



**SEMANA DE  
CAPACITAÇÃO 4**  
**EDIÇÃO ON-LINE**

**31 DE MARÇO, ÀS 9H (UTC-3)**

REALIZAÇÃO

**ceptro.br nic.br cgi.br**

# **DESCOMPLICANDO SEGMENT ROUTING - CONFIGURANDO SEGMENT ROUTING EM AMBIENTE EMULADO COM CISCO**



**GUSTAVO KALAU**  
Gustavo Kalau Treinamento

# Segment Routing



## Rede Tradicional

OSPF/ISIS (IGPs)

Diversas limitações na entrega de serviços,  
praticamente sem soluções de overlay



## Rede MPLS

MPLS + OSPF/ISIS + LDP + MPBGP

Vários serviços podem ser configurados,  
Underlay x Overlay

- Complexidade de configuração;
- Automação difícil;



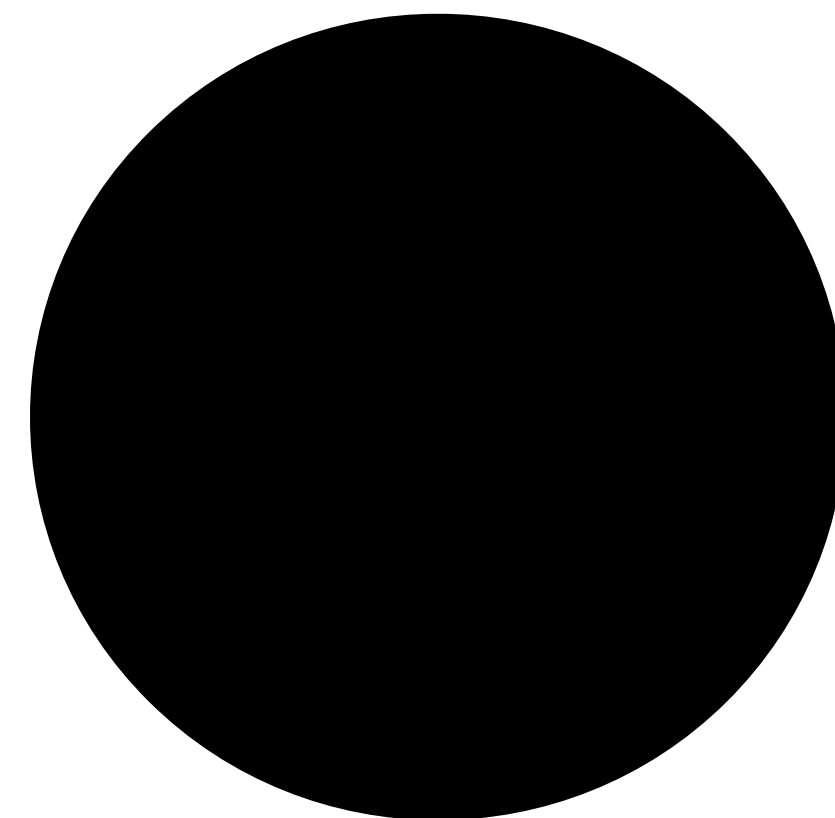
## Rede Segment Routing

Os mesmos serviços entregues pelo MPLS,

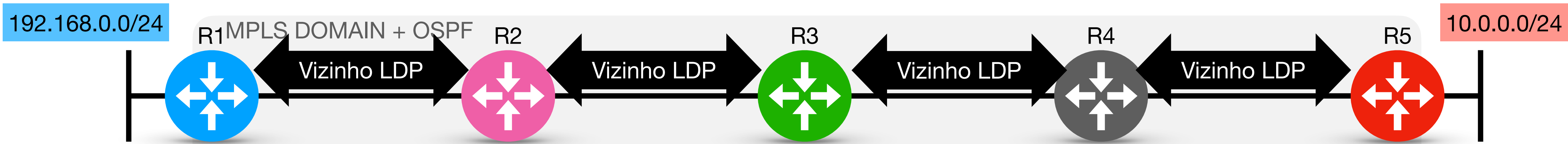
Configuração menos complexa,

Já pensada para automação,

- Momento 1: Coexistência com MPLS;
- Momento 2: SRv6 e fim do MPLS;



# Label Switching Review



LABEL	NETWORK	VIA
19	10.0.0.0/24	Local
87	10.0.0.0/24	R2

LABEL	NETWORK	VIA
87	10.0.0.0/24	Local
19	10.0.0.0/24	R1
11	10.0.0.0/24	R3

LABEL	NETWORK	VIA
11	10.0.0.0/24	Local
87	10.0.0.0/24	R2
65	10.0.0.0/24	R4

LABEL	NETWORK	VIA
65	10.0.0.0/24	Local
11	10.0.0.0/24	R3
23	10.0.0.0/24	R5

LABEL	NETWORK	VIA
23	10.0.0.0/24	Local
65	10.0.0.0/24	R4

LABEL IN	LABEL OUT	VIA
19	87	R2

LABEL IN	LABEL OUT	VIA
87	11	R3

LABEL IN	LABEL OUT	VIA
11	65	R4

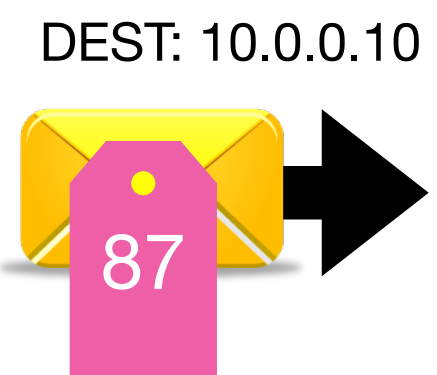
LABEL IN	LABEL OUT	VIA
65	23	R5

FIB		
DST IP - IN	LABEL - OUT	VIA
10.0.0.0/24	87	R2



LABEL IN	LABEL OUT	VIA
65	POP	R5

FIB		
DST IP - IN	LABEL - OUT	VIA
10.0.0.0/24	Não tem!	Connected

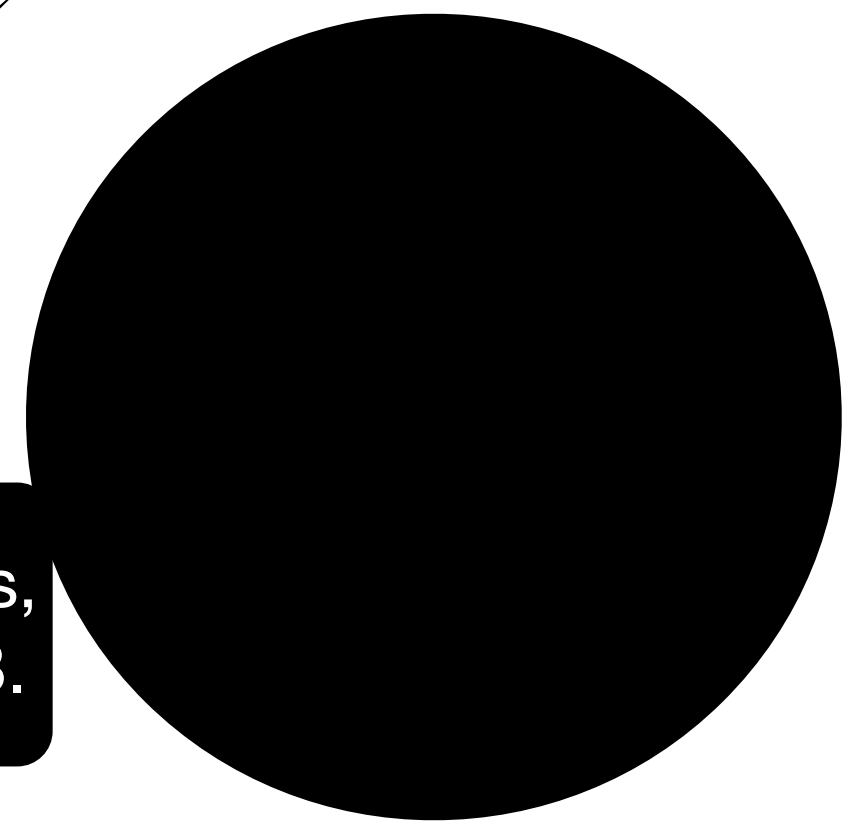


3 - Portanto, R5 recebe um pacote sem Label e pode fazer uma única pesquisa usando o FIB (não precisa passar pela LFIB) para encaminhar o pacote.

2 - Essencialmente, R5 diz a R4 que é o fim do LSP para a rede 10.0.0.0/24 e que R4 deve remover qualquer Label e encaminhar o pacote para R5.

R5 to R4 Pop = 10.0.0.0/24

1 - Repare que o R5 teve dois trabalhos, retirar o LABEL e depois encaminhar via FIB.



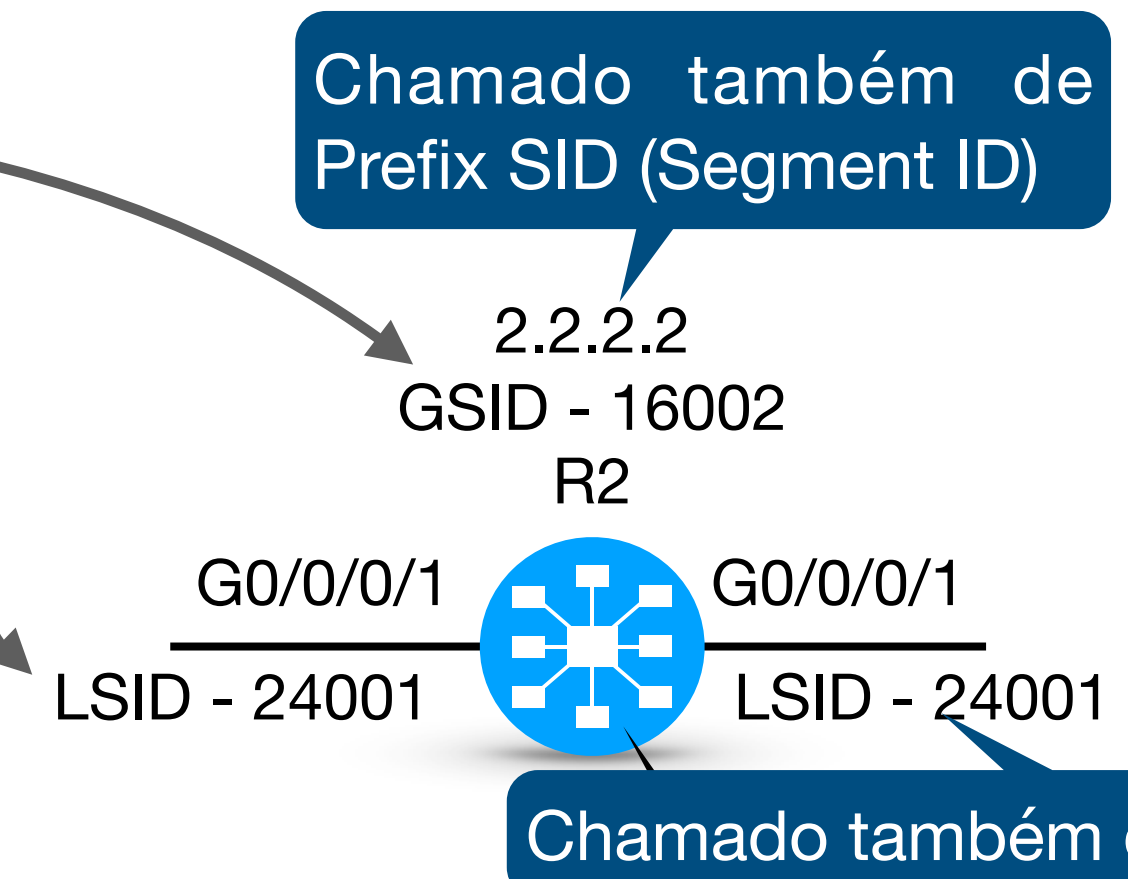
# Segment Routing

Internet Engineering Task Force (IETF)  
 Request for Comments: 8402  
 Category: Standards Track  
 ISSN: 2070-1721

O Segment Routing (SR) aproveita o paradigma de roteamento de origem. Um nó dirige um pacote através de uma lista ordenada de instruções, chamadas "segments". Um segmento pode representar qualquer instrução, topológica ou baseada em serviço. Um segmento pode ter uma semântica local para um node SR ou global um domínio SR. O SR fornece um mecanismo que permite que um fluxo seja restrito a um caminho topológico específico, mantendo o estado por fluxo apenas no(s) node(s) de entrada para o domínio SR.

O SR pode ser aplicado diretamente à arquitetura MPLS sem alterar o plano de encaminhamento. Um segmento é codificado como label MPLS. Uma lista ordenada de segmentos é codificada como uma pilha de labels. O segmento a ser processado está no topo da pilha. Após a conclusão de um segmento, o label relacionado é retirado da pilha.

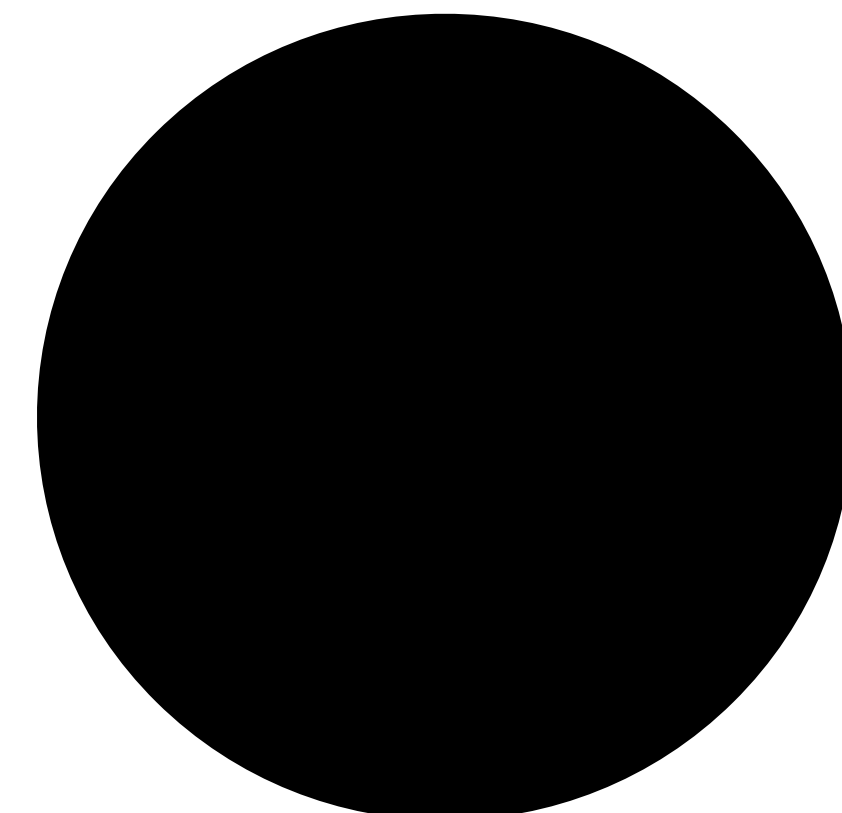
O SR pode ser aplicado à arquitetura IPv6, com um novo tipo de cabeçalho de roteamento. Um segmento é codificado como um endereço IPv6. Uma lista ordenada de segmentos é codificada como uma lista ordenada de endereços IPv6 no cabeçalho de roteamento. O segmento ativo é indicado pelo endereço de destino (DA) do pacote. O próximo segmento ativo é indicado por um ponteiro no novo cabeçalho de roteamento.



Os segmentos serão utilizados pelo MPLS e distribuídos via IGP (OSPF ou ISIS) não necessitando assim de LDP.



Labels Range	
0	Special Purpose And Static
- 15.999	
16.000	Preserved
- 23.999	
24.000	Dynamic Label Range
1.048.575	



Segment Routing Global Block = SRGB
Range de labels reservado para SRGB (default) = 16.000 - 23.999 <i>Obs: não é recomendado alterar</i>
GSID deve ser único
Index based: Prefixo 1.1.1.1/32 com index de 1 terá o label 16001

# The OSPF Opaque LSA Option

Network Working Group  
Request for Comments: 5250  
Obsoletes: [2370](#)  
Category: Standards Track

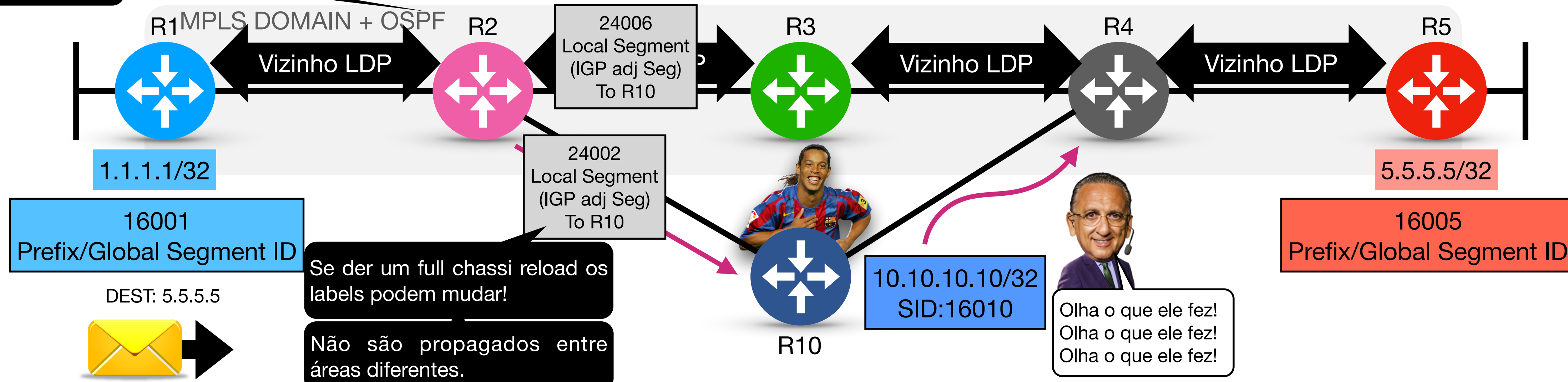
# Segment Routing

LSAs opacos fornecem um mecanismo generalizado para permitir a futura extensibilidade do OSPF. LSAs opacos consistem em um cabeçalho LSA padrão seguido por informações específicas da aplicação.

Mecanismos de OSPF link-state database flooding são usados para distribuir LSAs opacos para toda ou parte limitada da topologia OSPF.

- Router Information Opaque LSA (Type 4): TLV 8: SR-Algorithm TLV 9: SID/Label Range
- New Opaque LSAs para propagar os SIDs:
  1. OSPFv2 Extended Prefix Opaque LSA (Type 7)
  2. OSPFv2 Extended Link Opaque LSA (Type 8)

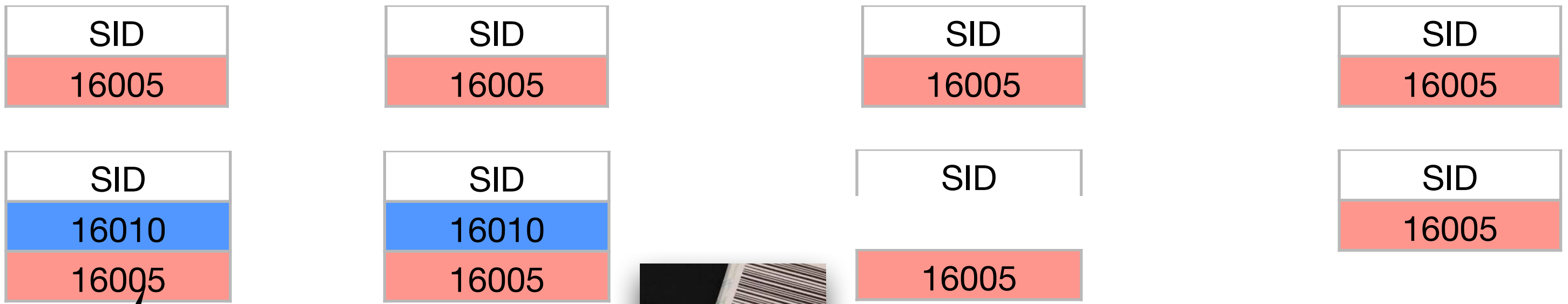
O próprio OSPF (ou ISIS) irá distribuir os LABELS agora!



Se der um full chassi reload os labels podem mudar!

Não são propagados entre áreas diferentes.

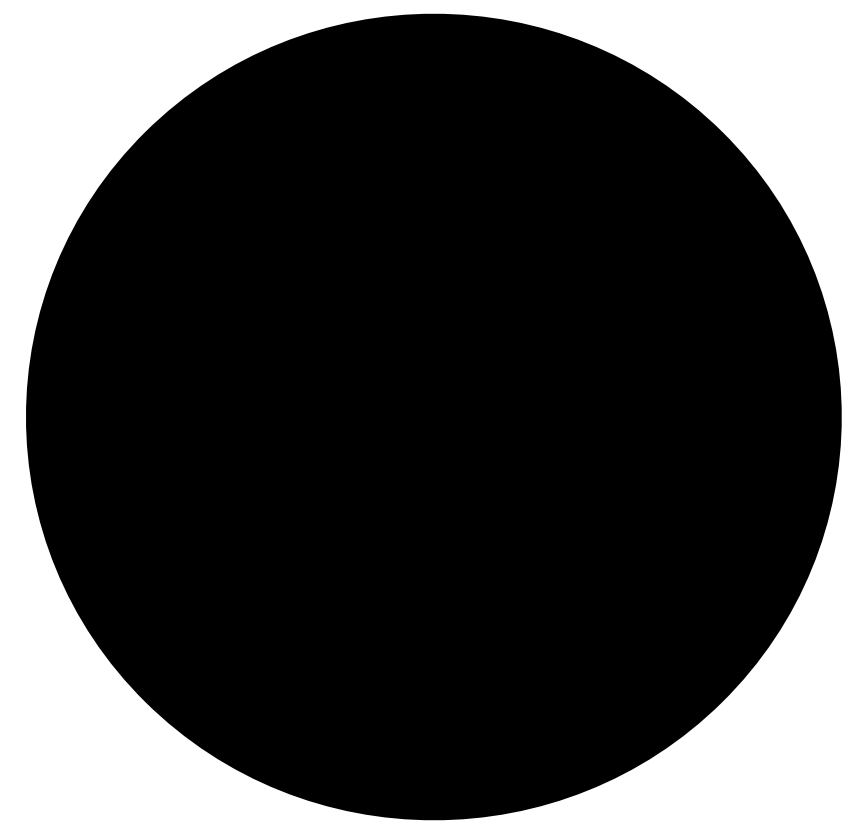
Olha o que ele fez!  
Olha o que ele fez!  
Olha o que ele fez!



Do ponto de origem do tráfego eu defini o caminho que queria que ele fizesse!

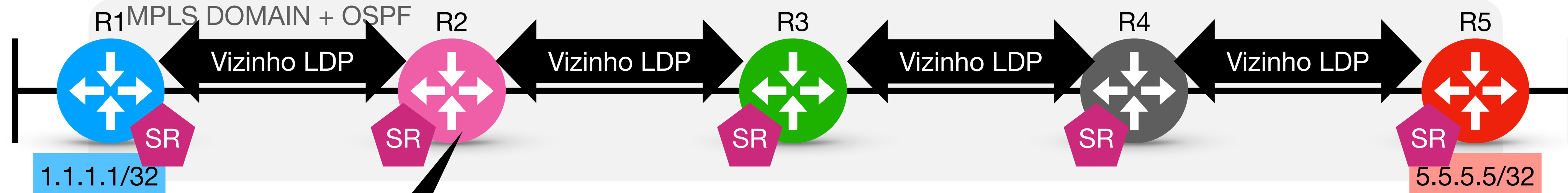


Shortest path para o prefixo via IGP e Equal-Cost Multipath (ECMP) - aware



# Segment Routing e LDP interworking data plane

Se tiver LDP e SR, LDP será o preferido, esse comportamento pode ser mudado com o comando `segment-routing mpls sr-prefer`.



Fase 1 - Todos os nodes rodam somente LDP

Fase 2 - Todos os nodes são atualizados para SR (em qualquer ordem), o LDP continua sendo o principal.

Fase 3 - Configura os PEs com o comando `segment-routing mpls sr-prefer`.

Fase 4 - Desabilita o LDP nas caixas.

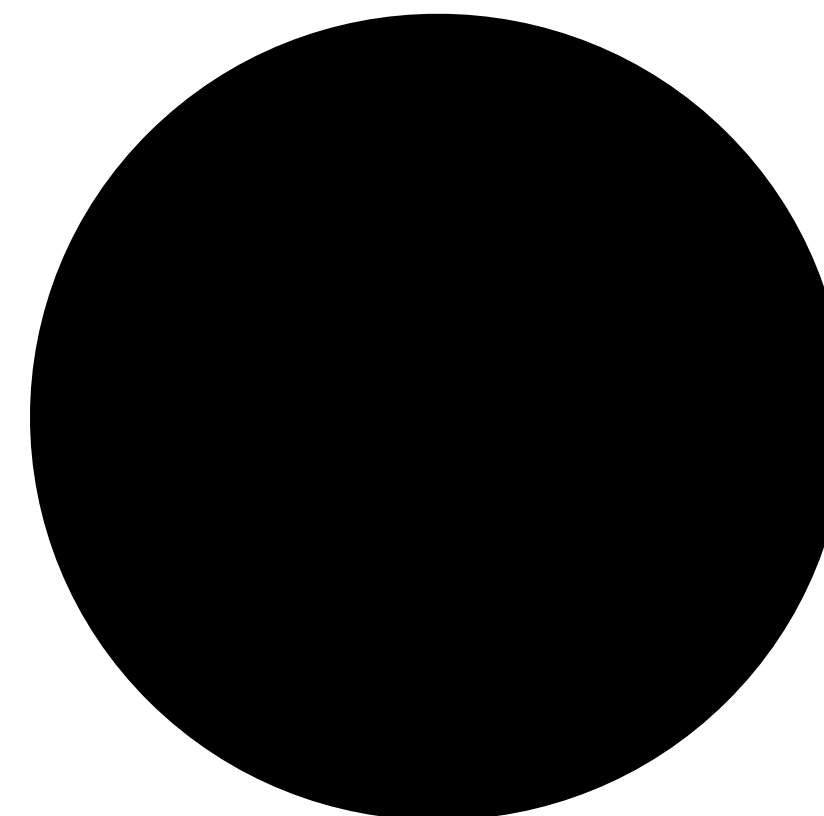
Obs.: Como o tráfego entra como SR ele continuará com SR nos próximos saltos pois temos uma rede com LDP + SR no core.

Se uma parte da rede ainda precisar falar LDP existe a possibilidade da rede SR criar labels para os prefixos da rede LDP pura, mas para isso é necessário a configuração de uma função chamada Mapping Server.

Mapping Server:

- É simplesmente um mecanismo de control plane;
- Não precisa estar no data path;
- O ideal é que seja redundante (parecido com um Router Reflector);
- Mapping server advertisements são propagados entre áreas OSPF mas não entre ISIS levels (ms por level)

Ao criar o mapeamento manual no Mapping Server, esses SIDs são propagados e recebidos automaticamente via OSPF (LSA type 7 opaque) pelos nodes com SR habilitado (clients), isso simplifica demais a configuração.



# Topology Independent Loop-Free Alternate (TI-LFA)

Cobertura de 100% e 50ms link, node e SRLG protection.

- Previne micro loop e suboptimal routing;
- Utiliza post-convergente path previamente calculado;
- Automaticamente calculado pelo IGP;
- Também protege o tráfego LDP e IP (em ambiente com deploy incremental do SR).

Calcula post-convergence SPT

- SPT da interface primária para o destino (D) é removida da topologia;
- Calcula nodes P e Q;
- Escolhe os nodes P e Q no path post-convergence;
- Constrói o backup path segment list para o TI-LFA backup (post-convergence path);
- Faz isso para cada destino;

1 - E se esse caminho cair? Será que existe um caminho alternativo? E se existir, qual a melhor forma de usar os SID para que não ocorra uLoops?

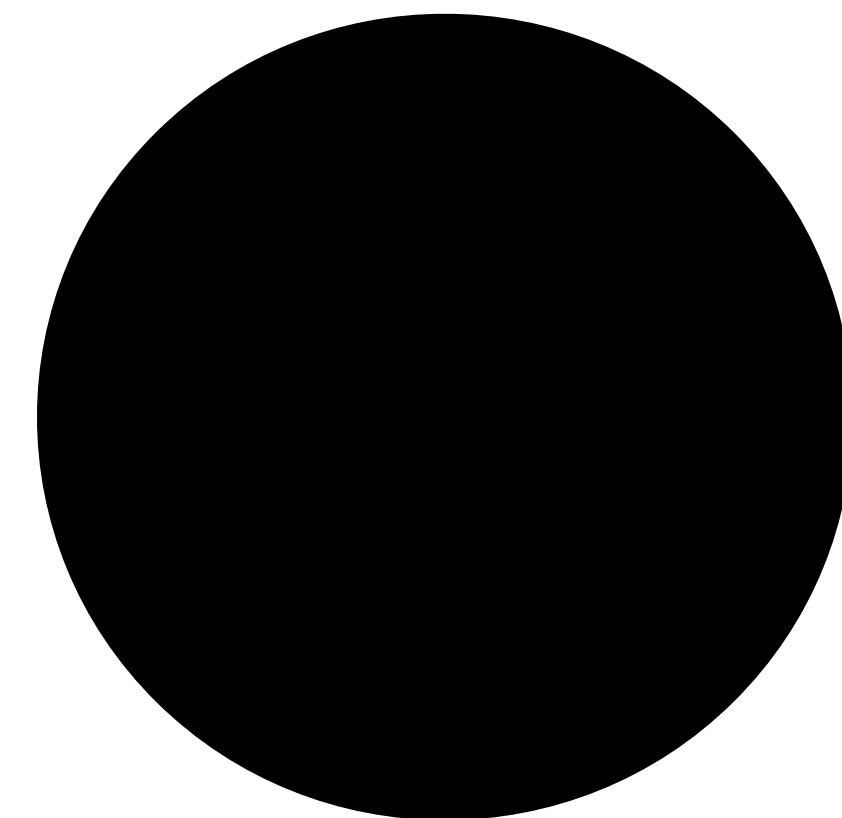
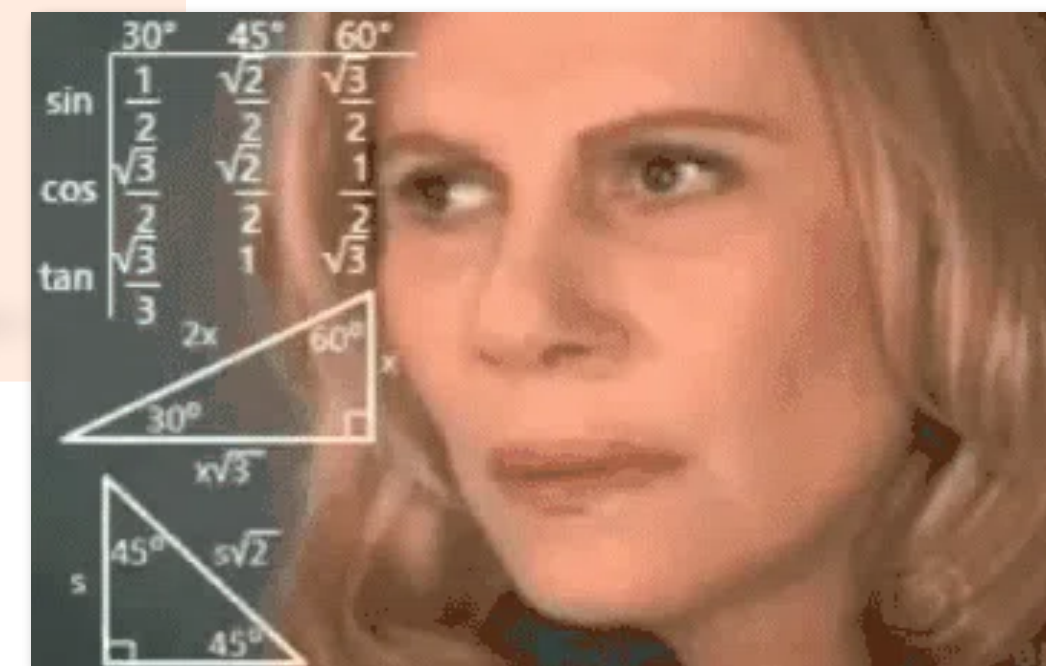
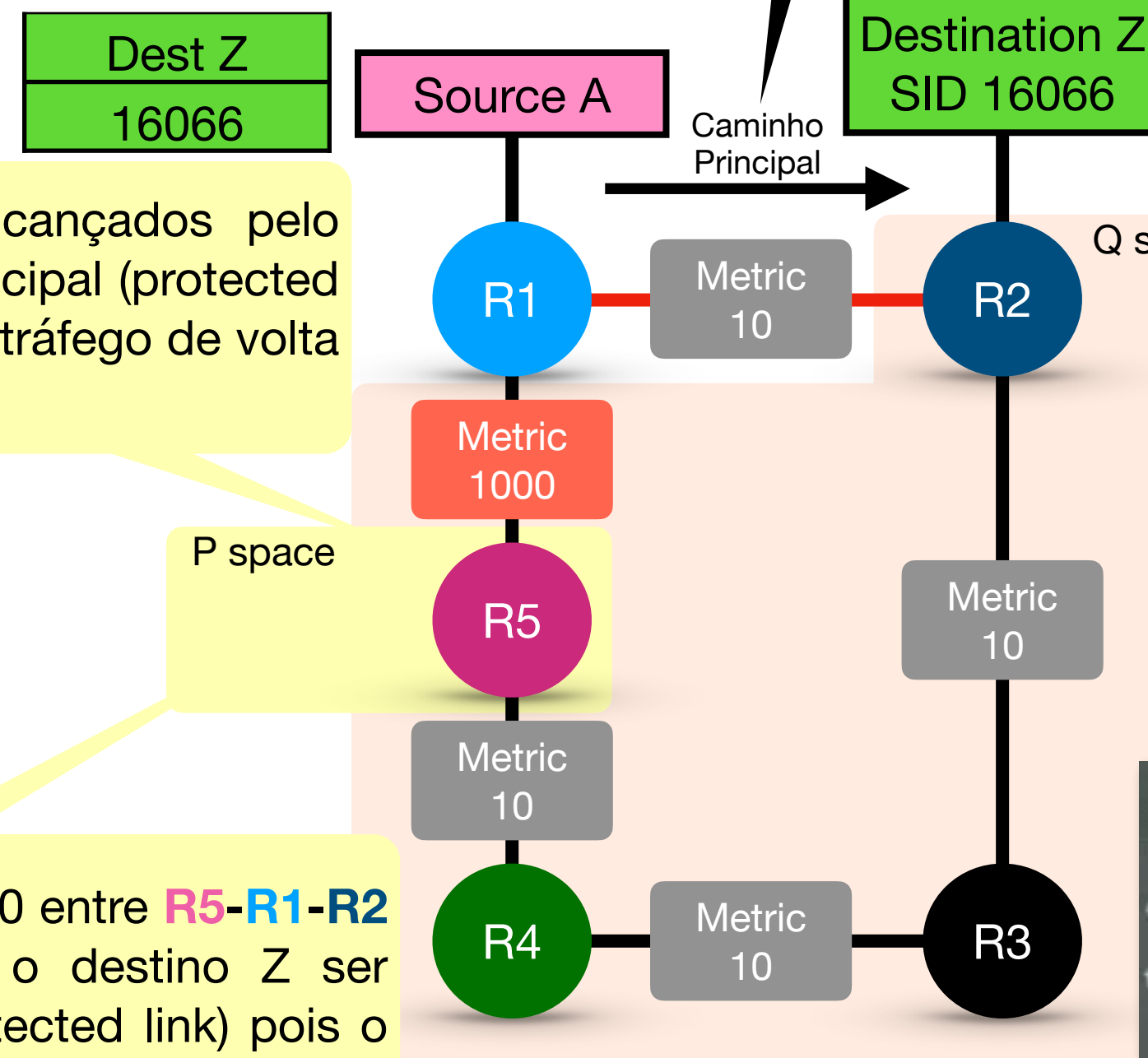
6 - Nesse cenário será enviado o tráfego utilizando o SID do destino sem necessidade de nenhum SID adicional, pois nenhum uLoop irá ocorrer (SR zero-segment).

2 - P space: nodes que são alcançados pelo Source (S) sem usar o caminho principal (protected link) e sem risco do node mandar o tráfego de volta via Link Primário (protected link);

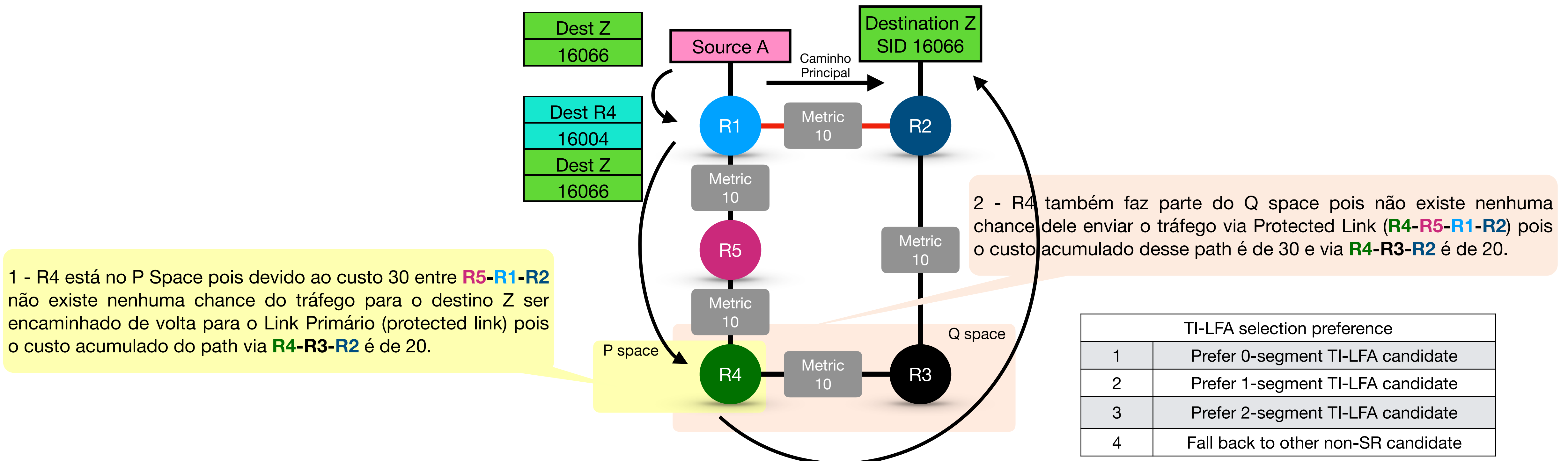
4 - Q space: nodes que podem ser alcançados pelo Destination (D) sem usar o caminho principal (protected link) e sem risco do node mandar o tráfego de volta.

5 - R5 também faz parte do Q space pois não existe nenhuma chance dele enviar o tráfego via Protected Link (R5-R1-R2) pois o custo acumulado desse path é de 1010

3 - R5 está no P Space pois devido ao custo 1010 entre R5-R1-R2 não existe nenhuma chance do tráfego para o destino Z ser encaminhado de volta para o Link Primário (protected link) pois o custo acumulado do path via R5-R4-R3-R2 é de 30.



# Topology Independent Loop-Free Alternate (TI-LFA)

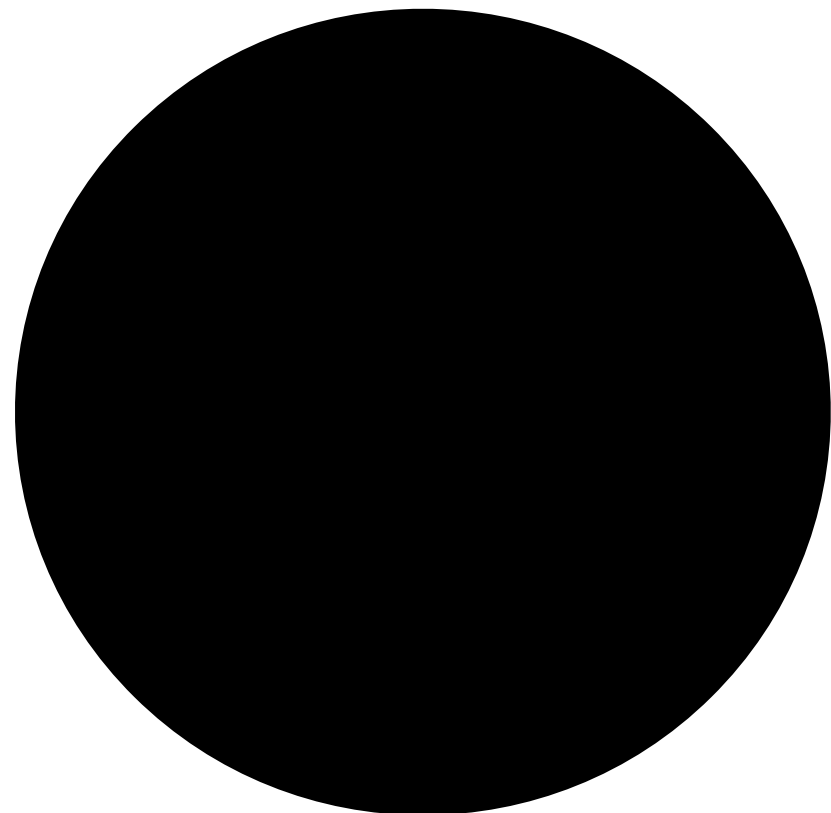


1 - R4 está no P Space pois devido ao custo 30 entre R5-R1-R2 não existe nenhuma chance do tráfego para o destino Z ser encaminhado de volta para o Link Primário (protected link) pois o custo acumulado do path via R4-R3-R2 é de 20.

2 - R4 também faz parte do Q space pois não existe nenhuma chance dele enviar o tráfego via Protected Link (R4-R5-R1-R2) pois o custo acumulado desse path é de 30 e via R4-R3-R2 é de 20.

3 - Como o SID de destino no topo da pilha é do R4 e não do destino Z é impossível que um micro loop (uLoop) ocorra, pois do ponto de vista do R4 ele nunca irá utilizar o caminho via R5-R1-R2 (custo 30) pois via R4-R3-R2 é 20. (SR single-segment)

TI-LFA selection preference	
1	Prefer 0-segment TI-LFA candidate
2	Prefer 1-segment TI-LFA candidate
3	Prefer 2-segment TI-LFA candidate
4	Fall back to other non-SR candidate



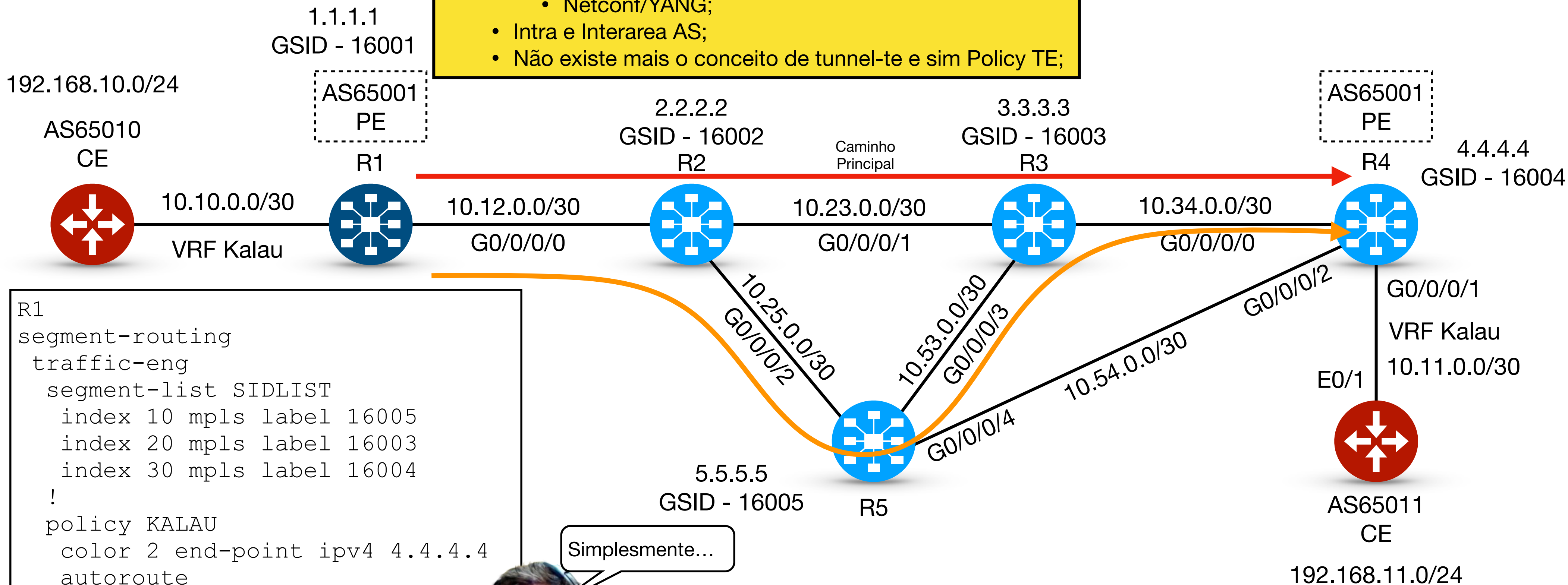


# Segment Routing Traffic Engineering (SR-TE)

SR-TE veio com a promessa de ser muito mais simples de implementar e manter do que RSVP-TE.

## Características:

- Unidirecional;
- Pode ser provisionado de várias formas:
  - Manualmente (CLI/XML);
  - Via PCE (Path Computation Element);
  - Netconf/YANG;
- Intra e Interarea AS;
- Não existe mais o conceito de tunnel-te e sim Policy TE;



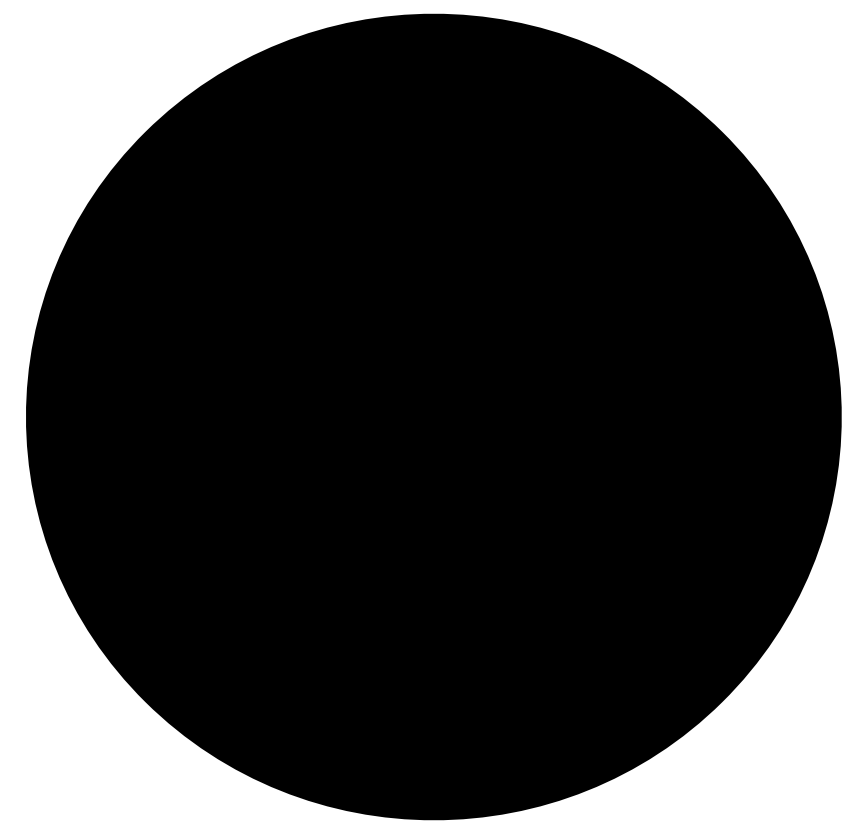
```

R1
segment-routing
traffic-eng
segment-list SIDLIST
index 10 mpls label 16005
index 20 mpls label 16003
index 30 mpls label 16004
!
policy KALAU
color 2 end-point ipv4 4.4.4.4
autoroute
include all
!
candidate-paths
preference 100
explicit segment-list SIDLIST
    
```



Simplesmente...

É criado um SID que identifica essa SR-TE policy, esse segmento é chamado de Binding-SID (BSID)



# Segment Routing Traffic Engineering (SR-TE)

```
R1
segment-routing
traffic-eng
  segment-list SIDLIST
  index 10 mpls label 16005
  index 20 mpls label 16003
  index 30 mpls label 16004
!
policy KALAU
color 2 end-point ipv4 4.4.4.4
autoroute
  include all
!
candidate-paths
preference 100
dynamic
  metric type te
  constraints
  affinity
  exclude-any color red
preference 200
explicit segment-list SIDLIST
```

```
RX
segment-routing
  affinity-map
  name blue bit-position 0
  name red bit-position 1
  name green bit-position 2
!
interface g0/0/0/0
  affinity color red
!
interface g0/0/0/0
  affinity color blue
  affinity color green
```

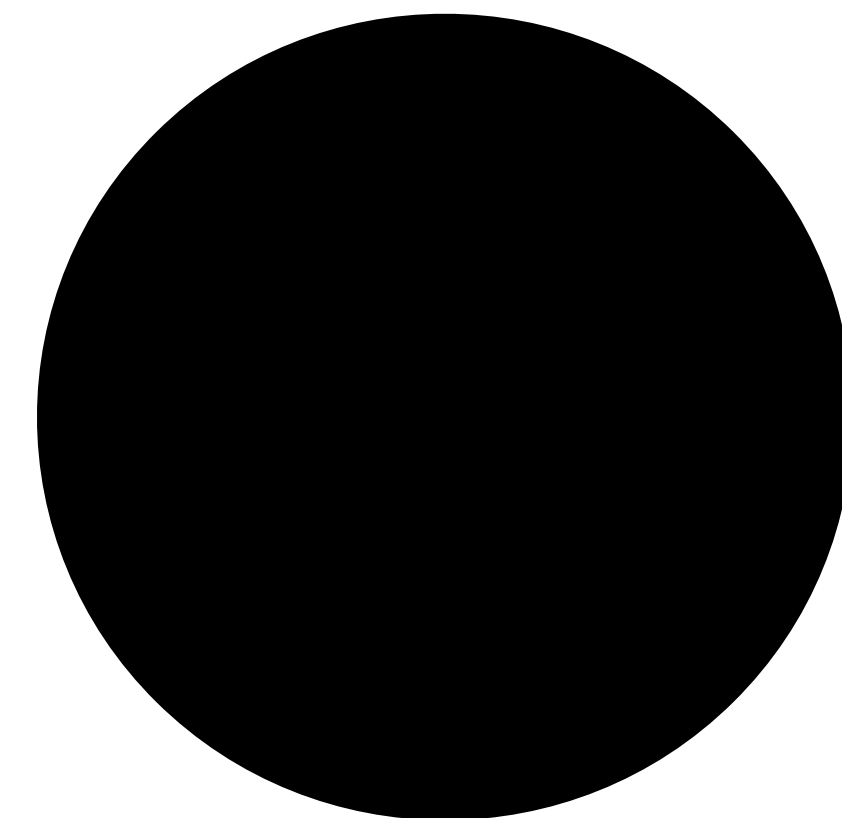
```
RX
prefix-set R3
  3.3.3.3/32
end-set
```

Constrain-SRLG(Shared risk link groups). Ex: Links com o mesmo SRLG-ID dividem um risco em comum, como por exemplo mesma fibra ou caixa.

```
srlg
interface g0/0/0/0
  10 value 1111
  20 value 2222
interface g0/0/0/1
  10 value 2222
  20 value 3333
```

```
RX
segment-routing
!
candidate-paths
preference 100
  constraints
  srlg
  exclude 1111
preference 200
explicit segment-list SIDLIST
```

```
RX
segment-routing
traffic-eng
  segment-list SIDLIST
  index 10 mpls label 16005
  index 20 mpls label 16003
  index 30 mpls label 16004
!
policy KALAU
color 2 end-point ipv4 4.4.4.4
autoroute
  include all
!
candidate-paths
preference 100
dynamic
  metric type te
  constraints
  affinity
  exclude-any R3
preference 200
explicit segment-list SIDLIST
```





# Obrigado!

Conheça os nossos treinamentos acesse:

**[gustavokalau.com.br](http://gustavokalau.com.br)**