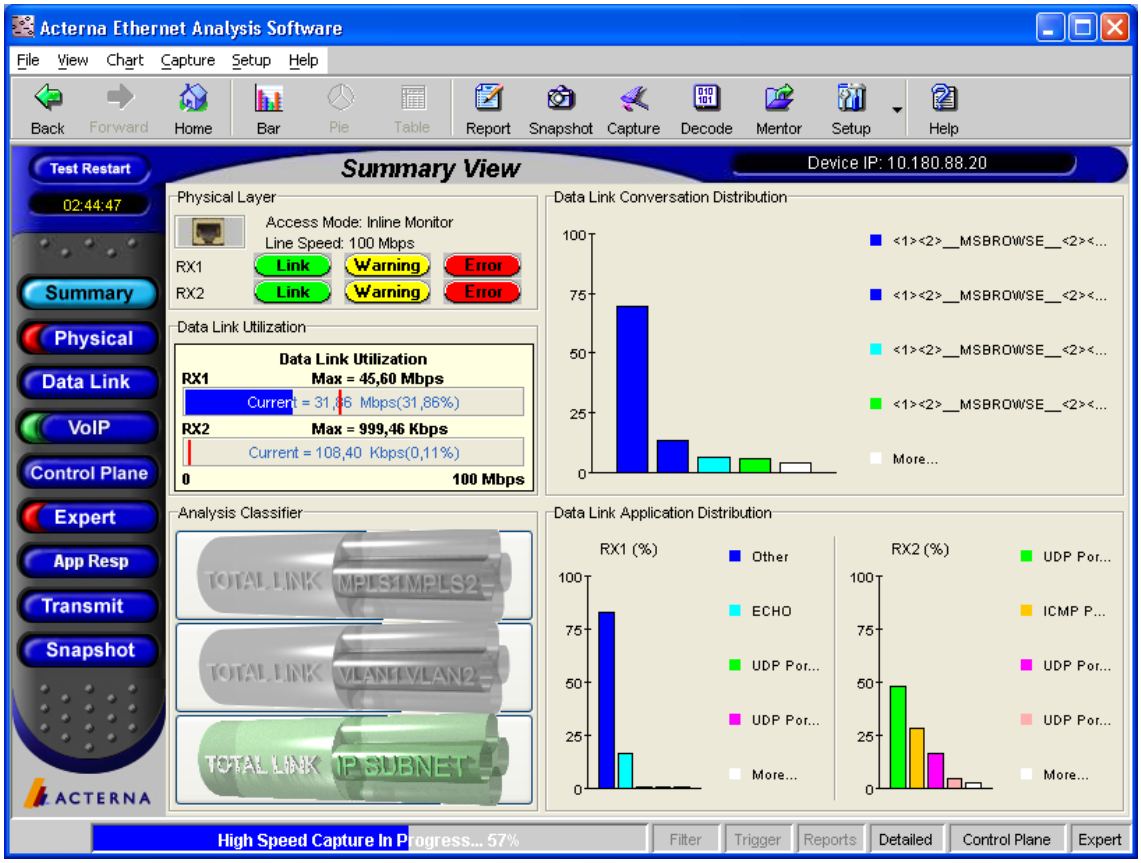


Figura 86: Analisador de protocolo *Acterna* conectado na rede 10.81.20.0 do CE11.

No mesmo instante que estava sendo gerado o tráfego foi estabelecido chamadas VoIP entre os clientes, como pode ser visto na figura 87, essa chamada utiliza o protocolo SIP (*Session Initiation Protocol*). SIP é protocolo de sinalização que controla a inicialização, modificação e terminação de sessões interativas multimídia, para troca de sinalização conforme definição da RFC 3261.

O SIP é um protocolo *peer-to-peer*, o que significa que as capacidades de rede, a exemplo de roteamento de chamadas e funções de gerenciamento de sessão, são distribuídas por todos os nós dentro da rede SIP [43].

Na figura 87 é apresentado o processo de negociação da chamada SIP entre os dois clientes da rede.

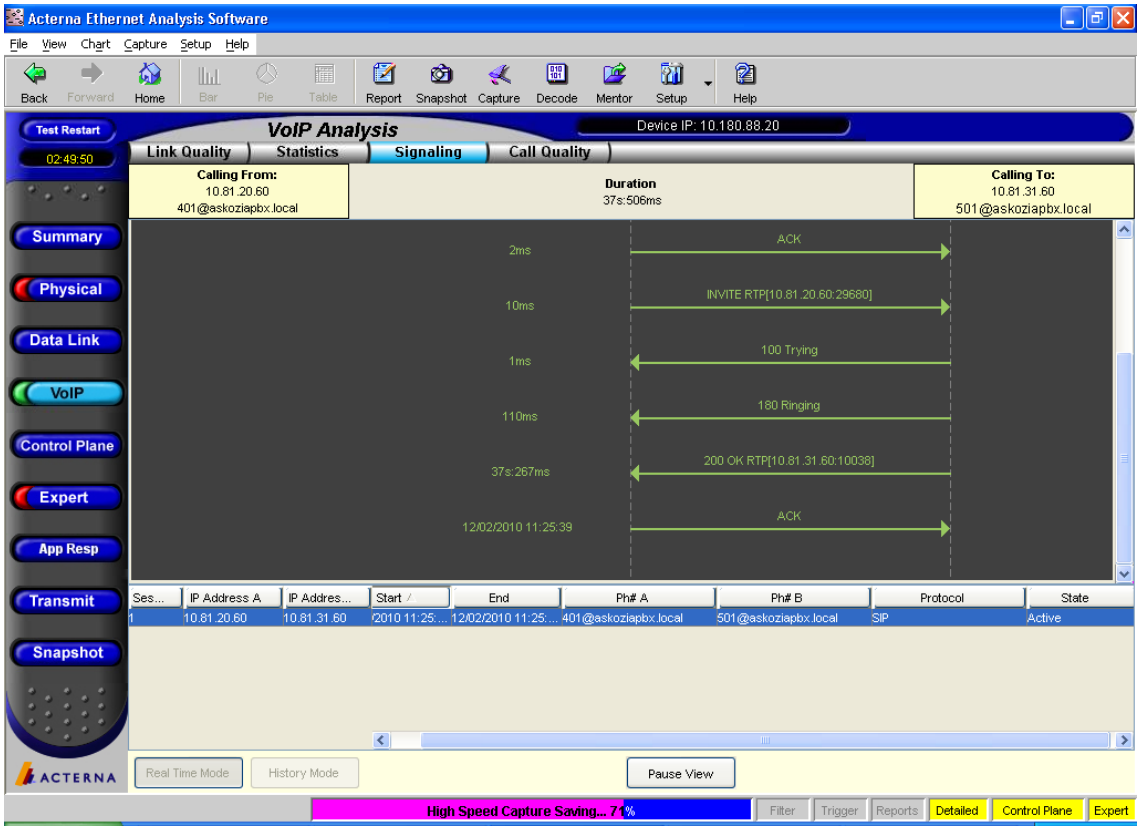


Figura 87: Detalhe da chamada VoIP com protocolo SIP.

Pode-se observar na figura 88 no *Traffic Grapher* que o tráfego está seguindo por apenas um único caminho a interface P2 ➔ PE1 já o caminho P1 ➔ PE1 está subutilizado, pois não foi aplicada a configuração de engenharia de tráfego apenas está configurado o protocolo de roteamento dinâmico OSPF.

Durante esse teste foram coletadas as telas do analisador de protocolo da *Acterna* com objetivo de apresentar os dados de serviços funcionando em tempo real e concorrendo com um tráfego contínuo e randômico. Uma dessas telas é mostrada na figura 89, onde é possível verificar a qualidade da chamada de voz em curso.

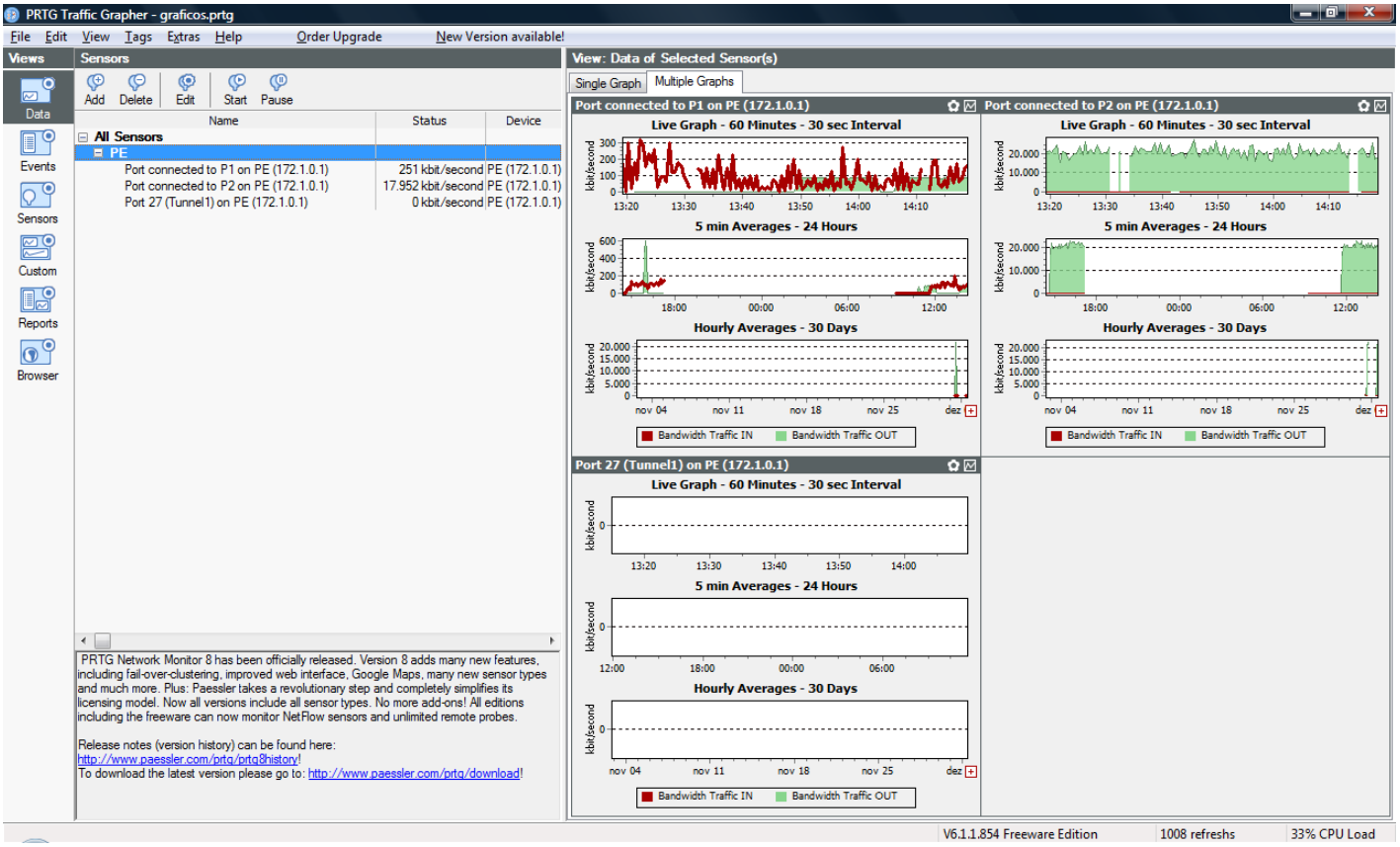


Figura 88: PRTG *Traffic Grapher* Apresentado o Tráfego Contínuo e Randômico com tráfego de pacotes de VoIP.

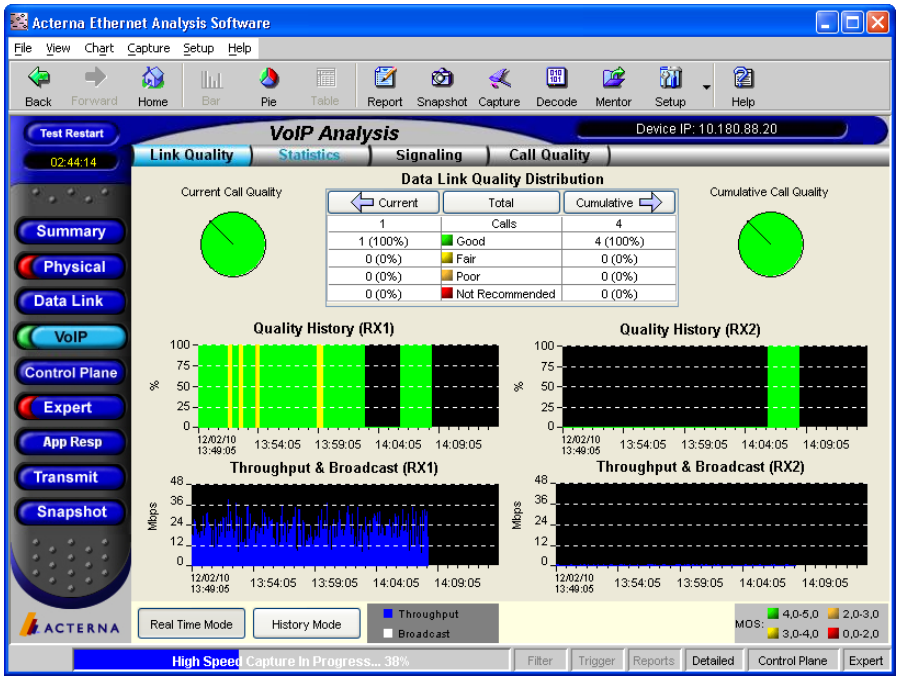


Figura 89: Analisador de protocolo *Acterna* VoIP *Analaysis*.

Outra tela bastante importante para um comparativo com aplicação do MPLS-TE é a da figura 90, pois é possível observar as estatísticas de alarmes de *jitter* nessa

rede antes da aplicação do MPLS-TE. Pode ser observado que para as chamadas geradas o analisador de protocolo detectou 4 alarmes críticos de *jitter*, baseado nos parâmetros predefinidos no instrumento de medição.

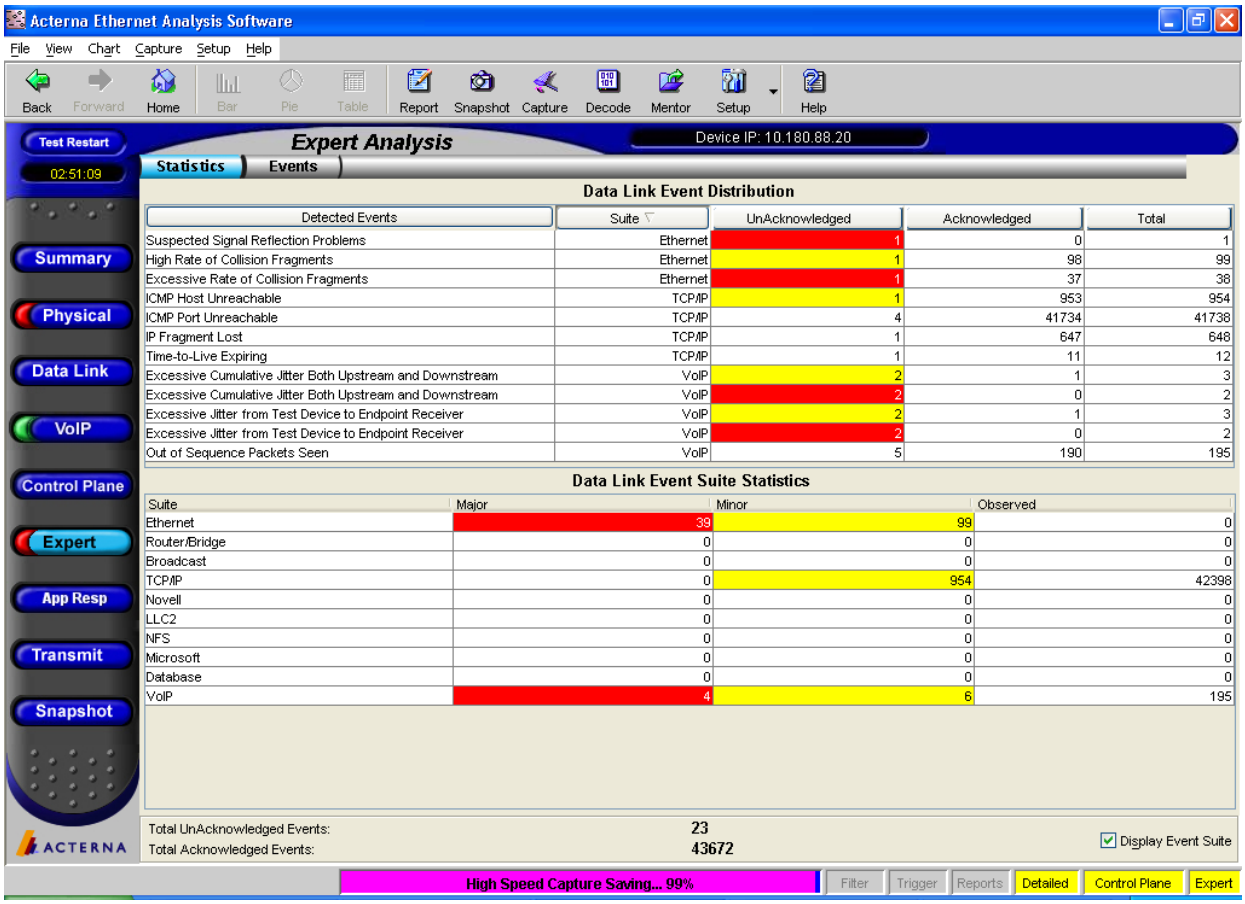


Figura 90: Analisador de protocolo *Acterna* tela *Expert Analysis*

O parâmetro da medição do *jitter* é baseado na definição do analisador de protocolo da *Acterna*, como pode ser observado na figura 91. Nessa é mostrado que os alarmes *jitter* serão gerados conforme a criticidade, como definido para variação de tempos, conforme verificado abaixo:

- de 0 – 75 ms não há registro de alarmes, eles são apenas observaveis;
- de 75 – 125 ms são gerados alarmes de criticidade menor;
- a partir de 125 ms são gerados alarmes de críticos.

Na figura 91 ainda pode ser visto como serão dectectados os alarmes de retardo (*delay*) e perda de pacotes (*packet loss*).

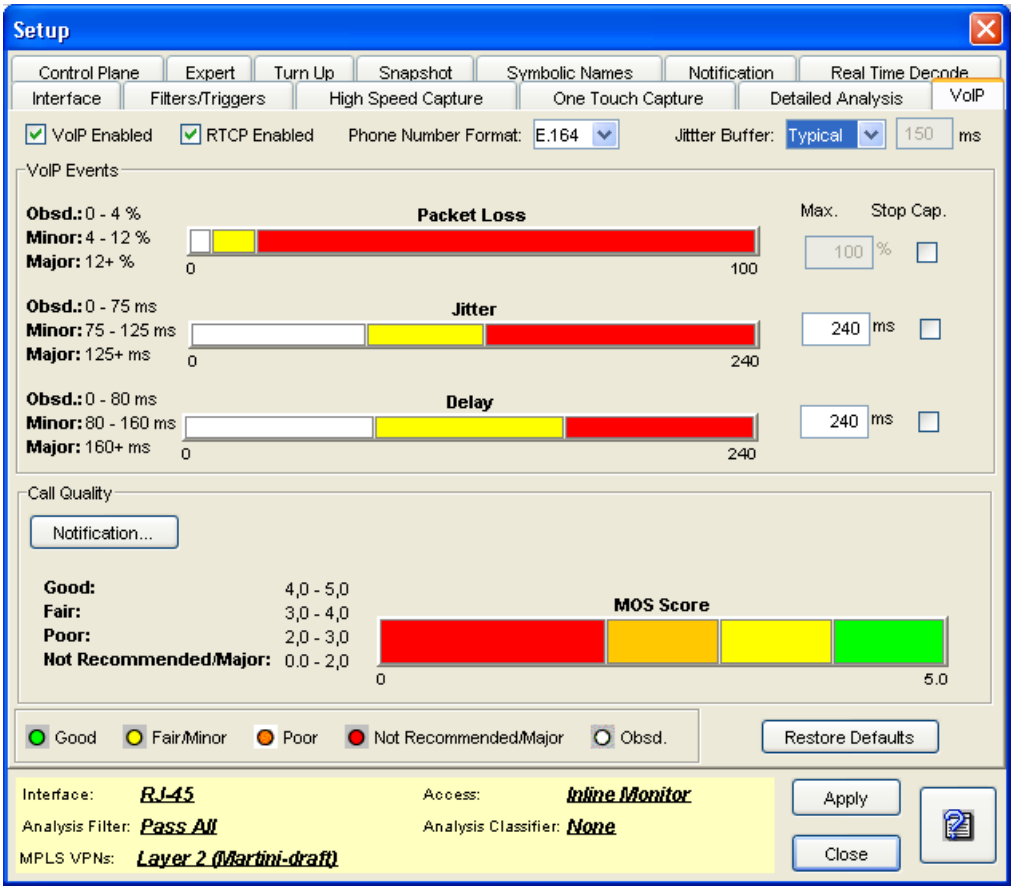


Figura 91: Analisador de protocolo Acterna parâmetros do *Jitter*

O objetivo deste teste é apresentar uma aplicação de tempo real concorrendo com outras aplicações em um *backbone* com protocolo de roteamento dinâmico e obter parâmetros de comparação com aplicação do MPLS-TE.

Quando são ativadas as interfaces túneis do MPLS-TE é observado um balanceamento do tráfego e neste instante é reiniciado a geração de chamadas de VoIP é possível observar, que com o analisador de protocolo não ocorreu incremento na estatística dos alarmes de *jitter* conforme a figura 92:

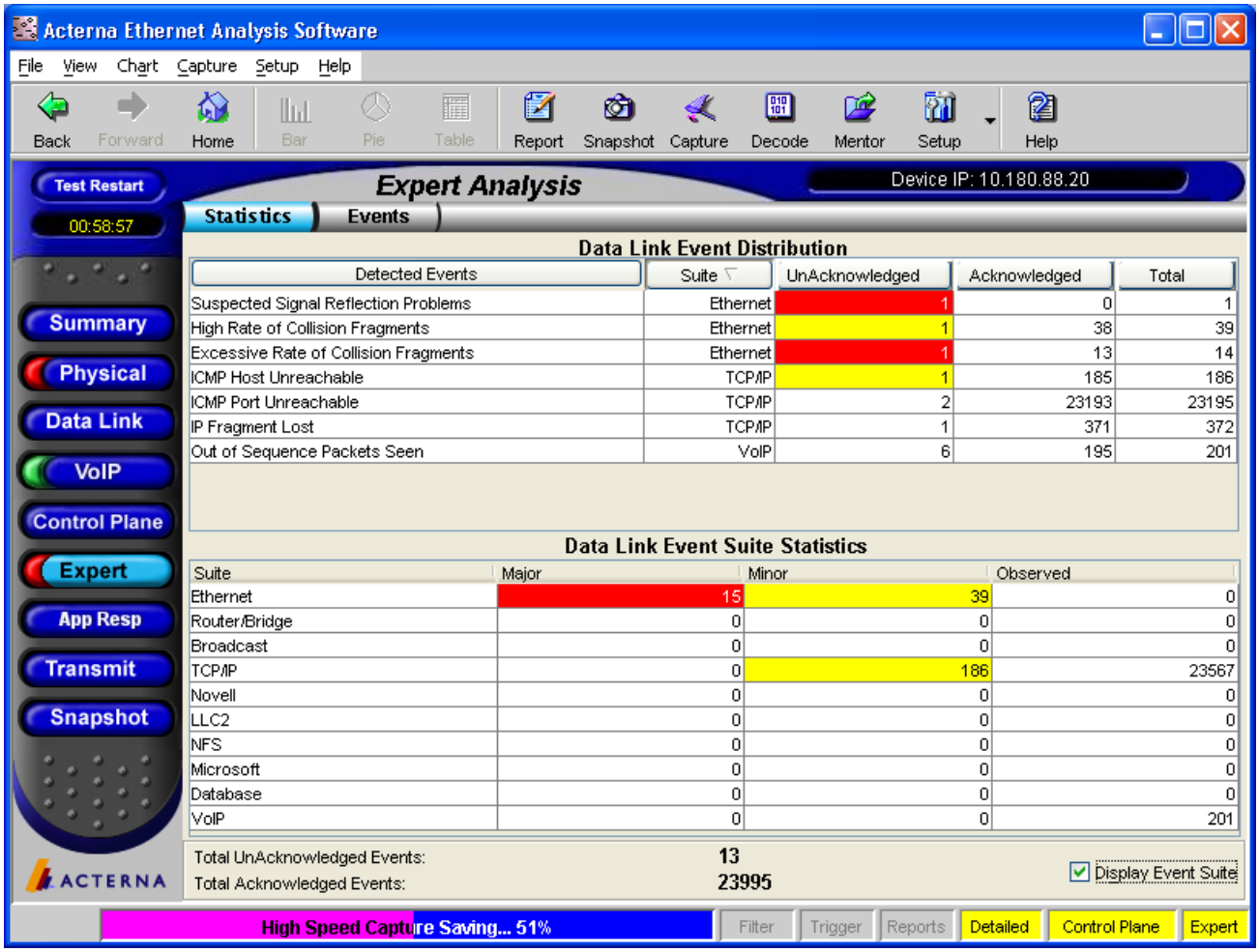


Figura 92: Tela do *Acterna DA-340* apresentando as estatísticas dos alarmes.

Visando evidenciar a não utilização de caminhos redundantes pelo protocolo de roteamento dinâmico foi realizada uma alteração na topologia do *backbone* de testes acrescentando um enlace de STM-1 (155,52 Mbps) entre o P1 (interface POS 4/0) e o PE1 (interface POS 1/0). Essa alteração pode ser vista na figura 93, que é o cenário do sexto teste que é utilizado para comparação do encaminhamento do OSPF com a engenharia de tráfego aplicada pelo MPLS-TE.

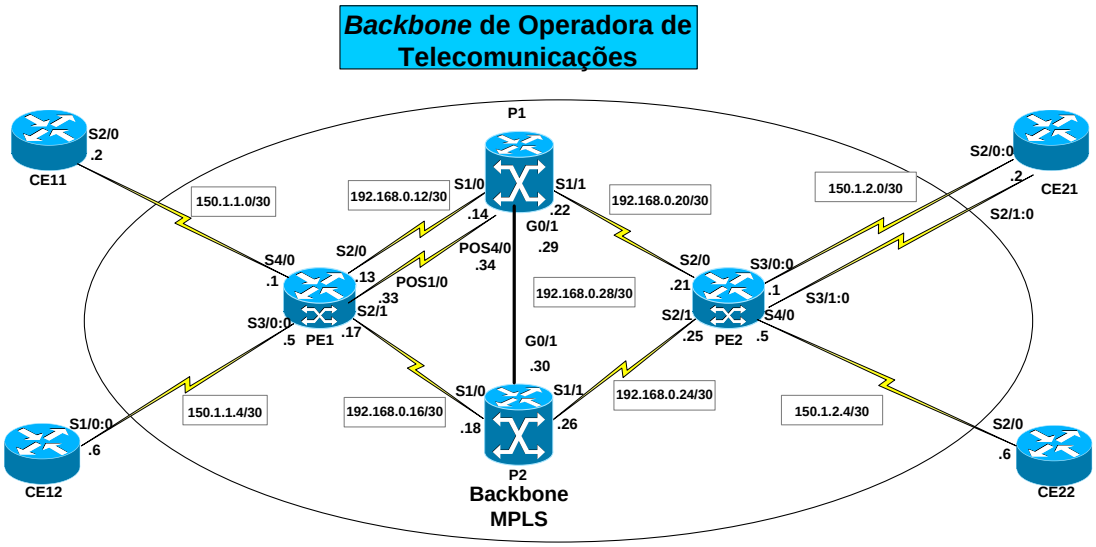


Figura 93: Fragmento de um *Backbone* de uma Operadora de Telecomunicações Comercial com enlace de 155 Mbps entre P1 e PE1.

Na figura 93 pode ser observado que a única alteração em relação a figura 55, que representa logicamente o *backbone* de testes, foi a ampliação de um novo enlace entre P1 e PE1. Feita essa alteração, é estabelecida uma chamada VoIP da estação conectada no CE11 para a estação conectada no CE22. É possível verificar na figura 94 que a chamada segue apenas por um único caminho, a interface de STM-1 entre o roteador P1 ➔ PE1, que é caminho escolhido pelo protocolo de roteamento OSPF.

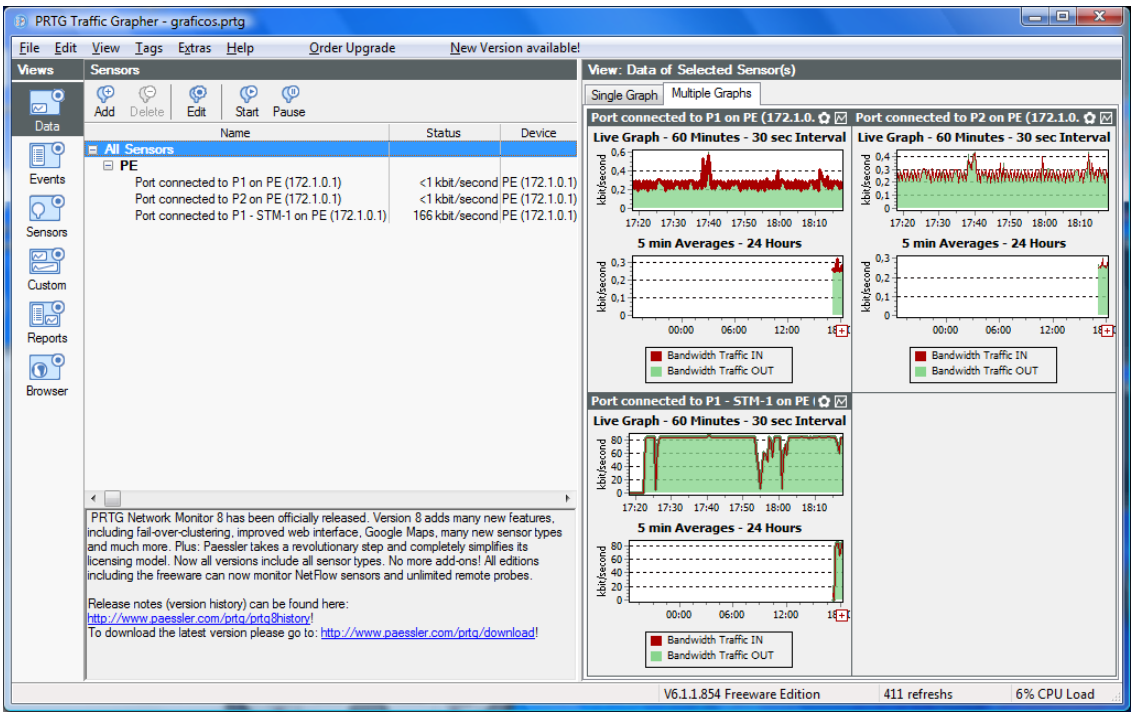


Figura 94: PRTG *Traffic Grapher* apresentado o Tráfego por um único caminho de maior banda STM-1.

Na figura 94 é visto que as outras interfaces (P1 ➔ PE1, P2 ➔ PE1) de 34 Mbps estão sem tráfego de dados, pois o único tráfego que está passando por elas é a sinalização do protocolo OSPF.

É possível observar a qualidade da chamada apresentada na figura 94, conforme a tela do analisador de protocolo da *Acterna*.

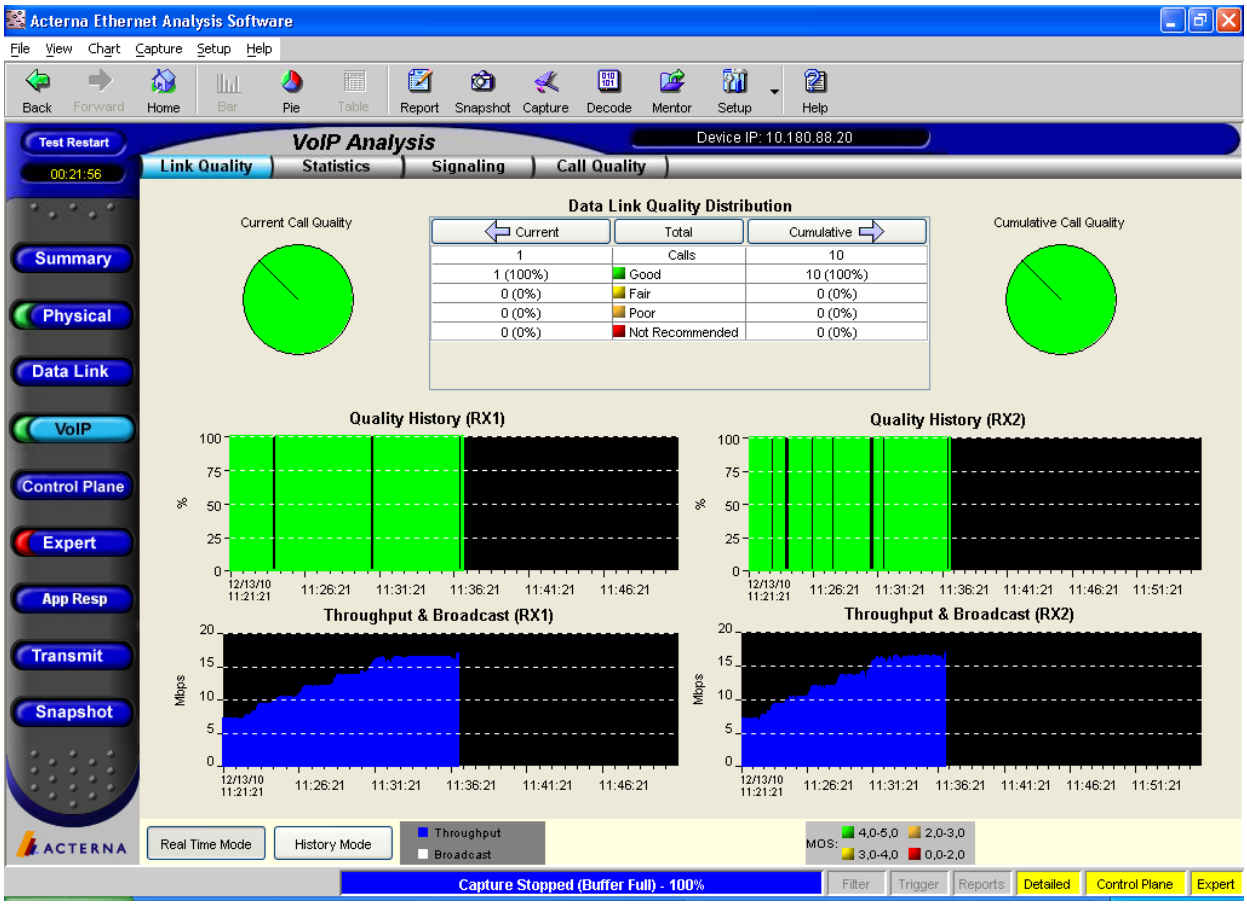


Figura 95: Analisador de protocolo Acterna detalhando
a ligação VoIP entre as estações CE11 e CE22.

Durante esse período foram geradas 10 chamadas relacionadas na figura 96 todas com MOS (*Mean Opinion Score*), que é um método comparativo subjetivo comum para quantificar o desempenho dos *codecs* de voz, igual a 4.41.

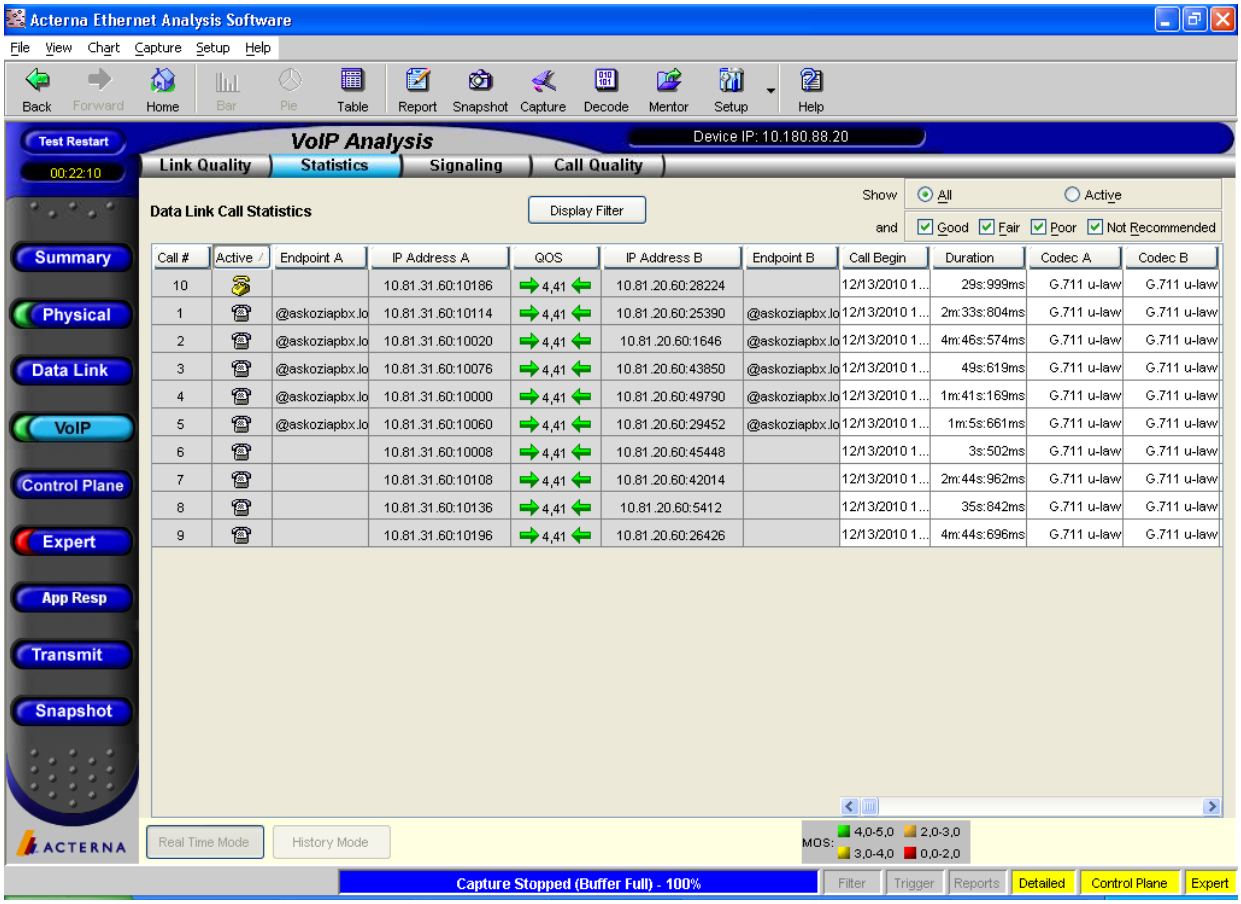


Figura 96: Analisador de protocolo *Acterna* apresentando número das chamadas passando pelo STM-1.

Iniciando uma nova medição agora gerando tráfego ICMP e via TfGen. Para ter mais volume de tráfego foi gerado, com protocolo ICMP, o comando “ping 10.80.20.60 –t –l 65500” da estação conectada no CE11 em direção ao CE22, essa ação foi repetida trinta vezes e ao mesmo tempo foi gerado um tráfego de 34 Mbps randômico e periódico para a estação conectada no CE22. Dessa forma, pode-se observar o tráfego seguindo apenas por um único caminho a interface STM-1 entre os roteadores P1 ➔ PE1, conforme apresentado na figura 97, enquanto que as outras interfaces P1 ➔ PE1 e P2 ➔ PE1 estão subutilizada. Dessa forma, é evidenciado ainda mais a problemática da má distribuição do tráfego pelo protocolo de roteamento OSPF.

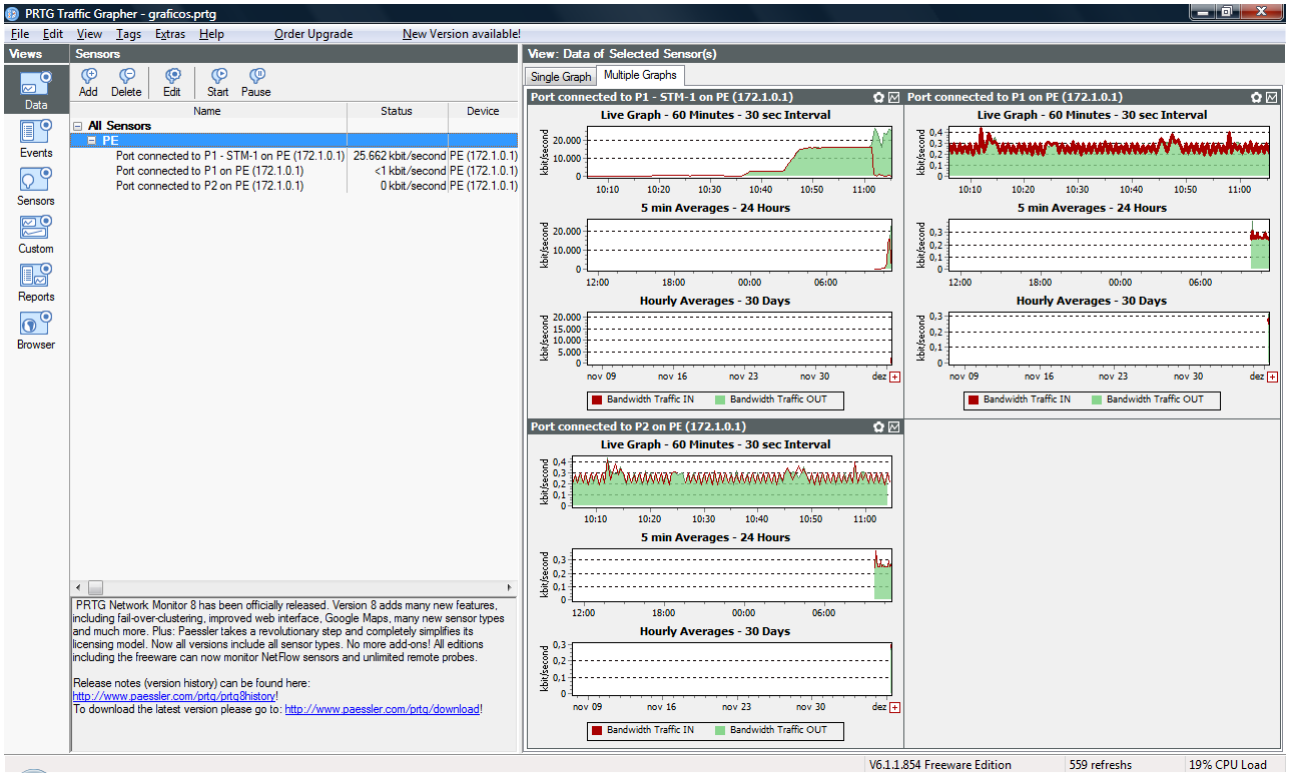
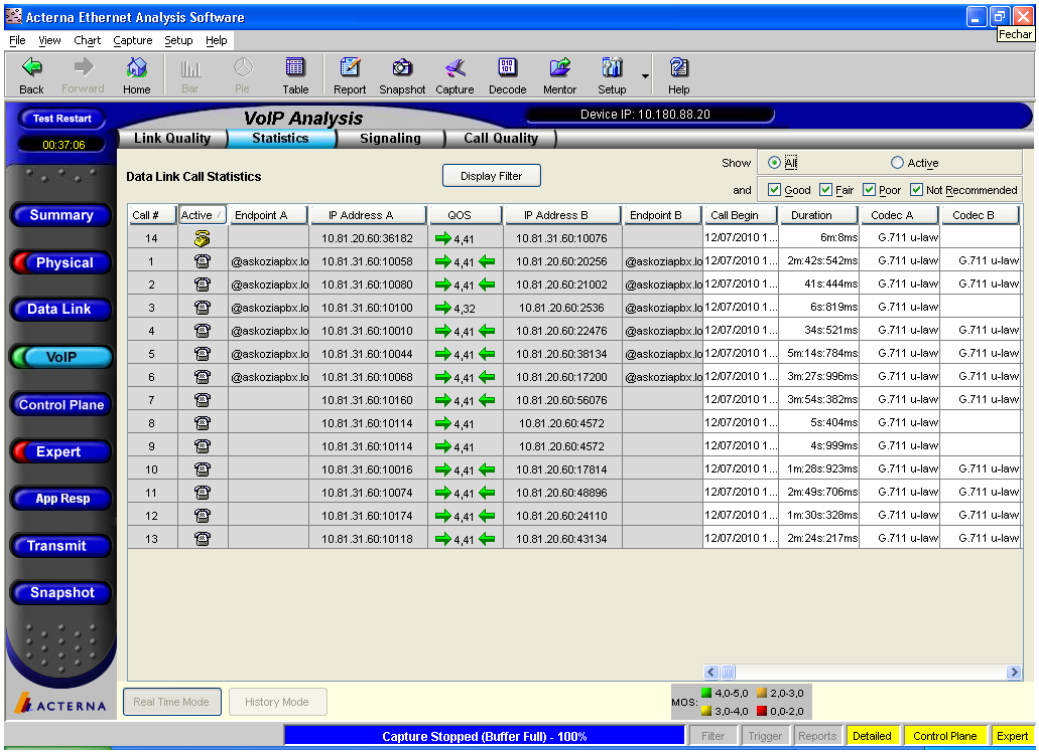


Figura 97: PRTG *Traffic Grapher* apresentado o Tráfego por um único caminho de maior banda.

Durante essa geração de tráfego foram feitas 14 chamadas via VoIP, conforme apresentado na figura 98, e com objetivo de obter parâmetros de *jitter* que podem ser observados na figura 99.



Call #	Active	Endpoint A	IP Address A	QOS	IP Address B	Endpoint B	Call Begin	Duration	Codec A	Codec B
14			10.81.20.60:36182	4,41	10.81.31.60:10076		12/07/2010 1...	6m:8ms	G.711 u-law	
1		@askoziapbx.lo	10.81.31.60:10058	4,41	10.81.20.60:20256	@askoziapbx.lo	12/07/2010 1...	2m:42s:542ms	G.711 u-law	G.711 u-law
2		@askoziapbx.lo	10.81.31.60:10080	4,41	10.81.20.60:21002	@askoziapbx.lo	12/07/2010 1...	41s:444ms	G.711 u-law	G.711 u-law
3		@askoziapbx.lo	10.81.31.60:10100	4,32	10.81.20.60:2536	@askoziapbx.lo	12/07/2010 1...	6s:819ms	G.711 u-law	
4		@askoziapbx.lo	10.81.31.60:10010	4,41	10.81.20.60:22476	@askoziapbx.lo	12/07/2010 1...	34s:521ms	G.711 u-law	G.711 u-law
5		@askoziapbx.lo	10.81.31.60:10044	4,41	10.81.20.60:38134	@askoziapbx.lo	12/07/2010 1...	5m:14s:784ms	G.711 u-law	G.711 u-law
6		@askoziapbx.lo	10.81.31.60:10068	4,41	10.81.20.60:17200	@askoziapbx.lo	12/07/2010 1...	3m:27s:996ms	G.711 u-law	G.711 u-law
7			10.81.31.60:10160	4,41	10.81.20.60:56076		12/07/2010 1...	3m:54s:382ms	G.711 u-law	G.711 u-law
8			10.81.31.60:10114	4,41	10.81.20.60:4572		12/07/2010 1...	5s:404ms	G.711 u-law	
9			10.81.31.60:10114	4,41	10.81.20.60:4572		12/07/2010 1...	4s:999ms	G.711 u-law	
10			10.81.31.60:10016	4,41	10.81.20.60:17814		12/07/2010 1...	1m:28s:923ms	G.711 u-law	G.711 u-law
11			10.81.31.60:10074	4,41	10.81.20.60:48896		12/07/2010 1...	2m:49s:706ms	G.711 u-law	G.711 u-law
12			10.81.31.60:10174	4,41	10.81.20.60:24110		12/07/2010 1...	1m:30s:328ms	G.711 u-law	G.711 u-law
13			10.81.31.60:10118	4,41	10.81.20.60:43134		12/07/2010 1...	2m:24s:217ms	G.711 u-law	G.711 u-law

Figura 98: Analisador de protocolo *Acterna* número das chamadas.

Na figura 99, é possível verificar que foram detectados 4 alarmes críticos de *jitter*, baseado no parâmetro definido no analisador de protocolo da *Acterna*.

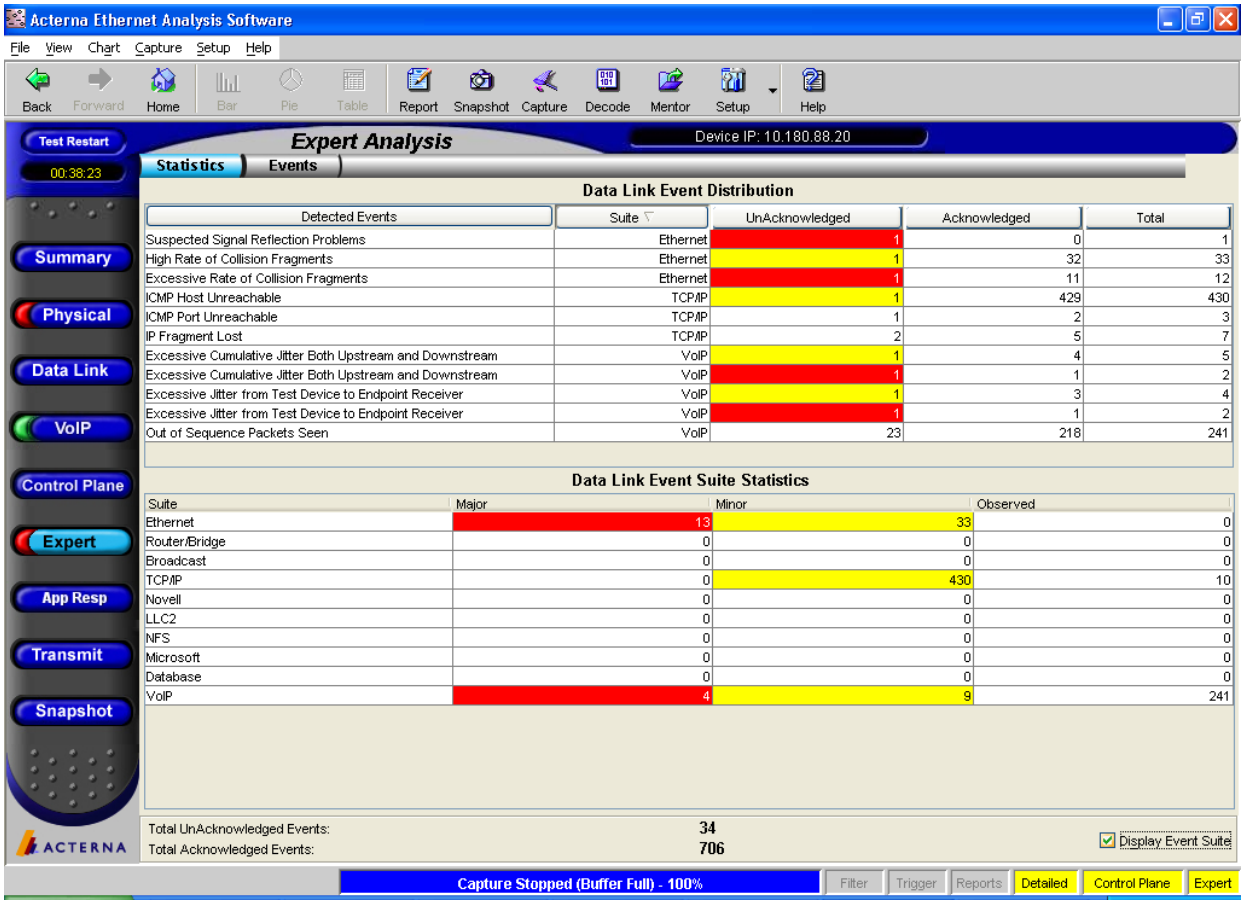


Figura 99: Analisador de protocolo *Acterna* reconhecimento de *Jitter*.

A figura 97 mostra que todo tráfego segue apenas pelo caminho de maior banda que neste caso é o enlace de 155,52 Mbps (STM-1).

Visando otimizar o uso dos enlaces foram criados três túneis seguindo os seguintes caminhos:

Túnel 1 – Interface Túnel 1 – PE1-P2-PE2;

Túnel 2 – Interface Túnel 2 – PE1-P1-PE2;

Túnel 3 – Interface Túnel 3 – PE1-STM-1-P1-PE2.

Criados esses caminhos e ativados ocorreu a divisão do tráfego pelas interfaces de PE1 com P1, conforme apresentado na figura 100.

Foram feitas 15 novas chamadas de VoIP como pode ser visto na figura 101, nas mesmas condições de tráfego.

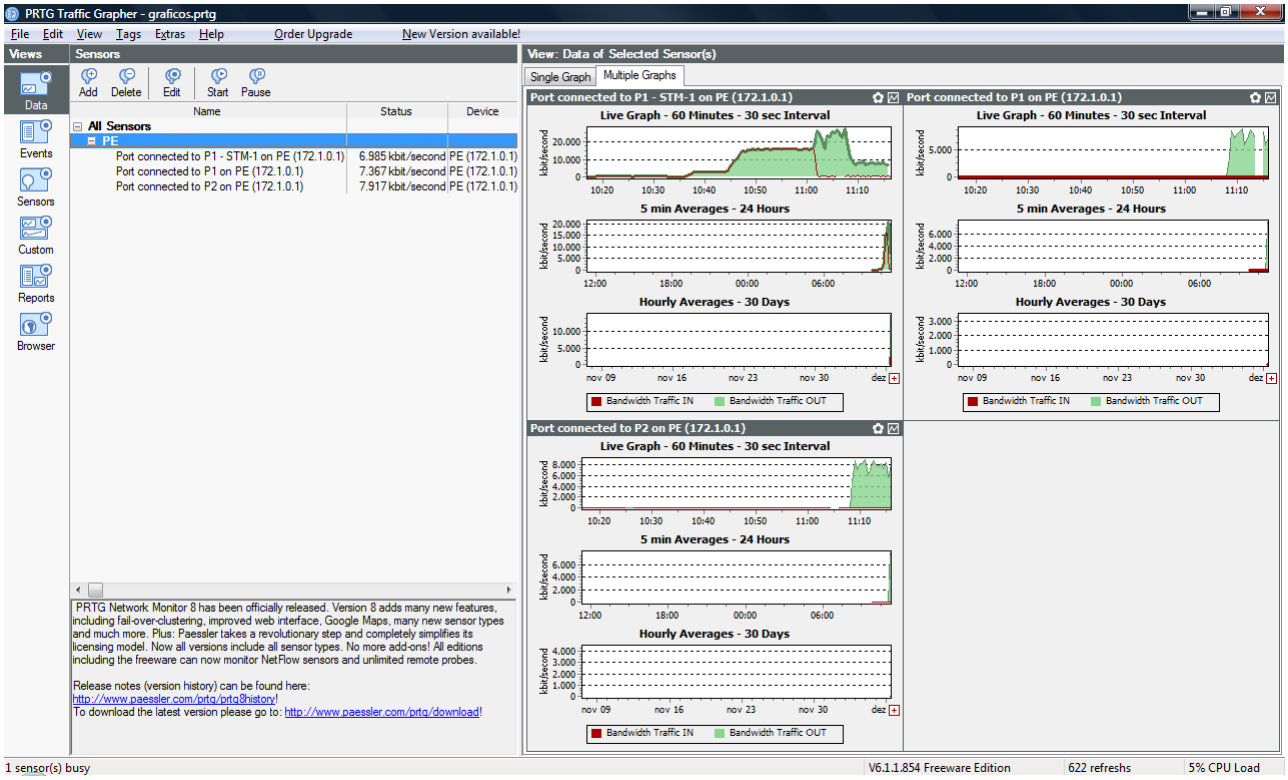
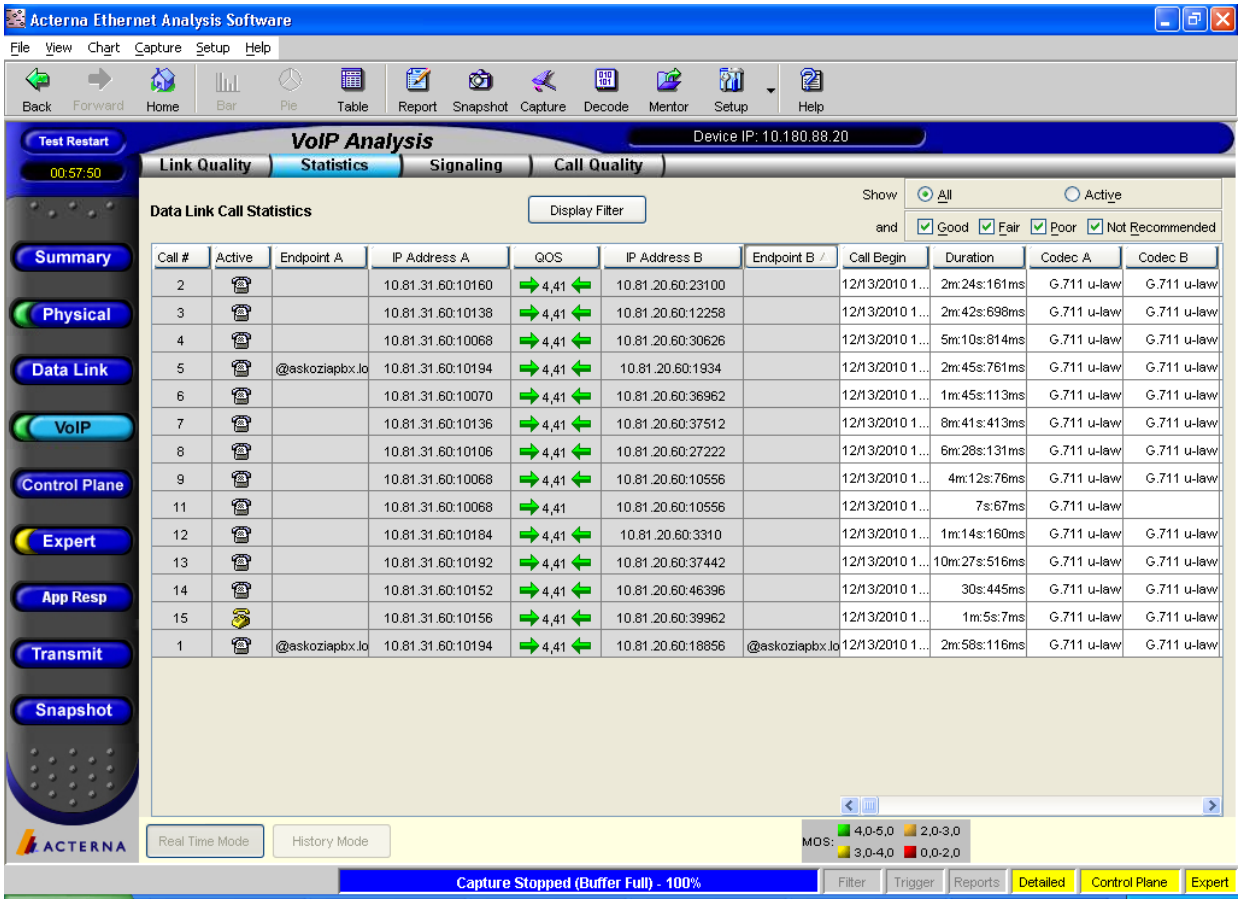


Figura 100: PRTG *Traffic Grapher* apresentado o Tráfego sendo dividido com a aplicação do MPLS-TE.



Call #	Active	Endpoint A	IP Address A	QOS	IP Address B	Endpoint B	Call Begin	Duration	Codec A	Codec B
2			10.81.31.60:10160	4,41	10.81.20.60:23100		12/13/2010 1...	2m:24s:161ms	G.711 u-law	G.711 u-law
3			10.81.31.60:10138	4,41	10.81.20.60:12258		12/13/2010 1...	2m:42s:698ms	G.711 u-law	G.711 u-law
4			10.81.31.60:10068	4,41	10.81.20.60:30626		12/13/2010 1...	5m:10s:814ms	G.711 u-law	G.711 u-law
5		@askoziapbx.lo	10.81.31.60:10194	4,41	10.81.20.60:1934		12/13/2010 1...	2m:45s:761ms	G.711 u-law	G.711 u-law
6			10.81.31.60:10070	4,41	10.81.20.60:36962		12/13/2010 1...	1m:45s:113ms	G.711 u-law	G.711 u-law
7			10.81.31.60:10136	4,41	10.81.20.60:37512		12/13/2010 1...	8m:41s:413ms	G.711 u-law	G.711 u-law
8			10.81.31.60:10106	4,41	10.81.20.60:27222		12/13/2010 1...	6m:28s:131ms	G.711 u-law	G.711 u-law
9			10.81.31.60:10068	4,41	10.81.20.60:10556		12/13/2010 1...	4m:12s:76ms	G.711 u-law	G.711 u-law
11			10.81.31.60:10068	4,41	10.81.20.60:10556		12/13/2010 1...	7s:67ms	G.711 u-law	
12			10.81.31.60:10184	4,41	10.81.20.60:3310		12/13/2010 1...	1m:14s:160ms	G.711 u-law	G.711 u-law
13			10.81.31.60:10192	4,41	10.81.20.60:37442		12/13/2010 1...	10m:27s:516ms	G.711 u-law	G.711 u-law
14			10.81.31.60:10152	4,41	10.81.20.60:46396		12/13/2010 1...	30s:445ms	G.711 u-law	G.711 u-law
15			10.81.31.60:10156	4,41	10.81.20.60:39962		12/13/2010 1...	1m:5s:7ms	G.711 u-law	G.711 u-law
1		@askoziapbx.lo	10.81.31.60:10194	4,41	10.81.20.60:18856	@askoziapbx.lo	12/13/2010 1...	2m:58s:116ms	G.711 u-law	G.711 u-law

Figura 101: Analisador de protocolo Acterna apresentado 14 novas chamadas.

Com este teste é possível observar na figura 102 que não ocorreu o incremento de alarmes de *jitter*, já que o anlisador de protocolo não detectou quaisquer alarmes críticos, o que para uma rede que tenha o tráfego de chamadas VoIP é perfeito.

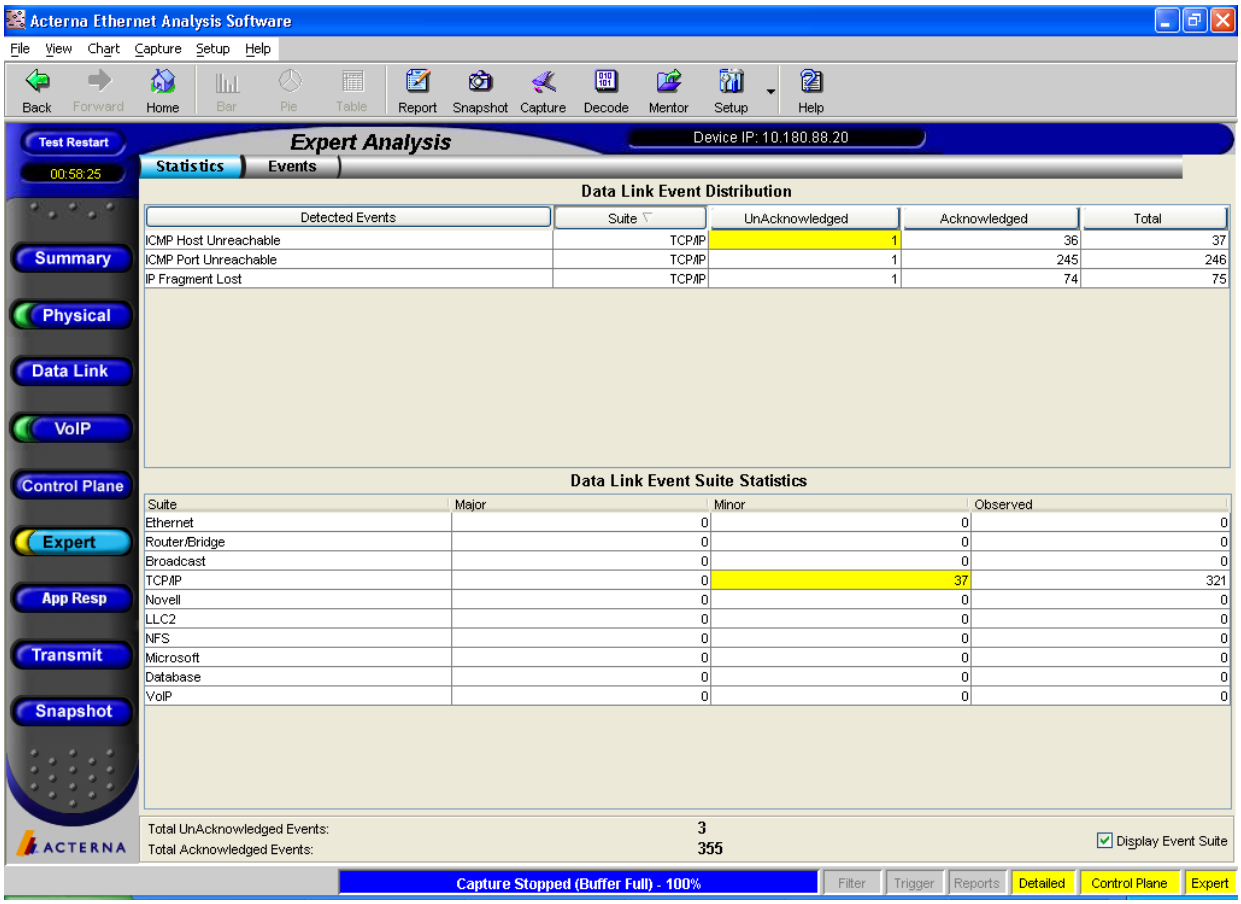


Figura 102: Tela do Analisador de protocolo Acterna do detalhe do *Expert Analysis*.

Para verificar se o comportamento do incremento de *jitter* também ocorre com chamadas VoIP sendo geradas a partir de telefones conectados em placas VIC FXS (*Voice Interface Cards Foreign eXchange Subscriber*) nos roteadores, foi conectado um roteador *Cisco* 1750 equipado com a placa apresentado na figura 103, na interface *Ethernet* do roteador CE12, dessa mesma forma foi conectado na *Ethernet* do roteador CE21. Feitas as conexões físicas e configurações necessárias, foi feito um sétimo teste a partir de telefones conectados nas placas FXS dos respectivos roteadores. Foram colocadas em *shutdown* as interfaces túneis e iniciada uma nova geração de tráfego e neste mesmo instante foram feitas 14 novas chamadas, conforme figura 104.

Figura 103: Roteador *Cisco* 1750 com placa FXS.

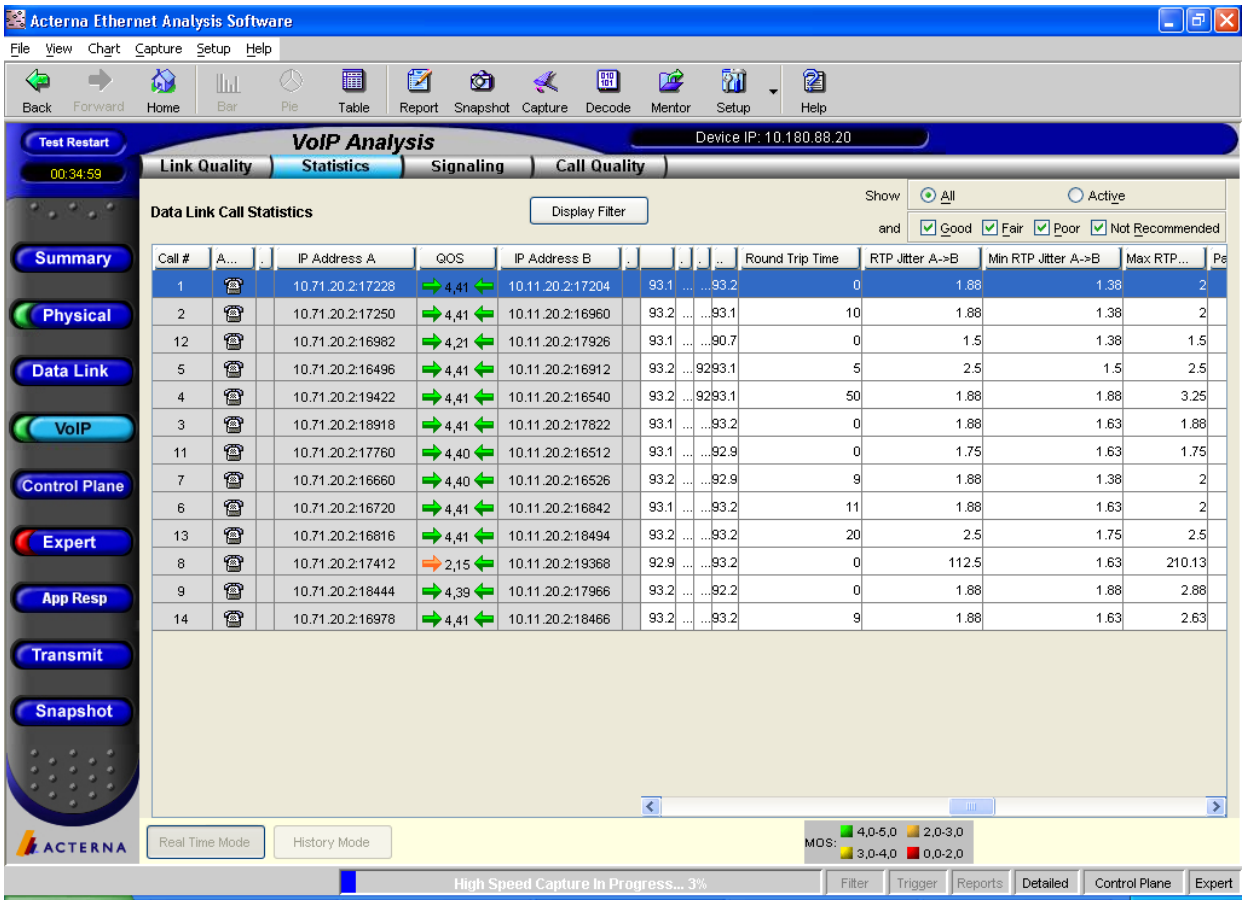


Figura 104: Analisador de protocolo Acterna número das chamadas.

O comportamento do tráfego é mostrado na figura 105, onde é observado que este está seguindo por apenas um único caminho, a interface de STM-1 entre o P1 ➔ PE1, deixando as outras duas interfaces de 34 Mbps subutilizadas. Na figura 106 pode ser visto que com o aumento do tráfego ocorre também um incremento nos alarmes de *jitter*, que são baseados nos parâmetros do analisador de protocolo.

É verificado na figura 106 que o total de alarmes de *jitter* críticos perfaz um somatório de 29, sendo esse muito maior em relação às chamadas VoIP que utilizam o protocolo SIP dos testes anteriores realizados.

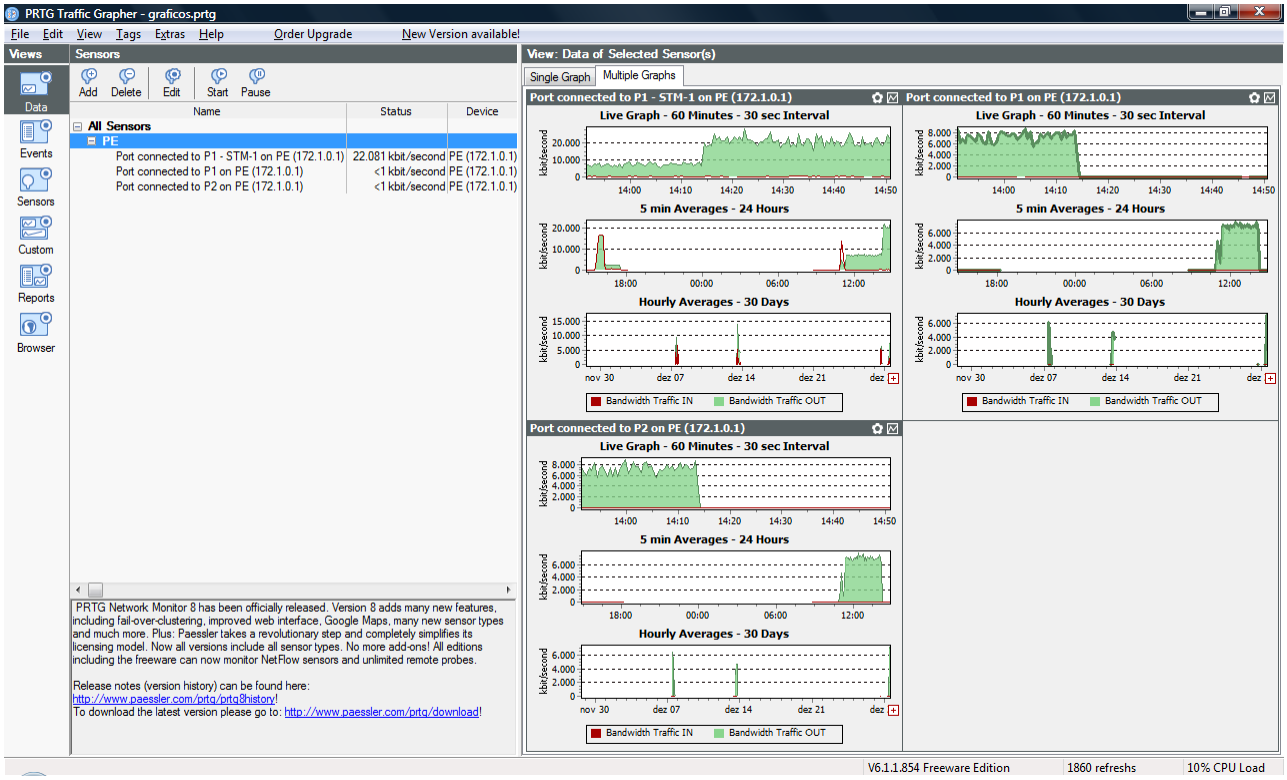


Figura 105: Incremento de tráfego seguindo apenas um único caminho.

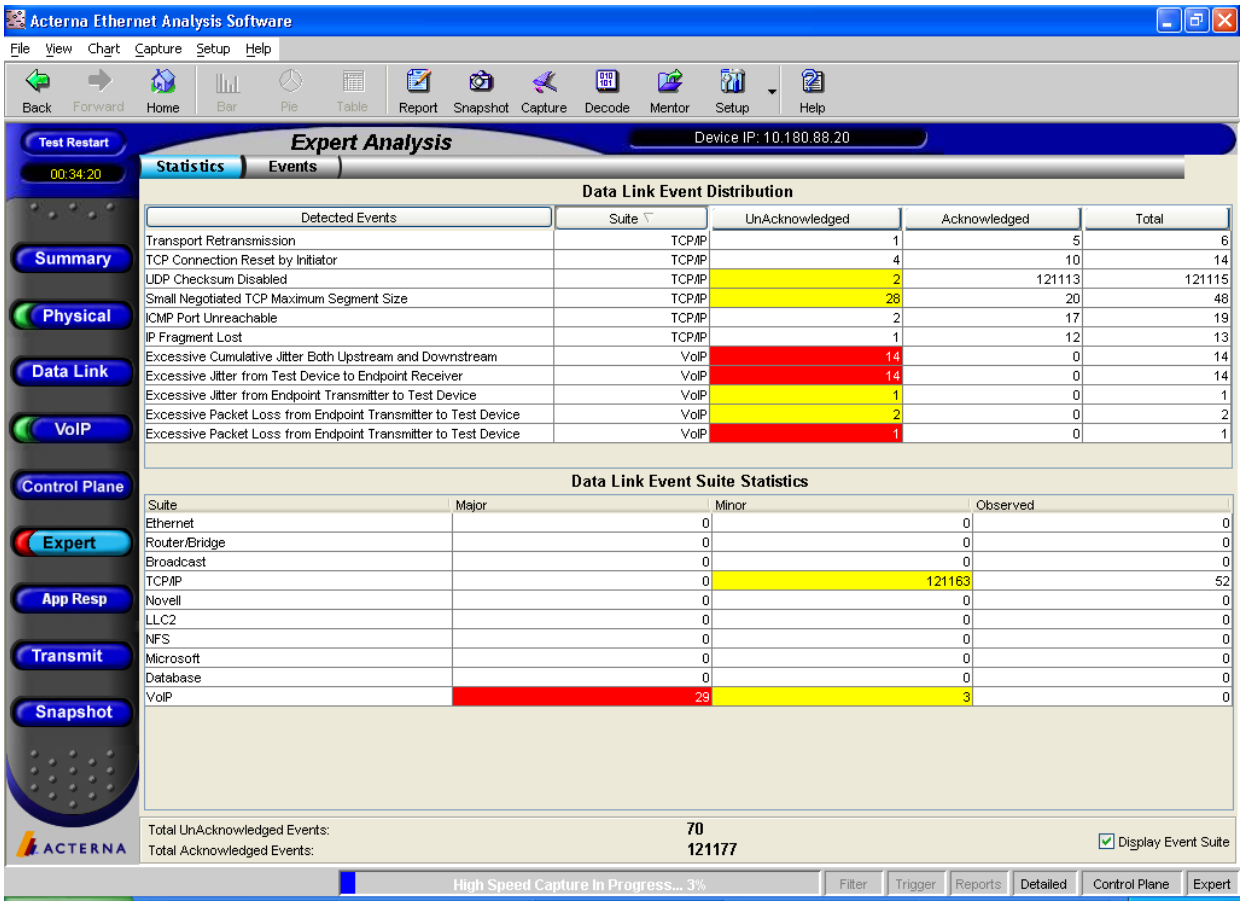


Figura 106: Incremento de alarmes de *jitter* baseado nos parâmetros do analisador.

A diferença entre essas chamadas VoIP feitas com telefones conectados a placas VIC FXS dos roteadores 1750 conectados nos CE12 e CE21 respectivamente e as feitas nos testes anteriores a esse é que elas não utilizam o protocolo SIP. No caso dessas novas chamadas é utilizado o protocolo H.225.0 que segue a recomendação ITU-T, a qual especifica o uso e suporte para mensagens de sinalização Q.931, como pode ser visto na figura 107.

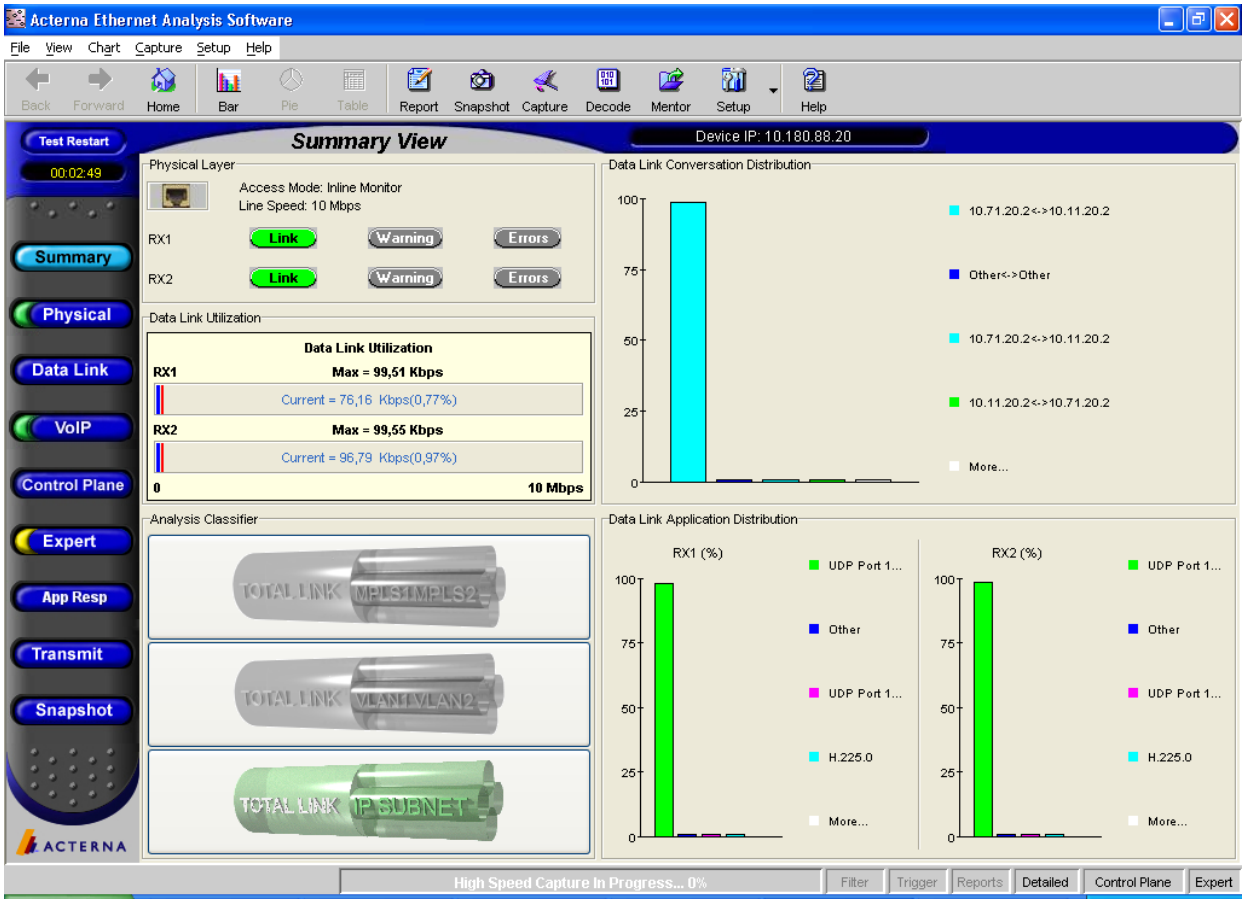


Figura 107: Detalhes do protocolo utilizado na chamada VoIP.

Foi refeito o teste agora aplicando o MPLS-TE e foi possível observar o balanceamento do tráfego na figura 108.

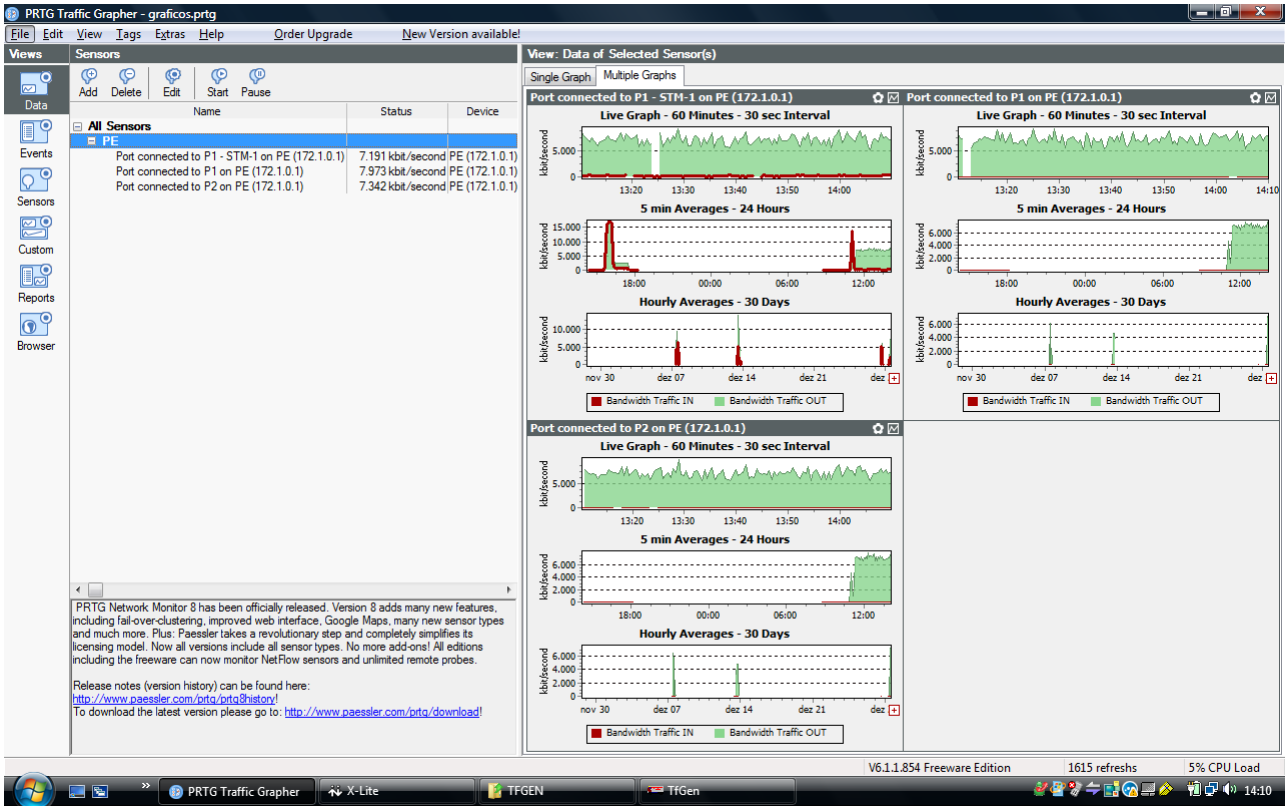


Figura 108: Balanceamento do tráfego.

Ainda pode ser observada na figura 109 a diminuição no número de alarmes de *jitter*, que neste caso o total de alarmes de críticos foi 17. Como mencionado no capítulo 6 desta dissertação, o MPLS-TE pode ajudar a reduzir os níveis de *jitter* e tal fato foi constatado com os experimentos realizados durante os cenários de testes, que apresentou a redução no número de alarmesde *jitter* nas chamadas VoIPs.

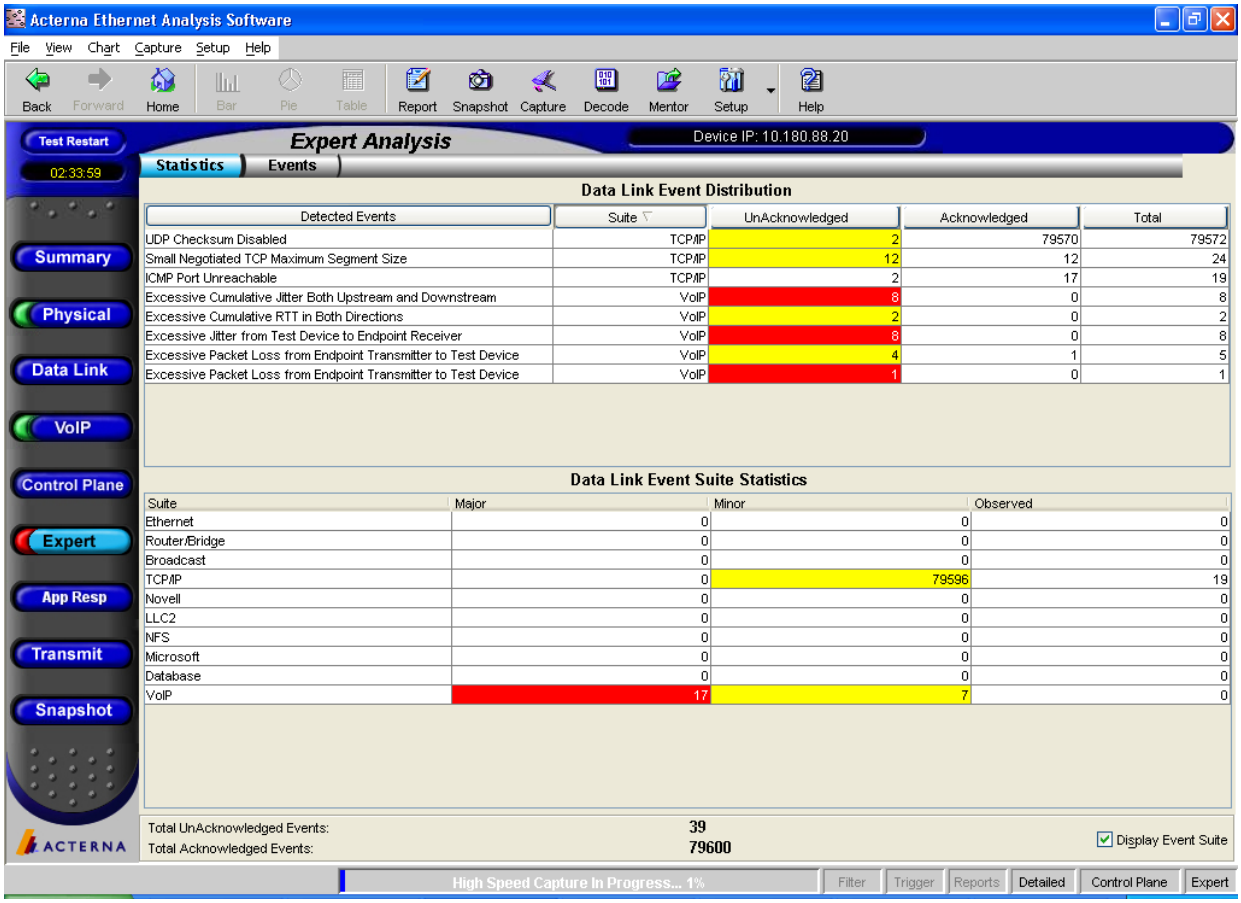


Figura 109: Alarmes de *jitter* de chamadas VoIP com protocolo H.225 utilizando o MPLS-TE

No gráfico apresentado na figura 110 foi feita uma comparação do número de alarmes de *jitter* sem MPLS-TE e com MPLS-TE. Neste ainda é possível comparar o quantativo de alarmes de *jitter* em chamadas VoIP-SIP com as chamadas VoIP-H.225.

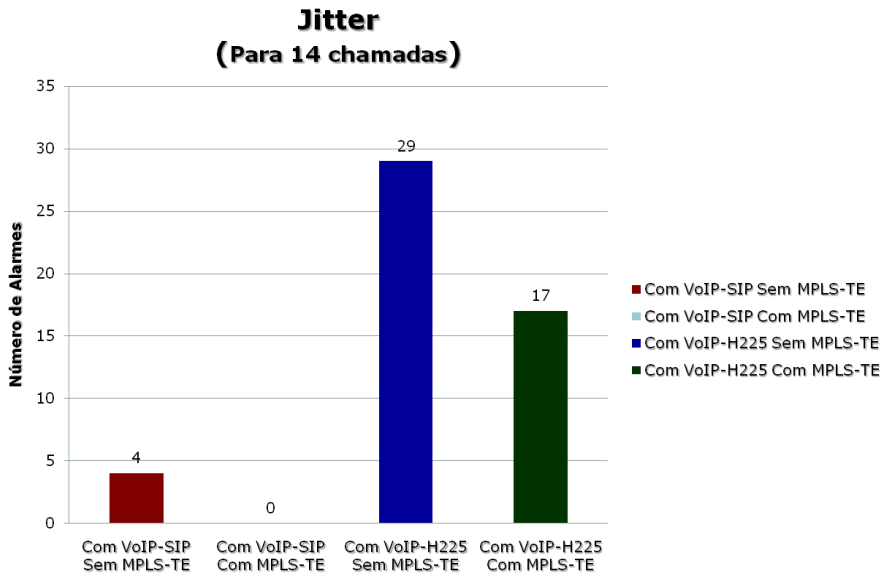


Figura 110: Comparação dos números de alarmes de *jitter* em ligações VoIP.

7.6 Considerações Finais

Neste capítulo foi feita a montagem do *backbone* representativo de uma operadora de telecomunicações, que foi utilizado para testes de comprovação da problemática do OSPF na distribuição do tráfego por diferentes caminhos de uma rede de pacotes IPs. Foi verificada, com os gráficos gerados pelo *software* de gerência de redes PRTG coletados através dos cenários de testes, a má distribuição do tráfego pelo protocolo de roteamento OSPF. Esse mesmo resultado pode ser validado com o analisador de protocolo da *Acterna* quando o mesmo foi utilizado para comprovar a supertutilização de apenas uma única interface do roteador de borda com os roteadores de núcleo do *backbone*.

Quando foi aplicada a funcionalidade do MPLS-TE foram feitos novos testes comparativos entre o protocolo de roteamento dinâmico OSPF, no encaminhamento de tráfego, e os túneis criados pelo MPLS-TE, para escoar o tráfego pelos diferentes caminhos. A partir dos gráficos gerados pelo *software* de gerência de redes PRTG foi verificado o balanceamento do tráfego pelos caminhos não utilizados no encaminhamento com OSPF. Com as informações geradas pelo *software* de gerência de redes PRTG foi possível consolidar em apenas um gráfico a comparação entre o encaminhamento com protocolo de roteamento OSPF e o serviço da boa engenharia de tráfego do MPLS.

Pode ser concluído que com o MPLS-TE é possível flexibilizar a passagem do tráfego por caminhos redundantes. Com isso, a operadora de telecomunicações obtem um ganho com transmissão de longa distância, pois os enlaces redundantes que estavam sem uso agora podem ser utilizados para balanceamento do tráfego, propocionando redução no tempo de envio de pacotes IPs no *backbone*.

Ainda foi utilizado o analisador de protocolo com objetivo de coletar alarmes de *jitter*. Nos cenários de testes montados foi observado que quando se tem um aumento do tráfego concomitante com as aplicações de VoIP ocorre um incremento desses alarmes, visto que o protocolo OSPF determina um único caminho para escoar o tráfego da rede. Desse modo, foi aplicado o MPLS-TE com a finalidade de obter parâmetros de comparação com OSPF. Com as medições realizadas utilizando o analisador de protocolo, foi possível verificar a diminuição nos números de alarmes de *jitter* quando utilizado o MPLS-TE.

Foram montados dois cenários envolvendo aplicações de VoIP com protocolos de sinalização diferentes. Nesses foram feitos testes com a utilização do OSPF para encaminhamento de tráfego e foram coletados os resultados da medição dos alarmes *jitter*, baseados nos parâmetros do analisador, para posperior comparação com a medição feita com a funcionalidade do MPLS-TE. Assim, foi possível comparar chamadas VoIP controladas com sinalização SIP e H.225.0 e verificar que o comportamento da chamada SIP apresenta um menor número de alarmes de *jitter*.

Capítulo 8 - Conclusões e Trabalhos Futuros

Nos dias atuais, com o constante aumento da integração entre serviços de voz e vídeo, o maior desafio das operadoras de telecomunicações é prover um *backbone* estável, escalável e otimizado. A finalidade disso é proporcionar aos seus clientes uma rede com maior disponibilidade, mas com menor custo. Sob esse contexto surge o MPLS (*Multi-Protocol Label Switching*), fazendo com que essas operadoras construam soluções para suprir a demanda por novos serviços.

Com a necessidade de atender a um mercado cada vez mais dominado pela comutação de pacotes e impulsionado pela Internet, as operadores de telecomunicações buscam reduzir custos de transmissão de longa distância migrando os seus serviços para uma rede IP. Devido à abrangência dos *backbones* das operadoras de telecomunicações, o roteamento puramente IP com os algoritmos de roteamento internos não atende a todos os requisitos necessários como, por exemplo, a distribuição do tráfego pelos diversos caminhos da rede.

O MPLS dispõe do serviço da boa engenharia de tráfego que pode ser aplicada a *backbones* de operadoras de telecomunicações fazendo com que haja uma distribuição dos tráfegos pelos diversos caminhos, trazendo ganhos para redes comutadas por pacotes. Essa tecnologia vem sendo utilizada para transporte de diversos serviços como, por exemplo, VoIP (*Voice over IP*), que uma aplicação que exige níveis de atraso constantes.

Observando como são encaminhados os pacotes em um *backbone* de uma operadora de telecomunicações que utiliza o OSPF (*Open Shortest Path First*) foi verificada a má distribuição dos recursos de transmissão e com essa problemática foi desenvolvida esta dissertação com a aplicação da funcionalidade da boa engenharia de tráfego que o MPLS pode provê.

Então esta dissertação apresentou a engenharia de tráfego sendo aplicada em um *backbone* IP MPLS de uma operadora de telecomunicações, com a aplicação prática do MPLS-TE (*Multi-Protocol Label Switching Traffic Engineering)*, mostrando a análise e a otimização dos recursos de transmissão e utilização de caminhos redundantes não utilizados pelo protocolo OSPF (*Open Shortest Path First*).

Com os cenários montados para os testes foi possível apresentar os ganhos da distribuição do tráfego pelo diversos caminhos e também constatar que o MPLS-TE pode reduzir a variação dos atrasos na rede.

A revisão de todo o conteúdo teórico é de grande valia, pois consolida o conhecimento de montagem de um *backbone* de uma operadora de telecomunicações, trazendo assim uma preparação para desenvolvimento de trabalhos relacionados a redes e telecomunicações.

A análise comparativa do roteamento IP e da comutação de rótulos do MPLS desenvolvida nesta dissertação pode gerar uma base de conhecimentos fundamentais para o desenvolvimento dos laboratórios aqui aplicados, onde foram obtidos os resultados da aplicação do MPLS. Foi possível apresentar o funcionamento do MPLS e aplicar a engenharia de tráfego no *backbone* de teste montado para aplicação de tal funcionalidade. Através das análises dos gráficos apresentados, fica constatado o balanceamento do tráfego por caminhos redundantes subutilizados, dessa forma foi otimizado o escoamento do tráfego por enlaces que estavam sem uso.

Apesar de ter sido aplicada a funcionalidade de engenharia de tráfego em um fragmento de um *backbone*, é possível escalar esta solução para grandes redes.

Vale observar que nesta dissertação não foi aplicada quaisquer melhorias possíveis de QoS apresentadas no capítulo 5, pois o objetivo é mostrar o balanceamento do tráfego por caminhos não utilizados pelo protocolo de roteamento dinâmico.

Foi constatada a redução dos níveis de *jitter* na aplicação do MPLS-TE nas redes com aplicações de VoIP controladas pelos protocolos SIP e H.225 e será possível desenvolver trabalhos futuros comparativos do SIP e do H.225, pois nos testes aqui aplicados foi observado que no SIP a redução foi total no número de alarmes já no H.225 a redução parcial.

Com o estudo aqui apresentado foi possível apontar para redes de próxima geração na qual o IP é a base do endereçamento lógico. O MPLS fará o transporte seja no núcleo da rede ou no acesso, pois como verificado é possível transportar qualquer serviço sobre MPLS, criar e agregar valor sobre os modelos de VPNs.

O MPLS possui outras aplicações como, por exemplo, MPLS VPNs, VPWS (*Virtual Private Wire Service*) e VPLS (*Virtual Private Lan Service*) que podem ser aplicadas e testadas e gerar trabalhos futuros.

As operadoras de telecomunicações de serviço móvel já estão migrando sua rede externa para uma rede comutada por rótulos baseado no protocolo MPLS, as chamadas

"*Mobile Backhaul Netwoks*". O IP/MPLS Forum considera bastante importante o uso de MPLS nesta rede e esse contexto está inserido no GMPLS, que pode ser considerado tema bastante importante desenvolvimento de trabalhos futuros.

Um estudo relacionado à aplicação de técnicas de QoS no *backbone* também pode ser citado como trabalhos futuros, o qual contribui para priorização do tráfego de determinadas aplicações na rede.

Ainda como trabalhos futuros, pode-se utilizar o protocolo IPv6 para implementar a engenharia de tráfego em *backbones* com MPLS e fazer comparativo de tempos de convergência de um *backbone* IPv4 com o IPv6, já as redes de próxima geração devem utilizar o IPv6 para o endereçamento.

Anexos

Aqui são apresentados os passos de configuração do MPLS para o funcionamento do *backbone* MPLS.

1.1 Configurações básicas para funcionamento do *backbone*

Será seguido o seguinte fluxo para configurações dos roteadores utilizados no *backbone*:

1. O primeiro passo será habilitar o CEF.

O CEF (*Cisco Express Forwarding*) é uma funcionalidade dos roteadores da *Cisco* [31], que é necessária para ativação do MPLS. A sua função básica é acelerar o processo de comutação dos roteadores, diminuindo a carga de processamento nesses. Ela é o componente principal da arquitetura *Tag Switching* da *Cisco*, e passa a ser fundamental a sua utilização nesse tipo de roteadores para ativação do MPLS. Essa função está ativada nos roteadores do *backbone* conforme o comando *ip cef* ou *ip cef distributed*:

- no PE1 – PE1(config)# ***ip cef [distributed]***
- no PE2 – PE2(config)# ***ip cef [distributed]***
- no P1 – P1(config)# ***ip cef [distributed]***
- no P2 – P2(config)# ***ip cef [distributed]***

2. O segundo passo é configurar o OSPF.

O protocolo de roteamento IGP escolhido para os testes foi o OSPF devido às comparações que devem ser feitas com o MPLS-TE. O comando "*redistribute static*" foi utilizado nesse caso, pois para a comunicação entre os CEs e os PEs foram utilizadas rotas estáticas, sendo necessário a redistribuição das mesmas dentro do *backbone* MPLS. O OSPF foi configurado nos seguintes roteadores:

- no PE1- PE1(config)#***router ospf 1***

  router ospf 1

router-id 172.1.0.1

log-adjacency-changes

auto-cost reference-bandwidth 10000

redistribute static

network 150.1.1.0 0.0.0.255 area 0

network 172.1.0.1 0.0.0.0 area 0

network 192.168.0.0 0.0.0.255 area 0

- no PE2 - PE2(config)# ***router ospf 1***

router ospf 1

router-id 172.2.0.1

log-adjacency-changes

auto-cost reference-bandwidth 10000

redistribute static

network 150.1.2.0 0.0.0.255 area 0

network 172.2.0.1 0.0.0.0 area 0

network 192.168.0.0 0.0.0.255 area 0

- no P1 – P1(config)# ***router ospf 1***

router ospf 1

router-id 172.10.0.1

log-adjacency-changes

auto-cost reference-bandwidth 10000

network 172.10.0.1 0.0.0.0 area 0

network 192.168.0.0 0.0.0.255 area 0

- no P2 –P2(config)# ***router ospf 1***

router ospf 1

router-id 172.20.0.1

log-adjacency-changes

auto-cost reference-bandwidth 10000

network 172.20.0.1 0.0.0.0 area 0

network 192.168.0.0 0.0.0.255 area 0

3. O terceiro passo é habilitar o protocolo de distribuição de rótulos.

O protocolo LDP de distribuição de rótulos deve ser configurado nos seguintes roteadores:

- no PE1 – PE1(config)# ***mpls label protocol ldp***

- no PE2 - PE2(config)# ***mpls label protocol ldp***
- no P1 – P1(config)# ***mpls label protocol ldp***
- no P2 – P2(config)# ***mpls label protocol ldp***

4. O quarto passo é configurar o MPLS nas interfaces dos roteadores PEs e Ps.

É necessário habilitar o comando para encaminhamento de rótulos nas interfaces entre dos roteadores do *backbone* da seguinte forma:

- no PE1 – PE1(config)# ***interface Serial2/0***

PE1(config-if)# ***tag-switching ip***

PE1(config-if)# ***interface Serial2/1***

PE1(config-if)# ***tag-switching ip***

- no PE2 - PE2(config)# ***interface Serial2/0***

PE2(config-if)# ***tag-switching ip***

PE2(config-if)# ***interface Serial2/1***

PE2(config-if)# ***tag-switching ip***

- no P1 – P1(config)# ***interface Serial1/0***

P1(config-if)# ***tag-switching ip***

P1(config-if)# ***interface Serial1/1***

P1(config-if)# ***tag-switching ip***

P1(config-if)# ***GigabitEthernet0/1***

P1(config-if)# ***tag-switching ip***

- no P2 – P2(config)# ***interface Serial1/0***

P2(config-if)# ***tag-switching ip***

P2(config-if)# ***interface Serial1/1***

P2(config-if)# ***tag-switching ip***

P2(config-if)# ***GigabitEthernet0/1***

P2(config-if)# ***tag-switching ip***

5. O quinto passo é a configuração do protocolo BGP.

O protocolo de roteamento BGP foi configurado nos roteadores PEs do *backbone* de testes, que é apenas um pequeno fragmento de um *backbone* de uma operadora de telecomunicações. Devido à simplicidade da topologia de testes não é necessário configuração de um roteador refletor de rotas. Os comandos aplicados nos roteadores PEs são:

- no PE1 – PE1(config)# ***router bgp 65500***

no synchronization
bgp log-neighbor-changes
network 150.1.1.0 mask 255.255.255.252
network 150.1.1.4 mask 255.255.255.252
redistribute static
neighbor 172.2.0.1 *remote-as* 65500
neighbor 172.2.0.1 *update-source Loopback* 0
neighbor 172.2.0.1 *next-hop-self*

- no PE2 – PE2(config)# ***router bgp 65500***

no synchronization
bgp log-neighbor-changes
network 150.1.2.0 mask 255.255.255.252
network 150.1.2.4 mask 255.255.255.252
redistribute static
neighbor 172.1.0.1 remote-as 65500
neighbor 172.1.0.1 *update-source Loopback* 0
neighbor 172.1.0.1 *next-hop-self*

Pelos mesmos motivos já explicados no OSPF foi implementado o comando *redistribute static*.

Também foi utilizado o comando "*next-hop-sef*", que força o BGP a usar o seu próprio endereço BGP como próximo salto, ao invés de deixar que o protocolo escolha o endereço do próximo salto a ser usado. Já o comando "*no synchronization*" desativa a sincronização do BGP, permitindo que um roteador use e anuncie para o BGP externo às rotas vizinhas aprendidas pelo iBGP antes de aprendê-las em um IGP.

A seguir são apresentadas as configurações aplicadas no backbone MPLS para o funcionamento dos cenários com MPLS-TE.

As configurações aplicadas nos roteadores foram:

No roteador P1:

```
hostname P1
!
boot system disk2:c7200-jk9s-mz.123-
17a.bin
!
ip cef
mpls label protocol ldp
mpls traffic-eng tunnels
!
interface Loopback0
 description connected to P1
 ip address 172.10.0.1 255.255.255.255
!
interface GigabitEthernet0/1
 description connected to P2
 ip address 192.168.0.29 255.255.255.252
 ip ospf flood-reduction
 duplex auto
 speed auto
 media-type rj45
 negotiation auto
 mpls traffic-eng tunnels
 tag-switching ip
 ip rsvp bandwidth 256 256
!
interface Serial1/0
 description connected to PE1
 ip address 192.168.0.14 255.255.255.252
 ip ospf flood-reduction
 mpls traffic-eng tunnels
 tag-switching ip
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
 ip rsvp bandwidth 256 256
```

```
!
interface Serial1/1
 description connected to PE2
 ip address 192.168.0.22 255.255.255.252
 ip ospf flood-reduction
 mpls traffic-eng tunnels
 tag-switching ip
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
 ip rsvp bandwidth 256 256
!
interface POS5/0
 description connected to PE1 -STM-1
 ip address 192.168.0.33 255.255.255.252
 mpls traffic-eng tunnels
 tag-switching ip
 ip rsvp bandwidth 256 256
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 router-id 172.10.0.1
 log-adjacency-changes
 auto-cost reference-bandwidth 10000
 network 172.10.0.1 0.0.0.0 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
snmp-server community public RW
```

No roteador P2:

```
hostname P2
!
boot system disk2:c7200-jk9s-mz.123-17a.bin
!
ip cef
mpls label protocol ldp
mpls traffic-eng tunnels
!
interface Loopback0
 description connected to P2
  ip address 172.20.0.1 255.255.255.255
 ip ospf flood-reduction
!
interface GigabitEthernet0/1
 description connected to P1
 bandwidth 1000000
 ip address 192.168.0.30 255.255.255.252
 ip ospf flood-reduction
 duplex auto
 speed auto
 media-type rj45
 negotiation auto
 mpls traffic-eng tunnels
 tag-switching ip
 ip rsvp bandwidth 256 256
!
!interface Serial1/0
 description connected to PE1
 bandwidth 34000
 ip address 192.168.0.18 255.255.255.252
 ip ospf flood-reduction
 mpls traffic-eng tunnels
 tag-switching ip
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
 ip rsvp bandwidth 256 256
!
```

```
!
interface Serial1/1
 description connected to PE2
 bandwidth 34000
 ip address 192.168.0.26 255.255.255.252
 ip ospf flood-reduction
 mpls traffic-eng tunnels
 tag-switching ip
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
 ip rsvp bandwidth 256 256
!

!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 router-id 172.20.0.1
 log-adjacency-changes
 auto-cost reference-bandwidth 10000
 network 172.20.0.1 0.0.0.0 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
snmp-server community public RW
```

No roteador PE1:

```
hostname PE1
!
boot system disk0:c7200-jk9s-mz.123-17a.bin

!
card type e1 3
enable password cisco
!
ip cef
mpls label protocol ldp
mpls traffic-eng tunnels
!

controller E1 3/0
 channel-group 0 timeslots 1-31
!

!
interface Loopback0
 description connected to PE1
 ip address 172.1.0.1 255.255.255.255
!
interface Tunnel1
 ip unnumbered Loopback0
 ip load-sharing per-packet
 shutdown
 tunnel destination 172.2.0.1
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 2 2
 tunnel mpls traffic-eng bandwidth  100
 tunnel mpls traffic-eng path-option 1 explicit
name PE1-P2-PE2
!
interface Tunnel2
 ip unnumbered Loopback0
 ip load-sharing per-packet
 shutdown
 tunnel destination 172.2.0.1
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 2 2
 tunnel mpls traffic-eng bandwidth  100
 tunnel mpls traffic-eng path-option 1 explicit
name PE1-P1-PE2
!
interface Tunnel3
 ip unnumbered Loopback0
 ip load-sharing per-packet
 shutdown
 tunnel destination 172.2.0.1
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 2 2
 tunnel mpls traffic-eng bandwidth  100
 tunnel mpls traffic-eng path-option 1 explicit
name PE1-STM-1-P1-PE2
```

```
interface POS1/0
 description connected to P1 - STM-
1
 ip address 192.168.0.34
255.255.255.252
 mpls traffic-eng tunnels
 tag-switching ip
 ip rsvp bandwidth 256 256
!
interface Serial2/0
 description connected to P1
 bandwidth 34000
 ip address 192.168.0.13
255.255.255.252
 mpls traffic-eng tunnels
 tag-switching ip
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
 ip rsvp bandwidth 256 256
!
interface Serial2/1
 description connected to P2
 bandwidth 34000
 ip address 192.168.0.17
255.255.255.252
 mpls traffic-eng tunnels
 tag-switching ip
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
 ip rsvp bandwidth 256 256
!
interface Serial3/0:0
 description connected to CE12
 ip address 150.1.1.5
255.255.255.252
!
interface Serial4/0
 description connected to CE11
 bandwidth 34000
 ip address 150.1.1.1
255.255.255.252
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 router-id 172.1.0.1
 log-adjacency-changes
 auto-cost reference-bandwidth
10000
 redistribute static
 network 150.1.1.0 0.0.0.255 area 0
 network 172.1.0.1 0.0.0.0 area 0
 network 192.168.0.0 0.0.0.255 area
0
!
```

```
ip classless
ip route 10.11.20.0 255.255.255.0
150.1.1.6
ip route 10.81.20.0 255.255.254.0
150.1.1.2
ip route 10.81.31.0 255.255.255.0
150.1.1.2
ip route 200.1.0.1 255.255.255.255
150.1.1.2
ip route 200.1.0.2 255.255.255.255
150.1.1.6
no ip http server
no ip http secure-server
!
!
ip explicit-path name PE1-P2-PE2
enable
 next-address 192.168.0.18
 next-address 192.168.0.25
 next-address 172.2.0.1
!
ip explicit-path name PE1-P1-P2-
PE2 enable
 next-address 192.168.0.14
 next-address 192.168.0.30
 next-address 192.168.0.25
 next-address 172.2.0.1
!
ip explicit-path name PE1-P1-PE2
enable
 next-address 192.168.0.14
 next-address 192.168.0.21
 next-address 172.2.0.1
!
ip explicit-path name PE1-STM-1-
P1-PE2 enable
 next-address 192.168.0.33
 next-address 192.168.0.21
 next-address 172.2.0.1
!
!
snmp-server community public RW
```

No roteador PE2:

```
hostname PE2
!
boot system disk:c7200-jk9s-mz.123-17a.bin
!
card type e1 3
!
ip cef
mpls label protocol ldp
mpls traffic-eng tunnels
!
controller E1 3/0
 channel-group 0 timeslots 1-31
!
interface Loopback0
 description connected to PE2
 ip address 172.2.0.1 255.255.255.255
 ip ospf flood-reduction
!
interface Multilink1
 description connected to CE21
 bandwidth 4096
 ip address 150.1.2.1 255.255.255.252
 ppp multilink
 ppp multilink group 1
!
interface Serial2/0
 description Connected to P1
 bandwidth 34000
 ip address 192.168.0.21 255.255.255.252
 ip ospf flood-reduction
 mpls traffic-eng tunnels
 tag-switching ip
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
 ip rsvp bandwidth 256 256
!
interface Serial2/1
 description connected to P2
 bandwidth 34000
 ip address 192.168.0.25 255.255.255.252
 ip ospf flood-reduction
 mpls traffic-eng tunnels
 tag-switching ip
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
 ip rsvp bandwidth 256 256
!
interface Serial3/0:0
 no ip address
 encapsulation ppp
 no fair-queue
 ppp multilink
 ppp multilink group 1
!
interface Serial3/1:0
 no ip address
 encapsulation ppp
 no fair-queue
 ppp multilink
 ppp multilink group 1
!
```

```
interface Serial4/0
 description connected to CE22
 bandwidth 34000
 ip address 150.1.2.5 255.255.255.252
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
!
interface Serial4/1
 no ip address
 shutdown
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 router-id 172.2.0.1
 log-adjacency-changes
 auto-cost reference-bandwidth 10000
 redistribute static
 network 150.1.2.0 0.0.0.255 area 0
 network 172.2.0.1 0.0.0.0 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
router bgp 65500
 no synchronization
 bgp log-neighbor-changes
 network 150.1.2.0 mask 255.255.255.252
 network 150.1.2.4 mask 255.255.255.252
 redistribute static
 neighbor 172.1.0.1 remote-as 65500
 neighbor 172.1.0.1 update-source Loopback0
 neighbor 172.1.0.1 next-hop-self
 no auto-summary
!
ip classless
ip route 10.1.1.0 255.255.255.0 150.1.2.2
ip route 10.71.20.0 255.255.255.0 150.1.2.2
ip route 10.80.20.0 255.255.255.0 150.1.2.6
ip route 10.180.88.0 255.255.255.224 150.1.2.6
ip route 10.181.88.0 255.255.255.224 150.1.2.6
no ip http server
no ip http secure-server
!
!
!
snmp-server community public RW
```

Nos roteadores CE11 e CE12:

```
hostname CE11
!
boot-start-marker
boot system disk2:c7200-jk9s-mz.123-17a.bin
!
interface Loopback0
 ip address 200.1.0.1 255.255.255.255
!
interface GigabitEthernet0/1
 ip address 10.81.20.1 255.255.254.0
 duplex full
 speed 100
 media-type rj45
 no negotiation auto
!
interface GigabitEthernet0/2
 ip address 10.81.31.1 255.255.255.0
 duplex full
 speed 100
 media-type rj45
 no negotiation auto
!
interface Serial2/0
 description Connected to PE1
 ip address 150.1.1.2 255.255.255.252
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
!
ip classless
ip route 0.0.0.0 0.0.0.0 150.1.1.1
!
snmp-server community public RW
!
```

```
hostname CE12
!
controller E1 1/0
 channel-group 0 timeslots 1-31
!
interface Loopback0
 ip address 200.1.0.2 255.255.255.255
!
interface Ethernet0/0
 ip address 10.11.20.1 255.255.255.0
 full-duplex
!
interface Serial1/0:0
 ip address 150.1.1.6 255.255.255.252
 no fair-queue
 serial restart-delay 0
!
ip classless
ip route 0.0.0.0 0.0.0.0 150.1.1.5
ip http server
!
snmp-server community public RW
```

Nos roteadores CE22 e CE21:

```
hostname CE21
!
card type e1 2
!
ip cef
!
controller E1 2/0
 channel-group 0 timeslots 1-31
!
controller E1 2/1
 channel-group 0 timeslots 1-31
!
interface Loopback0
 ip address 200.2.0.1 255.255.255.255
!
 interface Multilink1
 description connected to PE2
 bandwidth 4096
 ip address 150.1.2.2 255.255.255.252
 ppp multilink
 ppp multilink group 1
!
interface FastEthernet0/0
 ip address 10.71.20.1 255.255.255.0
 speed 100
 full-duplex
!
interface FastEthernet0/1
 ip address 10.1.1.120 255.255.255.0
 duplex auto
 speed auto
!
interface Serial2/0:0
 description connected to PE2
 no ip address
 encapsulation ppp
 serial restart-delay 0
 no fair-queue
 ppp multilink
 ppp multilink group 1
 !
interface Serial2/1:0
 description connected to PE2
 no ip address
 encapsulation ppp
 serial restart-delay 0
 no fair-queue
 ppp multilink
 ppp multilink group 1
!
ip classless
ip route 0.0.0.0 0.0.0.0 150.1.2.1
!
snmp-server community public RW
```

```
hostname CE22
!
boot system disk2:c7200-jk9s-mz.123-17a.bin
!
interface Loopback0
 ip address 200.2.0.2 255.255.255.255
!
interface GigabitEthernet0/1
 description LAN to Note Sony
 ip address 10.80.20.1 255.255.255.0
 duplex full
 speed 100
 media-type rj45
 negotiation auto
!
interface GigabitEthernet0/2
 ip address 10.180.88.1 255.255.255.224
 duplex auto
 speed auto
 media-type rj45
 negotiation auto
!
interface GigabitEthernet0/3
 ip address 10.181.88.1 255.255.255.224
 duplex auto
 speed auto
 media-type rj45
 negotiation auto
!
interface Serial2/0
 ip address 150.1.2.6 255.255.255.252
 framing g751
 dsu bandwidth 34010
 serial restart-delay 0
!
!
ip classless
ip route 0.0.0.0 0.0.0.0 150.1.2.5

!
snmp-server community public RW
```

Referências Bibliográficas

[1] TANEMBAUM, Andrew. Redes de Computadores. 4. ed. Rio de Janeiro: Elsevier, 2003.

[2] KUROSE, James; ROSS, Keith. Computer Networking: a top down approach featuring the Internet. 4. ed. Addison-Wesley, 2008.

[3] FOROUZAN, Benhrouz. Comunicações de Dados e Redes de Computadores. 4 ed. São Paulo: McGraw-Hill, 2008.

[4] PETERSON, Larry; DAVIE, Bruce. Redes de Computadores: uma abordagem de sistemas. 3. ed. Rio de Janeiro: Elsevier, 2004.

[5] OSBORNE, Eric; SIMHA, Ajay. Traffic Engineering with MPLS. Cisco Press, 2002.

[6] FARREL, Adrian. The Internet and Its Protocols: a comparative approach. Morgam Kaufmann, 2005.

[7] Disponível em: www.cisco.com, último acesso em 03/02/2011.

[8] ODOM, Wendell; CAVANAUGH, Michael. IP Telephony Self-Study Cisco DQOS. Cisco Press, 2004.

[9] ENNE, Antonio. TCP/IP sobre MPLS. 1 ed. Rio de Janeiro: Editora Ciência Moderna, 2009.

[10] Internet World Stats: http://www.internetworldstats.com/stats.htm, último acesso em 03/02/2011.

[11] SANTOS, Rodrigo; MOREIRAS, Antonio; DA ROCHA, Ailton. Curso de IPv6 básico. 1 ed. São Paulo: Comitê Gestor da Internet no Brasil, 2009.

[12] Disponível em: http://www.ipv6.com último acesso em 03/02/2011.

[13] http://sites.google.com/site/ipv6implementors/conference2009/agenda, último acesso em 02/02/2011.

[14] Disponível em: http://www.arbornetworks.com/IPv6research, último acesso em 02/02/2011.

[15] DOYLE, Jeff; CARROLL, Jennifer. Routing TCP/IP. 2. ed. CiscoPress, 2006. 1 v.

[16] PAQUET, Catherine; TEARE, Diane. Building Scalable Cisco Networks. Pearson Education, 2003.

[17] JACK, Terry. Building Cisco Multilayer Switched Networks. Alameda: Sybex, 2004.

[18] Disponível em: http://www.wired.com/threatlevel/2008/02/pakistans-accid/, último acesso em 02/02/2011.

[19] http://www.youtube.com/watch?v=IzLPKuAOe50, último acesso em 14/01/2011.

[20] MARTEY, A., STURGESS, S.. IS-IS Network Design Solutions. Cisco Press, 2002.

[21] Disponível em: http://tools.ietf.org/html/rfc5308, último acesso em 02/02/2011.

[22] Disponível em: http://www.faqs.org/rfcs/rfc3784.html, último acesso em 02/02/2011.

[23] Disponível em: http://tools.ietf.org/html/rfc5340, último aceso em 03/02/2011.

[24] NEWMAN, P.; EDWARDS, W. L.; HINDEN, R.; HOFFMAN, E.; CHING LIAW, F.;LYON, T.; MINSHALL, G. RFC 1953: Ipsilon Flow Management Protocol Specification for IPv4 Version 1.0. Network Working Group, 1996.

[25] REKHTER, Y.; DAVIE, B.; KATZ, D.; ROSEN, E.; SWALLOW, G. RFC 2105: Cisco Systems' Tag Swiching Architecture Overview. Network Working Group, 1997.

[26] DAVIE, B.; REKHTER, Y. MPLS Technology and Applications. Morgan Kaufmann, 2000.

[27] LUCEK, Julian; MINEI, Ina. MPLS - Enabled Applications: emerging developments and new technologies. Wiley, 2005.

[28] http://www.itu.int/itudoc/itu-t/aap/sg13aap/history/y1710/index.html, último acesso em 03/02/2011.

[29] Disponível em: http://standards.ieee.org/getieee802/802.1.html, último acesso em 03/02/2011.

[30] LOBO, Lancy. MPLS Configuration on Cisco IOS Software. Cisco Press, 2008.

[31] BOLLAPRAGADA, Vijay; MURPHY, Curtis; WHITE, Russ. Inside Cisco IOS Software Architecture. Cisco Press, 2000.

[32] ROSEN, E.; VISWANATHAN, A.; CALLON, R. RFC 3031: Multiprotocol Label Switching Architecture. The Internet Society, 2001.

[33] HARNEDY, S. The MPLS Primer. Prentice Hall, 2002.

[34] LUC DE GHEIN, MPLS Fundamentals. Indianapolis: Cisco Press, 2007.

[35] ROSEN, E.; TAPPAN, D.; FEDORKOW, G.; REKHTER, Y.; LI, T.; CONTA, A. RFC 3032: MPLS Label Stack Encoding. The Internet Society, 2001.

[36] DAVIE, Bruce; REKTER, Yakhov. MPLS Techology and Application. Morgam Kaufmann, 2000.

[37] Disponível em: http://tools.ietf.org/html/rfc3209.html, último acesso em 03/02/2011.

[38] Disponível em: http://www.apps.ietf.org/rfc/rfc3212.html, último acesso em 03/02/2011.

[39] Disponível em: http://www.faqs.org/rfcs/rfc2702.html, último acesso em 03/02/2011.

[40] Disponível em: http://www.ietf.org/rfc/rfc3630.txt, último acesso em 03/02/2011.

[41] ODOM, Wendell; CAVANAUGH, Michael. IP Telephony Self-Study Cisco DQOS. Cisco Press, 2004.

[42] SVERZUT, Jose. Redes Convergentes. 1 ed. Artliber, 2008.

[43] DAVIDSON, Jonathan; PETERS, James; BHATIA, Manoj; KALINDINDI, Satish; MUKHERJEE, Sudipto. Voice over IP Fundamentals. 2 ed. Cisco Press, 2007.

[44] CHOWDHURY, Dhiman. Projetos Avançados de Redes IP. 1 ed. Campus, 2002.

[45] COLCHER, SÉRGIO; GOMES, ANTONIO TADEU A.; SILVA, ANDERSON OLIVEIRA; FILHO, GUIDO L. DE SOUZA FILHO; SOARES, LUIZ FERNANDO G.: VoIP: Voz sobre IP 3ed; Ed. Campus. 2005.

[46] Disponível em: http://www.apps.ietf.org/rfc/rfc2474.html, http://www.ietf.org/rfc/rfc2474.txt, último acesso em 03/02/2011.

[47] Disponível em: http://www.ietf.org/rfc/rfc2475.txt, último acesso em 03/02/2011.

[48] Disponível em: http://www.ietf.org/rfc/rfc3260.txt, último acesso em 03/02/2011.

[49] Disponível em: http://www.ietf.org/rfc/rfc3260.txt, último acesso em 03/02/2011.

[50] Disponível em: http://www.ietf.org/rfc2597.txt, último acesso em 03/02/2011.

[51] Disponível em: http://www.ietf.org/rfc3246.txt, último acesso em 03/02/2011.

[52] Disponível em: http://www.ietf.org/rfc/rfc3985.txt, último acesso em 03/02/2011.

[53] Disponível em: http://www.ietf.org/rfc/rfc4447.txt, último acesso em 03/02/2011.

[54] Disponível em: http://www.ietf.org/rfc/rfc3270.txt, último acesso em 03/02/2011.

[55] Disponível em: http://www.ietf.org/rfc/rfc4364.txt, último acesso em 03/02/2011.

[56] Disponível em: http://www.ietf.org/rfc/rfc4760.txt, último acesso em 03/02/2011.

[57] Disponível em: http://www.ietf.org/rfc/rfc3107.txt, último acesso em 03/02/2011.

[58] Disponível em: http://www.ietf.org/rfc/rfc4397.txt, último acesso em 03/02/2011.

[59] GALLAHER, Rick. MPLS Training Guide. 1 ed. Syngress, 2003.

[60] Disponível em: http://www.ietf.org/rfc/rfc2858.txt, último acesso em 03/02/2011.

[61] Disponível em: http://www.ietf.org/rfc/rfc4136.txt, último acesso em 03/02/2011.

[62] GUIMARÃES, A.; LINS, R.; G., OLIVEIRA, R.. Segurança em Redes Privadas Virtuais. Editora Brasport Livros e Multimídia. 2006.

[63] Disponível em: http://delivery.acm.org/10.1145/230000/225992/p141-chandranmenon.pdf?key1=225992&key2=3073576921&coll=DL&dl=ACM&CFID=7757588&CFTOKEN=24834057, último acesso em 03/02/2011.