SERVIÇO DE REDE VIRTUAL PRIVADA APLICADO À UMA TOPOLOGIA DE REDE COM COMUTAÇÃO DE RÓTULOS MULTIPROTOCOLO

Henrique Araújo Lima, Angelo Travizan Neto, Leonardo da Silva Martins, Danielli Araújo Lima, Marcelo Zanchetta do Nascimento
Universidade Federal de Uberlândia (UFU) - Faculdade de Computação (FACOM), Uberlândia, MG, Brasil
E-mails: {henriquebrmg,travizanneto}@gmail.com,leonardosilva_6@hotmail.com,{danielli,marcelo.nascimento}@ufu.br

***Resumo*—Este artigo descreve a construção de uma topologia de rede MPLS (Multiprotocol Label Switching) e a implementação do serviço de VPN (Virtual Private Network) no modelo de circuito proposto. Para realizar a implementação do circuito foi necessário recorrer ao emulador GNS3. Além disso, o presente artigo discute as principais características do protocolo MPLS e do serviço VPN. Os resultados preliminares indicaram que esta simulação pode apresentar comportamentos interessantes que reproduz os principais aspectos do circuito em um cenário real.**

***Index Terms*—Redes de Computadores, Comutação de Rótulos Multiprotocolo, Rede Virtual Privada.**

VIRTUAL PRIVATE NETWORK SERVICE APPLIED TO A MULTIPROTOCOL LABEL SWITCHING NET TOPOLOGY

***Abstract*—This paper describes the construction of a network topology MPLS (Multiprotocol Label Switching) and a implementation of VPN Service (Virtual Private Network) in the proposed circuit model. To perform a circuit implementation was necessary appealing to the GNS3 emulator. In addition, the present article discusses the main features of the MPLS protocol and of the VPN service. Preliminary results have indicated that the simulation can exhibit interesting behaviors that reproduces the major circuit aspects into a real scenario.**

***Index Terms*—Computer Network, Multiprotocol Label Switching, Virtual Private Network.**

I. INTRODUÇÃO

Com o aumento dos sistemas de informação conectados à rede mundial de computadores [1] fez-se necessária a criação de redes virtuais privadas (VPN) para acesso e transmissão segura de dados e informações. Além disso, as VPNs possibilitam a redução de barreiras territoriais, pois permitem o acesso à redes locais (LANs) utilizando-se da Internet. O serviço VPN tem sido amplamente utilizado para diversos propósitos, entre eles destacam-se: a navegação privativa, navegação segura e o acesso à conteúdos de qualquer lugar do mundo [2].

Os modelos mais tradicionais para a implementação do serviço de VPN em um circuito virtual são *overlay* e *peer-to-peer*. O primeiro modelo permite que o serviço seja implementado de maneira segura e isolada. Já no último, a principal vantagem é o encaminhamento simplificado de dados. Um pouco menos tradicional, mas muito importante, o modelo Multiprotocol Label Switching (MPLS) possui os benefícios apresentados dos modelos *overlay* e *peer-to-peer* [3]. No modelo MPLS, a VPN usufrui de uma estrutura de rede IP pública a fim de criar "túneis" em uma infraestrutura de rede privada [4]. Sabe-se que o Open Shortest Path First (OSPF) é um dos protocolos de roteamento de redes mais largamente utilizado [5]. Portanto, este protocolo foi elegido para a transmissão de dados entre os roteadores presentes na topologia apresentada nesse trabalho.

Este trabalho tem por objetivo: (i) apresentar a implementação de um novo circuito virtual no simulador GNS3 que servirá para testes; (ii) apresentar as definições dos conceitos relativos à implementação do circuito; (iii) apresentar testes para a verificação de integridade do circuito virtual. Mais especificamente, o circuito virtual proposto será usado para a simulações de um serviço VPN rodando em uma rede MPLS e com roteadores configurados com o protocolo OSPF para a comutação de pacotes. Os testes serão realizados com o auxílio do próprio GNS3 e também da ferramenta de captura de tráfego de dados, o Wireshark [6].

II. FUNDAMENTAÇÃO TEÓRICA

Nesta seção será apresentada a fundamentação teórica conceituando os principais elementos envolvidos na elaboração do circuito proposto neste trabalho, dentre eles, serão destacados: MPLS, OSPF e VPN. Adicionalmente, serão apresentadas as principais características dos elementos envolvidos para a elaboração do circuito e realização das simulações.

A. *Multiprotocol Label Switching - MPLS*

De acordo com [7], o MPLS (Multiprotocol Label Switching) é uma técnica de repasse de dados em um contexto de redes de computadores e telecomunicações de alta performance. Este mecanismo baseia-se em cada informação possuir rótulos (ou labels), que identificam "links virtuais", ou caminhos entre nós distantes, ao invés de pontos terminais [8]. Cada nó da rede recebe o dado e repassa essa informação para o nó de menor caminho (baseando-se neste rótulo) [9].

O MPLS trabalha em uma camada que é considerada entre as camadas 2 e 3 do modelo OSI (de enlace e redes) sendo denominada de camada 2.5 [10], pelo fato de ser uma técnica de repasse, sem dar ênfase ao enlace, nem à rede [11]. O nome

Multiprotocol se dá através da capacidade que o MPLS possui de encapsular pacotes de vários protocolos de rede, suportando uma grande variedade de tecnologias [12].

O encaminhamento de pacotes dado pelo MPLS é feito basicamente com a adição de um rótulo nos pacotes de tráfego, o qual é utilizado para direcionar a informação, visando tomar sempre o menor caminho dentro deste contexto de rede [13]. Os nós que possuem esta técnica de repasse são configurados previamente [14]. Segundo [7], ao receber um pacote contendo rótulo MPLS, o nó verifica este label, e "entrega" a informação ao próximo nó pertencente ao menor caminho entre a origem e destino desta informação, dado pelo rótulo analisado [15]. O uso do Multiprotocol Label Switching é vantajoso em relação a utilizar endereços de rede longos [16]. Isto se justifica porque as consultas às tabelas de roteamento são complexas e custosas computacionalmente [17].

O MPLS trabalha colocando prefixos nos pacotes com um cabeçalho MPLS, contendo um ou mais labels. Um cabeçalho MPLS consiste de 20 bits reservado para o valor do rótulo, 3 bits para campo de classe de tráfego para prioridade em QoS (Quality of Service) e ECN (Explicit Congestion Notification), 1 bit reservado para notificar o fim da pilha e os últimos 8 bits são úteis para sinalizar o campo de TTL (Time to Live). Dessa forma, um cabeçalho MPLS consiste de 32 bits.

B. Open Shortest Path First - OSPF

O OSPF (Open Shortest Path First) é um protocolo para roteamento em redes de computadores que operam sobre o Protocolo IP [18]. Ele utiliza um algoritmo de estado do link, e é classificado entre os de roteamento de interiores, operando em um só sistema autônomo. Ele consiste em capturar o estado do link dos nós disponíveis da rede, e constrói um mapa da topologia. Seu objetivo é fazer o roteamento de dados entre o nó origem e o destino de maneira mais rápida, pelo caminho mais curto.

A topologia é representada pelo OSPF através de uma tabela de roteamento à camada de Rede, no modelo OSI, que direciona datagramas aos destinos com endereços IP encontrados nos pacotes IP. Cada nó da rede possui sua tabela de roteamento.

O roteamento realizado pelo protocolo OSPF baseia-se no algoritmo de menor caminho de Dijkstra [19] (aplicado em grafos), que consiste em construir uma árvore geradora de custo mínimo entre a origem e o destino da informação, e encontrar o menor caminho entre os dois nós citados anteriormente. O algoritmo de Dijkstra soluciona o problema do caminho mais curto em um grafo, com arestas de peso não negativo. Ele considera um conjunto S de menores caminhos, iniciando com um vértice inicial I. A cada iteração, busca-se nos nós adjacentes pertencentes à S aquele vértice com menor distância relativa à I, e adiciona-o à S e, então, repete-se estes passos até todos os vértices alcançáveis por I estejam em S. Isto o classifica como um algoritmo guloso. Ou seja, toma a decisão que parece ótima no momento [20]

O objetivo do Open Shortest Path First é encontrar o menor caminho entre dois nós em uma rede de computadores. Dessa forma, o OSPF é aplicável de maneira completa no algoritmo de Dijkstra, sabendo que a rede citada anteriormente é facilmente representada por um grafo, em que cada host na rede é representado por um vértice no grafo, e cada enlace é representado por uma aresta.

C. Virtual Private Network - VPN

Hoje em dia a Internet é tida como uma grande rede de comunicações pública, onde os dados são trafegados utilizando alguns protocolos que nem sempre são seguros. Com isso surge o sistema de VPN que representa a criação de uma rede de comunicações privada sobre uma rede de comunicações públicas, tornando possível conectar duas localidades diferentes como se fossem parte da mesma rede interna. Essa rede privada criada conta com as tecnologias de tunelamento e criptografia para fazer o tráfego de dados seguro. Sabe-se que a criptografia garante a autenticidade, a integridade e o sigilo das informações [1] e o tunelamento permite a utilização da rede pública para realizar o tráfego seguro dessas informações [21].

Atualmente o sistema de VPN é muito utilizado principalmente no ambiente empresarial. Com o grande avanço das empresas pelo mundo inteiro a necessidade de comunicação entre uma matriz com suas filiais, fornecedores, distribuidores e clientes fica cada vez maior, formando assim uma infra-estrutura empresarial. Nesse cenário a VPN fornece a segurança das informações, um requisito de grande importância em uma infra-estrutura empresarial, e além disso fornece uma alternativa interessante para diminuir os custos com a manutenção das redes, pois permite que as conexões dedicadas, que possuem um custo muito elevado, sejam substituídas por conexões públicas com o custo mais acessível [22].

Podemos dividir a VPN em três tipos conforme sua funcionalidade. O primeiro tipo de VPN é a de acesso, a qual oferece acesso remoto à uma rede intranet ou extranet em cima de um infra-estrutura que compartilha as mesmas políticas de rede privada. O segundo tipo de VPN é a para intranet, onde as ligações interligam e integram a rede da sede, os escritórios e as filiais em cima de uma infra-estrutura compartilhada. Por último, tem-se a VPN para extranet, a qual realiza perfis de interação por meio da rede Internet com uma infra-estrutura compartilhada que usa conexões dedicadas [23].

As VPNs baseiam-se no protocolo de tunelamento, onde um túnel é basicamente o caminho lógico percorrido pelo pacote ao longo da rede. O uso do tunelamento nas VPNs incorpora um novo componente a esta técnica, antes de encapsular o pacote, ele é criptografado. Esse protocolo de tunelamento encapsula o pacote com um cabeçalho adicional que contém as informações de roteamento [24].

O tunelamento das conexões VPN podem acontecer tanto na camada 2 (enlace) onde é utilizado PPP sobre IP, podendo citar os protocolos PPTP, L2TP e L2F, quanto na camada 3 (rede), onde é colocado um cabeçalho adicional antes de enviar o pacote, podendo citar o protocolo IPSec como exemplo. Para se estabelecer um túnel é necessário que as extremidades usem o mesmo protocolo [23].

Para a realização deste trabalho foi utilizado o protocolo de roteamento MPLS como protocolo de tunelamento para a criação da VPN. Portanto, criou-se uma VPN de camada 2.5.

III. METODOLOGIA

Para a implementação do modelo proposto neste trabalho fez-se necessário a realização de três tarefas principais: (i) a elaboração de um novo circuito virtual no software GNS3 para suportar as simulações; (ii) a configuração dos roteadores no mesmo software; (iii) a execução de testes e a captura de pacotes através dos software GNS3 e Wireshark.

A. *Circuito virtual proposto*

A primeira tarefa consistiu da implementação do circuito virtual, que foi elaborada e configurada no próprio software GNS3. A Figura 1 mostra a implementação da topologia do circuito VPN MPLS. Neste circuito, foram utilizados nove roteadores da CISCO 7200. Neles foram feitas as configurações básicas referentes às interfaces e também do protocolo de roteamento básico, OSPF. Além disso, pode-se observar que a parte da rede que está em azul escuro foi configurada com VPN MPLS, a fim de se criar as VPNs.

A Tabela I relaciona os nove roteadores utilizados neste tra-

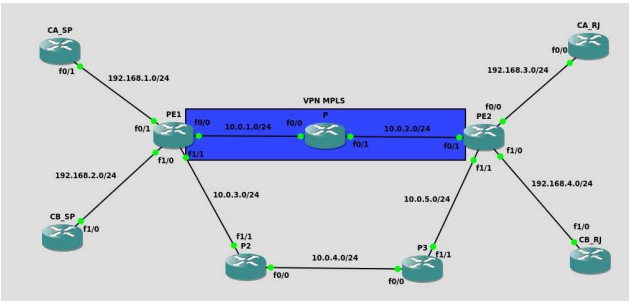

Figura 1. Topologia do circuito VPN MPLS.

balho com seus seus respectivos endereços IPs e as interfaces a eles associados. Esta tabela, portanto é um complemento da Figura 1, pois apresenta de forma simplificada as informações referentes as redes nas quais cada interface dos nove roteadores estão submersos.

B. *Comandos utilizados para configuração*

Os comandos utilizados nos nove roteadores serão descritos a seguir. Antes destas configurações, fez-se necessário que cada roteador estivesse em modo privilegiado (#). Portanto, antes da execução dos comandos que serão apresentados a seguir, em cada roteador utilizou-se o comando *enable*.

Habilitando o modo privilegiado

```
> enable
```

Roteador P

```
P# configure terminal
P# hostname P
P# ip cef
P# mpls ip
```

Tabela I
ENDEREÇOS DAS INTERFACES DOS ROTEADORES

Roteador	Interface	Endereço IP
CA_SP	f0/1	192.168.1.2
CA_RJ	f0/0	192.168.3.2
CB_SP	f1/0	192.168.2.2
CB_RJ	f1/0	192.168.4.2
PE1	f0/0	10.0.1.2
	f0/1	192.168.1.1
	f1/0	192.168.2.1
	f1/1	10.0.3.2
PE2	f0/0	192.168.3.1
	f0/1	10.0.2.2
	f1/0	192.168.4.1
	f1/1	10.0.5.2
P	f0/0	10.0.1.1
	f0/1	10.0.2.1
P2	f0/0	10.0.4.1
	f1/1	10.0.3.1
P3	f0/0	10.0.4.2
	f1/1	10.0.5.1

```
P# interface fastethernet 0/0
P# ip address 10.0.1.1 255.255.255.0
P# mpls ip
P# no shutdown
P# exit
P# interface fastethernet 0/1
P# ip address 10.0.2.1 255.255.255.0
P# mpls ip
P# no shutdown
P# exit
P# router ospf 99
P# router-id 10.0.0.11
P# network 10.0.1.0 0.255.255.255 area 0
P# network 10.0.2.0 0.255.255.255 area 0
P# end
P# copy running-config startup-config
```

Roteador P2

```
P2# configure terminal
P2# hostname P2
P2# interface fastethernet 1/1
P2# ip address 10.0.3.1 255.255.255.0
P2# no shutdown
P2# exit
P2# interface fastethernet 0/0
P2# ip address 10.0.4.1 255.255.255.0
P2# no shutdown
P2# exit
P2# router ospf 99
P2# router-id 10.0.0.12
P2# network 10.0.3.0 0.255.255.255 area 0
P2# network 10.0.4.0 0.255.255.255 area 0
P2# end
P2# copy running-config startup-config
```

Roteador P3

```
P3# configure terminal
P3# hostname P3
```

```
P3# interface fastethernet 1/1
P3# ip address 10.0.5.1 255.255.255.0
P3# no shutdown
P3# exit
P3# interface fastethernet 0/0
P3# ip address 10.0.4.2 255.255.255.0
P3# no shutdown
P3# exit
P3# router ospf 99
P3# router-id 10.0.0.13
P3# network 10.0.4.0 0.255.255.255 area 0
P3# network 10.0.5.0 0.255.255.255 area 0
P3# end
P3# copy running-config startup-config
```

Roteador PE1

```
PE1# configure terminal
PE1# hostname PE1
PE1# ip cef
PE1# mpls ip
PE1# interface fastethernet 0/0
PE1# ip address 10.0.1.2 255.255.255.0
PE1# mpls ip
PE1# no shutdown
PE1# exit
PE1# interface fastethernet 1/1
PE1# ip address 10.0.3.2 255.255.255.0
PE1# no shutdown
PE1# interface Loopback0
PE1# ip address 10.0.0.2 255.255.255.255
PE1# exit
PE1# router ospf 99
PE1# router-id 10.0.0.2
PE1# network 10.0.1.0 0.255.255.255 area 0
PE1# network 10.0.3.0 0.255.255.255 area 0
PE1# exit
PE1# ip vrf NetworkA
PE1# rd 100:1
PE1# route-target export 100:1
PE1# route-target import 100:1
PE1# exit
PE1# ip vrf NetworkB
PE1# rd 100:2
PE1# route-target export 100:2
PE1# route-target import 100:2
PE1# exit
PE1# interface fastethernet 0/1
PE1# ip vrf forwarding NetworkA
PE1# ip address 192.168.1.1 255.255.255.0
PE1# no shutdown
PE1# exit
PE1# interface fastethernet 1/0
PE1# ip vrf forwarding NetworkB
PE1# ip address 192.168.2.1 255.255.255.0
PE1# no shutdown
PE1# exit
PE1# router bgp 100
PE1# bgp log-neighbor-changes
PE1# neighbor 10.0.0.3 remote-as 100
PE1# neighbor 10.0.0.3 update-source Loopback0
PE1# address-family ipv4 vrf NetworkA
PE1# no auto-summary
PE1# no synchronization
PE1# redistribute connected
PE1# exit-address-family
PE1# address-family ipv4 vrf NetworkB
PE1# no auto-summary
PE1# no synchronization
PE1# redistribute connected
PE1# exit-address-family
PE1# address-family vpnv4
PE1# neighbor 10.0.0.3 activate
PE1# neighbor 10.0.0.3 send-community extended
PE1# exit-address-family
PE1# end
PE1# copy running-config startup-config
```

Roteador PE2

```
PE2# configure terminal
PE2# hostname PE2
PE2# ip cef
PE2# mpls ip
PE2# interface fastethernet 0/1
PE2# ip address 10.0.2.2 255.255.255.0
PE2# mpls ip
PE2# no shutdown
PE2# exit
PE2# interface fastethernet 1/1
PE2# ip address 10.0.5.2 255.255.255.0
PE2# no shutdown
PE2# interface Loopback0
PE2# ip address 10.0.0.3 255.255.255.255
PE2# exit
PE2# router ospf 99
PE2# router-id 10.0.0.3
PE2# network 10.0.2.0 0.255.255.255 area 0
PE2# network 10.0.5.0 0.255.255.255 area 0
PE2# exit
PE2# ip vrf NetworkA
PE2# rd 100:1
PE2# route-target export 100:1
PE2# route-target import 100:1
PE2# exit
PE2# ip vrf NetworkB
PE2# rd 100:2
PE2# route-target export 100:2
PE2# route-target import 100:2
PE2# exit
PE2# interface fastethernet 0/0
PE2# ip vrf forwarding NetworkA
PE2# ip address 192.168.3.1 255.255.255.0
PE2# no shutdown
PE2# exit
PE2# interface fastethernet 1/0
PE2# ip vrf forwarding NetworkB
PE2# ip address 192.168.4.1 255.255.255.0
PE2# no shutdown
PE2# exit
PE2# router bgp 100
PE2# bgp log-neighbor-changes
PE2# neighbor 10.0.0.2 remote-as 100
PE2# neighbor 10.0.0.2 update-source Loopback0
PE2# address-family ipv4 vrf NetworkA
PE2# no auto-summary
PE2# no synchronization
PE2# redistribute connected
PE2# exit-address-family
PE2# address-family ipv4 vrf NetworkB
PE2# no auto-summary
PE2# no synchronization
PE2# redistribute connected
PE2# exit-address-family
```

```
PE2# address-family vpnv4
PE2# neighbor 10.0.0.2 activate
PE2# neighbor 10.0.0.2 send-community extended
PE2# exit-address-family
PE2# end
PE2# copy running-config startup-config
```

Roteador CA_SP

```
CA_SP# configure terminal
CA_SP# hostname CA_SP
CA_SP# interface fastethernet 0/1
CA_SP# ip address 192.168.1.2 255.255.255.0
CA_SP# no shutdown
CA_SP# exit
CA_SP# ip route 0.0.0.0 0.0.0.0 192.168.1.1
CA_SP# end
CA_SP# copy running-config startup-config
```

Roteador CA_RJ

```
CA_RJ# configure terminal
CA_RJ# hostname CA_RJ
CA_RJ# interface fastethernet 0/0
CA_RJ# ip address 192.168.3.2 255.255.255.0
CA_RJ# no shutdown
CA_RJ# exit
CA_RJ# ip route 0.0.0.0 0.0.0.0 192.168.3.1
CA_RJ# end
CA_RJ# copy running-config startup-config
```

Roteador CB_SP

```
CB_SP# configure terminal
CB_SP# hostname CB_SP
CB_SP# interface fastethernet 1/0
CB_SP# ip address 192.168.2.2 255.255.255.0
CB_SP# no shutdown
CB_SP# exit
CB_SP# ip route 0.0.0.0 0.0.0.0 192.168.2.1
CB_SP# end
CB_SP# copy running-config startup-config
```

Roteador CB_RJ

```
CB_RJ# configure terminal
CB_RJ# hostname CB_RJ
CB_RJ# interface fastethernet 1/0
CB_RJ# ip address 192.168.4.2 255.255.255.0
CB_RJ# no shutdown
CB_RJ# exit
CB_RJ# ip route 0.0.0.0 0.0.0.0 192.168.4.1
CB_RJ# end
CB_RJ# copy running-config startup-config
```

IV. RESULTADOS

Após a implementação da arquitetura proposta e a configuração dos roteadores no software GNS3, foi necessário executar diversos testes para comprovar que o modelo de fato atendia as expectativas e estava funcionando corretamente. Portanto, para a realização destes experimentos, utilizou-se o software GNS3 para checar a conexão entre roteadores e descobrir a rota de uma rede a outra. Além do GNS3, fez-se necessário a aplicação do Wireshark para capturar quais pacotes estavam sendo trafegados no circuito.

O primeiro teste executado pode ser observado na Figura 2 e tem por objetivo comprovar que roteadores que pertencem à mesma VPN conseguem se conectar. Neste caso, o teste mostra a rota do roteador CA_SP ao roteador CA_RJ (192.168.3.2) através da VPN *NetworkA* utilizando o MPLS.

```
CA_SP#
CA_SP#ping 192.168.3.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/40/44 ms
CA_SP#
CA_SP#
CA_SP#
CA_SP#traceroute 192.168.3.2
Type escape sequence to abort.
Tracing the route to 192.168.3.2
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.1.1 36 msec 32 msec 20 msec
  2 10.0.1.1 [MPLS: Labels 20/20 Exp 0] 44 msec 40 msec 48 msec
  3 192.168.3.1 16 msec 48 msec 40 msec
  4 192.168.3.2 44 msec 40 msec *
CA_SP#
```

Figura 2. Rota do roteador CA_SP ao roteador CA_RJ.

O segundo teste executado pode ser observado na Figura 3 e possui o mesmo objetivo do primeiro teste. No entanto, este teste mostra a rota entre os roteadores CB_RJ e CB_SP (192.168.2.2) através da VPN *NetworkB* utilizando o MPLS.

```
CB_RJ#
CB_RJ#ping 192.168.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/37/56 ms
CB_RJ#
CB_RJ#
CB_RJ#
CB_RJ#
CB_RJ#traceroute 192.168.2.2
Type escape sequence to abort.
Tracing the route to 192.168.2.2
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.4.1 20 msec 32 msec 8 msec
  2 10.0.2.1 [MPLS: Labels 16/21 Exp 0] 12 msec 12 msec 28 msec
  3 192.168.2.1 16 msec 16 msec 28 msec
  4 192.168.2.2 60 msec 40 msec *
CB_RJ#
```

Figura 3. Rota do roteador CB_RJ ao roteador CB_SP.

O terceiro e último teste realizado no GNS3 tem por objetivo checar que não há conexão entre roteadores de VPNs diferentes. Neste experimento, foi realizada uma tentativa de conexão do roteador CA_SP pertencente à VPN *NetworkA* ao roteador CB_RJ pertencente à VPN *NetworkB*. Este teste pode ser visualizado na Figura 4 e comprova que não há comunicação entre roteadores de VPNs distintas.

```
CA_SP#
CA_SP#ping 192.168.4.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.4.2, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
CA_SP#
```

Figura 4. Tentativa fálida de comunicação entre roteadores de VPNs distintas

Por fim, foi realizado um teste utilizando-se o software Wireshark. Com a realização deste experimento foi possível capturar o tráfego de dados no enlace 10.0.1.0. Desta forma, foi possível confirmar que neste enlace estão passando pacotes MPLS como esperado. A Figura 5 é responsável por apresentar os resultados deste experimento. O protocolo LDP é responsável pelo tráfego do MPLS.

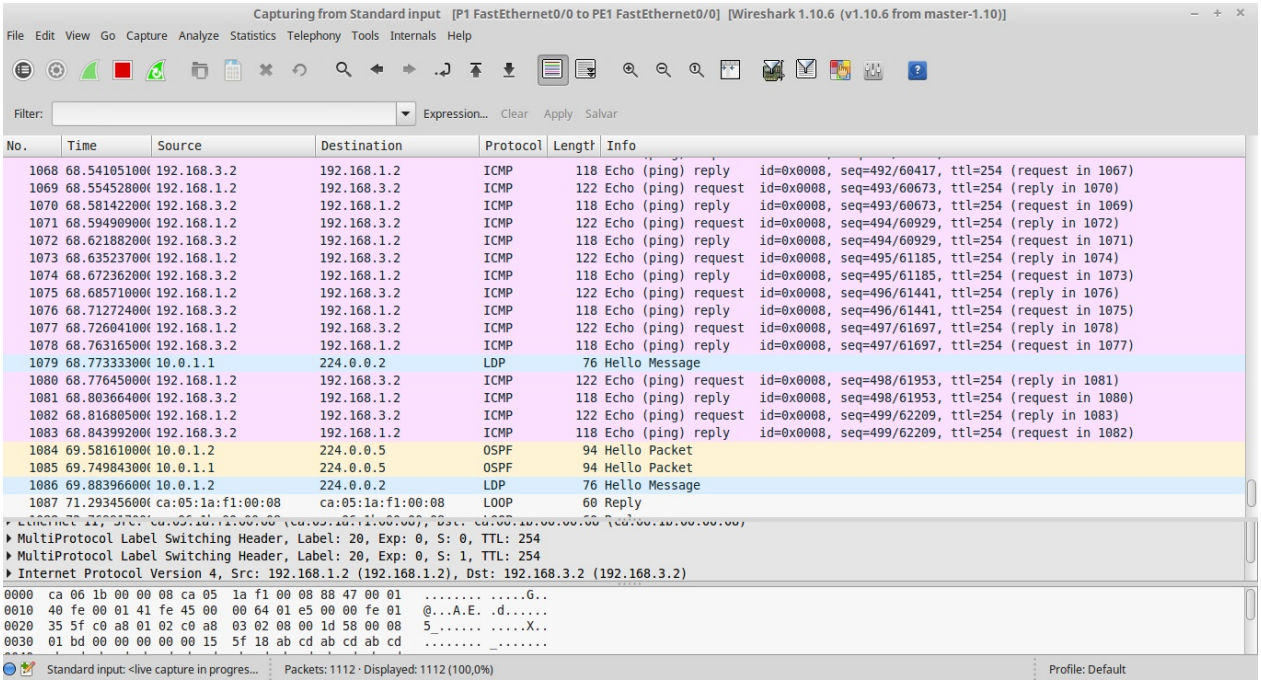

Figura 5. Captura de pacotes no enlace 10.0.1.0 através do software wireshark.

V. Conclusão

Neste trabalho foi investigado um modelo de Rede Vitual Privada (VPN) implementado sobre os protocolos MPLS e OSPF sobre um circuito proposto para gerar os testes deste trabalho. Além disso, foi possível apresentar os benefícios e a aplicabilidade da VPN e os principais conceitos fundamentais para a elaboração deste trabalho. Adicionalmente, uma análise das características do modelo desenvolvido foi abordada. Com a execução do modelo descrito foi possível gerar dados visuais, extraídos dos softwares GNS3 e wireshark, onde os comportamentos padrões de tráfego de dados puderam ser observados. Como trabalho futuro, espera-se criar novos circuitos para a elaboração de testes mais refinados.

Referências

[1] E. C. Silva, J. A. Soares, and D. A. Lima, "Autômatos celulares unidimensionais caóticos com borda fixa aplicados à modelagem de um sistema criptográfico para imagens digitais," *Revista de Informática Teórica e Aplicada*, vol. 23, no. 1, pp. 250–276, 2016.

[2] C. Cactus, "3 ways a vpn can make your life easier." http://www.criticalcactus.com/ways-a-vpn-can-make-your-life-easier/, 2015. Acessed: 2016-06-12.

[3] E. Silveira, *Tolerância a falhas em VPNs BGP/MPLS/VRF usando DMVPNs*. PhD thesis, Instituto Federal de Santa Catarina, 2013.

[4] L. H. Hammerle, "Survey da tecnologia mpls e suas aplicações," *CT - Teleinformática e Redes de Computadores*, 2015.

[5] A. K. Mishra and A. Sahoo, "S-ospf: A traffic engineering solution for ospf based best effort networks," in *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE*, pp. 1845–1849, IEEE, 2007.

[6] R. Verma and B. Bhushan, "Qos model for intranet area network based on media access control protocol over tcp connection," in *IT in Business, Industry and Government (CSIBIG), 2014 Conference on*, pp. 1–5, IEEE, 2014.

[7] L. D. Ghein, *MPLS fundamentals*. Cisco Press, 2007.

[8] A. T. C. Andrade, "Modelagem e anãlise de desempenho de uma rede baseada em tecnologia mpls," *RepositÓrio de RelatÓrios de Sistemas de Informação*, no. 2, 2014.

[9] B. S. Davie and Y. Rekhter, *MPLS: technology and applications*. Morgan Kaufmann Publishers Inc., 2000.

[10] K. Sethom, H. Afifi, and G. Pujolle, "Wireless mpls: a new layer 2.5 micro-mobility scheme," in *Proceedings of the second international workshop on Mobility management & wireless access protocols*, pp. 64–71, ACM, 2004.

[11] A. R. Sharafat, S. Das, G. Parulkar, and N. McKeown, "Mpls-te and mpls vpns with openflow," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 452–453, 2011.

[12] R. Aggarwal, K. Kompella, T. Nadeau, and G. Swallow, "Bidirectional forwarding detection (bfd) for mpls label switched paths (lsps)," tech. rep., 2010.

[13] M. Bocci, S. Bryant, D. Frost, L. Levrau, and L. Berger, "A framework for mpls in transport networks," tech. rep., 2010.

[14] E. Rosen and R. Aggarwal, "Multicast in mpls/bgp ip vpns," tech. rep., 2012.

[15] L. Fang, N. Bitar, R. Zhang, M. Daikoku, and P. Pan, "Mpls transport profile (mpls-tp) applicability: Use cases and design," tech. rep., 2013.

[16] S. Pichumani and R. Aggarwal, "Inter-site pim-dense mode and pim-bsr support for mpls/bgp ip vpns," Aug. 4 2015. US Patent 9,100,201.

[17] B. Niven-Jenkins, D. Brungard, M. Betts, N. Sprecher, and S. Ueno, "Requirements of an mpls transport profile," tech. rep., 2009.

[18] B. A. Forouzan, *TCP/IP protocol suite*. McGraw-Hill, Inc., 2002.

[19] S. Skiena, "Dijkstra's algorithm," *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica, Reading, MA: Addison-Wesley*, pp. 225–227, 1990.

[20] T. J. Misa and P. L. Frana, "An interview with edsger w. dijkstra," *Communications of the ACM*, vol. 53, no. 8, pp. 41–47, 2010.

[21] A. T. Van Zoest, M. J. DiMeo, B. M. Degenhardt, C. L. Sismondo, B. Callahan, J. W. DeRose, G. M. Costello, T. A. Barnum, J. Park, J. Stephens, *et al.*, "System and method for providing access to electronic works," Dec. 17 2002. US Patent 6,496,802.

[22] J. M. S. Pinheiro, "Segurança em redes privadas virtuais." http://www.projetoderedes.com.br/artigos/artigo_seguranca_vpn.php, 2004. Acessed: 2016-06-24.

[23] UFRJ, "Redes virtual private networking e ipsec." http://www.gta.ufrj.br/grad/04_1/vpn/Script/RDITunelamento.html, 2004. Acessed: 2016-06-24.

[24] M. Technet, "Virtual private networking: An overview." https://technet.microsoft.com/en-us/library/bb742566.aspx, 2001. Acessed: 2016-06-24.