

## 实验：VLAN

### HCIP 分解实验 - VLAN

臧家林制作



VLAN 实验 1：VLAN 基本配置

VLAN 实验 2：VLAN 间的通信

VLAN 实验 3：Mux VLAN

VLAN 实验 4：端口隔离

VLAN 实验 5：端口安全

VLAN 实验 6：交换机对数据帧的处理

VLAN 实验 7：基于 IP 地址划分 VLAN

= = = = =

### **VLAN 实验 1：VLAN 基本配置**

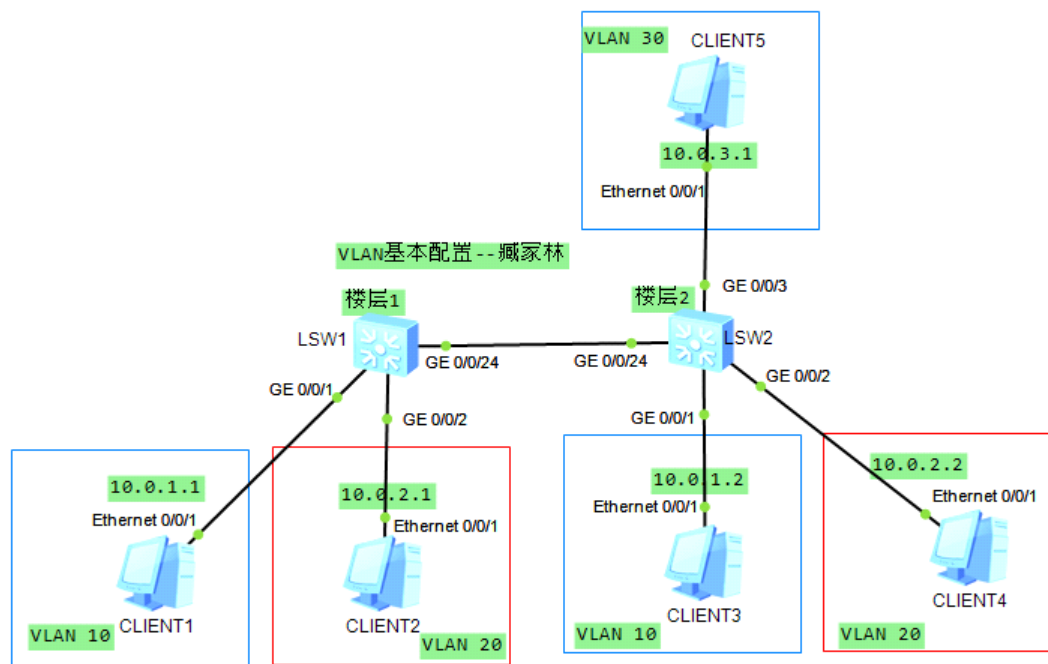
交换机的 VLAN 端口可以分为 Access , Trunk 和 Hybrid 3 种类型。

Access 端口是交换机上用来直接连接用户终端的端口，它只

允许属于该端口的缺省 VLAN 的帧通过。Access 端口发往用户终端的帧一定不带 VLAN 标签。

Trunk 端口是交换机上用来连接其他交换机的端口，它可以允许属于多个 VLAN 的帧通过。

Hybrid 端口是交换机上既可以连接用户终端，又可以连接其他交换机的端口。Hybrid 端口也可以允许属于多个 VLAN 的帧通过，并且可以在出端口的方向上将某些 VLAN 帧的标签剥掉。




VLAN 10 ,VLAN20 都能与 VLAN 30 通信  
但 VLAN 10 VLAN 20 不能相互通信

基本配置

在 SW1 创建 VLAN 10,20 ,30

SW1 :  
vlan 10  
vlan 20  
vlan 30

查看建立情况<Huawei>display vlan

VID	Type	Ports
-----		
1	common	UT:GE0/0/1(U)      GE0/0/2(U)      GE0/0/3(D) GE0/0/5(D)      GE0/0/6(D)      GE0/0/7(D) GE0/0/9(D)      GE0/0/10(D)      GE0/0/11(D) GE0/0/13(D)      GE0/0/14(D)      GE0/0/15(D) GE0/0/17(D)      GE0/0/18(D)      GE0/0/19(D) GE0/0/21(D)      GE0/0/22(D)      GE0/0/23(D)
10	common	
20	common	
30	common	

默认情况下，VLAN 1 无需手工创建就会自动存在，并且所有端口都默认属于 VLAN 1.

SW2 批量创建 VLAN 10,20,30

SW2 :  
vlan batch 10 20 30

配置 Access 端口并划分 VLAN

交换机上虽然创建了相应的 VLAN，但由于终端设备无法识别和处理 Tagged 帧，所以还需要配置 Access 端口。

SW1 上的，g0/0/1, g0/0/2 为 Access 端口

```
interface g0/0/1
port link-type access
interface g0/0/2
port link-type access
```

将接口划到相应的 VLAN 中

```
interface g0/0/1
port default vlan 10
interface g0/0/2
port default vlan 20
```

在 SW1 上查看 VLAN 的相关信息 display vlan

```
10    common    UT:GE0/0/1(U)

20    common    UT:GE0/0/2(U)

30    common
```

在 SW2 上也进行相应的操作

SW2 :

```
interface g0/0/1
port link-type access
port default vlan 10
interface g0/0/2
port link-type access
port default vlan 20
interface g0/0/3
port link-type access
port default vlan 30
```

=====

## 配置 Trunk 端口实现跨交换机通信

现在 PC1 ping PC3，是不通的

虽然 PC1 PC3 属于同一个 VLAN 10，但是二者间的通信需要跨越交换机，为此需要将 SW1 SW2 之间的链路配置为 Trunk 链路，并允许携带 VLAN10 标签的帧通过。

在 SW1 上的 g0/0/24 接口下配置为 Trunk 端口，让端口允许 VLAN 10,20,30 的帧通过。默认情况下，Trunk 端口只允许 VLAN 1 的帧通过。

```
int g0/0/24
port link-type trunk
port trunk allow-pass vlan 10 20 30
```

SW2 上也进行相应的操作

```
int g0/0/24
port link-type trunk
port trunk allow-pass vlan 10 20 30
```

查看接口的类型， display port vlan active

```
[SW2]dis port vlan active
```

```
T=TAG U=UNTAG
```

Port	Link Type	PVID	VLAN List
GE0/0/1	access	10	U: 10
GE0/0/2	access	20	U: 20
GE0/0/3	access	30	U: 30
GE0/0/4	hybrid	1	U: 1

在 SW1 上查看 VLAN 的相关信息 display vlan

```
10    common  UT:GE0/0/1(U)
                TG:GE0/0/24(U)
```

```
20    common  UT:GE0/0/2(U)
                TG:GE0/0/24(U)
```

```
30    common  TG:GE0/0/24(U)
```

可以看到，SW1 的 g0/0/24 端口现在以 Tagged 的形式被划分进了 VLAN 10,VLAN20,VLAN 30  
现在 PC1，PC3 之间是可以通信的

=====

### 使用 Hybrid 端口实现不同 VLAN 间的通信

要求 VLAN 10 能与 VLAN 30 通信，VLAN 20 能与 VLAN 30 通信

SW1 删掉之前的 Access 端口配置命令

```
int g0/0/1
undo port default vlan
undo port link-type
int g0/0/2
undo port default vlan
undo port link-type
```

修改端口为 Hybrid 端口，默认情况下交换机的端口就是 Hybrid 端口

```
int g0/0/1
port link-type hybrid
int g0/0/2
port link-type hybrid
```

指示端口需要将携带相应 VLAN 标签的帧以 Untagged 的形式进行发送

```
int g0/0/1
port hybrid untagged vlan 10 30
int g0/0/2
port hybrid untagged vlan 20 30
```

指示端口对收到的 Untagged 帧添加相应的 VLAN 标签

```
int g0/0/1
port hybrid pvid vlan 10
int g0/0/2
port hybrid pvid vlan 20
```

修改 SW1 的 g0/0/24 为 Hybrid，并将端口以 Tagged 方式加入进 VLAN 10,20,30

```
int g0/0/24
undo port trunk allow-pass vlan 10 20 30
port link-type hybrid
port hybrid tagged vlan 10 20 30
```

在 SW2 上也进行相应的操作

```
int g0/0/1
undo port default vlan
port link-type hybrid
port hybrid untagged vlan 10 30
port hybrid pvid vlan 10
int g0/0/2
undo port default vlan
port link-type hybrid
port hybrid untagged vlan 20 30
port hybrid pvid vlan 20
int g0/0/3
undo port default vlan
port link-type hybrid
port hybrid untagged vlan 10 20 30
port hybrid pvid vlan 30
int g0/0/24
undo port trunk allow-pass vlan 10 20 30
port link-type hybrid
port hybrid tagged vlan 10 20 30
```

在 SW1 上查看 VLAN 信息 display vlan



```
10    common    UT:GE0/0/1(U)
          TG:GE0/0/24(U)

20    common    UT:GE0/0/2(U)
          TG:GE0/0/24(U)

30    common    UT:GE0/0/1(U)          GE0/0/2(U)
          TG:GE0/0/24(U)
```

可以看到，此时 g0/0/1 已经以 Untagged 的形式被划分至 VLAN 10 和 VLAN30，对于 VLAN 10,VLAN 30 的帧，都会剥离其 VLAN 标签后发送。

g0/0/2 已经以 Untagged 的形式被划分至 VLAN 20 和 VLAN30，对于 VLAN 20,VLAN 30 的帧，都会剥离其 VLAN 标签后发送。

测试 PC1 PC5 的连通性，是可以通的 VLAN 10 可以访问 VLAN 30

```
PC>ping 10.0.3.1

Ping 10.0.3.1: 32 data bytes, Press Ctrl_C to break
From 10.0.3.1: bytes=32 seq=1 ttl=128 time=32 ms
From 10.0.3.1: bytes=32 seq=2 ttl=128 time=15 ms
From 10.0.3.1: bytes=32 seq=3 ttl=128 time=31 ms
From 10.0.3.1: bytes=32 seq=4 ttl=128 time=15 ms
From 10.0.3.1: bytes=32 seq=5 ttl=128 time=31 ms
```

VLAN 10 与 VLAN 20 间是不通的

```
PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl_C to break
From 10.0.1.1: Destination host unreachable
From 10.0.1.1: Destination host unreachable
From 10.0.1.1: Destination host unreachable
From 10.0.1.1: Destination host unreachable
From 10.0.1.1: Destination host unreachable
```

VLAN 30 是可能 ping 通 VLAN 10,VLAN 20，达到要求

= = = = =

## VLAN 实验 2：VLAN 间的通信

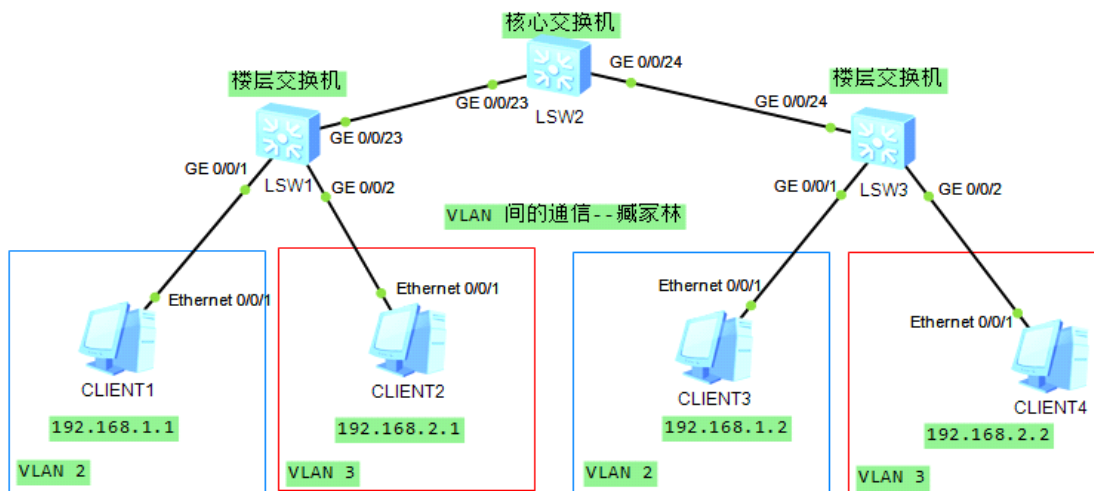
通常情况下，如果不采用一些特殊的方法（如采用 Hybrid 端口的办法），不同的 VLAN 之间是不能够进行二层（数据链路层）通信的，这也是 VLAN 技术的基本出发点；一般地，VLAN 之间的通信是需要第三层（网络层）才能实现的。

实现 VLAN 间的三层通信的方法有很多，最为传统的方法是使用路由器。除此之外，常用的方法还有很多，例如，在交换机上使用 VLANIF 接口，在交换机上使用 VLAN 聚合方法等。

VLANIF 接口只是一个逻辑意义上的三层接口。采用 VLANIF 接口的方法时，每一个 VLAN 都对应了交换机上的一个 VLANIF 接口，不同的 VLAN 对应了不同的 VLANIF 接口，并且每个 VLAN 中的终端设备的网关地址就是所对应的 VLANIF 接口的 IP 地址。显然，使用 VLANIF 接口方法的一个主要缺点就是比较耗费 IP 地址资源，这是因为每一个不同的 VLAN 都必须对应一个不同的 VLANIF 接口，而每个不同的 VLANIF 接口都必须配置一个不同的 IP 地址。

VLAN 间的通信也可以通过使用 VLAN 聚合的方法来实现。VLAN 聚合使用了两种类型的 VLAN，分别称为 Sub-VLAN 和 Super-VLAN。VLAN 聚合的方法可以节省大量的 IP 地址资源。这是因为一个 Super-VLAN 需要配置一个 VLANIF 接口，并为该 VLANIF 接口配置一个 IP 地址，但该 Super-VLAN 下

的各个 Sub-VLAN 都无需再配置 VLANIF 接口。



在 SW1 SW3 上创建 VLAN 2 VLAN3,并将相应的接口划分到 VLAN 中

```
SW1:  
undo ter mo  
sy  
sys SW1  
vlan batch 2 3  
int g0/0/1  
port link-type access  
port default vlan 2  
int g0/0/2  
port link-type access  
port default vlan 3  
q
```

SW3:

```
undo ter mo
sy
sys SW3
vlan batch 2 3
int g0/0/1
port link-type access
port default vlan 2
int g0/0/2
port link-type access
port default vlan 3
q
```

在 SW1 ,SW2 ,SW3 上完成 Trunk 端口配置，允许所有 VLAN 帧通过

=====

### Port-group

Port-group 是端口组的意思，Port-group 命令就是将要进行同样操纵的端口添加到一个组里，在这个组里进行操作就行了。将要处理的端口加入到一个组里，对这个组进行操作后，系统会自动对组里的成员进行操作。这样在处理多个端口时就变得很方便了。

把 SW2 的 2 个接口都做成 trunk

```
port-group 1
group-member g0/0/23
group-member g0/0/24
port link-type trunk
port trunk allow-pass vlan 2 3
```

=====

SW1:  
int g0/0/23  
port link-type trunk  
port trunk allow-pass vlan 2 3

SW2:  
undo ter mo  
sy  
sys SW2  
int g0/0/23  
port link-type trunk  
port trunk allow-pass vlan 2 3  
int g0/0/24  
port link-type trunk  
port trunk allow-pass vlan 2 3  
q

SW3:  
int g0/0/24  
port link-type trunk  
port trunk allow-pass vlan 2 3

现在用 PC 1 ping PC 3 发现不通

```
PC>ping 192.168.1.2

Ping 192.168.1.2: 32 data bytes, Press Ctrl_C to break
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
From 192.168.1.1: Destination host unreachable
```

原因是 SW2 上没有创建 VLAN 2,VLAN3

SW2 :

vlan batch 2 3

```
PC>ping 192.168.1.2

Ping 192.168.1.2: 32 data bytes, Press Ctrl_C to break
From 192.168.1.2: bytes=32 seq=1 ttl=128 time=78 ms
From 192.168.1.2: bytes=32 seq=2 ttl=128 time=62 ms
From 192.168.1.2: bytes=32 seq=3 ttl=128 time=62 ms
From 192.168.1.2: bytes=32 seq=4 ttl=128 time=78 ms
From 192.168.1.2: bytes=32 seq=5 ttl=128 time=46 ms
```

现在就可以 ping 通了

=====

使用 VLANIF 接口实现 VLAN 间的通信

SW2:

int vlanif 2

ip add 192.168.1.100 24

int vlanif 3

ip add 192.168.2.100 24

4 台 PC 的网关设置好

测试一下,PC 1 ping PC2 ,PC4 是可以通的

```
PC>ping 192.168.2.1

Ping 192.168.2.1: 32 data bytes, Press Ctrl_C to break
From 192.168.2.1: bytes=32 seq=1 ttl=127 time=156 ms
From 192.168.2.1: bytes=32 seq=2 ttl=127 time=62 ms
From 192.168.2.1: bytes=32 seq=3 ttl=127 time=47 ms
From 192.168.2.1: bytes=32 seq=4 ttl=127 time=47 ms
From 192.168.2.1: bytes=32 seq=5 ttl=127 time=78 ms
```

=====

使用 VLAN 聚合实现 VLAN 间通信

为了减少 VLANIF 接口的数目，节省 IP 地址的使用，我们还

可以使用 VLAN 聚合的方法来实现不同 VLAN 间的通信。

SW2:

```
undo interface vlanif 2  
undo interface vlanif 3  
vlan 4
```

```
int g0/0/23  
undo port trunk allow-pass vlan 4  
int g0/0/24  
undo port trunk allow-pass vlan 4
```

Super-VLAN 是不能包含任何物理接口的，但目前 SW2 的 g0/0/23, g0/0/24 已经做为 Trunk 端口被划分进了所有 VLAN，所以需要将 g0/0/23, g0/0/24 从 VLAN4 中移除。

### 配置 VLAN4 为 Super-VLAN

将 VLAN2 ,VLAN3 作为 Sub-VLAN 划分进 Super-VLAN

SW2:

```
vlan 4  
aggregate-vlan  
access-vlan 2 3
```

VLAN4 创建 VLANIF 接口，配置 IP 地址为，  
192.168.0.100 16，然后开启 ARP 代理功能

SW2:

```
int vlanif 4  
ip add 192.168.0.100 16  
arp-proxy inner-sub-vlan-proxy enable
```

修改 PC 上的网关为 192.168.0.100

IPv4 配置

☒ 静态 ☐ DHCP ☐ 自动获取 DNS 服务器地址

IP 地址: 192 . 168 . 1 . 1 DNS1: 0 . 0 . 0 . 0

子网掩码: 255 . 255 . 255 . 0 DNS2: 0 . 0 . 0 . 0

网关: 192 . 168 . 0 . 100

修改之后，4 台 PC 是可以相互 ping 通的

= = = = =

## VLAN 实验 3 : Mux VLAN

[multiplex\\_百度翻译](#)

**multiplex** 英 ['mʌltɪpleks] 美 ['mʌltə,pleks]

- adj. 多元的，多倍的，复式的；多部的，复合的，多样的，多重的；[电讯]多路传输的；  
n. 多路；多厅影院，多剧场影剧院；  
v. 多路传输，多路复用；多重发讯；

[全部释义>>](#)

[principal\\_百度翻译](#)

**principal** 英 ['prɪnsəpəl] 美 ['prɪnsəpəl]

- adj. 主要的；本金的；最重要的；资本的；  
n. 本金；首长，负责人；主要演员，主角；[法] 委托人，当事人；

[全部释义>>](#)



[subordinate\\_百度翻译](#)

**subordinate** 英 [sə'bo:dɪnət] 美 [sə'bo:rdɪnət]

adj. 下级的; 级别或职位较低的; 次要的; 附属的;

n. 部属; 部下, 下级;

vt. 使...居下位, 使在次级; 使服从; 使从属;

[全部释义>>](#)

[separate\\_百度翻译](#)

**separate** 英 ['seprət] 美 ['sepəreɪt]

vt.& vi. 分开; 分离 (使); 区分; 隔开;

vt. 分离 (混合物); 分居; 分类; 割开;

vi. 分手; 断裂; 分居 (夫妻); 断绝关系;

[全部释义>>](#)

MUX VLAN 实现了二层流量的弹性管控。

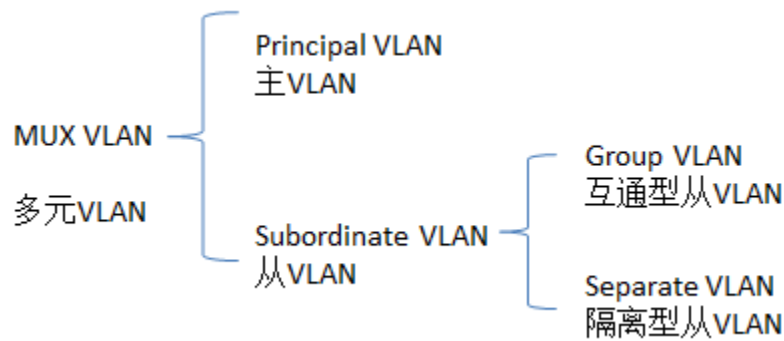
## MUX VLAN

作用：MUX VLAN ( Multiplex vlan ) 提供了一种在 VLAN 内的端口间进行二层流量隔离的机制。

需求：在企业网络中，企业员工和企业客户可以访问企业的服务器。对于企业来说，希望企业内部员工之间可以互相交流，而企业客户之间是隔离的，不能够互相访问。通过 MUX VLAN 提供的二层流量隔离的机制可以实现企业内部员工之间可以互相交流，而企业客户之间是隔离的。

原理：MUX VLAN 分为 Principal VLAN 和 Subordinate VLAN，

Subordinate VLAN 又分为 Separate VLAN 和 Group VLAN

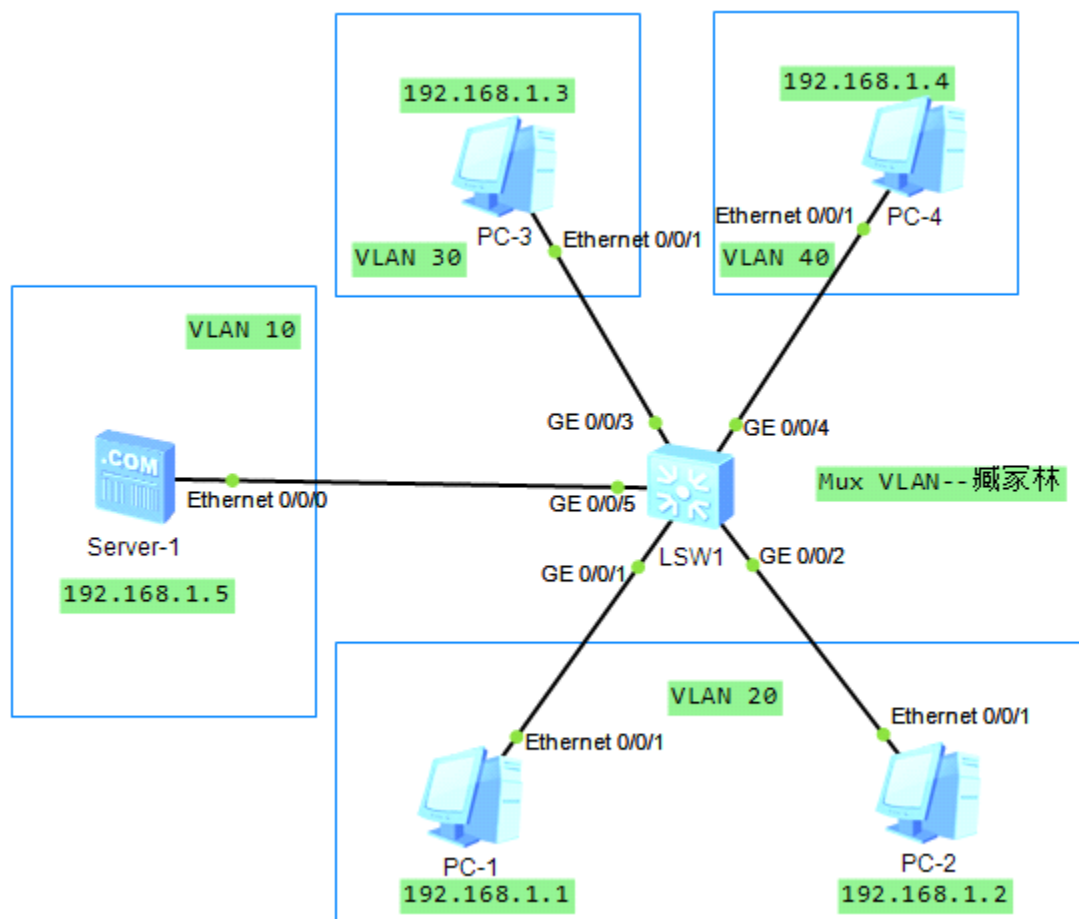


在实际的企业网络环境中，往往需要所有的终端用户都能够访问某些特定的服务器，而用户之间的访问控制规则比较复杂。在这样的场景下，使用普通 VLAN 划分的方法往往是很难满足需求的，通常的解决方法是使用 Mux VLAN

Mux VLAN 拥有一个 Principal VLAN,即主 VLAN，同时拥有多个与主 VLAN 关联的 Subordinate VLAN,即从 VLAN。从 VLAN 又有两种类型，一种是 Separate VLAN,即隔离型从 VLAN，另一种是 Group VLAN,即互通型从 VLAN。任何从 VLAN 中的设备都能够与主 VLAN 中的设备进行通信。除此之外，互通型从 VLAN 中的设备只能与本互通型从 VLAN 中的设备进行通信，不能与其他互通型从 VLAN 中的设备进行通信，也不能与隔离型从 VLAN 中的设备进行通信。隔离型从 VLAN 中的设备不能与互通型从 VLAN 中的设备进行通信，也不能与其他隔离型从 VLAN 中以及本隔离型从 VLAN 中的设备进行通信。

交换机上加入 Mux VLAN 的端口只能允许一个 VLAN 的帧通过，允许多个 VLAN 的帧通过的端口是不能被加入到 Mux VLAN 中的。

在一个主 VLAN 中，隔离型从 VLAN 有一个，互通型从 VLAN 可以有多个



要求 VLAN 30 ,VLAN 40 只能与 VLAN 10 通信 ,

VLAN 30 不能访问 VLAN40, 也不能访问 VLAN 20

VLAN 20 中的两台设备相互间可以访问 , 也可以访问 VLAN 10

没有配置之前 , 5 台设备都可以相互 ping 通

=====

使用 Hybrid 端口实现

g0/0/1,g0/0/2 配置为 Hybrid，并要求端口对收到的 Untagged 帧添加 VLAN 20 的标签后进行转发，且在发送属于 VLAN 10 和 VLAN 20 的帧之前进行去标签处理。

SW1:

```
vlan batch 10 20 30 40
```

```
int g0/0/1
port link-type hybrid
port hybrid untagged vlan 10 20
port hybrid pvid vlan 20
int g0/0/2
port link-type hybrid
port hybrid untagged vlan 10 20
port hybrid pvid vlan 20
```

g0/0/5 配置为 Hybrid，并要求端口对收到的 Untagged 帧添加 VLAN 10 的标签后进行转发，且在发送属于 VLAN 10 ,VLAN20,VLAN30 和 VLAN 40 的帧之前进行去标签处理。

```
int g0/0/5
port link-type hybrid
port hybrid untagged vlan 10 20 30 40
port hybrid pvid vlan 10
```

```
int g0/0/3
port link-type hybrid
port hybrid untagged vlan 10 30
port hybrid pvid vlan 30
```

```
int g0/0/4
```

```
port link-type hybrid
port hybrid untagged vlan 10 40
port hybrid pvid vlan 40
```

PC 间进行 ping 的测试

= = = = =

### 使用 MUX VLAN 实现

VLAN 10 为主 VLAN , VLAN20 为互通型从 VLAN , VLAN30 隔离型从 VLAN ( VLAN 30 和 VLAN40 合并为 VLAN 30 ) 。 Mux VLAN 的端口仅能够允许一个 VLAN 的帧通过 , 所以需要将加入 Mux VLAN 的端口类型修改为 Access,要删掉之前的配置

SW1:

```
int g0/0/1
undo port hybrid pvid vlan
undo port hybrid untagged vlan 10 20
port link-type access
port default vlan 20
int g0/0/2
undo port hybrid pvid vlan
undo port hybrid untagged vlan 10 20
port link-type access
port default vlan 20
int g0/0/3
undo port hybrid pvid vlan
undo port hybrid untagged vlan 10 30
port link-type access
port default vlan 30
int g0/0/4
undo port hybrid pvid vlan
```

```
undo port hybrid untagged vlan 10 40
port link-type access
port default vlan 30
int g0/0/5
undo port hybrid pvid vlan
undo port hybrid untagged vlan 10 20 30 40
port link-type access
port default vlan 10
```

VLAN 10 为主 VLAN, VLAN 20 互通型从 VLAN  
VLAN 30,为隔离型从 VLAN , 只能有一个隔离 VLAN  
SW1:

```
vlan 10
mux-vlan
subordinate group 20
subordinate separate 30
```

交换机端口开启 Mux VLAN 功能

```
int g0/0/1
port mux-vlan enable
int g0/0/2
port mux-vlan enable
int g0/0/3
port mux-vlan enable
int g0/0/4
port mux-vlan enable
int g0/0/5
port mux-vlan enable
```

```
<SW1>display mux-vlan
```

```
[SW1]dis mux-vlan
```

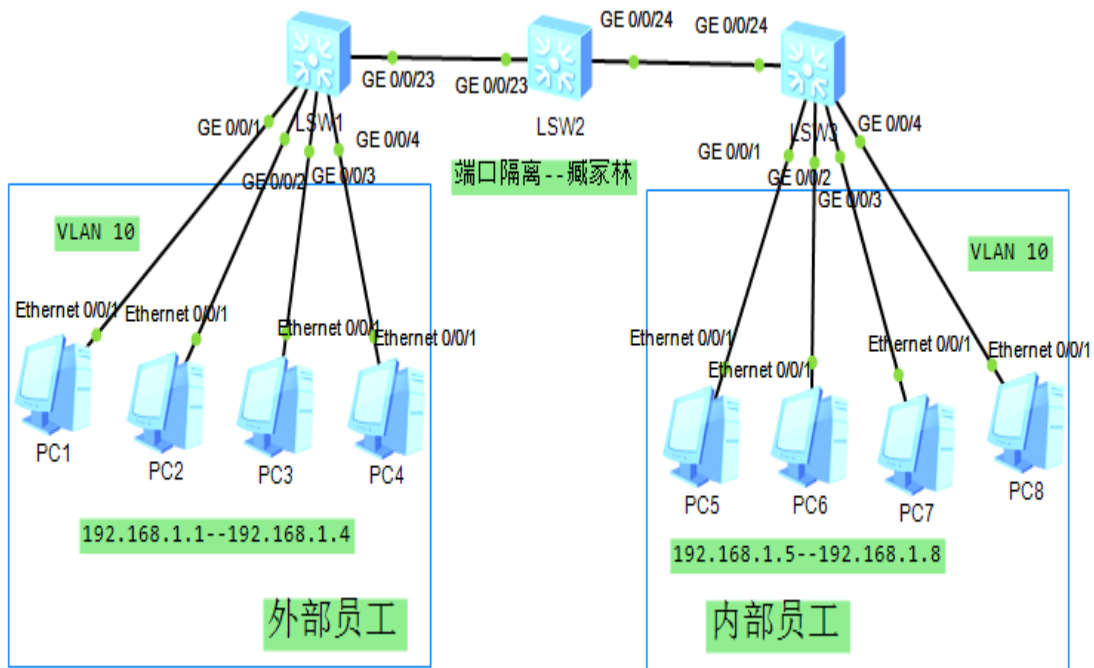
Principal	Subordinate	Type	Interface
10	-	principal	GigabitEthernet0/0/5
10	30	separate	GigabitEthernet0/0/3 GigabitEthernet0/0/4
10	20	group	GigabitEthernet0/0/1 GigabitEthernet0/0/2
10	40	group	

PC 间相互 ping 一下，满足要求

= = = = =

## **VLAN 实验 4：端口隔离**

为了实现用户之间的二层隔离，可以将不同的用户加入不同的 VLAN，但这样会浪费有限的 VLAN 资源。采用端口隔离功能，可以实现同一 VLAN 内端口之间的隔离。用户只需要将端口加入到同一隔离组中，就可以实现隔离组内端口之间二层数据的隔离。端口隔离功能为用户提供了更安全、更灵活的组网方案。



企业内部的员工允许相互通信，属于企业外部的员工不允许相互通信，外部员工与内部员工之间允许通信。

## 基本配置

SW1 :

```
undo ter mo
```

```
sy
```

```
sys SW1
```

```
vlan 10
```

```
int g0/0/1
```

```
port link-type acc
```

```
port default vlan 10
```

```
int g0/0/2
```

```
port link-type acc
```

```
port default vlan 10
```

```
int g0/0/3
```

```
port link-type acc
```



```
port default vlan 10
int g0/0/4
port link-type acc
port default vlan 10
int g0/0/23
port link-type trunk
port trunk allow-pass vlan 10
q
```

```
SW2 :
undo ter mo
sy
sys SW2
vlan 10
int g0/0/23
port link-type trunk
port trunk allow-pass vlan 10
int g0/0/24
port link-type trunk
port trunk allow-pass vlan 10
q
```

```
SW3:
undo ter mo
sy
sys SW3
vlan 10
int g0/0/1
port link-type acc
port default vlan 10
int g0/0/2
port link-type acc
port default vlan 10
int g0/0/3
```

```
port link-type acc
port default vlan 10
int g0/0/4
port link-type acc
port default vlan 10
int g0/0/24
port link-type trunk
port trunk allow-pass vlan 10
q
```

测试，几台 PC 在同一个 VLAN，同一个网段是可以相互通信的

PC>ping 192.168.1.5 -c 1

```
PC>ping 192.168.1.5 -c 1

Ping 192.168.1.5: 32 data bytes, Press Ctrl_C to break
From 192.168.1.5: bytes=32 seq=1 ttl=128 time=110 ms

--- 192.168.1.5 ping statistics ---
 1 packet(s) transmitted
 1 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 110/110/110 ms

PC>ping 192.168.1.6 -c 1

Ping 192.168.1.6: 32 data bytes, Press Ctrl_C to break
From 192.168.1.6: bytes=32 seq=1 ttl=128 time=141 ms
```

### 配置端口隔离

将 SW1 的 4 个接口配置为端口隔离，为外部员工提供服务

SW1：

```
int g0/0/1
port-isolate enable group 1
int g0/0/2
port-isolate enable group 1
int g0/0/3
port-isolate enable group 1
int g0/0/4
port-isolate enable group 1
```

查看一下

display port-isolate group 1 查看所有创建的隔离组情况

```
[SW1]dis port-isolate group 1
The ports in isolate group 1:
GigabitEthernet0/0/1    GigabitEthernet0/0/2    GigabitEthernet0/0/3
GigabitEthernet0/0/4
```

ping 测试一下。外部员工之间，不能通信，但可以与内部员工通信

192.168.1.1 不可以 ping 通 1.2 ， 1.3 ， 1.4  
可以与 1.5 ， 1.6 ， 1.7 ， 1.8 通信

也可以创建两个组，group 1，group 2，两个组内的 PC 是可以相互通信的

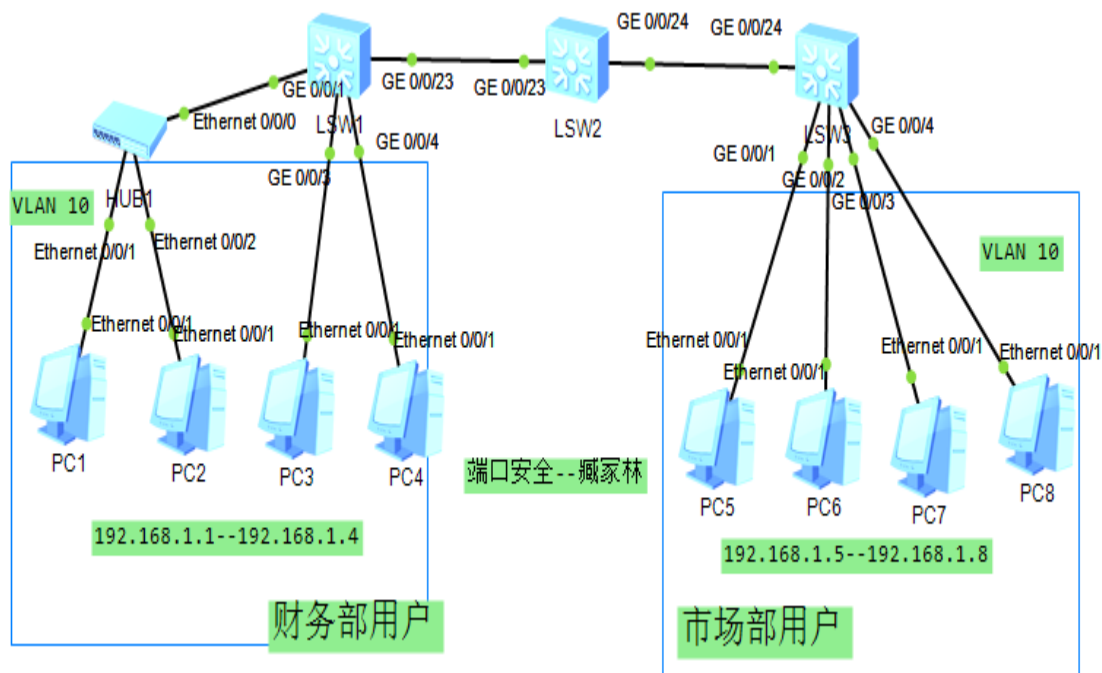
= = = = =

## **VLAN 实验 5：端口安全**

配置端口安全功能，将接口学习到的 MAC 地址转换为安全 M

AC 地址，接口学习的最大 MAC 数量达到上限后不再学习新的 MAC 地址，只允许学习到 MAC 地址的设备通信。这样可以阻止其他非信任用户通过本接口和交换机通信，提高设备与网络的安全性。

端口安全（Port Security）通过将接口学习到的动态 MAC 地址转换为安全 MAC 地址（包括安全动态 MAC、安全静态 MAC 和 Sticky MAC）阻止非法用户通过本接口和交换机通信，从而增强设备的安全性。



```
int g0/0/1
```

port-security enable，使能端口安全功能。

port-security mac-address sticky，使能接口 Sticky MAC 功能。

port-security max-mac-num max-number，配置接口 Sticky MAC 学习限制数量。

缺省情况下，接口学习的 MAC 地址限制数量为 1。

( 可选 ) port-security protect-action { protect | restrict | shutdown }，配置端口安全保护动作。

缺省情况下，端口安全保护动作为 restrict。

restrict :丢弃源 MAC 地址不存在的报文并上报告警。推荐使用 restrict 动作。

protect : 只丢弃源 MAC 地址不存在的报文，不上报告警。

shutdown : 接口状态被置为 error-down，并上报告警。

( 可选 ) 执行命令 port-security mac-address sticky mac-address vlan vlan-id，

手动配置一条 sticky-mac 表项。

## 基本配置

SW1 :

undo ter mo

sy

sys SW1

vlan 10

int g0/0/1

port link-type acc

port default vlan 10

int g0/0/3

port link-type acc

port default vlan 10

int g0/0/4

port link-type acc

```
port default vlan 10
int g0/0/23
port link-type trunk
port trunk allow-pass vlan 10
q
```

```
SW2 :
undo ter mo
sy
sys SW2
vlan 10
int g0/0/23
port link-type trunk
port trunk allow-pass vlan 10
int g0/0/24
port link-type trunk
port trunk allow-pass vlan 10
q
```

```
SW3:
undo ter mo
sy
sys SW3
vlan 10
int g0/0/1
port link-type acc
port default vlan 10
int g0/0/2
port link-type acc
port default vlan 10
int g0/0/3
port link-type acc
port default vlan 10
int g0/0/4
```

```
port link-type acc
port default vlan 10
int g0/0/24
port link-type trunk
port trunk allow-pass vlan 10
q
```

## 配置端口安全

```
SW1 :
int g0/0/1
port-security enable
```

默认接口学习的 MAC 地址限制数量为 1  
PC1 ping 192.168.1.3 可以通，但 PC2 ping 192.168.1.3  
就不可以通了

不通之后，查看 Trap 缓冲区记录的所有信息。  
<SW1>display trapbuffer

```
<SW1>display trapbuffer
```

```
Trapping buffer configuration and contents : enabled
```

```
Allowed max buffer size : 1024
```

```
Actual buffer size : 256
```

```
Channel number : 3 , Channel name : trapbuffer
```

```
Dropped messages : 0
```

```
Overwritten messages : 0
```

```
Current messages : 6
```

```
#Nov 29 2017 13:17:37-08:00 SW1 L2IFPPI/4/PORTSEC_ACTION_ALARM:OID 1.3.6.1.4.1.2  
011.5.25.42.2.1.7.6 The number of MAC address on interface (6/6) GigabitEthernet  
0/0/1 reaches the limit, and the port status is : 1 (1:restrict;2:protect;3:shu  
tdown)
```

```
#Nov 29 2017 13:16:30-08:00 SW1 DS/4/DATASYNC_CFGCHANGE:OID 1.3.6.1.4.1.2011.5.2
```

如果两个都要通，需要修改限制数量

```
int g0/0/1
```

```
port-security max-mac-num 5
```

使能接口 Sticky MAC 功能，修改端口安全保护动作为

```
int g0/0/1
```

```
port-security mac-address sticky
```

```
port-security protect-action protect
```

```
dis mac-address sticky
```



```
[SW1]display mac-address sticky
MAC address table of slot 0:
```

MAC Address	VLAN/ VSI/SI	PEVLAN	CEVLAN	Port	Type
5489-9814-1949	10	-	-	GE0/0/1	sticky
5489-98bf-740c	10	-	-	GE0/0/1	sticky

Total matching items on slot 0 displayed = 2

也可以手工添加

```
int g0/0/1
```

```
port-security mac-address sticky 5489-9800-0001 vlan
10
```

SW3 :

要 ping 一下，有数据通过交换机，再去查看

```
int g0/0/1
```

```
port-security enable
```

```
dis mac-address security
```

```
[SW3]display mac-address security
MAC address table of slot 0:
```

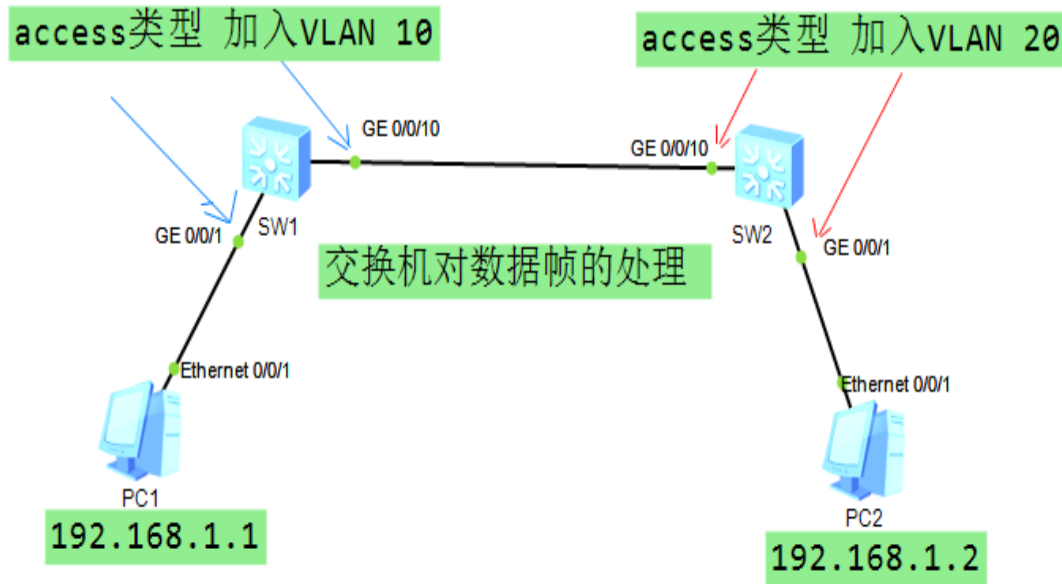
MAC Address	VLAN/ VSI/SI	PEVLAN	CEVLAN	Port	Type
5489-98ee-3e25	10	-	-	GE0/0/1	security

Total matching items on slot 0 displayed = 1

=====

## VLAN 实验 6：交换机对数据帧的处理

展示一个非常简单的场景，其中包含的问题却也难倒了不少初学者



### 基础配置

SW1：

un ter mo

sy

sys SW1

vlan 10

int g0/0/1

port link-type access

port default vlan 10

int g0/0/10

port link-type access

port default vlan 10

q

```
SW2 :  
un ter mo  
sy  
sys SW2  
vlan 20  
int g0/0/1  
port link-type access  
port default vlan 20  
int g0/0/10  
port link-type access  
port default vlan 20  
q
```

PC1 与 PC2 之间可以实现二层通信吗？答案是可以  
为什么？

现在假设 PC1 发送一个数据帧给 PC2，这个数据帧从 PC1 的网卡送时显然 是一个无标记帧，进入 SW1 的 g0/0/1 接口被打上 tag vlan 10，数据帧从 g0/0/10 接口发出，由于这个接口为 access 类型，而且最重要的是这个接口也加入了 vlan 10，因此数据帧的 tag 被剥除然后再从该接口发出。

SW2 的 g0/0/10 接口上收到这个数据帧，该帧进入交换机后被打上 tag vlan 20，数据帧从 g0/0/1 口发出，此时数据帧的 tag 被剥除。最终，PC1 发送出来数据帧是能够到达 PC2 的。

如果 g0/0/10 改为 trunk，是不能通信的。

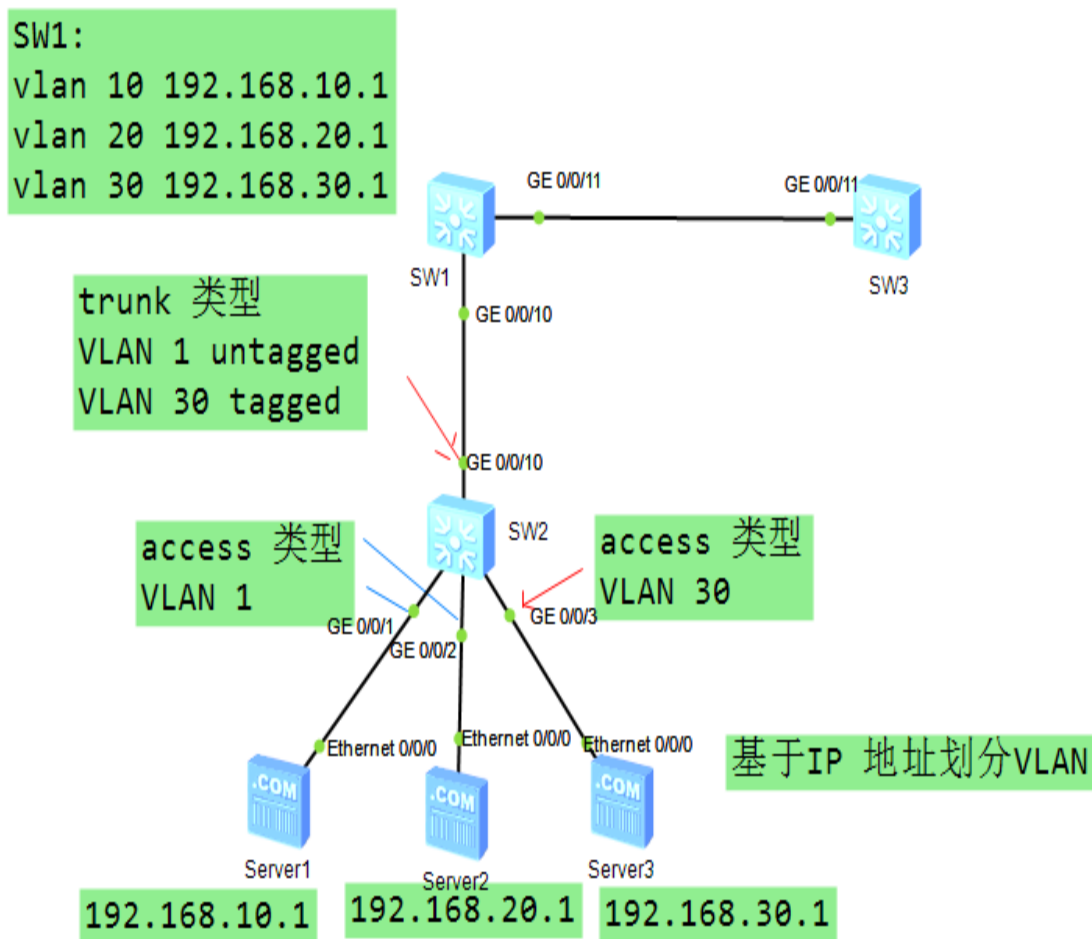
=====

## **VLAN 实验 7：基于 IP 地址划分 VLAN**

通常我们多采用基于接口划分 VLAN 的方式，也就是通过命令交换机的接口加入某个或者某些特定的 VLAN，当然也可以基于 IP 地址划分 VLAN。

SW2 下挂着三台服务器，server 1 和 server 2 的接口是 access 类型，而且加入了 VLAN 1，server 3 的接口也是 access 类型，但是加入 vlan 30. 上联接口是 trunk 类型，在允许通过的 vlan id 列表中添加了 vlan 1 ,vlan 30 .

现在网络的需求是，将来自这 3 个网段的数据帧在 SW1 上根据规划到相应的 VLAN，并且 SW1 转发这些帧给 SW3 时都携带 tag.



基本配置

```
SW2 :
un ter mo
sy
sys SW2
vlan batch 1 30
int g0/0/1
port link-type access
port default vlan 1
int g0/0/2
port link-type access
port default vlan 1
int g0/0/3
port link-type access
port default vlan 30
int g0/0/10
port link-type trunk
port trunk allow-pass vlan 30
q
```

在 SW1 上进行基本地 IP 地址划分 VLAN 的配置，配置 IP 网段与 VLAN 对应关系

```
SW1 :
un ter mo
sy
sys SW1
vlan batch 10 20 30

vlan 10
ip-subnet-vlan ip 192.168.10.0 24
vlan 20
ip-subnet-vlan ip 192.168.20.0 24

int g0/0/10
```

```
port link-type hybrid
port hybrid untagged vlan 10 20
port hybrid tagged vlan 30
ip-subnet-vlan enable
```

```
int g0/0/11
port link-type trunk
port trunk allow-pass vlan 10 20 30
q
```