计算机网络实验报告

警示

1.实验报告如有雷同，雷同各方当次实验成绩均以 0 分计。
2.当次小组成员成绩只计学号、姓名登录在下表中的。
3.在规定时间内未上交实验报告的，不得以其他方式补交，当次成绩按 0 分计。
4.实验报告文件以 PDF 格式提交。

院系	数据科学与计算机学院	班 级	15-1 班	组长	李佳
学号	15331151	15331150	15331143		
学生	李佳	李辉旭	黎皓斌		
实验分工					
李佳	实验 6-2 中负责 PC2 以及 VLAN10 的配置和指令，练习 9 中负责 PC3 和 PC4 以及 VLAN10 的配置和指令；		李辉旭	实验 6-2 中负责 PC1 以及 VLAN20 的配置和指令，练习 9 中负责 PC1 的配置和指令；	
黎皓斌	实验 6-2 中负责 PC3 的配置和指令，练习 9 中负责 PC2 以及 VLAN20 的配置和指令；			实验 6-2 和练习 9 由三个人负责截自己控制的电脑的图；	

【实验题目】跨交换机实现 VLAN
【实验目的】理解跨交换机之间 VLAN 的特点。使在同一 VLAN 里的计算机系统能跨交换机进行相互通信、而在不同 VLAN 里的计算机系统不能进行相互通信。
【实验内容】
(1)完成实验教材第 6 章实验 6-2 的实验(p172)。
(2)完成本章习题 6 的练习 9(p217)，用 Wireshark 进行抓包的时候注意截图，分析实验结果。
(3) 跨交换机实现 VLAN 通信时，思考不用 Trunk 模式且也能进行跨交换机 VLAN 通信的替代方法，并进行实验验证。
【实验要求】
一些重要信息比如 VLAN 信息需给出截图，注意实验步骤的前后对比！
【实验记录】(如有实验拓扑，要求自行画出拓扑图，并表明 VLAN 以及相关接口。)

(1)实验 6-2 跨交换机实现 VLAN

【实验步骤】
步骤 1：实验前的测试。

（1）实验开始时，用 netsh 命令将 PC1、PC2、PC3 的网卡分别配置为下列 IP、掩码：

PC1　192.168.10.10　　255.255.255.0

PC2　192.168.10.20　　255.255.255.0

PC3　192.168.10.30　　255.255.255.0

验证三台主机是否可以两两互相 ping 通。

```
C:\Users\B403>ping 192.168.10.20

正在 Ping 192.168.10.20 具有 32 字节的数据:
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=64

192.168.10.20 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms, 最长 = 0ms, 平均 = 0ms
```

```
C:\Users\B403>ping 192.168.10.30

正在 Ping 192.168.10.30 具有 32 字节的数据:
来自 192.168.10.30 的回复: 字节=32 时间=2ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64

192.168.10.30 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms, 最长 = 2ms, 平均 = 0ms
```

用 PC1 测试发现可以 ping 通其他两个 PC，另外两个 PC 的 ping 情况类似。

（2）记录交换机 A 和交换机 B 的 VLAN 信息。

```
SwitchA#show vlan id
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
---
   1 VLAN0001                         STATIC    Gi0/1, Gi0/2, Gi0/3, Gi0/4
                                                Gi0/5, Gi0/6, Gi0/7, Gi0/8
                                                Gi0/9, Gi0/10, Gi0/11, Gi0/12
                                                Gi0/13, Gi0/14, Gi0/15, Gi0/16
                                                Gi0/17, Gi0/18, Gi0/19, Gi0/20
                                                Gi0/21, Gi0/22, Gi0/23, Gi0/24
                                                Gi0/25, Gi0/26, Gi0/27, Gi0/28

SwitchA#
```

```
SwitchB#show vlan id
VLAN Name                                Status    Ports
---- ----------------------------------- --------- -------------------------------
---
   1 VLAN0001                            STATIC    Gi0/1, Gi0/2, Gi0/3, Gi0/4
                                                   Gi0/5, Gi0/6, Gi0/7, Gi0/8
                                                   Gi0/9, Gi0/10, Gi0/11, Gi0/12
                                                   Gi0/13, Gi0/14, Gi0/15, Gi0/16
                                                   Gi0/17, Gi0/18, Gi0/19, Gi0/20
                                                   Gi0/21, Gi0/22, Gi0/23, Gi0/24
                                                   Gi0/25, Gi0/26, Gi0/27, Gi0/28

SwitchB#
```

交换机 A 和交换机 B 的 VLAN 信息如上所示，此时没有创建 VLAN，也没有划分端口。

步骤 2：在交换机 A 上创建 VLAN 10，并将端口 0/5 划分到 VLAN10 中。

（1）在交换机 A 上通过命令 show vlan id 10 验证是否已创建了 VLAN 10，查看 0/5 端口是否已划分到 VLAN10 中。

```
SwitchA#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SwitchA(config)#vlan 10
SwitchA(config-vlan)#name sales
SwitchA(config-vlan)#exit
SwitchA(config)#interface gigabitethernet 0/5
SwitchA(config-if-GigabitEthernet 0/5)#switchport access vlan 10
SwitchA(config-if-GigabitEthernet 0/5)#exit
SwitchA(config)#show vlan id 10
VLAN Name                                Status    Ports
---- ----------------------------------- --------- -------------------------------
---
  10 sales                               STATIC    Gi0/5, Gi0/24

SwitchA(config)#
```

如上图所示所示交换机 A 端口 0/5 已经划分到 VLAN10 中。

（2）检查 PC1、PC2、PC3 此时的连通情况。

```
C:\Users\B403>ping 192.168.10.20

正在 Ping 192.168.10.20 具有 32 字节的数据:
请求超时。
请求超时。
来自 192.168.10.10 的回复: 无法访问目标主机。
请求超时。

192.168.10.20 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),

C:\Users\B403>ping 192.168.10.30

正在 Ping 192.168.10.30 具有 32 字节的数据:
来自 192.168.10.10 的回复: 无法访问目标主机。
请求超时。
请求超时。
请求超时。

192.168.10.30 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),
```

PC1 不能 ping 通 PC2 和 PC3，PC2 和 PC3 之间可以互相 ping 通，以 PC1 作为截图。

步骤 3：在交换机 A 上创建 VLAN 20，并将 0/15 端口划分到 VLAN 20 中。

（1）在交换机 A 上通过命令 show vlan id 20 验证是否已创建了 VLAN 20，查看 0/15 端口是否已划分到 VLAN 20 中。

```
SwitchA(config)#vlan 20
SwitchA(config-vlan)#name technical
SwitchA(config-vlan)#exit
SwitchA(config)#interface gigabitethernet 0/15
SwitchA(config-if-GigabitEthernet 0/15)#switchport access vlan 20
SwitchA(config-if-GigabitEthernet 0/15)#exit
SwitchA(config)#show vlan id 20
VLAN Name                                Status    Ports
---- ----------------------------------- --------- -------------------------------
---
  20 technical                           STATIC    Gi0/15, Gi0/24

SwitchA(config)#
```

如上图所示所示交换机 A 端口 0/15 已经划分到 VLAN20 中。

（2）检查 PC1、PC2、PC3 此时的连通情况。

PC1 ping PC2 和 PC3

```
C:\Users\B403>ping 192.168.10.20

正在 Ping 192.168.10.20 具有 32 字节的数据:
来自 192.168.10.10 的回复: 无法访问目标主机。
请求超时。
请求超时。
请求超时。

192.168.10.20 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),

C:\Users\B403>ping 192.168.10.30

正在 Ping 192.168.10.30 具有 32 字节的数据:
来自 192.168.10.10 的回复: 无法访问目标主机。
请求超时。
请求超时。
请求超时。

192.168.10.30 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),
```

PC2 ping PC1 和 PC3

```
C:\Users\B403>ping 192.168.10.30

正在 Ping 192.168.10.30 具有 32 字节的数据:
请求超时。
请求超时。
来自 192.168.10.20 的回复: 无法访问目标主机。
请求超时。

192.168.10.30 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),

C:\Users\B403>ping 192.168.10.10

正在 Ping 192.168.10.10 具有 32 字节的数据:
来自 192.168.10.20 的回复: 无法访问目标主机。
请求超时。
请求超时。
请求超时。

192.168.10.10 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),
```

PC3 ping PC1 和 PC2

```
C:\Users\B403>ping 192.168.10.20

正在 Ping 192.168.10.20 具有 32 字节的数据:
请求超时。
请求超时。
请求超时。
来自 192.168.10.30 的回复: 无法访问目标主机。

192.168.10.20 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),

C:\Users\B403>ping 192.168.10.10

正在 Ping 192.168.10.10 具有 32 字节的数据:
请求超时。
请求超时。
请求超时。
请求超时。

192.168.10.10 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 0, 丢失 = 4 (100% 丢失),
```

PC1 处于虚拟局域网 VLAN10 中，PC2 处于 VLAN20 中，PC3 还处于实验网中，所以三者不可以共用同一个广播网，三者不能相互连通。

步骤 4：把交换机 A 与交换机 B 相连的端口(假设为端口 0/24)定义为 Tag VLAN 模式。

```
SwitchA(config)#interface gigabitethernet 0/24
SwitchA(config-if-GigabitEthernet 0/24)#switchport mode trunk
SwitchA(config-if-GigabitEthernet 0/24)#exit
SwitchA(config)#show interface gigabitethernet 0/24 switchport
Interface                      Switchport Mode      Access Native Protected VL
AN lists
------------------------------ ---------- --------- ------ ------ --------- --
--------
GigabitEthernet 0/24           enabled    TRUNK     1      1      Disabled  AL
L
SwitchA(config)#
```

如上图信息所示：交换机 A 端口 0/24 已打开（Enabled 表示已打开），模式为 trunk。

检查 PC1、PC2、PC3 此时的联通情况。

三者不能互相 ping 通，现在只是将两个交换机的 0/24 端口连接并配置，三个 PC 依然处于不同的局域网中，无法 ping 通。

步骤 5：在交换机 B 上创建 VLAN 20，并将 0/5 端口划分到 VLAN 20 中。

（1）验证已在交换机 B 上创建 VLAN20，查看端口 0/5 划分情况。

```
SwitchB#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SwitchB(config)#vlan 20
SwitchB(config-vlan)#name technical
SwitchB(config-vlan)#exit
SwitchB(config)#interface gigabitethernet 0/5
SwitchB(config-if-GigabitEthernet 0/5)#switchport access vlan 20
SwitchB(config-if-GigabitEthernet 0/5)#exit
SwitchB(config)#show vlan id 20
VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
---
  20 technical                         STATIC    Gi0/5, Gi0/24

SwitchB(config)#
就绪                    Telnet          24, 17    24行, 80列    VT100          大写 数字
```

如上图所示所示:交换机 B 的 VLAN20 已创建成功且端口 0/5 已经划分到 VLAN20 中。

（2）检查 PC1、PC2、PC3 此时的连通情况

三者不能互相 ping 通，虽然 PC2 和 PC3 都处于 VLAN 20 中，但是二者处于两个交换

机上，二者同样是不能够相互 ping 通。

步骤 6：将交换机 B 与交换机 A 相连的端口(假设为端口 0/24)定义为 Tag VLAN 模式。

```
SwitchB(config)#interface gigabitethernet 0/24
SwitchB(config-if-GigabitEthernet 0/24)#switchport mode trunk
SwitchB(config-if-GigabitEthernet 0/24)#exit
SwitchB(config)#show interface gigabitethernet 0/24 switchport
Interface                      Switchport Mode      Access Native Protected VL
AN lists
------------------------------ ---------- ---------- ------ ------ --------- --
--------
GigabitEthernet 0/24           enabled    TRUNK      1      1      Disabled  AL
L
SwitchB(config)#
```

就绪

如上图所示：交换机 B 端口 0/24 已打开（Enabled 表示已打开），模式为 trunk。

步骤 7：验证 PC2 与 PC3 能互相通信，但 PC1 与 PC3 不能互相通信。

```
C:\Users\B403>ping 192.168.10.10

正在 Ping 192.168.10.10 具有 32 字节的数据:
来自 192.168.10.30 的回复: 无法访问目标主机。
请求超时。
请求超时。
请求超时。

192.168.10.10 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),

C:\Users\B403>ping 192.168.10.20

正在 Ping 192.168.10.20 具有 32 字节的数据:
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.20 的回复: 字节=32 时间<1ms TTL=64

192.168.10.20 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms, 最长 = 0ms, 平均 = 0ms
```

两个交换机之间的端口 0/24 的 VLAN 模式都设置为 trunk，而 trunk 的作用是让连接在不同交换机上的相同 VLAN 中的主机互通，故 PC2 和 PC3 之间此时可以互相 ping 通。

启动监控软件 Wireshark，用 ping 命令测试 3 台主机连接性，并进行以下观察:

（1）主机之间能否互相通信？

PC1 不能与 PC2、PC3 进行通信

PC2 与 PC3 之间可以通信。

（2）能否监测到 PC1、PC2、PC3 的 ICMP 包？

PC1 不能检测到 PC2、PC3 的 ICMP 包

PC2 和 PC3 之间可以相互抓取到 ICMP 包

Filter: icmp Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
46	19.7079940	192.168.10.30	192.168.10.20	ICMP	74	Echo (ping) request id=0x0001, seq=393/35073, ttl=64 (reply in 47)
47	19.7082700	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) reply id=0x0001, seq=393/35073, ttl=64 (request in 46)
48	20.7092590	192.168.10.30	192.168.10.20	ICMP	74	Echo (ping) request id=0x0001, seq=394/35329, ttl=64
49	20.7094820	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) reply id=0x0001, seq=394/35329, ttl=64 (request in 48)
53	21.7103140	192.168.10.30	192.168.10.20	ICMP	74	Echo (ping) request id=0x0001, seq=395/35585, ttl=64 (reply in 54)
54	21.7105360	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) reply id=0x0001, seq=395/35585, ttl=64 (request in 53)
57	22.7113340	192.168.10.30	192.168.10.20	ICMP	74	Echo (ping) request id=0x0001, seq=396/35841, ttl=64 (reply in 58)
58	22.7115510	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) reply id=0x0001, seq=396/35841, ttl=64 (request in 57)

（3）能否捕捉到 Trunk 链路上的 VLAN ID？请讨论原因。

ICMP 包中不能抓取到 Trunk 链路上的 VLAN ID，在 lldp 包中，可以查询到 VLAN ID。信息打上 tag 标签（封装成 802.1Q 帧）发生在数据传入交换机后，此时已经离开网卡，发送端只能捕获到普通帧，带有 tag 标记的信息（802.1Q 帧）在经过 trunk 的传出口时，会把 tag 剥离，重新恢复为普通帧。总之就是 tag 是一个方法信息，不会被外部的 PC 嗅探到。

（4）查看交换机的地址表，清除地址表，适当更改、增加网线接口，然后观察与分析地址表的形成与变化过程（配合 wireshark 分析洪泛现象）。show mac-address-table 命令显示的 MAC 地址与命令提示符下通过 ipconfig/all 命令显示的 MAC 地址是否相同？

```
SwitchA(config)#show mac-address-table
Vlan        MAC Address          Type      Interface
----------  -------------------- --------  --------------------
   1        50e5.498b.9b99       DYNAMIC   GigabitEthernet 0/24
   1        5869.6c15.59c8       DYNAMIC   GigabitEthernet 0/24
  10        80c1.6ee3.49c3       DYNAMIC   GigabitEthernet 0/5
  20        50e5.498b.9b99       DYNAMIC   GigabitEthernet 0/24
  20        50e5.498b.9b9b       DYNAMIC   GigabitEthernet 0/15
SwitchA(config)#
```

就绪

```
以太网适配器 实验网:

   连接特定的 DNS 后缀 . . . . . :
   描述. . . . . . . . . . . . . . : Realtek PCIe GBE Family Controller
   物理地址. . . . . . . . . . . . : 50-E5-49-8B-9B-9B
   DHCP 已启用 . . . . . . . . . . : 否
   自动配置已启用. . . . . . . . . : 是
   本地链接 IPv6 地址. . . . . . . : fe80::f0b1:b621:f4ef:d8a1%14(首选)
   IPv4 地址 . . . . . . . . . . . : 192.168.10.20(首选)
   子网掩码 . . . . . . . . . . . . : 255.255.255.0
   默认网关. . . . . . . . . . . . :
   DHCPv6 IAID . . . . . . . . . . . : 324068681
   DHCPv6 客户端 DUID  . . . . . . . : 00-01-00-01-19-A6-C6-74-00-88-89-00-6C-0D

   DNS 服务器  . . . . . . . . . . . : fec0:0:0:ffff::1%1
                                       fec0:0:0:ffff::2%1
                                       fec0:0:0:ffff::3%1
   TCPIP 上的 NetBIOS  . . . . . . . : 已启用
```

```
SwitchB(config)#show mac-address-table
Vlan        MAC Address          Type      Interface
----------  -------------------- --------  --------------------
   1        50e5.498b.9b9b       DYNAMIC   GigabitEthernet 0/24
   1        5869.6c15.5a00       DYNAMIC   GigabitEthernet 0/24
   1        80c1.6ee3.49c3       DYNAMIC   GigabitEthernet 0/24
  20        50e5.498b.9b99       DYNAMIC   GigabitEthernet 0/5
  20        50e5.498b.9b9b       DYNAMIC   GigabitEthernet 0/24
SwitchB(config)#
```

就绪

```
以太网适配器 实验网:

   连接特定的 DNS 后缀 . . . . . :
   描述. . . . . . . . . . . . . . : Realtek PCIe GBE Family Controller
   物理地址. . . . . . . . . . . . : 50-E5-49-8B-9B-99
   DHCP 已启用 . . . . . . . . . . : 否
   自动配置已启用. . . . . . . . . : 是
   本地链接 IPv6 地址. . . . . . . : fe80::a844:e1ed:4d6:cbda%14(首选)
   IPv4 地址 . . . . . . . . . . . : 192.168.10.30(首选)
   子网掩码 . . . . . . . . . . . . : 255.255.255.0
   默认网关. . . . . . . . . . . . :
   DHCPv6 IAID . . . . . . . . . . . : 324068681
   DHCPv6 客户端 DUID  . . . . . . . : 00-01-00-01-19-A6-C6-74-00-88-89-00-6C-0D

   DNS 服务器  . . . . . . . . . . . : fec0:0:0:ffff::1%1
                                       fec0:0:0:ffff::2%1
                                       fec0:0:0:ffff::3%1
   TCPIP 上的 NetBIOS  . . . . . . . : 已启用
```

根据以上截图所示的 show mac-address-table 命令显示的 MAC 地址与命令提示符下

通过 ipconfig/all 命令显示的 MAC 地址对比发现：ipconfig/all 显示的 MAC 地址与当前使用的电脑接入交换机 VLAN 中的端口 MAC 地址一致。

```
*Nov  1 09:33:15: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtherne
t 0/5, changed state to up.
*Nov  1 09:33:24: %LINK-3-UPDOWN: Interface GigabitEthernet 0/18, changed state
to down.
*Nov  1 09:33:24: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtherne
t 0/18, changed state to down.
*Nov  1 09:33:27: %LINK-3-UPDOWN: Interface GigabitEthernet 0/15, changed state
to up.
*Nov  1 09:33:27: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtherne
t 0/15, changed state to up.
*Nov  1 09:33:27: %LINK-3-UPDOWN: Interface GigabitEthernet 0/14, changed state
to down.
*Nov  1 09:33:27: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtherne
t 0/14, changed state to down.

SwitchA#show mac-address-table
Vlan        MAC Address            Type      Interface
----------  ---------------------- --------  --------------------
   1        5869.6c15.59c8         DYNAMIC   GigabitEthernet 0/24
  10        80c1.6ee3.49c3         DYNAMIC   GigabitEthernet 0/5
  20        50e5.498b.9b99         DYNAMIC   GigabitEthernet 0/24
  20        50e5.498b.9b9b         DYNAMIC   GigabitEthernet 0/15
SwitchA#
就绪                    Telnet          24, 9    24行, 80列    VT100          大写  数字
```

```
*Nov  1 09:34:58: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtherne
t 0/5, changed state to down.
*Nov  1 09:35:01: %LINK-3-UPDOWN: Interface GigabitEthernet 0/19, changed state
to up.
*Nov  1 09:35:01: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtherne
t 0/19, changed state to up.

SwitchA#show mac-address-table
Vlan        MAC Address            Type      Interface
----------  ---------------------- --------  --------------------
   1        5869.6c15.59c8         DYNAMIC   GigabitEthernet 0/24
   1        80c1.6ee3.49c3         DYNAMIC   GigabitEthernet 0/19
  20        50e5.498b.9b99         DYNAMIC   GigabitEthernet 0/24
  20        50e5.498b.9b9b         DYNAMIC   GigabitEthernet 0/15
SwitchA#*Nov  1 09:35:05: %LINK-3-UPDOWN: Interface GigabitEthernet 0/6, changed
 state to up.
*Nov  1 09:35:05: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEtherne
t 0/6, changed state to up.
就绪                    Telnet          24, 1    24行, 80列    VT100          大写  数字
```

Capturing from 实验网 [Wireshark 1.10.1 (SVN Rev 50926 from /trunk-1.10)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000000	192.168.10.20	192.168.10.255	BROWSER	243	Local Master Announcement ¿¼, Workstation, Server, NT Workstation, Potential Browser, Mast
2	11.4188140	FujianRu_15:5a:00	LLDP_Multicast	LLDP	244	Chassis Id = 58:69:6c:15:5a:00 Port Id = Gi0/15 TTL = 121 System Name = SwitchA
3	16.3861560	fe80::f0b1:b621:f4e	ff02::1:2	DHCPv6	144	Solicit XID: 0x58a884 CID: 0001000119a6c674008889006c0d
4	16.4242700	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
5	17.1740730	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
6	17.9241180	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
7	20.2397260	fe80::a844:e1ed:4d6	ff02::1:2	DHCPv6	144	Solicit XID: 0x44c76f CID: 0001000119a6c674008889006c0d
8	22.4542070	Giga-Byt_8b:9b:9b	Broadcast	ARP	42	Who has 192.168.10.30? Tell 192.168.10.20
9	22.4547020	Giga-Byt_8b:9b:99	Giga-Byt_8b:9b:9b	ARP	60	192.168.10.30 is at 50:e5:49:8b:9b:99
10	22.4547090	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=13/3328, ttl=64 (reply in 13)
11	22.4548930	Giga-Byt_8b:9b:99	Broadcast	ARP	60	Who has 192.168.10.20? Tell 192.168.10.30
12	22.4549000	Giga-Byt_8b:9b:9b	Giga-Byt_8b:9b:99	ARP	42	192.168.10.20 is at 50:e5:49:8b:9b:9b
13	22.4550890	192.168.10.30	192.168.10.20	ICMP	74	Echo (ping) reply id=0x0001, seq=13/3328, ttl=64 (request in 10)
14	23.4554160	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=14/3584, ttl=64
15	23.4556400	192.168.10.30	192.168.10.20	ICMP	74	Echo (ping) reply id=0x0001, seq=14/3584, ttl=64 (request in 14)
16	24.4564610	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=15/3840, ttl=64 (reply in 17)
17	24.4566830	192.168.10.30	192.168.10.20	ICMP	74	Echo (ping) reply id=0x0001, seq=15/3840, ttl=64 (request in 16)
18	25.4575060	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=16/4096, ttl=64 (reply in 19)
19	25.4577250	192.168.10.30	192.168.10.20	ICMP	74	Echo (ping) reply id=0x0001, seq=16/4096, ttl=64 (request in 18)
20	32.8601300	Giga-Byt_8b:9b:9b	Broadcast	ARP	42	Who has 192.168.10.10? Tell 192.168.10.20
21	33.5418130	Giga-Byt_8b:9b:9b	Broadcast	ARP	42	Who has 192.168.10.10? Tell 192.168.10.20
22	34.5478560	Giga-Byt_8b:9b:9b	Broadcast	ARP	42	Who has 192.168.10.10? Tell 192.168.10.20
23	38.6505320	192.168.10.20	192.168.10.255	BROWSER	245	Domain/Workgroup Announcement WORKGROUP, NT Workstation, Domain Enum
24	41.4191210	FujianRu_15:5a:00	LLDP_Multicast	LLDP	244	Chassis Id = 58:69:6c:15:5a:00 Port Id = Gi0/15 TTL = 121 System Name = SwitchA
25	59.2299140	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
26	59.9850980	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
27	60.7471540	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
28	61.6348420	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
29	62.3962310	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
30	63.1532790	192.168.10.20	192.168.10.255	NBNS	92	Name query NB <a8><ac><1c>
31	69.6150140	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=21/5376, ttl=64
32	71.4194110	FujianRu_15:5a:00	LLDP_Multicast	LLDP	244	Chassis Id = 58:69:6c:15:5a:00 Port Id = Gi0/15 TTL = 121 System Name = SwitchA
33	74.2347560	Giga-Byt_8b:9b:9b	Giga-Byt_8b:9b:99	ARP	42	Who has 192.168.10.30? Tell 192.168.10.20
34	74.2348280	192.168.10.20	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=22/5632, ttl=64
35	75.2408040	Giga-Byt_8b:9b:9b	Giga-Byt_8b:9b:99	ARP	42	Who has 192.168.10.30? Tell 192.168.10.20
36	76.2468470	Giga-Byt_8b:9b:9b	Giga-Byt_8b:9b:99	ARP	42	Who has 192.168.10.30? Tell 192.168.10.20
37	78.7540590	Giga-Byt_8b:9b:9b	Broadcast	ARP	42	Who has 192.168.10.30? Tell 192.168.10.20
38	79.7600150	Giga-Byt_8b:9b:9b	Broadcast	ARP	42	Who has 192.168.10.30? Tell 192.168.10.20
39	80.7660620	Giga-Byt_8b:9b:9b	Broadcast	ARP	42	Who has 192.168.10.30? Tell 192.168.10.20
40	95.8017810	Giga-Byt_8b:9b:9b	Broadcast	ARP	42	Who has 192.168.10.10? Tell 192.168.10.20
41	98.9164080	192.168.10.20	192.168.10.255	BROWSER	245	Domain/Workgroup Announcement WORKGROUP, NT Workstation, Domain Enum

```
⊞ Frame 1: 243 bytes on wire (1944 bits), 243 bytes captured (1944 bits) on interface 0
⊞ Ethernet II, Src: Giga-Byt_8b:9b:9b (50:e5:49:8b:9b:9b), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊞ Internet Protocol Version 4, Src: 192.168.10.20 (192.168.10.20), Dst: 192.168.10.255 (192.168.10.255)

0000  ff ff ff ff ff ff 50 e5  49 8b 9b 9b 08 00 45 00   ......P. I.....E.
0010  00 e5 16 ef 00 00 40 11  cc b5 c0 a8 0a 14 c0 a8   ......@. ........
0020  0a ff 00 8a 00 8a 00 d1  97 46 11 02 9f 3a c0 a8   ........ .F...:..
0030  0a 14 00 8a 00 bb 00 00  20 4b 49 4b 4d 43 41 43   ........  KIKMCAC
0040  41 43 41 43 41 43 41 43  41 43 41 43 41 43 41 43   ACACACAC ACACACAC
0050  41 43 41 43 41 43 41 43  41 00 20 46 48 45 50 46   ACACACAC A. FHEPF
```

Packets: 41 · Displayed: 41 (100.0%) Profile: Default

交换机中有地址表：一个端口分配到了一个地址，即一个端口连接到一个主机。

当一台主机通过交换机向 PC1 发送数据包，当在交换机地指标中有目的主机 PC1，则不会洪泛而会直接转发。但是如果发送一个数据包到一个端口，而该端口并没有分配一个地址，即不清楚目的主机，则采用洪泛方式传送。

洪泛：源节点首先通过网络将数据副本传送给它的每个邻居节点，每个邻居节点再将数据传送给各自的除发送数据来的节点之外的其他。如此继续下去，直到数据传送至目标节点或者数据设定的生存期限(TTL,Time To Live)为 0 为止。

从抓包结果看到，在未更改网线接口时，ping 命令可以顺利执行，而在更改后，ping 命令无法得到成功的通信，因为此时更改了端口，VLAN 不再有效。在 35-40 数据包中可以看到，洪泛现象发生，找不到目的主机时进行扩散广播到相邻端口直到结束。

（5）判断实验是否达到预期目标。

根据实验要求和实验结果的前后比较，实验达到了预期目标。

【实验思考】

（1）实验时，要注意两台交换机之间相连的端口应该设置为 Tag VLAN 模式。配置时要注意区别每个操作模式下可执行的命令种类。交换机不可以跨模式执行命令，返回上级模式一般用 exit 命令。交换机端口在默认情况下是开启的（up 表示开启状态，down 表示关闭状态）。一般配置好 IP 地址后要用 no shutdown 命令开启端口，这样才能使物理设备端口正常通信。

（2）为什么不同的 VLAN 之间不能直接相互通信?

VLAN 是虚拟局域网,不同的 VLAN 之间的关系就相当与两个没有任何联系的局域网(物理上虽然有联系，但是逻辑上是切断了的）。 逻辑意义上的两个不同局域网之间没有任何联系，不能互相访问。在 VLAN 里面的通信方法，必须使用 ARP 解析在数据报头中指定通信目标的 MAC 地址，而 ARP 解析 MAC 地址的方法为通过广播的方式，使用 VLAN 时，能将网络划分为多个广播域，故在不同的 VLAN 中的主机，其属于不同的广播域，所以他们不能获得彼此的广播报文，因为无法相互通信。

（3）说明 VLAN 技术中的 Trunk 模式端口的用途和特点。

用途：Trunk 类型的端口可以允许多个 VLAN 通过，可以接收和发送多个 VLAN 的报文，Trunk 口一般用于连接两台交换机，可以只用一条 trunk 连接实现多个 VLAN 的扩展（因为 Trunk 允许多个 VLAN 的数据通过，如果用 access 口，那么一个 VLAN 就要一条连接，多个 VLAN 要多个连接，而交换机的接口是有限的）。简而言之，Trunk 端口就是通过一条连接实现多个 VLAN 的跨交换机扩展。

特点：TRUNK 功能用于交换机之间的级联，通过牺牲端口数来给交换机之间的数据交换提供捆绑的高带宽，提高网络速度，突破网络瓶颈，进而大幅提高网络性能;Trunk 模式端口传输多个 VLAN 信息，实现统一 VLAN 跨越不同的交换机。

（4）如何查看 Trunk 接口允许哪些 VLAN 通过?

使用命令 show interface gigabitethernet 0/24 switchport 查看 Trunk 接口允许通过的 VLAN。

（5）实验开始前要先确定 3 台 PC 处于同一个网段内，为什么要做这样的限定?

如果三台 PC 不在同一个网段内，3 台 PC 在一开始就无法互相 ping 通。VLAN 分割广播域，如果本身不在同一网段 ping 不到，那么划分 VLAN 是否成功实验结果就无法验证。

(2)练习 9

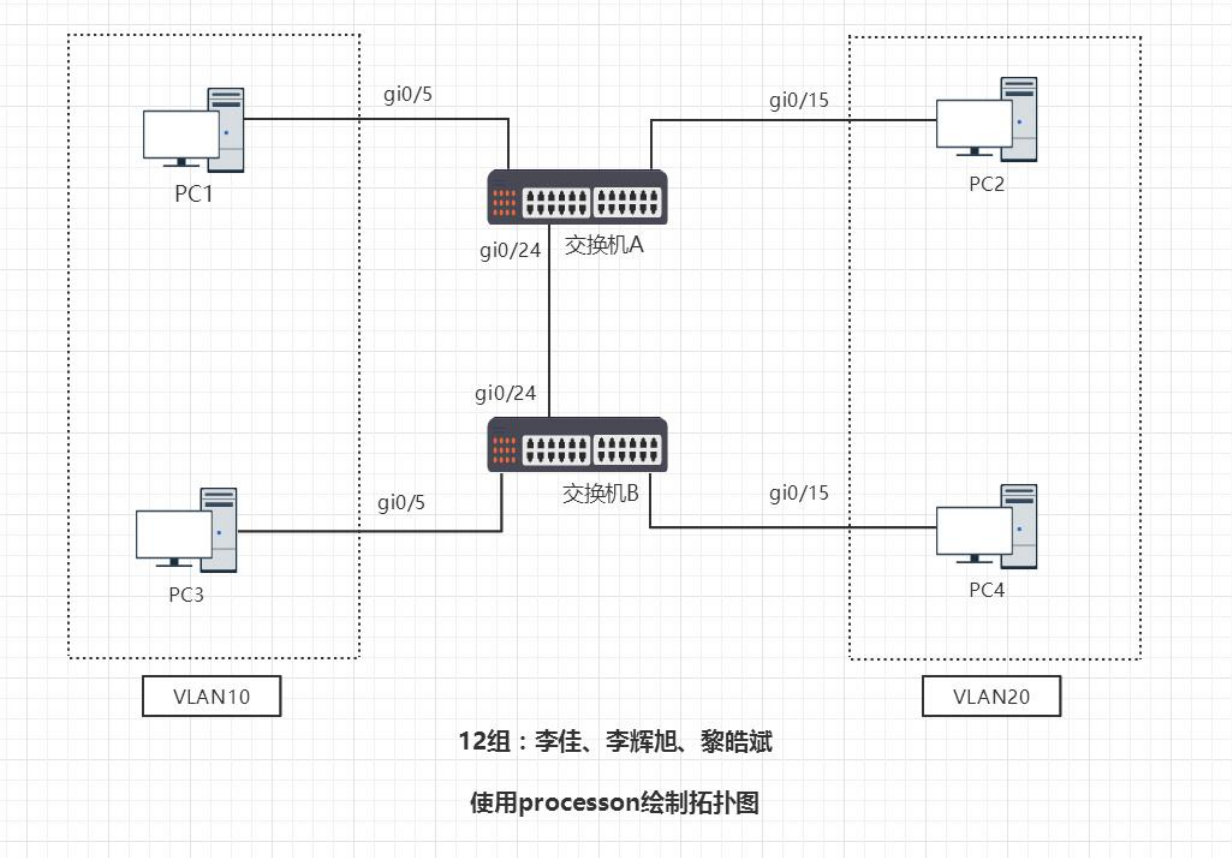
假设某企业的网络中，计算机 PC1 和 PC3 属于营销部门，PC2 和 PC4 属于技术部门，

PC1 和 PC2 连接在交换机 A 上，PC3 和 PC4 连接在交换机 B 上，而两个部门要求互相隔离。本实验的目的是实现跨两台交换机将不同端口划分到不同的 VLAN。

要求：

（1）画出拓扑图，并标明 VLAN 以及相关端口。

绘制拓扑图分析:两个部门要求互相隔离即是 PC1 只能和 PC3 互通,而 PC2 只能和 PC4 互通。其他情况两个 PC 之间不能互通。PC1 和 PC2 处于交换机 A，PC3 和 PC4 处于交换机 B 上，因此将两个交换机 0/24 端口相连并定义为 Tag VLAN 模式，PC1 和 PC3 分别连接交换机 A 和 B 的 0/5 端口并划分为 VLAN10;PC2 和 PC4 分别连接交换机 A 和 B 的 0/15 端口并划分为 VLAN20。



（2）在实验设备上完成“跨交换机实现 VLAN”实验并测试实验网连通性。

“跨交换机实现 VLAN”已经完成，实验过程及结果参考上文。

练习 9 步骤：

1. 首先设置 PC1、PC2、PC3、PC4 的 IP 和掩码如下：

PC1	192.168.10.10	255.255.255.0
PC2	192.168.10.20	255.255.255.0
PC3	192.168.10.30	255.255.255.0
PC4	192.168.10.40	255.255.255.0

设置完毕后，四台 PC 都在同一个网段中，每两台 PC 之间都可以互相 ping 通。测试结果与 6-2 类似，不再赘述。

2. 在交换机 A 上创建 VLAN10 和 VLAN20，并将 0/5 端口划分到 VLAN10 中，0/15 端口划分到 VLAN20 中，划分完成后查看端口结果如下：

```
SwitchA(config)#show vlan id 20
VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------------
  20 technical                         STATIC    Gi0/15, Gi0/24
SwitchA(config)#show vlan id 10
VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------------
  10 sales                             STATIC    Gi0/5, Gi0/24
```

3. 在交换机 B 上创建 VLAN10 和 VLAN20，并将 0/5 端口划分到 VLAN10 中，0/15 端口划分到 VLAN20 中，划分完成后查看端口结果如下：

```
SwitchB(config-if-GigabitEthernet 0/5)#show vlan id 10
VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------------
  10 sales                             STATIC    Gi0/5, Gi0/24
SwitchB(config-if-GigabitEthernet 0/5)#show vlan id 20
VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------------
  20 technical                         STATIC    Gi0/15, Gi0/24
```

4. 把交换机 A 与交换机 B 相连的 0/24 端口定义为 Tag VLAN 模式。

```
SwitchB(config-if-GigabitEthernet 0/24)#$e gigabitethernet 0/24 switchport
Interface                 Switchport Mode     Access Native Protected VLAN lists
------------------------- --------- --------- ------ ------ --------- ----------
GigabitEthernet 0/24      enabled   TRUNK     1      1      Disabled  ALL
```

配置完成后 PC1 和 PC3 处于交换机 A 和 B 上相同的虚拟局域网 VLAN10，PC2 和 PC4 处于交换机 A 和 B 上相同的虚拟局域网 VLAN20 中，PC1 和 PC3 能相互 ping 通，PC2 和 PC4 能相互 ping 通，其他情况不通。

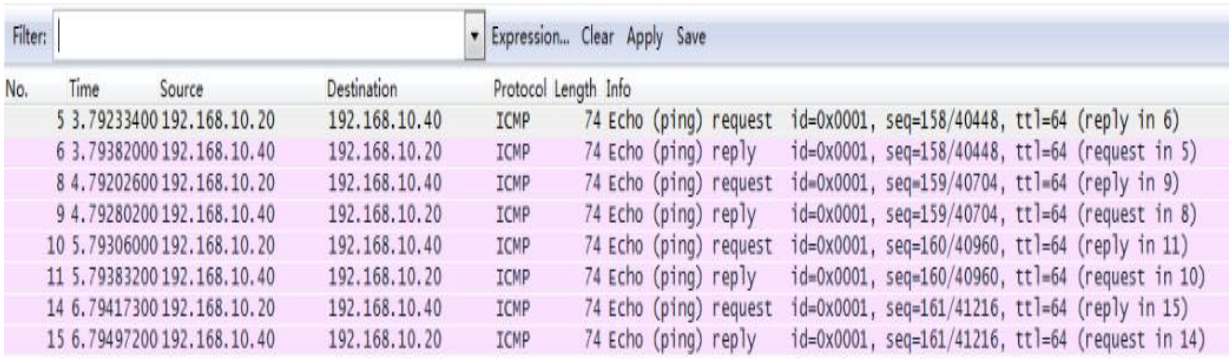
PC1 ping PC3 结果如图：

```
C:\Users\B403>ping 192.168.10.30

正在 Ping 192.168.10.30 具有 32 字节的数据:
来自 192.168.10.30 的回复: 字节=32 时间=1ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64

192.168.10.30 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms, 最长 = 1ms, 平均 = 0ms
```

PC2 ping PC4 结果类似。

（3）PC1 ping PC3, PC2 ping PC4, 在交换机 A 的端口抓包查看报文。捕获到的报文有 VLAN ID 吗?如果没有,讨论能捕获到的方法。

No.	Time	Source	Destination	Protocol	Length	Info
14	12.5193520	192.168.10.10	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=66/16896, ttl=64 (reply in 15)
15	12.5200980	192.168.10.30	192.168.10.10	ICMP	74	Echo (ping) reply id=0x0001, seq=66/16896, ttl=64 (request in 14)
18	13.5207490	192.168.10.10	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=67/17152, ttl=64 (reply in 19)
19	13.5215280	192.168.10.30	192.168.10.10	ICMP	74	Echo (ping) reply id=0x0001, seq=67/17152, ttl=64 (request in 18)
21	14.5228090	192.168.10.10	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=68/17408, ttl=64 (reply in 22)
22	14.5235880	192.168.10.30	192.168.10.10	ICMP	74	Echo (ping) reply id=0x0001, seq=68/17408, ttl=64 (request in 21)
23	15.5238580	192.168.10.10	192.168.10.30	ICMP	74	Echo (ping) request id=0x0001, seq=69/17664, ttl=64 (reply in 24)
24	15.5246360	192.168.10.30	192.168.10.10	ICMP	74	Echo (ping) reply id=0x0001, seq=69/17664, ttl=64 (request in 23)



Filter: Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
5	3.79233400	192.168.10.20	192.168.10.40	ICMP	74	Echo (ping) request id=0x0001, seq=158/40448, ttl=64 (reply in 6)
6	3.79382000	192.168.10.40	192.168.10.20	ICMP	74	Echo (ping) reply id=0x0001, seq=158/40448, ttl=64 (request in 5)
8	4.79202600	192.168.10.20	192.168.10.40	ICMP	74	Echo (ping) request id=0x0001, seq=159/40704, ttl=64 (reply in 9)
9	4.79280200	192.168.10.40	192.168.10.20	ICMP	74	Echo (ping) reply id=0x0001, seq=159/40704, ttl=64 (request in 8)
10	5.79306000	192.168.10.20	192.168.10.40	ICMP	74	Echo (ping) request id=0x0001, seq=160/40960, ttl=64 (reply in 11)
11	5.79383200	192.168.10.40	192.168.10.20	ICMP	74	Echo (ping) reply id=0x0001, seq=160/40960, ttl=64 (request in 10)
14	6.79417300	192.168.10.20	192.168.10.40	ICMP	74	Echo (ping) request id=0x0001, seq=161/41216, ttl=64 (reply in 15)
15	6.79497200	192.168.10.40	192.168.10.20	ICMP	74	Echo (ping) reply id=0x0001, seq=161/41216, ttl=64 (request in 14)

捕获到的报文没有 VLAN ID。计算机在交换机的连接端口设置为 access 模式，在发送报文的时候，会将 VLAN 信息剥离出来后进行发送，当 PC1 ping PC3 和 PC2 ping PC4 的时候，返回的信息不会含有 VLAN ID，在抓包的时候，无法抓取到 Trunk 链路上的 VLAN ID。

解决方法：将交换机 A 和交换机 B 相连的端口设置为 Hybrid Tag 模式可以抓取到 Trunk 链路上的 VLAN ID。此模式为不剥离 VLAN 信息的模式。Hybrid 类型可以允许多个 VLAN 报文发送时不打标签而 Trunk 端口只允许缺省 VLAN 报文发送时不打标签，使用 Hybrid 后，发送端口会检查是 tag 还是 untag，如果是 tag，则不撕下标签，直接发送，这样在接收端抓包可以抓到 VLAN ID。

(3)跨交换机实现 VLAN 通信时，思考不用 Trunk 模式且也能进行跨交换机 VLAN 通信的替代方法，并进行实验验证。

跨交换机实现 VLAN 通信时，不用 Trunk 模式且也能进行跨交换机 VLAN 通信的替代方法是将端口的模式设置为 Hybrid Tag Mode。Hybrid 类型的端口可以允许多个 VLAN 通过，可以接收和发送多个 VLAN 的报文，可以用于交换机之间连接，也可以用于连接用户的计算机。Hybrid 接口允许多个 VLAN 通过并且还能够指定哪些 VLAN 数据帧需要被剥离标签等。利用练习 9 把交换机 A 和交换机 B 的 0/24 端口设置为 Hybrid Tag 模式的配置方法如下：

```
SwitchA(config)# interface gigabitethernet 0/24
SwitchA(config-if-gigabitethernet 0/24)# switchport mode hybrid
SwitchA(config-if-gigabitethernet 0/24)# exit

SwitchB(config)# interface gigabitethernet 0/24
SwitchB(config-if-gigabitethernet 0/24)# switchport mode hybrid
SwitchB(config-if-gigabitethernet 0/24)# exit
```

设置完成后测试结果如下：PC1 ping PC3 可以连通，PC1 ping PC2 无法连通。

计算机网络实验报告

```
C:\Users\B403>ping 192.168.10.30

正在 Ping 192.168.10.30 具有 32 字节的数据:
来自 192.168.10.30 的回复: 字节=32 时间=2ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.10.30 的回复: 字节=32 时间<1ms TTL=64

192.168.10.30 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms, 最长 = 2ms, 平均 = 0ms
```

```
C:\Users\B403>ping 192.168.10.20

正在 Ping 192.168.10.20 具有 32 字节的数据:
来自 192.168.10.10 的回复: 无法访问目标主机。
请求超时。
请求超时。
请求超时。

192.168.10.20 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 1, 丢失 = 3 (75% 丢失),
```

本次实验完成后，请根据组员在实验中的贡献，请实事求是，自评在实验中应得的分数。（按百分制）

学号	学生	自评分
15331151	李佳	100
15331150	李辉旭	99
15331143	黎皓斌	99

【交实验报告】

上传实验报告：ftp://222.200.180.109/　　截止日期（不迟于）：1 周之内

上传包括两个文件：

（1）小组实验报告。上传文件名格式：小组号_ Ftp 协议分析实验.pdf （由组长负责上传）

例如: 文件名“10_ Ftp 协议分析实验.pdf”表示第 10 组的 Ftp 协议分析实验报告

（2）小组成员实验体会。每个同学单独交一份只填写了实验体会的实验报告。只需填写自己的学号和姓名。

文件名格式：小组号_学号_姓名_ Ftp 协议分析实验.pdf （由组员自行上传）

例如: 文件名“10_05373092_张三_ Ftp 协议分析实验.pdf”表示第 10 组的 Ftp 协议分析实验报告。

注意：不要打包上传！