UNIVERSIDADE FEDERAL DO CEARÁ

TRABALHO DE CONCLUSÃO DE CURSO

CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

FRANCISCO JEFFERSON MARQUES DE SOUSA

ANÁLISE COMPARATIVA ENTRE OS PROTOCOLOS MPLS E SEGMENT ROUTING: APLICAÇÕES DE ENGENHARIA DE TRÁFEGO EM NÚCLEOS DE REDES DE TELECOMUNICAÇÕES

SOBRAL

2025

FRANCISCO JEFFERSON MARQUES DE SOUSA

ANÁLISE COMPARATIVA ENTRE OS PROTOCOLOS MPLS E SEGMENT ROUTING: APLICAÇÕES DE ENGENHARIA DE TRÁFEGO EM NÚCLEOS DE REDES DE TELECOMUNICAÇÕES

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Computação da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Computação.
Orientador: Prof. Dr. Francisco Rafael Marques Lima

Coorientador: Prof. Dr. Diego Aguiar Sousa

SOBRAL

2025

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

M317a Marques de Sousa, Francisco Jefferson.
ANÁLISE COMPARATIVA ENTRE OS PROTOCOLOS MPLS E SEGMENT ROUTING: APLICAÇÕES DE ENGENHARIA DE TRÁFEGO EM NÚCLEOS DE REDES DE TELECOMUNICAÇÕES / Francisco Jefferson Marques de Sousa. – 2025.
92 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Sobral, Curso de Engenharia da Computação, Sobral, 2025.
Orientação: Prof. Dr. Francisco Rafael Marques Lima.
Coorientação: Prof. Dr. Diego Aguiar Sousa.

1. Engenharia de tráfego. 2. Políticas de redirecionamento. 3. Caminhos explicitos. 4. Comutação de rótulos. I. Título.

CDD 621.39

FRANCISCO JEFFERSON MARQUES DE SOUSA

ANÁLISE COMPARATIVA ENTRE OS PROTOCOLOS MPLS E SEGMENT ROUTING: APLICAÇÕES DE ENGENHARIA DE TRÁFEGO EM NÚCLEOS DE REDES DE TELECOMUNICAÇÕES

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Computação da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Computação.

Aprovada em: 11/03/2025

BANCA EXAMINADORA

Prof. Dr. Francisco Rafael Marques Lima (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Diego Aguiar Sousa (Coorientador)
Instituto Federal do Ceará (IFCE)

Prof. Dr. Victor Farias Monteiro
Universidade Federal do Ceará (UFC)

Esp. em redes Luiz Cosme Puppin Magalhães
Made4IT

Prof. Dr. Rodrigo Carvalho Souza Costa
Instituto Federal do Ceará (IFCE)

AGRADECIMENTOS

Ao Prof. Dr. Francisco Rafael Marques Lima, por me orientar em meu trabalho de conclusão de curso.

Ao Prof. Dr. Diego Aguiar Sousa, por ter desempenhado o papel de coorientador deste trabalho, fornecendo suporte técnico.

Ao professor e especialista em redes Luiz Cosme Puppin Magalhães, por ter colaborado de forma técnica para o desenvolvimento deste trabalho, esclarecendo dúvidas e apresentando propostas de melhoria.

À minha futura noiva, Ana Clara Barros Freitas, graduanda do curso de Engenharia Elétrica da Universidade Federal do Ceará, por me dar suporte na correção gramatical deste trabalho, bem como seu incentivo e apoio em todos os meus projetos pessoais e profissionais.

Ao mestre Carlos Augusto Melo de Pinho, do programa de mestrado em Engenharia elétrica e computação da Universidade Federal do Ceará, à época, que me incentivou e deu suporte no uso do *Latex*.

A todos os meus amigos e colegas de curso, por terem me ajudado nas disciplinas do curso, proporcionando-nos momentos de estudo em grupo e esclarecimento de dúvidas.

Aos meus familiares, que forneceram todo o suporte para minha permanência na cidade de Sobral e poder dar este passo em minha carreira profissional.

Ao coordenador Prof. Dr. Carlos Elmano de Alencar e Silva, por sempre se disponibilizar para ajudar-me em assuntos relacionados à coordenação e me aconselhar em assuntos profissionais.

A todos que foram meus professores no curso de Engenharia da Computação, incluindo professores do curso de Engenharia Elétrica, por me proporcionarem conhecimentos, além de terem feito parte desta importante etapa de minha vida.

À Universidade Federal do Ceará (UFC), em especial ao Campus de Sobral, pelo acolhimento, pelo respeito e por toda a experiência de aprendizado que tive durante estes 5 anos de graduação.

RESUMO

O presente trabalho tem como objetivo desenvolver um estudo acerca de métodos e aplicações de Engenharia de Tráfego, fazendo uso dos protocolos Comutação de Rótulos Multiprotocolo, ou do inglês, *Multiprotocol Label Switching* (MPLS) e Roteamento por Segmentos, ou do inglês, *Segment Routing* (SR). Para tanto, são abordados três estudos de caso, cujas arquiteturas são as mais próximas possíveis, em que buscam demonstrar as principais características dos referidos protocolos, bem como discutir suas diferenças, com a finalidade de propor a melhor solução para cenários de redes de comunicação. Neste aspecto, apropriou-se de ferramentas de virtualização, com o fito de construir a arquitetura das redes de comunicação, implementando serviços comumente ofertados pela indústria. Assim, são demonstradas implementações de manipulação de tráfego de dados com o protocolo MPLS, que depende de outros dois protocolos auxiliares: o Protocolo de Distribuição de Rótulos, ou do inglês, *Label Distribution Protocol* (LDP) e o Protocolo de Reserva de Recursos, ou do inglês, *Resource Reservation Protocol* (RSVP). Além disso, é necessário conceber caminhos explícitos pela topologia, de modo a ser possível perceber que o MPLS se trata de uma tecnologia que demanda altas taxas de dados de sinalização e possui implementação complexa para redes de larga escala. Desta forma, é exposto que o SR consiste em uma tecnologia mais adequada para redes baseadas em comutação de rótulos, visto ser um protocolo que é ativado a partir de extensões do protocolo de roteamento existente, o que implica em uma menor quantidade de dados de sinalização, como também, possui uma maior escalabilidade, uma vez que possibilita um redirecionamento de tráfego baseado em políticas.

Palavras-chave: Engenharia de tráfego. Políticas de redirecionamento. Caminhos explicitos. Comutação de rótulos.

ABSTRACT

This work aims to develop a study on methods and applications of Traffic Engineering, using the Multiprotocol Label Switching (MPLS) and Segment Routing (SR) protocols. To this end, three case studies are addressed, with architectures as similar as possible, aiming to demonstrate the main characteristics of these protocols, as well as to discuss their differences in order to propose the best solution for communication network scenarios. In this context, virtualization tools were employed to build the network architectures, implementing services commonly offered by the industry. Thus, implementations of data traffic manipulation using the MPLS protocol are presented, which relies on two auxiliary protocols: the Label Distribution Protocol (LDP) and the Resource Reservation Protocol (RSVP). Furthermore, it is necessary to define explicit paths within the topology, making it evident that MPLS is a technology that requires high signaling data rates and has a complex implementation for large-scale networks. Therefore, it is shown that SR is a more suitable technology for label-switching-based networks, as it is a protocol enabled through extensions of the existing routing protocol, which results in a lower amount of signaling data. Additionally, SR offers greater scalability by enabling policy-based traffic redirection.

Keywords: Traffic Engineering. Redirection Policies. Explicit Paths. Label Switching

LISTA DE ABREVIATURAS E SIGLAS

5G	*Fifth Generation*
6PE	*IPv6 Provider Edge over MPLS*
AS	*Autonomous System*
ATM	*Asynchronous Transfer Mode*
BE	*Best Effort*
BGP	*Border Gateway Protocol*
CE	*Customer Edge*
CLNP	*Connectionless Network Protocol*
ECMP	*Equal-Cost Multi-Path Routing*
FRR	*Fast Reroute*
iBGP	*Internal Border Gateway Protocol*
IGP	*Interior Gateway Protocol*
IoT	*Internet of Things*
IP	*Internet Protocol*
IS-IS	*Intermediate System to Intermediate System*
ISO	*International Organization for Standardization*
ISP	*Internet Service Provider*
ISPs	*Internet Service Providers*
LDP	*Label Distribution Protocol*
LS	*Link State*
LSP	*Label Switched Path*
MPLS	*Multiprotocol Label Switching*
MPLS-TE	*Multiprotocol Label Switching Traffic Engineering*
NET	*Network Entity Title*
OSI	*Open Systems Interconnection*
OSPF	*Open Shortest Path First*
P	*Provider*
PE	*Provider Edge*
QoS	*Quality of Service*
RIP	*Routing Information Protocol*
RSVP	*Resource Reservation Protocol*

SDN	*Software Defined Network*
SID	*Segment Identifier*
SL	*Segment List*
SLP	*Segment List Pointer*
SR	*Segment Routing*
SRGB	*Segment Routing Global Block*
TCP	*Transmission Control Protocol*
TE	*Traffic Engineering*
TI-LFA	*Topology Independent Loop-Free Alternate*
TTL	*Time To Live*
UDP	*User Datagram Protocol*
UFC	Universidade Federal do Ceará
VLSM	*Variable Length Subnet Mask*
VMs	*Virtual Machines*
VPC	*Virtual Personal Computer*
VPN	*Virtual Private Network*
VRF	*Virtual Routing and Forwarding*
WANs	*Wide Area Network*s

SUMÁRIO

1 INTRODUÇÃO

Ao passo que o desenvolvimento tecnológico exige maiores taxas de dados, tem-se a necessidade de desenvolver métodos mais eficientes de transmissão de pacotes pela rede, garantindo sua integridade, segurança e um melhor aproveitamento dos recursos espectrais do canal de transmissão (TROIA *et al.*, 2020). As redes baseadas em Protocolo de *Internet*, ou do inglês, *Internet Protocol* (IP) tradicionais oferecem pouca confiabilidade de serviço, o que é inaceitável para sistemas de telefonia ou aplicações de tempo real, além da alta complexidade de implementação e de inúmeras limitações que as tecnologias possuem, frente a um cenário tecnológico volátil e em constante expansão (PORWAL *et al.*, 2008).

Neste contexto, o conceito engenharia de tráfego é de suma importância para as operadoras de rede de dados e telefonia, além de estar em ascensão em Provedores de Serviço de Internet, ou do inglês, *Internet Service Providers* (ISPs). Neste sentido, um dos recursos possíveis através da engenharia de tráfego é a otimização do roteamento, que consiste no processo de determinar o caminho que o pacote deve seguir para ir de um nó ao outro, por meio da implementação de políticas que orientam o sentido do tráfego através da rede (NANDA, 2008).

Dessa forma, ao longo dos anos, foram desenvolvidas tecnologias para garantir melhor Qualidade de Serviço, ou do inglês, *Quality of Service* (QoS) e taxas de dados mais elevadas, bem como redução de custo com *hardware*. Em face disso, uma das soluções aplicadas a *backbones* de redes de computadores é a Comutação de Rótulos Multiprotocolo, ou do inglês, *Multiprotocol Label Switching* (MPLS) (MENDONÇA *et al.*, 2012). Este protocolo implementa o encaminhamento de pacotes baseado em rótulos, fazendo uso da extensão Comutação de Rótulos Multiprotocolo com Engenharia de Tráfego ou, *Multiprotocol Label Switching Traffic Engineering* (MPLS-TE) para engenharia de tráfego (SCHARF, 2017).

Apesar de ser uma tecnologia relativamente antiga, desenvolvida no final dos anos 90, o protocolo MPLS ainda é bastante utilizado, principalmente por parte de provedores de grande e médio porte, uma vez que estas empresas apresentam demandas de dados cada vez mais elevadas e necessitam implementar soluções de maior grau de confiabilidade até o usuário final (NANDA, 2008).

Contudo, a maior desvantagem do MPLS é o custo de processamento, uma vez que a aplicação de serviços como túneis de engenharia de tráfego ou mesmo implementação de QoS, necessitam de reservas de uma fração da banda do enlace que liga um nó ao outro, a fim de que os

dados sejam devidamente entregues ao usuário final. Neste aspecto, para que tais soluções sejam desenvolvidas, deve ser estabelecido um conjunto de políticas para controlar a conectividade e a qualidade de serviço de uma rede, o que leva a uma maior complexidade nas configurações dos equipamentos da rede e também um maior custo operacional (MENDONÇA *et al.*, 2012). Isso tem se tornado proibitivo no atual cenário tecnológico, em que os consumidores estão cada vez mais interessados em conteúdos de multimídia, que ocupam muita largura de banda, como videoconferências ou realidade aumentada, o que demanda maior reserva de recursos para construção dos túneis (CISCO, 2018).

Neste sentido, como uma proposta de solução à alta demanda operacional gerada pelo MPLS, tem-se a abordagem de Roteamento por Segmentos, ou do inglês, *Segment Routing* (SR). Conforme Hao *et al.* (2016), ele foi concebido para tornar possível o roteamento simplificado e flexível de conexões em redes baseadas em IP/MPLS, apropriando-se, em grande parte, nos recursos dos protocolos de rede existentes, uma vez que, a ideia principal do protocolo é utilizar uma sequência de segmentos para compor o caminho de conexão ponta a ponta desejado. Deste modo, o caminho entre os pontos finais de cada segmento é determinado por um protocolo de roteamento convencional como o protocolo Primeiro Caminho Aberto Mais Curto ou, do inglês, *Open Shortest Path First* (OSPF), ou do protocolo Sistema Intermediário para Sistema Intermediário ou, do inglês, *Intermediate System to Intermediate System* (IS-IS), que são protocolos de roteamento intra-área.

Nesta perspectiva, os rótulos dos segmentos são transportados no cabeçalho do pacote e, portanto, o estado por fluxo é mantido apenas no nó de ingresso. Desta maneira, um rótulo de segmento é como um rótulo MPLS e ações tradicionais *push, pop e swap*, que correspondem à inserção, troca e retirada do rótulo, respectivamente, podem ser aplicadas pelos roteadores no caminho do segmento (FILSFILS *et al.*, 2014). Com isso, segundo Hao *et al.* (2016), o roteamento por segmentos permite um controle mais preciso dos caminhos de roteamento e, assim, pode ser usado para distribuir o tráfego para melhor utilização da rede.

Neste ínterim, segundo Hao *et al.* (2016), as redes SR podem estender sua capacidade usando um controlador central que pode explorar todo o potencial do roteamento por segmentos, escolhendo-os com base na coleta de estatísticas correlacionadas ao padrão de tráfego, possibilitando assim, distribuir criteriosamente o tráfego na rede e evitar pontos de acesso locais. Portanto, este elemento de controle central pode ser feito por um elemento de cálculo de caminho ou, no caso de uma Rede Definida por Software, ou do inglês, *Software Defined Network* (SDN), através

de um controlador central (FUNDATION, 2012). Desta forma, uma rede roteada por segmentos controlada por SDN pode combinar a eficiência do controle centralizado com a resposta rápida e escalável a falhas.

Contudo, devido a limitações de recursos computacionais e por se distanciar do escopo deste trabalho, a integração do SR com o SDN não será abordada. Em vez disso, a discussão será focada nos protocolos MPLS e SR, explorando suas aplicações na engenharia de tráfego em redes de comunicação, com o fito de demonstrar a superioridade do SR em termos de flexibilidade e escalabilidade. Para tal, serão apresentados conceitos fundamentais da engenharia de tráfego, destacando sua importância na garantia de QoS para o usuário final em redes de um Provedor de Serviços de Internet, ou do inglês, *Internet Service Provider* (ISP).

2 OBJETIVOS

2.1 Objetivos Gerais

Este trabalho tem como objetivo levantar uma série de estudos direcionados a soluções de Engenharia de Tráfego ou, *Traffic Engineering* (TE), a partir da literatura vigente de cada solução, apresentada previamente no texto de introdução, como também implementar três cenários, onde os dois primeiros possuem a mesma topologia física, mas soluções lógicas distintas, nos quais, em um dos cenários, aplica-se o tradicional protocolo MPLS fazendo uso de sua extensão MPLS-TE e, para o outro cenário, aplica-se a solução MPLS-TE com SR, com o intuito de comparar os dois cenários e determinar qual possui o melhor desempenho. Já para o terceiro, se estabelece o objetivo de implementar políticas de redirecionamento de tráfego, baseadas em SR.

2.2 Objetivos Específicos

- Desenvolver um estudo sobre o estado da arte das soluções MPLS e SR aplicadas a *backbones* de redes de operadoras de *Internet*;
- Desenvolver soluções de Engenharia de Tráfego usando MPLS e SR;
- Produzir um estudo comparativo entre as soluções tomando como base a literatura vigente disponível em âmbito acadêmico e na indústria;
- Demonstrar a superioridade da tecnologia SR para cenários de ISP.

3 JUSTIFICATIVA

As Redes de Área Ampla ou do inglês, *Wide Area Network*s (WANs), tradicionais não foram constituídas para atender às necessidades das empresas modernas de hoje, que demandam cada vez mais altas taxas de dados para os diferentes serviços, tais como videoconferência ou *streaming* de vídeo, que exigem muito recurso de banda. Desta forma, mesmo com a adoção do MPLS pelos ISPs, que visava a solucionar as limitações das arquiteturas tradicionais, essa tecnologia ainda enfrenta desafios, principalmente devido ao alto custo operacional necessário para garantir QoS no fluxo de dados (TROIA *et al.*, 2020).

Vale salientar que, com a proliferação de dipositivos conectados à *Internet*, que variam de *smartphones* a dispositivos de *Internet* das Coisas ou, do inglês, *Internet of Things* (IoT), as redes enfrentam novos desafios, como acomodar e gerenciar um volume cada vez maior de tráfego de dados e serviços. Desta forma, segundo Mon e Mon (2018), esse aumento na demanda por conectividade tem incrementado a complexidade das redes e a necessidade de soluções que sejam não apenas escaláveis, mas também eficientes em termos de utilização de recursos. Em função disso, o SR surge para simplificar o uso do MPLS, eliminando a necessidade de Protocolos de Distribuição de Rótulos, ou do inglês, *Label Distribution Protocol* (LDP), e do Protocolo de Reserva de Recurso, ou do inglês, *Resource Reservation Protocol* (RSVP), e utiliza apenas o Protocolo de *Gateway* Interior, ou do inglês, *Interior Gateway Protocol* (IGP), que pode ser qualquer protocolo de Estado de *Link*, ou do inglês, *Link State* (LS), que troca rotas internas à rede, para atribuir e propagar os rótulos.

Neste contexto, o SR utiliza um protocolo IGP para distribuir as marcações e computar os caminhos, sem a necessidade de alterar a arquitetura MPLS já existente, permitindo sua coexistência com o MPLS (FILSFILS *et al.*, 2018). Além disso, conforme destacado no trabalho de Ginsberg *et al.* (2018), a arquitetura SR pode ser implementada em conjunto tanto com o MPLS, denominada SR-MPLS, quanto com o IPv6, conhecido como SRv6.

Adicionalmente, segundo Ventre *et al.* (2020), uma motivação vantajosa importante do SR, consiste na redução drástica do estado por fluxo que precisa ser mantido nos nós da rede para apoiar caminhos de TE. Deste modo, com o SR não é necessário configurar tabelas de encaminhamento nos nós ao longo do caminho, pois apenas o nó de ingresso armazenará a associação entre um fluxo e seu caminho a ser aplicado, ou seja, na Lista de Segmento, ou do inglês, *Segment List* (SL). Isso também significa que nenhuma mensagem de configuração, por exemplo, ao usar o protocolo RSVP, precisa ser enviada a todos os nós ao longo do caminho

para estabelecer um túnel. Outrossim, uma segunda importante motivação é usar segmentos em vez de “fixar” toda a sequência de saltos como no MPLS-TE tradicional, o que torna possível explorar vários caminhos do Roteamento de Múltiplos Caminhos de Custo Igual, ou do inglês, *Equal-Cost Multi-Path Routing* (ECMP), dentro de cada segmento.

Neste ínterim, segundo Ventre *et al.* (2020), o SR tem se popularizado entre as operadoras, permitindo-lhes implementar suas aplicações em diferentes tipos de redes, como *backbones* de transporte, redes de acesso, *data centers* e redes de telecomunicação celular sem fio de Quinta Geração ou do inglês, *Fifth Generation* (5G). Em redes implementadas em IPv4, o plano de dados MPLS (SR-MPLS) depende da estrutura da tecnologia MPLS estabelecida. Neste aspecto, SR-MPLS pode ser visto como uma melhoria e simplificação do plano de controle MPLS tradicional, sendo, portanto, benéfico para ISPs que fornecem serviços que funcionam sobre MPLS. Além disso, o plano de dados baseado em IPv6 com SRv6 está ganhando força, pois oferece a possibilidade de combinar serviços e recursos de rede sobrepostos e subjacentes usando apenas a tecnologia IPv6 e um protocolo IGP. Neste sentido, o modelo de programação de rede SRv6 oferece flexibilidade sem precedentes no projeto, operando serviços de rede, o que faz do SRv6 uma escolha atraente para operadoras que estão implantando novas redes ou planejando a evolução de suas arquiteturas.

4 FUNDAMENTAÇÃO TEÓRICA

4.1 Roteamento IP

Segundo Forouzan e Fegan (2009), o roteamento IP é responsavel por determinar o caminho pelo qual os pacotes de dados devem seguir ao serem enviados de uma origem para um destino em uma rede de computadores. Assim, ele é parte fundamental da comunicação em redes baseadas no protocolo IP, como a *Internet*.

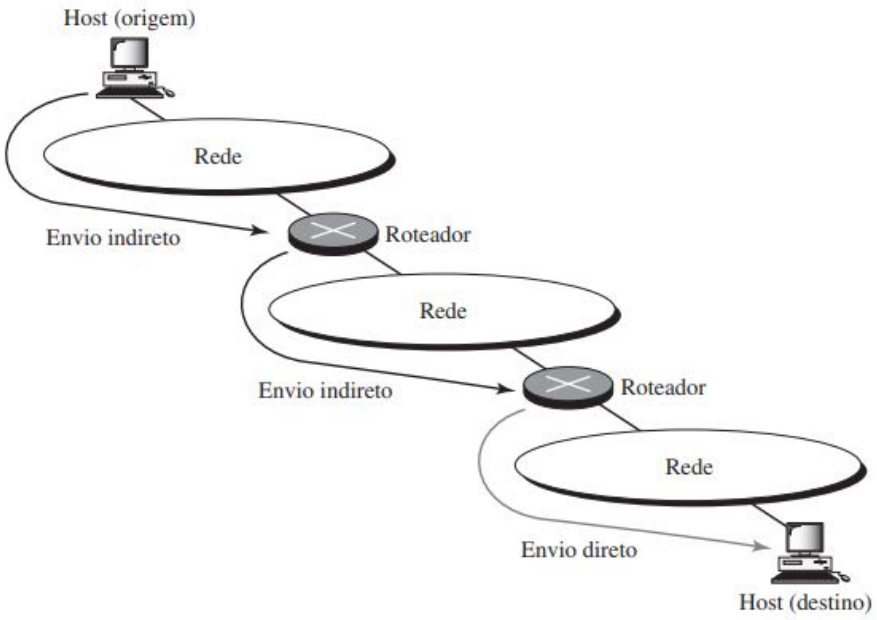
Neste sentido, quando um dispositivo em uma rede deseja enviar dados para outro em uma rede diferente, ele envia pacotes de dados que contêm o endereço IP de destino. Os roteadores, que são dispositivos de rede projetados para encaminhar pacotes, desempenham um papel crucial no processo de roteamento. Eles examinam o endereço IP de destino de cada pacote e tomam decisões com base em tabelas de roteamento para determinar para onde encaminhar o pacote (FOROUZAN; FEGAN, 2009).

Na Figura 1 sendo apresentado um sistema com três redes interligadas por dois roteadores, além de dois *hosts* conectados a redes distintas, sendo cabível ressaltar que os enlaces presentes entre os *hosts* e roteadores encontram-se também em redes distintas e, portanto, um algoritmo de roteamento deve ser implementado. Neste caso, o *Host* (origem) envia pacotes para o *Host* (destino) e, por estarem em redes diferentes, necessitam da intermediação dos roteadores, uma vez que o roteamento ocorre salto a salto. Em função disso, o *Host* (origem) faz o envio de forma indireta para o roteador mais próximo. Um outro ponto, é que os pacotes têm em seu cabeçalho o endereço de destino e, caso o prefixo de interesse não pertença ao equipamento que está processando o pacote, ele transmite os dados para o próximo salto, até o equipamento que detém o endereço de destino, que para esta situação, é o *HOST* (destino).

Atualmente, uma *internet* pode ser tão grande que um único protocolo de roteamento não consegue lidar com a tarefa de atualizar as tabelas de roteamento de todos os roteadores. Por isso, uma *internet* é dividida em vários Sistemas Autônomos ou, do inglês, *Autonomous System* (AS). Em face disso, um AS é um grupo de redes e roteadores sob a autoridade de uma única administração, de maneira que o roteamento dentro de um AS é referido como roteamento intradomínio. Por outro lado, o roteamento entre ASs é referido como roteamento entre domínios (FOROUZAN; FEGAN, 2009). Na Figura 2 está sendo apresentado um cenário de roteamento entre ASs.

Na Figura 2 tem-se os roteadores R1, R2, R3 e R4, ditos como roteadores de borda,

Figura 1 – Cenário de uma rede roteada

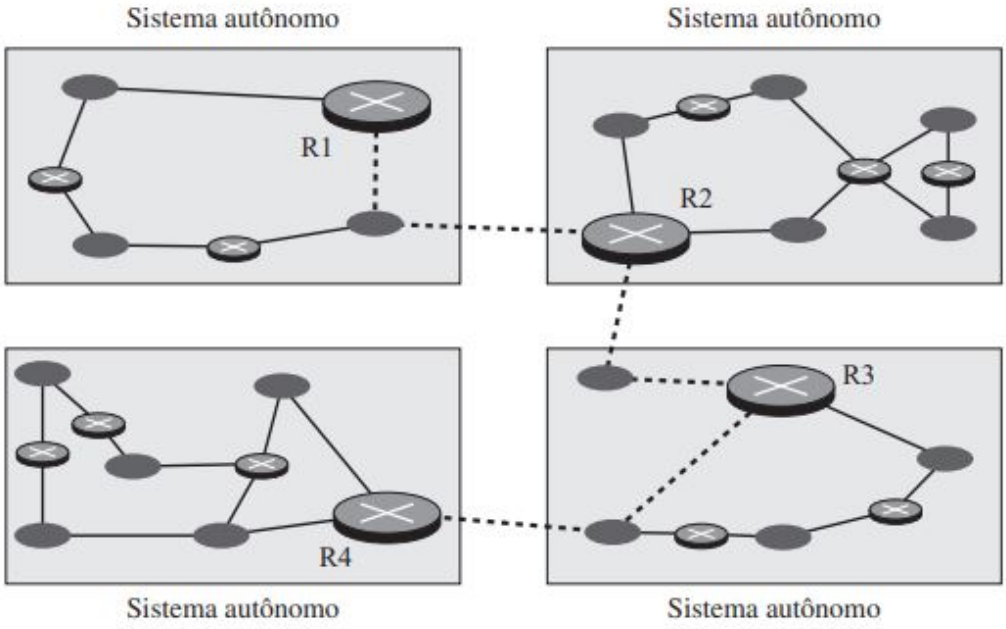


Fonte: (FOROUZAN; FEGAN, 2009).

ou seja, os dispositivos da rede que possuem enlaces diretos com a *Internet*, os quais estabelecem as seções usando Protocolo de *Gateway* de Borda ou, do inglês, *Border Gateway Protocol* (BGP), entre os ASs (FOROUZAN; FEGAN, 2009).

Com base nisso, cada AS pode escolher um ou mais protocolos de roteamento intradomínio para tratar o roteamento dentro do sistema autônomo. Entretanto, somente um

Figura 2 – Roteamento entre ASs



Fonte: (FOROUZAN; FEGAN, 2009).

protocolo de roteamento interdomínio manipula o roteamento entre ASs, que é o caso do BGP, o qual será apresentando posteriormente (FOROUZAN; FEGAN, 2009).

Existem vários protocolos e algoritmos de roteamento IP que auxiliam os roteadores a tomarem decisões eficazes sobre como encaminhar pacotes. Alguns dos protocolos de roteamento IP mais comuns incluem o Protocolo de Informação de Roteamento ou, do inglês, *Routing Information Protocol* (RIP), o OSPF, o IS-IS e o BGP, dentre outros. O RIP, o OSPF e o IS-IS se classificam na categoria de roteamento intradomínio, ou seja, internamente à rede da empresa, enquanto o BGP está dentro da classe de roteamento entre domínios, trabalhando com a divulgação de rotas para a *Internet*. Cada um desses protocolos têm suas próprias regras e características, adequadas a diferentes tipos de redes e cenários de roteamento (FOROUZAN; FEGAN, 2009).

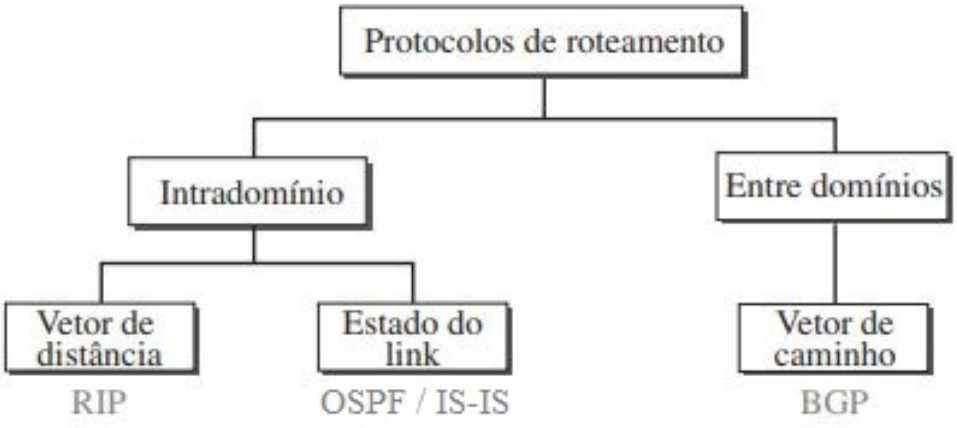
No que se refere ao protocolo RIP, ele utiliza o conceito de vetor de distância. Neste caso, a rota de menor custo entre dois nós quaisquer é aquela com o menor número de saltos. Nesse conceito, cada nó mantém um vetor (tabela) de distância mínima até cada nó, bem como a tabela em cada nó também guia os pacotes até o nó desejado, mostrando a próxima parada na rota (roteamento do próximo *hop*) (FOROUZAN; FEGAN, 2009).

Já os protocolos OSPF e IS-IS implementam o conceito de estado do *link*, que tem uma filosofia diferente do roteamento por vetor de distância. Nele, cada nó no domínio tem conhecimento da topologia inteira, com uma lista de nós e enlaces, bem como da forma que eles estão conectados, incluindo o tipo, custo (métrica) e a condição dos enlaces (ativos ou desativados). Desta maneira, o nó usa o algoritmo de Dijkstra para construir uma tabela de roteamento (FOROUZAN; FEGAN, 2009), que consiste em um dos algoritmos que calcula o caminho de custo mínimo entre vértices de um grafo (CARVALHO, 2008).

Por fim, o BGP opera com o conceito de vetor de caminho, que é semelhante ao do roteamento por vetor de distância. Nele, supõe-se que existe um nó em cada sistema autônomo que atua em nome do sistema autônomo inteiro, em que ele é chamado de nó orador. Nesta perspectiva, o nó orador em um AS cria uma tabela de roteamento e anuncia para os nós oradores nos ASs vizinhos. Com efeito, a ideia é a mesma do roteamento por vetor de distância, exceto que apenas nós oradores em cada AS podem se comunicar. Entretanto, o que é anunciado é diferente, visto que o nó orador anuncia a rota (e não a métrica dos nós, como é o caso do roteamento intradomínio) em seu sistema autônomo ou em outros AS (FOROUZAN; FEGAN, 2009). Na Figura 3 está sendo apresentado um breve resumo acerca dos protocolos mencionados

anteriomente, bem como suas respectivas classificações.

Figura 3 – Resumo acerca dos protocolos de roteamento



Fonte: (FOROUZAN; FEGAN, 2009).

4.1.1 Routing Information Protocol (RIP)

O RIP é um protocolo de roteamento de vetor de distância que utiliza os saltos (*hops*) como o parâmetro de custo, ou seja, cada enlace na rede possui um custo igual a 1. Desse modo, os saltos representam o número de subredes que o dado precisa percorrer ao sair de um roteador de origem para o destino (KUROSE; ROSS, 2006).

Contudo, segundo Guimarães *et al.* (2021), o algoritmo do protocolo RIP possui um limite máximo de 15 saltos por rede, ou seja, em um AS muito complexo com mais de 15 saltos em sua rede, o RIP não poderá ser utilizado. Ademais, neste protocolo os roteadores atualizam suas tabelas de roteamento a cada 30 segundos, compartilhando seus vetores de distância com seus vizinhos. À vista disso, devido às limitações citadas anteriormente, propõem-se outros protocolos intradomínios, como OSPF e IS-IS, uma vez que estes protocolos são comumente usados nos dias atuais, por empresas de telecomunicações (BANUPRIYA *et al.*, 2022).

4.1.2 Open Shortest Path First (OSPF)

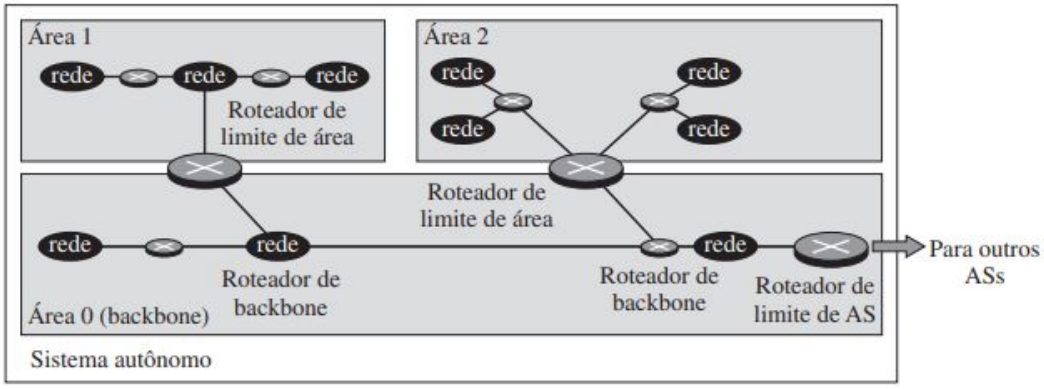
O OSPF implementa o algoritmo de estado de *link* e, nos dias de hoje, é o mais popular entre os protocolos IGP. Seu surgimento deve-se principalmente às limitações dos outros protocolos intradomínios, como é o caso do RIP. Além disso, é chamado de OSPF, pois utiliza o algoritmo *Shortest Path First* para o cálculo dos melhores caminhos, que também é conhecido

como algoritmo de Dijkstra (NEVES; TORRES, 2017).

Nesse sentido, apesar da complexidade do OSPF devido ao uso do algoritmo de estado de *link*, ele se mostra altamente robusto na detecção de falhas em enlaces. Assim, o algoritmo caracteriza o estado de um enlace como a descrição da respectiva interface e sua relação com os seus roteadores vizinhos. Desta forma, essa descrição inclui parâmetros como, o endereço IP da interface, a máscara de rede, o tipo de rede a que está ligada, os roteadores ligados a essa rede, etc. Vale ressaltar que este conjunto de informações constitui a base de dados do algoritmo de estado de *link* e permite a autenticação das mensagens trocadas entre roteadores (NEVES; TORRES, 2017).

Por fim, segundo Neves e Torres (2017), o protocolo OSPF converge em uma proporção logarítmica ao número de enlaces, enquanto o RIP converge proporcionalmente ao número de nós da rede. Isto torna a convergência do OSPF muito mais rápida. Além disso, no protocolo RIP, o tamanho da mensagem aumenta conforme o número de destinos na rede. Assim, em redes grandes, cada mensagem precisa ser dividida em múltiplos pacotes, o que resulta em uma redução adicional na velocidade de convergência. Por estas e outras razões, o OSPF é uma melhor escolha, tratando-se de protocolos IGP. Posto isto, na Figura 4 está sendo apresentado um cenário de uma rede IP, cujo protocolo IGP usado é o OSPF. Na figura, é possível observar a existência de divisões da rede em áreas, que é um típico método usado no OSPF.

Figura 4 – Implementação de OSPF em um AS



Fonte: (FOROUZAN; FEGAN, 2009).

4.1.3 Intermediate System-to-Intermediate System (IS-IS)

De acordo com Mendonça *et al.* (2012), o protocolo IS-IS foi desenvolvido pela Organização Internacional de Normalização ou, do inglês, *International Organization for Stan-*

dardization (ISO), e, por isso, ele pode ser diretamente relacionado ao modelo Interconexão de Sistemas Abertos ou, do inglês, *Open Systems Interconnection* (OSI). Contudo, de acordo com Martey (2002), o protocolo IS-IS tem uma adoção mais ampla na infraestrutura da *Internet* do que na arquitetura OSI, onde originalmente foi projetado para ser utilizado. Nesse contexto, trata-se de um protocolo que faz roteamento por estado do *link*, tendo suporte ao balanceamento de carga e à Máscara de Subrede de Comprimento Variável ou, do inglês, *Variable Length Subnet Mask* (VLSM), que se refere à capacidade de utilizar máscaras de subrede de tamanhos diferentes em uma mesma rede, sendo um protocolo para atuação intradomínio.

Neste ínterim, segundo Mendonça *et al.* (2012), embora o protocolo IS-IS tenha sido originalmente desenvolvido pela ISO para ser usado no roteamento de redes baseado em Protocolo de Rede sem Conexão ou, do inglês, *Connectionless Network Protocol* (CLNP), como parte da arquitetura OSI, com o tempo, ele foi adaptado para suportar o roteamento em redes baseadas em IP. Essa adaptação ficou conhecida como *Integrated* IS-IS, permitindo que o protocolo funcione tanto em redes CLNP quanto em redes IP de forma integrada. Atualmente, o termo IS-IS é amplamente utilizado para referir-se ao uso do *Integrated* IS-IS exclusivamente em redes IP.

Dessa forma, é possível destacar diversas semelhanças entre os protocolos IS-IS e OSPF. Primeiramente, ambos são protocolos de roteamento que utilizam o algoritmo de estado de *link*, suportam VLSM e adotam uma estrutura hierárquica, organizando a rede em áreas para otimizar o roteamento. Além disso, definem diferentes tipos de roteadores para funções específicas, como roteadores de núcleo e roteadores de borda. Isso possibilita a redução de processamento nos roteadores e melhora a escalabilidade da arquitetura. Por fim, ambos os protocolos dão suporte ao balanceamento de carga e possuem capacidade de autenticação (MENDONÇA *et al.*, 2012).

Contudo, é importante destacar algumas diferenças entre os dois protocolos, que consistem na forma como ambos manipulam pacotes de "*Hello*", utilizados para formação das adjacências entre os vizinhos, tendo em vista que, no OSPF, apenas um tipo de "*Hello*" é definido e, em redes IS-IS, os roteadores são capazes de enviar dois tipos distintos de pacotes "*Hello*": Nível 1 e Nível 2, onde dos dados de Nivel 1 indicam comunicação em uma mesma área e os de Nivel 2 indicam comunicação entre áreas distintas. Outra diferença, é com relação aos tipos de roteadores que são utilizados por esses protocolos, pois, apenas por possuírem nomenclaturas diferentes, pode haver o mapeamento entre eles, devido às suas funcionalidade (DOYLE, 2016).

4.1.4 Border Gateway Protocol (BGP)

O BGP é um protocolo de roteamento entre domínios que usa roteamento por vetor de caminho. Ele foi concebido em 1989 e passou por quatro versões. Como apresentado em seções passadas, a *Internet* é dividida em domínios hierárquicos, chamados de AS. Por exemplo, uma grande empresa que gerencia sua própria rede e tem controle total sobre ela é um AS. Outro exemplo, um ISP local que fornece serviços para clientes locais é um AS (FOROUZAN; FEGAN, 2009).

Em uma abordagem direta, a troca de informações de roteamento entre dois roteadores usando BGP ocorre em uma sessão. Uma sessão é uma conexão estabelecida entre dois roteadores de borda, somente com o propósito de trocar informações de roteamento. Nesse sentido, para criar um ambiente confiável, o BGP usa os serviços do Protocolo de Controle de Transmissão, ou do inglês, *Transmission Control Protocol* (TCP). Em outras palavras, uma sessão no nível do BGP, como um programa aplicativo, é uma conexão no nível do TCP. Entretanto, há uma diferença sutil entre uma conexão no TCP feita para o BGP e outros programas aplicativos. Quando uma conexão TCP é criada para o BGP, ela pode durar um longo tempo até que algo incomum aconteça. Por isso, às vezes, as sessões BGP são referidas como conexões semipermanentes (FOROUZAN; FEGAN, 2009).

4.1.4.1 Atributo de Comunidades BGP

A *RFC 1997*, definida por Li *et al.* (1996), introduz o atributo de comunidades BGP, que consiste em um mecanismo para marcar e agrupar rotas com base em políticas de roteamento. Neste aspecto, o recurso permite a implementação de regras específicas a grupos de prefixos, facilitando assim, a implementação de políticas de trânsito.

No entanto, a versão original do atributo de comunidades BGP apresentava algumas limitações, como um espaço de valores relativamente pequeno e a ausência de uma estrutura que permitisse diferenciar comunidades de diferentes aplicações. Assim, para superar essas restrições, a *RFC 4360* introduziu as Comunidades Estendidas, ou do inglês, *Extended Communities* do BGP, que expandem as capacidades das comunidades padrão, como apresentado em Tappan *et al.* (2006).

As principais melhorias do BGP *Extended Communities Attribute* são:

- Intervalo estendido, permitindo que comunidades possam ser atribuídas para uma ampla

variedade de usos, sem risco de sobreposição.

- Adição de um campo de Tipo, que fornece uma estrutura para organizar diferentes categorias de comunidades.

Essa estrutura adicional possibilita a criação de políticas mais refinadas. Por exemplo, um roteador pode filtrar todas as comunidades de um determinado tipo ou permitir apenas certos valores dentro de um tipo específico. Além disso, as *Extended Communities* permitem indicar se uma determinada comunidade deve ser propagada além dos limites de um AS, algo que, sem essa estrutura, exigiria a enumeração explícita de todas as comunidades a serem aceitas ou negadas pelos dispositivos que estabelecem comunicação via protocolo BGP em ASes vizinhos.

4.2 Multiprotocol Label Switching (MPLS)

O MPLS é uma tecnologia aberta que foi apresentada inicialmente como uma solução que possibilitava melhorar o desempenho das redes IP na função de encaminhamento de pacotes, combinando o processo de roteamento de nível 3 com a comutação de nível 2, para realizar o encaminhamento de datagramas através de pequenos rótulos de tamanho fixo. Tais rótulos são números utilizados no protocolo MPLS e, através deles, a decisão de qual interface encaminhar o datagrama é tomada (ROSEN *et al.*, 2001).

Segundo Rosen *et al.* (2001), o MPLS combina a funcionalidade dos protocolos de roteamento da camada de rede com a comutação por rótulos, oferecendo benefícios significativos para redes que utilizam IP, Modo de Transferência Assíncrona, ou do inglês, *Asynchronous Transfer Mode* (ATM), ou uma combinação de outras tecnologias no nível da camada de rede. Portanto, em uma arquitetura IP sobre MPLS, as informações necessárias para o encaminhamento são obtidas do cabeçalho MPLS (32 *bits*), que é bem menor e menos complexo que o cabeçalho IP (entre 20 e 60 *bytes*), contribuindo para que os equipamentos de menor poder de processamento e armazenamento tenham desempenho melhor nesse tipo de arquitetura em relação a outras (ROSEN *et al.*, 2001).

Outrossim, em redes baseadas puramente em IP, cada roteador na topologia deve manter uma tabela de roteamento atualizada para encaminhar pacotes com base no endereço IP de destino, calculando a melhor rota dinamicamente. Já em redes baseadas em IP sobre MPLS, o encaminhamento é baseado em rótulos, permitindo que os roteadores de núcleo apenas comutem pacotes com base nesses rótulos, sem a necessidade de analisar o cabeçalho IP para determinar o caminho. Isso reduz a sobrecarga de processamento nos roteadores de núcleo, pois elimina a

necessidade de consulta às tabelas de roteamento tradicionais, tornando o encaminhamento mais eficiente e previsível (MENDONÇA *et al.*, 2012).

4.2.1 Label Distribution Protocol (LDP)

Segundo Mendonça *et al.* (2012), o protocolo LDP é o responsável pela distribuição de rótulos para os prefixos IPs em uma rede baseada em MPLS. Desta maneira, os rótulos são atribuídos a cada prefixo aprendido na tabela de rotas globais de um roteador. Assim, todas as redes propagadas por um mesmo equipamento vizinho, recebem o mesmo rótulo. Com isso, os elementos intermediários em uma rede baseada em MPLS não precisam conhecer a tabela de roteamento completa da topologia. Neste caso, um pacote IP com rótulo é encaminhado para o próximo enlace, baseando-se somente no rótulo externo, isto é, aquele alocado pelo LDP com base na tabela de rotas, prosseguindo assim, até à rede de destino. Neste ínterim, o LDP é configurado para formar adjacências somente com seus vizinhos diretamente conectados. Isso é realizado através de um pacote com o endereço de multicast 224.0.0.2 e o Tempo de Vida, ou do inglês, *Time To Live* (TTL), igual a 1. Neste aspecto, assim que os vizinhos são descobertos inicia-se a troca de rótulos, e as sessões são mantidas através de mensagens de *HELLOS* periódicas. É importante ressaltar que, a perda de três mensagens *HELLO* resulta no encerramento da sessão LDP, uma vez que essas mensagens são enviadas a cada 5 segundos, totalizando um período de 15 segundos sem comunicação. Além disso, uma falha direta na interface também pode causar o encerramento imediato da sessão (ANDERSSON *et al.*, 2007).

4.2.2 Resource Reservation Protocol (RSVP)

Segundo Mendonça *et al.* (2012), o RSVP foi criado como um protocolo de sinalização para que aplicações fossem capazes de reservar recursos, ou seja, para informar a rede acerca de seus requisitos de QoS e efetuar a alocação de recursos ao longo do caminho que o pacote irá percorrer. É importante salientar que as reservas de recursos utilizadas pelo RSVP são *soft-state*, ou seja, tanto o transmissor quanto o receptor devem enviar dados de verificação periodicamente, a fim de manter o estado da sessão e, caso isso não ocorra, a reserva será cancelada. A grande vantagem do serviço integrado ao RSVP é que os recursos espectrais são previamente alocados, já que cada roteador é consultado ao longo do caminho para fazer essa reserva, garantindo assim, a entrega dos dados, caso a reserva seja aceita por todos.

4.2.2.1 Resource Reservation Protocol – Traffic Engineering (RSVP-TE)

O RSVP-TE é o protocolo mais adequado para a distribuição de rótulos em redes MPLS com engenharia de tráfego, sendo uma extensão do RSVP. Dessa forma, ele é adequado para ambientes com MPLS, pois gerencia a reserva de recursos fim a fim para fluxos de tráfego de maneira similar à engenharia de tráfego do MPLS (MENDONÇA *et al.*, 2012). Contudo, o RSVP não supri a todos os requisitos do MPLS, em especial, à distribuição de rótulos e ao controle de caminhos através de rotas explícitas (FARREL, 2004).

Neste ínterim, o RSVP foi inicialmente estendido para esse tipo de aplicação pela *Cisco Systems* durante o desenvolvimento da tecnologia de comutação por *tag*, que era a tecnologia baseada em rótulos, anterior ao MPLS. Posteriormente, o IETF (*Internet Engineering Task Force*) publicou o protocolo RSVP-TE, que reutiliza grande parte do funcionamento original do RSVP. Desta maneira, todas as sete mensagens definidas no protocolo RSVP encontram aplicação no RSVP-TE, embora a mensagem *ResvConf*, que é responsável por confirmar a reserva de recursos ao longo do caminho, tenha um papel menos relevante no RSVP-TE em comparação ao RSVP tradicional.

4.3 Software-Defined Network (SDN)

Segundo Yefan (2021), a SDN é um método de gerenciamento de rede que suporta configuração de rede programável dinâmica. Como tal, melhora o desempenho e a eficiência do gerenciamento da rede e permite que os serviços de rede forneçam recursos flexíveis de personalização, como a computação em nuvem. Deste modo, desacoplando o plano de encaminhamento e o plano de controle dos dispositivos de rede, o SDN usa o controlador para gerenciamento de dispositivos de rede, orquestração de serviços de rede e agendamento de tráfego de serviço, que apresenta baixos custos, além de gerenciamento centralizado e agendamento flexível.

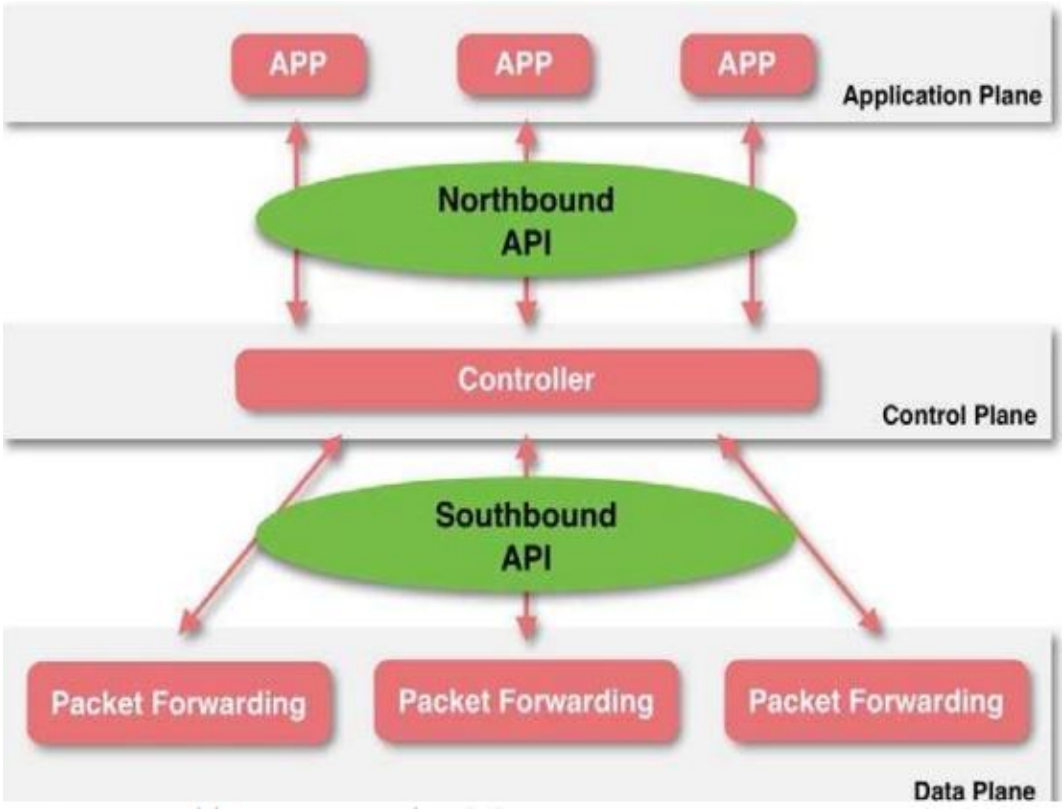
Ademais, de acordo com Chica *et al.* (2020), o SDN incorpora o conceito de programabilidade de rede, uma vez que todas as operações de rede devem ser descritas como programas de *software*, integrando algoritmos, estruturas de dados e conceitos de programação que pertencem ao ambiente de desenvolvimento de *software*. Desta forma, performance e segurança, que são aspectos sensíveis em redes de comunicação e dados por exemplo, podem se beneficiar de recursos SDN, incluindo a própria programabilidade da rede.

Vale ressaltar que esta rede programável é uma proposta recente e eficiente de

infraestrutura de rede, que individualmente administra a camada de dados e a camada de controle, muito diferente das redes legadas (NUNES *et al.*, 2014). A camada de controle é logicamente desacoplada e centralizada, enquanto a camada de encaminhamento de dados persegue as decisões da camada de controle (SINGH; SRIVASTAVA, 2018). Na Figura 5, está sendo apresentada a arquitetura lógica da tecnologia SDN, que ilustra um diagrama abrangente de um sistema SDN, delineando os diversos planos e seus componentes constituintes.

Deste modo, a partir da Figura 5 e com base em Yalda *et al.* (2022), define-se o Plano de Aplicação como um plano que consiste em vários aplicativos e redes serviços, que gerenciam comportamentos de rede. Alguma das aplicações são engenharia de tráfego, balanceamento de carga, roteamento, firewalls, etc. A camada de gerenciamento cria políticas de rede, configura e monitora os dispositivos.

Figura 5 – Arquitetuta lógica SDN



Fonte: (YALDA *et al.*, 2022).

Já o Plano de Controle, ou do inglês, *Control Plane*, tomando a definição de Yalda *et al.* (2022), é o plano que atua na orientação de todo o sistema de uma rede programável por *software*. Neste aspecto, o plano de controle toma todas as decisões de encaminhamento de dados e interpreta as requisições do plano de aplicação, definindo as políticas de encaminhamento

aos dispositivos, assim como atualizar a tabela de encaminhamento de roteadores e *switches* e gerenciar vários controladores auxiliares na mesma camada.

Do mesmo modo, segundo Yalda *et al.* (2022), o Plano de Dados, ou do inglês, *Data Plane*, consiste em dispositivos de rede que têm a capacidade de encaminhar dados com base em tabelas de encaminhamento. O Plano de Dados recebe ordens do Plano de Controle e encaminha os pacotes seguindo as políticas estabelecidas no Plano de Controle. Ademais, a Interface Norte, ou do inglês, *Northbound Interface*, funciona como uma ligação entre o plano de aplicação e o plano de controle. Desta forma, o controlador recebe as instruções da camada de aplicação por meio desta interface. A ideia é análoga para a Interface Sul, ou do inglês, *Southbound Interface*, que conecta o Plano de Controle ao Plano de Dados.

4.4 Segment Routing (SR)

O SR é um protocolo que se baseia no roteamento de origem de pacotes em redes IP, que permite definir caminhos customizados na rede. Desta forma, o nó de origem possui o mapeamento do caminho do nó de destino, introduzido assim a proposta de tornar a engenharia de tráfego mais eficiente, simplificando as operações de plano de controle, como mencionado em Moreno *et al.* (2017). Neste ínterim, o SR trabalha com a aplicação de pilhas de Identificadores de Segmento ou do inglês, *Segment Identifier* (SID), em que cada segmento representa um nó ou um conjunto de nós da rede. Assim, a informação do segmento é anexada ao cabeçalho do pacote e o encaminhamento é realizado de forma eficiente (FILSFILS *et al.*, 2018).

Segundo Maila *et al.* (2017), atualmente, na maioria das redes baseadas em IP, o encaminhamento dos pacotes é feito sobre o protocolo MPLS. No domínio MPLS, o encaminhamento dos pacotes é feito de um roteador de entrada para o roteador de saída, com o auxilio dos rótulos anexados a um pacote. Assim, o encaminhamento dos pacotes se tornará mais rápido, pois não haverá necessidade dos roteadores consultarem por vezes a tabela de encaminhamento de pacotes. Contudo, à medida que a rede se expande, também aumenta o congestionamento no domínio MPLS. Deste fato, uma das principais causas de congestionamento é a formação de milhares de túneis usados no MPLS.

Em face disso, foi criado o SR, um novo conceito que usa o MPLS sem criar túneis adicionais. Desse modo, com o SR, o encaminhamento dos pacotes é feito por meio de segmentos que representam uma lista de instruções. Esses segmentos são, em suma, caminhos curtos pela rede que podem ser combinados para criar uma rota até o destino. Neste aspecto, o SR pode ser

usado na arquitetura MPLS devido ao fato de ser interoperável com o plano de dados e controle MPLS existente, conforme mencionado em (KOS, 2014). Neste sentido, a lista de segmentos pode se assemelhar a uma lista de rótulos e cada operação MPLS tem seu correspondente no SR.

Segundo Maila *et al.* (2017), devido ao fato do SR se apropiar do método de roteamento de origem, um túnel temporário existirá da origem até o destino. Assim, o túnel temporário permanecerá ativo e funcional enquanto houver tráfego passando por ele. Em consonância a isso, no trabalho Kos (2014) é afirmado que o principal benefício do roteamento de origem é que os roteadores não precisam manter informações na tabela de roteamento em função das etapas de encaminhamento serem especificadas no segmento.

Em resumo, dentro de redes baseadas em IPv4, apesar do SR dispensar o uso dos protocolos RSVP-TE e do LDP, as operações de troca de rótulos possuem o mesmo princípio. Os conceitos são abordados a seguir:

SR *policy:* A Política de SR ou do inglês, SR *policy* é uma lista de segmentos ordenados, que pode ser especificada de forma explícita no SR-MPLS usando um pilha de rótulos. Dessa forma, a política de SR pode ser usada com distintas finalidades: Operações de TE, administração e gerenciamento a partir da integração com uma controladora usando SDN, como também para implementação de Redirecionamento Rápido ou do inglês, *Fast Reroute* (FRR), que consiste em uma técnica que garante a recuperação rápida da convergência do roteamento, em caso de falhas em algum enlace ou nó da rede (FILSFILS *et al.*, 2018).

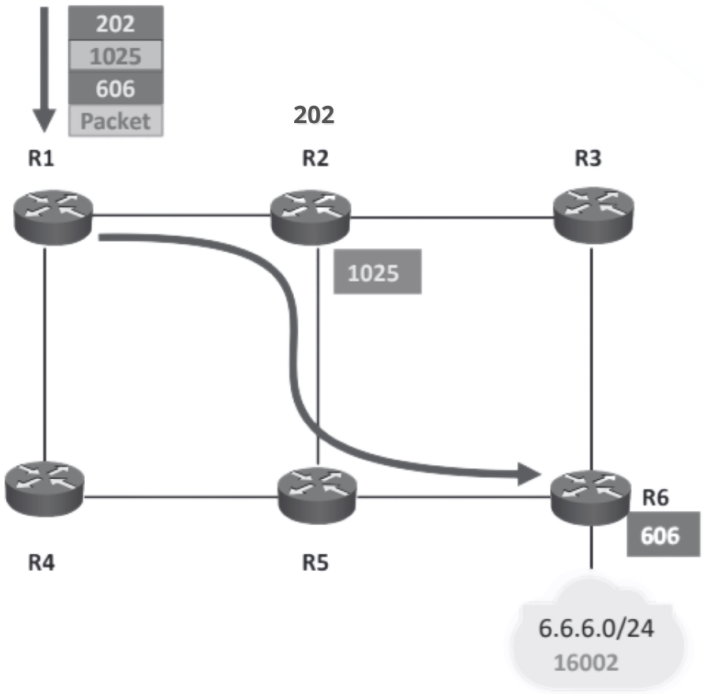
Operação PUSH: Em redes baseadas em SR-MPLS, a operação consiste na inserção de um ou mais rótulos na pilha de *labels* MPLS e, em arquiteturas baseadas em SRv6, equivale à adição de um SID na primeira posição da lista de segmentos, redirecionando o Ponteiro de Lista de Segmentos, ou do inglês, *Segment List Pointer* (SLP), que se trata de um ponteiro usado para indicar qual SID da lista de segmentos está ativo no momento, para o topo desta lista (FILSFILS *et al.*, 2018).

Operação NEXT: A operação NEXT realiza a ativação do próximo segmento na lista de segmentos, determinando a sequência de passos que o pacote deve seguir na rede. Neste aspecto, cada rótulo na pilha representa um segmento ou caminho específico na rede e quando o rótulo superior é processado e consequetemente descartado, inicia-se o processamento de seu subsequente, que será utilizado para o próximo salto ou segmento. Assim, o roteador determina o próximo destino do pacote sem depender de tabelas de encaminhamento (FILSFILS *et al.*, 2018).

Operação CONTINUE: Essa operação ocorre quando o nó atual não é o destino final do segmento ativo, mas sim um ponto intermediário na rota definida pelo protocolo IGP. No SR-MPLS, o nó examina o rótulo no topo da pilha sem removê-lo e encaminha o pacote conforme a tabela de encaminhamento baseada nesse rótulo. Esse processo se repete em cada nó intermediário até que o pacote chegue ao seu destino final (ROSEN *et al.*, 2001). Já para o SRv6, o nó verifica o cabeçalho SR, decrementa o índice que aponta para o SID ativo e então encaminha o pacote para o próximo SID indicado. Deste modo, assegura-se a continuidade do segmento até que ele seja completamente processado (FILSFILS *et al.*, 2018).

A Figura 6 está sendo ilustrado o encaminhamento de pacotes com destino ao prefixo 6.6.6.0/24 em uma topologia baseada em SR com SID de nó. Nessa arquitetura, cada nó da rede recebe um identificador global único. Na figura, é possível observar que o roteador R1 envia o pacote contendo as informações dos saltos a serem realizados na rede, que foram previamente computadas pela extensão do protocolo IGP. Esse protocolo é responsável por distribuir os rótulos pela infraestrutura, que, neste caso, funcionam como identificadores, permitindo que o encaminhamento do pacote seja definido previamente na origem.

Figura 6 – Encaminhamento de pacotes por rótulos de nó

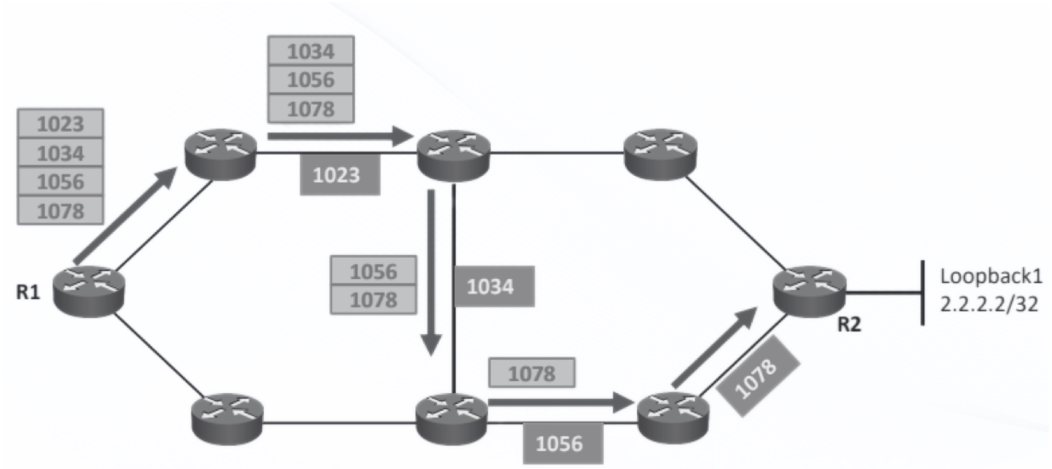


Fonte: (MENDONÇA ROBERTO, 2024).

Por conseguinte, a Figura 7 está sendo apresentado um cenário baseada em SR SID de adjacência, ou do inglês, *adjacency* SID, que consiste no encaminhamento de pacotes baseado

no identificador da adjacência que conecta dois nós, para o prefixo 2.2.2.2/32. Vale salientar que o SID de adjacência tem somente validade local para a topologia, sendo este somente conhecido pelo nó que o dentem. Desta forma, no momento que o nó recebe o pacote com o cabeçalho de rótulos definindo os saltos a serem seguidos, este somente retira o rótulo da interface antecessora e encaminha o pacote para a sucessora, como se segue na figura.

Figura 7 – Encaminhamento de pacotes por rótulos de adjacência.



Fonte: (MENDONÇA ROBERTO, 2024).

Por fim, vale destacar que, pelo fato do SR aproveitar a presença de um elemento centralizado que calcula caminhos usando uma visão global da rede, além de enviar o caminho para o nó de origem do fluxo de tráfego como uma lista ordenada de SIDs, ele pode ser usado de forma conjunta ao SDN e ao roteamento de engenharia de aplicação em redes de operadoras. Neste ínterim, as redes de operadoras normalmente são compostas por vários domínios heterogêneos, com sistemas autônomos divididos em sub-redes implementadas por vários fornecedores e com diferentes tecnologias (KUKREJA *et al.*, 2016).

Neste aspecto, segundo Cai *et al.* (2014), o controlador SDN pode aprender a topologia da rede e as informações de estado em tempo real. Usando essas informações, o controlador pode calcular o melhor caminho de rede com base nos critérios solicitados. Desta maneira, é necessário apenas enviar uma lista ordenada de rótulos para o roteador de origem, de modo a não haver a necessidade de programar cada nó ao longo do caminho, bem como os nós intermediários não necessitarem manter nenhum estado. Assim, obtém-se uma solução

muito mais escalável e simples para a engenharia de tráfego. Fundamentalmente, com esta solução é possível obter uma diversidade de aplicações ou fluxos, que podem ter um número consideravelmente grande de caminhos diferentes pela rede, o que seria impraticável com tecnologias tradicionais como o RSVP-TE. Além disso, todas as alterações aplicadas à pilha de rótulos do SR são refletidas instantaneamente nos caminhos de tráfego.

5 REVISÃO BIBLIOGRÁFICA

Neste capítulo serão apresentados trabalhos e artigos científicos que servem como base e fundamentação teórica para o presente trabalho e que corroboram na justificativa de sua relevância. Em face disso, no trabalho de Galvão (2023), o autor direciona sua atenção em discorrer acerca do protocolo SR, de modo a elaborar uma discussão teórica sobre suas vantagens com relação ao protocolo MPLS. Além disso, o autor propõe a tratativa de um estudo de caso abordado em Mendonça *et al.* (2012), em que ele aplica as tecnologias SR Melhor Esforço, ou do inglês, *Best Effort* (BE) e SR-TE *policy*. Contudo, o autor não explora os possíveis serviços que podem ser implementados através do SR-BE, mas somente compara os cenários em que são aplicados SR-BE e SR-TE, com a finalidade de demonstrar as tecnologias. Nesse sentido, este trabalho de final de curso propõe fazer uma abordagem que melhor explore os recursos da tecnologia SR, em que são implantados serviços de túneis com SR-BE e elaboradas políticas com um maior grau de flexibilidade, tendo em vista que sua proposta está direcionada a metodologias de TE.

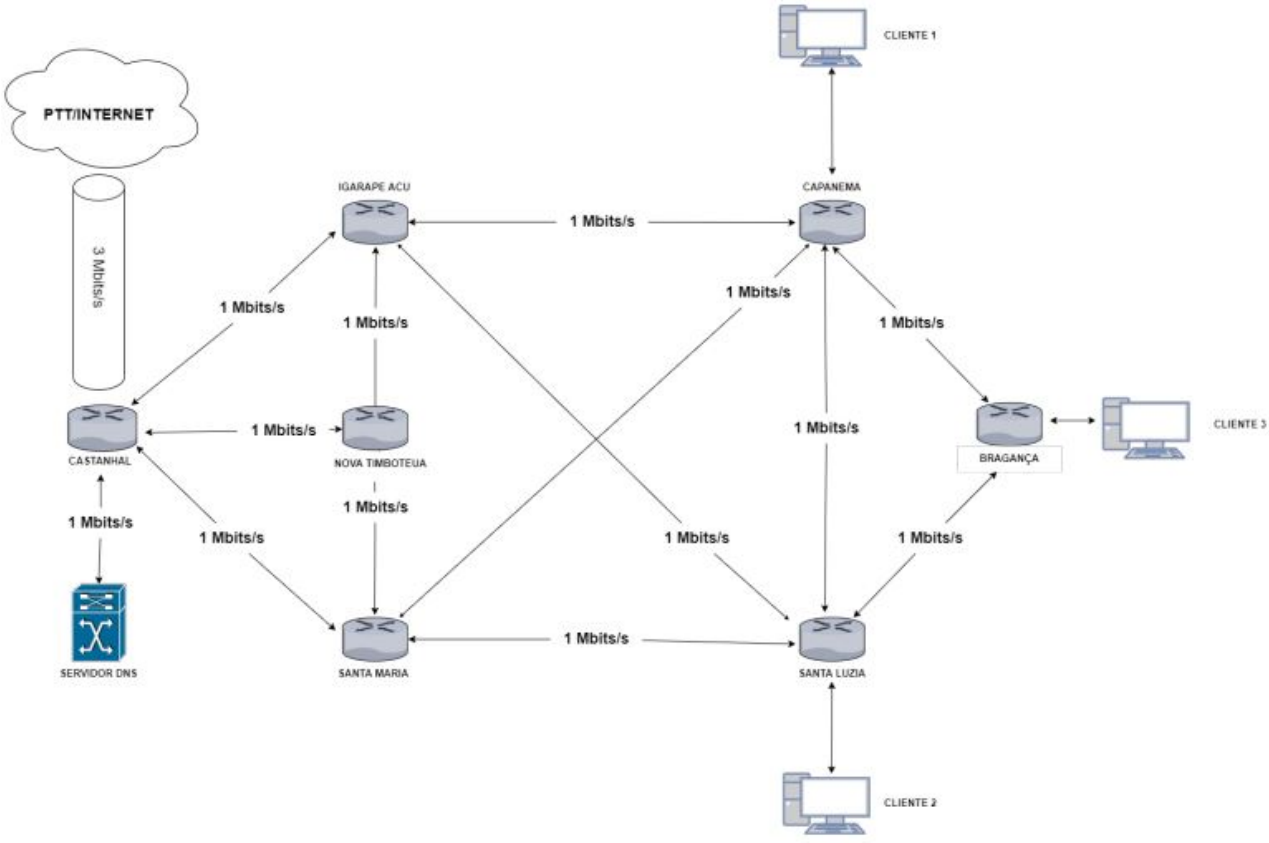
Por outro lado, no artigo Ch *et al.* (2018), os autores exploram conceitos fundamentais do SR-MPLS, apresentando uma discussão detalhada sobre a tecnologia e seu funcionamento. Além disso, são abordados aspectos relacionados à interoperabilidade do SR com o protocolo LDP, demonstrando que ambas as tecnologias podem coexistir em uma mesma rede. O autor também dedica-se a demonstrar que a tecnologia SR foi desenvolvida para ser intregrada ao SDN, com a finalidade de provar que o protocolo é a solução para o novo paradigma das telecomunicações, que busca integrar em suas arquiteturas, tecnologias como 5G, IoT, Computação em Nuvem, serviços de *Streaming*, Realidade Aumentada, etc. Contudo, não faz parte do escopo do artigo tratar de serviços baseados em rótulos, técnicas de TE ou QoS, que para este trabalho final de curso, são conceitos de suma importância e que devem ser amplamente discutidos.

Por fim, tem-se o trabalho JUNIOR *et al.* (2022) em que os autores elaboraram uma solução de manipulação de tráfego baseada no clássico protocolo MPLS, criando túneis de TE e comparou esta solução com o protocolo MPLS LDP, aplicando-os a uma infraestrutura de um ISP regional da cidade de Castanhal, no Pará. Desta forma, apropriando-se de métodos de monitoramento, o autor utilizou a plataforma *Zabbix* em um servidor *Linux*, com o intuito de fazer a coleta de dados de tráfego para um servidor de VoIP e outro de *streaming*. Entretanto, tais métricas não podem ser aplicadas neste trabalho final de curso, tendo em vista que serão realizados estudos avaliando topologias que fazem uso de Máquinas Virtuais, ou do inglês,

Virtual Machines (VMs) que, por sua vez, introduzem fatores que podem distorcer os resultados, em função de compartilharem recursos físicos, como CPU, memória e interfaces de rede, com outras VMs e com o próprio *hypervisor*. Em face disso, esse compartilhamento pode introduzir variabilidade no desempenho, dificultando a obtenção de medições precisas. Portanto, aferir medições de taxa de dados forneceria dados inconsistentes e irreplicáveis.

Todavia, o estudo de caso apresentado em JUNIOR *et al.* (2022) possui um *design* adequado para desenvolver técnicas de TE e que será adotado para este trabalho. Neste ínterim, o que se propõe é demonstrar a superioridade do SR a partir do fato de que os protocolos LDP e RSVP são dispensados, além de que, a partir de extensões do protocolo IGP, previamente configuradas na rede, é possível implementar serviços e políticas baseados em comutação de rótulos, como *Virtual Private Network* (VPN)s de camadas 2 e 3, túneis de TE e TE *policy*. A seguir, na Figura 8 está sendo apresentada a topologia abordada pelo autor do trabalho JUNIOR *et al.* (2022).

Figura 8 – Topologia do backbone.



Fonte: JUNIOR *et al.* (2022).

6 METODOLOGIA

Para o desenvolvimento deste trabalho, foram seguidas diversas etapas, cujo fluxo de atividades será detalhado e explicado nas seções subsequentes.

6.1 Desenvolvimento e Implementação

Como descrito anteriormente, o presente trabalho propõe realizar uma análise comparativa entre as tecnologias MPLS e SR, de maneira a apresentar as principais características e vantagens que o SR possui para aplicações de TE. Em face disso, inicialmente foi feita a implementação da primeira topologia proposta como o primeiro estudo de caso, em que foi efetuada a configuração do protocolo MPLS-TE com LDP e RSVP.

Por conseguinte, foi proposto um segundo estudo de caso, em que se demonstra a constituição de túneis explícitos de TE com SR-BE, ou seja, MPLS-TE com SR-BE. Por fim, foi concebida uma terceira topologia, tida como o terceiro estudo de caso deste trabalho, em que nela se busca aplicar redirecionamento de tráfego com SR-TE *policy*. Para tanto, foi adotado como ferramenta de simulação, o *software* EVE-NG, um emulador *multivendor*, frequentemente usado para estudos na área de redes de computadores (EVE-NG, 2025). Nesse sentido, foram utilizadas as imagens Cisco vIOS Router, vIOS Switch, CSR 1000V, XRv e XRv 9000.

Vale ressaltar que a adoção de imagens Cisco se deve pelo fato de elas serem mais acessíveis dentro do ambiente de simulação escolhido, além de contarem com maior suporte dentro da literatura vigente e possuírem sintaxe intuitiva. Ademais, cabe salientar que para o *software* de emulação em questão, somente a imagem XRv 9000 suporta SR-TE *policy*, visto tratar-se de uma tecnologia recente e ainda pouco explorada dentro do EVE-NG, não sendo possivel implementar com outra imagem.

Contudo, tais limitações não prejudicaram a premissa inicial, uma vez que foi possível observar a superioridade de uma tecnologia com relação à outra. É importante frisar que, no trabalho de Galvão (2023), o autor estava em posse de uma máquina que contava com as configurações de processador Intel Core i7-6700 de 3.40 GHz e 32 GB de memória RAM, podendo serem tomadas como referência para a reprodução deste trabalho. Contudo, com o intuito de se obter um maior grau de confiabilidade, foi adquirido acesso a um servidor remoto, obtido por meio da Network Labs Brasil, onde o plano contratado foi o *Advanced*, disponível no *site* oficial Adilson (2025), que segundo o próprio autor do *site*, é adequado para a implementação

de topologias que estão no nivel de certificação *Cisco Certified Network Professional* (CCNP), que é a certificação da Cisco para profissinais que estão no nivel profissional na área de redes de computadores.

Desta maneira, as seções subsequentes apresentarão as topologias propostas e suas configurações básicas. Vale destacar que o primeiro e o segundo estudo de caso são comparados diretamente, uma vez que o *design* de ambas as arquiteturas é bem próximo, com o intento de demostrar que o SR, em suas configurações básicas, é mais eficiente que o MPLS na constituição de caminhos explícitos de TE. Já para o terceiro estudo de caso, foi necessário abordar uma topologia com formato estrutural mais simples, dado que as VMs utilizadas demandam maior quantidade de recursos computacionais. Contudo, este fato não invalida os resultados obtidos no experimento, visto que, o único limitante para a implementação de uma rede maior, é somente a capacidade operacional da máquina utilizada na simulação.

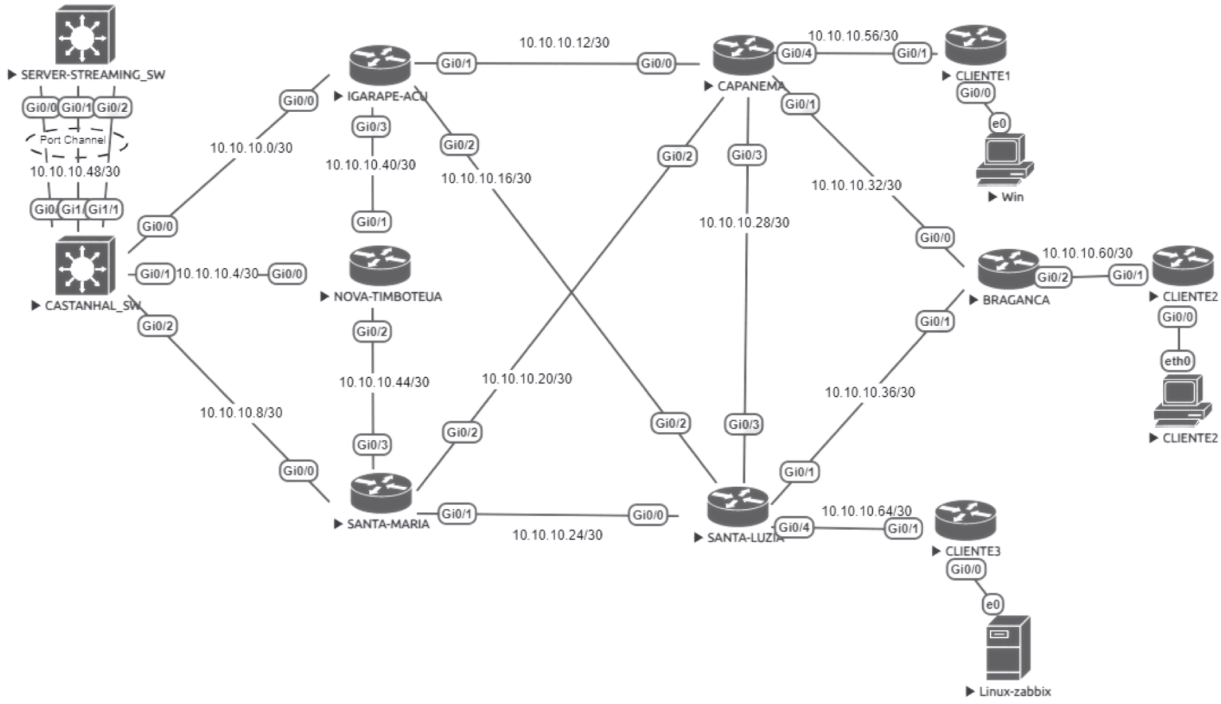
6.1.1 Primeiro Estudo de Caso

Nesta subseção encontra-se o primeiro estudo de caso, em que foi tomada como referência a topologia mostrada na Figura 8. Nela, foram realizadas algumas adaptações, com o intuito de torná-la mais adequada à proposta inicial, de modo a enfatizar somente os pontos essenciais para este trabalho. Dito isto, na Figura 9 está sendo apresentado o cenário adaptado para a simulação, no qual é possível observar a presença de três clientes na rede, que podem ser entendidos como empresas que contratam serviços de uma operadora, a fim de estabelecerem conectividade com um servidor de *streaming*, conectado ao *switch* CASTANHAL por uma interface *port-channel*, que, por sua vez, corresponde ao agrupamento da capacidade de tráfego das três interfaces físicas em uma única interface lógica, oferencendo assim, um maior grau de confiabilidade.

Deste modo, cada empresa demanda requisitos de alta performance e, com o objetivo de atendê-las, serão criados túneis de TE com MPLS, cuja a demonstração será apresentada adiante. É importante frisar que eles foram implementados de tal modo que não houvesse sobreposição de caminhos entre eles, com o intento de demonstrar as técnicas de forma mais didática. Contudo, em arquiteturas reais, os túneis podem ser implementados por um mesmo caminho sem qualquer prejuízo, uma vez que eles são constituídos por interfaces lógicas, chamadas de *Interface Tunnel*, que também serão apresentadas posteriormente.

Ademais, é necessário destacar que, o protocolo MPLS só pode ser implantado após

Figura 9 – Topologia do primeiro estudo de caso.



Fonte: Adaptada de JUNIOR *et al.* (2022)

a implementação de um protocolo IGP na rede, porquanto o LDP distribui os rótulos seguindo as métricas estabelecidas pelo protocolo de roteamento configurado. Deste modo, iniciou-se a ativação do protocolo IGP na topologia, que para o presente caso, foi o protocolo OSPF. Na Figura 10 estão sendo apresentados os comandos necessários para o funcionamento do OSPF na rede.

Em suma, observando-se a Figura 10, na linha 2 da imagem, foi criado um processo OSPF e foi-lhe atribuído um valor numérico. De mesmo modo, na linha 3 foi adicionado um identificador para o equipamento, que possui o formato de um endereço IP. Neste ínterim, foi configurada a interface de *Loopback* como passiva, como pode ser observado na linha 4, sendo esta ação tomada por questões de boas práticas de projeto, não fazendo, portanto, parte do

Figura 10 – Configuração do protocolo OSPF

```
1 !
2 router ospf <1-65535> % ID do processo
3  router-id A.B.C.D % router-id do OSPF em formato de end. IP
4  passive-interface Loopback <0-2147483647> % numero da interface
5 !
6 interface <type number>
7  ip ospf <1-65535> area <0-4294967295> % area do OSPF
8 !
```

Fonte: Elaborado pelo autor.

escopo deste trabalho explicar as razões para tal. Por fim, as linhas 6 e 7 correspondem às interfaces que fazem parte dos roteamentos que devem estar dentro do processo OSPF existente e atribuídas a uma área, definida por um identificador numérico. É importante salientar que esta configuração foi apresentada em formato genérico, visto conter comandos secundários, cujo o detalhamento foge do objetivo final deste trabalho final de curso. Contudo, não excetuando sua importância, cabe destacar que o OSPF deve ser incorporado a todos os equipamentos que necessitam compartilhar rotas internas pela rede.

Por fim, com o OSPF operacional na rede, foi feita a implementação do MPLS com LDP, conforme mostrado na Figura 11. Desta maneira, na linha 2, foi definido o *router-id* para o protocolo LDP como sendo o endereço da interface *Loopback 0*. Após isso, nas linhas 4 e 5 é feita a demonstração de que o MPLS deve ser ativado nas interfaces de redes internas da rede, de tal modo que estes comandos devem ser inseridos em todos os equipamentos da rede. Logo, para o presente caso de estudo, estas ativações foram aplicadas a todos os equipamentos, exceto nos roteadores dos clientes e nas interfaces de rede que os conectam.

Figura 11 – Configuração do protocolo MPLS com LDP

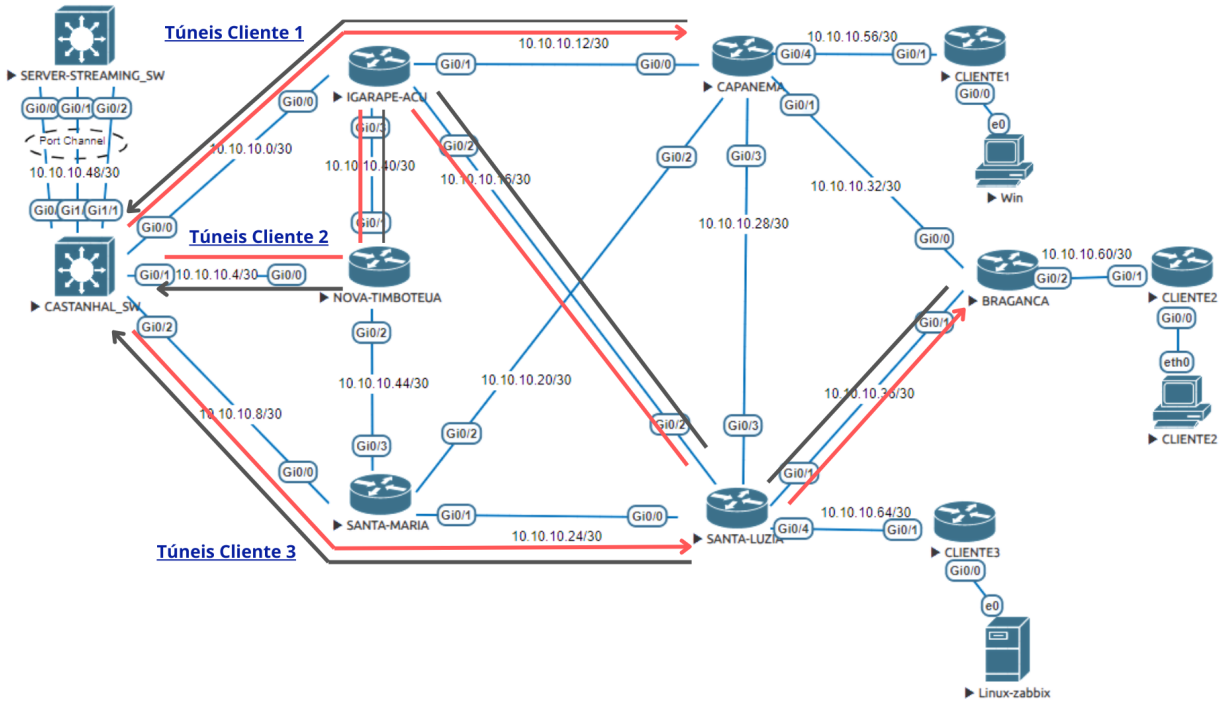
```
1 !
2 mpls ldp router-id Loopback0 force
3 !
4 interface <type number>
5  mpls ip
6 !
```

Fonte: Elaborado pelo autor.

Desta maneira, foi iniciada a implementação do MPLS-TE. Dada a Figura 9, foram criados túneis de TE para cada cliente, como pode ser observado na Figura 12, de modo que foram representados dois túneis para cada consumidor, uma vez que eles são unidirecionais. Logo, é necessário criar tanto o túnel de ida quanto o de volta. Vale ressaltar que os túneis não se destinam ao equipamento dos assinantes, visto que a comutação de rótulos só ocorre no núcleo da rede do ISP.

Em face disso, na Figura 13 estão sendo apresentados os comandos necessários para a implementação dos túneis de TE. Estes comandos são referentes ao equipamento CASTANHAL, onde dele são originados três túneis dedicados a cada um dos clientes. Assim, com o intuito de não tornar o *script* repetitivo, apresenta-se somente a criação de um túnel, que se destina ao roteador BRAGANÇA. Para os demais, o processo é análogo e também deve ser realizado nos

Figura 12 – Cenário MPLS com túneis de TE



Fonte: Adaptada de JUNIOR *et al.* (2022)

nós que conectam os equipamentos dos assinantes, porquanto é necessário criar os túneis da volta.

Desta forma, o comando que foi inserido na linha 2 corresponde à ativação do MPLS-TE. Já o da linha 4 corresponde à implementação do caminho explícito à rede de destino, o roteador BRAGANÇA. A partir disso, foi preciso descrever os saltos que o pacote deveria percorrer até chegar ao assinante, como é possível observar nas linhas 5 a 8. Por conseguinte, a linha 10 apresenta a instrução para a criação da *interface tunnel*, onde ela recebe o mesmo endereço de IP da interface de *Loopback 0*, como observado na linha 12. Assim, uma série de comandos são executados: primeiramente foi ativado o modo MPLS-TE; em seguida, adicionado o seu endereço de destino, bem como sua visibilidade para o protocolo OSPF. Após isso, foi inserida sua prioridade no processo de encaminhamento de pacotes, além dos dados de sinalização, que correspondem a uma largura de banda de 10.000 kbps, como é possível observar a partir das linhas 13 a 17, respectivamente. Por fim, associou-se o caminho explícito ao túnel, como pode-se perceber na linha 18.

Com isso, é possível observar na linha 21 que a interface *GigabitEthernet0/1*, que conecta o equipamento CASTANHAL a NOVA_TIMBOTEUA, conforme a Figura 12, foi acessada, além de que o suporte ao MPLS-TE foi ativado nela, bem como, o protocolo RSVP,

Figura 13 – Configuração do MPLS-TE na rede

```
1  !
2  mpls traffic-eng tunnels
3  !
4  ip explicit-path name CASTANHAL->BRAGANCA enable
5   next-address 10.10.10.6
6   next-address 10.10.10.41
7   next-address 10.10.10.18
8   next-address 10.10.10.38
9  !
10 interface Tunnel22
11  description CASTANHAL->BRANGANCA
12  ip unnumbered Loopback0 % associa com o end. da loopback0
13  tunnel mode mpls traffic-eng
14  tunnel destination 10.10.10.38
15  tunnel mpls traffic-eng autoroute announce % visivel para IGP
16  tunnel mpls traffic-eng priority 1 1
17  tunnel mpls traffic-eng bandwidth 10000 % em kbps
18  tunnel mpls traffic-eng path-option 2 explicit name CASTANHAL->
        BRAGANCA
19  no routing dynamic
20 !
21 interface GigabitEthernet0/1
22  mpls traffic-eng tunnels
23  ip rsvp bandwidth 10000 10000 % Reserva de banda
24 !
25 router ospf 1
26  mpls traffic-eng router-id Loopback0
27  mpls traffic-eng area 0.0.0.0
28 !
```

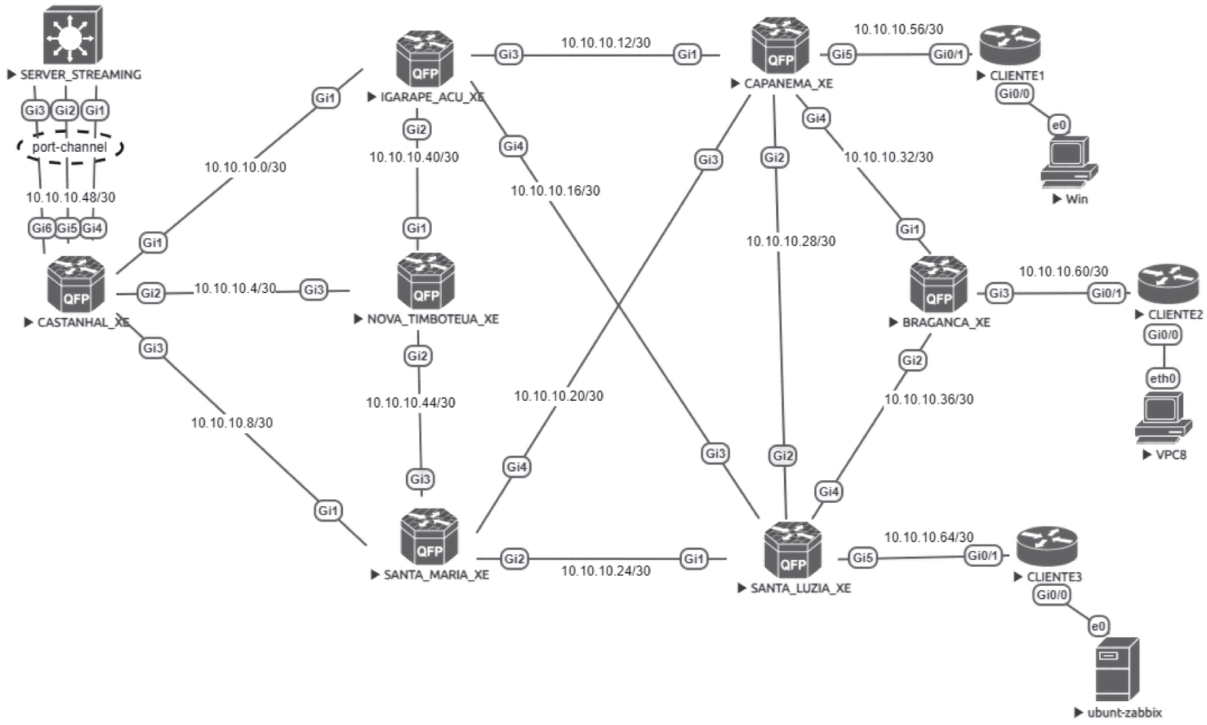
Fonte: Elaborado pelo autor.

com uma reversa de banda de 10.000 kbps. É importante destacar que estes comandos devem ser ativados somente nas interfaces de rede que correspondem ao caminho explícito do túnel. Por fim, a extensão do MPLS-TE foi implementada dentro do processo OSPF, como demonstrado nas linhas 26 a 28. Outrossim, faz-se mister destacar que estes comandos devem ser inseridos em todos os nós pertecentes ao núcleo baseado em MPLS.

6.1.2 Segundo Estudo de Caso

Nesta subseção são apresentadas as configurações para a implantação do MPLS-TE com SR-BE. Deste modo, a Figura 14 apresenta o cenário da Figura 9, com alterações no núcleo de sua arquitetura, em que as VMs Cisco vIOS Router e os vIOS Switch foram substituídos por Cisco CSR 1000, uma vez que o SR necessita de equipamentos com uma maior capacidade de processamento para ser suportado. Sendo assim, foi feito o projeto de configuração dos equipamentos, que tem como objetivo criar os mesmos túneis de TE apresentados na topologia

Figura 14 – Cenário modificado para a migração do SR-BE



Fonte: Adaptada de JUNIOR *et al.* (2022).

da Figura 12, visto que, para o presente caso, não mais dependem dos protocolos LDP e RSVP, mas somente do protocolo IGP, que, assim como para a topologia baseada em MPLS LDP, foi utilizado OSPF.

Nesta perspectiva, iniciou-se a implementação do protocolo OSPF, que consistiu no mesmo procedimento apresentado na Figura 10. Nesse contexto, o projeto foi prosseguido com a configuração do SR-BE, cujo o *script* é apresentando na Figura 15. Neste caso, a demonstração corresponde ao equipamento CASTANHAL, todavia, o processo é análogo para os demais nós da rede. Em suma, a linha 2 ativa o recurso de segmentação usando o plano de dados MPLS, permitindo que o equipamento encaminhe dados baseado em rótulos. Por conseguinte, na linha 3 foi definido o Bloco Global de Roteamento por Segmentos, ou do inglês, *Segment Routing Global Block* (SRGB), que nada mais é do que a faixa de rótulos, ou SID, para o caso de uma rede baseada em SR, que os roteadores usam como identificadores globais dentro da rede.

Já para o caso da linha 5, adentrou-se o submodo de configuração, no qual são definidos os mapeamentos específicos da família de endereços para prefixos locais e SIDs e, na linha seguinte, definiu-se a família de endereços IPv4. Desta maneira, na linha 7 foi associado o SID 16001 com o endereço 1.1.1.1/32, que corresponde ao endereço de IP da interface *Loopback 0*. Por fim, a partir da linha 11, foi necessário entrar no processo OSPF que foi configurado

inicialmente, além de ativar o modo SR-MPLS dentro da área 0 e habilitar a função de propagação de rótulos.

Neste aspecto, o procedimento foi feito em todos os nós que compõem o núcleo baseado em SR, em que foram alterados os valores dos SIDs globais, sendo que cada numeral possível precisa necessariamente estar dentro do SRGB, definido anteriormente. Vale salientar que por definição, o intervalo de valores que o SRGB assume é de 16000-23999, embora outros valores possam ser utilizados. Assim, a escolha dos SRGB foi definida de forma arbitrária, visando ser simples e de fácil identificação na rede. Deste modo, com o comando <*show segment-routing mpls connected-prefix-sid-map ipv4*> é possível visualizar os SIDs globais de todos os equipamentos que trocam rótulos, cuja saída do comando é apresenta na Figura 16.

Neste sentido, na primeira coluna da imagem são apresentados os IPs das interfaces de *Loopback* dos nós, sendo cada um deles mapeado por um índice. À vista disso, para o endereço 1.1.1.1/32 foi associado o índice 1, o que significa que o SID atribuído a esse prefixo é o 16001, que está em consonância com *script* apresentado na Figura 15. Seguindo essa lógica, o SID correspondente ao endereço 2.2.2.2/32 equivale ao 16002 e, assim por diante, até o mapeamento total da rede, possibilitando a comunicação ser baseada em rótulos.

Em posse disso, com o SR-BE habilitado em todos os equipamentos que compõem o núcleo, foi possível executar o projeto de criação dos túneis de TE, que consistiu na mesma arquitetura lógica apresentada na Figura 12, sendo, contudo, aplicada ao cenário apresentado na Figura 14. Com isso, obteve-se a topologia mostrada na Figura 17. Nesta representação é notório que as aplicabilidades apresentadas não se distanciam daquelas demonstradas para a topologia

Figura 15 – Configuração do SR-BE no equipamento CASTANHAL

```
1  !
2  segment-routing mpls % ativando o segment routing
3   global-block 16000 16010 % SRGB
4   !
5   connected-prefix-sid-map
6    address-family ipv4
7     1.1.1.1/32 absolute 16001 range 1
8    exit-address-family
9   !
10 !
11 router ospf 1
12  segment-routing area 0 mpls
13  segment-routing mpls
14 !
```

Fonte: Elaborado pelo autor.

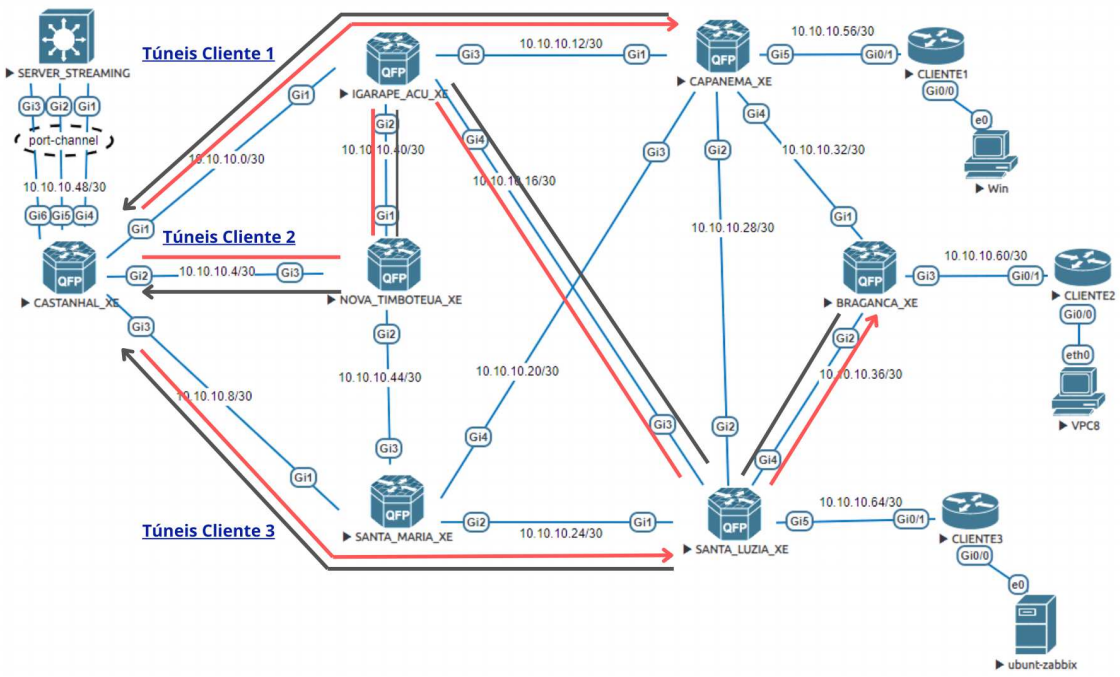
Figura 16 – SIDs Globais da rede SR

```
CASTANHAL_XE#show segment-routing mpls connected-prefix-sid-map ipv4

             PREFIX_SID_CONN_MAP ALGO_0

   Prefix/masklen   SID Type Range Flags SRGB
       1.1.1.1/32 16001  Abs     1         Y

             PREFIX_SID_PROTOCOL_ADV_MAP ALGO_0

   Prefix/masklen   SID Type Range Flags SRGB Source
       1.1.1.1/32     1 Indx     1         Y  OSPF Area 0 1.1.1.1
       2.2.2.2/32     2 Indx     1         Y  OSPF Area 0 2.2.2.2
       3.3.3.3/32     3 Indx     1         Y  OSPF Area 0 3.3.3.3
       4.4.4.4/32     4 Indx     1         Y  OSPF Area 0 4.4.4.4
       5.5.5.5/32     5 Indx     1         Y  OSPF Area 0 5.5.5.5
       6.6.6.6/32     6 Indx     1         Y  OSPF Area 0 6.6.6.6
       7.7.7.7/32     7 Indx     1         Y  OSPF Area 0 7.7.7.7
```

Fonte: Elaborado pelo autor.

baseada em MPLS, onde o mesmo problema foi resolvido. De fato, a proposta neste momento é demonstrar que o SR, em suas funcionalidades básicas, se equipara ao MPLS-TE, ao passo que se apresenta como um protocolo operacionalmemte menos custoso, visto que não foram adicionados protocolos de propagação de rótulos, como também, de reservas de recursos.

Figura 17 – Túneis de TE implementados com SR-BE



Fonte: Adaptada de JUNIOR *et al.* (2022).

Com isso, foram implementados nos equipamentos os comandos necessários para as configurações dos túneis. Assim como anteriormente, será apresentado somente o *script*

Figura 18 – Configuração do túnel de TE usando SR-BE na rede

```
!
mpls traffic-eng tunnels
!
ip explicit-path name CASTANHAL->BRAGANCA enable
 index 1 next-address 10.10.10.6
 index 2 next-address 10.10.10.41
 index 3 next-address 10.10.10.18
 index 4 next-address 10.10.10.38
!
interface Tunnel22
 description TUNNEL CASTANHAL->BRAGANCA
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.38
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name CASTANHAL->
    BRAGANCA segment-routing
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
interface GigabitEthernet2
 mpls traffic-eng tunnels
!
```

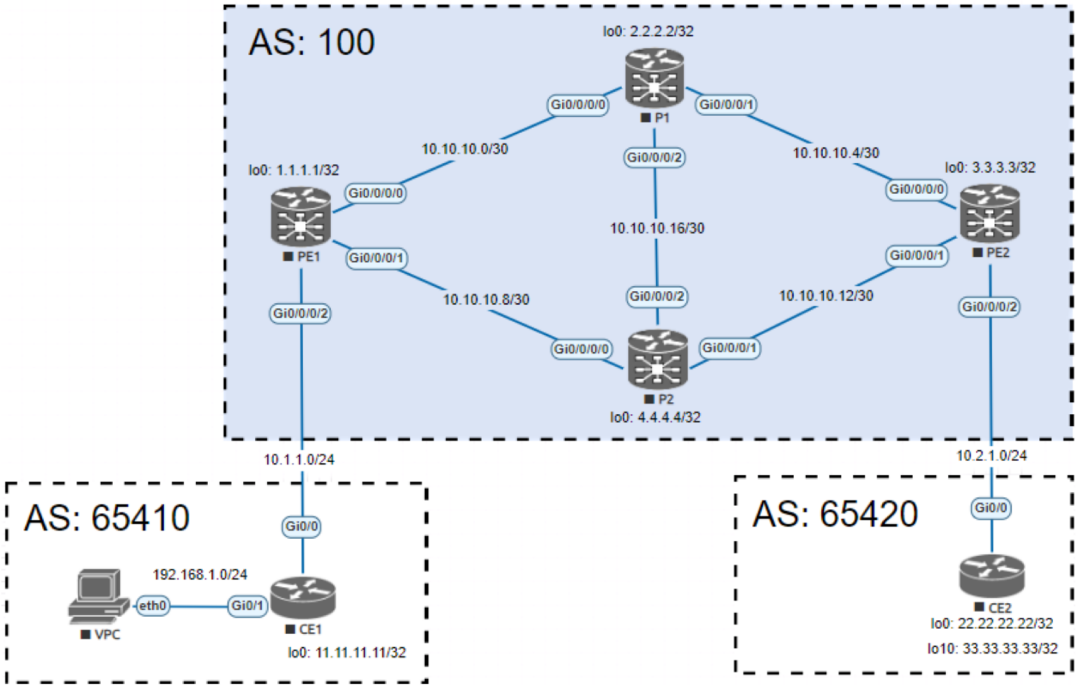
Fonte: Elaborado pelo autor.

de ativação do túnel, que tem como origem o roteador CASTANHAL e que se destina ao nó BRAGANÇA. Desta forma, na Figura 18 estão sendo apresentadas as instruções de constituição do túnel. Pela imagem, é possível perceber que os parâmetros acionados são os mesmos que o da Figura 13, com exceção das linhas que habilitam os protocolos LDP e RSVP.

6.1.3 Terceiro Estudo de Caso

Por fim, esta subseção dedica-se à apresentação do terceiro estudo de caso proposto para este trabalho, cujo objetivo é abordar as funcionalidades do SR-TE baseado em políticas de redirecionamento de tráfego. Neste caso, o que se propõe é implementar um método de manipulação de tráfego que é orientado pelo prefixo de IP de destino, ou seja, para um determinado bloco de endereços, o tráfego é direcionado para um caminho específico. Esta técnica pode ser implementada em ISPs que provêm serviços para assinantes de segmento corporativo e que necessitam de alta disponibilidade para aplicações críticas, de modo que o tráfego percorre um caminho dedicado no momento em que ele tem como destino um determinado endereço de IP,

Figura 19 – Cenário genérico para a implementação do SR-TE *policy*



Fonte: Adaptada de HUAWEI (2024)

que está associado a um serviço essencial.

Na Figura 19 está sendo apresentada a topologia proposta, que se trata de um cenário genérico retirado de HUAWEI (2024), em que estão conectados três ASs, sendo o 100 responsável por fazer a comunicação entre o 65410 e o 65420. Para esta arquitetura, estão sendo utilizadas as VMs Cisco XRv 9000, vIOS Router e um Computador Pessoal Virtual, ou do inglês, *Virtual Personal Computer* (VPC). Desta maneira, por se tratar de uma topologia genérica, os equipamentos possuem nomeclaturas gerais, que consistem em Borda do Cliente, ou do inglês, *Customer Edge* (CE), Roteador de Borda do Provedor, ou do inglês, *Provider Edge* (PE) e o Roteador de Núcleo do Provedor, ou do inglês, *Provider* (P).

Assim, este projeto conta com a integração da configuração de uma clássica VPNv4 de camada 3, em que os roteadores CEs estabelecem uma seção BGP com os PEs por um mesmo Roteamento e Encaminhamento Virtual, ou do inglês, *Virtual Routing and Forwarding* (VRF), que consiste em uma tecnologia que permite que os roteadores mantenham múltiplas tabelas de roteamento isoladas, possibilitando assim, a segmentação de redes dentro de um mesmo equipamento. Por outro lado, é necessário estabelecer uma seção de Protocolo de *Gateway* de Borda Interno, ou do inglês, *Internal Border Gateway Protocol* (iBGP) entre os PEs, uma vez que as rotas dos clientes não podem ser computadas na tabela de roteamento dos nós Ps. Contudo, estas configurações não serão apresentadas, pelo fato de se distanciarem do objetivo

Figura 20 – Configuração do IS-IS na rede

```
1  router isis <WORD> % ID do processo
2   max-metric level 1
3   is-type level-1
4   net XX.XXXX.XXXX.XXXX.XX % Definicao do NET
5   address-family ipv4 unicast
6    metric-style wide
7   !
8   interface <type number>
9    address-family ipv4 unicast
10   !
11  !
```

Fonte: Elaborado pelo autor.

final e serem ativações auxiliares.

Após estabelecidas a seções BGP e iBGP, foi preciso implementar o protocolo IGP para a troca de rotas internas dentro do rede. Para tanto, foi utilizado o protocolo IS-IS. Neste caso, ele foi utilizado unicamente para fins demonstrativos, visto que o SR pode ser integrado com qualquer protocolo de LS. Desta forma, a Figura 20 apresenta as configurações básicas do IS-IS, em formato genérico. Assim, na linha 1, foi iniciado o processo IS-IS, que pode receber um nome qualquer. Já na linha 2, foi definida a métrica máxima para o nível, o que evita a primazia do roteador para o cálculo do roteamento. Na linha 3, foi inserido o comando que define que o equipamento atue no nível 1, ou seja, dentro de uma única área.

Posto isto, na linha 4 é definido o Título da Entidade de Rede, ou do inglês, *Network Entity Title* (NET), que é o identificador exclusivo do nó para o IS-IS. Em seguida, na linha 5 foi ativada a família de endereços IPv4 e, por conseguinte, foi definida a métrica *wide* para o cálculo e a propagação de rotas, uma vez que ela permite um número maior de enlaces pertencentes ao roteamento. Por fim, na linha 8 é apresentado o comando para inserção das interfaces de interesse, dentro da instância IS-IS, como também a adição delas dentro da família de endereços IPv4. Novamente, essas configurações estão sendo demonstradas em formato genérico, pelo fato de serem ativações secundárias, que, apesar de sua extrema importância para a implementação do SR, não são o foco deste trabalho.

Munido disso, implementou-se o SR-BE na rede, porém agora, fazendo uso das extensões do IS-IS. A Figura 21 apresenta o *script* aplicado ao equipamento PE1. Entretanto, o processo é análogo para os demais equipamentos que compõem a rede do AS 100. Portanto, é possível perceber que o procedimento é semelhante ao que foi realizado na Figura 15, em que na linha 2 é definido o SRGB, no intervalo de 16000 à 16010. Em vista disso, após acessar a

Figura 21 – Configuração do SR-BE com IS-IS

```
1  segment-routing
2   global-block 16000 16010
3  !
4  router isis 1
5   address-family ipv4 unicast
6    segment-routing mpls
7   !
8  !
9   interface Loopback0
10    prefix-sid absolute 16001
11   !
12  !
13 !
```

Fonte: Elaborado pelo autor.

instância IS-IS e o conjunto de endereços IPv4, que correspondem aos comandos das linhas 4 e 5, respectivamente, é feita a ativação do SR-MPLS, como pode ser observado na linha 6. Por fim, ainda dentro da instância IS-IS, a interface *Loopback 0* deve ser incluída, assim como deve ser associada ao SID correspondente ao equipamento, sendo este processo correspondente às linhas 9 e 10.

Nesta perspectiva, foi viável validar o funcionamento do SR-BE na rede, com o comando <*show isis segment-routing label table*>. A saída do comando é apresentada na Figura 22. Nela, consegue-se perceber que o SID 16001 foi associado à interface *Loopback 0* do PE1 e os demais foram atribuídos aos prefixos pertencentes aos outros equipamentos da rede. Ademais, cabe também falar que estas redes pertencem à interface *Loopback* de cada equipamento, de modo a estarem em conformidade com o projeto apresentado na Figura 19.

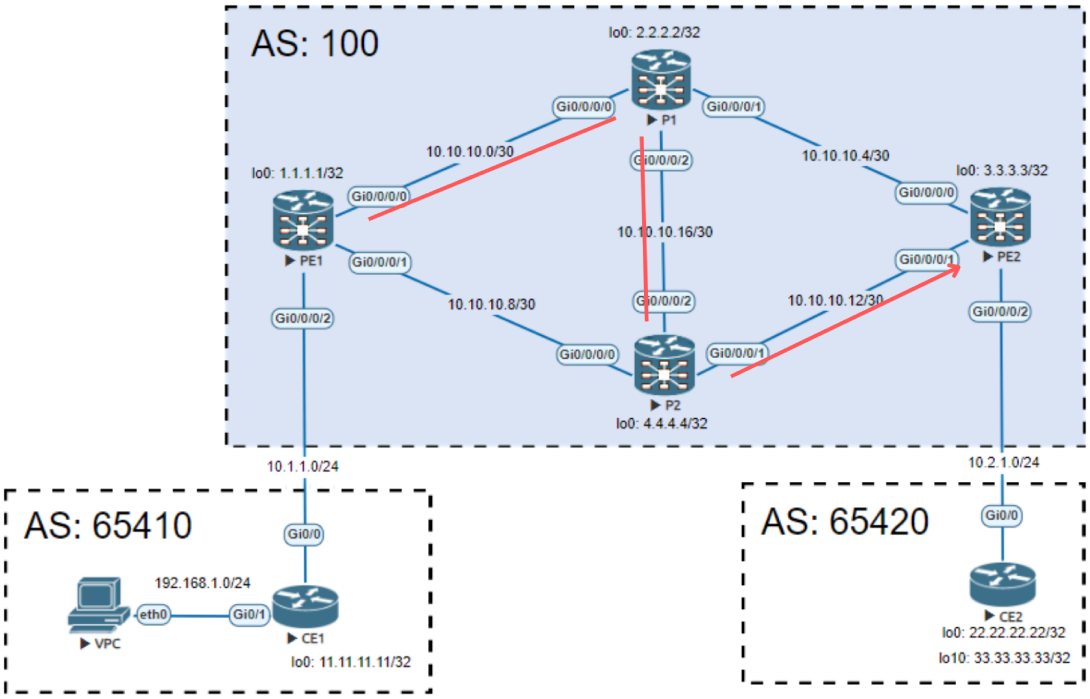
Figura 22 – SID globais do cenário do SR-TE policy

```
RP/0/RP0/CPU0:PE1#sh isis segment-routing label table
Tue Feb  4 23:44:55.074 UTC

IS-IS 1 IS Label Table
Label         Prefix/Interface
----------    ---------------
16001         Loopback0
16002         2.2.2.2/32
16003         3.3.3.3/32
16004         4.4.4.4/32
```

Fonte: Elaborada pelo autor.

Com isso, iniciou-se o desenvolvimento da política de TE para a topologia apresentada na Figura 19. Desta maneira, a Figura 23 demonstra a manipulação do tráfego realizada

Figura 23 – Cenário do estudo de caso 3 com a política de TE



Fonte: Adaptada de HUAWEI (2024).

para este cenário, de forma que as setas indicam o caminho que os dados devem percorrer. Para este caso, o que se propõe é que o tráfego percorra os trechos destacados quando o endereço de destino for 22.22.22.22/32, sendo este prefixo pertecente ao AS 65420. Vale ressaltar que este método pode ser útil quando existe a necessidade de se dedicar um enlace para uma aplicação crítica. Por outro lado, todo o tráfego não classificado da rede segue o caminho determinado pelo protocolo IGP.

Com isso, contornar o congestionamento de tráfego de dados é factível, porquanto faz-se com que as requisições destinadas a um determinado serviço sigam por uma rota distinta das demais aplicações existentes na rede. Logo, dada a justificativa para a implementação desta política de TE, iniciou-se a configuração dela na topologia. A Figura 24 apresenta os comandos que foram realizados no equipamento PE1 para a implementação da política de TE. Assim, na linha 3 foi feita a ativação das funcionalidades de TE para o SR. Por outro lado, na linha 4 foi criada uma lista de segmentos, chamada VPNA_CORE e, dentro da lista de segmentos, foram adicionados os SIDs, que mapeiam o caminho a ser percorrido pelos dados, sendo ele correspondente aos equipamentos P1,P2 e PE2, respectivamente.

Por conseguinte, na linha 9 foi implantada a SR-TE *policy*, definida como VPNA_CORE. Na linha 10, foi atribuída a ela a *color 100*, que consiste em um identificador usado para associar políticas de TE a caminhos específicos e, definindo como endereço de destino, o prefixo 3.3.3.3,

Figura 24 – Configuração da política baseada em SID

```
1  !
2  segment-routing
3   traffic-eng
4    segment-list VPNA_CORE
5     index 10 mpls label 16002
6     index 20 mpls label 16004
7     index 30 mpls label 16003
8    !
9    policy VPNA_CORE
10    color 100 end-point ipv4 3.3.3.3
11    candidate-paths
12     preference 100
13      explicit segment-list VPNA_CORE
14      !
15     !
16    !
17   !
18  !
19 !
20 end
```

Fonte: Elaborada pelo autor.

que é o endereço de IP da interface *Loopback 0* do nó PE2. Após isso, na linha 11 foi inserido o comando em que será definido um caminho candidato para a política. Em seguida, foi acrescentada sua preferência, visto que uma política pode possuir mais de um caminho candidato. Cabe frisar que quanto menor o valor inserido na *preference*, maior sua prioridade. Por fim, na linha 13 foi especificado que o caminho explícito que será usado é o VPNA_CORE.

Feito isso, foi implementada a política de roteamento, que faz a associação do tráfego de dados ao prefixo de destino. Neste caso, o que foi determinado, é que todo tráfego cujo endereço de destino fosse 22.22.22.22/32 seguisse o caminho associado a SR-TE *policy*, ou seja, o trecho descrito na Figura 23. Para tanto, os comandos apresentados na Figura 25 foram inseridos no roteador PE2. Assim, na linha 2 foi definida uma *Extended Community* (Comunidade Estendida) chamada VPNA_ACCESS do tipo *opaque* (opaco), que consiste em um atributo usado pelo protocolo BGP para fornecer informações adicionais sobre rotas, de modo que ela permite anexar marcadores às rotas, que podem ser aplicados a políticas de controle de tráfego. Em consonância a isso, o tipo *opaque* significa que o conteúdo desta comunidade não é interpretado diretamente pelo roteador, todavia contém informações específicas para a dada aplicação a qual está sendo incorporada que, neste caso, é a SR-TE *policy* VPNA_CORE.

Em face disso, foi atribuída a comunidade extendida 100, que coincide com a *color* da SR-TE *policy* VPNA_CORE. Após isso, criou-se na linha 6 uma política de roteamento,

Figura 25 – Configuração da política de roteamento no PE2

```
1  !
2  extcommunity-set opaque VPNA_ACCESS
3    100
4  end-set
5  !
6  route-policy VPNA_ACCESS
7    if destination in (22.22.22.22/32) then
8      set extcommunity color VPNA_ACCESS
9    else
10     pass
11   endif
12 end-policy
13 !
```

Fonte: Elaborada pelo autor.

denominada VPNA_ACCESS. Já na linha 7, realizou-se uma operação de condicionalidade, que verifica se o destino do tráfego é o prefixo 22.22.22.22/32 e, para o caso afirmativo, a rota recebe a comunidade estendida VPNA_ACCESS, ou seja, a comunidade extendida 100 definida anteriormente, conforme se observa na linha 8. Por outro lado, se o destino não for 22.22.22.22/32, a política permite que o tráfego siga pela rota de encaminhamento determinada pelo protocolo IGP, sendo essa instrução realizada nas linhas 9 a 11.

7 RESULTADOS E DISCUSSÕES

Neste capítulo são discutidas as técnicas de implementação de TE com MPLS e SR, que foram trabalhadas no capítulo anterior, com o propósito de avaliar qual das tecnologias possui melhor flexibilidade e escalabilidade. Neste ínterim, a primeira e a segunda seção compararam diretamente o primeiro e o segundo estudo de caso, visto que em ambos, são implementados túneis explícitos de TE, com a diferença de que, no segundo estudo caso, os protocolos LDP e RSVP são dispensados e, como será demonstrado posteriormente, possibilita uma diminuição considerável de dados de sinalização na rede. Já a ultima seção se destina a discorrer acerca do terceiro estudo de caso, com a finalidade de validar a flexibilidade e escalabilidade do SR-TE *policy*, para projetos de redes com altas demandas de tráfego de dados.

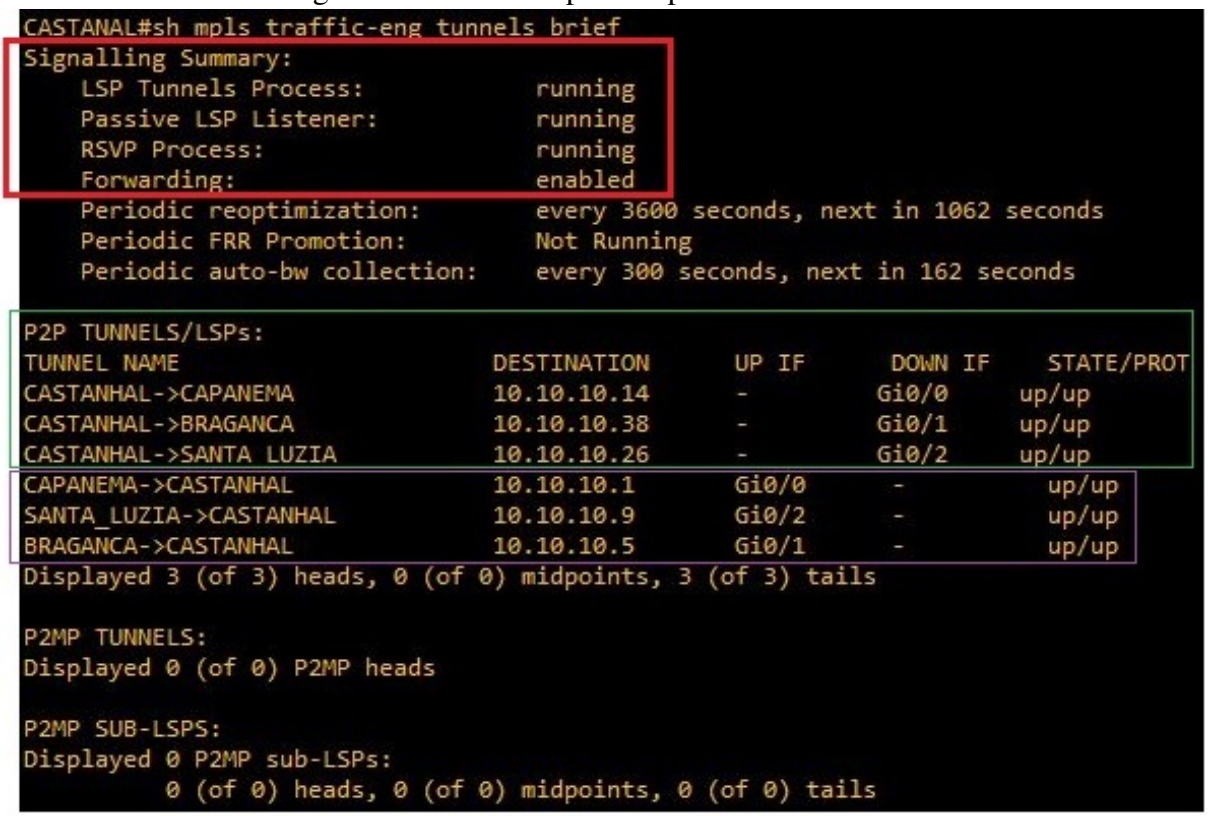
7.1 Avaliação do Primeiro Estudo de Caso

Dados os procedimentos descritos anteriormente, na subseção 6.1.1, voltada a apresentar o desenvolvimento do primeiro estudo, foi possível obter os resultados dos três túneis dedicados aos clientes da rede. A Figura 26 apresenta os túneis existentes na rede, de modo que ela foi obtida a partir do comando <*show mpls traffic-eng tunnels brief*>, executado no roteador CASTANHAL. Neste sentido, pode-se visualizar o resumo da sinalização destacado em vermelho, onde se consegue perceber que o processo de túneis de Caminho Comutado por Rótulos, ou do inglês, *Label Switched Path* (LSP), que é responsável pelo gerenciamento e manutenção dos túneis, está em execução.

Em conformidade a isso, é possível notar também que o ouvinte passivo do LSP, que permite ao roteador escutar e reconhecer túneis de TE estabelecidos por outros roteadores, bem como o RSVP, estão executando. Além disso, é notório que o roteador está encaminhando tráfego pelos túneis. Deste modo, em verde estão destacados os túneis que se originam em CASTANHAL e se destinam aos clientes, cujas interfaces de saída estão sendo apresentadas na coluna DOWN IF. Já em roxo, estão destacados os túneis que se originam nos equipamentos que conectam os clientes, que entram pelas mesmas interfaces usadas para a saída, como pode ser observado na coluna UP IF.

Dessa forma, a Figura 27 apresenta a saída do comando <*show mpls traffic-eng tunnels tunnel 22*>, na qual é possível visualizar o túnel explícito destinado à BRAGANÇA. Além disso, os parâmetros configurados anteriormente na Figura 13 estão destacados em azul e

Figura 26 – Túneis explícitos para os três clientes



Fonte: Elaborada pelo autor.

as informações destacadas em vermelho correspondem ao caminho mais curto. Neste trecho, são exibidos detalhes sobre uma rota explícita composta por saltos distintos daqueles configurados inicialmente. Isso ocorre pelo fato de que o protocolo OSPF determinou esse caminho como o de melhor esforço, resultando em um número menor de saltos até o endereço IP de destino. Vale salientar que essa é uma das premissas fundamentais para o TE, visto ser uma solução que contorna a problemática de caminhos ociosos dentro da rede, evitando, assim, o congestionamento do tráfego de dados nas interfaces de rede que descrevem sempre a trajetória mais curta até o nó de destino.

Posteriormente, foi feita uma captura de pacotes da interface GigabitEthernet 0/1 usando o aplicativo *Wireshark* (COMBS, 1998), conforme a Figura 28. Ele consiste em um analisador de pacotes gratuito e de código aberto, muito utilizado para solução de problemas de rede, análise, desenvolvimento de *software* e protocolos de comunicação, além de fins educacionais (BEALE *et al.*, 2006). Dito isto, ao observar a imagem é possível perceber a presença dos protocolos LDP e RSVP, sendo o primeiro responsável por enviar mensagens de sinalização, que indicam que a interface está com a função de troca rótulos habilitada, e o

Figura 27 – Túnel explicito de CASTANHAL com destino BRAGANÇA

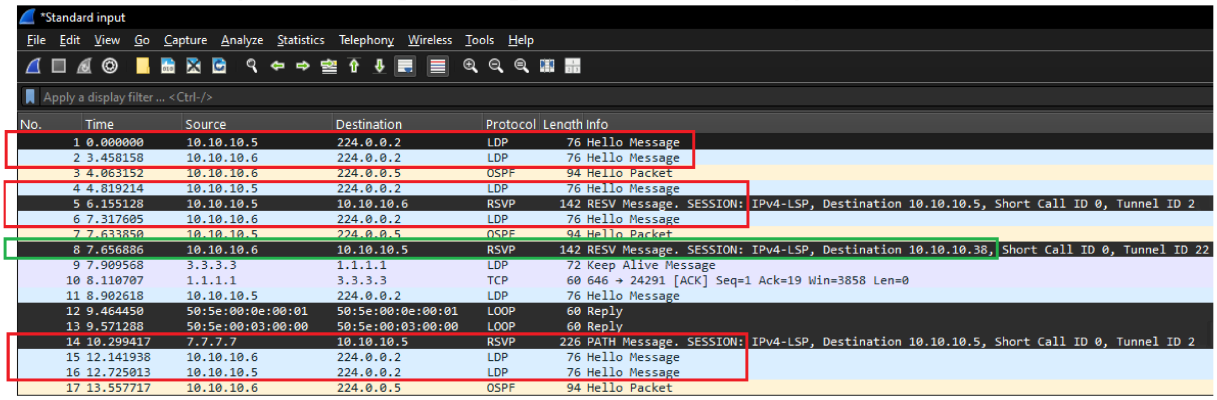
```
CASTANAL#sh mpls traffic-eng tunnels tunnel 22

Name: CASTANHAL->BRAGANCA                 (Tunnel22) Destination: 10.10.10.38
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 2, type explicit CASTANHAL->BRAGANCA (Basis for Setup, path weight 4)

  Config Parameters:
    Bandwidth: 10000    kbps (Global)  Priority: 1  1   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute announce: enabled  LockDown: disabled Loadshare: 10000 [200000] bw-based
    auto-bw: disabled
  Active Path Option Parameters:
    State: explicit path option 2 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled


  InLabel  :  -
  OutLabel : GigabitEthernet0/1, 20
  Next Hop : 10.10.10.6
  RSVP Signalling Info:
       Src 1.1.1.1, Dst 10.10.10.38, Tun_Id 22, Tun_Instance 18
    RSVP Path Info:
      My Address: 10.10.10.5
      Explicit Route: 10.10.10.6 10.10.10.41 10.10.10.18 10.10.10.38
                      7.7.7.7
      Record   Route:   NONE
      Tspec: ave rate=10000 kbits, burst=1000 bytes, peak rate=10000 kbits
    RSVP Resv Info:
      Record   Route:   NONE
      Fspec: ave rate=10000 kbits, burst=1000 bytes, peak rate=10000 kbits
  Shortest Unconstrained Path Info:
    Path Weight: 3 (TE)
    Explicit Route: 10.10.10.2 10.10.10.18 10.10.10.38 7.7.7.7
  History:
    Tunnel:
      Time since created: 1 hours, 50 minutes
      Time since path change: 1 hours, 49 minutes
      Number of LSP IDs (Tun_Instances) used: 18
    Current LSP: [ID: 18]
      Uptime: 1 hours, 49 minutes
```

Fonte: Elaborada pelo autor.

segundo, incumbido de fazer a validação da reserva de banda requisitada para o túnel de TE. Paralelamente, é destacada uma captura em verde, que corresponde à mensagem de reserva de recurso para o túnel com destino ao endereço de IP 10.10.10.38, ou seja, o túnel que se dirige à BRAGANÇA. Por outro lado, as demais capturas de pacote RSVP são do túnel originado de BRAGANÇA e destinadas a CASTANHAL, uma vez que o IP de destino é o 10.10.10.5, como se pode observar na imagem.

Considerando isso, a partir dos procedimentos demonstrados na subseção 6.1.1 e dos resultados expostos até o momento, foi possível perceber que o desenvolvimento de TE através do MPLS é demasiadamente complexo, visto serem necessárias configurações para a propagação

Figura 28 – Captura de pacote na interface GigabitEthernet 0/1



Fonte: Elaborada pelo autor.

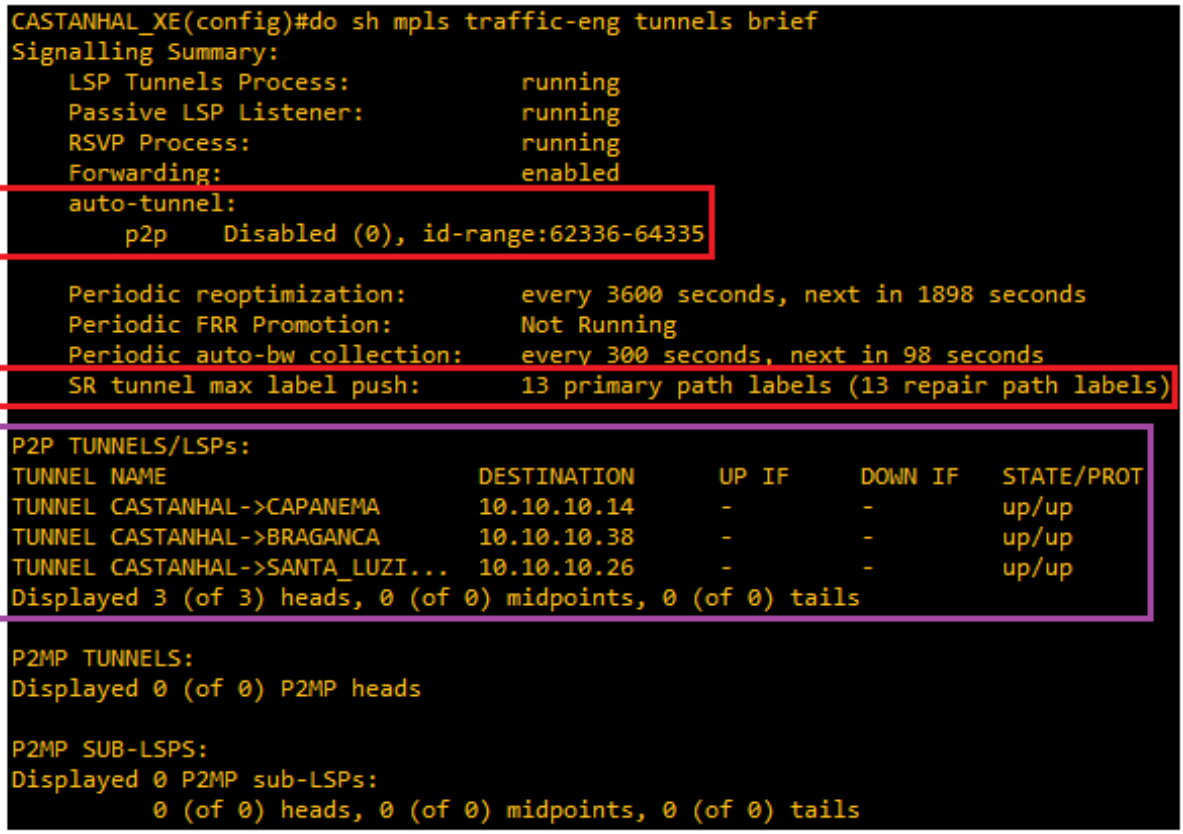
de rótulos pela rede que devem ser ativados em todas as interfaces internas da arquitetura de comunicação, bem como fazer uma alocação prévia de recursos de banda em cada enlace que descreve a trajetória do túnel a ser implementado. Isso resulta em uma quantidade elevada de dados de sinalização que consome recursos que poderiam ser utilizados para dados úteis, além de adicionar complexidade ao projeto. Desta maneira, foi estabelecido o objetivo de otimizar o esquema lógico da infraestrutura, fazendo uso do protocolo SR-BE que, como será indicado a seguir, dispensa o uso dos protocolos LDP e RSVP, de maneira a fazer a distribuição de rótulos usando o próprio protocolo IGP.

7.2 Avaliação do Segundo Estudo de Caso

Em conformidade à seção anterior, o comando <*show mpls traffic-eng tunnels brief*> foi executado no equipamento CASTANHAL, mas agora, fazendo referência à topologia apresentada na Figura 17. Deste modo, a Figura 29 apresenta a saída do comando, em que se consegue observar que na seção resumo de sinalização, os processos demonstrados são os mesmos da Figura 26, com exceção do *auto-tunnel*, o qual revela que túneis automáticos ponto a ponto (P2P) estão desativados e o *SR tunnel max label push*, que informa o número máximo de rótulos que podem ser empilhados no caminho principal e no caminho de reparo, de maneira a ser uma rota alternativa pré-calculada utilizada quando ocorre uma falha na rota principal.

Por outro lado, a seção de túneis ponto a ponto, destacada em roxo, expõe somente os túneis que são originados do equipamento CASTANHAL e que se destinam aos clientes, não retornando a informação dos túneis de ingresso, que são originados nos equipamentos que conectam os clientes. Além disso, nas colunas UP IF e DOWN IF, não são mostradas as

Figura 29 – Túneis explícitos com SR-BE para os três clientes



Fonte: Elaborada pelo autor.

interfaces físicas de entrada e saída dos túneis. Este fato será devidamente esclarecido com uma captura de pacotes, que será apresentada posteriormente.

Logo, assim como anteriormente, foi executado o comando <*show mpls traffic-eng tunnels tunnel 22*>, que possibilitou fazer a validação das descrições e dos parâmetros do túnel. Isto posto, a saída do comando é apresentada na Figura 30, de forma que, em azul, estão destacados os parâmetros que foram configurados no *script*. Nesse ínterim, constata-se que a saída obtida é muito próxima à saída da Figura 27, exceto pelo fato de não haver as informações de sinalização do protocolo RSVP. Contudo, faz-se mister apontar as informações destacadas em vermelho, que correspondem aos enlaces que descrevem a trajetória do túnel. Com isso, pode-se perceber que o SR atribuiu um rótulo a cada um deles e, assim, para a rede que conecta CASTANHAL e NOVA-TIMBOTEUA, foi alocado o rótulo 20. Do mesmo modo, para a conexão entre NOVA-TIMBOTEUA e IGARAPE-ACU, o rótulo 16 e assim sucessivamente, até à rede prefixo de destino.

Deste modo, assim como anteriormente, foi feita uma captura de pacote da interface Gi2 do roteador CASTANHAL, sendo ela apresentada na Figura 31. Nela é possível perceber

Figura 30 – Túnel SR explicito de CASTANHAL com destino BRAGANÇA

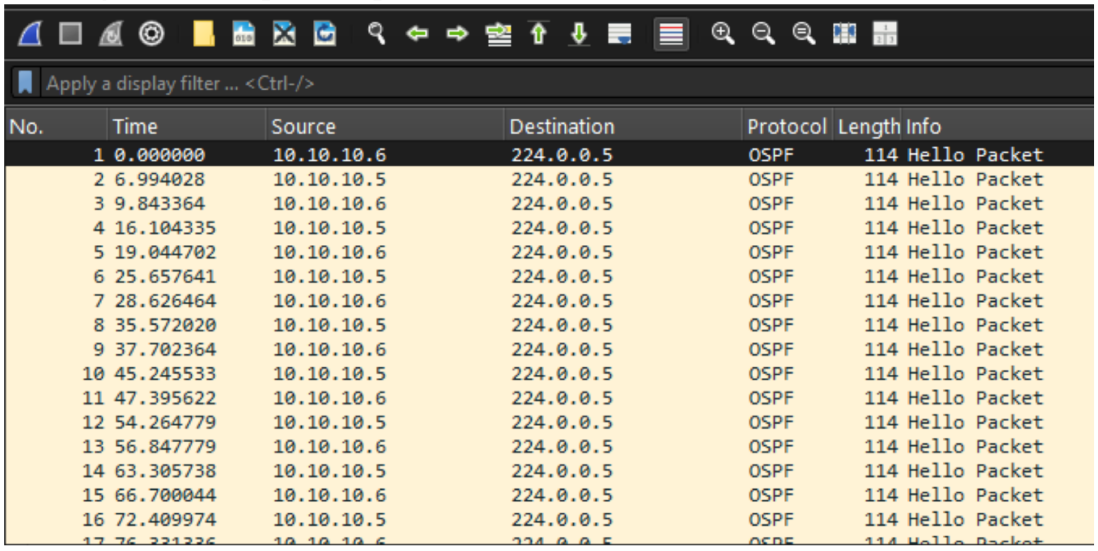
```
CASTANHAL_XE#show mpls traffic-eng tunnels tunnel 22

Name: TUNNEL CASTANHAL->BRAGANCA          (Tunnel22) Destination: 10.10.10.38
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 1, (SEGMENT-ROUTING) type explicit CASTANHAL->BRAGANCA (Basis for Setup, path weight 4)

  Config Parameters:
    Bandwidth: 10000    kbps (Global)  Priority: 1  1   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    Path Selection:
     Protection: any (default)
    Path-selection Tiebreaker:
      Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
    Hop Limit: disabled [ignore: Explicit Path Option with all Strict Hops]
    Cost Limit: disabled
    Path-invalidation timeout: 10000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10000 [200000] bw-based
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled

  History:
    Tunnel:
      Time since created: 4 hours, 4 minutes
      Time since path change: 4 hours, 4 minutes
      Number of LSP IDs (Tun_Instances) used: 9
    Current LSP: [ID: 9]
      Uptime: 4 hours, 4 minutes
  Tun Instance: 9
  Segment-Routing Path Info (ospf 1  area 0)
    Segment0[Link]: 10.10.10.5 - 10.10.10.6, Label: 20
    Segment1[Link]: 10.10.10.42 - 10.10.10.41, Label: 16
    Segment2[Link]: 10.10.10.17 - 10.10.10.18, Label: 18
    Segment3[Link]: 10.10.10.37 - 10.10.10.38, Label: 19
```

Fonte: Elaborada pelo autor.

a existência de somente *Hello Packet*, que são dados de sinalização que o OSPF envia para as adjacências, informando que a respectiva interface está ativa. Desta maneira, não há dados de sinalização que indiquem a existência de túneis de TE nas imediações. Sendo assim, este fato explica o motivo de não haver interfaces físicas nas colunas UP IF e DOWN IF, na seção de túneis ponto a ponto da Figura 29, pois para o caso dos túneis do primeiro estudo de caso, o protocolo RSVP envia dados de sinalização, validando a reserva de banda para os túneis, fato que não ocorre no presente caso de estudo.

Por fim, foi feita uma nova captura, agora executada por meio do comando <*traceroute 7.7.7.7*>, no qual este prefixo de IP pertence à interface *Loopback* 0 do roteador BRAGANÇA. Em face disso, ela é apresentada na Figura 32, de modo que, neste caso, o que se propõe é enviar pacotes, de tal modo que eles percorram o túnel. Com isso, quando é feita a validação do conteúdo do primeiro dado que faz uso do Protocolo de Datagrama do Usuário, ou do inglês, *User Datagram Protocol* (UDP), é possível notar a presença de cabeçalhos MPLS,

Figura 31 – Captura de pacotes da interface Gi2 do roteador CASTANHAL



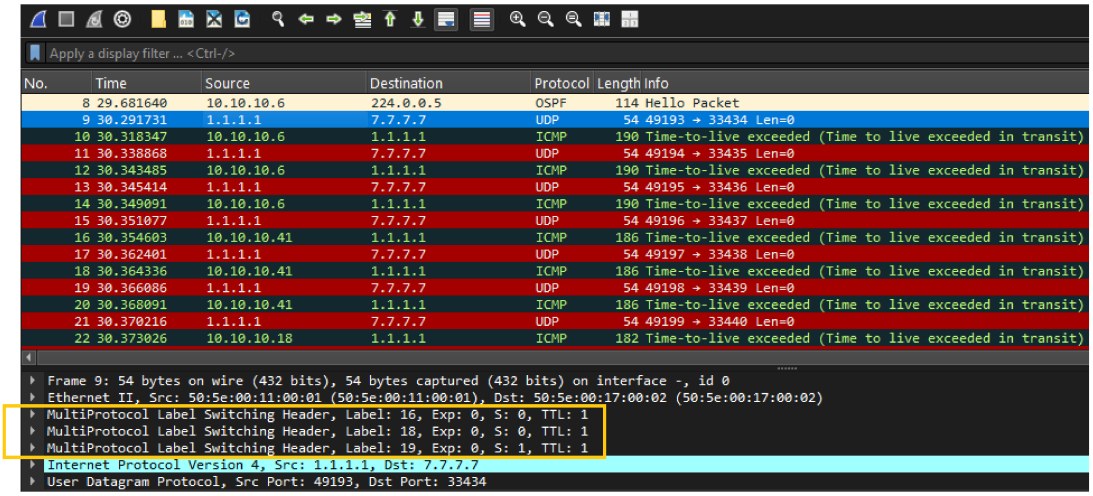
Fonte: Elaborada pelo autor.

cujos rótulos coincidem com os Figura 24, exceto o 20, uma vez que ele pertence à interface na qual foi feita a captura.

Logo, a partir dos procedimentos descritos ao longo desta seção, é notória a superioridade do SR-BE com relação ao MPLS tradicional, uma vez que ele se apresenta como uma tecnologia de configuração simplificada, independente de protocolos de propagação de rótulos e validação de recurso alocado para os túneis, fazendo assim, com que a quantidade de dados de sinalização sejam diminuídas na rede, permitindo uma manutenção sintetizada e de fácil compreensão.

Contudo, é possível perceber que, mesmo dispensando os protocolos LDP e RSVP, a necessidade da criação de interfaces de túnel promove dificuldades de escalabilidade no projeto.

Figura 32 – Captura de pacotes da interface Gi2 com o comando <*traceroute*>



Fonte: Elaborada pelo autor.

Isso se deve pelo fato de que, mesmo sem requerer dados de sinalização de vizinhanças e de reserva de banda, cada túnel exige a manutenção de entradas de encaminhamento, uma vez que cada túnel precisa ser computado na tabela de encaminhamento do roteador de entrada e, informações de estado no roteador de entrada, que requer que o roteador de entrada processe e empilhe os SIDs para cada pacote. Dessa maneira, com o aumento do número de túneis, a complexidade do gerenciamento cresce significativamente, tornando a solução menos eficiente em redes de larga escala.

Desta maneira, foi fundamental conceber uma solução mais sofisticada e com um maior grau de escalabilidade e flexibilidade: a tecnologia SR-TE *policy*. Dado que, em vez de usar túneis lógicos pela arquitetura, com a finalidade de manipular o tráfego, ela se baseia somente nos SIDs, fazendo assim, o redirecionamento do tráfego ocorrer de maneira mais inteligente.

7.3 Avaliação do Terceiro Estudo de Caso

Dados os procedimentos descritos na seção 6.1.3, consolidando assim, a constituição do SR-TE *policy*, serão discutidos agora os resultados de sua implementação. Neste aspecto, destacada em vermelho na Figura 33, está sendo apresentada a saída do comando <*show segment-routing traffic-eng policy candidate-path name VPNA_CORE*>, em que é possível perceber que foram retornados os parâmetros configurados, conforme o *script* da Figura 24. Outrossim, a política recebeu de forma automática o nome *srte_c_100_ep_3.3.3.3*, que tem correlação com os parâmetros ativados nela. Por fim, em verde no campo de atributos, está sendo ressaltado o *Binding* SID, que é um identificador único associado à SR-TE *policy*, recebeu o rótulo 24005. Logo, qualquer dado que possuir o rótulo 24005, seguirá associado à política VPNA_CORE.

Com isso, foi feita uma captura de pacotes da interface Gi0/0/0/0, como está sendo apresentada na Figura 34, que contém a presença de dados de pacotes IS-IS HELLO, os quais são transmitidos com a finalidade de manter as vizinhanças no roteamento. Já os IS-IS CSNP e LSP, que correspondem ao pacote de número de sequência completo e ao protocolo de estado de enlace, respectivamente, fornecem informações sobre a base de dados do roteamento e sobre a topologia da rede, nesta ordem. Por fim, estão presentes também pacotes com os protocolos TCP e BGP, que validam a seção iBGP entre PE1 e PE2.

Logo, constatou-se a obtenção da manipulação de tráfego desejada, em que todos os pacotes cujo o endereço de destino é o prefixo 22.22.22.22/32, seguem o caminho descrito

Figura 33 – Visualização da SR-TE *policy*

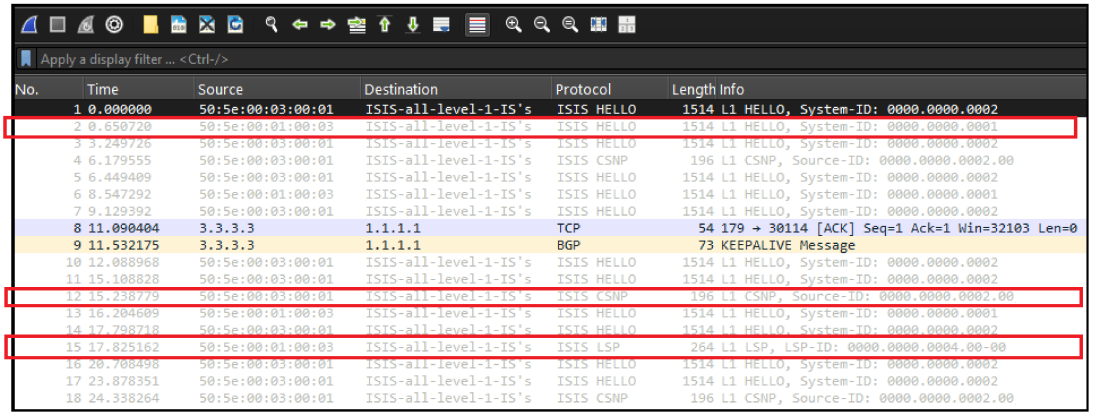
```
RP/0/RP0/CPU0:PE1#sh segment-routing traffic-eng policy candidate-path name VP$
Tue Feb 11 02:46:38.701 UTC

SR-TE policy database
---------------------

Color: 100, End-point: 3.3.3.3
  Name: srte_c_100_ep_3.3.3.3
  Status:
    Admin: up  Operational: up for 13:04:15 (since Feb 10 13:42:22.985)
  Candidate-paths:
    Preference: 100 (configuration) (active)
      Name: VPNA_CORE
      Requested BSID: dynamic
      Explicit: segment-list VPNA_CORE (valid)
        Weight: 1, Metric Type: TE
          16002
          16004
          16003
  Attributes:
    Binding SID: 24005
    Forward Class: Not Configured
    Steering labeled-services disabled: no
    Steering BGP disabled: no
    IPv6 caps enable: yes
```

Fonte: Elaborada pelo autor.

na Figura 23. Assim, a Figura 35 apresenta a execução dos comandos <*trace 22.22.22.22*> e <*trace 33.33.33.33*> no VPC. Desta forma, é possível perceber que os pacotes destinados a 22.22.22.22/32 seguem o caminho determinado pela lista de segmentos. Já para o caso dos pacotes gerados para 33.33.33.33, eles seguem por um caminho diferente, que foi determinado como o melhor pelo protocolo IGP, sendo ele descrito como CE1 $\rightarrow$ PE1 $\rightarrow$ P2 $\rightarrow$ PE2 $\rightarrow$ CE2.

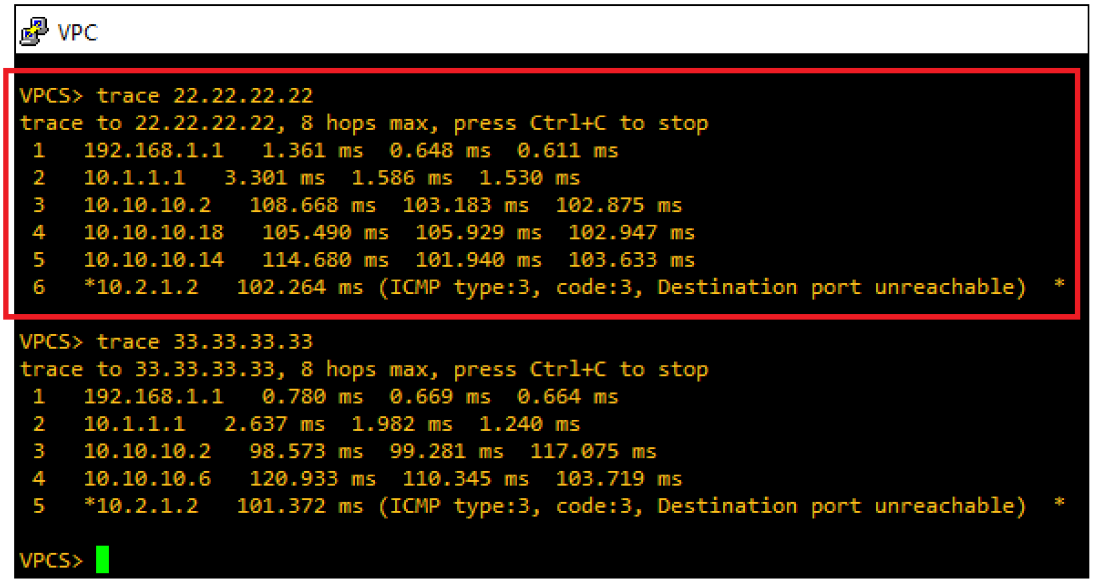
Nesse contexto, verificou-se a superioridade do SR, ao passo que se apresenta como um protocolo com maior escalabilidade, porquanto dispensa a necessidade da criação de interfaces de túnel, que necessitam dos protocolos LDP e RSVP. Além disso, possuem também uma maior flexibilidade, uma vez que podem ser implementadas políticas com redirecionamento de tráfego que permitem ser facilmente associadas com *Extended Community* do BGP, que consiste

Figura 34 – Captura de pacotes da interface física Gi0/0/0/0



Fonte: Elaborada pelo autor.

Figura 35 – Execução do comando <*trace*> para os prefixos 22.22.22.22 e 33.33.33.33



Fonte: Elaborada pelo autor.

em uma atributo que fornece um mecanismo para rotulagem das informações transportadas pelo protocolo (SANGLI *et al.*, 2006), que para o caso em questão, se baseia nas *colors*. Vale destacar que o SR pode coexistir com o MPLS, pois ambos fazem uso do mesmo plano de dados nas arquiteturas modernas dos equipamentos. Dessa forma, isso possibilita uma migração suave para as redes de comunicação, que podem implementar a tecnologia nas regiões com maior demanda de tráfego de dados, bem como manter o projeto inicial em áreas menos críticas.

8 CONCLUSÕES E TRABALHOS FUTUROS

Haja vista tudo que foi discutido até o presente momento, este trabalho teve como objetivo desenvolver uma discussão acerca das tecnologias MPLS e SR para aplicações de TE, com a finalidade de demonstrar a superioridade do SR em detrimento do anterior. Além disso, trata-se de um protocolo moderno e escalável, que não requer interfaces túnel nem protocolos de sinalização para validar a disponibilidade de recursos espectrais ou propagação de rótulos. Essa característica permite um projeto mais simples e flexível. O paradigma de roteamento de origem possibilita a manipulação do tráfego com base em políticas de marcação, que podem ser integradas ao protocolo BGP.

Nesta perspectiva, a discussão foi desenvolvida a partir de três estudos de caso, em que no primeiro, foi possível observar a rigidez do protocolo MPLS, que demanda ativação de interfaces túnel, que dependem dos protocolos LDP e RSVP. Já no segundo estudo de caso, foi desenvolvida a mesma solução do primeiro, mas com o uso do SR-BE, em que se percebe a diminuição dos dados de sinalização pela rede. Por fim, no terceiro estudo de caso, foi demonstrada a flexibilidade do SR a partir do seu recurso de marcação do tráfego por *colors*, o que possibilita um redirecionamento de dados a partir de suas características.

Portanto, conseguiu-se demonstrar que o SR se apresenta como um protocolo que suporta as novas necessidades das redes modernas, que demandam cada vez mais recursos de taxa de dados, devido à implementação de tecnologias 5G, IoT e realidade aumentada. Assim, o que se espera é que este trabalho auxilie profissionais e pesquisadores a elaborarem projetos de implantação do protocolo em suas redes de telecomunicações, de modo a colaborar com o futuro das comunicações no Brasil e no mundo.

8.1 Sugestões de Trabalhos Futuros

Como proposta de trabalhos futuros, pode-se citar os casos de comparação entre os protocolos Borda de Provedor IPv6 sobre MPLS, ou do inglês, *IPv6 Provider Edge over MPLS* (6PE), que é a solução para integrar IPv6 em redes MPLS, e o SRv6, de forma a comparar a escalabilidade e flexibilidade de ambos os casos. Outrossim, uma outra proposta seria elaborar uma discussão entre as técnicas FRR e Alternativa Livre de Loop Independente de Topologia, ou do inglês, *Topology Independent Loop-Free Alternate* (TI-LFA), que são técnicas utilizadas para melhorar a convergência do roteamento de redes de comunicação, fazendo assim, uma

comparação sobre o desempenho das tecnologias, de forma a definir qual é a mais benéfica. Por fim, também é viável explorar sinergias entre SR e SDN, investigando como essas duas abordagens podem ser integradas para melhorar a flexibilidade e o controle da rede.

REFERÊNCIAS

ADILSON, N. B. F. **Crie laboratórios Online, tenha a liberdade de estudar de qualquer lugar, sem precisar gastar com hardware caro. Invista no que é mais valioso)**. 2025. Disponível em: <https://networklabsbrasil.com.br/>. Acesso em: 27 fev. 2025.

ANDERSSON, L.; MINEI, I.; THOMAS, B. **RFC 5036: LDP Specification**. [S.l.]: RFC Editor, 2007.

BANUPRIYA, N.; NAGARAJAN, N. *et al.* Performance breakdown and redistribution amidst ospf, eigrp & is-is dynamic routing protocols in ipv6 network. In: IEEE. **2022 Smart Technologies, Communication and Robotics (STCR)**. [S.l.], 2022. p. 1–5.

BEALE, J.; OREBAUGH, A.; RAMIREZ, G. **Wireshark & Ethereal network protocol analyzer toolkit**. [S.l.]: Elsevier, 2006.

CAI, D.; WIELOSZ, A.; WEI, S. Evolve carrier ethernet architecture with sdn and segment routing. In: IEEE. **Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014**. [S.l.], 2014. p. 1–6.

CARVALHO, B. de. Algoritmo de dijkstra. **Universidade de Coimbra, Coimbra, Portugal**, 2008.

CH, G. D. S.; NARANJO, E. F.; MARRONE, L. Sdn-ready wan networks: Segment routing in mpls-based environments. In: IEEE. **2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)**. [S.l.], 2018. p. 173–178.

CHICA, J. C. C.; IMBACHI, J. C.; VEGA, J. F. B. Security in sdn: A comprehensive survey. **Journal of Network and Computer Applications**, Elsevier, v. 159, p. 102595, 2020.

CISCO. **Implementar QoS no Cisco SD-WAN**. 2018. Disponível em: <https://www.cisco.com/c/pt_br/support/docs/routers/vedge-router/213408-implement-qos-in-cisco-sd-wan.html>. Acesso em: 15 jun. 2019.

COMBS, G. **The world's most popular network protocol analyzer**. 1998. Disponível em: <https://www.wireshark.org/#downloadLinkain>. Acesso em: 27 fev. 2025.

DOYLE, J. **Routing TCP/IP: CCIE Professional Development, Volume 2**. [S.l.]: Cisco Press, 2016.

EVE-NG. **EVE - Emulated Virtual Enviroment - Next Generation**. 2025. Disponível em: <https://www.eve-ng.net/>. Acesso em: 27 fev. 2025.

FARREL, A. **The Internet and its protocols: A comparative approach**. [S.l.]: Elsevier, 2004.

FILSFILS, C.; PREVIDI, S.; BASHANDY, A.; DECRAENE, B.; LITKOWSKI, S.; HORNEFFER, M.; SHAKIR, R.; TANTSURA, J.; CRABBE, E. Segment routing with mpls data plane. **draft-ietf-spring-segment-routing-mpls-05**, 2014.

FILSFILS, C.; PREVIDI, S.; GINSBERG, L.; DECRAENE, B.; LITKOWSKI, S.; SHAKIR, R. **Segment routing architecture**. [S.l.], 2018.

FOROUZAN, B. A.; FEGAN, S. C. **Protocolo TCP/IP-3**. [S.l.]: AMGH Editora, 2009.

FUNDATION, O. N. Software-defined networking: The new norm for networks. **ONF white paper**, v. 2, n. 2-6, p. 11, 2012.

GALVÃO, J. S. d. S. **Segment routing: roteamento de tráfego baseado na origem**. Dissertação (B.S. thesis), 2023.

GINSBERG, L.; DECRAENE, B.; LITKOWSKI, S.; SHAKIR, R. **RFC 8402: Segment routing architecture**. [S.l.]: RFC Editor, 2018.

GUIMARÃES, J. V. M. *et al.* Estudo comparativo dos protocolos de roteamento rip e ospf usando o simulador cisco packet tracer. Universidade Federal de Uberlândia, 2021.

HAO, F.; KODIALAM, M.; LAKSHMAN, T. Optimizing restoration with segment routing. In: IEEE. **IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications**. [S.l.], 2016. p. 1–9.

HUAWEI. **Example for Configuring L3VPNv4 over SRv6 TE Policy Group (Manual Configuration)**. 2024. Disponível em: <https://support.huawei.com/hedex/hdx.do?docid=EDOC1100168821&id=EN-US_TASK_0252010694>. Acesso em: 03 fev. 2025.

JUNIOR, C. E. P. d. O. *et al.* Engenharia de tráfego aplicado à simulação de uma rede backbone de um provedor de internet regional utilizando o protocolo mpls te. 2022.

KOS, A. Segment routing principles and applications for sdn. Politecnico di Milano, 2014.

KUKREJA, N.; ALVIZU, R.; KOS, A.; MAIER, G.; MORRO, R.; CAPELLO, A.; CAVAZZONI, C. Demonstration of sdn-based orchestration for multi-domain segment routing networks. In: IEEE. **2016 18th International Conference on Transparent Optical Networks (ICTON)**. [S.l.], 2016. p. 1–4.

KUROSE, J. F.; ROSS, K. W. Redes de computadores e a internet. **São Paulo: Person**, v. 28, 2006.

LI, T.; CHANDRA, R.; TRAINA, P. S. **BGP Communities Attribute**. RFC Editor, 1996. RFC 1997. (Request for Comments, 1997). Disponível em: <https://www.rfc-editor.org/info/rfc1997>.

MAILA, G.; MARIUS, I.; VICTOR, C. Segment routing. In: IEEE. **2017 10th International Symposium on Advanced Topics in Electrical Engineering (ATEE)**. [S.l.], 2017. p. 34–38.

MARTEY, A. **IS-IS network design solutions**. [S.l.]: Cisco Press, 2002.

MENDONÇA, R.; OLIVEIRA, J. M.; LINS, R. D. **Redes MPLS: fundamentos e aplicações**. [S.l.]: Brasport, 2012.

MENDONÇA ROBERTO, e. C. **Segment Routing**. 2024.

MON, O. M.; MON, M. T. Quality of service sensitive routing for software defined network using segment routing. In: IEEE. **2018 18th International Symposium on Communications and Information Technologies (ISCIT)**. [S.l.], 2018. p. 180–185.

MORENO, E.; BEGHELLI, A.; CUGINI, F. Traffic engineering in segment routing networks. **Computer Networks**, Elsevier, v. 114, p. 23–31, 2017.

NANDA, P. Supporting qos guarantees using traffic engineering and policy based routing. In: IEEE. **2008 International Conference on Computer Science and Software Engineering**. [S.l.], 2008. v. 3, p. 137–142.

NEVES, J. S. D.; TORRES, W. R. **O Protocolo OSPF**. 2017.

NUNES, B. A. A.; MENDONCA, M.; NGUYEN, X.-N.; OBRACZKA, K.; TURLETTI, T. A survey of software-defined networking: Past, present, and future of programmable networks. **IEEE Communications surveys & tutorials**, IEEE, v. 16, n. 3, p. 1617–1634, 2014.

PORWAL, M. K.; YADAV, A.; CHARHATE, S. Traffic analysis of mpls and non mpls network including mpls signaling protocols and traffic distribution in ospf and mpls. In: IEEE. **2008 First International Conference on Emerging Trends in Engineering and Technology**. [S.l.], 2008. p. 187–192.

ROSEN, E.; VISWANATHAN, A.; CALLON, R. **Multiprotocol label switching architecture**. [S.l.], 2001.

SANGLI, S.; TAPPAN, D.; REKHTER, Y. **RFC 4360: BGP Extended Communities Attribute**. [S.l.]: RFC Editor, 2006.

SCHARF, A. L. Implantação de engenharia de tráfego com mpls-te em rede wan. 2017.

SINGH, A. K.; SRIVASTAVA, S. A survey and classification of controller placement problem in sdn. **International Journal of Network Management**, Wiley Online Library, v. 28, n. 3, p. e2018, 2018.

TAPPAN, D.; SANGLI, S. R.; REKHTER, Y. **BGP Extended Communities Attribute**. RFC Editor, 2006. RFC 4360. (Request for Comments, 4360). Disponível em: <https://www.rfc-editor.org/info/rfc4360>.

TROIA, S.; SAPIENZA, F.; VARÉ, L.; MAIER, G. On deep reinforcement learning for traffic engineering in sd-wan. **IEEE Journal on Selected Areas in Communications**, IEEE, v. 39, n. 7, p. 2198–2212, 2020.

VENTRE, P. L.; SALSANO, S.; POLVERINI, M.; CIANFRANI, A.; ABDELSALAM, A.; FILSFILS, C.; CAMARILLO, P.; CLAD, F. Segment routing: A comprehensive survey of research activities, standardization efforts, and implementation results. **IEEE Communications Surveys & Tutorials**, IEEE, v. 23, n. 1, p. 182–221, 2020.

YALDA, K. G.; HAMAD, D. J.; ŢĂPUŞ, N. A survey on software-defined wide area network (sd-wan) architectures. In: IEEE. **2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)**. [S.l.], 2022. p. 1–5.

YEFAN, L. **O que é SDN**. 2021. Disponível em: <https://info.support.huawei.com/info-finder/encyclopedia/en/SDN.html>. Acesso em: 29 jan. 2024.

REFERÊNCIAS

ADILSON, N. B. F. **Crie laboratórios Online, tenha a liberdade de estudar de qualquer lugar, sem precisar gastar com hardware caro. Invista no que é mais valioso)**. 2025. Disponível em: <https://networklabsbrasil.com.br/>. Acesso em: 27 fev. 2025.

ANDERSSON, L.; MINEI, I.; THOMAS, B. **RFC 5036: LDP Specification**. [S.l.]: RFC Editor, 2007.

BANUPRIYA, N.; NAGARAJAN, N. *et al.* Performance breakdown and redistribution amidst ospf, eigrp & is-is dynamic routing protocols in ipv6 network. In: IEEE. **2022 Smart Technologies, Communication and Robotics (STCR)**. [S.l.], 2022. p. 1–5.

BEALE, J.; OREBAUGH, A.; RAMIREZ, G. **Wireshark & Ethereal network protocol analyzer toolkit**. [S.l.]: Elsevier, 2006.

CAI, D.; WIELOSZ, A.; WEI, S. Evolve carrier ethernet architecture with sdn and segment routing. In: IEEE. **Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014**. [S.l.], 2014. p. 1–6.

CARVALHO, B. de. Algoritmo de dijkstra. **Universidade de Coimbra, Coimbra, Portugal**, 2008.

CH, G. D. S.; NARANJO, E. F.; MARRONE, L. Sdn-ready wan networks: Segment routing in mpls-based environments. In: IEEE. **2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)**. [S.l.], 2018. p. 173–178.

CHICA, J. C. C.; IMBACHI, J. C.; VEGA, J. F. B. Security in sdn: A comprehensive survey. **Journal of Network and Computer Applications**, Elsevier, v. 159, p. 102595, 2020.

CISCO. **Implementar QoS no Cisco SD-WAN**. 2018. Disponível em: <https://www.cisco.com/c/pt_br/support/docs/routers/vedge-router/213408-implement-qos-in-cisco-sd-wan.html>. Acesso em: 15 jun. 2019.

COMBS, G. **The world's most popular network protocol analyzer**. 1998. Disponível em: <https://www.wireshark.org/#downloadLinkain>. Acesso em: 27 fev. 2025.

DOYLE, J. **Routing TCP/IP: CCIE Professional Development, Volume 2**. [S.l.]: Cisco Press, 2016.

EVE-NG. **EVE - Emulated Virtual Enviroment - Next Generation**. 2025. Disponível em: <https://www.eve-ng.net/>. Acesso em: 27 fev. 2025.

FARREL, A. **The Internet and its protocols: A comparative approach**. [S.l.]: Elsevier, 2004.

FILSFILS, C.; PREVIDI, S.; BASHANDY, A.; DECRAENE, B.; LITKOWSKI, S.; HORNEFFER, M.; SHAKIR, R.; TANTSURA, J.; CRABBE, E. Segment routing with mpls data plane. **draft-ietf-spring-segment-routing-mpls-05**, 2014.

FILSFILS, C.; PREVIDI, S.; GINSBERG, L.; DECRAENE, B.; LITKOWSKI, S.; SHAKIR, R. **Segment routing architecture**. [S.l.], 2018.

FOROUZAN, B. A.; FEGAN, S. C. **Protocolo TCP/IP-3**. [S.l.]: AMGH Editora, 2009.

FUNDATION, O. N. Software-defined networking: The new norm for networks. **ONF white paper**, v. 2, n. 2-6, p. 11, 2012.

GALVÃO, J. S. d. S. **Segment routing: roteamento de tráfego baseado na origem**. Dissertação (B.S. thesis), 2023.

GINSBERG, L.; DECRAENE, B.; LITKOWSKI, S.; SHAKIR, R. **RFC 8402: Segment routing architecture**. [S.l.]: RFC Editor, 2018.

GUIMARÃES, J. V. M. *et al.* Estudo comparativo dos protocolos de roteamento rip e ospf usando o simulador cisco packet tracer. Universidade Federal de Uberlândia, 2021.

HAO, F.; KODIALAM, M.; LAKSHMAN, T. Optimizing restoration with segment routing. In: IEEE. **IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications**. [S.l.], 2016. p. 1–9.

HUAWEI. **Example for Configuring L3VPNv4 over SRv6 TE Policy Group (Manual Configuration)**. 2024. Disponível em: <https://support.huawei.com/hedex/hdx.do?docid=EDOC1100168821&id=EN-US_TASK_0252010694>. Acesso em: 03 fev. 2025.

JUNIOR, C. E. P. d. O. *et al.* Engenharia de tráfego aplicado à simulação de uma rede backbone de um provedor de internet regional utilizando o protocolo mpls te. 2022.

KOS, A. Segment routing principles and applications for sdn. Politecnico di Milano, 2014.

KUKREJA, N.; ALVIZU, R.; KOS, A.; MAIER, G.; MORRO, R.; CAPELLO, A.; CAVAZZONI, C. Demonstration of sdn-based orchestration for multi-domain segment routing networks. In: IEEE. **2016 18th International Conference on Transparent Optical Networks (ICTON)**. [S.l.], 2016. p. 1–4.

KUROSE, J. F.; ROSS, K. W. Redes de computadores e a internet. **São Paulo: Person**, v. 28, 2006.

LI, T.; CHANDRA, R.; TRAINA, P. S. **BGP Communities Attribute**. RFC Editor, 1996. RFC 1997. (Request for Comments, 1997). Disponível em: <https://www.rfc-editor.org/info/rfc1997>.

MAILA, G.; MARIUS, I.; VICTOR, C. Segment routing. In: IEEE. **2017 10th International Symposium on Advanced Topics in Electrical Engineering (ATEE)**. [S.l.], 2017. p. 34–38.

MARTEY, A. **IS-IS network design solutions**. [S.l.]: Cisco Press, 2002.

MENDONÇA, R.; OLIVEIRA, J. M.; LINS, R. D. **Redes MPLS: fundamentos e aplicações**. [S.l.]: Brasport, 2012.

MENDONÇA ROBERTO, e. C. **Segment Routing**. 2024.

MON, O. M.; MON, M. T. Quality of service sensitive routing for software defined network using segment routing. In: IEEE. **2018 18th International Symposium on Communications and Information Technologies (ISCIT)**. [S.l.], 2018. p. 180–185.

MORENO, E.; BEGHELLI, A.; CUGINI, F. Traffic engineering in segment routing networks. **Computer Networks**, Elsevier, v. 114, p. 23–31, 2017.

NANDA, P. Supporting qos guarantees using traffic engineering and policy based routing. In: IEEE. **2008 International Conference on Computer Science and Software Engineering**. [S.l.], 2008. v. 3, p. 137–142.

NEVES, J. S. D.; TORRES, W. R. **O Protocolo OSPF**. 2017.

NUNES, B. A. A.; MENDONCA, M.; NGUYEN, X.-N.; OBRACZKA, K.; TURLETTI, T. A survey of software-defined networking: Past, present, and future of programmable networks. **IEEE Communications surveys & tutorials**, IEEE, v. 16, n. 3, p. 1617–1634, 2014.

PORWAL, M. K.; YADAV, A.; CHARHATE, S. Traffic analysis of mpls and non mpls network including mpls signaling protocols and traffic distribution in ospf and mpls. In: IEEE. **2008 First International Conference on Emerging Trends in Engineering and Technology**. [S.l.], 2008. p. 187–192.

ROSEN, E.; VISWANATHAN, A.; CALLON, R. **Multiprotocol label switching architecture**. [S.l.], 2001.

SANGLI, S.; TAPPAN, D.; REKHTER, Y. **RFC 4360: BGP Extended Communities Attribute**. [S.l.]: RFC Editor, 2006.

SCHARF, A. L. Implantação de engenharia de tráfego com mpls-te em rede wan. 2017.

SINGH, A. K.; SRIVASTAVA, S. A survey and classification of controller placement problem in sdn. **International Journal of Network Management**, Wiley Online Library, v. 28, n. 3, p. e2018, 2018.

TAPPAN, D.; SANGLI, S. R.; REKHTER, Y. **BGP Extended Communities Attribute**. RFC Editor, 2006. RFC 4360. (Request for Comments, 4360). Disponível em: <https://www.rfc-editor.org/info/rfc4360>.

TROIA, S.; SAPIENZA, F.; VARÉ, L.; MAIER, G. On deep reinforcement learning for traffic engineering in sd-wan. **IEEE Journal on Selected Areas in Communications**, IEEE, v. 39, n. 7, p. 2198–2212, 2020.

VENTRE, P. L.; SALSANO, S.; POLVERINI, M.; CIANFRANI, A.; ABDELSALAM, A.; FILSFILS, C.; CAMARILLO, P.; CLAD, F. Segment routing: A comprehensive survey of research activities, standardization efforts, and implementation results. **IEEE Communications Surveys & Tutorials**, IEEE, v. 23, n. 1, p. 182–221, 2020.

YALDA, K. G.; HAMAD, D. J.; ŢĂPUŞ, N. A survey on software-defined wide area network (sd-wan) architectures. In: IEEE. **2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)**. [S.l.], 2022. p. 1–5.

YEFAN, L. **O que é SDN**. 2021. Disponível em: <https://info.support.huawei.com/info-finder/encyclopedia/en/SDN.html>. Acesso em: 29 jan. 2024.

APÊNDICE A – CONFIGURAÇÕES BÁSICAS DOS ROTEADORES PARA O PRIMEIRO E O SEGUNDO ESTUDO DE CASO

Figura A.1 – Configuração do *Port-Channel* no *switch* CASTANHAL

```
hostname CASTANHAL
!
interface Port-channel1
 no switchport
 ip address 10.10.10.49 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0.0.0.0
!
interface GigabitEthernet0/3
 no switchport
 no shut
 no ip address
 channel-group 1 mode active
!
interface GigabitEthernet1/0
 no switchport
 no shut
 no ip address
 channel-group 1 mode active
!
interface GigabitEthernet1/1
 no switchport
 no shut
 no ip address
 channel-group 1 mode active
!
```

Fonte: Elaborado pelo autor.

Figura A.2 – Configurações básicas do *switch* CASTANHAL

```
hostname CASTANHAL
!
interface Loopback0
 ip address 1.1.1.1 255.255.255.255
 ip ospf 1 area 0.0.0.0
!
interface GigabitEthernet0/0
 no shut
 no switchport
 ip address 10.10.10.1 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0.0.0.0
!
interface GigabitEthernet0/1
 no switchport
 no shut
 ip address 10.10.10.5 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0.0.0.0
!
interface GigabitEthernet0/2
 no switchport
 no shut
 ip address 10.10.10.9 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0.0.0.0
!
interface GigabitEthernet1/2
 no shut
 no switchport
 ip address 10.10.10.53 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0.0.0.0
!
router ospf 1
 router-id 1.1.1.1
 passive-interface Loopback0
!
end
```

Fonte: Elaborado pelo autor.

Figura A.3 – Configurações básicas do roteador CAPANEMA

```
hostname CAPANEMA
!
ip dhcp pool cliente1
 network 10.0.0.0 255.255.255.0
 default-router 10.0.0.1
!
interface Loopback0
 ip address 5.5.5.5 255.255.255.255
 ip ospf 1 area 0
!
interface GigabitEthernet0/0
 ip address 10.10.10.14 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/1
 ip address 10.10.10.33 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/2
 ip address 10.10.10.22 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/3
 ip address 10.10.10.29 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/4
 ip address 10.0.0.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1 area 0
!
router ospf 1
 router-id 5.5.5.5
 passive-interface Loopback0
!
end
```

Fonte: Elaborado pelo autor.

Figura A.4 – Configurações básicas do roteador BRAGANÇA

```
hostname BRAGANCA
!
ip dhcp pool CLIENTE2
 network 10.0.1.0 255.255.255.0
 default-router 10.0.1.1
!
interface Loopback0
 ip address 7.7.7.7 255.255.255.255
 ip ospf 1 area 0
!
interface GigabitEthernet0/0
 ip address 10.10.10.34 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/1
 ip address 10.10.10.38 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/2
 ip address 10.0.1.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1 area 0
!
router ospf 1
 router-id 7.7.7.7
 passive-interface Loopback0
!
end
```

Fonte: Elaborado pelo autor.

Figura A.5 – Configurações básicas do roteador SANTA_LUZIA

```
hostname SANTA-LUZIA
!
ip dhcp pool CLIENTE3
 network 10.0.3.0 255.255.255.0
 default-router 10.0.3.1
!
interface Loopback0
 ip address 6.6.6.6 255.255.255.255
 ip ospf 1 area 0
!
interface GigabitEthernet0/0
 ip address 10.10.10.26 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/1
 ip address 10.10.10.37 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/2
 ip address 10.10.10.18 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/3
 ip address 10.10.10.30 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/4
 ip address 10.0.3.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1 area 0
!
router ospf 1
 router-id 6.6.6.6
 passive-interface Loopback0
!
end
```

Fonte: Elaborado pelo autor.

Figura A.6 – Configurações básicas do SERVER-STREAMING e do CLIENTE 1

```
hostname SERVER-STREAMING
!
interface Loopback0
 ip address 11.11.11.11 255.255.255.255
 ip ospf 1 area 0
 ip ospf network point-to-point
!
interface Port-channel1
 no switchport
 ip address 10.10.10.50 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/0
 no switchport
 no ip address
 channel-group 1 mode active
!
interface GigabitEthernet0/1
 no shut
 no switchport
 no ip address
 channel-group 1 mode active
!
interface GigabitEthernet0/2
 no shut
 no switchport
 no ip address
 channel-group 1 mode active
!
router ospf 1
 router-id 11.11.11.11
!
end
!!!!!!!!!!!!!!!!!!! CONFIG. CLIENTE 1 !!!!!!!!!!!!!!!!!!!
hostname CLIENTE1
!
ip dhcp pool CLIENTE1
 network 192.168.1.0 255.255.255.0
 default-router 192.168.1.1
!
interface Loopback0
 ip address 30.30.30.30 255.255.255.255
!
interface GigabitEthernet0/0
 ip address 192.168.1.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1 area 0
!
interface GigabitEthernet0/1
 ip address 10.10.10.58 255.255.255.252
 ip ospf network point-to-point
 ip ospf 1 area 0
!
router ospf 1
 router-id 30.30.30.30
!
end
```

Fonte: Elaborado pelo autor.

APÊNDICE B – CONFIGURAÇÃO DO MPLS E O MPLS-TE PARA O PRIMEIRO ESTUDO DE CASO

Figura B.1 – Configuração do MPLS e do MPLS-TE no *switch* CASTANHAL

```
hostname CASTANHAL
!
mpls traffic-eng tunnels
!
interface Tunnel11
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.14
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name CASTANHAL->
    CAPANEMA
 no routing dynamic
!
interface Tunnel22
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.38
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 2 explicit name CASTANHAL->
    BRAGANCA
 no routing dynamic
!
interface Tunnel33
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.26
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name CASTANHAL->
    SANLUZIA
 no routing dynamic
!
```

Fonte: Elaborado pelo autor.

Figura B.2 – Continuação da configuração do MPLS e do MPLS-TE no *switch* CASTANHAL

```
interface GigabitEthernet0/0
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 10000 10000
!
interface GigabitEthernet0/1
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 10000 10000
!
interface GigabitEthernet0/2
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 10000 10000
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0.0.0.0
!
ip explicit-path name CASTANHAL->BRAGANCA enable
 next-address 10.10.10.6
 next-address 10.10.10.41
 next-address 10.10.10.18
 next-address 10.10.10.38
!
ip explicit-path name CASTANHAL->CAPANEMA enable
 next-address 10.10.10.2
 next-address 10.10.10.14
!
ip explicit-path name CASTANHAL->SANLUZIA enable
 next-address 10.10.10.10
 next-address 10.10.10.26
!
mpls ldp router-id Loopback0 force
!
end
```

Fonte: Elaborado pelo autor.

Figura B.3 – Continuação da configuração do MPLS e do MPLS-TE no roteador CAPANEMA

```
hostname CAPANEMA
!
mpls traffic-eng tunnels
!
interface Tunnel1
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.1
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name CAPENA->
    CASTANHAL
 no routing dynamic
!
interface GigabitEthernet0/0
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 10000 10000
!
interface GigabitEthernet0/1
 mpls traffic-eng tunnels
 mpls ip
!
interface GigabitEthernet0/2
 mpls traffic-eng tunnels
 mpls ip
!
interface GigabitEthernet0/3
 mpls traffic-eng tunnels
 mpls ip
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip explicit-path name CAPENA->CASTANHAL enable
 next-address 10.10.10.13
 next-address 10.10.10.1
!
mpls ldp router-id Loopback0 force
!
end
```

Fonte: Elaborado pelo autor.

Figura B.4 – Continuação da configuração do MPLS e do MPLS-TE no roteador BRAGANÇA

```
hostname BRAGANCA
!
mpls traffic-eng tunnels
!
interface Tunnel2
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.5
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 1000
 tunnel mpls traffic-eng path-option 1 explicit name BRAGANCA->
    CASTANHAL
 no routing dynamic
!
interface GigabitEthernet0/0
 mpls traffic-eng tunnels
 mpls ip
!
interface GigabitEthernet0/1
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 1000 1000
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip explicit-path name BRAGANCA->CASTANHAL enable
 next-address 10.10.10.37
 next-address 10.10.10.17
 next-address 10.10.10.42
 next-address 10.10.10.5
!
mpls ldp router-id Loopback0 force
!
end
```

Fonte: Elaborado pelo autor.

Figura B.5 – Continuação da configuração do MPLS e do MPLS-TE no roteador SANTA_LUZIA

```
hostname SANTA_LUZIA
!
mpls traffic-eng tunnels
!
interface Tunnel3
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.9
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name SANLUZIA->
    CASTANHAL
 no routing dynamic
!
interface GigabitEthernet0/0
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 10000 10000
!
interface GigabitEthernet0/1
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 10000 10000
!
interface GigabitEthernet0/2
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 10000 10000
!
interface GigabitEthernet0/3
 mpls traffic-eng tunnels
 mpls ip
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip explicit-path name SANLUZIA->CASTANHAL enable
 next-address 10.10.10.25
 next-address 10.10.10.9
!
end
```

Fonte: Elaborado pelo autor.

APÊNDICE C – CONFIGURAÇÃO DO MPLS-TE COM SEGMENT ROUTING PARA O SEGUNDO ESTUDO DE CASO

Figura C.1 – Configuração do MPLS-TE com SR no roteador CASTANHAL

```
hostname CASTANHAL
!
mpls traffic-eng tunnels
!
interface Tunnel11
 description TUNNEL CASTANHAL->CAPANEMA
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.14
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name CASTANHAL->
    CAPANEMA segment-routing
!
interface Tunnel22
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.38
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name CASTANHAL->
    BRAGANCA segment-routing
!
interface Tunnel33
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.26
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name CASTANHAL->
    SANTA_LUZIA segment-routing
!
```

Fonte: Elaborado pelo autor.

Figura C.2 – Continuação da configuração do MPLS-TE com SR no roteador CASTANHAL

```
interface GigabitEthernet1
 mpls traffic-eng tunnels
!
interface GigabitEthernet2
 mpls traffic-eng tunnels
!
interface GigabitEthernet3
 mpls traffic-eng tunnels
!
segment-routing mpls
 global-block 16000 16010
 !
 connected-prefix-sid-map
  address-family ipv4
   1.1.1.1/32 absolute 16001 range 1
  exit-address-family
 !
!
router ospf 1
 segment-routing area 0 mpls
 segment-routing mpls
 passive-interface Loopback0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip explicit-path name CASTANHAL->CAPANEMA enable
 index 1 next-address 10.10.10.2
 index 2 next-address 10.10.10.14
!
ip explicit-path name CASTANHAL->SANTA_LUZIA enable
 index 1 next-address 10.10.10.10
 index 2 next-address 10.10.10.26
!
ip explicit-path name CASTANHAL->BRAGANCA enable
 index 1 next-address 10.10.10.6
 index 2 next-address 10.10.10.41
 index 3 next-address 10.10.10.18
 index 4 next-address 10.10.10.38
!
!
end
```

Fonte: Elaborado pelo autor.

Figura C.3 – Configuração do MPLS-TE com SR no roteador CAPANEMA

```
hostname CAPANEMA
!
mpls traffic-eng tunnels
!
interface Tunnel1
 description Tunnel_CAPANEMA->CASTANHAL
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.1
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name CAPANEMA->
    CASTANHAL segment-routing
!
interface GigabitEthernet1
 mpls traffic-eng tunnels
!
interface GigabitEthernet2
 mpls traffic-eng tunnels
!
interface GigabitEthernet3
 mpls traffic-eng tunnels
!
interface GigabitEthernet4
 mpls traffic-eng tunnels
!
segment-routing mpls
 global-block 16000 16010
 !
 connected-prefix-sid-map
  address-family ipv4
   5.5.5.5/32 absolute 16005 range 1
  exit-address-family
 !
!
router ospf 1
 segment-routing area 0.0.0.0 mpls
 segment-routing mpls
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0.0.0.0
!
ip explicit-path name CAPANEMA->CASTANHAL enable
 index 1 next-address 10.10.10.13
 index 2 next-address 10.10.10.1
!
end
```

Fonte: Elaborado pelo autor.

Figura C.4 – Configuração do MPLS-TE com SR no roteador BRAGANÇA

```
hostname BRAGANCA
!
mpls traffic-eng tunnels
!
interface Tunnel2
 description Tunnel_BRANGANCA->CASTANHAL
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.5
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name BRANGANCA->
    CASTANHAL segment-routing
!
interface GigabitEthernet1
 mpls traffic-eng tunnels
!
interface GigabitEthernet2
 mpls traffic-eng tunnels
!
segment-routing mpls
 global-block 16000 16010
 !
 connected-prefix-sid-map
  address-family ipv4
   7.7.7.7/32 absolute 16007 range 1
  exit-address-family
 !
!
router ospf 1
 segment-routing area 0.0.0.0 mpls
 segment-routing mpls
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0.0.0.0
!
ip explicit-path name BRANGANCA->CASTANHAL enable
 index 1 next-address 10.10.10.37
 index 2 next-address 10.10.10.17
 index 3 next-address 10.10.10.42
 index 4 next-address 10.10.10.5
!
end
```

Fonte: Elaborado pelo autor.

Figura C.5 – Configuração do MPLS-TE com SR no roteador SANTA_LUZIA

```
hostname SANTA_LUZIA
!
mpls traffic-eng tunnels
!
interface Tunnel3
 description Tunnel_SANTA_LUZIA->CASTANHAL
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.10.10.9
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name SANTA_LUZIA->
    CASTANHAL segment-routing
!
interface GigabitEthernet1
 mpls traffic-eng tunnels
!
interface GigabitEthernet2
 mpls traffic-eng tunnels
!
interface GigabitEthernet3
 mpls traffic-eng tunnels
!
interface GigabitEthernet4
 mpls traffic-eng tunnels
!
segment-routing mpls
 global-block 16000 16010
 !
 connected-prefix-sid-map
  address-family ipv4
   6.6.6.6/32 absolute 16006 range 1
  exit-address-family
 !
!
router ospf 1
 router-id 6.6.6.6
 segment-routing area 0.0.0.0 mpls
 segment-routing mpls
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0.0.0.0
!
ip explicit-path name SANTA_LUZIA->CASTANHAL enable
 index 1 next-address 10.10.10.25
 index 2 next-address 10.10.10.9
!
end
```

Fonte: Elaborado pelo autor.

APÊNDICE D – CONFIGURAÇÕES BÁSICAS DOS ROTEADORES PARA O TERCEIRO ESTUDO DE CASO

Figura D.1 – Configurações básicas do roteador PE1

```
hostname PE1
!
vrf vpna
 rd 100:1
 address-family ipv4 unicast
  import route-target
   111:1
  !
  export route-target
   111:1
  !
 !
!
interface Loopback0
 ipv4 address 1.1.1.1 255.255.255.255
!
interface GigabitEthernet0/0/0/0
 ipv4 address 10.10.10.1 255.255.255.252
!
interface GigabitEthernet0/0/0/1
 ipv4 address 10.10.10.9 255.255.255.252
!
interface GigabitEthernet0/0/0/2
 vrf vpna
 ipv4 address 10.1.1.1 255.255.255.0
!
router isis 1
 max-metric level 1
 is-type level-1
 net 10.0000.0000.0001.00
 address-family ipv4 unicast
  metric-style wide
  segment-routing mpls
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid absolute 16001
  !
 !
 interface GigabitEthernet0/0/0/0
  address-family ipv4 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  address-family ipv4 unicast
  !
 !
!
```

Fonte: Elaborado pelo autor.

Figura D.2 – Configurações do roteador P1

```
hostname P1
!
interface Loopback0
 ipv4 address 2.2.2.2 255.255.255.255
!
interface GigabitEthernet0/0/0/0
 ipv4 address 10.10.10.2 255.255.255.252
!
interface GigabitEthernet0/0/0/1
 ipv4 address 10.10.10.5 255.255.255.252
!
interface GigabitEthernet0/0/0/2
 ipv4 address 10.10.10.17 255.255.255.252
!
router isis 1
 max-metric level 1
 is-type level-1
 net 10.0000.0000.0002.00
 address-family ipv4 unicast
  metric-style wide
  segment-routing mpls
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid absolute 16002
  !
 !
 interface GigabitEthernet0/0/0/0
  address-family ipv4 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  address-family ipv4 unicast
  !
 !
 interface GigabitEthernet0/0/0/2
  address-family ipv4 unicast
  !
 !
!
segment-routing
 global-block 16000 16010
!
end
```

Fonte: Elaborado pelo autor.

Figura D.3 – Configurações básicas do roteador PE2

```
hostname PE2
!
vrf vpna
 rd 200:1
 address-family ipv4 unicast
  import route-target
   111:1
  !
  export route-target
   111:1
  !
 !
!
interface Loopback0
 ipv4 address 3.3.3.3 255.255.255.255
!
interface GigabitEthernet0/0/0/0
 ipv4 address 10.10.10.6 255.255.255.252
!
interface GigabitEthernet0/0/0/1
 ipv4 address 10.10.10.14 255.255.255.252
!
interface GigabitEthernet0/0/0/2
 vrf vpna
 ipv4 address 10.2.1.1 255.255.255.0
!
router isis 1
 max-metric level 1
 is-type level-1
 net 10.0000.0000.0003.00
 address-family ipv4 unicast
  metric-style wide
  segment-routing mpls
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid absolute 16003
  !
 !
 interface GigabitEthernet0/0/0/0
  address-family ipv4 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  address-family ipv4 unicast
  !
 !
!
```

Fonte: Elaborado pelo autor.

Figura D.4 – Configurações dos CE's

```
hostname CE1
!
ip dhcp pool CE1
 network 192.168.1.0 255.255.255.0
 default-router 192.168.1.1
!
interface Loopback0
 ip address 11.11.11.11 255.255.255.255
!
interface GigabitEthernet0/0
 ip address 10.1.1.2 255.255.255.0
!
interface GigabitEthernet0/1
 ip address 192.168.1.1 255.255.255.0
!
router bgp 65410
 bgp log-neighbor-changes
 network 11.11.11.11 mask 255.255.255.255
 redistribute connected
 redistribute static
 neighbor 10.1.1.1 remote-as 100
 neighbor 10.1.1.1 ebgp-multihop 255
 neighbor 10.1.1.1 next-hop-self
!
ip route 0.0.0.0 0.0.0.0 10.1.1.1
!
end
!
!!!!!!!!!!!!!!!!!!!! CONFIG. CE2 !!!!!!!!!!!!!!!!!!!!
!
hostname CE2
!
interface Loopback0
 ip address 22.22.22.22 255.255.255.255
!
interface Loopback10
 ip address 33.33.33.33 255.255.255.255
!
interface GigabitEthernet0/0
 ip address 10.2.1.2 255.255.255.0
!
router bgp 65420
 bgp log-neighbor-changes
 network 22.22.22.22 mask 255.255.255.255
 network 33.33.33.33 mask 255.255.255.255
 redistribute connected
 redistribute static
 neighbor 10.2.1.1 remote-as 100
 neighbor 10.2.1.1 ebgp-multihop 255
 neighbor 10.2.1.1 next-hop-self
!
end
```

Fonte: Elaborado pelo autor.

APÊNDICE E – CONFIGURAÇÕES NECESSÁRIAS PARA A IMPLEMENTAÇÃO DA SR-TE *POLICY* PARA O TERCEIRO ESTUDO DE CASO

Figura E.1 – Configurações necessárias para a implementação da SR-TE *policy* no PE1

```
router bgp 100
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 neighbor 3.3.3.3
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
   route-policy IN-PREFIX in
   route-policy OUT-PREFIX out
  !
  address-family vpnv4 unicast
   route-policy IN-PREFIX in
   route-policy OUT-PREFIX out
  !
 !
 vrf vpna
  rd 100:1
  address-family ipv4 unicast
   redistribute connected
  !
  neighbor 10.1.1.2
   remote-as 65410
   ebgp-multihop 255
   address-family ipv4 unicast
    route-policy IN-PREFIX in
    route-policy OUT-PREFIX out
   !
  !
 !
!
end
```

Fonte: Elaborado pelo autor.

Figura E.2 – Continuação das configurações necessárias para a implementação da SR-TE *policy* no PE1

```
route-policy IN-PREFIX
  pass
end-policy
!
route-policy OUT-PREFIX
  pass
end-policy
!
segment-routing
 global-block 16000 16010
 traffic-eng
  segment-list VPNA_CORE
   index 10 mpls label 16002
   index 20 mpls label 16004
   index 30 mpls label 16003
  !
  policy VPNA_CORE
   color 100 end-point ipv4 3.3.3.3
   candidate-paths
    preference 100
     explicit segment-list VPNA_CORE
     !
    !
   !
  !
 !
!
end
```

Fonte: Elaborado pelo autor.

Figura E.3 – Configurações necessárias para a implementação da SR-TE *policy* no PE2

```
!
extcommunity-set opaque VPNA_ACCESS
  100
end-set
!
route-policy IN-PREFIX
  pass
end-policy
!
route-policy OUT-PREFIX
  pass
end-policy
!
route-policy VPNA_ACCESS
  if destination in (22.22.22.22/32) then
    set extcommunity color VPNA_ACCESS
  else
    pass
  endif
end-policy
!
router bgp 100
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
 !
 neighbor 1.1.1.1
  remote-as 100
  update-source Loopback0
  address-family ipv4 unicast
   route-policy IN-PREFIX in
   route-policy OUT-PREFIX out
  !
  address-family vpnv4 unicast
   route-policy IN-PREFIX in
   route-policy VPNA_ACCESS out
  !
 !
 vrf vpna
  rd 200:1
  address-family ipv4 unicast
   redistribute connected
  !
  neighbor 10.2.1.2
   remote-as 65420
   ebgp-multihop 255
   address-family ipv4 unicast
    route-policy IN-PREFIX in
    route-policy OUT-PREFIX out
   !
  !
 !
!
segment-routing
 global-block 16000 16010
!
end
```

Fonte: Elaborado pelo autor.