

Switching

目录

交换机接口配置.....	2
交换机MAC地址表.....	17
Trunk	23
DTP (Dynamic Trunking Protocol)	37
VTP (VLAN Trunking Protocol).....	49
STP (Spanning-Tree Protocol)	95
Common Spanning Tree (CST).....	107
Rapid Spanning Tree Protocol (RSTP)	107
Per-VLAN Spanning-Tree plus (PVST+)	108
Rapid PVST+	110
Multiple Spanning Tree Protocol (MSTP)	111
Spanning-Tree Feature.....	141
Port Fast	141
BPDU Guard	146
BPDU Filtering.....	152
UplinkFast.....	158
BackboneFast	162
Root Guard	166
Loop Guard.....	172
EtherChannel.....	178
Protected Port	190
Port Blocking	197
Port Security.....	198
IP Source Guard.....	210
Security with ACL	221
Port ACL.....	224
Router ACL.....	227
VLAN ACL.....	227
Storm Control.....	231
SPAN and RSPAN.....	234
UDLD (UniDirectional Link Detection).....	243
Fallback Bridging	246

IEEE 802.1x (DOT1X) Authentication	250
交换机故障恢复管理.....	265
交换机密码恢复.....	265
交换机密码恢复管理.....	269

交换机接口配置

因为交换机的特殊性，通常存在多个接口需要做相同的配置，如将多个接口划入相同 VLAN，这时就需要一种能够快速配置接口的方法。

对于 2 层交换机，所有的接口只能工作在二层，而对于三层交换机，接口除了可以工作在二层之外，并且还可以工作在三层，也就是说三层交换机的接口还可以配置 IP 地址，等同于路由器的接口。

配置

快速配置接口

1.快速对多个连续的接口做相同配置

（1）快速进入多个接口

```
Switch(config)#interface range f0/1-3
```

说明：同时进入接口 F0/1，F0/2，F0/3。

（2）配置接口参数

```
Switch(config)#interface range f0/1-3
```

```
Switch(config-if-range)#description ccie
```

说明：当同时进入多个接口后，所做的配置将对所有进入的接口生效。

（3）查看结果

```
Switch#sh run | b inter
```

```
vlan internal allocation policy ascending
```

```
!
```

```
!
```

```
interface FastEthernet0/1
```

```
description ccie
```

```
!
```

```
interface FastEthernet0/2
```

```
description ccie
```

```
!
```

```
interface FastEthernet0/3
```

```
description ccie
```

```
!
```

```
interface FastEthernet0/4
```

```
!
```

```
interface FastEthernet0/5
```

```
!
```

```
interface FastEthernet0/6
```

```
!
```



说明：可以看到配置文件中，之前的配置对 F0/1，F0/2，F0/3 生效，其它没有进入的接口配置保存原状。

2.快速对多个不连续的接口做相同配置

（1）快速进入多个不连续接口

```
Switch(config)#interface range f0/1 - 2 , f0/4 , f0/6 - 7
```

说明：同时进入接口 F0/1，F0/2，F0/4，F0/6，F0/7。

在配置多个不连续接口时，请注意在连字符 - 前后都加上空格，这样可以保证在任何 IOS 版本中输入有效。

（2）配置接口参数

```
Switch(config)#interface range f0/1 - 2 , f0/4 , f0/6 - 7
```

```
Switch(config-if-range)#description cisco
```

说明：当同时进入多个接口后，所做的配置将对所有进入的接口生效。

（3）查看结果

```
Switch#sh run | b inter
```

```
vlan internal allocation policy ascending
```

```
!
```

```
!
```

```
interface FastEthernet0/1
```

```
description cisco
```

```
!
```

```
interface FastEthernet0/2
```

```
description cisco
```

```
!
```

```
interface FastEthernet0/3
```

```
!
```

```
interface FastEthernet0/4
```

```
description cisco
```

```
!
```

```
interface FastEthernet0/5
```

```
!
```

```
interface FastEthernet0/6
```

```
description cisco
```

```
!
```

```
interface FastEthernet0/7
```

```
description cisco
```

```
!
```

```
interface FastEthernet0/8
```

```
!
```

说明：可以看到配置文件中，之前的配置对 F0/1，F0/2，F0/4，F0/6，F0/7 生效，其它没有进入的接口配置保存原状。

3.接口宏（macro）定义

说明：当在交换机的日常管理中，可能需要多次对多个非连续的接口进行管理和配置，而当需要进入多个非连续接口时，输入的命令较为烦琐，所以为了方便，



CCIE LAB认证经验分享千人群：539730342

系统允许人工将多个接口定义成一个组，这个组成为宏（macro），当进入这个宏（macro），就等于进入了组中的所有接口，对宏（macro）做出的配置，将对组中的所有接口生效，即便交换机重启后，所定义的宏（macro）仍旧存在，可以多次使用，直到手工删除。

（1）定义宏（macro）

```
Switch(config)#define interface-range ccie f0/1 - 2 , f0/4 , f0/6 - 7
```

说明：将接口 F0/1，F0/2，F0/4，F0/6，F0/7 放入了宏 ccie 中，对宏 ccie 所做的配置将对 F0/1，F0/2，F0/4，F0/6，F0/7 生效。

（2）配置接口参数

```
Switch(config)#interface range macro ccie
```

```
Switch(config-if-range)#description abc
```

说明：对宏 ccie 所做的配置将对 F0/1，F0/2，F0/4，F0/6，F0/7 生效。

（3）查看结果

```
Switch#sh run | b inter
```

```
vlan internal allocation policy ascending
```

```
!
```

```
!
```

```
interface FastEthernet0/1
```

```
description abc
```

```
!
```

```
interface FastEthernet0/2
```

```
description abc
```

```
!
```

```
interface FastEthernet0/3
```

```
!
```

```
interface FastEthernet0/4
```

```
description abc
```

```
!
```

```
interface FastEthernet0/5
```

```
!
```

```
interface FastEthernet0/6
```

```
description abc
```

```
!
```

```
interface FastEthernet0/7
```

```
description abc
```

```
!
```

```
interface FastEthernet0/8
```

```
!
```

说明：可以看到配置文件中，对宏 ccie 所做的配置将对 F0/1，F0/2，F0/4，F0/6，F0/7 生效，其它接口配置保存原状。

4.配置 SVI（switch virtual interface）接口

说明：三层交换机的物理接口既可以配置为 2 层接口，也可以配置为三层接口，对于这些配置，此文档将跳过不作详细介绍。

除了交换机物理接口外，交换机还可以将某个 VLAN 配置为 3 层接口，称为 SVI

(switch virtual interface) 接口，将 VLAN 配置为 3 层接口的作用在于为 VLAN 内的流量与外部流量提供 3 层路由转发功能，该 VLAN 内所有的流量都应该在同网段，而该 VLAN 的主机网关都应该指向 SVI 接口地址。此时的 SVI 接口，其实也等同于路由器的接口，但是 SVI 接口也只有在状态都为 UP 的时候，才能提供路由功能，一个状态为 down 的 SVI 接口是不能发送数据包的。要将 SVI 接口激活并且变成 UP 状态，必须将一个活动的物理接口划入该 VLAN，当某 VLAN 中没有活动物理接口时，该 VLAN 的 SVI 接口永远将处于 down 状态而不能转发数据。需要大家注意的是，一个 Trunk 接口允许某个 VLAN 通过，就表示该 Trunk 接口属于该 VLAN，也就是说某个 VLAN 被一个活动的 Trunk 接口允许通过时，那么就说明该 VLAN 中存在活动的物理接口，因此，该 VLAN 的 SVI 接口可以变成 UP 状态，也就可以转发数据包。

(1) 创建 SVI 接口，并配置 IP 地址

```
Switch(config)#vlan 2
```

```
Switch(config-vlan)#exit
```

```
Switch(config)#int vlan 2
```

```
Switch(config-if)#ip add 2.2.2.2 255.255.255.0
```

说明：创建 SVI 接口时，必须保证该 VLAN 已经在交换机上存在。

(2) 查看状态

查看 VLAN：

```
Switch#sh vlan
```

VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12

CCIE LAB认证经验分享千人群：539730342

Fa0/13, Fa0/14, Fa0/15, Fa0/16

Fa0/17, Fa0/18, Fa0/19, Fa0/20

Fa0/21, Fa0/22, Fa0/23, Fa0/24

Gi0/1, Gi0/2

2 VLAN0002 active

1002 fddi-default act/unsup

说明：可以看到，VLAN 2 中没有活动的物理接口存在。

查看 Trunk 所允许的 VLAN:

Switch#sh interfaces trunk

Switch#

说明：可以看到，交换机上没有任何 Trunk 接口，也表示 VLAN 2 中没有活动的物理接口存在。

查看 SVI 接口状态:

Switch#sh protocols vlan 2

Vlan2 is up, line protocol is down

Internet address is 2.2.2.2/24

Switch#

说明：和预期的一样，因为 VLAN 2 中没有任何活动物理接口，所以接口状态为 down，并不能提供数据转发。

（3）激活 SVI 接口

```
Switch(config)#int f0/1
```

```
Switch(config-if)#switchport mode access
```

```
Switch(config-if)#switchport access vlan 2
```


```
Switch(config-if)#no shutdown
```

说明：将物理接口 f0/1 划入 VLAN 2，只要 f0/1 状态为 UP，则 VLAN 2 的 SVI 接口便能变为 UP。

（4）再次查看状态

查看 VLAN：

```
Switch#sh vlan
```



VLAN Name	Status	Ports

1 default	active	Fa0/2, Fa0/3, Fa0/4, Fa0/5 Fa0/6, Fa0/7, Fa0/8, Fa0/9 Fa0/10, Fa0/11, Fa0/12, Fa0/13 Fa0/14, Fa0/15, Fa0/16, Fa0/17 Fa0/18, Fa0/19, Fa0/20, Fa0/21 Fa0/22, Fa0/23, Fa0/24, Gi0/1 Gi0/2
2 VLAN0002	active	Fa0/1

说明： VLAN 2 中存在物理接口 F0/1。

查看物理接口 **F0/1** 的状态：

```
Switch#sh protocols f0/1
```

```
FastEthernet0/1 is up, line protocol is up
```

```
Switch#
```

说明： 接口 F0/1 的状态为 UP。

```
Switch#sh protocols vlan 2
```

```
Vlan2 is up, line protocol is up
```

```
Internet address is 2.2.2.2/24
```

```
Switch#
```

说明： 因为 VLAN 2 中存在活动的物理接口 F0/1，所以 VLAN 2 的 SVI 接口状态变成了 UP，并且能够提供数据转发。

（5）通过 Trunk 允许 VLAN 来控制 SVI 接口状态

```
Switch(config)#vlan 3
```

```
Switch(config-vlan)#exit
```

```
Switch(config)#vlan 4
```

```
Switch(config-vlan)#exit
```

CCIE LAB认证经验分享千人群：539730342

```
Switch(config)#int vlan 3
```

```
Switch(config-if)#ip add 3.3.3.3 255.255.255.0
```

```
Switch(config-if)#exit
```

```
Switch(config)#int vlan 4
```

```
Switch(config-if)#ip add 4.4.4.4 255.255.255.0
```


```
Switch(config-if)#exit
```

```
Switch(config)#
```

说明：创建了 VLAN 3 和 VLAN4，并同时创建了相应的 SVI 接口。

查看 VLAN:

```
Switch#sh vlan
```



VLAN Name	Status	Ports

1 default	active	Fa0/2, Fa0/3, Fa0/4, Fa0/5 Fa0/6, Fa0/7, Fa0/8, Fa0/9 Fa0/10, Fa0/11, Fa0/12, Fa0/13 Fa0/14, Fa0/15, Fa0/16, Fa0/17 Fa0/18, Fa0/19, Fa0/20, Fa0/21 Fa0/22, Fa0/23, Fa0/24, Gi0/1

Gi0/2

2	VLAN0002	active	Fa0/1
3	VLAN0003	active	
4	VLAN0004	active	

说明： VLAN 3 和 VLAN4 中没有任何物理接口。

查看 Trunk

Switch#sh int trunk

Switch#

说明： 交换机上也没有任何 Trunk 接口。

查看 SVI 接口：

Switch#sh prot

Switch#sh protocols vlan 3

Vlan3 is up, line protocol is down

Internet address is 3.3.3.3/24

Switch#sh protocols vlan 4

Vlan4 is up, line protocol is down

Internet address is 4.4.4.4/24

Switch#

说明： 可以看见，由于 VLAN3 和 VLAN4 中没有任何活动物理接口，所以 SVI 接口都为 down 状态。

激活 VLAN 3 的 SVI 接口:

```
Switch(config)#int f0/23
```

```
Switch(config-if)#switchport trunk encapsulation dot1q
```

```
Switch(config-if)#switchport mode trunk
```

```
Switch(config-if)#switchport trunk allowed vlan 3
```

```
Switch(config-if)#no shutdown
```

查看 Trunk:

```
Switch#sh interfaces trunk
```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/23	on	802.1q	trunking	1

Port	Vlans allowed on trunk
------	------------------------

Fa0/23	3
--------	---

Port	Vlans allowed and active in management domain
------	---

Fa0/23	3
--------	---

Port	Vlans in spanning tree forwarding state and not pruned
------	--

CCIE LAB认证经验分享千人群：539730342

Fa0/23 3

Switch#

说明：可以看到 Trunk 接口 F0/23 允许 VLAN3 通过。

查看 SVI 接口：

Switch#sh protocols vlan 3

Vlan3 is up, line protocol is up

Internet address is 3.3.3.3/24

Switch#sh protocols vlan 4

Vlan4 is up, line protocol is down

Internet address is 4.4.4.4/24

Switch#

说明：因为 Trunk 允许 VLAN 3 通过，所以 VLAN 3 的 SVI 接口状态已变为 UP，而 VLAN 4 则仍旧为 down。

Switch(config)#int f0/23

Switch(config-if)#switchport trunk allowed vlan 3,4

Switch#sh interfaces trunk

Port	Mode	Encapsulation	Status	Native vlan
Fa0/23	on	802.1q	trunking	1

CCIE LAB认证经验分享千人群：539730342

Port Vlans allowed on trunk

Fa0/23 3-4

Port Vlans allowed and active in management domain

Fa0/23 3-4

Port Vlans in spanning tree forwarding state and not pruned

Fa0/23 3-4

说明：可以看到 Trunk 接口 F0/23 允许 VLAN3 和 VLAN 4 通过。

Switch#sh protocols vlan 3

Vlan3 is up, line protocol is up

Internet address is 3.3.3.3/24

Switch#sh protocols vlan 4

Vlan4 is up, line protocol is up

Internet address is 4.4.4.4/24

Switch#

说明：因为 Trunk 允许 VLAN 3 和 VLAN4 通过，所以 VLAN 3 和 VLAN4 的 SVI 接口状态都已变为 UP。

交换机 MAC 地址表

交换机在转发数据时，需要根据 MAC 地址表来做出相应转发，如果目标主机的 MAC 地址不在表中，交换机将收到的数据包在所有活动接口上广播发送。当交换机上的接口状态变成 UP 之后，将动态从该接口上学习 MAC 地址，并且将学习到的 MAC 地址与接口相对应后放入 MAC 地址表。

交换机的 MAC 地址表除了动态学习之外，还可以静态手工指定，并且在指定 MAC 地址时，还可以指定在某个 VLAN 的某个接口收到相应的 MAC 后，将数据包作丢弃处理。

注：交换机上，一个接口可以对应多个 MAC 地址，地址的数量无上限，但不超过交换机所支持的 MAC 地址最大数量。

一个 MAC 地址可以同时出现在交换机的多个接口上，但此特性并不被所有型号的交换机支持，在某些型号的交换机上，一个 MAC 地址只能出现在一个接口上，如果出现在另外一个接口上，将会报错，并且数据转发也会出错。

1.查看交换机 MAC 地址表

（1）查看接口 F0/1 的 MAC 地址表

```
Switch#sh mac-address-table interface f0/1
```

Mac Address Table

Vlan	Mac Address	Type	Ports
------	-------------	------	-------

2	0013.1a2f.0680	DYNAMIC	Fa0/1
---	----------------	---------	-------

Total Mac Addresses for this criterion: 1

Switch#

说明：交换机从 F0/1 上学习到了 MAC 地址 0013.1a2f.0680，并且说明是动态学习到的。

2.手工静态指定 MAC 地址

（1）手工静态指定 MAC 地址


Switch(config)#mac-address-table static 0013.1a2f.0680 vlan 1 interface f0/2

说明：指定 VLAN 1 的接口 F0/2 的 MAC 地址为 0013.1a2f.0680。

（2）查看接口 F0/2 的 MAC 地址表

Switch#sh mac-address-table interface f0/2

Mac Address Table



Vlan	Mac Address	Type	Ports
1	0013.1a2f.0680	STATIC	Fa0/2
1	0013.1a7f.a4a0	DYNAMIC	Fa0/2

Total Mac Addresses for this criterion: 2

Switch#

说明：接口 F0/2 上除了动态学习到的 MAC 地址之外，还有静态手工指定的地址。

（3）指定丢弃某个 MAC 地址

```
Switch(config)#mac-address-table static 0013.1a2f.0680 vlan 2 drop
```

说明：此配置将使源 MAC 为 0013.1a2f.0680 的数据包在 VLAN 2 被丢弃，但在别的 VLAN 通信正常。

3.MAC 地址老化时间（aging-time）

交换机在一个接口上学习到 MAC 地址之后，该 MAC 与接口的映射并不会永远被保存在 MAC 地址表中，除非是手工静态指定的。当一台主机从某个接口转移后，交换机再将目标 MAC 为该主机的数据从该接口发出去是毫无意义的，所以 MAC 地址在 MAC 地址表中是有最大停留时间的，称为老化时间（aging-time），当相应 MAC 地址在超出老化时间后还没有数据传输时，该 MAC 地址将从表中被清除。默认的 MAC 地址老化时间为 300 秒（5 分钟）。

（1）修改 MAC 地址的老化时间

说明：只能针对 VLAN 作修改

```
Switch(config)#mac-address-table aging-time 60 vlan 1
```

说明：将 VLAN 1 的 MAC 地址老化时间改为 60 秒。

（2）查看 MAC 地址的老化时间

```
Switch#sh mac-address-table aging-time
```

Global Aging Time: 300

Vlan	Aging Time
------	------------

1	60
---	----

2	300
---	-----

3	300
---	-----

4 300

Switch#

说明：可以看到，VLAN 1 的 MAC 地址老化时间为 60 秒，其它 VLAN 保存默认 300 秒。

交换机自身 MAC 地址

以太网中，每一个节点，都需要一个 MAC 地址，而以太网交换机可以与多个终端连接，也就有多个节点，因此，交换机上也会有多个 MAC 地址存在，如交换机的每个接口都有一个 MAC 地址，包含物理接口和 SVI 接口。除此之外，还有一个 MAC 地址是用来表示整台交换机的。

注：都知道 2 层交换机的 VLAN 1 为管理 VLAN，一个表示整台交换机的 MAC 地址通常就是 VLAN 1 的 MAC 地址，但这种情况又需要根据交换机型号而定，并不适用于任何型号的交换机。

某些型号的交换机，所有 VLAN 的 SVI 接口 MAC 地址全部相同，但某些型号却是不同的，但是连续的。

1.查看交换机的 MAC 地址

（1）查看表示整台交换机的 MAC 地址

Switch#sh version

（输出被省略）

512K bytes of flash-simulated non-volatile configuration memory.

Base ethernet MAC Address : 00:1A:6C:6F:FB:00

Motherboard assembly number : 73-9897-06

Power supply part number : 341-0097-02

CCIE LAB认证经验分享千人群：539730342

Motherboard serial number : CAT10475C57

Power supply serial number : AZS104407JE

Model revision number : D0

Motherboard revision number : A0

Model number : WS-C3560-24TS-S

System serial number : CAT1047RJNU

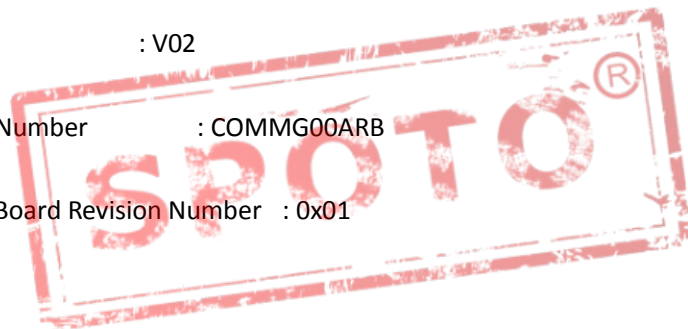
Top Assembly Part Number : 800-26160-02

Top Assembly Revision Number : C0

Version ID : V02

CLEI Code Number : COMMG00ARB

Hardware Board Revision Number : 0x01



Switch	Ports	Model	SW Version	SW Image
-----	-----	-----	-----	-----
* 1	26	WS-C3560-24TS	12.2(35)SE1	C3560-ADVIPSERVICESK

Configuration register is 0xF

Switch#

说明：表示整台交换机的 MAC 地址为 00:1A:6C:6F:FB:00。

（2）查看物理接口的 MAC 地址

```
Switch#sh int f0/1
```

```
FastEthernet0/1 is up, line protocol is up (connected)
```

```
Hardware is Fast Ethernet, address is 001a.6c6f.fb03 (bia 001a.6c6f.fb03)
```

（输出被省略）

```
Switch#sh int f0/2
```

```
FastEthernet0/2 is up, line protocol is up (connected)
```

```
Hardware is Fast Ethernet, address is 001a.6c6f.fb04 (bia 001a.6c6f.fb04)
```

（输出被省略）

```
Switch#sh int f0/3
```

```
FastEthernet0/3 is up, line protocol is up (connected)
```

```
Hardware is Fast Ethernet, address is 001a.6c6f.fb05 (bia 001a.6c6f.fb05)
```

（输出被省略）

说明：可以看到，物理接口的 MAC 地址是连续的，但无论什么型号的交换机，物理接口的 MAC 地址一定是不同的。

（3）查看 SVI 接口的 MAC 地址

CCIE LAB认证经验分享千人群：539730342

```
Switch#sh int vlan 1
```

Vlan1 is up, line protocol is up

Hardware is EtherSVI, address is 001a.6c6f.fb40 (bia 001a.6c6f.fb40)

（输出被省略）

```
Switch#sh int vlan 2
```

Vlan2 is up, line protocol is up

Hardware is EtherSVI, address is 001a.6c6f.fb41 (bia 001a.6c6f.fb41)

（输出被省略）

```
Switch#sh int vlan 3
```

Vlan3 is up, line protocol is up

Hardware is EtherSVI, address is 001a.6c6f.fb42 (bia 001a.6c6f.fb42)

（输出被省略）



说明：可以看到，交换机 SVI 接口的 MAC 地址是连续的，但某些型号的交换机，所有 SVI 接口的 MAC 地址全部是相同的。

Trunk

在交换机上，可以将 access 接口划入各个 VLAN 中，不同 VLAN 的流量是不被交换机转发的。如果能让两个 access 接口互相通信，就必须将这两个接口划入相同的 VLAN 中。

当需要在交换机与交换机之间通信时，连接交换机的链路就可能需要为多个

VLAN 提供数据传输,这样在一条链路上提供多个 VLAN 数据传输的链路,就是 Trunk,进入 Trunk 的数据包被打上标记,写上相应的 VLAN 号,当传输到对端时,则被去掉标记,并且根据 VLAN 号将数据包转发到相应的 VLAN 中。需要说明,在 access 接口上的数据包,是没有 VLAN 号标记的,并且也不允许 VLAN 标记,如果一个 access 接口收到一个带有 VLAN 标记的数据包,是要将数据包丢弃的。

在 Trunk 上为数据包打标记是通过协议来完成的,目前有两种协议可以完成 VLAN 标记工作,分别是 Inter-Switch Link (ISL)和 IEEE 802.1Q,其中 ISL 为思科私有协议。

当 Trunk 使用 ISL 封装时,将对进入 Trunk 的每个 VLAN 的数据包打上标记,当 ISL 收到一个没有标记的数据帧,直接丢弃。ISL 在原始以太网数据帧的基础上,额外加上 26 字节的标记,但最多只支持 1000 个 VLAN,除此之外,ISL 还将对整个数据帧重新计算 FCS,在帧的最后插入 4 字节的新 FCS,也就是说,ISL 会在原始数据帧的基础上再加 30 字节,数据包结构如下:



可以看出,原始以太网帧的大小范围为 64-1518 字节,而 ISL 帧的大小范围为 94-1548 字节。当 ISL Trunk 收到数据帧后,直接去掉 ISL 标记和新 FCS 后,就可马上转发。

当 Trunk 使用 IEEE 802.1Q 封装时,将对除了 Native VLAN 之外的所有 VLAN 打上标记,如果 802.1Q 收到一个没有 VLAN 标记的数据帧,将其在 Native VLAN 内转发,所以请确保 Trunk 两头的 Native VLAN 号是一致的。802.1Q 在原始以太网帧中插入 4

字节的标记，支持 4096 个 VLAN，

数据包结构如下：



可以看出，原始以太网帧的大小范围为 64-1518 字节，而 802.1Q 帧的大小范围为 68-1522 字节。当 802.1Q Trunk 收到数据帧后，去掉 802.1Q 标记之外，还要重新计算 FCS 才有转发。

配置



说明：只有数据帧经过 Trunk 时，才会打上 VLAN 标记，而经过 access 接口的数据帧是没有标记的，当一个 access 接口属于某个 VLAN，那么从此接口收到的数据帧都被认为是此 VLAN 的数据，因此就可与该接口相同 VLAN 的主机通信。下面以上图配置。

1.在交换机上将 access 接口划入相应 VLAN

（1）在 SW1 上做相应配置

```
sw1(config)#vlan 10
```

```
sw1(config-vlan)#exit
```

```
sw1(config)#vlan 20
```

```
sw1(config-vlan)#exit
```

```
sw1(config)#int f0/1
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#switchport access vlan 10
```

```
sw1(config-if)#no shutdown
```

```
sw1(config-if)#exit
```

```
sw1(config)#int f0/2
```

```
sw1(config-if)#switchport mode access
```



```
sw1(config-if)#switchport access vlan 20
```

```
sw1(config-if)#no shutdown
```

```
sw1(config-if)#exit
```

```
sw1(config)#int f0/23
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#switchport access vlan 10
```

```
sw1(config-if)#no shutdown
```

(2) 在 SW2 上做相应配置

```
sw2(config)#vlan 20
```

```
sw2(config-vlan)#exit
```

```
sw2(config)#int range f0/3 , f0/23
```

```
sw2(config-if-range)#switchport mode access
```

```
sw2(config-if-range)#switchport access vlan 20
```

```
sw2(config-if-range)#no shutdown
```

```
sw2(config-if-range)#exit
```

2.配置各路由器

(1) 配置 R1

```
r1(config)#int f0/0
```

```
r1(config-if)#ip add 10.1.1.1 255.255.255.0
```

```
r1(config-if)#no sh
```

(2) 配置 R2

```
r2(config)#int f0/0
```

```
r2(config-if)#ip add 10.1.1.2 255.255.255.0
```

```
r2(config-if)#no sh
```

(3) 配置 R3

```
r3(config)#int f0/1
```

```
r3(config-if)#ip add 10.1.1.3 255.255.255.0
```

```
r3(config-if)#no sh
```



3.测试结果

(1) 测试从 R1 到 R2 的连通性

```
r1#ping 10.1.1.2 repeat 10
```

Type escape sequence to abort.

Sending 10, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

.....

Success rate is 0 percent (0/10)

```
r1#
```

说明：因为从 R1 发送数据包到 10.1.1.2 时，数据包从 SW1 的接口 F0/1 进入，因为 F0/1 属于 VLAN 10，而 F0/2 属于 VLAN 20，所以 SW1 并不会将去往 10.1.1.2 的数据包从接口 F0/2 发出去，所以 R1 到 R2 的通信失败。

（2）测试从 R1 到 R3 的连通性

```
r1#ping 10.1.1.3 repeat 10
```

Type escape sequence to abort.

Sending 10, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:

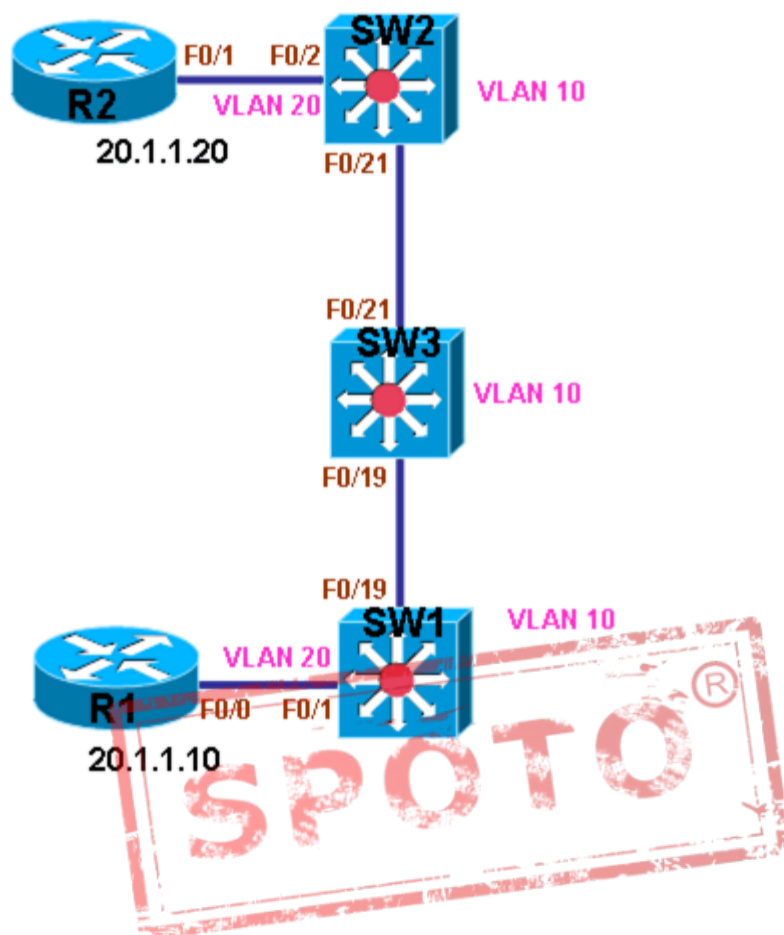
!!!!!!!

Success rate is 100 percent (10/10), round-trip min/avg/max = 1/2/4 ms

```
r1#
```

说明：因为从 R1 发送数据包到 10.1.1.3 时，数据包从 SW1 的接口 F0/1 进入，由于 F0/1 属于 VLAN 10，而 F0/23 也属于 VLAN 10，所以 SW1 将去往 10.1.1.3 的数据包从接口 F0/23 发出去，当 SW2 从 F0/23 收到数据包后，因为没有 VLAN 标记，所以认为数据包是属于 VLAN 20，便将数据包从 F0/3 发出去，最后 R3 收到数据包后，向 R1 回包，最终虽然 R1 和 R3 属于不同的 VLAN，但由于 access 接口没有 VLAN 标记，交换机并不认为是不同 VLAN，所以 R1 与 R3 的通信成功。

Trunk 重点实验



说明：以上图为例，配置实验，本实验在于说明，当一台交换机上的 VLAN 与另外一台交换机的相同 VLAN 通信时，如果中间还有交换机，当中间交换机上没有配置一个相同 VLAN 时，并且无论 Trunk 是否允许该 VLAN 通过，两边的交换机流量无法通过此 VLAN 进行通信，

1.配置交换机

(1) 配置 SW1 的 VLAN 与 Trunk

```
sw1(config)#vlan
```

```
sw1(config)#vlan 10
```

```
sw1(config-vlan)#exit
```

```
sw1(config)#vlan 20
```

```
sw1(config-vlan)#exit
```

```
sw1(config)#int f0/19
```

```
sw1(config-if)#switchport trunk encapsulation dot1q
```

```
sw1(config-if)#switchport mode trunk
```

```
sw1(config-if)#no shutdown
```

```
sw1(config-if)#exit
```

```
sw1(config)#int f0/1
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#switchport access vlan 20
```

```
sw1(config-if)#no shutdown
```

```
sw1(config-if)#exit
```

(2) 配置 SW2 的 VLAN 与 Trunk

```
sw2(config)#vlan 10
```

```
sw2(config-vlan)#exit
```

```
sw2(config)#vlan 20
```

```
sw2(config-vlan)#exit
```

```
sw2(config)#int f0/21
```

```
sw2(config-if)#switchport trunk encapsulation dot1q
```

```
sw2(config-if)#switchport mode trunk
```

```
sw2(config-if)#no shutdown
```

```
sw2(config-if)#exit
```

```
sw2(config)#int f0/2
```

```
sw2(config-if)#switchport mode access
```

```
sw2(config-if)#switchport access vlan 20
```

```
sw2(config-if)#no shutdown
```

```
sw2(config-if)#exit
```

(3) 配置 SW3 的 VLAN 与 Trunk

```
sw3(config)#vlan 10
```

```
sw3(config-vlan)#exit
```

```
sw3(config)#
```

```
sw3(config)#int range f0/19 , f0/21
```

```
sw3(config-if-range)#switchport trunk encapsulation dot1q
```

```
sw3(config-if-range)#switchport mode trunk
```

```
sw3(config-if-range)#no shutdown
```


2.配置 IP

(1) 配置各设备的 IP 地址

SW1:

```
sw1(config)#int vlan 10
```

```
sw1(config-if)#ip address 10.1.1.1 255.255.255.0
```

```
sw1(config-if)#exit
```

```
sw1(config)#int vlan 20
```

```
sw1(config-if)#ip address 20.1.1.1 255.255.255.0
```

```
sw1(config-if)#exit
```

SW2:

```
sw2(config)#int vlan 10
```

```
sw2(config-if)#ip add 10.1.1.2 255.255.255.0
```

```
sw2(config-if)#exit
```

```
sw2(config)#int vlan 20
```

```
sw2(config-if)#ip add 20.1.1.2 255.255.255.0
```

```
sw2(config-if)#exi
```



R1:

```
r1(config)#int f0/0
```

```
r1(config-if)#ip add 20.1.1.10 255.255.255.0
```

```
r1(config-if)#no sh
```

R2:

```
r2(config)#int f0/1
```

```
r2(config-if)#ip add 20.1.1.20 255.255.255.0
```

```
r2(config-if)#no sh
```



3.测试通信

(1) 测试 SW1 的 VLAN 10 到 SW2 的 VLAN 10 的连通性

SW1:

```
sw1#ping 10.1.1.2
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

sw1#

说明：因为 SW1，SW2，SW3 都有 VLAN 10，所以 VLAN 10 从 SW1 到 SW2 是畅通的。

（2）测试 SW1 的 VLAN 20 到 SW2 的 VLAN 20 的连通性

sw1#ping 20.1.1.2

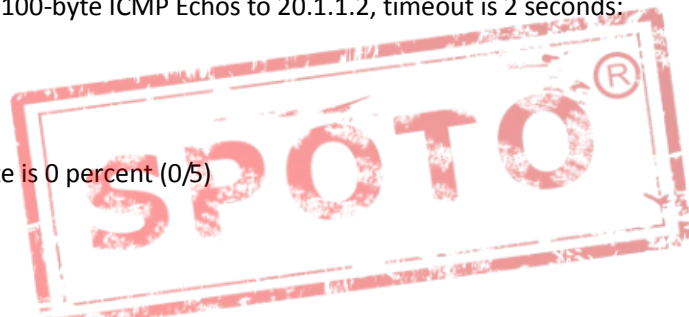
Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 20.1.1.2, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

sw1#



说明：虽然 SW1，SW2 有 VLAN 20，但是 SW1 与 SW2 的 VLAN 20 通信需要穿越 SW3，而 SW3 却没有 VLAN 20，因此 SW3 在自身没有 VLAN 20 的情况下，是不允许 VLAN 20 的流量从自己经过的，所以 SW1 的 VLAN 20 到 SW2 的 VLAN 20 不通。

（3）测试 R1 到 R2 的连通性

r1#ping 20.1.1.20

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 20.1.1.20, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

r1#

说明：虽然 R1 与 R2 都属于 VLAN 20，SW1 与 SW2 都有 VLAN 20，但是 SW3 却没有 VLAN 20，因此 SW3 在自身没有 VLAN 20 的情况下，是不允许 VLAN 20 的流量从自己经过的，所以 R1 到 R2 不通。

4.解决 VLAN 20 通信

说明：因为 SW1 与 SW2 中间的交换机 SW3 没有 VLAN 20，所以穿越 SW3 的 VLAN 20 的流量不能通过，解决方法为在 SW3 上创建 VLAN 20 即可。

(1) 在 SW3 上创建 VLAN 20

sw3(config)#vlan 20

sw3(config-vlan)#exit

sw3(config)#exi

(2) 测试 SW1 的 VLAN 20 到 SW2 的 VLAN 20 的连通性

sw1#ping 20.1.1.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 20.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

```
sw1#
```

说明：因为 SW1 与 SW2 中间的交换机 SW3 已经创建 VLAN 20，所以能够放行 VLAN 20 的流量，最终 SW1 的 VLAN 20 到 SW2 的 VLAN 20 通信正常。

(3) 测试 R1 到 R2 的连通性

```
r1#ping 20.1.1.20
```

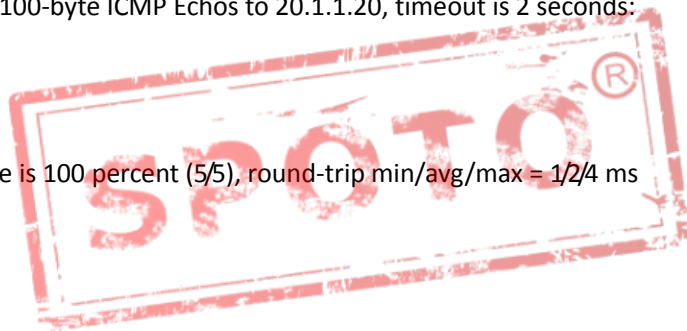
```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 20.1.1.20, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

```
r1#
```



说明：因为 SW1 与 SW2 中间的交换机 SW3 已经创建 VLAN 20，所以能够放行 VLAN 20 的流量，最终 R1 到 R2 的通信正常。

说明：所以基于以上结论，在多台交换机相连时，需要在跨交换机实现 VLAN 通信，即使是同 VLAN，也要解决好连通性问题。

DTP (Dynamic Trunking Protocol)

在需要使用 Trunk 链路时，通常是手工静态配置接口模式，并且手工指定 Trunk 封装协议。然而，当交换机与交换机的接口相连时，多数都需要配置为 Trunk 模式，而连接主机时，都需要配置为 access 模式，为了能够让交换机自动判断什么时候该将接口设置为 Trunk，因此开发出了动态 Trunk 配置协议 (Dynamic Trunking Protocol)，DTP 能够在需要将交换机接口配置为 Trunk 模式时，自动将接口配置为 Trunk，并自动选择 Trunk 封装协议，默认 ISL 优先。

DTP 采用协商的方式来决定是否将接口配置为 Trunk，可配置的接口模式，准确地讲，应该是 3 种，分别为 ON，desirable， auto，下面详细介绍各模式功能：

ON

其实就是手工静态配置为 Trunk，并且还会向对方主动发起 DTP 信息，要求对方也工作在 Trunk 模式，无论对方邻居在什么模式，自己永远工作在 Trunk 模式。

Desirable

此模式为 DTP 主动模式，工作在此模式的接口会主动向对方发起 DTP 信息，要求对方也工作在 Trunk 模式，如果对方回复同意工作在 Trunk 模式，则工作在 Trunk 模式，如果没有 DTP 回复，则工作在 access 模式。

Auto

此模式为 DTP 被动模式，工作在此模式的接口不会主动发起 DTP 信息，只会等待对方主动发起 DTP 信息，如果收到对方的 DTP 信息要求工作在 Trunk 模式，则自己回复对方同意工作在 Trunk 模式，最后的模式为 Trunk，如果 DTP 被动模式收不到 DTP 要求工作在 Trunk 的信息，则工作在 access 模式。

以上三种接口模式都会产生 DTP 信息，ON 和 desirable 是主动产生 DTP 信息，而 auto 是被动生产 DTP 信息，如果手工将接口配置成 Trunk 模式后，可以关闭 DTP 信息以节省资源，关闭 DTP 的模式为 nonegotiate。

注：

Access 模式不是 DTP 的一部分。

开启 DTP 协商的双方都必须在相同的 VTP 域内，否则协商不成功。

CCIE LAB认证经验分享千人群：539730342

交换机的型号不同，默认的 DTP 模式会有所不同，3550 默认为 desirable 模式，3560 默认为 auto 模式。

当收不到对方 DTP 回复时，则选择工作在 access 模式。

接口配置模式与最终工作模式对照表如下：

Port1 administrator mode	Port2 administrator mode	Port1&2 working mode
trunk	trunk	trunk
trunk	dynamic desirable	trunk
trunk	dynamic auto	trunk
trunk	access	fault
access	access	access
access	dynamic desirable	access
access	dynamic auto	access
dynamic desirable	dynamic desirable	trunk
dynamic desirable	dynamic auto	trunk
dynamic auto	dynamic auto	access

配置



1.配置 SW1 为 desirable，SW2 为 Trunk

(1) 配置 DTP

```
sw1(config)#int f0/23
```

```
sw1(config-if)#switchport mode dynamic desirable
```

```
sw2(config)#int f0/23
```

```
sw2(config-if)#switchport trunk encapsulation dot1q
```

```
sw2(config-if)#switchport mode trunk
```

（2）查看结果

```
sw1#sh int f0/23 switchport
```

Name: Fa0/23

Switchport: Enabled

Administrative Mode: dynamic desirable

Operational Mode: trunk

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: dot1q

Negotiation of Trunking: On

（输出被省略）

```
sw1#
```

```
sw2#sh int f0/23 switchport
```

Name: Fa0/23

Switchport: Enabled

Administrative Mode: trunk

Operational Mode: trunk

Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q

Negotiation of Trunking: On

（输出被省略）

sw2#

说明：可以看到，双方接口的 DTP 协商是开启的，因为双方都会主动发起 DTP 要求对方工作在 trunk，所以最终双方的工作模式为 Trunk。

2.配置 SW1 为 desirable，SW2 为 auto

（1）配置 DTP

```
sw1(config)#int f0/23
```

```
sw1(config-if)#switchport mode dynamic desirable
```

```
sw2(config)#int f0/23
```

```
sw2(config-if)#switchport mode dynamic auto
```

（2）查看结果

CCIE LAB认证经验分享千人群：539730342

```
sw1#sh int f0/23 switchport
```

Name: Fa0/23

Switchport: Enabled

Administrative Mode: dynamic desirable

Operational Mode: trunk

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: isl

Negotiation of Trunking: On

（输出被省略）

```
sw1#
```

```
sw2#sh int f0/23 switchport
```

Name: Fa0/23

Switchport: Enabled

Administrative Mode: dynamic auto

Operational Mode: trunk

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: isl

Negotiation of Trunking: On

（输出被省略）

```
sw2#
```



说明：可以看到，双方接口的 DTP 协商是开启的，因为 SW1 会主动发起 DTP 要求对方工作在 trunk，而 SW2 会同意工作在 Trunk，所以最终双方的工作模式为 Trunk，并且封装协议优选 ISL。

3.配置 SW1 为 auto，SW2 为 auto

（1）配置 DTP

```
sw1(config)#int f0/23
```

```
sw1(config-if)#switchport mode dynamic auto
```

```
sw2(config)#int f0/23
```

```
sw2(config-if)#switchport mode dynamic auto
```

（2）查看结果

```
sw1#sh int f0/23 switchport
```

```
Name: Fa0/23
```

```
Switchport: Enabled
```

```
Administrative Mode: dynamic auto
```

```
Operational Mode: static access
```

```
Administrative Trunking Encapsulation: negotiate
```

```
Operational Trunking Encapsulation: native
```

Negotiation of Trunking: On

（输出被省略）

sw1#

sw2#sh int f0/23 switchport

Name: Fa0/23

Switchport: Enabled

Administrative Mode: dynamic auto

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

（输出被省略）

sw2#

说明：可以看到，双方接口的 DTP 协商是开启的，但由于双方都不会主动发起 DTP 要求对方工作在 trunk，所以最终双方的工作模式为 access。

4.配置 SW1 为 desirable，SW2 为 Trunk，并且关闭 DTP（即为 nonegotiate）

（1）配置 DTP

sw1(config)#int f0/23

sw1(config-if)#switchport mo dynamic desirable

```
sw2(config)#int f0/23
```

```
sw2(config-if)#switchport trunk encapsulation dot1q
```

```
sw2(config-if)#switchport mode trunk
```

```
sw2(config-if)#switchport nonegotiate
```

(2) 查看结果

```
sw1#sh int f0/23 switchport
```

Name: Fa0/23

Switchport: Enabled

Administrative Mode: dynamic desirable

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

(输出被省略)

```
sw1#
```

```
sw2#sh int f0/23 switchport
```

Name: Fa0/23

Switchport: Enabled

Administrative Mode: trunk

Operational Mode: trunk

Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q

Negotiation of Trunking: Off

（输出被省略）

sw2#

说明：可以看到，SW1 的 DTP 协商是开启的，而 SW2 的 DTP 协商是关闭的，所以最终 SW1 的接口选择工作在 access 模式，而 SW2 的模式永远都为 Trunk。

5.配置双方都为 **desirable**，但 VTP 不在相同域内

（1）配置 DTP

sw1(config)#vtp domain ccie

sw1(config)#int f0/23

sw1(config-if)#switchport mode dynamic desirable

sw2(config)#vtp domain cisco

sw2(config)#int f0/23

sw2(config-if)#switchport mode dynamic desirable

（2）查看结果

CCIE LAB认证经验分享千人群：539730342

sw1#sh vtp status

VTP Version : 2

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 5

VTP Operating Mode : Server

VTP Domain Name : ccie

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x04 0x98 0x3D 0x1A 0xA5 0x42 0xDC 0x34

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Local updater ID is 0.0.0.0 (no valid interface found)

sw1#

sw1#sh int f0/23 switchport

Name: Fa0/23

Switchport: Enabled

Administrative Mode: dynamic desirable

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

CCIE LAB认证经验分享千人群：539730342

Negotiation of Trunking: On

（输出被省略）

sw1#

sw2#sh vtp status

VTP Version : 2

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 5

VTP Operating Mode : Server

VTP Domain Name : cisco

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x57 0x30 0x6D 0x7A 0x76 0x12 0x7B 0x40

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Local updater ID is 0.0.0.0 (no valid interface found)

sw2#

sw2#sh int f0/23 switchport

Name: Fa0/23



Switchport: Enabled

Administrative Mode: dynamic desirable

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

（输出被省略）

sw2#

说明：可以看到，双方的 DTP 协商都是开启的，并且模式都为 **desirable**，正常情况下，双方最终模式应为 **trunk**，然而，由于双方的 VTP 域名不同，所以 DTP 协商会失败，所以最终双方的工作模式为 **access** 模式。当双方 VTP 域名不匹配时，开启 DTP 协商的接口会有如下提示：

01:14:51: %LINK-3-UPDOWN: Interface FastEthernet0/23, changed state to up

01:14:51: %DTP-5-DOMAINMISMATCH: Unable to perform trunk negotiation on port Fa0/23 because of VTP domain mismatch.

VTP (VLAN Trunking Protocol)

在一个拥有多台交换机的交换网络中，通常会在多台交换机上配置相同的 VLAN，并且也会对多个接口做相同的配置。

对于需要对多个接口做相同的配置，通过快速接口配置，能够轻松实现，提高工作效率。而对于在多台交换机上做相同的 VLAN 配置，则通过 VTP 来实现。

VTP 为了在多台交换机上配置相同的 VLAN，通过将一台交换机的 VLAN 向其它交换机传播的方法来完成，其它交换机在接收到 VLAN 信息后，然后更新自己的 VLAN 数据库，以达到同步。

CCIE LAB认证经验分享千人群：539730342

要将自己的 VLAN 信息发送到网络中，交换机上必须配置 Trunk，IEEE 802.1Q 和 ISL 都支持，通过 Trunk 相连的交换机便能收到对方发来的 VLAN 信息。

VTP 通过域来管理网络中的交换机，任何交换机发出的 VLAN 信息只能在一个域内传播，只有相同域的交换机才能接收此 VLAN 信息，并且根据接收到的 VLAN 信息更新自己的 VLAN 数据库。交换机是否在同一个域，是通过域名来分辨的，比如域名 ccie 与域名 ccie 属于同一个域，而域名 ccie 与域名 cisco 就属于不同的域。默认交换机的域名为空，但是最重点的，需要大家牢记的是，如果自己的域名为空，则表示与任何非空域名相同，也就是说如果对方有域名，而自己却没有域名，则自己和对方属于相同的域。

在 VTP 中，交换机分三种模式：Server、Client、Transparent，他们的功能分别如下：

Server:

可以创建，更改和删除 VLAN，可以更改任何 VTP 参数，可以将自己的 VLAN 信息向网络中发送，并且也会根据收到的 VLAN 信息来选择是否同步自己的 VLAN 数据库。

Client:

不能创建，更改和删除 VLAN，但是可以更改部分 VTP 参数，也可以将自己的 VLAN 信息向网络中发送，并且也会根据收到的 VLAN 信息来选择是否同步自己的 VLAN 数据库。

Transparent:

可以创建，更改和删除 VLAN，可以更改任何 VTP 参数，不会将自己的 VLAN 信息向网络中发送，但是会转发接收到其它交换机发来的 VLAN 信息，并且不会根据收到的 VLAN 信息来同步自己的 VLAN 数据库。

CCIE LAB认证经验分享千人群：539730342

从上可以看出，Server 与 Client 的唯一区别在于，Server 可以随意修改自己的 VLAN 信息和 VTP 参数，而 Client 则不能，除此之外，其它完全相同。

Server 与 Transparent 的区别在于，Transparent 不会将自己的 VLAN 信息发送到网络中，并且也不会向别人同步自己的 VLAN 数据库。

所以最终的结论是，如果希望从网络中接收 VLAN 信息来同步自己的 VLAN 数据库，配置成 Server 与 Client 都可以实现，要将自己的 VLAN 信息发送到网络中，Server 与 Client 也都能实现。如果要具有修改 VLAN 数据库的权限，只有 Server 与 Transparent 能做到，Client 是不能自己更改 VLAN 数据库的。

Server 与 Client 发出的 VLAN 信息，都有一个 configuration revision 号码，每修改一次 VLAN 信息，configuration revision 号则加 1，如果做相同操作，configuration revision 号是不会有变化的。configuration revision 号越高（数字越大），则说明 VLAN 信息越新。

Server 与 Client 从网络中接收到 VLAN 信息后，是否根据此信息同步自己的 VLAN 数据库，则要将自己的 VLAN 信息与接收到的作对比，如果接收到的 VLAN 信息的 configuration revision 号比自己的大，则将自己的 VLAN 数据库与接收到的进行同步，如果 configuration revision 号比自己的小或者相等，则放弃同步。域中总是先使用 configuration revision 号码最高的 VLAN 信息

默认情况下，交换机的域名为空，无论是 Server 还是 Client，在空域名的情况下，是不会将自己的 VLAN 信息往外发的，但是在域名为空的情况下，无论收到任何 VLAN 信息，只要 configuration revision 号比自己的大，就会同步自己的 VLAN 数据库，并且添加上相同的域名。域名在配置之后，只能更改，但不能删除。如果网络中全是 Client，可想而知就不要配置域名了。

在谈及 VTP，不得不详细解释 VLAN，交换机所支持的 VLAN 数为 1-4094，VLAN 1-1005 称为 Normal VLAN，VLAN 1006 - 4094 称为 Extended VLAN。Normal VLAN (1-1005) 是保存在 VLAN 数据库中的，也就是 vlan.dat，而 Extended VLAN(1006-4094) 是保存在 startup-config 中的。Normal VLAN (1-1005) 可以随意配置，而 Extended VLAN(1006-4094) 只能在 VTP 模式为 Transparent 时才能配置。所以，VTP 只能将 Normal

CCIE LAB认证经验分享千人群：539730342

VLAN(1-1005)在网络中更新。当同时配置了 1-1005 的 VLAN 和 1006-4096 的 VLAN，在删除 vlan.dat 后，1-1005 的 VLAN 会被删除，但 1006-4096 的 VLAN 还在，如果删除了 startup-config，那么则会删除 1006-4096 的 VLAN，但不会影响 1-1005 的 VLAN。

VTP 现有两个版本，ver 1 和 ver 2，默认为 ver 1，因为 Transparent 会转发接收到其它交换机发来的 VLAN 信息，但是当自己的 VTP 版本为 ver 1 时，只有自己接收到的 VLAN 信息的域名和 VTP 版本与自己的相同，才会转发，但如果自己为 ver 2，则无论收到任何 VLAN 信息都会转发。

如果域中一台交换机开了 VTP ver 2，则应该全部都要打开，但是只有 Server 和 Transparent 才能更改 VTP 版本，而 Client 会根据收到的 VLAN 信息同步自己的 VTP 版本。

交换机还可以为 VTP 配置密码，当配置密码后，即使 VTP 域名相同，如果密码不同，也不能根据接收到的 VLAN 信息更新自己的 VLAN 数据库。要确认 VTP 密码是否相同，双方的 MD5 digest 值必须相同。

附:在交换机最新的 IOS 版本中,如果 3560 的 12.2(52)SE ,已经加入对 VTP version 3 的支持,最大的特点就是,可以在 VTP 信息中传递 Extended VLAN(1006-4094),但改为 Ver 3 之后,不能再切换到 Ver 1 和 Ver 2.

重点说明:

★ 交换机的配置信息保存在 nvram 存储器的 startup-config 文件中。

★ 而 Flash 中的文件 config.text 与 nvram 存储器的 startup-config 文件完全相同，删除任何一个，即同时删除两个。（注：此规则不完全适用于高端交换机）

★交换机的 Normal VLAN（1-1005）是保存在文件 vlan.dat 中，而 Extended VLAN(1006-4094)是保存在 nvram 存储器的文件 startup-config 中。

★ VTP 信息全部保存在 vlan.dat 中。

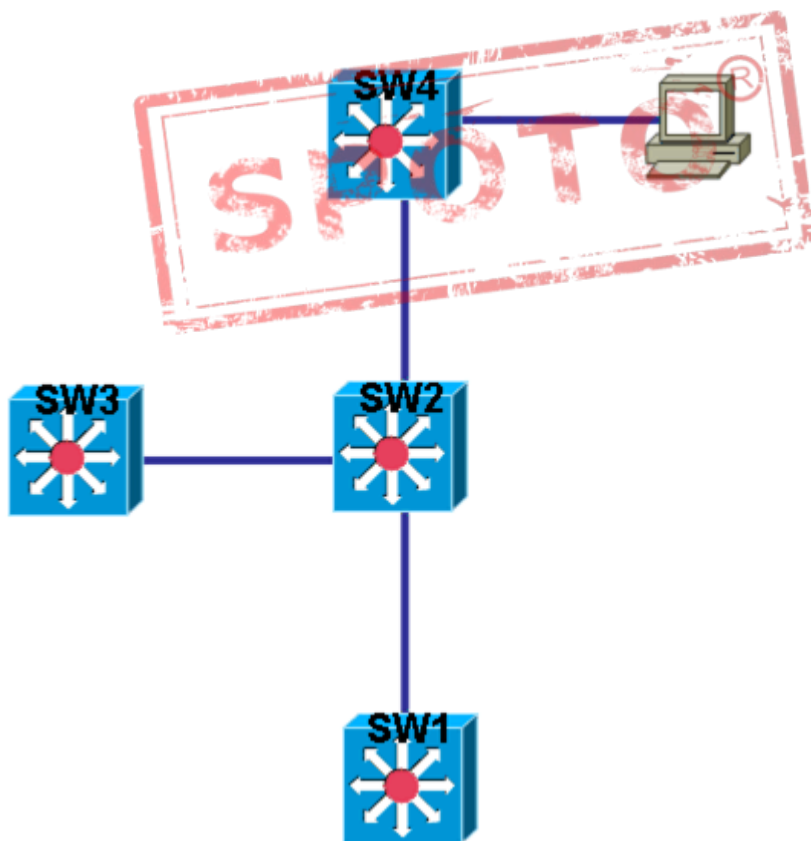
★ 当 VTP 模式为 Transparent 时,所有 VLAN 信息和 VTP 信息除了保存在 vlan.dat 中之外，还会保存在 nvram 存储器的 startup-config 中。

★ 当 VTP 模式为 Server 和 Client 时 所有 VLAN 信息和 VTP 信息只保存在 vlan.dat

中，不会保存在 nvram 存储器的 startup-config 中，所以 show running-config 时，也是看不到 VLAN 信息的。

- ★ 域名为空的交换机是不会发送任何 VTP 信息的。
- ★ 将模式改为 Transparent，可以清除所有 VTP 信息。

VTP Pruning



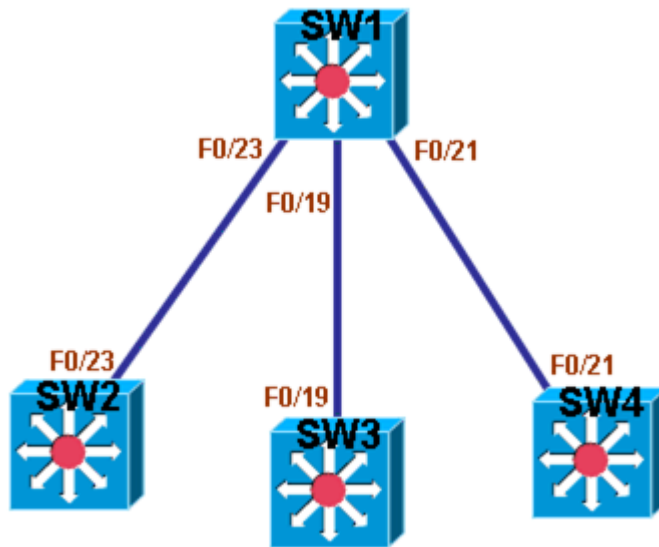
如上图所示，当交换机 SW1 收到 broadcast, multicast 以及 unknown unicast 后，会在所有 Trunk 上进行广播发送，最终结果造成 SW2 会转发给 SW3，也会转发给 SW4，而只有 SW4 上接有终端，也就是说只有 SW4 需要接收这些广播，对于 SW3，转发这些广播是毫无意义的，因为自己没有连接终端。

对于上述情况，当一台交换机在某 VLAN 进入广播发送数据时，流量应该只被发送到在此 VLAN 连接了终端的交换机，而对于没有连接终端的交换机，很明显，是没有必须接收这样的广播了。为了节省带宽，提高网络性能，VTP Pruning 限制交换机只将广播发送到连接了终端的交换机。如果上图中开启了 VTP Pruning，则 SW1 发出的广播只会被发送到 SW2，再转发到 SW4，而不会转发到 SW3。

在 Trunk 上，只有某 VLAN 允许被剪除，那么在此 VLAN 的广播才不会发到没有连接终端的交换机，如果不允许剪除，则广播照常。允许被剪除的 VLAN 范围是 2-1001，而 VLAN1 和 1002-1005 以及 1006-4094 是不能被剪除的，开启 VTP Pruning 后，默认 VLAN2-1001 被剪除，但剪除的 VLAN 号可以在 Trunk 上随意定义。

VTP 模式为 Transparent 时，是不支持 VTP Pruning 的，但无论支持 VTP Ver 1 还是 Ver 2 都支持 VTP Pruning。

配置



说明：以上图为例，配置 VTP。第一部分为验证交换机文件系统，第二部分为验证 VTP。

第一部分 （验证交换机文件系统）

1.在 SW1 上配置 VTP

（1）创建 vlan 2000，vlan 3000

```
sw1(config)#vlan 2000
```

```
sw1(config-vlan)#exit
```

```
% Failed to create VLANs 2000
```

```
Extended VLAN(s) not allowed in current VTP mode.
```

%Failed to commit extended VLAN(s) changes.

sw1(config)#

说明：因为交换机默认为 Server 模式，所以不能创建 Extended VLAN(1006-4094)。

(2) 在 VTPTransparent 下创建 vlan 2000, vlan 3000

sw1(config)#vtp domain ccie

sw1(config)#vtp mode transparent

Setting device to VTP TRANSPARENT mode.

sw1(config)#vlan 2000

sw1(config-vlan)#exit

sw1(config)#vlan 3000

sw1(config-vlan)#exit

sw1(config)#

说明：Vlan 2000 在 transparent 模式下创建成功。

(3)查看 VLAN

sw1#sh vlan

VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8

Fa0/9, Fa0/10, Fa0/11, Fa0/12

Fa0/13, Fa0/14, Fa0/15, Fa0/16

Fa0/17, Fa0/18, Fa0/23, Fa0/24

Gi0/1, Gi0/2

1002 fddi-default act/unsup

1003 token-ring-default act/unsup

1004 fddinet-default act/unsup

1005 trnet-default act/unsup

2000 VLAN2000 active

3000 VLAN3000 active

（输出被省略）



sw1#

说明：Vlan 2000 在 transparent 模式下创建成功。

（4）在 SW1 上创建 VLAN 2-5，以及 VLAN 3000

sw1(config)#vlan 2

sw1(config-vlan)#exit

sw1(config)#vlan 3

sw1(config-vlan)#exit

CCIE LAB认证经验分享千人群：539730342

```
sw1(config)#vlan 4
```

```
sw1(config-vlan)#exit
```

```
sw1(config)#vlan 5
```

```
sw1(config-vlan)#exit
```

(5)保存并查看

保存：

```
sw1#wr
```

```
Building configuration...
```

```
[OK]
```

```
sw1#
```



查看 VLAN：

```
sw1#sh vlan
```

VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4
		Fa0/5, Fa0/6, Fa0/7, Fa0/8
		Fa0/9, Fa0/10, Fa0/11, Fa0/12
		Fa0/13, Fa0/14, Fa0/15, Fa0/16

Fa0/17, Fa0/18, Fa0/19, Fa0/20

Fa0/21, Fa0/22, Fa0/23, Fa0/24

Gi0/1, Gi0/2

2 VLAN0002 active

3 VLAN0003 active

4 VLAN0004 active

5 VLAN0005 active

1002 fddi-default act/unsup

1003 token-ring-default act/unsup

1004 fddinet-default act/unsup

1005 trnet-default act/unsup

2000 VLAN2000 active

3000 VLAN3000 active

(输出被省略)

sw1#

查看 VTP:

sw1#sh vtp sta

VTP Version : 2

Configuration Revision : 0

CCIE LAB认证经验分享千人群：539730342

Maximum VLANs supported locally : 1005

Number of existing VLANs : 9

VTP Operating Mode : Transparent

VTP Domain Name : ccie

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x63 0xE7 0xF7 0x4B 0xFD 0xED 0x17 0xAA

Configuration last modified by 0.0.0.0 at 3-1-93 00:02:01

sw1#

说明： VLAN 创建成功，VTP 也修改成功。

（6）查看文件系统

sw1#dir flash:

Directory of flash:/

2	-rwx	7457899	Mar 1 1993 06:35:16	
+00:00 c3550-ipservicesk9-mz.122-35.SE3.bin				
3	-rwx	796	Mar 1 1993 00:02:44 +00:00	vlan.dat
4	-rwx	0	Mar 1 1993 05:57:14 +00:00	env_vars
5	-rwx	24	Mar 1 1993 05:57:14 +00:00	system_env_vars
6	-rwx	2416	Mar 1 1993 00:03:10 +00:00	config.text

CCIE LAB认证经验分享千人群：539730342

```
7 -rw-      24  Mar 1 1993 00:03:10 +00:00  private-config.text
```

15998976 bytes total (8535040 bytes free)

sw1#dir nv

sw1#dir nvram:

Directory of nvram:/

```
380 -rw-      2416      <no date>  startup-config
```

```
381 ----      24      <no date>  private-config
```

393216 bytes total (390724 bytes free)

sw1#

说明：存在 VLAN 信息和 VTP 信息的 vlan.dat 已经生成；nvram 中的 startup-config 也已经生成，相应的 config.text 也已经生成。

（7）共享文件系统

sw1(config)#int vlan 1

sw1(config-if)#ip add 1.1.1.1 255.255.255.0

sw1(config)#tftp-server flash:vlan.dat

sw1(config)#tftp-server flash:config.text

CCIE LAB认证经验分享千人群：539730342

```
sw1(config)#tftp-server nvram:startup-config
```

说明：交换机已经将 vlan.dat， config.text， startup-config 通过 TFTP 在网络中共享。

2.通过 SW2 验证 SW1 的 vlan.dat

(1) 查看当前 VTP 和 VLAN

```
sw2#sh vtp status
```

VTP Version : 2

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 5

VTP Operating Mode : Server

VTP Domain Name :

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x57 0xCD 0x40 0x65 0x63 0x59 0x47 0xBD

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Local updater ID is 1.1.1.2 on interface Vl1 (lowest numbered VLAN interface found)

```
sw2#
```

```
sw2#sh vlan
```

VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/24, Gi0/1 Gi0/2
1002 fddi-default	act/unsup	
1003 token-ring-default	act/unsup	
1004 fddinet-default	act/unsup	
1005 trnet-default	act/unsup	

（输出被省略）

sw2#

说明：SW2 的 VLAN 和 VTP 为默认配置。

（2）复制 SW1 的 vlan.dat

sw2(config)#int vlan 1

CCIE LAB认证经验分享千人群：539730342

```
sw2(config-if)#ip add 1.1.1.2 255.255.255.0
```

```
sw2#copy tftp: flash:
```

```
Address or name of remote host []? 1.1.1.1
```

```
Source filename []? vlan.dat
```

```
Destination filename [vlan.dat]?
```

```
Accessing tftp://1.1.1.1/vlan.dat...
```

```
Loading vlan.dat from 1.1.1.1 (via Vlan1): !
```

```
[OK - 796 bytes]
```

```
796 bytes copied in 0.032 secs (24875 bytes/sec)
```

```
sw2#
```

说明：SW1 的 vlan.dat 已经被 SW2 复制，接下来可以验证 vlan.dat 中的内容。

（3）查看 SW2 复制的 SW1 的 vlan.dat

```
sw2#dir flash:
```

```
Directory of flash:/
```

2	-rwx	7457899	Mar 1 1993	06:33:13	
+00:00 c3550-ipservicesk9-mz.122-35.SE3.bin					
3	-rwx	796	Mar 1 1993 00:10:41 +00:00		vlan.dat
4	drwx	0	Mar 1 1993 02:51:43 +00:00		test


```
7 -rwx      0  Mar 1 1993 01:52:09 +00:00  system_env_vars
8 -rwx      0  Mar 1 1993 01:52:09 +00:00  env_vars
```

15998976 bytes total (8538624 bytes free)

sw2#

说明：可以看到 vlan.dat 与 SW1 的 vlan.dat 相同。

(4)在 SW2 上使用 SW1 的 vlan.dat

说明：因为 SW1 的 vlan.dat 已经复制到 SW2 的 flash 中，所以重启 SW2 后，便可读取其中的内容。

重启 SW2 后，查看 VLAN 信息和 VTP 信息：

查看 VLAN 信息：

Sw2#sh vlan

VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20

Fa0/21, Fa0/22, Fa0/23, Fa0/24

Gi0/1, Gi0/2

2 VLAN0002 active

3 VLAN0003 active

4 VLAN0004 active

5 VLAN0005 active

1002 fddi-default act/unsup

1003 token-ring-default act/unsup

1004 fddinet-default act/unsup

1005 trnet-default act/unsup

(输出被省略)

Sw2#

查看 VTP 信息:

Sw2#sh vtp sta

VTP Version : 2

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 9

VTP Operating Mode : Transparent

CCIE LAB认证经验分享千人群：539730342

VTP Domain Name : ccie

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x63 0xE7 0xF7 0x4B 0xFD 0xED 0x17 0xAA

Configuration last modified by 0.0.0.0 at 3-1-93 00:02:01

Sw2#

说明：可以验证，vlan.dat 中只有 1-1005 的 VLAN，并且 VTP 信息保存在 vlan.dat 中。

3.通过 SW3 验证 SW1 的 startup-config

(1) 查看 SW3 当前的 startup-config

sw3#dir nvram:

Directory of nvram:/

382	-rw-	0	<no date>	startup-config
-----	------	---	-----------	----------------

383	----	0	<no date>	private-config
-----	------	---	-----------	----------------

393216 bytes total (393164 bytes free)

sw3#

说明：SW3 当前的 startup-config 为空。

(2) 复制 SW1 的 startup-config

```
sw3(config)#int vlan 1
```

```
sw3(config-if)#ip add 1.1.1.3 255.255.255.0
```

```
sw3#copy tftp: flash:
```

```
Address or name of remote host [1.1.1.1]?
```

```
Source filename [startup-config]?
```

```
Destination filename [startup-config]?
```

```
Accessing tftp://1.1.1.1/startup-config...
```

```
Loading startup-config from 1.1.1.1 (via Vlan1): !
```

```
[OK - 2416 bytes]
```

```
2416 bytes copied in 0.088 secs (27455 bytes/sec)
```

```
sw3#
```

说明：SW1 的 startup-config 已经被 SW3 复制，接下来可以验证 startup-config 中的内容。

(3) 在 SW3 上导入复制的 SW1 的 startup-config

```
sw3#copy flash:startup-config running-config
```

```
Destination filename [running-config]?
```

```
Failed to generate persistent self-signed certificate.
```

Secure server will use temporary self-signed certificate.

2416 bytes copied in 0.416 secs (5808 bytes/sec)

sw1#

说明：因为使用了 SW1 的 startup-config，所以主机名也变成了 SW1。

（4）查看 VLAN 与 VTP 信息

查看 VLAN 信息：

sw1#sh vlan

VLAN Name	Status	Ports
1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gi0/1, Gi0/2
2 VLAN0002	active	
3 VLAN0003	active	

CCIE LAB认证经验分享千人群：539730342

```
4  VLAN0004          active
5  VLAN0005          active

1002 fddi-default     act/unsup
1003 token-ring-default  act/unsup
1004 fddinet-default  act/unsup
1005 trnet-default    act/unsup

2000 VLAN2000        active
3000 VLAN3000        active
```

（输出被省略）

sw1#

查看 VTP:

```
sw1#sh vtp status
```

VTP Version : 2

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 9

VTP Operating Mode : Transparent

VTP Domain Name : ccie

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled



CCIE LAB认证经验分享千人群：539730342

VTP Traps Generation : Disabled

MD5 digest : 0x63 0xE7 0xF7 0x4B 0xFD 0xED 0x17 0xAA

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

sw1#

说明：SW3 上除了拥有 VLAN 1-1005 外，1006-4094 的 VLAN 也存在，说明在 Transparent 模式下，VLAN 信息不仅保存在 vlan.dat 中，还保存在 startup-config 中，并且 VTP 也成功保存在 startup-config 中。

4.通过 SW4 验证 SW1 的 config.text

(1) 从 SW4 复制 SW1 的 config.text

sw4(config)#int vlan 1

sw4(config-if)#ip address 1.1.1.4 255.255.255.0

sw4#copy tftp: flash:

Address or name of remote host []? 1.1.1.1

Source filename []? config.text

Destination filename [config.text]?

Accessing tftp://1.1.1.1/config.text...

Loading config.text from 1.1.1.1 (via Vlan1): !

[OK - 2416 bytes]

2416 bytes copied in 0.052 secs (46462 bytes/sec)

sw4#

说明： SW1 的 vlan.dat 已经被 SW4 复制，接下来可以验证 config.text 中的内容。


(2) SW4 上使用 SW1 的 config.text

说明： 因为 SW4 上没有保存配置文件，但拥有了 SW1 的 config.text，所以重启后，就会读取 config.text 的配置，重启后，SW1 的 config.text 内容就被验证

重启 SW4，查看结果：

查看 VLAN:

sw1#sh vlan



VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gi0/1, Gi0/2
2 VLAN0002	active	


```
3  VLAN0003          active
4  VLAN0004          active
5  VLAN0005          active
1002 fddi-default     act/unsup
1003 token-ring-default act/unsup
1004 fddinet-default  act/unsup
1005 trnet-default    act/unsup
2000 VLAN2000        active
3000 VLAN3000        active
```

（输出被省略）

sw1#

查看 VTP:

```
sw1#sh vtp status
```

VTP Version : 2

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 9

VTP Operating Mode : Transparent

VTP Domain Name : ccie

VTP Pruning Mode : Disabled



CCIE LAB认证经验分享千人群：539730342

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x63 0xE7 0xF7 0x4B 0xFD 0xED 0x17 0xAA

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

sw1#

说明：因为使用了 SW1 的 config.text，所以主机名也变成了 SW1。并且 VLAN 与 VTP 与 SW1 完全相同，说明 config.text 与 startup-config 完全相同。

5.在 SW1 上验证 VLAN 存放位置

(1) 在 SW1 上删除 startup-config

说明：由于删除 vlan.dat 是没有用的，因为 Transparent 会将所有 VLAN，如 VLAN 1006-4094 存放在 startup-config 中，即使删了 vlan.dat，所有内容还存在，所以直接删除 startup-config 来测试：

sw1#erase nvram:

Erasing the nvram filesystem will remove all configuration files! Continue? [confirm]

[OK]

Erase of nvram: complete

sw1#

sw1#dir nvram:

Directory of nvram:/

```
382 -rw-      0          <no date> startup-config
383 ----      0          <no date> private-config
```

393216 bytes total (393164 bytes free)

SW

说明：startup-config 已经为空，因为已被删除。

(2) 重启 SW1 后查看结果：

查看 VLAN：

Switch#sh vlan

VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4
		Fa0/5, Fa0/6, Fa0/7, Fa0/8
		Fa0/9, Fa0/10, Fa0/11, Fa0/12
		Fa0/13, Fa0/14, Fa0/15, Fa0/16
		Fa0/17, Fa0/18, Fa0/19, Fa0/20

Fa0/21, Fa0/22, Fa0/23, Fa0/24

Gi0/1, Gi0/2

2 VLAN0002 active

3 VLAN0003 active

4 VLAN0004 active

5 VLAN0005 active

1002 fddi-default act/unsup

1003 token-ring-default act/unsup

1004 fddinet-default act/unsup

1005 trnet-default act/unsup

(输出被省略)

Switch#

查看 VTP:

Switch#sh vtp status

VTP Version : 2

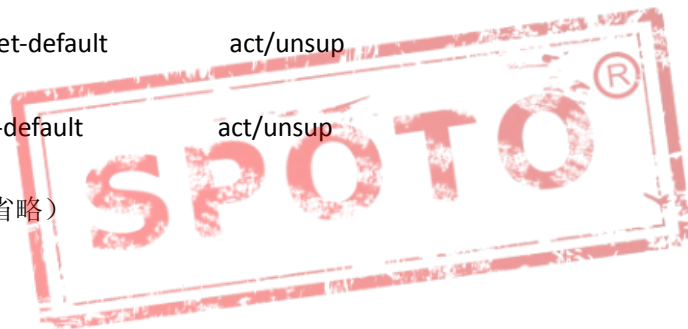
Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 9

VTP Operating Mode : Transparent

VTP Domain Name : ccie



VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

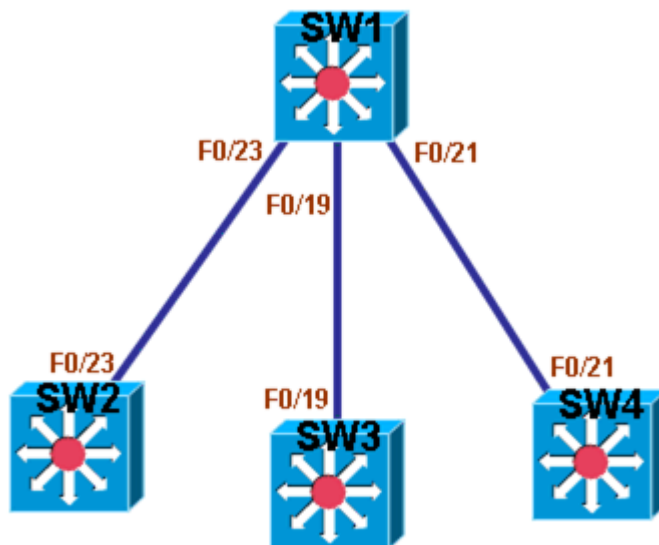
MD5 digest : 0x63 0xE7 0xF7 0x4B 0xFD 0xED 0x17 0xAA

Configuration last modified by 0.0.0.0 at 3-1-93 00:02:01

Switch#

说明：因为 VTP 信息和 VLAN 1-1005 存放在 vlan.dat 中，所以删除了 startup-config，只是删除了 VLAN 1006-4094，而 VTP 信息和 VLAN 1-1005 仍旧存在。

第二部分（验证 VTP）



说明：还是以上图为例，验证 VTP

1.关闭交换机上所有端口

(1) 在所有交换机上关闭所有端口

```
int range f0/1 - 24
```

```
shutdown
```

2.查看默认 VTP

(1) 所有交换机上，默认 VTP 如下：

```
switch#sh vtp status
```

```
VTP Version          : 2
```

```
Configuration Revision : 0
```

```
Maximum VLANs supported locally : 1005
```

```
Number of existing VLANs : 5
```

```
VTP Operating Mode    : Server
```

```
VTP Domain Name      :
```

```
VTP Pruning Mode      : Disabled
```

```
VTP V2 Mode          : Disabled
```



VTP Traps Generation : Disabled

MD5 digest : 0x57 0xCD 0x40 0x65 0x63 0x59 0x47 0xBD

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Local updater ID is 0.0.0.0 (no valid interface found)

switch#

说明：默认 VTP 域名为空，且默认模式为 Server。

3.配置 SW1 的 VTP

（1）在 SW1 上配置 VLAN

sw1(config)#vlan 3

sw1(config-vlan)#exit

sw1(config)#vlan 5

sw1(config-vlan)#exit

sw1(config)#vlan 7

sw1(config-vlan)#exit

sw1(config)#vlan 9

sw1(config-vlan)#exit

sw1(config)#vtp domain ccie

说明：SW1 上的 VLAN 为 3 5 7 9，全部是奇数，VTP 域名为 ccie。

（2）查看 SW1 的 VTP 信息

CCIE LAB认证经验分享千人群：539730342

sw1#sh vtp status

VTP Version : 2

Configuration Revision : 4

Maximum VLANs supported locally : 1005

Number of existing VLANs : 9

VTP Operating Mode : Server

VTP Domain Name : ccie

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x4C 0x22 0xDD 0xCA 0x61 0xA4 0x7C 0x65

Configuration last modified by 0.0.0.0 at 3-1-93 00:04:19

Local updater ID is 0.0.0.0 (no valid interface found)

sw1#

说明：现在 SW1 的 VTP 模式为 Server，域名为 ccie，Configuration Revision 为 4。

（3）查看 SW1 的 VLAN 信息

sw1#sh vlan

VLAN Name	Status	Ports
-----------	--------	-------

CCIE LAB认证经验分享千人群：539730342

```
1 default active Fa0/1, Fa0/2, Fa0/3, Fa0/4  
  
Fa0/5, Fa0/6, Fa0/7, Fa0/8  
  
Fa0/9, Fa0/10, Fa0/11, Fa0/12  
  
Fa0/13, Fa0/14, Fa0/15, Fa0/16  
  
Fa0/17, Fa0/18, Fa0/19, Fa0/20  
  
Fa0/21, Fa0/22, Fa0/23, Fa0/24  
  
Gi0/1, Gi0/2
```

```
3 VLAN0003 active
```

```
5 VLAN0005 active
```

```
7 VLAN0007 active
```

```
9 VLAN0009 active
```

```
1002 fddi-default act/unsup
```

```
1003 token-ring-default act/unsup
```

```
1004 fddinet-default act/unsup
```

```
1005 trnet-default act/unsup
```

（输出被省略）

```
sw1#
```

说明：SW1 上的 VLAN 为 1 3 5 7 9，全部奇数。

4.配置 SW2 的 VTP

（1）在 SW2 上配置 VLAN

```
sw2(config)#vlan 2
```

```
sw2(config-vlan)#exi
```

```
sw2(config)#vlan 4
```

```
sw2(config-vlan)#exi
```

```
sw2(config)#vlan 6
```

```
sw2(config-vlan)#exi
```

```
sw2(config)#vlan 8
```

```
sw2(config-vlan)#exi
```

```
sw2(config)#vlan 10
```

```
sw2(config-vlan)#exi
```

```
sw2(config)#vlan 12
```

```
sw2(config-vlan)#exit
```

```
sw2(config)#vtp domain ccie
```

```
sw2(config)#vtp mode client
```

说明：SW2 上的 VLAN 为 2 4 6 8 10 12 ，全部是偶数，VTP 域名为 ccie，并且模式为 Client。

（2）查看 SW2 的 VTP 信息

```
查看 VTPsw2#sh vtp status
```

```
VTP Version : 2
```

```
Configuration Revision : 6
```

```
Maximum VLANs supported locally : 1005
```

CCIE LAB认证经验分享千人群：539730342

Number of existing VLANs : 11

VTP Operating Mode : Client

VTP Domain Name : ccie

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x5E 0x0C 0x19 0x2B 0xC3 0x13 0x05 0x4F

Configuration last modified by 0.0.0.0 at 3-1-93 00:05:49

sw2#

说明：现在 SW2 的 VTP 模式为 Client，域名为 ccie，Configuration Revision 为 6。

(3) 查看 SW2 的 VLAN 信息

sw2#sh vlan

VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4
		Fa0/5, Fa0/6, Fa0/7, Fa0/8
		Fa0/9, Fa0/10, Fa0/11, Fa0/12
		Fa0/13, Fa0/14, Fa0/15, Fa0/16
		Fa0/17, Fa0/18, Fa0/19, Fa0/20

Fa0/21, Fa0/22, Fa0/23, Fa0/24

Gi0/1, Gi0/2

2 VLAN0002 active

4 VLAN0004 active

6 VLAN0006 active

8 VLAN0008 active

10 VLAN0010 active

12 VLAN0012 active

1002 fddi-default act/unsup

1003 token-ring-default act/unsup

1004 fddinet-default act/unsup

1005 trnet-default act/unsup

（输出被省略）

sw2#

说明：SW2 上的 VLAN 为 2 4 6 8 10 12 ， 全部是偶数。

5.验证 VTP

（1）开启 SW1 与 SW2 之间的 Trunk 链路：

SW1:

sw1(config)#int ran f0/23

```
sw1(config-if-range)#switchport trunk encapsulation dot1q
```

```
sw1(config-if-range)#switchport mode trunk
```

```
sw1(config-if-range)#no shut
```

SW2:

```
sw2(config)#int f0/23
```

```
sw2(config-if)#switchport trunk encapsulation dot1q
```

```
sw2(config-if)#switchport mode trunk
```

```
sw2(config-if)#no shutdown
```

说明： SW1 与 SW2 的 Trunk 已连通，VTP 即将同步。

(2) 查看 VTP 结果

SW1:

```
sw1#sh vtp status
```

VTP Version : 2

Configuration Revision : 6

Maximum VLANs supported locally : 1005

Number of existing VLANs : 11

VTP Operating Mode : Server

VTP Domain Name : ccie

CCIE LAB认证经验分享千人群：539730342

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x5E 0x0C 0x19 0x2B 0xC3 0x13 0x05 0x4F

Configuration last modified by 0.0.0.0 at 3-1-93 00:05:49

Local updater ID is 0.0.0.0 (no valid interface found)

sw1#

sw1#sh vlan

VLAN Name

Status

Ports

1 default

active Fa0/1, Fa0/2, Fa0/3, Fa0/4

Fa0/5, Fa0/6, Fa0/7, Fa0/8

Fa0/9, Fa0/10, Fa0/11, Fa0/12

Fa0/13, Fa0/14, Fa0/15, Fa0/16

Fa0/17, Fa0/18, Fa0/19, Fa0/20

Fa0/21, Fa0/22, Fa0/24, Gi0/1

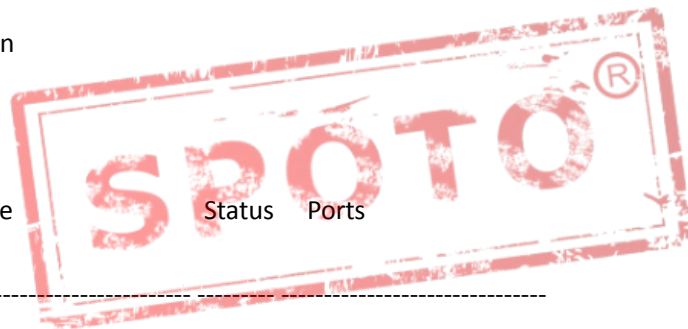
Gi0/2

2 VLAN0002

active

4 VLAN0004

active



CCIE LAB认证经验分享千人群：539730342

```
6  VLAN0006          active
8  VLAN0008          active
10 VLAN0010          active
12 VLAN0012          active
1002 fddi-default     act/unsup
1003 token-ring-default  act/unsup
1004 fddinet-default   act/unsup
1005 trnet-default     act/unsup
```

(输出被省略)

sw1#

SW2:

sw2#sh vtp status

VTP Version : 2

Configuration Revision : 6

Maximum VLANs supported locally : 1005

Number of existing VLANs : 11

VTP Operating Mode : Client

VTP Domain Name : ccie

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled



CCIE LAB认证经验分享千人群：539730342

VTP Traps Generation : Disabled

MD5 digest : 0x5E 0x0C 0x19 0x2B 0xC3 0x13 0x05 0x4F

Configuration last modified by 0.0.0.0 at 3-1-93 00:05:49

sw2#

sw2#sh vlan

VLAN Name	Status	Ports
-----------	--------	-------

1	default	active Fa0/1, Fa0/2, Fa0/3, Fa0/4
---	---------	-----------------------------------

Fa0/5, Fa0/6, Fa0/7, Fa0/8

Fa0/9, Fa0/10, Fa0/11, Fa0/12

Fa0/13, Fa0/14, Fa0/15, Fa0/16

Fa0/17, Fa0/18, Fa0/19, Fa0/20

Fa0/21, Fa0/22, Fa0/24, Gi0/1

Gi0/2

2	VLAN0002	active
---	----------	--------

4	VLAN0004	active
---	----------	--------

6	VLAN0006	active
---	----------	--------

8	VLAN0008	active
---	----------	--------

10	VLAN0010	active
----	----------	--------

12	VLAN0012	active
----	----------	--------


```
1002 fddi-default          act/unsup
1003 token-ring-default    act/unsup
1004 fddinet-default       act/unsup
1005 trnet-default         act/unsup
```

（输出被省略）

sw2#

说明：从结果中可以看出，VTP 模式为 Server 的 SW1 已经将自己的 VLAN 信息与 VTP 模式为 Client 的 SW2 同步，因为 SW1 的 Configuration Revision 为 4，而 SW2 的 Configuration Revision 为 6，所以无论 Server 与 Client，在收到 VTP 信息后，只要 Configuration Revision 比自己的大，则将自己的与收到的同步。

6. 验证 VTP 空域名

（1）查看 SW3 的 VTP 信息和 VLAN 信息

查看 VTP 信息：

sw3#sh vtp sta

VTP Version : 2

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 5

VTP Operating Mode : Server

VTP Domain Name :

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x57 0xCD 0x40 0x65 0x63 0x59 0x47 0xBD

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Local updater ID is 0.0.0.0 (no valid interface found)

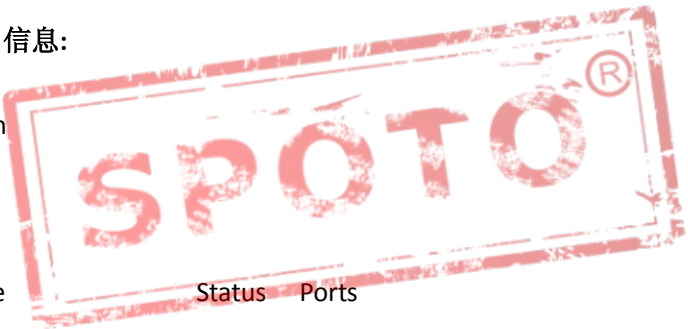
sw3#

查看 VLAN 信息:

sw3#sh vlan

VLAN Name	Status	Ports

1 default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4
		Fa0/5, Fa0/6, Fa0/7, Fa0/8
		Fa0/9, Fa0/10, Fa0/11, Fa0/12
		Fa0/13, Fa0/14, Fa0/15, Fa0/16
		Fa0/17, Fa0/18, Fa0/19, Fa0/20
		Fa0/21, Fa0/22, Fa0/23, Fa0/24
		Gi0/1, Gi0/2
1002 fddi-default	act/unsup	



```
1003 token-ring-default      act/unsup
```

```
1004 fddinet-default         act/unsup
```

```
1005 trnet-default           act/unsup
```

（输出被省略）

sw3#

说明：可以看到，SW3 的 VTP 域名为空，并且没有手工配置的 VLAN。

（2）开启 SW1 与 SW3 之间的 Trunk 链路：

SW1:

```
sw1(config)#int f0/19
```

```
sw1(config-if)#switchport trunk encapsulation dot1q
```

```
sw1(config-if)#switchport mode trunk
```

```
sw1(config-if)#no shutdown
```

SW2:

```
sw3(config)#int f0/19
```

```
sw3(config-if)#switchport trunk encapsulation dot1q
```

```
sw3(config-if)#switchport mode trunk
```

```
sw3(config-if)#no shutdown
```

说明：SW1 与 SW3 的 Trunk 已连通，VTP 即将同步。

(3) 查看 SW3 的 VTP 信息：

sw3#sh vtp status

VTP Version : 2

Configuration Revision : 6

Maximum VLANs supported locally : 1005

Number of existing VLANs : 11

VTP Operating Mode : Server

VTP Domain Name : ccie

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x5E 0x0C 0x19 0x2B 0xC3 0x13 0x05 0x4F

Configuration last modified by 0.0.0.0 at 3-1-93 00:05:49

Local updater ID is 0.0.0.0 (no valid interface found)

sw3#

说明：因为 SW3 的 VTP 域名为空，而 SW1 的 VTP 域名为 ccie，在域名为空的情况下，无论收到任何 VLAN 信息，只要 configuration revision 号比自己的大，就会同步自己的 VLAN 数据库，并且添加上相同的域名，所以空域名的 SW3 在收到 VTP 更新之后，同步了自己的信息。所以请谨慎使用空域名交换机。

VTP Pruning 与 VTP Ver 2

1.在 SW1 上开启 VTP Pruning 与 VTP Ver 2

(1) 在 SW1 上开启 VTP Pruning 与 VTP Ver 2

```
sw1(config)#vtp pruning
```

```
Pruning switched on
```

```
sw1(config)#vtp version 2
```

2 在 SW4 上查看结果

(1) 查看 SW4 上的 VTP Pruning 与 VTP Ver 2

```
sw4#sh vtp status
```

```
VTP Version          : 2
```

```
Configuration Revision : 10
```

```
Maximum VLANs supported locally : 1005
```

```
Number of existing VLANs : 11
```

```
VTP Operating Mode : Server
```

```
VTP Domain Name : ccie
```

```
VTP Pruning Mode : Enabled
```

```
VTP V2 Mode : Enabled
```

```
VTP Traps Generation : Disabled
```

CCIE LAB认证经验分享千人群：539730342

MD5 digest : 0xCA 0x78 0x25 0x9B 0x99 0x9B 0xE7 0x72

Configuration last modified by 1.1.1.1 at 3-1-93 00:35:06

Local updater ID is 1.1.1.4 on interface Vl1 (lowest numbered VLAN interface found)

sw4#

说明：VTP 域中，只要在一台上开启 VTP Pruning 与 VTP Ver 2，其它交换机全部开启，但只有 Server 和 Transparent 才能更改 VTP 版本 而 Transparent 是不支持 VTP Pruning 的。

3.更改 Pruning VLAN

说明：默认剪除 VLAN 2-1001，但可随意更改

(1) 在 SW1 的 Trunk 上更改 Pruning VLAN

sw1(config)#int f0/21

sw1(config-if)#switchport trunk pruning vlan remove 10

sw1(config-if)#exit

(2)查看 Pruning VLAN

sw1#sh int f0/21 switchport

(输出被省略)

Trunking VLANs Enabled: ALL

Pruning VLANs Enabled: 2-9,11-1001

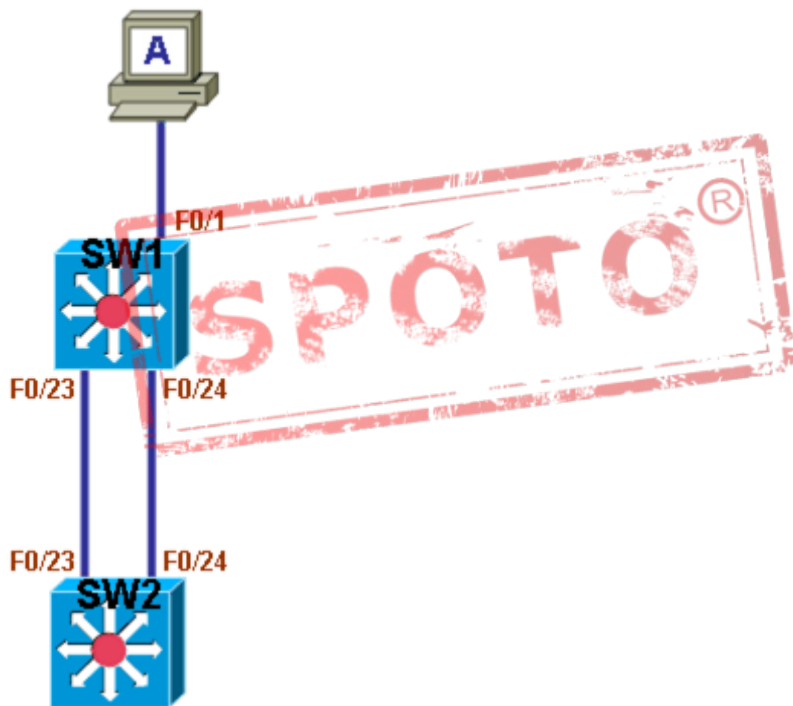
Capture Mode Disabled

（输出被省略）

sw1#

说明：可以看到，VLAN 10 已经从 Pruning VLAN 中移除，只剩 VLAN 2-9,11-1001。

STP（Spanning-Tree Protocol）



在上图所示的网络环境中，当交换机之间连有多条链路时，将存在一定的问题，如 SW1 的 MAC 地址表中会显示接口 F0/1 与主机 A 相对应，而当数据发往 SW2 后，SW2 的 MAC 地址表则记录接口 F0/23 与主机 A 相对应，当 SW2 再次将流量从接口 F0/24 发回 SW1 时，SW1 的 MAC 地址表又会记录接口 F0/24 与主机 A 相对应。

因此可以看出，当交换机之间存在多条活动链路时，交换机将从不正常的接口上学习到 MAC 地址，导致 MAC 地址表的不正确与不稳定，并且还会导致重复的数据包在网络中传递，引起广播风暴，使网络不稳定。

CCIE LAB认证经验分享千人群：539730342

为了防止交换机之间由于多条活动链路而导致的网络故障，必须将多余的链路置于非活动状态，即不转发用户数据包，而只留下单条链路作为网络通信，当唯一的活动链路不能工作时，再启用非活动链路，从而达到网络的冗余性。要实现此功能，需要依靠生成树协议（STP）来完成，STP 将交换网络中任何两个点之间的多余链路置于 Blocking（关闭）状态，而只留一条活动链路，当使用中的活动链路失效时，立即启用被 Block 的链路，以此来提供网络的冗余效果。

STP 并非思科私有协议，STP 为 IEEE 标准协议，并且有多个协议版本，版本与协议号的对应关系如下：

Common Spanning Tree (CST) = IEEE 802.1D

Rapid Spanning Tree Protocol (RSTP) = IEEE 802.1w

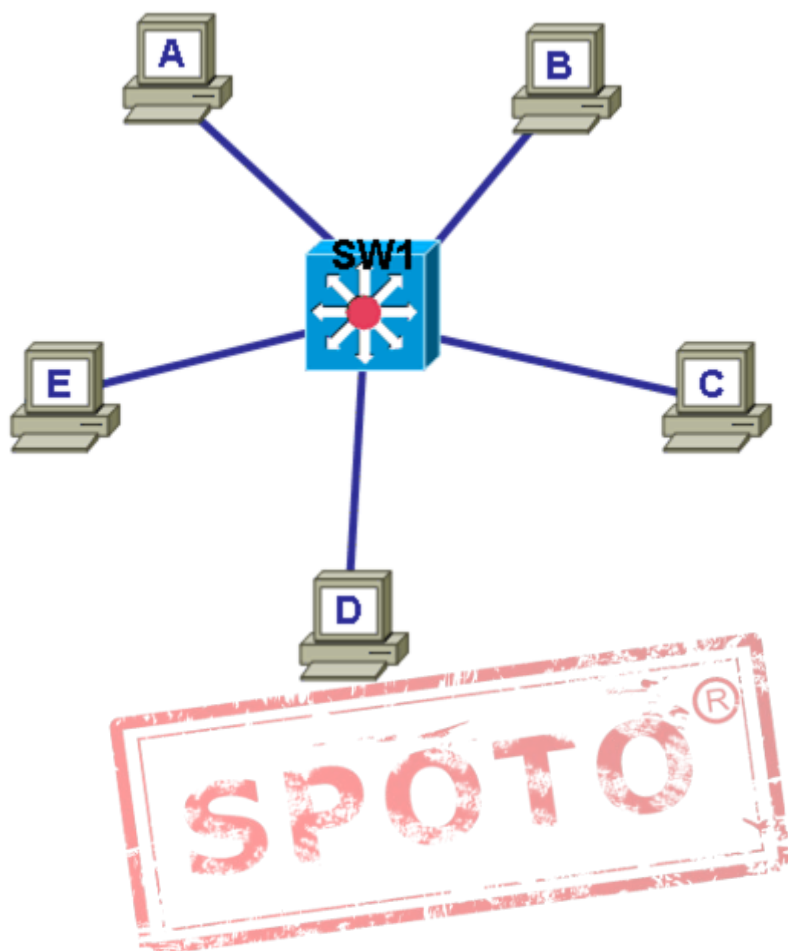
Per-VLAN Spanning-Tree plus (PVST+) = Per-VLAN IEEE 802.1D

Rapid PVST+ = Per-VLAN IEEE 802.1w

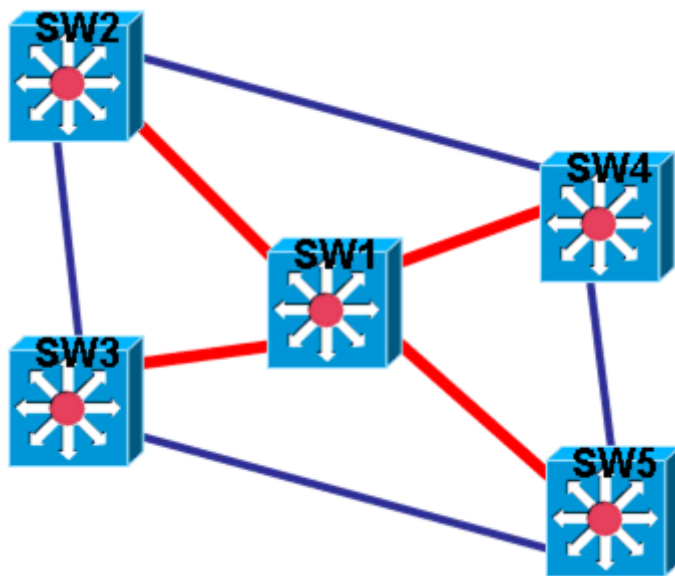
Multiple Spanning Tree Protocol (MSTP) = IEEE 802.1s

下面来详细介绍 STP 协议：

请观察如下网络环境：



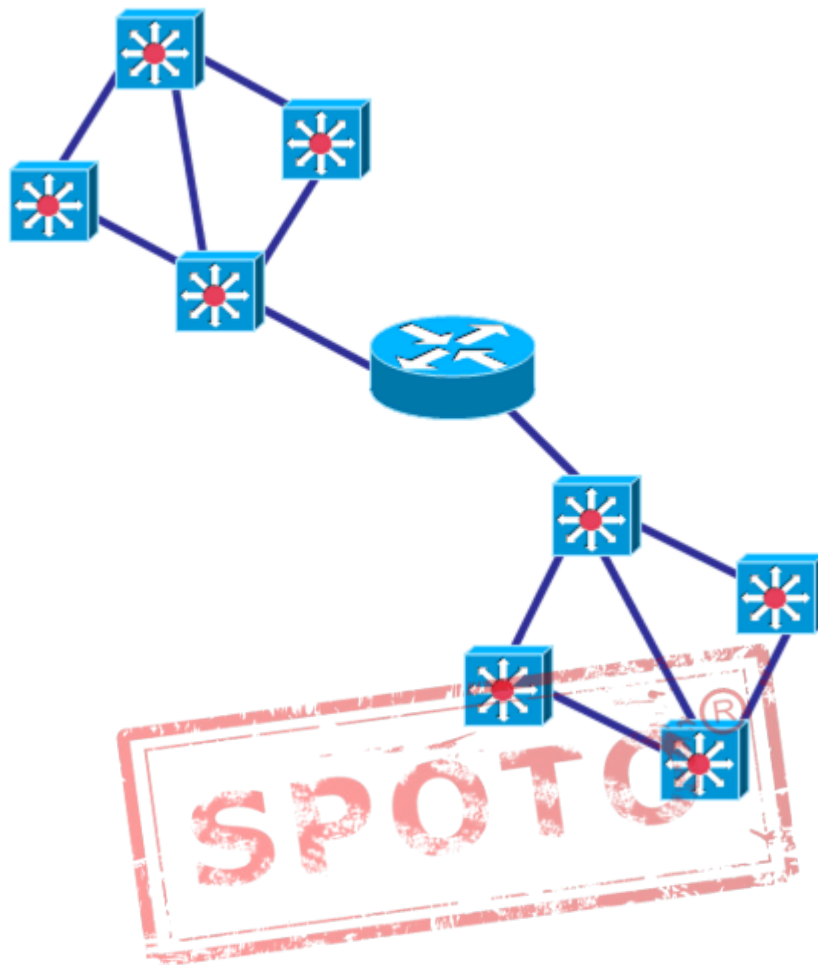
在如上所示的网络环境中，不难看出，当所有主机都使用单条链路与一台核心相连时，只要不再增加其它额外设备与链路，就不可能存在环路。交换机就当相于 Hub 一样连接了多台主机，而这样的网络结构，被称为 hub-spoke 网络结构，只要主机与 Hub 是连通的，那么就表示主机之间是连通的。基于此原因，STP 借助了 hub-spoke 网络结构无环的网络思想，将一个拥有多台交换机通过多条链路相连的网络，通过 Block 掉任意两点之间多余的链路而只留下单条链路，最终修整出一个 hub-spoke 的网络环境，创建一个无环的交换网络。



在上图的交换网络中，由于存在多台交换机，并且交换机之间有多条冗余链路，因此，只要在网络中找一台交换机充当核心，也就是相当于 hub-spoke 网络中的 Hub，而其它交换机则留出一条活动链路到核心交换机即可，其它链路全部被 block，当留出的活动链路失效之后，再启用 block 链路作为备份。上图中 SW1 被选作交换网络中的核心，而其它交换机则只留一条活动链路到核心交换机，只要其它交换机与核心交换机是通的，就证明交换机之间一定是通的。图中红色的连路表示被留出的普通交换机到核心交换机的活动链路，蓝色链路表示被 block 掉的链路，只要红色链路是通的，就表示整个网络都是通的，当某条红色链路断掉以后，只要启用相应的蓝色链路代替即可，也就实现了网络的冗余功能。

通过上述的解释，STP 要构建出无环的交换网络，就必须在网络中选出一台交换机做为核心交换机，STP 称其为 Root，也就是根，功能相当于 hub-spoke 网络中的 Hub。其它不是 Root 的交换机则需要留出一条活动链路去往根交换机，因为只要普通交换机到根是通的，到其它交换机也就是通的。

需要说明的是，只有在一个三层网络中，广播能够到达的范围内，才需要进行相同的 STP 计算与选举，也就是一个广播域内独立选举 STP：



上图中，因为网络被路由器分割成两个广播域，所以在两个网段中，需要进行独立的 STP 计算与选举。

STP 在计算与选举时，只会留下唯一一条活动链路，将其它所有多余链路全部 block，所以 STP 要确定两点之间是否存在多条链路，因为只有两点之间有多条链路时，才有链路需要被 block。要确认两点之间网络是否通畅，只要发送数据作个测试即可得到答案，而要确认两点之间是否有多条链路，方法还是发送数据作个测试就能得到答案。当然，要测试两点之间是否有多条链路，需要发送特殊的数据来做测试，比如给数据包都做上相同的标记，然后发出去，如果交换机同时从多个接口收到相同标记的数据包，很显示，交换机与发送者之间就是存在多条链路的，因此需要靠 STP 计算来断开多余链路。

STP 在发送数据包测试网络是否有多条链路，是靠发送 bridge protocol data units (BPDU)s来完成的，同台交换机发出去的 BPDU 都被做上了相同的标记，只要任何交换机从多个接口收到相同标记的 BPDU，就表示网络中有冗余链路，因此需要 STP 断开多余链路。BPDU 数据包里面有以下信息：

根交换机的 bridge ID。

发送交换机的 bridge ID 。

到根交换机的 Path Cost。

发送接口以及优先级。

Hello、forward delay、max-age 时间。

同台交换机发出的 BPDU，bridge ID 都是一样的，因为是用来标识自己的，其中 bridge ID 由两部分组成：Bridge 优先级和 MAC 地址，默认优先级为 32678。

交换机上的每个端口也是有优先级的，默认为 128，范围为 0-255。

注：在 STP 协议中，所有优先级数字越小，表示优先级越高，数字越大，优先级越低。

STP 在计算网络时，需要在网络中选举出根交换机（Root），根端口（Root Port），以及指定端口（Designated Port），才能保证网络的无环，选举规则分别如下：

根交换机（Root）

在同一个三层网络中需要选举，即一个广播域内要选举，并且一个网络中只能选举一台根交换机。Bridge-ID 中优先级最高（即数字最小）的为根交换机，优先级范围为 0-65535，如果优先级相同，则 MAC 地址越小的为根交换机。

根端口（Root Port）

所有非根交换机都要选举，非根交换机上选举的根端口就是普通交换机去往根交换机的唯一链路，选举规则为 到根交换机的 Path Cost 值最小的链路，如果多条

链路到达根交换机的 Path Cost 值相同，则选举上一跳交换机 Bridge-ID 最小的链路，如果是经过的同一台交换机，则上一跳交换机 Bridge-ID 也是相同的，再选举对端端口优先级最小的链路，如果到达对端的多个端口优先级相同，最后选举交换机对端端口号码最小的链路。

指定端口(Designated Port)

在每个二层网段都要选举，也就是在每个冲突域需要选举，简单地理解为每条连接交换机的物理线路的两个端口中，有一个要被选举为指定端口，每个网段选举指定端口后，就能保证每个网段都有链路能够到达根交换机，选举规则和选举根端口一样，即：到根交换机的 Path Cost 值最小的链路，如果多条链路到达根交换机的 Path Cost 值相同，则选举上一跳交换机 Bridge-ID 最小的链路，如果是经过的同一台交换机，则上一跳交换机 Bridge-ID 也是相同的，再选举对端端口优先级最小的链路，如果到达对端的多个端口优先级相同，最后选举交换机对端端口号码最小的链路。

在 STP 选出根交换机，根端口以及指定端口后，其它所有端口全部被 Block，为了防止环路，所以 Block 端口只有在根端口或指定端口失效的时候才有可能被启用。

交换机上的端口，根据端口的带宽不同，Path Cost 值也不同，以下参数为标准：

10 Mb/s: 100

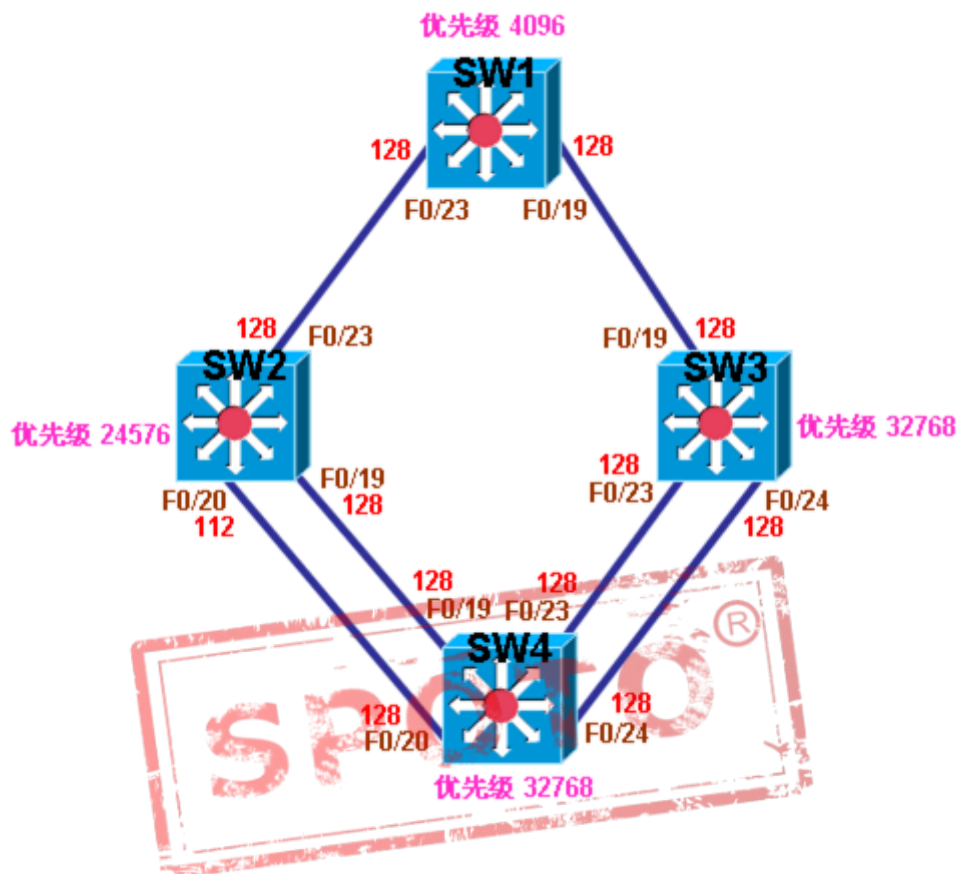
100 Mb/s: 19

1000 Mb/s: 4

10000 Mb/s: 2

可以看出，带宽越高，被选为根端口和指定端口的几率就越大，所以经过 STP 选举后，活动的链路总是性能最好的，其它被 Block 掉的端口，将在活动端口失效时被启用。

以下图为例来看 STP 计算：



上图的网络环境中，运行 STP 后，则选举如下角色：（所有链路为 100 Mb/s，即 Path Cost 值为 19）

根交换机（Root）

因为 4 台交换机的优先级分别为 SW1（4096），SW2（24576），SW3（32768），SW4（32768），选举优先级最高的（数字最低的）为根交换机，所以 SW1 被选为根交换机，如果优先级相同，则比较 MAC 地址。

根端口 (Root Port)

根端口需要在除 SW1 外的非根交换机上选举。

SW2 上从端口 F0/23 到达根的 Path Cost 值为 19, 从 F0/19 和 F0/20 到达根的 Path Cost 值都为 $19 \times 3 = 57$ 。因此, F0/23 被选为根端口。

SW3 上从端口 F0/19 到达根的 Path Cost 值为 19, 从 F0/23 和 F0/24 到达根的 Path Cost 值都为 $19 \times 3 = 57$ 。因此, F0/19 被选为根端口。

SW4 上从所有端口到达根的 Path Cost 值都为 $19 \times 2 = 38$, 所以从比较 Path Cost 值, 无法选出根端口, 接下来比较上一跳交换机 Bridge-ID, 也就是比较 SW2 与 SW3 的 Bridge-ID, 所以选择往 SW2 的方向, 然而通过端口 F0/19 和 F0/20 都可以从 SW2 到达根交换机, 所以接下来比较端口 F0/19 和 F0/20 对端交换机端口的优先级, 因为 SW2 的 F0/19 端口优先级为 128, 而 F0/20 的端口优先级为 112, 所以 SW4 选择连接 SW2 的 F0/20 的端口为根端口, 即 SW4 的 F0/20 为根端口, 如果此步还选不出, SW4 将根据对端端口号做出决定, 也就是 F0/19 和 F0/20, 数字小的为根端口, 也就是 F0/19。



指定端口 (Designated Port)

每个网段 (每个冲突域), 或理解为每条线路都要选举指定端口。

在根交换机 SW1 连接 SW2 的网段与连接 SW3 的网段中, 当然是根自己的端口离自己最近, 所以这两个网段中, 选举根交换机上的端口为指定端口, 因此, 根交换机上所有的端口都应该是指定端口。

在 SW3 连接 SW4 的两个网段中, 同样也是 SW3 上的两个端口离根交换机最近, 所以在这两个网段中, 选举 SW3 上的端口为指定端口。

在 SW2 连接 SW4 的两个网段中, 同样也是 SW2 上的两个端口离根交换机最近, 所以在这两个网段中, 选举 SW2 上的端口为指定端口。

注: 根交换机上所有的端口最终都为指定端口。

其它既不是根端口，也不是指定端口的落选的端口，就是 SW4 上的 F0/19, F0/23, F0/24，都将被 STP 放入 **Blocking** 状态，不为用户提供数据转发，以此来防止环路。最终的网络，构建出了任何两点之间，都是单链路的环境，不会有环路，当使用中的链路失效时，**Blocking** 的端口可以代替原端口。上图的 STP 选举结果如下：

根交换机 (Root)

SW1

根端口 (Root Port)

SW2: F0/23 SW3: F0/19 SW4: F0/20

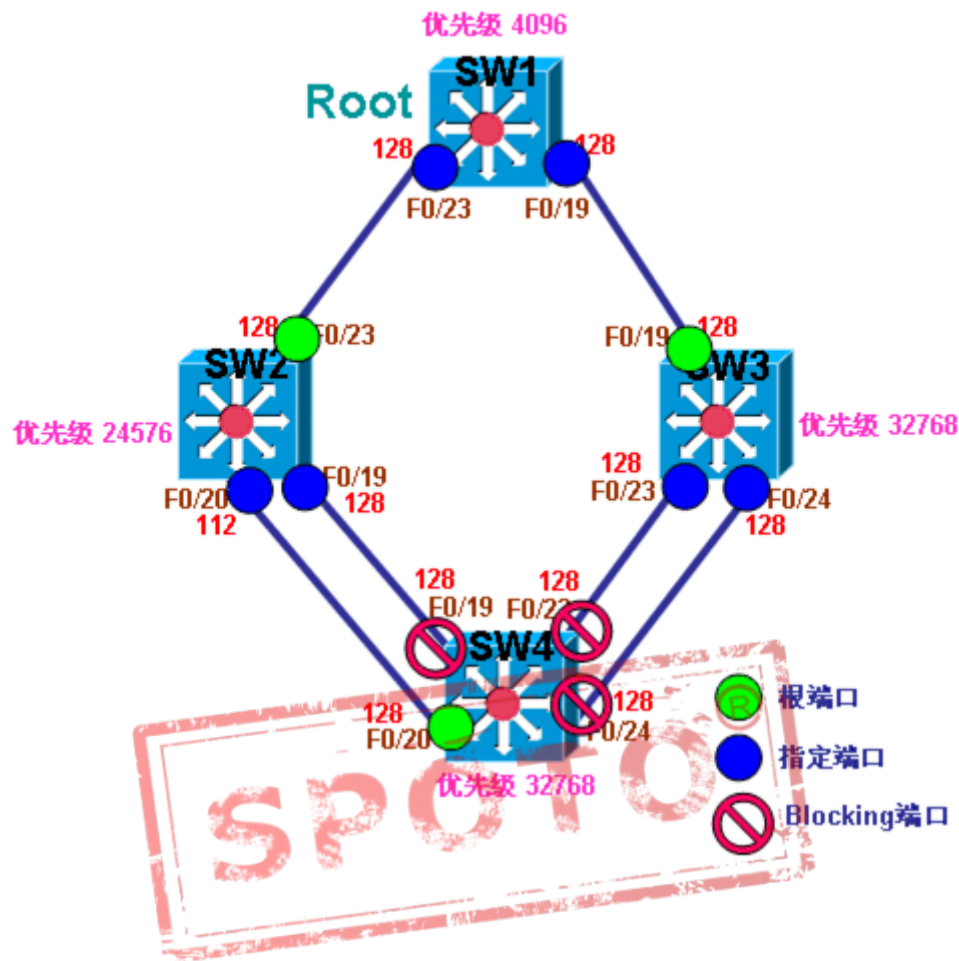
指定端口 (Designated Port)

SW1: F0/19, F0/23 SW2: F0/19, F0/20 SW3: F0/23, F0/24

Blocking 端口

SW4: F0/19, F0/23, F0/24

结果图如下：



注：一个端口，在 STP 中只能处于一种角色，不可能是两种角色。

在交换机启动后，端口要过渡到转发状态，需要经历以下的状态：

- 1 从 initialization（初始化）到 blocking
- 2 从 blocking 到 listening 或 disabled
- 3 从 listening 到 learning 或 disabled

4 从 learning 到 forwarding 或 disabled

被 Disabled 的接口相当于关闭了，每个状态有如下功能：

Blocking

丢弃所有收到的数据帧，不学习 MAC 地址，能收 BPDU 但不发 BPDU。

Listening

丢弃所有收到的数据帧，不学习 MAC 地址，能收 BPDU 的处理 BPDU，并进行 STP 计算。

Learning

丢弃所有收到的数据帧，会学习 MAC 地址，能收 BPDU 和处理 BPDU。

Forwarding

也就是正常转发状态，能转发收到的数据帧，能学习 MAC 地址，接收并处理 BPDU。

Disabled

丢弃所有收到的数据帧，不学习 MAC 地址，能收 BPDU，除此之外不会再做其它的。



当交换机启动后，都认为自己是根交换机，然后从所有接口向网络中发送 BPDU，称为 configuration BPDU，所以 configuration BPDU 是根交换机发出的。当交换机收到更优 Bridge-ID 的 configuration BPDU，会将它从自己所有接口转发出去，并保存在接口，如果收到差的 configuration BPDU，则全部丢掉，所以在交换网络中，只有根交换机的 BPDU 在转发，其它普通交换机的 BPDU 不会出现在网络中。

根交换机的 BPDU 会在每个 hello 时间往网络中发送一次，hello 时间默认为 2 秒钟，也就是交换机的 BPDU 会在每 2 秒钟往网络中发送一次，如果普通交换机在 max-age 时间内没有收到根交换机的 BPDU，则认为根交换机已经失效，便开始重新选举 BPDU，默认 max-age 时间为 20 秒，即 10 倍 hello 时间。

除了 hello 时间和 max-age 时间外，还有一个 forward delay 时间，默认为 15 秒，接口在经过 Listening 和 Learning 状态时，都会分别停留一个 forward delay 时间，也就是说接口从 Listening 状态到 Learning 状态，最后变成转发状态，需要经过两个 forward delay 时间共计 30 秒。

因为 STP 有多个版本，不同版本的 STP，在操作和运行上，会有所不同，但是需要说明，无论什么版本的 STP，对根交换机，根端口以及指定端口的选举规则完全是一样的，下面分别详细介绍各版本的运行过程：

Common Spanning Tree (CST)

CST 的协议号为 IEEE 802.1D，如果交换机运行在 CST，交换机只进行一次 STP 计算，无论交换机上有多少个 VLAN，所有流量都会走相同的路径。

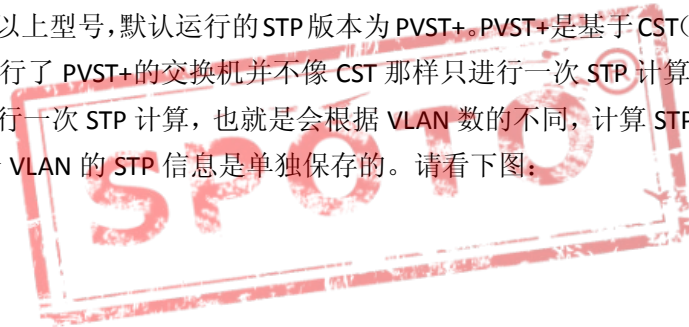
Rapid Spanning Tree Protocol (RSTP)

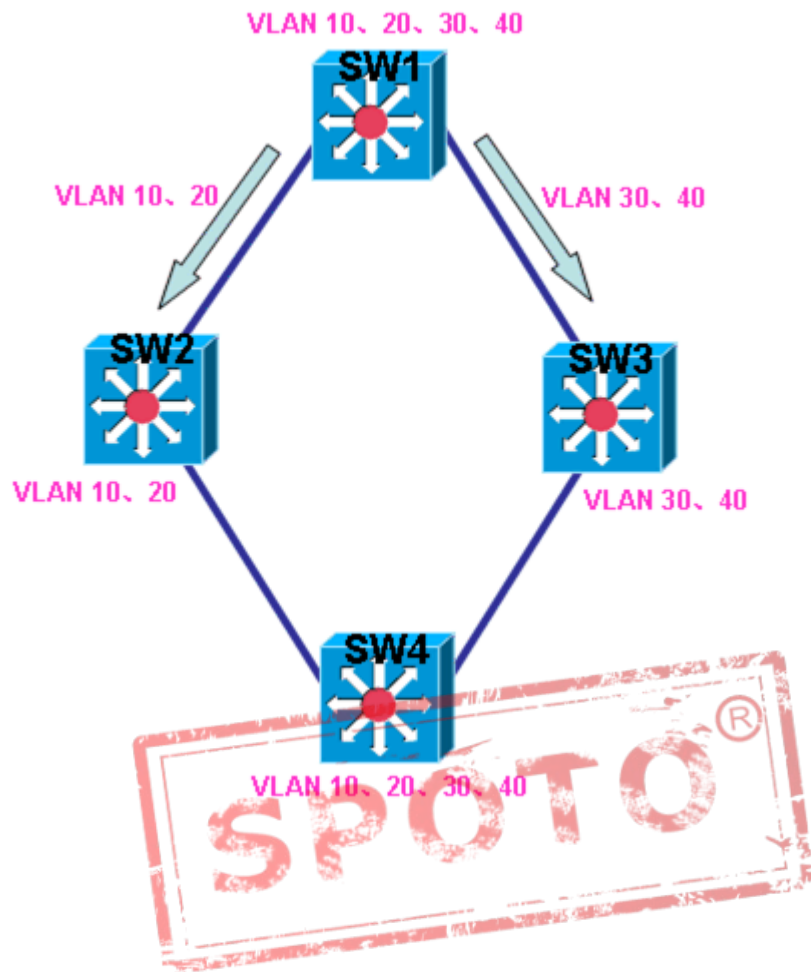
RSTP 是快速 STP，协议号为 IEEE 802.1w，在运行 CST 时，端口状态 blocking、listening、disabled 都不发送数据，RSTP 将这三个状态归为一个状态，discarding 状态。其次之外就是 learning 和 forwarding 状态，所以 RSTP 端口状态为 discarding、learning 和 forwarding。

当运行 CST 时，如果根交换机失效了，那么需要等待 10 个 hello 时间，也就是 20 秒收不到根交换机 BPDU 才能发现，再将 block 的端口过滤到 forwarding 状态，还需要经过两个 forward delay 时间共计 30 秒，所以 CST 在网络出现故障时，要经过 50 秒才能启用 block 端口，而 RSTP 则只需要在 3 个 hello 时间，即 6 秒收不到根交换机 BPDU，便认为根交换机已经失去连接，就立刻启用 discarding 状态的接口，RSTP 在根交换机失效后，并不会进行完整的 STP 计算，会在该启用备用端口时立即启用，因此网络收敛速度快，RSTP 会在低于 1 秒的时间内恢复网络。

Per-VLAN Spanning-Tree plus (PVST+)

PVST+是思科自己的协议，在之前有一个 PVST，但由于 PVST 只能支持 ISL Trunk，所以思科为了扩展 PVST 支持 IEEE 802.1Q，诞生了 PVST+，在多数三层交换机，如 3550、3560 及以上型号，默认运行的 STP 版本为 PVST+。PVST+是基于 CST(IEEE 802.1D)运行的，但运行了 PVST+的交换机并不像 CST 那样只进行一次 STP 计算，PVST+会在每个 VLAN 进行一次 STP 计算，也就是会根据 VLAN 数的不同，计算 STP 的次数也不同，并且每个 VLAN 的 STP 信息是单独保存的。请看下图：





在上图的网络中，各台交换机上都有 VLAN 10，VLAN 20，VLAN 30，VLAN 40，在运行 CST 的情况下，因为只进行一次 STP 计算，所以 SW1 到 SW4 的流量要么从 SW2 走，要么从 SW3 走，在这种情况下，流量只能走同一条路径，而另一条路径完全被空闲而得不到利用。

当在上图的网络中运行 PVST+ 后，因为 PVST+ 会在每个 VLAN 进行不同的 STP 计算，称为 STP 实例 (instance)，所以可以控制每个 VLAN 流量的路径走向。上图中，就可以通过 PVST+ 控制 SW1 的 VLAN10 和 VLAN20 从连接 SW2 的接口到达 SW4，控制 SW1 的 VLAN 30 和 VLAN 40 从连接 SW3 的接口到达 SW4，这样之后，将不同的 VLAN 流量分担到不同的路径，即实现了负载均衡，也通过 STP 避免了环路。

重点说明：

PVST+ 只支持 128 个实例 (instance)，如果交换机上配置的 VLAN 数超过 128 个，那么 128 个以外的 VLAN 将没有 STP 在运行，所以此时剩余的 VLAN 将出现环路。可以单独在特定的 VLAN 上打开或关闭 STP 功能，即使一台没有运行 STP 的交换机或没有运行 STP 的 VLAN，在收到 BPDU 时，也会转发的，所以在对单个 VLAN 进行开启或关闭 STP 时，请确保交换机能够计算出无环的网络，否则网络将出现预想不到的故障。

在 PVST+ 可以配置全局关闭某 VLAN 的 STP，如关闭 VLAN 10 的 STP

no spanning-tree vlan 10，恢复使用命令 spanning-tree vlan 10

Extended System ID

默认交换机的 Bridge-ID 的优先级为 32768，当开启 Extended System ID 功能后，每个 VLAN 的默认的 Bridge-ID 优先级就不再是 32768 了，需要再加上 VLAN 号码，如 VLAN 1 的 Bridge-ID 优先级就是 $32768+1=32769$ ，VLAN 8 的 Bridge-ID 优先级就是 $32768+8=32776$ 。

如果网络中即有开启了 Extended System ID 功能的交换机，也有关闭的，那么关闭 Extended System ID 功能的交换机有更大的机会成为根交换机，因为自己默认的优先级就比其它开启了 Extended System ID 功能的优先级更高（数字更小）。

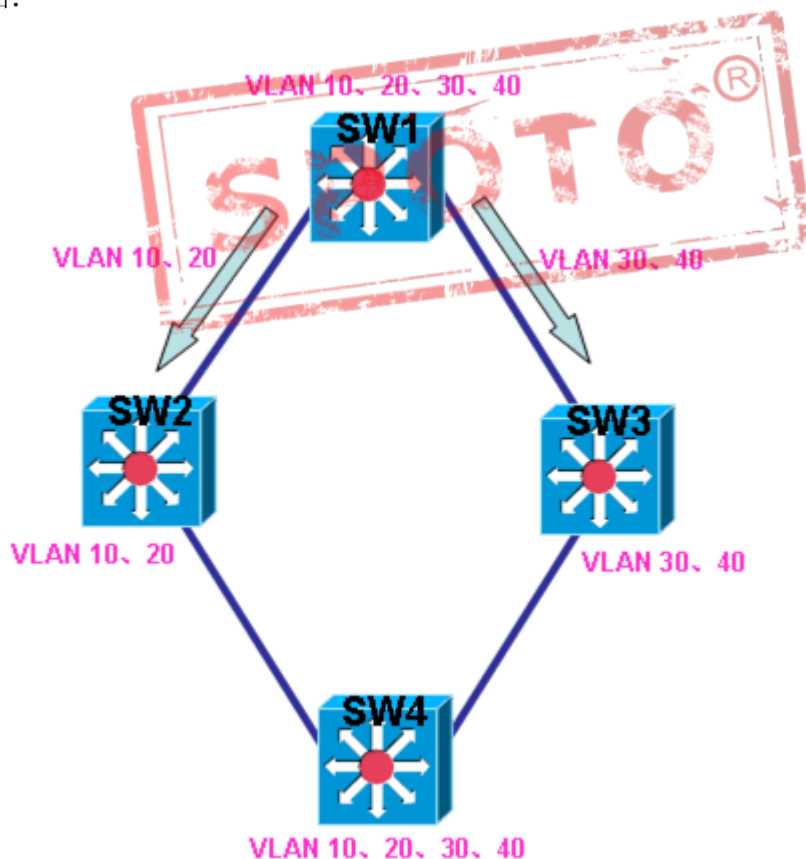
[返回目录](#)

Rapid PVST+

Rapid PVST+ 就是具有 RSTP 特性的 PVST+，是像 RSTP 一样基于 IEEE 802.1w 运行的，其它所有运行与规则与 PVST+ 完全相同，不再做详细介绍。

Multiple Spanning Tree Protocol (MSTP)

MSTP 的协议号为 IEEE 802.1s，因为在交换机存在多个 VLAN 时，CST 会将所有流量放在单条路径中传输，而 PVST+ 则可以通过为每个 VLAN 运行一个 STP 实例，从而将不同 VLAN 的流量放在不同的路径上传输。但正是由于 PVST+ 为每个 VLAN 都运行了一个 STP 实例，可能会多达 128 个 STP 实例，所以 PVST+ 会极其消耗系统资源。比如交换机上有 20 个 VLAN，而 PVST+ 会维护 20 个 STP 实例，但是这 20 个 VLAN 的流量也许只需要被分担到几条不同路径上，那就只需要维护几个 STP 实例即可，而并不需要维护 20 个 STP 实例。MSTP 正因为这个原因，将需要进行相同 STP 计算的 VLAN 映射到同一个 STP 实例中，即无论有多少个 VLAN，只要实际需要多少条不同的路径，就根据需要的路径维护相同的 STP 实例数，从而大大节省系统资源，如下图：



还是以此图为例，因为各台交换机上都有 VLAN 10, VLAN 20, VLAN 30, VLAN 40，为了能够在 SW1 上让不同 VLAN 的流量从不同的路径到达 SW4，所以可以运行 PVST+，将流量分担到不同的路径上，即 SW1 通过 PVST+将 VLAN10 和 VLAN20 的流量从连接 SW2 的接口到达 SW4，将 VLAN 30 和 VLAN 40 的流量从连接 SW3 的接口到达 SW4，但 PVST+维护了 4 个 STP 实例，才达到此效果，不难看出，其实网络中只有两个不同的路径，VLAN 10 和 VLAN 20 的路径完全是相同的，VLAN 30 和 VLAN 40 的路径也是完全相同的，此时，MSTP 就可以通过将相同的 VLAN 映射到同一个 STP 实例，如将 VLAN 10 和 VLAN 20 映射到一个实例，再将 VLAN 30 和 VLAN 40 映射到另外一个实例，总共只有两个 STP 实例，既像 PVST+那样实现了负载均衡的效果，也节省了系统资源。

MSTP 是在 RSTP 的基础之上运行的，所以具有快速收敛的功能，但不能不运行 RSTP 时运行 MSTP，RSTP 是随着开启 MSTP 时自动开启的。MSTP 最多支持 65 个 STP 实例，但是映射到实例的 VLAN 数量是没有限制的。默认所有 VLAN 都在实例 0。

MSTP 还需要通过分区域管理，即 region，交换机要在同一 region 进行相同的 STP 计算，必须 region name 和 revision number 一致，最重要的是 VLAN 和实例的映射也要一致，否则 STP 计算出来的网络，将不是你想要的网络，一个 VLAN 只能被映射到一个实例，一个网络可以有多个 MSTP revision，便于各自独立。

拓朴变更

当网络中的链路出现变化时，也就需要进行新的 STP 计算，并且由于交换机的 MAC 地址在表中的老化时间默认为 300 秒（5 分钟），所以当原有的链路发生变化后，MAC 地址与接口的对应关系也会发生变化，因此不能再等 5 分钟才更新，所以基于拓朴变化的因素，还需要将 MAC 地址的老化时间设置的更短，此动作在 STP 拓朴变更时，会自动更改为 forward_delay 的时间。

当网络链路发生变化后，必须进行新的 STP 计算，但是在正常的 STP 状态下，只有根交换机才能往网络里发送 BPDU，称为 configuration BPDU，而普通交换机只有接收 configuration BPDU 的权限，并不能向网络中发送 BPDU。但是当交换机检测到链路变化时，可以通知网络中的根交换机，此时可以发送一种特殊的 BPDU，叫做 topology change notification (TCN)，也就是 TCN BPDU。TCN BPDU 是用来告诉根交换机网络链路有变化，因此 TCN BPDU 只能从根端口发出去，如果接收者不是根交

交换机，则必须回复一个确认消息，这个消息是一个设置了 TCA 位的 configuration BPDU，然后自己再从根端口向根发送 TCN BPDU，直到根收到为止，当根收到 TCN BPDU 后，需要回复该 BPDU，方式为发送一个设置了 TC 位的 configuration BPDU。

其中，TCN 是一种特殊的 BPDU，而 TCA 只是设置了 TCA 位的 configuration BPDU，TC 也只是设置了 TC 位的 configuration BPDU。最终 STP 网络中，出现了两种 BPDU，即 TCN BPDU 和 configuration BPDU。

注：

★在配置 STP 时，Bridge-ID 的优先级，端口优先级，hello 时间，max-age 时间，forward delay 时间都可以手工修改，而 Bridge-ID 的优先级必须为 4096 的整数倍，端口优先级必须为 16 的整数倍。

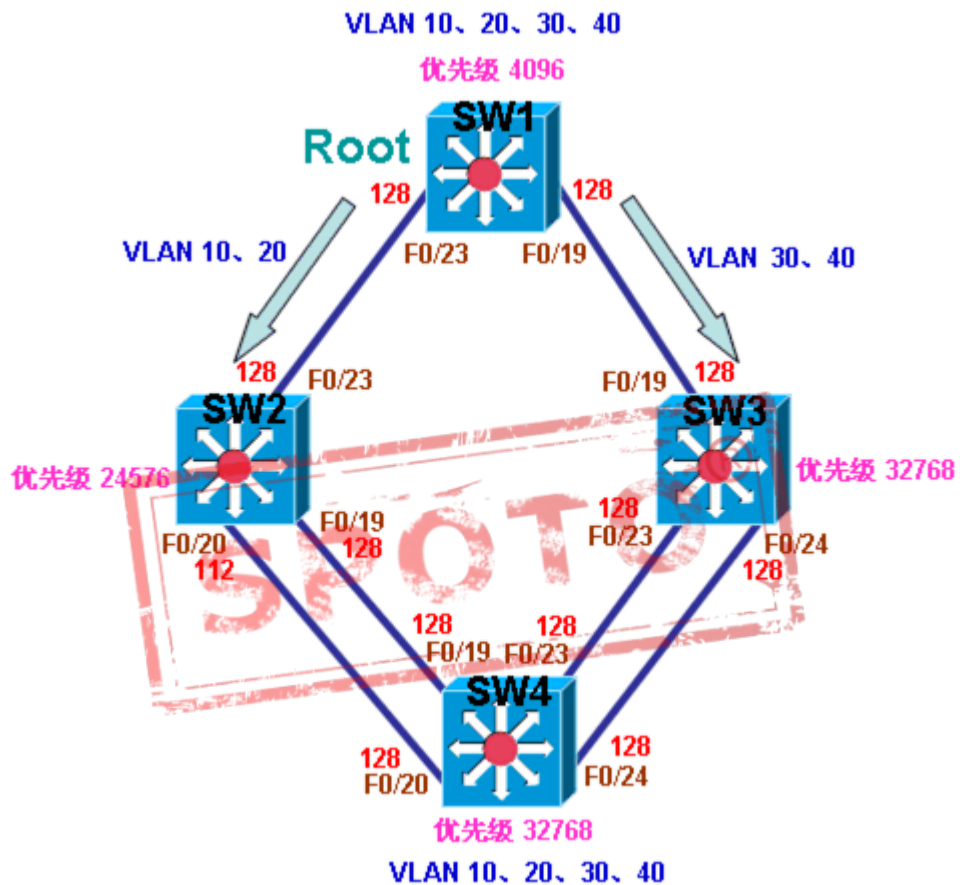
★在修改时，PVST+可以基于每个 VLAN 修改，而 MSTP 则只能基于实例，而不能基于 VLAN，因为一个实例会有多个 VLAN。

可以通过命令来强制指定某台交换机为根交换机，当使用命令强制指定某交换机为根后，此交换机将通过修改一个比当前根交换机更高优先级的 Bridge-ID，以此来抢夺根交换机的角色，如果命令再到别的交换机上输入，那么那台交换机将再次抢夺根交换机的角色，因为它可以修改自己的 Bridge-ID 比当前根更高的优先级，所以此命令最后在网络中的哪台交换机上输入后，哪台交换机就能成为根交换机，但是也有个限度，因为交换机的 Bridge-ID 不能自动改的比 1 小，又不能改 MAC 地址，所以如果需要修改优先级到 1 以下才能抢夺根交换机的角色，那么此命令将提示错误。

注：链路的全双工与半双工，在 STP 中，被分为不同的链路类型，如果是全双工（full-duplex），叫做 point-to-point(P2p)，如果是半双工，叫做（half-duplex）。接口下可以手工更改：spanning-tree link-type point-to-point。

配置

配置 PVST+



说明：以上图为例，配置 PVST+，默认交换机上都配置有 VLAN 10，VLAN 20，VLAN 30，VLAN 40，要求控制 SW1 与 SW4 之间的流量路径为 VLAN 10 和 VLAN 20 从 SW1—SW2—SW4，VLAN 30 和 VLAN 40 从 SW1—SW3—SW4。

注：默认为 PVST+，所以 STP 版本不用改。

1.配置各交换机优先级（只能为 4096 的整数倍）

（1）配置 SW1 在所有 VLAN 的优先级为 4096


```
sw1(config)#spanning-tree vlan 10-40 priority 4096
```

（2）配置 SW2 在所有 VLAN 的优先级 24576

```
sw2(config)#spanning-tree vlan 10-40 priority 24576
```

（3）配置 SW3 在所有 VLAN 的优先级 32768

```
sw3(config)#spanning-tree vlan 10-40 priority 32768
```



（4）配置 SW4 在所有 VLAN 的优先级 32768

```
sw4(config)#spanning-tree vlan 10-40 priority 32768
```

2.配置 SW2 的 F0/20 的端口优先级（必须为 16 的整数倍）

（1）在所有 VLAN 将 SW2 的 F0/20 的端口优先级配置为 112

```
sw2(config)#int f0/20
```

```
sw2(config-if)#spanning-tree vlan 10-40 port-priority 112
```

3.查看根交换机

（1）查看根交换机 SW1

CCIE LAB认证经验分享千人群：539730342

说明：因为现在 4 个 VLAN 的配置是一样的，结果也是一样的，所以只提供一个 VLAN 的结果：

```
sw1#sh spanning-tree
```

（输出被省略）

```
VLAN0010
```

```
Spanning tree enabled protocol ieee
```

```
Root ID    Priority    4106
```

```
Address    001a.6c6f.fb00
```

```
This bridge is the root
```

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

```
Bridge ID  Priority    4106 (priority 4096 sys-id-ext 10)
```

```
Address    001a.6c6f.fb00
```

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

```
Aging Time 300
```

```
Interface    Role Sts Cost    Prio.Nbr Type
```

```
-----  
Fa0/23      Desg FWD 19      128.25 P2p
```

（输出被省略）

```
sw1#
```

CCIE LAB认证经验分享千人群：539730342

说明：从结果中看出，SW1 手工配置的优先级为 4096，但由于 Extended System ID 功能，所以优先级加上了 VLAN 号码 10，结果优先级变为 4106，因为优先级在网络中数字最小，所以自己就是当前网络的根交换机。

4.查看根端口

（1）查看 SW2 的根端口

```
sw2#sh spanning-tree
```

（输出被省略）

```
VLAN0010
```

```
Spanning tree enabled protocol ieee
```

```
Root ID    Priority    4106
```

```
Address     001a.6c6f.fb00
```

```
Cost        19
```

```
Port        23 (FastEthernet0/23)
```

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

```
Bridge ID  Priority    24586 (priority 24576 sys-id-ext 10)
```

```
Address     0013.805c.9d00
```

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

```
Aging Time  300
```

```
Interface    Role Sts Cost    Prio.Nbr Type
```

```
-----  
Fa0/19      Desg FWD 19      128.19  P2p  
Fa0/20      Desg FWD 19      112.20  P2p  
Fa0/23      Root FWD 19      128.23  P2p
```

（输出被省略）

sw2#

说明：因为 SW2 上从端口 F0/23 到达根的 Path Cost 值为 19，从 F0/19 和 F0/20 到达根的 Path Cost 值都为 $19 \times 3 = 57$ 。因此，F0/23 被选为根端口。

（2）查看 SW3 的根端口

sw3#sh spanning-tree

（输出被省略）

VLAN0010

Spanning tree enabled protocol ieee

Root ID Priority 4106

Address 001a.6c6f.fb00

Cost 19

Port 21 (FastEthernet0/19)

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

Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)

CCIE LAB认证经验分享千人群：539730342

Address 001a.a256.f300

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

Aging Time 15

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Root	FWD	19	128.21		P2p
--------	------	-----	----	--------	--	-----

Fa0/23	Desg	FWD	19	128.25		P2p
--------	------	-----	----	--------	--	-----

Fa0/24	Desg	FWD	19	128.26		P2p
--------	------	-----	----	--------	--	-----

(输出被省略)

sw3#

说明：因为 SW3 上从端口 F0/19 到达根的 Path Cost 值为 19，从 F0/23 和 F0/24 到达根的 Path Cost 值都为 $19 \times 3 = 57$ 。因此，F0/19 被选为根端口。

(3) 查看 SW4 的根端口

sw4#sh spanning-tree

(输出被省略)

VLAN0010

Spanning tree enabled protocol ieee

Root ID Priority 4106

Address 001a.6c6f.fb00

CCIE LAB认证经验分享千人群：539730342

Cost 38

Port 22 (FastEthernet0/20)

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

Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)

Address 001e.14cf.0980

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

Aging Time 300

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Altn	BLK	19	128.21		P2p
--------	------	-----	----	--------	--	-----

Fa0/20	Root	FWD	19	128.22		P2p
--------	------	-----	----	--------	--	-----

Fa0/23	Altn	BLK	19	128.25		P2p
--------	------	-----	----	--------	--	-----

Fa0/24	Altn	BLK	19	128.26		P2p
--------	------	-----	----	--------	--	-----

(输出被省略)

sw4#

说明：因为 SW4 上从所有端口到达根的 Path Cost 值都为 $19 \times 2 = 38$ ，所以从比较 Path Cost 值，无法选出根端口，接下来比较上一跳交换机 Bridge-ID，也就是比较 SW2 与 SW3 的 Bridge-ID，所以选择往 SW2 的方向，然而通过端口 F0/19 和 F0/20 都可以从 SW2 到达根交换机，所以接下来比较端口 F0/19 和 F0/20 对端交换机端口的优先级，因为 SW2 的 F0/19 端口优先级为 128，而 F0/20 的端口优先级为 112，所以 SW4 选择连接 SW2 的 F0/20 的端口为根端口，即 SW4 的 F0/20 为根端口

5.查看指定端口

（1）查看 SW1 的指定端口

```
sw1#sh spanning-tree
```

（输出被省略）

```
VLAN0010
```

```
Spanning tree enabled protocol ieee
```

```
Root ID    Priority    4106
```

```
Address    001a.6c6f.fb00
```

```
This bridge is the root
```

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

```
Bridge ID  Priority    4106  (priority 4096 sys-id-ext 10)
```

```
Address    001a.6c6f.fb00
```

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

```
Aging Time 300
```

```
Interface    Role Sts Cost    Prio.Nbr Type
```

```
-----  
Fa0/19       Desg FWD 19    128.21 P2p
```

```
Fa0/23       Desg FWD 19    128.25 P2p
```

CCIE LAB认证经验分享千人群：539730342

（输出被省略）

SW1#

说明：在根交换机 SW1 连接 SW2 的网段与连接 SW3 的网段中，当然是根自己的端口离自己最近，所以这两个网段中，选举根交换机上的端口为指定端口，因此，根交换机上所有的端口都应该是指定端口。

（2）查看 SW2 的指定端口

sw2#sh spanning-tree

（输出被省略）

VLAN0010

Spanning tree enabled protocol ieee

Root ID Priority 4106

Address 001a.6c6f.fb00

Cost 19

Port 23 (FastEthernet0/23)

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

Bridge ID Priority 24586 (priority 24576 sys-id-ext 10)

Address 0013.805c.9d00

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

Aging Time 300

CCIE LAB认证经验分享千人群：539730342

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Desg	FWD	19	128.19		P2p
--------	------	-----	----	--------	--	-----

Fa0/20	Desg	FWD	19	112.20		P2p
--------	------	-----	----	--------	--	-----

Fa0/23	Root	FWD	19	128.23		P2p
--------	------	-----	----	--------	--	-----

（输出被省略）

Sw2#

说明：在 SW2 连接 SW4 的两个网段中，同样也是 SW2 上的两个端口离根交换机最近，所以在这两个网段中，选举 SW2 上的端口为指定端口。

（3）查看 SW2 的指定端口

sw3#sh spanning-tree

（输出被省略）

VLAN0010

Spanning tree enabled protocol ieee

Root ID Priority 4106

Address 001a.6c6f.fb00

Cost 19

Port 21 (FastEthernet0/19)

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

CCIE LAB认证经验分享千人群：539730342

Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)

Address 001a.a256.f300

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

Aging Time 300

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Root	FWD	19	128.21		P2p
--------	------	-----	----	--------	--	-----

Fa0/23	Desg	FWD	19	128.25		P2p
--------	------	-----	----	--------	--	-----

Fa0/24	Desg	FWD	19	128.26		P2p
--------	------	-----	----	--------	--	-----

(输出被省略)

Sw3#

说明：在 SW3 连接 SW4 的两个网段中，同样也是 SW3 上的两个端口离根交换机最近，所以在这两个网段中，选举 SW3 上的端口为指定端口。

(4) 查看 SW2 的指定端口

sw4#sh spanning-tree

(输出被省略)

VLAN0010

Spanning tree enabled protocol ieee

Root ID	Priority	4106
---------	----------	------

CCIE LAB认证经验分享千人群：539730342

Address 001a.6c6f.fb00

Cost 38

Port 22 (FastEthernet0/20)

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

Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)

Address 001e.14cf.0980

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

Aging Time 300

Interface Role Sts Cost Prio.Nbr Type

Fa0/19	Altn	BLK	19	128.21	P2p
Fa0/20	Root	FWD	19	128.22	P2p
Fa0/23	Altn	BLK	19	128.25	P2p
Fa0/24	Altn	BLK	19	128.26	P2p

（输出被省略）

sw4#

说明：除了根端口和指定端口，其它的都为落选端口，也就是 SW4 上的 F0/19，F0/23，F0/24，都将被 STP 放入 Blocking 状态，不为用户提供数据转发，以此来防止环路

6.调整 VLAN 30 和 VLAN 40 的路径为 SW1—SW3—SW4。

说明：因为默认 4 个 VLAN 相同配置，所以全部和 VLAN 10 一样，路径为 SW1—SW2—SW4，现只对 VLAN 30 和 VLAN 40 做修改，以调整路径为 SW1—SW3—SW4。

（1）修改 SW3 在 VLAN 30 和 VLAN 40 的 Bridge-ID 优先级

说明：因为选举根端口和指定端口的第一步为比较到根的 Path Cost 值，第二步为比较上一跳 Bridge-ID，而 SW4 从 SW2 到 SW1 和从 SW3 到 SW1 的 Path Cost 值全部是一样的，所以可以选择修改 SW3 在 VLAN 30 和 VLAN 40 的 Bridge-ID 优先级来做调整：

```
sw3(config)#spanning-tree vlan 30,40 priority 20480
```

说明：SW3 在 VLAN 30 和 VLAN 40 的 Bridge-ID 优先级必须比 SW2 的 Bridge-ID 优先级小，才能将 VLAN 30 与 VLAN 40 的流量引过来。

7.查看修改后的 VLAN 30 与 VLAN 40 的路径

说明：因为 VLAN 30 与 VLAN 40 相同配置，所以只查看一个 VLAN 即可。

（1）查看 SW4 上 VLAN 10 与 VLAN 30 的路径对比

```
sw4#sh spanning-tree
```

（输出被省略）

```
VLAN0010
```

```
Spanning tree enabled protocol ieee
```

```
Root ID    Priority    4106
```

```
Address    001a.6c6f.fb00
```

```
Cost        38
```

```
Port       22 (FastEthernet0/20)
```

CCIE LAB认证经验分享千人群：539730342

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

Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)

Address 001e.14cf.0980

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

Aging Time 300

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Altn	BLK	19	128.21		P2p
Fa0/20	Root	FWD	19	128.22		P2p
Fa0/23	Altn	BLK	19	128.25		P2p
Fa0/24	Altn	BLK	19	128.26		P2p

VLAN0030

Spanning tree enabled protocol ieee

Root ID Priority 4126

Address 001a.6c6f.fb00

Cost 38

Port 25 (FastEthernet0/23)

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

CCIE LAB认证经验分享千人群：539730342

Bridge ID Priority 32798 (priority 32768 sys-id-ext 30)

Address 001e.14cf.0980

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

Aging Time 300

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Altn	BLK	19	128.21	P2p	
--------	------	-----	----	--------	-----	--

Fa0/20	Altn	BLK	19	128.22	P2p	
--------	------	-----	----	--------	-----	--

Fa0/23	Root	FWD	19	128.25	P2p	
--------	------	-----	----	--------	-----	--

Fa0/24	Altn	BLK	19	128.26	P2p	
--------	------	-----	----	--------	-----	--

(输出被省略)

sw4#

说明：可以看到，SW4 的 VLAN 10 还是保持原来的路径 SW4—SW2—SW1，而 VLAN 30 的路径已经变成 SW4—SW3—SW1 并且 VLAN 30 的根端口为 F0/23。

8.调整 STP 参数

(1) 调整 SW4 在 VLAN 30 的根端口为 F0/24

说明：因为 SW4 在 VLAN 30 从 F0/23 和 F0/24 到达根的 Path Cost 值都为 $19 \times 2 = 38$ ，所以从比较 Path Cost 值，无法选出根端口，接下来比较上一跳交换机 Bridge-ID，由于都是 SW3，所以 Bridge-ID 相同，接下来比较 F0/23 和 F0/24 对端交换机端口的优先级，但对方优先级都为 128，所以最后选择了本地端口号码小的，即 F0/23 比 F0/24

CCIE LAB认证经验分享千人群：539730342

小，F0/23 被选为根端口，我们现在通过修改本地 F0/24 对端设备的端口优先级来调整路径，也就是修改 SW3 的 F0/24 的优先级：

```
sw3(config)#int f0/24
```

```
sw3(config-if)#spanning-tree vlan 30 port-priority 112
```

说明：端口优先级为 16 的整数倍。

（2）查看 SW4 在 VLAN 30 的根端口

```
sw4#sh spanning-tree vlan 30
```

VLAN0030

Spanning tree enabled protocol ieee

Root ID Priority 4126

Address 001a.6c6f.fb00

Cost 38

Port 26 (FastEthernet0/24)

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

Bridge ID Priority 32798 (priority 32768 sys-id-ext 30)

Address 001e.14cf.0980

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

Aging Time 300

CCIE LAB认证经验分享千人群：539730342

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Altn	BLK	19	128.21		P2p
Fa0/20	Altn	BLK	19	128.22		P2p
Fa0/23	Altn	BLK	19	128.25		P2p
Fa0/24	Root	FWD	19	128.26		P2p

sw4#

说明：因为选举时，比较对方的端口优先级，成功调整了路径，此时的根端口已变为 F0/24。

(3) 修改 SW4 在 VLAN 10 的 hello 时间为 3 秒，max-age 为 25 秒，forward delay 为 10 秒

sw4(config)#spanning-tree vlan 10 hello-time 3

sw4(config)#spanning-tree vlan 10 max-age 30

sw4(config)#spanning-tree vlan 10 forward-time 10

(4) 查看 SW4 在 VLAN 10 的 hello 时间，max-age，forward delay

sw4#sh spanning-tree

(输出被省略)

VLAN0010

CCIE LAB认证经验分享千人群：539730342

Spanning tree enabled protocol ieee

Root ID Priority 4106

Address 001a.6c6f.fb00

Cost 38

Port 22 (FastEthernet0/20)

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

Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)

Address 001e.14cf.0980

Hello Time 3 sec Max Age 30 sec Forward Delay 10 sec

Aging Time 300



Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Altn	BLK	19	128.21		P2p
Fa0/20	Root	FWD	19	128.22		P2p
Fa0/23	Altn	BLK	19	128.25		P2p
Fa0/24	Altn	BLK	19	128.26		P2p

VLAN0020

CCIE LAB认证经验分享千人群：539730342

Spanning tree enabled protocol ieee

Root ID Priority 4116

Address 001a.6c6f.fb00

Cost 38

Port 22 (FastEthernet0/20)

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

Bridge ID Priority 32788 (priority 32768 sys-id-ext 20)

Address 001e.14cf.0980

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

Aging Time 300



Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Altn	BLK	19	128.21		P2p
Fa0/20	Root	FWD	19	128.22		P2p
Fa0/23	Altn	BLK	19	128.25		P2p
Fa0/24	Altn	BLK	19	128.26		P2p

（输出被省略）

sw4#

说明：可以看到，修改的时间只对 VLAN 10 生效，VLAN 20 还是保持原状，PVST+ 可以单独修改每个 VLAN 的参数。

9.强制指定根与备份根

（1）指定 SW2 为 VLAN 10 的根

```
sw2(config)#spanning-tree vlan 10 root primary
```

（2）在 SW2 查看 VLAN 10 的根

```
sw2#sh spanning-tree vlan 10
```

VLAN0010

Spanning tree enabled protocol ieee

Root ID Priority 4106

Address 0013.805c.9d00

This bridge is the root

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

Bridge ID Priority 4106 (priority 4096 sys-id-ext 10)

Address 0013.805c.9d00

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

Aging Time 15

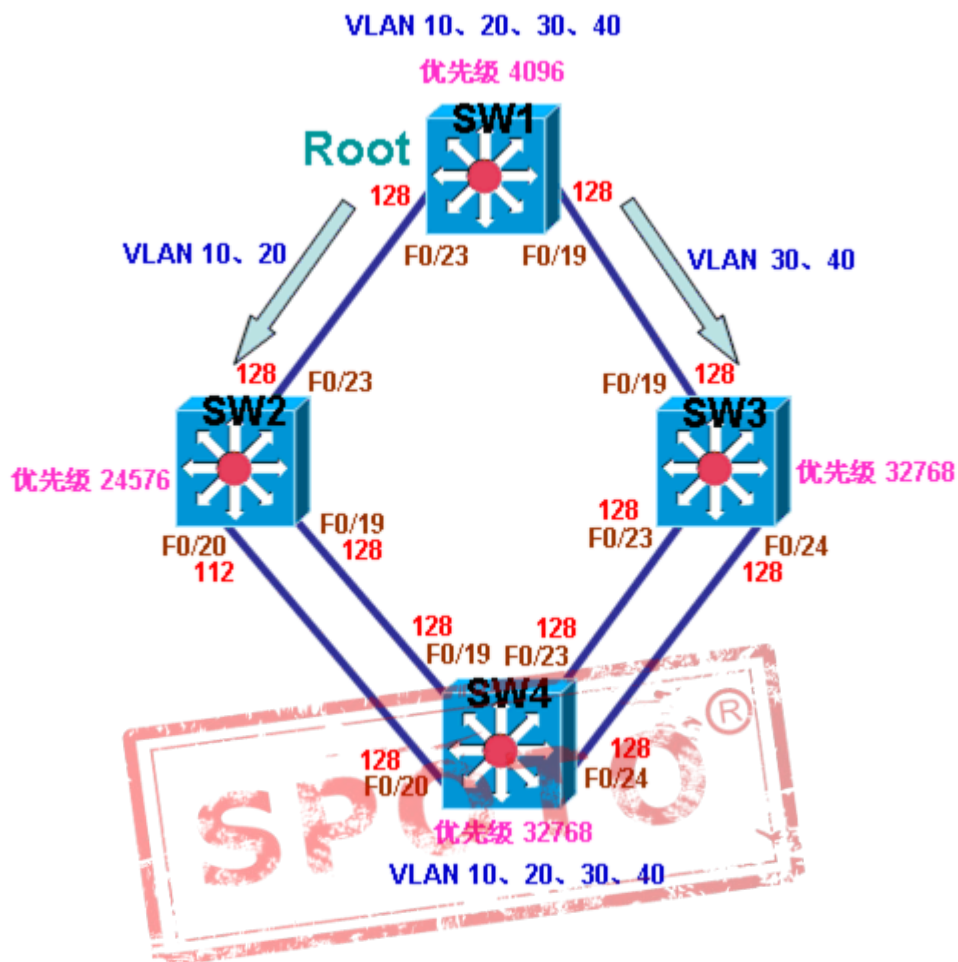
Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

```
-----  
  
Fa0/19      Desg FWD 19      128.19  P2p  
  
Fa0/20      Desg FWD 19      112.20  P2p  
  
Fa0/23      Desg FWD 19      128.23  P2p
```

```
sw2#
```

重点说明：当使用命令强制指定某交换机为根后，此交换机将通过修改一个比当前根交换机更高优先级的 **Bridge-ID**，以此来抢夺根交换机的角色，如果命令再到别的交换机上输入，那么那台交换机将再次抢夺根交换机的角色，因为它可以修改自己的 **Bridge-ID** 比当前根更高的优先级，所以此命令最后在网络中的哪台交换机上输入后，哪台交换机就能成为根交换机，但是也有个限度，因为交换机的 **Bridge-ID** 不能自动改的比 1 小，又不能改 **MAC** 地址，所以如果需要修改优先级到 1 以下才能抢夺根交换机的角色，那么此命令将提示错误。

配置 MSTP



1.配置 MSTP

(1) 改变所有交换机的 STP 模式为 MSTP

```
Sw1(config)#spanning-tree mode mst
```

```
Sw2(config)#spanning-tree mode mst
```

```
Sw3(config)#spanning-tree mode mst
```

```
Sw4(config)#spanning-tree mode mst
```

(2) 映射 VLAN 到实例

```
sw1(config)#spanning-tree mst configuration
```

```
sw1(config-mst)#name ccie
```

```
sw1(config-mst)#revision 1
```

```
sw1(config-mst)#instance 1 vlan 10,20
```

```
sw1(config-mst)#instance 2 vlan 30,40
```

说明：其它交换机配置和 SW1 配置完全相同，必须 region name 和 revision number 完全相同，否则属于不同的 region。

2.控制 VLAN 10 和 VLAN 20（实例 1）的路径为 SW1—SW2—SW4，VLAN 30 和 VLAN 40（实例 2）的路径为 SW1—SW3—SW4。

(1)配置 SW1 为实例 1 和实例 2 的根交换机

```
sw1(config)#spanning-tree mst 1 root primary
```

```
sw1(config)#spanning-tree mst 2 root primary
```

(2) 控制 SW4 在实例 1 连 SW2 的端口 Path Cost 值为 10

```
sw4(config)#int range f0/19-20
```

```
sw4(config-if-range)#spanning-tree mst 1 cost 10
```

(3) 控制 SW4 在实例 2 连 SW3 的端口 Path Cost 值为 10

```
sw4(config)#int ran f0/23-24
```

```
sw4(config-if-range)#spanning-tree mst 2 cost 10
```

3.查看 STP 状态

CCIE LAB认证经验分享千人群：539730342

(1) 查看根交换机

sw1#sh spanning-tree

(输出被省略)

MST1

Spanning tree enabled protocol mstp

Root ID Priority 24577

Address 001a.6c6f.fb00

This bridge is the root

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

Bridge ID Priority 24577 (priority 24576 sys-id-ext 1)

Address 001a.6c6f.fb00

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

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Desg	FWD	200000	128.21		P2p
--------	------	-----	--------	--------	--	-----

Fa0/23	Desg	FWD	200000	128.25		P2p
--------	------	-----	--------	--------	--	-----

MST2

Spanning tree enabled protocol mstp

CCIE LAB认证经验分享千人群：539730342

Root ID Priority 24578

Address 001a.6c6f.fb00

This bridge is the root

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

Bridge ID Priority 24578 (priority 24576 sys-id-ext 2)

Address 001a.6c6f.fb00

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

Interface	Role	Sts	Cost	Prio.	Nbr	Type
<hr/>						
Fa0/19	Desg	FWD	200000	128.21		P2p
Fa0/23	Desg	FWD	200000	128.25		P2p

sw1#

说明：可以看到 SW1 已经成为实例 1 和实例 2 的根交换机。

（2）查看 SW4 的路径

sw4#sh spanning-tree

（输出被省略）

CCIE LAB认证经验分享千人群：539730342

MST1

Spanning tree enabled protocol mstp

Root ID Priority 24577

Address 001a.6c6f.fb00

Cost 200010

Port 21 (FastEthernet0/19)

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

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)

Address 001e.14cf.0980

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

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/19	Root	FWD	10	128.21		P2p
--------	------	-----	----	--------	--	-----

Fa0/20	Altn	BLK	10	128.22		P2p
--------	------	-----	----	--------	--	-----

Fa0/23	Altn	BLK	200000	128.25		P2p
--------	------	-----	--------	--------	--	-----

Fa0/24	Altn	BLK	200000	128.26		P2p
--------	------	-----	--------	--------	--	-----

MST2

CCIE LAB认证经验分享千人群：539730342

Spanning tree enabled protocol mstp

Root ID Priority 24578

Address 001a.6c6f.fb00

Cost 200010

Port 25 (FastEthernet0/23)

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

Bridge ID Priority 32770 (priority 32768 sys-id-ext 2)

Address 001e.14cf.0980

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

Interface Role Sts Cost Prio.Nbr Type

Fa0/19 Altn BLK 200000 128.21 P2p

Fa0/20 Altn BLK 200000 128.22 P2p

Fa0/23 Root FWD 10 128.25 P2p

Fa0/24 Altn BLK 10 128.26 P2p

sw4#

说明：可以看到，实例 1 与实例 2 的流量已经分担到两条不同的路径上，既实现了与 PVST+相同的负载效果，也节省了系统资源，因为只有两个 STP 实例，而 PVST+要 4 个 STP 实例。

Spanning-Tree Feature

Port Fast

因为一个默认情况下的交换机端口，在交换机启动后，由于 STP 的原因，端口状态需要从 initialization（初始化）到 blocking，从 blocking 到 listening，从 listening 到 learning，从 learning 到 forwarding，其中经历了两个 forward delay，也就是说一个端口在交换机启动后，至少需要 30 秒后才能够为用户提供数据转发。对于一个连接了主机或服务器的端口，进行 STP 计算是毫无必要的，因为此类端口即使直接转发数据，也不会造成环路，并且 30 秒的时间对于需要立即传递数据的主机或服务器来说，是漫长的，因此，此类端口可以配置为跳过 STP 的计算，从而直接过渡到 forwarding 状态。

此类端口通常称为边缘端口，在思科交换机上，通过配置 Port Fast 功能，便可以使接口跳过 STP 的计算，从而直接过渡到 forwarding 状态。

access 接口和 Trunk 接口都可以配置 Port Fast 功能。如果将交换机连交换机的接口变成 Port Fast，则是制造环路。

当开启了 Port Fast 功能的接口，如果在接口上收到 BPDU 后，就认为对端连接的是交换机，而并非主机或服务器，因此默认在接口收到 BPDU 后会立即关闭该接口的 Port Fast 功能。

配置



1.在接口下配置 Port Fast

(1) 将 SW2 的端口 F0/23 和 F0/24 改成三层接口

说明：因为如果 SW2 的端口是二层接口，那么就会向 SW1 发送 BPDU，最终会造成 SW1 由于收到 BPDU 而关闭 Port Fast 功能，所以就无法验证 Port Fast。

```
sw2(config)#int ran f0/23 - 24
```

```
sw2(config-if-range)#no switchport
```

说明：禁止从端口上向 SW1 发送 BPDU。

(2) 在 SW1 的 F0/23 和 F0/24 上开启 Port Fast

说明：access 和 trunk 接口模式都可以配置

```
sw1(config)#int ran f0/23 - 24
```

```
sw1(config-if-range)#switchport mode access
```

```
sw1(config-if-range)#spanning-tree portfast
```

说明：将端口变为静态 access，再开 portfast。（无论什么模式的接口都可以开启 Port Fast）

(3) 验证 Port Fast

CCIE LAB认证经验分享千人群：539730342

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
VLAN0001      enabled
```

```
sw1#
```

```
sw1#sh spanning-tree interface f0/24 portfast
```

```
VLAN0001      enabled
```

```
sw1#
```

说明：端口 F0/23 和 F0/24 已经开启 portfast 功能。

(4) 在 SW2 的端口 F0/23 向 SW1 发送 BPDU

```
sw2(config)#int f0/23
```

```
sw2(config-if)#switchport
```

说明：只要将 SW2 的端口 F0/23 变成二层端口，便可以从此端口向外发送 BPDU。

(5) 查看 SW1 的端口的 portfast 状态：

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
VLAN0001      disabled
```

```
sw1#
```

```
sw1#sh spanning-tree interface f0/24 portfast
```

```
VLAN0001      enabled
```

```
sw1#
```

说明：可以看见，SW1 的端口 F0/23，在收到 BPDU 后，portfast 功能自动丢失。

2.在全局模式下配置 Port Fast（只能对 access 接口生效）

（1）将 SW2 的端口 F0/23 和 F0/24 改成三层接口

```
sw2(config)#int ran f0/23 - 24
```

```
sw2(config-if-range)#no switchport
```

（2）将 SW1 的端口配置为 access

```
sw1(config)#int ran f0/23 - 24
```

```
sw1(config-if-range)#switchport mode dynamic desirable
```

说明：因为对方是三层端口，在本地配置 DTP 后，会自动形成 access 模式。

（3）查看 SW1 的端口状态

```
sw1#show interfaces f0/23 switchport
```

Name: Fa0/23

Switchport: Enabled

Administrative Mode: dynamic desirable

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

（输出被省略）

sw1#

sw1#show interfaces f0/24 switchport

Name: Fa0/24

Switchport: Enabled

Administrative Mode: dynamic desirable

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

（输出被省略）

sw1#

说明：DTP 已经将本地端口变为 access 模式。

(4)在 SW1 全局开启 Port Fast

sw1(config)#spanning-tree portfast default

(5) 查看 SW1 上端口的 Port Fast 状态

sw1#sh spanning-tree interface f0/23 portfast

VLAN0001 enabled

sw1#sh spanning-tree interface f0/24 portfast

```
VLAN0001    enabled
```

```
sw1#
```

说明：SW1 上的 access 端口受全局配置影响，已经变成 Port Fast 端口。

（6）验证同上，省略

BPDU Guard

因为开启了 Port Fast 功能的端口，会跳过 STP 的计算，从而直接过渡到 forwarding 状态。当端口连接的是主机或服务器，这样的操作不会有任何问题，但如果连接的是交换机，就会收到 BPDU，就证明在此接口开启 Port Fast 功能是错误的配置。为了杜绝此类错误配置，BPDU Guard 功能可以使端口在收到 BPDU 时，立即被 shutdown 或进入 err-disabled 状态。

BPDU Guard 可以在接口下或全局开启，但操作会有所不同。

如果 BPDU Guard 是全局开启，则只对 portfast 端口有影响，当 portfast 端口收到 BPDU 后，会 shutdown 此端口，需要注意，某些型号的交换机会将接口 error-disabled。

如果 BPDU Guard 是接口下开启，将对任何端口有影响，无论是正常端口还是 portfast 端口；当端口收到 BPDU 后，会变成 error-disabled 状态。

配置



CCIE LAB认证经验分享千人群：539730342

1. 在全局模式下配置 BPDU Guard（只对 Port Fast 端口有影响）

（1）将 SW2 的端口 F0/23 和 F0/24 改成三层接口

```
sw2(config)#int ran f0/23 - 24
```

```
sw2(config-if-range)#no switchport
```

说明：禁止从端口上向 SW1 发送 BPDU。

（2）将 SW1 的端口 F0/23 配置为 Port Fast，F0/24 为正常端口

```
sw1(config)#int f0/23
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#spanning-tree portfast
```

```
sw1(config)#int f0/24
```

```
sw1(config-if)#switchport mode access
```

（3）查看 SW1 的端口 F0/23 和 F0/24 的状态

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
VLAN0001      enabled
```

```
sw1#
```

```
sw1#sh spanning-tree interface f0/24 portfast
```

```
VLAN0001      disabled
```

CCIE LAB认证经验分享千人群：539730342

sw1#

sw1#sh protocols f0/23

FastEthernet0/23 is up, line protocol is up

sw1#sh protocols f0/24

FastEthernet0/24 is up, line protocol is up

sw1#

说明：SW1 的端口 F0/23 已经变成 Port Fast 状态，而 F0/24 为正常端口，并且两个端口都为正常 UP 状态。

(4) 在 SW1 全局开启 BPDU Guard (只对 Port Fast 端口有影响)

sw1(config)#spanning-tree portfast bpduguard default

(5) 在 SW2 的端口 F0/23 和 F0/24 向 SW1 发送 BPDU，测试 BPDU Guard

sw2(config)#int ran f0/23 - 24

sw2(config-if-range)#switchport

说明：只要将 SW2 的端口 F0/23 和 F0/24 变成二层端口，便可以从此端口向外发送 BPDU。

(6) 查看 SW1 的端口状态：

sw1#sh spanning-tree interface f0/23 portfast

no spanning tree info available for FastEthernet0/23

sw1#

sw1#sh spanning-tree interface f0/24 portfast

VLAN0001 disabled

sw1#

sw1#sh protocols f0/23

FastEthernet0/23 is down, line protocol is down

sw1#

sw1#sh protocols f0/24

FastEthernet0/24 is up, line protocol is up

sw1#sh int f0/23

FastEthernet0/23 is down, line protocol is down (err-disabled)

（输出被省略）

sw1#

说明：可以看见，SW1 的 portfast 端口 F0/23 收到 BPDU 后，受到 BPDU Guard 的影响，端口被 shutdown，并且变成 error-disabled，（某些型号不会），而全局 BPDU Guard 不能影响非 portfast 端口，所以 F0/24 还是正常状态。

2.在接口模式下配置 BPDU Guard（将对任何端口生效）

CCIE LAB认证经验分享千人群：539730342

(1) 将 SW2 的端口 F0/23 和 F0/24 改成三层接口

```
sw2(config)#int ran f0/23 - 24
```

```
sw2(config-if-range)#no switchport
```

说明：禁止从端口上向 SW1 发送 BPDU。

(2) 将 SW1 的端口 F0/23 配置为 portfast，将 F0/24 配置为正常端口

```
sw1(config)#int f0/23
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#spanning-tree portfast
```

```
sw1(config)#int f0/24
```

```
sw1(config-if)#switchport mode access
```

(3) 在 SW1 的端口 F0/23 和 F0/24 开启 BPDU Guard

```
sw1(config)#int ran f0/23 - 24
```

```
sw1(config-if-range)#spanning-tree bpduguard enable
```

(4) 查看 SW1 的端口 F0/23 和 F0/24 的状态

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
VLAN0001      enabled
```

```
sw1#
```

CCIE LAB认证经验分享千人群：539730342

```
sw1#sh spanning-tree interface f0/24 portfast
```

```
VLAN0001      disabled
```

```
sw1#
```

```
sw1#sh protocols f0/23
```

```
FastEthernet0/23 is up, line protocol is up
```

```
sw1#
```

```
sw1#sh protocols f0/24
```

```
FastEthernet0/24 is up, line protocol is up
```

```
sw1#
```

说明： SW1 上的端口 F0/23 为 portfast 状态，F0/24 为正常状态，并且状态都为 UP。



（5）在 SW2 的端口 F0/23 和 F0/24 向 SW1 发送 BPDU，测试 BPDU Guard

```
sw2(config)#int range f0/23 - 24
```

```
sw2(config-if-range)#switchport
```

（6）查看 SW1 的端口状态：

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
no spanning tree info available for FastEthernet0/23
```

```
sw1#
```

```
sw1#sh spanning-tree interface f0/24 portfast
```

```
no spanning tree info available for FastEthernet0/24
```

```
sw1#
```

```
sw1#sh int f0/23
```

```
FastEthernet0/23 is down, line protocol is down (err-disabled)
```

（输出被省略）

```
sw1#sh int f0/24
```

```
FastEthernet0/24 is down, line protocol is down (err-disabled)
```

（输出被省略）

```
sw1#
```

说明：SW1 的端口在收到 BPDU 后，受到 BPDU Guard 的影响，无论是正常端口还是 portfast 端口，都被 err-disabled。

BPDU Filtering

BPDU Filtering 可以过滤掉在接口上发出或收到的 BPDU，这就相当于关闭了接口的 STP，将会有引起环路的可能。

BPDU Filtering 的配置同样也分两种，可以在接口下或在全局模式开启，但是不同的模式开启，会有不同效果。

如果 BPDU Filtering 是全局开启的，则只能在开启了 portfast 的接口上过滤 BPDU，并且只能过滤掉发出的 BPDU，并不能过滤收到的 BPDU，因为 BPDU Filtering 的设计目的是当交换机端口上连接的是主机或服务器时，就没有必要向对方发送 BPDU，所以要过滤掉 BPDU，但如果连接的是交换机，则会收到 BPDU，而且会引起环路，所以这样的情况，配置 BPDU Filtering 就是错误的。而当一个开启了 portfast 功能的

接口，在开启了 BPDU Filtering 后，如果还能收到 BPDU，则 BPDU Filtering 特性会丢失，因此，还会造成接口 portfast 特性的丢失。

如果是在接口模式下开启的，则可以过滤掉任何接口收到和发出的 BPDU。(此理论为重点)

配置



1. 在全局模式下配置 BPDU Filtering（只能过滤 portfast 上的 BPDU）

（1）将 SW2 的端口 F0/23 和 F0/24 改成三层接口

```
sw2(config)#int ran f0/23 - 24
```

```
sw2(config-if-range)#no switchport
```

说明：禁止从端口上向 SW1 发送 BPDU。

（2）将 SW1 的端口 F0/23 配置为 Port Fast，将 F0/24 配置为正常端口，但开启 BPDU Guard

```
sw1(config)#int f0/23
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#spanning-tree portfast
```

```
sw1(config)#int f0/24
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#spanning-tree bpduguard enable
```

(3) 查看 SW1 的端口 F0/23 和 F0/24 的状态

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
VLAN0001      enabled
```

```
sw1#
```

```
sw1#sh int f0/24
```

```
FastEthernet0/24 is up, line protocol is up (connected)
```

(输出被省略)

```
sw1#
```

说明：SW1 的端口 F0/23 为 Port Fast 状态，F0/24 为正常状态，并且状态为 UP。

(4) 在 SW1 上全局开启 BPDU Filtering (只能过滤 portfast 上的 BPDU)

CCIE LAB认证经验分享千人群：539730342

```
sw1(config)#spanning-tree portfast bpdupfilter default
```

(5) 在 SW2 的端口 F0/23 和 F0/24 向 SW1 发送 BPDU，测试 BPDU Filtering

```
sw2(config)#int range f0/23 - 24
```

```
sw2(config-if-range)#switchport
```

说明：只要将 SW2 的端口 F0/23 和 F0/24 变成二层端口，便可以从此端口向外发送 BPDU。

(6) 查看 SW1 的端口状态：

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
VLAN0001 disabled
```

```
sw1#
```

```
sw1#sh int f0/24
```

```
FastEthernet0/24 is down, line protocol is down (err-disabled)
```

(输出被省略)

```
sw1#
```

说明：因为 F0/24 是普通端口，全局配置的 BPDU Filtering 不能过滤普通端口上的 BPDU，所以收到了 BPDU 后，但由于 BPDU Guard，最后端口被 err-disabled。

而 F0/23 是开启了 portfast 功能的接口，在开启了 BPDU Filtering 后，如果还能收到 BPDU，则 BPDU Filtering 特性会丢失，因此，造成了端口 F0/23 的 portfast 特性丢失。

2.在接口模式下配置 BPDU Filtering（将对任何端口生效）

（1）将 SW2 的端口 F0/23 和 F0/24 改成三层接口

```
sw2(config)#int ran f0/23 - 24
```

```
sw2(config-if-range)#no switchport
```

说明：禁止从端口上向 SW1 发送 BPDU。

（2）将 SW1 的端口 F0/23 配置为 portfast，将 F0/24 配置为正常端口

```
sw1(config)#int f0/23
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#spanning-tree portfast
```

```
sw1(config)#int f0/24
```

```
sw1(config-if)#switchport mode access
```

（3）在 SW1 的端口 F0/24 开启 BPDU Guard

```
sw1(config)#int ran f0/23 - 24
```

```
sw1(config-if-range)#spanning-tree bpduguard enable
```

（4）查看 SW1 的端口 F0/23 和 F0/24 的状态

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
VLAN0001      enabled
```

sw1#

sw1#sh spanning-tree interface f0/24 portfast

VLAN0001 disabled

sw1#

sw1#sh protocols f0/23

FastEthernet0/23 is up, line protocol is up

sw1#

sw1#sh protocols f0/24

FastEthernet0/24 is up, line protocol is up

sw1#

说明：SW1 上的端口 F0/23 为 portfast 状态，F0/24 为正常状态，并且状态都为 UP。

（5）在 SW1 的端口 F0/23 和 F0/24 下开启 BPDU Filtering

sw1(config)#int range f0/23 - 24

sw1(config-if-range)#spanning-tree bpdupfilter enable

（6）在 SW2 的端口 F0/23 和 F0/24 向 SW1 发送 BPDU，测试 BPDU Filtering

sw2(config)#int range f0/23 - 24

sw2(config-if-range)#switchport

(7) 查看 SW1 的端口状态：

```
sw1#sh spanning-tree interface f0/23 portfast
```

```
VLAN0001      enabled
```

```
sw1#
```

```
sw1#sh spanning-tree interface f0/24 portfast
```

```
VLAN0001      disabled
```

```
sw1#
```

```
sw1#sh protocols f0/23
```

```
FastEthernet0/23 is up, line protocol is up
```

```
sw1#sh protocols f0/24
```

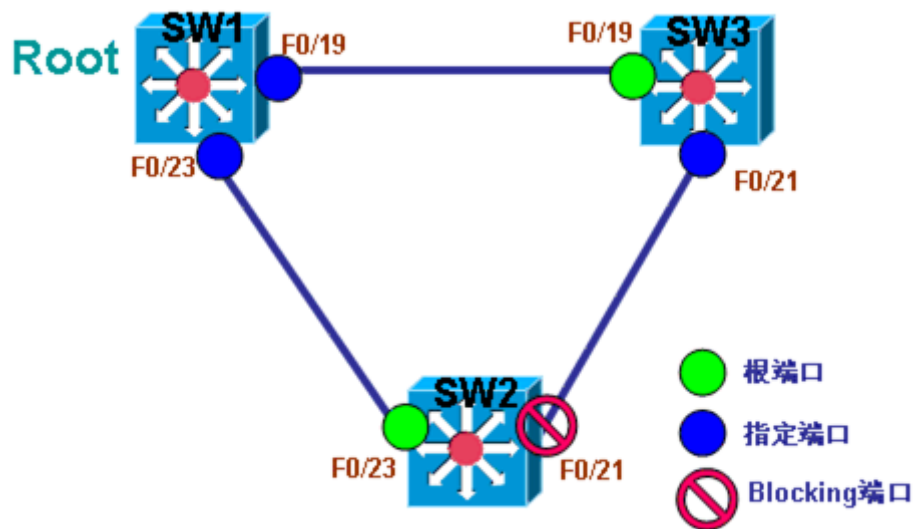
```
FastEthernet0/24 is up, line protocol is up
```

```
sw1#
```

说明：接口下开启的 BPDU Filtering，过滤掉了正常端口下的 BPDU，也过滤掉了 portfast 端口下的 BPDU。

UplinkFast

以下图为例来解释 UplinkFast 的功能与作用：



在如上图一个运行 STP 的网络环境中，SW1 被选为根交换机，SW2 与 SW3 为普通交换机，其中 SW3 上的两个端口都为转发状态；SW2 上的端口 F0/23 为转发状态，而 F0/21 却为 Blocking 状态，因此无论 SW2 去往根交换机 SW1 还是去往 SW3，都只能从 F0/23 走。

当 SW2 的端口 F0/23 失效后，那么 SW2 去往 SW1 和 SW3 的唯一出口也就断掉了，如果 SW2 要启用 Blocking 的端口 F0/21，在 CST 下必须等待一个 max-age 时间（20 秒），再加两个 forward delay（共 30 秒），总共 50 秒的时间后，才能启用 Blocking 端口，即使是 RSTP，也有可能要等待 6 秒才能启用 Blocking 端口。

对于 SW2 来说，自己的端口 F0/23 断掉后，完全可以立刻检测出来，并且完全可以立刻启用 Blocking 的端口 F0/21，从而缩短网络的故障恢复时间。开启了 UplinkFast 交换机就能够在检测到交换机上直连的转发状态的接口失效后，立即启用 Blocking 的端口，提供网络快速收敛和恢复的功能。

很明显，UplinkFast 要真正起到作用，交换机上必须有 Blocking 的端口存在才行，因为根交换机上所有的接口最终都会变成指定端口，所以 UplinkFast 在根交换机上开启是毫无意义的，UplinkFast 只适合在非根交换机，即普通交换机上开启，普通交换机有时又称接入交换机。

★UplinkFast 只能在交换机上全局开启，不能针对 VLAN 单独开启，也不支持 MSTP 模式。

★UplinkFast 恢复网络的时间大约在 1 到 5 秒。

配置

1 在非根交换机上开启 UplinkFast

(1)在交换机 SW2 上开启 UplinkFast

```
Sw2(config)#spanning-tree uplinkfast
```

(2) 在交换机 SW3 上开启 UplinkFast

```
sw3(config)#spanning-tree uplinkfast
```

2 查看 UplinkFast

(1)查看交换机 SW2 的 UplinkFast

```
sw2#sh spanning-tree uplinkfast
```

```
UplinkFast is enabled
```

```
Station update rate set to 150 packets/sec.
```

```
UplinkFast statistics
```

```
-----
```

```
Number of transitions via uplinkFast (all VLANs)      : 3
```

```
Number of proxy multicast addresses transmitted (all VLANs) : 0
```


Name	Interface List
------	----------------

VLAN0001	Fa0/23(fwd), Fa0/21
----------	---------------------

sw2#

说明：SW2 在直连活动链路 F0/23 失效后，可将 F0/21 恢复。

(2)查看交换机 SW3 的 UplinkFast

sw3#sh spanning-tree uplinkfast

UplinkFast is enabled

Station update rate set to 150 packets/sec.



UplinkFast statistics

Number of transitions via uplinkFast (all VLANs) : 0

Number of proxy multicast addresses transmitted (all VLANs) : 0

Name	Interface List
------	----------------

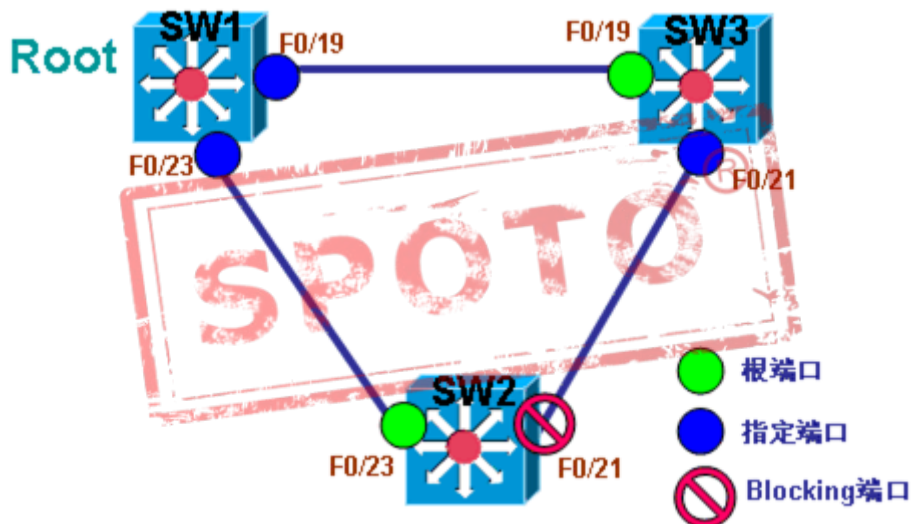
VLAN0001	Fa0/19(fwd)
----------	-------------

sw3#

说明：SW3 在直连活动链路 F0/19 失效后，没有可恢复的端口。

BackboneFast

还是以下图为例来解释 BackboneFast 的功能与作用：



还是与 UplinkFast 同样的网络环境，当 SW2 上直连的活动链路 F0/23 断掉之后，SW2 可以立刻检测出来，并且通过 UplinkFast 功能可以立即启用 Blocking 的端口 F0/21，提供网络快速收敛和恢复的功能。

交换机 SW3 无论去往根交换机 SW1 还是去往 SW2，都只能从 F0/19 走，如果 SW3 上的 F0/19 中断了，那么 SW3 去往 SW1 和 SW2 的唯一出口也就断掉了，此时的 SW3 不能与外界通信，正常情况下，需要等待 SW2 将自己的端口 F0/21 从 Blocking

CCIE LAB认证经验分享千人群：539730342

状态过渡到转发状态后，才能为 SW3 提供一条新的通路。然而，即使是开启了 UplinkFast 功能的 SW2，也并不能在 SW3 的 F0/19 中断后立刻检测到，因为 UplinkFast 只能检测到自己直连的链路中断，并不能检测到远程链路中断。要在 SW3 的 F0/19 中断时，让其它所有交换机都检测到，从而为中断链路的交换机打开一条新的通路，需要在网络中所有交换机上开启 BackboneFast 功能。

因为正常网络中，除了根交换机，其它交换机不能发出 BPDU，所以 SW3 不能发送 BPDU，如果网络中出现了除根交换机发出的 configuration BPDU 之外的其它优先级更低的 BPDU，则称为 Inferior BPDU，当交换机收到 Inferior BPDU 时，默认是丢弃处理。

如果 SW3 上开启了 BackboneFast 功能，则当自己使用中的链路中断时，并且自己又没有 Blocking 状态的端口可以立即启用来代替活动链路，在这种情况下，就可以以自己为根交换机，向网络中发出 Inferior BPDU，inferior BPDU 表示一台交换机即是根交换机，又是普通交换机，inferior BPDU 用来宣告自己的链路中断，已经与网络失去联系。当其它非根交换机，当 SW2 在 Blocking 端口收到 Inferior BPDU 后，如果自己也开启了 BackboneFast 功能，就会将 Blocking 端口变成转发状态，还会向根交换机发出一个 root link query (RLQ) 根链路查询，并且开启了 BackboneFast 功能的根交换机会做出回应，此时 SW2 就可以立刻启用自己 Blocking 状态的端口 F0/21，为 SW3 连接网络提供一条新的通路。

从上述可以看出，要网络中所有的交换机都能理解 Inferior BPDU，要全部配合工作，为链路中断的交换机开辟一条新通路，就必须在网络中所有交换机上都开启 BackboneFast 功能。

所以在开启 BackboneFast 功能时，需要在网络中所有交换机上开启，不能针对 VLAN 单独开启，也不支持 MSTP 模式。BackboneFast 是对 UplinkFast 的强化与补充。

配置

1. 在非根交换机上开启 BackboneFast

(1) 在交换机 SW1 上开启 BackboneFast

```
sw1(config)#spanning-tree backbonefast
```

(2) 在交换机 SW2 上开启 **BackboneFast**

```
sw2(config)#spanning-tree backbonefast
```

(3) 在交换机 SW3 上开启 **BackboneFast**

```
sw3(config)#spanning-tree backbonefast
```

2.查看 **BackboneFast**

(1)查看交换机 SW1 的 **BackboneFast**

```
sw1#sh spanning-tree backbonefast
```

BackboneFast is enabled

BackboneFast statistics

Number of transition via backboneFast (all VLANs) : 0

Number of inferior BPDUs received (all VLANs) : 0

Number of RLQ request PDUs received (all VLANs) : 0

Number of RLQ response PDUs received (all VLANs) : 0

Number of RLQ request PDUs sent (all VLANs) : 0

Number of RLQ response PDUs sent (all VLANs) : 0

```
sw1#
```

(2)查看交换机 SW2 的 **BackboneFast**

```
sw2#sh spanning-tree backbonefast
```

```
BackboneFast is enabled
```

```
BackboneFast statistics
```

```
-----
```

```
Number of transition via backboneFast (all VLANs)      : 0
```

```
Number of inferior BPDUs received (all VLANs)          : 0
```

```
Number of RLQ request PDUs received (all VLANs)        : 0
```

```
Number of RLQ response PDUs received (all VLANs)       : 0
```

```
Number of RLQ request PDUs sent (all VLANs)            : 0
```

```
Number of RLQ response PDUs sent (all VLANs)           : 0
```

```
sw2#
```

(3)查看交换机 SW3 的 BackboneFast

```
sw3#sh spanning-tree backbonefast
```

```
BackboneFast is enabled
```

```
BackboneFast statistics
```

```
-----
```

```
Number of transition via backboneFast (all VLANs)      : 0
```

```
Number of inferior BPDUs received (all VLANs)          : 0
```

Number of RLQ request PDUs received (all VLANs) : 0

Number of RLQ response PDUs received (all VLANs) : 0

Number of RLQ request PDUs sent (all VLANs) : 0

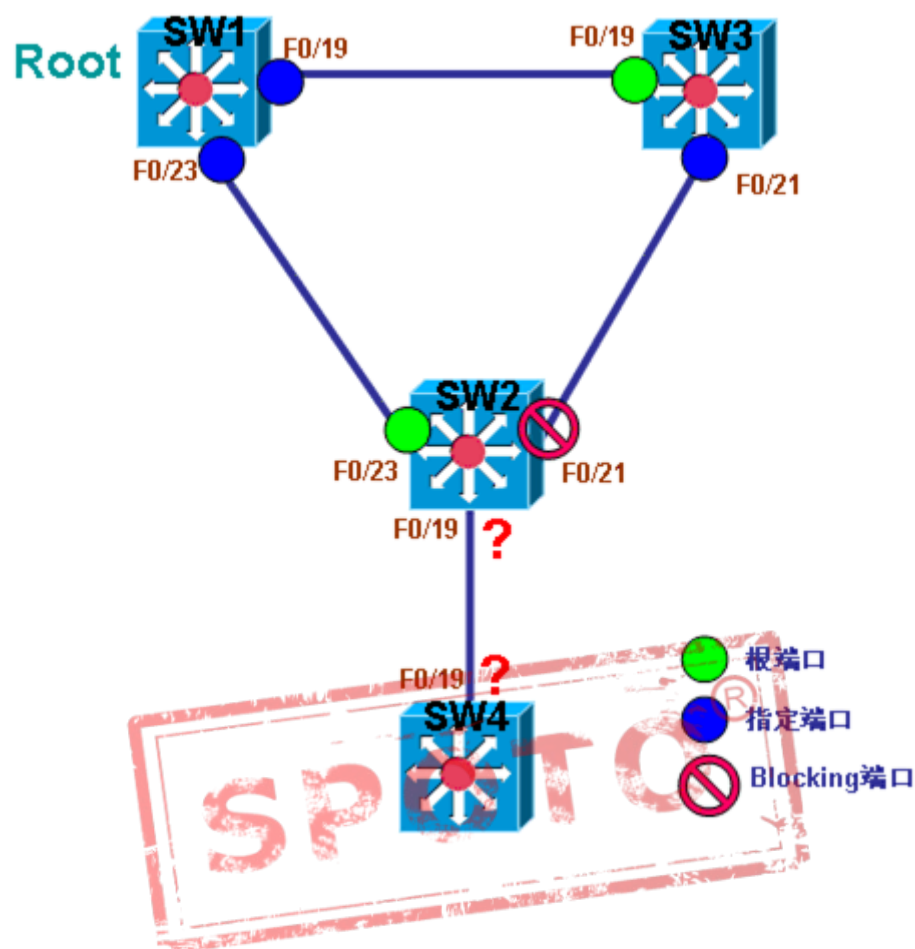
Number of RLQ response PDUs sent (all VLANs) : 0

sw3#

Root Guard

以下图为例来解释 Root Guard 的功能与作用：





在上图中，交换机 SW1, SW2 与 SW3 为网络中运行正常的交换机，其中 SW1 被选为根交换机，当其它交换机之间要通信，都必须选出一个去往根交换机的端口，也就是根端口，所以当 SW2 与 SW3 承认 SW1 为网络中的根交换机时，SW2 便将连接 SW1 的端口 F0/23 选为根端口，SW3 将连 SW1 的端口 F0/19 选为根端口，此后网络通信正常。

考虑到网络的合理性与稳定性，将 SW1 选为根交换机是最佳选择，如果要将其它交换机选为根交换机或网络需要变动，都会引起不必要的麻烦。由于可以任意将一台交换机接入网络，而新接入的交换机，有很大的可能会因为自己拥有更高的 Bridge-ID 而抢夺当前根交换机的角色，这样就会引起网络的麻烦。上图中新加入的交换机 SW4，如果拥有比当前根交换机 SW1 更高的 Bridge-ID，就会抢夺根交换机的角色。如果 SW2 要承认 SW4 为根交换机，就必须将连接 SW4 的端口 F0/19 变成根

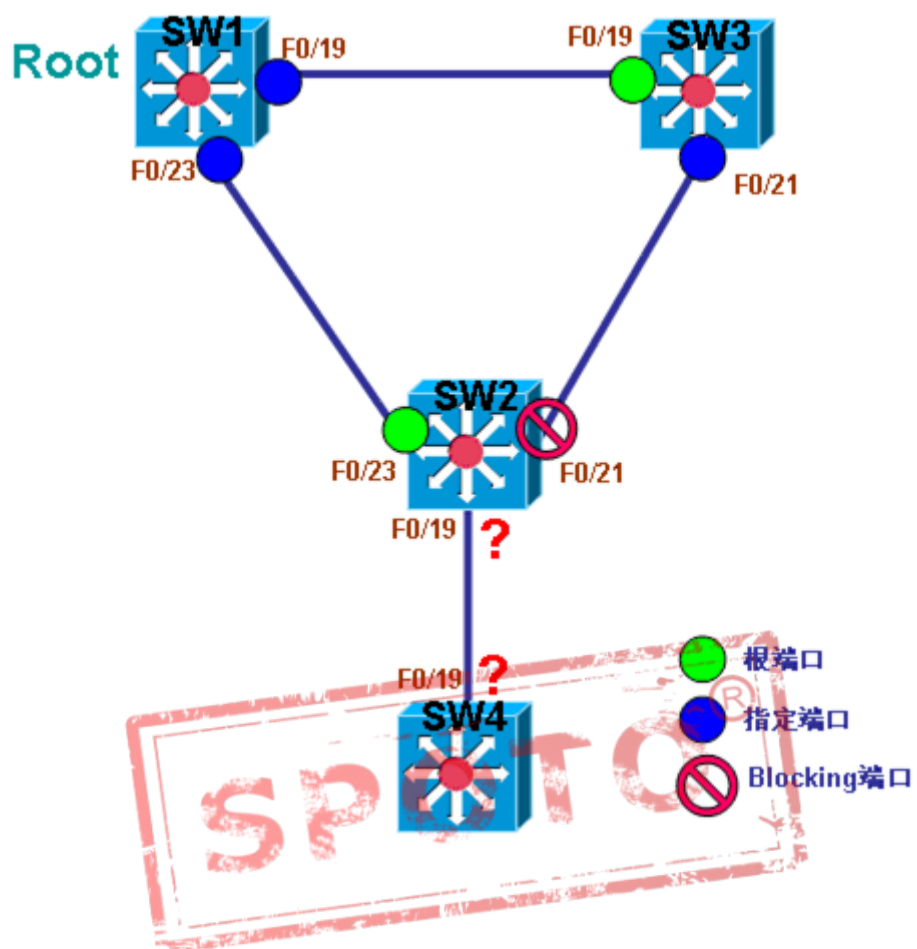
端口，.如果 SW2 将端口 F0/19 中断或者阻塞，都将禁止 SW4 对网络的影响。所以只要控制好连接新加入交换机的端口角色，就能够阻止对方成为根交换机。

特性 **Root Guard** 正是利用上述原因，控制 SW2 用来连接新加入交换机的那个端口的角色，可以决定是否让其影响当前网络。开启了 **Root Guard** 功能的端口，如果在此端口上连接的新交换机试图成为根交换机，那么此端口并不会成为根端口，相反，此端口将进入 **inconsistent (blocked)** 状态，从而防止新加入交换机抢占根角色来影响网络。

注：

- ★运行 MSTP 时，开启了 **Root guard** 的端口强制成为指定端口。
- ★开启 **Root guard** 的端口在哪个 **vlan**，**Root guard** 就对哪些 **vlan** 生效。
- ★不能在需要被 **UplinkFast** 使用的端口上开启 **Root Guard**。
- ★**Root Guard** 在可能连接新交换机的端口上开启。

配置



1. 开启 Root guard

(1) 在 SW2 上连接新交换机的端口 F0/19 上开启 Root guard

```
sw2(config-if)#spanning-tree guard root
```

2. 查看 Root guard

(1) 查看 SW2 上的 Root guard

sw2#sh spanning-tree detail

(输出被省略)

Port 19 (FastEthernet0/19) of VLAN0001 is forwarding

Port path cost 19, Port priority 128, Port Identifier 128.19.

Designated root has priority 16385, address 007d.618d.0300

Designated bridge has priority 32769, address 0013.805c.4b00

Designated port id is 128.19, designated path cost 19

Timers: message age 0, forward delay 0, hold 0

Number of transitions to forwarding state: 1

Link type is point-to-point by default

Root guard is enabled on the port

BPDU: sent 204, received 0

(输出被省略)

sw2#

说明：可以看到，F0/19 已经开启 Root guard

3.测试 Root guard

(1)配置 SW4 为根

sw4(config)#spanning-tree vlan 1 priority 4096

说明：给 SW4 配置一个更高优先级的 Bridge-ID，以此来抢夺根交换机的角色。

(2) 查看 SW2 的状态

当开启了 Root guard 的端口对方如果要成为根交换机，则会有如下提示，并且接口被放入 inconsistent (blocked) 状态：

```
sw2#
```

```
01:18:56: %SPANTREE-2-ROOTGUARD_BLOCK: Root guard blocking port
FastEthernet0/19 on VLAN0001.
```

```
sw2#
```

(3) 查看被放入 inconsistent (blocked) 状态的端口：

```
sw2#sh spanning-tree inconsistentports
```

Name	Interface	Inconsistency

VLAN0001	FastEthernet0/19	Root Inconsistent

Number of inconsistent ports (segments) in the system : 1

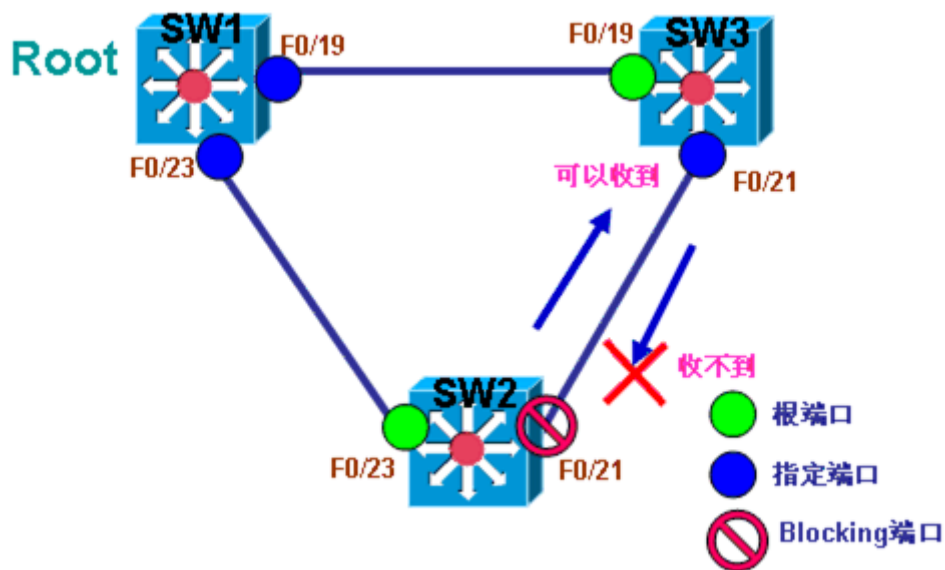
```
sw2#
```

说明：由于端口 F0/19 开启了 Root guard，而对端要成为根交换机，所以此端口被放入 inconsistent (blocked) 状态的端口。

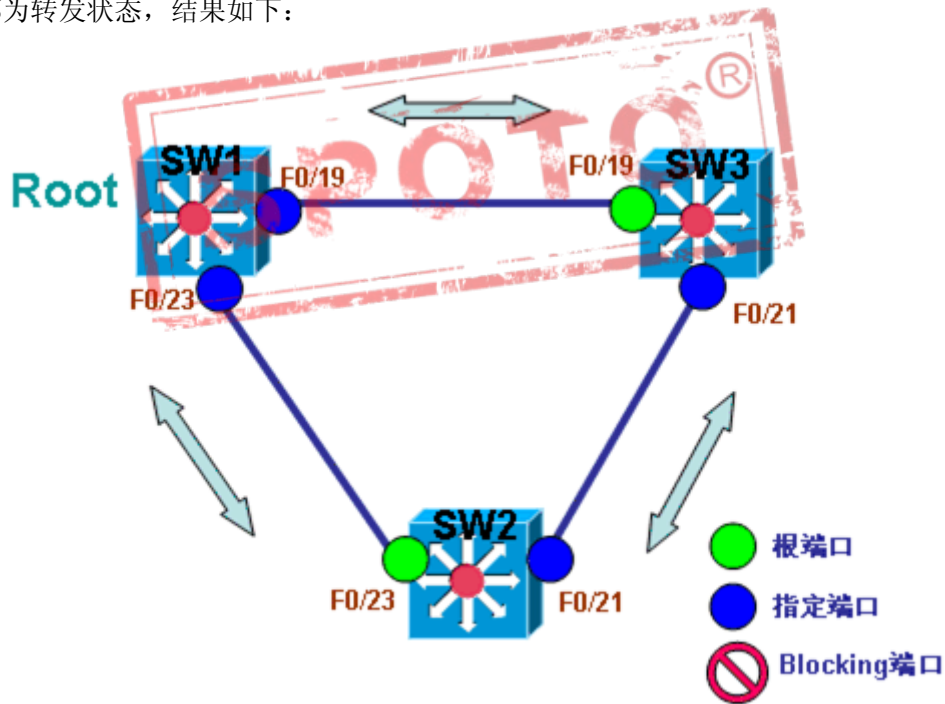
Loop Guard

在交换网络中，当两点之间存在多条冗余链路时，就会因为重复的数据包在网络中传递，引起广播风暴，并且还会造成交换机 MAC 地址表错误，使网络不稳定，因此造成环路。所以需要借助 STP 来阻塞网络中两点之间多余的链路，而只留一条活动链路，即为转发状态，其它多余链路变为 Blocking 状态，当转发状态的链路中断时，再启用 Blocking 状态的端口。

当 STP 运行时，只有两点之间存在多条冗余链路时，才会阻塞多余链路而只留一条活动链路。如果 STP 认为两点之间并没有多条链路，也就不会产生环路，那么就不会有端口被 Blocking。因为 STP 在判断两点之间是否有多条链路，是靠发送 BPDU，如果从多个端口收到同一台交换机的 BPDU，则认为与那个点之间有多条链路，所以会阻塞多余链路而只留一条。如果只从一个端口收到同台交换机的 BPDU，或者是没有收到重复 BPDU，则认为网络是无环的，也就没有端口被 Blocking，其它不需要被 Blocking 的端口，都会被变为指定端口。



在上图中的网络环境中，如果交换机所有端口收发数据的功能正常，则交换机就能够靠收发 BPDU 来检测出网络中的多余链路，就会将其 Block，从而避免环路。但是当网络中出现单向链路故障时，也就是某个端口只能收数据而不能发数据，或者只能发数据而不能收数据，这时网络会出现意想不到的麻烦。如上图，由于 SW2 的端口 F0/21 出现单向链路故障，导致从 F0/21 发出去的数据包能被 SW3 收到，而 SW3 发的数据包却不能被 SW2 收到，此时造成的结果是，SW3 认为网络是正常的，又由于 SW3 拥有更高优先级的 Bridge-ID，所以 SW3 上 F0/19 为根端口，F0/21 为指定端口，SW3 上所有端口都是转发状态，而没有 Blocking 的端口。但是由于 SW2 不能收到 SW3 发来的数据包，也就不能从 SW3 收到 BPDU，最终 SW2 只能从 F0/23 收到数据包，所以 SW2 认为网络是无环的，因此做出了一个错误的决定，就是在 STP 计算结束后，认为网络无环，而将原本应该被 Block 的端口 F0/21 变为指定端口，造成 SW 上 F0/23 和 F0/21 同时为转发状态。不难看出，此时，网络中所有的交换机端口都为转发状态，结果如下：



最终造成网络中所有的交换机端口都为转发状态，流量在所有端口上被转发，

引起广播风暴，出现环路。此结果是非常严重的。

单向链路故障不仅会使 **Blocking** 状态的端口错误地变成指定端口，还会造成根端口错误地变成指定端口。

对于上述问题，可以通过 **Loop Guard** 来解决，开启了 **Loop Guard** 的端口在收不到 **BPDUs** 的情况下，并不会认为网络是无环的，并不会错误地将端口变成指定端口，而是将收不到 **BPDUs** 的端口变成 **loop-inconsistent** 状态，此状态等同于 **blocking** 状态。

Loop Guard 可以全局开启，也可以在接口下开启，但不建议在全局开启，请在相应接口下开启。什么端口最需要开，很明显，当然是被 **blocking** 的端口，但并不完全正确，准确答案是在所有非指定端口开启，其实就是根端口和 **blocking** 端口。

当在接口开启后 **Loop Guard**，接口所在的所有 **VLAN** 都会生效，如果是接口是 **trunk**，哪个 **VLAN** 没有收到 **BPDUs**，接口就会在哪个 **VLAN** 被 **blocking**。在 **EtherChannel** 上是对整条生效。

只有交换机上的 **blocking** 端口和根端口才需要开启 **Loop Guard**。如果一个网络中所有交换机没有 **blocking** 的端口，就表示此网络无环，所以就不需要开 **Loop Guard**。并且根交换机上所有端口都是指定端口，所以在根上开 **Loop Guard** 是没有意义的。

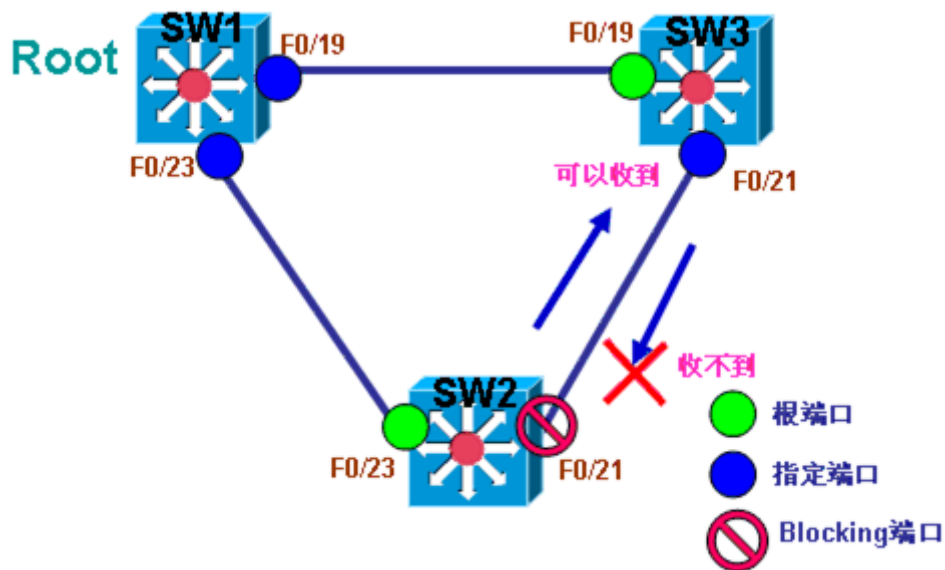
注：

★PortFast 的接口不能开启 **Loop Guard**。

★Root guard 和 **Loop Guard** 不能同时开。

★Root guard 支持 **PVST+**，**rapid PVST+**，**MSTP**。

配置



1. 开启 Loop Guard

(1) 在 SW2 的根端口与 Blocking 端口开启 Loop Guard

```
sw2(config)#int f0/21
```

```
sw2(config-if)#spanning-tree guard loop
```

```
sw2(config)#int f0/23
```

```
sw2(config-if)#spanning-tree guard loop
```

说明： 在所有根端口与 Blocking 端口开启 Loop Guard

2. 查看 Loop Guard

(1) 查看开启了 Loop Guard 的端口

```
sw2#sh spanning-tree detail
```

（输出被省略）

Port 21 (FastEthernet0/21) of VLAN0001 is blocking

Port path cost 19, Port priority 128, Port Identifier 128.21.

Designated root has priority 16385, address 007d.618d.0300

Designated bridge has priority 24577, address 0013.8065.bd80

Designated port id is 128.21, designated path cost 19

Timers: message age 3, forward delay 0, hold 0

Number of transitions to forwarding state: 0

Link type is point-to-point by default

Loop guard is enabled on the port

BPDU: sent 3, received 3917

（输出被省略）

```
sw2#
```

说明：可以看到，F0/21 已经开启 Loop Guard

3.测试 Loop Guard

(1)过滤掉 SW3 从 F0/21 发往 SW2 的 BPDU

```
sw3(config)#int f0/21
```

```
sw3(config-if)#spanning-tree bpdupfilter enable
```

说明：让 SW2 开启了 Loop Guard 的端口 F0/21 收不到 BPDU。

（2）查看 SW2 的 Loop Guard 状态

当 SW2 开启了 Loop Guard 的端口收不到 BPDU 时，会有如下提示：

```
sw2#
```

```
02:16:28: %SPANTREE-2-LOOPGUARD_BLOCK: Loop guard blocking port
FastEthernet0/21 on VLAN0001.
```

```
sw2#
```

（3）查看被放入 inconsistent (blocked) 状态的端口：

```
sw2#sh spanning-tree inconsistentports
```

Name	Interface	Inconsistency

VLAN0001	FastEthernet0/21	Loop Inconsistent

```
Number of inconsistent ports (segments) in the system : 1
```

```
sw2#
```

说明：由于端口 F0/21 开启了 Loop Guard，所有在收不到 BPDU 时，此端口被放入 inconsistent (blocked) 状态的端口。

EtherChannel

当在两台交换机之间连接多条线路来增加带宽时，由于 STP 的原因，最终会阻断其它多余的线路而只留下一条活动链路来转发数据，因此，在两台交换机之间连接多条线路，并不能起到增加带宽的作用。为了能够让两台交换机之间连接的多条线路同时提供数据转发以达到增加带宽的效果，可以通过 EtherChannel 来实现。

EtherChannel 将交换机上的多条线路捆绑成一个组，相当于逻辑链路，组中活动的物理链路同时提供数据转发，可以提高链路带宽。当组中有物理链路断掉后，那么流量将被转移到剩下的活动链路中去，只要组中还有活动链路，用户的流量就不会中断。

EtherChannel 只支持对 Fast Ethernet 接口或 Gigabit Ethernet 接口的捆绑，对于 10M 的接口还不支持。一个 EtherChannel 组中，最多只能有 8 个接口为用户转发数据。

在两台交换机之间连接多条链路时，如果只有一边交换机做了 EtherChannel 捆绑，而另一边不做捆绑，那么接口会工作在异常状态，而不能正常转发流量。所以，必须同时在两边交换机都做 EtherChannel 捆绑。

为了让两边交换机的接口都工作在 EtherChannel 组中，可以通过手工强制指定接口工作在组中，也可以通过协议自动协商。如果是手工强制指定，则不需要协议，自动协议的协议有以下两种：

Port Aggregation Protocol (PAgP)

Link Aggregation Control Protocol (LACP)

无论是手工指定，还是通过协议协商，交换机双方都必须采取相同的方式和协议，否则将导致接口异常。

EtherChannel 自动协商协议 PAgP 为思科专有，只有在双方交换机都为思科交换机时，才可以使用，而 LACP 为 IEEE 协议，任何交换机，只要支持 EtherChannel 的都可以使用该协议。

当将接口使用 PAgP 作为协商协议时，有以下两种模式可供选择：

Auto

只接收 PAgP 协商消息，并做出回应同意工作在 EtherChannel 下，并不主动发出 PAgP 协商，属于被动状态。

Desirable

主动发送 PAgP 协商消息，主动要求对方工作在 EtherChannel 下，属于主动模式。

如果两边交换机都是 Desirable 模式，则可以协商成功，如果两边都是 Auto 模式，则不能工作在 EtherChannel。

当将接口使用 LACP 作为协商协议时，有以下两种模式可供选择：

Passive

只接收 LACP 协商消息，并做出回应同意工作在 EtherChannel 下，并不主动发出 LACP 协商，属于被动状态。

Active

主动发送 LACP 协商消息，主动要求对方工作在 EtherChannel 下，属于主动模式。

如果两边交换机都是 Active 模式，则可以协商成功，如果两边都是 Passive 模式，则不能工作在 EtherChannel。

CCIE LAB认证经验分享千人群：539730342

在配置 EtherChannel 时，除了在接口上配置以上两种协议来自动协商外，还可以强制让接口工作在 EtherChannel 而不需要协商，配置为 ON 模式即可，如果配置 ON，则两边都必须配置为 ON，否则不能转发数据。

下表为配置 EtherChannel 的模式总结：

模式	协议	描述
ON	无	手工静态强制接口工作在 EtherChannel 下。
Auto	PAGP	只接收 PAGP 协商消息，并做出回应同意工作在 EtherChannel 下，并不主动发出 PAGP 协商。
Desirable	PAGP	主动发送 PAGP 协商消息，主动要求对方工作在 EtherChannel 下。
Passive	LACP	只接收 LACP 协商消息，并做出回应同意工作在 EtherChannel 下，并不主动发出 LACP 协商。
Active	LACP	主动发送 LACP 协商消息，主动要求对方工作在 EtherChannel 下。

当配置 PAGP 时，可以使用关键字 non-silent，如果不指定 non-silent，默认为 silent。

Silent 表示即使不能从对端设备收到 PAGP 协商数据，也使物理接口工作在 EtherChannel 组中，思科建议接口连接服务器或分析仪时使用。
non-silent 表示只有在和对方协商成功之后，才使物理接口工作在 EtherChannel 组中。也就是说只有双方都支持 PAGP 的情况下，才使物理接口工作在 EtherChannel 组中。

CCIE LAB认证经验分享千人群：539730342

因为三层交换机的接口既可以工作在二层模式，也可以工作在三层模式，所以 EtherChannel 捆绑后的逻辑接口也有二层和三层之分。

当将接口 EtherChannel 捆绑后，会自动生成逻辑接口，称为 port-channel 接口，port-channel 接口与 EtherChannel 组的号码相同，但范围是 1-48。当使用二层接口时，在物理接口下配置参数后，port-channel 接口将读取物理接口下的参数，但必须组成的所有接口都做相同的配置；在 port-channel 接口下做的配置也会自动在物理接口下生效。当使用三层接口时，必须先将物理接口变成三层接口后，再做捆绑，因为 port-channel 接口是不能在二层与三层之间转换的，配置三层接口，应该到 port-channel 接口下做的配置，而不应该直接配置物理接口。

如果是使用 2 层 EtherChannel，那么组中第一个正常工作的口接口的 MAC 地址就是 port-channel 接口接口的 MAC 地址。

注：

★在配置 EtherChannel 组时，需要定义组号码，但不要配置超过 48 个组。

★两边交换机的 EtherChannel 组号码可以采用不同号码。

★PAGP 组中不能配超过 8 个接口。

★LACP 中不能超过 16 个接口，但只有 8 个活动接口。

★两个协议可以配置在同台交换机上，但不能配置在同一个组中。

★组中的接口不能是 SPAN 的目标接口和安全接口以及 802.1x 端口。

★将接口配置为 2 层时，全部必须在相同 VLAN，如果是 trunk，native vlan 必须相同。

★配好 EtherChannel 组后后，在 port-channel 下配的参数会对所有物理接口生效，但对单个物理接口配置的只对单物理接口生效。

★多个接口捆绑成单条 EtherChannel 后，在 STP 中，被当作单条链路来计算，同时 Path Cost 值会和原物理链路有所不同。

EtherChannel Load Balancing

CCIE LAB认证经验分享千人群：539730342

当将多个接口捆绑成 EtherChannel 组之后，流量将同时从多个接口上被发出去，称为 Load Balancing, 即负载均衡，对于流量以什么样的负载均衡方式从 EtherChannel 组中的多个接口上发出去，可以有以下几种方式：

Source-MAC

基于源 MAC，默认为此模式，不同源主机，流量可能从不同的接口被发出去，但相同源主机肯定走相同接口。

Source-and-Destination MAC

同时基于源和目标 MAC，流量从主机 A 到主机 B，从主机 A 到主机 C 以及从主机 C 到主机 B 都可能走不同的接口。

Source-IP

基于源 IP，不能源 IP 的流量可能走不同接口，相同 IP 则走相同接口。

Destination-IP

基于目的 IP，到不同目标 IP 的流量，会走不同接口，不同主机发往相同 IP 的流量会走相同接口。

Source-and-Destination IP

同时基于源和目标 IP，流量从主机 A 到主机 B，从主机 A 到主机 C 以及从主机 C 到主机 B 都可能走不同的接口。

注：并不是所有型号的交换机所有 IOS 都支持所有负载方式，需要视 IOS 版本而定。

在交换机之间通过 EtherChannel 捆绑了多条链路后，默认执行基于源 MAC 的负载均衡，而每条链路的流量比例却是固定的，也就是说，你只能改变 EtherChannel 负载均衡方式，但却改不了每条物理链路上的流量比例，接口上的流量比例，执行以下标准：

Number of Ports in the EtherChannel	Load Balancing
8	1:1:1:1:1:1:1:1
7	2:1:1:1:1:1:1
6	2:2:1:1:1:1
5	2:2:2:1:1
4	2:2:2:2
3	3:3:2
2	4:4

配置



1. 配置 2 层 EtherChannel

(1) 配置 SW1

```
sw1(config)#int range f0/23 - 24
```

```
sw1(config-if-range)#channel-group 12 mode desirable
```

说明：在接口 F0/23-24 下选用 PAGP 配置 EtherChannel

(2) 配置 SW2

```
sw2(config)#int range f0/23-24
```

```
sw2(config-if-range)#channel-group 12 mode desirable
```

说明：在接口 F0/23-24 下选用 PAGP 配置 EtherChannel

(3) 查看 EtherChannel

```
sw1#show etherchannel summary
```

Flags: D - down P - in port-channel

I - stand-alone s - suspended

H - Hot-standby (LACP only)

R - Layer3 S - Layer2

U - in use f - failed to allocate aggregator

u - unsuitable for bundling

w - waiting to be aggregated

d - default port

Number of channel-groups in use: 1

Number of aggregators: 1

CCIE LAB认证经验分享千人群：539730342

Group	Port-channel	Protocol	Ports	
-----+-----+-----+-----				
12	Po12(SU)	PAgP	Fa0/23 (P)	Fa0/24 (P)

sw1#

说明：可以看到，已捆绑的接口为 2 层接口，并且所有物理接口都工作在 EtherChannel 下。

（4）在 port-channel 接口下配置接口

sw1(config)#int port-channel 12

sw1(config-if)#switchport mode access

sw1(config-if)#switchport access vlan 10

说明：port-channel 接口下将接口划入 VLAN。

（5）查看 port-channel 接口 MAC 地址

F0/23:

sw1#sh int f0/23

FastEthernet0/23 is up, line protocol is up (connected)

Hardware is Fast Ethernet, address is 007d.618d.0317 (bia 007d.618d.0317)

（输出被省略）

sw1#

F0/24

```
sw1#sh int f0/24
```

```
FastEthernet0/24 is up, line protocol is up (connected)
```

```
Hardware is Fast Ethernet, address is 007d.618d.0318 (bia  
007d.618d.0318)
```

（输出被省略）

```
sw1#
```

```
port-channel:
```

```
sw1#sh int port-channel 12
```

```
Port-channel12 is up, line protocol is up (connected)
```

```
Hardware is EtherChannel, address is 007d.618d.0318 (bia  
007d.618d.0318)
```

（输出被省略）

```
sw1#
```

说明：port-channel 使用了接口 F0/24 下的 MAC 地址，说明接口 F0/24 先工作正常。

2. 配置 3 层 EtherChannel

(1) 配置 SW1

```
sw1(config)#int range f0/23 - 24
```

CCIE LAB认证经验分享千人群：539730342

```
sw1(config-if-range)#no switchport
```

```
sw1(config-if-range)#channel-group 12 mode active
```

```
sw1(config)#int port-channel 12
```

```
sw1(config-if)#ip address 10.1.1.1 255.255.255.0
```

说明：配置 3 层 EtherChannel，需要先将物理接口变成 3 层接口后，才能正常配置，IP 地址必须在 port-channel 下配置。

(2) 配置 SW2

```
sw2(config)#int range f0/23 - 24
```

```
sw2(config-if-range)#no switchport
```

```
sw2(config-if-range)#channel-group 12 mo active
```

```
sw2(config)#int port-channel 12
```

```
sw2(config-if)#ip address 10.1.1.2 255.255.255.0
```

说明：配置 3 层 EtherChannel，需要先将物理接口变成 3 层接口后，才能正常配置，IP 地址必须在 port-channel 下配置。

(3) 查看 EtherChannel

```
sw1#sh eth summary
```

Flags: D - down P - in port-channel

CCIE LAB认证经验分享千人群：539730342

I - stand-alone s - suspended

H - Hot-standby (LACP only)

R - Layer3 S - Layer2

U - in use f - failed to allocate aggregator

u - unsuitable for bundling

w - waiting to be aggregated

d - default port

Number of channel-groups in use: 1

Number of aggregators: 1

Group Port-channel Protocol Ports

-----+-----+-----+-----

12 Po12(RU) LACP Fa0/23(P) Fa0/24(P)

sw1#

说明：可以看到，已捆绑的接口为 3 层接口，并且所有物理接口都工作在 EtherChannel 下。

(4) 测试 port-channel 连通性

```
sw1#ping 10.1.1.2
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

```
sw1#
```

说明：port-channel 正常工作在 3 层。

3. 配置 Load Balancing

(1) 配置基于目标 MAC 的负载均衡

```
sw1(config)#port-channel load-balance dst-mac
```

说明：开启了基于目标 MAC 的负载均衡，默认为基于源 MAC，其它负载方式，可自行配置。

(2) 查看 EtherChannel Load Balancing

```
sw1#sh etherchannel load-balance
```

EtherChannel Load-Balancing Configuration:

dst-mac

EtherChannel Load-Balancing Addresses Used Per-Protocol:

Non-IP: Destination MAC address

IPv4: Destination MAC address

sw1#

说明：可以看到，EtherChannel 已经基于 MAC 的负载均衡。

附：当配置 PAGP 时，可以选择配置 non-silent，默认为 silent，配置如下：

```
sw1(config)#int range f0/23 - 24
```

```
sw1(config-if-range)#channel-group 12 mode desirable non-silent
```

Protected Port

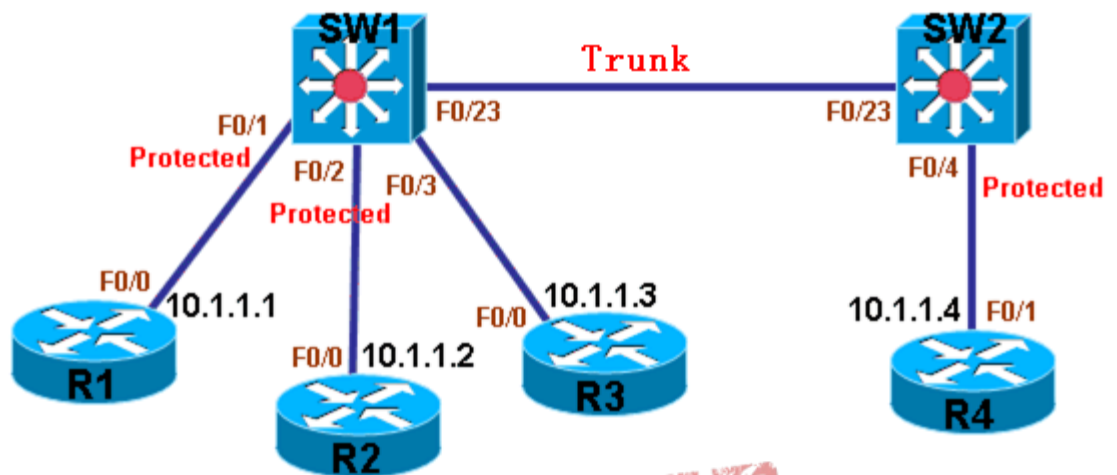


在某些特殊需求下，需要禁止同台交换机上相同 VLAN 的主机之间通信，但又不能将这些不能通信的主机划到不同 VLAN，因为还需要和 VLAN 中的其它主机通信，只是不能和部分主机通信。要限制交换机上相同 VLAN 的主机通信，通过将交换机上的接口配置成 Protected Port 来实现，如果交换机上某个 VLAN 有三个接口，其中有两个是 Protected Port，有一个是正常端口，那么两个 Protected Port 之间是不能通信的，但是 Protected Port 与正常端口之间的流量还是保持正常，而不受任何限制。

Protected Port 可以拒绝 unicast，broadcast 以及 multicast 在这些端口之间通信，也就是说 Protected Port 与 Protected Port 之间没有任何流量发送。Protected Port 只在单台交换机上有效，也就是说只有单台交换机上的 Protected Port 与 Protected Port 之间是不能通信的，但是不同交换机的 Protected Port 与 Protected Port 之间通信还是保持正常。

配置 Protected Port 时，可以在物理接口和 EtherChannel 上配置，如果是配在 EtherChannel 上，那么配置将对 EtherChannel 中的所有物理接口生效。

配置



说明：以上图为例，配置 protected port，SW1 的 F0/1，F0/2，F0/3 以及 SW2 的 F0/4 都在 VLAN 10 中。

1. 配置交换机

(1) 配置 SW1

```
sw1(config)#vlan 10
```

```
sw1(config-vlan)#exit
```

```
sw1(config)#int range f0/1 - 3
```

```
sw1(config-if-range)#switchport mode access
```

```
sw1(config-if-range)#switchport access vlan 10
```

```
sw1(config)#int f0/23
```

CCIE LAB认证经验分享千人群：539730342

```
sw1(config-if)#switchport trunk encapsulation dot1q
```

```
sw1(config-if)#switchport mode trunk
```

(2) 配置 SW2

```
sw2(config)#vlan 10
```

```
sw2(config-vlan)#exit
```

```
sw2(config)#int f0/4
```

```
sw2(config-if)#switchport mode access
```

```
sw2(config-if)#switchport access vlan 10
```

```
sw2(config)#int f0/23
```

```
sw2(config-if)#switchport trunk encapsulation dot1q
```

```
sw2(config-if)#switchport mode trunk
```

2. 配置路由器

(1) 配置 R1

```
r1(config)#int f0/0
```

```
r1(config-if)#ip add 10.1.1.1 255.255.255.0
```

(2) 配置 R2

```
r2(config)#int f0/0
```

```
r2(config-if)#ip add 10.1.1.2 255.255.255.0
```

(3) 配置 R3


```
r3(config)#int f0/0
```

```
r3(config-if)#ip add 10.1.1.3 255.255.255.0
```

(4) 配置 R4

```
r4(config)#int f0/1
```

```
r4(config-if)#ip add 10.1.1.4 255.255.255.0
```

3. 测试正常情况下的通信

(1) 测试 R1 到 R2 的连通性

```
r1#ping 10.1.1.2
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

```
r1#
```

说明：因为没有配置 protected port，所以 R1 到 R2 通信正常。

(2) 测试 R1 到 R3 的连通性

```
r1#ping 10.1.1.3
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:
```

```
!!!!
```

CCIE LAB认证经验分享千人群：539730342

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r1#

说明：因为没有配置 protected port，所以 R1 到 R3 通信正常。

(3) 测试 R1 到 R4 的连通性

r1#ping 10.1.1.4

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.4, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r1#

说明：因为没有配置 protected port，所以 R1 到 R4 通信正常。

3. 配置 protected port

(1) 在 SW1 上将 F0/1 和 F0/2 配置为 protected port

sw1(config)#int f0/1

sw1(config-if)#switchport protected

sw1(config)#int f0/2

sw1(config-if)#switchport protected

(2) 在 SW2 上将 F0/4 配置为 protected port

```
sw2(config)#int f0/4
```

```
sw2(config-if)#switchport protected
```

4. 测试配置了 protected port 的网络通信

(1) 测试 R1 到同台交换机的正常端口 F0/3 的连通性

```
r1#ping 10.1.1.3
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

```
r1#
```

说明：因为 protected port 与正常端口之间的通信不受影响，所以 R1 到 R3 通信正常。

(2) 测试 R1 到同台交换机的 protected port F0/2 的连通性

```
r1#ping 10.1.1.2
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
```

```
.....
```

```
Success rate is 0 percent (0/5)
```

```
r1#
```

说明：因为同台交换机上 protected port 与 protected port 之间的流量被拒绝，所以 R1 到 R2 通信失败。

(3) 测试 R1 到远程交换机 SW2 的 protected port F0/4 的连通性

```
r1#ping 10.1.1.4
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.1.4, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

```
r1#
```

说明：因为只有单台交换机上的 Protected Port 与 Protected Port 之间是不能通信的，但是不同交换机的 Protected Port 与 Protected Port 之间通信还是保持正常，所以 R1 到 R4 的通信正常。

(4) 在交换机上查看 Protected Port

```
sw1#sh int f0/1 switchport
```

(输出被省略)

```
Protected: true
```

```
Unknown unicast blocked: disabled
```

```
Unknown multicast blocked: disabled
```

```
Appliance trust: none
```

```
sw1#
```

说明：可以看到交换机上接口的 Protected Port 功能已经开启。

Port Blocking

默认情况下，交换机收到未知目标 MAC 的流量，也就是目标 MAC 地址不在 MAC 地址表中的流量，会将此流量在所有接口上泛洪。用户可以选择在交换机接口上拒绝泛洪未知目标 MAC 的流量，配置可以对 unicast 和 multicast 生效，但不能限制广播流量。

接口上默认是没有 Port Blocking 配置的。

配置 Port Blocking 时，可以在物理接口和 EtherChannel 上配置，如果是配在 EtherChannel 上，那么配置将对 EtherChannel 中的所有物理接口生效。

配置

1. 在接口上配置 Port Blocking

(1) 配置 Port Blocking 限制 unicast

```
sw1(config)#int f0/1
sw1(config-if)#switchport block unicast
```

(2) 配置 Port Blocking 限制 multicast

```
sw1(config)#int f0/1

sw1(config-if)#switchport block multicast
```

(3) 查看 Port Blocking

```
sw1#sh interfaces f0/1 switchport
```

（输出被省略）

```
Unknown unicast blocked: enabled
```

```
Unknown multicast blocked: enabled
```

Appliance trust: none

sw1#

说明：可以看到，交换机接口上已经开启拒绝泛洪未知目标 MAC 的单播流量和组播流量，并且两个可以同时开启。

Port Security

交换机在转发数据包时，需要根据数据包的目标 MAC 地址来决定出口，因此，交换机会将 MAC 地址与相对应的接口记录在一张表中，以供转发数据包使用，这张表就是 MAC 地址表。在正常情况下，MAC 地址表允许一个接口可以与多个 MAC 地址相对应，只要接口上有相应的 MAC 地址，那么数据包就可以从这个接口发出去。一个接口上对应着什么样的 MAC 地址，一个接口允许多多少个 MAC 地址与之相对应，这都影响到交换机对数据的转发。为了让用户对交换机的 MAC 地址表有更高的控制权限，交换机接口上的 Port Security 功能提供更多的安全保护。

Port Security 可以控制交换机上特定的接口与特定的 MAC 地址的对应关系，也可以限制接口上最大的 MAC 地址数量。

具有 Port Security 功能的接口，被称为 secure port，secure port 接口上通过控制数据包的源 MAC 地址来控制流量，绝不会转发预先定义好的 MAC 地址之外的流量。准确地说，是 secure port 只转发合法的流量，对于违规的流量，是不放行的。区别是否违则，有以下两种情况：

1. 当接口上 MAC 地址数量达到最大允许数量后，还有更多的 MAC 要访问，就算违规。
2. 一个 secure port 接口上的合法 MAC 在另外一个 secure port 接口上访问，也算违规。

被 Port Security 允许的 MAC 地址，就是合法的 MAC 地址，称为安全 MAC 地址（Secure MAC Addresses），secure port 接口只放行源 MAC 为安全 MAC 地址的数据

包。

要在 secure port 接口上定义安全 MAC 地址，有以下几种方法：

静态手工配置

手工添加 MAC 地址与接口的对应关系，会保存在地址表和 running configuration 中。

动态学习

将接口上动态学习到的 MAC 地址作为安全 MAC 地址，但此 MAC 地址只保存在 MAC 地址表中，交换机重启后将丢失。

Sticky secure MAC addresses

为了结合静态手工配置与动态学习 MAC 地址的优势，Sticky 将动态学习到的 MAC 地址作为安全 MAC 地址，并且将结果保存到 running configuration 中。

一个 secure port 接口上可以允许的 MAC 地址数量是系统可支持的最大 MAC 地址数量。对于违规的流量，可以采取以下四个可执行的动作：

Protect

只丢弃不允许 MAC 地址的流量，其它合法流量正常，但不会通知有流量违规了。

Restric

只丢弃不允许 MAC 地址的流量，其它合法流量正常，但会有通知，发送 SNMP trap，并会记录 syslog。

Shutdown

（默认模式） 将接口变成 error-disabled 并 shut down，并且接口 LED 灯会关闭，也会发 SNMP trap，并会记录 syslog。

shutdown vlan

相应 VLAN 变成 error-disabled，但接口不会关，也会发 SNMP trap，并会记录 syslog。

注：



★当一个 secure port 接口上的 MAC 地址在另外一个 secure port 接口出现后，就算违规，而违规动作是对出现重复地址的接口实施的，是为了防止攻击。

★当接口被 error-disabled 后，要恢复，请在接口上使用命令：shutdown 后 no shutdown。

以下是来自思科官方的模式与结果对应表：

Security Violation Mode Actions						
Violation Mode	Traffic is forwarded ¹	Sends SNMP trap	Sends syslog message	Displays error message ²	Violation counter increments	Shuts down port
protect	No	No	No	No	No	No

restrict	No	Yes	Yes	No	Yes	No
shutdown	No	Yes	Yes	No	Yes	Yes
shutdown vlan	No	Yes	Yes	No	Yes	No ³

注：

★默认接口上 Port Security 是关闭的，Port Security 默认只允许 1 个安全 MAC 地址。

★只能在静态 access 接口和静态 trunk 接口上配 Port Security，不能在 dynamic 接口上配。

★Port Security 接口不能是 SPAN 的目标接口，不能在 EtherChannel 中。

Port Security Aging Time （Port Security MAC 地址老化时间）

在正常接口下动态学习到的 MAC 地址，在老化时间到了之后，交换机会将它从 MAC 地址表中删除。

而对于 Port Security 接口下的 MAC 地址，如果是通过安全命令静态手工添加的，则不受 MAC 地址老化时间的限制，也就是说通过安全命令静态手工添加的 MAC 在 MAC 地址表中永远不会消失。而即使在 Port Security 接口下动态学习到的 MAC 地址，也永远不会消失。

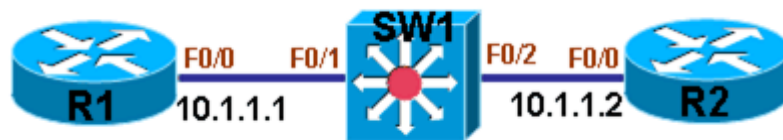
基于上述原因，有时限制了 Port Security 接口下的最大 MAC 地址数量后，当相应的地址没有活动了，为了腾出空间给其它需要通信的主机使用，则需要让 Port Security 接口下的 MAC 地址具有老化时间，也就是说需要交换机自动将安全 MAC 地址删除。

对于在 Port Security 接口下设置 MAC 地址的老化时间，分两种类型：absolute 和 inactivity，其中 absolute 表示绝对时间，即无论该 MAC 地址是否在通信，超过老化时间后，立即从表中删除；inactivity 为非活动时间，即该 MAC 地址在没有流量的

情况下，超过一定时间后，才会从表中删除。

配置 MAC 地址老化时间的单位是分钟，范围是 0-1440，对于 sticky 得到的 MAC 地址，不受老化时间限制，并且不能更改。

配置



1.查看当前路由器的 MAC 地址

(1) 查看 R1 的接口 F0/0 的 MAC 地址

```
r1#sh int f0/0
```

```
FastEthernet0/0 is up, line protocol is up
```

```
Hardware is AmdFE, address is 0013.1a85.d160 (bia 0013.1a85.d160)
```

```
Internet address is 10.1.1.1/24
```

(输出被省略)

```
r1#
```

说明： R1 的接口 F0/0 的 MAC 地址为 0013.1a85.d160

（2）查看 R2 的接口 F0/0 的 MAC 地址

```
r2#sh int f0/0
```

```
FastEthernet0/0 is up, line protocol is up
```

```
Hardware is AmdFE, address is 0013.1a2f.1200 (bia 0013.1a2f.1200)
```

（输出被省略）

R2:

说明：R2 的接口 F0/0 的 MAC 地址为 0013.1a2f.1200

2.配置交换机的 port-security

（1）配置 F0/1

```
sw1(config)#int f0/1
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#switchport port-security
```

```
sw1(config-if)#switchport port-security maximum 1
```

```
sw1(config-if)#switchport port-security mac-address 0013.1a85.d160
```

```
sw1(config-if)#switchport port-security violation shutdown
```

说明：将接口静态配置成 access 后，再开启 port-security，允许最大地址数量为 1，默认也是为 1，定义的最大地址数量值不能比已学到的 MAC 地址少，否则无效。手工静态指定的安全 MAC 地址为 0013.1a85.d160，在违规后采取动作 shutdown。

（2）查看 F0/1 的配置

```
sw1#sh run int f0/1
```

Building configuration...

Current configuration : 136 bytes

!

interface FastEthernet0/1

switchport mode access

switchport port-security

switchport port-security mac-address 0013.1a85.d160

end

sw1#

说明：因为默认允许的最大地址数量为 1，所有不显示出来。

(3) 配置 F0/2

sw1(config)#int f0/2

sw1(config-if)#switchport mode access

sw1(config-if)#switchport port-security

sw1(config-if)#switchport port-security maximum 2

sw1(config-if)#switchport port-security mac-address sticky

sw1(config-if)#switchport port-security violation shutdown

说明：将接口静态配置成 access 后，再开启 port-security，允许最大地址数量为 2，指定安全 MAC 地址的方式为 sticky，在违规后采取动作 shutdown。

（4）查看 F0/2 的配置

```
sw1#sh run int f0/2
```

```
Building configuration...
```

```
Current configuration : 224 bytes
```

```
!
```

```
interface FastEthernet0/2
```

```
switchport mode access
```

```
switchport port-security maximum 2
```

```
switchport port-security
```

```
switchport port-security mac-address sticky
```

```
switchport port-security mac-address sticky 0013.1a2f.1200
```

```
end
```

```
sw1#
```

说明：因为指定安全 MAC 地址的方式为 sticky，所以此接口连接的 R2 上的 MAC 地址 0013.1a2f.1200 已经被载入配置中。

3.测试 port-security

（1）测试 R1 以合法 MAC 地址访问 R2

```
r1#ping 10.1.1.2
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r1#

说明：因为 R1 以源 MAC0013.1a85.d160 访问 R2，交换机的接口 F0/1 认为 0013.1a85.d160 是安全 MAC，所以 R1 访问 R2 成功。

（2）测试交换机的 F0/1 上的 port-security 违规

r1(config)#int f0/0

r1(config-if)#standby 1 ip 10.1.1.10

说明：因为交换机的 F0/1 允许的最大 MAC 地址数量为 1，而 R1 已经有了一个 MAC 地址，在接口上配置 HSRP 之后，还会产生一个虚拟 MAC 地址，所以这个 HSRP 虚拟 MAC 地址就是第 2 个 MAC 地址，而第 2 个 MAC 地址在交换机的 F0/1 上出现就算是违规。

（3）查看交换机上 port-security 违规后的现象

sw1#

01:39:48: %PM-4-ERR_DISABLE: psecure-violation error detected on Fa0/1, putting Fa0/1 in

CCIE LAB认证经验分享千人群：539730342

err-disable state

01:39:48: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred, caused by MAC

address 0000.0c07.ac01 on port FastEthernet0/1.

01:39:49: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to

down

01:39:50: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to down

sw1#

sw1#sh int f0/1

FastEthernet0/1 is down, line protocol is down (err-disabled)

（输出被省略）

sw1#

说明：当交换机的 port-security 违规后,会出现以上 log 提示，并且查看交换机的接口为 err-disabled 状态，并且被 shutdown。

（4）测试交换机上 port-security 另一种违规

r2(config)#int f0/0

r2(config-if)#mac-address 0013.1a85.d160

CCIE LAB认证经验分享千人群：539730342

说明：将 R1 的 MAC 地址 0013.1a85.d160 添加到 R2 的接口上，因为一个 secure port 接口上的合法 MAC 在另外一个 secure port 接口上访问，也算违规。

（5）查看交换机上 port-security 违规后的现象

sw1#

01:46:27: %PM-4-ERR_DISABLE: psecure-violation error detected on Fa0/2, putting Fa0/2 in

err-disable state

01:46:27: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred, caused by MAC

address 0013.1a85.d160 on port FastEthernet0/2.

01:46:28: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to

down

01:46:29: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed state to down

sw1#

sw1#sh int f0/1

FastEthernet0/1 is up, line protocol is up (connected)

（输出被省略）


```
sw1#
```

```
sw1#sh int f0/2
```

```
FastEthernet0/2 is down, line protocol is down (err-disabled)
```

（输出被省略）

```
sw1#
```

说明：可以看见，当一个 secure port 接口上的 MAC 地址在另外一个 secure port 接口出现后，就算违规，而违规动作是对出现重复地址的接口实施的，是为了防止攻击。

4.配置 Port Security MAC 地址老化时间

（1）配置 Port Security MAC 地址老化时间

```
sw1(config)#int f0/1
```

```
sw1(config-if)#switchport port-security aging time 1
```

```
sw1(config-if)#switchport port-security aging type inactivity
```

说明：配置 Port Security MAC 地址老化时间为 1 分钟，并且相应 MAC 在 1 分钟没有流量的情况下被删除。但此配置只对接口下动态学习到的 MAC 地址生效。

（2）配置手工静态指定的 MAC 地址的老化时间

```
sw1(config)#int f0/1
```

```
sw1(config-if)#switchport port-security aging static
```

说明：配置手工静态指定的 MAC 地址在 1 分钟没有流量的情况下被删除。

IP Source Guard

默认情况下，交换机在二层接口上转发数据时，只查看数据包的 MAC 地址，并不会查看数据包的 IP 地址，如果要让交换机根据数据包的 IP 或者同时根据 IP 与 MAC 做出转发决定，有多种方法可以实现。如根据 IP 转发，可以通过在接口上应用 Port ACL 来实现，如果要根据 MAC 转发，可以通过应用 port-security 来实现，但无论是 Port ACL 还是 port-security，都存在一些局限性，下面介绍一种扩展性较高的安全防护特性- IP Source Guard，IP Source Guard 可以根据数据包的 IP 地址或 IP 与 MAC 地址做出转发决定，如果数据包的 IP 或 MAC 是不被允许的，那么数据包将做丢弃处理。

因为 IP Source Guard 需要根据数据包的 IP 或者同时根据 IP 与 MAC 做出转发决定，所以 IP Source Guard 在工作时，需要有一张 IP 和 MAC 的转发表，在这张表中，明确记录着哪些 IP 是可以转发的，哪些 MAC 可以被转发，其它不能被转发的统统丢弃。这表转发表称为 IP source binding table，并且只能被 IP source guard 使用。而 IP source binding table 表，只有在交换机上开启 DHCP snooping 功能后，才会生成。

IP Source Guard 的这张转发表的条目可以自动学习，也可以手工静态添加，如果是自动学习，是靠 DHCP snooping 功能学习的，所以只有客户端是通过 DHCP 请求获得地址，并且 DHCP 服务器的回复是经过交换机时，才能被 DHCP snooping 学习到。

当在交换机上同时开启了 IP Source Guard 与 DHCP snooping 后，交换机将在开启了的接口上拒绝所有流量通过，只放行 DHCP 流量，并且会自动应用一条 ACL 到接口上，也只有 ACL 允许的 IP 才能通过，这条 ACL 无法在正常配置中查看，只能通过表的方式查看。因为默认可以允许 DHCP 通过，所以主机的 DHCP 请求可以帮助他们从 DHCP 服务器获得地址，当 DHCP 服务器向主机提供地址时，这个信息在穿越交换机时，IP 信息会被记录，并且被自动生成的 ACL 允许转发，这样以后，只有主机从 DHCP 服务器自动获得的 IP 可以通过交换机，而其它 IP 的流量都是被拒绝的，因此可以看出，IP Source Guard 和 DHCP snooping 的配合使用，可以防止一个主机使

CCIE LAB认证经验分享千人群：539730342

用其它主机的 IP 地址来攻击网络，因为只有 DHCP 获得的地址能够被交换机转发，其它接口即使配置了相同 IP，都会被 IP Source Guard 拒绝放行。

注： IP Source Guard 自动生成的 ACL 优先于任何 Router ACLs 和 VLAN map。

IP source binding table 的条目除了靠 DHCP snooping 自动学习之外，还可以手工静态配置。

当 IP source binding table 表的内容有任何变化时，自动生成的 ACL 也会自动改变，自动生成的 ACL 总是与 IP source binding table 表中的条目同步。如果 IP source binding table 表是空的，那么 ACL 将拒绝所有流量通过，也就是最初的状态。

因为 IP Source Guard 可以根据数据包的 IP 地址做出转发，也可以同时根据 IP 与 MAC 地址做出转发，如果要开启同时根据 IP 与 MAC 做出转发，还必须在接口上开启 port security 功能，port security 帮助识别流量的 MAC 地址，并将其添加到 source binding table 表中，最后被 ACL 设置为允许。

注：

★ IP source guard 只能在二层接口上开启，并且不支持 EtherChannel。

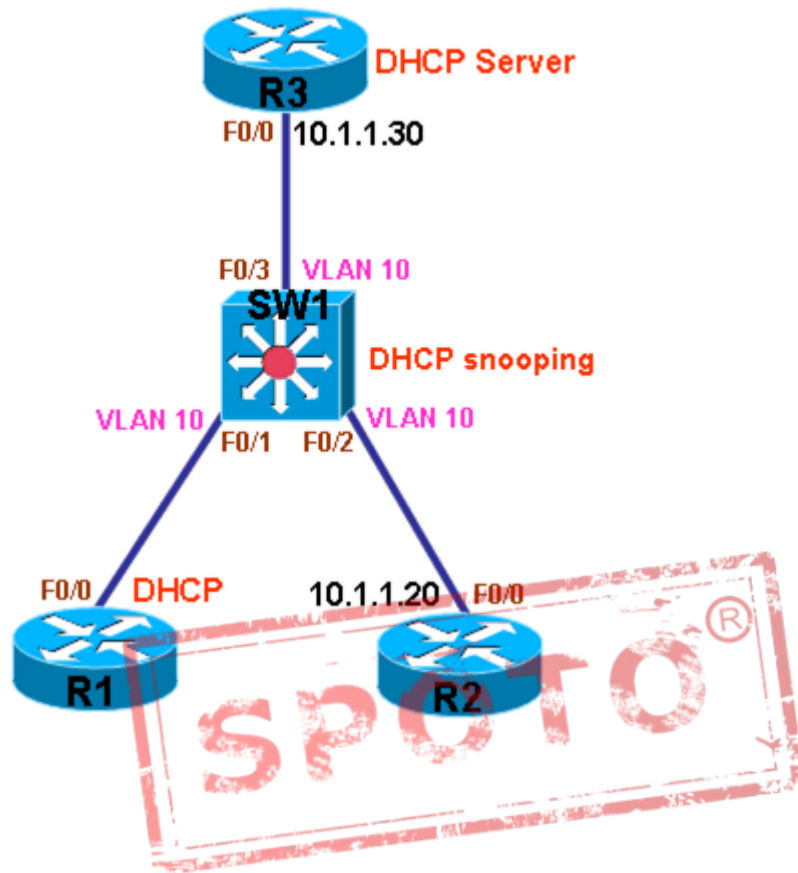
★ 当 IP source guard 根据 IP 转发在接口上开启，这个接口所在的 VLAN 必须开启 DHCP snooping，

★ 如果是在 trunk 上开启，DHCP snooping 应该在所有 VLAN 上开启，过滤也对所有 VLAN 生效。

★ 如果在 trunk 上打开或关闭某一个 VLAN DHCP snooping 的，可能会出问题。

★ 当 IP source guard 根据 IP 和 MAC 转发在接口上开启，那么 DHCP snooping 和 port security 必须同时开启。

配置



1.网络初始

(1) 配置交换机

```
sw1(config)#int range f0/1 - 3
```

```
sw1(config-if-range)#switchport mode access
```

```
sw1(config-if-range)#switchport access vlan 10
```

说明： 已经将 R1、R2、R3 划到同一 VLAN 10。

（2）测试 R2 到 R3 的连通性

```
r2#ping 10.1.1.30
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.30, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

```
r2#
```

说明：因为 R2 与 R3 在同一 VLAN 10，所以通信正常。

2. 在 SW1 上配置 IP Source Guard

（1）在交换机上开启 DHCP snooping

```
sw1(config)#ip dhcp snooping
```

```
sw1(config)#ip dhcp snooping vlan 10
```

```
sw1(config)#int f0/3
```

```
sw1(config-if)#ip dhcp snooping trust
```

说明：开启 IP Source Guard，交换机上必须先开启 DHCP snooping，并且将 DHCP Server 的接口设置为 trust 接口。

（2）在接口下开启基于 IP 的 IP Source Guard

```
sw1(config)#int range f0/1 - 2
```

```
sw1(config-if-range)# ip verify source
```

3.查看结果

(1) 查看 IP source binding table

```
sw1#sh ip source binding
```

MacAddress	IpAddress	Lease(sec)	Type	VLAN	Interface
------------	-----------	------------	------	------	-----------

Total number of bindings: 0

说明：初始状态下，IP source binding table 为空，意味着 ACL 会拒绝所有流量通过，只有 DHCP 能通过。



(2)查看自动生成的 ACL

```
sw1#
```

```
##
```

```
sw1#sh ip verify source
```

Interface	Filter-type	Filter-mode	IP-address	Mac-address	Vlan
-----------	-------------	-------------	------------	-------------	------

Fa0/1	ip	active	deny-all	10	
-------	----	--------	----------	----	--

Fa0/2	ip	active	deny-all	10	
-------	----	--------	----------	----	--

```
sw1#
```

说明：因为初始状态下，IP source binding table 为空，所以 ACL 会拒绝所有流量

通过，只有 DHCP 能通过。

(3) 在 R1 开启 DHCP 自动获得地址

```
r1(config)#int f0/0
```

```
r1(config-if)#ip address dhcp
```

注：因为开启了 DHCP snooping，交换机会在 DHCP 请求中插入 option 82，所以 DHCP server 要接收此数据包：

R3:

```
r3(config)#int f0/0
```

```
r3(config-if)#ip dhcp relay information trusted
```



(4) 查看 R1 DHCP 自动获得的地址

```
r1#sh protocols f0/0
```

```
FastEthernet0/0 is up, line protocol is up
```

```
Internet address is 10.1.1.4/24
```

```
r1#
```

说明：R1 从 DHCP Server 获得的地址为 10.1.1.4。

(5) 查看 IP source binding table

```
sw1#sh ip source binding
```

MacAddress	IpAddress	Lease(sec)	Type	VLAN	Interface
------------	-----------	------------	------	------	-----------

```
-----  
00:13:1A:85:D1:60 10.1.1.4      86348    dhcp-snooping 10  FastEthernet0/1
```

Total number of bindings: 1

sw1#

说明：因为 R1 从 DHCP Server 获得的地址为 10.1.1.4，所以 DHCP snooping 将该地址记录在 IP source binding table 中。

（6）查看自动生成的 ACL

sw1#sh ip verify source

Interface	Filter-type	Filter-mode	IP-address	Mac-address	Vlan
-----------	-------------	-------------	------------	-------------	------

```
-----  
Fa0/1    ip      active   10.1.1.4      10
```

```
Fa0/2    ip      active   deny-all     10
```

sw1#

说明：因为 R1 从 DHCP Server 获得的地址被 DHCP snooping 记录在 IP source binding table 中，所以自动被 ACL 允许通过。

（7）测试 R1 到 R3 的连通性

r1#ping 10.1.1.30

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.30, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r1#

说明：因为 R1 的地址被 DHCP snooping 记录在 IP source binding table 中，并且自动被 ACL 允许通过，所以 R1 到 R3 通信正常。

（8）测试 R2 到 R3 的连通性

r2#ping 10.1.1.30

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.30, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

r2#

说明：因为 IP source binding table 通过 DHCP snooping 只记录了 R1 从 DHCP 自动获得的地址，但表中没有 R2 的地址，因此 R2 的地址默认被拒绝，所以 R2 到 R3 通信失败。

4.手工添加 IP source binding table

(1) 手工添加 R2 的 IP 地址到 IP source binding table 中

```
sw1(config)#ip source binding 0013.1a2f.1200 vlan 10 10.1.1.20 interface f0/2
```

说明：手工添加要，要同时指定 MAC 地址，IP 地址，VLAN 号，接口。

(2) 查看 IP source binding table

```
sw1#sh ip verify source
```

Interface	Filter-type	Filter-mode	IP-address	Mac-address	Vlan
Fa0/1	ip	active	10.1.1.4	10	
Fa0/2	ip	active	10.1.1.20	10	

```
sw1#
```

说明：R2 的 IP 地址已经被手工添加到 IP source binding table。

(3) 查看自动生成的 ACL

```
sw1#sh ip verify source
```

Interface	Filter-type	Filter-mode	IP-address	Mac-address	Vlan
Fa0/1	ip	active	10.1.1.4	10	
Fa0/2	ip	active	10.1.1.20	10	

sw1#

说明：自动 ACL 已经允许 IP source binding table 表中的地址通过。

（4）测试 R2 到 R3 的连通性

r2#ping 10.1.1.30

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.30, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

r2#

说明：因为 R2 的 IP 地址已经被手工添加到 IP source binding table 中，并且被自动 ACL 允许通过，所以 R2 到 R3 通信成功。

5.开启基于 IP 与 MAC 的 IP Source Guard

(1)在接口上开启基于 IP 与 MAC 的 IP Source Guard

sw1(config)#int f0/1

sw1(config-if)#ip verify source port-security

(2)开启 port-security

sw1(config)#int f0/1

sw1(config-if)#switchport port-security

说明：启基于 IP 与 MAC 的 IP Source Guard，必须开启 port-security

（3）查看 IP source binding table

```
sw1#sh sou
```

```
sw1#sh ip source binding
```

MacAddress	IpAddress	Lease(sec)	Type	VLAN	Interface
00:13:1A:85:D1:60	10.1.1.4	86034	dhcp-snooping	10	FastEthernet0/1
00:13:1A:2F:12:00	10.1.1.20	infinite	static	10	FastEthernet0/2

Total number of bindings: 2

```
sw1#
```

说明：可以看到，IP source binding table 中不仅有 IP 地址记录，还有了 MAC 地址记录，只有当数据包的 IP 和 MAC 同时匹配时，才能被放行。

（4）查看自动生成的 ACL

```
sw1#sh ip verify source
```

Interface	Filter-type	Filter-mode	IP-address	Mac-address	Vlan
Fa0/1	ip-mac	active	10.1.1.4	00:13:1A:85:D1:60	10
Fa0/2	ip	active	10.1.1.20		10

```
sw1#
```

说明：自动 ACL 已经与 IP source binding table 表中的内容同步。

（5）测试 R1 到 R3 的连通性

```
r1#ping 10.1.1.30
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.30, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

```
r1#
```

说明：因为 R1 的 IP 与 MAC 在 IP source binding table 中被记录，并且被自动 ACL 允许通过，所以 R1 到 R3 通信正常。



Security with ACL

在三层交换机上，支持多种类型的 ACL，有支持 IP 的 ACL 还有支持 MAC 的 Ethernet ACL。

IP ACL 只过滤 IPv4 流量，而 Ethernet ACL 过滤除 IP 之外的流量，也就是非 IP 流量。

根据 ACL 不同的应用，可以分成三种：Port ACL，Router ACL，VLAN ACL（VLAN map）

Port ACL

是应用在二层接口上的，并没有任何特别之处，将任何 ACL 应用到二层接口后，就称为 Port ACL，但在二层接口只支持 in 方向，一个接口只能使用一条 ACL，IP 或 MAC 的 ACL 都可以应用到二层接口，但不能在应用到 EtherChannel。

Router ACL

和路由器接口上应用的 ACL 没有区别，将任何 ACL 应用到三层接口后，就称为 Router ACL，但只能是 IP ACL，不能是 MAC ACL。Router ACL 在 in 和 out 方向上都可以使用，每个接口每个方向只能使用一条 ACL。

VLAN ACL (VLAN map)

是应用在 VLAN 与 VLAN 之间的 ACL，相同 VLAN 也是可以过滤的，只要流量是进入或离开指定的 VLAN，都会被过滤，可以同时控制二层与三层流量。准确地讲，VLAN ACL 并不是一个 ACL，只是一个能调用 ACL 到 VLAN 的技术，只要被调用的 ACL 支持什么功能，那么 VLAN ACL 就支持什么功能。当需要控制 IPv4 流量时，VLAN ACL 需要调用 IP ACL，而其它流量则需要靠调用 MAC ACL。当应用 VLAN ACL 后，进入或离开 VLAN 的流量都会被检测，无论是通过二层转发的还是三层转发的。而 VLAN ACL 是不能定义方向的，所以所有经过指定 VLAN 的流都会被过滤到。

VLAN ACL 根据所调用的 ACL 匹配到的流量，来做出是转发还是丢弃的动作，当被调用的 ACL 匹配到流量后，默认动作是转发，可以改为丢弃，而没有被匹配到的流量，默认也全部丢弃。

以上三种方式的 ACL 可以同时使用，但 Port ACL 优先于任何 ACL。

说明：

★Port ACL 支持标准，扩展 ACL 和 MAC ACL，也就是支持所有类型 ACL。

★接口上可同时应用 IP 和 MAC ACL，而 MAC ACL 是不能过滤 IP 流量的。

CCIE LAB认证经验分享千人群：539730342

★Router ACL 可用在 SVI，三层接口，或 3 层 EtherChannel，每个接口每个方向只能使用一个。

★在配置 IP ACL 时，可以使用数字，也可以使用名字，而命名 ACL 的好处是可以删除单条 ACL 语句，而数字 ACL 则不可以，只能删除整条 ACL。

★交换机上不支持 Dynamic ACL 和 Reflexive ACL。

★VLAN ACL 是不能定义方向的，也就相当于会在两个方向上同时生效，所以在配置 VLAN ACL 时，所以请注意所写的 ACL 一定要考虑来回两个方向，否则只匹配到单向的流量，另外一方向可能被丢掉。

★一个 VLAN ACL 可以用于多个 VLAN，但一个 VLAN 只能使用一个 VLAN ACL。

★被 ACL 拒绝的 ICMP，ACL 将向源发送 ICMP-unreachable。

配置



1.配置 MAC ACL

说明:在 VLAN 或 2 层接口上过滤非 IP 流量，3 层接口上不能使用 MAC ACL。

(1) 定义 MAC ACL:

```
sw1(config)#mac access-list extended ccie
```

```
sw1(config-ext-macl)#deny host 0001.0001.0001 host 0002.0002.0002 netbios
```

```
sw1(config-ext-macl)#permit any any
```

说明: 拒绝源 MAC 为 0001.0001.0001 发送到 MAC 为 0002.0002.0002 的 netbios 流量，并允许其它所有，MAC ACL 匹配的协议为非 IP 协议，协议可以不指定。

(2) 应用 MAC ACL

```
sw1(config)#int f0/1
```

```
sw1(config-if)#mac access-group ccie in
```

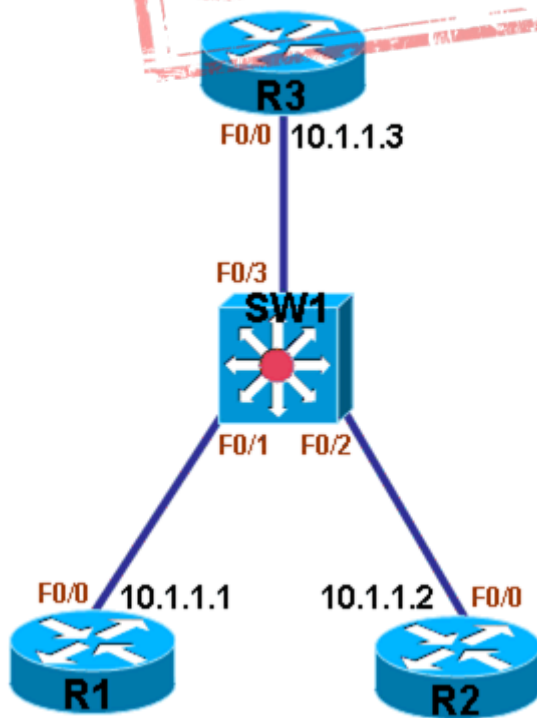
说明：在二层接口 F0/1 上使用，并且只能应用于 in 方向。因为 MAC ACL 只能过滤非 IP 流量，所以难以用实验来验证效果。

Port ACL

2.配置 Port ACL

说明：将任何 ACL 应用于二层接口，便称为 Port ACL。

以下图为例，在 SW1 上配置 Port ACL 拒绝 R3 去往 R1 的流量，并放行其它所有流量。



（1）测试在没有配置 Port ACL 之前的网络连通性

```
r3#ping 10.1.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

```
r3#ping 10.1.1.2
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

```
r3#
```

说明：在没有配置 Port ACL 之前，R3 到 R1 与 R2 的通信正常。

（2）在 SW1 上配置 Port ACL（只能用于 in 方向）

```
sw1(config)#access-list 100 deny ip host 10.1.1.3 host 10.1.1.1
```

```
sw1(config)#access-list 100 permit ip any any
```

```
sw1(config)#int f0/3
```

```
sw1(config-if)#ip access-group 100 in
```

（3）测试配置 Port ACL 之后的网络连通性

```
r3#ping 10.1.1.2
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

```
r3#
```

```
r3#ping 10.1.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

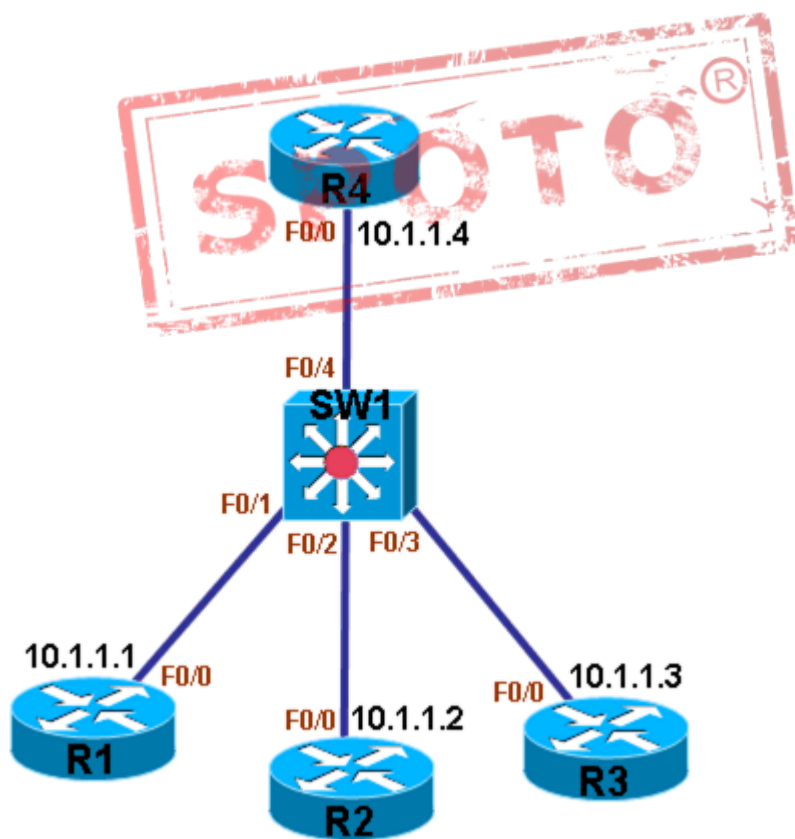
```
r3#
```

说明：可以看到，Port ACL 拒绝了 R3 去往 R1 的流量，并放行其它流量。

Router ACL

VLAN ACL

3.配置 VLAN ACL (VLAN map)



CCIE LAB认证经验分享千人群：539730342

说明：以上图为例，所有接口在 VLAN1，配置 VLAN ACL，设置 R4 到 R1 的为默认动作，设置 R4 到 R2 的为丢弃动作，其它流量不作设置。

（1）测试在没有配置 VLAN ACL 之前的网络连通性

r4#ping 10.1.1.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r4#

r4#ping 10.1.1.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

r4#

r4#ping 10.1.1.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r4#

说明：在没有配置 VLAN ACL 之前，所有通信正常。

（2）创建各 ACL

sw1(config)#access-list 111 permit ip host 10.1.1.4 host 10.1.1.1

sw1(config)#access-list 111 permit ip host 10.1.1.1 host 10.1.1.4

sw1(config)#access-list 112 permit ip host 10.1.1.4 host 10.1.1.2

sw1(config)#access-list 112 permit ip host 10.1.1.2 host 10.1.1.4

说明：分别匹配 R4 到 R1 的流量，R4 到 R2 的流量，因为 VLAN ACL 没有方向，也就是在两个方向生效，所以请注意所写的 ACL 一定要考虑来回两个方向，否则只匹配到单向的流量，另外一方向可能被丢掉。

（3）配置 VLAN ACL

sw1(config)#vlan access-map ccie 10

sw1(config-access-map)#match ip address 111

sw1(config-access-map)#exit

sw1(config)#vlan access-map ccie 20

```
sw1(config-access-map)#match ip address 112
```

```
sw1(config-access-map)#action drop
```

```
sw1(config-access-map)#exit
```

说明：设置 ACL 111 的流量为默认动作，ACL 112 的流量被明确丢弃，其它不匹配。

（4）应用 VLAN ACL

```
sw1(config)#vlan filter ccie vlan-list 1
```

说明：将 VLAN ACL 应用于 VLAN 1，也可以应用于多个 VLAN，但不能设置方向。

（5）测试配置 VLAN ACL 之后的网络连通性

```
r4#ping 10.1.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

```
r4#
```

```
r4#ping 10.1.1.2
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

r4#

r4#ping 10.1.1.3

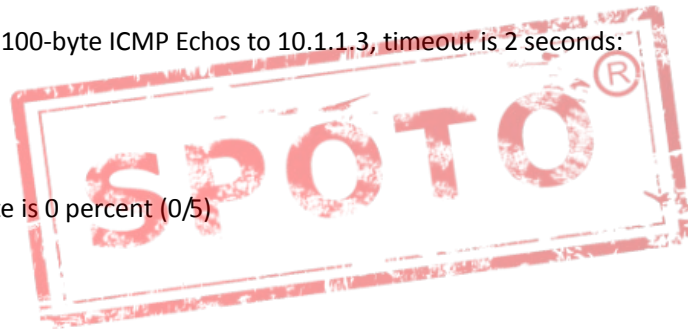
Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

r4#



说明：R4 去往 R1 的流量是默认动作，证明默认动作是允许转发的，而 R4 去往 R2 的流量被明确丢弃了，最后其它没有设置的流量，全部是被隐含拒绝的。

Storm Control

在默认情况下，交换机在接口上收到任何数据包，将尽全力转发，只有在硬件性能不足的情况下，才会丢弃数据包。在某些时候，由于协议错误，配置错误或人为攻击，导致网络流量增大时，将影响网络的性能，在这种情况下，需要在交换机上限制流量占用接口的带宽，则可以使用 Storm control 来实现。

Storm control 可以在交换机接口上限制 broadcast, ,multicast, 以及 unicast 的流量带宽，在接口上开启了 Storm control 后，Storm control 便开始监控流量从接口到交换机总线的速度，并统计每秒通过的数据包，将当前流量的速度与预先配置好的阈值作比较，阈值分为上限(rising suppression level))和下限 (falling

suppression level)，当流量的速度达到上限的阈值后，流量就会被 block，直到流量低于下限后，才会恢复正常。

在配置阈值时，可以使用以下标准来衡量带宽：

使用接口总带宽的百分比。

每秒通过的数据包个数（PPS-Packets Per Second）。

每秒通过的 Bit 数（Bps-Bit Per Second）

注：使用接口带宽百分比时，100 percent 表示不限制，0.0 表示流量全部丢弃。

默认情况下，接口上是没有流量限制的。在配置阈值时，上限（rising suppression level）必须配置，而下限（falling suppression level）却可以不配置，在没有配置下限时，下限将采用和上限相同的值。

当配置了 storm-control 限制 multicast 时 如果 multicast 的流量超过上限，那么所有的 multicast 流量都会被丢弃，其中包含如 OSPF，EIGRP 的流量都会被丢弃，但是 BPDU，CDP 的流量不会被丢弃。

配置 storm-control 时，可以在物理接口和 EtherChannel 上配置，如果是配在 EtherChannel 上，那么配置将对 EtherChannel 中的所有物理接口生效。

在配置 storm-control 时，可以设置流量达到上限后，采取相应的处理动作，可配置的动作分为 Shutdown 和 Trap，Shutdown 是在流量达到上限后，将接口陷入 error-disable 状态，Trap 是在流量达到上限后，产生一条 SNMP Trap 消息，而默认的动作是丢弃流量而不生产 SNMP Trap 消息。

配置

说明：在配置 BPS 和 PPS 作为标准时，可以使用 K，M 以及 G 为单位。

1. 在交换机接口上配置 storm-control

（1）在交换机上开启 storm-control，并定义上限和下限：

```
sw1(config)#int f0/1
```



```
sw1(config-if)# storm-control unicast level pps 100 80
```

说明：定义的上限为 PPS 100，下限为 PPS 80。

(2) 定义流量达到上限后，采限的动作：

```
sw1(config)#int f0/1
```

```
sw1(config-if)#storm-control action trap
```

说明：定义的违规动作为生产 SNMP Trap。

(3) 查看接口 storm-control 状态：

正常状态：

```
sw1#sh storm-control unicast
```

Interface	Filter State	Upper	Lower	Current
Fa0/1	Forwarding	100 pps	80 pps	10 pps

```
sw1#
```

说明：接口流量正常的状态，流量都被放行。

被丢的状态：

```
sw1#sh storm-control unicast
```

Interface	Filter State	Upper	Lower	Current
Fa0/1	Blocking	100 pps	80 pps	273 pps

```
sw1#
```

说明：接口流量达到上限的状态，流量都被丢弃。

SPAN and RSPAN

交换机在收到数据包后，将根据数据包的目标 MAC 地址来做出转发决定，只有与目标 MAC 地址对应的接口才能收到数据包，交换机并不会将数据包转发到不相关的接口上。

当网络管理者需要监控网络中的流量时，装有监控软件的主机接到交换机上之后，并不能像预期那样能够收到所要监控的流量，除非流量是原本就要发送给自己的，或者是广播流量。对于装有监控软件的主机想要从交换机上接收到其它流量，就必须依靠交换机的协助，通过交换机将其它正常流量复制一份发送到接有监控主机的接口即可。

要让交换机将正常流量复制下来并发送到相关端口，需要靠 SPAN (Switched Port Analyzer) 来实现。SPAN 允许将交换机的任意端口或任意 VLAN 上的流量复制之后发送到其它任何端口上。

因为 SPAN 要将某端口或 VLAN 的流量复制一份发送到其它端口，所以 SPAN 需要明确源和目的，SPAN 只复制从源收到的流量，然后只发送到目的。

SPAN 可以将某些接口或某些 VLAN 的流量复制下来，所以 SPAN 的源可以是物理接口，也可以是 VLAN，并且可以定义多个物理接口或多个 VLAN。复制的流量可以是接收到的，可以是发送出去的，也可以是双向的，默认为双向流量。而 SPAN 的目的有时只能是物理接口，有时只能是 VLAN，需要视情况而定。

并不是经过交换机的任何流量都能被 SPAN 复制，某些流量是不能被复制的，如三层流量需要被交换机路由到源 VLAN 的流量是不能被复制的，也就是说需要交换机查路由表将流量发送到源 VLAN 的流量是不能被复制的，但是从源 VLAN 被路由到外面去的流量还是可以被复制的。

当物理接口或 VLAN 被 SPAN 定义为源之后，源端口或源 VLAN 的流量是不会受到任何影响的，但 SPAN 的目标端口除了接收 SPAN 的流量外，不能再接收其它任何正常的流量。因此在 SPAN 的源和目标在同台交换机与不同交换机时，操作是不一样的，当 SPAN 的源和目的在同台交换机上时，被称为 Local SPAN，即 SPAN，而当 SPAN 的源和目的在不同交换机上时，被称为 Remote SPAN，即 RSPAN。

因为 RSPAN 是跨越了多台交换机的，而目标端口除了接收 SPAN 的流量外，不能再接收其它任何正常的流量，所以在为 RSPAN 定义目标时，不能将目标定义为物理接口，因为连接交换机的物理接口通常还有其它流量传播，所以在实施 RSPAN 时，必须将 SPAN 复制的流量通过发送到某个 VLAN，然后从 Trunk 上传到目标交换机，这个 VLAN 就是 RSPAN VLAN，从源到目标的每台交换机都应该配置 RSPAN VLAN。

在配置 RSPAN 时，只需要在源交换机和目标交换机上配置即可，如果中间还有交换机，中间的交换机只需要配置 RSPAN VLAN，而不需要配置其它任何参数。在源交换机上，将 SPAN 的源定义为物理接口或 VLAN，且必须将目的定义为 RSPAN VLAN，不能定义为物理接口。在目标交换机上将 SPAN 的源定义为 RSPAN VLAN，并且将目的定义为物理接口，目标交换机从 RSPAN VLAN 中收到流量后，将转发到目标接口。

在某些 IOS 版本中，配置 RSPAN 时，源交换机在定义目标时，不仅需要将 RSPAN VLAN 指定为目标，并且还需要指定一个 reflector-port 端口，系统是将流量发送到 reflector-port 端口后，再由 reflector-port 端口发送到 RSPAN VLAN，最终发送到目标交换机。而当配置一个接口为 reflector-port 端口后，这个接口就不能正常使用了。

在配置 SPAN 时，会有以下一些限制条件：

★交换机上支持最多两个 SPAN 会话。

★最多可以有 64 个目标端口，而源端口无上限。

★3 层接口也可以作为源或目的。

★源和目标的速率要一致。

★源端口可以是 EtherChannel, Fast Ethernet, Gigabit Ethernet 以及其它接口

★源也可以是 access port, trunk port, routed port, voice VLAN port

★如果一个目标端口在源 VLAN 中，则会被源排除在外。

★当源端口是 trunk 时，那就是所有 VLAN 的流量都被复制，但可以过滤某些 VLAN，配置时，

★就只有在 list 中的 vlan 的流量才会被复制。

★当一个接口变成 SPAN 的目标端口后，所有配置丢失，关闭 SPAN 后，则配置恢复。

★如果目标端口在 EtherChannel 组中，将从组中消失。

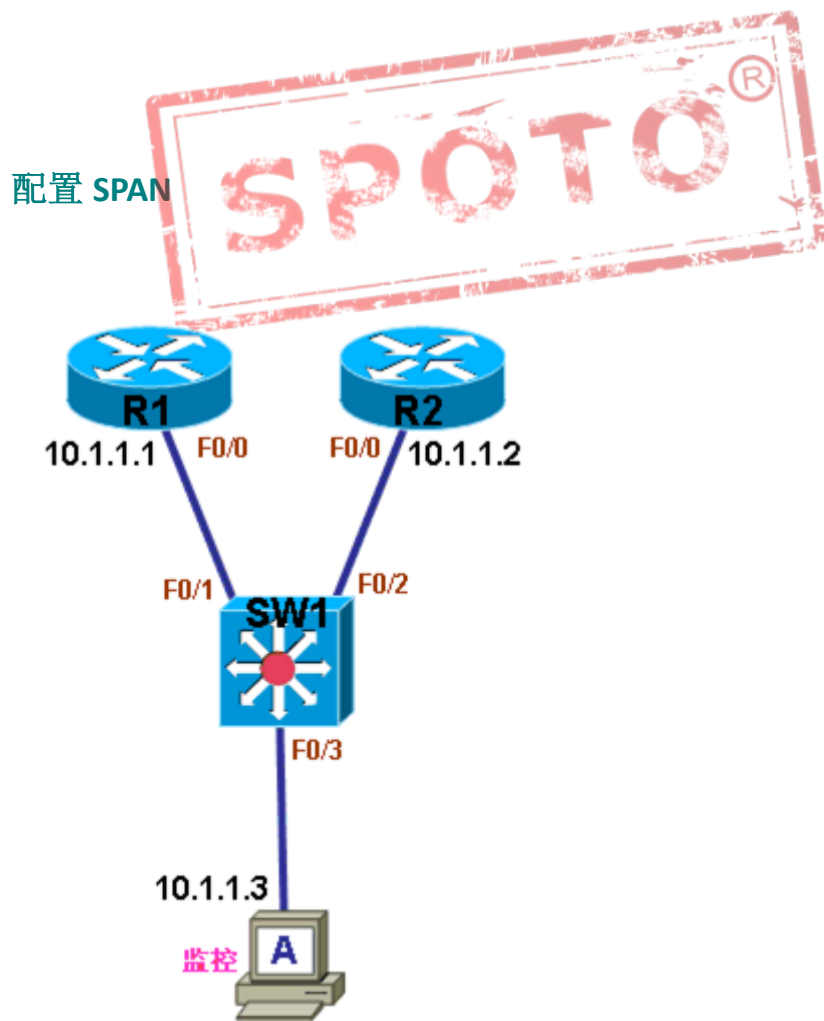
★目标不能是安全端口，也不能是源。

★目标也不能是 EtherChannel group 或正常 VLAN。

★一个目标端口不能成为两个会话的目标。

★目标端口不会转发 SPAN 流量之外的任何流量。

★在配置源时，多个源可以一条命令配完，也可以分多条命令配置。



1.测试网络连通性

（1）测试 R2 到 R1 的网络连通性

r2#ping 10.1.1.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r2#

说明：因为交换机为正常状态，所以 R2 到 R1 的通信正常。

（2）测试 R2 到主机 A 的网络连通性

r2#ping 10.1.1.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r2#

说明：因为交换机为正常状态，所以 R2 到主机 A 的通信正常。

2.在 SW1 上配置 SPAN

(1) 指定接口 F0/1 为 SPAN 源

```
sw1(config)#monitor session 1 source interface f0/1
```

说明：指定连接 R1 的接口 F0/1 为 SPAN 源接口，并且监控双向流量（默认为双向）。

(2) 指定接口 F0/3 为 SPAN 目的

```
sw1(config)#monitor session 1 destination interface f0/3
```

说明：指定连接主机 A 的接口 F0/3 为 SPAN 目的。

3.测试配置 SPAN 后的网络连通性

(1) 测试 R2 到 R1 的网络连通性

```
r2#ping 10.1.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

r2#

说明：因为 SPAN 不影响源接口的通信，所以 R2 到 R1 的通信正常。

（2）测试 R2 到主机 A 的网络连通性

r2#ping 10.1.1.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:

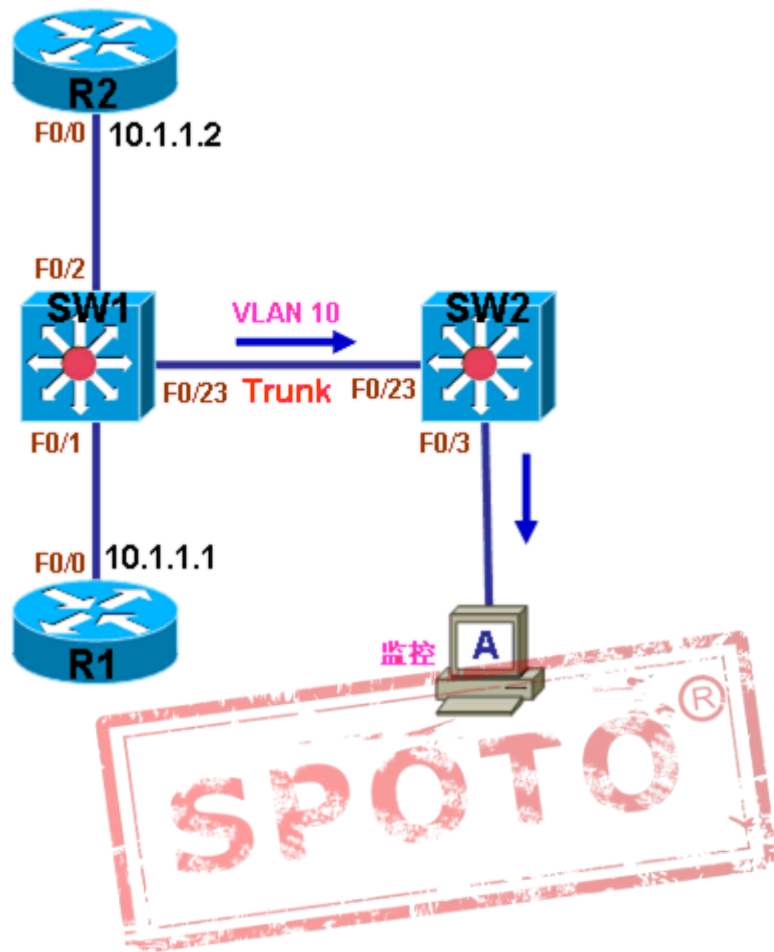
.....

Success rate is 0 percent (0/5)

r2#

说明：因为 SPAN 的目标端口除了接收 SPAN 的流量外，不能再接收其它任何正常的流量，所以 R2 到 R1 的通信失败。

配置 RSPAN



1.测试网络连通性

(1) 测试 R2 到 R1 的网络连通性

```
r2#ping 10.1.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

r2#

说明：因为交换机为正常状态，所以 R2 到 R1 的通信正常。

2.交换机上配置 RSPAN

（1）在两台交换机上配置 RSPAN VLAN 10

SW1:

```
sw1(config)#vlan 10
```

```
sw1(config-vlan)#remote-span
```

SW2:

```
sw2(config)#vlan 10
```

```
sw2(config-vlan)#remote-span
```

说明：VLAN 号可任意配置，只需将其定义为 RSPAN VLAN 即可。

（2）在源交换机上指定 RSPAN 源

```
sw1(config)#monitor session 1 source interface f0/1
```

说明：指定连接 R1 的接口 F0/1 为 RSPAN 源接口，并且监控双向流量（默认为双向）。

（3）在源交换机上指定 RSPAN 目的

```
sw1(config)#monitor session 1 destination remote vlan 10 reflector-port f0/2
```

说明：RSPAN 的目的只能为 RSPAN VLAN，并且此 IOS 版本需要指定 reflector-port，将连接 R2 的接口 F0/2 指定为 reflector-port。

（4）在目标交换机上指定 RSPAN 源

```
sw2(config)#monitor session 1 source remote vlan 10
```

说明：目标交换机上的 RSPAN 源只能为 RSPAN VLAN。

（5）在目标交换机上指定 RSPAN 目的

```
sw2(config)#monitor session 1 destination interface f0/3
```

说明：目标交换机上的 RSPAN 目的是物理端口。

3.测试配置 RSPAN 后的网络连通性

（1）测试 R2 到 R1 的网络连通性

```
r2#ping 10.1.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

```
r2#
```

说明：虽然 SPAN 不影响源接口的通信，但是当配置一个接口为 reflector-port 端

口后，这个接口就不能正常使用了，所以 R2 到 R1 的通信失败。

4.更多 SPAN 配置命令

(1) 配置过滤 Trunk 上的 VLAN

```
sw1(config)#monitor session 1 filter vlan 1 - 3 , 5
```

说明：当将 Trunk 接口配置为 SPAN 源时，Trunk 上所有的 VLAN 流量都会被复制，以上配置为只复制 Trunk 上 VLAN 1，VLAN2，VLAN3，VLAN5 的流量。

UDLD (UniDirectional Link Detection)

在交换机没有使用任何模块的接口上，如果接口出现故障，如物理故障，或不能发送数据与接收数据，自己能够快速察觉；而当接口上使用了模块后，如光纤模块，当模块上出现故障后，交换机并不能保证在任何时候都能察觉。当交换机的接口如果自己不能发送或接收数据，而对方却能发送或接收数据时，这可能会引起 STP 环路，这样的故障被称为单向链路故障，而交换机上的特性 UDLD 则是用来专门检测此类单向链路故障的。

UDLD 使用一层协议来做单向链路检测，开启了 UDLD 的接口会向外发送 Uddld hello，可以理解为交换机之间的心跳，收到 Uddld hello 的交换机必须向邻居回复，如果超过一定时间没有回复，那么就认为单向链路故障已经出现，就会采取相应措施。

UDLD 的运行分为两种模式：normal (默认) 和 aggressive。

Normal 模式只能检测光纤上的单向链路故障，而 aggressive 模式不仅能够检测光纤上的单向链路故障，还能检测双绞线上的单向链路故障。当使用 normal 模式时，检测到单向链路故障后，接口没有变化，而使用 aggressive 模式时，检测到单向链路故障后，接口会被 disable。

当交换机一端支持 UDLD，而另一端不支持 UDLD，这样的链路连在一起，是不

能做 UDLD 检测的，所以在配置 UDLD 时，必须相连的链路两端都要配置 UDLD，并且要配置成相同的模式。

配置

1.配置 UDLD

(1) 在所有接口上开启 UDLD（必须为光纤接口）

```
sw1(config)#udld enable
```

或

```
sw1(config)#udld aggressive
```



(2) 查看 UDLD:

```
sw1#show udld
```

说明：没有光纤口，没有结果。

(3) 在接口下开启 UDLD（接口下的配置将覆盖全局模式下的配置）

```
sw1(config)#int f0/8
```

```
sw1(config-if)#udld port aggressive
```

(4) 查看接口下的 UDLD

```
sw1#show udld f0/8
```

Interface Fa0/8

Port enable administrative configuration setting: Enabled / in aggressive mode

Port enable operational state: Enabled / in aggressive mode

Current bidirectional state: Unknown

Current operational state: Link down

Message interval: 7

Time out interval: 5

No neighbor cache information stored

sw1#

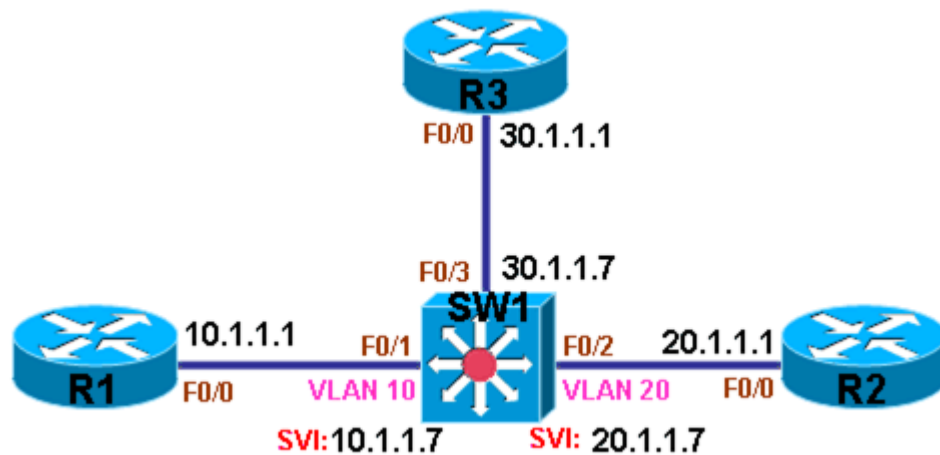
说明：可以看到，接口下已经开启 UDLD。

2. 恢复被 UDLD 关闭的接口

（1）恢复被 UDLD 关闭的接口

先 Shut 再 no shut

Fallback Bridging



从上图中的网络环境可以看出，当 R1，R2 和 R3 使用 IP 协议互相通信时，只需要让 SW1 支持 IP 路由转发功能即可，方法为在 SW1 上输入命令 `ip routing`。

当 R1，R2 和 R3 不使用 IP 协议互相通信时，如使用 DECnet 等非 IP 协议通信时，交换机并不能为其提供 VLAN 与 VLAN 之间的数据转发以及 VLAN 与三层接口之间的数据转发。要让这些在不同 VLAN 的主机能够以非 IP 协议通信，交换机必须将这些需要通信的 VLAN 或三层接口配置到同一个组中即可，这个组就是 VLAN bridging，通常称为 fallback bridging。

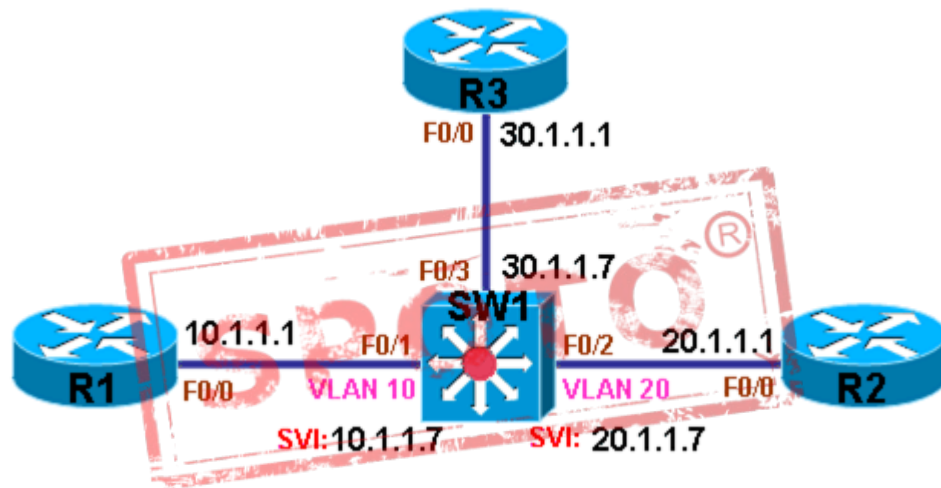
fallback bridging 为交换机上 VLAN 与 VLAN 之间以及 VLAN 与三层接口之间提供数据转发，需要通信的 VLAN 或三层接口，需要配置到同一个组中。可以看出，既然使用非 IP 协议进行通信，交换机上无论是 VLAN，还是三层接口，都是不需要配置 IP 地址的。

★交换机上最多可配置 32 个 fallback bridging 组。

★一个接口或 VLAN 只能属于一个组。

★fallback bridging 除了不能转发 IP 协议外， Address Resolution Protocol (ARP), reverse ARP (RARP), LOOPBACK, and Frame Relay ARP 都可以转发。

配置



1. 在交换机上配置 fallback bridging

(1) 在交换机上开启 fallback bridging

```
sw1(config)#bridge 10 protocol vlan-bridge
```

说明：创建 fallback bridging 组，组号为 10。

(2) 将相应接口与 VLAN 划入 fallback bridging 组中

```
sw1(config)#int vlan 10
```

```
sw1(config-if)#bridge-group 10
```

```
sw1(config)#int vlan 20
```

CCIE LAB认证经验分享千人群：539730342

```
sw1(config-if)#bridge-group 10
```

```
sw1(config)#int f0/3
```

```
sw1(config-if)#bridge-group 10
```

说明：将 VLAN 10, VLAN 20, F0/3 划入组 10 中, VLAN 10, VLAN 20 与 F0/3 可以使用非 IP 协议进行通信。

2. 查看 fallback bridging

(1) 查看 fallback bridging

```
sw1#show bridge group
```

```
Bridge Group 10 is running the VLAN Bridge compatible Spanning Tree protocol
```

```
Port 31 (Vlan10) of bridge group 10 is forwarding
```

```
Port 32 (Vlan20) of bridge group 10 is forwarding
```

```
Port 30 (FastEthernet0/3) of bridge group 10 is forwarding
```

```
sw1#
```

说明：VLAN 10, VLAN 20, F0/3 已经划入组 10 中。

(2) 查看 fallback bridging 转发状态

```
sw1#sh bridge
```

Br Group	Mac Address	State	Type	Ports
-----	-----	----	----	-----
10	0013.1a2f.0380	Forward	DYNAMIC	Fa0/3
10	0013.1a2f.1200	Forward	DYNAMIC	VI20 Fa0/2
10	0013.1a85.d160	Forward	DYNAMIC	VI10 Fa0/1
10	0013.805c.4b17	Forward	DYNAMIC	VI10 Fa0/23

sw1#

说明：因为是以以太网交换机，即使不使用 IP 协议，也会有 MAC 地址，**fallback bridging** 开启后，交换机也需要根据主机的 MAC 地址做出转发决定，并且可以手工定义哪些 MAC 被丢弃或被转发。

3.手工定义 **fallback bridging** 转发表

(1) 定义转发表

```
sw1(config)#bridge 10 address 0001.0002.0003 discard
```


```
sw1(config)#bridge 10 address 0004.0005.0006 forward
```

说明：定义 MAC 地址为 0001.0002.0003 的流量被丢弃，MAC 地址为 0004.0005.0006 的流量被转发。

(2)查看 **fallback bridging** 的地址表

```
sw1#sh bridge
```

```
00:11:32: %SYS-5-CONFIG_I: Configured from console by console
```



Br Group	Mac Address	State	Type	Ports
-----	-----	----	----	-----
10	0001.0002.0003	Discard	STATIC	-
10	0004.0005.0006	Forward	STATIC	-
10	0013.1a2f.0380	Forward	DYNAMIC	Fa0/3
10	0013.1a2f.1200	Forward	DYNAMIC	VI20 Fa0/2
10	0013.1a85.d160	Forward	DYNAMIC	VI10 Fa0/1
10	0013.805c.4b17	Forward	DYNAMIC	VI10 Fa0/23

sw1#

说明：存在动态学习的，根据交换机 MAC 地址为准，也有手工静态配置的。

IEEE 802.1x (DOT1X) Authentication

简单的说，IEEE 802.1x 是一种认证技术，是对交换机上的 2 层接口所连接的主机做认证，当主机接到开启了 IEEE 802.1x 认证的接口上，就有可能被认证，否则就有可能被拒绝访问网络。在接口上开启 IEEE 802.1x 认证后，在没有通过认证之前，只有 IEEE 802.1x 认证消息，CDP，以及 STP 的数据包能够通过。

因为主机接到开启了 IEEE 802.1x 认证的接口后，需要通过认证才能访问网络，要通过认证，只要输入合法的用户名和密码即可。交换机收到用户输入的账户信息后，要判断该账户是否合法，就必须和用户数据库做个较对，如果是数据库中存在的，则认证通过，否则认证失败，最后拒绝该用户访问网络。

交换机提供给 IEEE 802.1x 认证的数据库可以是交换机本地的，也可以是远程服务器上的，这需要通过 AAA 来定义，如果 AAA 指定认证方式为本地用户数据库 (Local)，则读取本地的账户信息，如果 AAA 指定的认证方式为远程服务器，则读取远程服务器的账户信息，AAA 为 IEEE 802.1x 提供的远程服务器认证只能是 RADIUS 服务器，该 RADIUS 服务器只要运行 Access Control Server Version 3.0 (ACS 3.0) 或更高版本即可。

当开启 IEEE 802.1x 后,并且连接的主机支持 IEEE 802.1x 认证时,将得出如下结果:

- ★如果认证通过，交换机将接口放在配置好的 VLAN 中，并放行主机的流量。
- ★如果认证超时，交换机则将接口放入 guest vlan。
- ★如果认证不通过，但是定义了失败 VLAN，交换机则将接口放入定义好的失败 VLAN 中。
- ★如果服务器无响应，定义放行，则放行。

注：不支持 IEEE 802.1x 认证的主机，也会被放到 guest vlan 中。

提示：

当交换机使用 IEEE 802.1x 对主机进行认证时，如果主机通过了认证，交换机还可以根据主机输入的不同账户而将接口划入不同的 VLAN，此方式称为 IEEE 802.1x 动态 VLAN 认证技术，并且需要在 RADIUS 服务器上做更多的设置。本文档并不对 IEEE 802.1x 动态 VLAN 认证技术做更多的介绍，如有需要，本文档将补充对 IEEE 802.1x 动态 VLAN 认证技术的详细介绍与配置说明。

当主机认证失败后，交换机可以让主机多次尝试认证，称为重认证（re-authentication），在交换机上开启 re-authentication 功能即可，默认是关闭的。并且还可以配置认证时间间隔，默认 60 秒，默认可以尝试 2 次重认证。

如果要手工对某接口重认证，在 enable 模式输入命令 `dot1x re-authenticate interface`

在交换机接口上开启认证后，只要接口从 down 状态到 up 状态，就需要再次认证。

`dot1x port-control auto` 接口开认证

对于开启了认证的接口，分为两种状态，unauthorized（未认证的）和 authorized（认证过的）。

接口的状态，可以手工强制配置，接口可选的配置状态分以下 3 种：

force-authorized：就是强制将接口直接变认证后的状态，即 authorized 状态。

force-unauthorized：就是强制将接口直接变没有认证的状态，即 unauthorized 状态。

Auto：就是正常认证状态，主机通过认证，则接口在 authorized 状态，认证失败，则在 unauthorized 状态。

注：当交换机接口从 up 到 down，或者主机离开了发送 logoff，都将合接口重新变成 unauthorized 状态。

开启了 IEEE 802.1x 认证的接口除了有状态之外，还有主机模式，称为 Host Mode，分两种模式：single-host 和 multiple-host。

在 single-host 模式（默认），表示只有一台主机能连上来。

在 multiple-hosts 模式，表示可以有多台主机连上来，并且一台主机认证通过后，所有主机都可以访问网络。

当认证超时或主机不支持认证时，接口将被划到 guest VLAN[®]，当认证失败时，将被划失败 VLAN，也就是受限制的 VLAN (restricted VLAN)，guest VLAN 和 restricted VLAN 可以定义为同一个 VLAN，并且每接配置的。其实即使划入这个 VLAN 后，也会告诉客户是认证通过，要不然会得不到 DHCP。

注：

★IEEE 802.1x 认证只能配置在静态 access 模式的接口上。

★正常工作在 IEEE 802.1x 的接口被称为 port access entity (PAE) authenticator。

★在认证超时或主机不支持认证时，才会将接口划入 guest-vlan，在 IOS 12.2(25)SE 之前，是不会将支持认证但认证失败的接口划入 guest-vlan 的，如果要开启 guest-vlan supplicant 功能，要全局配置 dot1x guest-vlan supplicant。

配置

1. 定义认证方式

（1）定义 IEEE 802.1x 使用本地账户数据库认证

```
sw1(config)#aaa new-model
```

```
sw1(config)#aaa authentication dot1x default local
```

（2）定义 IEEE 802.1x 使用 RADIUS 服务器认证

```
sw1(config)#aaa new-model
```

```
sw1(config)#aaa authentication dot1x default group radius
```

（3）定义 RADIUS 服务器

```
sw1(config)#radius-server host 10.1.1.1 auth-port 1645 acct-port 1646 key cisco
```

说明：定义 RADIUS 服务器地址为 10.1.1.1，密码为 cisco，认证端口 UDP 1645，默认为 1812，accounting 端口为 1646，默认为 1813。

2. 开启 IEEE 802.1x 认证

（1）全局开启 IEEE 802.1x 认证

```
sw1(config)#dot1x system-auth-control
```

说明：必须在全局开启 IEEE 802.1x 认证。

（2）在接口下开启 IEEE 802.1x 认证

说明：接口模式必须为静态 access

```
sw1(config)#int f0/1
```

```
sw1(config-if)#swi
```

```
sw1(config-if)#switchport mo
```

```
sw1(config-if)#switchport mode acce
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#dot1x port-control auto
```

说明：将接口设置为 auto 状态，即正常认证状态。

(3) 定义 guest VLAN 和 restricted VLAN

```
sw1(config)#vlan 10
```

```
sw1(config-vlan)#exit
```

```
sw1(config)#vlan 20
```

```
sw1(config-vlan)#exit
```

```
sw1(config)#int f0/1
```

```
sw1(config-if)#dot1x guest-vlan 10
```

```
sw1(config-if)#dot1x auth-fail vlan 20
```

说明：先在交换机上配置好 VLAN，然后定义 guest-vlan 和 restricted VLAN (auth-fail vlan)，两个 VLAN 可以设置为同一个。

3. 查看配置

(1) 查看接口配置信息

```
sw1#sh run int f0/1
```

```
Building configuration...
```

```
Current configuration : 153 bytes
```

```
!
```

```
interface FastEthernet0/1
```

```
switchport mode access

dot1x pae authenticator

dot1x port-control auto

dot1x guest-vlan 10

dot1x auth-fail vlan 20

end
```

```
sw1#
```

说明：可以看到，将接口配置为 IEEE 802.1x 后，接口自动设置为 dot1x pae authenticator。

(2) 查看接口 IEEE 802.1x 状态

```
sw1#sh dot1x interface f0/1
```

```
Dot1x Info for FastEthernet0/1
```

```
-----

PAE                                = AUTHENTICATOR

PortControl                        = AUTO

ControlDirection                  = Both

HostMode                           = SINGLE_HOST

ReAuthentication                   = Disabled

QuietPeriod                        = 60

ServerTimeout                     = 30

SuppTimeout                        = 30
```

ReAuthPeriod	= 3600 (Locally configured)
ReAuthMax	= 2
MaxReq	= 2
TxPeriod	= 30
RateLimitPeriod	= 0
Auth-Fail-Vlan	= 20
Auth-Fail-Max-attempts	= 3
Guest-Vlan	= 10

sw1#

说明：以上是 IEEE 802.1x 的默认参数。

4. 设置其它 IEEE 802.1x 参数

(1) 开启重认证功能（默认关闭）

```
sw1(config)#int f0/1
```

```
sw1(config-if)#dot1x reauthentication、
```

(2) 重认证次数（默认 2 次）

```
sw1(config)#int f0/1
```

```
sw1(config-if)#dot1x max-reauth-req 3
```

(3) 在划到限制 VLAN 之前，可以尝试几次输入（默认是 3 次）

```
sw1(config)#int f0/1
```

```
sw1(config-if)#dot1x auth-fail max-attempts 2
```


(4) 设置主机模式（默认 Single-host）

```
sw1(config)#int f0/1
```

```
sw1(config-if)#dot1x host-mode multi-host
```

(5) 查看配置

```
sw1#sh dot1x interface f0/1
```

```
Dot1x Info for FastEthernet0/1
```

```
-----  
PAE = AUTHENTICATOR
```

```
PortControl = AUTO
```

```
ControlDirection = Both
```

```
HostMode = MULTI_HOST
```

```
ReAuthentication = Enabled
```

```
QuietPeriod = 60
```

```
ServerTimeout = 30
```

```
SuppTimeout = 30
```

```
ReAuthPeriod = 3600 (Locally configured)
```

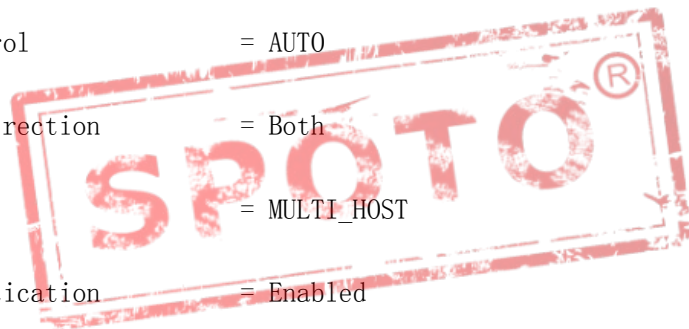
```
ReAuthMax = 3
```

```
MaxReq = 2
```

```
TxPeriod = 30
```

```
RateLimitPeriod = 0
```

```
Auth-Fail-Vlan = 20
```



```
Auth-Fail-Max-attempts    = 2
```

```
Guest-Vlan                = 10
```

```
sw1#
```

说明： 以上的配置已经显示。

5. 强制接口为 authorized 状态

(1) 直接将接口 F0/2 配置为 authorized 状态

```
sw1(config)#int f0/2
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#dot1x port-control force-authorized
```

(2) 查看接口 F0/2 的配置

```
sw1#sh run int f0/2
```

```
Building configuration...
```

```
Current configuration : 82 bytes
```

```
!
```

```
interface FastEthernet0/2
```

```
    switchport mode access
```

```
    dot1x pae authenticator
```

```
end
```

```
sw1#
```

说明：可以看到，接口只有 dot1x pae authenticator，没有 auto 字样

(3) 查看接口 F0/2 的状态

```
sw1#sh dot1x interface f0/2
```

```
Dot1x Info for FastEthernet0/2
```

```
-----  
  
PAE                                = AUTHENTICATOR  
  
PortControl                        = FORCE_AUTHORIZED  
  
ControlDirection                  = Both  
  
HostMode                          = SINGLE_HOST  
  
ReAuthentication                   = Disabled  
  
QuietPeriod                       = 60  
  
ServerTimeout                     = 30  
  
SuppTimeout                       = 30  
  
ReAuthPeriod                      = 3600 (Locally configured)  
  
ReAuthMax                         = 2  
  
MaxReq                            = 2  
  
TxPeriod                          = 30  
  
RateLimitPeriod                   = 0
```



```
sw1#
```

说明：接口已经为 authorized 状态。

(4) 直接将接口 F0/3 配置为 authorized 状态

```
sw1(config)#int f0/3
```

```
sw1(config-if)#switchport mode access
```

```
sw1(config-if)#dot1x pae authenticator
```

说明：命令 dot1x pae authenticator 也将接口直接设置为 authorized 状态。

(5) 查看接口 F0/3 的配置

```
sw1#sh run int f0/3
```

```
Building configuration...
```

```
Current configuration : 82 bytes
```

```
!
```

```
interface FastEthernet0/3
```

```
    switchport mode access
```

```
    dot1x pae authenticator
```

```
end
```

```
sw1#
```

说明：可以看到，命令 dot1x pae authenticator 等同命令 dot1x port-control force-authorized。

(6) 查看接口 F0/2 的状态

```
sw1#sh dot1x interface f0/3
```

Dot1x Info for FastEthernet0/3

```
-----  
  
PAE                                = AUTHENTICATOR  
  
PortControl                        = FORCE_AUTHORIZED  
  
ControlDirection                  = Both  
  
HostMode                           = SINGLE_HOST  
  
ReAuthentication                  = Disabled  
  
QuietPeriod                        = 60  
  
ServerTimeout                      = 30  
  
SuppTimeout                        = 30  
  
ReAuthPeriod                       = 3600 (Locally configured)  
  
ReAuthMax                          = 2  
  
MaxReq                             = 2  
  
TxPeriod                           = 30  
  
RateLimitPeriod                    = 0
```



sw1#

说明：接口已经为 authorized 状态。

6.guest-vlan supplicant

说明：在认证超时或主机不支持认证时，才会将接口划入 guest-vlan，在 IOS 12.2(25)SE 之前，是不会将支持认证但认证失败的接口划入 guest-vlan 的，如果要开

启 guest-vlan supplicant 功能，做如下配置：

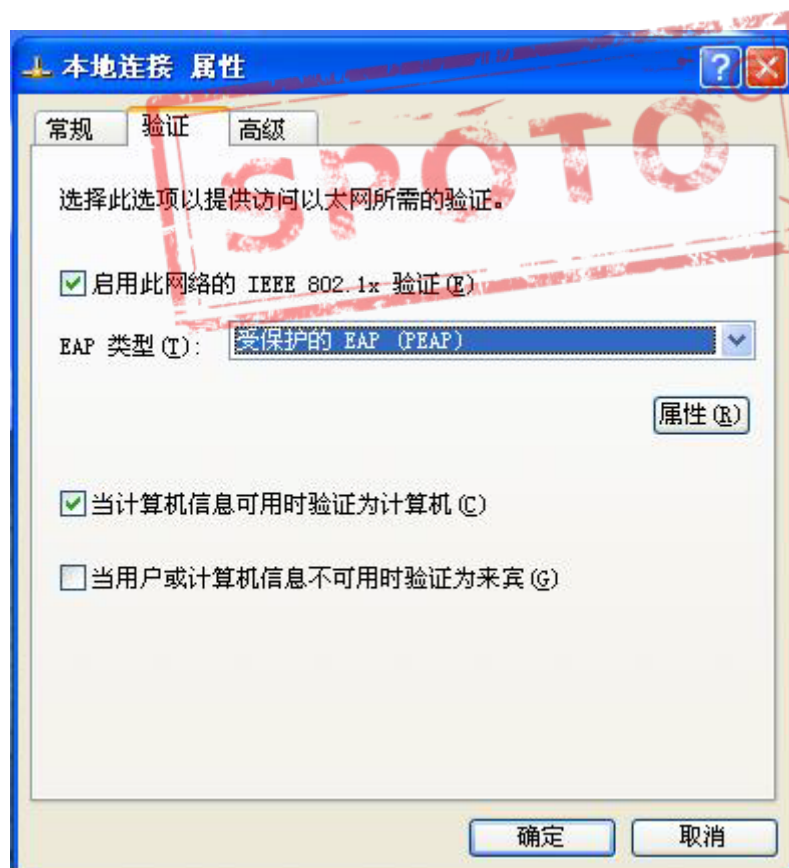
```
sw1(config)#dot1x guest-vlan supplicant
```

说明：某些 IOS 版本，此命令是隐藏命令，需要手工全部输入。

7.主机设置网卡 IEEE 802.1x 认证

说明：下面以 XP 操作系统的主机说明如何开启网卡的 IEEE 802.1x 认证

(1) 点击本地连接的“属性”，出现如下窗口：



说明：勾选“启用此网络的 IEEE 802.1x 验证”，并选择 EAP 类型为“受保护的 EAP (PEAP)”，然后点“属性”，出现如下窗口：

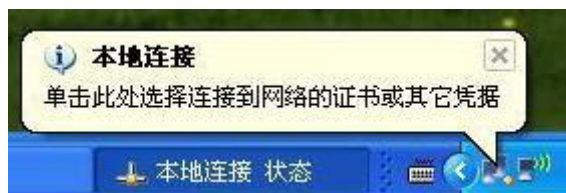


说明：请不要勾选“验证服务器证书”，选择验证方法为“安全密码 (EAP-MSCHAP v2)”，然后点击“配置”，出现如下窗口：



说明：请不要勾选“自动使用 Windows 登录名和密码（以及域，如果有的话）”。

(2) 当将此网卡连上开启 IEEE 802.1x 认证的交换机端口后，如出现如下提示：



说明：点击对话框，会弹出输入用户名和密码的窗口，如下图：



说明：只要在此对话框中输入正确的用户名和密码，即可通过 IEEE 802.1x 认证。

交换机故障恢复管理

交换机密码恢复

通常情况下，可以为交换机设置 enable 密码来提供安全，在没有 enable 密码的情况下，无法对交换机修改任何配置，因此，在忘记 enable 密码的时候，就意味着无法改动交换机信息。但是，如果能够物理上接触到交换机，便可以通过某些方法来清除已经配置的 enable 密码，甚至是交换机的所有配置。

Cisco 交换机有个叫做 nvram:的存储器，类似于 PC 机的硬盘，断电后数据不会丢

失，交换机的配置都会存放在 nvram: 里面，配置文件名为 startup-config，只要配置保存过，nvram: 里面都会出现 startup-config 这个文件，每当交换机启动时，都会读取 nvram: 里面的 startup-config 文件，如果有密码，我们就无法进入 enable 模式，在这里需要提示，在 cisco 低型号的交换机中，如 3750 以下（包括 3750），nvram: 里面的 startup-config 文件会同时在 flash: 存储器里再生成一份名为 config.text 的文件，要注意的是这两个文件属同一文件，删除任何一个，两个同时丢失；在高型号的交换机中，如 4500 系列以上（包括 45 系列），只有 nvram:，并没有名为 flash: 的这个存储器，所以就没有 config.text 这个文件，与 flash: 存储器相对应的存储器叫做 bootflash:，在这里面除了 IOS 文件，没有别的。

所以我们在交换机有 enable 密码的情况下，要清除密码，只需要让交换机不读取 nvram: 中的 startup-config 即可，由于上述原因，需要分两种情况来进行操作。

操作步骤:

第一种情况（3750 以下系列）



（1）通常在交换机前面板（正视有端口的一面）左侧，有一个名为“mode”的按钮，用手按住此按钮不放，然后拔掉交换机电源线，过 5 秒钟再次插回该电源线，如果是 3550，这时会看到交换机所有端口都会亮，等到交换机第一个端口熄灭后，则把按按钮的手放开；如果是 3560，则端口灯不会亮，但左侧面板的灯会闪烁，等到第一个灯变为绿色不闪时，放开按按钮的手，（其它型号交换机类同），这时会看到登陆交换机的界面如下

switch:

（2）接下来我们要改掉 nvram: 里面的 startup-config 文件，让交换机找不到此配置文件，便认为没有密码，但我们只须改 flash: 下的 config.text 即可。

switch: flash_init （初始化 flash 文件系统）

switch: load_helper （装载并初始化辅助映像 ROM 中的最小 IOS 映像）

CCIE LAB认证经验分享千人群：539730342

switch: dir flash: （显示 flash 文件内的文件和目录）

switch: rename flash:config.text flash:config.old （修改配置文件名，以便跳过密码）

switch: boot （重启）

（3）这时我们不需要密码便可顺利进入交换机 enable 模式，但此时交换机并未装载任何配置，此时我们在 enable 模式便可把配置文件名改回 config.text:

Switch#rename flash:config.old flash:config.text

然后再将配置导入内存正常运行

Switch#copy startup-config running-config

最后再改掉之前忘记的 enable 密码即可，并且之前的配置也没有丢失。

Switch(config)#no enable secret

第二种情况（4500 以上系列）

(1)将交换机的电源直接拔掉，并反复按 **ctrl C** 两个组合键，直到出现

rommon 1 >

模式为止。

(2)输入 **confreg:**

rommon 1 > confreg

(3)照以下情况选择

Configuration Summary :

CCIE LAB认证经验分享千人群：539730342

=> load ROM after netboot fails

=> console baud: 9600

=> autoboot from: commands specified in 'BOOT' environment variable

do you wish to change the configuration? y/n [n]: y （重要）

enable "diagnostic mode"? y/n [n]: n

enable "use net in IP bcast address"? y/n [n]: n

disable "load ROM after netboot fails"? y/n [n]: n

enable "use all zero broadcast"? y/n [n]: n

enable "break/abort has effect"? y/n [n]: n

enable "ignore system config info"? y/n [n]: y （重要）

change console baud rate? y/n [n]: n

change the boot characteristics? y/n [n]: n

Configuration Summary :

=> load ROM after netboot fails

=> ignore system config info

=> console baud: 9600

=> autoboot from: commands specified in 'BOOT' environment variable

do you wish to save this configuration? y/n [n]: y （重要）

You must reset or power cycle for new configuration to take effect

rommon 2 > reset （最后重启）

(4)此时可不需要密码便能进入交换机 **enable** 模式，但交换机此时也是按默认的引导方式从 CF 卡中启动系统，如果没有 CF 卡，那么再次重启交换机，将会再进入 **rommon 1>**这个模式，所以必须现在将交换机引导模式改为 **bootflash:**中的 IOS 启动，

如果 **bootflash:**中只有一个 IOS，则输入：

```
Switch(config)#boot system bootflash:
```

如果 **bootflash:**中有两个文件名分别为 **a.bin** 和 **b.bin** 的 IOS，如想从 **a.bin** 引导，则输入：

```
Switch(config)#boot system bootflash:a.bin
```

(5)最后保存所有改动信息：

```
Switch#write
```

交换机密码恢复管理



即使是一个规模较大的公司，可能需要的路由器设备是少数的，可以将路由器放在机房的机柜里，锁上机房门和机柜门，一般人是碰不到路由器设备的。但是如果由于公司人数众多，或者由于场地的原因，或者再由于别的原因，如双绞线的有效传输范围为 100 米，也就表示主机离交换机的距离不能超过 100 米，在这些情况下，可能需要将交换机放置在离员工近距离的地方，有时选择直接放在员工办公的 Office 房间里。因为我们知道，当交换机放置在一个能让人物理接触到的位置时，就可以通过绕过密码的方式对交换机进行配置修改，比如网络管理员在交换机上对网络做过某些访问限制，或者 QOS 带宽限制，那么就有人会尝试着接触交换机修改其中的配置，这对于网络管理员来说，是不容易发现的。

因为无论是路由器还是交换机，都可以通过物理接触来绕过密码，从而修改配置信息，而由于交换机的特殊性，所以交换机系统集成了一个特殊的功能，就是可以开启或关闭对交换机密码恢复功能，如果此功能打开，则可以通过物理接触来绕过密码，如果关闭，则交换机的全部配置信息会自动全部删除，在因为交换机被别人误动而造成配置全部丢失，作为网络管理员，是有足够的理由发现的。

配置管理密码恢复

1.查看当前的密码恢复功能

（1）查看默认的交换机密码恢复功能

Switch#sh version

Cisco IOS Software, C3560 Software (C3560-ADVIPSERVICESK9-M), Version 12.2(35)SE1, RELEASE SOFTWARE (fc1)

Copyright (c) 1986-2006 by Cisco Systems, Inc.

Compiled Tue 19-Dec-06 10:54 by antonino

Image text-base: 0x00003000, data-base: 0x01362CA0

（输出被省略）

cisco WS-C3560-24TS (PowerPC405) processor (revision D0) with 122880K/8184K bytes of memory.

Processor board ID CAT1047RJNU

Last reset from power-on

1 Virtual Ethernet interface

24 FastEthernet interfaces

2 Gigabit Ethernet interfaces

The password-recovery mechanism is disabled.

（输出被省略）

Switch#

说明：可以看到，3560 默认和密码恢复功能是关闭的，此特性会因为交换机型号的不同，默认会有所不同，如 3550 是默认打开的。

2.测试关闭了密码恢复功能的效果

（1）为交换机配置 enable 密码

Switch(config)#enable secret cisco

（2）保存配置

Switch#wr

Building configuration...

[OK]

Switch#



（3）通过物理接触来做密码恢复

说明：密码恢复方法，请参见上一节

在做密码恢复时，交换机会出现以下提示：

Switch#

Base ethernet MAC Address: 00:1a:6c:6f:fb:00

Xmodem file system is available.

The password-recovery mechanism is disabled.

CCIE LAB认证经验分享千人群：539730342

Initializing Flash...

flashfs[0]: 30 files, 3 directories

flashfs[0]: 0 orphaned files, 0 orphaned directories

flashfs[0]: Total bytes: 32514048

flashfs[0]: Bytes used: 11811840

flashfs[0]: Bytes available: 20702208

flashfs[0]: flashfs fsck took 13 seconds.

...done Initializing Flash.

Boot Sector Filesystem (bs) installed, fsid: 3

done.



The password-recovery mechanism has been triggered, but

is currently disabled. Access to the boot loader prompt

through the password-recovery mechanism is disallowed at

this point. However, if you agree to let the system be

reset back to the default system configuration, access

to the boot loader prompt can still be allowed.

Would you like to reset the system back to the default configuration (y/n)?

CCIE LAB认证经验分享千人群：539730342

说明：提示信息说明交换机的密码恢复功能已被关闭，并给出提问，如果回答 n，则等于什么都没有做，交换机将做正常启动，如下：

Would you like to reset the system back to the default configuration (y/n)?n

Boot process continuing...

Loading "flash:c3560-advipservicesk9-mz.122-35.SE1.bin"...@@@@@@@

(4) 回答 y

说明：如果上一问中回答 y，则像上述所说一样，配置将被全部清除，以下是回答 y 后，进行配置密码恢复的结果：

Would you like to reset the system back to the default configuration (y/n)?y

The system has been interrupted, and the config file

has been deleted. The following command will finish

loading the operating system software:

boot

switch:

CCIE LAB认证经验分享千人群：539730342

switch: flash_init

Initializing Flash...


...The flash is already initialized.

Setting console baud rate to 9600...

switch: load_helper

switch: dir flash:

Directory of flash:/



```
2  -rwx 8450865  <date>      c3560-advipservicesk9-mz.122-35.SE1.bin
3  drwx 832    <date>      crashinfo_ext
26 -rwx 24     <date>      private-config.text
7  drwx 832    <date>      crashinfo
```

20706304 bytes available (11807744 bytes used)

switch:

说明：可以看到，回答 y，交换机已经自动删除配置，而进入之后会发现 flash: 中根本就再也没有了 config.text 这个文件。被人误清除了配置的交换机，管理员是应该有责任发现的。