

Troubleshooting do Spanning Tree Protocol em um 5000 Series Switch do nexa

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Troubleshooting](#)

[Raiz STP](#)

[Relação STP](#)

[Investigação BPDU com Ethalyzer](#)

[Convergência de STP](#)

[Mapeamento externo VLAN](#)

[O STP debuga](#)

[O nexa 5000 não processou BPDU](#)

Introdução

Este documento descreve vários métodos para pesquisar defeitos os problemas comuns relativos ao Spanning Tree Protocol (STP).

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Sistema operacional CLI do nexa
- STP

[Componentes Utilizados](#)

Este documento não se restringe a versões de software e hardware específicas.

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a sua rede estiver ativa, certifique-se de que entende o impacto

potencial de qualquer comando.

Troubleshooting

Esta seção cobre alguns métodos para pesquisar defeitos problemas comuns com STP.

Raiz STP

A fim pesquisar defeitos uma edição STP, é crítico saber que interruptor é atualmente a raiz. O comando mostrar a raiz STP em um 5000 Series Switch do nexo é:

```
Nexus-5000# show spanning-tree vlan 1

VLAN0001
Spanning tree enabled protocol rstp
Root ID Priority 32769
Address c84c.75fa.6000
This bridge is the root
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
Address c84c.75fa.6000
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
```

Estão aqui alguns outros comandos relevant:

```
Nexus-5000# show spanning-tree vlan 1 detail
Nexus-5000# show spanning-tree vlan 1 summary
```

Uma vez que você determinou quem a raiz atual é, você pode verificar a história do evento para ver se mudou e de onde as notificações da alteração de topologia originam.

```
Nexus-5000# show spanning-tree internal event-history tree 1 brief
2012:11:06 13h:44m:20s:528204us T_EV_UP
VLAN0001 [0000.0000.0000.0000 C 0 A 0 R none P none]
2012:11:06 13h:44m:21s:510394us T_UT_SBPDU
VLAN0001 [8001.547f.ee18.e441 C 0 A 0 R none P Pol]
2012:11:06 13h:44m:21s:515129us T_EV_M_FLUSH_L
VLAN0001 [1001.001b.54c2.5a42 C 6 A 5 R Pol P none]
2012:11:06 13h:44m:23s:544632us T_EV_M_FLUSH_R
VLAN0001 [1001.001b.54c2.5a42 C 6 A 5 R Pol P Pol]
2012:11:06 13h:44m:24s:510352us T_EV_M_FLUSH_R
VLAN0001 [1001.001b.54c2.5a42 C 6 A 5 R Pol P Pol]
```

Dica: Estão aqui algumas definições para os acrônimos que aparecem na saída dos comandos. **SBPDU**: Bridge Protocol Data Unit superior recebido; **FLUSH_L**: Resplendor do Local; **FLUSH_R**: Nivele do switch remoto.

Nota: As versões NX-OS antes da versão 5.1(3)N1(1) não registram mais de 149 eventos, e o log não rola.

Relação STP

Este comando é usado a fim indicar os eventos para uma relação.

```
Nexus-5000# show spanning-tree internal event-history tree 1 interface ethernet 1/3 brief
```

```
2012:11:05 13h:42m:20s:508027us P_EV_UP Eth1/3 [S DIS R Unkw A 0 Inc no]
2012:11:05 13h:42m:20s:508077us P_STATE Eth1/3 [S BLK R Desg A 0 Inc no]
2012:11:05 13h:42m:20s:508294us P_STATE Eth1/3 [S LRN R Desg A 0 Inc no]
2012:11:05 13h:42m:20s:508326us P_STATE Eth1/3 [S FWD R Desg A 0 Inc no]
```

Este comando é usado a fim investigar mudanças STP em uma relação. Esta saída oferece muitos detalhes:

```
Nexus-5000# show spanning-tree internal info tree 1 interface port-channel 11
```

```
----- STP Port Info (vdc 1, tree 1, port Po11) -----
dot1d info: port_num=4106, ifi=0x1600000a (port-channell1)
ISSU FALSE non-disr, prop 0, ag 0, flush 0 peer_not_disputed_count 0
if_index          0x1600000a
namestring port-channell1
..... cut to save space .....
```

stats

```
fwd_transition_count 1          bpdus_in      40861  bpdus_out    40861
config_bpdu_in      0          rstp_bpdu_in 40861  tcn_bpdu_in  0
config_bpdu_out     0          rstp_bpdu_out 40861  tcn_bpdu_out 0
bpdufilter_drop_in  0
bpduguard_drop_in  0
err_dropped_in     0
sw_flood_in        0
..... cut to save space .....
```

Investigação BPDU com Ethalyzer

Esta seção descreve como usar Ethalyzer a fim capturar BPDU:

```
Ethalyzer local interface inbound-hi display-filter "vlan.id == 1 && stp"
```

Example:

```
Nexus-5000# ethalyzer local interface inbound-hi display-filter "vlan.id == 1 && stp"
```

Capturing on eth4

```
2013-05-11 13:55:39.280951 00:05:73:f5:d6:27 -> 01:00:0c:cc:cc:cd STP RST.
Root = 33768/00:05:73:ce:a9:7c Cost = 1 Port = 0x900a
2013-05-11 13:55:40.372434 00:05:73:ce:a9:46 -> 01:00:0c:cc:cc:cd STP RST.
Root = 33768/00:05:73:ce:a9:7c Cost = 0 Port = 0x900a
2013-05-11 13:55:41.359803 00:05:73:f5:d6:27 -> 01:00:0c:cc:cc:cd STP RST.
Root = 33768/00:05:73:ce:a9:7c Cost = 1 Port = 0x900a
2013-05-11 13:55:42.372405 00:05:73:ce:a9:46 -> 01:00:0c:cc:cc:cd STP RST.
Root = 33768/00:05:73:ce:a9:7c Cost = 0 Port = 0x900a
```

A fim ver pacotes detalhados, use o comando detail:

```
Nexus-5000# ethalyzer local interface inbound-hi detail display-filter "vlan.id == 1 && stp"
```

Capturing on eth4

Frame 7 (68 bytes on wire, 68 bytes captured)

```
Arrival Time: May 11, 2013 13:57:02.382227000
[Time delta from previous captured frame: 0.000084000 seconds]
[Time delta from previous displayed frame: 1368280622.382227000 seconds]
[Time since reference or first frame: 1368280622.382227000 seconds]
Frame Number: 7
Frame Length: 68 bytes
Capture Length: 68 bytes
[Frame is marked: False]
[Protocols in frame: eth:vlan:llc:stp]
```

Ethernet II, Src: 00:05:73:ce:a9:46 (00:05:73:ce:a9:46), Dst: 01:00:0c:cc:cc:cd

```

(01:00:0c:cc:cc:cd)
  Destination: 01:00:0c:cc:cc:cd (01:00:0c:cc:cc:cd)
    Address: 01:00:0c:cc:cc:cd (01:00:0c:cc:cc:cd)
      .... .1. .... = IG bit: Group address (multicast/broadcast)
      .... .0. .... = LG bit: Globally unique address
(factory default)
  Source: 00:05:73:ce:a9:46 (00:05:73:ce:a9:46)
    Address: 00:05:73:ce:a9:46 (00:05:73:ce:a9:46)
      .... .0. .... = IG bit: Individual address (unicast)
      .... .0. .... = LG bit: Globally unique address
(factory default)
  Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN
  111. .... = Priority: 7
  ...0 .... = CFI: 0
  .... 0000 0000 0001 = ID: 1
  Length: 50
Logical-Link Control
  DSAP: SNAP (0xaa)
  IG Bit: Individual
  SSAP: SNAP (0xaa)
  CR Bit: Command
  Control field: U, func=UI (0x03)
    000. 00.. = Command: Unnumbered Information (0x00)
    .... .11 = Frame type: Unnumbered frame (0x03)
  Organization Code: Cisco (0x00000c)
  PID: PVSTP+ (0x010b)
Spanning Tree Protocol
  Protocol Identifier: Spanning Tree Protocol (0x0000)
  Protocol Version Identifier: Rapid Spanning Tree (2)
  BPDU Type: Rapid/Multiple Spanning Tree (0x02)
  BPDU flags: 0x3c (Forwarding, Learning, Port Role: Designated)
    0... .... = Topology Change Acknowledgment: No
    .0.. .... = Agreement: No
    ..1. .... = Forwarding: Yes
    ...1 .... = Learning: Yes
    .... 11.. = Port Role: Designated (3)
    .... .0. = Proposal: No
    .... ...0 = Topology Change: No
  Root Identifier: 33768 / 00:05:73:ce:a9:7c
  Root Path Cost: 0
  Bridge Identifier: 33768 / 00:05:73:ce:a9:7c
  Port identifier: 0x900a
  Message Age: 0
  Max Age: 20
  Hello Time: 2
  Forward Delay: 15
  Version 1 Length: 0

```

A fim redigir esta informação a um arquivo PCAP, use este comando:

```

Nexus-5000# ethanalyzer local interface inbound-hi display-filter
"vlan.id == 1 && stp" write bootflash:bpdu.pcap
Capturing on eth4
3 << Lists how many packets were captured.

```

Em captções BPDU, o endereço MAC de origem é o endereço MAC de interface do dispositivo de extremidade oposta.

Na captação de Ethanalyzer, a porta aparece em um formato hexadecimal. A fim identificar o número de porta, você precisa de converter primeiramente o número no hexadecimal:

0x900a (do traço precedente) = 36874

Este é o comando que descodifica esse número a uma porta:

```
Nexus-5000# show spanning-tree internal info all |  
grep -b 50 "port_id 36874" | grep "Port Info"  
----- STP Port Info (vdc 1, tree 1, port Pol1) -----  
----- STP Port Info (vdc 1, tree 300, port Pol1) -----  
----- STP Port Info (vdc 1, tree 800, port Pol1) -----  
----- STP Port Info (vdc 1, tree 801, port Pol1) -----  
----- STP Port Info (vdc 1, tree 802, port Pol1) -----  
----- STP Port Info (vdc 1, tree 803, port Pol1) -----  
----- STP Port Info (vdc 1, tree 999, port Pol1) -----
```

Neste caso, é o canal de porta 11.

Convergência de STP

Se você precisa de investigar a convergência de STP, use o comando **interno das interações da medir-árvore da mostra**. Este comando fornece a introspecção em que eventos provocaram as mudanças STP. É importante recolher esta informação assim que a edição ocorrer, porque os logs são grandes, e envolvem ao longo do tempo.

```
Nexus-5000#show spanning-tree internal interactions  
- Event:(null), length:123, at 81332 usecs after Sat May 11 12:01:47 2013  
Success: pixm_send_set_mult_cbl_vlans_for_multiple_ports, num ports 1  
VDC 1, state FWD, rr_token 0x21b9c3 msg_size 584  
- Event:(null), length:140, at 81209 usecs after Sat May 11 12:01:47 2013  
vb_vlan_shim_set_vlans_multi_port_state(2733): Req (type=12) to PIXM  
vdc 1, inst 0, num ports 1, state FWD  
[Pol7 v 800-803,999-1000]  
- Event:(null), length:123, at 779644 usecs after Sat May 11 12:01:46 2013  
Success: pixm_send_set_mult_cbl_vlans_for_multiple_ports, num ports 1  
VDC 1, state FWD, rr_token 0x21b99a msg_size 544<  
- Event:(null), length:127, at 779511 usecs after Sat May 11 12:01:46 2013  
vb_vlan_shim_set_vlans_multi_port_state(2733): Req (type=12) to PIXM  
vdc 1, inst 0, num ports 1, state FWD  
[Pol7 v 300]  
- Event:(null), length:123, at 159142 usecs after Sat May 11 12:01:32 2013  
Success: pixm_send_set_mult_cbl_vlans_for_multiple_ports, num ports 1  
VDC 1, state LRN, rr_token 0x21b832 msg_size 584  
- Event:(null), length:140, at 159023 usecs after Sat May 11 12:01:32 2013  
vb_vlan_shim_set_vlans_multi_port_state(2733): Req (type=12) to PIXM  
vdc 1, inst 0, num ports 1, state LRN  
[Pol7 v 800-803,999-1000]  
- Event:(null), length:123, at 858895 usecs after Sat May 11 12:01:31 2013  
Success: pixm_send_set_mult_cbl_vlans_for_multiple_ports, num ports 1  
VDC 1, state LRN, rr_token 0x21b80b msg_size 544  
- Event:(null), length:127, at 858772 usecs after Sat May 11 12:01:31 2013  
vb_vlan_shim_set_vlans_multi_port_state(2733): Req (type=12) to PIXM  
vdc 1, inst 0, num ports 1, state LRN  
[Pol7 v 300]  
..... cut to save space .....
```

Mapeamento externo VLAN

Os 5000 Series Switch do nexa usam VLAN internos a fim traçar aos números de VLAN externos para enviar. Às vezes o ID de VLAN é a identificação interna VLAN a fim obter o mapeamento a um VLAN externo, entra:

```
Nexus-5000# show platform afm info global
```

```

Gatos Hardware version 0
Hardware instance mapping
-----
Hardware instance: 0 asic id: 0 slot num: 0
----- cut to save space -----
Hardware instance: 12 asic id: 1 slot num: 3
AFM Internal Status
-----
[unknown label ]: 324
[no free statistics counter ]: 2
[number of verify ]: 70
[number of commit ]: 70
[number of request ]: 785
[tcam stats full ]: 2

Vlan mapping table
-----
Ext-vlan: 1 - Int-vlan: 65

```

O STP debuga

Uma outra maneira de pesquisar defeitos edições STP é usar-se debuga. Contudo, o uso do STP debuga pôde fazer com que o USO de CPU crave, que causa interesses em alguns ambientes. A fim reduzir drasticamente o USO de CPU ao ser executado debuga, use um debugar-filtro, e registre a atividade a um arquivo de registro.

1. Crie o arquivo de registro, que salvar sob o log do diretório.

```

Nexus-5000#debug logfile spanning-tree.txt
Nexus-5548P-L3# dir log:
31 Nov 06 12:46:35 2012 dmesg
----- cut to save space-----
7626 Nov 08 22:41:58 2012 messages
0 Nov 08 23:05:40 2012 spanning-tree.txt
4194304 Nov 08 22:39:05 2012 startupdebug

```

2. Execute debugar.

```

Nexus-5000# debug spanning-tree bpdurx interface e1/30
<<<setup your spanning-tree for bpdus
Nexus-5000# copy log:spanning-tree.txt bootflash:

```

Exemplo do arquivo de registro:

```

Nexus-5000# debug spanning-tree bpdurx interface e1/30
<<<setup your spanning-tree for bpdus
Nexus-5000# copy log:spanning-tree.txt bootflash:

```

O nexo 5000 não processou BPDU

A fim pesquisar defeitos este problema, verifique a história do evento para determinar se o 5000 Series Switch do nexo supôs a raiz. O nexo 5000 supõe a raiz se ou não processou BPDU ou não os recebeu. A fim investigar que é a causa, você deve determinar se há o outro Switches anexado ao bridge designada que teve este problema também. Se nenhuma outra ponte teve o problema, é mais provável que o nexo 5000 não processou os BPDU. Se outras pontes tiveram o problema, é mais provável que a ponte não enviou os BPDU.

Nota: Coisas a manter-se na mente ao pesquisar defeitos STP e canal da porta virtual (vPC). Somente o vPC preliminar envia BPDU. Quando o vPC secundário é a raiz STP, o preliminar ainda envia os BPDU. Se a raiz é conectada através de um vPC, simplesmente

os contadores preliminares dos incrementos RX BPDU, mesmo quando o secundário os recebe.