



# MPLS: Teoria e Implementação Utilizando L2VPNs

# MPLS: Multiprotocol Label Switching

Tecnologia baseada em **comutação** de pacotes para a implantação de **múltiplos serviços** através da seleção e aplicação de **QoS**, sendo o serviço principal a **VPN**.

Utiliza-se de marcações específicas chamadas de **LABELs** que determinam como serão transmitidos os pacotes.

Dentro da nuvem MPLS **não há consulta aos endereços IPs**, apenas as LABELs direcionam o pacote a ser transmitido, pois não há intervenção de outros protocolos. O controle das rotas IPs continua sendo realizado por protocolos de roteamento, como por exemplo, OSPF.

# MPLS: Multiprotocol Label Switching

**LDP**

**RSVP**

*Infraestrutura*

**VPWS**

**VPLS**

**H-VPLS**

*Serviço L2VPN*

**P2P**

**M2P**

**Hub-Spoke**

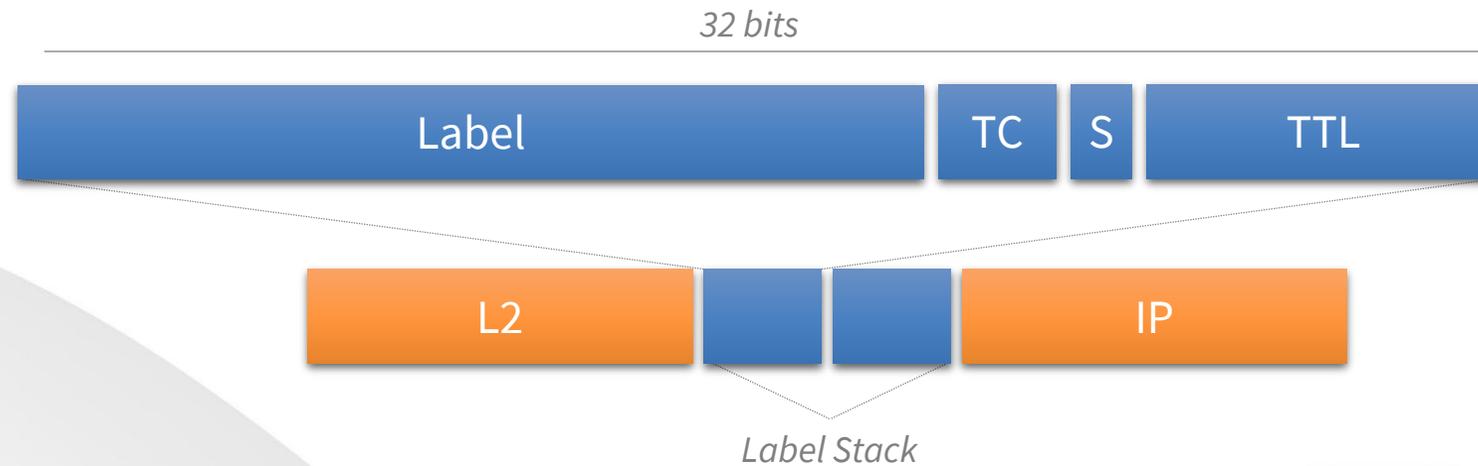
*Serviço L3VPN*

# MPLS: Benefícios

1. Método simples de transporte de **múltiplos protocolos** em uma rede. Pode transportar IPv4, IPv6, Ethernet e outras tecnologias L2;
2. O provedor de serviço utiliza apenas **uma única infraestrutura** para transportar todo o tipo de serviço;
3. O provisionamento dos serviços é feito **apenas nos PEs**;
4. Maior **desempenho** e **disponibilidade** da rede;
5. Facilidade de **crescimento** e ajuste a **novas tecnologias**;
6. Rede **livre de loop**;
7. **Traffic Engineering (TE)**: Escolhe caminhos diferentes daqueles eleitos pelo roteamento (source-based routing);
8. **Fast Reroute (FRR)**: Garante tempos de convergência inferiores a 50ms.

# MPLS: Shim Header

É o cabeçalho do **MPLS**, elemento fundamental desta tecnologia. Aloca de forma dinâmica a label.



Campo	Descrição
<b>Label</b>	Utilizado para decisão de encaminhamento
<b>TC ou EXP</b>	Qualidade de Serviço
<b>S</b>	Bit de “stack” utilizado para o empilhamento de cabeçalhos MPLS (Label Infra e Label VPN)
<b>TTL</b>	Tempo de vida do pacote

# MPLS: Shim Header

Os valores dos campos EXP, S e TTL são obtidos no primeiro cabeçalho da pilha MPLS. Através de um analisador de tráfego de redes, é possível verificar a estrutura do cabeçalho MPLS.

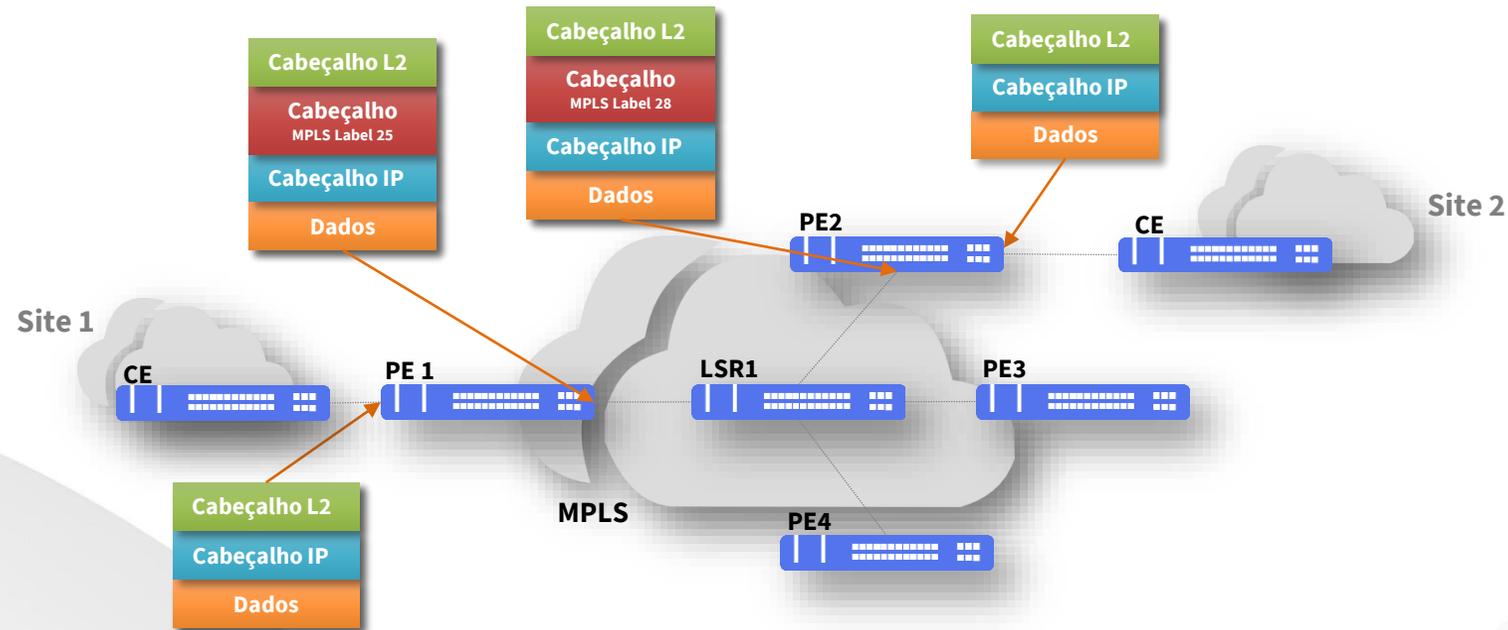
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.0.1	192.168.0.2	IPv4	82	Unknown (253)
<b>Frame 1: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface 0</b>						
Ethernet II, Src: TeracomT_5e:31:28 (00:04:df:5e:31:28), Dst: TeracomT_5e:2b:5c (00:04:df:5e:2b:5c)						
Destination: TeracomT_5e:2b:5c (00:04:df:5e:2b:5c)						
Source: TeracomT_5e:31:28 (00:04:df:5e:31:28)						
Type: MPLS label switched packet (0x8847)						
<b>MultiProtocol Label Switching Header, Label: 25, Exp: 0, S: 0, TTL: 255</b>						
0000 0000 0000 0001 1001 ..... = MPLS Label: 25						
..... 000. .... = MPLS Experimental Bits: 0						
..... 0 .... = MPLS Bottom Of Label Stack: 0						
..... 1111 1111 = MPLS TTL: 255						
<b>MultiProtocol Label Switching Header, Label: 27, Exp: 0, S: 1, TTL: 255</b>						
0000 0000 0000 0001 1011 ..... = MPLS Label: 27						
..... 000. .... = MPLS Experimental Bits: 0						
..... 1 .... = MPLS Bottom Of Label Stack: 1						
..... 1111 1111 = MPLS TTL: 255						
Ethernet II, Src: TeracomT_1d:b6:26 (00:04:df:1d:b6:26), Dst: Broadcast (ff:ff:ff:ff:ff:ff)						
802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 1000						
000. .... = Priority: Best Effort (default) (0)						
...0 .... = CFI: Canonical (0)						
.... 0011 1110 1000 = ID: 1000						
Type: IP (0x0800)						
Internet Protocol Version 4, Src: 192.168.0.1 (192.168.0.1), Dst: 192.168.0.2 (192.168.0.2)						
Data (22 bytes)						

Label de infraestrutura

Qualidade de Serviço

Empilhamento de labels  
 S=0 - Infraestrutura  
 S=1 - Serviço VPN

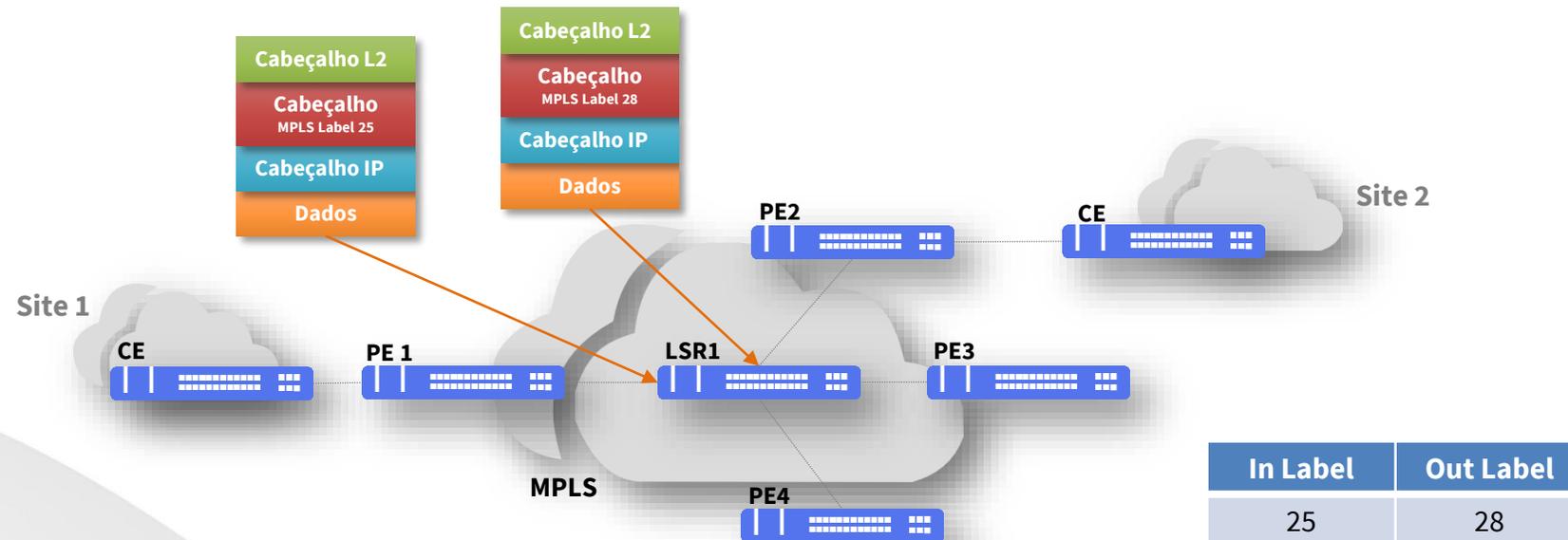
# MPLS: PE – Provider Edge



São os pontos de **entrada** e **saída** do domínio MPLS, com o objetivo de **adicionar** ou **retirar** o cabeçalho MPLS/Labels.

- Denominado como equipamento que recebe as conexões dos CEs – *Customer Edges*;
- Também conhecido por LER - *Label Edge Router* ou *Edge LSR*;
- **Adiciona** cabeçalho MPLS quando um pacote ingressa no domínio MPLS - **pushing** e **remove** cabeçalho MPLS quando um pacote deixa o domínio MPLS - **poping**.

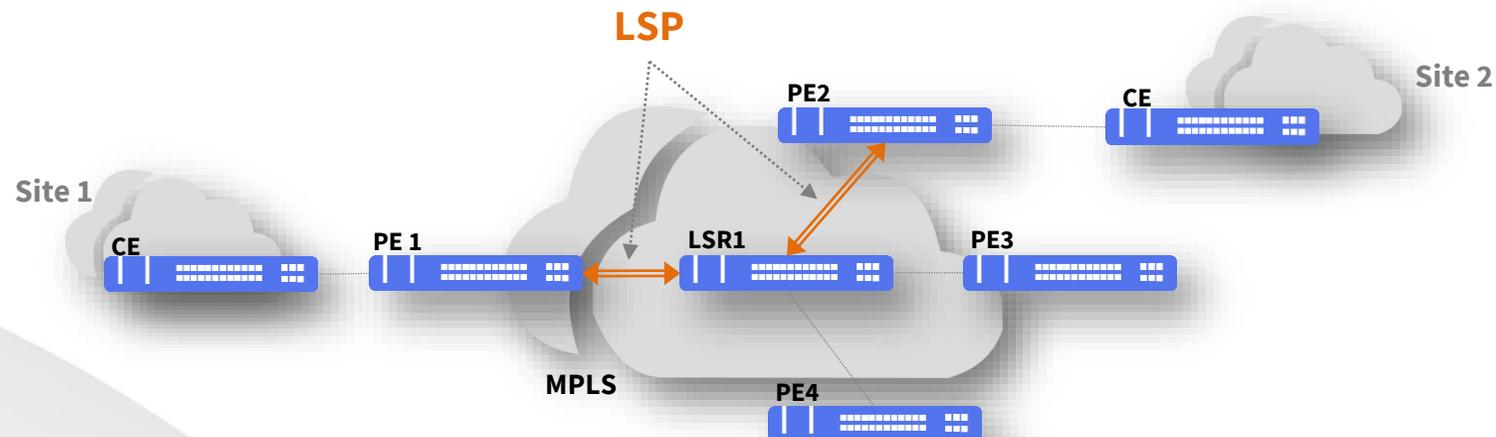
# MPLS: LSR – Label Switch Router



Dispositivo que exerce a **função de inspecionar o label de entrada e mapeá-lo em label de saída.**

- A decisão de encaminhamento é baseada exclusivamente nos labels;
- Denominado também como P - *Provider* ou *Intermediate* LSR;
- Sempre ao passar por um LSR o pacote sofrerá uma ação, conforme a sua posição na topologia – Push / Swap / Pop;
- Os labels possuem significado local;
- Não considera as informações encapsuladas (camada 3).

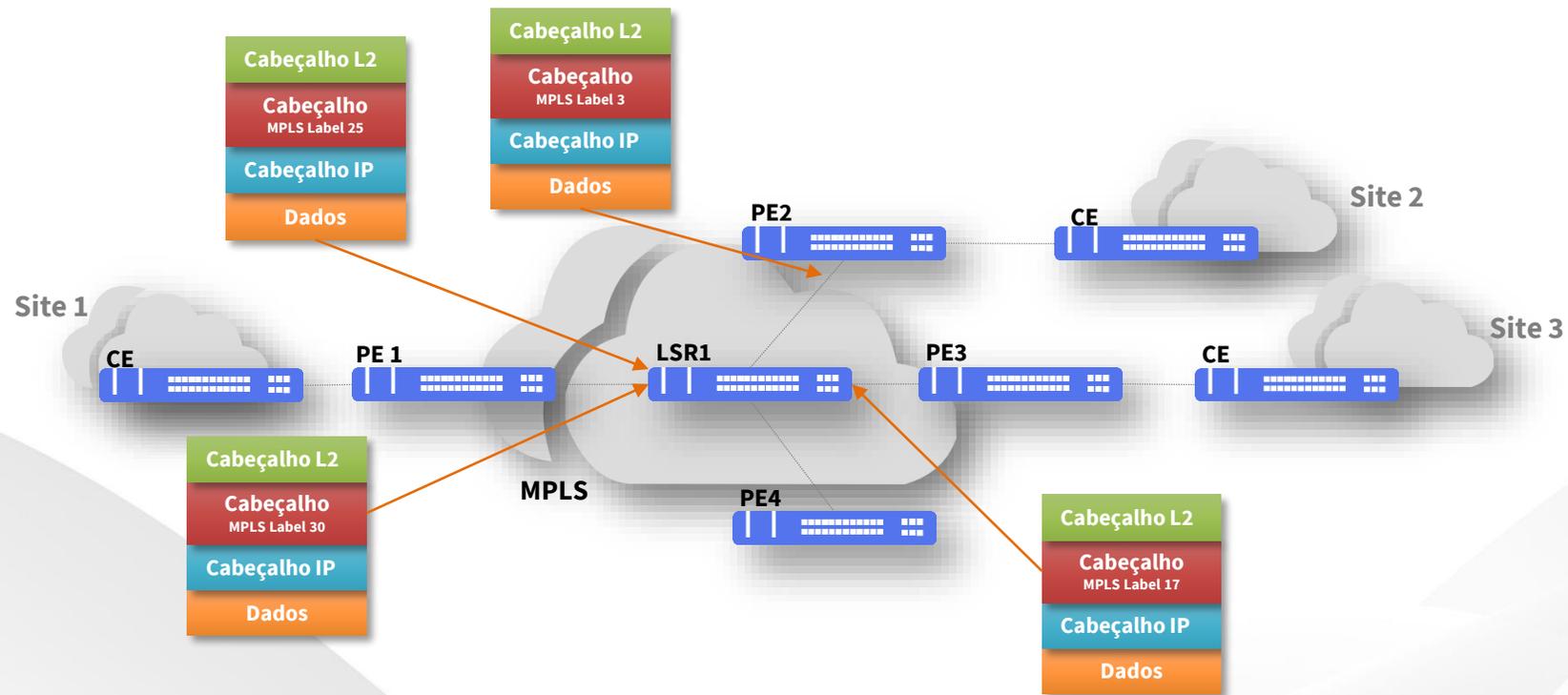
# MPLS: LSP – Label Switch Path



É o caminho percorrido pelo MPLS entre dois LSRs. Um LSP é **unidirecional** e contém todas as sequências de labels que serão utilizadas para encaminhar o tráfego.

- Ao associar uma rede a um caminho (LSP), todos os pacotes destinados àquela rede serão encaminhados com a mesma sequência de labels (mesmo túnel).

# MPLS: ILM – Incoming Label Mapping

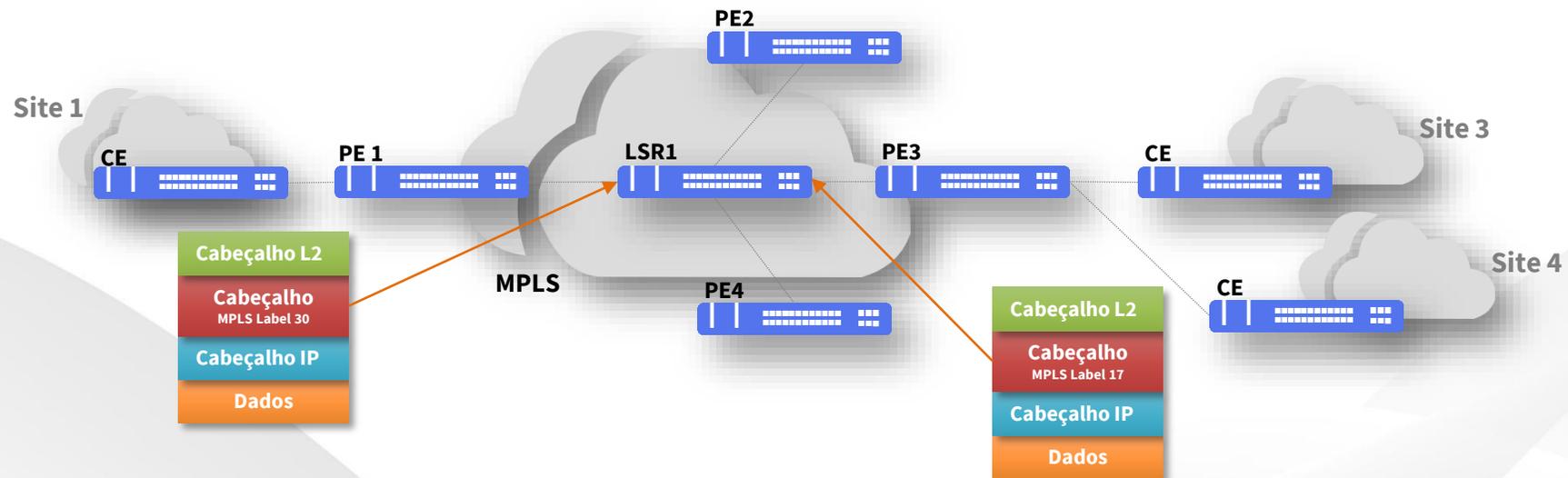


## Mapeamento de label de entrada.

- Determina as ações para cada pacote, podendo eliminar, trocar ou trocar e inserir;
- Mapeamento sempre 1:1 para LSR.

Neighbor	In Label	Action	Out Label
PE2	25	PHP	3
PE3	30	SWAP	17

# MPLS: FEC – Forwarding Equivalent Class

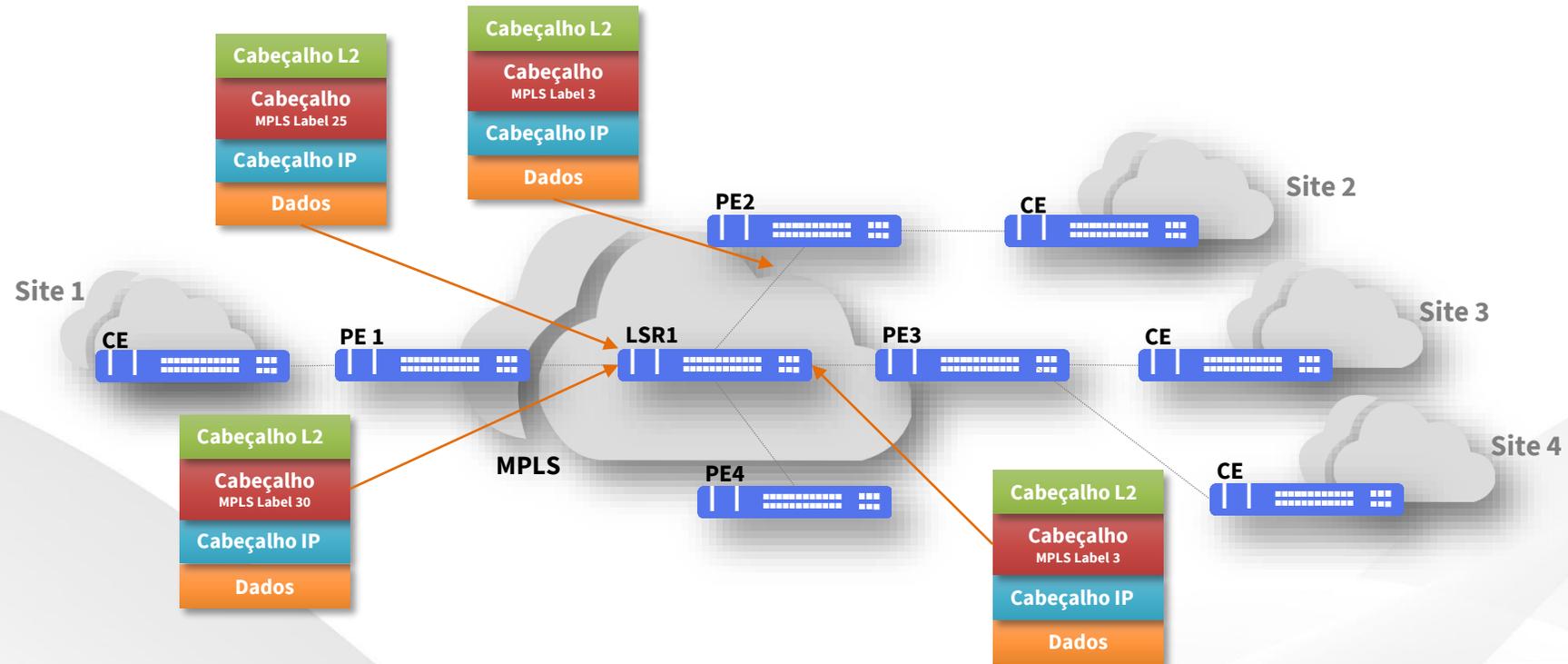


Consiste no **agrupamento** de pacotes que possuem **algum tipo de equivalência** com o objetivo de **fornecer o mesmo tratamento / mesmo label**.

O dispositivo ao receber um pacote faz a verificação de qual FEC pertence e o encaminha através de um LSP correspondente.

Neighbor	In Label	Action	Out Label
PE3 / Site 3	30	SWAP	17
PE3 / Site 4	30	SWAP	17

# MPLS: PHP – Penultimate Hop Popping



A função PHP consiste na **retirada** do cabeçalho MPLS referente a *outer label* no penúltimo LSR da topologia, não comprometendo o funcionamento de um LSP. Com esta técnica, o PE inspeciona apenas o *inner label* (label do serviço MPLS).

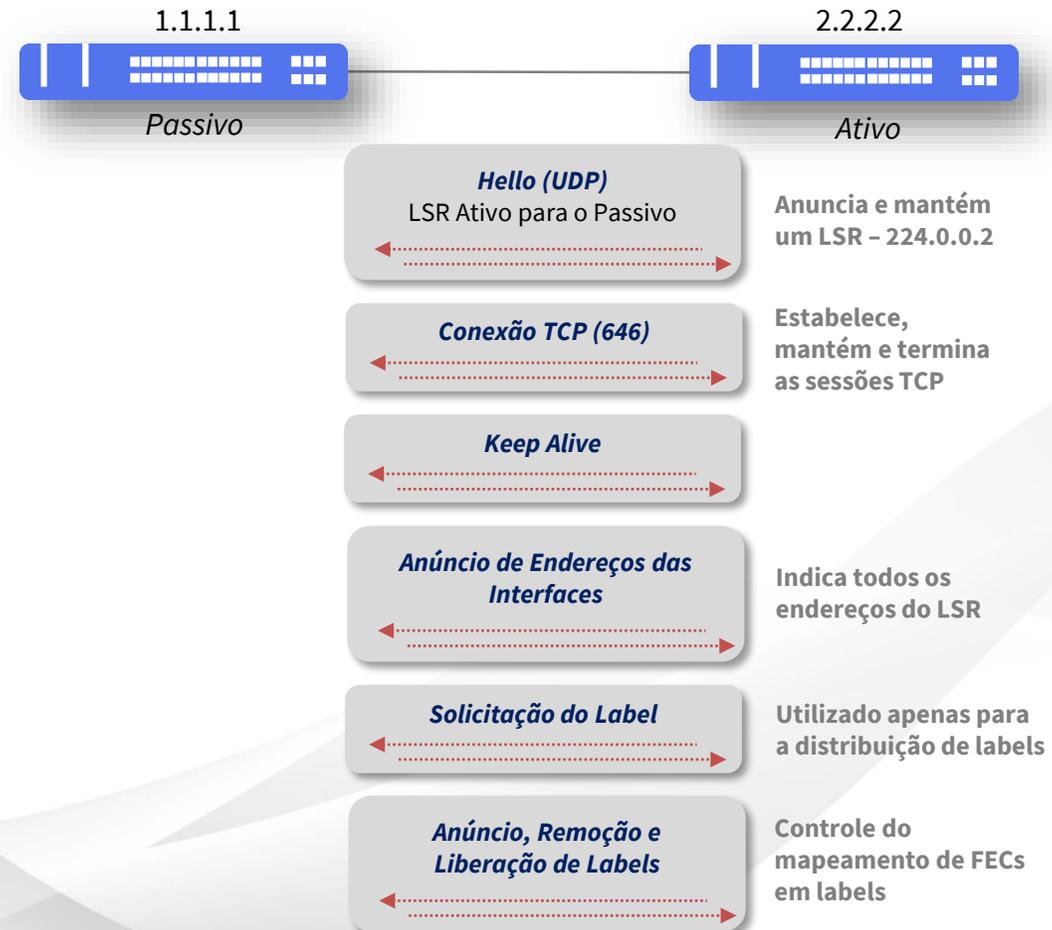
Label	Nome	Descrição
3	Implicit null	Uso exclusivo do LDP, sinaliza ao penúltimo dispositivo a retirada do label antes do envio (PHP)

# MPLS: LDP – Label Distribution Protocol

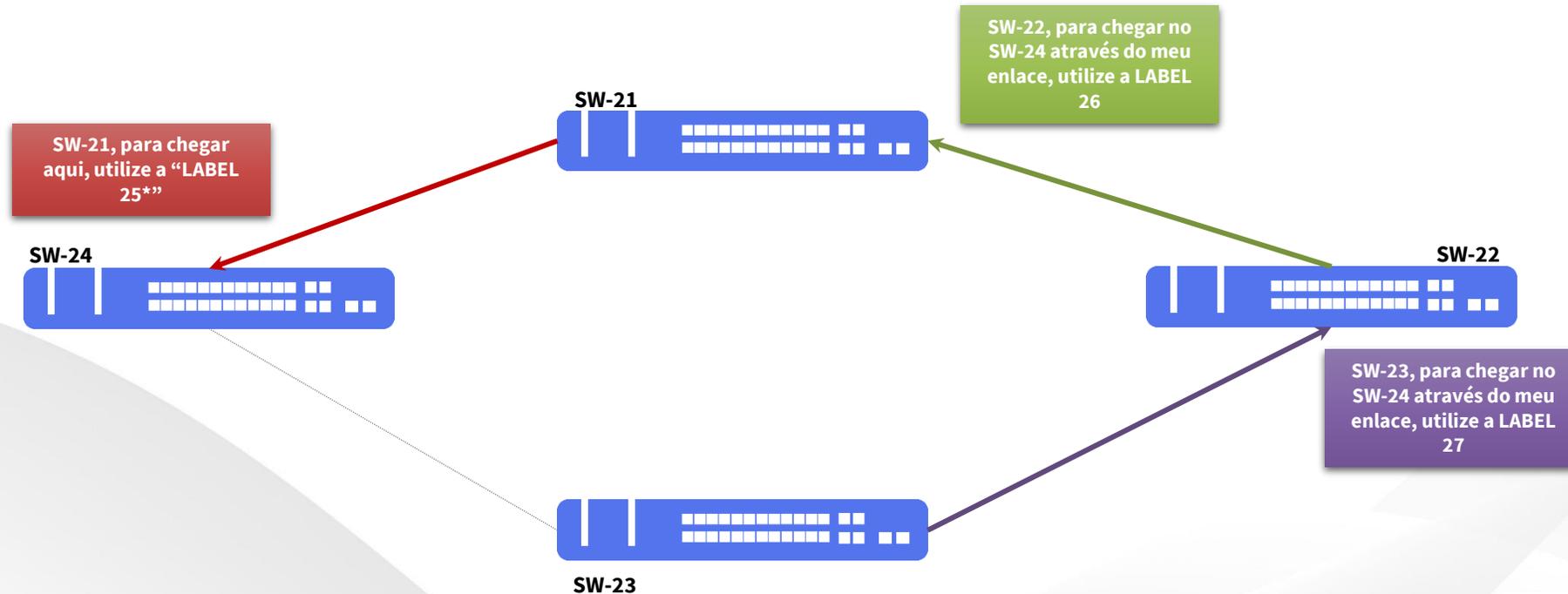
Protocolo **específico** e **responsável** pela **distribuição** de labels, desenvolvido pelo IETF nas RFCs 3036 e 3037, os quais são atribuídos para cada prefixo IGP aprendido na tabela de roteamento.

Para as rotas direcionadas para o mesmo destino tem-se os mesmos rótulos, pois utiliza o conceito da FEC.

A sua configuração é executada nos **dispositivos diretamente** conectados com o com o intuito de formar uma **adjacência**, o qual é estabelecida através de um pacote com endereço de **multicast 224.0.0.2** e com TTL igual a um.



# MPLS: Aprendizagem das Labels



**LSP do SW-23 até o SW-24**

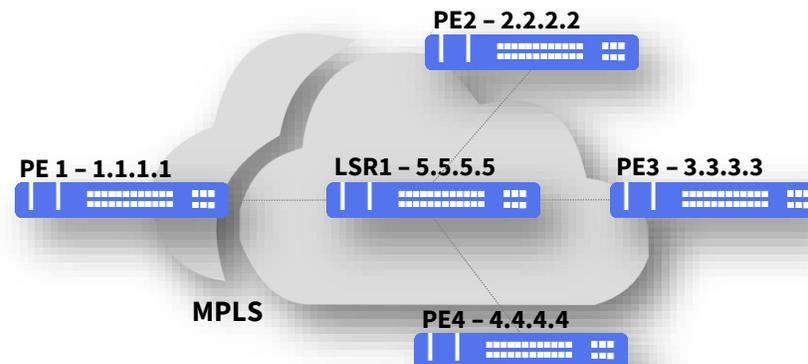
**LABEL 27 >> LABEL 26 >> LABEL 25**

*O LSP é unidirecional, então teremos LABELs de IN e OUT*

\*PHP label especial: 3

# MPLS: LDP – Label Distribution Protocol

Especialmente desenhado para **estabelecimento de caminhos MPLS**. A sinalização ocorre na direção downstream e upstream.

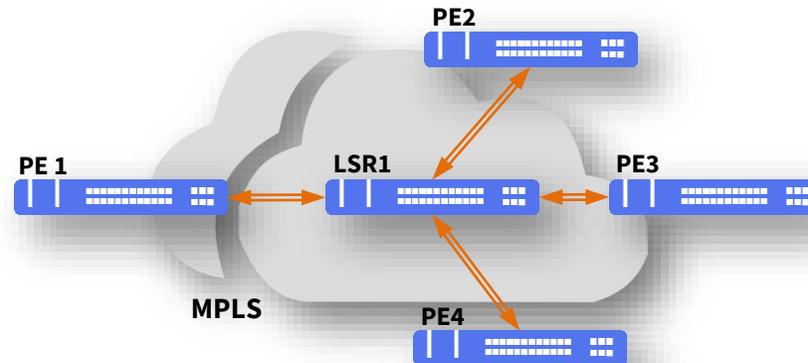


*\*\* Estão sendo demonstradas os labels do PE1*

*\*\* Foram omitidas algumas saídas*

Prefix	Action	In Label	Out Label	Out Protocol	Out Interface	Status
2.2.2.2	SWP	44	28	ldp	l3-vlan 15	Active
3.3.3.3	SWP	61	36	ldp	l3-vlan 15	Active
4.4.4.4	SWP	57	33	ldp	l3-vlan 15	Active
5.5.5.5	PHP	21	ImpNull	ldp	l3-vlan 15	Active
2.2.2.2	PSH	-	28	ldp	l3-vlan 15	Active
3.3.3.3	PSH	-	36	ldp	l3-vlan 15	Active
4.4.4.4	PSH	-	33	ldp	l3-vlan 15	Active
5.5.5.5	FWD	-	ImpNull	ldp	l3-vlan 15	Active

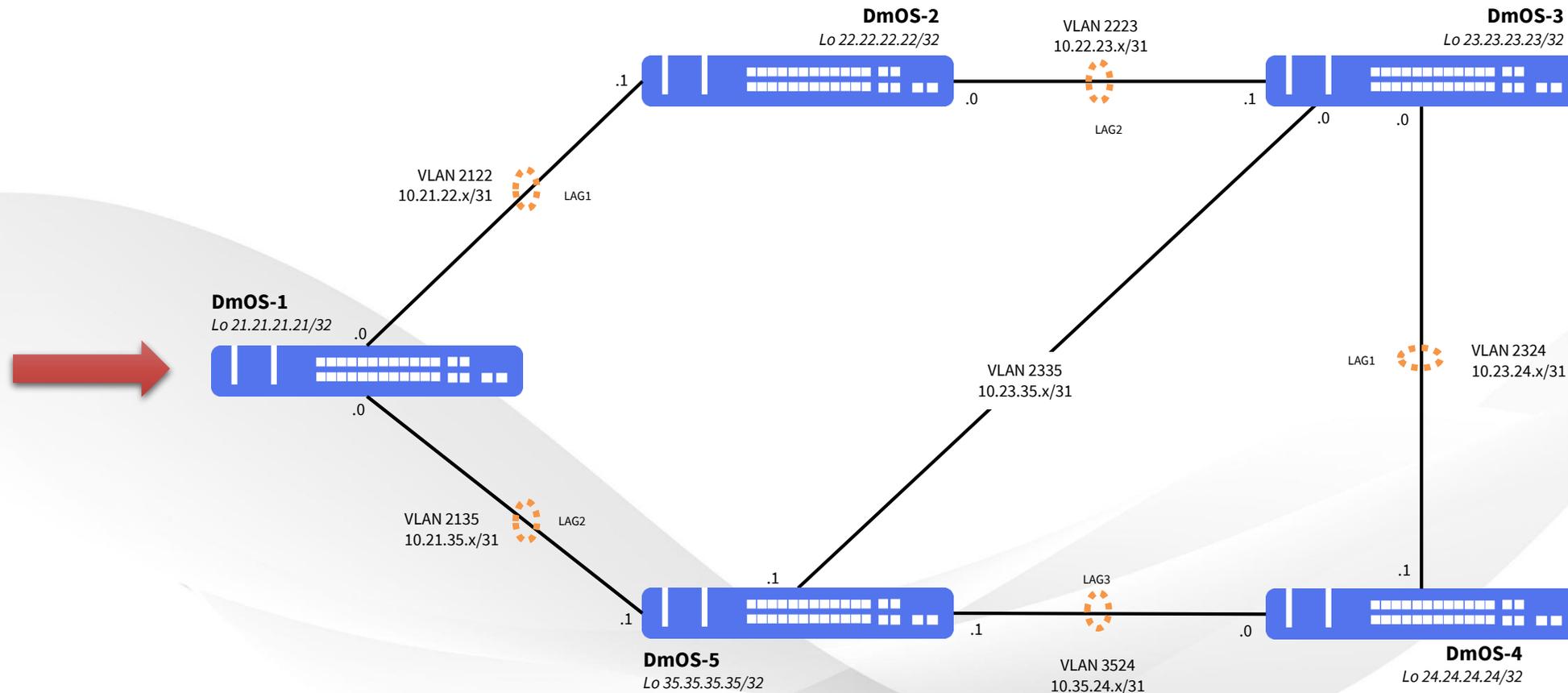
# MPLS: LDP – Label Distribution Protocol



## Características

- Trabalha em conjunto com os protocolos de roteamento IP (OSPF, BGP,...);
- Tempo de convergência depende do protocolo de roteamento;
- Há labels para o sentido downstream e outras para o sentido upstream.

# MPLS LDP: Prática



# MPLS: Configurações para Infraestrutura LDP

## Apontar a interface loopback como LSR-ID

```
DmOS(config)#mpls ldp lsr-id <loopback_id>
```

## Especificar a interface L3 utilizada para a infraestrutura (linked)

```
DmOS(config-lsr-id-loopback-0)#interface l3-<name>
```

## Definir o endereço de loopback do equipamento vizinho (targeted)

```
DmOS(config-lsr-id-loopback-0)#neighbor targeted <ip_loopback_neighbor>
```

## Autenticação MD5 para a sessão LDP

```
DmOS(config-neighbor-2.0.0.2)#password <password>
```

## Verificar as configurações

```
DmOS#show running-config mpls ldp  
DmOS#show running-config mpls ldp | tab  
DmOS#show mpls ldp <database | neighbor | parameters>
```



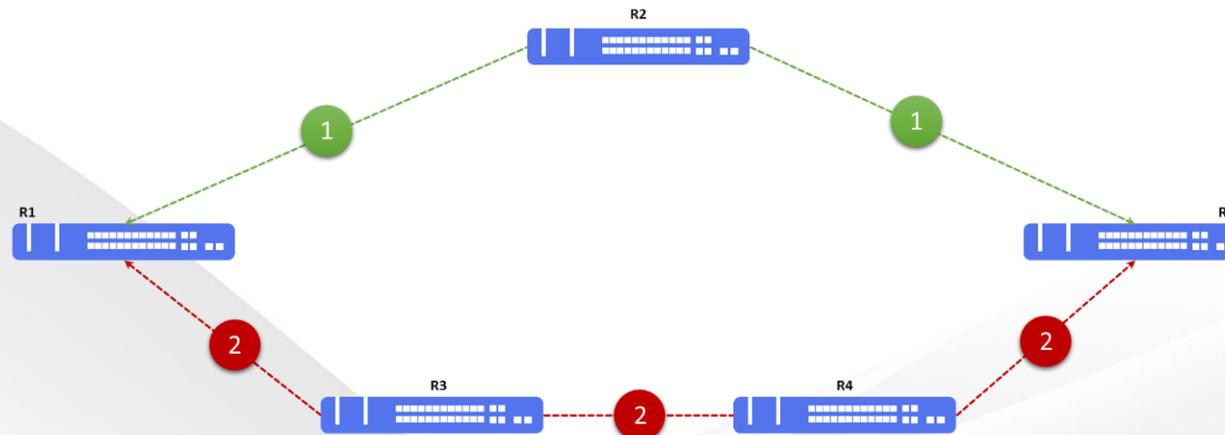
# MultiProtocol Label Switching Traffic Engineering

# Por que Traffic Engineering?

Uma das **características** dos protocolos de roteamento **é a de realizarem a utilização de links com maior banda**, o que ocasiona um tráfego ou utilização maior destes links, deixando outros ociosos.

Utilizando um IGP - *Interior Gateway Protocol* como o OSPF, o caminho escolhido entre dois pontos sempre será o caminho com o menor custo.

Na topologia abaixo, considerando que todos os links possuem o mesmo custo, pacotes com origem em **R1** e destino **R5** sempre tomarão o caminho mais curto: R1-R2-R5.



Mesmo que o caminho **1** esteja saturado, ele será utilizado para todo o tráfego entre R1 e R5, e o caminho **2** ficará subutilizado.

# Por que Traffic Engineering?

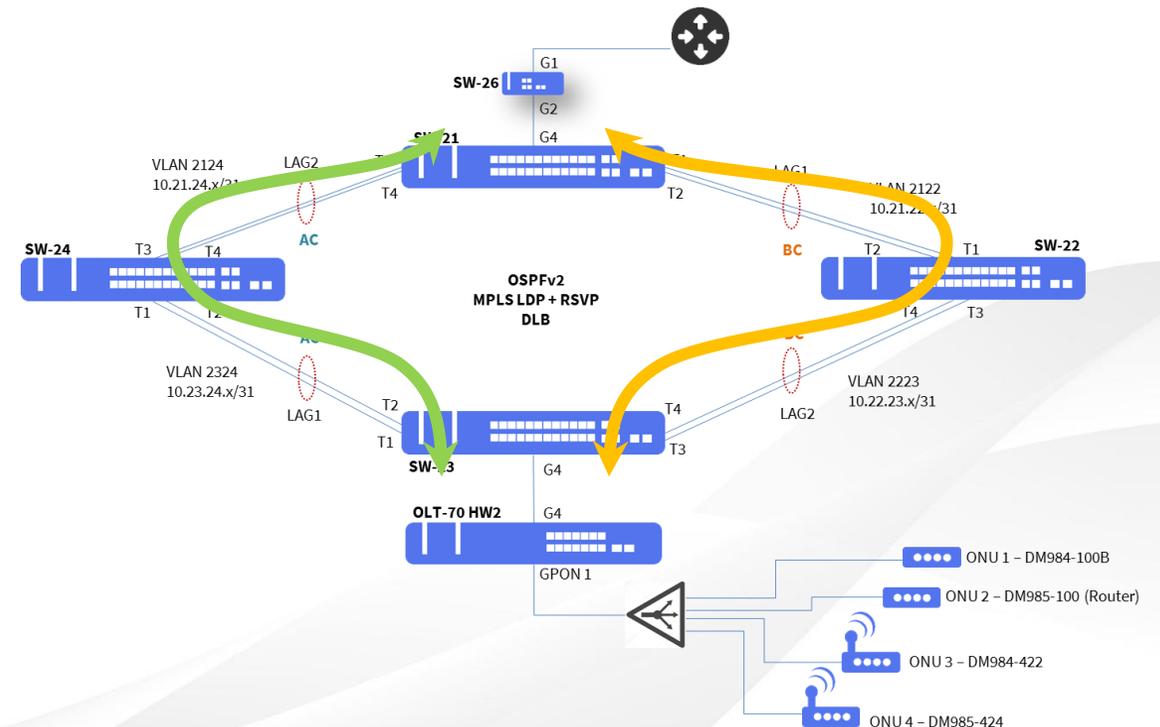
A ideia da engenharia de tráfego em redes MPLS é a de **buscar o uso mais eficiente** dos recursos presentes na infraestrutura do provedor ou da operadora, como equipamentos já instalados, a redução da ociosidade de links próprios ou contratados e do melhor balanceamento da carga do tráfego entre as diversas velocidades das interfaces/transceivers.

# RSVP: Resource reSerVation Protocol

O protocolo RSVP é utilizado para o **estabelecimento de túneis entre os equipamentos**.

Diferentemente do LDP, que utiliza o melhor caminho escolhido pelo roteamento para alcançar o destino, no RSVP o caminho dos túneis é configurado pelo administrador da rede.

O RSVP com a **extensão TE - Traffic Engineering** é definido através da **RFC 3209**, onde incorpora ao protocolo a permissão para o estabelecimento de **LSPs** em **MPLS**.



# Nomenclatura do RSVP

O túnel TE é **unidirecional**, similar ao comportamento do **LSP** – *Label Switch Path*. Apenas o dispositivo chamado de **Head End** recebe a configuração facilitando assim a administração das VPNs e todo o encaminhamento é baseado em labels.

Head end LSR



Tail end LSR



# OSPF – TE: LSA Type 10 opaque

Para a operação correta do MPLS-TE é necessário que o IGP - *Interior Gateway Protocol* seja capaz de enviar o estado do link aos demais dispositivos presentes na área em que o TE for habilitado.

A **RFC 3630** especifica a **adição de extensões ao protocolo OSPF** para a operação e a **RFC 2370** define as **melhorias** do protocolo OSPF para **suportar uma nova classe de estado de link**, o LSA - *Link-State Advertisements* chamado de opaque.

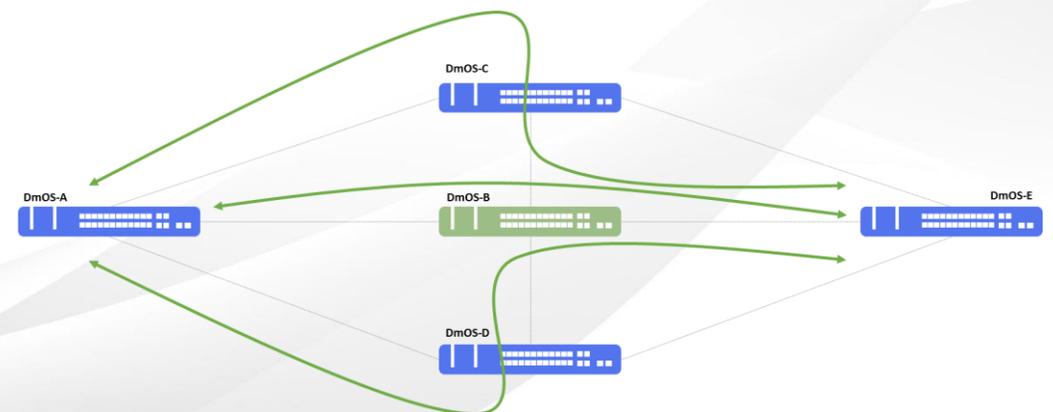
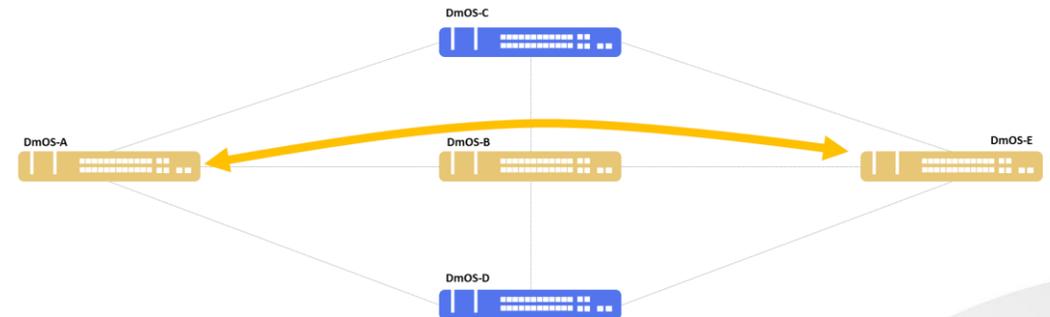
Para o MPLS-TE é utilizado em específico o **LSA type 10 opaque**, que abrange a sua atuação a **uma área OSPF**.

Age	Options	Type = 10
Opaque Type	Opaque ID	
Advertising Router		
Sequence Number		
Checksum	Length	
Opaque Information		

# Critérios para Estabelecimento de Túneis: Explicit Path

Como o próprio nome indica Explicit-Path é a definição de caminhos de forma **explícita** do **Head End** até o **Tail End**.

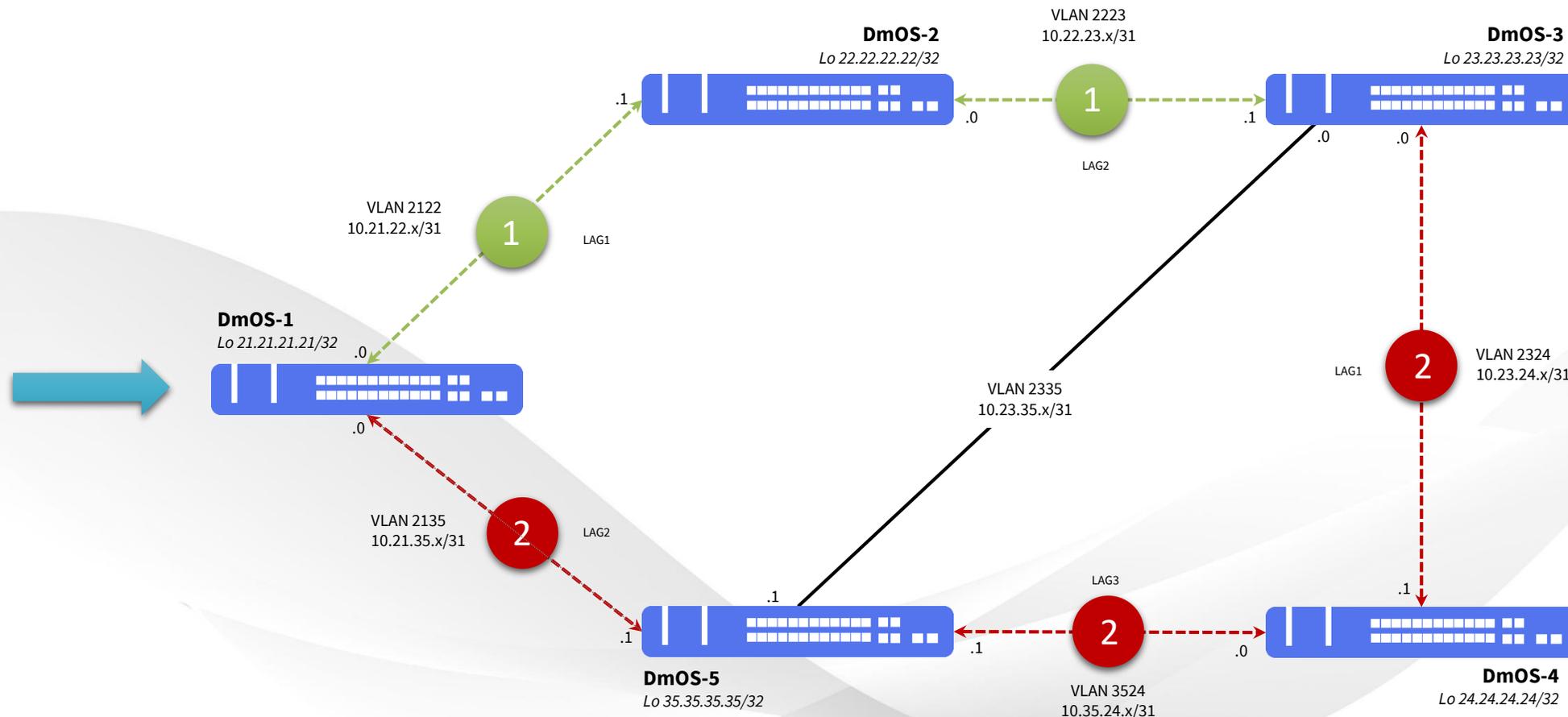
- **Strict Routes:** Consiste na declaração ordenada de todo o caminho a ser percorrido, salto-a-salto. O túnel obedecerá **fiel e rigorosamente** o caminho determinado e não o indicado pelo roteamento.
- **Loose Routes:** Não é necessário determinar todos os saltos entre o *Head End* e o *Tail End*, mas sim, **endereços IPs intermediários como pontos obrigatórios e estratégicos** para a formação do LSP. Os equipamentos não precisam estar conectados diretamente. A escolha do caminho, será uma combinação entre a tabela de roteamento e o endereço IP informado.



# Configurações Necessárias

1. Roteamento configurado – OSPF
2. Infra MPLS - LDP
3. Habilitar o MPLS-TE no roteamento OSPF
4. Associar as interfaces ao RSVP
5. Definir o caminho para Engenharia de tráfego
6. Definir a interface túnel e possíveis opções de caminhos
7. Associar o túnel ao L2VPN

# Critérios de Explicit-Path – Strict: Prática



# MPLS-TE, RSVP e Explicit-Path: Configurações

## Habilitar o MPLS-TE

```
DmOS (config) #router ospf <1-65535>
```

```
DmOS (config-ospf-21-vrf-global) #mpls-te router-id loopback-<0-7>
```

*'router ospf 21 global': Enabling mpls-te will cause OSPF process to be restarted Proceed? [yes,no]*

## Associar as interfaces L3 ao RSVP

```
DmOS (config) #mpls rsvp
```

```
DmOS (config-rsvp) #interface l3-<name>
```

## Definir os caminhos explícitos

```
DmOS (config) #mpls traffic-eng
```

```
DmOS (config-traffic-eng) #explicit-path <name>
```

```
DmOS (config-explicit-path-CAMINHO_PRINCIPAL) #hop <1-65535> ipv4 next-address <ipv4_address> <loose | strict>
```

## Verificar as configurações

```
DmOS#show running-config router ospf mpls-te
```

```
DmOS#show running-config mpls rsvp
```

```
DmOS#show running-config mpls traffic-eng | tab
```

# Interface Túnel TE com Explicit-Path: Configurações

## Criação do túnel com engenharia de tráfego

```
DmOS(config)#interface tunnel-te <1-65535>  
DmOS(config-tunnel-te-1)#name <name_tunnel>  
DmOS(config-tunnel-te-1)#description <text>  
DmOS(config-tunnel-te-1)#destination <ip_loopback_destination>  
DMOS(config-tunnel-te-1)#path-option <1-255> explicit name <name_path_option>
```

**\*\*\* A criação do túnel é unidirecional, portanto é necessário realizar a mesma configuração no outro vizinho da VPN \*\*\***

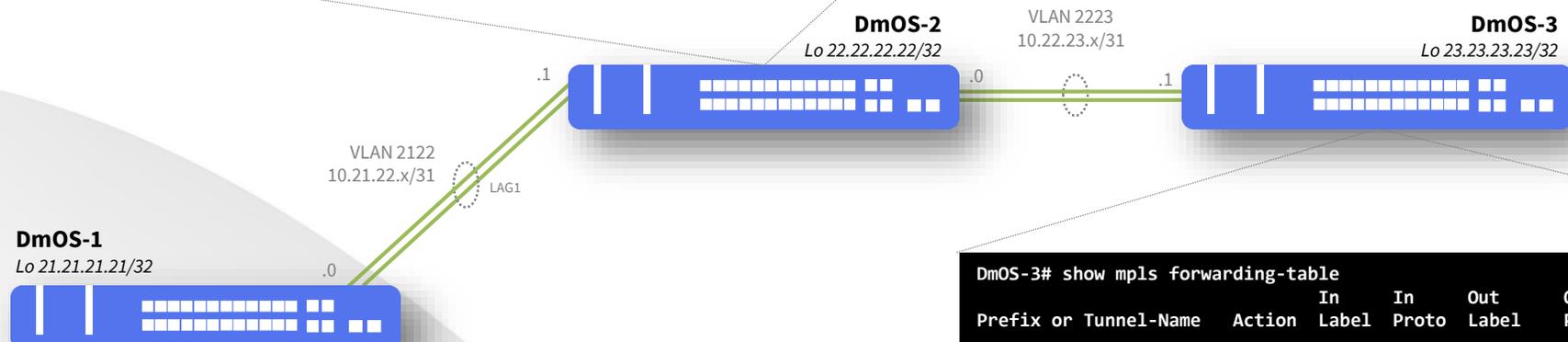
## Para verificar:

```
DmOS#show mpls forwarding-table  
DmOS#show running-config mpls traffic-eng | tab  
DmOS#show mpls traffic-eng tunnel-te <brief | id | name>  
DmOS#show running-config interface tunnel-te | tab
```

# Explicit-Path Strict: Verificações

```
DmOS-2# show mpls forwarding-table
```

Prefix or Tunnel-Name	Action	In Label	In Proto	Out Label	Out Proto	Out interface	Status
TUNEL-1_SW21_PARA_SW23	php	89	rsvp	ImpNull	rsvp	13-vlan 2223	active
TUNEL-1_SW23_PARA_SW21	php	88	rsvp	ImpNull	rsvp	13-vlan 2122	active



```
DmOS-3# show mpls forwarding-table
```

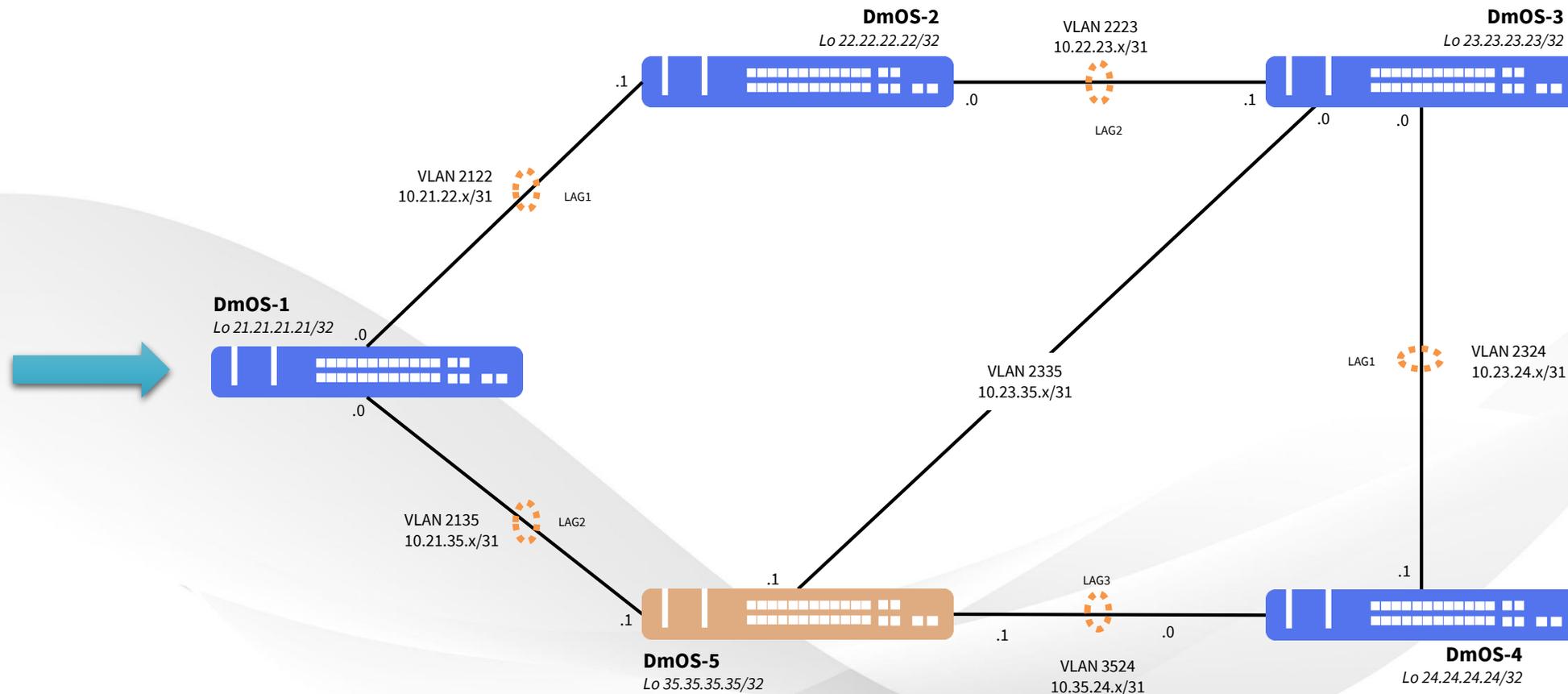
Prefix or Tunnel-Name	Action	In Label	In Proto	Out Label	Out Proto	Out interface	Status
TUNEL-1_SW23_PARA_SW21	psh	--	--	88	rsvp	13-vlan 2223	active

```
DmOS-1# show mpls forwarding-table
```

Prefix or Tunnel-Name	Action	In Label	In Proto	Out Label	Out Proto	Out interface	Status
TUNEL-1_SW21_PARA_SW23	psh	--	--	89	rsvp	13-vlan 2122	active

*\*Apenas as informações referentes ao protocolo RSVP, o protocolo LDP foi omitido da visualização*

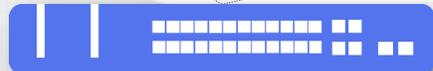
# Critérios de Explicit-Path – Loose: Prática



# Explicit-Path Loose: Verificações

```
DmOS-1# show mpls forwarding-table
Prefix or Tunnel-Name  Action  In  In  Out  Out  Out
-----
TUNEL-LOOSE_SW21_SW23  psh    -- --  129  rsvp  13-vlan 2135  active
```

**DmOS-1**  
Lo 21.21.21.21/32



VLAN 2135  
10.21.35.x/31

LAG2



**DmOS-5**  
Lo 35.35.35.35/32

VLAN 2335  
10.23.35.x/31

**DmOS-3**  
Lo 23.23.23.23/32



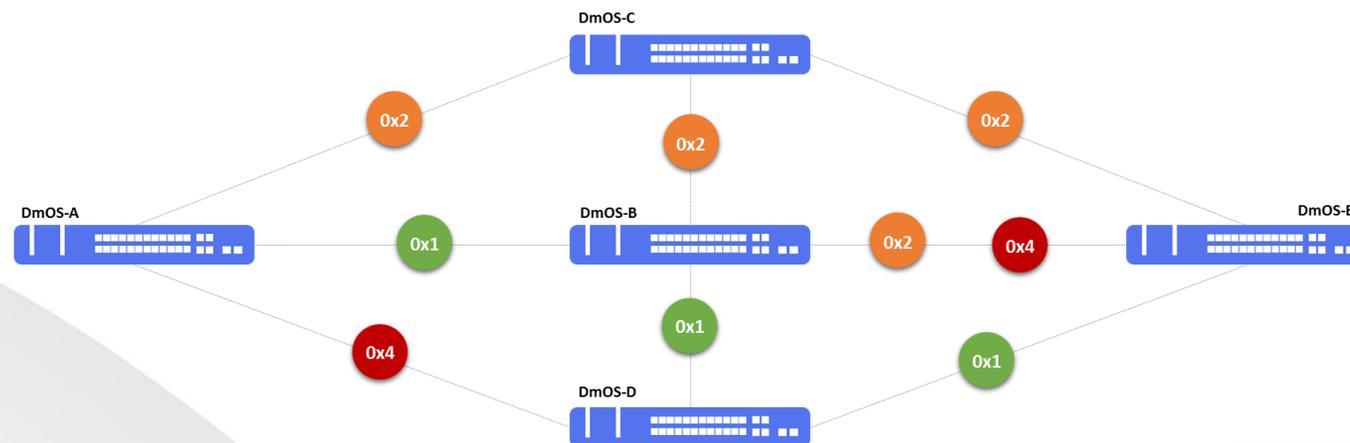
```
DmOS-3# show mpls forwarding-table
Prefix or Tunnel-Name  Action  In  In  Out  Out  Out
-----
TUNEL-LOOSE_SW23_SW21  psh    -- --  128  rsvp  13-vlan 2335  active
```

```
DmOS-5# show mpls forwarding-table
Prefix or Tunnel-Name  Action  In  In  Out  Out  Out
-----
TUNEL-LOOSE_SW21_SW23  php    129  rsvp  ImpNull  rsvp  13-vlan 2335  active
TUNEL-LOOSE_SW23_SW21  php    128  rsvp  ImpNull  rsvp  13-vlan 2135  active
```

\*Apenas as informações referentes ao protocolo RSVP, o protocolo LDP foi omitido da visualização

# Critérios para Estabelecimento de Túneis: Affinity

São atributos definidos em **forma de valores configurados e expressos em hexadecimal** (0x0 - 0xffffffff) nas interfaces associadas ao RSVP.



Os caminhos utilizados pelos túneis podem ser configurados para serem estabelecidos por links que possuem um valor de affinity específico ou que evitem links com determinados valores de affinity. Desta forma, não é necessário configurar manualmente os endereços pelos quais os túneis devem ser formados, tornando o processo de configuração menos trabalhoso.

Três parâmetros podem ser utilizados para definição de um caminho utilizando os atributos affinity: **include-any**, **include-all** e **exclude-any**.

# Hexadecimal x Binário

Hexadecimal	Binário
0	00000
1	00001
2	00010
3	00011
4	00100
5	00101
6	00110
7	00111

Hexadecimal	Binário
8	01000
9	01001
A	01010
B	01011
C	01100
D	01101
E	01110
F	01111

Hexadecimal	Binário
10	10000
11	10001
12	10010
13	10011
14	10100
15	10101
16	10110
17	10111

Hexadecimal	Binário
18	11000
19	11001
1A	11010
1B	11011
1C	11100
1D	11101
1E	11110
1F	11111

# Affinity: Definição dos Atributos do Link

Os atributos de Affinity podem ser relacionados com uma **coloração** a ser definida para os links, onde cada bit definido como "1" representa uma única cor.

Como consequência, será necessário fazer as contas/organizar as cores para segmentar em uma combinação. Por exemplo:

Tipo de Link	Cor		Binário	Hexadecimal
Alta Capacidade	Verde		00000001	0x1
Baixa Capacidade	Laranja		00000010	0x2
Alta Latência	Vermelho		00000100	0x4
Link de Rádio	Azul		00001000	0x8
Link Satélite	Preto		00010000	0x10

Um link pode possuir duas ou mais características simultaneamente: Baixa Capacidade e Alta Latência ao mesmo tempo. Neste caso, o seu atributo será a soma de 0x2 e 0x4, sendo seu valor final 0x6.

Os atributos podem conter valores de 0x0 a 0xFFFFFFFF, representando 32 atributos para classificação dos links, estes valores são de livre escolha.

# Affinity: Path Attribute e Link Attribute

## **Path Attribute**

Atributo considerado para a formação do caminho do túnel.

## **Link Attribute**

Atributo do link, conforme as suas características e informado pelo administrador da rede. Pode ser definido como a pontuação do link/enlace ou relacionado a uma cor.

# Affinity: Include Any

## Include-any

Para a utilização deste parâmetro é necessário que ocorra **match pelo menos em um dos bits**.

Por exemplo, caso o valor do include-any para o *Path attribute* seja 0x00000002 e um *link attribute* tenha valor de 0x00000006, o *match* ocorrerá no segundo bit, conforme destacado em vermelho, permitindo assim que este link seja utilizado nesta definição de caminho.

**Path attribute**    0x00000002 → 000000**10**

**Link attribute**    0x00000006 → 00000**110**

Este indicador utilizar o operador lógico **OR** (ou).

**Caso o valor do parâmetro include-any seja zero, o link é aceito independente do valor do seu atributo de affinity.**

# Affinity: Include All

## Include-all

Para o parâmetro **include-all** é necessário que ocorra *match* de **todos** os bits do *path*.

Para um *path attribute* definido como, 0x00000005 e um *link attribute* com valor de 0x00000006, observaremos que não ocorrerá o *match* em todos os bits, apenas o terceiro bit possuirá o mesmo valor, porém, o link não será utilizado para este caminho.

<b>Path attribute</b>	0x00000005 → 0000	<b>0101</b>
<b>Link attribute</b>	0x00000006 → 0000	<b>0110</b>

Este indicador utilizar o operador lógico **AND**(e).

**Caso o valor do parâmetro include-all seja zero, o link é aceito independente do valor do seu atributo de affinity.**

# Affinity: Exclude Any

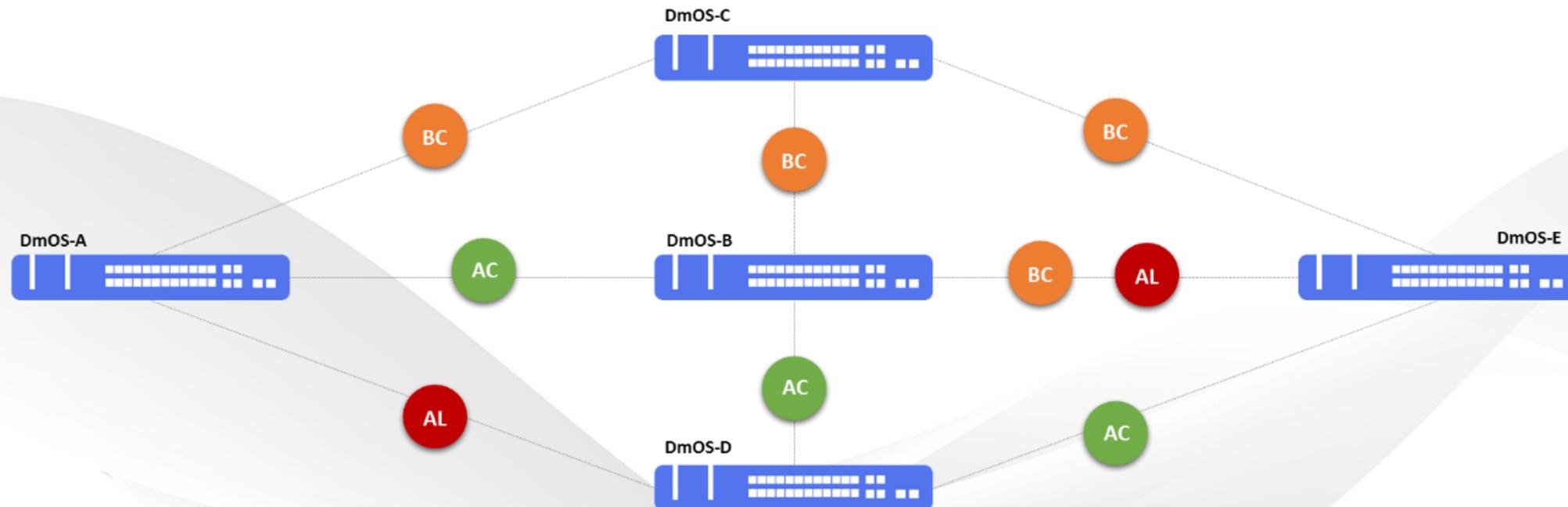
## Exclude-any

Links que possuam qualquer um dos bits especificados serão excluídos.

Este indicador utilizar o operador lógico **OR** (ou) para a exclusão de links.

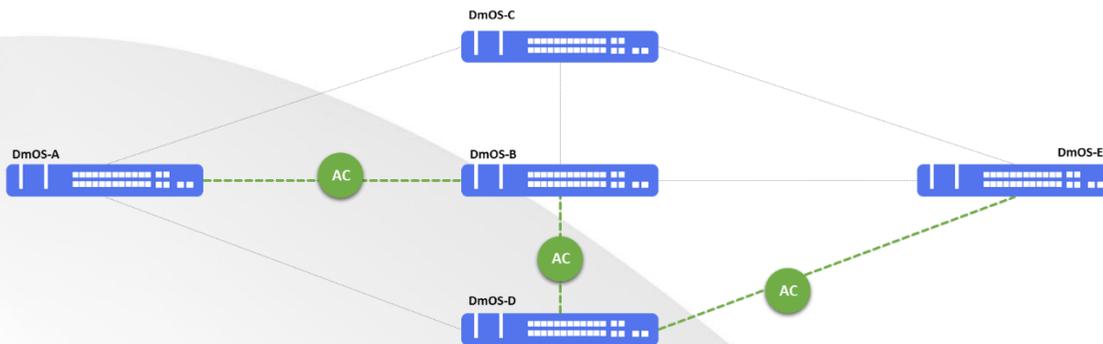
# Classificação dos Links : Critério Affinity

A topologia que utilizaremos como referência para as configurações e para a classificação de cada link pode ser observada abaixo.

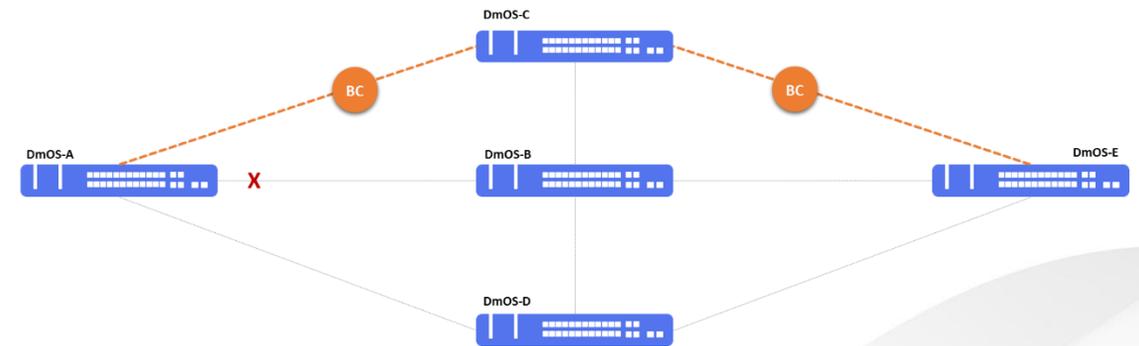


# Funcionamento: Critério Affinity

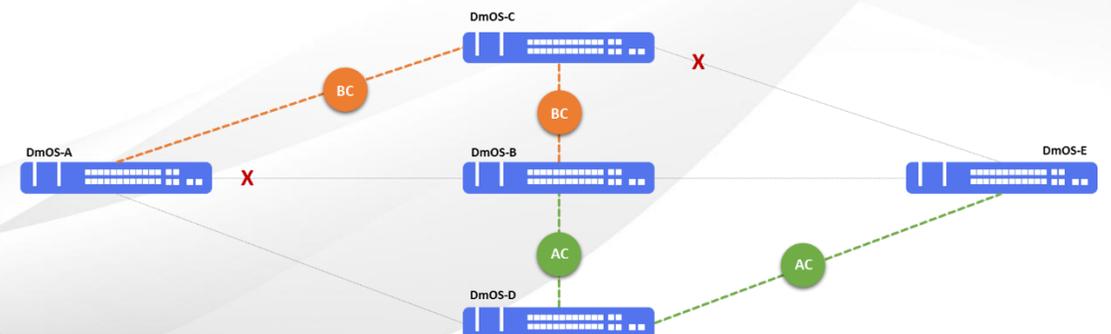
*Túnel utilizando apenas os caminhos de alta capacidade:*



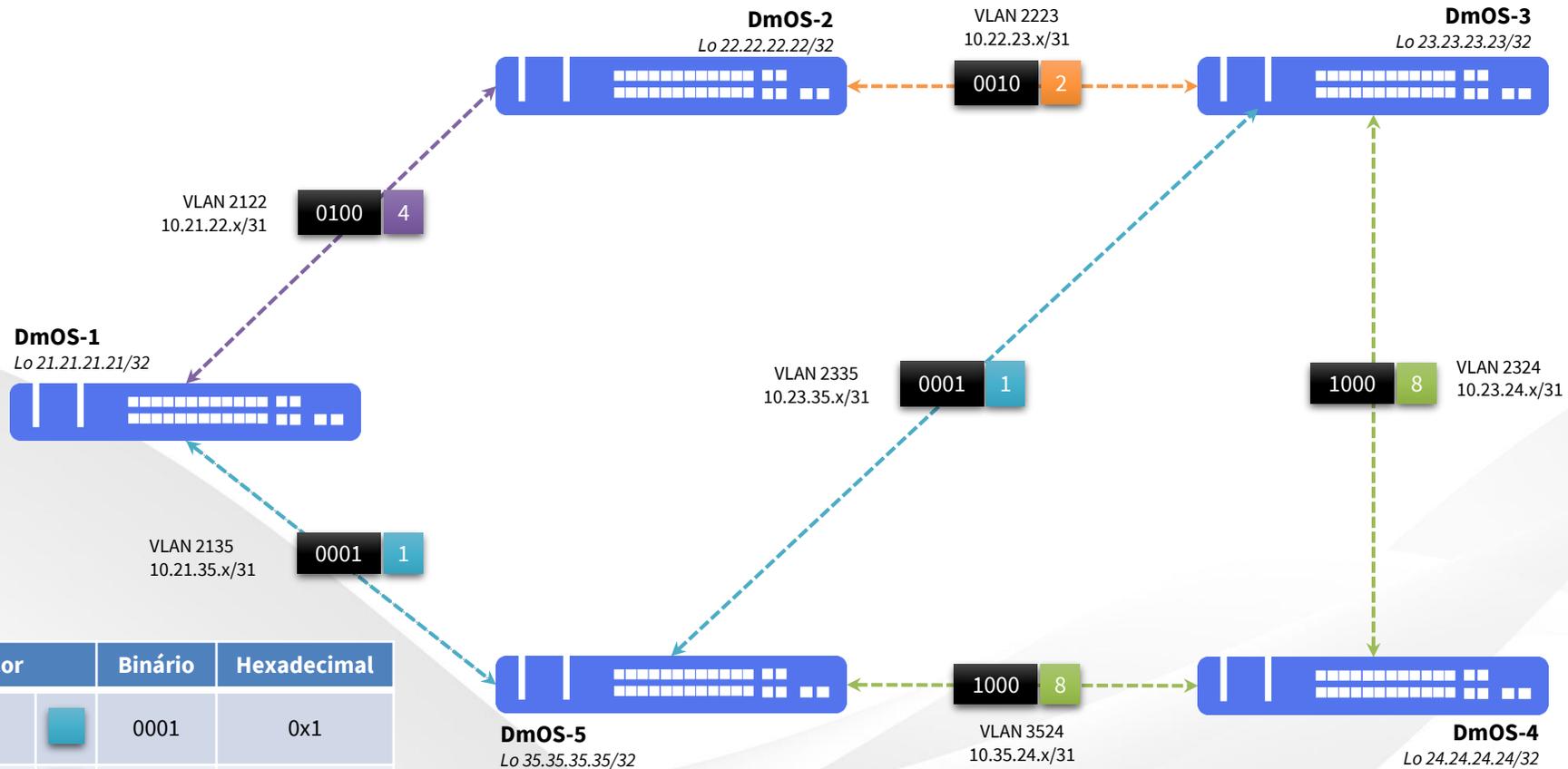
*Rompimento entre DmOS-A e DmOS-B:*



*Rompimento entre DmOS-C e DmOS-E:*



# Critérios de Affinity: Prática



Tipo de Link	Cor	Binário	Hexadecimal
Alta Capacidade (40G +LAG)	Azul	0001	0x1
Média Capacidade (LAG)	Laranja	0010	0x2
Média Capacidade –Curta Distância (LAG)	Lilás	0100	0x4
Média Capacidade –Longa Distância (LAG)	Verde	1000	0x8

# Configurações Necessárias

1. Roteamento configurado – OSPF
2. Infra MPLS - LDP
3. Habilitar o MPLS-TE no roteamento OSPF
4. Associar as interfaces ao RSVP
5. Definir os critérios de Affinity para as interfaces – atributos da interface (cores/características)
6. Definir as opções de caminhos para Engenharia de tráfego – o atributo do caminho
7. Definir a interface túnel e possíveis opções de caminhos
8. Associar o túnel ao L2VPN

# MPLS-TE, RSVP e Affinity: Configurações

## Habilitar o MPLS-TE

```
DmOS (config) #router ospf <1-65535>
```

```
DmOS (config-ospf-21-vrf-global) #mpls-te router-id loopback-<0-7>
```

*'router ospf 21 global': Enabling mpls-te will cause OSPF process to be restarted Proceed? [yes,no]*

## Associar as interfaces L3 ao RSVP

```
DmOS (config) #mpls rsvp
```

```
DmOS (config-rsvp) #interface l3-<name>
```

## Definir os critérios Affinity para as interfaces L3 (atributos)

```
DmOS (config) #mpls traffic-eng
```

```
DmOS (config-traffic-eng) #interface l3-<name>
```

```
DmOS (config-interface-l3-Infra-OSPF-MPLS-SW21-SW22) #affinity-flags <0x0-0xffffffff>
```

## Verificar as configurações

```
DmOS#show running-config router ospf mpls-te
```

```
DmOS#show running-config mpls rsvp
```

```
DmOS#show running-config mpls traffic-eng | tab
```

# MPLS-TE e Túnel TE: Configurações

## Definição das opções de caminhos para Engenharia de Tráfego

```
DmOS (config) #mpls traffic-eng
DmOS (config-traffic-eng) #attribute-set
DmOS (config-attribute-set) #path-option <name_path>
DMOS (config-path-option-caminho-alta-capacidade) #affinity-flags <exclude-any | include-all | include-any> <0x0-0xffffffff>
```

## Criação do túnel com engenharia de tráfego

```
DmOS (config) #interface tunnel-te <1-65535>
DmOS (config-tunnel-te-1) #name <name_tunnel>
DmOS (config-tunnel-te-1) #description <text>
DmOS (config-tunnel-te-1) #destination <ip_loopback_destination>
DMOS (config-tunnel-te-1) #path-option <1-255> dynamic attribute-set <name_path_option>
```

**\*\*\* A criação do túnel é unidirecional, portanto é necessário realizar a mesma configuração no outro vizinho da VPN \*\*\***

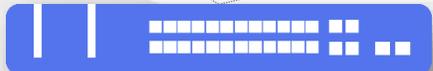
## Para verificar:

```
DmOS#show mpls forwarding-table
DmOS#show running-config mpls traffic-eng | tab
DmOS#show mpls traffic-eng tunnel-te <brief | id | name>
DmOS#show running-config interface tunnel-te | tab
```

# Affinity: Verificações

```
DmOS-1# show mpls forwarding-table
Prefix or Tunnel-Name  Action  In  In  Out  Out  Out
Label Proto Label Proto interface  Status
-----
AFFINITY_SW21_SW23_PPPE  psh    --  --  343  rsvp  13-vlan 2135  active
AFFINITY_SW21_SW23_VOIP  psh    --  --  342  rsvp  13-vlan 2135  active
```

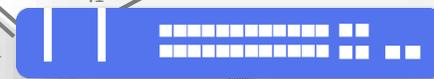
**DmOS-1**  
Lo 21.21.21.21/32



VLAN 2135  
10.21.35.x/31

LAG2

.1



**DmOS-5**  
Lo 35.35.35.35/32

VLAN 2335  
10.23.35.x/31

**DmOS-3**  
Lo 23.23.23.23/32



```
DmOS-3# show mpls forwarding-table
Prefix or Tunnel-Name  Action  In  In  Out  Out  Out
Label Proto Label Proto interface  Status
-----
AFFINITY_SW23_SW21_PPPE  psh    --  --  345  rsvp  13-vlan 2335  active
AFFINITY_SW23_SW21_VOIP  psh    --  --  344  rsvp  13-vlan 2335  active
```

```
DmOS-5# show mpls forwarding-table
Prefix or Tunnel-Name  Action  In  In  Out  Out  Out
Label Proto Label Proto interface  Status
-----
AFFINITY_SW21_SW23_PPPE  php    343  rsvp  ImpNull  rsvp  13-vlan 2335  active
AFFINITY_SW21_SW23_VOIP  php    342  rsvp  ImpNull  rsvp  13-vlan 2335  active
AFFINITY_SW23_SW21_PPPE  php    345  rsvp  ImpNull  rsvp  13-vlan 2135  active
AFFINITY_SW23_SW21_VOIP  php    344  rsvp  ImpNull  rsvp  13-vlan 2135  active
```

\*Apenas as informações referentes ao protocolo RSVP, o protocolo LDP foi omitido da visualização

# MPLS Serviços: Monitor com MPLS-TE

```

⊕ Frame 908: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface 0
⊕ Ethernet II, Src: TeracomT_5e:29:eb (00:04:df:5e:29:eb), Dst: TeracomT_5e:26:99 (00:04:df:5e:26:99)
⊖ MultiProtocol Label Switching Header, Label: 18, Exp: 0, S: 0, TTL: 255
  0000 0000 0000 0001 0010 .... = MPLS Label: 18
  .... = MPLS Experimental Bits: 0
  .... = MPLS Bottom of Label Stack: 0
  .... 1111 1111 = MPLS TTL: 255
⊖ MultiProtocol Label Switching Header, Label: 17, Exp: 0, S: 0, TTL: 255
  0000 0000 0000 0001 0001 .... = MPLS Label: 17
  .... = MPLS Experimental Bits: 0
  .... = MPLS Bottom of Label Stack: 0
  .... 1111 1111 = MPLS TTL: 255
⊖ MultiProtocol Label Switching Header, Label: 52165, Exp: 0, S: 1, TTL: 0
  0000 1100 1011 1100 0101 .... = MPLS Label: 52165
  .... = MPLS Experimental Bits: 0
  .... = MPLS Bottom of Label Stack: 1
  .... 0000 0000 = MPLS TTL: 0
⊕ Ethernet II, Src: Giga-Byt_f7:94:77 (50:e5:49:f7:94:77), Dst: TeracomT_b0:4f:de (00:04:df:b0:4f:de)
⊕ PPP-over-Ethernet Session
⊕ Point-to-Point Protocol
⊕ PPP Link Control Protocol
  
```

```

SW21# show mpls forwarding-table
Prefix or      In      In      Out      Out      Out
Tunnel-Name    Action  Label  Proto   Label    Proto   interface  Status
-----
22.22.22.22/32 fwd     --     --     ImpNull  ldp     13-vlan 2122  active
22.22.22.22/32 php     18     ldp     ImpNull  ldp     13-vlan 2122  active
23.23.23.23/32 psh     --     --     21       ldp     13-vlan 2122  active
23.23.23.23/32 swp     17     ldp     21       ldp     13-vlan 2122  active
24.24.24.24/32 fwd     --     --     ImpNull  ldp     13-vlan 2124  active
24.24.24.24/32 php     16     ldp     ImpNull  ldp     13-vlan 2124  active
SW21-SW23-AC  psh     --     --     19       rsvp    13-vlan 2124  active
SW21-SW23-BC  psh     --     --     18       rsvp    13-vlan 2122  active
  
```

```

VPLS-Group: OLT; VPN-Name: PPPoE; Admin status: up;
Bridge-Domain: Admin status: up; Oper state: up;
MAC learning: enabled
MAC aging time: 600, Type: inactivity
MAC limit: 1024, Action: drop
Transparent-LAN-Service: disabled
Bridge-MTU: auto
Dot1q: 2612
AC:  gigabit-ethernet-1/1/4.2612; Admin status: up; Oper state: up;
    MTU: 9198;
    Encapsulation Dot1q: none;
VFI: Admin status: up; Oper state: up;
    Pw-type: ethernet;
    Signalling protocol: ldp;
PW:  Neighbor address: 23.23.23.23; Admin status: up; Oper state: up;
    Up time: 0 day(s) 18 hour(s) 44 minute(s) 26 second(s);
    Last state change time: Wed Feb 17 20:28:28 2021;
    Pw-ID: 2123; Pw-MTU: auto; Tunnel interface: tunnel-te-2;
    FAT: Flow-label receive: true; Flow-label transmit: true;
    Split-horizon: enabled;
    Remote access interface state: up;
    MPLS VC labels: Local: 20; Remote: 17;
    MTU: Local: 9198; Remote: 9198;
  
```

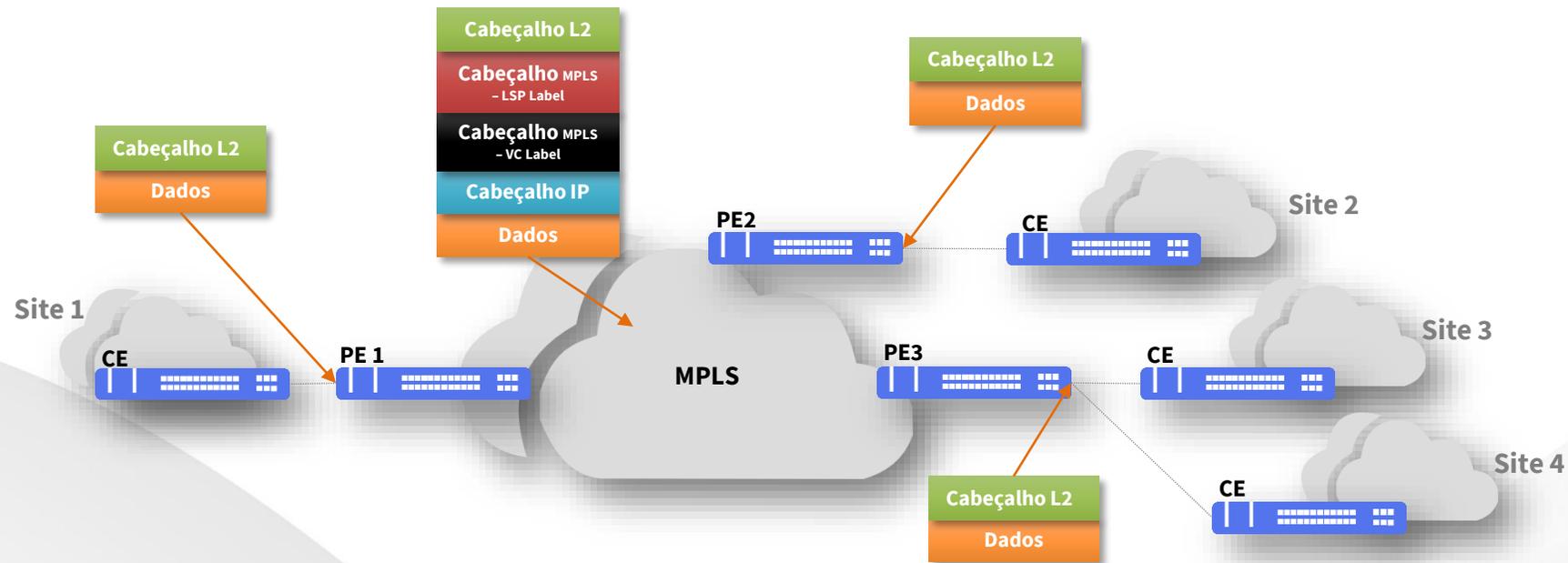
# Vantagem do MPLS-TE

1. **Distribuição do tráfego entre os enlaces disponíveis**, garantindo que a carga possa ser dividida através dos diferentes túneis;
2. **Associação a diferentes caminhos à um túnel**, permitindo caso o caminho primário falhe, outros possam ser utilizados, provendo assim **redundância**;
3. **Sem alterar a rede atual**, permite manipular os caminhos para o transporte dos fluxos desejadas;
4. A sua aplicação **não reduz os problemas de falta de recursos na rede**, mas **melhora o mapeamento dos seus recursos**.
5. **Os túneis são definidos apenas nas pontas** (*Head End*), não há necessidade de configuração constante da rede, facilitando a administração.



# MPLS – L2VPN

# MPLS Serviços L2VPN

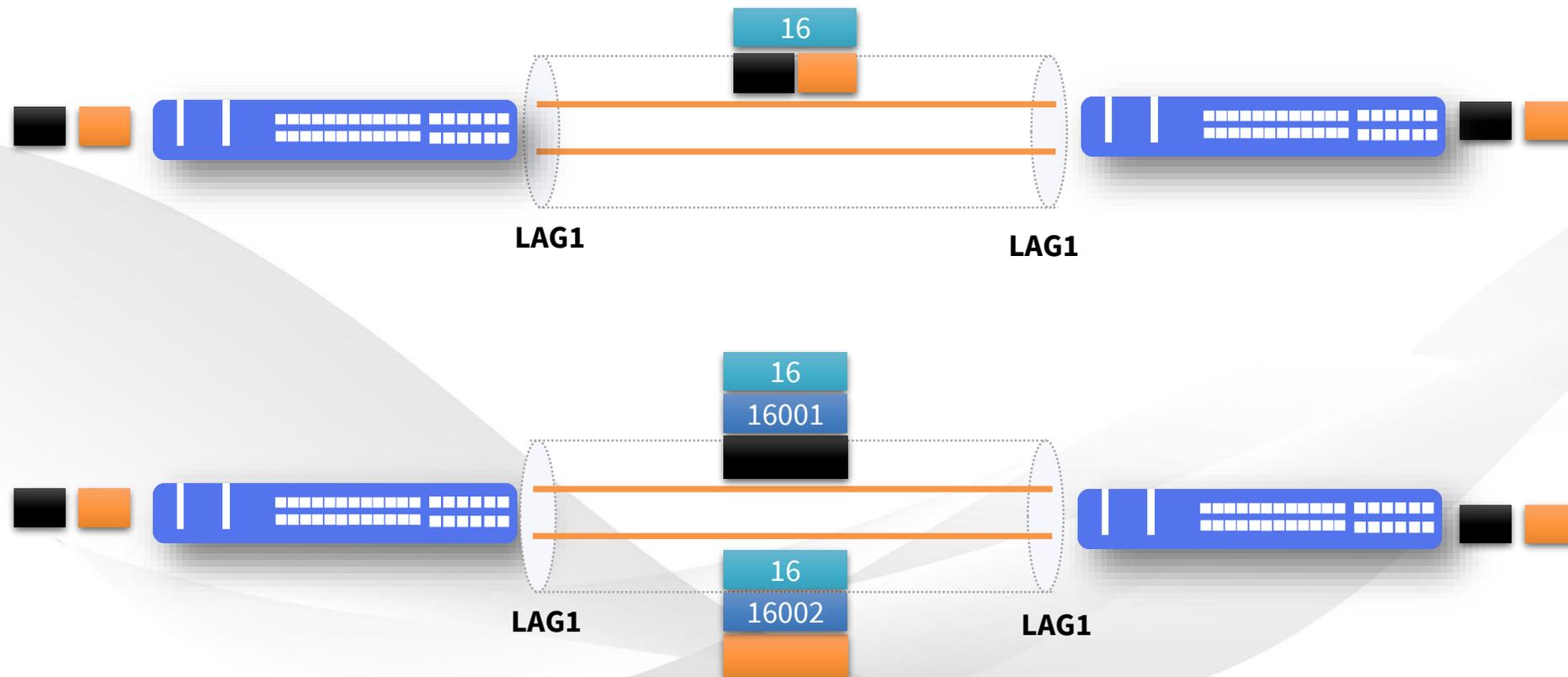


**Transporte transparente** entre ponto-a-ponto e multiponto, inclusive dos **protocolos** de suporte a camada 2, garantindo **isolamento** total do tráfego do cliente e do provedor.

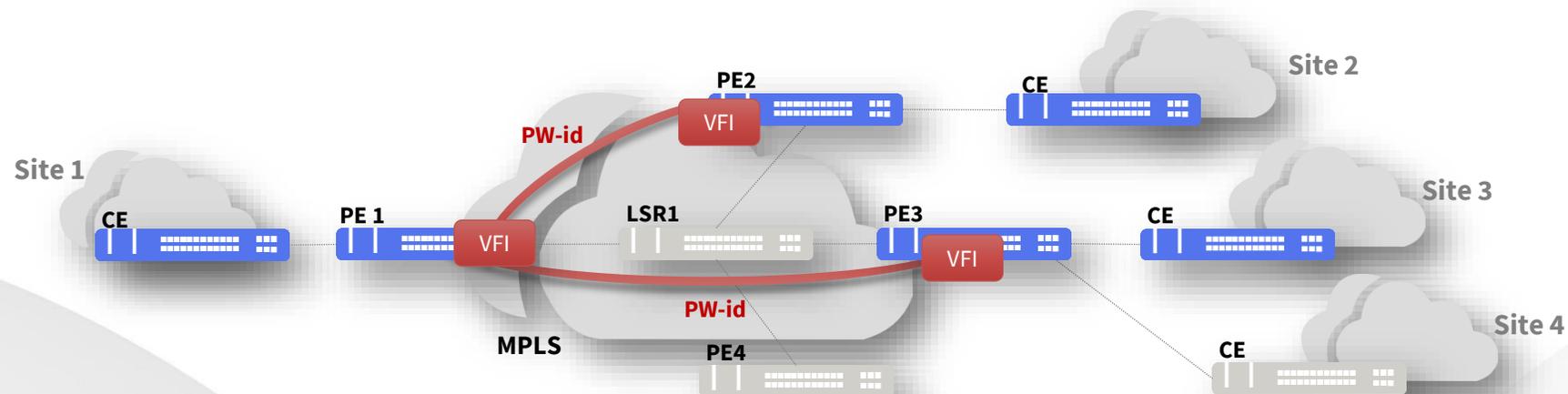
- Escalabilidade das redes de camada 3 e a simplicidade de encaminhamentos de quadros da camada 2;
- Baseia-se no empilhamento de labels MPLS – técnica draft Martini;
- O label externo é o LSP entre dois PEs e o label interno é o identificador do circuito - VPN;
- Modo **VPWS**: Ponto-a-Ponto ou **VPLS**: Ponto-Multiponto.

# Balanceamento - FAT

**FAT- Flow Aware Transport:** Insere um flow-label para cada fluxo recebido em uma porta de acesso da VPN, aumentando a variabilidade do tráfego do ponto de vista do MPLS.



# Virtual Private LAN Service (VPLS)



O tráfego do CE que ingressa no PE é **encapsulado em uma PWE3 e transportado até o PE de destino**, onde é desencapsulado e entregue ao CE remoto. Cada PE implementa uma **Virtual Forwarding Instance – VFI**;

- O critério de decisão do encapsulamento pode ser por **Ethernet** (apenas um cliente por interface) ou **VLAN** (mais de um cliente por interface);
- O **serviço MP2MP** emula um switch WAN entre os CEs;
- **Há aprendizado de MACs** nos PEs;
- O Split Horizon impede o trânsito de pacotes entre as PWs. Utilizado para prevenir loops em VPLS com full-mesh de PWs.

# VPLS: Configuração com MPLS-TE

## Criar o VPLS

```
DmOS (config) #mpls l2vpn  
DmOS (config-l2vpn) #vpls-group <vpls_group_name>
```

## Criar a VPN

```
DmOS (config-vpls-group-VPLS1) #vpn <vpn_name>  
DmOS (config-vpn-1) #description <text>
```

## Adicionar os PEs de destino

```
DmOS (config-vpn-1) #vfi  
DmOS (config-vpn-1-vfi) #pw-type <ethernet | vlan> <vlan-delimiting>  
DmOS (config-vpn-1-vfi) #neighbor <IP_Loopback_destination> pw-id <1-4294967294>  
DmOS (config-neighbor-6.6.6.6) #pw-mtu <64-9198>  
DmOS (config-neighbor-6.6.6.6) #pw-load-balance flow-label <both | receive | transmit>  
DmOS (config-neighbor-6.6.6.6) #tunnel-interface tunnel-te-<1-65535>
```

## Interface do cliente a ser associada ao VPLS

```
DmOS (config-vpn-1) #bridge-domain access-interface <gigabit-ethernet | ten-gigabit-ethernet | forty-gigabit-ethernet | hundred-gigabit-ethernet>--<chassis/slot/port>
```

\*\*\* Se o serviço for por VLAN (mais de um cliente por interface ethernet), adicione a sintaxe a seguir. Não é necessário criar esta VLAN no switch, apenas no cliente - CE \*\*\*

```
DmOS (config-vpn-1) #bridge-domain dot1q <1-4094>
```

# VPLS: Configuração com MPLS-TE

## **TLS - Transparent Lan Service: Transporte de PDUs dos protocolos L2 em uma VPLS**

```
DmOS (config-vpn-1) #bridge-domain transparent-lan-service
```

## **Configuração do limite de MACs a serem aprendidos em uma VPLS**

```
DmOS (config-vpn-1) #bridge-domain mac-limit <1-32767>
```

DATACOM 

<https://ead.datacom.com.br>

# DATACOM



Rua América, 1000 | Eldorado do Sul | RS | Brasil

+55 51 3933 3000

[www.datacom.com.br](http://www.datacom.com.br)