

Dépannage du protocole Spanning Tree sur un commutateur de la gamme Nexus 5000

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Dépannage](#)

[Racine STP](#)

[Interface STP](#)

[Enquête BPDU avec Ethalyzer](#)

[Convergence STP](#)

[Mappage de VLAN externe](#)

[Débogues STP](#)

[Nexus 5000 n'a pas traité les BPDU](#)

Introduction

Ce document décrit diverses méthodes de dépannage des problèmes courants liés au protocole STP (Spanning Tree Protocol).

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- CLI du système d'exploitation Nexus
- STP

Components Used

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

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. If your network is live, make sure that you understand the potential impact of any command.

Dépannage

Cette section traite de certaines méthodes de dépannage des problèmes courants avec STP.

Racine STP

Afin de dépanner un problème STP, il est essentiel de savoir quel commutateur est actuellement la racine. La commande permettant d'afficher la racine STP sur un commutateur de la gamme Nexus 5000 est la suivante :

```
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
```

Voici quelques autres commandes pertinentes :

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

Une fois que vous avez déterminé qui est la racine actuelle, vous pouvez vérifier l'historique des événements pour voir s'il a changé et d'où proviennent les notifications de modification de topologie.

```
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 Po1]
2012:11:06 13h:44m:21s:515129us T_EV_M_FLUSH_L
VLAN0001 [1001.001b.54c2.5a42 C 6 A 5 R Po1 P none]
2012:11:06 13h:44m:23s:544632us T_EV_M_FLUSH_R
VLAN0001 [1001.001b.54c2.5a42 C 6 A 5 R Po1 P Po1]
2012:11:06 13h:44m:24s:510352us T_EV_M_FLUSH_R
VLAN0001 [1001.001b.54c2.5a42 C 6 A 5 R Po1 P Po1]
```

Astuce : Voici quelques définitions d'acronymes qui apparaissent dans le résultat des commandes. **SBPDU** : Unité de données de protocole de pont supérieure reçue ; **FLUSH_L** : Vidage local ; **FLUSH_R** : Vider du commutateur distant.

Note: Les versions de NX-OS antérieures à la version 5.1(3)N1(1) ne consignent pas plus de 149 événements et le journal ne s'affiche pas.

Interface STP

Cette commande est utilisée afin d'afficher les événements d'une interface.

```
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]
```

Cette commande est utilisée afin d'étudier les modifications STP sur une interface. Cette sortie offre de nombreux détails :

```
Nexus-5000# show spanning-tree internal info tree 1 interface port-channel 11
----- STP Port Info (vdc 1, tree 1, port Poll) -----
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 .....
```

Enquête BPDU avec Ethalyzer

Cette section décrit comment utiliser Ethalyzer afin de capturer des 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
```

Afin de voir les paquets détaillés, utilisez la commande **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

```

Afin d'écrire ces informations dans un fichier PCAP, utilisez cette commande :

```

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.

Sur les captures BPDU, l'adresse MAC source est l'adresse MAC de l'interface du périphérique final.

Sur la capture Ethalyzer, le port apparaît au format hexadécimal. Afin d'identifier le numéro de port, vous devez d'abord convertir le nombre en nombre hexadécimal :

0x900a (à partir de la trace précédente) = 36874

Il s'agit de la commande qui décode ce numéro vers un port :

```
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 Poll) -----
----- STP Port Info (vdc 1, tree 300, port Poll) -----
----- STP Port Info (vdc 1, tree 800, port Poll) -----
----- STP Port Info (vdc 1, tree 801, port Poll) -----
----- STP Port Info (vdc 1, tree 802, port Poll) -----
----- STP Port Info (vdc 1, tree 803, port Poll) -----
----- STP Port Info (vdc 1, tree 999, port Poll) -----
```

Dans ce cas, il s'agit du port-channel 11.

Convergence STP

Si vous devez étudier la convergence STP, utilisez la commande **show spanning-tree internal interactions**. Cette commande fournit des informations sur les événements qui ont déclenché les modifications STP. Il est important de recueillir ces informations dès que le problème se produit, car les journaux sont volumineux et ils s'enroulent au fil du temps.

```
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
[Po17 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
[Po17 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
[Po17 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
[Po17 v 300]
..... cut to save space .....
```

Mappage de VLAN externe

Les commutateurs de la gamme Nexus 5000 utilisent des VLAN internes afin de mapper les numéros de VLAN externes pour le transfert. Parfois, l'ID de VLAN est l'ID de VLAN interne. Afin d'obtenir le mappage à un VLAN externe, saisissez :

```
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
```

Débogues STP

Une autre façon de dépanner les problèmes STP est d'utiliser les débogages. Cependant, l'utilisation de débogages STP peut provoquer une augmentation de l'utilisation du CPU, ce qui entraîne des problèmes dans certains environnements. Afin de réduire considérablement l'utilisation du CPU lors de l'exécution de débogages, utilisez un filtre de débogage et l'activité de journal dans un fichier journal.

1. Créez le fichier journal, qui est enregistré dans le journal du répertoire.

```
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. Exécutez le débogage.

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

Exemple du fichier journal :

```
2012 Nov 8 23:08:24.238953 stp: BPDU RX: vb 1 vlan 300, ifi 0x1a01d000
(Ethernet1/30)
2012 Nov 8 23:08:24.239095 stp: BPDU Rx: Received BPDU on vb 1 vlan 300
port Ethernet1/30 pkt_len 60 bpdu_len 36 netstack flags 0x00ed enc_type ieee
2012 Nov 8 23:08:35.968453 stp: RSTP(300): Ethernet1/30 superior msg
2012 Nov 8 23:08:35.968466 stp: RSTP(300): Ethernet1/30 rcvd info remaining 6
2012 Nov 8 23:08:36.928415 stp: BPDU RX: vb 1 vlan 300, ifi 0x1a01d000
(Ethernet1/30)
2012 Nov 8 23:08:36.928437 stp: BPDU Rx: Received BPDU on vb 1 vlan 300
port Ethernet1/30 pkt_len 60 bpdu_len 36 netstack flags 0x00ed enc_type ieee
2012 Nov 8 23:08:36.928476 stp: RSTP(300): msg on Ethernet1/30
..... cut to save space .....
```

Nexus 5000 n'a pas traité les BPDU

Afin de résoudre ce problème, consultez l'historique des événements pour déterminer si le commutateur Nexus 5000 a pris racine. Le Nexus 5000 prend racine s'il ne traite pas les BPDU ou s'il ne les reçoit pas. Afin d'en déterminer la cause, vous devez déterminer si d'autres commutateurs sont reliés au pont désigné qui a également rencontré ce problème. Si aucun autre pont n'a rencontré le problème, il est probable que le Nexus 5000 n'a pas traité les BPDU. Si d'autres ponts ont rencontré le problème, il est probable que le pont n'a pas envoyé les BPDU.

Note: Points à garder à l'esprit lors du dépannage de STP et de Virtual Port Channel (vPC). Seul le vPC Primary envoie des BPDU. Lorsque le vPC secondaire est la racine STP, le primaire envoie toujours les BPDU. Si la racine est connectée via un vPC, seul le principal incrémente les compteurs BPDU Rx, même lorsque le secondaire les reçoit.