

在Nexus 9000上配置选择性队列并对其进行故障排除

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[网络图](#)

[配置](#)

[故障排除](#)

[一致性检查器](#)

[用于收集和检查任何问答问题的有用命令](#)

[相关信息](#)

简介

本文档介绍如何在Nexus 9300 (EX-FX-GX)模型中配置和排除选择性Q-in-Q故障。

先决条件

要求

Cisco 建议您了解以下主题：

- Q-in-Q封装
- Dot1q交换
- Nexus 9000

使用的组件

本文档中的信息基于以下软件和硬件版本：

- 硬件：N9K-C9336C-FX2
- 版本：10.2(3)

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

Q-in-Q VLAN隧道使服务提供商能够隔离其基础设施中不同用户的流量，同时通过向已标记的帧添加第二个802.1Q标记，仍然为您提供完整的VLAN供内部使用。

具有多个提供商VLAN的选择性队列是隧道功能，它允许端口上用户特定范围的VLAN与一个特定提供商VLAN相关联。

Q-in-Q允许在一个端口上具有多个用户VLAN到提供商VLAN的映射。

使用服务提供商VLAN的属性，传入的VLAN标记与端口上配置的任何用户VLAN匹配的数据包将通过交换矩阵进行隧道传输。封装的数据包将用户VLAN标记作为内部数据包的第2层报头的一部分。

网络图



配置

配置选择性Q-in-Q的目的是在接口eth1/1上映射N9KFX2和N9KFX2-1中接收的流量中的VLAN。

Nexus封装流量并将其与DOT1Q报头VLAN 20一起发送到提供商，由Nexus 93180FX模拟。

主机A

```
interface Ethernet1/1
  switchport
  switchport mode trunk
  no shutdown

interface Vlan10
  no shutdown
  ip address 192.168.1.1/24
```

主机B

```
interface Ethernet1/1
  switchport
  switchport mode trunk
  no shutdown

interface Vlan10
  no shutdown
  ip address 192.168.1.2/24
```

N9KFX2

```
interface Ethernet1/1
  switchport
  switchport mode trunk
  switchport vlan mapping all dot1q-tunnel 20
  switchport trunk allowed vlan 20
  spanning-tree port type edge trunk
  spanning-tree bpdufilter enable
  no shutdown

interface Ethernet1/35
  switchport
  switchport mode trunk
  no shutdown
```

N93180FX

```
system dot1q-tunnel transit

interface Ethernet1/17
  switchport
  switchport mode trunk
  no shutdown

interface Ethernet1/18
  switchport
  switchport mode trunk
  no shutdown
```

N9KFX2-1

```
interface Ethernet1/1
  switchport
  switchport mode trunk
```

```
switchport vlan mapping all dot1q-tunnel 20
switchport trunk allowed vlan 20
spanning-tree port type edge trunk
spanning-tree bpdupfilter enable
no shutdown
```

```
interface Ethernet1/35
switchport
switchport mode trunk
no shutdown
```

故障排除

Q-in-Q编程通过组件从控制平面移动到硬件：

:



一致性检查器

Nexus交换机的最新版本具有一致性检查器，可用于防止编程错误。

```
N9KFX2# show consistency-checker selective-qinq
```

```
Box level hardware programming checks      : PASS
Fetching ingressVlanXlate entries from slice: 0 HW
Fetching ingressVlanXlate entries from slice: 1 HW
Fetching ingressVlanXlate TCAM entries
Performing port specific checks for intf Eth1/1
Port specific selective Qinq checks for interface Eth1/1 : PASS
Selective Qinq Overall status              : PASS
```

控制层面

确认端口上的VLAN是否处于活动状态。

```
N9KFX2#sh vlan
```

VLAN	Name	Status	Ports
1	default	active	Eth1/1
20	VLAN0020	active	Eth1/1, Eth1/35

确认已清楚地看到VLAN映射。

```
N9KFX2# sh vlan internal info mapping
```

Per Port Vlan Translation Details

```
=====
Entry No          : 0
-----
  ifindex Eth1/1(0x1a000000)
  vlan mapping enabled: TRUE
  vlan translation mapping information (count=1):
    Original Vlan    Translated Vlan
    -----
    all              20

Entry No          : 1
-----
  ifindex Eth1/35(0x1a004400)
  vlan mapping enabled: FALSE
  vlan translation mapping information (count=0):
    Original Vlan    Translated Vlan
    -----
```

ELTM

确认是否有任何错误增加或非零。

```
N9KFX2# show system internal eltm error count
interface config error: 0
dce error: 0
port struct error: 0
port channel error: 0
tunnel error: 0
tlv count error: 0
vpn count error: 0
SI add to DB error: 0
```

运行attach module X命令。X是分配物理接口的模块。

```
N9KFX2# attach module 1
module-1# show system internal eltm info vlan 20
IFTMC Detailed info for VLAN = 20
cr_flags = VLAN , bd = 21(0x15) stg-id = 0 type = 1 stp:0
shut_state = 2 , oper_state = 1
vni = 0, ipmc_idx = 0x0 ipmc_remap_idx = 0x0 ipmc_remap_idx2 = 0x0 dvif: 0
ipmc_mrouters_idx = 0x0 ipmc_mrouters_remap_idx = 0x0 ipmc_mrouters_remap_idx2 = 0x0
pbp_idx=0
eid[0] = 0, alt_eid[0] = 0eid[1] = 0, alt_eid[1] = 0eid[2] = 0, alt_eid[2] = 0eid[3] = 0, alt_eid[3] = 0
```

```

Active ports:                                     <<<<< Confirm Active ports are listed for the
  Ethernet1/1(0x1a000000)  Ethernet1/35(0x1a004400)

Active Local ports:
  Ethernet1/1(0x1a000000)  Ethernet1/35(0x1a004400)

Ports using this VLAN as Provider:                <<<<< Displays the interface that is doing
  Ethernet1/1(0x1a000000)

SDB Active ports:
  Ethernet1/1

```

pbp_idx:0

```

VLAN Features:
  prim_vlan      : 0, prim_intf_vlan : 0, vlan mode      : 0(CE)
  pt_cam_en     : 1, ipv4_igmp_snoop : 1, ipv4_pim_snoop : 0
  ipv6_mld_snoop : 0, ipv6_pim_snoop : 0, l2l3_lkup_cfg : 1
  v4_table_id   : 1      (0x00000001), v4_vpn_id       : 1 (0x1)
  v6_table_id   : 2147483649 (0x80000001), v6_vpn_id    : 1 (0x1)
  age_tmr_sel   : 1, sm_en : 0 fcf : 0 omf : 1, dhcp_snoop_en : 0
  proxy_learn   : 0, iic_en : 0, use_oxid : 0, age_grp_sel : 0
  disable_learn : 0, otv_en : 0, global : FALSE
  bc_uc_ftag    : 4095 , mcast_ftag_num : 1, arp_sup: 0x0 , icmpv6 nd local proxy: 0x0
  isdot1qvlan  : 0 , num_dot1q_ports: 0 num_sel_dot1q QinVni_ports: 0
  num_multi_tag_ports: 0, eid: 0, xconnect: 0, mcast_ftag[1] : {4095}
  is_transit    : 0 mdns_en: 0,

```

TAH:VLAN_CONTROL:

```

-----
vrf: 0 forwarding_vlan: 0 forwarding mode: 0
Flags: 0x0
up: 0 vni: 0 is_sr_evpn: 0

```

您可以检查接口编程。输出显示有关端口编程的信息。

```

N9KFX2# attach module 1
module-1# show system internal eltm info interface e1/1
IFTMC Detailed info for Interface Ethernet1/1(0x1a000000)
CR_flags = INTF LIF (0x3), LTL = 6144 (0x1800), (S 0x0, P 0x0)
IF_INDEX = Ethernet1/1(0x1a000000), LIF = 4100 (0x1004), SDB LIF = 4100 (0x1004)

State = UP
Layer = L2, Mode = TRUNK, native_vlan = 1 tag_native_vlan = 0 <<< Confirm mode and Layer
AdminMode = 0x100000 is_monitor = 0 src_if_pruning 1
Operational VLAN's (2):                                     <<< Confirm if the VLAN for tunnel and
  1,20
voice vlan = 4096
MVR Configured VLAN's (0):

Local_port = 1, ldb_sharing = 1, ilm_sharing = 0x1
ldb_port_flags = 0x0, dsm = 0, dnl = 0
pc_ifindx = 0x0, fpc = 0x0
Interface Features:
  ipv4_en      : 0, ipv4_mcast_en : 0, df_mask : 0, mpls_en : 0
  ipv6_en      : 0, ipv6_mcast_en : 0 pim_dr_en :0 ipsg_en : 0
  FC port mode:0 vsan: 0 trunk mode: 0 speed: 0

```

FCF Enabled : 0 FCF MAC:0000:0000:0000
PTP fid = 0
mpls_strip_en : 0
external port : 0
Operational VSAN's (0):

v4_table_id : 1 (0x00000001), v4_vpn_id : 1 (0x1)
v6_table_id : 2147483649 (0x80000001), v6_vpn_id : 1 (0x1)
per_pkt_ls_en : 0, bd : N/A (0x0)
icmp_redirect : 1, ipv6_redirect : 1
v4_same_if_chk : 0, v6_same_if_check : 0
mtu_index : 1 (0x0001), new_mtu_index : 1 (0x0001)
mtu : 1500 (0x05dc), port_trust : 0
v4_rpf_mode : 0 , v6_rpf_mode : 0
v4_rpf_mode (sdb) : 0 , v6_rpf_mode (sdb) : 0
intf_type : 0x0, intf_flags : 0x20
sub_flags : 0x0, port_type : 0x0, is_aida : 0x0, vpc_dvp_no : 0x0
fcoe_mac_ip_chk: 0 buffer_boost: 0 openflow_en: 0 span_buffer_cfg: 0 span_mode: 0 outer_bd = 512
interface mac: 00:00:00:00:00:00 old_mac: 00:00:00:00:00:00 mac_index:0
Src-Idx:1
mc_index : 4100, egress_vs1_drop: 0, pv_en : 0, is_selective_qinvni_port 1 ether_type 0x8100 vlan_bmp.
is_multi_tag_port 0, nat_realm_id 0
global_id 0
enm_if_type : 0
pbb_idx : 0
pinned_ifindex : 0x0
normal_vlan_bmp (0):

arp_suppress inst[0] 0 arp_suppress inst[1] 0 arp_suppress inst[2] 0 arp_suppress inst[3] 0 arp_suppress
inst[4] 0 mdns_en inst[5] 0 mdns_en inst[6] 0 mdns_en inst[7] 0 arp_suppress
inst[8] 0 mdns_en inst[9] 0 mdns_en inst[10] 0 mdns_en inst[11] 0 mdns_en inst[12] 0 mdns_en
inst[13] 0 mdns_en inst[14] 0 mdns_en inst[15] 0 mdns_en inst[16] 0 mdns_en inst[17] 0 mdns_en
inst[18] 0 mdns_en inst[19] 0 mdns_en inst[20] 0 mdns_en inst[21] 0 mdns_en inst[22] 0 mdns_en
inst[23] 0 mdns_en inst[24] 0 mdns_en inst[25] 0 mdns_en inst[26] 0 mdns_en inst[27] 0 mdns_en
inst[28] 0 mdns_en inst[29] 0 mdns_en inst[30] 0 mdns_en inst[31] 0 mdns_en inst[32] 0 mdns_en
inst[33] 0 mdns_en inst[34] 0 mdns_en inst[35] 0 mdns_en inst[36] 0 mdns_en inst[37] 0 mdns_en
inst[38] 0 mdns_en inst[39] 0 mdns_en inst[40] 0 mdns_en inst[41] 0 mdns_en inst[42] 0 mdns_en
inst[43] 0 mdns_en inst[44] 0 mdns_en inst[45] 0 mdns_en inst[46] 0 mdns_en inst[47] 0 mdns_en
inst[48] 0 mdns_en inst[49] 0 mdns_en inst[50] 0 mdns_en inst[51] 0 mdns_en inst[52] 0 mdns_en
inst[53] 0 mdns_en inst[54] 0 mdns_en inst[55] 0 mdns_en inst[56] 0 mdns_en inst[57] 0 mdns_en
inst[58] 0 mdns_en inst[59] 0 mdns_en inst[60] 0 mdns_en inst[61] 0 mdns_en inst[62] 0 mdns_en
inst[63] 0 mdns_en inst[64] 0 mdns_en inst[65] 0 mdns_en inst[66] 0 mdns_en inst[67] 0 mdns_en
inst[68] 0 mdns_en inst[69] 0 mdns_en inst[70] 0 mdns_en inst[71] 0 mdns_en inst[72] 0 mdns_en
inst[73] 0 mdns_en inst[74] 0 mdns_en inst[75] 0 mdns_en inst[76] 0 mdns_en inst[77] 0 mdns_en
inst[78] 0 mdns_en inst[79] 0 mdns_en inst[80] 0 mdns_en inst[81] 0 mdns_en inst[82] 0 mdns_en
inst[83] 0 mdns_en inst[84] 0 mdns_en inst[85] 0 mdns_en inst[86] 0 mdns_en inst[87] 0 mdns_en
inst[88] 0 mdns_en inst[89] 0 mdns_en inst[90] 0 mdns_en inst[91] 0 mdns_en inst[92] 0 mdns_en
inst[93] 0 mdns_en inst[94] 0 mdns_en inst[95] 0 mdns_en inst[96] 0 mdns_en inst[97] 0 mdns_en
inst[98] 0 mdns_en inst[99] 0 mdns_en
service provider Vlans 0

vlan_xlt_tlb_en_ingress : 1 num_vlan_xlt_ingr : 1

Vlan Translation Table (INGRESS) <<< This table displays xlt the vlan used to send to the pro

in_vlan xlt_vlan
1 20

SDB INFO:

MVR VLAN's (0):

STATE = UP, is_mcec = 0, is_mct = 0
TAH:L2 INFO: 4100

Flags : 0x41

NIV Index: 1 Source Chip: 1 Source Port: 72

slice: 1 slice_port: 0

is_monitor: 0 lif: 4100 phy_intf_id: 1 vnic_if: 0 learn_en: 1 native_vlan 1

trunk_port: 1 src_if_check: 1

FC info: intf_type = 0x0 port_mode = 0x0 port_vsan = 0 port_speed = 0 fcf_mac = 0:0:0:0:0:0

IFTMC

对于IFTMC，您可以运行2条命令。在运行以下命令之前，请运行attach module 1命令。

```
N9KFX2# attach module 1
module-1# show system internal iftmc info interface e1/1
module-1# show system internal iftmc info vlan 20
```

两个命令

- show system internal iftmc info interface ethernet1/1
- show system internal iftmc info vlan 20

显示与ELTM输出类似的信息，确保您在ELTM和IFTMC模块之间具有相同的值。

Hardware

要确认输出已在硬件上正确编程，您可以运行show hardware internal tah interface ethernet 1/1命令。在该命令的输出中，您可以转到BdStateTable部分。

```
N9KFX2-1(config-if)# show hardware internal tah interface ethernet 1/1
BdStateTable:
ENTRY: 1
  info_leaf_flood_dst_ptr : 1                <<<Displays native vlan
  info_leaf_igmp_mld_dst_ptr : 4098
  info_leaf_fid : 1                          <<<Displays native vlan
  info_leaf_vrf : 1
  info_leaf_igmp_snp_en : 1
  info_leaf_l3_bind_check_en : 1
  info_leaf_v4_omf : 1
  info_leaf_unknown_uc_flood : 1
  info_leaf_unknown_mc_flood : 1
  info_leaf_ecn_mark_en : 1
  info_leaf_l2_bind_check_en : 1
  info_leaf_bd_profile_idx : 1
  info_leaf_analytics_en : 1
  info_leaf_flow_collect_en : 1
  info_spine_info0_gipo_idx : 1              <<<Displays native vlan
  info_spine_info0_scope : 64
  info_spine_info0_spare : 0x00000000:0x00100044
  info_spine_info1_spare : 0x00001000:0x00000000
  info_spine_info2_gipo_idx : 512
  info_spine_info2_scope : 12416
  info_spine_info2_spare : 0x00005000:0x10000000
  info_spine_info3_gipo_idx : 384
  info_spine_info3_spare : 0x00000000:0x00000000
=====
```

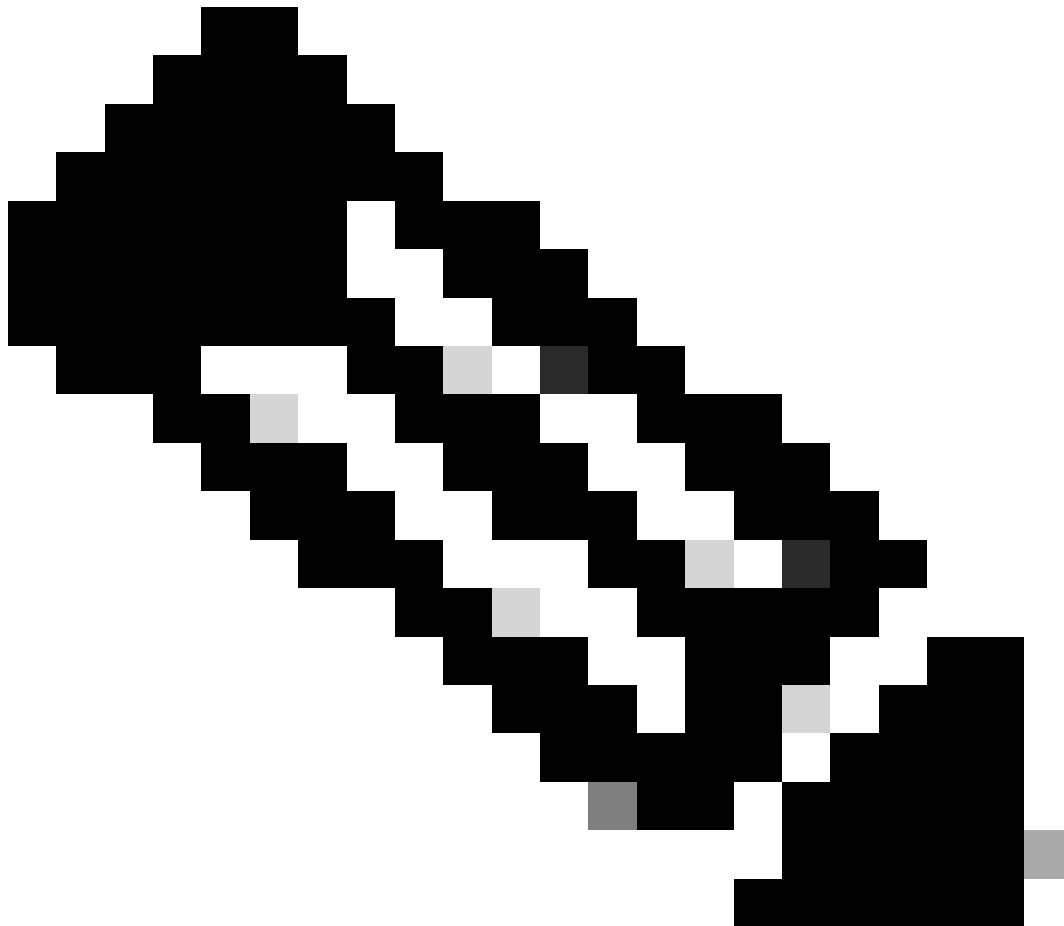
用于收集和检查任何问答问题的有用命令

- show vlan internal info mapping
- show consistency-checker selective-qinq
- show consistency-checker selective-qinq interface <ethX/Y , port-channel X>
- show hardware internal tah interface ethernet x/y

- show hardware internal tah interface all
- show hardware internal tah sdk event-history error
- show system internal iftmc info interface <>

在Nexus平台上，您可以收集show tech-support比一般信息更具体的内容，一般信息提供一系列命令：

- 显示技术详细信息
- show tech vlan
- show tech usd-all
- show tech eltm
- 附加模块1；show tech iftmc



注意：查看Nexus运行版本的相关信息中列出的配置指南的准则和限制部分，确认您的配置未达到任何硬件或软件限制。

相关信息

- [版本9.3.X](#)
- [版本10.1.X](#)
- [版本10.2.X](#)
- [版本10.3.X](#)
- [版本10.4.X](#)
- [思科技术支持和下载](#)

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）。