排除Catalyst交換機上的STP故障

目錄

簡介

背景資訊

必要條件

- 需求
- 採用元件

慣例

STP故障的原因

排除轉發環路故障

- 1.識別環路
- 2.發現環路的拓撲（範圍）
- 3.打破循環
- 4.查詢並修復環路的原因
- 5.還原備援

調查拓撲更改

- 找出泛濫的原因
- 查詢TC的來源
- 採取措施防止過多的TC

排除與收斂時間相關的問題

使用STP Debug命令

保護網路免受轉發環路的影響

- 1.在所有交換機到交換機鏈路上啟用單向鏈路檢測(UDLD)
- 2.在所有交換機上啟用環路防護
- 3.在所有終端站連線埠上啟用Portfast
- 4.在雙側（支援時）和Non-SilentOption上將EtherChannel設定為DesirableMode
- 5.請勿在交換器到交換器連結上停用自動交涉（如支援）
- 6.調整STP計時器時請務必小心
- 7.如果可能存在拒絕服務攻擊，請使用根防護來保護網路STP邊界
- 8.在已啟用Portfast的埠上啟用BPDU防護，以防止STP受到連線到埠的未授權網路裝置（如集線器、交換機和橋接路由器）的影響
- 9.避免管理VLAN上的使用者流量
- 10.可預測的（硬編碼）STP根和備份STP根位置

相關資訊

簡介

本文說明如何使用Cisco IOS®軟體來排解跨距樹狀目錄通訊協定(STP)的問題。

背景資訊

有些特定命令僅適用於Catalyst 6500/6000；但是，您可以將大多數原則應用於運行Cisco IOS軟體的任何Cisco Catalyst交換機。

大多數STP的問題都有以下三個問題：

- 轉發環路。
- 由於STP拓撲更改率(TC)高而導致泛洪過多。
- 與收斂時間相關的問題。

由於網橋沒有跟蹤某個資料包是否多次轉發的機制（例如，IP生存時間[TTL]），因此它用於丟棄網路中循環時間過長的流量。同一第2層(L2)域中的兩台裝置之間只能存在一條路徑。

STP的目的是根據STP演算法阻塞冗餘埠，並將冗餘物理拓撲解析為樹狀拓撲。當冗餘拓撲中沒有埠被阻塞時，會發生轉發環路（如STP環路），流量會無限期地循環轉發。

一旦轉送環路開始，它就會在其路徑上擁塞頻寬最低的鏈路。如果所有鏈路的頻寬相同，則所有鏈路都會擁塞。這種擁塞會導致資料包丟失，並在受影響的L2域中導致網路癱瘓。

如果發生過度泛濫，症狀就不會那麼明顯。慢速鏈路可能會因泛洪流量而擁塞，這些擁塞鏈路背後的裝置或使用者可能會遇到速度緩慢或完全失去連線的情況。

必要條件

需求

思科建議您瞭解以下主題：

- 各種生成樹型別以及如何配置它們。有關詳細資訊，請參閱配置STP和IEEE 802.1s。
- 各種生成樹功能以及如何配置它們。有關詳細資訊，請參閱配置STP功能。

採用元件

本文中的資訊係根據以下軟體和硬體版本：

- 採用Supervisor 2引擎的Catalyst 6500
- Cisco IOS 軟體版本 12.1(13)E

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

慣例

如需更多文件慣例的相關資訊，請參閱思科技術提示慣例。

STP故障的原因

STP對其操作環境做出某些假設。以下為與本檔案最相關的假設：

- 兩個網橋之間的每條鏈路都是雙向的。這意味著，如果A直接連線到B，則A接收B已傳送的內容，而B接收A已傳送的內容，只要它們之間的鏈路處於接通狀態。
- 每個運行STP的網橋都能夠定期接收、處理和傳輸STP網橋協定資料單元(BPDU)，也稱為STP資料包。

雖然這些假設似乎是符合邏輯且顯而易見的，但有些情況下它們並未得到滿足。其中大多數情況都涉及到一種硬體問題；但是，軟體缺陷也可能導致STP故障。各種硬體故障、配置錯誤、連線問題導致大部分STP故障，而軟體故障佔少數。由於交換機之間存在不必要的其他連線，也可能發生STP故障。由於這些額外的連線，VLAN會進入關閉狀態。要解決此問題，請刪除交換機之間所有不需要的連線。

當其中一個假設未滿足時，一個或多個網橋將無法接收或處理BPDU。這表示網橋（或多個網橋）沒有發現網路拓撲。如果不知道正確的拓撲，交換機將無法阻塞環路。因此，泛洪流量通過環路拓撲循環，消耗所有頻寬，並中斷網路。

交換機無法接收BPDU的原因示例包括收發器或Gigabit介面轉換器(GBIC)故障、電纜問題或埠、線卡或Supervisor引擎上的硬體故障。STP故障的一個常見原因是網橋之間的單向鏈路。在這種情況下，一個網橋會傳送BPDU，但下游網橋永遠不會收到它們。由於交換機無法處理收到的BPDU，STP處理也可能被超載CPU（99%或更多）中斷。BPDU可能會在從一個網橋到另一個網橋的路徑中損壞，這也阻止了正確的STP行為。

除了轉發環路之外，當沒有埠被阻塞時，也存在只有某些資料包通過阻塞流量的埠被錯誤轉發的情況。在大多數情況下，這是軟體問題造成的。這種行為可能導致「慢循環」。 這表示某些封包已回圈，但大部分流量仍流經網路，因為連結沒有擁塞。

排除轉發環路故障

轉發環路的來源（原因）和效果有很大不同。由於存在各種可能影響STP的問題，本文檔僅提供有關如何排除轉發環路故障的一般指南。

開始疑難排解之前，您需要以下資訊：

- 詳細介紹所有交換機和網橋的實際拓撲圖。
- 它們對應的埠號（互連）。
- STP配置詳細資訊，例如哪台交換機是根和備份根，哪些鏈路具有非預設成本或優先順序，以及阻塞流量的埠的位置。

1.識別環路

當網路中出現轉發環路時，通常的症狀為：

- 與受影響網路區域之間、與受影響網路區域之間以及經過受影響網路區域之間失去連線。
- 連線到受影響網段或VLAN的路由器上的CPU使用率高，這可能會導致各種症狀，例如路由協定鄰居抖動或熱備份路由器協定(HSRP)活動路由器抖動。

- 鏈路利用率高（通常為100%）。
- 高交換機背板利用率（與基線利用率相比）。
- 指示資料包在網路中循環的系統日誌消息（例如HSRP重複IP地址消息）。
- 指示持續地址重新學習或MAC地址擺動消息的系統日誌消息。
- 許多介面上的輸出丟棄數量增加。

其中任何一個原因都可以指示不同的問題（或者根本不存在問題）。但是，如果同時觀察到其中的許多情況，則很可能網路中已經形成轉發環路。驗證此情況的最快方法是檢查交換器背板流量利用率：

```
<#root>

cat#

show catalyst6000 traffic-meter



traffic meter =   13%

  Never cleared

peak =   14%

  reached at 12:08:57 CET Fri Oct 4 2002
```

註：搭載Cisco IOS軟體的Catalyst 4000目前不支援此命令。

如果當前流量級別過高，或者基線級別未知，請檢查峰值級別最近是否達到，以及它是否接近當前流量級別。例如，如果峰值流量級別為15%，並且在兩分鐘前到達該值，而當前流量級別為14%，則意味著交換機具有異常高的負載。如果流量負載處於正常水準，則可能意味著沒有環路或此裝置未參與環路。但是，它仍可能處於慢循環中。

2.發現環路的拓撲（範圍）

一旦確定網路中斷的原因是轉發環路，則最高優先順序是停止環路並恢復網路運行。

要停止環路，您必須知道哪些埠參與了環路：請檢視具有最高鏈路利用率（每秒資料包數）的埠。show interfaceCisco IOS軟體命令顯示每個介面的利用率。

要僅顯示利用率資訊和介面名稱（以便快速分析），請使用Cisco IOS軟體過濾正規表示式輸出。發出show interface | include line|\/seccommand僅顯示每秒資料包統計資訊和介面名稱：

```
<#root>
```

```
cat#
show interface | include line|\/sec


GigabitEthernet2/1 is up, line protocol is down
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet2/2 is up, line protocol is down
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet2/3 is up, line protocol is up
  5 minute input rate 99765230 bits/sec, 24912 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet2/4 is up, line protocol is up
  5 minute input rate 1000 bits/sec, 27 packets/sec


 5 minute output rate 101002134 bits/sec, 25043 packets/sec
GigabitEthernet2/5 is administratively down, line protocol is down
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet2/6 is administratively down, line protocol is down
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet2/7 is up, line protocol is down
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet2/8 is up, line protocol is up
  5 minute input rate 2000 bits/sec, 41 packets/sec


5 minute output rate 99552940 bits/sec, 24892 packets/sec
```

請注意鏈路利用率最高的介面。在本例中，這些埠是介面g2/3、g2/4和g2/8；它們是參與環路的埠。

3.打破循環

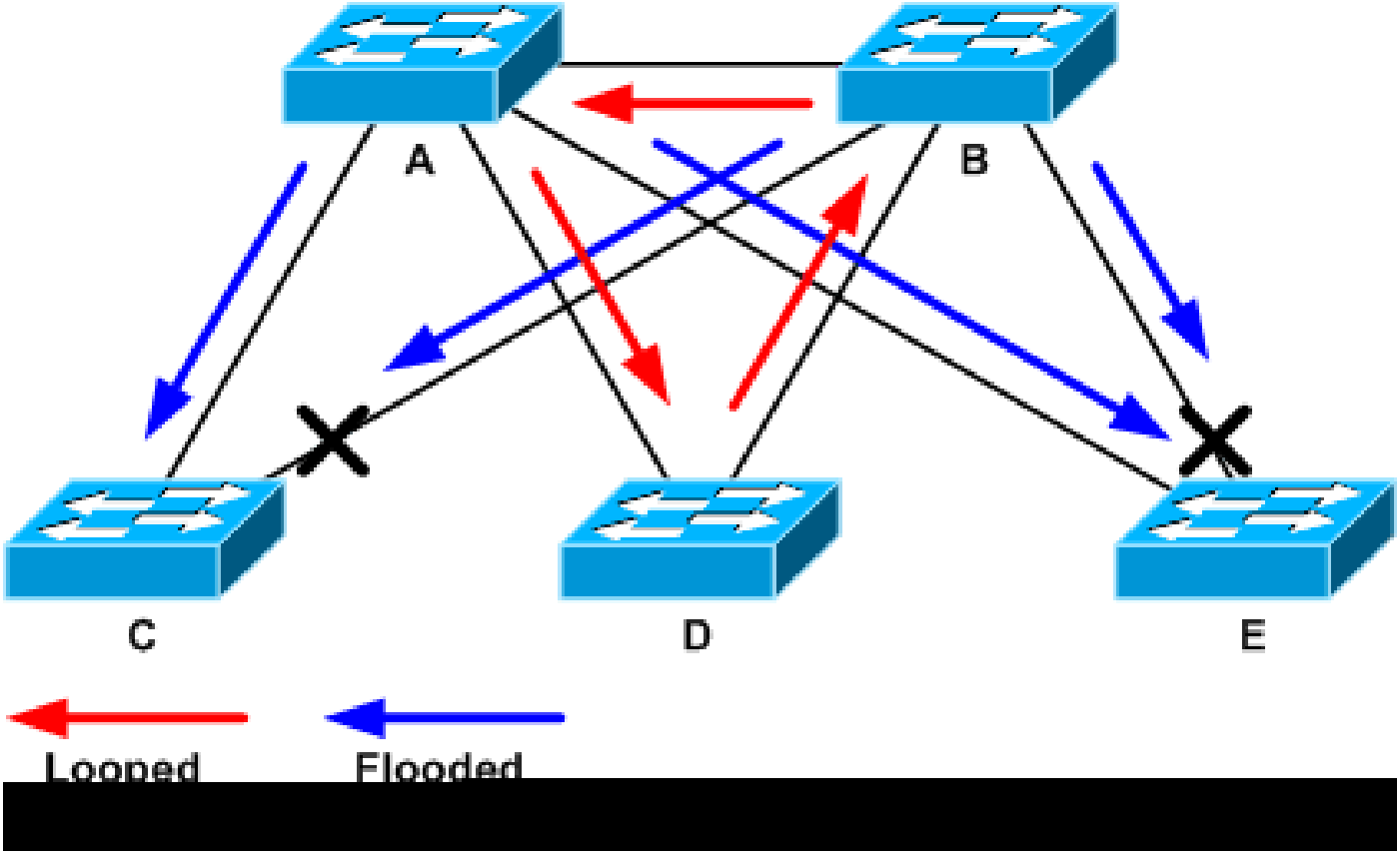
若要中斷回圈，您必須關閉或斷開相關的連線埠。不僅要停止環路，而且要找到並修復環路的根本原因，這一點特別重要。打破這個循環相對比較容易

註：不必同時關閉或斷開所有埠。你可以一次關閉一個裝置。最好關閉受環路影響的匯聚點的埠，如分佈層或核心層交換機。如果一次關閉所有埠，並逐一啟用或重新連線這些埠，則不起作用；環路會停止，並且在故障埠重新連線後無法立即啟動。因此，很難將故障與任何特定埠相關聯。

註：為了中斷循環，建議在重新啟動交換機之前收集資訊。否則，後續根本原因分析將非常困難。 禁用或斷開每個埠後，必須檢查交換機背板利用率是否恢復到正常水準。

註：請記住，埠不維持環路，而是泛洪以環路到達的流量。關閉此類泛洪埠時，只會減少少量背板利用率，而不會停止環路。

在下一個示例拓撲中，環路位於交換機A、B和D之間。因此，連結AB、AD和BD是持續的。如果關閉這些鏈路中的任何一個，就會停止環路。鏈路AC、AE、BC和BE僅泛洪通過環路到達的流量。



環路和泛洪流量

在關閉支援埠後，底板利用率降至正常值。您需要知道哪個埠的關閉使背板利用率（和其他埠的利用率）達到正常水準。

此時，環路停止，網路操作改善；但是，由於環路的原始原因尚未修復，仍然存在其他問題。

4.查詢並修復環路的原因

一旦循環停止，就需要確定循環開始的原因。這是整個過程的難點，因為原因可能各不相同。同樣難以正式確定在每種情況下都有效的確切程式。

指南:

- 檢查拓撲圖以查詢冗餘路徑。其中包括上一步中找到的返回到同一交換機的支援埠（在環路期間交換的路徑資料包）。在上一個示例拓撲中，此路徑為AD-DB-BA。
- 對於冗餘路徑上的每台交換機，檢查交換機是否知道正確的STP根。

L2網路中的所有交換機必須在通用STP根上達成一致。當網橋始終為特定VLAN或STP例項中的STP根顯示不同的ID時，這是問題的明顯症狀。發出show spanning-tree vlan vlan-id命令以顯示給

定VLAN的根網橋ID:

```
<#root>

cat#

show spanning-tree vlan 333


MST03
  Spanning tree enabled protocol mstp


Root ID    Priority    32771
             Address     0050.14bb.6000

             Cost        20000
             Port        136 (GigabitEthernet3/8)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32771  (priority 32768 sys-id-ext 3)
             Address     00d0.003f.8800
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface        Role Sts Cost      Prio.Nbr Status
---------------- ---- --- --------- -------- ------------------------
Gi3/8            Root FWD 20000     128.136  P2p
Po1              Desg FWD 20000     128.833  P2p
```

從埠可以找到VLAN號，因為環路中涉及的埠是在前面的步驟中建立的。如果涉及的埠是中繼，則通常涉及中繼上的所有VLAN。如果情況並非如此（例如，如果回圈似乎發生在一個VLAN上），則可以嘗試發出這些介面 | 包括L2|line|broadcastcommand（僅限Catalyst 6500/6000系列交換機上的Supervisor 2和更高版本引擎，因為Supervisor 1不提供每VLAN交換統計資訊）。請僅檢視VLAN介面。交換封包數量最多的VLAN通常會發生回圈：

```
<#root>

cat#

show interface | include L2|line|broadcast


Vlan1 is up, line protocol is up
  L2 Switched: ucast: 653704527 pkt, 124614363025 bytes - mcast:
         23036247 pkt, 1748707536 bytes
     Received 23201637 broadcasts, 0 runts, 0 giants, 0 throttles

Vlan10 is up, line protocol is up
  L2 Switched: ucast: 2510912 pkt, 137067402 bytes - mcast:
         41608705 pkt, 1931758378 bytes
     Received 1321246 broadcasts, 0 runts, 0 giants, 0 throttles
Vlan11 is up, line protocol is up
  L2 Switched: ucast: 73125 pkt, 2242976 bytes - mcast:
         3191097 pkt, 173652249 bytes
     Received 1440503 broadcasts, 0 runts, 0 giants, 0 throttles
Vlan100 is up, line protocol is up
```

```
  L2 Switched: ucast: 458110 pkt, 21858256 bytes - mcast:
        64534391 pkt, 2977052824 bytes
     Received 1176671 broadcasts, 0 runts, 0 giants, 0 throttles
Vlan101 is up, line protocol is up
  L2 Switched: ucast: 70649 pkt, 2124024 bytes - mcast:
        2175964 pkt, 108413700 bytes
     Received 1104890 broadcasts, 0 runts, 0 giants, 0 throttles
```

在本例中，VLAN 1佔用的廣播和L2交換流量最多。 確保正確識別根埠。

根埠到根網橋的成本必須最低（有時一條路徑的跳數會更短，但成本會更長，因為低速埠的成本較高）。 要確定哪個埠被視為給定VLAN的根，請發出show spanning-tree vlan 命令：

<#root>

```
cat#
```

show spanning-tree vlan 333

```
MST03
  Spanning tree enabled protocol mstp
  Root ID    Priority    32771
             Address     0050.14bb.6000
             Cost        20000
```

Port   136 (GigabitEthernet3/8)

```
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32771  (priority 32768 sys-id-ext 3)
             Address     00d0.003f.8800
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface        Role Sts Cost      Prio.Nbr Status
---------------- ---- --- --------- -------- ------------------------
Gi3/8            Root FWD 20000     128.136  P2p
Po1              Desg FWD 20000     128.833  P2p
```

確保在根埠和應該阻塞的埠上定期接收BPDU。

BPDU由根網橋按每個螺旋間隔（預設情況下為兩秒）傳送。非根網橋接收、處理、修改和傳播從根接收的BPDU。 發出show spanning-tree interface detail命令以確認是否收到BPDU:

<#root>

```
cat#
```

show spanning-tree interface g3/2 detail

```
Port 130 (GigabitEthernet3/2) of MST00 is backup blocking
```

```
   Port path cost 20000, Port priority 128, Port Identifier 128.130.
   Designated root has priority 0, address 0007.4f1c.e847
   Designated bridge has priority 32768, address 00d0.003f.8800
   Designated port id is 128.129, designated path cost 2000019
   Timers: message age 4, forward delay 0, hold 0
```

Number of transitions to forwarding state: 0

```
   Link type is point-to-point by default, Internal
   Loop guard is enabled by default on the port
   BPDU: sent 3,
```

received 53

```
cat#
```

show spanning-tree interface g3/2 detail

```
Port 130 (GigabitEthernet3/2) of MST00 is backup blocking
   Port path cost 20000, Port priority 128, Port Identifier 128.130.
   Designated root has priority 0, address 0007.4f1c.e847
   Designated bridge has priority 32768, address 00d0.003f.8800
   Designated port id is 128.129, designated path cost 2000019
   Timers: message age 5, forward delay 0, hold 0
   Number of transitions to forwarding state: 0
   Link type is point-to-point by default, Internal
   Loop guard is enabled by default on the port
   BPDU: sent 3,
```

received 54

註：在命令的兩個輸出之間收到一個BPDU（計數器從53變為54）。

顯示的計數器實際上是STP進程本身維護的計數器。這意味著如果接收計數器增加，不僅物理埠接收了BPDU，STP進程也會接收它。 如果 received 在應該作為根備用或備用埠的連線埠上，BPDU計數器不會遞增，然後檢查連線埠是否收到任何多點傳送（BPDU以多點傳送方式傳送）。發出show interface interface counters命令：

<#root>

```
cat#
```

show interface g3/2 counters

```
Port            InOctets   InUcastPkts
```

InMcastPkts

```
   InBcastPkts
Gi3/2           14873036             2
```

89387

```
             0
```

```
Port           OutOctets  OutUcastPkts  OutMcastPkts  OutBcastPkts
Gi3/2          114365997         83776        732086            19

cat#
```

show interface g3/2 counters

```
Port            InOctets   InUcastPkts
```

InMcastPkts

```
   InBcastPkts
Gi3/2           14873677             2
```

89391

```
             0

Port           OutOctets  OutUcastPkts  OutMcastPkts  OutBcastPkts
Gi3/2          114366106         83776        732087            19
```

使用環路防護和BPDU遲滯檢測功能的生成樹協定增強功能的使用環路防護和BPDU遲滯檢測增強STP 一節中提供了有關STP埠角色的簡要說明。 如果未收到BPDU，請檢查埠是否對錯誤計數。發出show interface counters errorscommand:

<#root>

```
cat#
```

show interface g4/3 counters errors

```
Port     Align-Err    FCS-Err   Xmit-Err    Rcv-Err UnderSize OutDiscards
Gi4/3            0          0          0          0         0           0

Port  Single-Col Multi-Col  Late-Col Excess-Col Carri-Sen   Runts    Giants
Gi4/3          0         0         0          0         0        0         0
```

BPDU可能由物理埠接收，但仍未到達STP進程。如果前兩個示例中使用的命令顯示已收到某些多播且未計數錯誤，則檢查BPDU是否在STP進程級別被丟棄。在Catalyst 6500上發出remote命令switch test spanning-tree process-stats命令：

<#root>

```
cat#
```

remote command switch test spanning-tree process-stats

```
------------------TX STATS------------------
 transmission rate/sec     = 2
 paks transmitted          = 5011226
 paks transmitted (opt)    = 0
 opt chunk alloc failures  = 0
 max opt chunk allocated   = 0
```

```
------------------RX STATS------------------

receive rate/sec          = 1

 paks received at stp isr  = 3947627
 paks queued at stp isr    = 3947627


paks dropped at stp isr   = 0
 drop rate/sec             = 0

 paks dequeued at stp proc = 3947627
 paks waiting in queue     = 0
 queue depth               = 7(max) 12288(total)
--------------PROCESSING STATS---------------
 queue wait time (in ms)   = 0(avg) 540(max)
 processing time (in ms)   = 0(avg) 4(max)
 proc switch count         = 100
 add vlan ports            = 20
 time since last clearing  = 2087269 sec
```

本示例中使用的命令顯示STP進程統計資訊。檢驗丟棄計數器是否不增加以及接收資料包是否增加非常重要。 如果收到的資料包沒有增加，但物理埠確實接收了組播，請驗證交換機帶內介面（CPU介面）是否接收了資料包。發出remote command switch show ibc | Catalyst 6500/6000上的rx_inputcommand:

```
<#root>

cat#

remote command switch show ibc | i rx_input


rx_inputs=

5626468

, rx_cumbytes=859971138

cat#

remote command switch show ibc | i rx_input


rx_inputs=

5626471

, rx_cumbytes=859971539
```

此範例顯示，在輸出之間，帶內連線埠已接收23個封包。

註：這23個資料包不僅是BPDU資料包；它是帶內埠接收的所有資料包的全域性計數器。

如果沒有在本地交換機或埠上丟棄BPDU的指示，您必須移至鏈路另一端的交換機，並驗證該交換

機是否傳送了BPDU。檢查BPDU是否定期在非根指定埠上傳送。如果埠角色同意，則埠會傳送BPDU，但鄰居不會收到它們。檢查是否傳送了BPDU。發出show spanning-tree interface interface detail命令：

```
<#root>

cat#

show spanning-tree interface g3/1 detail


Port 129 (GigabitEthernet3/1) of MST00 is

designated

 forwarding
   Port path cost 20000, Port priority 128, Port Identifier 128.129.
   Designated root has priority 0, address 0007.4f1c.e847
   Designated bridge has priority 32768, address 00d0.003f.8800
   Designated port id is 128.129, designated path cost 2000019
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 0
   Link type is point-to-point by default, Internal
   Loop guard is enabled by default on the port


BPDU: sent 1774

, received 1

cat#

show spanning-tree interface g3/1 detail


Port 129 (GigabitEthernet3/1) of MST00 is

designated

 forwarding
   Port path cost 20000, Port priority 128, Port Identifier 128.129.
   Designated root has priority 0, address 0007.4f1c.e847
   Designated bridge has priority 32768, address 00d0.003f.8800
   Designated port id is 128.129, designated path cost 2000019
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 0
   Link type is point-to-point by default, Internal
   Loop guard is enabled by default on the port


BPDU: sent 1776

, received 1
```

在本例中，輸出之間傳送兩個BPDU。

註:STP進程維護BPDU:sentcounter符。這表示計數器表示BPDU已傳送到物理埠並被傳送。檢查傳輸的多播資料包的埠計數器是否增加。發出show interface interface counters命令。這有

助於確定BPDU通訊流。

<#root>

cat#

show interface g3/1 counters

```
Port            InOctets    InUcastPkts    InMcastPkts    InBcastPkts
Gi3/1          127985312          83776         812319             19

Port           OutOctets   OutUcastPkts
```

OutMcastPkts

```
  OutBcastPkts
Gi3/1          131825915           3442
```

872342

```
           386
```

cat#

show interface g3/1 counters

```
Port            InOctets    InUcastPkts    InMcastPkts    InBcastPkts
Gi3/1          127985312          83776         812319             19

Port           OutOctets   OutUcastPkts
```

OutMcastPkts

```
  OutBcastPkts
Gi3/1          131826447           3442
```

872346

```
           386
```

通過所有這些步驟，目的是查詢未接收、傳送或處理BPDU的交換機或鏈路。 STP可能已經計算埠的正確狀態，但由於控制平面問題，它無法在轉發硬體上設定此狀態。如果埠在硬體級別未被阻塞，則可能會產生環路。如果您認為這是您網路中的問題，請聯絡思科技術支援以取得進一步的協助。

5.還原備援

找到導致環路的裝置或鏈路後，必須將此裝置與網路隔離，否則必須解決問題。（例如更換光纖或GBIC）。必須恢復步驟3中斷開的冗餘鏈路。

重要的是不要操作導致環路的裝置或鏈路，因為導致環路的許多情況都是瞬時、間歇性和不穩定的。這意味著，如果在調查中或調查後清除該情況，則這種情況暫時不會發生，或者根本沒有發生。必須記錄條件，以便思科技術支持可以進一步調查。在重設交換器之前，請收集有關情況的資訊非常重要。如果條件消失，則無法確定循環的根本原因。如果收集資訊，請確保此問題不會再次導致

環路。如需詳細資訊，請參閱保護網路免受轉送回圈的影響。

調查拓撲更改

拓撲更改(TC)機制的作用是在拓撲更改後糾正L2轉發表。這是避免連線中斷所必需的，因為以前可通過特定埠訪問的MAC地址可能會更改並通過不同埠訪問。TC縮短發生TC的VLAN中所有交換機的轉發表時間。因此，如果地址未重新獲知，則會出現超時和泛洪，以確保資料包到達目標MAC地址。

TC由埠的STP狀態更改為STPforwardingstate或從STPforwardingstate更改觸發。在TC之後，即使特定目標MAC地址已過期，泛洪也不會持續很長時間。此位址由來自MAC位址已過時的主機的第一個封包重新整理。當TC重複發生（時間間隔較短）時，可能會出現此問題。交換機不斷快速老化其轉發表，因此泛洪幾乎可以保持恆定。

註意：使用快速STP或多重STP（IEEE 802.1w和IEEE 802.1s），TC由埠狀態更改為轉發以及角色從designatedtoroot更改觸發。使用快速STP時，會立即刷新L2轉發表，而不是802.1d，從而縮短了老化時間。立即刷新轉發表可以更快地恢復連線，但可能會導致更多泛洪

在配置良好的網路中，TC是一個罕見的事件。當交換機埠上的鏈路接通或斷開時，一旦埠的STP狀態更改為轉發或從轉發狀態便最終會出現TC。當連線埠擺動時，會導致重複的TC和泛洪。

啟用STP portfast功能的埠在進入或離開轉發狀態時無法引起TC。在所有終端裝置埠（如印表機、PC和伺服器）上配置portfast可以將TC限制在低數量，強烈建議這樣做。

如果網路上存在重複的TC，您必須確定這些TC的來源，然後採取措施減少這些TC，從而最大限度地減少泛洪。

在802.1d中，有關TC事件的STP資訊通過TC通知(TCN)在網橋之間傳播，TCN是一種特殊型別的BPDU。如果您按照接收TCN BPDU的連線埠操作，可以找到產生TC的裝置。

找出泛濫的原因

您可以確定存在因效能低下導致的泛洪、鏈路上的資料包丟棄，以及資料包分析器顯示多個單播資料包到不在本地網段上的同一目標。 有關單播泛洪的詳細資訊，請參閱交換園區網路中的單播泛洪。

在執行Cisco IOS軟體的Catalyst 6500/6000上，您可以檢查轉送引擎計數器（僅位於Supervisor 2引擎上）以估計泛濫的量。發出remote command switch show earl statistics | i MISS_DA|ST_FRcommand:

```
<#root>

cat#

remote command switch show earl statistics | i MISS_DA|ST_FR

         ST_MISS_DA      =       18                      530308834
         ST_FRMS         =       97                      969084354
```

```
cat#
remote command switch show earl statistics | i MISS_DA|ST_FR

         ST_MISS_DA      =       4                       530308838
         ST_FRMS         =       23                      969084377
```

在此示例中，第一列顯示自上次執行此命令以來的更改，第二列顯示自上次重新啟動以來的累積值。第一行顯示泛洪幀的數量，第二行顯示已處理的幀的數量。如果兩個值接近在一起，或第一個值以高速增加，則可能是交換機正在泛洪流量。但是，由於計數器不是粒度的，因此只能與其他驗證泛洪的方法結合使用。每台交換機有一個計數器，而不是每個埠或VLAN。看到某些泛洪資料包是正常的，因為如果目標MAC地址不在轉發表中，交換機總是可以泛洪。當交換器收到目的地位址尚未得知的封包時，可能會發生這種情況。

查詢TC的來源

如果發生過度泛洪的VLAN的VLAN編號已知，請檢查STP計數器以檢視TC的數量是否高或定期增加。發出show spanning-tree vlan vlan-id 詳細命令（在本範例中使用的是VLAN 1）：

```
<#root>
cat#
show spanning-tree vlan 1 detail

VLAN0001 is executing the ieee compatible Spanning Tree protocol
  Bridge Identifier has priority 32768, sysid 1, address 0007.0e8f.04c0
  Configured hello time 2, max age 20, forward delay 15
  Current root has priority 0, address 0007.4f1c.e847
  Root port is 65 (GigabitEthernet2/1), cost of root path is 119
  Topology change flag not set, detected flag not set

Number of topology changes 1 last change occurred 00:00:35 ago
          from GigabitEthernet1/1

  Times:  hold 1, topology change 35, notification 2
          hello 2, max age 20, forward delay 15
  Timers: hello 0, topology change 0, notification 0, aging 300
```

如果VLAN編號未知，您可以使用資料包分析器或檢查所有VLAN的TC計數器。

採取措施防止過多的TC

您可以監控拓撲更改計數器的數量，以檢視它是否定期增加。然後，移動到與所示埠連線的網橋，以接收最後一個TC（在上一個示例中，為埠GigabitEthernet1/1），並檢視該網橋的TC來自何處。必須重複此過程，直到找到未啟用STP portfast的終端站埠，或者直到找到需要修復的擺動鏈路。如果TC來自其他來源，則需要重複整個程式。如果鏈路屬於終端主機，則可以配置portfast功能以防止

生成TC。

註：在Cisco IOS軟體STP實施中，如果VLAN中的埠接收了TCN BPDU，則TC的計數器只能遞增。如果接收到具有已設定TC標誌的正常配置BPDU，則TC計數器不遞增。這表示如果您懷疑TC是泛洪的原因，請開始從該VLAN中的STP根網橋中追蹤TC的來源。它可以獲得有關TC數量和來源的最準確的資訊。

排除與收斂時間相關的問題

在某些情況下，STP的實際操作與預期行為不匹配。以下是兩個最常見的問題：

- STP收斂或重新收斂所需的時間比預期要長。
- 拓撲結果不同於預期。

最常見的原因有以下幾點：

- 實際拓撲與記錄的拓撲不匹配。
- 配置錯誤，如STP計時器配置不一致、STP直徑增加或portfast配置錯誤。
- 收斂或重新收斂期間交換機CPU過載。
- 軟體缺陷。

如前所述，由於可能會影響STP的各種問題，本文檔只能提供故障排除的一般指南。 要瞭解為什麼收斂時間比預期要長，請檢視STP事件的順序，瞭解發生的情況和順序。由於Cisco IOS軟體中的STP實施不會記錄結果（埠不一致等特定事件除外），因此您可以使用Cisco IOS軟體對STP進行調試，以便更清楚地檢視。 對於STP，使用運行Cisco IOS軟體的Catalyst 6500/6000時，處理在交換機處理器(SP)（或Supervisor）上完成，因此需要在SP上啟用調試。對於Cisco IOS軟體橋接組，處理在路由處理器(RP)上完成，因此需要在RP(MSFC)上啟用調試。

使用STP Debug命令

許多STPdebug命令旨在用於開發工程。如果沒有詳細瞭解Cisco IOS軟體中的STP實施，它們不會提供任何對使用者有意義的輸出。某些調試可以提供可立即讀取的輸出，例如埠狀態更改、角色更改、事件（例如TC）以及已接收和已傳輸BPDU的轉儲。本部分並不提供所有調試程式的完整說明，而是簡要介紹最常用的調試程式。

注意：使用debug命令時，請啟用所需的最低調試。如果不需要即時調試，請將輸出記錄到日誌中，而不是列印到控制檯。過多的調試會使CPU過載並中斷交換機操作。

要將調試輸出定向到日誌，而不是定向到控制檯或Telnet會話，請在全域性配置模式下發出logging console informationaland no logging monitor命令。 要檢視常規事件日誌，請對每個VLAN生成樹(PVST)和快速PVST發出debug spanning-tree命令。這是提供有關STP發生情況的資訊的第一個調試。 在多生成樹(MST)模式下，發出debug spanning-tree命令不起作用。因此，發出debug

spanning-tree mstp rolescommand，以檢視埠角色更改。 要檢視埠STP狀態更改，請發出debug spanning-tree switch statecommand和debug pm vpcommand:

<#root>

```
cat-sp#

debug spanning-tree switch state


Spanning Tree Port state changes debugging is on

cat-sp#

debug pm vp


Virtual port events debugging is on
Nov 19 14:03:37: SP:     pm_vp 3/1(333): during state forwarding, got event 4(remove)

Nov 19 14:03:37: SP:

 @@@

pm_vp 3/1(333):
  forwarding -> notforwarding



port 3/1 (was forwarding) goes down in vlan 333


Nov 19 14:03:37: SP: *** vp_fwdchange: single: notfwd: 3/1(333)
Nov 19 14:03:37: SP: @@@ pm_vp 3/1(333): notforwarding -> present
Nov 19 14:03:37: SP: *** vp_linkchange: single: down: 3/1(333)
Nov 19 14:03:37: SP: @@@ pm_vp 3/1(333): present -> not_present
Nov 19 14:03:37: SP: *** vp_statechange: single: remove: 3/1(333)

Nov 19 14:03:37: SP:     pm_vp 3/2(333): during state notforwarding,
  got event 4(remove)
Nov 19 14:03:37: SP:

 @@@

pm_vp 3/2(333): notforwarding -> present
Nov 19 14:03:37: SP: *** vp_linkchange: single: down: 3/2(333)

Port 3/2 (was not forwarding) in vlan 333 goes down


Nov 19 14:03:37: SP: @@@ pm_vp 3/2(333): present -> not_present
Nov 19 14:03:37: SP: *** vp_statechange: single: remove: 3/2(333)

Nov 19 14:03:53: SP:     pm_vp 3/1(333): during state not_present,
  got event 0(add)
Nov 19 14:03:53: SP: @@@ pm_vp 3/1(333): not_present -> present
Nov 19 14:03:53: SP: *** vp_statechange: single: added: 3/1(333)

Nov 19 14:03:53: SP:     pm_vp 3/1(333): during state present,
  got event 8(linkup)
Nov 19 14:03:53: SP:

 @@@
```

```
pm_vp 3/1(333): present ->
  notforwarding
Nov 19 14:03:53: SP: STP SW: Gi3/1 new blocking req for 0 vlans
Nov 19 14:03:53: SP: *** vp_linkchange: single: up: 3/1(333)

Port 3/1 link goes up and blocking in vlan 333


Nov 19 14:03:53: SP:     pm_vp 3/2(333): during state not_present,
  got event 0(add)
Nov 19 14:03:53: SP: @@@ pm_vp 3/2(333): not_present -> present
Nov 19 14:03:53: SP: *** vp_statechange: single: added: 3/2(333)

Nov 19 14:03:53: SP:     pm_vp 3/2(333): during state present,
  got event 8(linkup)
Nov 19 14:03:53: SP:

 @@@

pm_vp 3/2(333): present ->
  notforwarding
Nov 19 14:03:53: SP: STP SW: Gi3/2 new blocking req for 0 vlans
Nov 19 14:03:53: SP: *** vp_linkchange: single: up: 3/2(333)

Port 3/2 goes up and blocking in vlan 333

Nov 19 14:04:08: SP: STP SW: Gi3/1 new learning req for 1 vlans
Nov 19 14:04:23: SP: STP SW: Gi3/1 new forwarding req for 0 vlans
Nov 19 14:04:23: SP: STP SW: Gi3/1 new forwarding req for 1 vlans
Nov 19 14:04:23: SP:     pm_vp 3/1(333): during state notforwarding,
  got event 14(forward_notnotify)
Nov 19 14:04:23: SP:



@@@ pm_vp 3/1(333): notforwarding ->
  forwarding
Nov 19 14:04:23: SP: *** vp_list_fwdchange: forward: 3/1(333)

Port 3/1 goes via learning to forwarding in vlan 333
```

要瞭解STP為什麼以某種方式工作，檢視交換機接收和傳送的BPDU通常很有用：

<#root>

```
cat-sp#

debug spanning-tree bpdu receive


Spanning Tree BPDU Received debugging is on
Nov  6 11:44:27: SP: STP: VLAN1 rx BPDU: config protocol = ieee,
  packet from GigabitEthernet2/1  , linktype IEEE_SPANNING ,
  enctype 2, encsize 17
Nov  6 11:44:27: SP: STP: enc 01 80 C2 00 00 00 00 06 52 5F 0E 50 00 26 42 42 03
Nov  6 11:44:27: SP: STP: Data 0000000000000000074F1CE847000000138048000 6525F0E4
  0801001001400020 00F00
Nov  6 11:44:27: SP: STP: VLAN1 Gi2/1:0000 00 00 00 000000074F1CE847 00000013
  80480006525F0E40 8010 0100 1400 0200 0F00
```

此調試適用於PVST、快速PVST和MST模式；但它不會解碼BPDU的內容。但是，您可以使用它確保收到BPDU。 要檢視BPDU的內容，請對PVST和快速PVST發出debug spanning-tree switch rx decodecommand和debug spanning-tree switch rx processcommand。發出debug spanning-tree mstp bpdu-rxcommand以檢視MST的BPDU內容：

```
<#root>

cat-sp#

debug spanning-tree switch rx decode


Spanning Tree Switch Shim decode received packets debugging is on

cat-sp#

debug spanning-tree switch rx process


Spanning Tree Switch Shim process receive bpdu debugging is on

Nov  6 12:23:20: SP: STP SW: PROC RX: 0180.c200.0000<-0006.525f.0e50 type/len 0026
Nov  6 12:23:20: SP:     encap SAP linktype ieee-st vlan 1 len 52 on v1 Gi2/1
Nov  6 12:23:20: SP:     42 42 03 SPAN
Nov  6 12:23:20: SP:     CFG P:0000 V:00 T:00 F:00 R:0000 0007.4f1c.e847 00000013
Nov  6 12:23:20: SP:     B:8048 0006.525f.0e40 80.10 A:0100 M:1400 H:0200 F:0F00

Nov  6 12:23:22: SP: STP SW: PROC RX: 0180.c200.0000<-0006.525f.0e50 type/len 0026
Nov  6 12:23:22: SP:     encap SAP linktype ieee-st vlan 1 len 52 on v1 Gi2/1
Nov  6 12:23:22: SP:     42 42 03 SPAN
Nov  6 12:23:22: SP:     CFG P:0000 V:00 T:00 F:00 R:0000 0007.4f1c.e847 00000013
Nov  6 12:23:22: SP:     B:8048 0006.525f.0e40 80.10 A:0100 M:1400 H:0200 F:0F00
```

在MST模式下，您可以使用以下命令啟用詳細的BPDU解碼：

```
<#root>

cat-sp#

debug spanning-tree mstp bpdu-rx


Multiple Spanning Tree Received BPDUs debugging is on
Nov 19 14:37:43: SP: MST:BPDU DUMP [

rcvd_bpdu Gi3/2

 Repeated]
Nov 19 14:37:43: SP: MST:   Proto:0 Version:3 Type:2 Role: DesgFlags[   F    ]
Nov 19 14:37:43: SP: MST:   Port_id:32897 cost:2000019
Nov 19 14:37:43: SP: MST:   root_id  :0007.4f1c.e847 Prio:0
Nov 19 14:37:43: SP: MST:   br_id    :00d0.003f.8800 Prio:32768
Nov 19 14:37:43: SP: MST:   age:2 max_age:20 hello:2 fwdelay:15
Nov 19 14:37:43: SP: MST:   V3_len:90 PathCost:30000 region:STATIC rev:1
Nov 19 14:37:43: SP: MST:   ist_m_id :0005.74
Nov 19 14:37:43: SP: MST:BPDU DUMP [

rcvd_bpdu Gi3/2
```

```
 Repeated]
Nov 19 14:37:43: SP: MST:   Proto:0 Version:3 Type:2 Role: DesgFlags[   F    ]
Nov 19 14:37:43: SP: MST:   Port_id:32897 cost:2000019
Nov 19 14:37:43: SP: MST:   root_id  :0007.4f1c.e847 Prio:0
Nov 19 14:37:43: SP: MST:   br_id    :00d0.003f.8800 Prio:32768
Nov 19 14:37:43: SP: MST:   age:2 max_age:20 hello:2 fwdelay:15
Nov 19 14:37:43: SP: MST:   V3_len:90 PathCost:30000 region:STATIC rev:1
Nov 19 14:37:43: SP: MST:   ist_m_id :0005.7428.1440 Prio:32768  Hops:18
  Num Mrec: 1
Nov 19 14:37:43: SP: MST: stci=3  Flags[ F    ] Hop:19 Role:Desg [Repeated]
Nov 19 14:37:43: SP: MST:         br_id:00d0.003f.8800 Prio:32771 Port_id:32897
  Cost:2000028.1440 Prio:32768  Hops:18  Num Mrec: 1
Nov 19 14:37:43: SP: MST: stci=3  Flags[ F    ] Hop:19 Role:Desg [Repeated]
Nov 19 14:37:43: SP: MST:         br_id:00d0.003f.8800 Prio:32771 Port_id:32897
  Cost:20000
```

註：對於Cisco IOS軟體版本12.1.13E及更高版本，支援STP的條件調試。這表示您可以對基於每個埠或每個VLAN接收或傳輸的BPDU進行調試。

發出debug condition vlan_num 或debug condition interface命令，將調試輸出的範圍限製為每介面或每VLAN。

保護網路免受轉發環路的影響

Cisco開發了許多功能和增強功能，可在STP無法管理某些故障時保護網路免受轉發環路的影響。

當您對STP進行故障排除時，它有助於隔離特定故障並可能找到其原因，而實施這些增強是保護網路免受轉發環路影響的唯一方法。

以下是保護您的網路免受轉發環路影響的方法：

1.在所有交換機到交換機鏈路上啟用單向鏈路檢測(UDLD)

如需更多有關UDLD的資訊，請參閱瞭解和設定單向連結偵測通訊協定功能。

2.在所有交換機上啟用環路防護

有關環路防護的詳細資訊，請參閱使用環路防護和BPDU遲滯檢測功能的生成樹協定增強功能。

啟用後，UDLD和環路防護可消除大部分轉發環路的原因。有缺陷的鏈路（或依賴於有缺陷的硬體的所有鏈路）關閉或被阻塞，而不是造成轉發環路。

注意：雖然這兩個功能看起來有些冗餘，但每個功能都有其獨特的功能。因此，同時使用這兩個功能可提供最高級別的保護。有關UDLD和環路防護的詳細比較，請參閱環路防護與單向鏈路檢測。

對於必須使用主動還是普通UDLD，有不同的意見。與正常模式UDLD相比，主動式UDLD無法提供更多環路保護。主動UDLD檢測埠停滯情況（當鏈路處於開啟狀態但沒有關聯的流量黑洞）。此新

增功能的缺點在於，如果沒有出現一致故障，進取UDLD可能會潛在停用連結。通常，人們會將UDLDhellointerval的修改與攻擊性的UDLD功能混淆。這是不正確的。可以在兩種UDLD模式下修改計時器。

注意：在極少數情況下，主動式UDLD可以關閉所有上行鏈路埠，這基本上將交換機與網路的其餘部分隔離。例如，如果兩個上游交換機的CPU使用率都極高，並且都使用主動模式UDLD，就可能會發生這種情況。因此，如果交換機沒有帶外管理，則建議您配置無法侵蝕的超時。

3.在所有終端站連線埠上啟用Portfast

必須啟用portfast以限制TC的量和後續泛洪，這可能會影響網路的效能。僅對連線到終端站的連線埠使用此命令。否則，意外拓撲環路可能導致資料包環路，並中斷交換機和網路運行。

注意:使用no spanning-tree portfast命令時請務必小心。此命令僅刪除任何埠特定的portfast命令。如果在全域性配置模式下定義spanning-tree portfast default命令，並且埠不是中繼埠，則此命令隱式啟用portfast。如果沒有全域性配置portfast，no spanning-tree portfast命令相當於spanning-tree portfast disable命令。

4.將兩端（支援時）的EtherChannel設定為`Desirable`模式，並設定`Non-Silent`選項

Desirablemode可以啟用連線埠聚合通訊協定(PAgP)，以確保通道化的對等點之間的運行時一致性。這提供了額外的環路保護功能，尤其是在通道重新配置期間（例如鏈路加入或離開通道時，以及鏈路故障檢測）。有一個內建的通道配置錯誤防護，該防護預設啟用，可防止由於通道配置錯誤或其他情況導致的轉發環路。有關此功能的詳細資訊，請參閱瞭解EtherChannel不一致偵測。

5.請勿在交換器到交換器連結上停用自動交涉（如支援）

自動協商機制可以傳遞遠端故障資訊，這是檢測遠端故障的最快捷方式。如果在遠端端檢測到故障，即使鏈路收到脈衝，本地端也會關閉鏈路。與UDLD等高級檢測機制相比，自動交涉的速度極快（在微秒內），但缺少UDLD的端到端覆蓋範圍（例如整個資料路徑：CPU — 轉發邏輯 — 埠1 — 埠2 — 轉發邏輯 — CPU與埠1 — 埠2）。積極UDLD模式在故障檢測方面提供類似自動交涉的功能。當連結的兩端支援交涉時，不需要啟用主動模式UDLD。

6.調整STP計時器時請務必小心

STP計時器取決於彼此和網路拓撲。如果對計時器進行任意修改，STP將無法正常工作。有關STP計時器的詳細資訊，請參閱瞭解和調整生成樹協定計時器。

7.如果可能存在拒絕服務攻擊，請使用根防護來保護網路STP邊界

根防護和BPDU防護使您能夠保護STP不受外部影響。如果有可能發生此類攻擊，則必須使用根防護和BPDU防護來保護網路。有關根防護和BPDU防護的詳細資訊，請參閱以下文檔：

- 跨距樹狀目錄通訊協定根防護增強功能

- 跨距樹狀目錄 PortFast BPDU 防護增強功能

8.在已啟用Portfast的埠上啟用BPDU防護，以防止STP受到連線到埠的未授權網路裝置（如集線器、交換機和橋接路由器）的影響

如果正確配置根防護，它將防止STP受到外部影響。如果啟用BPDU防護，則會關閉接收任何BPDU的連線埠。這對調查事件很有用，因為BPDU防護會生成系統日誌消息並關閉埠。如果根防護或BPDU防護無法防止短週期環路，則兩個快速啟用的埠直接連線或通過集線器連線。

9.避免管理VLAN上的使用者流量

管理VLAN包含在構建塊中，而不是整個網路。

交換器管理介面會在管理VLAN上接收廣播封包。如果出現過多的廣播（例如廣播風暴或應用程式故障），交換機CPU可能會過載，從而可能扭曲STP操作。

10.可預測的（硬編碼）STP根和備份STP根位置

必須配置STP根和備份STP根，以便在發生故障時以可預測的方式進行收斂，並在每個場景中建立最佳拓撲。請勿將STP優先順序保留為預設值，以防止無法預測的根交換機選擇。

相關資訊

- LAN 產品支援
- LAN 交換技術支援

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。