

# Identificar e Solucionar Problemas do MST nos Switches Catalyst 9000

## Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Terminology](#)

[Restrições](#)

[Troubleshoot](#)

[MST \(uma única região\)](#)

[Topologia](#)

[Configuração](#)

[Validação](#)

[Sincronização entre regiões](#)

[Topologia](#)

[Validação](#)

[Debugs](#)

[Falha de Simulação de PVST](#)

[PVST BPDUs vs MST BPDUs](#)

[Topologia](#)

[Validação](#)

[Debugs](#)

[Contestação P2P](#)

[Topologia](#)

[Explicação](#)

[Abordagens MST](#)

[Informações Relacionadas](#)

## Introduction

Este documento descreve os conceitos básicos necessários para entender como o MST funciona em uma topologia com PVST ou outras regiões.

## Prerequisites

## Requirements

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Rapid-PVST (Rapid Per VLAN Spanning Tree)

## Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- Catalyst 9300.
- A partir de 17.3 comboio em diante.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

## Produtos Relacionados

Este documento também pode ser usado com estes hardwares:

- Toda a família Catalyst 9000.

## Informações de Apoio

### Terminology

Antes de iniciar e aplicar qualquer tipo de solução de problemas, considere esta terminologia:

Conceito	Descrição
Instância de STP	Uma instância é uma sessão executada na CPU: No PVST, uma VLAN é uma instância no. No MST, uma instância é um grupo de VLANs. Este documento usaria o termo instância com base neste significado.
IST	<b>IST (Internal Spanning Tree)</b> também é conhecido como <b>Instance 0</b> ou MSTI0: - Este é um caso especial. - O MSTI 0 é usado para criar uma topologia livre de loop único em todo o domínio L2. Quando o MST se comunica com outras regiões ou switches que executam outras versões <b>spanning tree</b> , as configurações do IST ou do MSTI0 são usadas para a comunicação. - O MSTI 0 é o único BPDU e a raiz escolhida pelo switch do MSTI 0 é responsável por ser a raiz de todas as regiões ou por transportar informações sobre raízes regionais dentro de uma região do MST O IST é a única instância de <b>spanning tree</b> que envia e recebe BPDUs. Todas as outras informações de instância de <b>spanning tree</b> estão contidas em M-records, que são encapsuladas dentro de BPDUs de MSTP. Porque o MSTP BPDU transporta informações para todas as instâncias. Essa é a única instância que tem parâmetros relacionados ao temporizador. Quando o MST se comunica com outras regiões e versões do <b>spanning tree</b> , as configurações do IST ou do MSTI0 são as usadas para a comunicação.
MSTIs	<b>MSTIs</b> significa <b>Multiple Spanning Tree Instances</b> . De 1 a 15 A implementação da Cisco suporta 16 instâncias: uma IST (instância 0) e 15 MSTIs.
Região	Um grupo de switches que executa o MST. Todos eles têm a mesma configuração de MST.
CIST e CST	- A <b>árvore de abrangência comum</b> interconecta as regiões MST e árvores de abrangência únicas.

- Uma **Árvore de abrangência interna e comum** é uma coleção de ISTs em cada região de e **Árvore de abrangência comum**.

Esse é o processo de eleição para cada instância em uma região, com exceção da instância 0. É possível ter uma raiz diferente em uma região de spanning tree para cada instância, se necessário.

Isso é feito se forem consideradas as informações no IST BPDU que têm as informações necessárias para executar uma eleição de spanning tree normal.  
**Raiz regional** A bridge raiz CIST foi chamada de mestre IST na implementação pré-padrão. Se a bridge CIST estiver na região, a raiz regional será a bridge raiz CIST. Caso contrário, a raiz regional é o switch mais próximo da raiz CIST na região. A raiz regional atua como uma bridge raiz para o IST.

Como há apenas uma BPDU e essa BPDU reflete as informações necessárias para cada instância 0, é necessário outro mecanismo para formar raízes para outras instâncias.  
**M-Record** Isso é chamado de M-Record. Dentro de cada M-Record estão todas as informações da spanning tree para uma instância individual. Essas informações são transportadas com TLVs no IST BPDU.

O mecanismo **Dispute** é um mecanismo integrado de detecção de link unidirecional. Isso não está disponível na versão original do 802.1d (o RSTP foi integrado ao padrão 802.1d em 2004) ou PVST.

**Contestação** O mecanismo de **disputa** é acionado no recebimento de um BPDU inferior que tenha um endereço designado e esteja em um estado de aprendizagem e encaminhamento.

Isso indica um link unidirecional e, para evitar loops, a porta que recebe o BPDU inferior bloqueia o link. Este mecanismo de **acordo de proposta** é uma das alterações mais importantes para o RSTP. Isso é o que permite que o rapid spanning tree seja realmente rápido.

Uma explicação simplificada do processo de **acordo de proposta** é que quando 2 vizinhos aparecem, ambos começam com sua transmissão de BPDU com um bit de proposta.  
**Proposta/Contrato** Quando um dos pares faz a transição para o acordo (ele declara que o vizinho é aceito como o melhor caminho superior para a raiz), os links imediatamente fazem a transição para um estado de encaminhamento.

Começa com as duas portas que enviam BPDUs. Eles alegam ser a raiz com designação de bits de proposta.

Quando o switch inferior reconhece que essa porta não é uma bridge raiz e tem o melhor caminho para a raiz, ele não tem mais o bit de proposta definido e faz a transição para o estado de Raiz e o encaminhamento.

O RSTP/MST coloca um link half-duplex em um estado "compartilhado". Isso significa que o processo do **Acordo da Proposta** não acontece.

**Segmentos compartilhados** Como a sequência tem como objetivo ativar rapidamente os links P2P, uma transição prematura para um estado de encaminhamento poderia causar um loop. Isso pode ser visto nos comandos show para spanning tree

Você pode digitar spanning-tree link-type point-to-point na interface para forçá-la a estar no estado P2P. Use-a com cuidado.

**Várias Regiões**  
·Várias regiões são determinadas quando as configurações de MST não correspondem.  
·O CIST é eleito entre Regiões via MST10 BPDU  
·Várias regiões aparecem como um switch lógico por região para outros dispositivos.

**Porta de limite** Essas portas estão nos limites da Região, geralmente nessas portas são recebidas BPDUs de outras regiões. O MST, portanto, não é possível nessa porta.

**Simulação de PVST** A Simulação de PVST é a forma como o MST e o PVST podem trabalhar na mesma rede. Em determinados cenários, como migrações ou alterações na topologia de uma rede, mais de um tipo de STP é encontrado junto e uma região do MST é conectada a outro domínio. Por exemplo, uma rede que muda de PVST+ para MST e todos os switches não podem ser modificados ao mesmo tempo. Além disso, é necessário trabalhar com o MST e o PVST+

conjunto.

Como o PVST+ não pode processar BPUs MST, há um mecanismo de compatibilidade entre eles, de modo que ambos os protocolos possam interagir. Esse mecanismo de compatibilidade é chamado de simulação PVST.

Falha de

Simulação de PVST Se as regras declaradas na simulação PVST não forem atendidas

## Restrições

- PVST+, Rapid PVST+ e MSTP são suportados, mas apenas uma versão pode estar ativa a qualquer momento. (Por exemplo, todas as VLANs executam o PVST+, todas as VLANs executam o Rapid PVST+ ou todas as VLANs executam o MSTP.)
- A propagação do VLAN Trunking Protocol (VTP) da configuração do MST não é suportada.

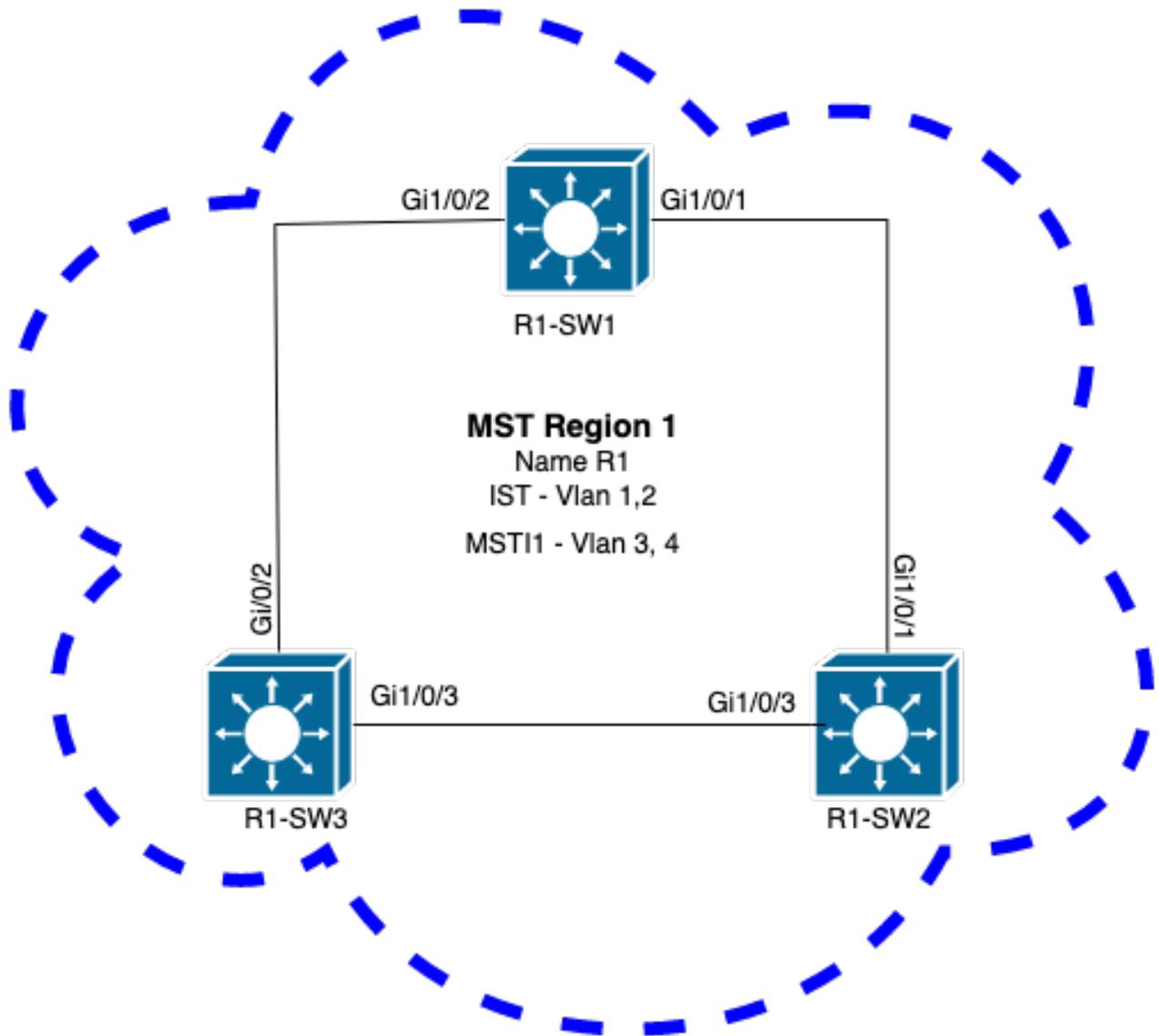
## Troubleshoot

O objetivo é fazer com que a região do MST se comporte como uma ponte CST virtual, do ponto de vista externo à região.

Outros switches, seja em uma região diferente ou em um domínio PVST, veem a região MST como apenas um switch, pois o custo de RootID e Root Path permanece inalterado.

## MST (uma única região)

Topologia



## Configuração

Esses três atributos devem ser configurados da mesma maneira em todos os switches em uma região do MST para convergir corretamente. Os comandos são aplicados no modo de configuração MST.

- Nome
- Número da revisão
- VLAN para mapeamento de instância

```
spanning-tree mst configuration
name <region name>
revision <number>
instance <number> vlan <vlan number>
```

Valide a configuração de atributos com este comando:

```
show running-config | section span
```

Exemplo: configuração de atributos para os switches 1, 2 e 3 na região 1

## R1-SW1

```
R1-SW1#show running-config | section spann
spanning-tree mode mst
spanning-tree extend system-id
spanning-tree mst configuration
  name R1          <---
  revision 1       <---
  instance 1 vlan 3-4 <---
```

## R1-SW2

```
R1-SW2#show running-config | section spann
spanning-tree mode mst
spanning-tree extend system-id
spanning-tree mst configuration
  name R1
  revision 1
  instance 1 vlan 3-4
```

## R1-SW3

```
R1-SW3#show running-config | section spann
spanning-tree mode mst
spanning-tree extend system-id
spanning-tree mst configuration
  name R1
  revision 1
  instance 1 vlan 3-4
```

## Validação

Durante a migração do MST, você pode configurar os parâmetros do MST sem a necessidade de alterar ainda o modo STP.

Siga estas recomendações para evitar possíveis interrupções na rede devido a erros de configuração.

- Verifique a configuração do MST antes de confirmar.
- Verificar a configuração do MST após confirmar

Verifique a configuração do MST antes de confirmar.

Esta verificação ocorre quando o **modo spanning-tree mst** ainda não foi aplicado.

```
show spanning-tree mst
show current
show spanning-tree mst configuration digest
```

**Observação:** `show current` está disponível somente no modo de configuração MST (`spanning-tree mst configuration` submode)

Exemplo: para o switch 1 na região 1

Verifique se o modo STP ainda não está no modo MST

```
R1-SW1#show spanning-tree mst
% Switch is not in mst mode <--
```

Verificar a configuração atual do MST

```
R1-SW1(config-mst)#show current
Current MST configuration
Name [R1]
Revision 1 Instances configured 2
```

```
Instance Vlans mapped
```

```
-----
0 1-2,6-4094
1 3-4
-----
```

**Observação:** `show current` está disponível somente no modo de configuração MST.

**Observação:** `show span mst configuration` e `show current` são comandos equivalentes.

Verificar hash de resumo

```
R1-SW1#show spanning-tree mst configuration digest
% Switch is not in mst mode <--
Name [R1]
Revision 1 Instances configured 2
Digest 0xA423B8DBB209CCF6560F55618AB58726 <--
Pre-std Digest 0x8C9BE88BBC9B84CB8AED635EE008436A
```

**Observação:** a saída `Digest` permite que você saiba se o switch já está no modo MST. O hash de resumo não muda, mesmo que o modo MST ainda não tenha sido habilitado.

**Observação:** os switches Catalyst 9000 executam o protocolo MST do padrão IEEE. Portanto, você deve se concentrar no hash `Digest` em vez do `Digest pré-padrão`

Verificar a configuração do MST após confirmar

```
show current
show pending
show spanning-tree mst configuration digest
abort
```

**Observação:** `show pending` (assim como `show current`) está disponível somente no modo de configuração MST

A saída de `show current` mostra a configuração do MST após sair do submodo MST (que é quando a alteração de configuração é aplicada) enquanto a saída de `show pending` mostra a configuração do MST que foi configurada recentemente, mas não aplicada.

Se, por algum motivo, você precisar reverter as alterações de configuração e ainda estiver no submodo MST, você poderá aplicar o comando **abort** que sai do submodo MST sem aplicar as alterações.

**Observação:** **show pending** (assim como **show current**) está disponível somente no modo de configuração MST

Exemplo: para o switch 1 na região 1

Observe que as configurações atuais e pendentes são as mesmas, o que significa que nenhuma alteração foi feita.

O hash de resumo é o mesmo que o validado na saída anterior.

```
R1-SW1(config)#spanning-tree mst configuration
```

```
R1-SW1(config-mst)#show current
```

```
Current MST configuration
```

```
Name [R1]
```

```
Revision 1 Instances configured 2
```

```
Instance Vlans mapped
```

```
-----  
0 1-2,5-4094
```

```
1 3-4  
-----
```

```
R1-SW1(config-mst)#show pending
```

```
Pending MST configuration
```

```
Name [R1]
```

```
Revision 1 Instances configured 2
```

```
Instance Vlans mapped
```

```
-----  
0 1-2,5-4094
```

```
1 3-4  
-----
```

```
R1-SW1(config-mst)#do show spanning-tree mst configuration digest
```

```
Name [R1]
```

```
Revision 1 Instances configured 2
```

```
Digest 0xA423B8DBB209CCF6560F55618AB58726 <--
```

```
Pre-std Digest 0x8C9BE88BBC9B84CB8AED635EE008436A
```

Uma nova instância é criada e a VLAN 5 é mapeada para ela. Desta vez, a saída de **show current** não mostra a nova instância recentemente configurada, mas **show pending** mostra. Isso é esperado.

Observe que o hash de resumo não foi alterado. Isso ocorre porque a nova configuração se aplica somente quando você sai do modo de configuração MST (submodo de configuração **spanning-tree mst**)

```
R1-SW1(config-mst)#instance 2 vlan 5 <--
```

```
R1-SW1(config-mst)#show current
```

```
Current MST configuration
```

```
Name [R1]
```

```
Revision 1      Instances configured 2
```

```
Instance  Vlans mapped
```

```
-----  
0          1-2,5-4094
```

```
1          3-4  
-----
```

```
R1-SW1(config-mst)#show pending
```

```
Pending MST configuration
```

```
Name      [R1]
```

```
Revision 1      Instances configured 3
```

```
Instance  Vlans mapped
```

```
-----  
0          1-2,6-4094
```

```
1          3-4
```

```
2          5          <--  
-----
```

```
R1-SW1(config-mst)#do show spanning-tree mst configuration digest
```

```
Name      [R1]
```

```
Revision 1      Instances configured 2
```

```
Digest      0xA423B8DBB209CCF6560F55618AB58726      <--
```

```
Pre-std Digest 0x8C9BE88BBC9B84CB8AED635EE008436A
```

Após sair do modo de configuração do MST, as alterações são refletidas. O hash de resumo também é recalculado para corresponder às novas alterações feitas.

```
R1-SW1(config-mst)#exit
```

```
R1-SW1(config)#spanning-tree mst configuration
```

```
R1-SW1(config-mst)#show current
```

```
Current MST configuration
```

```
Name      [R1]
```

```
Revision 1      Instances configured 3
```

```
Instance  Vlans mapped
```

```
-----  
0          1-2,6-4094
```

```
1          3-4
```

```
2          5          <--  
-----
```

```
R1-SW1(config-mst)#show pending
```

```
Pending MST configuration
```

```
Name      [R1]
```

```
Revision 1      Instances configured 3
```

```
Instance  Vlans mapped
```

```
-----  
0          1-2,6-4094
```

```
1          3-4
```

```
2          5          <--  
-----
```

```
R1-SW1(config-mst)#do show spanning-tree mst configuration digest
```

```
Name      [R1]
```

```
Revision 1      Instances configured 3
```

```
Digest      0x083305551908B9A2CC50B482DC577B8F      <--
```

```
Pre-std Digest 0xA8AC09BDF2942058FAF4CE727C9D258F
```

Esses comandos são úteis para validar os parâmetros e a convergência do MST. Além disso, eles fornecem informações relacionadas aos temporizadores, custo e assim por diante.

```
show spanning-tree pathcost method
show spanning-tree root
show spanning-tree summary
show spanning-tree mst
show spanning-tree interface <interface>
```

**Observação:** `show spanning-tree mst` e `show spanning-tree` são equivalentes

Exemplo: para o switch 1 na região 1

Há dois métodos para medir o custo do caminho: curto (legado) e longo. É sempre preferível ser homogêneo ao longo de sua rede de camada 2. Se você executar o método de custo de caminho longo, faça-o ao longo de todos os seus switches que são executados no STP.

```
R1-SW1#show spanning-tree pathcost method
Spanning tree default pathcost method used is long <--
```

Essa saída permite que você use agora o método de custo de caminho, mas também permite que você saiba que o switch executa o protocolo MST padrão e usa o ID de sistema estendido (que é obrigatório quando o MST é usado).

```
R1-SW1#show spanning-tree summary
Switch is in mst mode (IEEE Standard) <--
Root bridge for: none
EtherChannel misconfig guard is enabled
Extended system ID is enabled <--
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Configured Pathcost method used is long <--
```

Name	Blocking	Listening	Learning	Forwarding	STP Active
MST0	0	0	0	3	3
MST1	0	0	0	3	3
2 msts	0	0	0	6	6

IDs de bridge e raiz, prioridades, custos, funções e status de porta, bem como mapeamento de VLAN, podem ser observados nesta saída:

```
R1-SW1#show spanning-tree mst
##### MST0 vlans mapped: 1-2,5-4094
Bridge address 3473.2db8.be80 priority 32768 (32768 sysid 0)
Root address f04a.021e.9500 priority 24576 (24576 sysid 0)
port Gi1/0/2 path cost 0
```

```

Regional Root address f04a.021e.9500 priority      24576 (24576 sysid 0)
                                internal cost 20000      rem hops 19
Operational hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured  hello time 2 , forward delay 15, max age 20, max hops    20

```

```

Interface                Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/1                  Desg FWD 20000    128.1    P2p
Gi1/0/2                  Root FWD 20000    128.2    P2p
Gi1/0/4                  Desg FWD 20000    128.4    P2p

```

```

##### MST1    vlans mapped:    3-4
Bridge        address 3473.2db8.be80 priority    32769 (32768 sysid 1)
Root          address f04a.021e.9500 priority    24577 (24576 sysid 1)
              port    Gi1/0/2      cost      20000      rem hops 19

```

```

Interface                Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/1                  Desg FWD 20000    128.1    P2p
Gi1/0/2                  Root FWD 20000    128.2    P2p
Gi1/0/4                  Desg FWD 20000    128.4    P2p

```

Esse comando mostra o status das funções, a prioridade e o tipo de link do STP a partir da perspectiva da interface, em vez da perspectiva por instância.

```
R1-SW1#show spanning-tree interface gigabitEthernet 1/0/1
```

```

Mst Instance      Role Sts Cost      Prio.Nbr Type
-----
MST0              Desg FWD 20000    128.1    P2p
MST1              Desg FWD 20000    128.1    P2p

```

```
R1-SW1#show spanning-tree interface gigabitEthernet 1/0/2
```

```

Mst Instance      Role Sts Cost      Prio.Nbr Type
-----
MST0              Root FWD 20000    128.2    P2p
MST1              Root FWD 20000    128.2    P2p

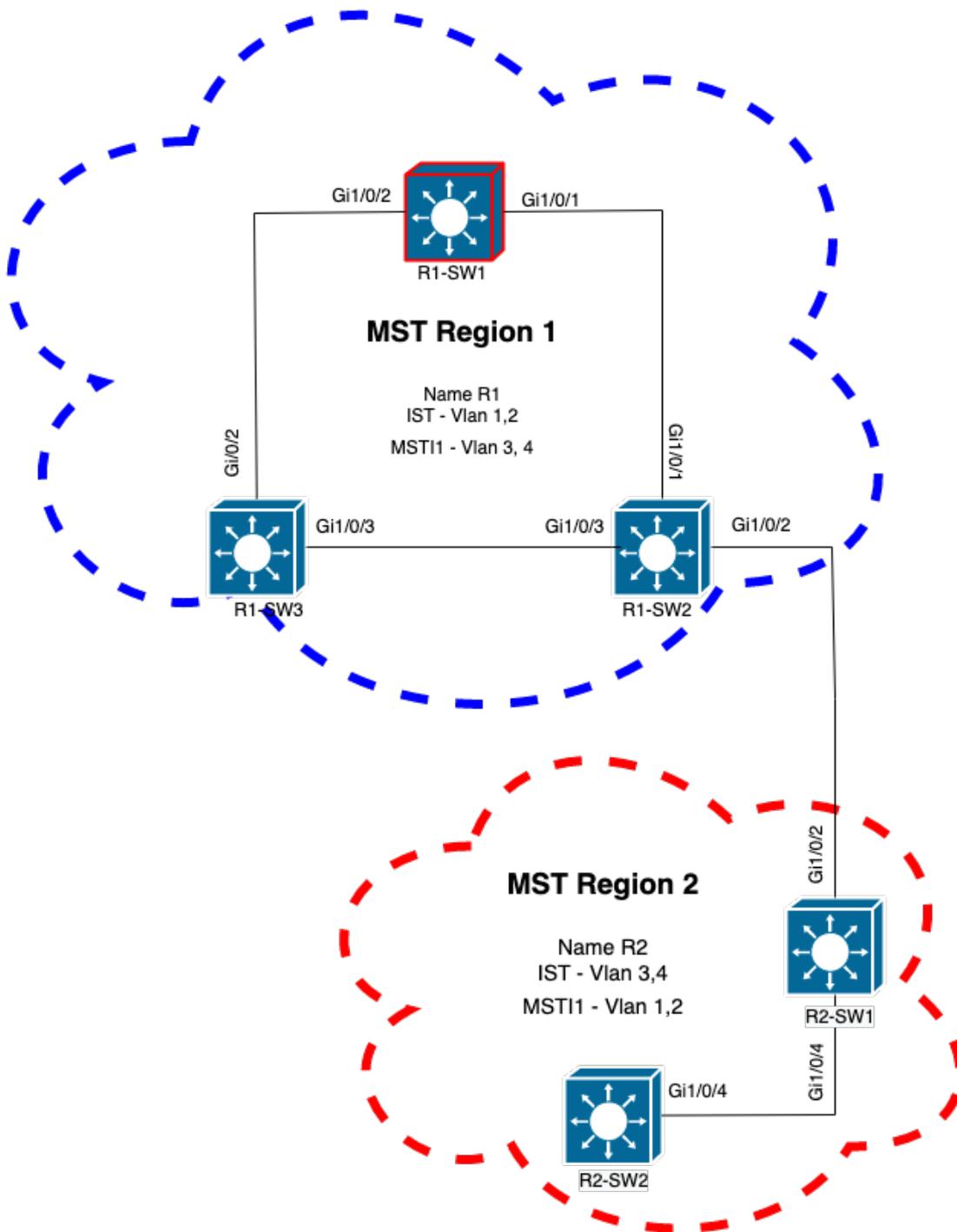
```

## Sincronização entre regiões

A região 2 foi adicionada à topologia. A finalidade é verificar o processo de como duas regiões diferentes interagem e convergem. Somente os switches de limite ocorrem nesta comunicação.

Já que ambas as extremidades do link têm o mesmo processo de comunicação. Esta seção enfoca as saídas de **show spanning-tree mst** de R1-SW2 e duas BPDUs tiradas de uma captura de pacotes.

## Topologia



## Validação

Essa é a comunicação inicial entre R1-SW2 da Região 1 e R2-SW1 da Região 2. Assim que uma conexão é estabelecida entre os dois dispositivos, eles enviam uma BPDU.

Concentre-se na interface Gi1/0/2 de R2-SW1, que está bloqueando (BLK) como estado inicial. Lembre-se de que uma porta de switch entra no estado BLK no momento do processo de eleição.

```
R2-SW1#show spanning-tree mst
```

```

MST0
! Output omitted for brevity Interface Role Sts Cost Prio.Nbr Type -----
-----
                               Gi1/0/2           Desg BLK 20000      128.2
P2p
Gi1/0/4           Root FWD 20000      128.4      P2p

MST1
! Output omitted for brevity Interface Role Sts Cost Prio.Nbr Type -----
-----
                               Gi1/0/2           Desg BLK 20000      128.2
P2p
Gi1/0/4           Root FWD 20000      128.4      P2p

```

Na captura de pacotes, é observado esse primeiro BPDU, com os sinalizadores de Função de Porta mostrados como Designado e a Proposta.

Isso significa que a comunicação já começou e que ambas as portas iniciaram o processo de sincronização para estabelecer um acordo e definir as funções e os estados das portas. Tudo começa com o mecanismo da proposta.

```

IEEE 802.3 Ethernet
  Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
  Source: Cisco_05:d6:02 (f0:4a:02:05:d6:02)
  Length: 121
Logical-Link Control
Spanning Tree Protocol
  Protocol Identifier: Spanning Tree Protocol (0x0000)
  Protocol Version Identifier: Multiple Spanning Tree (3)
  BPDU Type: Rapid/Multiple Spanning Tree (0x02)
BPDU flags: 0x0e, Port Role: Designated, Proposal
  0... .... = Topology Change Acknowledgment: No
  .0.. .... = Agreement: No
  ..0. .... = Forwarding: No
  ...0 .... = Learning: No
  .... 11.. = Port Role: Designated (3)
  .... ..1. = Proposal: Yes
  .... ...0 = Topology Change: No
Root Identifier: 24576 / 0 / f0:4a:02:1e:95:00
Root Path Cost: 20004
Bridge Identifier: 32768 / 0 / a0:f8:49:10:47:80
Port identifier: 0x8002
Message Age: 2
Max Age: 20
Hello Time: 2
Forward Delay: 15
Version 1 Length: 0
Version 3 Length: 80
MST Extension

```

Após a troca de BPDUs entre switches, o estado muda para learning (LRN).

Depois que R2-SW1 recebe o primeiro BPDU mostrado anteriormente, o estado LRN é o primeiro estado de transição após o estado de bloqueio.

```
R2-SW1#show spanning-tree mst
```

MST0

! Output omitted for brevity

Interface	Role	Sts	Cost	Prio.Nbr	Type
<b>Gi1/0/2</b>	<b>Desg</b>	<b>LRN</b>	<b>20000</b>	<b>128.2</b>	<b>P2p</b>
Gi1/0/4	Root	FWD	20000	128.4	P2p

MST1

! Output omitted for brevity

Interface	Role	Sts	Cost	Prio.Nbr	Type
<b>Gi1/0/2</b>	<b>Desg</b>	<b>LRN</b>	<b>20000</b>	<b>128.2</b>	<b>P2p</b>
Gi1/0/4	Root	FWD	20000	128.4	P2p

Quando um dos pares estabelece um acordo e ocorre a sincronização (o vizinho é aceito como o caminho superior para a raiz), os links imediatamente passam para o estado forwarding.

Aqui você pode observar o BPDU com os Flags definidos como aprendizagem, ele também inclui o flag de notificação de alteração de topologia que é acionado assim que a porta faz a transição de LRN para encaminhamento (FWR).

Nesse estado, o MST determina se a porta participa ou não do encaminhamento de quadros (estado BLK).

IEEE 802.3 Ethernet

Logical-Link Control

Spanning Tree Protocol

Protocol Identifier: Spanning Tree Protocol (0x0000)

Protocol Version Identifier: Multiple Spanning Tree (3)

BPDU Type: Rapid/Multiple Spanning Tree (0x02)

**BPDU flags: 0x3d, Forwarding, Learning, Port Role: Designated, Topology Change**

0... .... = Topology Change Acknowledgment: No

.0.. .... = Agreement: No

**..1. .... = Forwarding: Yes**

**...1 .... = Learning: Yes**

**.... 11.. = Port Role: Designated (3)**

.... ..0. = Proposal: No

**.... ...1 = Topology Change: Yes**

Root Identifier: 24576 / 0 / f0:4a:02:1e:95:00

Root Path Cost: 20004

Bridge Identifier: 32768 / 0 / a0:f8:49:10:47:80

Port identifier: 0x8002

Message Age: 2

Max Age: 20

Hello Time: 2

Forward Delay: 15

Version 1 Length: 0

Version 3 Length: 80

MST Extension

Finalmente, a porta do Switch entra no estado forwarding após atravessar todos os estados envolvidos na criação da topologia de rede.

Esse seria o último estado da porta, com a função designada (Desg) e o status FDW.

```
R2-SW1#show spanning-tree mst
```

```
MST0
```

```
! Output omitted for brevity
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
Gi1/0/2	<b>Desg</b>	<b>FWD</b>	<b>20000</b>	<b>128.2</b>	<b>P2p</b>
Gi1/0/4	Root	FWD	20000	128.4	P2p

```
MST1
```

```
! Output omitted for brevity
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
Gi1/0/2	<b>Desg</b>	<b>FWD</b>	<b>20000</b>	<b>128.2</b>	<b>P2p</b>
Gi1/0/4	Root	FWD	20000	128.4	P2p

## Debugs

Esses bugs foram ativados durante a comunicação entre R2-SW1 e R1-SW2.

```
debug spanning-tree mstp roles
debug spanning-tree mstp tc
debug spanning-tree mstp boundary
```

### Exemplo:

```
R2-SW1#show debugging
```

```
Packet Infra debugs:
```

Ip Address	Port
-----	-----

```
Multiple Spanning Tree:
```

```
MSTP port ROLES changes debugging is on
MSTP Topology Change notifications debugging is on
MSTP port BOUNDARY flag changes debugging is on
```

## Logs observados

```
%LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to down
%LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to up
MST[0]: Gi1/0/2 is now designated port
MST[0]: Gi1/0/2 becomes designated - clearing BOUNDARY flag
MST[1]: Gi1/0/2 is now designated port
MST[0]: port Gi1/0/2 received external tc
MST[0]: port Gi1/0/2 received external tc
MST[1]: port Gi1/0/2 received tc
MST[0]: port Gi1/0/2 received external tc
MST[0]: port Gi1/0/2 received external tc
MST[1]: port Gi1/0/2 received tc
```

```

MST[0]: port Gi1/0/2 received external tc
MST[0]: port Gi1/0/2 received external tc
MST[1]: port Gi1/0/2 received tc
MST[0]: port Gi1/0/2 initiating tc
MST[1]: port Gi1/0/2 initiating tc
MST[0]: port Gi1/0/2 received external tc
MST[0]: port Gi1/0/2 received external tc
MST[1]: port Gi1/0/2 received tcsho span
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/2, changed state to up
MST[0]: port Gi1/0/3 received internal tc
MST[0]: port Gi1/0/3 received internal tc
MST[0]: port Gi1/0/3 received internal tc

```

## Falha de Simulação de PVST

A simulação de PVST é o mecanismo que o MST usa para se comunicar com switches não MST.

Os switches PVST não reconhecem as BPDUs do MST porque são simplesmente diferentes. É por isso que é importante entender as diferenças entre PVST e MST BPDUs.

### PVST BPDUs vs MST BPDUs

Duas BPDUs foram capturadas, uma para PVST e outra para MST, observe as diferenças entre elas.

#### PVST

- O PVST envia um BPDU para cada VLAN configurada no switch. Portanto, ter 100 VLANS configuradas significa que 100 BPDUs são enviados por todas as portas para criar sua própria topologia sem loops.
- O PVST é baseado no STP clássico

```

Ethernet II, Src: Cisco_06:19:01 (f0:4a:02:06:19:01), Dst: PVST+ (01:00:0c:cc:cc:cd)
  Destination: PVST+ (01:00:0c:cc:cc:cd)
  Source: Cisco_06:19:01 (f0:4a:02:06:19:01)
  Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 7, DEI: 0, ID: 3
  111. .... = Priority: Network Control (7)
  ...0 .... = DEI: Ineligible
  .... 0000 0000 0011 = ID: 3
  Length: 50
Logical-Link Control
  DSAP: SNAP (0xaa)
  SSAP: SNAP (0xaa)
  Control field: U, func=UI (0x03)
  Organization Code: 00:00:0c (Cisco Systems, Inc)
  PID: PVSTP+ (0x010b)
Spanning Tree Protocol
  Protocol Identifier: Spanning Tree Protocol (0x0000)
  Protocol Version Identifier: Spanning Tree (0)
  BPDU Type: Configuration (0x00)
  BPDU flags: 0x01, Topology Change
    0... .... = Topology Change Acknowledgment: No
    .... ...1 = Topology Change: Yes
  Root Identifier: 32768 / 0 / 68:9e:0b:a0:f5:80
  Root Bridge Priority: 32768
  Root Bridge System ID Extension: 0
  Root Bridge System ID: Cisco_a0:f5:80 (68:9e:0b:a0:f5:80)

```

Root Path Cost: 20000  
 Bridge Identifier: 32768 / 0 / f0:4a:02:06:19:00  
     Bridge Priority: 32768  
     Bridge System ID Extension: 0  
     Bridge System ID: Cisco\_06:19:00 (f0:4a:02:06:19:00)  
 Port identifier: 0x8001  
 Message Age: 1  
 Max Age: 20  
 Hello Time: 2  
 Forward Delay: 15  
**Originating VLAN (PVID): 3**  
     Type: Originating VLAN (0x0000)  
     Length: 2  
     **Originating VLAN: 3**

## MST

- O MST envia um único BPDU para todas as instâncias do MST configuradas no switch. Isso é obtido graças à extensão MST (M records), que tem as informações de todas as instâncias.
- O MST é baseado no RSTP, o que significa que todos os mecanismos intrínsecos deste protocolo foram herdados do MST.
- Os temporizadores são definidos pelo IST e afetam todas as outras instâncias dentro de uma região

### IEEE 802.3 Ethernet

**Destination: Spanning-tree-(for-bridges)\_00 (01:80:c2:00:00:00)**  
 Source: Cisco\_b8:be:81 (34:73:2d:b8:be:81)  
 Length: 121

### Logical-Link Control

DSAP: Spanning Tree BPDU (0x42)  
 SSAP: Spanning Tree BPDU (0x42)  
 Control field: U, func=UI (0x03)

### Spanning Tree Protocol

**Protocol Identifier: Spanning Tree Protocol (0x0000)**  
**Protocol Version Identifier: Multiple Spanning Tree (3)**  
**BPDU Type: Rapid/Multiple Spanning Tree (0x02)**  
**BPDU flags: 0x0e, Port Role: Designated, Proposal**  
     0... .... = **Topology Change Acknowledgment: No**  
     .0.. .... = **Agreement: No**  
     ..0. .... = **Forwarding: No**  
     ...0 .... = **Learning: No**  
     .... 11.. = **Port Role: Designated (3)**  
     .... ..1. = **Proposal: Yes**  
     .... ...0 = **Topology Change: No**

Root Identifier: 32768 / 0 / 34:73:2d:b8:be:80  
     Root Bridge Priority: 32768  
     Root Bridge System ID Extension: 0  
     Root Bridge System ID: Cisco\_b8:be:80 (34:73:2d:b8:be:80)  
 Root Path Cost: 0  
 Bridge Identifier: 32768 / 0 / 34:73:2d:b8:be:80  
     Bridge Priority: 32768  
     Bridge System ID Extension: 0  
     Bridge System ID: Cisco\_b8:be:80 (34:73:2d:b8:be:80)  
 Port identifier: 0x8001  
 Message Age: 0  
 Max Age: 20  
 Hello Time: 2  
 Forward Delay: 15  
 Version 1 Length: 0  
 Version 3 Length: 80

### **MST Extension**

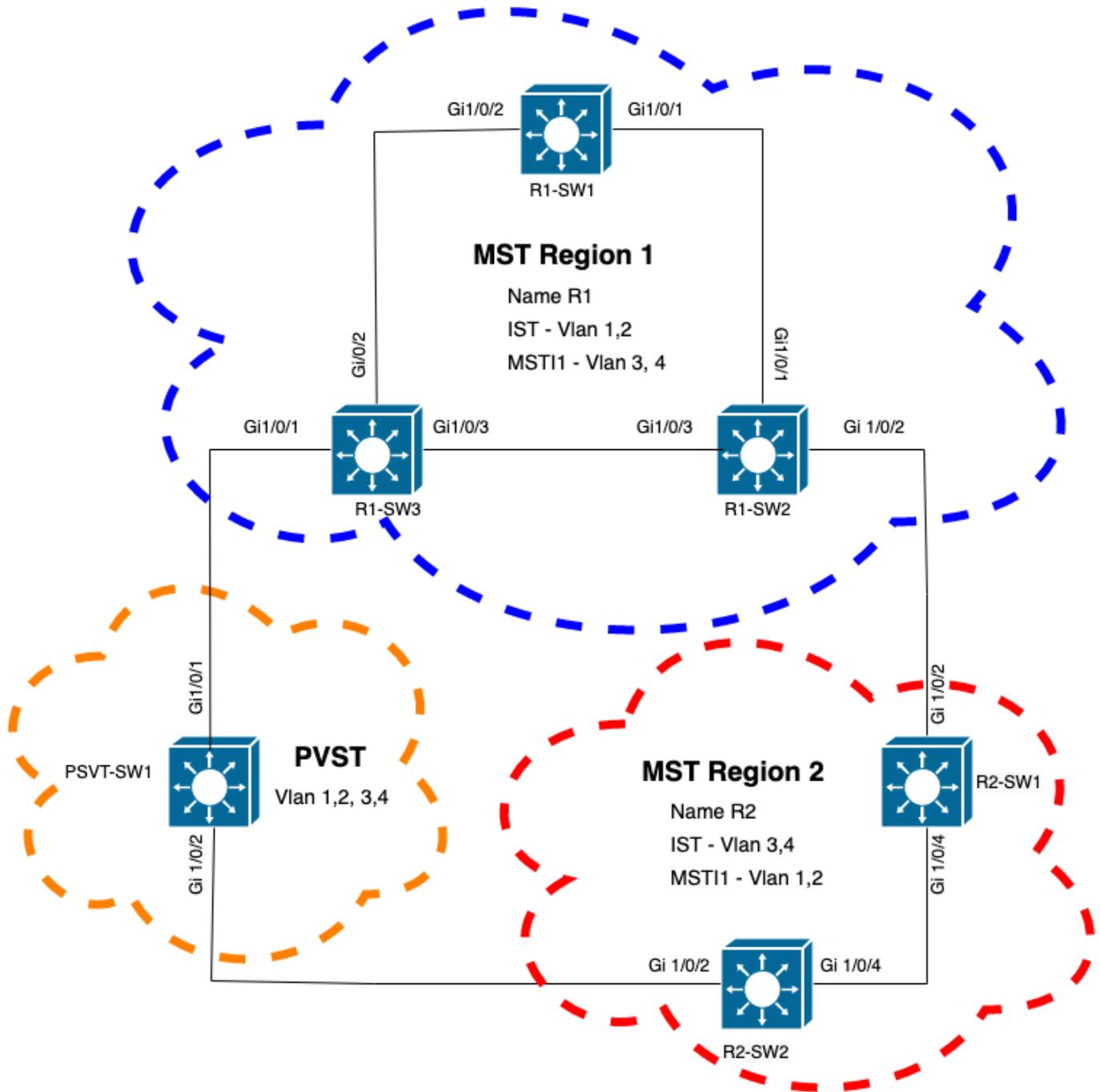
```
MST Config ID format selector: 0
MST Config name: R1
MST Config revision: 1
MST Config digest: a423b8dbb209ccf6560f55618ab58726
CIST Internal Root Path Cost: 0
CIST Bridge Identifier: 32768 / 0 / 34:73:2d:b8:be:80
  CIST Bridge Priority: 32768
  CIST Bridge Identifier System ID Extension: 0
  CIST Bridge Identifier System ID: Cisco_b8:be:80 (34:73:2d:b8:be:80)
CIST Remaining hops: 20
```

### **MSTID 1, Regional Root Identifier 32768 / 34:73:2d:b8:be:80**

```
MSTI flags: 0x0e, Port Role: Designated, Proposal
  0... .... = Topology Change Acknowledgment: No
  .0.. .... = Agreement: No
  ..0. .... = Forwarding: No
  ...0 .... = Learning: No
  .... 11.. = Port Role: Designated (3)
  .... ..1. = Proposal: Yes
  .... ...0 = Topology Change: No
1000 .... = Priority: 0x8
.... 0000 0000 0001 = MSTID: 1
Regional Root: Cisco_b8:be:80 (34:73:2d:b8:be:80)
Internal root path cost: 0
Bridge Identifier Priority: 8
Port identifier priority: 8
Remaining hops: 20
```

## **Topologia**

Switch com PVST foi adicionado à rede. Ela interconecta as regiões 1 e 2.



## Validação

Depois que o switch PVST foi conectado, a porta de limite (gi1/0/1) do switch R1-SW3 da região 1 vai para o PVST inconsistente e bloqueia a porta.

```
R1-SW3#show spanning-tree mst
```

```
##### MST0      vlans mapped: 1-2,5-4094
Bridge          address f04a.021e.9500  priority      32768 (32768 sysid 0)
Root            address 689e.0ba0.f580  priority      16385 (16384 sysid 1)
                port      Gi1/0/1          path cost     20000
Regional Root  this switch
Operational     hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured      hello time 2 , forward delay 15, max age 20, max hops 20
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

```

-----
Gi1/0/1          Root BKN*20000    128.1    P2p Bound (PVST) *PVST_Inc
Gi1/0/2          Desg FWD 20000    128.2    P2p
Gi1/0/3          Desg FWD 20000    128.3    P2p

##### MST1      vlans mapped:    3-4
Bridge          address f04a.021e.9500 priority    32769 (32768 sysid 1)
Root            address 3473.2db8.be80 priority    32769 (32768 sysid 1)
                port      Gi1/0/2      cost      20000      rem hops 19

Interface          Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/1          Mstr BKN*20000    128.1    P2p Bound (PVST) *PVST_Inc
Gi1/0/2          Root FWD 20000    128.2    P2p
Gi1/0/3          Altn BLK 20000    128.3    P2p

```

**Observação:** saídas semelhantes são observadas em R2-SW2 da região 2, que é outra porta de limite.

Isso aconteceu porque qualquer uma dessas regras foi quebrada

- Se a bridge raiz para CIST estiver dentro de uma região não MST, a prioridade de spanning-tree das VLANs 2 em diante dentro desse domínio deverá ser melhor (menor) que a da VLAN 1.
- Se a bridge raiz para o CIST estiver dentro de uma região do MST, as VLANs 2 e posteriores definidas nos domínios não-MST devem ter suas prioridades de spanning tree piores (maiores) que a raiz do CIST.

Observe as configurações inválidas que foram configuradas no switch para enfrentar esse problema:

Caso 1. O switch PVST é a raiz para as VLANs 2-4, no entanto, as VLANs 2-4 têm uma prioridade pior (maior) que a VLAN 1. Nesse caso, todos os switches, exceto o switch PVST, têm a prioridade de STP padrão (32768)

```

PVST-SW1# show run | inc span
spanning-tree mode pvst
spanning-tree extend system-id
spanning-tree vlan 1 priority 4096 <--
spanning-tree vlan 2-4 priority 16384 <--
spanning-tree mst configuration

```

Log observado:

```

%SPANNTREE-2-PVSTSIM_FAIL: Blocking root port Gi1/0/1: Inconsitent inferior PVST BPDU received on
VLAN 2, claiming root 16386:689e.0ba0.f580

```

Caso 2. O switch PVST não é a raiz para as VLANs 1, no entanto, as VLANs 2 a 4 têm uma prioridade melhor (menor) que a raiz. Nesse caso, a raiz tem a prioridade padrão 24576. Isso significa que a bridge raiz não é a raiz de todas as VLANs

```

PVST-SW1#show run | inc span
spanning-tree mode pvst

```

```
spanning-tree extend system-id
spanning-tree vlan 1 prio 32768 <-- higher priority than the root
spanning-tree vlan 2-4 priority 16384 <-- lower priority than the root
spanning-tree mst configuration
```

Log observado:

```
%SPANTREE-2-PVSTSIM_FAIL: Blocking root port Gi1/0/1: Inconsistent inferior PVST BPDU received
on VLAN 2, claiming root 40962:689e.0ba0.f580
```

Depois de considerar as regras mencionadas anteriormente, você poderá usar essas configurações válidas para excluir esse problema.

Caso 1.

```
PVST-SW1# show run | inc span
spanning-tree mode pvst
spanning-tree extend system-id
spanning-tree vlan 1 priority 16384 <-- VLAN 1 has a higher priority than all other VLANs
spanning-tree vlan 2-4 priority 4096 <--
spanning-tree mst configuration
```

Log observado:

```
%SPANTREE-2-PVSTSIM_OK: PVST Simulation nconsistency cleared on port GigabitEthernet1/0/1.
```

Caso 2.

```
PVST-SW1#show run | inc span
spanning-tree mode pvst
spanning-tree extend system-id
spanning-tree vlan 1 prio 32768 <-- higher priority than the root
spanning-tree vlan 2-4 priority 40960 <-- higher priority than the root
spanning-tree mst configuration
```

Log observado:

```
%SPANTREE-2-PVSTSIM_OK: PVST Simulation nconsistency cleared on port GigabitEthernet1/0/1.
```

## Debugs

Verifique as BPDUs com as depurações de BPDU se a captura de pacotes não for possível.

```
debug spanning-tree mstp bpdu receive
debug spanning-tree mstp bpdu transmit
```

Exemplo: Para o switch 2 na região 2 conectado ao switch PVST

```
R2-SW2#debug spanning-tree mstp bpdu receive
MSTP BPDUs RECEIVED dump debugging is on
R2-SW2#debug spanning-tree mstp bpdu transmit
MSTP BPDUs TRANSMITTED dump debugging is on
R2-SW2#debug condition interface gigabitEthernet 1/0/2 <-- interface facing PVST switch
```

```
R2-SW2#show logging
```

```
! Output omitted for brevity
```

```
%LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to down
```

```

%LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to up
MST[0]:-TX> Gi1/0/2 BPDU Prot:0 Vers:3 Type:2
MST[0]: Role :Desg Flags[P] Age:2 RemHops:19
MST[0]: CIST_root:16385.689e.0ba0.f580 Cost :40000
MST[0]: Reg_root :32768.f04a.0205.d600 Cost :20000
MST[0]: Bridge_ID:32768.a0f8.4910.4780 Port_ID:32770
MST[0]: max_age:20 hello:2 fwdelay:15
MST[0]: V3_len:80 region:R2 rev:1 Num_mrec: 1
MST[1]:-TX> Gi1/0/2 MREC
MST[1]: Role :Desg Flags[MAP] RemHops:20
MST[1]: Root_ID :32769.a0f8.4910.4780 Cost :0
MST[1]: Bridge_ID:32769.a0f8.4910.4780 Port_id:130
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/2, changed state to up
MST[0]:-TX> Gi1/0/2 BPDU Prot:0 Vers:3 Type:2
MST[0]: Role :Desg Flags[P] Age:2 RemHops:19
MST[0]: CIST_root:16385.689e.0ba0.f580 Cost :40000
MST[0]: Reg_root :32768.f04a.0205.d600 Cost :20000
MST[0]: Bridge_ID:32768.a0f8.4910.4780 Port_ID:32770
MST[0]: max_age:20 hello:2 fwdelay:15
MST[0]: V3_len:80 region:R2 rev:1 Num_mrec: 1
MST[1]:-TX> Gi1/0/2 MREC
MST[1]: Role :Desg Flags[MAP] RemHops:20
MST[1]: Root_ID :32769.a0f8.4910.4780 Cost :0
MST[1]: Bridge_ID:32769.a0f8.4910.4780 Port_id:130
MST[0]:

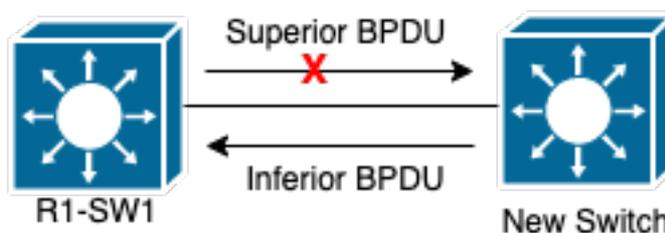
```

```
MST[0]: Role :Desg Flags[FLTc] Age:0
```

```
MST[0]: CIST_root:16385.689e.0ba0.f580 Cost :0
```

```
MST[0]: Bridge_ID:16385.689e.0ba0.f580 Port_ID:32770
```

```
MST[0]: max_age:20 hello:2 fwdelay:15
```



## Contestação P2P Topologia

Nesta seção, você pode observar um problema com dois dispositivos que não conseguiram estabelecer um acordo e definir o status das portas corretamente.

```
R1-SW1#show spanning-tree mst
```

```

##### MST0      vlans mapped: 1-2,5-4094
Bridge          address 3473.2db8.be80  priority      32768 (32768 sysid 0)
Root            address 689e.0ba0.f580    priority      4097 (4096 sysid 1)
                port    Gi1/0/2             path cost    20000
Regional Root  address f04a.021e.9500 priority      24576 (24576 sysid 0)

```

Explicação Nesta seção, você pode observar um problema com dois dispositivos que não conseguiram estabelecer um acordo e definir o status das portas corretamente.

```

                                internal cost 20000      rem hops 19
Operational hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured  hello time 2 , forward delay 15, max age 20, max hops    20

```

```

Interface          Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/1            Desg FWD 20000   128.1   P2p
Gi1/0/2            Root FWD 20000   128.2   P2p
Gi1/0/4            Desg BLK 20000 128.2   P2p Dispute

```

```

##### MST1      vlans mapped: 3-4
Bridge          address 3473.2db8.be80 priority 32769 (32768 sysid 1)
Root            address f04a.021e.9500 priority 24577 (24576 sysid 1)
                port Gi1/0/2      cost 20000      rem hops 19

```

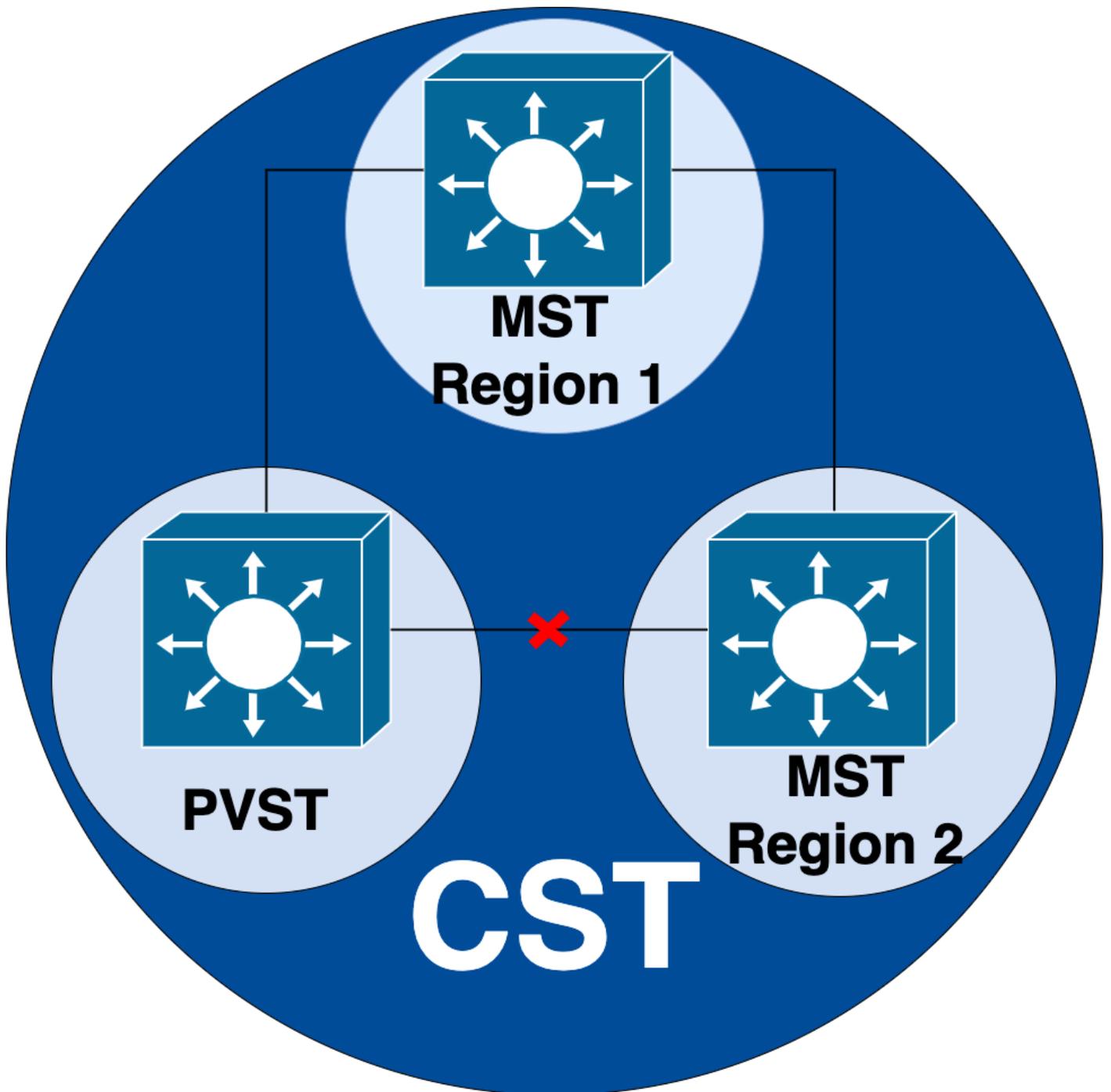
```

Interface          Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/1            Desg FWD 20000   128.1   P2p
Gi1/0/2            Root FWD 20000   128.2   P2p
Gi1/0/4            Desg BLK 20000 128.2   P2p Dispute

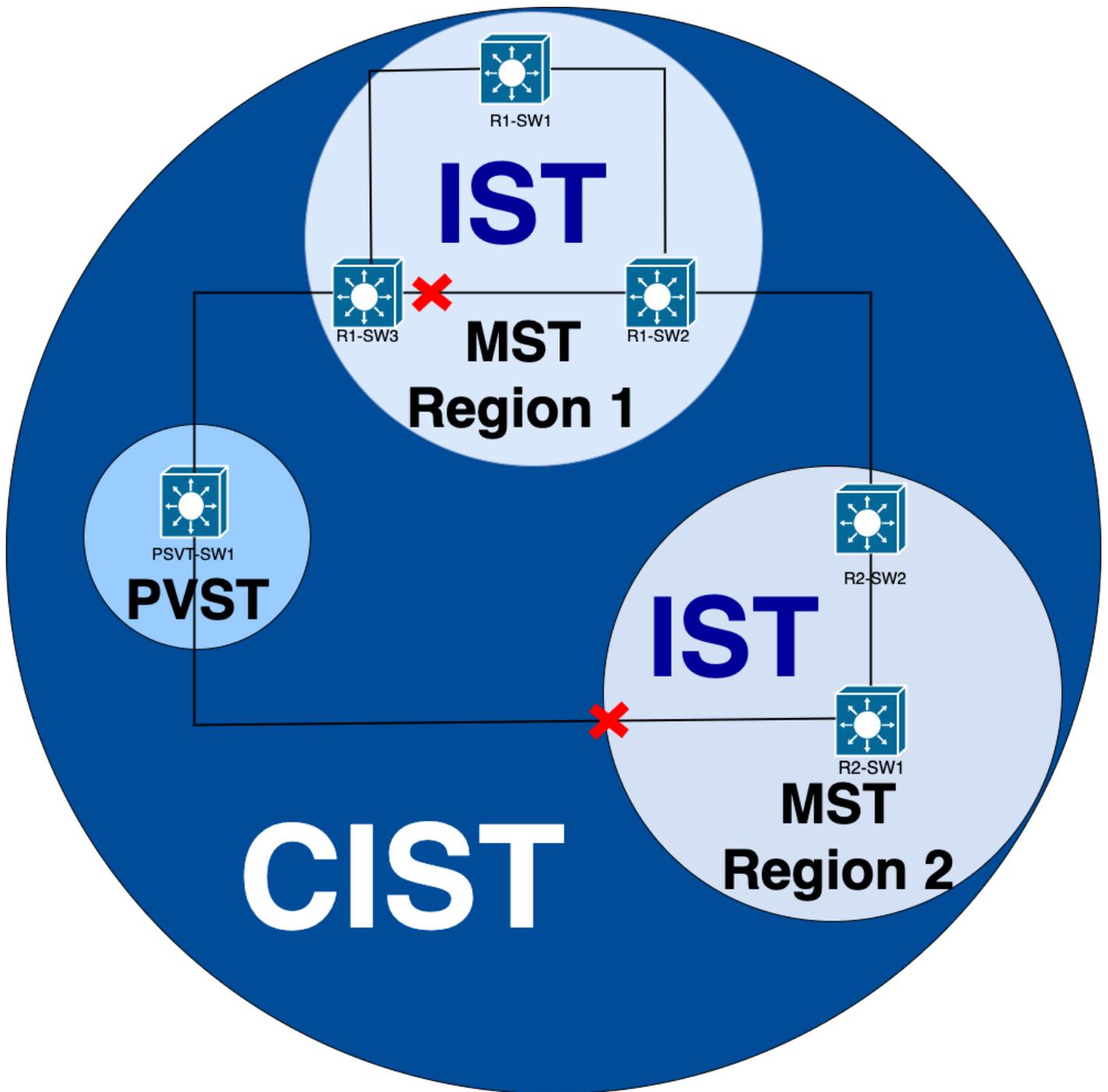
```

R1-SW1 (raiz) notou que novos dispositivos estavam conectados a ele. Assim, ele envia seu BPDU e se define como raiz. Ele recebeu um BPDU que especifica que, do outro lado no link, os flags são definidos como função de porta: designado, encaminhamento e aprendizagem. Isso significa que o novo switch conectou estados que têm melhor caminho para acessar a raiz. No entanto, isso não é possível, pois R1-SW1 é a raiz e não há caminho melhor para ela. Como os dois switches não puderam estabelecer o acordo e definir as portas corretamente (pois ambos os BPDUs mostram um caminho melhor para a raiz), R1-SW1 assume que o novo switch não recebe seus BPDUs e define o status da porta como Contestação P2P para evitar cenários unidirecionais que podem causar loops.

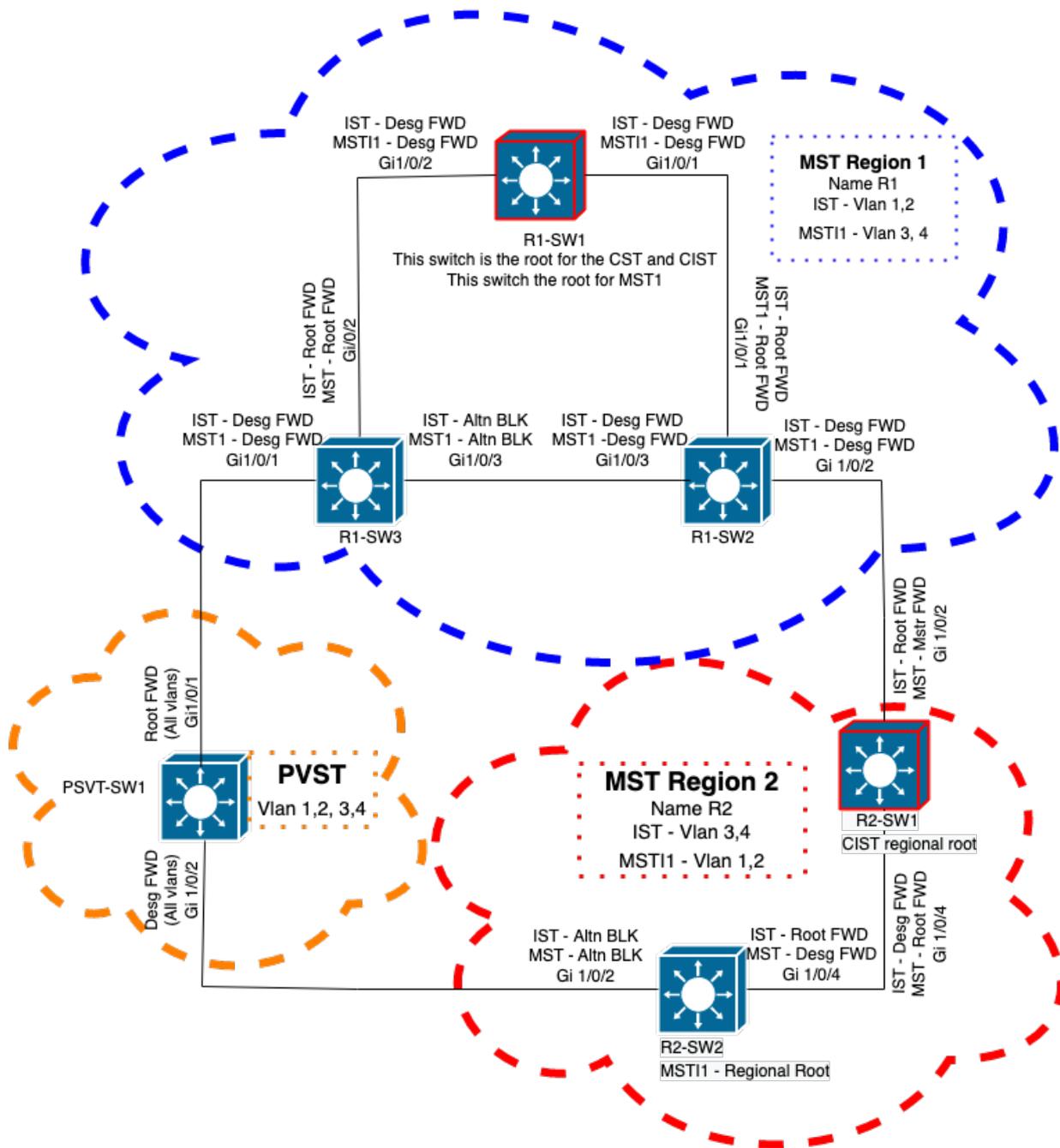
**Abordagens MST** Como observado neste documento, o MST pode ser mais complicado desde que mais switches sejam adicionados à rede. Por isso, é importante ter abordagens diferentes para a mesma rede. Exemplo: Se o problema observado não estiver na região do MST, mas em um domínio PVST, você poderá ter uma visão mais ampla e ignorar qualquer coisa dentro das regiões do MST (perspectiva do CST).



Por outro lado, se houver suspeita de que o problema ocorre entre as regiões do MST ou dentro de uma região, o CIST oferece uma perspectiva melhor.



Se necessário, você pode se concentrar nas funções de porta e no status dos switches



## Informações Relacionadas

- [Compreendendo o protocolo múltiplo de extensão de árvore \(802.1s\)](#)
- [Guia de configuração da camada 2, Cisco IOS XE Amsterdam 17.3.x \(Switches Catalyst 9300\)](#)
- [Guia de configuração de Camada 2 e Camada 3, Cisco IOS XE Everest 16.5.1a \(Switches Catalyst 9300\)](#)
- [Simulação de PVST em Switches MST](#)
- ID de bug Cisco [CSCvy02075](#) - O switch encaminha o tráfego recebido nas portas no estado BLK de bloqueio

## Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.

## Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.