

排除ACI交换矩阵端口跟踪故障

目录

[简介](#)

[概述](#)

[拓扑](#)

[推荐的故障排除核对表](#)

[确认症状](#)

[验证交换矩阵端口跟踪策略](#)

[验证交换矩阵上行链路LLDP邻居](#)

[检查交换矩阵上行链路接口状态和抖动历史记录](#)

[检查受影响的下行链路接口状态](#)

[查看交换矩阵端口跟踪调试日志](#)

[检查收发器详细信息](#)

[将物理接口映射到内部端口](#)

[查看平台链接事件历史记录](#)

[如果需要，检查并配置链路反退回](#)

[统一命令参考](#)

[案例角场景1:接口未摆动，但交换矩阵端口跟踪已触发](#)

[案例角场景2:BiDi光纤和无源TAP干扰](#)

[vPC注意事项](#)

[APIC连接端口的重要说明](#)

[参考](#)

简介

本文档介绍ACI交换矩阵端口跟踪功能、补救步骤和案例情景。

概述

思科ACI交换矩阵端口跟踪（也称为交换矩阵跟踪或端口跟踪）是ACI枝叶交换机上使用的恢复功能，用于根据面向交换矩阵/上行链路端口的运行状态控制面向主机/下行链路端口的状态。

交换矩阵端口跟踪旨在防止枝叶失去到ACI交换矩阵的足够连接时流量黑洞。如果没有此功能，即使枝叶已丢失其交换矩阵上行链路，面向主机的接口也可以保持物理正常运行。在这种情况下，连接的终端可以继续将流量转发到枝叶，但枝叶必须无法将该流量转发到交换矩阵。

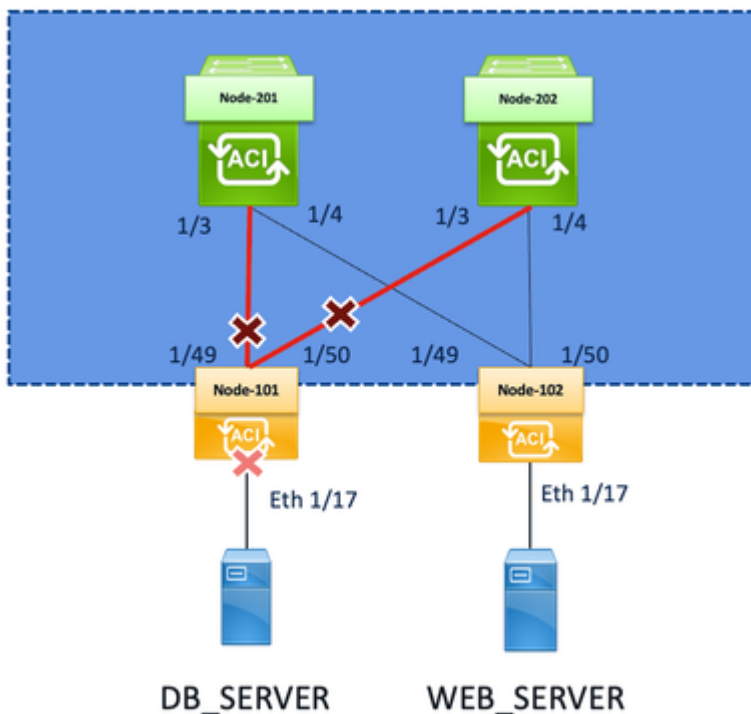
启用交换矩阵端口跟踪时，枝叶会监控其主干层上的活动交换矩阵上行链路，并将运行交换矩阵链路的数量与配置的阈值进行比较。如果可用的交换矩阵链路数低于配置的最小值，枝叶会自动关闭所选面向主机/下行链路接口。这样，连接的终端、服务器或外部设备可以检测链路断开事件并故障切换到另一个可用的路径或枝叶，而不是继续将流量发送到不再具有足够交换矩阵连接的枝叶。

一旦所需的交换矩阵上行链路数恢复，且运行交换矩阵链路数恢复超过配置的阈值，下行链路接口将在配置的恢复延迟后重新打开。

示例行为：

- 枝叶具有两个到主干的上行链路。
- 交换矩阵端口跟踪已启用。
- 未满足最低活动交换矩阵链路阈值。
- 枝叶禁用面向主机/下行链路接口。
- 连接的服务器或外部交换机检测到链路断开事件并进行故障转移。
- 交换矩阵连接恢复后，枝叶在配置的延迟后重新启用面向主机的端口。

拓扑



推荐的故障排除核对表

调查思科ACI交换矩阵端口跟踪问题时，请使用核对表。每个步骤都包含相关的验证或故障排除命令。

确认症状

验证面向主机/下行链路的端口是否关闭，以及事件是否与交换矩阵端口跟踪相关。

检查交换矩阵端口跟踪故障F0532:

```
moquery -c faultInst -f 'fault.Inst.code=="F0532"'
```

示例指示：

```
descr    : Port is down, reason being fabricTrack(connected)
severity : critical
subject  : port-down
```

如果存在故障F0532，则由于交换矩阵端口跟踪导致接口关闭。

验证交换矩阵端口跟踪策略

确认是否已启用交换矩阵端口跟踪并查看配置参数。

```
moquery -c infraPortTrackPol | egrep "adminSt|delay|includeApicPorts|minlinks"
```

查看显示的值：

参数	目的
adminSt	指示是启用还是禁用交换矩阵端口跟踪。
延迟	在重新启用下行链路端口之前恢复延迟。
包括ApicPorts	指示是否包括APIC连接的端口。
minlinks	所需操作交换矩阵链路的最小数量。

示例：

```
adminSt      : on
delay        : 300
includeApicPorts : no
minlinks     : 0
```

验证交换矩阵上行链路LLDP邻居

确认枝叶仍能通过交换矩阵上行链路看到预期的主干邻居。

```
show lldp neighbors
```

有关特定交换矩阵上行链路的详细信息：

```
show lldp neighbors int ethernet 1/49 detail
```

使用此输出确认：

- 本地枝叶接口。
 - 远程主干节点。
 - 远程主干接口。
 - LLDP保持时间。
 - 预期邻居是否仍然存在。
-

检查交换矩阵上行链路接口状态和抖动历史记录

验证面向交换矩阵的接口最近是否出现摆动。

```
show int eth 1/49 | egrep "flapped|state"
```

示例：

```
admin state is up, Dedicated Interface
Last link flapped 00:02:57
```

交换矩阵上行链路上的最新抖动可以解释为什么触发交换矩阵端口跟踪。

检查受影响的下行链路接口状态

检验面向主机/下行链路接口的状态和抖动历史记录。

```
show int eth 1/17 | egrep "flapped|state|fabric-track"
```

这有助于将下行链路端口事件与交换矩阵上行链路故障相关联。

查看交换矩阵端口跟踪调试日志

检查受影响枝叶上的Fabric Port-Track进程日志。

```
cat /var/sysmgr/tmp_logs/fabric_track.py.dbg | tail -n 15
```

正常操作期间的日志输出示例：

```
cat /var/sysmgr/tmp_logs/fabric_track.py.dbg | tail -n 15
Reading the port track Mo
...
Reading the port track Mo
```

故障窗口期间的示例日志文件：

<#root>

```
cat /var/sysmgr/tmp_logs/fabric_track.py.dbg | tail -n 15
Reading Isis Mo to check for Isis Adjacency
1 Fabric links are up
Reading 11PhysIf Mos of fabric links to check number of up fabric links
Bringdown: 0 Fabric links left up
PortTrackIf Mo is not present. Creating PortTrackIf Mo for
```

eth1/17

Committing the port track Mo

这些消息表明枝叶检测到交换矩阵链路不足，并为受影响的下行链路端口创建了PortTrack接口对象。

主要观察结果：

- 枝叶检测到一个交换矩阵链路已打开。
- 此后不久，它检测到零个交换矩阵链路处于打开状态。
- Fabric Port-Track创建的PortTrack（如果受管对象受影响接口）。
- 下行链路接口（例如eth1/17）关闭。

检查收发器详细信息

收集受影响交换矩阵上行链路的光纤信息。

```
show interface ethernet 1/49 transceiver details | egrep "type|name|serial"
```

示例：

```
type is QSFP-40/100-SRBD
name is CISCO-FINISAR
serial number is FIW2440004Z-B
```

在排除故障时，这一点尤为重要：

- 光纤故障。
- BiDi光纤。
- 被动TAP
- 监控工具交互。
- 意外的链路摆动。

将物理接口映射到内部端口

确定与物理接口相关的内部端口号。

```
vsh_lc -c 'show platform internal usd port info' | egrep "Eth1/49" -A 1
```

示例：

```
<#root>
```

```
Port 61.0 (Eth1/49) : Admin UP (1) Link UP Cfg_Fec Disabled Fec Disabled Fcot Fiber retimer 0x0  
AN_knob No AN_cfg Yes AN_operSt No In_debounce 0,
```

```
Debounce-Time 0
```

```
usecs qsa: No
```

在本示例中，Eth1/49映射到内部端口61.0。

查看平台链接事件历史记录

识别内部端口后，查看链路事件历史记录。

```
vsh_lc -c 'show platform internal tah event-history linkevents' | grep Port "61.0" -A 1
```

无反退回的示例：

```
Port 61.0: tahusd_port_handle_debounce: No debounce required!!
```

配置了反退回的示例：

```
Port 61.0: tahusd_port_handle_debounce/9481: Started Debounce Timer for 10000 ms
```

这将确认链路事件期间是否应用了链路反退回。

如果需要，检查并配置链路反退回

验证是否为交换矩阵接口配置了链路退回。链路反退回有助于防止瞬变微摆动立即触发交换矩阵端口跟踪行为。

检查交换矩阵接口策略：

```
moquery -c fabricFifPol | egrep "dn|linkDebounce"
```

示例：

```
dn          : uni/fabric/fintfpol-default
linkDebounce : 0
```

从接口直接检查退回：

```
show interface eth1/49 debounce
```

无反退回的示例：

```
-----
Port          Debounce time  Value(ms)
-----
Eth1/49       disable        0
```

如果禁用了反退回功能，并且怀疑存在微抖动，请在交换矩阵接口上配置反退回：

```
configure
leaf 101
interface ethernet 1/49
link debounce time 100
```

重要信息：

- 该值以毫秒为单位。
- 100等于1秒。

检查配置:

```
show interface eth1/49 debounce
```

预期输出：

Port	Debounce time	Value(ms)
Eth1/49	enable	100

默认的反退回间隔为0毫秒。我们建议使用100毫秒的值，但您可以选择适合您的交换矩阵的值。

统一命令参考

任务	命令
检查交换矩阵端口跟踪故障	moquery -c faultInst -f 'fault.Inst.code=="F0532"'
验证交换矩阵端口跟踪策略	moquery -c infraPortTrackPol egrep "adminSt delay includeApicPorts minlinks"
检查LLDP邻居	show lldp neighbors
检查详细的LLDP邻居	show lldp neighbors int ethernet 1/49 detail
检查交换矩阵上行链路状态	show int eth 1/49 egrep "flapped state"
检查下行链路状态	show int eth 1/17 egrep "flapped state fabric-track"
检查交换矩阵端口跟踪调试日志	cat /var/sysmgr/tmp_logs/fabric_track.py.dbg tail -n 15
检查收发器详细信息	show interface ethernet 1/49收发器详细信息 egrep "type name serial"
将物理接口映射到内部端口	vsh_lc -c 'show platform internal usd port info' egrep "Eth1/49" -A 1
检查平台链接事件	vsh_lc -c 'show platform internal tah event-history linkevents' grep端口"61.0" -A 1
检查交换矩阵反退回策略	moquery -c fabricFlfPol egrep "dn linkDebounce"
检查接口退回	show interface eth1/49 debounce
配置退回	链路反退回时间10000

案例角场景1:接口未摆动，但交换矩阵端口跟踪已触发

当物理交换矩阵接口未摆动，但交换矩阵端口跟踪仍表现为交换矩阵链路不可用时，可能出现拐角情况。

示例：

```
show int eth 1/49 | egrep "flapped|state"  
admin state is up, Dedicated Interface  
Last link flapped 1y14w
```

在此场景中，接口最近没有摆动。

由于Fabric Port-Track依赖于受管对象查询，因此验证枝叶是否能够成功执行相关moquery:

```
moquery -c l1PhysIf -x 'query-target-filter=and(anybit(l1PhysIf.usage,"fabric"),eq(l1PhysIf.switchingSt
```

还要检查磁盘利用率，问题条件示例：

```
df -h  
Filesystem Size Used Avail Use% Mounted on  
rootfs 2.5G 2.5G 0 100% /bin
```

如果根文件系统已满，枝叶可能会丢弃或使内部功能(包括moquery)失败。因此，Fabric Port-Track必须无法确认交换矩阵链路是否已启用，并且必须错误地关闭下行链路接口。

建议操作:

- 检查可用磁盘空间。
- 根据思科指南清除或修复文件系统问题。
- 重新验证moquery功能。
- 确认正确检测到交换矩阵上行链路。

案例角场景2:BiDi光纤和无源TAP干扰

有一个特定问题涉及用于监控的BiDi QSFP光纤和无源光纤TAP。

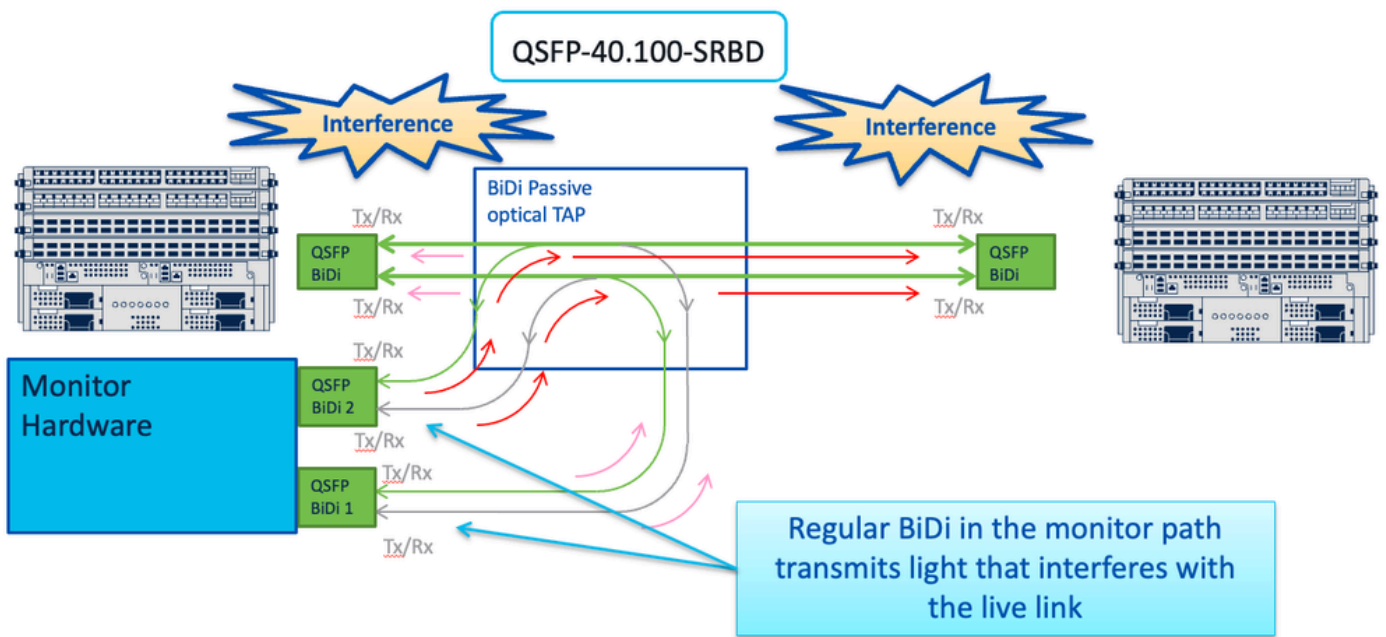
被动TAP风险

当被动TAP基础设施插入枝叶和主干之间，并且监控设备使用常规BiDi光纤时，监控路径可以将光线传回实时生产链路。

这可能导致：

- 意外光信号注入。
- 两端的链路中断事件。
- 枝叶和主干均报告远程故障情况。
- 由于瞬时交换矩阵上行链路丢失，触发交换矩阵端口跟踪。

在这种情况下，重新加载监控交换机导致出现意外光信号，导致枝叶和主干链路发生故障。

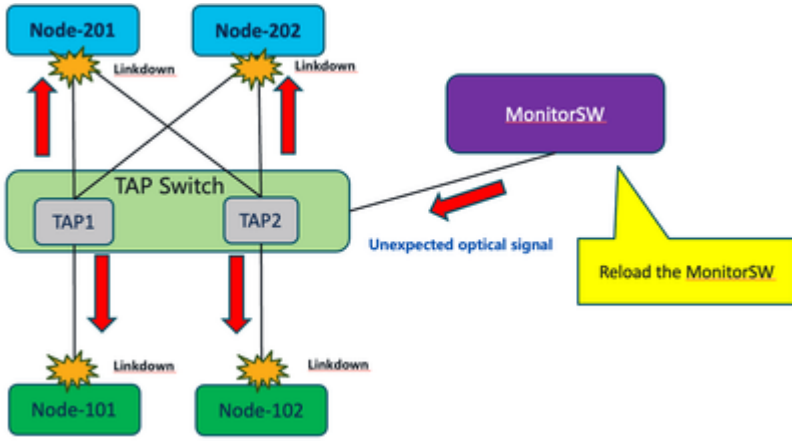


标准SR光纤 — QSFP-40/100-SRBD

使用标准SR光纤时，传输和接收路径是分开的：

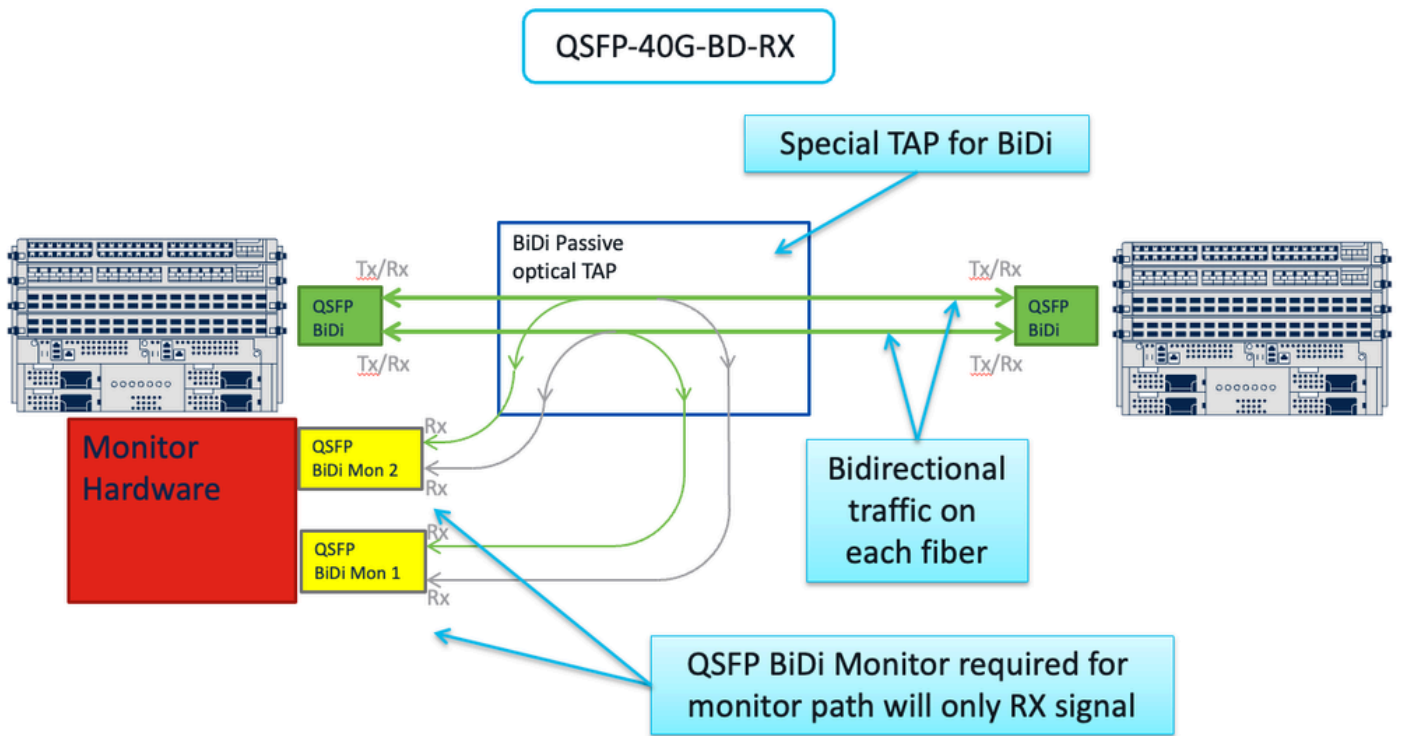
Tx -> Rx
Rx <- Tx

每个光纤的流量都是单向的。



推荐的缓解

对于BiDi监控方案，请使用仅接收且不传输到生产路径的适当的仅监控BiDi光纤。



BiDi光纤 — QSFP-40G-BD-RX

使用BiDi光纤，传输和接收都存在于每根光纤上：

Tx/Rx <-> Tx/Rx

这被描述为一个特殊的TAP/监视器BiDi光学元件，其中监视器路径只接收信号。

vPC注意事项

对于vPC连接的下行链路端口，恢复行为可能同时受交换矩阵端口跟踪延迟计时器和vPC延迟计时器影响。

对于vPC配置，如果枝叶节点丢失所有交换矩阵端口并因此丢失ISIS邻接关系，则无法与其vPC对等设备通信。在这种情况下，下行链路端口将在vPC延迟计时器或端口跟踪延迟计时器超时后重新激活。

运营影响：

- 非vPC下行链路端口必须与交换矩阵端口跟踪恢复延迟匹配。
- 如果vPC延迟计时器大于交换矩阵端口跟踪延迟，vPC下行链路端口可以保持更长的关闭时间。

示例：

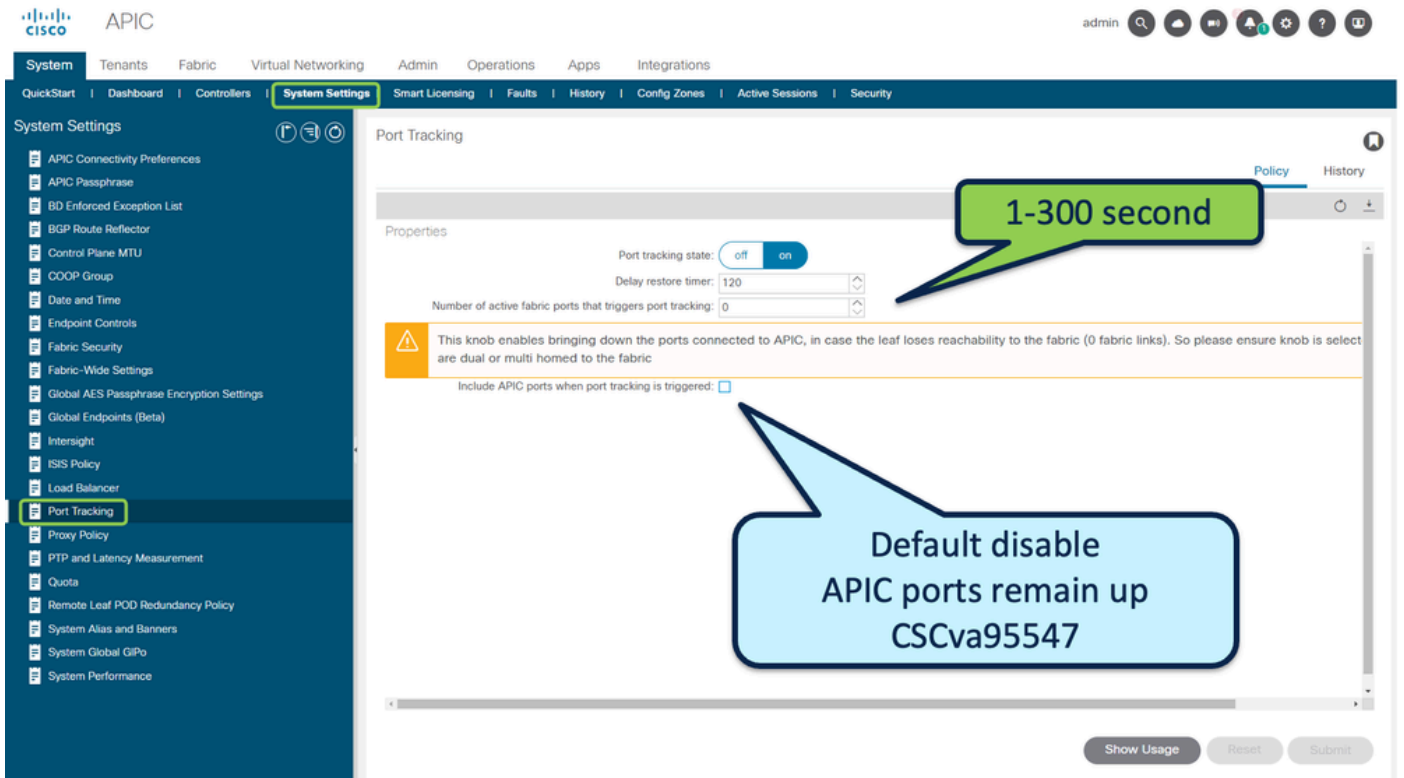
- 交换矩阵端口跟踪延迟：300 秒
- vPC延迟计时器：600 秒
- vPC下行链路端口必须在600秒后恢复

APIC连接端口的重要说明

与APIC连接的端口和交换矩阵端口跟踪行为相关的思科漏洞ID [CSCva9547](#)。

一个关键的操作考虑事项是，在瞬变上行链路故障期间，面向APIC的端口通常不能被交换矩阵端口跟踪关闭，因为这可能影响管理和控制器连接。

包括ApicPorts选项用于控制行为中是否包含APIC连接的接口。



这表示交换矩阵端口跟踪不会禁用与APIC连接的端口。

参考

[Cisco APIC基本配置指南>章节：调配核心ACI交换矩阵服务>链路反退回间隔](#)

[Cisco APIC第2层网络配置指南>章节：交换矩阵端口跟踪](#)

[思科以应用为中心的基础设施\(ACI\)设计指南>端口跟踪](#)

[Cisco Application Centric Infrastructure Fundamentals > Chapter:交换矩阵调配>用于交换矩阵端口故障检测的端口跟踪策略](#)

引用错误：

[Cisco Bug ID CSCva95547 \(仅限注册用户 \)：端口跟踪功能请求：用于禁用转至APIC的枝叶交换机端口的旋钮](#)

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。