UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM TECNOLOGIA
CURSO DE ESPECIALIZAÇÃO EM CONFIGURAÇÃO E GERENCIAMENTO DE SERVIDORES E EQUIPAMENTOS DE REDE

JUNIOR CESAR CAETANO

IMPLEMENTAÇÃO DE TOPOLOGIA DE REDES UTILIZANDO MPLS

MONOGRAFIA

CURITIBA
2013

JUNIOR CESAR CAETANO

IMPLEMENTAÇÃO DE TOPOLOGIA DE REDES UTILIZANDO MPLS

Monografia apresentada como requisito para a obtenção do grau de Especialista em Configuração e Gerenciamento de servidores e equipamentos de redes. Universidade Tecnológica Federal do Paraná. Área de Concentração: Redes de Computadores
Orientador: Prof. MSc. Fabiano Scriptore de Carvalho

CURITIBA
2013

RESUMO

CAETANO, Junior C. I**mplementação de topologia de redes utilizando MPLS.** Monografia (Especialização em Configuração e Gerenciamento de Servidores e Equipamentos de Redes). Universidade Tecnológica Federal do Paraná. Curitiba, 2013.

A presente monografia tem como objetivo apresentar as principais caracteristicas existentes em redes MPLS (Multiprotocol Label Switching), com foco na implementação de Quality of Service (QoS). Na primeira parte deste trabalho são apresentados alguns conceitos básicos da tecnologia MPLS, juntamente com uma breve descrição do seu funcionamento. Também são descritos os principais servicos disponibilizados, como a criação de VPN (*Virtual Private Network*), o gerenciamento utilizando a Engenharia de Tráfego e a implementação de Qos em MPLS. Por fim, são apresentados alguns exemplos de implementação de Qos sobre MPLS.

Palavra-Chave: MPLS, engenharia de tráfego, qualidade de servicos.

ABSTRACT

Caetano, Junior C. **Implementation topology networks using MPLS**. Monograph (Specialization in Configuring and Managing Servers and Networking Equipment). Federal Technological University of Paraná.Curitiba, 2013.

This monograph aims to present the main features existents in MPLS (Multiprotocol Label Switching), focusing on the implementation of Quality of Service (QoS). In the first part of this paper presents some basic concepts of MPLS technology, along with a brief description of its operation. Also described are the main services available, such as the creation of a VPN (Virtual Private Network) management using Traffic Engineering and implementation of QoS in MPLS. Finally, some examples are presented to implement QoS over MPLS.

Keyword: MPLS, traffic engineering, quality of service.

LISTA DE SIGLAS

ATM – AsynchronousTransferMode
BGP – Border Gateway Protocol
CBWFQ – Class-based Weighted Fair Queueing
CSR – Cell Switching Routers
DLCI – Data-link connection identifier
DSCP – DiffServCodepoint
FEC – Forwarding Equivalency Class
FTP – File Transfer Protocol
HTTP – Hypertext Transfer Protocol
IETF – Internet Engineering Task Force
IOS – Internetwork Operating System
IP – Internet Protocol – Protocolo Internet
LFIB – Label Forwarding Information Base
LIB – Label Informations Base
LSP – Label Swith Path
LSR – Label Switch Routers
LLQ – Low-latency Queuing
MPLS – Multiprotocol Label Switching
MQC – Modular Quality of Service Command Line Interface
OSPF – Open Shortest-Path-First Protocol
QoS – Quality of Service
RFC – Request for Comments
RIP – Routing Information Protocol
RSVP – Reservation Protocol
TED – Traffic Engineering Database
TCP/IP – Transmission Control Protocol/Internet Protocol
TI – Tecnologia da Informação
TTL – Time to Live
VC – Virtual Circuit
VCI – Virtual Circuit Identifier
VPI – Virtual Path Identifier

VPN – Virtual Private Network

WRED – Weighted Random Early Detection

LISTA DE ILUSTRAÇÕES

SUMÁRIO

1INTRODUÇÃO

Nos últimos anos a Internet teve um crescimento exponencial no número de usuários e na demanda por maior largura de banda. Tradicionalmente os serviços de Internet disponibilizam para cada usuário serviços do tipo “melhor esforço”, sem levar em conta o tipo de aplicação utilizada (voz, dados e video). Como cada usuário recebe o mesmo nível de serviço, o congestionamento na rede muitas vezes resulta em séria degradação para aplicações que necessitam uma quantidade minima de largura de banda para funcionarem corretamente.

Devido ao crescente interesse em garantir a entrega de serviço em tempo real para algumas aplicações, como a telefonia IP, surgiu a necessidade de garantir algum nível de Qualidade de Serviço (QoS) na Internet. O protocolo IP apresenta algumas limitações, fruto de sua simplicidade original, que limitam a implementação de Qos nas redes baseadas neste protocolo.

O *Multiprotocol Label Switching* (MPLS) surge como uma tecnologia capaz de oferecer as potencialidades da engenharia de tráfego às redes baseadas em pacotes, fornece recursos para garantia de QoS sobre IP e permite a criação de VPN´s. Além disso é facilmente escalonável e possui interoperabilidade, ou seja, suporta redes com tecnologias distintas (Ethernet, ATM, Frame Relay, entre outras). (MCDYSAN; PAW, 2002).

Utilizando como base as diferentes tecnologias proprietárias existentes, como *IP Switching* da Nokia; o CSR – *Cell Switching Routers* da Toshiba; o TAG Switching da Cisco; o ARIS da IBM; o IP Navigator da Ascend; o Fast IP da 3Com, o MPLS surgiu em meados de 1997, *Internet EngineeringTask Force (*IETF), grupo internacional de padronização trabalhou para que fosse desenvolvida uma tecnologia padrão para a comutação de dados, que pudesse ser utilizada e implementada por qualquer fabricante. (MPLS: CONFORMENCE AND PERFORMANCE TESTING, 17 out 2007).

1.1 Justificativa

Devido ao avanço das tecnologias de comunicações, o número de usuários multiplica-se a cada dia e junto com esse crescimento existe a necessidade da rapida transmissão e da sua confiabilidade, podendo assim garantir a satisfação do cliente. O grande fator de se falar hoje dessa tecnologia(MPLS) é o fato da importância que ela se faz trazendo grandes benefícios que antes não se via e não podia ser atendidas. O MPLS surge como a principal tecnologia de viabilizar múltiplos serviços baseados em IP.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Implementar e analisar uma topologia de redes utilizando a tecnologia MPLS, verificando os requisitos fundamentais da tecnologia.

1.2.2 Objetivos Específicos

- Fazer um levantamento do estado da arte da tecnologia MPLS;
- Analisar os tipos de serviços oferecidos pelo MPLS;
- Implementar uma topologia utilizando a tecnologia MPLS;
- Fazer a análise das informações;

1.3 Roteamento IP

Em um ambiente tradicional de roteamento, os pacotes são encaminhados através da rede usando um algoritmo de roteamento nível 3 como RIP, OSPF, ou o BGP. Cada roteador que o pacote passa faz uma pesquisa no cabeçalho IP do pacote, esta pesquisa é feita para determiner qual o próximo *hop* que o pacote deve ser enviado para chegar ao seu destino final. Isto é feito pelo referenciamento do endereço de destino, contido no cabeçalho do pacote, emuma tabela de roteamento que aponta qual o próximo *hop.* Esta

pesquisa e refereciamento é a partir do cabeçalho, dependendo da complexidade da rede pode demandar muitos recursos de processador

1.4 Comutação de *labels*

Em uma rede MPLS o encaminhamento dos pacotes é baseado em *labels*, este encaminhamento funciona da seguinte forma: quando um pacote ingressa na rede MPLS ele recebe um *headers* MPLS que pode conter um ou mais*labels.* Os *labels*são associados a uma *Forward Equivalent Class*(FEC).

Uma FEC consiste numa classe de equivalência, ou seja, um conjunto de parâmetros, que irão determinar um caminho para os pacotes, assim, os demais roteadores irão somente substituir, ou seja, fazer um chaveamento, de *labels* até que o pacote chegue ao seu destino. Os pacotes associados a uma mesma FEC serão encaminhados pelo mesmo caminho. A FEC pode ser determinada por um ou mais parâmetros, especificados pelo gerente da rede. Alguns desses parâmetros são:

- Endereço IP da fonte ou destino ou endereço IP da rede;
- Número da porta da fonte ou destino;
- ID do protocolo IP;
- Qos desejado.

2 Referencial teórico

2.1 REDES DE COMPUTADORES

Uma rede de computadores pode ser definidocomo um grupo de computadores que são conectados entre si de forma que compartilhem arquivos e periféricos de forma simultânea utilizando meio de transmissão comum. Uma rede de computador pode ser composta de no mínimo 2 computadores conforme ilustra a figura 1.

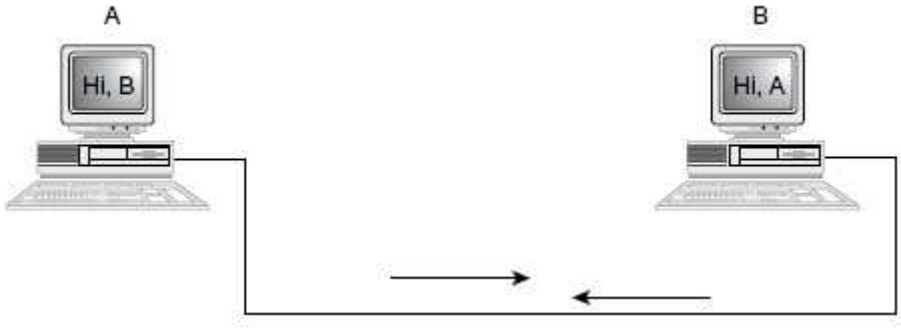


Figura 1. Rede de computadores

Fonte: http://www.juliobattisti.com.br/tutoriais/paulocfarias/redesbasico001.asp

A utilização de rede de computadores pode trazer uma certa economia na aquisição de *hardware*, por exemplo se existem 3 pessoas trabalhando em computadores destintos sem utilizarem uma rede e um necessita pegar um arquivo do outro, sera necessário sair do seu local de trabalho se dirigir até o outro computador e com um disquete ou outro dispositivo fazer uma cópia do arquivo para que o mesmo seja utilizado, caso necessite imprimir esse arquivo se a impressora estiver alocada em outro computador, ele terá que novamente se levantar ir até ao computador onde encontra-se instalada a impressora para que possa imprimir o arquivo. Nota-se a perda de tempo e desgaste que teria para realizar seu trabalho sem a utilização de uma rede de computadores. Isso sem contar que se fosse para agilizar o processo de impressão teria que adquirir uma impressora para cada usuário. Mas se estiverem conectados a

uma rede de computadores será necessário somente 1 impressora para a realização do trabalho e a possibilidade de compartilhamento de arquivos.

2.1.1 Modelo OSI

O modelo OSI é baseado em uma proposta desenvolvida pela ISO *(International Standards Organization).* Foi revisto em 1995 (Day, 1995). O modelo é chamado **Modelo de Referência ISO OSI (*Open systems Interconnection),*** ele trata da interconexão de sistemas abertos, sistemas que estão abertos à comunicação com outros sistemas.

O modelo OSI como é chamado mais tradicionamente tem sete camadas. Conforme mostra a figura 2.

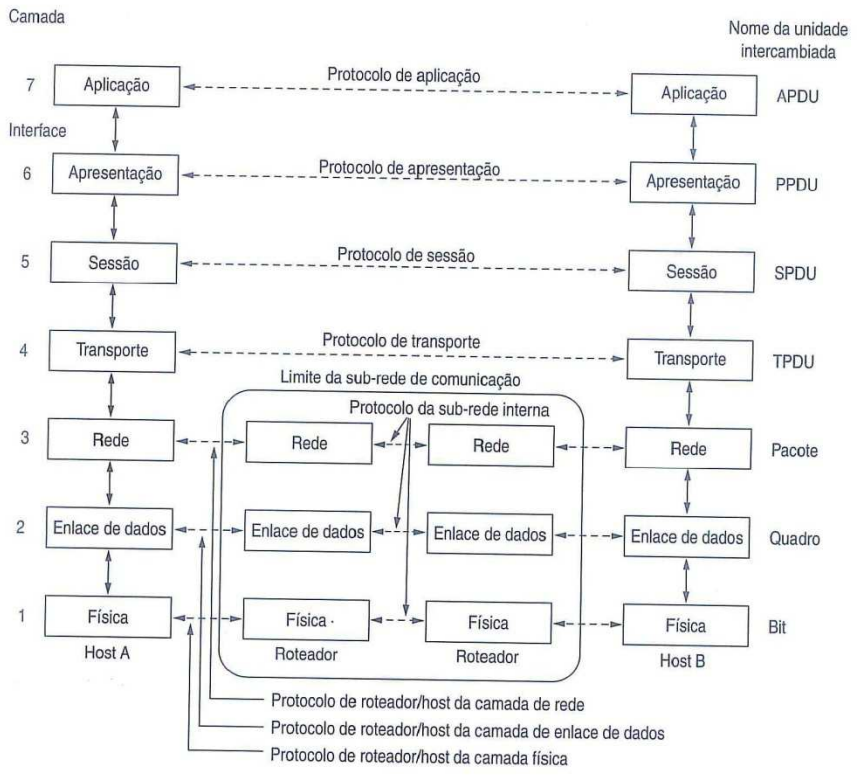


Figura 2: O modelo de referência OSI

Fonte: (TANENBAUM, ANDREW S., 2003)

2.1.2 CamadaFísica

A camada física é a primeira camada do modelo OSI e trata-se da transmissão e recepção de sequências de bits não processados nem estruturados sobre um suporte físico. As funções típicas dos protocolos deste nível são fazer com que um bit “1” transmitido por uma estação seja entendido pelo receptor como bit “1” e não como bit “0”. Dessa forma esse nível trabalha com as características mecânicas e elétricas do meio físico.

2.1.3 Camada de enlace de dados

O objetivo principal da camada de enlace de dados é fazer que um canal de transmissão bruto seja transformado em uma linha que pareça livre de erros de transmissão não detectados para a camada de rede. Para que essa tarefa seja executada, a camada de enlace de dados tem que fazer com que o transmissor divida os dados de entrada em quadros de dados e transmita os quadros sequencialmente. Caso o serviço seja confiável, o receptor confirmará a recepção correta de cada quadro, enviando um quadro de confirmação de volta.

2.1.4 Camada de rede

A camada de rede tem a função de controlar a operação da sub-rede. O roteamento de pacotes entre fonte e destino são suas principais funções. As rotas podem ser baseadas em tabelas estáticas, “amarradas” à rede e dificilmente alteradas. Se muitos pacotes estão sendo transmitidos através dos mesmos caminhos, eles vão diminuir o desempenho global da rede, formando gargalos. O controle desses congestionamentos também é tarefa da camada de rede.

2.1.5 Camada de transporte

A camada de transporte tem a função de aceitar dados da camada acima dela, dividi-los em unidades menores caso necessário, repassar essas unidades à camada de rede e assegurar que todos os fragmentos chegarão corretamente à outra extremidade. (TANENBAUM, ANDREW S., 2003, p.43).

A camada de transporte é a primeira que trabalha com conexões fim a fim, ou seja, um determinado programa na máquina fonte conversa com um programa similar na máquina destino, diferentes das camadas inferiores, que conversavam somente com o nó vizinho.

2.1.6 Camada de sessão

A camada de sessão tem a função de administrar e sincronizar diálogos entre dois processos de aplicação. Em determinadas aplicações, uma sessão permite o transporte de dados de uma maneira mais refinada que o nível de transporte.

2.1.7 Camada de apresentação

A camada de apresentação tem a função de assegurar que a informação transmitida seja entendida e usada pelo receptor. Esta camada pode modificar a sintaxe da mensagem, mas preserva a semântica. Um exemplo é que se uma aplicação gera uma mensagem em uma codificação diferente da interlocutora, a tradução entre os dois formatos é feito nessa camada.

2.1.8 Camada de aplicação

A camada de aplicação possui o maior número de protocolos, essa camada fornece aos usuários uma interface que permite diversos serviços de aplicação. Um protocolo de aplicação amplamente utilizado é o HTTP (*Hyper Text Transfer Protocol),* que constitui a base para a *World Wide Web.* (TANENBAUM, ANDREW S., 2003, p.44).

2.2 Modelo TCP/IP

No início dos anos 60, uma associação entre o DARPA (*Defense Advanced Research Projects Agency),* um grupo de universidades e algumas instituições, criaram o **“ARPANET Network Working Group”.** Em 1969, a rede **ARPANET** entrou em operação, consistindo inicialmente de quatro nós e utilizando comutação de pacotes para efetuar a comunicação.

Em 1974, um estudo feito por Vinton Cert e Robert Kahn, propôs um grupo de protocolos centrais para satisfazer as seguintes necessidades:

- Permitir o roteamento entre delesdiferentes (chamadas subnets ou subredes);
- Independência da tecnologia de redes utilizada para poder conectar as subredes;
- Independência do *hardware*;
- Possibilidade de recobrar-se de falhas.

Originalmente, esses protocolos foram chamados de NCP *(Network Control Program),* mas, em 1978, passaram a ser chamados de TCP/IP.

Em 1980, o DARPA começou a implementar o TCP/IP na ARPANET, dando origem à Internet. Em 1983, o DARPA finalizou a conversão de todos seus computadores e exigiuaimplementação do TCP/IP em todos os computadores que quisessem se conectar à ARPANET.

Além disso, o DARPA também financiou a implementação do TCP/IP como parte integral do sistema operacional Unix, exigindo que este fosse distribuído de forma gratuita. Dessa formao Unix e, consequentemente, o TCP/IP, se difundiram, cobrindo múltiplas plataformas.

Assim, o TCP/IP ficou sendo utilizado como o padrão de fato para interconectar sistemas de diferentes fabricantes, não apenas na Internet, mas em diversos ramos de negócios que requerem tal forma de comunicação. (http://www.abusar.org.br/ftp/pitanga/Aulas/a01_modelos.pdf).

2.2.1 Camada inter-redes

A camada de inter-redes tem a tarefa de permitir que os hosts injetem pacotes em qualquer rede e garantir o tráfego independentemente até o destino. Os pacotes podem chegar desordenados e assim obrigando as camadas superiores a reorganizá-los, caso a entrega em ordem seja realmente necessário. Esta camada também é responsável por receber os bits 0 e 1 da camada Internet, após o recebimento, os bits serão convertidos em tensões elétricas para que sejam enviados ao destino através dos cabos UTP ou STP.

2.2.2 Camada de transporte

No modelo TCP/IP a camada de transporte é localizada acima da camada inter-redes. Esta camada é responsável pela comunicação entre dois hosts, nela encontramos dois protocolos TCP e UDP.

TCP (*Transmission Control Protocol)* – é um protocolo orientado a conexão e confiável permitindo assim que uma determinada máquina em qualquer computador da inter-rede faça a entrega sem erros de um fluxo de bytes. Os roteadores que trabalham na camada inter-rede têm como único papel o encaminhamento dos dados sob a forma de datagramas, assim não precisam se preocupar com o controle dos dados porque essa função é realizada pela camada de transporte.

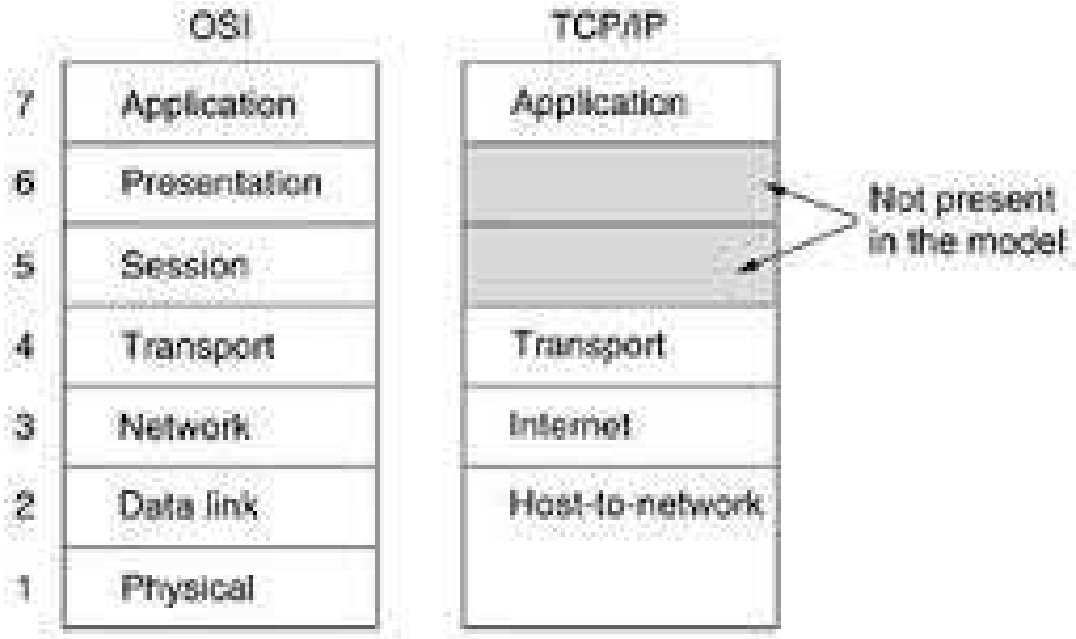


Figura 3. Modelo de referência TCP/IP
Fonte: (Tanenbaum, Andrew S., 2003)

UDP (*User Datagram Protocol)* não é um protocolo seguro porque não há nenhum dispositivo capaz de prover a confirmação de recebimento, com isso não existe garantia que os datagramas chegarão ao seu destino.

2.2.3 Camada de aplicação

A camada de aplicação está acima da camada de transporte. Nela encontramos todos os protocolos de nível mais alto. Dentre eles estão o protocolo de terminal virtual (TELNET), o protocolo de transferência de arquivos (FTP) e o protocolo de correio eletrônico (SMTP), como mostra a figura 4.

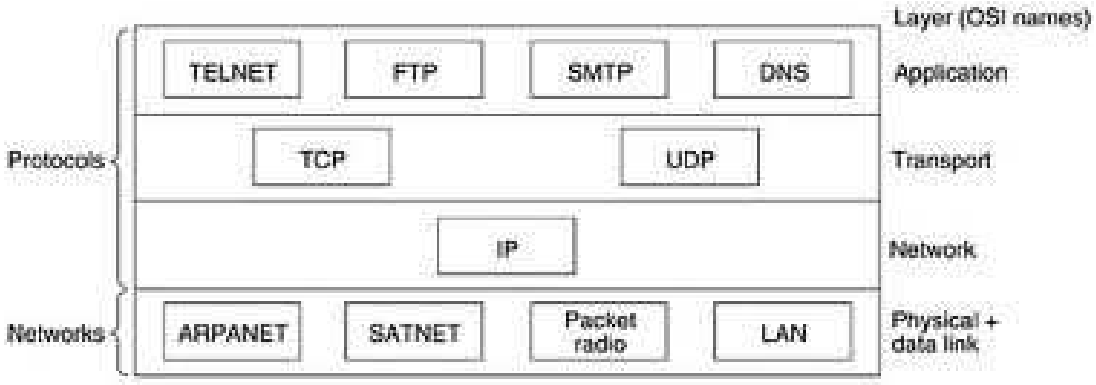


Figura 4. Protocolos e redes no modelo TCP/IP inicial

Fonte: (Tanenbaum, Andrew S., 2003).

O protocolo de terminal virtual permite que um usuário de um determinado computador se conecte em outra máquina distante para trabalhar nela. O protocolo de transferência de arquivos permite fazer transferência de arquivos com eficiência de uma máquina para outra. O protocolo de correio eletrônico era um tipo de transferência de arquivos, mais tarde foi desenvolvido um protocolo específico para essa função (SMTP).

2.3 Origens do MPLS

A técnica de comutação de *labels* não é nova, esta técnica já era utilizada em redes Frame Relay e ATM para transportar quadros e células através da rede. No Frame Relay o quadro pode ter qualquer comprimento, enquanto que no ATM a célula tem um tamanho fixo, com um cabeçalho de 5 bytes e um *payload* de 48 bytes. O cabeçalho da célula ATM e do quadro no Frame Relay fazem referência ao circuito virtual ao qual eles pertencem. A semelhança entre ATM e o Frame Relay é que o valor no cabeçalho pode ser alterado a cada *hop* atravessado na rede (GHEIN, 2007).

A comutação de *labels* utilize o *label* para executar uma pesquisa diretamente em uma entrada na tabela de conexões para determinar o próximo *hop,* executando uma operação com baixa utilização de recursos de *hardware* e uma elevada taxa de transmissão. O encaminhamento utilizando a comutação de *labels* também é considerado mais atrativo que o encaminhamento baseado no destino, pois permite que pacotes com o mesmo destino percorram diferentes fluxos. Por este motivo, a comutação de *labels*

tem sido considerada uma das melhores opções para implementação de engenharia de tráfego.

Na metade dos anos 90, os provedores de Internet construíram *backbones* de roteadores IP, interconectados através de uma rede de chaveamento de pacotes ATM que proporcionava uma conectividade de rede completa de forma a evitar a passagem por múltiplos *hops.* Esta abordagem forneceu a infra estrutura inicial para a Internet pública (McDYSAN; PAW, 2002). No entanto, este modelo de rede que utilizava IP sobre ATM tinha o inconveniente de que duas infra estruturas de rede tinham que ser gerenciadas separadamente, cada uma com seu proprio endereçamento, roteamento e sistema de gerenciamento. Consequentemente, diversas abordagens para integrar IP e ATM foram propostas (GARCIA; WIDJAJA, 2004).

Exemplos destas tecnologias são Tag Switch (Cisco), ARIS (IBM) e *Cell Switched Router* (Toshiba). Tratavam-se de tecnologias proprietárias, incapazes de interoperarem. Surgiu então a necessidade de um modelo padrão de comutação por *labels.* As tentativas de padronizar essas tecnologias através do IETF resultaram na combinação de várias tecnologias, gerando o *Multiprotocol Label Switching* (MPLS). Assim, não é surpresa que a implementação de comutação de *tag* da Cisco tivesse uma grande semelhança com o encaminhamento MPLS de hoje.

O MPLS usa a técnica de comutação de *labels* para encaminhar os dados através da rede. Um pequeno cabeçalho de formato fixo é inserido em cada pacote que entra na rede MPLS. Em cada *hop* através da rede, o pacote é encaminhado com base no *label* de entrada e enviado por uma interface de saída com o novo valor de *label.* O caminho que os dados fazem pela rede é definido pela transição feita nos valores do *label* que é alterado em cada LSR. Todo o caminho percorrido por um pacote é determinado pelo valor inicial do *label,* este caminho é chamado de *Label Switched Path* (LSP).

Ao entrar em uma rede MPLS, cada pacote é examinado para determinar a qual LSP o pacote vai ser associado e qual *label* vai ser inserido no pacote. Esta decisão pode ser baseado em fatores como: endereço de destino, requisitos de qualidade de serviço ou o estado atual da rede. O conjunto de todos os pacotes que são transmitidos da mesma forma é conhecido como *Forwarding Equivalence Class* (FEC) (OSBORNE, 2002).

A motivação real para a implantação do MPLS e toda a complexidade adicional emuma rede está na aplicação das funcionalidades existentes, que são dificeis de realizar em redes IP tradicionais. As duas principais funcionalidades do MPLS são a engenharia de tráfego e a criação de VPN´s (OSBORNE, 2002).

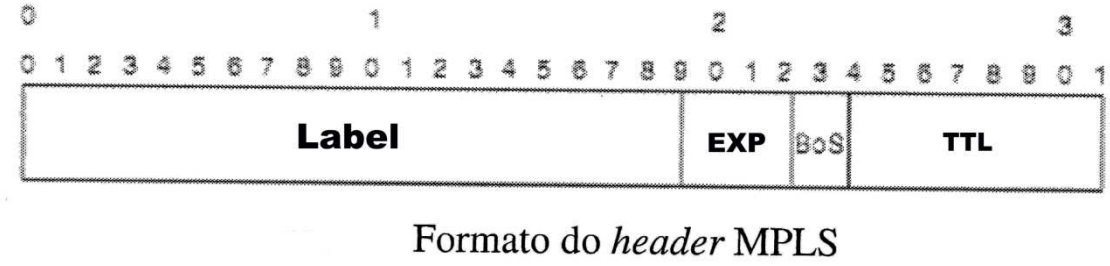
2.4 Conceitos e componentes MPLS

Nesta sessão serão apresentados alguns conceitos e definições dos principais componentes que fazem parte de uma rede MPLS.

2.4.1 Labels

O cabeçalho MPLS é um identificador de 32 bits que é usado no encaminhamento do pacote. É descrito na RFC 3031 "*Multiprotocol Label Switching Architecture"* como "um identificador curto, de tamanho fixo e localmente significativo que é utilizado para identificar uma FEC".

O formato do *header* é mostrado na Figura 5



Formato do *header* MPLS

Figura 5 – Formato do cabeçalho MPLS
Fonte:(KAKIHARA, 2006)

Os seguintes campos fazem parte do cabeçalho*:*

- *Label* (20 bits): valor atual do *label,* identificador de LSP;
- EXP (3 bits): experimental bits – pode ser usado para filas de espera, rejeição, QoS etc;

- BoS (1 bit): bit de sinalização de fim de *stack*, esse valor é setado para 1 para a ultima entrada na pilha e 0 para as demais;
- TTL (8 bits): *time tolive*, possui a mesma função do TTL do cabeçalho IP.

O MPLS suporta três diferentes tipos de cabeçalho. Em redes ATM ele utiliza os campos VPI/VCI de cada célula e em redes Frame Relay ele utilize o campo DLCI (*Data-link connection identifier)* de cada quadro. Nas tecnologias que não carregam *labels*, como o Ethernet, é inserido um pequeno campo adicional ao cabeçalho do pacote, entre os cabeçalhos da camada de enlace e camada de rede, denominado "*shim header*".

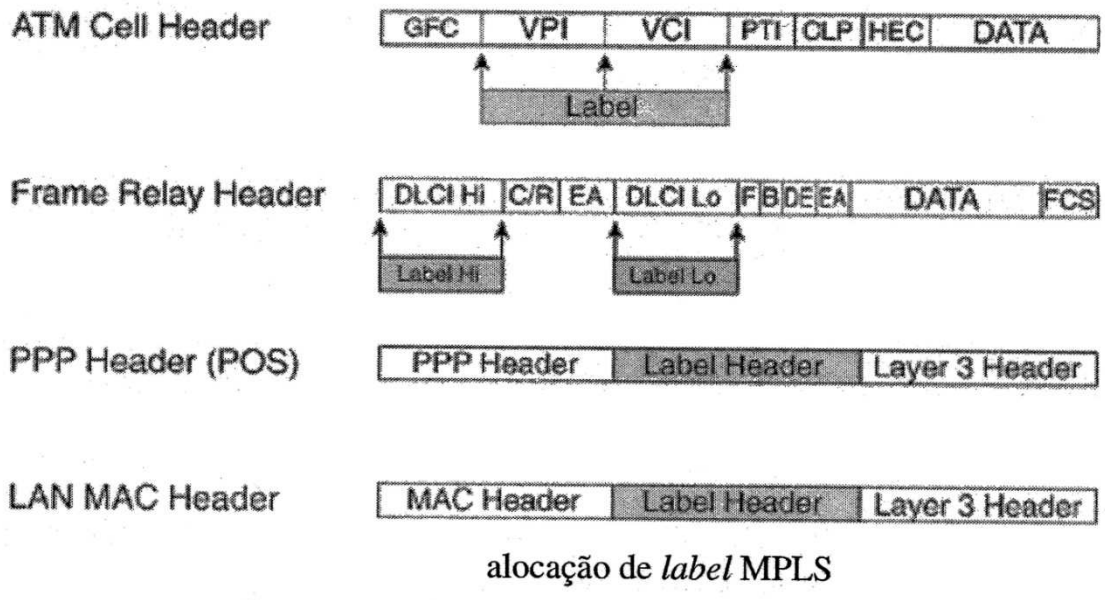


Figura 6 – Alocação de Label MPLS

Fonte: (KAKIHARA, 2006)

2.4.2 Empilhamento de *labels*

Roteadores MPLS podem precisar mais de um *label* no pacote para roteá-lo através da rede MPLS. O mecanismo de empilhamento de *labels* permite operações hierárquicas no domínio MPLS. Isto significa que cada nível em uma pilha de *labels* corresponde a um nível hierárquico.

O primeiro *label* na pilha é chamado de *"top label" e o ultimo é chamado de* "*bottom label",* sendo que entre eles pode-se ter qualquer quantidade de*labels.* Na pilha de *labels* todos os bits do campo BoS têm o valor 0, exceto pelo *"bottom label"* que tem o valor 1 para indicar o fim da pilha. A figura7 mostra a estrutura da pilha de *labels.*

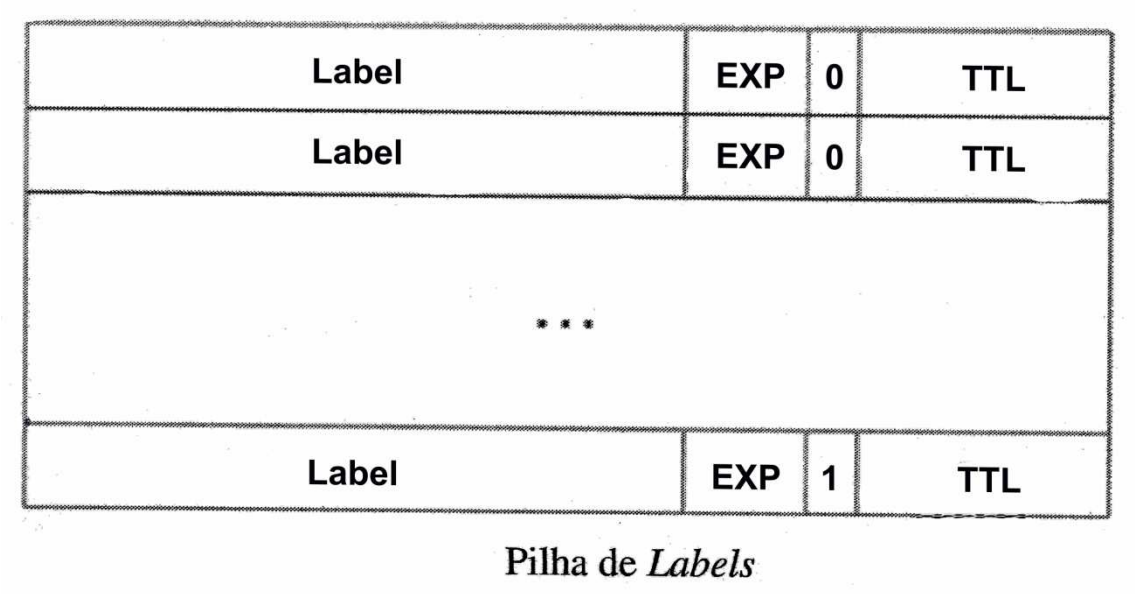


Figura 7 - Pilha de *Labels*

Fonte: (GHEIN 2007)

2.4.3 Label switch router (LSR)

Um LSR é um roteador com suporte MPLS. Ele é capaz de entender pacotes com *labels* MPLS e de receber e transmitir estes pacotes. Existem dois tipos de LSR em redes MPLS que podem ser classificados em LSR de borda (*Core* LSR) e LSR de núcleo (*Edge* LSR).

Um LSR de borda, situado na entrada de uma rede MPLS, é responsável por inserir um ou mais *labels* ao pacote, associá-lo a uma FEC (*Forwarding Equivalency Class*) e encaminhar o mesmo através de um LSP (*Label Switched Path*). Quando está situado na saída é responsavel por remover os *labels* e encaminhar o pacote para uma rede não MPLS.

O LSR de núcleo faz o trabalho de receber e encaminhar os pacotes MPLS baseado no *label,* através de um LSP. Cada LSR recebe o pacote, troca o *label* e encaminha para o LSR seguinte até chegar ao LSR de borda.

2.4.4 Label switched path (LSP)

Um LSP é uma sequência de LSR que encaminham um pacote rotulado por meio de uma rede MPLS, ou seja, é o caminho que um pacote percorre dentro de uma rede MPLS.

Na figura 8 é mostrada uma seta indicando o fluxo em um LSP (unidirecional). Para um fluxo de pacotes na direção contrária e entre os mesmos pontos é necessário outro LSP.

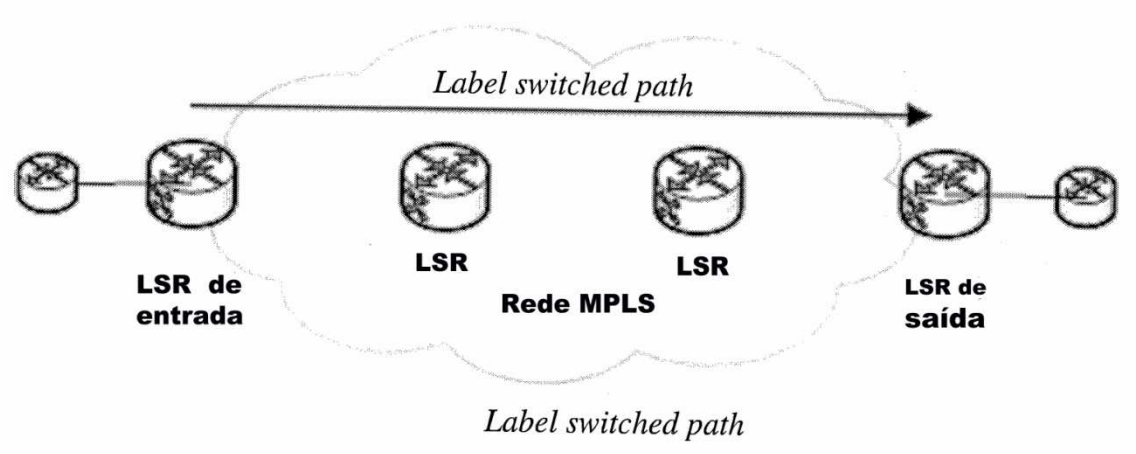


Figura 8: *Label switched path*
Fonte: (GHEIN2007)

Um LSR de entrada não é necessariamente o primeiro roteador a colocar um *label* no pacote, os pacotes podem ter sido marcados por um LSR

precedente. Este caso é chamado de “LSP aninhado”, isto é, um LSP dentro de outro LSP. Na figura 6 pode ser visto um LSP que abrange toda a extensão de uma rede MPLS e outro LSP que tem início no LSR 1 e termine no LSR 4 então, quando o pacote entra no segundo LSP ele já está rotulado, este roteador coloca um segundo *label* no topo da pilha do pacote. Este *label* inserido no pacote pode ter um valor de QoS diferente, isto significa que um mesmo pacote pode ter diferentes valores de QoS em cada LSP.

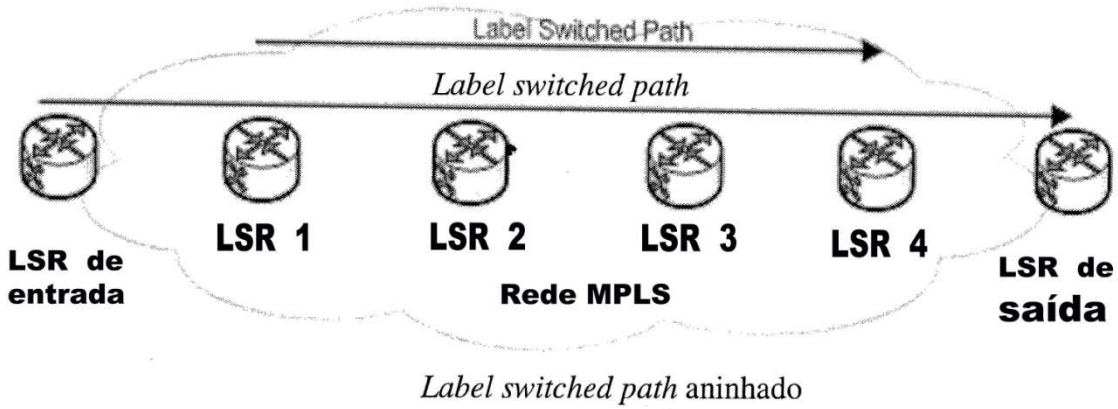


Figura 9: *Label switched path aninhado*
Fonte: (GHEIN 2007)

2.4.5 Forwarding equivalence class (FEC)

Uma FEC é um grupo ou fluxo de pacotes que são encaminhados por meio de um mesmo caminho e são tratados da mesma forma no que diz respeito ao tratamento do encaminhamento. Todos os pacotes pertencentes a mesma FEC têm o mesmo *label.* Cada LSR constrói uma tabela com a especificação de como um pacote deve ser enviado, esta tabela é chamada de *Label Information Base* (LIB).

2.4.6 Distribuição de *labels*

O primeiro *label* é colocado pelo LSR de borda, este *label* indica que o

pacote pertence a uma determinada FEC. Os LSRs de núcleo recebem os pacotes com *labels*, trocam este *label* de entrada por um *label* de saída e encaminham o pacote. Quando um LSR de borda atribui um *label* de uma determinada FEC a umpacote é necessário que ele consiga comunicar os LSR relevantes sobre este *label* e o seu significado para que estas informações sejam usadas na construção das tabelas de encaminhamento. Isto significa que um mecanismo de distribuicao de *labels* é necessário.

A distribuição de *labels* pode ser feita de duas maneira:

a) Transportando os *labels* em um protocol de roteamento existente. A vantagem é que não é necessário um novo protocolo rodando nos LSR. As informações dos *labels* podem ser adicionadas em versões estendidas dos protocolos tradicionais de roteamento como o BGP ou o RSVP. A grande vantagem de ter o protocolo de roteamento transportando os *labels* é que o roteamento e a distribuição dos *labels* estão sempre em sincronia.

b) Usando um protocolo de distribuição de *labels*. Este método tem a vantagem da existência de um protocolo de roteamento independente: qualquer que seja o protocolo de roteamento utilizado, se ele tem a capacidade de distribuição de *labels* ou não, um protocolo separado é utilizado para a tarefa de distribuição de *labels*. O protocolo definido pelo IETF para executar esta função foi o *Label Distribution Protocol* (LDP). LDP tem quatro funções principais (GHEIN, 2007):

- A descoberta dos LSRs que estão executando o LDP;
- O estabelecimento e a manutenção de sessões;
- O anúncio de mapeamento de *labels*;
- A manutenção de sessões LDP por meio de notificação.

2.4.7 Tabelas de Encaminhamento

Cada LSR mantém duas tabelas relevantes para o encaminhamento dos

pacotes MPLS: a LIB, que contém todos os *labels* atribuídos pelo nó local de MPLS, e o mapeamento destes com os *labels* recebidos de seus vizinhos que utilizam o MPLS e a LFIB que usa um subconjunto dos *labels* contidos na LIB para o atual encaminhamento.

A LIB é uma tabela que é construída por cada LSR para especificar como os pacotes devem ser encaminhados. Ela contém informações sobre a associação das ligações de *labels* negociados com outros roteadores MPLS (McDYSAN& PAW, 2002).

A LFIB, mantida por um nó MPLS, consiste de uma sequência de entradas. Cada entrada consiste de um label de entrada e de uma ou mais sub-entradas. A LFIB é indexada pelo valor contido no *label* de entrada. Cada sub-entrada consiste de um *label* de saida, interface de saida e endereço do próximo *hop.* Sub-entradas contidas dentro de uma entrada individual pode ter o mesmo ou diferentes *labels* de saida.

Quando um LSP é criado, a relação dos *labels* com a interface será armazenada na tabela LFIB (*Label Forwarding Base*). O pacote entra no LSR e este, por sua vez, verifica na LFIB para qual interface deve ser encaminhado. Então, realize a troca do *label* de entrada por um *label* de saida, de maneira que o pacote possa alcançar o próximo nó.

O processo de preenchimento do LFIB pode ser controlado por meio de configuração ou por meio de protocolos de distribuição de *labels* e, para evitar laços, pacotes com *labels* inválidos são descartados (FARREL; BRYSKIN, 2006).

2.4.8 VPN, QOS E ENGENHARIA DE TRÁFEGO EM MPLS

Serão apresentados os conceitos dos principais serviços Implementados sobre redes MPLS.

2.4.9 VPN

Virtual Private Network (VPN), ou Rede Privada Virtual, é uma rede privativa construída sobre a infra-estrutura de uma rede pública, como a

Internet. São usadas a criptografia e a autenticação para proteger os dados, enquanto estes estiverem em trânsito.

A segurança é a primeira e mais importante das funções das VPNs. Uma vez que dados privados serão transmitidos pela Internet, que é um meio de transmissão inseguro, eles devem ser protegidos de forma a não permitir que sejam modificados ou interceptados.

Outro serviço oferecido pelas VPNs é a conexão entre corporações (Extranets) através da Internet, além de possibilitar conexões *dial-up* criptografadas que podem ser muito úteis para usuários móveis ou remotos, bem como filiais distantes de uma empresa.

A topologia de uma VPN é dividida em dois modelos:

- Modelo *overlay,* onde o provedor de serviços permite a interconexão de múltiplas localidades através de sua rede WAN, que aparece como “privativa” para o cliente.
- Modelo *peer-to-peer,* onde o provedor de serviços e o cliente trocam informações sobre o roteamento e o provedor transmite os dados do cliente utilizando o melhor caminho entre os sites, sem o envolvimento do cliente.

2.4.10 Modelo *overlay*

O modelo *overlay* é o mais fácil de entender, pois ele exibe uma clara separação de responsabilidades entre o cliente e o provedor de serviços.

O provedor de serviços oferece ao cliente um conjunto de linhas. Estas linhas são chamadas de VCs, que podem estar constantemente disponíveis (PVCs) ou estabelecidas sob demanda (SVCs).

O cliente estabelece uma comunicação entre os seus roteadores, sobre os VCs fornecidos pelo provedor do serviço. O protocolo de roteamento é sempre trocado entre os roteadores do cliente, e o provedor não tem conhecimento da estrutura interna da rede do cliente.

2.4.11 Modelo peer-to-peer

O modelo *peer-to-peer* foi introduzido com o objetivo de minimizar os inconvenientes do modelo *overlay*. No modelo *peer-to-peer* o equipamento de borda do provedor é um roteador, que troca informações de roteamento diretamente com o roteador do cliente.

Este modelo é mais simples porque os roteadores do provedor têm o conhecimento da topologia de rede do cliente, tornando mais simples o trabalho de incorporação de novas localidades numa rede *fullmesh*, em comparação com aquele demandado em redes *overlay*. O roteador de borda do provedor pode ser dedicado ou compartilhado por VPN´s de clientes diferentes. Em qualquer dos casos, não é possível o isolamento do tráfego nem o uso de endereçamento privado nas redes dos clientes, pois os endereços IP devem ter significância global no *backbone* do provedor.

2.4.12 Modelo MPLS

Com a introdução do MPLS, que combina os benefícios de comutação da camada de enlace e o roteamento e comutação da camada de rede, foi possível construir uma tecnologia que combina os benefícios do modelo *overlay*, tais como a segurança e isolamento entre clientes, com os benefícios da simplificação de roteamento que uma VPN *peer-to-peer* traz. Esta nova tecnologia torna possível a criação de diferentes topologias, difíceis de implementar nos modelos *overlay* e *peer-to-peer.*

A construção de VPNs é uma das implementações mais comuns que fazem uso da tecnologia MPLS. Sua popularidade tem crescido exponencialmente desde que foi proposta. VPNs sobre MPLS podem fornecer escalabilidade e dividir a rede em pequenas subredes separadas, o que muitas vezes é necessário em grandes redes corporativas, onde a infra-estrutura de TI precisa oferecer o isolamento de redes de diferentes departamentos (PEPELNJAK; GUICHARD, 2007).

A sobreposição de endereços, normalmente resultante do uso de endereço IP privados dos clientes, é um dos maiores obstáculos para a implementação de VPNs *peer-to-peer*. A implementação de VPN sobre MPLS oferece uma solução elegante para este dilema: cada VPN tem sua própria tabela de roteamento e encaminhamento no roteador, de tal forma que, para um cliente que pertença a uma VPN sera fornecido acesso somente ao conjunto de rotas contido na tabela correspondente.

2.5 QoS

Quality of Service (QoS) é um tema que tem se tornado muito popular nos últimos anos Refere-se a capacidade da rede em priorizar um determinado tipo de tráfego, considerado mais importantes, sobre um tráfego menos importante, além da garantia de entrega.

Para que se possa garantir QoS em uma rede, todos os pacotes de dados pertencentes a uma mesma sessão devem seguir o mesmo caminho (como em um tráfego orientado a conexão) e devem existir meios de garantir a reserve de recursos ao longo deste caminho. O tráfego IP não é orientado a conexão e os roteadores geralmente não têm recursos sofisticados para reservar recursos a cada *hop*. Por isso, a garantia de QoS em uma rede IP é tão difícil. Para tentar resolver este problema, o IETF desenvolveu dois mecanismos para implementar QoS em uma rede IP: *Integrated Services* (IntServ) e *Differentiated Services* (DiffServ).

O IntServ utilize o protocolo *Resource Reservation Protocol* (RSVP) para reservar recursos para determinados fluxos de dados. Na sinalização RSVP existe troca de mensagens de controle entre emissor e receptor de forma que, em um determinadoperiodo de tempo, estará alocada uma parte da banda disponível para a transmissão dos dados.

Conforme descrito por Tanenbaum (2003), o DiffServ define um conjunto de classes de serviço com regras de encaminhamento correspondentes. É uma estratégia que pode ser implementada em grande parte localmente a um roteador, sem configuração antecipada e sem ter de envolver todo o caminho. São utilizados bits do cabeçalho IP para indicar diferentes tipos de tráfego e prioridades.

O MPLS trata a questão de QoS com a criação de caminhos explícitos através da rede. É possível criar rotas explícitas para os fluxos de dados que são classificados de acordo com a disponibilidade de recursos e qualidade de serviço solicitada.

Todos os pacotes que fazem parte de um mesmo fluxo recebem o mesmo *label*, em cada *hop,* este é encaminhado para uma interface de saída com base no valor do *label*. O caminho percorrido pelo pacote é conhecido como *label-switched path* (LSP). O LSP deve ser capaz de garantir um determinado nível de QoS com base na infra-estrutura de rede utilizada.

2.6 Engenharia de tráfego

O termo Engenharia de tráfego refere-se à capacidade de orientar o tráfego através de uma rede. É uma técnica utilizada para otimizar o uso dos recursos de rede fazendo uma utilização de forma balanceada. Segundo, os principais objetivos da engenharia de tráfego que podem ser destacados são: o uso eficiente dos recursos de rede, com consequente economia de recursos financeiros; redução nos congestionamentos; satisfação dos requisitos das aplicações e dos usuários e a melhoria geral de desempenho da rede.

O roteamento tradicional utilizando IP é baseado no encaminhamento pelo caminho de menor custo. Além disso, os pacotes IP são encaminhados por cada roteador com base apenas no endereço IP de destino e sem levar em conta a forma como estes pacotes foram encaminhados nos roteadores anteriores e como serão encaminhados nos próximos roteadores. Além disso, o paradigma de encaminhamento IP não leva em conta a largura de banda disponível no *link*. O resultado deste comportamento, no envio de pacotes IP, é que alguns *links* da rede podem ficar com sobrecarga de tráfego, enquanto outros *links* ficam subutilizados (OSBORNE, 2002).

Os padrões de tráfego entre sites podem variar constantemente, desta forma a engenharia de tráfego pode trazer uma solução para o gerenciamento do tráfego, evitando *links* sobrecarregados. A figura 10 mostra o exemplo de uma rede utilizando o roteamento IP.

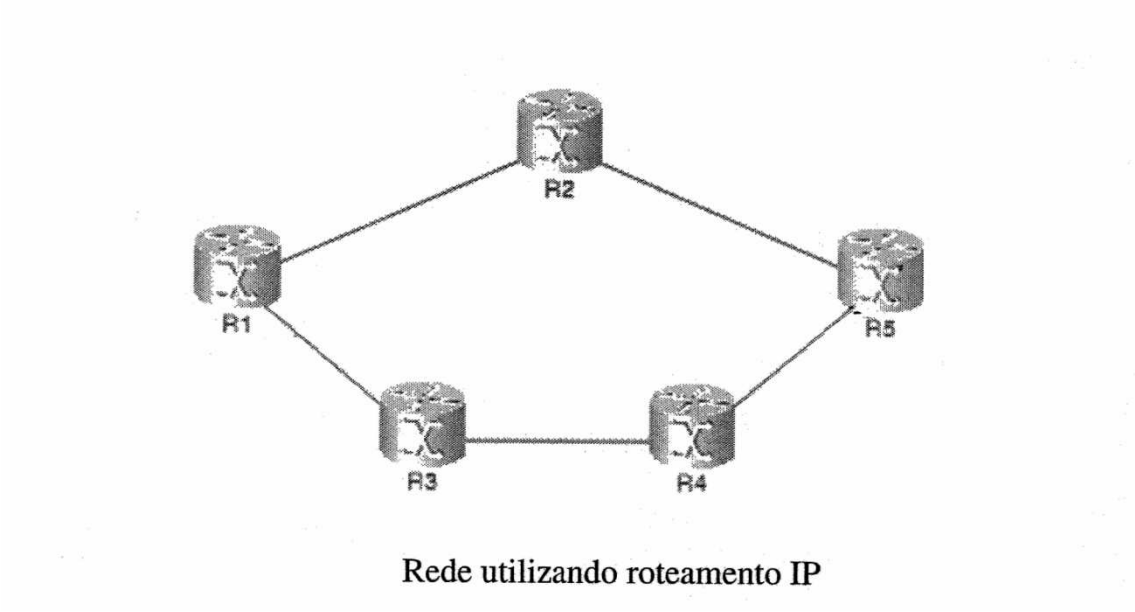


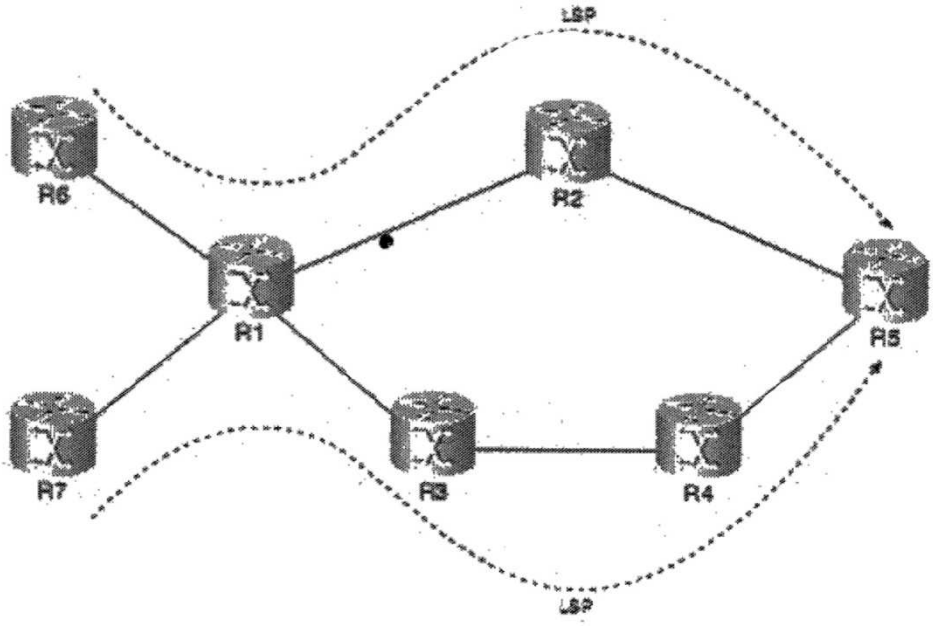
Figura 10. Rede utilizando roteamento IP
Fonte: (OSBORNE 2002)

Se na rede apresentada na Figura 10 todos os enlaces tiveram a mesma taxa de transmissão, o menor custo para o roteador R1 comunicar com o roteador R5 é: R1-R2-R5. Claramente, todo o tráfego de R1 para R5 vai utilizar o caminho R1-R2-R5, e o caminho R1-R3-R4-R5 não terá tráfego. Em uma rede real o funcionamento pode ser um pouco diferente. Muitos fluxos de tráfego de dados podem existir, e as cargas sobre os *links* podem variar muito.

2.6.1 Engenharia de Tráfego em MPLS

Em redes MPLS a engenharia de tráfego pode ser feita a partir do LSR de entrada da rede, elepode calcular a rota mais eficiente através da rede, em direção ao LSR de saída. O LSR pode fazer isto se tiver conhecimento da topologia da rede. Além disso, ele precisa saber a largura de banda disponível de todos os *links* da rede.

A Figura 11 mostra um exemplo de rede, se esta rede estiver utilizando MPLS é possível configurar dois caminhos (LSP) diferentes, assim como os *labels* que serão utilizados para identificar cada caminho. No roteador R1 é feita uma verificação do *label* para identificar a qual LSP o pacote pertence, em seguida, o roteador encaminha opacote por um dos dois LSP.



LSP em rede MPLS

Figura 11 Exemplo de LSP em rede MPLS
Fonte: (Redes MPLS, 2012).

É possível implementar engenharia de tráfego em qualquer rede que possua LSRs. Entretanto, devido à largura de banda e outros atributos sobre os *links* que devem ser conhecidos pelas LSRs, o protocolo de roteamento entre os LSRs deve ser um protocolo de roteamento por estado de enlace. Com um protocolo de roteamento por estado de enlace, cada roteador constrói um estado de seus próprios *links*, transmitindo esta informação para todos os outros roteadores na mesma area (GHEIN, 2007).

2.6.2 Componentes de Engenharia de Tráfego

A aplicação de engenharia de tráfego em redes MPLS envolve basicamente quatro componentes funcionais (ALVAREZ, 2006):

1. Distribuição de Informação – a Engenharia de Tráfego requer um conhecimento detalhado da topologia da rede, assim como conhecimento dinâmico sobre a capacidade da rede. Isso pode ser implementado por meio de

protocolos IGP com extensões específicas, de forma que atributos específicos de *links* (como largura de banda maxima, utilização de banda e banda reservada) sejam incluídos nos anúncios *"link state"* destes protocolos. Em uma rede MPLS,cada LSR mantém uma base de dados chamada TED (TE Database), utilizada para calcular caminhos específicos pela rede MPLS.

2. Componente de seleção de caminho – Baseado na topologia de rede e nos atributos de *link* presentes na TED, cada LSR calcula caminhos específicos para seus LSP. Estes caminhos podem ser "*strict*" ou "*loose*". Uma rota "*Strict*" é aquela em que o LSR de ingresso específica todos os LSR para o LSP. A rota "*loose*", por sua vez, tem apenas alguns LSR definidos no LSR de ingresso.

3. Componente de Sinalização e definição da rota – A rota calculada pelo componente anterior não é dita "funcional" até que um LSP seja, de fato, estabelecido pelo componente de sinalização. Isso porque o componente de "*pathSelection*" utilize as informações presentes na TED, que podem estar desatualizadas. O component de sinalização, portanto, é responsável pela checagem de todas as informações necessárias durante o processo de definição de rota.

4. Componente de encaminhamento de pacotes – Uma vez que o caminho seja estabelecido, o processo de encaminhamento é iniciado no LSR, baseado no conceito de comutação de *labels*.

Os principais protocolos de sinalização utilizados em conjunto com o MPLS são o "*Resource Reservation Protocolwith Traffic engineering Extensions*" (RSVP-TE) e o *Constraint-based Router Label Distribution Protocol* (CR-LDP).

2.6.3 SIMULADOR DE REDES GNS3

O GNS3 é um simulador de rede gráfico que funciona com imagens IOS (*Internetworking Operating System)* da Cisco, com sua interface gráfica e intuitiva fica fácil sua utilização, tornando assim uma ferramenta poderosa capaz de emular redes complexas.
(http://www.dltec.com.br/blog/cisco/simulador-de-redes-gns3/)

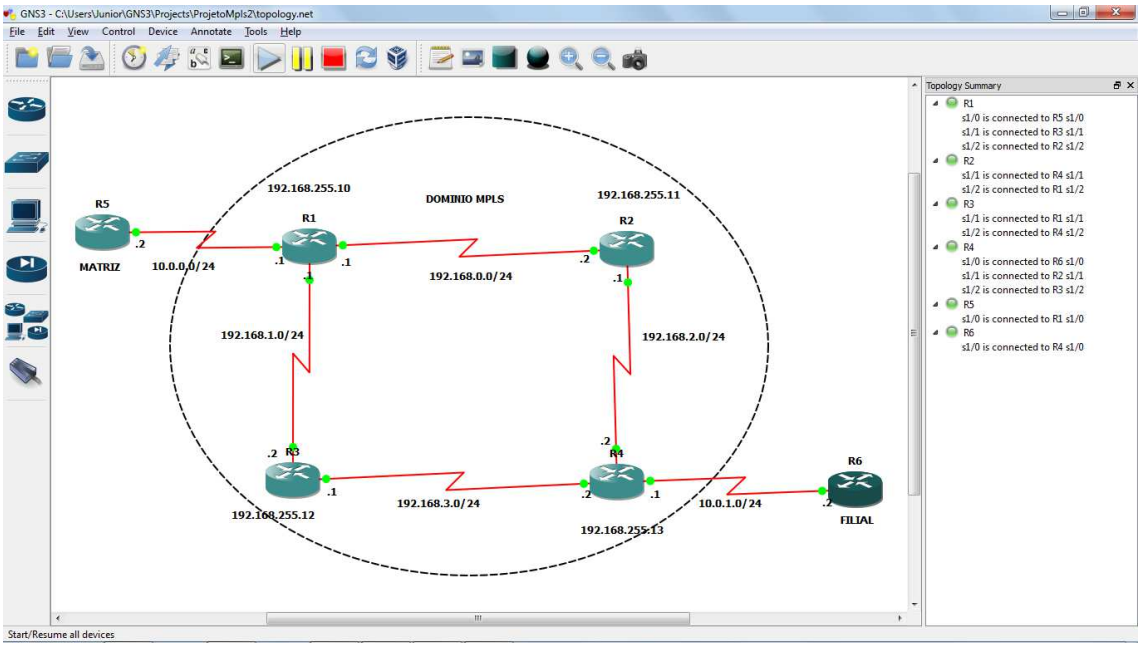


Figura 12: GNS3

Fonte: Autoria Própria

Para a simulação da tecnologia, utilizaremos o software GNS3, com roteadores CISCO da série 3700, com IOS (*Internetworking Operating System*).

3 IMPLEMENTAÇÃO DE QOS / MPLS

Neste capítulo serão apresentadas alguns recursos disponíveis em equipamentos Cisco para permitir a implementação de QoS sobre MPLS. Também serão apresentados alguns detalhes de configuração através da utilização do Cisco IOS (*Internetwork Operating System*) e do MQC (*Modular Quality of Service Command Line Interface).*

3.1 Funções de QoS

Um bom exemplo em que QoS é necessário é o tráfego VoIP. Esse tipo de tráfego apresenta restrições quanto ao limite máximo de tempo de entrega no seu destino, ou ele torna-se obsolete. Por este motive é necessário que as políticas de QoS priorizem este tráfego, de maneira que seja entregue dentro dos limites de tempo. Para conseguir isto, pode-se utilizar o Cisco IOS para agrupar o tráfego VoIP em uma fila com prioridade maior que aquela dos tráfegos FTP ou HTTP, garantindo que, em caso de congestionamento, os pacotes FTP e HTTP serão descartados antes dos pacotes VoIP. O Cisco IOS possui vários mecanismos para que os roteadores operem dessa forma. A Tabela 4.1 mostra algumas funções de QoS presentes no Cisco IOS e as correspondentes características (GHEIN, 2007).

Tabela 4.1 Funções de QoS e características correspondentes no Cisco IOS

Funções de QoS	**Recursos do Cisco IOS**
Classificação de Tráfego	Busca em lista de controle de acesso
Marcação de Tráfego	*DiffServ Code Point* (DSCP) Campo EXP – MPLS
Gerência de congestionamento	*Low-Latency Queuing* (LLQ) *Class-Based Weighted Fair Queuing* (CBWFQ)
Prevenção de Congestionamento	*Weighted Randon Early Detection* (WRED)

Condicionamento de Tráfego	Conformidade e Policiamento

3.2 Cisco IOS

Quando um LSR encaminha um pacote que já possui um *label*, ele precisa apenas procurar este *label* na sua tabela de encaminhamento de *labels* (LFIB) para decidir por onde encaminhar o pacote. O mesmo é válido para o tratamento de QoS: o LSR précisa apenas verificar os bits EXP do *label* para determinar como tratar o pacote. A melhor forma de efetuar a configuração de QoS sobre MPLS no Cisco IOS é através do MQC. O modelo MQC segue um padrão específico para configurações via linha de comando. Ele é o mais usado nos equipamentos Cisco, por ser padronizado e apresentar apenas três etapas, de acordo com Alvarez são elas:

- Definir as classes de tráfego utilizando regras de correspondência;
- Definir políticas de QoS para serem aplicadas as classes;
- Apontar dentro da interface a política como saída ou entrada.

3.3 Funcionamento do QoS sobre MPLS no Cisco IOS

O comportamento padrão do Cisco IOS quando são inseridos um ou mais *labels* no pacote IP é o de copiar o valor dos bits de precedência para os bits EXP de todos os *labels* inseridos isto é chamado de “reflexão TOS” porque nada muda em relação ao QoS. Entretanto, se os seis bits do campo DSCP são usados, somente os três primeiros bits serão copiados para os bits EXP do cabeçalho. Isto leva a primeira regra de QoS sobre MPLS (GHEIN, 2007).

- Regra1: por padrão, no Cisco IOS, os bits de precedência ou os três primeiros bits do campo DSCP no cabeçalho IP são copiados para os bits EXP de todos os *labels* inseridos no LSR de entrada.

O encaminhamento de um pacote com *label* é um pouco mais complicado, pois devem ser considerados dois casos: de um lado, a troca de

label com a possibilidade de adicionar um ou mais *labels* ao pacote, de outro lado, a troca de label com a possibilidade de remover um ou mais *labels* do pacote. No caso da troca de um *label* de entrada por um *label* de saída no LSR, os bits EXP são copiados do *label* de entrada para o *label* de saída. O mesmo é verdadeiro quando um *label* é trocado e são adicionados um ou mais *labels*. O valor dos bits EXP é copiado do *label* de entrada para o *label* de saída e também para os *labels* que são empilhados no pacote encaminhado.

Entretanto, o encaminhamento de pacotes com a retirada do *label* é um pouco diferente. Quando um roteador retira o *label* do topo da pilha de um pacote que encaminha, o valor dos bits EXP não é copiado para o novo *label* do topo ou para os bits de precedência do cabeçalho do pacote IP sem *label*. Isto significa que, por padrão, no Cisco IOS, os bits EXP do novo *label* do topo ou o campo DSCP do cabeçalho IP permanecem inalterados, ditando o novo QoS do pacote. Isto leva a segunda, Terceira e quarta regra de QoS sobre MPLS (GHEIN, 2007). Este é o comportamento padrão do Cisco IOS. Este comportamento pode ser alterado através do Cisco IOS para manter o valor de QoS quando os *labels* são retirados.

- Regra2: por padrão, no Cisco IOS, os bits EXP do *label* de entrada são copiados para o *label* de saída e para qualquer outro *label* empilhado no pacote;

- Regra3: por padrão, no Cisco IOS, os bits EXP do *label* do topo da pilha não são copiados para o *label* de saída quando o *label* do pacote de entrada é removido;

- Regra4: por padrao, no Cisco IOS, os bits EXP do *label* de entrada não são copiados para os bits de precedência ou os bits DSCP quando a pilha de *labels* é removida e o cabeçalho IP é exposto.

Além disso, quando o MCQ é utilizado para trocar o QoS de um pacote rotulado, somente o *label* do topo e os possíveis novos *labels* inseridos recebem o novo valor para os bits EXP. Isto significa que, quando o QoS de um

pacote rotulado é alterado manualmente em algum LSR, este valor de QoS sera novamente alterado na rede algum tempo depois. Ou seja, quando um *label* é retirado do topo da pilha, o valor do campo EXP não é copiado para o novo *label* exposto, conforme descrito na regra 3. Isto significa que o antigo valor de QoS do pacote esta novamente ativo. Isto leva a quinta regra.

- Regra5: Quando o valor do campo EXP é alterado por meio de configuração, os *labels* que estão abaixo do topo da pilha não recebem o novo valor do campo EXP.

As regras 4 e 5 levam ao fato que o tunelamento de QoS é possível. Isto significa que o valor de QoS do pacote IP pode ser transportado através de uma rede MPLS sem sofrer alteração.

3.4 DiffservTunneling Modes

Tunneling é a capacidade oferecida por uma rede MPLS de transportar o valor DiffServ de um pacote IP de uma forma transparente, de uma borda a outra da rede MPLS. O tunel tem inicio quando o *label* é adicionado ao pacote e termina quando o *label* é removido.

A regra 4 dá origem ao seguinte comportamento: indiferentemente do valor dos bits EXP introduzidos pelo LSR de entrada ou em qualquer outro LSR, este não é copiado para o pacote IP, no LSR de saída da rede MPLS. Por padrão, os bits de precedência ou DSCP do pacote IP são preservados.
O MPLS fornece QoS para pacotes MPLS usando os seguintes modos de tunel (LEWIS; PICKAVANCE, 2006):

- *Uniform Mode*– neste modo, as mudanças feitas no valor do campo EXP do label do topo da pilha são propagandas tanto para os *labels* inseridos na pilha como para os *labels* de baixo, quando os *labels* da pilha são removidos. A premissa, é que a rede está em um domínio DiffServ. Logo, qualquer mudança feita no campo EXP do pacote MPLS em trânsito sera aplicada para todos os *labels* do pacote, bem como para o pacote IP.

- *Short Pipe Mode*– este modo é útil para aplicação de políticas de QoS nos provedores, independentemente da política de QoS do cliente. Os bits de precedência do pacote IP são propagados para cima na pilha de *labels*. Quando o *label* é trocado, o valor do campo EXP é mantido. Se o valor do campo EXP do *label* do topo da pilha é alterado, esta mudança é propaganda para todos os *labels* da pilha, mas não para o pacote IP.
- *Pipe Mode*– neste modo duas marcações são importantes para um pacote quando ele percorre a rede MPLS. Primeiro, a marcação usada pelos LSR intermediaries ao longo do LSP, incluindo o LSR de saída. Segundo, a marcação original do pacote antes da entrada na rede MPLS, que continuará sendo usada quando o pacote sair da rede MPLS. No LSR de saída todos os *labels* são removidos, mas, a fim de preservar a marcação transportada no *label*, o LSR de borda copia este valor antes de remover os *labels*. Esta cópia interna é utilizada para classificar os pacotes na interface de saída.

No Cisco IOS a configuração feita para ativar um dos três modos DiffServ é feita através do MQC. O MQC é configurável por interface. Portanto, é possível escolher o modo por interface, consequentemente, por cliente conectado a rede MPLS.

3.5 Como o QoS funciona para tráfego MPLS

Em roteadores Cisco da série 10000, a classificação dos pacotes MPLS não leva em conta o cabeçalho IP, ou seja, não é possível classificar pacotes MPLS em classes distintas utilizando-se o cabeçalho IP que se encontra encapsulado no pacote MPLS. O roteador classifica os pacotes MPLS como pertencentes a uma mesma classe padrão, exceto se for especificado um *qos-group* ou se existirem na interface de entrada regras para classes de tráfego.

Após a imposição do *label* MPLS, por padrão, o roteador copia o valor do campo EXP para todos os *labels* adicionados ao pacote. O valor do campo EXP pode ser modificado através de diretivas de comando “*set*” ou “*police*”.

3.6 Configurando QoS sobre MPLS em um LSR de entrada

Um LSR pode ser um roteador de borda de um provedor ou um roteador intermediário da rede MPLS, então ele pode ser a ligação de uma rede não-MPLS a uma rede MPLS, tanto de entrada como de saída. A definição do valor dos bits EXP é somente válida para pacotes que chegam pela interface de entrada (não MPLS) do LSR e saem por uma interface MPLS. Portanto, somente políticas de entrada podem definir os bits EXP de um pacote quando ele sai por uma interface MPLS. Se o pacote chega por uma interface MPLS,a definição dos bits EXP não tem efeito.

Para configurar a política de QoS MPLS no LSR de entrada da rede MPLS devem ser executados os seguintes passos:

- Classificar pacotes IP utilizando uma "*Class Map";*
- Definir o campo EXP usando um "*Policy Map";*
- Atribuir uma política de serviço de QoS para uma interface.

3.6.1 Classificação de pacotes IP utilizando uma "*Class Map"*

Uma *Class Map* define uma classe de tráfego através da utilização de regras de correspondência. Os pacotes IP são classificados em *Class Maps* de acordo com seus bits de precedência. Esta classificação é feita no LSR de entrada da rede MPLS e configurada utilizando o MQC.

Para classificar pacotes IP utilizando uma *Class Map*s seguintes comandos devem ser utilizados no LSR de entrada da rede MPLS.

- **class-map** *class-map-name* - cria ou modifica uma *Class Map;*
- **match mpls experimental top most** valor – especifica o valor do campo EXP utilizado para classificar o tráfego;
- **match** *critério* – define o critério utilizado pelo roteador para associar os pacotes as classes de tráfego.

O exemplo abaixo mostra a criação de uma *Class Map* com o nome de exp4, com o campo MPLS EXP 4 definido como critério de classificação.

```
Router(config)#class-map match-all exp4
Router(config-cmap)#match mpls experimental topmost 4
Router(config-cmap)#end
```

3.6.2 Implementação MPLS

Cenário 1

Simulação de implementação MPLS, utilizando o simulador GNS3

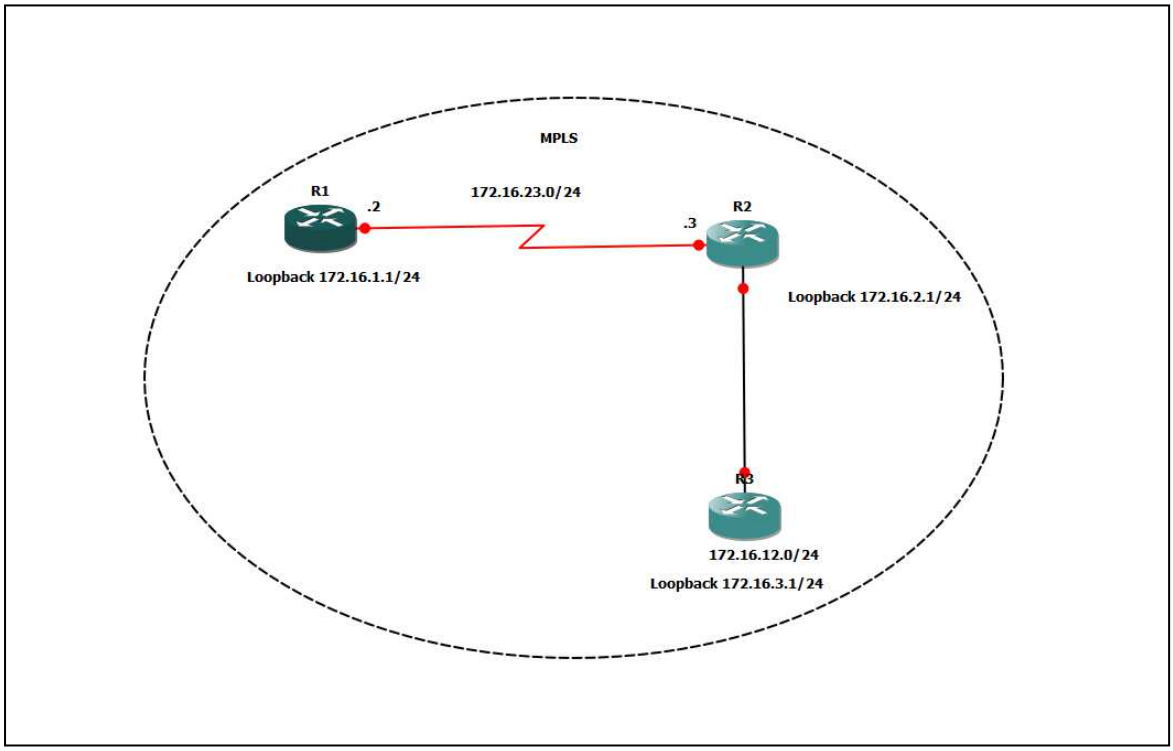
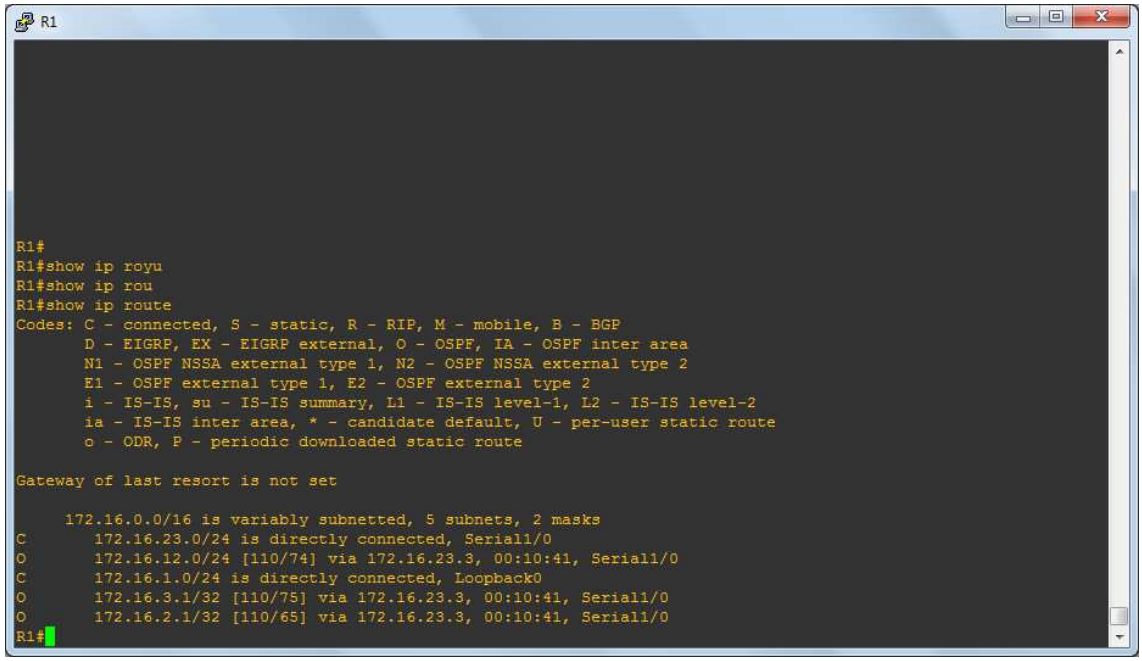


Figura 13 Topologia da rede

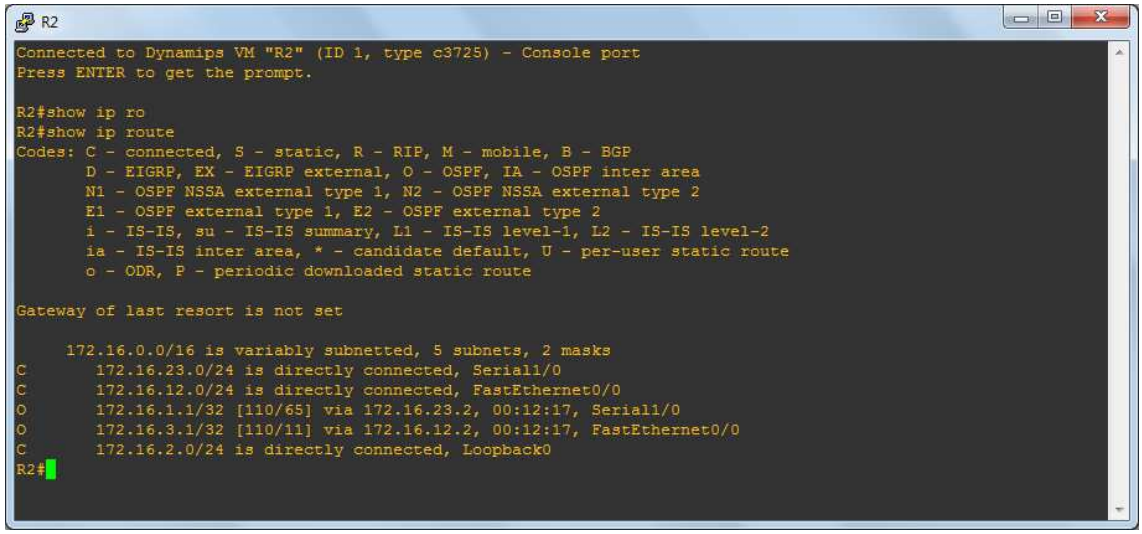
Fonte: Autoria própria

Mostrando as rotas dos roteadores

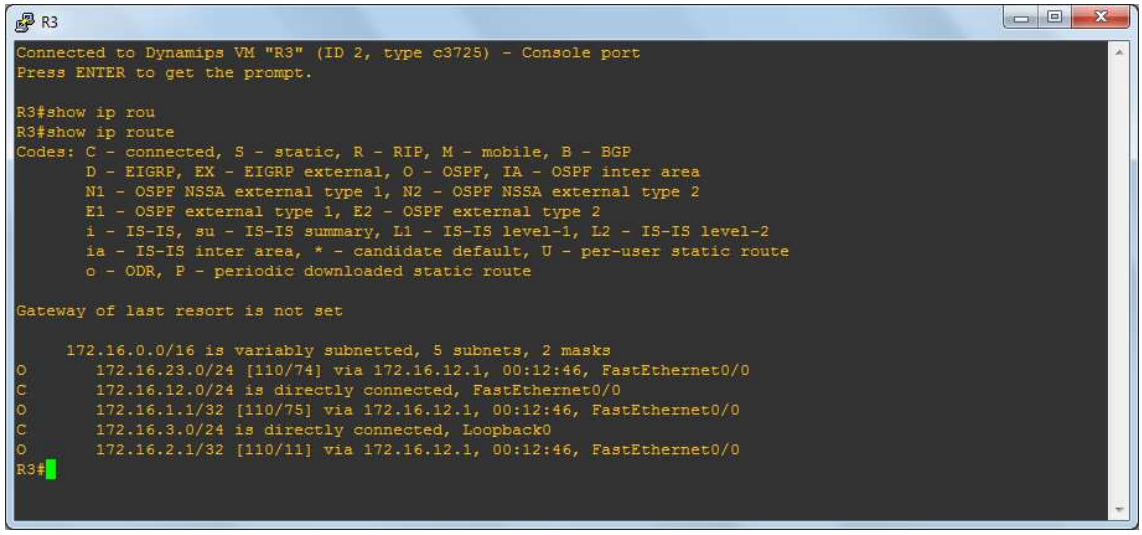
Router 1



Router 2



Router 3



Configuração router 1

version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R1
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!

```
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface Loopback0
 ip address 172.16.1.1 255.255.255.0
!
interface FastEthernet0/0
 no ip address
```

```
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 172.16.23.2 255.255.255.0
 mpls ip
 serial restart-delay 0
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
router ospf 1
 log-adjacency-changes
 network 172.16.0.0 0.0.255.255 area 0
!
```

```
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
!
!
!
control-plane
!
!
!
!
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 login
!
!
end
```

Configuração router 2

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R2
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
!
!
!
!
!
!
!
```

```
!
!
!
!
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface Loopback0
 ip address 172.16.2.1 255.255.255.0
!
interface FastEthernet0/0
 ip address 172.16.12.1 255.255.255.0
 duplex auto
 speed auto
 mpls ip
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 172.16.23.3 255.255.255.0
 mpls ip
 serial restart-delay 0
```

```
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
router ospf 1
 log-adjacency-changes
 network 172.16.0.0 0.0.255.255 area 0
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
!
!
!
control-plane
!
!
!
```

```
!
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 login
!
!
end
```

Configuração router 3

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R3
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
```

```
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface Loopback0
```

```
 ip address 172.16.3.1 255.255.255.0
!
interface FastEthernet0/0
 ip address 172.16.12.2 255.255.255.0
 duplex auto
 speed auto
 mpls ip
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
```

```
router ospf 1
 log-adjacency-changes
 network 172.16.0.0 0.0.255.255 area 0
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
!
!
!
control-plane
!
!
!
!
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
```

login

!

end

Cenário 2

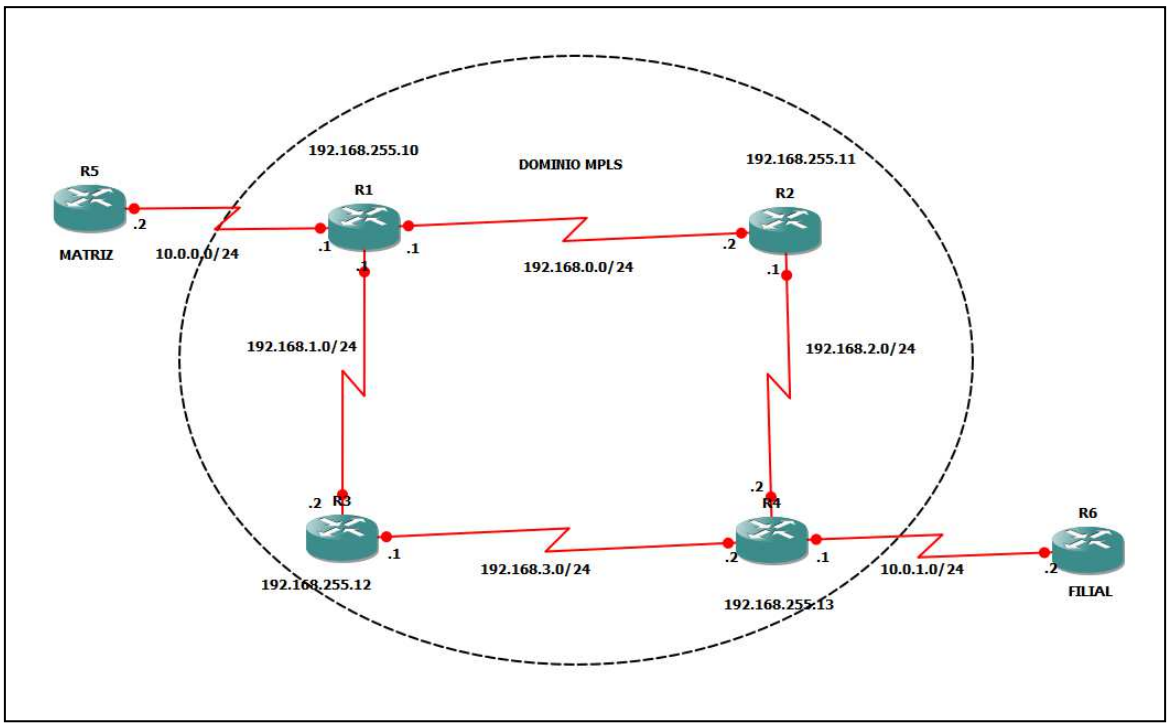
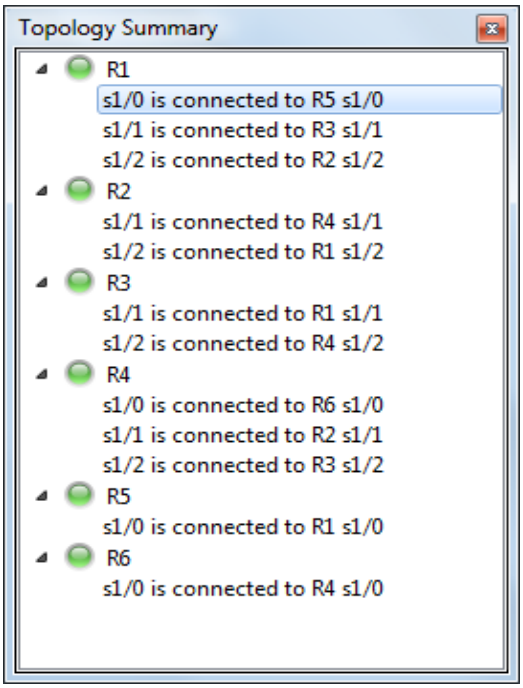


Figura 14 Topologia de configuração MPLS

Fonte: Autoria própria

Sumário da topologia da figura acima



Configuração Router 1

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R1
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
mpls traffic-eng tunnels
!
!
!
!
!
!
```

```
!
!
!
!
!
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface Loopback0
 ip address 192.168.255.10 255.255.255.255
!
interface Tunnel2
 ip unnumbered Loopback0
 tunnel destination 192.168.255.13
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 2 2
 tunnel mpls traffic-eng bandwidth  158
 tunnel mpls traffic-eng path-option 1 explicit name BOTTOM
 no routing dynamic
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
```

```
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 10.0.0.1 255.255.255.0
 ip ospf 1 area 0
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 ip address 192.168.1.1 255.255.255.0
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 ip ospf 1 area 0
 shutdown
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 ip rsvp bandwidth 750000
 ip rsvp resource-provider none
!
interface Serial1/2
 ip address 192.168.0.1 255.255.255.0
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 ip ospf 1 area 0
 shutdown
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
```

```
 ip rsvp bandwidth 750000
 ip rsvp resource-provider none
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
router ospf 1
 mpls ldp autoconfig area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 timers throttle spf 1000 1000 1000
 network 192.168.0.0 0.0.0.255 area 0
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.255.10 0.0.0.0 area 0
!
ip forward-protocol nd
ip route 10.0.255.0 255.255.255.0 10.0.0.2
!
!
no ip http server
no ip http secure-server
!
ip explicit-path name BOTTOM enable
 next-address 192.168.1.2
 next-address 192.168.3.2
!
ip explicit-path name TOP enable
 next-address 192.168.0.2
 next-address 192.168.2.2
!
!
```

```
!
!
!
control-plane
!
!
!
!
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 login
!
!
end
```

Configuração Router 2

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
```

```
hostname R2
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
mpls traffic-eng tunnels
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
```

```
!
ip tcp synwait-time 5
!
!
!
!
!
interface Loopback0
 ip address 192.168.255.11 255.255.255.255
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 ip address 192.168.2.1 255.255.255.0
```

```
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 ip ospf 1 area 0
 shutdown
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 ip rsvp bandwidth 750000
 ip rsvp resource-provider none
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
router ospf 1
 mpls ldp autoconfig area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 timers throttle spf 1000 1000 1000
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
 network 192.168.255.11 0.0.0.0 area 0
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
!
!
```

```
!
control-plane
!
!
!
!
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 login
!
!
end
```

Configuração Router 3

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R3
!
```

```
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
mpls traffic-eng tunnels
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
ip tcp synwait-time 5
```

```
!
!
!
!
!
interface Loopback0
 ip address 192.168.255.12 255.255.255.255
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 ip address 192.168.1.2 255.255.255.0
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 ip ospf 1 area 0
 shutdown
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
```

```
 ip rsvp bandwidth 750000
 ip rsvp resource-provider none
!
interface Serial1/2
 ip address 192.168.3.1 255.255.255.0
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 ip ospf 1 area 0
 shutdown
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 ip rsvp bandwidth 750000
 ip rsvp resource-provider none
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
router ospf 1
 mpls ldp autoconfig area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 timers throttle spf 1000 1000 1000
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
 network 192.168.255.12 0.0.0.0 area 0
!
ip forward-protocol nd
!
!
no ip http server
```

```
no ip http secure-server
!
!
!
!
!
control-plane
!
!
!
!
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 login
!
!
end
```

Configuração Router 4

```
version 12.4
service timestamps debug datetime msec
```

```
service timestamps log datetime msec
no service password-encryption
!
hostname R4
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
mpls traffic-eng tunnels
!
!
!
!
!
!
!
!
!
!
!
!
!
```

```
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface Loopback0
 ip address 192.168.255.13 255.255.255.255
!
interface Tunnel2
 ip unnumbered Loopback0
 tunnel destination 192.168.255.10
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 2 2
 tunnel mpls traffic-eng bandwidth  158
 tunnel mpls traffic-eng path-option 1 explicit name BOTTOM
 no routing dynamic
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
```

```
!
interface Serial1/0
 ip address 10.0.1.1 255.255.255.0
 ip ospf 1 area 0
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 ip address 192.168.2.2 255.255.255.0
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 ip ospf 1 area 0
 shutdown
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 ip rsvp bandwidth 750000
 ip rsvp resource-provider none
!
interface Serial1/2
 ip address 192.168.3.2 255.255.255.0
 ip ospf hello-interval 1
 ip ospf dead-interval 3
 ip ospf 1 area 0
 shutdown
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 ip rsvp bandwidth 750000
 ip rsvp resource-provider none
!
interface Serial1/3
 no ip address
 shutdown
```

```
 serial restart-delay 0
!
router ospf 1
 mpls ldp autoconfig area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 timers throttle spf 1000 1000 1000
 network 192.168.2.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
 network 192.168.255.13 0.0.0.0 area 0
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
ip explicit-path name TOP enable
 next-address 192.168.2.1
 next-address 192.168.0.1
!
ip explicit-path name BOTTOM enable
 next-address 192.168.3.1
 next-address 192.168.1.1
!
!
!
!
!
control-plane
!
!
!
```

```
!
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 login
!
!
end
```

Configuração Router 5

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R5
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
```

```
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface FastEthernet0/0
```

```
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 10.0.0.2 255.255.255.0
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 10.0.0.1
!
```

```
!
ip http server
no ip http secure-server
!
!
!
!
!
control-plane
!
!
!
!
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 login
!
!
end
```

Configuração Router 6

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R6
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 5
no ip icmp rate-limit unreachable
ip cef
!
!
!
!
no ip domain lookup
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
!
!
!
!
!
!
!
!
!
!
```

```
!
!
!
!
!
!
ip tcp synwait-time 5
!
!
!
!
!
interface Loopback0
 ip address 10.0.254.1 255.255.255.0
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 10.0.1.2 255.255.255.0
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 no ip address
```

```
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 10.0.1.1
!
!
ip http server
no ip http secure-server
!
!
!
!
!
control-plane
!
!
!
!
!
!
!
!
!
```

```
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 login
!
!
end
```

4 CONSIDERAÇÕES FINAIS

Umas das grandes vantagens que a tecnologia MPLS possibilita é que a implementação pode ser feita sobre tecnologia já existente, como o Frame-Relay e ATM. Assim, pode-se implementar este novo conceito sem trocar toda a parte de *hardware* (equipamentos) da rede. Como o MPLS faz o encaminhamento por meio dos rótulos, o processamento no núcleo não é elevado, permitindo com isso, que se tenha equipamentos mais simples no núcleo, concentrando os equipamentos mais robustos nas bordas. As configurações realizadas nas implementações de laboratório permitiram verificar que a implementação não e complexa. Analisando as implementações atuais das operadoras de telecom, foi visto que o serviço mais utilizado é o de VPN (Virtual Private Network). Em relação à QoS (Qualidade de Serviiço), a tecnologia MPLS pode implementar e dar suporte a fluxos diferentes de dados, voz, sendo uma vantagem em relação à outras tecnologias de rede.

REFERÊNCIAS

ALVAREZ, S. **Qos for IP/MPLS Networks.**Indianápolis: Cisco Press, 2006. 336p.

FARREL, A.; BRYSKIN, I. **GMPLS:** Architecture and Applications. San Francisco: Elsevier, 2006. 412p.

GARCIA, A.L.; WIDJAJA, I. **Communication Networks:** Fundamental Concepts and Key Architectures. Columbus: McGraw-Hill Professional, 2004. 900p.

GHEIN, L.D. **MPLS Fundamentals.**Indianápolis: Cisco Press, 2007.651p.

MCDYSAN, D.E.; PAW, D. **ATM & MPLS Theory & Application:** Foundations of Multi-Service Networking. Columbus: McGraw Hill/Osborne, 2002. 962p.

OSBORNE, E. **Engenharia de tráfego com MPLS.** Rio de Janeiro: Campus, 2002. 640p.

PEPELNJAK, I.; GUICHARD, J. **MPLS and VPN Architectures.**Indianápolis: Cisco Press, 2007. 336p.

TANENBAUM, A.S. **Redes de Computadores.** 4.ed. Rio de Janeiro: Campus, 2003.

LEWIS, C.; PICKAVANCE, S. **Selecting MPLS VPN Services.**Indianápolis: Cisco Press, 2006. 465p.

OLIVEIRA,J.M.;LINS,R.D.;MENDONÇA,R. **REDES MPLS Fundamentos e Aplicações** Rio de Janeiro: Brasport, 2012. 223p.