UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE TELECOMUNICAÇÕES

ENGENHARIA DE TRÁFEGO UTILIZANDO O PROTOCOLO MPLS

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2016

HENRIQUE MACIEL SIQUEIRA

ENGENHARIA DE TRÁFEGO UTILIZANDO O PROTOCOLO MPLS

Trabalho de Conclusão de Curso de Graduação, apresentado ao Curso Superior de Tecnologia em Sistemas de Telecomunicações, do Departamento Acadêmico de Eletrônica, da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Tecnólogo. Orientador: Prof. Kleber Kendy Horikawa Nabas, D.Sc

CURITIBA
2016

TERMO DE APROVAÇÃO

HENRIQUE MACIEL SIQUEIRA

ENGENHARIA DE TRÁFEGO UTILIZANDO O PROTOCOLO MPLS

Este trabalho de conclusão de curso foi apresentado no dia 23 de Maio de 2016, como requisito parcial para obtenção do título de Tecnólogo em Sistemas de Telecomunicações, outorgado pela Universidade Tecnológica Federal do Paraná. O aluno foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

_____
Prof. Dr. Kleber Kendy Horikawa Nabas
Coordenador de Curso
Departamento Acadêmico de Eletrônica

_____
Prof. M.Sc. Sérgio Moribe
Responsável pela Atividade de Trabalho de Conclusão de Curso
Departamento Acadêmico de Eletrônica

BANCA EXAMINADORA

_____
Prof. M.Sc Danillo Leal Belmonte
UTFPR

_____
Prof. Dr. Edenilson José da Silva
UTFPR

_____
Dr. Kleber Kendy Horikawa Nabas
Orientador - UTFPR

"A Folha de Aprovação assinada encontra-se na Coordenação do Curso"

RESUMO

SIQUEIRA, Henrique Maciel. **Engenharia de tráfego utilizando o protocolo MPLS**. 2016. 61 f. Trabalho de Conclusão de Curso (Curso Superior de Tecnologia em Sistemas de Telecomunicações), Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2016.

Com o avanço tecnológico e o crescimento da demanda por velocidades cada vez maiores, os provedores de internet dependem de alta performance em seu *backbone* para garantir a entrega de seus serviços com qualidade. Este trabalho apresenta uma abordagem dos mecanismos utilizados para se aplicar a engenharia de tráfego em um *backbone*. Discute os conceitos teóricos dos protocolos de roteamento interno, *overlay models* e do protocolo MPLS. Apresenta as razões pelas quais a engenharia de tráfego utilizando o protocolo MPLS é uma alternativa interessante para o provedores de serviços para a internet.

Palavras chave: MPLS. Engenharia de Tráfego. Internet.

ABSTRACT

Siqueira, Henrique Siqueira. **Traffic Enginnering over MPLS.** 2016. f. Trabalho de Conclusão de Curso - Curso Superior de Tecnologia em Sistemas de Telecomunicações, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2016.

With advances in technology and the growing demand for increasingly higher speeds, internet service providers rely on high performance in its backbone to ensure delivery of their services with quality. It presents an approach to the mechanisms used to apply to traffic engineering in a backbone. It argues the theoretical concepts of internal routing protocols, overlay models and MPLS protocol. It presents the reasons the traffic engineering using MPLS protocol is an interesting alternative for service providers to the Internet.

Keywords: MPLS. Traffic Enginnering. Internet

LISTA DE ILUSTRAÇÔES

FIGURA 1: FORMATO DA CÉLULA ATM. 26
FIGURA 2: REDE ATM EM CAMADAS 27
FIGURA 3: CABEÇALHO MPLS. 31
FIGURA 4: EXEMPLO DE UMA LFIB. 32
FIGURA 5: OPERAÇÃO *LABEL PUSHING*. 33
FIGURA 6: OPERAÇÃO *LABEL SWAPPING*. 33
FIGURA 7: OPERAÇÃO *LABEL POPPING*. 34
FIGURA 8: OPERAÇÃO *LABEL PEN-ULTIMATE POPPING*. 34
FIGURA 9: MPLS *LABLE SWITCHED PATHS* 37
FIGURA 10: TOPOLOGIA DO LAB MPLS-TE 42
FIGURA 11: SAÍDA DO COMANDO *SHOW MPLS INTERFACES* 51
FIGURA 12: SAÍDA DO COMANDO *SHOW MPLS LDP NEIGHBOR* 52
FIGURA 13: SAÍDA DO COMANDO *SHOW IP CEF* 52
FIGURA 14: SAÍDA DO COMANDO *SHOW MPLS LDP BINDINGS* 53
FIGURA 15: SAÍDA DO COMANDO *SHOW MPLS FORWARDING-TABLE* 53
FIGURA 16: SAÍDA DO COMANDO *SHOW OSPF NEIGHBOR* 54
FIGURA 17: SAÍDA DO COMANDO *SHOW IP ROUTE* 54
FIGURA 18: SAÍDA DO COMANDO *SHOW MPLS TRAFFIC-ENG TUNNELS TUNNEL 100* 55
FIGURA 19: TESTE DE *PING* E *TRACEROUTE* 56
FIGURA 20: *SHUTDOWN* NA INTERFACE ENTRE R4 E R2 57
FIGURA 21: TESTE DE PING E *TRACEROUTE* APÓS SIMULAR A FALHA 58
FIGURA 22: REDUNDÂNCIA ASSUMIU APÓS A FALHA 58

LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

.AAL - ATM Adaption Layer
AS - Sistema Autônomo
ATM - ASYNCRONOUS TRANSFER MODE
BECN - Backward Explicit Congestion Notification
C/R - Command / Response
CLNS - Connectionless Network Service
CLP - Cell Loss Priority
CPU - Central Processing Unit
DCE - Equipamento de Terminação de Circuito
DE - Discard Eligibility Indicator
DLCI - Data Link Connection Identifier
DTE - Equipamento Terminal de Dados
EA - Extension Bit
ES - Sistemas Finais
FEC - Forwarding Equivalence Classes
FECN - Foward Explicit Congestion Notification
FIB - Forwarding Information Base
FR - Frame Relay
GFC - Generic Flow Control
IETF - Internet Enginnering Task Force
IGP - Internal Gateway Protocol
IP - Internet Protocol
IPv4 - Internet Protocol version 4
IPv6 - Internet Protocol version 6
IS - Sistemas Intermediários
ISHs - Intermediate System Hellos
IS-IS - Intermediate System to Intermediate System
ISP - Internet Service Provider
LAFP - Frame Mode Bearer Services
LDP - Label Distribution Protocol
LFIB - Label Forwarding Instance Base
LIB - Label Information Base
LMI - Local Management Interface
LSA - Link State Advertisement
LSP - Label Switching Path
LSP - Link State Packet
LSR - Label Switching Router
MPLS - Multiprotocol Label Switching
MPLS-TE - Multiprotocol Label Switching Traffic Enginnering

MTU - Maximun Transmission Unit
NNI - Network-to-Network Interface
OSI - Open System Interconnection
OSPF - Open Shortest Path First
PTI - Payload Type Identifier
PVC - Permanent Virtual Circuit
QoS - Quality of Service
RIB - Routing Information Base
RSVP - Resource Reservation Protocol
SAR - Segmentação e Rmontagem
SFP - Shortest Path First
SVC - Switched Virtual Circuit
TC - Traffic Class
TCP/IP - Transmission Control Protocol / Internet Protocol
TDM - Multiplexação por Divisão de Tempo
TTL - Timo to Live
UNI - User-Network Interface
VC - Circuitos Virtuais
VCC - Virtual Channel Connection
VCI - Virtual Channel Identifier
VPC - Virtual Path Connection
VPI - Virtual Path Identifier

SUMÁRIO

1 INTRODUÇÃO 9
1.1 PROBLEMA 10
1.2 JUSTIFICATIVA 11
1.3 OBJETIVOS 11
1.3.1 OBJETIVO GERAL 11
1.3.2 OBJETIVO ESPECÍFICO 12
1.4 PROCEDIMENTOS METODOLÓGICOS 12
2 FUNDAMENTAÇÃO TEÓRICA 13
2.1 *OPEN SHORTEST PATH FIRST (OSPF)* 14
2.1.1 DEFINIÇÕES DE TERMOS 14
2.1.2 O *BACKBONE* DO SISTEMA AUTÔNOMO 16
2.1.3 ROTEAMENTO INTRA-ÁREA 16
2.1.4 CLASSIFICAÇÃO DOS ROTEADORES 17
2.2 INTERMEDIATE SYSTEM TO INTERMEDIATE SYSTEM (IS-IS) 18
2.2.1 CLNS 19
2.2.2 OPERAÇÃO IS-IS 19
2.2.3 ÁREAS E DOMÍNIOS DE ROTEAMENTO 20
2.2.4 ROTEADORES DE NÍVEL 1 20
2.2.5 ROTEADORES DE NÍVEL 2 21
2.2.6 ROTEADORES DE NÍVEL 1 / NÍVEL 2 21
2.2.7 VANTAGENS EM SE USAR O PROTOCOLO IS-IS 21
2.3 *FRAME RELAY (*FR) 22
2.3.1 INTERFACES DE REDE 23
2.3.2 CIRCUITOS VIRTUAIS (VC) 24
2.3.2 ESTRUTURA DO *FRAME* 24
2.4 *ASYNCRONOUS TRANSFER MODE (ATM)* 25
2.4.1 FORMATO DA CÉLULA ATM 26
2.4.2 REDE ATM EM CAMADAS 27
2.4.3 IDENTIFICADORES DE CONEXÃO LÓGICA 28
2.4.4 TIPOS DE CONEXÕES 29
2.5 *MULTIPROTOCOL LABEL SWITCHING (MPLS)* 29
2.5.1 TERMINOLOGIA MPLS 30
2.5.2 CABEÇALHO MPLS 31
2.5.3 *LABEL* 32
2.5.4 ROTEADORES EM UM LSP 35
2.5.5 TIPOS DE LSP 36
3 MPLS *TRAFFIC ENGINEERING* – MPLS-TE 38
3.1 POR QUE USAR MPLS-TE? 39
3.2 COMO O MPLS-TE TRABALHA 40
4 LAB – CONFIGURAÇÃO MPLS-TE UTILIZANDO EXPLICIT PATH 42
4.1 CONFIGURAÇÃO DOS ROTEADORES 43
4.2 TESTES – MOSTRANDO OS RESULTADOS OBTIDOS 51
5 CONCLUSÃO 59
REFERÊNCIAS 60

1 INTRODUÇÃO

O presente trabalho de conclusão de curso irá discorrer e colocar em prática a engenharia de tráfego para a *internet* utilizando o protocolo *Multiprotocol Label Switching* (MPLS). A engenharia de tráfego para a internet pode ser definida como o aspecto da engenharia de rede que lida com questões de avaliação de desempenho e otimização de redes *Internet Protocol* (IP).

Uma das mais importantes funções desempenhadas pela *internet* é o roteamento de tráfego entre nós de ingresso para nós de egresso, assim como, alguns dos principais objetivos da engenharia de tráfego para a internet é facilitar a confiabilidade das operações de rede, controle e otimização das funções de roteamento, para conduzir o tráfego através da rede de forma mais eficaz.

Na parte inicial do trabalho serão apresentados conceitos sobre alguns modelos de engenharia de tráfego utilizados na *internet*, quais são as limitações dos mecanismos de controle dos principais *Internal Gateway Protocol* (IGPs), em seguida serão apresentados as vantagens de se utilizar o *Multiprotocol Label Switching Traffic Enginnering* (MPLS-TE).

Através documentação teórica, um ambiente virtual será emulado com roteadores do fabricante *Cisco Systems,* configurados com o MPLS-TE, para isto será utilizado o *software* GNS3. Neste cenário, as principais características da engenharia de trafego, utilizando MPLS, poderão ser apresentadas.

Portanto, pretende-se aqui, difundir essa técnica entre os estudantes do curso de Tecnologia em Sistemas de Telecomunicações e a comunidade acadêmica em geral, como isso é tratado no *backbone* de empresas que provem serviços, não apenas a *internet,* mas todo serviço que depende de confiabilidade, escalabilidade entre outras características que garantem a qualidade dos serviços.

1.1 PROBLEMA

Os atuais IGPs não possuem um adequado controle de capacidades que são ideais para a Engenharia de tráfego, isso acaba gerando muita dificuldade para a criação de políticas de endereçamento de rede capazes de minimizar problemas de performances em um *backbone*. Os IGPs são baseados em *shortest path first algorititms* (SPF), que contribui significantemente para o aumento de problemas de congestionamento dentro de um Sistema Autônomo (AS) na *internet*. (RFC 2702, 1999) [1]

A métrica destes protocolos é baseada em topologia, onde fatores como disponibilidade de banda e características de tráfego são detalhes que não são considerados em decisões de roteamento, com isso, congestionamentos frequentemente ocorrem. (RFC 2702, 1999) [1].

Em muitos ambientes onde a engenharia de tráfego utiliza IGPs, uma série de problemas começa a ocorrer conforme o crescimento da rede. Um exemplo seria, pela característica de funcionamento dos IGPs, um roteador poderia rotear um determinado tráfego através de um *link* ou uma interface que não possui banda suficiente para alocá-lo.

Ao longo dos anos surgiram os *overlay models* para tentar contornar essa inadequações dos IGPs, tais como, IP sobre ATM ou IP sobre *frame relay*. Ambos modelos, possuem a capacidade de criar topologias virtuais, onde as mesmas são provisionadas em cima de uma topologia física, de modo que, estas topologias lógicas apareçam como *links* físicos para os protocolos de roteamento IGP. (RFC 2702) [1] Devido a capacidade de controle de recursos que estes modelos possuem, tais como: caminhos explícitos através de circuitos virtuais, controle de banda, política para controle de tráfego, entre outros, por um bom tempo, eles se tornaram uma solução encontrada para suprir as limitações dos IGPs, porém com o aumento agressivo da demanda de banda dentro da *internet*, a utilização destes modelos, passou a ser muito caro para os provedores de serviço, devido ao fato, de que o *upgrade* dos equipamentos que suportam IP sobre ATM e IP sobre *frame relay*, são baseados em *hardware* e não em *software.*

1.2 JUSTIFICATIVA

Em redes com alta densidade, o uso do da engenharia de tráfego sobre MPLS (MPLS-TE) torna-se desejável. MPLS é estrategicamente significante para a engenharia de tráfego, pois ela pode potencializar e prover muitas funcionalidades, que também são abordadas pelos *overlay models* (IP sobre ATM e IP sobre *frame relay)*, de uma forma integrada e com um custo menor em relação a eles.

A engenharia de tráfego sobre MPLS, possui uma série de facilidades que podem ser tranquilamente configuradas, de modo que facilite o gerenciamento do tráfego, otimize a utilização de *links,* diminua o *overhead*, diminua a necessidade da alta utilização de *Central Processing Unit* (CPU) dos roteadores.

Além dessas características citadas acima, existe uma série de outros atributos que devem ser considerados para a implementação do MPLS-TE, porém não fazem parte do escopo deste trabalho, que está voltado para o público acadêmico.

1.3 OBJETIVOS

1.3.1 Objetivo geral

Implementar uma topologia com roteadores em ambiente virtualizado, utilizando GNS3, afim de comprovar a parte teórica que será transcrita neste trabalho, apresentando as principais características da utilização do MPLS-TE, tendo em vista, despertar o interesse do público acadêmico para este assunto.

1.3.2 Objetivo específico

- Criar uma topologia de uma malha MPLS, utilizando o GNS3;
- Caracterizar as vantagens da utilização do MPLS para a Engenharia de tráfego;
- Descrever características, conceitos e importâncias deste método;
- Implementar as configurações no software GNS3 que se apliquem a este objetivo;
- Realizar a análise dos resultados obtidos.

1.4 PROCEDIMENTOS METODOLÓGICOS

O projeto será desenvolvido em etapas, na primeira parte, será realizado pesquisa em manuais, guias e bibliografias de referências que tratam deste tema e será mostrado as motivações que levaram a escolha da abordagem deste assunto.

Na segunda etapa, o laboratório virtual será montado, de modo que seja possível aplicar as configurações necessárias para a implementação do MPLS-TE, baseado na documentação teórica.

Na terceira etapa, será realizado a simulação do ambiente virtual, no qual será possível demonstrar de uma forma prática o funcionamento do protocolo, assim como realizar teste e analisar os resultados.

E por fim, na quarta e última etapa, todo o conhecimento que foi obtido no decorrer da pesquisa será demonstrado na simulação com o intuito de apresentar os benefícios em se aplicar as configuração do MPLS-TE.

2 FUNDAMENTAÇÃO TEÓRICA

Ultimamente o desempenho de rede é primordial para o funcionamento dos serviços dos usuários finais. Baseado neste quesito, os provedores de serviços tendem a buscar as melhores maneiras de otimizar a qualidade do serviço em seu *backbone*.

Existem algumas maneiras de tratar o tráfego dentro de um *backbone*, umas delas é através de roteamento de pacotes utilizando os IGPs, como por exemplo o *Intermediate System to Intermediate System* (IS-IS) e *Open Shortest Path First* (OSPF), ambos são protocolos baseados no estado do link (*Link state)*, utilizam o algoritmo *Dijkstra*'s SPF para computar a árvore de menor caminho para os destinos conhecidos dentro de todos os nós de uma rede. (RFC 3906, 2004) [2].

Uma das métricas dos protocolos baseado em *link state* é o custo, o que indica o *overhead* requerido pra enviar pacotes através de uma interface. O custo de uma interface é inversamente proporcional a largura de banda da própria interface, ou seja, maior largura de banda indica um custo menor. Para a criação do melhor caminho para uma rede que utiliza OSPF, por exemplo, o algoritmo leva em consideração o custo das interfaces, além de outros fatores. Há a possibilidade de se ter múltiplos caminhos com o mesmo custo, porém com uma limitação nesta quantidade.

Existem algumas vantagens em se utilizar os protocolos *link state*, eles possuem a capacidade de criar uma mapa topológico, porque os protocolos de roteamento *link state* trocam entre si mensagens contendo a informação do estado do *link* durante a criação da árvore de menor caminho e após ela estabelecida para sempre manter atualizado as melhores opções para encaminhar o tráfego. Devido a sua rápida convergência quando um dispositivo configurado com um protocolo *link state*, recebe um *Link State Advertisement (*LSA), ele inunda as demais interfaces que estão configuradas com o protocolo, exceto a que enviou o LSA, para informar que houve uma mudança na rede, permitindo assim que um novo cálculo do melhor caminho seja feito. Os protocolos de *link state* utilizam o conceito de áreas. Múltiplas áreas criam um *design* de rede hierárquico, permitindo melhor sumarização de rotas e isolando problemas de roteamento dentro de uma área específica. (CCNA – Introduction to Routing Dynamically, 2014) [3].

2.1 *OPEN SHORTEST PATH FIRST (OSPF)*

O protocolo OSPF é um protocolo baseado em estado do link. Ele é designado para rodar dentro de um único sistema autônomo. Cada roteador OSPF mantém uma base de dados idêntica, descrevendo a topologia do sistema autônomo. Através dessa base de dados, a tabela de roteamento é calculada para a construção da árvore de menor caminho.

OSPF recalcula as rotas rapidamente em face de mudanças na topologia, este protocolo também provê suporte a custo igual para múltiplos caminhos. OSPF utiliza o roteamento por áreas, o que permite uma proteção adicional ao roteamento diminuindo assim problemas no encaminhamento dos pacotes. (RFC 2328, 1998)[4]

O protocolo OSPF é classificado como um IGP. Isso significa que ele distribui informações de roteamento entre roteadores pertencentes a um único sistema autônomo. Ele foi desenvolvido por um grupo de trabalho da *Internet Enginnering Task Force* (IETF).

OSPF também provê autenticação para as atualizações de roteamento, e utiliza IP *multicast* quando envia e recebe estas atualizações. OSPF roteia pacotes IP baseado somente no endereçamento IP de destino, localizado no cabeçalho do pacote IP. (RFC 2328, 1998) [4].

OSPF permite que conjuntos de rede trabalhem agrupados, em que cada grupamento é chamado de área. A topologia de uma área é escondida do resto do Sistema Autônomo. Esta informação escondida garante uma redução significante do tráfego de roteamento.

2.1.1 Definições de termos

A seguir serão definidos os termos que tem significado para o protocolo OSPF e será utilizado em todo este trabalho.

- Roteador

Equipamento responsável por fazer o encaminhamento dos pacotes em uma rede roteada, faz parte da camada do 3 do modelo OSI.

- Sistema Autônomo

Um grupo de roteadores trocando informações através de um protocolo de roteamento. Abreviado como AS.

- *Interior Gateway Protocol*

O protocolo de roteamento “falado” por roteadores que pertencem a um único sistema autônomo. Abreviado como IGP. Cada AS tem um único IGP. AS separados podem rodar diferentes IGP.

- *Router ID*

Um número de 32-bit atribuído a cada roteador rodando o protocolo OSPF. Este número identifica o roteador dentro de um AS.

- Máscara de Rede

Um número de 32-bit que indica o *range* de endereçamento IP residindo em uma única rede IP/ sub-rede / super rede.

- Roteadores de vizinhança

Dois roteadores que possuem *interface* com uma rede comum. Relacionamento de vizinhança são mantidos e geralmente são dinamicamente descobertos pelo protocolo OSPF *HELLO*.

- Adjacência

O relacionamento formado entre roteadores de vizinhança com o propósito de trocar informações de roteamento.

- *Link state Advertisement*

Unidade de dados que descreve o estado local de um roteador ou de uma rede. Para um roteador, isto inclui o estado das *interfaces* do roteador e adjacências. Cada *link state advertisement* é inundado por todo o domínio de roteamento. Os anúncios de todos os roteadores e redes são agrupados e formam a base de dados do *link state.* O *link state advertisement* é abreviado com LSA.

- Protocolo *Hello*

A parte do protocolo OSPF usado para estabelecer e manter o relacionamento de vizinhança.

- Roteador designado

Cada rede de *broadcast* ou não *broadcast* (NBMA) que possua ao menos dois roteadores tem um roteador designado. O Roteador designado gera LSA para a rede e tem outra responsabilidade especial no funcionamento do protocolo. O roteador designado ou DR é eleito pelo protocolo *Hello*, ele possibilita a redução do número de adjacências requeridas em uma rede, com isso o tráfego de roteamento e o tamanho da base de dados *link state* diminui. (RFC 2328, 1998)[4].

2.1.2 O *backbone* do sistema autônomo

O *backbone* OSPF é o especial a área 0 do OSPF *e* é sempre aquele que contém todos os roteadores de borda de área. O *backbone* é responsável pela distribuição do roteamento entre uma área não *backbone*. O *backbone* deve ser contíguo. Entretanto, não precisa ser fisicamente contíguo, a conectividade com o *backbone* pode ser estabelecida e mantida através da configuração de *links* virtuais.

Os *links* virtuais podem ser configurados entre qualquer dois roteadores de *backbone* que tenham uma interface em comum com uma área não *backbone*. *Link* virtuais pertencem ao *backbone*. O protocolo trata dois roteadores unidos por uma ligação virtual como se estivessem conectados por uma rede *backbone* ponto a ponto sem numeração.

2.1.3 Roteamento intra-área

Quando há a necessidade de roteamento entre dois áreas que não são *backbone*, o *backbone* precisa ser utilizado. O caminho que o pacote irá percorrer pode ser dividido em três partes: o caminho intra-área entre origem até o roteador de borda da área, um caminho entre origem e destinos intra-área e por fim outro

caminho intra-área até o destino. O algoritmo sempre irá encontrar o caminho que possui o menor custo.

2.1.4 Classificação dos Roteadores

Para o melhor entendimento da topologia OSPF, a seguir será feito uma breve descrição sobre de que maneira são classificados os roteadores.

- Roteadores Internos

Um roteador em que todas as suas conexões diretas pertencem a uma mesma área.

- Roteadores de borda de área

Um roteador que está conectado a múltiplas áreas. Ele executa múltiplas cópias do algoritmo básico, uma para cada área diferente. Os roteadores de borda de área condensam as informações topológicas dos áreas conectadas para distribuição ao *backbone*.

- **Roteadores de *Backbone***

Um roteador que tem uma interface com a área *backbone*. Isso inclui todos os roteadores que tem interfaces com mais de uma área. Entretanto, roteadores de *backbone* não podem ser roteadores de borda de área.

- AS roteador de borda

Um roteador que realiza a distribuição de rotas com roteadores que outros sistemas autônomos para dentro do domínio OSPF.

2.2 *INTERMEDIATE SYSTEM TO INTERMEDIATE SYSTEM (IS-IS)*

O IS-IS é um protocolo de roteamento que provê roteamento eficiente tanto para o modelo *Transmission Control Protocol / Internet Protocol* (TCP/IP) quanto modelo *Open System Interconnection* (OSI). No IS-IS, a rede é particionada em domínios de roteamento. Os limites dos domínios de roteamento são definidos pelo gerenciamento de redes.

No IS-IS existem dois tipos de roteadores:

- Nível 1 sistemas intermediários (IS) – esses nós roteiam baseado no ID do endereçamento ISO, são roteadores que roteiam apenas dentro de uma área. Eles reconhecem, como base no endereço de destino do pacote se o destino está dentro área, se assim for ele roteia o pacote, caso não encaminha para o próximo roteador de nível 2.

- Nível 2 sistemas intermediários (IS) - esse nós roteiam baseados no endereçamento da área. Eles roteiam para as áreas, sem ter em conta a estrutura interna da área. Um IS de nível 2 também pode ser um IS de nível 1 dentro de uma

Assim como o OSPF o IS-IS também trabalha com o conceito de áreas. Os roteadores de nível 1 conhecem a topologia em sua área, incluindo todos os roteadores ou sistemas, roteadores de nível 1 não são capazes de identificar roteadores ou destinos fora da área. Roteadores nível 1 encaminham todo o tráfego com destino fora da área para os roteadores de nível 2 dentro da área. Similarmente, roteadores de nível 2 conhecem toda a topologia de nível 2, e conhecem quais endereços são acessíveis via cada roteador de nível 2. No entanto, roteadores de nível 2 não tem a necessidade de conhecer a topologia dentro de qualquer área no nível 1, exceto para os roteadores que podem ser ao mesmo tempo de nível 2 e nível 1 dentro de uma mesma área. Apenas roteadores de nível 2 podem trocar informações de roteamento ou trocar pacotes de dados diretamente com roteadores fora dos domínios de roteamento. (RFC 1195, 1990)[5]

2.2.1 CLNS

OSI *Connectionless Network Service (*CLNS) é um serviço da camada de rede semelhantemente o IP puro. A entidade CLNS comunica-se através da *connectionless network protocol* com seu par de entidade CLNS. Na arquitetura OSI existem “sistemas”, roteadores são sistemas intermediários (ISs) e *host* são sistemas finais (ESs).

ESs não possuem informações de roteamento, eles descobrem ISs por escutarem *Intermediate System Hellos* (ISHs), e enviam tráfico para qualquer roteador randomicamente.

2.2.2 Operação IS-IS

Roteadores que estão operando com IS-IS irão enviar pacotes *hello* para todas as interfaces que estão com o IS-IS habilitados para descobrir novos vizinhos e estabelecer adjacências.

Roteadores que compartilham de uma ligação comum de dados vão se tornar vizinhos IS-IS, caso os pacotes *hello* contenham os critérios necessários para a formação de uma adjacência. Estes critérios podem variar dependendo do tipo de mídia utilizado, os principais critérios são autenticação, IS-*type* e tamanho da *Maximun Transmission Unit* (MTU).

Roteadores podem construir um *link state packet (*LSP) baseado em suas interfaces locais que estão configuradas com IS-IS e os prefixos que foram aprendidos por outros roteadores adjacentes.

Geralmente, os roteadores inundam LSP para todas as interfaces adjacentes exceto para os vizinhos que já receberam o mesmo LSP. Entretanto, existem diferentes formas de inundação e também uma série de cenários em que cada operação de inundação pode ser diferente.

Todos os roteadores irão construir suas bases de dados a partir desses LSPs. A árvore de menor caminho (SPT) é calculada por cada IS e a partir disso a tabela de roteamento é criada.

2.2.3 Áreas e domínios de roteamento

Um domínio de roteamento IS-IS é similar a um sistema autônomo OSPF, é um conjunto de áreas sob uma administração que implementa políticas dentro de um domínio.

IS-IS não tem uma área *backbone* como a área 0 do OSPF. O IS-IS *backbone* é um conjunto contíguo de roteadores de nível 2, dos quais cada um pode estar em uma área diferente.

Com IS-IS, um roteador individual está em apenas uma área, e a fronteira entre duas áreas é o *link* que conecta dois roteadores que estão em áreas diferentes. Este é um contraste com o OSPF, em que os roteadores de área de fronteira estão dentro de uma área de fronteira. A razão para esta diferença é que roteador IS-IS geralmente tem uma rede de pontos de acesso de serviço (NSAP) e um IP roteador tem múltiplos endereçamentos IP. (Intermediate System-to-Intermediate System Protocol, 2016) [6]

2.2.4 Roteadores de nível 1

Os roteadores de nível 1 conhecem apenas a topologia da sua própria área e tem vizinhança com roteadores nível 1 ou nível 1 para nível 2 nesta área. Possuem a base de dados nível 1 com todas as informações pertinentes para roteamento intra-área. Os roteadores de nível 1 enviam pacotes para fora da área através dos roteadores de nível 2 dentro da sua área.

2.2.5 Roteadores de nível 2

Um roteador de nível 2 pode ter vizinhos na mesma área ou em áreas diferentes, ele possui a base de dados de nível 2 com todas as informações pertinentes para roteamento intra-área. Roteadores de nível 2 tem informações sobre outras áreas, por outro lado, não tem informações sobre os roteadores de nível 1 da sua própria área. No mundo OSI, um roteador deve conhecer a topologia de sua própria área, então roteadores de nível 2 não devem ser configurados quando apenas tráfego OSI está sendo roteado. Se o tráfego em uma área é apenas IP, então todos os roteadores podem ser configurados com nível 2. (Intermediate System-to-Intermediate System Protocol, 2016) [6].

2.2.6 Roteadores de nível 1 / Nível 2

Um roteador de nível 1 / nível 2 pode vizinhos em qualquer área. Ele possuem os dois tipos de base de dados *link state*: uma base de dados *link state* de nível 1 para roteamento intra-área e uma base de dados *link state* de nível 2 para roteamento inter-área.

2.2.7 Vantagens em se usar o protocolo IS-IS

A primeira vantagem em usar IS-IS refere-se ao esforço requerido para o gerenciamento. Desde que o IS-IS forneça um único protocolo de roteamento, dentro de um domínio de roteamento dentro de um *backbone*, isso implica em menos configurações a se fazer.

Outra vantagem do uso do IS-IS é que com menos recursos para rodar somente um protocolo de roteamento (IP por exemplo) os recursos de CPU e memória utilizados no roteador serão menor.

Protocolos de roteamento tem requerimentos de tempo real significantes. No IS-IS esses requerimentos de tempo real são explicitamente especificados. Em outros protocolos de roteamento, esses requerimentos são implícitos. Todavia, em todos protocolos de roteamento, existe garantias de tempo real que devem ser cumpridas para garantir a operação correta.

2.3 *FRAME RELAY (*FR)

O *Frame Relay* é um tecnologia de comunicação de dados em alta velocidade que é utilizada para interligar aplicações do tipo *internet,* voz e dados. A tecnologia *frame relay*, fornece um meio para enviar informações, ela utiliza uma forma simplificada de chaveamento de pacotes que é compatível com vários protocolos e principalmente com o TCP/IP (Teleco, 2003) [7].

Quando o FR é utilizado para a internet, por exemplo, o papel básico desta tecnologia é encapsular o pacote IP, para que ele possa percorrer o meio através de circuitos virtuais e o FR não altera as informações do pacote.

Uma rede baseada em FR, provê um número de *Virtual Circuits (*VC's), para conexões básicas entre estações pertencentes a uma mesma rede FR.

O FR pode ser configurado como ponto-a-ponto ou ponto-multiponto e utiliza a comutação por circuitos para transportar os dados de uma ponta a outra.

FR oferece uma capacidade de comunicação de dados e comutação de pacotes que é usado através de uma interface entre dispositivos de usuários como: roteadores, *bridges* e máquinas de *host* e equipamentos de redes como nós comutados. Dispositivos de usuários geralmente são conhecidos como equipamento terminal de dados (DTE), enquanto equipamentos de redes que faz interface com os equipamentos DTE são conhecidos como equipamento de terminação de circuito (DCE). Um rede FR pode ser fornecida por uma rede pública por um *carrier* ou por uma rede de equipamentos de propriedade privada que serve uma única empresa.

FR difere significativamente do X.25 em sua funcionalidade e formato. Em particular FR é um protocolo mais simples, facilitando maior desempenho e maior eficiência.

Ao longo de um único *link* de transmissão física, o FR prove através de multiplexação estatística muita conversação lógica de dados conhecidos como circuitos virtuais (vc). Isso contrasta com sistemas que usam apenas técnicas de multiplexação por divisão de tempo (TDM) para suportar múltiplos *streams* de dados. A técnica de multiplexação estatística do FR fornece maior flexibilidade e maior eficiência no uso de banda disponível. (Comprehensive Guide to Configuring and Troubleshooting Frame Relay, 2005) [8].

Conforme Jeff T. Buckwalter, Ph. D (2000, p. 31) o FR atual nas camadas 1 e 2 do modelo OSI. O protocolo básico FR é composto por um subconjunto procedimentos de *link* de acesso para *Frame Mode Bearer Services* (LAFP) como definido no ITU-T Q.922. Esse subconjunto é conhecido como protocolo de *core* LAPF e as vezes como as principais funções no enlace de dados.

As principais características fornece um túnel de transferência de quadros de assinante a outro, com um controle de fluxo e controle de erros não sofisticados. Estas funções são:

- Delimitação do quadro, alinhamento e transparência;
- Multiplexação utilizando o campo DLCI e no cabeçalho do quadro;
- Detecção do formato ou transmissão de erros.

2.3.1 Interfaces de rede

O FR possui três tipos de interface são elas *User-Network Interface* (UNI), *Network-to-Network Interface* (NNI) e *Local Management Interface* (LMI).

A UNI é um simples conjunto de procedimentos que permitem que o FR acesse o equipamento para se comunicar com a rede FR. A NNI interconecta duas redes FR. A LMI ajuda garantir que ocorra uma operação válida no FR local, ela não transfere nenhum tipo de tráfego direto para o usuário, porém fornece o *status* e informações de configuração dentro da operação de um PVC através de uma

interface FR. Com o LMI o DTE FR pode solicitar informações sobre o DCE do outro lado.

2.3.2 Circuitos Virtuais (VC)

O FR é baseado na utilização de circuitos virtuais (vc) o vc é circuito virtual de dados bidirecional e dedicado que é configurado entre duas portas quaisquer na rede. Existem dois tipos de circuitos virtuais: *Permanent Virtual Circuit* (PVC) e *Switched Virtual Circuit* (SVC). A diferença entre estes dois tipos de VCs é que um configurado por um operador através um sistema de gerência entre dois pontos na rede (PVC) e o outro SVC que é disponibilizado de maneira automática sem a intervenção do operador. A implementação de ambos na rede exige planejamento visto alguns fatores devem ser considerados antes da configuração como: padrão de tráfego, banda disponível, se a conexão deve ser fixa ou dinâmica, etc. O PVC oferece um ganho estatístico na utilização de banda, por outro lado o SVC permite que haja conectividade entre quaisquer pontos origem e destino dentro da rede.

2.3.2 Estrutura do *Frame*

A estrutura do *frame* do protocolo *Frame Relay* é bastante simples e comum, ela carrega as informações de controle do protocolo, é composto por 2 bytes e abaixo estão as informações conforme Filho B.Huber (Teleco, 2003) [7]:

- DLCI (Data Link Connection Identifier), com 10 bits, representa o número (endereço) designado para o destinatário de um PVC dentro de um canal de usuário, e tem significado local apenas para a porta de origem (vide figura abaixo);
- C/R (Command / Response), com 1 bit, é usado pela aplicação usuária;
- FECN (Foward Explicit Congestion Notification), com 1 bit, é usado pela rede para informar um equipamento receptor de informações que procedimentos de prevenção de congestionamento devem ser iniciados;
- BECN (Backward Explicit Congestion Notification), com 1 bit, é usado pela rede para informar um equipamento transmissor de informações

que procedimentos de prevenção de congestionamento devem ser iniciados;
- DE (Discard Eligibility Indicator), com 1 bit, indica se o frame pode ser preferencialmente descartado em caso de congestionamento na rede;
- EA (Extension Bit), com 2 bits, é usado para indicar que o cabeçalho tem mais de 2 bytes, em caso especiais; (Teleco, 2004)

2.4 *ASYNCRONOUS TRANSFER MODE (ATM)*

O ATM é uma tecnologia de comunicação de alta velocidade que é utilizada em redes locais e de longa distância para diversas aplicações como dados e voz. O ATM provê funcionalidades tanto para as redes comutadas por circuitos quanto para as redes comutadas por pacotes. Ele utiliza multiplexação assíncrona por divisão de tempo.

Assim com o *frame relay*, o ATM utiliza circuitos virtuais. Os responsáveis pela conversão dos dados para o protocolo ATM são os equipamentos de acesso. A conexão entre os pontos dentro da nuvem ATM são realizados através de caminhos virtuais que são configurados com uma determinada banda.

O ATM, otimiza a utilização de banda, que um recurso muito útil para a engenharia de tráfego, além de possibilitar a integração de vários tipos de tráfegos diferentes (dados, voz e vídeo), possui múltiplas classes de Qualidade de Serviço (QoS), contém alta disponibilidade de serviço e pode incorporar outros protocolos e aplicações como: *Frame Relay,* DSL, *Gigabit Ethernet,* tecnologias *wireless*, SDH/SONET, entre outros. (Teleco, 2003) [9].

Diferentemente do FR o ATM utiliza tamanho de frame fixo que é chamado de *cell* (célula), esta célula possui 53 bytes, onde 48 bytes são os dados e 5 bytes para o cabeçalho. Cada célula ATM contém uma informação de endereçamento para que seja possível o estabelecimento de uma conexão entre a origem e o destino.

O ATM é um protocolo orientado a conexão, ele precisa que haja uma processo de troca de sinalização para o estabelecimento da conexão. Para o início de uma nova conexão a origem envia uma sinalização até o destino, se o destinatário concorda com a conexão, então, um *Virtual Channel Connection* (VCC) e *Virtual Path Connection* (VPC) é estabelecido na rede, de modo que, um *Virtual Path Identifier* (VPI) e um *Virtual Channel Identifier* (VCI) é adotado. Estas

informações são enviadas para as tabelas de roteamento dos equipamentos de rede, que as usam para encaminhar as células. (Teleco, 2003) [9]

O ATM possuí células fixas de 55 bytes, algumas das vantagens são:

- Alta velocidade de chaveamento;
- Qualquer tipo de informação pode ser transmitida por células ATM (dados, vídeo e áudio);
- Alocação dinâmica de banda;
- *Switching* / Circuitos;
- Canais Virtuais;
- Serviço orientado a conexão;
- QoS (*Quality of Service)*;
- UNI;
- NNI.

Todo tráfego de entrada em uma rede ATM é transformado em células ATM. A camada de adaptação do ATM (AAL) fornece suporte para diferentes tipos de tráfego e a camada de segmentação e remontagem (SAR) monta e desmonta o tráfego que entra e o tráfego que sai.

2.4.1 Formato da célula ATM

Na figura 1 iremos ver o formato de uma célula ATM, 5 bytes são utilizados para o cabeçalho e os 48 bytes são utilizados para *payload*.

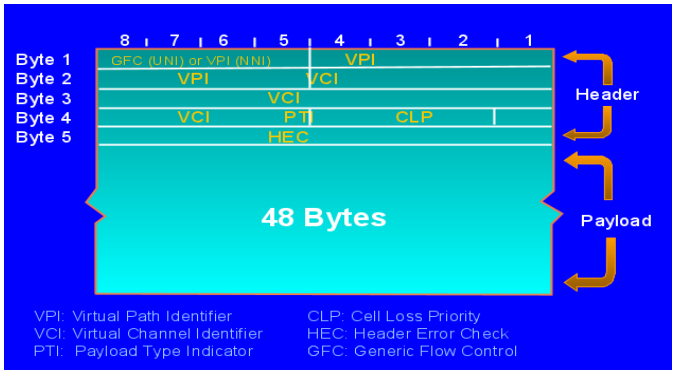


Figura 1: Formato da célula ATM.
Fonte: Guenaga (2001).

A seguir será feito uma breve descrição sobre cada parte da célula ATM:

VPI – *Virtual Path Identifier;*

VCI – *Virtual Channel Identifier;*

- VPI/VCI: Identifica a conexão VP/VC

- Canais de usuários: VCI > 32

- Canais de gerência e sinalização: VC 0-15: Falha, tráfego, desempenho de VP, sinalização, etc.

- Canais Engenharia de Rede: 16-31: SMDS, LANE, ILMI, PNNI.

PTI – *Payload Type Identifier*

- Canais para supervisão e gerência;

- Suporte para congestionamento, falhas, tráfego, desempenho, etc.

CLP – *Cell Loss Priority*

- 2 Classes de QoS;

- Critério de descarte em congestionamento;

- Controle de tráfego prioritário e agregado.

- GFC – *Generic Flow Control*

- Controle de tráfego em múltiplos acessos sobre o meio compartilhado.

2.4.2 Rede ATM em camadas

A rede ATM é dividida em camadas, conforme a figura 2 esta divisão é feita em três camadas: PHY; ATM; AAL.

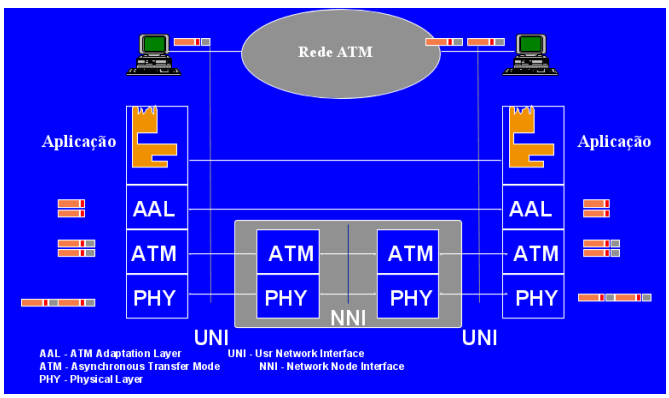


Figura 2: Rede ATM em camadas
Fonte: Guenaga (2001).

Na camada PHY (camada física), ocorre a adaptação de taxa variável do serviço para o meio de transmissão (PDH, SDH, etc.), assim como a sincronização de células e adaptação ao quadro de transmissão e ocorre a codificação de linha para a transmissão digital.

Na camada ATM o cabeçalho é adicionado aos 48 bytes de informação, a conexão é identificada as conexões virtuais são multiplexadas e as células da conexão virtual são transmitidas sequencialmente, além disso o processamento na rede é feito: roteamento, controle de tráfego e de prioridade, suporte para a sinalização e informações de operação e manutenção.

Na camada de adaptação (AAL), é feito o mapeamento da aplicação para ATM, o *payload de* 48 bytes é segmentado e remontado e também ocorre o tratamento de erros e atrasos. Dentro da camada AAL existem subcamadas que exercem algumas funções:

- AAL 1: acumula bytes em blocos de 48 bytes;
    - serviços de taxas constante;
    - emulação de circuitos digitais;
    - serviços sensíveis e *delay;*
- AAL 2: serviço de tempo real em taxa variável
    - vídeo e áudio;
    - serviços sensíveis e *delay.*
- AAL 3/4: segmenta mensagens grandes em blocos de 48 bytes
    - Serviços de dados em taxa variável
    - transferência de dados sensível a erro
- ALL 5: serviços de dados em taxa variável
    - transferência de dados sensível a erro com baixo *overhead*.

2.4.3 Identificadores de conexão lógica

Assim como o protocolo FR o protocolo ATM também trabalha com os conceitos de circuitos e conexões virtuais:

- *Virtual Channel Connection* (VCC): Cada conexão é alocada por uma combinação de VPI/VCI e a combinação de VPI/VCI guia a célula ATM através da rede.
- *Virtual Path Connection* (VPC) – identifica o caminho lógico do grupo de canais, não deve assumir um *path ID* igual ao *port* ID, e muitos *paths* podem operar sobre um único *port*.

2.4.4 Tipos de conexões

Analogamente aos tipos de conexões no FR o ATM também possui PVC e SVC.

- PVC – Conexão iniciada pelo administrador, estabelecida e liberada manualmente, conexões geralmente são longas.
- SVC – Conexão iniciada pelo usuário, estabelecida e liberada dinamicamente, estabelecida via protocolo de sinalização, duração da conexão varia.

2.5 *MULTIPROTOCOL LABEL SWITCHING (MPLS)*

Um pacote viaja de um roteador para o outro, cada roteador faz o encaminhamento independente da decisão para aquele pacote, o roteador analisa o cabeçalho do pacote e executa um algoritmo de roteamento da camada de rede.

O cabeçalho do pacote IP contém consideravelmente muita mais informação do que o necessário para encontrar o próximo salto. Após a análise do cabeçalho os roteadores associam um possível conjunto de pacotes dentro de um *Forwarding Equivalence Classes* (FECs). Em um segundo momento cada FEC é mapeado para um próximo salto. Todos os pacotes pertencentes a um particular FEC e que seguirão até uma determinado nó, irão seguir o mesmo caminho. (RFC 3031, 2001)[10].

No MPLS, a associação de um conjunto particular de pacotes para um determinado FEC é feito apenas uma vez, assim que o pacote entre na rede. O FEC no qual o pacote é associado é codificado como um valor de tamanho fixo, conhecido com *label*. Quando um pacote é encaminhado para o próximo salto, o rótulo é enviado junto com ele, isto é, os pacotes são “rotulados” antes de serem encaminhados. (RFC 3031, 2001)[10]

No paradigma do encaminhamento utilizando MPLS, uma vez que o pacote é associado a um FEC, não é mais necessário a análise do cabeçalho IP, então encaminhamento é feito através dos *labels.*

Uma das vantagens do MPLS é que possível manipular o encaminhamento dos *labels*, através de uma caminho específico, diferente daquele que já foi definido pelo algoritmo de roteamento dinâmico. Isso pode ser feito através de política, ou como será abordado neste trabalho, para suportar a engenharia de tráfego. No encaminhamento convencional, o pacote precisa carregar consigo o endereçamento de origem, sendo sempre necessário que ao roteador analise o cabeçalho e saiba para qual origem deve enviar aquela informação que foi solicitada. No MPLS um *label* pode ser utilizada para representar o trajeto, de modo que, a identidade do percurso expresso não precisa ser realizada com o pacote. MPLS fornece um mecanismo simples de tunelamento integrado com o protocolo IP. (RFC 3031, 2001)[10]

2.5.1 Terminologia MPLS

A seguir será realizada uma breve descrição sobre os principais termos utilizados em uma malha MPLS.

Forwarding Equivalence Class – abreviado como FEC, trata-se de um grupo de pacotes IP que são encaminhados da mesma maneira (ex: através de um mesmo caminho, como o mesmo tratamento de encaminhamento).

Label – trata-se de um identificador de tamanho fixo, que serve para identificar um FEC.

Label Switching Path – abreviado como LSP, trata-se do caminho entre um ou mais LSR em um nível de hierarquia seguido por pacotes em uma determinada FEC.
Label Switching Router – abreviado como LSR, trata-se de um nó MPLS que é capaz de encaminhar pacotes L3.
Routing Information Base – abreviado como RIB, trata-se da tradicional tabela de roteamento onde todas as rotas conhecidas são fixadas, sejam por roteamento estático, dinâmico ou diretamente conectadas.
Label Information Base - abreviado como LIB, trata-se da tabela que faz a associação entre os *labels* e às interfaces dos roteadores. O LSR utiliza esta tabela para determinar por qual interface o pacote deverá ser encaminhado.
Forwarding Information Base – abreviado como FIB, trata-se da tabela versão otimizada da RIB, ou mais corretamente é a tabela que o roteador checa quando precisa determinar para onde o tráfego deverá ser encaminhado.
Label Forwarding Instance Base – abreviado como LFIB, trata-se da tabela que o roteador utiliza para os pacote “rotulados” através da rede. Assim como a RIB usa a FIB para encaminhar o tráfego, então a LIB utiliza a LFIB para encaminhar o tráfego. RFC 3031, 2001)[10]

2.5.2 Cabeçalho MPLS

MPLS utilizada um 32 bit em um cabeçalho *shim*, na figura 3 será possível observar a estrutura do cabeçalho MPLS.

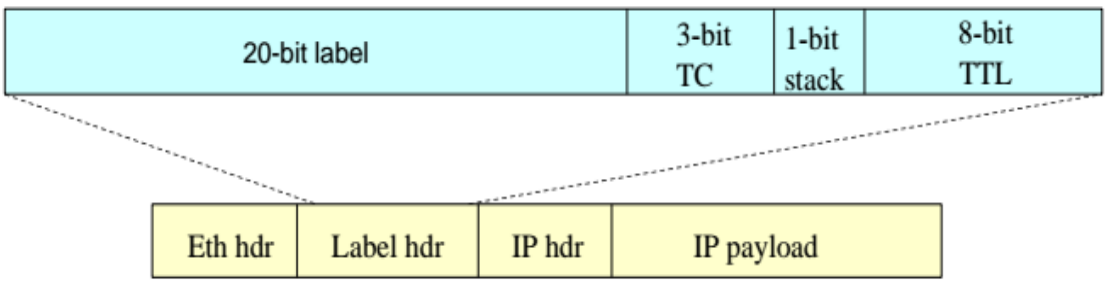


Figura 3: Cabeçalho MPLS.
Fonte: Olof Hagsand KTH CSC (2011).

- ***LABEL:*** valor para a consulta no roteador.
- ***Traffic Class* (TC):** é o campo da classificação do tráfego, pode ser usado como *class-of-service* para QoS.
- ***STACK:*** indica que a parte inferior de um *stack labels* foi alcançado.
- ***Time to Live* (TTL):** assemelha-se ao TTL do protocolo IP.

2.5.3 *Label*

O *label* trata-se de um número inteiro que identifica um FEC, não se pode ter *labels* globais ou uma rede de *labels*, eles são únicos entre dois nós, podem ser valores entre 0-1048575, sendo que entre 0-15 são reservados pelo IETF. Os *labels* são trocados assim que o pacote atravessa a rede, eles podem ser manualmente configurados na rede ou usar a distribuição automática de *label*. Na figura 4 será possível ver um exemplo desta operação:

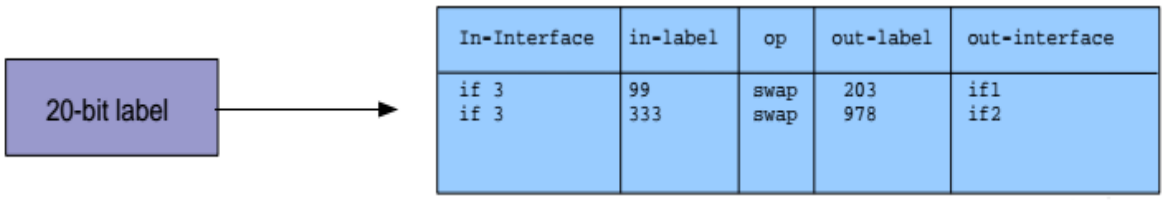


Figura 4: Exemplo de uma LFIB.
Fonte: Olof Hagsand KTH CSC (2011).

O *label* possui diferentes tipos de operação em uma rede MPLS abaixo será feito uma breve descrição sobre estas operações:

- *LABEL PUSHING:* O roteador de borda classifica os pacotes em FECs, associa um *label* para um pacote, na verdade ele mapeia o FEC para um LSP que por sua vez define um *label*. “Empurra” um cabeçalho MPLS no pacote e então o encaminha através da interface do LSP. Na figura 5 será possível ver um exemplo desta operação:

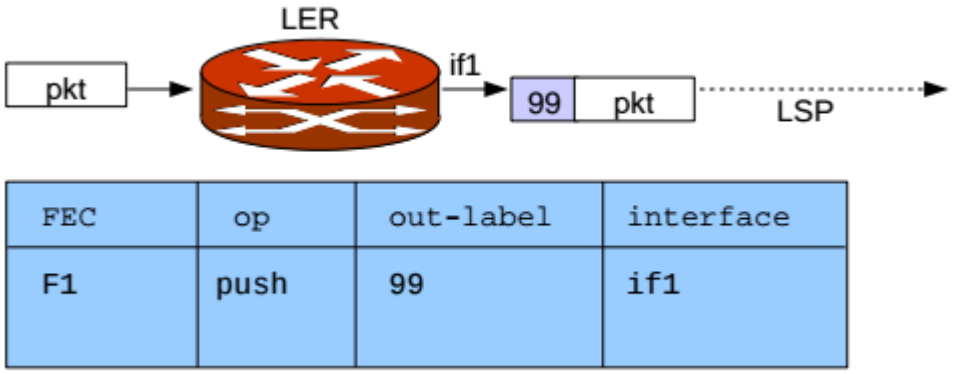


Figura 5: Operação *Label Pushing*.
Fonte: Olof Hagsand KTH CSC (2011).

- *LABEL SWAPPING:* O roteador (LSR) faz uma pesquisa de *label* e troca o *label*, reescreve o cabeçalho MPLS e envia o *label* através do LSP. Na figura 6 será possível ver um exemplo desta operação:

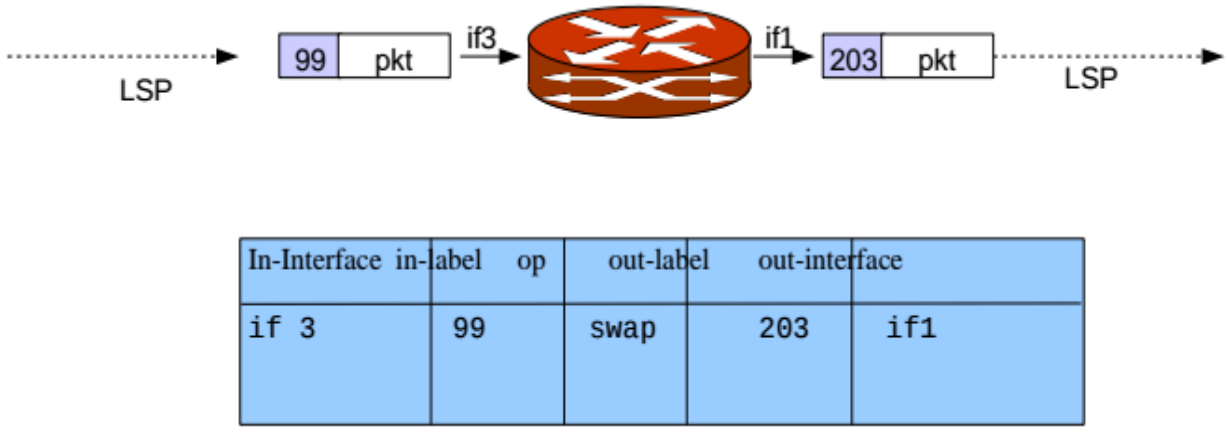


Figura 6: Operação *Label Swapping*.
Fonte: Olof Hagsand KTH CSC (2011).

- *LABEL POPPING*: O roteador (LER) retira o pacote MPLS e então encaminha o pacote como de costume dependendo do protocolo de pacote: Ex: o pacote é um pacote IP, então o pacote é enviado para o encaminhamento IP. Na figura 7 será possível ver um exemplo desta operação:

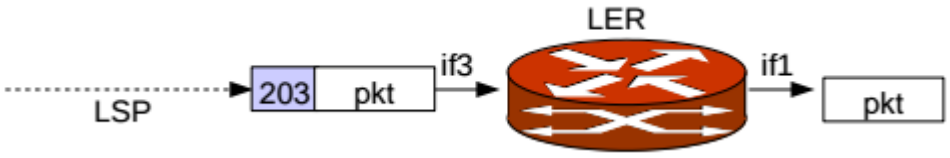
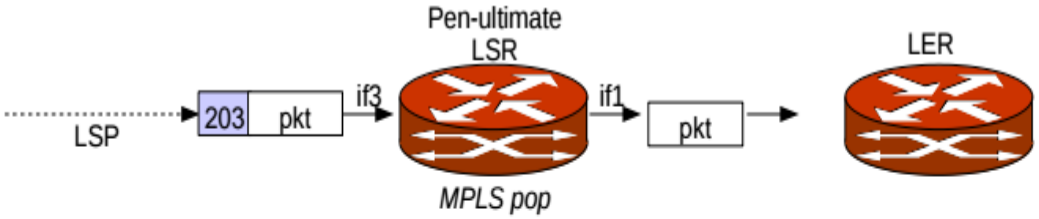


Figura 7: Operação *Label Popping*.
Fonte: Olof Hagsand KTH CSC (2011).

- *PEN-ULTIMATE POPPING*: Para facilitar para o roteador de borda, o *label* é retirado do roteador antes (penúltimo), então o penúltimo LSR faz o MPLS pop, e o LER "fala" apenas roteamento IP. Na figura 8 será possível ver um exemplo desta operação:



IP routing

In-Interface	in-labelop	out-label	out-interface	
if 3	203	pop	-	if1

Figura 8: Operação *Label Pen-Ultimate Popping*.
Fonte: Olof Hagsand KTH CSC (2011).

Além das operações citadas anteriormente, os *label* ainda desempenha algumas operações especiais para o funcionamento do MPLS:

- 0: IPv4 *explicit* NULL: o *downstream* LSR deve retirar o *label* incondicionalmente, o pacote do qual o *label* foi retirado trata-se de uma datagrama IPv4.
- 1: Roteador alerta: encaminhado para o painel de controle, não encaminhado.
- 2: IPv6 *explicit* NULL: o *downstream* LSR deve retirar o *label* incondicionalmente, o pacote do qual o *label* foi retirado trata-se de uma datagrama IPv6.
- 3: *Implicit* NULL: retira o *label* imediatamente é trata com um pacote IPv4, este *label* na verdade não aparece no *link*, usado como *pen-ultimate popping*.

2.5.4 Roteadores em um LSP

Cada roteador em LSP exerce uma determinada função como será descrito abaixo:

- **Roteadores de ingresso**: trata-se do roteador que fica no início de um LSP. Ele é responsável por encapsular o pacote IP em MPLS L2 e encaminha-lo para o próximo roteador no caminho. Cada LSP pode ter apenas um roteador de ingresso.
- **Roteadores de egresso:** trata-se do roteador que fica no fim de um LSP. Ele é responsável pela remoção do encapsulamento MPLS, transformando-o de um pacote MPLS para um pacote IP, tem a função de encaminhar o pacote até o destino final utilizando as informações na tabela de roteamento.
- **Roteadores de transito:** qualquer roteador intermediário no LSP entre os roteadores de ingresso e egresso. O roteador de transito encaminha os pacotes MPLS para o próximo roteador no caminho MPLS. Um LSP pode conter ou nenhum ou mais roteadores de transito, em um máximo de 253 roteadores de transito em um único LSP. (Juniper, 2013)[11].

2.5.5 Tipos de LSP

Existem três tipos de LSP:

- **LSP estático**: para caminhos estáticos, é preciso assinalar os *labels* manualmente em todos os roteadores envolvidos (ingresso, egresso e transito). Nenhum protocolo de sinalização é necessário. Este procedimento é similar ao de se configurar rotas estáticas em roteadores.
- **LSP sinalizado por LDP** – *Label Distribution Protocol* (LDP) é um protocolo para a distribuição de *labels*, ele permite roteadores estabelecerem LSP através de uma rede, mapeando todas as informações de roteamento da camada de rede para caminhos de comutação da camada de enlace.
- **LSP sinalizado por RSVP** - *Resource Reservation Protocol* (RSVP), para caminhos sinalizados, RSVP é utilizado para estabelecer o caminho e dinamicamente assinalar os *labels*. Deve ser configurado apenas no roteador de ingresso. Os roteadores de transito e de egresso aceitam as informações de sinalização a partir do roteador de ingresso e eles mantém cooperativamente o LSP. Qualquer erro encontrando enquanto o LSP está sendo estabelecido é enviado ao roteador de ingresso para diagnósticos. Para sinalizar um LSP utilizando RSVP, a versão do RSVP que suporta extensão de túnel deve estar habilitada em todos os roteadores. (Juniper, 2013)[11]

Na figura 9 será possível um tipo de MPLS LSP.

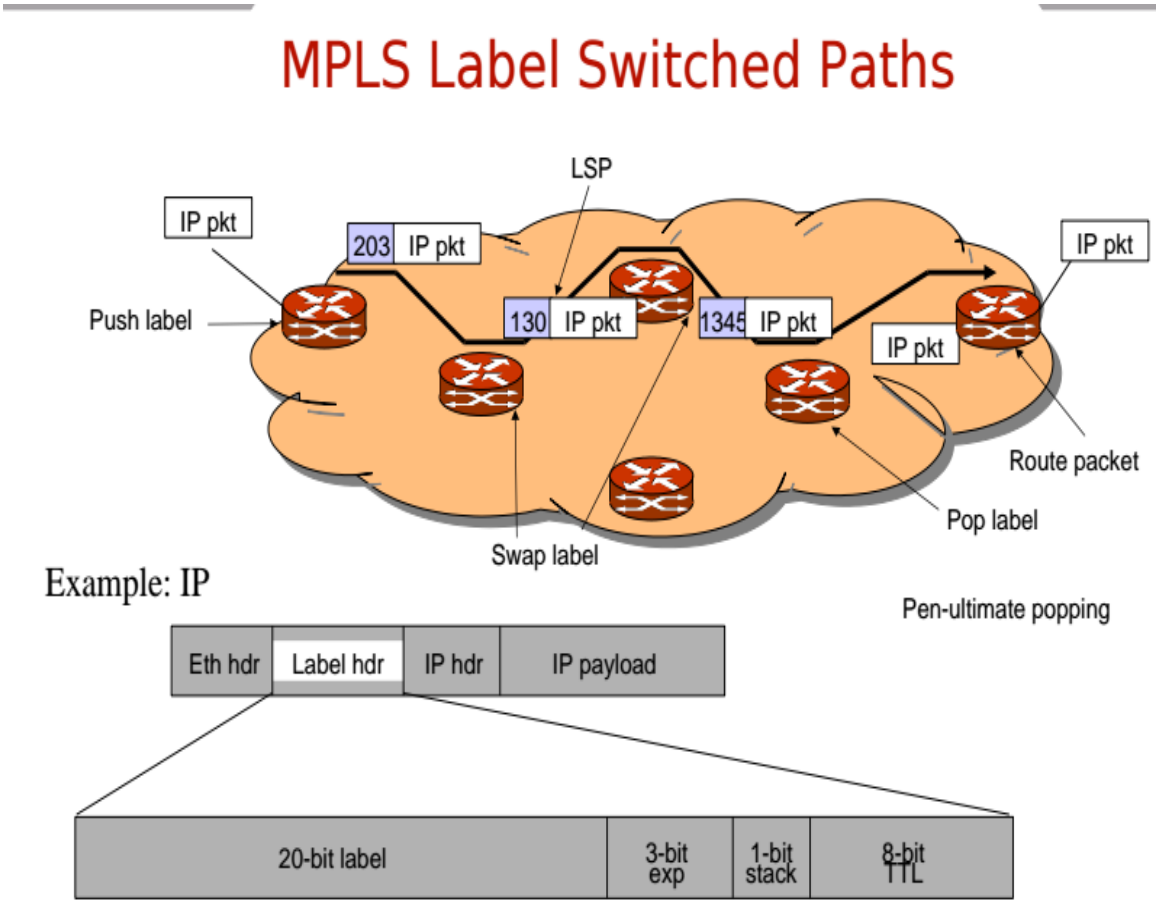


Figura 9: MPLS *Lable Switched Paths*
Fonte: Olof Hagsand KTH CSC (2011).

3 MPLS *TRAFFIC ENGINEERING* – MPLS-TE

A engenharia de tráfego permite o controle dos pacotes de dados e seguem, ignorando o modelo de roteamento padrão, que utiliza tabelas de roteamento. Com a engenharia de tráfego é possível comutar o tráfego de *links* congestionados para *links* alternativos que não seriam selecionados pelo algoritmo de definição de caminho mais curto. O MPLS-TE fornece a capacidade de mover o fluxo de tráfego para longe do caminho mais curto selecionado pelo IGP e para um caminho físico potencialmente menos congestionado em uma rede. Com o MPLS-TE é possível:

- Utilizar de maneira mais eficiente o uso fibras de longa distâncias;
- Controlar como o tráfego é roteado em face de uma ou várias falhas;
- Classificar o tráfego crítico e regular em uma base *per-path;*
- Rotas de uso primário em torno de pontos congestionados em uma rede;
- Fornece controle preciso através de como o tráfego será re-roteado quando um caminho primário se depara com uma ou múltiplas falhas;
- Proporciona uma utilização mais eficiente da largura de banda agregada de fibras de longa distância disponíveis, garantindo que subconjuntos de rede não fiquem subutilizados enquanto outros subgrupos da rede, juntamente com potenciais caminhos alternativos são subutilizados;
- Maximiza a eficiência da operação;
- Melhora as características de performance de tráfego orientado de uma rede e minimiza a perda de pacotes, minimizando períodos prolongados de congestionamentos e máxima o *throughput*.
- Melhora as características estatisticamente de desempenho de rede (como *loss ratio,* variação de *delay* e transferência de *delay)*, necessárias para apoiar os multisserviços utilizados na Internet.

O núcleo do *design* do MPLS-TE é baseada na construção de LSP entre os roteadores. Um LSP é uma conexão orientada, assim como um circuito virtual no *Frame Relay* ou ATM. LSPs não são confiáveis, pacotes que entram em um LSP não tem garantias que serão entregues, embora tratamento preferencial seja possível. LSPs também são similares aos túneis unidirecionais em que pacotes que entram em um caminho são encapsulados em um “envelope” e são comutados em todo o caminho sem serem trocados por nós intermediários. LSP fornece controle

refinado sobre como os pacotes são enviados em uma rede. Para fornecer confiabilidade, um LSP pode usar um conjunto de caminhos primários e secundários.

LSPs podem ser configurados para apenas tráfego BGP (tráfego este que é destinado para fora do AS). Neste caso, tráfego dentro do AS não é afetado pela presença do LSP. LSP também podem ser configurado para ambos tráfego BGP e tráfego IGP, portanto, ambos intra-AS e inter-AS são afetados pelos LSPs. (Juniper, 2013)[12].

3.1 POR QUE USAR MPLS-TE?

Conexões de WAN são um item caro para o orçamento de um *Internet Service Provider* (ISP). A engenharia de tráfego permite aos ISPs rotear o tráfego para oferecer o melhor serviço para os seus usuários em termos de rendimento de rede e *delay,* por tornar mais eficiente o fornecimento do serviço, a engenharia de tráfego reduz o custo da rede.

Atualmente alguns ISPs baseiam seus serviços em um *overlay model*. No *overlay model*, as facilidades de transmissão são gerenciadas por comutação L2. Os roteadores enxergam apenas a topologia virtual *full mesh*, tornando a maioria dos destinos apenas um *hop* de distância. Se utilizar como transito o *explicit* L2, é possível controlar com precisão as formas em que o tráfego utiliza a largura de banda disponível. Entretanto, os *overlay models*, possuem um certo número de desvantagens. MPLS-TE fornece uma maneira de obter os mesmos benefícios da engenharia de tráfego dos *overlay models* sem a necessidade de rodar em uma rede separada, e sem precisar de um *full mesh* não escalável de roteadores interconectados. (Cisco, 1999) [13].

3.2 COMO O MPLS-TE TRABALHA

MPLS é uma integração das tecnologias de *layer 2* e *layer 3*. Para fazer os recursos de L2 disponíveis para L3, MPLS habilita a engenharia de tráfego. Assim, pode-se oferecer uma rede *one-tier* que agora apenas pode ser alcançada sobrepondo uma rede L3 em uma rede L2.

MPLS-TE automaticamente estabelece e mantém LSPs através de um *backbone*, utilizando RSVP. O caminho utilizado para um dado LSP até um ponto qualquer no tempo é baseado nos requerimentos de recursos do LSP e nos recursos de rede, como largura de banda.

Recursos disponíveis são inundados através de extensões para o *link-state* baseado no IGP. Caminhos para o LSPs são calculados como em um ajuste necessário e os recursos disponíveis (roteamento baseado em restrições). O IGP automaticamente roteia tráfego a estes LSPs. Tipicamente, o pacote atravessa o *backbone* MPLS-TE viajando em um único LSP que conecta o ponto de ingresso ao ponto de egresso.

A engenharia de tráfego é construída pelos seguinte mecanismos:

- Do ponto de vista L2, uma interface túnel LSP representa o *head* de um LSP. Isto é configurado como um conjunto de requerimentos de recursos, como largura de banda, requerimentos de mídia e prioridade;
- Do ponto de vista L3, uma interface túnel LSP é o *head-end* de um *link* virtual unidirecional para o túnel de destino.
- Um módulo de cálculo do caminho de engenharia de tráfego MPLS: este mecanismos opera no LSP *head.* Isto determina um caminho a ser usado por LSP usando um banco de dados *link-state* contendo informações de topologia e de recursos.
- RSVP com extensão de engenharia de tráfego: isto opera em cada *hop* LSP e é usado para sinalizar e manter LSPs baseado no caminho calculado.
- Módulo de gerenciamento de *link* MPLS-TE: este módulo opera em cada *hop* LSP faz admissão de *link call* no RSVP, sinaliza mensagens e mantém as informações da topologia que serão inundadas.

- Um IGP *link-state* (IS-IS ou OSPF – cada um com extensão para engenharia de tráfego): estes IGPs são utilizados globalmente para inundar informações de topologia e informações de recursos a partir do módulo de gerenciamento.
- Melhoramento pra o cálculo SFP usado pelos IGPs *link-state*: eles automaticamente roteiam o tráfego para o apropriado LSP túnel baseado no túnel de destino. Rotas estática também podem ser usadas para tráfego direto para túneis LSPs.
- *Label Switching Forwarding*: este mecanismo de encaminhamento fornece roteadores com uma capacidade L2 para direcionar o tráfego direto através de múltiplos *hops* de LSP que foi estabelecido pela sinalização RSVP.

4 LAB – CONFIGURAÇÃO MPLS-TE UTILIZANDO EXPLICIT PATH

O experimento a seguir foi realizado utilizando o *software* GNS3 que tem como objetivo principal mostrar o funcionamento do MPLS-TE como uma ferramenta de extrema importância para a manipulação de tráfego que permite garantir o controle e eficiência na entrega dos serviços de um ISP.

Basicamente forem utilizados quatro roteadores cisco 7200 rodando a versão de IOS 12.4, abaixo na figura 10 está representa a topologia utilizada para realizar o lab:

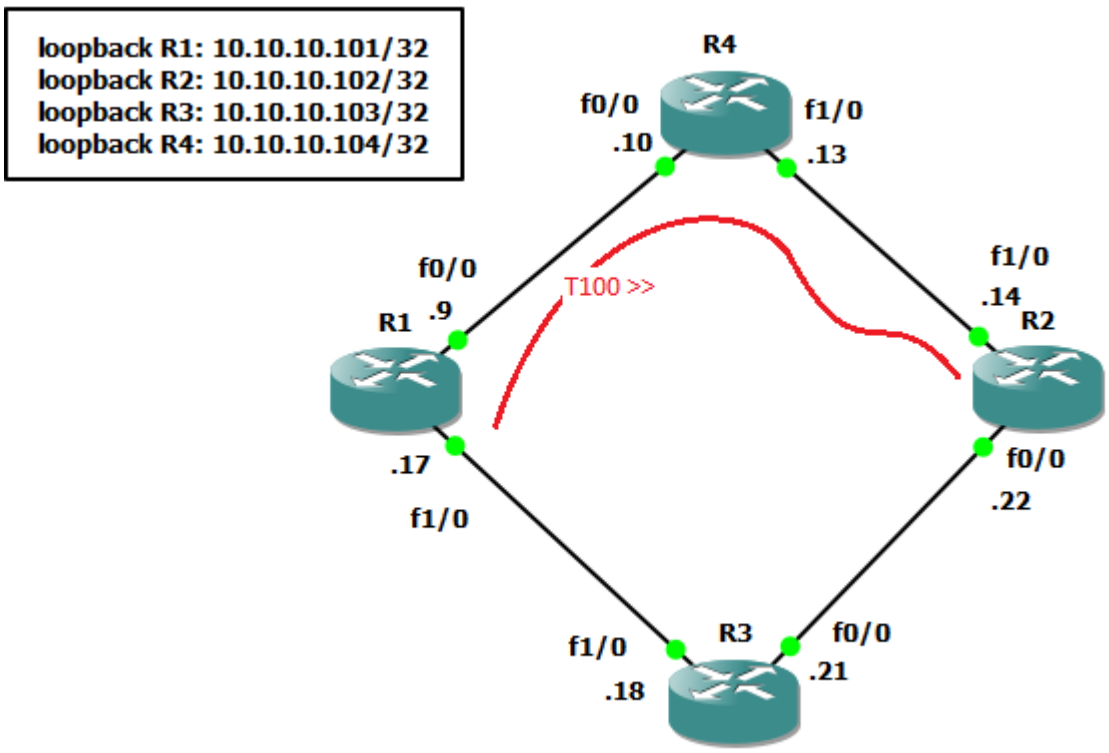


Figura 10: Topologia do lab MPLS-TE
Fonte: Autoria própria.

Para esta simulação foi utilizado o protocolo OSPF como IGP para anúncio da rede 10.10.10.0/24 o OSPF está utilizando como *router-id* a interface de *loopback* de cada roteador, a distribuição de *labels* está sendo feito através do protocolo LDP que foi habilitado de maneira global em todos os roteadores. Foi configurado RSVP nas

interfaces que falam MPLS utilizando uma reserva de banda de 50000 Kbps. Um túnel foi criado para o encaminhamento do tráfego entre os roteadores R1 e R2, o túnel foi identificado como túnel 100 e foi configurado com *explicit path* com prioridade de 10, o que significa que este será o caminho escolhido para o tráfego ser encaminhado, orientando para que o tráfego siga através dos roteadores 1 <> roteador 4 <> roteador <> 2, como alternativa para garantir a redundância foi configurado o *dynamic path* com prioridade 20, em caso de alguma falha, o protocolo calcula dinamicamente por qual caminho o tráfego será encaminhado.

4.1 CONFIGURAÇÃO DOS ROTEADORES

Abaixo segue as configurações que foram realizadas em cada roteador:

ROTEADOR 1

```
R1#show running-config
Building configuration...

Current configuration : 3105 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R1
!
ip cef
no ip domain lookup
!
!
mpls label protocol ldp
```

```
mpls traffic-eng tunnels
!
interface Loopback0
 ip address 10.10.10.101 255.255.255.255
!
interface Tunnel100
 description from_R1_R4_R2
 ip unnumbered Loopback0
 tunnel destination 10.10.10.102
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  25000
 tunnel mpls traffic-eng path-option 10 explicit name R1_R4_R2
 tunnel mpls traffic-eng path-option 20 dynamic
 no routing dynamic
!
interface FastEthernet0/0
 description to R2
 bandwidth 100000
 ip address 10.10.10.9 255.255.255.252
 duplex full
 mpls ip
 mpls traffic-eng tunnels
 ip rsvp bandwidth 50000
!
interface FastEthernet1/0
 description to R3
 bandwidth 100000
 ip address 10.10.10.17 255.255.255.252
 duplex half
 mpls ip
 mpls traffic-eng tunnels
```

```
 ip rsvp bandwidth 50000
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 10.10.10.0 0.0.0.255 area 0
!
ip explicit-path name R1_R4_R2 enable
 next-address 10.10.10.10
 next-address 10.10.10.14
 next-address 10.10.10.102
!
```

ROTEADOR 2

```
R2#show running-config
Building configuration...

Current configuration : 3105 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R2
!
ip cef
no ip domain lookup
!
!
mpls label protocol ldp
mpls traffic-eng tunnels
```

```
!
interface Loopback0
 ip address 10.10.10.102 255.255.255.255
!
interface Tunnel100
 description from_R1_R4_R2
 ip unnumbered Loopback0
 tunnel destination 10.10.10.102
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  25000
 tunnel mpls traffic-eng path-option 10 explicit name R1_R4_R2
 tunnel mpls traffic-eng path-option 20 dynamic
 no routing dynamic
!
interface FastEthernet0/0
 description to R3
 bandwidth 100000
 ip address 10.10.10.22 255.255.255.252
 duplex full
 mpls ip
 mpls traffic-eng tunnels
 ip rsvp bandwidth 50000
!
interface FastEthernet1/0
 description to R4
 bandwidth 100000
 ip address 10.10.10.14 255.255.255.252
 duplex half
 mpls ip
 mpls traffic-eng tunnels
 ip rsvp bandwidth 50000
```

```
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 10.10.10.0 0.0.0.255 area 0
!
ip explicit-path name R1_R4_R2 enable
 next-address 10.10.10.10
 next-address 10.10.10.14
 next-address 10.10.10.102
!
```

ROTEADOR 3

```
R3#show running-config
Building configuration...

Current configuration : 3105 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R3
!
ip cef
no ip domain lookup
!
!
mpls label protocol ldp
mpls traffic-eng tunnels
```

```
!
interface Loopback0
 ip address 10.10.10.103 255.255.255.255
!
interface Tunnel100
 description from_R1_R4_R2
 ip unnumbered Loopback0
 tunnel destination 10.10.10.102
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  25000
 tunnel mpls traffic-eng path-option 10 explicit name R1_R4_R2
 tunnel mpls traffic-eng path-option 20 dynamic
 no routing dynamic
!
interface FastEthernet0/0
 description to R2
 bandwidth 100000
 ip address 10.10.10.21 255.255.255.252
 duplex full
 mpls ip
 mpls traffic-eng tunnels
 ip rsvp bandwidth 50000
!
interface FastEthernet1/0
 description to R1
 bandwidth 100000
 ip address 10.10.10.18 255.255.255.252
 duplex half
 mpls ip
 mpls traffic-eng tunnels
 ip rsvp bandwidth 50000
```

```
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 10.10.10.0 0.0.0.255 area 0
!
ip explicit-path name R1_R4_R2 enable
 next-address 10.10.10.10
 next-address 10.10.10.14
 next-address 10.10.10.102
!
```

ROTEADOR 4

```
R4#show running-config
Building configuration...

Current configuration : 3105 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R4
!
ip cef
no ip domain lookup
!
!
mpls label protocol ldp
mpls traffic-eng tunnels
```

```
!
interface Loopback0
 ip address 10.10.10.104 255.255.255.255
!
interface Tunnel100
 description from_R1_R4_R2
 ip unnumbered Loopback0
 tunnel destination 10.10.10.102
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  25000
 tunnel mpls traffic-eng path-option 10 explicit name R1_R4_R2
 tunnel mpls traffic-eng path-option 20 dynamic
 no routing dynamic
!
interface FastEthernet0/0
 description to R1
 bandwidth 100000
 ip address 10.10.10.10 255.255.255.252
 duplex full
 mpls ip
 mpls traffic-eng tunnels
 ip rsvp bandwidth 50000
!
interface FastEthernet1/0
 description to R2
 bandwidth 100000
 ip address 10.10.10.13 255.255.255.252
 duplex half
 mpls ip
 mpls traffic-eng tunnels
 ip rsvp bandwidth 50000
```

```
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 10.10.10.0 0.0.0.255 area 0
!
ip explicit-path name R1_R4_R2 enable
 next-address 10.10.10.10
 next-address 10.10.10.14
 next-address 10.10.10.102
!
```

4.2 TESTES – MOSTRANDO OS RESULTADOS OBTIDOS

Para a comprovação da teoria na prática a seguir será apresentado os resultados dos testes realizados em laboratório. Na figura 11 tem-se a saída do comando *show mpls interfaces*, que mostra o *status* das interface e em quais o protocolo MPSL está habilitado.

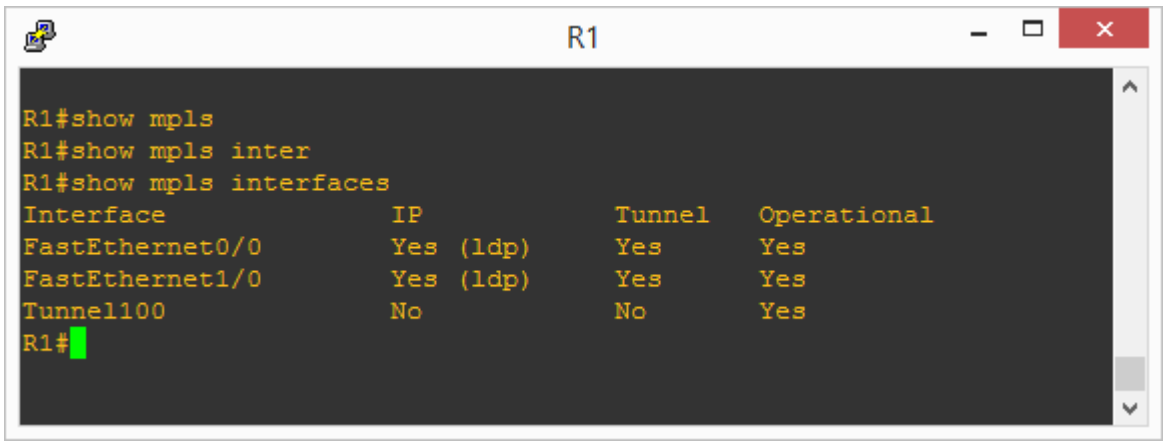


Figura 11: Saída do comando *show mpls interfaces*
Fonte: Autoria própria.

Na figura 12 é apresentado a saída do comando *show mpls ldp neighbor*, este comando mostra as adjacências LDP que foram formadas para o roteador 1, ele formou adjacência LDP com os roteadores R3 e R4.

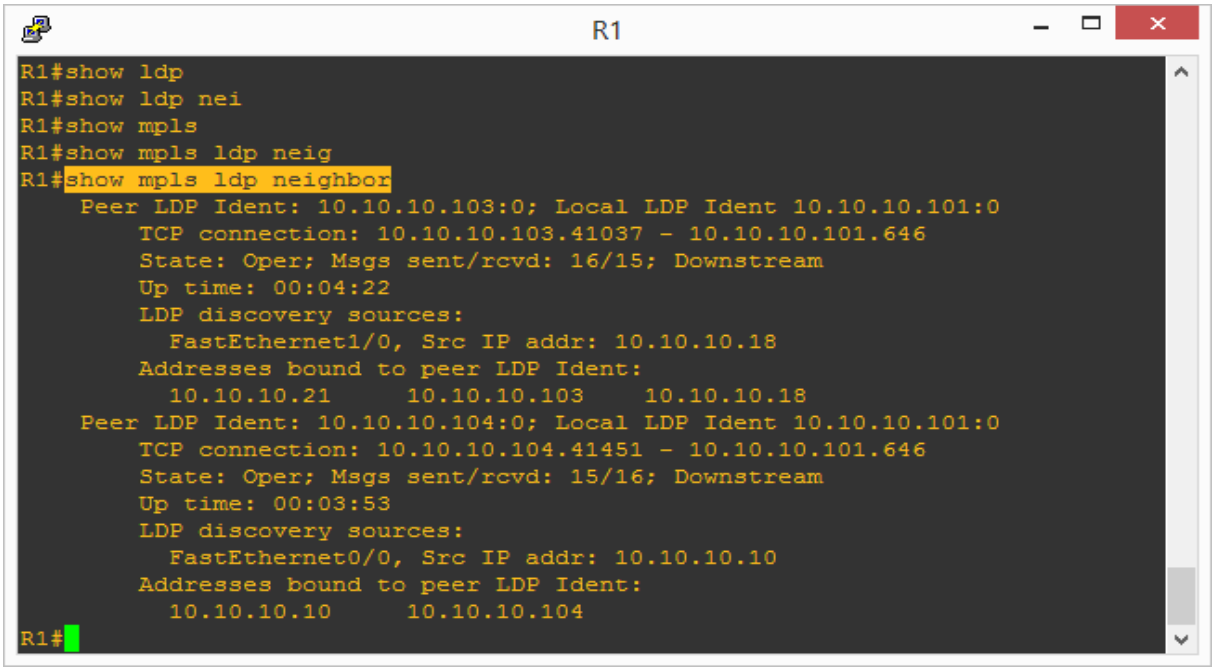


Figura 12: Saída do comando *show mpls ldp neighbor*
Fonte: Autoria própria.

Na figura 13 segue o exemplo da saída do comando *show ip cef* que representa a tabela FIB:

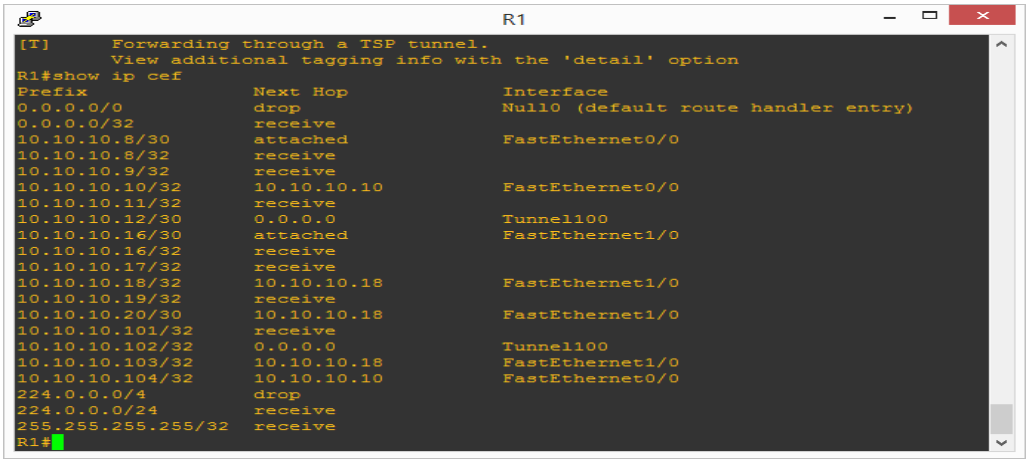


Figura 13: Saída do comando *show ip cef*
Fonte: Autoria própria.

Na figura 14 está representa a saída do comando *show mpls ldp bindings* que representa a tabela LIB:

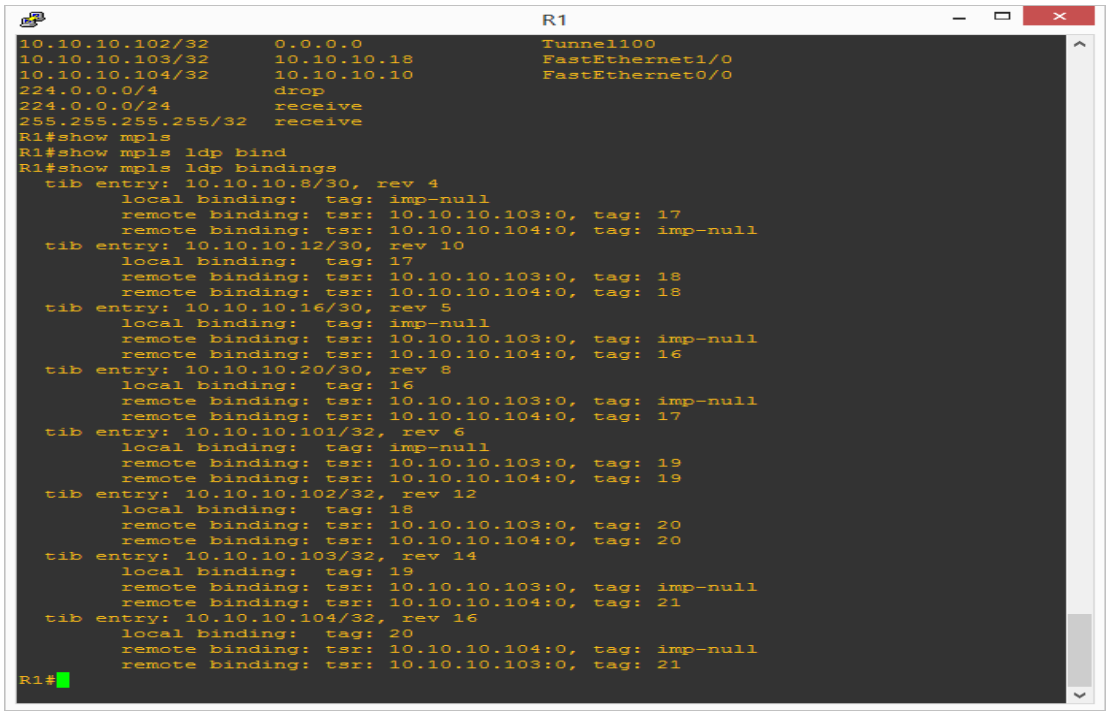


Figura 14: Saída do comando *show mpls ldp bindings*
Fonte: Autoria própria.

Na figura 15 tem-se a saída do como *show mpls forwarding-table* que representa a tabela LFIB:

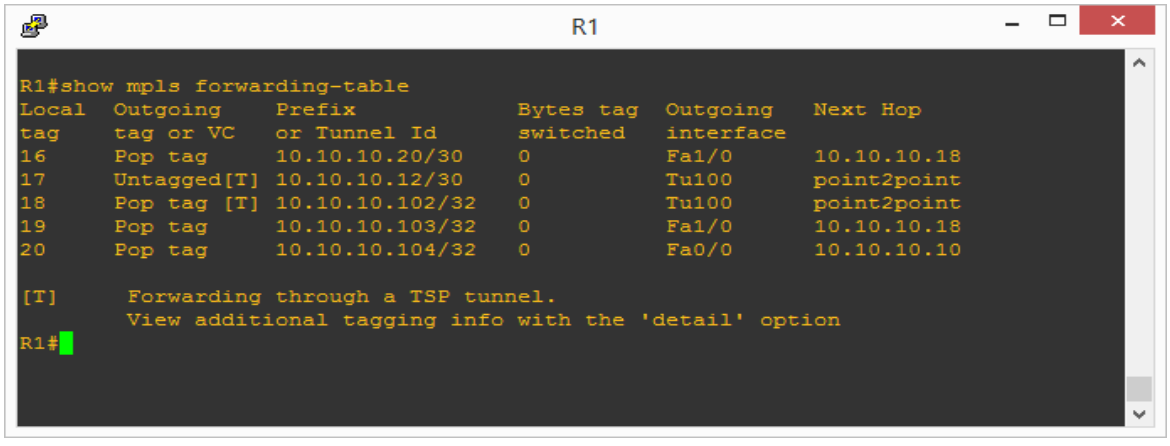


Figura 15: Saída do comando *show mpls forwarding-table*
Fonte: Autoria própria.

Na figura 16 é mostrada a saída do comando *show ospf neighbor* que mostra a adjacência OSPF formada entre os roteadores 1, 3 e 4.

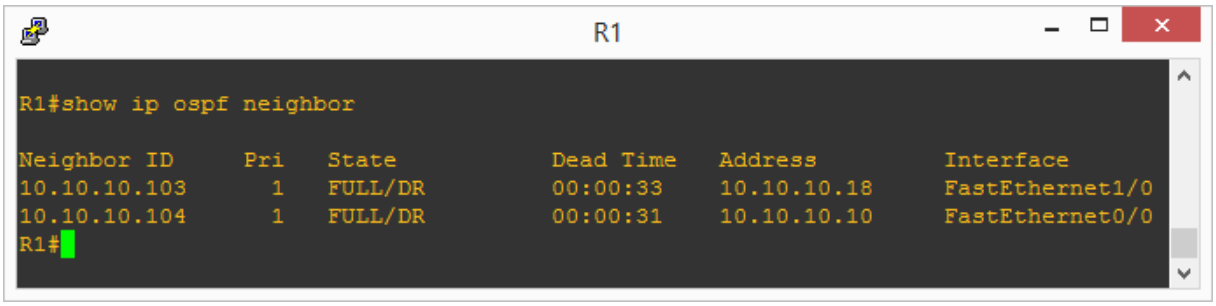


Figura 16: Saída do comando *show ospf neighbor*
Fonte: Autoria própria.

Na figura 17 é possível ver a saída do comando *show ip route*, esta saída mostra de que maneira as rotas estão sendo conhecidas pelos roteadores, como no lab foi utilizado OSPF como IGP, pode-se notar que os endereços de *loopback* foram aprendidos através do OSPF e também por qual interface ou túnel o tráfego deve ser encaminhado.

Figura 17: Saída do comando *show ip route*
Fonte: Autoria própria.

Até agora foram apenas demostradas as saídas dos comandos para comprovação de que a configuração do protocolo MPLS e do protocolo OSPF estão corretas, a seguir será demostrado o primeiro teste, os roteadores R1 e R2 não estão diretamente conectados, ou seja, para que o tráfego seja encaminhado entre esses dois roteadores, foi preciso estabelecer uma adjacência OSPF para que as *loopback* fossem anunciadas e então assim conhecidas pelos os roteadores e também foi configurado um túnel utilizando o MPLS-TE com *explicit path* orientando o tráfego a seguir um caminho pré-determinado pelo administrador. Na figura 18 é possível ver em destaque o *status* do túnel, qual o *path option* selecionado, neste caso é o *explicit path* que foi configurado com prioridade 10 e qual é o *explicit route* que foi configurado entre os roteadores através do comando *show mpls traffic-eng tunnels tunnel 100*:

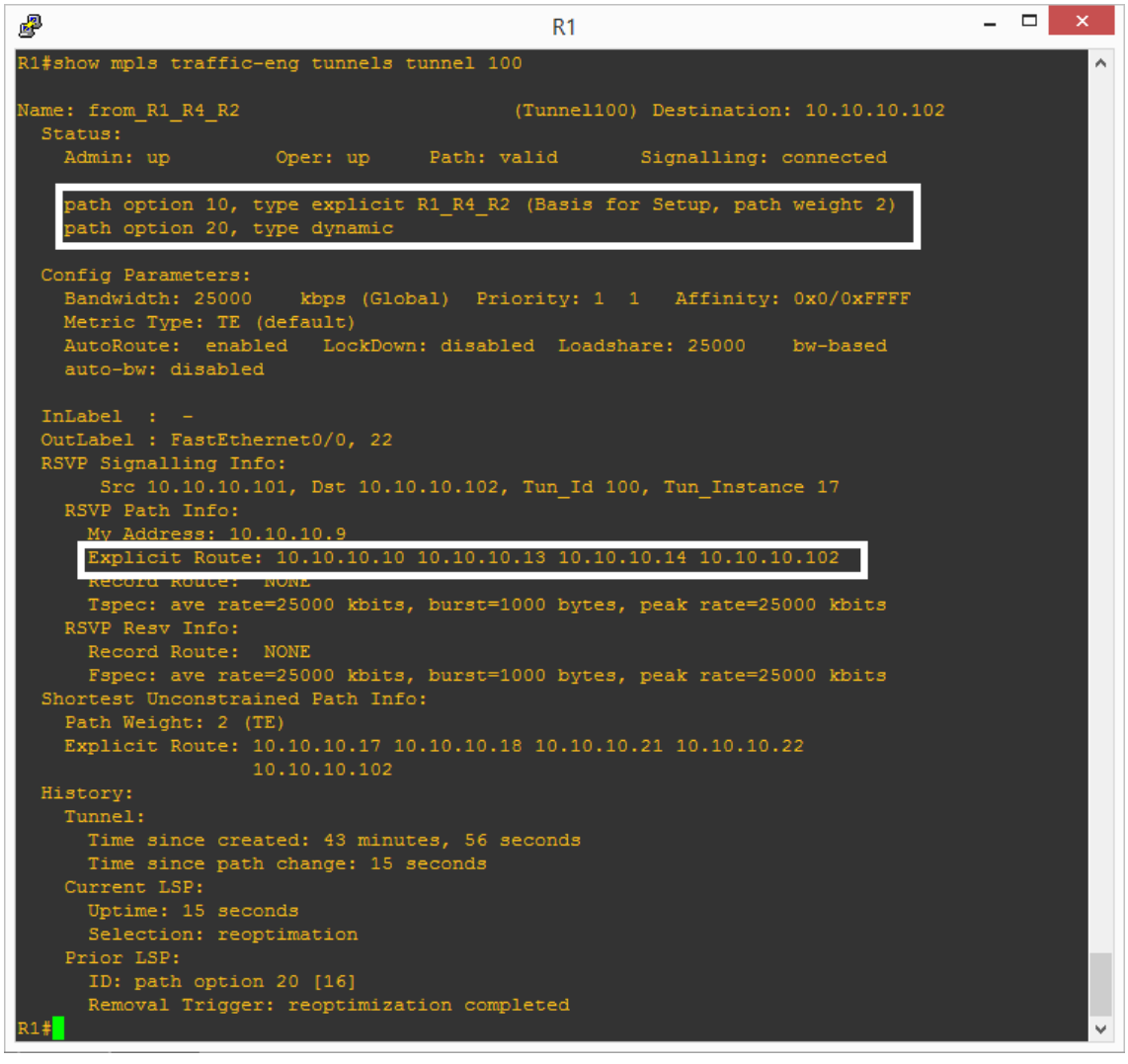


Figura 18: Saída do comando *show mpls traffic-eng tunnels tunnel 100*
Fonte: Autoria própria.

Para verificar se existe conectividade entre os roteadores R1 e R2 é preciso realizar um teste de *ping* para teste, porém para saber se o pacote está seguindo o caminho que foi configurado no túnel é preciso utilizado o comando *traceroute*, que traça a rota por onde o pacote passa até chegar ao destino. Na figura 19 tem-se a saída desse dois comados, é importante destacar que a teoria se comprava com estes dois comandos, pois o *ping* mostra que há conectividade entre os roteadores R1 e R2 mesmo não estando diretamente conectados e o caminho que o pacote percorre saindo de R1 em direção a R2 é o que previamente configurado, conforme abaixo:

```
ip explicit-path name R1_R4_R2 enable
 next-address 10.10.10.10
 next-address 10.10.10.14
next-address 10.10.10.102
```

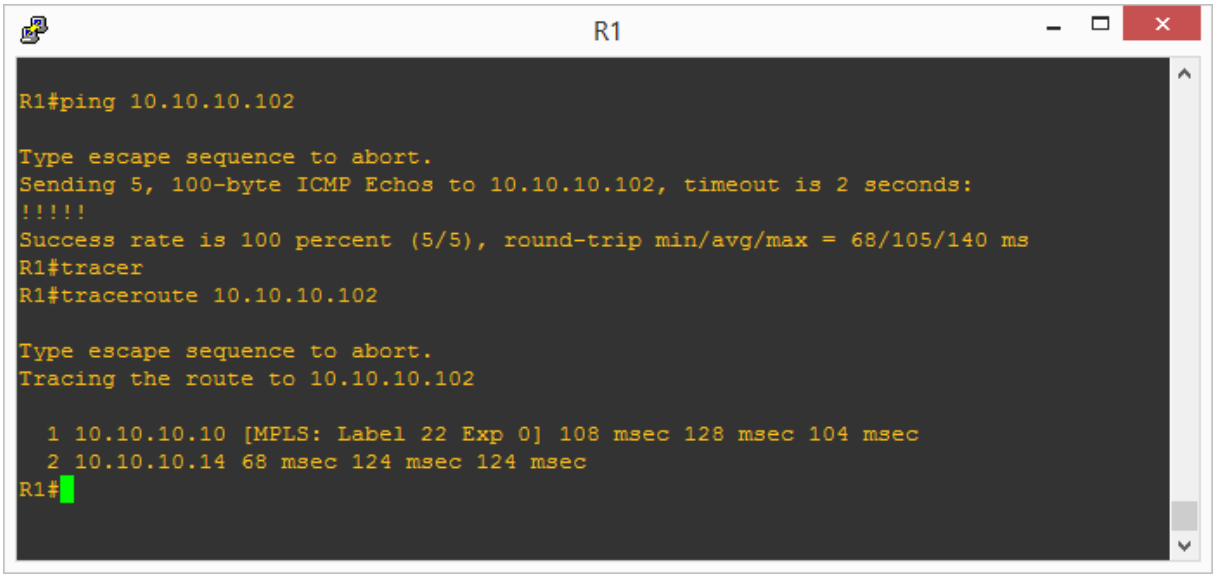


Figura 19: Teste de *ping* e *traceroute*
Fonte: Autoria própria.

Mas e agora, se ocorrer uma falha nesta malha MPLS, como o tráfego será encaminhado? Como já foi citado anteriormente, foi configurado o *explicit path* como sendo o caminho principal para que o tráfego entre R1 e R2 fosse encaminhado, porém também foi configurado um *dynamic path* com prioridade maior, justamente para em caso de falha o algoritmo possa calcular um caminho alternativo. Para comprovar esta parte da teoria, a seguir será mostrado os resultados desse teste,

existe um conexão entre o roteador R4 *fastEthernet 1/0* e o roteador R2 *fastEthernet 1/0* essa interfaces fazem parte do *explicit path*, então para simular uma falha, uma das interfaces (*fastEthernet 1/0*) entre os roteadores R4 e R2 foi configurada como administrativamente *down*, isso fez que com que o algoritmo recalcula-se o caminho por onde o tráfego deveria ser encaminhado entre os roteadores R1 e R2. Na figura 20 a interface *fastEthernet 1/0* do roteador R4 foi configurada com administrativamente *down*.

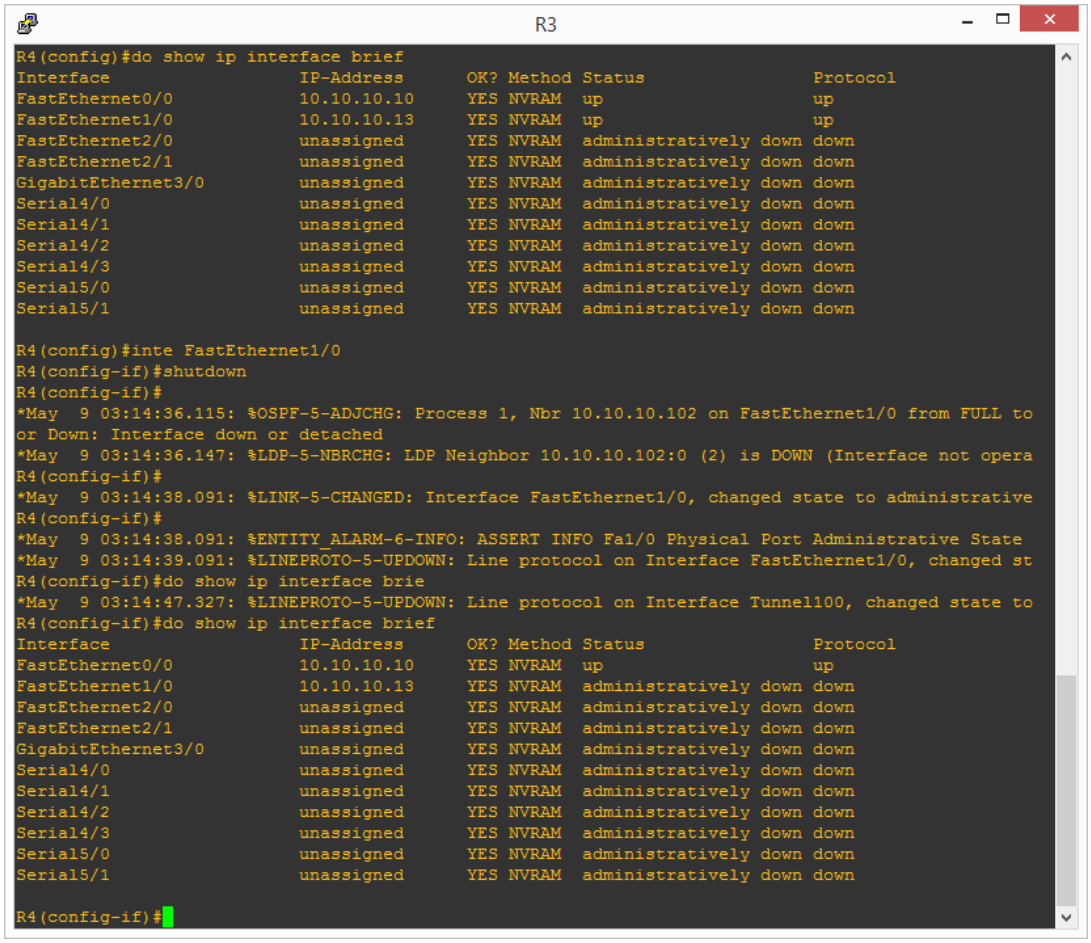


Figura 20: *Shutdown* na interface entre R4 e R2
Fonte: Autoria própria.

Na figura 21, nota-se que mesmo com a interface entre R4 e R2 com falha (administrativamente *down*) ainda existe conectividade entre R1 e R2, porém o caminho que o pacote está seguindo é outro, antes da falha o pacote seguia o caminho R1 > R4 > R2, agora passa a ser R1 > R3 > R2.

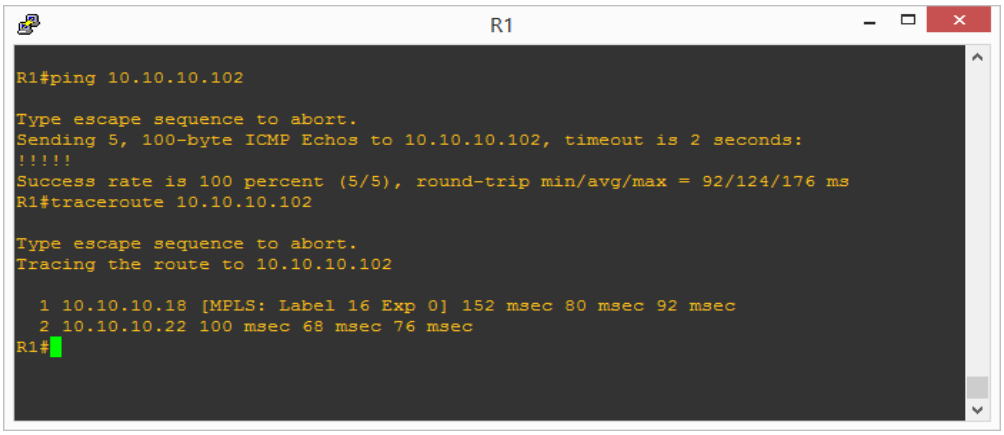


Figura 21: Teste de ping e *traceroute* após simular a falha
Fonte: Autoria própria.

Na figura 22, tem-se a comprovação de que o *dynamic path* que estava configurado como opção redundante assumiu seu papel após a falha, como o caminho principal está comprometido o *path option* agora está pelo *dynamic path*.

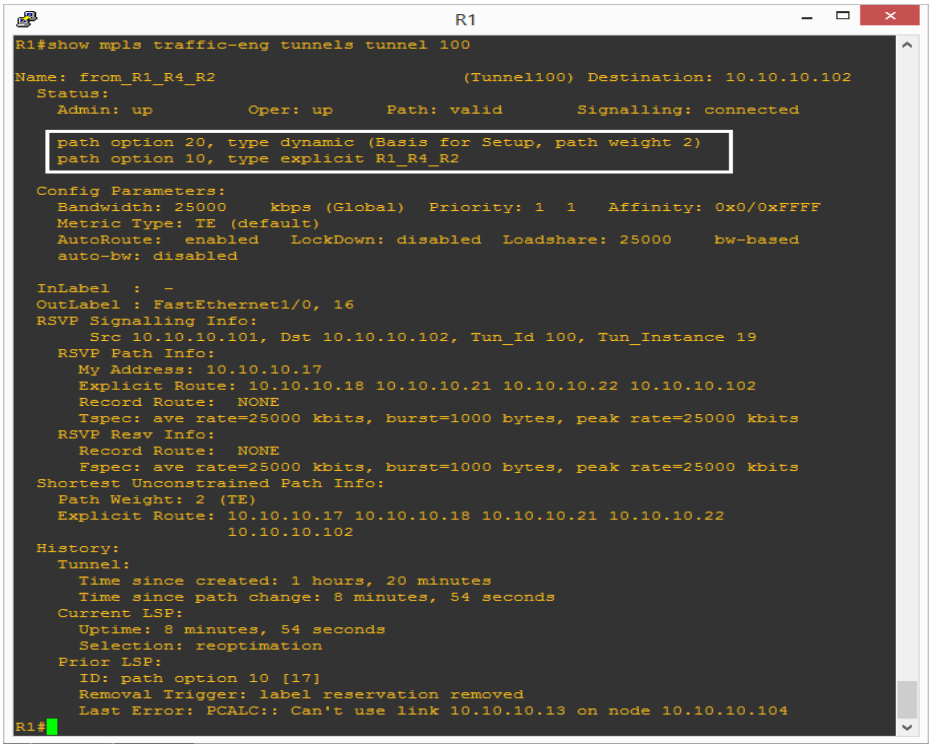


Figura 22: Redundância assumiu após a falha
Fonte: Autoria própria.

5 CONCLUSÃO

Este trabalho teve como finalidade apresentar as principais vantagens em se usar *Traffic Enginnering* com o protocolo MPLS em um *backbone* de um operadora. Quando se tem uma rede assimétrica a engenharia de tráfego é extremamente importante, pois é possível balancear e direcionar o tráfego por caminhos pelos quais o administrador pode determinar previamente sem levar em consideração as decisões realizados pelo algoritmo do IGP.

Com a engenharia de tráfego utilizando o protocolo MPLS, é possível se ter várias vantagens para se administrar o tráfego, onde uma dessas vantagens é garantir a qualidade de serviço para o usuário final, diminuindo o *delay* e a latência. Com o uso desse protocolo o usuário final sente uma melhora significante na performance se seu serviço.

Um dos pontos interessantes deste trabalho foi o paralelo feito entre o uso dos IGPs e dos *overlay models* para a engenharia de tráfego e quais são vantagens em se usar o protocolo MPLS para este fim. Através de uma prática de simulação utilizando o *software* GNS3 com equipamentos do fabricante Cisco foi possível demonstrar as características de utilização do MPLS-TE, usando o *explicit patch* para demonstrar como conduzir o tráfego de um ponto a outro por um caminho pré-determinado pelo administrador.

O MPLS-TE tem sido uma alternativa muito utilizada por grandes ISPs que ainda possuem redes assimétricas, nas quais precisam de um controle dos recursos de redes a fim de manipular o caminho por onde o tráfego deve seguir fim a fim para garantir os melhores serviços aos seus usuários finais. Este trabalho apresentou baseado na literatura teórica e na prática de que maneira isto pode executado em ambiente real.

REFERÊNCIAS

1 Portal IETF, RFC 2702. Disponível em: https://tools.ietf.org/html/rfc2702. Acesso em 01 de fevereiro de 2016

2 Portal IETF, RFC 3906. Disponível em: https://tools.ietf.org/html/rfc3906. Acesso em 10 de fevereiro de 2016

3 Portal Cisco, Cisco Networking Academy's Introduction to Routing Dynamically. Disponível em: http://www.ciscopress.com/articles/article.asp?p=2180210&seqNum=11. Acesso em 18 de fevereiro de 2016

4 Portal IETF, RFC 2328. Disponível em: https://www.ietf.org/rfc/rfc2328.txt. Acesso em 20 de fevereiro de 2016

5 Portal IETF, RFC 1195. Disponível em: https://www.ietf.org/rfc/rfc1195.txt. Acesso em 20 de fevereiro de 2016

6 Portal Cisco, Intermediate System-to-Intermediate System Protocol. Disponível em: http://www.cisco.com/en/US/products/ps6599/products_white_paper09186a00800a3e6f.shtml. Acesso em 30 de abril de 2016.

7 Portal Teleco, Frame Relay: http://www.teleco.com.br/tutoriais/tutorialfr/pagina_1.asp. Acesso em 20 de fevereiro de 2016

8 Portal Cisco, Comprehensive Guide to Configuring and Troubleshooting Frame Relay. Disponível em: http://www.cisco.com/c/en/us/support/docs/wan/frame-relay/16563-12.html. Acesso 01 de maio de 2016.

9 Portal Teleco, Asynchronous Transfer Mode (ATM). Disponível em: http://www.teleco.com.br/tutoriais/tutorialatm/pagina_1.asp. Acesso em 20 de fevereiro de 2016

10 Porta IETF, RFC 3031. Disponível em: https://www.ietf.org/rfc/rfc3031.txt. Acesso em 16 de fevereiro 2016

11 Portal Juniper, Routers in an LSP. Disponível em: http://www.juniper.net/techpubs/en_US/junos13.3/topics/concept/mpls-lsp-routers.html. Acesso 08 de maio de 2016.

12 Portal Juniper, MPLS Traffic Engineering Packet Path Control. Disponível em: http://www.juniper.net/documentation/en_US/junos15.1/topics/concept/mpls-traffic-engineering-packet-path-control.html. Acesso em 03 de fevereiro de 2016

13 Portal Cisco, MPLS Traffic Enginnering. Disponível em: http://www.cisco.com/c/en/us/td/docs/ios/12_0s/feature/guide/TE_1208S.html. Acesso em 08 de maio de 2016